

# **Logos Family Product GTPs User Guide**

(UG020007, V1.7)

(25.04.2023)

**Shenzhen Pango Microsystems Co., Ltd.**

**All Rights Reserved. Any infringement will be subject to legal action.**

---

## Revisions History

---

### Document Revisions

Versions	Date of Release	Revisions
V1.7	25.04.2023	Initial release.

## About this Manual

---

### Terms and Abbreviations

Terms and Abbreviations	Meaning
GTP	Generic Technology Primitive
ADC	Analogue to Digital Converter
HSST	High-Speed Serial Transceiver

### Related Documentation

The following documentation is related to this manual:

- 1. UG020003\_Logos Family FPGAs Arithmetic Processing Module (APM) User Guide*
- 2. UG020004\_Logos Family FPGAs Clock Resources (Clock) User Guide*
- 3. UG020009\_Logos Family FPGAs Analogue-to-Digital Converter (ADC) Module User Guide*
- 4. UG0200013\_Logos Family FPGAs High-Speed Serial Transceiver (HSST) User Guide*

## Table of Contents

---

<b>Revisions History .....</b>	<b>1</b>
<b>About this Manual.....</b>	<b>2</b>
<b>Table of Contents .....</b>	<b>3</b>
<b>Tables .....</b>	<b>19</b>
<b>Figures .....</b>	<b>27</b>
<b>Chapter 1 GTP Classification .....</b>	<b>31</b>
<b>Chapter 2 Usage Instructions for Trigger GTPs.....</b>	<b>34</b>
2.1 General Introduction.....	34
2.2 Usage Instructions for GTP_DFF.....	34
2.2.1 Supported Devices .....	34
2.2.2 Description of Functionality.....	34
2.2.3 Port Description .....	34
2.2.4 Paramater Description.....	35
2.2.5 Instantiation template .....	35
2.3 Usage Instructions for GTP_DFF_C .....	35
2.3.1 Supported Devices .....	35
2.3.2 Description of Functionality.....	35
2.3.3 Port Description .....	36
2.3.4 Paramater Description.....	36
2.3.5 Instantiation template .....	36
2.4 Usage Instructions for GTP_DFF_E .....	37
2.4.1 Supported Devices .....	37
2.4.2 Description of Functionality.....	37
2.4.3 Port Description .....	37
2.4.4 Paramater Description.....	38
2.4.5 Instantiation template .....	38
2.5 Usage Instructions for GTP_DFF_CE.....	38
2.5.1 Supported Devices .....	38
2.5.2 Description of Functionality.....	38
2.5.3 Port Description .....	39
2.5.4 Instantiation template .....	39
2.6 Usage Instructions for GTP_DFF_P .....	39
2.6.1 Supported Devices .....	39
2.6.2 Description of Functionality.....	40
2.6.3 Port Description .....	40
2.6.4 Paramater Description.....	40

2.6.5 Instantiation template .....	40
2.7 Usage Instructions for GTP_DFF_PE .....	41
2.7.1 Supported Devices .....	41
2.7.2 Description of Functionality.....	41
2.7.3 Port Description .....	42
2.7.4 Paramater Description.....	42
2.7.5 Instantiation template .....	42
2.8 Usage Instructions for GTP_DFF_R .....	43
2.8.1 Supported Devices .....	43
2.8.2 Description of Functionality.....	43
2.8.3 Port Description .....	43
2.8.4 Paramater Description.....	43
2.8.5 Instantiation template .....	44
2.9 Usage Instructions for GTP_DFF_RE .....	44
2.9.1 Supported Devices .....	44
2.9.2 Description of Functionality.....	44
2.9.3 Port Description .....	45
2.9.4 Paramater Description.....	45
2.9.5 Instantiation template .....	45
2.10 Usage Instructions for GTP_DFF_S .....	46
2.10.1 Supported Devices .....	46
2.10.2 Description of Functionality.....	46
2.10.3 Port Description .....	46
2.10.4 Paramater Description.....	46
2.10.5 Instantiation template .....	47
2.11 Usage Instructions for GTP_DFF_SE .....	47
2.11.1 Supported Devices.....	47
2.11.2 Description of Functionality.....	47
2.11.3 Port Description.....	48
2.11.4 Paramater Description.....	48
2.11.5 Instantiation template .....	48
<b>Chapter 3 Usage Instructions for Latch GTPs.....</b>	<b>49</b>
3.1 General Introduction.....	49
3.2 Usage Instructions for GTP_DLATCH .....	49
3.2.1 Supported Devices .....	49
3.2.2 Description of Functionality.....	49
3.2.3 Port Description .....	49
3.2.4 Paramater Description.....	50
3.2.5 Instantiation template .....	50

3.3 Usage Instructions for GTP_DLATCH_E.....	50
3.3.1 Supported Devices .....	50
3.3.2 Description of Functionality.....	50
3.3.3 Port Description .....	51
3.3.4 Paramater Description.....	51
3.3.5 Instantiation template .....	51
3.4 Usage Instructions for GTP_DLATCH_C.....	52
3.4.1 Supported Devices .....	52
3.4.2 Description of Functionality.....	52
3.4.3 Port Description .....	52
3.4.4 Paramater Description.....	53
3.4.5 Instantiation template .....	53
3.5 Usage Instructions for GTP_DLATCH_CE .....	53
3.5.1 Supported Devices .....	53
3.5.2 Description of Functionality.....	53
3.5.3 Port Description .....	54
3.5.4 Paramater Description.....	54
3.5.5 Instantiation template .....	54
3.6 Usage Instructions for GTP_DLATCH_P .....	55
3.6.1 Supported Devices .....	55
3.6.2 Description of Functionality.....	55
3.6.3 Port Description .....	55
3.6.4 Paramater Description.....	56
3.6.5 Instantiation template .....	56
3.7 Usage Instructions for GTP_DLATCH_PE.....	56
3.7.1 Supported Devices .....	56
3.7.2 Description of Functionality.....	56
3.7.3 Port Description .....	57
3.7.4 Paramater Description.....	57
3.7.5 Instantiation template .....	57
<b>Chapter 4 Usage Instructions for LUT-related GTP .....</b>	<b>59</b>
4.1 Usage Instructions for GTP_ONE.....	59
4.1.1 Supported Devices .....	59
4.1.2 Description of Functionality.....	59
4.1.3 Port Description .....	59
4.1.4 Instantiation template .....	59
4.2 Usage Instructions for GTP_ZERO.....	60
4.2.1 Supported Devices .....	60
4.2.2 Description of Functionality.....	60

4.2.3 Port Description .....	60
4.2.4 Instantiation template .....	60
4.3 Usage Instructions for GTP_BUF .....	60
4.3.1 Supported Devices .....	60
4.3.2 Description of Functionality.....	61
4.3.3 Port Description .....	61
4.3.4 Instantiation template .....	61
4.3.5 Detailed Functional Description.....	61
4.4 Usage Instructions for GTP_INV .....	61
4.4.1 Supported Devices .....	61
4.4.2 Description of Functionality.....	61
4.4.3 Port Description .....	62
4.4.4 Instantiation template .....	62
4.5 Usage Instructions for GTP_LUT1 .....	62
4.5.1 Supported Devices .....	62
4.5.2 Description of Functionality.....	62
4.5.3 Port Description .....	63
4.5.4 Paramater Description.....	63
4.5.5 Instantiation template .....	63
4.6 Usage Instructions for GTP_LUT2 .....	64
4.6.1 Supported Devices .....	64
4.6.2 Description of Functionality.....	64
4.6.3 Port Description .....	64
4.6.4 Paramater Description.....	65
4.6.5 Instantiation template .....	65
4.7 Usage Instructions for GTP_LUT3 .....	65
4.7.1 Supported Devices .....	65
4.7.2 Description of Functionality.....	65
4.7.3 Port Description .....	66
4.7.4 Paramater Description.....	66
4.7.5 Instantiation template .....	66
4.8 Usage Instructions for GTP_LUT4 .....	67
4.8.1 Supported Devices .....	67
4.8.2 Description of Functionality.....	67
4.8.3 Port Description .....	67
4.8.4 Paramater Description.....	68
4.8.5 Instantiation template .....	68
4.9 Usage Instructions for GTP_LUT5 .....	69
4.9.1 Supported Devices .....	69

4.9.2 Description of Functionality.....	69
4.9.3 Port Description .....	69
4.9.4 Paramater Description.....	70
4.9.5 Instantiation template .....	70
4.10 Usage Instructions for GTP_LUT5CARRY .....	71
4.10.1 Supported Devices .....	71
4.10.2 Description of Functionality.....	71
4.10.3 Port Description .....	72
4.10.4 Paramater Description.....	72
4.10.5 Instantiation template .....	72
4.11 Usage Instructions for GTP_LUT6 .....	73
4.11.1 Supported Devices.....	73
4.11.2 Description of Functionality.....	73
4.11.3 Port Description.....	74
4.11.4 Paramater Description.....	76
4.11.5 Instantiation template .....	76
4.12 Usage Instructions for GTP_LUT7 .....	76
4.12.1 Supported Devices .....	76
4.12.2 Description of Functionality.....	76
4.12.3 Port Description .....	77
4.12.4 Paramater Description.....	81
4.12.5 Instantiation template .....	81
4.13 Usage Instructions for GTP_LUT8 .....	81
4.13.1 Supported Devices .....	81
4.13.2 Description of Functionality.....	81
4.13.3 Port Description .....	82
4.13.4 Paramater Description.....	89
4.13.5 Instantiation template .....	89
4.14 GTP_LUTMUX4 Usage Instructions.....	90
4.14.1 Supported Devices .....	90
4.14.2 Description of Functionality.....	90
4.14.3 Port Description .....	90
4.14.4 Instantiation template .....	91
4.15 Usage Instructions for GTP_MUX2LUT6.....	91
4.15.1 Supported Devices .....	91
4.15.2 Description of Functionality.....	91
4.15.3 Port Description .....	92
4.15.4 Instantiation template .....	92
4.16 Usage Instructions for GTP_MUX2LUT7 .....	92

4.16.1 Supported Devices .....	92
4.16.2 Description of Functionality.....	92
4.16.3 Port Description .....	93
4.16.4 Instantiation template .....	93
<b>4.17 Usage Instructions for GTP_MUX2LUT8.....</b>	<b>93</b>
4.17.1 Supported Devices .....	93
4.17.2 Description of Functionality.....	93
4.17.3 Port Description .....	94
4.17.4 Instantiation template .....	94
<b>4.18 Usage Instructions for GTP_LUT5M.....</b>	<b>95</b>
4.18.1 Supported Devices .....	95
4.18.2 Description of Functionality.....	95
4.18.3 Port Description .....	95
4.18.4 Paramater Description.....	96
4.18.5 Instantiation template .....	97
<b>Chapter 5 Usage Instructions for ROM &amp; RAM-related GTPs.....</b>	<b>98</b>
5.1 General Introduction.....	98
5.2 Usage Instructions for GTP_ROM32X1 .....	98
5.2.1 Supported Devices .....	98
5.2.2 Description of Functionality.....	98
5.2.3 Port Description .....	99
5.2.4 Paramater Description.....	99
5.2.5 Detailed Functional Description.....	99
5.2.6 Instantiation template .....	99
5.3 Usage Instructions for GTP_ROM64X1 .....	100
5.3.1 Supported Devices .....	100
5.3.2 Description of Functionality.....	100
5.3.3 Port Description .....	101
5.3.4 Paramater Description.....	101
5.3.5 Detailed Functional Description.....	101
5.3.6 Instantiation template .....	102
5.4 Usage Instructions for GTP_ROM128X1 .....	102
5.4.1 Supported Devices .....	102
5.4.2 Description of Functionality.....	102
5.4.3 Port Description .....	103
5.4.4 Paramater Description.....	103
5.4.5 Detailed Functional Description.....	103
5.4.6 Instantiation template .....	104
5.5 Usage Instructions for GTP_ROM256X1 .....	105

5.5.1 Supported Devices .....	105
5.5.2 Description of Functionality.....	105
5.5.3 Port Description .....	105
5.5.4 Paramater Description.....	106
5.5.5 Detailed Functional Description.....	106
5.5.6 Instantiation template .....	106
<b>5.6 Usage Instructions for GTP_RAM16X4SP.....</b>	<b>107</b>
5.6.1 Supported Devices .....	107
5.6.2 Description of Functionality.....	107
5.6.3 Port Description .....	108
5.6.4 Paramater Description.....	108
5.6.5 Detailed Functional Description.....	108
5.6.6 Instantiation template .....	109
<b>5.7 Usage Instructions for GTP_RAM16X4DP.....</b>	<b>109</b>
5.7.1 Supported Devices .....	109
5.7.2 Description of Functionality.....	109
5.7.3 Port Description .....	110
5.7.4 Paramater Description.....	110
5.7.5 Detailed Functional Description.....	110
5.7.6 Instantiation template .....	111
<b>5.8 Usage Instructions for GTP_RAM16X1SP.....</b>	<b>112</b>
5.8.1 Supported Devices .....	112
5.8.2 Description of Functionality.....	112
5.8.3 Port Description .....	112
5.8.4 Paramater Description.....	113
5.8.5 Detailed Functional Description.....	113
5.8.6 Instantiation template .....	113
<b>5.9 Usage Instructions for GTP_RAM16X1DP.....</b>	<b>114</b>
5.9.1 Supported Devices .....	114
5.9.2 Description of Functionality.....	114
5.9.3 Port Description .....	114
5.9.4 Paramater Description.....	115
5.9.5 Detailed Functional Description.....	115
5.9.6 Instantiation template .....	115
<b>5.10 Usage Instructions for GTP_RAM32X1SP.....</b>	<b>116</b>
5.10.1 Supported Devices .....	116
5.10.2 Description of Functionality.....	116
5.10.3 Port Description .....	117
5.10.4 Paramater Description.....	117

5.10.5 Detailed Functional Description.....	117
5.10.6 Instantiation template .....	117
<b>5.11 Usage Instructions for GTP_RAM32X1DP .....</b>	<b>118</b>
5.11.1 Supported Devices.....	118
5.11.2 Description of Functionality.....	118
5.11.3 Port Description.....	119
5.11.4 Paramater Description .....	119
5.11.5 Detailed Functional Description.....	119
5.11.6 Instantiation template .....	120
<b>5.12 Usage Instructions for GTP_DRM9K.....</b>	<b>120</b>
5.12.1 Supported Devices .....	120
5.12.2 Description of Functionality.....	121
5.12.3 Port Description .....	122
5.12.4 Paramater Description .....	123
5.12.5 Detailed Functional Description.....	124
5.12.6 Instantiation template .....	127
<b>5.13 Usage Instructions for GTP_DRM18K.....</b>	<b>131</b>
5.13.1 Supported Devices .....	131
5.13.2 Description of Functionality.....	131
5.13.3 Port Description .....	132
5.13.4 Paramater Description .....	133
5.13.5 Detailed Functional Description.....	134
5.13.6 Instantiation template .....	137
5.13.7 How to Use GTP .....	143
<b>5.14 Usage Instructions for GTP_FIFO18K .....</b>	<b>143</b>
5.14.1 Supported Devices .....	143
5.14.2 Description of Functionality.....	143
5.14.3 Port Description .....	144
5.14.4 Paramater Description .....	144
5.14.5 Detailed Functional Description.....	145
5.14.6 Expansion via Cascading .....	149
5.14.7 Instantiation template .....	152
<b>Chapter 6 Usage Instructions for IO-related GTPs.....</b>	<b>154</b>
<b>6.1 Usage Instructions for GTP_INBUF.....</b>	<b>154</b>
6.1.1 Supported Devices .....	154
6.1.2 Description of Functionality.....	154
6.1.3 Port Description .....	154
6.1.4 Paramater Description .....	154
6.1.5 Instantiation template .....	155

6.2 Usage Instructions for GTP_INBUFDS .....	155
6.2.1 Supported Devices .....	155
6.2.2 Description of Functionality.....	155
6.2.3 Port Description .....	155
6.2.4 Paramater Description.....	156
6.2.5 Instantiation template .....	156
6.3 Usage Instructions for GTP_INBUFG .....	156
6.3.1 Supported Devices .....	156
6.3.2 Description of Functionality.....	156
6.3.3 Port Description .....	157
6.3.4 Paramater Description.....	157
6.3.5 Instantiation template .....	157
6.4 Usage Instructions for GTP_INBUFGDS.....	158
6.4.1 Supported Devices .....	158
6.4.2 Description of Functionality.....	158
6.4.3 Port Description .....	158
6.4.4 Paramater Description.....	158
6.4.5 Instantiation template .....	158
6.5 Usage Instructions for GTP_IOBUF .....	159
6.5.1 Description of Functionality.....	159
6.5.2 Port Description .....	159
6.5.3 Paramater Description.....	160
6.5.4 Instantiation template .....	160
6.6 Usage Instructions for GTP_IOBUFCO .....	160
6.6.1 Supported Devices .....	160
6.6.2 Description of Functionality.....	161
6.6.3 Port Description .....	161
6.6.4 Paramater Description.....	161
6.6.5 Instantiation template .....	162
6.7 Usage Instructions for GTP_IOBUFDS .....	162
6.7.1 Supported Devices .....	162
6.7.2 Description of Functionality.....	162
6.7.3 Port Description .....	163
6.7.4 Paramater Description.....	163
6.7.5 Instantiation template .....	163
6.8 GTP_OUTBUF Usage Instructions.....	164
6.8.1 Supported Devices .....	164
6.8.2 Description of Functionality.....	164
6.8.3 Port Description .....	164

6.8.4 Paramater Description.....	164
6.8.5 Instantiation template .....	164
6.9 Usage Instructions for GTP_OUTBUFCO .....	165
6.9.1 Supported Devices .....	165
6.9.2 Description of Functionality.....	165
6.9.3 Port Description .....	165
6.9.4 Paramater Description.....	166
6.9.5 Instantiation template .....	166
6.10 Usage Instructions for GTP_OUTBUFDS .....	166
6.10.1 Supported Devices .....	166
6.10.2 Description of Functionality.....	166
6.10.3 Port Description .....	167
6.10.4 Paramater Description.....	167
6.10.5 Instance Template.....	167
6.11 Usage Instructions for GTP_OUTBUFT .....	167
6.11.1 Supported Devices.....	167
6.11.2 Description of Functionality.....	167
6.11.3 Port Description.....	168
6.11.4 Paramater Description.....	168
6.11.5 Instance Template.....	168
6.12 Usage Instructions for GTP_OUTBUFTCO .....	169
6.12.1 Supported Devices .....	169
6.12.2 Description of Functionality.....	169
6.12.3 Port Description .....	169
6.12.4 Paramater Description.....	169
6.12.5 Instance Template.....	170
6.13 Usage Instructions for GTP_OUTBUFTDS.....	170
6.13.1 Supported Devices .....	170
6.13.2 Description of Functionality.....	170
6.13.3 Port Description .....	170
6.13.4 Paramater Description.....	171
6.13.5 Instance Template.....	171
6.14 Usage Instructions for GTP_IODELAY.....	171
6.14.1 Supported Devices .....	171
6.14.2 Description of Functionality.....	171
6.14.3 Port Description .....	172
6.14.4 Paramater Description.....	172
6.14.5 Instance Template.....	172
6.15 Usage Instructions for GTP_IOBUF_RX_MIPI.....	173

6.15.1 Supported Devices .....	173
6.15.2 Description of Functionality.....	173
6.15.3 Port Description .....	174
6.15.4 Paramater Description.....	174
6.15.5 Instance Template.....	174
<b>6.16 Usage Instructions for GTP_Iobuf_TX_MIPI .....</b>	<b>175</b>
6.16.1 Supported Devices .....	175
6.16.2 Description of Functionality.....	175
6.16.3 Paramater Description.....	177
6.16.4 Instance Template.....	177
<b>6.17 Usage Instructions for GTP_ISERDES.....</b>	<b>178</b>
6.17.1 Supported Devices .....	178
6.17.2 Description of Functionality.....	178
6.17.3 Port Description .....	178
6.17.4 Paramater Description.....	179
6.17.5 Instantiation template .....	179
<b>6.18 Usage Instructions for GTP_OSERDES .....</b>	<b>180</b>
6.18.1 Supported Devices .....	180
6.18.2 Description of Functionality.....	180
6.18.3 Port Description .....	180
6.18.4 Paramater Description.....	181
6.18.5 Instantiation template .....	181
<b>Chapter 7 Usage Instructions for Clock-Related GTPs .....</b>	<b>183</b>
7.1 GTP_CLKBUFG.....	183
7.1.1 Supported Devices .....	183
7.1.2 Description of Functionality.....	183
7.1.3 Port Description .....	183
7.1.4 Instantiate Module.....	183
7.1.5 Detailed Description of Functionality .....	183
7.2 GTP_CLKBUFGCE.....	184
7.2.1 Supported Devices .....	184
7.2.2 Description of Functionality.....	184
7.2.3 Port Description .....	184
7.2.4 Paramater Description.....	184
7.2.5 Instantiate Module.....	184
7.2.6 Detailed Description of Functionality .....	185
7.3 GTP_CLKBUFGMUX .....	185
7.3.1 Supported Devices .....	185
7.3.2 Description of Functionality.....	185

7.3.3 Port Description .....	186
7.3.4 Paramater Description .....	186
7.3.5 Instantiate Module.....	186
7.3.6 Detailed Description of Functionality .....	187
<b>7.4 GTP_CLKBUFR.....</b>	<b>188</b>
7.4.1 Supported Devices .....	188
7.4.2 Description of Functionality.....	188
7.4.3 Port Description .....	188
7.4.4 Instantiate Module.....	188
7.4.5 Detailed Description of Functionality .....	189
<b>7.5 GTP_IOCLKBUF .....</b>	<b>189</b>
7.5.1 Supported Devices .....	189
7.5.2 Description of Functionality.....	189
7.5.3 Port Description .....	189
7.5.4 Paramater Description.....	189
7.5.5 Instantiate Module.....	190
7.5.6 Detailed Description of Functionality .....	190
<b>7.6 GTP_IOCLKDIV .....</b>	<b>191</b>
7.6.1 Supported Devices .....	191
7.6.2 Description of Functionality.....	191
7.6.3 Port Description .....	191
7.6.4 Paramater Description.....	191
7.6.5 Instantiate Module.....	191
7.6.6 Detailed Description of Functionality .....	192
<b>7.7 GTP_IOCLKDELAY .....</b>	<b>192</b>
7.7.1 Supported Devices .....	192
7.7.2 Description of Functionality.....	192
7.7.3 Port Description .....	192
7.7.4 Paramater Description.....	193
7.7.5 Instantiate Module.....	193
7.7.6 Detailed Description of Functionality .....	193
<b>7.8 GTP_OSC_E1 .....</b>	<b>194</b>
7.8.1 Supported Devices .....	194
7.8.2 Description of Functionality.....	194
7.8.3 Port Description .....	194
7.8.4 Paramater Description.....	194
7.8.5 Instantiate Module.....	194
7.8.6 Detailed Description of Functionality .....	195
<b>7.9 GTP_OSC_E3 .....</b>	<b>195</b>

7.9.1 Supported Devices .....	195
7.9.2 Description of Functionality.....	195
7.9.3 Port Description .....	195
7.9.4 Paramater Description.....	195
7.9.5 Instantiate Module.....	196
7.9.6 Detailed Description of Functionality .....	196
<b>7.10 GTP_CRYSTAL.....</b>	<b>196</b>
7.10.1 Supported Devices .....	196
7.10.2 Description of Functionality.....	196
7.10.3 Port Description .....	197
7.10.4 Paramater Description.....	197
7.10.5 Instantiate Module.....	197
7.10.6 Detailed Description of Functionality .....	197
<b>7.11 GTP_PLL_E1 .....</b>	<b>198</b>
7.11.1 Supported Devices.....	198
7.11.2 Description of Functionality.....	198
7.11.3 Port Description.....	198
7.11.4 Paramater Description.....	200
7.11.5 Instantiate Module.....	202
7.11.6 Detailed Functional Description.....	206
<b>7.12 GTP_PLL_E3 .....</b>	<b>206</b>
7.12.1 Supported Devices .....	206
7.12.2 Description of Functionality.....	206
7.12.3 Port Description .....	206
7.12.4 Paramater Description.....	208
7.12.5 Instantiate Module.....	210
7.12.6 Detailed Functional Description.....	213
<b>7.13 GTP_DLL.....</b>	<b>213</b>
7.13.1 Supported Devices .....	213
7.13.2 Description of Functionality.....	213
7.13.3 Port Description .....	214
7.13.4 Paramater Description.....	214
7.13.5 Instantiate Module.....	214
7.13.6 Detailed Description of Functionality .....	215
<b>Chapter 8 Usage Instructions for Configuration-related GTPs .....</b>	<b>217</b>
8.1 Usage Instructions for GTP_EFUSECODE .....	217
8.1.1 Supported Devices .....	217
8.1.2 Description of Functionality.....	217
8.1.3 Port Description .....	217

8.1.4 Paramater Description.....	217
8.1.5 Instantiation template .....	217
8.1.6 Detailed Functional Description.....	218
<b>8.2 Usage Instructions for GTP_IPAL_E1 .....</b>	<b>218</b>
8.2.1 Supported Devices .....	218
8.2.2 Description of Functionality.....	218
8.2.3 Port Description .....	218
8.2.4 Paramater Description.....	219
8.2.5 Instantiation template .....	219
8.2.6 Detailed Functional Description.....	220
<b>8.3 Usage Instructions forGTP_SCANCHAIN_E1 .....</b>	<b>220</b>
8.3.1 Supported Devices .....	220
8.3.2 Description of Functionality.....	220
8.3.3 Port Description .....	221
8.3.4 Paramater Description.....	221
8.3.5 Instantiation template .....	221
8.3.6 Detailed Functional Description.....	222
<b>8.4 Usage Instructions for GTP_FLASHIF.....</b>	<b>222</b>
8.4.1 Supported Devices .....	222
8.4.2 Description of Functionality.....	222
8.4.3 Port Description .....	222
8.4.4 Instantiation template .....	223
<b>8.5 Usage Instructions for GTP_UDID .....</b>	<b>223</b>
8.5.1 Supported Devices .....	223
8.5.2 Description of Functionality.....	223
8.5.3 Port Description .....	224
8.5.4 Paramater Description.....	224
8.5.5 Instantiation template .....	225
8.5.6 Detailed Functional Description.....	225
<b>Chapter 9 Usage Instructions for DDR GTP.....</b>	<b>226</b>
9.1 Usage Instructions for GTP_DDC_E1 .....	226
9.1.1 Supported Devices .....	226
9.1.2 Description of Functionality.....	226
9.1.3 Port Description .....	226
9.1.4 Paramater Description.....	228
9.1.5 Instantiation template .....	229
9.1.6 Detailed Functional Description.....	230
9.2 Usage Instructions for GTP_DDRC .....	231
9.2.1 Supported Devices .....	231

9.2.2 Description of Functionality.....	231
9.2.3 Port Description .....	231
9.2.4 Instantiation template .....	236
9.2.5 Detailed Functional Description.....	242
9.2.6 How to Use GTP .....	242
<b>9.3 Usage Instructions for GTP_DDRPHY.....</b>	<b>242</b>
9.3.1 Supported Devices .....	242
9.3.2 Description of Functionality.....	242
9.3.3 Port Description .....	243
9.3.4 Paramater Description.....	248
9.3.5 Detailed Functional Description.....	253
<b>9.4 Usage Instructions for GTP_IDDR_E2.....</b>	<b>253</b>
9.4.1 Supported Devices .....	253
9.4.2 Description of Functionality.....	253
9.4.3 Port Description .....	254
9.4.4 Paramater Description.....	254
9.4.5 Instantiation template .....	254
<b>9.5 Usage Instructions for GTP_ODDR_E2 .....</b>	<b>255</b>
9.5.1 Supported Devices .....	255
9.5.2 Description of Functionality.....	255
9.5.3 Port Description .....	255
9.5.4 Paramater Description.....	255
9.5.5 Instantiation template .....	256
<b>Chapter 10 Usage Instructions for Other GTPs.....</b>	<b>257</b>
10.1 Usage Instructions for GTP_GRS .....	257
10.1.1 Supported Devices .....	257
10.1.2 Description of Functionality.....	257
10.1.3 Port Description .....	257
10.1.4 Instantiation template .....	257
10.2 Usage Instructions for GTP_START_E1 .....	257
10.2.1 Supported Devices .....	257
10.2.2 Description of Functionality.....	257
10.2.3 Port Description .....	258
10.2.4 Instantiation template .....	258
10.2.5 Detailed Functional Description.....	258
10.3 Usage Instructions for GTP_RES_CAL.....	258
10.3.1 Supported Devices .....	258
10.3.2 Description of Functionality.....	258
10.3.3 Port Description .....	259

10.3.4 Paramater Description.....	259
10.3.5 Instantiation template .....	259
10.3.6 Detailed Functional Description.....	259
10.3.7 How to Use GTP .....	260
10.4 Usage Instructions for GTP_APM_E1 .....	260
10.4.1 Supported Devices .....	260
10.4.2 Description of Functionality.....	260
10.4.3 Port Description .....	261
10.4.4 Paramater Description.....	262
10.4.5 Detailed Functional Description.....	264
10.4.6 Instantiation template .....	268
10.5 Usage Instructions for GTP_HSST_E2.....	270
10.5.1 Supported Devices .....	270
10.5.2 Description of Functionality.....	270
10.5.3 Port Description .....	271
10.5.4 Paramater Description.....	274
10.5.5 Instantiation template .....	294
10.6 Usage Instructions for GTP_ADC_E1 .....	354
10.6.1 Supported Devices .....	354
10.6.2 Description of Functionality.....	355
10.6.3 Port Description .....	355
10.6.4 Paramater Description.....	356
10.6.5 Instantiation template .....	359
10.7 Usage Instructions for GTP_PCIEGEN2 .....	360
10.7.1 Supported Devices .....	360
10.7.2 Description of Functionality.....	360
10.7.3 Port Description .....	361
10.7.4 Paramater Description.....	373
10.7.5 Instantiation template .....	374
<b>Disclaimer.....</b>	<b>381</b>

## Tables

Table 1-1 Logos Family GTP Classification.....	31
Table 2-1 Device Models That Support GTP_DFF .....	34
Table 2-2 GTP_DFF Port Description .....	34
Table 2-3 GTP_DFF Parameter Description.....	35
Table 2-4 Device Models That Support GTP_DFF_C.....	35
Table 2-5 GTP_DFF_C Port Description .....	36
Table 2-6 GTP_DFF_C Parameter Description.....	36
Table 2-7 Device Models That Support GTP_DFF_E.....	37
Table 2-8 GTP_DFF_E Port Description.....	37
Table 2-9 GTP_DFF_E Parameter Description .....	38
Table 2-10 Device Models That Support GTP_DFF_CE .....	38
Table 2-11 GTP_DFF_CE Port Description .....	39
Table 2-12 Device Models That Support GTP_DFF_P .....	39
Table 2-13 GTP_DFF_P Port Description .....	40
Table 2-14 GTP_DFF_P Parameter Description .....	40
Table 2-15 Device Models That Support GTP_DFF_PE.....	41
Table 2-16 GTP_DFF_PE Port Description .....	42
Table 2-17 GTP_DFF_PE Parameter Description .....	42
Table 2-18 Device Models That Support GTP_DFF_R.....	43
Table 2-19 GTP_DFF_R Port Description .....	43
Table 2-20 GTP_DFF_R Parameter Description .....	43
Table 2-21 Device Models That Support GTP_DFF_RE .....	44
Table 2-22 GTP_DFF_RE Port Description .....	45
Table 2-23 GTP_DFF_RE Parameter Description .....	45
Table 2-24 Device Models That Support GTP_DFF_S .....	46
Table 2-25 GTP_DFF_S Port Description .....	46
Table 2-26 GTP_DFF_S Parameter Description .....	46
Table 2-27 Device Models That Support GTP_DFF_SE.....	47
Table 2-28 GTP_DFF_SE Port Description .....	48
Table 2-29 GTP_DFF_SE Parameter Description .....	48
Table 3-1 Device Models That Support GTP_DLATCH.....	49
Table 3-2 GTP_DLATCH Port Description.....	49
Table 3-3 GTP_DLATCH Parameter Description .....	50
Table 3-4 Device Models That Support GTP_DLATCH_E .....	50
Table 3-5 GTP_DLATCH_E Port Description .....	51
Table 3-6 GTP_DLATCH_E Parameter Description.....	51

Table 3-7 GTP_DLATCH_C .....	52
Table 3-8 GTP_DLATCH_C Port Description.....	52
Table 3-9 GTP_DLATCH_C Parameter Description .....	53
Table 3-10 Device Models That Support GTP_DLATCH.....	53
Table 3-11 GTP_DLATCH_CE Port Description.....	54
Table 3-12 GTP_DLATCH_C Parameter Description.....	54
Table 3-13 Device Models That Support GTP_DLATCH_P.....	55
Table 3-14 GTP_DLATCH_P Port Description.....	55
Table 3-15 GTP_DLATCH_P Parameter Description .....	56
Table 3-16 Device Models That Support GTP_DLATCH_PE .....	56
Table 3-17 GTP_DLATCH_PE Port Description.....	57
Table 3-18 GTP_DLATCH_PE Parameter Description.....	57
Table 4-1 Device Models That Support GTP_ONE .....	59
Table 4-2 GTP_ONE Port Description .....	59
Table 4-3 Device Models That Support GTP_ZERO .....	60
Table 4-4 GTP_ZERO Port Description .....	60
Table 4-5 Device Models That Support GTP_BUF.....	60
Table 4-6 GTP_BUF Port Description.....	61
Table 4-7 Device Models That Support GTP_INV .....	61
Table 4-8 GTP_INV Port Description .....	62
Table 4-9 Device Models That Support GTP_LUT1 .....	62
Table 4-10 GTP_LUT1 Port Description.....	63
Table 4-11 GTP_LUT1 Truth Table.....	63
Table 4-12 GTP_LUT1 Parameter Description .....	63
Table 4-13 Device Models That Support GTP_LUT2.....	64
Table 4-14 GTP_LUT2 Port Description.....	64
Table 4-15 GTP_LUT2 Truth Table .....	64
Table 4-16 GTP_LUT2 Parameter Description .....	65
Table 4-17 Device Models That Support GTP_LUT3.....	65
Table 4-18 GTP_LUT3 Port Description.....	66
Table 4-19 GTP_LUT3 Truth Table .....	66
Table 4-20 GTP_LUT3 Parameter Description .....	66
Table 4-21 Device Models That Support GTP_LUT4.....	67
Table 4-22 GTP_LUT4 Port Description.....	67
Table 4-23 GTP_LUT4 Truth Table .....	67
Table 4-24 GTP_LUT4 Parameter Description .....	68
Table 4-25 Device Models That Support GTP_LUT5.....	69
Table 4-26 GTP_LUT5 Port Description.....	69
Table 4-27 GTP_LUT5 Truth Table .....	69

Table 4-28 GTP_LUT5 Parameter Description .....	70
Table 4-29 Device Models That Support GTP_LUT5CARRY .....	71
Table 4-30 GTP_LUT5CARRY Port Description .....	72
Table 4-31 GTP_LUT5CARRY Parameter Description.....	72
Table 4-32 Device Models That Support GTP_LUT6.....	73
Table 4-33 GTP_LUT6 Port Description.....	74
Table 4-34 GTP_LUT6 Truth Table .....	74
Table 4-35 GTP_LUT6 Parameter Description .....	76
Table 4-36 Device Models That Support GTP_LUT7.....	76
Table 4-37 GTP_LUT7 Port Description.....	77
Table 4-38 GTP_LUT7 Truth Table .....	77
Table 4-39 GTP_LUT7 Parameter Description .....	81
Table 4-40 Device Models That Support GTP_LUT8.....	81
Table 4-41 GTP_LUT8 Port Description.....	82
Table 4-42 GTP_LUT8 Truth Table .....	82
Table 4-43 GTP_LUT8 Parameter Description .....	89
Table 4-44 Device Models That Support GTP_LUTMUX4.....	90
Table 4-45 GTP_LUTMUX4Port Description .....	90
Table 4-46 GTP_LUTMUX4 Truth Table .....	90
Table 4-47 Device Models That Support GTP_MUX2LUT6.....	91
Table 4-48 GTP_MUX2LUT6 Port Description .....	92
Table 4-49 Device Models That Support GTP_MUX2LUT7.....	92
Table 4-50 GTP_MUX2LUT7 Port Description .....	93
Table 4-51 Device Models That Support GTP_MUX2LUT8.....	93
Table 4-52 GTP_MUX2LUT8 Port Description .....	94
Table 4-53 Device Models That Support GTP_LUT5M .....	95
Table 4-54 GTP_LUT5CARRY Port Description .....	95
Table 4-55 GTP_LUT5M Truth Table .....	95
Table 4-56 GTP_LUT5M Parameter Description.....	96
Table 5-1 Device Models That Support GTP_ROM32X1.....	98
Table 5-2 GTP_ROM32X1 Port Description .....	99
Table 5-3 GTP_ROM32X1 Parameter Description .....	99
Table 5-4 Device Models That Support GTP_ROM64X1.....	100
Table 5-5 GTP_ROM64X1 Port Description .....	101
Table 5-6 GTP_ROM64X1 Parameter Description .....	101
Table 5-7 Device Models That Support GTP_ROM128X1.....	102
Table 5-8 GTP_ROM128X1 Port Description .....	103
Table 5-9 GTP_ROM128X1 Parameter Description .....	103
Table 5-10 Device Models That Support GTP_ROM256X1.....	105

Table 5-11 GTP_ROM256X1 Port Description.....	105
Table 5-12 GTP_ROM256X1 Parameter Description .....	106
Table 5-13 Device Models That Support GTP_RAM16X4SP .....	107
Table 5-14 GTP_RAM16X4SP Port Description .....	108
Table 5-15 GTP_RAM16X4SP Parameter Description.....	108
Table 5-16 Device Models That Support GTP_RAM16X4DP.....	109
Table 5-17 GTP_RAM16X4DP Port Description .....	110
Table 5-18 GTP_RAM16X4DP Parameter Description.....	110
Table 5-19 Device Models That Support GTP_RAM16X1SP .....	112
Table 5-20 GTP_RAM16X1SP Port Description .....	112
Table 5-21 GTP_RAM16X4SP Parameter Description.....	113
Table 5-22 Device Models That Support GTP_RAM16X1DP.....	114
Table 5-23 GTP_RAM16X1DP Port Description .....	114
Table 5-24 GTP_RAM16X1DP Parameter Description.....	115
Table 5-25 Device Models That Support GTP_RAM32X1SP .....	116
Table 5-26 GTP_RAM32X1SP Port Description .....	117
Table 5-27 GTP_RAM32X1SP Parameter Description.....	117
Table 5-28 Device Models That Support GTP_RAM32X1DP.....	118
Table 5-29 GTP_RAM32X1DP Port Description .....	119
Table 5-30 GTP_RAM32X1DP Parameter Description.....	119
Table 5-31 Device Models That Support GTP_DRM9K.....	120
Table 5-32 GTP_DRM9K Port Description .....	122
Table 5-33 GTP_DRM9K Parameter Description .....	123
Table 5-34 Device Models That Support GTP_DRM18K.....	131
Table 5-35 GTP_DRM18K Port Description .....	132
Table 5-36 GTP_DRM18K Parameter Description .....	133
Table 5-37 Device Models That Support GTP_FIFO18K .....	143
Table 5-38 GTP_FIFO18K Port List .....	144
Table 5-39 GTP_FIFO18K Parameter List.....	145
Table 6-1 Device Models That Support GTP_INBUF.....	154
Table 6-2 GTP_INBUF Port List Description .....	154
Table 6-3 GTP_INBUF Parameter List .....	154
Table 6-4 Device Models That Support GTP_INBUFDS .....	155
Table 6-5 GTP_INBUFDS Port List Description .....	155
Table 6-6 GTP_INBUFDS Parameter List .....	156
Table 6-7 Device Models That Support GTP_INBUFG .....	156
Table 6-8 GTP_INBUFG Port List Description .....	157
Table 6-9 GTP_INBUFG Parameter List Description.....	157
Table 6-10 Device Models That Support GTP_INBUFGDS.....	158

Table 6-11 GTP_INBUFGDS Port List Description .....	158
Table 6-12 GTP_INBUFGDS Parameter List Description.....	158
Table 6-13 Device Models That Support GTP_IOBUF.....	159
Table 6-14 GTP_IOBUF Port List Description .....	159
Table 6-15 GTP_IOBUF Parameter List Description.....	160
Table 6-16 Device Models That Support GTP_IOBUFCO .....	160
Table 6-17 GTP_IOBUFCO Port List Description.....	161
Table 6-18 GTP_IOBUFCO Parameter List Description .....	161
Table 6-19 Device Models That Support GTP_IOBUFDS.....	162
Table 6-20 GTP_IOBUFDS Port List Description .....	163
Table 6-21 GTP_IOBUFDS Parameter List Description.....	163
Table 6-22 Device Models That Support GTP_OUTBUF.....	164
Table 6-23 GTP_OUTBUF Port List Description .....	164
Table 6-24 GTP_OUTBUF Parameter List Description.....	164
Table 6-25 Device Models That Support GTP_OUTBUFCO .....	165
Table 6-26 GTP_OUTBUFCO Port List Information .....	165
Table 6-27 GTP_OUTBUFCO Parameter List Information.....	166
Table 6-28 Device Models That Support GTP_OUTBUFDS.....	166
Table 6-29 GTP_OUTBUFDS Port List Description .....	167
Table 6-30 GTP_OUTBUFDS Parameter List Description.....	167
Table 6-31 Device Models That Support GTP_OUTBUFT .....	167
Table 6-32 GTP_OUTBUFT Port List Description .....	168
Table 6-33 GTP_OUTBUFT Parameter List Description .....	168
Table 6-34 Device Models That Support GTP_OUTBUFTCO .....	169
Table 6-35 GTP_OUTBUFTCO Port Description .....	169
Table 6-36 GTP_OUTBUFTCO Parameter Description.....	169
Table 6-37 Device Models That Support GTP_OUTBUFTDS .....	170
Table 6-38 GTP_OUTBUFTDS Port Description .....	170
Table 6-39 GTP_OUTBUFTDS Parameter Description .....	171
Table 6-40 Device Models That Support GTP_IODELAY .....	171
Table 6-41 GTP_IODELAY Port Description .....	172
Table 6-42 GTP_IODELAY Parameter Description.....	172
Table 6-43 Device Models That Support GTP_IOBUF_RX_MIPI.....	173
Table 6-44 GTP_IOBUF_RX_MIPI Port Description .....	174
Table 6-45 GTP_IOBUF_RX_MIPI Parameter Description .....	174
Table 6-46 Device Models That Support GTP_IOBUF_TX_MIPI.....	175
Table 6-47 GTP_IOBUF_TX_MIPI Port Description.....	176
Table 6-48 GTP_IOBUF_TX_MIPI Parameter Description .....	177
Table 6-49 Device Models That Support GTP_ISERDES .....	178

Table 6-50 GTP_I SERDES Port List.....	178
Table 6-51 GTP_I SERDES Parameter List .....	179
Table 6-52 Device Models That Support GTP_O SERDES .....	180
Table 6-53 GTP_O SERDES Port Description.....	180
Table 6-54 GTP_O SERDES Parameter Description .....	181
Table 7-1 Device Models That Support GTP_CLKBUFG .....	183
Table 7-2 GTP_CLKBUFG Port Description.....	183
Table 7-3 Device Models That Support GTP_CLKBUFGCE .....	184
Table 7-4 GTP_CLKBUFGCE Port Description.....	184
Table 7-5 GTP_CLKBUFGCE Parameter Description .....	184
Table 7-6 Device Models That Support GTP_CLKBUFGMUX.....	185
Table 7-7 GTP_CLKBUFGMUX Port Description .....	186
Table 7-8 GTP_CLKBUFGMUX Parameter Description .....	186
Table 7-9 Device Models That Support GTP_CLKBUFR .....	188
Table 7-10 GTP_CLKBUFR Port Description .....	188
Table 7-11 Device Models That Support GTP_IOCLKBUF.....	189
Table 7-12 GTP_IOCLKBUF Port Description .....	189
Table 7-13 Parameter Description of GTP_IOCLKBUF.....	189
Table 7-14 Device Models That Support GTP_IOCLKDIV .....	191
Table 7-15 GTP_IOCLKDIV Port Description .....	191
Table 7-16 GTP_IOCLKDIV Parameter Description.....	191
Table 7-17 Device Models That Support GTP_IOCLKDELAY .....	192
Table 7-18 GTP_IOCLKDELAY Port Description .....	192
Table 7-19 GTP_IOCLKDELAY Parameter Description.....	193
Table 7-20 Device Models That Support GTP_OSC_E1 .....	194
Table 7-21 GTP_OSC_E1 Port Description .....	194
Table 7-22 GTP_OSC_E1 Parameter Description.....	194
Table 7-23 GTP_OSC_E1 Division Factor and Output Clock Relationship .....	195
Table 7-24 Device Models That Support GTP_OSC_E3 .....	195
Table 7-25 GTP_OSC_E3 Port Description .....	195
Table 7-26 GTP_OSC_E3 Parameter Description.....	195
Table 7-27 GTP_OSC_E3 Division Factor and Output Clock Relationship .....	196
Table 7-28 Device Models That Support GTP_CRYSTAL.....	196
Table 7-29 GTP_CRYSTAL Port Description.....	197
Table 7-30 GTP_CRYSTAL Parameter Description .....	197
Table 7-31 Device Models That Support GTP_PLL_E1 .....	198
Table 7-32 GTP_PLL_E1 Port Description.....	198
Table 7-33 GTP_PLL_E1 Parameter Description .....	200
Table 7-34 Device Models That Support GTP_PLL_E3 .....	206

Table 7-35 GTP_PLL_E3 Port Description.....	206
Table 7-36 GTP_PLL_E3 Parameter Description .....	208
Table 7-37 Device Models That Support GTP_DLL.....	213
Table 7-38 GTP_DLL Port Description.....	214
Table 7-39 GTP_DLL Parameter Description .....	214
Table 7-40 List of Phase Shifts for Different Configurations .....	215
Table 7-41 List of Digital Codes for Different Frequencies and Phase Shift Settings.....	215
Table 8-1 Device Models That Support GTP_EFUSECODE.....	217
Table 8-2 GTP_EFUSECODE Port List.....	217
Table 8-3 GTP_EFUSECODE Parameter List .....	217
Table 8-4 Device Models That Support GTP_IPAL_E1.....	218
Table 8-5 GTP_IPAL_E1 Port Description .....	218
Table 8-6 GTP_IPAL_E1 Parameter Description .....	219
Table 8-7 Device Models That Support GTP_SCANCHAIN_E1 .....	220
Table 8-8 GTP_SCANCHAIN_E1 Port Description.....	221
Table 8-9 GTP_SCANCHAIN_E1 Parameter Description .....	221
Table 8-10 Device Models That Support GTP_FLASHIF .....	222
Table 8-11 GTP_FLASHIF Port Description .....	222
Table 8-12 Device Models That Support GTP_UDID.....	223
Table 8-13 GTP_UDID Port Description .....	224
Table 8-14 GTP_UDID Parameter Description.....	224
Table 9-1 Device Models That Support GTP_DDC_E1.....	226
Table 9-2 GTP_DDC_E1 Port Description .....	226
Table 9-3 GTP_DDC_E1 Parameter Description .....	228
Table 9-4 Device Models That Support GTP_DDRC .....	231
Table 9-5 GTP_DDRC Port Description .....	231
Table 9-6 Device Models That Support GTP_DDRPHY .....	242
Table 9-7 GTP_DDRPHY Port Description .....	243
Table 9-8 GTP_DDRPHY Parameter Description.....	248
Table 9-9 Device Models That Support GTP_IDDR_E2 .....	253
Table 9-10 GTP_IDDR_E2 Port Description .....	254
Table 9-11 GTP_IDDR_E2 Parameter Description .....	254
Table 9-12 Device Models That Support GTP_ODDR_E2 .....	255
Table 9-13 GTP_ODDR_E2 Port Description.....	255
Table 9-14 GTP_ODDR_E2 Parameter Description .....	255
Table 10-1 Device Models That Support GTP_GRS.....	257
Table 10-2 GTP_GRS Port Description.....	257
Table 10-3 Device Models That Support GTP_START_E1 .....	257
Table 10-4 GTP_START_E1 Port Description.....	258

Table 10-5 Device Models That Support GTP_RES_CAL .....	258
Table 10-6 GTP_RES_CAL Port Description .....	259
Table 10-7 GTP_RES_CAL Parameter Description.....	259
Table 10-8 Device Models That Support GTP_APM_E1.....	260
Table 10-9 GTP_APM_E1 Port Description .....	261
Table 10-10 GTP_APM_E1 Parameter Description.....	262
Table 10-11 GTP_APM_E1 MODE Port Description.....	263
Table 10-12 ZMUX ListDescription.....	264
Table 10-13 Device Models That Support GTP_HSST_E2 .....	270
Table 10-14 GTP_HSST_E2 Port Description .....	271
Table 10-15 GTP_HSST_E2 Parameter List .....	274
Table 10-16 Device Models That Support GTP_ADC_E1 .....	354
Table 10-17 GTP_ADC_E1 Port Description.....	355
Table 10-18 GTP_ADC_E1 Parameter Description .....	356
Table 10-19 Device Models That Support GTP_PCIEGEN2.....	360
Table 10-20 GTP_PCIEGEN2 Port Description.....	361
Table 10-21 GTP_PCIEGEN2 Parameter Description .....	373

## Figures

---

Figure 2-1 GTP_DFF.....	34
Figure 2-2 GTP_DFF_C .....	36
Figure 2-3 GTP_DFF_E .....	37
Figure 2-4 GTP_DFF_CE .....	39
Figure 2-5 GTP_DFF_P .....	40
Figure 2-6 GTP_DFF_PE .....	41
Figure 2-7 GTP_DFF_R .....	43
Figure 2-8 GTP_DFF_RE .....	44
Figure 2-9 GTP_DFF_S .....	46
Figure 2-10 GTP_DFF_SE .....	47
Diagram 3-1 GTP_DLATCH Diagram.....	49
Figure 3-2 GTP_DLATCH_E Diagram.....	51
Figure 3-3 GTP_DLATCH_C Diagram.....	52
Figure 3-4 GTP_DLATCH_CE Diagram .....	54
Figure 3-5 GTP_DLATCH_P Diagram .....	55
Figure 3-6 GTP_DLATCH_PE Diagram.....	57
Figure 4-1 GTP_ONEStructure Diagram .....	59
Figure 4-2 GTP_ZERO Structure Diagram .....	60
Figure 4-3 GTP_BUFSstructure Diagram.....	61
Figure 4-4 GTP_INV Structure Diagram .....	62
Figure 4-5 GTP_LUT1 Structure Diagram.....	63
Figure 4-6 GTP_LUT2 Structure Diagram.....	64
Figure 4-7 GTP_LUT3 Structure Diagram.....	65
Figure 4-8 GTP_LUT4 Structure Diagram.....	67
Figure 4-9 GTP_LUT5 Structure Diagram.....	69
Figure 4-10 GTP_LUT5CARRY Structure Diagram.....	71
Figure 4-11 GTP_LUT6 Structure Diagram.....	73
Figure 4-12 GTP_LUT7 Structure Diagram.....	77
Figure 4-13 GTP_LUT8 Structure Diagram.....	82
Figure 4-14 GTP_LUTMUX4Structure Diagram .....	90
Figure 4-15 GTP_MUX2LUT6 Structure Diagram .....	92
Figure 4-16 GTP_MUX2LUT7 Structure Diagram .....	93
Figure 4-17 GTP_MUX2LUT8 Structure Diagram .....	94
Figure 4-18 GTP_LUT5M Structure Diagram .....	95
Figure 5-1 GTP_ROM32X1 Structure Block Diagram.....	98
Figure 5-2 GTP_ROM32X1 Waveform Diagram .....	99

Figure 5-3 GTP_ROM64X1 Structure Block Diagram .....	100
Diagram 5-4 GTP_ROM64X1 Waveform Diagram .....	101
Figure 5-5 GTP_ROM128X1 Structure Block Diagram .....	103
Figure 5-6 GTP_ROM128X1 Waveform Diagram .....	104
Figure 5-7 GTP_ROM256X1 Structure Block Diagram .....	105
Figure 5-8 GTP_ROM256X1 Waveform Diagram .....	106
Figure 5-9 GTP_RAM16X4SP Structure Block Diagram .....	107
Figure 5-10 GTP_RAM16X4SP Waveform Diagram .....	108
Figure 5-11 GTP_RAM16X4DP Structure Block Diagram .....	110
Figure 5-12 GTP_RAM16X4DP Waveform Diagram .....	111
Figure 5-13 GTP_RAM16X1SP Structure Block Diagram .....	112
Figure 5-14 GTP_RAM16X1SP Waveform Diagram .....	113
Figure 5-15 GTP_RAM16X1DP Structure Block Diagram .....	114
Figure 5-16 GTP_RAM16X1DP Waveform Diagram .....	115
Diagram 5-17 GTP_RAM32X1SP Structure Block Diagram .....	116
Diagram 5-18 GTP_RAM32X1SP Waveform Diagram .....	117
Diagram 5-19 GTP_RAM32X1DP Structure Block Diagram .....	118
Diagram 5-20 GTP_RAM32X1DP Waveform Diagram .....	120
Figure 5-21 GTP_DRM9K Block Diagram .....	122
Figure 5-22 GTP_DRM9K NW Mode Waveform Diagram .....	124
Figure 5-23 GTP_DRM9K TW Mode Waveform Diagram .....	125
Figure 5-24 GTP_DRM9K RBW Mode Waveform Diagram .....	126
Figure 5-25 GTP_DRM9K SP Interface Diagram .....	127
Figure 5-26 GTP_DRM9K Register Mode Waveform Diagram .....	127
Figure 5-27 GTP_DRM18K Structure Block Diagram .....	132
Figure 5-28 GTP_DRM18K NW Mode Waveform Diagram .....	135
Figure 5-29 GTP_DRM18K TW Mode Waveform Diagram .....	135
Figure 5-30 GTP_DRM18K RBW Mode Waveform Diagram .....	136
Figure 5-31 GTP_DRM18K SP Interface Diagram .....	137
Figure 5-32 GTP_DRM18K Register Mode Waveform Diagram .....	137
Figure 5-33 GTP_FIFO18K Structure Block Diagram .....	144
Figure 5-34 GTP_FIFO18K Waveform Diagram for Writing to Empty FIFO Mode .....	146
Figure 5-35 GTP_FIFO18K Waveform Diagram for Writing to Almost Full FIFO Mode .....	146
Figure 5-36 GTP_FIFO18K Waveform Diagram for Full FIFO Readout Mode .....	147
Figure 5-37 GTP_FIFO18K Waveform Diagram for Reading from Almost Empty FIFO Mode .....	147
Figure 5-38 GTP_FIFO18K Waveform Diagram for FIFO Write Packet Loss Mode .....	148
Figure 5-39 GTP_FIFO18K Waveform Diagram for FIFO Read Resend Mode .....	148
Figure 5-40 GTP_FIFO18K FIFO Depth Expansion Diagram .....	151
Figure 5-41 GTP_FIFO18K FIFO Width Expansion Diagram .....	152

Figure 6-1 GTP_INBUF.....	154
Figure 6-2 GTP_INBUFDS.....	155
Figure 6-3 GTP_INBUFG .....	157
Figure 6-4 GTP_INBUFGDS .....	158
Figure 6-5 GTP_IOBUF.....	159
Figure 6-6 GTP_IOBUFCO .....	161
Figure 6-7 GTP_IOBUFDS .....	162
Figure 6-8 GTP_OUTBUF .....	164
Figure 6-9 GTP_OUTBUFCO .....	165
Figure 6-10 GTP_OUTBUFDS .....	166
Figure 6-11 GTP_OUTBUFT .....	168
Figure 6-12 GTP_OUTBUFTCO .....	169
Figure 6-13 GTP_OUTBUFTDS .....	170
Figure 6-14 GTP_IODELAY.....	172
Figure 6-15 GTP_IOBUF_RX_MIPI .....	173
Figure 6-16 GTP_IOBUF_TX_MIPI .....	176
Figure 6-17 GTP_ISERDES Structure Block Diagram.....	178
Figure 6-18 GTP_OSERDES Structure Block Diagram .....	180
Figure 7-1 GTP_CLKBUFGCE Waveform Diagram (DEFAULT_VALUE=1) .....	185
Figure 7-2 GTP_CLKBUFGCE Waveform Diagram (DEFAULT_VALUE=0) .....	185
Figure 7-3 Timing Diagram for GTP_CLKBUFGMUX with Parameter TRIGGER_MODE="NORMAL" .....	187
Figure 7-4 Timing Diagram for GTP_CLKBUFGMUX with Parameter TRIGGER_MODE="NEGEDGE" .....	188
Figure 7-5 GTP_CLKBUFR Waveform Diagram .....	189
Figure 7-6 Timing Diagram of GTP_IOCLKBUF with GATEN="FALSE" .....	190
Figure 7-7Timing Diagram of GTP_IOCLKBUF with GATEN="TRUE" .....	190
Figure 7-8 GTP_CRYSTAL Waveform Diagram .....	197
Figure 7-9 GTP_DLL Waveform Diagram .....	216
Figure 8-1 GTP_IPAL_E1 Structure Block Diagram .....	218
Figure 8-2Read/Write Timing for GTP_IPAL_E1 Internal Slave Parallel Interface .....	220
Figure 8-3 Read UID Timing.....	224
Figure 8-4 Extended UID Bit Width.....	224
Figure 9-1 GTP_DDC_E1 Structure Block Diagram .....	226
Figure 9-2 GTP_IDDR_E2 Structure Block Diagram.....	253
Figure 9-3 GTP_ODDR_E2 Structure Block Diagram .....	255
Figure 10-1 GTP_RES_CAL Calibration Mode Timing .....	260
Figure 10-2 GTP_APM_E1 Structure Block Diagram.....	261
Figure 10-3 Application Diagram of GTP_APM_E1 Multiplication Mode .....	264
Figure 10-4 Application Diagram of Pre-addition Multiplication Mode .....	265
Figure 10-5 Typical Timing Diagram for Multiplication Mode.....	265

---

Figure 10-6 Application Diagram of GTP_APM_E1 General Multiply-Add Mode .....	266
Figure 10-7 Typical Timing Diagram for General Multiply-Add Mode.....	266
Figure 10-8 Application Diagram of GTP_APM_E1 Multiply-Accumulate Mode .....	267
Figure 10-9 Application Diagram of Multiply-Accumulate Mode with Pre-Addition .....	267
Figure 10-10 Typical Timing Diagram for Multiply-Accumulate Mode .....	268
Figure 10-11 Systolic FIR Functional Diagram.....	268
Figure 10-12 GTP_HSST_E2 Structure Block Diagram.....	271
Figure 10-13 GTP_ADC_E1 Structure Block Diagram .....	355
Figure 10-14 GTP_ADC_E1 Structure Block Diagram .....	361

## Chapter 1 GTP Classification

---

This section categorizes the GTP primitives described in this document by function into nine categories: flip-flop, latch, LUT-related, ROM and RAM-related, IO-related, clock-related, configuration-related, DDR-related, and other. See the table below for specific classifications.

Table 1-1 Logos Family GTP Classification

GTP Name	GTP Description
<b>Flip-flop</b>	
GTP_DFF	Flip-flop
GTP_DFF_C	Asynchronous clear flip-flop
GTP_DFF_E	Flip-flop with enable
GTP_DFF_CE	Asynchronous clear flip-flop with enable
GTP_DFF_P	Asynchronously set flip-flop
GTP_DFF_PE	Asynchronously set flip-flop with enable
GTP_DFF_R	Synchronous clear flip-flop
GTP_DFF_RE	Synchronous clear flip-flop with enable
GTP_DFF_S	Synchronously set flip-flop
GTP_DFF_SE	Synchronously set flip-flop with enable
<b>Latch</b>	
GTP_DLATCH	Latch
GTP_DLATCH_E	Latch with enable
GTP_DLATCH_C	Asynchronous clear latch
GTP_DLATCH_CE	Asynchronous clear latch with enable
GTP_DLATCH_P	Asynchronously set latch
GTP_DLATCH_PE	Asynchronously set latch with enable
<b>LUT-related</b>	
GTP_ONE	GTP_ONE outputs a high level
GTP_ZERO	GTP_ZERO outputs a low level
GTP_BUF	General-purpose 1bit BUF
GTP_INV	Inverter
GTP_LUT1	1-input LUT
GTP_LUT2	2-input LUT
GTP_LUT3	3-input LUT
GTP_LUT4	4-input LUT
GTP_LUT5	5-input LUT
GTP_LUT5CARRY	5-input LUT with carry chain
GTP_LUT6	6-input LUT
GTP_LUT7	7-input LUT

GTP Name	GTP Description
GTP_LUT8	8-input LUT
GTP_LUTMUX4	4:1 MUX selector
GTP_MUX2LUT6	GTP_LUT6 dedicated 2-input selector
GTP_MUX2LUT7	GTP_LUT7 dedicated 2-input selector
GTP_MUX2LUT8	GTP_LUT8 dedicated 2-input selector
GTP_LUT5M	5-input Look-up table, with different INIT parameter values corresponding to different functions of LUT5M
<b>ROM and RAM-related</b>	
GTP_ROM32X1	ROM with a 32-bit address and 1-bit data
GTP_ROM64X1	ROM with a 64-bit address and 1-bit data
GTP_ROM128X1	ROM with a 128-bit address and 1-bit data
GTP_ROM256X1	ROM with a 256-bit address and 1-bit data
GTP_RAM16X4SP	Single-port RAM with a 16-bit address and 4-bit data
GTP_RAM16X4DP	Simple dual-port RAM with a 16-bit address and 4-bit data
GTP_RAM16X1SP	Single-port RAM with a 16-bit address and 1-bit data
GTP_RAM16X1DP	Simple dual-port RAM with a 16-bit address and 1-bit data
GTP_RAM32X1SP	Single-port RAM with a 32-bit address and 1-bit data
GTP_RAM32X1DP	Single dual-port RAM with a 32-bit address and 1-bit data
GTP_DRM9K	9K bits dedicated RAM module (DRM)
GTP_DRM18K	18K bits dedicated RAM module (DRM)
GTP_FIFO18K	18K bits FIFO
<b>IO-related</b>	
GTP_INBUF	Single-ended input BUFFER
GTP_INBUFDSDS	Differential input BUFFER
GTP_INBUFG	Single-ended input clock BUFFER
GTP_INBUFGDS	Differential input clock BUFFER
GTP_Iobuf	Bidirectional input/output BUFFER
GTP_Iobufco	Bidirectional pseudo-differential input/output BUFFER
GTP_Iobufds	Bidirectional true differential input/output BUFFER
GTP_OUTBUF	Single-ended output BUFFER
GTP_OUTBUFCO	Pseudo-differential output BUFFER
GTP_OUTBUFDSDS	True differential output BUFFER
GTP_OUTBUFT	Tri-state output BUFFER
GTP_OUTBUFTCO	Tri-state pseudo-differential output BUFFER
GTP_OUTBUFTDS	Tri-state true differential output BUFFER
GTP_Iodelay	Data Delay Unit, supporting dynamic or static delay control
GTP_Iobuf_rx_miPI	Support MIPI DPHY high-speed input, and single-ended input/output under Low-Power (LP) mode
GTP_Iobuf_tx_miPI	Support MIPI DPHY tri-state output, including input/output in Low-Power (LP) mode and differential output in High-Speed (HS) mode
GTP_ISERDES	Input data processing module

GTP Name	GTP Description
GTP_OSERDES	Data output conversion module
<b>Clock-related</b>	
GTP_CLKBUFG	Global clock BUFFER
GTP_CLKBUFGCE	Global clock BUFFER with port enabled
GTP_CLKBUFGMUX	GTP for dynamic switching between two global clock Inputs
GTP_CLKBUFR	Regional clock BUFFER
GTP_IOCLKBUF	Like the gate function, this GTP can be used to turn off or on the IO clock
GTP_IOCLKDIV	Clock divide GTP
GTP_IOCLKDELAY	Input/output clock delay unit, supporting dynamic or static delay control
GTP_OSC_E1	Provide user-configurable output clock
GTP_OSC_E3	Provide user-configurable output clock
GTP_CRYSTAL	The output clock of GTP_CRYSTAL can serve as a global clock for driving logic cells or as an IO clock
GTP_PLL_E1	PLL is the core subsystem of FPGA that provides clock resources, mainly used for implementing frequency division, frequency multiplication, and phase adjustment functions
GTP_PLL_E3	PLL is the core subsystem of FPGA that provides clock resources, mainly used for implementing frequency division, frequency multiplication, and phase adjustment functions
GTP_DLL	Dynamically lock the frequency of the input reference clock and output an equivalent number of delay steps
<b>Configuration-related</b>	
GTP_EFUSECODE	This GTP is used to read EFUSECODE and parallelly outputs 32-bit data stored in efuse to the user
GTP_IPAL_E1	Used for CRC checking or SEU checking of readback data
GTP_SCANCHAIN_E1	Read the values of user data registers through the JTAG interface
GTP_FLASHIF	This module is the clock interface between the user and flash, which can be either an internal or external flash
GTP_UDID	This GTP is used to read UDID CODE value
<b>DDR-related</b>	
GTP_DDC_E1	Generate the write clock for DDR memory
GTP_DDRC	Mainly functions as a memory controller in the DDR system
GTP_DDRPHY	Work with DDRC to complete the calibration and read-write for DDR
GTP_IDDR_E2	GTP_IDDR_E2 supports edge-aligned pipeline DDR1TO2 rate conversion
GTP_ODDR_E2	GTP_ODDR_E2 supports edge-aligned DDR2TO1 rate conversion
<b>Other</b>	
GTP_GRS	This GTP is used to control the global reset signal
GTP_START_E1	This GTP describes the process of releasing the global clock signal for the wake-up operation.
GTP_RES_CAL	Used to describe the input/output characteristics of IO impedance calibration in user mode
GTP_APM_E1	Arithmetic logic unit
GTP_HSST_E2	High-speed serial transceiver module
GTP_ADC_E1	Used to implement ADC functions
GTP_PCIEGEN2	High-speed serial interface module, compatible with PCIe 2.1 protocol

## Chapter 2 Usage Instructions for Trigger GTPs

### 2.1 General Introduction

GTP\_DFF family GTP mainly implements flip-flop-related functions.

### 2.2 Usage Instructions for GTP\_DFF

#### 2.2.1 Supported Devices

Table 2-1 Device Models That Support GTP\_DFF

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 2.2.2 Description of Functionality

GTP\_DFF is a D flip-flop with a data input D and a data output Q. It triggers and transfers the input signal to the output on the rising edge of the clock. The output is low upon power-up.

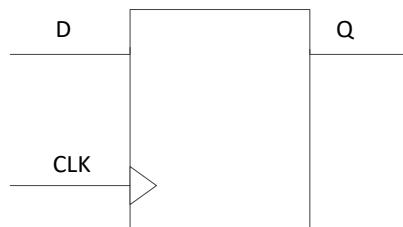


Figure 2-1 GTP\_DFF

#### 2.2.3 Port Description

Table 2-2 GTP\_DFF Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger

## 2.2.4 Parameter Description

Table 2-3 GTP\_DFF Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

## 2.2.5 Instantiation template

```
GTP_DFF #
(
    .GRS_EN("TRUE"),
    .INIT    (1'b0)
)GTP_DFF(
    .CLK (CLK),
    .D (D),
    .Q (Q)
);
```

## 2.3 Usage Instructions for GTP\_DFF\_C

### 2.3.1 Supported Devices

Table 2-4 Device Models That Support GTP\_DFF\_C

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 2.3.2 Description of Functionality

GTP\_DFF\_C is a D flip-flop with a data input D and a data output Q. It triggers and transfers the input signal to the output on the rising edge of the clock. The output is low in power-up state.

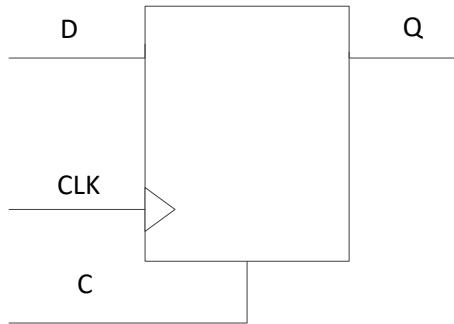


Figure 2-2 GTP\_DFF\_C

### 2.3.3 Port Description

Table 2-5 GTP\_DFF\_C Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
C	Input	Asynchronous clear signal

### 2.3.4 Parameter Description

Table 2-6 GTP\_DFF\_C Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 2.3.5 Instantiation template

```
GTP_DFF_C #
```

```
(
```

```
.GRS_EN("TRUE"),
```

```
.INIT (1'b0)
```

```
)GTP_DFF_C(
```

```
.CLK (CLK),
```

```
.D (D),
```

.C (C),  
.Q (Q)  
);

## 2.4 Usage Instructions for GTP\_DFF\_E

### 2.4.1 Supported Devices

Table 2-7 Device Models That Support GTP\_DFF\_E

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 2.4.2 Description of Functionality

GTP\_DFF\_E is a D flip-flop with a data input D and a data output Q. It triggers and transfers the input signal to the output only on the rising edge of the clock when CE is high.

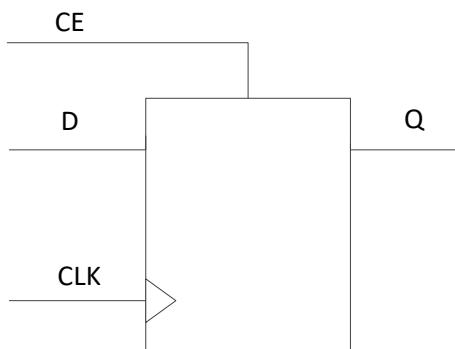


Figure 2-3 GTP\_DFF\_E

### 2.4.3 Port Description

Table 2-8 GTP\_DFF\_E Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
CE	Input	Active-high enable signal

#### 2.4.4 Parameter Description

Table 2-9 GTP\_DFF\_E Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

#### 2.4.5 Instantiation template

```
GTP_DFF_E #
(
    .GRS_EN("TRUE"),
    .INIT    (1'b0)
)GTP_DFF_E(
    .CLK (CLK),
    .CE  (CE),
    .D   (D),
    .Q   (Q)
);
```

### 2.5 Usage Instructions for GTP\_DFF\_CE

#### 2.5.1 Supported Devices

Table 2-10 Device Models That Support GTP\_DFF\_CE

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 2.5.2 Description of Functionality

GTP\_DFF\_CE is a D flip-flop with a data input D and a data output Q. It triggers and transfers the input signal to the output only on the rising edge of the clock when CE is high. Asynchronous clear.

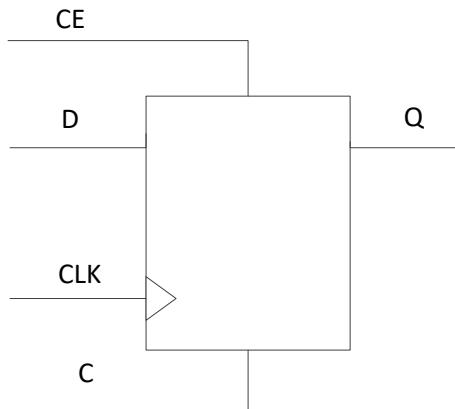


Figure 2-4 GTP\_DFF\_CE

### 2.5.3 Port Description

Table 2-11 GTP\_DFF\_CE Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
C	Input	Asynchronous clear signal
CE	Input	Active-high enable signal

### 2.5.4 Instantiation template

```
GTP_DFF_CE(
    .CLK(CLK),
    .C(C),
    .CE(CE),
    .D(D),
    .Q(Q)
);
```

## 2.6 Usage Instructions for GTP\_DFF\_P

### 2.6.1 Supported Devices

Table 2-12 Device Models That Support GTP\_DFF\_P

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

## 2.6.2 Description of Functionality

GTP\_DFF\_P is a D flip-flop with a data input D and a data output Q. It triggers and transfers the input signal to the output on the rising edge of the clock. It is asynchronously set.

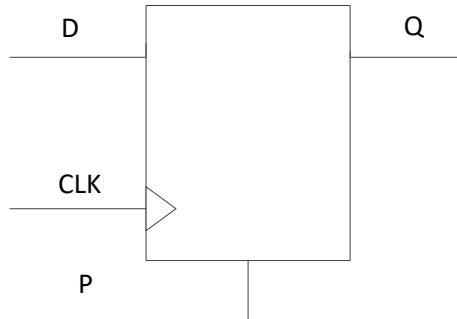


Figure 2-5 GTP\_DFF\_P

## 2.6.3 Port Description

Table 2-13 GTP\_DFF\_P Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
P	Input	Asynchronously set signal

## 2.6.4 Parameter Description

Table 2-14 GTP\_DFF\_P Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

## 2.6.5 Instantiation template

```

GTP_DFF_P #
(
    .GRS_EN("TRUE"),
    .INIT    (1'b0)
)

```

.CLK (CLK),  
.P(P),  
.D (D),  
.Q (Q)  
);

## 2.7 Usage Instructions for GTP\_DFF\_PE

### 2.7.1 Supported Devices

Table 2-15 Device Models That Support GTP\_DFF\_PE

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 2.7.2 Description of Functionality

GTP\_DFF\_PE is a D flip-flop with a data input D and a data output Q. It is triggered on the rising edge of the clock when the CE is high, transferring the input signal to the output. It is asynchronously set.

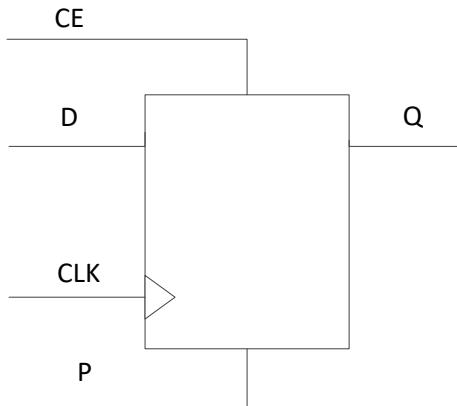


Figure 2-6 GTP\_DFF\_PE

### 2.7.3 Port Description

Table 2-16 GTP\_DFF\_PE Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
P	Input	Asynchronously set signal
CE	Input	Active-high enable signal

### 2.7.4 Parameter Description

Table 2-17 GTP\_DFF\_PE Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 2.7.5 Instantiation template

```
GTP_DFF_PE #
(
    .GRS_EN("TRUE"),
    .INIT    (1'b0),
    )GTP_DFF_PE(
        .CLK (CLK),
        .P (P),
        .CE (CE),
        .D (D),
        .Q (Q)
    );
```

## 2.8 Usage Instructions for GTP\_DFF\_R

### 2.8.1 Supported Devices

Table 2-18 Device Models That Support GTP\_DFF\_R

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 2.8.2 Description of Functionality

GTP\_DFF\_R is a D flip-flop with a data input D and a data output Q. It triggers and transfers the input signal to the output on the rising edge of the clock. Synchronous clear.

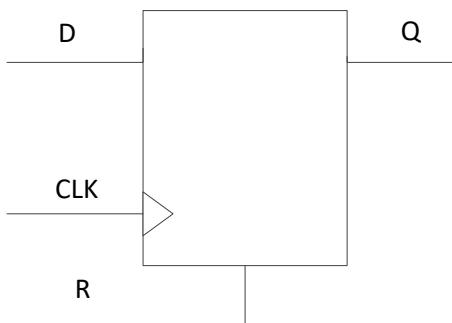


Figure 2-7 GTP\_DFF\_R

### 2.8.3 Port Description

Table 2-19 GTP\_DFF\_R Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
R	Input	Synchronization clear signal

### 2.8.4 Parameter Description

Table 2-20 GTP\_DFF\_R Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

## 2.8.5 Instantiation template

```
GTP_DFF_R #
(
    .GRS_EN("TRUE"),
    .INIT  (1'b0)
)GTP_DFF_R(
    .CLK (CLK),
    .R (R),
    .D (D),
    .Q (Q)
);
```

## 2.9 Usage Instructions for GTP\_DFF\_RE

### 2.9.1 Supported Devices

Table 2-21 Device Models That Support GTP\_DFF\_RE

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 2.9.2 Description of Functionality

GTP\_DFF\_RE is a D flip-flop with a data input D and a data output Q. It is triggered on the rising edge of the clock when the CE is high, transferring the input signal to the output. Synchronous clear.

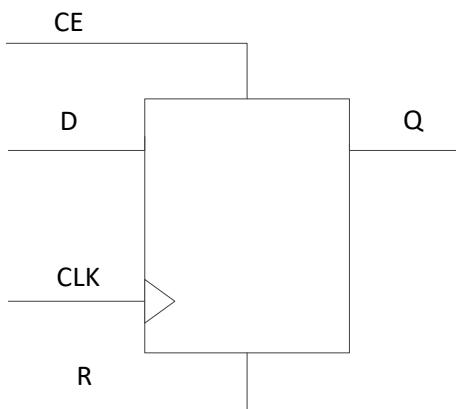


Figure 2-8 GTP\_DFF\_RE

### 2.9.3 Port Description

Table 2-22 GTP\_DFF\_RE Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
R	Input	Synchronization clear signal
CE	Input	Active-high enable signal

### 2.9.4 Parameter Description

Table 2-23 GTP\_DFF\_RE Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 2.9.5 Instantiation template

```
GTP_DFF_RE #
(
    .GRS_EN("TRUE"),
    .INIT    (1'b0)
)
GTP_DFF_RE(
    .CLK (CLK),
    .R   (R),
    .CE  (CE),
    .D   (D),
    .Q   (Q)
);
```

## 2.10 Usage Instructions for GTP\_DFF\_S

### 2.10.1 Supported Devices

Table 2-24 Device Models That Support GTP\_DFF\_S

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 2.10.2 Description of Functionality

GTP\_DFF\_S is a D flip-flop with a data input D and a data output Q. It triggers and transfers the input signal to the output on the rising edge of the clock. It is synchronously set.

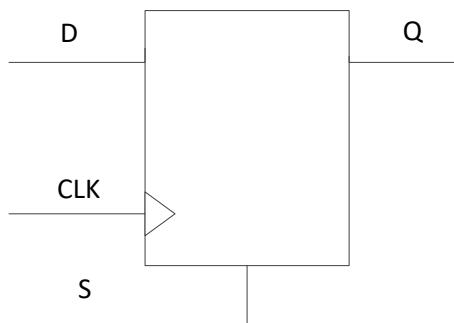


Figure 2-9 GTP\_DFF\_S

### 2.10.3 Port Description

Table 2-25 GTP\_DFF\_S Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
S	Input	Synchronously set signal

### 2.10.4 Parameter Description

Table 2-26 GTP\_DFF\_S Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

## 2.10.5 Instantiation template

```
GTP_DFF_S #
(
    .GRS_EN("TRUE"),
    .INIT  (1'b0)
)GTP_DFF_S(
    .CLK (CLK),
    .S (S),
    .D (D),
    .Q (Q)
);
```

## 2.11 Usage Instructions for GTP\_DFF\_SE

### 2.11.1 Supported Devices

Table 2-27 Device Models That Support GTP\_DFF\_SE

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 2.11.2 Description of Functionality

GTP\_DFF\_SE is a D flip-flop with a data input D and a data output Q. It triggers and transfers the input signal to the output only on the rising edge of the clock when CE is high. It is synchronously set.

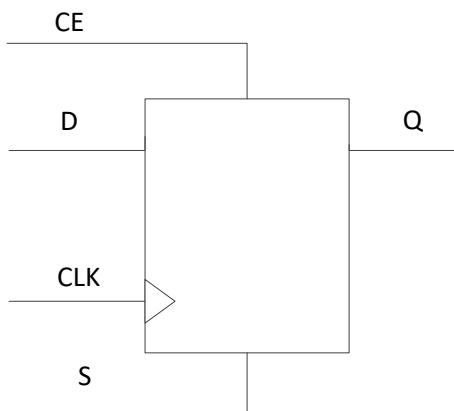


Figure 2-10 GTP\_DFF\_SE

### 2.11.3 Port Description

Table 2-28 GTP\_DFF\_SE Port Description

Port	Orientation	Function Description
CLK	Input	Input Clock
D	Input	Input signal of trigger
Q	Output	Output signal of trigger
S	Input	Synchronously set signal
CE	Input	Active-high enable signal

### 2.11.4 Paramater Description

Table 2-29 GTP\_DFF\_SE Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 2.11.5 Instantiation template

```
GTP_DFF_SE #
(
    .GRS_EN("TRUE"),
    .INIT    (1'b0)
)GTP_DFF_SE(
    .CLK (CLK),
    .S (S),
    .CE (CE),
    .D (D),
    .Q (Q)
);
```

## Chapter 3 Usage Instructions for Latch GTPs

### 3.1 General Introduction

GTP\_DLATCH family GTP mainly implements flip-flop-related functions.

### 3.2 Usage Instructions for GTP\_DLATCH

#### 3.2.1 Supported Devices

Table 3-1 Device Models That Support GTP\_DLATCH

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 3.2.2 Description of Functionality

GTP\_DLATCH is a latch with D as data input and Q as data output. It can send the input signal to the output when G is high.

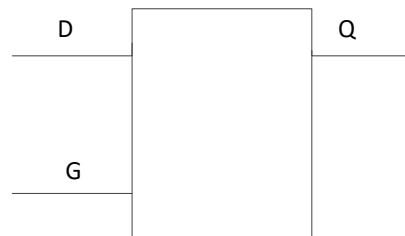


Diagram 3-1 GTP\_DLATCH Diagram

#### 3.2.3 Port Description

Table 3-2 GTP\_DLATCH Port Description

Port	Orientation	Function Description
G	Input	Level input
D	Input	Latch input signal
Q	Output	Latch output signal

### 3.2.4 Parameter Description

Table 3-3 GTP\_DLATCH Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 3.2.5 Instantiation template

```
GTP_DLATCH#(
    .INIT(1'b0),
    .GRS_EN("TURE")// "TURE"; "FALSE"
)
GTP_DLATCH (
    .Q (q),
    .D (d),
    .G (g)
);
```

## 3.3 Usage Instructions for GTP\_DLATCH\_E

### 3.3.1 Supported Devices

Table 3-4 Device Models That Support GTP\_DLATCH\_E

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 3.3.2 Description of Functionality

GTP\_DLATCH\_E is a latch with D as data input and Q as data output. It can send the input signal to the output when G and GE are both high.

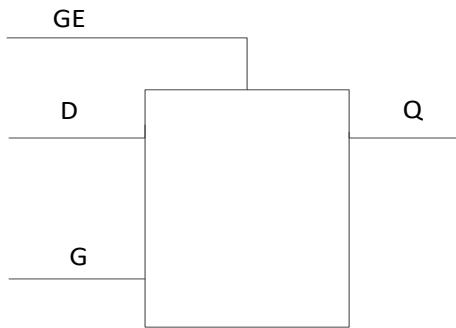


Figure 3-2 GTP\_DLATCH\_E Diagram

### 3.3.3 Port Description

Table 3-5 GTP\_DLATCH\_E Port Description

Port	Orientation	Function Description
G	Input	Input level
D	Input	Latch input signal
Q	Output	Latch output signal
GE	Input	Active-high enable signal

### 3.3.4 Parameter Description

Table 3-6 GTP\_DLATCH\_E Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 3.3.5 Instantiation template

```

GTP_DLATCH_E#(
    .INIT(1'b0),
    .GRS_EN("TURE")// "TURE"; "FALSE"
)
GTP_DLATCH_E_inst (
    .Q(q),
    .D(d),
    
```

.G (g),  
.GE (en)  
);

### 3.4 Usage Instructions for GTP\_DLATCH\_C

#### 3.4.1 Supported Devices

Table 3-7 GTP\_DLATCH\_C

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 3.4.2 Description of Functionality

GTP\_DLATCH\_C is a latch with D as data input and Q as data output. It can send the input signal to the output when G is high. Asynchronous clear.

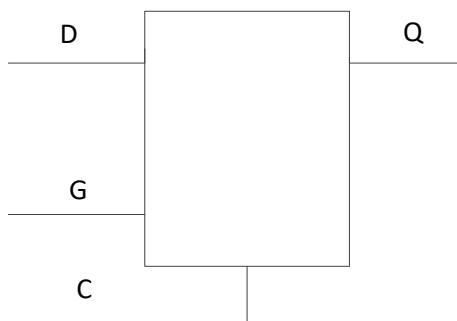


Figure 3-3 GTP\_DLATCH\_C Diagram

#### 3.4.3 Port Description

Table 3-8 GTP\_DLATCH\_C Port Description

Port	Orientation	Function Description
G	Input	Input level
D	Input	Latch input signal
Q	Output	Latch output signal
C	Input	Asynchronous clear signal

### 3.4.4 Parameter Description

Table 3-9 GTP\_DLATCH\_C Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 3.4.5 Instantiation template

```
GTP_DLATCH_C#(
    .INIT(1'b0),
    .GRS_EN("TURE")//"TURE"; "FALSE"
)
GTP_DLATCH_C_inst (
    .Q(q),
    .D(d),
    .G(g),
    .C(c)
);
```

## 3.5 Usage Instructions for GTP\_DLATCH\_CE

### 3.5.1 Supported Devices

Table 3-10 Device Models That Support GTP\_DLATCH

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 3.5.2 Description of Functionality

GTP\_DLATCH\_CE is a latch with D as data input and Q as data output. It can send the input signal to the output when G and GE are both high. Asynchronous clear.

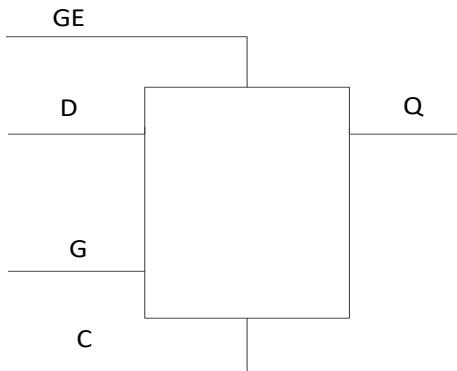


Figure 3-4 GTP\_DLATCH\_CE Diagram

### 3.5.3 Port Description

Table 3-11 GTP\_DLATCH\_CE Port Description

Port	Orientation	Function Description
G	Input	Input level
D	Input	Latch input signal
Q	Output	Latch output signal
C	Input	Asynchronous clear signal
GE	Input	Active-high enable signal

### 3.5.4 Parameter Description

Table 3-12 GTP\_DLATCH\_C Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 3.5.5 Instantiation template

```

GTP_DLATCH_C#(
    .INIT(1'b0),
    .GRS_EN("TURE")//TURE; "FALSE"
)
GTP_DLATCH_C_inst (
    .Q(q),

```

.D (d),  
.G (g),  
.C (c)  
);

### 3.6 Usage Instructions for GTP\_DLATCH\_P

#### 3.6.1 Supported Devices

Table 3-13 Device Models That Support GTP\_DLATCH\_P

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 3.6.2 Description of Functionality

GTP\_DLATCH\_P is a latch with D as data input and Q as data output. It can send the input signal to the output when G is high. It is asynchronously set.

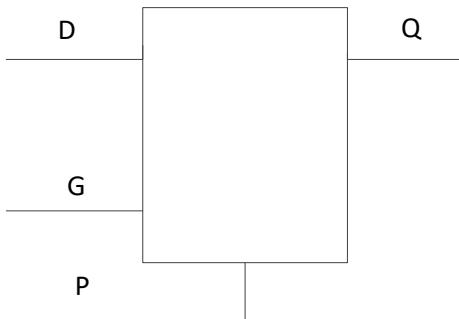


Figure 3-5 GTP\_DLATCH\_P Diagram

#### 3.6.3 Port Description

Table 3-14 GTP\_DLATCH\_P Port Description

Port	Orientation	Function Description
G	Input	Input level
D	Input	Latch input signal
Q	Output	Latch output signal
P	Input	Asynchronously set signal

### 3.6.4 Parameter Description

Table 3-15 GTP\_DLATCH\_P Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 3.6.5 Instantiation template

```
GTP_DLATCH_P#(
    .INIT(1'b0),
    .GRS_EN("TURE")// "TURE"; "FALSE"
)
GTP_DLATCH_P_inst (
    .Q(q),
    .D(d),
    .G(g),
    .P(p)
);
```

## 3.7 Usage Instructions for GTP\_DLATCH\_PE

### 3.7.1 Supported Devices

Table 3-16 Device Models That Support GTP\_DLATCH\_PE

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 3.7.2 Description of Functionality

GTP\_DLATCH\_PE is a latch with D as data input and Q as data output. It can send the input signal to the output when G and GE are both high. It is asynchronously set.

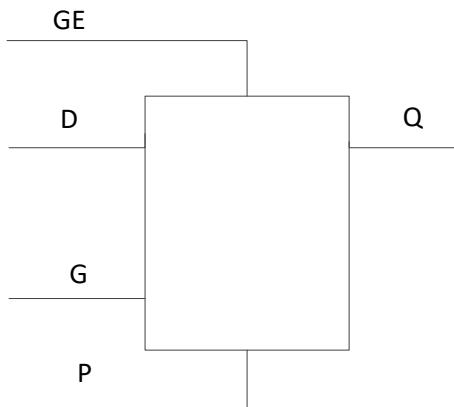


Figure 3-6 GTP\_DLATCH\_PE Diagram

### 3.7.3 Port Description

Table 3-17 GTP\_DLATCH\_PE Port Description

Port	Orientation	Function Description
G	Input	Input level
D	Input	Latch input signal
Q	Output	Latch output signal
P	Input	Asynchronously set signal
GE	Input	Active-high enable signal

### 3.7.4 Parameter Description

Table 3-18 GTP\_DLATCH\_PE Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
GRS_EN	<string>	"TRUE", "FALSE"	"TRUE"	Global reset signal enable
INIT	<binary>	1'b0, 1'b1	1'b0	Initial value, asynchronously triggered by the global reset signal. Note: Use the default value. This parameter cannot be configured.

### 3.7.5 Instantiation template

```

GTP_DLATCH_PE#(
    .INIT(1'b0),
    .GRS_EN("TURE")// "TURE"; "FALSE"
)
GTP_DLATCH_PE_inst (
    .Q(q),

```

.D (d),

.G (g),

.P (p),

.GE (en)

## Chapter 4 Usage Instructions for LUT-related GTP

### 4.1 Usage Instructions for GTP\_ONE

#### 4.1.1 Supported Devices

Table 4-1 Device Models That Support GTP\_ONE

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 4.1.2 Description of Functionality

GTP\_ONE outputs a high level signal. This is shown in the following figure:

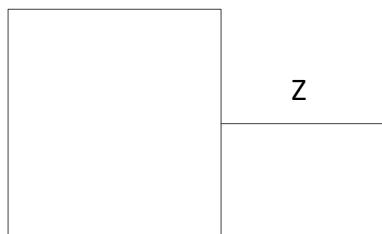


Figure 4-1 GTP\_ONE Structure Diagram

#### 4.1.3 Port Description

Table 4-2 GTP\_ONE Port Description

Port	Orientation	Function Description
Z	Output	High-level output

#### 4.1.4 Instantiation template

```
GTP_ONE GTP_ONE_inst (
    .Z (Z)
);
```

## 4.2 Usage Instructions for GTP\_ZERO

### 4.2.1 Supported Devices

Table 4-3 Device Models That Support GTP\_ZERO

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.2.2 Description of Functionality

GTP\_ZERO outputs a low level signal. The structure is shown below:

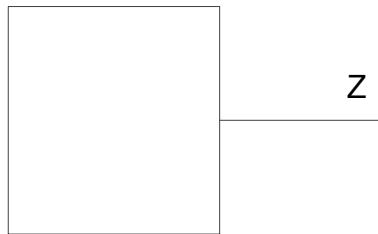


Figure 4-2 GTP\_ZERO Structure Diagram

### 4.2.3 Port Description

Table 4-4 GTP\_ZERO Port Description

Port	Orientation	Function Description
Z	Output	Low-level output

### 4.2.4 Instantiation template

```

GTP_ZERO GTP_ZERO_inst (
    .Z  (z)
);
  
```

## 4.3 Usage Instructions for GTP\_BUF

### 4.3.1 Supported Devices

Table 4-5 Device Models That Support GTP\_BUF

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.3.2 Description of Functionality

GTP\_BUF is a 1-bit BUFFER, and its structure is shown as follows:

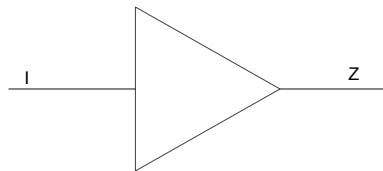


Figure 4-3 GTP\_BUF Structure Diagram

### 4.3.3 Port Description

Table 4-6 GTP\_BUF Port Description

Port	Orientation	Function Description
Z	Output	Output signal
I	Input	Input Signal

### 4.3.4 Instantiation template

```
GTP_BUF GTP_BUF_inst (
    .I    (I),
    .Z    (Z)
);
```

### 4.3.5 Detailed Functional Description

GTP\_BUF is not a mandatory GTP, which will be optimized away by the software after MAP.

## 4.4 Usage Instructions for GTP\_INV

### 4.4.1 Supported Devices

Table 4-7 Device Models That Support GTP\_INV

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.4.2 Description of Functionality

GTP\_INV is a inverter. The structure is shown below:

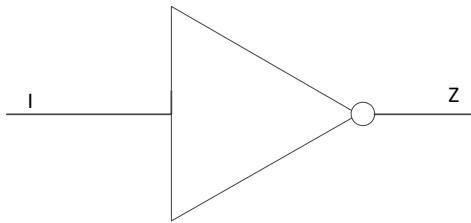


Figure 4-4 GTP\_INV Structure Diagram

#### 4.4.3 Port Description

Table 4-8 GTP\_INV Port Description

Port	Orientation	Function Description
Z	Output	Inverted output signal
I	Input	Input Signal

#### 4.4.4 Instantiation template

```
GTP_INV GTP_INV_inst (
    .I  (i),
    .Z  (z)
);
```

### 4.5 Usage Instructions for GTP\_LUT1

#### 4.5.1 Supported Devices

Table 4-9 Device Models That Support GTP\_LUT1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 4.5.2 Description of Functionality

GTP\_LUT1 is a lookup table with 1-bit input and 1-bit output. The structure block diagram is shown below.

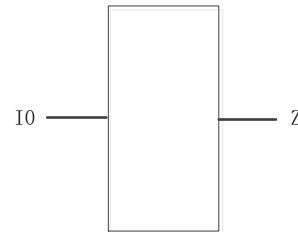


Figure 4-5 GTP\_LUT1 Structure Diagram

#### 4.5.3 Port Description

Table 4-10 GTP\_LUT1 Port Description

<b>Port</b>	<b>Orientation</b>	<b>Function Description</b>
Z	Output	Output signal
I0	Input	Input Signal

Truth Table:

Table 4-11 GTP\_LUT1 Truth Table

<b>Input</b>	<b>Output</b>
I0	Z
0	INIT[0]
1	INIT[1]

#### 4.5.4 Paramater Description

Table 4-12 GTP\_LUT1 Parameter Description

<b>Parameter Name</b>	<b>Description</b>	<b>Defaults</b>	<b>Valid Values</b>
INIT	Output Configuration Parameter	2'h0	0~3

#### 4.5.5 Instantiation template

```

GTP_LUT1
#(
    .INIT    (2'h0)
    )GTP_LUT1_inst (
        .Z      (z),
        .I0     (i0)
    );

```

## 4.6 Usage Instructions for GTP\_LUT2

### 4.6.1 Supported Devices

Table 4-13 Device Models That Support GTP\_LUT2

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.6.2 Description of Functionality

GTP\_LUT2 is a dual-input lookup table, with different INIT parameter values corresponding to different functions of LUT2; the corresponding INIT parameter must be specified when used. The structure block diagram is shown below.

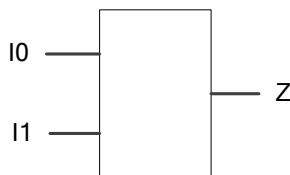


Figure 4-6 GTP\_LUT2 Structure Diagram

### 4.6.3 Port Description

Table 4-14 GTP\_LUT2 Port Description

Port	Orientation	Function Description
Z	Output	Output signal
I0	Input	Input Signal
I1	Input	Input Signal

Truth Table:

Table 4-15 GTP\_LUT2 Truth Table

Input		Output
I1	I0	Z
0	0	INIT[0]
0	1	INIT[1]
1	0	INIT[2]
1	1	INIT[3]

#### 4.6.4 Parameter Description

Table 4-16 GTP\_LUT2 Parameter Description

Parameter Name	Description	Defaults	Valid Values
INIT	Output Configuration Parameter	4'h0	0~15

#### 4.6.5 Instantiation template

```
GTP_LUT2
#(
    .INIT    (4'h0),
    )GTP_LUT2_inst (
        .Z      (z),
        .I0     (i0),
        .I1     (i1)
    );

```

### 4.7 Usage Instructions for GTP\_LUT3

#### 4.7.1 Supported Devices

Table 4-17 Device Models That Support GTP\_LUT3

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 4.7.2 Description of Functionality

GTP\_LUT3 is a triple-input lookup table, with different INIT parameter values corresponding to different functions of LUT3; the corresponding INIT parameter must be specified when used. The structure block diagram is shown below.

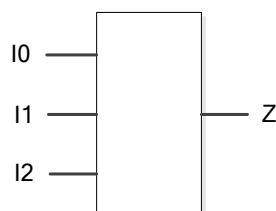


Figure 4-7 GTP\_LUT3 Structure Diagram

### 4.7.3 Port Description

Table 4-18 GTP\_LUT3 Port Description

<b>Port</b>	<b>Orientation</b>	<b>Function Description</b>
Z	Output	Output signal
I0	Input	Input Signal
I1	Input	Input Signal
I2	Input	Input Signal

Truth Table:

Table 4-19 GTP\_LUT3 Truth Table

<b>Input</b>	<b>Output</b>		
I2	I1	I0	Z
0	0	0	INIT[0]
0	0	1	INIT[1]
0	1	0	INIT[2]
0	1	1	INIT[3]
1	0	0	INIT[4]
1	0	1	INIT[5]
1	1	0	INIT[6]
1	1	1	INIT[7]

### 4.7.4 Paramater Description

Table 4-20 GTP\_LUT3 Parameter Description

<b>Parameter Name</b>	<b>Description</b>	<b>Defaults</b>	<b>Valid Values</b>
INIT	Output Configuration Parameter	8'h00	0~255

### 4.7.5 Instantiation template

```

GTP_LUT3
#(
    .INIT      (8'h00)
    )GTP_LUT3_inst (
        .Z          (z),
        .I0         (i0),
        .I1         (i1),
        .I2         (i2)
    )

```

);

## 4.8 Usage Instructions for GTP\_LUT4

### 4.8.1 Supported Devices

Table 4-21 Device Models That Support GTP\_LUT4

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.8.2 Description of Functionality

GTP\_LUT4 is a 4-input lookup table, with different INIT parameter values corresponding to different functions of LUT4; the corresponding INIT parameter must be specified when used. The structure block diagram is shown below.

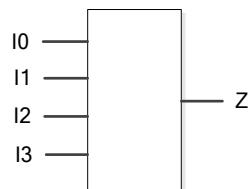


Figure 4-8 GTP\_LUT4 Structure Diagram

### 4.8.3 Port Description

Table 4-22 GTP\_LUT4 Port Description

Port	Orientation	Function Description
Z	Output	Output signal
I0	Input	Input Signal
I1	Input	Input Signal
I2	Input	Input Signal
I3	Input	Input Signal

Truth Table:

Table 4-23 GTP\_LUT4 Truth Table

Input				Output
I3	I2	I1	I0	Z
0	0	0	0	INIT[0]
0	0	0	1	INIT[1]
0	0	1	0	INIT[2]

<b>Input</b>				<b>Output</b>
0	0	1	1	INIT[3]
0	1	0	0	INIT[4]
0	1	0	1	INIT[5]
0	1	1	0	INIT[6]
0	1	1	1	INIT[7]
1	0	0	0	INIT[8]
1	0	0	1	INIT[9]
1	0	1	0	INIT[10]
1	0	1	1	INIT[11]
1	1	0	0	INIT[12]
1	1	0	1	INIT[13]
1	1	1	0	INIT[14]
1	1	1	1	INIT[15]

#### 4.8.4 Paramater Description

Table 4-24 GTP\_LUT4 Parameter Description

Parameter Name	Description	Defaults	Valid Values
INIT	Output Configuration Parameter	16'h0000	0~2^16-1

#### 4.8.5 Instantiation template

```

GTP_LUT4
#(
    .INIT  (16'h0000)
)GTP_LUT4_inst (
    .Z      (z),
    .I0     (i0),
    .I1     (i1),
    .I2     (i2),
    .I3     (i3)
);
    
```

## 4.9 Usage Instructions for GTP\_LUT5

### 4.9.1 Supported Devices

Table 4-25 Device Models That Support GTP\_LUT5

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.9.2 Description of Functionality

GTP\_LUT5 is a 5-input lookup table, with different INIT parameter values corresponding to different functions of LUT5; the corresponding INIT parameter must be specified when used. The structure block diagram is shown below.

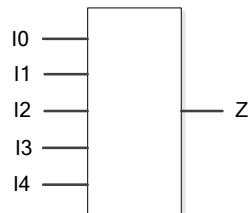


Figure 4-9 GTP\_LUT5 Structure Diagram

### 4.9.3 Port Description

Table 4-26 GTP\_LUT5 Port Description

Port	Orientation	Function Description
Z	Output	Output signal
I0	Input	Input Signal
I1	Input	Input Signal
I2	Input	Input Signal
I3	Input	Input Signal
I4	Input	Input Signal

Truth Table:

Table 4-27 GTP\_LUT5 Truth Table

Input					Output
I4	I3	I2	I1	I0	Z
0	0	0	0	0	INIT[0]
0	0	0	0	1	INIT[1]
0	0	0	1	0	INIT[2]
0	0	0	1	1	INIT[3]

<b>Input</b>					<b>Output</b>
0	0	1	0	0	INIT[4]
0	0	1	0	1	INIT[5]
0	0	1	1	0	INIT[6]
0	0	1	1	1	INIT[7]
0	1	0	0	0	INIT[8]
0	1	0	0	1	INIT[9]
0	1	0	1	0	INIT[10]
0	1	0	1	1	INIT[11]
0	1	1	0	0	INIT[12]
0	1	1	0	1	INIT[13]
0	1	1	1	0	INIT[14]
1	0	0	0	0	INIT[16]
1	0	0	0	1	INIT[17]
1	0	0	1	0	INIT[18]
1	0	0	1	1	INIT[19]
1	0	1	0	0	INIT[20]
1	0	1	0	1	INIT[21]
1	0	1	1	0	INIT[22]
1	0	1	1	1	INIT[23]
1	1	0	0	0	INIT[24]
1	1	0	0	1	INIT[25]
1	1	0	1	0	INIT[26]
1	1	0	1	1	INIT[27]
1	1	1	0	0	INIT[28]
1	1	1	0	1	INIT[29]
1	1	1	1	0	INIT[30]
1	1	1	1	1	INIT[31]

#### 4.9.4 Paramater Description

Table 4-28 GTP\_LUT5 Parameter Description

Parameter	Description	Defaults	Setting Value
INIT	Output Configuration Parameter	32'h0000_0000	0 ~ 2^32-1

#### 4.9.5 Instantiation template

GTP\_LUT5

```

#(
.INIT (32'h0000_0000)
)GTP_LUT5_inst (
.Z      (z),
.I0     (i0),
.I1     (i1),
.I2     (i2),
.I3     (i3),
.I4     (i4)
);

```

## 4.10 Usage Instructions for GTP\_LUT5CARRY

### 4.10.1 Supported Devices

Table 4-29 Device Models That Support GTP\_LUT5CARRY

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.10.2 Description of Functionality

GTP\_LUT5CARRY is a fast carry logic, typically used for counter counting, subtraction, address logic, etc. The structure block diagram is shown below.

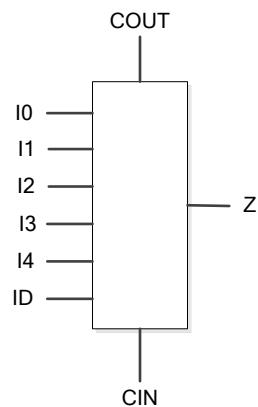


Figure 4-10 GTP\_LUT5CARRY Structure Diagram

#### 4.10.3 Port Description

Table 4-30 GTP\_LUT5CARRY Port Description

<b>Port</b>	<b>Orientation</b>	<b>Function Description</b>
I0	Input	Input Signal
ID	Input	Input signal, valid when ID_TO_LUT = "TRUE"
I1	Input	Input Signal
I2	Input	Input Signal
I3	Input	Input Signal
I4	Input	Input signal, valid when I4_TO_CARRY = "TRUE"
CIN	Input	Cascade input, valid when CIN_TO_LUT = "TRUE", must be connected to COUT
COUT	Output	Cascade output
Z	Output	Output signal

#### 4.10.4 Parameter Description

Table 4-31 GTP\_LUT5CARRY Parameter Description

<b>Parameter</b>	<b>Description</b>	<b>Defaults</b>	<b>Setting Value</b>
INIT	Output Configuration Parameter	32'h0000_0000	32'h0 ~ 32'hffff_ffff
ID_TO_LUT	Select either I0 or ID as an input for the LUT	"FALSE"	TRUE: The selection of ID is valid; FALSE: The selection of I0 is valid.
CIN_TO_LUT	Select either CIN or I0 as an input for the LUT	"TRUE"	TRUE: The selection of CIN is valid; FALSE: The selection of I0 is valid.
I4_TO_CARRY	Select the output of either I4 or LUT4 for the CARRY CHAIN	"TRUE"	TRUE: The selection of I4 is valid; FALSE: The selection of LUT4 output is valid.
I4_TO_LUT	Select either I4 or 1'b1 as an input for the LUT	"FALSE"	TRUE: The selection of I4 is valid; FALSE: The selection of 1'b1 is valid.

#### 4.10.5 Instantiation template

GTP\_LUT5CARRY

```
#(
    .INIT      (32'h0000_0000),
    .ID_TO_LUT ("FALSE"),
    .CIN_TO_LUT ("TRUE"),
```

```

.I4_TO_CARRY ("TRUE"),
.I4_TO_LUT      ("TRUE")
)GTP_LUT5CARRY_inst (
.COUT      (cout),
.Z         (z),
.CIN      (cin),
.I0        (i0),
.ID        (id),
.I1        (i1),
.I2        (i2),
.I3        (i3),
.I4        (i4)
);
    
```

## 4.11 Usage Instructions for GTP\_LUT6

### 4.11.1 Supported Devices

Table 4-32 Device Models That Support GTP\_LUT6

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.11.2 Description of Functionality

GTP\_LUT6 is a 6-input lookup table, with different INIT parameter values corresponding to different functions of LUT6. The corresponding INIT parameter must be specified when used. The structure block diagram is shown below.

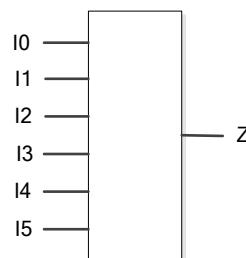


Figure 4-11 GTP\_LUT6 Structure Diagram

### 4.11.3 Port Description

Table 4-33 GTP\_LUT6 Port Description

<b>Port</b>	<b>Orientation</b>	<b>Function Description</b>
Z	Output	Output signal
I0	Input	Input Signal
I1	Input	Input Signal
I2	Input	Input Signal
I3	Input	Input Signal
I4	Input	Input Signal
I5	Input	Input Signal

Truth Table:

Table 4-34 GTP\_LUT6 Truth Table

<b>Input</b>						<b>Output</b>
I5	I4	I3	I2	I1	I0	Z
0	0	0	0	0	0	INIT[0]
0	0	0	0	0	1	INIT[1]
0	0	0	0	1	0	INIT[2]
0	0	0	0	1	1	INIT[3]
0	0	0	1	0	0	INIT[4]
0	0	0	1	0	1	INIT[5]
0	0	0	1	1	0	INIT[6]
0	0	0	1	1	1	INIT[7]
0	0	1	0	0	0	INIT[8]
0	0	1	0	0	1	INIT[9]
0	0	1	0	1	0	INIT[10]
0	0	1	0	1	1	INIT[11]
0	0	1	1	0	0	INIT[12]
0	0	1	1	0	1	INIT[13]
0	0	1	1	1	0	INIT[14]
0	0	1	1	1	1	INIT[15]
0	1	0	0	0	0	INIT[16]
0	1	0	0	0	1	INIT[17]
0	1	0	0	1	0	INIT[18]
0	1	0	0	1	1	INIT[19]
0	1	0	1	0	0	INIT[20]
0	1	0	1	0	1	INIT[21]
0	1	0	1	1	0	INIT[22]
0	1	0	1	1	1	INIT[23]

<b>Input</b>						<b>Output</b>
0	1	1	0	0	0	INIT[24]
0	1	1	0	0	1	INIT[25]
0	1	1	0	1	0	INIT[26]
0	1	1	0	1	1	INIT[27]
0	1	1	1	0	0	INIT[28]
0	1	1	1	0	1	INIT[29]
0	1	1	1	1	0	INIT[30]
0	1	1	1	1	1	INIT[31]
1	0	0	0	0	0	INIT[32]
1	0	0	0	0	1	INIT[33]
1	0	0	0	1	0	INIT[34]
1	0	0	0	1	1	INIT[35]
1	0	0	1	0	0	INIT[36]
1	0	0	1	0	1	INIT[37]
1	0	0	1	1	0	INIT[38]
1	0	0	1	1	1	INIT[39]
1	0	1	0	0	0	INIT[40]
1	0	1	0	0	1	INIT[41]
1	0	1	0	1	0	INIT[42]
1	0	1	0	1	1	INIT[43]
1	0	1	1	0	0	INIT[44]
1	0	1	1	0	1	INIT[45]
1	0	1	1	1	0	INIT[46]
1	0	1	1	1	1	INIT[47]
1	1	0	0	0	0	INIT[48]
1	1	0	0	0	1	INIT[49]
1	1	0	0	1	0	INIT[50]
1	1	0	0	1	1	INIT[51]
1	1	0	1	0	0	INIT[52]
1	1	0	1	0	1	INIT[53]
1	1	0	1	1	0	INIT[54]
1	1	0	1	1	1	INIT[55]
1	1	1	0	0	0	INIT[56]
1	1	1	0	0	1	INIT[57]
1	1	1	0	1	0	INIT[58]
1	1	1	0	1	1	INIT[59]
1	1	1	1	0	0	INIT[60]
1	1	1	1	0	1	INIT[61]
1	1	1	1	1	0	INIT[62]

Input						Output
1	1	1	1	1	1	INIT[63]

#### 4.11.4 Paramater Description

Table 4-35 GTP\_LUT6 Parameter Description

Parameter	Description	Defaults	Setting Value
INIT	Output Configuration Parameter	64'h0000_0000_0000_0000	0 ~ 2^64-1

#### 4.11.5 Instantiation template

GTP\_LUT6

```
#(
    .INIT  (64'h0000_0000_0000_0000)
)GTP_LUT6_inst (
    .Z      (z),
    .I0     (i0),
    .I1     (i1),
    .I2     (i2),
    .I3     (i3),
    .I4     (i4),
    .I5     (i5)
);
```

## 4.12 Usage Instructions for GTP\_LUT7

### 4.12.1 Supported Devices

Table 4-36 Device Models That Support GTP\_LUT7

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.12.2 Description of Functionality

GTP\_LUT7 is a 7-input lookup table, with different INIT parameter values corresponding to different functions of LUT7; the corresponding INIT parameter must be specified when used. The

structure block diagram is shown below.

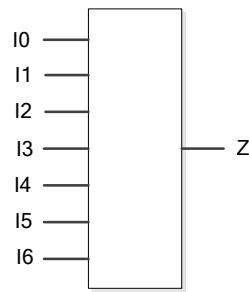


Figure 4-12 GTP\_LUT7 Structure Diagram

#### 4.12.3 Port Description

Table 4-37 GTP\_LUT7 Port Description

<b>Port</b>	<b>Orientation</b>	<b>Function Description</b>
Z	Output	Output signal
I0	Input	Input Signal
I1	Input	Input Signal
I2	Input	Input Signal
I3	Input	Input Signal
I4	Input	Input Signal
I5	Input	Input Signal
I6	Input	Input Signal

Truth Table:

Table 4-38 GTP\_LUT7 Truth Table

<b>Input</b>							<b>Output</b>
I6	I5	I4	I3	I2	I1	I0	Z
0	0	0	0	0	0	0	INIT[0]
0	0	0	0	0	0	1	INIT[1]
0	0	0	0	0	1	0	INIT[2]
0	0	0	0	0	1	1	INIT[3]
0	0	0	0	1	0	0	INIT[4]
0	0	0	0	1	0	1	INIT[5]
0	0	0	0	1	1	0	INIT[6]
0	0	0	0	1	1	1	INIT[7]
0	0	0	1	0	0	0	INIT[8]
0	0	0	1	0	0	1	INIT[9]
0	0	0	1	0	1	0	INIT[10]
0	0	0	1	0	1	1	INIT[11]
0	0	0	1	1	0	0	INIT[12]
0	0	0	1	1	0	1	INIT[13]

Input							Output
0	0	0	1	1	1	0	INIT[14]
0	0	0	1	1	1	1	INIT[15]
0	0	1	0	0	0	0	INIT[16]
0	0	1	0	0	0	1	INIT[17]
0	0	1	0	0	1	0	INIT[18]
0	0	1	0	0	1	1	INIT[19]
0	0	1	0	1	0	0	INIT[20]
0	0	1	0	1	0	1	INIT[21]
0	0	1	0	1	1	0	INIT[22]
0	0	1	0	1	1	1	INIT[23]
0	0	1	1	0	0	0	INIT[24]
0	0	1	1	0	0	1	INIT[25]
0	0	1	1	0	1	0	INIT[26]
0	0	1	1	0	1	1	INIT[27]
0	0	1	1	1	0	0	INIT[28]
0	0	1	1	1	0	1	INIT[29]
0	0	1	1	1	1	0	INIT[30]
0	0	1	1	1	1	1	INIT[31]
0	1	0	0	0	0	0	INIT[32]
0	1	0	0	0	0	1	INIT[33]
0	1	0	0	0	1	0	INIT[34]
0	1	0	0	0	1	1	INIT[35]
0	1	0	0	1	0	0	INIT[36]
0	1	0	0	1	0	1	INIT[37]
0	1	0	0	1	1	0	INIT[38]
0	1	0	0	1	1	1	INIT[39]
0	1	0	1	0	0	0	INIT[40]
0	1	0	1	0	0	1	INIT[41]
0	1	0	1	0	1	0	INIT[42]
0	1	0	1	0	1	1	INIT[43]
0	1	0	1	1	0	0	INIT[44]
0	1	0	1	1	0	1	INIT[45]
0	1	0	1	1	1	0	INIT[46]
0	1	0	1	1	1	1	INIT[47]
0	1	1	0	0	0	0	INIT[48]
0	1	1	0	0	0	1	INIT[49]
0	1	1	0	0	1	0	INIT[50]
0	1	1	0	0	1	1	INIT[51]
0	1	1	0	1	0	0	INIT[52]

Input							Output
0	1	1	0	1	0	1	INIT[53]
0	1	1	0	1	1	0	INIT[54]
0	1	1	0	1	1	1	INIT[55]
0	1	1	1	0	0	0	INIT[56]
0	1	1	1	0	0	1	INIT[57]
0	1	1	1	0	1	0	INIT[58]
0	1	1	1	0	1	1	INIT[59]
0	1	1	1	1	0	0	INIT[60]
0	1	1	1	1	0	1	INIT[61]
0	1	1	1	1	1	0	INIT[62]
0	1	1	1	1	1	1	INIT[63]
1	0	0	0	0	0	0	INIT[64]
1	0	0	0	0	0	1	INIT[65]
1	0	0	0	0	1	0	INIT[66]
1	0	0	0	0	1	1	INIT[67]
1	0	0	0	1	0	0	INIT[68]
1	0	0	0	1	0	1	INIT[69]
1	0	0	0	1	1	0	INIT[70]
1	0	0	0	1	1	1	INIT[71]
1	0	0	1	0	0	0	INIT[72]
1	0	0	1	0	0	1	INIT[73]
1	0	0	1	0	1	0	INIT[74]
1	0	0	1	0	1	1	INIT[75]
1	0	0	1	1	0	0	INIT[76]
1	0	0	1	1	0	1	INIT[77]
1	0	0	1	1	1	0	INIT[78]
1	0	0	1	1	1	1	INIT[79]
1	0	1	0	0	0	0	INIT[80]
1	0	1	0	0	0	1	INIT[81]
1	0	1	0	0	1	0	INIT[82]
1	0	1	0	0	1	1	INIT[83]
1	0	1	0	1	0	0	INIT[84]
1	0	1	0	1	0	1	INIT[85]
1	0	1	0	1	1	0	INIT[86]
1	0	1	0	1	1	1	INIT[87]
1	0	1	1	0	0	0	INIT[88]
1	0	1	1	0	0	1	INIT[89]
1	0	1	1	0	1	0	INIT[90]
1	0	1	1	0	1	1	INIT[91]

Input							Output
1	0	1	1	1	0	0	INIT[92]
1	0	1	1	1	0	1	INIT[93]
1	0	1	1	1	1	0	INIT[94]
1	0	1	1	1	1	1	INIT[95]
1	1	0	0	0	0	0	INIT[96]
1	1	0	0	0	0	1	INIT[97]
1	1	0	0	0	1	0	INIT[98]
1	1	0	0	0	1	1	INIT[99]
1	1	0	0	1	0	0	INIT[100]
1	1	0	0	1	0	1	INIT[101]
1	1	0	0	1	1	0	INIT[102]
1	1	0	0	1	1	1	INIT[103]
1	1	0	1	0	0	0	INIT[104]
1	1	0	1	0	0	1	INIT[105]
1	1	0	1	0	1	0	INIT[106]
1	1	0	1	0	1	1	INIT[107]
1	1	0	1	1	0	0	INIT[108]
1	1	0	1	1	0	1	INIT[109]
1	1	0	1	1	1	0	INIT[110]
1	1	0	1	1	1	1	INIT[111]
1	1	1	0	0	0	0	INIT[112]
1	1	1	0	0	0	1	INIT[113]
1	1	1	0	0	1	0	INIT[114]
1	1	1	0	0	1	1	INIT[115]
1	1	1	0	1	0	0	INIT[116]
1	1	1	0	1	0	1	INIT[117]
1	1	1	0	1	1	0	INIT[118]
1	1	1	0	1	1	1	INIT[119]
1	1	1	1	0	0	0	INIT[120]
1	1	1	1	0	0	1	INIT[121]
1	1	1	1	0	1	0	INIT[122]
1	1	1	1	0	1	1	INIT[123]
1	1	1	1	1	0	0	INIT[124]
1	1	1	1	1	0	1	INIT[125]
1	1	1	1	1	1	0	INIT[126]
1	1	1	1	1	1	1	INIT[127]

#### 4.12.4 Parameter Description

Table 4-39 GTP\_LUT7 Parameter Description

Parameter	Description	Defaults	Setting Value
INIT	Output Configuration Parameter	128'h00000000_00000000_00000000_00000000	0 ~ 2^128-1

#### 4.12.5 Instantiation template

```
GTP_LUT7
#(
    .INIT  (128'h00000000_00000000_00000000_00000000)
)GTP_LUT7_inst (
    .Z      (z),
    .I0     (i0),
    .I1     (i1),
    .I2     (i2),
    .I3     (i3),
    .I4     (i4),
    .I5     (i5),
    .I6     (i6)
);
```

### 4.13 Usage Instructions for GTP\_LUT8

#### 4.13.1 Supported Devices

Table 4-40 Device Models That Support GTP\_LUT8

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 4.13.2 Description of Functionality

GTP\_LUT8 is an 8-input lookup table, with different INIT parameter values corresponding to different functions of LUT8; the corresponding INIT parameter must be specified when used. The structure block diagram is shown below.

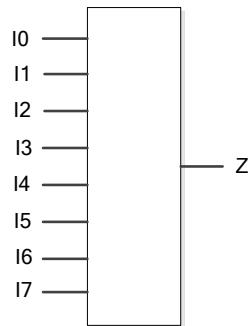


Figure 4-13 GTP\_LUT8 Structure Diagram

#### 4.13.3 Port Description

Table 4-41 GTP\_LUT8 Port Description

Port	Orientation	Function Description
Z	Output	Output signal
I0	Input	Input Signal
I1	Input	Input Signal
I2	Input	Input Signal
I3	Input	Input Signal
I4	Input	Input Signal
I5	Input	Input Signal
I6	Input	Input Signal
I7	Input	Input Signal

Truth Table:

Table 4-42 GTP\_LUT8 Truth Table

Input									Output
I7	I6	I5	I4	I3	I2	I1	I0	Z	
0	0	0	0	0	0	0	0	INIT[0]	
0	0	0	0	0	0	0	1	INIT[1]	
0	0	0	0	0	0	1	0	INIT[2]	
0	0	0	0	0	0	1	1	INIT[3]	
0	0	0	0	0	1	0	0	INIT[4]	
0	0	0	0	0	1	0	1	INIT[5]	
0	0	0	0	0	1	1	0	INIT[6]	
0	0	0	0	0	1	1	1	INIT[7]	
0	0	0	0	1	0	0	0	INIT[8]	
0	0	0	0	1	0	0	1	INIT[9]	
0	0	0	0	1	0	1	0	INIT[10]	
0	0	0	0	1	0	1	1	INIT[11]	
0	0	0	0	1	1	0	0	INIT[12]	

Input								Output
0	0	0	0	1	1	0	1	INIT[13]
0	0	0	0	1	1	1	0	INIT[14]
0	0	0	0	1	1	1	1	INIT[15]
0	0	0	1	0	0	0	0	INIT[16]
0	0	0	1	0	0	0	1	INIT[17]
0	0	0	1	0	0	1	0	INIT[18]
0	0	0	1	0	0	1	1	INIT[19]
0	0	0	1	0	1	0	0	INIT[20]
0	0	0	1	0	1	0	1	INIT[21]
0	0	0	1	0	1	1	0	INIT[22]
0	0	0	1	0	1	1	1	INIT[23]
0	0	0	1	1	0	0	0	INIT[24]
0	0	0	1	1	0	0	1	INIT[25]
0	0	0	1	1	0	1	0	INIT[26]
0	0	0	1	1	0	1	1	INIT[27]
0	0	0	1	1	1	0	0	INIT[28]
0	0	0	1	1	1	0	1	INIT[29]
0	0	0	1	1	1	1	0	INIT[30]
0	0	0	1	1	1	1	1	INIT[31]
0	0	1	0	0	0	0	0	INIT[32]
0	0	1	0	0	0	0	1	INIT[33]
0	0	1	0	0	0	1	0	INIT[34]
0	0	1	0	0	0	1	1	INIT[35]
0	0	1	0	0	1	0	0	INIT[36]
0	0	1	0	0	1	0	1	INIT[37]
0	0	1	0	0	1	1	0	INIT[38]
0	0	1	0	0	1	1	1	INIT[39]
0	0	1	0	1	0	0	0	INIT[40]
0	0	1	0	1	0	0	1	INIT[41]
0	0	1	0	1	0	1	0	INIT[42]
0	0	1	0	1	0	1	1	INIT[43]
0	0	1	0	1	1	0	0	INIT[44]
0	0	1	0	1	1	0	1	INIT[45]
0	0	1	0	1	1	1	0	INIT[46]
0	0	1	0	1	1	1	1	INIT[47]
0	0	1	1	0	0	0	0	INIT[48]
0	0	1	1	0	0	0	1	INIT[49]
0	0	1	1	0	0	1	0	INIT[50]
0	0	1	1	0	0	1	1	INIT[51]

Input								Output
0	0	1	1	0	1	0	0	INIT[52]
0	0	1	1	0	1	0	1	INIT[53]
0	0	1	1	0	1	1	0	INIT[54]
0	0	1	1	0	1	1	1	INIT[55]
0	0	1	1	1	0	0	0	INIT[56]
0	0	1	1	1	0	0	1	INIT[57]
0	0	1	1	1	0	1	0	INIT[58]
0	0	1	1	1	0	1	1	INIT[59]
0	0	1	1	1	1	0	0	INIT[60]
0	0	1	1	1	1	0	1	INIT[61]
0	0	1	1	1	1	1	0	INIT[62]
0	0	1	1	1	1	1	1	INIT[63]
0	1	0	0	0	0	0	0	INIT[64]
0	1	0	0	0	0	0	1	INIT[65]
0	1	0	0	0	0	1	0	INIT[66]
0	1	0	0	0	0	1	1	INIT[67]
0	1	0	0	0	1	0	0	INIT[68]
0	1	0	0	0	1	0	1	INIT[69]
0	1	0	0	0	1	1	0	INIT[70]
0	1	0	0	0	1	1	1	INIT[71]
0	1	0	0	1	0	0	0	INIT[72]
0	1	0	0	1	0	0	1	INIT[73]
0	1	0	0	1	0	1	0	INIT[74]
0	1	0	0	1	0	1	1	INIT[75]
0	1	0	0	1	1	0	0	INIT[76]
0	1	0	0	1	1	0	1	INIT[77]
0	1	0	0	1	1	1	0	INIT[78]
0	1	0	0	1	1	1	1	INIT[79]
0	1	0	1	0	0	0	0	INIT[80]
0	1	0	1	0	0	0	1	INIT[81]
0	1	0	1	0	0	1	0	INIT[82]
0	1	0	1	0	0	1	1	INIT[83]
0	1	0	1	0	1	0	0	INIT[84]
0	1	0	1	0	1	0	1	INIT[85]
0	1	0	1	0	1	1	0	INIT[86]
0	1	0	1	0	1	1	1	INIT[87]
0	1	0	1	1	0	0	0	INIT[88]
0	1	0	1	1	0	0	1	INIT[89]
0	1	0	1	1	0	1	0	INIT[90]

Input								Output
0	1	0	1	1	0	1	1	INIT[91]
0	1	0	1	1	1	0	0	INIT[92]
0	1	0	1	1	1	0	1	INIT[93]
0	1	0	1	1	1	1	0	INIT[94]
0	1	0	1	1	1	1	1	INIT[95]
0	1	1	0	0	0	0	0	INIT[96]
0	1	1	0	0	0	0	1	INIT[97]
0	1	1	0	0	0	1	0	INIT[98]
0	1	1	0	0	0	1	1	INIT[99]
0	1	1	0	0	1	0	0	INIT[100]
0	1	1	0	0	1	0	1	INIT[101]
0	1	1	0	0	1	1	0	INIT[102]
0	1	1	0	0	1	1	1	INIT[103]
0	1	1	0	1	0	0	0	INIT[104]
0	1	1	0	1	0	0	1	INIT[105]
0	1	1	0	1	0	1	0	INIT[106]
0	1	1	0	1	0	1	1	INIT[107]
0	1	1	0	1	1	0	0	INIT[108]
0	1	1	0	1	1	0	1	INIT[109]
0	1	1	0	1	1	1	0	INIT[110]
0	1	1	0	1	1	1	1	INIT[111]
0	1	1	1	0	0	0	0	INIT[112]
0	1	1	1	0	0	0	1	INIT[113]
0	1	1	1	0	0	1	0	INIT[114]
0	1	1	1	0	0	1	1	INIT[115]
0	1	1	1	0	1	0	0	INIT[116]
0	1	1	1	0	1	0	1	INIT[117]
0	1	1	1	0	1	1	0	INIT[118]
0	1	1	1	0	1	1	1	INIT[119]
0	1	1	1	1	0	0	0	INIT[120]
0	1	1	1	1	0	0	1	INIT[121]
0	1	1	1	1	0	1	0	INIT[122]
0	1	1	1	1	0	1	1	INIT[123]
0	1	1	1	1	1	0	0	INIT[124]
0	1	1	1	1	1	0	1	INIT[125]
0	1	1	1	1	1	1	0	INIT[126]
0	1	1	1	1	1	1	1	INIT[127]
1	0	0	0	0	0	0	0	INIT[128]
1	0	0	0	0	0	0	1	INIT[129]

<b>Input</b>									<b>Output</b>
1	0	0	0	0	0	1	0		INIT[130]
1	0	0	0	0	0	1	1		INIT[131]
1	0	0	0	0	1	0	0		INIT[132]
1	0	0	0	0	1	0	1		INIT[133]
1	0	0	0	0	1	1	0		INIT[134]
1	0	0	0	0	1	1	1		INIT[135]
1	0	0	0	1	0	0	0		INIT[136]
1	0	0	0	1	0	0	1		INIT[137]
1	0	0	0	1	0	1	0		INIT[138]
1	0	0	0	1	0	1	1		INIT[139]
1	0	0	0	1	1	0	0		INIT[140]
1	0	0	0	1	1	0	1		INIT[141]
1	0	0	0	1	1	1	0		INIT[142]
1	0	0	0	1	1	1	1		INIT[143]
1	0	0	1	0	0	0	0		INIT[144]
1	0	0	1	0	0	0	1		INIT[145]
1	0	0	1	0	0	1	0		INIT[146]
1	0	0	1	0	0	1	1		INIT[147]
1	0	0	1	0	1	0	0		INIT[148]
1	0	0	1	0	1	0	1		INIT[149]
1	0	0	1	0	1	1	0		INIT[150]
1	0	0	1	0	1	1	1		INIT[151]
1	0	0	1	1	0	0	0		INIT[152]
1	0	0	1	1	0	0	1		INIT[153]
1	0	0	1	1	0	1	0		INIT[154]
1	0	0	1	1	0	1	1		INIT[155]
1	0	0	1	1	1	0	0		INIT[156]
1	0	0	1	1	1	1	0		INIT[157]
1	0	0	1	1	1	1	1		INIT[158]
1	0	0	1	1	1	1	1		INIT[159]
1	0	1	0	0	0	0	0		INIT[160]
1	0	1	0	0	0	0	1		INIT[161]
1	0	1	0	0	0	1	0		INIT[162]
1	0	1	0	0	0	1	1		INIT[163]
1	0	1	0	0	1	0	0		INIT[164]
1	0	1	0	0	1	0	1		INIT[165]
1	0	1	0	0	1	1	0		INIT[166]
1	0	1	0	0	1	1	1		INIT[167]
1	0	1	0	1	0	0	0		INIT[168]

Input								Output
1	0	1	0	1	0	0	1	INIT[169]
1	0	1	0	1	0	1	0	INIT[170]
1	0	1	0	1	0	1	1	INIT[171]
1	0	1	0	1	1	0	0	INIT[172]
1	0	1	0	1	1	0	1	INIT[173]
1	0	1	0	1	1	1	0	INIT[174]
1	0	1	0	1	1	1	1	INIT[175]
1	0	1	1	0	0	0	0	INIT[176]
1	0	1	1	0	0	0	1	INIT[177]
1	0	1	1	0	0	1	0	INIT[178]
1	0	1	1	0	0	1	1	INIT[179]
1	0	1	1	0	1	0	0	INIT[180]
1	0	1	1	0	1	0	1	INIT[181]
1	0	1	1	0	1	1	0	INIT[182]
1	0	1	1	0	1	1	1	INIT[183]
1	0	1	1	1	0	0	0	INIT[184]
1	0	1	1	1	0	0	1	INIT[185]
1	0	1	1	1	0	1	0	INIT[186]
1	0	1	1	1	1	0	0	INIT[188]
1	0	1	1	1	1	0	1	INIT[189]
1	0	1	1	1	1	1	0	INIT[190]
1	0	1	1	1	1	1	1	INIT[191]
1	1	0	0	0	0	0	0	INIT[192]
1	1	0	0	0	0	0	1	INIT[193]
1	1	0	0	0	0	1	0	INIT[194]
1	1	0	0	0	0	1	1	INIT[195]
1	1	0	0	0	1	0	0	INIT[196]
1	1	0	0	0	1	0	1	INIT[197]
1	1	0	0	0	1	1	0	INIT[198]
1	1	0	0	0	1	1	1	INIT[199]
1	1	0	0	1	0	0	0	INIT[200]
1	1	0	0	1	0	0	1	INIT[201]
1	1	0	0	1	0	1	0	INIT[202]
1	1	0	0	1	0	1	1	INIT[203]
1	1	0	0	1	1	0	0	INIT[204]
1	1	0	0	1	1	0	1	INIT[205]
1	1	0	0	1	1	1	0	INIT[206]
1	1	0	0	1	1	1	1	INIT[207]

Input								Output
1	1	0	1	0	0	0	0	INIT[208]
1	1	0	1	0	0	0	1	INIT[209]
1	1	0	1	0	0	1	0	INIT[210]
1	1	0	1	0	0	1	1	INIT[211]
1	1	0	1	0	1	0	0	INIT[212]
1	1	0	1	0	1	0	1	INIT[213]
1	1	0	1	0	1	1	0	INIT[214]
1	1	0	1	0	1	1	1	INIT[215]
1	1	0	1	1	0	0	0	INIT[216]
1	1	0	1	1	0	0	1	INIT[217]
1	1	0	1	1	0	1	0	INIT[218]
1	1	0	1	1	0	1	1	INIT[219]
1	1	0	1	1	1	0	0	INIT[220]
1	1	0	1	1	1	0	1	INIT[221]
1	1	0	1	1	1	1	0	INIT[222]
1	1	0	1	1	1	1	1	INIT[223]
1	1	1	0	0	0	0	0	INIT[224]
1	1	1	0	0	0	0	1	INIT[225]
1	1	1	0	0	0	1	0	INIT[226]
1	1	1	0	0	0	1	1	INIT[227]
1	1	1	0	0	1	0	0	INIT[228]
1	1	1	0	0	1	0	1	INIT[229]
1	1	1	0	0	1	1	0	INIT[230]
1	1	1	0	0	1	1	1	INIT[231]
1	1	1	0	1	0	0	0	INIT[232]
1	1	1	0	1	0	0	1	INIT[233]
1	1	1	0	1	0	1	0	INIT[234]
1	1	1	0	1	0	1	1	INIT[235]
1	1	1	0	1	1	0	0	INIT[236]
1	1	1	0	1	1	0	1	INIT[237]
1	1	1	0	1	1	1	0	INIT[238]
1	1	1	0	1	1	1	1	INIT[239]
1	1	1	1	0	0	0	0	INIT[240]
1	1	1	1	0	0	0	1	INIT[241]
1	1	1	1	0	0	1	0	INIT[242]
1	1	1	1	0	0	1	1	INIT[243]
1	1	1	1	0	1	0	0	INIT[244]
1	1	1	1	0	1	0	1	INIT[245]
1	1	1	1	0	1	1	0	INIT[246]

Input								Output
1	1	1	1	0	1	1	1	INIT[247]
1	1	1	1	1	0	0	0	INIT[248]
1	1	1	1	1	0	0	1	INIT[249]
1	1	1	1	1	0	1	0	INIT[250]
1	1	1	1	1	0	1	1	INIT[251]
1	1	1	1	1	1	0	0	INIT[252]
1	1	1	1	1	1	0	1	INIT[253]
1	1	1	1	1	1	1	0	INIT[254]
1	1	1	1	1	1	1	1	INIT[255]

#### 4.13.4 Paramater Description

Table 4-43 GTP\_LUT8 Parameter Description

Parameter	Description	Defaults	Setting Value
INIT	Output Configuration Parameter	256'h0000000000000000_0000000000000000_0000000000000000 0000_0000000000000000	0 ~ 2^256-1

#### 4.13.5 Instantiation template

GTP\_LUT8

```
#(
    .INIT  (256'h0000000000000000_0000000000000000_0000000000000000_0000000000000000),
    )GTP_LUT8_inst (
        .Z      (z),
        .I0     (i0),
        .I1     (i1),
        .I2     (i2),
        .I3     (i3),
        .I4     (i4),
        .I5     (i5),
        .I6     (i6),
        .I7     (i7)
    );
```

## 4.14 GTP\_LUTMUX4 Usage Instructions

### 4.14.1 Supported Devices

Table 4-44 Device Models That Support GTP\_LUTMUX4

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.14.2 Description of Functionality

GTP\_LUTMUX4 is a 4:1 MUX selector, with I0, I1, I2, and I3 as data inputs, and S1 and S0 as selection signals. It outputs I0 when {S1, S0} is 2'b00, I1 when {S1, S0} is 2'b01, I2 when {S1, S0} is 2'b10, and I3 when {S1, S0} is 2'b11. Its structure block diagram is as follows:

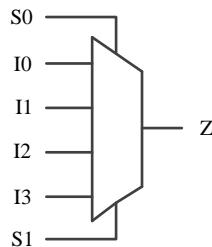


Figure 4-14 GTP\_LUTMUX4 Structure Diagram

### 4.14.3 Port Description

Table 4-45 GTP\_LUTMUX4 Port Description

Port	Orientation	Function Description
S0	Input	Select signal 0
S1	Input	Select signal 1
I0	Input	MUX input signal
I1	Input	MUX input signal
I2	Input	MUX input signal
I3	Input	MUX input signal
Z	Output	MUX output signal

Truth Table:

Table 4-46 GTP\_LUTMUX4 Truth Table

Input						Output
S1	S0	I3	I2	I1	I0	Z
0	0	?	?	?	0	0
0	0	?	?	?	1	1

Input						Output
0	1	?	?	0	?	0
0	1	?	?	1	?	1
1	0	?	0	?	?	0
1	0	?	1	?	?	1
1	1	0	?	?	?	0
1	1	1	?	?	?	1

#### 4.14.4 Instantiation template

```
GTP_LUTMUX4 GTP_LUTMUX4_inst
(
    .S0  (s0),
    .S1  (s1),
    .I0  (i0),
    .I1  (i1),
    .I2  (i2),
    .I3  (i3),
    .Z   (z)
);
```

### 4.15 Usage Instructions for GTP\_MUX2LUT6

#### 4.15.1 Supported Devices

Table 4-47 Device Models That Support GTP\_MUX2LUT6

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 4.15.2 Description of Functionality

GTP\_MUX2LUT6 is a dual-input selector dedicated to GTP\_LUT6, with its structure block diagram shown below. Where I0 and I1 are data inputs, S is the selection signal. It outputs I0 when S is 1'b0 and I1 when S is 1'b1. The structure block diagram is shown below. GTP\_MUX2LUT6 is used to select the outputs of two LUT5s to form a single LUT6, thus it must be used in conjunction with LUT5.

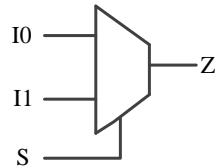


Figure 4-15 GTP\_MUX2LUT6 Structure Diagram

#### 4.15.3 Port Description

Table 4-48 GTP\_MUX2LUT6 Port Description

Port	Orientation	Function Description
I0	Input	Input Signal
I1	Input	Input Signal
S	Input	Selection signal
Z	Output	Output signal

#### 4.15.4 Instantiation template

```
GTP_MUX2LUT6 GTP_MUX2LUT6_inst (
    .Z(), // OUTPUT
    .I0(),// INPUT
    .I1(),// INPUT
    .S()  // INPUT
);
```

### 4.16 Usage Instructions for GTP\_MUX2LUT7

#### 4.16.1 Supported Devices

Table 4-49 Device Models That Support GTP\_MUX2LUT7

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 4.16.2 Description of Functionality

GTP\_MUX2LUT7 is a dual-input selector dedicated to GTP\_LUT7, with its structure block diagram shown below. Where I0 and I1 are data inputs, S is the selection signal. It outputs I0 when S is 1'b0 and I1 when S is 1'b1. The structure block diagram is shown below. GTP\_MUX2LUT7 is used to select the outputs of two LUT6s to form a single LUT7, thus it must be used in conjunction

with LUT6.

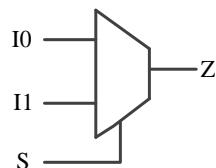


Figure 4-16 GTP\_MUX2LUT7 Structure Diagram

#### 4.16.3 Port Description

Table 4-50 GTP\_MUX2LUT7 Port Description

Port	Orientation	Function Description
I0	Input	Input Signal
I1	Input	Input Signal
S	Input	Selection signal
Z	Output	Output signal

#### 4.16.4 Instantiation template

```
GTP_MUX2LUT7 GTP_MUX2LUT7_inst (
    .Z(), // OUTPUT
    .I0(),// INPUT
    .I1(),// INPUT
    .S() // INPUT
);
```

### 4.17 Usage Instructions for GTP\_MUX2LUT8

#### 4.17.1 Supported Devices

Table 4-51 Device Models That Support GTP\_MUX2LUT8

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 4.17.2 Description of Functionality

GTP\_MUX2LUT8 is a dual-input selector dedicated to GTP\_LUT8, with its structure block diagram shown below. Where I0 and I1 are data inputs, S is the selection signal. It outputs I0 when

S is 1'b0 and I1 when S is 1'b1. The structure block diagram is shown below. GTP\_MUX2LUT8 is used to select the outputs of two LUT7s to form a single LUT8, thus it must be used in conjunction with LUT7.

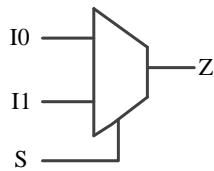


Figure 4-17 GTP\_MUX2LUT8 Structure Diagram

#### 4.17.3 Port Description

Table 4-52 GTP\_MUX2LUT8 Port Description

<b>Port</b>	<b>Orientation</b>	<b>Function Description</b>
I0	Input	Input Signal
I1	Input	Input Signal
S	Input	Selection signal
Z	Output	Output signal

#### 4.17.4 Instantiation template

```

GTP_MUX2LUT8 GTP_MUX2LUT8_inst (
    .Z(), // OUTPUT
    .I0(),// INPUT
    .I1(),// INPUT
    .S() // INPUT
);
  
```

## 4.18 Usage Instructions for GTP\_LUT5M

### 4.18.1 Supported Devices

Table 4-53 Device Models That Support GTP\_LUT5M

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 4.18.2 Description of Functionality

GTP\_LUT5M is a 5-input lookup table, with different INIT parameter values corresponding to different functions of LUT5; the corresponding INIT parameter must be specified when used. The structure block diagram is shown below.

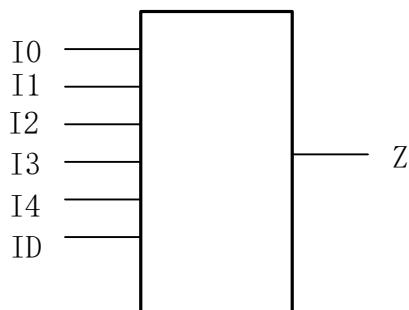


Figure 4-18 GTP\_LUT5M Structure Diagram

### 4.18.3 Port Description

Table 4-54 GTP\_LUT5CARRY Port Description

Port	Orientation	Function Description
Z	Output	Output signal
I0	Input	Input Signal
I1	Input	Input Signal
I2	Input	Input Signal
I3	Input	Input Signal
I4	Input	Input Signal
ID	Input	Input Signal

Truth Table:

Table 4-55 GTP\_LUT5M Truth Table

Input	Output					
I4	I3	I2	I1	I0	ID	Z

<b>Input</b>						<b>Output</b>
0	0	0	0	0/1	0	INIT[0]
0	0	0	0	0/1	1	INIT[1]
0	0	0	1	0/1	0	INIT[2]
0	0	0	1	0/1	1	INIT[3]
0	0	1	0	0/1	0	INIT[4]
0	0	1	0	0/1	1	INIT[5]
0	0	1	1	0/1	0	INIT[6]
0	0	1	1	0/1	1	INIT[7]
0	1	0	0	0/1	0	INIT[8]
0	1	0	0	0/1	1	INIT[9]
0	1	0	1	0/1	0	INIT[10]
0	1	0	1	0/1	1	INIT[11]
0	1	1	0	0/1	0	INIT[12]
0	1	1	0	0/1	1	INIT[13]
0	1	1	1	0/1	0	INIT[14]
0	1	1	1	0/1	1	INIT[15]
1	0	0	0	0	0/1	INIT[16]
1	0	0	0	1	0/1	INIT[17]
1	0	0	1	0	0/1	INIT[18]
1	0	0	1	1	0/1	INIT[19]
1	0	1	0	0	0/1	INIT[20]
1	0	1	0	1	0/1	INIT[21]
1	0	1	1	0	0/1	INIT[22]
1	0	1	1	1	0/1	INIT[23]
1	1	0	0	0	0/1	INIT[24]
1	1	0	0	1	0/1	INIT[25]
1	1	0	1	0	0/1	INIT[26]
1	1	0	1	1	0/1	INIT[27]
1	1	1	0	0	0/1	INIT[28]
1	1	1	0	1	0/1	INIT[29]
1	1	1	1	0	0/1	INIT[30]
1	1	1	1	1	0/1	INIT[31]

#### 4.18.4 Paramater Description

Table 4-56 GTP\_LUT5M Parameter Description

<b>Parameter</b>	<b>Description</b>	<b>Defaults</b>	<b>Setting Value</b>
INIT	Output Configuration Parameter	32'h0000_0000	32'h0 ~ 32'hffff_ffff

#### 4.18.5 Instantiation template

```
GTP_LUT5M #(  
    .INIT('b00000000000000000000000000000000)  
) GTP_LUT5M_inst (  
    .Z(), // OUTPUT  
    .I0(),// INPUT  
    .I1(),// INPUT  
    .I2(),// INPUT  
    .I3(),// INPUT  
    .I4(),// INPUT  
    .ID() // INPUT  
);
```

## Chapter 5 Usage Instructions for ROM & RAM-related GTPs

### 5.1 General Introduction

ROM & RAM related GTPs primarily implement various types of registers. GTP\_RAM family function as simple single/dual input port registers. GTP\_DRM family implements dual-port/simple dual-port/single-port registers, and three different write modes: NW, TW, and RBW modes. GTP\_FIFO implements registers that indicate empty/full status and provides a rewrite/resend function.

### 5.2 Usage Instructions for GTP\_ROM32X1

#### 5.2.1 Supported Devices

Table 5-1 Device Models That Support GTP\_ROM32X1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 5.2.2 Description of Functionality

GTP\_ROM32X1 is a storage ROM with an address depth of 32 bits and a data width of 1 bit. This structure block diagram is shown in the figure below.

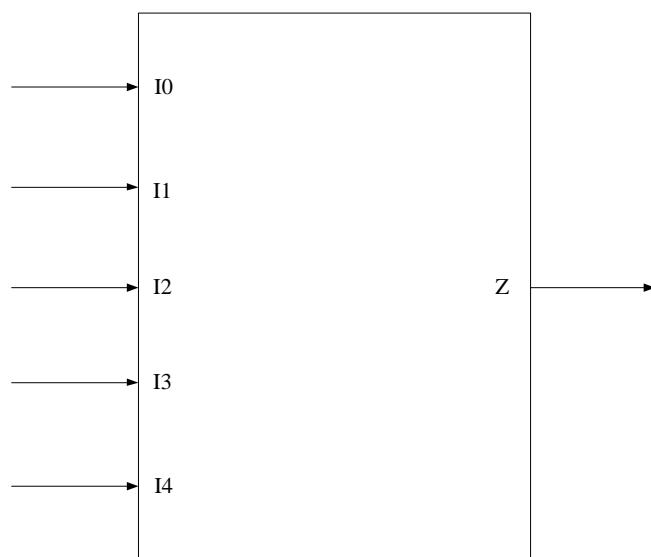


Figure 5-1 GTP\_ROM32X1 Structure Block Diagram

### 5.2.3 Port Description

Table 5-2 GTP\_ROM32X1 Port Description

Port	Orientation	Function Description
I0	Input	ROM read address addr[0]
I1	Input	ROM read address addr[1]
I2	Input	ROM read address addr[2]
I3	Input	ROM read address addr[3]
I4	Input	ROM read address addr[4]
Z	Output	Read data

### 5.2.4 Parameter Description

Table 5-3 GTP\_ROM32X1 Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
INIT	<binary>	32'h00000000~32'hfffffff	32'h00000000	ROM Initialization Configuration Parameters

### 5.2.5 Detailed Functional Description

This GTP implements the ROM storage function. Inputs I4~I0 form the data read address for reading the value of the specified bit of the ROM initial configuration parameters.

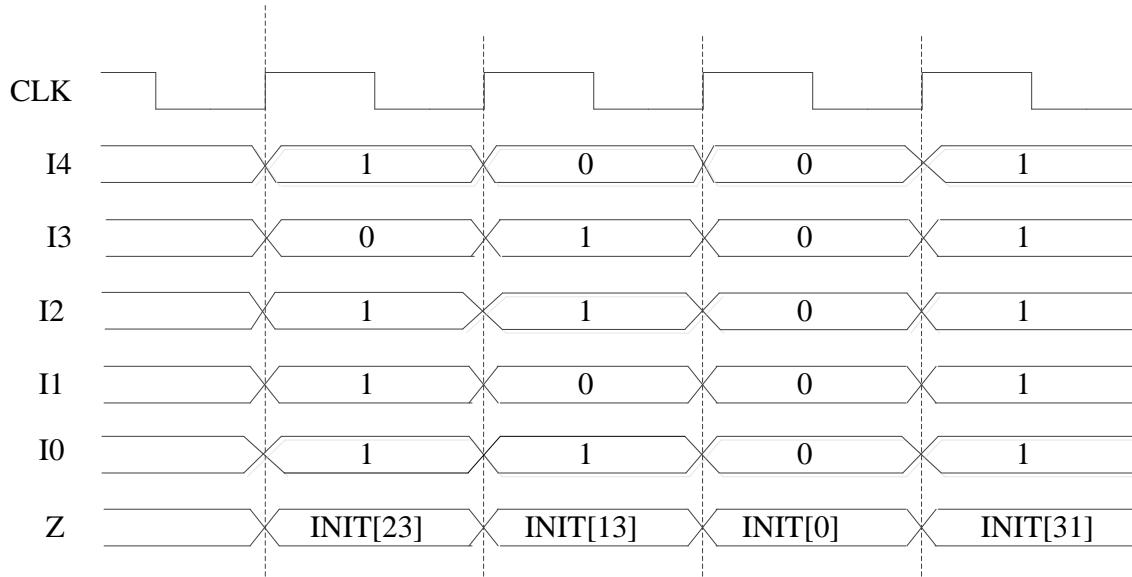


Figure 5-2 GTP\_ROM32X1 Waveform Diagram

### 5.2.6 Instantiation template

```
GTP_ROM32X1 #(
```

```
    .INIT      (32'h00000000)
```

)

GTP\_ROM32X1\_inst(

```

.I0      ( I0      ),
.I1      ( I1      ),
.I2      ( I2      ),
.I3      ( I3      ),
.I4      ( I4      ),
.Z       ( Z       )
);
```

### 5.3 Usage Instructions for GTP\_ROM64X1

#### 5.3.1 Supported Devices

Table 5-4 Device Models That Support GTP\_ROM64X1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 5.3.2 Description of Functionality

GTP\_ROM64X1 is a storage ROM with an address depth of 64 bits and a data width of 1 bit. This structure block diagram is shown in the figure below.

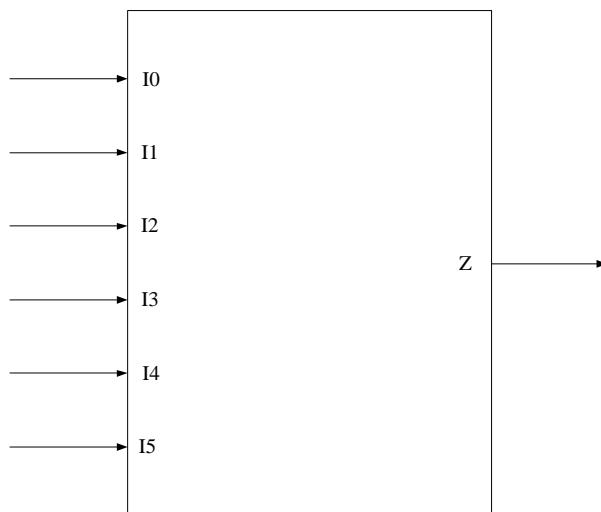


Figure 5-3 GTP\_ROM64X1 Structure Block Diagram

### 5.3.3 Port Description

Table 5-5 GTP\_ROM64X1 Port Description

Port	Orientation	Function Description
I0	Input	ROM read address addr[0]
I1	Input	ROM read address addr[1]
I2	Input	ROM read address addr[2]
I3	Input	ROM read address addr[3]
I4	Input	ROM read address addr[4]
I5	Input	ROM read address addr[5]
Z	Output	Read data

### 5.3.4 Parameter Description

Table 5-6 GTP\_ROM64X1 Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
INIT	<binary>	64'h00000000_00000000~64'hfffffff_ff ffffff	64'h00000000_00000000	ROM Initialization Configuration Parameters

### 5.3.5 Detailed Functional Description

This GTP implements the ROM storage function. Inputs I4~I0 form the data read address for reading the value of the specified bit of the ROM initial configuration parameters.

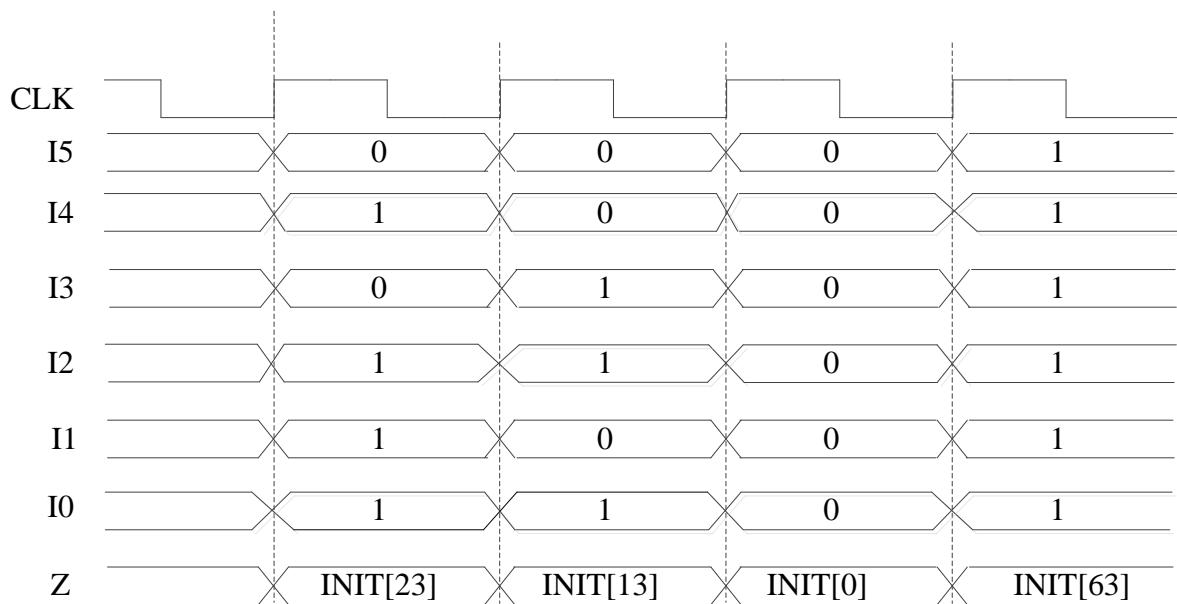


Diagram 5-4 GTP\_ROM64X1 Waveform Diagram

### 5.3.6 Instantiation template

```
GTP_ROM64X1#(
    .INIT      (64'h00000000_00000000)
)
GTP_ROM64X1_inst(
    .I0      ( I0      ),
    .I1      ( I1      ),
    .I2      ( I2      ),
    .I3      ( I3      ),
    .I4      ( I4      ),
    .I5      ( I5      ),
    .Z       ( Z       )
);
```

## 5.4 Usage Instructions for GTP\_ROM128X1

### 5.4.1 Supported Devices

Table 5-7 Device Models That Support GTP\_ROM128X1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 5.4.2 Description of Functionality

GTP\_ROM128X1 is a storage ROM with an address depth of 128 bits and a data width of 1 bit. This structure block diagram is shown in the figure below.

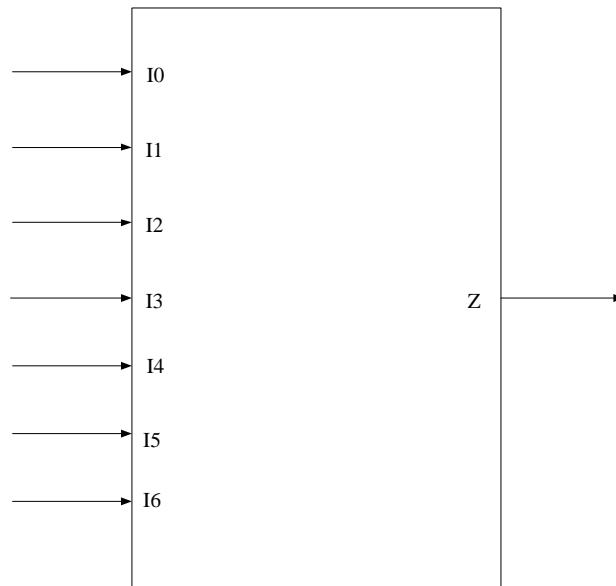


Figure 5-5 GTP\_ROM128X1 Structure Block Diagram

#### 5.4.3 Port Description

Table 5-8 GTP\_ROM128X1 Port Description

Port	Orientation	Function Description
I0	Input	ROM read address addr[0]
I1	Input	ROM read address addr[1]
I2	Input	ROM read address addr[2]
I3	Input	ROM read address addr[3]
I4	Input	ROM read address addr[4]
I5	Input	ROM read address addr[5]
I6	Input	ROM read address addr[6]
Z	Output	Read data

#### 5.4.4 Parameter Description

Table 5-9 GTP\_ROM128X1 Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
INIT	<binary>	128'h00000000_00000000_00000000_00000 000~128'hfffffff_ffffffff_ffffffff_ffffffff	128'h00000000_00000 000_00000000_00000 000	ROM Initialization Configuration Parameters

#### 5.4.5 Detailed Functional Description

This GTP implements the ROM storage function. Inputs I4~I0 form the data read address for (UG020007, V1.7)

reading the value of the specified bit of the ROM initial configuration parameters.

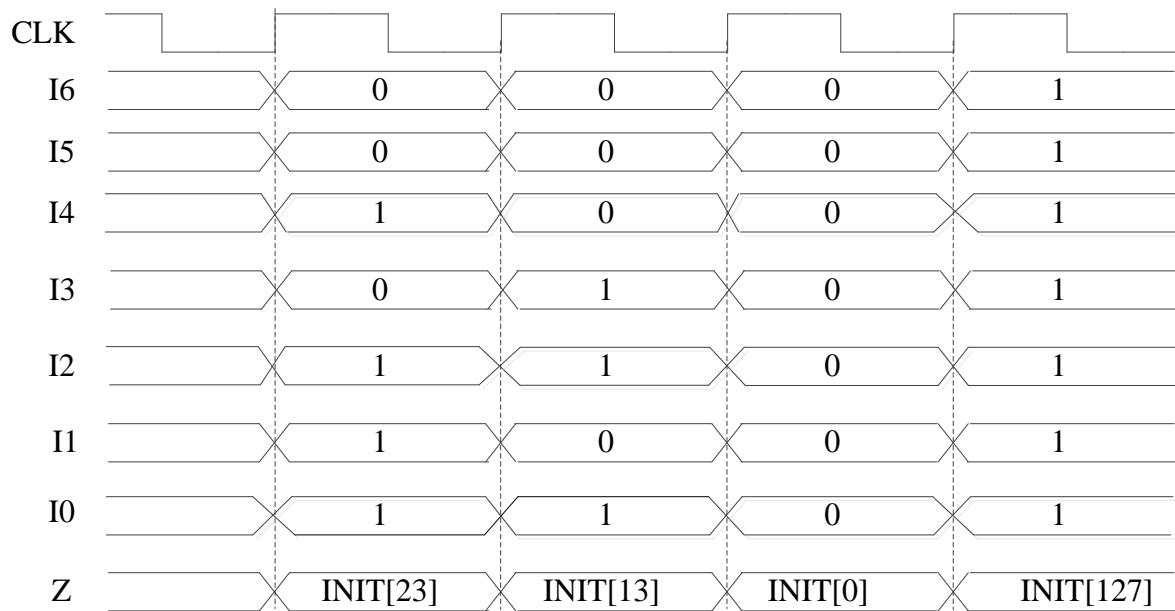


Figure 5-6 GTP\_ROM128X1 Waveform Diagram

#### 5.4.6 Instantiation template

```
GTP_ROM128X1 #(  
    .INIT      (128'h00000000_00000000_00000000_00000000)  
)  
  
GTP_ROM128X1_inst(  
    .I0        (  I0      ),  
    .I1        (  I1      ),  
    .I2        (  I2      ),  
    .I3        (  I3      ),  
    .I4        (  I4      ),  
    .I5        (  I5      ),  
    .I6        (  I6      ),  
    .Z         (  Z      )  
);
```

## 5.5 Usage Instructions for GTP\_ROM256X1

### 5.5.1 Supported Devices

Table 5-10 Device Models That Support GTP\_ROM256X1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 5.5.2 Description of Functionality

GTP\_ROM256X1 is a storage ROM with an address depth of 256 bits and a data width of 1 bit. This structure block diagram is shown in the figure below.

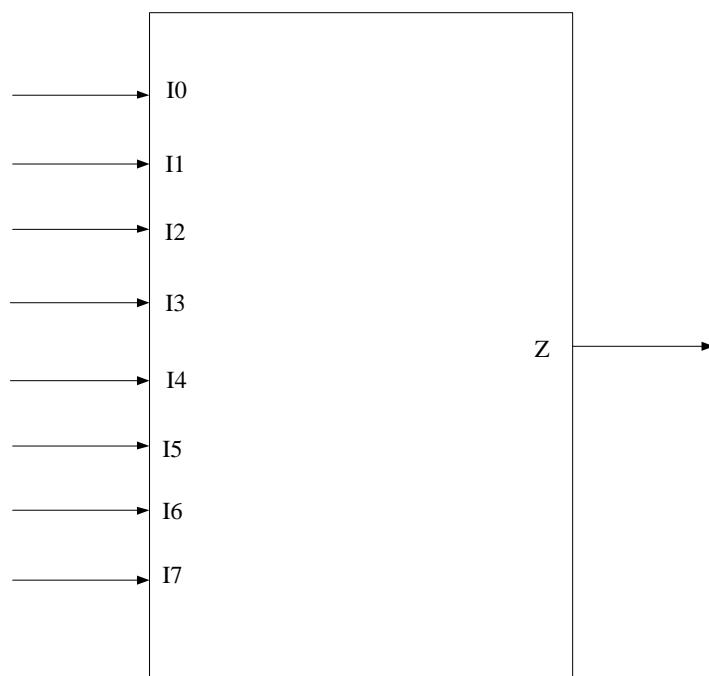


Figure 5-7 GTP\_ROM256X1 Structure Block Diagram

### 5.5.3 Port Description

Table 5-11 GTP\_ROM256X1 Port Description

Port	Orientation	Function Description
I0	Input	ROM read address addr[0]
I1	Input	ROM read address addr[1]
I2	Input	ROM read address addr[2]
I3	Input	ROM read address addr[3]
I4	Input	ROM read address addr[4]
I5	Input	ROM read address addr[5]
I6	Input	ROM read address addr[6]

Port	Orientation	Function Description
I7	Input	ROM read address addr[7]
Z	Output	Read data

### 5.5.4 Parameter Description

Table 5-12 GTP\_ROM256X1 Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
INIT	<binary>	256'h00000000_00000000_00000000_00000000 00000000_00000000_00000000_00000000 00_00000000~256'hffffffffff_ffffffff_ffffff ff_ffffffff_ffffffff_ffffffff_ffffffff_ffffffff	256'h00000000_00000000 00_00000000_00000000 0_00000000_00000000 _00000000_00000000	ROM Initialization Configuration Parameters

### 5.5.5 Detailed Functional Description

This GTP implements the ROM storage function. Inputs I4~I0 form the data read address for reading the value of the specified bit of the ROM initial configuration parameters.

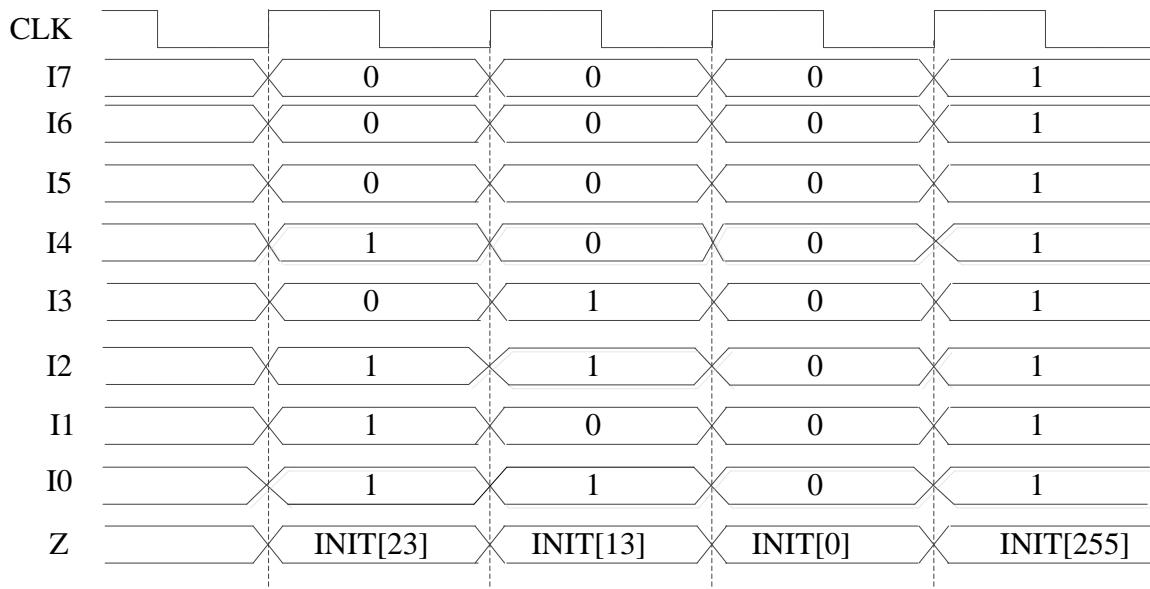


Figure 5-8 GTP\_ROM256X1 Waveform Diagram

### 5.5.6 Instantiation template

```
GTP_ROM256X1 #(  
    .INIT  
    (256'h00000000_00000000_00000000_00000000_00000000_00000000_00000000)  
)  
GTP_ROM256X1_inst(
```

```

.I0      ( I0      ),
.I1      ( I1      ),
.I2      ( I2      ),
.I3      ( I3      ),
.I4      ( I4      ),
.I5      ( I5      ),
.I6      ( I6      ),
.I7      ( I7      ),
.Z      ( Z      )
);

```

## 5.6 Usage Instructions for GTP\_RAM16X4SP

### 5.6.1 Supported Devices

Table 5-13 Device Models That Support GTP\_RAM16X4SP

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Not supported	Not supported	Not supported	Not supported

### 5.6.2 Description of Functionality

GTP\_RAM16X4SP is a single-port random storage module with a data width of 4 bits and an address depth of 16 bits, with the same read/write address. This structure block diagram is shown in the figure below.

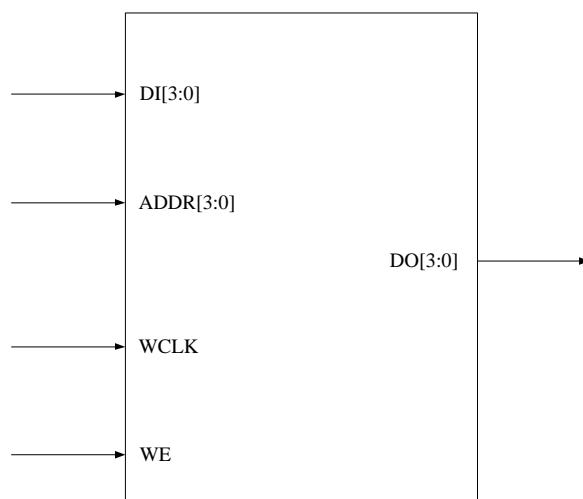


Figure 5-9 GTP\_RAM16X4SP Structure Block Diagram

### 5.6.3 Port Description

Table 5-14 GTP\_RAM16X4SP Port Description

Port	Orientation	Function Description
DI	Input	WRITE DATA
ADDR	Input	Read/write address
WCLK	Input	Write clock
WE	Input	Write enable
DO	Output	Read data

### 5.6.4 Parameter Description

Table 5-15 GTP\_RAM16X4SP Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
INIT_0	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters
INIT_1	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters
INIT_2	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters
INIT_3	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters

### 5.6.5 Detailed Functional Description

This GTP functions as a single-port RAM, writing only when WE is high-level. When WE is high-level, the input data DI is written to the current address ADDR on the rising edge of WCLK and read from the DO port. When WE is low-level, the data of the current address ADDR is read from the DO port on the rising edge of WCLK without any writing. Its waveform is shown in the following figure.

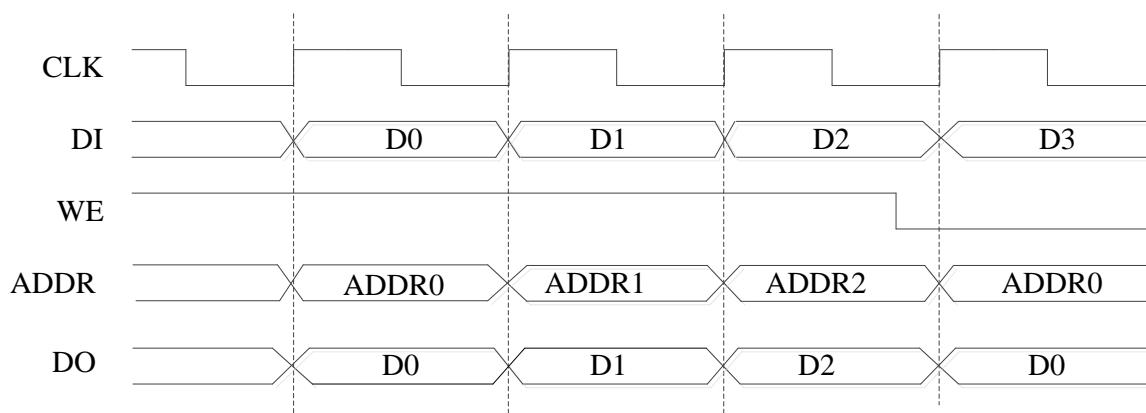


Figure 5-10 GTP\_RAM16X4SP Waveform Diagram

### 5.6.6 Instantiation template

```

GTP_RAM16X4SP #(

    .INIT_0      (16'h0000),
    .INIT_1      (16'h0000),
    .INIT_2      (16'h0000),
    .INIT_3      (16'h0000)

)

GTP_RAM16X4SP_inst(
    .DO          ( DO ),
    .DI          ( DI ),
    .ADDR        ( ADDR ),
    .WCLK        ( WCLK ),
    .WE          ( WE )
);

```

## 5.7 Usage Instructions for GTP\_RAM16X4DP

### 5.7.1 Supported Devices

Table 5-16 Device Models That Support GTP\_RAM16X4DP

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Not supported	Not supported	Not supported	Not supported

### 5.7.2 Description of Functionality

GTP\_RAM16X4DP is a dual-port random storage module with a data width of 4 bits and an address depth of 16 bits, with read/write addresses input through two separate ports. This structure block diagram is shown in the figure below.

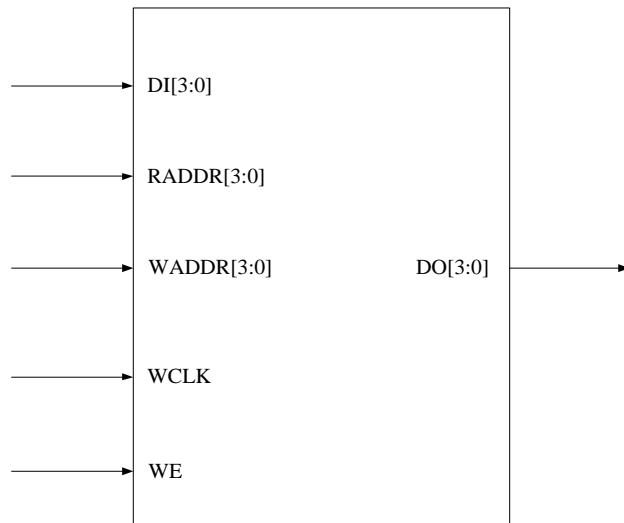


Figure 5-11 GTP\_RAM16X4DP Structure Block Diagram

### 5.7.3 Port Description

Table 5-17 GTP\_RAM16X4DP Port Description

Port	Orientation	Function Description
DI	Input	WRITE DATA
RADDR	Input	Read Address
WADDR	Input	Write Address
WCLK	Input	Write clock
WE	Input	Write enable
DO	Output	Read data

### 5.7.4 Parameter Description

Table 5-18 GTP\_RAM16X4DP Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Function Description
INIT_0	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters
INIT_1	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters
INIT_2	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters
INIT_3	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters

### 5.7.5 Detailed Functional Description

This GTP serves as a dual-port RAM, with read/write addresses being independent. Data is written to RAM only when WE is high-level. When WE is high-level, input data DI is written to the write

address WADDR on the rising edge of WCLK, and data from the read address RADDR is read from the DO port. When WE is low-level, data from the read address RADDR is read from the DO port on the rising edge of WCLK, without any write operation.

Read and write operations are asynchronous and read/write addresses cannot be the same simultaneously, otherwise it will cause read-write conflict. Its waveform is shown in the following figure.

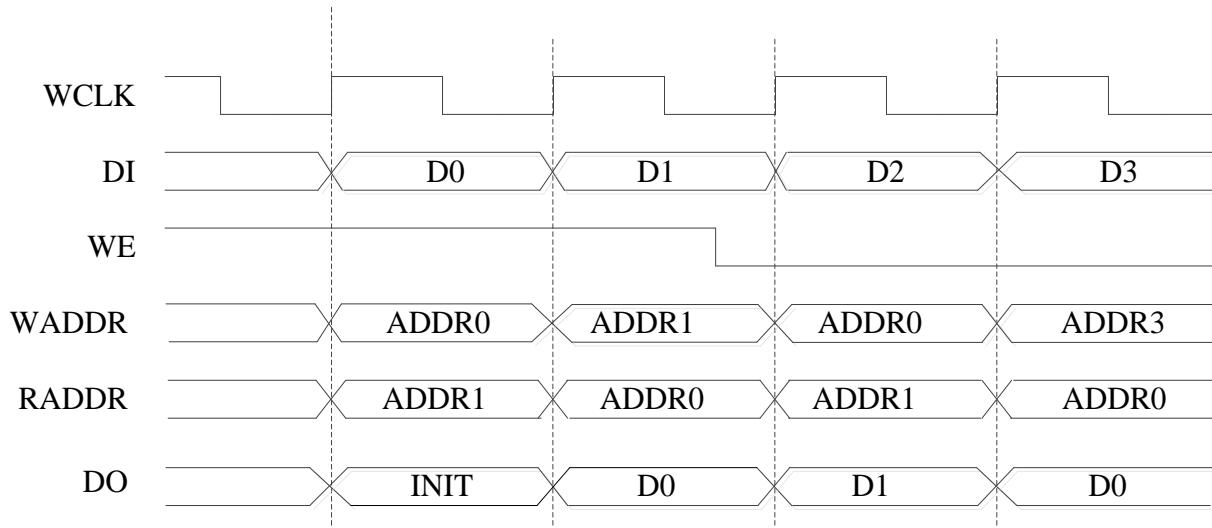


Figure 5-12 GTP\_RAM16X4DP Waveform Diagram

### 5.7.6 Instantiation template

```
GTP_RAM16X4DP #(  
    .INIT_0      (16'h0000),  
    .INIT_1      (16'h0000),  
    .INIT_2      (16'h0000),  
    .INIT_3      (16'h0000)  
)  
  
GTP_RAM16X4DP_inst(  
    .DO          (    DO    ),  
    .DI          (    DI    ),  
    .RADDR       (    RADDR  ),  
    .WADDR       (    WADDR  ),  
    .WCLK        (    WCLK  ),  
    .WE          (    WE    )  
);
```

## 5.8 Usage Instructions for GTP\_RAM16X1SP

### 5.8.1 Supported Devices

Table 5-19 Device Models That Support GTP\_RAM16X1SP

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 5.8.2 Description of Functionality

GTP\_RAM16X1SP is a single-port random storage module with a data width of 1 bit and an address depth of 16 bits, with the same read/write addresses. This structure block diagram is shown in the figure below.

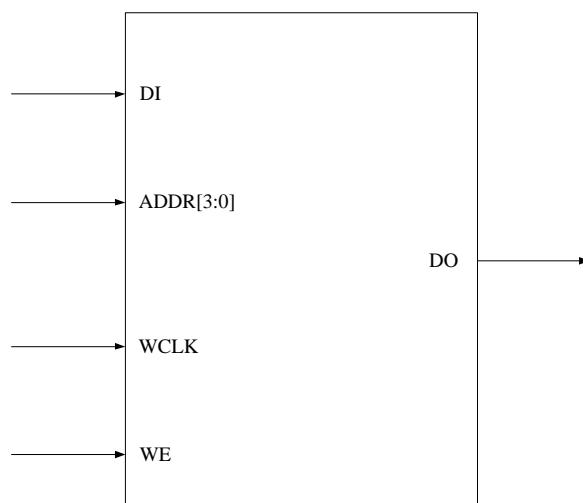


Figure 5-13 GTP\_RAM16X1SP Structure Block Diagram

### 5.8.3 Port Description

Table 5-20 GTP\_RAM16X1SP Port Description

Port	Direction	Description
DI	Input	WRITE DATA
ADDR	Input	Read/write address
WCLK	Input	Write clock
WE	Input	Write enable
DO	Output	Read data

#### 5.8.4 Parameter Description

Table 5-21 GTP\_RAM16X4SP Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Description
INIT	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters

#### 5.8.5 Detailed Functional Description

This GTP functions as a single-port RAM, writing only when WE is high-level. When WE is high-level, the input data DI is written to the current address ADDR on the rising edge of WCLK and read from the DO port. When WE is low-level, data from the current address ADDR is read from the DO port on the rising edge of WCLK, without a write operation. Its waveform is shown in the following figure.

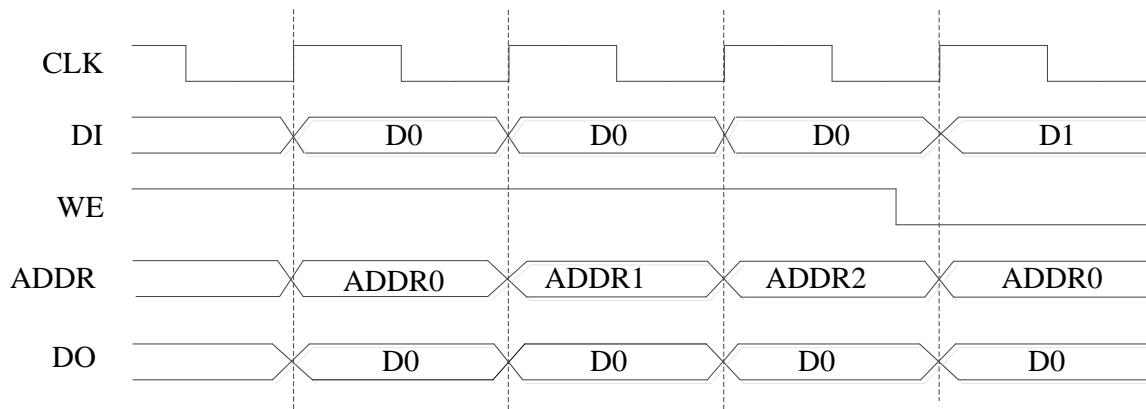


Figure 5-14 GTP\_RAM16X1SP Waveform Diagram

#### 5.8.6 Instantiation template

```
GTP_RAM16X1SP#(
    .INIT      (16'h0000),
    )
GTP_RAM16X1SP_inst(
    .DO        (DO),
    .DI        (DI),
    .ADDR     (ADDR),
    .WCLK     (WCLK),
    .WE        (WE)
);
```

## 5.9 Usage Instructions for GTP\_RAM16X1DP

### 5.9.1 Supported Devices

Table 5-22 Device Models That Support GTP\_RAM16X1DP

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 5.9.2 Description of Functionality

GTP\_RAM16X1DP is a dual-port random storage module with a data width of 1 bit and an address depth of 16 bits, with read/write addresses input through two separate ports. This structure block diagram is shown in the figure below.

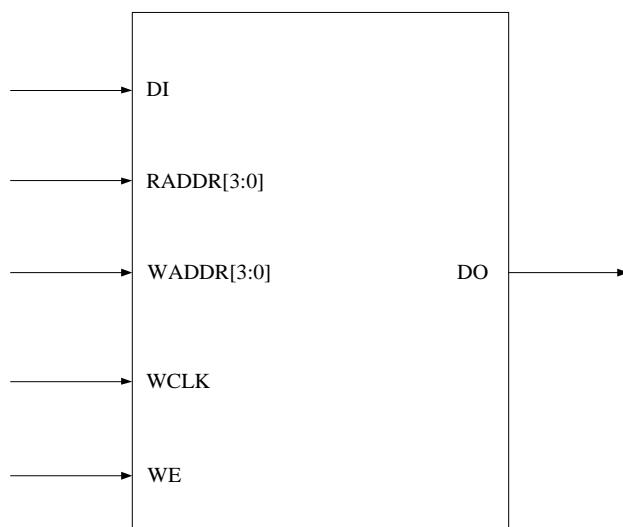


Figure 5-15 GTP\_RAM16X1DP Structure Block Diagram

### 5.9.3 Port Description

Table 5-23 GTP\_RAM16X1DP Port Description

Port	Direction	Description
DI	Input	WRITE DATA
RADDR	Input	Read Address
WADDR	Input	Write Address
WCLK	Input	Write clock
WE	Input	Write enable
DO	Output	Read data

### 5.9.4 Parameter Description

Table 5-24 GTP\_RAM16X1DP Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Description
INIT	<binary>	16'h0000~16'hffff	16'h0000	Memory initialization configuration parameters

### 5.9.5 Detailed Functional Description

This GTP serves as a dual-port RAM, with read/write addresses being independent. Data is written to RAM only when WE is high-level. When WE is high-level, input data DI is written to the write address WADDR on the rising edge of WCLK, and data from the read address RADDR is read from the DO port. When WE is low-level, data from the read address RADDR is read from the DO port on the rising edge of WCLK, without any write operation.

Read and write operations are asynchronous and read/write addresses cannot be the same simultaneously, otherwise it will cause read-write conflict. Its waveform is shown in the following figure.

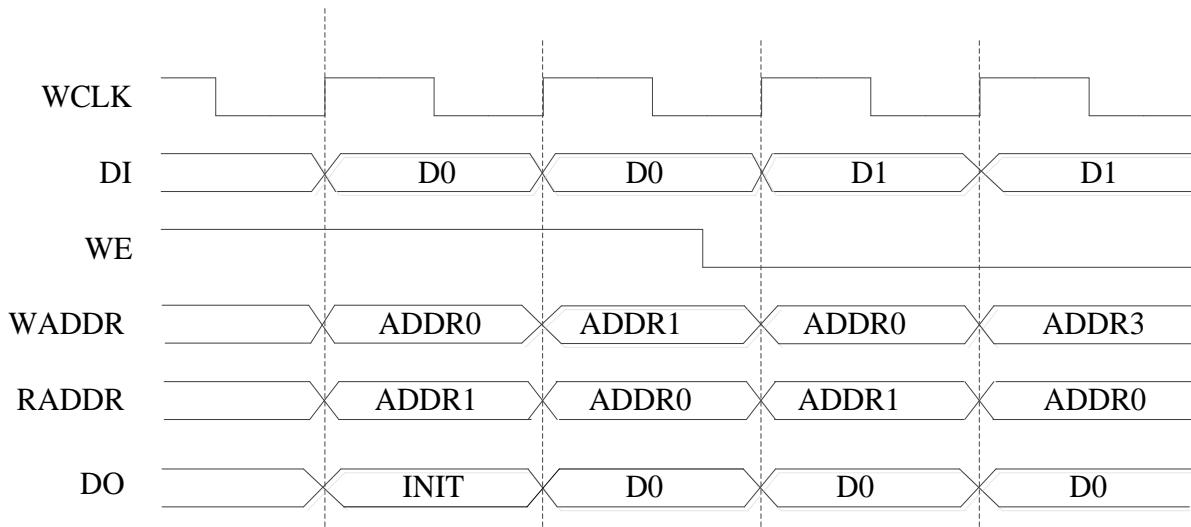


Figure 5-16 GTP\_RAM16X1DP Waveform Diagram

### 5.9.6 Instantiation template

```
GTP_RAM16X1DP #(
    .INIT      (16'h0000),
    )
GTP_RAM16X1DP_inst(
    .DO        (    DO     ),
```

```

.DI      (  DI   ),
.RADDR  (  RADDR  ),
.WADDR  (  WADDR  ),
.WCLK   (  WCLK  ),
.WE     (  WE   )
);

);

```

## 5.10 Usage Instructions for GTP\_RAM32X1SP

### 5.10.1 Supported Devices

Table 5-25 Device Models That Support GTP\_RAM32X1SP

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Not supported	Supported	Supported	Supported	Supported

### 5.10.2 Description of Functionality

GTP\_RAM32X1SP is a dual-port random storage module with a data width of 1 bit and an address depth of 32 bits, with read/write addresses input through two separate ports. This structure block diagram is shown in the figure below.

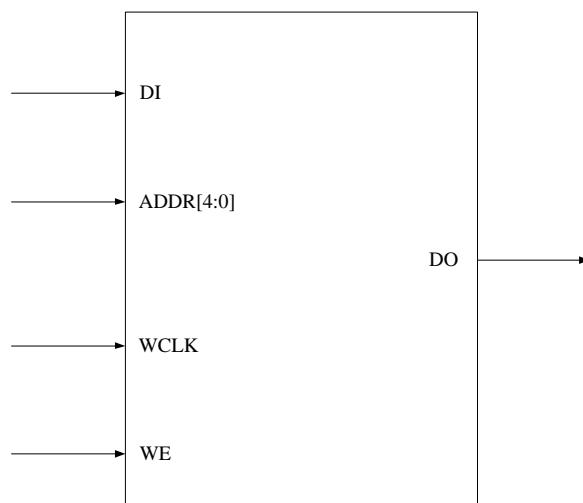


Diagram 5-17 GTP\_RAM32X1SP Structure Block Diagram

### 5.10.3 Port Description

Table 5-26 GTP\_RAM32X1SP Port Description

Port	Direction	Description
DI	Input	WRITE DATA
ADDR	Input	Write Address
WCLK	Input	Write clock
WE	Input	Write enable
DO	Output	Read data

### 5.10.4 Paramater Description

Table 5-27 GTP\_RAM32X1SP Parameter Description

Parameter Name	Parameter Type	Setting Value	Defaults	Description
INIT	<binary>	32'h00000000~32'hfffffff	32'h00000000	Memory initialization configuration parameters

### 5.10.5 Detailed Functional Description

This GTP functions as a single-port RAM, writing only when WE is high-level. When WE is high-level, the input data DI is written to the current address ADDR on the rising edge of WCLK and read from the DO port. When WE is low-level, data from the current address ADDR is read from the DO port on the rising edge of WCLK, without a write operation. Its waveform is shown in the following figure.

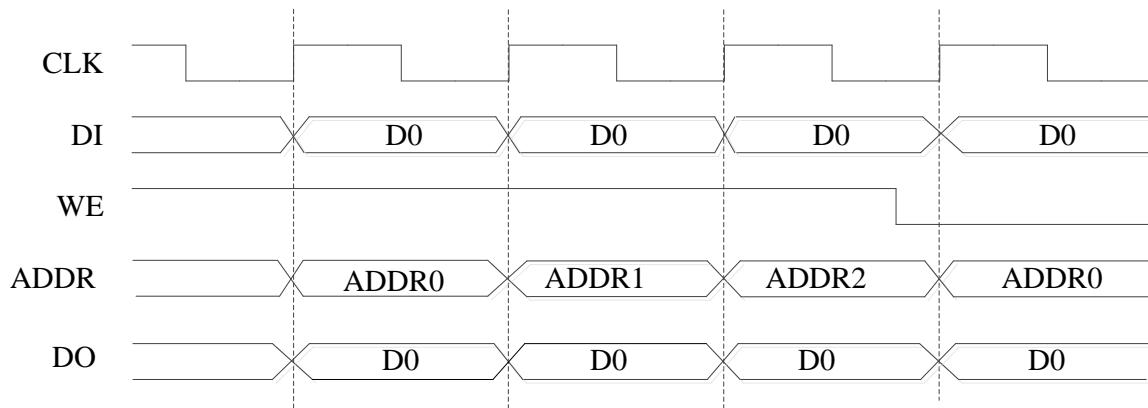


Diagram 5-18 GTP\_RAM32X1SP Waveform Diagram

### 5.10.6 Instantiation template

GTP\_RAM32X1SP#(

```

.INIT      (32'h00000000)
)

GTP_RAM32X1SP_inst(
    .DO       ( DO      ),
    .DI       ( DI      ),
    .ADDR     ( ADDR   ),
    .WCLK     ( WCLK   ),
    .WE       ( WE      )
);

);

```

## 5.11 Usage Instructions for GTP\_RAM32X1DP

### 5.11.1 Supported Devices

Table 5-28 Device Models That Support GTP\_RAM32X1DP

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Not supported	Supported	Supported	Supported	Supported

### 5.11.2 Description of Functionality

GTP\_RAM32X1DP is a dual-port random storage module with a data width of 1 bit and an address depth of 32 bits, with read/write addresses input through two separate ports. This structure block diagram is shown in the figure below.

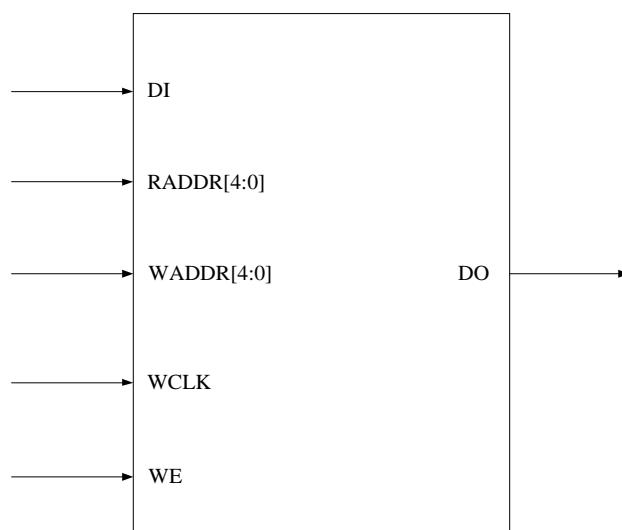


Diagram 5-19 GTP\_RAM32X1DP Structure Block Diagram

### 5.11.3 Port Description

Table 5-29 GTP\_RAM32X1DP Port Description

<b>Port</b>	<b>Direction</b>	<b>Description</b>
DI	Input	WRITE DATA
RADDR	Input	Read Address
WADDR	Input	Write Address
WCLK	Input	Write clock
WE	Input	Write enable
DO	Output	Read data

### 5.11.4 Parameter Description

Table 5-30 GTP\_RAM32X1DP Parameter Description

<b>Parameter Name</b>	<b>Parameter Type</b>	<b>Setting Value</b>	<b>Defaults</b>	<b>Description</b>
INIT	<binary>	32'h00000000~32'hffffffff	32'h00000000	Memory initialization configuration parameters

### 5.11.5 Detailed Functional Description

This GTP serves as a dual-port RAM, with read/write addresses being independent. Data is written to RAM only when WE is high-level. When WE is high-level, input data DI is written to the write address WADDR on the rising edge of WCLK, and data from the read address RADDR is read from the DO port. When WE is low-level, data from the read address RADDR is read from the DO port on the rising edge of WCLK, without any write operation.

Read and write operations are asynchronous and read/write addresses cannot be the same simultaneously, otherwise it will cause read-write conflict. Its waveform is shown in the following figure.

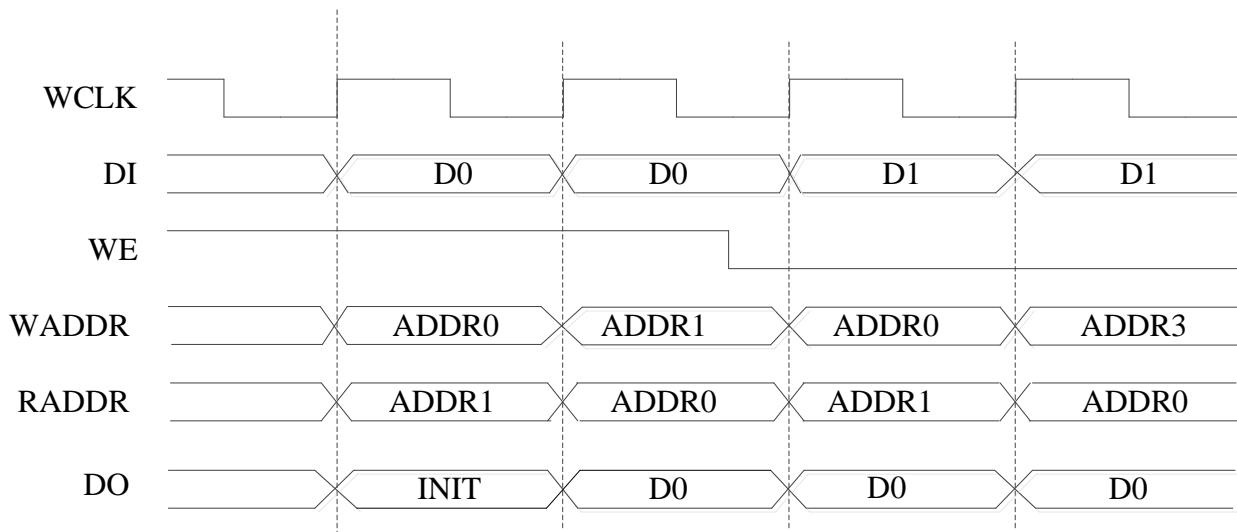


Diagram 5-20 GTP\_RAM32X1DP Waveform Diagram

### 5.11.6 Instantiation template

```

GTP_RAM32X1DP #(

    .INIT      (32'h00000000)

);

GTP_RAM32X1DP_inst(
    .DO        (),
    .DI        (),
    .RADDR    (),
    .WADDR    (),
    .WCLK    (),
    .WE        ()
);
    
```

## 5.12 Usage Instructions for GTP\_DRM9K

### 5.12.1 Supported Devices

Table 5-31 Device Models That Support GTP\_DRM9K

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 5.12.2 Description of Functionality

GTP\_DRM9K has a 9K bits storage unit, supporting multiple working modes, including DP (True Dual Port) RAM, SDP (Simple Dual Port) RAM, SP (Single Port) RAM or ROM mode. DRM supports configurable data widths and supports dual-port mixed data widths in DP RAM and SDP RAM modes.

The DP RAM module has two independent ports, A and B, both of which can perform read and write operations independently, allowing simultaneous read or write, or one port to read while the other to write. Both ports support different clocks, with the bit width set independently. Simultaneous read and write operations on the same address via both ports are not possible, as this would cause a conflict.

In SDP RAM, port A is fixed as the write port and port B as the read port; both ports support different clocks, with the bit width set independently. Simultaneous read and write operations on the same address via both ports are not possible, as this would cause a conflict.

In SP RAM mode, the two ports of the DRM can be operated independently. Simultaneous read and write operations on the same address via both ports are not possible, as this would cause a conflict.

The DRM can be configured as ROM, with ROM contents initialized through the configuration interface. Since there is no built-in logic to control the ROM mode, write operations must be disabled by the PDS software.

The DRM's port write operations support three modes: Normal Write mode (NW), Transparent Write mode (TW), and Read before Write mode (RBW).

All ports are active-high by default.

The block diagram of GTP\_DRM9K structure is shown in the figure below.

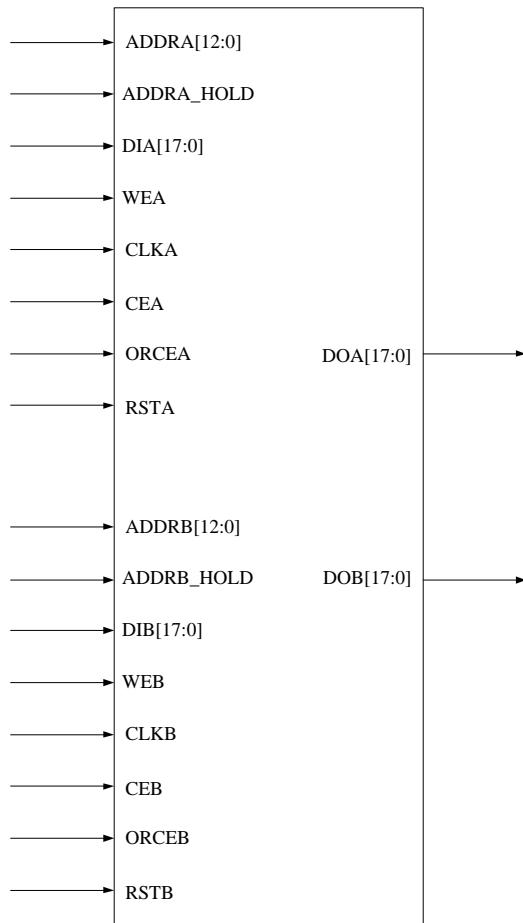


Figure 5-21 GTP\_DRM9K Block Diagram

### 5.12.3 Port Description

Table 5-32 GTP\_DRM9K Port Description

Port	Direction	Description
ADDRA	Input	Port A input address
ADDRA_HOLD	Input	Port A address input selection
DIA	Input	Port A data input
WEA	Input	Port A write enable
CLKA	Input	Port A clock
CEA	Input	Port A clock enable
ORCEA	Input	Port A output register enable
RSTA	Input	Port A data register reset
DOA	Output	Port A data output
ADDRB	Input	Port B input address
ADDRB_HOLD	Input	Port B address input selection
DIB	Input	Port B data input
WEB	Input	Port B write enable
CLKB	Input	Port B clock

Port	Direction	Description
CEB	Input	Port B clock enable
ORCEB	Input	Port B output register enable
RSTB	Input	Port B data register reset
DOB	Output	Port B data output

#### 5.12.4 Paramater Description

Table 5-33 GTP\_DRM9K Parameter Description

Parameter	Description	Setting Value
DATA_WIDTH_A	Maximum data width of Port A	1, 2, 4, 8, 16, 32, 9, 18, 36
DATA_WIDTH_B	Maximum data width of Port B	1, 2, 4, 8, 16, 32, 9, 18, 36
WRITE_MODE_A	Port A write mode	"NORMAL_WRITE" "TRANSPARENT_WRITE" "READ_BEFORE_WRITE"
WRITE_MODE_B	Port B write mode	"NORMAL_WRITE" "TRANSPARENT_WRITE" "READ_BEFORE_WRITE"
DOA_REG	Port A output register	0 = Do not enable output register 1 = Enable output register
DOB_REG	Port B output register	0 = Do not enable output register 1 = Enable output register
RST_TYPE	Reset mode selection	"SYNC": Synchronous Reset "ASYNC": Asynchronous Reset "ASYNC_SYNC_RELEASE": Asynchronous Reset with Synchronous Release
RAM_MODE	RAM mode selection	"TRUE_DUAL_PORT": True Dual-Port RAM "SIMPLE_DUAL_PORT": Simple Dual-Port RAM "SINGLE_PORT": Single-Port RAM ROM: ROM
GRS_EN	Global reset enable signal (Internal Chip)	"FALSE": Global Reset Not Enabled; "TRUE": Global Reset enabled.
DOA_REG_CLKINV	Port A output register clock inversion	0 = Clock not inverted 1 = Clock inverted
DOB_REG_CLKINV	Port B output register clock inversion	0 = Clock not inverted 1 = Clock inverted
INIT_00 INIT_01 INIT_02 ... INIT_1F	RAM Initialization Configuration Parameters	0 ~2^288-1
INIT_FILE	Initialization files	"NONE": If no initialization file is specified, the default initialization data is all zeros; "XXX": XXX represents the specific initialization file path
BLOCK_X	Data cascade coordinates when DRM18K is cascaded	Depends on the number of cascaded DRMs
BLOCK_Y	Address cascade coordinates when RAM18K is cascaded	Depends on the number of cascaded DRMs
RAM_DATA_WIDTH	Maximum data width after RAM is cascaded	Depends on the number of cascaded DRMs

Parameter	Description	Setting Value
RAM_ADDR_WIDTH	Maximum address width after RAM is cascaded	Depends on the number of cascaded DRMs
INIT_FORMAT	Initialization file format	"BIN": Binary "HEX": Hexadecimal
SIM_DEVICE	Supported Devices	"PGL22G": PGL22G, PGL22GS "LOGOS": Subsequent devices in the PGL Family

Note: The following conditions are not supported simultaneously:

SIM\_DEVICE = "PGL22G"

RAM\_MODE = "TRUE\_DUAL\_PORT"

WRITE\_MODE\_A = "READ\_BEFORE\_WRITE"

WRITE\_MODE\_B = "READ\_BEFORE\_WRITE"

## 5.12.5 Detailed Functional Description

### 5.12.5.1 Write Mode---NW Mode

NW mode is Normal Write mode, which is the default mode. When the write enable signal WEA is high-level, no read operation is performed, and the input data DIA is written to the current address. When the write enable signal WEA is low-level, the data saved in the current address ADDRA is read and output from the DOA port in the next clock cycle. Its waveform is shown in the following figure.

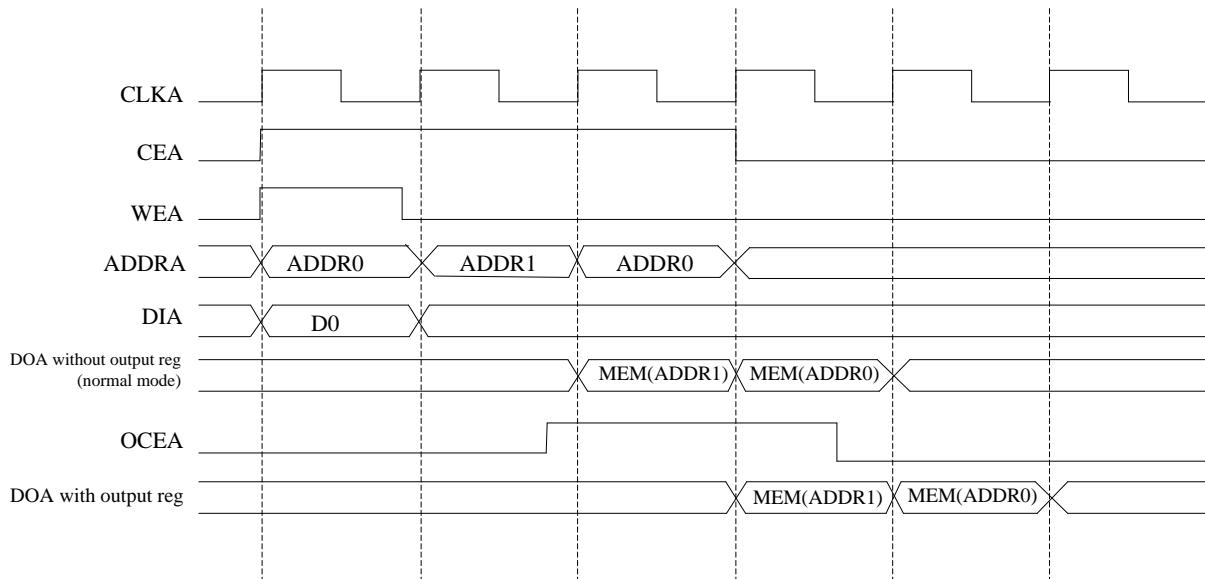


Figure 5-22 GTP\_DRM9K NW Mode Waveform Diagram

Notes:

1. The port write operating mode of DRM is only applicable to ports with read/write functions, i.e., the A/B ports of DP RAM and the ports of SP RAM.
2. SDP mode does not support write mode setting. It is disabled to set to TW or RBW mode during use; it can only be set to the default NW mode, or DRM will enter an abnormal write operation state.

### 5.12.5.2 Write Mode---TW Mode

In TW mode, when the write enable signal WEA is high-level, the input data DIA is written to the address ADDRA and read directly from the DOA port in the next clock cycle. When the enable signal WEA is low-level, the data saved in the current address ADDRA is read and output from the DOA port in the next clock cycle. Its waveform is shown in the following figure.

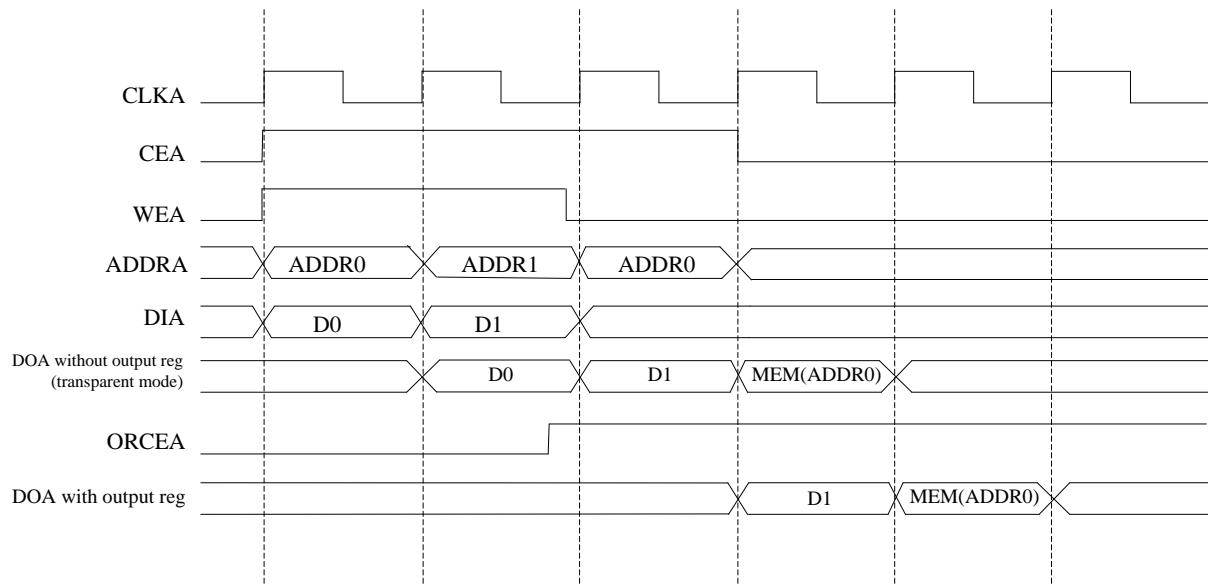


Figure 5-23 GTP\_DRM9K TW Mode Waveform Diagram

### 5.12.5.3 Write Mode---RBW Mode

In RBW mode, when the write enable signal WEA is high-level, the original data referenced by the address ADDRA is read firstly and output to the output port in the next clock cycle of the write operation, and then the input data DIA is written into the address ADDRA to overwrite the original data. When the enable signal WEA is low-level, the data saved in the current address ADDRA is read and output from the DOA port in the next clock cycle. Its waveform is shown in the following figure.

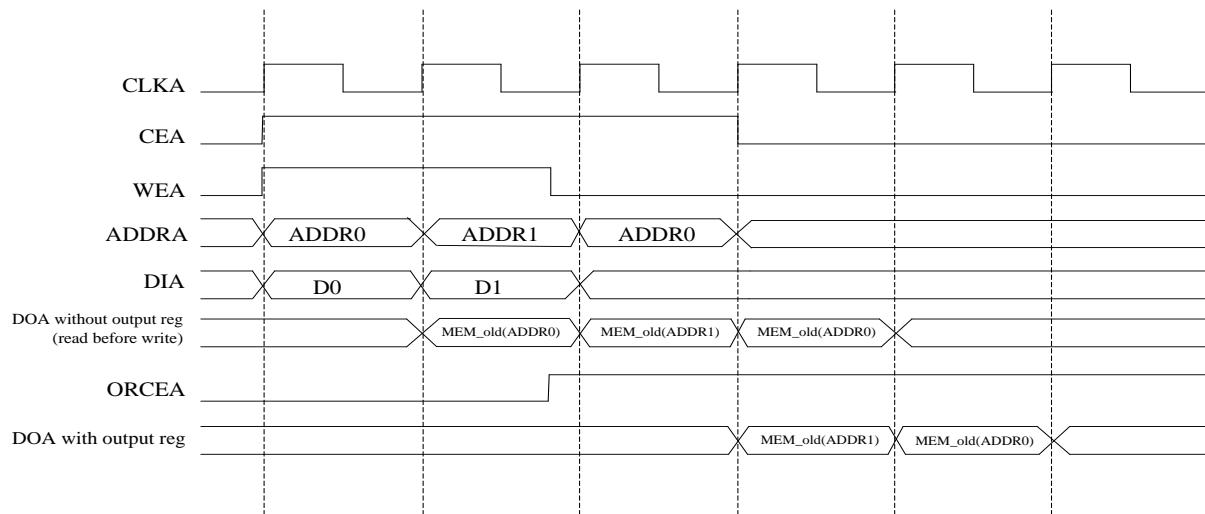


Figure 5-24 GTP\_DRM9K RBW Mode Waveform Diagram

#### 5.12.5.4 Implementation of TW and RBW Modes for SP RAM

The TW (Transparent Write) and RBW (Read before Write) modes under 32/36 bits SP mode for DRM cannot be implemented by direct configuration (The setting of the write mode to TW or RBW mode is disabled when GTP is configured as 32/36 bit SP mode, or DRM will enter an abnormal write operation state). It requires additional configuration for its implementation, which is shown as follows:

5. Configure both Port A and Port B to 16/18 bits in DP mode (for 32/36 bits SP respectively);
6. Port A and Port B are configured to the same write mode; other configuration bits also need to be the same.
7. The CE, WE, ADDR, ORCE, RST, CLK, and other control signals of Port A and Port B are paralleled respectively;
8. Parallel DIA and DIB as data inputs; parallel DOA and DOB as data outputs;
9. Both Port A and Port B address input ports must be connected to the address input simultaneously; ADDRA[4] and ADDRB[4] must be connected to 0 or 1 respectively (in opposite).

This is shown in the following figure.

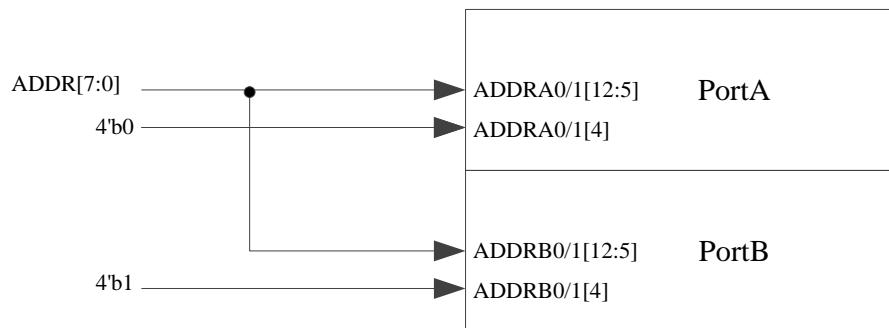


Figure 5-25 GTP\_DRM9K SP Interface Diagram

### 5.12.5.5 Output Register Mode

For data output ports, DRM provides an optional Output Register for improved timing performance. As shown in the figure below, the output register can be controlled by an independent enable signal OCE. When OCE is a constant 1, the use of the output register will increase the delay of the read operation from one clock cycle to two clock cycles (Case 1); in pipeline designs with flow control, the user can also flexibly control OCE with logic (Case 2).

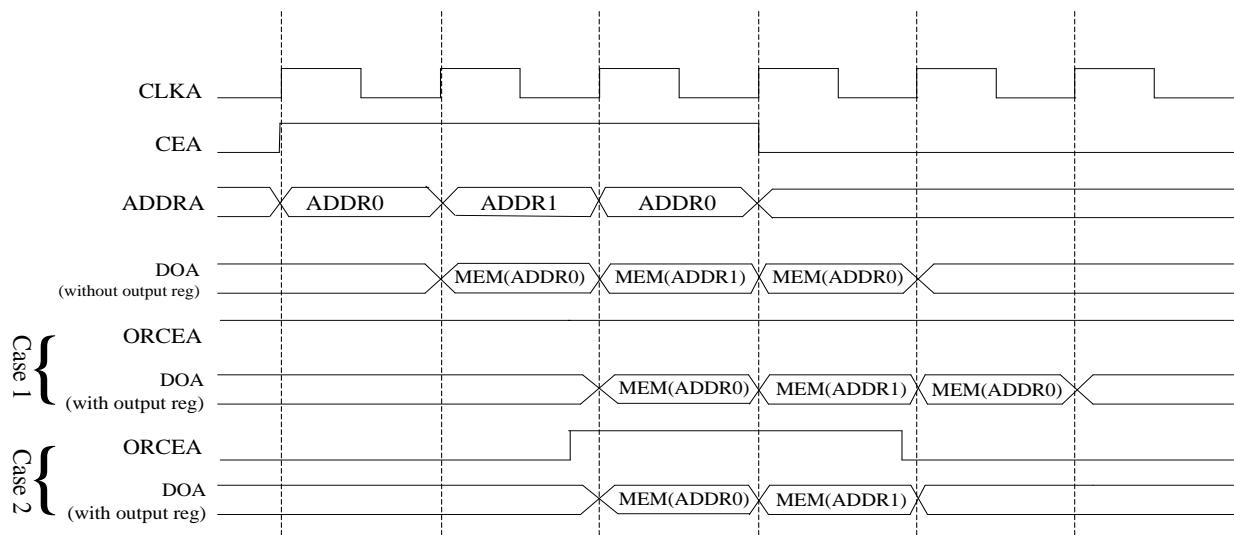


Figure 5-26 GTP\_DRM9K Register Mode Waveform Diagram

### 5.12.6 Instantiation template

#### GTP\_DRM9K

```

#(
    .DATA_WIDTH_A      ( 18 ), // 1 2 4 8 16 32 9 18 36
    .DATA_WIDTH_B      ( 18 ), // 1 2 4 8 16 32 9 18 36
    .WRITE_MODE_A      ( "NORMAL_WRITE" ),           // TRANSPARENT_WRITE
)

```







```
.ORCEB      (ORCEB      ),
.RSTB       (RSTB       )
);
```

## 5.13 Usage Instructions for GTP\_DRM18K

### 5.13.1 Supported Devices

Table 5-34 Device Models That Support GTP\_DRM18K

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 5.13.2 Description of Functionality

GTP\_DRM18K has a 18K bits storage unit, supporting multiple working modes, including DP (True Dual Port) RAM, SDP (Simple Dual Port) RAM, SP (Single Port) RAM or ROM mode. DRM supports configurable data widths and supports dual-port mixed data widths in DP RAM and SDP RAM modes.

The DP RAM module has two independent ports, A and B, both of which can perform read and write operations independently, allowing simultaneous read or write, or one port to read while the other to write. Both ports support different clocks, with the bit width set independently. Simultaneous read and write operations on the same address via both ports are not possible, as this would cause a conflict.

In SDP RAM, port A is fixed as the write port and port B as the read port; both ports support different clocks, with the bit width set independently. Simultaneous read and write operations on the same address via both ports are not possible, as this would cause a conflict.

In SP RAM mode, the two ports of the DRM can be operated independently. Simultaneous read and write operations on the same address via both ports are not possible, as this would cause a conflict.

The DRM can be configured as ROM, with ROM contents initialized through the configuration interface.

The DRM's port write operations support three modes: Normal Write mode (NW), Transparent Write mode (TW), and Read before Write mode (RBW).

All ports are active-high by default.

The block diagram of GTP\_DRM18K structure is shown in the figure below.

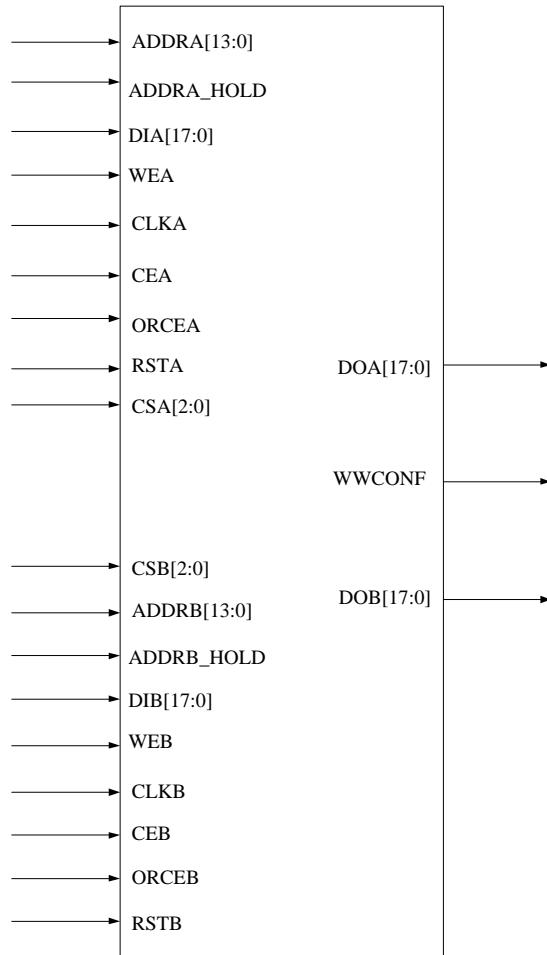


Figure 5-27 GTP\_DRM18K Structure Block Diagram

### 5.13.3 Port Description

Table 5-35 GTP\_DRM18K Port Description

Port	Direction	Description
ADDRA	Input	Port A input address
ADDRA_HOLD	Input	Port A address input selection
DIA	Input	Port A data input
CSA	Input	Port A address extension
WEA	Input	Port A write enable
CLKA	Input	Port A clock
CEA	Input	Port A clock enable
ORCEA	Input	Port A output register enable
RSTA	Input	Port A data register reset
DOA	Output	Port A data output
ADDRB	Input	Port B input address
ADDRB_HOLD	Input	Port B address input selection
DIB	Input	Port B data input

<b>Port</b>	<b>Direction</b>	<b>Description</b>
CSB	Input	Port B address extension
WEB	Input	Port B write enable
CLKB	Input	Port B clock
CEB	Input	Port B clock enable
ORCEB	Input	Port B output register enable
RSTB	Input	Port B data register reset
DOB	Output	Port B data output
WWCONF	Output	Write Conflict

#### 5.13.4 Paramater Description

Table 5-36 GTP\_DRM18K Parameter Description

<b>Parameter</b>	<b>Description</b>	<b>Setting Value</b>
CSA_MASK[2:0]	Port A address extension control signal	0 ~ 7
CSB_MASK[2:0]	Port B address extension control signal	0 ~ 7
DATA_WIDTH_A	Maximum data width of Port A	1, 2, 4, 8, 16, 32, 9, 18, 36
DATA_WIDTH_B	Maximum data width of Port B	1, 2, 4, 8, 16, 32, 9, 18, 36
WRITE_MODE_A	Port A write mode	"NORMAL_WRITE" "TRANSPARENT_WRITE" "READ_BEFORE_WRITE"
WRITE_MODE_B	Port B write mode	"NORMAL_WRITE" "TRANSPARENT_WRITE" "READ_BEFORE_WRITE"
DOA_REG	Port A output register	0 = Do not enable output register 1 = Enable output register
DOB_REG	Port B output register	0 = Do not enable output register 1 = Enable output register
RST_TYPE	Reset mode selection	"SYNC": Synchronous Reset "ASYNC": Asynchronous Reset "ASYNC_SYNC_RELEASE": Asynchronous Reset with Synchronous Release
RAM_MODE	RAM mode selection	"TRUE_DUAL_PORT": True Dual-Port RAM "SIMPLE_DUAL_PORT": Simple Dual-Port RAM "SINGLE_PORT": Single-Port RAM "ROM": ROM
WRITE_COLLISION_ARBITER	Write conflict arbitration	"NULL": No arbitration
GRS_EN	Global reset enable signal (Internal Chip)	"FALSE": Global Reset Not Enabled; "TRUE": Global Reset enabled.
DOA_REG_CLKINV	Port A output register clock inversion	0 = Clock not inverted 1 = Clock inverted
DOB_REG_CLKINV	Port B output register clock inversion	0 = Clock not inverted 1 = Clock inverted

Parameter	Description	Setting Value
INIT_00 INIT_01 INIT_02 ... INIT_3F	RAM Initialization Configuration Parameters	0 ~2^288-1
INIT_FILE	Initialization files	"NONE": If no initialization file is specified, the default initialization data is all zeros; "XXX": XXX represents the specific initialization file path
BLOCK_X	Data cascade coordinates when DRM18K is cascaded	Depends on the number of cascaded DRMs
BLOCK_Y	Address cascade coordinates when RAM18K is cascaded	Depends on the number of cascaded DRMs
RAM_DATA_WIDTH	Maximum data width after RAM is cascaded	Depends on the number of cascaded DRMs
RAM_ADDR_WIDTH	Maximum address width after RAM is cascaded	Depends on the number of cascaded DRMs
INIT_FORMAT	Initialization file format	"BIN": Binary "HEX": Hexadecimal
SIM_DEVICE	Simulation Model Device Identification	"TITAN": For TITAN devices "PGL22G": For PGL22G and PGL22GS devices "LOGOS": For subsequent devices of Logos Family

The following conditions are not supported simultaneously:

`SIM_DEVICE = "PGL22G"`

`RAM_MODE = "TRUE_DUAL_PORT"`

`WRITE_MODE_A = "READ_BEFORE_WRITE"`

`WRITE_MODE_B = "READ_BEFORE_WRITE"`

## 5.13.5 Detailed Functional Description

### 5.13.5.1 Write Mode - NW Mode

NW mode is Normal Write mode, which is the default mode. When the write enable signal WEA is high-level, no read operation is performed, and the input data DIA is written to the current address. When the write enable signal WEA is low-level, the data saved in the current address ADDRA is read and output from the DOA port in the next clock cycle. Its waveform is shown in the following figure.

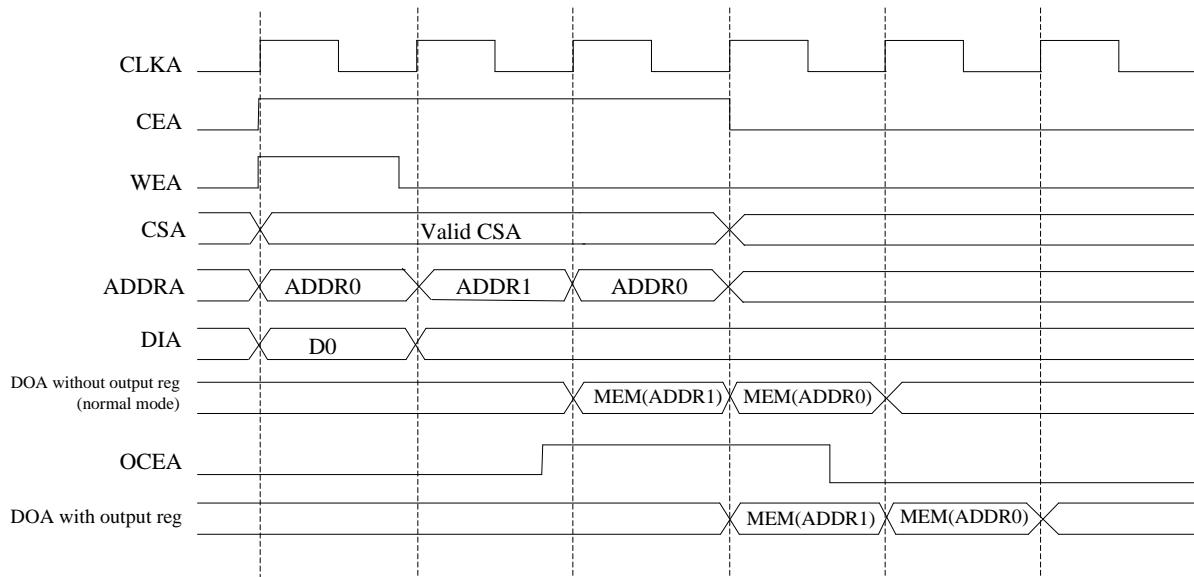


Figure 5-28 GTP\_DRM18K NW Mode Waveform Diagram

Notes:

1. The port write operating mode of DRM is only applicable to ports with read/write functions, i.e., the A/B ports of DP RAM and the ports of SP RAM.
2. SDP mode does not support write mode setting. It is disabled to set to TW or RBW mode during use; it can only be set to the default NW mode, or DRM will enter an abnormal write operation state.

### 5.13.5.2 Write Mode - TW Mode

In TW mode, when the write enable signal WEA is high-level, the input data DIA is written to the address ADDRA and read directly from the DOA port in the next clock cycle. When the enable signal WEA is low-level, the data saved in the current address ADDRA is read and output from the DOA port in the next clock cycle. Its waveform is shown in the following figure.

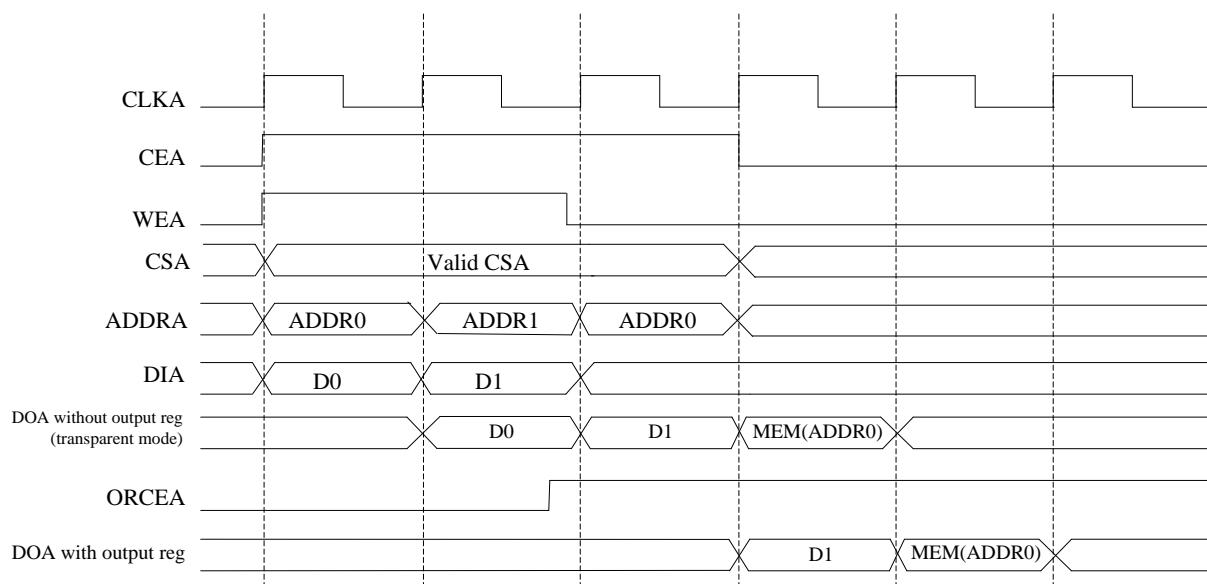


Figure 5-29 GTP\_DRM18K TW Mode Waveform Diagram

### 5.13.5.3 Write Mode - RBW Mode

In RBW mode, when the write enable signal WEA is high, the original data referenced by address ADDRA is first read and output to the output port in the next clock cycle of the write operation, then the input data DIA is written into address ADDRA to overwrite the original data. When the enable signal WEA is low-level, the data saved in the current address ADDRA is read and output from the DOA port in the next clock cycle. Its waveform is shown in the following figure.

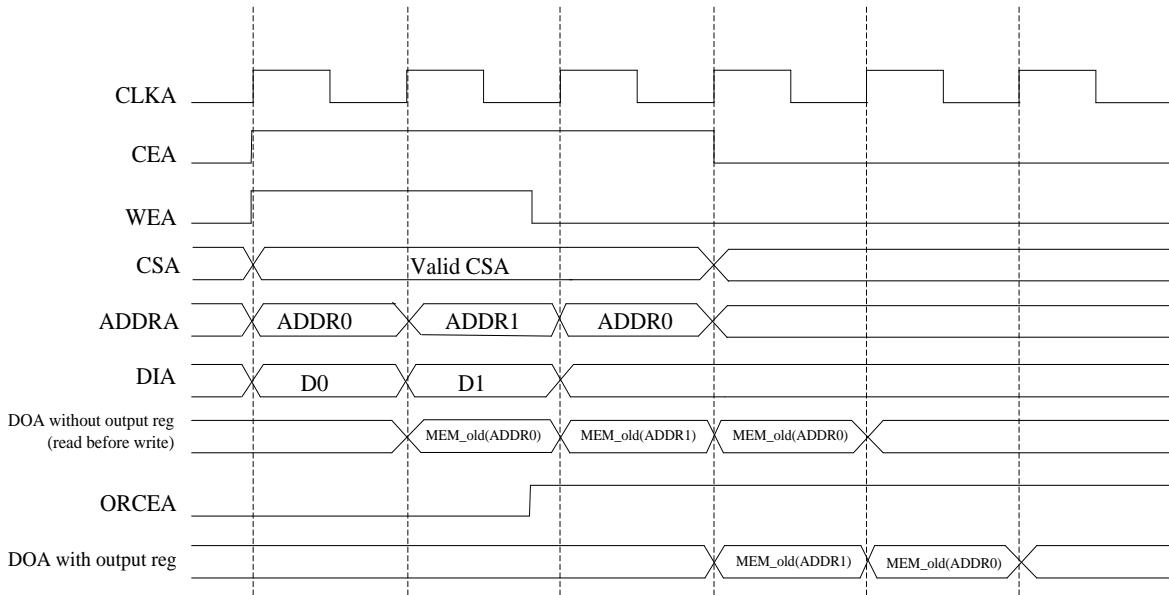


Figure 5-30 GTP\_DRM18K RBW Mode Waveform Diagram

### 5.13.5.4 Implementation of TB and RBW Modes for SP RAM

The TW (Transparent Write) and RBW (Read before Write) modes under 32/36 bits SP mode for DRM cannot be implemented by direct configuration (The setting of the write mode to TW or RBW mode is disabled when GTP is configured as 32/36 bit SP mode, or DRM will enter an abnormal write operation state). It requires additional configuration for its implementation, which is shown as follows:

10. Configure both Port A and Port B to 16/18 bits in DP mode (for 32/36 bits SP respectively);
11. Port A and Port B are configured to the same write mode; other configuration bits also need to be the same.
12. The CE, CS, WE, ADDR, ORCE, RST, CLK, and other control signals for Port A and Port B are paralleled respectively;
13. DIA and DIB are paralleled as data inputs; DOA and DOB are paralleled as data outputs;
14. Both Port A and Port B address input ports must be connected to the address input simultaneously; ADDRA[4] and ADDRB[4] must be connected to 0 or 1 respectively (in opposite).

This is shown in the following figure.

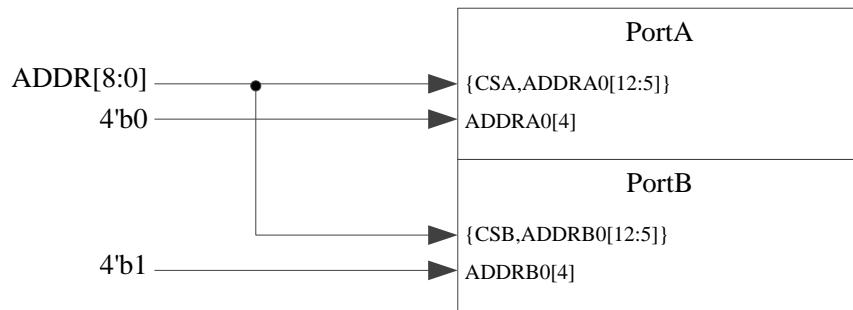


Figure 5-31 GTP\_DRM18K SP Interface Diagram

### 5.13.5.5 Output Register Mode

For data output ports, DRM provides an optional Output Register for improved timing performance. As shown in the figure below, the output register can be controlled by an independent enable signal OCE. When OCE is a constant 1, the use of the output register will increase the delay of the read operation from one clock cycle to two clock cycles (Case 1); in pipeline designs with flow control, the user can also flexibly control OCE with logic (Case 2).

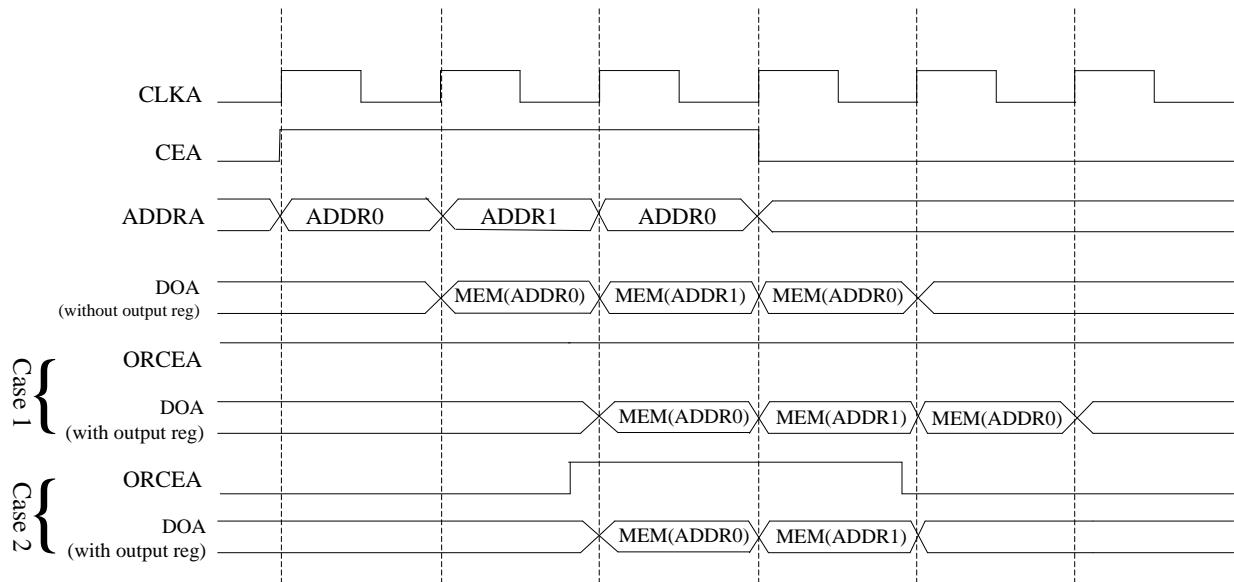


Figure 5-32 GTP\_DRM18K Register Mode Waveform Diagram

### 5.13.6 Instantiation template

GTP\_DRM18K

```
#(
    .CSA_MASK      (3'b000),
```











```

.WEB      (WEB      ),
.CLKB     (CLKB     ),
.CEB       (CEB      ),
.ORCEB    (ORCEB   ),
.RSTB     (RSTB    ),
.WWCONF   (WWCONF )
);

);

```

### 5.13.7 How to Use GTP

When mapping LOGOS devices: No WWCONF port available.

## 5.14 Usage Instructions for GTP\_FIFO18K

### 5.14.1 Supported Devices

Table 5-37 Device Models That Support GTP\_FIFO18K

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 5.14.2 Description of Functionality

DRM\_FIFO supports 18K bits storage units and can be configured to data width modes of 16K\*1, 8K\*2, 4K\*4, 2K\*9(8), 1K\*18(16), and 512\*36(32). When DRM is configured as FIFO mode, it is configured to Simple Dual Port mode, and does not support mixed bit width or byte enable write operation.

DRM\_FIFO can be used to configure PG DRM into synchronous FIFO or asynchronous FIFO mode. In DRM SDP RAM mode, one port is dedicated to FIFO data writing, and another port is dedicated to FIFO data reading, with read and write ports using different clocks. DRM\_FIFO supports an optional rewrite/resend mode in both synchronous and asynchronous modes. See the structure block diagram below.

DRM\_FIFO supports register mode, which is used in the same way as the GTP\_DRM register mode.

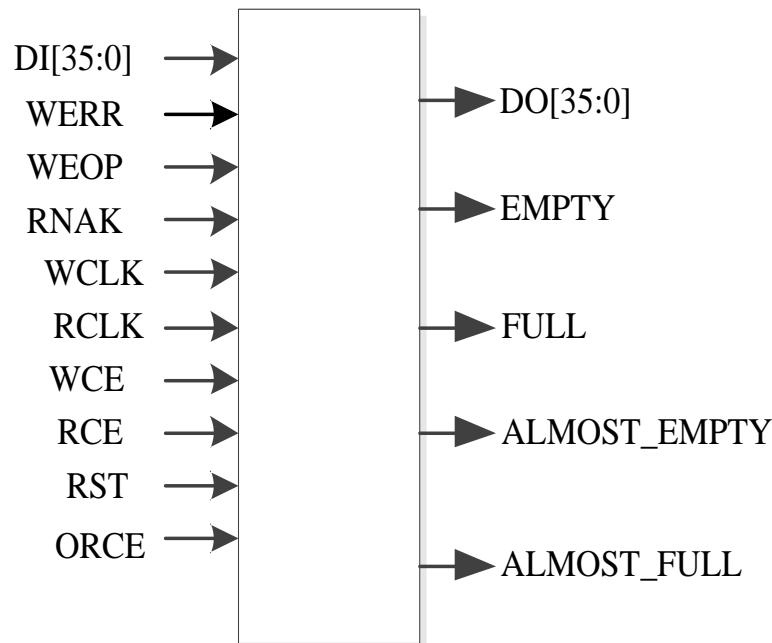


Figure 5-33GTP\_FIFO18K Structure Block Diagram

### 5.14.3 Port Description

Table 5-38 GTP\_FIFO18K Port List

Port	Direction	Description
DI	Input	Data write in
WERR	Input	Write port error packet indicator
WEOP	Input	End-of-packet signal indicator
RNAK	Input	Read port resend indicator
WCLK	Input	Write clock signal
RCLK	Input	Read clock signal
WCE	Input	Write enable signal
RCE	Input	Read enable signal
RST	Input	Reset signal
ORCE	Input	Output register enable signal
DO	Output	Data read out
EMPTY	Output	Read port empty flag
FULL	Output	Write port full flag
ALMOST_EMPTY	Output	Read port almost empty flag
ALMOST_FULL	Output	Write port almost full flag

### 5.14.4 Parameter Description

The table below is a description of the GTP parameters, with default values all set to 0.

Table 5-39 GTP\_FIFO18K Parameter List

Parameter	Description	Setting Value
GRS_EN	Global reset enable signal (Internal Chip)	"TRUE": Global Reset enabled "FALSE": Global Reset Not Enabled
ALMOST_FULL_OFFSET	When the FIFO is almost full and the difference between the write and read pointers equals to ALMOST_FULL_OFFSET, the almost_full flag is generated.	16382: 16K address 8190: 8K address 4094: 4K address 2046: 2K address 1022: 1K address 510: 512 address
ALMOST_EMPTY_OFFSET	When the FIFO is almost empty and the difference between the read and write pointers equals to ALMOST_EMPTY_OFFSET, the almost_empty flag is generated.	16382: 16K address 8190: 8K address 4094: 4K address 2046: 2K address 1022: 1K address 510: 512 address
SYNC_FIFO	Asynchronous/Synchronous FIFO Selection	"TRUE": Use synchronous FIFO "FALSE": Use asynchronous FIFO
DATA_WIDTH	FIFO data width	1,2,4,8,9,16,18,32,36
REWRITE_EN	Packet loss enable	"FALSE": No packet loss; "TRUE": Packet loss enabled
RESEND_EN	Resend enable	"FALSE": No resend; "TRUE": Resend enabled
USE_EMPTY	Enable read empty flag	1: Read empty flag enabled; 0: Read empty flag not enabled
USE_FULL	Enable write full flag	1: Write full flag enabled; 0: Write full flag not enabled
DO_REG	Output register enable	1: Enabled; 0: Not enabled

### 5.14.5 Detailed Functional Description

FIFO mode supports synchronous/asynchronous modes. In synchronous mode, the read/write clocks share the same clock. In asynchronous mode, the read/write clocks are two independent clocks.

#### 5.14.5.1 Write Timing to Empty FIFO

The EMPTY signal indicates the FIFO is empty. When WR\_EN is active and data is successfully written, in synchronous FIFO, the EMPTY signal is cleared after one RCLK clock cycle; in asynchronous FIFO, the EMPTY signal is cleared after two RCLK clock cycles. When WR\_EN remains active, the ALMOST\_EMPTY signal is cleared with a delay based on the configuration of ALMOST\_EMPTY\_OFFSET. Its waveform is shown in the following figure.

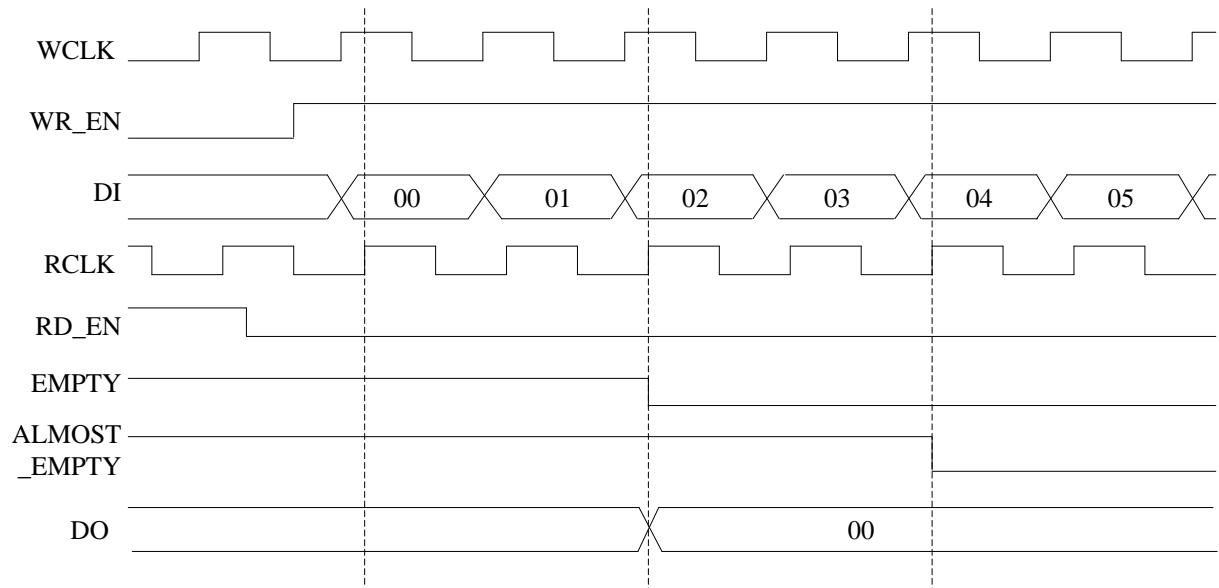


Figure 5-34 GTP\_FIFO18K Waveform Diagram for Writing to Empty FIFO Mode

#### 5.14.5.2 Write Timing to Almost Full FIFO

When the FIFO is almost full, **ALMOST\_FULL** will be set to 1 in advance based on the configuration of **ALMOST\_FULL\_OFFSET**. When the FIFO is full, the write address will no longer increase. Its waveform is shown in the following figure.

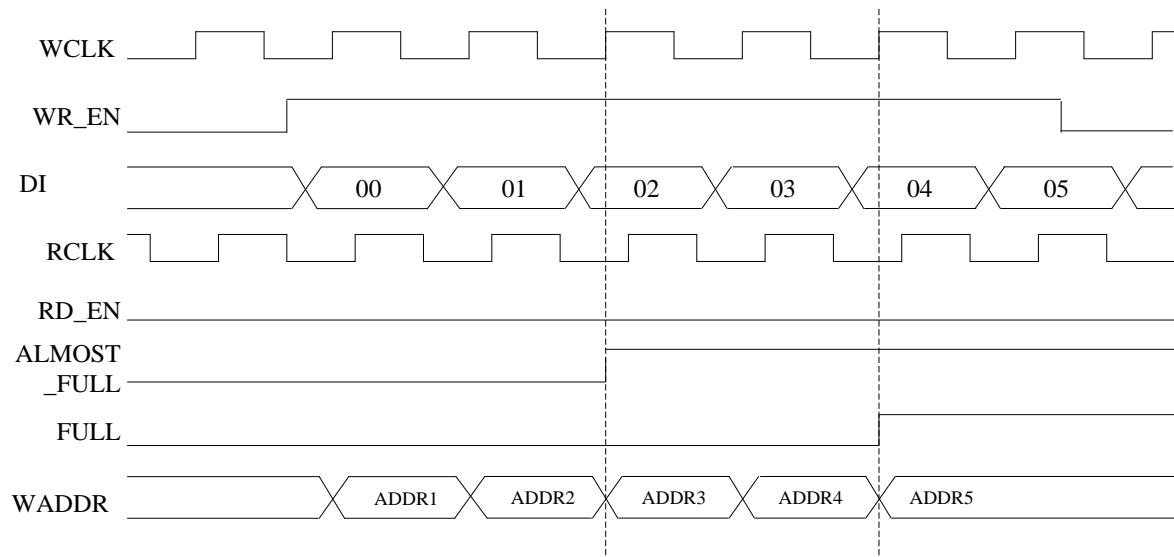


Figure 5-35 GTP\_FIFO18K Waveform Diagram for Writing to Almost Full FIFO Mode

#### 5.14.5.3 Read Timing from Full FIFO

The **FULL** signal indicates the FIFO is full. After data is read out when **RD\_EN** is active, the **FULL** signal clears within a delay of 1 (synchronous FIFO) to 2 (asynchronous FIFO) **WCLK** clock cycles.

When RD\_EN remains active, the ALMOST\_FULL signal is cleared with a delay based on the configuration of ALMOST\_FULL\_OFFSET. Read timing from full FIFO is shown below

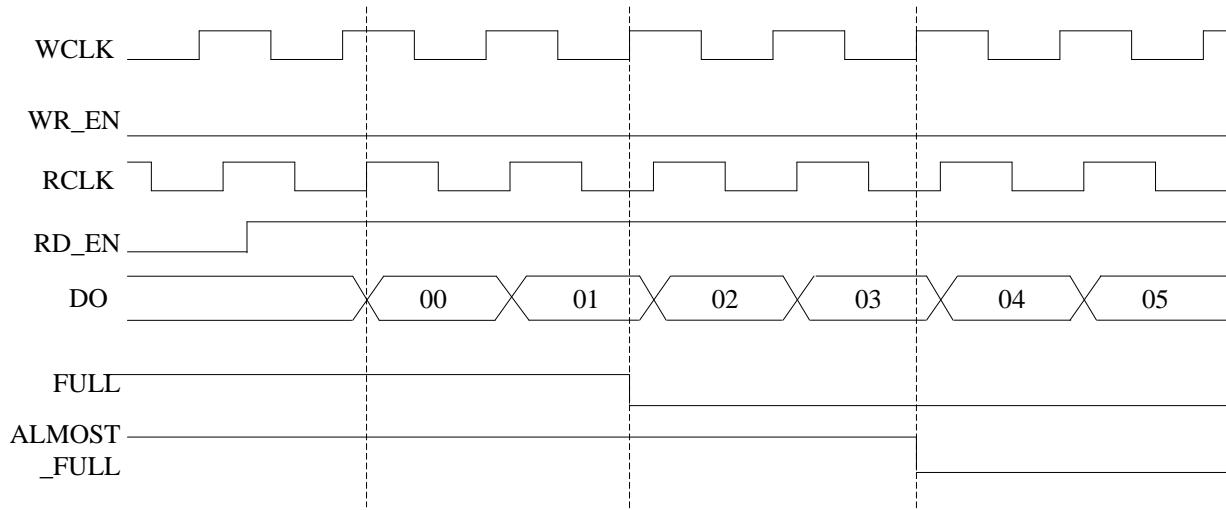


Figure 5-36 GTP\_FIFO18K Waveform Diagram for Full FIFO Readout Mode

#### 5.14.5.4 Read Timing from Almost Empty FIFO

When the FIFO is almost empty, ALMOST\_EMPTY will be set to 1 in advance based on the configuration of ALMOST\_EMPTY\_OFFSET. When the FIFO is EMPTY, the read address will no longer increase. Its waveform is shown in the following figure.

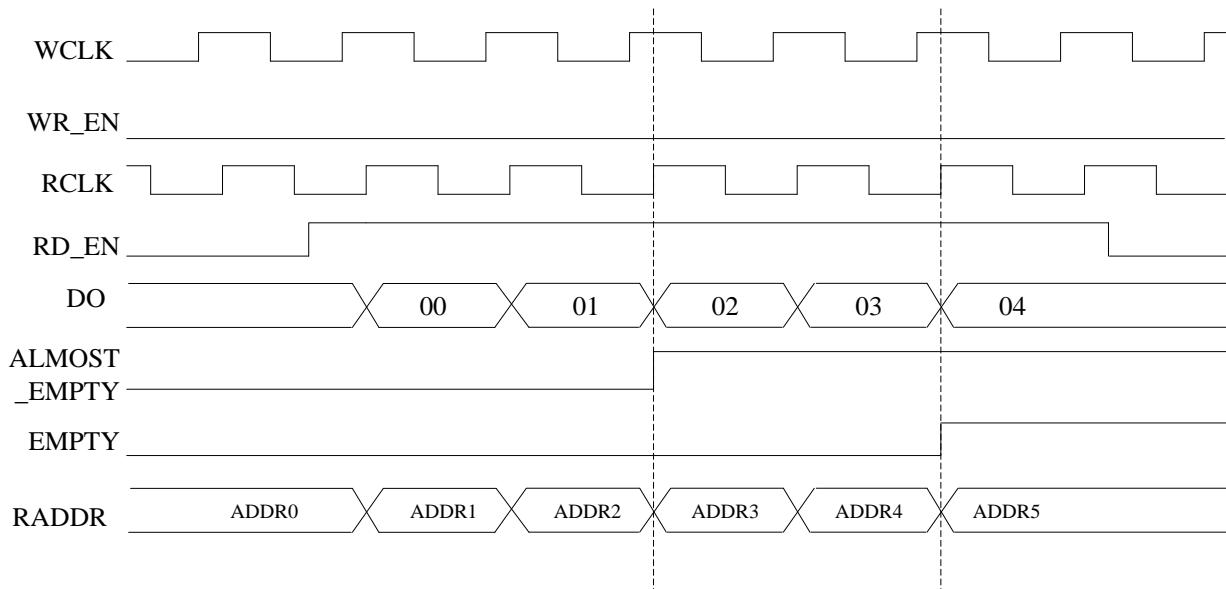


Figure 5-37 GTP\_FIFO18K Waveform Diagram for Reading from Almost Empty FIFO Mode

#### 5.14.5.5 Write Timing to Packet Loss

When the FIFO write port detects a wr\_err packet error signal, the write address of the write port

reverts back to the write address of the packet header in the next WCLK cycle. The packet loss function can be enabled independently by setting the input signal wr\_eop to 1. In rewrite/resend mode, the wr\_eop signal occupies one data bit and must be connected to the data input DI[8] (2K\*9, 1K\*18) or DI[26] (512\*36). Its waveform is shown in the following figure. Its waveform is shown in the following figure.

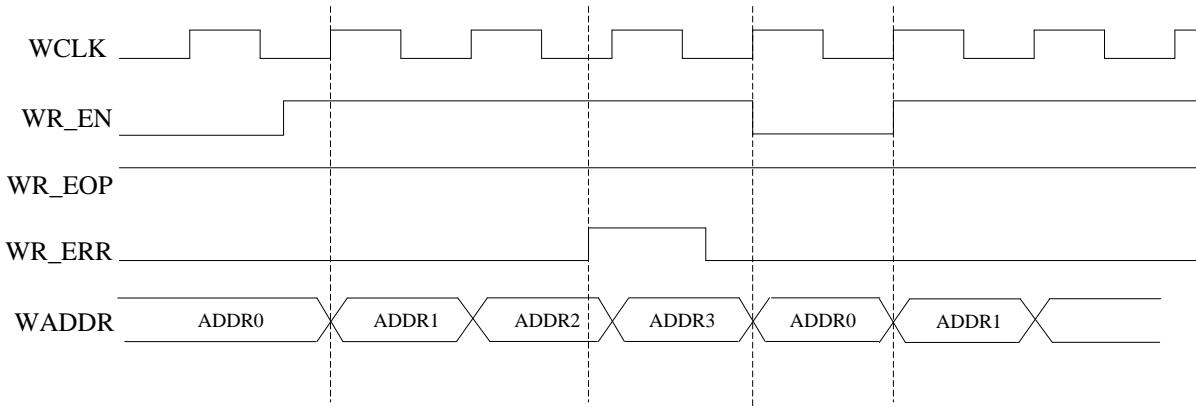


Figure 5-38 GTP\_FIFO18K Waveform Diagram for FIFO Write Packet Loss Mode

#### 5.14.5.6 Read Timing from Resend

When the FIFO write port detects a wr\_err packet error signal, the write address of the write end reverts to the write address of the packet header. The read pointer reverts to the address of the current packet header after the RD\_NAK signal is set to '1'. Both the rewrite/resend functions need to be enabled simultaneously.

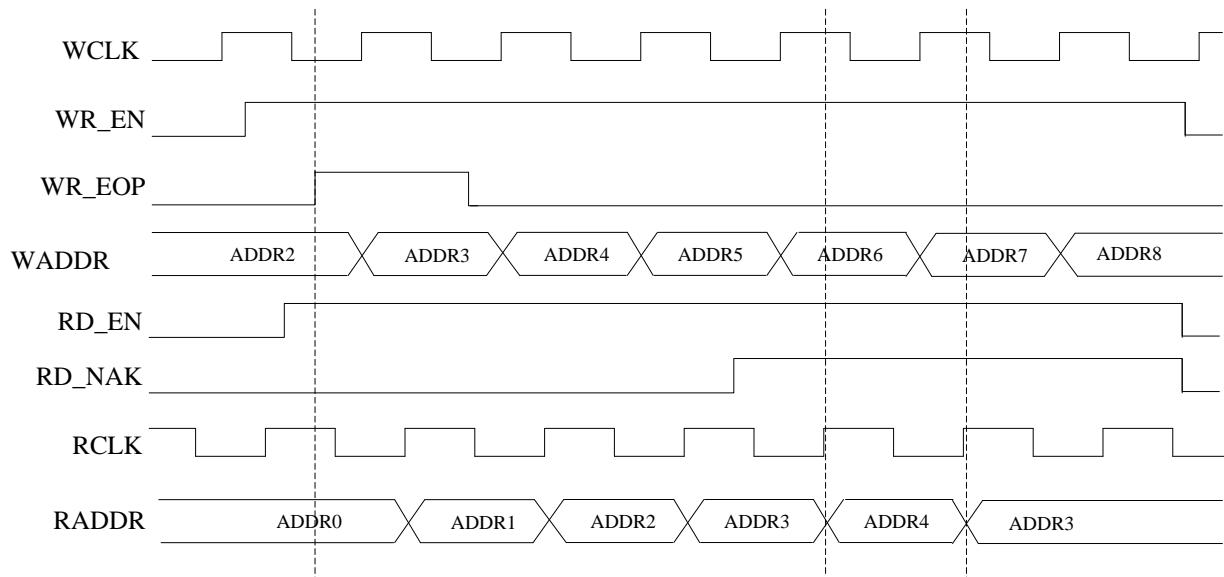
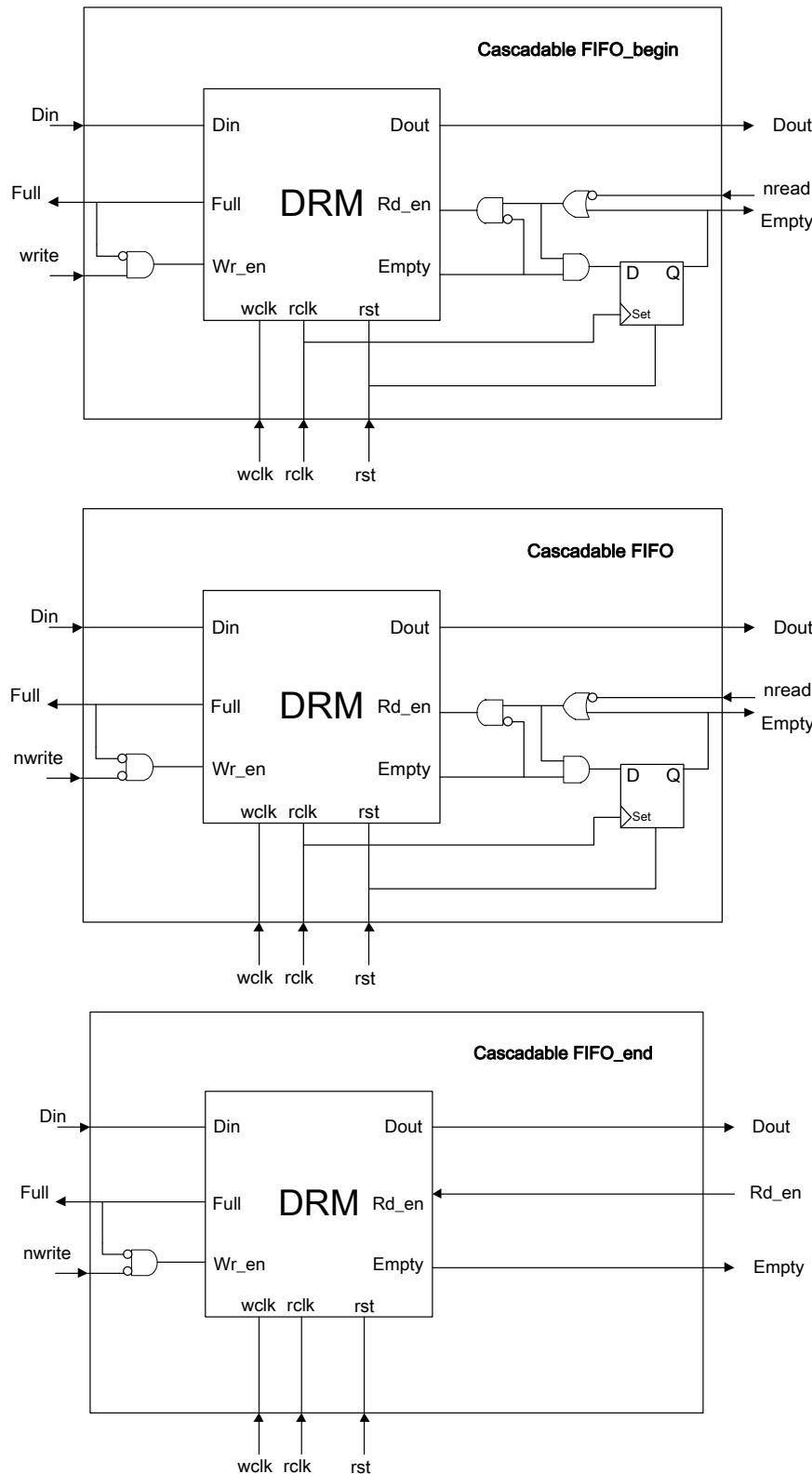


Figure 5-39 GTP\_FIFO18K Waveform Diagram for FIFO Read Resend Mode

## 5.14.6 Expansion via Cascading

### 5.14.6.1 Depth Expansion

Depth expansion can be achieved by cascading N FIFOs: if the frequency of WCLK is higher than that of the RCLK, then INTCLK = WCLK; if the frequency of WCLK is lower than or equal to that of the RCLK, then INTCLK = RCLK. The ALMOST\_EMPTY signal is generated by the last FIFO, and the ALMOST\_FULL signal is generated by the first FIFO. This is shown in the following figure.



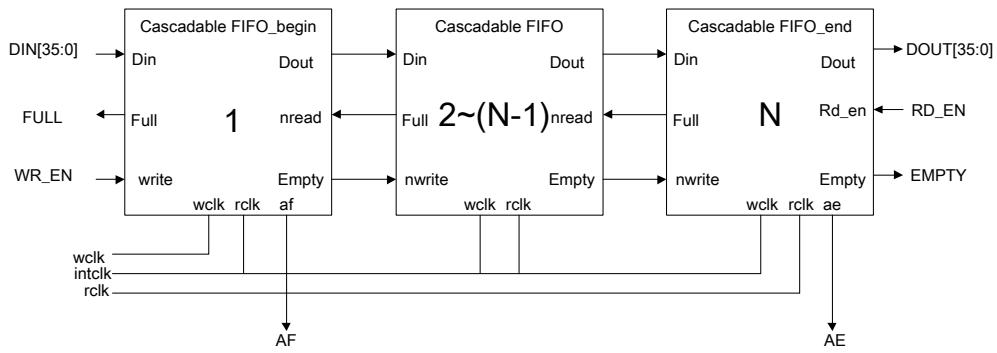


Figure 5-40 GTP\_FIFO18K FIFO Depth Expansion Diagram

For example , when N=3, the depth expansion application are described as below.

The drm1 is instantiated as cascadable FIFO\_begin. Where the data input port after cascading is din[35:0]; AF and Full serve as after-cascading flags; WR\_EN is input to "write".

The drm2 is instantiated as cascadable FIFO.

The drm3 is instantiated as cascadable FIFO\_end. Where AE and Empty serve as after-cascading flags; RD\_EN is input to rd\_en. Dout is the output.

### 5.14.6.2 Width Expansion

Width expansion can be implemented by parallel FIFOs, as shown in the diagram below.

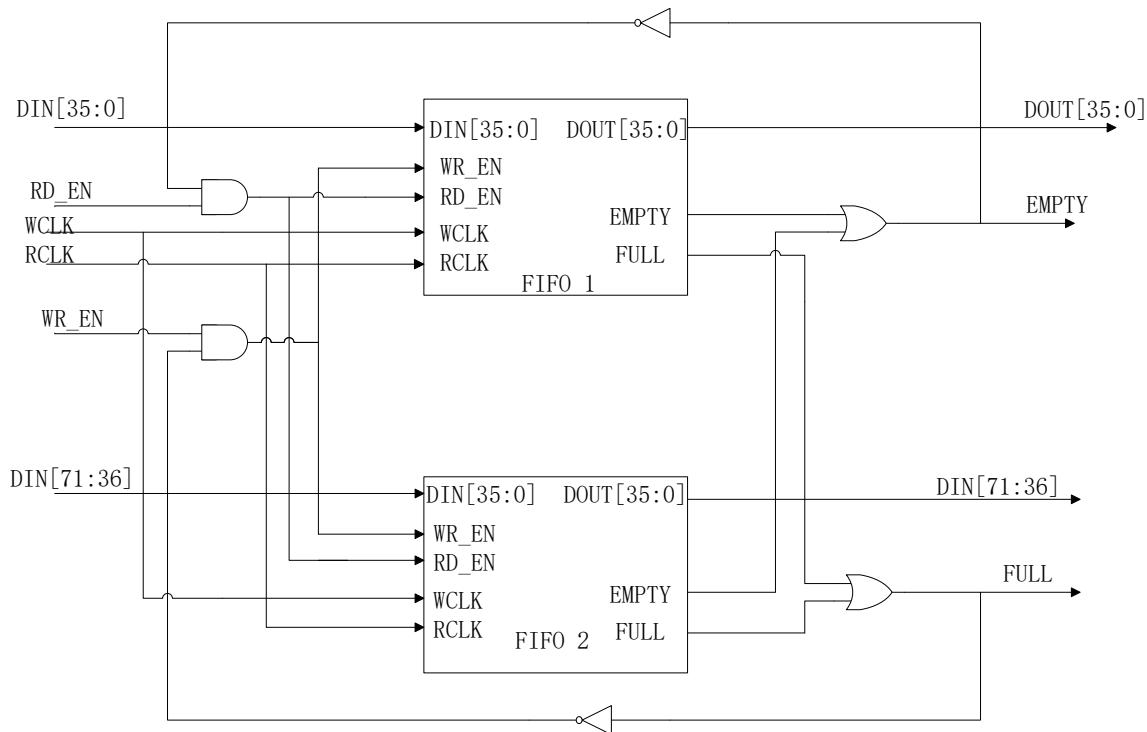


Figure 5-41 GTP\_FIFO18K FIFO Width Expansion Diagram

### 5.14.7 Instantiation template

```
GTP_FIFO18K #(
    .GRS_EN("TRUE"),
    .DATA_WIDTH(18),
    .DO_REG(0),
    .ALMOST_FULL_OFFSET('b0000000000000000),
    .ALMOST_EMPTY_OFFSET('b0000000000000000),
    .USE_EMPTY(0),
    .USE_FULL(0),
    .REWRITE_EN("FALSE"),
    .RESEND_EN("FALSE"),
    .SYNC_FIFO("FALSE")
)
```

```
GTP_FIFO18K_inst (
    .DO(),           // OUTPUT[35:0]
    .DI(),           // INPUT[35:0]
```

```
.ALMOST_EMPTY(),// OUTPUT  
.ALMOST_FULL(), // OUTPUT  
.EMPTY(),        // OUTPUT  
.FULL(),         // OUTPUT  
.ORCE(),         // INPUT  
.RCE(),          // INPUT  
.RCLK(),         // INPUT  
.RNAK(),         // INPUT  
.RST(),          // INPUT  
.WCE(),          // INPUT  
.WCLK(),         // INPUT  
.WEOP(),         // INPUT  
.WERR()          // INPUT  
);
```

## Chapter 6 Usage Instructions for IO-related GTPs

### 6.1 Usage Instructions for GTP\_INBUF

#### 6.1.1 Supported Devices

Table 6-1 Device Models That Support GTP\_INBUF

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 6.1.2 Description of Functionality

GTP\_INBUF is an input BUFFER. It is shown in the following figure:

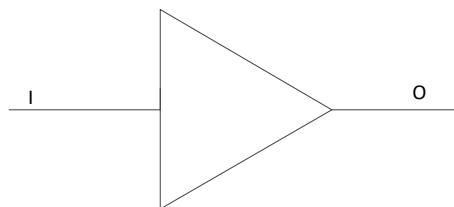


Figure 6-1 GTP\_INBUF

#### 6.1.3 Port Description

Table 6-2 GTP\_INBUF Port List Description

Port Name	Input/Output	Description
I	Input	Input Signal
O	Output	Output signal

#### 6.1.4 Parameter Description

Table 6-3 GTP\_INBUF Parameter List

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "LVTTLL33", "PCI33", etc.
TERM_DDR	When inputting HSTL, SSTL standards, the built-in terminal resistor can be enabled or disabled	"ON"	"ON", "OFF"

### 6.1.5 Instantiation template

```
GTP_INBUF#(
    .IOSTANDARD ("DEFAULT"),
    .TERM_DDR      ("ON")
)
GTP_INBUF_inst (
    .I  (i),
    .O  (o)
);
```

## 6.2 Usage Instructions for GTP\_INBUFDS

### 6.2.1 Supported Devices

Table 6-4 Device Models That Support GTP\_INBUFDS

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.2.2 Description of Functionality

GTP\_INBUFDS is a differential input BUFFER. It is shown in the following figure:

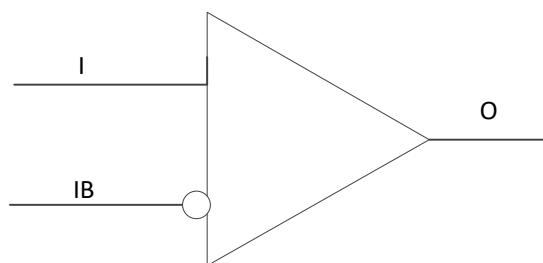


Figure 6-2 GTP\_INBUFDS

### 6.2.3 Port Description

Table 6-5 GTP\_INBUFDS Port List Description

Port Name	Input/Output	Description
I	Input	Differential P input signals
IB	Input	Differential N input signals

Port Name	Input/Output	Description
O	Output	Output signal

#### 6.2.4 Paramater Description

Table 6-6 GTP\_INBUFDS Parameter List

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "LVDS", "MINI-LVDS", "LVPECL", etc.
TERM_DIFF	When in differential input, The built-in terminal resistor is enabled or disabled	"ON"	"ON", "OFF"

#### 6.2.5 Instantiation template

```
GTP_INBUFDS#(
    .IOSTANDARD ("DEFAULT"),
    .TERM_DIFF("ON")
)
GTP_INBUFDS_inst (
    .I      (i),
    .IB     (ib ),
    .O      (o)
);
```

### 6.3 Usage Instructions for GTP\_INBUFG

#### 6.3.1 Supported Devices

Table 6-7 Device Models That Support GTP\_INBUFG

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 6.3.2 Description of Functionality

GTP\_INBUFG is a clock input BUFFER. It is shown in the following figure:

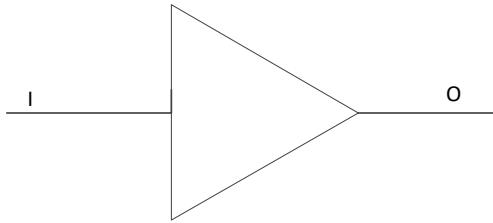


Figure 6-3 GTP\_INBUFG

### 6.3.3 Port Description

Table 6-8 GTP\_INBUFG Port List Description

Port Name	Input/Output	Description
I	Input	Input Signal
O	Output	Output signal

### 6.3.4 Parameter Description

Table 6-9 GTP\_INBUFG Parameter List Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "LVTTL33", "LVCMOS25", etc.
TERM_DDR	When inputting HSTL, SSTL standards, the built-in terminal resistor can be enabled or disabled	"ON"	"ON", "OFF"

### 6.3.5 Instantiation template

```

GTP_INBUFG#(
    .IOSTANDARD ("DEFAULT"),
    .TERM_DDR ("ON")
)
GTP_INBUFG_inst (
    .I  (i),
    .O  (o)
);

```

## 6.4 Usage Instructions for GTP\_INBUFGDS

### 6.4.1 Supported Devices

Table 6-10 Device Models That Support GTP\_INBUFGDS

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.4.2 Description of Functionality

GTP\_INBUFGDS is a differential clock input BUFFER. It is shown in the following figure:

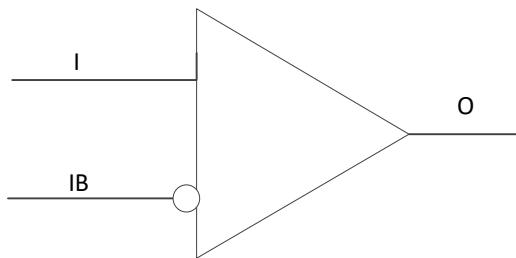


Figure 6-4 GTP\_INBUFGDS

### 6.4.3 Port Description

Table 6-11 GTP\_INBUFGDS Port List Description

Port Name	Input/Output	Description
I	Input	Differential P input signals
IB	Input	Differential N input signals
O	Output	Output signal

### 6.4.4 Parameter Description

Table 6-12 GTP\_INBUFGDS Parameter List Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "LVTT33", "LVCMS25", etc.
TERM_DIFF	When in differential input, the built-in terminal resistor is enabled or disabled	"ON"	"ON", "OFF"

### 6.4.5 Instantiation template

GTP\_INBUFGDS#(

```

.IOSTANDARD ("DEFAULT"),
.TERM_DIFF ("ON")
)
GTP_INBUFGDS_inst (
.I (i),
.IB (ib ),
.O (o)
);
    
```

## 6.5 Usage Instructions for GTP\_IOBUF

Table 6-13 Device Models That Support GTP\_IOBUF

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.5.1 Description of Functionality

GTP\_IOBUF is a bidirectional single-ended BUFFER, connecting external bidirectional pins with internal logic circuits. It is shown in the following figure:

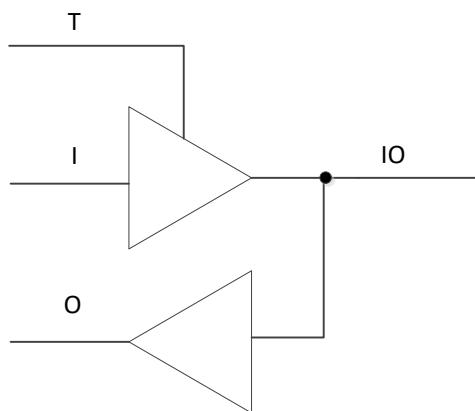


Figure 6-5 GTP\_IOBUF

### 6.5.2 Port Description

Table 6-14 GTP\_IOBUF Port List Description

Port Name	Input/Output	Description
I	Input	Input Signal
T	Input	Tri-state input enable signal, high for tri-state and low for output
O	Output	Output signal

Port Name	Input/Output	Description
IO	Input/output	Input/output signal (in/out)

### 6.5.3 Paramater Description

Table 6-15 GTP\_IOBUF Parameter List Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "LVTTL33", "LVCMOS25", etc.
SLEW_RATE	Slew rate property	"SLOW"	"FAST", "SLOW"
DRIVE_STRENGTH	Output Drive	"8"	2, 4, 6, 8, 12, 16, 24
TERM_DDR	The built-in terminal resistor is enabled or disabled	"ON"	"ON", "OFF"

### 6.5.4 Instantiation template

```
GTP_IOBUF#(
    .IOSTANDARD ("DEFAULT"),
    .SLEW_RATE ("SLOW"),
    .DRIVE_STRENGTH ("8"),
    .TERM_DDR("ON")
)
GTP_IOBUF_inst (
    .I  (i),
    .T  (t),
    .IO (io),
    .O  (o)
);
```

## 6.6 Usage Instructions for GTP\_IOBUFCO

### 6.6.1 Supported Devices

Table 6-16 Device Models That Support GTP\_IOBUFCO

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.6.2 Description of Functionality

GTP\_IOBUFCO is a bidirectional pseudo-differential buffer that only supports pseudo-differential levels (corresponding to GTP\_IOBUFDS, which only supports true differential levels). The structure block diagram is shown below.

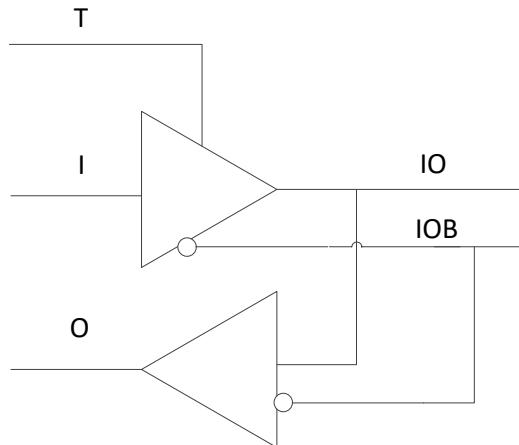


Figure 6-6 GTP\_IOBUFCO

### 6.6.3 Port Description

Table 6-17 GTP\_IOBUFCO Port List Description

Port Name	Input/Output	Description
I	Input	Input Signal
T	Input	Tri-state enable signal, high for tri-state and low for output
IO	Input/Output	Differential P side input/output
IOB	Input/Output	Differential Q side input/output
O	Output	Output signal

### 6.6.4 Parameter Description

Table 6-18 GTP\_IOBUFCO Parameter List Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	"SSTL18D_I", "SSTL18D_II", "SSTL15D_I", "SSTL15D_II", "HSTL15D_I", "SSTL25D_I", "SSTL25D_II", "SSTL15D_I_CAL", "SSTL15D_II_CAL", "HSTL15D_I_CAL", "LVPECL", "RSDS", "PPDS", "BLVDS", "DEFAULT"
TERM_DDR	When inputting HSTL, SSTL standards, the built-in terminal resistor can be enabled or disabled	"ON"	"ON", "OFF"

### 6.6.5 Instantiation template

```
GTP_IOBUFCO#(
    .IOSTANDARD ("DEFAULT"),
    .TERM_DDR ("ON")
)
GTP_IOBUFCO_inst(
    .I (i),
    .T (t),
    .IO (io),
    .IOB(io),
    .O (o)
);
```

## 6.7 Usage Instructions for GTP\_IOBUFDS

### 6.7.1 Supported Devices

Table 6-19 Device Models That Support GTP\_IOBUFDS

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.7.2 Description of Functionality

GTP\_IOBUFDS is a bidirectional single-ended differential input BUFFER. It is shown in the following figure:

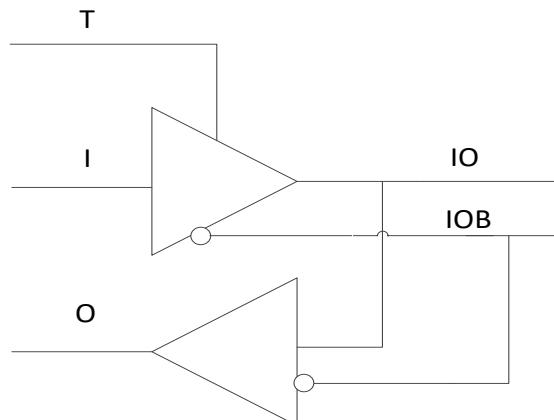


Figure 6-7 GTP\_IOBUFDS

### 6.7.3 Port Description

Table 6-20 GTP\_IOBUFDS Port List Description

Port Name	Input/Output	Description
I	Input	Input Signal
T	Input	Tri-state input enable signal, high for tri-state and low for output
O	Output	Output signal
IO	Input/output	Differential P input/output(in/out)
IOB	Input/output	Differential N input/output(in/out)

### 6.7.4 Parameter Description

Table 6-21 GTP\_IOBUFDS Parameter List Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	"LVDS", "MINI-LVDS", "SUB-LVDS", "TMDS"
TERM_DIFF	When using standard inputs such as LVDS, the built-in differential terminal resistor is enabled or disabled	"ON"	"ON", "OFF"

### 6.7.5 Instantiation template

```
GTP_IOBUFDS#(
    .IOSTANDARD ("DEFAULT"),
    .TERM_DIFF  ("ON")
)
GTP_IOBUFDS_inst (
    .I      (i),
    .T      (t),
    .IO     (io),
    .IOB    (iob),
    .O      (o)
);
```

## 6.8 GTP\_OUTBUF Usage Instructions

### 6.8.1 Supported Devices

Table 6-22 Device Models That Support GTP\_OUTBUF

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.8.2 Description of Functionality

GTP\_OUTBUF is an output BUFFER. It is shown in the following figure:

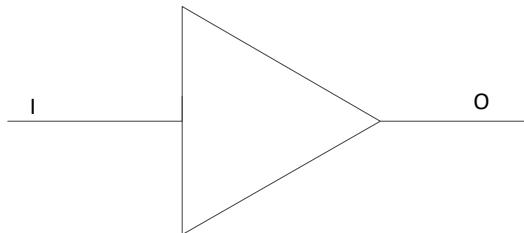


Figure 6-8 GTP\_OUTBUF

### 6.8.3 Port Description

Table 6-23 GTP\_OUTBUF Port List Description

Port Name	Input/Output	Description
I	Input	Input Signal
O	Output	Output signal

### 6.8.4 Parameter Description

Table 6-24 GTP\_OUTBUF Parameter List Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "LVTTL33", "PCI33", etc.
SLEW_RATE	Slew rate property	"SLOW"	"FAST", "SLOW"
DRIVE_STRENGTH	Output Drive	"8"	2,4,6,8,12,16,24

### 6.8.5 Instantiation template

```
GTP_OUTBUF#(
    .IOSTANDARD ("DEFAULT"),
    .
```

```

.SLEW_RATE ("SLOW"),
.DRIVE_STRENGTH (8)
)
GTP_OUTBUF_inst (
.I (i),
.O (o)
);

```

## 6.9 Usage Instructions for GTP\_OUTBUFCO

### 6.9.1 Supported Devices

Table 6-25 Device Models That Support GTP\_OUTBUFCO

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.9.2 Description of Functionality

GTP\_OUTBUFCO is a pseudo-differential output buffer. It is shown in the following figure:

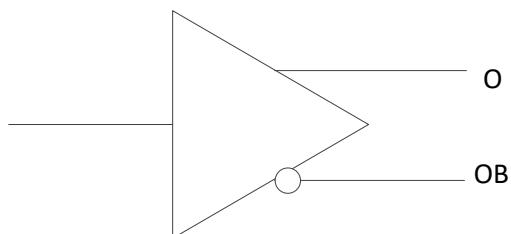


Figure 6-9 GTP\_OUTBUFCO

### 6.9.3 Port Description

Table 6-26 GTP\_OUTBUFCO Port List Information

Port Name	Input/Output	Description
I	Input	Input Signal
OB	Output	Differential N output signals
O	Output	Differential P output signals

#### 6.9.4 Parameter Description

Table 6-27 GTP\_OUTBUFCO Parameter List Information

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "SSTL18D_I", "SSTL18D_II", "SSTL15D_I", etc.

#### 6.9.5 Instantiation template

```
GTP_OUTBUFCO#(
    .IOSTANDARD ("DEFAULT")
)
GTP_OUTBUFCO_inst (
    .I  (i),
    .OB(ob ),
    .O  (o)
);
```

### 6.10 Usage Instructions for GTP\_OUTBUFDS

#### 6.10.1 Supported Devices

Table 6-28 Device Models That Support GTP\_OUTBUFDS

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 6.10.2 Description of Functionality

GTP\_OUTBUFDS is a true differential output buffer as shown below:

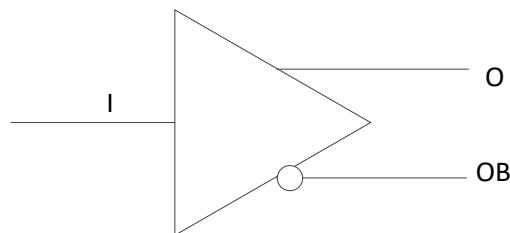


Figure 6-10 GTP\_OUTBUFDS

### 6.10.3 Port Description

Table 6-29 GTP\_OUTBUFDS Port List Description

Port Name	Input/Output	Description
I	Input	Input Signal
OB	Output	Differential N output signals
O	Output	Differential P output signals

### 6.10.4 Paramater Description

Table 6-30 GTP\_OUTBUFDS Parameter List Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "MINI-LVDS", "SUB-LVDS", "TMDS", etc.

### 6.10.5 Instance Template

```
GTP_OUTBUFDS#(
    .IOSTANDARD ("DEFAULT "),
    .GTP_OUTBUFDS_inst (
        .I  (i),
        .OB(ob ),
        .O  (o)
    );
)
```

## 6.11 Usage Instructions for GTP\_OUTBUFT

### 6.11.1 Supported Devices

Table 6-31 Device Models That Support GTP\_OUTBUFT

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.11.2 Description of Functionality

GTP\_OUTBUFT is a tri-state output BUFFER. It is shown in the following figure:

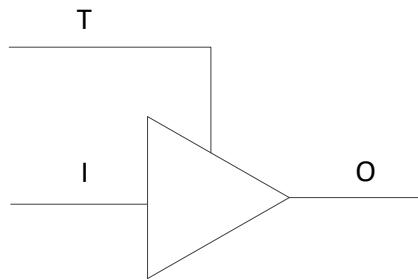


Figure 6-11 GTP\_OUTBUFT

### 6.11.3 Port Description

Table 6-32 GTP\_OUTBUFT Port List Description

Port Name	Input/Output	Description
I	Input	Input Signal
O	Output	Output signal
T	Input	Tri-state input enable signal, high for tri-state and low for output

### 6.11.4 Paramater Description

Table 6-33 GTP\_OUTBUFT Parameter List Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "LVTTL33", "PCI33", "LVCMOS33", etc.
SLEW_RATE	Slew rate property	"SLOW "	"FAST", "SLOW"
DRIVE_STRENGTH	Output Drive	"8"	2, 4, 6, 8, 12, 16, 24

### 6.11.5 Instance Template

```

GTP_OUTBUFT#(
    .IOSTANDARD ("DEFAULT"),
    .SLEW_RATE ("SLOW "),
    .DRIVE_STRENGTH (8)
)
    GTP_OUTBUFT_inst (
        .I (i),
        .O (o),
        .T (t)
    );

```

## 6.12 Usage Instructions for GTP\_OUTBUFTCO

### 6.12.1 Supported Devices

Table 6-34 Device Models That Support GTP\_OUTBUFTCO

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.12.2 Description of Functionality

GTP\_OUTBUFTCO is a tri-state output pseudo-differential BUFFER. It is shown in the following figure:

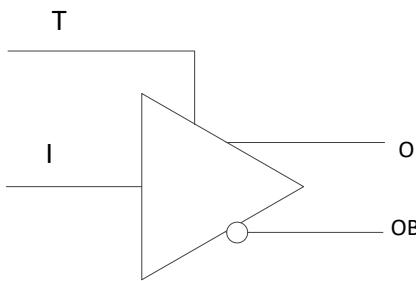


Figure 6-12 GTP\_OUTBUFTCO

### 6.12.3 Port Description

Table 6-35 GTP\_OUTBUFTCO Port Description

Port Name	Input/Output	Description
I	Input	Input Signal
OB	Output	Differential N output signals
O	Output	Differential P output signals
T	Input	Tri-state input enable signal, high for tri-state and low for output

### 6.12.4 Paramater Description

Table 6-36 GTP\_OUTBUFTCO Parameter Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "SSTL18D_I", "SSTL18D_II", "SSTL15D_I", etc.

### 6.12.5 Instance Template

```
GTP_OUTBUFTCO#(
    .IOSTANDARD ("DEFAULT"),
)
GTP_OUTBUFTCO_inst (
    .I    (i),
    .T    (t),
    .OB(ob),
    .O    (o)
);
```

## 6.13 Usage Instructions for GTP\_OUTBUFTDS

### 6.13.1 Supported Devices

Table 6-37 Device Models That Support GTP\_OUTBUFTDS

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.13.2 Description of Functionality

GTP\_OUTBUFTDS is a tri-state output true differential BUFFER. It is shown in the following figure:

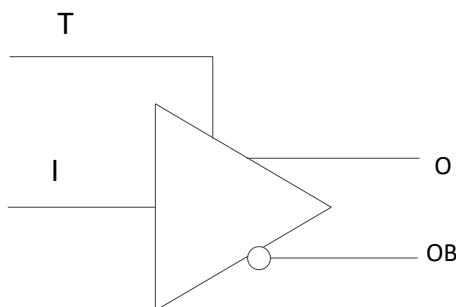


Figure 6-13 GTP\_OUTBUFTDS

### 6.13.3 Port Description

Table 6-38 GTP\_OUTBUFTDS Port Description

Port Name	Input/Output	Description
I	Input	Input Signal

Port Name	Input/Output	Description
OB	Output	Differential N output signals
O	Output	Differential P output signals
T	Input	Tri-state input enable signal, high for tri-state and low for output

#### 6.13.4 Paramater Description

Table 6-39 GTP\_OUTBUFTDS Parameter Description

Parameter	Description	Defaults	Setting Value
IOSTANDARD	IO Standard	"DEFAULT"	The standards supported by IO include "MINI-LVDS", "SUB-LVDS", "TMDS", "DEFAULT", etc.

#### 6.13.5 Instance Template

```
GTP_OUTBUFTDS#(
    .IOSTANDARD ("DEFAULT"),
    .IOTANDARD ("DEFAULT"),
    .GTP_OUTBUFTDS_inst (
        .I (i),
        .T (t),
        .OB (ob ),
        .O (o)
    );
)
```

### 6.14 Usage Instructions for GTP\_IODELAY

#### 6.14.1 Supported Devices

Table 6-40 Device Models That Support GTP\_IODELAY

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 6.14.2 Description of Functionality

Each IO PAD includes an IO DELAY unit, which can be used for providing an input or output delay of 16 steps at approximately 25ps per step; it can offer a DELAY mode that is either statically configured or dynamically adjusted. IO DELAY is typically used to adjust the sampling window or

the output timing. The structure block diagram is shown below.

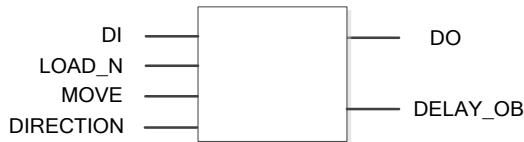


Figure 6-14 GTP\_IODELAY

### 6.14.3 Port Description

Table 6-41 GTP\_IODELAY Port Description

Port Name	Input/Output	Description
DI	Input	Input data
LOAD_N	Input	Active-low, resets the delay step to the DELAY_STEP value.
MOVE	Input	The falling edge triggers dynamic fine-tuning, increasing or decreasing one step depending on the DIRECTION.
DIRECTION	Input	0: Dynamically increase delay step; 1: Dynamically decrease delay step
DO	Output	Output data
DELAY_OB	Output	Overflow indicator. It goes high when internal delay step is 127 with DIRECTION reset; it goes high when delay step is 0 with DIRECTION set.

### 6.14.4 Parameter Description

Table 6-42 GTP\_IODELAY Parameter Description

Parameter	Description	Defaults	Setting Value
DELAY_STEP	Step count setting; 0~127 corresponds to a delay of 1~128 steps	0	0 ~ 127
DELAY_DEPTH	Step count bit width setting	7	7, 4. Default value = 7

### 6.14.5 Instance Template

#### GTP\_IODELAY

```

#(
    .DELAY_STEP (7'd0),          //set delay parameter
    .DELAY_DEPTH  (7)
)
I_GTP_IODELAY(
    .DI      (din),
    .DO      (dout),
    .LOAD_N (load_n),
)
  
```

```
.MOVE      (move),
.DIRECTION (direction),
DELAY_OB   (delay_ob)
);
```

## 6.15 Usage Instructions for GTP\_IOBUF\_RX\_MIPI

### 6.15.1 Supported Devices

Table 6-43 Device Models That Support GTP\_IOBUF\_RX\_MIPI

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.15.2 Description of Functionality

GTP\_IOBUF\_RX\_MIPI supports MIPI DPHY high-speed input, and single-ended input/output in low power mode (LP).

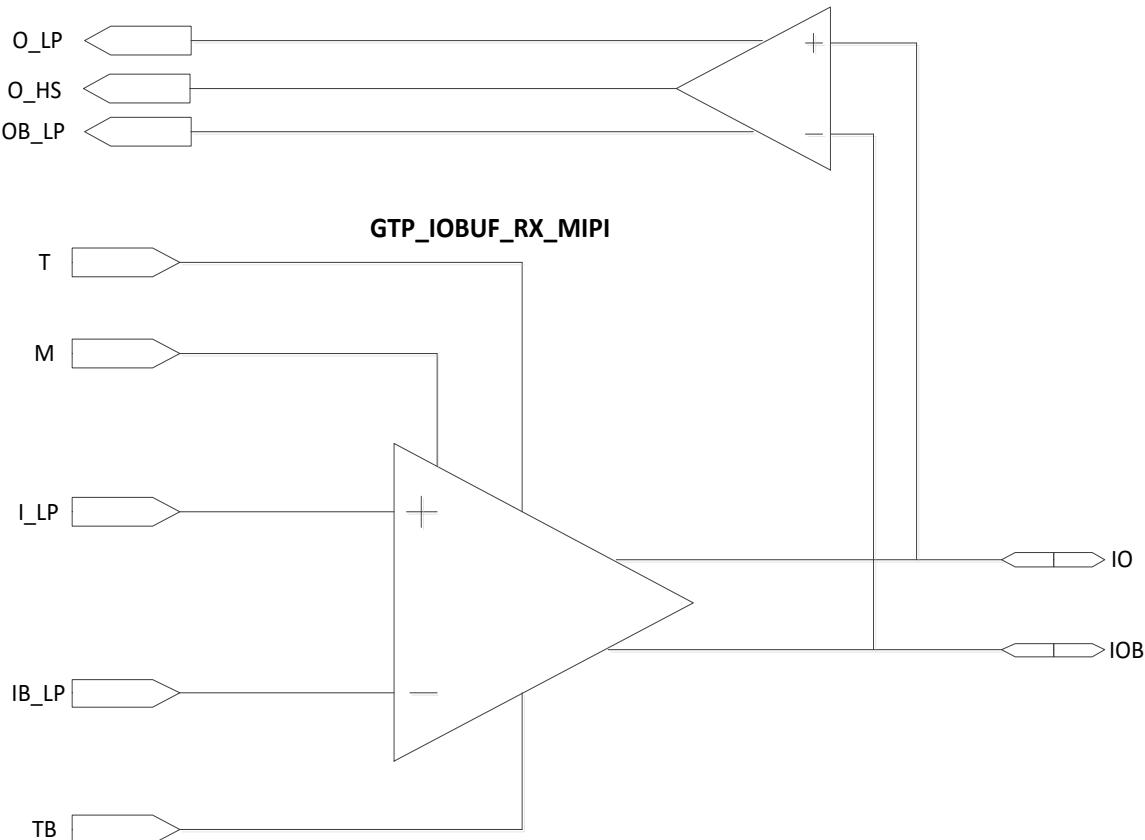


Figure 6-15 GTP\_IOBUF\_RX\_MIPI

### 6.15.3 Port Description

Table 6-44 GTP\_IOBUF\_RX\_MIPI Port Description

Port Name	Input/Output	Description
I_LP	Input	The input signal for the single-ended output buffer from "fabric" in LP mode
IB_LP	Input	The input signal for the single-ended output buffer from "fabric" in LP mode
M	Input	Mode selection signal. 1: Differential input in HS mode; 0: from "fabric" in LP mode
T	Input	Single-ended output enable signal; when it is 0, IO serves as output, and when it is 1, IO serves as input
TB	Input	Single-ended output enable signal; when it is 0, IOB serves as output, and when it is 1, IOB serves as input
O_HS	Output	Differential output (HS) to "fabric"
O_LP	Output	Single-ended (LP) output
OB_LP	Output	Single-ended (LP) output
IO	Bi-Directional	Noninverting differential input (HS) and input/output (LP)
IOB	Bi-Directional	Reverse differential input (HS) and input/output (LP)

### 6.15.4 Paramater Description

Table 6-45 GTP\_IOBUF\_RX\_MIPI Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
IOSTANDARD	string	MIPI	DEFAULT	Input IO standard
DRIVE_STRENGTH	string	"2","6"	"6"	Drive current strength
SLEW_RATE	string	"SLOW" , "FAST"	"SLOW"	Slew rate
TERM_DIFF	string	"ON", "OFF"	"OFF"	When in differential input, the built-in terminal resistor is enabled or disabled
HPIO	string	"TRUE", "FALSE"	"FALSE"	When mapped to HPIO BANK, it is "TRUE"; when mapped to HRIO BANK, it is "FALSE";

Note: The Logos Family does not support HPIO, therefore the parameter HPIO must be set to "FALSE".

### 6.15.5 Instance Template

```
GTP_IOBUF_RX_MIPI #(
    .HPIO("FALSE"),
    .IOSTANDARD("DEFAULT"),
```

```

.SLEW_RATE("SLOW"),
.DRIVE_STRENGTH("6"),
.TERM_DIFF("ON")

) GTP_IOBUF_RX_MIPI_inst (
    .IO(), // INOUT
    .IOB(), // INOUT
    .OB_LP(),// OUTPUT
    .O_HS(), // OUTPUT
    .O_LP(), // OUTPUT
    .IB_LP(),// INPUT
    .I_LP(), // INPUT
    .M(), // INPUT
    .T(), // INPUT
    .TB() // INPUT
);

```

## 6.16 Usage Instructions for GTP\_IOBUF\_TX\_MIPI

### 6.16.1 Supported Devices

Table 6-46 Device Models That Support GTP\_IOBUF\_TX\_MIPI

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.16.2 Description of Functionality

GTP\_IOBUF\_TX\_MIPI supports MIPI DPHY tri-state output, including input/output in low-power mode (LP) and differential output in high-speed (HS) mode, with both modes being switchable according to practical application.

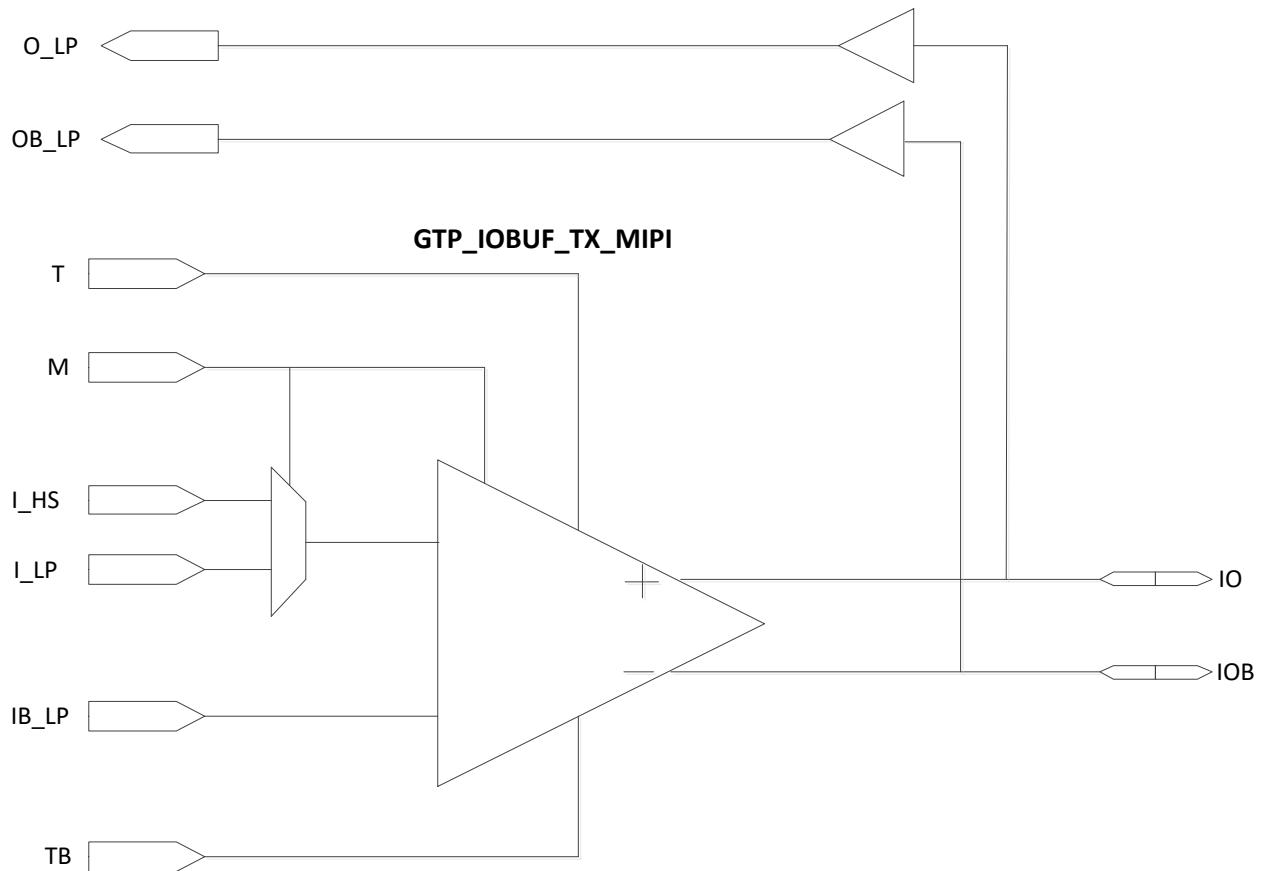


Figure 6-16 GTP\_IOBUF\_TX\_MIPI

## Port Description

Table 6-47 GTP\_IOBUF\_TX\_MIPI Port Description

Port Name	Input/Output	Description
I_HS	Input	In HS mode, the input signal for the differential output buffer comes from IOL
I_LP	Input	In LP mode, the input signal for the single-ended output buffer comes from IOL
IB_LP	Input	In LP mode, the input signal for the single-ended output buffer comes from IOL
M	Input	Mode selection signal. 1: HS mode, differential input; 0: LP mode, single-ended input and output. From IOL
T	Input	Differential and single-ended output enable signal; when it is 0, IO serves as output, and when it is 1, IO serves as input
TB	Input	Single-ended output enable signal; when it is 0, IO serves as output, and when it is 1, IO serves as input
O_LP	Output	Single-ended (LP) input to IOL

Port Name	Input/Output	Description
OB_LP	Output	Single-ended (LP) input to IOL
IO	Bi-Directional	Non-inverting differential output
IOB	Bi-Directional	Reverse differential output

### 6.16.3 Paramater Description

Table 6-48 GTP\_IOBUF\_TX\_MIPI Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
IOSTANDARD	string	MIPI	DEFAULT	Input IO standard VDDIO=1.2V
DRIVE_STRENGTH	string	"2", "6"	"6"	Drive current strength
SLEW_RATE	string	"SLOW", "FAST"	"SLOW"	Slew rate
TERM_DIFF	string	"ON", "OFF"	"ON"	When in differential input, the built-in terminal resistor is enabled or disabled

### 6.16.4 Instance Template

```
GTP_IOBUF_TX_MIPI #
(
    .IOSTANDARD ( "DEFAULT" ),//"MIPI","DEFAULT"
    .DRIVE_STRENGTH ( "2" ),
    .SLEW_RATE ( "FAST" ),
    .TERM_DIFF ( "OFF" )
)
GTP_IOBUF_TX_MIPI_INST
(
    .O_LP    (O_LP),
    .OB_LP   (OB_LP),
    .IO      ( IO ),
    .IOB     ( IOB ),
    .I_HS    ( I_HS ),
    .I_LP    ( I_LP ),
    .IB_LP   ( IB_LP),
    .T       ( T ),
    .TB      ( TB ),
)
```

.M ( M )  
);

## 6.17 Usage Instructions for GTP\_ISERDES

### 6.17.1 Supported Devices

Table 6-49 Device Models That Support GTP\_ISERDES

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.17.2 Description of Functionality

IOL flexibly supports various application interfaces. In addition to the common direct input/output and input/output registers, IOL also supports data input/output rate conversion. GTP\_ISERDES is a data input deserializer that performs double-edge data sampling on both rising and falling edges and supports 1:2, 1:4, 1:7, and 1:8 rate conversion. The structure block diagram is shown below.

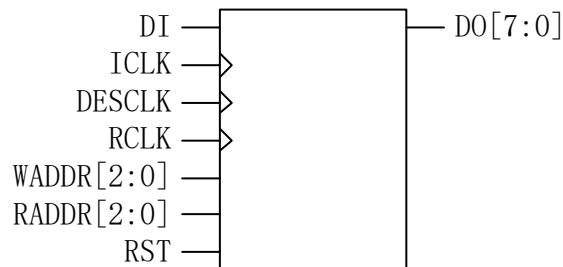


Figure 6-17 GTP\_ISERDES Structure Block Diagram

### 6.17.3 Port Description

Table 6-50 GTP\_ISERDES Port List

Port	Width	Direction	Description
DI	1	Input	Data input
WADDR	3	Input	FIFO write control pointer connected to DDC
RADDR	3	Input	FIFO read control pointer connected to DDC
ICLK	1	Input	Input high-speed clock
DESCLK	1	Input	Deserialization clock
RCLK	1	Input	Input system clock
RST	1	Input	Active-high reset signal
DO	8	Output	Data output

#### 6.17.4 Parameter Description

Table 6-51 GTP\_ISERDES Parameter List

Parameter Name	Parameter Type	Valid Values	Defaults	Description
ISERDES_MODE	String	"IDDR" "IMDDR" "IDES4" "IMDES4" "IDES7" "IDES8" "IMDES8"	"IDDR"	Deserializer modes: "IDDR": General DDR 1:2 deserialization mode "IMDDR": Memory DDR 1:2 deserialization mode "IDES4": General DDR 1:4 deserialization mode "IMDES4": Memory DDR 1:4 deserialization mode "IDES7": General DDR 1:7 deserialization mode "IDES8": General DDR 1:8 deserialization mode "IMDES8": Memory DDR 1:8 deserialization mode
GRS_EN	String	"TRUE" "FLASE"	"TRUE"	Global reset enable signal "TRUE": Enabled; "FALSE": Disabled
LRS_EN	String	"TRUE" "FLASE"	"TRUE"	Local reset enable signal "TRUE": Enabled; "FALSE": Disabled

#### 6.17.5 Instantiation template

```

GTP_ISERDES #(
    .ISERDES_MODE("IDDR"),
    .GRS_EN("TRUE"),
    .LRS_EN("TRUE")
) GTP_ISERDES_inst (
    .DO(),      // OUTPUT[7:0]
    .RADDR(), // INPUT[2:0]
    .WADDR(), // INPUT[2:0]
    .DESCLK(),// INPUT
    .DI(),      // INPUT
    .ICLK(),   // INPUT
    .RCLK(),   // INPUT
    .RST()     // INPUT
);

```

## 6.18 Usage Instructions for GTP\_OSERDES

### 6.18.1 Supported Devices

Table 6-52 Device Models That Support GTP\_OSERDES

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 6.18.2 Description of Functionality

IOL flexibly supports various application interfaces. In addition to the common direct input/output and input/output registers, IOL also supports data input/output rate conversion. GTP\_OSERDES is a DDR data output conversion module that supports 2:1, 4:1, 7:1, and 8:1 rate conversion. The structure block diagram is shown below.

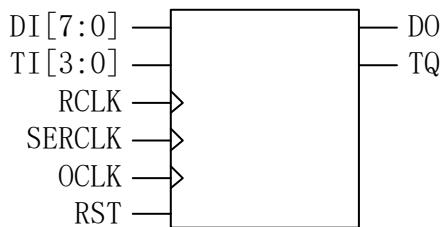


Figure 6-18 GTP\_OSERDES Structure Block Diagram

### 6.18.3 Port Description

Table 6-53 GTP\_OSERDES Port Description

Port	Width	Direction	Description
DI	Input	8	Data input
TI	Input	4	Strobe data input
RCLK	Input	1	Input system clock
SERCLK	Input	1	Serial Clock
OCLK	Input	1	High-speed input Clock
RST	Input	1	Active-high reset signal
DO	Output	1	Data output
TQ	Output	1	Output strobe

### 6.18.4 Parameter Description

Table 6-54 GTP\_OSERDES Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
OSERDES_MODE	String	"ODDR" "OMDDR" "OSER4" "OMSER4" "OSER7" "OSER8" "OMSER8"	"ODDR"	Output conversion modes: "ODDR": General DDR 2:1 output conversion mode "OMDDR": Memory DDR 2:1 output conversion mode "OSER4": General DDR 4:1 output conversion mode "OMSER4": Memory DDR 4:1 output conversion mode "OSER7": General DDR 7:1 output conversion mode "OSER8": General DDR 8:1 output conversion mode "OMSER8": Memory DDR 8:1 output conversion mode
WL_EXTEND	String	"TRUE" "FALSE"	"FALSE"	Write Leveling extension
GRS_EN	String	"TRUE" "FLASE"	"TRUE"	Global reset enable signal "TRUE": Enabled; "FALSE": Disabled
LRS_EN	String	"TRUE" "FLASE"	"TRUE"	Local reset enable signal "TRUE": Enabled; "FALSE": Disabled
TSDDR_INIT	Bit	1'b0,1'b1	1'b0	TO initial state control, 1'b0: TO initial state is 0, 1'b1: TO initial state is 1

### 6.18.5 Instantiation template

```
GTP_OSERDES #(
    .OSERDES_MODE("ODDR"),
    .WL_EXTEND("FALSE"),
    .GRS_EN("TRUE"),
    .LRS_EN("TRUE"),
    .TSDDR_INIT(1'b0)
```

```
) GTP_OSERDES_inst (
    .DI(),      // INPUT[7:0]
    .TI(),      // INPUT[3:0]
    .DO(),      // OUTPUT
    .TQ(),      // OUTPUT
    .OCLK(),    // INPUT
    .RCLK(),    // INPUT
    .RST(),     // INPUT
```

```
.SERCLK() // INPUT  
);
```

## Chapter 7 Usage Instructions for Clock-Related GTPs

---

### 7.1 GTP\_CLKBUFG

#### 7.1.1 Supported Devices

Table 7-1 Device Models That Support GTP\_CLKBUFG

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 7.1.2 Description of Functionality

GTP\_CLKBUFG provides a simple clock BUFFER function and can be instantiated to implement the global clock.

#### 7.1.3 Port Description

Table 7-2 GTP\_CLKBUFG Port Description

Port Signal	Input/Output	Description
CLKIN	Input	Input Clock
CLKOUT	Output	Output clock

#### 7.1.4 Instantiate Module

```
GTP_CLKBUFG U_CLKBUFG (
    .CLKOUT      (CLKOUT),
    .CLKIN      (CLKIN )
);
```

#### 7.1.5 Detailed Description of Functionality

GTP\_CLKBUFG provides a simple clock BUFFER function and can be instantiated to implement the global clock.

## 7.2 GTP\_CLKBUFGCE

### 7.2.1 Supported Devices

Table 7-3 Device Models That Support GTP\_CLKBUFGCE

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 7.2.2 Description of Functionality

GTP\_CLKBUFGCE is a clock BUFFER with an enable control port and can be instantiated to implement the global clock.

### 7.2.3 Port Description

Table 7-4 GTP\_CLKBUFGCE Port Description

Port Signal	Input/Output	Description
CLK	Input	Input Clock
CE	Input	Active-high clock enable signal
CLKOUT	Output	Output clock

### 7.2.4 Parameter Description

Table 7-5 GTP\_CLKBUFGCE Parameter Description

Parameter	Description
DEFAULT_VALUE	0/1; When CE is 0, it controls the state of the output clock CLKOUT

### 7.2.5 Instantiate Module

```

GTP_CLKBUFGCE
#(.DEFAULT_VALUE      (1'b0      )  //1'b0; 1'b1
) I_GTP_CLKBUFGCE (
    .CLKIN      (CLK      ),
    .CE        ( CE      ),
    .CLKOUT     (CLKOUT  )
);

```

## 7.2.6 Detailed Description of Functionality

When enable CE="1" or "0", the corresponding waveform diagrams are as follows:

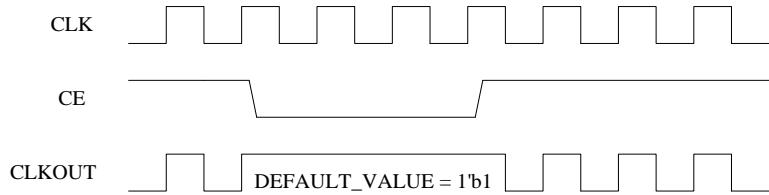


Figure 7-1 GTP\_CLKBUFGCE Waveform Diagram (DEFAULT\_VALUE=1)

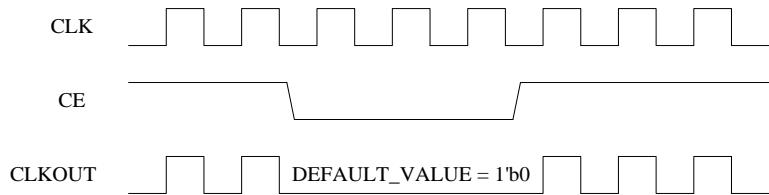


Figure 7-2 GTP\_CLKBUFGCE Waveform Diagram (DEFAULT\_VALUE=0)

## 7.3 GTP\_CLKBUFGMUX

### 7.3.1 Supported Devices

Table 7-6 Device Models That Support GTP\_CLKBUFGMUX

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 7.3.2 Description of Functionality

GTP\_CLKBUFGMUX can be used for dynamic switching between two global clock sources, allowing users to choose forced switching (corresponding to TRIGGER\_MODE="NORMAL") or debounced switching triggered by the falling edge of the clock (corresponding to TRIGGER\_MODE="NEGEDGE"). Note that forced switching may introduce glitches, and users need to reset the related logic after switching to achieve the intended functionality.

### 7.3.3 Port Description

Table 7-7 GTP\_CLKBUFGMUX Port Description

Port Signal	Input/Output	Description
CLKIN0	Input	Input clock CLKIN0
CLKIN1	Input	Input clock CLKIN1
SEL	Input	Clock selection signal: 0 for CLKIN0; 1 for CLKIN1;
CLKOUT	Output	Output clock

### 7.3.4 Parameter Description

Table 7-8 GTP\_CLKBUFGMUX Parameter Description

Parameter Name	Setting Value	Defaults	Description
TRIGGER_MODE	"NORMAL" "NEGEDGE" "POSEDGE"	"NORMAL"	(1) "NORMAL": In this mode, input clocks (CLKIN0 and CLKIN1) can be freely switched and the GLITCHLESS function is not available (2) "NEGEDGE": In this mode, input clocks (CLKIN0 and CLKIN1) can be freely switched and the GLITCHLESS function triggered on the falling edge of the clock is available. (3) "POSEDGE": In this mode, input clocks (CLKIN0 and CLKIN1) can be freely switched and the GLITCHLESS function triggered on the rising edge of the clock is available.
SIM_DEVICE	"TITAN" "LOGOS" "COMP" "LOGOS2" "TITAN2"	"TITAN"	Simulation Model Device Identification "TITAN": When triggered on a rising or falling edge, it switches after one beat in each of the two clock domains "LOGOS", "COMP", "LOGOS2", and "TITAN2": When triggered on a rising or falling edge, it switches after two beats in each of the two clock domains

Notes:

1. The Logos Family only supports "NORMAL" and "NEGEDGE" modes. For modes supported by other devices, please refer to the corresponding clock resource user guide.
2. When TRIGGER\_MODE is set to "NEGEDGE" or "POSEDGE", the clocks can be only switched when the rising or falling edge of the clock is captured in both clock domains. Therefore, the GTP can only be used to switch clocks normally when both CLKIN0/CLKIN1 ports have a rising or falling edge.

### 7.3.5 Instantiate Module

GTP\_CLKBUFGMUX

```
#(
    .TRIGGER_MODE ("NORMAL"),
    .SIM_DEVICE("TITAN")
```

```
)I_CLKBUFGMUX (
    .CLKIN0      (clkin0 ),
    .CLKIN1      (clkin1 ),
    .SEL         (sel   ),
    .CLKOUT     (clkout )
);
```

### 7.3.6 Detailed Description of Functionality

Instantiated GTP\_CLKBUFGMUX can be used for dynamic switching between two global clock input sources, allowing users to choose forced switching (corresponding to TRIGGER\_MODE="NORMAL") or debounced switching triggered by the falling edge of the clock (corresponding to TRIGGER\_MODE="NEGEDGE"). Note that forced switching may introduce glitches, and users need to reset the related logic after switching to achieve the intended functionality. When the TRIGGER\_MODE parameter is set to "NORMAL", the CLKOUT is switched immediately after the SEL signal toggles. The corresponding timing diagram is as follows:

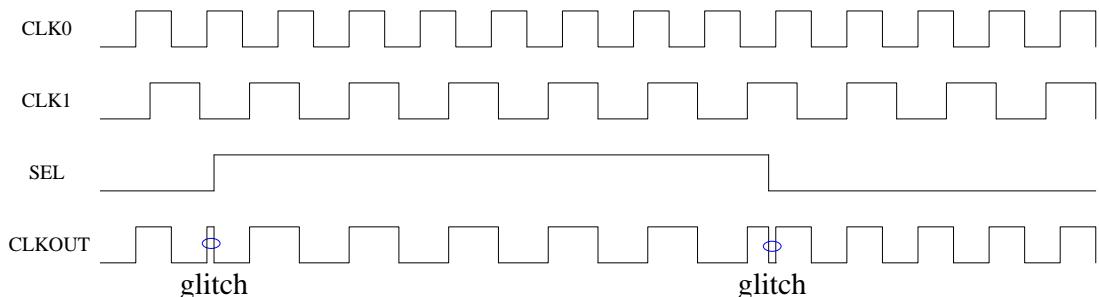


Figure 7-3 Timing Diagram for GTP\_CLKBUFGMUX with Parameter TRIGGER\_MODE="NORMAL"

When the parameter TRIGGER\_MODE is set to "NEGEDGE" and the value of the parameter SIM\_DEVICE is "LOGOS", if SEL switches from 0 to 1, it requires detecting the falling edge of the current clock CLKIN0 twice (maintaining the CLKOUT output as the current clock during the detection of the rising edge of the current clock) and then the falling edge of the switch-to clock CLKIN1 twice (with the CLKOUT output low during the detection of the falling edge of the switch-to clock). After the detection is completed, CLKOUT switches from CLKIN0 to CLKIN1. Otherwise, if SEL changes from 1 to 0, it requires detecting the falling edge of CLKIN1 twice (maintaining the CLKOUT output as the current clock during the detection of the rising edge of the current clock) and then the falling edge of CLKIN0 twice (with the CLKOUT output low during the detection of the falling edge of the switch-to clock CLKIN0). After detection is completed,

CLKOUT switches from CLKIN1 to CLKIN0. The corresponding timing diagram is as follows:

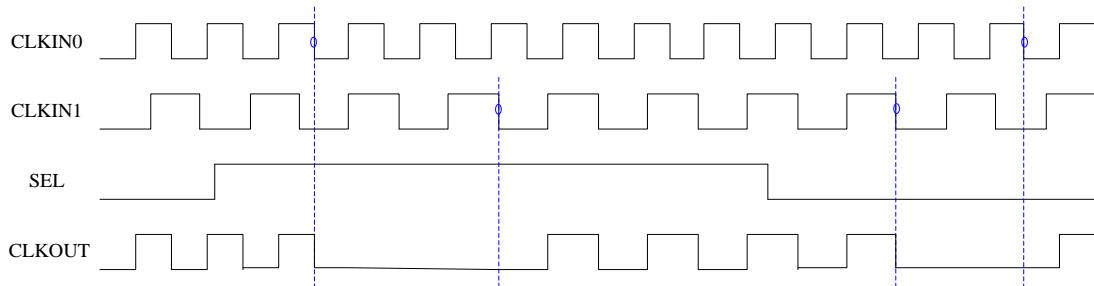


Figure 7-4 Timing Diagram for GTP\_CLKBUFGMUX with Parameter TRIGGER\_MODE="NEGEDGE"

## 7.4 GTP\_CLKBUFR

### 7.4.1 Supported Devices

Table 7-9 Device Models That Support GTP\_CLKBUFR

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 7.4.2 Description of Functionality

GTP\_CLKBUFR is a regional clock BUFFER, where the regional clock drives the unit logic of a certain area.

### 7.4.3 Port Description

Table 7-10 GTP\_CLKBUFR Port Description

Port Signal	Input/Output	Description
CLKIN	Input	Input Clock
CLKOUT	Output	Output clock

### 7.4.4 Instantiate Module

```
GTP_CLKBUFR U_BUFR(
    .CLKOUT      (CLKOUT),
    .CLKIN       (CLKIN  )
);
```

#### 7.4.5 Detailed Description of Functionality

GTP\_CLKBUFR is a regional clock BUFFER, where the regional clock drives the unit logic of a certain area.

The figure below shows the input and output waveform diagram of GTP\_CLKBUFR.

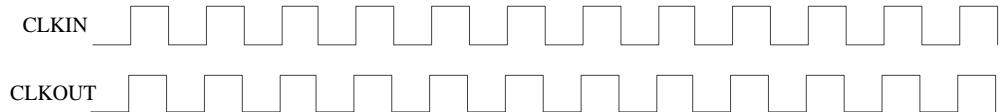


Figure 7-5 GTP\_CLKBUFR Waveform Diagram

### 7.5 GTP\_IOCLKBUF

#### 7.5.1 Supported Devices

Table 7-11 Device Models That Support GTP\_IOCLKBUF

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 7.5.2 Description of Functionality

Like a "gate", GTP\_IOCLKBUF can be instantiated to turn off or on the IO clock.

#### 7.5.3 Port Description

Table 7-12 GTP\_IOCLKBUF Port Description

Port Signal	Input/Output	Description
CLKIN	Input	Input Clock
DI	Input	Enable signal
CLKOUT	Output	Output clock

#### 7.5.4 Parameter Description

Table 7-13 Parameter Description of GTP\_IOCLKBUF

Parameter	Description
GATE_EN	When "TRUE", the enable signal DI is valid; when "FALSE", the enable signal DI is invalid;

### 7.5.5 Instantiate Module

**GTP\_IOCLKBUF**

```
#(
    .GATE_EN      ("FALSE"  ), //FALSE; TRUE
    ) u_IOCLKBUF (
        .CLKOUT      (CLKOUT  ),
        .CLKIN       (CLIN     ),
        .DI          (DI       )
    );
```

### 7.5.6 Detailed Description of Functionality

Like a "gate", GTP\_IOCLKBUF can be instantiated to turn off or on the IO clock. When the parameter GATE\_EN is "TRUE" or "FALSE", the corresponding waveform diagrams are as shown in the two figures below.

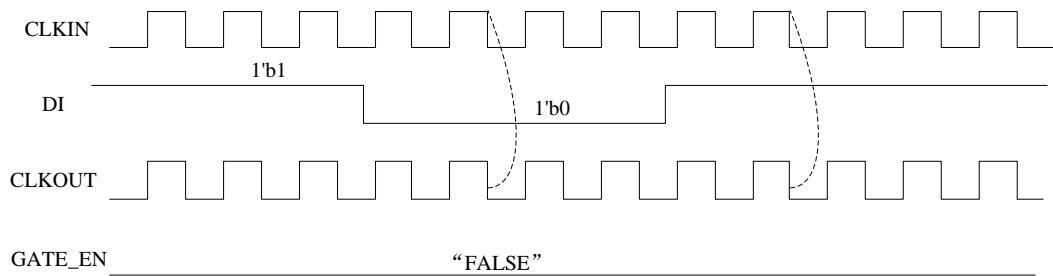


Figure 7-6 Timing Diagram of GTP\_IOCLKBUF with GATEEN="FALSE"

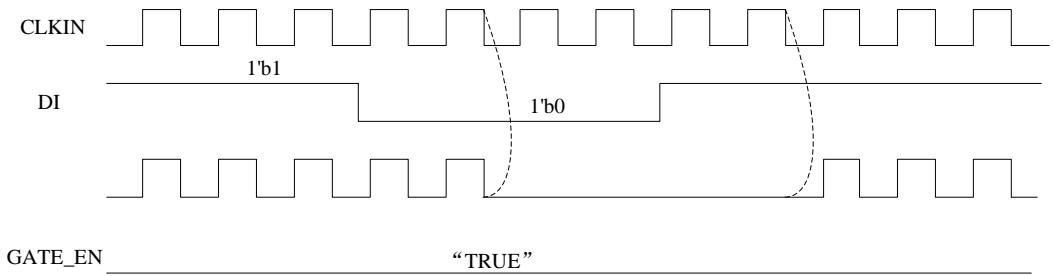


Figure 7-7 Timing Diagram of GTP\_IOCLKBUF with GATEEN="TRUE"

## 7.6 GTP\_IOCLKDIV

### 7.6.1 Supported Devices

Table 7-14 Device Models That Support GTP\_IOCLKDIV

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 7.6.2 Description of Functionality

GTP\_IOCLKDIV divides the clock from GTP\_IOCLKBUF by factors of 2, 3.5, 4, and 5.

### 7.6.3 Port Description

Table 7-15 GTP\_IOCLKDIV Port Description

Port Signal	Input/Output	Description
CLKIN	Input	Input Clock
RST_N	Input	Active-low reset
CLKDIVOUT	Output	Divided output clock

### 7.6.4 Parameter Description

Table 7-16 GTP\_IOCLKDIV Parameter Description

Parameter	Description
DIV_FACTOR	When DIV_FACTOR = "DIV_DIS", CLKOUT=1'b1 and CLKDIVOUT=1'b1; when DIV_FACTOR is a division ratio of "2", "3.5", or "4", CLKOUT=CLKIN and CLKDIVOUT outputs the division clock.
GRS_EN	Global enable signal, "TRUE" indicates the global enable signal is active; "FALSE" indicates the global enable signal is inactive.

### 7.6.5 Instantiate Module

```

GTP_IOCLKDIV #(
    .DIV_FACTOR ("2"      ),      // "2"; "3.5"; "4"; "5";
    .GRS_EN     ("FALSE"   ) // "TRUE"; "FALSE"
)
I_IOCLKDIV(
    .CLKIN      (CLKIN    ),
    .RST_N      (RST_N    ),
    .CLKDIVOUT (CLKDIVOUT)
)

```

);

## 7.6.6 Detailed Description of Functionality

GTP\_IOCLKDIV divides the clock from GTP\_IOCLKBUF by factors of 2, 3.5, 4, and 5. The clock after division can be used as an input source for the global clock.

## 7.7 GTP\_IOCLKDELAY

### 7.7.1 Supported Devices

Table 7-17 Device Models That Support GTP\_IOCLKDELAY

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 7.7.2 Description of Functionality

To meet the requirements of adjusting the input clock phase in some applications, the clock input pin provides an optional IOCLKDELAY function. Users can statically configure the delay step (adjust the number of delay chains, with each step delaying by 25ps; the minimum delay chain count is 0, and the maximum is 127 (8'b0–8'b0111\_1111)), or dynamically obtain the delay step using the on-chip DLL. Furthermore, dynamic fine-tuning of the delay step can be performed through user control logic on top of dynamic/static configuration.

### 7.7.3 Port Description

Table 7-18GTP\_IOCLKDELAY Port Description

Port Signal	Input/Output	Description
CLKIN	Input	Clock input from pin
DELAY_STEP[7:0]	Input	Delay step from DLL
DIRECTION	Input	Set to 0 to dynamically increase the delay step; set to 1 to dynamically decrease the delay step.
LOAD	Input	Active-high, reset the delay step to DELAY_STEP or a static delay value.
MOVE	Input	The falling edge triggers dynamic fine-tuning, increasing or decreasing one step depending on the DIRECTION.
DELAY_OB	Output	Dynamic fine-tuning overflow flag for delay step; goes high when internal delay step is 255 with DIRECTION set to 0; goes high when internal delay step is 0 with DIRECTION set to 1.
CLKOUT	Output	Delayed clock output

#### 7.7.4 Parameter Description

Table 7-19GTP\_IOCLKDELAY Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
DELAY_STEP_VAL UE	binary	8'd0~8'd255	8'd0	Static delay step, effective when the parameter DELAY_STEP_SEL is set to "PARAMETER"
DELAY_STEP_SEL	string	"PARAMETER", "PORT"	"PARAMETER"	"PORT" selects the DELAY_STEP from the DLL as the reference delay step; "PARAMETER" selects CLKDLY_STEP as the reference delay step
SIM_DEVICE	string	"TITAN", "LOGOS", "COMP"	"TITAN"	When set to "TITAN", the maximum CLOCK STEP is 255; when set to "LOGOS" or "COMP", the maximum CLOCK STEP is 127.

#### 7.7.5 Instantiate Module

```
GTP_IOCLKDELAY #(
    .DELAY_STEP_VALUE('b00000000),
    .DELAY_STEP_SEL("PARAMETER"),
    .SIM_DEVICE("TITAN")
) <InstanceName> (
    .DELAY_STEP(), // INPUT[7:0]
    .CLKOUT(), // OUTPUT
    .DELAY_OB(), // OUTPUT
    .CLKIN(), // INPUT
    .DIRECTION(), // INPUT
    .LOAD(), // INPUT
    .MOVE() // INPUT
);
```

#### 7.7.6 Detailed Description of Functionality

Refer to "**UG020004\_Logos Family FPGAs Clock Resources (Clock) User Guide**"

## 7.8 GTP\_OSC\_E1

### 7.8.1 Supported Devices

Table 7-20 Device Models That Support GTP\_OSC\_E1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Not supported	Not supported	Not supported	Not supported

### 7.8.2 Description of Functionality

Provide user-configurable output clock.

### 7.8.3 Port Description

Table 7-21 GTP\_OSC\_E1 Port Description

Port Name	Types	Description
CLKOUT	Output	Output clock, with frequency adaptable to the setting of parameter values.
RST_N	Output	Reset (active-low)
EN	Input	Active-high enable signal

### 7.8.4 Parameter Description

Table 7-22 GTP\_OSC\_E1 Parameter Description

Parameter	Defaults	Description
CLK_DIV	2	Clock division ratio setting, valid values are 0–127

### 7.8.5 Instantiate Module

GTP\_OSC\_E1

```
#(
    .CLK_DIV      (2),
    .U_OSC_E1 (
        .CLKOUT(CLKOUT),
        .EN      (EN),
        .RST_N   (RST_N)
    );
)
```

### 7.8.6 Detailed Description of Functionality

Provide user-configurable output clock, which can serve as an input source for the global clock. The relationship between the output clock and the configured division factor is as follows:

Table 7-23 GTP\_OSC\_E1 Division Factor and Output Clock Relationship

CLK_DIV	CLKOUT(MHz)
0	200/128
1	200/1
2	200/2
3	200/3
...	...
125	200/125
126	200/126
127	200/127

## 7.9 GTP\_OSC\_E3

### 7.9.1 Supported Devices

Table 7-24 Device Models That Support GTP\_OSC\_E3

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Not supported	Supported	Supported	Supported	Supported

### 7.9.2 Description of Functionality

Provide user-configurable output clock.

### 7.9.3 Port Description

Table 7-25 GTP\_OSC\_E3 Port Description

Port Name	Types	Description
CLKOUT	Output	Output clock, with frequency adaptable to the setting of parameter values.
RST_N	Output	Reset (active-low)
EN_N	Input	Active-high enable signal

### 7.9.4 Parameter Description

Table 7-26 GTP\_OSC\_E3 Parameter Description

Parameter	Defaults	Description
-----------	----------	-------------

CLK_DIV	2	Clock division ratio setting, valid values are 0–127
---------	---	--

### 7.9.5 Instantiate Module

GTP\_OSC\_E3

```
#(
    .CLK_DIV      (2),
    ) U_OSC_E3(
        .CLKOUT(CLKOUT),
        .EN_N      (EN),
        .RST_N     (RST_N)
    );
)
```

### 7.9.6 Detailed Description of Functionality

Provide user-configurable output clock, which can serve as an input source for the global clock.

Table 7-27 GTP\_OSC\_E3 Division Factor and Output Clock Relationship

CLK_DIV	CLKOUT(MHz)
0	200/128
1	200/1
2	200/2
3	200/3
...	...
125	200/125
126	200/126
127	200/127

## 7.10 GTP\_CRYSTAL

### 7.10.1 Supported Devices

Table 7-28 Device Models That Support GTP\_CRYSTAL

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 7.10.2 Description of Functionality

The output clock of GTP\_CRYSTAL can drive logic cells as a global clock, or serve as an IO clock, with ports XTALA and XTALB being differential signals.

### 7.10.3 Port Description

Table 7-29 GTP\_CRYSTAL Port Description

Port Signal	Input/Output	Description
EN_N	Input	Enable signal (active-low)
XTALA	Input	One input pin for the crystal
XTALB	Input	Another input pin for the crystal, where XTALA and XTALB are differential signals
CLKOUT	Output	Output clock

### 7.10.4 Parameter Description

Table 7-30 GTP\_CRYSTAL Parameter Description

Parameter	Description
CLK_EN	Clock enable configuration, with valid values of "TRUE" or "FALSE".

### 7.10.5 Instantiate Module

```
GTP_CRYSTAL #(
    .CLK_EN      ("TRUE"          ), // "TRUE", "FALSE"
    ) u_crystal (
        .CLKOUT(CLKOUT  ),
        .XTALA  (XTALA   ),
        .XTALB  (XTALB   ),
        .EN_N    (EN_N    )
    );
```

### 7.10.6 Detailed Description of Functionality

When the CLK\_EN parameter is configured to TRUE and EN\_N to 0, the waveform diagram is as follows.

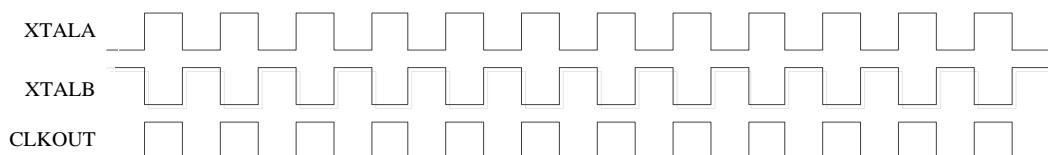


Figure 7-8 GTP\_CRYSTAL Waveform Diagram

## 7.11 GTP\_PLL\_E1

### 7.11.1 Supported Devices

Table 7-31 Device Models That Support GTP\_PLL\_E1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Not supported	Not supported	Not supported	Not supported

### 7.11.2 Description of Functionality

Mainly used for implementing frequency division, frequency multiplication, and phase adjustment. This GTP supports dynamic selection of input clock, internal and external feedback modes, dynamic reconfiguration, and output clock gating functions.

### 7.11.3 Port Description

Table 7-32 GTP\_PLL\_E1 Port Description

Port	Direction	Description
CLKOUT0	Output	Output divider0 divided output clock;
CLKOUT0_EXT	Output	Output divider0 divided output clock, which shares the same source with CLKOUT0;
CLKOUT1	Output	Output divider1 divided output clock;
CLKOUT2	Output	Output divider2 divided output clock;
CLKOUT3	Output	Output divider3 divided output clock;
CLKOUT4	Output	Output divider4 divided output clock or bypass output of the input clock;
CLKOUT5	Output	Output divider0–4, one out of five divided clocks is selected by static configuration CLKOUT5_SEL;
CLKSWITCH_FLAG	Output	Clock auto-switching indicator, in the automatic mode of dynamic clock selection: A value of 0 indicates CLKIN1 is selected as the PLL reference clock; A value of 1 indicates CLKIN2 is selected as the PLL reference clock;
LOCK	Output	PLL lock output signal; 0 indicates PLL is not locked; 1 indicates PLL is locked;
CLKIN1	Input	Input clock 1;
CLKIN2	Input	Input clock 2;
CLKFB	Input	Feedback clock;
CLKIN_SEL	Input	Manual selection port for input clock dynamic selection;
CLKIN_SEL_EN	Input	CLKIN_SEL port enable: When set to 0, the CLKIN_SEL port input is invalid, which is the automatic mode for dynamic selection; When set to 1, the CLKIN_SEL port input is valid, which is the manual mode for dynamic selection;
PFDEN	Input	Active-high PLL PFD enable signal
CLKOUT0_SYN	Input	Output clock CLKOUT0 GATE control, active-high
CLKOUT0_EXT_SYN	Input	Output clock CLKOUT0_EXT GATE control, active-high

Port	Direction	Description
CLKOUT1_SYN	Input	Output clock CLKOUT1 GATE control, active-high
CLKOUT2_SYN	Input	Output clock CLKOUT2 GATE control, active-high
CLKOUT3_SYN	Input	Output clock CLKOUT3 GATE control, active-high
CLKOUT4_SYN	Input	Output clock CLKOUT4 GATE control, active-high
CLKOUT5_SYN	Input	Output clock CLKOUT5 GATE control, active-high
RATIOI	Input	Input divider divider ratio dynamic setting;
RATIO0	Input	Output divider0 divider ratio dynamic setting;
RATIO1	Input	Output divider1 divider ratio dynamic setting;
RATIO2	Input	Output divider2 divider ratio dynamic setting;
RATIO3	Input	Output divider3 divider ratio dynamic setting;
RATIO4	Input	Output divider4 divider ratio dynamic setting;
RATIOF	Input	Feedback divider divider ratio dynamic setting;
DUTY0	Input	Output divider0 duty cycle dynamic setting;
DUTY1	Input	Output divider1 duty cycle dynamic setting;
DUTY2	Input	Output divider2 duty cycle dynamic setting;
DUTY3	Input	Output divider3 duty cycle dynamic setting;
DUTY4	Input	Output divider4 duty cycle dynamic setting;
DUTYF	Input	Feedback divider duty cycle dynamic setting;
PHASE0	Input	Output divider0 fine phase dynamic setting;
PHASE1	Input	Output divider1 fine phase dynamic setting;
PHASE2	Input	Output divider2 fine phase dynamic setting;
PHASE3	Input	Output divider3 fine phase dynamic setting;
PHASE4	Input	Output divider4 fine phase dynamic setting;
PHASEF	Input	Feedback divider fine phase dynamic setting;
CPHASE0	Input	Output divider0 coarse phase dynamic setting;
CPHASE1	Input	Output divider1 coarse phase dynamic setting;
CPHASE2	Input	Output divider2 coarse phase dynamic setting;
CPHASE3	Input	Output divider3 coarse phase dynamic setting;
CPHASE4	Input	Output divider4 coarse phase dynamic setting;
CPHASEF	Input	Feedback divider coarse phase dynamic setting;
PLL_PWD	Input	Active-high power down signal;
RST	Input	Active-high global reset signal;
RSTODIV_PHASE	Input	When the phase adjustment function is enabled, the output reset signal for divider0–4, active-high; When the phase adjustment function is disabled, the output for divider0–4 is controlled by the RST reset signal;

#### 7.11.4 Parameter Description

Table 7-33 GTP\_PLL\_E1 Parameter Description

Parameter Name	Valid Values	Defaults	Description
DYNAMIC_RATIOI_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for input divider frequency division;
STATIC_RATIOI	1~512	1	Static setting for input divider frequency division;
PFDEN_EN	"FALSE","TRUE"	"FALSE"	PLL PFDEN signal setting
VCOCLK_DIV2	1'b0-1'b1	1'b0	PLL VCO output clock 2 division enable;
DYNAMIC_RATIO0_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider0 frequency division;
DYNAMIC_RATIO1_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider1 frequency division;
DYNAMIC_RATIO2_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider2 frequency division;
DYNAMIC_RATIO3_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider3 frequency division;
DYNAMIC_RATIO4_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider4 frequency division;
DYNAMIC_RATIOF_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for feedback divider frequency division;
STATIC_RATIO0	1~512	1	Static setting for output divider0 frequency division;
STATIC_RATIO1	1~512	1	Static setting for output divider1 frequency division;
STATIC_RATIO2	1~512	1	Static setting for output divider2 frequency division;
STATIC_RATIO3	1~512	1	Static setting for output divider3 frequency division;
STATIC_RATIO4	1~512	1	Static setting for output divider4 frequency division;
STATIC_RATIOF	1~512	1	Static setting for feedback divider frequency division;
DYNAMIC_DUTY0_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider0 duty;
DYNAMIC_DUTY1_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider1 duty;
DYNAMIC_DUTY2_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider2 duty;
DYNAMIC_DUTY3_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider3 duty;
DYNAMIC_DUTY4_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider4 duty;
DYNAMIC_DUTYF_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for feedback divider duty;
STATIC_DUTY0	1~1022	1	Output divider0 duty static setting;
STATIC_DUTY1	1~1022	1	Output divider1 duty static setting;
STATIC_DUTY2	1~1022	1	Output divider2 duty static setting;
STATIC_DUTY3	1~1022	1	Output divider3 duty static setting;
STATIC_DUTY4	1~1022	1	Output divider4 duty static setting;
STATIC_DUTYF	1~1022	1	Feedback divider duty static setting;
PHASE_ADJUST0_EN	"FALSE","TRUE"	"FALSE"	Output divider0 phase adjustment enable;

Parameter Name	Valid Values	Defaults	Description
PHASE_ADJUST1_EN	"FALSE", "TRUE"	"FALSE"	Output divider1 phase adjustment enable;
PHASE_ADJUST2_EN	"FALSE", "TRUE"	"FALSE"	Output divider2 phase adjustment enable;
PHASE_ADJUST3_EN	"FALSE", "TRUE"	"FALSE"	Output divider3 phase adjustment enable;
PHASE_ADJUST4_EN	"FALSE", "TRUE"	"FALSE"	Output divider4 phase adjustment enable;
DYNAMIC_PHASE0_EN	"FALSE", "TRUE"	"FALSE"	Dynamic control enable for output divider0 phase;
DYNAMIC_PHASE1_EN	"FALSE", "TRUE"	"FALSE"	Dynamic control enable for output divider1 phase;
DYNAMIC_PHASE2_EN	"FALSE", "TRUE"	"FALSE"	Dynamic control enable for output divider2 phase;
DYNAMIC_PHASE3_EN	"FALSE", "TRUE"	"FALSE"	Dynamic control enable for output divider3 phase;
DYNAMIC_PHASE4_EN	"FALSE", "TRUE"	"FALSE"	Dynamic control enable for output divider4 phase;
DYNAMIC_PHASEF_EN	"FALSE", "TRUE"	"FALSE"	Dynamic control enable for feedback divider phase;
STATIC_PHASE0	0~7	0	Output divider0 fine phase static setting;
STATIC_PHASE1	0~7	0	Output divider1 fine phase static setting;
STATIC_PHASE2	0~7	0	Output divider2 fine phase static setting;
STATIC_PHASE3	0~7	0	Output divider3 fine phase static setting;
STATIC_PHASE4	0~7	0	Output divider4 fine phase static setting;
STATIC_PHASEF	0~7	0	Feedback divider fine phase static setting;
STATIC_CPHASE0	2~513	2	Output divider0 coarse phase static setting;
STATIC_CPHASE1	2~513	2	Output divider1 coarse phase static setting;
STATIC_CPHASE2	2~513	2	Output divider2 coarse phase static setting;
STATIC_CPHASE3	2~513	2	Output divider3 coarse phase static setting;
STATIC_CPHASE4	2~513	2	Output divider4 coarse phase static setting;
STATIC_CPHASEF	2~513	2	Feedback divider coarse phase static setting;
CLK_CAS0_EN	"FALSE", "TRUE"	"FALSE"	Output divider0 clock cascade enable;
CLK_CAS1_EN	"FALSE", "TRUE"	"FALSE"	Output divider1 clock cascade enable;
CLK_CAS2_EN	"FALSE", "TRUE"	"FALSE"	Output divider2 clock cascade enable;
CLK_CAS3_EN	"FALSE", "TRUE"	"FALSE"	Output divider3 clock cascade enable;
CLK_CAS4_EN	"FALSE", "TRUE"	"FALSE"	Output divider4 clock cascade enable;
INTERNAL_FB	"DISABLE", "ENABLE"	"ENABLE"	Internal feedback divider selection setting;
EXTERNAL_FB	"CLKOUT0", "CLKOUT1" "CLKOUT2", "CLKOUT3" "CLKOUT4", "DISABLE"	"DISABLE"	External feedback divider selection setting;

Parameter Name	Valid Values	Defaults	Description
BANDWIDTH	"LOW", "HIGH" "OPTIMIZED"	"OPTIMIZE D"	Bandwidth selection settings;
CLKOUT5_SEL	0~4	0	Output clock CLKOUT5 selection setting;
CLKIN_BYPASS_EN	"FALSE", "TRUE"	"FALSE"	Input clock bypass enable;
CLKOUT0_SYN_EN	"FALSE", "TRUE"	"FALSE"	CLKOUT0_SYN port enable;
CLKOUT0_EXT_SYN_EN	"FALSE", "TRUE"	"FALSE"	CLKOUT0_EXT_SYN port enable;
CLKOUT1_SYN_EN	"FALSE", "TRUE"	"FALSE"	CLKOUT1_SYN port enable;
CLKOUT2_SYN_EN	"FALSE", "TRUE"	"FALSE"	CLKOUT2_SYN port enable;
CLKOUT3_SYN_EN	"FALSE", "TRUE"	"FALSE"	CLKOUT3_SYN port enable;
CLKOUT4_SYN_EN	"FALSE", "TRUE"	"FALSE"	CLKOUT4_SYN port enable;
CLKOUT5_SYN_EN	"FALSE", "TRUE"	"FALSE"	CLKOUT5_SYN port enable;
RSTODIV_PHASE_EN	"FALSE", "TRUE"	"FALSE"	RSTODIV_PHASE reset signal enable;
CLKIN_FREQ	5~625	50	Input clock frequency;
SIM_DEVICE	"PGL22G", "PGL35ES"	"PGL22G"	Device selection setting;

### 7.11.5 Instantiate Module

When the input is 50M, the output clock clkout0 50M, the output pll\_lock signal, and the internal feedback instantiation are as follows:

```
GTP_PLL_E1 #(
    .CLKIN_FREQ(50),
    .PFDEN_EN("FALSE"),
    .VCOCLK_DIV2(1'b0),
    .DYNAMIC_RATIOI_EN("FALSE"),
    .DYNAMIC_RATIO0_EN("FALSE"),
    .DYNAMIC_RATIO1_EN("FALSE"),
    .DYNAMIC_RATIO2_EN("FALSE"),
    .DYNAMIC_RATIO3_EN("FALSE"),
    .DYNAMIC_RATIO4_EN("FALSE"),
    .DYNAMIC_RATIOF_EN("FALSE"),
    .STATIC_RATIOI(1),
    .STATIC_RATIO0(1),
    .STATIC_RATIO1(1),
    .STATIC_RATIO2(1),
    .STATIC_RATIO3(1),
    .STATIC_RATIO4(1),
```

```
.STATIC_RATIOF(1),  
.DYNAMIC_DUTY0_EN("FALSE"),  
.DYNAMIC_DUTY1_EN("FALSE"),  
.DYNAMIC_DUTY2_EN("FALSE"),  
.DYNAMIC_DUTY3_EN("FALSE"),  
.DYNAMIC_DUTY4_EN("FALSE"),  
.DYNAMIC_DUTYF_EN("FALSE"),  
.STATIC_DUTY0(2),  
.STATIC_DUTY1(2),  
.STATIC_DUTY2(2),  
.STATIC_DUTY3(2),  
.STATIC_DUTY4(2),  
.STATIC_DUTYF(2),  
.PHASE_ADJUST0_EN("FALSE"),  
.PHASE_ADJUST1_EN("FALSE"),  
.PHASE_ADJUST2_EN("FALSE"),  
.PHASE_ADJUST3_EN("FALSE"),  
.PHASE_ADJUST4_EN("FALSE"),  
.DYNAMIC_PHASE0_EN("FALSE"),  
.DYNAMIC_PHASE1_EN("FALSE"),  
.DYNAMIC_PHASE2_EN("FALSE"),  
.DYNAMIC_PHASE3_EN("FALSE"),  
.DYNAMIC_PHASE4_EN("FALSE"),  
.DYNAMIC_PHASEF_EN("FALSE"),  
.STATIC_PHASE0(0),  
.STATIC_PHASE1(0),  
.STATIC_PHASE2(0),  
.STATIC_PHASE3(0),  
.STATIC_PHASE4(0),  
.STATIC_PHASEF(0),  
.STATIC_CPHASE0(2),  
.STATIC_CPHASE1(2),  
.STATIC_CPHASE2(2),  
.STATIC_CPHASE3(2),
```

```

    .STATIC_CPHASE4(2),
    .STATIC_CPHASEF(2),
    .CLK_CAS0_EN("FALSE"),
    .CLK_CAS1_EN("FALSE"),
    .CLK_CAS2_EN("FALSE"),
    .CLK_CAS3_EN("FALSE"),
    .CLK_CAS4_EN("FALSE"),
    .CLKOUT5_SEL(0),
    .CLKIN_BYPASS_EN("FALSE"),
    .CLKOUT0_SYN_EN("FALSE"),
    .CLKOUT0_EXT_SYN_EN("FALSE"),
    .CLKOUT1_SYN_EN("FALSE"),
    .CLKOUT2_SYN_EN("FALSE"),
    .CLKOUT3_SYN_EN("FALSE"),
    .CLKOUT4_SYN_EN("FALSE"),
    .CLKOUT5_SYN_EN("FALSE"),
    .INTERNAL_FB("ENABLE"),
    .EXTERNAL_FB("DISABLE"),
    .BANDWIDTH("OPTIMIZED"),
    .RSTODIV_PHASE_EN("TRUE"),
    .SIM_DEVICE("PGL22G")

) <InstanceName> (
    .CPHASE0(),           // INPUT[9:0]
    .CPHASE1(),           // INPUT[9:0]
    .CPHASE2(),           // INPUT[9:0]
    .CPHASE3(),           // INPUT[9:0]
    .CPHASE4(),           // INPUT[9:0]
    .CPHASEF(),           // INPUT[9:0]
    .DUTY0(),             // INPUT[9:0]
    .DUTY1(),             // INPUT[9:0]
    .DUTY2(),             // INPUT[9:0]
    .DUTY3(),             // INPUT[9:0]
    .DUTY4(),             // INPUT[9:0]
    .DUTYF(),             // INPUT[9:0]

```

```

.PHASE0(),          // INPUT[2:0]
.PHASE1(),          // INPUT[2:0]
.PHASE2(),          // INPUT[2:0]
.PHASE3(),          // INPUT[2:0]
.PHASE4(),          // INPUT[2:0]
.PHASEF(),          // INPUT[2:0]
.RATIO0(),          // INPUT[9:0]
.RATIO1(),          // INPUT[9:0]
.RATIO2(),          // INPUT[9:0]
.RATIO3(),          // INPUT[9:0]
.RATIO4(),          // INPUT[9:0]
.RATIOF(),          // INPUT[9:0]
.RATIOI(),          // INPUT[9:0]
.CLKOUT0(),         // OUTPUT
.CLKOUT0_EXT(),     // OUTPUT
.CLKOUT1(),         // OUTPUT
.CLKOUT2(),         // OUTPUT
.CLKOUT3(),         // OUTPUT
.CLKOUT4(),         // OUTPUT
.CLKOUT5(),         // OUTPUT
.CLKSWITCH_FLAG(), // OUTPUT
.LOCK(),            // OUTPUT
.CLKFB(),           // INPUT
.CLKIN1(),          // INPUT
.CLKIN2(),          // INPUT
.CLKIN_SEL(),        // INPUT
.CLKIN_SEL_EN(),    // INPUT
.CLKOUT0_EXT_SYN(), // INPUT
.CLKOUT0_SYN(),      // INPUT
.CLKOUT1_SYN(),      // INPUT
.CLKOUT2_SYN(),      // INPUT
.CLKOUT3_SYN(),      // INPUT
.CLKOUT4_SYN(),      // INPUT
.CLKOUT5_SYN(),      // INPUT

```

```

.PFDEN(),           // INPUT
.PLL_PWD(),         // INPUT
.RST(),             // INPUT
.RSTODIV_PHASE()   // INPUT
);

```

### 7.11.6 Detailed Functional Description

Refer to "**UG020004\_Logos Family FPGAs Clock Resources (Clock) User Guide**"

## 7.12 GTP\_PLL\_E3

### 7.12.1 Supported Devices

Table 7-34 Device Models That Support GTP\_PLL\_E3

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Not supported	Supported	Supported	Supported	Supported

### 7.12.2 Description of Functionality

Mainly used for implementing frequency division, frequency multiplication, and phase adjustment. This GTP supports dynamic selection of input clock, internal and external feedback modes, dynamic reconfiguration, and output clock gating functions.

### 7.12.3 Port Description

Table 7-35 GTP\_PLL\_E3 Port Description

Port	Direction	Description
CLKOUT0	Output	Output divider0 divided output clock;
CLKOUT0_EXT	Output	Output divider0 divided output clock, which shares the same source with CLKOUT0;
CLKOUT1	Output	Output divider1 divided output clock;
CLKOUT2	Output	Output divider2 divided output clock;
CLKOUT3	Output	Output divider3 divided output clock;
CLKOUT4	Output	Output divider4 divided output clock or bypass output of the input clock;
CLKOUT5	Output	Output divider0–4, one out of five divided clocks is selected by static configuration CLKOUT5_SEL;
CLKSWITCH_FLAG	Output	Clock auto-switching indicator, in the automatic mode of dynamic clock selection: A value of 0 indicates CLKIN1 is selected as the PLL reference clock;

Port	Direction	Description
		A value of 1 indicates CLKIN2 is selected as the PLL reference clock;
LOCK	Output	PLL lock output signal; 0 indicates PLL is not locked; 1 indicates PLL is locked;
CLKIN1	Input	Input clock 1;
CLKIN2	Input	Input clock 2;
CLKFB	Input	Feedback clock;
CLKIN_SEL	Input	Clock input selection port; when set to 0, select CLKIN1; when set to 1, select CLKIN2;
CLKIN_SEL_EN	Input	CLKIN_SEL port enable: A value of 0 renders the CLKIN_SEL port input invalid, automatic mode; A value of 1 renders the CLKIN_SEL port input valid, with the input clock selected by CLKIN_SEL;
PFDEN	Input	Active-high PLL PFD enable signal
ICP_BASE	Input	Charge Pump current reference setting
ICP_SEL[3:0]	Input	Charge Pump current dynamic setting
LPFRES_SEL[2:0]	Input	Loop Filter resistance dynamic setting
CRIPPLE_SEL	Input	Loop Filter resistance dynamic setting
PHASE_SEL[2:0]	Input	Select the corresponding PLL output clock for phase adjustment; 3'b000: Select CLKOUT0, 3'b001: Select CLKOUT1, 3'b010: Select CLKOUT2, 3'b011: Select CLKOUT3; 3'b100: Select CLKOUT4; 3'b101: Select internal feedback clock;
PHASE_DIR	Input	Select the direction for dynamic phase fine-tuning; 1'b0: lag, 1'b1: lead;
PHASE_STEP_N	Input	Dynamic phase fine-tuning adjustment toggle signal; Each toggle adjusts the output clock phase by 1/8T <sub>vco</sub> ;
LOAD_PHASE	Input	Load signal for the current phase fine-tuning value of the selected channel, pulse signal, active-high;
RATIOI[9:0]	Input	Input divider divider ratio dynamic setting;
RATIOM[6:0]	Input	Feedback M divider divider ratio dynamic setting;
RATIO0[9:0]	Input	Output divider0 divider ratio dynamic setting;
RATIO1[9:0]	Input	Output divider1 divider ratio dynamic setting;
RATIO2[9:0]	Input	Output divider2 divider ratio dynamic setting;
RATIO3[9:0]	Input	Output divider3 divider ratio dynamic setting;
RATIO4[9:0]	Input	Output divider4 divider ratio dynamic setting;
RATIOF[9:0]	Input	Feedback F divider divider ratio dynamic setting;
DUTY0[9:0]	Input	Output divider0 duty cycle dynamic setting;
DUTY1[9:0]	Input	Output divider1 duty cycle dynamic setting;
DUTY2[9:0]	Input	Output divider2 duty cycle dynamic setting;
DUTY3[9:0]	Input	Output divider3 duty cycle dynamic setting;
DUTY4[9:0]	Input	Output divider4 duty cycle dynamic setting;
CLKOUT0_SYN	Input	Output clock CLKOUT0 GATE control, active-high;
CLKOUT0_EXT_SYN	Input	Output clock CLKOUT0_EXT GATE control, active-high;
CLKOUT1_SYN	Input	Output clock CLKOUT1 GATE control, active-high;

Port	Direction	Description
CLKOUT2_SYN	Input	Output clock CLKOUT2 GATE control, active-high;
CLKOUT3_SYN	Input	Output clock CLKOUT3 GATE control, active-high;
CLKOUT4_SYN	Input	Output clock CLKOUT4 GATE control, active-high;
CLKOUT5_SYN	Input	Output clock CLKOUT5 GATE control, active-high;
PLL_PWD	Input	Active-high power down signal;
RST	Input	Active-high global reset signal;
RSTODIV	Input	Output reset signal for divider0~4 and FDIV, active-high;

#### 7.12.4 Paramater Description

Table 7-36 GTP\_PLL\_E3 Parameter Description

Parameter Name	Valid Values	Defaults	Description
CLKIN_FREQ	5~625	50	Input Clock Frequency
PFDEN_EN	"FALSE","TRUE"	"FALSE"	PLL PFDEN signal enable;
VCOCLK_DIV2	1'b0-1'b1	1'b0	PLL VCO output clock 2 division enable;
DYNAMIC_RATIOI_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for input divider frequency division;
DYNAMIC_RATIOM_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for feedback M divider frequency division;
DYNAMIC_RATIO0_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider0 frequency division;
DYNAMIC_RATIO1_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider1 frequency division;
DYNAMIC_RATIO2_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider2 frequency division;
DYNAMIC_RATIO3_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider3 frequency division;
DYNAMIC_RATIO4_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider4 frequency division;
DYNAMIC_RATIOF_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for feedback F divider frequency division;
STATIC_RATIOI	1~512	1	Static setting for input divider frequency division;
STATIC_RATIOM	1~64	1	Static setting for feedback M divider frequency division;
STATIC_RATIO0	1~512	1	Static setting for output divider0 frequency division;
STATIC_RATIO1	1~512	1	Static setting for output divider1 frequency division;
STATIC_RATIO2	1~512	1	Static setting for output divider2 frequency division;
STATIC_RATIO3	1~512	1	Static setting for output divider3 frequency division;
STATIC_RATIO4	1~512	1	Static setting for output divider4 frequency division;
STATIC_RATIOF	1~512	1	Static setting for feedback F divider frequency division;
DYNAMIC_DUTY0_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider0 duty;
DYNAMIC_DUTY1_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output

Parameter Name	Valid Values	Defaults	Description
			divider1 duty;
DYNAMIC_DUTY2_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider2 duty;
DYNAMIC_DUTY3_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider3 duty;
DYNAMIC_DUTY4_EN	"FALSE","TRUE"	"FALSE"	Dynamic control enable for output divider4 duty;
STATIC_DUTY0	2~1022	2	Output divider0 duty static setting;
STATIC_DUTY1	2~1022	2	Output divider1 duty static setting;
STATIC_DUTY2	2~1022	2	Output divider2 duty static setting;
STATIC_DUTY3	2~1022	2	Output divider3 duty static setting;
STATIC_DUTY4	2~1022	2	Output divider4 duty static setting;
STATIC_PHASE0	0~7	0	Output divider0 fine phase static setting;
STATIC_PHASE1	0~7	0	Output divider1 fine phase static setting;
STATIC_PHASE2	0~7	0	Output divider2 fine phase static setting;
STATIC_PHASE3	0~7	0	Output divider3 fine phase static setting;
STATIC_PHASE4	0~7	0	Output divider4 fine phase static setting;
STATIC_PHASEF	0~7	0	Feedback divider fine phase static setting;
STATIC_CPHASE0	0~511	0	Output divider0 coarse phase static setting;
STATIC_CPHASE1	0~511	0	Output divider1 coarse phase static setting;
STATIC_CPHASE2	0~511	0	Output divider2 coarse phase static setting;
STATIC_CPHASE3	0~511	0	Output divider3 coarse phase static setting;
STATIC_CPHASE4	0~511	0	Output divider4 coarse phase static setting;
STATIC_CPHASEF	0~511	0	Feedback divider coarse phase static setting;
CLK_CAS1_EN	"FALSE","TRUE"	"FALSE"	Output divider1 clock cascade enable;
CLK_CAS2_EN	"FALSE","TRUE"	"FALSE"	Output divider2 clock cascade enable;
CLK_CAS3_EN	"FALSE","TRUE"	"FALSE"	Output divider3 clock cascade enable;
CLK_CAS4_EN	"FALSE","TRUE"	"FALSE"	Output divider4 clock cascade enable;
CLKOUT5_SEL	0~4	0	Output clock CLKOUT5 selection setting;
CLKIN_BYPASS_EN	"FALSE","TRUE"	"FALSE"	Input clock bypass enable;
CLKOUT0_SYN_EN	"FALSE","TRUE"	"FALSE"	CLKOUT0_SYN port enable;
CLKOUT0_EXT_SYN_EN	"FALSE","TRUE"	"FALSE"	CLKOUT0_EXT_SYN port enable;
CLKOUT1_SYN_EN	"FALSE","TRUE"	"FALSE"	CLKOUT1_SYN port enable;
CLKOUT2_SYN_EN	"FALSE","TRUE"	"FALSE"	CLKOUT2_SYN port enable;
CLKOUT3_SYN_EN	"FALSE","TRUE"	"FALSE"	CLKOUT3_SYN port enable;
CLKOUT4_SYN_EN	"FALSE","TRUE"	"FALSE"	CLKOUT4_SYN port enable;
CLKOUT5_SYN_EN	"FALSE","TRUE"	"FALSE"	CLKOUT5_SYN port enable;

Parameter Name	Valid Values	Defaults	Description
INTERNAL_FB	"DISABLE","ENABLE"	"ENABLE"	Internal feedback divider selection setting;
EXTERNAL_FB	"CLKOUT0","CLKOUT1" "CLKOUT2","CLKOUT3" "CLKOUT4","DISABLE"	"DISABLE"	External feedback divider selection setting;
DYNAMIC_LOOP_EN	"FALSE","TRUE"	"FALSE"	PLL loop bandwidth parameter dynamic enable;
LOOP_MAPPING_EN	"FALSE","TRUE"	"FALSE"	PLL loop bandwidth parameter automatic mapping enable;
BANDWIDTH	"LOW", "HIGH" "OPTIMIZED"	"OPTIMIZED"	Bandwidth selection settings;

### 7.12.5 Instantiate Module

When the input is 50M, the output clock clkout0 50M, the output pll\_lock signal, and the internal feedback instantiation are as follows:

```
GTP_PLL_E3 #(
    .CLKIN_FREQ(50),
    .PFDEN_EN("FALSE"),
    .VCOCLK_DIV2(0),
    .DYNAMIC_RATIOI_EN("FALSE"),
    .DYNAMIC_RATIOM_EN("FALSE"),
    .DYNAMIC_RATIO0_EN("FALSE"),
    .DYNAMIC_RATIO1_EN("FALSE"),
    .DYNAMIC_RATIO2_EN("FALSE"),
    .DYNAMIC_RATIO3_EN("FALSE"),
    .DYNAMIC_RATIO4_EN("FALSE"),
    .DYNAMIC_RATIOF_EN("FALSE"),
    .STATIC_RATIOI(2),
    .STATIC_RATIOM(1),
    .STATIC_RATIO0(12),
    .STATIC_RATIO1(16),
    .STATIC_RATIO2(16),
    .STATIC_RATIO3(16),
    .STATIC_RATIO4(16),
    .STATIC_RATIOF(24),
    .DYNAMIC_DUTY0_EN("FALSE"),
    .DYNAMIC_DUTY1_EN("FALSE"),
)
```

.DYNAMIC\_DUTY2\_EN ("FALSE"),  
.DYNAMIC\_DUTY3\_EN ("FALSE"),  
.DYNAMIC\_DUTY4\_EN ("FALSE"),  
.STATIC\_DUTY0 (12),  
.STATIC\_DUTY1 (16),  
.STATIC\_DUTY2 (16),  
.STATIC\_DUTY3 (16),  
.STATIC\_DUTY4 (16),  
.STATIC\_PHASE0 (0),  
.STATIC\_PHASE1 (0),  
.STATIC\_PHASE2 (0),  
.STATIC\_PHASE3 (0),  
.STATIC\_PHASE4 (0),  
.STATIC\_PHASEF (0),  
.STATIC\_CPHASE0 (0),  
.STATIC\_CPHASE1 (0),  
.STATIC\_CPHASE2 (0),  
.STATIC\_CPHASE3 (0),  
.STATIC\_CPHASE4 (0),  
.STATIC\_CPHASEF (0),  
.CLK\_CAS1\_EN ("FALSE"),  
.CLK\_CAS2\_EN ("FALSE"),  
.CLK\_CAS3\_EN ("FALSE"),  
.CLK\_CAS4\_EN ("FALSE"),  
.CLKOUT5\_SEL (0),  
.CLKIN\_BYPASS\_EN ("FALSE"),  
.CLKOUT0\_SYN\_EN ("FALSE"),  
.CLKOUT0\_EXT\_SYN\_EN ("FALSE"),  
.CLKOUT1\_SYN\_EN ("FALSE"),  
.CLKOUT2\_SYN\_EN ("FALSE"),  
.CLKOUT3\_SYN\_EN ("FALSE"),  
.CLKOUT4\_SYN\_EN ("FALSE"),  
.CLKOUT5\_SYN\_EN ("FALSE"),  
.INTERNAL\_FB ("ENABLE"),

```
.EXTERNAL_FB ("DISABLE"),
.DYNAMIC_LOOP_EN ("FALSE"),
 LOOP_MAPPING_EN ("FALSE"),
.BANDWIDTH ("OPTIMIZED")

) u_pll_e3 (
.CLKOUT0 (clkout0),
.CLKOUT0_EXT (),
.CLKOUT1 (),
.CLKOUT2 (),
.CLKOUT3 (),
.CLKOUT4 (),
.CLKOUT5 (),
.CLKSWITCH_FLAG (),
.LOCK (pll_lock),
.CLKIN1 (clkin1),
.CLKIN2 (1'b0),
.CLKFB (),
.CLKIN_SEL (1'b0),
.CLKIN_SEL_EN (1'b0),
.PFDEN (),
.ICP_BASE (1'b0),
.ICP_SEL (4'b0),
.LPFRES_SEL (3'b0),
.CRIPPLE_SEL (1'b0),
.PHASE_SEL (3'b0),
.PHASE_DIR (1'b0),
.PHASE_STEP_N (1'b0),
.LOAD_PHASE (1'b0),
.RATIOM (7'b0),
.RATIOI (),
.RATIO0 (),
.RATIO1 (),
.RATIO2 (),
.RATIO3 (),
```

.RATIO4 (),  
 .RATIOF (),  
 .DUTY0 (),  
 .DUTY1 (),  
 .DUTY2 (),  
 .DUTY3 (),  
 .DUTY4 (),  
 .CLKOUT0\_SYN (),  
 .CLKOUT0\_EXT\_SYN (),  
 .CLKOUT1\_SYN (),  
 .CLKOUT2\_SYN (),  
 .CLKOUT3\_SYN (),  
 .CLKOUT4\_SYN (),  
 .CLKOUT5\_SYN (),  
 .PLL\_PWD (1'b0),  
 .RST (1'b0),  
 .RSTODIV (1'b0)

### 7.12.6 Detailed Functional Description

Refer to "***UG020004\_Logos Family FPGAs Clock Resources (Clock) User Guide***".

## 7.13 GTP\_DLL

### 7.13.1 Supported Devices

Table 7-37 Device Models That Support GTP\_DLL

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 7.13.2 Description of Functionality

GTP\_DLL is mainly used to dynamically lock the frequency of the input reference clock and output the number of delay steps equivalent to a quarter of the clock cycle. The delay step can be used to track changes in temperature and voltage in real time. The typical application of the delay step includes the following: Used by the DDC in IO LOGIC to complete a 1/4 cycle delay of the input

signal (such as input DQS) to correctly sample data; by the DDC in IO LOGIC to generate WCLK\_DEL, and by IOCLKDELY in the clock input pins to create clock delays.

### 7.13.3 Port Description

Table 7-38 GTP\_DLL Port Description

Port Signal	Input/Output	Description
CLKIN	Input	Clock input from pin
UPDATE_N	Input	Request to update DLL's delay step (active-low)
RST	Input	Active-low reset signal
PWD	Input	Active-high power down signal
DELAY_STEP	Output	Delay step of DLL output
LOCK	Output	LOCK flag signal, with active-high indicating locked

### 7.13.4 Paramater Description

Table 7-39 GTP\_DLL Parameter Description

Port Signal	Input/Output	Description
GRS_EN	Parameter	Global reset enable signal (internal chip) with default value of "TRUE"
FAST_LOCK	Parameter	DLL LOCK mode selection, with the default value of "TRUE"
DELAY_STEP_OFFSET	Parameter	Static fine-tuning of the DELAY_STEP of DLL output, with valid values of <-4, -3, -2, -1, 0, 1, 2, 3, 4> and a default value of <0>

### 7.13.5 Instantiate Module

```
GTP_DLL
#(
    .GRS_EN          ("TRUE"),
    .FAST_LOCK       ("TRUE"),
    .DELAY_STEP_OFFSET ( 0 ) // -4, -3, -2, -1, 0, 1, 2, 3, 4
) GTP_DLL_INST(
    .CLKIN          (CLKIN),
    .UPDATE_N        (UPDATE_N),
    .RST             (RST),
    .PWD             (PWD),
    .LOCK            (LOCK),
    .DELAY_STEP      (DELAY_STEP)
)
```

);

### 7.13.6 Detailed Description of Functionality

GTP\_DLL converts the relationship between the input clock CLKIN and the delay of the delay unit into a digital code, and proportionally controls the digital code based on phase difference requirements to generate the required code.

Different phase differences can be set through the parameter **DELAY\_STEP\_OFFSET**, as shown in the following table.

Table 7-40 List of Phase Shifts for Different Configurations

<b>DELAY_STEP_OFFSET</b>	<b>Phase Shift</b>
-4	45
-3	57
-2	68
-1	79
0	90
1	101
2	112
3	123
4	135

Digital codes vary with different input frequencies, with the relationship as shown in the following table.

Table 7-41 List of Digital Codes for Different Frequencies and Phase Shift Settings

<b>Typical Frequency Points (MHz)</b>	<b>DELAY_STEP</b>								
	<b>45 Degree</b>	<b>57 Degree</b>	<b>68 Degree</b>	<b>79 Degree</b>	<b>90 Degree</b>	<b>101 Degree</b>	<b>112 Degree</b>	<b>123 Degree</b>	<b>135 Degree</b>
100	8'h32	8'h3E	8'h4B	8'h57	8'h64	8'h70	8'h7D	8'h89	8'h96
133	8'h25	8'h2F	8'h38	8'h42	8'h4B	8'h54	8'h5E	8'h67	8'h71
150	8'h21	8'h29	8'h32	8'h3A	8'h42	8'h4B	8'h53	8'h5B	8'h64
200	8'h19	8'h1F	8'h25	8'h2C	8'h32	8'h38	8'h3E	8'h45	8'h4B
267	8'h12	8'h17	8'h1C	8'h21	8'h25	8'h2A	8'h2F	8'h33	8'h38
300	8'h10	8'h15	8'h19	8'h1D	8'h21	8'h25	8'h29	8'h2E	8'h32
400	8'h0C	8'h0F	8'h13	8'h16	8'h19	8'h1C	8'h1F	8'h22	8'h25
500	8'h0A	8'h0C	8'h0F	8'h11	8'h14	8'h16	8'h19	8'h1B	8'h1E
533	8'h0A	8'h0B	8'h0E	8'h10	8'b12	8'h15	8'h17	8'h1A	8'h1C
600	8'h09	8'h0A	8'h0C	8'h0E	8'b10	8'h13	8'h15	8'h17	8'h19
625	8'h08	8'h0A	8'h0C	8'h0E	8'h10	8'h12	8'h14	8'h16	8'h18

Typical Frequency Points (MHz)	DELAY_STEP								
	45 Degree	57 Degree	68 Degree	79 Degree	90 Degree	101 Degree	112 Degree	123 Degree	135 Degree
667	8'h07	8'h09	8'h0B	8'h0D	8'h0F	8'h11	8'h13	8'h14	8'h16

When the frequency of the input clock is 100MHz and the phase shift is 90, the GTP\_DLL waveform diagram is shown as follows.

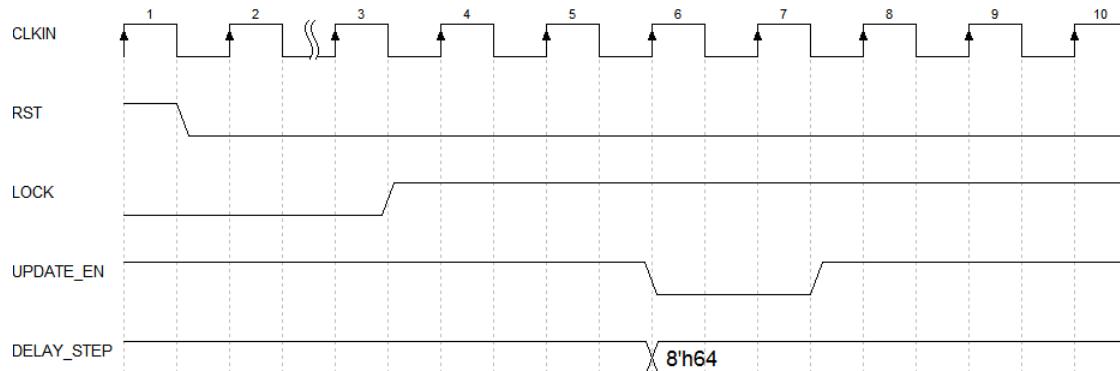


Figure 7-9 GTP\_DLL Waveform Diagram

## Chapter 8 Usage Instructions for Configuration-related GTPs

---

### 8.1 Usage Instructions for GTP\_EFUSECODE

#### 8.1.1 Supported Devices

Table 8-1 Device Models That Support GTP\_EFUSECODE

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 8.1.2 Description of Functionality

This GTP is used to read EFUSECODE.

#### 8.1.3 Port Description

Table 8-2 GTP\_EFUSECODE Port List

Port	Direction	Description
EFUSE_CODE[31:0]	Output	Efusecode output data bus

#### 8.1.4 Parameter Description

Table 8-3 GTP\_EFUSECODE Parameter List

Parameter Name	Parameter Type	Valid Values	Defaults	Description
SIM_EFUSE_VALUE	Binary	0~32'hffffffff ;	32'h12345678	Efusecode Value

#### 8.1.5 Instantiation template

```
GTP_EFUSECODE#(
    .SIM_EFUSE_VALUE  (32'h12345678)
  ) GTP_EFUSECODE_inst(
    .EFUSE_CODE      (efuse)
  );
```

### 8.1.6 Detailed Functional Description

Output 32-bit data stored in efuse to the user in parallel.

## 8.2 Usage Instructions for GTP\_IPAL\_E1

### 8.2.1 Supported Devices

Table 8-4 Device Models That Support GTP\_IPAL\_E1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 8.2.2 Description of Functionality

Configure or read back bitstreams through the IPAL interface.

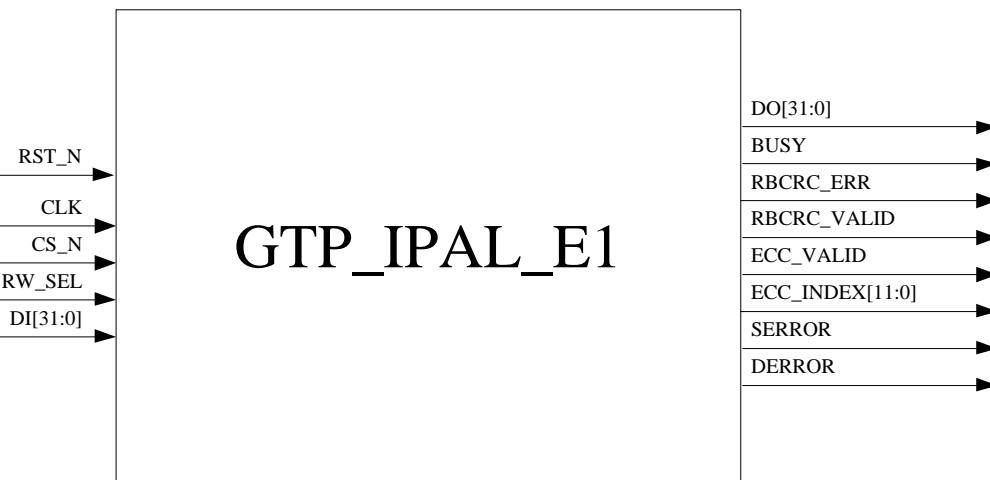


Figure 8-1 GTP\_IPAL\_E1 Structure Block Diagram

### 8.2.3 Port Description

Table 8-5 GTP\_IPAL\_E1 Port Description

Port	Direction	Description
RST_N	Input	Reset
CLK	Input	Clock
CS_N	Input	Chip Select enable signal
RW_SEL	Input	Read/write selection
BUSY	Output	Readback with busy internal data bus

Port	Direction	Description
DI	Input	Input data
DO	Output	Output data
RBCRC_ERR	Output	Readback CRC error
RBCRC_VALID	Output	Readback CRC valid
ECC_VALID	Output	ECC valid flag; pulse signal; lasts for one clock cycle
ECC_INDEX	Output	Address index of single bit error
SERROR	Output	SEU detection single-bit error flag
DERROR	Output	SEU detection double-bit error flag

#### 8.2.4 Paramater Description

Table 8-6 GTP\_IPAL\_E1 Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
IDCODE	binary	0~32'hfffffff	32'haaaa5555	Chip IDCODE
DATA_WIDTH	String	X8 X16 X32	"X8"	Bit width selection
SIM_DEVICE	String	"PGL25G", "PGL50G", "PGL50H", "PGL100H"	"PGL25G"	Device selection

Notes:

1. PGL25G, PGL20H, PGL50G, PGL50H, and PGL100H only support X8 and X16 and do not support X32.
2. Valid values of SIM\_DEVICE indicate that this GTP supports the bitstream structure of these devices and can be used for SEU IP. For applications other than SEU, any device from the PGL family, such as PGL12G, PGL22G, and PGL20H, can be used.

#### 8.2.5 Instantiation template

```

GTP_IPAL_E1 #(
    .DATA_WIDTH("X8"),
    .IDCODE('b101010101010101001010101010101),
    .SIM_DEVICE("PGL25G")
) GTP_IPAL_E1_inst (
    .DO(),           // OUTPUT[31:0]
    .ECC_INDEX(),   // OUTPUT[11:0]
    .DI(),           // INPUT[31:0]
    .BUSY(),          // OUTPUT
    .DERROR(),        // OUTPUT
    .ECC_VALID(),     // OUTPUT
    .RBCRC_ERR(),    // OUTPUT
    .RBCRC_VALID() // OUTPUT
)

```

```

.SERROR(),      // OUTPUT
.CLK(),         // INPUT
.CS_N(),        // INPUT
.RST_N(),        // INPUT
.RW_SEL()       // INPUT
);
    
```

### 8.2.6 Detailed Functional Description

Users can operate the internal slave parallel interface by instantiating this GTP, and can also conveniently perform Readback CRC and SEU operations using the Readback CRC and SEU interfaces. This part focuses on the introduction to the internal slave parallel interface (The RW\_SEL signal transition can only occur when CS\_N is high).

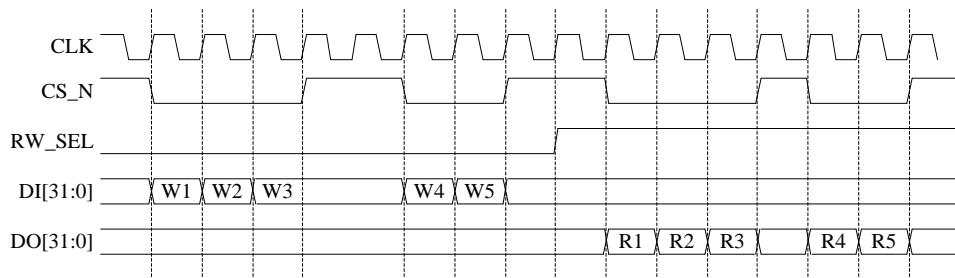


Figure 8-2 Read/Write Timing for GTP\_IPAL\_E1 Internal Slave Parallel Interface

## 8.3 Usage Instructions for GTP\_SCANCHAIN\_E1

### 8.3.1 Supported Devices

Table 8-7 Device Models That Support GTP\_SCANCHAIN\_E1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 8.3.2 Description of Functionality

Read the value of the user data register through the JTAG interface.

### 8.3.3 Port Description

Table 8-8 GTP\_SCANCHAIN\_E1 Port Description

Port	Direction	Description
TDI	Input	JTAG Interface
TDO	Output	JTAG Interface
TMS	Input	JTAG Interface
TCK	Input	JTAG Interface
RST	Output	Jtag soft reset output
CAPDR	Output	Jtag captureddr status indicator
SHFTDR	Output	Shiftdr status indicator
UPDR	Output	Updatedr status indicator
JCLK	Output	Shiftdr as the clock
FLG_USER	Output	User instruction indicator
TDI_USER	Output	Input data of user registers
TDO_USER	Input	Output data of user registers
JRTI	Output	Run/test idle status indicator
TCK_USER	Output	Tck to user
TMS_USER	Output	Tms to user

### 8.3.4 Paramater Description

Table 8-9 GTP\_SCANCHAIN\_E1 Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
IDCODE	Binary	0~32'hffffffff ;	32'haaaa5555	IDCODE
CHAIN_NUM	Integer	1, 2, 3, 4	1	User DR number

### 8.3.5 Instantiation template

```
GTP_SCANCHAIN
#(
    .IDCODE (32'haaaa5555),
    .CHAIN_NUM (1)
)
GTP_SCANCHAIN_inst(
    .TCK      (tck),
    .TDI      (tdi),
    .TMS      (tms),
```

```

.TDO      (tdo),
.CAPDR   (capture),
.JCLK     (jclk),
.RST      (rst),
.FLG_USER (flg_user),
.SHFTDR  (shift),
.TDI_USER (tdi_user),
.TMS_USER (tms_user),
.JRTI     (jrti),
.UPDR     (update),
.TDO_USER (tdo_user)
);

```

### 8.3.6 Detailed Functional Description

Users can use it to read the chip IDCODE, or read/write multiple user logic data register values separately through the JTAG interface.

## 8.4 Usage Instructions for GTP\_FLASHIF

### 8.4.1 Supported Devices

Table 8-10 Device Models That Support GTP\_FLASHIF

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Not supported	Not supported	Not supported

### 8.4.2 Description of Functionality

This module serves as a clock interface for the user and internal or external Flash.

### 8.4.3 Port Description

Table 8-11 GTP\_FLASHIF Port Description

Port	Direction	Description
EN_N	Input	Active-low signal for user internal FLASH flag
CLK	Input	User internal FLASH clock

Port	Direction	Description
CS_N	Input	Active-low signal for user internal FLASH Chip Select
DIN	Input	User internal FLASH data bus input
DOUT_EN_N	Input	Active-low enable signal for internal FLASH data bus output
DOUT	Output	User internal FLASH data bus output

#### 8.4.4 Instantiation template

```
GTP_FLASHIF GTP_FLASHIF_inst(
    .CLK      (clk),
    .EN_N     (en_n),
    .CS_N     (cs_n),
    .DIN      (din),
    .DOUT_EN_N (dout_en_n),
    .DOUT     (dout)
);
```

### 8.5 Usage Instructions for GTP\_UDID

#### 8.5.1 Supported Devices

Table 8-12 Device Models That Support GTP\_UDID

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 8.5.2 Description of Functionality

This GTP is used to read the UDID CODE value.

The UID is 64-bit long.

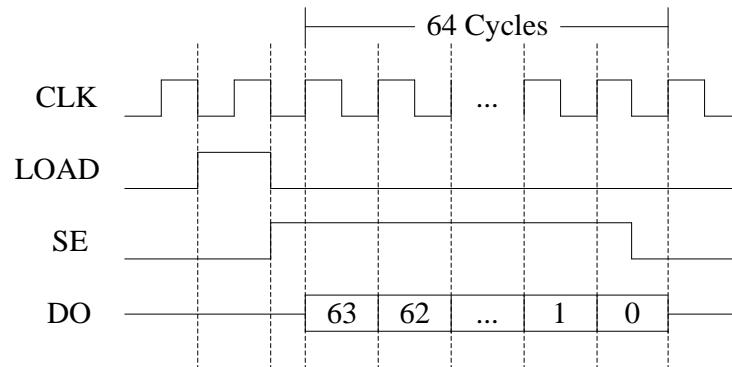


Figure 8-3 Read UID Timing

Users can extend the bit width of the UID.

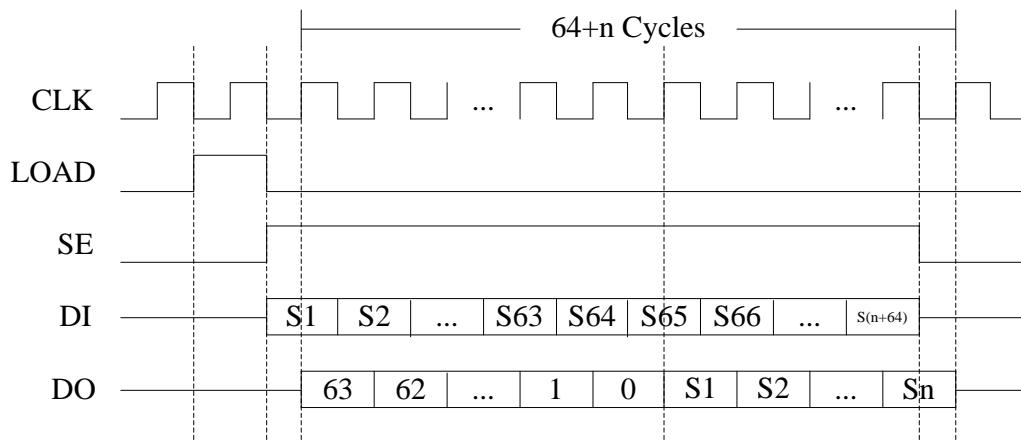


Figure 8-4 Extended UID Bit Width

### 8.5.3 Port Description

Table 8-13 GTP\_UDID Port Description

Port	Direction	Description
DI	Input	Serial data input
DO	Output	Serial data output
SE	Input	Enables data shift
LOAD	Input	Data registers parallel load UID CODE
CLK	Input	Clock

### 8.5.4 Parameter Description

Table 8-14 GTP\_UDID Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
UDID_CODE	Binary	0~64'hffffffff ffffffff;	0	Chip identifier
UDID_WIDTH	integer	0~32'hffff_ffff	64	Set the bit width of UDID_CODE

### 8.5.5 Instantiation template

### 8.5.6 Detailed Functional Description

Serially output 64-bit chip identifier data to the user.

## Chapter 9 Usage Instructions for DDR GTP

### 9.1 Usage Instructions for GTP\_DDC\_E1

#### 9.1.1 Supported Devices

Table 9-1 Device Models That Support GTP\_DDC\_E1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 9.1.2 Description of Functionality

To support DDR/DDR2/DDR3 memory interfaces, each IO Group contains dedicated circuits for GTP\_DDC\_E1, with the structure block diagram as follows:

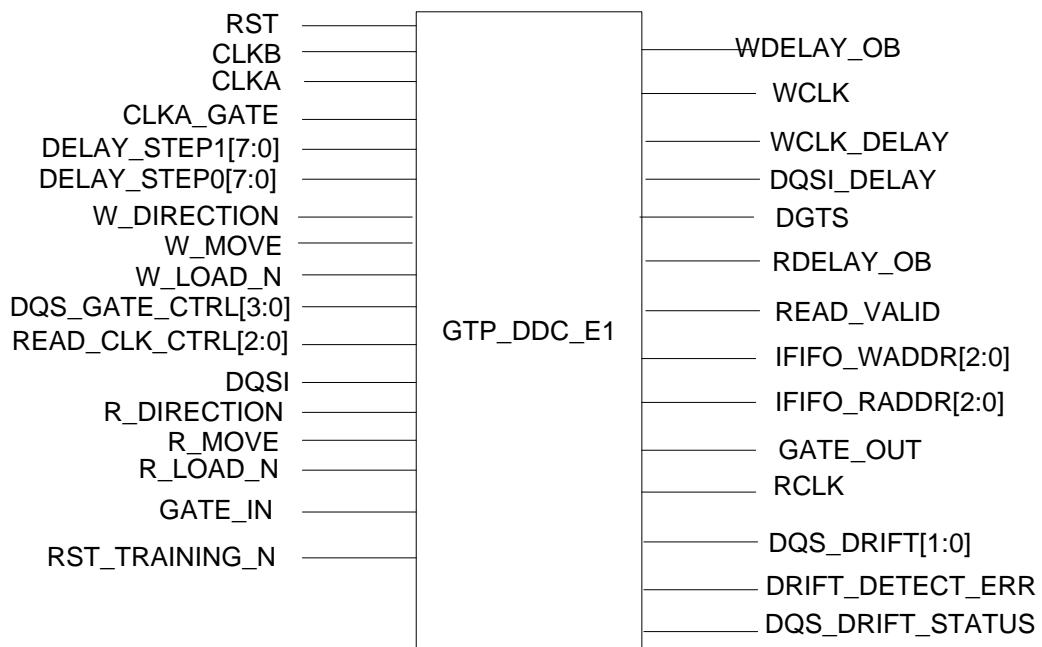


Figure 9-1 GTP\_DDC\_E1 Structure Block Diagram

#### 9.1.3 Port Description

Table 9-2 GTP\_DDC\_E1 Port Description

Port	Direction	Description
RST	Input	Local reset; from SRB. As the GTP does not include polarity selection, polarity control must be added to the GTP stimulus for verification against the grid device result if

Port	Direction	Description
		needed.
CLKB	Input	System clock, which can be a regional clock or a global clock
CLKA	Input	Dedicated IOCLK which connects with IOCLK0 and IOCLK1 in DQS_DDC implementation
CLKA_GATE	Input	Used for GATE internal clock
DELAY_STEP1	Input	Delay step from DLL
DELAY_STEP0	Input	Delay step for write leveling
W_DIRECTION	Input	Write clock delay control code for direction control adjustment
W_MOVE	Input	Write clock delay control code for trigger signal adjustment
W_LOAD_N	Input	Write clock delay control code for writing
DQS_GATE_CTRL	Input	Used to control the timing of DQS gate
READ_CLK_CTRL	Input	Read clock control signal
DQSI	Input	DQS input
R_DIRECTION	Input	Read clock delay control code for direction control adjustment
R_MOVE	Input	Read clock delay control code for trigger signal adjustment
R_LOAD_N	Input	Read clock delay control code for writing
GATE_IN	Input	External input gate window
RST_TRAINING_N	Input	Reset signal after training completion, used for reset control of DGTS, read/write pointer generation, and READ_VALID signal.
WDELAY_OB	Output	Overflow of write leveling delay
WCLK	Output	DQS Write clock
WCLK_DELAY	Output	Write clock delays 270 degrees
DQSI_DEL	Output	DQS read clock
DGTS	Output	DQS gate status indicator, DQS_GATE_TRAINING_STATUS
GATE_OUT	Output	Gate window output
RDELAY_OB	Output	Overflow of read clock
DQS_DRIFT[1:0]	Output	DQS delay variation detection signal, update logic records this signal and determines whether to execute the update adjustment of gate window position
DRIFT_DETECT_ERR	Output	Error flag for DQS delay variation detection; trigger PHY to perform update training operation
DQS_DRIFT_STATUS	Output	DQS delay overrange identification signal; a high level indicates excessive delay variation, which triggers the PHY to perform update

Port	Direction	Description
		training operation
READ_VALID	Output	Flag for correct data read
IFIFO_WADDR	Output	IFIFO write address
IFIFO_RADDR	Output	IFIFO read address
RCLK	Output	Serial clock provided for IOL OSERDES

### 9.1.4 Parameter Description

Table 9-3 GTP\_DDC\_E1 Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
GRS_EN	string	" TRUE " " FALSE "	" TRUE "	Global reset signal enable
DDC_MODE	string	" FULL_RATE" " HALF_RATE " " QUAD_RATE "	" FULL_RATE"	DDC working mode
IFIFO_GENERIC	string	" TRUE " " FALSE "	" FALSE "	IFIFO working mode selection
WCLK_DELAY_OFFSET	integer	0~511	0	WCLK_DEL offset: The most significant bit of this parameter is the sign bit, sign=0, positive adjustment sign=1, negative adjustment
DQSI_DELAY_OFFSET	integer	0~511	0	DQSI_DEL offset, with the same functions of WCLK_DELAY_OFFSET
CLKA_GATE_EN	string	" TRUE " " FALSE "	" FALSE "	GATE enable
R_DEALY_STEP_EN	string	" TRUE " " FALSE "	" TRUE "	DLL_STEP enable
R_MOVE_EN	string	" TRUE " " FALSE "	" FALSE "	Read delay control signal enable
W_MOVE_EN	string	" TRUE " " FALSE "	" FALSE "	Write leveling control signal enable
R_EXTEND	string	" TRUE " " FALSE "	" FALSE "	write leveling extension
GATE_SEL	string	" TRUE " " FALSE "	" FALSE "	Read clock gate window source selection
RADDR_INIT	string	0~7	0	RD_ADDR initial value
WCLK_DELAY_SEL	string	" TRUE " " FALSE "	" FALSE "	WCLK_DELAY source selection: "FALSE": a 270-degree phase shift of WCLK; "TRUE": a 90-degree phase shift of WCLK;
RCLK_SEL	string	" TRUE " " FALSE "	" FALSE "	RCLK source selection: "FALSE": CLKA/CLKB inverted; "TRUE"

Parameter Name	Parameter Type	Valid Values	Defaults	Description
				CLKA/CLKB;

### 9.1.5 Instantiation template

```
GTP_DDC_E1 #(
    .GRS_EN ("FALSE"),
    .DDC_MODE ("QUAD_RATE"),
    .IFIFO_GENERIC ("FALSE"),
    .WCLK_DELAY_OFFSET (0),
    .DQSI_DELAY_OFFSET (0),
    .CLKA_GATE_EN ("FALSE"),
    .R_DELAY_STEP_EN ("TRUE"),
    .R_MOVE_EN ("TRUE"),
    .W_MOVE_EN ("FALSE"),
    .R_EXTEND ("TRUE"),
    .GATE_SEL ("TRUE"),
    .RADDR_INIT (3'd0)
) u_GTP_DDC_CMD(
    .WDELAY_OB (),
    .WCLK (wclk_cmd),
    .WCLK_DELAY (),
    .RCLK (ioclk_cmd),
    .RDELAY_OB (),
    .DQSI_DELAY (),
    .DGTS (),
    .READ_VALID (),
    .GATE_OUT (),
    .IFIFO_WADDR (),
    .IFIFO_RADDR (),
    .DQS_DRIFT (),
    .DRIFT_DETECT_ERR (),
    .DQS_DRIFT_STATUS (),
    .RST (~reset_n_prim),
    .CLKB (rclk),//glck
);
```

```
.CLKA (ioclk[0]),  
.CLKA_GATE (1'b1),  
.DELAY_STEP1 (dll_step),  
.DELAY_STEP0 (8'd0),  
.W_DIRECTION (1'b0),  
.W_MOVE (1'b0),  
.W_LOAD_N (1'b0),  
.DQS_GATE_CTRL (4'd0),  
.READ_CLK_CTRL (3'd0),  
.DQSI (),  
.GATE_IN (),  
.R_DIRECTION (1'b0),  
.R_MOVE (1'b0),  
.R_LOAD_N (1'b0),  
.RST_TRAINING_N (1'b1)  
);
```

### 9.1.6 Detailed Functional Description

GTP\_DDC\_E1 mainly provides the following functions:

1. Generate the write clock for DDR memory;
2. Phase-shift the DQSI signal by 90 degrees during read operations to capture input data and generate the correct gate identification signal DGTS;
3. Generate a signal for core logic to determine whether the read operation is valid;
4. Generate read/write direction signals for input FIFO;
5. Support the write leveling for DDR3. Support clock shutdown operation in quad/half rate mode, with the shutdown lasting an integer multiple of four CLKA cycles

## 9.2 Usage Instructions for GTP\_DDRC

### 9.2.1 Supported Devices

Table 9-4 Device Models That Support GTP\_DDRC

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Not supported	Not supported	Not supported	Not supported

### 9.2.2 Description of Functionality

GTP\_DDRC mainly serves as a memory controller in the DDR system to generate the DFI interface signals required by DDR\_PHY. It controls DDR\_PHY to perform corresponding operations, thus controlling the working state of SDRAM and completing data read and write operations.

### 9.2.3 Port Description

Table 9-5 GTP\_DDRC Port Description

Port	Direction	Description
CORE_DDRC_CORE_CLK	Input	ddrc clock signal
CORE_DDRC_RST	Input	DDRC reset signal (active-low)
ARESET_0	Input	Asynchronous reset signal of the AXI interface (active-low)
ACLK_0	Input	Clock signal of the AXI interface
AWID_0[7:0]	Input	AXI write address ID; different addresses are flagged by group
AWADDR_0[31:0]	Input	AXI write address; the first transaction address in burst transfers
AWLEN_0[7:0]	Input	AXI write burst length; number of transactions in burst transfers
AWSIZE_0[2:0]	Input	AXI write burst size; size of each transaction in burst transfers
AWBURST_0[1:0]	Input	AXI write burst type
AWLOCK_0	Input	AXI lock write transaction operation that is related to atomic transfer
AWVALID_0	Input	AXI write address valid
AWREADY_0	Output	AXI write address ready, indicating that the slave can receive the write address
AWQOS_0[3:0]	Input	AXI port write priority: the higher the value, the higher the priority
AWURGENT_0	Input	AXI urgent write operation; when enabled, the port arbitration immediately enters the write operation
AWPOISON_0	Input	AXI invalid write transaction
WDATA_0[127:0]	Input	AXI write data
WSTRB_0[15:0]	Input	AXI write data strobe signal
WLAST_0	Input	Indicate the last write transaction in AXI burst transfers

<b>Port</b>	<b>Direction</b>	<b>Description</b>
WVALID_0	Input	AXI write data valid
WREADY_0	Output	Indicate that the slave can receive write data
BID_0[7:0]	Output	AXI write response ID signal
BRESP_0[1:0]	Output	AXI write response signal
BVALID_0	Output	AXI write response valid
BREADY_0	Input	Indicate that the master can receive the write response signal
ARID_0 [7:0]	Input	AXI read address ID; different address set flags
ARADDR_0[31:0]	Input	AXI read address; the first transaction address in burst transfers
ARLEN_0 [7:0]	Input	AXI read burst length; number of transactions in burst transfers
ARSIZE_0 [2:0]	Input	AXI read burst size; size of each transaction in burst transfers
ARBURST_0[1:0]	Input	AXI read burst type
ARLOCK_0	Input	AXI lock read transaction operation that is related to atomic transfer
ARVALID_0	Input	AXI read address valid
ARREADY_0	Output	AXI read address ready, indicating the slave can receive the read address
ARQOS_0[3:0]	Input	AXI port read priority; the higher the value, the higher the priority
ARPOISON_0	Input	AXI invalid read transaction
RID_0[7:0]	Output	AXI read data ID
RDATA_0[127:0]	Output	AXI read data
RRESP_0[1:0]	Output	AXI read response
RLAST_0	Output	Indicate the last read transaction in an AXI burst transfer
RVALID_0	Output	AXI read data valid
RREADY_0	Input	Indicate the master can receive read data and respond
ARURGENT_0	Input	AXI urgent read operation; when enabled, the port arbitration immediately enters read write operation
RAQ_PUSH_0	Output	Transaction written to the AXI read address FIFO
RAQ_SPLIT_0	Output	The starting part of the Wrap transaction written to the AXI read address FIFO
WAQ_PUSH_0	Output	Transaction written to the AXI write address FIFO
WAQ_SPLIT_0	Output	The starting part of the Wrap transaction written to the AXI write address FIFO
ARESET_1	Input	Asynchronous reset signal of the AXI interface (active-low)
ACLK_1	Input	Clock signal of the AXI interface
AWID_1[7:0]	Input	AXI write address ID; different addresses are flagged by group
AWADDR_1[31:0]	Input	AXI write address; the first transaction address in burst transfers
AWLEN_1[7:0]	Input	AXI write burst length; number of transactions in burst transfers
AWSIZE_1[2:0]	Input	AXI write burst size; size of each transaction in burst transfers
AWBURST_1[1:0]	Input	AXI write burst type

<b>Port</b>	<b>Direction</b>	<b>Description</b>
AWLOCK_1	Input	AXI lock write transaction operation that is related to atomic transfer
AWVALID_1	Input	AXI write address valid
AWREADY_1	Output	AXI write address ready, indicating that the slave can receive the write address
AWQOS_1[3:0]	Input	AXI port write priority: the higher the value, the higher the priority
AWURGENT_1	Input	AXI urgent write operation; when enabled, the port arbitration immediately enters the write operation
AWPOISON_1	Input	AXI invalid write transaction
WDATA_1[63:0]	Input	AXI write data
WSTRB_1[7:0]	Input	AXI write data strobe signal
WLAST_1	Input	Indicate the last write transaction in AXI burst transfers
WVALID_1	Input	AXI write data valid
WREADY_1	Output	Indicate that the slave can receive write data
BID_1[7:0]	Output	AXI write response ID signal
BRESP_1[1:0]	Output	AXI write response signal
BVALID_1	Output	AXI write response valid
BREADY_1	Input	Indicate that the master can receive the write response signal
ARID_1[7:0]	Input	AXI read address ID; different address set flags
ARADDR_1[31:0]	Input	AXI read address; the first transaction address in burst transfers
ARLEN_1 [7:0]	Input	AXI read burst length; number of transactions in burst transfers
ARSIZE_1[2:0]	Input	AXI read burst size; size of each transaction in burst transfers
ARBURST_1[1:0]	Input	AXI read burst type
ARLOCK_1	Input	AXI lock read transaction operation that is related to atomic transfer
ARVALID_1	Input	AXI read address valid
ARREADY_1	Output	AXI read address ready, indicating the slave can receive the read address
ARQOS_1[3:0]	Input	AXI port read priority; the higher the value, the higher the priority
ARPOISON_1	Input	AXI invalid read transaction
RID_1[7:0]	Output	AXI read data ID
RDATA_1[63:0]	Output	AXI read data
RRESP_1[1:0]	Output	AXI read response
RLAST_1	Output	Indicate the last read transaction in an AXI burst transfer
RVALID_1	Output	AXI read data valid
RREADY_1	Input	Indicate the master can receive read data and respond
ARURGENT_1	Input	AXI urgent read operation; when enabled, the port arbitration immediately enters read write operation

Port	Direction	Description
RAQ_PUSH_1	Output	Transaction written to the AXI read address FIFO
RAQ_SPLIT_1	Output	The starting part of the Wrap transaction written to the AXI read address FIFO
WAQ_PUSH_1	Output	Transaction written to the AXI write address FIFO
WAQ_SPLIT_1	Output	The starting part of the Wrap transaction written to the AXI write address FIFO
ARESET_2	Input	Asynchronous reset signal of the AXI interface (active-low)
ACLK_2	Input	Clock signal of the AXI interface
AWID_2[7:0]	Input	AXI write address ID; different addresses are flagged by group
AWADDR_2[31:0]	Input	AXI write address; the first transaction address in burst transfers
AWLEN_2[7:0]	Input	AXI write burst length; number of transactions in burst transfers
AWSIZE_2[2:0]	Input	AXI write burst size; size of each transaction in burst transfers
AWBURST_2 [1:0]	Input	AXI write burst type
AWLOCK_2	Input	AXI lock write transaction operation that is related to atomic transfer
AWVALID_2	Input	AXI write address valid
AWREADY_2	Output	AXI write address ready, indicating that the slave can receive the write address
AWQOS_2[3:0]	Input	AXI port write priority: the higher the value, the higher the priority.
AWURGENT_2	Input	AXI urgent write operation; when enabled, the port arbitration immediately enters the write operation
AWPOISON_2	Input	AXI invalid write transaction
WDATA_2[63:0]	Input	AXI write data
WSTRB_2[7:0]	Input	AXI write data strobe signal
WLAST_2	Input	Indicate the last write transaction in AXI burst transfers
WVALID_2	Input	AXI write data valid
WREADY_2	Output	Indicate that the slave can receive write data
BID_2 [7:0]	Output	AXI write response ID signal
BRESP_2 [1:0]	Output	AXI write response signal
BVALID_2	Output	AXI write response valid
BREADY_2	Input	Indicate that the master can receive the write response signal
ARID_2 [7:0]	Input	AXI read address ID; different address set flags
ARADDR_2 [31:0]	Input	AXI read address; the first transaction address in burst transfers
ARLEN_2 [7:0]	Input	AXI read burst length; number of transactions in burst transfers
ARSIZE_2 [2:0]	Input	AXI read burst size; size of each transaction in burst transfers
ARBURST_2 [1:0]	Input	AXI read burst type
ARLOCK_2	Input	AXI lock read transaction operation that is related to atomic transfer

<b>Port</b>	<b>Direction</b>	<b>Description</b>
ARVALID_2	Input	AXI read address valid
ARREADY_2	Output	AXI read address ready, indicating the slave can receive the read address
ARQOS_2[3:0]	Input	AXI port read priority: the higher the value, the higher the priority.
ARPOISON_2	Input	AXI invalid read transaction
RID_2[7:0]	Output	AXI read data ID
RDATA_2[63:0]	Output	AXI read data
RRESP_2 [1:0]	Output	AXI read response
RLAST_2	Output	Indicate the last read transaction in an AXI burst transfer
RVALID_2	Output	AXI read data valid
RREADY_2	Input	Indicate the master can receive read data and respond
ARURGENT_2	Input	AXI urgent read operation; when enabled, the port arbitration immediately enters read write operation
RAQ_PUSH_2	Output	Transaction written to the AXI read address FIFO
RAQ_SPLIT_2	Output	The starting part of the Wrap transaction written to the AXI read address FIFO
WAQ_PUSH_2	Output	Transaction written to the AXI write address FIFO
WAQ_SPLIT_2	Output	The starting part of the Wrap transaction written to the AXI write address FIFO
CSYSREQ_0	Input	AXI lo- power request signal
CSYSACK_0	Output	AXI low-power request response signal
CACTIVE_0	Output	Indicate that AXI enters Low-power state (active-low)
CSYSREQ_1	Input	AXI lo- power request signal
CSYSACK_1	Output	AXI low-power request response signal
CACTIVE_1	Output	Indicate that AXI enters Low-power state (active-low)
CSYSREQ_2	Input	AXI lo- power request signal
CSYSACK_2	Output	AXI low-power request response signal
CACTIVE_2	Output	Indicate that AXI enters Low-power state (active-low)
CSYSREQ_DDRC	Input	DDRC low-power request signal
CSYSACK_DDRC	Output	DDRC low-power request response signal
CACTIVE_DDRC	Output	Indicate that it is already in DDRC low-power state (active-low)
PA_RMASK [2:0]	Input	Mask the read address request to PA from the corresponding port. [2] corresponds to port2
PA_WMASK [2:0]	Input	Mask the write address request to PA from the corresponding port. [2] corresponds to port2
DFI_ADDRESS[31:0]	Output	Send PHY address signal
DFI_BANK[5:0]	Output	BANK address sent to PHY by DDRC
DFI_CAS_N[1:0]	Output	CAS signal (active-low)
DFI_RAS_N[1:0]	Output	RAS signal (active-low)
DFI_WE_N[1:0]	Output	Write enable signal (active-low)

<b>Port</b>	<b>Direction</b>	<b>Description</b>
DFI_CKE[1:0]	Output	Clock enable signal
DFI_CS[1:0]	Output	Chip select signal, active-low
DFI_ODT[1:0]	Output	On-chip calibration signal
DFI_RESET_N[1:0]	Output	Reset memory signal sent to PHY (active-low), only supporting DDR3
DFI_WRDATA[63:0]	Output	DFI write data signal
DFI_WRDATA_MASK[7:0]	Output	DFI write data bit mask signal
DFI_WRDATA_EN[3:0]	Output	DFI write data enable
DFI_RDDATA[63:0]	Input	DFI read data signal
DFI_RDDATA_EN[3:0]	Output	DFI read data enable
DFI_RDDATA_VALID[3:0]	Input	DFI read data valid signal
DFI_CTRLUPD_ACK	Input	DDRC update request response signal
DFI_CTRLUPD_REQ	Output	DDRC update request signal
DFI_PHYUPD_REQ	Input	PHY update request signal
DFI_PHYUPD_TYPE[1:0]	Input	PHY update type selection signal
DFI_PHYUPD_ACK	Output	PHY update request response signal
DFI_LP_REQ	Output	DFI low-power request signal
DFI_LP_WAKEUP[3:0]	Output	DFI low-power wake-up time signal
DFI_LP_ACK	Input	DFI low-power request response signal
DFI_DRAM_CLK_DISABLE	Output	Turn off SDRAM clock signal
DFI_INIT_START	Output	PHY initialization start signal
DFI_INIT_COMPLETE	Input	PHY initialization complete signal
DFI_FREQUENCY[4:0]	Output	Signal indicating system operation frequency
PCLK	Input	APB clock signal
PRESET	Input	APB reset signal
PADDR[11:0]	Input	APB address signal
PWDATA[31:0]	Input	APB write data signal
PWRITE	Input	APB read/write direction signal
PSEL	Input	APB selection signal
PENABLE	Input	APB enable signal
PREADY	Output	APB ready signal
PRDATA[31:0]	Output	APB read data signal
PSLVERR	Output	APB error signal

#### 9.2.4 Instantiation template

```
GTP_DDRC u_ddrc(
    .CORE_DDRC_CORE_CLK (core_clk),
    .CORE_DDRC_RST (ddrc_ddrc_rst),
```

.ARESET\_0 (ddrc\_axi\_reset0),  
.ACLK\_0 (aclk\_0),  
.AWID\_0 (awid\_0),  
.AWADDR\_0 (awaddr\_0),  
.AWLEN\_0 (awlen\_0),  
.AWSIZE\_0 (awslice\_0),  
.AWBURST\_0 (awburst\_0),  
.AWLOCK\_0 (awlock\_0),  
.AWVALID\_0 (awvalid\_0),  
.AWREADY\_0 (awready\_0),  
.AWURGENT\_0 (awurgent\_0),  
.AWPOISON\_0 (awpoison\_0),  
.WDATA\_0 (wdata\_0),  
.WSTRB\_0 (wstrb\_0),  
.WLAST\_0 (wlast\_0),  
.WVALID\_0 (wvalid\_0),  
.WREADY\_0 (wready\_0),  
.BID\_0 (bid\_0),  
.BRESP\_0 (bresp\_0),  
.BVALID\_0 (bvalid\_0),  
.BREADY\_0 (bready\_0),  
.ARID\_0 (arid\_0),  
.ARADDR\_0 (araddr\_0),  
.ARLEN\_0 (arlen\_0),  
.ARSIZE\_0 (arsize\_0),  
.ARBURST\_0 (arburst\_0),  
.ARLOCK\_0 (arlock\_0),  
.ARVALID\_0 (arvalid\_0),  
.ARREADY\_0 (arready\_0),  
.ARPOISON\_0 (arpoison\_0),  
.RID\_0 (rid\_0),  
.RDATA\_0 (rdata\_0),  
.RRESP\_0 (rresp\_0),  
.RLAST\_0 (rlast\_0),

.RVALID\_0 (rvalid\_0),  
.RREADY\_0 (rready\_0),  
.ARURGENT\_0 (arurgent\_0),  
.RAQ\_PUSH\_0 (raq\_push\_0),  
.RAQ\_SPLIT\_0 (raq\_split\_0),  
.WAQ\_PUSH\_0 (waq\_push\_0),  
.WAQ\_SPLIT\_0 (waq\_split\_0),  
.ARESET\_1 (ddrc\_axi\_reset1),  
.ACLK\_1 (aclk\_1),  
.AWID\_1 (awid\_1),  
.AWADDR\_1 (awaddr\_1),  
.AWLEN\_1 (awlen\_1),  
.AWSIZE\_1 (awscale\_1),  
.AWBURST\_1 (awburst\_1),  
.AWLOCK\_1 (awlock\_1),  
.AWVALID\_1 (awvalid\_1),  
.AWREADY\_1 (awready\_1),  
.AWURGENT\_1 (awurgent\_1),  
.AWPOISON\_1 (awpoison\_1),  
.WDATA\_1 (wdata\_1),  
.WSTRB\_1 (wstrb\_1),  
.WLAST\_1 (wlast\_1),  
.WVALID\_1 (wvalid\_1),  
.WREADY\_1 (wready\_1),  
.BID\_1 (bid\_1),  
.BRESP\_1 (bresp\_1),  
.BVALID\_1 (bvalid\_1),  
.BREADY\_1 (bready\_1),  
.ARID\_1 (arid\_1),  
.ARADDR\_1 (araddr\_1),  
.ARLEN\_1 (arlen\_1),  
.ARSIZE\_1 (arsize\_1),  
.ARBURST\_1 (arburst\_1),  
.ARLOCK\_1 (arlock\_1),

.ARVALID\_1 (arvalid\_1),  
.ARREADY\_1 (arready\_1),  
.ARPOISON\_1 (arpoison\_1),  
.RID\_1 (rid\_1),  
.RDATA\_1 (rdata\_1),  
.RRESP\_1 (rresp\_1),  
.RLAST\_1 (rlast\_1),  
.RVALID\_1 (rvalid\_1),  
.RREADY\_1 (rready\_1),  
.ARURGENT\_1 (arurgent\_1),  
.RAQ\_PUSH\_1 (raq\_push\_1),  
.RAQ\_SPLIT\_1 (raq\_split\_1),  
.WAQ\_PUSH\_1 (waq\_push\_1),  
.WAQ\_SPLIT\_1 (waq\_split\_1),  
.ARESET\_2 (ddrc\_axi\_reset2),  
.ACLK\_2 (aclk\_2),  
.AWID\_2 (awid\_2),  
.AWADDR\_2 (awaddr\_2),  
.AWLEN\_2 (awlen\_2),  
.AWSIZE\_2 (awszie\_2),  
.AWBURST\_2 (awburst\_2),  
.AWLOCK\_2 (awlock\_2),  
.AWVALID\_2 (awvalid\_2),  
.AWREADY\_2 (awready\_2),  
.AWURGENT\_2 (awurgent\_2),  
.AWPOISON\_2 (awpoison\_2),  
.WDATA\_2 (wdata\_2),  
.WSTRB\_2 (wstrb\_2),  
.WLAST\_2 (wlast\_2),  
.WVALID\_2 (wvalid\_2),  
.WREADY\_2 (wready\_2),  
.BID\_2 (bid\_2),  
.BRESP\_2 (bresp\_2),  
.BVALID\_2 (bvalid\_2),

.BREADY\_2 (bready\_2),  
.ARID\_2 (arid\_2),  
.ARADDR\_2 (araddr\_2),  
.ARLEN\_2 (arlen\_2),  
.ARSIZE\_2 (arsize\_2),  
.ARBURST\_2 (arburst\_2),  
.ARLOCK\_2 (arlock\_2),  
.ARVALID\_2 (arvalid\_2),  
.ARREADY\_2 (arready\_2),  
.ARPOISON\_2 (arpoison\_2),  
.RID\_2 (rid\_2),  
.RDATA\_2 (rdata\_2),  
.RRESP\_2 (rresp\_2),  
.RLAST\_2 (rlast\_2),  
.RVALID\_2 (rvalid\_2),  
.RREADY\_2 (rready\_2),  
.ARURGENT\_2 (arurgent\_2),  
.RAQ\_PUSH\_2 (raq\_push\_2),  
.RAQ\_SPLIT\_2 (raq\_split\_2),  
.WAQ\_PUSH\_2 (waq\_push\_2),  
.WAQ\_SPLIT\_2 (waq\_split\_2),  
.AWQOS\_0 (awqos\_0),  
.ARQOS\_0 (arqos\_0),  
.AWQOS\_1 (awqos\_1),  
.ARQOS\_1 (arqos\_1),  
.AWQOS\_2 (awqos\_2),  
.ARQOS\_2 (arqos\_2),  
.CSYSREQ\_0 (csysreq\_0),  
.CSYSACK\_0 (csysack\_0),  
.CACTIVE\_0 (cactive\_0),  
.CSYSREQ\_1 (csysreq\_1),  
.CSYSACK\_1 (csysack\_1),  
.CACTIVE\_1 (cactive\_1),  
.CSYSREQ\_2 (csysreq\_2),

.CSYSACK\_2 (csysack\_2),  
.CACTIVE\_2 (cactive\_2),  
.CSYSREQ\_DDRC (csysreq\_ddrc),  
.CSYSACK\_DDRC (csysack\_ddrc),  
.CACTIVE\_DDRC (cactive\_ddrc),  
.PA\_RMASK (pa\_rmask),  
.PA\_WMASK (pa\_wmask),  
.DFI\_ADDRESS (dfi\_address),  
.DFI\_BANK (dfi\_bank),  
.DFI\_CAS\_N (dfi\_cas\_n),  
.DFI\_RAS\_N (dfi\_ras\_n),  
.DFI\_WE\_N (dfi\_we\_n),  
.DFI\_CKE (dfi\_cke),  
.DFI\_CS (dfi\_cs),  
.DFI\_ODT (dfi\_odt),  
.DFI\_RESET\_N (dfi\_reset\_n),  
.DFI\_WRDATA (dfi\_wrdata),  
.DFI\_WRDATA\_MASK (dfi\_wrdata\_mask),  
.DFI\_WRDATA\_EN (dfi\_wrdata\_en),  
.DFI\_RDDATA (dfi\_rddata),  
.DFI\_RDDATA\_EN (dfi\_rddata\_en),  
.DFI\_RDDATA\_VALID (dfi\_rddata\_valid),  
.DFI\_CTRLUPD\_ACK (dfi\_ctrlupd\_ack),  
.DFI\_CTRLUPD\_REQ (dfi\_ctrlupd\_req),  
.DFI\_DRAM\_CLK\_DISABLE (dfi\_dram\_clk\_disable),  
.DFI\_INIT\_COMPLETE (dfi\_init\_complete),  
.DFI\_INIT\_START (dfi\_init\_start),  
.DFI\_FREQUENCY (dfi\_frequency),  
.DFI\_PHYUPD\_REQ (dfi\_phyupd\_req),  
.DFI\_PHYUPD\_TYPE (dfi\_phyupd\_type),  
.DFI\_PHYUPD\_ACK (dfi\_phyupd\_ack),  
.DFI\_LP\_REQ (dfi\_lp\_req),  
.DFI\_LP\_WAKEUP (dfi\_lp\_wakeup),  
.DFI\_LP\_ACK (dfi\_lp\_ack),

.PCLK (pclk),  
 .PRESET (ddrc\_preset),  
 .PADDR (ddrc\_paddr),  
 .PWDATA (ddrc\_pwdata),  
 .PWRITE (ddrc\_pwrite),  
 .PSEL (ddrc\_psel),  
 .PENABLE (ddrc\_penable),  
 .PREADY (ready),  
 .PRDATA (prdata),  
 .PSLVERR (pslverr)

### 9.2.5 Detailed Functional Description

DDRC converts commands and data on the system bus (AXI4) into commands and data on the DFI interface that comply with the DDR protocol.

### 9.2.6 How to Use GTP

1. AXI DFI related interface signals must comply with AXI4 and DFI3.1 protocol requirements
2. DDRC and DDRPHY must be used together

## 9.3 Usage Instructions for GTP\_DDRPHY

### 9.3.1 Supported Devices

Table 9-6 Device Models That Support GTP\_DDRPHY

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Not supported	Not supported	Not supported	Not supported

### 9.3.2 Description of Functionality

PGL DDRPHY and PGL DDRC work together to perform DDR calibration and read-write operations, with the following main features:

- Fully compatible with the DFI interface of DDRC (synopsys umctl2);
- Support SDRAM types: LPDDR, DDR2, DDR3, and DDR3L;
- Support DDR frequency up to 1066Mbps;

- Support the working mode with the frequency ratio of DDRC and DDRPHY at 1:2;
- Support the data width of 16bit, and even 8bit after appropriate configuration. Support the burst length of BL8 and BL4;
- Only support single rank and not support ECC;
- Provide a debug interface for monitoring and controlling the DDR calibration process; provide an APB interface for configuring PHY registers.

### 9.3.3 Port Description

Table 9-7 GTP\_DDRPHY Port Description

<b>Port</b>	<b>Direction</b>	<b>Description</b>
DFI_RDDATA[63:0]	Output	DFI read data signal
DFI_RDDATA_VALID[3:0]	Output	DFI read data valid signal
DFI_CTRLUPD_ACK	Output	DDRC update request response signal
DFI_PHYUPD_REQ	Output	PHY update request signal
DFI_PHYUPD_TYPE[1:0]	Output	PHY update type selection signal
DFI_INIT_COMPLETE	Output	PHY initialization complete signal
DFI_LP_ACK	Output	DFI low-power request response signal
DFI_ERROR	Output	Indicates an error occurs inside PHY
DFI_ERROR_INFO[2:0]	Output	Indicate the error message content generated by PHY
DFI_ADDRESS[31:0]	Input	DFI address signal sent to PHY
DFI_BANK[5:0]	Input	BANK address sent to PHY
DFI_CAS_N[1:0]	Input	CAS signal sent to PHY (active-low)
DFI_CKE[1:0]	Input	Clock enable signal sent to PHY
DFI_CS[1:0]	Input	Chip Select signal sent to PHY (active-low)
DFI_ODT[1:0]	Input	On-chip calibration signal sent to PHY
DFI_RAS_N[1:0]	Input	RAS signal sent to PHY (active-low)
DFI_WE_N[1:0]	Input	Write enable signal sent to PHY (active-low)
DFI_WRDATA[63:0]	Input	DFI write data signal sent to PHY
DFI_WRDATA_EN[3:0]	Input	DFI write data enable signal sent to PHY
DFI_WRDATA_MASK[7:0]	Input	DFI write data bit mask signal sent to PHY
DFI_RDDATA_EN[3:0]	Input	DFI read data enable signal sent to PHY
DFI_RESET_N[1:0]	Input	Reset memory signal sent to PHY (active-low), only supporting DDR3
DFI_CTRLUPD_REQ	Input	Update request signal
DFI_PHYUPD_ACK	Input	PHY update request response signal send to PHY
DFI_DRAM_CLK_DISABLE	Input	Turn off SDRAM clock signal sent to PHY
DFI_INIT_START	Input	Initialization start signal sent to PHY
DFI_FREQUENCY[4:0]	Input	Signal sent to PHY indicating system operation

Port	Direction	Description
		frequency
DFI_LP_REQ	Input	DFI low-power request signal sent to PHY
DFI_LP_WAKEUP[3:0]	Input	DFI low-power wake-up time sent to PHY
PCLK	Input	APB clock signal
PRESET	Input	APB reset signal
PADDR[11:0]	Input	APB address signal
PWDATA[31:0]	Input	APB write data signal
PWRITE	Input	APB read/write direction signal
PSEL	Input	APB peripheral selection signal
PENABLE	Input	APB enable signal
PREADY	Output	APB ready signal
PRDATA[31:0]	Output	APB read data signal
DDRPHY_CLKIN	Input	PHY clock signal
DDYPHY_RST	Input	PHY logical reset
DDRPHY_RST_REQ	Output	DDR PHY request DDR interface reset
DDRPHY_RST_ACK	Input	DDR interface reset completion flag from reset logic (Soft)
DDRPHY_UPDATE	Input	PHY Update
DDRPHY_UPDATE_DONE	Output	PHY Update completed.
DDRPHY_WL_STEP_L[7:0]	Output	Code output to DQS_DDC for write leveling training (for lower 8bit data)
DDRPHY_WL_CTRL_L[2:0]	Output	Control fine-tuning of WCLK_DEL clock delay for DQS_DDC (for lower 8bit data)
DDRPHY_RDQS_STEP_L[2:0]	Output	Control fine-tuning of CLK_DQSI_DEL clock delay (for lower 8bit data)
DDRPHY_DQS_GATE_CTRL_L[1:0]	Output	Signal for gate operation on CLK_DQSI (for lower 8bit data)
DDRPHY_READ_CLK_CTRL_L[2:0]	Output	[2]: Gate source selection signal; [1:0]: Gate training clock source selection signal (for lower 8bit data)
DDRPHY_WL_OV_L	Input	Margin test output identifier for CLK_W_DEL clock delay fine-tuning (for lower 8bit data)
DDRPHY_DGTS_L	Input	Gate window position identifier (for lower 8bit data)
DDRPHY_READ_VALID_L	Input	Read valid signal from DQS_DDC
DDRPHY_DLL_STEP[7:0]	Input	DLL adjustment control code
DDRPHY_RDEL_OV_L	Input	Margin test output identifier for CLK_W_DEL clock delay fine-tuning (for 8bit lower) "1" indicates overflow; "0" indicates no overflow
DDRPHY_WEN_L[15:0]	Output	Control corresponding IOL to output state
DDRPHY_RDATA_L[31:0]	Input	8bit lower read data from IOL

Port	Direction	Description
DDRPHY_WDATA_L[31:0]	Output	8bit lower write data to IOL
DDRPHY_WDQS_EN_L[1:0]	Output	Control corresponding IOL to output state
DDRPHY_WDQS_L[3:0]	Output	Data Strobe for write data. Centered in write data. For the x16, this corresponds to the data on DQ0-DQ7
DDRPHY_DM_L[3:0]	Output	Input Data Mask: DM is an input mask signal for write data. Input data is masked when DM is sampled HIGH coincident with that input data during a Write access
DDRPHY_WL_STEP_H[7:0]	Output	Code output to DQS_DDC for write leveling training (for upper 8bit data)
DDRPHY_WL_CTRL_H[2:0]	Output	Control fine-tuning of WCLK_DEL clock delay for DQS_DDC (for upper 8bit data)
DDRPHY_RDQS_STEP_H[2:0]	Output	Control fine-tuning of CLK_DQSI_DEL clock delay (for upper 8bit data)
DDRPHY_DQS_GATE_CTRL_H[1:0]	Output	Signal for gate operation on CLK_DQSI (for upper 8bit data)
DDRPHY_READ_CLK_CTRL_H[2:0]	Output	[2]: Gate source selection signal; [1:0]: Gate training clock source selection signal (for upper 8bit data)
DDRPHY_RDEL_OV_H	Input	Margin test output identifier for CLK_W_DEL clock delay fine-tuning (for upper 8bit) "1" indicates overflow; "0" indicates no overflow
DDRPHY_WEN_H[15:0]	Input	Control upper 8bit daqIOL to output state
DDRPHY_WL_OV_H	Input	Margin test output identifier for CLK_W_DEL clock delay fine-tuning (for upper 8bit data)
DDRPHY_DGTS_H	Input	Gate window position identifier (for upper 8bit data)
DDRPHY_READ_VALID_H	Input	Read valid signal from DQS_DDC
DDRPHY_RDATA_H[31:0]	Input	Upper 8bit read data from IOL
DDRPHY_WDATA_H[31:0]	Output	Upper 8bit write data to IOL
DDRPHY_DM_H[3:0]	Output	Input Data Mask: DM is an input mask signal for write data. Input data is masked when DM is sampled HIGH coincident with that input data during a Write access.
DDRPHY_WDQS_H[3:0]	Output	Data Strobe for write data. Centered in write data. For the x16, this corresponds to the data on DQ8-DQ15
DDRPHY_WDQS_EN_H[1:0]	Output	Control corresponding IOL to output state
DDRPHY_CA_EN[55:0]	Output	Control command-related IOLs and set them to output state, where two signals controlling the output state of ddrphy_dm_l and ddrphy_dm_h come from ddrphy_wen_1 and ddrphy_wen_h; the remaining 26 IOLs (excluding ddrphy_mem_reset) for transmitting CA commands are tied to 0.
DDRPHY_ADDR[63:0]	Output	PHY-output-to-DRAM address signal

Port	Direction	Description
DDRPHY_BA[11:0]	Output	SDRAM Bank selection address
DDRPHY_CK[3:0]	Output	Clock, All address and control input signals are sampled on the the positive edge of CK
DDRPHY_CKE[3:0]	Output	Clock Enable: CKE HIGH activates, and CKE Low deactivates
DDRPHY_RAS_N[3:0]	Output	RAS command
DDRPHY_CS_N[3:0]	Output	Chip Select: All commands are masked when CS# is registered HIGH
DDRPHY_CAS_N[3:0]	Output	CAS command
DDRPHY_WE_N[3:0]	Output	WE command
DDRPHY_ODT[3:0]	Output	On Die Termination: ODT (registered HIGH) enables termination resistance internal to the DDR3 SDRAM
DDRPHY_MEM_RST	Output	Active Low Asynchronous Reset
DDRPHY_GATEI_H	Output	CLK_IO input is shut down; when pulled high, clk_io_out will be pulled low to zero after three CLK_IO falling edges. In QUAD/HALF RATE mode, the write clock CLK_W is pulled low to zero and the CLK_W_DEL is pulled high for the upper 8bit DQ
DDRPHY_GATEI_L	Output	CLK_IO input is shut down; when pulled high, clk_io_out will be pulled low to zero after three CLK_IO falling edges. In QUAD/HALF RATE mode, the write clock CLK_W is pulled low to zero and the CLK_W_DEL is pulled low for the lower 8bit DQ
DDRPHY_DQ_L[7:0]	Input	An input directly-driven by IOL; When it corresponds to the 8 IOLs transmitting DQ[7:0], the IOL does not perform gearing operations but is used for read training.
DDRPHY_DQ_H[7:0]	Input	An input directly-driven by IOL; When it corresponds to the 8 IOLs transmitting DQ[15:8], the IOL does not perform gearing operations but is used for read training.
DLL_UPDATE_ACK	Input	Dll_Update response signal
DLL_UPDATE_REQ	Output	Dll_Update request signal
DDRPHY_UPDATE_TYPE[1:0]	Input	User update request type from soft logic
DDRPHY_UPDATE_COMP_VAL_L[1:0]	Input	The position value in units of 1/4 cycle for the lower 8bit gate movement during the update operation
DDRPHY_UPDATE_COMP_DIR_L	Input	Direction indication for the lower 8bit gate movement 1'b0: gate moves to the left; 1'b1: gate moves to the right
DDRPHY_UPDATE_COMP_VAL_H[1:0]	Input	The position value in units of 1/4 cycle for the upper 8bit gate movement during the update operation
DDRPHY_UPDATE_COMP_DIR_H	Input	Direction indication for the upper 8bit gate movement 1'b0: gate moves to the left; 1'b1: gate moves to the right
IOL_CE[59:0]	Output	60 IOL clock registers are enabled in

Port	Direction	Description
		input/output mode and internally tied to 1
IOL_CLK_SYS[59:0]	Output	60 IOL system clocks, of which 47 used IOLs are the same as core_ddrc_core_clk while the remaining 13 unused IOL clocks are tied to 1
IOL_LRS[59:0]	Output	47 used IOL input/output registers are locally reset/set 13 unused IOLs are tied to 1
RST_DLL	Output	DLL local reset control signal; polarity selectable
UPDATE_N	Output	DLL output code update signal (active-low); needs to be set high for DDR read and write operations. The minimum signal level width is required to be twice the cycle of two DLL input reference clocks. "1": hold "0": update
DLL_CLK_INPUT	Output	Input reference clock; the clock source is SRB's regional clock or the IO clock tree. This signal is used by DLL only in 1:1 mode. As DLL is now in 1:2 mode, the signal is tied to 1 (either tied to 0 or tied to 1)
DLL_FREEZE	Output	DLL freeze control "1": freeze, the output signals CTRL_CODE, CTRL_CODE_T, LOCK, DIV_CLKIN_T remain unchanged; In Fast_lock mode, DIV_OSCCLK_T is pulled low; in non-fast_lock mode, DIV_OSCCLK_T remains unchanged; "0": release, the circuit works normally
DQS_RST[4:0]	Output	DQS local reset control signal, with polarity selectable
DQS_RST_TRAINING_N[4:0]	Output	The reset signal, which is active-low, is used to reset the read/write pointer generation and read pointer control logic of DQSL upon completion of DQS gate training
DQS_CLK_REGIONAL [4:0]	Output	Regional clock from HMEMC
DQS_GATEI[2:0]	Output	Input CLK_IO is shut down; when pulled high, clk_io_out is pulled low to zero after three CLK_IO falling edges. In QUAD/HALF RATE mode, the write clock CLK_W is pulled low to zero, and the CLK_W_DEL is pulled high
DQS_WL_STEP[23:0]	Output	Code for write leveling training The three dqs_ddc use for CA do not perform write leveling
DQS_WL_CTRL[8:0]	Output	Fine tuning of CLK_W_DEL clock delay [0]: Load input code (DLL+Adjustment settings) [1]: Use the falling edge of this signal and the code bit direction signal to change the code word [2]: Code bit direction control, with "0" for increasing the code word, "1" for decreasing the code word
DQS_DQS_GATE_CTRL[11:0]	Output	The signal for gating CLK_DQSI, which is internally tied to 0

Port	Direction	Description
DQS_DQS_GATE_CTRL_TF2[3:0]	Output	The upper 2 bits of the two corresponding DQS_DDC signals that control DQ are unused and need to be tied to 1
DQS_READ_CLK_CTRL[8:0]	Output	Internally tied to 0, controlling three dqs_ddc for CMD
DQS_RDEL_CTRL[8:0]	Output	Internally tied to 1, controlling three dqs_ddc for CMD
IOL_TX_DATA_TF8[103:0]	Output	The 13 unused IOL TX_DATA are tied to 1
IOL_TX_DATA_TF4[183:0]	Output	Aside from reset, the 46 used IOLs utilize only TX_DATA [3:0], with the remaining bits tied to 1
IOL_TX_DATA_TF7[6:0]	Output	Mem_rst uses only TX_DATA[7], so the remaining 6 bits are tied to 1
IOL_IODLY_CTRL[179:0]	Output	All 60 IOLs are unused and need to be tied to 1
IOL_MIPI_SW_DYN_I[59:0]	Output	All 60 IOLs are unused and need to be tied to 1
IOL_TS_CTRL_TF4[51:0]	Output	The 13 unused IOL TS_CTRL need to be tied to 1
IOL_TS_CTRL_TF2[91:0]	Output	The 46 used IOL ts_ctrl utilize only the lower 2 bits, with the remaining upper 2 bits tied to 1
IOL_TS_CTRL_TF3[2:0]	Output	The IOL ts_ctrl for transfer reset uses only [0], with the remaining upper 3 bits tied to 1
MEM_RST_EN	Output	The IOL for transfer reset is in an output state
SRB_RST_DLL	Input	dll reset signal from srb, controlled by FPGA core, active-high enable signal.
DLL_UPDATE_N	Input	Passing through HMEMC to output to DLL update signal
SRB_DLL_FREEZE	Input	Passing through HMEMC to output to DLL freeze signal
SRB_IOL_RST	Input	Reset signal for 47 IOLs from srb, controlled by FPGA core, active-high enable signal
SRB_DQS_RST	Input	Reset signal for 5 dqs_ddc from srb, controlled by FPGA core, active-high enable signal.
SRB_DQS_RST_TRAINING	Input	The reset signal of the DQS_DDC controls the reset of the read/write pointer generation and read pointer control logic of DQSL upon completion of DQS gate training, with the signal being active-high.

### 9.3.4 Parameter Description

Table 9-8 GTP\_DDRPHY Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
MR0_DDR3	binary	16'h0~16'hffff	16'h1108	Phy-cached mr0_ddr3 value For the specific meaning of each bit in the registers, please refer to the DDR protocol
MR1_DDR3	binary	16'h0~16'hffff	16'h0001	Phy-cached mr1_ddr3 value
MR2_DDR3	binary	16'h0~16'hffff	16'h0000	Phy-cached mr2_ddr3 value

Parameter Name	Parameter Type	Valid Values	Defaults	Description
MR3_DDR3	binary	16'h0~16'hffff	16'h0000	Phy-cached mr3_ddr3 value
MR_DDR2	binary	16'h0~16'hffff	16'h0100	Phy-cached mr_ddr2 value
EMR1_DDR2	binary	16'h0~16'hffff	16'h0401	Phy-cached emr1_ddr2 value
EMR2_DDR2	binary	16'h0~16'hffff	16'h0000	Phy-cached emr2_ddr2 value
EMR3_DDR2	binary	16'h0~16'hffff	16'h0000	Phy-cached emr3_ddr2 value
MR_LPDDR	binary	16'h0~16'hffff	16'h0003	Phy-cached mr_lpddr value
EMR_LPDDR	binary	16'h0~16'hffff	16'h0000	Phy-cached emr_lpddr value
TMRD	integer	0~3	0	tMRD counter value, if there is a remainder, then add 1 to the quotient, all above are in phy cycles
TMOD	integer	0~7	0	tMOD counter value, if there is a remainder, then add 1 to the quotient, all above are in phy cycles
TZQINIT	Integer	0~1023	0	tZQinit counter value, if there is a remainder, then add 1 to the quotient, all above are in phy cycles
TRP	integer	0~7	0	Trp counter value, if there is a remainder, then add 1 to the quotient, all above are in phy cycles
TXPR	integer	0~15	0	tXPR counter value, if there is a remainder, then add 1 to the quotient, all above are in phy cycles
TRFC	Integer	0~255	0	Trfc counter value, if there is a remainder, then add 1 to the quotient, all above are in phy cycles.
WL_EN	string	"TRUE" "FALSE"	"FALSE"	Write Leveling enable 1'b0 : disable, default. 1'b1: enable
DDR_TYPE	string	"DDR3","DDR2","LPDDR"	"DDR3"	DDR Type
DATA_WIDTH	string	"16BIT","8BIT"	"16BIT"	"16BIT":full data width "8BIT": half data width
DQS_GATE_MODE	binary	2'b00~2'b11	2'b00	Dqs_gate mode select 2,b00: Determine gate boundary by value, default; 2'b01: Determine gate boundary by surplus pin loopback; 2'b10: Determine gate boundary by the improved dgts;
WRDATA_PATH_ADJ	string	"TRUE" "FALSE"	"FALSE"	Write data path offset adjustment 1'b0: No data offset, default; 1'b1: Data offset

Parameter Name	Parameter Type	Valid Values	Defaults	Description
CTRL_PATH_ADJ	string	"TRUE" "FALSE"	"FALSE"	Write command path offset adjustment 1'b0: No command offset, default; 1'b1: Command offset
WL_MAX_STEP	binary	8'h0~8'hff	8'h0	Maximum adjustment steps of write leveling, range: 0~255
WL_MAX_CHECK	binary	5'h0~5'h11111	5'h0	Maximum adjustment count of write leveling, range: 0~31
MAN_WRLVL_DQS_L	string	"TRUE" "FALSE"	"FALSE"	Manually adjust lower 8bit (dqsh) write leveling enable 1'b0:disable 1'b1:enable
MAN_WRLVL_DQS_H	string	"TRUE" "FALSE"	"FALSE"	Manually adjust upper 8bit (dqsh) write leveling enable 1'b0:disable 1'b1:enable
WL_CTRL_L	binary	3'b000~3'b111	3'b000	Step fine-tuning of lower 8bit (dqsl) write leveling. Refer to DDC SPEC for usage instructions
WL_CTRL_H	binary	3'b000~3'b111	3'b000	Step fine-tuning of upper 8bit (dqsh) write leveling. Refer to DDC SPEC for usage instructions
INIT_READ_CLK_CTRL	binary	2'b00~2'b11	2'b00	Provide initial gate read_clk_ctrl offset, determined by the actual SDRAM. For the lower 8bit DQ 2'b00: No offset 2'b01: Offset by 1/4 cycle 2'b10: Offset by 1/2 cycle 2'b11: Offset by 3/4 cycle
INIT_READ_CLK_CTRL_H	binary	2'b00~2'b11	2'b00	Provide initial gate read_clk_ctrl offset, determined by the actual SDRAM. For the upper 8bit DQ 2'b00: No offset 2'b01: Offset by 1/4 cycle 2'b10: Offset by 1/2 cycle 2'b11: Offset by 3/4 cycle
INIT_SLIP_STEP	binary	4'h0~4'hf	4'h0	Provide initial position for gate dqs_gate_ctrl, Value range: 0~7, for the lower 8bit DQ
INIT_SLIP_STEP_H	binary	4'h0~4'hf	4'h0	Provide initial position for gate dqs_gate_ctrl, Value range: 0~7, for the upper 8bit DQ.
FORCE_READ_CLK_CTR_L_L	string	"TRUE" "FALSE"	"FALSE"	Manually adjust the lower 8bit (dqsl) gate

Parameter Name	Parameter Type	Valid Values	Defaults	Description
				1'b0:disable 1'b1:enable
FORCE_READ_CLK_CTR_L_H	string	"TRUE" "FALSE"	"FALSE"	Manually adjust the upper 8bit (dqsh) gate 1'b0:disable 1'b1:enable
STOP_WITH_ERROR	string	"TRUE" "FALSE"	"TRUE"	Exit immediately after the first gate failure 1'b0 : disable,default
DQGT_DEBUG	binary	1'b0~1'b1	1'b0	Enable signal for keeping PHY always in gate state: 1'b0:disable,default 1'b1:enable
WRITE_DEBUG	binary	1'b0~1'b1	1'b0	Enable signal for continuous data transmission when PHY serves as DQS gate 1'b0:disable, default 1'b1:enable
RDEL_ADJ_MAX_RANG	binary	5'b00000~5'b1111	5'h0	Maximum number of rdel adjustments, 0~31
MIN_DQSI_WIN	binary	4'h0~4'hf	4'h0	Effective window size for dqsi, 0~15
INIT_SAMP_POSITION	binary	8'h0~8'hff	8'h0	Initial sampling point position for dqsi: 0~255 For the lower 8bit DQ
INIT_SAMP_POSITION_H	binary	8'h0~8'hff	8'h0	Initial sampling point position for dqsi: 0~255 For the upper 8bit DQ
FORCE_SAMP_POSITION_L	string	"TRUE""FALSE"	"FALSE"	Enable force use of the lower 8bits (dqsl) to set the initial sampling position 1'b0:disable 1'b1:enable
FORCE_SAMP_POSITION_H	string	"TRUE" "FALSE"	"FALSE"	Enable force use of the upper 8bits (dqsl) to set the initial sampling position 1'b0:disable 1'b1:enable
T200US	integer	0~131071	17'h0D2F0	Provide the timing counter value for 200μs timing in unit of PHY clock cycle; if there is a remainder, add 1 to the quotient
T400NS	integer	0~127	0	1. For DDR2, timing is 400ns, calculated by phy cycle 2. Others, default to 0
T_LPDDR	binary	9'h0~9'h1ff	9'h0	[2:0] Configured only for ddr2/lpddr [8:3] Configured only for lpddr in phy cycle
RDEL_RD_CNT	binary	19'h0~19'hffff	19'h 0	Number of read operations sent during rdel calibration; during simulation, please set this value to 2 to the 16th power

Parameter Name	Parameter Type	Valid Values	Defaults	Description
				During actual board debugging, configure the appropriate value based on the situation
REF_CNT	binary	8'h00~8'hff	8'h00	[3:0] init_ref_cnt Number of refreshes during initialization [7:4] rd_ref_cnt Number of refreshes during read calibration Needs to be determined based on the actual sdram
APB_VLD	string	"TURE" "FALSE"	TURE	Data validity indicator configured by APB 1'b0: Invalid, default 1'B1: Valid
TEST_PATTERN1	binary	128'h0000_0000_0000_0000_0000_0000_0000_0000~128'hffff_ffff_ffff_ffff_ffff_ffff_ffff_ffff	128'h0000_ffff_0000_ffff_0000_ffff_0000_ffff	Test pattern1. dqs gate training pattern,
TEST_PATTERN2	binary	32'h0000_0000~32'hffff_ffff	32'h0000_0000	Test pattern2
TEST_PATTERN3	binary	32'h0000_0000~32'hffff_ffff	32'h0000_0000	Test pattern3
TXS	binary	8'h0~8'hff	8'h0	Timing counter value for txs in phy cycle
TRAIN_RST_TYPE	string	"TRUE" "FALSE"	FALSE	Enable switching between two versions: False: Use the original version, which only supports data comparison to determine gate position; DDR will enter a self-refresh state during reset in calibration; True: Use the modified version, which supports data comparison or dgts method to determine gate position; DDR does not need to enter a Self-refresh state during reset in calibration.
WL_SETTING	binary	1'b0 ~1'b1	1'b1	Write leveling strategy select. 0: End when the initial sample is 1; 1: Continue when the initial sample is 1.
WCLK_DEL_SEL	binary	1'b0 ~1'b1	1'b0	Phase select of drive clock WCLK_DEL for output DQ signal 0:270 °, 1:90 °
INIT_WRLVL_STEP_L	binary	8'h00~8'hff	8'h00	Fixed wrlvl step used by the lower eight bits DQ when Wl_en is 0

Parameter Name	Parameter Type	Valid Values	Defaults	Description
INIT_WRLVL_STEP_H	binary	8'h00~8'hff	8'h00	Fixed wrlvl step used by the upper eight bits DQ when W1_en is 0

### 9.3.5 Detailed Functional Description

- Fully compatible with the DFI3.1 interface of uMCTL2 (synopsys);
  - Support SDRAM types: LPDDR, DDR2, DDR3, and DDR3L;
  - Support DDR frequency up to 1066Mbps;
  - Only supports a frequency ratio of 1:2 between MC and PHY;
  - The supported data width is 16/8;
  - DDR3 supports BL8; DDR2 and LPDDR support BL4 and BL8;
  - Only supports single rank;
  - Not support ECC
  - Provide a debug interface for monitoring and controlling the DDR calibration process;
- Provide an APB interface for receiving configuration information or phy parameters;

## 9.4 Usage Instructions for GTP\_IDDR\_E2

### 9.4.1 Supported Devices

Table 9-9 Device Models That Support GTP\_IDDR\_E2

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 9.4.2 Description of Functionality

GTP\_IDDR\_E2 supports edge-aligned pipeline DDR1TO2 rate conversion. The structure block diagram is shown below.



Figure 9-2 GTP\_IDDR\_E2 Structure Block Diagram

### 9.4.3 Port Description

Table 9-10 GTP\_IDDR\_E2 Port Description

Port	Direction	Width	Description
D	Input	1	Data input
RS	Input	1	Local reset signal
CLK	Input	1	Input system clock
Q0	Output	1	Data output
Q1	Output	1	Data output

### 9.4.4 Parameter Description

Table 9-11 GTP\_IDDR\_E2 Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
GRS_EN	String	"TRUE""FALSE"	"TRUE"	Global reset enable
LRS_EN	String	"TRUE""FALSE"	"FALSE"	Local reset enable

### 9.4.5 Instantiation template

```

GTP_IDDR_E2#(
    .GRS_EN ("TRUE"),
    .LRS_EN ("FALSE")
)
GTP_IDDR_E2_inst(
    .D      (d),
    .RS     (rs),
    .CLK   (clk),
    .Q0    (q0),
    .Q1    (q1)
);

```

## 9.5 Usage Instructions for GTP\_ODDR\_E2

### 9.5.1 Supported Devices

Table 9-12 Device Models That Support GTP\_ODDR\_E2

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 9.5.2 Description of Functionality

GTP\_ODDR\_E2 supports edge-aligned DDR2TO1 rate conversion. The structure block diagram is shown below.

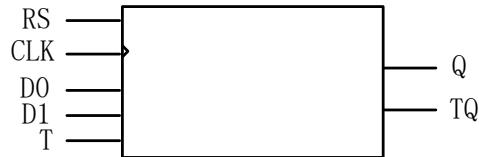


Figure 9-3 GTP\_ODDR\_E2 Structure Block Diagram

### 9.5.3 Port Description

Table 9-13 GTP\_ODDR\_E2 Port Description

Port	Direction	Width	Description
RS	Input	1	Local reset/set signal
CLK	Input	1	Input system clock
D0	Input	1	Input data signal
D1	Input	1	Input data signal
T	Input	1	Strobe data input
Q	Output	1	Data output
TQ	Output	1	Output strobe

### 9.5.4 Parameter Description

Table 9-14 GTP\_ODDR\_E2 Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
GRS_EN	String	"TRUE""FALSE"	"TRUE"	Global reset enable
LRS_EN	String	"TRUE""FALSE"	"FALSE"	Local reset enable

### 9.5.5 Instantiation template

```
GTP_ODDR_E2#  
(  
    .GRS_EN ("TRUE"),  
    .LRS_EN ("TRUE")  
) GTP_ODDR_E2_INST (  
    . RS(RS),  
    .CLK (RCLK),  
    .D0(D0),  
    .D1(D1),  
    .T (T),  
    .Q (Q),  
    .TQ (TQ)  
)
```

## Chapter 10 Usage Instructions for Other GTPs

---

### 10.1 Usage Instructions for GTP\_GRS

#### 10.1.1 Supported Devices

Table 10-1 Device Models That Support GTP\_GRS

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 10.1.2 Description of Functionality

This GTP is used to control the global reset signal and must be instantiated during simulation.

#### 10.1.3 Port Description

Table 10-2 GTP\_GRS Port Description

Port	Direction	Description
GRS_N	Input	Global reset

#### 10.1.4 Instantiation template

```
GTP_GRS GRS_INST(
    .GRS_N (grs_n)
);
```

## 10.2 Usage Instructions for GTP\_START\_E1

#### 10.2.1 Supported Devices

Table 10-3 Device Models That Support GTP\_START\_E1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

#### 10.2.2 Description of Functionality

This GTP describes the process of releasing the global signal for the wake-up operation.

### 10.2.3 Port Description

Table 10-4 GTP\_START\_E1 Port Description

Port	Direction	Description
CLK	Input	Wake-up clock
GOE	Input	Global IO output enable
GRS_N	Input	Global reset
GWE_N	Input	Memory write enable
WAKEUP_OVER	Output	Wake-up completion flag signal

### 10.2.4 Instantiation template

```
GTP_START GTP_START_inst(
```

```
    .CLK      (clk ),  
    .GOE      (gouten),  
    .GRS_N    (grs_n),  
    .GWE      (gwe),  
    .WAKEUP_OVER   (wakeup_over)
```

```
);
```

### 10.2.5 Detailed Functional Description

Used for wake-up operation and the release of the global signal.

## 10.3 Usage Instructions for GTP\_RES\_CAL

### 10.3.1 Supported Devices

Table 10-5 Device Models That Support GTP\_RES\_CAL

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Not supported	Not supported	Not supported	Not supported

### 10.3.2 Description of Functionality

GTP\_RES\_CAL describes the input/output characteristics for IO impedance calibration in user mode.

### 10.3.3 Port Description

Table 10-6 GTP\_RES\_CAL Port Description

<b>Port</b>	<b>Direction</b>	<b>Description</b>
CAL_REQ	Input	Calibration request signal from the user; 1 indicates validity
CAL_DONE	Output	Output calibration completion signal; 1 indicates completion

### 10.3.4 Parameter Description

Table 10-7 GTP\_RES\_CAL Parameter Description

<b>Parameter Name</b>	<b>Parameter Type</b>	<b>Valid Values</b>	<b>Defaults</b>	<b>Description</b>
BANK_INDEX	String	L0,L1,L2,R0,R1,R2	L0	IO bank location

### 10.3.5 Instantiation template

```
GTP_RES_CAL#(
    .BANK_INDEX (L0),
    .CAL_DONE    (cal_done),
    .CAL_REQ     (cal_req)
);
```

### 10.3.6 Detailed Functional Description

The timing for the input/output signals (signals in lowercase letters are intermediate signals) is shown in the figure below. Initially, CAL\_DONE=1; when CAL\_REQ goes through a pulse, the calibration begins, and CAL\_DONE flips to 0; After the counter counts 1050 cycles, the calibration is completed and CAL\_DONE flips back to 1.

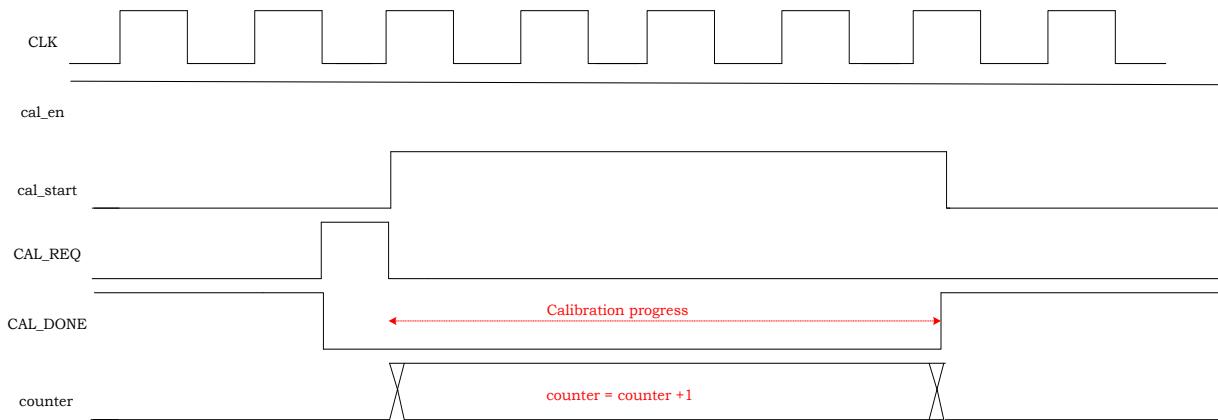


Figure 10-1 GTP\_RES\_CAL Calibration Mode Timing

### 10.3.7 How to Use GTP

GTP\_RES\_CAL is directly mapped onto RESCAL.

## 10.4 Usage Instructions for GTP\_APM\_E1

### 10.4.1 Supported Devices

Table 10-8 Device Models That Support GTP\_APM\_E1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Supported	Supported	Supported

### 10.4.2 Description of Functionality

GTP\_APM\_E1 is an arithmetic logic unit that supports various types of logical operations, including 9-bit and 18-bit multiplication, multiply-accumulate operations, general multiply-add operations, and FIR logic operations. It also functions an optional Preadd function supporting signed and unsigned numbers, optional input/output registers, and enable signals being active-high. For detailed functions, please refer to "**UG020003\_Logos Family FPGAs Arithmetic Processing Module (APM) User Guide**". The block diagram of GTP\_DRM18K structure is shown in the figure below.

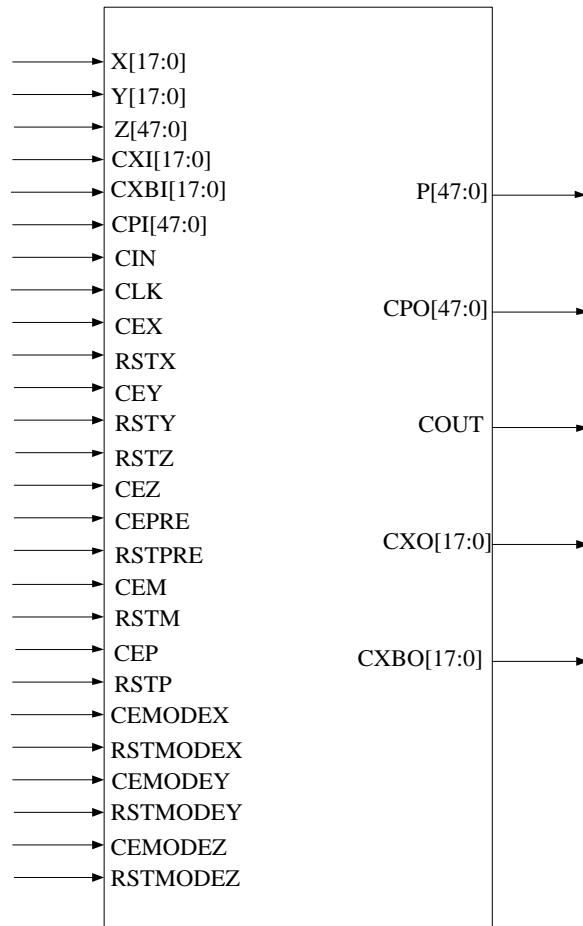


Figure 10-2 GTP\_APM\_E1 Structure Block Diagram

#### 10.4.3 Port Description

Table 10-9 GTP\_APM\_E1 Port Description

<b>Port</b>	<b>Direction</b>	<b>Description</b>
X[17:0]	Input	Parallel data input X, and X[17:0] serves as the X input for the multiplier or preadder
CXI[17:0]	Input	Cascade X input, from the CXO port of the previous APM module
CXBI[17:0]	Input	Cascade XB input, from the CXBO port of the previous APM module
Y[17:0]	Input	Parallel data input Y, the Y input for the multiplier
Z[47:0]	Input	Parallel data input Z, Z[47:0] as Postadder input data, and Z[47:30] can serve as preadder input
CPI[47:0]	Input	Cascade P input, from the CPO port of the previous APM module
CIN	Input	Cascade CIN input, from the COUT port of the previous APM module
MODEX	Input	APM dynamic X-side control operator, see Mode Description for function details
MODEY[2:0]	Input	APM dynamic Y-side control operator, see Mode Description for function details
MODEZ[3:0]	Input	APM dynamic Z-side control operator, see Mode Description for function details
CLK	Input	Clock input, used for all internal registers
CEX	Input	Clock enable signal, active-high, for X port input registers; the specific

Port	Direction	Description
		number of registers used is determined by the parameters X_REG and CXO_REG
RSTX	Input	and CXO_REG
CEY	Input	Active-high X reset signal to reset all X input registers
RSTY	Input	Clock enable signal for YREG, used when Y_REG=1
CEZ	Input	Active-high Y reset signal to reset Y input registers
RSTZ	Input	Clock enable signal for ZREG, used when Z_REG=1
CEPRE	Input	Active-high Z reset signal to reset Z input registers
RSTPRE	Input	Clock enable signal for PREREG
CEM	Input	Active-high PRE reset signal to reset PRE registers
RSTM	Input	Clock enable signal for MREG
CEP	Input	Active-high M reset signal to reset M registers
RSTP	Input	Clock enable signal for PREG
CEMODEX	Input	Active-high P reset signal to reset P registers
RSTMODEX	Input	Clock enable signal for MODEXREG
CEMODEY	Input	Active-high MODEX reset signal to reset MODEX input registers
RSTMODEY	Input	Clock enable signal for MODEYREG
CEMODEZ	Input	Active-high MODEY reset signal to reset MODEY input registers
RSTMODEZ	Input	Clock enable signal for MODEZREG
P[47:0]	Output	Active-high MODEZ reset signal to reset MODEZ input registers
CPO[47:0]	Output	48-bit parallel data output for APM
COUT	Output	Cascaded P output to connect to the next APM's CPI
CXO[17:0]	Output	Cascaded CIN output to connect to the next APM's CIN
CXBO[17:0]	Output	Cascaded X output to connect to the next APM's CXI

#### 10.4.4 Paramater Description

Table 10-10 GTP\_APM\_E1 Parameter Description

Parameter	Defaults	Setting Value	Description
GRS_EN	TRUE	FALSE: Not enabled TRUE: Enabled	Global reset signal enable
ASYNC_RST	0	0: Synchronous reset 1: Asynchronous reset	Reset selection
X_SIGNED	0	0 = Unsigned 1 = Signed	Signedness of X
Y_SIGNED	0	0 = Unsigned 1 = Signed	Signedness of Y
USE_PREADD	0	0: Not used 1: Used	Preadder selection
USE_POSTADD	0	0: Not used 1: Used	Postadder selection
X_REG	0	0: Not used 1: Used	X Register selection
CXO_REG	0	2'b00 = Zero cycle 2'b01 = One cycle	X output register delay selection

Parameter	Defaults	Setting Value	Description
		2'b10 = Two cycles 2'b11 = Three cycles	
Y_REG	0	0: Not used 1: Used	Y register selection
Z_REG	0	0: Not used 1: Used	Z register selection
PREADD_REG	0	0: Not used 1: Used	Preadder register selection
P_REG	0	0: Not used 1: Used	Postadder register selection
MULT_REG	0	0: Not used 1: Used	Multiplier register selection
MODEX_REG	0	0: Not used 1: Used	MODEX register selection
MODEY_REG	0	0: Not used 1: Used	MODEY register selection
MODEZ_REG	0	0: Not used 1: Used	MODEZ register selection
CPO_REG	0	0: Not used 1: Used	COUT, CPO register selection
X_SEL	0	0: X 1: XI	X input selection
XB_SEL	0	2'b00: Zi[47:30] 2'b01: XBI 2'b10: XBI_reg 2'b11: XO	Cascaded preadder input selection
Z_INIT	0	User defined	Z static input
USE_ACCLOW	0	0: Not used 1: Used	Only the lower 18 bits feedback input selection is used to enable Postadder
USE SIMD	0	0: Mode 18 1: Mode 9	SIMD mode selection
CIN_SEL	0	0: (MODEZ[3]  MODEY[2]) 1: CIN	Postadder carry input selection

The signals controlling the APM working mode are MODEX, MODEY[2:0], and MODEZ[3:0], with their port as described in the following table:

Table 10-11 GTP\_APM\_E1 MODE Port Description

Signals	Description
MODEX	Invert Zmux output, active-high
MODEZ[3]	Zmux Input selection
MODEZ[2:1]	18-bit arithmetic right shift to the previous stage's cascaded output, active-high, used for implementing wide multipliers such as 36*36
MODEZ[0]	Invert Ymux output, active-high
MODEY[2]	Invert Ymux output, active-high
MODEY[1]	Multiplier output to Postadder, active-low
MODEY[0]	0: Short-circuit multiplier output 1: Use Postadder feedback

Zmux input has multiple options, specifically determined by MODEZ[2:1]. This is shown in the

table below:

Table 10-12 ZMUX ListDescription

<b>MODEZ[2:1]</b>	<b>Description</b>
00	"ZINIT"[47:0] Postadder initialization constant
01	Postadder feedback
10	Z Input
11	Cascaded output of the previous stage APM

## 10.4.5 Detailed Functional Description

### 10.4.5.1 Multiplication Mode

As shown in the following figure, when APM is configured in multiplication mode, its equivalent arithmetic expression is:

$$P=X*Y$$

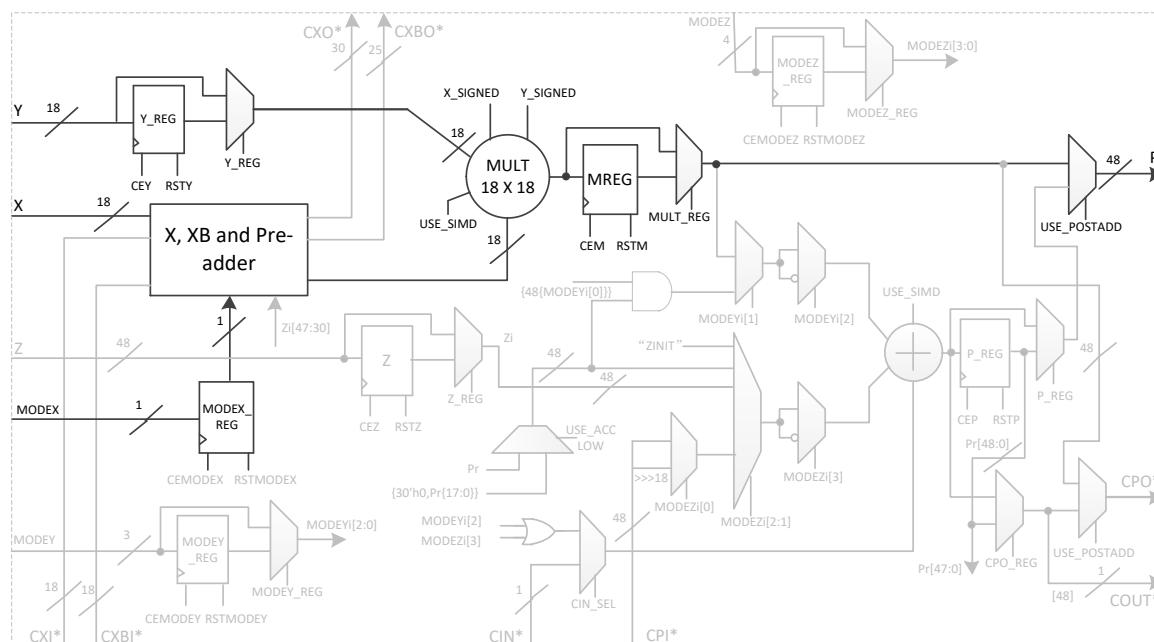


Figure 10-3 Application Diagram of GTP\_APM\_E1 Multiplication Mode

As shown in the following figure, after the Preadd Unit in APM is enabled, APM can be configured into a pre-addition multiplication mode, and its arithmetic expression is:

$$P=Y*(X+/-Z[47:30])$$

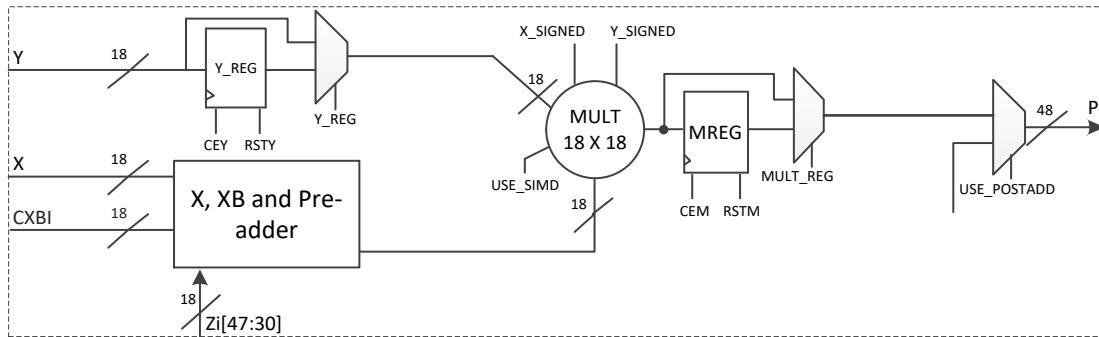


Figure 10-4 Application Diagram of Pre-addition Multiplication Mode

In this mode, each APM can perform two  $9*(9+/-9)$  operations or one  $18*(18+/-18)$  operation. In addition to the optional input/output registers, the multiplication mode with pre-addition can also choose to enable the internal pipeline registers, as illustrated in the following figure:

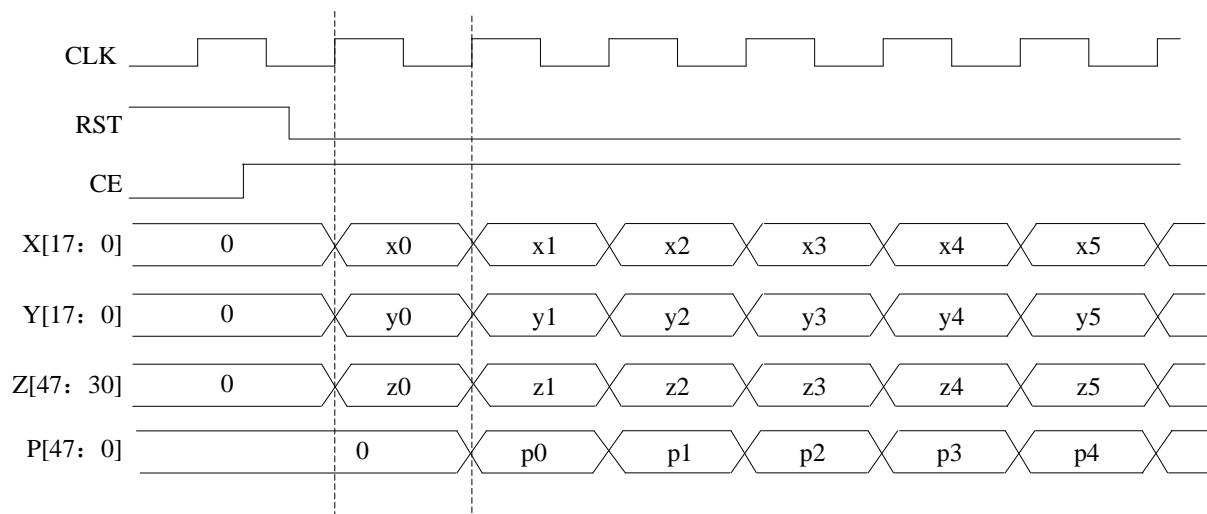


Figure 10-5 Typical Timing Diagram for Multiplication Mode

#### 10.4.5.2 General Multiply-Add Mode

As shown in the following figure, when APM is configured in general multiply-add mode, its arithmetic expression is:

$$P = X * Y +/ - Z$$

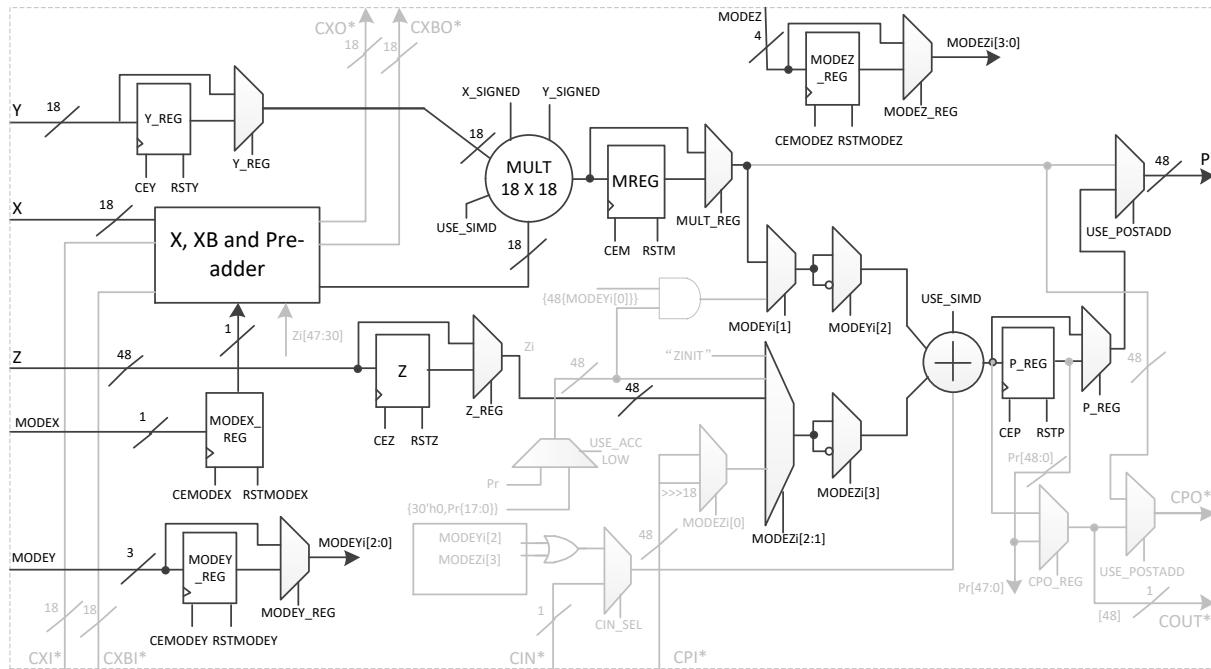


Figure 10-6 Application Diagram of GTP\_APM\_E1 General Multiply-Add Mode

The typical timing for the general multiply-add mode is as follows, where  $P=X*Y+Z$ . After P\_REG is enabled, the calculation result of the input data from the X and Y ports at the P-side will be output on the next rising edge of the clock.

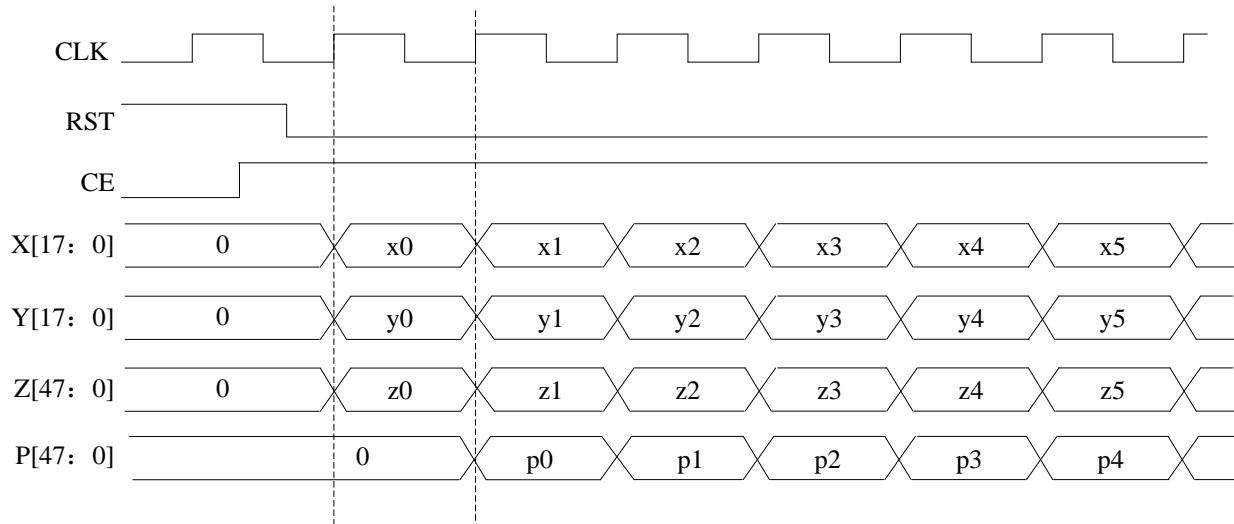


Figure 10-7 Typical Timing Diagram for General Multiply-Add Mode

#### 10.4.5.3 Multiply-Accumulate Mode

As shown in the following figure, when APM is configured in multiply-accumulate mode, its equivalent arithmetic expression is:

$$P = P +/- X * Y$$

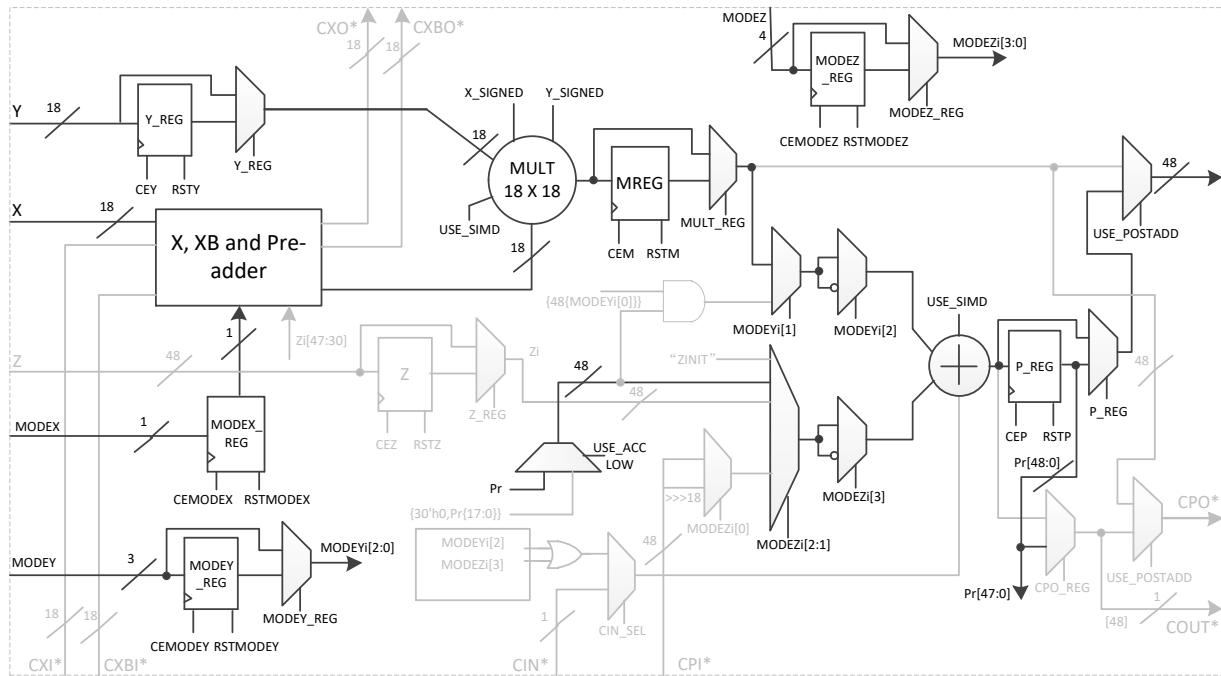


Figure 10-8 Application Diagram of GTP\_APM\_E1 Multiply-Accumulate Mode

As shown in the following figure, after the Preadd Unit in APM is enabled, APM can be configured into a multiply-accumulate mode with pre-addition, and its arithmetic expression is:

$$P = P +/- Y * (X +/- Z[47:30])$$

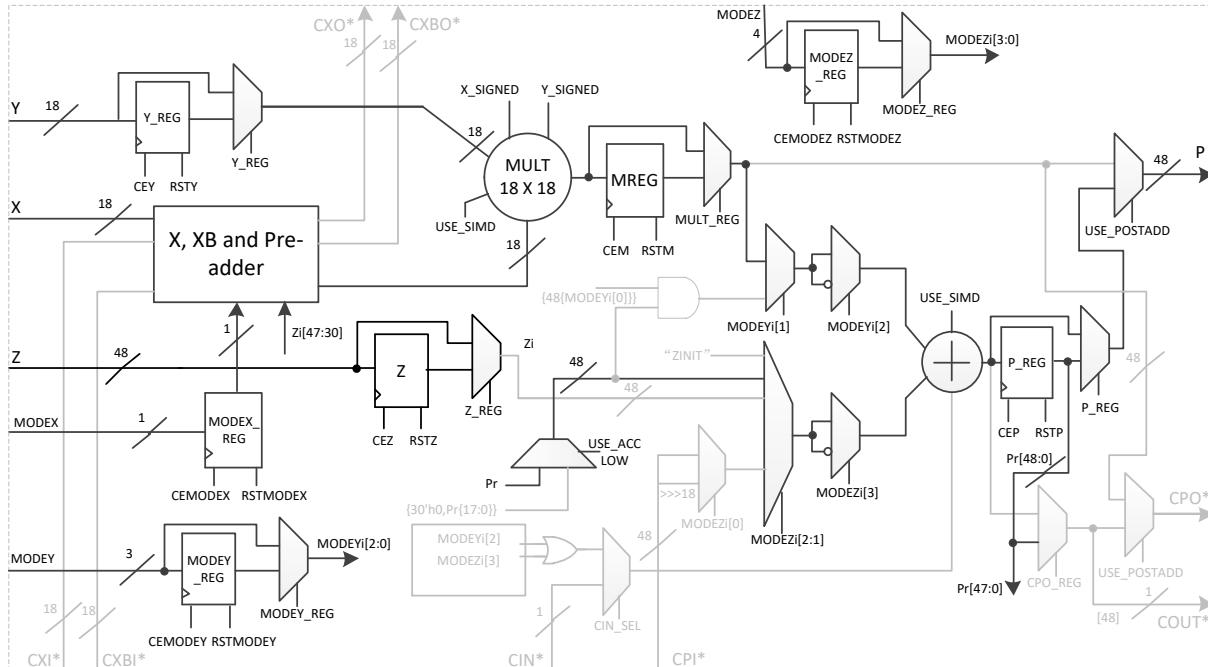


Figure 10-9 Application Diagram of Multiply-Accumulate Mode with Pre-Addition

The typical timing for the multiply-accumulate mode is as follows: where  $X=1$ ,  $Y=1$ ,  $P=P+X*Y$ , and  $P$  is output only when  $CEP$  is valid. After  $P\_REG$  is enabled, the calculation result of the input data from the  $X$  and  $Y$  ports at the  $P$ -side will be output on the next rising edge of the clock.

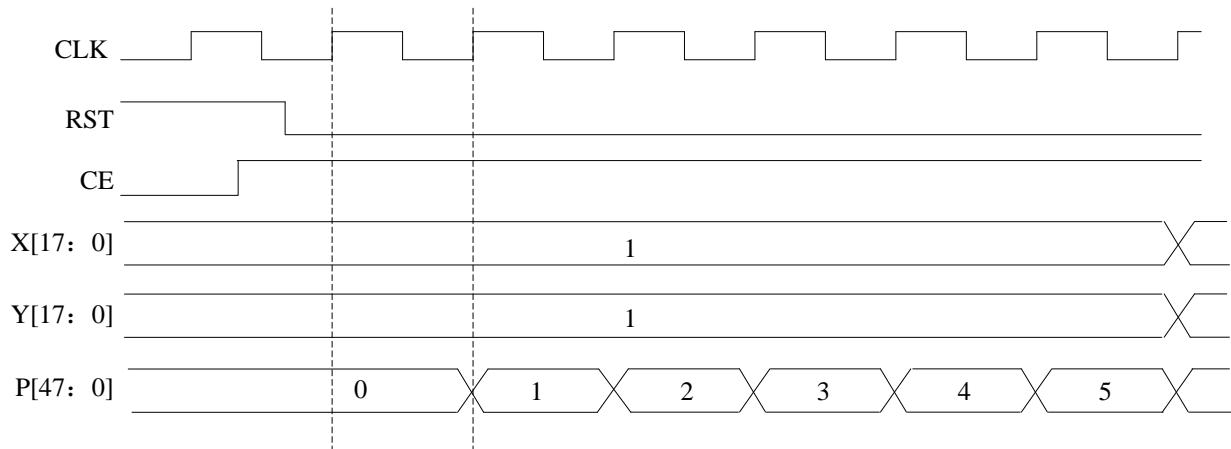


Figure 10-10 Typical Timing Diagram for Multiply-Accumulate Mode

#### 10.4.5.4 FIR Mode

FIR operations can also be achieved through APM cascading, and a typical FIR filter can be described by the following expression:

$$y_n = \sum_{i=0}^{N-1} x_{n-i} h_i = x_n h_0 + x_{n-1} h_1 + \dots + x_{n-N+1} h_{N-1}$$

$x$  is the input data stream,  $y$  is the output data stream, and  $h$  is the coefficient. The figure below is the functional implementation diagram for implementing a 24-tap Systolic FIR computation by cascading 24 APMs.

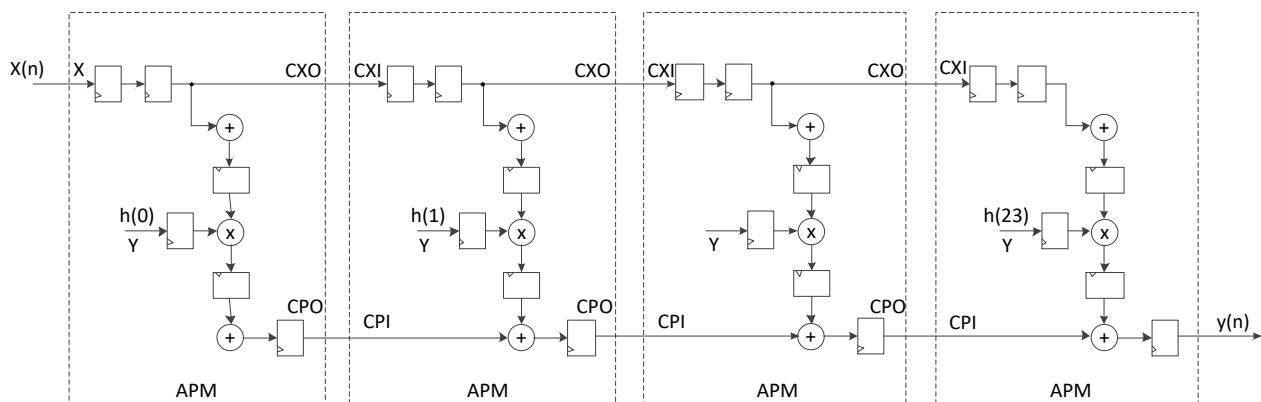


Figure 10-11 Systolic FIR Functional Diagram

#### 10.4.6 Instantiation template

GTP APM E1 #(

.GRS\_EN ( "TRUE" ),  
.X\_SIGNED ( 0 ),

```

.Y_SIGNED          ( 0      ),
.USE_POSTADD      ( 0      ),
.USE_PREADD       ( 0      ),
.PREADD_REG       ( 0      ),
.X_REG             ( 0      ),
.CXO_REG           ( 0      ),
.Y_REG             ( 0      ),
.Z_REG             ( 0      ),
.MULT_REG          ( 0      ),
.P_REG             ( 0      ),
.MODEX_REG         ( 0      ),
.MODEY_REG         ( 0      ),
.MODEZ_REG         ( 0      ),
.X_SEL             ( 0      ),
.XB_SEL            ( 0      ),
.ASYNC_RST          ( 0      ),
.USE SIMD          ( 0      ),
.Z_INIT             ( 48'h0000_0000_0000 ),
.CPO_REG           ( 0      ),
.USE_ACCLOW         ( 0      ),
.CIN_SEL            ( 0      )
)

```

#### GTP\_APM\_E1\_inst(

```

.P                ( P      ),
.CPO              ( CPO     ),
.COUT             ( COUT    ),
.CXO              ( CXO    ),
.CXBO             ( CXBO   ),
.X                ( X      ),
.CXI              ( CXI    ),
.CXBI             ( CXBI   ),
.Y                ( Y      ),
.Z                ( Z      ),
.CPI              ( CPI    ),

```

```

.CIN          ( CIN      ),
.MODEX        ( MODEX     ),
.MODEY        ( MODEY     ),
.MODEZ        ( MODEZ     ),
.CLK          ( CLK      ),
.RSTX          ( RSTX     ),
.RSTY          ( RSTY     ),
.RSTZ          ( RSTZ     ),
.RSTM          ( RSTM     ),
.RSTP          ( RSTP     ),
.RSTPRE        ( RSTPRE    ),
.RSTMODEX      ( RSTMODEX   ),
.RSTMODEY      ( RSTMODEY   ),
.RSTMODEZ      ( RSTMODEZ   ),
.CEX          ( CEX      ),
.CEY          ( CEY      ),
.CEZ          ( CEZ      ),
.CEM          ( CEM      ),
.CEP          ( CEP      ),
.CEPRE         ( CEPRE     ),
.CEMODEX      ( CEMODEX   ),
.CEMODEY      ( CEMODEY   ),
.CEMODEZ      ( CEMODEZ   ),
);

);
    
```

## 10.5 Usage Instructions for GTP\_HSST\_E2

### 10.5.1 Supported Devices

Table 10-13 Device Models That Support GTP\_HSST\_E2

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Not supported	Not supported	Not supported	Supported	Supported

### 10.5.2 Description of Functionality

GTP\_HSST\_E2 (High Speed Serial Transceiver) supports a variety of serial protocol standards.

Key items include: configurable DataRate speed; flexible selection of reference clock; independently configurable transceiver channels; configurable swing and de-emphasis; with data channels supporting 8bit only, 10bit only, 8B10B 8bit, 16bit only, 20bit only, 8B10B 16bit, 32bit only, 40bit only, 8B10B 32bit, 64B66B/64B67B 16bit, and 64B66B/64B67B 32bit modes; support protocols such as PCI Express GEN1, PCI Express GEN2, XAUI, Gigabit Ethernet, CPRI, SRIO. For detailed functionality and usage instructions, please refer to the "**UG020013\_Logos FPGA High-Speed Serial Transceiver (HSST) User Guide**". A simplified block diagram of its structure is shown below:

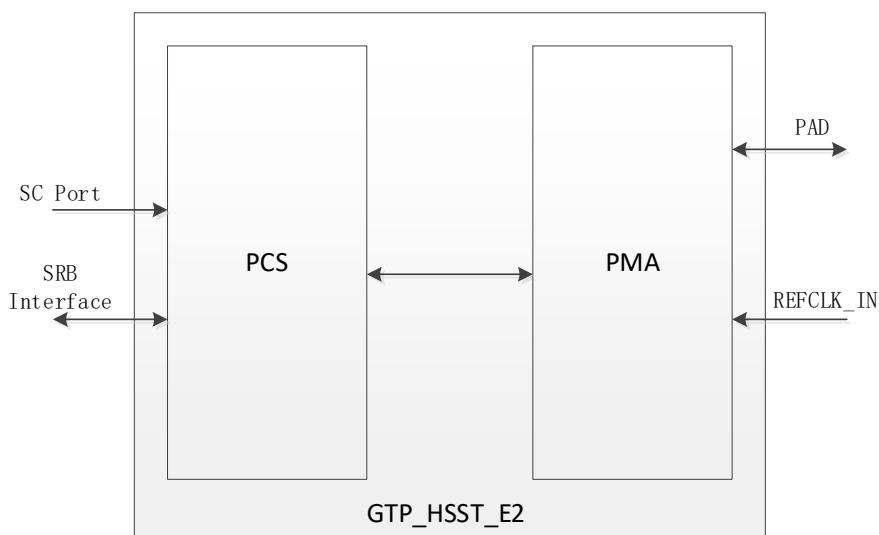


Figure 10-12 GTP\_HSST\_E2 Structure Block Diagram

### 10.5.3 Port Description

Table 10-14 GTP\_HSST\_E2 Port Description

Port	Width	Direction	Description
P_REFCLKP_y	1	I	Differential input reference clock positive end, HSST dedicated pin
P_REFCLKN_y	1	I	Differential input reference clock negative end, HSST dedicated pin
P_PLL_TEST_y	1	O	Test signal pad for pll debugging
P_RX_SDPx	1	I	Differential input data positive end, HSST dedicated pin
P_RX_SDNx	1	I	Differential input data negative end, HSST dedicated pin
P_TX_SDPx	1	O	Differential output data positive end, HSST dedicated pin
P_TX_SDNx	1	O	Differential output data negative end, HSST dedicated pin
P_RXx_CLK_FR_CORE	1	I	Receive clock from Fabric, serving as the RX interface data clock
P_RXx_CLK2_FR_CORE	1	I	Receive clock from Fabric, which is generated by P_REFCK2CORE through PLL frequency

Port	Width	Direction	Description
			multiplication and twice the frequency of P_RX_CLK_FR_CORE
P_TXx_CLK_FR_CORE	1	I	Transmit clock from Fabric, serving as the TX interface data clock
P_TXx_CLK_FR_CORE	1	I	Transmit clock from Fabric, which is generated by P_REFCK2CORE through PLL frequency multiplication and twice the frequency of P_RX_CLK_FR_CORE
P_HSST_RST	1	I	Perform a global reset on PCS, including TX and RX
P_PCS_RX_RST_x	1	I	Reset the PCS Receiver
P_PCS_TX_RST_x	1	I	Reset the PCS Transmitter
P_PCS_CB_RST_x	1	I	Reset the module after channel bonding
P_RXGEAR_SLIP_x	I	I	Slip indication to RX gearbox with 64B66B/67b decoder mode, edge-triggered, a shift of 1 bit occurs on the detection of either a rising or falling edge
P_CFG_CLK	1	I	Dynamically configure the clock input of the interface
P_CFG_RST	1	I	Dynamic configuration interface reset signal; All registers revert to the initial values set by the Parameter after reset
P_CFG_PSEL	1	I	Dynamic configuration interface selection signal
P_CFG_ENABLE	1	I	Dynamic configuration interface access enable
P_CFG_WRITE	1	I	Dynamic configuration interface read/write selection signal
P_CFG_ADDR[15:0]	16	I	Dynamic configuration interface write address
P_CFG_WDATA[7:0]	8	I	Dynamic configuration interface write data
P_TDATA_x[44:0]	45	I	Transmit data
P_PCS_WORD_ALIGN_EN[3:0]	4	I	When configured as effective control by the RX CLK Slip port control method, it serves as the RX CLK Slip control signal, an asynchronous signal. A rising edge from 0 to 1 causes the Deserializer module of the PMA RX to slip one bit; When configured for an external state machine, it serves as the Word Alignment enable signal.
P_RX_POLARITY_INVERT[3:0]	4	I	RX Sample Reg's polarity inversion enable
P_CEB_ADETECT_EN[3:0]	4	I	Test signal, set externally to 4'b1111' in normal mode
P_PCS_MCB_EXT_EN[3:0]	4	I	Channel bonding enable under external state machine mode
P_PCS_NEAREND_LOOP[3:0]	4	I	PCS near-end loopback control signal
P_PCS_FAREND_LOOP[3:0]	4	I	PCS far-end loopback control signal
P_PMA_NEAREND_PLOOP[3:0]	4	I	PMA near-end parallel loopback control signal
P_PMA_NEAREND_SLOOP[3:0]	4	I	PMA near-end serial loopback control signal
P_PMA_FAREND_PLOOP[3:0]	4	I	PMA far-end parallel loopback control signal
P_CFG_READY	1	O	Read/write ready output for the dynamic configuration interface
P_CFG_RDATA[7:0]	8	O	Read data for the dynamic configuration interface
P_CFG_INT	1	O	Dynamic configuration interface interrupt output

<b>Port</b>	<b>Width</b>	<b>Direction</b>	<b>Description</b>
P_PCS_RX_MCB_STATUS[3:0]	4	O	Channel Bonding control state machine lock signal
P_PCS_LSM_SYNCED[3:0]	4	O	Word Alignment successful, state machine lock flag
P_RDATA_x[46:0]	47	O	Receive data
P_RCLK2FABRIC[x]	1	O	Receive clock sent to Fabric
P_TCLK2FABRIC[x]	1	O	Transmit clock sent to Fabric
P_RESCAL_RST_I	1	I	Resistor calibration reset, internal test signal, connected to a fixed value of 0
P_RESCAL_I_CODE_I[5:0]	6	I	PMA manual configuration of the resistance value, internal test signal
P_RESCAL_I_CODE_O[5:0]	6	O	The resistor control output code when invalid, internal test signal
P_REFCK2CORE_y	1	O	Pin input reference clock output to Fabric
P_PLL_REF_CLK_y	1	I	PLL reference clock from Fabric
P_PLL_RST_y	1	I	PLL reset control
P_PLLPOWERDOWN_y	1	I	PLL powered-off control
P_PLL_READY_y	1	O	PLL Lock Status
P_LANE_SYNC_y	1	I	Synchronization signal of the transmit channel
P_LANE_SYNC_EN_y	1	I	Enable signal for the transmit channel synchronization signal
P_RATE_CHANGE_TCLK_ON_y	1	I	Synchronization control signal enable for dynamic switching
P_LANE_PD_x	1	I	LANE power control
P_LANE_RST_x	1	I	Lane reset, including RX LANE and TX LANE
P_RX_LANE_PD_x	1	I	RX LANE power control, including RX PMA and RX PCS
P_RX_PMA_RST_x	1	I	Reset the PMA Receiver
P_CTLE_ADJ_RST_x	1	I	Reset the linear equalizer at the PMA receive side
P_RX_SIGDET_STATUS_x	1	O	Port valid signal detection
P_RX_SATA_COMINIT_x	1	O	SATA COMINIT state
P_RX_SATA_COMWAKE_x	1	O	SATA COMWAKE state
P_RX_LS_DATA_x	1	O	Low-frequency signal output to Fabric
P_RX_READY_x	1	O	CDR has successfully locked the flag signal
P_TEST_STATUS_x[19:0]	20	O	RX output test status register, internal test signal
P_TX_DEEMP_x[1:0]	2	I	Transmitter de-emphasis control
P_TX_LS_DATA_x	1	I	Transmitted low-frequency signal
P_TX_BEACON_EN_x	1	I	Send beacon enable signal when it is high active and EI TX is enabled
P_TX_SWING_x	1	I	TX beacon enable signal
P_TX_RXDET_REQ_x	1	I	Receiver Detection request signal
P_TX_RATE_x[2:0]	3	I	TX line rate control signal
P_TX_BUSWIDTH_x[2:0]	3	I	Bit width selection from TX PCS to TX PMA
P_TX_MARGIN_x[2:0]	3	I	DAC source selection for Transmitter output swing

Port	Width	Direction	Description
P_TX_RXDET_STATUS_x	1	O	Receiver Detection result
P_TX_PMA_RST_x	1	I	Reset the PMA Transmitter
P_TX_LANE_PD_x	1	I	TX LANE power control, including TX PMA and TX PCS
P_RX_RATE_x[2:0]	3	I	RX line rate control signal
P_RX_BUSWIDTH_x[2:0]	3	I	Bit width selection from RX PMA to RX PCS
P_RX_HIGHZ_x	1	I	RX input high impedance control signal
P_CA_ALIGN_RX[3:0]	4	O	Receiving LANE CLK Aligner dynamic state output, a 0-to-1 transition indicates aligner success
P_CA_ALIGN_TX[3:0]	4	O	Transmitting LANE CLK Aligner dynamic state output, a 0-to-1 transition indicates aligner success
P_CIM_CLK_ALIGNER_RXx[7:0]	8	I	CLK Aligner delay step selection at the receive side
P_CIM_CLK_ALIGNER_TXx[7:0]	8	I	CLK Aligner delay step selection at the transmit side
P_CIM_DYN_DLY_SEL_RXx	1	I	Receive LANE's CLK Aligner enable signal
P_CIM_DYN_DLY_SEL_TXx	1	I	Transmit LANE's CLK Aligner enable signal
P_CIM_START_ALIGN_RXx	1	I	Input source for generating receive LANE CLK Aligner pulse, the setting value is triggered once on the rising edge
P_CIM_START_ALIGN_TXx	1	I	Input source for generating transmit LANE CLK Aligner pulse, the setting value is triggered once on the rising edge

Notes:

1. The signal named "SIGNAL\_NAME\_y" indicates there is one signal for each PLL:

SIGNAL\_NAME\_0: For PLL 0

SIGNAL\_NAME\_1: For PLL 1

2. The signal named "SIGNAL\_NAME\_x" indicates there is one signal for each Channel:

SIGNAL\_NAME\_0: For Channel 0

SIGNAL\_NAME\_1: For Channel 1

SIGNAL\_NAME\_2: For Channel 2

SIGNAL\_NAME\_3: For Channel 3

#### 10.5.4 Paramater Description

Table 10-15 GTP\_HSST\_E2 Parameter List

Parameter Name	Parameter Type	Valid Values	Description
PCS_CHx_BYPASS_W ORD_ALIGN	<string>	"FALSE","TRUE"	Active-high Bypass module Word Alignment
PCS_CHx_BYPASS_DE NC	<string>	"FALSE","TRUE"	Active-high Bypass module 8B10B Decoder
PCS_CHx_BYPASS_BO NDING	<string>	"FALSE","TRUE"	Active-high Bypass module Channel Bonding
PCS_CHx_BYPASS_CT C	<string>	"FALSE","TRUE"	Active-high Bypass module CTC

Parameter Name	Parameter Type	Valid Values	Description
PCS_CHx_BYPASS_GEAR	<string>	"FALSE","TRUE"	Active-high Bypass module RX Gear
PCS_CHx_BYPASS_BRIDGE	<string>	"FALSE","TRUE"	Active-high Bypass module RX Bridge unit
PCS_BYPASS_BRIDGE_FIFO	<string>	"FALSE","TRUE"	Active-high Bypass module RX Bridge FIFO
PCS_CHx_DATA_MODE	<string>	"X8", "X10", "X16", "X20"	PCS Receiver module bit width selection
PCS_CHx_RX_POLARITY_INV	<string>	"DELAY", "BIT_POLARITY_INVERSION", "BIT_REVERSAL", "BOTH"	RX Sample Reg module polarity inversion and bit order inversion
PCS_CHx_ALIGN_MODE	<string>	"1GB", "10GB", "RAPIDIO", "OUTSIDE"	Word Alignment Link State Machine selection
PCS_CHx_SAMP_16B	<string>	"X16", "X20"	PCS Receiver module bit width selection
PCS_CHx_FARLP_PWR_REDUCTION	<string>	"FALSE","TRUE"	Active-high, disable PCS far-end loopback FIFO
PCS_CHx_COMMAR_EG0	<integer>	0 to 1023	Word Alignment Comma byte definition 0
PCS_CHx_COMMASK	<integer>	0 to 1023	Word Alignment Comma Mask bit definition
PCS_CHx_CEB_MODE	<string>	"10GB", "RAPIDIO", "OUTSIDE"	Channel Bonding State Machine selection
PCS_CHx_CTC_MODE	<string>	"1SKIP", "2SKIP", "PCIE 2BYTE", "4SKIP"	CTC mode selection
PCS_CHx_A_REG	<integer>	0 to 255	Definition of Align Pattern used for channel bonding
PCS_CHx_GE_AUTO_EN	<string>	"FALSE","TRUE"	Active-high, enable automatic replacement from /C/ to /I2/ based on 1 Gig Ethernet
PCS_CHx_SKIP_REG0	<integer>	0 to 1023	SKIP character Byte 0 used by CTC
PCS_CHx_SKIP_REG1	<integer>	0 to 1023	SKIP character Byte 1 used by CTC
PCS_CHx_SKIP_REG2	<integer>	0 to 1023	SKIP character Byte 2 used by CTC
PCS_CHx_SKIP_REG3	<integer>	0 to 1023	SKIP character Byte 3 used by CTC
PCS_CHx_DEC_DUAL	<string>	"FALSE","TRUE"	PCS Receiver module bit width selection
PCS_CHx_SPLIT	<string>	"FALSE","TRUE"	PCS Receiver module bit width selection
PCS_CHx_FIFOFLAG_CTC	<string>	"FALSE","TRUE"	CTC priority encoding flag signal in RXstatus encoding
PCS_CHx_COMMADET_MODE	<string>	"COMMA_PATTERN", "RX_CLK_SLIP"	Alignment mode in Word Alignment module
PCS_CHx_ERRDETECT_SILENCE	<string>	"FALSE","TRUE"	Active-high, report an error when K-code encoding is incorrect
PCS_CHx_PMA_RCLK_POLINV	<string>	"PMA_RCLK", "REVERSE_OF_PMA_RCLK"	PMA_RCLK clock source selection
PCS_CHx_PCS_RCLK_SEL	<string>	"PMA_RCLK", "PMA_TCLK", "MCB_RCLK", "	PCS_RCLK clock source selection

Parameter Name	Parameter Type	Valid Values	Description
		"RCLK"	
PCS_CHx_CB_RCLK_SEL	<string>	"PMA_RCLK", "PMA_TCLK" "MCB_RCLK", " RCLK"	CB_RCLK clock source selection
PCS_CHx_AFTER_CTC_RCLK_SEL_y	<string>	"PMA_RCLK", "PMA_TCLK" "MCB_RCLK", " RCLK2"	AFTER_CTC_RCLK clock source selection
PCS_CHx_RCLK_POLINV	<string>	"RCLK", "REVERSE_OF_RCLK"	RCLK clock source selection
PCS_CHx_BRIDGE_RCLK_SEL	<string>	"PMA_RCLK", "PMA_TCLK", "MCB_RCLK", "RCLK"	CB_RCLK clock source selection
PCS_CHx_PCS_RCLK_EN	<string>	"FALSE", "TRUE"	PCS Receiver module bit width selection
PCS_CHx_CB_RCLK_EN	<string>	"FALSE", "TRUE"	PCS Receiver module bit width selection
PCS_CHx_AFTER_CTC_RCLK_EN	<string>	"FALSE", "TRUE"	PCS Receiver module bit width selection
PCS_CHx_AFTER_CTC_RCLK_EN_GB_y	<string>	"FALSE", "TRUE"	PCS Receiver module bit width selection
PCS_CHx_PCS_RX_RSTN	<string>	"FALSE", "TRUE"	Reset register of PCS Receiver
PCS_CHx_SLAVE	<string>	"MASTER", "SLAVE"	Channel bonding setting excluding PCI Express
PCS_CHx_PCIE_SLAVE	<string>	"MASTER", "SLAVE"	Channel bonding setting based on PCI Express
PCS_CHx_RX_64B66B_67B	<string>	"NORMAL", "64B_66B" "RESERVED"	64B66B_67B selection
PCS_CHx_RX_BRIDGE_CLK_POLINV	<string>	"RX_BRIDGE_CLK", "REVERSE OF RX_BRIDGE_CLK"	RX_BRIDGE_CLK clock source selection
PCS_CHx_PCS_CB_RSTN	<string>	"FALSE", "TRUE"	PCS CB reset register
PCS_CHx_TX_BRIDGE_GEAR_SEL	<string>	"FALSE", "TRUE"	Configuration selection for the order of bridge unit and gear modules in TX direction
PCS_CHx_TX_BYPASS_BRIDGE_UINT	<string>	"FALSE", "TRUE"	Bypass module TX Bridge unit control
PCS_CH0_TX_BYPASS_BRIDGE_FIFO	<string>	"FALSE", "TRUE"	Bypass module TX Bridge FIFO control
PCS_CHx_TX_BYPASS_GEAR	<string>	"FALSE", "TRUE"	TX gear module Bypass enable
PCS_CHx_TX_BYPASS_ENC	<string>	"FALSE", "TRUE"	8B10B Encoder module Bypass enable
PCS_CHx_TX_BYPASS_BIT_SLIP	<string>	"FALSE", "TRUE"	TX BitSlip module Bypass enable
PCS_CHx_TX_GEAR_SPLIT	<string>	"FALSE", "TRUE"	PCS Transmitter module bit width selection
PCS_CHx_TX_DRIVE_	<string>	"NO_CHANGE",	TX Drive Reg module polarity inversion and bit

Parameter Name	Parameter Type	Valid Values	Description
REG_MODE		"EN_POLARITY_REV", "EN_BIT_REV", "EN_BOTH"	order inversion
PCS_CHx_TX_BIT_SLI_P_CYCLES	<integer>	0 to 31	Determine the number of bits for Slip in the TX Bit Slip module
PCS_CHx_INT_TX_MASK_0	<string>	"FALSE","TRUE"	Active-high Mask transmit channel interrupt status register bit0
PCS_CHx_INT_TX_MASK_1	<string>	"FALSE","TRUE"	Active-high Mask transmit channel interrupt status register bit1
PCS_CHx_INT_TX_MASK_2	<string>	"FALSE","TRUE"	Active-high Mask transmit channel interrupt status register bit2
PCS_CHx_INT_TX_CLR_0	<string>	"FALSE","TRUE"	Active-high clear transmit channel interrupt status register bit0
PCS_CHx_INT_TX_CLR_1	<string>	"FALSE","TRUE"	Active-high clear transmit channel interrupt status register bit1
PCS_CHx_INT_TX_CLR_2	<string>	"FALSE","TRUE"	Active-high clear transmit channel interrupt status register bit2
PCS_CHx_TX_PMA_TCLK_POLINV	<string>	"PMA_TCLK", "REVERSE_OF_PMA_TCLK"	Whether PMA_TCLK is inverted
PCS_CHx_TX_PCS_CLK_EN_SEL	<string>	"FALSE","TRUE"	PCS Transmitter module bit width selection
PCS_CHx_TX_BRIDGE_TCLK_SEL	<string>	"TCLK2", "TCLK"	TCLK clock source selection
PCS_CHx_TX_TCLK_POLINV	<string>	"TCLK", "REVERSE_OF_TCLK"	Whether TCLK is an inverted clock
PCS_CHx_TX_PCS_TC_LK_SEL	<string>	"TCLK", "PMA_TCLK"	PCS_TCLK selection
PCS_CHx_TX_PCS_TX_RSTN	<string>	"FALSE","TRUE"	Reset register of PCS Transmitter
PCS_CHx_TX_SLAVE	<string>	"MASTER", "SLAVE"	Channel bonding setting at the transmit side
PCS_CHx_TX_GEAR_TCLK_EN_SEL	<string>	"FALSE","TRUE"	PCS Transmitter module bit width selection
PCS_CHx_DATA_WID_TH_MODE	<string>	"X20", "X16", "X10", "X8"	PCS Transmitter module bit width selection
PCS_CHx_TX_64B66B_67B	<string>	"NORMAL", "64B_66B", "RESERVED"	TX 64B66B_67B selection
PCS_CHx_TX_GEAR_TCLK_SEL	<string>	"TCLK2", "PMA_TCLK"	GEAR_TCLK selection
PCS_CHx_TX_TCLK2F_ABRIC_SEL	<string>	"FALSE","TRUE"	PCS Transmitter module bit width selection
PCS_CHx_TX_OUTZZ			Reserved
PCS_CHx_ENC_DUAL	<string>	"FALSE","TRUE"	PCS Transmitter module bit width selection
PCS_CHx_TX_BITSLIP_DATA_MODE			Reserved
PCS_CHx_TX_BRIDGE_CLK_POLINV	<string>	"TX_BRIDGE_CLK", "REVERSE OF TX_BRIDGE_CLK"	TX_BRIDGE_CLK clock selection
PCS_CHx_COMMA_REG1	<integer>	0 to 1023	Word Alignment Comma byte definition 1

Parameter Name	Parameter Type	Valid Values	Description
PCS_CHx_RAPID_IMA_X	<integer>	0 to 7	Number of bytes for lock state detection in Rapid IO Link State Machine
PCS_CHx_RAPID_VMI_N_1	<integer>	0 to 255	Number of bytes for exit state detection in Rapid IO Link State Machine, lower 8bits
PCS_CHx_RAPID_VMI_N_2	<integer>	0 to 255	Number of bytes for exit state detection in Rapid IO Link State Machine, upper 8bits
PCS_CHx_RX_PRBS_MODE	<string>	"DISABLE", "PRBS_7", "PRBS_15", "PRBS_23", "PRBS_31",	PRBS Checker mode selection at the receive side
PCS_CHx_RX_ERRCN_T_CLR	<string>	"FALSE", "TRUE"	PCS_ERR_CNT counter clear
PCS_CH0_RX_PRBS_E_RR_LPBK	<string>	"FALSE", "TRUE"	PRBS_ERR_LPBK enable
PCS_CHx_TX_PRBS_MODE	<string>	"DISABLE", "PRBS_7", "PRBS_15", "PRBS_23", "PRBS_31", "LONG_1", "LONG_0", "20UI", "D10_2", "PCIE"	PCS PRBS Generator mode selection at the transmit side
PCS_CHx_TX_INSERT_ER	<string>	"FALSE", "TRUE"	PRBS error code injection
PCS_CHx_ENABLE_P_RBS_GEN	<string>	"FALSE", "TRUE"	PCS PRBS Generator enable
PCS_CHx_ERR_CNT	<integer>	0 to 255	Checking error counter in PCS PRBS checker
PCS_CHx_DEFAULT_R_ADDR	<integer>	0 to 15	For Channel Bonding
PCS_CHx_MASTER_CHECK_OFFSET	<integer>	0 to 15	For Channel Bonding
PCS_CHx_DELAY_SET	<integer>	0 to 15	For Channel Bonding
PCS_CHx_SEACH_OF_FSET	<string>	"20BIT", "30BIT", "40BIT", "50BIT", "60BIT", "70BIT"	Channel Bonding range configuration
PCS_CHx_CEB_RAPID_LS_MMAX	<integer>	0 to 7	MMAX value used by Channel Bonding RapidIO state machine
PCS_CHx_CTC_AFULL	<integer>	0 to 31	CTC FIFO Almost Full Threshold
PCS_CHx_CTC_AEMPTY	<integer>	0 to 31	CTC FIFO Almost Empty Threshold
PCS_CHx_CTC_CONTI_SKP_SET	<integer>	0 to 1	CTC continuous SKP deletion enable
PCS_CHx_FAR_LOOP	<string>	"FALSE", "TRUE"	PCS far-end loopback enable
PCS_CHx_NEAR_LOOP	<string>	"FALSE", "TRUE"	PCS near-end loopback enable
PCS_CHx_REG_TX2RX_PLOOP_EN	<string>	"FALSE", "TRUE"	PMA near-end parallel loopback enable

Parameter Name	Parameter Type	Valid Values	Description
PCS_CHx_REG_RX2T_X_SLOOP_EN	<string>	"FALSE","TRUE"	PMA near-end serial loopback enable
PCS_CHx_REG_RX2T_X_PLOOP_EN	<string>	"FALSE","TRUE"	PMA far-end parallel loopback enable
PCS_CHx_INT_RX_MASK_0	<string>	"FALSE","TRUE"	Receive channel interrupt status register bit0 mask
PCS_CHx_INT_RX_MASK_1	<string>	"FALSE","TRUE"	Receive channel interrupt status register bit1 mask
PCS_CHx_INT_RX_MASK_2	<string>	"FALSE","TRUE"	Receive channel interrupt status register bit2 mask
PCS_CHx_INT_RX_MASK_3	<string>	"FALSE","TRUE"	Receive channel interrupt status register bit3 mask
PCS_CHx_INT_RX_MASK_4	<string>	"FALSE","TRUE"	Receive channel interrupt status register bit4 mask
PCS_CHx_INT_RX_MASK_5	<string>	"FALSE","TRUE"	Receive channel interrupt status register bit5 mask
PCS_CHx_INT_RX_MASK_6	<string>	"FALSE","TRUE"	Receive channel interrupt status register bit6 mask
PCS_CHx_INT_RX_MASK_7	<string>	"FALSE","TRUE"	Receive channel interrupt status register bit7 mask
PCS_CHx_INT_RX_CL_R_0	<string>	"FALSE","TRUE"	Clear receive channel interrupt status register bit0
PCS_CHx_INT_RX_CL_R_1	<string>	"FALSE","TRUE"	Clear receive channel interrupt status register bit1
PCS_CHx_INT_RX_CL_R_2	<string>	"FALSE","TRUE"	Clear receive channel interrupt status register bit2
PCS_CHx_INT_RX_CL_R_3	<string>	"FALSE","TRUE"	Clear receive channel interrupt status register bit3
PCS_CHx_INT_RX_CL_R_4	<string>	"FALSE","TRUE"	Clear receive channel interrupt status register bit4
PCS_CHx_INT_RX_CL_R_5	<string>	"FALSE","TRUE"	Clear receive channel interrupt status register bit5
PCS_CHx_INT_RX_CL_R_6	<string>	"FALSE","TRUE"	Clear receive channel interrupt status register bit6
PCS_CHx_INT_RX_CL_R_7	<string>	"FALSE","TRUE"	Clear receive channel interrupt status register bit7
PCS_CHx_CA_RX	<integer>	0 to 255	Static setting of CLK Aligner delay step at receive channel
PCS_CHx_CA_TX	<integer>	0 to 255	Static setting of CLK Aligner delay step at transmit channel
PCS_CHx_CA_DYN_DLY_EN_RX	<string>	"FALSE","TRUE"	Enable the CLK Aligner delay step of the receive channel
PCS_CHx_CA_DYN_DLY_EN_TX	<string>	"FALSE","TRUE"	Enable the CLK Aligner delay step of the transmit channel
PCS_CHx_CA_DYN_DLY_SEL_RX	<string>	"FALSE","TRUE"	Enable CLK Aligner of the corresponding receive channel
PCS_CHx_CA_DYN_DLY_SEL_TX	<string>	"FALSE","TRUE"	Enable CLK Aligner of corresponding transmit channel
PCS_CHx_CA_RSTN_RX	<string>	"FALSE","TRUE"	RX CLK Aligner reset
PCS_CHx_CA_RSTN_TX	<string>	"FALSE","TRUE"	TX CLK Aligner reset
PCS_CHx_RXPRBS_PWR_REDUCTION	<string>	"NORMAL","POWER_REDUCITION"	Active high to disable RX PRBS

Parameter Name	Parameter Type	Valid Values	Description
PCS_CHx_WDALIGN_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active high to disable RX wordalign
PCS_CHx_RXDEC_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable RX decoder
PCS_CHx_RXCB_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable RX channel bonding
PCS_CHx_RXCTC_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to Disable RX CTC Function
PCS_CHx_RXGEAR_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable RX GEAR
PCS_CHx_RXBRG_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable RX bridge
PCS_CHx_RXTEST_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable RX test_status
PCS_CHx_TXBRG_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable TX bridge
PCS_CHx_TXGEAR_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable TX GEAR
PCS_CHx_TXENC_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable TX encoder
PCS_CHx_TXBSLP_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable TX bitslip
PCS_CHx_TXPRBS_PWR_REDUCTION	<string>	"NORMAL","PO WER_REDUCTIO N"	Active-high to disable TX PRBS
PCS_CHx_TXBRG_FULL_CHK_EN	<string>	"FALSE","TRUE"	tx bridge unit fifo full detection enable
PCS_CHx_TXBRG_EMPTY_CHK_EN	<string>	"FALSE","TRUE"	tx bridge unit fifo empty detection enable
PCS_CHx_RXBRG_FUL_CHK_EN	<string>	"FALSE","TRUE"	Receive side bridge unit FIFO write full detection enable
PCS_CHx_RXBRG_EMPT_CHK_EN	<string>	"FALSE","TRUE"	Receive side bridge unit FIFO read empty detection enable
PCS_CHx_CTC_FULL_CHK_EN	<string>	"FALSE","TRUE"	Receive side CTC unit FIFO write full detection enable
PCS_CHx_CTC_EMPT_Y_CHK_EN	<string>	"FALSE","TRUE"	Receive side CTC unit FIFO read empty detection enable
PCS_CHx_CEB_FULL_CHK_EN	<string>	"FALSE","TRUE"	Receive side channel bonding unit FIFO write full detection enable
PCS_CHx_CEB_EMPT_Y_CHK_EN	<string>	"FALSE","TRUE"	Receive side channel bonding unit FIFO read empty detection enable
PCS_CHx_FLP_FULL_CHK_EN	<string>	"FALSE","TRUE"	Far-end loopback FIFO write full detection enable
PCS_CHx_FLP_EMPT_Y_CHK_EN	<string>	"FALSE","TRUE"	Far-end loopback FIFO read empty detection enable

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_RX_PD	<string>	"ON","OFF"	RX power control
PMA_CHx_REG_RX_PD_EN	<string>	"FALSE","TRUE"	enable RX power down control from register
PMA_CHx_REG_RX_C_LKPATH_PD	<string>	"ON","OFF"	RX clock path open enable
PMA_CHx_REG_RX_C_LKPATH_PD_EN	<string>	"FALSE","TRUE"	enable the RX clk path power down control from register
PMA_CHx_REG_RX_D_ATAPATH_PD	<string>	"ON","OFF"	RX data path open enable
PMA_CHx_REG_RX_D_ATAPATH_PD_EN	<string>	"FALSE","TRUE"	enable the RX data path power down control from register
PMA_CHx_REG_RX_S_IGDET_PD	<string>	"ON","OFF"	RX signal detection open enable
PMA_CHx_REG_RX_S_IGDET_PD_EN	<string>	"FALSE","TRUE"	enable RX signal detect power down control from register
PMA_CHx_REG_RX_D_CC_RST_N			Reserved
PMA_CHx_REG_RX_D_CC_RST_N_EN			Reserved
PMA_CHx_REG_RX_C_DR_RST_N	<string>	"FALSE","TRUE"	CDR reset signal
PMA_CHx_REG_RX_C_DR_RST_N_EN	<string>	"FALSE","TRUE"	enable the CDR reset control from register
PMA_CHx_REG_RX_S_IGDET_RST_N			Reserved
PMA_CHx_REG_RX_S_IGDET_RST_N_EN			Reserved
PMA_CHx_REG_RXPC_LK_SLIP	<string>	"FALSE","TRUE"	When the register changes from 0 to 1, rxpclk delays by one UI
PMA_CHx_REG_RXPC_LK_SLIP_OW	<string>	"DISABLE","ENABLE"	enable rxpclk slip control from register
PMA_CHx_REG_RX_P_CLKSWITCH_RST_N			Reserved
PMA_CHx_REG_RX_P_CLKSWITCH_RST_N_EN			Reserved
PMA_CHx_REG_RX_P_CLKSWITCH			Reserved
PMA_CHx_REG_RX_P_CLKSWITCH_EN			Reserved
PMA_CHx_REG_RX_H_IGHZ			Reserved
PMA_CHx_REG_RX_H_IGHZ_EN			Reserved
PMA_CHx_REG_RX_E_Q_C_SET			Reserved
PMA_CHx_REG_RX_E_Q_R_SET			Reserved
PMA_CHx_REG_RX_B_USWIDTH	<string>	"8BIT","10BIT","16BIT","20BIT"	RX data width
PMA_CHx_REG_RX_B_USWIDTH_EN	<string>	"FALSE","TRUE"	enable the RXdatawidth control from register
PMA_CHx_REG_RX_RATE	<string>	"DIV8","DIV4","DIV2","DIV1"	RX clock path division ratio

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_RX_RATE_EN	<string>	"FALSE","TRUE"	enable the RX rate control from register
PMA_CHx_REG_RX_RX_ES_TRIM			Reserved
PMA_CHx_REG_RX_RX_ES_TRIM_EN			Reserved
PMA_CHx_REG_RX_EQ_OF	<string>	"FALSE","TRUE"	Receive equalizer open enable
PMA_CHx_REG_RX_PREAMP_IC			Reserved
PMA_CHx_REG_RX_P_CLK_EDGE_SEL	<string>	"POS_EDGE","NEG_EDGE"	rpxclk edge select
PMA_CHx_REG_RX_P_IBUF_IC			Reserved
PMA_CHx_REG_RX_DCC_IC_RX			Reserved
PMA_CHx_REG_RX_DCC_IC_TX			Reserved
PMA_CHx_REG_RX_ICTRL_TRX			Reserved
PMA_CHx_REG_RX_ICTRL_SIGDET			Reserved
PMA_CHx_REG_RX_ICTRL_PREAMP			Reserved
PMA_CHx_REG_RX_ICTRL_SLICER			Reserved
PMA_CHx_REG_RX_ICTRL_PIBUF			Reserved
PMA_CHx_REG_RX_ICTRL_PI			Reserved
PMA_CHx_REG_RX_ICTRL_DCC			Reserved
PMA_CHx_REG_RX_ICTRL_PREDRV			Reserved
PMA_CHx_REG_RX_RATE	<string>	"DIV8","DIV4","DIV2","DIV1"	TX clock path division ratio
PMA_CHx_REG_RX_RATE_EN	<string>	"FALSE","TRUE"	enable the TX rate control from register
PMA_CHx_REG_RX_TX2RX_PLPBK_RST_N	<string>	"FALSE","TRUE"	PMA parallel loopback module reset enable
PMA_CHx_REG_RX_TX2RX_PLPBK_RST_N_EN	<string>	"FALSE","TRUE"	enable the TX2RX parallel loop back reset control from register
PMA_CHx_REG_RX_TX2RX_PLPBK_EN	<string>	"FALSE","TRUE"	PMA parallel loopback enable
PMA_CHx_REG_TXCLK_SEL	<string>	"PLL","RXCLK"	enable the RX rate control from register
PMA_CHx_REG_RX_DATA_POLARITY	<string>	"NORMAL","REVERSE"	Receive data polarity inversion control register
PMA_CHx_REG_RX_EQ_INSERT	<string>	"FALSE","TRUE"	Insertion error on the rising edge
PMA_CHx_REG_UDP_CHK_EN	<string>	"FALSE","TRUE"	BIST UDP check enable
PMA_CHx_REG_PRBS_SEL	<string>	"PRBS7","PRBS15","PRBS23","PRBS31"	PMA RX PRBS mode selection

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_PRBS_CHK_EN	<string>	"FALSE","TRUE"	PMA RX PRBS check enable
PMA_CHx_REG_PRBS_CHK_WIDTH_SEL	<string>	"8BIT","10BIT","16BIT","20BIT"	PRBS check width selection
PMA_CHx_REG_BIST_CHK_PAT_SEL	<string>	"PRBS","CONSTANT"	BIST constant pattern or PRBS pattern selection
PMA_CHx_REG_LOAD_ERR_CNT	<string>	"DISABLE","ENABLE"	enable to load BIST checker error counter to 32bit status
PMA_CHx_REG_CHK_COUNTER_EN	<string>	"FALSE","TRUE"	Checking error detection count enable/clear control
PMA_CHx_REG_CDR_PROP_GAIN	<integer>	0 to 7	CDR proportional gain control
PMA_CHx_REG_CDR_PROP_TURBO_GAIN			Reserved
PMA_CHx_REG_CDR_INT_GAIN	<integer>	0 to 7	CDR integral gain control
PMA_CHx_REG_CDR_INT_TURBO_GAIN			Reserved
PMA_CHx_REG_CDR_INT_SAT_MAX			Reserved
PMA_CHx_REG_CDR_INT_SAT_MIN			Reserved
PMA_CHx_REG_CDR_INT_RST			Reserved
PMA_CHx_REG_CDR_INT_RST_OW			Reserved
PMA_CHx_REG_CDR_PROP_RST			Reserved
PMA_CHx_REG_CDR_PROP_RST_OW			Reserved
PMA_CHx_REG_CDR_LOCK_RST			Reserved
PMA_CHx_REG_CDR_LOCK_RST_OW			Reserved
PMA_CHx_REG_CDR_RX_PI_FORCE_SEL	<integer>	0 to 1	CDR RX PI control force value selection
PMA_CHx_REG_CDR_RX_PI_FORCE_D	<integer>	0 to 255	CDR RX PI control force data
PMA_CHx_REG_CDR_LOCK_TIMER			Reserved
PMA_CHx_REG_CDR_TURBO_MODE_TIMER			Reserved
PMA_CHx_REG_CDR_LOCK_VAL			Reserved
PMA_CHx_REG_CDR_LOCK_OW			Reserved
PMA_CHx_REG_CDR_INT_SAT_DET_EN			Reserved
PMA_CHx_REG_CDR_SAT_DET_STATUS_EN			Reserved
PMA_CHx_REG_CDR_SAT_DET_STATUS_RESET_EN			Reserved
PMA_CHx_REG_CDR_PI_CTRL_RST			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_CDR_PI_CTRL_RST_OW			Reserved
PMA_CHx_REG_CDR_SAT_DET_RST			Reserved
PMA_CHx_REG_CDR_SAT_DET_RST_OW			Reserved
PMA_CHx_REG_CDR_SAT_DET_STICKY_RST			Reserved
PMA_CHx_REG_CDR_SAT_DET_STICKY_RST_OW			Reserved
PMA_CHx_REG_CDR_SIGDET_STATUS_DIS			Reserved
PMA_CHx_REG_CDR_SAT_DET_TIMER			Reserved
PMA_CHx_REG_CDR_SAT_DET_STATUS_VA_L			Reserved
PMA_CHx_REG_CDR_SAT_DET_STATUS_OW			Reserved
PMA_CHx_REG_CDR_TURBO_MODE_EN			Reserved
PMA_CHx_REG_CDR_STATUS_RADDR_INIT			Reserved
PMA_CHx_REG_CDR_STATUS_FIFO_EN			Reserved
PMA_CHx_REG_PMA_TEST_SEL			Reserved
PMA_CHx_REG_OOB_COMWAKE_GAP_MIN			Reserved
PMA_CHx_REG_OOB_COMWAKE_GAP_MAX			Reserved
PMA_CHx_REG_RX_P_IBUF_IC_TX			Reserved
PMA_CHx_REG_COM_WAKE_STATUS_CLEA_R			Reserved
PMA_CHx_REG_COMINIT_STATUS_CLEAR			Reserved
PMA_CHx_REG_RX_SYNC_RST_N_EN			Reserved
PMA_CHx_REG_RX_SYNC_RST_N			Reserved
PMA_CHx_REG_RX_SATA_COMINIT_OW			Reserved
PMA_CHx_REG_RX_SATA_COMINIT		"	Reserved
PMA_CHx_REG_RX_SATA_COMWAKE_OW			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_RX_S ATA_COMWAKE			Reserved
PMA_CHx_REG_RX_D CC_DISABLE			Reserved
PMA_CHx_REG_TX_D CC_DISABLE			Reserved
PMA_CHx_REG_RX_S LIP_SEL_EN	<string>	"FALSE", "TRUE"	overwrite enable signal of reg_rx_slip_sel
PMA_CHx_REG_RX_S LIP_SEL	<integer>	0 to 15	slip output selection signal based on counter value of slip control
PMA_CHx_REG_RX_S LIP_EN	<string>	"FALSE", "TRUE"	CDR DEMUX Bypass enable
PMA_CHx_REG_RX_S IGDET_STATUS_SEL			Reserved
PMA_CHx_REG_RX_S IGDET_FSM_RST_N			Reserved
PMA_CHx_REG_RX_S IGDET_STATUS_OW	<string>	"DISABLE", "ENABLE"	RX signal detection status overwrite
PMA_CHx_REG_RX_S IGDET_STATUS	<string>	"FALSE", "TRUE"	Receive signal detection status
PMA_CHx_REG_RX_S IGDET_VTH			Reserved
PMA_CHx_REG_RX_S IGDET_GRM			Reserved
PMA_CHx_REG_RX_S IGDET_PULSE_EXT			Reserved
PMA_CHx_REG_RX_S IGDET_CH2_SEL			Reserved
PMA_CHx_REG_RX_S IGDET_CH2_CHK_WI NDOW			Reserved
PMA_CHx_REG_RX_S IGDET_CHK_WINDO W_EN			Reserved
PMA_CHx_REG_RX_S IGDET_NOSIG_COUN T_SETTING			Reserved
PMA_CHx_REG_RX_S IGDET_OOB_DET_CO UNT_VAL			Reserved
PMA_CHx_REG_SLIP_ FIFO_INV_EN			Reserved
PMA_CHx_REG_SLIP_ FIFO_INV			Reserved
PMA_CHx_REG_RX_S IGDET_4OOB_DET_SE L			Reserved
PMA_CHx_REG_RX_S IGDET_IC_I			Reserved
PMA_CHx_REG_RX_O OB_DETECTOR_RESE T_N_OW			Reserved
PMA_CHx_REG_RX_O OB_DETECTOR_RESE T_N			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_RX_OB_DETECTOR_PD_OW			Reserved
PMA_CHx_REG_RX_OB_DETECTOR_PD			Reserved
PMA_CHx_REG_RX_TERM_CM_CTRL			Reserved
PMA_CHx_REG_TX_PD	<string>	"ON", "OFF"	TX power control
PMA_CHx_REG_TX_PD_OW	<string>	"DISABLE", "ENABLE"	enable tx power down control from register
PMA_CHx_REG_TXCLKPATH_PD	<string>	"ON", "OFF"	TX clock path power control
PMA_CHx_REG_TXCLKPATH_PD_OW	<string>	"DISABLE", "ENABLE"	enable the clock path power down control from register
PMA_CHx_REG_TX_BEACON_TIMER_SEL			Reserved
PMA_CHx_REG_TX_RXDET_REQ_OW			Reserved
PMA_CHx_REG_TX_RXDET_REQ			Reserved
PMA_CHx_REG_TX_BEACON_EN_OW			Reserved
PMA_CHx_REG_TX_BEACON_EN			Reserved
PMA_CHx_REG_TX_EI_EN_OW	<string>	"DISABLE", "ENABLE"	Electric Ideal enable register overwrite
PMA_CHx_REG_TX_EI_EN	<string>	"FALSE", "TRUE"	register Electric Ideal enable
PMA_CHx_REG_TX_RXES_CAL_EN			Reserved
PMA_CHx_REG_TX_RXES_CAL			Reserved
PMA_CHx_REG_TX_BIAS_CAL_EN			Reserved
PMA_CHx_REG_TX_BIAS_CTRL			Reserved
PMA_CHx_REG_TX_RXDET_TIMER_SEL			Reserved
PMA_CHx_REG_TX_SYNC_OW			Reserved
PMA_CHx_REG_TX_SYNC			Reserved
PMA_CHx_REG_TX_PD_POST	<string>	"ON", "OFF"	Post-scaler de-emphasis enable
PMA_CHx_REG_TX_PD_POST_OW	<string>	"DISABLE", "ENABLE"	overwrite of reg_tx_pd_post
PMA_CHx_REG_TX_RESET_N_OW	<string>	"DISABLE", "ENABLE"	TX reset n register overwrite
PMA_CHx_REG_TX_RESET_N	<string>	"FALSE", "TRUE"	TX reset enable
PMA_CHx_REG_TX_DCC_RESET_N_OW			Reserved
PMA_CHx_REG_TX_DCC_RESET_N			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_TX_B_USWIDTH_OW	<string>	"DISABLE", "ENABLE"	TX buswidth register overwrite
PMA_CHx_REG_TX_B_USWIDTH	<string>	"8BIT", "10BIT", "16BIT", "20BIT"	TX buswidth control
PMA_CHx_REG_PLL_READY_OW	<string>	"DISABLE", "ENABLE"	TX pll ready register overwrite
PMA_CHx_REG_PLL_READY	<string>	"FALSE", "TRUE"	register pll ready enable
PMA_CHx_REG_TX_P_CLK_SW_OW			Reserved
PMA_CHx_REG_TX_P_CLK_SW			Reserved
PMA_CHx_REG_EI_PC_LK_DELAY_SEL			Reserved
PMA_CHx_REG_TX_D_RV01_DAC0	<integer>	0 to 63	Post-scaler de-emphasis configuration value
PMA_CHx_REG_TX_D_RV01_DAC1			Reserved
PMA_CHx_REG_TX_D_RV01_DAC2			Reserved
PMA_CHx_REG_TX_D_RV00_DAC0	<integer>	0 to 63	main cursor TX bias ctrl code setting
PMA_CHx_REG_TX_D_RV00_DAC1			Reserved
PMA_CHx_REG_TX_D_RV00_DAC2			Reserved
PMA_CHx_REG_TX_A_MP0	<integer>	0 to 63	TX full swing bias register control
PMA_CHx_REG_TX_A_MP1			Reserved
PMA_CHx_REG_TX_A_MP2			Reserved
PMA_CHx_REG_TX_A_MP3			Reserved
PMA_CHx_REG_TX_A_MP4			Reserved
PMA_CHx_REG_TX_MARGIN			Reserved
PMA_CHx_REG_TX_MARGIN_OW			Reserved
PMA_CHx_REG_TX_D_EEMP			Reserved
PMA_CHx_REG_TX_D_EEMP_OW			Reserved
PMA_CHx_REG_TX_S_WING	<string>	"FALSE", "TRUE"	tx_swing control register
PMA_CHx_REG_TX_S_WING_OW	<string>	"DISABLE", "ENABLE"	overwrite signal of tx_swing
PMA_CHx_REG_TX_R_XDET_THRESHOLD			Reserved
PMA_CHx_REG_TX_B_EACON_OSC_CTRL			Reserved
PMA_CHx_REG_TX_P_REDVR_DAC			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_TX_P REDRV_CM_CTRL			Reserved
PMA_CHx_REG_TX_T X2RX_SLPBACK_EN	<string>	"FALSE", "TRUE"	PMA serial loopback enable
PMA_CHx_REG_TX_P CLK_EDGE_SEL	<string>	"POS_EDGE", "NEG_EDGE"	txpclk edge selection
PMA_CHx_REG_TX_R XDET_STATUS_OW	<string>	"DISABLE", "ENABLE"	register RX detect status force enable
PMA_CHx_REG_TX_R XDET_STATUS	<string>	"FALSE", "TRUE"	RX detect status force value
PMA_CHx_REG_TX_P RBS_GEN_EN	<string>	"FALSE", "TRUE"	PMA TX PRBS generator enable
PMA_CHx_REG_TX_P RBS_GEN_WIDTH_SE L	<string>	"8BIT", "10BIT", "16BIT", "20BIT"	PMA TX PRBS bit width selection
PMA_CHx_REG_TX_P RBS_SEL	<string>	"PRBS7", "PRBS15", "PRBS23", "PRBS31"	PMA TX PRBS mode selection
PMA_CHx_REG_TX_U DP_DATA			Reserved
PMA_CHx_REG_TX_F IFO_RST_N			Reserved
PMA_CHx_REG_TX_F IFO_WP_CTRL			Reserved
PMA_CHx_REG_TX_F IFO_EN	<string>	"FALSE", "TRUE"	Register configuration TX_fifo_en enable
PMA_CHx_REG_TX_D ATA_MUX_SEL	<integer>	0 to 3	Data selection control signal
PMA_CHx_REG_TX_E RR_INSERT	<string>	"FALSE", "TRUE"	PRBS error code injection
PMA_CHx_REG_TX_S ATA_EN			Reserved
PMA_CHx_REG_RATE _CHANGE_TXPCLK_ ON_OW			Reserved
PMA_CHx_REG_RATE _CHANGE_TXPCLK_ ON			Reserved
PMA_CHx_REG_TX_P ULLUP_DAC0			Reserved
PMA_CHx_REG_TX_P ULLUP_DAC1			Reserved
PMA_CHx_REG_TX_P ULLUP_DAC2			Reserved
PMA_CHx_REG_TX_P ULLUP_DAC3			Reserved
PMA_CHx_REG_TX_O OB_DELAY_SEL			Reserved
PMA_CHx_REG_TX_P OLARITY	<string>	"NORMAL", "REVERSE"	PMA TX polarity inversion
PMA_CHx_REG_TX_S LPBK_AMP			Reserved
PMA_CHx_REG_TX_L S_MODE_EN			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_TX_J TAG_MODE_EN_OW			Reserved
PMA_CHx_REG_TX_J TAG_MODE_EN			Reserved
PMA_CHx_REG_RX_J TAG_MODE_EN_OW			Reserved
PMA_CHx_REG_RX_J TAG_MODE_EN			Reserved
PMA_CHx_REG_RX_J TAG_OE			Reserved
PMA_CHx_REG_RX_A CJTAG_VHYSTSE			Reserved
PMA_CHx_REG_TX_F BCLK_FAR_EN			Reserved
PMA_CHx_REG_RX_T ERM_MODE_CTRL	<integer>	0 to 7	Coupling mode selection
PMA_CHx_REG_PLPB K_TXPCLK_EN	<string>	"FALSE", "TRUE"	txpclk enable register in parallel loopback mode
PMA_CHx_REG_TX_6 09_600			Reserved
PMA_CHx_REG_RX_C DR_617_610			Reserved
PMA_CHx_REG_RX_C DR_623_618			Reserved
PMA_CHx_REG_RX_6 31_624			Reserved
PMA_CHx_REG_RX_6 39_632			Reserved
PMA_CHx_REG_RX_6 47_640			Reserved
PMA_CHx_REG_RX_6 55_648			Reserved
PMA_CHx_REG_RX_6 59_656			Reserved
PMA_CHx_CFG_LANE _POWERUP	<string>	"ON", "OFF"	Lane power-up configuration
PMA_CHx_CFG_PMA_ POR_N	<string>	"FALSE", "TRUE"	Lane reset configuration
PMA_CHx_CFG_RX_L ANE_POWERUP	<string>	"ON", "OFF"	Lane's RX channel power-up configuration
PMA_CHx_CFG_RX_P MA_RSTN	<string>	"FALSE", "TRUE"	Lane's RX channel reset configuration
PMA_CHx_CFG_TX_L ANE_POWERUP	<string>	"ON", "OFF"	Lane's TX channel power-up configuration
PMA_CHx_CFG_TX_P MA_RSTN	<string>	"FALSE", "TRUE"	Lane's TX channel reset configuration
PMA_CHx_REG_RESE RVED_48_45			Reserved
PMA_CHx_REG_RESE RVED_69			Reserved
PMA_CHx_REG_RESE RVED_77_76			Reserved
PMA_CHx_REG_RESE RVED_171_164			Reserved
PMA_CHx_REG_RESE RVED_175_172			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_CHx_REG_RESE_RVED_190			Reserved
PMA_CHx_REG_RESE_RVED_233_232			Reserved
PMA_CHx_REG_RESE_RVED_235_234			Reserved
PMA_CHx_REG_RESE_RVED_241_240			Reserved
PMA_CHx_REG_RESE_RVED_285_283			Reserved
PMA_CHx_REG_RESE_RVED_286			Reserved
PMA_CHx_REG_RESE_RVED_295			Reserved
PMA_CHx_REG_RESE_RVED_298			Reserved
PMA_CHx_REG_RESE_RVED_332_325			Reserved
PMA_CHx_REG_RESE_RVED_340_333			Reserved
PMA_CHx_REG_RESE_RVED_348_341			Reserved
PMA_CHx_REG_RESE_RVED_354_349			Reserved
PMA_CHx_REG_RESE_RVED_373			Reserved
PMA_CHx_REG_RESE_RVED_376			Reserved
PMA_CHx_REG_RESE_RVED_452			Reserved
PMA_CHx_REG_RESE_RVED_502_499			Reserved
PMA_CHx_REG_RESE_RVED_506_505			Reserved
PMA_CHx_REG_RESE_RVED_550_549			Reserved
PMA_CHx_REG_RESE_RVED_556_552			Reserved
PMA_PLLy_REG_PLL_POWERDOWN_OW	<string>	"DISABLE", "ENABLE"	PLL power control enable
PMA_PLLy_REG_PLL_POWERDOWN	<string>	"ON", "OFF"	PLL power control
PMA_PLLy_REG_PLL_RESET_N_OW	<string>	"DISABLE", "ENABLE"	PLL reset control enable
PMA_PLLy_REG_PLL_RESET_N	<string>	"FALSE", "TRUE"	PLL reset control
PMA_PLLy_REG_PLL_READY_OW	<string>	"DISABLE", "ENABLE"	PLL ready status control enable
PMA_PLLy_REG_PLL_READY	<string>	"FALSE", "TRUE"	PLL ready status control
PMA_PLLy_REG_LAN_E_SYNC_OW			Reserved
PMA_PLLy_REG_LAN_E_SYNC			Reserved
PMA_PLLy_REG_LOC_KDET_REPEAT			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_PLLy_REG_RES_CAL_I_CODE_PMA			Reserved
PMA_PLLy_REG_RES_CAL_RESET_N_OW			Reserved
PMA_PLLy_REG_RES_CAL_RESET_N			Reserved
PMA_PLLy_REG_RES_CAL_DONE_OW			Reserved
PMA_PLLy_REG_RES_CAL_DONE			Reserved
PMA_PLLy_REG_RES_CAL_CODE_OW			Reserved
PMA_PLLy_REG_LDO_VREF_SEL			Reserved
PMA_PLLy_REG_BIAS_VCOREP_C			Reserved
PMA_PLLy_REG_RES_CAL_I_CODE			Reserved
PMA_PLLy_REG_RES_CAL_ONCHIP_SMALL_OW			Reserved
PMA_PLLy_REG_RES_CAL_ONCHIP_SMALL			Reserved
PMA_PLLy_REG_JTAG_OE			Reserved
PMA_PLLy_REG_JTAG_AC_MODE			Reserved
PMA_PLLy_REG_JTAG_VHYSTSEL			Reserved
PMA_PLLy_REG_PLL_LOCKDET_EN_OW			Reserved
PMA_PLLy_REG_PLL_LOCKDET_EN			Reserved
PMA_PLLy_REG_PLL_LOCKDET_RESET_N_OW			Reserved
PMA_PLLy_REG_PLL_LOCKDET_RESET_N			Reserved
PMA_PLLy_REG_PLL_LOCKED_OW			Reserved
PMA_PLLy_REG_PLL_LOCKED			Reserved
PMA_PLLy_REG_PLL_LOCKED_STICKY_CLEAR			Reserved
PMA_PLLy_REG_PLL_UNLOCKED_STICKY_CLEAR			Reserved
PMA_PLLy_REG_NOF_BCLK_STICKY_CLEA_R			Reserved
PMA_PLLy_REG_PLL_LOCKDET_REFCT			Reserved
PMA_PLLy_REG_PLL_LOCKDET_FBCT			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_PLLy_REG_PLL_LOCKDET_LOCKCT			Reserved
PMA_PLLy_REG_PLL_LOCKDET_ITER			Reserved
PMA_PLLy_REG_PLL_UNLOCKDET_ITER			Reserved
PMA_PLLy_REG_PD_VCO			Reserved
PMA_PLLy_REG_FBC_LK_TEST_EN			Reserved
PMA_PLLy_REG_REF_CLK_TEST_EN			Reserved
PMA_PLLy_REG_TES_T_SEL			Reserved
PMA_PLLy_REG_TES_T_V_EN			Reserved
PMA_PLLy_REG_TES_T_SIG_HALF_EN			Reserved
PMA_PLLy_REG_TES_T_FSM			Reserved
PMA_PLLy_REG_REF_CLK_OUT_PD	<string>	"ON", "OFF"	Reference clock output buffer switch control
PMA_PLLy_REG_BGR_STARTUP_EN			Reserved
PMA_PLLy_REG_BGR_STARTUP			Reserved
PMA_PLLy_REG_PD_BGR			Reserved
PMA_PLLy_REG_REF_CLK_TERM_VCM_EN	<string>	"FALSE", "TRUE"	refclk terminal common-mode voltage selection register
PMA_PLLy_REG_FBDI_VA_5_EN	<string>	"FALSE", "TRUE"	feedback divider A div-5 enable
PMA_PLLy_REG_FBDI_VB	<integer>	0 to 2	feedback divider B
PMA_PLLy_REG_RES_ET_N_PFDQP_OW			Reserved
PMA_PLLy_REG_RES_ET_N_PFDQP			Reserved
PMA_PLLy_REG_QPC_URRENT			Reserved
PMA_PLLy_REG_VC_FORCE_EN			Reserved
PMA_PLLy_REG_VCR_ESET_C_RING			Reserved
PMA_PLLy_REG_LPF_R_C			Reserved
PMA_PLLy_REG_LPF_TR_C			Reserved
PMA_PLLy_REG_PD_BIAS			Reserved
PMA_PLLy_REGICTR_L_PLL			Reserved
PMA_PLLy_REG_BIAS_QP			Reserved
PMA_PLLy_REG_BIAS_LANE_SYNC			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_PLLy_REG_BIAS_CLKBUFS1			Reserved
PMA_PLLy_REG_TXP_CLK_SEL			Reserved
PMA_PLLy_REG_BIAS_CLKBUFS3			Reserved
PMA_PLLy_REG_LAN_E_SYNC_EN			Reserved
PMA_PLLy_REG_LAN_E_SYNC_EN_OW			Reserved
PMA_PLLy_REG_BIAS_D2S			Reserved
PMA_PLLy_REG_BIAS_REFD2S_C			Reserved
PMA_PLLy_REG_BIAS_VCRST_C			Reserved
PMA_PLLy_REG_BIAS_REFBUF_C			Reserved
PMA_PLLy_REG_CLK_BUFS1_C			Reserved
PMA_PLLy_REG_CLK_BUFS2_C			Reserved
PMA_PLLy_REG_CLK_BUFS3_C			Reserved
PMA_PLLy_REG_CLK_BUFS4_C			Reserved
PMA_PLLy_REG_PLL_REFCLK_CML_SEL	<integer>	0 to 3	Reference clock pin selection
PMA_PLLy_REG_REF_CLK_SEL	<string>	"FALSE", "TRUE"	Reference clock source selection
PMA_PLLy_REG_RES_CAL_R_CODE_SIGN			Reserved
PMA_PLLy_REG_PLL_UNLOCKED_OW	<string>	"DISABLE", "ENABLE"	PLL lock status control enable
PMA_PLLy_REG_PLL_UNLOCKED	<string>	"FALSE", "TRUE"	PLL lock status control
PMA_PLLy_REG_PLL_LOCKDET_MODE			Reserved
PMA_PLLy_REG_PLL_CLKBUF_PD_LEFT			Reserved
PMA_PLLy_REG_PLL_CLKBUF_PD_RIGHT			Reserved
PMA_PLLy_REG_RES_CAL_EN			Reserved
PMA_PLLy_REG_RES_CAL_I_CODE_VAL			Reserved
PMA_PLLy_REG_RES_CAL_I_CODE_OW			Reserved
PMA_PLLy_REG_RES_CAL_ITER_VALID_SEL			Reserved
PMA_PLLy_REG_RES_CAL_WAIT_SEL			Reserved
PMA_PLLy_REG_I_CTL_MAX			Reserved

Parameter Name	Parameter Type	Valid Values	Description
PMA_PLLy_REG_I_CT_RL_MIN			Reserved
PMA_PLLy_REG_RESERVED_167_160	<integer>	0 to 255	[7:1]:Reserved [0]: pll_refclk_div2_en_o, PLL reference clock divide-by-two enable
PMA_PLLy_REG_RESERVED_175_168			Reserved
PMA_PLLy_REG_RESERVED_183_176			Reserved
PMA_PLLy_REG_RESERVED_191_184			Reserved
PARM_CFG_HSST_RSTN	<string>	"FALSE", "TRUE"	Configuring PLL's cfg_hsst_rstn
PARM_PLLy_POWERUP	<string>	"ON", "OFF"	Configuring PLL's cfg_pllpowerup
PARM_PLLy_RSTN	<string>	"FALSE", "TRUE"	Configuring PLL's cfg_pll_rstn
PARM_CFG_RSTN	<string>	"FALSE", "TRUE"	Reserved
PARM_GRSN_DIS	<string>	"FALSE", "TRUE"	Reserved

Notes:

1. Each signal named "CHx" of the above parameters corresponds to a channel:

CH0: For Channel 0

CH1: For Channel 1

CH2: For Channel 2

CH3: For Channel 3

2. Each signal named "PLL" of the above parameters corresponds to a PLL:

PLL0: For PLL0

PLL1: For PLL1

### 10.5.5 Instantiation template

```
GTP_HSST_E2 #(
    .PCS_CH0_BYPASS_WORD_ALIGN("FALSE"),
    .PCS_CH0_BYPASS_DENC("FALSE"),
    .PCS_CH0_BYPASS_BONDING("FALSE"),
    .PCS_CH0_BYPASS_CTC("FALSE"),
    .PCS_CH0_BYPASS_GEAR("FALSE"),
    .PCS_CH0_BYPASS_BRIDGE("FALSE"),
    .PCS_CH0_BYPASS_BRIDGE_FIFO("FALSE"),
    .PCS_CH0_DATA_MODE("X8"),
    .PCS_CH0_RX_POLARITY_INV("DELAY"),
    .PCS_CH0_ALIGN_MODE("1GB"),
    .PCS_CH0_SAMP_16B("X16"),
    .PCS_CH0_FARLP_PWR_REDUCTION("FALSE"),
```

```
.PCS_CH0_COMMREG(0),  
.PCS_CH0_COMMASK(0),  
.PCS_CH0_CEB_MODE("10GB"),  
.PCS_CH0_CTC_MODE("1SKIP"),  
.PCS_CH0_A_REG(0),  
.PCS_CH0_GE_AUTO_EN("FALSE"),  
.PCS_CH0_SKIP_REG0(0),  
.PCS_CH0_SKIP_REG1(0),  
.PCS_CH0_SKIP_REG2(0),  
.PCS_CH0_SKIP_REG3(0),  
.PCS_CH0_DEC_DUAL("FALSE"),  
.PCS_CH0_SPLIT("FALSE"),  
.PCS_CH0_FIFOFLAG_CTC("FALSE"),  
.PCS_CH0_COMMODE("COMMA_PATTERN"),  
.PCS_CH0_ERRDETECT_SILENCE("FALSE"),  
.PCS_CH0_PMA_RCLK_POLINV("PMA_RCLK"),  
.PCS_CH0_PCS_RCLK_SEL("PMA_RCLK"),  
.PCS_CH0_CB_RCLK_SEL("PMA_RCLK"),  
.PCS_CH0_AFTER_CTC_RCLK_SEL("PMA_RCLK"),  
.PCS_CH0_AFTER_CTC_RCLK_SEL_1("PMA_RCLK"),  
.PCS_CH0_RCLK_POLINV("RCLK"),  
.PCS_CH0_BRIDGE_RCLK_SEL("PMA_RCLK"),  
.PCS_CH0_PCS_RCLK_EN("FALSE"),  
.PCS_CH0_CB_RCLK_EN("FALSE"),  
.PCS_CH0_AFTER_CTC_RCLK_EN("FALSE"),  
.PCS_CH0_AFTER_CTC_RCLK_EN_GB("FALSE"),  
.PCS_CH0_AFTER_CTC_RCLK_EN_GB_1("FALSE"),  
.PCS_CH0_PCS_RX_RSTN("FALSE"),  
.PCS_CH0_SLAVE("MASTER"),  
.PCS_CH0_PCIE_SLAVE("MASTER"),  
.PCS_CH0_RX_64B66B_67B("NORMAL"),  
.PCS_CH0_RX_BRIDGE_CLK_POLINV("RX_BRIDGE_CLK"),  
.PCS_CH0_PCS_CB_RSTN("FALSE"),  
.PCS_CH0_TX_BRIDGE_GEAR_SEL("FALSE"),
```

```
.PCS_CH0_TX_BYPASS_BRIDGE_UINT("FALSE"),
.PCS_CH0_TX_BYPASS_BRIDGE_FIFO("FALSE"),
.PCS_CH0_TX_BYPASS_GEAR("FALSE"),
.PCS_CH0_TX_BYPASS_ENC("FALSE"),
.PCS_CH0_TX_BYPASS_BIT_SLIP("FALSE"),
.PCS_CH0_TX_GEAR_SPLIT("FALSE"),
.PCS_CH0_TX_DRIVE_REG_MODE("NO_CHANGE"),
.PCS_CH0_TX_BIT_SLIP_CYCLES(0),
.PCS_CH0_INT_TX_MASK_0("FALSE"),
.PCS_CH0_INT_TX_MASK_1("FALSE"),
.PCS_CH0_INT_TX_MASK_2("FALSE"),
.PCS_CH0_INT_TX_CLR_0("FALSE"),
.PCS_CH0_INT_TX_CLR_1("FALSE"),
.PCS_CH0_INT_TX_CLR_2("FALSE"),
.PCS_CH0_TX_PMA_TCLK_POLINV("PMA_TCLK"),
.PCS_CH0_TX_PCS_CLK_EN_SEL("FALSE"),
.PCS_CH0_TX_BRIDGE_TCLK_SEL("TCLK"),
.PCS_CH0_TX_TCLK_POLINV("TCLK"),
.PCS_CH0_TX_PCS_TCLK_SEL("PMA_TCLK"),
.PCS_CH0_TX_PCS_TX_RSTN("FALSE"),
.PCS_CH0_TX_SLAVE("MASTER"),
.PCS_CH0_TX_GEAR_TCLK_EN_SEL("FALSE"),
.PCS_CH0_DATA_WIDTH_MODE("X20"),
.PCS_CH0_TX_64B66B_67B("NORMAL"),
.PCS_CH0_TX_GEAR_TCLK_SEL("PMA_TCLK"),
.PCS_CH0_TX_TCLK2FABRIC_SEL("FALSE"),
.PCS_CH0_TX_OUTZZ("FALSE"),
.PCS_CH0_ENC_DUAL("FALSE"),
.PCS_CH0_TX_BITSLIP_DATA_MODE("X10"),
.PCS_CH0_TX_BRIDGE_CLK_POLINV("TX_BRIDGE_CLK"),
.PCS_CH0_COMM_REG1(0),
.PCS_CH0_RAPID_IMAX(0),
.PCS_CH0_RAPID_VMIN_1(0),
.PCS_CH0_RAPID_VMIN_2(0),
```

```
.PCS_CH0_RX_PRBS_MODE("DISABLE"),
.PCS_CH0_RX_ERRCNT_CLR("FALSE"),
.PCS_CH0_RX_PRBS_ERR_LPBK("FALSE"),
.PCS_CH0_TX_PRBS_MODE("DISABLE"),
.PCS_CH0_TX_INSERT_ER("FALSE"),
.PCS_CH0_ENABLE_PRBS_GEN("FALSE"),
.PCS_CH0_ERR_CNT(0),
.PCS_CH0_DEFAULT_RADDR(0),
.PCS_CH0_MASTER_CHECK_OFFSET(0),
.PCS_CH0_DELAY_SET(0),
.PCS_CH0_SEACH_OFFSET("20BIT"),
.PCS_CH0_CEB_RAPIDL_S_MMAX(0),
.PCS_CH0_CTC_AFULL(0),
.PCS_CH0_CTC_AEMPTY(0),
.PCS_CH0_CTC_CONTI_SKP_SET(0),
.PCS_CH0_FAR_LOOP("FALSE"),
.PCS_CH0_NEAR_LOOP("FALSE"),
.PCS_CH0_REG_TX2RX_PLOOP_EN("FALSE"),
.PCS_CH0_REG_TX2RX_SLOOP_EN("FALSE"),
.PCS_CH0_REG_RX2TX_PLOOP_EN("FALSE"),
.PCS_CH0_INT_RX_MASK_0("FALSE"),
.PCS_CH0_INT_RX_MASK_1("FALSE"),
.PCS_CH0_INT_RX_MASK_2("FALSE"),
.PCS_CH0_INT_RX_MASK_3("FALSE"),
.PCS_CH0_INT_RX_MASK_4("FALSE"),
.PCS_CH0_INT_RX_MASK_5("FALSE"),
.PCS_CH0_INT_RX_MASK_6("FALSE"),
.PCS_CH0_INT_RX_MASK_7("FALSE"),
.PCS_CH0_INT_RX_CLR_0("FALSE"),
.PCS_CH0_INT_RX_CLR_1("FALSE"),
.PCS_CH0_INT_RX_CLR_2("FALSE"),
.PCS_CH0_INT_RX_CLR_3("FALSE"),
.PCS_CH0_INT_RX_CLR_4("FALSE"),
.PCS_CH0_INT_RX_CLR_5("FALSE"),
```

```
.PCS_CH0_INT_RX_CLR_6("FALSE"),
.PCS_CH0_INT_RX_CLR_7("FALSE"),
.PCS_CH0_CA_RSTN_RX("FALSE"),
.PCS_CH0_CA_DYN_DLY_EN_RX("FALSE"),
.PCS_CH0_CA_DYN_DLY_SEL_RX("FALSE"),
.PCS_CH0_CA_RX(0),
.PCS_CH0_CA_RSTN_TX("FALSE"),
.PCS_CH0_CA_DYN_DLY_EN_TX("FALSE"),
.PCS_CH0_CA_DYN_DLY_SEL_TX("FALSE"),
.PCS_CH0_CA_TX(0),
.PCS_CH0_RXPRBS_PWR_REDUCTION("NORMAL"),
.PCS_CH0_WDALIGN_PWR_REDUCTION("NORMAL"),
.PCS_CH0_RXDEC_PWR_REDUCTION("NORMAL"),
.PCS_CH0_RXCB_PWR_REDUCTION("NORMAL"),
.PCS_CH0_RXCTC_PWR_REDUCTION("NORMAL"),
.PCS_CH0_RXGEAR_PWR_REDUCTION("NORMAL"),
.PCS_CH0_RXBRG_PWR_REDUCTION("NORMAL"),
.PCS_CH0_RXTEST_PWR_REDUCTION("NORMAL"),
.PCS_CH0_TXBRG_PWR_REDUCTION("NORMAL"),
.PCS_CH0_TXGEAR_PWR_REDUCTION("NORMAL"),
.PCS_CH0_TXENC_PWR_REDUCTION("NORMAL"),
.PCS_CH0_TXBSLP_PWR_REDUCTION("NORMAL"),
.PCS_CH0_TXPRBS_PWR_REDUCTION("NORMAL"),
.PCS_CH0_TXBRG_FULL_CHK_EN("FALSE"),
.PCS_CH0_TXBRG_EMPTY_CHK_EN("FALSE"),
.PCS_CH0_RXBRG_FULL_CHK_EN("FALSE"),
.PCS_CH0_RXBRG_EMPTY_CHK_EN("FALSE"),
.PCS_CH0_CTC_FULL_CHK_EN("TRUE"),
.PCS_CH0_CTC_EMPTY_CHK_EN("TRUE"),
.PCS_CH0_CEB_FULL_CHK_EN("FALSE"),
.PCS_CH0_CEB_EMPTY_CHK_EN("FALSE"),
.PCS_CH0_FLP_FULL_CHK_EN("TRUE"),
.PCS_CH0_FLP_EMPTY_CHK_EN("TRUE"),
.PCS_CH1_BYPASS_WORD_ALIGN("FALSE"),
```

```
.PCS_CH1_BYPASS_DENC("FALSE"),
.PCS_CH1_BYPASS_BONDING("FALSE"),
.PCS_CH1_BYPASS_CTC("FALSE"),
.PCS_CH1_BYPASS_GEAR("FALSE"),
.PCS_CH1_BYPASS_BRIDGE("FALSE"),
.PCS_CH1_BYPASS_BRIDGE_FIFO("FALSE"),
.PCS_CH1_DATA_MODE("X8"),
.PCS_CH1_RX_POLARITY_INV("DELAY"),
.PCS_CH1_ALIGN_MODE("1GB"),
.PCS_CH1_SAMP_16B("X16"),
.PCS_CH1_FARLP_PWR_REDUCTION("FALSE"),
.PCS_CH1_COMM_A_REG0(0),
.PCS_CH1_COMM_A_MASK(0),
.PCS_CH1_CEB_MODE("10GB"),
.PCS_CH1_CTC_MODE("1SKIP"),
.PCS_CH1_A_REG(0),
.PCS_CH1_GE_AUTO_EN("FALSE"),
.PCS_CH1_SKIP_REG0(0),
.PCS_CH1_SKIP_REG1(0),
.PCS_CH1_SKIP_REG2(0),
.PCS_CH1_SKIP_REG3(0),
.PCS_CH1_DEC_DUAL("FALSE"),
.PCS_CH1_SPLIT("FALSE"),
.PCS_CH1_FIFOFLAG_CTC("FALSE"),
.PCS_CH1_COMM_A_DET_MODE("COMM_A_PATTERN"),
.PCS_CH1_ERRDETECT_SILENCE("FALSE"),
.PCS_CH1_PMA_RCLK_POLINV("PMA_RCLK"),
.PCS_CH1_PCS_RCLK_SEL("PMA_RCLK"),
.PCS_CH1_CB_RCLK_SEL("PMA_RCLK"),
.PCS_CH1_AFTER_CTC_RCLK_SEL("PMA_RCLK"),
.PCS_CH1_AFTER_CTC_RCLK_SEL_1("PMA_RCLK"),
.PCS_CH1_RCLK_POLINV("RCLK"),
.PCS_CH1_BRIDGE_RCLK_SEL("PMA_RCLK"),
.PCS_CH1_PCS_RCLK_EN("FALSE"),
```

```
.PCS_CH1_CB_RCLK_EN("FALSE"),
.PCS_CH1_AFTER_CTC_RCLK_EN("FALSE"),
.PCS_CH1_AFTER_CTC_RCLK_EN_GB("FALSE"),
.PCS_CH1_AFTER_CTC_RCLK_EN_GB_1("FALSE"),
.PCS_CH1_PCS_RX_RSTN("FALSE"),
.PCS_CH1_SLAVE("MASTER"),
.PCS_CH1_PCIE_SLAVE("MASTER"),
.PCS_CH1_RX_64B66B_67B("NORMAL"),
.PCS_CH1_RX_BRIDGE_CLK_POLINV("RX_BRIDGE_CLK"),
.PCS_CH1_PCS_CB_RSTN("FALSE"),
.PCS_CH1_TX_BRIDGE_GEAR_SEL("FALSE"),
.PCS_CH1_TX_BYPASS_BRIDGE_UINT("FALSE"),
.PCS_CH1_TX_BYPASS_BRIDGE_FIFO("FALSE"),
.PCS_CH1_TX_BYPASS_GEAR("FALSE"),
.PCS_CH1_TX_BYPASS_ENC("FALSE"),
.PCS_CH1_TX_BYPASS_BIT_SLIP("FALSE"),
.PCS_CH1_TX_GEAR_SPLIT("FALSE"),
.PCS_CH1_TX_DRIVE_REG_MODE("NO_CHANGE"),
.PCS_CH1_TX_BIT_SLIP_CYCLES(0),
.PCS_CH1_INT_TX_MASK_0("FALSE"),
.PCS_CH1_INT_TX_MASK_1("FALSE"),
.PCS_CH1_INT_TX_MASK_2("FALSE"),
.PCS_CH1_INT_TX_CLR_0("FALSE"),
.PCS_CH1_INT_TX_CLR_1("FALSE"),
.PCS_CH1_INT_TX_CLR_2("FALSE"),
.PCS_CH1_TX_PMA_TCLK_POLINV("PMA_TCLK"),
.PCS_CH1_TX_PCS_CLK_EN_SEL("FALSE"),
.PCS_CH1_TX_BRIDGE_TCLK_SEL("TCLK"),
.PCS_CH1_TX_TCLK_POLINV("TCLK"),
.PCS_CH1_TX_PCS_TCLK_SEL("PMA_TCLK"),
.PCS_CH1_TX_PCS_TX_RSTN("FALSE"),
.PCS_CH1_TX_SLAVE("MASTER"),
.PCS_CH1_TX_GEAR_TCLK_EN_SEL("FALSE"),
.PCS_CH1_DATA_WIDTH_MODE("X20"),
```

```
.PCS_CH1_TX_64B66B_67B("NORMAL"),
.PCS_CH1_TX_GEAR_TCLK_SEL("PMA_TCLK"),
.PCS_CH1_TX_TCLK2FABRIC_SEL("FALSE"),
.PCS_CH1_TX_OUTZZ("FALSE"),
.PCS_CH1_ENC_DUAL("FALSE"),
.PCS_CH1_TX_BITSLIP_DATA_MODE("X10"),
.PCS_CH1_TX_BRIDGE_CLK_POLINV("TX_BRIDGE_CLK"),
.PCS_CH1_COMM_A_REG1(0),
.PCS_CH1_RAPID_IMAX(0),
.PCS_CH1_RAPID_VMIN_1(0),
.PCS_CH1_RAPID_VMIN_2(0),
.PCS_CH1_RX_PRBS_MODE("DISABLE"),
.PCS_CH1_RX_ERRCNT_CLR("FALSE"),
.PCS_CH1_RX_PRBS_ERR_LPBK("FALSE"),
.PCS_CH1_TX_PRBS_MODE("DISABLE"),
.PCS_CH1_TX_INSERT_ER("FALSE"),
.PCS_CH1_ENABLE_PRBS_GEN("FALSE"),
.PCS_CH1_ERR_CNT(0),
.PCS_CH1_DEFAULT_RADDR(0),
.PCS_CH1_MASTER_CHECK_OFFSET(0),
.PCS_CH1_DELAY_SET(0),
.PCS_CH1_SEACH_OFFSET("20BIT"),
.PCS_CH1_CEB_RAPIDL_S_MMAX(0),
.PCS_CH1_CTC_AFULL(0),
.PCS_CH1_CTC_AEMPTY(0),
.PCS_CH1_CTC_CONTI_SKP_SET(0),
.PCS_CH1_FAR_LOOP("FALSE"),
.PCS_CH1_NEAR_LOOP("FALSE"),
.PCS_CH1_REG_TX2RX_PLOOP_EN("FALSE"),
.PCS_CH1_REG_TX2RX_SLOOP_EN("FALSE"),
.PCS_CH1_REG_RX2TX_PLOOP_EN("FALSE"),
.PCS_CH1_INT_RX_MASK_0("FALSE"),
.PCS_CH1_INT_RX_MASK_1("FALSE"),
.PCS_CH1_INT_RX_MASK_2("FALSE"),
```

```
.PCS_CH1_INT_RX_MASK_3("FALSE"),
.PCS_CH1_INT_RX_MASK_4("FALSE"),
.PCS_CH1_INT_RX_MASK_5("FALSE"),
.PCS_CH1_INT_RX_MASK_6("FALSE"),
.PCS_CH1_INT_RX_MASK_7("FALSE"),
.PCS_CH1_INT_RX_CLR_0("FALSE"),
.PCS_CH1_INT_RX_CLR_1("FALSE"),
.PCS_CH1_INT_RX_CLR_2("FALSE"),
.PCS_CH1_INT_RX_CLR_3("FALSE"),
.PCS_CH1_INT_RX_CLR_4("FALSE"),
.PCS_CH1_INT_RX_CLR_5("FALSE"),
.PCS_CH1_INT_RX_CLR_6("FALSE"),
.PCS_CH1_INT_RX_CLR_7("FALSE"),
.PCS_CH1_CA_RSTN_RX("FALSE"),
.PCS_CH1_CA_DYN_DLY_EN_RX("FALSE"),
.PCS_CH1_CA_DYN_DLY_SEL_RX("FALSE"),
.PCS_CH1_CA_RX(0),
.PCS_CH1_CA_RSTN_TX("FALSE"),
.PCS_CH1_CA_DYN_DLY_EN_TX("FALSE"),
.PCS_CH1_CA_DYN_DLY_SEL_TX("FALSE"),
.PCS_CH1_CA_TX(0),
.PCS_CH1_RXPRBS_PWR_REDUCTION("NORMAL"),
.PCS_CH1_WDALIGN_PWR_REDUCTION("NORMAL"),
.PCS_CH1_RXDEC_PWR_REDUCTION("NORMAL"),
.PCS_CH1_RXCB_PWR_REDUCTION("NORMAL"),
.PCS_CH1_RXCTC_PWR_REDUCTION("NORMAL"),
.PCS_CH1_RXGEAR_PWR_REDUCTION("NORMAL"),
.PCS_CH1_RXBRG_PWR_REDUCTION("NORMAL"),
.PCS_CH1_RXTEST_PWR_REDUCTION("NORMAL"),
.PCS_CH1_TXBRG_PWR_REDUCTION("NORMAL"),
.PCS_CH1_TXGEAR_PWR_REDUCTION("NORMAL"),
.PCS_CH1_TXENC_PWR_REDUCTION("NORMAL"),
.PCS_CH1_TXBSP_PWR_REDUCTION("NORMAL"),
.PCS_CH1_TXPRBS_PWR_REDUCTION("NORMAL"),
```

```
.PCS_CH1_TXBRG_FULL_CHK_EN("FALSE"),
.PCS_CH1_TXBRG_EMPTY_CHK_EN("FALSE"),
.PCS_CH1_RXBRG_FULL_CHK_EN("FALSE"),
.PCS_CH1_RXBRG_EMPTY_CHK_EN("FALSE"),
.PCS_CH1_CTC_FULL_CHK_EN("TRUE"),
.PCS_CH1_CTC_EMPTY_CHK_EN("TRUE"),
.PCS_CH1_CEB_FULL_CHK_EN("FALSE"),
.PCS_CH1_CEB_EMPTY_CHK_EN("FALSE"),
.PCS_CH1_FLP_FULL_CHK_EN("TRUE"),
.PCS_CH1_FLP_EMPTY_CHK_EN("TRUE"),
.PCS_CH2_BYPASS_WORD_ALIGN("FALSE"),
.PCS_CH2_BYPASS_DENC("FALSE"),
.PCS_CH2_BYPASS_BONDING("FALSE"),
.PCS_CH2_BYPASS_CTC("FALSE"),
.PCS_CH2_BYPASS_GEAR("FALSE"),
.PCS_CH2_BYPASS_BRIDGE("FALSE"),
.PCS_CH2_BYPASS_BRIDGE_FIFO("FALSE"),
.PCS_CH2_DATA_MODE("X8"),
.PCS_CH2_RX_POLARITY_INV("DELAY"),
.PCS_CH2_ALIGN_MODE("1GB"),
.PCS_CH2_SAMP_16B("X16"),
.PCS_CH2_FARLP_PWR_REDUCTION("FALSE"),
.PCS_CH2_COMMA_REG0(0),
.PCS_CH2_COMMA_MASK(0),
.PCS_CH2_CEB_MODE("10GB"),
.PCS_CH2_CTC_MODE("1SKIP"),
.PCS_CH2_A_REG(0),
.PCS_CH2_GE_AUTO_EN("FALSE"),
.PCS_CH2_SKIP_REG0(0),
.PCS_CH2_SKIP_REG1(0),
.PCS_CH2_SKIP_REG2(0),
.PCS_CH2_SKIP_REG3(0),
.PCS_CH2_DEC_DUAL("FALSE"),
.PCS_CH2_SPLIT("FALSE"),
```

```
.PCS_CH2_FIFOFLAG_CTC("FALSE"),
.PCS_CH2_COMMA_DET_MODE("COMMA_PATTERN"),
.PCS_CH2_ERRDETECT_SILENCE("FALSE"),
.PCS_CH2_PMA_RCLK_POLINV("PMA_RCLK"),
.PCS_CH2_PCS_RCLK_SEL("PMA_RCLK"),
.PCS_CH2_CB_RCLK_SEL("PMA_RCLK"),
.PCS_CH2_AFTER_CTC_RCLK_SEL("PMA_RCLK"),
.PCS_CH2_AFTER_CTC_RCLK_SEL_1("PMA_RCLK"),
.PCS_CH2_RCLK_POLINV("RCLK"),
.PCS_CH2_BRIDGE_RCLK_SEL("PMA_RCLK"),
.PCS_CH2_PCS_RCLK_EN("FALSE"),
.PCS_CH2_CB_RCLK_EN("FALSE"),
.PCS_CH2_AFTER_CTC_RCLK_EN("FALSE"),
.PCS_CH2_AFTER_CTC_RCLK_EN_GB("FALSE"),
.PCS_CH2_AFTER_CTC_RCLK_EN_GB_1("FALSE"),
.PCS_CH2_PCS_RX_RSTN("FALSE"),
.PCS_CH2_SLAVE("MASTER"),
.PCS_CH2_PCIE_SLAVE("MASTER"),
.PCS_CH2_RX_64B66B_67B("NORMAL"),
.PCS_CH2_RX_BRIDGE_CLK_POLINV("RX_BRIDGE_CLK"),
.PCS_CH2_PCS_CB_RSTN("FALSE"),
.PCS_CH2_TX_BRIDGE_GEAR_SEL("FALSE"),
.PCS_CH2_TX_BYPASS_BRIDGE_UINT("FALSE"),
.PCS_CH2_TX_BYPASS_BRIDGE_FIFO("FALSE"),
.PCS_CH2_TX_BYPASS_GEAR("FALSE"),
.PCS_CH2_TX_BYPASS_ENC("FALSE"),
.PCS_CH2_TX_BYPASS_BIT_SLIP("FALSE"),
.PCS_CH2_TX_GEAR_SPLIT("FALSE"),
.PCS_CH2_TX_DRIVE_REG_MODE("NO_CHANGE"),
.PCS_CH2_TX_BIT_SLIP_CYCLES(0),
.PCS_CH2_INT_TX_MASK_0("FALSE"),
.PCS_CH2_INT_TX_MASK_1("FALSE"),
.PCS_CH2_INT_TX_MASK_2("FALSE"),
.PCS_CH2_INT_TX_CLR_0("FALSE"),
```

```
.PCS_CH2_INT_TX_CLR_1("FALSE"),
.PCS_CH2_INT_TX_CLR_2("FALSE"),
.PCS_CH2_TX_PMA_TCLK_POLINV("PMA_TCLK"),
.PCS_CH2_TX_PCS_CLK_EN_SEL("FALSE"),
.PCS_CH2_TX_BRIDGE_TCLK_SEL("TCLK"),
.PCS_CH2_TX_TCLK_POLINV("TCLK"),
.PCS_CH2_TX_PCS_TCLK_SEL("PMA_TCLK"),
.PCS_CH2_TX_PCS_TX_RSTN("FALSE"),
.PCS_CH2_TX_SLAVE("MASTER"),
.PCS_CH2_TX_GEAR_TCLK_EN_SEL("FALSE"),
.PCS_CH2_DATA_WIDTH_MODE("X20"),
.PCS_CH2_TX_64B66B_67B("NORMAL"),
.PCS_CH2_TX_GEAR_TCLK_SEL("PMA_TCLK"),
.PCS_CH2_TX_TCLK2FABRIC_SEL("FALSE"),
.PCS_CH2_TX_OUTZZ("FALSE"),
.PCS_CH2_ENC_DUAL("FALSE"),
.PCS_CH2_TX_BITSLIP_DATA_MODE("X10"),
.PCS_CH2_TX_BRIDGE_CLK_POLINV("TX_BRIDGE_CLK"),
.PCS_CH2_COMM_REG1(0),
.PCS_CH2_RAPID_IMAX(0),
.PCS_CH2_RAPID_VMIN_1(0),
.PCS_CH2_RAPID_VMIN_2(0),
.PCS_CH2_RX_PRBS_MODE("DISABLE"),
.PCS_CH2_RX_ERRCNT_CLR("FALSE"),
.PCS_CH2_RX_PRBS_ERR_LPBK("FALSE"),
.PCS_CH2_TX_PRBS_MODE("DISABLE"),
.PCS_CH2_TX_INSERT_ER("FALSE"),
.PCS_CH2_ENABLE_PRBS_GEN("FALSE"),
.PCS_CH2_ERR_CNT(0),
.PCS_CH2_DEFAULT_RADDR(0),
.PCS_CH2_MASTER_CHECK_OFFSET(0),
.PCS_CH2_DELAY_SET(0),
.PCS_CH2_SEACH_OFFSET("20BIT"),
.PCS_CH2_CEB_RAPIDL_S_MMAX(0),
```

```
.PCS_CH2_CTC_AFULL(0),  
.PCS_CH2_CTC_AEMPTY(0),  
.PCS_CH2_CTC_CONTI_SKP_SET(0),  
.PCS_CH2_FAR_LOOP("FALSE"),  
.PCS_CH2_NEAR_LOOP("FALSE"),  
.PCS_CH2_REG_TX2RX_PLOOP_EN("FALSE"),  
.PCS_CH2_REG_TX2RX_SLOOP_EN("FALSE"),  
.PCS_CH2_REG_RX2TX_PLOOP_EN("FALSE"),  
.PCS_CH2_INT_RX_MASK_0("FALSE"),  
.PCS_CH2_INT_RX_MASK_1("FALSE"),  
.PCS_CH2_INT_RX_MASK_2("FALSE"),  
.PCS_CH2_INT_RX_MASK_3("FALSE"),  
.PCS_CH2_INT_RX_MASK_4("FALSE"),  
.PCS_CH2_INT_RX_MASK_5("FALSE"),  
.PCS_CH2_INT_RX_MASK_6("FALSE"),  
.PCS_CH2_INT_RX_MASK_7("FALSE"),  
.PCS_CH2_INT_RX_CLR_0("FALSE"),  
.PCS_CH2_INT_RX_CLR_1("FALSE"),  
.PCS_CH2_INT_RX_CLR_2("FALSE"),  
.PCS_CH2_INT_RX_CLR_3("FALSE"),  
.PCS_CH2_INT_RX_CLR_4("FALSE"),  
.PCS_CH2_INT_RX_CLR_5("FALSE"),  
.PCS_CH2_INT_RX_CLR_6("FALSE"),  
.PCS_CH2_INT_RX_CLR_7("FALSE"),  
.PCS_CH2_CA_RSTN_RX("FALSE"),  
.PCS_CH2_CA_DYN_DLY_EN_RX("FALSE"),  
.PCS_CH2_CA_DYN_DLY_SEL_RX("FALSE"),  
.PCS_CH2_CA_RX(0),  
.PCS_CH2_CA_RSTN_TX("FALSE"),  
.PCS_CH2_CA_DYN_DLY_EN_TX("FALSE"),  
.PCS_CH2_CA_DYN_DLY_SEL_TX("FALSE"),  
.PCS_CH2_CA_TX(0),  
.PCS_CH2_RXPRBS_PWR_REDUCTION("NORMAL"),  
.PCS_CH2_WDALIGN_PWR_REDUCTION("NORMAL"),
```

```
.PCS_CH2_RXDEC_PWR_REDUCTION("NORMAL"),
.PCS_CH2_RXCB_PWR_REDUCTION("NORMAL"),
.PCS_CH2_RXCTC_PWR_REDUCTION("NORMAL"),
.PCS_CH2_RXGEAR_PWR_REDUCTION("NORMAL"),
.PCS_CH2_RXBRG_PWR_REDUCTION("NORMAL"),
.PCS_CH2_RXTEST_PWR_REDUCTION("NORMAL"),
.PCS_CH2_TXBRG_PWR_REDUCTION("NORMAL"),
.PCS_CH2_TXGEAR_PWR_REDUCTION("NORMAL"),
.PCS_CH2_TXENC_PWR_REDUCTION("NORMAL"),
.PCS_CH2_TXBSLP_PWR_REDUCTION("NORMAL"),
.PCS_CH2_TXPRBS_PWR_REDUCTION("NORMAL"),
.PCS_CH2_TXBRG_FULL_CHK_EN("FALSE"),
.PCS_CH2_TXBRG_EMPTY_CHK_EN("FALSE"),
.PCS_CH2_RXBRG_FULL_CHK_EN("FALSE"),
.PCS_CH2_RXBRG_EMPTY_CHK_EN("FALSE"),
.PCS_CH2_CTC_FULL_CHK_EN("TRUE"),
.PCS_CH2_CTC_EMPTY_CHK_EN("TRUE"),
.PCS_CH2_CEB_FULL_CHK_EN("FALSE"),
.PCS_CH2_CEB_EMPTY_CHK_EN("FALSE"),
.PCS_CH2_FLP_FULL_CHK_EN("TRUE"),
.PCS_CH2_FLP_EMPTY_CHK_EN("TRUE"),
.PCS_CH3_BYPASS_WORD_ALIGN("FALSE"),
.PCS_CH3_BYPASS_DENC("FALSE"),
.PCS_CH3_BYPASS_BONDING("FALSE"),
.PCS_CH3_BYPASS_CTC("FALSE"),
.PCS_CH3_BYPASS_GEAR("FALSE"),
.PCS_CH3_BYPASS_BRIDGE("FALSE"),
.PCS_CH3_BYPASS_BRIDGE_FIFO("FALSE"),
.PCS_CH3_DATA_MODE("X8"),
.PCS_CH3_RX_POLARITY_INV("DELAY"),
.PCS_CH3_ALIGN_MODE("1GB"),
.PCS_CH3_SAMP_16B("X16"),
.PCS_CH3_FARLP_PWR_REDUCTION("FALSE"),
.PCS_CH3_COMMREG0(0),
```

```
.PCS_CH3_COMMA_MASK(0),
.PCS_CH3_CEB_MODE("10GB"),
.PCS_CH3_CTC_MODE("1SKIP"),
.PCS_CH3_A_REG(0),
.PCS_CH3_GE_AUTO_EN("FALSE"),
.PCS_CH3_SKIP_REG0(0),
.PCS_CH3_SKIP_REG1(0),
.PCS_CH3_SKIP_REG2(0),
.PCS_CH3_SKIP_REG3(0),
.PCS_CH3_DEC_DUAL("FALSE"),
.PCS_CH3_SPLIT("FALSE"),
.PCS_CH3_FIFOFLAG_CTC("FALSE"),
.PCS_CH3_COMMA_DET_MODE("COMMA_PATTERN"),
.PCS_CH3_ERRDETECT_SILENCE("FALSE"),
.PCS_CH3_PMA_RCLK_POLINV("PMA_RCLK"),
.PCS_CH3_PCS_RCLK_SEL("PMA_RCLK"),
.PCS_CH3_CB_RCLK_SEL("PMA_RCLK"),
.PCS_CH3_AFTER_CTC_RCLK_SEL("PMA_RCLK"),
.PCS_CH3_AFTER_CTC_RCLK_SEL_1("PMA_RCLK"),
.PCS_CH3_RCLK_POLINV("RCLK"),
.PCS_CH3_BRIDGE_RCLK_SEL("PMA_RCLK"),
.PCS_CH3_PCS_RCLK_EN("FALSE"),
.PCS_CH3_CB_RCLK_EN("FALSE"),
.PCS_CH3_AFTER_CTC_RCLK_EN("FALSE"),
.PCS_CH3_AFTER_CTC_RCLK_EN_GB("FALSE"),
.PCS_CH3_AFTER_CTC_RCLK_EN_GB_1("FALSE"),
.PCS_CH3_PCS_RX_RSTN("FALSE"),
.PCS_CH3_SLAVE("MASTER"),
.PCS_CH3_PCIE_SLAVE("MASTER"),
.PCS_CH3_RX_64B66B_67B("NORMAL"),
.PCS_CH3_RX_BRIDGE_CLK_POLINV("RX_BRIDGE_CLK"),
.PCS_CH3_PCS_CB_RSTN("FALSE"),
.PCS_CH3_TX_BRIDGE_GEAR_SEL("FALSE"),
.PCS_CH3_TX_BYPASS_BRIDGE_UINT("FALSE"),
```

```
.PCS_CH3_TX_BYPASS_BRIDGE_FIFO("FALSE"),
.PCS_CH3_TX_BYPASS_GEAR("FALSE"),
.PCS_CH3_TX_BYPASS_ENC("FALSE"),
.PCS_CH3_TX_BYPASS_BIT_SLIP("FALSE"),
.PCS_CH3_TX_GEAR_SPLIT("FALSE"),
.PCS_CH3_TX_DRIVE_REG_MODE("NO_CHANGE"),
.PCS_CH3_TX_BIT_SLIP_CYCLES(0),
.PCS_CH3_INT_TX_MASK_0("FALSE"),
.PCS_CH3_INT_TX_MASK_1("FALSE"),
.PCS_CH3_INT_TX_MASK_2("FALSE"),
.PCS_CH3_INT_TX_CLR_0("FALSE"),
.PCS_CH3_INT_TX_CLR_1("FALSE"),
.PCS_CH3_INT_TX_CLR_2("FALSE"),
.PCS_CH3_TX_PMA_TCLK_POLINV("PMA_TCLK"),
.PCS_CH3_TX_PCS_CLK_EN_SEL("FALSE"),
.PCS_CH3_TX_BRIDGE_TCLK_SEL("TCLK"),
.PCS_CH3_TX_TCLK_POLINV("TCLK"),
.PCS_CH3_TX_PCS_TCLK_SEL("PMA_TCLK"),
.PCS_CH3_TX_PCS_TX_RSTN("FALSE"),
.PCS_CH3_TX_SLAVE("MASTER"),
.PCS_CH3_TX_GEAR_TCLK_EN_SEL("FALSE"),
.PCS_CH3_DATA_WIDTH_MODE("X20"),
.PCS_CH3_TX_64B66B_67B("NORMAL"),
.PCS_CH3_TX_GEAR_TCLK_SEL("PMA_TCLK"),
.PCS_CH3_TX_TCLK2FABRIC_SEL("FALSE"),
.PCS_CH3_TX_OUTZZ("FALSE"),
.PCS_CH3_ENC_DUAL("FALSE"),
.PCS_CH3_TX_BITSLIP_DATA_MODE("X10"),
.PCS_CH3_TX_BRIDGE_CLK_POLINV("TX_BRIDGE_CLK"),
.PCS_CH3_COMM_REG1(0),
.PCS_CH3_RAPID_IMAX(0),
.PCS_CH3_RAPID_VMIN_1(0),
.PCS_CH3_RAPID_VMIN_2(0),
.PCS_CH3_RX_PRBS_MODE("DISABLE"),
```

```
.PCS_CH3_RX_ERRCNT_CLR("FALSE"),
.PCS_CH3_RX_PRBS_ERR_LPBK("FALSE"),
.PCS_CH3_TX_PRBS_MODE("DISABLE"),
.PCS_CH3_TX_INSERT_ER("FALSE"),
.PCS_CH3_ENABLE_PRBS_GEN("FALSE"),
.PCS_CH3_ERR_CNT(0),
.PCS_CH3_DEFAULT_RADDR(0),
.PCS_CH3_MASTER_CHECK_OFFSET(0),
.PCS_CH3_DELAY_SET(0),
.PCS_CH3_SEACH_OFFSET("20BIT"),
.PCS_CH3_CEB_RAPIDL_S_MMAX(0),
.PCS_CH3_CTC_AFULL(0),
.PCS_CH3_CTC_AEMPTY(0),
.PCS_CH3_CTC_CONTI_SKP_SET(0),
.PCS_CH3_FAR_LOOP("FALSE"),
.PCS_CH3_NEAR_LOOP("FALSE"),
.PCS_CH3_REG_TX2RX_PLOOP_EN("FALSE"),
.PCS_CH3_REG_TX2RX_SLOOP_EN("FALSE"),
.PCS_CH3_REG_RX2TX_PLOOP_EN("FALSE"),
.PCS_CH3_INT_RX_MASK_0("FALSE"),
.PCS_CH3_INT_RX_MASK_1("FALSE"),
.PCS_CH3_INT_RX_MASK_2("FALSE"),
.PCS_CH3_INT_RX_MASK_3("FALSE"),
.PCS_CH3_INT_RX_MASK_4("FALSE"),
.PCS_CH3_INT_RX_MASK_5("FALSE"),
.PCS_CH3_INT_RX_MASK_6("FALSE"),
.PCS_CH3_INT_RX_MASK_7("FALSE"),
.PCS_CH3_INT_RX_CLR_0("FALSE"),
.PCS_CH3_INT_RX_CLR_1("FALSE"),
.PCS_CH3_INT_RX_CLR_2("FALSE"),
.PCS_CH3_INT_RX_CLR_3("FALSE"),
.PCS_CH3_INT_RX_CLR_4("FALSE"),
.PCS_CH3_INT_RX_CLR_5("FALSE"),
.PCS_CH3_INT_RX_CLR_6("FALSE"),
```

```
.PCS_CH3_INT_RX_CLR_7("FALSE"),
.PCS_CH3_CA_RSTN_RX("FALSE"),
.PCS_CH3_CA_DYN_DLY_EN_RX("FALSE"),
.PCS_CH3_CA_DYN_DLY_SEL_RX("FALSE"),
.PCS_CH3_CA_RX(0),
.PCS_CH3_CA_RSTN_TX("FALSE"),
.PCS_CH3_CA_DYN_DLY_EN_TX("FALSE"),
.PCS_CH3_CA_DYN_DLY_SEL_TX("FALSE"),
.PCS_CH3_CA_TX(0),
.PCS_CH3_RXPRBS_PWR_REDUCTION("NORMAL"),
.PCS_CH3_WDALIGN_PWR_REDUCTION("NORMAL"),
.PCS_CH3_RXDEC_PWR_REDUCTION("NORMAL"),
.PCS_CH3_RXCB_PWR_REDUCTION("NORMAL"),
.PCS_CH3_RXCTC_PWR_REDUCTION("NORMAL"),
.PCS_CH3_RXGEAR_PWR_REDUCTION("NORMAL"),
.PCS_CH3_RXBRG_PWR_REDUCTION("NORMAL"),
.PCS_CH3_RXTEST_PWR_REDUCTION("NORMAL"),
.PCS_CH3_TXBRG_PWR_REDUCTION("NORMAL"),
.PCS_CH3_TXGEAR_PWR_REDUCTION("NORMAL"),
.PCS_CH3_TXENC_PWR_REDUCTION("NORMAL"),
.PCS_CH3_TXBSLP_PWR_REDUCTION("NORMAL"),
.PCS_CH3_TXPRBS_PWR_REDUCTION("NORMAL"),
.PCS_CH3_TXBRG_FULL_CHK_EN("FALSE"),
.PCS_CH3_TXBRG_EMPTY_CHK_EN("FALSE"),
.PCS_CH3_RXBRG_FULL_CHK_EN("FALSE"),
.PCS_CH3_RXBRG_EMPTY_CHK_EN("FALSE"),
.PCS_CH3_CTC_FULL_CHK_EN("TRUE"),
.PCS_CH3_CTC_EMPTY_CHK_EN("TRUE"),
.PCS_CH3_CEB_FULL_CHK_EN("FALSE"),
.PCS_CH3_CEB_EMPTY_CHK_EN("FALSE"),
.PCS_CH3_FLP_FULL_CHK_EN("TRUE"),
.PCS_CH3_FLP_EMPTY_CHK_EN("TRUE"),
.PMA_CH0_REG_RX_PD("ON"),
.PMA_CH0_REG_RX_PD_EN("FALSE"),
```

```
.PMA_CH0_REG_RX_CLKPATH_PD("ON"),
.PMA_CH0_REG_RX_CLKPATH_PD_EN("FALSE"),
.PMA_CH0_REG_RX_DATAPATH_PD("ON"),
.PMA_CH0_REG_RX_DATAPATH_PD_EN("FALSE"),
.PMA_CH0_REG_RX_SIGDET_PD("ON"),
.PMA_CH0_REG_RX_SIGDET_PD_EN("FALSE"),
.PMA_CH0_REG_RX_DCC_RST_N("TRUE"),
.PMA_CH0_REG_RX_DCC_RST_N_EN("FALSE"),
.PMA_CH0_REG_RX_CDR_RST_N("TRUE"),
.PMA_CH0_REG_RX_CDR_RST_N_EN("FALSE"),
.PMA_CH0_REG_RX_SIGDET_RST_N("TRUE"),
.PMA_CH0_REG_RX_SIGDET_RST_N_EN("FALSE"),
.PMA_CH0_REG_RX_PXPCLK_SLIP("FALSE"),
.PMA_CH0_REG_RX_PXPCLK_SLIP_OW("DISABLE"),
.PMA_CH0_REG_RX_PCLKSWITCH_RST_N("TRUE"),
.PMA_CH0_REG_RX_PCLKSWITCH_RST_N_EN("FALSE"),
.PMA_CH0_REG_RX_PCLKSWITCH("FALSE"),
.PMA_CH0_REG_RX_PCLKSWITCH_EN("FALSE"),
.PMA_CH0_REG_RX_HIGHZ("FALSE"),
.PMA_CH0_REG_RX_HIGHZ_EN("FALSE"),
.PMA_CH0_REG_RX_EQ_C_SET(8),
.PMA_CH0_REG_RX_EQ_R_SET(8),
.PMA_CH0_REG_RX_BUSWIDTH("20BIT"),
.PMA_CH0_REG_RX_BUSWIDTH_EN("FALSE"),
.PMA_CH0_REG_RX_RATE("DIV1"),
.PMA_CH0_REG_RX_RATE_EN("FALSE"),
.PMA_CH0_REG_RX_RES_TRIM(51),
.PMA_CH0_REG_RX_RES_TRIM_EN("FALSE"),
.PMA_CH0_REG_RX_EQ_OFF("FALSE"),
.PMA_CH0_REG_RX_PREAMP_IC(1367),
.PMA_CH0_REG_RX_PCLK_EDGE_SEL("POS_EDGE"),
.PMA_CH0_REG_RX_PIBUF_IC(2),
.PMA_CH0_REG_RX_DCC_IC_RX(3),
.PMA_CH0_REG_RX_DCC_IC_TX(3),
```

```
.PMA_CH0_REG_RX_ICTRL_TRX("100PCT"),
.PMA_CH0_REG_RX_ICTRL_SIGDET(5),
.PMA_CH0_REG_RX_ICTRL_PREAMP("100PCT"),
.PMA_CH0_REG_RX_ICTRL_SLICER("100PCT"),
.PMA_CH0_REG_RX_ICTRL_PIBUF("100PCT"),
.PMA_CH0_REG_RX_ICTRL_PI("100PCT"),
.PMA_CH0_REG_RX_ICTRL_DCC("100PCT"),
.PMA_CH0_REG_RX_ICTRL_PREDRV("100PCT"),
.PMA_CH0_REG_TX_RATE("DIV1"),
.PMA_CH0_REG_TX_RATE_EN("FALSE"),
.PMA_CH0_REG_RX_TX2RX_PLPBK_RST_N("TRUE"),
.PMA_CH0_REG_RX_TX2RX_PLPBK_RST_N_EN("FALSE"),
.PMA_CH0_REG_RX_TX2RX_PLPBK_EN("FALSE"),
.PMA_CH0_REG_TXCLK_SEL("PLL"),
.PMA_CH0_REG_RX_DATA_POLARITY("NORMAL"),
.PMA_CH0_REG_RX_ERR_INSERT("FALSE"),
.PMA_CH0_REG_UDP_CHK_EN("FALSE"),
.PMA_CH0_REG_PRBS_SEL("PRBS7"),
.PMA_CH0_REG_PRBS_CHK_EN("FALSE"),
.PMA_CH0_REG_PRBS_CHK_WIDTH_SEL("20BIT"),
.PMA_CH0_REG_BIST_CHK_PAT_SEL("PRBS"),
.PMA_CH0_REG_LOAD_ERR_CNT("DISABLE"),
.PMA_CH0_REG_CHK_COUNTER_EN("FALSE"),
.PMA_CH0_REG_CDR_PROP_GAIN(5),
.PMA_CH0_REG_CDR_PROP_TURBO_GAIN(6),
.PMA_CH0_REG_CDR_INT_GAIN(5),
.PMA_CH0_REG_CDR_INT_TURBO_GAIN(6),
.PMA_CH0_REG_CDR_INT_SAT_MAX(992),
.PMA_CH0_REG_CDR_INT_SAT_MIN(32),
.PMA_CH0_REG_CDR_INT_RST("FALSE"),
.PMA_CH0_REG_CDR_INT_RST_OW("DISABLE"),
.PMA_CH0_REG_CDR_PROP_RST("FALSE"),
.PMA_CH0_REG_CDR_PROP_RST_OW("DISABLE"),
.PMA_CH0_REG_CDR_LOCK_RST("FALSE"),
```

```
.PMA_CH0_REG_CDR_LOCK_RST_OW("DISABLE"),
.PMA_CH0_REG_CDR_RX_PI_FORCE_SEL(0),
.PMA_CH0_REG_CDR_RX_PI_FORCE_D(0),
.PMA_CH0_REG_CDR_LOCK_TIMER("1_2U"),
.PMA_CH0_REG_CDR_TURBO_MODE_TIMER(1),
.PMA_CH0_REG_CDR_LOCK_VAL("FALSE"),
.PMA_CH0_REG_CDR_LOCK_OW("DISABLE"),
.PMA_CH0_REG_CDR_INT_SAT_DET_EN("TRUE"),
.PMA_CH0_REG_CDR_SAT_DET_STATUS_EN("FALSE"),
.PMA_CH0_REG_CDR_SAT_DET_STATUS_RESET_EN("FALSE"),
.PMA_CH0_REG_CDR_PI_CTRL_RST("FALSE"),
.PMA_CH0_REG_CDR_PI_CTRL_RST_OW("DISABLE"),
.PMA_CH0_REG_CDR_SAT_DET_RST("FALSE"),
.PMA_CH0_REG_CDR_SAT_DET_RST_OW("DISABLE"),
.PMA_CH0_REG_CDR_SAT_DET_STICKY_RST("FALSE"),
.PMA_CH0_REG_CDR_SAT_DET_STICKY_RST_OW("DISABLE"),
.PMA_CH0_REG_CDR_SIGDET_STATUS_DIS("FALSE"),
.PMA_CH0_REG_CDR_SAT_DET_TIMER(2),
.PMA_CH0_REG_CDR_SAT_DET_STATUS_VAL("FALSE"),
.PMA_CH0_REG_CDR_SAT_DET_STATUS_OW("DISABLE"),
.PMA_CH0_REG_CDR_TURBO_MODE_EN("TRUE"),
.PMA_CH0_REG_CDR_STATUS_RADDR_INIT(0),
.PMA_CH0_REG_CDR_STATUS_FIFO_EN("TRUE"),
.PMA_CH0_REG_PMA_TEST_SEL(0),
.PMA_CH0_REG_OOB_COMWAKE_GAP_MIN(3),
.PMA_CH0_REG_OOB_COMWAKE_GAP_MAX(11),
.PMA_CH0_REG_OOB_COMINIT_GAP_MIN(15),
.PMA_CH0_REG_OOB_COMINIT_GAP_MAX(35),
.PMA_CH0_REG_RX_PIBUF_IC_TX(1),
.PMA_CH0_REG_COMWAKE_STATUS_CLEAR(0),
.PMA_CH0_REG_COMINIT_STATUS_CLEAR(0),
.PMA_CH0_REG_RX_SYNC_RST_N_EN("FALSE"),
.PMA_CH0_REG_RX_SYNC_RST_N("TRUE"),
.PMA_CH0_REG_RX_SATA_COMINIT_OW("DISABLE"),
```

```
.PMA_CH0_REG_RX_SATA_COMINIT("FALSE"),
.PMA_CH0_REG_RX_SATA_COMWAKE_OW("DISABLE"),
.PMA_CH0_REG_RX_SATA_COMWAKE("FALSE"),
.PMA_CH0_REG_RX_DCC_DISABLE("ENABLE"),
.PMA_CH0_REG_TX_DCC_DISABLE("ENABLE"),
.PMA_CH0_REG_RX_SLIP_SEL_EN("FALSE"),
.PMA_CH0_REG_RX_SLIP_SEL(0),
.PMA_CH0_REG_RX_SLIP_EN("FALSE"),
.PMA_CH0_REG_RX_SIGDET_STATUS_SEL(5),
.PMA_CH0_REG_RX_SIGDET_FSM_RST_N("TRUE"),
.PMA_CH0_REG_RX_SIGDET_STATUS_OW("DISABLE"),
.PMA_CH0_REG_RX_SIGDET_STATUS("FALSE"),
.PMA_CH0_REG_RX_SIGDET_VTH("50MV"),
.PMA_CH0_REG_RX_SIGDET_GRM(0),
.PMA_CH0_REG_RX_SIGDET_PULSE_EXT("DISABLE"),
.PMA_CH0_REG_RX_SIGDET_CH2_SEL(0),
.PMA_CH0_REG_RX_SIGDET_CH2_CHK_WINDOW(3),
.PMA_CH0_REG_RX_SIGDET_CHK_WINDOW_EN("TRUE"),
.PMA_CH0_REG_RX_SIGDET_NOSIG_COUNT_SETTING(4),
.PMA_CH0_REG_RX_SIGDET_OOB_DET_COUNT_VAL(0),
.PMA_CH0_REG_SLIP_FIFO_INV_EN("FALSE"),
.PMA_CH0_REG_SLIP_FIFO_INV("POS_EDGE"),
.PMA_CH0_REG_RX_SIGDET_4OOB_DET_SEL(7),
.PMA_CH0_REG_RX_SIGDET_IC_I(10),
.PMA_CH0_REG_RX_OOB_DETECTOR_RESET_N_OW("DISABLE"),
.PMA_CH0_REG_RX_OOB_DETECTOR_RESET_N("FALSE"),
.PMA_CH0_REG_RX_OOB_DETECTOR_PD_OW("DISABLE"),
.PMA_CH0_REG_RX_OOB_DETECTOR_PD("ON"),
.PMA_CH0_REG_RX_TERM_CM_CTRL("5DIV7"),
.PMA_CH0_REG_TX_PD("ON"),
.PMA_CH0_REG_TX_PD_OW("DISABLE"),
.PMA_CH0_REG_TX_CLKPATH_PD("ON"),
.PMA_CH0_REG_TX_CLKPATH_PD_OW("DISABLE"),
.PMA_CH0_REG_TX_BEACON_TIMER_SEL(0),
```

```
.PMA_CH0_REG_TX_RXDET_REQ_OW("DISABLE"),
.PMA_CH0_REG_TX_RXDET_REQ("FALSE"),
.PMA_CH0_REG_TX_BEACON_EN_OW("DISABLE"),
.PMA_CH0_REG_TX_BEACON_EN("FALSE"),
.PMA_CH0_REG_TX_EI_EN_OW("DISABLE"),
.PMA_CH0_REG_TX_EI_EN("FALSE"),
.PMA_CH0_REG_TX_RES_CAL_EN("FALSE"),
.PMA_CH0_REG_TX_RES_CAL(51),
.PMA_CH0_REG_TX_BIAS_CAL_EN("FALSE"),
.PMA_CH0_REG_TX_BIAS_CTRL(48),
.PMA_CH0_REG_TX_RXDET_TIMER_SEL("12CYCLE"),
.PMA_CH0_REG_TX_SYNC_OW("DISABLE"),
.PMA_CH0_REG_TX_SYNC("DISABLE"),
.PMA_CH0_REG_TX_PD_POST("OFF"),
.PMA_CH0_REG_TX_PD_POST_OW("DISABLE"),
.PMA_CH0_REG_TX_RESET_N_OW("DISABLE"),
.PMA_CH0_REG_TX_RESET_N("TRUE"),
.PMA_CH0_REG_TX_DCC_RESET_N_OW("DISABLE"),
.PMA_CH0_REG_TX_DCC_RESET_N("TRUE"),
.PMA_CH0_REG_TX_BUSWIDTH_OW("DISABLE"),
.PMA_CH0_REG_TX_BUSWIDTH("20BIT"),
.PMA_CH0_REG_PLL_READY_OW("DISABLE"),
.PMA_CH0_REG_PLL_READY("TRUE"),
.PMA_CH0_REG_TX_PCLK_SW_OW("DISABLE"),
.PMA_CH0_REG_TX_PCLK_SW("TRUE"),
.PMA_CH0_REG_EI_PCLK_DELAY_SEL(0),
.PMA_CH0_REG_TX_DRV01_DAC0(0),
.PMA_CH0_REG_TX_DRV01_DAC1(10),
.PMA_CH0_REG_TX_DRV01_DAC2(16),
.PMA_CH0_REG_TX_DRV00_DAC0(63),
.PMA_CH0_REG_TX_DRV00_DAC1(53),
.PMA_CH0_REG_TX_DRV00_DAC2(48),
.PMA_CH0_REG_TX_AMP0(8),
.PMA_CH0_REG_TX_AMP1(16),
```

```
.PMA_CH0_REG_TX_AMP2(32),  
.PMA_CH0_REG_TX_AMP3(48),  
.PMA_CH0_REG_TX_AMP4(56),  
.PMA_CH0_REG_TX_MARGIN(0),  
.PMA_CH0_REG_TX_MARGIN_OW("DISABLE"),  
.PMA_CH0_REG_TX_DEEMP(0),  
.PMA_CH0_REG_TX_DEEMP_OW("DISABLE"),  
.PMA_CH0_REG_TX_SWING("FALSE"),  
.PMA_CH0_REG_TX_SWING_OW("DISABLE"),  
.PMA_CH0_REG_TX_RXDET_THRESHOLD("100MV"),  
.PMA_CH0_REG_TX_BEACON_OSC_CTRL(4),  
.PMA_CH0_REG_TX_PREDRV_DAC(1),  
.PMA_CH0_REG_TX_PREDRV_CM_CTRL(1),  
.PMA_CH0_REG_TX_TX2RX_SLPBACK_EN("FALSE"),  
.PMA_CH0_REG_TX_PCLK_EDGE_SEL("POS_EDGE"),  
.PMA_CH0_REG_TX_RXDET_STATUS_OW("DISABLE"),  
.PMA_CH0_REG_TX_RXDET_STATUS("TRUE"),  
.PMA_CH0_REG_TX_PRBS_GEN_EN("FALSE"),  
.PMA_CH0_REG_TX_PRBS_GEN_WIDTH_SEL("20BIT"),  
.PMA_CH0_REG_TX_PRBS_SEL("PRBS7"),  
.PMA_CH0_REG_TX_UDP_DATA(256773),  
.PMA_CH0_REG_TX_FIFO_RST_N("FALSE"),  
.PMA_CH0_REG_TX_FIFO_WP_CTRL(2),  
.PMA_CH0_REG_TX_FIFO_EN("FALSE"),  
.PMA_CH0_REG_TX_DATA_MUX_SEL(0),  
.PMA_CH0_REG_TX_ERR_INSERT("FALSE"),  
.PMA_CH0_REG_TX_SATA_EN("FALSE"),  
.PMA_CH0_REG_RATE_CHANGE_TXPCLK_ON_OW("DISABLE"),  
.PMA_CH0_REG_RATE_CHANGE_TXPCLK_ON("ENABLE"),  
.PMA_CH0_REG_TX_PULLUP_DAC0(8),  
.PMA_CH0_REG_TX_PULLUP_DAC1(8),  
.PMA_CH0_REG_TX_PULLUP_DAC2(8),  
.PMA_CH0_REG_TX_PULLUP_DAC3(8),  
.PMA_CH0_REG_TX_OOB_DELAY_SEL(0),
```

```
.PMA_CH0_REG_TX_POLARITY("NORMAL"),
.PMA_CH0_REG_TX_SLPBK_AMP(1),
.PMA_CH0_REG_TX_LS_MODE_EN("FALSE"),
.PMA_CH0_REG_TX_JTAG_MODE_EN_OW("DISABLE"),
.PMA_CH0_REG_TX_JTAG_MODE_EN("FALSE"),
.PMA_CH0_REG_RX_JTAG_MODE_EN_OW("DISABLE"),
.PMA_CH0_REG_RX_JTAG_MODE_EN("FALSE"),
.PMA_CH0_REG_RX_JTAG_OE("DISABLE"),
.PMA_CH0_REG_RX_ACJTAG_VHYSTSE(0),
.PMA_CH0_REG_TX_FBCLK_FAR_EN("FALSE"),
.PMA_CH0_REG_RX_TERM_MODE_CTRL(6),
.PMA_CH0_REG_PLPBK_TXPCLK_EN("TRUE"),
.PMA_CH0_REG_RX_609_600(0),
.PMA_CH0_REG_RX_CDR_617_610(0),
.PMA_CH0_REG_RX_CDR_623_618(0),
.PMA_CH0_REG_RX_631_624(0),
.PMA_CH0_REG_RX_639_632(0),
.PMA_CH0_REG_RX_647_640(0),
.PMA_CH0_REG_RX_655_648(0),
.PMA_CH0_REG_RX_659_656(0),
.PMA_CH0_CFG_LANE_POWERUP("OFF"),
.PMA_CH0_CFG_PMA_POR_N("FALSE"),
.PMA_CH0_CFG_RX_LANE_POWERUP("OFF"),
.PMA_CH0_CFG_RX_PMA_RSTN("FALSE"),
.PMA_CH0_CFG_TX_LANE_POWERUP("OFF"),
.PMA_CH0_CFG_TX_PMA_RSTN("FALSE"),
.PMA_CH0_CFG_CTLE_ADP_RSTN("TRUE"),
.PMA_CH0_REG_RESERVED_48_45(0),
.PMA_CH0_REG_RESERVED_69(0),
.PMA_CH0_REG_RESERVED_77_76(0),
.PMA_CH0_REG_RESERVED_171_164(0),
.PMA_CH0_REG_RESERVED_175_172(0),
.PMA_CH0_REG_RESERVED_190(0),
.PMA_CH0_REG_RESERVED_233_232(0),
```

```
.PMA_CH0_REG_RESERVED_235_234(0),  
.PMA_CH0_REG_RESERVED_241_240(0),  
.PMA_CH0_REG_RESERVED_285_283(0),  
.PMA_CH0_REG_RESERVED_286(0),  
.PMA_CH0_REG_RESERVED_295(0),  
.PMA_CH0_REG_RESERVED_298(0),  
.PMA_CH0_REG_RESERVED_332_325(0),  
.PMA_CH0_REG_RESERVED_340_333(0),  
.PMA_CH0_REG_RESERVED_348_341(0),  
.PMA_CH0_REG_RESERVED_354_349(0),  
.PMA_CH0_REG_RESERVED_373(0),  
.PMA_CH0_REG_RESERVED_376(0),  
.PMA_CH0_REG_RESERVED_452(0),  
.PMA_CH0_REG_RESERVED_502_499(0),  
.PMA_CH0_REG_RESERVED_506_505(0),  
.PMA_CH0_REG_RESERVED_550_549(0),  
.PMA_CH0_REG_RESERVED_556_552(0),  
.PMA_CH1_REG_RX_PD("ON"),  
.PMA_CH1_REG_RX_PD_EN("FALSE"),  
.PMA_CH1_REG_RX_CLKPATH_PD("ON"),  
.PMA_CH1_REG_RX_CLKPATH_PD_EN("FALSE"),  
.PMA_CH1_REG_RX_DATAPATH_PD("ON"),  
.PMA_CH1_REG_RX_DATAPATH_PD_EN("FALSE"),  
.PMA_CH1_REG_RX_SIGDET_PD("ON"),  
.PMA_CH1_REG_RX_SIGDET_PD_EN("FALSE"),  
.PMA_CH1_REG_RX_DCC_RST_N("TRUE"),  
.PMA_CH1_REG_RX_DCC_RST_N_EN("FALSE"),  
.PMA_CH1_REG_RX_CDR_RST_N("TRUE"),  
.PMA_CH1_REG_RX_CDR_RST_N_EN("FALSE"),  
.PMA_CH1_REG_RX_SIGDET_RST_N("TRUE"),  
.PMA_CH1_REG_RX_SIGDET_RST_N_EN("FALSE"),  
.PMA_CH1_REG_RX_PCLKSLIP("FALSE"),  
.PMA_CH1_REG_RX_PCLKSLIP_OW("DISABLE"),  
.PMA_CH1_REG_RX_PCLKSWITCH_RST_N("TRUE"),
```

```
.PMA_CH1_REG_RX_PCLKSWITCH_RST_N_EN("FALSE"),
.PMA_CH1_REG_RX_PCLKSWITCH("FALSE"),
.PMA_CH1_REG_RX_PCLKSWITCH_EN("FALSE"),
.PMA_CH1_REG_RX_HIGHZ("FALSE"),
.PMA_CH1_REG_RX_HIGHZ_EN("FALSE"),
.PMA_CH1_REG_RX_EQ_C_SET(8),
.PMA_CH1_REG_RX_EQ_R_SET(8),
.PMA_CH1_REG_RX_BUSWIDTH("20BIT"),
.PMA_CH1_REG_RX_BUSWIDTH_EN("FALSE"),
.PMA_CH1_REG_RX_RATE("DIV1"),
.PMA_CH1_REG_RX_RATE_EN("FALSE"),
.PMA_CH1_REG_RX_RES_TRIM(51),
.PMA_CH1_REG_RX_RES_TRIM_EN("FALSE"),
.PMA_CH1_REG_RX_EQ_OFF("FALSE"),
.PMA_CH1_REG_RX_PREAMP_IC(1367),
.PMA_CH1_REG_RX_PCLK_EDGE_SEL("POS_EDGE"),
.PMA_CH1_REG_RX_PIBUF_IC(2),
.PMA_CH1_REG_RX_DCC_IC_RX(3),
.PMA_CH1_REG_RX_DCC_IC_TX(3),
.PMA_CH1_REG_RX_ICTRL_TRX("100PCT"),
.PMA_CH1_REG_RX_ICTRL_SIGDET(5),
.PMA_CH1_REG_RX_ICTRL_PREAMP("100PCT"),
.PMA_CH1_REG_RX_ICTRL_SLICER("100PCT"),
.PMA_CH1_REG_RX_ICTRL_PIBUF("100PCT"),
.PMA_CH1_REG_RX_ICTRL_PI("100PCT"),
.PMA_CH1_REG_RX_ICTRL_DCC("100PCT"),
.PMA_CH1_REG_RX_ICTRL_PREDRV("100PCT"),
.PMA_CH1_REG_TX_RATE("DIV1"),
.PMA_CH1_REG_TX_RATE_EN("FALSE"),
.PMA_CH1_REG_RX_TX2RX_PLPBK_RST_N("TRUE"),
.PMA_CH1_REG_RX_TX2RX_PLPBK_RST_N_EN("FALSE"),
.PMA_CH1_REG_RX_TX2RX_PLPBK_EN("FALSE"),
.PMA_CH1_REG_TXCLK_SEL("PLL"),
.PMA_CH1_REG_RX_DATA_POLARITY("NORMAL"),
```

```
.PMA_CH1_REG_RX_ERR_INSERT("FALSE"),
.PMA_CH1_REG_UDP_CHK_EN("FALSE"),
.PMA_CH1_REG_PRBS_SEL("PRBS7"),
.PMA_CH1_REG_PRBS_CHK_EN("FALSE"),
.PMA_CH1_REG_PRBS_CHK_WIDTH_SEL("20BIT"),
.PMA_CH1_REG_BIST_CHK_PAT_SEL("PRBS"),
.PMA_CH1_REG_LOAD_ERR_CNT("DISABLE"),
.PMA_CH1_REG_CHK_COUNTER_EN("FALSE"),
.PMA_CH1_REG_CDR_PROP_GAIN(5),
.PMA_CH1_REG_CDR_PROP_TURBO_GAIN(6),
.PMA_CH1_REG_CDR_INT_GAIN(5),
.PMA_CH1_REG_CDR_INT_TURBO_GAIN(6),
.PMA_CH1_REG_CDR_INT_SAT_MAX(992),
.PMA_CH1_REG_CDR_INT_SAT_MIN(32),
.PMA_CH1_REG_CDR_INT_RST("FALSE"),
.PMA_CH1_REG_CDR_INT_RST_OW("DISABLE"),
.PMA_CH1_REG_CDR_PROP_RST("FALSE"),
.PMA_CH1_REG_CDR_PROP_RST_OW("DISABLE"),
.PMA_CH1_REG_CDR_LOCK_RST("FALSE"),
.PMA_CH1_REG_CDR_LOCK_RST_OW("DISABLE"),
.PMA_CH1_REG_CDR_RX_PI_FORCE_SEL(0),
.PMA_CH1_REG_CDR_RX_PI_FORCE_D(0),
.PMA_CH1_REG_CDR_LOCK_TIMER("1_2U"),
.PMA_CH1_REG_CDR_TURBO_MODE_TIMER(1),
.PMA_CH1_REG_CDR_LOCK_VAL("FALSE"),
.PMA_CH1_REG_CDR_LOCK_OW("DISABLE"),
.PMA_CH1_REG_CDR_INT_SAT_DET_EN("TRUE"),
.PMA_CH1_REG_CDR_SAT_DET_STATUS_EN("FALSE"),
.PMA_CH1_REG_CDR_SAT_DET_STATUS_RESET_EN("FALSE"),
.PMA_CH1_REG_CDR_PI_CTRL_RST("FALSE"),
.PMA_CH1_REG_CDR_PI_CTRL_RST_OW("DISABLE"),
.PMA_CH1_REG_CDR_SAT_DET_RST("FALSE"),
.PMA_CH1_REG_CDR_SAT_DET_RST_OW("DISABLE"),
.PMA_CH1_REG_CDR_SAT_DET_STICKY_RST("FALSE"),
```

```
.PMA_CH1_REG_CDR_SAT_DET_STICKY_RST_OW("DISABLE"),
.PMA_CH1_REG_CDR_SIGDET_STATUS_DIS("FALSE"),
.PMA_CH1_REG_CDR_SAT_DET_TIMER(2),
.PMA_CH1_REG_CDR_SAT_DET_STATUS_VAL("FALSE"),
.PMA_CH1_REG_CDR_SAT_DET_STATUS_OW("DISABLE"),
.PMA_CH1_REG_CDR_TURBO_MODE_EN("TRUE"),
.PMA_CH1_REG_CDR_STATUS_RADDR_INIT(0),
.PMA_CH1_REG_CDR_STATUS_FIFO_EN("TRUE"),
.PMA_CH1_REG_PMA_TEST_SEL(0),
.PMA_CH1_REG_OOB_COMWAKE_GAP_MIN(3),
.PMA_CH1_REG_OOB_COMWAKE_GAP_MAX(11),
.PMA_CH1_REG_OOB_COMINIT_GAP_MIN(15),
.PMA_CH1_REG_OOB_COMINIT_GAP_MAX(35),
.PMA_CH1_REG_RX_PIBUF_IC_TX(1),
.PMA_CH1_REG_COMWAKE_STATUS_CLEAR(0),
.PMA_CH1_REG_COMINIT_STATUS_CLEAR(0),
.PMA_CH1_REG_RX_SYNC_RST_N_EN("FALSE"),
.PMA_CH1_REG_RX_SYNC_RST_N("TRUE"),
.PMA_CH1_REG_RX_SATA_COMINIT_OW("DISABLE"),
.PMA_CH1_REG_RX_SATA_COMINIT("FALSE"),
.PMA_CH1_REG_RX_SATA_COMWAKE_OW("DISABLE"),
.PMA_CH1_REG_RX_SATA_COMWAKE("FALSE"),
.PMA_CH1_REG_RX_DCC_DISABLE("ENABLE"),
.PMA_CH1_REG_TX_DCC_DISABLE("ENABLE"),
.PMA_CH1_REG_RX_SLIP_SEL_EN("FALSE"),
.PMA_CH1_REG_RX_SLIP_SEL(0),
.PMA_CH1_REG_RX_SLIP_EN("FALSE"),
.PMA_CH1_REG_RX_SIGDET_STATUS_SEL(5),
.PMA_CH1_REG_RX_SIGDET_FSM_RST_N("TRUE"),
.PMA_CH1_REG_RX_SIGDET_STATUS_OW("DISABLE"),
.PMA_CH1_REG_RX_SIGDET_STATUS("FALSE"),
.PMA_CH1_REG_RX_SIGDET_VTH("50MV"),
.PMA_CH1_REG_RX_SIGDET_GRM(0),
.PMA_CH1_REG_RX_SIGDET_PULSE_EXT("DISABLE"),
```

```
.PMA_CH1_REG_RX_SIGDET_CH2_SEL(0),  
.PMA_CH1_REG_RX_SIGDET_CH2_CHK_WINDOW(3),  
.PMA_CH1_REG_RX_SIGDET_CHK_WINDOW_EN("TRUE"),  
.PMA_CH1_REG_RX_SIGDET_NOSIG_COUNT_SETTING(4),  
.PMA_CH1_REG_RX_SIGDET_OOB_DET_COUNT_VAL(0),  
.PMA_CH1_REG_SLIP_FIFO_INV_EN("FALSE"),  
.PMA_CH1_REG_SLIP_FIFO_INV("POS_EDGE"),  
.PMA_CH1_REG_RX_SIGDET_4OOB_DET_SEL(7),  
.PMA_CH1_REG_RX_SIGDET_IC_I(10),  
.PMA_CH1_REG_RX_OOB_DETECTOR_RESET_N_OW("DISABLE"),  
.PMA_CH1_REG_RX_OOB_DETECTOR_RESET_N("FALSE"),  
.PMA_CH1_REG_RX_OOB_DETECTOR_PD_OW("DISABLE"),  
.PMA_CH1_REG_RX_OOB_DETECTOR_PD("ON"),  
.PMA_CH1_REG_RX_TERM_CM_CTRL("5DIV7"),  
.PMA_CH1_REG_TX_PD("ON"),  
.PMA_CH1_REG_TX_PD_OW("DISABLE"),  
.PMA_CH1_REG_TX_CLKPATH_PD("ON"),  
.PMA_CH1_REG_TX_CLKPATH_PD_OW("DISABLE"),  
.PMA_CH1_REG_TX_BEACON_TIMER_SEL(0),  
.PMA_CH1_REG_TX_RXDET_REQ_OW("DISABLE"),  
.PMA_CH1_REG_TX_RXDET_REQ("FALSE"),  
.PMA_CH1_REG_TX_BEACON_EN_OW("DISABLE"),  
.PMA_CH1_REG_TX_BEACON_EN("FALSE"),  
.PMA_CH1_REG_TX_EI_EN_OW("DISABLE"),  
.PMA_CH1_REG_TX_EI_EN("FALSE"),  
.PMA_CH1_REG_TX_RES_CAL_EN("FALSE"),  
.PMA_CH1_REG_TX_RES_CAL(51),  
.PMA_CH1_REG_TX_BIAS_CAL_EN("FALSE"),  
.PMA_CH1_REG_TX_BIAS_CTRL(48),  
.PMA_CH1_REG_TX_RXDET_TIMER_SEL("12CYCLE"),  
.PMA_CH1_REG_TX_SYNC_OW("DISABLE"),  
.PMA_CH1_REG_TX_SYNC("DISABLE"),  
.PMA_CH1_REG_TX_PD_POST("OFF"),  
.PMA_CH1_REG_TX_PD_POST_OW("DISABLE"),
```

```
.PMA_CH1_REG_TX_RESET_N_OW("DISABLE"),
.PMA_CH1_REG_TX_RESET_N("TRUE"),
.PMA_CH1_REG_TX_DCC_RESET_N_OW("DISABLE"),
.PMA_CH1_REG_TX_DCC_RESET_N("TRUE"),
.PMA_CH1_REG_TX_BUSWIDTH_OW("DISABLE"),
.PMA_CH1_REG_TX_BUSWIDTH("20BIT"),
.PMA_CH1_REG_PLL_READY_OW("DISABLE"),
.PMA_CH1_REG_PLL_READY("TRUE"),
.PMA_CH1_REG_TX_PCLK_SW_OW("DISABLE"),
.PMA_CH1_REG_TX_PCLK_SW("TRUE"),
.PMA_CH1_REG_EI_PCLK_DELAY_SEL(0),
.PMA_CH1_REG_TX_DRV01_DAC0(0),
.PMA_CH1_REG_TX_DRV01_DAC1(10),
.PMA_CH1_REG_TX_DRV01_DAC2(16),
.PMA_CH1_REG_TX_DRV00_DAC0(63),
.PMA_CH1_REG_TX_DRV00_DAC1(53),
.PMA_CH1_REG_TX_DRV00_DAC2(48),
.PMA_CH1_REG_TX_AMP0(8),
.PMA_CH1_REG_TX_AMP1(16),
.PMA_CH1_REG_TX_AMP2(32),
.PMA_CH1_REG_TX_AMP3(48),
.PMA_CH1_REG_TX_AMP4(56),
.PMA_CH1_REG_TX_MARGIN(0),
.PMA_CH1_REG_TX_MARGIN_OW("DISABLE"),
.PMA_CH1_REG_TX_DEEMP(0),
.PMA_CH1_REG_TX_DEEMP_OW("DISABLE"),
.PMA_CH1_REG_TX_SWING("FALSE"),
.PMA_CH1_REG_TX_SWING_OW("DISABLE"),
.PMA_CH1_REG_TX_RXDET_THRESHOLD("100MV"),
.PMA_CH1_REG_TX_BEACON_OSC_CTRL(4),
.PMA_CH1_REG_TX_PREDRV_DAC(1),
.PMA_CH1_REG_TX_PREDRV_CM_CTRL(1),
.PMA_CH1_REG_TX_TX2RX_SLPBACK_EN("FALSE"),
.PMA_CH1_REG_TX_PCLK_EDGE_SEL("POS_EDGE"),
```

```
.PMA_CH1_REG_TX_RXDET_STATUS_OW("DISABLE"),
.PMA_CH1_REG_TX_RXDET_STATUS("TRUE"),
.PMA_CH1_REG_TX_PRBS_GEN_EN("FALSE"),
.PMA_CH1_REG_TX_PRBS_GEN_WIDTH_SEL("20BIT"),
.PMA_CH1_REG_TX_PRBS_SEL("PRBS7"),
.PMA_CH1_REG_TX_UDP_DATA(256773),
.PMA_CH1_REG_TX_FIFO_RST_N("FALSE"),
.PMA_CH1_REG_TX_FIFO_WP_CTRL(2),
.PMA_CH1_REG_TX_FIFO_EN("FALSE"),
.PMA_CH1_REG_TX_DATA_MUX_SEL(0),
.PMA_CH1_REG_TX_ERR_INSERT("FALSE"),
.PMA_CH1_REG_TX_SATA_EN("FALSE"),
.PMA_CH1_REG_RATE_CHANGE_TXPCLK_ON_OW("DISABLE"),
.PMA_CH1_REG_RATE_CHANGE_TXPCLK_ON("ENABLE"),
.PMA_CH1_REG_TX_PULLUP_DAC0(8),
.PMA_CH1_REG_TX_PULLUP_DAC1(8),
.PMA_CH1_REG_TX_PULLUP_DAC2(8),
.PMA_CH1_REG_TX_PULLUP_DAC3(8),
.PMA_CH1_REG_TX_OOB_DELAY_SEL(0),
.PMA_CH1_REG_TX_POLARITY("NORMAL"),
.PMA_CH1_REG_TX_SLPBK_AMP(1),
.PMA_CH1_REG_TX_LS_MODE_EN("FALSE"),
.PMA_CH1_REG_TX_JTAG_MODE_EN_OW("DISABLE"),
.PMA_CH1_REG_TX_JTAG_MODE_EN("FALSE"),
.PMA_CH1_REG_RX_JTAG_MODE_EN_OW("DISABLE"),
.PMA_CH1_REG_RX_JTAG_MODE_EN("FALSE"),
.PMA_CH1_REG_RX_JTAG_OE("DISABLE"),
.PMA_CH1_REG_RX_ACJTAG_VHYSTSE(0),
.PMA_CH1_REG_TX_FBCLOCK_FAR_EN("FALSE"),
.PMA_CH1_REG_RX_TERM_MODE_CTRL(6),
.PMA_CH1_REG_PLPBK_TXPCLK_EN("TRUE"),
.PMA_CH1_REG_TX_609_600(0),
.PMA_CH1_REG_RX_CDR_617_610(0),
.PMA_CH1_REG_RX_CDR_623_618(0),
```

```
.PMA_CH1_REG_RX_631_624(0),  
.PMA_CH1_REG_RX_639_632(0),  
.PMA_CH1_REG_RX_647_640(0),  
.PMA_CH1_REG_RX_655_648(0),  
.PMA_CH1_REG_RX_659_656(0),  
.PMA_CH1_CFG_LANE_POWERUP("OFF"),  
.PMA_CH1_CFG_PMA_POR_N("FALSE"),  
.PMA_CH1_CFG_RX_LANE_POWERUP("OFF"),  
.PMA_CH1_CFG_RX_PMA_RSTN("FALSE"),  
.PMA_CH1_CFG_TX_LANE_POWERUP("OFF"),  
.PMA_CH1_CFG_TX_PMA_RSTN("FALSE"),  
.PMA_CH1_CFG_CTLE_ADP_RSTN("TRUE"),  
.PMA_CH1_REG_RESERVED_48_45(0),  
.PMA_CH1_REG_RESERVED_69(0),  
.PMA_CH1_REG_RESERVED_77_76(0),  
.PMA_CH1_REG_RESERVED_171_164(0),  
.PMA_CH1_REG_RESERVED_175_172(0),  
.PMA_CH1_REG_RESERVED_190(0),  
.PMA_CH1_REG_RESERVED_233_232(0),  
.PMA_CH1_REG_RESERVED_235_234(0),  
.PMA_CH1_REG_RESERVED_241_240(0),  
.PMA_CH1_REG_RESERVED_285_283(0),  
.PMA_CH1_REG_RESERVED_286(0),  
.PMA_CH1_REG_RESERVED_295(0),  
.PMA_CH1_REG_RESERVED_298(0),  
.PMA_CH1_REG_RESERVED_332_325(0),  
.PMA_CH1_REG_RESERVED_340_333(0),  
.PMA_CH1_REG_RESERVED_348_341(0),  
.PMA_CH1_REG_RESERVED_354_349(0),  
.PMA_CH1_REG_RESERVED_373(0),  
.PMA_CH1_REG_RESERVED_376(0),  
.PMA_CH1_REG_RESERVED_452(0),  
.PMA_CH1_REG_RESERVED_502_499(0),  
.PMA_CH1_REG_RESERVED_506_505(0),
```

```
.PMA_CH1_REG_RESERVED_550_549(0),  
.PMA_CH1_REG_RESERVED_556_552(0),  
.PMA_CH2_REG_RX_PD("ON"),  
.PMA_CH2_REG_RX_PD_EN("FALSE"),  
.PMA_CH2_REG_RX_CLKPATH_PD("ON"),  
.PMA_CH2_REG_RX_CLKPATH_PD_EN("FALSE"),  
.PMA_CH2_REG_RX_DATAPATH_PD("ON"),  
.PMA_CH2_REG_RX_DATAPATH_PD_EN("FALSE"),  
.PMA_CH2_REG_RX_SIGDET_PD("ON"),  
.PMA_CH2_REG_RX_SIGDET_PD_EN("FALSE"),  
.PMA_CH2_REG_RX_DCC_RST_N("TRUE"),  
.PMA_CH2_REG_RX_DCC_RST_N_EN("FALSE"),  
.PMA_CH2_REG_RX_CDR_RST_N("TRUE"),  
.PMA_CH2_REG_RX_CDR_RST_N_EN("FALSE"),  
.PMA_CH2_REG_RX_SIGDET_RST_N("TRUE"),  
.PMA_CH2_REG_RX_SIGDET_RST_N_EN("FALSE"),  
.PMA_CH2_REG_RXPCLK_SLIP("FALSE"),  
.PMA_CH2_REG_RXPCLK_SLIP_OW("DISABLE"),  
.PMA_CH2_REG_RX_PCLKSWITCH_RST_N("TRUE"),  
.PMA_CH2_REG_RX_PCLKSWITCH_RST_N_EN("FALSE"),  
.PMA_CH2_REG_RX_PCLKSWITCH("FALSE"),  
.PMA_CH2_REG_RX_PCLKSWITCH_EN("FALSE"),  
.PMA_CH2_REG_RX_HIGHZ("FALSE"),  
.PMA_CH2_REG_RX_HIGHZ_EN("FALSE"),  
.PMA_CH2_REG_RX_EQ_C_SET(8),  
.PMA_CH2_REG_RX_EQ_R_SET(8),  
.PMA_CH2_REG_RX_BUSWIDTH("20BIT"),  
.PMA_CH2_REG_RX_BUSWIDTH_EN("FALSE"),  
.PMA_CH2_REG_RX_RATE("DIV1"),  
.PMA_CH2_REG_RX_RATE_EN("FALSE"),  
.PMA_CH2_REG_RX_RES_TRIM(51),  
.PMA_CH2_REG_RX_RES_TRIM_EN("FALSE"),  
.PMA_CH2_REG_RX_EQ_OFF("FALSE"),  
.PMA_CH2_REG_RX_PREAMP_IC(1367),
```

```
.PMA_CH2_REG_RX_PCLK_EDGE_SEL("POS_EDGE"),
.PMA_CH2_REG_RX_PIBUF_IC(2),
.PMA_CH2_REG_RX_DCC_IC_RX(3),
.PMA_CH2_REG_RX_DCC_IC_TX(3),
.PMA_CH2_REG_RX_ICTRL_TRX("100PCT"),
.PMA_CH2_REG_RX_ICTRL_SIGDET(5),
.PMA_CH2_REG_RX_ICTRL_PREAMP("100PCT"),
.PMA_CH2_REG_RX_ICTRL_SLICER("100PCT"),
.PMA_CH2_REG_RX_ICTRL_PIBUF("100PCT"),
.PMA_CH2_REG_RX_ICTRL_PI("100PCT"),
.PMA_CH2_REG_RX_ICTRL_DCC("100PCT"),
.PMA_CH2_REG_RX_ICTRL_PREDRV("100PCT"),
.PMA_CH2_REG_TX_RATE("DIV1"),
.PMA_CH2_REG_TX_RATE_EN("FALSE"),
.PMA_CH2_REG_RX_TX2RX_PLPBK_RST_N("TRUE"),
.PMA_CH2_REG_RX_TX2RX_PLPBK_RST_N_EN("FALSE"),
.PMA_CH2_REG_RX_TX2RX_PLPBK_EN("FALSE"),
.PMA_CH2_REG_TXCLK_SEL("PLL"),
.PMA_CH2_REG_RX_DATA_POLARITY("NORMAL"),
.PMA_CH2_REG_RX_ERR_INSERT("FALSE"),
.PMA_CH2_REG_UDP_CHK_EN("FALSE"),
.PMA_CH2_REG_PRBS_SEL("PRBS7"),
.PMA_CH2_REG_PRBS_CHK_EN("FALSE"),
.PMA_CH2_REG_PRBS_CHK_WIDTH_SEL("20BIT"),
.PMA_CH2_REG_BIST_CHK_PAT_SEL("PRBS"),
.PMA_CH2_REG_LOAD_ERR_CNT("DISABLE"),
.PMA_CH2_REG_CHK_COUNTER_EN("FALSE"),
.PMA_CH2_REG_CDR_PROP_GAIN(5),
.PMA_CH2_REG_CDR_PROP_TURBO_GAIN(6),
.PMA_CH2_REG_CDR_INT_GAIN(5),
.PMA_CH2_REG_CDR_INT_TURBO_GAIN(6),
.PMA_CH2_REG_CDR_INT_SAT_MAX(992),
.PMA_CH2_REG_CDR_INT_SAT_MIN(32),
.PMA_CH2_REG_CDR_INT_RST("FALSE"),
```

```
.PMA_CH2_REG_CDR_INT_RST_OW("DISABLE"),
.PMA_CH2_REG_CDR_PROP_RST("FALSE"),
.PMA_CH2_REG_CDR_PROP_RST_OW("DISABLE"),
.PMA_CH2_REG_CDR_LOCK_RST("FALSE"),
.PMA_CH2_REG_CDR_LOCK_RST_OW("DISABLE"),
.PMA_CH2_REG_CDR_RX_PI_FORCE_SEL(0),
.PMA_CH2_REG_CDR_RX_PI_FORCE_D(0),
.PMA_CH2_REG_CDR_LOCK_TIMER("1_2U"),
.PMA_CH2_REG_CDR_TURBO_MODE_TIMER(1),
.PMA_CH2_REG_CDR_LOCK_VAL("FALSE"),
.PMA_CH2_REG_CDR_LOCK_OW("DISABLE"),
.PMA_CH2_REG_CDR_INT_SAT_DET_EN("TRUE"),
.PMA_CH2_REG_CDR_SAT_DET_STATUS_EN("FALSE"),
.PMA_CH2_REG_CDR_SAT_DET_STATUS_RESET_EN("FALSE"),
.PMA_CH2_REG_CDR_PI_CTRL_RST("FALSE"),
.PMA_CH2_REG_CDR_PI_CTRL_RST_OW("DISABLE"),
.PMA_CH2_REG_CDR_SAT_DET_RST("FALSE"),
.PMA_CH2_REG_CDR_SAT_DET_RST_OW("DISABLE"),
.PMA_CH2_REG_CDR_SAT_DET_STICKY_RST("FALSE"),
.PMA_CH2_REG_CDR_SAT_DET_STICKY_RST_OW("DISABLE"),
.PMA_CH2_REG_CDR_SIGDET_STATUS_DIS("FALSE"),
.PMA_CH2_REG_CDR_SAT_DET_TIMER(2),
.PMA_CH2_REG_CDR_SAT_DET_STATUS_VAL("FALSE"),
.PMA_CH2_REG_CDR_SAT_DET_STATUS_OW("DISABLE"),
.PMA_CH2_REG_CDR_TURBO_MODE_EN("TRUE"),
.PMA_CH2_REG_CDR_STATUS_RADDR_INIT(0),
.PMA_CH2_REG_CDR_STATUS_FIFO_EN("TRUE"),
.PMA_CH2_REG_PMA_TEST_SEL(0),
.PMA_CH2_REG_OOB_COMWAKE_GAP_MIN(3),
.PMA_CH2_REG_OOB_COMWAKE_GAP_MAX(11),
.PMA_CH2_REG_OOB_COMINIT_GAP_MIN(15),
.PMA_CH2_REG_OOB_COMINIT_GAP_MAX(35),
.PMA_CH2_REG_RX_PIBUF_IC_TX(1),
.PMA_CH2_REG_COMWAKE_STATUS_CLEAR(0),
```

```
.PMA_CH2_REG_COMINIT_STATUS_CLEAR(0),
.PMA_CH2_REG_RX_SYNC_RST_N_EN("FALSE"),
.PMA_CH2_REG_RX_SYNC_RST_N("TRUE"),
.PMA_CH2_REG_RX_SATA_COMINIT_OW("DISABLE"),
.PMA_CH2_REG_RX_SATA_COMINIT("FALSE"),
.PMA_CH2_REG_RX_SATA_COMWAKE_OW("DISABLE"),
.PMA_CH2_REG_RX_SATA_COMWAKE("FALSE"),
.PMA_CH2_REG_RX_DCC_DISABLE("ENABLE"),
.PMA_CH2_REG_TX_DCC_DISABLE("ENABLE"),
.PMA_CH2_REG_RX_SLIP_SEL_EN("FALSE"),
.PMA_CH2_REG_RX_SLIP_SEL(0),
.PMA_CH2_REG_RX_SLIP_EN("FALSE"),
.PMA_CH2_REG_RX_SIGDET_STATUS_SEL(5),
.PMA_CH2_REG_RX_SIGDET_FSM_RST_N("TRUE"),
.PMA_CH2_REG_RX_SIGDET_STATUS_OW("DISABLE"),
.PMA_CH2_REG_RX_SIGDET_STATUS("FALSE"),
.PMA_CH2_REG_RX_SIGDET_VTH("50MV"),
.PMA_CH2_REG_RX_SIGDET_GRM(0),
.PMA_CH2_REG_RX_SIGDET_PULSE_EXT("DISABLE"),
.PMA_CH2_REG_RX_SIGDET_CH2_SEL(0),
.PMA_CH2_REG_RX_SIGDET_CH2_CHK_WINDOW(3),
.PMA_CH2_REG_RX_SIGDET_CHK_WINDOW_EN("TRUE"),
.PMA_CH2_REG_RX_SIGDET_NOSIG_COUNT_SETTING(4),
.PMA_CH2_REG_RX_SIGDET_OOB_DET_COUNT_VAL(0),
.PMA_CH2_REG_SLIP_FIFO_INV_EN("FALSE"),
.PMA_CH2_REG_SLIP_FIFO_INV("POS_EDGE"),
.PMA_CH2_REG_RX_SIGDET_4OOB_DET_SEL(7),
.PMA_CH2_REG_RX_SIGDET_IC_I(10),
.PMA_CH2_REG_RX_OOB_DETECTOR_RESET_N_OW("DISABLE"),
.PMA_CH2_REG_RX_OOB_DETECTOR_RESET_N("FALSE"),
.PMA_CH2_REG_RX_OOB_DETECTOR_PD_OW("DISABLE"),
.PMA_CH2_REG_RX_OOB_DETECTOR_PD("ON"),
.PMA_CH2_REG_RX_TERM_CM_CTRL("5DIV7"),
.PMA_CH2_REG_TX_PD("ON"),
```

```
.PMA_CH2_REG_TX_PD_OW("DISABLE"),
.PMA_CH2_REG_TX_CLKPATH_PD("ON"),
.PMA_CH2_REG_TX_CLKPATH_PD_OW("DISABLE"),
.PMA_CH2_REG_TX_BEACON_TIMER_SEL(0),
.PMA_CH2_REG_TX_RXDET_REQ_OW("DISABLE"),
.PMA_CH2_REG_TX_RXDET_REQ("FALSE"),
.PMA_CH2_REG_TX_BEACON_EN_OW("DISABLE"),
.PMA_CH2_REG_TX_BEACON_EN("FALSE"),
.PMA_CH2_REG_TX_EI_EN_OW("DISABLE"),
.PMA_CH2_REG_TX_EI_EN("FALSE"),
.PMA_CH2_REG_TX_RES_CAL_EN("FALSE"),
.PMA_CH2_REG_TX_RES_CAL(51),
.PMA_CH2_REG_TX_BIAS_CAL_EN("FALSE"),
.PMA_CH2_REG_TX_BIAS_CTRL(48),
.PMA_CH2_REG_TX_RXDET_TIMER_SEL("12CYCLE"),
.PMA_CH2_REG_TX_SYNC_OW("DISABLE"),
.PMA_CH2_REG_TX_SYNC("DISABLE"),
.PMA_CH2_REG_TX_PD_POST("OFF"),
.PMA_CH2_REG_TX_PD_POST_OW("DISABLE"),
.PMA_CH2_REG_TX_RESET_N_OW("DISABLE"),
.PMA_CH2_REG_TX_RESET_N("TRUE"),
.PMA_CH2_REG_TX_DCC_RESET_N_OW("DISABLE"),
.PMA_CH2_REG_TX_DCC_RESET_N("TRUE"),
.PMA_CH2_REG_TX_BUSWIDTH_OW("DISABLE"),
.PMA_CH2_REG_TX_BUSWIDTH("20BIT"),
.PMA_CH2_REG_PLL_READY_OW("DISABLE"),
.PMA_CH2_REG_PLL_READY("TRUE"),
.PMA_CH2_REG_TX_PCLK_SW_OW("DISABLE"),
.PMA_CH2_REG_TX_PCLK_SW("TRUE"),
.PMA_CH2_REG_EI_PCLK_DELAY_SEL(0),
.PMA_CH2_REG_TX_DRV01_DAC0(0),
.PMA_CH2_REG_TX_DRV01_DAC1(10),
.PMA_CH2_REG_TX_DRV01_DAC2(16),
.PMA_CH2_REG_TX_DRV00_DAC0(63),
```

```
.PMA_CH2_REG_TX_DRV00_DAC1(53),  
.PMA_CH2_REG_TX_DRV00_DAC2(48),  
.PMA_CH2_REG_TX_AMP0(8),  
.PMA_CH2_REG_TX_AMP1(16),  
.PMA_CH2_REG_TX_AMP2(32),  
.PMA_CH2_REG_TX_AMP3(48),  
.PMA_CH2_REG_TX_AMP4(56),  
.PMA_CH2_REG_TX_MARGIN(0),  
.PMA_CH2_REG_TX_MARGIN_OW("DISABLE"),  
.PMA_CH2_REG_TX_DEEMP(0),  
.PMA_CH2_REG_TX_DEEMP_OW("DISABLE"),  
.PMA_CH2_REG_TX_SWING("FALSE"),  
.PMA_CH2_REG_TX_SWING_OW("DISABLE"),  
.PMA_CH2_REG_TX_RXDET_THRESHOLD("100MV"),  
.PMA_CH2_REG_TX_BEACON_OSC_CTRL(4),  
.PMA_CH2_REG_TX_PREDRV_DAC(1),  
.PMA_CH2_REG_TX_PREDRV_CM_CTRL(1),  
.PMA_CH2_REG_TX_TX2RX_SLPBACK_EN("FALSE"),  
.PMA_CH2_REG_TX_PCLK_EDGE_SEL("POS_EDGE"),  
.PMA_CH2_REG_TX_RXDET_STATUS_OW("DISABLE"),  
.PMA_CH2_REG_TX_RXDET_STATUS("TRUE"),  
.PMA_CH2_REG_TX_PRBS_GEN_EN("FALSE"),  
.PMA_CH2_REG_TX_PRBS_GEN_WIDTH_SEL("20BIT"),  
.PMA_CH2_REG_TX_PRBS_SEL("PRBS7"),  
.PMA_CH2_REG_TX_UDP_DATA(256773),  
.PMA_CH2_REG_TX_FIFO_RST_N("FALSE"),  
.PMA_CH2_REG_TX_FIFO_WP_CTRL(2),  
.PMA_CH2_REG_TX_FIFO_EN("FALSE"),  
.PMA_CH2_REG_TX_DATA_MUX_SEL(0),  
.PMA_CH2_REG_TX_ERR_INSERT("FALSE"),  
.PMA_CH2_REG_TX_SATA_EN("FALSE"),  
.PMA_CH2_REG_RATE_CHANGE_TXPCLK_ON_OW("DISABLE"),  
.PMA_CH2_REG_RATE_CHANGE_TXPCLK_ON("ENABLE"),  
.PMA_CH2_REG_TX_PULLUP_DAC0(8),
```

```
.PMA_CH2_REG_TX_PULLUP_DAC1(8),  
.PMA_CH2_REG_TX_PULLUP_DAC2(8),  
.PMA_CH2_REG_TX_PULLUP_DAC3(8),  
.PMA_CH2_REG_TX_OOB_DELAY_SEL(0),  
.PMA_CH2_REG_TX_POLARITY("NORMAL"),  
.PMA_CH2_REG_TX_SLPBK_AMP(1),  
.PMA_CH2_REG_TX_LS_MODE_EN("FALSE"),  
.PMA_CH2_REG_TX_JTAG_MODE_EN_OW("DISABLE"),  
.PMA_CH2_REG_TX_JTAG_MODE_EN("FALSE"),  
.PMA_CH2_REG_RX_JTAG_MODE_EN_OW("DISABLE"),  
.PMA_CH2_REG_RX_JTAG_MODE_EN("FALSE"),  
.PMA_CH2_REG_RX_JTAG_OE("DISABLE"),  
.PMA_CH2_REG_RX_ACJTAG_VHYSTSE(0),  
.PMA_CH2_REG_TX_FBCLOCK_FAR_EN("FALSE"),  
.PMA_CH2_REG_RX_TERM_MODE_CTRL(6),  
.PMA_CH2_REG_PLPBK_TXPCLK_EN("TRUE"),  
.PMA_CH2_REG_TX_609_600(0),  
.PMA_CH2_REG_RX_CDR_617_610(0),  
.PMA_CH2_REG_RX_CDR_623_618(0),  
.PMA_CH2_REG_RX_631_624(0),  
.PMA_CH2_REG_RX_639_632(0),  
.PMA_CH2_REG_RX_647_640(0),  
.PMA_CH2_REG_RX_655_648(0),  
.PMA_CH2_REG_RX_659_656(0),  
.PMA_CH2_CFG_LANE_POWERUP("OFF"),  
.PMA_CH2_CFG_PMA_POR_N("FALSE"),  
.PMA_CH2_CFG_RX_LANE_POWERUP("OFF"),  
.PMA_CH2_CFG_RX_PMA_RSTN("FALSE"),  
.PMA_CH2_CFG_TX_LANE_POWERUP("OFF"),  
.PMA_CH2_CFG_TX_PMA_RSTN("FALSE"),  
.PMA_CH2_CFG_CTLE_ADPRSTN("TRUE"),  
.PMA_CH2_REG_RESERVED_48_45(0),  
.PMA_CH2_REG_RESERVED_69(0),  
.PMA_CH2_REG_RESERVED_77_76(0),
```

```
.PMA_CH2_REG_RESERVED_171_164(0),
.PMA_CH2_REG_RESERVED_175_172(0),
.PMA_CH2_REG_RESERVED_190(0),
.PMA_CH2_REG_RESERVED_233_232(0),
.PMA_CH2_REG_RESERVED_235_234(0),
.PMA_CH2_REG_RESERVED_241_240(0),
.PMA_CH2_REG_RESERVED_285_283(0),
.PMA_CH2_REG_RESERVED_286(0),
.PMA_CH2_REG_RESERVED_295(0),
.PMA_CH2_REG_RESERVED_298(0),
.PMA_CH2_REG_RESERVED_332_325(0),
.PMA_CH2_REG_RESERVED_340_333(0),
.PMA_CH2_REG_RESERVED_348_341(0),
.PMA_CH2_REG_RESERVED_354_349(0),
.PMA_CH2_REG_RESERVED_373(0),
.PMA_CH2_REG_RESERVED_376(0),
.PMA_CH2_REG_RESERVED_452(0),
.PMA_CH2_REG_RESERVED_502_499(0),
.PMA_CH2_REG_RESERVED_506_505(0),
.PMA_CH2_REG_RESERVED_550_549(0),
.PMA_CH2_REG_RESERVED_556_552(0),
.PMA_CH3_REG_RX_PD("ON"),
.PMA_CH3_REG_RX_PD_EN("FALSE"),
.PMA_CH3_REG_RX_CLKPATH_PD("ON"),
.PMA_CH3_REG_RX_CLKPATH_PD_EN("FALSE"),
.PMA_CH3_REG_RX_DATAPATH_PD("ON"),
.PMA_CH3_REG_RX_DATAPATH_PD_EN("FALSE"),
.PMA_CH3_REG_RX_SIGDET_PD("ON"),
.PMA_CH3_REG_RX_SIGDET_PD_EN("FALSE"),
.PMA_CH3_REG_RX_DCC_RST_N("TRUE"),
.PMA_CH3_REG_RX_DCC_RST_N_EN("FALSE"),
.PMA_CH3_REG_RX_CDR_RST_N("TRUE"),
.PMA_CH3_REG_RX_CDR_RST_N_EN("FALSE"),
.PMA_CH3_REG_RX_SIGDET_RST_N("TRUE"),
```

```
.PMA_CH3_REG_RX_SIGDET_RST_N_EN("FALSE"),
.PMA_CH3_REG_RXPCLK_SLIP("FALSE"),
.PMA_CH3_REG_RXPCLK_SLIP_OW("DISABLE"),
.PMA_CH3_REG_RX_PCLKSWITCH_RST_N("TRUE"),
.PMA_CH3_REG_RX_PCLKSWITCH_RST_N_EN("FALSE"),
.PMA_CH3_REG_RX_PCLKSWITCH("FALSE"),
.PMA_CH3_REG_RX_PCLKSWITCH_EN("FALSE"),
.PMA_CH3_REG_RX_HIGHZ("FALSE"),
.PMA_CH3_REG_RX_HIGHZ_EN("FALSE"),
.PMA_CH3_REG_RX_EQ_C_SET(8),
.PMA_CH3_REG_RX_EQ_R_SET(8),
.PMA_CH3_REG_RX_BUSWIDTH("20BIT"),
.PMA_CH3_REG_RX_BUSWIDTH_EN("FALSE"),
.PMA_CH3_REG_RX_RATE("DIV1"),
.PMA_CH3_REG_RX_RATE_EN("FALSE"),
.PMA_CH3_REG_RX_RES_TRIM(51),
.PMA_CH3_REG_RX_RES_TRIM_EN("FALSE"),
.PMA_CH3_REG_RX_EQ_OFF("FALSE"),
.PMA_CH3_REG_RX_PREAMP_IC(1367),
.PMA_CH3_REG_RX_PCLK_EDGE_SEL("POS_EDGE"),
.PMA_CH3_REG_RX_PIBUF_IC(2),
.PMA_CH3_REG_RX_DCC_IC_RX(3),
.PMA_CH3_REG_RX_DCC_IC_TX(3),
.PMA_CH3_REG_RXICTRL_TRX("100PCT"),
.PMA_CH3_REG_RXICTRL_SIGDET(5),
.PMA_CH3_REG_RXICTRL_PREAMP("100PCT"),
.PMA_CH3_REG_RXICTRL_SLICER("100PCT"),
.PMA_CH3_REG_RXICTRL_PIBUF("100PCT"),
.PMA_CH3_REG_RXICTRL_PI("100PCT"),
.PMA_CH3_REG_RXICTRL_DCC("100PCT"),
.PMA_CH3_REG_RXICTRL_PREDRV("100PCT"),
.PMA_CH3_REG_TX_RATE("DIV1"),
.PMA_CH3_REG_TX_RATE_EN("FALSE"),
.PMA_CH3_REG_RX_TX2RX_PLPBK_RST_N("TRUE"),
```

```
.PMA_CH3_REG_RX_TX2RX_PLPBK_RST_N_EN("FALSE"),
.PMA_CH3_REG_RX_TX2RX_PLPBK_EN("FALSE"),
.PMA_CH3_REG_TXCLK_SEL("PLL"),
.PMA_CH3_REG_RX_DATA_POLARITY("NORMAL"),
.PMA_CH3_REG_RX_ERR_INSERT("FALSE"),
.PMA_CH3_REG_UDP_CHK_EN("FALSE"),
.PMA_CH3_REG_PRBS_SEL("PRBS7"),
.PMA_CH3_REG_PRBS_CHK_EN("FALSE"),
.PMA_CH3_REG_PRBS_CHK_WIDTH_SEL("20BIT"),
.PMA_CH3_REG_BIST_CHK_PAT_SEL("PRBS"),
.PMA_CH3_REG_LOAD_ERR_CNT("DISABLE"),
.PMA_CH3_REG_CHK_COUNTER_EN("FALSE"),
.PMA_CH3_REG_CDR_PROP_GAIN(5),
.PMA_CH3_REG_CDR_PROP_TURBO_GAIN(6),
.PMA_CH3_REG_CDR_INT_GAIN(5),
.PMA_CH3_REG_CDR_INT_TURBO_GAIN(6),
.PMA_CH3_REG_CDR_INT_SAT_MAX(992),
.PMA_CH3_REG_CDR_INT_SAT_MIN(32),
.PMA_CH3_REG_CDR_INT_RST("FALSE"),
.PMA_CH3_REG_CDR_INT_RST_OW("DISABLE"),
.PMA_CH3_REG_CDR_PROP_RST("FALSE"),
.PMA_CH3_REG_CDR_PROP_RST_OW("DISABLE"),
.PMA_CH3_REG_CDR_LOCK_RST("FALSE"),
.PMA_CH3_REG_CDR_LOCK_RST_OW("DISABLE"),
.PMA_CH3_REG_CDR_RX_PI_FORCE_SEL(0),
.PMA_CH3_REG_CDR_RX_PI_FORCE_D(0),
.PMA_CH3_REG_CDR_LOCK_TIMER("1_2U"),
.PMA_CH3_REG_CDR_TURBO_MODE_TIMER(1),
.PMA_CH3_REG_CDR_LOCK_VAL("FALSE"),
.PMA_CH3_REG_CDR_LOCK_OW("DISABLE"),
.PMA_CH3_REG_CDR_INT_SAT_DET_EN("TRUE"),
.PMA_CH3_REG_CDR_SAT_DET_STATUS_EN("FALSE"),
.PMA_CH3_REG_CDR_SAT_DET_STATUS_RESET_EN("FALSE"),
.PMA_CH3_REG_CDR_PI_CTRL_RST("FALSE"),
```

```
.PMA_CH3_REG_CDR_PI_CTRL_RST_OW("DISABLE"),
.PMA_CH3_REG_CDR_SAT_DET_RST("FALSE"),
.PMA_CH3_REG_CDR_SAT_DET_RST_OW("DISABLE"),
.PMA_CH3_REG_CDR_SAT_DET_STICKY_RST("FALSE"),
.PMA_CH3_REG_CDR_SAT_DET_STICKY_RST_OW("DISABLE"),
.PMA_CH3_REG_CDR_SIGDET_STATUS_DIS("FALSE"),
.PMA_CH3_REG_CDR_SAT_DET_TIMER(2),
.PMA_CH3_REG_CDR_SAT_DET_STATUS_VAL("FALSE"),
.PMA_CH3_REG_CDR_SAT_DET_STATUS_OW("DISABLE"),
.PMA_CH3_REG_CDR_TURBO_MODE_EN("TRUE"),
.PMA_CH3_REG_CDR_STATUS_RADDR_INIT(0),
.PMA_CH3_REG_CDR_STATUS_FIFO_EN("TRUE"),
.PMA_CH3_REG_PMA_TEST_SEL(0),
.PMA_CH3_REG_OOB_COMWAKE_GAP_MIN(3),
.PMA_CH3_REG_OOB_COMWAKE_GAP_MAX(11),
.PMA_CH3_REG_OOB_COMINIT_GAP_MIN(15),
.PMA_CH3_REG_OOB_COMINIT_GAP_MAX(35),
.PMA_CH3_REG_RX_PIBUF_IC_TX(1),
.PMA_CH3_REG_COMWAKE_STATUS_CLEAR(0),
.PMA_CH3_REG_COMINIT_STATUS_CLEAR(0),
.PMA_CH3_REG_RX_SYNC_RST_N_EN("FALSE"),
.PMA_CH3_REG_RX_SYNC_RST_N("TRUE"),
.PMA_CH3_REG_RX_SATA_COMINIT_OW("DISABLE"),
.PMA_CH3_REG_RX_SATA_COMINIT("FALSE"),
.PMA_CH3_REG_RX_SATA_COMWAKE_OW("DISABLE"),
.PMA_CH3_REG_RX_SATA_COMWAKE("FALSE"),
.PMA_CH3_REG_RX_DCC_DISABLE("ENABLE"),
.PMA_CH3_REG_TX_DCC_DISABLE("ENABLE"),
.PMA_CH3_REG_RX_SLIP_SEL_EN("FALSE"),
.PMA_CH3_REG_RX_SLIP_SEL(0),
.PMA_CH3_REG_RX_SLIP_EN("FALSE"),
.PMA_CH3_REG_RX_SIGDET_STATUS_SEL(5),
.PMA_CH3_REG_RX_SIGDET_FSM_RST_N("TRUE"),
.PMA_CH3_REG_RX_SIGDET_STATUS_OW("DISABLE"),
```

```
.PMA_CH3_REG_RX_SIGDET_STATUS("FALSE"),
.PMA_CH3_REG_RX_SIGDET_VTH("50MV"),
.PMA_CH3_REG_RX_SIGDET_GRM(0),
.PMA_CH3_REG_RX_SIGDET_PULSE_EXT("DISABLE"),
.PMA_CH3_REG_RX_SIGDET_CH2_SEL(0),
.PMA_CH3_REG_RX_SIGDET_CH2_CHK_WINDOW(3),
.PMA_CH3_REG_RX_SIGDET_CHK_WINDOW_EN("TRUE"),
.PMA_CH3_REG_RX_SIGDET_NOSIG_COUNT_SETTING(4),
.PMA_CH3_REG_RX_SIGDET_OOB_DET_COUNT_VAL(0),
.PMA_CH3_REG_SLIP_FIFO_INV_EN("FALSE"),
.PMA_CH3_REG_SLIP_FIFO_INV("POS_EDGE"),
.PMA_CH3_REG_RX_SIGDET_4OOB_DET_SEL(7),
.PMA_CH3_REG_RX_SIGDET_IC_I(10),
.PMA_CH3_REG_RX_OOB_DETECTOR_RESET_N_OW("DISABLE"),
.PMA_CH3_REG_RX_OOB_DETECTOR_RESET_N("FALSE"),
.PMA_CH3_REG_RX_OOB_DETECTOR_PD_OW("DISABLE"),
.PMA_CH3_REG_RX_OOB_DETECTOR_PD("ON"),
.PMA_CH3_REG_RX_TERM_CM_CTRL("5DIV7"),
.PMA_CH3_REG_TX_PD("ON"),
.PMA_CH3_REG_TX_PD_OW("DISABLE"),
.PMA_CH3_REG_TX_CLKPATH_PD("ON"),
.PMA_CH3_REG_TX_CLKPATH_PD_OW("DISABLE"),
.PMA_CH3_REG_TX_BEACON_TIMER_SEL(0),
.PMA_CH3_REG_TX_RXDET_REQ_OW("DISABLE"),
.PMA_CH3_REG_TX_RXDET_REQ("FALSE"),
.PMA_CH3_REG_TX_BEACON_EN_OW("DISABLE"),
.PMA_CH3_REG_TX_BEACON_EN("FALSE"),
.PMA_CH3_REG_TX_EI_EN_OW("DISABLE"),
.PMA_CH3_REG_TX_EI_EN("FALSE"),
.PMA_CH3_REG_TX_RES_CAL_EN("FALSE"),
.PMA_CH3_REG_TX_RES_CAL(51),
.PMA_CH3_REG_TX_BIAS_CAL_EN("FALSE"),
.PMA_CH3_REG_TX_BIAS_CTRL(48),
.PMA_CH3_REG_TX_RXDET_TIMER_SEL("12CYCLE"),
```

```
.PMA_CH3_REG_TX_SYNC_OW("DISABLE"),
.PMA_CH3_REG_TX_SYNC("DISABLE"),
.PMA_CH3_REG_TX_PD_POST("OFF"),
.PMA_CH3_REG_TX_PD_POST_OW("DISABLE"),
.PMA_CH3_REG_TX_RESET_N_OW("DISABLE"),
.PMA_CH3_REG_TX_RESET_N("TRUE"),
.PMA_CH3_REG_TX_DCC_RESET_N_OW("DISABLE"),
.PMA_CH3_REG_TX_DCC_RESET_N("TRUE"),
.PMA_CH3_REG_TX_BUSWIDTH_OW("DISABLE"),
.PMA_CH3_REG_TX_BUSWIDTH("20BIT"),
.PMA_CH3_REG_PLL_READY_OW("DISABLE"),
.PMA_CH3_REG_PLL_READY("TRUE"),
.PMA_CH3_REG_TX_PCLK_SW_OW("DISABLE"),
.PMA_CH3_REG_TX_PCLK_SW("TRUE"),
.PMA_CH3_REG_EI_PCLK_DELAY_SEL(0),
.PMA_CH3_REG_TX_DRV01_DAC0(0),
.PMA_CH3_REG_TX_DRV01_DAC1(10),
.PMA_CH3_REG_TX_DRV01_DAC2(16),
.PMA_CH3_REG_TX_DRV00_DAC0(63),
.PMA_CH3_REG_TX_DRV00_DAC1(53),
.PMA_CH3_REG_TX_DRV00_DAC2(48),
.PMA_CH3_REG_TX_AMP0(8),
.PMA_CH3_REG_TX_AMP1(16),
.PMA_CH3_REG_TX_AMP2(32),
.PMA_CH3_REG_TX_AMP3(48),
.PMA_CH3_REG_TX_AMP4(56),
.PMA_CH3_REG_TX_MARGIN(0),
.PMA_CH3_REG_TX_MARGIN_OW("DISABLE"),
.PMA_CH3_REG_TX_DEEMP(0),
.PMA_CH3_REG_TX_DEEMP_OW("DISABLE"),
.PMA_CH3_REG_TX_SWING("FALSE"),
.PMA_CH3_REG_TX_SWING_OW("DISABLE"),
.PMA_CH3_REG_TX_RXDET_THRESHOLD("100MV"),
.PMA_CH3_REG_TX_BEACON_OSC_CTRL(4),
```

```
.PMA_CH3_REG_TX_PREDRV_DAC(1),  
.PMA_CH3_REG_TX_PREDRV_CM_CTRL(1),  
.PMA_CH3_REG_TX_RX2RX_SLPBACK_EN("FALSE"),  
.PMA_CH3_REG_TX_PCLK_EDGE_SEL("POS_EDGE"),  
.PMA_CH3_REG_TX_RXDET_STATUS_OW("DISABLE"),  
.PMA_CH3_REG_TX_RXDET_STATUS("TRUE"),  
.PMA_CH3_REG_TX_PRBS_GEN_EN("FALSE"),  
.PMA_CH3_REG_TX_PRBS_GEN_WIDTH_SEL("20BIT"),  
.PMA_CH3_REG_TX_PRBS_SEL("PRBS7"),  
.PMA_CH3_REG_TX_UDP_DATA(256773),  
.PMA_CH3_REG_TX_FIFO_RST_N("FALSE"),  
.PMA_CH3_REG_TX_FIFO_WP_CTRL(2),  
.PMA_CH3_REG_TX_FIFO_EN("FALSE"),  
.PMA_CH3_REG_TX_DATA_MUX_SEL(0),  
.PMA_CH3_REG_TX_ERR_INSERT("FALSE"),  
.PMA_CH3_REG_TX_SATA_EN("FALSE"),  
.PMA_CH3_REG_RATE_CHANGE_TXPCLK_ON_OW("DISABLE"),  
.PMA_CH3_REG_RATE_CHANGE_TXPCLK_ON("ENABLE"),  
.PMA_CH3_REG_TX_PULLUP_DAC0(8),  
.PMA_CH3_REG_TX_PULLUP_DAC1(8),  
.PMA_CH3_REG_TX_PULLUP_DAC2(8),  
.PMA_CH3_REG_TX_PULLUP_DAC3(8),  
.PMA_CH3_REG_TX_OOB_DELAY_SEL(0),  
.PMA_CH3_REG_TX_POLARITY("NORMAL"),  
.PMA_CH3_REG_TX_SLPBK_AMP(1),  
.PMA_CH3_REG_TX_LS_MODE_EN("FALSE"),  
.PMA_CH3_REG_TX_JTAG_MODE_EN_OW("DISABLE"),  
.PMA_CH3_REG_TX_JTAG_MODE_EN("FALSE"),  
.PMA_CH3_REG_RX_JTAG_MODE_EN_OW("DISABLE"),  
.PMA_CH3_REG_RX_JTAG_MODE_EN("FALSE"),  
.PMA_CH3_REG_RX_JTAG_OE("DISABLE"),  
.PMA_CH3_REG_RX_ACJTAG_VHYSTSE(0),  
.PMA_CH3_REG_TX_FBCLK_FAR_EN("FALSE"),  
.PMA_CH3_REG_RX_TERM_MODE_CTRL(6),
```

```
.PMA_CH3_REG_PLPBK_TXPCLK_EN("TRUE"),
.PMA_CH3_REG_RX_609_600(0),
.PMA_CH3_REG_RX_CDR_617_610(0),
.PMA_CH3_REG_RX_CDR_623_618(0),
.PMA_CH3_REG_RX_631_624(0),
.PMA_CH3_REG_RX_639_632(0),
.PMA_CH3_REG_RX_647_640(0),
.PMA_CH3_REG_RX_655_648(0),
.PMA_CH3_REG_RX_659_656(0),
.PMA_CH3_CFG_LANE_POWERUP("OFF"),
.PMA_CH3_CFG_PMA_POR_N("FALSE"),
.PMA_CH3_CFG_RX_LANE_POWERUP("OFF"),
.PMA_CH3_CFG_RX_PMA_RSTN("FALSE"),
.PMA_CH3_CFG_TX_LANE_POWERUP("OFF"),
.PMA_CH3_CFG_TX_PMA_RSTN("FALSE"),
.PMA_CH3_CFG_CTLE_ADP_RSTN("TRUE"),
.PMA_CH3_REG_RESERVED_48_45(0),
.PMA_CH3_REG_RESERVED_69(0),
.PMA_CH3_REG_RESERVED_77_76(0),
.PMA_CH3_REG_RESERVED_171_164(0),
.PMA_CH3_REG_RESERVED_175_172(0),
.PMA_CH3_REG_RESERVED_190(0),
.PMA_CH3_REG_RESERVED_233_232(0),
.PMA_CH3_REG_RESERVED_235_234(0),
.PMA_CH3_REG_RESERVED_241_240(0),
.PMA_CH3_REG_RESERVED_285_283(0),
.PMA_CH3_REG_RESERVED_286(0),
.PMA_CH3_REG_RESERVED_295(0),
.PMA_CH3_REG_RESERVED_298(0),
.PMA_CH3_REG_RESERVED_332_325(0),
.PMA_CH3_REG_RESERVED_340_333(0),
.PMA_CH3_REG_RESERVED_348_341(0),
.PMA_CH3_REG_RESERVED_354_349(0),
.PMA_CH3_REG_RESERVED_373(0),
```

```
.PMA_CH3_REG_RESERVED_376(0),  
.PMA_CH3_REG_RESERVED_452(0),  
.PMA_CH3_REG_RESERVED_502_499(0),  
.PMA_CH3_REG_RESERVED_506_505(0),  
.PMA_CH3_REG_RESERVED_550_549(0),  
.PMA_CH3_REG_RESERVED_556_552(0),  
.PMA_PLL0_REG_PLL_POWERDOWN_OW("DISABLE"),  
.PMA_PLL0_REG_PLL_POWERDOWN("ON"),  
.PMA_PLL0_REG_PLL_RESET_N_OW("DISABLE"),  
.PMA_PLL0_REG_PLL_RESET_N("TRUE"),  
.PMA_PLL0_REG_PLL_READY_OW("DISABLE"),  
.PMA_PLL0_REG_PLL_READY("FALSE"),  
.PMA_PLL0_REG_LANE_SYNC_OW("DISABLE"),  
.PMA_PLL0_REG_LANE_SYNC("FALSE"),  
.PMA_PLL0_REG_LOCKDET_REPEAT("DISABLE"),  
.PMA_PLL0_REG_RESCAL_I_CODE_PMA("DISABLE"),  
.PMA_PLL0_REG_RESCAL_RESET_N_OW("DISABLE"),  
.PMA_PLL0_REG_RESCAL_RESET_N("FALSE"),  
.PMA_PLL0_REG_RESCAL_DONE_OW("DISABLE"),  
.PMA_PLL0_REG_RESCAL_DONE("FALSE"),  
.PMA_PLL0_REG_RESCAL_CODE_OW("DISABLE"),  
.PMA_PLL0_REG_LDO_VREF_SEL(2),  
.PMA_PLL0_REG_BIAS_VCOREP_C(1),  
.PMA_PLL0_REG_RESCAL_I_CODE(32),  
.PMA_PLL0_REG_RESCAL_ONCHIP_SMALL_OW("DISABLE"),  
.PMA_PLL0_REG_RESCAL_ONCHIP_SMALL(0),  
.PMA_PLL0_REG_JTAG_OE("DISABLE"),  
.PMA_PLL0_REG_JTAG_AC_MODE("DISABLE"),  
.PMA_PLL0_REG_JTAG_VHYSTSEL(0),  
.PMA_PLL0_REG_PLL_LOCKDET_EN_OW("DISABLE"),  
.PMA_PLL0_REG_PLL_LOCKDET_EN("FALSE"),  
.PMA_PLL0_REG_PLL_LOCKDET_RESET_N_OW("DISABLE"),  
.PMA_PLL0_REG_PLL_LOCKDET_RESET_N("FALSE"),  
.PMA_PLL0_REG_PLL_LOCKED_OW("DISABLE"),
```

```
.PMA_PLL0_REG_PLL_LOCKED("FALSE"),
.PMA_PLL0_REG_PLL_LOCKED_STICKY_CLEAR("FALSE"),
.PMA_PLL0_REG_PLL_UNLOCKED_STICKY_CLEAR("FALSE"),
.PMA_PLL0_REG_NOFBCLK_STICKY_CLEAR("FALSE"),
.PMA_PLL0_REG_PLL_LOCKDET_REFCT(7),
.PMA_PLL0_REG_PLL_LOCKDET_FBCT(7),
.PMA_PLL0_REG_PLL_LOCKDET_LOCKCT(4),
.PMA_PLL0_REG_PLL_LOCKDET_ITER(3),
.PMA_PLL0_REG_PLL_UNLOCKDET_ITER(2),
.PMA_PLL0_REG_PD_VCO("ON"),
.PMA_PLL0_REG_FBCLK_TEST_EN("FALSE"),
.PMA_PLL0_REG_REFCLK_TEST_EN("FALSE"),
.PMA_PLL0_REG_TEST_SEL(0),
.PMA_PLL0_REG_TEST_V_EN("FALSE"),
.PMA_PLL0_REG_TEST_SIG_HALF_EN("FALSE"),
.PMA_PLL0_REG_TEST_FSM(0),
.PMA_PLL0_REG_REFCLK_OUT_PD("OFF"),
.PMA_PLL0_REG_BGR_STARTUP_EN("FALSE"),
.PMA_PLL0_REG_BGR_STARTUP("FALSE"),
.PMA_PLL0_REG_PD_BGR("ON"),
.PMA_PLL0_REG_REFCLK_TERM_VCM_EN("TRUE"),
.PMA_PLL0_REG_FBDIVA_5_EN("TRUE"),
.PMA_PLL0_REG_FBDIVB(1),
.PMA_PLL0_REG_RESET_N_PFDQP_OW("DISABLE"),
.PMA_PLL0_REG_RESET_N_PFDQP("FALSE"),
.PMA_PLL0_REG_QPCURRENT(12),
.PMA_PLL0_REG_VC_FORCE_EN("FALSE"),
.PMA_PLL0_REG_VCRESET_C_RING(24),
.PMA_PLL0_REG_LPF_R_C(0),
.PMA_PLL0_REG_LPF_TR_C(2),
.PMA_PLL0_REG_PD_BIAS("ON"),
.PMA_PLL0_REGICTRL_PLL(1),
.PMA_PLL0_REG_BIAS_QP(1),
.PMA_PLL0_REG_BIAS_LANE_SYNC(1),
```

```
.PMA_PLL0_REG_BIAS_CLKBUFS1(1),  
.PMA_PLL0_REG_TXPCLK_SEL(0),  
.PMA_PLL0_REG_BIAS_CLKBUFS3(1),  
.PMA_PLL0_REG_LANE_SYNC_EN("FALSE"),  
.PMA_PLL0_REG_LANE_SYNC_EN_OW("DISABLE"),  
.PMA_PLL0_REG_BIAS_D2S(1),  
.PMA_PLL0_REG_BIAS_REFD2S_C(1),  
.PMA_PLL0_REG_BIAS_VCRST_C(1),  
.PMA_PLL0_REG_BIAS_REFBUF_C(1),  
.PMA_PLL0_REG_CLKBUFS1_C(1),  
.PMA_PLL0_REG_CLKBUFS2_C(6),  
.PMA_PLL0_REG_CLKBUFS3_C(6),  
.PMA_PLL0_REG_CLKBUFS4_C(1),  
.PMA_PLL0_REG_PLL_REFCLK_CML_SEL(0),  
.PMA_PLL0_REG_REFCLK_SEL("FALSE"),  
.PMA_PLL0_REG_RESCAL_R_CODE_SIGN("TRUE"),  
.PMA_PLL0_REG_PLL_UNLOCKED_OW("DISABLE"),  
.PMA_PLL0_REG_PLL_UNLOCKED("FALSE"),  
.PMA_PLL0_REG_PLL_LOCKDET_MODE("FALSE"),  
.PMA_PLL0_REG_PLL_CLKBUF_PD_LEFT("ON"),  
.PMA_PLL0_REG_PLL_CLKBUF_PD_RIGHT("ON"),  
.PMA_PLL0_REG_RESCAL_EN("FALSE"),  
.PMA_PLL0_REG_RESCAL_I_CODE_VAL(0),  
.PMA_PLL0_REG_RESCAL_I_CODE_OW(0),  
.PMA_PLL0_REG_RESCAL_ITER_VALID_SEL(0),  
.PMA_PLL0_REG_RESCAL_WAIT_SEL(0),  
.PMA_PLL0_REG_I_CTRL_MAX(45),  
.PMA_PLL0_REG_I_CTRL_MIN(19),  
.PMA_PLL0_REG_RESERVED_167_160(0),  
.PMA_PLL0_REG_RESERVED_175_168(0),  
.PMA_PLL0_REG_RESERVED_183_176(0),  
.PMA_PLL0_REG_RESERVED_191_184(0),  
.PARM_CFG_HSST_RSTN("FALSE"),  
.PARM_PLL0_POWERUP("OFF"),
```

```
.PARM_PLL0_RSTN("FALSE"),
.PMA_PLL1_REG_PLL_POWERDOWN_OW("DISABLE"),
.PMA_PLL1_REG_PLL_POWERDOWN("ON"),
.PMA_PLL1_REG_PLL_RESET_N_OW("DISABLE"),
.PMA_PLL1_REG_PLL_RESET_N("TRUE"),
.PMA_PLL1_REG_PLL_READY_OW("DISABLE"),
.PMA_PLL1_REG_PLL_READY("FALSE"),
.PMA_PLL1_REG_LANE_SYNC_OW("DISABLE"),
.PMA_PLL1_REG_LANE_SYNC("FALSE"),
.PMA_PLL1_REG_LOCKDET_REPEAT("DISABLE"),
.PMA_PLL1_REG_RESCAL_I_CODE_PMA("DISABLE"),
.PMA_PLL1_REG_RESCAL_RESET_N_OW("DISABLE"),
.PMA_PLL1_REG_RESCAL_RESET_N("FALSE"),
.PMA_PLL1_REG_RESCAL_DONE_OW("DISABLE"),
.PMA_PLL1_REG_RESCAL_DONE("FALSE"),
.PMA_PLL1_REG_RESCAL_CODE_OW("DISABLE"),
.PMA_PLL1_REG_LDO_VREF_SEL(2),
.PMA_PLL1_REG_BIAS_VCOREP_C(1),
.PMA_PLL1_REG_RESCAL_I_CODE(32),
.PMA_PLL1_REG_RESCAL_ONCHIP_SMALL_OW("DISABLE"),
.PMA_PLL1_REG_RESCAL_ONCHIP_SMALL(0),
.PMA_PLL1_REG_JTAG_OE("DISABLE"),
.PMA_PLL1_REG_JTAG_AC_MODE("DISABLE"),
.PMA_PLL1_REG_JTAG_VHYSTSEL(0),
.PMA_PLL1_REG_PLL_LOCKDET_EN_OW("DISABLE"),
.PMA_PLL1_REG_PLL_LOCKDET_EN("FALSE"),
.PMA_PLL1_REG_PLL_LOCKDET_RESET_N_OW("DISABLE"),
.PMA_PLL1_REG_PLL_LOCKDET_RESET_N("FALSE"),
.PMA_PLL1_REG_PLL_LOCKED_OW("DISABLE"),
.PMA_PLL1_REG_PLL_LOCKED("FALSE"),
.PMA_PLL1_REG_PLL_LOCKED_STICKY_CLEAR("FALSE"),
.PMA_PLL1_REG_PLL_UNLOCKED_STICKY_CLEAR("FALSE"),
.PMA_PLL1_REG_NOFBCLK_STICKY_CLEAR("FALSE"),
.PMA_PLL1_REG_PLL_LOCKDET_REFCT(7),
```

```
.PMA_PLL1_REG_PLL_LOCKDET_FBCT(7),  
.PMA_PLL1_REG_PLL_LOCKDET_LOCKCT(4),  
.PMA_PLL1_REG_PLL_LOCKDET_ITER(3),  
.PMA_PLL1_REG_PLL_UNLOCKDET_ITER(2),  
.PMA_PLL1_REG_PD_VCO("ON"),  
.PMA_PLL1_REG_FBCLOCK_TEST_EN("FALSE"),  
.PMA_PLL1_REG_REFCLK_TEST_EN("FALSE"),  
.PMA_PLL1_REG_TEST_SEL(0),  
.PMA_PLL1_REG_TEST_V_EN("FALSE"),  
.PMA_PLL1_REG_TEST_SIG_HALF_EN("FALSE"),  
.PMA_PLL1_REG_TEST_FSM(0),  
.PMA_PLL1_REG_REFCLK_OUT_PD("OFF"),  
.PMA_PLL1_REG_BGR_STARTUP_EN("FALSE"),  
.PMA_PLL1_REG_BGR_STARTUP("FALSE"),  
.PMA_PLL1_REG_PD_BGR("ON"),  
.PMA_PLL1_REG_REFCLK_TERM_VCM_EN("TRUE"),  
.PMA_PLL1_REG_FBDIVA_5_EN("TRUE"),  
.PMA_PLL1_REG_FBDIVB(1),  
.PMA_PLL1_REG_RESET_N_PFDQP_OW("DISABLE"),  
.PMA_PLL1_REG_RESET_N_PFDQP("FALSE"),  
.PMA_PLL1_REG_QPCURRENT(12),  
.PMA_PLL1_REG_VC_FORCE_EN("FALSE"),  
.PMA_PLL1_REG_VCRESET_C_RING(24),  
.PMA_PLL1_REG_LPF_R_C(0),  
.PMA_PLL1_REG_LPF_TR_C(2),  
.PMA_PLL1_REG_PD_BIAS("ON"),  
.PMA_PLL1_REG_ICTRL_PLL(1),  
.PMA_PLL1_REG_BIAS_QP(1),  
.PMA_PLL1_REG_BIAS_LANE_SYNC(1),  
.PMA_PLL1_REG_BIAS_CLKBUFS1(1),  
.PMA_PLL1_REG_TXPCLK_SEL(0),  
.PMA_PLL1_REG_BIAS_CLKBUFS3(1),  
.PMA_PLL1_REG_LANE_SYNC_EN("FALSE"),  
.PMA_PLL1_REG_LANE_SYNC_EN_OW("DISABLE"),
```

```

.PMA_PLL1_REG_BIAS_D2S(1),
.PMA_PLL1_REG_BIAS_REFD2S_C(1),
.PMA_PLL1_REG_BIAS_VCRST_C(1),
.PMA_PLL1_REG_BIAS_REFBUF_C(1),
.PMA_PLL1_REG_CLKBUFS1_C(1),
.PMA_PLL1_REG_CLKBUFS2_C(6),
.PMA_PLL1_REG_CLKBUFS3_C(6),
.PMA_PLL1_REG_CLKBUFS4_C(1),
.PMA_PLL1_REG_PLL_REFCLK_CML_SEL(0),
.PMA_PLL1_REG_REFCLK_SEL("FALSE"),
.PMA_PLL1_REG_RESCAL_R_CODE_SIGN("TRUE"),
.PMA_PLL1_REG_PLL_UNLOCKED_OW("DISABLE"),
.PMA_PLL1_REG_PLL_UNLOCKED("FALSE"),
.PMA_PLL1_REG_PLL_LOCKDET_MODE("FALSE"),
.PMA_PLL1_REG_PLL_CLKBUF_PD_LEFT("ON"),
.PMA_PLL1_REG_PLL_CLKBUF_PD_RIGHT("ON"),
.PMA_PLL1_REG_RESCAL_EN("FALSE"),
.PMA_PLL1_REG_RESCAL_I_CODE_VAL(0),
.PMA_PLL1_REG_RESCAL_I_CODE_OW(0),
.PMA_PLL1_REG_RESCAL_ITER_VALID_SEL(0),
.PMA_PLL1_REG_RESCAL_WAIT_SEL(0),
.PMA_PLL1_REG_I_CTRL_MAX(45),
.PMA_PLL1_REG_I_CTRL_MIN(19),
.PMA_PLL1_REG_RESERVED_167_160(0),
.PMA_PLL1_REG_RESERVED_175_168(0),
.PMA_PLL1_REG_RESERVED_183_176(0),
.PMA_PLL1_REG_RESERVED_191_184(0),
.PARM_PLL1_POWERUP("OFF"),
.PARM_PLL1_RSTN("FALSE"),
.PARM_GRSN_DIS("FALSE"),
.PARM_CFG_RSTN("FALSE")

) GTP_HSST_E2_inst (
    .P_CA_ALIGN_RX(),           // OUTPUT[3:0]
    .P_CA_ALIGN_TX(),           // OUTPUT[3:0]

```

```

.P_CFG_RDATA(),           // OUTPUT[7:0]
.P_PCS_LSM_SYNCED(),      // OUTPUT[3:0]
.P_PCS_RX_MCB_STATUS(),   // OUTPUT[3:0]
.P_RCLK2FABRIC(),         // OUTPUT[3:0]
.P_RDATA_0(),              // OUTPUT[46:0]
.P_RDATA_1(),              // OUTPUT[46:0]
.P_RDATA_2(),              // OUTPUT[46:0]
.P_RDATA_3(),              // OUTPUT[46:0]
.P_RESCAL_I_CODE_O(),      // OUTPUT[5:0]
.P_TCLK2FABRIC(),          // OUTPUT[3:0]
.P_TEST_STATUS_0(),         // OUTPUT[19:0]
.P_TEST_STATUS_1(),         // OUTPUT[19:0]
.P_TEST_STATUS_2(),         // OUTPUT[19:0]
.P_TEST_STATUS_3(),         // OUTPUT[19:0]
.P_CEB_ADETECT_EN(),        // INPUT[3:0]
.P_CFG_ADDR(),              // INPUT[15:0]
.P_CFG_WDATA(),             // INPUT[7:0]
.P_CIM_CLK_ALIGNER_RX0(),   // INPUT[7:0]
.P_CIM_CLK_ALIGNER_RX1(),   // INPUT[7:0]
.P_CIM_CLK_ALIGNER_RX2(),   // INPUT[7:0]
.P_CIM_CLK_ALIGNER_RX3(),   // INPUT[7:0]
.P_CIM_CLK_ALIGNER_TX0(),   // INPUT[7:0]
.P_CIM_CLK_ALIGNER_TX1(),   // INPUT[7:0]
.P_CIM_CLK_ALIGNER_TX2(),   // INPUT[7:0]
.P_CIM_CLK_ALIGNER_TX3(),   // INPUT[7:0]
.P_PCS_FAREND_LOOP(),       // INPUT[3:0]
.P_PCS_MCB_EXT_EN(),        // INPUT[3:0]
.P_PCS_NEAREND_LOOP(),      // INPUT[3:0]
.P_PCS_WORD_ALIGN_EN(),      // INPUT[3:0]
.P_PMA_FAREND_PLOOP(),      // INPUT[3:0]
.P_PMA_NEAREND_PLOOP(),     // INPUT[3:0]
.P_PMA_NEAREND_SLOOP(),     // INPUT[3:0]
.P_RESCAL_I_CODE_I(),        // INPUT[5:0]
.P_RX_BUSWIDTH_0(),          // INPUT[2:0]

```

```

.P_RX_BUSWIDTH_1(),          // INPUT[2:0]
.P_RX_BUSWIDTH_2(),          // INPUT[2:0]
.P_RX_BUSWIDTH_3(),          // INPUT[2:0]
.P_RX_POLARITY_INVERT(),    // INPUT[3:0]
.P_RX_RATE_0(),              // INPUT[2:0]
.P_RX_RATE_1(),              // INPUT[2:0]
.P_RX_RATE_2(),              // INPUT[2:0]
.P_RX_RATE_3(),              // INPUT[2:0]
.P_TDATA_0(),                // INPUT[45:0]
.P_TDATA_1(),                // INPUT[45:0]
.P_TDATA_2(),                // INPUT[45:0]
.P_TDATA_3(),                // INPUT[45:0]
.P_TX_BUSWIDTH_0(),          // INPUT[2:0]
.P_TX_BUSWIDTH_1(),          // INPUT[2:0]
.P_TX_BUSWIDTH_2(),          // INPUT[2:0]
.P_TX_BUSWIDTH_3(),          // INPUT[2:0]
.P_TX_DEEMP_0(),             // INPUT[1:0]
.P_TX_DEEMP_1(),             // INPUT[1:0]
.P_TX_DEEMP_2(),             // INPUT[1:0]
.P_TX_DEEMP_3(),             // INPUT[1:0]
.P_TX_MARGIN_0(),            // INPUT[2:0]
.P_TX_MARGIN_1(),            // INPUT[2:0]
.P_TX_MARGIN_2(),            // INPUT[2:0]
.P_TX_MARGIN_3(),            // INPUT[2:0]
.P_TX_RATE_0(),               // INPUT[2:0]
.P_TX_RATE_1(),               // INPUT[2:0]
.P_TX_RATE_2(),               // INPUT[2:0]
.P_TX_RATE_3(),               // INPUT[2:0]
.P_CFG_INT(),                // OUTPUT
.P_CFG_READY(),              // OUTPUT
.P_PLL_READY_0(),             // OUTPUT
.P_PLL_READY_1(),             // OUTPUT
.P_PLL_TEST_0(),              // OUTPUT
.P_PLL_TEST_1(),              // OUTPUT

```

```

.P_REFCK2CORE_0(),           // OUTPUT
.P_REFCK2CORE_1(),           // OUTPUT
.P_RX_LS_DATA_0(),           // OUTPUT
.P_RX_LS_DATA_1(),           // OUTPUT
.P_RX_LS_DATA_2(),           // OUTPUT
.P_RX_LS_DATA_3(),           // OUTPUT
.P_RX_READY_0(),             // OUTPUT
.P_RX_READY_1(),             // OUTPUT
.P_RX_READY_2(),             // OUTPUT
.P_RX_READY_3(),             // OUTPUT
.P_RX_SATA_COMINIT_0(),      // OUTPUT
.P_RX_SATA_COMINIT_1(),      // OUTPUT
.P_RX_SATA_COMINIT_2(),      // OUTPUT
.P_RX_SATA_COMINIT_3(),      // OUTPUT
.P_RX_SATA_COMWAKE_0(),      // OUTPUT
.P_RX_SATA_COMWAKE_1(),      // OUTPUT
.P_RX_SATA_COMWAKE_2(),      // OUTPUT
.P_RX_SATA_COMWAKE_3(),      // OUTPUT
.P_RX_SIGDET_STATUS_0(),     // OUTPUT
.P_RX_SIGDET_STATUS_1(),     // OUTPUT
.P_RX_SIGDET_STATUS_2(),     // OUTPUT
.P_RX_SIGDET_STATUS_3(),     // OUTPUT
.P_TX_RXDET_STATUS_0(),      // OUTPUT
.P_TX_RXDET_STATUS_1(),      // OUTPUT
.P_TX_RXDET_STATUS_2(),      // OUTPUT
.P_TX_RXDET_STATUS_3(),      // OUTPUT
.P_TX_SDN0(),                // OUTPUT
.P_TX_SDN1(),                // OUTPUT
.P_TX_SDN2(),                // OUTPUT
.P_TX_SDN3(),                // OUTPUT
.P_TX_SDP0(),                // OUTPUT
.P_TX_SDP1(),                // OUTPUT
.P_TX_SDP2(),                // OUTPUT
.P_TX_SDP3(),                // OUTPUT

```

```
.P_CFG_CLK(),           // INPUT
.P_CFG_ENABLE(),         // INPUT
.P_CFG_PSEL(),          // INPUT
.P_CFG_RST(),           // INPUT
.P_CFG_WRITE(),          // INPUT
.P_CIM_DYN_DLY_SEL_RX0(), // INPUT
.P_CIM_DYN_DLY_SEL_RX1(), // INPUT
.P_CIM_DYN_DLY_SEL_RX2(), // INPUT
.P_CIM_DYN_DLY_SEL_RX3(), // INPUT
.P_CIM_DYN_DLY_SEL_TX0(), // INPUT
.P_CIM_DYN_DLY_SEL_TX1(), // INPUT
.P_CIM_DYN_DLY_SEL_TX2(), // INPUT
.P_CIM_DYN_DLY_SEL_TX3(), // INPUT
.P_CIM_START_ALIGN_RX0(), // INPUT
.P_CIM_START_ALIGN_RX1(), // INPUT
.P_CIM_START_ALIGN_RX2(), // INPUT
.P_CIM_START_ALIGN_RX3(), // INPUT
.P_CIM_START_ALIGN_TX0(), // INPUT
.P_CIM_START_ALIGN_TX1(), // INPUT
.P_CIM_START_ALIGN_TX2(), // INPUT
.P_CIM_START_ALIGN_TX3(), // INPUT
.P_CTLE_ADG_RST_0(),    // INPUT
.P_CTLE_ADG_RST_1(),    // INPUT
.P_CTLE_ADG_RST_2(),    // INPUT
.P_CTLE_ADG_RST_3(),    // INPUT
.P_HSST_RST(),           // INPUT
.P_LANE_PD_0(),          // INPUT
.P_LANE_PD_1(),          // INPUT
.P_LANE_PD_2(),          // INPUT
.P_LANE_PD_3(),          // INPUT
.P_LANE_RST_0(),          // INPUT
.P_LANE_RST_1(),          // INPUT
.P_LANE_RST_2(),          // INPUT
.P_LANE_RST_3(),          // INPUT
```

```

.P_LANE_SYNC_0(),           // INPUT
.P_LANE_SYNC_1(),           // INPUT
.P_LANE_SYNC_EN_0(),         // INPUT
.P_LANE_SYNC_EN_1(),         // INPUT
.P_PCS_CB_RST_0(),          // INPUT
.P_PCS_CB_RST_10(),         // INPUT
.P_PCS_CB_RST_20(),         // INPUT
.P_PCS_CB_RST_30(),         // INPUT
.P_PCS_RX_RST_00(),          // INPUT
.P_PCS_RX_RST_10(),          // INPUT
.P_PCS_RX_RST_20(),          // INPUT
.P_PCS_RX_RST_30(),          // INPUT
.P_PCS_TX_RST_00(),          // INPUT
.P_PCS_TX_RST_10(),          // INPUT
.P_PCS_TX_RST_20(),          // INPUT
.P_PCS_TX_RST_30(),          // INPUT
.P_PLLPOWERDOWN_00(),         // INPUT
.P_PLLPOWERDOWN_10(),         // INPUT
.P_PLL_REF_CLK_0(),          // INPUT
.P_PLL_REF_CLK_1(),          // INPUT
.P_PLL_RST_0(),              // INPUT
.P_PLL_RST_1(),              // INPUT
.P_RATE_CHANGE_TCLK_ON_0(), // INPUT
.P_RATE_CHANGE_TCLK_ON_1(), // INPUT
.P_REFCLKN_0(),              // INPUT
.P_REFCLKN_1(),              // INPUT
.P_REFCLKP_0(),              // INPUT
.P_REFCLKP_1(),              // INPUT
.P_RESCAL_RST_I(),            // INPUT
.P_RX0_CLK2_FR_CORE(),        // INPUT
.P_RX0_CLK_FR_CORE(),         // INPUT
.P_RX1_CLK2_FR_CORE(),        // INPUT
.P_RX1_CLK_FR_CORE(),         // INPUT
.P_RX2_CLK2_FR_CORE(),        // INPUT

```

```
.P_RX2_CLK_FR_CORE(),      // INPUT
.P_RX3_CLK2_FR_CORE(),     // INPUT
.P_RX3_CLK_FR_CORE(),     // INPUT
.P_RXGEAR_SLIP_0(),       // INPUT
.P_RXGEAR_SLIP_1(),       // INPUT
.P_RXGEAR_SLIP_2(),       // INPUT
.P_RXGEAR_SLIP_3(),       // INPUT
.P_RX_HIGHZ_0(),          // INPUT
.P_RX_HIGHZ_1(),          // INPUT
.P_RX_HIGHZ_2(),          // INPUT
.P_RX_HIGHZ_3(),          // INPUT
.P_RX_LANE_PD_0(),        // INPUT
.P_RX_LANE_PD_1(),        // INPUT
.P_RX_LANE_PD_2(),        // INPUT
.P_RX_LANE_PD_3(),        // INPUT
.P_RX_PMA_RST_0(),        // INPUT
.P_RX_PMA_RST_1(),        // INPUT
.P_RX_PMA_RST_2(),        // INPUT
.P_RX_PMA_RST_3(),        // INPUT
.P_RX_SDN0(),             // INPUT
.P_RX_SDN1(),             // INPUT
.P_RX_SDN2(),             // INPUT
.P_RX_SDN3(),             // INPUT
.P_RX_SDP0(),             // INPUT
.P_RX_SDP1(),             // INPUT
.P_RX_SDP2(),             // INPUT
.P_RX_SDP3(),             // INPUT
.P_TX0_CLK2_FR_CORE(),    // INPUT
.P_TX0_CLK_FR_CORE(),     // INPUT
.P_TX1_CLK2_FR_CORE(),    // INPUT
.P_TX1_CLK_FR_CORE(),     // INPUT
.P_TX2_CLK2_FR_CORE(),    // INPUT
.P_TX2_CLK_FR_CORE(),     // INPUT
.P_TX3_CLK2_FR_CORE(),    // INPUT
```

```

.P_TX3_CLK_FR_CORE(),           // INPUT
.P_TX_BEACON_EN_0(),           // INPUT
.P_TX_BEACON_EN_1(),           // INPUT
.P_TX_BEACON_EN_2(),           // INPUT
.P_TX_BEACON_EN_3(),           // INPUT
.P_TX_LANE_PD_0(),             // INPUT
.P_TX_LANE_PD_1(),             // INPUT
.P_TX_LANE_PD_2(),             // INPUT
.P_TX_LANE_PD_3(),             // INPUT
.P_TX_LS_DATA_0(),             // INPUT
.P_TX_LS_DATA_1(),             // INPUT
.P_TX_LS_DATA_2(),             // INPUT
.P_TX_LS_DATA_3(),             // INPUT
.P_TX_PMA_RST_0(),             // INPUT
.P_TX_PMA_RST_1(),             // INPUT
.P_TX_PMA_RST_2(),             // INPUT
.P_TX_PMA_RST_3(),             // INPUT
.P_TX_RXDET_REQ_0(),           // INPUT
.P_TX_RXDET_REQ_1(),           // INPUT
.P_TX_RXDET_REQ_2(),           // INPUT
.P_TX_RXDET_REQ_3(),           // INPUT
.P_TX_SWING_0(),               // INPUT
.P_TX_SWING_1(),               // INPUT
.P_TX_SWING_2(),               // INPUT
.P_TX_SWING_3()                // INPUT
);

```

## 10.6 Usage Instructions for GTP\_ADC\_E1

### 10.6.1 Supported Devices

Table 10-16 Device Models That Support GTP\_ADC\_E1

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Supported	Supported	Not supported	Not supported	Not supported

### 10.6.2 Description of Functionality

GTP\_ADC\_E1 is used to implement ADC functions. For detailed functionality, please refer to "***UG020009\_Logos Family FPGAs Analog-to-Digital Conversion (ADC) Module User Guide***". The structure block diagram is shown in the figure below.

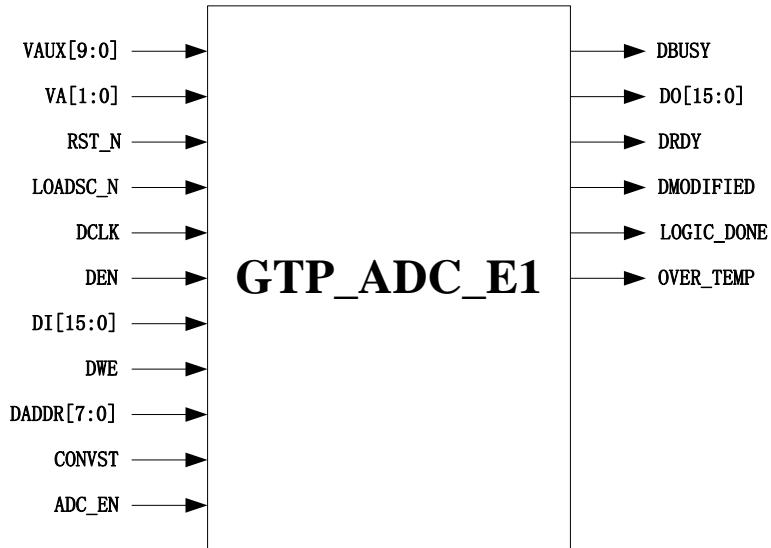


Figure 10-13 GTP\_ADC\_E1 Structure Block Diagram

### 10.6.3 Port Description

Table 10-17 GTP\_ADC\_E1 Port Description

Port	Direction	Description
VAUX[9:0]	I	Multiplexed pin analog signal input
VA[1:0]	I	Dedicated channel analog input
RST_N	I	ADC system reset signal (active-low)
LOADSC_N	I	ADC control register download static configuration value signal (active-low)
DCLK	I	DRP clock
DEN	I	Operation enable; initiates a read/write operation when active-high
DI[15:0]	I	Data input
DWE	I	Write enable, 1'b0: read operation; 1'b1: write operation;
DADDR[7:0]	I	Address bits
CONVST	I	Control signal for event driven mode
ADC_EN	I	ADC enable signal
DBUSY	O	JTAG DRP operation flag, indicating that JTAG is performing DRP operation via JDRP instruction
DO[15:0]	O	Data output
DRDY	O	Operation completion flag (active-high)
DMODIFIED	O	Control register modification flag, indicating that the control register has been written by JTAG DRP and the user has not

Port	Direction	Description
		yet performed DRP operation
LOGIC_DONE	O	ADC status register update signal
OVER_TEMP	O	ADC temperature alarm signal

#### 10.6.4 Paramater Description

Table 10-18 GTP\_ADC\_E1 Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
AVERAGE	string	"1", "16", "64", "256"	"1"	Multiple ADC output CODE averaging setting
CALIB	string	"NONE", "OFFSET", "OFFSET_GAIN"	"NONE"	ADC calibration setting "NONE": offset and gain calculation disable "OFFSET": offset calculation enable "OFFSET_GAIN": offset and gain calculation enable
REFERENCE	string	"INTERNAL", "EXTERNAL"	"INTERNAL"	ADC reference source setting "INTERNAL": internal reference source "EXTERNAL": external reference source
CALIB_REFERENCE	string	"INTERNAL", "EXTERNAL"	"INTERNAL"	ADC calibration reference source setting "INTERNAL": internal reference source "EXTERNAL": external reference source
FULL_SWING	string	"0.5V", "0.6V", "0.7V", "0.8V", "0.9V", "1.0V", "1.1V", "1.2V"	"0.5V"	ADC range setting
VCM	string	"0.8V", "0.9V", "1.0V", "1.1V", "1.2V", "1.3V", "1.4V", "1.5V"	"0.8V"	ADC internal comparator input common-mode voltage setting
DIVIDER	string	"2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16"	"2"	Input clock division setting
ADC_MODE	string	"DEFAULT", "SINGLE_PASS", "CONTINUE_SEQ", "SINGLE_CHANNEL"	"DEFAULT"	ADC Working Mode Selection "DEFAULT": default mode "SINGLE_PASS": single scan mode "CONTINUE_SEQ": continuous scan mode "SINGLE_CHANNEL": single-channel mode

Parameter Name	Parameter Type	Valid Values	Defaults	Description
EVENT_DRIVE	string	"FALSE", "TRUE"	"FALSE"	Sampling mode setting "FALSE": continuous sampling mode "TRUE": event-driven sampling mode
ADC_MODE_1MSPS	string	"FALSE", "TRUE"	"FALSE"	1MSPS mode setting "FALSE": mode disabled "TRUE": mode enabled
CLKSWITCH	string	"FALSE", "TRUE"	"FALSE"	Clock switching configuration "FALSE": select CLK_OSC "TRUE": select DCLK
INTERNAL_VOL_SEL	string	"VDD33", "VDD11", "VDDM"	"VDD33"	channel 13 signal selection
SINGLE_CH_SEL	string	"0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15"	"0"	Channel selection
SINGLE_CH_IN	string	"SINGLE_END", "UNIPOLAR", "BIPOLAR"	"SINGLE_END"	Input mode setting
SEQ_CH11_10_SEL	string	"NONE", "CH10", "CH11", "ALL"	"NONE"	Channels 11/10 scan setting "NONE": channels 11/10 not scanned "CH10": channel 10 scanned "CH11": channel 11 scanned "ALL": channels 11/10 scanned
SEQ_CH9_8_SEL	string	"NONE", "CH8", "CH9", "ALL"	"NONE"	Channels 9/8 setting "NONE": channels 9/8 not scanned "CH8": channel 8 scanned "CH9": channel 9 scanned "ALL": channels 9/8 scanned
SEQ_CH7_6_SEL	string	"NONE", "CH6", "CH7", "ALL"	"NONE"	Channels 7/ 6 setting "NONE": channels 7/ 6 not scanned "CH6": channel 6 scanned "CH7": channel 7 scanned "ALL": channels 7/ 6 scanned
SEQ_CH5_4_SEL	string	"NONE", "CH4", "CH5", "ALL"	"NONE"	Channels 5/ 4 setting "NONE": channels 5/ 4 not scanned "CH4": channel 4 scanned

Parameter Name	Parameter Type	Valid Values	Defaults	Description
				"CH5": channel 5 scanned "ALL": channels 5/ 4 scanned
SEQ_CH3_2_SEL	string	"NONE", "CH2", "CH3", "ALL"	"NONE"	Channels 3/ 2 setting "NONE": channels 3/ 2 not scanned "CH2": channel 2 scanned "CH3": channel 3 scanned "ALL": channels 3/ 2 scanned
SEQ_CH1_0_SEL	string	"NONE", "CH0", "CH1", "ALL"	"NONE"	Channels 1/0 scan setting "NONE": channels 1/0 not scanned "CH0": channel 0 scanned "CH1": channel 1 scanned "ALL": channels 1/0 scanned
SEQ_CH11_10_IN	string	"SINGLE-END", "UNIPOLAR", "BIPOLAR"	"SINGLE-END"	Channels 11/10 input mode setting
SEQ_CH9_8_IN	string	"SINGLE-END", "UNIPOLAR", "BIPOLAR"	"SINGLE-END"	Channel 9/8 input mode setting
SEQ_CH7_6_IN	string	"SINGLE-END", "UNIPOLAR", "BIPOLAR"	"SINGLE-END"	Channels 7/ 6 input mode setting
SEQ_CH5_4_IN	string	"SINGLE-END", "UNIPOLAR", "BIPOLAR"	"SINGLE-END"	Channels 5/ 4 input mode setting
SEQ_CH3_2_IN	string	"SINGLE-END", "UNIPOLAR", "BIPOLAR"	"SINGLE-END"	Channels 3/ 2 input mode setting
SEQ_CH1_0_IN	string	"SINGLE-END", "UNIPOLAR", "BIPOLAR"	"SINGLE-END"	Channels 1/0 input mode setting
TEMP_SENSOR_HIGH	integer	0~255	0	Temperature sense upper threshold
TEMP_SENSOR_LOW	integer	0~255	0	Temperature sense lower threshold
ADC_EN_ENABLE	string	"FALSE", "TRUE"	"FALSE"	ADC_EN port enable setting "FALSE": ADC_EN port input invalid "TRUE": ADC_EN port input valid

## 10.6.5 Instantiation template

```
GTP_ADC_E1 #(  
    .AVERAGE("1"),  
    .CALIB("NONE"),  
    .REFERENCE("INTERNAL"),  
    .CALIB_REFERENCE("INTERNAL"),  
    .FULL_SWING("0.5V"),  
    .VCM("0.8V"),  
    .DIVIDER("2"),  
    .ADC_MODE("DEFAULT"),  
    .EVENT_DRIVE("FALSE"),  
    .ADC_MODE_1MSPS("FALSE"),  
    .CLKSWITCH("FALSE"),  
    .INTERNAL_VOL_SEL("VDD33"),  
    .SINGLE_CH_SEL("0"),  
    .SINGLE_CH_IN("SINGLE_END"),  
    .SEQ_CH11_10_SEL("NONE"),  
    .SEQ_CH9_8_SEL("NONE"),  
    .SEQ_CH7_6_SEL("NONE"),  
    .SEQ_CH5_4_SEL("NONE"),  
    .SEQ_CH3_2_SEL("NONE"),  
    .SEQ_CH1_0_SEL("NONE"),  
    .SEQ_CH11_10_IN("SINGLE_END"),  
    .SEQ_CH9_8_IN("SINGLE_END"),  
    .SEQ_CH7_6_IN("SINGLE_END"),  
    .SEQ_CH5_4_IN("SINGLE_END"),  
    .SEQ_CH3_2_IN("SINGLE_END"),  
    .SEQ_CH1_0_IN("SINGLE_END"),  
    .TEMP_SENSOR_HIGH(0),  
    .TEMP_SENSOR_LOW(0),  
    .ADC_EN_ENABLE("FALSE")  
) GTP_ADC_E1 (  
    .DO(),           // OUTPUT[15:0]  
    .DADDR(),        // INPUT[7:0]
```

```

.DI(),          // INPUT[15:0]
.VA(),          // INPUT[1:0]
.VAUX(),        // INPUT[9:0]
.DBUSY(),       // OUTPUT
.DMODIFIED(),  // OUTPUT
.DRDY(),        // OUTPUT
.LOGIC_DONE(), // OUTPUT
.OVER_TEMP(),   // OUTPUT
.ADC_EN(),      // INPUT
.CONVST(),      // INPUT
.DCLK(),        // INPUT
.DEN(),         // INPUT
.DWE(),         // INPUT
.LOADSC_N(),   // INPUT
.RST_N()        // INPUT
);

```

## 10.7 Usage Instructions for GTP\_PCIEGEN2

### 10.7.1 Supported Devices

Table 10-19 Device Models That Support GTP\_PCIEGEN2

Device Model	PGL22G	PGL12G	PGL25G	PGL50H	PGL100H
Whether supports the GTP	Not supported	Not supported	Not supported	Supported	Supported

### 10.7.2 Description of Functionality

GTP\_PCIEGEN2 (Peripheral Component Interconnect Express) high-speed serial interface module is compatible with PCIE2.1 protocol. Key items include: support both Gen1 and Gen2 rates (2.5Gb/s or 5Gb/s); support EP and RC modes as well as x1, x2, and x4; TLP packet max\_payload can be flexibly configured to 128Bytes, 256 Bytes, 512 Bytes or 1024 Bytes; BAR registers are configurable; support three types of interrupt mechanisms: Legacy, MSI, and MSIX; Power Management, Lane Reversal, Lane Polarity Inversion are all achievable. A simplified block diagram of its structure is shown below:

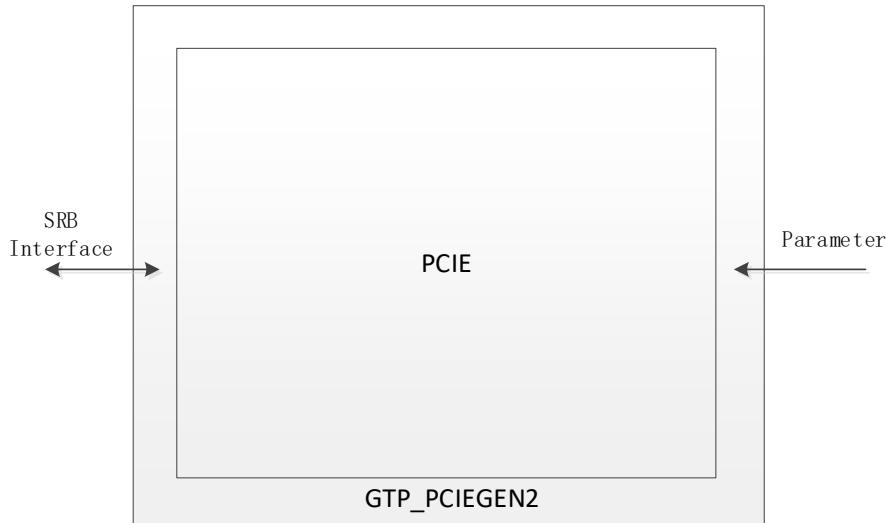


Figure 10-14 GTP\_ADC\_E1 Structure Block Diagram

### 10.7.3 Port Description

Table 10-20 GTP\_PCIEGEN2 Port Description

<b>Port</b>	<b>Direction</b>	<b>Width</b>	<b>Description</b>
PCLK	I	1	PIPE clock from external PHY Multiplexed as test_clk in DFT mode Gen1: 125Mhz Gen2: 250Mhz
PCLK_DIV2	I	1	pclk_div2 clock from external PHY, two divided from PCLK Multiplexed as test_clk in DFT mode Gen1: 62.5Mhz Gen2: 125Mhz
MEM_CLK	I	1	Used to receive the memory clock, which shares the same source with pclk.
BUTTON_RST	I	1	Button reset, Active-high
POWER_UP_RST	I	1	Warm reset, Active-high
PERST	I	1	Reset from Socket, Active-high
CORE_RST_N	O	1	Resets the controller, except for the PMC module, It is recommended that users reset users application logic together with the controller, Active-low
TRAINING_RST_N	O	1	Hot reset from upstream component, Actice low
APP_INIT_RST	I	1	Request from users application to send a hot reset to the upstream port; Active-high
PHY_RST_N	O	1	For phy rst, Active-low
DEVICE_TYPE	I	3	Device/port type. Indicates the specific type of this PCI Express function. 3'b000: PCI Express endpoint 3'b001: Legacy PCI Express endpoint 3'b100: Root port of PCI Express root complex

<b>Port</b>	<b>Direction</b>	<b>Width</b>	<b>Description</b>
RX_LANE_FLIP_EN	I	1	Performs manual lane reversal for receive lanes.
TX_LANE_FLIP_EN	I	1	Performs manual lane reversal for transmit lanes
APP_LTSSM_ENABLE	I	1	Driven low by users application after cold, warm or hot reset to hold the LTSSM in the Detect state until users application is ready for the link training to begin.
SMLH_LINK_UP	O	1	PHY Link up/down indicator
RDLH_LINK_UP	O	1	Data link layer up/down indicator
APP_REQ_RETRY_EN	I	1	Provides a capability to defer incoming configuration requests until initialization is complete
SMLH_LTSSM_STATE	O	5	Current state of the LTSSM
AXIS_MASTER_TVALID	O	1	Active high to indicate that the ctrl is driving a valid transfer
AXIS_MASTER_TREADY	I	1	Active high to indicate that the user can accept a transfer in the current cycle
AXIS_MASTER_TDATA	O	128	Primary payload provide to user logic
AXIS_MASTER_TKEEP	O	4	Ativie high to enable related DW in axis_master_tdata; axis_master_tkeep[3] control axis_master_tdata[127:96]; axis_master_tkeep[2] control axis_master_tdata[95:64]; axis_master_tkeep[1] control axis_master_tdata[63:32]; axis_master_tkeep[0] control axis_master_tdata[31:0]; Note: The difference from the AXIS standard is that here one bit corresponds to the enable of 4 bytes
AXIS_MASTER_TLAST	O	1	Active high to indicate last valid transfer of a packet from ctrl
AXIS_MASTER_TUSER	O	8	Sideband information transmitted alongside axis_master_tdata: 0: radm_trgt1_tlp_abort (Indicates to users application to drop the TLP because of malformed TLP on TRGT1, ECRC error, or completion lookup failures) 1: radm_trgt1_dllp_abort (Indicates to users application to drop the TLP on TRGT1 because of a Data Link Layer error such as LCRC or otherwise) 2: radm_trgt1_ecrc_err (Indicates to users application to drop the TLP because of an ECRC error in the received TLP on TRGT1) 3: radm_trgt1_cpl_last: Indicates the last completion TLP of a split completion transaction. [6:4]: radm_trgt1_in_membar_range[2:0] (Indicates which of the configured BARs contains the target address in the received TLP) 7: radm_trgt1_rom_in_range (Indicates that the target address in the received TLP in range of the expansion ROM)
TRGT1_RADM_PKT_HOLD	I	3	Halts the transfer of packets from individual queues. There is one bit of trgt1_radm_pkt_halt for each TLP type for each configured VC: Bit 0: Halt posted TLPs for VC0 Bit 1: Halt non-posted TLPs for VC0 Bit 2: Halt CPL TLPs for VC0
RADM_GRANT_TLP_TYPE	O	6	Indicates that a particular VC and type transaction has been granted to output from the receive queue. There is one bit for each TLP type for each configured VC: Bit [1:0]: Grant posted TLPs for VC0 Bit [3:2]: Grant non-posted TLPs for VC0 Bit [5:4]: Grant CPL TLPs for VC0

<b>Port</b>	<b>Direction</b>	<b>Width</b>	<b>Description</b>
AXIS_SLAVE0/1/2_TRE <sub>ADY</sub>	O	1	Active high to indicate that the ctrl can accept a transfer in the current cycle
AXIS_SLAVE0/1/2_TVA <sub>LID</sub>	I	1	Active high to indicate that the user is driving a valid transfer
AXIS_SLAVE0/1/2_TDA <sub>TA</sub>	I	128	Primary payload provide to ctrl
AXIS_SLAVE0/1/2_TLA <sub>ST</sub>	I	1	Active high to indicate last valid transfer of a packet to ctrl
AXIS_SLAVE0/1/2_TUS <sub>ER</sub>	I	1	Sideband information transmitted alongside axis_slave_tdata: Bit 0: client0/1/2_tlp_bad_eot, Indicates that the current TLP must be nullified.
DBI_ADDR	I	32	Address of the configuration register for the current DBI access [31:2] register address, must be dword-aligned [1] not used [0] DBI access direction,0: internal register 1: external register via ELBI
DBI_DIN	I	32	Write data bus to the selected configuration register
DBI_CS	I	1	Chip select input to access the CDM or ELBI
DBI_CS2	I	1	Additional chip select that enables writing to BAR mask registers
DBI_WR	I	4	Indicates the configuration register access type (read or write).
APP_DBI_RO_WR_DIS <sub>ABLE</sub>	I	1	DBI Read-only Write Disable 0: MISC_CONTROL_1_OFF.DBI_RO_WR_EN register field is read-write. 1: MISC_CONTROL_1_OFF.DBI_RO_WR_EN register field is forced to 0 and is read-only.
LBC_DBI_ACK	O	1	Indicates that the requested read or write operation to the selected configuration register is complete.
LBC_DBI_DOUT	O	32	Read data bus from the selected configuration register.
SEDI	I	1	Serial signal input, driven by slave
SEDI_ACK	I	1	Serial signal input ACK, driven by slave
SEDO	O	1	Serial signal output, driven by master
SEDO_EN	O	1	Serial signal output enable, driven by master
CFG_INT_DISABLE	O	1	When high a functions ability to generate INTx messages is Disabled
SYS_INT	I	1	When sys_int goes from low to high, the controller generates an Assert_INTx Message
INTA_GRT_MUX	O	1	EP: 0->1: The signal indicates that the controller sent an Assert_INTA Message to the upstream device. 1->0: The signal indicates that the controller sent an Deassert_INTA Message to the upstream device. RC: 0->1: The signal indicates that the controller received an Assert_INTA Message from the downstream device 1->0: The signal indicates that the controller received an Deassert_INTA Message from the downstream device
INTB_GRT_MUX	O	1	EP:Not used RC: 0->1: The signal indicates that the controller received an Assert_INTB Message from the downstream device

<b>Port</b>	<b>Direction</b>	<b>Width</b>	<b>Description</b>
			1->0: The signal indicates that the controller received an Deassert_INTB Message from the downstream device
INTC_GRT_MUX	O	1	EP:Not used RC: 0->1: The signal indicates that the controller received an Assert_INTC Message from the downstream device 1->0: The signal indicates that the controller received an Deassert_INTC Message from the downstream device
INTD_GRT_MUX	O	1	EP:Not used RC: 0->1: The signal indicates that the controller received an Assert_INTD Message from the downstream device 1->0: The signal indicates that the controller received an Deassert_INTD Message from the downstream device
VEN_MSI_REQ	I	1	Request from users application to send an MSI when MSI is enabled
VEN_MSI_TC	I	3	Traffic Class of the MSI request, valid when ven_msi_req is asserted
VEN_MSI_VECTOR	I	5	Used to modulate the lower five bits of the MSI Data register when multiple message mode is enabled
VEN_MSI_GRANT	O	1	One-cycle pulse that indicates that the controller has accepted the request to send an MSI
CFG_MSI_PENDING	I	32	Indication from application about which functions have a pending associated message
CFG_MSI_EN	O	1	Indicates that MSI is enabled (INTx message is not sent)
CFG_MSIX_EN	O	1	The MSI-X Enable bit of the MSI-X Control register in the MSI-X Capability structure
MSIX_ADDR	I	64	The address value for the MSI-X.
MSIX_DATA	I	32	The data value for the MSI-X.
CFG_MSIX_FUNC_MASK	O	1	The function Mask bit of the MSI-X Control register in the MSI-X Capability structure.
CFG_LINK_AUTO_BW_MUX	O	1	cfg_link_auto_bw_int when legacy int is used, cfg_link_auto_bw_msi when MSI or MSI-X is enabled. Only for RC
CFG_BW_MGT_MUX	O	1	cfg_bw_mgt_int when legacy int is used, cfg_bw_mgt_msi when MSI or MSI-X is enabled. Only for RC
CFG_PME_MUX	O	1	cfg_pme_int when legacy int is used, cfg_pme_msi when MSI or MSI-X is enabled.
CFG_AER_RC_ERR_MUX	O	1	cfg_aer_rc_err_int: when MSI/MSI-X is NOT enabled. Otherwise used as cfg_aer_rc_err_msi
RADM_PM_TURNOFF	O	1	One-clock-cycle pulse that indicates that the controller received a PME Turnoff message
RADM_MSG_UNLOCK	O	1	One-cycle pulse that indicates that the controller received an Unlock message
OUTBAND_PWRUP_CMD	I	1	Wake Up. Used by application logic to wake up the PMC state machine from a D1, D2 or D3 power state.
PM_XTLH_BLOCK_TLP	O	1	Indicates that users application must stop generating new outgoing request TLPs due to the current power management state
PM_STATUS	O	1	PME Status bit from the PMCSR
PM_DSTATE	O	3	The current power management D-state of the function
AUX_PM_EN	O	1	Auxiliary Power Enable bit in the Device Control register

<b>Port</b>	<b>Direction</b>	<b>Width</b>	<b>Description</b>
PM_PME_EN	O	1	PME Enable bit in the PMCSR
PM_LINKST_IN_L0S	O	1	Power management is in L0s state.
PM_LINKST_IN_L1	O	1	Power management is in L1 state.
PM_LINKST_IN_L2	O	1	Power management is in L2 state.
PM_LINKST_L2_EXIT	O	1	Power management is exiting L2 state
APP_REQ_ENTR_L1	I	1	Application request to Enter L1 ASPM state
APP_READY_ENTR_L2 3	I	1	Application Ready to Enter L23.
APP_REQ_EXIT_L1	I	1	Application request to Exit L1
APP_XFER_PENDING	I	1	Indicates that users application has transfers pending and prevents the controller from entering L1.
WAKE	O	1	Wake Up. Wake up from power management unit.
RADM_PM_PME	O	1	One-clock-cycle pulse that indicates that the controller received a PM_PME message
RADM_PM_TO_ACK	O	1	One-clock-cycle pulse that indicates that the controller received a PME_TO_Ack message
APPS_PM_XMT_TURN OFF	I	1	Request from users application to generate a PM_Turn_Off message.
APP_UNLOCK_MSG	I	1	Request from users application to generate an Unlock message
APPS_PM_XMT_PME	I	1	Wake Up. Used by application logic to wake up the PMC state machine from a D1, D2 or D3 power state
APP_CLK_PM_EN	I	1	Clock PM feature enabled by application.
PM_MASTER_STATE	O	5	Power management master FSM state
PM_SLAVE_STATE	O	5	Power management slave FSM state
SYS_AUX_PWR_DET	I	1	Auxiliary Power Detected. Used to report to the host software that auxiliary power (Vaux) is present
CFG_MAX_RD_REQ_SI ZE	O	3	The value of the Max_Read_Request_Size field in the Device Control register.
CFG_BUS_MASTER_E N	O	1	The state of the bus master enable bit in the PCI-compatible Command register.
CFG_MAX_PAYLOAD_ SIZE	O	3	The value of the Max_Payload_Size field in the Device Control register
CFG_EXT_TAG_EN	O	1	When enabled, controller supports up to 8-bit tag values.
CFG_RCB	O	1	The value of the RCB bit in the Link Control register.
CFG_MEM_SPACE_EN	O	1	The state of the Memory Space Enable bit in the PCI-compatible Command register.
CFG_PM_NO_SOFT_RS T	O	1	This is the value of the No Soft Reset bit in the Power Management Control and Status Register
CFG_CRS_SW_VIS_EN	O	1	Indicates the value of the CRS Software Visibility enable bit in the Root Control register
CFG_NO_SNOOP_EN	O	1	Contents of the "Enable No Snoop" field (PCIE_CAP_EN_NO_SNOOP) in the "Device Control and Status" register (DEVICE_CONTROL_DEVICE_STATUS) register
CFG_RELAX_ORDER_ EN	O	1	Contents of the "Enable Relaxed Ordering" field (PCIE_CAP_EN_REL_ORDER) in the "Device Control and Status" register (DEVICE_CONTROL_DEVICE_STATUS) register
CFG_TPH_REQ_EN	O	2	The 2-bit TPH Requester Enabled field of each TPH Requester Control register.

Port	Direction	Width	Description
CFG_PF_TPH_ST_MOD_E	O	3	Steering Tag Mode of Operation for Physical Function
CFG_PBUS_NUM	O	8	The primary bus number assigned to the function
CFG_PBUS_DEV_NUM	O	5	The device number assigned to the function
CFG_ATOMIC_REQ_EN	O	1	The AtomicOp Requester Enable field of the Device Control 2 register
CFG_ATOMIC_EGRESS_BLOCK	O	1	The AtomicOp Egress Blocking field of the Device Control 2 register
RBAR_CTRL_UPDATE	O	1	Indicates that a resizable BAR control register has been updated
APP_HDR_VALID	I	1	One-clock-cycle pulse indicating that the data app_hdr_log,
APP_HDR_LOG	I	128	The header of the TLP that contained the error indicated app_err_bus, valid when app_hdr_valid is asserted; Provide app_tlp_prfx_log before app_hdr_valid is asserted if needed
APP_ERR_BUS	I	13	The type of error that users application detected
APP_ERR_ADVISORY	I	1	Indicates that users application error is an advisory error.
CFG_SEND_COR_ERR_MUX	O	1	EP: cfg_send_cor_err Sent Correctable Error. RC: radm_correctable_err, One-clock-cycle pulse that indicates that the controller received an ERR_COR message.
CFG_SEND_NF_ERR_MUX	O	1	EP: radm_nonfatal_err, Sent Non-Fatal Error RC: One-clock-cycle pulse that indicates that the controller received an ERR_NONFATAL message
CFG_SEND_F_ERR_MUX	O	1	EP: radm_fatal_err, Sent Fatal Error RC: One-clock-cycle pulse that indicates that the controller received an ERR_FATAL message.
CFG_SYS_ERR_RC	O	1	System error detected
RADM_CPL_TIMEOUT	O	1	Indicates that the completion TLP for a request has not been received within the expected time window
RADM_TIMEOUT_CPL_TC	O	3	The Traffic Class of the timed out completion
RADM_TIMEOUT_CPL_TAG	O	8	The Tag field of the timed out completion
RADM_TIMEOUT_CPL_ATTR	O	2	The Attributes field of the timed out completion.
RADM_TIMEOUT_CPL_LEN	O	11	Length (in bytes) of the timed out completion. For a split completion, it indicates the number of bytes remaining to be delivered when the completion timed out.
DYN_DEBUG_INFO_SEL	I	4	Dynamic debug_info_mux selection
APP_RAS DES SD_HOLDD_LTSSM	I	1	Hold and release LTSSM. For as long as this signal is "1", the controller stays in the current LTSSM.
APP_RAS DES TBA_CTRL	I	2	Controls the start/end of time based analysis. Users must only set the pins to the required value for the duration of one clock cycle. <ul style="list-style-type: none"> <li>■ 2'b00: No action</li> <li>■ 2'b01: Start</li> <li>■ 2'b10: End. This setting is only used when the TIME_BASED_DU-RATION_SELECT field of TIME_BASED_ANALYSIS_CONTROL_REG is set to "manual control".</li> <li>■ 2'b11: Reserved</li> </ul>
RADM_IDLE	O	1	RADM activity status signal

<b>Port</b>	<b>Direction</b>	<b>Width</b>	<b>Description</b>
RADM_Q_NOT_EMPTY	O	1	Level indicating that the receive queues contain TLP header/data.
RADM_QOVERFLOW	O	1	Pulse indicating that one or more of the P/NP/CPL receive queues have overflowed
DIAG_CTRL_BUS	I	2	Diagnostic Control Bus, 01: Insert LCRC error by inverting the LSB of LCRC 10: Insert ECRC error by inverting the LSB of ECRC
DEBUG_INFO_MUX	O	133	<p>When DYN_DEBUG_SEL_EN =0,use DEBUG_INFO_SEL</p> <p>When DYN_DEBUG_SEL_EN =1,use DYN_DEBUG_INFO_SEL</p> <p>*_DEBUG_INFO_SEL=0: MSI  [31:0] cfg_msi_mask[31:0]  [95:32] cfg_msi_addr[63:0]  [127:96] cfg_msi_data[31:0]  128 cfg_msi_64  [131:129] cfg_multi_msi_en[2:0]  132 cfg_msi_ext_data_en</p> <p>*_DEBUG_INFO_SEL=1: MSI-X  [10:0] cfg_msix_table_size[10:0]  [13:11] cfg_msix_table_bir[2:0]  [42:14] cfg_msix_table_offset[28:0]  [45:43] cfg_msix_pba_bir[2:0]  [74:46] cfg_msix_pba_offset[28:0]</p> <p>*_DEBUG_INFO_SEL=2: TX debug &amp;DL debug</p> <p>0 xadm_no_fc_credit[NVC-1:0]  1 xadm_notlp_pending  2 xadm_had_enough_credit[NVC-1:0]  3 xdlh_not_expecting_ack  4 xdlh_xmt_pme_ack  5 xdlh_nodllp_pending  [14:6] rtfngen_incr_amt[8:0]  15 rtfngen_incr_enable  [17:16] rtfngen_fctype[1:0]  18 xdlh_xth_halt  19 xtlh_xdlh_badeot  20 xtlh_xdlh_eot  21 xtlh_xdlh_sot  [26:22] active_grant[NCL+2-1:0]  [31:27] grant_ack[NCL+2-1:0]  [36:32] fc_cds_pass[(NCL+2)*NVC-1:0]  [41:37] arb_reqs[NCL+2-1:0]  42 xmlh_xdlh_halt  [44:43] xdlh_xmlh_sdp[1:0]  [46:45] xdlh_xmlh_stp[1:0]  [48:47] xdlh_xmlh_eot[1:0]  [60:49] rdlh_xdlh_req_acknack_seqnum[11:0]  61 rdlh_xdlh_req2send_nack  62 rdlh_xdlh_req2send_ack_due2dup  63 rdlh_xdlh_req2send_ack  [75:64] rdlh_xdlh_rcvd_acknack_seqnum[11:0]  76 rdlh_xdlh_rcvd_ack  77 rdlh_xdlh_rcvd_nack</p>

Port	Direction	Width	Description
			78 cfg_link_retrain 79 rtlh_req_link_retrain 80 xdlh_smlh_start_link_retrain 81 rdlh_rtlh_tlp_dv [83:82] rdlh_rtlh_tlp_eot[1:0] [85:84] rdlh_rtlh_tlp_sot[1:0] 86 ecrc_err_asserted 87 lcrc_err_asserted  *_DEBUG_INFO_SEL=3: 0 unexpected_cpl_err 1 cpl_ca_err 2 cpl_ur_err 3 flt_q_cpl_last 4 flt_q_cpl_abort 5 cpl_mlf_err [8:6]flt_q_header_cpl_status[2:0] [10:9] flt_q_header_destination[1:0] 11 form_filt_ecrc_err 12 form_filt_malform_tlp_err 13 form_filt_dllp_err 14 form_filt_eot [16:15] form_filt_dwen[NW-1:0] 17 form_filt_dv 18 form_filt_hv [20:19] rmlh_rdlh_pkt_err[NW-1:0] 21 rmlh_rdlh_pkt_dv [23:22] rmlh_rdlh_pkt_edb[NW-1:0] [25:24] rmlh_rdlh_pkt_end[NW-1:0] [27:26] rmlh_rdlh_tlp_start[NW-1:0] [29:28] rmlh_rdlh_dllp_start[NW-1:0] [31:30] rmlh_rdlh_nak[NW-1:0] [35:32] smlh_lanes_rcving[NL-1:0] 36 rmlh_rcvd_eidle_set 37 rmlh_rcvd_idle0 38 rmlh_rcvd_idle1 39 smlh_rcvd_lane_rev 40 smlh_ts_link_num_is_k237 41 rmlh_deskew_alignment_err 42 smlh_ts_lane_num_is_k237 43 smlh_ts2_rcvd 44 smlh_ts1_rcvd 45 smlh_ts_rcv_err 46 smlh_inskip_rcv  *_DEBUG_INFO_SEL=4: [48:0] cxpl_debug_info[63:0] [64:49] cxpl_debug_info_ei[15:0] [67:65] pm_curnt_state[2:0] 68 pm_sel_aux_clk 69 en_muxd_aux_clk_g 70 en_radm_clk_g 71 link_req_RST_not 72 pm_req_core_RST 73 pm_req_phy_RST 74 pm_req_sticky_RST 75 pm_req_non_sticky_RST

Port	Direction	Width	Description
			<pre> *_DEBUG_INFO_SEL=5: [7:0]cfg_int_pin[7:0] [43:8]  cfg_rbar_size[35:0] 44  cfg_br_ctrl_serren 45  xdlh_replay_timeout_err 46  xdlh_replay_num_rlover_err 47  rdlh_bad_dllp_err 48  rdlh_bad_tlp_err 49  rdlh_prot_err 50  rtlh_fc_prot_err 51  rmlh_rcvd_err 52  int_xadm_fc_prot_err 53  radm_unexp_cpl_err 54  radm_rcvd_cpl_ur 55  radm_rcvd_cpl_ca 56  radm_rcvd_req_ca 57  radm_rcvd_req_ur 58  radm_ecrc_err 59  radm_mlf_tlp_err 60  radm_rcvd_cpl_poisoned 61  radm_rcvd_wreq_poisoned 62  cfg_sys_err_rc_cor 63  cfg_sys_err_rc_nf 64  cfg_sys_err_rc_f  *_DEBUG_INFO_SEL=6: 0  cdm_lbc_ack [4:1]lbc_cdm_wr[3:0] 5  lbc_cdm_cs [37:6]  lbc_cdm_data[31:0] [69:38]  lbc_cdm_addr[31:0] *_debug_info_sel=7 0  smlh_eidle_inferred_in_l0: Level: Detect EI Infer 1  rmlh_rcvd_err: Pulse: Receiver Error 2  smlh_rx_rcvry_req: Level: Rx Recovery Request 3  smlh_timeout_nfts: Level: FTS Timeout 4  rmlh_framing_err: Pulse: Framing Error 5  rmlh_deskew_alignment_err: Level: Deskew Error 6  rdlh_bad_tlp_err_perltlp : Pulse: BAD TLP 7  rdlh_lcrc_tlp_err_perltlp : Pulse: LCRC Error 8  rdlh_bad_dllp_err_perdllp: Pulse: BAD DLLP 9  xdlh_replay_num_rlover_err: Pulse: Replay_Num Rollover 10 xdlh_replay_timeout_err: Pulse: Replay Timeout 11 rdlh_rcvd_nack_perdllp: Pulse: Rx Nak DLLP 12 xdlh_nak_sent: Pulse: Tx Nak DLLP 13 xdlh_retry_req: Pulse: Retry TLP 14 rtlh_req_link_retrain: Level: FC Timeout [16:15]  cfg_poisned_tlp: Pulse: Poisoned TLP [18:17]  cfg_ecrc_tlp_err: Pulse: ECRC Error [20:19]  cfg_ur_tlp: Pulse: Unsupported Request [22:21]  cfg_ca_tlp: Pulse: Completer Abort [24:23]  cfg_cpl_timeout[1:0]: Pulse: Completion Timeout 25  smlh_l0_to_recovery: Pulse: L0 to Recovery Entry 26  smlh_l1_to_recovery: Pulse: L1 to Recovery Entry 27  smlh_in_l0s: Level: Tx L0s Entry </pre>

Port	Direction	Width	Description
			28 smlh_in_rl0s: Level: Rx L0s Entry 29 pm_asnak: Level: ASPM L1 reject 30 smlh_in_l1: Level: L1 Entry 31 pm_in_l11: Level: L1.1 Entry 32 pm_in_l12: Level: L1.2 Entry 33 pm_in_l1_short: Level: L1 short duration 34 pm_in_l1_cpm: Level: L1 Clock PM (L1 with REFCLK removal/PLL Off) 35 pm_in_l1_abort: Level: L1.2 abort 36 smlh_in_l23: Level: L2 Entry 37 smlh_spd_change: Pulse: Speed Change 38 smlh_lwd_change: Pulse: Link width Change 39 xdlh_ack_sent: Pulse: Tx Ack DLLP 40 xdlh_update_fc_sent: Pulse: Tx Update FC DLLP 41 rdlh_rcvd_ack_perdlpp: Pulse: Rx Ack DLLP 42 rtlh_rcvd_ufc_perdlpp: Pulse: Rx Update FC DLLP 43 rdlh_nulified_tlp_err_perltlp : Pulse: Rx Nullified TLP 44 xtlh_xadm_restore_enable: Pulse: Tx Nullified TLP 45 rdlh_duplicate_tlp_err_perltlp: Pulse: Rx Duplicate TLP 46 xtlh_tx_memwr_evt: Pulse: Tx Memory Write 47 xtlh_tx_memrd_evt: Pulse: Tx Memory Read 48 xtlh_tx_cfgwr_evt: Pulse: Tx Config Write 49 xtlh_tx_cfgrd_evt: Pulse: Tx Config Read 50 xtlh_tx_iowr_evt: Pulse: Tx IO Write 51 xtlh_tx_iord_evt: Pulse: Tx IO Read 52 xtlh_tx_cplwod_evt: Pulse: Tx Completion wo data 53 xtlh_tx_cplwd_evt: Pulse: Tx Completion w data 54 xtlh_tx_msg_evt: Pulse: Tx Message 55 xtlh_tx_atmcop_evt: Pulse: Tx AtomicOp 56 xtlh_tx_tlpwprefix_evt: Pulse: Tx TLP with Prefix 57 rtlh_rx_memwr_evt: Pulse: Rx Memory Write 58 rtlh_rx_memrd_evt: Pulse: Rx Memory Read 59 rtlh_rx_cfgwr_evt: Pulse: Rx Config Write 60 rtlh_rx_cfgrd_evt: Pulse: Rx Config Read 61 rtlh_rx_iowr_evt: Pulse: Rx IO Write 62 rtlh_rx_iord_evt: Pulse: Rx IO Read 63 rtlh_rx_cplwod_evt: Pulse: Rx Completion wo data 64 rtlh_rx_cplwd_evt: Pulse: Rx Completion w data 65 rtlh_rx_msg_evt: Pulse: Rx Message TLP 66 rtlh_rx_atmcop_evt: Pulse: Rx Atomic 67 rtlh_rx_tlpwprefix_evt: Pulse: Rx TLP with Prefix 68 xtlh_tx_ccix_tlp_evt: Pulse: Tx CCIX TLP 69 rtlh_rx_ccix_tlp_evt: Pulse: Rx CCIX TLP 82:70 cdm_ras_des_ec_info_l0[12:0] 95:83 cdm_ras_des_ec_info_l1[12:0] 108:96 cdm_ras_des_ec_info_l2[12:0] 121:109 cdm_ras_des_ec_info_l3[12:0] 128:122 cdm_ras_des_tba_info_common[6:0]  *_debug_info_sel=8 [4:0]pm_master_state[4:0]: Level: PM Internal State (Master) [8:5]pm_slave_state[3:0]: Level: PM Internal State (Slave) [15:9] rmlh_framing_err_ptr[6:0]: Pulse: 1st Framing Error Pointer [16] smlh_lane_reversed: Level: Lane Reversal Operation [17] pm_pme_resend_flag: Pulse: PME Re-Send flag [33:18] smlh_ltssm_variable [15:0]: Level: LTSSM

<b>Port</b>	<b>Direction</b>	<b>Width</b>	<b>Description</b>
			Variable [36:34] ltssm_powerdown[1:0]: Level: PIPE: Power Down [44:37] latched_ts_nfts[7:0]: Level: Latched NFTS [46:45] rdlh_dlcntrl_state [1:0]: Level: DLCM [47] rdlh_vc0_initfc1_status: Level: Init-FC Flag1 VC0 [48] rdlh_vc0_initfc2_status: Level: Init-FC Flag2 VC0 [60:49] rdlh_curnt_rx_ack_seqnum[11:0]: Level: Rx ACK SEQ# [72:61] xdlh_curnt_seqnum [11:0]: Level: Tx TLP SEQ# 85:73 cdm_ras_des_sd_info_l0[12:0] 98:86 cdm_ras_des_sd_info_l1[12:0] 111:99 cdm_ras_des_sd_info_l2[12:0] 124:112 cdm_ras_des_sd_info_l3[12:0]  *_debug_info_sel=9 119:0 cdm_ras_des_sd_info_v0[239:120]  *_debug_info_sel=10 119:0 cdm_ras_des_sd_info_v0[119:0]
CFG_IDO_REQ_EN	O	1	ID-Based Ordering Requests Enabled
CFG_IDO_CPL_EN	O	1	ID-Based Ordering Completions Enabled
XADM_PH_CDTs[7:0]	O	8	The amount of posted header buffer space currently available at the receiver at the other end of the link
XADM_PD_CDTs[11:0]	O	12	The amount of posted data buffer space currently available at the receiver at the other end of the link
XADM_NPH_CDTs[7:0]	O	8	The amount of non-posted header buffer space currently available at the receiver at the other end of the link
XADM_NPD_CDTs[11:0]	O	12	The amount of non-posted data buffer space currently available at the receiver at the other end of the link
XADM_CPLH_CDTs[7:0]	O	8	The amount of completion header buffer space currently available at the receiver at the other end of the link
XADM_CPLD_CDTs[11:0]	O	12	The amount of completion data buffer space currently available at the receiver at the other end of the link
MAC_PHY_POWERDOWN	O	2	Power control bits to the PHY 00: P0 (L0): normal 01: P0s (L0s): low recovery time, power saving. 10: P1 (L1): longer recovery time, additional power saving. 11: P2 (L2): lowest power state.
PHY_MAC_RXELECIDLE	I	4	Indicates receiver detection of an Electrical Idle for each lane
PHY_MAC_PHYSTATUS	I	4	Communicates completion of PHY functions, including power management transitions, receiver detection, speed change.
PHY_MAC_RXDATA	I	128	Parallel received data, 32 bits per lane
PHY_MAC_RXDATAK	I	16	Control (K character) indicator bits for received data
PHY_MAC_RXVALID	I	4	Indicates symbol lock and valid data for each lane
PHY_MAC_RXSTATUS	I	12	Receive status and error codes for each lane(3bits per lane)
MAC_PHY_TXDATA	O	128	Parallel data for transmission
MAC_PHY_TXDATAK	O	16	Control (K character) indicator bits for transmitted data:
MAC_PHY_TXDETECT_RX_LOOPBACK	O	4	Combined loopback and transmit detect control, as per PIPE specification.
MAC_PHY_TXELECID	O	4	Forces transmit output to Electrical Idle for each lane which it

<b>Port</b>	<b>Direction</b>	<b>Width</b>	<b>Description</b>
LE_H			is asserted.
MAC_PHY_TXELECID LE_L	O	4	See the pcie2_hard_pipe_adapter section for details
MAC_PHY_TXCOMPLI ANCE	O	4	Sets the running disparity to negative
MAC_PHY_RXPOLARI TY	O	4	Directs the PHY to perform a polarity inversion the received data the specified lanes
MAC_PHY_RATE	O	1	Controls the link signaling rate
MAC_PHY_TXDEEMP H	O	2	For Gen2 configurations, this two bits wide output selects transmitter de-emphasis as follows
MAC_PHY_TXMARGIN	O	3	Selects transmitter voltage levels
MAC_PHY_TXSWING	O	1	Controls the PHY transmitter voltage swing level
CFG_HW_AUTO_SP_DI S	O	1	Autonomous speed disable.
RAM_TEST_EN	I	1	Active-high external_ram test enable signal
RAM_TEST_ADDRH	I	1	Test pattern: 1: test addr at the high data bit 0: test addr at the low data bit
RETRY_TEST_DATA_E N	I	1	Test mode (used only in single-port ram): 1: Test data line 0: Test control line
P_DATAQ_DATAOUT	I	66	Data buffer data output
P_DATAQ_ADDRA	O	10	Data buffer port A address
P_DATAQ_ADDRB	O	10	Data buffer port B address
P_DATAQ_DATAIN	O	66	Data buffer data input
P_DATAQ_ENA	O	1	Data buffer port A enable
P_DATAQ_ENB	O	1	Data buffer port B enable
P_DATAQ_WEA	O	1	Data buffer port A write enable
RETRYRAM_XDLH_DA TA	I	68	retry buffer data output
XDLH_RETRYRAM_A DDR	O	10	retry buffer address
XDLH_RETRYRAM_DA TA	O	68	retry buffer data input
XDLH_RETRYRAM_W E	O	1	retry buffer write enable
XDLH_RETRYRAM_EN	O	1	retry buffer enable
P_HDRQ_DATAOUT	I	138	Hdr buffer data output
P_HDRQ_ADDRA	O	9	Hdr buffer port A address
P_HDRQ_ADDRB	O	9	Hdr buffer port B address
P_HDRQ_DATAIN	O	138	Hdr buffer data input
P_HDRQ_ENA	O	1	Hdr buffer port A enable
P_HDRQ_ENB	O	1	Hdr buffer port B enable
P_HDRQ_WEA	O	1	Hdr buffer port A write enable
RAM_TEST_MODE_N	I	1	ram_test_mode_n for opening ram test mode Active-low

Note: The above are the names and functional descriptions of all ports under working mode. If in DFT test mode, the following three ports will be added

TEST_SE_N	I	1	Test shift enable, active-low
TEST_RST_N	I	1	Test mode reset, active-low
TEST_MODE_N	I	1	Active low to enable test mode

#### 10.7.4 Parameter Description

Table 10-21 GTP\_PCIEGEN2 Parameter Description

Parameter Name	Parameter Type	Valid Values	Defaults	Description
GRS_EN	<string>	"FALSE","TRUE"	TRUE	Bypass global rst
PIN_MUX_INT_FORCE_EN	<string>	"FALSE","TRUE"	FALSE	When setting "TRUE", it uses "PIN_MUX_INT_DISABLE" to judge whether "legacy int" should be disable .
PIN_MUX_INT_DISABLE	<string>	"FALSE","TRUE"	FALSE	Replace internal signal: "cfg_int_disable" when setting this signal and "PIN_MUX_INT_FORCE_EN" are "TRUE".
DIAG_CTRL_BUS_B2	<string>	"NORMAL", "FAST_LINK_MODE"	NORMAL	Select fast Link Mode for simulation when it is "FAST_LINK_MODE "
DYN_DEBUG_SEL_EN	<string>	"FALSE","TRUE"	FALSE	FALSE: "DEBUG_INFO_SEL" controls "debug_info_mux" TRUE: "DYN_DEBUG_INFO_SEL" controls "debug_info_mux"
DEBUG_INFO_SEL	<integer>	0 to 15	0	Control "DEBUG_DEBUG_INFO_MUX"
BAR_RESIZABLE	<integer>	"0 to 56" Note: Please do not set more than 3 resizable bars at the same time	21 Note: 6'b010101 in binary	High active indicates that the corresponding bar is set to resizable. Note: the bar must be enabled; with a corresponding BAR_INDEX_0/1/2 indication and the maximum number of simultaneously valid bits is three.
NUM_OF_RBARS	<integer>	"0", "1", "2", "3"	3	Number of Resizable Bars in Resizable Bar Control Register
BAR_INDEX_0	<integer>	"0 to 5"	0	Correspond to the bar index in the Resizable BAR Control Register (at offset 008h)
BAR_INDEX_1	<integer>	"0 to 5"	2	Correspond to the bar index in the Resizable BAR Control Register (at offset 010h)
BAR_INDEX_2	<integer>	"0 to 5"	4	Correspond to the bar index in the Resizable BAR Control Register (at offset 018h)
TPH_DISABLE	<string>	"FALSE","TRUE"	FALSE	Bypass "TLP Processing Hints Supported"
MSIX_CAP_DISABLE	<string>	"FALSE","TRUE"	FALSE	Bypass "MSIX Capability"
MSI_CAP_DISABLE	<string>	"FALSE","TRUE"	FALSE	Bypass "MSI Capability"

Parameter Name	Parameter Type	Valid Values	Defaults	Description
MSI_PVM_DISABLE	<string>	"FALSE","TRUE"	FALSE	Bypass "Per vector Masking Capable"
BAR_MASK_WRITABLE	<integer>	"0 to 63"	32	Active-high indicates that the corresponding bar_mask value is writable
APP_DEV_NUM	<integer>	"0 to 31"	0	Device number. Your application must drive this signal to set the device number in the Requester ID for RC mode.
APP_BUS_NUM	<integer>	"0 to 255"	0	Bus number. Your application must drive this signal to set the bus number in the Requester ID for RC mode.
RAM_MUX_EN	<string>	"FALSE","TRUE"	FALSE	Enable merging of rcv hdr ram and rcv data ram outputs
ATOMIC_DISABLE	<string>	"FALSE","TRUE"	FALSE	Used to configure whether to disable the ATOMIC (active-high)

### 10.7.5 Instantiation template

```
GTP_PCIEGEN2 #(
    .GRS_EN("TRUE"),
    .PIN_MUX_INT_FORCE_EN("FALSE"),
    .PIN_MUX_INT_DISABLE("FALSE"),
    .DIAG_CTRL_BUS_B2("NORMAL"),
    .DYN_DEBUG_SEL_EN("FALSE"),
    .DEBUG_INFO_SEL(0),
    .BAR_RESIZABLE(21),
    .NUM_OF_RBARS(3),
    .BAR_INDEX_0(0),
    .BAR_INDEX_1(2),
    .BAR_INDEX_2(4),
    .TPH_DISABLE("FALSE"),
    .MSIX_CAP_DISABLE("FALSE"),
    .MSI_CAP_DISABLE("FALSE"),
    .MSI_PVM_DISABLE("FALSE"),
    .BAR_MASK_WRITABLE(32),
    .APP_DEV_NUM(0),
    .APP_BUS_NUM(0),
    .RAM_MUX_EN("FALSE"),
```

```

.ATOMIC_DISABLE("FALSE")

) GTP_PCIEGEN2_inst (
    .AXIS_MASTER_TDATA(),           // OUTPUT[127:0]
    .AXIS_MASTER_TKEEP(),           // OUTPUT[3:0]
    .AXIS_MASTER_TUSER(),           // OUTPUT[7:0]
    .CFG_MAX_PAYLOAD_SIZE(),        // OUTPUT[2:0]
    .CFG_MAX_RD_REQ_SIZE(),         // OUTPUT[2:0]
    .CFG_PBUS_DEV_NUM(),           // OUTPUT[4:0]
    .CFG_PBUS_NUM(),               // OUTPUT[7:0]
    .CFG_PF_TPH_ST_MODE(),         // OUTPUT[2:0]
    .CFG_TPH_REQ_EN(),             // OUTPUT[1:0]
    .DEBUG_INFO_MUX(),             // OUTPUT[132:0]
    .LBC_DBI_DOUT(),               // OUTPUT[31:0]
    .MAC_PHY_POWERDOWN(),          // OUTPUT[1:0]
    .MAC_PHY_RXPOLARITY(),          // OUTPUT[3:0]
    .MAC_PHY_TXCOMPLIANCE(),        // OUTPUT[3:0]
    .MAC_PHY_TXDATA(),              // OUTPUT[127:0]
    .MAC_PHY_TXDATAK(),             // OUTPUT[15:0]
    .MAC_PHY_TXDEEMPH(),            // OUTPUT[1:0]
    .MAC_PHY_TXDETECTRX_LOOPBACK(), // OUTPUT[3:0]
    .MAC_PHY_TXELECIDLE_H(),        // OUTPUT[3:0]
    .MAC_PHY_TXELECIDLE_L(),        // OUTPUT[3:0]
    .MAC_PHY_TXMARGIN(),            // OUTPUT[2:0]
    .PM_DSTATE(),                  // OUTPUT[2:0]
    .PM_MASTER_STATE(),             // OUTPUT[4:0]
    .PM_SLAVE_STATE(),              // OUTPUT[4:0]
    .P_DATAQ_ADDRA(),              // OUTPUT[9:0]
    .P_DATAQ_ADDRB(),              // OUTPUT[9:0]
    .P_DATAQ_DATAIN(),              // OUTPUT[65:0]
    .P_HDRQ_ADDRA(),                // OUTPUT[8:0]
    .P_HDRQ_ADDRB(),                // OUTPUT[8:0]
    .P_HDRQ_DATAIN(),               // OUTPUT[137:0]
    .RADM_GRANT_TLP_TYPE(),          // OUTPUT[5:0]
    .RADM_TIMEOUT_CPL_ATTR(),        // OUTPUT[1:0]

```

```

.RADM_TIMEOUT_CPL_LEN(),           // OUTPUT[10:0]
.RADM_TIMEOUT_CPL_TAG(),          // OUTPUT[7:0]
.RADM_TIMEOUT_CPL_TC(),           // OUTPUT[2:0]
.SMLH_LTSSM_STATE(),             // OUTPUT[4:0]
.XADM_CPLD_CDTs(),               // OUTPUT[11:0]
.XADM_CPLH_CDTs(),               // OUTPUT[7:0]
.XADM_NPD_CDTs(),                // OUTPUT[11:0]
.XADM_NPH_CDTs(),                // OUTPUT[7:0]
.XADM_PD_CDTs(),                 // OUTPUT[11:0]
.XADM_PH_CDTs(),                 // OUTPUT[7:0]
.XDLH_RETRYRAM_ADDR(),           // OUTPUT[10:0]
.XDLH_RETRYRAM_DATA(),            // OUTPUT[67:0]
.APP_ERR_BUS(),                  // INPUT[12:0]
.APP_HDR_LOG(),                  // INPUT[127:0]
.APP_RAS DES_TBA_CTRL(),          // INPUT[1:0]
.AXIS_SLAVE0_TDATA(),             // INPUT[127:0]
.AXIS_SLAVE1_TDATA(),             // INPUT[127:0]
.AXIS_SLAVE2_TDATA(),             // INPUT[127:0]
.CFG_MSI_PENDING(),              // INPUT[31:0]
.DBI_ADDR(),                     // INPUT[31:0]
.DBI_DIN(),                      // INPUT[31:0]
.DBI_WR(),                        // INPUT[3:0]
.DEVICE_TYPE(),                  // INPUT[2:0]
.DIAG_CTRL_BUS(),                // INPUT[1:0]
.DYN_DEBUG_INFO_SEL(),            // INPUT[3:0]
.MSIX_ADDR(),                     // INPUT[63:0]
.MSIX_DATA(),                     // INPUT[31:0]
.PHY_MAC_PHYSTATUS(),             // INPUT[3:0]
.PHY_MAC_RXDATA(),                // INPUT[127:0]
.PHY_MAC_RXDATAK(),               // INPUT[15:0]
.PHY_MAC_RXELECIDLE(),            // INPUT[3:0]
.PHY_MAC_RXSTATUS(),              // INPUT[11:0]
.PHY_MAC_RXVALID(),               // INPUT[3:0]
.P_DATAQ_DATAOUT(),               // INPUT[65:0]

```

```

.P_HDRQ_DATAOUT(),           // INPUT[137:0]
.RETRYRAM_XDLH_DATA(),      // INPUT[67:0]
.TRGT1_RADM_PKT_HALT(),     // INPUT[2:0]
.VEN_MSI_TC(),              // INPUT[2:0]
.VEN_MSI_VECTOR(),          // INPUT[4:0]
.AUX_PM_EN(),                // OUTPUT
.AXIS_MASTER_TLAST(),        // OUTPUT
.AXIS_MASTER_TVALID(),       // OUTPUT
.AXIS_SLAVE0_TREADY(),       // OUTPUT
.AXIS_SLAVE1_TREADY(),       // OUTPUT
.AXIS_SLAVE2_TREADY(),       // OUTPUT
.CFG_AER_RC_ERR_MUX(),      // OUTPUT
.CFG_ATOMIC_EGRESS_BLOCK(),  // OUTPUT
.CFG_ATOMIC_REQ_EN(),        // OUTPUT
.CFG_BUS_MASTER_EN(),        // OUTPUT
.CFG_BW_MGT_MUX(),          // OUTPUT
.CFG_CRS_SW_VIS_EN(),        // OUTPUT
.CFG_EXT_TAG_EN(),           // OUTPUT
.CFG_HW_AUTO_SP_DIS(),       // OUTPUT
.CFG_IDO_CPL_EN(),           // OUTPUT
.CFG_IDO_REQ_EN(),           // OUTPUT
.CFG_INT_DISABLE(),          // OUTPUT
.CFG_LINK_AUTO_BW_MUX(),     // OUTPUT
.CFG_MEM_SPACE_EN(),          // OUTPUT
.CFG_MSIX_EN(),              // OUTPUT
.CFG_MSIX_FUNC_MASK(),        // OUTPUT
.CFG_MSI_EN(),                // OUTPUT
.CFG_NO_SNOOP_EN(),           // OUTPUT
.CFG_PME_MUX(),               // OUTPUT
.CFG_PM_NO_SOFT_RST(),        // OUTPUT
.CFG_RCB(),                  // OUTPUT
.CFG_RELAX_ORDER_EN(),        // OUTPUT
.CFG_SEND_COR_ERR_MUX(),      // OUTPUT
.CFG_SEND_F_ERR_MUX(),        // OUTPUT

```

```

.CFG_SEND_NF_ERR_MUX(),           // OUTPUT
.CFG_SYS_ERR_RC(),               // OUTPUT
.CORE_RST_N(),                   // OUTPUT
.INTA_GRT_MUX(),                // OUTPUT
.INTB_GRT_MUX(),                // OUTPUT
.INTC_GRT_MUX(),                // OUTPUT
.INTD_GRT_MUX(),                // OUTPUT
.LBC_DBI_ACK(),                 // OUTPUT
.MAC_PHY_RATE(),                // OUTPUT
.MAC_PHY_TXSWING(),             // OUTPUT
.PHY_RST_N(),                   // OUTPUT
.PM_LINKST_IN_L0S(),             // OUTPUT
.PM_LINKST_IN_L1(),              // OUTPUT
.PM_LINKST_IN_L2(),              // OUTPUT
.PM_LINKST_L2_EXIT(),            // OUTPUT
.PM_PME_EN(),                   // OUTPUT
.PM_STATUS(),                    // OUTPUT
.PM_XTLH_BLOCK_TLP(),            // OUTPUT
.P_DATAQ_ENA(),                  // OUTPUT
.P_DATAQ_ENB(),                  // OUTPUT
.P_DATAQ_WEA(),                  // OUTPUT
.P_HDRQ_ENA(),                   // OUTPUT
.P_HDRQ_ENB(),                   // OUTPUT
.P_HDRQ_WEA(),                   // OUTPUT
.RADM_CPL_TIMEOUT(),             // OUTPUT
.RADM_IDLE(),                    // OUTPUT
.RADM_MSG_UNLOCK(),              // OUTPUT
.RADM_PM_PME(),                  // OUTPUT
.RADM_PM_TO_ACK(),                // OUTPUT
.RADM_PM_TURNOFF(),              // OUTPUT
.RADM_QOVERFLOW(),                // OUTPUT
.RADM_Q_NOT_EMPTY(),              // OUTPUT
.RBAR_CTRL_UPDATE(),              // OUTPUT
.RDLH_LINK_UP(),                 // OUTPUT

```

```

.SEDO(),                                // OUTPUT
.SEDO_EN(),                               // OUTPUT
.SMLH_LINK_UP(),                          // OUTPUT
.TRAINING_RST_N(),                        // OUTPUT
.VEN_MSI_GRANT(),                         // OUTPUT
.WAKE(),                                  // OUTPUT
.XDLH_RETRYRAM_EN(),                      // OUTPUT
.XDLH_RETRYRAM_WE(),                      // OUTPUT
.APPS_PM_XMT_PME(),                       // INPUT
.APPS_PM_XMT_TURNOFF(),                   // INPUT
.APP_CLK_PM_EN(),                          // INPUT
.APP_DBI_RO_WR_DISABLE(),                  // INPUT
.APP_ERR_ADVISORY(),                       // INPUT
.APP_HDR_VALID(),                          // INPUT
.APP_INIT_RST(),                           // INPUT
.APP_LTSSM_EN(),                           // INPUT
.APP_RAS DES_SD_HOLD_LTSSM(), // INPUT
.APP_READY_ENTR_L23(),                     // INPUT
.APP_REQ_ENTR_L1(),                        // INPUT
.APP_REQ_EXIT_L1(),                        // INPUT
.APP_REQ_RETRY_EN(),                       // INPUT
.APP_UNLOCK_MSG(),                         // INPUT
.APP_XFER_PENDING(),                       // INPUT
.AXIS_MASTER_TREADY(),                     // INPUT
.AXIS_SLAVE0_TLAST(),                      // INPUT
.AXIS_SLAVE0_TUSER(),                      // INPUT
.AXIS_SLAVE0_TVALID(),                     // INPUT
.AXIS_SLAVE1_TLAST(),                      // INPUT
.AXIS_SLAVE1_TUSER(),                      // INPUT
.AXIS_SLAVE1_TVALID(),                     // INPUT
.AXIS_SLAVE2_TLAST(),                      // INPUT
.AXIS_SLAVE2_TUSER(),                      // INPUT
.AXIS_SLAVE2_TVALID(),                     // INPUT
.BUTTON_RST(),                            // INPUT

```

```
.DBI_CS(),           // INPUT
.DBI_CS2(),          // INPUT
.MEM_CLK(),          // INPUT
.OUTBAND_PWRUP_CMD(), // INPUT
.PCLK(),             // INPUT
.PCLK_DIV2(),        // INPUT
.PERST(),            // INPUT
.POWER_UP_RST(),     // INPUT
.RAM_TEST_ADDRH(),   // INPUT
.RAM_TEST_EN(),      // INPUT
.RAM_TEST_MODE_N(),  // INPUT
.RETRY_TEST_DATA_EN(), // INPUT
.RX_LANE_FLIP_EN(),  // INPUT
.SEDI(),              // INPUT
.SEDI_ACK(),          // INPUT
.SYS_AUX_PWR_DET(),   // INPUT
.SYS_INT(),            // INPUT
.TX_LANE_FLIP_EN(),  // INPUT
.VEN_MSI_REQ()        // INPUT
);
```

## Disclaimer

---

### Copyright Notice

This document is copyrighted by Shenzhen Pango Microsystems Co., Ltd., and all rights are reserved. Without prior written approval, no company or individual may disclose, reproduce, or otherwise make available any part of this document to any third party. Non-compliance will result in the Company initiating legal proceedings.

### Disclaimer

1. This document only provides information in stages and may be updated at any time based on the actual situation of the products without further notice. The Company assumes no legal responsibility for any direct or indirect losses caused by improper use of this document.
2. This document is provided "as is" without any warranties, including but not limited to warranties of merchantability, fitness for a particular purpose, non-infringement, or any other warranties mentioned in proposals, specifications, or samples. This document does not grant any explicit or implied intellectual property usage license, whether by estoppel or otherwise.
3. The Company reserves the right to modify any documents related to its family products at any time without prior notice.