

# **JEDEC PUBLICATION**

---

## **Part Model Electrical Guidelines for Electronic-Device Packages – XML Requirements**

---

### **JEP30-E100D**

(Revision of JEP30-E100C, November 2023)

**February 2024**

---

**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.

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](http://www.jedec.org) under Standards and Documents for alternative contact information.

Published by  
©JEDEC Solid State Technology Association 2024  
3103 North 10th Street  
Suite 240 South  
Arlington, VA 22201-2108

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**

Printed in the U.S.A.  
All rights reserved

DO NOT VIOLATE  
THE  
LAW!

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

Organizations may obtain permission to reproduce a limited number of copies  
through entering into a license agreement. For information, contact:

JEDEC Solid State Technology Association  
3103 North 10th Street  
Suite 240 South  
Arlington, VA 22201-2107  
<https://www.jedec.org/contact>





## PART MODEL ELECTRICAL GUIDELINE 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
3	Requirements .....	6
3.1.	TERMS AND DEFINITIONS .....	6
3.2.	XML SCHEMA KEY TERMS AND DEFINITIONS .....	6
4	Part Model Schema Definition .....	7
4.1.	PART MODEL -> ELECTRICAL SECTION .....	7
4.2.	MANUFACTURER PART NUMBER-ARRAY .....	8
4.3.	LINKING THE MANUFACTURING PART NUMBER TO A SPECIFIC ELECTRICAL DATA SET .....	9
4.3.1.	Linking the Manufacturing Part Number to Electrical Parameters .....	11
4.3.2.	Linking the Manufacturing Part Number to Part Classification .....	12
4.3.3.	Linking the Manufacturing Part Number to Terminal Details .....	13
4.3.4.	Linking the Manufacturing Part Number to Functions .....	14
4.3.5.	Linking the Manufacturing Part Number to Electrical Specification .....	15
4.3.6.	Linking the Manufacturing Part Number to Truth Table .....	16
4.3.7.	Linking the Manufacturing Part Number to ESD .....	17
4.3.8.	Linking the Manufacturing Part Number to Schematic Data .....	18
4.3.9.	Linking the Manufacturing Part Number to Symbol .....	19
4.3.10.	Linking the Manufacturing Part Number to Required Circuitry .....	20
4.3.11.	Linking the Manufacturing Part Number to Mapping data .....	21
4.3.12.	Linking the Manufacturing Part Number to Electrical Map data .....	22
4.3.13.	Linking the Manufacturing Part Number to Package Terminal Map data .....	23
4.3.14.	Linking the Manufacturing Part Number to Terminal Map data .....	24
4.3.15.	Linking the Manufacturing Part Number to Simulation Map data .....	25
4.3.16.	Linking the Manufacturing Part Number to Simulation Models .....	26

4.3.17.	Linking the Manufacturing Part Number to Reference Design .....	27
4.3.18.	Linking the Manufacturing Part Number to Software Interface Description .....	28
4.4.	ELECTRICAL SECTION .....	29
4.5.	ELECTRICAL PARAMETERS .....	30
4.5.1.	Part Classification Array .....	31
4.5.1.1.	Cable and Wiring Classification .....	34
4.5.1.1.1.	Cable and Wiring Classification Property-Array .....	34
4.5.1.1.2.	Other Cable and Wiring Classification .....	35
4.5.1.2.	Connector Classification .....	36
4.5.1.2.1.	Board-to-Board Classification .....	37
4.5.1.2.1.1.	Backplane Classification .....	37
4.5.1.2.1.1.1.	Board-to-Board Backplane Classification Property-Array .....	38
4.5.1.2.2.	Cable-to-Board Classification .....	39
4.5.1.2.2.1.	Audio Video Display Cable-to-Board Connector Classification .....	40
4.5.1.2.2.1.1.	D-Sub Connector Classification .....	40
4.5.1.2.2.1.2.	Display Port Connector Classification .....	41
4.5.1.2.2.1.3.	USB Connector Classification .....	41
4.5.1.2.2.1.4.	HDMI Connector Classification .....	41
4.5.1.2.2.1.5.	Serial Cable Connector Classification .....	42
4.5.1.2.2.1.6.	VGA Connector Classification .....	42
4.5.1.2.2.1.7.	Audio Video Display Cable-to-Board Other Connector Classification .....	43
4.5.1.2.2.2.	Cable-to-Board Backplane Connector Classification .....	43
4.5.1.2.2.2.1.	Cable-to-Board Backplane Connector Classification Property-Array .....	44
4.5.1.2.2.3.	Cable-to-Board PowerAndSignal Connector Classification .....	45
4.5.1.2.2.3.1.	Cable-to-Board Power Connector Classification .....	45
4.5.1.2.2.4.	Cable-to-Board Signal Connector Classification .....	46
4.5.1.2.2.5.	Cable-to-Board Other Connector Classification .....	46
4.5.1.2.3.	Cable-to-Cable Classification .....	47
4.5.1.2.3.1.	Cable-to-Cable PowerAndSignal Connector Classification .....	47
4.5.1.2.3.2.	Cable-to-Cable Power Connector Classification .....	48
4.5.1.2.3.3.	Cable-to-Cable Signal Connector Classification .....	48
4.5.1.2.3.4.	Other Cable-to-Cable Connector Classification .....	49
4.5.1.2.4.	CardEdge-to-Board Classification .....	49
4.5.1.2.5.	CardEdge-to-Cable Classification .....	50
4.5.1.2.5.1.	CardEdge-to-Cable Interface Connector Classification .....	50
4.5.1.2.5.2.	CardEdge-to-Cable Other Connector Classification .....	51
4.5.1.2.6.	Optical Classification .....	51
4.5.1.2.7.	Socket Classification .....	52
4.5.1.2.7.1.	LIF Socket Classification .....	52
4.5.1.2.7.2.	ZIF Socket Classification .....	53
4.5.1.2.7.3.	Other PCB Mountable Socket Classification .....	53
4.5.1.2.8.	Other Connector Classification .....	54
4.5.1.3.	Electrical Classification .....	55
4.5.1.3.1.	Amplifier Classification .....	56
4.5.1.3.1.1.	Other Amplifier Classification .....	57
4.5.1.3.2.	Audio Classification .....	58
4.5.1.3.3.	Battery Classification .....	59
4.5.1.3.3.1.	Rechargeable Battery Classification .....	59
4.5.1.3.3.1.1.	Rechargeable Battery Material Property .....	60
4.5.1.3.3.2.	Non-rechargeable Battery Classification .....	61
4.5.1.3.3.2.1.	Non-rechargeable Battery Material Property .....	62
4.5.1.3.4.	Capacitor Classification .....	63
4.5.1.3.4.1.	Ceramic Capacitor Classification and Property-Array .....	64
4.5.1.3.4.1.1.	Ceramic Capacitor Class Property .....	65

4.5.1.3.4.1.2.	Ceramic Capacitor Dielectric Property .....	65
4.5.1.3.4.2.	Electrolytic Capacitor Classification and Property-Array .....	66
4.5.1.3.4.2.1.	Electrolytic Capacitor Material Property .....	66
4.5.1.3.4.2.2.	Electrolytic Capacitor Electrolyte Property .....	67
4.5.1.3.4.3.	Film Capacitor Classification and Property-Array .....	68
4.5.1.3.4.3.1.	Film Capacitor Dielectric Property .....	68
4.5.1.3.4.4.	Silicon Capacitor Classification .....	69
4.5.1.3.4.5.	Super-cap Capacitor Classification .....	69
4.5.1.3.4.5.1.	Super-cap Capacitor Class Property .....	70
4.5.1.3.4.6.	Other Fixed Capacitor Classification and Property-Array .....	70
4.5.1.3.4.7.	Variable Capacitor Classification and Property-Array .....	71
4.5.1.3.5.	Circuit Protection Classification .....	71
4.5.1.3.5.1.	Over Current Circuit Protection Classification .....	72
4.5.1.3.5.1.1.	Resettable Over Current Circuit Protection Classification .....	72
4.5.1.3.5.1.2.	Resettable Over Current Circuit Protection Technology Property .....	73
4.5.1.3.5.1.3.	Non-Resettable Over Current Circuit Protection Classification .....	73
4.5.1.3.5.1.4.	Non-Resettable Over Current Circuit Protection Technology Property .....	74
4.5.1.3.5.2.	Over Voltage Circuit Protection Classification .....	74
4.5.1.3.5.2.1.	Over Voltage Circuit Protection Varistor Classification .....	75
4.5.1.3.5.2.2.	Over Voltage Circuit Protection Varistor Type Property .....	75
4.5.1.3.5.2.3.	Over Voltage Circuit Protection Classification .....	76
4.5.1.3.5.3.	Over Temperature Circuit Protection Classification .....	76
4.5.1.3.5.4.	Other Circuit Protection Classification .....	77
4.5.1.3.6.	Data Converter Classification .....	77
4.5.1.3.6.1.	Data Converter Classification .....	78
4.5.1.3.7.	Diode Classification .....	79
4.5.1.3.7.1.	LED Classification .....	80
4.5.1.3.7.2.	Rectifier Diode Classification .....	81
4.5.1.3.7.3.	Schottky Diode Classification .....	81
4.5.1.3.7.4.	Silicon Carbide Diode Classification .....	82
4.5.1.3.7.5.	ESD Diode Classification .....	82
4.5.1.3.7.6.	Signal Diode Classification .....	83
4.5.1.3.7.7.	Other Diode Classification .....	83
4.5.1.3.8.	Filter Classification .....	84
4.5.1.3.8.1.	EMI-RFI Filter Classification .....	85
4.5.1.3.8.2.	Other Filter Classification .....	85
4.5.1.3.9.	Frequency Source Classification .....	86
4.5.1.3.10.	IC Classification .....	87
4.5.1.3.11.	Inductor Classification .....	88
4.5.1.3.11.1.	Ferrite Inductor Classification .....	89
4.5.1.3.11.1.1.	Variable Ferrite Inductor Property-Array .....	89
4.5.1.3.11.1.2.	Ferrite Inductor Property-Array .....	90
4.5.1.3.11.2.	MetalAlloys Inductor Classification .....	90
4.5.1.3.11.2.1.	MetalAlloys Inductor Classification .....	91
4.5.1.3.11.3.	Other Inductor Classification .....	91
4.5.1.3.11.4.	Inductor Property-Array .....	92
4.5.1.3.11.4.1.	Inductor Shielding Property .....	92
4.5.1.3.11.4.2.	Inductor Technology Property .....	93
4.5.1.3.12.	Memory Classification .....	94
4.5.1.3.12.1.	Volatile Memory Classification .....	95
4.5.1.3.12.2.	NonVolatile Memory Classification .....	96
4.5.1.3.13.	Optoelectronics Classification .....	97
4.5.1.3.13.1.	Display Classification .....	98
4.5.1.3.13.1.1.	LED Display Classification .....	98

4.5.1.3.13.1.2.	Organic Display Classification .....	99
4.5.1.3.13.1.3.	Other Display Classification.....	99
4.5.1.3.13.2.	Photoemitter Classification.....	100
4.5.1.3.13.2.1.	Photoemitter LED Classification.....	100
4.5.1.3.13.2.2.	Other Display Classification.....	101
4.5.1.3.13.3.	Photosensitive Classification.....	101
4.5.1.3.13.3.1.	Photodiode Photosensitive Classification .....	102
4.5.1.3.13.3.2.	Phototransistor Photosensitive Classification.....	102
4.5.1.3.13.3.3.	Other Photosensitive Classification .....	102
4.5.1.3.13.4.	Optocoupler Classification .....	103
4.5.1.3.13.4.1.	Photodiode Optocoupler Classification .....	103
4.5.1.3.13.4.2.	Phototransistor Optocoupler Classification .....	104
4.5.1.3.13.4.3.	Other Optocoupler Classification.....	104
4.5.1.3.14.	Regulator Classification.....	105
4.5.1.3.14.1.	Linear Current Regulator Classification.....	105
4.5.1.3.14.2.	Linear Current Regulator Property-Array.....	106
4.5.1.3.14.3.	Linear Voltage Regulator Classification.....	106
4.5.1.3.14.4.	Linear Voltage Regulator Property-Array.....	107
4.5.1.3.14.5.	Switching Regulator Classification .....	107
4.5.1.3.14.5.1.	Isolated Switching Regulator Classification.....	108
4.5.1.3.14.5.2.	Other IsolatedSwitching Regulator Classification .....	108
4.5.1.3.14.5.3.	Non-isolated Switching Regulator Classification.....	109
4.5.1.3.14.5.4.	Other Non-IsolatedSwitching Regulator Classification .....	109
4.5.1.3.14.6.	Other Regulator Classification .....	110
4.5.1.3.15.	Relay Classification.....	110
4.5.1.3.15.1.	ElectroMechanical Relay Classification and Property-Array .....	111
4.5.1.3.15.2.	SolidState Relay Classification and Property-Array.....	112
4.5.1.3.16.	Resistor Classification .....	112
4.5.1.3.16.1.	Fixed Resistor Classification and Property-Array .....	113
4.5.1.3.16.2.	Adjustable Resistor Classification .....	114
4.5.1.3.16.2.1.	Other Adjustable Resistor Classification .....	114
4.5.1.3.16.3.	Non-Linear Resistor Classification.....	115
4.5.1.3.16.4.	Thermistor Classification and Property-Array.....	116
4.5.1.3.16.5.	Other Adjustable Resistor Classification .....	116
4.5.1.3.17.	RF Classification .....	117
4.5.1.3.17.1.	RF Detector Classification .....	118
4.5.1.3.17.2.	RF Mixer Classification .....	118
4.5.1.3.17.3.	Other RF Classification .....	119
4.5.1.3.18.	Sensor Classification.....	120
4.5.1.3.18.1.	Other Sensor Classification .....	121
4.5.1.3.19.	Switch Classification.....	122
4.5.1.3.19.1.	Other Switch Classification .....	123
4.5.1.3.19.2.	Switch Property-Array.....	124
4.5.1.3.20.	Thyristor Classification.....	125
4.5.1.3.20.1.	Triode Thyristor Classification.....	126
4.5.1.3.20.2.	Other Thyristor Classification.....	126
4.5.1.3.20.3.	Thyristor Property-Array.....	127
4.5.1.3.21.	Transformer Classification .....	128
4.5.1.3.21.1.	Other Transformer Classification .....	128
4.5.1.3.22.	Transistor Classification .....	129
4.5.1.3.22.1.	Bipolar Junction Transistor Classification and Property-Array.....	130
4.5.1.3.22.2.	Unijunction Transistor Classification and Property-Array.....	131
4.5.1.3.22.3.	Programmable Unijunction Transistor Classification and Property-Array .....	132
4.5.1.3.22.4.	Field Effect Transistor Classification and Property-Array.....	133

4.5.1.3.22.5.	Insulated Gate Bipolar Transistor Classification and Property-Array.....	134
4.5.1.3.22.6.	Other Transistor Classification and Property-Array .....	135
4.5.1.3.23.	Tube Classification .....	135
4.5.1.3.24.	Other Electrical Classification.....	136
4.5.1.4.	Hardware Classification .....	137
4.5.1.4.1.	MetalCage Classification .....	137
4.5.1.4.2.	Other Hardware Classification .....	138
4.5.1.5.	Optics Classification .....	139
4.5.1.5.1.	Other Optics Classification .....	140
4.5.2.	Terminal Details - Array .....	140
4.5.2.1.	Properties - Array.....	141
4.5.2.1.1.	Linking the Terminal Map and the Internal Node to the Property ID .....	145
4.5.2.1.2.	Analog and Power Connection Type .....	146
4.5.2.1.3.	Signal Direction .....	147
4.5.2.1.4.	Internal Pullup / Pulldown .....	148
4.5.2.1.5.	Series Component.....	149
4.5.2.1.6.	Output Circuit.....	150
4.5.2.1.6.1.	Bipolar .....	150
4.5.2.1.6.1.1.	Passive – Pullup.....	151
4.5.2.1.6.1.2.	Passive - Pulldown.....	152
4.5.2.1.6.1.3.	Totem - Pole.....	153
4.5.2.1.6.1.4.	Push - Pull.....	154
4.5.2.1.6.2.	Unipolar .....	157
4.5.2.1.6.3.	Programmable .....	159
4.5.2.1.7.	Reference .....	159
4.5.2.2.	Terminal Function – Array.....	163
4.5.2.2.1.	Digital Function – Signal .....	164
4.5.2.2.2.	Digital Function – Terminal State .....	165
4.5.2.3.	Terminal Grouping – Array.....	166
4.5.2.3.1.	Same Potential Group - Array .....	167
4.5.2.3.2.	Terminal Swap - Array.....	168
4.5.2.3.3.	Function Swap - Array .....	169
4.5.2.3.4.	Internal Electrical Connection - Array .....	171
4.5.2.3.5.	Differential Pair - Array .....	172
4.5.2.3.6.	Logical Group - Array.....	175
4.5.2.3.7.	Terminal-to-Terminal Signal Path – Array.....	178
4.5.2.3.7.1.	Uni-directional Terminal-to-Terminal Signal Path .....	179
4.5.2.3.7.2.	BI-directional Terminal-to-Terminal Signal Path .....	182
4.5.2.4.	External Connection - Array .....	185
4.5.2.4.1.	Decoupling .....	186
4.5.2.4.2.	Pullup to Power.....	187
4.5.2.4.3.	Pulldown to Ground.....	188
4.5.3.	Function Group - Array .....	189
4.5.3.1.	Super Interface - Array.....	193
4.5.3.2.	Audio .....	194
4.5.3.2.1.	Analog Microphone.....	194
4.5.3.2.2.	Digital Microphone .....	195
4.5.3.2.3.	Speaker .....	195
4.5.3.3.	Amplifier.....	196
4.5.3.3.1.	Differential Input Differential Output .....	197
4.5.3.3.2.	Differential Input Single Ended Output.....	198
4.5.3.3.3.	Single Ended Input Differential Output.....	199
4.5.3.3.4.	Single Ended Input Single Ended Output .....	200
4.5.3.4.	Capacitor .....	201

4.5.3.4.1.	Fixed Non Polarized .....	201
4.5.3.4.2.	Fixed Polarized .....	202
4.5.3.4.3.	Differential Non Polarized.....	202
4.5.3.4.4.	Feed Through .....	203
4.5.3.4.5.	Variable Non Polarized.....	203
4.5.3.5.	Diode .....	204
4.5.3.5.1.	Diode Standard Terminal Name Assignment Type .....	205
4.5.3.6.	Filter .....	206
4.5.3.7.	Frequency Source.....	206
4.5.3.7.1.	Generator .....	207
4.5.3.7.2.	Timer .....	208
4.5.3.7.3.	Crystal .....	209
4.5.3.7.4.	Oscillator .....	210
4.5.3.7.5.	Resonator .....	211
4.5.3.7.6.	Voltage Controlled Oscillator .....	212
4.5.3.8.	Fuse .....	213
4.5.3.9.	Inductor.....	214
4.5.3.9.1.	Basic Inductor Standard Terminal Name Assignment.....	215
4.5.3.9.2.	Coupled Inductor Standard Terminal Name Assignment.....	215
4.5.3.10.	Interface .....	216
4.5.3.10.1.	AIB .....	218
4.5.3.10.1.1.	AIB-Base .....	218
4.5.3.10.1.2.	AIB-Plus .....	219
4.5.3.10.2.	Battery Interface Function .....	219
4.5.3.10.3.	Camera Interface Function.....	220
4.5.3.10.3.1.	CSI .....	221
4.5.3.10.3.2.	CSI-2 C-PHY-x1.....	221
4.5.3.10.3.3.	CSI-2 C-PHY-x2.....	222
4.5.3.10.3.4.	CSI-2 C-PHY-x3.....	222
4.5.3.10.3.5.	CSI-2 C-PHY-x4.....	223
4.5.3.10.3.6.	CSI-2 C-PHY-x5.....	223
4.5.3.10.3.7.	CSI-2 C-PHY-x6.....	224
4.5.3.10.3.8.	CSI-2 D-PHY-x1 .....	224
4.5.3.10.3.9.	CSI-2 D-PHY-x2 .....	225
4.5.3.10.3.10.	CSI-2 D-PHY-x3 .....	225
4.5.3.10.3.11.	CSI-2 D-PHY-x4 .....	226
4.5.3.10.3.12.	CSI-2 D-PHY-x5 .....	226
4.5.3.10.3.13.	CSI-2 D-PHY-x6 .....	227
4.5.3.10.3.14.	CSI-2 D-PHY-x7 .....	227
4.5.3.10.3.15.	CSI-2 D-PHY-x8 .....	228
4.5.3.10.3.16.	CSI-3 .....	228
4.5.3.10.3.1.	CPI .....	229
4.5.3.10.4.	Compute Express Link Function .....	229
4.5.3.10.4.1.	CXL-x4.....	230
4.5.3.10.4.2.	CXL-x8.....	231
4.5.3.10.4.3.	CXL-x16.....	232
4.5.3.10.5.	Display Bus Function .....	233
4.5.3.10.5.1.	DBI-TypeA .....	234
4.5.3.10.5.2.	DBI-TypeB.....	235
4.5.3.10.5.3.	DBI-TypeC.....	236
4.5.3.10.5.4.	DPI-Type1 .....	236
4.5.3.10.5.5.	DPI-Type2-3.....	237
4.5.3.10.5.6.	DPI-Type4.....	238
4.5.3.10.5.7.	DSI-2-OptionC .....	239

4.5.3.10.5.8.	DSI-2-OptionD .....	240
4.5.3.10.6.	DDR3 Interface Function .....	241
4.5.3.10.6.1.	DDR3-x4 .....	242
4.5.3.10.6.2.	DDR3-x4 – Dual Die .....	243
4.5.3.10.6.3.	DDR3-x4 – Quad Die.....	244
4.5.3.10.6.4.	DDR3-x8 .....	245
4.5.3.10.6.5.	DDR3-x8 – Dual Die .....	246
4.5.3.10.6.6.	DDR3-x8 – Quad Die.....	247
4.5.3.10.6.7.	DDR3-x16 .....	248
4.5.3.10.6.8.	DDR3-x16 – Dual Die .....	249
4.5.3.10.6.9.	DDR3-x16 – Quad Die.....	250
4.5.3.10.6.10.	DDR3 Controller .....	251
4.5.3.10.7.	DDR4 Interface Function .....	253
4.5.3.10.7.1.	DDR4-x4 .....	254
4.5.3.10.7.2.	DDR4-x4 – Dual Die .....	255
4.5.3.10.7.3.	DDR4-x8 .....	256
4.5.3.10.7.4.	DDR4-x8 Dual Die .....	257
4.5.3.10.7.5.	DDR4-x16 .....	258
4.5.3.10.7.6.	DDR4-x16 – Dual Die .....	259
4.5.3.10.7.7.	DDR4-x32 .....	260
4.5.3.10.7.8.	DDR4-x72 .....	261
4.5.3.10.7.9.	LPDDR4 – Single Channel .....	263
4.5.3.10.7.10.	LPDDR4 – Dual Channel .....	264
4.5.3.10.7.11.	DDR4DB02.....	265
4.5.3.10.7.12.	DDR4-NVDIMM-N .....	266
4.5.3.10.7.13.	DDR4RCD02.....	268
4.5.3.10.8.	DDR5 Interface Function .....	269
4.5.3.10.8.1.	DDR5-x4 .....	270
4.5.3.10.8.2.	DDR5-x8 .....	271
4.5.3.10.8.3.	DDR5-x16 .....	272
4.5.3.10.8.4.	DDR5DB01.....	273
4.5.3.10.8.5.	LPDDR5.....	274
4.5.3.10.8.6.	GDDR5.....	275
4.5.3.10.8.7.	GDDR5X.....	276
4.5.3.10.9.	DDR6 Interface Function .....	277
4.5.3.10.9.1.	GDDR6.....	277
4.5.3.10.10.	DigRF3G Interface Function .....	278
4.5.3.10.11.	DigRFv4 Interface Function .....	278
4.5.3.10.12.	EE1002-SPD-EEPROM.....	279
4.5.3.10.13.	Embedded Display Port Interface .....	280
4.5.3.10.14.	Ethernet .....	281
4.5.3.10.14.1.	Ethernet1000BASE-LX .....	281
4.5.3.10.14.2.	Ethernet1000BASE-SX.....	282
4.5.3.10.14.3.	Ethernet1000BASE-CX.....	283
4.5.3.10.15.	eTrak Interface .....	283
4.5.3.10.16.	FC-PI-6 Interface Function.....	284
4.5.3.10.17.	HBM Interface Function .....	284
4.5.3.10.17.1.	HBM1, HBM2 and HBM2E Interface Functions.....	285
4.5.3.10.17.2.	HBM3 Interface Functions .....	296
4.5.3.10.18.	HDMI Interface Function.....	308
4.5.3.10.18.1.	HDMI-TypeA.....	309
4.5.3.10.18.2.	HDMI-TypeB.....	309
4.5.3.10.18.3.	HDMI-TypeC.....	310
4.5.3.10.18.4.	HDMI-Source.....	310

4.5.3.10.18.5.	HDMI-Sink .....	311
4.5.3.10.19.	HSI Interface.....	312
4.5.3.10.20.	HTI Interface Function.....	312
4.5.3.10.21.	HTlv1 Interface Function.....	313
4.5.3.10.22.	I2C Interface Function.....	313
4.5.3.10.23.	I3C Interface Function.....	314
4.5.3.10.24.	LLI-Serial Interface Function.....	314
4.5.3.10.25.	LVSTL06 Interface Function.....	315
4.5.3.10.26.	MMC Interface Function .....	315
4.5.3.10.26.1.	eMMC Interface Function .....	316
4.5.3.10.26.2.	MMC Interface Function .....	316
4.5.3.10.26.3.	SPI-Mode Interface Function .....	317
4.5.3.10.27.	MII Interface Function.....	318
4.5.3.10.27.1.	CGMII .....	319
4.5.3.10.27.2.	GMII .....	321
4.5.3.10.27.3.	MII .....	322
4.5.3.10.27.4.	RGMI .....	323
4.5.3.10.27.5.	RMII .....	324
4.5.3.10.27.6.	SMII .....	325
4.5.3.10.27.7.	XGMII .....	326
4.5.3.10.27.8.	XLGMII .....	327
4.5.3.10.28.	OIF-CEI-04.0 Interface Function .....	328
4.5.3.10.29.	PCIe Interface Function.....	329
4.5.3.10.29.1.	PCIe-x1 .....	330
4.5.3.10.29.2.	PCIe-x2 .....	331
4.5.3.10.29.3.	PCIe-x4 .....	332
4.5.3.10.29.4.	PCIe-x8 .....	333
4.5.3.10.29.5.	PCIe-x16 .....	334
4.5.3.10.29.6.	PCIe-x32 .....	335
4.5.3.10.29.7.	SFF-8639Connector .....	337
4.5.3.10.29.8.	ATX-PowerConnector150W .....	338
4.5.3.10.29.9.	AuxiliaryPowerConnector2x4.....	338
4.5.3.10.30.	Cabling PCIe Interface Function .....	339
4.5.3.10.30.1.	CablingPCIe-x1.....	339
4.5.3.10.30.2.	CablingPCIe-x4.....	340
4.5.3.10.30.3.	CablingPCIe-x8.....	341
4.5.3.10.30.4.	CablingPCIe-x16 .....	342
4.5.3.10.31.	A-PHY .....	343
4.5.3.10.32.	BoW-PHY.....	343
4.5.3.10.33.	C-PHY Interface Function .....	344
4.5.3.10.33.1.	C-PHY-x1.....	345
4.5.3.10.33.2.	C-PHY-x2.....	345
4.5.3.10.33.3.	C-PHY-x3.....	346
4.5.3.10.33.4.	C-PHY-x4.....	346
4.5.3.10.33.5.	C-PHY-x5.....	347
4.5.3.10.33.6.	C-PHY-x6.....	347
4.5.3.10.34.	D-PHY Interface Function .....	348
4.5.3.10.34.1.	D-PHY-x1 .....	349
4.5.3.10.34.2.	D-PHY-x2 .....	349
4.5.3.10.34.3.	D-PHY-x3 .....	350
4.5.3.10.34.4.	D-PHY-x4 .....	350
4.5.3.10.34.5.	D-PHY-x5 .....	351
4.5.3.10.34.6.	D-PHY-x6 .....	351
4.5.3.10.34.7.	D-PHY-x7 .....	352



4.5.3.10.34.8.	D-PHY-x8 .....	352
4.5.3.10.35.	M-PHY .....	353
4.5.3.10.36.	OpenHBI .....	354
4.5.3.10.37.	PTI Interface Function .....	355
4.5.3.10.38.	Radio Front End Interface Function .....	355
4.5.3.10.38.1.	RBDP.....	356
4.5.3.10.38.2.	RF-BB.....	357
4.5.3.10.39.	RFFE.....	357
4.5.3.10.40.	SD .....	358
4.5.3.10.41.	SD-UHS-II.....	358
4.5.3.10.42.	Serial Interface Function .....	359
4.5.3.10.42.1.	SI 359	
4.5.3.10.42.2.	SPI.....	360
4.5.3.10.42.3.	eSPI.....	360
4.5.3.10.42.4.	xSPI.....	361
4.5.3.10.42.5.	SPD5118Hub .....	361
4.5.3.10.43.	SLIMbus.....	362
4.5.3.10.44.	SMB .....	362
4.5.3.10.45.	SoundWire Interface Function .....	363
4.5.3.10.46.	SPMI .....	363
4.5.3.10.47.	UART .....	364
4.5.3.10.48.	UCle.....	364
4.5.3.10.48.1.	UCle-Advanced.....	365
4.5.3.10.48.2.	UCleStandard_x16 .....	366
4.5.3.10.48.3.	UCleStandard_x32 .....	367
4.5.3.10.49.	UniPro .....	368
4.5.3.10.50.	Universal Flash Storage Interface Function .....	368
4.5.3.10.50.1.	UFS .....	369
4.5.3.10.50.2.	UFSHCI.....	370
4.5.3.10.51.	USB Interface Function .....	370
4.5.3.10.51.1.	USB2.0.....	371
4.5.3.10.51.2.	USB3.0.....	372
4.5.3.10.51.3.	USB_Type-C_Receptacle .....	373
4.5.3.10.51.4.	USB_Type-C_Plug.....	373
4.5.3.10.51.5.	XFM .....	374
4.5.3.10.52.	Other Interface Standard .....	374
4.5.3.11.	Non Linear – Frequency Mixer .....	375
4.5.3.11.1.	Balanced.....	376
4.5.3.11.2.	Other Non Linear Function Standard .....	377
4.5.3.12.	Optoelectronics.....	377
4.5.3.12.1.	Photoemitter.....	378
4.5.3.12.1.1.	Infrared Emitting Diode.....	378
4.5.3.12.1.2.	LED .....	379
4.5.3.12.1.3.	Laser.....	379
4.5.3.12.2.	Photosensitive Device .....	380
4.5.3.12.2.1.	Photodiode.....	381
4.5.3.12.2.2.	Photothyristor .....	381
4.5.3.12.2.3.	Phototriac.....	382
4.5.3.12.2.4.	Phototransistor .....	383
4.5.3.12.2.5.	Photodarlington .....	383
4.5.3.12.2.6.	Photovoltaic Diode.....	384
4.5.3.12.3.	Optocoupler .....	384
4.5.3.12.3.1.	Photodiode.....	385
4.5.3.12.3.2.	Photothyristor .....	385

4.5.3.12.3.3.	Phototriac.....	386
4.5.3.12.3.4.	Phototransistor .....	387
4.5.3.12.3.5.	Photodarlington .....	388
4.5.3.13.	Relay.....	389
4.5.3.13.1.	Solid State Relay – Single Throw .....	390
4.5.3.13.2.	Solid State Relay – Double Throw .....	391
4.5.3.13.3.	Electromagnetic Relay – Single Throw .....	392
4.5.3.13.3.1.	Electromagnetic Relay – Single Throw – Contact Array .....	393
4.5.3.13.4.	Electromagnetic Relay – Double Throw .....	394
4.5.3.13.5.	Electromagnetic Relay – Double Throw – Contact Array .....	394
4.5.3.14.	Resistor .....	395
4.5.3.14.1.	Two Terminal Resistor Standard Terminal Name Assignment Type .....	396
4.5.3.14.2.	Potentiometer Function.....	397
4.5.3.15.	RF .....	399
4.5.3.15.1.	Antenna.....	400
4.5.3.15.2.	Attenuator.....	400
4.5.3.15.3.	Balun .....	401
4.5.3.15.4.	Circulator.....	401
4.5.3.15.5.	Coupler.....	402
4.5.3.15.5.1.	BiDirectional.....	402
4.5.3.15.5.2.	Directional.....	403
4.5.3.15.6.	DC Block .....	404
4.5.3.15.7.	Detector .....	405
4.5.3.15.8.	Divider .....	406
4.5.3.15.9.	Isolator .....	407
4.5.3.15.10.	Limiter .....	407
4.5.3.15.11.	Mixer .....	408
4.5.3.15.12.	Modulator .....	408
4.5.3.15.12.1.	IQ - Modulator Standard Terminal Name Assignment.....	409
4.5.3.15.12.1.1.	I-SingleDifferentialConnectionType .....	410
4.5.3.15.12.1.2.	Q-SingleDifferentialConnectionType .....	411
4.5.3.15.12.2.	QPSK - Modulator Standard Terminal Name Assignment.....	412
4.5.3.15.13.	Demodulator .....	412
4.5.3.15.14.	Multiplier .....	413
4.5.3.15.15.	Phase Detector.....	413
4.5.3.15.16.	Phase Shifter .....	414
4.5.3.16.	Source .....	414
4.5.3.16.1.	Battery Source.....	415
4.5.3.16.2.	Generic Voltage Source.....	415
4.5.3.16.3.	Generic Current Source.....	416
4.5.3.16.3.1.	AC-Source Type .....	416
4.5.3.16.3.2.	DC-Source Type .....	417
4.5.3.17.	Switch.....	417
4.5.3.17.1.	Single Throw.....	418
4.5.3.17.2.	Double Throw.....	418
4.5.3.17.3.	Many Throw .....	419
4.5.3.18.	Thyristor .....	420
4.5.3.18.1.	Bidirectional Thyristor Diode .....	420
4.5.3.18.2.	Bidirectional Thyristor Triode .....	421
4.5.3.18.3.	Unidirectional.....	421
4.5.3.18.3.1.	Reverse Blocking Diode.....	422
4.5.3.18.3.2.	Reverse Blocking Triode .....	422
4.5.3.18.3.3.	Reverse Conducting Diode .....	423
4.5.3.18.3.4.	Reverse Conducting Triode .....	423

4.5.3.19.	Transformer .....	424
4.5.3.19.1.	Transformer Coil Mapping .....	425
4.5.3.20.	Transistor .....	425
4.5.3.20.1.	Bipolar Junction Transistor .....	426
4.5.3.20.1.1.	Bipolar Transistor Standard Terminal Name Assignment .....	427
4.5.3.20.1.2.	Darlington Transistor Standard Terminal Name Assignment .....	428
4.5.3.20.2.	Unijunction Transistor.....	429
4.5.3.20.2.1.	N-type Unijunction Transistor.....	429
4.5.3.20.2.2.	P-type Unijunction Transistor .....	430
4.5.3.20.3.	Field Effect Transistor (FET) .....	431
4.5.3.20.3.1.	FET – Standard Terminal Name Assignment .....	432
4.5.3.20.4.	Insulated Gate Bipolar Transistor (IGBT).....	433
4.5.3.20.4.1.	IGBT – Standard Terminal Name Assignment Type .....	434
4.5.3.20.5.	Programmable Unijunction Transistor.....	434
4.5.3.20.5.1.	Terminal Mapping.....	435
4.5.3.21.	Other Standard Electrical Functions .....	435
4.5.4.	Electrical Specification .....	436
4.5.4.1.	Test Condition .....	437
4.5.4.1.1.	Units.....	438
4.5.4.1.1.1.	Complex UOM .....	440
4.5.4.2.	Parameter Set .....	441
4.5.4.2.1.	Parameter .....	442
4.5.4.3.	Parameter Graph .....	443
4.5.4.3.1.	Formatting .....	444
4.5.4.3.1.1.	Linking the Data-Array to the Appropriate Parameter Definition .....	447
4.5.4.3.2.	Data-Array.....	449
4.5.5.	Truth Table.....	458
4.5.6.	ESD .....	460
4.5.6.1.	HBM – Human Body Model .....	461
4.5.6.2.	CDM – Charged Device Model .....	461
4.6.	SCHEMATIC DATA - ARRAY.....	462
4.6.1.	Symbol - Array.....	462
4.6.1.1.	Symbol Graphics – Array.....	463
4.6.1.1.1.	Part Symbol Version.....	464
4.6.1.1.2.	Part Reference Designation .....	465
4.6.1.1.3.	Fracture – Array .....	466
4.6.1.1.3.1.	Symbol Name .....	466
4.6.1.1.3.1.1.	Function Symbol Version .....	467
4.6.1.1.3.2.	Function Reference Designation .....	467
4.6.1.1.3.3.	Graphical Representation .....	468
4.6.1.1.3.3.1.	Body .....	469
4.6.1.1.3.3.1.1.	Shape.....	470
4.6.1.1.3.3.1.1.1.	Vertex – Array .....	471
4.6.1.1.3.3.1.1.2.	Vertex.....	471
4.6.1.1.3.3.1.1.3.	Edge.....	472
4.6.1.1.3.3.1.1.4.	Line.....	473
4.6.1.1.3.3.1.1.5.	Arc .....	474
4.6.1.1.3.3.1.1.6.	Elliptical Arc.....	475
4.6.1.1.3.3.1.1.7.	Primitive-Shape .....	476
4.6.1.1.3.3.1.1.8.	Shape Origin.....	477
4.6.1.1.3.3.1.1.9.	Transformation .....	478
4.6.1.1.3.3.1.2.	SVG-Shape.....	479
4.6.1.1.3.3.1.3.	Image .....	480
4.6.1.1.3.3.1.4.	Shape Text.....	481

4.6.1.1.3.3.2.	Graphical Format – Array .....	482
4.6.1.1.3.3.2.1.	Graphical Format .....	483
4.6.1.1.3.3.2.2.	Default Graphical Format.....	484
4.6.1.1.3.3.3.	Attribute – Array .....	485
4.6.1.1.3.3.3.1.	Text Variant for One of the Views – Array .....	486
4.6.1.1.3.3.4.	Text Format – Array .....	487
4.6.1.1.3.3.4.1.	Text Format .....	487
4.6.1.1.3.3.4.2.	Default Text Format .....	488
4.6.1.1.3.3.5.	Terminal – Array.....	489
4.6.1.1.3.3.5.1.	Location.....	491
4.6.1.1.3.3.5.2.	Label.....	492
4.6.1.1.3.3.5.3.	Function Graphics .....	493
4.6.2.	Required Circuitry - Array .....	494
4.6.2.1.	Net - Array.....	495
4.7.	MAPPING - ARRAY .....	497
4.7.1.	Electrical Map – Array .....	498
4.7.1.1.	Operational Mode.....	501
4.7.1.1.1.	Functional Operational Mode .....	502
4.7.1.1.2.	Power Operational Mode.....	503
4.7.1.1.3.	Load Operational Mode .....	503
4.7.1.1.4.	Test Operational Mode .....	504
4.7.2.	Package Terminal Map.....	505
4.7.3.	Die Terminal Map.....	508
4.7.4.	Simulation Map .....	508
4.8.	SIMULATION MODEL - ARRAY.....	509
4.9.	REFERENCE DESIGN - ARRAY .....	510
4.10.	SOFTWARE INTERFACE DESCRIPTION - ARRAY .....	511
4.10.1.	IP-XACT .....	511
4.10.2.	System RDL.....	512
5	Quick TeX reference.....	513
5.1.	SYNTAX .....	513
5.2.	SYMBOLS.....	513
6	Rule Syntax .....	514
6.1.	RULE APPLIED TO A LOGICAL-GROUP.....	515
6.2.	RULE APPLIED TO A TERMINAL MAP.....	516
6.3.	RULE FUNCTIONS .....	517
6.3.1.	Arithmetic Operations .....	517
6.3.2.	Modifying Precedence .....	517
6.3.3.	Function to Reference Other Terminals.....	517
6.3.3.1.	Functions.....	518
6.3.4.	Constants .....	518
Annex A (informative)	Graphic Symbol Definition .....	519
A.1.	GENERAL RECOMMENDED GRAPHIC SIZING.....	519
A.1.1.	Input and Output Terminals Graphics.....	520
A.1.2.	Bidirectional Signal – Only One Direction at a time .....	521
A.1.3.	Bidirectional Signal – Both directions Simultaneously.....	521
A.1.4.	Text Height .....	521
A.1.5.	Line to Line with Text.....	521
A.1.6.	Line to Line without Text .....	521
A.1.7.	Graphic Sizing for Bus without Text .....	521
A.1.8.	Graphic Sizing for Bus with Text.....	522
A.2.	RECOMMENDED GRAPHIC SYMBOL REPRESENTATION OF SIGNALS, PROPERTIES, FUNCTIONS.....	522
A.2.1.	Negation.....	522
A.2.2.	Active Low & Active High .....	522

A.2.3.	Clock.....	523
A.2.4.	Differential Clock.....	523
A.2.5.	Enable .....	523
A.2.6.	Amplifier.....	524
A.2.7.	Tri-State Output .....	524
A.2.8.	Passive Pulldown.....	525
A.2.9.	Passive Pullup.....	525
A.2.10.	Open Circuit Output .....	525
A.2.11.	Open Circuit Output (H-Type) .....	525
A.2.12.	Open Circuit Output (L-Type).....	526
A.2.13.	Schmitt Trigger .....	526
A.2.14.	Analog Terminal .....	527
A.2.15.	Digital Terminal .....	527
A.2.18.	Subsidiary Connection.....	527
A.2.19.	Postponed Output.....	528
A.3.	RECOMMENDED RULES .....	528
A.3.1.	Symbol Ordering Rule .....	529
A.3.2.	Terminal Grouping .....	529
A.3.2.1	Bit Grouping .....	530
A.3.2.2	Function Grouping .....	530
A.3.2.3	Common Name Grouping .....	531
A.3.3.	Group Direction Symbols .....	531
Annex B (informative) Differences between JEP30-E100 and its predecessors.....		533

---

## Tables

---

Table 1 – Cable Style Configuration	43
Table 2 — Switch Function Table	183
Table 3 — UOM Enumerated Lists	439
Table 4 — NOR logic States	459
Table 5 — LaTeX Syntax Sample	513
Table 6 — LaTeX Symbols and Descriptions Sample	513
Table 7 — Arithmetic Operations	517
Table 8 — Rule Functions	518
Table 9 — Mathematical Functions	518

---

## Figures

---

Figure 1 - Ceramic Capacitor Array	64
Figure 2— Fused Electrolytic Capacitor	66
Figure 3 — Coupled Inductor	92
Figure 4— NPN Darlington	130
Figure 5 — BJT	130
Figure 6 — p-type UJT	131
Figure 7 — Gas Discharge Tube	135
Figure 8 — Vacuum Fluorescent Display	136
Figure 9 — Sample Device with no internal die connection	142
Figure 10 — Clamp Circuit	142
Figure 11 — Hysteresis Loop Curve	143
Figure 12 — Schmitt Trigger	143
Figure 13 — Signal Inversion	144
Figure 14 — Tri-State	144
Figure 15 — Analog Connection Types	146
Figure 16 — Internal Pullup / Pulldown Circuits	148
Figure 17 — Series Component Types	149
Figure 18 — Passive Pull-up Output Circuit	151
Figure 19 — Passive Pull-down Output Circuit	152
Figure 20 — Totem Pole	153
Figure 21 — Rail-to-Rail Push-Pull	154
Figure 22 — High Side Driver (Source Driver)	155
Figure 23 — Low Side Driver (Sink Driver)	156
Figure 24 — Open-circuit Low Side Driver Type	158
Figure 25 — Open-circuit High Side Driver Type	158
Figure 26 — Signal Reference	160
Figure 27 — Same potential	167
Figure 28 — Sample Mixed Gate Device	168
Figure 29 — Sample NAND Gate Device	170
Figure 30 — Internal Electrical Connection	171
Figure 31 — System with Differential Receiver	172
Figure 32 — Differential Pair Device	173
Figure 33 — Function Block Diagram of a 12-bit Bus Switching Device	176
Figure 34 — Clock Buffer	179
Figure 35 — Differential Multiplexer	181
Figure 36 — Circuit with DC-Blocking	185
Figure 37 — Decoupling Circuit	186
Figure 38 — Pull-up Resistor to Power	187
Figure 39 — Pull-down Current Source to Ground	188
Figure 40 — Sample Transistor Circuit	190
Figure 41 — Pull-up Pull-down Resistor Array	397
Figure 42 — Aging Rate	448
Figure 43 — Capacitance Value versus Temperature	452
Figure 44 — Voltage Coefficient of Capacitance	454
Figure 45 — Alternative Symbols	463
Figure 46 — Text Views for Alternative Symbol Rotation	485
Figure 47 — Sample NAND Gate Device	505
Figure 48 — Sample Transistor Circuit	507

## PART MODEL ELECTRICAL GUIDELINE FOR ELECTRONIC-DEVICE PACKAGES - XML REQUIREMENTS

(From JEDEC Board Ballots JCB-17-48, JCB-23-10, JCB-23-27, JCB-23-33, and JCB-24-08 formulated under the cognizance of the JC-11 Committee on Part Model XML Schema Definition.)

---

### 1 Scope

---

This standard establishes the requirements for exchanging part data between part manufacturers and their customers for electrical and electronic products. This standard applies to all forms of electronic parts. It forms part of the Part Model XML Schema, which covers the parental structure for the electrical, physical, Electrical, assembly process classification data along with materials and substances that may be present in the supplied product or sub-products. This Guideline specifically focuses on the “Electrical” sub-section of the Part Model.

All releases of the [ElectricalSection](#) sub-schema must be under the umbrella of the Part Model Schema to ensure that the Part Model 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 data from part manufacturer to customers. 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](http://www.jedec.org))**

**JEDEC Publication, JEP30, *Part Model Guideline for Electronic Device packages – XML Requirements.***

**JEDEC Publication, JEP30-10, *Part Model Schema.***

**JEDEC Publication, JEP30-E101, *Part Model Electrical Schema.***

**JEDEC Publication, JEP30-D10, *Part Model Schema Types Dictionary*** (Required to support the Part Model 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](http://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/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](http://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](http://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](http://www.ansi.org))

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

## 2.5. IEC ([www.iec.org](http://www.iec.org))

**IEC 60617**, *Graphical symbols for diagrams.*

## 2.6. IPC ([www.ipc.org](http://www.ipc.org))

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

## 2.7. INCITIS ([www.incitis.org](http://www.incitis.org))

**2221-D**, *Fibre Channel Physical Interface-6.*

## 2.8. MIPI ([www.mipi.org](http://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](http://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](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](http://www.hdmi.com))**

**High-Definition Multimedia Interface Specification Version 1.3a**

**2.12. PCI-SIG ([www.pcisig.com](http://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](http://www.hp.com))**

**RGMII - Reduced Gigabit Media Independent Interface (RGMII) Version 2.0** - [http://www.hp.com/rnd/pdfs/RGMIIv2\\_0\\_final\\_hp.pdf](http://www.hp.com/rnd/pdfs/RGMIIv2_0_final_hp.pdf)

**2.14. SDA ([www.sdcard.org](http://www.sdcard.org))****2.15. SMI ([www.powersig.org/](http://www.powersig.org/))**

**System Management Bus (SMBus) Specification Version 3.0**

**2.16. USB-IF ([www.usb.org](http://www.usb.org))**

**Universal Serial Bus Specification Revision 2.0**

**Universal Serial Bus 3.2 Specification Revision 1.0**

**2.17. UCle (<https://www.uciexpress.org/>)****2.18. Accellera (<https://www.accellera.org/>)****2.19. American Mathematical Society**

“Short Math Guide for L<sup>A</sup>T<sub>E</sub>X”, Version 1.09 (2002-03-22), currently available at <http://www.ams.org/tex/short-math-guide.html>.

---

## 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 Part Model 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 “Part Model Common Types Library”

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

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

NOTE 1 Companies who use the Part Model XML Files and claim compliance to JEDEC, must ensure that their Part Model XML file conforms to the specific released version of the Part Model 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 Part Model XML Schema and is not intended to act as a standalone schema. In addition, there is a “Part Model Schema Types Library” XML Schema, which is a common set of xml structures shared across the Part Model XML Schema and all its sub-section schemas.

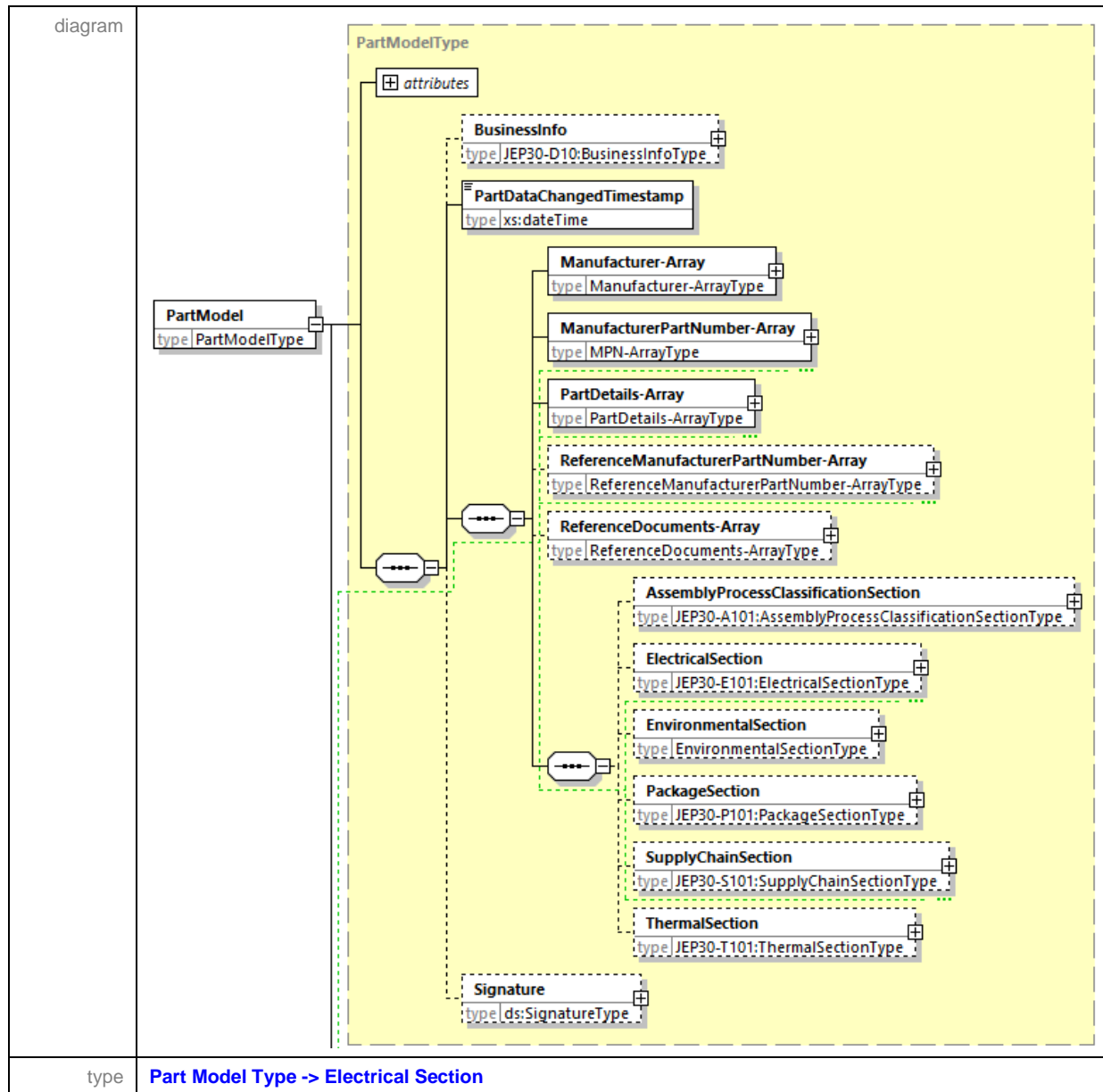
### 3.2. XML Schema Key Terms and Definitions

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

## 4 Part Model Schema Definition

The following section describes the XML Schema structure.

### 4.1. Part Model -> Electrical Section



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

#### 4.1 Part Model -> Electrical Section (cont'd)

This document covers the [ElectricalSection](#), which is referenced from its parent's structure, the [PartModel](#). The contents under the [ElectricalSection](#) are tied to the Manufacturer's name and Manufacturer's part number

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 Part Model.

*"Each time that a Sub-schema gets updated, then the part model version also gets updated in order to release that Sub-schema under the umbrella of the Part model. This is because the Part Model 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	<a href="#">PartModel/ManufacturerPartNumber-Array.</a>
diagram	<p>The diagram illustrates the XML Schema structure for <b>ManufacturerPartNumber-Array</b>. It is an array of <b>ManufacturerPartNumbers</b> (type <b>ManufacturerPartNumbersType</b>). The <b>ManufacturerPartNumbersType</b> contains the following elements:</p> <ul style="list-style-type: none"> <li><b>ID</b> (type <b>xs:string</b>)</li> <li><b>PartNumberSeries</b> (type <b>PartNumberType</b>, 0..∞)</li> <li><b>OrderablePartNumber</b> (type <b>OrderablePartNumberType</b>, 0..∞)</li> <li><b>ManufacturerID</b> (type <b>xs:string</b>)</li> <li><b>ManufacturerSignatureDigest</b> (type <b>JEP30-D10:SignatureDigestLinkType</b>)</li> <li><b>ManufacturerPartNumbersIdentitySignature</b> (type <b>ds:SignatureType</b>)</li> </ul> <p>A <b>constraints</b> icon is present at the bottom of the diagram.</p>
type	<a href="#">MPN-ArrayType</a> , <a href="#">ManufacturerPartNumbersType</a> , <a href="#">PartNumberType</a> , <a href="#">OrderablePartNumber-ArrayType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> , <a href="#">ds:SignatureType</a> .

The [ManufacturerPartNumber-Array](#) consists of a [ManufacturerPartNumbers](#) section that provide the definition of the part number, either via the [PartNumberSeries](#) or the [OrderablePartNumber](#). All Parts via their Part Number families or via their Orderable Part Numbers are connected to the details in the [ElectricalSection](#) via the [PartDetails-Array](#) section.

### 4.3. 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 - Part Model 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 supply chain content. Reference the JEP30 parent document for more details on this association.

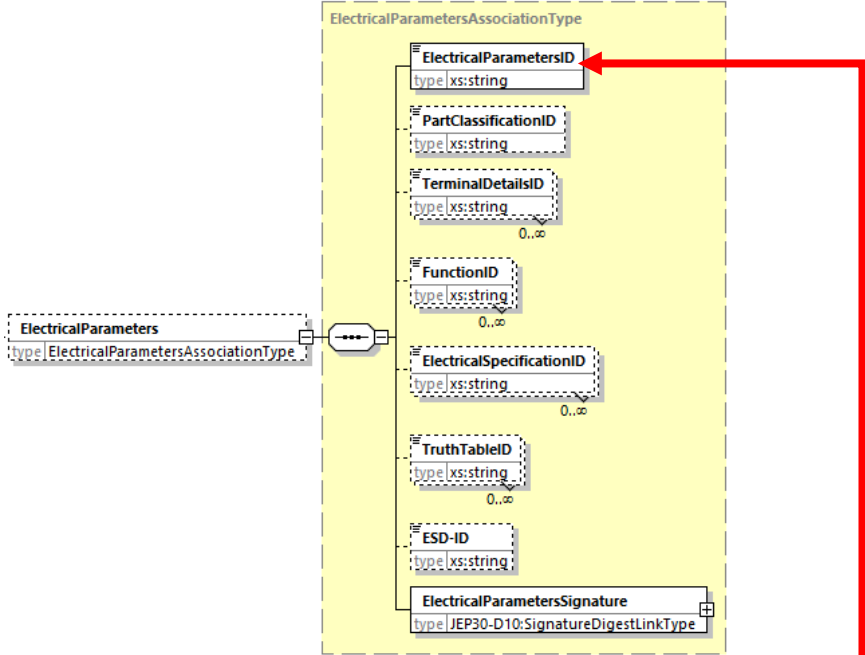
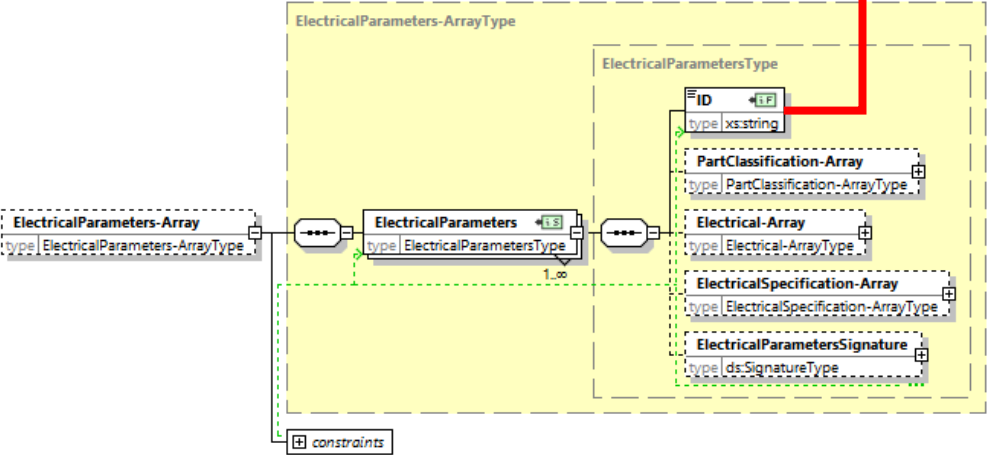
path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array.</a>
diagram at the Association level	<p>The diagram shows a box labeled <b>Electrical-Array</b> with the type <code>ElectricalAssociation-ArrayType</code>. A dashed line with a multiplicity of 1 on the left and 0..∞ on the right connects it to a larger box labeled <b>ElectricalAssociation-ArrayType</b>. Inside this box are five sub-elements, each in a dashed box: <b>ElectricalParameters</b> (type <code>ElectricalParametersAssociationType</code>), <b>SchematicData</b> (type <code>SchematicDataAssociationType</code>), <b>Mapping</b> (type <code>MappingAssociationType</code>), <b>ReferenceDesign</b> (type <code>ReferenceDesignAssociationType</code>), and <b>SoftwareInterfaceDescription</b> (type <code>SoftwareInterfaceDescriptionAssociationType</code>). Each sub-element has a small box with a plus sign in the top right corner.</p>
type	<a href="#">ElectricalAssociation-ArrayType</a> , <a href="#">ElectricalParametersAssociationType</a> , <a href="#">SchematicDataAssociationType</a> , <a href="#">MappingAssociationType</a> , <a href="#">ReferenceDesignAssociationType</a> , <a href="#">SoftwareInterfaceDescriptionAssociationType</a>
diagram at the Electrical Section level	<p>The diagram shows a box labeled <b>ElectricalSection</b> with the type <code>ElectricalSectionType</code>. A dashed line with a multiplicity of 1 on the left and 0..∞ on the right connects it to a larger box labeled <b>ElectricalSectionType</b>. Inside this box are six sub-elements, each in a dashed box: <b>ElectricalParameters-Array</b> (type <code>ElectricalParameters-ArrayType</code>), <b>SchematicData-Array</b> (type <code>SchematicData-ArrayType</code>), <b>Mapping-Array</b> (type <code>Mapping-ArrayType</code>), <b>SimulationModel-Array</b> (type <code>SimulationModel-ArrayType</code>), <b>ReferenceDesign-Array</b> (type <code>ReferenceDesign-ArrayType</code>), and <b>SoftwareInterfaceDescription-Array</b> (type <code>SoftwareInterfaceDescription-ArrayType</code>). Each sub-element has a small box with a plus sign in the top right corner.</p>
type	<a href="#">ElectricalSectionType</a> , <a href="#">ElectricalParameters-ArrayType</a> , <a href="#">SchematicData-ArrayType</a> , <a href="#">Mapping-ArrayType</a> , <a href="#">SimulationModel-ArrayType</a> , <a href="#">ReferenceDesign-ArrayType</a> , <a href="#">SoftwareInterfaceDescription-ArrayType</a> .

#### **4.3 Linking the Manufacturing Part Number to a specific Electrical Data set (cont'd)**

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.



#### 4.3.1. Linking the Manufacturing Part Number to Electrical Parameters

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters</a>
diagram at the Electrical Parameters Association level	 <p>The diagram shows the <b>ElectricalParametersAssociationType</b> structure. It is a container for several elements: <b>ElectricalParametersID</b> (type xs:string), <b>PartClassificationID</b> (type xs:string), <b>TerminalDetailsID</b> (type xs:string, 0..∞), <b>FunctionID</b> (type xs:string, 0..∞), <b>ElectricalSpecificationID</b> (type xs:string, 0..∞), <b>TruthTableID</b> (type xs:string, 0..∞), <b>ESD-ID</b> (type xs:string), and <b>ElectricalParametersSignature</b> (type JEP30-D10:SignatureDigestLinkType). A red arrow points from the <b>ElectricalParametersID</b> element to the <b>ID</b> element in the <b>ElectricalParametersType</b> structure shown in the next row.</p>
type	<a href="#">ElectricalParametersAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters</a>
diagram at the Electrical Parameters-Array level	 <p>The diagram shows the <b>ElectricalParameters-ArrayType</b> structure. It contains an <b>ElectricalParameters</b> element (type <a href="#">ElectricalParametersType</a>, 1..∞). The <b>ElectricalParameters</b> element is further detailed in the <b>ElectricalParametersType</b> structure, which includes: <b>ID</b> (type xs:string, KeyRef), <b>PartClassification-Array</b> (type <a href="#">PartClassification-ArrayType</a>), <b>Electrical-Array</b> (type <a href="#">Electrical-ArrayType</a>), <b>ElectricalSpecification-Array</b> (type <a href="#">ElectricalSpecification-ArrayType</a>), and <b>ElectricalParametersSignature</b> (type ds:SignatureType). A red arrow points from the <b>ID</b> element in the <b>ElectricalParametersType</b> structure to the <b>ElectricalParametersID</b> element in the <b>ElectricalParametersAssociationType</b> structure shown in the first row. A green dashed line connects the <b>ElectricalParameters</b> element in the <b>ElectricalParameters-ArrayType</b> to the <b>ElectricalParametersType</b> structure. A <b>constraints</b> box is also present at the bottom left of the diagram.</p>
type	<a href="#">ElectricalParametersType</a> , <a href="#">ElectricalParametersType</a> , <a href="#">Electrical-ArrayType</a> , <a href="#">ElectricalSpecification-ArrayType</a> .

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](#).

### 4.3.2. Linking the Manufacturing Part Number to Part Classification

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters</a>
diagram at the Electrical Parameters Association level	
type	<a href="#">ElectricalParametersAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification</a>
diagram at the Part Classification-Array level	
type	<a href="#">PartClassification-ArrayType</a> , <a href="#">PartClassificationType</a> , ...

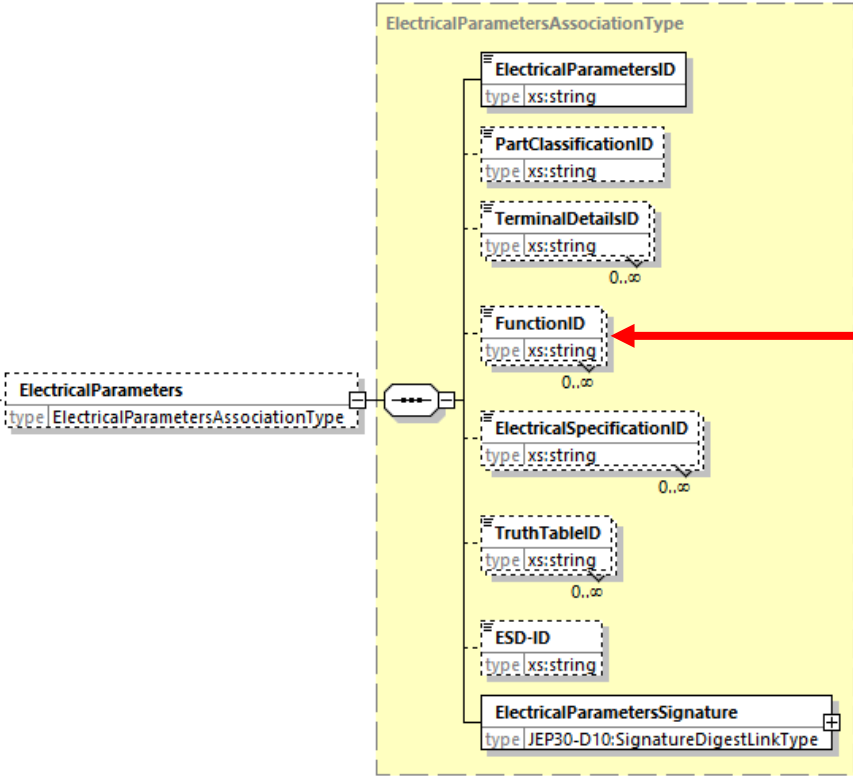
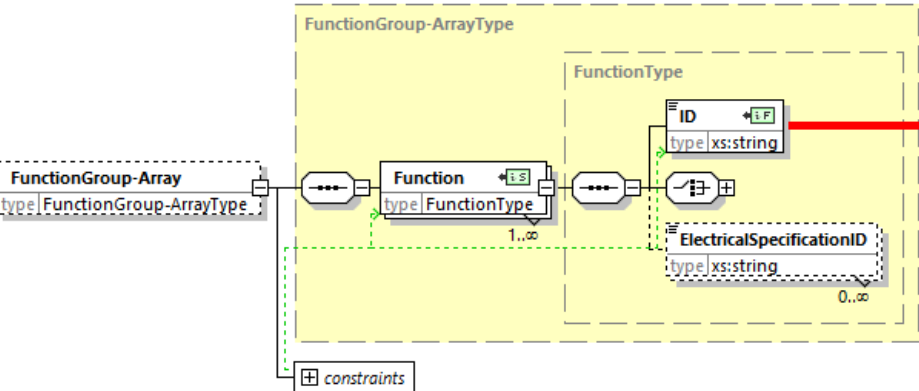
The [PartClassificationID](#) references the [PartClassification/ID](#) under the [ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array](#). This is enforced by the key named as [PartClassificationKey](#) that is assigned to the [PartClassification/ID](#) element,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

### 4.3.3. Linking the Manufacturing Part Number to Terminal Details

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters</a>
diagram at the Electrical Parameters Association level	
type	<a href="#">ElectricalParametersAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array</a>
diagram at the Electrical-Array level	
type	<a href="#">TerminalDetails-ArrayType</a> , <a href="#">TerminalDetailsType</a> , ...

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](#).

#### 4.3.4. Linking the Manufacturing Part Number to Functions

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters</a>
diagram at the Electrical Parameters Association level	 <p>The diagram shows the <b>ElectricalParametersAssociationType</b> structure. It is a dashed box containing several elements: <b>ElectricalParametersID</b> (type xs:string), <b>PartClassificationID</b> (type xs:string), <b>TerminalDetailsID</b> (type xs:string, 0..∞), <b>FunctionID</b> (type xs:string, 0..∞), <b>ElectricalSpecificationID</b> (type xs:string, 0..∞), <b>TruthTableID</b> (type xs:string, 0..∞), <b>ESD-ID</b> (type xs:string), and <b>ElectricalParametersSignature</b> (type JEP30-D10:SignatureDigestLinkType). A red arrow points from the <b>FunctionID</b> element to the <b>Function/ID</b> element in the next diagram.</p>
type	<a href="#">ElectricalParametersAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array</a>
diagram at the Electrical-Array level	 <p>The diagram shows the <b>FunctionGroup-ArrayType</b> structure. It is a dashed box containing a <b>FunctionGroup-Array</b> element (type <a href="#">FunctionGroup-ArrayType</a>) and a <b>Function</b> element (type <a href="#">FunctionType</a>, 1..∞). The <b>Function</b> element is further detailed with a <b>FunctionType</b> sub-structure containing an <b>ID</b> element (type xs:string, KeyRef to <a href="#">JEP30-E101: FunctionKey</a>) and an <b>ElectricalSpecificationID</b> element (type xs:string, 0..∞). A red arrow points from the <b>FunctionID</b> element in the first diagram to the <b>ID</b> element in this diagram.</p>
type	<a href="#">Electrical-ArrayType</a> , <a href="#">ElectricalType</a> , ...

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](#).

#### 4.3.5. Linking the Manufacturing Part Number to Electrical Specification

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters</a>
diagram at the Electrical Parameters Association level	<p>The diagram shows the <b>ElectricalParametersAssociationType</b> structure. It contains a list of IDs: <b>ElectricalParametersID</b> (type xs:string), <b>PartClassificationID</b> (type xs:string), <b>TerminalDetailsID</b> (type xs:string, 0..∞), <b>FunctionID</b> (type xs:string, 0..∞), <b>ElectricalSpecificationID</b> (type xs:string, 0..∞), <b>TruthTableID</b> (type xs:string, 0..∞), and <b>ESD-ID</b> (type xs:string). At the bottom is <b>ElectricalParametersSignature</b> (type JEP30-D10:SignatureDigestLinkType). A red arrow points from the <b>ElectricalSpecificationID</b> to the <b>ID</b> field in the <b>ElectricalSpecificationType</b> diagram below.</p>
type	<a href="#">ElectricalParametersAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification</a>
diagram at the Electrical Specification-Array level	<p>The diagram shows the <b>ElectricalSpecification-ArrayType</b> structure. It contains a list of <b>ElectricalSpecification</b> (type <a href="#">ElectricalSpecificationType</a>, 0..∞) and <b>TruthTable</b> (type <a href="#">TruthTableType</a>, 0..∞). A red arrow points from the <b>ID</b> field in the <b>ElectricalSpecificationType</b> to the <b>ID</b> field in the <b>ElectricalSpecificationType</b>.</p>
type	<a href="#">ElectricalSpecification-ArrayType</a> , <a href="#">ElectricalSpecificationType</a> , ...

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](#).

#### 4.3.6. Linking the Manufacturing Part Number to Truth Table

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters</a>
diagram at the Electrical Parameters Association level	
type	<a href="#">ElectricalParametersAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/TruthTable</a>
diagram at the Electrical Specification-Array level	
type	<a href="#">TruthTable-ArrayType</a> , <a href="#">TruthTableType</a> , ...

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](#).

#### 4.3.7. Linking the Manufacturing Part Number to ESD

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ElectricalParameters</a>
diagram at the Electrical Parameters Association level	
type	<a href="#">ElectricalParametersAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array</a>
diagram at the Electrical Specification-Array level	
type	<a href="#">ESD-ArrayType</a> , <a href="#">ESDType</a> , ...

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](#).

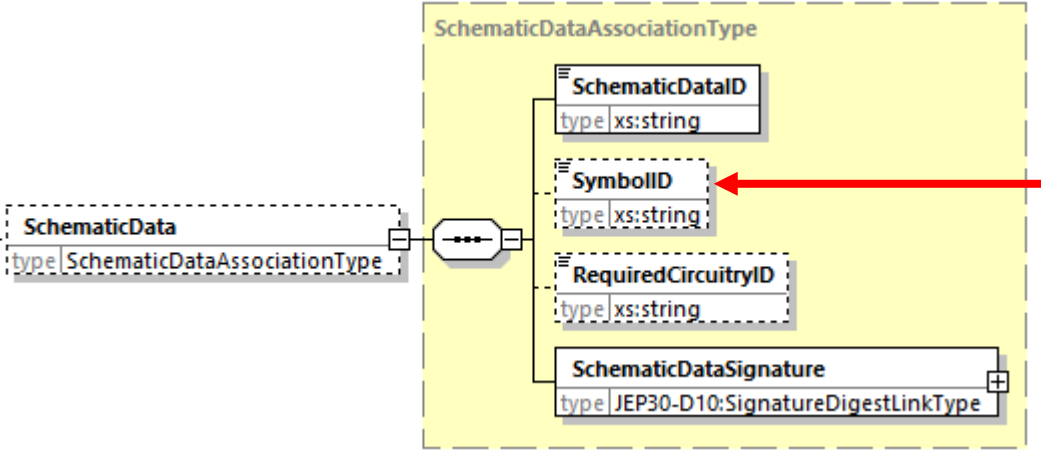
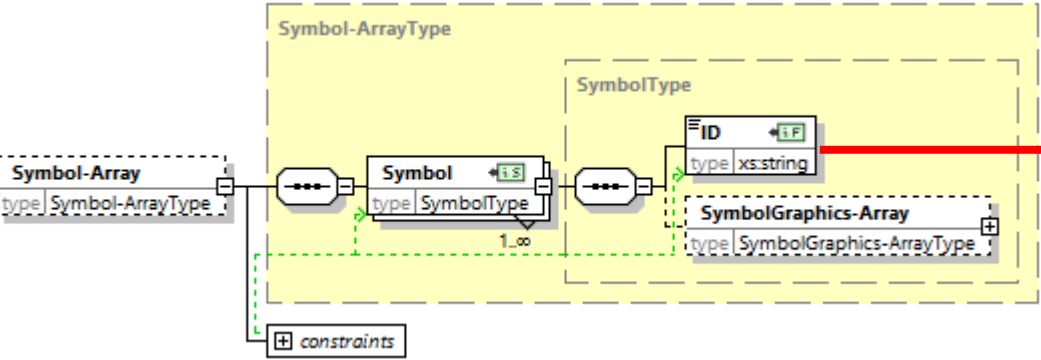
#### 4.3.8. Linking the Manufacturing Part Number to Schematic Data

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/SchematicData</a>
diagram at the Schematic Data Association level	
type	<a href="#">SchematicDataAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/SchematicData-Array</a>
diagram at the Schematic Data-Array level	
type	<a href="#">SchematicData-ArrayType</a> , <a href="#">SchematicDataType</a> , ...

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](#).



#### 4.3.9. 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	 <p>The diagram shows the <b>SchematicDataAssociationType</b> structure. It includes a <b>SchematicData</b> element (type <b>SchematicDataAssociationType</b>) and a <b>SchematicDataAssociationType</b> container. Inside this container are <b>SchematicDataID</b> (type <b>xs:string</b>), <b>SymbolID</b> (type <b>xs:string</b>), <b>RequiredCircuitryID</b> (type <b>xs:string</b>), and <b>SchematicDataSignature</b> (type <b>JEP30-D10:SignatureDigestLinkType</b>). A red arrow points from the <b>SymbolID</b> element to the <b>ID</b> element in the <b>SymbolType</b> diagram below.</p>
type	<b>SchematicDataAssociationType</b> , <b>JEP30-D10:SignatureDigestLinkType</b> .
path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array
diagram at the Symbol-Array level	 <p>The diagram shows the <b>Symbol-ArrayType</b> structure. It includes a <b>Symbol-Array</b> element (type <b>Symbol-ArrayType</b>) and a <b>SymbolType</b> container. Inside <b>SymbolType</b> are <b>Symbol</b> (type <b>SymbolType</b>) and <b>SymbolGraphics-Array</b> (type <b>SymbolGraphics-ArrayType</b>). The <b>Symbol</b> element has an <b>ID</b> (type <b>xs:string</b>) and is referenced by the <b>SymbolID</b> in the <b>SchematicDataAssociationType</b> diagram above. A red arrow points from the <b>SymbolID</b> to the <b>ID</b> element. A green dashed line connects the <b>Symbol</b> element to the <b>SymbolGraphics-Array</b> element, labeled with <b>1..∞</b>. A <b>constraints</b> box is also shown.</p>
type	<b>Symbol-ArrayType</b> , <b>SymbolType</b> , ...

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**.

#### 4.3.10. Linking the Manufacturing Part Number to Required Circuitry

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/SchematicData</a>
diagram at the Schematic Data Association level	
type	<a href="#">SchematicDataAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/RequiredCircuitry-Array</a>
diagram at the Required Circuitry-Array level	
type	<a href="#">RequiredCircuitry-ArrayType</a> , <a href="#">RequiredCircuitryType</a> , ...

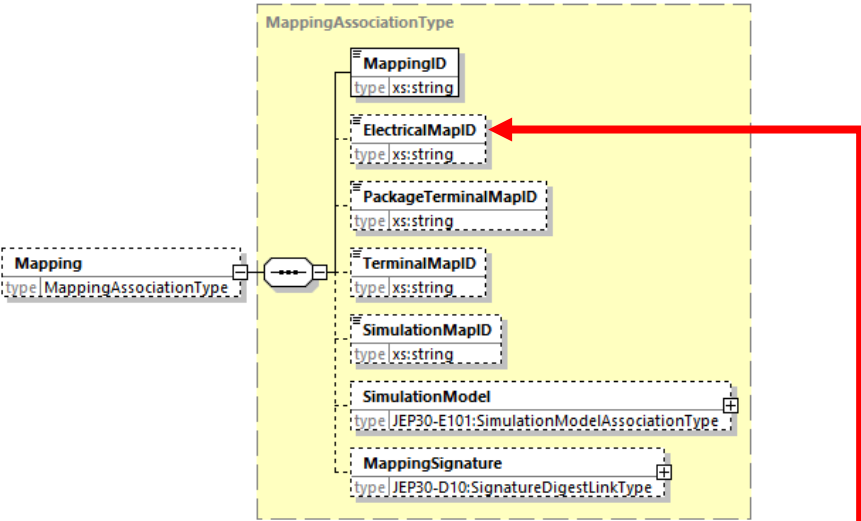
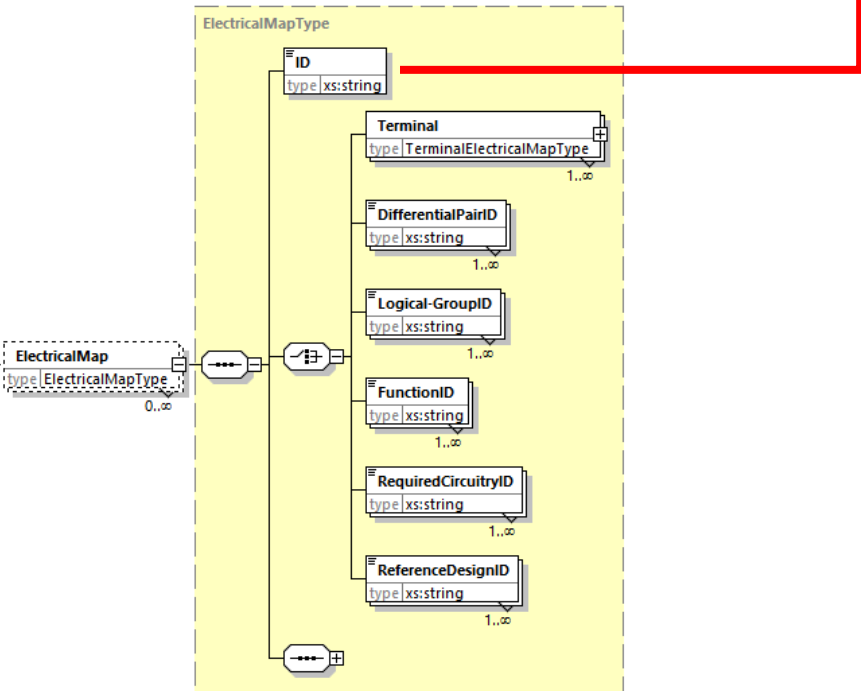
The [RequiredCircuitryID](#) references the [RequiredCircuitry/ID](#) under the [ElectricalSection/SchematicData-Array/SchematicData/RequiredCircuitry-Array](#). This is enforced by the key named as [RequiredCircuitryKey](#) that is assigned to the [RequiredCircuitry/ID](#) element, which is referenced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

#### 4.3.11. Linking the Manufacturing Part Number to Mapping data

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping</a>
diagram at the Mapping Association level	<p>The diagram shows a <b>Mapping</b> element (type <code>MappingAssociationType</code>) connected to a <b>MappingAssociationType</b> container. Inside this container, there are several elements: <b>MappingID</b> (type <code>xs:string</code>), <b>ElectricalMapID</b> (type <code>xs:string</code>), <b>PackageTerminalMapID</b> (type <code>xs:string</code>), <b>TerminalMapID</b> (type <code>xs:string</code>), <b>SimulationMapID</b> (type <code>xs:string</code>), <b>SimulationModel</b> (type <code>JEP30-E101:SimulationModelAssociationType</code>), and <b>MappingSignature</b> (type <code>JEP30-D10:SignatureDigestLinkType</code>). A red arrow points from the <b>MappingID</b> element to the <b>ID</b> element in the MappingType diagram below.</p>
type	<a href="#">MappingAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> , ...
path	<a href="#">PartModel/ElectricalSection/Mapping-Array</a>
diagram at the Mapping-Array level	<p>The diagram shows a <b>Mapping-Array</b> element (type <code>Mapping-ArrayType</code>) connected to a <b>Mapping-ArrayType</b> container. Inside this container, there is a <b>Mapping</b> element (type <code>MappingType</code>). The <b>MappingType</b> container has several components: <b>ID</b> (type <code>xs:string</code>), <b>ElectricalMap</b> (type <code>ElectricalMapType</code>), <b>PackageTerminalMap</b> (type <code>PackageTerminalMapType</code>), <b>SimulationMap</b> (type <code>SimulationMapType</code>), and <b>MappingSignature</b> (type <code>ds:SignatureType</code>). A red arrow points from the <b>ID</b> element in the MappingType container to the <b>MappingID</b> element in the MappingAssociationType diagram above. A <b>constraints</b> box is also present at the bottom left of the Mapping-ArrayType container.</p>
type	<a href="#">Mapping-ArrayType</a> , <a href="#">MappingType</a> , ...

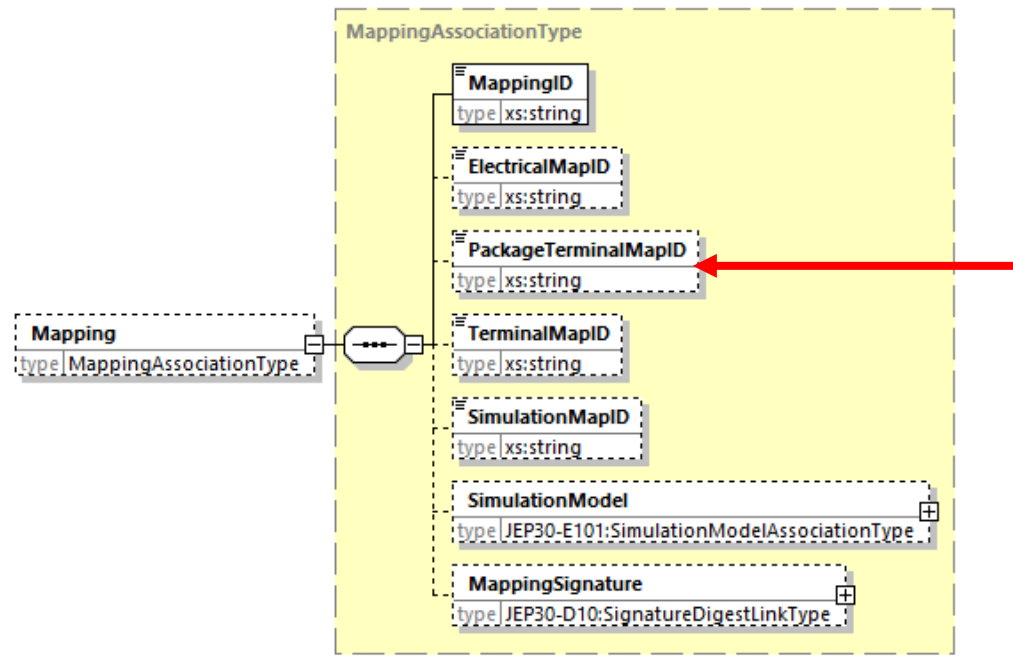
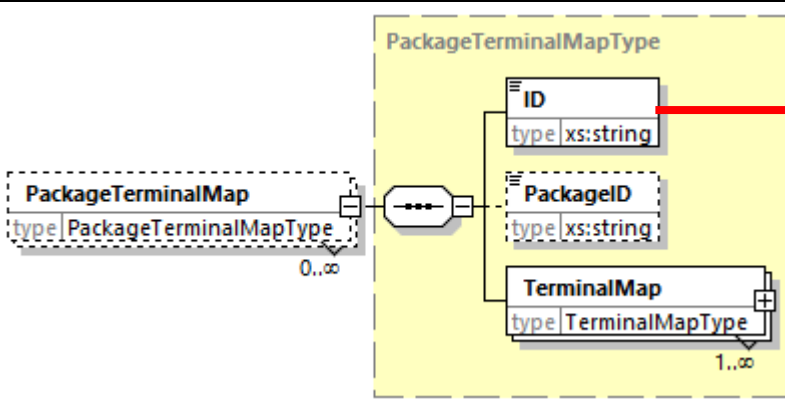
The [MappingID](#) references the [Mapping/ID](#) under the [ElectricalSection/Mapping-Array](#). This is enforced by the key named as [MappingKey](#) that is assigned to the [Mapping/ID](#) element, which is referenced by the [MappingID](#) which has a KeyRef that refers to the [JEP30-E101:MappingKey](#).

#### 4.3.12. Linking the Manufacturing Part Number to Electrical Map data

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping</a>	
diagram at the Mapping Association level	 <p>The diagram shows the <b>MappingAssociationType</b> structure. It includes elements: <b>MappingID</b> (type xs:string), <b>ElectricalMapID</b> (type xs:string), <b>PackageTerminalMapID</b> (type xs:string), <b>TerminalMapID</b> (type xs:string), <b>SimulationMapID</b> (type xs:string), <b>SimulationModel</b> (type JEP30-E101:SimulationModelAssociationType), and <b>MappingSignature</b> (type JEP30-D10:SignatureDigestLinkType). A red arrow points from the <b>ElectricalMapID</b> element to the <b>ElectricalMap</b> element in the diagram below.</p>	
type	<b>MappingAssociationType</b> , JEP30-D10:SignatureDigestLinkType, ...	
path	<a href="#">PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap</a>	
diagram at Terminal Map level	 <p>The diagram shows the <b>ElectricalMapType</b> structure. It includes elements: <b>ID</b> (type xs:string), <b>Terminal</b> (type TerminalElectricalMapType, 1..∞), <b>DifferentialPairID</b> (type xs:string, 1..∞), <b>Logical-GroupID</b> (type xs:string, 1..∞), <b>FunctionID</b> (type xs:string, 1..∞), <b>RequiredCircuitryID</b> (type xs:string, 1..∞), and <b>ReferenceDesignID</b> (type xs:string, 1..∞). A red arrow points from the <b>ID</b> element to the <b>ElectricalMapID</b> element in the diagram above.</p>	
type	<b>ElectricalMapType</b> , ...	

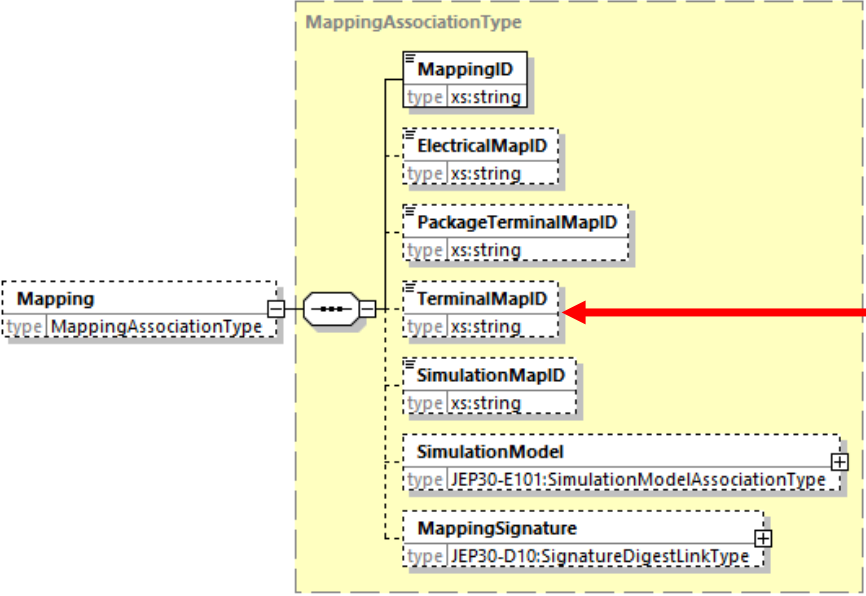
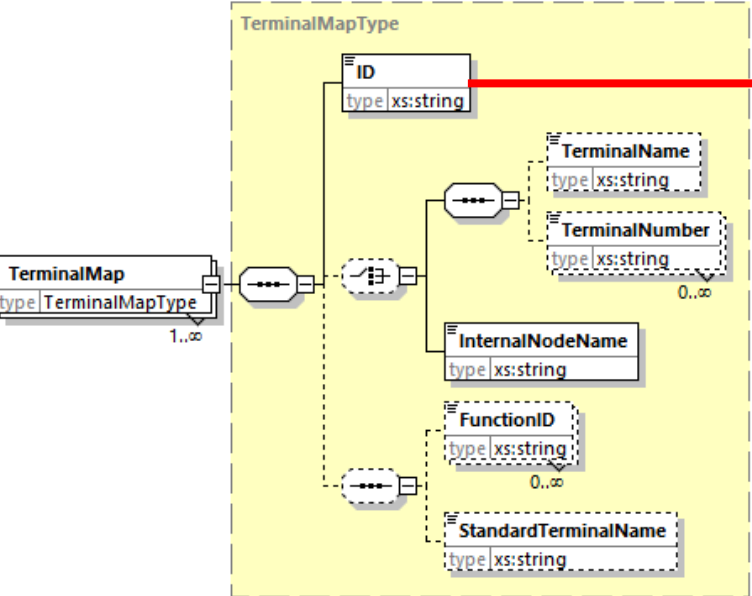
The [ElectricalMapID](#) references the [MappingID](#) under the [ElectricalSection/Mapping-Array/Mapping/ElectricalMap](#). This is enforced by the key named as [ElectricalMapKey](#) that is assigned to the [ElectricalMap/ID](#) element, which is referenced by the [ElectricalMapID](#) which has a KeyRef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

#### 4.3.13. Linking the Manufacturing Part Number to Package Terminal Map data

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping</a>
diagram at the Mapping Association level	 <p>The diagram shows a <b>Mapping</b> element (type <b>MappingAssociationType</b>) connected to a <b>MappingAssociationType</b> structure. This structure contains several elements: <b>MappingID</b> (type <b>xs:string</b>), <b>ElectricalMapID</b> (type <b>xs:string</b>), <b>PackageTerminalMapID</b> (type <b>xs:string</b>), <b>TerminalMapID</b> (type <b>xs:string</b>), <b>SimulationMapID</b> (type <b>xs:string</b>), <b>SimulationModel</b> (type <b>JEP30-E101:SimulationModelAssociationType</b>), and <b>MappingSignature</b> (type <b>JEP30-D10:SignatureDigestLinkType</b>). A red arrow points from the <b>PackageTerminalMapID</b> element to the <b>ID</b> element in the <b>PackageTerminalMapType</b> diagram below.</p>
type	<a href="#">MappingAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> , ...
path	<a href="#">PartModel/ElectricalSection/Mapping-Array/Mapping/PackageTerminalMap</a>
diagram at Terminal Map level	 <p>The diagram shows a <b>PackageTerminalMap</b> element (type <b>PackageTerminalMapType</b>) connected to a <b>PackageTerminalMapType</b> structure. This structure contains three elements: <b>ID</b> (type <b>xs:string</b>), <b>PackageID</b> (type <b>xs:string</b>), and <b>TerminalMap</b> (type <b>TerminalMapType</b>). A red arrow points from the <b>ID</b> element to the <b>PackageTerminalMapID</b> element in the <b>MappingAssociationType</b> diagram above.</p>
type	<a href="#">PackageTerminalMapType</a> , ...

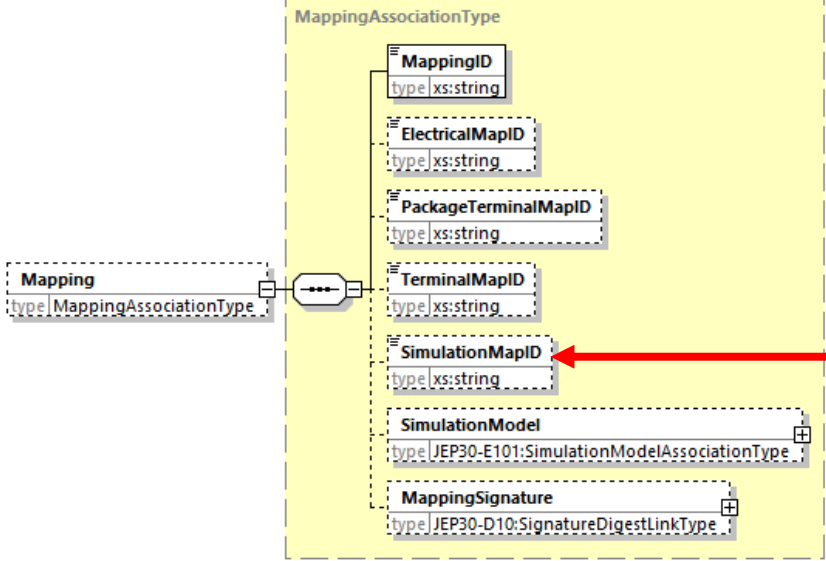
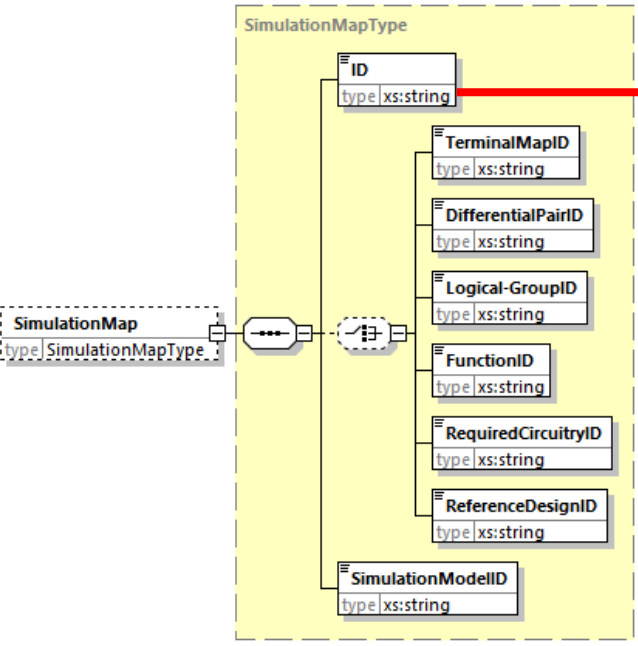
The [PackageTerminalMapID](#) references the [PackageTerminalMap/ID](#) under the [ElectricalSection/Mapping-Array/Mapping/PackageTerminalMap](#). This is enforced by the key named as [PackageTerminalMapKey](#) that is assigned to the [PackageTerminalMap /ID](#) element, which is referenced by the [PackageTerminalMapID](#)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

#### 4.3.14. Linking the Manufacturing Part Number to Terminal Map data

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping</a>	
diagram at the Mapping Association level	 <p>The diagram shows the <b>MappingAssociationType</b> structure. It includes elements: <b>MappingID</b> (type xs:string), <b>ElectricalMapID</b> (type xs:string), <b>PackageTerminalMapID</b> (type xs:string), <b>TerminalMapID</b> (type xs:string), <b>SimulationMapID</b> (type xs:string), <b>SimulationModel</b> (type JEP30-E101:SimulationModelAssociationType), and <b>MappingSignature</b> (type JEP30-D10:SignatureDigestLinkType). A red arrow points from the <b>TerminalMapID</b> element to the <b>ID</b> element in the <b>TerminalMapType</b> diagram below.</p>	
type	<a href="#">MappingAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> , ...	
path	<a href="#">PartModel/ElectricalSection/Mapping-Array/Mapping/PackageTerminalMap/TerminalMap</a>	
diagram at Terminal Map level	 <p>The diagram shows the <b>TerminalMapType</b> structure. It includes elements: <b>ID</b> (type xs:string), <b>TerminalName</b> (type xs:string), <b>TerminalNumber</b> (type xs:string, 0..∞), <b>InternalNodeName</b> (type xs:string), <b>FunctionID</b> (type xs:string, 0..∞), and <b>StandardTerminalName</b> (type xs:string). A red arrow points from the <b>ID</b> element to the <b>TerminalMapID</b> element in the <b>MappingAssociationType</b> diagram above.</p>	
type	<a href="#">TerminalMapType</a> , ...	

The [TerminalMapID](#) references the [TerminalMap/ID](#) under the [ElectricalSection/Mapping-Array//Mapping/PackageTerminalMap/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](#).

#### 4.3.15. Linking the Manufacturing Part Number to Simulation Map data

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping</a>
diagram at the Mapping Association level	 <p>The diagram shows a <b>Mapping</b> element (type <code>MappingAssociationType</code>) connected to a <b>MappingAssociationType</b> structure. This structure contains several elements: <b>MappingID</b> (type <code>xs:string</code>), <b>ElectricalMapID</b> (type <code>xs:string</code>), <b>PackageTerminalMapID</b> (type <code>xs:string</code>), <b>TerminalMapID</b> (type <code>xs:string</code>), <b>SimulationMapID</b> (type <code>xs:string</code>), <b>SimulationModel</b> (type <code>JEP30-E101:SimulationModelAssociationType</code>), and <b>MappingSignature</b> (type <code>JEP30-D10:SignatureDigestLinkType</code>). A red arrow points from the <b>SimulationMapID</b> element to the <b>SimulationMap/ID</b> element in the next diagram.</p>
type	<a href="#">MappingAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> , ...
path	<a href="#">PartModel/ElectricalSection/Mapping-Array/Mapping/SimulationMap</a>
diagram at the Simulation Map level	 <p>The diagram shows a <b>SimulationMap</b> element (type <code>SimulationMapType</code>) connected to a <b>SimulationMapType</b> structure. This structure contains several elements: <b>ID</b> (type <code>xs:string</code>), <b>TerminalMapID</b> (type <code>xs:string</code>), <b>DifferentialPairID</b> (type <code>xs:string</code>), <b>Logical-GroupID</b> (type <code>xs:string</code>), <b>FunctionID</b> (type <code>xs:string</code>), <b>RequiredCircuitryID</b> (type <code>xs:string</code>), <b>ReferenceDesignID</b> (type <code>xs:string</code>), and <b>SimulationModelID</b> (type <code>xs:string</code>). A red arrow points from the <b>SimulationMapID</b> element in the previous diagram to the <b>ID</b> element in this diagram.</p>
type	<a href="#">SimulationMapType</a> .

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](#).

#### 4.3.16. Linking the Manufacturing Part Number to Simulation Models

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/Mapping/SimulationModel</a>
diagram at the Simulation Model Association level	
type	<a href="#">JEP30-E101:SimulationModelAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .
path	<a href="#">PartModel/ElectricalSection/SimulationModel-Array</a>
diagram at the Simulation Model-Array level	
type	<a href="#">SimulationModel-ArrayType</a> , <a href="#">SimulationModelType</a> , ...

The [SimulationModelID](#) references the [SimulationModel/ID](#) under the [ElectricalSection/SimulationModel-Array/SimulationModel](#). This is enforced by the key named as [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](#).



#### 4.3.17. Linking the Manufacturing Part Number to Reference Design

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/ReferenceDesign</a>
diagram at the Reference Design Association level	
type	<a href="#">ReferenceDesignAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> , ...
path	<a href="#">PartModel/ElectricalSection/ReferenceDesign-Array</a>
diagram at the Reference Design-Array level	
type	<a href="#">ReferenceDesign-ArrayType</a> , <a href="#">ReferenceDesignType</a> , ...

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](#).

#### 4.3.18. Linking the Manufacturing Part Number to Software Interface Description

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Electrical-Array/SoftwareInterfaceDescription</a>
diagram at the Software Interface Description Association level	
type	<a href="#">SoftwareInterfaceDescriptionAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> , ...
path	<a href="#">PartModel/ElectricalSection/SoftwareInterfaceDescription-Array</a>
diagram at the Software Interface Description -Array level	
type	<a href="#">SoftwareInterfaceDescription-ArrayType</a> , <a href="#">SoftwareInterfaceDescription Type</a> , ...

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](#).

#### 4.4. Electrical Section

path	<b>PartModel/ElectricalSection</b>
diagram	
type	<b>ElectricalParameters-ArrayType, SchematicData-ArrayType, Mapping-ArrayType, SimulationModel-ArrayType, ReferenceDesign-ArrayType.</b>

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 is 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 of full reference designs. These reference design are grouped under the [ReferenceDesign-Array](#) branch.

These are described in more detail in the following sections.

## 4.5. Electrical Parameters

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array</b>
diagram	<p>The diagram illustrates the XSD structure for the <b>ElectricalParameters-Array</b>. It is an array of <b>ElectricalParameters</b> (type <b>ElectricalParametersType</b>) with a cardinality of 1..∞. The <b>ElectricalParametersType</b> is a complex type containing the following elements:</p> <ul style="list-style-type: none"> <li><b>ID</b> (type <b>xs:string</b>)</li> <li><b>PartClassification-Array</b> (type <b>PartClassification-ArrayType</b>)</li> <li><b>TerminalDetails-Array</b> (type <b>TerminalDetails-ArrayType</b>)</li> <li><b>FunctionGroup-Array</b> (type <b>FunctionGroup-ArrayType</b>)</li> <li><b>ElectricalSpecification-Array</b> (type <b>ElectricalSpecification-ArrayType</b>)</li> <li><b>ESD-Array</b> (type <b>ESD-ArrayType</b>)</li> <li><b>ElectricalParametersSignature</b> (type <b>ds:SignatureType</b>)</li> </ul> <p>A <b>constraints</b> box is also present at the bottom of the diagram.</p>
type	<b>ElectricalParameters-ArrayType</b> , <b>ElectricalParametersType</b> , <b>PartClassification-ArrayType</b> , <b>Electrical-ArrayType</b> , <b>ElectricalSpecification-ArrayType</b> , <b>ds:SignatureType</b>

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.

#### 4.5.1. Part Classification Array

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array</b>
diagram	
type	<b>PartClassification-ArrayType, PartClassificationType, CableAndWiringClassificationType, ConnectorClassificationType, ElectricalClassificationType, HardwareClassificationType, OpticsClassificationType, CompanionPartType.</b>

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

#### 4.5.1 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](#) is another could be stored in a separate PartModel file. The identity of the [CompanionPart](#) could be via an an Orderable Part Number that is already within this PartModel file or referenced to a Manufacturer's Part Number form a specific Manufacturer. In some cases, a hard identity is not required such as sttachment 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.

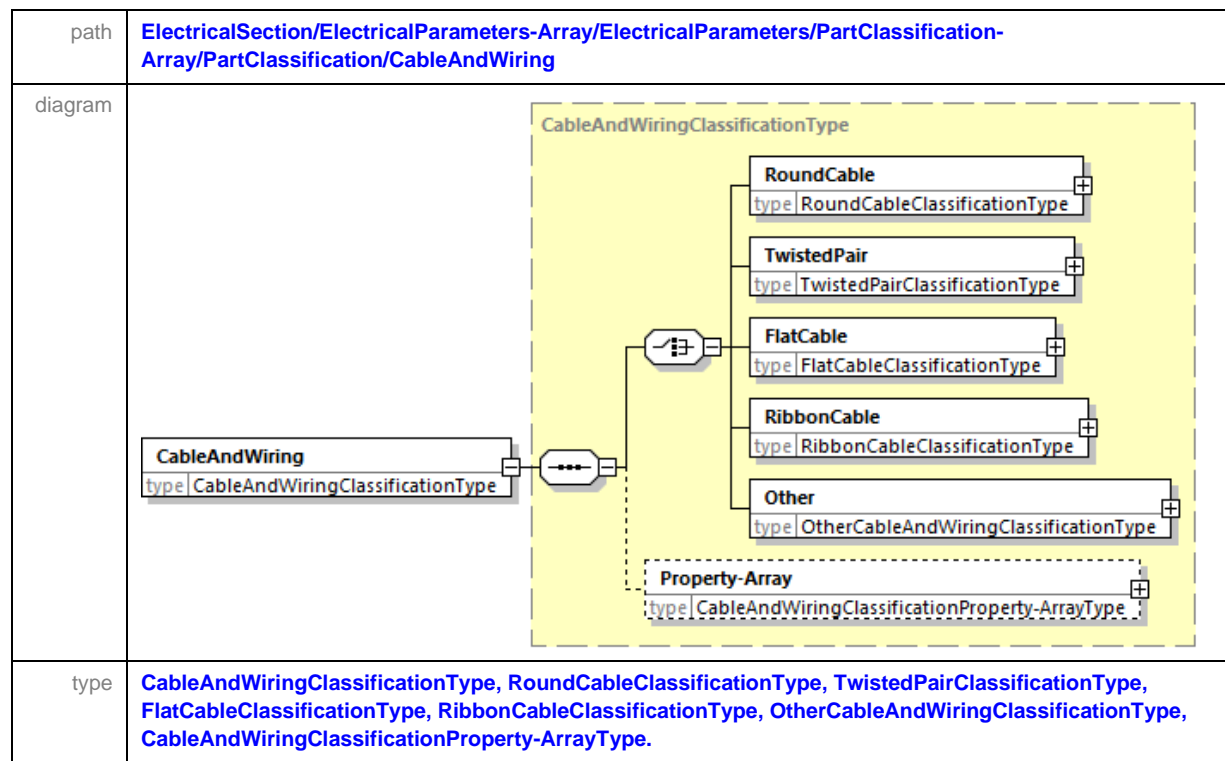
Each of the 5 classifications above have a hierarichal structure that sub-divide the higher-level classification into smaller and smaller grouping of parts. At all levels, a set of properties can be defined tunder a Property-Array that provide additional details for the parts. The purpose of classifying parts into a hierarichal structure is to assist the user to search for the parts that they are looking for. Because of this a deep hierarichal 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 scabality 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.

## 4.5.1 Part Classification Array (cont'd)

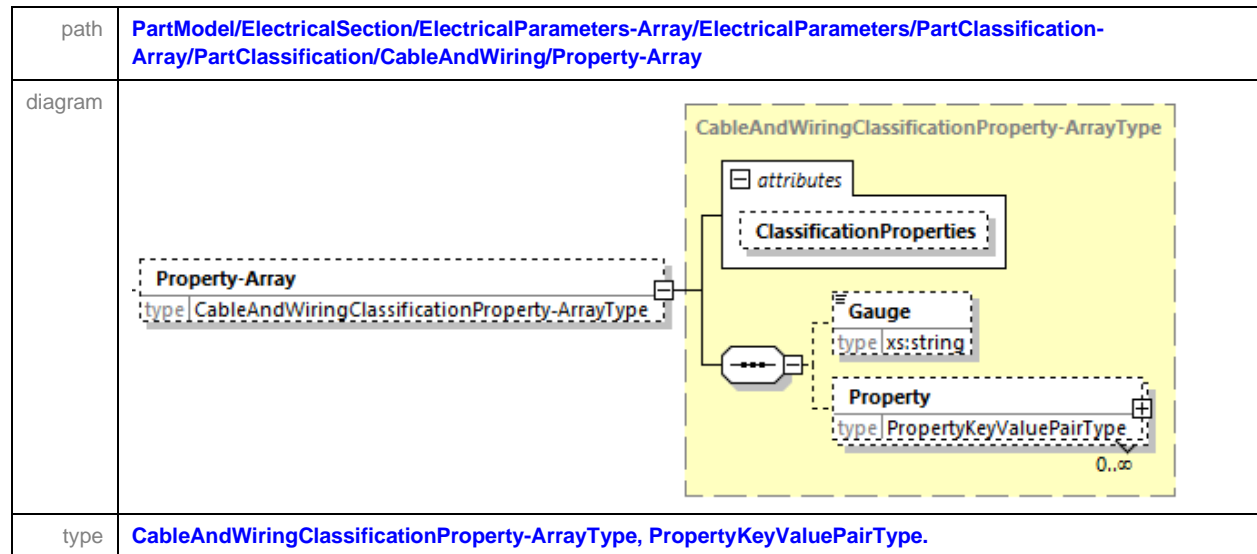
path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/....(all classification branches to lowest level classification branch. The Resistor branch is just one of these branches)</b>
diagram	<p>The diagram illustrates the structure of the Resistor classification system. It starts with <b>ResistorClassificationType</b> (type: ResistorClassificationType), which is associated with a <b>Property-Array</b> (type: ResistorClassificationProperty-ArrayType). This array is composed of three elements: <b>Fixed</b> (type: FixedResistorClassificationType), <b>Adjustable</b> (type: AdjustableResistorClassificationType), and <b>NonLinearResistor</b> (type: NonLinearResistorClassificationType). The <b>Fixed</b> element is further associated with a <b>Property-Array</b> (type: FixedResistorClassificationProperty-ArrayType), which contains <b>Material</b> (type: FixedResistorMaterialPropertyType) and <b>Property</b> (type: PropertyKeyValuePairType). The <b>Property</b> element is associated with a <b>PropertyKeyValuePairType</b> (type: PropertyKeyValuePairType), which has <b>Name</b> (type: xs:string) and <b>Value</b> (type: xs:string) attributes. The <b>ResistorClassificationProperty-ArrayType</b> also contains <b>ClassificationProperties</b> (type: ResistorClassificationProperty-ArrayType) and <b>Property</b> (type: PropertyKeyValuePairType).</p>
type (sample specifically for Resistor category)	<b>ResistorClassificationType, FixedResistorClassificationType, FixedResistorClassificationProperty-ArrayType, JEP30-D10:EmptyType, FixedResistorMaterialPropertyType, FixedResistorClassificationPropertyType, AdjustableResistorClassificationType, NonLinearResistorClassificationType, ResistorClassificationProperty-ArrayType, PropertyKeyValuePairType.</b>

#### 4.5.1.1. Cable and Wiring Classification



Each of the above sub-classifications of [RoundCable](#), [TwistedPair](#), [FlatCable](#), [RibbonCable](#) and [OtherCableAndWiring](#) 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.

##### 4.5.1.1.1. Cable and Wiring Classification Property-Array





#### 4.5.1.1.1 Cable and Wiring Classification Property - Array (cont'd)

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 *OtherCableAndWiring*. Each *Property* has its own key value pairs, namely *Name* and *Value*.

#### 4.5.1.1.2. Other Cable and Wiring Classification

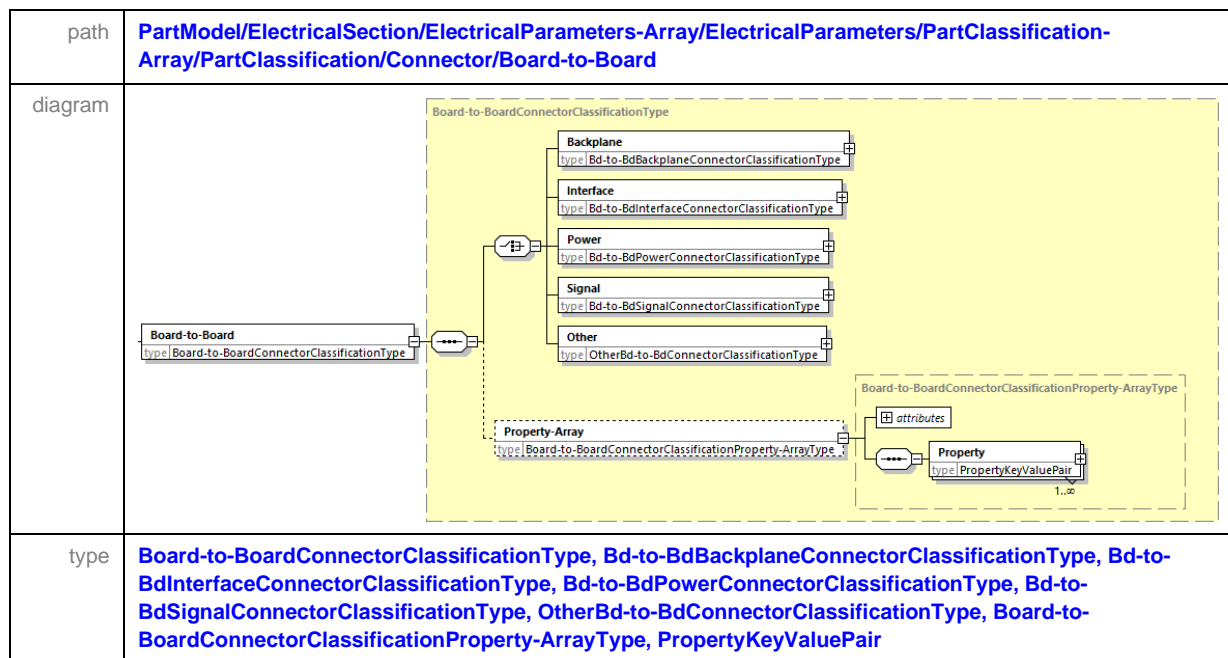
path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/CableAndWiring/OtherCableAndWiring</b>
diagram	<pre> classDiagram     class OtherConnectorClassificationType {         Sub-CategoryName xs:string         Property-Array OtherConnectorClassificationProperty-ArrayType     }     class OtherConnectorClassificationProperty-ArrayType {         ClassificationProperties attributes         Property PropertyKeyValuePairType     }     class PropertyKeyValuePairType {         Name         Value     }     OtherConnectorClassificationType "1" -- "1..∞" OtherConnectorClassificationProperty-ArrayType     OtherConnectorClassificationProperty-ArrayType "1" -- "1..∞" PropertyKeyValuePairType </pre>
type	<b>OtherCableAndWiringClassificationType, OtherCableAndWiringProperty-ArrayType, PropertyKeyValuePairType.</b>

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.

4.5.1.2. Connector Classification

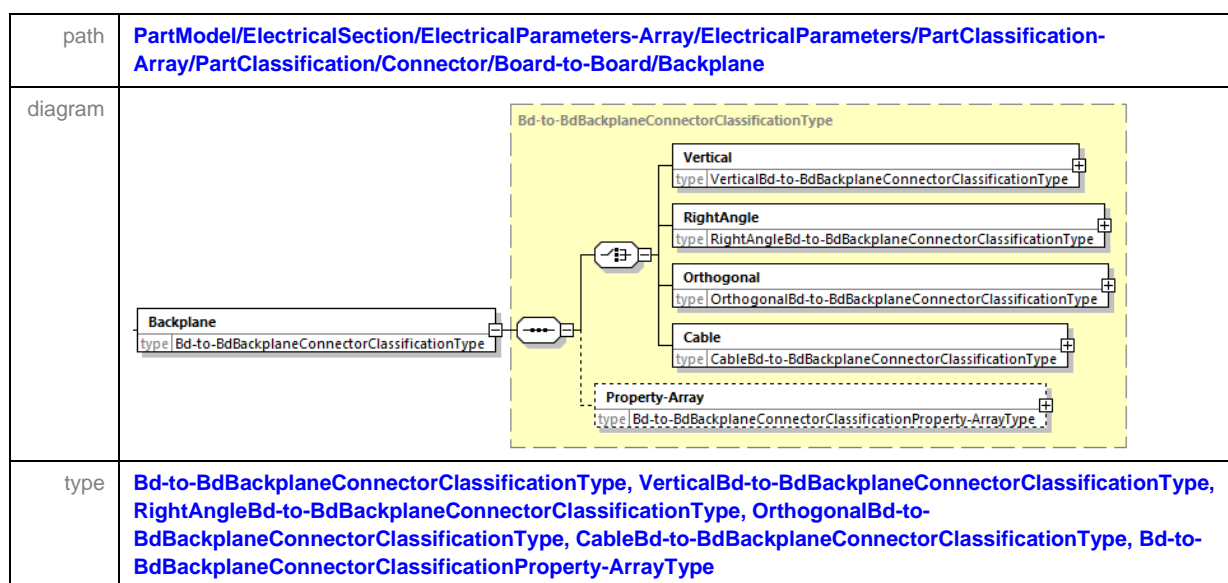
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector
diagram	<p>The diagram illustrates the Connector Classification hierarchy. A central dashed box labeled <b>ConnectorClassificationType</b> contains the following classes, each with a <code>type</code> attribute:</p> <ul style="list-style-type: none"><li><b>Board-to-Board</b> (type: Board-to-BoardConnectorClassificationType)</li><li><b>Cable-to-Board</b> (type: Cable-to-BoardConnectorClassificationType)</li><li><b>Cable-to-Cable</b> (type: Cable-to-CableConnectorClassificationType)</li><li><b>CardEdge-to-Board</b> (type: CardEdge-to-BoardConnectorClassificationType)</li><li><b>CardEdge-to-Cable</b> (type: CardEdge-to-CableConnectorClassificationType)</li><li><b>Optical</b> (type: OpticalConnectorClassificationType)</li><li><b>Socket</b> (type: SocketConnectorClassificationType)</li><li><b>Other</b> (type: OtherConnectorClassificationType)</li><li><b>Property-Array</b> (type: ConnectorClassificationProperty-ArrayType)</li></ul> <p>A <b>Connector</b> class (type: ConnectorClassificationType) is shown to the left, connected to the <b>ConnectorClassificationType</b> box by a dashed line.</p>
type	ConnectorClassificationType, Board-to-BoardConnectorClassificationType, Cable-to-BoardConnectorClassificationType, Cable-to-CableConnectorClassificationType, CardEdge-to-BoardConnectorClassificationType, CardEdge-to-CableConnectorClassificationType, OpticalConnectorClassificationType, SocketConnectorClassificationType, OtherConnectorClassificationType, ConnectorClassificationProperty-ArrayType.

#### 4.5.1.2.1. Board-to-Board Classification



A *Board-to-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*, *Power*, and *Signal* do not have additional sub-classifications in this release.

##### 4.5.1.2.1.1. Backplane Classification



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

#### 4.5.1.2.1.1.1. Board-to-Board Backplane Classification Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Board-to-Board/Backplane</a>
diagram	
type	<a href="#">Bd-to-BdBackplaneConnectorClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">SMD-TH-PF-MountingTechnologyType</a> , <a href="#">PropertyKeyValuePairType</a> .

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.

The [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.

#### 4.5.1.2.2. Cable-to-Board Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board</a>
diagram	<pre> classDiagram     class CableToBoard {         type Cable-to-BoardConnectorClassificationType     }     class CableToBoardConnectorClassificationType {         type AudioVideoDisplayCable-to-BoardConnectorClassificationType         type BackplaneCable-to-BoardConnectorClassificationType         type PowerAndSignalCable-to-BoardConnectorClassificationType         type PowerCable-to-BoardConnectorClassificationType         type SignalCable-to-BoardConnectorClassificationType         type RFCable-to-BoardConnectorClassificationType         type OtherCable-to-BoardConnectorClassificationType     }     class PropertyArray {         type Cable-to-BoardConnectorClassificationProperty-ArrayType     }     CableToBoard --&gt; CableToBoardConnectorClassificationType     CableToBoardConnectorClassificationType --&gt; PropertyArray     PropertyArray --&gt; AudioVideoDisplayCableToBoardConnectorClassificationType     PropertyArray --&gt; BackplaneCableToBoardConnectorClassificationType     PropertyArray --&gt; PowerAndSignalCableToBoardConnectorClassificationType     PropertyArray --&gt; PowerCableToBoardConnectorClassificationType     PropertyArray --&gt; SignalCableToBoardConnectorClassificationType     PropertyArray --&gt; RFCableToBoardConnectorClassificationType     PropertyArray --&gt; OtherCableToBoardConnectorClassificationType   </pre>
type	<a href="#">Cable-to-BoardConnectorClassificationType</a> , <a href="#">AudioVideoDisplayCable-to-BoardConnectorClassificationType</a> , <a href="#">BackplaneCable-to-BoardConnectorClassificationType</a> , <a href="#">PowerAndSignalCable-to-BoardConnectorClassificationType</a> , <a href="#">PowerCable-to-BoardConnectorClassificationType</a> , <a href="#">SignalCable-to-BoardConnectorClassificationType</a> , <a href="#">RFCable-to-BoardConnectorClassificationType</a> , <a href="#">OtherCable-to-BoardConnectorClassificationType</a> , <a href="#">Cable-to-BoardConnectorClassificationProperty-ArrayType</a>

#### 4.5.1.2.2.1. Audio Video Display Cable-to-Board Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/AudioVideoDisplay</b>
diagram	<p>The diagram illustrates the structure of the <b>AudioVideoDisplayCable-to-BoardConnectorClassificationType</b>. It features a <b>Property-Array</b> (type: <b>AudioVideoDisplayCable-to-BoardConnectorClassificationProperty-ArrayType</b>) that contains several connector types, each with its own <b>Cable-to-BoardConnectorClassificationType</b>:</p> <ul style="list-style-type: none"> <li><b>D-Sub</b> (type: <b>D-SubCable-to-BoardConnectorClassificationType</b>)</li> <li><b>DisplayPort</b> (type: <b>DisplayPortCable-to-BoardConnectorClassificationType</b>)</li> <li><b>USB</b> (type: <b>USBCable-to-BoardConnectorClassificationType</b>)</li> <li><b>HDMI</b> (type: <b>HDMICable-to-BoardConnectorClassificationType</b>)</li> <li><b>SerialCable</b> (type: <b>SerialCable-to-BoardConnectorClassificationType</b>)</li> <li><b>VGA</b> (type: <b>VGACable-to-BoardConnectorClassificationType</b>)</li> <li><b>Other</b> (type: <b>OtherAudioVideoDisplayCable-to-BoardConnectorClassificationType</b>)</li> </ul>
type	<b>AudioVideoDisplayCable-to-BoardConnectorClassificationType</b> , <b>D-SubCable-to-BoardConnectorClassificationType</b> , <b>DisplayPortCable-to-BoardConnectorClassificationType</b> , <b>USBCable-to-BoardConnectorClassificationType</b> , <b>HDMICable-to-BoardConnectorClassificationType</b> , <b>SerialCable-to-BoardConnectorClassificationType</b> , <b>VGACable-to-BoardConnectorClassificationType</b> , <b>OtherAudioVideoDisplayCable-to-BoardConnectorClassificationType</b> , <b>AudioVideoDisplayCable-to-BoardConnectorClassificationProperty-ArrayType</b> .

##### 4.5.1.2.2.1.1. D-Sub Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/AudioVideoDisplay/D-Sub</b>
diagram	<p>The diagram illustrates the structure of the <b>D-SubCable-to-BoardConnectorClassificationType</b>. It features a <b>Property-Array</b> (type: <b>D-SubCable-to-BoardConnectorClassificationProperty-ArrayType</b>) that contains:</p> <ul style="list-style-type: none"> <li><b>CableStyle</b> (type: <b>D-SubCable-to-BoardConnectorCableStylePropertyType</b>)</li> <li><b>Property</b> (type: <b>PropertyKeyValuePairType</b>)</li> </ul>
type	<b>D-SubCable-to-BoardConnectorClassificationType</b> , <b>D-SubCable-to-BoardConnectorClassificationProperty-ArrayType</b> , <b>D-SubCable-to-BoardConnectorCableStylePropertyType</b> , <b>PropertyKeyValuePairType</b> .

See Table 1 – Cable Style Configuration for choice of cables applicable to *D-Sub* connector.

#### 4.5.1.2.2.1.2. Display Port Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/AudioVideoDisplay/DisplayPort</b>
diagram	
type	<b>DisplayPortCable-to-BoardConnectorClassificationType, DisplayPortCable-to-BoardConnectorClassificationProperty-ArrayType, DisplayPortCable-to-BoardConnectorCableStylePropertyType, PropertyKeyValuePairType.</b>

See Table 1 – Cable Style Configuration for choice of cables applicable to *DisplayPort* connector.

#### 4.5.1.2.2.1.3. USB Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/AudioVideoDisplay/USB</b>
diagram	
type	<b>USB-Cable-to-BoardConnectorClassificationType, USB-Cable-to-BoardConnectorClassificationProperty-ArrayType, USB-Cable-to-BoardConnectorCableStylePropertyType, PropertyKeyValuePairType.</b>

See Table 1 – Cable Style Configuration for choice of cables applicable to *USB* connector.

#### 4.5.1.2.2.1.4. HDMI Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/AudioVideoDisplay/HDMI</b>
diagram	
type	<b>HDMI-Cable-to-BoardConnectorClassificationType, HDMI-Cable-to-BoardConnectorClassificationProperty-ArrayType, HDMI-Cable-to-BoardConnectorCableStylePropertyType, PropertyKeyValuePairType.</b>

See Table 1 – Cable Style Configuration for choice of cables applicable to *HDMI* connector.

#### 4.5.1.2.2.1.5. Serial Cable Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/AudioVideoDisplay/SerialCable</b>
diagram	
type	<b>SerialCable-to-BoardConnectorClassificationType, SerialCable-to-BoardConnectorClassificationProperty-ArrayType, SerialCable-to-BoardConnectorCableStylePropertyType, PropertyKeyValueType.</b>

See Table 1 – Cable Style Configuration for choice of cables applicable to a *Serial Cable* connector.

#### 4.5.1.2.2.1.6. VGA Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/AudioVideoDisplay/VGA</b>
diagram	
type	<b>VGA-Cable-to-BoardConnectorClassificationType, VGA-Cable-to-BoardConnectorClassificationProperty-ArrayType, VGA-Cable-to-BoardConnectorCableStylePropertyType, PropertyKeyValueType.</b>

See Table 1 – Cable Style Configuration for choice of cables applicable to *VGA* connector.



#### 4.5.1.2.2.1.7. Audio Video Display Cable-to-Board Other Connector Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/AudioVideoDisplay/Other</a>
diagram	<pre> classDiagram     class OtherAudioVideoDisplayCable-to-BoardConnectorClassificationType {         Property-Array         ClassificationProperties         CableStyle         PropertyKeyValueType     }     class OtherAudioVideoDisplayCable-to-BoardConnectorClassificationProperty-ArrayType {         PropertyKeyValueType     }     class OtherAudioVideoDisplayCable-to-BoardConnectorCableStylePropertyType {         PropertyKeyValueType     }     OtherAudioVideoDisplayCable-to-BoardConnectorClassificationType --&gt; OtherAudioVideoDisplayCable-to-BoardConnectorClassificationProperty-ArrayType     OtherAudioVideoDisplayCable-to-BoardConnectorClassificationType --&gt; OtherAudioVideoDisplayCable-to-BoardConnectorCableStylePropertyType     </pre>
type	<a href="#">OtherAudioVideoDisplayCable-to-BoardConnectorClassificationType</a> , <a href="#">OtherAudioVideoDisplayCable-to-BoardConnectorClassificationProperty-ArrayType</a> , <a href="#">OtherAudioVideoDisplayCable-to-BoardConnectorCableStylePropertyType</a> , <a href="#">PropertyKeyValueType</a> .

The CableStyle has the following configuration for each of the AudioVideoDisplay Cable-to-Board Connectors. The *Other* category can be specified as a string.

**Table 1 – Cable Style Configuration**

Cable Style	D-Sub	Display Port	USB	HDMI	Serial Cable	VGA	Other
Coaxial					√		√
Flat Cable	√	√	√			√	√
Ribbon Cable	√						√
Round Cable	√	√	√	√	√	√	√
Twisted Pair	√	√	√	√	√	√	√
Other	√	√	√	√	√	√	√

#### 4.5.1.2.2.2. Cable-to-Board Backplane Connector Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/Backplane</a>
diagram	<pre> classDiagram     class BackplaneCable-to-BoardConnectorClassificationType {         Vertical         RightAngle         Orthogonal         Property-Array     }     class VerticalBackplaneCable-to-BoardConnectorClassificationType     class RightAngleBackplaneCable-to-BoardConnectorClassificationType     class OrthogonalBackplaneCable-to-BoardConnectorClassificationType     class BackplaneCable-to-BoardConnectorClassificationProperty-ArrayType {         PropertyKeyValueType     }     BackplaneCable-to-BoardConnectorClassificationType --&gt; VerticalBackplaneCable-to-BoardConnectorClassificationType     BackplaneCable-to-BoardConnectorClassificationType --&gt; RightAngleBackplaneCable-to-BoardConnectorClassificationType     BackplaneCable-to-BoardConnectorClassificationType --&gt; OrthogonalBackplaneCable-to-BoardConnectorClassificationType     BackplaneCable-to-BoardConnectorClassificationType --&gt; BackplaneCable-to-BoardConnectorClassificationProperty-ArrayType     </pre>
type	<a href="#">BackplaneCable-to-BoardConnectorClassificationType</a> , <a href="#">VerticalBackplaneCable-to-BoardConnectorClassificationType</a> , <a href="#">RightAngleBackplaneCable-to-BoardConnectorClassificationType</a> , <a href="#">OrthogonalBackplaneCable-to-BoardConnectorClassificationType</a> , <a href="#">BackplaneCable-to-BoardConnectorClassificationProperty-ArrayType</a> .

#### 4.5.1.2.2.2.1. Cable-to-Board Backplane Connector Classification Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/ Cable-to-Board/Backplane/Property-Array</a>
diagram	<p>The diagram illustrates the structure of the <b>BackplaneCable-to-BoardConnectorClassificationProperty-ArrayType</b>. It is a <b>Property-Array</b> (type: <b>BackplaneCable-to-BoardConn...</b>) containing an <b>attributes</b> element and a complex element. The complex element is a choice between <b>DifferentialPair</b> (type: <b>EmptyType</b>) and a choice between <b>MountingTechnology</b> (type: <b>SMD-TH-PF-MountingTechnolog...</b>), <b>ConnectorRetention</b> (type: <b>EmptyType</b>), and <b>Property</b> (type: <b>PropertyKeyValuePairType</b>). The <b>Property</b> element is marked with a cardinality of <b>0..∞</b>.</p>
type	<a href="#">BackplaneCable-to-BoardConnectorClassificationProperty-ArrayType</a> , <a href="#">SMD-TH-PF-MountingTechnologyType</a> , <a href="#">PropertyKeyValuePairType</a> .

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.

The [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.

#### 4.5.1.2.2.3. Cable-to-Board PowerAndSignal Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/PowerAndSignal</b>
diagram	
type	<b>PowerAndSignalCable-to-BoardConnectorClassificationType, PowerAndSignalCable-to-BoardConnectorClassificationProperty-ArrayType, PowerRatingType, PowerRatingUOMType, JEP30-D10:EmptyType, PropertyKeyValuePairType.</b>

The *PowerRating/UOM* can be specified in *A* (Amperes), *V* (Voltage) or *W* (Watts).

##### 4.5.1.2.2.3.1. Cable-to-Board Power Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/Power</b>
diagram	
type	<b>PowerCable-to-BoardConnectorClassificationType, PowerCable-to-BoardConnectorClassificationProperty-ArrayType, PowerRatingType, JEP30-D10:EmptyType, PropertyKeyValuePairType.</b>

#### 4.5.1.2.2.4. Cable-to-Board Signal Connector Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/Signal</a>
diagram	
type	<a href="#">SignalCable-to-BoardConnectorClassificationType</a> , <a href="#">SignalCable-to-BoardConnectorClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValuePairType</a> .

#### 4.5.1.2.2.5. Cable-to-Board Other Connector Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Board/Other</a>
diagram	
type	<a href="#">OtherCable-to-BoardConnectorClassificationType</a> , <a href="#">OtherCable-to-BoardConnectorClassificationProperty-ArrayType</a> , <a href="#">OtherCable-to-BoardConnectorClassificationPropertyCableStyleType</a> , <a href="#">PropertyKeyValuePairType</a> .

The *CableStyle* via type *OtherCable-to-BoardConnectorClassificationPropertyCableStyleType* has the same full enumeration values consisting of *Coaxial*, *FlatCable*, *RibbonCable*, *RoundCable*, *TwistedPair* and *Other* where *Other* can be specified as a string

#### 4.5.1.2.3. Cable-to-Cable Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Cable</b>
diagram	
type	<b>Cable-to-CableConnectorClassificationType, CoaxialCable-to-CableConnectorClassificationType, PowerAndSignalCable-to-CableConnectorClassificationType, PowerCable-to-CableConnectorClassificationType, SignalCable-to-CableConnectorClassificationType, RFCable-to-CableConnectorClassificationType, OtherCable-to-CableConnectorClassificationType, Cable-to-CableConnectorClassificationProperty-ArrayType, PropertyKeyValuePair</b>

#### 4.5.1.2.3.1. Cable-to-Cable PowerAndSignal Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Cable/PowerAndSignal</b>
diagram	
type	<b>PowerAndSignalCable-to-CableConnectorClassificationType, PowerAndSignalCable-to-CableConnectorClassificationProperty-ArrayType, PowerRatingType, JEP30-D10:EmptyType, PropertyKeyValuePairType.</b>

#### 4.5.1.2.3.2. Cable-to-Cable Power Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Cable/Power</b>
diagram	
type	<b>PowerCable-to-CableConnectorClassificationType, PowerCable-to-CableConnectorClassificationProperty-ArrayType, PowerRatingType, JEP30-D10:EmptyType, PropertyKeyValuePairType.</b>

#### 4.5.1.2.3.3. Cable-to-Cable Signal Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Cable/Signal</b>
diagram	
type	<b>SignalCable-to-CableConnectorClassificationType, SignalCable-to-CableConnectorClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValuePairType.</b>

#### 4.5.1.2.3.4. Other Cable-to-Cable Connector Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Cable-to-Cable/Other</a>
diagram	
type	<a href="#">OtherCable-to-CableConnectorClassificationType</a> , <a href="#">OtherCable-to-CableConnectorClassificationProperty-ArrayType</a> , <a href="#">OtherCable-to-CableConnectorClassificationPropertyCableStyleType</a> , <a href="#">PropertyKeyValuePairType</a> .

The [CableStyle](#) via type [OtherCable-to-CableConnectorClassificationPropertyCableStyleType](#) has the same full enumeration values consisting of [Coaxial](#), [FlatCable](#), [RibbonCable](#), [RoundCable](#), [TwistedPair](#) and [Other](#) where [Other](#) can be specified as a string

#### 4.5.1.2.4. CardEdge-to-Board Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/CardEdge-to-Board</a>
diagram	
type	<a href="#">CardEdge-to-BoardConnectorClassificationType</a> , <a href="#">BackplaneCardEdge-to-BoardConnectorClassificationType</a> , <a href="#">InterfaceCardEdge-to-BoardConnectorClassificationType</a> , <a href="#">MemoryCardEdge-to-BoardConnectorClassificationType</a> , <a href="#">OtherCardEdge-to-BoardConnectorClassificationType</a> , <a href="#">CardEdge-to-BoardConnectorClassificationProperty-ArrayType</a> , <a href="#">PropertyKeyValuePair</a> .

#### 4.5.1.2.5. CardEdge-to-Cable Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/CardEdge-to-Cable</a>
diagram	
type	<a href="#">CardEdge-to-CableConnectorClassificationType</a> , <a href="#">InterfaceCardEdge-to-CableConnectorClassificationType</a> , <a href="#">OpticalCardEdge-to-CableConnectorClassificationType</a> , <a href="#">OtherCardEdge-to-CableConnectorClassificationType</a> , <a href="#">CardEdge-to-CableConnectorClassificationProperty-ArrayType</a> .

##### 4.5.1.2.5.1. CardEdge-to-Cable Interface Connector Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/CardEdge-to-Cable/Interface</a>
diagram	
type	<a href="#">InterfaceCardEdge-to-CableConnectorClassificationType</a> , <a href="#">InterfaceCardEdge-to-CableConnectorClassificationProperty-ArrayType</a> , <a href="#">InterfaceCardEdge-to-CableConnectorClassificationPropertyCableStyleType</a> , <a href="#">PropertyKeyValuePairType</a> .



#### 4.5.1.2.5.2. CardEdge-to-Cable Other Connector Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/CardEdge-to-Cable/Other</b>
diagram	
type	<b>OtherCardEdge-to-CableConnectorClassificationType, OtherCardEdge-to-CableConnectorClassificationProperty-ArrayType, OtherCardEdge-to-CableConnectorClassificationPropertyCableStyleType, PropertyKeyValuePairType.</b>

#### 4.5.1.2.6. Optical Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Optical</b>
diagram	
type	<b>OpticalConnectorClassificationType, OpticalConnectorClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValuePair</b>

4.5.1.2.7.      **Socket Classification**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Socket</a>
diagram	<p>SocketConnectorClassificationType</p> <p>Socket</p> <p>type SocketConnectorClassificationType</p> <p>LIF</p> <p>type LIF-SocketConnectorClassificationType</p> <p>ZIF</p> <p>type ZIF-SocketConnectorClassificationType</p> <p>LGA</p> <p>type LGA-SocketConnectorClassificationType</p> <p>BGA</p> <p>type BGA-SocketConnectorClassificationType</p> <p>OtherPCB-Mountable</p> <p>type OtherPCB-MountableSocketConnectorClassificationType</p> <p>OtherNon-PCB-Mountable</p> <p>type OtherNon-PCB-MountableSocketConnectorClassificationType</p> <p>Property-Array</p> <p>type SocketConnectorClassificationProperty-ArrayType</p> <p>SocketConnectorClassificationProperty-ArrayType</p> <p>ClassificationProperties</p> <p>Property</p> <p>type PropertyKeyValuePairType</p> <p>1..∞</p>
type	<a href="#">SocketConnectorClassificationType</a> , <a href="#">LIF-SocketConnectorClassificationType</a> , <a href="#">ZIF-SocketConnectorClassificationType</a> , <a href="#">LGA-SocketConnectorClassificationType</a> , <a href="#">BGA-SocketConnectorClassificationType</a> , <a href="#">OtherPCB-MountableSocketConnectorClassificationType</a> , <a href="#">OtherNon-PCB-MountableSocketConnectorClassificationType</a> , <a href="#">SocketConnectorClassificationProperty-ArrayType</a> , <a href="#">PropertyKeyValuePair</a>

4.5.1.2.7.1.    **LIF Socket Classification**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Socket/LIF</a>
diagram	<p>LIF-SocketConnectorClassificationType</p> <p>LIF</p> <p>type LIF-SocketConnectorClassificationType</p> <p>Property-Array</p> <p>type LIF-SocketConnectorClassificationProperty-ArrayType</p> <p>LIF-SocketConnectorClassificationProperty-ArrayType</p> <p>ClassificationProperties</p> <p>MountingTechnology</p> <p>type SMD-TH-MountingTechnologyType</p> <p>Property</p> <p>type PropertyKeyValuePairType</p> <p>0..∞</p>
type	<a href="#">LIF-SocketConnectorClassificationType</a> , <a href="#">LIF-SocketConnectorClassificationProperty-ArrayType</a> , <a href="#">SMD-TH-MountingTechnologyType</a> , <a href="#">PropertyKeyValuePairType</a> .

The *MountingTechnology* element has enumerated values of *SMD* and *Thru-Hole*.

#### 4.5.1.2.7.2. ZIF Socket Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Socket/ZIF</b>
diagram	
type	<b>ZIF-SocketConnectorClassificationType, ZIF-SocketConnectorClassificationProperty-ArrayType, SMD-TH-MountingTechnologyType, PropertyKeyValuePairType.</b>

#### 4.5.1.2.7.3. Other PCB Mountable Socket Classification

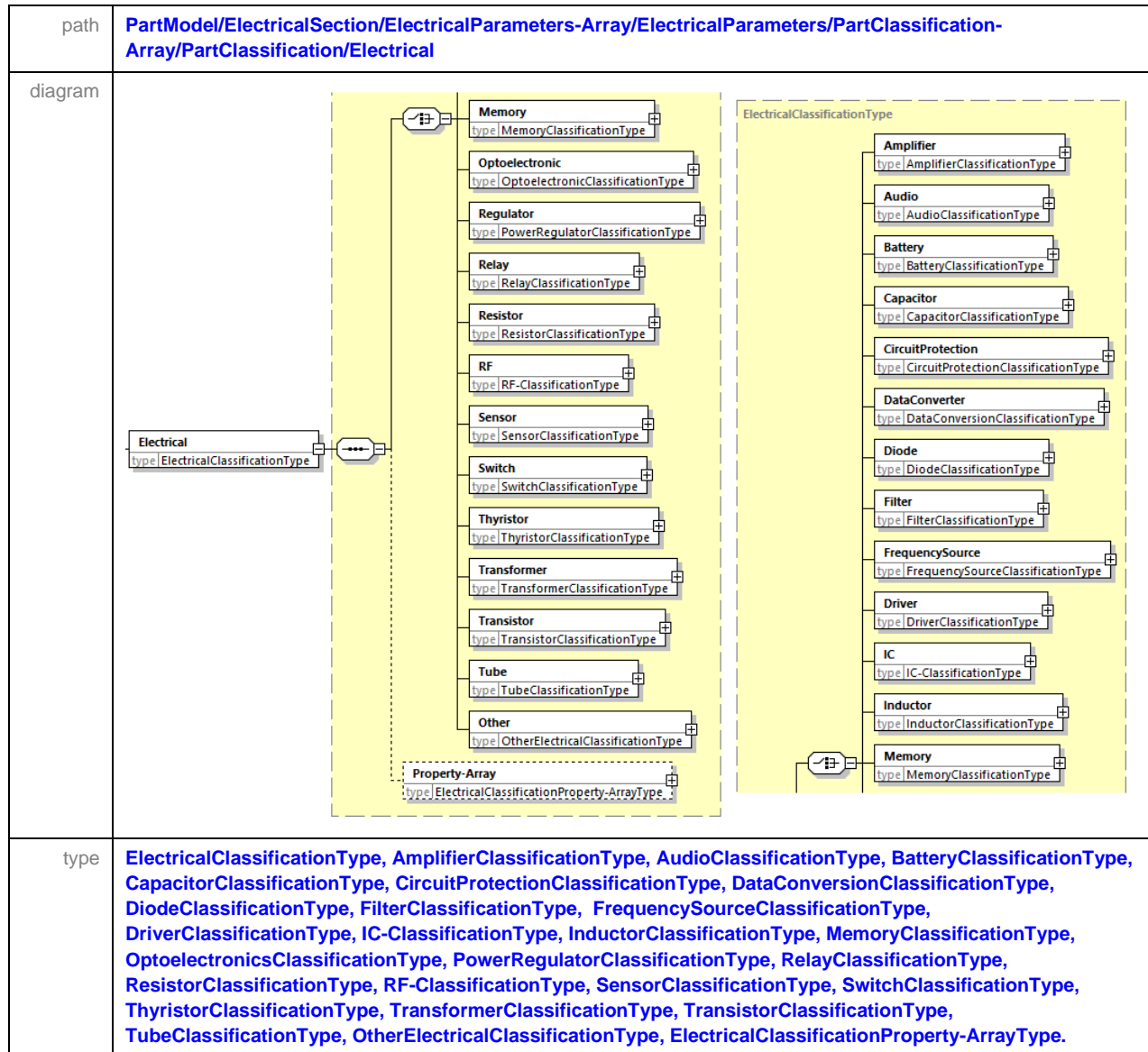
path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Socket/OtherPCB-Mountable</b>
diagram	
type	<b>OtherPCB-MountableSocketConnectorClassificationType, OtherPCB-MountableSocketConnectorClassificationProperty-ArrayType, SMD-TH-MountingTechnologyType, PropertyKeyValuePairType.</b>

4.5.1.2.8. Other Connector Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Connector/Other
diagram	
type	OtherConnectorClassificationType, OtherConnectorClassificationProperty-ArrayType, PropertyKeyValuePair

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.

### 4.5.1.3. Electrical Classification



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.

#### 4.5.1.3.1. Amplifier Classification

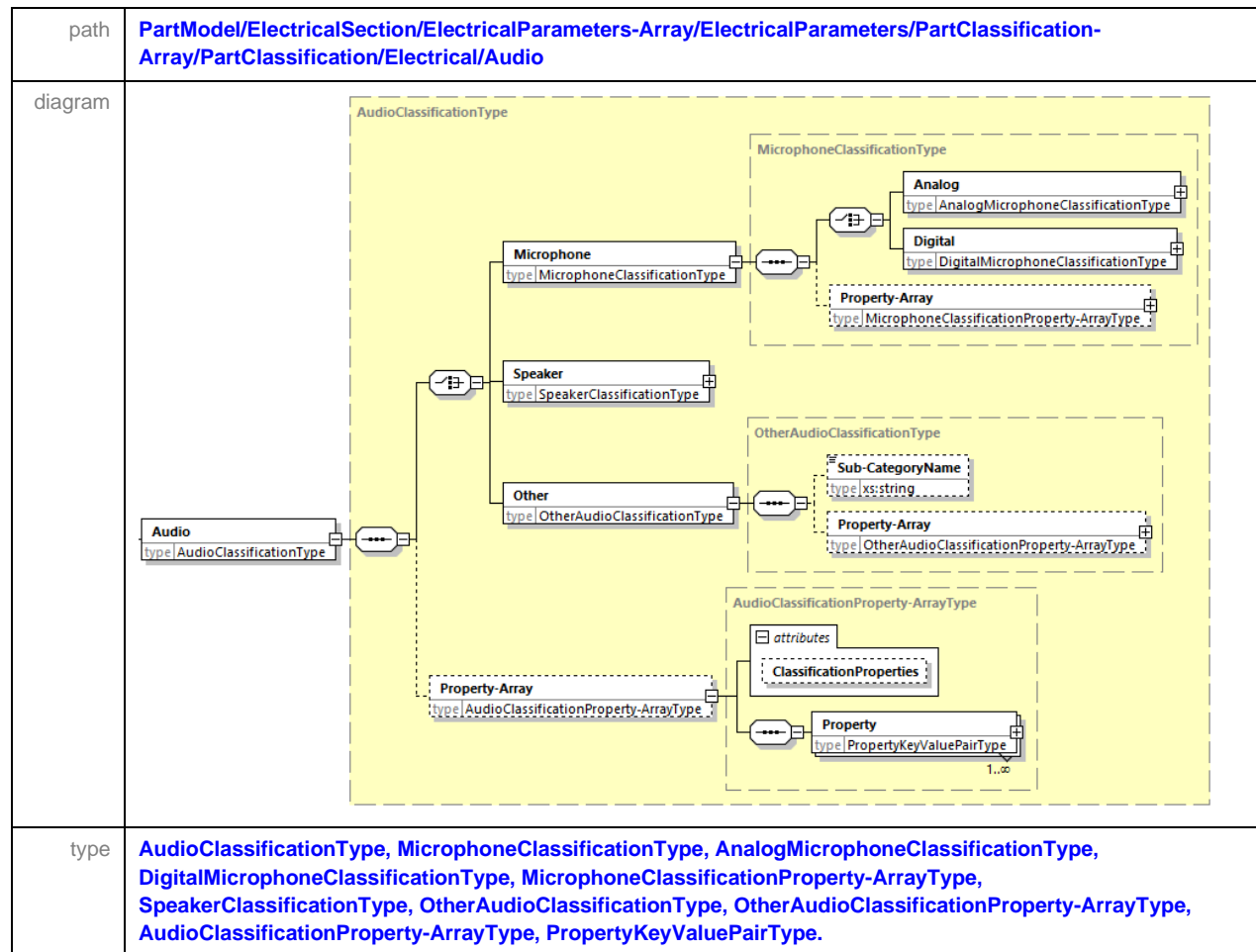
path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Amplifier
diagram	<p>The diagram illustrates the classification structure for an amplifier. An <b>Amplifier</b> entity (type: <code>AmplifierClassificationType</code>) is linked to a collection of specific amplifier types. These types are organized within a container labeled <b>AmplifierClassificationType</b>. The sub-classifications include:</p> <ul style="list-style-type: none"> <li><b>Audio</b> (type: <code>AudioAmplifierClassificationType</code>)</li> <li><b>Comparator</b> (type: <code>ComparatorAmplifierClassificationType</code>)</li> <li><b>Instrumentation</b> (type: <code>InstrumentationAmplifierClassificationType</code>)</li> <li><b>Isolation</b> (type: <code>IsolationAmplifierClassificationType</code>)</li> <li><b>Logarithmic</b> (type: <code>LogarithmicAmplifierClassificationType</code>)</li> <li><b>OperationalAmplifier</b> (type: <code>OperationalAmplifierClassificationType</code>)</li> <li><b>PowerAmplifier</b> (type: <code>PowerAmplifierClassificationType</code>)</li> <li><b>RF</b> (type: <code>RF-AmplifierClassificationType</code>)</li> <li><b>Specialty</b> (type: <code>SpecialtyAmplifierClassificationType</code>)</li> <li><b>VariableGain</b> (type: <code>VariableGainAmplifierClassificationType</code>)</li> <li><b>Video</b> (type: <code>VideoAmplifierClassificationType</code>)</li> <li><b>Other</b> (type: <code>OtherAmplifierClassificationType</code>)</li> <li><b>Property-Array</b> (type: <code>AmplifierClassificationProperty-ArrayType</code>)</li> </ul>
type	AmplifierClassificationType, AudioAmplifierClassificationType, ComparatorAmplifierClassificationType, InstrumentationAmplifierClassificationType, IsolationAmplifierClassificationType, LogarithmicAmplifierClassificationType, OperationalAmplifierClassificationType, PowerAmplifierClassificationType, RF-AmplifierClassificationType, SpecialtyAmplifierClassificationType, VariableGainAmplifierClassificationType, VideoAmplifierClassificationType, OtherAmplifierClassificationType, AmplifierClassificationProperty-ArrayType

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

#### 4.5.1.3.1.1. Other Amplifier Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Amplifier</a>
diagram	
type	<a href="#">OtherAmplifierClassificationType</a> , <a href="#">OtherAmplifierClassificationProperty-ArrayType</a> , <a href="#">PropertyKeyValuePair</a>

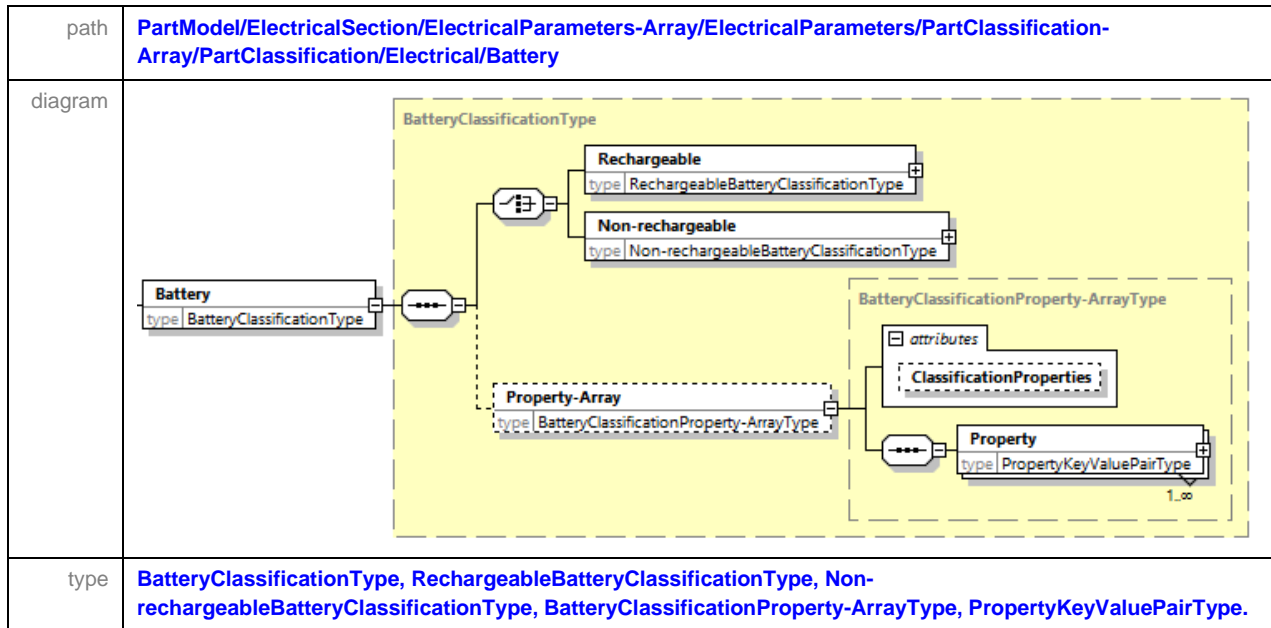
Classifications under the category *Other* are candidates for future standardization via this publication. The *Sub-CategoryName* enables the component manufacturer to propose specific sub-classifications via this structure.

**4.5.1.3.2. Audio Classification**

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

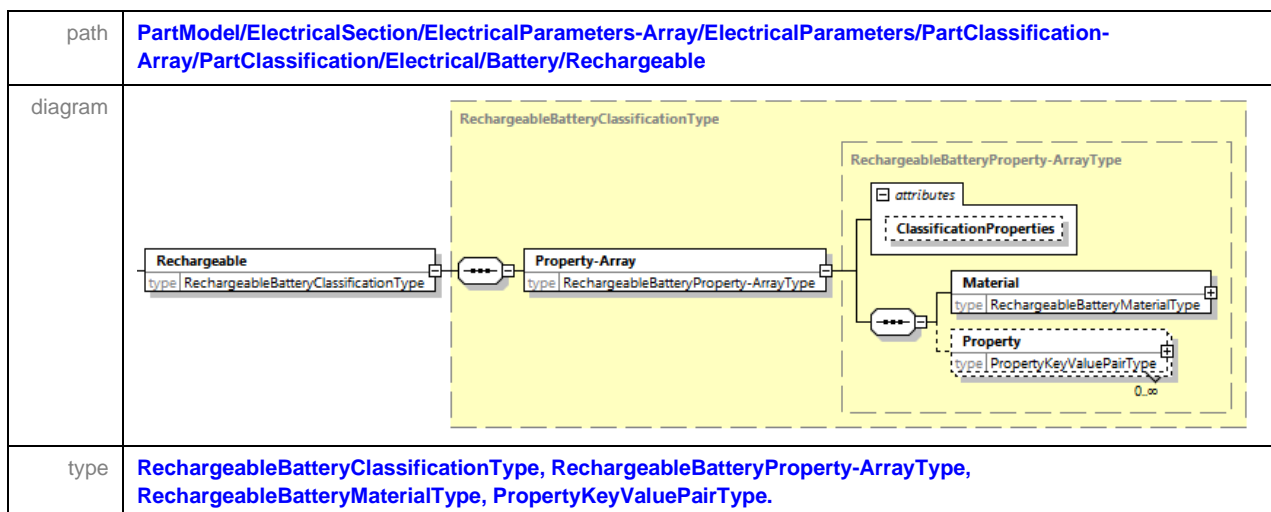


### 4.5.1.3.3. Battery Classification



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.

#### 4.5.1.3.3.1. Rechargeable Battery Classification



#### 4.5.1.3.3.1.1. Rechargeable Battery Material Property

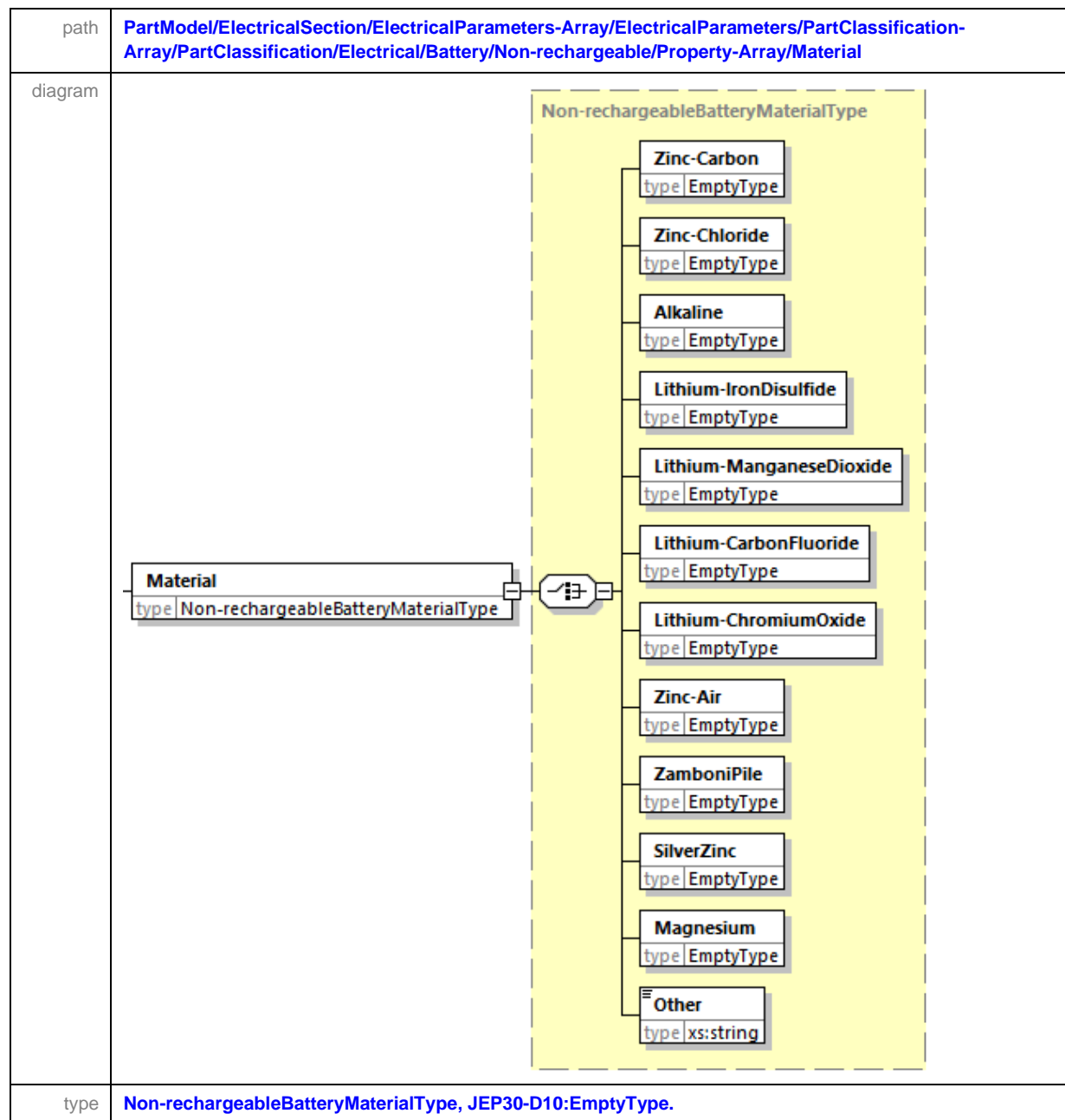
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Battery/Rechargeable/Property-Array/Material</a>
diagram	<p>The diagram illustrates the <b>Material</b> property of a Rechargeable Battery. The property is represented by a box labeled <b>Material</b> with the type <code>RechargeableBatteryMaterialType</code>. This property is linked to a dashed yellow box titled <b>RechargeableBatteryMaterialType</b>, which contains a list of material types. Each material type is represented by a box with its name and a type attribute. The types are:</p> <ul style="list-style-type: none"> <li><b>Nickel-Cadmium</b>: type <code>EmptyType</code></li> <li><b>Lead-Acid</b>: type <code>EmptyType</code></li> <li><b>Nickel-MetalHydride</b>: type <code>EmptyType</code></li> <li><b>Nickel-Zinc</b>: type <code>EmptyType</code></li> <li><b>Silver-Zinc</b>: type <code>EmptyType</code></li> <li><b>Lithium-Iron-Phosphate</b>: type <code>EmptyType</code></li> <li><b>LithiumIonLithiumCobaltOxide</b>: type <code>EmptyType</code></li> <li><b>LithiumIonManganeseOxideBattery</b>: type <code>EmptyType</code></li> <li><b>LithiumIonPolymerBattery</b>: type <code>EmptyType</code></li> <li><b>Other</b>: type <code>xs:string</code></li> </ul>
type	<a href="#">RechargeableBatteryMaterialType</a> , <a href="#">JEP30-D10:EmptyType</a> .

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*.

4.5.1.3.3.2. Non-rechargeable Battery Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Battery/Non-rechargeable
diagram	
type	Non-rechargeableBatteryClassificationType, Non-rechargeableBatteryProperty-ArrayType, Non-rechargeableBatteryMaterialType, PropertyKeyValuePairType.

## 4.5.1.3.3.2.1. Non-rechargeable Battery Material Property



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*.

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor
diagram	<pre> classDiagram     class Capacitor {         type CapacitorClassificationType     }     class Fixed {         type FixedCapacitorClassificationType     }     class Variable {         type VariableCapacitorClassificationType     }     class PropertyArray1 {         type CapacitorClassificationProperty-ArrayType     }     class Electrolytic {         type ElectrolyticCapacitorClassificationType     }     class Film {         type FilmCapacitorClassificationType     }     class Silicon {         type SiliconCapacitorClassificationType     }     class SuperCap {         type Super-capCapacitorClassificationType     }     class Other {         type OtherFixedCapacitorClassificationType     }     class Ceramic {         type CeramicCapacitorClassificationType     }     class PropertyArray2 {         type FixedCapacitorClassificationProperty-ArrayType     }      Capacitor "1" -- "*" Fixed     Capacitor "1" -- "*" Variable     Fixed "1" -- "*" Electrolytic     Fixed "1" -- "*" Film     Fixed "1" -- "*" Silicon     Fixed "1" -- "*" SuperCap     Fixed "1" -- "*" Other     Fixed "1" -- "*" Ceramic     Fixed "1" -- "*" PropertyArray2     Variable "1" -- "*" PropertyArray1     </pre> <p>The diagram illustrates the classification hierarchy for capacitors. The <b>Capacitor</b> class is the root, with a <code>type</code> attribute of <code>CapacitorClassificationType</code>. It is associated with <b>Fixed</b> and <b>Variable</b> classes. The <b>Fixed</b> class has a <code>type</code> attribute of <code>FixedCapacitorClassificationType</code> and is associated with <b>Electrolytic</b>, <b>Film</b>, <b>Silicon</b>, <b>Super-cap</b>, <b>Other</b>, and <b>Ceramic</b> classes. The <b>Variable</b> class has a <code>type</code> attribute of <code>VariableCapacitorClassificationType</code> and is associated with a <b>Property-Array</b> class. The <b>Property-Array</b> class has a <code>type</code> attribute of <code>CapacitorClassificationProperty-ArrayType</code>. The <b>Fixed</b> class is also associated with a <b>Property-Array</b> class, which has a <code>type</code> attribute of <code>FixedCapacitorClassificationProperty-ArrayType</code>.</p>
type	CapacitorClassificationType, FixedCapacitorClassificationType, CeramicCapacitorClassificationType, ElectrolyticCapacitorClassificationType, FilmCapacitorClassificationType, SiliconCapacitorClassificationType, Super-capCapacitorClassificationType, OtherFixedCapacitorClassificationType, FixedCapacitorClassificationProperty-ArrayType, VariableCapacitorClassificationType, CapacitorClassificationProperty-ArrayType.

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

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

4.5.1.3.4.1. Ceramic Capacitor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Ceramic
diagram	
type	CeramicCapacitorClassificationType, CeramicCapacitorClassificationProperty-ArrayType, JEP30-D10:EmptyType, CeramicCapacitorClassType, CeramicCapacitorDielectricType, PropertyKeyValuePairType.

Ceramic capacitors are often consolidated in array form, and this can be additionally classified via the optional element Array.



Figure 1 - Ceramic Capacitor Array

#### 4.5.1.3.4.1.1. Ceramic Capacitor Class Property

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Ceramic/Property-Array/Class</a>
diagram	<p>The diagram shows a class hierarchy for <b>CeramicCapacitorClassType</b>. A dashed box labeled <b>Class</b> with <code>type CeramicCapacitorClassType</code> is connected to a central node. This node branches into three sub-classes: <b>Class1</b> (type EmptyType), <b>Class2</b> (type EmptyType), and <b>Other</b> (type xs:string).</p>
type	<a href="#">CeramicCapacitorClassType</a> , JEP30-D10:EmptyType.

Most [Ceramic](#) capacitors can be sub-classified as either [Class1](#) or [Class2](#), however some can be specified under the category [Other](#).

#### 4.5.1.3.4.1.2. Ceramic Capacitor Dielectric Property

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Ceramic/Property-Array/Dielectric</a>
diagram	<p>The diagram shows a class hierarchy for <b>CeramicCapacitorDielectricType</b>. A dashed box labeled <b>Dielectric</b> with <code>type CeramicCapacitorDielectricType</code> is connected to a central node. This node branches into two columns of sub-classes. The left column includes: <b>P100</b>, <b>NP0</b>, <b>N33</b>, <b>N75</b>, <b>N150</b>, <b>N220</b>, <b>N330</b>, <b>N470</b>, <b>N750</b>, and <b>N1000</b>, all with <code>type EmptyType</code>. The right column includes: <b>N1500</b>, <b>X5R</b>, <b>X6R</b>, <b>X7R</b>, <b>X7S</b>, <b>X8R</b>, <b>Y5V</b>, <b>Z5U</b>, and <b>Other</b> (type xs:string), all with <code>type EmptyType</code>.</p>
type	<a href="#">CeramicCapacitorDielectricType</a> , JEP30-D10:EmptyType.

4.5.1.3.4.2. Electrolytic Capacitor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Electrolytic
diagram	
type	ElectrolyticCapacitorClassificationType, ElectrolyticCapacitorClassificationProperty-ArrayType, JEP30-D10:EmptyType, ElectrolyticCapacitorMaterialType, ElectrolyticCapacitorElectrolyteType, PropertyKeyValuePairType.

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



Figure 2— Fused Electrolytic Capacitor

4.5.1.3.4.2.1. Electrolytic Capacitor Material Property

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Electrolytic/Property-Array/Material
diagram	
type	ElectrolyticCapacitorMaterialType, JEP30-D10:EmptyType.



#### 4.5.1.3.4.2.1 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*.

#### 4.5.1.3.4.2.2. Electrolytic Capacitor Electrolyte Property

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Electrolytic/Property-Array/Electrolyte</a>
diagram	<pre> classDiagram     class Electrolyte {         type ElectrolyticCapacitorElectrolyteType     }     class ElectrolyticCapacitorElectrolyteType {         +EthyleneGlycolBoraxElectrolyte type EmptyType         +MnO2Electrolyte type EmptyType         +MultianodeMnO2Electrolyte type EmptyType         +OrganicPolymer type EmptyType         +PolymerElectrolyte type EmptyType         +PolymerNonSolidElectrolyte type EmptyType         +WaterBasedElectrolyte type EmptyType         +MultianodePolymerElectrolyte type EmptyType     }     Electrolyte --&gt; ElectrolyticCapacitorElectrolyteType   </pre>
type	<a href="#">ElectrolyticCapacitorElectrolyteType</a> , JEP30-D10:EmptyType.

#### 4.5.1.3.4.3. Film Capacitor Classification and Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Film</a>
diagram	
type	<a href="#">FilmCapacitorClassificationType</a> , <a href="#">FilmCapacitorProperty-ArrayType</a> , <a href="#">FilmCapacitorDielectricType</a> , <a href="#">PropertyKeyValuePairType</a>

##### 4.5.1.3.4.3.1. Film Capacitor Dielectric Property

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Film/ Property-Array/Dielectric</a>
diagram	
type	<a href="#">FilmCapacitorDielectricType</a> , <a href="#">JEP30-D10:EmptyType</a> .

*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.

#### 4.5.1.3.4.4. Silicon Capacitor Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Silicon</a>
diagram	
type	<a href="#">SiliconCapacitorClassificationType</a> , <a href="#">SiliconCapacitorProperty-ArrayType</a> , <a href="#">PropertyKeyValuePairType</a> .

#### 4.5.1.3.4.5. Super-cap Capacitor Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Super-cap</a>
diagram	
type	<a href="#">Super-capCapacitorClassificationType</a> , <a href="#">Super-capCapacitorProperty-ArrayType</a> , <a href="#">Super-capCapacitorClassType</a> , <a href="#">PropertyKeyValuePairType</a> .

#### 4.5.1.3.4.5.1. Super-cap Capacitor Class Property

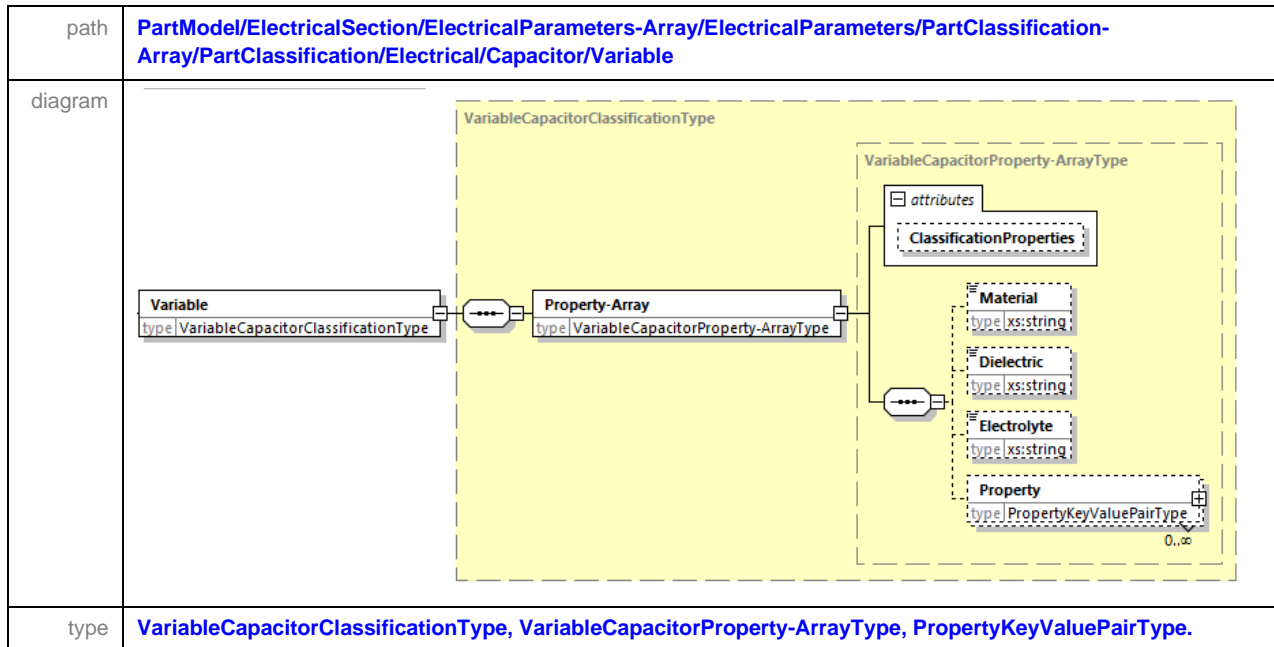
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Super-cap/Property-Array/Class</a>
diagram	
type	<a href="#">Super-capCapacitorClassificationType</a> .

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 a [Super-cap](#) as shown above.

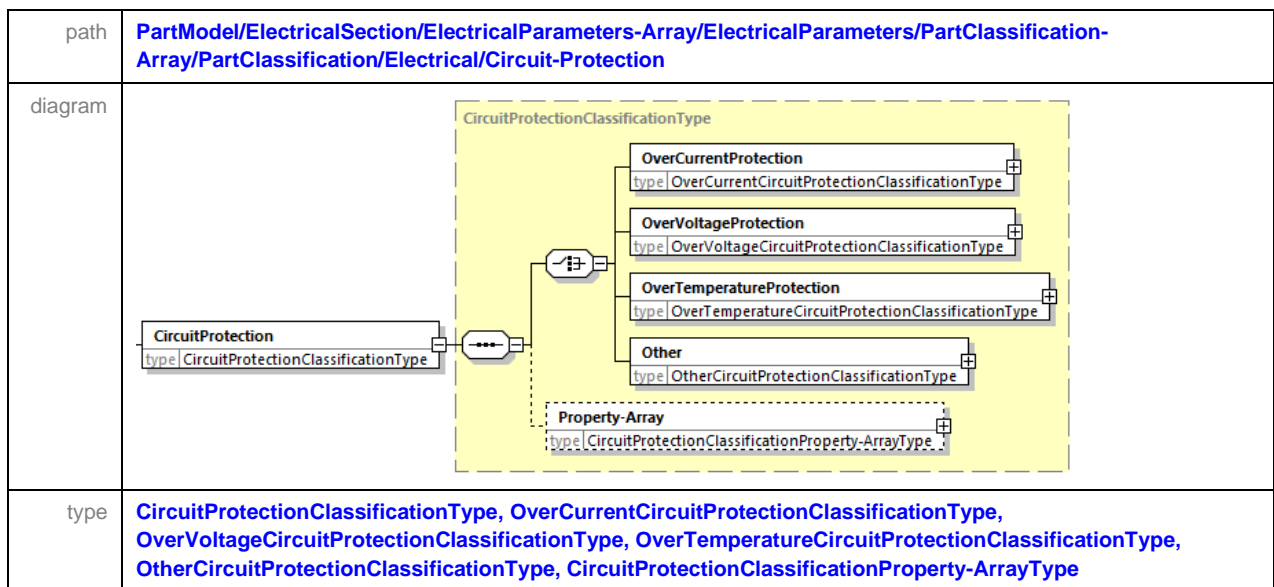
#### 4.5.1.3.4.6. Other Fixed Capacitor Classification and Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Capacitor/Fixed/Other/Property-Array/Class</a>
diagram	
type	<a href="#">OtherFixedCapacitorClassificationType</a> , <a href="#">OtherFixedCapacitorClassificationProperty-ArrayType</a> , <a href="#">PropertyKeyValuePairType</a> .

#### 4.5.1.3.4.7. Variable Capacitor Classification and Property-Array



#### 4.5.1.3.5. Circuit Protection Classification



*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. 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*.

#### 4.5.1.3.5.1. Over Current Circuit Protection Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection</a>
diagram	
type	<a href="#">OverCurrentCircuitProtectionClassificationType</a> , <a href="#">ResettableOverCurrentCircuitProtectionClassificationType</a> , <a href="#">Non-ResettableOverCurrentCircuitProtectionClassificationType</a> , <a href="#">OverCurrentCircuitProtectionClassificationProperty-ArrayType</a>

##### 4.5.1.3.5.1.1. Resettable Over Current Circuit Protection Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection/Resettable</a>
diagram	
type	<a href="#">ResettableOverCurrentCircuitProtectionClassificationType</a> , <a href="#">ResettableOverCurrentCircuitProtectionClassificationProperty-ArrayType</a> , <a href="#">ResettableOverCurrentCircuitProtectionTechnologyType</a> , <a href="#">PropertyKeyValuePairType</a> .

#### 4.5.1.3.5.1.2. Resettable Over Current Circuit Protection Technology Property

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

#### 4.5.1.3.5.1.3. Non-Resettable Over Current Circuit Protection Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection/Non-Resettable</a>
diagram	
type	<a href="#">Non-ResettableOverCurrentCircuitProtectionClassificationType</a> , <a href="#">Non-ResettableOverCurrentCircuitProtectionClassificationProperty-ArrayType</a> , <a href="#">Non-ResettableOverCurrentCircuitProtectionTechnologyType</a> , <a href="#">PropertyKeyValuePairType</a> .

#### 4.5.1.3.5.1.4. Non-Resettable Over Current Circuit Protection Technology Property

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverCurrentProtection/Non-Resettable/Property-Array/Technology</a>
diagram	
type	<a href="#">Non-ResettableOverCurrentCircuitProtectionTechnologyType</a> , <a href="#">JEP30-D10:EmptyType</a>

#### 4.5.1.3.5.2. Over Voltage Circuit Protection Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverVoltageProtection</a>
diagram	
type	<a href="#">OverVoltageCircuitProtectionClassificationType</a> , <a href="#">VaristorClassificationType</a> , <a href="#">AvalancheBreakdownDiodeClassificationType</a> , <a href="#">TransientVoltageSuppressorClassificationType</a> , <a href="#">ProtectiveGasDischargeTubeClassificationType</a> , <a href="#">SparkGapCircuitProtectionClassificationType</a> , <a href="#">OtherCircuitProtectionClassificationType</a> , <a href="#">OverVoltageCircuitProtectionClassificationProperty-ArrayType</a>



#### 4.5.1.3.5.2.1. Over Voltage Circuit Protection Varistor Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverVoltageProtection/Varistor</a>
diagram	
type	<a href="#">VaristorClassificationType</a> , <a href="#">VaristorClassificationProperty-ArrayType</a> , <a href="#">OverVoltageCircuitProtectionVaristorType</a> , <a href="#">PropertyKeyValuePairType</a>

#### 4.5.1.3.5.2.2. Over Voltage Circuit Protection Varistor Type Property

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverVoltageProtection/Varistor/Property-Array/VaristorType</a>
diagram	
type	<a href="#">OverVoltageCircuitProtectionVaristorType</a> .

#### 4.5.1.3.5.2.3. Over Voltage Circuit Protection Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/OverVoltageProtection/Other</a>
diagram	
type	<a href="#">OtherCircuitProtectionClassificationType</a> , <a href="#">OtherCircuitProtectionClassificationProperty-ArrayType</a> , <a href="#">PropertyKeyValuePairType</a>

#### 4.5.1.3.5.3. Over Temperature Circuit Protection Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/ OverTemperatureProtection</a>
diagram	
type	<a href="#">OverTemperatureCircuitProtectionClassificationType</a> , <a href="#">ThermalCutoffsCircuitProtectionClassificationType</a> , <a href="#">OtherCircuitProtectionClassificationType</a> , <a href="#">OverTemperatureProtectionClassificationProperty-ArrayType</a> .

#### 4.5.1.3.5.4. Other Circuit Protection Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/CircuitProtection/Other</a>
diagram	
type	<a href="#">OtherCircuitProtectionClassificationType</a> , <a href="#">OtherCircuitProtectionClassificationProperty-ArrayType</a> , <a href="#">PropertyKeyValuePairType</a>

#### 4.5.1.3.6. Data Converter Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/DataConverter</a>
diagram	
type	<a href="#">Data-ConversionClassificationType</a> , <a href="#">AnalogToDigitalConverterClassificationType</a> , <a href="#">DigitalToAnalogConverterClassificationType</a> , <a href="#">FrequencyToVoltageConverterClassificationType</a> , <a href="#">CurrentToVoltageConverterClassificationType</a> , <a href="#">OtherDataConversionClassificationType</a> , <a href="#">DataConversionClassificationProperty-ArrayType</a> , <a href="#">PropertyKeyValuePairType</a>

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

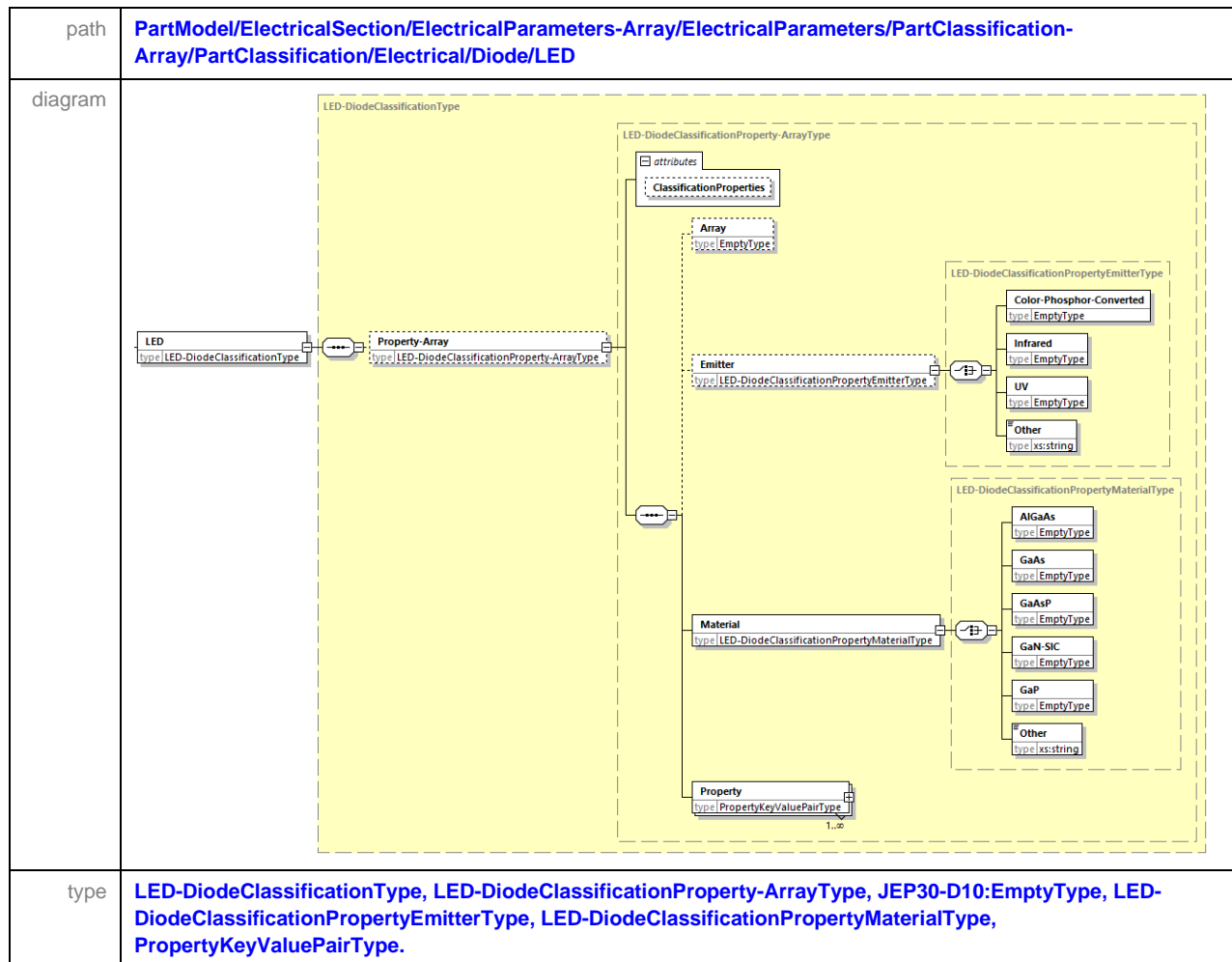
4.5.1.3.6.1. Data Converter Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/DataConverter/Other
diagram	<p>The diagram illustrates the structure of the <code>OtherDataConversionClassificationType</code>. It is a complex type containing two main elements:</p> <ul style="list-style-type: none"><li><code>Sub-CategoryName</code>: A string element of type <code>xs:string</code>.</li><li><code>Property-Array</code>: An array of <code>OtherDataConversionClassificationProperty-ArrayType</code> elements.</li></ul> <p>The <code>Property-Array</code> element is further detailed as containing an <code>attributes</code> section with <code>ClassificationProperties</code>. These properties are represented as an array of <code>Property</code> elements, where each <code>Property</code> is of type <code>PropertyKeyValueType</code>. The cardinality for the <code>Property</code> array is <code>1..∞</code>.</p>
type	OtherDataConversionClassificationType, OtherDataConversionClassificationProperty-ArrayType, PropertyKeyValueType.

#### 4.5.1.3.7. Diode Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode
diagram	<p>The diagram illustrates the structure of the Diode Classification. A central <b>Diode</b> class (type: <b>DiodeClassificationType</b>) is linked to a collection of specific diode types. These types are listed vertically within a dashed boundary labeled <b>DiodeClassificationType</b>. Each type has its own associated classification type (e.g., <b>BridgeRectifier</b> has <b>BridgeRectifierClassificationType</b>). Additionally, a <b>Property-Array</b> (type: <b>DiodeClassificationProperty-ArrayType</b>) is associated with the main structure, containing <b>ClassificationProperties</b> and a <b>Property</b> (type: <b>PropertyKeyValuePairType</b>) with a multiplicity of 1..∞.</p>
type	DiodeClassificationType, BridgeRectifierClassificationType, CurrentRegulatorClassificationType, ESD-DiodeClassificationType, LED-DiodeClassificationType, MicrowaveDiodeClassificationType, PIN-DiodeClassificationType, RectifierDiodeClassificationType, SchottkyDiodeClassificationType, SignalDiodeClassificationType, SiliconCarbideDiodeClassificationType, TunnelDiodeClassificationType, Uni-tunnelDiodeClassificationType, VaractorDiodeClassificationType, VoltageRegulatorClassificationType, ZenerDiodeClassificationType, OtherDiodeClassificationType, DiodeClassificationProperty-ArrayType, PropertyKeyValuePairType.

#### 4.5.1.3.7.1. LED Classification



#### 4.5.1.3.7.2. Rectifier Diode Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/Rectifier</a>
diagram	<p>The diagram illustrates the structure of the Rectifier Diode Classification. It shows a 'Rectifier' entity (type: RectifierDiodeClassificationType) connected to a 'Property-Array' entity (type: RectifierDiodeClassificationProperty-ArrayType). The 'Property-Array' entity is further detailed as containing an 'attributes' container with 'ClassificationProperties', an 'Array' of 'EmptyType', and a 'Property' of 'PropertyKeyValuePairType' with a cardinality of 0..∞.</p>
type	<a href="#">RectifierDiodeClassificationType</a> , <a href="#">RectifierDiodeClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValuePairType</a> .

#### 4.5.1.3.7.3. Schottky Diode Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/Schottky</a>
diagram	<p>The diagram illustrates the structure of the Schottky Diode Classification. It shows a 'Schottky' entity (type: SchottkyDiodeClassificationType) connected to a 'Property-Array' entity (type: SchottkyDiodeClassificationProperty-ArrayType). The 'Property-Array' entity is further detailed as containing an 'attributes' container with 'ClassificationProperties', an 'Array' of 'EmptyType', and a 'Property' of 'PropertyKeyValuePairType' with a cardinality of 0..∞.</p>
type	<a href="#">SchottkyDiodeClassificationType</a> , <a href="#">SchottkyDiodeClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValuePairType</a> .

4.5.1.3.7.4. Silicon Carbide Diode Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/SiliconCarbide
diagram	
type	SiliconCarbideDiodeClassificationType, SiliconCarbideDiodeClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValuePairType.

4.5.1.3.7.5. ESD Diode Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/ESD
diagram	
type	ESD-DiodeClassificationType, ESD-DiodeClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValuePairType.



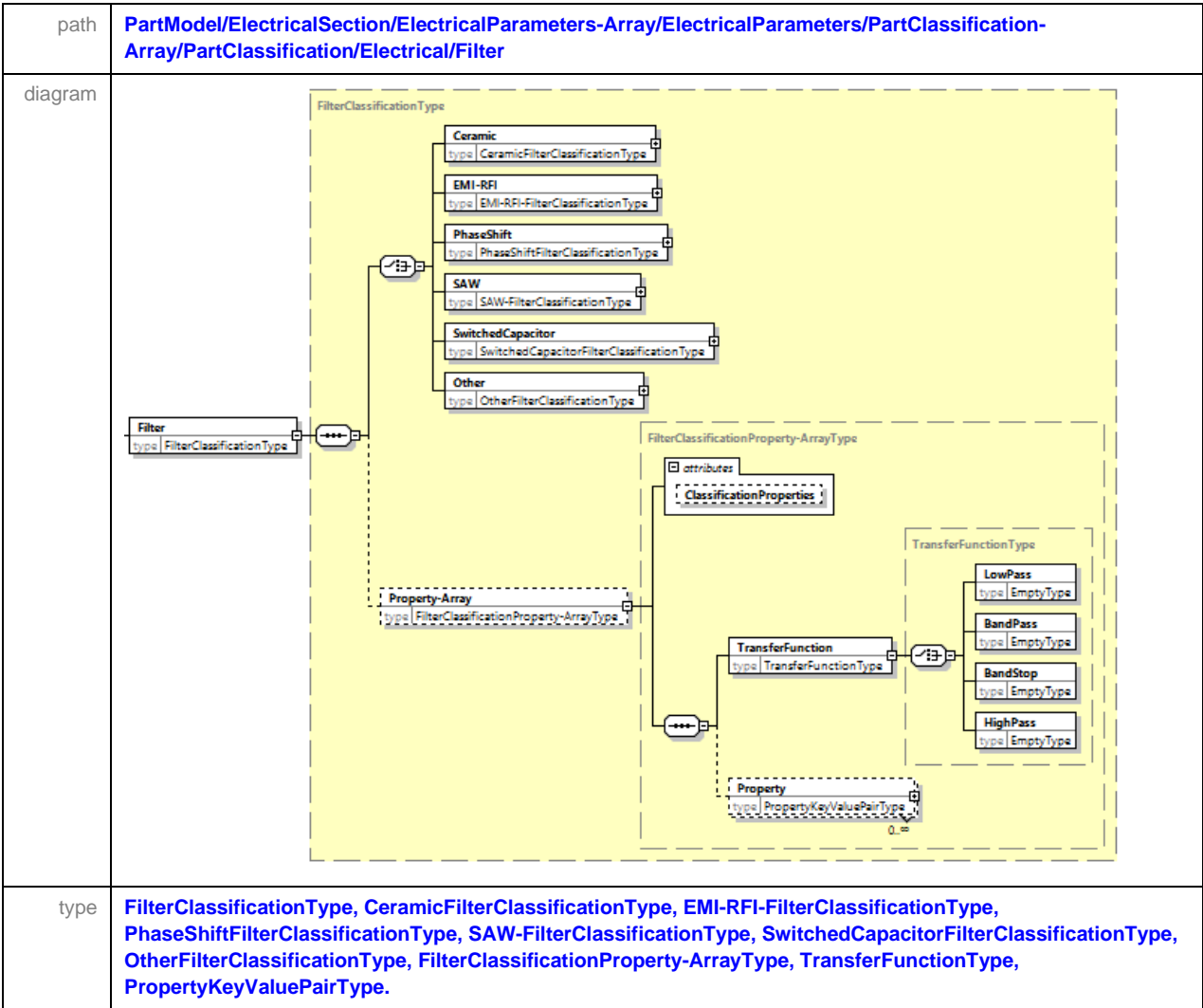
#### 4.5.1.3.7.6. Signal Diode Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/Signal</b>
diagram	
type	<b>SignalDiodeClassificationType, SignalDiodeClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValueType.</b>

#### 4.5.1.3.7.7. Other Diode Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Diode/Other</b>
diagram	
type	<b>OtherDiodeClassificationType, OtherDiodeClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValueType.</b>

4.5.1.3.8. Filter Classification



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**.

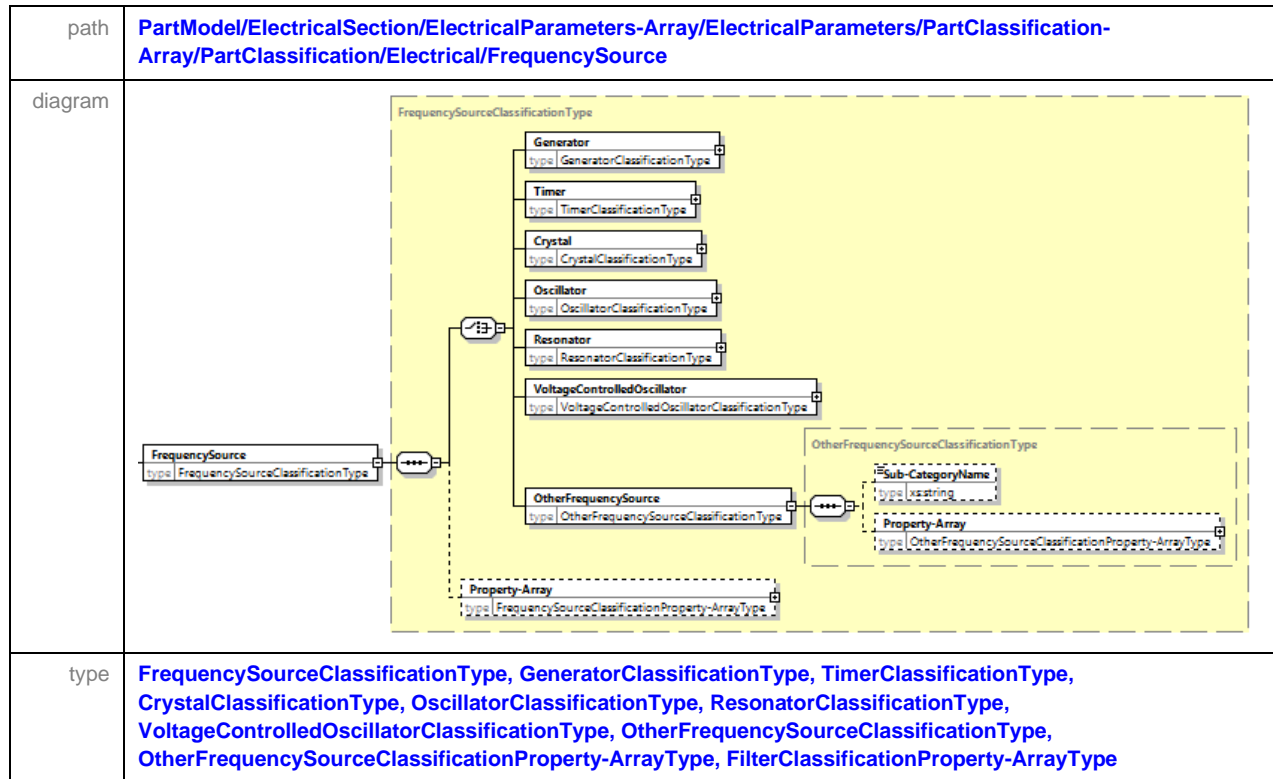
#### 4.5.1.3.8.1. EMI-RFI Filter Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Filter/EMI-RFI</a>
diagram	
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, OtherEMI-RFI-FilterClassificationType, OtherEMI-RFI-FilterClassificationProperty-ArrayType, EMI-RFI-FilterClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValuePairType.

#### 4.5.1.3.8.2. Other Filter Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Filter/Other</a>
diagram	
type	OtherFilterClassificationType, OtherFilterClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValuePairType.

#### 4.5.1.3.9. Frequency Source Classification



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**.

#### 4.5.1.3.10. IC Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/IC</b>
diagram	
type	<b>IC-ClassificationType, CPLD-IC-ClassificationType, DataAcquisitionIC-ClassificationType, DigitalSignalProcessingIC-ClassificationType, FPGA-IC-ClassificationType, InterfaceIC-ClassificationType, LogicIC-ClassificationType, MCM-IC-ClassificationType, MicrocontrollerIC-ClassificationType, PowerManagementIC-ClassificationType, ProcessorIC-ClassificationType, TimingIC-ClassificationType, OtherIC-ClassificationType, OtherIC-ClassificationProperty-ArrayType, IC-ClassificationProperty-ArrayType.</b>

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*.

4.5.1.3.11. Inductor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor
diagram	<p>The diagram illustrates the classification of inductors. A base class, <b>InductorClassificationType</b>, is shown in a yellow-shaded box. It has four subclasses: <b>Ferrite</b>, <b>MetalAlloys</b>, <b>Air</b>, and <b>Other</b>. Each subclass has a 'type' attribute pointing to a specific classification type (e.g., <b>FerriteInductorClassificationType</b>). Additionally, <b>InductorClassificationType</b> has an optional association (indicated by a dashed line and a circle with a plus sign) to a <b>Property-Array</b> class, which has a 'type' attribute pointing to <b>InductorClassificationProperty-ArrayType</b>. To the left of the main diagram, an <b>Inductor</b> class is shown with a 'type' attribute pointing to <b>InductorClassificationType</b>. A solid line connects the <b>Inductor</b> class to the <b>InductorClassificationType</b> class, indicating a generalization or association.</p>
type	InductorClassificationType, FerriteInductorClassificationType, MetalAlloysInductorClassificationType, AirInductorClassificationType, OtherInductorClassificationType, InductorClassificationProperty-ArrayType

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.

#### 4.5.1.3.11.1. Ferrite Inductor Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Ferrite</a>
diagram	
type	<a href="#">FerriteInductorClassificationType</a> , <a href="#">FixedFerriteInductorClassificationType</a> , <a href="#">VariableFerriteInductorClassificationType</a> , <a href="#">FerriteInductorClassificationProperty-ArrayType</a>

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.

##### 4.5.1.3.11.1.1. Variable Ferrite Inductor Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Ferrite/Variable/Property-Array</a>
diagram	
type	<a href="#">VariableFerriteInductorClassificationType</a> , <a href="#">VariableFerriteInductorClassificationProperty-ArrayType</a> , <a href="#">VariableInductorAdjustmentType</a> , <a href="#">PropertyKeyValuePairType</a>

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

#### 4.5.1.3.11.1.2. Ferrite Inductor Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Ferrite/Property-Array
diagram	<p>The diagram illustrates the structure of the <b>FerriteInductorClassificationProperty-ArrayType</b>. It is a class with an <b>attributes</b> compartment containing a <b>ClassificationProperties</b> compartment. A <b>Property-Array</b> compartment is associated with the main class, with a type of <b>FerriteInductorClassificationProperty-ArrayType</b>. A <b>Material</b> compartment is associated with the main class, with a type of <b>FerriteInductorMaterialPropertyType</b>. A <b>Property</b> compartment is associated with the main class, with a type of <b>PropertyKeyValuePairType</b> and a cardinality of <b>1..∞</b>. The <b>FerriteInductorMaterialPropertyType</b> class is shown as a dashed box containing a list of materials: <b>Air</b>, <b>IronPowder</b>, <b>NiFeMoPowder</b>, <b>NiFePowder</b>, <b>FeSiAlPowder</b>, <b>FeSiPowder</b>, <b>ManganeseZincFerrite</b>, <b>NickelZincFerrite</b>, and <b>Other</b>. Each material has a type of <b>EmptyType</b>, except for <b>Other</b> which has a type of <b>xs:string</b>.</p>
type	FerriteInductorClassificationProperty-ArrayType, FerriteInductorMaterialPropertyType, JEP30-D10:EmptyType, PropertyKeyValuePairType

**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.

#### 4.5.1.3.11.2. MetalAlloys Inductor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/MetalAlloys
diagram	<pre> classDiagram     class MetalAlloysInductorClassificationType {         +MetalAlloys MetalAlloysInductorClassificationType         +Fixed MetalAlloysInductorClassificationType         +Variable MetalAlloysInductorClassificationType         +Property-Array MetalAlloysInductorClassificationProperty-ArrayType     }     class MetalAlloys {         +MetalAlloysInductorClassificationType     }     class Fixed {         +MetalAlloysInductorClassificationType     }     class Variable {         +MetalAlloysInductorClassificationType     }     class Property-Array {         +MetalAlloysInductorClassificationProperty-ArrayType     }     MetalAlloysInductorClassificationType "1" -- "*" MetalAlloys : MetalAlloysInductorClassificationType     MetalAlloysInductorClassificationType "1" -- "*" Fixed : MetalAlloysInductorClassificationType     MetalAlloysInductorClassificationType "1" -- "*" Variable : MetalAlloysInductorClassificationType     MetalAlloysInductorClassificationType "1" -- "*" Property-Array : MetalAlloysInductorClassificationProperty-ArrayType   </pre>
type	MetalAlloysInductorClassificationType, FixedMetalAlloysInductorClassificationType, VariableMetalAlloysInductorClassificationType, MetalAlloysInductorClassificationProperty-ArrayType.



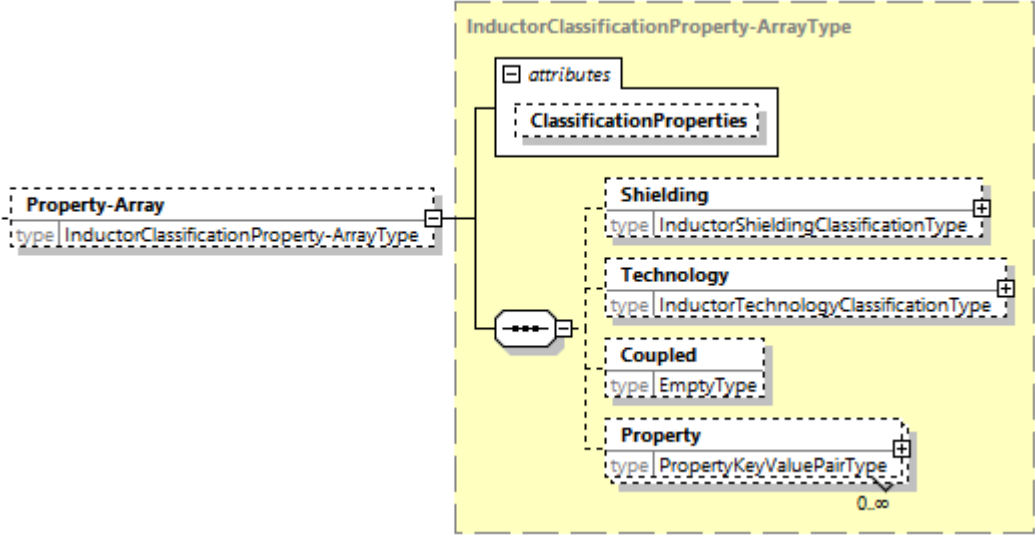
#### 4.5.1.3.11.2.1. MetalAlloys Inductor Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/MetalAlloys/Variable</b>
diagram	
type	<b>VariableMetalAlloysInductorClassificationType, VariableMetalAlloysInductorClassificationProperty-ArrayType, VariableInductorAdjustmentType, PropertyKeyValuePairType</b>

#### 4.5.1.3.11.3. Other Inductor Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Other</b>
diagram	
type	<b>OtherInductorClassificationType, OtherInductorClassificationProperty-ArrayType, PropertyKeyValuePairType</b>

4.5.1.3.11.4. Inductor Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Property-Array
diagram	
type	InductorClassificationProperty-ArrayType, InductorShieldingClassificationType, InductorTechnologyClassificationType, JEP30-D10:EmptyType, PropertyKeyValuePairType

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.

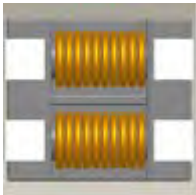
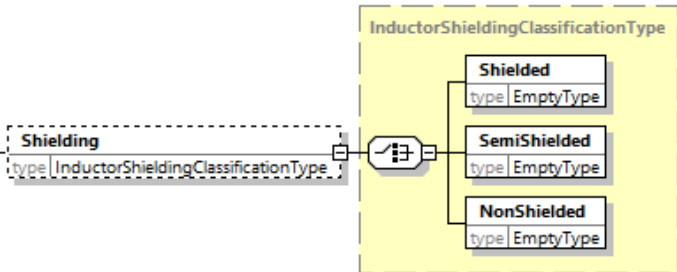
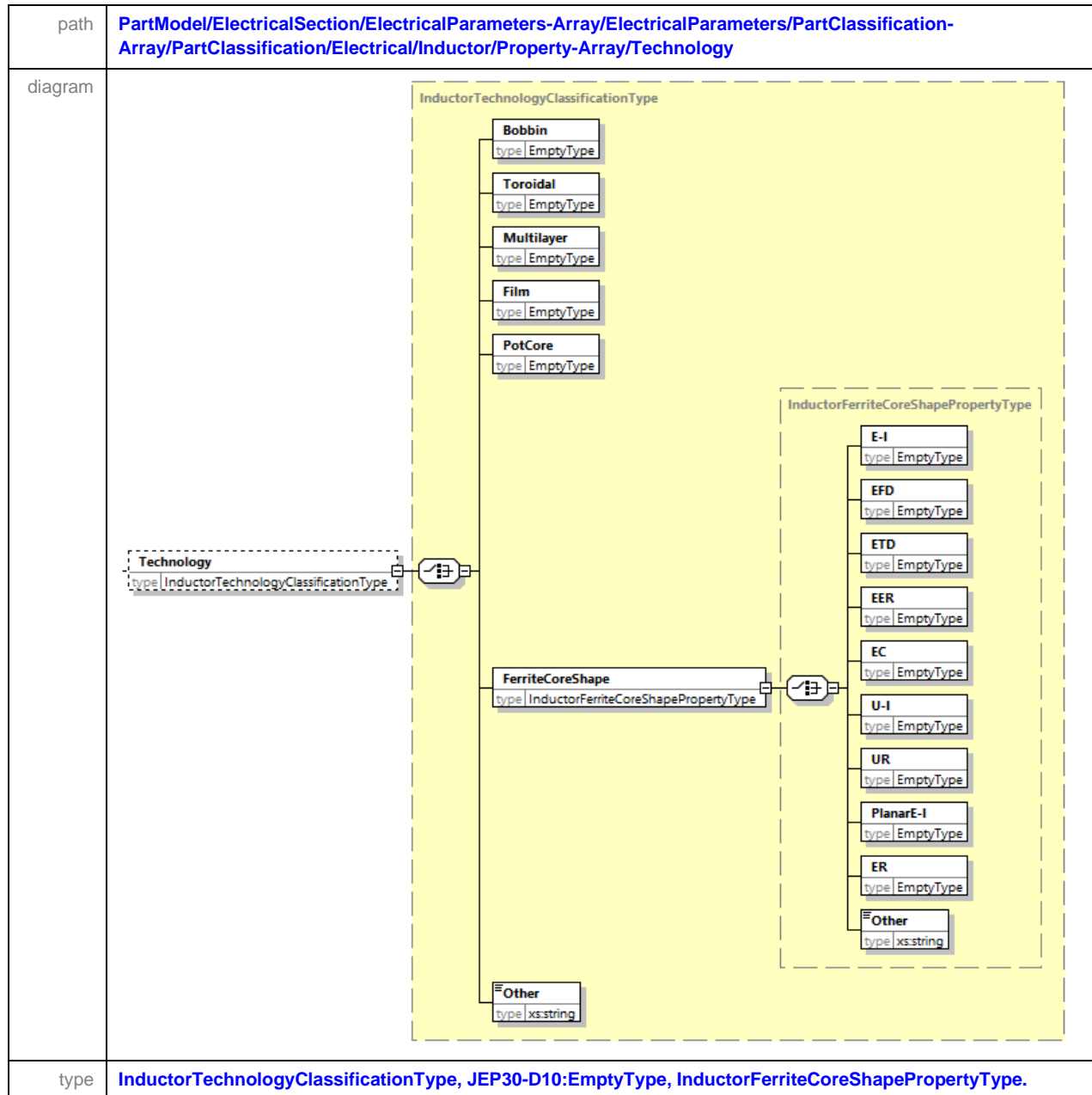


Figure 3 — Coupled Inductor

4.5.1.3.11.4.1. Inductor Shielding Property

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Inductor/Property-Array/Shielding
diagram	
type	InductorShieldingClassificationType, JEP30-D10:EmptyType.

## 4.5.1.3.11.4.2. Inductor Technology Property



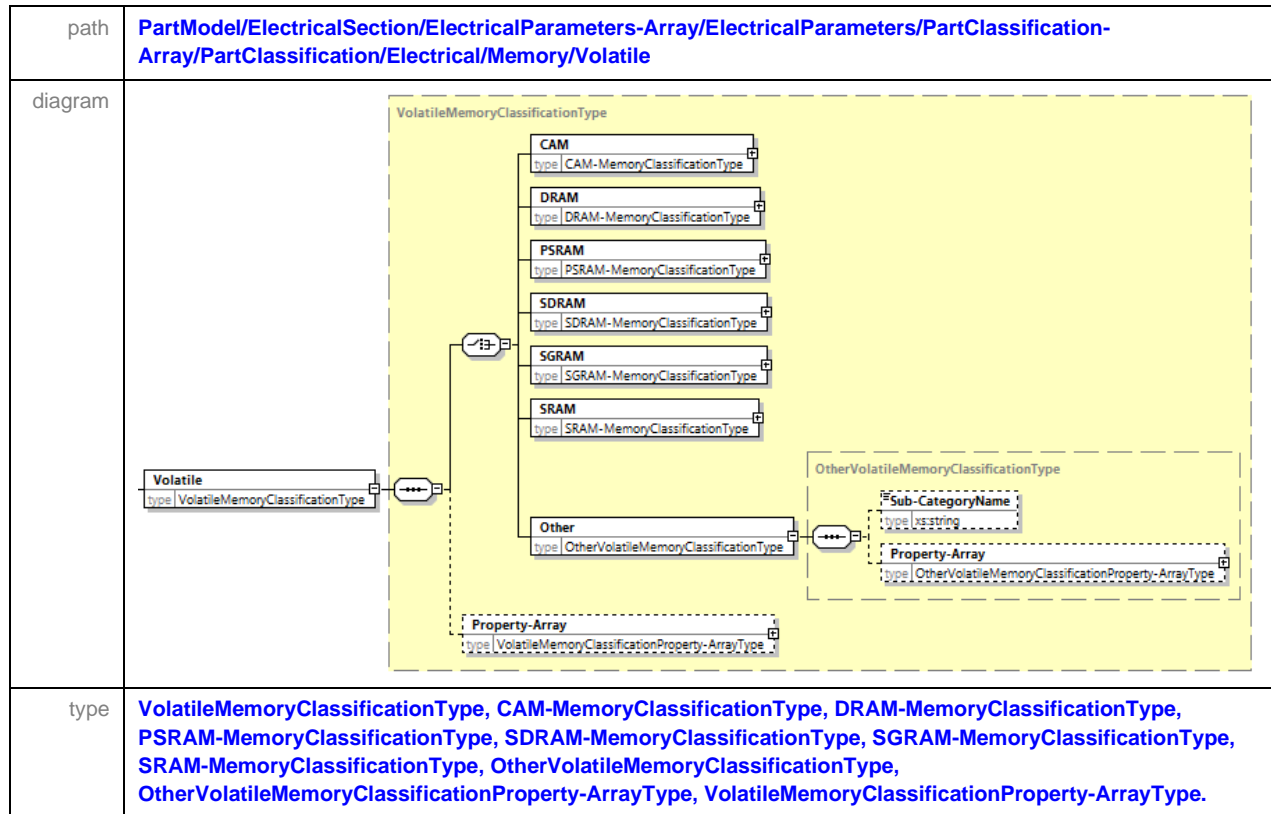
All Inductors whether fixed or variable can have different *Technology* construction as outlined above, but other technologies can be captured under the *Other* property elements.

**4.5.1.3.12. Memory Classification**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Memory</a>
diagram	<pre> classDiagram     class Memory {         type MemoryClassificationType     }     class MemoryClassificationType {         +Volatile VolatileMemoryClassificationType         +NonVolatile NonVolatileMemoryClassificationType         +Property-Array MemoryClassificationProperty-ArrayType     }     Memory --&gt; MemoryClassificationType     MemoryClassificationType --&gt; Volatile     MemoryClassificationType --&gt; NonVolatile     MemoryClassificationType --&gt; Property-Array   </pre>
type	<a href="#">MemoryClassificationType</a> , <a href="#">VolatileMemoryClassificationType</a> , <a href="#">NonVolatileMemoryClassificationType</a> .

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.

#### 4.5.1.3.12.1. Volatile Memory Classification



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

4.5.1.3.12.2. NonVolatile Memory Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Memory/NonVolatile
diagram	<p>The diagram illustrates the NonVolatileMemoryClassificationType hierarchy. It includes subclasses for various memory types: CBRAM, EPROM, EEPROM, FLASH, FRAM, MASKROM, MRAM, NVSRAM, PCM, PROM, and ReRam. Each subclass is associated with a specific MemoryClassificationType. An 'Other' subclass is also present, which is associated with OtherNonVolatileMemoryClassificationType. A 'Property-Array' is associated with the base class and the 'Other' subclass. The 'Other' subclass has a 'Sub-CategoryName' attribute and a 'Property-Array' attribute pointing to OtherNonVolatileMemoryClassificationProperty-ArrayType.</p>
type	NonVolatileMemoryClassificationType, CBRAM-MemoryClassificationType, EPROM-MemoryClassificationType, EEPROM-MemoryClassificationType, FLASH-MemoryClassificationType, FRAM-MemoryClassificationType, MASKROM-MemoryClassificationType, MRAM-MemoryClassificationType, NVSRAM-MemoryClassificationType, PCM-MemoryClassificationType, PROM-MemoryClassificationType, ReRam-MemoryClassificationType, OtherNonVolatileMemoryClassificationType, OtherNonVolatileMemoryClassificationProperty-ArrayType, NonVolatileMemoryClassificationProperty-ArrayType

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*.

#### 4.5.1.3.13. Optoelectronics Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic
diagram	<pre> classDiagram     class Optoelectronic {         type OptoelectronicClassificationType     }     class Display {         type DisplayClassificationType     }     class Photoemitter {         type PhotoemitterClassificationType     }     class Photosensitive {         type PhotosensitiveClassificationType     }     class Optocoupler {         type OptocouplerClassificationType     }     class Other {         type OtherOptoelectronicClassificationType     }     class PropertyArray {         type OptoelectronicClassificationProperty-ArrayType     }     Optoelectronic "1" -- "*" Display     Optoelectronic "1" -- "*" Photoemitter     Optoelectronic "1" -- "*" Photosensitive     Optoelectronic "1" -- "*" Optocoupler     Optoelectronic "1" -- "*" Other     Optoelectronic "1" -- "*" PropertyArray     </pre> <p>The diagram illustrates the structure of the <b>OptoelectronicClassificationType</b>. It is a base class for several subclasses: <b>Display</b>, <b>Photoemitter</b>, <b>Photosensitive</b>, <b>Optocoupler</b>, and <b>Other</b>. Each subclass has its own <b>type</b> property. The <b>Optoelectronic</b> class also has a <b>Property-Array</b> association, which is a collection of <b>OptoelectronicClassificationProperty-ArrayType</b> objects.</p>
type	OptoelectronicsClassificationType, DisplayClassification, PhotoemitterClassificationType, PhotosensitiveClassificationType, OptocouplerClassificationType.

**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.

#### 4.5.1.3.13.1. Display Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Display</a>
diagram	
type	<a href="#">DisplayClassificationType</a> , <a href="#">TFT-DisplayClassificationType</a> , <a href="#">LED-DisplayClassificationType</a> , <a href="#">OrganicDisplayClassificationType</a> , <a href="#">OtherDisplayClassificationType</a> , <a href="#">DisplayClassificationProperty-ArrayType</a> .

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 [LED](#) and the [Organic](#) displays can be provided in [Array](#) form as shown below.

##### 4.5.1.3.13.1.1. LED Display Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Display/LED</a>
diagram	
type	<a href="#">LED-DisplayClassificationType</a> , <a href="#">LED-DisplayClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValuePairType</a> .



#### 4.5.1.3.13.1.2. Organic Display Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Display/Organic</a>
diagram	
type	<a href="#">OrganicDisplayClassificationType</a> , <a href="#">OrganicDisplayClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValuePairType</a>

#### 4.5.1.3.13.1.3. Other Display Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Display/Other</a>
diagram	
type	<a href="#">OtherDisplayClassificationType</a> , <a href="#">OtherDisplayClassificationProperty-ArrayType</a>

#### 4.5.1.3.13.2. Photoemitter Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photoemitter</a>
diagram	
type	<a href="#">PhotoemitterClassificationType</a> , <a href="#">InfraredEmittingDiodePhotoemitterClassificationType</a> , <a href="#">LED-PhotoemitterClassificationType</a> , <a href="#">LaserPhotoemitterClassificationType</a> , <a href="#">OtherPhotoemitterClassificationType</a> , <a href="#">PhotoemitterClassificationProperty-ArrayType</a> .

*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*.

##### 4.5.1.3.13.2.1. Photoemitter LED Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photoemitter/LED</a>
diagram	
type	<a href="#">LED-PhotoemitterClassificationType</a> , <a href="#">LED-PhotoemitterClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValuePairType</a>

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

#### 4.5.1.3.13.2.2. Other Display Classification

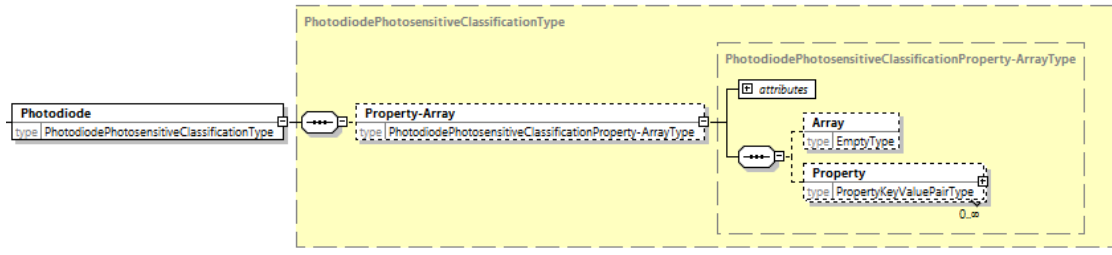
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photoemitter/Other</a>
diagram	
type	<a href="#">OtherPhotoemitterClassificationType</a> , <a href="#">OtherPhotoemitterClassificationProperty-ArrayType</a>

#### 4.5.1.3.13.3. Photosensitive Classification


path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photosensitive</a>
diagram	
type	<a href="#">PhotosensitiveClassificationType</a> , <a href="#">PhotodiodePhotosensitiveClassificationType</a> , <a href="#">PhotothyristorPhotosensitiveClassificationType</a> , <a href="#">PhototriacPhotosensitiveClassificationType</a> , <a href="#">PhototransistorPhotosensitiveClassificationType</a> , <a href="#">PhotodarlingtonPhotosensitiveClassificationType</a> , <a href="#">PhotovoltaicDiodePhotosensitiveClassificationType</a> , <a href="#">OtherPhotosensitiveClassificationType</a> , <a href="#">PhotosensitiveClassificationProperty-ArrayType</a>

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.

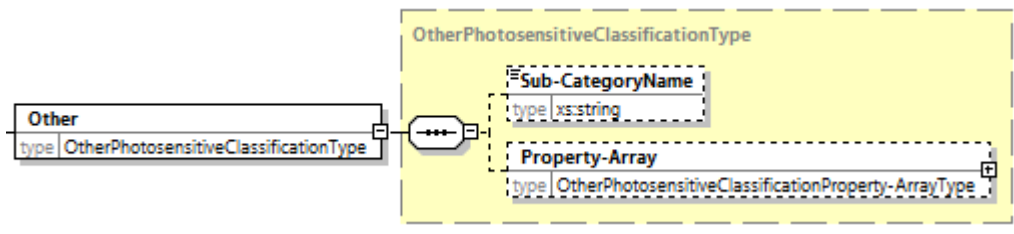
#### 4.5.1.3.13.3.1. Photodiode Photosensitive Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photosensitive/Photodiode</a>
diagram	 <p>The diagram illustrates the structure of the Photodiode Photosensitive Classification. It shows a <b>Photodiode</b> element with a type of <b>PhotodiodePhotosensitiveClassificationType</b>. This type is associated with a <b>Property-Array</b> (type: <b>PhotodiodePhotosensitiveClassificationProperty-ArrayType</b>). The <b>Property-Array</b> is further detailed with an <b>attributes</b> section containing an <b>Array</b> (type: <b>EmptyType</b>) and a <b>Property</b> (type: <b>PropertyKeyValuePairType</b>).</p>
type	<b>PhotodiodePhotosensitiveClassificationType</b> , <b>PhotodiodePhotosensitiveClassificationProperty-ArrayType</b> , <b>JEP30-D10:EmptyType</b> , <b>PropertyKeyValuePairType</b>

#### 4.5.1.3.13.3.2. Phototransistor Photosensitive Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photosensitive/Phototransistor</a>
diagram	 <p>The diagram illustrates the structure of the Phototransistor Photosensitive Classification. It shows a <b>Phototransistor</b> element with a type of <b>PhototransistorPhotosensitiveClassificationType</b>. This type is associated with a <b>Property-Array</b> (type: <b>PhototransistorPhotosensitiveClassificationProperty-ArrayType</b>). The <b>Property-Array</b> is further detailed with an <b>attributes</b> section containing an <b>Array</b> (type: <b>EmptyType</b>) and a <b>Property</b> (type: <b>PropertyKeyValuePairType</b>).</p>
type	<b>PhototransistorPhotosensitiveClassificationType</b> , <b>PhototransistorPhotosensitiveClassificationProperty-ArrayType</b> , <b>JEP30-D10:EmptyType</b> , <b>PropertyKeyValuePairType</b>

#### 4.5.1.3.13.3.3. Other Photosensitive Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Photosensitive/Other</a>
diagram	 <p>The diagram illustrates the structure of the Other Photosensitive Classification. It shows an <b>Other</b> element with a type of <b>OtherPhotosensitiveClassificationType</b>. This type is associated with a <b>Sub-CategoryName</b> (type: <b>xs:string</b>) and a <b>Property-Array</b> (type: <b>OtherPhotosensitiveClassificationProperty-ArrayType</b>).</p>
type	<b>OtherPhotosensitiveClassificationType</b> , <b>OtherPhotosensitiveClassificationProperty-ArrayType</b> .

#### 4.5.1.3.13.4. Optocoupler Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Optocoupler</a>
diagram	
type	<a href="#">OptocouplerClassificationType</a> , <a href="#">PhotodiodeOptocouplerClassificationType</a> , <a href="#">PhotothyristorOptocouplerClassificationType</a> , <a href="#">PhototransistorOptocouplerClassificationType</a> , <a href="#">PhotodarlingtonOptocouplerClassificationType</a> , <a href="#">OtherOptocouplerClassificationType</a> , <a href="#">OptocouplerClassificationProperty-ArrayType</a> .

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 below.

##### 4.5.1.3.13.4.1. Photodiode Optocoupler Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Optocoupler/Photodiode</a>
diagram	
type	<a href="#">PhotodiodeOptocouplerClassificationType</a> , <a href="#">PhotodiodeOptocouplerClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValueType</a>

#### 4.5.1.3.13.4.2. Phototransistor Optocoupler Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Optocoupler/ Phototransistor, JEP30-D10:EmptyType, PropertyKeyValuePairType</a>
diagram	
type	<a href="#">PhototransistorOptocouplerClassificationType, PhototransistorOptocouplerClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValuePairType</a>

#### 4.5.1.3.13.4.3. Other Optocoupler Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Optoelectronic/Optocoupler/Other</a>
diagram	
type	<a href="#">OtherOptocouplerClassificationType, OtherOptocouplerClassificationProperty-ArrayType</a>

#### 4.5.1.3.14. Regulator Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator</a>
diagram	
type	<a href="#">PowerRegulatorClassificationType</a> , <a href="#">LinearCurrentRegulatorClassificationType</a> , <a href="#">LinearVoltageRegulatorClassificationType</a> , <a href="#">SwitchingPowerRegulatorClassificationType</a> , <a href="#">SCR-PowerRegulatorClassificationType</a> , <a href="#">OtherPowerRegulatorClassificationType</a> , <a href="#">PowerRegulatorClassificationProperty-ArrayType</a>

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](#).

##### 4.5.1.3.14.1. Linear Current Regulator Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/LinearCurrent</a>
diagram	
type	<a href="#">LinearCurrentRegulatorClassificationType</a> , <a href="#">FixedLinearCurrentRegulatorClassificationType</a> , <a href="#">VariableLinearCurrentRegulatorClassificationType</a> , <a href="#">LinearCurrentRegulatorClassificationProperty-ArrayType</a>

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.

#### 4.5.1.3.14.2. Linear Current Regulator Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/LinearCurrent/Property-Array</a>
diagram	
type	<a href="#">LinearCurrentRegulatorClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">LinearCurrentRegulatorProtectionType</a> , <a href="#">PropertyKeyValuePairType</a>

Linear Current regulators come with a *LowDropout* option, and have *Protection* against *OverTemperature*, *OverVoltage*, and or *UnderVoltage*.

#### 4.5.1.3.14.3. Linear Voltage Regulator Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/LinearVoltage</a>
diagram	
type	<a href="#">LinearVoltageRegulatorClassificationType</a> , <a href="#">FixedLinearVoltageRegulatorClassificationType</a> , <a href="#">VariableLinearVoltageRegulatorClassificationType</a> , <a href="#">LinearVoltageRegulatorClassificationProperty-ArrayType</a>

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.



#### 4.5.1.3.14.4. Linear Voltage Regulator Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/LinearVoltage/Property-Array</a>
diagram	
type	<a href="#">LinearVoltageRegulatorClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">LinearVoltageRegulatorProtectionType</a> , <a href="#">PropertyKeyValuePairType</a>

Linear Voltage regulators come with a [LowDropout](#) option, and have [Protection](#) against [ShortCircuit](#), [OverCurrent](#), and or [OverTemperature](#).

#### 4.5.1.3.14.5. Switching Regulator Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Switching</a>
diagram	
type	<a href="#">SwitchingPowerRegulatorClassificationType</a> , <a href="#">IsolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">Non-isolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">SwitchingPowerRegulatorClassificationProperty-ArrayType</a>

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.

#### 4.5.1.3.14.5.1. Isolated Switching Regulator Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Switching/Isolated</a>
diagram	
type	<a href="#">IsolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">FlybackIsolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">ForwardIsolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">Push-PullIsolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">OtherIsolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">IsolatedSwitchingPowerRegulatorClassificationProperty-ArrayType</a>

#### 4.5.1.3.14.5.2. Other IsolatedSwitching Regulator Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Switching/Isolated/Other</a>
diagram	
type	<a href="#">OtherIsolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">OtherIsolatedSwitchingPowerRegulatorClassificationProperty-ArrayType</a> .

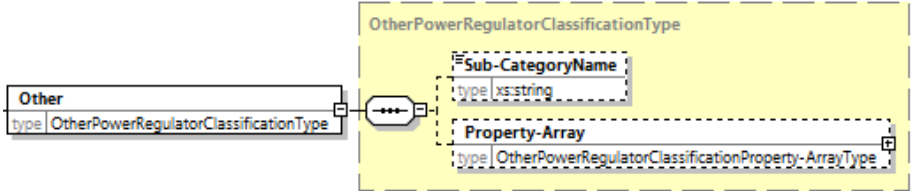
#### 4.5.1.3.14.5.3. Non-isolated Switching Regulator Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Switching/Non-isolated</a>
diagram	
type	<a href="#">Non-isolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">BoostNon-isolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">BuckNon-isolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">Buck-BoostNon-isolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">ChargePumpNon-isolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">OtherNon-isolatedSwitchingPowerRegulatorClassificationType</a> , <a href="#">Non-isolatedSwitchingPowerRegulatorClassificationProperty-ArrayType</a>

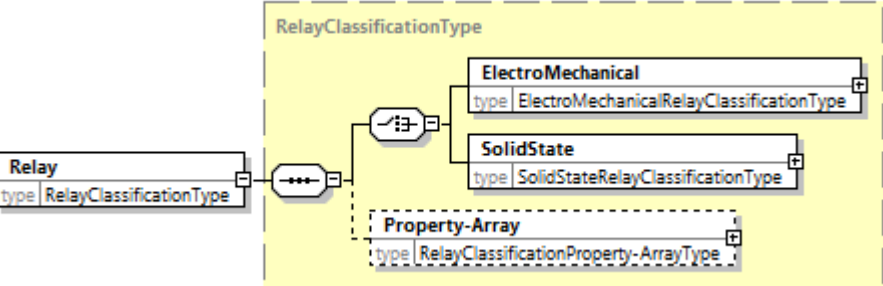
#### 4.5.1.3.14.5.4. Other Non-IsolatedSwitching Regulator Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Switching/Non-isolated/Other</a>
diagram	
type	<a href="#">IsolatedSwitchingPowerRegulatorClassificationProperty-ArrayType</a> ,

**4.5.1.3.14.6. Other Regulator Classification**

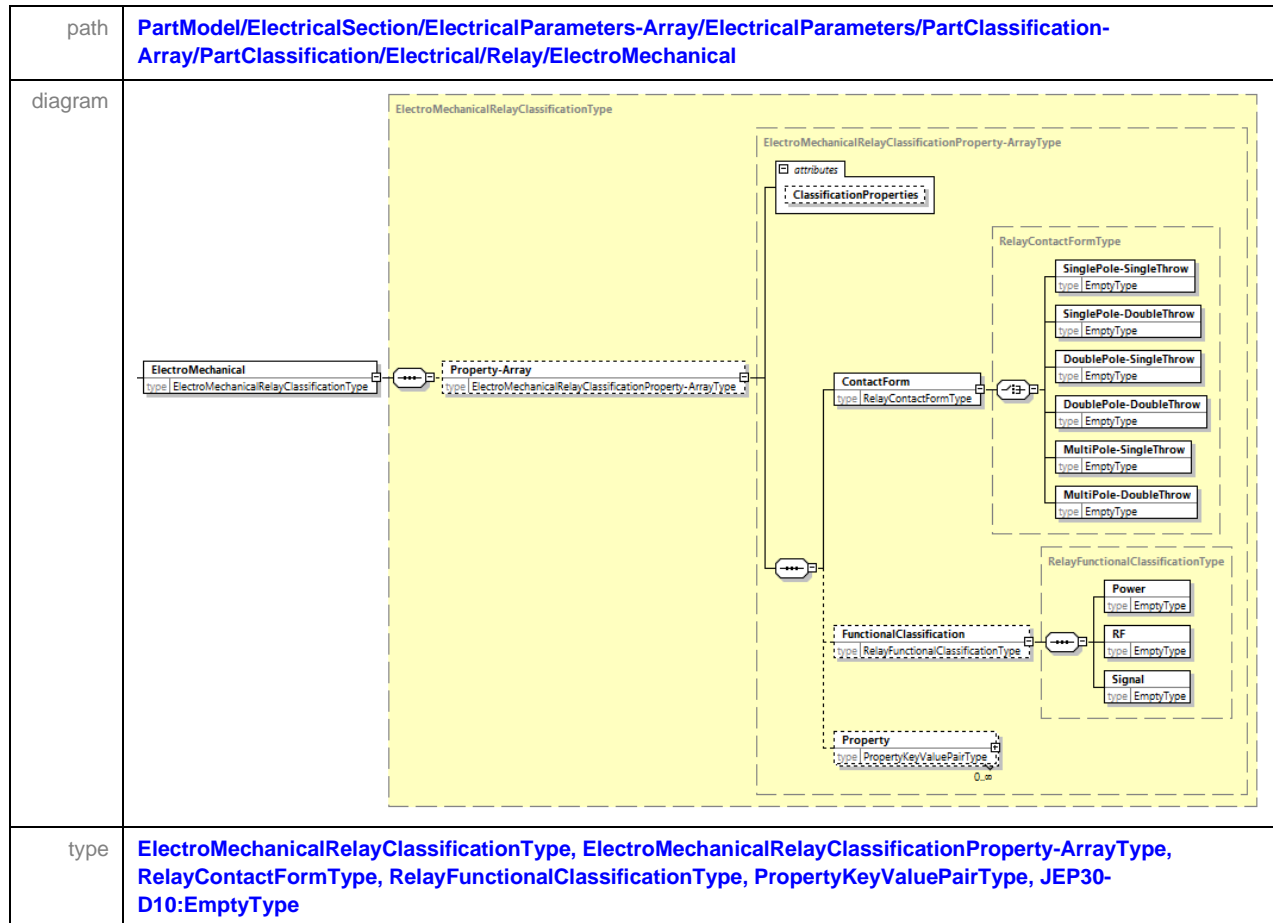
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Regulator/Other</a>
diagram	 <p>The diagram shows a class <b>Other</b> with a type <code>OtherPowerRegulatorClassificationType</code>. It is connected to a dashed box representing the <code>OtherPowerRegulatorClassificationType</code> structure. Inside this box, there is a <code>Sub-CategoryName</code> property of type <code>xs:string</code> and a <code>Property-Array</code> of type <code>OtherPowerRegulatorClassificationProperty-ArrayType</code>.</p>
type	<a href="#">OtherPowerRegulatorClassificationType</a> , <a href="#">OtherPowerRegulatorClassificationProperty-ArrayType</a>

**4.5.1.3.15. Relay Classification**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Relay</a>
diagram	 <p>The diagram shows a class <b>Relay</b> with a type <code>RelayClassificationType</code>. It is connected to a dashed box representing the <code>RelayClassificationType</code> structure. Inside this box, there is a choice between <code>ElectroMechanical</code> (type <code>ElectroMechanicalRelayClassificationType</code>) and <code>SolidState</code> (type <code>SolidStateRelayClassificationType</code>), and a <code>Property-Array</code> of type <code>RelayClassificationProperty-ArrayType</code>.</p>
type	<a href="#">RelayClassificationType</a> , <a href="#">ElectroMechanicalRelayClassificationType</a> , <a href="#">SolidStateRelayClassificationType</a> , <a href="#">RelayClassificationProperty-ArrayType</a>

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.

## 4.5.1.3.15.1. ElectroMechanical Relay Classification and Property-Array



#### 4.5.1.3.15.2. SolidState Relay Classification and Property-Array

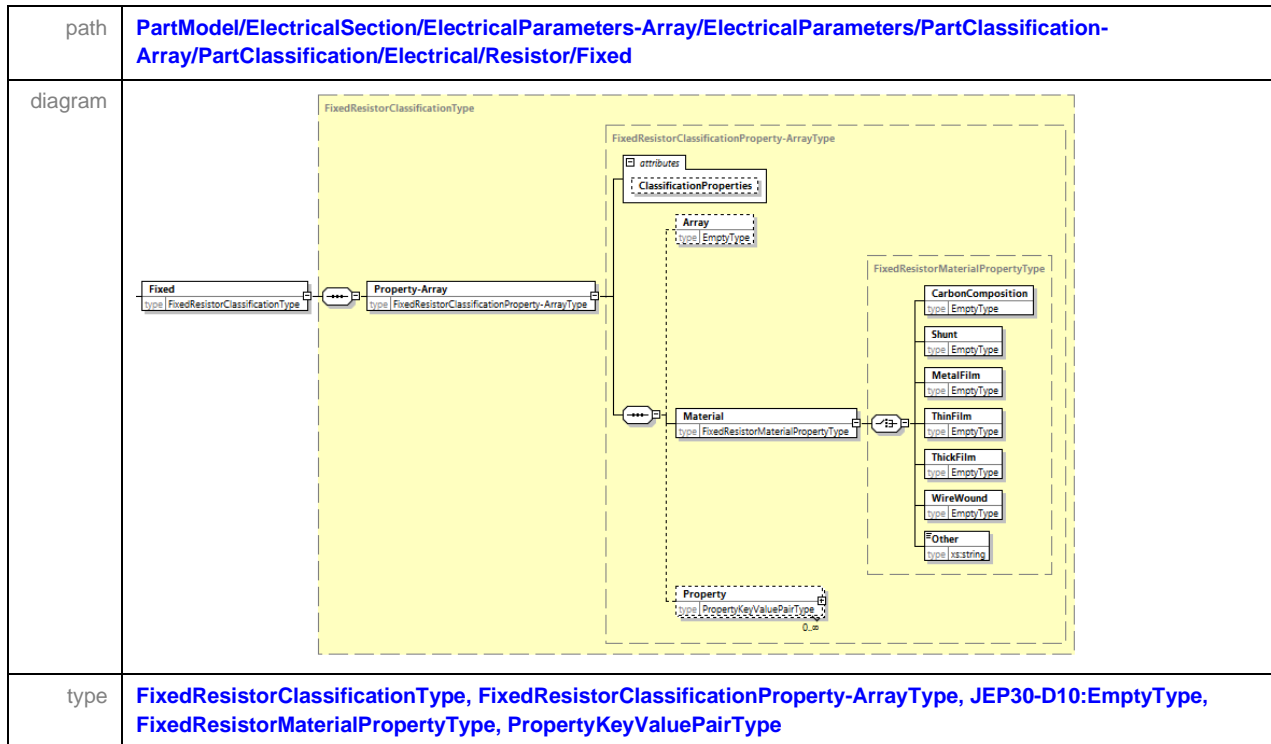
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Relay/SolidState</a>
diagram	
type	<a href="#">SolidStateRelayClassificationType</a> , <a href="#">SolidStateRelayClassificationProperty-ArrayType</a> , <a href="#">RelayContactFormType</a> , <a href="#">RelayFunctionalClassificationType</a> , <a href="#">PropertyKeyValueType</a> , <a href="#">JEP30-D10:EmptyType</a>

#### 4.5.1.3.16. Resistor Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor</a>
diagram	
type	<a href="#">ResistorClassificationType</a> , <a href="#">FixedResistorClassificationType</a> , <a href="#">AdjustableResistorClassificationType</a> , <a href="#">NonLinearResistorClassificationType</a> , <a href="#">ResistorClassificationProperty-ArrayType</a> .

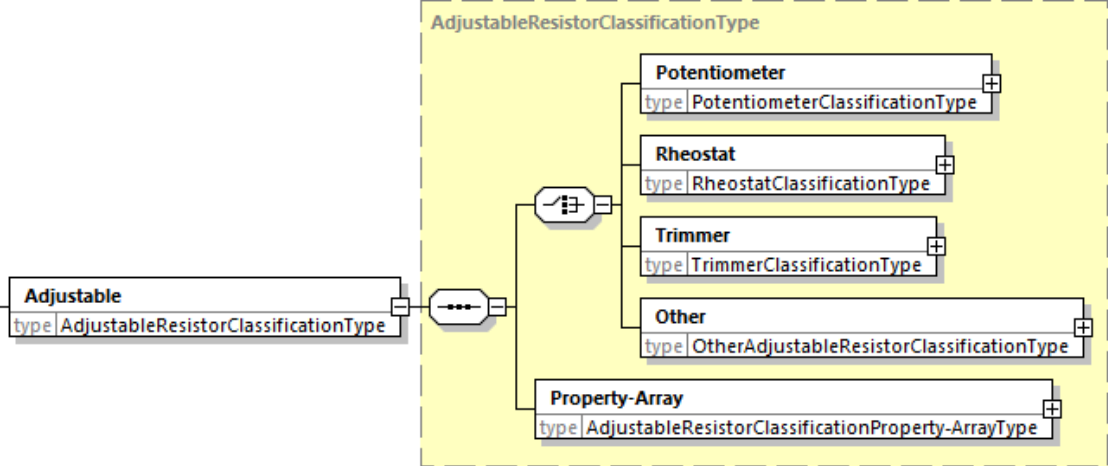
A [Resistor](#) is a passive electrical component that implements electrical resistance as a circuit element. They are used to reduce current flow, adjust signal levels, to divide voltages, bias active elements, and terminate transmission lines, among other uses.

#### 4.5.1.3.16.1. Fixed Resistor Classification and Property-Array



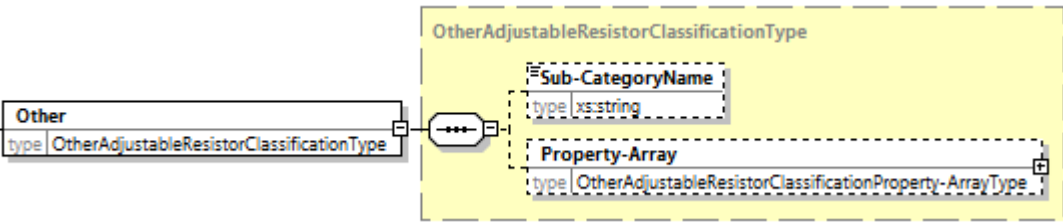
**Fixed** resistors have resistances that only change slightly with temperature, time or operating voltage.

4.5.1.3.16.2. Adjustable Resistor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Adjustable
diagram	 <p>The diagram illustrates the structure of the <b>Adjustable</b> resistor classification. It shows a box labeled <b>Adjustable</b> with the type <code>AdjustableResistorClassificationType</code>. This box is connected via a dashed line to a larger yellow-shaded box labeled <b>AdjustableResistorClassificationType</b>. Inside this yellow box, there are four sub-categories: <b>Potentiometer</b> (type <code>PotentiometerClassificationType</code>), <b>Rheostat</b> (type <code>RheostatClassificationType</code>), <b>Trimmer</b> (type <code>TrimmerClassificationType</code>), and <b>Other</b> (type <code>OtherAdjustableResistorClassificationType</code>). Each sub-category box has a small '+' icon in its top right corner. Below these sub-categories is a <b>Property-Array</b> box with the type <code>AdjustableResistorClassificationProperty-ArrayType</code>, also featuring a '+' icon. A dashed line connects the <b>Adjustable</b> box to the <b>Property-Array</b> box.</p>
type	AdjustableResistorClassificationType, PotentiometerClassificationType, RheostatClassificationType, TrimmerClassificationType, OtherAdjustableResistorClassificationType, AdjustableResistorClassificationProperty-ArrayType.

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

4.5.1.3.16.2.1. Other Adjustable Resistor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Adjustable
diagram	 <p>The diagram illustrates the structure of the <b>Other</b> adjustable resistor classification. It shows a box labeled <b>Other</b> with the type <code>OtherAdjustableResistorClassificationType</code>. This box is connected via a dashed line to a larger yellow-shaded box labeled <b>OtherAdjustableResistorClassificationType</b>. Inside this yellow box, there are two sub-categories: <b>Sub-CategoryName</b> (type <code>xs:string</code>) and <b>Property-Array</b> (type <code>OtherAdjustableResistorClassificationProperty-ArrayType</code>). Both sub-category boxes have a small '+' icon in their top right corner. A dashed line connects the <b>Other</b> box to the <b>Property-Array</b> box.</p>
type	OtherAdjustableResistorClassificationType, OtherAdjustableResistorClassificationProperty-ArrayType



#### 4.5.1.3.16.3. Non-Linear Resistor Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Non-Linear</a>
diagram	<pre> classDiagram     class Non-LinearResistor {         type Non-LinearResistorClassificationType     }     class Non-LinearResistorClassificationType {         Thermistor         Varistor         PhotoResistor         Magnetic         Other         Property-Array     }     class Thermistor {         type ThermistorClassificationType     }     class Varistor {         type VaristorResistorClassificationType     }     class PhotoResistor {         type PhotoResistorClassificationType     }     class Magnetic {         type MagneticResistorClassificationType     }     class Other {         type OtherNon-LinearResistorClassificationType     }     class Property-Array {         type Non-LinearResistorProperty-ArrayType     }      Non-LinearResistor --&gt; Non-LinearResistorClassificationType     Non-LinearResistorClassificationType --&gt; Thermistor     Non-LinearResistorClassificationType --&gt; Varistor     Non-LinearResistorClassificationType --&gt; PhotoResistor     Non-LinearResistorClassificationType --&gt; Magnetic     Non-LinearResistorClassificationType --&gt; Other     Non-LinearResistorClassificationType --&gt; Property-Array </pre>
type	<a href="#">LinearResistorClassificationType</a> , <a href="#">ThermistorClassificationType</a> , <a href="#">VaristorResistorClassificationType</a> , <a href="#">PhotoResistorClassificationType</a> , <a href="#">MagneticResistorClassificationType</a> , <a href="#">OtherNon-LinearResistorClassificationType</a>

4.5.1.3.16.4. Thermistor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Non-Linear/Thermistor
diagram	
type	ThermistorClassificationType, ThermistorClassificationProperty-ArrayType, ThermistorTemperatureCoefficientType, ThermistorMaterialType, PropertyKeyValuePairType, JEP30-D10:EmptyType

4.5.1.3.16.5. Other Adjustable Resistor Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Resistor/Non-Linear/Other
diagram	
type	OtherNon-LinearResistorClassificationType, OtherNonLinearResistorClassificationProperty-ArrayType

## 4.5.1.3.17. RF Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/RF
diagram	<p>The diagram illustrates the RF Classification hierarchy. A central class <b>RF</b> (type <code>RF-ClassificationType</code>) is connected to a dashed box representing the <b>RF-ClassificationType</b> container. Inside this container, a vertical list of subclasses is shown, each with a 'type' attribute pointing to its specific classification type:</p> <ul style="list-style-type: none"> <li><b>Antenna</b> (type <code>RF-AntennaClassificationType</code>)</li> <li><b>Attenuator</b> (type <code>RF-AttenuatorClassificationType</code>)</li> <li><b>Balun</b> (type <code>RF-BalunClassificationType</code>)</li> <li><b>Circulator</b> (type <code>RF-CirculatorClassificationType</code>)</li> <li><b>Combiner</b> (type <code>RF-CombinerClassificationType</code>)</li> <li><b>Coupler</b> (type <code>RF-CouplerClassificationType</code>)</li> <li><b>Detector</b> (type <code>RF-DetectorClassificationType</code>)</li> <li><b>Divider</b> (type <code>RF-DividerClassificationType</code>)</li> <li><b>Isolator</b> (type <code>RF-IsolatorClassificationType</code>)</li> <li><b>Limiter</b> (type <code>RF-LimiterClassificationType</code>)</li> <li><b>Multiplier</b> (type <code>RF-MultiplierClassificationType</code>)</li> <li><b>PhaseShifter</b> (type <code>RF-PhaseShifterClassificationType</code>)</li> <li><b>Receiver</b> (type <code>RF-ReceiverClassificationType</code>)</li> <li><b>Mixer</b> (type <code>RF-MixerClassificationType</code>)</li> <li><b>Tranceiver</b> (type <code>RF-TranceiverClassificationType</code>)</li> <li><b>Transmitter</b> (type <code>RF-TransmitterClassificationType</code>)</li> <li><b>Other</b> (type <code>OtherRF-ClassificationType</code>)</li> </ul> <p>At the bottom of the dashed box is a <b>Property-Array</b> class (type <code>RF-ClassificationProperty-ArrayType</code>).</p>
type	RF-ClassificationType, RF-AntennaClassificationType, RF-AttenuatorClassificationType, RF-BalunClassificationType, RF-CirculatorClassificationType, RF-CombinerClassificationType, RF-CouplerClassificationType, RF-DetectorClassificationType, RF-DividerClassificationType, RF-IsolatorClassificationType, RF-LimiterClassificationType, RF-MultiplierClassificationType, RF-PhaseShifterClassificationType, RF-ReceiverClassificationType, RF-MixerClassificationType, RF-TranceiverClassificationType, RF-TransmitterClassificationType, OtherRF-ClassificationType, RF-ClassificationProperty-ArrayType

#### 4.5.1.3.17 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*.

##### 4.5.1.3.17.1. RF Detector Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/RF/Detector</a>
diagram	<p>The diagram shows a 'Detector' block with type 'RF-DetectorClassificationType'. It is connected to a 'Property-Array' block with type 'RF-DetectorClassificationProperty-ArrayType'. This array contains three elements: 'BalancedInput' (type 'BalancedInputRF-DetectorClassificationType'), 'UnBalancedInput' (type 'UnBalancedInputRF-DetectorClassificationType'), and another 'Property-Array' (type 'RF-DetectorClassificationProperty-ArrayType').</p>
type	<a href="#">RF-DetectorClassificationType</a> , <a href="#">BalancedInputRF-DetectorClassificationType</a> , <a href="#">UnBalancedInputRF-DetectorClassificationType</a> , <a href="#">RF-DetectorClassificationProperty-ArrayType</a>

##### 4.5.1.3.17.2. RF Mixer Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/RF/Mixer</a>
diagram	<p>The diagram shows a 'Mixer' block with type 'RF-MixerClassificationType'. It is connected to a 'Property-Array' block with type 'RF-MixerClassificationProperty-ArrayType'. This array contains four elements: 'Balanced' (type 'BalancedRF-MixerClassificationType'), 'DoubleBalanced' (type 'DoubleBalancedRF-MixerClassificationType'), 'TripleBalanced' (type 'TripleBalancedRF-MixerClassificationType'), and 'Other' (type 'OtherRF-MixerClassificationType'). The 'Other' element is further connected to another 'Property-Array' block with type 'OtherRF-MixerClassificationProperty-ArrayType', which contains a 'Sub-CategoryName' (type 'sys:string') and another 'Property-Array' (type 'OtherRF-MixerClassificationProperty-ArrayType').</p>
type	<a href="#">RF-MixerClassificationType</a> , <a href="#">BalancedRF-MixerClassificationType</a> , <a href="#">DoubleBalancedRF-MixerClassificationType</a> , <a href="#">TripleBalancedRF-MixerClassificationType</a> , <a href="#">OtherRF-MixerClassificationType</a> , <a href="#">OtherRF-MixerClassificationProperty-ArrayType</a> , <a href="#">RF-MixerClassificationProperty-ArrayType</a>

**4.5.1.3.17.3. Other RF Classification**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/RF/Other</a>
diagram	<p>The diagram illustrates the structure of the <b>OtherRF-ClassificationType</b> class. It is a class with a property <b>Other</b> of type <b>OtherRF-ClassificationType</b>. It also has a dashed box containing <b>Sub-CategoryName</b> of type <b>xs:string</b> and <b>Property-Array</b> of type <b>OtherRF-ClassificationProperty-ArrayType</b>.</p>
type	<a href="#">OtherRF-ClassificationType</a> , <a href="#">OtherRF-ClassificationProperty-ArrayType</a>

**4.5.1.3.18. Sensor Classification**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Sensor</a>
diagram	<pre> classDiagram     class SensorClassificationType {         +Accelerometer         +Capacitive-Touch         +Current         +Encoder         +Flow         +Gyro         +Humidity         +Magnetic         +Photo         +Pressure         +Resistance         +Temperature         +Tilt         +Vibration         +Other         +Property-Array     }     class Accelerometer {         type AccelerometerSensorClassificationType     }     class Capacitive-Touch {         type Capacitive-TouchSensorClassificationType     }     class Current {         type CurrentSensorClassificationType     }     class Encoder {         type EncoderSensorClassificationType     }     class Flow {         type FlowSensorClassificationType     }     class Gyro {         type GyroSensorClassificationType     }     class Humidity {         type HumiditySensorClassificationType     }     class Magnetic {         type MagneticSensorClassificationType     }     class Photo {         type PhotoSensorClassificationType     }     class Pressure {         type PressureSensorClassificationType     }     class Resistance {         type ResistanceSensorClassificationType     }     class Temperature {         type TemperatureSensorClassificationType     }     class Tilt {         type TiltSensorClassificationType     }     class Vibration {         type VibrationSensorClassificationType     }     class Other {         type OtherSensorClassificationType     }     class Property-Array {         type SensorClassificationProperty-ArrayType     }     class Sensor {         type SensorClassificationType     }     SensorClassificationType &lt; -- Accelerometer     SensorClassificationType &lt; -- Capacitive-Touch     SensorClassificationType &lt; -- Current     SensorClassificationType &lt; -- Encoder     SensorClassificationType &lt; -- Flow     SensorClassificationType &lt; -- Gyro     SensorClassificationType &lt; -- Humidity     SensorClassificationType &lt; -- Magnetic     SensorClassificationType &lt; -- Photo     SensorClassificationType &lt; -- Pressure     SensorClassificationType &lt; -- Resistance     SensorClassificationType &lt; -- Temperature     SensorClassificationType &lt; -- Tilt     SensorClassificationType &lt; -- Vibration     SensorClassificationType &lt; -- Other     SensorClassificationType &lt; -- Property-Array     Sensor --&gt; SensorClassificationType   </pre> <p>The diagram illustrates the Sensor Classification hierarchy. A central <b>SensorClassificationType</b> class (highlighted in yellow) serves as the base for various sensor types. These include Accelerometer, Capacitive-Touch, Current, Encoder, Flow, Gyro, Humidity, Magnetic, Photo, Pressure, Resistance, Temperature, Tilt, Vibration, and Other. Each of these sensor types is represented by a class box with a 'type' attribute pointing to a specific <i>SensorClassificationType</i> (e.g., AccelerometerSensorClassificationType). Additionally, a <b>Property-Array</b> class is shown with a 'type' attribute pointing to <i>SensorClassificationProperty-ArrayType</i>. A <b>Sensor</b> class is also present, with a 'type' attribute pointing to <i>SensorClassificationType</i>. The relationships are indicated by solid lines with open arrowheads pointing towards the base class.</p>
type	<a href="#">SensorClassificationType</a> , <a href="#">AccelerometerSensorClassificationType</a> , <a href="#">Capacitive-TouchSensorClassificationType</a> , <a href="#">CurrentSensorClassificationType</a> , <a href="#">EncoderSensorClassificationType</a> , <a href="#">FlowSensorClassificationType</a> , <a href="#">GyroSensorClassificationType</a> , <a href="#">HumiditySensorClassificationType</a> , <a href="#">MagneticSensorClassificationType</a> , <a href="#">PhotoSensorClassificationType</a> , <a href="#">PressureSensorClassificationType</a> , <a href="#">ResistanceSensorClassificationType</a> , <a href="#">TemperatureSensorClassificationType</a> , <a href="#">TiltSensorClassificationType</a> , <a href="#">VibrationSensorClassificationType</a> , <a href="#">OtherSensorClassificationType</a> , <a href="#">SensorClassificationProperty-ArrayType</a>

#### 4.5.1.3.18 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*.

##### 4.5.1.3.18.1. Other Sensor Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Sensor/Other</a>
diagram	<pre> graph LR     Other[Other type   OtherSensorClassificationType] --- ComplexType[OtherSensorClassificationType]     subgraph ComplexType [OtherSensorClassificationType]         direction TB         SubCategoryName[Sub-CategoryName type   xs:string]         PropertyArray[Property-Array type   OtherSensorClassificationProperty-ArrayType]     end </pre>
type	<a href="#">OtherSensorClassificationType</a> , <a href="#">OtherSensorClassificationProperty-ArrayType</a>

4.5.1.3.19. Switch Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Switch
diagram	<pre>classDiagram     class Switch {         type SwitchClassificationType     }     class SwitchClassificationType {         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 OtherSwitchClassificationType     }     class PropertyArray {         type SwitchClassificationProperty-ArrayType     }      Switch --&gt; SwitchClassificationType     SwitchClassificationType &lt; -- DIP     SwitchClassificationType &lt; -- KeyLock     SwitchClassificationType &lt; -- Rotary     SwitchClassificationType &lt; -- Rocker     SwitchClassificationType &lt; -- PushButton     SwitchClassificationType &lt; -- Reed     SwitchClassificationType &lt; -- Slide     SwitchClassificationType &lt; -- Tactile     SwitchClassificationType &lt; -- Toggle     SwitchClassificationType &lt; -- Other     SwitchClassificationType --&gt; PropertyArray</pre>
type	SwitchClassificationType, DIP-SwitchClassificationType, KeyLockSwitchClassificationType, RotarySwitchClassificationType, RockerSwitchClassificationType, PushButtonSwitchClassificationType, ReedSwitchClassificationType, SlideSwitchClassificationType, TactileSwitchClassificationType, ToggleSwitchClassificationType, OtherSwitchClassificationType, SwitchClassificationProperty-ArrayType.

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.



**4.5.1.3.19.1. Other Switch Classification**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Switch/Other</a>
diagram	<p>The diagram illustrates the structure of the <b>OtherSwitchClassificationType</b>. It is a class with a type attribute <b>OtherSwitchClassificationType</b>. This class is composed of a complex type, also named <b>OtherSwitchClassificationType</b> (highlighted in yellow). This complex type contains a required attribute <b>Sub-CategoryName</b> of type <b>xs:string</b> and a <b>Property-Array</b> of type <b>OtherSwitchClassificationProperty-ArrayType</b>.</p>
type	<a href="#">OtherSwitchClassificationType</a> , <a href="#">OtherSwitchClassificationProperty-ArrayType</a>

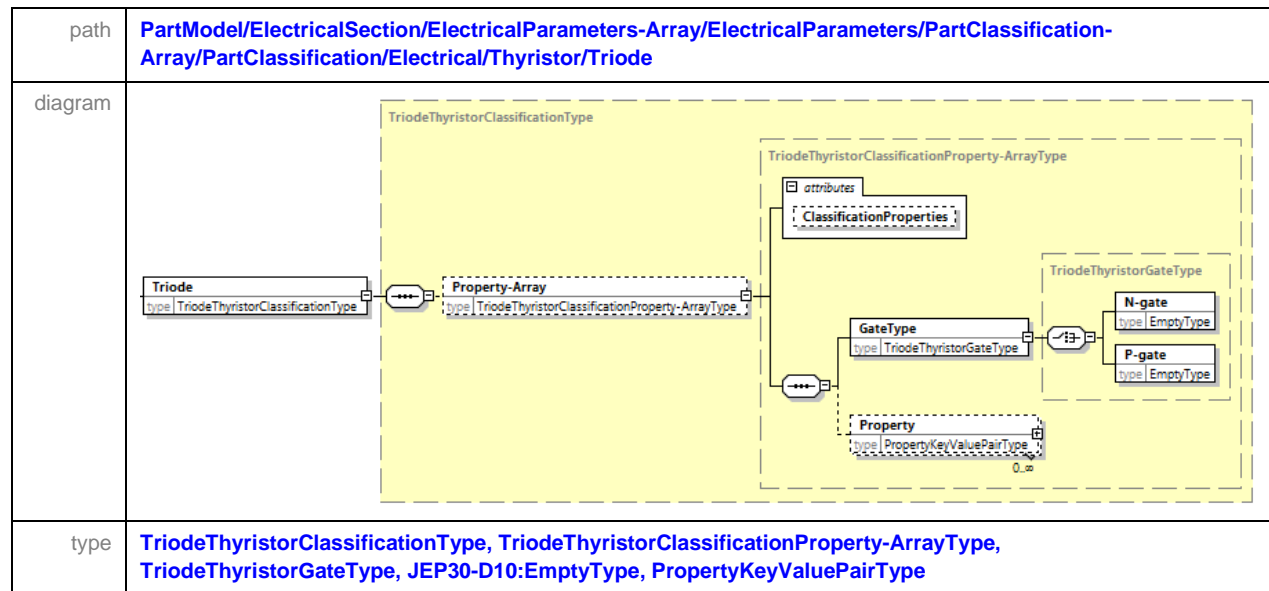
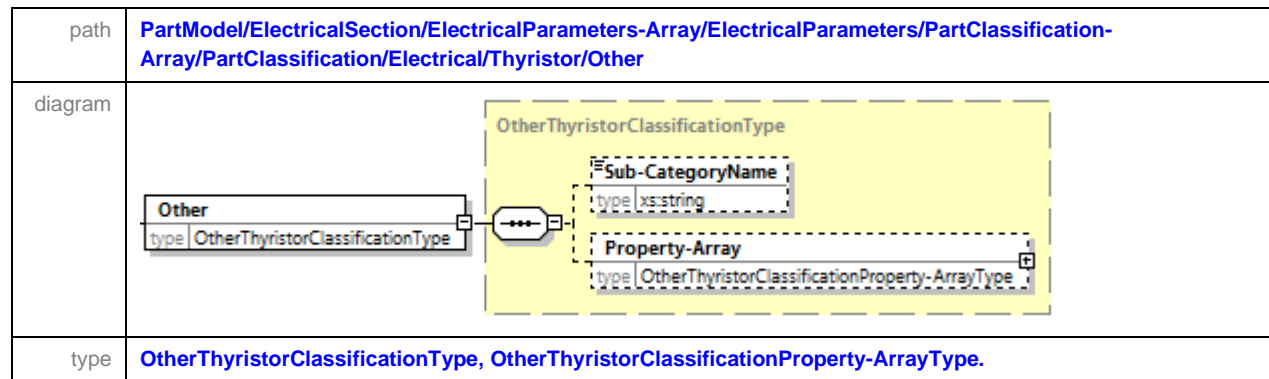
4.5.1.3.19.2. Switch Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Switch/Property-Array
diagram	<p>The diagram illustrates the structure of the <b>SwitchClassificationProperty-ArrayType</b>. It features a main container with an <b>attributes</b> section containing <b>ClassificationProperties</b>. A <b>Property-Array</b> box is linked to the main container, with its type specified as <b>SwitchClassificationProperty-ArrayType</b>. A <b>ContactForm</b> box, with type <b>SwitchContactFormType</b>, is also linked. A <b>Property</b> box, with type <b>PropertyKey/ValuePairType</b>, is shown with a multiplicity of <b>0..2</b>. The <b>SwitchContactFormType</b> box contains several sub-entities, each with a type of <b>EmptyType</b>: <b>SinglePole-SingleThrow</b>, <b>SinglePole-DoubleThrow</b>, <b>SinglePole-ManyThrow</b>, <b>DoublePole-SingleThrow</b>, <b>DoublePole-DoubleThrow</b>, <b>DoublePole-ManyThrow</b>, <b>MomentarySwitch-Make</b>, <b>MomentarySwitch-Break</b>, and <b>MomentarySwitch-TwoCircuit</b>. Multiplicities <b>0..1</b> and <b>0..2</b> are indicated near the <b>Property</b> box.</p>
type	SwitchClassificationProperty-ArrayType, SwitchContactFormType, JEP30-D10:EmptyType, PropertyKey/ValuePairType

#### 4.5.1.3.20. Thyristor Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Thyristor</a>
diagram	
type	<a href="#">ThyristorClassificationType</a> , <a href="#">DIAC-ThyristorClassificationType</a> , <a href="#">MCT-ThyristorClassificationType</a> , <a href="#">SCR-ThyristorClassificationType</a> , <a href="#">SIDAC-ThyristorClassificationType</a> , <a href="#">TRIAC-ThyristorClassificationType</a> , <a href="#">TriodeThyristorClassificationType</a> , <a href="#">OtherThyristorClassificationType</a> , <a href="#">ThyristorClassificationProperty-ArrayType</a>

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).

**4.5.1.3.20.1. Triode Thyristor Classification****4.5.1.3.20.2. Other Thyristor Classification**

#### 4.5.1.3.20.3. Thyristor Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Thyristor/Property-Array</a>
diagram	<p>The diagram illustrates the structure of the <b>ThyristorClassificationProperty-ArrayType</b>. It is a class with an <b>attributes</b> compartment containing a <b>ClassificationProperties</b> compartment. The main body of the class is divided into three sections: <b>Direction</b>, <b>Conduction</b>, and <b>Property</b>. The <b>Direction</b> section is of type <b>ThyristorDirectionType</b> and contains a <b>Direction</b> compartment of type <b>ThyristorDirectionType</b>, which further branches into <b>Bidirectional</b> (type <b>EmptyType</b>) and <b>UnidirectionalThyristor</b> (type <b>EmptyType</b>). The <b>Conduction</b> section is of type <b>ThyristorConductionType</b> and contains a <b>Conduction</b> compartment of type <b>ThyristorConductionType</b>, which branches into <b>Reverse-Blocking</b> (type <b>EmptyType</b>) and <b>Reverse-Conducting</b> (type <b>EmptyType</b>). The <b>Property</b> section is of type <b>PropertyKeyValuePairType</b> and contains a <b>Property</b> compartment of type <b>PropertyKeyValuePairType</b>. The <b>Property</b> compartment has a multiplicity of <b>0..∞</b>.</p>
type	<a href="#">ThyristorClassificationProperty-ArrayType</a> , <a href="#">ThyristorDirectionType</a> , <a href="#">ThyristorConductionType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValuePairType</a>

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

**4.5.1.3.21. Transformer Classification**

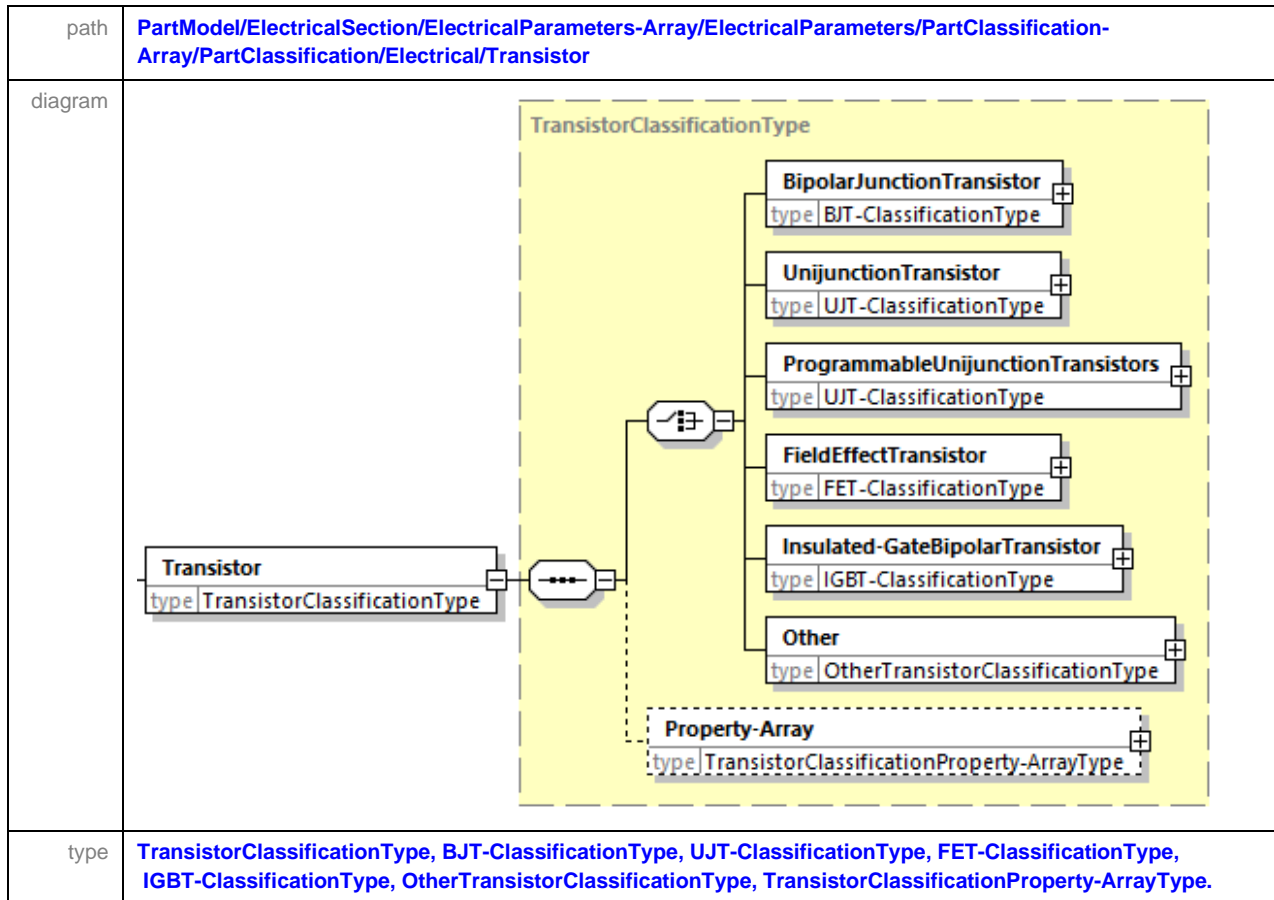
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transformer</a>
diagram	
type	<a href="#">TransformerClassificationType</a> , <a href="#">SignalTransformerClassificationType</a> , <a href="#">PowerTransformerClassificationType</a> , <a href="#">OtherTransformerClassificationType</a> , <a href="#">TransformerClassificationProperty-ArrayType</a> .

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.

**4.5.1.3.21.1. Other Transformer Classification**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Thyristor/Other</a>
diagram	
type	<a href="#">OtherTransformerClassificationType</a> , <a href="#">OtherTransformerClassificationType-ArrayType</a> .

#### 4.5.1.3.22. Transistor Classification



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.

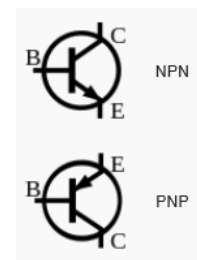
There are several classifications of transistors as shown above, however new classifications can be captured under the category [Other](#).

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalPartClassification-Array/PartClassification/Electrical/Transistor/BipolarJunctionTransistor
diagram	<pre> graph LR     BJT[BJT-ClassificationType] --&gt; BJT_Array[BJT-ClassificationProperty-ArrayType]     BJT_Array --&gt; ClassificationProperties[ClassificationProperties type EmptyType]     BJT_Array --&gt; BJT_Type[BJT-Type type BJT-Type]     BJT_Type --&gt; NPN_PNP[NPN-PNP type EmptyType]     BJT_Type --&gt; NPN[NPN type EmptyType]     BJT_Type --&gt; PNP[PNP type EmptyType]     BJT_Type --&gt; DarlingtonNPN[DarlingtonNPN type EmptyType]     BJT_Type --&gt; DarlingtonPNP[DarlingtonPNP type EmptyType]     BJT_Type --&gt; Other1[Other type xs:string]     BJT_Type --&gt; Material[Material type BJT-MaterialPropertyType]     Material --&gt; GaAs[GaAs type EmptyType]     Material --&gt; GaN[GaN type EmptyType]     Material --&gt; Other2[Other type xs:string]     BJT_Array --&gt; Property_Array[Property-Array type BJT-ClassificationProperty-ArrayType]     Property_Array --&gt; Property[Property type PropertyKeyValuePairType]     </pre>
type	BJT-ClassificationType, BJT-ClassificationProperty-ArrayType, JEP30-D10:EmptyType, BJT-Type, BJT-MaterialPropertyType, PropertyKeyValuePairType.



### Figure 4— NPN Darlington

A *Bipolar Junction Transistor* (bipolar transistor or BJT) is a type of transistor that uses both electron and hole charge carriers. BJT's 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



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.

#### 4.5.1.3.22.2. Unijunction Transistor Classification and Property-Array

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

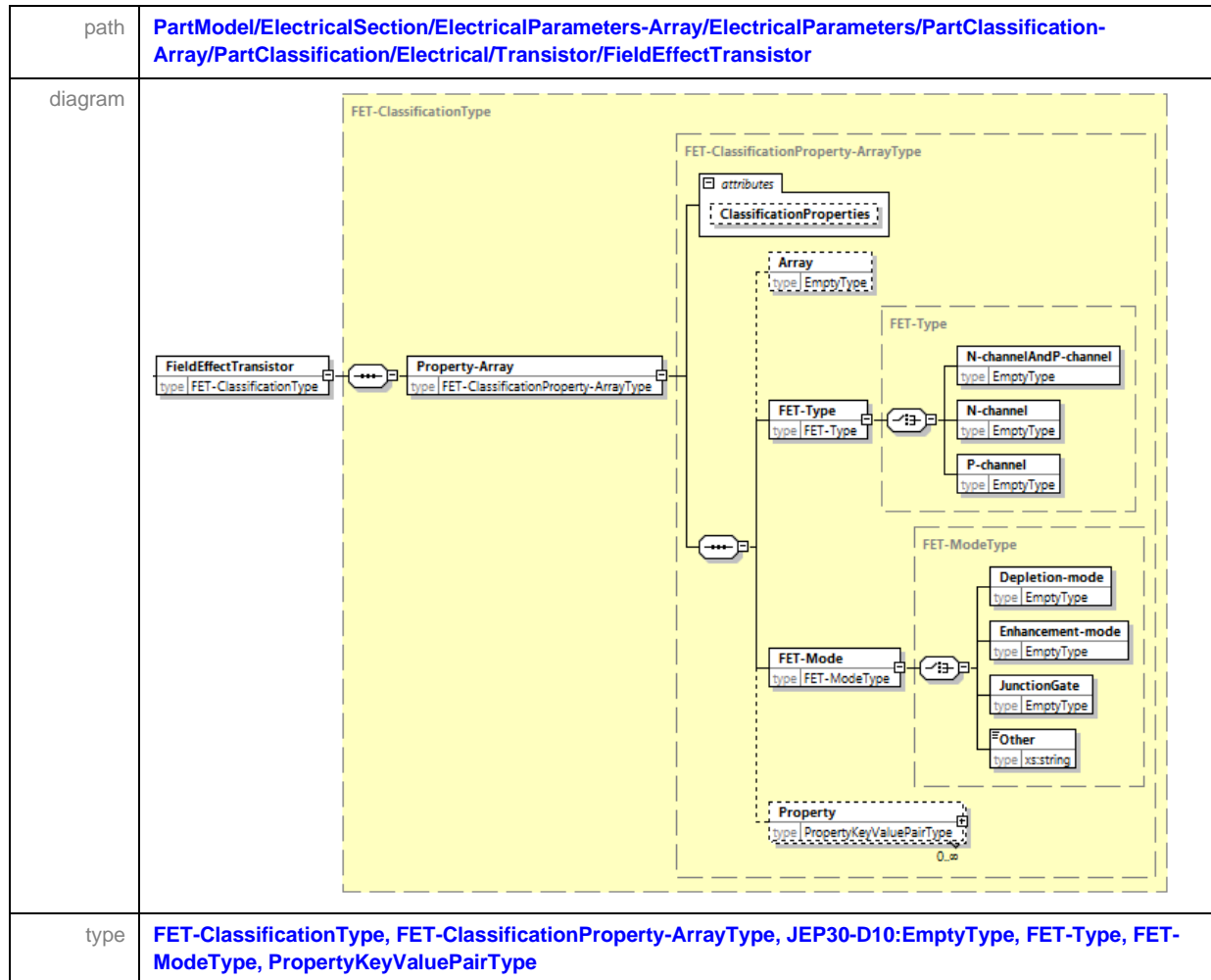
### Figure 6 — p-type UJT

4.5.1.3.22.3. Programmable Unijunction Transistor Classification and Property-Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/UnijunctionTransistor
diagram	<p>The diagram illustrates the relationships between several classes in a UML model. A <b>ProgrammableUnijunctionTransistors</b> class (type <code>UJT-ClassificationType</code>) is associated with a <b>Property-Array</b> class (type <code>UJT-ClassificationProperty-ArrayType</code>) via a directed association with a multiplicity of 4. The <b>Property-Array</b> class is contained within a dashed boundary labeled <b>UJT-ClassificationType</b>. Inside this boundary, the <b>Property-Array</b> class is associated with a <b>UJT-Type</b> class (type <code>UJT-Type</code>) via a directed association with a multiplicity of 4. The <b>UJT-Type</b> class is also contained within the <b>UJT-ClassificationType</b> boundary. The <b>UJT-Type</b> class is associated with a <b>Property</b> class (type <code>PropertyKeyValuePairType</code>) via a directed association with a multiplicity of 4. The <b>Property</b> class is also contained within the <b>UJT-ClassificationType</b> boundary. The <b>UJT-Type</b> class is associated with a <b>UJT-ClassificationProperty-ArrayType</b> class (type <code>UJT-ClassificationProperty-ArrayType</code>) via a directed association with a multiplicity of 4. The <b>UJT-ClassificationProperty-ArrayType</b> class is also contained within the <b>UJT-ClassificationType</b> boundary. The <b>UJT-ClassificationProperty-ArrayType</b> class has an <b>attributes</b> compartment containing a <b>ClassificationProperties</b> class (type <code>UJT-Type</code>). The <b>UJT-Type</b> class has two subclasses: <b>N-emitter</b> (type <code>EmptyType</code>) and <b>P-emitter</b> (type <code>EmptyType</code>). The <b>Property</b> class has a multiplicity of 0..∞.</p>
type	UJT-ClassificationType, UJT-ClassificationProperty-ArrayType, UJT-Type, JEP30-D10:EmptyType, PropertyKeyValuePairType

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

#### 4.5.1.3.22.4. Field Effect Transistor Classification and Property-Array



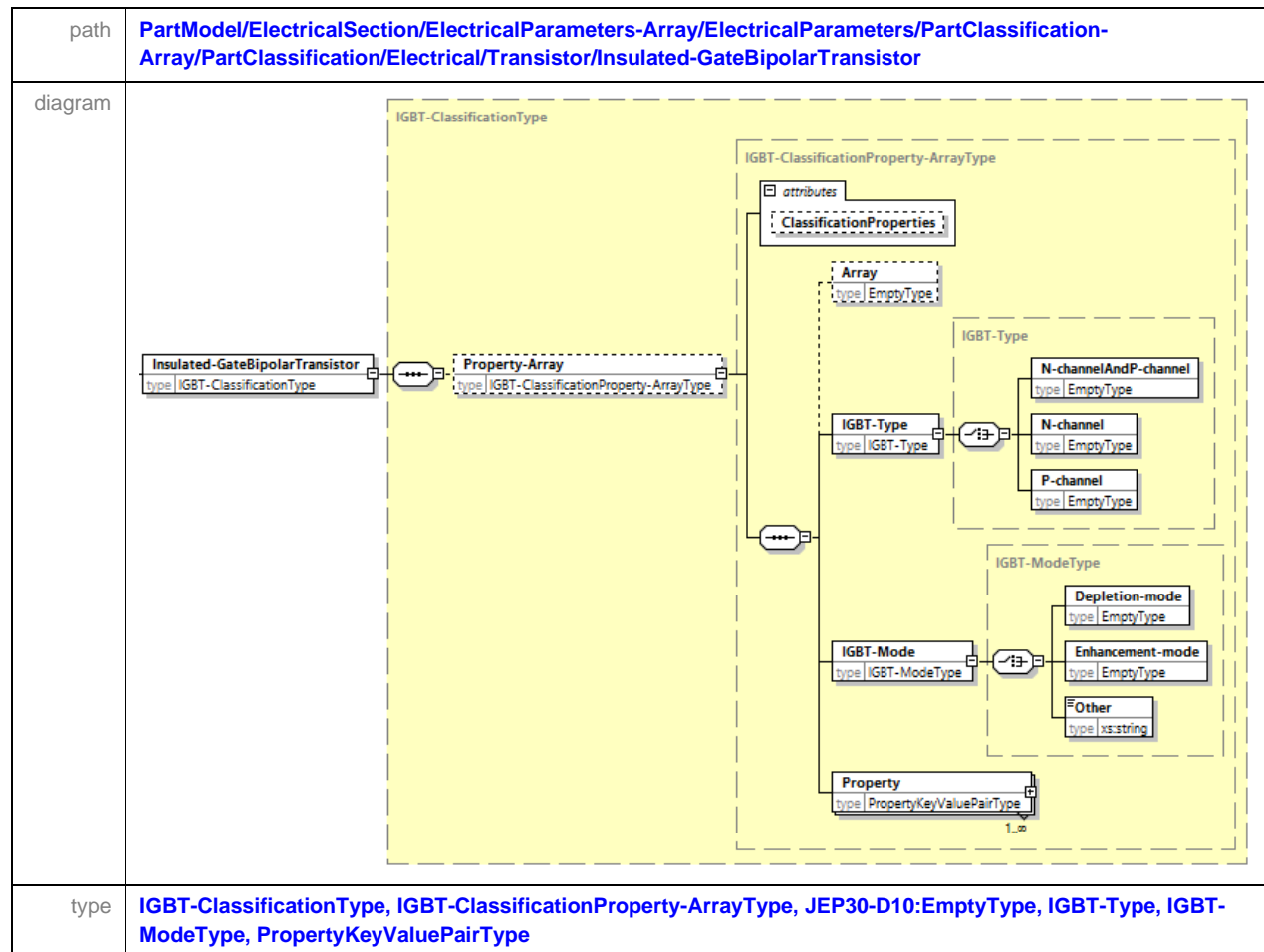
The [FieldEffectTransistor](#) (FET) is a transistor that uses an electric field to control the electrical behaviour of the device. FET's are also known as unipolar transistors since they involve single-carrier-type operation. Many different implementations of field effect transistors exist, as shown 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.

#### 4.5.1.3.22.5. Insulated Gate Bipolar Transistor Classification and Property-Array



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*.

#### 4.5.1.3.22.6. Other Transistor Classification and Property-Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Transistor/Other</a>
diagram	
type	<a href="#">OtherTransistorClassificationType</a> , <a href="#">OtherTransistorClassificationProperty-ArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">PropertyKeyValueType</a> .

#### 4.5.1.3.23. Tube Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Tube</a>
diagram	
type	<a href="#">TubeClassificationType</a> , <a href="#">GasDischargeTubeClassificationType</a> , <a href="#">VacuumFluorescentDisplayTubeClassificationType</a> , <a href="#">OtherTubeClassificationType</a> , <a href="#">OtherTubeClassificationProperty-ArrayType</a> , <a href="#">TubeClassificationProperty-ArrayType</a>

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

#### 4.5.1.3.23. Tube Classification (cont'd)

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.



Figure 8 — Vacuum Fluorescent Display

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*.

#### 4.5.1.3.24. Other Electrical Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Electrical/Other</a>
diagram	<pre> classDiagram     class Other {         type OtherElectricalClassificationType     }     class OtherElectricalClassificationType {         Sub-CategoryName xs:string         Property-Array OtherElectricalClassificationProperty-ArrayType     }     Other --&gt; OtherElectricalClassificationType </pre>
type	<a href="#">OtherElectricalClassificationType</a> , <a href="#">OtherElectricalClassificationProperty-ArrayType</a>

#### 4.5.1.4. Hardware Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Hardware</b>
diagram	
type	<b>HardwareClassificationType, BatteryAccessoriesClassificationType, CableHardwareClassificationType, HardwareClampClassificationType, HardwareClipsClassificationType, EMI-ShieldClassificationType, FaceplateClassificationType, HardwareFastenersClassificationType, GasketClassificationType, HeatsinkClassificationType, HardwareInsulatorsMountsSpacersClassificationType, JackScrewsClassificationType, MetalCageClassificationType, HardwareNutClassificationType, PickAndPlaceHardwareClassificationType, HardwareScrewClassificationType, HardwareSpringClassificationType, StudsClassificationType, HardwareWasherClassificationType, OtherHardwareClassificationType, HardwareClassificationProperty-ArrayType.</b>

##### 4.5.1.4.1. MetalCage Classification

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Hardware/MetalCage</b>
diagram	
type	<b>MetalCageClassificationType, MetalCageClassificationProperty-ArrayType, JEP30-D10:EmptyType, PropertyKeyValuePairType.</b>

4.5.1.4.2. Other Hardware Classification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Hardware/Other
diagram	<p>The diagram illustrates the structure of the <code>OtherHardwareClassificationType</code>. It is a complex type containing the following elements:</p> <ul style="list-style-type: none"><li><b>Sub-CategoryName</b>: An attribute of type <code>xs:string</code>.</li><li><b>Property-Array</b>: A collection of type <code>OtherHardwareClassificationProperty-ArrayType</code>.</li><li><b>ClassificationProperties</b>: An attribute of type <code>PropertyKeyValuePairType</code> with a cardinality of <code>1..∞</code>.</li></ul> <p>The <code>OtherHardwareClassificationType</code> is associated with the <code>Other</code> element, which has a type of <code>OtherHardwareClassificationType</code>.</p>
type	OtherHardwareClassificationType, OtherHardwareClassificationProperty-ArrayType, PropertyKeyValuePairType.



#### 4.5.1.5. Optics Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Optics</a>
diagram	<pre> classDiagram     class Optics {         type OpticsClassificationType     }     class Amplifier {         type OpticalAmplifierClassificationType     }     class Attenuator {         type OpticalAttenuatorClassificationType     }     class Circulator {         type OpticalCirculatorClassificationType     }     class Coupler {         type OpticalCouplerClassificationType     }     class Demultiplexers {         type OpticalDemultiplexersClassificationType     }     class Receiver {         type OpticalReceiverClassificationType     }     class Switch {         type OpticalSwitchClassificationType     }     class Transceivers {         type OpticalTransceiversClassificationType     }     class Transmitters {         type OpticalTransmittersClassificationType     }     class Other {         type OtherOpticalClassificationType     }     class PropertyArray {         type OpticClassificationProperty-ArrayType     }      Optics --&gt; PropertyArray     PropertyArray --&gt; Amplifier     PropertyArray --&gt; Attenuator     PropertyArray --&gt; Circulator     PropertyArray --&gt; Coupler     PropertyArray --&gt; Demultiplexers     PropertyArray --&gt; Receiver     PropertyArray --&gt; Switch     PropertyArray --&gt; Transceivers     PropertyArray --&gt; Transmitters     PropertyArray --&gt; Other   </pre>
type	<a href="#">OpticsClassificationType</a> , <a href="#">OpticalAmplifierClassificationType</a> , <a href="#">OpticalAttenuatorClassificationType</a> , <a href="#">OpticalCirculatorClassificationType</a> , <a href="#">OpticalCouplerClassificationType</a> , <a href="#">OpticalDemultiplexersClassificationType</a> , <a href="#">OpticalReceiverClassificationType</a> , <a href="#">OpticalSwitchClassificationType</a> , <a href="#">OpticalTransceiversClassificationType</a> , <a href="#">OpticalTransmittersClassificationType</a> , <a href="#">OtherOpticalClassificationType</a> , <a href="#">OpticClassificationProperty-ArrayType</a> .

#### 4.5.1.5.1. Other Optics Classification

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/PartClassification-Array/PartClassification/Optics/Other</a>
diagram	
type	<a href="#">OtherOpticalClassificationType</a> , <a href="#">OtherOpticalClassificationProperty-ArrayType</a> , <a href="#">PropertyKeyValuePairType</a> .

#### 4.5.2. Terminal Details - Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array</a>
diagram	
type	<a href="#">TerminalDetails-ArrayType</a> , <a href="#">TerminalDetailsType</a> , <a href="#">Properties-ArrayType</a> , <a href="#">TerminalFunction-ArrayType</a> , <a href="#">TerminalGroupingType</a> , <a href="#">ExternalConnection-ArrayType</a> .

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

Some of this detail 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.

## 4.5.2.1. Properties - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array
diagram	<p>The diagram illustrates the structure of the Properties-Array. It is a class of type Properties-ArrayType. It contains an array of Properties objects, each of type PropertiesType. The PropertiesType class has the following attributes and properties:</p> <ul style="list-style-type: none"> <li><b>ID</b>: type xs:string</li> <li><b>Digital</b>: type JEP30-D10:EmptyType</li> <li><b>Analog</b>: type AnalogConnectionType</li> <li><b>Power</b>: type PowerConnectionType</li> <li><b>Reserved</b>: type JEP30-D10:EmptyType</li> <li><b>NoDieConnection</b>: type JEP30-D10:EmptyType</li> <li><b>Ground</b>: type JEP30-D10:EmptyType</li> <li><b>ActiveLow</b>: type JEP30-D10:EmptyType</li> <li><b>ActiveHigh</b>: type JEP30-D10:EmptyType</li> <li><b>EdgeTriggered</b>: type JEP30-D10:EmptyType</li> <li><b>Amplifier</b>: type JEP30-D10:EmptyType</li> <li><b>Clamp</b>: type JEP30-D10:EmptyType</li> <li><b>Hysteresis</b>: type JEP30-D10:EmptyType</li> <li><b>Inversion</b>: type JEP30-D10:EmptyType</li> <li><b>SchmittTriggered</b>: type JEP30-D10:EmptyType</li> <li><b>TriState</b>: type JEP30-D10:EmptyType</li> <li><b>Direction</b>: type SignalDirectionType</li> <li><b>InternalPullupPulldown</b>: type SignalInternalPullupPulldownType</li> <li><b>SeriesComponent</b>: type SignalSeriesComponentType</li> <li><b>OutputCircuit</b>: type OutputCircuitPropertyType</li> <li><b>Reference</b>: type SignalReferenceType</li> </ul> <p>There are also constraints defined for the Properties-Array class.</p>
type	Properties-ArrayType, PropertiesType, JEP30-D10:EmptyType, AnalogConnectionType, PowerConnectionType, SignalDirectionType, SignalInternalPullUp-DownType, SignalSeriesComponentType, OutputCircuitPropertyType, SignalReferenceType

#### 4.5.2.1 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.

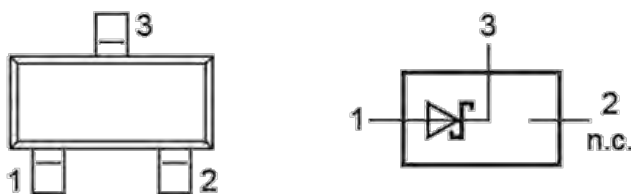


Figure 9 — Sample Device with no internal die connection

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.

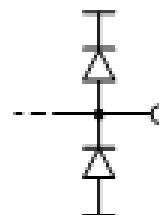
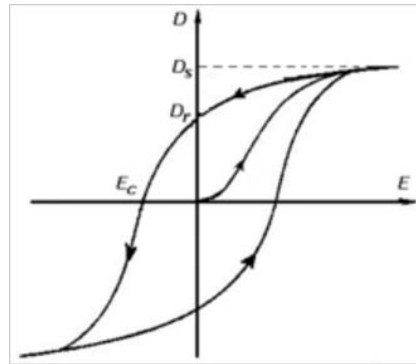


Figure 10 — Clamp Circuit

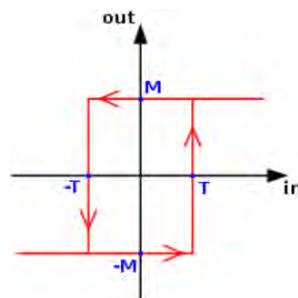
#### 4.5.2.1 Properties – Array (cont'd)

**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. **SchmittTriggered** 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**

#### 4.5.2.1 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.

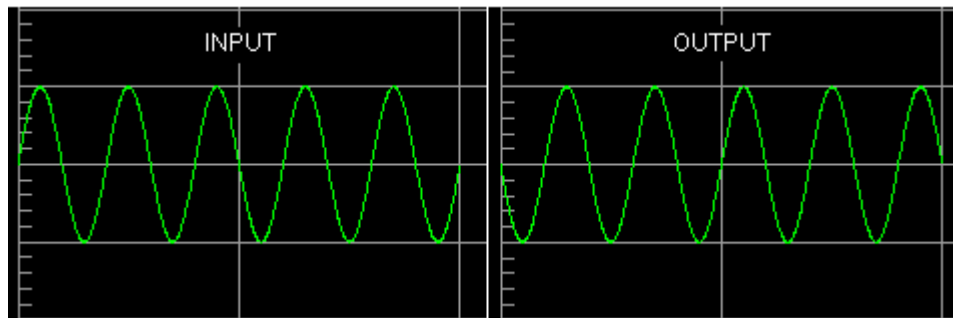


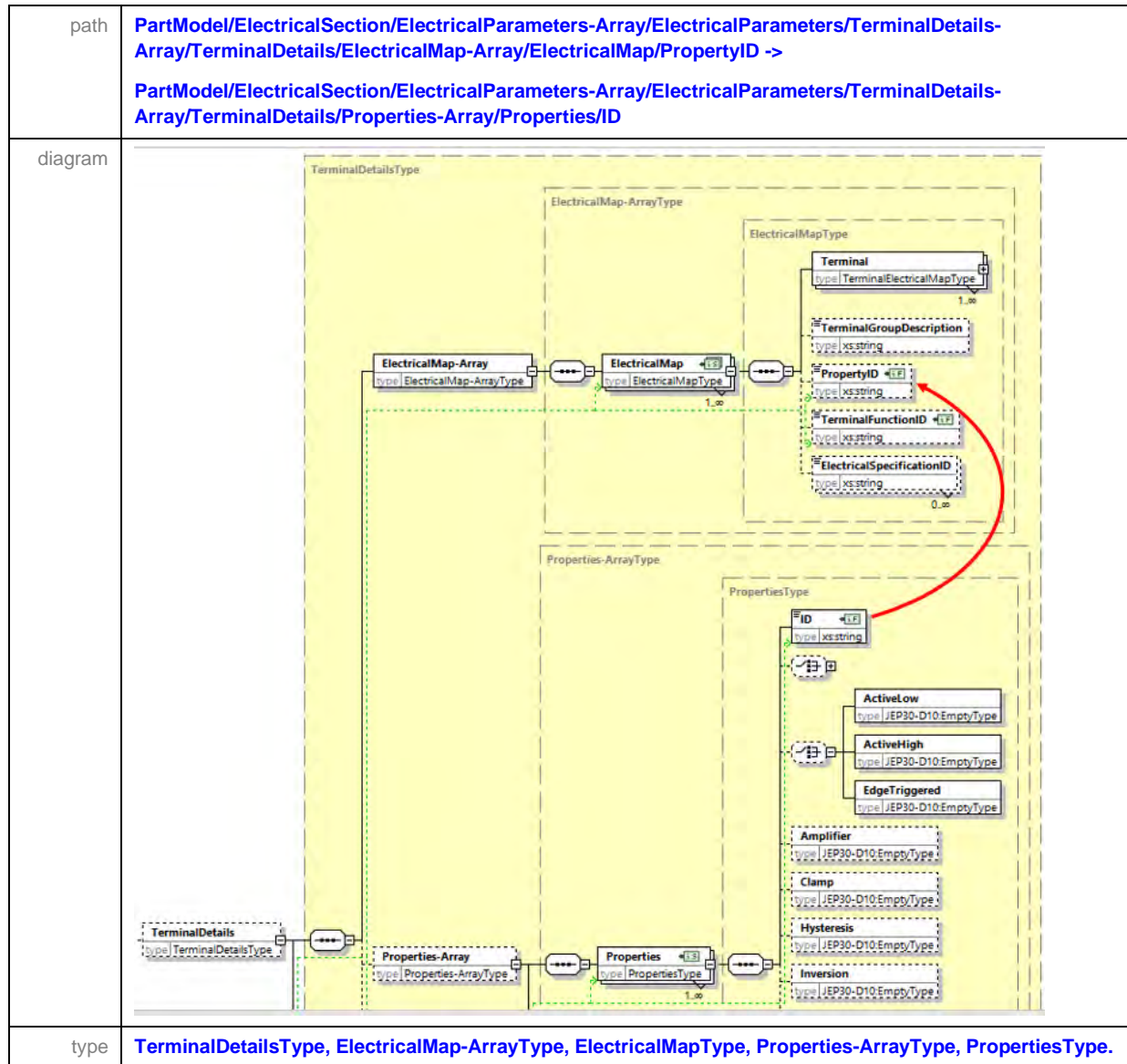
Figure 13 — Signal Inversion

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.



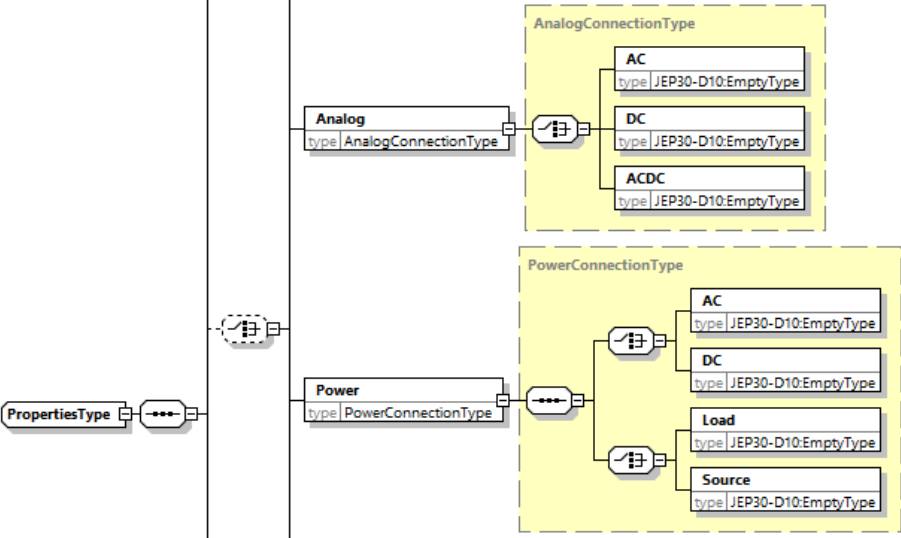
Figure 14 — Tri-State

#### 4.5.2.1.1. Linking the Terminal Map and the Internal Node to the Property ID



The [PropertyID](#) under [ElectricalMap](#) connect to the [Properties/ID](#) under the [Properties-Array](#), thereby connecting the set of electrical [Properties](#) identified under a specific ID back to the [Terminal](#).

4.5.2.1.2. Analog and Power Connection Type

path	<div>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/Analog.</a></div> <div>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/PowerConnectionType</a></div>		
diagram			
type	<a href="#">AnalogConnectionType</a> , <a href="#">PowerConnectionType</a> .		

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](#).

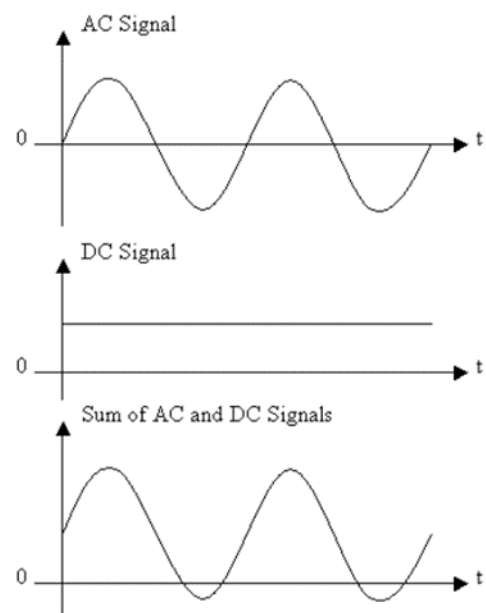
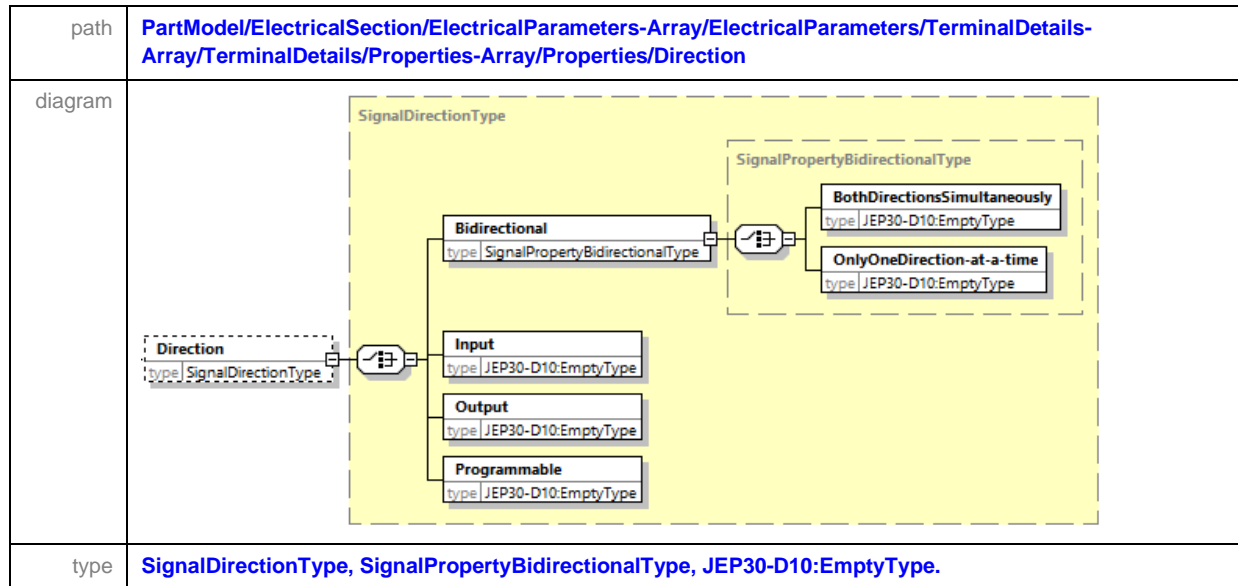


Figure 15 — Analog Connection Types



#### 4.5.2.1.3. Signal Direction



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:-

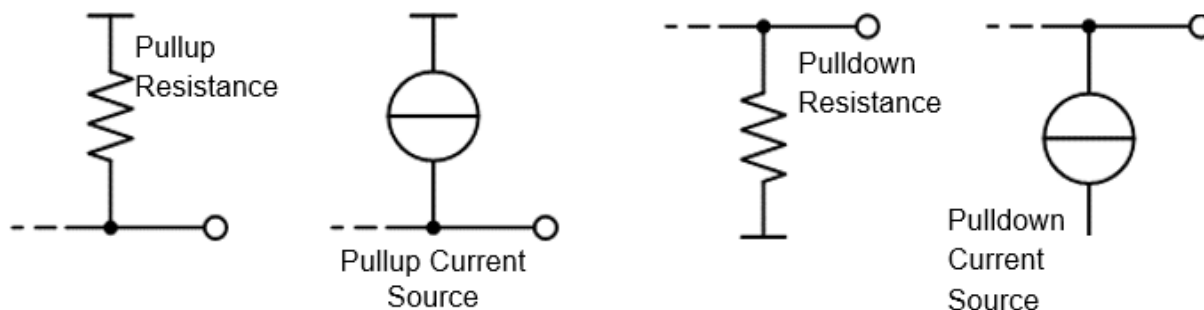
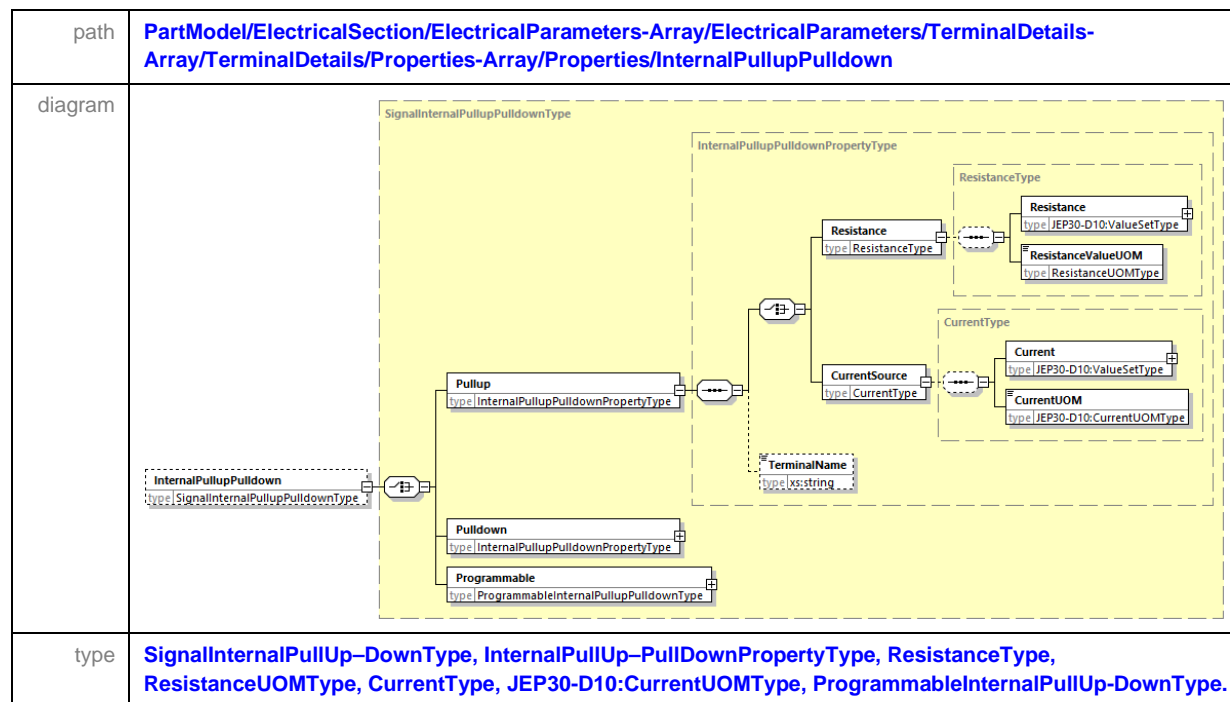
1. **BothDirectionsSimultaneously**,
2. **OnlyOneDirection-at-a-time**.

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.

#### 4.5.2.1.4. Internal Pullup / Pulldown



**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.

#### 4.5.2.1.5. Series Component

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/SeriesComponent</a>
diagram	
type	<a href="#">SignalSeriesComponentType</a> , <a href="#">ResistanceValueType</a> , <a href="#">ValueSetType</a> , <a href="#">ResistanceUOMType</a> , <a href="#">CapacitanceValueType</a> , <a href="#">CapacitanceUOMType</a> .

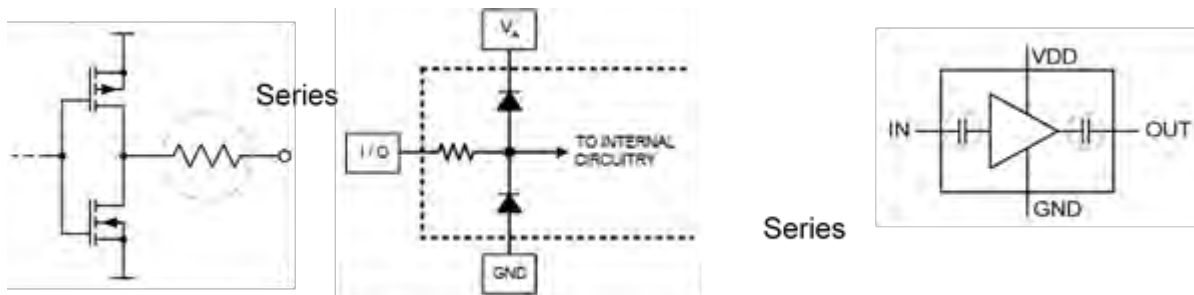


Figure 17 — Series Component Types

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.

**4.5.2.1.6. Output Circuit**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit</a>
diagram	<p>The diagram shows a hierarchy starting with a dashed box labeled <b>OutputCircuit</b> (type: OutputCircuitPropertyType). This connects to a central node, which then branches into three sub-properties: <b>Bipolar</b> (type: BipolarOutputCircuitType), <b>Unipolar</b> (type: UnipolarOutputCircuitType), and <b>Programmable</b> (type: ProgrammableOutputCircuitType). These three sub-properties are grouped within a larger dashed box labeled <b>OutputCircuitPropertyType</b>.</p>
type	<b>OutputCircuitPropertyType, BipolarOutputCircuitType, UnipolarOutputCircuitType, ProgrammableOutputCircuitType.</b>

*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”.

**4.5.2.1.6.1. Bipolar**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar</a>
diagram	<p>The diagram shows a hierarchy starting with a box labeled <b>Bipolar</b> (type: BipolarOutputCircuitType). This connects to a central node, which then branches into four sub-properties: <b>Passive-Pullup</b> (type: BipolarOutputPassive-PullupType), <b>Passive-Pulldown</b> (type: BipolarOutputPassive-Pulldown...), <b>Totem-Pole</b> (type: BipolarOutputTotem-PoleType), and <b>Push-Pull</b> (type: BipolarOutputPush-PullType). These four sub-properties are grouped within a larger dashed box labeled <b>BipolarOutputCircuitType</b>.</p>
type	<b>BipolarOutputCircuitType, BipolarOutputPassive-PullupType, BipolarOutputPassive-PulldownType, BipolarOutputTotem-PoleType, BipolarOutputPush-PullType,</b>

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

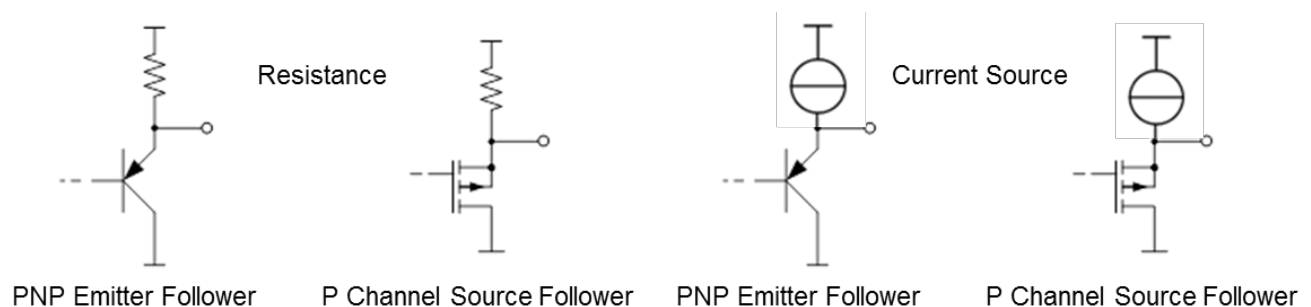
#### 4.5.2.1.6.1.1. Passive – Pullup

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar/Passive-Pullup</a>
diagram	
type	<a href="#">BipolarOutputPassive-PullupType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">ResistanceType</a> , <a href="#">CurrentType</a> , <a href="#">JEP30-D10:CurrentUOMType</a>

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-ChanellSourceFollower](#) is a [Passive-Pullup](#) output.



**Figure 18 — Passive Pull-up Output Circuit**

## 4.5.2.1.6.1.2. Passive - Pulldown

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar/Passive-Pulldown</a>
diagram	
type	<a href="#">BipolarOutputPassive-PulldownType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">ResistanceType</a> , <a href="#">ResistanceUOMType</a> , <a href="#">CurrentType</a> , <a href="#">JEP30-D10:CurrentUOMType</a> .

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-ChannelSourceFollower* are *Passive-Pulldown* outputs.

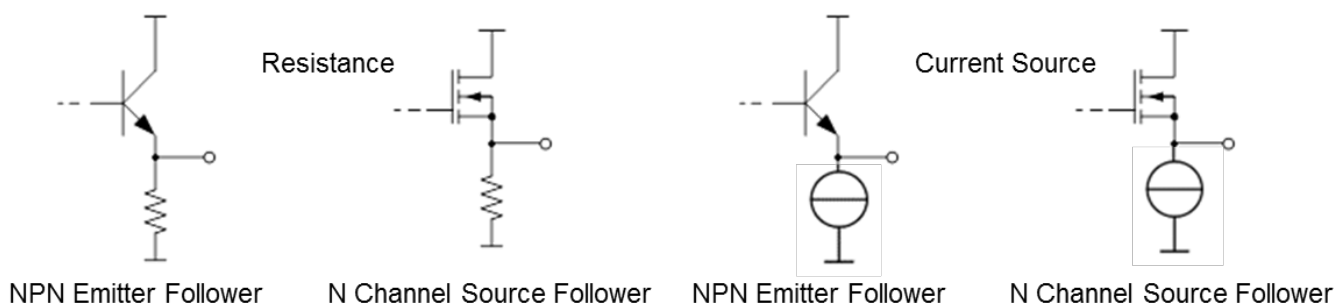


Figure 19 — Passive Pull-down Output Circuit

#### 4.5.2.1.6.1.3. Totem - Pole

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar/Totem-Pole</a>
diagram	
type	<a href="#">BipolarOutputTotem-PoleType</a> , <a href="#">JEP30-D10:EmptyType</a> .

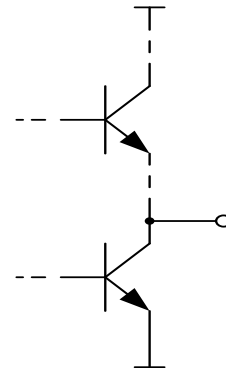
A **Totem-Pole** output is a bipolar output whose active devices are so controlled that as the resistance of one increases, the resistance of the other decreases so that, according to the relative states of the two active devices, the output voltage can swing between levels approaching the two supply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

4.5.2.1.6.1.4. Push - Pull

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Bipolar/Push-Pull
diagram	
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”.

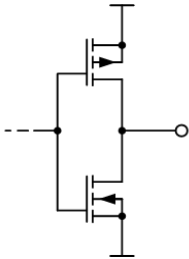
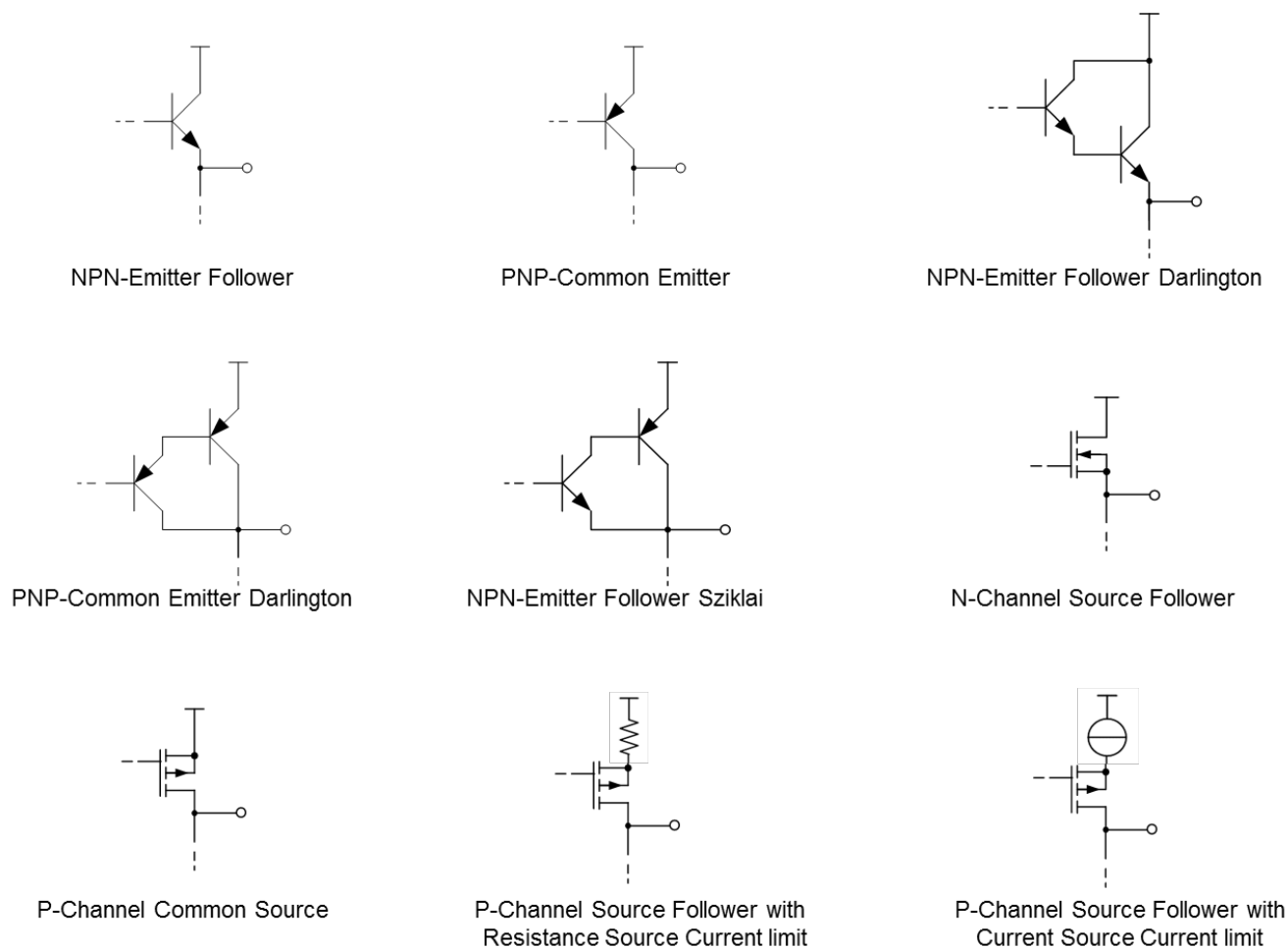


Figure 21 — Rail-to-Rail Push-Pull



#### 4.5.2.1.6.1.4 Push – Pull (cont'd)

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

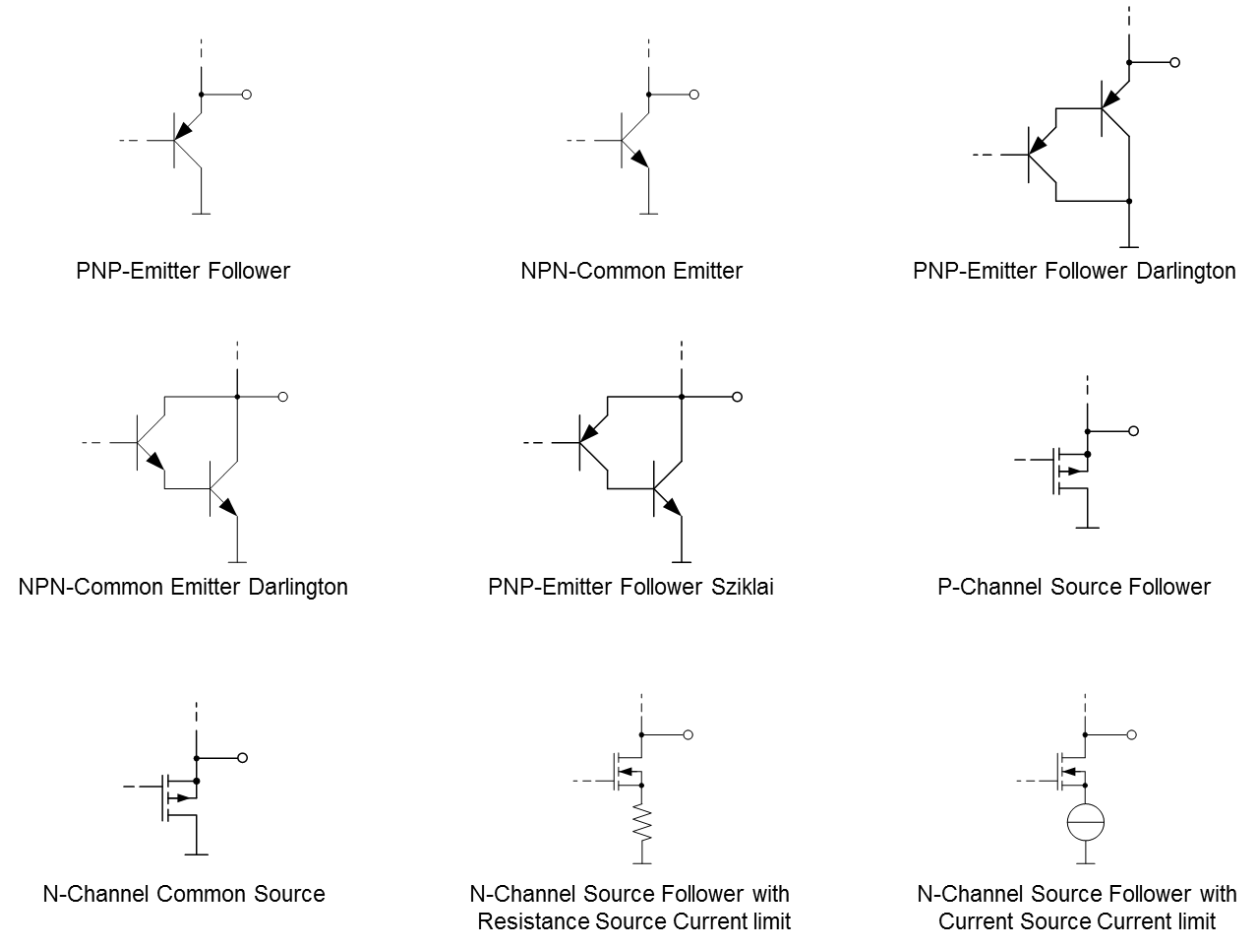


**Figure 22 — High Side Driver (Source Driver)**

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

**4.5.2.1.6.1.4 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

#### 4.5.2.1.6.2. Unipolar

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Unipolar</a>
diagram	<p>The diagram illustrates the internal structure of a Unipolar Output Circuit. It features a central 'Unipolar' block (type: UnipolarOutputCircuitType) connected to two main driver blocks: 'HighSideDriver' (type: UnipolarOutputSourceDriverType) and 'LowSideDriver' (type: UnipolarOutputSinkDriverType). The 'HighSideDriver' block is connected to a 'UnipolarOutputSourceDriverType' block, which contains four sub-components: 'NPNOpenEmitter' (type: EmptyType), 'PNPOpenCollector' (type: EmptyType), 'N-ChannelOpenSource' (type: EmptyType), and 'P-ChannelOpenDrain' (type: EmptyType). The 'LowSideDriver' block is connected to a 'UnipolarOutputSinkDriverType' block, which contains four sub-components: 'PNPOpenEmitter' (type: EmptyType), 'NPNOpenCollector' (type: EmptyType), 'P-ChannelOpenSource' (type: EmptyType), and 'N-ChannelOpenDrain' (type: EmptyType). The entire structure is enclosed in a yellow dashed box labeled 'UnipolarOutputCircuitType'.</p>
type	<a href="#">UnipolarOutputCircuitType</a> , <a href="#">UnipolarOutputSinkDriverType</a> , <a href="#">UnipolarOutputSourceDriverType</a> , <a href="#">JEP30-D10:EmptyType</a> .

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”.

**4.5.2.1.6.2 Unipolar (cont'd)****5. Open-source output,**

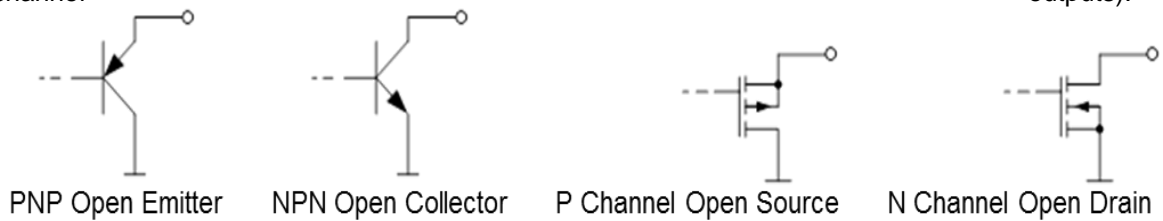
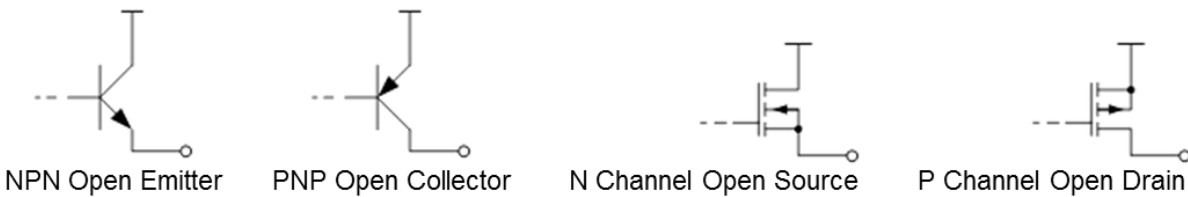
- a. An open-circuit output whose internal connection is to the source of a field-effect transistor. Ref. JESD99 “open-source output”.

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-outputs).

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

## 4.5.2.1.6.3. Programmable

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/OutputCircuit/Programmable</a>
diagram	
type	<a href="#">ProgrammableOutputCircuitType</a> , <a href="#">OutputCircuitSourcePropertyType</a> .

## 4.5.2.1.7. Reference

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/Properties-Array/Properties/Reference</a>
diagram	
type	<a href="#">SignalReferenceType</a> .

#### 4.5.2.1.7 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.

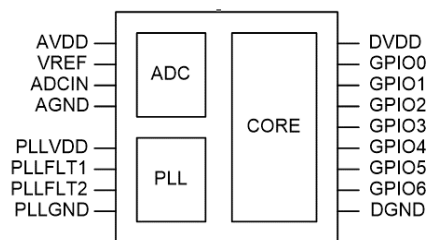


Figure 26 — Signal Reference

```

<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>

```

**4.5.2.1.7 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>
    </ElectricalMap>
  </ElectricalMap-Array>

```

**4.5.2.1.7 Reference (cont'd)**

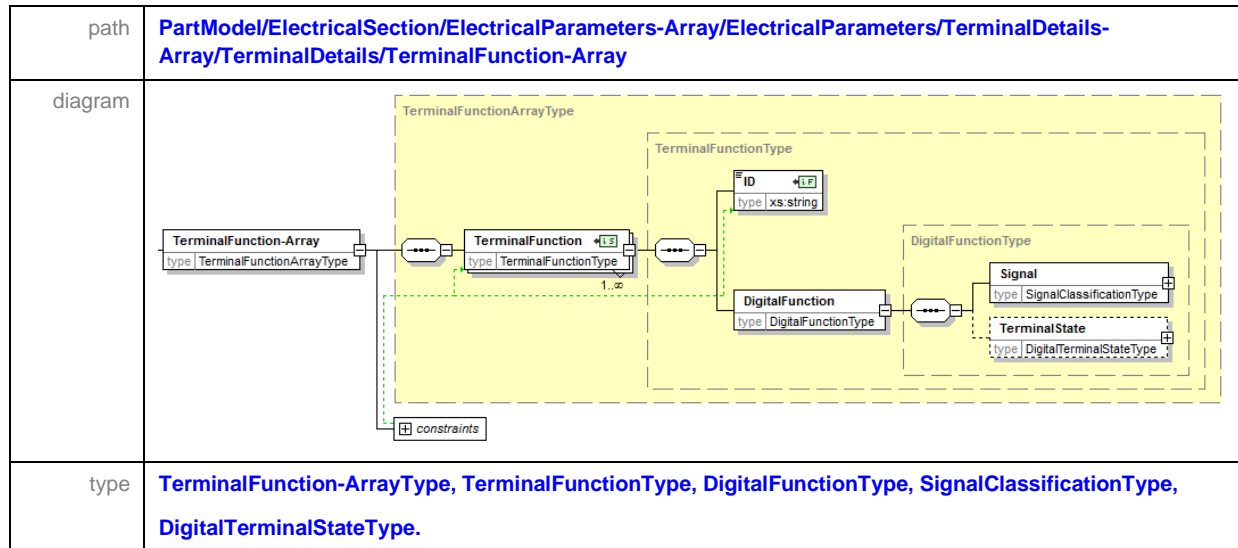
```

    < 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
PLLVD D -->
    <Reference>
      <GroundReference-TerminalName>PLLGND</GroundReference-TerminalName>
      <PositiveReference-TerminalName>PLLVD D</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>

```



#### 4.5.2.2. Terminal Function – Array



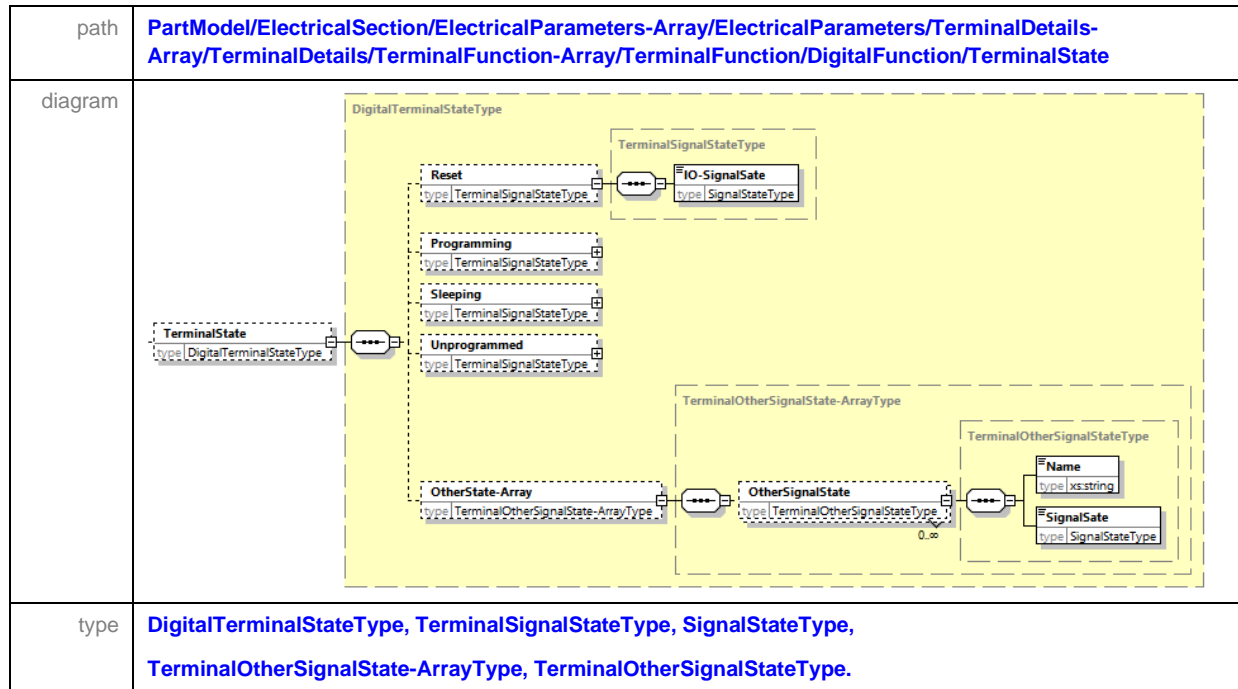
This section captures additional information about digital terminals, basically the classification of the digital signal and the logic state of the terminal necessary to perform various functions for certain types of devices.

4.5.2.2.1. Digital Function – Signal

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalFunction-Array/TerminalFunction/DigitalFunction/Signal
diagram	
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).

#### 4.5.2.2.2. Digital Function – Terminal State



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 value of the state are as follows:-

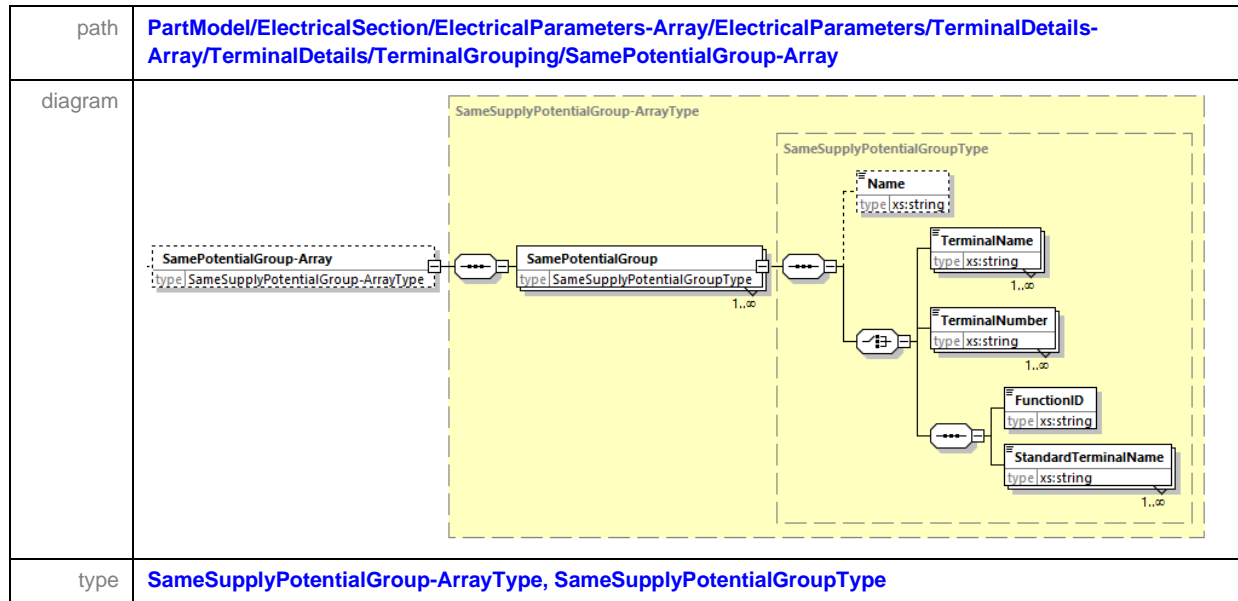
1. *0*,
2. *0Pullup*,
3. *1*,
4. *1Pullup*,
5. *HighImpedanceState*,
6. *LowState*,
7. *HighState*,
8. *Input*,
9. *Output*,

### 4.5.2.3. Terminal Grouping – Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping
diagram	<p>The diagram illustrates the structure of the <b>TerminalGroupingType</b> and its associated array types and types. The <b>TerminalGroupingType</b> is a base class that contains several array types, each of which is associated with a specific type. The array types are:</p> <ul style="list-style-type: none"> <li><b>SameSupplyPotentialGroup-Array</b>: Associated with <b>SameSupplyPotentialGroupType</b> (1..∞).</li> <li><b>TerminalSwap-Array</b>: Associated with <b>TerminalSwapType</b> (1..∞).</li> <li><b>FunctionSwap-Array</b>: Associated with <b>FunctionSwapType</b> (1..∞).</li> <li><b>InternalElectricalConnection-Array</b>: Associated with <b>InternalElectricalConnectionType</b> (1..∞).</li> <li><b>DifferentialPair-Array</b>: Associated with <b>DifferentialPairType</b> (1..∞). It includes a <b>constraints</b> box.</li> <li><b>Logical-Group-Array</b>: Associated with <b>Logical-GroupType</b> (1..∞). It includes a <b>constraints</b> box.</li> <li><b>Terminal-to-TerminalSignalPath-Array</b>: Associated with <b>Terminal-to-TerminalSignalPathType</b> (1..∞). It includes a <b>SignalPathCondition-ArrayType</b> (0..∞) and a <b>SignalPathCondition</b> box.</li> </ul> <p>The <b>TerminalGroupingType</b> is also associated with a <b>TerminalGrouping</b> type (1..∞).</p>
type	TerminalGroupingType, SameSupplyPotentialGroup-ArrayType, SameSupplyPotentialGroupType, TerminalSwap-ArrayType, TerminalSwapType, FunctionSwap-ArrayType, FunctionSwapType, InternalElectricalConnection-ArrayType, InternalElectricalConnectionType, DifferentialPair-ArrayType, DifferentialPairType, LogicalGroup-ArrayType, LogicalGroupType, Terminal-to-TerminalSignalPath-ArrayType, SignalPathCondition-ArrayType, SignalPathCondition, Terminal-to-TerminalSignalPathType,

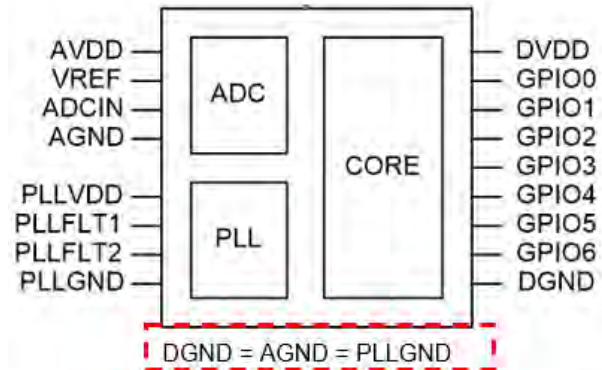
This section captures the data about various terminal groups based on their similarities.

#### 4.5.2.3.1. Same Potential Group - Array



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

```
<SamePotentialGroup>
  <TerminalName>DGND</TerminalName>
  <TerminalName>AGND</TerminalName>
  <TerminalName>PLLGN</TerminalName>
</SamePotentialGroup>
```



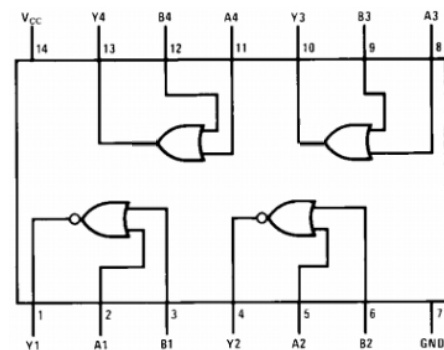
**Figure 27 – Same potential**

**4.5.2.3.2. Terminal Swap - Array**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/TerminalSwap-Array</a>
diagram	<pre> classDiagram     class TerminalSwapArray {         TerminalSwap TerminalSwapType 1..∞     }     class TerminalSwapType {         TerminalName xs:string 1..∞         TerminalNumber xs:string 1..∞         FunctionID xs:string         StandardTerminalName xs:string 1..∞     }     TerminalSwapArray "1" -- "1..∞" TerminalSwapType     TerminalSwapType "1" -- "1..∞" TerminalName     TerminalSwapType "1" -- "1..∞" TerminalNumber     TerminalSwapType "1" -- "1" FunctionID     TerminalSwapType "1" -- "1..∞" StandardTerminalName </pre>
type	<a href="#">TerminalSwap-ArrayType</a> , <a href="#">TerminalSwapType</a> .

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>

```

#### 4.5.2.3.2 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.

#### 4.5.2.3.3. Function Swap - Array

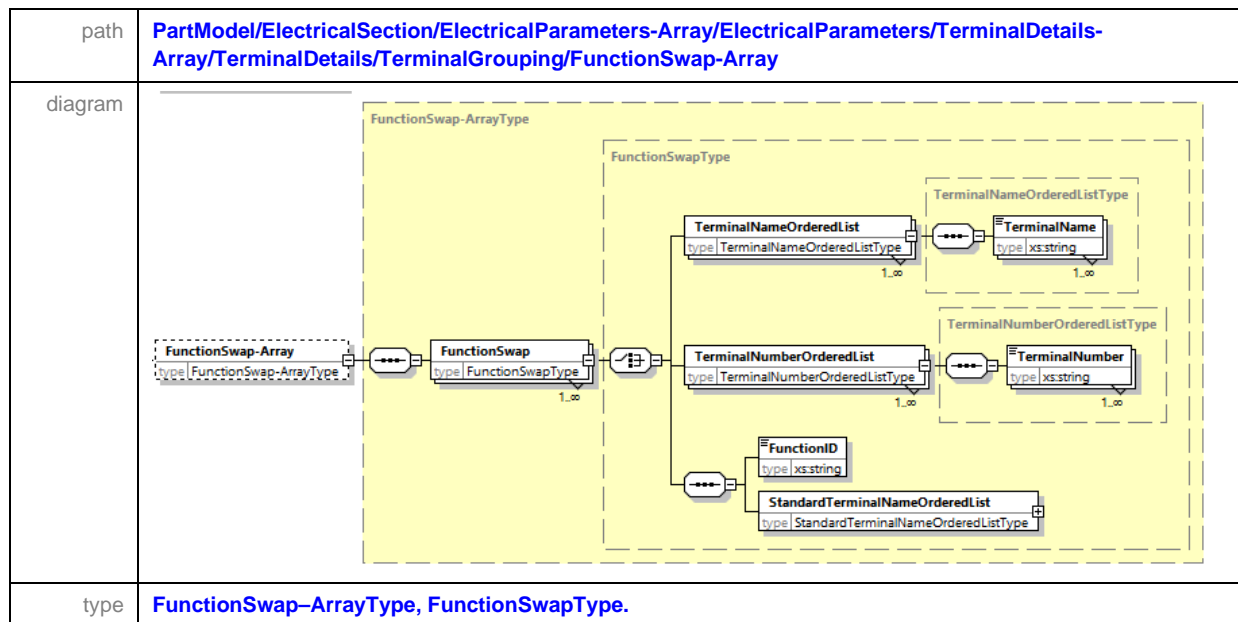


Figure 28 — Sample Mixed Gate Device gave an example of four gates that could be interchangeable, as follows:

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.

**4.5.2.3.3 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>

```

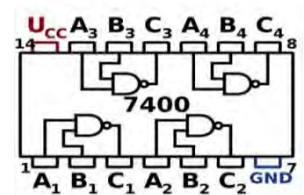
Alternatively, terminal name could have been used as follows

```

<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.



**Figure 29 – Sample NAND Gate Device**

```

<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>

```



#### 4.5.2.3.4. Internal Electrical Connection - Array

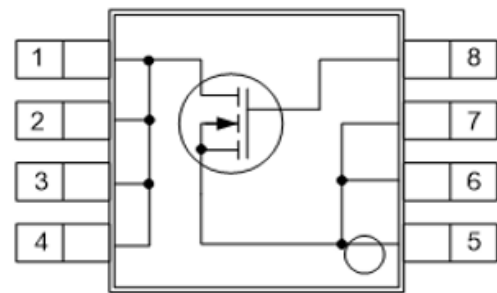
path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/InternalElectricalConnection-Array</a>
diagram	<pre> classDiagram     class InternalElectricalConnectionArray {         InternalElectricalConnection 1..∞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

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**

#### 4.5.2.3.5. Differential Pair - Array

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/DifferentialPair-Array</a>
diagram	
type	<a href="#">DifferentialPair-ArrayType</a> , <a href="#">DifferentialPairType</a> .

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.

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).

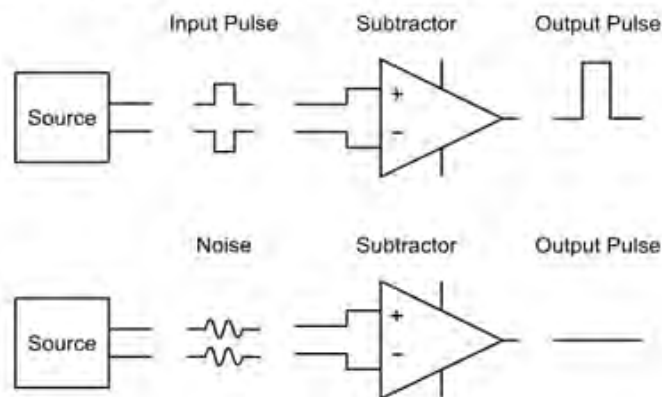


Figure 31 — System with Differential Receiver

#### 4.5.2.3.5 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*.

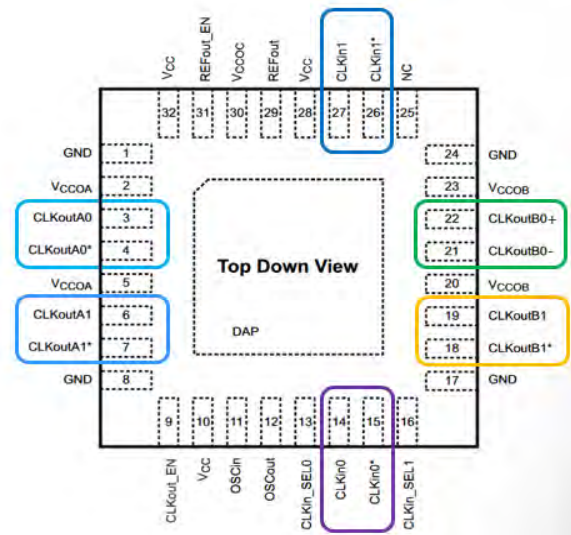


Figure 32 — Differential Pair Device

```

<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>

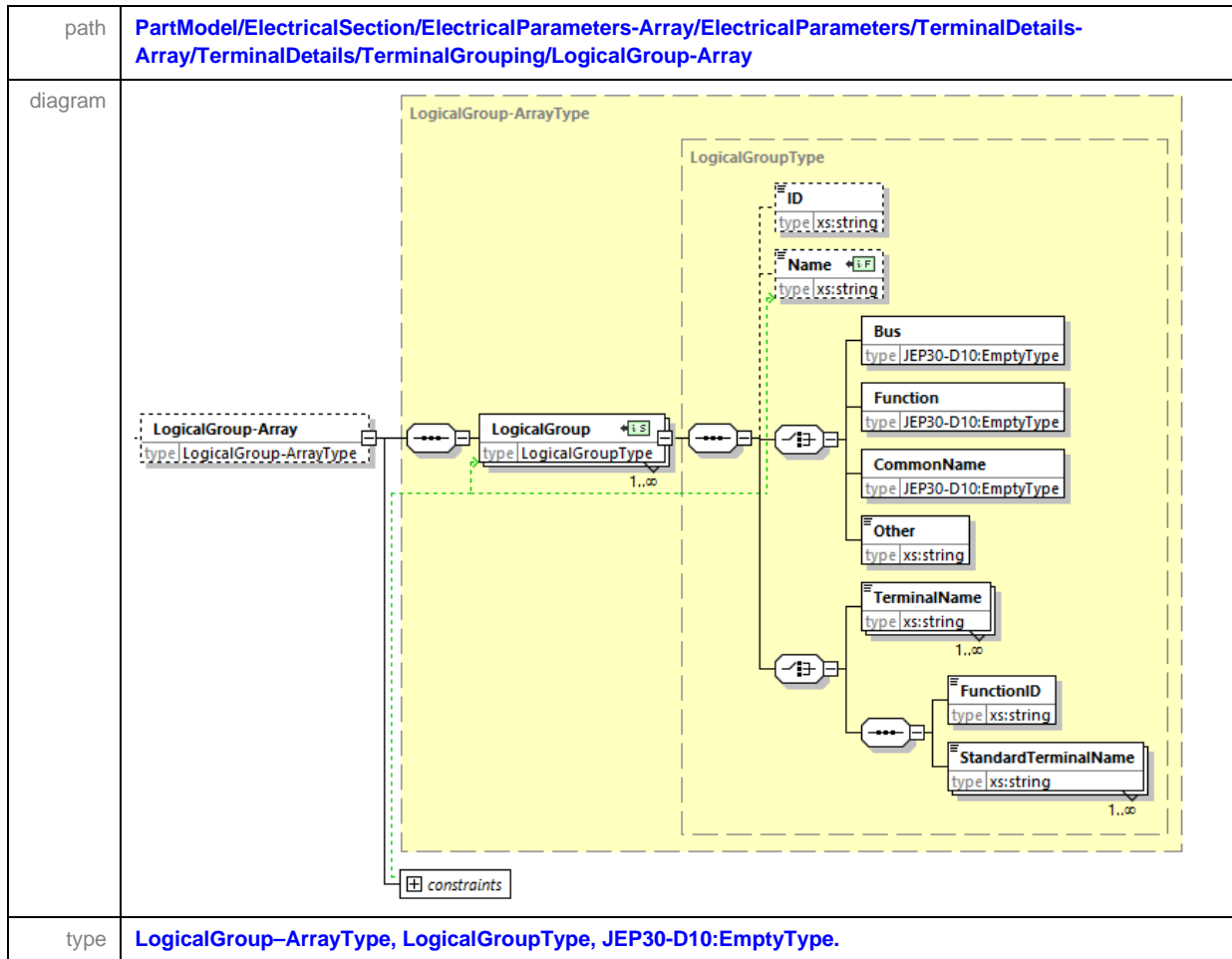
```

#### 4.5.2.3.5 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>
```

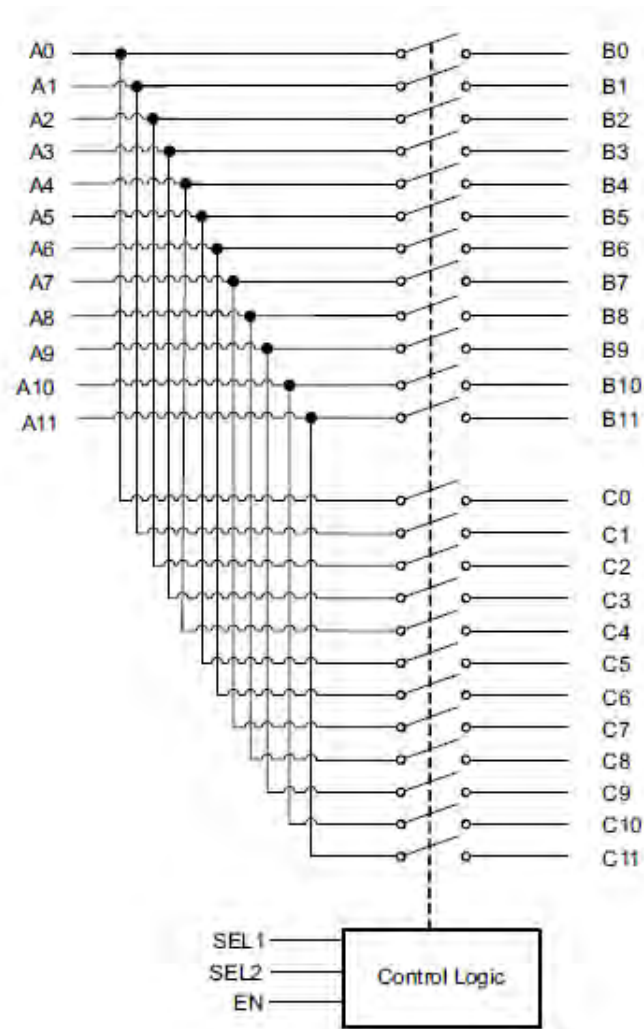
#### 4.5.2.3.6. Logical Group - Array



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.

**4.5.2.3.6 Logical Group - Array (cont'd)**

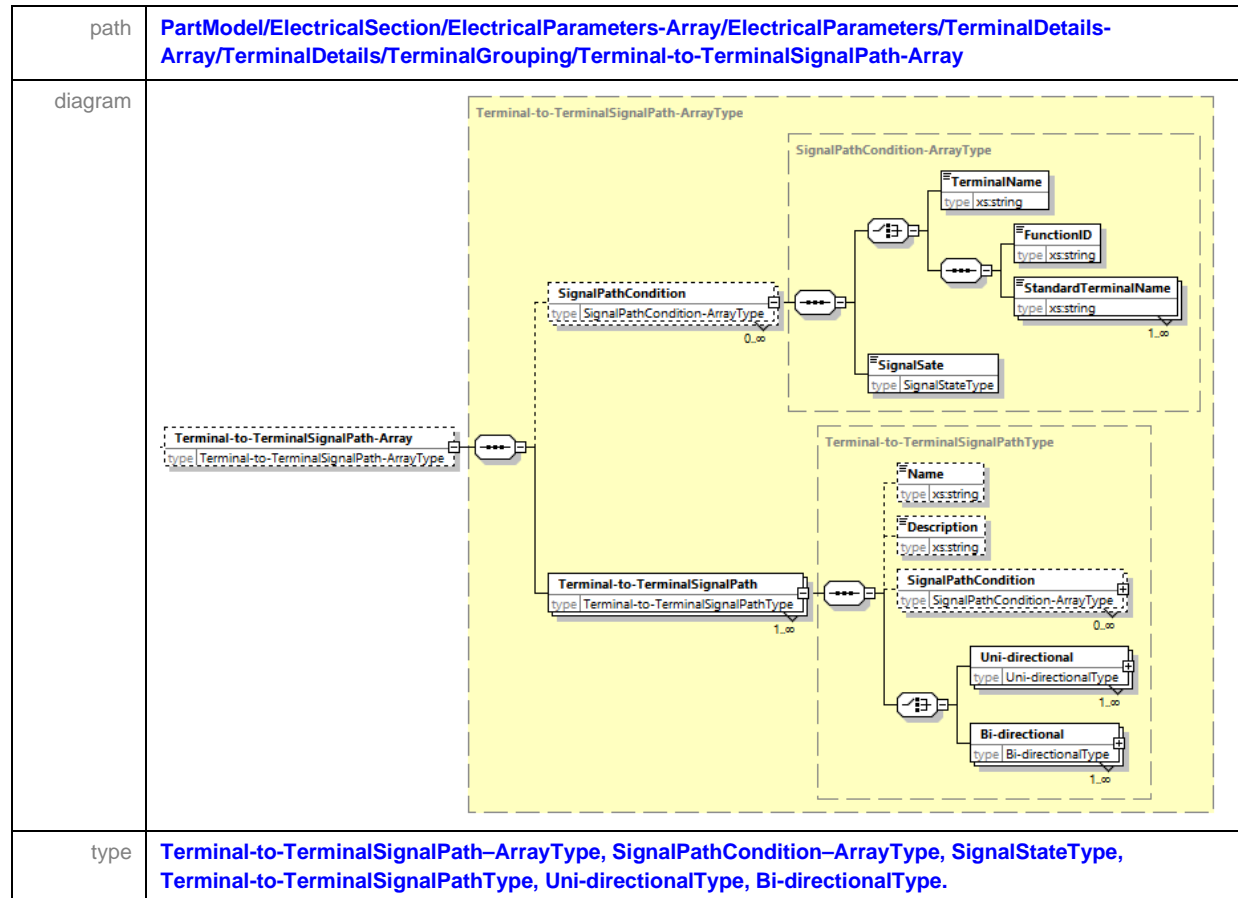
**Figure 33 — Function Block Diagram of a 12-bit Bus Switching Device**

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.

#### 4.5.2.3.6 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>
```

**4.5.2.3.7. Terminal-to-Terminal Signal Path – Array**

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.



#### 4.5.2.3.7.1. Uni-directional Terminal-to-Terminal Signal Path

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/Terminal-to-TerminalSignalPath-Array/Terminal-to-TerminalSignalPath/Uni-directional</a>
diagram	
type	<a href="#">Terminal-to-TerminalSignalPath-ArrayType</a> , <a href="#">SignalPathCondition-ArrayType</a> , <a href="#">SignalStateType</a> , <a href="#">Terminal-to-TerminalSignalPathType</a> , <a href="#">Uni-directionalType</a> , <a href="#">Bi-directionalType</a> .

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

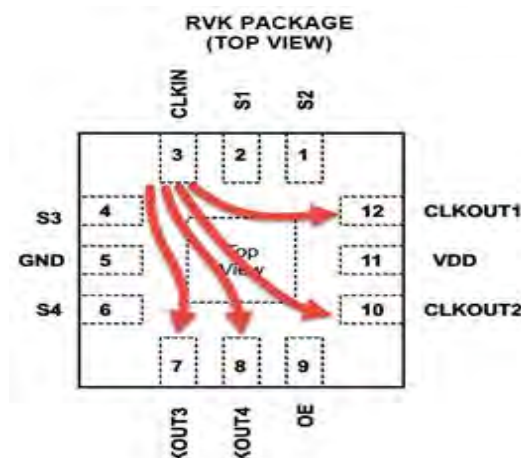


Figure 34 — Clock Buffer

**4.5.2.3.7.1 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>

```

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

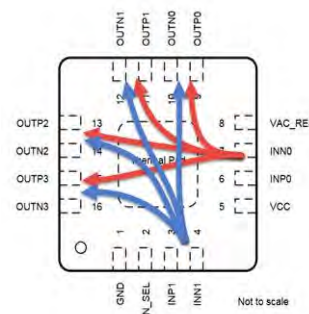
In this differential multiplexer example, the Differential Pair *Terminal-to-TerminalSignalPath* map could be captured by using the *DifferentialPair/Name*. This would make the data representation more condense.

```

<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**





**4.5.2.3.7.2. BI-directional Terminal-to-Terminal Signal Path**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/TerminalGrouping/Terminal-to-TerminalSignalPath-Array/Terminal-to-TerminalSignalPath/Bi-directional</a>
diagram	<p>The diagram illustrates the Bi-directional Terminal-to-Terminal Signal Path. It shows a 'Bi-directional' block (type Bi-directionalType, multiplicity 1..∞) connected to a 'Bi-directionalType' container. Inside this container, four bus symbols connect to various terminal details: 1) TerminalName1 and TerminalName2 (type xs:string); 2) DifferentialPairName1 and DifferentialPairName2 (type xs:string); 3) Logical-GroupName1 and Logical-GroupName2 (type xs:string); 4) FunctionID (type xs:string), StandardTerminalName1, and StandardTerminalName2 (type xs:string).</p>
type	<a href="#">Terminal-to-TerminalSignalPath-ArrayType</a> , <a href="#">SignalPathCondition-ArrayType</a> , <a href="#">SignalStateType</a> , <a href="#">Terminal-to-TerminalSignalPathType</a> , <a href="#">Uni-directionalType</a> , <a href="#">Bi-directionalType</a> .

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.

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

Table 2 — 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>

```

**4.5.2.3.7.2 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>

```

#### 4.5.2.4. External Connection - Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/ExternalConnection-Array	
diagram		
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.

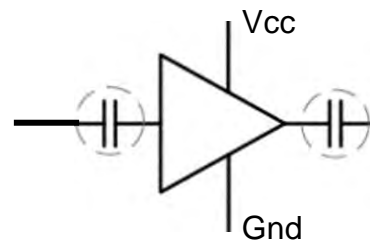


Figure 36 — Circuit with DC-Blocking

*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.

4.5.2.4.1. Decoupling

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/ExternalConnection-Array/ExternalConnection/Decoupling
diagram	
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.

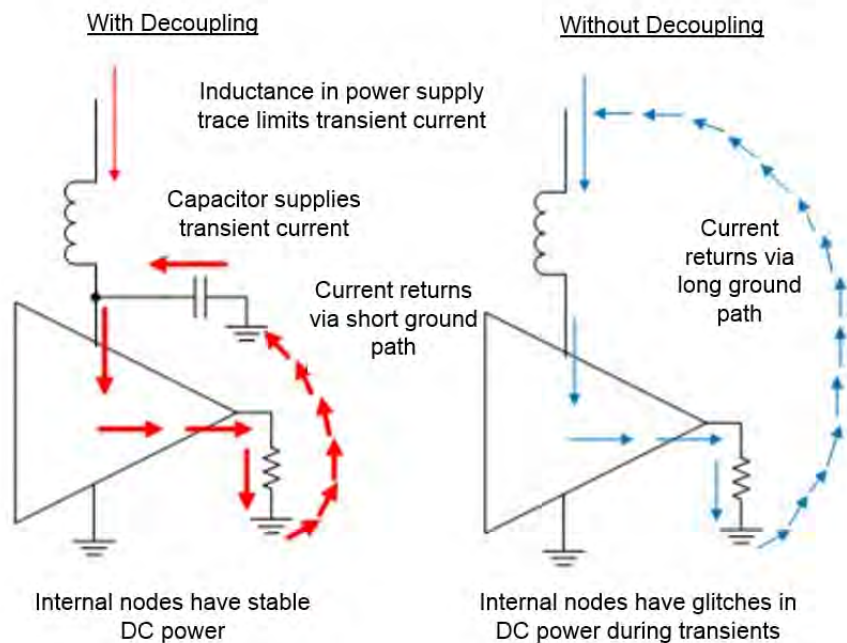


Figure 37 — Decoupling Circuit



#### 4.5.2.4.2. Pullup to Power

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/TerminalDetails-Array/TerminalDetails/ExternalConnection-Array/ExternalConnection/Pullup</a>
diagram	
type	<a href="#">Pull-up-to-PowerType</a> , <a href="#">Pull-up-to-PowerRecommendationType</a> , <a href="#">ResistanceType</a> , <a href="#">ResistanceUOMType</a> , <a href="#">PowerType</a> , <a href="#">JEP30-D10:PowerUOMType</a> , <a href="#">CurrentType</a> , <a href="#">JEP30-D10:CurrentUOMType</a> .

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*.

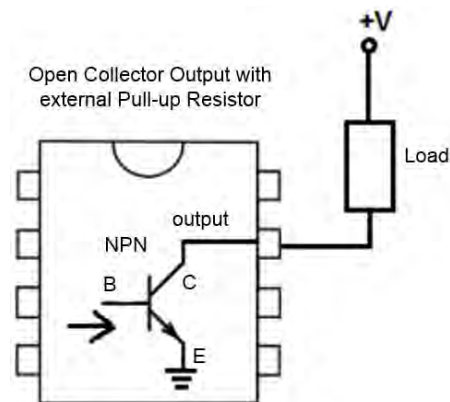


Figure 38 — Pull-up Resistor to Power

4.5.2.4.3. Pulldown to Ground

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

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.

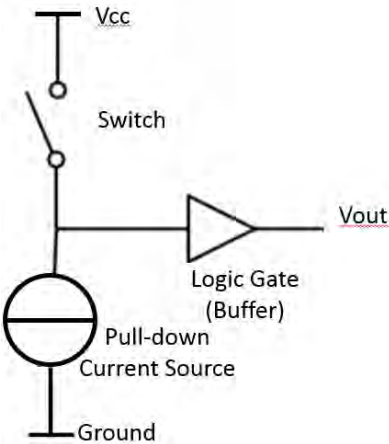


Figure 39 — Pull-down Current Source to Ground

### 4.5.3. Function Group - Array

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array</b>
diagram	<pre> classDiagram     class FunctionGroup-Array {         type FunctionGroup-ArrayType     }     class SuperInterface-Array {         type SuperInterface-ArrayType     }     class FunctionType {         ID xstring         Amplifier AmplifierFunctionType         Audio AudioFunctionType         Capacitor CapacitorFunctionType         Diode DiodeFunctionType         Filter FilterFunctionType         FrequencySource FrequencySourceFunctionType         Fuse FuseFunctionType         Inductor InductorFunctionType         Interface InterfaceFunctionType         NonLinear NonLinearFunctionType         Optoelectronic OptoelectronicFunctionType         Relay RelayFunctionType         Resistor ResistorFunctionType         RF RF-FunctionType         Source SourceFunctionType         Switch SwitchFunctionType         Thyristor ThyristorFunctionType         Transformer TransformerFunctionType         Transistor TransistorFunctionType         OtherStandard FunctionMap-to-StandardNameType     }     FunctionGroup-Array "1" -- "1..∞" FunctionType     SuperInterface-Array "1" -- "1" FunctionType     </pre> <p>The diagram illustrates the structure of the <b>FunctionGroup-Array</b>. It is an array of <b>FunctionGroup-ArrayType</b>. Each <b>FunctionGroup-ArrayType</b> contains a <b>SuperInterface-Array</b> of <b>SuperInterface-ArrayType</b>. Each <b>SuperInterface-ArrayType</b> contains a <b>FunctionType</b>. The <b>FunctionType</b> has an <b>ID</b> (xstring) and a list of function types: <b>Amplifier</b>, <b>Audio</b>, <b>Capacitor</b>, <b>Diode</b>, <b>Filter</b>, <b>FrequencySource</b>, <b>Fuse</b>, <b>Inductor</b>, <b>Interface</b>, <b>NonLinear</b>, <b>Optoelectronic</b>, <b>Relay</b>, <b>Resistor</b>, <b>RF</b>, <b>Source</b>, <b>Switch</b>, <b>Thyristor</b>, <b>Transformer</b>, <b>Transistor</b>, and <b>OtherStandard</b>. The <b>OtherStandard</b> type is a <b>FunctionMap-to-StandardNameType</b>. Constraints are indicated at the bottom of the diagram.</p>
type	<b>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.</b>

### 4.5.3 Function Group – Array (cont'd)

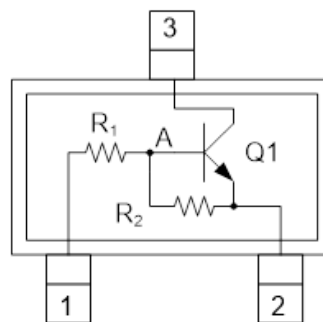
The [FunctionGroup-Array](#) section captures some of the basic internal functions of the device. Simple discrete parts might consist of a single function, while other may have an array of such functions. Other more complex parts can have multiple different functions, while at the furthest extreme, some 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.

**4.5.3 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>
        </Resistor>
      </Function>
    </FunctionalGroup-Array>
  </Electrical>
</Electrical-Array>

```

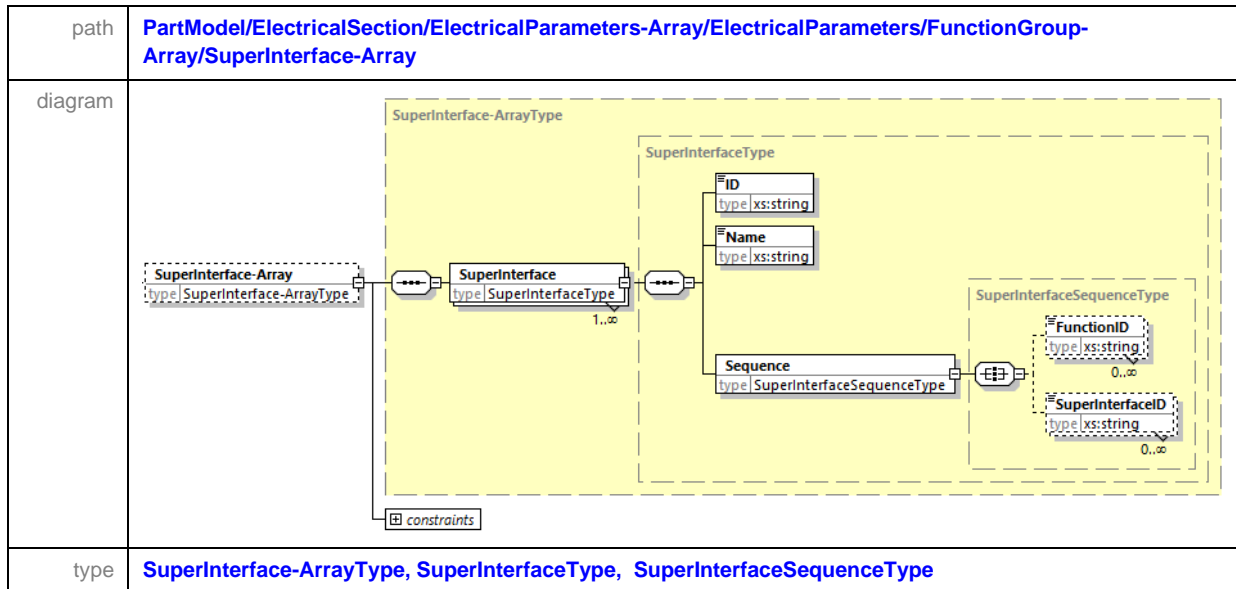
**4.5.3 Function Group – Array (cont'd)**

```

        </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>
        <Symbol>R</Symbol>
        <SymbolDescription>Resistance Value</SymbolDescription>
        <Values>
          <Standard>
            <Nominal>10</Nominal>
          </Standard>
        </Values>
        <Units>
          <Resistance>KOhm</Resistance>
        </Units>
      </Parameter>
    </ParameterSet>
  </ElectricalSpecification>
</ElectricalSpecification-Array>

```

#### 4.5.3.1. Super Interface - Array



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.

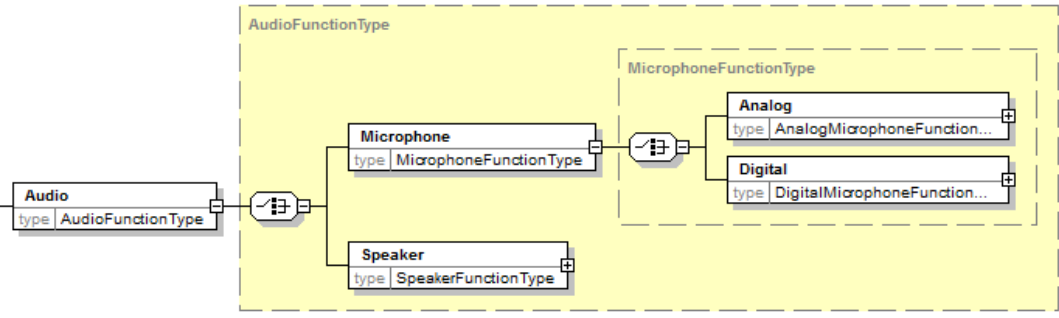
A typical example of how this may be used would be in a multi-channel device, in which each channel was made up a set of Interfaces and functions. Let's assume the following example. A device has 2 or more Banks, where each Bank contains 3 channels A to C. Each channel consists of a DDR6 interface, and a PCIe Interface. Channels A through C are swappable, whereas the Banks are not swappable, 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>Channel1</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>

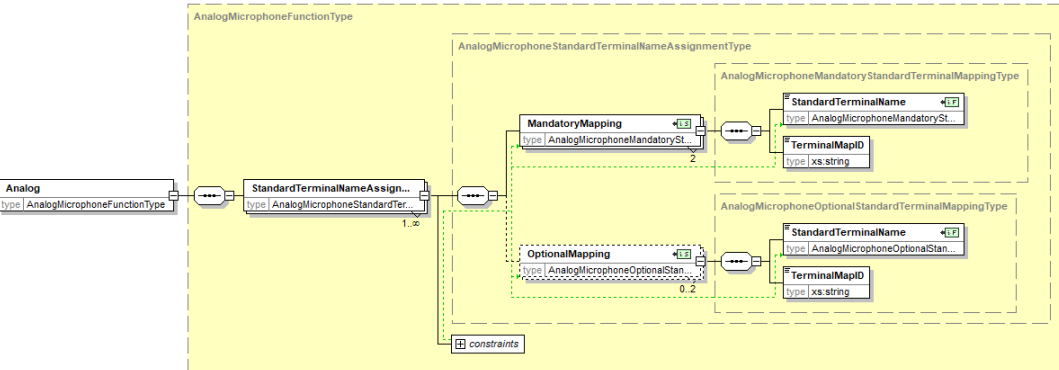
```

4.5.3.2. Audio

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Audio
diagram	
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.

4.5.3.2.1. Analog Microphone

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Audio/Microphone			
diagram				
type	AnalogMicrophoneFunctionType, AnalogMicrophoneStandardTerminalNameAssignmentType, AnalogMicrophoneMandatoryStandardTerminalMappingType, AnalogMicrophoneMandatoryStandardTerminalNameType, AnalogMicrophoneOptionalStandardTerminalMappingType, AnalogMicrophoneOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Output	2. Ground		
	OptionalMapping/StandardTerminalName			
	3. Case	4. Power		



#### 4.5.3.2.2. Digital Microphone

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Audio/Microphone		
diagram			
type	DigitalMicrophoneFunctionType, DigitalMicrophoneStandardTerminalNameAssignmentType, DigitalMicrophoneStandardTerminalMappingType, DigitalMicrophoneStandardTerminalNameType.		
list of enumerate values	Mapping/StandardTerminalName		
	1. Clock	2. Data	3. Ground
			4. Power

#### 4.5.3.2.3. Speaker

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Audio/Speaker		
diagram			
type	SpeakerFunctionType, SpeakerStandardTerminalNameAssignmentType, SpeakerMandatoryStandardTerminalMappingType, SpeakerOptionalStandardTerminalMappingType, SpeakerOptionalStandardTerminalNameType, SpeakerMandatoryStandardTerminalNameType		
list of enumerate values	MandatoryMapping/StandardTerminalName		
	1. Positive	2. Negative	
	OptionalMapping/StandardTerminalName		
	1. Case	2.	

4.5.3.3. Amplifier

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier
diagram	
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.

#### 4.5.3.3.1. Differential Input Differential Output

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier/DifferentialInput/DifferentialOutput			
diagram				
type	DifferentialInputDifferentialOutputAmplifierType, DifferentialInputDifferentialOutputAmplifierStandardTerminalNameAssignmentType, DifferentialInputDifferentialOutputAmplifierMandatoryStandardTerminalMappingType, DifferentialInputDifferentialOutputAmplifierMandatoryStandardTerminalNameType, DifferentialInputDifferentialOutputAmplifierOptionalStandardTerminalMappingType, DifferentialInputDifferentialOutputAmplifierOptionalStandardTerminalNameType.			
list of enumerate values	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

4.5.3.3.2. Differential Input Single Ended Output

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier/DifferentialInput/SingleEndedOutput			
diagram	<p>The diagram illustrates the structure of a Differential Input Single Ended Output Amplifier. It starts with a <b>SingleEndedOutput</b> class (type <code>DifferentialInputSingleEndedOut</code>) which is associated with a <b>StandardTerminalNameAssignment</b> class (type <code>DifferentialInputSingleEndedOut</code>, multiplicity <code>1..*</code>). The <b>StandardTerminalNameAssignment</b> class is further associated with <b>MandatoryMapping</b> (type <code>DifferentialInputSingleEndedOut</code>, multiplicity <code>0..1</code>) and <b>OptionalMapping</b> (type <code>DifferentialInputSingleEndedOut</code>, multiplicity <code>0..4</code>). Both <b>MandatoryMapping</b> and <b>OptionalMapping</b> are associated with <b>StandardTerminalName</b> (type <code>DifferentialInputSingleEndedOut</code>, multiplicity <code>0..1</code>) and <b>TerminalMapID</b> (type <code>AS.STRING</code>, multiplicity <code>1</code>). A <b>constants</b> class is also present.</p>			
type	<b>DifferentialInputSingleEndedOutputAmplifierType,</b> <b>DifferentialInputSingleEndedOutputAmplifierStandardTerminalNameAssignmentType,</b> <b>DifferentialInputSingleEndedOutputAmplifierMandatoryStandardTerminalMappingType,</b> <b>DifferentialInputSingleEndedOutputAmplifierOptionalStandardTerminalMappingType,</b> <b>DifferentialInputSingleEndedOutputAmplifierMandatoryStandardTerminalNameType,</b> <b>DifferentialInputSingleEndedOutputAmplifierOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Positive Input</b>	<b>2. Negative Input</b>	<b>3. Output</b>	<b>4. Negative Rail</b>
	<b>5. Positive Rail</b>	<b>6. Common</b>		
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. Gain Resistor Terminal 1</b>	<b>2. Gain Resistor Terminal 2</b>	<b>3. Offset Node 1</b>	<b>4. Offset Node 2</b>

### 4.5.3.3.3. Single Ended Input Differential Output

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier/SingleEndedInput/DifferentialOutput			
diagram				
type	SingleEndedInputDifferentialOutputAmplifierType, SingleEndedInputDifferentialOutputAmplifierStandardTerminalNameAssignmentType, SingleEndedInputDifferentialOutputAmplifierMandatoryStandardTerminalMappingType, SingleEndedInputDifferentialOutputAmplifierMandatoryStandardTerminalNameType, SingleEndedInputDifferentialOutputAmplifierOptionalStandardTerminalMappingType, SingleEndedInputDifferentialOutputAmplifierOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Input	2. Positive Output	3. Negative 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

4.5.3.3.4. Single Ended Input Single Ended Output

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Amplifier/SingleEndedInput/SingleEndedOutput			
diagram				
type	SingleEndedInputDifferentialOutputAmplifierType, SingleEndedInputSingleEndedOutputAmplifierStandardTerminalNameAssignmentType, SingleEndedInputSingleEndedOutputAmplifierMandatoryStandardTerminalMappingType, SingleEndedInputSingleEndedOutputAmplifierMandatoryStandardTerminalNameType, SingleEndedInputSingleEndedOutputAmplifierOptionalStandardTerminalMappingType, SingleEndedInputSingleEndedOutputAmplifierOptionalStandardTerminalNameType.			
list of enumerate values	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

#### 4.5.3.4. Capacitor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor
diagram	<pre> classDiagram     class Capacitor {         type CapacitorFunctionType     }     class CapacitorFunctionType {         &lt;&lt;abstract&gt;&gt;     }     class FixedNonPolarized {         type FixedNonPolarizedCapacitorType     }     class FixedPolarized {         type FixedPolarizedCapacitorType     }     class DifferentialNonPolarized {         type DifferentialNonPolarizedCapacitorType     }     class FeedThrough {         type FeedThroughCapacitorType     }     class VariableNonPolarized {         type VariableNonPolarizedCapacitorType     }     Capacitor --&gt; CapacitorFunctionType     CapacitorFunctionType &lt; -- FixedNonPolarized     CapacitorFunctionType &lt; -- FixedPolarized     CapacitorFunctionType &lt; -- DifferentialNonPolarized     CapacitorFunctionType &lt; -- FeedThrough     CapacitorFunctionType &lt; -- VariableNonPolarized     </pre>
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.

#### 4.5.3.4.1. Fixed Non Polarized

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/FixedNonPolarized			
diagram	<pre> classDiagram     class FixedNonPolarizedCapacitorType     class StandardTerminalNameAssignmentType {         type FixedNonPolarizedCapacitorSta...     }     class Mapping {         type FixedNonPolarizedCapacitorSta...     }     class StandardTerminalName {         type FixedNonPolarizedCapacitorSta...     }     class TerminalMapID {         type xs:string     }     FixedNonPolarizedCapacitorType "1" -- "∞" StandardTerminalNameAssignmentType     StandardTerminalNameAssignmentType "1" -- "2" Mapping     Mapping "1" -- "2" StandardTerminalName     StandardTerminalName "1" -- "1" TerminalMapID     </pre>			
type	FixedNonPolarizedCapacitorType, FixedNonPolarizedCapacitorStandardTerminalNameAssignmentType, FixedNonPolarizedCapacitorStandardTerminalMappingType, FixedNonPolarizedCapacitorStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. Terminal 1	2. Terminal 2		

**4.5.3.4.2. Fixed Polarized**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/FixedPolarized</b>			
diagram				
type	<b>FixedPolarizedCapacitorType, FixedPolarizedCapacitorStandardTerminalNameAssignmentType, FixedPolarizedCapacitorStandardTerminalMappingType, FixedPolarizedCapacitorStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Positive</b>	<b>2. Negative</b>		

**4.5.3.4.3. Differential Non Polarized**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/DifferentialNonPolarized</b>			
diagram				
type	<b>DifferentialNonPolarizedCapacitorType, DifferentialNonPolarizedCapacitorStandardTerminalNameAssignmentType, DifferentialNonPolarizedCapacitorStandardTerminalMappingType, DifferentialNonPolarizedCapacitorStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Terminal 1</b>	<b>2. Terminal 2</b>	<b>3. Common</b>	



#### 4.5.3.4.4. Feed Through

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/FeedThrough</b>			
diagram				
type	<b>FeedThroughCapacitorType, FeedThroughCapacitorStandardTerminalNameAssignmentType, FeedThroughCapacitorStandardTerminalMappingType, FeedThroughCapacitorStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Terminal 1</b>	<b>2. Terminal 2</b>	<b>3. Common</b>	

#### 4.5.3.4.5. Variable Non Polarized

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Capacitor/VariableNonPolarized</b>			
diagram				
type	<b>VariableNonPolarizedCapacitorType, VariableNonPolarizedCapacitorStandardTerminalNameAssignmentType, VariableNonPolarizedCapacitorStandardTerminalMappingType, VariableNonPolarizedCapacitorStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Terminal 1</b>	<b>2. Terminal 2</b>		

**4.5.3.5. Diode**

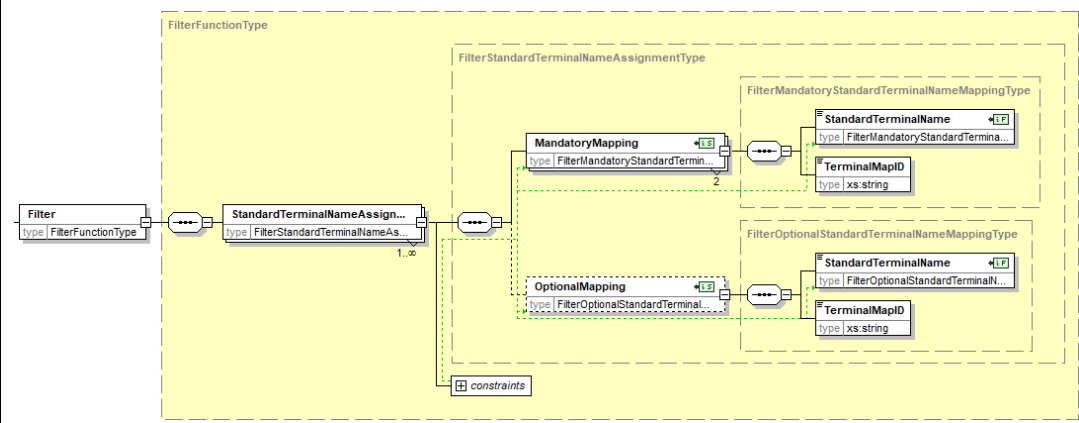
path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode</b>
diagram	
type	<b>DiodeFunctionType, CurrentRegulatorDiodeFunctionType, ESD-DiodeFunctionType, LED-DiodeFunctionType, MicrowaveDiodeFunctionType, PIN-DiodeFunctionType, RectifierDiodeFunctionType, SchottkyDiodeFunctionType, SignalDiodeFunctionType, SiliconCarbideDiodeFunctionType, TunnelDiodeFunctionType, Uni-tunnelDiodeFunctionType, VaractorDiodeFunctionType, VoltageRegulatorDiodeFunctionType, ZenerDiodeFunctionType, DiodeStandardTerminalNameAssignmentType.</b>

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.

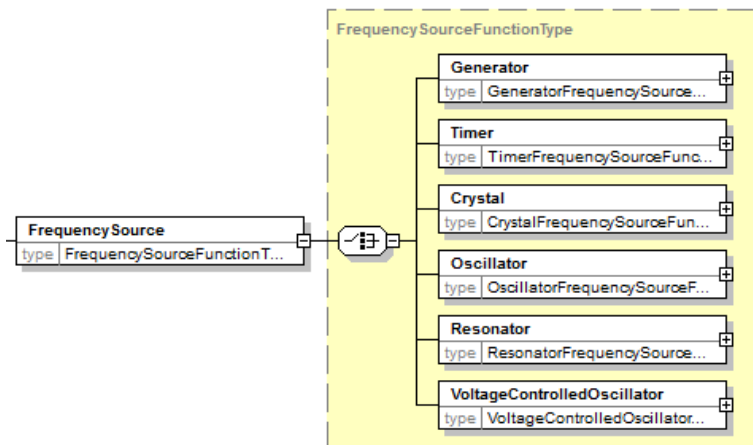
#### 4.5.3.5.1. Diode Standard Terminal Name Assignment Type

path	<ol style="list-style-type: none"> <li>1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/CurrentRegulator/StandardTerminalNameAssignment</li> <li>2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/ESD/StandardTerminalNameAssignment/Mapping/StandardTerminalName</li> <li>3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/LED/StandardTerminalNameAssignment/Mapping/StandardTerminalName</li> <li>4. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Microwave/StandardTerminalNameAssignment</li> <li>5. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/PIN /StandardTerminalNameAssignment</li> <li>6. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Rectifier/StandardTerminalNameAssignment</li> <li>7. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Schottky/StandardTerminalNameAssignment</li> <li>8. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Signal/StandardTerminalNameAssignment</li> <li>9. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/SiliconCarbide/StandardTerminalNameAssignment</li> <li>10. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Tunnel/StandardTerminalNameAssignment</li> <li>11. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Uni-tunnel/StandardTerminalNameAssignment</li> <li>12. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Varactor/StandardTerminalNameAssignment</li> <li>13. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/VoltageRegulator/StandardTerminalNameAssignment</li> <li>14. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Diode/Zener/StandardTerminalNameAssignment</li> </ol>			
diagram				
type	<b>DiodeStandardTerminalNameAssignmentType, DiodeStandardTerminalMappingType, DiodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		

4.5.3.6. Filter

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Filter		
diagram			
type	FilterFunctionType, FilterStandardTerminalNameAssignmentType, FilterStandardTerminalMappingType, FilterStandardTerminalNameType.		
list of enumerate values	MandatoryMapping/StandardTerminalName		
	1. Terminal 1	2. Terminal 2	
	OptionalMapping/StandardTerminalName		
	1. Ground		

4.5.3.7. Frequency Source

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource		
diagram			
type	FrequencySourceFunctionType, GeneratorFrequencySourceFunctionType, TimerFrequencySourceFunctionType, CrystalFrequencySourceFunctionType, OscillatorFrequencySourceFunctionType, ResonatorFrequencySourceFunctionType, VoltageControlledOscillatorFrequencySourceFunctionType.		

#### 4.5.3.7. Frequency Source (cont'd)

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

##### 4.5.3.7.1. Generator

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Generator</b>		
diagram			
type	<b>GeneratorFrequencySourceFunctionType,</b> <b>GeneratorFrequencySourceStandardTerminalNameAssignmentType,</b> <b>GeneratorFrequencySourceMandatoryStandardTerminalMappingType,</b> <b>GeneratorFrequencySourceMandatoryStandardTerminalNameType,</b> <b>GeneratorFrequencySourceOptionalStandardTerminalMappingType,</b> <b>GeneratorFrequencySourceOptionalStandardTerminalNameType.</b>		
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>		
	<b>1. Output</b>	<b>2. Ground</b>	<b>3. Power</b>
	<b>OptionalMapping/StandardTerminalName</b>		
	<b>1. Enable</b>		

4.5.3.7.2. Timer

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Timer			
diagram				
type	TimerFrequencySourceFunctionType, TimerFrequencySourceStandardTerminalNameAssignmentType, TimerFrequencySourceMandatoryStandardTerminalMappingType, TimerFrequencySourceMandatoryStandardTerminalNameType, TimerFrequencySourceOptionalStandardTerminalMappingType, TimerFrequencySourceOptionalStandardTerminalNameType, TimerFrequencySourceOtherStandardTerminalMappingType, TimerFrequencySourceOtherStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Output	2. Trigger	3. Power	4. Ground
	OptionalMapping/StandardTerminalName			
	1. Enable	2. Reset		

4.5.3.7.3. Crystal

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Crystal			
diagram	<p>The diagram illustrates the structure of the <code>CrystalFrequencySourceFunctionType</code>. It is composed of several nested and associated components:</p> <ul style="list-style-type: none"><li><code>Crystal</code> (type <code>CrystalFrequencySourceFuncti...</code>) is associated with <code>StandardTerminalNameAssign...</code> (type <code>CrystalFrequencySourceStand...</code>) via a 1..* relationship.</li><li><code>StandardTerminalNameAssign...</code> is associated with <code>MandatoryMapping</code> (type <code>CrystalFrequencySourceMand...</code>) via a 1..* relationship.</li><li><code>MandatoryMapping</code> is associated with <code>StandardTerminalName</code> (type <code>CrystalFrequencySourceMand...</code>) via a 1..* relationship.</li><li><code>StandardTerminalName</code> is associated with <code>TerminalMapID</code> (type <code>xs:string</code>) via a 1..* relationship.</li><li><code>OptionalMapping</code> (type <code>CrystalFrequencySourceOpto...</code>) is associated with <code>StandardTerminalName</code> (type <code>CrystalFrequencySourceOption...</code>) via a 1..* relationship.</li><li><code>StandardTerminalName</code> is associated with <code>TerminalMapID</code> (type <code>xs:string</code>) via a 1..* relationship.</li><li>A <code>constraints</code> box is also present.</li></ul>			
type	<code>CrystalFrequencySourceFunctionType</code> , <code>CrystalFrequencySourceStandardTerminalNameAssignmentType</code> , <code>CrystalFrequencySourceMandatoryStandardTerminalMappingType</code> , <code>CrystalFrequencySourceMandatoryStandardTerminalNameType</code> , <code>CrystalFrequencySourceOptionalStandardTerminalMappingType</code> , <code>CrystalFrequencySourceOptionaStandardTerminalNameType</code> .			
list of enumerate values	<code>MandatoryMapping/StandardTerminalName</code>			
	1. Terminal 1	2. Terminal 2		
	<code>OptionalMapping/StandardTerminalName</code>			
	1. Case			

4.5.3.7.4. Oscillator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Oscillator			
diagram	<p>The diagram illustrates the UML structure for the Oscillator type. It shows a hierarchy of nested types: <b>OscillatorFrequencySourceFunctionType</b> contains <b>Oscillator</b> (type: OscillatorFrequencySourceFun...) and <b>StandardTerminalNameAssign...</b> (type: OscillatorFrequencySourceSta...). <b>StandardTerminalNameAssign...</b> contains <b>MandatoryMapping</b> (type: OscillatorFrequencySourceMa...) and <b>OptionalMapping</b> (type: OscillatorFrequencySourceOpt...). <b>MandatoryMapping</b> contains <b>StandardTerminalName</b> (type: OscillatorFrequencySourceMan...) and <b>TerminalMapID</b> (type: xs:string). <b>OptionalMapping</b> contains <b>StandardTerminalName</b> (type: OscillatorFrequencySourceOpt...) and <b>TerminalMapID</b> (type: xs:string). A <b>constraints</b> box is also present.</p>			
type	<b>OscillatorFrequencySourceFunctionType,</b> <b>OscillatorFrequencySourceStandardTerminalNameAssignmentType,</b> <b>OscillatorFrequencySourceMandatoryStandardTerminalMappingType,</b> <b>OscillatorFrequencySourceMandatoryStandardTerminalNameType,</b> <b>OscillatorFrequencySourceOptionalStandardTerminalMappingType,</b> <b>OscillatorFrequencySourceOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	1. Output	2. Ground	3. Power	
	<b>OptionalMapping/StandardTerminalName</b>			
	1. Enable			



4.5.3.7.5. Resonator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/FrequencySource/Resonator			
diagram				
type	ResonatorFrequencySourceFunctionType, ResonatorFrequencySourceStandardTerminalNameAssignmentType, ResonatorFrequencySourceMandatoryStandardTerminalMappingType, ResonatorFrequencySourceMandatoryStandardTerminalNameType, ResonatorFrequencySourceOptionalStandardTerminalMappingType, ResonatorFrequencySourceOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Terminal 1	2. Terminal 2		
	OptionalMapping/StandardTerminalName			
	1. Case			

**4.5.3.7.6. Voltage Controlled Oscillator**

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

4.5.3.8. Fuse

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Fuse			
diagram	<p>The diagram illustrates the structure of the Fuse class. It is composed of several nested types: FuseFunctionType, FuseStandardTerminalNameAssignmentType, FuseStandardTerminalMappingType, and FuseStandardTerminalNameType. The Fuse class (type FuseFunctionType) is associated with the StandardTerminalNameAssignment class (type FuseStandardTerminalNameAssignmentType) with a multiplicity of 1..∞. This is then associated with the Mapping class (type FuseStandardTerminalMappingType) with a multiplicity of 2. The Mapping class is associated with the StandardTerminalName class (type FuseStandardTerminalNameType) with a multiplicity of 1. The StandardTerminalName class has a TerminalMapID attribute of type xs:string. A dashed box labeled 'constraints' is shown below the Mapping class.</p>			
type	FuseType, FuseStandardTerminalNameAssignmentType, FuseStandardTerminalMappingType, FuseStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. Terminal 1	2. Terminal 2		

4.5.3.9. 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 "1" -- "1..*" StandardTerminalNameAssignment     Coupled "1" -- "1..*" StandardTerminalNameAssignment     Ferrite "1" -- "1..*" StandardTerminalNameAssignment     Variable "1" -- "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.

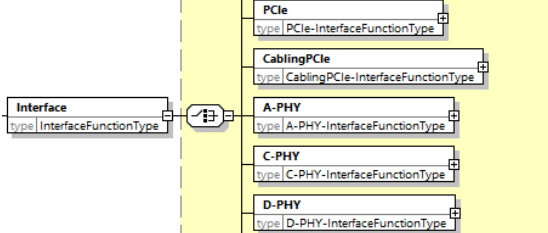
#### 4.5.3.9.1. Basic Inductor Standard Terminal Name Assignment

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor/Air</a></li> <li>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor/Ferrite</a></li> <li>3. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor/Variable</a></li> </ol>			
diagram				
type	<a href="#">BasicInductorStandardTerminalNameAssignmentType</a> , <a href="#">BasicInductorStandardTerminalMappingType</a> , <a href="#">BasicInductorStandardTerminalNameType</a> .			
list of enumerate values	<a href="#">Mapping/StandardTerminalName</a>			
	1. Terminal 1	2. Terminal 2		

#### 4.5.3.9.2. Coupled Inductor Standard Terminal Name Assignment

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Inductor/Coupled</a>			
diagram				
type	<a href="#">InductorFunctionType</a> , <a href="#">AirInductorFunctionType</a> , <a href="#">BasicInductorStandardTerminalNameAssignmentType</a> , <a href="#">CoupledInductorFunctionType</a> , <a href="#">CoupledInductorStandardTerminalNameAssignmentType</a> , <a href="#">FerriteInductorFunctionType</a> , <a href="#">VariableInductorFunctionType</a> .			
list of enumerate values	<a href="#">Mapping/StandardTerminalName</a>			
	1. Primary Coil Terminal 1	2. Primary Coil Terminal 2	3. Secondary Coil Terminal 1	4. Secondary Coil Terminal 2

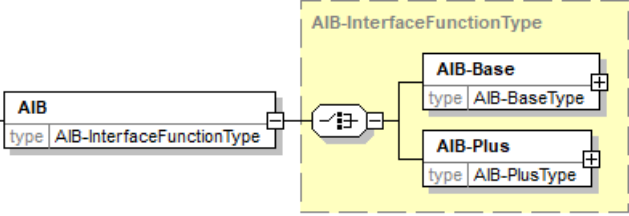
**4.5.3.10. Interface**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface</b>	
Sample diagram (See list of defined Interfaces below)		
Standard interface versus type	<b>InterfaceFunctionType,</b>	
	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	MI-InterfaceFunctionType

**4.5.3.10 Interface Function (cont'd)**

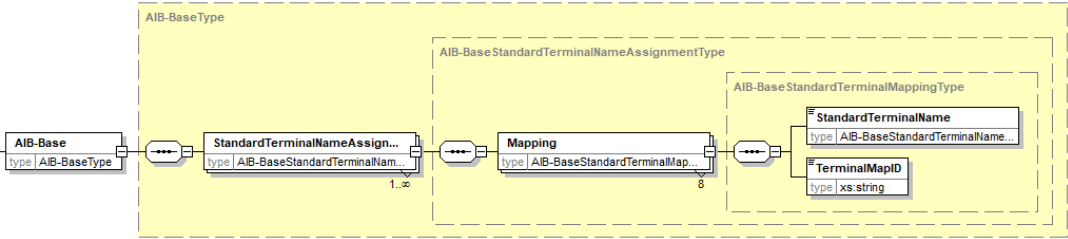
Standard interface versus type	28. OIF-CEI-04.0	OIF-CEI-04.0-InterfaceFunctionType
	29. PCIe	PCIe-InterfaceFunctionType
	30. CablingPCIe	CablingPCIe-InterfaceFunctionType
	31. A-PHY	A-PHY-InterfaceFunctionType
	32. BoW-PHY	BoW-PHY-InterfaceFunctionType
	33. C-PHY	C-PHY-InterfaceFunctionType
	34. D-PHY	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-InterfaceFunction	UCle-InterfaceFunctionType
	50. UniversalFlashStorage	UniversalFlashStorage-InterfaceFunctionType
	51. USB	USB-InterfaceFunctionType
	52. XFM	XFM-InterfaceFunctionType
	53. OtherInterfaceStandard	FunctionMap-to-StandardNameType

4.5.3.10.1. AIB

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/AIB
diagram	
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.

4.5.3.10.1.1. AIB-Base

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/AIB/AIB-Base		
diagram			
type	AIB-BaseType, BaseStandardTerminalNameAssignmentType, AIB-BaseStandardTerminalMappingType, AIB-BaseStandardTerminalNameType.		
list of enumerate values	Mapping/StandardTerminalName		
	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	



#### 4.5.3.10.1.2. AIB-Plus

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/AIB/AIB-Base			
diagram				
type	AIB-BaseType, BaseStandardTerminalNameAssignmentType, AIB-BaseStandardTerminalMappingType, AIB-BaseStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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		

#### 4.5.3.10.2. Battery Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/Battery			
diagram				
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.

**4.5.3.10.3. Camera Interface Function**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/Camera</a>
diagram	<pre> classDiagram     class Camera {         type CameraInterfaceFunctionType     }     class CSI {         type CSIType     }     class CSI2_CPHY_x1 {         type CSI2_CPHY_x1Type     }     class CSI2_CPHY_x2 {         type CSI2_CPHY_x2Type     }     class CSI2_CPHY_x3 {         type CSI2_CPHY_x3Type     }     class CSI2_CPHY_x4 {         type CSI2_CPHY_x4Type     }     class CSI2_CPHY_x5 {         type CSI2_CPHY_x5Type     }     class CSI2_CPHY_x6 {         type CSI2_CPHY_x6Type     }     class CSI2_DPHY_x1 {         type CSI2_DPHY_x1Type     }     class CSI2_DPHY_x2 {         type CSI2_DPHY_x2Type     }     class CSI2_DPHY_x3 {         type CSI2_DPHY_x3Type     }     class CSI2_DPHY_x4 {         type CSI2_DPHY_x4Type     }     class CSI2_DPHY_x5 {         type CSI2_DPHY_x5Type     }     class CSI2_DPHY_x6 {         type CSI2_DPHY_x6Type     }     class CSI2_DPHY_x7 {         type CSI2_DPHY_x7Type     }     class CSI2_DPHY_x8 {         type CSI2_DPHY_x8Type     }     class CSI3 {         type CSI3InterfaceFunctionType     }     class CPI {         type CPIInterfaceFunctionType     }      Camera --&gt; CSI     Camera --&gt; CSI2_CPHY_x1     Camera --&gt; CSI2_CPHY_x2     Camera --&gt; CSI2_CPHY_x3     Camera --&gt; CSI2_CPHY_x4     Camera --&gt; CSI2_CPHY_x5     Camera --&gt; CSI2_CPHY_x6     Camera --&gt; CSI2_DPHY_x1     Camera --&gt; CSI2_DPHY_x2     Camera --&gt; CSI2_DPHY_x3     Camera --&gt; CSI2_DPHY_x4     Camera --&gt; CSI2_DPHY_x5     Camera --&gt; CSI2_DPHY_x6     Camera --&gt; CSI2_DPHY_x7     Camera --&gt; CSI2_DPHY_x8     Camera --&gt; CSI3     Camera --&gt; CPI </pre>
type	<a href="#">CameraInterfaceFunctionType</a> , <a href="#">CSIType</a> , <a href="#">CSI-2_C-PHY-x1Type</a> , <a href="#">CSI-2_C-PHY-x2Type</a> , <a href="#">CSI-2_C-PHY-x3Type</a> , <a href="#">CSI-2_C-PHY-x4Type</a> , <a href="#">CSI-2_C-PHY-x5Type</a> , <a href="#">CSI-2_C-PHY-x6Type</a> , <a href="#">CSI-2_D-PHY-x1Type</a> , <a href="#">CSI-2_D-PHY-x2Type</a> , <a href="#">CSI-2_D-PHY-x3Type</a> , <a href="#">CSI-2_D-PHY-x4Type</a> , <a href="#">CSI-2_D-PHY-x5Type</a> , <a href="#">CSI-2_D-PHY-x6Type</a> , <a href="#">CSI-2_D-PHY-x7Type</a> , <a href="#">CSI-2_D-PHY-x8Type</a> , <a href="#">CSI-3-InterfaceFunctionType</a> , <a href="#">CPI-InterfaceFunctionType</a> .

**4.5.3.10.3.1. CSI**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI</b>			
diagram				
type	<b>CSIType, CSI-StandardTerminalNameAssignmentType, CSI-StandardTerminalMappingType, CSI-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data+</b>	<b>2. Data-</b>	<b>3. Clk+</b>	<b>4. Clk-</b>
	<b>5. SCL</b>	<b>6. SDA</b>		

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

**4.5.3.10.3.2. CSI-2 C-PHY-x1**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x1</b>			
diagram				
type	<b>CSI-2_C-PHY-x1Type, CSI-2_C-PHY-x1-StandardTerminalNameAssignmentType, CSI-2_C-PHY-x1-StandardTerminalMappingType, CSI-2_C-PHY-x1-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1_A</b>	<b>2. Data1_B</b>	<b>3. Data1_C</b>	<b>4. SCL</b>
	<b>5. SDA</b>			

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.

**4.5.3.10.3.3. CSI-2 C-PHY-x2**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x2</b>			
diagram				
type	<b>CSI-2_C-PHY-x2Type, CSI-2_C-PHY-x2-StandardTerminalNameAssignmentType, CSI-2_C-PHY-x2-StandardTerminalMappingType, CSI-2_C-PHY-x2-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1_A</b>	<b>2. Data1_B</b>	<b>3. Data1_C</b>	<b>4. SCL</b>
	<b>5. SDA</b>	<b>6. Data2_A</b>	<b>7. Data2_B</b>	<b>8. Data2_C</b>

**4.5.3.10.3.4. CSI-2 C-PHY-x3**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_C-PHY-x3</b>			
diagram				
type	<b>CSI-2_C-PHY-x3Type, CSI-2_C-PHY-x3-StandardTerminalNameAssignmentType, CSI-2_C-PHY-x3-StandardTerminalMappingType, CSI-2_C-PHY-x3-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1_A</b>	<b>2. Data1_B</b>	<b>3. Data1_C</b>	<b>4. SCL</b>
	<b>5. SDA</b>	<b>6. Data2_A</b>	<b>7. Data2_B</b>	<b>8. Data2_C</b>
	<b>9. Data3_A</b>	<b>10. Data3_B</b>	<b>11. Data3_C</b>	

## 4.5.3.10.3.5. 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	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		

## 4.5.3.10.3.6. 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	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			

**4.5.3.10.3.7. CSI-2 C-PHY-x6**

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

**4.5.3.10.3.8. CSI-2 D-PHY-x1**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x1</b>			
diagram				
type	<b>CSI-2_D-PHY-x1Type, CSI-2_D-PHY-x1-StandardTerminalNameAssignmentType, CSI-2_D-PHY-x1-StandardTerminalMappingType, CSI-2_D-PHY-x1-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1+</b>	<b>2. Data1-</b>	<b>3. Clock+</b>	<b>4. Clock-</b>
	<b>5. SCL</b>	<b>6. SDA</b>		

**4.5.3.10.3.9. CSI-2 D-PHY-x2**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x2</b>			
diagram				
type	<b>CSI-2_D-PHY-x2Type, CSI-2_D-PHY-x2-StandardTerminalNameAssignmentType, CSI-2_D-PHY-x2-StandardTerminalMappingType, CSI-2_D-PHY-x2-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1+</b>	<b>2. Data1-</b>	<b>3. Clock+</b>	<b>4. Clock-</b>
	<b>5. SCL</b>	<b>6. SDA</b>	<b>7. Data2+</b>	<b>8. Data2-</b>

**4.5.3.10.3.10. CSI-2 D-PHY-x3**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x3</b>			
diagram				
type	<b>CSI-2_D-PHY-x3Type, CSI-2_D-PHY-x3-StandardTerminalNameAssignmentType, CSI-2_D-PHY-x3-StandardTerminalMappingType, CSI-2_D-PHY-x3-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1+</b>	<b>2. Data1-</b>	<b>3. Clock+</b>	<b>4. Clock-</b>
	<b>5. SCL</b>	<b>6. SDA</b>	<b>7. Data2+</b>	<b>8. Data2-</b>
	<b>9. Data3+</b>	<b>10. Data3-</b>		

**4.5.3.10.3.11. CSI-2 D-PHY-x4**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x4</b>			
diagram				
type	<b>CSI-2_D-PHY-x4Type, CSI-2_D-PHY-x4-StandardTerminalNameAssignmentType, CSI-2_D-PHY-x4-StandardTerminalMappingType, CSI-2_D-PHY-x4-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1+</b>	<b>2. Data1-</b>	<b>3. Clock+</b>	<b>4. Clock-</b>
	<b>5. SCL</b>	<b>6. SDA</b>	<b>7. Data2+</b>	<b>8. Data2-</b>
	<b>9. Data3+</b>	<b>10. Data3-</b>	<b>11. Data4+</b>	<b>12. Data4-</b>

**4.5.3.10.3.12. CSI-2 D-PHY-x5**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x5</b>			
diagram				
type	<b>CSI-2_D-PHY-x5Type, CSI-2_D-PHY-x5-StandardTerminalNameAssignmentType, CSI-2_D-PHY-x5-StandardTerminalMappingType, CSI-2_D-PHY-x5-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1+</b>	<b>2. Data1-</b>	<b>3. Clock+</b>	<b>4. Clock-</b>
	<b>5. SCL</b>	<b>6. SDA</b>	<b>7. Data2+</b>	<b>8. Data2-</b>
	<b>9. Data3+</b>	<b>10. Data3-</b>	<b>11. Data4+</b>	<b>12. Data4-</b>
	<b>13. Data5+</b>	<b>14. Data5-</b>		



## 4.5.3.10.3.13. 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	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-

## 4.5.3.10.3.14. 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	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-		

**4.5.3.10.3.15. CSI-2 D-PHY-x8**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-2_D-PHY-x8</b>			
diagram				
type	<b>CSI-2_D-PHY-x8Type, CSI-2_D-PHY-x8-StandardTerminalNameAssignmentType, CSI-2_D-PHY-x8-StandardTerminalMappingType, CSI-2_D-PHY-x8-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Data1+</b>	<b>2. Data1-</b>	<b>3. Clock+</b>	<b>4. Clock-</b>
	<b>5. SCL</b>	<b>6. SDA</b>	<b>7. Data2+</b>	<b>8. Data2-</b>
	<b>9. Data3+</b>	<b>10. Data3-</b>	<b>11. Data4+</b>	<b>12. Data4-</b>
	<b>13. Data5+</b>	<b>14. Data5-</b>	<b>15. Data6+</b>	<b>16. Data6-</b>
	<b>17. Data7+</b>	<b>18. Data7-</b>	<b>19. Data8+</b>	<b>20. Data8-</b>

**4.5.3.10.3.16. CSI-3**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CSI-3</b>			
diagram				
type	<b>CSI-3-InterfaceFunctionType, CSI-3-StandardTerminalNameAssignmentType, CSI-3-StandardTerminalMappingType, CSI-3-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. M-TX</b>	<b>2. M-RX</b>		

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.

#### 4.5.3.10.3.1. CPI

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/Camera/CPI</b>			
diagram				
type	<b>CPI-InterfaceFunctionType, CPI-StandardTerminalNameAssignmentType, CPI-StandardTerminalMappingType, CPI-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. I2C_SCL</b>	<b>2. I2C_SDA</b>	<b>3. CPI_CLK</b>	<b>4. CPI_DAT[0]</b>
	<b>5. CPI_DAT[1]</b>	<b>6. CPI_DAT[2]</b>	<b>7. CPI_DAT[3]</b>	<b>8. CPI_DAT[4]</b>
	<b>9. CPI_DAT[5]</b>	<b>10. CPI_DAT[6]</b>	<b>11. CPI_DAT[7]</b>	

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

#### 4.5.3.10.4. Compute Express Link Function

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ComputeExpressLink</b>			
diagram				
type	<b>ComputeExpressLink-InterfaceFunctionType, CXL-x4-InterfaceFunctionType, CXL-x8-InterfaceFunctionType, CXL-x16-InterfaceFunctionType.</b>			

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

**4.5.3.10.4.1. CXL-x4**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/ComputeExpressLink/CXL-x4</b>			
diagram				
type	<b>CXL-x4-InterfaceFunctionType, CXL-x4-StandardTerminalNameAssignmentType, CXL-x4-MandatoryMappingType, CXL-x4-MandatoryStandardTerminalNameType, CXL-x4-OptionalMappingType, CXL-x4-OptionalStandardTerminalNameType</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	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	32.
	<b>OptionalMapping/StandardTerminalName</b>			
	1. REFCLKp1	2. REFCLKn1	3. PERST1#	4. CLKREQ#

## 4.5.3.10.4.2. CXL-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/ComputeExpressLink/CXL-x8			
diagram				
type	CXL-x8-InterfaceFunctionType, CXL-x8-StandardTerminalNameAssignmentType, CXL-x8-MandatoryMappingType, CXL-x8-MandatoryStandardTerminalNameType, CXL-x8-OptionalMappingType, CXL-x8-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. 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
	OptionalMapping/StandardTerminalName			
	1. REFCLKp1	2. REFCLKn1	3. PERST1#	4. CLKREQ#

**4.5.3.10.4.3. 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#

**4.5.3.10.5. Display Bus Function**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus</a>
diagram	<p>The diagram illustrates the structure of the Display Bus Interface. A central node, represented by a circle with a plus sign, connects to a 'DisplayBus' entity on the left. This central node then branches into eight sub-entities, each enclosed in a dashed yellow box. These sub-entities are: DBI-TypeA, DBI-TypeB, DBI-TypeC, DPI-Type1, DPI-Type2-3, DPI-Type4, DSI-2OptionC, and DSI-2OptionD. Each sub-entity has a 'type' attribute pointing to its respective InterfaceFunctionType.</p>
type	<a href="#">DisplayBusInterfaceFunctionType</a> , <a href="#">DBI-TypeA-InterfaceFunctionType</a> , <a href="#">DBI-TypeB-InterfaceFunctionType</a> , <a href="#">DBI-TypeC-InterfaceFunctionType</a> , <a href="#">DPI-Type1-InterfaceFunctionType</a> , <a href="#">DPI-Type2-3-InterfaceFunctionType</a> , <a href="#">DPI-Type4-InterfaceFunctionType</a> , <a href="#">DSI-2OptionC-InterfaceFunctionType</a> , <a href="#">DSI-2OptionD-InterfaceFunctionType</a> .

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

4.5.3.10.5.1. DBI-TypeA

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DBI-TypeA			
diagram				
type	DBI-TypeA-InterfaceFunctionType, DBI-TypeA-StandardTerminalNameAssignmentType, DBI-TypeA-MandatoryStandardTerminalMappingType, DBI-TypeA-MandatoryStandardTerminalNameType, DBI-TypeA-OptionalStandardTerminalMappingType, DBI-TypeA-OptionalStandardTerminalNameType.			
list of enumerate values	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			



## 4.5.3.10.5.2. DBI-TypeB

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DBI-TypeB		
diagram			
type	DBI-TypeB-InterfaceFunctionType, DBI-TypeB-StandardTerminalNameAssignmentType, DBI-TypeB-MandatoryStandardTerminalMappingType, DBI-TypeB-MandatoryStandardTerminalNameType, DBI-TypeB-OptionalStandardTerminalMappingType, DBI-TypeB-OptionalStandardTerminalNameType.		
list of enumerate values	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]		

**4.5.3.10.5.3. DBI-TypeC**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DBI-TypeC</b>			
diagram				
type	<b>DBI-TypeB-InterfaceFunctionType, DBI-TypeC-StandardTerminalNameAssignmentType, DBI-TypeC-MandatoryStandardTerminalMappingType, DBI-TypeC-MandatoryStandardTerminalNameType, DBI-TypeC-OptionalStandardTerminalMappingType, DBI-TypeC-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. VDD</b>	<b>2. AGND</b>	<b>3. VDDI</b>	<b>4. DGND</b>
	<b>5. CSX</b>	<b>6. RESX</b>	<b>7. SCL</b>	
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. D/CX</b>	<b>2. DOUT</b>	<b>3. DIN</b>	<b>4. SDA</b>

**4.5.3.10.5.4. DPI-Type1**

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

#### 4.5.3.10.5.4. DPI - Type 1 (cont'd)

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

#### 4.5.3.10.5.5. 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	MandatoryMapping/StandardTerminalName		
	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]
	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]	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

4.5.3.10.5.6. DPI-Type4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DPI-Type4		
diagram			
type	DPI-Type4-InterfaceFunctionType, DPI-Type4-StandardTerminalNameAssignmentType, DPI-Type4-MandatoryStandardTerminalMappingType, DPI-Type4-MandatoryStandardTerminalNameType, DPI-Type4-OptionalStandardTerminalMappingType, DPI-Type4-OptionalStandardTerminalNameType.		
list of enumerate values	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]		

## 4.5.3.10.5.7. DSI-2-OptionC

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DSI-2OptionC</b>			
diagram				
type	<b>DSI-2OptionC-InterfaceFunctionType, DSI-2OptionC-StandardTerminalNameAssignmentType, DSI-2OptionC-MandatoryStandardTerminalMappingType, DSI-2OptionC-MandatoryStandardTerminalNameType, DSI-2OptionC-OptionalStandardTerminalMappingType, DSI-2OptionC-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Data0_A</b>	<b>2. Data0_B</b>	<b>3. Data0_C</b>	
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. Data1_A</b>	<b>2. Data1_B</b>	<b>3. Data1_C</b>	<b>4. Data2_A</b>
	<b>5. Data2_B</b>	<b>6. Data2_C</b>	<b>7. Data3_A</b>	<b>8. Data3_B</b>
	<b>9. Data3_C</b>			

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

4.5.3.10.5.8. DSI-2-OptionD

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/DisplayBus/DSI-2OptionD			
diagram				
type	DSI-2OptionD-InterfaceFunctionType, DSI-2OptionD-StandardTerminalNameAssignmentType, DSI-2OptionD-MandatoryStandardTerminalMappingType, DSI-2OptionD-MandatoryStandardTerminalNameType, DSI-2OptionD-OptionalStandardTerminalMappingType, DSI-2OptionD-OptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Data0+	2. Data0-	3. Clock+	4. Clock-
	OptionalMapping/StandardTerminalName			
	1. Data1+	2. Data1-	3. Data2+	4. Data2-
	5. Data3+	6. Data3-		

#### 4.5.3.10.6. DDR3 Interface Function

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3</a>
diagram	<pre> classDiagram     class DDR3 {         type DDR3-InterfaceFunctionType     }     class DDR3_x4 {         type DDR3-x4-InterfaceFunctionType     }     class DDR3_x4_DualDie {         type DDR3-x4-DualDie-InterfaceFunctionType     }     class DDR3_x4_QuadDie {         type DDR3-x4-QuadDie-InterfaceFunctionType     }     class DDR3_x8 {         type DDR3-x8-InterfaceFunctionType     }     class DDR3_x8_DualDie {         type DDR3-x8-DualDie-InterfaceFunctionType     }     class DDR3_x8_QuadDie {         type DDR3-x8-QuadDie-InterfaceFunctionType     }     class DDR3_x16 {         type DDR3-x16-InterfaceFunctionType     }     class DDR3_x16_DualDie {         type DDR3-x16-DualDie-InterfaceFunctionType     }     class DDR3_x16_QuadDie {         type DDR3-x16-QuadDie-InterfaceFunctionType     }     class DDR3_Controller {         type DDR3-Controller-InterfaceFunctionType     }     DDR3 --&gt; DDR3_x4     DDR3 --&gt; DDR3_x4_DualDie     DDR3 --&gt; DDR3_x4_QuadDie     DDR3 --&gt; DDR3_x8     DDR3 --&gt; DDR3_x8_DualDie     DDR3 --&gt; DDR3_x8_QuadDie     DDR3 --&gt; DDR3_x16     DDR3 --&gt; DDR3_x16_DualDie     DDR3 --&gt; DDR3_x16_QuadDie     DDR3 --&gt; DDR3_Controller   </pre>
type	<a href="#">DDR3-InterfaceFunctionType</a> , <a href="#">DDR3-x4-InterfaceFunctionType</a> , <a href="#">DDR3-x4-DualDie-InterfaceFunctionType</a> , <a href="#">DDR3-x4-QuadDie-InterfaceFunctionType</a> , <a href="#">DDR3-x8-InterfaceFunctionType</a> , <a href="#">DDR3-x8-DualDie-InterfaceFunctionType</a> , <a href="#">DDR3-x8-QuadDie-InterfaceFunctionType</a> , <a href="#">DDR3-x16-InterfaceFunctionType</a> , <a href="#">DDR3-x16-DualDie-InterfaceFunctionType</a> , <a href="#">DDR3-x16-QuadDie-InterfaceFunctionType</a> , <a href="#">DDR3-Controller-InterfaceFunctionType</a> .

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

**4.5.3.10.6.1. DDR3-x4**

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



## 4.5.3.10.6.2. DDR3-x4 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x4-DualDie			
diagram				
type	DDR3-x4-DualDie-InterfaceFunctionType, DDR3-x4-DualDieStandardTerminalNameAssignmentType, DDR3-x4-DualDieMandatoryStandardTerminalMappingType, DDR3-x4-DualDieMandatoryStandardTerminalNameType, DDR3-x4-DualDieOptionalStandardTerminalMappingType, DDR3-x4-DualDieOptionalStandardTerminalNameType.			
list of enumerate values	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	

## 4.5.3.10.6.3. DDR3-x4 – Quad Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x4-QuadDie			
diagram				
type	DDR3-x4–QuadDie-InterfaceFunctionType, DDR3-x4-QuadDieStandardTerminalNameAssignmentType, DDR3-x4-QuadDieMandatoryStandardTerminalMappingType, DDR3-x4-QuadDieMandatoryStandardTerminalNameType, DDR3-x4-QuadDieOptionalStandardTerminalMappingType, DDR3-x4-QuadDieOptionalStandardTerminalNameType.			
list of enumerate values	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. 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]
	OptionalMapping/StandardTerminalName			
	1. A13	2. A13	3. A13	

## 4.5.3.10.6.4. DDR3-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x8			
diagram				
type	DDR3-x8-InterfaceFunctionType, DDR3-x8StandardTerminalNameAssignmentType, DDR3-x8MandatoryStandardTerminalMappingType, DDR3-x8MandatoryStandardTerminalNameType, DDR3-x8OptionalStandardTerminalMappingType, DDR3-x8OptionalStandardTerminalNameType.			
list of enumerate values	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	

## 4.5.3.10.6.5. DDR3-x8 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x8-DualDie			
diagram				
type	DDR3-x8-DualDie-InterfaceFunctionType, DDR3-x8-DualDieStandardTerminalNameAssignmentType, DDR3-x8-DualDieMandatoryStandardTerminalMappingType, DDR3-x8-DualDieMandatoryStandardTerminalNameType, DDR3-x8-DualDieOptionalStandardTerminalMappingType, DDR3-x8-DualDieOptionalStandardTerminalNameType.			
list of enumerate values	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. 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]	50.	51.	52.
	OptionalMapping/StandardTerminalName			
	1. A13	2. A13	3. A13	

## 4.5.3.10.6.6. DDR3-x8 – Quad Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x8-QuadDie			
diagram				
type	DDR3-x8-QuadDie-InterfaceFunctionType, DDR3-x8-QuadDieStandardTerminalNameAssignmentType, DDR3-x8-QuadDieMandatoryStandardTerminalMappingType, DDR3-x8-QuadDieMandatoryStandardTerminalNameType, DDR3-x8-QuadDieOptionalStandardTerminalMappingType, DDR3-x8-QuadDieOptionalStandardTerminalNameType.			
list of enumerate values	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	

**4.5.3.10.6.7. DDR3-x16**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x16</b>			
diagram				
type	<b>DDR3-x16–InterfaceFunctionType, DDR3-x16StandardTerminalNameAssignmentType, DDR3-x16MandatoryStandardTerminalMappingType, DDR3-x16MandatoryStandardTerminalNameType, DDR3-x16OptionalStandardTerminalMappingType, DDR3-x16OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	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]	
	<b>OptionalMapping/StandardTerminalName</b>			
	1. A13	2. A13	3. A13	

## 4.5.3.10.6.8. DDR3-x16 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x16-DualDie			
diagram				
type	DDR3-x16–DualDie-InterfaceFunctionType, DDR3-x16-DualDieStandardTerminalNameAssignmentType, DDR3-x16-DualDieMandatoryStandardTerminalMappingType, DDR3-x16-DualDieMandatoryStandardTerminalNameType, DDR3-x16-DualDieOptionalStandardTerminalMappingType, DDR3-x16-DualDieOptionalStandardTerminalNameType.			
list of enumerate values	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. 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	

**4.5.3.10.6.9. DDR3-x16 – Quad Die**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3/DDR3-x16-QuadDie</b>			
diagram				
type	<b>DDR3-x16-QuadDie-InterfaceFunctionType, DDR3-x16-QuadDieStandardTerminalNameAssignmentType, DDR3-x16-QuadDieMandatoryStandardTerminalMappingType, DDR3-x16-QuadDieMandatoryStandardTerminalNameType, DDR3-x16-QuadDieOptionalStandardTerminalMappingType, DDR3-x16-QuadDieOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	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. 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]	
list of enumerate values	<b>OptionalMapping/StandardTerminalName</b>			
	1. A13	2. A13	3. A13	



## 4.5.3.10.6.10. DDR3 Controller

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR3-Controller			
diagram				
type	DDR3-Controller-InterfaceFunctionType, DDR3-ControllerStandardTerminalNameAssignmentType, DDR3-ControllerMandatoryStandardTerminalMappingType, DDR3-ControllerMandatoryStandardTerminalNameType, DDR3-ControllerOptionalStandardTerminalMappingType, DDR3-ControllerOptionalStandardTerminalNameType.			
list of enumerate values	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]

**4.5.3.10.6.10 DDR3 Controller (cont'd)**

list of enumerate values  (cont.)	<b>OptionalMapping/StandardTerminalName</b>			
	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	

## 4.5.3.10.7. DDR4 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4
diagram	<pre> classDiagram     class DDR4 {         type: DDR4-InterfaceFunctionType     }     class DDR4_x4 {         type: DDR4-x4-InterfaceFunctionType     }     class DDR4_x4_DualDie {         type: DDR4-x4-DualDie-InterfaceFun...     }     class DDR4_x8 {         type: DDR4-x8-InterfaceFunctionType     }     class DDR4_x8_DualDie {         type: DDR4-x8-DualDie-InterfaceFun...     }     class DDR4_x16 {         type: DDR4-x16-InterfaceFunctionType     }     class DDR4_x16_DualDie {         type: DDR4-x16-DualDie-InterfaceFu...     }     class DDR4_x32 {         type: DDR4-x32-InterfaceFunctionType     }     class DDR4_x72 {         type: DDR4-x72-InterfaceFunctionType     }     class DDR4DB02 {         type: DDR4DB02-InterfaceFunctionT...     }     class DDR4RCD02 {         type: DDR4RCD02-InterfaceFunction...     }     class DDR4_NVDIMM_N {         type: DDR4-NVDIMM-NinterfaceFunc...     }     class LPDDR4_SingleChannel {         type: LPDDR4-SingleChannelInterfac...     }     class LPDDR4_DualChannel {         type: LPDDR4-DualChannelInterface...     }     DDR4 &lt; -- DDR4_x4     DDR4 &lt; -- DDR4_x4_DualDie     DDR4 &lt; -- DDR4_x8     DDR4 &lt; -- DDR4_x8_DualDie     DDR4 &lt; -- DDR4_x16     DDR4 &lt; -- DDR4_x16_DualDie     DDR4 &lt; -- DDR4_x32     DDR4 &lt; -- DDR4_x72     DDR4 &lt; -- DDR4DB02     DDR4 &lt; -- DDR4RCD02     DDR4 &lt; -- DDR4_NVDIMM_N     DDR4 &lt; -- LPDDR4_SingleChannel     DDR4 &lt; -- LPDDR4_DualChannel   </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-NInterfaceFunctionType, DDR4-x72-InterfaceFunctionType, LPDDR4-SingleChannelInterfaceFunctionType, LPDDR4-DualChannelInterfaceFunctionType,

**4.5.3.10.7.1. DDR4-x4**

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x4			
diagram				
type	DDR4-x4-InterfaceFunctionType, DDR3-x4StandardTerminalNameAssignmentType, DDR4-x4MandatoryStandardTerminalMappingType, DDR4-x4MandatoryStandardTerminalNameType, DDR4-x4OptionalStandardTerminalMappingType, DDR4-x4OptionalStandardTerminalNameType.			
list of enumerate values	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.

## 4.5.3.10.7.2. DDR4-x4 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x4-DualDie			
diagram				
type	DDR4-x4-DualDie-InterfaceFunctionType, DDR4-x4-DualDieStandardTerminalNameAssignmentType, DDR4-x4-DualDieMandatoryStandardTerminalMappingType, DDR4-x4-DualDieMandatoryStandardTerminalNameType, DDR4-x4-DualDieOptionalStandardTerminalMappingType, DDR4-x4-DualDieOptionalStandardTerminalNameType.			
list of enumerate values	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. 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]		
	OptionalMapping/StandardTerminalName			
	1. TEN	2. A17		

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

**4.5.3.10.7.3. DDR4-x8**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x8</b>			
diagram				
type	<b>DDR4-x8–InterfaceFunctionType, DDR4-x8StandardTerminalNameAssignmentType, DDR4-x8MandatoryStandardTerminalMappingType, DDR4-x8MandatoryStandardTerminalNameType, DDR4-x8OptionalStandardTerminalMappingType, DDR4-x8OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	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]			
	<b>OptionalMapping/StandardTerminalName</b>			
	1. TEN			

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

## 4.5.3.10.7.4. DDR4-x8 Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x8-DualDie			
diagram				
type	DDR4-x8-DualDie-InterfaceFunctionType, DDR4-x8-DualDieStandardTerminalNameAssignmentType, DDR4-x8-DualDieMandatoryStandardTerminalMappingType, DDR4-x8-DualDieMandatoryStandardTerminalNameType, DDR4-x8-DualDieOptionalStandardTerminalMappingType, DDR4-x8-DualDieOptionalStandardTerminalNameType.			
list of enumerate values	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. 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]
	OptionalMapping/StandardTerminalName			
	1. TEN			

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

**4.5.3.10.7.5. DDR4-x16**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x16</b>			
diagram				
type	<b>DDR4-x16–InterfaceFunctionType, DDR4-x16StandardTerminalNameAssignmentType, DDR4-x16StandardTerminalMappingType, DDR4-x16StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.



## 4.5.3.10.7.6. DDR4-x16 – Dual Die

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x16-DualDie			
diagram	<pre>graph LR     subgraph "DDR4-x16-DualDie-InterfaceFunctionType"         direction LR         A[DDR4-x16-DualDie type   DDR4-x16-DualDie-InterfaceFu...] --- B[...]         B --- C[StandardTerminalNameAssign... type   DDR4-x16-DualDie-StandardTe... 1..∞]         C --- D[...]         D --- E[Mapping type   DDR4-x16-DualDie-StandardTe... 61]         E --- F[...]         F --- G[StandardTerminalName type   DDR4-x16-DualDie-StandardTe... StandardTerminalMappingType TerminalMapID type   xs:string]     end</pre>			
type	DDR4-x16-DualDie-InterfaceFunctionType, DDR4-x16-DualDieStandardTerminalNameAssignmentType, DDR4-x16-DualDieStandardTerminalMappingType, DDR4-x16-DualDieStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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.

**4.5.3.10.7.7. DDR4-x32**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x32</b>			
diagram				
type	<b>DDR4-x32-InterfaceFunctionType, DDR4-x32StandardTerminalNameAssignmentType, DDR4-x32StandardTerminalMappingType, DDR4-x32StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.

## 4.5.3.10.7.8. DDR4-x72

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-x72			
diagram				
type	DDR4-x72-InterfaceFunctionType, DDR4-x72-StandardTerminalNameAssignmentType, DDR4-x72-StandardTerminalMappingType, DDR4-x72-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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

**4.5.3.10.7.8 DDR4-x72 (cont'd)**

list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	109. DQS15_t/TDQS15_t	110. DQS16_c/TDQS16_c	111. DQS16_t/TDQS16_t	112. DQS17_c/TDQS17_c
	113. DQS1_t	114. DQS10_c/TDQS10_c	115. DQS10_t/TDQS10_t	116. DQS11_c/TDQS11_c
	117. DQS11_t/TDQS11_t	118. DQS12_c/TDQS12_c	119. DQS12_t/TDQS12_t	120. DQS13_c/TDQS13_c
	121. DQS13_t/TDQ13_t	122. DQS14_c/TDQS14_c	123. DQS14_t/TDQS14_t	124. DQS15_c/TDQS15_c
	125. DQS15_t/TDQS15_t	126. DQS16_c/TDQS16_c	127. DQS16_t/TDQS16_t	128. DQS17_c/TDQS17_c
	129. DQS17_t/TDQS17_t	130. DQS2_c	131. DQS2_t	132. DQS3_c
	133. DQS3_t	134. DQS4_c	135. DQS4_t	136. DQS5_c
	137. DQS5_t	138. DQS6_c	139. DQS6_t	140. DQS7_c
	141. DQS7_t	142. DQS8_c	143. DQS8_t	144. DQS9_t/TDQS9_t
	145. EVENT_n	146. NC	147. NC/C2	148. ODT0
	149. ODT1/NC	150. PARITY	151. RAS_n/A16	152. RESET_n
	153. SA0	154. SA1	155. SA2	156. SCL
	157. SDA	158. VDD	159. VDDSPD	160. VPP
	161. VREFCA	162. VSS	163. VTT	164. WE_n/A14

#### 4.5.3.10.7.9. LPDDR4 – Single Channel

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/LPDDR4-SingleChannel</b>			
diagram	<p>The diagram illustrates the UML structure for the LPDDR4-SingleChannel interface. It shows a hierarchy of types: <b>LPDDR4-SingleChannel</b> (type   LPDDR4-SingleChannelInterfac...) is associated with <b>StandardTerminalNameAssign...</b> (type   LPDDR4-SingleChannelStandar...). This is followed by <b>Mapping</b> (type   LPDDR4-SingleChannelStandar...) which is associated with <b>LPDDR4-SingleChannelStandardTerminalMappingType</b>. This type contains <b>StandardTerminalName</b> (type   LPDDR4-SingleChannelStandar...) and <b>TerminalMapID</b> (type   xs:string). A <b>constraints</b> box is also shown. Multiplicities like 1..∞ and 40 are indicated on the associations.</p>			
type	<b>LPDDR4-SingleChannel-InterfaceFunctionType, LPDDR4-SingleChannelStandardTerminalNameAssignmentType, LPDDR4-SingleChannelStandardTerminalMappingType, LPDDR4-SingleChannelStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.

**4.5.3.10.7.10. LPDDR4 – Dual Channel**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/LPDDR4-DualChannel</b>			
diagram				
type	<b>LPDDR4-DualChannelInterfaceFunctionType, LPDDR4-DualChannelStandardTerminalNameAssignmentType, LPDDR4-DualChannelStandardTerminalMappingType, LPDDR4-DualChannelStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.

**4.5.3.10.7.11. DDR4DB02**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4DB02</b>			
diagram				
type	<b>DDR4DB02-InterfaceFunctionType, DDR4DB02-StandardTerminalNameAssignmentType, DDR4DB02-StandardTerminalMappingType, DDR4DB02-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. BCOM0</b>	<b>2. BCOM1</b>	<b>3. BCOM2</b>	<b>4. BCOM3</b>
	<b>5. BCKE</b>	<b>6. BODT</b>	<b>7. BCK_t</b>	<b>8. BCK_c</b>
	<b>9. BVrefCA</b>			

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

## 4.5.3.10.7.12. DDR4-NVDIMM-N

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4-NVDIMM-N			
diagram				
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	MandatoryMapping/StandardTerminalName			
	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]



## 4.5.3.10.7.12 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		

4.5.3.10.7.13. DDR4RCD02

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/DDR4RCD02			
diagram	<p>The diagram illustrates the structure of the DDR4RCD02 interface. It shows a sequence of elements: <b>DDR4RCD02</b> (type <code>DDR4RCD02-InterfaceFunctionType</code>), <b>StandardTerminalNameAssignment</b> (type <code>DDR4RCD02-StandardTerminalNameAssignmentType</code>, multiplicity <code>1..∞</code>), <b>Mapping</b> (type <code>DDR4RCD02-StandardTerminalMappingType</code>, multiplicity <code>9</code>), and <b>StandardTerminalName</b> (type <code>DDR4RCD02-StandardTerminalNameType</code>, multiplicity <code>9</code>). The <b>Mapping</b> element is associated with <b>StandardTerminalName</b> via a relationship (type <code>xs:string</code>). The diagram is enclosed in a dashed box labeled <code>DDR4RCD02-InterfaceFunctionType</code>.</p>			
type	DDR4RCD02-InterfaceFunctionType, DDR4RCD02-StandardTerminalNameAssignmentType, DDR4RCD02-StandardTerminalMappingType, DDR4RCD02-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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.

**4.5.3.10.8. DDR5 Interface Function**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5</a>
diagram	
type	<a href="#">DDR5-InterfaceFunctionType</a> , <a href="#">DDR5-x4-InterfaceFunctionType</a> , <a href="#">DDR5-x8-InterfaceFunctionType</a> , <a href="#">DDR5-x16-InterfaceFunctionType</a> , <a href="#">DDR5DB01-InterfaceFunctionType</a> , <a href="#">LPDDR5-InterfaceFunctionType</a> , <a href="#">GDDR5-InterfaceFunctionType</a> <a href="#">GDDR5X-InterfaceFunctionType</a> .

4.5.3.10.8.1. DDR5-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5-x4			
diagram				
type	DDR5-x4-InterfaceFunctionType, DDR5-x4-StandardTerminalNameAssignmentType, DDR5-x4-StandardTerminalMappingType, DDR5-x4-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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.

**4.5.3.10.8.2. DDR5-x8**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5-x8</b>			
diagram				
type	<b>DDR5-x8-InterfaceFunctionType, DDR5-x8-StandardTerminalNameAssignmentType, DDR5-x8-StandardTerminalMappingType, DDR5-x8-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.

**4.5.3.10.8.3. DDR5-x16**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5-x16</b>		
diagram			
type	<b>DDR5-x16-InterfaceFunctionType, DDR5-x16-StandardTerminalNameAssignmentType, DDR5-x16-StandardTerminalMappingType, DDR5-x16-StandardTerminalNameType.</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	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.

**4.5.3.10.8.4. DDR5DB01**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5DB01</b>			
diagram				
type	<b>DDR5DB01-InterfaceFunctionType, DDR5DB01-StandardTerminalNameAssignmentType, DDR5DB01-StandardTerminalMappingType, DDR5DB01-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.

**4.5.3.10.8.5. LPDDR5**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/LPDDR5</b>		
diagram			
type	<b>LPDDR5-InterfaceFunctionType, LPDDR5-StandardTerminalNameAssignmentType, LPDDR5-StandardTerminalMappingType, LPDDR5-StandardTerminalNameType.</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	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. DMI0	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.



**4.5.3.10.8.6. GDDR5**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/GDDR5</b>		
diagram			
type	<b>GDDR5-InterfaceFunctionType, GDDR5-StandardTerminalNameAssignmentType, GDDR5-StandardTerminalMappingType, GDDR5-StandardTerminalNameType.</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	1. A0	2. A1	3. A2
	5. A4	6. A5	7. A6
	9. A8	10. A9	11. A10
	13. A12	14. A13	15. ABL_n
	17. BA1	18. BA2	19. BA3
	21. CK_c	22. CK_t	23. CKE_n
	25. DBI0_n	26. DBI1_n	27. DBI2_n
	29. DQ[0]	30. DQ[1]	31. DQ[2]
	33. DQ[4]	34. DQ[5]	35. DQ[6]
	37. DQ[8]	38. DQ[9]	39. DQ[10]
	41. DQ[12]	42. DQ[13]	43. DQ[14]
	45. DQ[16]	46. DQ[17]	47. DQ[18]
	49. DQ[20]	50. DQ[21]	51. DQ[22]
	53. DQ[24]	54. DQ[25]	55. DQ[26]
	57. DQ[28]	58. DQ[29]	59. DQ[30]
	61. EDC0	62. EDC1	63. EDC2
	65. MF	66. RAS_n	67. RESET_n
	69. VDD	70. VDDQ	71. VPP
	73. VREFD	74. VSS	75. VSSQ
	77. WCK01_t	78. WCK23_c	79. WCK23_t
	81. ZQ		

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

**4.5.3.10.8.7. GDDR5X**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/GDDR5X</b>			
diagram				
type	<b>GDDR5X-InterfaceFunctionType, GDDR5X-StandardTerminalNameAssignmentType, GDDR5X-StandardTerminalMappingType, GDDR5X-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.

#### 4.5.3.10.9. DDR6 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR5
diagram	
type	DDR6-InterfaceFunctionType, GDDR6-InterfaceFunctionType

#### 4.5.3.10.9.1. GDDR6

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DDR4/LPDDR4-DualChannel		
diagram			
type	GDDR6-InterfaceFunctionType, GDDR6-StandardTerminalNameAssignmentType, GDDR6-StandardTerminalMappingType, GDDR6-StandardTerminalNameType.		
list of enumerate values	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. CA11
	13. CA12	14. CA13	15. CABI_n
	16. CK_c	17. CK_t	18. CKE_n
	19. DBI0_n	20. DBI1_n	21. DQ[0]
	22. DQ[1]	23. DQ[2]	24. DQ[3]
	25. DQ[4]	26. DQ[5]	27. DQ[6]
	28. DQ[7]	29. DQ[8]	30. DQ[9]
	31. DQ[10]	32. DQ[11]	33. DQ[12]
	34. DQ[13]	35. DQ[14]	36. DQ[15]
	37. EDC0	38. EDC1	39. RESET_n
	40. TCK	41. TDI	42. TDO
	43. TMS	44. VDD	45. VDDQ
	46. VPP	47. VREFC	48. VSS
	49. WCK0_c	50. WCK0_t	51. WCK1_c
	52. WCK1_t	53. ZQ	

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

**4.5.3.10.10. DigRF3G Interface Function**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DigRF3G</b>			
diagram				
type	<b>DigRF3G-InterfaceFunctionType, DigRF3G-StandardTerminalNameAssignmentType, DigRF3G-StandardTerminalMappingType, DigRF3G-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. SysClk</b>	<b>2. SysClkEn</b>	<b>3. TxDataP</b>	<b>4. TxDataN</b>
	<b>5. RxDataP</b>	<b>6. RxDataN</b>		

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.

**4.5.3.10.11. DigRFv4 Interface Function**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/DigRFv4</b>			
diagram				
type	<b>DigRFv4-InterfaceFunctionType, DigRFv4-StandardTerminalNameAssignmentType, DigRFv4-StandardTerminalMappingType, DigRFv4-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. TxDataP</b>	<b>2. TxDataN</b>	<b>3. RxDataP</b>	<b>4. RxDataN</b>
	<b>5. DigRFEN</b>	<b>6. RefClkEn</b>	<b>7. RefClk</b>	

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

4.5.3.10.12. 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 EE1002_SPD_EEPROM_InterfaceFunctionType {         type EE1002-SPD-EEPROM-StandardTerminalNameAssignmentType     }     class EE1002_SPD_EEPROM_StandardTerminalNameAssignmentType {         type EE1002-SPD-EEPROM-StandardTerminalMappingType     }     class EE1002_SPD_EEPROM_StandardTerminalMappingType {         type EE1002-SPD-EEPROM-StandardTerminalNameType         type xs:string     }     EE1002_SPD_EEPROM --&gt; EE1002_SPD_EEPROM_InterfaceFunctionType     EE1002_SPD_EEPROM_InterfaceFunctionType --&gt; EE1002_SPD_EEPROM_StandardTerminalNameAssignmentType     EE1002_SPD_EEPROM_StandardTerminalNameAssignmentType --&gt; EE1002_SPD_EEPROM_StandardTerminalMappingType     EE1002_SPD_EEPROM_StandardTerminalMappingType --&gt; EE1002_SPD_EEPROM_StandardTerminalNameType     EE1002_SPD_EEPROM_StandardTerminalMappingType --&gt; xs_string</pre>			
type	EE1002-SPD-EEPROM-InterfaceFunctionType, EE1002-SPD-EEPROM-StandardTerminalNameAssignmentType, EE1002-SPD-EEPROM-StandardTerminalMappingType, EE1002-SPD-EEPROM-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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.

**4.5.3.10.13. Embedded Display Port Interface**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/eTrak</b>			
diagram				
type	<b>EmbeddedDisplayPort-InterfaceFunctionType, EmbeddedDisplayPort-StandardTerminalNameAssignmentType, EmbeddedDisplayPort-MandatoryMappingType, EmbeddedDisplayPort-OptionalMappingType, EmbeddedDisplayPort-MandatoryStandardTerminalNameType, EmbeddedDisplayPort-OptionalStandardTerminalNameType .</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	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	
	<b>OptionalMapping/StandardTerminalName</b>			
	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.

#### 4.5.3.10.14. Ethernet

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/Ethernet</a>
diagram	
type	<a href="#">EthernetInterfaceFunctionType</a> , <a href="#">Ethernet1000BASE-LX-InterfaceFunctionType</a> , <a href="#">Ethernet1000BASE-SX-InterfaceFunctionType</a> , <a href="#">Ethernet1000BASE-CX-InterfaceFunctionType</a> .

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

##### 4.5.3.10.14.1. Ethernet1000BASE-LX

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ Ethernet1000BASE-LX</a>			
diagram				
type	<a href="#">Ethernet1000BASE-LX-InterfaceFunctionType</a> , <a href="#">Ethernet1000BASE-LX-StandardTerminalNameAssignmentType</a> , <a href="#">Ethernet1000BASE-LX-StandardTerminalMappingType</a> , <a href="#">Ethernet1000BASE-LX-StandardTerminalNameType</a> .			
list of enumerate values	<a href="#">Mapping/StandardTerminalName</a>			
	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

**4.5.3.10.14.2. Ethernet1000BASE-SX**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ Ethernet1000BASE-SX</b>			
diagram				
type	<b>Ethernet1000BASE-SX-InterfaceFunctionType, Ethernet1000BASE-SX-StandardTerminalNameAssignmentType, Ethernet1000BASE-SX-StandardTerminalMappingType, Ethernet1000BASE-SX-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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



## 4.5.3.10.14.3. Ethernet1000BASE-CX

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ Ethernet1000BASE-CX			
diagram				
type	Ethernet1000BASE-CX-InterfaceFunctionType, Ethernet1000BASE-CX-StandardTerminalNameAssignmentType, Ethernet1000BASE-CX-StandardTerminalMappingType, Ethernet1000BASE-CX-StandardTerminalNameType.			
list of enumerate values	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

## 4.5.3.10.15. 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			
	25. I	26. Q	27. 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.



## 4.5.3.10.17.1. HBM1, HBM2 and HBM2E Interface Functions

path	<ol style="list-style-type: none"> <li>1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM1</li> <li>2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM2</li> <li>3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM-HBM2E</li> </ol>			
diagram				
type	HBM1-2-2E-InterfaceFunctionType, HBM1-2-2E-StandardTerminalNameAssignmentType, HBM1-2-2E-StandardTerminalMappingType, HBM1-2-2E-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	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. CKa_c	10. CKb_c	11. CKc_c	12. CKd_c
	13. CKe_c	14. CKf_c	15. CKg_c	16. CKh_c
	17. CKEa	18. CKEb	19. CKEc	20. CKEd
	21. CKEe	22. CKEf	23. CKEg	24. CKEh
	25. Ca0	26. Ca1	27. Ca2	28. Ca3
	29. Ca4	30. Ca5	31. Ca6	32. Ca7
	33. Ca8	34. Cb0	35. Cb1	36. Cb2
	37. Cb3	38. Cb4	39. Cb5	40. Cb6
	41. Cb7	42. Cb8	43. Cc0	44. Cc1
	45. Cc2	46. Cc3	47. Cc4	48. Cc5
	49. Cc6	50. Cc7	51. Cc8	52. Cd0
	53. Cd1	54. Cd2	55. Cd3	56. Cd4
	57. Cd5	58. Cd6	59. Cd7	60. Cd8
	61. Ce0	62. Ce1	63. Ce2	64. Ce3
	65. Ce4	66. Ce5	67. Ce6	68. Ce7
	69. Ce8	70. Cf0	71. Cf1	72. Cf2
	73. Cf3	74. Cf4	75. Cf5	76. Cf6
	77. Cf7	78. Cf8	79. Cg0	80. Cg1
	81. Cg2	82. Cg3	83. Cg4	84. Cg5
	85. Cg6	86. Cg7	87. Cg8	88. Ch0
	89. Ch1	90. Ch2	91. Ch3	92. Ch4
	93. Ch5	94. Ch6	95. Ch7	96. Ch8
	97. Ra0	98. Ra1	99. Ra2	100. Ra3
	101. Ra4	102. Ra5	103. Ra6	104. Rb0
	105. Rb1	106. Rb2	107. Rb3	108. Rb4

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	109. Rb5	110. Rb6	111. Rc0	112. Rc1
	113. Rc2	114. Rc3	115. Rc4	116. Rc5
	117. Rc6	118. Rd0	119. Rd1	120. Rd2
	121. Rd3	122. Rd4	123. Rd5	124. Rd6
	125. Re0	126. Re1	127. Re2	128. Re3
	129. Re4	130. Re5	131. Re6	132. Rf0
	133. Rf1	134. Rf2	135. Rf3	136. Rf4
	137. Rf5	138. Rf6	139. Rg0	140. Rg1
	141. Rg2	142. Rg3	143. Rg4	144. Rg5
	145. Rg6	146. Rh0	147. Rh1	148. Rh2
	149. Rh3	150. Rh4	151. Rh5	152. Rh6
	153. DQa[0]	154. DQa[1]	155. DQa[2]	156. DQa[3]
	157. DQa[4]	158. DQa[5]	159. DQa[6]	160. DQa[7]
	161. DQa[8]	162. DQa[9]	163. DQa[10]	164. DQa[11]
	165. DQa[12]	166. DQa[13]	167. DQa[14]	168. DQa[15]
	169. DQa[16]	170. DQa[17]	171. DQa[18]	172. DQa[19]
	173. DQa[20]	174. DQa[21]	175. DQa[22]	176. DQa[23]
	177. DQa[24]	178. DQa[25]	179. DQa[26]	180. DQa[27]
	181. DQa[28]	182. DQa[29]	183. DQa[30]	184. DQa[31]
	185. DQa[32]	186. DQa[33]	187. DQa[34]	188. DQa[35]
	189. DQa[36]	190. DQa[37]	191. DQa[38]	192. DQa[39]
	193. DQa[40]	194. DQa[41]	195. DQa[42]	196. DQa[43]
	197. DQa[44]	198. DQa[45]	199. DQa[46]	200. DQa[47]
	201. DQa[48]	202. DQa[49]	203. DQa[50]	204. DQa[51]
	205. DQa[52]	206. DQa[53]	207. DQa[54]	208. DQa[55]
	209. DQa[56]	210. DQa[57]	211. DQa[58]	212. DQa[59]
	213. DQa[60]	214. DQa[61]	215. DQa[62]	216. DQa[63]
	217. DQa[64]	218. DQa[65]	219. DQa[66]	220. DQa[67]
	221. DQa[68]	222. DQa[69]	223. DQa[70]	224. DQa[71]
	225. DQa[72]	226. DQa[73]	227. DQa[74]	228. DQa[75]
	229. DQa[76]	230. DQa[77]	231. DQa[78]	232. DQa[79]
	233. DQa[80]	234. DQa[81]	235. DQa[82]	236. DQa[83]
	237. DQa[84]	238. DQa[85]	239. DQa[86]	240. DQa[87]
	241. DQa[88]	242. DQa[89]	243. DQa[90]	244. DQa[91]
	245. DQa[92]	246. DQa[93]	247. DQa[94]	248. DQa[95]
	249. DQa[96]	250. DQa[97]	251. DQa[98]	252. DQa[99]
	253. DQa[100]	254. DQa[101]	255. DQa[102]	256. DQa[103]
	257. DQa[104]	258. DQa[105]	259. DQa[106]	260. DQa[107]
	261. DQa[108]	262. DQa[109]	263. DQa[110]	264. DQa[111]
	265. DQa[112]	266. DQa[113]	267. DQa[114]	268. DQa[115]
	269. DQa[116]	270. DQa[117]	271. DQa[118]	272. DQa[119]
	273. DQa[120]	274. DQa[121]	275. DQa[122]	276. DQa[123]
	277. DQa[124]	278. DQa[125]	279. DQa[126]	280. DQa[127]
	281. DQb[0]	282. DQb[1]	283. DQb[2]	284. DQb[3]

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	285. DQb[4]	286. DQb[5]	287. DQb[6]	288. DQb[7]
	289. DQb[8]	290. DQb[9]	291. DQb[10]	292. DQb[11]
	293. DQb[12]	294. DQb[13]	295. DQb[14]	296. DQb[15]
	297. DQb[16]	298. DQb[17]	299. DQb[18]	300. DQb[19]
	301. DQb[20]	302. DQb[21]	303. DQb[22]	304. DQb[23]
	305. DQb[24]	306. DQb[25]	307. DQb[26]	308. DQb[27]
	309. DQb[28]	310. DQb[29]	311. DQb[30]	312. DQb[31]
	313. DQb[32]	314. DQb[33]	315. DQb[34]	316. DQb[35]
	317. DQb[36]	318. DQb[37]	319. DQb[38]	320. DQb[39]
	321. DQb[40]	322. DQb[41]	323. DQb[42]	324. DQb[43]
	325. DQb[44]	326. DQb[45]	327. DQb[46]	328. DQb[47]
	329. DQb[48]	330. DQb[49]	331. DQb[50]	332. DQb[51]
	333. DQb[52]	334. DQb[53]	335. DQb[54]	336. DQb[55]
	337. DQb[56]	338. DQb[57]	339. DQb[58]	340. DQb[59]
	341. DQb[60]	342. DQb[61]	343. DQb[62]	344. DQb[63]
	345. DQb[64]	346. DQb[65]	347. DQb[66]	348. DQb[67]
	349. DQb[68]	350. DQb[69]	351. DQb[70]	352. DQb[71]
	353. DQb[72]	354. DQb[73]	355. DQb[74]	356. DQb[75]
	357. DQb[76]	358. DQb[77]	359. DQb[78]	360. DQb[79]
	361. DQb[80]	362. DQb[81]	363. DQb[82]	364. DQb[83]
	365. DQb[84]	366. DQb[85]	367. DQb[86]	368. DQb[87]
	369. DQb[88]	370. DQb[89]	371. DQb[90]	372. DQb[91]
	373. DQb[92]	374. DQb[93]	375. DQb[94]	376. DQb[95]
	377. DQb[96]	378. DQb[97]	379. DQb[98]	380. DQb[99]
	381. DQb[100]	382. DQb[101]	383. DQb[102]	384. DQb[103]
	385. DQb[104]	386. DQb[105]	387. DQb[106]	388. DQb[107]
	389. DQb[108]	390. DQb[109]	391. DQb[110]	392. DQb[111]
	393. DQb[112]	394. DQb[113]	395. DQb[114]	396. DQb[115]
	397. DQb[116]	398. DQb[117]	399. DQb[118]	400. DQb[119]
	401. DQb[120]	402. DQb[121]	403. DQb[122]	404. DQb[123]
	405. DQb[124]	406. DQb[125]	407. DQb[126]	408. DQb[127]
	409. DQc[0]	410. DQc[1]	411. DQc[2]	412. DQc[3]
	413. DQc[4]	414. DQc[5]	415. DQc[6]	416. DQc[7]
	417. DQc[8]	418. DQc[9]	419. DQc[10]	420. DQc[11]
	421. DQc[12]	422. DQc[13]	423. DQc[14]	424. DQc[15]
	425. DQc[16]	426. DQc[17]	427. DQc[18]	428. DQc[19]
	429. DQc[20]	430. DQc[21]	431. DQc[22]	432. DQc[23]
	433. DQc[24]	434. DQc[25]	435. DQc[26]	436. DQc[27]
	437. DQc[28]	438. DQc[29]	439. DQc[30]	440. DQc[31]
	441. DQc[32]	442. DQc[33]	443. DQc[34]	444. DQc[35]
	445. DQc[36]	446. DQc[37]	447. DQc[38]	448. DQc[39]
	449. DQc[40]	450. DQc[41]	451. DQc[42]	452. DQc[43]
	453. DQc[44]	454. DQc[45]	455. DQc[46]	456. DQc[47]
	457. DQc[48]	458. DQc[49]	459. DQc[50]	460. DQc[51]

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	461. DQc[52]	462. DQc[53]	463. DQc[54]	464. DQc[55]
	465. DQc[56]	466. DQc[57]	467. DQc[58]	468. DQc[59]
	469. DQc[60]	470. DQc[61]	471. DQc[62]	472. DQc[63]
	473. DQc[64]	474. DQc[65]	475. DQc[66]	476. DQc[67]
	477. DQc[68]	478. DQc[69]	479. DQc[70]	480. DQc[71]
	481. DQc[72]	482. DQc[73]	483. DQc[74]	484. DQc[75]
	485. DQc[76]	486. DQc[77]	487. DQc[78]	488. DQc[79]
	489. DQc[80]	490. DQc[81]	491. DQc[82]	492. DQc[83]
	493. DQc[84]	494. DQc[85]	495. DQc[86]	496. DQc[87]
	497. DQc[88]	498. DQc[89]	499. DQc[90]	500. DQc[91]
	501. DQc[92]	502. DQc[93]	503. DQc[94]	504. DQc[95]
	505. DQc[96]	506. DQc[97]	507. DQc[98]	508. DQc[99]
	509. DQc[100]	510. DQc[101]	511. DQc[102]	512. DQc[103]
	513. DQc[104]	514. DQc[105]	515. DQc[106]	516. DQc[107]
	517. DQc[108]	518. DQc[109]	519. DQc[110]	520. DQc[111]
	521. DQc[112]	522. DQc[113]	523. DQc[114]	524. DQc[115]
	525. DQc[116]	526. DQc[117]	527. DQc[118]	528. DQc[119]
	529. DQc[120]	530. DQc[121]	531. DQc[122]	532. DQc[123]
	533. DQc[124]	534. DQc[125]	535. DQc[126]	536. DQc[127]
	537. DQd[0]	538. DQd[1]	539. DQd[2]	540. DQd[3]
	541. DQd[4]	542. DQd[5]	543. DQd[6]	544. DQd[7]
	545. DQd[8]	546. DQd[9]	547. DQd[10]	548. DQd[11]
	549. DQd[12]	550. DQd[13]	551. DQd[14]	552. DQd[15]
	553. DQd[16]	554. DQd[17]	555. DQd[18]	556. DQd[19]
	557. DQd[20]	558. DQd[21]	559. DQd[22]	560. DQd[23]
	561. DQd[24]	562. DQd[25]	563. DQd[26]	564. DQd[27]
	565. DQd[28]	566. DQd[29]	567. DQd[30]	568. DQd[31]
	569. DQd[32]	570. DQd[33]	571. DQd[34]	572. DQd[35]
	573. DQd[36]	574. DQd[37]	575. DQd[38]	576. DQd[39]
	577. DQd[40]	578. DQd[41]	579. DQd[42]	580. DQd[43]
	581. DQd[44]	582. DQd[45]	583. DQd[46]	584. DQd[47]
	585. DQd[48]	586. DQd[49]	587. DQd[50]	588. DQd[51]
	589. DQd[52]	590. DQd[53]	591. DQd[54]	592. DQd[55]
	593. DQd[56]	594. DQd[57]	595. DQd[58]	596. DQd[59]
	597. DQd[60]	598. DQd[61]	599. DQd[62]	600. DQd[63]
	601. DQd[64]	602. DQd[65]	603. DQd[66]	604. DQd[67]
	605. DQd[68]	606. DQd[69]	607. DQd[70]	608. DQd[71]
	609. DQd[72]	610. DQd[73]	611. DQd[74]	612. DQd[75]
	613. DQd[76]	614. DQd[77]	615. DQd[78]	616. DQd[79]
	617. DQd[80]	618. DQd[81]	619. DQd[82]	620. DQd[83]
	621. DQd[84]	622. DQd[85]	623. DQd[86]	624. DQd[87]
	625. DQd[88]	626. DQd[89]	627. DQd[90]	628. DQd[91]
	629. DQd[92]	630. DQd[93]	631. DQd[94]	632. DQd[95]
	633. DQd[96]	634. DQd[97]	635. DQd[98]	636. DQd[99]

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	637. DQd[100]	638. DQd[101]	639. DQd[102]	640. DQd[103]
	641. DQd[104]	642. DQd[105]	643. DQd[106]	644. DQd[107]
	645. DQd[108]	646. DQd[109]	647. DQd[110]	648. DQd[111]
	649. DQd[112]	650. DQd[113]	651. DQd[114]	652. DQd[115]
	653. DQd[116]	654. DQd[117]	655. DQd[118]	656. DQd[119]
	657. DQd[120]	658. DQd[121]	659. DQd[122]	660. DQd[123]
	661. DQd[124]	662. DQd[125]	663. DQd[126]	664. DQd[127]
	665. DQe[0]	666. DQe[1]	667. DQe[2]	668. DQe[3]
	669. DQe[4]	670. DQe[5]	671. DQe[6]	672. DQe[7]
	673. DQe[8]	674. DQe[9]	675. DQe[10]	676. DQe[11]
	677. DQe[12]	678. DQe[13]	679. DQe[14]	680. DQe[15]
	681. DQe[16]	682. DQe[17]	683. DQe[18]	684. DQe[19]
	685. DQe[20]	686. DQe[21]	687. DQe[22]	688. DQe[23]
	689. DQe[24]	690. DQe[25]	691. DQe[26]	692. DQe[27]
	693. DQe[28]	694. DQe[29]	695. DQe[30]	696. DQe[31]
	697. DQe[32]	698. DQe[33]	699. DQe[34]	700. DQe[35]
	701. DQe[36]	702. DQe[37]	703. DQe[38]	704. DQe[39]
	705. DQe[40]	706. DQe[41]	707. DQe[42]	708. DQe[43]
	709. DQe[44]	710. DQe[45]	711. DQe[46]	712. DQe[47]
	713. DQe[48]	714. DQe[49]	715. DQe[50]	716. DQe[51]
	717. DQe[52]	718. DQe[53]	719. DQe[54]	720. DQe[55]
	721. DQe[56]	722. DQe[57]	723. DQe[58]	724. DQe[59]
	725. DQe[60]	726. DQe[61]	727. DQe[62]	728. DQe[63]
	729. DQe[64]	730. DQe[65]	731. DQe[66]	732. DQe[67]
	733. DQe[68]	734. DQe[69]	735. DQe[70]	736. DQe[71]
	737. DQe[72]	738. DQe[73]	739. DQe[74]	740. DQe[75]
	741. DQe[76]	742. DQe[77]	743. DQe[78]	744. DQe[79]
	745. DQe[80]	746. DQe[81]	747. DQe[82]	748. DQe[83]
	749. DQe[84]	750. DQe[85]	751. DQe[86]	752. DQe[87]
	753. DQe[88]	754. DQe[89]	755. DQe[90]	756. DQe[91]
	757. DQe[92]	758. DQe[93]	759. DQe[94]	760. DQe[95]
	761. DQe[96]	762. DQe[97]	763. DQe[98]	764. DQe[99]
	765. DQe[100]	766. DQe[101]	767. DQe[102]	768. DQe[103]
	769. DQe[104]	770. DQe[105]	771. DQe[106]	772. DQe[107]
	773. DQe[108]	774. DQe[109]	775. DQe[110]	776. DQe[111]
	777. DQe[112]	778. DQe[113]	779. DQe[114]	780. DQe[115]
	781. DQe[116]	782. DQe[117]	783. DQe[118]	784. DQe[119]
	785. DQe[120]	786. DQe[121]	787. DQe[122]	788. DQe[123]
	789. DQe[124]	790. DQe[125]	791. DQe[126]	792. DQe[127]
	793. DQf[0]	794. DQf[1]	795. DQf[2]	796. DQf[3]
	797. DQf[4]	798. DQf[5]	799. DQf[6]	800. DQf[7]
	801. DQf[8]	802. DQf[9]	803. DQf[10]	804. DQf[11]
	805. DQf[12]	806. DQf[13]	807. DQf[14]	808. DQf[15]
	809. DQf[16]	810. DQf[17]	811. DQf[18]	812. DQf[19]

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	813. DQf[20]	814. DQf[21]	815. DQf[22]	816. DQf[23]
	817. DQf[24]	818. DQf[25]	819. DQf[26]	820. DQf[27]
	821. DQf[28]	822. DQf[29]	823. DQf[30]	824. DQf[31]
	825. DQf[32]	826. DQf[33]	827. DQf[34]	828. DQf[35]
	829. DQf[36]	830. DQf[37]	831. DQf[38]	832. DQf[39]
	833. DQf[40]	834. DQf[41]	835. DQf[42]	836. DQf[43]
	837. DQf[44]	838. DQf[45]	839. DQf[46]	840. DQf[47]
	841. DQf[48]	842. DQf[49]	843. DQf[50]	844. DQf[51]
	845. DQf[52]	846. DQf[53]	847. DQf[54]	848. DQf[55]
	849. DQf[56]	850. DQf[57]	851. DQf[58]	852. DQf[59]
	853. DQf[60]	854. DQf[61]	855. DQf[62]	856. DQf[63]
	857. DQf[64]	858. DQf[65]	859. DQf[66]	860. DQf[67]
	861. DQf[68]	862. DQf[69]	863. DQf[70]	864. DQf[71]
	865. DQf[72]	866. DQf[73]	867. DQf[74]	868. DQf[75]
	869. DQf[76]	870. DQf[77]	871. DQf[78]	872. DQf[79]
	873. DQf[80]	874. DQf[81]	875. DQf[82]	876. DQf[83]
	877. DQf[84]	878. DQf[85]	879. DQf[86]	880. DQf[87]
	881. DQf[88]	882. DQf[89]	883. DQf[90]	884. DQf[91]
	885. DQf[92]	886. DQf[93]	887. DQf[94]	888. DQf[95]
	889. DQf[96]	890. DQf[97]	891. DQf[98]	892. DQf[99]
	893. DQf[100]	894. DQf[101]	895. DQf[102]	896. DQf[103]
	897. DQf[104]	898. DQf[105]	899. DQf[106]	900. DQf[107]
	901. DQf[108]	902. DQf[109]	903. DQf[110]	904. DQf[111]
	905. DQf[112]	906. DQf[113]	907. DQf[114]	908. DQf[115]
	909. DQf[116]	910. DQf[117]	911. DQf[118]	912. DQf[119]
	913. DQf[120]	914. DQf[121]	915. DQf[122]	916. DQf[123]
	917. DQf[124]	918. DQf[125]	919. DQf[126]	920. DQf[127]
	921. DQg[0]	922. DQg[1]	923. DQg[2]	924. DQg[3]
	925. DQg[4]	926. DQg[5]	927. DQg[6]	928. DQg[7]
	929. DQg[8]	930. DQg[9]	931. DQg[10]	932. DQg[11]
	933. DQg[12]	934. DQg[13]	935. DQg[14]	936. DQg[15]
	937. DQg[16]	938. DQg[17]	939. DQg[18]	940. DQg[19]
	941. DQg[20]	942. DQg[21]	943. DQg[22]	944. DQg[23]
	945. DQg[24]	946. DQg[25]	947. DQg[26]	948. DQg[27]
	949. DQg[28]	950. DQg[29]	951. DQg[30]	952. DQg[31]
	953. DQg[32]	954. DQg[33]	955. DQg[34]	956. DQg[35]
	957. DQg[36]	958. DQg[37]	959. DQg[38]	960. DQg[39]
	961. DQg[40]	962. DQg[41]	963. DQg[42]	964. DQg[43]
	965. DQg[44]	966. DQg[45]	967. DQg[46]	968. DQg[47]
	969. DQg[48]	970. DQg[49]	971. DQg[50]	972. DQg[51]
	973. DQg[52]	974. DQg[53]	975. DQg[54]	976. DQg[55]
	977. DQg[56]	978. DQg[57]	979. DQg[58]	980. DQg[59]
	981. DQg[60]	982. DQg[61]	983. DQg[62]	984. DQg[63]



**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	985. DQg[64]	986. DQg[65]	987. DQg[66]	988. DQg[67]
	989. DQg[68]	990. DQg[69]	991. DQg[70]	992. DQg[71]
	993. DQg[72]	994. DQg[73]	995. DQg[74]	996. DQg[75]
	997. DQg[76]	998. DQg[77]	999. DQg[78]	1000. DQg[79]
	1001. DQg[80]	1002. DQg[81]	1003. DQg[82]	1004. DQg[83]
	1005. DQg[84]	1006. DQg[85]	1007. DQg[86]	1008. DQg[87]
	1009. DQg[88]	1010. DQg[89]	1011. DQg[90]	1012. DQg[91]
	1013. DQg[92]	1014. DQg[93]	1015. DQg[94]	1016. DQg[95]
	1017. DQg[96]	1018. DQg[97]	1019. DQg[98]	1020. DQg[99]
	1021. DQg[100]	1022. DQg[101]	1023. DQg[102]	1024. DQg[103]
	1025. DQg[104]	1026. DQg[105]	1027. DQg[106]	1028. DQg[107]
	1029. DQg[108]	1030. DQg[109]	1031. DQg[110]	1032. DQg[111]
	1033. DQg[112]	1034. DQg[113]	1035. DQg[114]	1036. DQg[115]
	1037. DQg[116]	1038. DQg[117]	1039. DQg[118]	1040. DQg[119]
	1041. DQg[120]	1042. DQg[121]	1043. DQg[122]	1044. DQg[123]
	1045. DQg[124]	1046. DQg[125]	1047. DQg[126]	1048. DQg[127]
	1049. DQh[0]	1050. DQh[1]	1051. DQh[2]	1052. DQh[3]
	1053. DQh[4]	1054. DQh[5]	1055. DQh[6]	1056. DQh[7]
	1057. DQh[8]	1058. DQh[9]	1059. DQh[10]	1060. DQh[11]
	1061. DQh[12]	1062. DQh[13]	1063. DQh[14]	1064. DQh[15]
	1065. DQh[16]	1066. DQh[17]	1067. DQh[18]	1068. DQh[19]
	1069. DQh[20]	1070. DQh[21]	1071. DQh[22]	1072. DQh[23]
	1073. DQh[24]	1074. DQh[25]	1075. DQh[26]	1076. DQh[27]
	1077. DQh[28]	1078. DQh[29]	1079. DQh[30]	1080. DQh[31]
	1081. DQh[32]	1082. DQh[33]	1083. DQh[34]	1084. DQh[35]
	1085. DQh[36]	1086. DQh[37]	1087. DQh[38]	1088. DQh[39]
	1089. DQh[40]	1090. DQh[41]	1091. DQh[42]	1092. DQh[43]
	1093. DQh[44]	1094. DQh[45]	1095. DQh[46]	1096. DQh[47]
	1097. DQh[48]	1098. DQh[49]	1099. DQh[50]	1100. DQh[51]
	1101. DQh[52]	1102. DQh[53]	1103. DQh[54]	1104. DQh[55]
	1105. DQh[56]	1106. DQh[57]	1107. DQh[58]	1108. DQh[59]
	1109. DQh[60]	1110. DQh[61]	1111. DQh[62]	1112. DQh[63]
	1113. DQh[64]	1114. DQh[65]	1115. DQh[66]	1116. DQh[67]
	1117. DQh[68]	1118. DQh[69]	1119. DQh[70]	1120. DQh[71]
	1121. DQh[72]	1122. DQh[73]	1123. DQh[74]	1124. DQh[75]
	1125. DQh[76]	1126. DQh[77]	1127. DQh[78]	1128. DQh[79]
	1129. DQh[80]	1130. DQh[81]	1131. DQh[82]	1132. DQh[83]
	1133. DQh[84]	1134. DQh[85]	1135. DQh[86]	1136. DQh[87]
	1137. DQh[88]	1138. DQh[89]	1139. DQh[90]	1140. DQh[91]
	1141. DQh[92]	1142. DQh[93]	1143. DQh[94]	1144. DQh[95]
	1145. DQh[96]	1146. DQh[97]	1147. DQh[98]	1148. DQh[99]
	1149. DQh[100]	1150. DQh[101]	1151. DQh[102]	1152. DQh[103]
	1153. DQh[104]	1154. DQh[105]	1155. DQh[106]	1156. DQh[107]

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	1157. DQh[108]	1158. DQh[109]	1159. DQh[110]	1160. DQh[111]
	1161. DQh[112]	1162. DQh[113]	1163. DQh[114]	1164. DQh[115]
	1165. DQh[116]	1166. DQh[117]	1167. DQh[118]	1168. DQh[119]
	1169. DQh[120]	1170. DQh[121]	1171. DQh[122]	1172. DQh[123]
	1173. DQh[124]	1174. DQh[125]	1175. DQh[126]	1176. DQh[127]
	1177. DBIa0	1178. DBIa1	1179. DBIa2	1180. DBIa3
	1181. DBIa4	1182. DBIa5	1183. DBIa6	1184. DBIa7
	1185. DBIa8	1186. DBIa9	1187. DBIa10	1188. DBIa11
	1189. DBIa12	1190. DBIa13	1191. DBIa14	1192. DBIa15
	1193. DBIb0	1194. DBIb1	1195. DBIb2	1196. DBIb3
	1197. DBIb4	1198. DBIb5	1199. DBIb6	1200. DBIb7
	1201. DBIb8	1202. DBIb9	1203. DBIb10	1204. DBIb11
	1205. DBIb12	1206. DBIb13	1207. DBIb14	1208. DBIb15
	1209. DBIc0	1210. DBIc1	1211. DBIc2	1212. DBIc3
	1213. DBIc4	1214. DBIc5	1215. DBIc6	1216. DBIc7
	1217. DBIc8	1218. DBIc9	1219. DBIc10	1220. DBIc11
	1221. DBIc12	1222. DBIc13	1223. DBIc14	1224. DBIc15
	1225. DBId0	1226. DBId1	1227. DBId2	1228. DBId3
	1229. DBId4	1230. DBId5	1231. DBId6	1232. DBId7
	1233. DBId8	1234. DBId9	1235. DBId10	1236. DBId11
	1237. DBId12	1238. DBId13	1239. DBId14	1240. DBId15
	1241. DBIe0	1242. DBIe1	1243. DBIe2	1244. DBIe3
	1245. DBIe4	1246. DBIe5	1247. DBIe6	1248. DBIe7
	1249. DBIe8	1250. DBIe9	1251. DBIe10	1252. DBIe11
	1253. DBIe12	1254. DBIe13	1255. DBIe14	1256. DBIe15
	1257. DBIf0	1258. DBIf1	1259. DBIf2	1260. DBIf3
	1261. DBIf4	1262. DBIf5	1263. DBIf6	1264. DBIf7
	1265. DBIf8	1266. DBIf9	1267. DBIf10	1268. DBIf11
	1269. DBIf12	1270. DBIf13	1271. DBIf14	1272. DBIf15
	1273. DBIg0	1274. DBIg1	1275. DBIg2	1276. DBIg3
	1277. DBIg4	1278. DBIg5	1279. DBIg6	1280. DBIg7
	1281. DBIg8	1282. DBIg9	1283. DBIg10	1284. DBIg11
	1285. DBIg12	1286. DBIg13	1287. DBIg14	1288. DBIg15
	1289. DBIh0	1290. DBIh1	1291. DBIh2	1292. DBIh3
	1293. DBIh4	1294. DBIh5	1295. DBIh6	1296. DBIh7
	1297. DBIh8	1298. DBIh9	1299. DBIh10	1300. DBIh11
	1301. DBIh12	1302. DBIh13	1303. DBIh14	1304. DBIh15
	1305. DMA0	1306. DMA1	1307. DMA2	1308. DMA3
	1309. DMA4	1310. DMA5	1311. DMA6	1312. DMA7
	1313. DMA8	1314. DMA9	1315. DMA10	1316. DMA11
	1317. DMA12	1318. DMA13	1319. DMA14	1320. DMA15
	1321. DMb0	1322. DMb1	1323. DMb2	1324. DMb3
	1325. DMb4	1326. DMb5	1327. DMb6	1328. DMb7

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	1329. DMb8	1330. DMb9	1331. DMb10	1332. DMb11
	1333. DMb12	1334. DMb13	1335. DMb14	1336. DMb15
	1337. DMc0	1338. DMc1	1339. DMc2	1340. DMc3
	1341. DMc4	1342. DMc5	1343. DMc6	1344. DMc7
	1345. DMc8	1346. DMc9	1347. DMc10	1348. DMc11
	1349. DMc12	1350. DMc13	1351. DMc14	1352. DMc15
	1353. DMd0	1354. DMd1	1355. DMd2	1356. DMd3
	1357. DMd4	1358. DMd5	1359. DMd6	1360. DMd7
	1361. DMd8	1362. DMd9	1363. DMd10	1364. DMd11
	1365. DMd12	1366. DMd13	1367. DMd14	1368. DMd15
	1369. DMe0	1370. DMe1	1371. DMe2	1372. DMe3
	1373. DMe4	1374. DMe5	1375. DMe6	1376. DMe7
	1377. DMe8	1378. DMe9	1379. DMe10	1380. DMe11
	1381. DMe12	1382. DMe13	1383. DMe14	1384. DMe15
	1385. DMf0	1386. DMf1	1387. DMf2	1388. DMf3
	1389. DMf4	1390. DMf5	1391. DMf6	1392. DMf7
	1393. DMf8	1394. DMf9	1395. DMf10	1396. DMf11
	1397. DMf12	1398. DMf13	1399. DMf14	1400. DMf15
	1401. DMg0	1402. DMg1	1403. DMg2	1404. DMg3
	1405. DMg4	1406. DMg5	1407. DMg6	1408. DMg7
	1409. DMg8	1410. DMg9	1411. DMg10	1412. DMg11
	1413. DMg12	1414. DMg13	1415. DMg14	1416. DMg15
	1417. DMh0	1418. DMh1	1419. DMh2	1420. DMh3
	1421. DMh4	1422. DMh5	1423. DMh6	1424. DMh7
	1425. DMh8	1426. DMh9	1427. DMh10	1428. DMh11
	1429. DMh12	1430. DMh13	1431. DMh14	1432. DMh15
	1433. PARa0	1434. PARa1	1435. PARa2	1436. PARa3
	1437. PARb0	1438. PARb1	1439. PARb2	1440. PARb3
	1441. PARc0	1442. PARc1	1443. PARc2	1444. PARc3
	1445. PARd0	1446. PARd1	1447. PARd2	1448. PARd3
	1449. PARe0	1450. PARe1	1451. PARe2	1452. PARe3
	1453. PARf0	1454. PARf1	1455. PARf2	1456. PARf3
	1457. PARg0	1458. PARg1	1459. PARg2	1460. PARg3
	1461. PARh0	1462. PARh1	1463. PARh2	1464. PARh3
	1465. DERRa0	1466. DERRa1	1467. DERRa2	1468. DERRa3
	1469. DERRb0	1470. DERRb1	1471. DERRb2	1472. DERRb3
	1473. DERRc0	1474. DERRc1	1475. DERRc2	1476. DERRc3
	1477. DERRd0	1478. DERRd1	1479. DERRd2	1480. DERRd3
	1481. DERRe0	1482. DERRe1	1483. DERRe2	1484. DERRe3
	1485. DERRf0	1486. DERRf1	1487. DERRf2	1488. DERRf3
	1489. DERRg0	1490. DERRg1	1491. DERRg2	1492. DERRg3
	1493. DERRh0	1494. DERRh1	1495. DERRh2	1496. DERRh3
	1497. WDQSa0_t	1498. WDQSa1_t	1499. WDQSa2_t	1500. WDQSa3_t

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	1501. WDQSB0_t	1502. WDQSB1_t	1503. WDQSB2_t	1504. WDQSB3_t
	1505. WDQSC0_t	1506. WDQSC1_t	1507. WDQSC2_t	1508. WDQSC3_t
	1509. WDQSD0_t	1510. WDQSD1_t	1511. WDQSD2_t	1512. WDQSD3_t
	1513. WDQSE0_t	1514. WDQSE1_t	1515. WDQSE2_t	1516. WDQSE3_t
	1517. WDQSF0_t	1518. WDQSF1_t	1519. WDQSF2_t	1520. WDQSF3_t
	1521. WDQSG0_t	1522. WDQSG1_t	1523. WDQSG2_t	1524. WDQSG3_t
	1525. WDQSH0_t	1526. WDQSH1_t	1527. WDQSH2_t	1528. WDQSH3_t
	1529. WDQSA0_c	1530. WDQSA1_c	1531. WDQSA2_c	1532. WDQSA3_c
	1533. WDQSB0_c	1534. WDQSB1_c	1535. WDQSB2_c	1536. WDQSB3_c
	1537. WDQSC0_c	1538. WDQSC1_c	1539. WDQSC2_c	1540. WDQSC3_c
	1541. WDQSD0_c	1542. WDQSD1_c	1543. WDQSD2_c	1544. WDQSD3_c
	1545. WDQSE0_c	1546. WDQSE1_c	1547. WDQSE2_c	1548. WDQSE3_c
	1549. WDQSF0_c	1550. WDQSF1_c	1551. WDQSF2_c	1552. WDQSF3_c
	1553. WDQSG0_c	1554. WDQSG1_c	1555. WDQSG2_c	1556. WDQSG3_c
	1557. WDQSH0_c	1558. WDQSH1_c	1559. WDQSH2_c	1560. WDQSH3_c
	1561. RDQSA0_t	1562. RDQSA1_t	1563. RDQSA2_t	1564. RDQSA3_t
	1565. RDQSB0_t	1566. RDQSB1_t	1567. RDQSB2_t	1568. RDQSB3_t
	1569. RDQSC0_t	1570. RDQSC1_t	1571. RDQSC2_t	1572. RDQSC3_t
	1573. RDQSD0_t	1574. RDQSD1_t	1575. RDQSD2_t	1576. RDQSD3_t
	1577. RDQSE0_t	1578. RDQSE1_t	1579. RDQSE2_t	1580. RDQSE3_t
	1581. RDQSF0_t	1582. RDQSF1_t	1583. RDQSF2_t	1584. RDQSF3_t
	1585. RDQSG0_t	1586. RDQSG1_t	1587. RDQSG2_t	1588. RDQSG3_t
	1589. RDQSH0_t	1590. RDQSH1_t	1591. RDQSH2_t	1592. RDQSH3_t
	1593. RDQSA0_c	1594. RDQSA1_c	1595. RDQSA2_c	1596. RDQSA3_c
	1597. RDQSB0_c	1598. RDQSB1_c	1599. RDQSB2_c	1600. RDQSB3_c
	1601. RDQSC0_c	1602. RDQSC1_c	1603. RDQSC2_c	1604. RDQSC3_c
	1605. RDQSD0_c	1606. RDQSD1_c	1607. RDQSD2_c	1608. RDQSD3_c
	1609. RDQSE0_c	1610. RDQSE1_c	1611. RDQSE2_c	1612. RDQSE3_c
	1613. RDQSF0_c	1614. RDQSF1_c	1615. RDQSF2_c	1616. RDQSF3_c
	1617. RDQSG0_c	1618. RDQSG1_c	1619. RDQSG2_c	1620. RDQSG3_c
	1621. RDQSH0_c	1622. RDQSH1_c	1623. RDQSH2_c	1624. RDQSH3_c
	1625. DA0	1626. DA1	1627. DA2	1628. DA3
	1629. DA4	1630. DA5	1631. DA6	1632. DA7
	1633. DA8	1634. DA9	1635. DA10	1636. DA11
	1637. DA12	1638. DA13	1639. DA14	1640. DA15
	1641. DA16	1642. DA17	1643. DA18	1644. DA19
	1645. DA20	1646. DA21	1647. DA22	1648. DA23
	1649. DA24	1650. DA25	1651. DA26	1652. DA27
	1653. DA28	1654. DA29	1655. DA30	1656. DA31
	1657. DA32	1658. DA33	1659. DA34	1660. DA35
	1661. DA36	1662. DA37	1663. DA38	1664. DA39
	1665. DA40	1666. DA41	1667. DA42	1668. DA43
	1669. DA44	1670. DA45	1671. DA46	1672. DA47

**4.5.3.10.17.1 HBM1, HBM2, and HBM2E Interface Functions (cont'd)**

list of enumerate values (cont.)	1673. DA48	1674. DA49	1675. DA50	1676. DA51
	1677. DA52	1678. DA53	1679. DA54	1680. DA55
	1681. DA56	1682. DA57	1683. DA58	1684. DA59
	1685. RESET_n	1686. WRCK	1687. WRST_n	1688. SelectWIR
	1689. ShiftWR	1690. CaptureWR	1691. UpdateWR	1692. WSI
	1693. WSOa	1694. WSOB	1695. WSOc	1696. WSOd
	1697. WSOe	1698. WSOF	1699. WSOg	1700. WSOH
	1701. RSVD0	1702. RSVD1	1703. RSVD2	1704. RSVD3
	1705. RSVD4	1706. RSVD5	1707. RDa0	1708. RDa1
	1709. RDa2	1710. RDa3	1711. RDa4	1712. RDa5
	1713. RDa6	1714. RDa7	1715. RDb0	1716. RDb1
	1717. RDb2	1718. RDb3	1719. RDb4	1720. RDb5
	1721. RDb6	1722. RDb7	1723. RDc0	1724. RDc1
	1725. RDc2	1726. RDc3	1727. RDc4	1728. RDc5
	1729. RDc6	1730. RDc7	1731. RDd0	1732. RDd1
	1733. RDd2	1734. RDd3	1735. RDd4	1736. RDd5
	1737. RDd6	1738. RDd7	1739. RDe0	1740. RDe1
	1741. RDe2	1742. RDe3	1743. RDe4	1744. RDe5
	1745. RDe6	1746. RDe7	1747. Rdf0	1748. Rdf1
	1749. Rdf2	1750. Rdf3	1751. Rdf4	1752. Rdf5
	1753. Rdf6	1754. Rdf7	1755. RDg0	1756. RDg1
	1757. RDg2	1758. RDg3	1759. RDg4	1760. RDg5
	1761. RDg6	1762. RDg7	1763. RDh0	1764. RDh1
	1765. RDh2	1766. RDh3	1767. RDh4	1768. RDh5
	1769. RDh6	1770. RDh7	1771. RCa	1772. RCb
	1773. RCc	1774. RCd	1775. RCe	1776. RCf
	1777. RCg	1778. RCh	1779. RRa	1780. RRb
	1781. RRc	1782. RRd	1783. RRe	1784. RRf
	1785. RRg	1786. RRh	1787. ARFUa0	1788. ARFUa2
	1789. TEMP0	1790. TEMP1	1791. TEMP2	1792. CATTRIP
	1793. VSS	1794. VDDC	1795. VDDQ	1796. VPP

For more information about the HBM1, HBM2, HBM2E Interfaces, refer to the JEDEC standard JESD235D.

## 4.5.3.10.17.2. HBM3 Interface Functions

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HBM/HBM3			
diagram	<p>The diagram illustrates the relationships between several classes in the HBM3-InterfaceFunctionType hierarchy. The main class, HBM3-InterfaceFunctionType, is associated with HBM3-StandardTerminalNameAssignmentType (1..∞). This assignment type is further associated with HBM3-MandatoryStandardTerminalMappingType (1912) and HBM3-OptionalStandardTerminalMappingType (0..80). Both mapping types are associated with HBM3-MandatoryStandardTerminalNameType and HBM3-OptionalStandardTerminalNameType, which in turn are associated with StandardTerminalName and TerminalMapID attributes.</p>			
type	HBM3-InterfaceFunctionType, HBM3-StandardTerminalNameAssignmentType, HBM3-MandatoryStandardTerminalMappingType, HBM3-MandatoryStandardTerminalNameType, HBM3-OptionalStandardTerminalMappingType, HBM3-OptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	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

**4.5.3.10.17.2 HBM3 Interface Function (cont'd)**

list of enumerate values (cont.)	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
	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

**4.5.3.10.17.2 HBM3 Interface Function (cont'd)**

list of enumerate values (cont.)	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
	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. ARF Ud
	325. ARFUe	326. ARFUf	327. ARFUg	328. ARFUh
	329. ARFUi	330. ARFUj	331. ARF Uk	332. ARF UI
	333. ARFU m	334. ARFU n	335. ARFU o	336. ARFU p
	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]



**4.5.3.10.17.2 HBM3 Interface Function (cont'd)**

list of enumerate values (cont.)	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]
	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]

**4.5.3.10.17.2 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]

**4.5.3.10.17.2 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]

**4.5.3.10.17.2 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. DQl[0]	1058. DQl[1]	1059. DQl[2]	1060. DQl[3]
	1061. DQl[4]	1062. DQl[5]	1063. DQl[6]	1064. DQl[7]
	1065. DQl[8]	1066. DQl[9]	1067. DQl[10]	1068. DQl[11]
	1069. DQl[12]	1070. DQl[13]	1071. DQl[14]	1072. DQl[15]
	1073. DQl[16]	1074. DQl[17]	1075. DQl[18]	1076. DQl[19]
	1077. DQl[20]	1078. DQl[21]	1079. DQl[22]	1080. DQl[23]
	1081. DQl[24]	1082. DQl[25]	1083. DQl[26]	1084. DQl[27]
	1085. DQl[28]	1086. DQl[29]	1087. DQl[30]	1088. DQl[31]
	1089. DQl[32]	1090. DQl[33]	1091. DQl[34]	1092. DQl[35]
	1093. DQl[36]	1094. DQl[37]	1095. DQl[38]	1096. DQl[39]
	1097. DQl[40]	1098. DQl[41]	1099. DQl[42]	1100. DQl[43]
	1101. DQl[44]	1102. DQl[45]	1103. DQl[46]	1104. DQl[47]
	1105. DQl[48]	1106. DQl[49]	1107. DQl[50]	1108. DQl[51]
	1109. DQl[52]	1110. DQl[53]	1111. DQl[54]	1112. DQl[55]
	1113. DQl[56]	1114. DQl[57]	1115. DQl[58]	1116. DQl[59]
	1117. DQl[60]	1118. DQl[61]	1119. DQl[62]	1120. DQl[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]

**4.5.3.10.17.2 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]

**4.5.3.10.17.2 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. DBIa0	1378. DBIa1	1379. DBIa2	1380. DBIa3
	1381. DBIa4	1382. DBIa5	1383. DBIa6	1384. DBIa7
	1385. DBIb0	1386. DBIb1	1387. DBIb2	1388. DBIb3
	1389. DBIb4	1390. DBIb5	1391. DBIb6	1392. DBIb7
	1393. DBIc0	1394. DBIc1	1395. DBIc2	1396. DBIc3
	1397. DBIc4	1398. DBIc5	1399. DBIc6	1400. DBIc7
	1401. DBId0	1402. DBId1	1403. DBId2	1404. DBId3
	1405. DBId4	1406. DBId5	1407. DBId6	1408. DBId7
	1409. DBIe0	1410. DBIe1	1411. DBIe2	1412. DBIe3
	1413. DBIe4	1414. DBIe5	1415. DBIe6	1416. DBIe7
	1417. DBIf0	1418. DBIf1	1419. DBIf2	1420. DBIf3
	1421. DBIf4	1422. DBIf5	1423. DBIf6	1424. DBIf7
	1425. DBIg0	1426. DBIg1	1427. DBIg2	1428. DBIg3
	1429. DBIg4	1430. DBIg5	1431. DBIg6	1432. DBIg7
	1433. DBIh0	1434. DBIh1	1435. DBIh2	1436. DBIh3
	1437. DBIh4	1438. DBIh5	1439. DBIh6	1440. DBIh7
	1441. DBIi0	1442. DBIi1	1443. DBIi2	1444. DBIi3
	1445. DBIi4	1446. DBIi5	1447. DBIi6	1448. DBIi7
	1449. DBIj0	1450. DBIj1	1451. DBIj2	1452. DBIj3
	1453. DBIj4	1454. DBIj5	1455. DBIj6	1456. DBIj7
	1457. DBIk0	1458. DBIk1	1459. DBIk2	1460. DBIk3
	1461. DBIk4	1462. DBIk5	1463. DBIk6	1464. DBIk7
	1465. DBIl0	1466. DBIl1	1467. DBIl2	1468. DBIl3
	1469. DBIl4	1470. DBIl5	1471. DBIl6	1472. DBIl7

**4.5.3.10.17.2 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. ECCI0	1550. ECCI1	1551. ECCI2	1552. ECCI3
	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. DPARIO

**4.5.3.10.17.2 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. WDQ Sb0_t	1715. WDQSc0_t	1716. WDQ Sd0_t
	1717. WDQSe0_t	1718. WDQ Sf0_t	1719. WDQ Sg0_t	1720. WDQ Sh0_t
	1721. WDQSi0_t	1722. WDQ Sj0_t	1723. WDQ Sk0_t	1724. WDQ Sl0_t
	1725. WDQSm0_t	1726. WDQ Sn0_t	1727. WDQ So0_t	1728. WDQ Sp0_t
	1729. WDQSa1_t	1730. WDQ Sb1_t	1731. WDQSc1_t	1732. WDQ Sd1_t
	1733. WDQSe1_t	1734. WDQ Sf1_t	1735. WDQ Sg1_t	1736. WDQ Sh1_t
	1737. WDQSi1_t	1738. WDQ Sj1_t	1739. WDQ Sk1_t	1740. WDQ Sl1_t
	1741. WDQSm1_t	1742. WDQ Sn1_t	1743. WDQ So1_t	1744. WDQ Sp1_t
	1745. WDQSa0_c	1746. WDQ Sb0_c	1747. WDQSc0_c	1748. WDQ Sd0_c
	1749. WDQSe0_c	1750. WDQ Sf0_c	1751. WDQ Sg0_c	1752. WDQ Sh0_c
	1753. WDQSi0_c	1754. WDQ Sj0_c	1755. WDQ Sk0_c	1756. WDQ Sl0_c
	1757. WDQSm0_c	1758. WDQ Sn0_c	1759. WDQ So0_c	1760. WDQ Sp0_c
	1761. WDQSa1_c	1762. WDQ Sb1_c	1763. WDQSc1_c	1764. WDQ Sd1_c
	1765. WDQSe1_c	1766. WDQ Sf1_c	1767. WDQ Sg1_c	1768. WDQ Sh1_c
	1769. WDQSi1_c	1770. WDQ Sj1_c	1771. WDQ Sk1_c	1772. WDQ Sl1_c
	1773. WDQSm1_c	1774. WDQ Sn1_c	1775. WDQ So1_c	1776. WDQ Sp1_c
	1777. RDQSa0_t	1778. RDQ Sb0_t	1779. RDQSc0_t	1780. RDQ Sd0_t
	1781. RDQSe0_t	1782. RDQ Sf0_t	1783. RDQ Sg0_t	1784. RDQ Sh0_t
	1785. RDQSi0_t	1786. RDQ Sj0_t	1787. RDQ Sk0_t	1788. RDQ Sl0_t
	1789. RDQSm0_t	1790. RDQ Sn0_t	1791. RDQ So0_t	1792. RDQ Sp0_t
	1793. RDQSa1_t	1794. RDQ Sb1_t	1795. RDQSc1_t	1796. RDQ Sd1_t
	1797. RDQSe1_t	1798. RDQ Sf1_t	1799. RDQ Sg1_t	1800. RDQ Sh1_t
	1801. RDQSi1_t	1802. RDQ Sj1_t	1803. RDQ Sk1_t	1804. RDQ Sl1_t
	1805. RDQSm1_t	1806. RDQ Sn1_t	1807. RDQ So1_t	1808. RDQ Sp1_t
	1809. RDQSa0_c	1810. RDQ Sb0_c	1811. RDQSc0_c	1812. RDQ Sd0_c
	1813. RDQSe0_c	1814. RDQ Sf0_c	1815. RDQ Sg0_c	1816. RDQ Sh0_c



## 4.5.3.10.17.2 HBM3 Interface Function (cont'd)

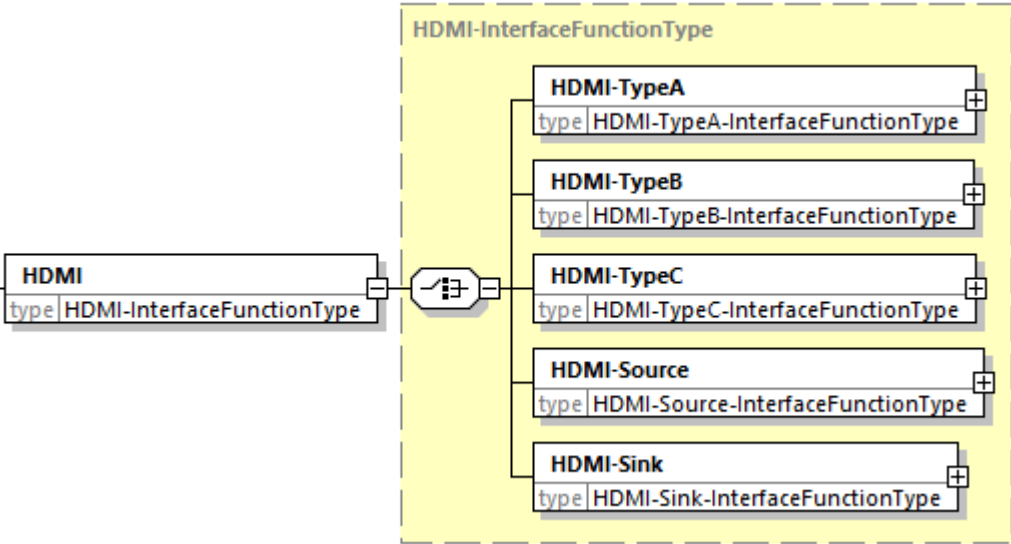
list of enumerate values (cont.)	1817. RDQSi0_c	1818. RDQSi0_c	1819. RDQSk0_c	1820. RDQSi0_c
	1821. RDQSm0_c	1822. RDQSn0_c	1823. RDQSo0_c	1824. RDQSp0_c
	1825. RDQSa1_c	1826. RDQSa1_c	1827. RDQSc1_c	1828. RDQSi1_c
	1829. RDQSe1_c	1830. RDQSi1_c	1831. RDQSi1_c	1832. RDQSi1_c
	1833. RDQSi1_c	1834. RDQSi1_c	1835. RDQSi1_c	1836. RDQSi1_c
	1837. RDQSi1_c	1838. RDQSi1_c	1839. RDQSi1_c	1840. RDQSi1_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. WSOa	1891. WSOc	1892. WSOd
	1893. WSOe	1894. WSOe	1895. WSOg	1896. WSOh
	1897. WSOi	1898. WSOj	1899. WSOk	1900. WSOl
	1901. WSOm	1902. WSON	1903. WSOo	1904. WSOo
	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. RDe0	7. RDg0	8. RDh0
	9. RDi0	10. RDj0	11. RDk0	12. RDi0
	13. RDm0	14. RDn0	15. RDo0	16. RDp0
	17. RDa1	18. RDb1	19. RDc1	20. RDd1
	21. RDe1	22. RDe1	23. RDg1	24. RDh1
	25. RDi1	26. RDj1	27. RDk1	28. RDi1
	29. RDm1	30. RDn1	31. RDo1	32. RDp1
	33. RDa2	34. RDb2	35. RDc2	36. RDd2
	37. RDe2	38. RDe2	39. RDg2	40. RDh2
	41. RDi2	42. RDj2	43. RDk2	44. RDi2
	45. RDm2	46. RDn2	47. RDo2	48. RDp2
	49. RDa3	50. RDb3	51. RDc3	52. RDd3
	53. RDe3	54. RDe3	55. RDg3	56. RDh3
	57. RDi3	58. RDj3	59. RDk3	60. RDi3
	61. RDm3	62. RDn3	63. RDo3	64. RDp3
	65. RAa	66. RAb	67. RAc	68. RAd

4.5.3.10.17.2 HBM3 Interface Function (cont'd)

list of enumerate values  (cont.)	69. RAe	70. RAf	71. RAg	72. RAh
	73. RAi	74. RAj	75. RAk	76. RAl
	77. RAm	78. RAn	79. RAo	80. RAp

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

4.5.3.10.18. HDMI Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI		
diagram			
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.

## 4.5.3.10.18.1. HDMI-TypeA

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-TypeA			
diagram				
type	HDMI-TypeA-InterfaceFunctionType, HDMI-TypeA-StandardTerminalNameAssignmentType, HDMI-TypeA-StandardTerminalMappingType, HDMI-TypeA-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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		

## 4.5.3.10.18.2. HDMI-TypeB

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-TypeB			
diagram				
type	HDMI-TypeB-InterfaceFunctionType, HDMI-TypeB-StandardTerminalNameAssignmentType, HDMI-TypeB-StandardTerminalMappingType, HDMI-TypeB-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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		

## 4.5.3.10.18.3. HDMI-TypeC

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-TypeC			
diagram				
type	HDMI-TypeC-InterfaceFunctionType, HDMI-TypeC-StandardTerminalNameAssignmentType, HDMI-TypeC-StandardTerminalMappingType, HDMI-TypeC-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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	

## 4.5.3.10.18.4. HDMI-Source

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-Source			
diagram				
type	HDMI-Source-InterfaceFunctionType, HDMI-SourceStandardTerminalNameAssignmentType, HDMI-SourceStandardTerminalMappingType, HDMI-SourceStandardTerminalNameType.			
list of enumerate values	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		

## 4.5.3.10.18.5. HDMI-Sink

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI/HDMI-Sink			
diagram				
type	HDMI-Sink-InterfaceFunctionType, HDMI-SinkStandardTerminalNameAssignmentType, HDMI-SinkStandardTerminalMappingType, HDMI-SinkStandardTerminalNameType.			
list of enumerate values	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		

**4.5.3.10.19. HSI Interface**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HSI</b>			
diagram				
type	<b>HSI-InterfaceFunctionType, HSI-StandardTerminalNameAssignmentType, HSI-MandatoryStandardTerminalMappingType, HSI-MandatoryStandardTerminalNameType, HSI-OptionalStandardTerminalMappingType, HSI-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. CADATA</b>	<b>2. CAFLAG</b>	<b>3. ACREADY</b>	<b>4. ACDATA</b>
	<b>5. ACFLAG</b>	<b>6. CAREADY</b>		
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. CAWAKE</b>	<b>2. ACWAKE</b>		

For more information about the HSI Interface, refer to the MIPI Alliance standard Specification for High-Speed Synchronous Serial Interface Version 1.01.

**4.5.3.10.20. HTI Interface Function**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HTI</b>			
diagram				
type	<b>HTI-InterfaceFunctionType, HTI-StandardTerminalNameAssignmentType, HTI-StandardTerminalMappingType, HTI-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Lane1</b>	<b>2. Lane2</b>	<b>3. Lane3</b>	<b>4. Lane4</b>
	<b>5. Lane5</b>	<b>6. Lane6</b>		

For more information about the HTI Interface, refer to the MIPI Alliance standard Specification for High-Speed Trace Interface (HTI) Version 1.1.

#### 4.5.3.10.21. HTlv1 Interface Function

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HTlv1</b>			
diagram				
type	<b>HTlv1-InterfaceFunctionType, HTlv1-StandardTerminalNameAssignmentType, HTlv1-StandardTerminalMappingType, HTlv1-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Lane1</b>	<b>2. Lane2</b>	<b>3. Lane3</b>	<b>4. Lane4</b>
	<b>5. Lane5</b>	<b>6. Lane6</b>	<b>7. Lane7</b>	<b>8. Lane8</b>

#### 4.5.3.10.22. I2C Interface Function

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/I2C</b>			
diagram				
type	<b>I2C-InterfaceFunctionType, I2C-StandardTerminalNameAssignmentType, I2C-StandardTerminalMappingType, I2C-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. SCL</b>	<b>2. SDA</b>		

**4.5.3.10.23. I3C Interface Function**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/I3C</b>			
diagram				
type	<b>I3CType, I3C-StandardTerminalNameAssignmentType, I3C-StandardTerminalMappingType, I3C-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. SDA</b>	<b>2. SCL</b>		

**4.5.3.10.24. LLI-Serial Interface Function**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/LLI-Serial</b>			
diagram				
type	<b>LLI-Serial-InterfaceFunctionType, LLI-Serial-StandardTerminalNameAssignmentType, LLI-Serial-StandardTerminalMappingType, LLI-Serial-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. TXDP</b>	<b>2. TXDN</b>	<b>3. RXDP</b>	<b>4. RXDN</b>

For more information about the LLI-Serial Interface, refer to the MIPI Alliance standard Specification for Low Latency Interface (LLI) Version 2.1.



#### 4.5.3.10.25. LVSTL06 Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/LVSTL06			
diagram				
type	LVSTL06-InterfaceFunctionType, LVSTL06-StandardTerminalNameAssignmentType, LVSTL06-StandardTerminalMappingType, LVSTL06-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. VDDQ	2. VSSQ	3. DQ	

For more information about the LVSTL06 Interface, refer to the JEDEC Standard JESD8-29.

#### 4.5.3.10.26. MMC Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/MultiMediaCard			
diagram				
type	MultiMediaCard-InterfaceFunctionsType, eMMC-InterfaceFunctionType, MMC-InterfaceFunctionType, SPI-Mode-InterfaceFunctionType.			

#### 4.5.3.10.26.1. eMMC Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MultiMediaCard/MMC			
diagram				
type	eMMC-InterfaceFunctionType, eMMC-StandardTerminalNameAssignmentType, eMMC-StandardTerminalMappingType, eMMC-StandardTerminalNameType			
list of enumerate values	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.

#### 4.5.3.10.26.2. MMC Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MultiMediaCardMMC			
diagram				
type	MMC-InterfaceFunctionType, MMC-StandardTerminalNameAssignmentType, MMC-StandardTerminalMappingType, MMC-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	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.

4.5.3.10.26.3. SPI-Mode Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MultiMediaCard/SPI-Mode			
diagram	<p>The diagram illustrates the SPI-Mode interface structure. It shows a sequence of components: SPI-Mode-InterfaceFunctionType (type SPI-Mode-InterfaceFunctionType) is connected to StandardTerminalNameAssignmentType (type SPI-Mode-StandardTerminalNameAssignmentType) via a 1..∞ relationship. This is followed by a Mapping (type SPI-Mode-StandardTerminalMappingType) with a 7 relationship. The Mapping is connected to SPI-Mode-StandardTerminalMappingType (type SPI-Mode-StandardTerminalMappingType) via a 7 relationship. This type contains two sub-elements: StandardTerminalName (type SPI-Mode-StandardTerminalNameType) and TerminalMapID (type xs:string). A constraints box is also shown.</p>			
type	SPI-Mode-InterfaceFunctionType, SPI-Mode-StandardTerminalNameAssignmentType, SPI-Mode-StandardTerminalMappingType, SPI-Mode-StandardTerminalNameType			
list of enumerat e values	Mapping/StandardTerminalName			
	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.

4.5.3.10.27. MII Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/HDMI
diagram	<p>The diagram illustrates the structure of the MII-InterfaceFunctionsType. It is a container type that holds eight sub-elements, each representing a different interface function. The sub-elements are CGMII, GMII, MII, RGMII, RMII, SMII, XGMII, and XLGMII. Each sub-element is associated with a specific InterfaceFunctionType (e.g., CGMII-InterfaceFunctionType, GMII-InterfaceFunctionType, etc.). The diagram shows a box labeled 'MII' with 'type MII-InterfaceFunctionsType' connected to a dashed yellow box labeled 'MII-InterfaceFunctionsType'. Inside this box are eight sub-elements, each with a '+' icon in the top right corner, indicating multiplicity.</p>
type	MII-InterfaceFunctionsType, CGMII-InterfaceFunctionType, GMII-InterfaceFunctionType, MII-InterfaceFunctionType, RGMII-InterfaceFunctionType, RMII-InterfaceFunctionType, SMII-InterfaceFunctionType, XGMII-InterfaceFunctionType, XLGMII-InterfaceFunctionType.

## 4.5.3.10.27.1. CGMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/CGMII			
diagram				
type	CGMIIType, CGMII-StandardTerminalNameAssignmentType, CGMII-StandardTerminalMappingType, CGMII-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	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]

**4.5.3.10.27.1 CGMII – Mandatory Mapping (cont'd)**

list of enumerate values  (cont.)	<a href="#">105. RXD[22]</a>	<a href="#">106. RXD[23]</a>	<a href="#">107. RXD[24]</a>	<a href="#">108. RXD[25]</a>
	<a href="#">109. RXD[26]</a>	<a href="#">110. RXD[27]</a>	<a href="#">111. RXD[28]</a>	<a href="#">112. RXD[29]</a>
	<a href="#">113. RXD[30]</a>	<a href="#">114. RXD[31]</a>	<a href="#">115. RXD[32]</a>	<a href="#">116. RXD[33]</a>
	<a href="#">117. RXD[34]</a>	<a href="#">118. RXD[35]</a>	<a href="#">119. RXD[36]</a>	<a href="#">120. RXD[37]</a>
	<a href="#">121. RXD[38]</a>	<a href="#">122. RXD[39]</a>	<a href="#">123. RXD[40]</a>	<a href="#">124. RXD[41]</a>
	<a href="#">125. RXD[42]</a>	<a href="#">126. RXD[43]</a>	<a href="#">127. RXD[44]</a>	<a href="#">128. RXD[45]</a>
	<a href="#">129. RXD[46]</a>	<a href="#">130. RXD[47]</a>	<a href="#">131. RXD[48]</a>	<a href="#">132. RXD[49]</a>
	<a href="#">133. RXD[50]</a>	<a href="#">134. RXD[51]</a>	<a href="#">135. RXD[52]</a>	<a href="#">136. RXD[53]</a>
	<a href="#">137. RXD[54]</a>	<a href="#">138. RXD[55]</a>	<a href="#">139. RXD[56]</a>	<a href="#">140. RXD[57]</a>
	<a href="#">141. RXD[58]</a>	<a href="#">142. RXD[59]</a>	<a href="#">143. RXD[60]</a>	<a href="#">144. RXD[61]</a>
	<a href="#">145. RXD[62]</a>	<a href="#">146. RXD[63]</a>		

For more information about the CGMII Interface, refer to the IEEE standard IEEE 802.3ba-2010.

## 4.5.3.10.27.2. GMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/GMII		
diagram			
type	<b>GMIIType, GMII-StandardTerminalNameAssignmentType,</b> <b>GMII-MandatoryStandardTerminalMappingType, GMII-MandatoryStandardTerminalNameType,</b> <b>GMII-OptionalStandardTerminalMappingType, GMII-OptionalStandardTerminalNameType</b>		
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>		
	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	
	<b>OptionalMapping/StandardTerminalName</b>		
	1. TX_CLK		

For more information about the GMII Interface, refer to the IEEE standard IEEE 802.3z-1998.

4.5.3.10.27.3. MII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/MII			
diagram	<p>The diagram illustrates the MII interface structure. It starts with a box labeled 'MII' of type 'MII-InterfaceFunctionType'. This is followed by a box labeled 'StandardTerminalNameAssign...' of type 'MII-StandardTerminalNameAssi...'. This is then followed by a box labeled 'MandatoryMapping' of type 'MII-MandatoryStandardTerminal...'. This box is connected to a box labeled 'OptionalMapping' of type 'MII-OptionalStandardTerminalM...'. The 'MandatoryMapping' box is also connected to a box labeled 'MII-MandatoryStandardTerminalMappingType', which contains a 'StandardTerminalName' attribute of type 'MII-MandatoryStandardTerminal...' and a 'TerminalMapID' attribute of type 'xs:string'. The 'OptionalMapping' box is connected to a box labeled 'MII-OptionalStandardTerminalMappingType', which contains a 'StandardTerminalName' attribute of type 'MII-OptionalStandardTerminalNa...' and a 'TerminalMapID' attribute of type 'xs:string'. A 'constraints' box is also shown at the bottom.</p>			
type	MIIType, RMII-StandardTerminalNameAssignmentType, MII-MandatoryStandardTerminalMappingType, MII-MandatoryStandardTerminalNameType, MII-OptionalStandardTerminalMappingType, MII-OptionalStandardTerminalNameType.			
list of enumerate values	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.

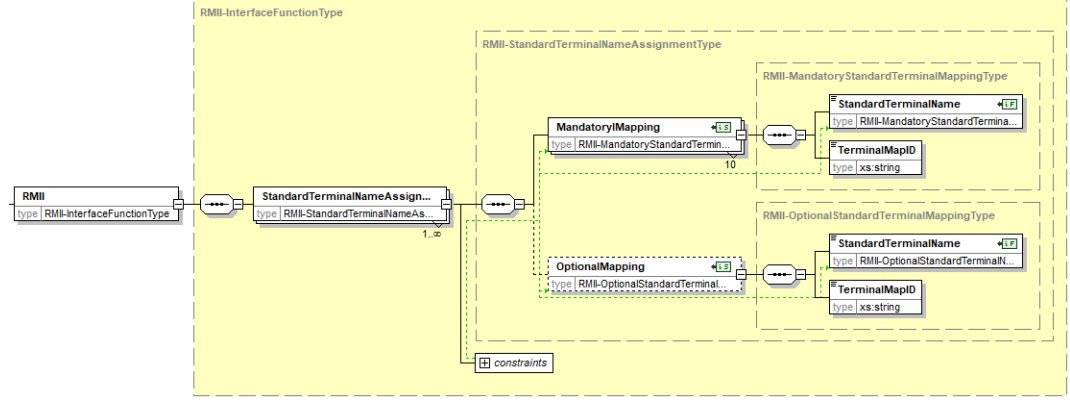


## 4.5.3.10.27.4. RGMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/RGMII			
diagram				
type	<b>RGMIIType, RGMII-StandardTerminalNameAssignmentType,</b> <b>RGMII-StandardTerminalMappingType, RGMII-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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. RD3	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](http://www.hp.com/rnd/pdfs/RGMIIv2_0_final_hp.pdf).

4.5.3.10.27.5. RMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/RMII			
diagram				
type	RMIIType, RMII-StandardTerminalNameAssignmentType, RMII-MandatoryStandardTerminalMappingType, RMII-MandatoryStandardTerminalNameType, RMII-OptionalStandardTerminalMappingType, RMII-OptionalStandardTerminalNameType.			
list of enumerate values	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](https://en.wikipedia.org/wiki/Media-independent_interface#RMII).

4.5.3.10.27.6. SMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/SMII			
diagram	<p>The diagram illustrates the SMII structure. It shows a sequence of elements: SMII (type SMII-InterfaceFunctionType) connected to StandardTerminalNameAssign... (type SMII-StandardTerminalNameAs...) with a multiplicity of 1..∞. This is connected to Mapping (type SMII-StandardTerminalMapping...) with a multiplicity of 4. The Mapping is connected to StandardTerminalName (type SMII-StandardTerminalNameType) with a multiplicity of 4. The StandardTerminalName is connected to TerminalMapID (type xs:string) with a multiplicity of 4. A constraints box is also shown.</p>			
type	SMIIType, SMII-StandardTerminalNameAssignmentType, SMII-StandardTerminalMappingType, SMII-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	1. RX	2. TX	3. SYNC	4. CLOCK

**4.5.3.10.27.7. XGMII**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/XGMII</b>			
diagram				
type	<b>XGMIIType, XGMII-StandardTerminalNameAssignmentType, XGMII-StandardTerminalMappingType, XGMII-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.

## 4.5.3.10.27.8. XLGMII

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/MII/XLGMII			
diagram				
type	XLGMIIType, XLGMII-StandardTerminalNameAssignmentType, XLGMII-StandardTerminalMappingType, XLGMII-StandardTerminalNameType			
list of enumerat e values	Mapping/StandardTerminalName			
	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]

**4.5.3.10.27.8 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.

**4.5.3.10.28. OIF-CEI-04.0 Interface Function**

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/OIF-CEI-04.0			
diagram				
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	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.

**4.5.3.10.29. PCIe Interface Function**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe</a>
diagram	<p>The diagram illustrates the PCIe Interface Function structure. A main block labeled <b>PCIe</b> (type: <code>PCIe-InterfaceFunctionType</code>) is connected to a dashed yellow box labeled <b>PCIe-InterfaceFunctionType</b>. Inside this box, there are eight sub-blocks, each with a '+' icon indicating it is an array element:</p> <ul style="list-style-type: none"> <li><b>PCIe-x1</b> (type: <code>PCIe-x1-InterfaceFunctionType</code>)</li> <li><b>PCIe-x2</b> (type: <code>PCIe-x2-InterfaceFunctionType</code>)</li> <li><b>PCIe-x4</b> (type: <code>PCIe-x4-InterfaceFunctionType</code>)</li> <li><b>PCIe-x8</b> (type: <code>PCIe-x8-InterfaceFunctionType</code>)</li> <li><b>PCIe-x16</b> (type: <code>PCIe-x16-InterfaceFunctionType</code>)</li> <li><b>PCIe-x32</b> (type: <code>PCIe-x32-InterfaceFunctionType</code>)</li> <li><b>SFF-8639Connector</b> (type: <code>SFF-8639ConnectorInterfaceF...</code>)</li> <li><b>ATX-PowerConnector150W</b> (type: <code>ATX-PowerConnector150W-Int...</code>)</li> <li><b>AuxiliaryPowerConnector2x4</b> (type: <code>AuxiliaryPowerConnector2x4-I...</code>)</li> </ul>
type	<a href="#">PCIe-InterfaceFunctionType</a> , <a href="#">PCIe-x1-InterfaceFunctionType</a> , <a href="#">PCIe-x2-InterfaceFunctionType</a> , <a href="#">PCIe-x4-InterfaceFunctionType</a> , <a href="#">PCIe-x8-InterfaceFunctionType</a> , <a href="#">PCIe-x16-InterfaceFunctionType</a> , <a href="#">PCIe-x32-InterfaceFunctionType</a> , <a href="#">SFF-8639ConnectorInterfaceFunctionType</a> , <a href="#">ATX-PowerConnector150W-InterfaceFunctionType</a> , <a href="#">AuxiliaryPowerConnector2x4-InterfaceFunctionType</a> .

For more information about the PCIe Interfaces, refer to the PCI-SIG standard PCI Express Card Electromechanical Specification Rev 2. 0.

4.5.3.10.29.1. PCIe-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x1			
diagram	<p>The diagram illustrates the UML class structure for PCIe-x1. It shows a hierarchy where <b>PCIe-x1-InterfaceFunctionType</b> (type: PCIe-x1-InterfaceFunctionType) is associated with <b>StandardTerminalNameAssignmentType</b> (type: PCIe-x1-StandardTerminalNameAssignmentType) with a cardinality of 1..*. This assignment type is further associated with <b>MandatoryMapping</b> (type: PCIe-x1-MandatoryStandardTerminalMappingType) and <b>OptionalMapping</b> (type: PCIe-x1-OptionalStandardTerminalMappingType). The <b>MandatoryMapping</b> type has a cardinality of 6 and is associated with <b>StandardTerminalName</b> (type: PCIe-x1-MandatoryStandardTerminalNameType) and <b>TerminalMapID</b> (type: xs:string). The <b>OptionalMapping</b> type has a cardinality of 0..5 and is associated with <b>StandardTerminalName</b> (type: PCIe-x1-OptionalStandardTerminalNameType) and <b>TerminalMapID</b> (type: xs:string). A <b>constraints</b> box is also present.</p>			
type	PCIe-x1-InterfaceFunctionType, PCIe-x1-StandardTerminalNameAssignmentType, PCIe-x1-MandatoryStandardTerminalMappingType, PCIe-x1-MandatoryStandardTerminalNameType, PCIe-x1-OptionalStandardTerminalMappingType, PCIe-x1-OptionalStandardTerminalNameType.			
list of enumerate values	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#			



## 4.5.3.10.29.2. PCIe-x2

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x2			
diagram				
type	<b>PCIe-x2-InterfaceFunctionType, PCIe-x2-StandardTerminalNameAssignmentType,</b> <b>PCIe-x2-MandatoryStandardTerminalMappingType, PCIe-x2-MandatoryStandardTerminalNameType,</b> <b>PCIe-x2-OptionalStandardTerminalMappingType, PCIe-x2-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	1. REFCLK+	2. REFCLK-	3. PETp0	4. PETp1
	5. PETn0	6. PETn1	7. PERp0	8. PERp1
	9. PERn0	10. PERn1		
	<b>OptionalMapping/StandardTerminalName</b>			
	1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#
	5. PRSNT2#			

4.5.3.10.29.3. PCIe-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x4			
diagram	<p>The diagram illustrates the UML structure for PCIe-x4 types. It shows a hierarchy where <b>PCIe-x4-InterfaceFunctionType</b> (type: PCIe-x4-InterfaceFunctionType) is associated with <b>StandardTerminalNameAssign...</b> (type: PCIe-x4-StandardTerminalNameAssign...). This is further associated with <b>MandatoryMapping</b> (type: PCIe-x4-MandatoryStandardTerminalMappingType) and <b>OptionalMapping</b> (type: PCIe-x4-OptionalStandardTerminalMappingType). Both mapping types contain <b>StandardTerminalName</b> (type: PCIe-x4-MandatoryStandardTerminalNameType or PCIe-x4-OptionalStandardTerminalNameType) and <b>TerminalMapID</b> (type: xs:string). A 'constraints' box is also present.</p>			
type	PCIe-x4-InterfaceFunctionType, PCIe-x4-StandardTerminalNameAssignmentType, PCIe-x4-MandatoryStandardTerminalMappingType, PCIe-x4-MandatoryStandardTerminalNameType, PCIe-x4-OptionalStandardTerminalMappingType, PCIe-x4-OptionalStandardTerminalNameType.			
list of enumerate values	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#			

## 4.5.3.10.29.4. PCIe-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x8			
diagram				
type	<b>PCIe-x8-InterfaceFunctionType, PCIe-x8-StandardTerminalNameAssignmentType, PCIe-x8-MandatoryStandardTerminalMappingType, PCIe-x8-MandatoryStandardTerminalNameType, PCIe-x8-OptionalStandardTerminalMappingType, PCIe-x8-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	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		
	<b>OptionalMapping/StandardTerminalName</b>			
	1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#
	5. PRSNT2#			

## 4.5.3.10.29.5. PCIe-x16

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/PCIe/PCIe-x16			
diagram	<p>The diagram illustrates the UML structure for PCIe-x16 types. It shows a hierarchy where <b>PCIe-x16-InterfaceFunctionType</b> (type: PCIe-x16-InterfaceFunctionType) is associated with <b>StandardTerminalNameAssignmentType</b> (type: PCIe-x16-StandardTerminalNameAssignmentType) via a 1..∞ relationship. <b>StandardTerminalNameAssignmentType</b> is further associated with <b>MandatoryMapping</b> (type: PCIe-x16-MandatoryStandardTerminalMappingType) and <b>OptionalMapping</b> (type: PCIe-x16-OptionalStandardTerminalMappingType) via a 0..5 relationship. <b>MandatoryMapping</b> is associated with <b>StandardTerminalName</b> (type: PCIe-x16-MandatoryStandardTerminalNameType) and <b>TerminalMapID</b> (type: xs:string). <b>OptionalMapping</b> is associated with <b>StandardTerminalName</b> (type: PCIe-x16-OptionalStandardTerminalNameType) and <b>TerminalMapID</b> (type: xs:string). A <b>constraints</b> box is also present.</p>			
type	<b>PCIe-x16-InterfaceFunctionType, PCIe-x16-StandardTerminalNameAssignmentType, PCIe-x16-MandatoryStandardTerminalMappingType, PCIe-x16-MandatoryStandardTerminalNameType, PCIe-x16-OptionalStandardTerminalMappingType, PCIe-x16-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	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		
	<b>OptionalMapping/StandardTerminalName</b>			
	1. WAKE#	2. CLKREQ#	3. PERST#	4. PRSNT1#
	5. PRSNT2#			

## 4.5.3.10.29.6. 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	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

**4.5.3.10.29.6 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#			

## 4.5.3.10.29.7. SFF-8639Connector

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/PCIe/SFF-8639Connector		
diagram			
type	SFF-8639ConnectorInterfaceFunctionType, SFF-8639ConnectorStandardTerminalNameAssignmentType, SFF-8639ConnectorMandatoryStandardTerminalMappingType, SFF-8639ConnectorMandatoryStandardTerminalNameType, SFF-8639ConnectorOptionalStandardTerminalMappingType, SFF-8639ConnectorOptionalStandardTerminalNameType.		
list of enumerate values	MandatoryMapping/StandardTerminalName		
	1. WAKE#	2. PWRDIS	3. IfDet#
	5. PRSNT#	6. ACTIVITY#	7. +12 V Precharge
	9. HPT0	10. PETp1	11. PETn1
	13. PERP1	14. PETp2	15. PETn2
	17. PERp2	18. REFCLKB+	19. REFCLKB-
	21. PERST#	22. REFCLK+	23. REFCLK-
	25. PETn0	26. PERn0	27. PERp0
	29. PETp3	30. PETn3	31. PERn3
	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

**4.5.3.10.29.8. ATX-PowerConnector150W**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/PCIe/ATX-PowerConnector150W</b>			
diagram				
type	<b>ATX-PowerConnector150W-InterfaceFunctionType, ATX-PowerConnector150W-StandardTerminalNameAssignmentType, ATX-PowerConnector150W-StandardTerminalMappingType, ATX-PowerConnector150W-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. +12V</b>	<b>2. Ground</b>	<b>3. Sense</b>	

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

**4.5.3.10.29.9. AuxiliaryPowerConnector2x4**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/PCIe/AuxiliaryPowerConnector2x4.</b>			
diagram				
type	<b>AuxiliaryPowerConnector2x4-InterfaceFunctionType, AuxiliaryPowerConnector2x4-StandardTerminalNameAssignmentType, AuxiliaryPowerConnector2x4-StandardTerminalMappingType, AuxiliaryPowerConnector2x4-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. +12V</b>	<b>2. Sense0</b>	<b>3. Sense1</b>	<b>4. Ground</b>

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.



#### 4.5.3.10.30. Cabling PCIe Interface Function

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCle</a>
diagram	
type	<a href="#">CablingPCle-InterfaceFunctionType</a> , <a href="#">CablingPCle-x1-InterfaceFunctionType</a> , <a href="#">CablingPCle-x2-InterfaceFunctionType</a> , <a href="#">CablingPCle-x4-InterfaceFunctionType</a> , <a href="#">CablingPCle-x8-InterfaceFunctionType</a> , <a href="#">CablingPCle-x16-InterfaceFunctionType</a> , <a href="#">CablingPCle-x32-InterfaceFunctionType</a> .

For more information about the PCIe Interfaces, refer to the PCI-SIG standard PCI Express Card Electromechanical Specification Revision 2.0.

##### 4.5.3.10.30.1. CablingPCle-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCle/CablingPCle-x1			
diagram	<p>The diagram illustrates the hierarchical structure of the CablingPCle-x1-InterfaceFunctionType. It is composed of several nested types and a mapping:</p> <ul style="list-style-type: none"><li><b>CablingPCle-x1-InterfaceFunctionType</b> (type: CablingPCle-x1-InterfaceFunctionType) is the root type.</li><li>It contains a <b>StandardTerminalNameAssignmentType</b> (type: CablingPCle-x1-StandardTerminalNameAssignmentType) with a cardinality of 1..∞.</li><li>The <b>StandardTerminalNameAssignmentType</b> contains a <b>Mapping</b> (type: CablingPCle-x1-StandardTerminalMappingType) with a cardinality of 11.</li><li>The <b>Mapping</b> type is associated with a <b>StandardTerminalNameType</b> (type: CablingPCle-x1-StandardTerminalNameType) and a <b>TerminalMapID</b> (type: xs:string).</li><li>A <b>constraints</b> box is also shown, indicating additional constraints on the structure.</li></ul>			
type	CablingPCle-x1-InterfaceFunctionType, CablingPCle-x1-StandardTerminalNameAssignmentType, CablingPCle-x1-StandardTerminalMappingType, CablingPCle-x1-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. SB_RTN	2. CREFCLKp	3. CREFCLKn	4. CPWRON
	5. CWAKE#	6. CPRSNT#	7. CPERST#	8. PETp0
	9. PETn0	10. PERp0	11. PERn0	

4.5.3.10.30.2. CablingPCle-x4

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCle/CablingPCle-x4			
diagram	<p>The diagram illustrates the structure of the CablingPCle-x4 entity. It is connected to a StandardTerminalNameAssignmentType (1..20), which is connected to a Mapping (23), which is connected to a StandardTerminalMappingType. The StandardTerminalMappingType is connected to a StandardTerminalName (type   CablingPCle-x4-StandardTermin...) and a TerminalMapID (type   xs:string). A constraints box is also shown.</p>			
type	CablingPCle-x4-InterfaceFunctionType, CablingPCle-x4-StandardTerminalNameAssignmentType, CablingPCle-x4-StandardTerminalMappingType, CablingPCle-x4-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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	

## 4.5.3.10.30.3. CablingPCle-x8

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCle/CablingPCle-x8			
diagram				
type	CablingPCle-x8-InterfaceFunctionType, CablingPCle-x8-StandardTerminalNameAssignmentType, CablingPCle-x8-StandardTerminalMappingType, CablingPCle-x8-StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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	

**4.5.3.10.30.4. CablingPCle-x16**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/CablingPCle/CablingPCle-x16</b>			
diagram				
type	<b>CablingPCle-x16-InterfaceFunctionType, CablingPCle-x16-StandardTerminalNameAssignmentType, CablingPCle-x16-StandardTerminalMappingType, CablingPCle-x16-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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		

**4.5.3.10.31. A-PHY**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/A-PHY</b>		
diagram	<p>The diagram illustrates the structure of the A-PHY type. It is a nested sequence of types: A-PHY-InterfaceFunctionType, followed by A-PHY-StandardTerminalNameAssignmentType (containing StandardTerminalNameAssign...), and then A-PHY-StandardTerminalMappingType (containing Mapping). The Mapping type further contains StandardTerminalName and TerminalMapID. Multiplicities are indicated: 1..∞ for StandardTerminalNameAssign... and 3 for Mapping.</p>		
type	<b>A-PHY-InterfaceFunctionType, A-PHY-StandardTerminalNameAssignmentType, A-PHY-StandardTerminalMappingType, A-PHY-StandardTerminalNameType</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	<b>1. DP</b>	<b>2. DN</b>	<b>3. GND</b>

For more information about the A-PHY Interface, refer to the IEEE standard IEEE 2977-2021.

**4.5.3.10.32. BoW-PHY**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/BoW-PHY</b>		
diagram	<p>The diagram illustrates the structure of the BoW-PHY type. It is a nested sequence of types: BoW-PHY-InterfaceFunctionType, followed by BoW-PHY-StandardTerminalNameAssignmentType (containing StandardTerminalNameAssign...), and then a choice between BoW-PHY-MandatoryStandardTerminalMappingType (containing MandatoryMapping) and BoW-PHY-OptionalStandardTerminalMappingType (containing OptionalMapping). Both mapping types contain StandardTerminalName and TerminalMapID. Multiplicities are indicated: 1..∞ for StandardTerminalNameAssign..., 18 for MandatoryMapping, and 0..2 for OptionalMapping.</p>		
type	<b>BoW-PHY-InterfaceFunctionType, BoW-PHY-StandardTerminalNameAssignmentType, BoW-PHY-MandatoryStandardTerminalMappingType, BoW-PHY-MandatoryStandardTerminalNameType, BoW-PHY-OptionalStandardTerminalMappingType, BoW-PHY-OptionalStandardTerminalNameType.</b>		
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>		
	<b>1. CLK+</b>	<b>2. CLK-</b>	<b>3. D[0]</b>
	<b>5. D[2]</b>	<b>6. D[3]</b>	<b>7. D[4]</b>
	<b>9. D[6]</b>	<b>10. D[7]</b>	<b>11. D[8]</b>
	<b>13. D[10]</b>	<b>14. D[11]</b>	<b>15. D[12]</b>
	<b>17. D[14]</b>	<b>18. D[15]</b>	
	<b>OptionalMapping/StandardTerminalName</b>		
	<b>1. FEC</b>	<b>2. AUX</b>	

4.5.3.10.32. BoW - PHY (cont'd)

For more information about the BoW-PHY Interface, refer to the OPEN Compute Project standard Bunch of Wires (BoW) PHY Specification.

4.5.3.10.33. C-PHY Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/C-PHY
diagram	
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.

## 4.5.3.10.33.1. C-PHY-x1

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x1			
diagram				
type	C-PHY-x1-InterfaceFunctionType, C-PHY-x1-StandardTerminalNameAssignmentType, C-PHY-x1-StandardTerminalMappingType, C-PHY-x1-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	1. A	2. B	3. C	

## 4.5.3.10.33.2. C-PHY-x2

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x2			
diagram				
type	C-PHY-x2-InterfaceFunctionType, C-PHY-x2-StandardTerminalNameAssignmentType, C-PHY-x2-StandardTerminalMappingType, C-PHY-x2-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	1. A1	2. B1	3. C1	4. A2
	5. B2	6. C2		

**4.5.3.10.33.3. C-PHY-x3**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x3</b>			
diagram	<p>The diagram illustrates the structure of C-PHY-x3. It shows a sequence of types: C-PHY-x3-InterfaceFunctionType, C-PHY-x3-StandardTerminalNameAssignmentType, C-PHY-x3-StandardTerminalMappingType, and C-PHY-x3-StandardTerminalNameType. The C-PHY-x3-StandardTerminalNameAssignmentType is associated with C-PHY-x3-StandardTerminalMappingType via a 'Mapping' type. The C-PHY-x3-StandardTerminalMappingType is associated with C-PHY-x3-StandardTerminalNameType via a 'StandardTerminalName' type. The C-PHY-x3-StandardTerminalNameType is associated with C-PHY-x3-StandardTerminalMappingType via a 'TerminalMapID' type. The diagram also includes a 'constraints' box and a 'list of enumerate values' table.</p>			
type	<b>C-PHY-x3-InterfaceFunctionType, C-PHY-x3-StandardTerminalNameAssignmentType, C-PHY-x3-StandardTerminalMappingType, C-PHY-x3-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	1. A1	2. B1	3. C1	4. A2
	5. B2	6. C2	7. A3	8. B3
	9. C3			

**4.5.3.10.33.4. C-PHY-x4**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x4</b>			
diagram	<p>The diagram illustrates the structure of C-PHY-x4. It shows a sequence of types: C-PHY-x4-InterfaceFunctionType, C-PHY-x4-StandardTerminalNameAssignmentType, C-PHY-x4-StandardTerminalMappingType, and C-PHY-x4-StandardTerminalNameType. The C-PHY-x4-StandardTerminalNameAssignmentType is associated with C-PHY-x4-StandardTerminalMappingType via a 'Mapping' type. The C-PHY-x4-StandardTerminalMappingType is associated with C-PHY-x4-StandardTerminalNameType via a 'StandardTerminalName' type. The C-PHY-x4-StandardTerminalNameType is associated with C-PHY-x4-StandardTerminalMappingType via a 'TerminalMapID' type. The diagram also includes a 'constraints' box and a 'list of enumerate values' table.</p>			
type	<b>C-PHY-x4-InterfaceFunctionType, C-PHY-x4-StandardTerminalNameAssignmentType, C-PHY-x4-StandardTerminalMappingType, C-PHY-x4-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	1. A1	2. B1	3. C1	4. A2
	5. B2	6. C2	7. A3	8. B3
	9. C3	10. A4	11. B4	12. C4



**4.5.3.10.33.5. C-PHY-x5**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x5</b>			
diagram				
type	<b>C-PHY-x5-InterfaceFunctionType, C-PHY-x5-StandardTerminalNameAssignmentType, C-PHY-x5-StandardTerminalMappingType, C-PHY-x5-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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	

**4.5.3.10.33.6. C-PHY-x6**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/C-PHY/C-PHY-x6</b>			
diagram				
type	<b>C-PHY-x6-InterfaceFunctionType, C-PHY-x6-StandardTerminalNameAssignmentType, C-PHY-x6-StandardTerminalMappingType, C-PHY-x6-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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		

4.5.3.10.34. D-PHY Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/D-PHY
diagram	<p>The diagram illustrates the D-PHY interface function structure. On the left, a box labeled 'D-PHY' with the attribute 'type   D-PHY-InterfaceFunctionType' is connected to a dashed yellow box labeled 'D-PHY-InterfaceFunctionType'. Inside this dashed box, there are eight stacked boxes representing individual interface functions: 'D-PHY-x1' through 'D-PHY-x8'. Each of these boxes has a 'type' attribute pointing to its corresponding 'InterfaceFunctionType' (e.g., 'D-PHY-x1-InterfaceFunctionType').</p>
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.

**4.5.3.10.34.1. D-PHY-x1**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x1</b>		
diagram			
type	<b>D-PHY-x1-InterfaceFunctionType, D-PHY-x1-StandardTerminalNameAssignmentType, D-PHY-x1-StandardTerminalMappingType, D-PHY-x1-StandardTerminalNameType</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	<b>1. CLOCK+</b>	<b>2. CLOCK-</b>	<b>3. DATA0+</b>
			<b>4. DATA0-</b>

**4.5.3.10.34.2. D-PHY-x2**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x2</b>		
diagram			
type	<b>D-PHY-x2-InterfaceFunctionType, D-PHY-x2-StandardTerminalNameAssignmentType, D-PHY-x2-StandardTerminalMappingType, D-PHY-x2-StandardTerminalNameType</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	<b>1. CLOCK+</b>	<b>2. CLOCK-</b>	<b>3. DATA0+</b>
	<b>5. DATA1+</b>	<b>6. DATA1-</b>	
			<b>4. DATA0-</b>

**4.5.3.10.34.3. D-PHY-x3**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x3</b>		
diagram			
type	<b>D-PHY-x3-InterfaceFunctionType, D-PHY-x3-StandardTerminalNameAssignmentType, D-PHY-x3-StandardTerminalMappingType, D-PHY-x3-StandardTerminalNameType</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	<b>1. CLOCK+</b>	<b>2. CLOCK-</b>	<b>3. DATA0+</b>
	<b>5. DATA1+</b>	<b>6. DATA1-</b>	<b>7. DATA2+</b>
			<b>8. DATA2-</b>

**4.5.3.10.34.4. D-PHY-x4**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x4</b>		
diagram			
type	<b>D-PHY-x4-InterfaceFunctionType, D-PHY-x4-StandardTerminalNameAssignmentType, D-PHY-x4-StandardTerminalMappingType, D-PHY-x4-StandardTerminalNameType</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	<b>1. CLOCK+</b>	<b>2. CLOCK-</b>	<b>3. DATA0+</b>
	<b>5. DATA1+</b>	<b>6. DATA1-</b>	<b>7. DATA2+</b>
	<b>9. DATA3+</b>	<b>10. DATA3-</b>	<b>11.</b>
			<b>12.</b>

## 4.5.3.10.34.5. D-PHY-x5

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x5			
diagram				
type	D-PHY-x5-InterfaceFunctionType, D-PHY-x5-StandardTerminalNameAssignmentType, D-PHY-x5-StandardTerminalMappingType, D-PHY-x5-StandardTerminalNameType			
list of enumerate values	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-

## 4.5.3.10.34.6. D-PHY-x6

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x6			
diagram				
type	D-PHY-x6-InterfaceFunctionType, D-PHY-x6-StandardTerminalNameAssignmentType, D-PHY-x6-StandardTerminalMappingType, D-PHY-x6-StandardTerminalNameType			
list of enumerate values	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-		

**4.5.3.10.34.7. D-PHY-x7**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x7</b>			
diagram				
type	<b>D-PHY-x7-InterfaceFunctionType, D-PHY-x7-StandardTerminalNameAssignmentType, D-PHY-x7-StandardTerminalMappingType, D-PHY-x7-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. CLOCK+</b>	<b>2. CLOCK-</b>	<b>3. DATA0+</b>	<b>4. DATA0-</b>
	<b>5. DATA1+</b>	<b>6. DATA1-</b>	<b>7. DATA2+</b>	<b>8. DATA2-</b>
	<b>9. DATA3+</b>	<b>10. DATA3-</b>	<b>11. DATA4+</b>	<b>12. DATA4-</b>
	<b>13. DATA5+</b>	<b>14. DATA5-</b>	<b>15. DATA6+</b>	<b>16. DATA6-</b>

**4.5.3.10.34.8. D-PHY-x8**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/D-PHY/D-PHY-x8</b>			
diagram				
type	<b>D-PHY-x8-InterfaceFunctionType, D-PHY-x8-StandardTerminalNameAssignmentType, D-PHY-x8-StandardTerminalMappingType, D-PHY-x8-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. CLOCK+</b>	<b>2. CLOCK-</b>	<b>3. DATA0+</b>	<b>4. DATA0-</b>
	<b>5. DATA1+</b>	<b>6. DATA1-</b>	<b>7. DATA2+</b>	<b>8. DATA2-</b>
	<b>9. DATA3+</b>	<b>10. DATA3-</b>	<b>11. DATA4+</b>	<b>12. DATA4-</b>
	<b>13. DATA5+</b>	<b>14. DATA5-</b>	<b>15. DATA6+</b>	<b>16. DATA6-</b>
	<b>17. DATA7+</b>	<b>18. DATA7-</b>		

**4.5.3.10.35. 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.

**4.5.3.10.36. OpenHBI**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/OpenHBI</b>			
diagram				
type	<b>OpenHBI-InterfaceFunctionType, OpenHBI -StandardTerminalNameAssignmentType, OpenHBI -StandardTerminalMappingType, OpenHBI -StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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



#### 4.5.3.10.37. 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.

#### 4.5.3.10.38. 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.			

4.5.3.10.38.1. RBDP

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/RadioFrontEnd/RBDP			
diagram	<p>The diagram illustrates the RBDP interface structure. It shows a sequence of types: RBDP (type RBDP-InterfaceFunctionType), StandardTerminalNameAssignmentType (type RBDP-StandardTerminalNameA, 1..∞), RBDP-MandatoryStandardTermin... (type RBDP-MandatoryStandardTermin..., 14), and RBDP-OptionalStandardTermin... (type RBDP-OptionalStandardTermin..., 0..2). The mandatory and optional types are further detailed with StandardTerminalName and TerminalMapID attributes.</p>			
type	RBDP-InterfaceFunctionType, RBDP-StandardTerminalNameAssignmentType, RBDP-MandatoryStandardTerminalMappingType, RBDP-MandatoryStandardTerminalNameType, RBDP-OptionalStandardTerminalMappingType, RBDP-OptionalStandardTerminalNameType.			
list of enumerate values	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.

**4.5.3.10.38.2. RF-BB**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/RadioFrontEnd/RF-BB</b>			
diagram				
type	<b>RF-BB-InterfaceFunctionType, RF-BB-StandardTerminalNameAssignmentType, RF-BB-MandatoryStandardTerminalMappingType, RF-BB-MandatoryStandardTerminalNameType, RF-BB-OptionalStandardTerminalMappingType, RF-BB-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. CLK</b>	<b>2. CLKN</b>	<b>3. TX</b>	<b>4. TXN</b>
	<b>5. RX</b>	<b>6. RXN</b>		
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. Tx_Clock</b>	<b>2. Tx_ClockN</b>		

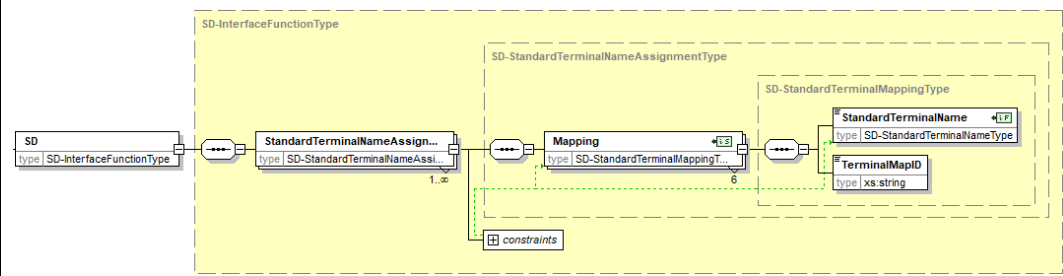
For more information about the RBDP Interface, refer to JEDEC Standard JESD96A.

**4.5.3.10.39. RFFE**

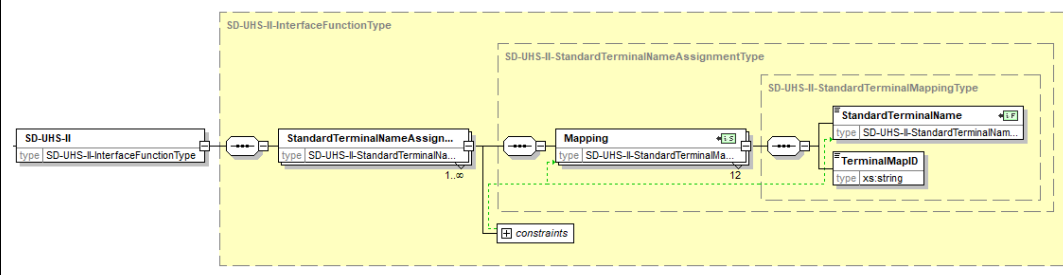
path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/RFFE</b>			
diagram				
type	<b>RFFE-InterfaceFunctionType, RFFE-StandardTerminalNameAssignmentType, RFFE-StandardTerminalMappingType, RFFE-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. SCLK</b>	<b>2. SDATA</b>	<b>3. VIO</b>	

For more information about the M-PHY Interfaces, refer to the MIPI Alliance standard Specification for RF Front-End Control Interface Version 1.10.

4.5.3.10.40. SD

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SD			
diagram				
type	SD-InterfaceFunctionType, SD-StandardTerminalNameAssignmentType, SD-StandardTerminalMappingType, SD-StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	1. CLK	2. DAT[0]	3. DAT[1]	4. DAT[2]
	5. DAT[3]	6. CMD		

4.5.3.10.41. SD-UHS-II

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SD-UHS-II			
diagram				
type	SD-UHS-II-InterfaceFunctionType, SD-UHS-II-StandardTerminalNameAssignmentType, SD-UHS-II-StandardTerminalMappingType, SD-UHS-II-StandardTerminalNameType			
list of enumerate values	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-

#### 4.5.3.10.42. Serial Interface Function

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface</a>
diagram	
type	<a href="#">SerialInterfaceFunctionType</a> , <a href="#">SI-InterfaceFunctionType</a> , <a href="#">SPI-InterfaceFunctionType</a> , <a href="#">eSPI-InterfaceFunctionType</a> , <a href="#">xSPI-InterfaceFunctionType</a> , <a href="#">SPD5118Hub-InterfaceFunctionType</a> .

##### 4.5.3.10.42.1. SI

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/SI</a>			
diagram				
type	<a href="#">SI-InterfaceFunctionType</a> , <a href="#">SI-StandardTerminalNameAssignmentType</a> , <a href="#">SI-StandardTerminalMappingType</a> , <a href="#">SI-StandardTerminalNameType</a> .			
list of enumerate values	<a href="#">Mapping/StandardTerminalName</a>			
	1. TX	2. RX	3. SYSREF	4. Device Clock
	5. SYNC~			

For more information about the RBDP Interface, refer to JEDEC Standard JESD204C.1.

**4.5.3.10.42.2. SPI**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/SPI</b>			
diagram	<p>The diagram illustrates the SPI interface structure. It starts with a box labeled 'SPI' of type 'SPI-InterfaceFunctionType'. This is connected to a box labeled 'StandardTerminalNameAssign...' of type 'SPI-StandardTerminalNameAssi...'. This box is then connected to a box labeled 'Mapping' of type 'SPI-StandardTerminalMappingT...'. The 'Mapping' box is connected to a box labeled 'StandardTerminalName' of type 'SPI-StandardTerminalNameType'. The 'StandardTerminalName' box is connected to a box labeled 'TerminalMapID' of type 'xs:string'. A dashed box labeled 'constraints' is also shown.</p>			
type	<b>SPI-InterfaceFunctionType, SPI-StandardTerminalNameAssignmentType, SPI-StandardTerminalMappingType, SPI-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. SCK</b>	<b>2. MOSI</b>	<b>3. MISO</b>	<b>4. ~SS</b>

**4.5.3.10.42.3. eSPI**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/eSPI</b>			
diagram	<p>The diagram illustrates the eSPI interface structure. It starts with a box labeled 'eSPI' of type 'eSPI-InterfaceFunctionType'. This is connected to a box labeled 'StandardTerminalNameAssign...' of type 'eSPI-StandardTerminalNameAs...'. This box is then connected to a box labeled 'MandatoryMapping' of type 'eSPI-MandatoryStandardTermin...'. The 'MandatoryMapping' box is connected to a box labeled 'StandardTerminalName' of type 'eSPI-MandatoryStandardTermin...'. The 'StandardTerminalName' box is connected to a box labeled 'TerminalMapID' of type 'xs:string'. A dashed box labeled 'OptionalMapping' is also shown, connected to the 'MandatoryMapping' box. The 'OptionalMapping' box is connected to a box labeled 'StandardTerminalName' of type 'eSPI-OptionalStandardTermin...'. The 'StandardTerminalName' box is connected to a box labeled 'TerminalMapID' of type 'xs:string'.</p>			
type	<b>eSPI-InterfaceFunctionType, eSPI-StandardTerminalNameAssignmentType, eSPI-MandatoryStandardTerminalMappingType, eSPI-MandatoryStandardTerminalNameType, eSPI-OptionalStandardTerminalMappingType, eSPI-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. eSPI Reset#</b>	<b>2. Chip Select#</b>	<b>3. Serial Clock</b>	<b>4. Alert#</b>
	<b>5. I/O[0]</b>			
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>6. I/O[1]</b>	<b>7. I/O[2]</b>	<b>8. I/O[3]</b>	

For more information about the eSPI Interface, refer to INTEL Specification 327432-004 Enhanced Serial Peripheral Interface (eSPI) Revision 1.0.

## 4.5.3.10.42.4. xSPI

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/xSPI</b>			
diagram				
type	<b>xSPI-InterfaceFunctionType, xSPI-StandardTerminalNameAssignmentType, xSPI-StandardTerminalMappingType, xSPI-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	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.

## 4.5.3.10.42.5. SPD5118Hub

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SerialInterface/SPD5118Hub</b>			
diagram				
type	<b>SPD5118Hub-InterfaceFunctionType, SPD5118Hub-StandardTerminalNameAssignmentType, SPD5118Hub-StandardTerminalMappingType, SPD5118Hub-StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	1. 1.8VVDSPD	2. VDDIO	3. HSCL	4. HSDA
	5. LSCL	6. LSDA		

For more information about the RBDP Interface, refer to JEDEC Standard JESD300-5B.

**4.5.3.10.43. SLIMbus**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SLIMbus</b>			
diagram				
type	SLIMbus-InterfaceFunctionType, SLIMbus-StandardTerminalNameAssignmentType, SLIMbus-MandatoryStandardTerminalMappingType, SLIMbus-MandatoryStandardTerminalNameType, SLIMbus-OptionalStandardTerminalMappingType, SLIMbus-OptionalStandardTerminalNameType.			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	1. CLK	2. DP0		
	<b>OptionalMapping/StandardTerminalName</b>			
	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.

**4.5.3.10.44. SMB**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/SMB</b>			
diagram				
type	SMB-InterfaceFunctionType, SMB-StandardTerminalNameAssignmentType, SMB-StandardTerminalMappingType, SMB-StandardTerminalNameType.			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	1. SMBCLK	2. SMBDAT		



#### 4.5.3.10.44 SMB (cont'd)

For more information about the SMB Interfaces, refer to the System Management Interface Forum SMI for System Management Bus (SMBus) Specification Version 3.0.

#### 4.5.3.10.45. SoundWire Interface Function

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/SoundWire</b>			
diagram				
type	<b>SoundWire-InterfaceFunctionType, SoundWire -StandardTerminalNameAssignmentType, SoundWire -StandardTerminalMappingType, SoundWire -StandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Clock</b>	<b>2. Data</b>		

For more information about the SoundWire Interface, refer to the MIPI Alliance standard Specification for SoundWire Version 1.1.

#### 4.5.3.10.46. SPMI

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/SPMI</b>			
diagram				
type	<b>SPMI-InterfaceFunctionType, SPMI-StandardTerminalNameAssignmentType, SPMI-StandardTerminalMappingType, SPMI-StandardTerminalNameType</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. SCLK</b>	<b>2. SDATA</b>		

For more information about the SPMI Interfaces, refer to the MIPI Alliance standard System Power Management Interface V2.0.

**4.5.3.10.47. UART**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UART</b>			
diagram				
type	<b>UART-InterfaceFunctionType, UART-StandardTerminalNameAssignmentType, UART-MandatoryStandardTerminalMappingType, UART-MandatoryStandardTerminalNameType, UART-OptionalStandardTerminalMappingType, UART-OptionalStandardTerminalNameType</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	1. RX	2. TX		
	<b>OptionalMapping/StandardTerminalName</b>			
	1. CTS	2. RTS	3. DSR	4. RI
	5. DCD	6. DTR		

**4.5.3.10.48. UCle**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UCle-InterfaceFunction</b>			
diagram				
type	<b>UCle-InterfaceFunctionType, UCle-AdvancedType, UCleStandard_x16Type, UCleStandard_x32Type.</b>			

For more information about the UCle Interfaces, refer to the UCle Specification Universal Chiplet Interconnect Express (UCle) Specification Revision 1.0

## 4.5.3.10.48.1. UCle-Advanced

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/UCle-InterfaceFunction/UCle-Advanced			
diagram				
type	UCle-AdvancedType, UCle-AdvancedStandardTerminalNameAssignmentType, UCle-AdvancedStandardTerminalMappingType, UCle-AdvancedStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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]

**4.5.3.10.48.1 UCle- Advanced Interface Function Mandatory Mapping (cont'd)**

list of enumerate values  (cont.)	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
	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			

**4.5.3.10.48.2. UCleStandard\_x16**

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/UCle-InterfaceFunction/UCleStandard_x16			
diagram				
type	UCleStandard_x16Type, UCleStandard_x16StandardTerminalNameAssignmentType, UCleStandard_x16StandardTerminalMappingType, UCleStandard_x16StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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	

## 4.5.3.10.48.3. UCleStandard\_x32

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/UCle-InterfaceFunction/ UCleStandard_x32			
diagram				
type	UCleStandard_x32Type, UCleStandard_x32StandardTerminalNameAssignmentType, UCleStandard_x32StandardTerminalMappingType, UCleStandard_x32StandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	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	

**4.5.3.10.49. UniPro**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/ UniPro</b>			
diagram				
type	<b>UniPro-InterfaceFunctionType, UniPro-StandardTerminalNameAssignmentType, UniPro-MandatoryStandardTerminalMappingType, UniPro-MandatoryStandardTerminalNameType, UniPro-OptionalStandardTerminalMappingType, UniPro-OptionalStandardTerminalNameType</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. T_SAP</b>	<b>2. DME_SAP</b>		
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>3. N_SAP</b>	<b>4. DL_SAP</b>	<b>5. PA_SAP</b>	<b>6. PHY_SAP</b>
	<b>7. T_LM_SAP</b>	<b>8. N_LM_SAP</b>	<b>9. DL_LM_SAP</b>	<b>10. PA_LM_SAP</b>

For more information about the SPMI Interfaces, refer to the MIPI Alliance standard Specification for Unified Protocol (UniPro) Version 1.8.

**4.5.3.10.50. Universal Flash Storage Interface Function**

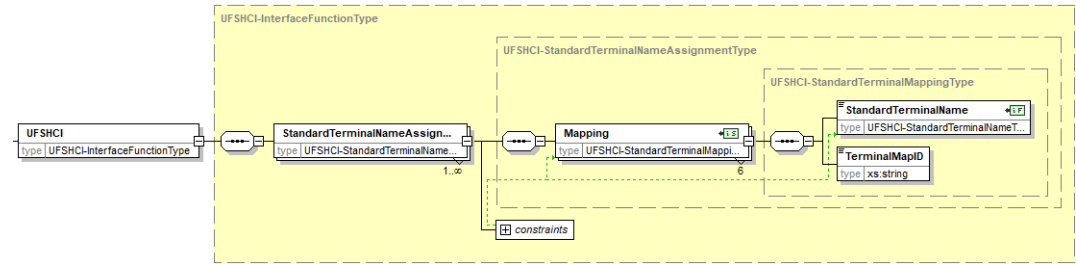
path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UFS</b>			
diagram				
type	<b>UniversalFlashStorage-InterfaceFunctionType, USB2.0-InterfaceFunctionType, USB3.0-InterfaceFunctionType.</b>			

## 4.5.3.10.50.1. UFS

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UFSHCI			
diagram				
type	UFS-InterfaceFunctionType, UFS-StandardTerminalNameAssignmentType, UFS-MandatoryStandardTerminalMappingType, UFS-MandatoryStandardTerminalNameType,			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. VCC	2. VCCQ	3. VCCQ2	4. VSS
	5. RST_n	6. REF_CLK	7. DIN_t	8. DIN_c
	9. DOUT_t	10. C+	11. C-	12. CPOUT1
	13. CPOUT2	14. 0	15. 0	16. 0
	OptionalMapping/StandardTerminalName			
	1. VDDiQ	2. VDDiQ2	3. VDDi	4. DIN0_t
	5. DIN0_c	6. DIN1_t	7. DIN1_c	8. DOUT0_t
	9. DOUT_c	10. DOUT0_c	11. DOUT1_t	12. DOUT1_c

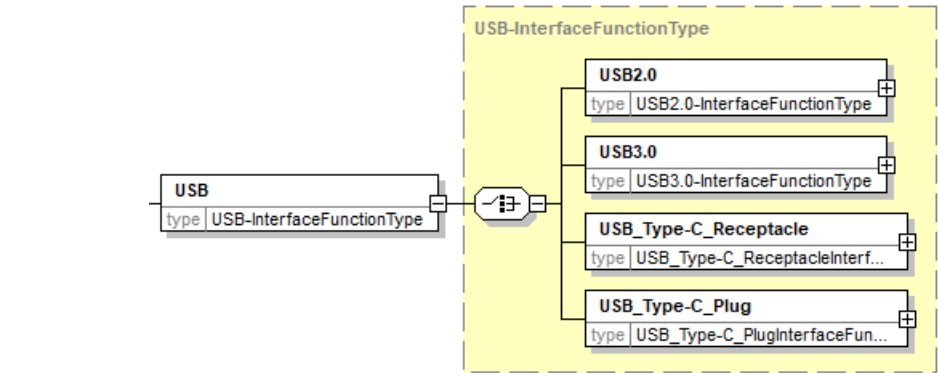
For more information about the UFS Interface, refer to the JEDEC standard JESD220E.

4.5.3.10.50.2. UFSHCI

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/UFSHCI			
diagram				
type	UFSHCI-InterfaceFunctionType, UFSHCI-StandardTerminalNameAssignmentType, UFSHCI-MandatoryStandardTerminalMappingType, UFSHCI-MandatoryStandardTerminalNameType,			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Tx+	2. Tx-	3. Ref_Clock	4. Reset
	5. RX+	6. RX-		

For more information about the UFSHCI Interface, refer to the JEDEC standard JESD223C.

4.5.3.10.51. USB Interface Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/USB			
diagram				
type	USB-InterfaceFunctionType, USB2.0-InterfaceFunctionType, USB3.0-InterfaceFunctionType, USB_Type-C_ReceptacleInterfaceFunctionType, USB_Type-C_PlugInterfaceFunctionType.			



## 4.5.3.10.51.1. USB2.0

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/USB2.0			
diagram				
type	<b>USB2.0-InterfaceFunctionType, USB2.0-StandardTerminalNameAssignmentType, USB2.0-MandatoryStandardTerminalMappingType, USB2.0-MandatoryStandardTerminalNameType, USB2.0-OptionalStandardTerminalMappingType, USB2.0-OptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	1. D-	2. D+		
	<b>OptionalMapping/StandardTerminalName</b>			
	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.

4.5.3.10.51.2. USB3.0

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/FunctionGroup/Interface/USB3.0			
diagram	<p>The diagram illustrates the UML structure for USB3.0 types. It shows a hierarchy where USB3.0-InterfaceFunctionType is associated with USB3.0-StandardTerminalNameAssignmentType. This assignment type is further divided into MandatoryMapping and OptionalMapping. MandatoryMapping is associated with USB3.0-MandatoryStandardTerminalMappingType, which includes StandardTerminalName and TerminalMapID. OptionalMapping is associated with USB3.0-OptionalStandardTerminalMappingType, which also includes StandardTerminalName and TerminalMapID. A constraints box is also shown at the bottom.</p>			
type	USB3.0-InterfaceFunctionType, USB3.0-StandardTerminalNameAssignmentType, USB3.0-MandatoryStandardTerminalMappingType, USB3.0-MandatoryStandardTerminalNameType, USB3.0-OptionalStandardTerminalMappingType, USB3.0-OptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. SSTX+	2. SSTX-	3. SSRX+	4. SSRX-
	5. D+	6. D-	7.	8.
	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.

### 4.5.3.10.51.3. USB\_Type-C\_Receptacle

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/USB/USB_Type-C_Receptacle			
diagram				
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			
	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

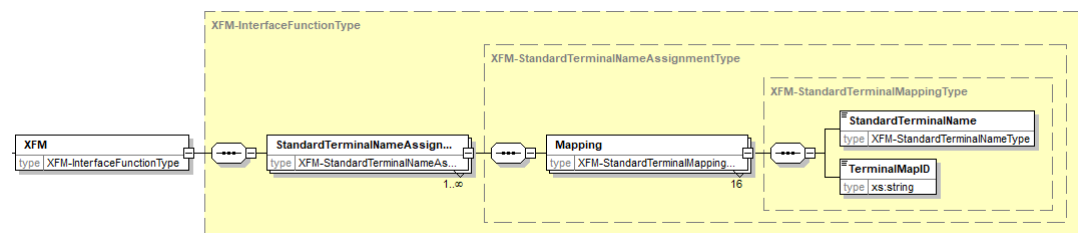
For more information about the USB Type-C Receptacle Interface, refer to the USB Type-C Specification 2.0.

### 4.5.3.10.51.4. USB\_Type-C\_Plug

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/USB/USB_Type-C_Plug			
diagram				
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			
	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

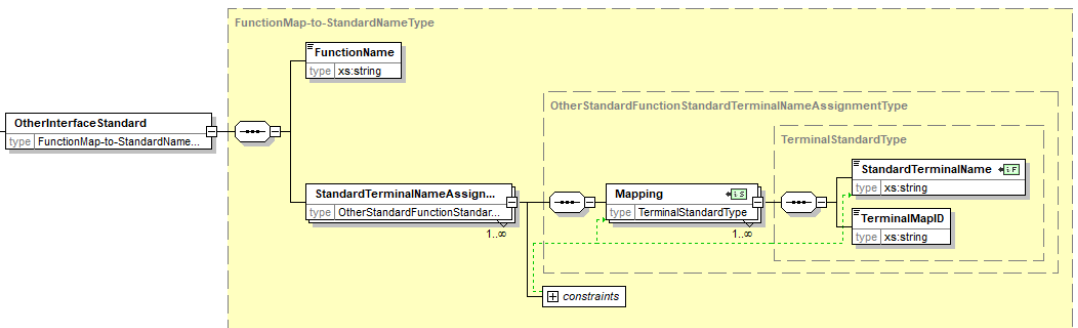
For more information about the USB Type-C Plug Interface, refer to the USB Type-C Specification 2.0.

4.5.3.10.51.5. XFM

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/XFM			
diagram				
type	XFM-InterfaceFunctionType, XFM -StandardTerminalNameAssignmentType, XFM -StandardTerminalMappingType, XFM -StandardTerminalNameType			
list of enumerate values	Mapping/StandardTerminalName			
	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#

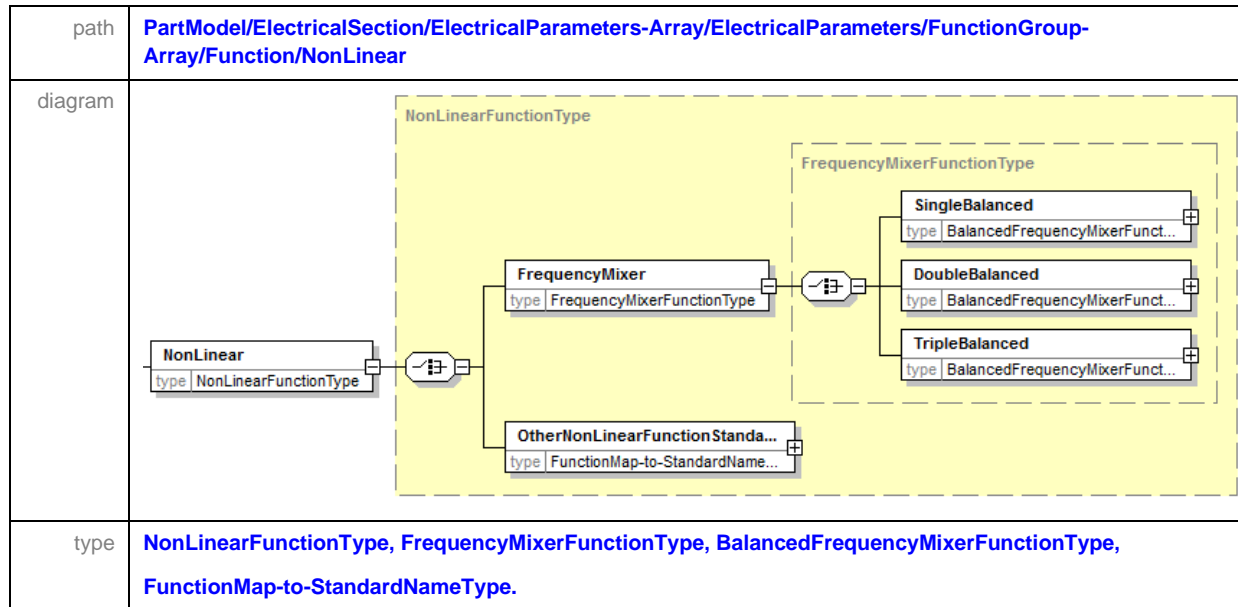
For more information about the XFM Interface, refer to the JEDEC Standard JESD233.

4.5.3.10.52. Other Interface Standard

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Interface/OtherInterfaceStandard			
diagram				
type	FunctionMap-to-StandardNameType, OtherStandardFunctionStandardTerminalNameAssignmentType, TerminalStandardType.			

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*.

#### 4.5.3.11. Non Linear – Frequency Mixer



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.

4.5.3.11.1.     **Balanced**

path	<div>1.   PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear/FrequencyMixer/SingleBalanced</div> <div>2.   PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear/FrequencyMixer/DoubleBalanced</div> <div>3.   PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear/FrequencyMixer/TripleBalanced</div>			
diagram				
type	<div>BalancedFrequencyMixerFunctionType, FrequencyMixerStandardTerminalNameAssignmentType,</div> <div>BalanacedFrequencyMixerMandatoryStandardTerminalMappingType,</div> <div>BalanacedFrequencyMixerMandatoryStandardTerminalNameType,</div> <div>BalanacedFrequencyMixerOptionalStandardTerminalMappingType,</div> <div>BalanacedFrequencyMixerOptionalStandardTerminalNameType.</div>			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1.   Output			
	OptionalMapping/StandardTerminalName			
	1.   Enable	2.   Positive Rail	3.   Negative Rail	

#### 4.5.3.11.2. Other Non Linear Function Standard

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/NonLinear/OtherNonLinearFunctionStandard</a>
diagram	
type	<a href="#">FunctionMap-to-StandardNameType</a> , <a href="#">OtherStandardFunctionStandardTerminalNameAssignmentType</a> , <a href="#">TerminalStandardType</a> .

#### 4.5.3.12. Optoelectronics

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic</a>
diagram	
type	<a href="#">OptoelectronicFunctionType</a> , <a href="#">PhotoemitterFunctionType</a> , <a href="#">PhotosensitiveDeviceFunctionType</a> , <a href="#">OptocouplerFunctionType</a> .

An *Optoelectronic* can be one of the following three types: *Photoemitter*, *PhotosensitiveDevice*, and *Optocoupler*. Each of these types is specified in further detail below.

**4.5.3.12.1. Photoemitter**

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Photoemitter		
diagram			
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.

**4.5.3.12.1.1. 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	



## 4.5.3.12.1.2. LED

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Photoemitter/LED</b>			
diagram				
type	<b>LEDFunctionType, LEDStandardTerminalNameAssignmentType, DiodeStandardTerminalMappingType, DiodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		

## 4.5.3.12.1.3. Laser

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Photoemitter/Laser</b>			
diagram				
type	<b>LaserFunctionType, LaserStandardTerminalNameAssignmentType, LaserMandatoryStandardTerminalMappingType, LaserMandatoryStandardTerminalNameType, LaserOptionalStandardTerminalMappingType, LaserOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. Ground</b>			

4.5.3.12.2.    Photosensitive Device

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice
diagram	<p>The diagram illustrates the class hierarchy for photosensitive devices. A base class, <b>PhotosensitiveDeviceFunctionType</b>, is shown in a yellow-shaded box. It has six subclasses: <b>Photodiode</b>, <b>Photothyristor</b>, <b>Phototriac</b>, <b>Phototransistor</b>, <b>Photodarlington</b>, and <b>PhotovoltaicDiode</b>. Each subclass has a 'type' attribute pointing to its respective type (e.g., <b>PhotodiodeType</b>). A <b>PhotosensitiveDevice</b> class is shown to the left, with a 'type' attribute pointing to <b>PhotosensitiveDeviceFunctionType</b>. A dashed line with a small circle containing a plus sign connects the <b>PhotosensitiveDevice</b> class to the <b>PhotosensitiveDeviceFunctionType</b> box, indicating a generalization relationship.</p>
type	PhotosensitiveDeviceFunctionType, PhotodiodeType, PhotothyristorType, PhototriacType, PhototransistorType, PhotodarlingtonType, PhotovoltaicDiodeType.

A *PhotosensitiveDevice* can be one of the following six types: *Photidode*, *Photothyristor*, *Phototriac*, *Phototransistor*, *Photodarlington*, and *PhotovoltaicDiode*. Each of these types is explained in further detail below.

## 4.5.3.12.2.1. Photodiode

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Photodiode</b>			
diagram				
type	<b>PhotodiodeType, PhotodiodeStandardTerminalNameAssignmentType, PhotodiodeStandardTerminalMappingType, PhotodiodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		

## 4.5.3.12.2.2. Photothyristor

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Photothyristor</b>			
diagram				
type	<b>PhotothyristorType, PhotothyristorStandardTerminalNameAssignmentType, PhotothyristorMandatoryStandardTerminalMappingType, PhotothyristorMandatoryStandardTerminalNameType, PhotothyristorOptionalStandardTerminalMappingType, PhotothyristorOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. Gate</b>			

4.5.3.12.2.3. Phototriac

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Phototriac			
diagram	<p>The diagram illustrates the structure of the PhototriacType and its associated subtypes. The main class, PhototriacType, is shown with a composition relationship (indicated by a solid line with a filled diamond) with the StandardTerminalNameAssignmentType. The StandardTerminalNameAssignmentType is further composed of two subtypes: PhototriacMandatoryStandardTerminalMappingType and PhototriacOptionalStandardTerminalMappingType. The PhototriacMandatoryStandardTerminalMappingType is composed of two elements: StandardTerminalName (type: PhototriacMandatoryStandardTerminalNameType) and TerminalMapID (type: xs:string). The PhototriacOptionalStandardTerminalMappingType is composed of two elements: StandardTerminalName (type: PhototriacOptionalStandardTerminalNameType) and TerminalMapID (type: xs:string). The diagram also shows a composition relationship between PhototriacType and PhototriacMandatoryStandardTerminalMappingType, and a composition relationship between PhototriacType and PhototriacOptionalStandardTerminalMappingType. A 'constraints' box is also present, indicating that there are constraints on the relationships between the classes.</p>			
type	PhototriacType, PhototriacStandardTerminalNameAssignmentType, PhototriacMandatoryStandardTerminalMappingType, PhototriacMandatoryStandardTerminalNameType, PhototriacOptionalStandardTerminalMappingType, PhototriacOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Main Terminal 1	2. Main Terminal 2		
	OptionalMapping/StandardTerminalName			
	1. Gate			

## 4.5.3.12.2.4. Phototransistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Phototransistor			
diagram				
type	<b>PhototransistorType, PhototransistorStandardTerminalNameAssignmentType,</b> <b>PhototransistorMandatoryStandardTerminalMappingType,</b> <b>PhototransistorMandatoryStandardTerminalNameType,</b> <b>PhototransistorOptionalStandardTerminalMappingType,</b> <b>PhototransistorOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Collector</b>	<b>2. Emitter</b>		
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. Base</b>			

## 4.5.3.12.2.5. Photodarlington

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/Photodarlington			
diagram				
type	<b>PhotodarlingtonType, PhotodarlingtonStandardTerminalNameAssignmentType,</b> <b>PhotodarlingtonStandardTerminalMappingType, PhotodarlingtonStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Collector</b>	<b>2. Emitter</b>		

#### 4.5.3.12.2.6. Photovoltaic Diode

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/PhotosensitiveDevice/PhotovoltaicDiode</b>			
diagram				
type	<b>PhotovoltaicDiodeType, PhotovoltaicDiodeStandardTerminalNameAssignmentType, PhotovoltaicDiodeStandardTerminalMappingType, PhotovoltaicDiodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		

#### 4.5.3.12.3. Optocoupler

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Optocoupler</b>			
diagram				
type	<b>OptocouplerFunctionType, PhotodiodeOptocouplerType, PhotothyristorOptocouplerType, PhototriacOptocouplerType, PhototransistorOptocouplerType, PhotodarlingtonOptocouplerType.</b>			

An *Optocoupler* can be one of the following five types: *Photodiode*, *Photothyristor*, *Phototriac*, *Phototransistor*, and *Photodarlington*. Each of these types is further specified below.

## 4.5.3.12.3.1. Photodiode

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Optocoupler/Photodiode</b>			
diagram				
type	<b>PhotodiodeOptocouplerType, PhotodiodeOptocouplerStandardTerminalNameAssignmentType, PhotodiodeOptocouplerStandardTerminalMappingType, PhotodiodeOptocouplerStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b> <b>1. LED-Anode      2. LED-Cathode      3. Detector-Anode      4. Detector-Cathode</b>			

## 4.5.3.12.3.2. Photothyristor

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Optocoupler/Photothyristor</b>			
diagram				
type	<b>PhotothyristorOptocouplerType, PhotothyristorOptocouplerStandardTerminalNameAssignmentType, PhotothyristorOptocouplerMandatoryStandardTerminalMappingType, PhotothyristorOptocouplerMandatoryStandardTerminalNameType, PhotothyristorOptocouplerOptionalStandardTerminalMappingType, PhotothyristorOptocouplerOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b> <b>1. LED-Anode      2. LED-Cathode      3. Detector-Anode      4. Detector-Cathode</b> <b>OptionalMapping/StandardTerminalName</b> <b>1. Detector-Gate</b>			

4.5.3.12.3.3. Phototriac

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Optocoupler/Phototriac			
diagram				
type	PhototriacOptocouplerType, PhototriacOptocouplerStandardTerminalNameAssignmentType, PhototriacOptocouplerMandatoryStandardTerminalMappingType, PhototriacOptocouplerMandatoryStandardTerminalNameType, PhototriacOptocouplerOptionalStandardTerminalMappingType, PhototriacOptocouplerOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. LED-Anode	2. LED-Cathode	3. Detector – Main Terminal 1	4. Detector – Main Terminal 2
	OptionalMapping/StandardTerminalName			
	1. Detector-Gate			



## 4.5.3.12.3.4. Phototransistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Optocoupler/Phototransistor			
diagram				
type	PhototransistorOptocouplerType, PhototransistorOptocouplerStandardTerminalNameAssignmentType, PhototransistorOptocouplerMandatoryStandardTerminalMappingType, PhototransistorOptocouplerMandatoryStandardTerminalNameType, PhototransistorOptocouplerOptionalStandardTerminalMappingType, PhototransistorOptocouplerOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. LED-Anode	2. LED-Cathode	3. Detector-Anode,	4. Detector-Cathode
	OptionalMapping/StandardTerminalName			
	1. Detector-Base			

4.5.3.12.3.5. Photodarlington

path	2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Optoelectronic/Optocoupler/Photodarlington			
diagram				
type	PhotodarlingtonOptocouplerType, PhotodarlingtonOptocouplerStandardTerminalNameAssignmentType, PhotodarlingtonOptocouplerBi-directionalStandardTerminalMappingType, PhotodarlingtonOptocouplerBi-directionalStandardTerminalNameType, PhotodarlingtonOptocouplerUni-directionalStandardTerminalMappingType, PhotodarlingtonOptocouplerUni-directionalStandardTerminalNameType, PhotodarlingtonOptocouplerMandatoryStandardTerminalMappingType, PhotodarlingtonOptocouplerMandatoryStandardTerminalNameType.			
list of enumerate values	Bi-directionalMapping/StandardTerminalName			
	1. Cathode-Anode	2. Anode-Cathode		
	Uni-directionalMapping /StandardTerminalName			
	1. Anode	2. Cathode		
	MandatoryMapping/StandardTerminalName			
	1. Detector-Collector	2. Detector-Emitter		

**4.5.3.13. Relay**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay</a>
diagram	<p>The diagram illustrates the UML class hierarchy for the 'Relay' type. It starts with a 'Relay' type (RelayFunctionType) which is connected to a choice between 'Solid State' (SolidStateRelayFunctionType) and 'Electromagnetic' (ElectromagneticRelayFunctionType). 'Solid State' is further connected to a choice between 'Single Throw' (SolidStateRelaySingleThrowType) and 'Double Throw' (SolidStateRelayDoubleThrowType). 'Electromagnetic' is connected to a choice between 'Single Throw' (ElectromagneticRelaySingleThrowType) and 'Double Throw' (ElectromagneticRelayDoubleThrowType). The diagram is enclosed in a yellow dashed box labeled 'RelayFunctionType'.</p>
type	<a href="#">RelayFunctionType</a> , <a href="#">SolidStateRelayFunctionType</a> , <a href="#">SolidStateRelaySingleThrowType</a> , <a href="#">SolidStateRelayDoubleThrowType</a> , <a href="#">ElectromagneticRelayFunctionType</a> , <a href="#">ElectromagneticRelaySingleThrowType</a> , <a href="#">ElectromagneticRelayDoubleThrowType</a> .

**4.5.3.13.1. Solid State Relay – Single Throw**

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/SolidState/SingleThrow			
diagram				
type	<b>SolidStateRelayFunctionType, SolidStateRelaySingleThrowType,</b> <b>SolidStateRelaySingleThrowStandardTerminalNameAssignmentType,</b> <b>SolidStateRelayBi-directionalStandardTerminalMappingType,</b> <b>SolidStateRelayBi-directionalStandardTerminalNameType,</b> <b>SolidStateRelayUni-directionalStandardTerminalMappingType,</b> <b>SolidStateRelayUni-directionalStandardTerminalNameType,</b> <b>SolidStateRelaySingleThrowChoiceStandardTerminalMappingType,</b> <b>SolidStateRelaySingleThrowChoiceStandardTerminalNameType,</b> <b>SolidStateRelaySingleThrowOptionalStandardTerminalMappingType,</b> <b>SolidStateRelaySingleThrowOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>Bi-directionalMapping/StandardTerminalName</b>			
	<b>1. Cathode-Anode</b>	<b>2. Anode-Cathode</b>		
	<b>Uni-directionalMapping /StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		
	<b>ChoiceMapping/StandardTerminalName</b>			
	<b>1. Normally Closed</b>	<b>2. Normally Open</b>		
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. Pole</b>			

#### 4.5.3.13.2. Solid State Relay – Double Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/SolidState/DoubleThrow		
diagram			
type	SolidStateRelayFunctionType, SolidStateRelayDoubleThrowStandardTerminalNameAssignmentType, SolidStateRelayBi-directionalStandardTerminalMappingType, SolidStateRelayBi-directionalStandardTerminalNameType, SolidStateRelayUni-directionalStandardTerminalMappingType, SolidStateRelayUni-directionalStandardTerminalNameType, SolidStateRelayDoubleThrowMandatoryStandardTerminalMappingType, SolidStateRelayDoubleThrowMandatoryStandardTerminalNameType.		
list of enumerate values	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

4.5.3.13.3. Electromagnetic Relay – Single Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/Electromagnetic/SingleThrow			
diagram				
type	ElectromagneticRelayFunctionType, ElectromagneticRelaySingleThrowType, ElectromagneticRelaySingleThrowStandardTerminalNameAssignmentType, ElectromagneticRelayMandatoryStandardTerminalMappingType, ElectromagneticRelayMandatoryStandardTerminalNameType, ElectromagneticRelaySingleThrowContact-ArrayType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. Input 1	2. Input 2		

#### 4.5.3.13.3.1. Electromagnetic Relay – Single Throw – Contact Array

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/Electromagnetic/SingleThrow/StandardTerminalNameAssignment/Contact-Array			
diagram				
type	ElectromagneticRelaySingleThrowContact-ArrayType, ElectromagneticRelaySingleThrowContactType, ElectromagneticRelaySingleThrowChoiceStandardTerminalMappingType, ElectromagneticRelaySingleThrowChoiceStandardTerminalNameType, ElectromagneticRelaySingleThrowMandatoryStandardTerminalMappingType, ElectromagneticRelaySingleThrowMandatoryStandardTerminalNameType.			
list of enumerate values	ChoiceMapping/StandardTerminalName			
	1. Normally Closed	2. Normally Open		
	MandatoryMapping/StandardTerminalName			
	1. Pole			

**4.5.3.13.4. Electromagnetic Relay – Double Throw**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/Electromagnetic/DoubleThrow</b>			
diagram				
type	<b>ElectromagneticRelayFunctionType, ElectromagneticRelaySingleThrowType,</b> <b>ElectromagneticRelaySingleThrowStandardTerminalNameAssignmentType,</b> <b>ElectromagneticRelayMandatoryStandardTerminalMappingType,</b> <b>ElectromagneticRelayMandatoryStandardTerminalNameType,</b> <b>ElectromagneticRelaySingleThrowContact-ArrayType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Input 1</b>	<b>2. Input 2</b>		

**4.5.3.13.5. Electromagnetic Relay – Double Throw – Contact Array**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Relay/Electromagnetic/DoubleThrow/StandardTerminalNameAssignment/Contact-Array</b>			
diagram				
type	<b>ElectromagneticRelayDoubleThrowContact-ArrayType,</b> <b>ElectromagneticRelayDoubleThrowContactType,</b> <b>ElectromagneticRelayDoubleThrowMandatoryStandardTerminalMappingType,</b> <b>ElectromagneticRelayDoubleThrowMandatoryStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Normally Closed</b>	<b>2. Normally Open</b>	<b>3. Pole</b>	



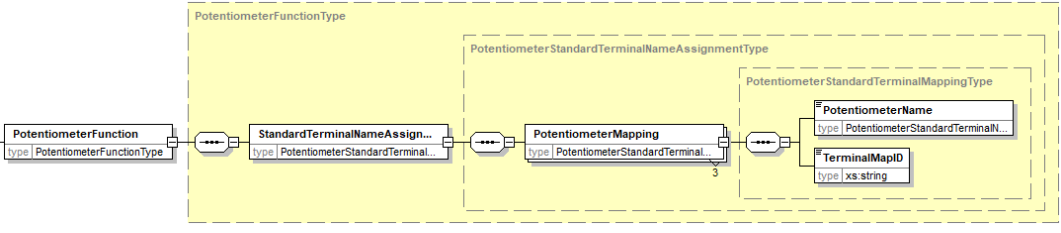
## 4.5.3.14. Resistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor
diagram	<p>The diagram illustrates the Resistor class hierarchy and its association with terminal name assignments. The Resistor class (type ResistorFunctionType) is associated with a collection of specialized ResistorFunctionType subclasses: Fixed (FixedResistorFunctionType), Variable (VariableResistorFunctionType), Photo (PhotoResistorFunctionType), PotentiometerFunction (PotentiometerFunctionType), Varistor (VaristorResistorFunctionType), Thermistor (ThermistorFunctionType), LightDependent (LightDependentResistorFunctionType), Shunt (ShuntResistorFunctionType), and Magnetic (MagneticResistorFunctionType). Each specialized class is associated with a StandardTerminalNameAssignment class (type TwoTerminalResistorStandardTerminalNameAssignmentType). The association between the specialized classes and the StandardTerminalNameAssignment class is labeled with '1..∞'.</p>
type	ResistorFunctionType, FixedResistorFunctionType, VariableResistorFunctionType, PhotoResistorFunctionType, PotentiometerFunctionType, VaristorResistorFunctionType, ThermistorFunctionType, LightDependentResistorFunctionType, ShuntResistorFunctionType, MagneticResistorFunctionType, TwoTerminalResistorStandardTerminalNameAssignmentType, PotentiometerStandardTerminalNameAssignmentType.

**4.5.3.14.1. Two Terminal Resistor Standard Terminal Name Assignment Type**

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Fixed/StandardTerminalNameAssignment</a></li> <li>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Variable/StandardTerminalNameAssignment</a></li> <li>3. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Photo/StandardTerminalNameAssignment</a></li> <li>4. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/PotentiometerFunction/StandardTerminalNameAssignment</a></li> <li>5. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Varistor/StandardTerminalNameAssignment</a></li> <li>6. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Thermistor/StandardTerminalNameAssignment</a></li> <li>7. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/LightDependent/StandardTerminalNameAssignment</a></li> <li>8. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Shunt/StandardTerminalNameAssignment</a></li> <li>9. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/Magnetic/StandardTerminalNameAssignment</a></li> </ol>			
diagram	<pre> classDiagram     class StandardTerminalNameAssignmentType {         type TwoTerminalResistorStandardTerminalNameAssignmentType     }     class Mapping {         type TwoTerminalResistorStandardTerminalNameAssignmentType     }     class TwoTerminalResistorStandardTerminalMappingType {         StandardTerminalName type TwoTerminalResistorStandardTerminalNameAssignmentType         TerminalMapID type xs:string     }     StandardTerminalNameAssignmentType "1..∞" -- "2" Mapping     Mapping -- TwoTerminalResistorStandardTerminalMappingType     </pre>			
type	<a href="#">TwoTerminalResistorStandardTerminalNameAssignmentType</a> , <a href="#">TwoTerminalResistorStandardTerminalMappingType</a> , <a href="#">TwoTerminalResistorStandardTerminalNameType</a> .			
list of enumerate values	<a href="#">Mapping/StandardTerminalName</a>			
	1. <a href="#">Terminal 1</a>	2. <a href="#">Terminal 2</a>		

4.5.3.14.2. Potentiometer Function

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Resistor/PotentiometerFunction			
diagram				
type	PotentiometerFunctionType, PotentiometerStandardTerminalNameAssignmentType, PotentiometerStandardTerminalMappingType, PotentiometerStandardTerminalNameType.			
list of enumerate values	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:

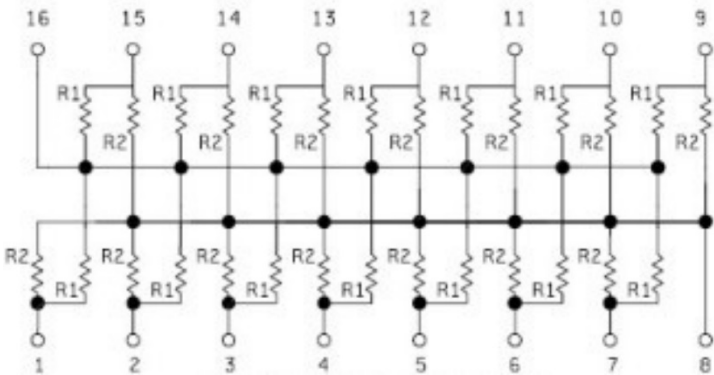


Figure 41 — Pull-up Pull-down Resistor Array

**4.5.3.14.2 Potentiometer Function (cont'd)**

```

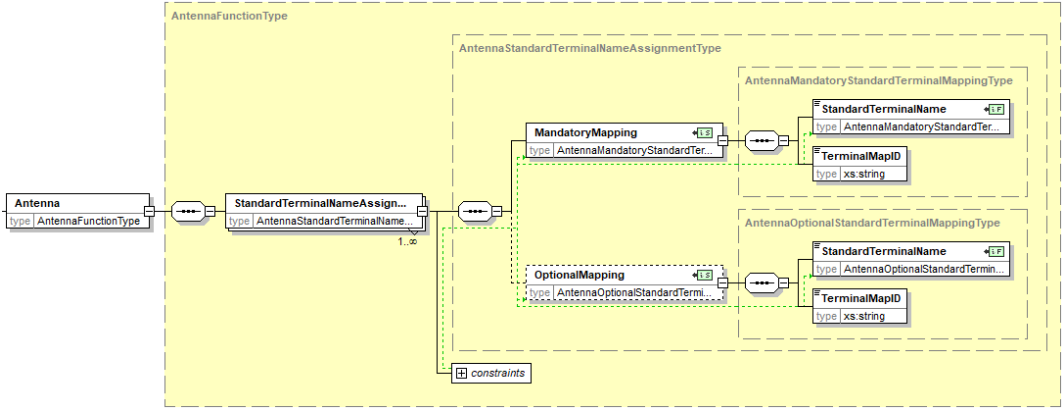
<FunctionGroup-Array>
  <Function>
    <Resistor>
      <StandardTerminalNameAssignment>
        <Terminal1>
          <TerminalNumber>1</TerminalNumber>
        </Terminal1>
        <Terminal2>
          <TerminalNumber>16</TerminalNumber>
        </Terminal2>
      </StandardTerminalNameAssignment>
      :
      :
      <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>

```

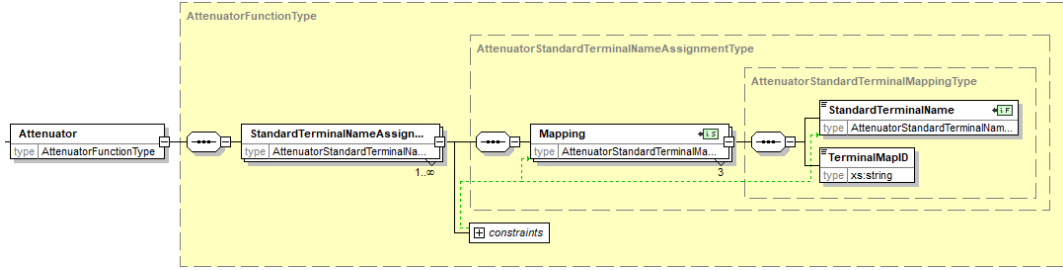
**4.5.3.15. RF**

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF</a>
diagram	<p>The diagram illustrates the RF-FunctionType hierarchy. A central 'RF' block (type RF-FunctionType) is connected to a dashed yellow box labeled 'RF-FunctionType'. Inside this box is a vertical list of 15 function types, each with a '+' icon:</p> <ul style="list-style-type: none"> <li>Antenna (type AntennaFunctionType)</li> <li>Attenuator (type AttenuatorFunctionType)</li> <li>Balun (type BalunFunctionType)</li> <li>Circulator (type CirculatorFunctionType)</li> <li>Coupler (type CouplerFunctionType)</li> <li>DCBlock (type DCBlockType)</li> <li>Detector (type DetectorFunctionType)</li> <li>Divider (type DividerFunctionType)</li> <li>Isolator (type IsolatorFunctionType)</li> <li>Limiter (type LimiterFunctionType)</li> <li>Mixer (type MixerFunctionType)</li> <li>Modulator (type ModulatorType)</li> <li>Demodulator (type DemodulatorType)</li> <li>Multiplier (type MultiplierFunctionType)</li> <li>PhaseDetector (type PhaseDetectorType)</li> <li>PhaseShifter (type PhaseShifterFunctionType)</li> </ul>
type	<a href="#">FunctionType</a> , <a href="#">AntennaFunctionType</a> , <a href="#">AttenuatorFunctionType</a> , <a href="#">BalunFunctionType</a> , <a href="#">CirculatorFunctionType</a> , <a href="#">CouplerFunctionType</a> , <a href="#">DCBlockType</a> , <a href="#">DetectorFunctionType</a> , <a href="#">DividerFunctionType</a> , <a href="#">IsolatorFunctionType</a> , <a href="#">LimiterFunctionType</a> , <a href="#">MixerFunctionType</a> , <a href="#">ModulatorType</a> , <a href="#">DemodulatorType</a> , <a href="#">MultiplierFunctionType</a> , <a href="#">PhaseDetectorType</a> , <a href="#">PhaseShifterFunctionType</a> .

4.5.3.15.1. Antenna

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Antenna			
diagram				
type	AntennaFunctionType, AntennaStandardTerminalNameAssignmentType, AntennaMandatoryStandardTerminalMappingType, AntennaMandatoryStandardTerminalNameType, AntennaOptionalStandardTerminalMappingType, AntennaOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Port 1			
	OptionalMapping/StandardTerminalName			
	1. Port2			

4.5.3.15.2. Attenuator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Attenuator			
diagram				
type	AttenuatorFunctionType, AttenuatorStandardTerminalNameAssignmentType, AttenuatorStandardTerminalMappingType, AttenuatorStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. Port 1	2. Port 2	3. Ground	

**4.5.3.15.3. Balun**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Balun</b>			
diagram				
type	<b>BalunFunctionType, BalunStandardTerminalNameAssignmentType, BalunStandardTerminalMappingType, BalunStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Balanced 1</b>	<b>2. Balanced 2</b>	<b>3. Unbalanced</b>	<b>4. Ground</b>

**4.5.3.15.4. Circulator**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Circulator</b>			
diagram				
type	<b>CirculatorFunctionType, CirculatorStandardTerminalNameAssignmentType, CirculatorMandatoryStandardTerminalMappingType, CirculatorMandatoryStandardTerminalNameType, CirculatorOptionalStandardTerminalMappingType, CirculatorOptionalStandardTerminalNameType.</b>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Port 1</b>	<b>2. Port 2</b>	<b>3. Port 3</b>	<b>4. Ground</b>
	<b>OptionalMapping/StandardTerminalName</b>			
	<b>1. Port 4</b>			

4.5.3.15.5. Coupler

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Coupler
diagram	
type	CouplerFunctionType, CouplerPropertiesType, BidirectionalCouplerType, DirectionalCouplerType.

4.5.3.15.5.1. BiDirectional

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Coupler/BiDirectional			
diagram				
type	BidirectionalCouplerType, BidirectionalCouplerStandardTerminalNameAssignmentType, BiDirectionalCouplerStandardTerminalMappingType, BiDirectionalCouplerStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. Input	2. Output	3. Coupled Forward	4. Coupled Reverse
	5. Ground			



4.5.3.15.5.2. Directional

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Coupler/Directional			
diagram	<p>The diagram illustrates the structure of the DirectionalCouplerType. It shows a 'Directional' entity (type DirectionalCouplerType) connected to a 'StandardTerminalNameAssign...' entity (type DirectionalCouplerStandardTer...). This is further connected to a 'MandatoryMapping' entity (type DirectionalCouplerMandatorySt...) and an 'OptionalMapping' entity (type DirectionalCouplerOptionalStan...). Both mapping entities are connected to 'StandardTerminalName' and 'TerminalMapID' entities (type xs:string). The diagram is enclosed in a yellow box labeled 'DirectionalCouplerType' and includes a 'constraints' box at the bottom.</p>			
type	DirectionalCouplerType, DirectionalCouplerStandardTerminalNameAssignmentType, ConnectionType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Input	2. Output	3. Coupled Forward	4. Ground
	OptionalMapping/StandardTerminalName			
	1. Termination			

4.5.3.15.6. DC Block

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/DCBlock			
diagram	<p>The diagram illustrates the structure of the DCBlockType. It is a class hierarchy where DCBlockType (type DCBlockType) contains a 1..26 StandardTerminalNameAssignmentType (type DCBlockStandardTerminalName...). This assignment type contains a MandatoryMapping (type DCBlockMandatoryStandardTer...) and an OptionalMapping (type DCBlockOptionalStandardTermin...). The MandatoryMapping contains a 2 StandardTerminalName (type DCBlockMandatoryStandardTer...) and a TerminalMapID (type xs:string). The OptionalMapping contains a StandardTerminalName (type DCBlockOptionalStandardTermin...) and a TerminalMapID (type xs:string). A constraints box is also shown.</p>			
type	DCBlockType, DCBlockStandardTerminalNameAssignmentType, DCBlockMandatoryStandardTerminalMappingType, DCBlockMandatoryStandardTerminalNameType, DCBlockOptionalStandardTerminalMappingType, DCBlockOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Port 1	2. Port 2		
	OptionalMapping/StandardTerminalName			
	1. Ground			

4.5.3.15.7. Detector

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Detector			
diagram	<p>The diagram illustrates the structure of the Detector type. It is composed of several nested types and associations:</p> <ul style="list-style-type: none"><li><b>Detector</b> (type <code>DetectorFunctionType</code>) is associated with <b>StandardTerminalNameAssignmentType</b> (type <code>DetectorStandardTerminalNameAssignmentType</code>) via a 1..* association.</li><li><b>StandardTerminalNameAssignmentType</b> is associated with <b>MandatoryMapping</b> (type <code>DetectorMandatoryStandardTerminalMappingType</code>) and <b>OptionalMapping</b> (type <code>DetectorOptionalStandardTerminalMappingType</code>) via a 1..* association.</li><li><b>MandatoryMapping</b> is associated with <b>StandardTerminalName</b> (type <code>DetectorMandatoryStandardTerminalNameType</code>) and <b>TerminalMapID</b> (type <code>xs:string</code>) via a 1..* association.</li><li><b>OptionalMapping</b> is associated with <b>StandardTerminalName</b> (type <code>DetectorOptionalStandardTerminalNameType</code>) and <b>TerminalMapID</b> (type <code>xs:string</code>) via a 1..* association.</li><li>A <b>constraints</b> compartment is also present.</li></ul>			
type	DetectorFunctionType, DetectorStandardTerminalNameAssignmentType, DetectorMandatoryStandardTerminalMappingType, DetectorMandatoryStandardTerminalNameType, DetectorOptionalStandardTerminalMappingType, DetectorOptionalStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. Input 1	2. Output	3. Ground	
	OptionalMapping/StandardTerminalName			
	4. Input 2			

4.5.3.15.8. Divider

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Divider			
diagram				
type	DividerFunctionType, DividerPropertiesType, DividerStandardTerminalNameAssignmentType, DividerMandatoryStandardTerminalMappingType, DividerMandatoryStandardTerminalNameType, DividerOptionalStandardTerminalMappingType, DividerOptionalStandardTerminalNameType.			
list of enumerate values	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		

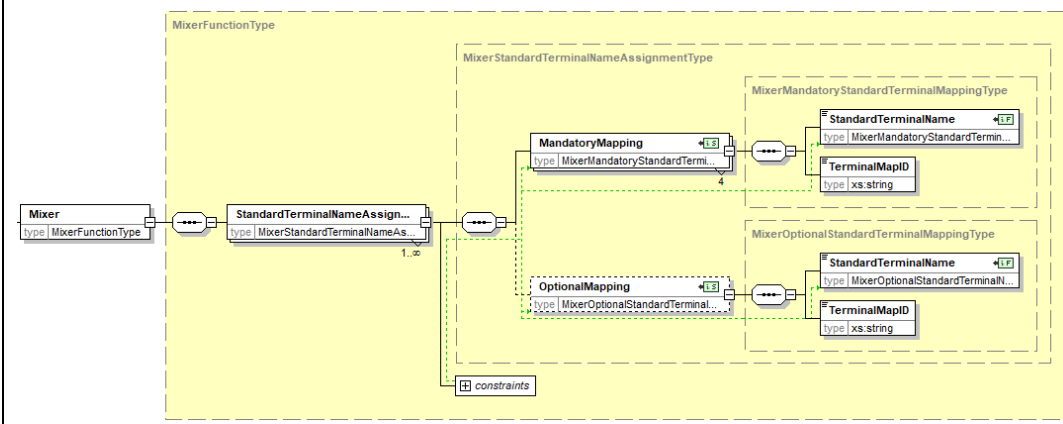
**4.5.3.15.9. Isolator**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Isolator</b>			
diagram	<p>The diagram illustrates the structure of the Isolator function group. It shows a sequence of relationships: Isolator (type IsolatorFunctionType) is associated with StandardTerminalNameAssignment (type IsolatorStandardTerminalNameAssignmentType) via a 1..∞ relationship. This is followed by a Mapping (type IsolatorStandardTerminalMappingType) with a 3 relationship. The Mapping is associated with StandardTerminalName (type IsolatorStandardTerminalNameType) via a 3 relationship. The StandardTerminalName class has a TerminalMapID attribute of type xs:string. A constraints box is also present.</p>			
type	<b>IsolatorFunctionType, IsolatorStandardTerminalNameAssignmentType, IsolatorStandardTerminalMappingType, IsolatorStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Input</b>	<b>2. Output</b>	<b>3. Ground</b>	

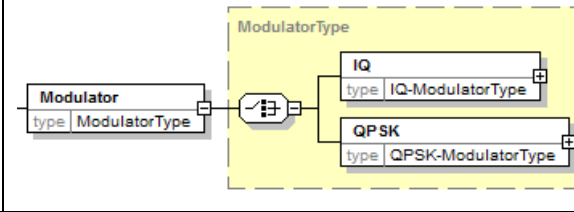
**4.5.3.15.10. Limiter**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Limiter</b>			
diagram	<p>The diagram illustrates the structure of the Limiter function group. It shows a sequence of relationships: Limiter (type LimiterFunctionType) is associated with StandardTerminalNameAssignment (type LimiterStandardTerminalNameAssignmentType) via a 1..∞ relationship. This is followed by a Mapping (type LimiterStandardTerminalMappingType) with a 3 relationship. The Mapping is associated with StandardTerminalName (type LimiterStandardTerminalNameType) via a 3 relationship. The StandardTerminalName class has a TerminalMapID attribute of type xs:string. A constraints box is also present.</p>			
type	<b>LimiterFunctionType, LimiterStandardTerminalNameAssignmentType, LimiterStandardTerminalMappingType, LimiterStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Input</b>	<b>2. Output</b>	<b>3. Ground</b>	

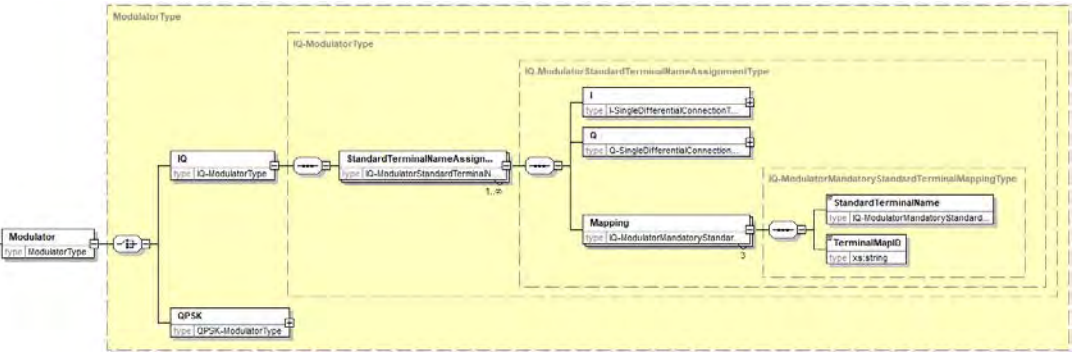
4.5.3.15.11. Mixer

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Mixer			
diagram				
type	MixerFunctionType, MixerStandardTerminalNameAssignmentType, MixerMandatoryStandardTerminalMappingType, MixerMandatoryStandardTerminalNameType.			
list of enumerate values	MandatoryMapping/StandardTerminalName			
	1. RF	2. LO	3. IF	4. Ground
	OptionalMapping/StandardTerminalName			
	1. PowerLO			

4.5.3.15.12. Modulator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator			
diagram				
type	ModulatorType, IQ-ModulatorType, IQ-ModulatorStandardTerminalNameAssignmentType, QPSK-ModulatorType, QPSK-ModulatorStandardTerminalNameAssignmentType, ConnectionType.			

4.5.3.15.12.1. IQ - Modulator Standard Terminal Name Assignment

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator/IQ			
diagram				
type	IQ-ModulatorStandardTerminalNameAssignmentType, I-SingleDifferentialConnectionType, Q-SingleDifferentialConnectionType, IQ-ModulatorMandatoryStandardTerminalMappingType, IQ-ModulatorMandatoryStandardTerminalNameType.			
list of enumerate values	IQ-ModulatorMandatoryStandardTerminalMapping/StandardTerminalName			
	1. RF	2. LO	3. Ground	

4.5.3.15.12.1.1. I-SingleDifferentialConnectionType

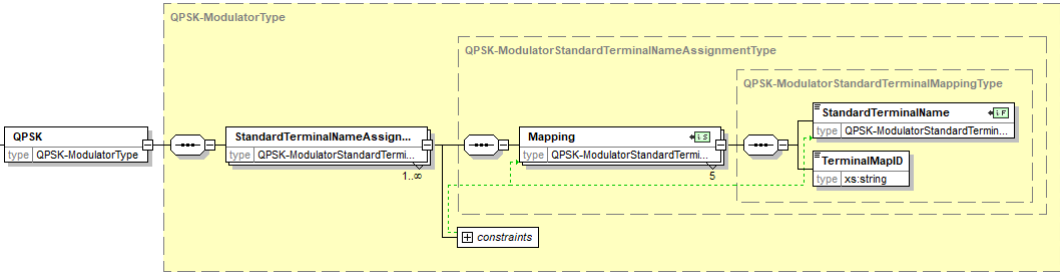
path	<div>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator/IQ/StandardTerminalNameAssignment/I</a></div> <div>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Demodulator/IQ/StandardTerminalNameAssignment/I</a></div>			
diagram				
type	<div>I-SingleDifferentialConnectionType,</div> <div><a href="#">IQ-Modulator-I-SingleEndedConnectionStandardTerminalMappingType</a>,</div> <div><a href="#">IQ-Modulator-I-SingleEndedConnectionStandardTerminalNameType</a>, <a href="#">I-DifferentialConnectionType</a>,</div> <div><a href="#">I-DifferentialConnectionMappingType</a>, <a href="#">I-DifferentialStandardTerminalNameType</a></div>			
list of enumerate values	<a href="#">SingleEndedConnectionMapping/StandardTerminalName</a>			
	1. I			
	<a href="#">DifferentialConnectionMapping/I-DifferentialConnectionMapping/StandardTerminalName</a>			
	1. I+	2. I-		



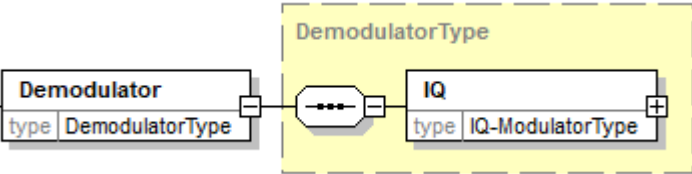
#### 4.5.3.15.12.1.2. Q-SingleDifferentialConnectionType

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator/IQ/StandardTerminalNameAssignment/Q</a></li> <li>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Demodulator/IQ/StandardTerminalNameAssignment/Q</a></li> </ol>			
diagram				
type	<b>SingleDifferentialConnectionType,</b> <b>IQ-Modulator-Q-SingleEndedConnectionStandardTerminalMappingType,</b> <b>IQ-Modulator-Q-SingleEndedConnectionStandardTerminalNameType, Q-DifferentialConnectionType,</b> <b>Q-DifferentialConnectionMappingType, Q-DifferentialStandardTerminalNameType</b>			
list of enumerate values	<b>SingleEndedConnectionMapping/StandardTerminalName</b>			
	1. Q			
	<b>DifferentialConnectionMapping/I-DifferentialConnectionMapping/StandardTerminalName</b>			
	1. Q+	2. Q-		

4.5.3.15.12.2. QPSK - Modulator Standard Terminal Name Assignment

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Modulator/QPSK			
diagram				
type	QPSK-ModulatorType, QPSK-ModulatorStandardTerminalNameAssignmentType, QPSK-ModulatorStandardTerminalMappingType, QPSK-ModulatorStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. RF Input	2. RF Output	3. Ctrl 1	4. Ctrl 2
	5. Ground			

4.5.3.15.13. Demodulator

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Demodulator			
diagram				
type	DemodulatorType, IQ-ModulatorType			

**4.5.3.15.14. Multiplier**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/Multiplier</b>			
diagram				
type	<b>MultiplierFunctionType, MultiplierPropertiesType, MultiplierStandardTerminalNameAssignmentType, MultiplierStandardTerminalMappingType, MultiplierStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Input</b>	<b>2. Output</b>	<b>3. Ground</b>	

**4.5.3.15.15. Phase Detector**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/PhaseDetector</b>			
diagram				
type	<b>PhaseDetectorType, PhaseDetectorStandardTerminalNameAssignmentType, PhaseDetectorStandardTerminalMappingType, PhaseDetectorStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Input</b>	<b>2. Reference</b>	<b>3. Output</b>	<b>4. Ground</b>

4.5.3.15.16. Phase Shifter

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/RF/PhaseShifter			
diagram				
type	PhaseShifterFunctionType, PhaseShifterStandardTerminalNameAssignmentType, ConnectionType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. Input	2. Output	3. Control	4. Ground

4.5.3.16. Source

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source			
diagram				
type	SourceFunctionType, BatteryFunctionType, GenericSourceFunctionType, VoltageSourceFunctionType, CurrentSourceFunctionType.			

#### 4.5.3.16.1. Battery Source

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Battery</b>		
diagram			
type	<b>BatteryFunctionType, DC-SourceStandardTerminalNameAssignmentType, DC-SourceStandardTerminalMappingType, DC-SourceStandardTerminalNameType.</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	<b>1. Positive</b>	<b>2. Negative</b>	

#### 4.5.3.16.2. Generic Voltage Source

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/VoltageSource</b>		
diagram			
type	<b>VoltageSourceFunctionType, AC-VoltageSourceType, AC-SourceStandardTerminalNameAssignmentType, AC-SourceStandardTerminalMappingType, AC-SourceStandardTerminalNameType, DC-VoltageSourceType, DC-SourceStandardTerminalNameAssignmentType, DC-SourceStandardTerminalMappingType, DC-SourceStandardTerminalNameType,</b>		

4.5.3.16.3. Generic Current Source

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/CurrentSource
diagram	
type	GenericSourceFunctionType, VoltageSourceFunctionType, AC-SourceType, DC-SourceType, CurrentSourceFunctionType.

4.5.3.16.3.1. AC-Source Type

path	<ol style="list-style-type: none"><li>1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/VoltageSource/AC-VoltageSource</li><li>2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/CurrentSource/AC-VoltageSource</li></ol>			
diagram				
type	AC-SourceType, AC-SourceStandardTerminalNameAssignmentType, AC-SourceStandardTerminalMappingType, AC-SourceStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. Terminal 1	2. Terminal 2		

#### 4.5.3.16.3.2. DC-Source Type

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/VoltageSource/DC-VoltageSource</a></li> <li>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Source/Generic/CurrentSource/DC-VoltageSource</a></li> </ol>			
diagram				
type	<a href="#">DC-Source Type</a> , <a href="#">DC-SourceStandardTerminalNameAssignmentType</a> , <a href="#">DC-SourceStandardTerminalMappingType</a> , <a href="#">DC-SourceStandardTerminalNameType</a> .			
list of enumerate values	<a href="#">Mapping/StandardTerminalName</a>			
	1. Positive	2. Positive		

#### 4.5.3.17. Switch

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Switch</a>			
diagram				
type	<a href="#">SwitchFunctionType</a> , <a href="#">SingleThrowSwitchType</a> , <a href="#">DoubleThrowSwitchType</a> , <a href="#">ManyThrowSwitchType</a> .			

**4.5.3.17.1. Single Throw**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Switch/SingleThrow</b>			
diagram				
type	<b>SingleThrowSwitchType, SingleThrowSwitchStandardTerminalNameAssignmentType, SingleThrowSwitchChoiceStandardTerminalMappingType, SingleThrowSwitchChoiceStandardTerminalNameType, SingleThrowSwitchMandatoryStandardTerminalMappingType, SingleThrowSwitchMandatoryStandardTerminalNameType.</b>			
list of enumerate values	<b>ChoiceMapping/StandardTerminalName</b>			
	<b>1. Normally Closed</b>	<b>2. Normally Open</b>		
	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Pole</b>			

**4.5.3.17.2. Double Throw**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Switch/DoubleThrow</b>			
diagram				
type	<b>DoubleThrowSwitchType, DoubleThrowSwitchStandardTerminalNameAssignmentType, DoubleThrowSwitchStandardTerminalMappingType, DoubleThrowSwitchStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Normally Closed</b>	<b>2. Normally Open</b>	<b>3. Pole</b>	



4.5.3.17.3. Many Throw

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Switch/SingleThrow		
diagram			
type	ManyThrowSwitchType, ManyThrowSwitchStandardTerminalNameAssignmentType, ManyThrowSwitchNormallyOpenStandardTerminalMappingType, ManyThrowSwitchNormallyOpenStandardTerminalNameType, ManyThrowSwitchMandatoryStandardTerminalMappingType, ManyThrowSwitchMandatoryStandardTerminalNameType.		
list of enumerate values	MultiMapping/StandardTerminalName		
	1. Normally Open		
	MandatoryMapping/StandardTerminalName		
	1. Normally Closed	2. Pole	

**4.5.3.18. Thyristor**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor</b>		
diagram			
type	<b>ThyristorFunctionType, BidirectionalThyristorType,</b> <b>BidirectionalThyristorDiodeStandardTerminalNameAssignmentType, BidirectionalThyristorTriodeType,</b> <b>UnidirectionalThyristorType, UnidirectionalReverseBlockingThyristorType,</b> <b>UnidirectionalReverseConductingThyristorType.</b>		

**4.5.3.18.1. Bidirectional Thyristor Diode**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Bidirectional/Diode</b>		
diagram			
type	<b>BidirectionalThyristorDiodeType, BidirectionalThyristorDiodeStandardTerminalNameAssignmentType,</b> <b>BidirectionalThyristorDiodeStandardTerminalMappingType,</b> <b>BidirectionalThyristorDiodeStandardTerminalNameType.</b>		
list of enumerate values	<b>Mapping/StandardTerminalName</b>		
	<b>1. Terminal 1</b>	<b>2. Terminal 2</b>	

#### 4.5.3.18.2. Bidirectional Thyristor Triode

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Bidirectional/Triode</b>			
diagram				
type	<b>BidirectionalThyristorDiodeType, BidirectionalThyristorDiodeStandardTerminalNameAssignmentType, BidirectionalThyristorTriodeStandardTerminalMappingType, BidirectionalThyristorTriodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Terminal 1</b>	<b>2. Terminal 2</b>	<b>3. Gate</b>	

#### 4.5.3.18.3. Unidirectional

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional</b>			
diagram				
type	<b>UnidirectionalThyristorType, UnidirectionalReverseBlockingThyristorType, UnidirectionalReverseConductingThyristorType.</b>			

**4.5.3.18.3.1. Reverse Blocking Diode**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional/ReverseBlocking/Diode</b>			
diagram				
type	<b>UnidirectionalReverseBlockingThyristorDiodeType, DiodeStandardTerminalNameAssignmentType, DiodeStandardTerminalMappingType, DiodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		

**4.5.3.18.3.2. Reverse Blocking Triode**

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional/ReverseBlocking/Triode</b>			
diagram				
type	<b>UnidirectionalReverseBlockingThyristorTriodeType, UnidirectionalTriodeStandardTerminalNameAssignmentType, UnidirectionalTriodeStandardTerminalMappingType, UnidirectionalTriodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>	<b>3. Gate</b>	

#### 4.5.3.18.3.3. Reverse Conducting Diode

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional/ReverseConducting/Diode</b>			
diagram				
type	<b>UnidirectionalReverseConductingThyristorDiodeType, DiodeStandardTerminalNameAssignmentType, DiodeStandardTerminalMappingType, DiodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>		

#### 4.5.3.18.3.4. Reverse Conducting Triode

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Thyristor/Unidirectional/ReverseConducting/Triode</b>			
diagram				
type	<b>UnidirectionalReverseConductingThyristorTriodeType, UnidirectionalTriodeStandardTerminalNameAssignmentType, UnidirectionalTriodeStandardTerminalMappingType, UnidirectionalTriodeStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Anode</b>	<b>2. Cathode</b>	<b>3. Gate</b>	

#### 4.5.3.19. Transformer

path	PartModel/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transformer
diagram	
type	TransformerFunctionType, TransformerCoil-to-Coil-ArrayType, TransformerCoil-to-CoilFunctionType, TransformerFromCoilType, TransformerToCoilType, TransformerCoil-ArrayType, TransformerCoilType, TransformerStandardTerminalNameAssignmentType, TransformerStandardTerminalNameAssignmentType.

### 4.5.3.19.1. Transformer Coil Mapping

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transformer			
diagram				
type	TransformerCoilType, TransformerStandardTerminalNameAssignmentType, TransformerStandardTerminalMappingType, TransformerStandardTerminalNameType.			
list of enumerate values	Mapping/StandardTerminalName			
	1. Start Wiring	2. End Wiring		

### 4.5.3.20. Transistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor			
diagram				
type	TransistorFunctionType, BipolarJunctionTransistorFunctionType, UnijunctionTransistorFunctionType, FieldEffectTransistorFunctionType, GateBipolarTransistorFunctionType, ProgrammableUnijunctionTransistorFunctionType.			

4.5.3.20.1. Bipolar Junction Transistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction
diagram	<p>The diagram illustrates the UML class hierarchy for Bipolar Junction Transistors. A central class <b>BipolarJunction</b> (type: <i>BipolarJunctionTransistorFunctionType</i>) is connected to four subclasses: <b>NPN</b>, <b>PNP</b>, <b>NPN-Darlington</b>, and <b>PNP-Darlington</b>. Each of these subclasses is further connected to a <b>StandardTerminalNameAssignment</b> class. The subclasses are grouped into four dashed boxes: <b>NPN-BipolarJunctionTransistorType</b>, <b>PNP-BipolarJunctionTransistorType</b>, <b>NPN-DarlingtonFunctionType</b>, and <b>PNP-DarlingtonFunctionType</b>. Each <b>StandardTerminalNameAssignment</b> class has a multiplicity of 1..∞.</p>
type	<b>BipolarJunctionTransistorFunctionType</b> , <b>NPN-BipolarJunctionTransistorType</b> , <b>PNP-BipolarJunctionTransistorType</b> , <b>BipolarTransistorStandardTerminalNameAssignmentType</b> , <b>NPN-DarlingtonFunctionType</b> , <b>PNP-DarlingtonFunctionType</b> , <b>DarlingtonTransistorStandardTerminalNameAssignmentType</b> .

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.



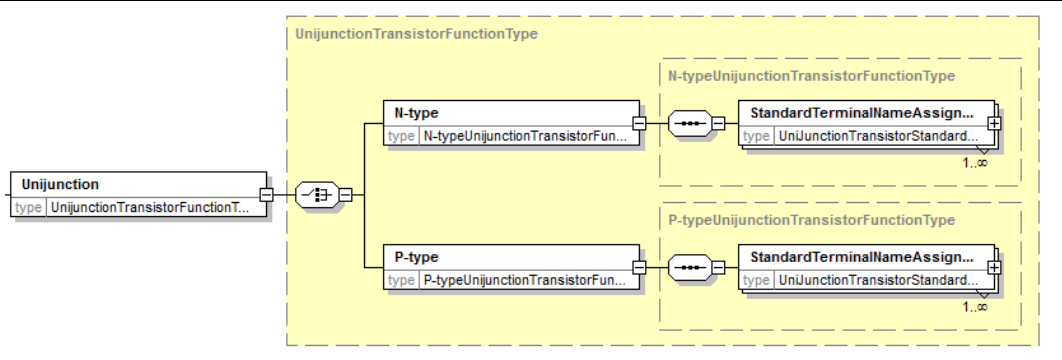
4.5.3.20.1.1. Bipolar Transistor Standard Terminal Name Assignment

path	<div>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction/NPN/StandardTerminalNameAssignment</a></div> <div>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction/PNP/StandardTerminalNameAssignment</a></div>			
diagram	<p>The diagram illustrates the UML class structure for BipolarTransistorStandardTerminalNameAssignmentType. It shows a sequence of associations: StandardTerminalNameAssignmentType (1..∞) to Mapping (3) to BJT-StandardTerminalMappingType. The Mapping class has a type attribute 'BJT-StandardTerminalMappingT...'. The BJT-StandardTerminalMappingType class contains two attributes: 'StandardTerminalName' of type 'BJT-StandardTerminalNameType' and 'TerminalMapID' of type 'xs:string'. A 'constraints' box is also present, indicating additional constraints on the structure.</p>			
type	<a href="#">BipolarTransistorStandardTerminalNameAssignmentType</a> , <a href="#">BJT-StandardTerminalMappingType</a> , <a href="#">BJT-StandardTerminalNameType</a> .			
list of enumerate values	<a href="#">Mapping/StandardTerminalName</a>			
	1. Base	2. Collector	3. Emitter	

#### 4.5.3.20.1.2. Darlington Transistor Standard Terminal Name Assignment

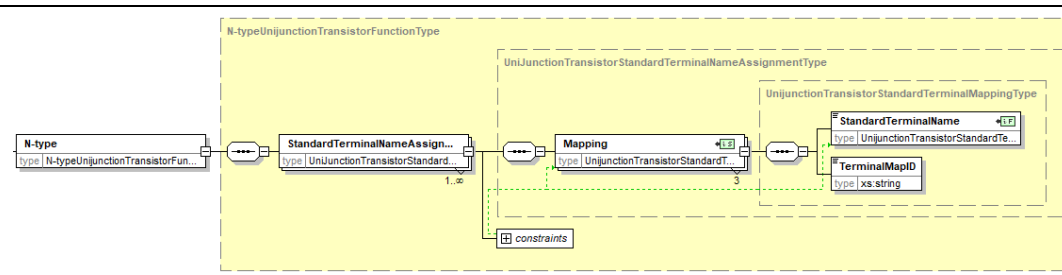
path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction/NPN-Darlington</a></li> <li>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/BipolarJunction/PNP-Darlington</a></li> </ol>			
diagram				
type	<p> <a href="#">DarlingtonTransistorStandardTerminalNameAssignmentType</a>,  <a href="#">DarlingtonTransistorMandatoryStandardTerminalMappingType</a>,  <a href="#">DarlingtonTransistorMandatoryStandardTerminalNameType</a>,  <a href="#">DarlingtonTransistorSingleEmitterStandardTerminalMappingType</a>,  <a href="#">DarlingtonTransistorSingleEmitterStandardTerminalNameType</a>,  <a href="#">DarlingtonTransistorDoubleEmitterStandardTerminalMappingType</a>,  <a href="#">DarlingtonTransistorDoubleEmitterStandardTerminalNameType</a>. </p>			
list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	1. Base	2. Collector		
	<b>SingleEmitterMapping/StandardTerminalName</b>			
	1. Emitter			
	<b>DoubleEmitterMapping/StandardTerminalName</b>			
	1. Emitter 1	2. Emitter 2		

#### 4.5.3.20.2. Unijunction Transistor

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Unijunction</b>
diagram	 <p>The diagram illustrates the structure of a Unijunction Transistor. It shows a central 'Unijunction' block (type: UnijunctionTransistorFunctionT...) connected to a 'UnijunctionTransistorFunctionType' block. This function type is further divided into two sub-types: 'N-type' (type: N-typeUnijunctionTransistorFun...) and 'P-type' (type: P-typeUnijunctionTransistorFun...). Each sub-type is connected to a 'StandardTerminalNameAssign...' block (type: UniJunctionTransistorStandard...). The 'N-type' block is connected to a 'StandardTerminalNameAssign...' block with a multiplicity of 1..∞. The 'P-type' block is connected to a 'StandardTerminalNameAssign...' block with a multiplicity of 1..∞. The entire structure is enclosed in a dashed yellow box.</p>
type	<b>UnijunctionTransistorFunctionType, N-typeUnijunctionTransistorFunctionType, P-typeUnijunctionTransistorFunctionType, UniJunctionTransistorStandardTerminalNameAssignmentType.</b>

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.

##### 4.5.3.20.2.1. N-type Unijunction Transistor

path	<ol style="list-style-type: none"> <li><b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Unijunction/N-type</b></li> <li><b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Unijunction/P-type</b></li> </ol>			
diagram	 <p>The diagram illustrates the structure of an N-type Unijunction Transistor. It shows an 'N-type' block (type: N-typeUnijunctionTransistorFun...) connected to a 'StandardTerminalNameAssign...' block (type: UniJunctionTransistorStandard...). This block is connected to a 'Mapping' block (type: UniJunctionTransistorStandardT...). The 'Mapping' block is connected to a 'UnijunctionTransistorStandardTerminalMappingType' block. This mapping type is further divided into two sub-types: 'StandardTerminalName' (type: UniJunctionTransistorStandardTe...) and 'TerminalMapID' (type: xs:string). The 'StandardTerminalName' block is connected to a 'Mapping' block with a multiplicity of 3. The 'TerminalMapID' block is connected to a 'Mapping' block with a multiplicity of 3. The entire structure is enclosed in a dashed yellow box.</p>			
type	<b>UnijunctionTransistorStandardTerminalNameAssignmentType, UnijunctionTransistorStandardTerminalMappingType, UniJunctionTransistorStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Base 1</b>	<b>2. Base 2</b>	<b>3. Emitter</b>	

4.5.3.20.2.2. P-type Unijunction Transistor

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Unijunction/P-type			
diagram	<p>The diagram illustrates the structure of a P-type Unijunction Transistor. It shows a hierarchy of classes: <b>P-type</b> (type <code>P-typeUnijunctionTransistorFun...</code>) is associated with <b>StandardTerminalNameAssign...</b> (type <code>UnijunctionTransistorStandard...</code>) with a multiplicity of <code>1..∞</code>. <b>StandardTerminalNameAssign...</b> is associated with <b>Mapping</b> (type <code>UnijunctionTransistorStandardT...</code>) with a multiplicity of <code>3</code>. <b>Mapping</b> is associated with <b>UnijunctionTransistorStandardTerminalMappingType</b>, which contains <b>StandardTerminalName</b> (type <code>UnijunctionTransistorStandardTe...</code>) and <b>TerminalMapID</b> (type <code>xs:string</code>). <b>StandardTerminalName</b> has a multiplicity of <code>3</code>. A <b>constraints</b> compartment is also shown.</p>			
type	<b>P-typeUnijunctionTransistorFunctionType,</b> <b>UnijunctionTransistorStandardTerminalNameAssignmentType,</b> <b>UnijunctionTransistorStandardTerminalMappingType,</b> <b>UnijunctionTransistorStandardTerminalNameType.</b>			
list of enumerate values	<b>Mapping/StandardTerminalName</b>			
	<b>1. Base 1</b>	<b>2. Base 2</b>	<b>3. Emitter</b>	

## 4.5.3.20.3. Field Effect Transistor (FET)

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect
diagram	
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.

#### 4.5.3.20.3.1. FET – Standard Terminal Name Assignment

path	<ol style="list-style-type: none"> <li>1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/JunctionGate/N-channel/StandardTerminalNameAssignment</li> <li>2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/JunctionGate/P-channel/StandardTerminalNameAssignment</li> <li>3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/IGFET-EnhancementType/N-channel/StandardTerminalNameAssignment</li> <li>4. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/IGFET-EnhancementType/P-channel/StandardTerminalNameAssignment</li> <li>5. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/IGFET-DepletionType/N-channelIGFET-DepletionType/StandardTerminalNameAssignment</li> <li>6. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/FieldEffect/IGFET-DepletionType/P-channelIGFET-DepletionType/StandardTerminalNameAssignment</li> </ol>
diagram	<p>The diagram illustrates the structure of the <b>FET-StandardTerminalNameAssignmentType</b> and its associated mapping types. The main class, <b>StandardTerminalNameAssign...</b> (type <b>FET-StandardTerminalNameAss...</b>), has a multiplicity of 1..∞. It is associated with three mapping types: <b>MandatoryMapping</b> (type <b>FET-MandatoryStandardTermin...</b>), <b>SingleGateMapping</b> (type <b>FET-SingleGateStandardTermin...</b>), and <b>DoubleGateMapping</b> (type <b>FET-DoubleGateStandardTermin...</b>). Each mapping type is associated with a <b>StandardTerminalName</b> (type <b>FET-MandatoryStandardTerminal...</b>, <b>FET-SingleGateStandardTerminal...</b>, and <b>FET-DoubleGateStandardTermin...</b> respectively) and a <b>TerminalMapID</b> (type <b>xs:string</b>). The <b>DoubleGateMapping</b> type has a multiplicity of 2. A <b>constraints</b> box is also present at the bottom left of the diagram area.</p>
type	<p><b>FET-StandardTerminalNameAssignmentType, FET-MandatoryStandardTerminalMappingType, FET-MandatoryStandardTerminalNameType, FET-SingleGateStandardTerminalMappingType, FET-SingleGateStandardTerminalNameType, FET-DoubleGateStandardTerminalMappingType, FET-DoubleGateStandardTerminalNameType.</b></p>

#### 4.5.3.20.3.1 FET – Standard Terminal Name Assignment (cont'd)

list of enumerate values	<b>MandatoryMapping/StandardTerminalName</b>			
	<b>1. Drain</b>	<b>2. Source</b>		
	<b>SingleEmitterMapping/StandardTerminalName</b>			
	<b>3. Gate</b>			
	<b>DoubleEmitterMapping/StandardTerminalName</b>			
	<b>1. Gate 1</b>	<b>2. Gate 2</b>		

#### 4.5.3.20.4. Insulated Gate Bipolar Transistor (IGBT)

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar</b>
diagram	
type	<b>Insulated-GateBipolarTransistorFunctionType, IGBT-EnhancementFunctionType, N-channel-IGBT-EnhancementType, P-channel-IGBT-EnhancementType, IGBT-DepletionFunctionType, N-channel-IGBT-DepletionFunctionType, P-channel-IGBT-DepletionFunctionType, IGBT-StandardTerminalNameAssignmentType.</b>

#### 4.5.3.20.4.1. IGBT – Standard Terminal Name Assignment Type

path	<ol style="list-style-type: none"> <li>1. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar/IGBT-EnhancementType/N-channel/IGBT-StandardTerminalNameAssignment</li> <li>2. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar/IGBT-EnhancementType/P-channel/IGBT-StandardTerminalNameAssignment</li> <li>3. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar/IGBT-DepletionType/N-channel/IGBT-StandardTerminalNameAssignment</li> <li>4. PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/Insulated-GateBipolar/IGBT-DepletionType/P-channel/IGBT-StandardTerminalNameAssignment</li> </ol>			
diagram				
type	<b>IGBT-StandardTerminalNameAssignmentType, IGBT-StandardTerminalMappingType,</b> <b>IGBT-StandardTerminalNameType.</b>			
	<b>Mapping/StandardTerminalName</b>			
	1. Gate	2. Collector	3. Emitter	

#### 4.5.3.20.5. Programmable Unijunction Transistor

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/ProgrammableUnijunction</b>			
diagram				
type	<b>ProgrammableUnijunctionTransistorFunctionType,</b> <b>N-typeProgrammableUnijunctionTransistorFunctionType,</b> <b>P-typeProgrammableUnijunctionTransistorFunctionType,</b> <b>ProgrammableUnijunctionTransistorStandardTerminalNameAssignmentType,</b> <b>ProgrammableUnijunctionTransistorStandardTerminalNameMappingType.</b>			



#### 4.5.3.20.5.1. Terminal Mapping

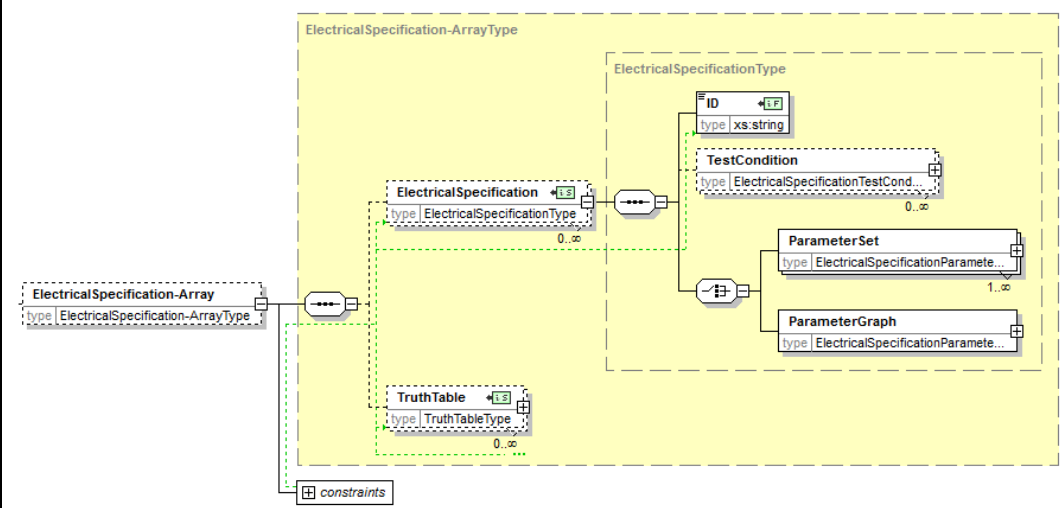
path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/ProgrammableUnijunction/N-type/ProgrammableUnijunctionTransistorStandardTerminalNameAssignment</a></li> <li>2. <a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/Transistor/ProgrammableUnijunction/P-type/ProgrammableUnijunctionTransistorStandardTerminalNameAssignment</a></li> </ol>			
diagram				
type	<a href="#">ProgrammableUnijunctionTransistorStandardTerminalMappingType</a> , <a href="#">ProgrammableUnijunctionTransistorStandardTerminalNameType</a> .			
	<a href="#">Mapping/StandardTerminalName</a>			
	1. Gate	2. Anode	3. Cathode	

#### 4.5.3.21. Other Standard Electrical Functions

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/FunctionGroup-Array/Function/OtherStandard</a>			
diagram				
type	<a href="#">FunctionMap-to-StandardNameType</a> , <a href="#">OtherStandardFunctionStandardTerminalNameAssignmentType</a> , <a href="#">TerminalStandardType</a> .			

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.

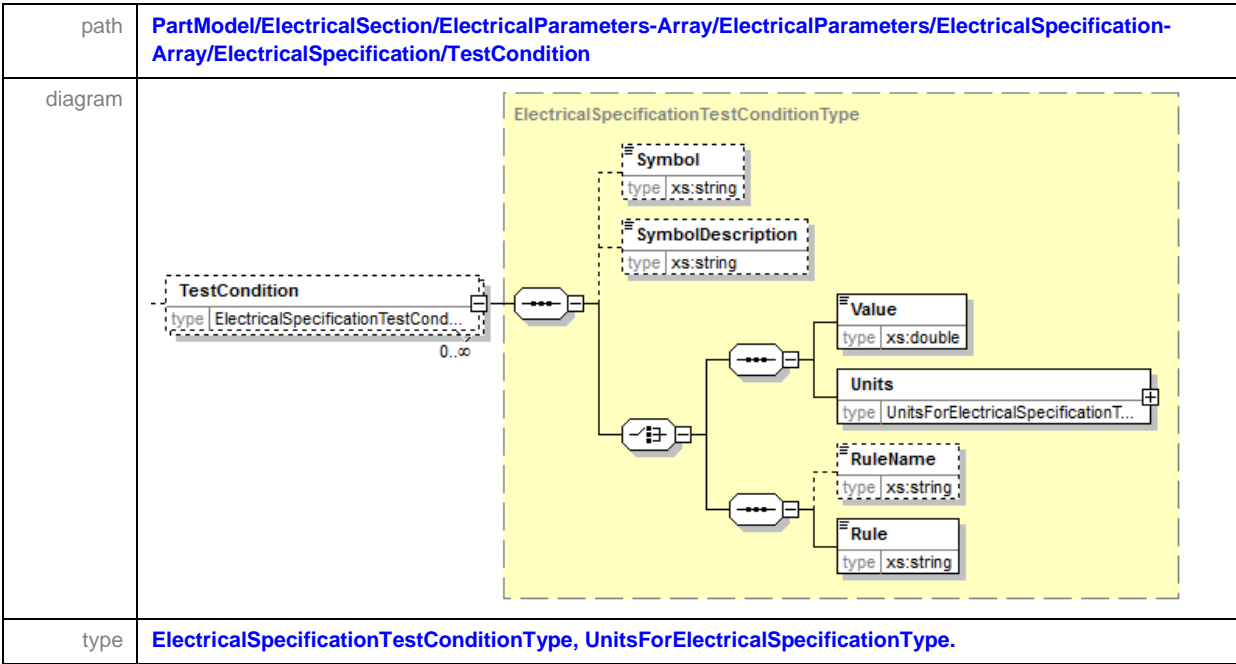
4.5.4. Electrical Specification

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array
diagram	 <p>The diagram illustrates the structure of the <b>ElectricalSpecification-Array</b>. It is a container for <b>ElectricalSpecification-ArrayType</b> elements. The array is split into two main sections: <b>ElectricalSpecification</b> and <b>TruthTable</b>. The <b>ElectricalSpecification</b> section contains an <b>ElectricalSpecificationType</b> element, which is further divided into <b>TestCondition</b> and <b>ParameterSet</b> or <b>ParameterGraph</b>. The <b>TestCondition</b> element has an <b>ID</b> attribute of type <b>xs:string</b> and a <b>type</b> attribute of type <b>ElectricalSpecificationTestCond...</b>. The <b>ParameterSet</b> element has a <b>type</b> attribute of type <b>ElectricalSpecificationParamete...</b>. The <b>ParameterGraph</b> element has a <b>type</b> attribute of type <b>ElectricalSpecificationParamete...</b>. The <b>TruthTable</b> section contains a <b>TruthTableType</b> element. The diagram also shows a <b>constraints</b> section at the bottom.</p>
type	ElectricalSpecification-ArrayType, ElectricalSpecificationType, ElectricalSpecificationTestConditionType, ElectricalSpecificationParameterSetType, ElectricalSpecificationParameterGraphType, TruthTableType.

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 ( $T_A = 25\text{ }^{\circ}\text{C}$ ) in which a set of parameters are defined.

4.5.4.1. Test Condition



An example formula ( $di/dt \leq 70 \text{ A}/\mu\text{s}$ ) is shown below in its XML representation.

```
<TestCondition>
  <SymbolString>di/dt</SymbolString>
  <SymbolDescription>rate of rise of the current</SymbolDescription>
  <Rule>\leq 70 A/\mu s</Rule>
</TestCondition>
```

NOTE The *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>
  <Rule>V_{DD} \leq V_{DS}</Rule>
</TestCondition>
```

4.5.4.1.1. Units

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/TestCondition/Units
diagram	<p>The diagram illustrates the structure of the 'Units' class, which is a specialization of 'UnitsForElectricalSpecificationType'. It is composed of various unit types, each represented by a class with its own specific UOMType. The units listed are: Ampere-Hour, Candela, Capacitance, Coulomb, Current, Decibel, Decibel-Meter, Dimension, Frequency, Inductance, Joule, PartPerMillion, Percent, Power, Resistance, Temperature, ThermalResistance, Time, Voltage, Voltage-per-Second, and ComplexUOM. The 'ComplexUOM' class is shown with a self-referencing association.</p>
type	UnitsForElectricalSpecificationType, CapacitanceUOMType, CurrentUOMType, FrequencyUOMType, PowerUOMType, ResistanceUOMType, TemperatureUOMType, ThermalResistanceUOMType, Time-in-uSec-to-Sec-UOMType, VoltageUOMType, ComplexUOMType.

#### 4.5.4.1.1. 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

[illegible]

4.5.4.1.1.1. Complex UOM

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/TestCondition/Units/ComplexUOM
diagram	
type	ComplexUOMType, FactorType.

Other UOM can be defined under the *ComplexUOM* branch. An example formula ( $di/dt = 100 \text{ A}/\mu\text{s}$ ) is shown below in its XML representation.

```
<TestCondition>
  <SymbolString>di/dt</SymbolString>
  <SymbolDescription>rate of rise of the current</SymbolDescription>
  <Value>100</Value>
  <Units>
    <ComplexUOM>
      <Factor>
        <Prefix>Unity</Prefix>
        <UOM>Ampere</UOM>
      </Factor>
      <Factor>
        <Prefix>Micro</Prefix>
        <UOM>Second</UOM>
        <Exponent>-1</Exponent>
      </Factor>
    </ComplexUOM>
  </Units>
</TestCondition>
```

#### 4.5.4.2. Parameter Set

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterSet</a>
diagram	<pre> classDiagram     class ParameterSet {         type ElectricalSpecificationParameter...     }     class TestCondition {         type ElectricalSpecificationTestCond...     }     class Parameter {         type ElectricalSpecificationParamete...     }     ParameterSet "1..∞" -- "0..∞" TestCondition     TestCondition "0..∞" -- "1..∞" Parameter     </pre>
type	<a href="#">ElectricalSpecificationParameterSetType</a> , <a href="#">ElectricalSpecificationTestConditionType</a> , <a href="#">ElectricalSpecificationParameterType</a> .

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](#) occurred at an ambient temperature ( $T_A = 25$  °C), therefore the ambient temperature test condition is set at the [ElectricalSpecification/TestCondition](#) branch.

4.5.4.2.1. Parameter

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterSet/Parameter
diagram	<p>The diagram illustrates the structure of the <b>Parameter</b> class and its associated <b>ElectricalSpecificationParameterType</b> class. The <b>Parameter</b> class is associated with the <b>ElectricalSpecificationParameterType</b> class, which is a complex type containing several nested classes. The <b>ElectricalSpecificationParameterType</b> class is shown as a dashed box containing the following nested classes:</p> <ul style="list-style-type: none"><li><b>TestCondition</b>: type <code>ElectricalSpecificationTestCond...</code>, multiplicity <code>0..∞</code>.</li><li><b>Symbol</b>: type <code>xs:string</code>.</li><li><b>SymbolDescription</b>: type <code>xs:string</code>.</li><li><b>Values</b>: type <code>ValueSetType</code>.</li><li><b>Units</b>: type <code>UnitsForElectricalSpecificationT...</code>.</li><li><b>RuleName</b>: type <code>xs:string</code>.</li><li><b>Rule</b>: type <code>xs:string</code>.</li></ul> <p>The <b>Parameter</b> class is associated with the <b>ElectricalSpecificationParameterType</b> class, with a multiplicity of <code>1..∞</code>.</p>
type	<b>ElectricalSpecificationParameterType</b> , <b>ElectricalSpecificationTestConditionType</b> , <b>ValueSetType</b> , <b>UnitsForElectricalSpecificationType</b> .



#### 4.5.4.3. Parameter Graph

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph</a>
diagram	
type	<a href="#">ElectricalSpecificationParameterGraphType</a> , <a href="#">ElectricalParameterGraphChartXAxisType</a> , <a href="#">ElectricalParameterGraphChartYAxisType</a> , <a href="#">GraphFormattingType</a> , <a href="#">UnitsForElectricalSpecificationType</a> , <a href="#">ElectricalSpecificationParameterGraphData-ArrayType</a> , <a href="#">GraphChartXAxisFormattingType</a> , <a href="#">GraphChartYAxisFormattingType</a> .

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 [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.

#### 4.5.4.3. Parameter Graph (cont'd)

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*).

##### 4.5.4.3.1. 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	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/TestConditionDefinition/Formatting.</a>
diagram	<pre> graph TD     Formatting["Formatting type: GraphChartXAxisFormattingType"]     Range["Range type: GraphAxisRangeType"]     Inverted["Inverted type: EmptyType"]     Scale["Scale type: GraphAxisScaleType"]     Position["Position type: GraphChartXAxisPositionType"]     Minimum["Minimum type: xs:int"]     Maximum["Maximum type: xs:int"]     Linear["Linear type: GraphAxisScaleLinearType"]     Logarithmic["Logarithmic type: GraphAxisScaleLogarithmicType"]     Step["Step type: xs:float"]     Natural["Natural type: xs:string"]     Base["Base type: xs:float default: 10.0"]      Formatting --- Range     Formatting --- Inverted     Formatting --- Scale     Formatting --- Position     Range --- Minimum     Range --- Maximum     Scale --- Linear     Scale --- Logarithmic     Linear --- Step     Logarithmic --- Natural     Logarithmic --- Base   </pre>
type	<a href="#">GraphChartXAxisFormattingType</a> , <a href="#">GraphAxisRangeType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">GraphAxisScaleType</a> , <a href="#">GraphAxisScaleLinearType</a> , <a href="#">GraphAxisScaleLogarithmicType</a> , <a href="#">GraphChartXAxisPositionType</a> .

## 4.5.4.3.1. Formatting (cont'd)

path	<b>PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/ParameterDefinition/Formatting.</b>
diagram	<pre> classDiagram     class GraphChartYAxisFormattingType {         Range         Inverted         Scale         Position     }     class GraphAxisRangeType {         Minimum         Maximum     }     class GraphAxisScaleType {         Linear         Logarithmic     }     class GraphAxisScaleLinearType {         Step     }     class GraphAxisScaleLogarithmicType {         Natural         Base     }     class GraphChartYAxisPositionType {     }      GraphChartYAxisFormattingType "1" -- "*" GraphAxisRangeType : Range     GraphChartYAxisFormattingType "1" -- "*" EmptyType : Inverted     GraphChartYAxisFormattingType "1" -- "*" GraphAxisScaleType : Scale     GraphChartYAxisFormattingType "1" -- "*" GraphChartYAxisPositionType : Position     GraphAxisRangeType "1" -- "*" xs:int : Minimum     GraphAxisRangeType "1" -- "*" xs:int : Maximum     GraphAxisScaleType "1" -- "*" GraphAxisScaleLinearType : Linear     GraphAxisScaleType "1" -- "*" GraphAxisScaleLogarithmicType : Logarithmic     GraphAxisScaleLinearType "1" -- "*" xs:float : Step     GraphAxisScaleLogarithmicType "1" -- "*" xs:string : Natural     GraphAxisScaleLogarithmicType "1" -- "*" xs:float : Base     GraphAxisScaleLogarithmicType "1" -- "10.0" : default   </pre>
type	<b>GraphChartYAxisFormattingType, GraphAxisRangeType, JEP30-D10:EmptyType, GraphAxisScaleType, GraphAxisScaleLinearType, GraphAxisScaleLogarithmicType, GraphChartYAxisPositionType.</b>

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	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/Formatting</a>
diagram	
type	<a href="#">GraphFormattingType</a> , <a href="#">GraphDisplayType</a> , <a href="#">GraphLegendType</a> , <a href="#">GraphLegendLocationType</a> , <a href="#">GraphLegendVerticalPositionType</a> , <a href="#">GraphLegendHorizontalPositionType</a> .

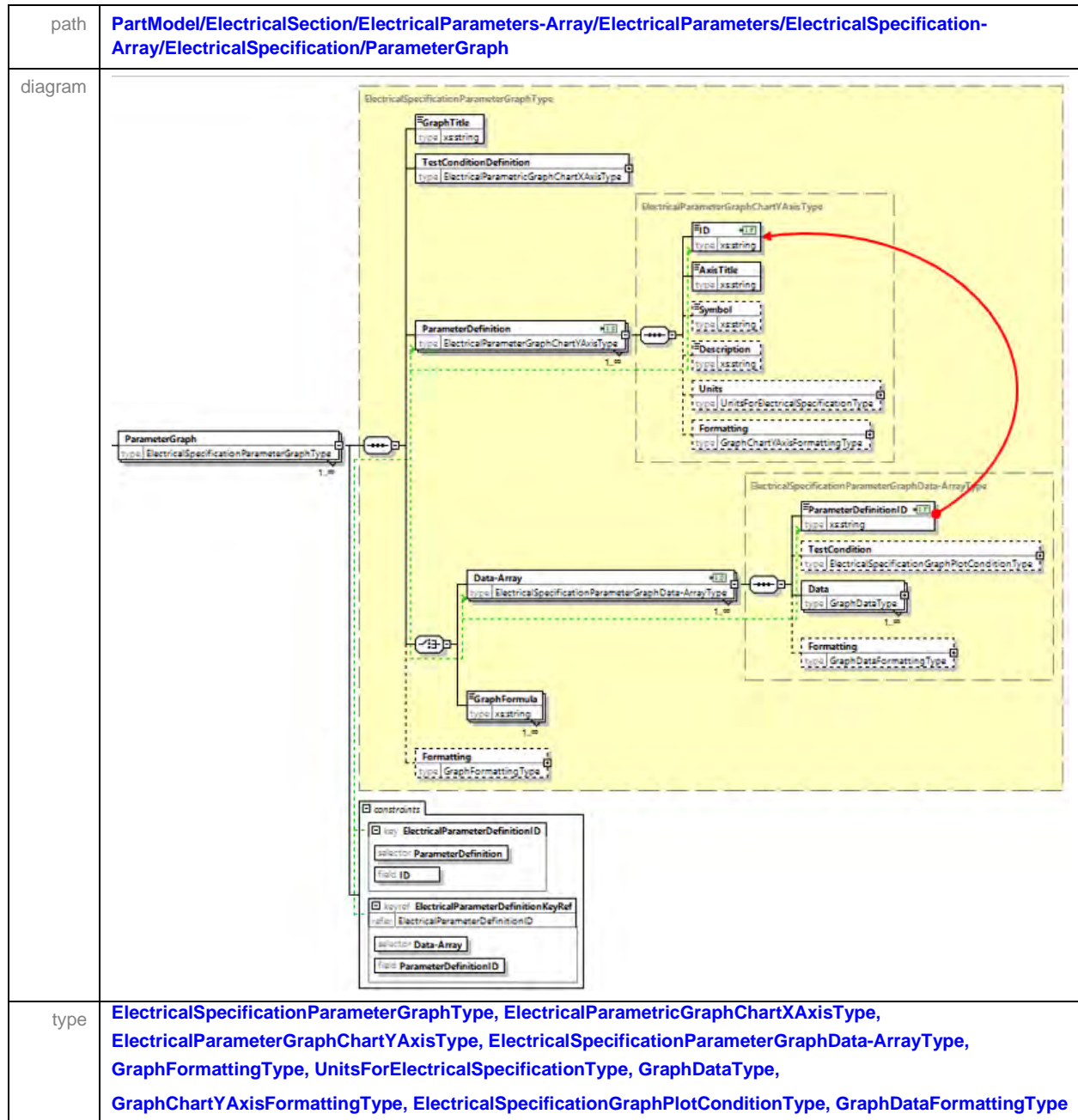
The body of the graph can be formatted under the [GraphFormattingType](#). The [DisplayType](#) enumerated list is

- 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.

#### 4.5.4.3.1.1. Linking the Data-Array to the Appropriate Parameter Definition

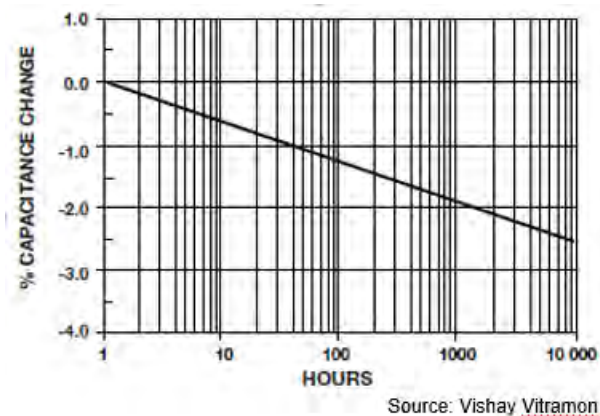


When populating the [Data-Array](#) for a given graph, the set of data is referenced to the specific [ParameterDefinition](#) via the [ParameterDefinitionID](#). The process is replicated for each [ParameterDefinition](#) if there are 2 or more vertical [ParameterDefinition](#) axis defined. Each set of data captured under the [Data-Array](#) is now tied to the appropriate [ParameterDefinition](#) axis.

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](#).

#### 4.5.4.3.1.1. Linking the Data-Array to the Appropriate Parameter Definition (cont'd)

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>
    <Symbol>h</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>
```

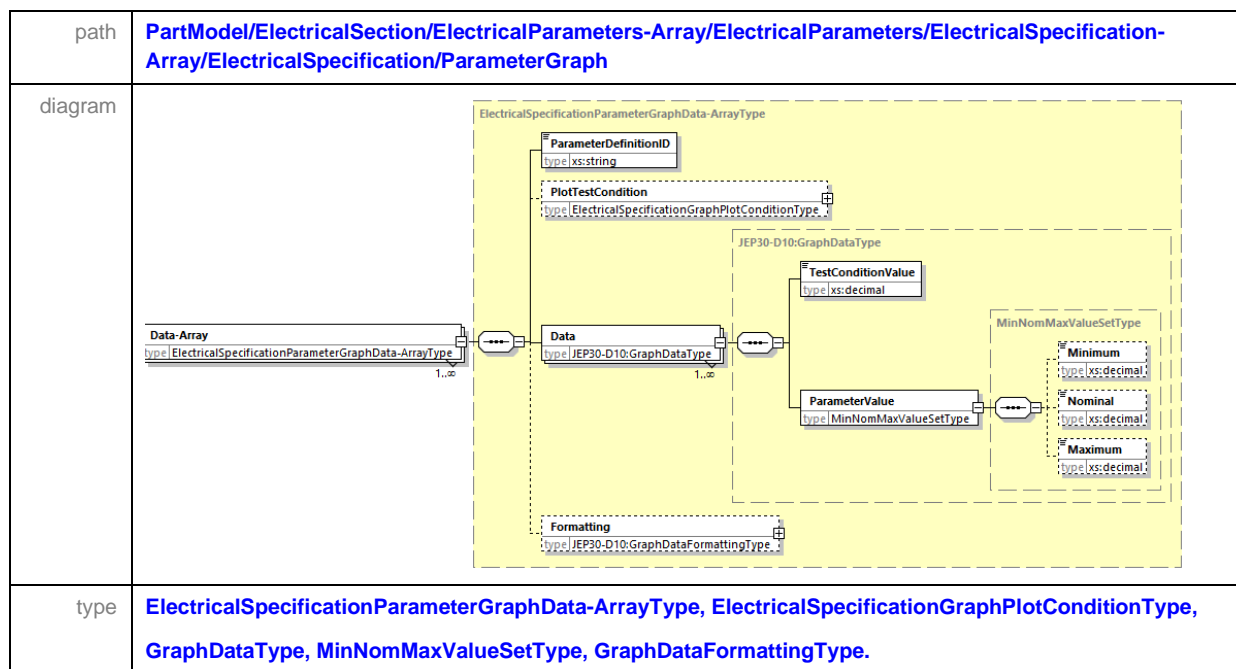
#### 4.5.4.3.1.1. Linking the Data-Array to the Appropriate Parameter Definition (cont'd)

```

</TestConditionDefinition>
<ParameterDefinition>
  <ID>Plot1</ID>
  <AxisTitle>% CAPACITANCE CHANGE</AxisTitle>
  <Symbol>\Delta C</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>log10(h ^ -0.625)</GraphFormula>
</ParameterGraph>

```

#### 4.5.4.3.2. Data-Array



#### 4.5.4.3.2. Data - Array (cont'd)

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.

path	PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ElectricalSpecification-Array/ElectricalSpecification/ParameterGraph/Data-Array/TestCondition.	
diagram		
type	ElectricalSpecificationGraphPlotConditionType, UnitsForElectricalSpecificationType, JEP30-D10:GraphDataFormattingLegendType.	

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.



#### 4.5.4.3.2. Data-Array (cont'd)

path	<b>PartModel/ThermalSection/ThermalFamily-Array/ThermalFamily/ThermalData/ThermalMetrics-Array/ThermalMetricGraph/Data-Array/Formatting.</b>
diagram	<pre> classDiagram     class GraphDataFormattingType {     }     class GraphDataFormattingPointType {         Color : type GraphDataFormattingColorType         Style : type GraphDataFormattingPointStyleType     }     class GraphDataFormattingLineType {         Color : type GraphDataFormattingColorType         Style : type GraphDataFormattingLineStyleType     }     class Formatting {         type GraphDataFormattingType     }     GraphDataFormattingType &lt; -- GraphDataFormattingPointType     GraphDataFormattingType &lt; -- GraphDataFormattingLineType     Formatting ..&gt; GraphDataFormattingType   </pre>
type	<b>GraphDataFormattingType, GraphDataFormattingPointType, GraphDataFormattingLineType, GraphDataFormattingColorType, GraphDataFormattingPointStyleType, GraphDataFormattingLineStyleType.</b>

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.

## 4.5.4.3.2. Data-Array (cont'd)

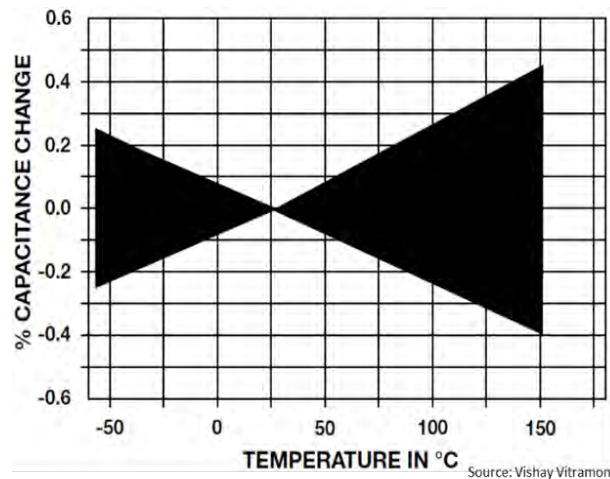


Figure 43 — Capacitance Value versus Temperature

```

<ParameterGraph>
  <GraphTitle>Capacitance Value versus Temperature</GraphTitle>
  <TestConditionDefinition>
    <AxisTitle>Temperature</AxisTitle>
    <Symbol>T</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>
    <Symbol> $\Delta C$ </Symbol>
    <Description>Capacitance change</Description>
    <Units>
      <Percent>
    </Units>
  </ParameterDefinition>

```

**4.5.4.3.2. 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>

```

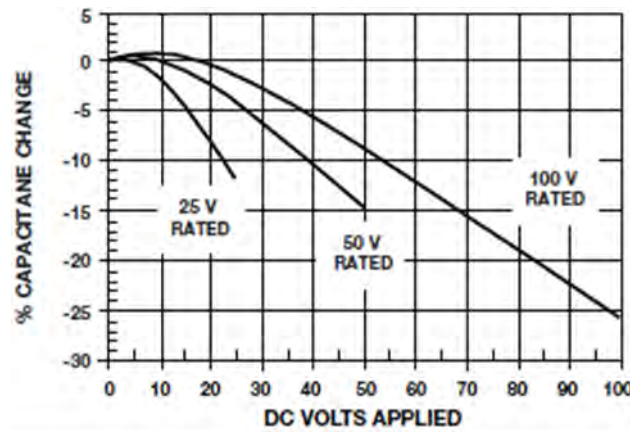
**4.5.4.3.2. Data-Array (cont'd)**

```

<Formatting>
  <DisplayType>Line</DisplayType>
  <Legend>
    <Location>Line</Location>
    <VerticalPosition>Line</VerticalPosition>
    <HorizontalPosition>Line</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>
    <Symbol>V_{DC}</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>
  </ParameterType>

```

**4.5.4.3.2. Data-Array (cont'd)**

```

<Symbol>\Delta C</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>
  <TestCondition>
    <Symbol>VDC</Symbol>
    <SymbolDescription>Rated Voltage DC</SymbolDescription>
    <Value>25</Value>
    <Units>
      <Voltage>V</Voltage>
    </Units>
  </TestCondition>
  <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>

```

**4.5.4.3.2. Data-Array (cont'd)**

```

    <Data>
      <TestConditionValue>25</TestConditionValue>
      <ParameterValue>
        <Nominal>-12</Nominal>
      </ParameterValue>
    </Data>
  </Data-Array>
  <Data-Array>
    <TestCondition>
      <Symbol>VDC</Symbol>
      <SymbolDescription>Rated Voltage DC</SymbolDescription>
      <Value>50</Value>
      <Units>
        <Voltage>V</Voltage>
      </Units>
    </TestCondition>
    <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>
    <TestCondition>
      <Symbol>VDC</Symbol>
      <SymbolDescription>Rated Voltage DC</SymbolDescription>
      <Value>100</Value>

```

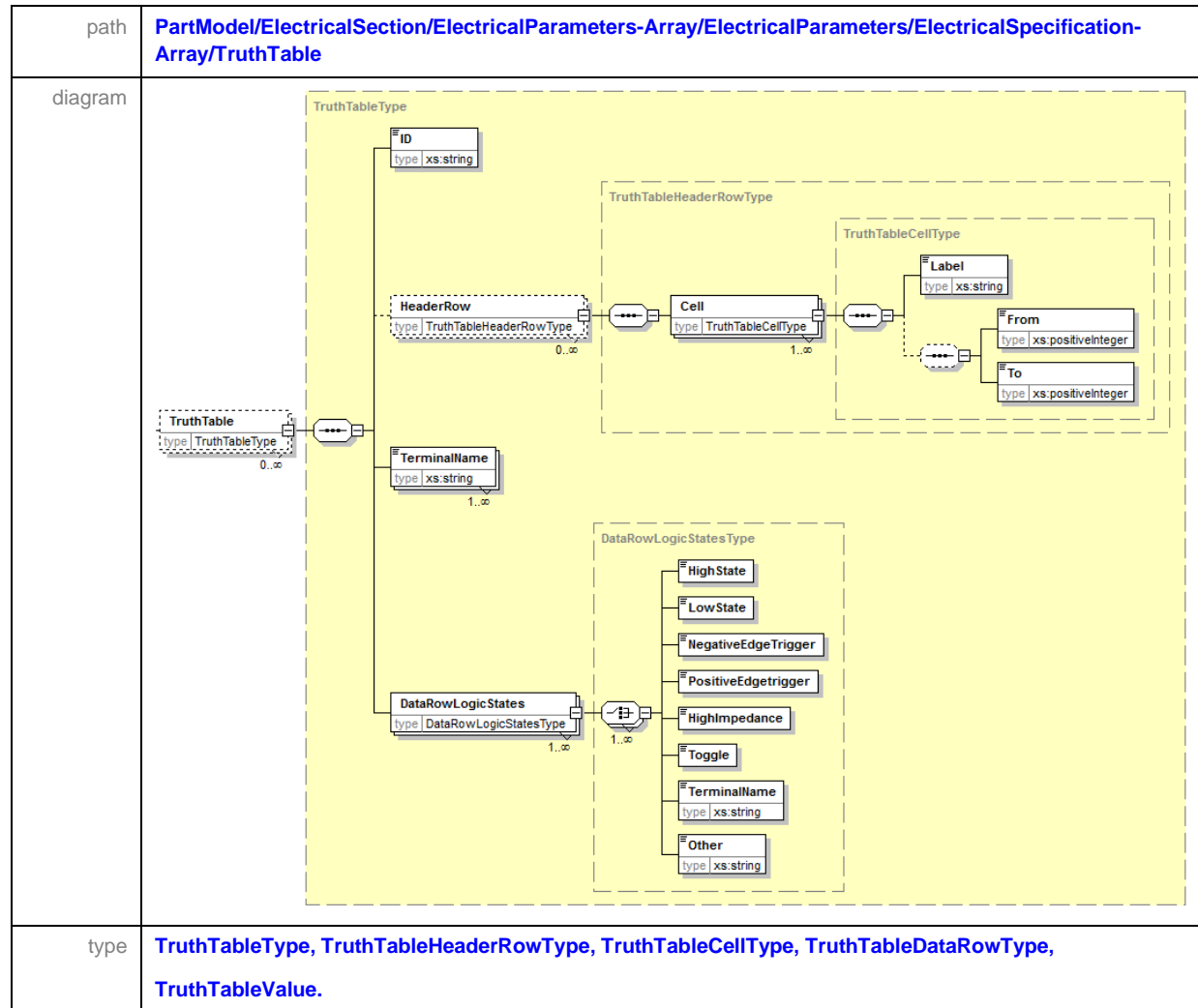
**4.5.4.3.2. Data-Array (cont'd)**

```

        <Units>
          <Voltage>V</Voltage>
        </Units>
      </TestCondition>
    <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>

```

#### 4.5.5. Truth Table



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.



#### 4.5.5 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>

```

#### 4.5.5 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"

#### 4.5.6. ESD

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array</a>
diagram	<pre> classDiagram     class ESD_Array {         type ESD_ArrayType     }     class ESD {         type ESDType         ID xs:string     }     class HBM_HumanBodyModel {         type HBM_Type     }     class CDM_ChargedDeviceModel {         type CDM_Type     }     ESD_Array "1" -- "0..∞" ESD     ESD "1" -- "1" HBM_HumanBodyModel     ESD "1" -- "1" CDM_ChargedDeviceModel     </pre>
type	<a href="#">ESD-ArrayType</a> , <a href="#">ESDType</a> , <a href="#">HBM-Type</a> , <a href="#">CDM-Type</a> .

#### 4.5.6.1. HBM – Human Body Model

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array/ESD/HBM-HumanBodyModel</a>
diagram	
type	<a href="#">HBM-Type</a> , <a href="#">HBM-ClassificationType</a> , <a href="#">ESD-VoltageRangeType</a> , <a href="#">StaticVoltageUOMType</a> .

[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.

#### 4.5.6.2. CDM – Charged Device Model

path	<a href="#">PartModel/ElectricalSection/ElectricalParameters-Array/ElectricalParameters/ESD-Array/ESD/CDM-ChargedDeviceModel</a>
diagram	
type	<a href="#">CDM-Type</a> , <a href="#">CDM-ClassificationType</a> , <a href="#">ESD-VoltageRangeType</a> , <a href="#">StaticVoltageUOMType</a> .

[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.

## 4.6. Schematic Data - Array

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array</a>
diagram	
type	<a href="#">SchematicData-Array</a> , <a href="#">SchematicDataType</a> , <a href="#">Symbol-ArrayType</a> , <a href="#">RequiredCircuitry-ArrayType</a> , <a href="#">ds:SignatureType</a>

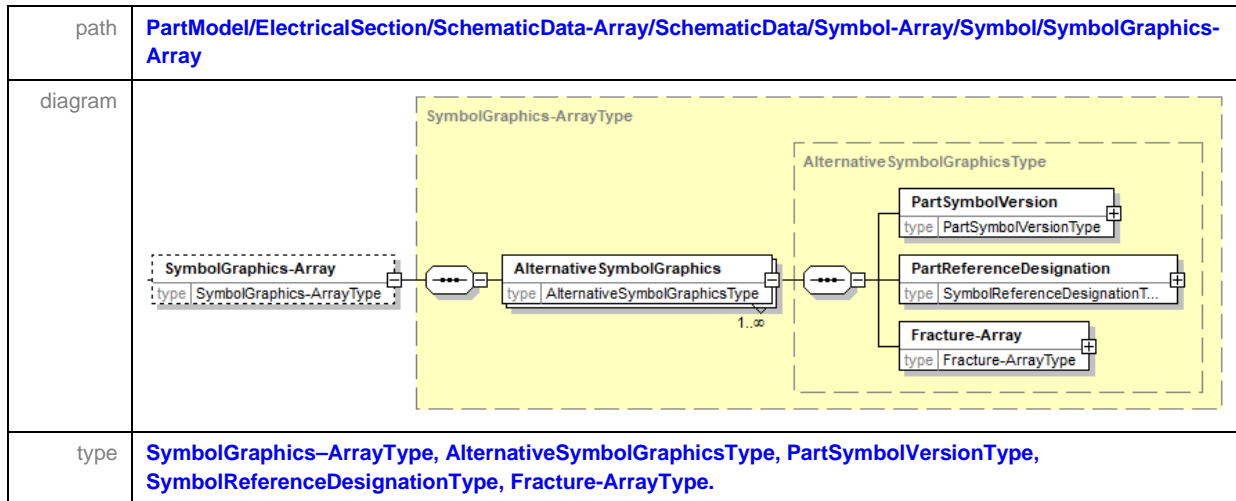
### 4.6.1. Symbol - Array

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array</a>
diagram	
type	<a href="#">Symbol-ArrayType</a> , <a href="#">SymbolType</a> , <a href="#">SymbolGraphics-ArrayType</a> , <a href="#">ds:SignatureType</a>

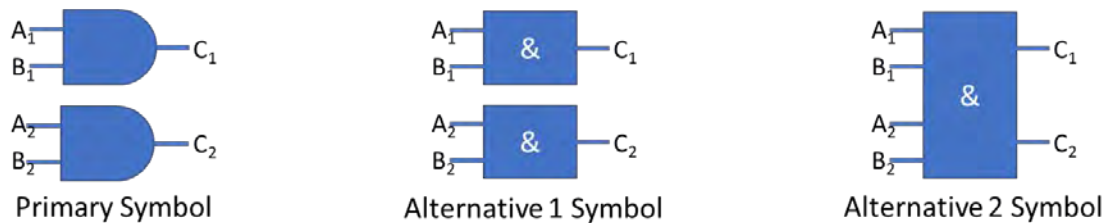
[Symbol](#) data used in Schematic design is not normally provided by the component manufacturer. However, its provision can enhance the efficiency of utilizing that Part within a design process. There are various drafting standards to which a symbol can be generated to, such as ANSI, IEEE or IEC Drafting 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.

#### 4.6.1.1. Symbol Graphics – Array



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*

4.6.1.1.1. Part Symbol Version

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/PartSymbolVersion
diagram	
type	PartSymbolVersionType.

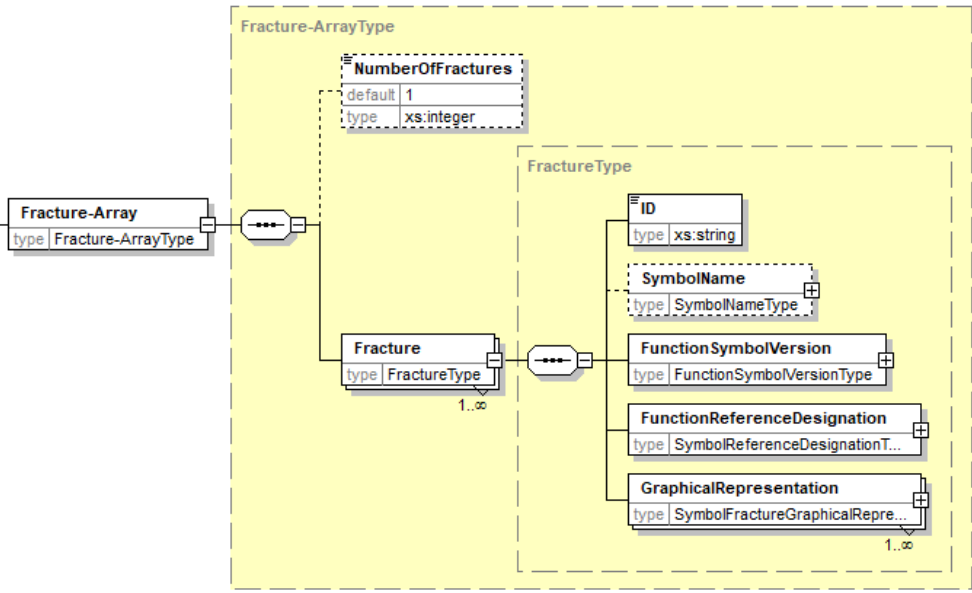
*PartSymbolVersion captures the version number of the symbol, timestamp of the latest changes, change description, and reason for change.*

#### 4.6.1.1.2. Part Reference Designation

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/PartReferenceDesignation</a>		
diagram			
type	<a href="#">SymbolReferenceDesignationType</a> , <a href="#">Symbol-ANSI-RefDesType</a> , <a href="#">Symbol-IEC-RefDesType</a> .		
list of enumerate values	<b>ANSI-Standard</b>		
	<b>A</b>	<b>AR</b>	<b>AT</b>
	<b>BT</b>	<b>C</b>	<b>CB</b>
	<b>CR</b>	<b>D</b>	<b>DC</b>
	<b>DS</b>	<b>E</b>	<b>F</b>
	<b>FL</b>	<b>G</b>	<b>H</b>
	<b>J</b>	<b>K</b>	<b>L</b>
	<b>LS</b>	<b>MK</b>	<b>MT</b>
	<b>P</b>	<b>PS</b>	<b>Q</b>
	<b>R</b>	<b>RE</b>	<b>RT</b>
	<b>RV</b>	<b>S</b>	<b>T</b>
	<b>TC</b>	<b>TR</b>	<b>U</b>
	<b>V</b>	<b>VR</b>	<b>Y</b>
list of enumerate values	<b>IEC-Standard</b>		
	<b>B</b>	<b>G</b>	<b>C</b>
	<b>Q</b>	<b>F</b>	<b>V</b>
	<b>R</b>	<b>Z</b>	<b>L</b>
	<b>A</b>	<b>U</b>	<b>K</b>
	<b>S</b>	<b>T</b>	

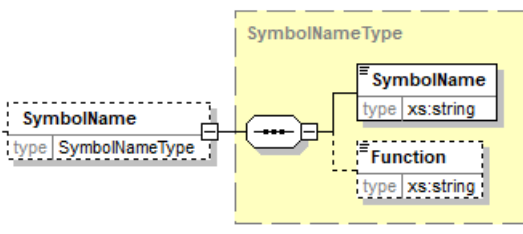
[PartReferenceDesignation](#) captures the reference designator based on the ANSI standard or the IEC standard. If the reference designator is not defined by either of those standards, [Other](#) reference designator may be defined. If the symbol is made up of multiple fractures, and each [FractureReferenceDesignation](#) is the same, then that value can roll up to be the same for the [PartReferenceDesignation](#).

4.6.1.1.3. Fracture – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array
diagram	 <p>The diagram illustrates the structure of the <b>Fracture-Array</b> type. It is a container type that holds an array of <b>Fracture</b> types. The <b>Fracture-Array</b> type is shown with a <b>NumberOfFractures</b> attribute (default 1, type xs:integer) and a <b>Fracture</b> element (type FractureType, cardinality 1..∞). The <b>FractureType</b> is a complex type containing an <b>ID</b> attribute (type xs:string), a <b>SymbolName</b> attribute (type SymbolNameType, cardinality 1), and three optional elements: <b>FunctionSymbolVersion</b> (type FunctionSymbolVersionType), <b>FunctionReferenceDesignation</b> (type SymbolReferenceDesignationType), and <b>GraphicalRepresentation</b> (type SymbolFractureGraphicalRepresentationType, cardinality 1..∞).</p>
type	Fracture-ArrayType, FractureType, SymbolNameType, FunctionSymbolVersionType, SymbolReferenceDesignationType, SymbolFractureGraphicalRepresentationType.

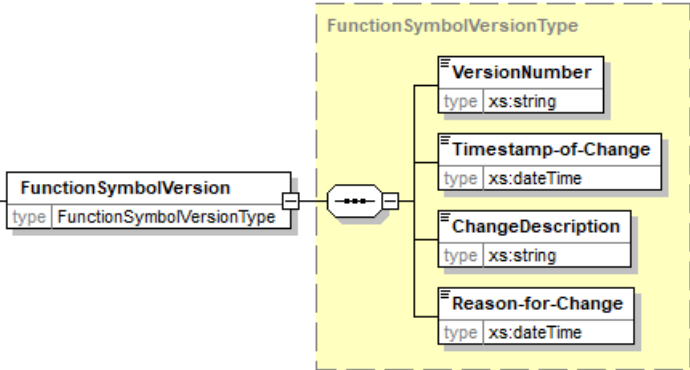
*Fracture-Array* contains fractures of symbols. When there are many terminals on a Part, it might be more practical to split the symbol into several Fractures. Terminal are then typically assigned to a *Fracture*, based on some logical organization of the terminals. The fracture name is a string that 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.

4.6.1.1.3.1. Symbol Name

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/SymbolName
diagram	 <p>The diagram illustrates the structure of the <b>SymbolName</b> type. It is a complex type containing an <b>SymbolName</b> attribute (type xs:string) and an optional <b>Function</b> attribute (type xs:string).</p>
type	SymbolNameType.

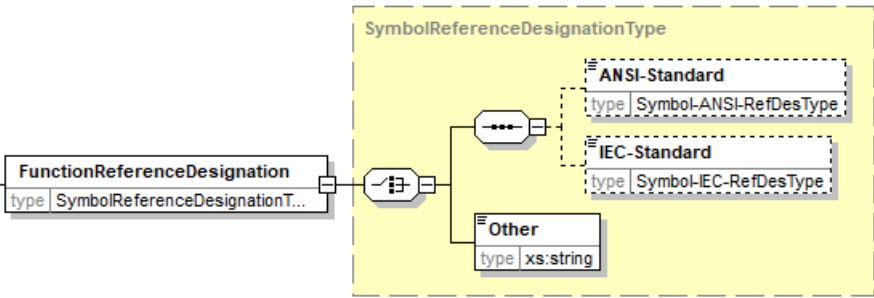


#### 4.6.1.1.3.1.1. Function Symbol Version

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/FunctionSymbolVersion</a>
diagram	 <p>The diagram shows a class <b>FunctionSymbolVersion</b> with a type <b>FunctionSymbolVersionType</b>. This class is connected via a composition relationship (indicated by a solid line with a filled square at the <b>FunctionSymbolVersion</b> end and a hollow diamond at the <b>FunctionSymbolVersionType</b> end) to a complex type <b>FunctionSymbolVersionType</b>. This complex type is enclosed in a dashed yellow box and contains four elements: <b>VersionNumber</b> (type <b>xs:string</b>), <b>Timestamp-of-Change</b> (type <b>xs:dateTime</b>), <b>ChangeDescription</b> (type <b>xs:string</b>), and <b>Reason-for-Change</b> (type <b>xs:dateTime</b>).</p>
type	<a href="#">FunctionSymbolVersionType.</a>

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.

#### 4.6.1.1.3.2. Function Reference Designation

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/FunctionReferenceDesignation</a>
diagram	 <p>The diagram shows a class <b>FunctionReferenceDesignation</b> with a type <b>SymbolReferenceDesignationT...</b>. This class is connected via a composition relationship (indicated by a solid line with a filled square at the <b>FunctionReferenceDesignation</b> end and a hollow diamond at the <b>SymbolReferenceDesignationType</b> end) to a complex type <b>SymbolReferenceDesignationType</b>. This complex type is enclosed in a dashed yellow box and contains three elements: <b>ANSI-Standard</b> (type <b>Symbol-ANSI-RefDesType</b>), <b>IEC-Standard</b> (type <b>Symbol-IEC-RefDesType</b>), and <b>Other</b> (type <b>xs:string</b>). The <b>ANSI-Standard</b> and <b>IEC-Standard</b> elements are grouped together within a dashed box.</p>
type	<a href="#">SymbolReferenceDesignationType.</a>

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](#).

**4.6.1.1.3.3. Graphical Representation**

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation</a>
diagram	
type	<a href="#">SymbolFractureGraphicalRepresentationType</a> , <a href="#">FractureSymbolBodyType</a> , <a href="#">GraphicalFormat-ArrayType</a> , <a href="#">SymbolAttribute-ArrayType</a> , <a href="#">TextFormat-ArrayType</a> , <a href="#">TerminalGraphicalRepresentation-ArrayType</a> .

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](#).

## 4.6.1.1.3.3.1. Body

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body					
diagram	<pre>classDiagram     class FractureSymbolBodyType {         type FractureSymbolBodyType     }     class ShapeArray {         type ShapeArrayType     }     class Shape {         type ShapeType     }     class SVGShape {         type SVGShapeType     }     class Image {         type ImageType     }     class ShapeOrder {         type ShapeOrderType         default Back-to-Front     }     class ShapeText {         type SymbolBodyAnnotationType     }     class constraints      FractureSymbolBodyType "1" -- "1..∞" ShapeArray     ShapeArray "1" -- "1..∞" Shape     Shape "1" -- "1..∞" SVGShape     SVGShape "1" -- "1..∞" Image     ShapeArray "1" -- "1..∞" ShapeOrder     ShapeArray "1" -- "0..∞" ShapeText     constraints</pre>					
type	FractureSymbolBodyType, Shape-ArrayType, ShapeType, SVG-ShapeType, ImageType, SymbolBodyAnnotationType.					
list of enumerate values	<table><tr><th>ShapeOrder</th></tr><tr><td>Back-to-Front</td></tr><tr><td>Front-to-Back</td></tr></table>			ShapeOrder	Back-to-Front	Front-to-Back
ShapeOrder						
Back-to-Front						
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.

**4.6.1.1.3.3.1.1. Shape**

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape</a>
diagram	
type	<a href="#">ShapeType</a> , <a href="#">Vertex-ArrayType</a> , <a href="#">LineType</a> , <a href="#">ArcType</a> , <a href="#">EllipticalArcType</a> , <a href="#">Primitive-ShapeType</a> , <a href="#">GraphicalFormatType</a> .

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.

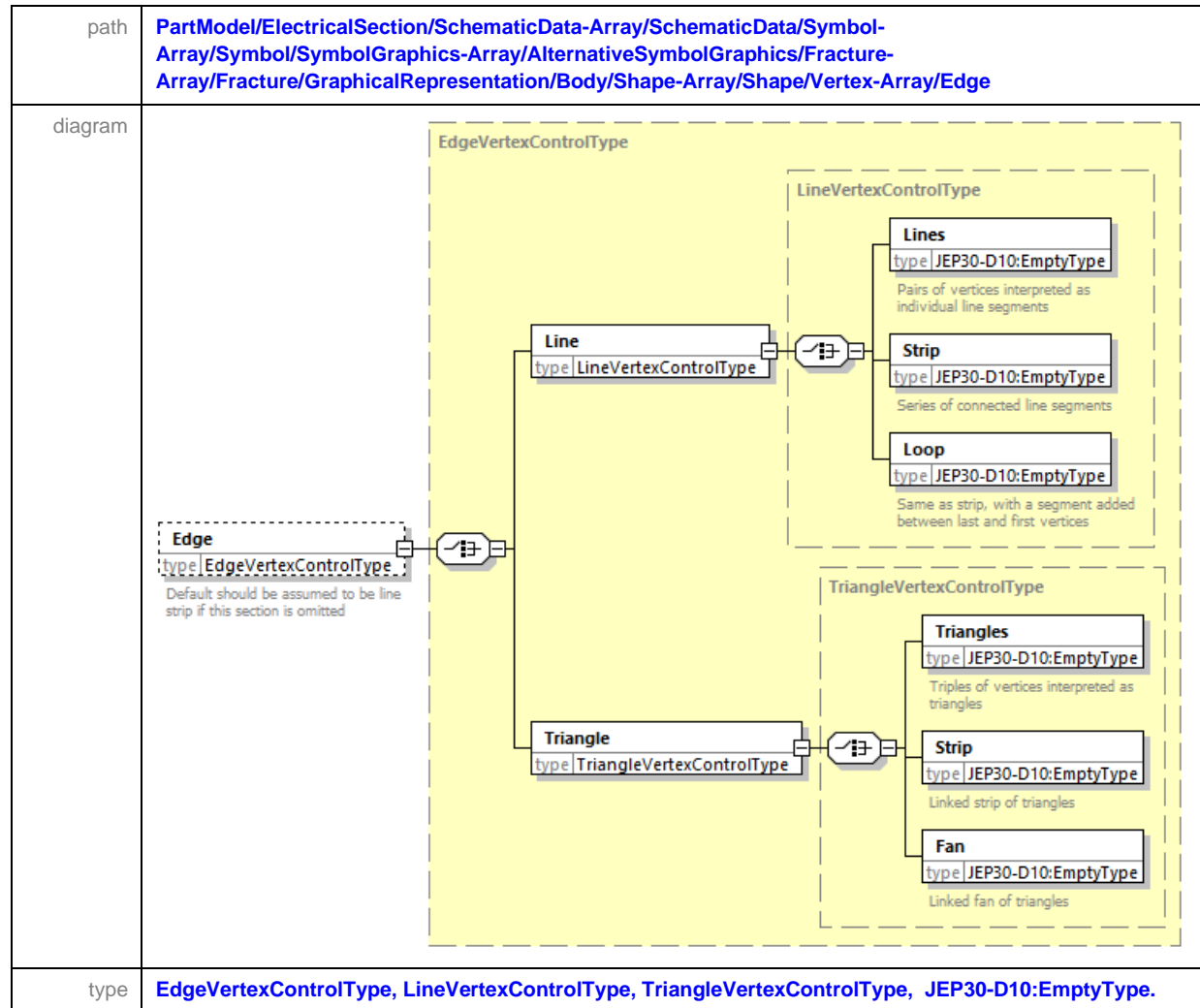
The [ShapeOrderSequence](#) can be used to define the order of shapes for a complex symbol that may consist of various shapes. It works in tandem with the [ShapeOrder](#) that is defined under [Body/Shape-Array](#).

## 4.6.1.1.3.3.1.1.1. Vertex – Array

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Vertex-Array</a>
diagram	
type	<a href="#">Vertex-ArrayType</a> , <a href="#">VertexType</a> , <a href="#">EdgeVertexControlType</a> .

## 4.6.1.1.3.3.1.1.2. Vertex

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Vertex-Array/Vertex</a>
diagram	
type	<a href="#">VertexType</a> , <a href="#">ArcVertexType</a> , <a href="#">SplineType</a> , <a href="#">JEP30-D10:PointXYType</a> .

**4.6.1.1.3.3.1.1.3. Edge**

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 2<sup>nd</sup> 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 2<sup>nd</sup> line segment, and vertices 3 and 4 makes the 3<sup>rd</sup> 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/Traingles](#) 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 2<sup>nd</sup> triangle.

#### 4.6.1.1.3.3.1.1.3. 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

#### 4.6.1.1.3.3.1.1.4. Line

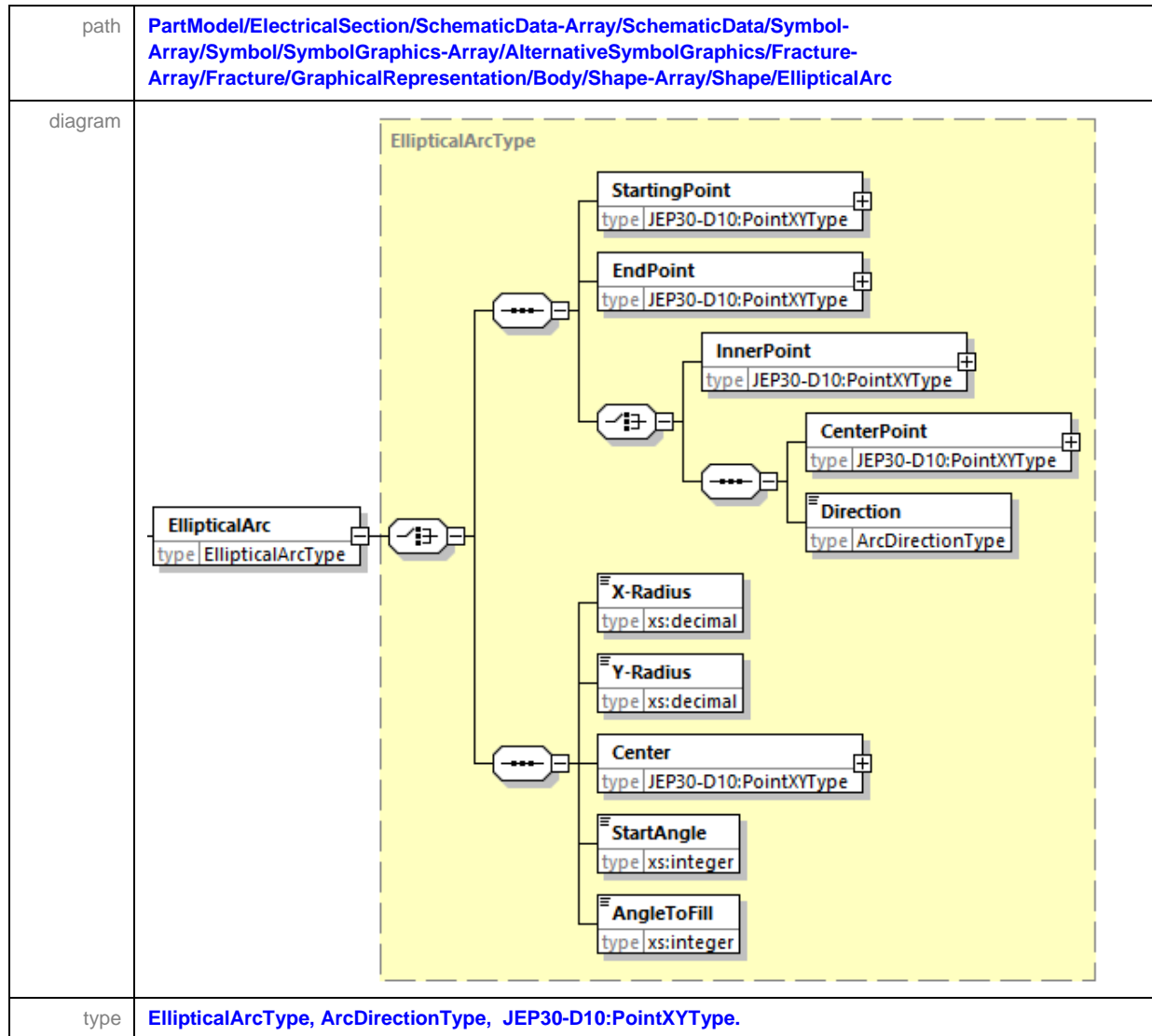
path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Shape/Line</a>
diagram	<pre> classDiagram     class Line {         type LineType     }     class StartingPoint {         type JEP30-D10:PointXYType     }     class EndPoint {         type JEP30-D10:PointXYType     }     Line "1" -- "*" StartingPoint     Line "1" -- "*" EndPoint   </pre>
type	<a href="#">LineType</a> , <a href="#">JEP30-D10:PointXYType</a> .

4.6.1.1.3.3.1.1.5. 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 {         type ArcType     }     class ArcType {         StartingPoint JEP30-D10:PointXYType         EndPoint JEP30-D10:PointXYType         InnerPoint JEP30-D10:PointXYType         CenterPoint JEP30-D10:PointXYType         Direction ArcDirectionType         Radius xs:decimal         Center JEP30-D10:PointXYType         StartAngle xs:integer         AngleToFill xs:integer     }     Arc "1" -- "*" ArcType     ArcType "1" -- "*" ArcType</pre>		
type	ArcType, ArcDirectionType, JEP30-D10:PointXYType.		
list of enumerate values	Direction		
	Clockwise	Anti-clockwise	



## 4.6.1.1.3.3.1.1.6. Elliptical Arc



4.6.1.1.3.3.1.1.7. 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	<p>The diagram illustrates the structure of the Primitive-ShapeType hierarchy. A central <b>Primitive-Shape</b> class (type Primitive-ShapeType) is connected via a composition relationship (indicated by a solid line with a filled square) to a dashed box labeled <b>Primitive-ShapeType</b>. Inside this dashed box, the <b>Primitive-Shape</b> class is further decomposed into three subclasses: <b>Rectangle</b> (type RectangleShapeType), <b>Circle</b> (type CircleShapeType), and <b>Ellipse</b> (type EllipseShapeType). Each subclass is connected to its respective shape type via a composition relationship. The <b>Rectangle</b> class is further decomposed into two subclasses: <b>DimensionX</b> (type xs:decimal) and <b>DimensionY</b> (type xs:decimal). The <b>Circle</b> class is further decomposed into one subclass: <b>Radius</b> (type xs:decimal). The <b>Ellipse</b> class is further decomposed into two subclasses: <b>X-Radius</b> (type xs:decimal) and <b>Y-Radius</b> (type xs:decimal). Additionally, the <b>Primitive-ShapeType</b> dashed box contains two other classes: <b>ShapeOrigin</b> (type ShapeOriginType) and <b>Transformation</b> (type ShapeTransformationType), both connected to the <b>Primitive-Shape</b> class via composition relationships.</p>
type	Primitive-ShapeType, RectangleShapeType, CircleShapeType, EllipseShapeType, ShapeOriginType, ShapeTransformationType.

## 4.6.1.1.3.3.1.1.8. 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 {         Horizontal HorizontalShapeOriginType         Vertical VerticalShapeOriginType         Coordinate JEP30-D10:PointXYType     }     class HorizontalShapeOriginType {         Left JEP30-D10:EmptyType         Center JEP30-D10:EmptyType         Right JEP30-D10:EmptyType     }     class VerticalShapeOriginType {         Top JEP30-D10:EmptyType         Center JEP30-D10:EmptyType         Bottom JEP30-D10:EmptyType     }     class JEP30-D10:PointXYType {         x xs:decimal         y xs:decimal     }     class ShapeOrigin {         HorizontalShapeOriginType         VerticalShapeOriginType     }     ShapeOriginType --&gt; HorizontalShapeOriginType     ShapeOriginType --&gt; VerticalShapeOriginType     ShapeOriginType --&gt; JEP30-D10:PointXYType     ShapeOrigin --&gt; HorizontalShapeOriginType     ShapeOrigin --&gt; VerticalShapeOriginType   </pre> <p><b>ShapeOrigin</b> type ShapeOriginType 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*.

## 4.6.1.1.3.3.1.1.9. 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	
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).

## 4.6.1.1.3.3.1.2. SVG-Shape

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/SVG-Shape		
diagram			
type	SVG-ShapeType, BoundingBoxType, FitStyleType, SymbolTerminalLabelOriginType, JEP30-D10:PointXYType.		
list of enumerate values	FitStyleType		
	Scale to Fit	Stretch to Fit	Crop
list of enumerate values	SymbolTerminalLabelOriginType		
	1. Northwest	2. Leftcenter	3. Southwest
	4. Uppercenter	5. Center	6. Backcenter
	7. Northeast	8. Rightcenter	9. Southeast

**4.6.1.1.3.3.1.3. Image**

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/Image		
diagram	<p>The diagram illustrates the structure of the <b>Image</b> element. The <b>Image</b> element (type <b>ImageType</b>) is shown with a cardinality of 1..∞. It contains a choice of <b>ImageData</b> and <b>BoundingBox</b>. <b>ImageData</b> is of type <b>ImageDataType</b>, which contains a choice of <b>Data</b> (type <b>xs:base64Binary</b>) and <b>GraphicalFileFormat</b> (type <b>GraphicalFileType</b>). <b>BoundingBox</b> is of type <b>BoundingBoxType</b>. The diagram also shows <b>Origin-of-Image</b> (type <b>SymbolTerminalLabelOriginType</b>, default 5 - Center) and <b>ImageLocation</b> (type <b>JEP30-D10:PointXYType</b>) as optional elements.</p>		
type	ImageType, ImageDataType, GraphicalFileType, BoundingBoxType, FitStyleType, SymbolTerminalLabelOriginType, JEP30-D10:PointXYType.		
list of enumerate values	GraphicalFileType		
	.jpg	.png	.gif
	.bmp		

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]]>

## 4.6.1.1.3.3.1.4. Shape Text

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Body/Shape-Array/ShapeText		
diagram			
type	SymbolBodyAnnotationType, TextLabelFormatType, SymbolTextOrientationType, JEP30-D10:PointXYType, SymbolTerminalLabelOriginType.		
list of enumerate values	SymbolTextOrientationType		
	0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation
	3 - 270 Degree rotation		
list of enumerate values	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*

**4.6.1.1.3.3.2. Graphical Format – Array**

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/GraphicalFormat-Array</a>
diagram	<pre> classDiagram     class GraphicalFormatArray {         type GraphicalFormat-ArrayType     }     class GraphicalFormats {         type GraphicalFormatsType     }     class DefaultGraphicalFormat {         type DefaultGraphicalFormatType     }     class GraphicalFormatsType {         ID xs:string         GraphicalFormat type GraphicalFormatType     }     GraphicalFormatArray "1" -- "1..∞" GraphicalFormats     GraphicalFormatArray "1" -- "1" DefaultGraphicalFormat     GraphicalFormatsType -- GraphicalFormat   </pre>
type	<a href="#">GraphicalFormat-ArrayType</a> , <a href="#">GraphicalFormatsType</a> , <a href="#">GraphicalFormatType</a> , <a href="#">DefaultGraphicalFormatType</a> .

[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](#).



## 4.6.1.1.3.3.2.1. Graphical Format

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/GraphicalFormat-Array/GraphicalFormats/GraphicalFormat		
diagram			
type	GraphicalFormatType, StrokeWidthType, StrokeOpacityType, StrokeLineCapType, StrokeDash-ArrayType, ColorType, ColorNameType, ColorHexType, Color-RType, Color-GType, Color-BType, FillOpacityType.		
list of enumerate values	StrokeLineCapType		
	Butt	Round	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.

4.6.1.1.3.3.2.2. Default Graphical Format

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/GraphicalFormat-Array/DefaultGraphicalFormat
diagram	<p>The diagram illustrates the structure of the <b>DefaultGraphicalFormatType</b>. It is a class that contains several attributes: <b>StrokeWidth</b> (type: <b>DefaultStrokeWidthType</b>, minIncl/maxIncl: 1, default: 1), <b>StrokeOpacity</b> (type: <b>DefaultStrokeOpacityType</b>, minIncl/maxIncl: 0, default: 100), <b>StrokeLineCap</b> (type: <b>DefaultStrokeLineCapType</b>, default: <b>Butt</b>), <b>StrokeColor</b> (type: <b>DefaultColorType</b>), <b>FillColor</b> (type: <b>DefaultColorType</b>), and <b>FillOpacity</b> (type: <b>DefaultFillOpacityType</b>, minIncl/maxIncl: 0, default: 100). The <b>StrokeColor</b> attribute is associated with the <b>DefaultColorType</b> class. The <b>DefaultColorType</b> class has attributes: <b>Name</b> (type: <b>DefaultColorNameType</b>, default: <b>Black</b>), <b>Hex</b> (pattern: <b>#[A-Fa-f0-9]{6}</b>, type: <b>DefaultColorHexType</b>, default: <b>#000000</b>), <b>R</b> (type: <b>DefaultColor-RType</b>, minIncl/maxIncl: 0, default: 0), <b>G</b> (type: <b>DefaultColor-GType</b>, minIncl/maxIncl: 0, default: 0), and <b>B</b> (type: <b>DefaultColor-BType</b>, minIncl/maxIncl: 0, default: 0). The <b>DefaultColorType</b> class also has composition relationships with <b>DefaultColorRType</b>, <b>DefaultColorGType</b>, and <b>DefaultColorBType</b>, which are represented by dashed lines with open diamonds. These three classes have attributes: <b>minIncl</b>, <b>maxIncl</b>, and <b>default</b>. The <b>DefaultColorType</b> class also has a composition relationship with <b>DefaultColorHexType</b>, which is represented by a dashed line with an open diamond. The <b>DefaultColorHexType</b> class has attributes: <b>pattern</b>, <b>type</b>, and <b>default</b>.</p>
type	<b>DefaultGraphicalFormatType</b> , <b>DefaultStrokeWidthType</b> , <b>DefaultStrokeOpacityType</b> , <b>DefaultStrokeLineCapType</b> , <b>DefaultColorType</b> , <b>DefaultColorNameType</b> , <b>DefaultColorHexType</b> , <b>DefaultColor-RType</b> , <b>DefaultColor-GType</b> , <b>DefaultColor-BType</b> , <b>DefaultFillOpacityType</b> .

If no independent *GraphicalFormat* are defined, then the *DefaultGraphicalFormat* will define the default values as follows,

- StrokeWidth* is defaulted to the value 1
- StrokeLineCap* is defaulted to value “Butt”
- StrokeColor/Name* is defaulted to value “Black”
- StrokeColor/Hex* is defaulted to value “#000000”
- StrokeColor/R*, *G* and *B* are defaulted to the value 0
- StrokeOpacity* and *FillOpacity* are defaulted to 100

#### 4.6.1.1.3.3.3. Attribute – Array

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Attribute-Array		
diagram			
type	SymbolAttribute-ArrayType, SymbolAttributeType, AttributeDetailsType, TextLabelFormatType, VisibilityType, SymbolAttributeTextVariant-for-One-of-the-Views-ArrayType.		
list of enumerate values	Visibility		
	Invisible	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.

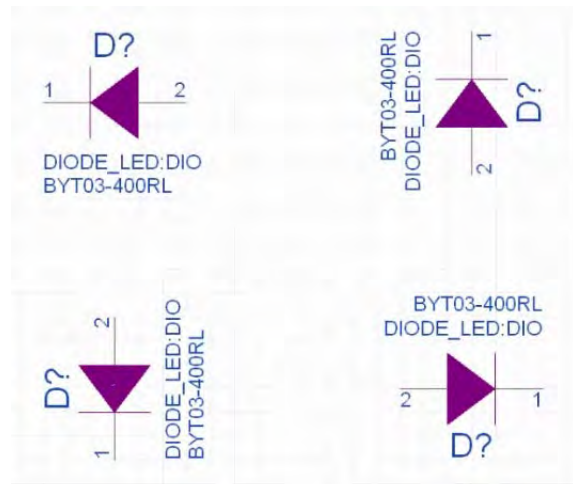
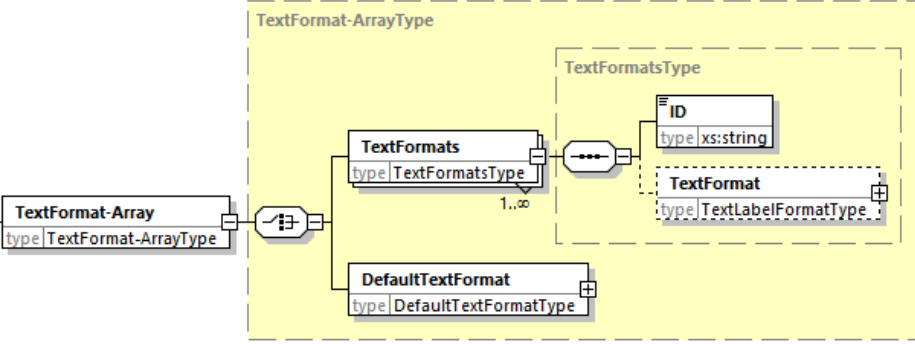


Figure 46 – Text Views for Alternative Symbol Rotation

## 4.6.1.1.3.3.1. Text Variant for One of the Views – Array

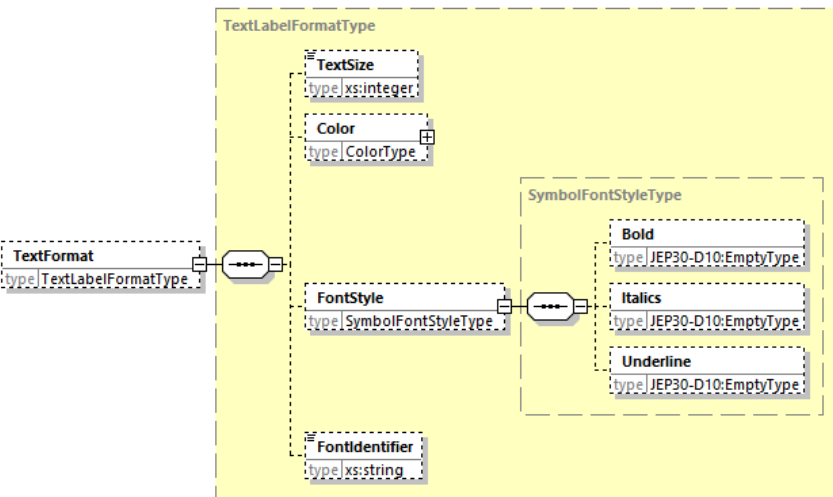
path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Attribute-Array/Attribute/TextVariant-for-One-of-the-Views-Array		
diagram			
type	SymbolAttributeTextVariant-for-One-of-the-Views-ArrayType, SymbolAttributeTextVariant-for-One-of-the-ViewsType, SymbolOrientationType, SymbolAttributeLocationType, JEP30-D10:PointXYType, SymbolTerminalLabelOriginType.		
list of enumerate values	SymbolOrientationType		
	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	SymbolTextOrientationType		
	0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation
	3 - 270 Degree rotation		
list of enumerate values	SymbolTerminalLabelOriginType		
	1. Northwest	2. Leftcenter	3. Southwest
	4. Uppercenter	5. Center	6. Backcenter
	7. Northeast	8. Rightcenter	9. Southeast

#### 4.6.1.1.3.3.4. Text Format – Array

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/TextFormat-Array</a>
diagram	 <p>The diagram illustrates the structure of the <b>TextFormat-ArrayType</b>. It is a dashed box containing three main components: <b>TextFormats</b>, <b>DefaultTextFormat</b>, and <b>TextFormatsType</b>. <b>TextFormats</b> is a box with a multiplicity of 1..∞, containing a <b>TextFormat</b> box (type <b>TextLabelFormatType</b>) and an <b>ID</b> box (type <b>xs:string</b>). <b>DefaultTextFormat</b> is a box (type <b>DefaultTextFormatType</b>). <b>TextFormatsType</b> is a dashed box containing <b>TextFormat</b> (type <b>TextLabelFormatType</b>) and <b>ID</b> (type <b>xs:string</b>). The <b>TextFormat-Array</b> box (type <b>TextFormat-ArrayType</b>) is connected to the <b>TextFormats</b> box via a connector.</p>
type	<a href="#">TextFormat-ArrayType</a> , <a href="#">TextFormatsType</a> , <a href="#">TextLabelFormatType</a> , <a href="#">DefaultTextFormatType</a> .

[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](#).

#### 4.6.1.1.3.3.4.1. Text Format

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/TextFormat-Array/TextFormats/TextFormat</a>
diagram	 <p>The diagram illustrates the structure of the <b>TextLabelFormatType</b>. It is a dashed box containing <b>TextSize</b> (type <b>xs:integer</b>), <b>Color</b> (type <b>ColorType</b>), <b>FontStyle</b> (type <b>SymbolFontStyleType</b>), and <b>FontIdentifier</b> (type <b>xs:string</b>). <b>FontStyle</b> is connected to a <b>SymbolFontStyleType</b> dashed box, which contains <b>Bold</b> (type <b>JEP30-D10:EmptyType</b>), <b>Italics</b> (type <b>JEP30-D10:EmptyType</b>), and <b>Underline</b> (type <b>JEP30-D10:EmptyType</b>). The <b>TextFormat</b> box (type <b>TextLabelFormatType</b>) is connected to the <b>TextLabelFormatType</b> dashed box via a connector.</p>
type	<a href="#">TextFormatsType</a> , <a href="#">ColorType</a> , <a href="#">SymbolFontStyleType</a> , <a href="#">JEP30-D10:EmptyType</a> .

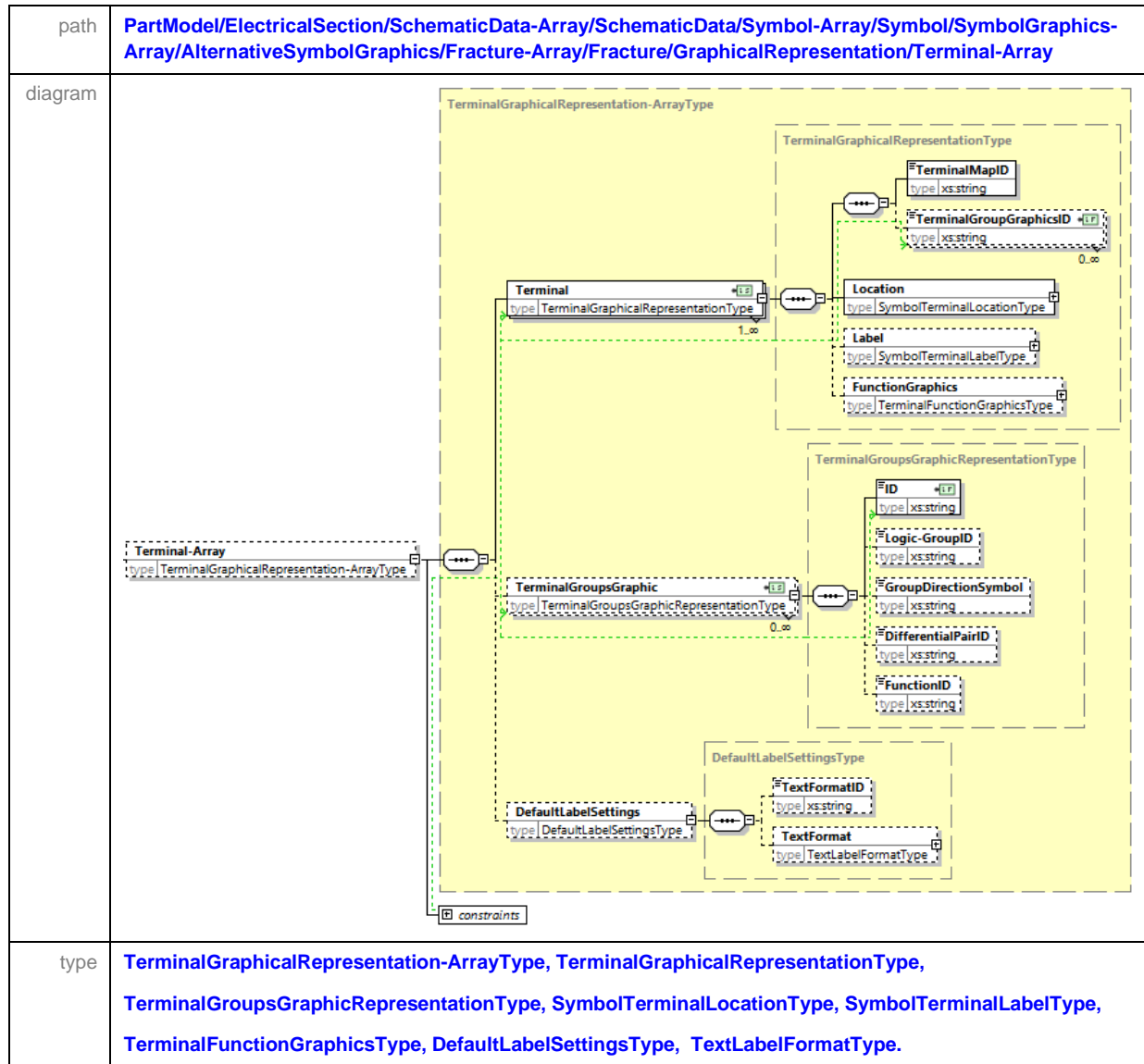
**4.6.1.1.3.3.4.2. Default Text Format**

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/TextFormat-Array/DefaultTextFormat</a>
diagram	<pre> classDiagram     class DefaultTextFormat {         type DefaultTextFormatType     }     class DefaultTextFormatType {         TextSize xs:integer 16         Color DefaultColorType         FontIdentifier xs:string Arial     }     DefaultTextFormat -- &gt; DefaultTextFormatType   </pre>
type	<a href="#">DefaultTextFormatType</a> , <a href="#">DefaultColorType</a> .

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”

## 4.6.1.1.3.3.5. Terminal – Array



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.

**4.6.1.1.3.3.5. 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 hierarical 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*.



## 4.6.1.1.3.3.5.1. Location

path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Terminal-Array/Terminal/Location		
diagram			
type	SymbolTerminalLocationType, SymbolSideType, SymbolTerminalPositionLocationType, JEP30-D10:PointXYType.		
list of enumerate values	SymbolSideType		
	Left	Right	Top
	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)

4.6.1.1.3.3.5.2. Label

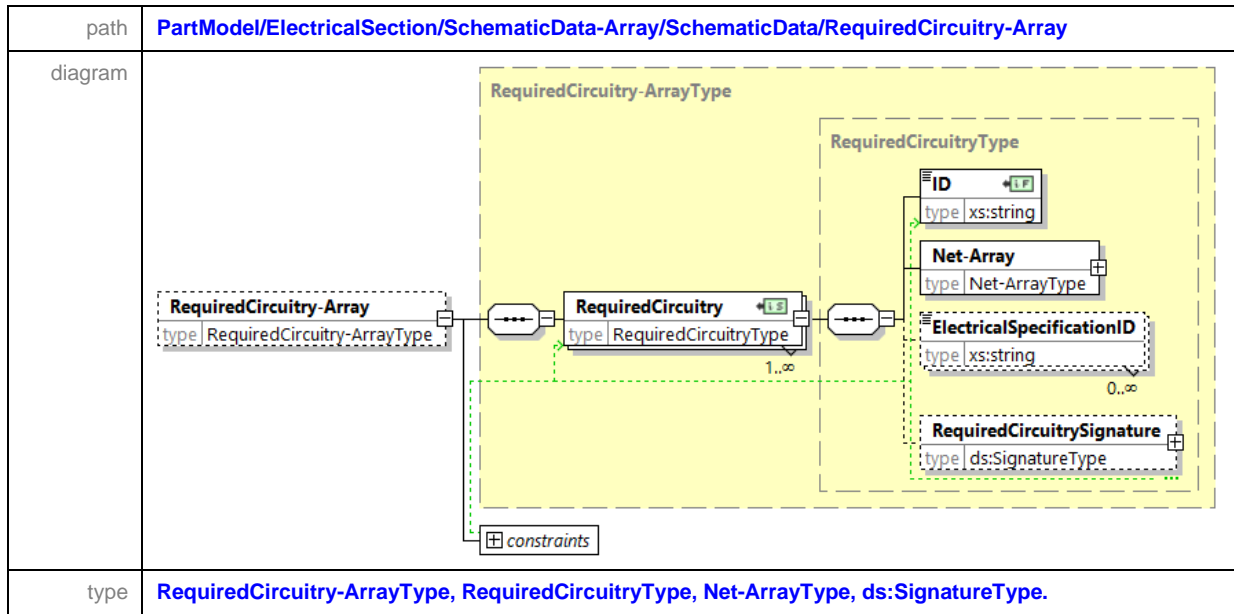
path	PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Terminal-Array/Terminal/Label		
diagram	<p>The diagram illustrates the structure of the <b>SymbolTerminalLabelType</b> class. It is a class with several attributes: <b>Value</b> (type xs:string), <b>TextFormatID</b> (type xs:string), <b>TextFormat</b> (type TextLabelFormatType), <b>Visibility</b> (type SymbolTerminalLabelVisibilityType), <b>Orientation</b> (type SymbolTextOrientationType), <b>OriginOfTextString</b> (type SymbolTerminalLabelOriginType), and <b>LabelLocation</b> (type JEP30-D10:PointXYType). A dashed box labeled <b>Label</b> points to the <b>SymbolTerminalLabelType</b> class.</p>		
type	SymbolTerminalLabelType, TextLabelFormatType, SymbolTerminalLabelVisibilityType, SymbolTextOrientationType, SymbolTerminalLabelOriginType, JEP30-D10:PointXYType.		
list of enumerate values	<b>Visibility</b>		
	Invisible	Visible	
list of enumerate values	<b>Orientation</b>		
	0 - Normal orientation	1 - 90 Degree rotation	2- 180 Degree rotation
	3 - 270 Degree rotation		
list of enumerate values	<b>OriginOfTextString</b>		
	1. Northwest	2. Leftcenter	3. Southwest
	4. Uppercenter	5. Center	6. Backcenter
	7. Northeast	8. Rightcenter	9. Southeast

**4.6.1.1.3.3.5.3. Function Graphics**

path	<a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/Symbol-Array/Symbol/SymbolGraphics-Array/AlternativeSymbolGraphics/Fracture-Array/Fracture/GraphicalRepresentation/Terminal-Array/Terminal/FunctionGraphics</a>
diagram	
type	<a href="#">TerminalFunctionGraphicsType</a> .

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](#).

**4.6.2. Required Circuitry - Array**

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.

#### 4.6.2.1. Net - Array

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/ElectricalSection/SchematicData-Array/SchematicData/RequiredCircuitry-Array/RequiredCircuitry/Net-Array</a></li> <li>2. <a href="#">PartModel/ElectricalSection/ReferenceDesign-Array/ReferenceDesign/Net-Array</a></li> </ol>
diagram	<pre> classDiagram     class NetArray {         type Net-ArrayType     }     class Net {         type NetType     }     class NetType {         Name type xs:string     }     class NetConnection {         type NetConnectionsType     }     class NetConnectionsType {         OrderablePartNumberID type xs:string         PartDetailsID type xs:string         ReferencePartDetailsID type xs:string         ManufacturerPartNumber type xs:string         ManufacturerName type xs:string         TerminalName type xs:string         TerminalNumber type xs:string         ReferenceDesignator type xs:string     }      NetArray "1" -- "1..∞" Net     NetType "1" -- "1..∞" NetConnection     NetConnectionsType "1" -- "1" OrderablePartNumberID     NetConnectionsType "1" -- "1" PartDetailsID     NetConnectionsType "1" -- "1" ReferencePartDetailsID     NetConnectionsType "1" -- "1" ManufacturerPartNumber     NetConnectionsType "1" -- "1" ManufacturerName     NetConnectionsType "1" -- "1" TerminalName     NetConnectionsType "1" -- "1" TerminalNumber     NetConnectionsType "1" -- "1" ReferenceDesignator   </pre>
type	<b>Net-ArrayType, NetType, NetConnectionType.</b>

#### 4.6.2.1 Net – Array (cont'd)

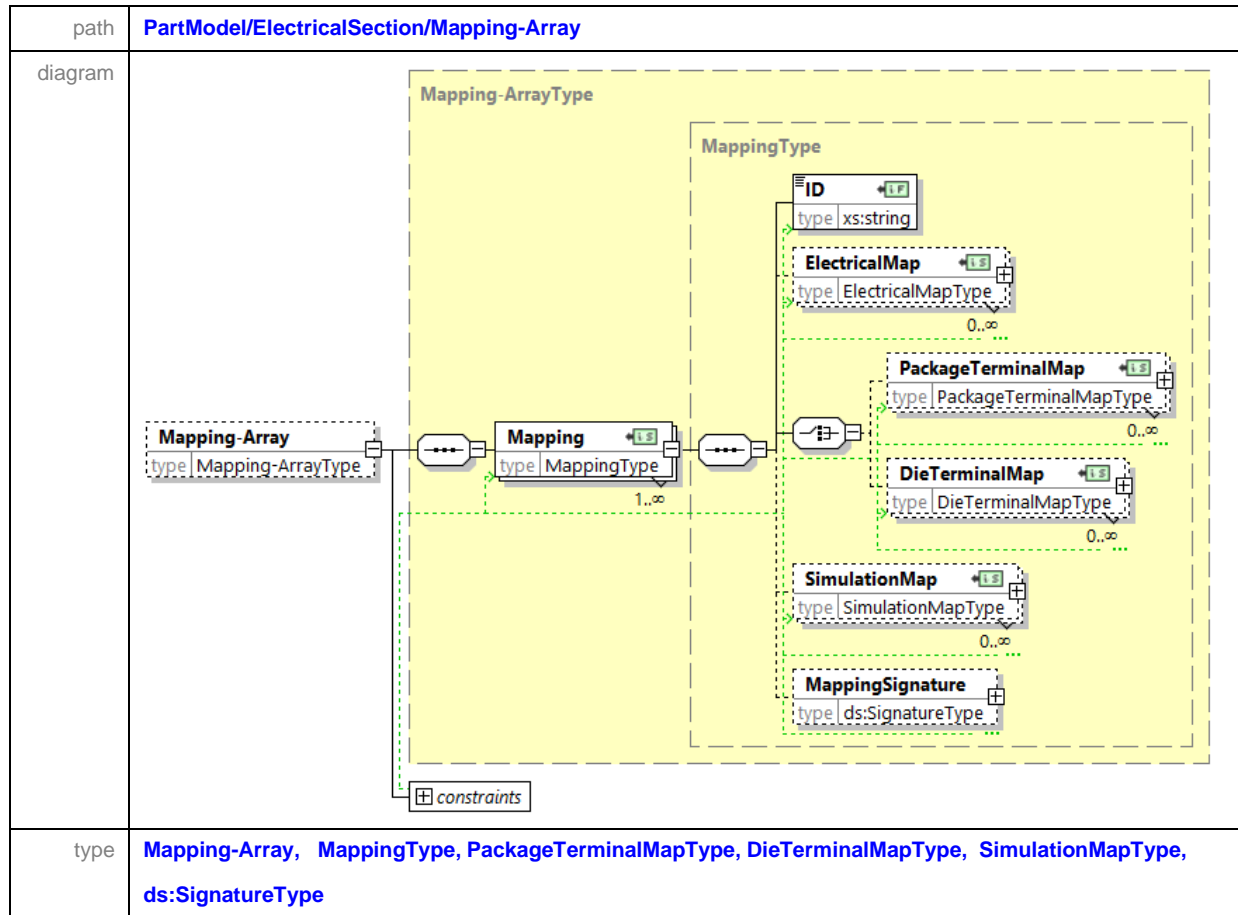
This section captures the list of *NetConnections* (otherwise known as the Netlist) between devices that makes up the required circuitry for the primary device being defined in the XML file. Since many parts can be defined within this xml structure, each part is identified via an ID or *ManufacturerPartNumber* and *ManufacturerName* combination. The details of use are defined as follows.

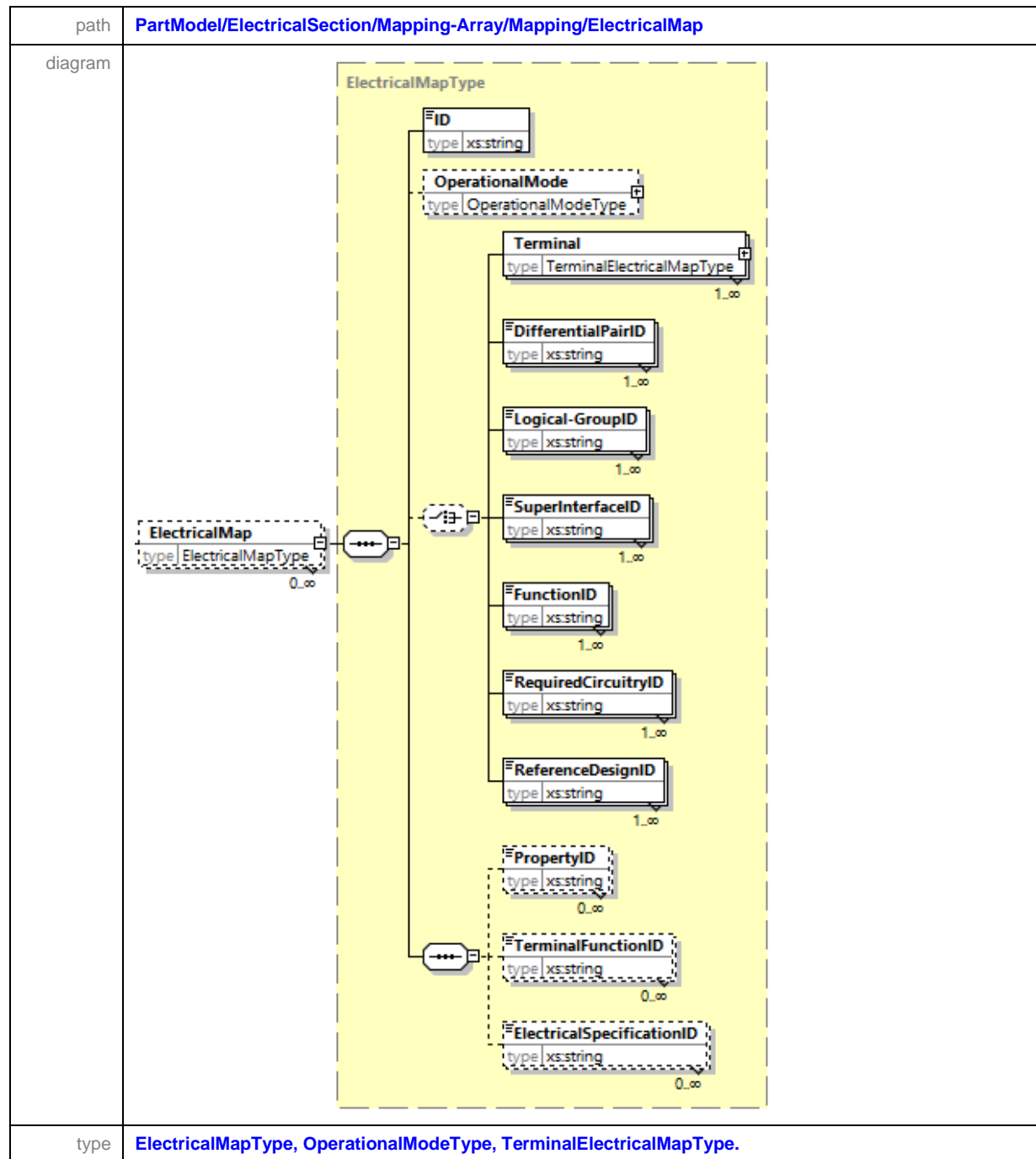
1. *OrderablePartNumberID*, or
  - a. Specify the exact Part from a specific Manufacturer. This part may be described via several Part Details ID's that are connected together under an orderable part number ID. This is a fully identifiable part.
2. *PartDetailsID*, or
  - a. Specify a limited set of 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.

#### 4.7. Mapping - Array



**4.7.1. Electrical Map – Array**

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



### 4.7.1 Electrical Map – Array (cont'd)

- 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>
    </ElectricalMap>
    <ElectricalMap>
      < Terminal>
        <TerminalMapID>Terminal Map ID 3</TerminalMapID>
      </ Terminal>
      < Terminal>
        <TerminalMapID>Terminal Map ID 6</TerminalMapID>
```

**4.7.1 Electrical Map – Array (cont'd)**

```

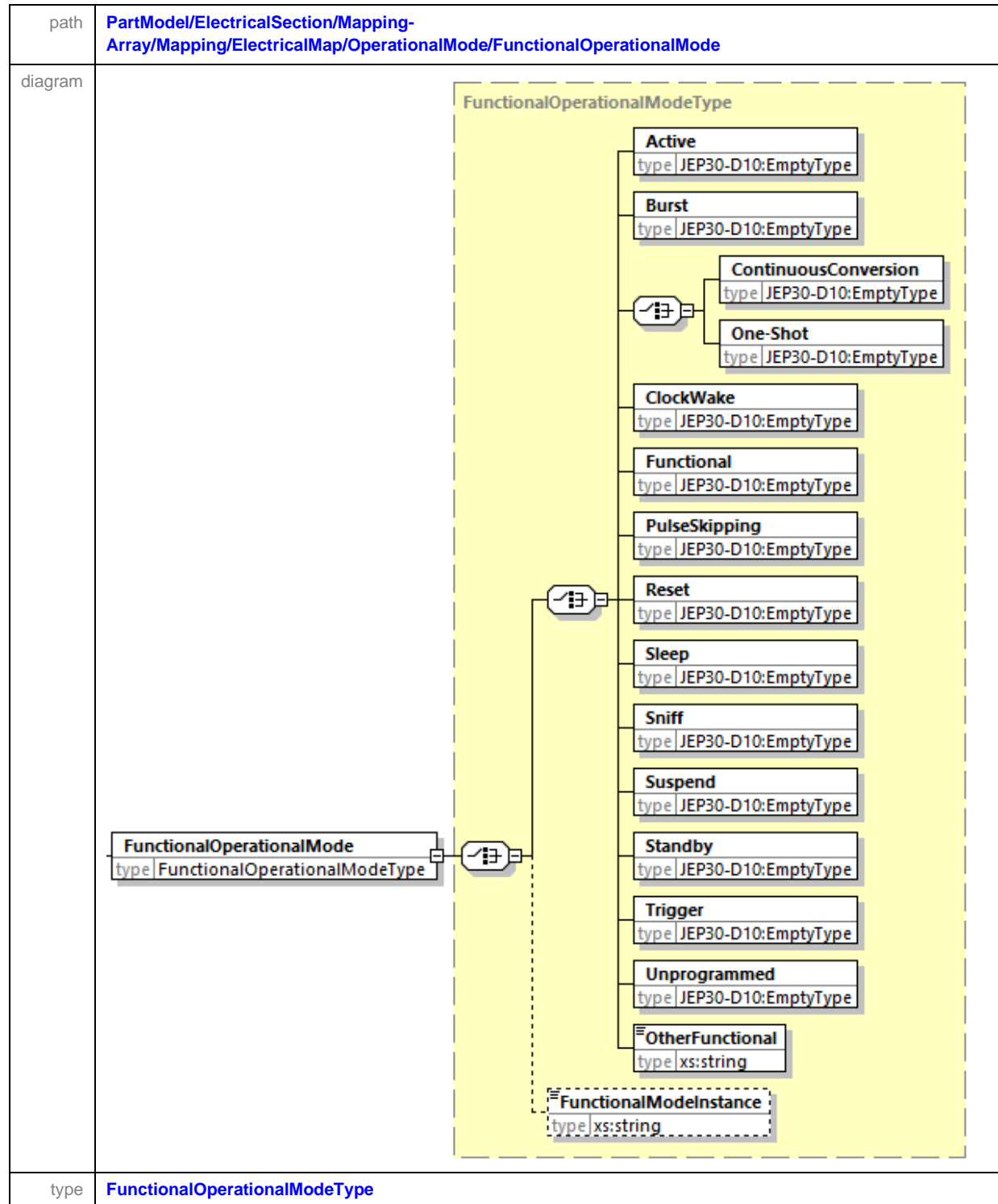
    < Terminal>
      <TerminalMapID>Terminal Map ID 8</TerminalMapID>
    </ Terminal>
    < Terminal>
      <TerminalMapID>Terminal Map ID 11</TerminalMapID>
    </ Terminal>
    <TerminalGroupDescription>Output Logic Signal</TerminalGroupDescription>
    <PropertyID>Terminal Property ID 2</PropertyID>
    <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>

```

4.7.1.1. Operational Mode

path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode
diagram	<p>The diagram illustrates the structure of the <b>OperationalModeType</b>. It is a base type that is composed of several derived types, each represented by a box with a plus sign in the top right corner. The derived types are:</p> <ul style="list-style-type: none"><li><b>FunctionalOperationalMode</b> (type: FunctionalOperationalModeType)</li><li><b>PowerOperationalMode</b> (type: PowerOperationalModeType)</li><li><b>LoadOperationalMode</b> (type: LoadOperationalModeType)</li><li><b>TestOperationalMode</b> (type: TestOperationalModeType)</li><li><b>Other</b> (type: xs:string)</li></ul> <p>The <b>OperationalModeType</b> is shown as a dashed box on the left, and the derived types are shown as solid boxes on the right. A line with a plus sign connects the <b>OperationalModeType</b> box to the derived types, indicating a composition relationship.</p>
type	OperationalModeType

## 4.7.1.1.1. Functional Operational Mode



#### 4.7.1.1.2. Power Operational Mode

path	<a href="#">PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode/PowerOperationalMode</a>
diagram	
type	<a href="#">PowerOperationalModeType</a>

[PowerDownMode](#) differs from [PowerOffSequence](#), insofar that [PowerDownMode](#) brings the device to a Sleep or Hibernation mode, whereas [PowerOffSequence](#), completely shuts the device down.

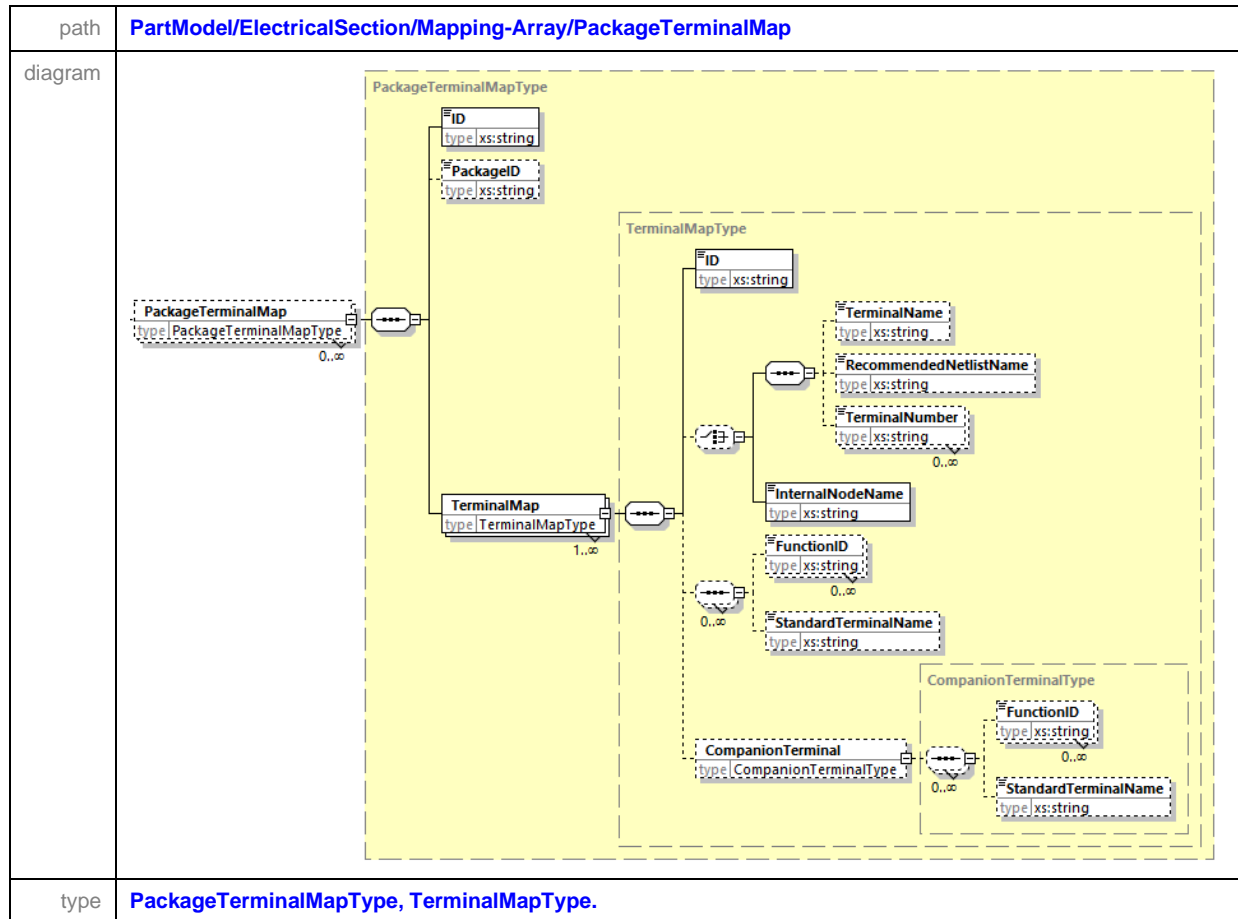
#### 4.7.1.1.3. Load Operational Mode

path	<a href="#">PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode/LoadOperationalMode</a>
diagram	
type	<a href="#">LoadOperationalModeType</a>

4.7.1.1.4. Test Operational Mode

path	PartModel/ElectricalSection/Mapping-Array/Mapping/ElectricalMap/OperationalMode/TestOperationalMode
diagram	
type	TestOperationalModeType

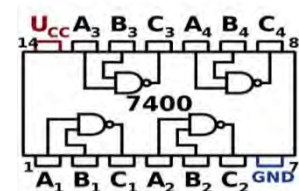
#### 4.7.2. Package Terminal Map



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 47 — Sample NAND Gate Device**

**4.7.2 Package Terminal Map (cont'd)**

```

<Mapping-Array>
  <Mapping>
    <ID>Mapping ID 1</ID>
    <PackageTerminalMap>
      <ID>Package Terminal Map ID 1</ID>
      <PackageID>Package ID 1</PackageID> <!--S0IC-->
      <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>

```



#### 4.7.2 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.

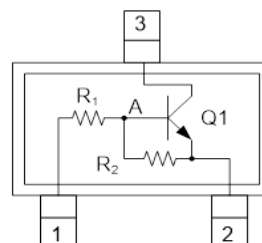
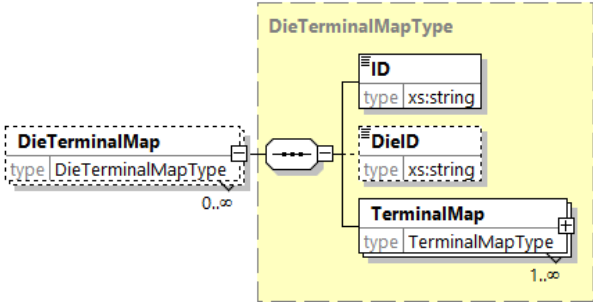
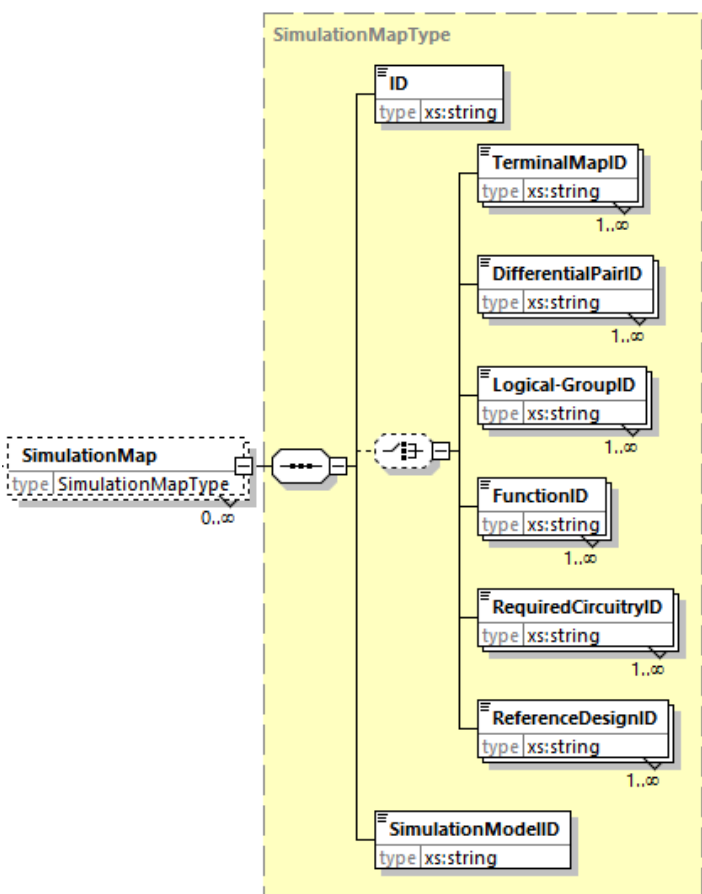


Figure 48 — Sample Transistor

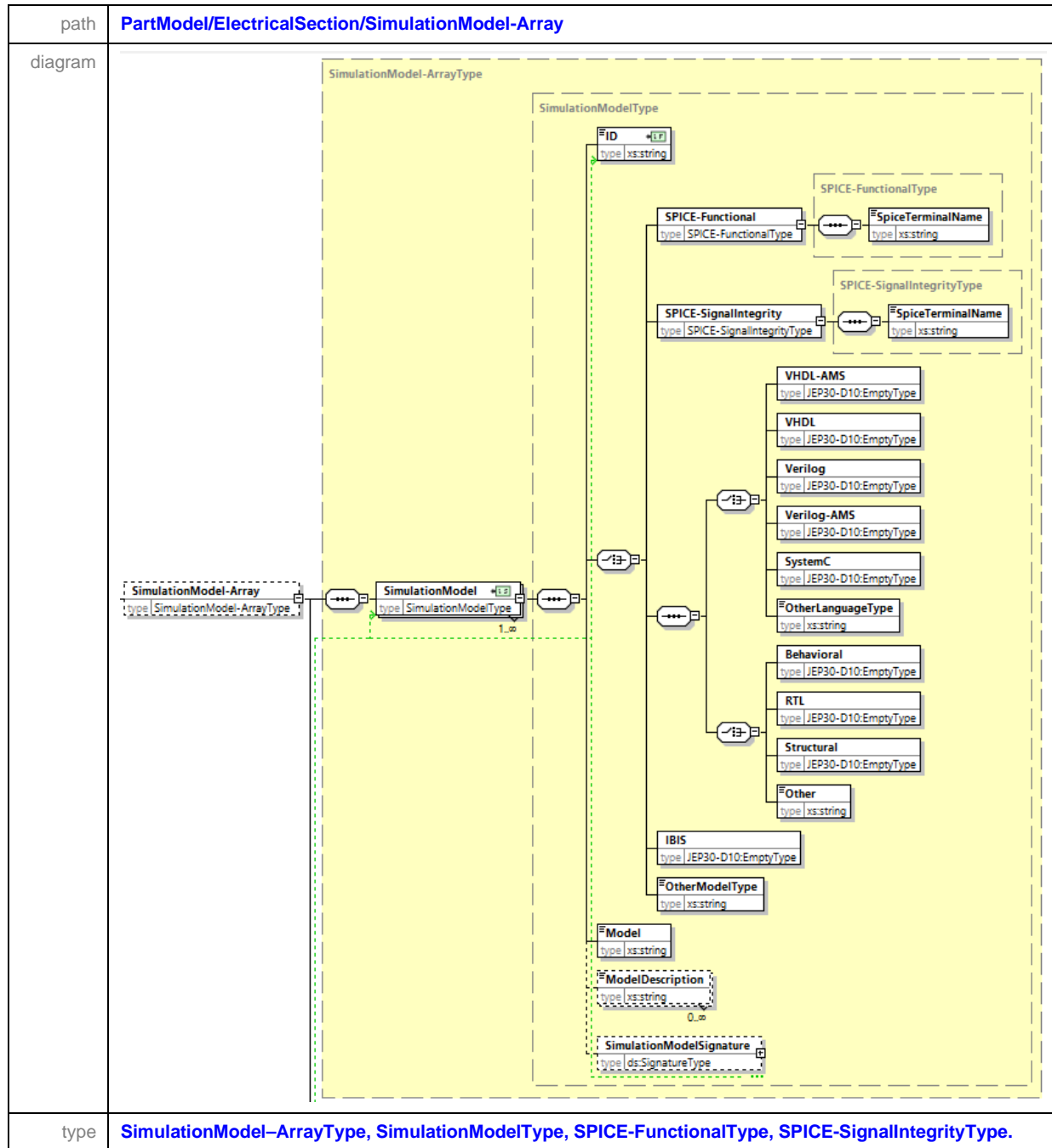
4.7.3. Die Terminal Map

path	PartModel/ElectricalSection/Mapping-Array/DieTerminalMap
diagram	 <p>The diagram illustrates the structure of the DieTerminalMap. On the left, a dashed box labeled 'DieTerminalMap' contains the text 'type DieTerminalMapType' and a cardinality of '0..∞'. This box is connected via a line with a circle and three dots to a larger dashed box labeled 'DieTerminalMapType'. Inside this box, there are three elements: 'ID' (type xs:string), 'DieID' (type xs:string), and 'TerminalMap' (type TerminalMapType, 1..∞). The 'TerminalMap' element is further connected to a box containing 'type TerminalMapType' and a cardinality of '1..∞'.</p>
type	DieTerminalMapType, TerminalMapType.

4.7.4. Simulation Map

path	PartModel/ElectricalSection/Mapping-Array/SimulationMap
diagram	 <p>The diagram illustrates the structure of the SimulationMap. On the left, a dashed box labeled 'SimulationMap' contains the text 'type SimulationMapType' and a cardinality of '0..∞'. This box is connected via a line with a circle and three dots to a larger dashed box labeled 'SimulationMapType'. Inside this box, there are several elements: 'ID' (type xs:string), 'TerminalMapID' (type xs:string, 1..∞), 'DifferentialPairID' (type xs:string, 1..∞), 'Logical-GroupID' (type xs:string, 1..∞), 'FunctionID' (type xs:string, 1..∞), 'RequiredCircuitryID' (type xs:string, 1..∞), 'ReferenceDesignID' (type xs:string, 1..∞), and 'SimulationModelID' (type xs:string). Each of these elements is connected to a box containing its type and cardinality.</p>
type	SimulationMapType.

#### 4.8. Simulation Model - Array



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.

#### 4.8 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.

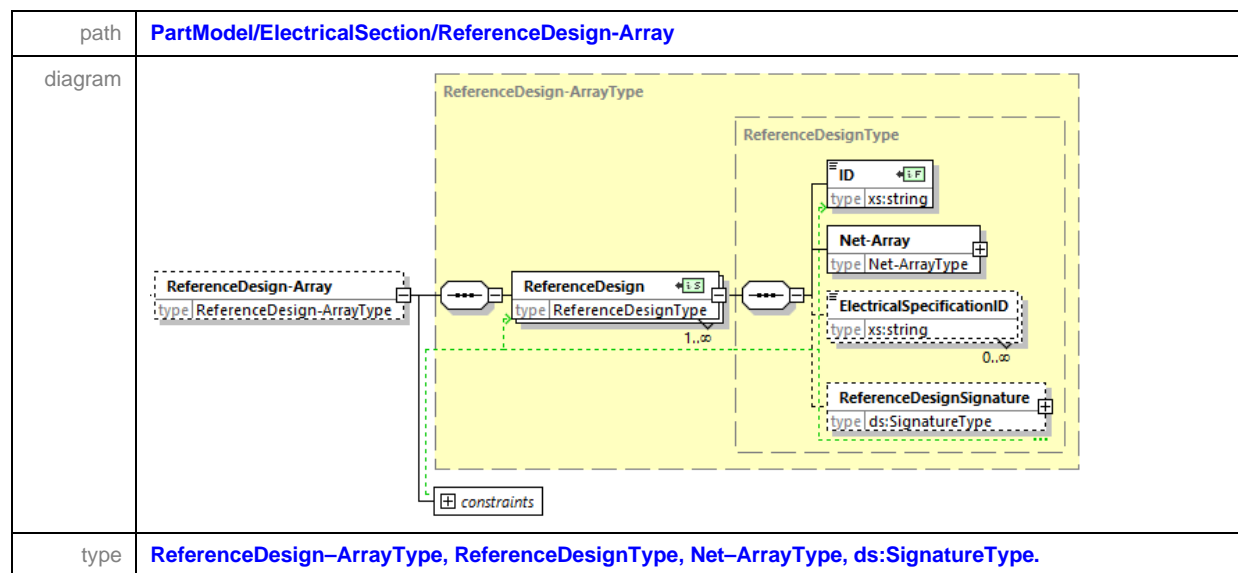
**VHDL** (VHSIC Hardware Description Language) is a hardware description language used in electronic design automation to describe digital and mixed-signal systems such as field-programmable gate arrays and integrated circuits.

**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.

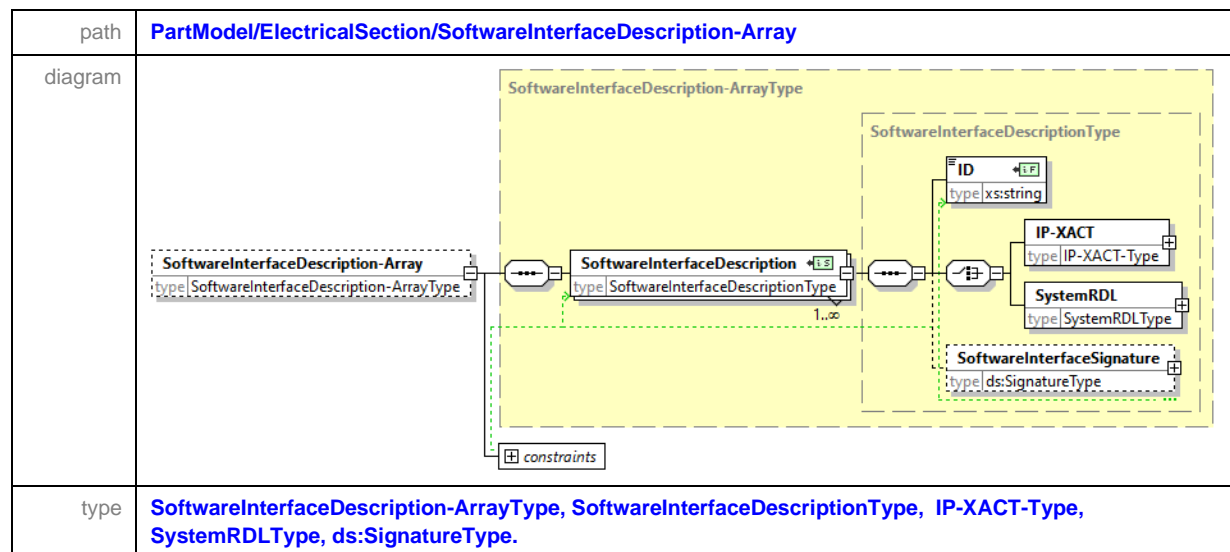
**Model** captures the URL link for the appropriate simulation model, or in the event that the simulation model is provided with the XML File, then the Model file name.

#### 4.9. Reference Design - Array



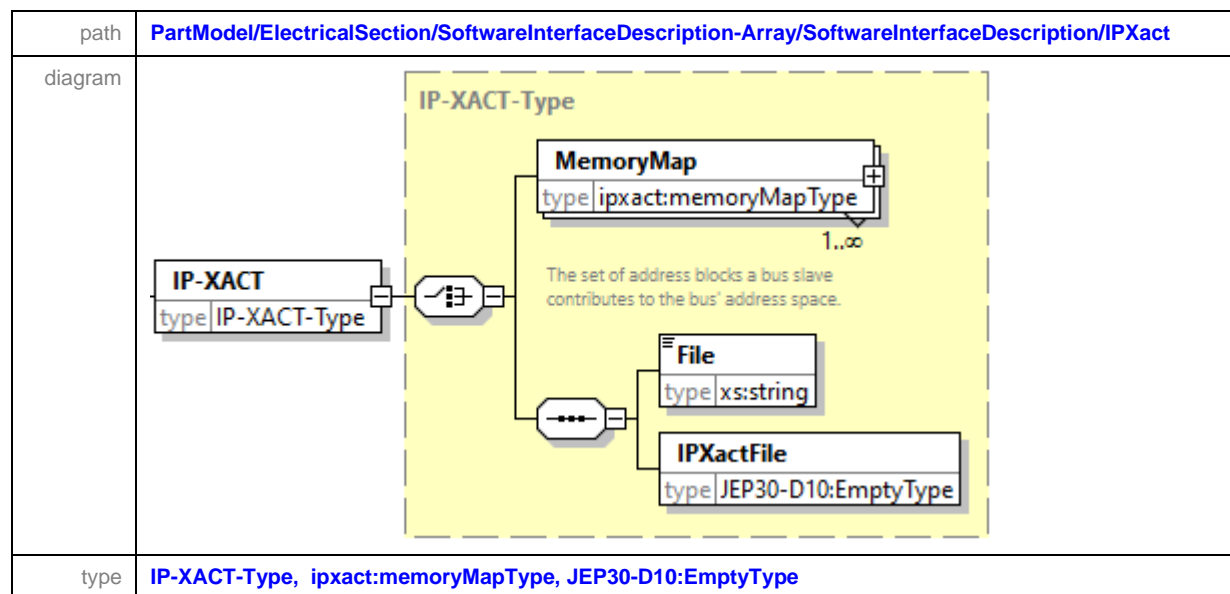
This section has the same structure as section 4.6.2 Required Circuitry - Array above, however it is to capture **ReferenceDesigns** as opposed to **RequiredCircuitry**. Component Manufacturers can provide reference designs via this section, to enable and support their users with various Product designs, in which their parts will be used.

#### 4.10. Software Interface Description - Array



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.

##### 4.10.1. IP-XACT



*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 subsection of the *IP-XACT* specification, memoryMap, describes the registers, memories and address maps used in the circuit designs.

4.10.1 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>.

4.10.2. System RDL

path	PartModel/ElectricalSection/SoftwareInterfaceDescription-Array/SoftwareInterfaceDescription/SystemRDL
diagram	<p>The diagram illustrates the SystemRDL type and its relationships. On the left, a box labeled 'SystemRDL' has a 'type' attribute set to 'SystemRDLType'. This box is connected via a directed association line to a central box labeled 'SystemRDLType'. The 'SystemRDLType' box is highlighted with a yellow dashed border and contains two internal components: a 'File' component with a 'type' attribute set to 'xs:string', and a 'SystemRDLFile' component with a 'type' attribute set to 'JEP30-D10:EmptyType'.</p>
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 use 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 5 — LaTeX Syntax Sample**

Syntax	Description
$\_ {abc}$	Subscript $_{abc}$
$\^ {abc}$	Superscript $^{abc}$
$\overline{abc}$	Overline $\overline{abc}$
$\bar{a}$	$\bar{a}$
$\hat{a}$	$\hat{a}$
$\tilde{a}$	$\tilde{a}$
$\vec{a}$	a with arrow on top
$\sqrt{a}$	Square root of a

### 5.2. Symbols

**Table 6 — LaTeX Symbols and Descriptions Sample**

Symbol	Description
$\alpha, \beta, \gamma, \dots, \omega$	$\alpha, \beta, \gamma, \dots, \omega$ (lowercase greek letters)
$\Gamma, \Delta, \Theta, \dots, \Omega$	$\Gamma, \Delta, \Theta, \dots, \Omega$ (non-latin-looking uppercase greek letters)
$\_, \^, \#, \%, \&, \backslash$	$\_, \^, \#, \%, \&, \backslash$ (symbols with special meaning)
$\infty$	$\infty$
$\circ$	$\circ$
$\pm$	$\pm$
$\leq, \geq, \neq$	$\leq, \geq, \neq$
$\backslash$	$\backslash$
$\sim$	$\sim$
$\approx$	$\approx$
$\leftarrow, \rightarrow$	$\leftarrow, \rightarrow$
NOTE < and > are special symbols in XML, so they have to be escaped in the document as &lt; and &gt;. 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 Part Model 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  $\pm 48$  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 Part Model 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>
    <Symbol> $I_{\{OHmax\}}$ </Symbol>
    <SymbolDescription>Capacitance change</SymbolDescription>
    <Rule> $I(13)+I(14)+I(15)+I(16)+I(17)+I(18)<48mA$ </Rule>
  </Parameter>
  <Parameter>
    <Symbol> $I_{\{OLmax\}}$ </Symbol>
    <Rule> $-48mA<I(13)+I(14)+I(15)+I(16)+I(17)+I(18)$ </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 Part Model 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>
    <Symbol>V_{IL}</Symbol>
    <Rule>=V(4)+0.6V</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>
    <Symbol>V_{IL}</Symbol>
    <Rule>=V(V_{SS})+0.6V</Rule>
  </Parameter>
</ParameterSet>
```

A rule such as this would be placed 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 7 — 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 8 — 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 9 — 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

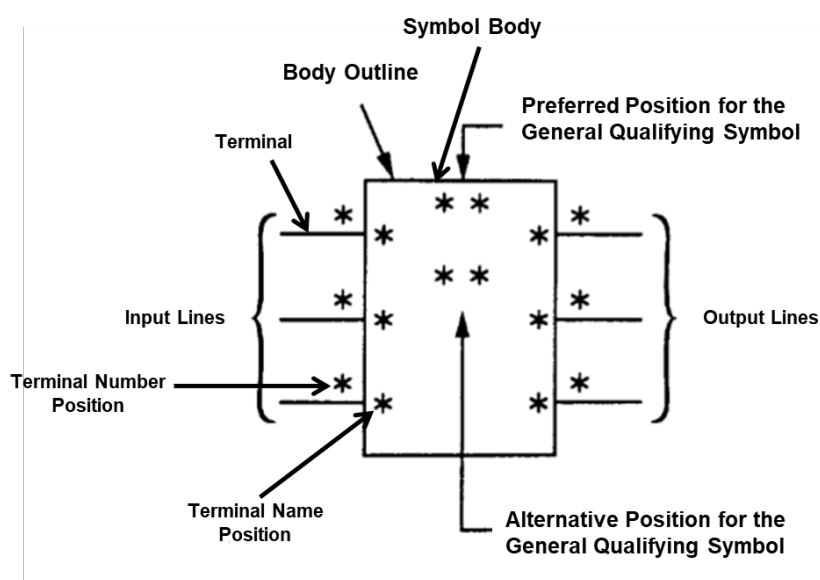
The valid unit abbreviations can be found in Table 3 — UOM Enumerated Lists.

## 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



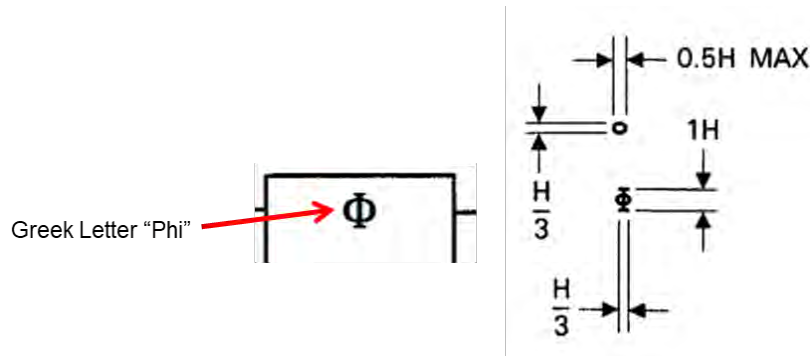
Typical symbol properties include Symbol Name and Reference Designator. Terminal Names are placed inside the symbol body. As the text grows within the body outline, the body outline should grow to prevent terminal names on the left from overlapping the terminal names on the right.

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)

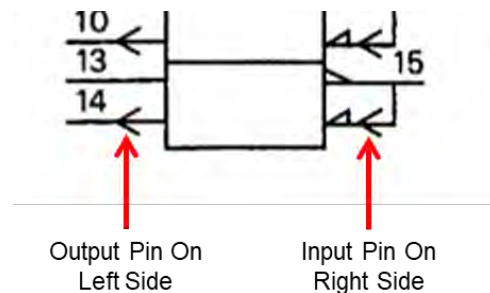
The “General Qualifying Symbol” is the Greek Letter “Phi”



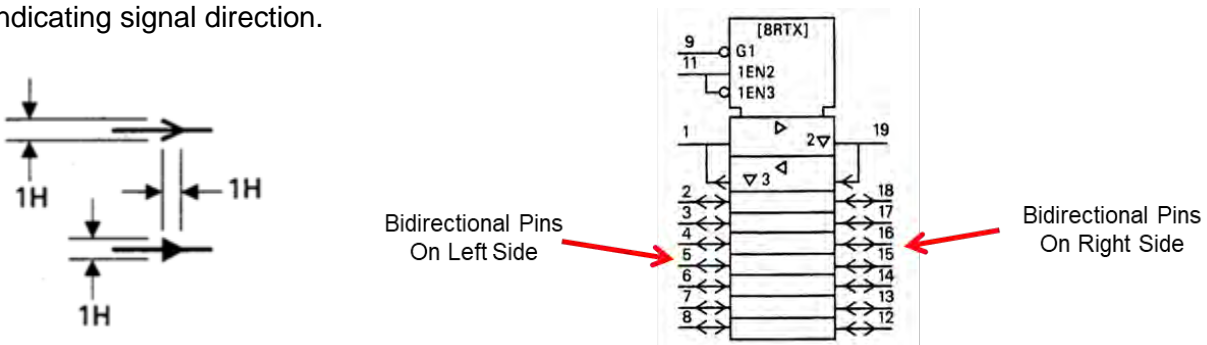
### A.1.1. Input and Output Terminals Graphics.

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.

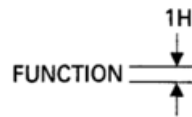
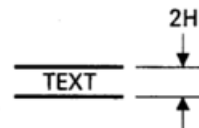
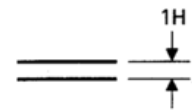
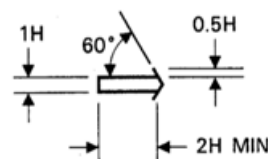


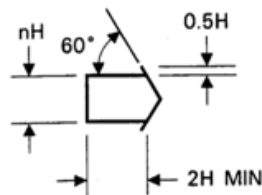
**A.1.2. Bidirectional Signal – Only One Direction at a time**

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.

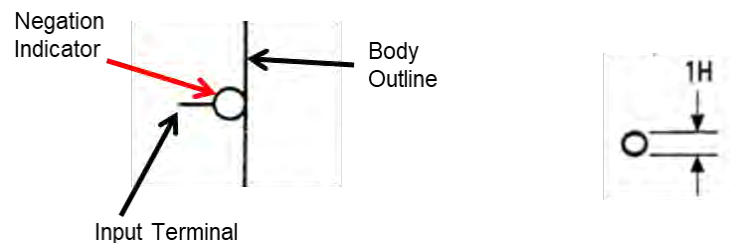
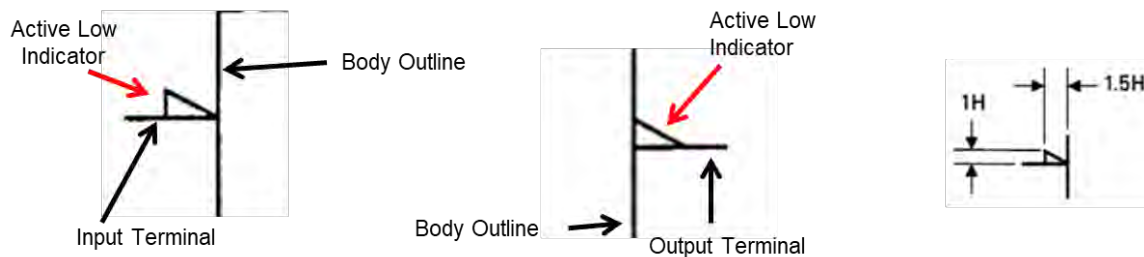
**A.1.3. Bidirectional Signal – Both directions Simultaneously**

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**

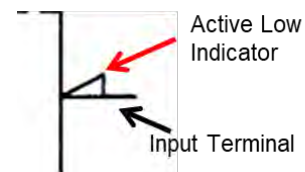
The following graphics are recommended on the symbol graphics to visibly show the function of the symbol or part.

**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.

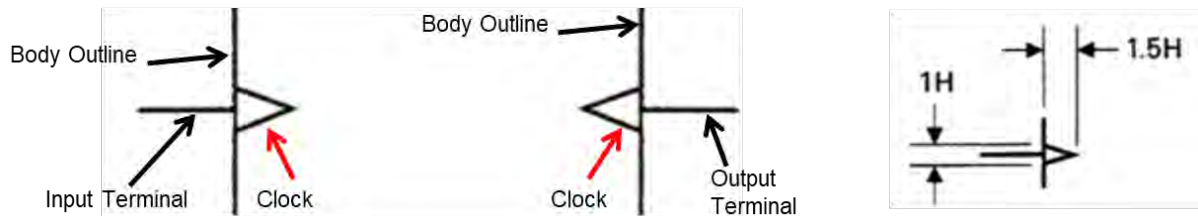


Since Active High is the assumed behavior, graphical indicators for “active high” are omitted from the symbol.

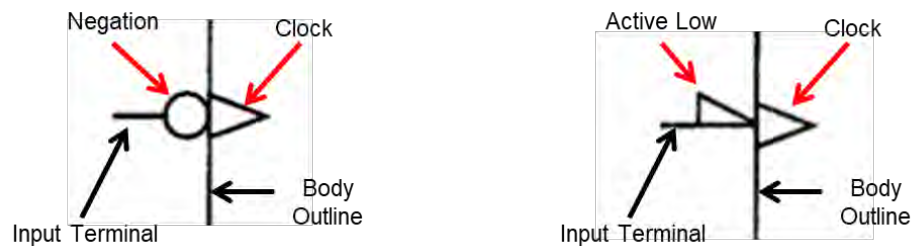


### A.2.3. Clock

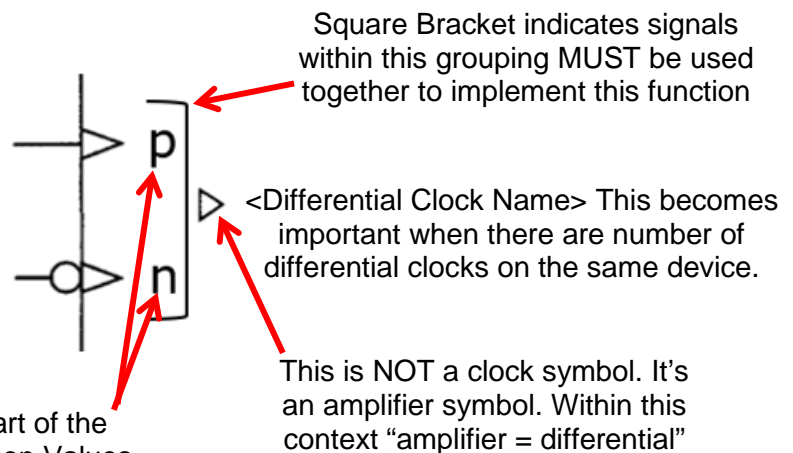
The Clock signal should be represented on the input terminal and the output terminal as shown below



When combining Clocks with polarity indicators, the symbols look like the image below.

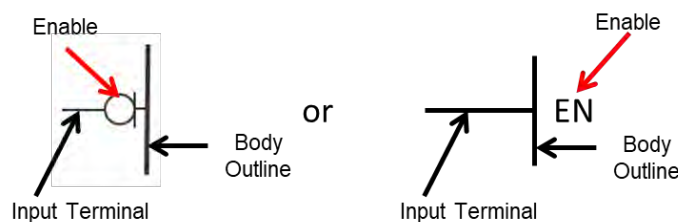


### A.2.4. Differential Clock



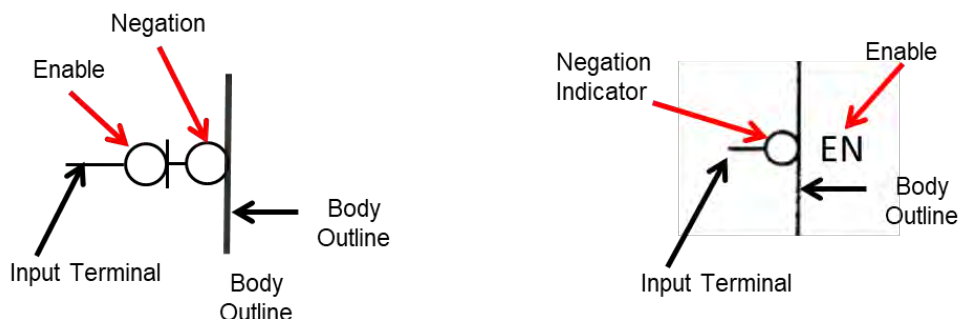
Diff Clock Name "Suffixes". The part of the clock name that is "different". Common Values include "p & n" as well as "+" & "-".

### A.2.5. Enable



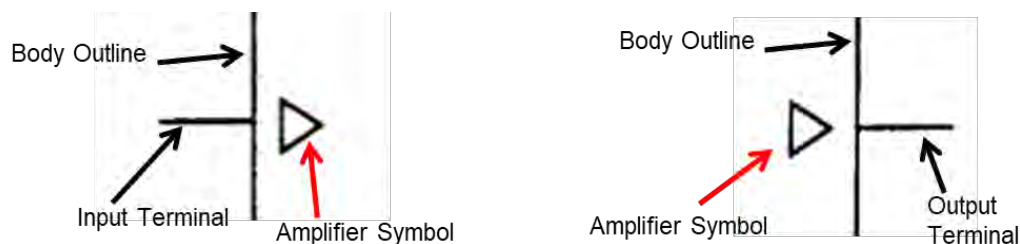
### A.2.5 Enable (cont'd)

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

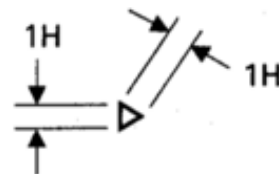


The Amplifier symbol applies only to inputs and outputs. It does not apply to Bi-directional signals. It may be used in combination with other terminal function symbols.

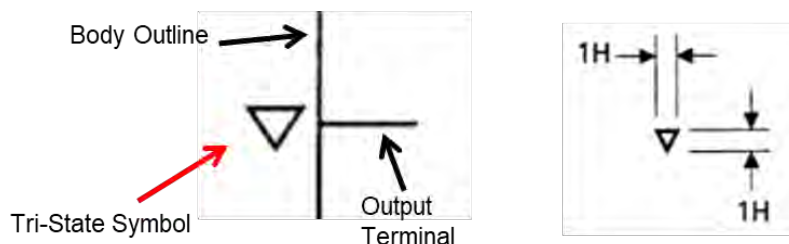
**Output Combination Restrictions:** If this symbol is used with Open Circuit Output, Open Circuit Output H-Type, Open Circuit Output L-Type, Passive-pulldown, Passive-pullup or Tri-State symbols, those symbols are placed between the amplification symbol and the edge of the Body outline.

**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.

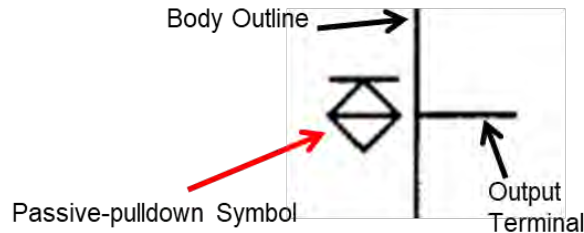


### A.2.7. Tri-State Output

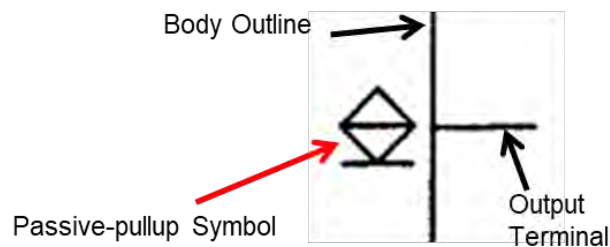


**A.2.8. Passive Pulldown**

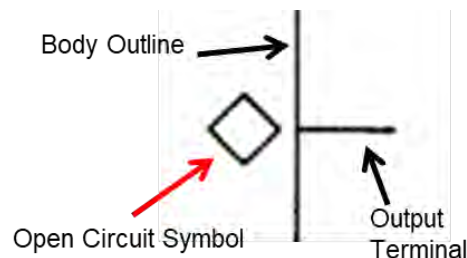
Supports simple wired logic without the need for an external component (Resistor). The passive pulldown should be represented on the output terminal as shown below.

**A.2.9. Passive Pullup**

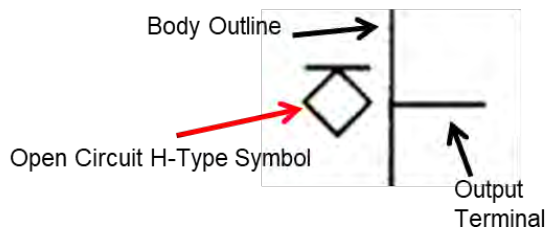
Supports simple wired logic without the need for an external component (Resistor). The passive pullup should be represented on the output terminal as shown below.

**A.2.10. Open Circuit Output**

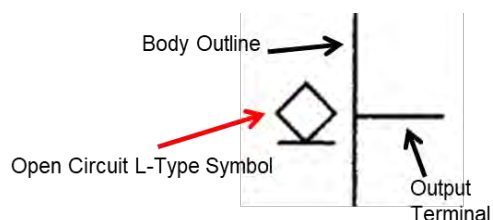
Examples of open circuit output include open-emitter, open-collector, open-source, or open-drain output. Open Circuit Output should be represented on the output terminal as shown below.

**A.2.11. Open Circuit Output (H-Type)**

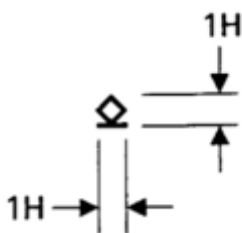
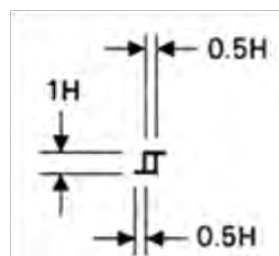
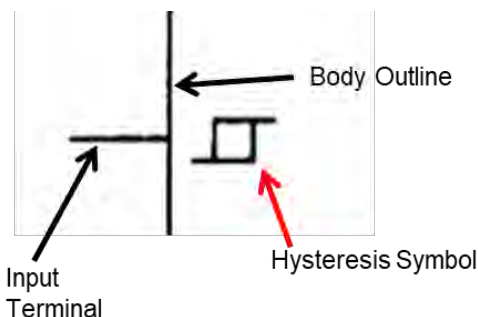
Examples of open circuit output High type include PNP Open-collector, NPN open-emitter, P-Channel open-drain, or N-Channel open Source. Open Circuit Output H-Type should be represented on the output terminal as shown below.

**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.



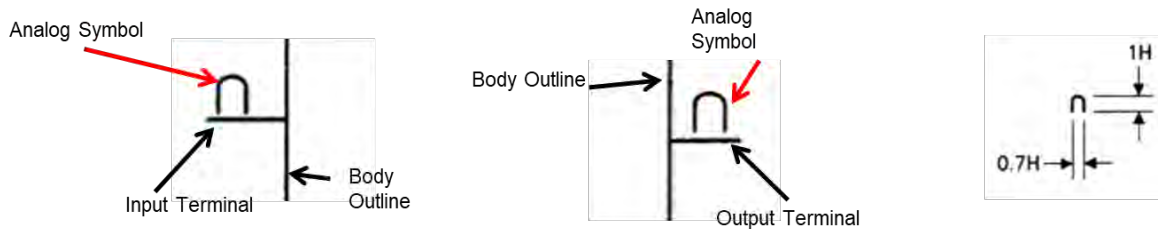
The recommended graphic sizing for an Open Circuit (L Type) symbol is shown below.

**A.2.13. Schmitt Trigger**

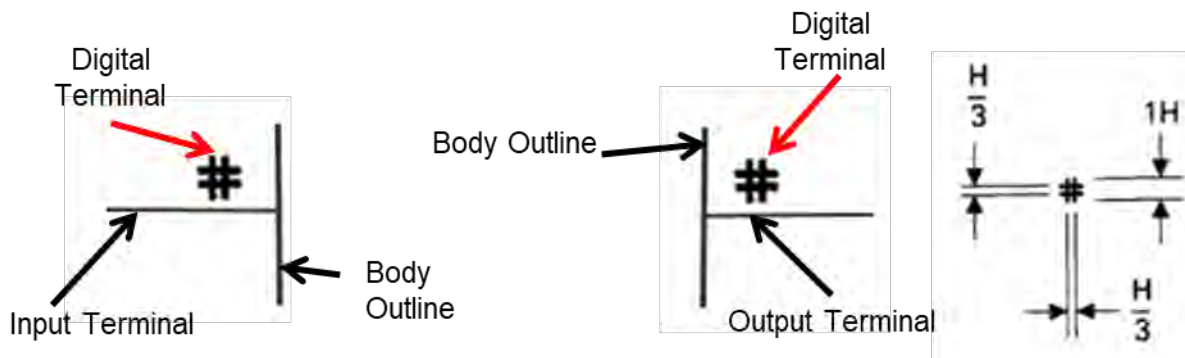
The Hysteresis symbol may be used in combinations with other terminal function symbols. When used with a dynamic input symbol (clock symbol), the hysteresis symbol shall be shown following the dynamic input symbol even though its effect occurs first. For use with the special amplification symbol, the line-grouping symbol, or dependency notation

**A.2.14. Analog Terminal**

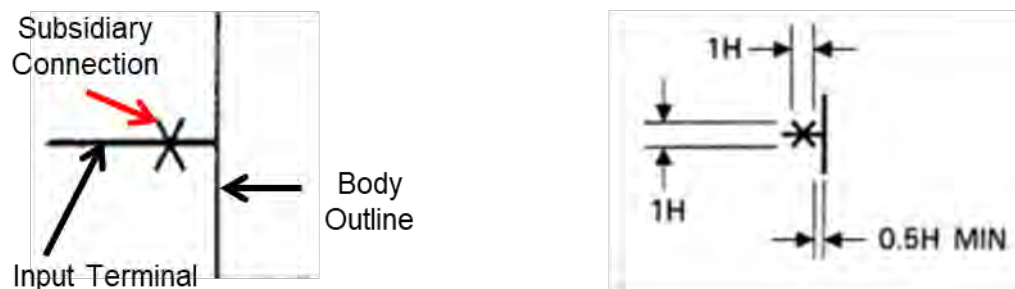
The Analog terminal symbol should be used on the input and output terminals only when it is necessary to distinguish analog signals on parts that contain both analog and digital signals.

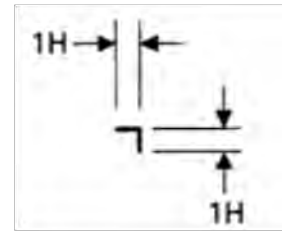
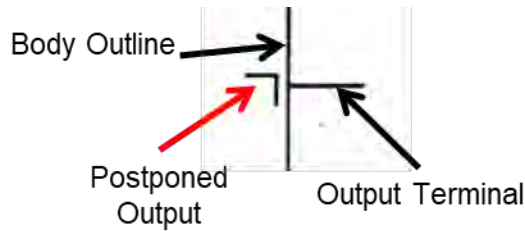
**A.2.15. Digital Terminal**

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).



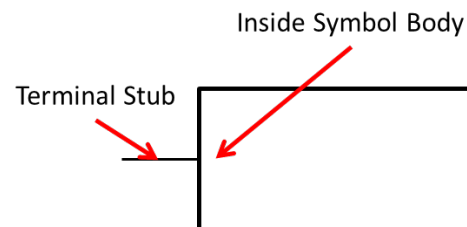
**A.2.19. Postponed Output****A.3. Recommended Rules**

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”

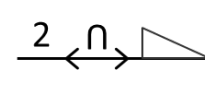
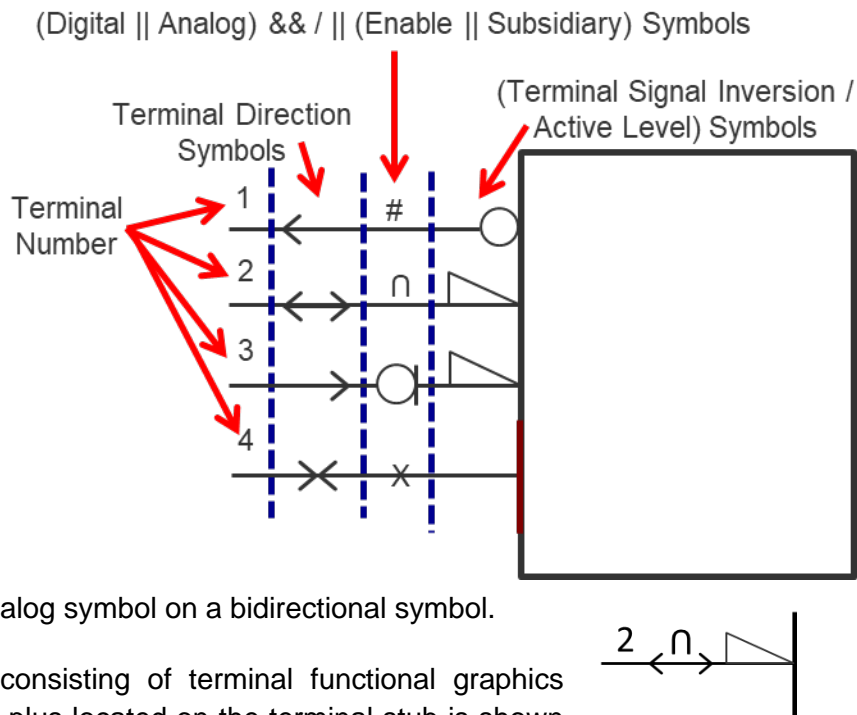
- Terminal Number
- Terminal Signal Direction
- Analog or Digital and/or Enable or Subsidiary marker
- Inversion or Level activation market



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.

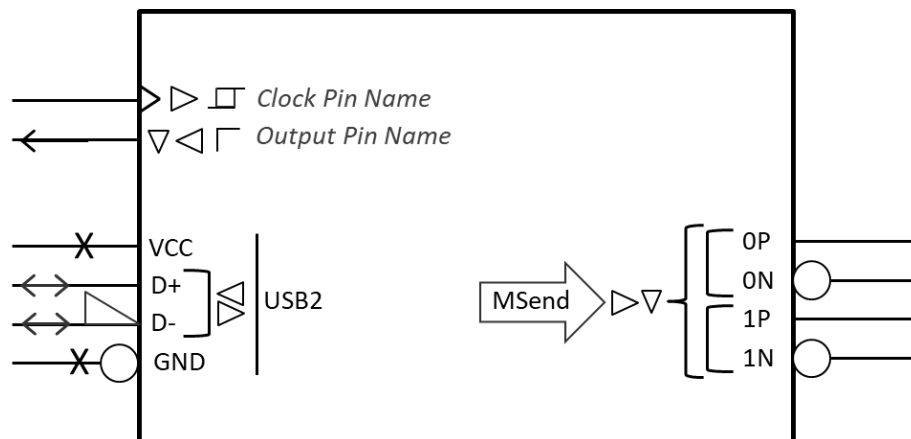


### A.3 Recommended Rules (cont'd)

#### Inside the Symbol Body

- Terminal Specific Functional Symbol may be comprised of up to 3 symbols
- Terminal Name (or Terminal Name Suffix / Bit)
- Grouping Symbol
- Group Functional Symbol may be composed of up to 2 symbols
- Group Name (Terminal Base Name), may be wrapped with "Group Direction Symbol"
- Groups symbols may be "nested" or stacked.

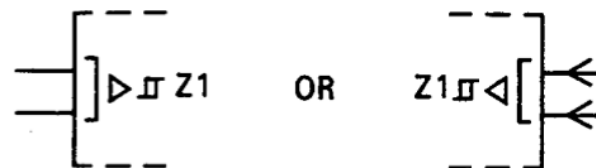
Where Possible move the function symbols to the highest common level {furthest from Terminal}. This will minimize symbol duplication.



#### 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

- Terminal Grouping
- Amplification
- Hysteresis



#### A.3.2. Terminal Grouping

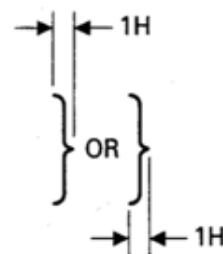
Three Forms of Terminal Grouping are defined by the ANSI IEEE spec:

- Bit Grouping
- Function Grouping, and
- Common Name Grouping.

### A.3.2.1 Bit Grouping

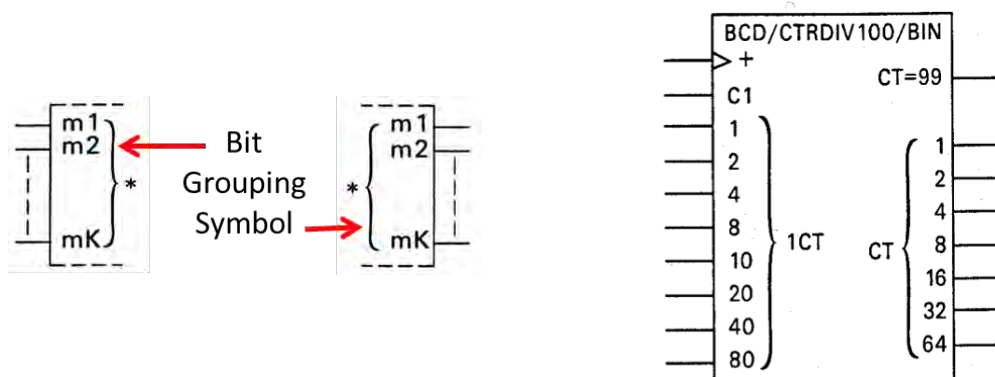
Bit-Grouping is denoted using Curly Braces “}” & “{”

- Inputs & Outputs {No Bidirectional Terminals}
- Does NOT support mixing inputs & outputs in the same group
- “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.

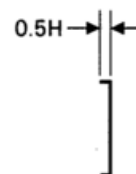
A few examples of Bit Grouping are shown below:



### A.3.2.2 Function Grouping

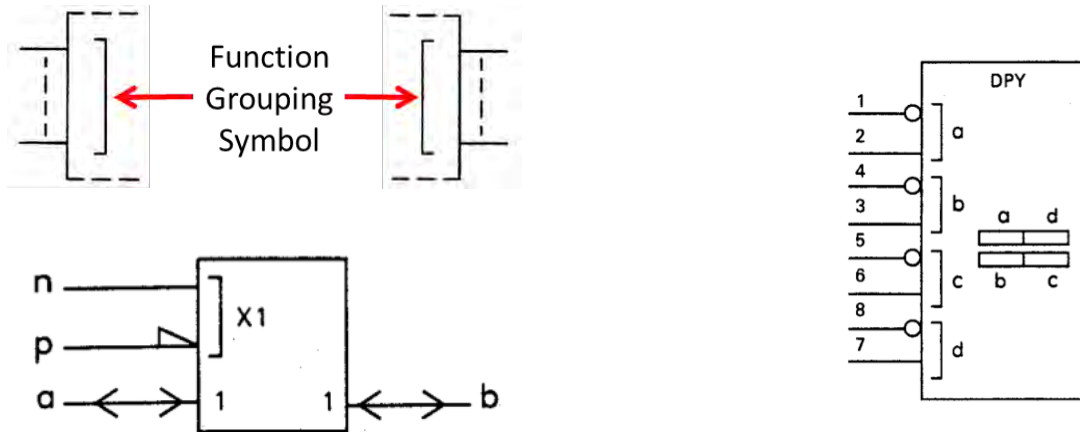
Function Grouping is denoted by using Square Bracket “]” & “[”

- Inputs & Outputs {No Bidirectional Terminals}
- Does NOT support mixing inputs & outputs in the same group
- “This symbol indicates that two or more terminals are needed to implement a single logic input.”



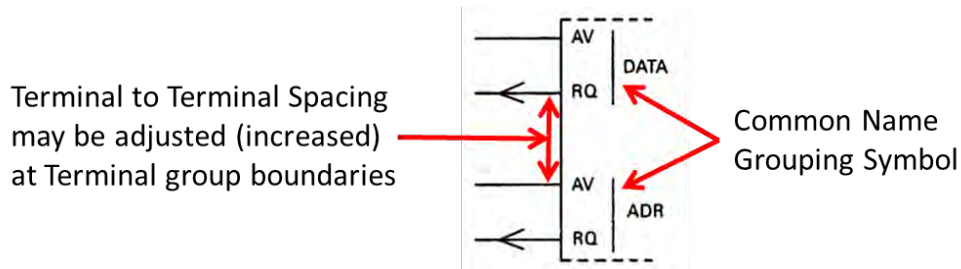
A few examples of Function Grouping are shown below:



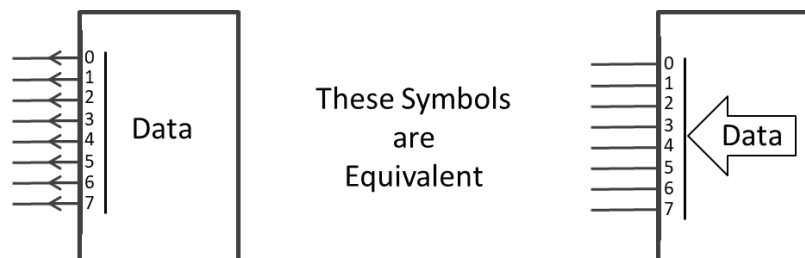
**A.3.2.2 Function Grouping (cont'd)****A.3.2.3 Common Name Grouping**

- Inputs, Outputs & Bidirectional Terminals.
- Supports Mixing inputs, outputs & Bidirectional terminals in the same group.

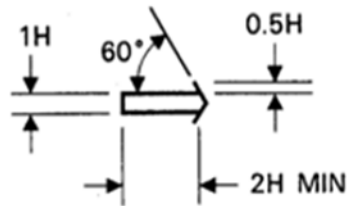
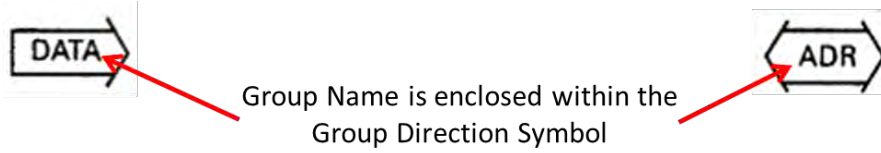
A few examples of Common Name Grouping are shown below:

**A.3.3. Group Direction Symbols**

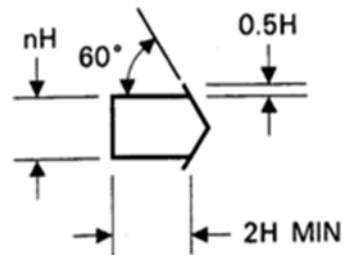
Group Direction Graphics may be used with label names to indicate signal direction. If group direction graphics are used direction symbols on individual terminals are not required.



### A.3.3 Group Direction Symbols (cont'd)



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 <i>CableAndWiring</i> . Added in section for <i>CompanionPart</i> . Classification types are added or updated to add a <i>Property-Array</i> 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 <i>CableAndWiring</i>		
Section 4.5.1.2.x Connector Classification Type: Added in a classification structure with Properties for <i>Connector</i> .		
Section 4.5.1.3.1.x Amplifier Classification: Added in 2 new sub-classifications “ <i>Audio</i> ” and “ <i>Video</i> ”.		
Section 4.5.1.3.3.x Battery Classification: Moved Material sub-classification into <i>Material</i> property.		
Section 4.5.1.3.4.x Capacitor Classification: Added <i>Silicon</i> as a new Sub-classification to Fixed Capacitor. Added <i>Class</i> , <i>Dielectric</i> , <i>Electrolyte</i> and <i>Material</i> properties throughout the Capacitor Classification structure.		
Section 4.5.1.3.5 Circuit Protection Classification: Revised this section using <i>OverCurrent</i> , <i>OverVoltage</i> and <i>OverTemperature</i> Protection classifications.		
Section 4.5.1.3.7 Diode Classification: Added <i>LED</i> , <i>PIN</i> , <i>SiliconCarbide</i> , and <i>ESD</i> as new sub-classifications		
Section 4.5.1.3.8 Filter Classification: Added <i>Ceramic</i> and <i>SAW</i> 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>PCle</i> , <i>CablingPCle</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>UFSHCI</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		

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		



---

**Standard Improvement Form****JEDEC Standard No.** \_\_\_\_\_

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 North 10th Street  
Suite 240 South  
Arlington, VA 22201-2107

Email: [angies@jedec.org](mailto: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: \_\_\_\_\_

Company: \_\_\_\_\_

Address: \_\_\_\_\_

City/State/Zip: \_\_\_\_\_

Phone: \_\_\_\_\_

E-mail: \_\_\_\_\_

Date \_\_\_\_\_

---

