Mistral documentation

Release 1.0

Oct 20, 2021

CONTENTS

1 The Cyclone V FPGA **1**
- 1.1 The FPGAs . . . 1
- 1.2 Bitstream stucture . . . 2
- 1.3 Logic blocks . . . 2
- 1.4 Routing network . . . 3
- 1.5 Programmable inverters . . . 3

2 CycloneV internals description **5**
- 2.1 Routing network . . . 5
- 2.2 Inner logic blocks . . . 6
- 2.3 Peripheral logic blocks . . . 22
- 2.4 Options . . . 121

3 CycloneV library usage **123**
- 3.1 Library structure . . . 123
- 3.2 Packages . . . 123
- 3.3 Model information . . . 124
- 3.4 pos, rnode and pnode . . . 125
- 3.5 Routing network management . . . 127
- 3.6 Logic block management . . . 127
- 3.7 Inverters management . . . 128
- 3.8 Pin/package management . . . 128
- 3.9 Options . . . 130
- 3.10 Bitstream management . . . 130
- 3.11 HMC bypass . . . 130

4 The mistral-cv command-line program **133**
- 4.1 models . . . 133
- 4.2 routes . . . 133
- 4.3 routes2 . . . 133
- 4.4 cycle . . . 133
- 4.5 bels . . . 134
- 4.6 decomp . . . 134
- 4.7 comp . . . 134
- 4.8 diff . . . 134

5 Mistral CycloneV library internals **135**
- 5.1 Structure . . . 135
- 5.2 Routing data . . . 135
- 5.3 Block muxes . . . 135

5.4 Logic blocks . . . 137
5.5 Inverters . . . 137
5.6 Forced-1 bits . . . 137
5.7 Packages . . . 137
5.8 Models . . . 137
5.9 Binary data . . . 137

CHAPTER

ONE

THE CYCLONE V FPGA

1.1 The FPGAs

The Cyclone V is a series of FPGAs produced initially by Altera, now Intel. It is based on a series of seven dies with varying levels of capability, which is then derived into more than 400 SKUs with variations in speed, temperature range, and enabled internal hardware.

As pretty much every FPGA out there, the dies are organized in grids.

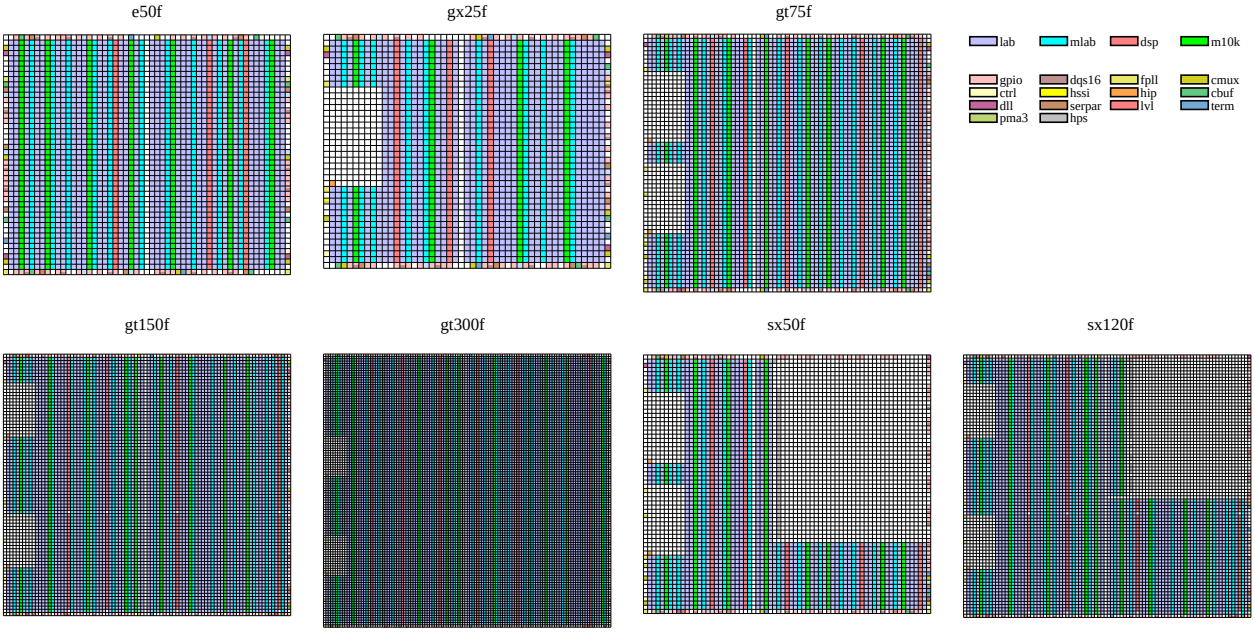

Fig. 1: Floor plan of the seven die types

The FPGA, structurally, is a set of logic blocks of different types communicating with each other either through direct links or through a large routing network that spans the whole grid.

Some of the logic blocks take visible floor space. Specifically, the notches on the left are the space taken by the high speed serial interfaces (hssi and pma3). Also, the top-right corner in the sx50f and sx120f variants is used to fit the hps, a dual-core arm.

1.2 Bitstream stucture

The bitstream is built from three rams:

- Option ram
- Peripheral ram
- Configuration ram

The option ram is composed of 32 blocks of 40 bits, of which only 12 are actually used. It includes the global configurations for the chip, such as the jtag user id, the programming voltage, the internal oscillator configuration, etc.

The peripheral ram stores the configuration of all the blocks situated on the borders of the chip, e.g. everything outside of labs, mlabs, dsps and m10ks. It is built of 13 to 16 blocks of bits that are sent through shift registers to the tiles.

The configuration ram stores the configuration of the labs, mlabs, dsps and m10ks, plus all the routing configuration. It also includes the programmable inverters which allows inverting essentially all the inputs to the peripheral blocks. It is organised as a rectangle of bits.

Die	Tiles	Pram	Cram
e50f	55x46	51101	4958x3928
gx25f	49x40	54083	3856x3412
gt75f	69x62	90162	6006x5304
gt150f	90x82	113922	7605x7024
gt300f	122x116	130828	10038x9948
sx50f	69x62	80505	6006x5304
sx120f	90x82	99574	7605x7024

1.3 Logic blocks

The logic blocks are of two categories, the inner blocks and the peripheral blocks. To a first approximation all the inner blocks are configured through configuration ram, and the peripheral blocks through the peripheral ram. It only matters where it comes to partial reconfiguration, because only the configuration ram can be dynamically modified. We do not yet support it though.

The inner blocks are:

- lab: a logic blocks group with 20 LUTs with 5 inputs and 40 Flip-Flops.
- mlab: a lab that can be reconfigured as 64*20 bits of ram
- dsp: a flexible multiply-add block
- m10k: a block of 10240 bits of dual-ported memory

The peripheral blocks are:

- gpio: general-purpose i/o, a block that controls up to 4 package pins
- dqs16: a block that manage differential input/output for 4 gpio blocks, e.g. up to 16 pins
- fpll: a fractional PLL
- cmux: the clock muxes that drive the clock part of the routing network
- ctrl: the control block with things like jtag
- hssi: the high speed serial interfaces

- hip: the pcie interfaces
- cbuf: a clock buffer for the dqs16
- dll: a delay-locked loop for the dqs16
- serpar: TODO
- lvl: TODO
- term: termination control blocks
- pma3: manages the channels of the hssi
- hmc: hardware memory controller, a block managing sdr/ddr ram interfaces
- hps: a series of 37 blocks managing the interface with the integrated dual-core arm

All of these blocks are configured similarly, through the setup of block muxes. They can be of 4 types: * Boolean * Symbolic, where the choice is between alphanumeric states * Numeric, where the choice is between a fixed set of numeric value * Ram, where a series of bits can be set to any value

Configuring that part of the FPGA consists of configuring the muxes associated to each block.

1.4 Routing network

A massive routing network is present all over the FPGA. It has two almost-disjoint parts. The data network has a series of inputs, connected to the outputs of all the blocks, and a series of outputs that go to data inputs of the blocks. The clock network consists of 16 global clocks signals that cover the whole FPGA, up to 88 regional clocks that cover an half of the FPGA, and when an hssi is present a series of horizontal peripheral clocks that are driven by the serial communications. Global and regional clock signals are driven by dedicated cmux blocks (not the fpll in particular, but they do have dedicated connections to the cmuxes).

These two networks join on data/clock muxes, which allow peripheral blocks to select for their clock-like inputs which network the signal should come from.

1.5 Programmable inverters

Essentially every output of the routing network that enters a peripheral block can optionally be inverted by activating the associated configuration bit.

CHAPTER

TWO

CYCLONEV INTERNALS DESCRIPTION

2.1 Routing network

The routing network follows a single-driver structure: a number of inputs are grouped together in one place, one is selected through the configuration, then it is amplified and used to drive a metal line. There is also usually one bit configuration to disable the driver, which can be all-off (probably leaving the line floating) or a specific combination to select vcc. The drivers correspond to a 2d pattern in the configuration ram. There are 70 different patterns, configured by 1 to 18 bits and mixing 1 to 44 inputs.

The network itself can be split in two parts: the data network and the clock network.

The data network is a grid of connections. Horizontal lines (H14, H6 and H3, numbered by the number of tiles they span) and vertical lines (V12, V4 and V2) helped by wire muxes (WM) connect to each over to ensure routing over the whole surface. Then at the tile level tile-data dispatch (TD) nodes allow to select between the available signals.

Generic output (GOUT) nodes then select between TD nodes to connect to logic blocks inputs. Logic block outputs go to Generic Input (GIN) nodes which feed in the connections. In addition a dedicated network, the Loopback dispatch (LD) connects some of the outputs from the labs/mlabs to their inputs for fast local data routing.

The clock network is more of a top-down structure. The top structures are Global clocks (GCLK), Regional clocks (RCLK) and Peripheral clocks (PCLK). They're all driven by specialized logic blocks we call Clock Muxes (cmux). There are two horizontal cmux in the middle of the top and bottom borders, each driving 4 GCLK and 20 RCLK, two vertical in the middle of the left and right borders each driving 4 GCLK and 12 RCLK, and 3 to 4 in the corners driving 6 RCLK each. The dies including an HPS (sx50f and sx120f) are missing the top-right cmux plus some of the middle-of-border-driven RCLK. That gives a total of 16 GCLK and 66 to 88 RCLK. In addition PCLK start from HSSI blocks to distribute serial clocks to the network.

The GCLK span the whole grid. A RCLK spans half the grid. A PCLK spans a number of tiles horizontally to its right.

The second level is Sector clocks, SCLK, which spans small rectangular zones of tiles and connect from GCLK, RCLK and PCLK. The on the third level, connecting from SCLK, is Horizontal clocks (HCLK) spanning 10-15 horizontal tiles and Border clocks (BCLK) rooted regularly on the top and bottom borders. Finally Tile clocks (TCLK) connect from HCLK and BCLK and distribute the clocks within a tile.

In addition the PMUX nodes at the entrance of plls select between SCLKs, and the GCLKFB and RCLKFB bring back feedback signals from the cmux to the pll.

Inner blocks directly connect to TCLK and have internal muxes to select between clock and data inputs for their control. Peripheral blocks tend to use a secondary structure composed from a TDMUX that selects one TD between multiple ones followed by a DCMUX that selects between the TDMUX and a TCLK so that their clock-like inputs can be driven from either a clock or a data signal.

Most GOUT and DCMUX connected to inputs to peripheral blocks are also provided with an optional inverter.

2.2 Inner logic blocks

2.2.1 LAB

The LABs are the main combinatorial and register blocks of the FPGA. A LAB tile includes 10 sub-blocks called cells with 64 bits of LUT splitted in 6 parts, four Flip-Flops, two 1-bit adders and a lot of routing logic. In addition a common control subblock selects and dispatches clock, enable, clear, etc signals.

Carry and share chain in the order lab (x, y+1) cell 9 -> cells 0-9 -> lab (x, u-1) cell 0. The BTO, TTO and BYPASS muxes control the connections in between 5-cell blocks.

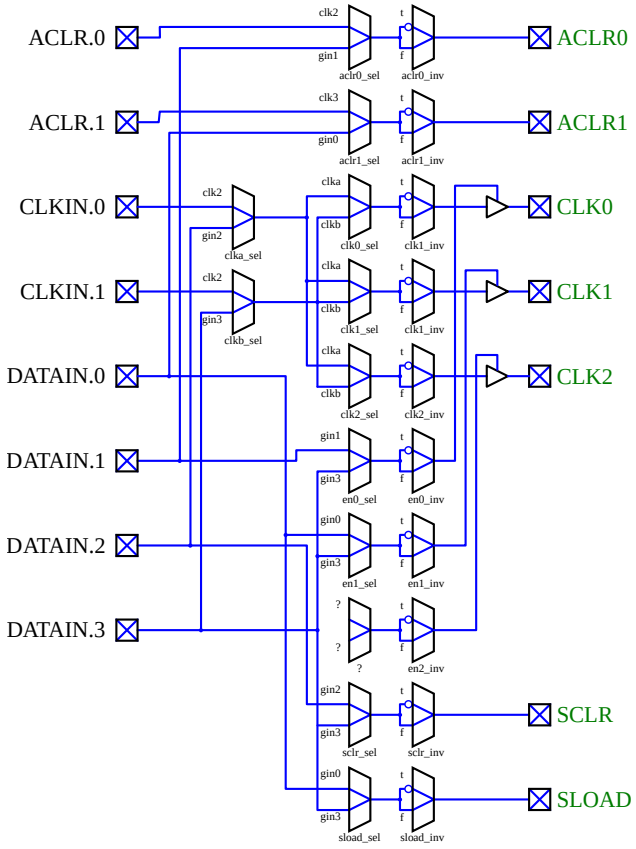

Fig. 1: The part of the LAB shared by all ten cells that generates the common signals.

Name	Instance	Type	Values	Default	Documentation
ARITH_SEL	0-9	Mux	• adder • lut	lut	Select whether the data input of the FF is the LUTs or the adder

continues on next page

Table 1 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
BCLK_SEL	0-9	Mux	• off • clk0 • clk1 • clk2	off	Select the clock input to the two bottom FFs
BCLR_SEL	0-9	Num	• 0-1	0	Select the aclr input to the two bottom FFs
BDFF0	0-9	Mux	• reg • nlut	reg	Select between LUT and FF for that output
BDFF1	0-9	Mux	• reg • nlut	reg	Select between LUT and FF for that output
BDFF1L	0-9	Mux	• reg • nlut	reg	Select between LUT and FF for that output
BEF_SEL	0-9	Mux	• e • f	e	Select which input goes to the sdata input of the two bottom FFs
BPKREG0	0-9	Bool	t/f	f	Force the top FF of the bottom half to get its input from tef_sel
BPKREG1	0-9	Bool	t/f	f	Force the bottom FF of the bottom half to get its input from tef_sel
BSCLR_DIS	0-9	Bool	t/f	f	Disable sync clear for the bottom half
BSLOAD_EN	0-9	Bool	t/f	f	Select whether to enable the sync load line of the two bottom FFs
B_FEEDBACK_SEL	0-9	Num	• 0-1	0	Select which of the FFs goes to the bottom feedback line

continues on next page

Table 1 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
LUT_MASK	0-9	Ram	64 bits	0	LUT values, A has bits 0-15, B 16-23, C 24-31, D 32-47, E 48-55. F 56-63
MODE	0-9	Mux	• l5 • l5_ft • l5_fb • l5_ftb • l6 • l6_ft • l6_fb • l6_ftb • l7_e0 • l7_e0_ft • l7_e0_fb • l7_e0_ftb • l7_e1 • l7_e1_ft • l7_e1_fb • l7_e1_ftb	l6	Connectivity mode of the cell
SHARE	0-9	Bool	t/f	f	Route the share line to the addition
TCLK_SEL	0-9	Mux	• off • clk0 • clk1 • clk2	off	Select the clock input to the two top FFs
TCLR_SEL	0-9	Num	• 0-1	0	Select the aclr input to the two top FFs
TDFF0	0-9	Mux	• reg • nlut	reg	Select between LUT and FF for that output
TDFF1	0-9	Mux	• reg • nlut	reg	Select between LUT and FF for that output
TDFF1L	0-9	Mux	• reg • nlut	reg	Select between LUT and FF for that output

continues on next page

Table 1 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
TEF_SEL	0-9	Mux	• e • f	e	Select which input goes to the sdata input of the two top FFs
TPKREG0	0-9	Bool	t/f	f	Force the top FF of the top half to get its input from tef_sel
TPKREG1	0-9	Bool	t/f	f	Force the bottom FF of the top half to get its input from tef_sel
TSCLR_DIS	0-9	Bool	t/f	f	Disable sync clear for the top half
TSLOAD_EN	0-9	Bool	t/f	f	Select whether to enable the sync load line of the two top FFs
T_FEEDBACK_SEL	0-9	Num	• 0-1	0	Select which of the FFs goes to the top feedback line
ACLR0_INV		Bool	t/f	f	Optional inverter for asynchronous clear 0
ACLR0_SEL		Mux	• gin1 • clki2	gin1	Selects between clock and data for async clear 0
ACLR1_INV		Bool	t/f	f	Optional inverter for asynchronous clear 1
ACLR1_SEL		Mux	• gin0 • clki3	gin0	Selects between clock and data for async clear 1
BTO_DIS		Bool	t/f	f	When disabled, allows carry in/share in from local cell 4 into local cell 5

continues on next page

Table 1 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
BYPASS_DIS		Bool	t/f	t	Bypass skips the top half (lab) or bottom half (mlab) of the cells for the carry and share chains (needs BTO, resp. TTO disabled too)
CLK0_INV		Bool	t/f	f	Optional inverter for clock 0
CLK0_SEL		Mux	• clka • clkb	clka	Selects between the two intermedaite clock lines for clock 0
CLK1_INV		Bool	t/f	f	Optional inverter for clock 1
CLK1_SEL		Mux	• clka • clkb	clka	Selects between the two intermedaite clock lines for clock 1
CLK2_INV		Bool	t/f	f	Optional inverter for clock 2
CLK2_SEL		Mux	• clka • clkb	clka	Selects between the two intermedaite clock lines for clock 2
CLKA_SEL		Mux	• clki0 • gin2	clki0	Selects between clock and data for the clka intermediate line
CLKB_SEL		Mux	• clki1 • gin3	clki1	Selects between clock and data for the clkb intermediate line
DFT_MODE		Mux	• off • on • dft_pprog	on	TODO
EN0_EN		Bool	t/f	t	Enables the enable 0 line (else always on)
EN0_NINV		Bool	t/f	t	Optional inverter for enable 0

continues on next page

Table 1 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
EN0_SEL		Mux	• gin1 • gin3	gin1	Source selection for enable 0
EN1_EN		Bool	t/f	t	Enables the enable 1 line (else always on)
EN1_NINV		Bool	t/f	t	Optional inverter for enable 1
EN1_SEL		Mux	• gin0 • gin3	gin3	Source selection for enable 1
EN2_EN		Bool	t/f	t	Enables the enable 2 line (else always on)
EN2_NINV		Bool	t/f	t	Optional inverter for enable 2
EN_SCLK_LOAD_WHAT		Bool	t/f	f	Unclear, possibly source selection for enable 2
REGSCAN_LATCH_EN		Bool	t/f	f	TODO
SCLR_INV		Bool	t/f	f	Optional inverter for synchronous clear
SCLR_MUX		Mux	• gin3 • gin2	gin3	Source selection for sync clear, possibly more subtle (interaction with en2 and sload)
SLOAD_INV		Bool	t/f	t	Optional inverter for synchronous load
SLOAD_SEL		Mux	• gin0 • gin3	gin0	Source selection for sync load, possibly more subtle (interaction with en2 and sclr)

continues on next page

Table 1 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
TTO_DIS		Bool	t/f	f	When disabled, allows carry in/share in from the lab at (x, y+1) cell 9 into local cell 0

Port Name	Instance	Port bits	Route node type	Documentation
A	0-9		GOUT	Data input to the lab cell
ACLR		0-1	TCLK	Common clock inputs for asynchronous clear of the FFs
B	0-9		GOUT	Data input to the lab cell
C	0-9		GOUT	Data input to the lab cell
CLKIN		0-1	TCLK	Common clock inputs for clocking of the FFs
D	0-9		GOUT	Data input to the lab cell
DATAIN		0-3	GOUT	Common data inputs for enables, sync clear and load
E0	0-9		GOUT	Data input to the lab cell
E1	0-9		GOUT	Data input to the lab cell
F0	0-9		GOUT	Data input to the lab cell
F1	0-9		GOUT	Data input to the lab cell
FFB0	0-9		GIN	Output from either the top FF of the bottom hslf of the lab cell or the bottomlut to data routing
FFB1	0-9		GIN	Output from either the bottom FF of the bottom hslf of the lab cell or the bottom lut to data routing
FFB1L	0-9		LD	Output from either the bottom FF of the bottom hslf of the lab cell or the bottom lut to local dispatch
FFT0	0-9		GIN	Output from either the top FF of the top hslf of the lab cell or the top lut to data routing
FFT1	0-9		GIN	Output from either the bottom FF of the top hslf of the lab cell or the top lut to data routing
FFT1L	0-9		LD	Output from either the bottom FF of the top hslf of the lab cell or the top lut to local dispatch

2.2.2 MLAB

A MLAB is a lab that can optionally be turned into a 640-bits RAM or ROM. The wiring is identical to the LAB, only some additional muxes are provided to select the RAM/ROM mode.

TODO: address/data wiring in RAM/ROM mode.

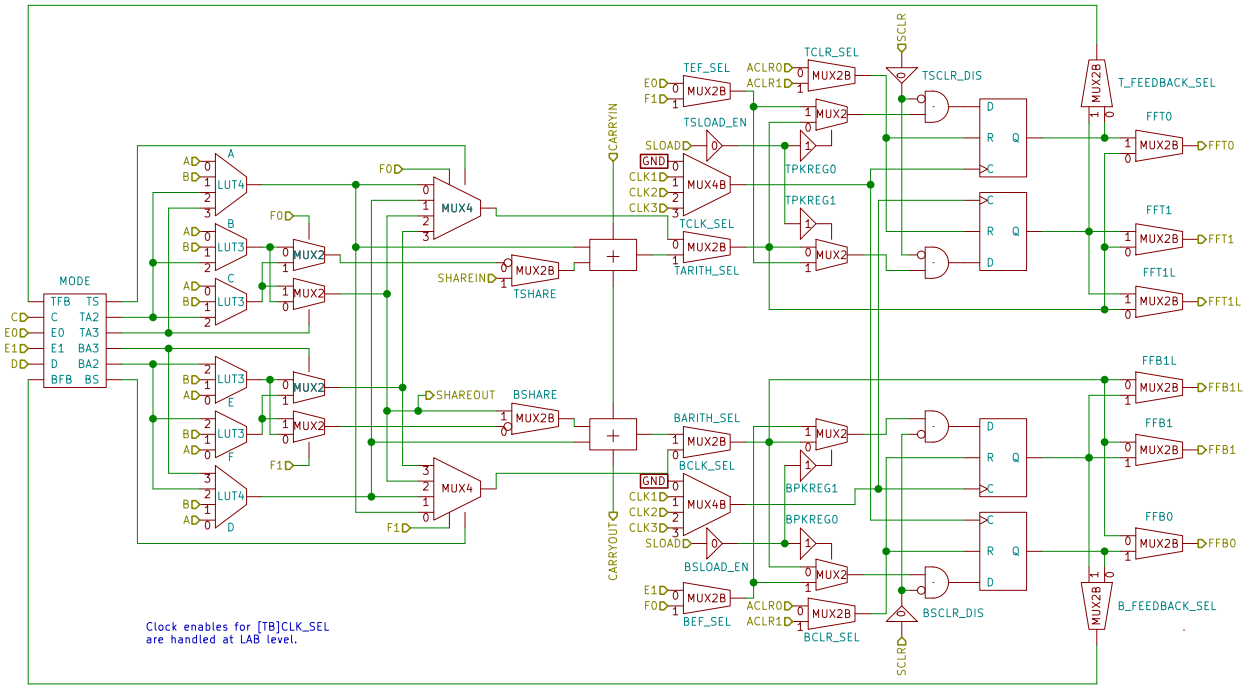

Fig. 2: One of the 10 cells of the LAB.

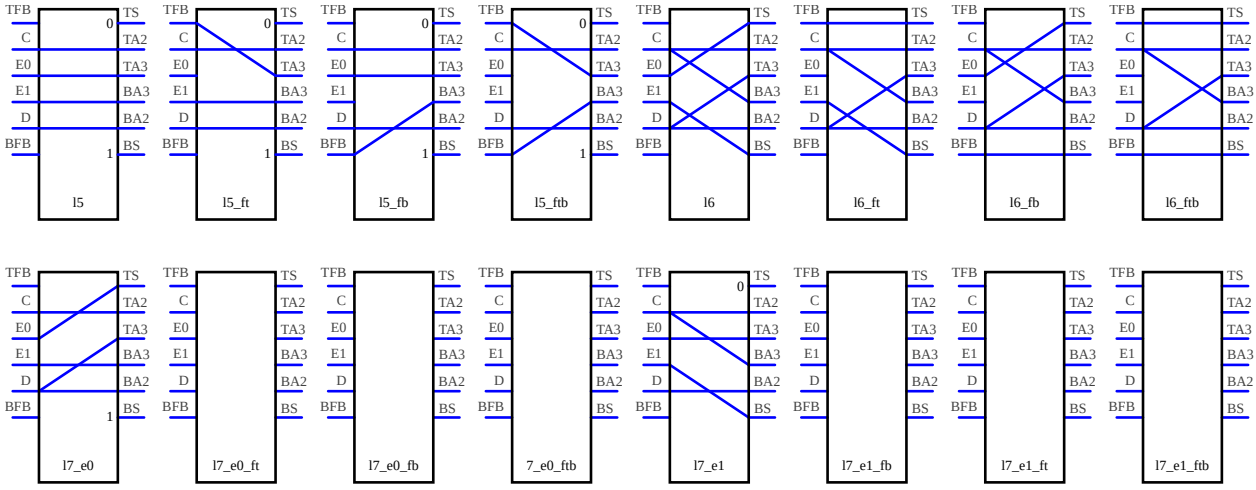

Fig. 3: The 16 possible interconnection modes.

Name	Instance	Type	Values	Default	Documentation
MADDG_VOLTAGE		Mux	• vccl • vcchg	vccl	TODO
MCRG_VOLTAGE		Mux	• vcchg • vccl	vcchg	TODO
RAM_DIS		Bool	t/f	t	TODO
REGSCAN_LATCH_EN		Bool	t/f	f	TODO
WRITE_EN		Bool	t/f	f	TODO
WRITE_PULSE_LENGTH		Num	• 500 • 650 • 800 • 950	500	TODO

2.2.3 DSP

The DSP blocks provide a multiply-adder with differents modes. Its large number of inputs and output makes it span two tiles vertically.

The modes are are:

- Three 9x9 multipliers in parallel
- Two 18x19 multipliers in parallel
- Two 18x19 multipliers with the results combined through add or sub
- One 18x18 multiplier added to a 36-bits value
- One 27x27 multiplier

Data input is through 12 blocks of 9 bits, the mapping of their use depending on the mode. Each bit can be individually inverted. Unconnected bits default to 1 and must be inverted to get a 0. We are only able to do 18x18 multipliers, 18x19 configuration is not understood.

The two operands of a multiplier are called X and Y. The Z operand is use in preadder mode and acts on Y. When in two-multiplier mode they are called A and B. Three-multiplier mode is very similar to single with the inputs and outputs packed in the 27-bits inputs/54-bits output registers. Preadder is not officially supported in 3-multiplier mode.

Mapping of data input blocks to multiplier ports is as follows:

Multiplier mode	AX	AY	AZ	BX	BY	BZ
1 or 3, no preadder	7, 6, 0	9, 8, 2				
3, preadder active	7, 6, 0	8, 3, 2	10, 5, 4			
2	1, 0	3, 2	5, 4	7, 6	9, 8	11, 10
18x18+36	1, 0	3, 2	5, 4	9, 8, 7, 6		

Result is in the single 74-bits wide RESULT port, which is split in half in two-18x19-parallel mode with the B result in bits [73:37].

Name	Instance	Type	Values	Default	Documentation
ACC_INV		Bool	t/f	f	TODO
ACLR0_INV		Bool	t/f	f	Invert aclr 0
ACLR0_SEL		Num	• 0 • 2	0	Input for aclr 0
ACLR1_INV		Bool	t/f	f	Invert aclr 1
ACLR1_SEL		Num	• 1 • 3	1	Input for aclr 1
AX_SIGNED		Bool	t/f	f	Is port X of multiplier A signed?
AY_SIGNED		Bool	t/f	f	Is port Y of multiplier A signed?
BX_SIGNED		Bool	t/f	f	Is port X of multiplier B signed?
BY_SIGNED		Bool	t/f	f	Is port Y of multiplier B signed?
CASCADE_1ST_EN		Bool	t/f	f	TODO
CASCADE_EN		Bool	t/f	f	TODO
CHAIN_OUTPUT_EN		Bool	t/f	f	TODO
CLK0_INV		Bool	t/f	f	Invert clock 0
CLK0_SEL		Num	• 0 • 3	0	Input for clock 0
CLK1_INV		Bool	t/f	f	Invert clock 1
CLK1_SEL		Num	• 1 • 4	1	Input for clock 1
CLK2_INV		Bool	t/f	f	Invert clock 2
CLK2_SEL		Num	• 2 • 5	2	Input for clock 2
CLK_AX17_SEL		Num	• 0-2	0	TODO
CLK_AYZ17_SEL		Num	• 0-2	0	TODO
CLK_BX17_SEL		Num	• 0-2	0	TODO
CLK_BYZ17_SEL		Num	• 0-2	0	TODO

continues on next page

Table 2 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
CLK_DYN_CTRL_SEL		Num	• 0-2	0	TODO
CLK_OPREG_SEL		Num	• 0-2	0	TODO
COEF_INPUT_EN		Bool	t/f	f	Use coefficient for multiplier port X
DEC_INV		Bool	t/f	f	TODO
DELAY_CASCADE_AY_EN		Bool	t/f	f	TODO
DELAY_CASCADE_BY_EN		Bool	t/f	f	TODO
DFT_CLK_DIS		Bool	t/f	t	TODO
DFT_ITG_EN		Bool	t/f	f	TODO
DFT_TDF_EN		Bool	t/f	f	TODO
DOUBLE_ACC_EN		Bool	t/f	f	TODO
ENABLE0_FORCE		Bool	t/f	f	Clock 0 always enabled
ENABLE0_INV		Bool	t/f	f	Invert enable on clock 0
ENABLE1_FORCE		Bool	t/f	f	Clock 1 always enabled
ENABLE1_INV		Bool	t/f	f	Invert enable on clock 1
ENABLE2_FORCE		Bool	t/f	f	Clock 2 always enabled
ENABLE2_INV		Bool	t/f	f	Invert enable on clock 2
IDIREG_ACC_CTRL		Mux	• bypass • reg	bypass	TODO
IDIREG_DEC_CTRL		Mux	• bypass • reg	bypass	TODO
IDIREG_PRELOAD_CTRL		Mux	• bypass • reg	bypass	TODO
IDIREG_SUB		Mux	• bypass • reg	bypass	TODO

continues on next page

Table 2 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
INREG_CTRL_AX		Mux	• bypass • reg	bypass	TODO
INREG_CTRL_AY		Mux	• bypass • reg	bypass	TODO
INREG_CTRL_AZ		Mux	• bypass • reg	bypass	TODO
INREG_CTRL_BX		Mux	• bypass • reg	bypass	TODO
INREG_CTRL_BY		Mux	• bypass • reg	bypass	TODO
INREG_CTRL_BZ		Mux	• bypass • reg	bypass	TODO
LOAD_VALUE		Ram	00-3f	0	Value to load in the accumulator (1<<n)
MODE		Mux	• m9x9 • m18x19 • m27x27 • m18x19_combined • m18x18p36	m18x19	Multiplication configuration
OREG_CTRL		Mux	• bypass • reg	bypass	TODO
PARTIAL_RECONFIG_EN		Bool	t/f	f	TODO
PREADDER_EN		Bool	t/f	f	Preadder activation
PREADDER_SUB		Bool	t/f	f	Preadder substraction mode
PRELOAD_INV		Bool	t/f	f	TODO
SUB_INV		Bool	t/f	f	TODO

continues on next page

Table 2 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
SYS-TOLIC_REG_EN		Bool	t/f	f	TODO
COEF_A	0-7	Ram	18 bits	0	Low 18 bits of the A multiplier coefficients
COEF_B	0-7	Ram	18 bits	0	High 9 bits of A or 18 bits of B multiplier coefficients
DATA_INV	0-11	Ram	000-1ff	0	Per-bit inversion of DATA_IN. Unconnected inputs default as 1 and should be inverted to get a 0.

Port Name	Instance	Port bits	Route node type	Documentation
ACCUMULATE			GOUT	TODO
ACLR		2-3	GOUT	Asynchronous clear inputs
ACLR		0-1	TCLK	Asynchronous clear inputs
CLKIN		3-5	GOUT	Clock inputs
CLKIN		0-2	TCLK	Clock inputs
DATAIN	0-11	0-8	GOUT	The 12 9-bit data input blocks
ENABLE		0-2	GOUT	Clock enable inputs
LOADCONST			GOUT	TODO
NEGATE			GOUT	TODO
RESULT		0-73	GIN	Final multiplication output
SUB			GOUT	TODO
UNK_IN		30-31, 62-63, 94-95, 126-127	GOUT	TODO

2.2.4 M10K

The M10K blocks provide 10240 (256*40) bits of dual-ported rom or ram.

TODO: everything, GOUT/GIN/DCMUX mapping is done

Name	Instance	Type	Values	Default	Documentation
A_ADDCLR_EN		Bool	t/f	f	TODO
A_DATA_FLOW_THRU		Bool	t/f	f	TODO

continues on next page

Table 3 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
A_DATA_WIDTH		Num	• 1-2 • 5 • 10 • 20 • 40	40	TODO
A_DMY_PWDWN		Ram	0-f	6	TODO
A_FAST_READ		Bool	t/f	f	TODO
A_FAST_WRITE		Mux	• off • fast • slow	off	TODO
A_OUTCLR_EN		Mux	• off • reg • lat	off	TODO
A_OUTEN_DELAY		Ram	0-7	1	TODO
A_OUTEN_PULSE		Ram	0-3	3	TODO
A_OUTPUT_SEL		Mux	• async • reg	async	TODO
A_SAEN_DELAY		Ram	0-7	0	TODO
A_SA_WREN_DELAY		Ram	0-3	0	TODO
A_WL_DELAY		Ram	0-3	1	TODO
A_WR_TIMER_PULSE		Ram	00-1f	06	TODO
BIST_MODE		Bool	t/f	f	TODO
BOT_1_ADDCLR_SEL		Num	• 0-1	0	TODO
BOT_1_CORECLK_SEL		Num	• 0-1	0	TODO
BOT_1_INCLK_SEL		Num	• 0-1	0	TODO
BOT_1_OUTCLK_SEL		Num	• 0-1	0	TODO
BOT_1_OUTCLR_SEL		Num	• 0-1	0	TODO
BOT_CE0_INV		Bool	t/f	f	TODO
BOT_CE0_SEL		Num	• 0-1	0	TODO

continues on next page

Table 3 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
BOT_CE1_INV		Bool	t/f	f	TODO
BOT_CE1_SEL		Num	• 0-1	0	TODO
BOT_CLK_INV		Bool	t/f	f	TODO
BOT_CLK_SEL		Num	• 0-1	0	TODO
BOT_CLR_INV		Bool	t/f	f	TODO
BOT_CLR_SEL		Num	• 0-1	0	TODO
BOT_CORECLK_SEL		Num	• 0-2	0	TODO
BOT_INCLK_SEL		Num	• 0-2	0	TODO
BOT_OUTCLK_SEL		Num	• 0-1	0	TODO
BOT_R_INV		Bool	t/f	f	TODO
BOT_R_SEL		Num	• 0-2	0	TODO
BOT_W_INV		Bool	t/f	f	TODO
BOT_W_SEL		Num	• 0-2	0	TODO
B_ADDCLR_EN		Bool	t/f	f	TODO
B_DATA_FLOW_THRU		Bool	t/f	f	TODO
B_DATA_WIDTH		Num	• 1-2 • 5 • 10 • 20 • 40	1	TODO
B_DMY_DELAY		Ram	0-3	1	TODO
B_DMY_DELAY		Ram	0-3	1	TODO
B_DMY_PWDWN		Ram	0-f	6	TODO
B_FAST_READ		Bool	t/f	f	TODO
B_FAST_WRITE		Mux	• off • fast • slow	off	TODO

continues on next page

Table 3 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
B_OUTCLR_EN		Mux	• off • reg • lat	off	TODO
B_OUTEN_DELAY		Ram	0-7	1	TODO
B_OUTEN_PULSE		Ram	0-3	3	TODO
B_OUTPUT_SEL		Mux	• async • reg	async	TODO
B_SAEN_DELAY		Ram	0-7	0	TODO
B_SA_WREN_DELAY		Ram	0-3	0	TODO
B_WL_DELAY		Ram	0-3	1	TODO
B_WR_TIMER_PULSE		Ram	00-1f	06	TODO
DISABLE_UNUSED		Bool	t/f	t	TODO
ITG_LFSR		Bool	t/f	f	TODO
PACK_MODE		Bool	t/f	f	TODO
PR_EN		Bool	t/f	f	TODO
TDF_ATPG		Bool	t/f	f	TODO
TEST_MODE_OFF		Bool	t/f	t	TODO
TOP_ADDCLR_SEL		Num	• 0-1	0	TODO
TOP_CE0_INV		Bool	t/f	f	TODO
TOP_CE0_SEL		Num	• 0-1	0	TODO
TOP_CE1_INV		Bool	t/f	f	TODO
TOP_CE1_SEL		Num	• 0-1	0	TODO
TOP_CLK_INV		Bool	t/f	f	TODO
TOP_CLK_SEL		Num	• 0-1	0	TODO
TOP_CLR_INV		Bool	t/f	f	TODO
TOP_CLR_SEL		Num	• 0-1	0	TODO
TOP_CORECLK_SEL		Num	• 0-2	0	TODO
TOP_INCLK_SEL		Num	• 0-2	0	TODO

continues on next page

Table 3 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
TOP_OUTCLK_SEL		Num	• 0-1	0	TODO
TOP_OUTCLR_SEL		Num	• 0-1	0	TODO
TOP_R_INV		Bool	t/f	f	TODO
TOP_R_SEL		Num	• 0-2	0	TODO
TOP_W_INV		Bool	t/f	f	TODO
TOP_W_SEL		Num	• 0-2	0	TODO
TRUE_DUAL_PORT		Bool	t/f	f	TODO
RAM	0-255	Ram	40 bits	0	TODO

Port Name	Instance	Port bits	Route node type	Documentation
ACLR		0-1	GOUT	Asynchronous clear
ADDRA		0-11	GOUT	Address for port A
ADDRB		0-11	GOUT	Address for port B
ADDRSTALLA			GOUT	Lock address on port A
ADDRSTALLB			GOUT	Lock address on port B
BYTEENABLEA		0-1	GOUT	Write enables for the two halves of port A
BYTEENABLEB		0-1	GOUT	Write enables for the two halves of port B
CLKIN		6-7	GOUT	Clock inputs, only 0-1 and 6-7 used
CLKIN		0-5	TCLK	Clock inputs, only 0-1 and 6-7 used
DATAAIN		0-19	GOUT	Input data for port A
DATAAOUT		0-19	GIN	Output data for port A
DATABIN		0-19	GOUT	Input data for port B
DATABOUT		0-19	GIN	Output data for port A
ENABLE		0-3	GOUT	Clock enables
RDEN		0-1	GOUT	Read enables
WREN		0-1	GOUT	Write enables

2.3 Peripheral logic blocks

2.3.1 GPIO

The GPIO blocks connect the FPGA with the exterior through the package pins. Each block controls 4 pads, which are connected to up to 4 pins.

TODO: everything, GOUT/GIN/DCMUX mapping is done

Name	Instance	Type	Values	Default	Documentation
IOCSR_STD	0-3	Mux	• nvr_high • nvr_low • vr • dis	nvr_high	TODO
OUTPUT_DUTY_CYCLE_DELAY_FALL	0-3	Bool	t/f	f	TODO
OUTPUT_DUTY_CYCLE_DELAY_PS	0-3	Num	• 0 • 50 • 100 • 150	0	TODO
OUTPUT_DUTY_CYCLE_DELAY_RISE	0-3	Bool	t/f	f	TODO
PLL_SELECT	0-3	Mux	• codin • pll	codin	TODO
SLEW_RATE_SLOW	0-3	Bool	t/f	f	TODO
TERMINATION_CONTROL	0-3	Mux	• regio • rupdn	regio	TODO
TERMINATION_CONTROL_SHIFT	0-3	Bool	t/f	f	TODO
TERMINATION_MODE	0-3	Mux	• pds • rs_static • rt_pds_dynamic • rt_rs_dynamic • rt_static	pds	TODO
USE_BUS_HOLD	0-3	Bool	t/f	f	TODO
USE_OPEN_DRAIN	0-3	Bool	t/f	f	TODO
USE_PCI_DIODE_CLAMP	0-3	Bool	t/f	f	TODO
USE_WEAK_PULLUP	0-3	Bool	t/f		TODO
DRIVE_STRENGTH	0-3	Mux	• off • prog_gnd • prog_pwr • lvds_1r • lvds_3r • v3p0_pci_pcix • v3p0_lvttl_4ma • v3p0_lvttl_8ma • v3p0_lvttl_12ma • v3p0_lvttl_16ma • v3p3_lvttl_4ma • v3p0_lvcmos_4ma		TODO

Port Name	Instance	Port bits	Route node type	Documentation
ACLR	0-3		GOUT	TODO
BSLIPMAX	0-3		GIN	TODO
CEIN	0-3		GOUT	TODO
CEOUT	0-3		GOUT	TODO
CLKIN_IN	0-3	0-1	DCMUX	TODO
CLKIN_OUT	0-3	0-1	DCMUX	TODO
DATAIN	0-3	0-3	GOUT	TODO
DATAOUT	0-3	0-4	GIN	TODO
OEIN	0-3	0-1	GOUT	TODO
SCLR	0-3		GOUT	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
ACLR	0-3		<	HMC:PHYDDIOADDRACLR	TODO
ACLR	1		<	HMC:PHYDDIOBAACLR	TODO
ACLR	2		<	HMC:PHYDDIOCASNACLR	TODO
ACLR	2-3		<	HMC:PHYDDIOCKEACLR	TODO
ACLR	0-1		<	HMC:PHYDDIOCSNACLR	TODO
ACLR	2-3		<	HMC:PHYDDIOODTACLR	TODO
ACLR	3		<	HMC:PHYDDIORASNACLR	TODO
ACLR	2		<	HMC:PHYDDIORESETNACLR	TODO
ACLR	2		<	HMC:PHYDDIOWENACLR	TODO
COMBOUT	0		>	CMUXCR:CLKPIN	Raising-edge clock pin to clock mux
COMBOUT	1		>	CMUXCR:NCLKPIN	Falling-edge clock pin to clock mux
COMBOUT	0		>	CMUXHG:CLKPIN	Raising-edge clock pin to clock mux
COMBOUT	1		>	CMUXHG:NCLKPIN	Falling-edge clock pin to clock mux
COMBOUT	0		>	CMUXHR:CLKPIN	Raising-edge clock pin to clock mux
COMBOUT	1		>	CMUXHR:NCLKPIN	Falling-edge clock pin to clock mux
COMBOUT	0		>	CMUXVG:CLKPIN	Raising-edge clock pin to clock mux
COMBOUT	1		>	CMUXVG:NCLKPIN	Falling-edge clock pin to clock mux
COMBOUT	0		>	CMUXVR:CLKPIN	Raising-edge clock pin to clock mux
COMBOUT	1		>	CMUXVR:NCLKPIN	Falling-edge clock pin to clock mux
COMBOUT	0		>	FPLL:CLKIN	Raising-edge or differential clock pin to pll
COMBOUT	2		>	FPLL:ZDB_IN	Zero-delay buffer pin to pll
DATAIN	0-3	0-3	<	HMC:PHYDDIOADDRDOUT	TODO
DATAIN	0-2	0-3	<	HMC:PHYDDIOBADOUT	TODO
DATAIN	2	0-3	<	HMC:PHYDDIOCASNDOUT	TODO
DATAIN	0	0-3	<	HMC:PHYDDIOCKDOUT	TODO
DATAIN	2-3	0-3	<	HMC:PHYDDIOCKEDOUT	TODO
DATAIN	1	0-3	<	HMC:PHYDDIOCKNDOUT	TODO
DATAIN	0-1	0-3	<	HMC:PHYDDIOCSNDOUT	TODO
DATAIN	2	0-3	<	HMC:PHYDDIODMDOUT	TODO
DATAIN	0-3	0-3	<	HMC:PHYDDIODQDOUT	TODO
DATAIN	1	0-3	<	HMC:PHYDDIODQSBDOUT	TODO
DATAIN	0	0-3	<	HMC:PHYDDIODQSDOUT	TODO
DATAIN	2-3	0-3	<	HMC:PHYDDIOODTDOUT	TODO
DATAIN	3	0-3	<	HMC:PHYDDIORASNDOUT	TODO
DATAIN	2	0-3	<	HMC:PHYDDIORESETNDOUT	TODO
DATAIN	2	0-3	<	HMC:PHYDDIOWENDOUT	TODO
DATAOUT	0-3	0-3	>	HMC:DDIOPHYDQDIN	TODO

continues on next page

Table 4 – continued from previous page

Port Name	Instance	Port bits	Dir	Remote port	Documentation
OEIN	0-3	0-1	<	HMC:PHYDDIODQOE	TODO
OEIN	1	0-1	<	HMC:PHYDDIODQSBOE	TODO
OEIN	0	0-1	<	HMC:PHYDDIODQSOE	TODO
PLLDIN	3		<	FPLL:EXTCLK	TODO

2.3.2 DQS16

The DQS16 blocks handle differential signaling protocols. Each supervises 4 GPIO blocks for a total of 16 signals, hence their name.

TODO: everything

Name	Instance	Type	Values	Default	Documentation
ADDR_DQS_DELAY_CHAIN_LENGTH		Ram	0-3	0	TODO
DELAY_CHAIN_CONTROL_INPUT		Mux	• dll1in • dll2in • core_in • sel_0	dll1in	TODO
DELAY_CHAIN_LATCHES_BYPASS		Bool	t/f	f	TODO
DFT_RB_RSCANOVRD_REG_EN		Bool	t/f	f	TODO
DFT_RB_RSCANOVRD_TDF_EN		Bool	t/f	f	TODO
DQS_BUS_WIDTH		Num	• 0 • 8 • 16 • 32	8	TODO
DQS_DELAY_CHAIN_PWDOWN_DBG_DEF_DIS		Bool	t/f	t	TODO
DQS_DELAY_CHAIN_PWDOWN_DBG_SEL_DEF_DIS		Bool	t/f	f	TODO
DQS_DELAY_CHAIN_RB_ADDI_EN		Bool	t/f	f	TODO
DQS_DELAY_CHAIN_RB_CO		Ram	0-3	3	TODO
DQS_DELAY_CHAIN_TWO_DLY_EN		Bool	t/f	t	TODO
DQS_ENABLE_SEL		Mux	• combi_pst • pst • ht_pst • pst_ena	combi_pst	TODO
DQS_PHASE_TRANSFER_NEG_EN		Bool	t/f	f	TODO
DQS_POSTAMBLE_EN		Bool	t/f	f	TODO
DQS_POSTAMBLE_NEJ_SEL		Mux	• cff • ip_sc	cff	TODO

continues on next page

Table 5 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
DQS_PWR_SVG_EN		Bool	t/f	t	TODO
HR_CLK_PST_INV		Bool	t/f	t	TODO
HR_CLK_PST_SEL		Mux	• dqs_clkout • seq_hr_clk	seq_hr_clk	TODO
PST_DQS_CLK_INV_PHASE_INV		Bool	t/f	f	TODO
PST_DQS_CLK_INV_PHASE_SEL		Mux	• cff • ip_sc	cff	TODO
PST_DQS_DELAY_CHAIN_LENGTH		Ram	0-3	0	TODO
PST_USE_PHASECTRLIN		Bool	t/f	f	TODO
RBT_BYPASS_VAL		Ram	0-1	0	TODO
RBT_NEJ_OCT_HALFT_EN		Bool	t/f	f	TODO
RB_2X_CLK_DQS_EN		Bool	t/f	f	TODO
RB_2X_CLK_DQS_INV		Bool	t/f	f	TODO
RB_2X_CLK_OCT_EN		Bool	t/f	f	TODO
RB_2X_CLK_OCT_INV		Bool	t/f	f	TODO
RB_ACLR_LFIFO_EN		Bool	t/f	f	TODO
RB_ACLR_PST_EN		Bool	t/f	f	TODO
RB_BYP_OCT_SEL		Mux	• combi • reg • reg_2x • bypass_val	bypass_val	TODO
RB_CLK_AC_EN		Bool	t/f	f	TODO
RB_CLK_AC_INV		Bool	t/f	t	TODO
RB_CLK_DQ_EN		Bool	t/f	f	TODO
RB_CLK_HR_EN		Bool	t/f	f	TODO
RB_CLK_OP_EN		Bool	t/f	f	TODO
RB_CLK_OP_SEL		Mux	• clk0 • delay_clk	clk0	TODO
RB_CLK_PST_EN		Bool	t/f	f	TODO
RB_FIFO_WEN_EN		Bool	t/f	f	TODO
RB_FR_CLK_OCT_EN		Bool	t/f	f	TODO
RB_FR_CLK_OCT_INV		Bool	t/f	f	TODO
RB_FR_CLK_OCT_SEL		Mux	• clk_out_1 • seq_hr_clk	clk_out_1	TODO
RB_HR_BYPASS_CFF_EN		Bool	t/f	t	TODO

continues on next page

Table 5 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RB_HR_BYPASS_SEL_IPEN		Mux	• cff • ip_sc	cff	TODO
RB_HR_CLK_OCT_EN		Bool	t/f	f	TODO
RB_HR_CLK_OCT_INV		Bool	t/f	f	TODO
RB_HR_CLK_OCT_SEL		Mux	• clk_out_1 • seq_hr_clk	clk_out_1	TODO
RB_LFIFO		Ram	32 bits	0	TODO
RB_LFIFO_BYPASS		Bool	t/f	t	TODO
RB_LFIFO_OCT_EN		Bool	t/f	t	TODO
RB_LFIFO_PHY_CLK_INV		Bool	t/f	f	TODO
RB_LFIFO_PHY_CLK_SEL		Ram	0-1	0	TODO
RB_T11_GATING_SEL_CFF		Ram	00-1f	0	TODO
RB_T11_GATING_SEL_IPEN		Mux	• cff • ip_sc	cff	TODO
RB_T11_UNGATING_SEL_CFF		Ram	00-1f	0	TODO
RB_T11_UNGATING_SEL_IPEN		Mux	• cff • ip_sc	cff	TODO
RB_T7_DQS_SEL_DQS_IPEN		Mux	• cff • ip_sc	cff	TODO
RB_T7_SEL_IREG_CFF_DELAY		Ram	00-1f	0	TODO
RB_T9_SEL_OCT_CFF		Ram	00-1f	0	TODO
RB_T9_SEL_OCT_IPEN		Mux	• cff • ip_sc	cff	TODO
RB_VFIFO_EN		Bool	t/f	f	TODO
RDFT_ITG_XOR_EN		Bool	t/f	f	TODO
RX-CLK_01_SEL		Ram	0-1	0	TODO
RX-CLK_45_SEL		Ram	0-1	0	TODO
RX-CLK_89_SEL		Ram	0-1	0	TODO
RX-CLK_CD_SEL		Ram	0-1	0	TODO
TX-CLK_23_SEL		Ram	0-1	0	TODO

continues on next page

Table 5 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
TX-CLK_67_SEL		Ram	0-1	0	TODO
TX-CLK_AB_SEL		Ram	0-1	0	TODO
TX-CLK_EF_SEL		Ram	0-1	0	TODO
UP-DATE_ENABLE_INPUT		Mux	• sel1 • sel2 • core • sel0	sel1	TODO
BITSLIP_CFG	0-15	Num	• 1-11	1	TODO
CE_OEREG_TIEOFF_EN	0-15	Bool	t/f	f	TODO
CE_OUTREG_TIEOFF_EN	0-15	Bool	t/f	f	TODO
DDIO_OE_EN	0-15	Bool	t/f	f	TODO
DQS_CLK_SEL	0-15	Mux	• clkout0 • dq_clk • dqs_clk • addr_clk	clkout0	TODO
FIFO_MODE_SEL	0-15	Mux	• fifo_hr_mode • fifo_fr_mode • bit-slip_mode • des_bs_input • des_io_input • ser_output	fifo_hr_mode	TODO
FIFO_RCLK_IPEN	0-15	Mux	• cff • ip_sc	cff	TODO
FIFO_RCLK_SEL	0-15	Mux	• clkin1 • dqs_clk • seq_hr_clk • vcc	vcc	TODO

continues on next page

Table 5 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
INPUT_PATH_CE_IN	0-15	Bool	t/f	f	TODO
INPUT_REG0_SEL	0-15	Mux	• sel_bypass • sel_group_fifo0 • sel_cdatamxin0 • sel_cdatamxin5	sel_bypass	TODO
INPUT_REG1_SEL	0-15	Mux	• sel_bypass • sel_group_fifo1 • sel_cdatamxin1 • sel_cdatamxin6	sel_bypass	TODO
INPUT_REG2_SEL	0-15	Mux	• sel_bypass • sel_group_fifo2 • sel_cdatamxin2 • sel_cdatamxin7	sel_bypass	TODO
INPUT_REG3_SEL	0-15	Mux	• sel_bypass • sel_group_fifo3 • sel_cdatamxin3 • sel_cdatamxin8	sel_bypass	TODO

continues on next page

Table 5 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
IN-PUT_REG4_SEL	0-15	Mux	• sel_bypass • sel_locked_dpa • sel_cdatamxin4 • sel_cdatamxin9	sel_bypass	TODO
IN-REG_POWER_UP_STATE	0-15	Ram	0-1	0	TODO
IN-REG_SCLR_EN	0-15	Bool	t/f	f	TODO
IN-REG_SCLR_VAL	0-15	Ram	0-1	0	TODO
IOREG_PWR_SVG_EN	0-15	Bool	t/f	t	TODO
IP_SC_OR_FIFO_SEL	0-15	Mux	• cff • ip_sc	cff	TODO
IR_FIFO_RCLK_INV	0-15	Bool	t/f	f	TODO
IR_FIFO_TCLK_EN	0-15	Bool	t/f	f	TODO
OEREG_ACLR_EN	0-15	Bool	t/f	f	TODO
OEREG_CLK_INV	0-15	Bool	t/f	f	TODO
OEREG_HR_CLK_EN	0-15	Bool	t/f	f	TODO
OEREG_OUTPUT_SEL	0-15	Mux	• sel_oe0 • sel_1x • sel_1x_delay • sel_2x	sel_oe0	TODO
OEREG_POWER_UP_STATE	0-15	Ram	0-1	0	TODO
OEREG_SCLR_DREG	0-15	Ram	0-1	0	TODO
OEREG_SCLR_EN	0-15	Bool	t/f	f	TODO
OE_2X_CLK_EN	0-15	Bool	t/f	f	TODO
OE_2X_CLK_INV	0-15	Bool	t/f	f	TODO
OE_HALF_RATE_BYPASS	0-15	Bool	t/f	t	TODO
OE_HALF_RATE_EN	0-15	Mux	• cff • ip_sc	cff	TODO
OUT-REG_MODE_SEL	0-15	Mux	• sdr • ddr	sdr	TODO

continues on next page

Table 5 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
OUT-REG_OUTPUT_SEL	0-15	Mux	• sel_iodout0 • sel_sdr • sel_sdr_delay • sel_2xff	sel_iodout0	TODO
OUT-REG_POWER_UP_STATE	0-15	Ram	0-1	0	TODO
OUT-REG_SCLR_EN	0-15	Bool	t/f	f	TODO
OUT-REG_SCLR_VAL	0-15	Ram	0-1	0	TODO
RBE_HRATE_CLK_SEL	0-15	Mux	• clkout1 • hr_clk	clkout1	TODO
RBOE_LVL_FR_CLK_EN	0-15	Bool	t/f	f	TODO
RBOE_LVL_FR_CLK_INV	0-15	Bool	t/f	f	TODO
RB_FIFO_WCLK_EN	0-15	Bool	t/f	f	TODO
RB_FIFO_WCLK_INV	0-15	Bool	t/f	f	TODO
RB_FIFO_WCLK_SEL	0-15	Mux	• clkin0 • dqs_bus	clkin0	TODO
RB_IREG_T1T1_BYPASS_EN	0-15	Bool	t/f	f	TODO
RB_OEO_INV	0-15	Bool	t/f	t	TODO
RB_T1_SEL_IREG_CFF_DELAY	0-15	Ram	00-1f	0	TODO
RB_T1_SEL_IREG_IPEN	0-15	Mux	• cff • ip_sc	cff	TODO
RB_T9_SEL_EREG_CFF_DELAY	0-15	Ram	00-1f	0	TODO
RB_T9_SEL_EREG_IPEN	0-15	Mux	• cff • ip_sc	cff	TODO
RB_T9_SEL_OREG_CFF_DELAY	0-15	Ram	00-1f	0	TODO
RB_T9_SEL_OREG_IPEN	0-15	Mux	• cff • ip_sc	cff	TODO
SET_T3_FOR_CDATA0IN	0-15	Ram	0-7	0	TODO
SET_T3_FOR_CDATA1IN	0-15	Ram	0-7	0	TODO
TX-OUT_FCLK_SEL	0-15	Mux	• txout • fclk	txout	TODO

continues on next page

Table 5 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
USE_CLR_INREG0_EN	0-1	Bool	t/f	f	TODO
USE_CLR_OUTREG_EN	0-1	Bool	t/f	f	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
			<	HMC	TODO

2.3.3 FPLL

The Fractional PLL blocks synthesize 9 frequencies from an input with integer or fractional ratios.

TODO: everything, GOUT/GIN/DCMUX mapping is done

Name	Instance	Type	Values	Default	Documentation
ATB		Ram	0-f	0	TODO
AUTO_CLK_SW_EN		Bool	t/f	f	TODO
BWCTRL		Ram	0-f	4	TODO
C0_COUT_EN		Bool	t/f	f	TODO
C0_EXTCLK_DLLOUT_EN		Bool	t/f	f	TODO
C1_COUT_EN		Bool	t/f	f	TODO
C1_EXTCLK_DLLOUT_EN		Bool	t/f	f	TODO
C2_COUT_EN		Bool	t/f	f	TODO
C2_EXTCLK_DLLOUT_EN		Bool	t/f	f	TODO
C3_COUT_EN		Bool	t/f	f	TODO
C3_EXTCLK_DLLOUT_EN		Bool	t/f	f	TODO
C4_COUT_EN		Bool	t/f	f	TODO
C5_COUT_EN		Bool	t/f	f	TODO
C6_COUT_EN		Bool	t/f	f	TODO
C7_COUT_EN		Bool	t/f	f	TODO
C8_COUT_EN		Bool	t/f	f	TODO
CLKIN_0_SRC		Ram	0-f	2	TODO
CLKIN_1_SRC		Ram	0-f	3	TODO
CLK_LOSS_EDGE		Ram	0-1	0	TODO
CLK_LOSS_SW_EN		Bool	t/f	f	TODO
CLK_SW_DELAY		Ram	0-7	0	TODO
CMP_BUF_DELAY		Ram	0-7	0	TODO
CP_COMP		Bool	t/f	f	TODO
CP_CURRENT		Ram	0-7	2	TODO
CTRL_OVERRIDE_SETTING		Bool	t/f	t	TODO
DLL_SRC		Ram	00-1f	1c	TODO
DPADIV_VCOPH_DIV		Ram	0-3	0	TODO
DPRIO0_BASE_ADDR		Ram	00-3f	0	TODO
DPRIO_DPS_ATPGMODE_INVERT		Bool	t/f	f	TODO
DPRIO_DPS_CLK_INVERT		Bool	t/f	f	TODO
DPRIO_DPS_CSR_TEST_INVERT		Bool	t/f	f	TODO
DPRIO_DPS_ECN_MUX		Ram	0-1	0	TODO
DPRIO_DPS_RESERVED_INVERT		Bool	t/f	f	TODO
DPRIO_DPS_RST_N_INVERT		Bool	t/f	f	TODO

continues on next page

Table 6 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
DPRIO_DPS_SCANEN_INVERT		Bool	t/f	f	TODO
DSM_DITHER		Ram	0-3	0	TODO
DSM_OUT_SEL		Ram	0-3	0	TODO
DSM_RESET		Bool	t/f	f	TODO
ECN_BYPASS		Bool	t/f	f	TODO
ECN_TEST_EN		Bool	t/f	f	TODO
FBCLK_MUX_1		Ram	0-3	0	TODO
FBCLK_MUX_2		Ram	0-1	0	TODO
FORCELOCK		Bool	t/f	f	TODO
FPLL_ENABLE		Bool	t/f	f	TODO
FRACTIONAL_CARRY_OUT		Ram	0-3	3	TODO
FRACTIONAL_DIVISION_SETTING		Ram	32 bits	0	TODO
FRACTIONAL_VALUE_READY		Bool	t/f	t	TODO
LF_TESTEN		Bool	t/f	f	TODO
LOCK_FILTER_CFG_SETTING		Ram	000-fff	001	TODO
LOCK_FILTER_TEST		Bool	t/f	f	TODO
MANUAL_CLK_SW_EN		Bool	t/f	f	TODO
M_CNT_BYPASS_EN		Bool	t/f	f	TODO
M_CNT_COARSE_DELAY		Ram	0-7	0	TODO
M_CNT_FINE_DELAY		Ram	0-3	0	TODO
M_CNT_HI_DIV_SETTING		Ram	00-ff	01	TODO
M_CNT_IN_SRC		Ram	0-3	0	TODO
M_CNT_LO_DIV_SETTING		Ram	00-ff	01	TODO
M_CNT_LO_PRESET_SETTING		Ram	00-ff	01	TODO
M_CNT_ODD_DIV_DUTY_EN		Bool	t/f	f	TODO
M_CNT_PH_MUX_PRESET_SETTING		Ram	0-7	0	TODO
NREVERT_INVERT		Bool	t/f	f	TODO
N_CNT_BYPASS_EN		Bool	t/f	f	TODO
N_CNT_COARSE_DELAY		Ram	0-7	0	TODO
N_CNT_FINE_DELAY		Ram	0-3	0	TODO
N_CNT_HI_DIV_SETTING		Ram	00-ff	01	TODO
N_CNT_LO_DIV_SETTING		Ram	00-ff	01	TODO
N_CNT_ODD_DIV_DUTY_EN		Bool	t/f	f	TODO
PL_AUX_ATB		Bool	t/f	f	TODO
PL_AUX_ATB_COMP_MINUS		Bool	t/f	f	TODO
PL_AUX_ATB_COMP_PLUS		Bool	t/f	f	TODO
PL_AUX_ATB_EN0		Bool	t/f	f	TODO
PL_AUX_ATB_EN0_PRECOMP		Bool	t/f	f	TODO
PL_AUX_ATB_EN1		Bool	t/f	f	TODO
PL_AUX_ATB_EN1_PRECOMP		Bool	t/f	f	TODO
PL_AUX_ATB_MODE		Ram	00-1f	0	TODO
PL_AUX_BG_KICKSTART		Bool	t/f	f	TODO
PL_AUX_BG_POWERDOWN		Bool	t/f	f	TODO
PL_AUX_BYPASS_MODE_CTRL_CURRENT		Bool	t/f	f	TODO
PL_AUX_BYPASS_MODE_CTRL_VOLTAGE		Bool	t/f	f	TODO
PL_AUX_COMP_POWERDOWN		Bool	t/f	f	TODO
PL_AUX_VBGMON_POWERDOWN		Bool	t/f	f	TODO
PM_AUX_CAL_CLK_TEST_SEL		Bool	t/f	f	TODO
PM_AUX_CAL_RESULT_STATUS		Bool	t/f	f	TODO

continues on next page

Table 6 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PM_AUX_IQCLK_CAL_CLK_SEL		Ram	0-7	0	TODO
PM_AUX_RX_IMP		Ram	0-3	0	TODO
PM_AUX_TERM_CAL		Bool	t/f	f	TODO
PM_AUX_TERM_CAL_RX_OVER_VAL		Ram	00-1f	0	TODO
PM_AUX_TERM_CAL_RX_OVER_VAL_EN		Bool	t/f	f	TODO
PM_AUX_TERM_CAL_TX_OVER_VAL		Ram	00-1f	0	TODO
PM_AUX_TERM_CAL_TX_OVER_VAL_EN		Bool	t/f	f	TODO
PM_AUX_TEST_COUNTER		Bool	t/f	f	TODO
PM_AUX_TX_IMP		Ram	0-3	0	TODO
REF_BUF_DELAY		Ram	0-7	0	TODO
REGULATION_BYPASS		Bool	t/f	f	TODO
REG_BOOST		Ram	0-7	0	TODO
RIPPLECAP_CTRL		Ram	0-3	0	TODO
SLF_RST		Ram	0-3	0	TODO
SW_REFCLK_SRC		Ram	0-1	0	TODO
TCLK_MUX_EN		Bool	t/f	f	TODO
TCLK_SEL		Ram	0-1	1	TODO
TESTDN_ENABLE		Bool	t/f	f	TODO
TESTUP_ENABLE		Bool	t/f	f	TODO
TEST_ENABLE		Bool	t/f	f	TODO
UNLOCK_FILTER_CFG_SETTING		Ram	0-7	0	TODO
VC0DIV_OVERRIDE		Bool	t/f	t	TODO
VCCD0G_ATB		Ram	0-3	0	TODO
VCCD0G_OUTPUT		Ram	0-7	0	TODO
VCCD1G_ATB		Ram	0-3	0	TODO
VCCD1G_OUTPUT		Ram	0-7	0	TODO
VCCM1G_TAP		Ram	0-f	b	TODO
VCCR_PD		Bool	t/f	f	TODO
VCO0PH_EN		Bool	t/f	f	TODO
VCO_DIV		Ram	0-1	1	TODO
VCO_PH0_EN		Bool	t/f	f	TODO
VCO_PH1_EN		Bool	t/f	f	TODO
VCO_PH2_EN		Bool	t/f	f	TODO
VCO_PH3_EN		Bool	t/f	f	TODO
VCO_PH4_EN		Bool	t/f	f	TODO
VCO_PH5_EN		Bool	t/f	f	TODO
VCO_PH6_EN		Bool	t/f	f	TODO
VCO_PH7_EN		Bool	t/f	f	TODO
VCTRL_TEST_VOLTAGE		Ram	0-7	3	TODO
EXTCLK_CNT_SRC	0-1	Ram	00-1f	1c	TODO
EXTCLK_ENABLE	0-1	Bool	t/f	t	TODO
EXTCLK_INVERT	0-1	Bool	t/f	f	TODO
BYPASS_EN	0-8	Bool	t/f	f	TODO
CNT_COARSE_DELAY	0-8	Ram	0-7	0	TODO
CNT_FINE_DELAY	0-8	Ram	0-3	0	TODO
CNT_IN_SRC	0-8	Ram	0-3	2	TODO
CNT_PH_MUX_PRESET	0-8	Ram	0-7	0	TODO
CNT_PRESET	0-8	Ram	00-ff	01	TODO
DPRIO0_CNT_HI_DIV	0-8	Ram	00-ff	01	TODO

continues on next page

Table 6 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
DPRIO0_CNT_LO_DIV	0-8	Ram	00-ff	01	TODO
DPRIO0_CNT_ODD_DIV_EVEN_DUTY_EN	0-8	Bool	t/f	f	TODO
SRC	0-8	Bool	t/f	f	TODO
LOADEN_COARSE_DELAY	0-1	Ram	0-7	0	TODO
LOADEN_ENABLE	0-1	Bool	t/f	f	TODO
LOADEN_FINE_DELAY	0-1	Ram	0-3	0	TODO
LVDSCLK_COARSE_DELAY	0-1	Ram	0-7	0	TODO
LVDSCLK_ENABLE	0-1	Bool	t/f	f	TODO
LVDSCLK_FINE_DELAY	0-1	Ram	0-3	0	TODO

Port Name	Instance	Port bits	Route node type	Documentation
ATPGMODE			GOUT	TODO
CLK0_BAD			GIN	TODO
CLK1_BAD			GIN	TODO
CLKEN		0-1	GOUT	TODO
CLKSEL			GIN	TODO
CNT_SEL		0-4	GOUT	TODO
CSR_TEST			GOUT	TODO
EXTSWITCH			GOUT	TODO
FBCLK_IN_L			DCMUX	TODO
FBCLK_IN_R			DCMUX	TODO
LOCK			GIN	TODO
NRESET			GOUT	TODO
PFDEN			GOUT	TODO
PHASE_DONE			GIN	TODO
PHASE_EN			GOUT	TODO
REG_BYTE_EN		0-1	GOUT	TODO
REG_CLK			DCMUX	TODO
REG_CLK			GOUT	TODO
REG_MDIO_DIS			GOUT	TODO
REG_READ			GOUT	TODO
REG_READDATA		0-15	GIN	TODO
REG_REG_ADDR		0-5	GOUT	TODO
REG_RST_N			GOUT	TODO
REG_SER_SHIFT_LOAD			GOUT	TODO
REG_WRITE			GOUT	TODO
REG_WRITEDATA		0-15	GOUT	TODO
SCANEN			GOUT	TODO
UP_DN			GOUT	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
CLKDOUT		0	>	DLL:CLKIN	Dedicated differential I/O PLL counter to DLL
CLKIN		0-3	<	GPIO:COMBOUT	Raising-edge or differential clock pin to pll
CLKOUT		0-8	>	CMUXCR:PLLIN	PLL counter output to clock mux
CLKOUT		0-8	>	CMUXHG:PLLIN	PLL counter output to clock mux
CLKOUT		0-8	>	CMUXHR:PLLIN	PLL counter output to clock mux
CLKOUT		5-8	>	CMUXVG:PLLIN	PLL counter output to clock mux
CLKOUT		0-8	>	CMUXVR:PLLIN	PLL counter output to clock mux
EXTCLK			>	GPIO:PLLDIN	TODO
ZDB_IN			<	GPIO:COMBOUT	Zero-delay buffer pin to pll

2.3.4 CBUF

Name	Instance	Type	Values	Default	Documentation
EFB_MUX		Ram	0-1	0	TODO
EFB_MUX_EN		Bool	t/f	f	TODO
EXTCLKOUT_MUX_EN		Bool	t/f	f	TODO
FBIN_MUX	0-1	Ram	0-1	0	TODO
MUX0	0-1	Ram	0-1	0	TODO
MUX0_EN	0-1	Bool	t/f	f	TODO
MUX1	0-1	Ram	0-1	0	TODO
MUX1_EN	0-1	Bool	t/f	f	TODO
MUX2	0-1	Ram	0-1	0	TODO
MUX2_EN	0-1	Bool	t/f	f	TODO
MUX3	0-1	Ram	0-1	0	TODO
MUX3_EN	0-1	Bool	t/f	f	TODO
VCOPH_MUX	0-1	Ram	0-1	0	TODO
VCOPH_MUX_EN	0-1	Bool	t/f	f	TODO

2.3.5 CMUXCR

The three or four Corner CMUX drives 3 horizontal RCLK grids and 3 vertical each.

Name	Instance	Type	Values	Default	Documentation
CLKPIN_INPUT_SELECT_0	0-5	Mux	• pin0 • pin2	pin0	Raising-edge clock input selector for mux input 0
CLKPIN_INPUT_SELECT_1	0-5	Mux	• pin1 • pin3	pin1	Raising-edge clock input selector for mux input 1
ENABLE_REGISTER_MODE	0-5	Mux	• enout • reg1_enout • reg2_enout • vcc	vcc	Enable line buffering mode
ENABLE_REGISTER_POWER_UP	0-5	Num	• 0-1	1	Value of the enable ff outputs at reset time
INPUT_SELECT	0-5	Ram	0-f	f	Clock mux main input selector
NCLKPIN_INPUT_SELECT_0	0-5	Mux	• npin0 • npin2	npin0	Falling-edge clock input selector for mux input 4
NCLKPIN_INPUT_SELECT_1	0-5	Mux	• npin1 • npin3	npin1	Falling-edge clock input selector for mux input 5
PLL_FEEDBACK_ENABLE_0		Mux	• vcc • pll_mcnt0	vcc	TODO
PLL_FEEDBACK_ENABLE_1		Mux	• vcc • pll_mcnt0	vcc	TODO
TOP_PRE_INPUT_SELECT_0		Ram	00-1f	1f	TODO
TOP_PRE_INPUT_SELECT_1		Ram	00-1f	1f	TODO
TOP_PRE_INPUT_SELECT_2		Ram	00-1f	1f	TODO
TOP_PRE_INPUT_SELECT_3		Ram	00-1f	1f	TODO

Port Name	Instance	Port bits	Route node type	Documentation
CLKFBOUT		0-1	RCLKFB	TODO
CLKIN		0-3	DCMUX	Routing grid clock inputs
CLKOUT	0-5		RCLK	Clock mux clock grid driver
ENABLE	0-5		GOUT	Clock enable

Port Name	Instance	Port bits	Dir	Remote port	Documentation
CLKPIN		0-3	<	GPIO:COMBOUT	Raising-edge clock pin to clock mux
NCLKPIN		0-3	<	GPIO:COMBOUT	Falling-edge clock pin to clock mux
PLLIN		0-17	<	FPLL:CLKOUT	PLL counter output to clock mux

2.3.6 CMUXHG

The two Global Horizontal CMUX drive four GCLK grids each. The mux provides selection between positive and negative clock pins, pll counter outputs, HPS clocks and HSSI clocks (TODO). There's also four DCMUX inputs bringing clocks from the clock or the data network. The enable management circuit allows to sync on the inverted output clock through one or two FFs. The burst block is undocumented, but probably keeps enable up for a specific number of clocks upon recieving an input enable edge. There's a system to switch dynamically between 4 clock sources (TODO). There's also a possible selection between feedback signals to send to PLLs.

The circuit is present in 4 instances, each driving a different GCLK betwork. The connections between the CLKIN (DCMUX) inputs and the selection mux depends on the instance:

Inst. - CLKIN	0	1	2	3
0	27	33		
1	27	33		
2			27	33
3			27	33

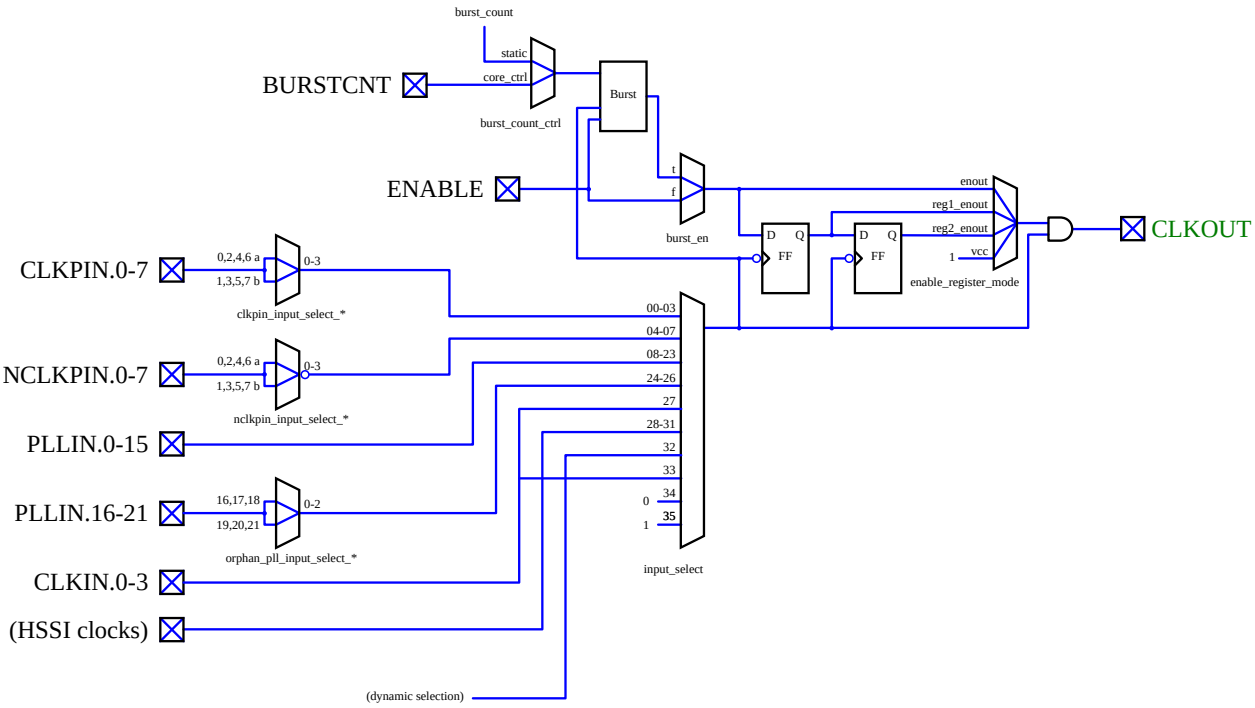

Fig. 4: Global horizontal cmux..

Name	Instance	Type	Values	Default	Documentation
BURST_COUNT	0-3	Ram	0-7	0	Optional fixed burst count
BURST_COUNT_CTRL	0-3	Mux	• static • core_ctrl	static	Selection of the burst count between fixed and coming from the routing network
BURST_EN	0-3	Bool	t/f	f	Whether to use the burst system
CLKPIN_INPUT_SELECT_0	0-3	Mux	• pina • pinb	pina	Raising-edge clock input selector for mux input 0
CLKPIN_INPUT_SELECT_1	0-3	Mux	• pina • pinb	pina	Raising-edge clock input selector for mux input 1
CLKPIN_INPUT_SELECT_2	0-3	Mux	• pina • pinb	pina	Raising-edge clock input selector for mux input 2
CLKPIN_INPUT_SELECT_3	0-3	Mux	• pina • pinb	pina	Raising-edge clock input selector for mux input 3
CLK_SELECT_A	0-3	Ram	0-3	0	TODO
CLK_SELECT_B	0-3	Ram	0-3	0	TODO
CLK_SELECT_C	0-3	Ram	0-3	0	TODO
CLK_SELECT_D	0-3	Ram	0-3	0	TODO
ENABLE_REGISTER_MODE	0-3	Mux	• enout • reg1_enout • reg2_enout • vcc	vcc	Enable line buffering mode
ENABLE_REGISTER_POWER_UP	0-3	Num	• 0-1	1	Value of the enable ff outputs at reset time
INPUT_SELECT	0-3	Ram	00-3f	23	Clock mux main input selector
NCLKPIN_INPUT_SELECT_0	0-3	Mux	• npina • npinb	npina	Falling-edge clock input selector for mux input 4
NCLKPIN_INPUT_SELECT_1	0-3	Mux	• npina • npinb	npina	Falling-edge clock input selector for mux input 5

continues on next page

Table 7 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
NCLKPIN_INPUT0_SELECT_2	0-3	Mux	• npina • npinb	npina	Falling-edge clock input selector for mux input 6
NCLKPIN_INPUT0_SELECT_3	0-3	Mux	• npina • npinb	npina	Falling-edge clock input selector for mux input 7
ORPHAN_PLL_INPUT_SELECT_0	0-3	Mux	• orphan_pll0 • orphan_pll3	orphan_pll0	Select between two pll outputs before the main mux input 24
ORPHAN_PLL_INPUT_SELECT_1	0-3	Mux	• orphan_pll1 • orphan_pll4	orphan_pll1	Select between two pll outputs before the main mux input 25
ORPHAN_PLL_INPUT_SELECT_2	0-3	Mux	• orphan_pll2 • orphan_pll5	orphan_pll2	Select between two pll outputs before the main mux input 26 (unused in practice, inputs not connected)
TESTSYN_ENOUT_SELECT	0-3	Mux	• core_en • pre_synenb	core_en	TODO
DYNAMIC_CLK_SELECT		Bool	t/f	f	TODO
FEEDBACK_DRIVER_SELECT_0		Mux	• in0_vcc • in1 • in2_vcc • in3_vcc • in4_vcc • in5 • in6 • in7	in0_vcc	TODO

continues on next page

Table 7 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
FEEDBACK_DRIVER_SELECT_1		Mux	• in0_vcc • in1 • in2_vcc • in3_vcc • in4_vcc • in5 • in6 • in7	in0_vcc	TODO
ORPHAN_PLL_FEEDBACK_OUT_SELECT_0		Ram	0-1	0	TODO
ORPHAN_PLL_FEEDBACK_OUT_SELECT_1		Ram	0-1	0	TODO
PLL_FEEDBACK_ENABLE_0		Mux	• vcc • pll_mcnt0	vcc	TODO
PLL_FEEDBACK_ENABLE_1		Mux	• vcc • pll_mcnt0	vcc	TODO
PLL_FEEDBACK_OUT_SELECT_0		Ram	0-1	0	TODO
PLL_FEEDBACK_OUT_SELECT_1		Ram	0-1	0	TODO

Port Name	Instance	Port bits	Route node type	Documentation
BURSTCNT		0-2	GOUT	Burst block counter value
CLKFBOUT		0-1	GCLKFB	TODO
CLKIN		0-3	DCMUX	Routing grid clock inputs
CLKOUT	0-3		GCLK	Clock mux clock grid driver
ENABLE	0-3		GOUT	Clock enable
SWITCHCLK	0-3		GIN	Dynamically selected clock output
SWITCHIN	0-3	0-1	GOUT	Dynamic clock selection input
SYN_EN	0-3		GIN	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
CLKPIN		0-7	<	GPIO:COMBOUT	Raising-edge clock pin to clock mux
NCLKPIN		0-7	<	GPIO:COMBOUT	Falling-edge clock pin to clock mux
PLLIN		0-17, 19	<	FPLL:CLKOUT	PLL counter output to clock mux
PLLIN		0-3	<	HPS_CLOCKS:CLKOUT	HPS clock output to clock mux

2.3.7 CMUXVG

The two Global Vertical CMUX drive four GCLK grids each.

Name	Instance	Type	Values	Default	Documentation
BURST_COUNT	0-3	Ram	0-7	0	Optional fixed burst count
BURST_COUNT_CTRL	0-3	Mux	• static • core_ctrl	static	Selection of the burst count between fixed and coming from the routing network
BURST_EN	0-3	Bool	t/f	f	Whether to use the burst system
CLK_SELECT_A	0-3	Ram	0-3	0	TODO
CLK_SELECT_B	0-3	Ram	0-3	0	TODO
CLK_SELECT_C	0-3	Ram	0-3	0	TODO
CLK_SELECT_D	0-3	Ram	0-3	0	TODO
ENABLE_REGISTER_MODE	0-3	Mux	• enout • reg1_enout • reg2_enout • vcc	vcc	Enable line buffering mode
ENABLE_REGISTER_POWER_UP	0-3	Num	• 0-1	1	Value of the enable ff outputs at reset time
INPUT_SELECT	0-3	Ram	00-1f	1b	Clock mux main input selector
TESTSYN_ENOUT_SELECT	0-3	Mux	• core_en • pre_synenb	pre_synenb	TODO
DYNAMIC_CLK_SELECT		Bool	t/f	f	TODO
PLL_FEEDBACK_ENABLE_0		Mux	• vcc • pll_mcnt0	vcc	TODO
PLL_FEEDBACK_ENABLE_1		Mux	• vcc • pll_mcnt0	vcc	TODO
PLL_FEEDBACK_ENABLE_1		Mux	• vcc • pll_mcnt0	vcc	TODO
PLL_FEEDBACK_ENABLE_2		Mux	• vcc • pll_mcnt0	vcc	TODO
PLL_FEEDBACK_ENABLE_3		Mux	• vcc • pll_mcnt0	vcc	TODO

Port Name	Instance	Port bits	Route node type	Documentation
BURSTCNT		0-2	GOUT	TODO
CLKFBOUT		0-2	GCLKFB	TODO
CLKIN		0-3	DCMUX	Routing grid clock inputs
CLKOUT	0-3		GCLK	Clock mux clock grid driver
ENABLE	0-3		GOUT	Clock enable
SWITCHCLK	0-3		GIN	TODO
SWITCHIN	0-3	0-1	GOUT	Dynamic clock selection input
SYN_EN	0-3		GIN	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
CLKPIN		0-3	<	GPIO:COMBOUT	Raising-edge clock pin to clock mux
NCLKPIN		0-3	<	GPIO:COMBOUT	Falling-edge clock pin to clock mux
PLLIN		0-11	<	FPLL:CLKOUT	PLL counter output to clock mux
PLLIN		4-7	<	HPS_CLOCKS:CLKOUT	HPS clock output to clock mux

2.3.8 CMUXHR

The two Regional Horizontal CMUX drive 12 vertical RCLK grids each, half on each side. Six are lost when touching the HPS.

Name	Instance	Type	Values	Default	Documentation
CLKPIN_INPUT_SELECT	0-1	Mux	• pina • pinb	pina	TODO
ENABLE_REGISTER_MODE	0-11	Mux	• enout • reg1_enout • reg2_enout • vcc	vcc	Enable line buffering mode
ENABLE_REGISTER_POWER_UP	0-11	Num	• 0-1	1	Value of the enable ff outputs at reset time
INPUT_SELECT	0-11	Ram	00-1f	13	Clock mux main input selector
NCLKPIN_INPUT_SELECT	0-1	Mux	• npina • npinb	npina	TODO
BOT_PRE_INPUT_SELECT_0		Ram	00-1f	1f	TODO
BOT_PRE_INPUT_SELECT_1		Ram	00-1f	1f	TODO
BOT_PRE_INPUT_SELECT_2		Ram	00-1f	1f	TODO
BOT_PRE_INPUT_SELECT_3		Ram	00-1f	1f	TODO
FEEDBACK_DRIVER_SELECT_0		Mux	• vcc • orphan_pll_mcnto0 • orphan_pll_mcnto1 • orphan_pll_mcnto2	vcc	TODO
FEEDBACK_DRIVER_SELECT_1		Mux	• vcc • orphan_pll_mcnto0 • orphan_pll_mcnto1 • orphan_pll_mcnto2	vcc	TODO
PLL_FEEDBACK_ENABLE_0		Mux	• vcc • pll_mcnt0	vcc	TODO
PLL_FEEDBACK_ENABLE_1		Mux	• vcc • pll_mcnt0	vcc	TODO
PRE_INPUT_SELECT_0		Ram	00-1f	1f	TODO
PRE_INPUT_SELECT_1		Ram	00-1f	1f	TODO
PRE_INPUT_SELECT_2		Ram	00-1f	1f	TODO
PRE_INPUT_SELECT_3		Ram	00-1f	1f	TODO
TOP_PRE_INPUT_SELECT_0		Ram	00-1f	1f	TODO
TOP_PRE_INPUT_SELECT_1		Ram	00-1f	1f	TODO
TOP_PRE_INPUT_SELECT_2		Ram	00-1f	1f	TODO
TOP_PRE_INPUT_SELECT_3		Ram	00-1f	1f	TODO

Port Name	Instance	Port bits	Route node type	Documentation
CLKFBIN		0-3	DCMUX	TODO
CLKFBOUT		0-1	RCLKFB	TODO
CLKIN		0-3	DCMUX	Routing grid clock inputs
CLKOUT	0-11		RCLK	Clock mux clock grid driver
ENABLE	0-11		GOUT	Clock enable

Port Name	Instance	Port bits	Dir	Remote port	Documentation
CLKPIN		0-7	<	GPIO:COMBOUT	Raising-edge clock pin to clock mux
NCLKPIN		0-7	<	GPIO:COMBOUT	Falling-edge clock pin to clock mux
PLLIN		0-25	<	FPLL:CLKOUT	PLL counter output to clock mux
PLLIN		0-6, 20-21	<	HPS_CLOCKS:CLKOUT	HPS clock output to clock mux

2.3.9 CMUXVR

The two Global Vertical CMUX drive 20 horizontal RCLK grids each half on each side. Ten are lost when touching the HPS.

Name	Instance	Type	Values	Default	Documentation
ENABLE_REGISTER_MODE	0-19	Mux	• enout • reg1_enout • reg2_enout • vcc	vcc	Enable line buffering mode
ENABLE_REGISTER_POWER_UP	0-19	Num	• 0-1	1	Value of the enable ff outputs at reset time
INPUT_SELECT	0-19	Ram	0-f	b	Clock mux main input selector
PLL_FEEDBACK_ENABLE_0		Mux	• vcc • pll_mcnt0	vcc	TODO

Port Name	Instance	Port bits	Route node type	Documentation
CLKIN		0-3	DCMUX	Routing grid clock inputs
CLKOUT	0-19		RCLK	Clock mux clock grid driver
ENABLE	0-19		GOUT	Clock enable

Port Name	Instance	Port bits	Dir	Remote port	Documentation
CLKPIN		0-3	<	GPIO:COMBOUT	Raising-edge clock pin to clock mux
NCLKPIN		0-3	<	GPIO:COMBOUT	Falling-edge clock pin to clock mux
PLLIN		0-8, 18-24	<	FPLL:CLKOUT	PLL counter output to clock mux
PLLIN		0-8	<	HPS_CLOCKS:CLKOUT	HPS clock output to clock mux

2.3.10 CMUXP

The CMUXP drive two PCLK each.

Port Name	Instance	Port bits	Route node type	Documentation
CLKIN		0	DCMUX	Routing grid clock input
CLKOUT		0-1	PCLK	Clock mux clock grid driver

2.3.11 CTRL

The Control block gives access to a number of anciliary functions of the FPGA.

TODO: everything, GOUT/GIN/DCMUX mapping is done

Port Name	Instance	Port bits	Route node type	Documentation
CAPTNUPDT_RU			GOUT	TODO
CLKDRUSER			GIN	TODO
CLK_OUT			GIN	Internal oscillator clock output
CLK_OUT1			GIN	Internal oscillator clock 1 output
CLOCK_CHIPID			DCMUX	TODO
CLOCK_CRC			DCMUX	TODO
CLOCK_OPREG			DCMUX	TODO
CLOCK_PR			DCMUX	TODO
CLOCK_RU			DCMUX	TODO
CLOCK_SPI			DCMUX	TODO
CONFIG			GOUT	TODO
CORECTL_JTAG			GOUT	TODO
CORECTL_PR			GOUT	TODO
CRCERROR			GIN	TODO
DATA		0-15	GOUT	TODO
DATA0IN			GIN	TODO
DATA0OE			GOUT	TODO
DATA0OUT			GOUT	TODO
DATA1IN			GIN	TODO
DATA1OE			GOUT	TODO
DATA1OUT			GOUT	TODO
DATA2IN			GIN	TODO
DATA2OE			GOUT	TODO
DATA2OUT			GOUT	TODO
DATA3IN			GIN	TODO
DATA3OE			GOUT	TODO
DATA3OUT			GOUT	TODO
DFT_IN		0-5	GOUT	TODO
DFT_OUT		0-24	GIN	TODO
DONE			GIN	TODO
END_OF_ED_FULLCHIP			GIN	TODO
EXTERNALREQUEST			GIN	TODO
NCE_OUT			GIN	TODO
NTDOPINENA			GOUT	TODO
OERROR			GIN	TODO

continues on next page

Table 8 – continued from previous page

Port Name	Instance	Port bits	Route node type	Documentation
OSC_ENA			GOUT	Internal oscillator enable
OUTPUT_ENABLE			GOUT	TODO
PRREQUEST			GOUT	TODO
READY			GIN	TODO
REGIN			GOUT	TODO
REG_OUT_CHIPID			GIN	TODO
REG_OUT_CRC			GIN	TODO
REG_OUT_OPREG			GIN	TODO
REG_OUT_RU			GIN	TODO
RSTTIMER			GOUT	TODO
RUNIDLEUSER			GIN	TODO
SCE_IN			GOUT	TODO
SHIFTNLD_CHIPID			GOUT	TODO
SHIFTNLD_CRC			GOUT	TODO
SHIFTNLD_OPREG			GOUT	TODO
SHIFTNLD_RU			GOUT	TODO
SHIFTUSER			GIN	TODO
TCKCORE			DCMUX	TODO
TCKUTAP			GIN	TODO
TDICORE			GOUT	TODO
TDIUTAP			GIN	TODO
TDOCORE			GIN	TODO
TDOUTAP			GOUT	TODO
TMSCORE			GOUT	TODO
TMSUTAP			GIN	TODO
UPDATEUSER			GIN	TODO
USR1USER			GIN	TODO

2.3.12 HSSI

The High speed serial interface blocks control the serializing/deserializing capabilities of the FPGA.

TODO: everything

Name	Instance	Type	Values	Default	Documentation
PCS8G_AGGREGATE_DSKW_CONTROL		TODO	• write • read	write	TODO
PCS8G_AGGREGATE_DSKW_SM_OPERATION		TODO	• xaui_sm • srio_sm	xaui_sm	TODO
PCS8G_AGGREGATE_PCS_DW_BONDING		TODO	• disable	disable	TODO
PCS8G_AGGREGATE_POWERDOWN_EN		Bool	t/f	f	TODO
PCS8G_AGGREGATE_REFCLK_DIG_SEL_EN		Bool	t/f	f	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCS8G_AGGREGATE_XAUI_SM		Mux	• xaui_legacy_sm • xaui_sm • disable	xaui_legacy_sm	TODO
COM_PCS_PLD_IP_HIP_EN	0-2	Bool	t/f	f	TODO
COM_PCS_PLD_IP_HRDRSTCTRL_CFG_USR_EN	0-2	Bool	t/f	f	TODO
COM_PCS_PLD_IP_HRDRSTCTRL_CFG_EN	0-2	Bool	t/f	f	TODO
COM_PCS_PLD_IP_TESTBUF_SEL	0-2	Mux	• pcs8g • pma_if	pcs8g	TODO
COM_PCS_PLD_IP_USRMODE_SEL_HRST	0-2	Mux	• usermode • last_frz	usermode	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC0	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC1	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC10	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC11	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC2	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC3	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC4	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC5	0-2	Mux	• pld • b_hip	pld	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
COM_PCS_PLD_PLD_SIDE_RES_SRC6	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC7	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC8	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_PLD_SIDE_RES_SRC9	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PLD_SIDE_DATA_SRC	0-2	Mux	• pld • b_hip	pld	TODO
COM_PCS_PMA_IF_AUTO_SPEED_EN	0-2	Bool	t/f	f	TODO
COM_PCS_PMA_IF_BLOCK_SEL	0-2	Bool	t/f	f	TODO
COM_PCS_PMA_IF_FORCE_FREQDET	0-2	Mux	• off • force0 • force1	off	TODO
COM_PCS_PMA_IF_G3PCS	0-2	Bool	t/f	f	TODO
COM_PCS_PMA_IF_PMA_IF_DFT_EN	0-2	Bool	t/f	f	TODO
COM_PCS_PMA_IF_PMA_IF_DFT_VAL	0-2	Raw	0-1	0	TODO
COM_PCS_PMA_IF_PM_GEN1_2_CNT	0-2	Mux	• cnt_32k • cnt_64k	cnt_32k	TODO
COM_PCS_PMA_IF_PPMSEL	0-2	Mux	• default • ppm_100 • ppm_125 • ppm_62_5 • ppm_200 • ppm_300 • ppm_250 • ppm_500 • ppm_1000 • ppm_other	default	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
COM_PCS_PMA_PHY_CNT_RST	1-2	Bool	t/f	f	TODO
COM_PCS_PMA_PHY_EARLY_DEASSERT	1-2	Bool	t/f	f	TODO
COM_PCS_PMA_PHY_POST_EIDLE_DLY	1-2	Num	• 200 • 400	200	TODO
PCS8G_BASE_ADDR	0-2	Ram	000-7ff		TODO
PCS8G_DEFAULT_BROADCAST_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_SYMBOL_BOX	0-2	Ram	000-fff	0	TODO
PCS8G_DIGI_RX_8B10B_DECODER	0-2	Mux	• off • sgx • ibm	off	TODO
PCS8G_DIGI_RX_8B10B_DECODER_MODE	0-2	OUTPUT_SEL	• data_8b10b • data_xaui_sm	data_8b10b	TODO
PCS8G_DIGI_RX_AGC_BLOCK_SEL	0-2	Mux	• same • other	same	TODO
PCS8G_DIGI_RX_AUTO_ERROR_REPLACE_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_AUTO_SPEED_NEG	0-2	Ram	40 bits	0	TODO
PCS8G_DIGI_RX_BDS_DEC_CLOCK_GATING_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_BIST_CLOCK_GATING_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_BIST_CLR_FLAG_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_BIST_VER	0-2	Mux	• disable • incremental • cjpat • crpat	disable	TODO
PCS8G_DIGI_RX_BIT_REVERSAL_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_BYTEORDER_CLOCK_GATING_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_BYTE_DESERIALIZER	0-2	Mux	• disable • bds_by_2 • bds_by_2_det	disable	TODO
PCS8G_DIGI_RX_BYTE_ORDER	0-2	Ram	23 bits	0	TODO
PCS8G_DIGI_RX_CDR_CTRL	0-2	Ram	30 bits	0	TODO
PCS8G_DIGI_RX_FIFO_RST_PLD_CTRL_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_PRBS_PATTERN	0-2	Ram	00-ff	0	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCS8G_DIGI_RX_CLK1	C1	Mux	• clk1 • tx_pma • agg • agg_top_or_bottom	clk1	TODO
PCS8G_DIGI_RX_CLK2	C1	Mux	• rcvd_clk • tx_pma • ref-clk_dig2	rcvd_clk	TODO
PCS8G_DIGI_RX_CLK_FREE_RUNNING_EN	C1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_DESKEW	D1	Mux	• disable • xaui • srio_v2p1	disable	TODO
PCS8G_DIGI_RX_DESKEW_PROG_PAT_ONLY_EN	D1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_DESKEW_RDCLK_GATING_EN	D1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_DW_DESKEW_WRCLK_GATING_EN	D1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_DW_PC_WRCLK_GATING_EN	D1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_DW_RM_RDCLK_GATING_EN	D1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_DW_RM_WRCLK_GATING_EN	D1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_DW_WA_CLOCK_GATING_EN	D1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_EIDLE_CLOCK_GATING_EN	E1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_EIDLE_EIOS_EN	E1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_EIDLE_ENTRY_IB_EN	E1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_EIDLE_ENTRY_SB_EN	E1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_ERR_FLAGS_SEL	E1	Mux	• flags_8b10b • flags_wa	flags_8b10b	TODO
PCS8G_DIGI_RX_INVALID_CODE_FLAG_ONLY_EN	F1	Bool	t/f	f	TODO
PCS8G_DIGI_RX_PAD_EDB_ERROR	F1	MUX_REPLACE	• edb • pad • edb_dynamic	edb	TODO
PCS8G_DIGI_RX_PARALLEL_LOOPBACK_EN	F2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_PCFIFO_RST_PLD_CTRL_EN	F2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_PCS_BYPASS_EN	F2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_PCS_URST_EN	F2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_PC_RDCLK_GATING_EN	F2	Bool	t/f	f	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCS8G_DIGI_RX_0_PHASE_COMPENSATION_FIFO	PCS8G_DIGI_RX_0_PHASE_COMPENSATION_FIFO	Mux	• normal_latency • pid_ctrl_normal_latency • low_latency • pid_ctrl_low_latency • register_fifo	normal_latency	TODO
PCS8G_DIGI_RX_0_PIPE_IF_EN	PCS8G_DIGI_RX_0_PIPE_IF_EN	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_PLANE_BONDING_0_COMP_EN	PCS8G_DIGI_RX_0_PLANE_BONDING_0_COMP_EN	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_PLANE_BONDING_0_MASTER	PCS8G_DIGI_RX_0_PLANE_BONDING_0_MASTER	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_PMA_DW	PCS8G_DIGI_RX_0_PMA_DW	Num	• 8 • 10 • 16 • 20	8	TODO
PCS8G_DIGI_RX_0_POLARITY_INVERSION_EN	PCS8G_DIGI_RX_0_POLARITY_INVERSION_EN	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_POLINV_8B10B_DEC_EN	PCS8G_DIGI_RX_0_POLINV_8B10B_DEC_EN	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_PRBS_CLOCK_GATING_EN	PCS8G_DIGI_RX_0_PRBS_CLOCK_GATING_EN	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_PRBS_CLR_FLAG_EN	PCS8G_DIGI_RX_0_PRBS_CLR_FLAG_EN	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_PRBS_VER	PCS8G_DIGI_RX_0_PRBS_VER	Mux	• disable • prbs_7_dw_8_10 • prbs_23_dw_hf_sw • prbs_7_sw_hf_dw_lf_sw • prbs_lf_dw_mf_sw • prbs_23_sw_mf_dw • prbs_15 • prbs_31	disable	TODO
PCS8G_DIGI_RX_0_RATHER_MATCH	PCS8G_DIGI_RX_0_RATHER_MATCH	Ram	68 bits	0	TODO
PCS8G_DIGI_RX_0_RCVD_CLK	PCS8G_DIGI_RX_0_RCVD_CLK	Mux	• rcvd_clk • tx_pma	rcvd_clk	TODO
PCS8G_DIGI_RX_0_RD_CLK	PCS8G_DIGI_RX_0_RD_CLK	Mux	• rx_clk • pld	rx_clk	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCS8G_DIGI_RX_0_REFCLK_SEL_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_RE_BO_ON_WA_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_RUNLENGTH_CHECK	0-2	Ram	00-7f	0	TODO
PCS8G_DIGI_RX_0_SW_DESKEW_WRCLK_GATING_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_SW_PC_WRCLK_GATING_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_SW_RM_RDCLK_GATING_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_SW_RM_WRCLK_GATING_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_SYMBOL_SWAP_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_TEST_BUS_SEL	0-2	Mux	• prbs_bist • tx • tx_ctrl_plane • wa • deskew • rm • rx_ctrl • pcie_ctrl • rx_ctrl_plane • agg	prbs_bist	TODO
PCS8G_DIGI_RX_0_VALID_MASK_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_WA_BOUNDARY_LOCK	0-2	Mux	• auto_align_pld_ctrl • sync_sm • deterministic_latency • bit_slip	auto_align_pld_ctrl	TODO
PCS8G_DIGI_RX_0_WA_CLK_SLIP_SPACING	0-2	Ram	000-3ff	0	TODO
PCS8G_DIGI_RX_0_WA_CLOCK_GATING_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_WA_DET_LATENCY_SYNC_STATUS	0-2	Mux	• delayed • immediate	delayed	TODO
PCS8G_DIGI_RX_0_WA_DISP_ERR_FLAG_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_WA_KCHAR_EN	0-2	Bool	t/f	f	TODO
PCS8G_DIGI_RX_0_WA_PD	0-2	Ram	43 bits	0	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCS8G_DIGI_RX_0_V2A_PLD_CONTROL_L	0	Mux	• level_sensitive • pid_ctrl_sw • rising_edge_sensitive	level_sensitive	TODO
PCS8G_DIGI_RX_0_V2A_SYNC_SM_CTRL	0	Ctrl	38 bits	0	TODO
PCS8G_DIGI_RX_0_V2R_CLK	0	Mux	• rx_clk2 • tx_fifo_rd_clk	rx_clk2	TODO
PCS8G_DIGI_TX_0_8B10B_DISP_CTRL	0	Mux	• off • on_ib • on	off	TODO
PCS8G_DIGI_TX_0_8B10B_ENCODER	0	Mux	• off • ibm • sgx	off	TODO
PCS8G_DIGI_TX_0_8B10B_ENCODER_IN	0	MUX	• xaui_sm • normal_data_path • gige_idle_conversion	xaui_sm	TODO
PCS8G_DIGI_TX_0_ASC_BLOCK_SEL	0	Mux	• same • other	same	TODO
PCS8G_DIGI_TX_0_BIST_CLOCK_GATE_EN	0	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_BIST_GEN	0	Mux	• disable • incremental • cjpat • crpat	disable	TODO
PCS8G_DIGI_TX_0_BITSLIP_EN	0	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_BIT_REVERSAL_EN	0	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_BS_CLOCK_GATE_EN	0	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_BYPASS_PIPELINE_REG_EN	0	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_BYTE_SERIALIZER_EN	0	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_C2_DISPARITY_EN	0	Bool	t/f	f	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCS8G_DIGI_TX_0_CID_PATTERN	0-23	Ram	000-1ff	0	TODO
PCS8G_DIGI_TX_0_DYNAMIC_CLOCK_SWITCH_EN	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_FORD_CLOCK_GATE_EN	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_FOWR_CLOCK_GATE_EN	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_FORCE_ECHAR_EN	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_FORCE_KCHAR_EN	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_G2_FREQUENCY_SCALING	0-23	Mux	• off • on	off	TODO
PCS8G_DIGI_TX_0_LOOPBACK	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_PCFIFO_URST_EN	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_PCS_BYPASS_EN	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_PHASE_COMPENSATION_FIFO	0-23	Mux	• normal_latency • pid_ctrl_normal_latency • low_latency • pid_ctrl_low_latency • register_fifo	normal_latency	TODO
PCS8G_DIGI_TX_0_PLFIFO_REFCLK_MUX_SEL	0-23	Mux	• refclk • tx_pma	refclk	TODO
PCS8G_DIGI_TX_0_PLFIFO_WRITE_CLK_SEL	0-23	Mux	• pld • tx_clk	pld	TODO
PCS8G_DIGI_TX_0_PLANE_BONDING_COMP_EN	0-23	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0_PLANE_BONDING_CONSUMPTION	0-23	Mux	• individual • bundled_master • slave_above • slave_below	individual	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCS8G_DIGI_TX_0-2	PLANE_BONDING_CONSUMPTION	Mux	• individual • bundled_master • slave_above • slave_below	individual	TODO
PCS8G_DIGI_TX_0-2	PLANE_BONDING_MASTER	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0-2	PMA_DW	Num	• 8 • 10 • 16 • 20	8	TODO
PCS8G_DIGI_TX_0-2	POLARITY_INVERSION_EN	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0-2	PRBS_CLOCK_GATE_EN	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0-2	PRBS_GEN	Mux	• disable • prbs_7_dw_8_10 • prbs_23_dw_hf_sw • prbs_7_sw_hf_dw_lf_sw • prbs_lf_dw_mf_sw • prbs_23_sw_mf_dw • prbs_15 • prbs_31	disable	TODO
PCS8G_DIGI_TX_0-2	SYMBOL_SWAP_EN	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0-2	TXCLK_FREERUN_EN	Bool	t/f	f	TODO
PCS8G_DIGI_TX_0-2	TXPCS_URST_EN	Bool	t/f	f	TODO
PCS8G_MDIO_DIS_0-2	CVP_EN	Bool	t/f	f	TODO
PCS8G_MDIO_DIS_0-2	FORCE_EN	Bool	t/f	f	TODO
PCS8G_PIPE_INTF_0-2	TOP_DESERIAL_EN	Bool	t/f	f	TODO
PCS8G_PIPE_INTF_0-2	TOP_ERROR_REPLACE_PAD	Mux	• edb • pad	edb	TODO
PCS8G_PIPE_INTF_0-2	TOP_IND_ERROR_REPORTING	Bool	t/f	f	TODO
PCS8G_PIPE_INTF_0-2	TOP_PHYSTATUS_RST_TOGGLE	Bool	t/f	f	TODO
PCS8G_PIPE_INTF_0-2	TOP_RPRE_EMPH_SETTINGS	Param	30 bits	0	TODO
PCS8G_PIPE_INTF_0-2	TOP_RVOD_SEL_SETTINGS	Param	30 bits	0	TODO
PCS8G_PIPE_INTF_0-2	TOP_RXDETECT_BYPASS_EN	Bool	t/f	f	TODO
PCS8G_PIPE_INTF_0-2	TOP_RX_PIPE_EN	Bool	t/f	f	TODO

continues on next page

Table 9 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCS8G_PIPE_INTF_TOP_TXSWINGBEN	0-2	Bool	t/f	f	TODO
PCS8G_PIPE_INTF_TOP_TX_PIPE_EN	0-2	Bool	t/f	f	TODO
PCS8G_POWER_ISOLATION_EN	0-2	Bool	t/f	f	TODO
PCS9G_PIPE_INTF_TOP_ELECIDLE_DELAY	0-2	Ram	0-7	0	TODO
PCS9G_PIPE_INTF_TOP_PHY_STATUS_DELAY	0-2	Ram	0-7	0	TODO
PLD_PCS_DEFAULT_BROADCAST_EN	0-2	Bool	t/f	f	TODO
PLD_PCS_IF_BASE_ADDR	0-2	Ram	000-7ff		TODO
PLD_PCS_MDIO_DIS_CVP_EN	0-2	Bool	t/f	f	TODO
PLD_PCS_MDIO_DIS_FORCE_EN	0-2	Bool	t/f	f	TODO
PLD_PCS_POWER_ISOLATION_EN	0-2	Bool	t/f	f	TODO
PMA_PCS_DEFAULT_BROADCAST_EN	0-2	Bool	t/f	f	TODO
PMA_PCS_IF_BASE_ADDR	0-2	Ram	000-7ff		TODO
PMA_PCS_MDIO_DIS_CVP_EN	0-2	Bool	t/f	f	TODO
PMA_PCS_MDIO_DIS_FORCE_EN	0-2	Bool	t/f	f	TODO
PMA_PCS_POWER_ISOLATION_EN	0-2	Bool	t/f	f	TODO
RX_PCS_PLD_IF_PCS_SIDE_BLOCK_SEL	0-2	Mux	• default • pcs8g	default	TODO
RX_PCS_PLD_SIDE_DATA_SRC	0-2	Mux	• pld • b_hip	pld	TODO
RX_PCS_PMA_IF	0-2	Mux	• default • pcs8g	default	TODO
RX_PCS_PMA_IF_CLKSLIP_SEL	0-2	Mux	• pld • slip_pcs8g	pld	TODO
TX_PCS_PLD_SIDE_DATA_SRC	0-2	Mux	• pld • b_hip	pld	TODO
TX_PCS_PMA_IF_BLOCK_SEL	0-2	Mux	• default • pcs8g	default	TODO

2.3.13 HIP

The PCIe Hard-IP blocks control the PCIe interfaces of the FPGA.

TODO: everything

Name	Instance	Type	Values	Default	Documentation
BIST_MEMORY_SETTINGS_DATA		Ram	75 bits	0	TODO
BRIDGE_66MHZCAP		Bool	t/f	f	TODO
BR_RCB		Mux	• ro • rw	ro	TODO
BYPASS_CDC		Bool	t/f	f	TODO
BYPASS_CLK_SWITCH		Bool	t/f	f	TODO
BYPASS_TL		Bool	t/f	f	TODO
CDC_CLK_RELATION		Mux	• plesiochronous • mesochronous	plesiochronous	TODO
CDC_DUMMY_INSERT_LIMIT_DATA		Ram	0-f	0	TODO
CORE_CLK_DISABLE_CLK_SWITCH		Mux	• core_clk_out • pld_clk	core_clk_out	TODO
CORE_CLK_DIVIDER		Num	• 1-2 • 4 • 8 • 16	4	TODO
CORE_CLK_OUT_SEL		Mux	• div_1 • div_2	div_1	TODO
CORE_CLK_SEL		Mux	• core_clk_out • pld_clk	core_clk_out	TODO
CORE_CLK_SOURCE		Mux	• pll_fixed_clk • core_clk_in • pclk_in	pll_fixed_clk	TODO
CVP_CLK_RESET		Bool	t/f	f	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
CVP_DATA_COMPRESSED		Bool	t/f	f	TODO
CVP_DATA_ENCRYPTED		Bool	t/f	f	TODO
CVP_ISOLATION		Bool	t/f	f	TODO
CVP_MODE_RESET		Bool	t/f	f	TODO
CVP_RATE_SEL		Mux	• full_rate • half_rate	full_rate	TODO
DEVICE_NUMBER_DATA		Ram	00-1f	0	TODO
DEVSELTIM		Mux	• fast_devsel_decoding • medium_devsel_decoding • slow_devsel_decoding	fast_devsel_decoding	TODO
DISABLE_AUTO_CRS		Bool	t/f	f	TODO
DISABLE_CLK_SWITCH		Bool	t/f	f	TODO
DISABLE_LINK_X2_SUPPORT		Bool	t/f	f	TODO
DISABLE_TAG_CHECK		Bool	t/f	f	TODO
EI_DELAY_POWERDOWN_COUNT_DATA		Ram	00-ff	0	TODO
ENABLE_ADAPTER_HALF_RATE_MODE		Bool	t/f	f	TODO
ENABLE_CH01_PCLK_OUT		Mux	• pclk_ch0 • pclk_ch1	pclk_ch0	TODO
ENABLE_CH0_PCLK_OUT		Mux	• pclk_central • pclk_ch01	pclk_central	TODO
ENABLE_RX_BUFFER_CHECKING		Bool	t/f	f	TODO
ENABLE_RX_REORDERING		Bool	t/f	f	TODO
FASTB2BCAP		Bool	t/f	f	TODO
FC_INIT_TIMER_DATA		Ram	000-7ff	0	TODO
FLOW_CONTROL_TIMEOUT_COUNT_DATA		Ram	00-ff	0	TODO
FLOW_CONTROL_UPDATE_COUNT_DATA		Ram	00-1f	0	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
GEN12_LANE_RATE_MODE		Mux	• gen1 • gen1_gen2	gen1	TODO
HARD_RESET_BYPASS		Bool	t/f	f	TODO
IEI_ENABLE_SETTINGS		Mux	• disabled • disable_iei_logic • gen2_infei_gen1_infei • gen2_infei_gen1_infei_sd • gen2_infei_infsd_gen1_infei_sd • gen2_infei_infsd_gen1_infei_infsd	disabled	TODO
JTAG_ID_DATA		Ram	128 bits	0	TODO
L01_ENTRY_LATENCY_DATA		Ram	00-1f	0	TODO
LANE_MASK		Mux	• x8 • x1 • x2 • x4	x8	TODO
LATTIM_RO_DATA		Ram	00-7f	0	TODO
MDIO_CB_OPBIT_ENABLE		Bool	t/f	f	TODO
MEMWRINV		Mux	• ro • rw	ro	TODO
MILLISECOND_CYCLE_COUNT_DATA		Ram	20 bits	0	TODO
MULTI_FUNCTION		Num	• 1-8	1	TODO
NATIONAL_INST_THRU_ENHANCE		Bool	t/f	f	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PCIE_MODE		Mux	• ep_native • ep_legacy • rp • sw_up • sw_dn • bridge • switch_mode • shared_mode	ep_native	TODO
PCIE_SPEC_1P0_COMPLIANCE		Mux	• spec_1p0a • spec_1p1	spec_1p0a	TODO
PCLK_OUT_SEL		Mux	• core_clk_en • pclk_out	core_clk_en	TODO
PIPEX1_DEBUG_SEL		Bool	t/f	f	TODO
PLNIOTRI_GATE		Bool	t/f	f	TODO
PORT_LINK_NUMBER_DATA		Ram	00-ff	0	TODO
REGISTER_PIPE_SIGNALS		Bool	t/f	f	TODO
RETRY_BUFFER_LAST_ACTIVE_ADDRESS_DATA		Ram	00-ff	0	TODO
RETRY_BUFFER_MEMORY_SETTINGS_DATA		Ram	0000-ffff	0	TODO
RSTCTRL_1MS_COUNT_FREF_CLK_VALUE		Ram	20 bits	0	TODO
RSTCTRL_1US_COUNT_FREF_CLK_VALUE		Ram	20 bits	0	TODO
RSTCTRL_ALTPE2_CRST_N_INV		Bool	t/f	f	TODO
RSTCTRL_ALTPE2_RST_N_INV		Bool	t/f	f	TODO
RSTCTRL_ALTPE2_SRST_N_INV		Bool	t/f	f	TODO
RSTCTRL_DEBUG_EN		Bool	t/f	f	TODO
RSTCTRL_FORCE_INACTIVE_RST		Bool	t/f	f	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RSTC-TRL_FREF_CLK_SELECT		Mux	• disabled • ch0_sel • ch1_sel • ch2_sel • ch3_sel • ch4_sel • ch5_sel • ch6_sel • ch7_sel • ch8_sel • ch9_sel • ch10_sel • ch11_sel	disabled	TODO
RSTC-TRL_HARD_BLOCK_ENABLE		Mux	• hard_rst_ctl • pld_rst_ctl	hard_rst_ctl	TODO
RSTC-TRL_HIP_EP		Mux	• hip_not_ep • hip_ep	hip_not_ep	TODO
RSTC-TRL_LTSSM_DISABLE		Bool	t/f	f	TODO
RSTC-TRL_MASK_TX_PLL_LOCK_SELECT		Mux	• disabled • ch1_sel • ch4_sel • ch4_10_sel	disabled	TODO
RSTC-TRL_OFF_CAL_DONE_SELECT		Mux	• disabled • ch0_out • ch01_out • ch0123_out • ch0123_5678_out	disabled	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RSTC-TRL_OFF_CAL_EN_SELECT		Mux	• disabled • ch0_out • ch01_out • ch0123_out • ch0123_5678_out	disabled	TODO
RSTC-TRL_PERSTN_SELECT		Mux	• perstn_pin • perstn_pld	perstn_pin	TODO
RSTC-TRL_PERST_ENABLE		Mux	• level • neg_edge	level	TODO
RSTC-TRL_PLD_CLR		Bool	t/f	f	TODO
RSTC-TRL_RX_PCS_RST_N_INV		Bool	t/f	f	TODO
RSTC-TRL_RX_PCS_RST_N_SELECT		Mux	• disabled • ch0_out • ch01_out • ch0123_out • ch012345678_out • ch012345678_10_out	disabled	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RSTC-TRL_RX_PLL_FREQ_LOCK_SELECT		Mux	• disabled • ch0_sel • ch01_sel • ch0123_sel • ch0123_5678_sel • ch0123_5678_phs_sel • ch0123_phs_sel • ch01_phs_sel • ch0_phs_sel	disabled	TODO
RSTC-TRL_RX_PLL_LOCK_SELECT		Mux	• disabled • ch0_sel • ch01_sel • ch0123_sel • ch0123_5678_sel	disabled	TODO
RSTC-TRL_RX_PMA_RSTB_CMU_SELECT		Mux	• disabled • ch1cmu_sel • ch4cmu_sel • ch4_10cmu_sel	disabled	TODO
RSTC-TRL_RX_PMA_RSTB_INV		Bool	t/f	f	TODO
RSTC-TRL_RX_PMA_RSTB_SELECT		Mux	• disabled • ch0_out • ch01_out • ch0123_out • ch012345678_out • ch012345678_10_out	disabled	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RSTC-TRL_TIMER_A_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_A_VALUE		Ram	00-ff	0	TODO
RSTC-TRL_TIMER_B_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_B_VALUE		Ram	00-ff	0	TODO
RSTC-TRL_TIMER_C_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_C_VALUE		Ram	00-ff	0	TODO
RSTC-TRL_TIMER_D_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_D_VALUE		Ram	00-ff	0	TODO
RSTC-TRL_TIMER_E_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_E_VALUE		Ram	00-ff	0	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RSTC-TRL_TIMER_F_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_F_VALUE		Ram	00-ff	0	TODO
RSTC-TRL_TIMER_G_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_G_VALUE		Ram	00-ff	0	TODO
RSTC-TRL_TIMER_H_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_H_VALUE		Ram	00-ff	0	TODO
RSTC-TRL_TIMER_I_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_I_VALUE		Ram	00-ff	0	TODO
RSTC-TRL_TIMER_J_TYPE		Mux	• disabled • milli_secs • micro_secs • fref_cycles	disabled	TODO
RSTC-TRL_TIMER_J_VALUE		Ram	00-ff	0	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RSTC-TRL_TX_CMU_PLL_LOCK_SELECT		Mux	• disabled • ch1_sel • ch4_sel • ch4_10_sel	disabled	TODO
RSTC-TRL_TX_LC_PLL_LOCK_SELECT		Mux	• disabled • ch1_sel • ch7_sel	disabled	TODO
RSTC-TRL_TX_LC_PLL_RSTB_SELECT		Mux	• disabled • ch1_out • ch7_out	disabled	TODO
RSTC-TRL_TX_PCS_RST_N_INV		Bool	t/f	f	TODO
RSTC-TRL_TX_PCS_RST_N_SELECT		Mux	• disabled • ch0_out • ch01_out • ch0123_out • ch012345678_out • ch012345678_10_out	disabled	TODO
RSTC-TRL_TX_PMA_RSTB_INV		Bool	t/f	f	TODO
RSTC-TRL_TX_PMA_SYNCP_INV		Bool	t/f	f	TODO
RSTC-TRL_TX_PMA_SYNCP_SELECT		Mux	• disabled • ch1_out • ch4_out • ch4_10_out	disabled	TODO
RXFRE-QLK_CNT_DATA		Ram	20 bits	0	TODO
RXFRE-QLK_CNT_EN		Bool	t/f	f	TODO
RX_CDC_ALMOST_FULL_DATA		Ram	0-f	0	TODO
RX_L0S_COUNT_IDL_DATA		Ram	00-ff	0	TODO
RX_PTR0_NONPOSTED_DPRAM_MAX_DATA		Ram	000-3ff	0	TODO
RX_PTR0_NONPOSTED_DPRAM_MIN_DATA		Ram	000-3ff	0	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RX_PTR0_POSTED_DPRAM_MAX_DATA	0-7	Ram	000-3ff	0	TODO
RX_PTR0_POSTED_DPRAM_MIN_DATA	0-7	Ram	000-3ff	0	TODO
SINGLE_RX_DETECT_DATA		Ram	0-f	0	TODO
SKP_INSERTION_CONTROL		Bool	t/f	f	TODO
SKP_OS_SCHEDULE_COUNT_DATA		Ram	000-7ff	0	TODO
SLOTCLK_CFG		Mux	• dynamic_slotclkcfg • static_slotclkcfgoff • static_slotclkcfgon	dynamic_slotclkcfg	TODO
SLOT_REGISTER_EN		Bool	t/f	f	TODO
TESTMODE_CONTROL		Bool	t/f	f	TODO
TX_CDC_ALMOST_FULL_DATA		Ram	0-f	0	TODO
TX_L0S_ADJUST		Bool	t/f	f	TODO
TX_SWING_DATA		Ram	00-ff	0	TODO
USER_ID_DATA		Ram	0000-ffff	0	TODO
USE_CRC_FORWARDING		Bool	t/f	f	TODO
VC0_CLK_ENABLE		Bool	t/f	f	TODO
VC0_RX_BUFFER_MEMORY_SETTINGS_DATA		Ram	0000-ffff	0	TODO
VC0_RX_FLOW_CTRL_COMPL_DATA_DATA		Ram	000-fff	0	TODO
VC0_RX_FLOW_CTRL_COMPL_HEADER_DATA		Ram	00-ff	0	TODO
VC0_RX_FLOW_CTRL_NONPOSTED_DATA_DATA		Ram	00-ff	0	TODO
VC0_RX_FLOW_CTRL_NONPOSTED_HEADER_DATA		Ram	00-ff	0	TODO
VC0_RX_FLOW_CTRL_POSTED_DATA_DATA		Ram	000-fff	0	TODO
VC0_RX_FLOW_CTRL_POSTED_HEADER_DATA		Ram	00-ff	0	TODO
VC1_CLK_ENABLE		Bool	t/f	f	TODO
VC_ENABLE		Bool	t/f	f	TODO
VSEC_CAP_DATA		Ram	0-f	0	TODO
VSEC_ID_DATA		Ram	0000-ffff	0	TODO
ASPM_OPTIONALITY	0-7	Bool	t/f	f	TODO
BAR0_64BIT_MEM_SPACE	0-7	Bool	t/f	f	TODO
BAR0_IO_SPACE	0-7	Bool	t/f	f	TODO
BAR0_PREFETCHABLE	0-7	Bool	t/f	f	TODO
BAR0_SIZE_MASK_DATA	0-7	Ram	28 bits	0	TODO
BAR1_64BIT_MEM_SPACE	0-7	Mux	• disabled • enabled • all_one	disabled	TODO
BAR1_IO_SPACE	0-7	Bool	t/f	f	TODO
BAR1_PREFETCHABLE	0-7	Bool	t/f	f	TODO
BAR1_SIZE_MASK_DATA	0-7	Ram	28 bits	0	TODO
BAR2_64BIT_MEM_SPACE	0-7	Bool	t/f	f	TODO
BAR2_IO_SPACE	0-7	Bool	t/f	f	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
BAR2_PREFETCHABLE	0-7	Bool	t/f	f	TODO
BAR2_SIZE_MASK	0-7	Ram	28 bits	0	TODO
BAR3_64BIT_MEM_SPACE	0-7	Mux	• disabled • enabled • all_one	disabled	TODO
BAR3_IO_SPACE	0-7	Bool	t/f	f	TODO
BAR3_PREFETCHABLE	0-7	Bool	t/f	f	TODO
BAR3_SIZE_MASK	0-7	Ram	28 bits	0	TODO
BAR4_64BIT_MEM_SPACE	0-7	Bool	t/f	f	TODO
BAR4_IO_SPACE	0-7	Bool	t/f	f	TODO
BAR4_PREFETCHABLE	0-7	Bool	t/f	f	TODO
BAR4_SIZE_MASK	0-7	Ram	28 bits	0	TODO
BAR5_64BIT_MEM_SPACE	0-7	Mux	• disabled • enabled • all_one	disabled	TODO
BAR5_IO_SPACE	0-7	Bool	t/f	f	TODO
BAR5_PREFETCHABLE	0-7	Bool	t/f	f	TODO
BAR5_SIZE_MASK	0-7	Ram	28 bits	0	TODO
BRIDGE_PORT_SSID_SUPPORT	0-7	Bool	t/f	f	TODO
BRIDGE_PORT_VGA_ENABLE	0-7	Bool	t/f	f	TODO
CLASS_CODE_DATA	0-7	Ram	24 bits	0	TODO
COMPLETION_TIMEOUT	0-7	Mux	• cmpl_a • cmpl_ab • cmpl_abc • cmpl_abcd • cmpl_b • cmpl_bc • cmpl_bcd • disabled	cmpl_a	TODO
D0_PME	0-7	Bool	t/f	f	TODO
D1_PME	0-7	Bool	t/f	f	TODO
D1_SUPPORT	0-7	Bool	t/f	f	TODO
D2_PME	0-7	Bool	t/f	f	TODO
D2_SUPPORT	0-7	Bool	t/f	f	TODO
D3_COLD_PME	0-7	Bool	t/f	f	TODO
D3_HOT_PME	0-7	Bool	t/f	f	TODO
DEEMPHASIS_ENABLE	0-7	Bool	t/f	f	TODO
DEVICE_ID_DATA	0-7	Ram	0000-ffff	0	TODO
DEVICE_SPECIFIC_INIT	0-7	Bool	t/f	f	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
DIFFCLOCK_NFTS_COUNT_DATA	0-7	Ram	00-ff	0	TODO
DISABLE_SNOOP_PACKET	0-7	Bool	t/f	f	TODO
DLL_ACTIVE_REPORT_SUPPORT	0-7	Bool	t/f	f	TODO
ECRC_CHECK_CAPABLE	0-7	Bool	t/f	f	TODO
ECRC_GEN_CAPABLE	0-7	Bool	t/f	f	TODO
EIE_BEFORE_NFTS_COUNT_DATA	0-7	Ram	0-f	0	TODO
ELECTROMECH_INTERLOCK	0-7	Bool	t/f	f	TODO
ENABLE_COMPLETION_TIMEOUT_DISABLE	0-7	Bool	t/f	f	TODO
ENABLE_FUNCTION_MSIX_SUPPORT	0-7	Bool	t/f	f	TODO
ENABLE_L0S_ASPM	0-7	Bool	t/f	f	TODO
ENABLE_L1_ASPM	0-7	Bool	t/f	f	TODO
ENDPOINT_L0_LATENCY_DATA	0-7	Ram	0-7	0	TODO
ENDPOINT_L1_LATENCY_DATA	0-7	Ram	0-7	0	TODO
EXPANSION_BASE_ADDRESS_REGISTER_DATA_0	0-7	Ram	32 bits	0	TODO
EXTEND_TAG_FIELD	0-7	Bool	t/f	f	TODO
FLR_CAPABILITY	0-7	Bool	t/f	f	TODO
GEN2_DIFFCLOCK_NFTS_COUNT_DATA	0-7	Ram	00-ff	0	TODO
GEN2_SAMECLOCK_NFTS_COUNT_DATA	0-7	Ram	00-ff	0	TODO
HOT_PLUG_SUPPORT_DATA	0-7	Ram	00-7f	0	TODO
INDICATOR_DATA	0-7	Ram	0-7	0	TODO
INTEL_ID_ACCESS	0-7	Bool	t/f	f	TODO
INTERRUPT_PIN	0-7	Mux	• disabled • inta • intb • intc • intd	disabled	TODO
IO_WINDOW_ADDR_WIDTH	0-7	Mux	• disabled • window_16_bit • window_32_bit	disabled	TODO
L0_EXIT_LATENCY_DIFFCLOCK_DATA	0-7	Ram	0-7	0	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
L0_EXIT_LATENCY_SAMECLOCK_DATA	0-7	Ram	0-7	0	TODO
L1_EXIT_LATENCY_DIFFCLOCK_DATA	0-7	Ram	0-7	0	TODO
L1_EXIT_LATENCY_SAMECLOCK_DATA	0-7	Ram	0-7	0	TODO
L2_ASYNC_LOGIC	0-7	Bool	t/f	f	TODO
LOW_PRIORITY_VC	0-7	Bool	t/f	f	TODO
MAXIMUM_CURRENT_DATA	0-7	Ram	0-7	0	TODO
MAX_LINK_WIDTH	0-7	Mux	• disabled • x4 • x2 • x1 • x8	disabled	TODO
MAX_PAYLOAD_SIZE	0-7	Num	• 128 • 256 • 512	128	TODO
MSIX_PBA_BIR_DATA	0-7	Ram	0-7	0	TODO
MSIX_PBA_OFFSET_DATA	0-7	Ram	29 bits	0	TODO
MSIX_TABLE_BIR_DATA	0-7	Ram	0-7	0	TODO
MSIX_TABLE_OFFSET_DATA	0-7	Ram	29 bits	0	TODO
MSIX_TABLE_SIZE_DATA	0-7	Ram	000-7ff	0	TODO
MSI_64BIT_ADDRESSING_CAPABLE	0-7	Bool	t/f	f	TODO
MSI_MASKING_CAPABLE	0-7	Bool	t/f	f	TODO
MSI_MULTI_MESSAGE_CAPABLE	0-7	Num	• 1-2 • 4 • 8 • 16 • 32	1	TODO
MSI_SUPPORT	0-7	Bool	t/f	f	TODO
NO_COMMAND_COMPLETED	0-7	Bool	t/f	f	TODO
NO_SOFT_RESET	0-7	Bool	t/f	f	TODO
PCIE_SPEC_VERSION	0-7	Num	• 0-2	0	TODO

continues on next page

Table 10 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PORT-TYPE_FUNC	0-7	Mux	• ep_native • ep_legacy • rp • sw_up • sw_dn • bridge • switch_mode • shared_mode	ep_native	TODO
PREFETCH-ABLE_MEM_WINDOW_ADDR_WIDTH	0-7	Num	• 0 • 32 • 64	0	TODO
REVI-SION_ID_DATA	0-7	Ram	00-ff	0	TODO
ROLE_BASED_ERROR_REPORTING	0-7	Bool	t/f	f	TODO
RX_EI_L0S	0-7	Bool	t/f	f	TODO
SAME-CLOCK_NFTS_COUNT_DATA	0-7	Ram	00-ff	0	TODO
SLOT_NUMBER_DATA	0-7	Ram	0000-1fff	0	TODO
SLOT_POWER_LIMIT_DATA	0-7	Ram	00-ff	0	TODO
SLOT_POWER_SCALE_DATA	0-7	Ram	0-3	0	TODO
SSID_DATA	0-7	Ram	0000-ffff	0	TODO
SSVID_DATA	0-7	Ram	0000-ffff	0	TODO
SUBSYS-TEM_DEVICE_ID_DATA_0	0-7	Ram	0000-ffff	0	TODO
SUBSYS-TEM_VENDOR_ID_DATA_0	0-7	Ram	0000-ffff	0	TODO
SUR-PRISE_DOWN_ERROR_SUPPORT	0-7	Bool	t/f	f	TODO
USE_AER	0-7	Bool	t/f	f	TODO
VC_ARBITRATION	0-7	Bool	t/f	f	TODO
VEN-DOR_ID_DATA	0-7	Ram	0000-ffff	0	TODO
ALTPE2_HIP_BASE_ADDR_USER_	0-5	Ram	000-3ff	0	TODO
CVP_MDIO_DIS_CSR_CTRL_1	0-5	Bool	t/f	f	TODO
DFT_BROADCAST_EN_1	0-5	Bool	t/f	f	TODO
FORCE_MDIO_DIS_CSR_CTRL_1	0-5	Bool	t/f	f	TODO
POWER_ISOLATION_EN_1_DATA	0-5	Bool	t/f	f	TODO

2.3.14 DLL

The Delay-Locked loop does phase control for the DQS16.

TODO: everything

Name	Type	Values	Default	Documentation
A5_COUNTER_INIT	Num	• 3 • 12 • 24 • 40 • 48 • 72 • 80 • 96	3	TODO
ALOAD_INVERT_EN	Bool	t/f	f	TODO
ARMSTRONG_EN	Bool	t/f	f	TODO
DELAY_CHAIN_GLITCHCTRL_EN	Bool	t/f	f	TODO
DELAY_CONTROL	Mux	• bit7 • static	static	TODO
DLL_ADDI_EN	Bool	t/f	f	TODO
DLL_INPUT	Mux	• vss • sd_pll0 • sd_pll1 • cn_pll0 • cn_pll1 • tb_pll0 • tb_pll1	vss	TODO
DLL_RD_PD	Ram	0-7	0	TODO
JITTER_COUNTER_EN	Bool	t/f	t	TODO
JITTER_REDUCE_EN	Bool	t/f	t	TODO
RB_CO	Ram	0-3	3	TODO
STATIC_DLL_SETTING	Ram	00-7f	0	TODO
UPDNEN_EN	Bool	t/f	t	TODO
UPNDNIN	Mux	• bit4 • core	core	TODO
UPNDNIN_EN	Bool	t/f	t	TODO
UPNDNIN_INVERT_EN	Bool	t/f	t	TODO
UPNDNIN_INV_EN	Bool	t/f	t	TODO
UPWNDCORE	Mux	• upndn • updnen • up_ndn • refclk	upndn	TODO
USE_ALOAD	Bool	t/f	t	TODO

Port Name	Instance	Port bits	Route node type	Documentation
ASYNC_LOAD			GOUT	TODO
CTRL_OUT		0-6	GIN	TODO
LOCKED			GIN	TODO
UPNDN_IN			GOUT	TODO
UPNDN_IN_CLK_ENA			GOUT	TODO
UPNDN_OUT			GIN	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
CLKIN			<	FPLL:CLKDOUT	Dedicated differential I/O PLL counter to DLL

2.3.15 SERPAR

Unclear yet.

TODO: everything

Name	Type	Values	Default	Documentation
ENSER_SELECT	Mux	• disabled • block_0 • block_1 • block_2 • block_3	disabled	TODO

2.3.16 LVL

The Leveling Delay Chain does something linked to the DQS16.

TODO: everything

Name	Instance	Type	Values	Default	Documentation
ADDI_EN		Bool	t/f	f	TODO
CO_DELAY		Ram	0-3	3	TODO
DLL_SEL		Ram	0-1	0	TODO
FBOUT0_DELAY		Ram	0-3	0	TODO
FBOUT0_DELAY_PWR_SVG_EN		Bool	t/f	t	TODO
FBOUT1_DELAY		Ram	0-3	0	TODO
FBOUT1_DELAY_PWR_SVG_EN		Bool	t/f	t	TODO
PHY-CLK_GATING_DIS		Bool	t/f	f	TODO
PHYCLK_SEL		Ram	0-3	0	TODO
PHY-CLK_SEL_INV_EN		Bool	t/f	f	TODO
CLK_DELAY	0-3	Ram	0-3	0	TODO
CLK_DELAY_PWR_SVG_EN	0-3	Bool	t/f	f	TODO
CLK_GATING_DIS	0-3	Bool	t/f	f	TODO
CORE_INV_EN	0-3	Bool	t/f	f	TODO
DE-LAY_CLK_SEL	0-3	Mux	• core • pll	core	TODO
PLL_SEL	0-3	Num	• 1-3	1	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
			<	HMC	TODO

2.3.17 TERM

The TERM blocks control the On-Chip Termination circuitry

TODO: everything

Name	Type	Values	Default	Documentation
CALCLR_EN	Bool	t/f	f	TODO
CAL_MODE	Mux	• disabled • rs_12_15v • rs_18_30v	disabled	TODO
CLKENUSR_INV	Bool	t/f	f	TODO
ENSERUSR_INV	Bool	t/f	f	TODO
INTOSC_2_EN	Bool	t/f	t	TODO
NCLRUSR_INV	Bool	t/f	f	TODO
PLLBIAS_EN	Bool	t/f	f	TODO
POWERUP	Bool	t/f	f	TODO
RSADJUST_VAL	Mux	• disabled • rsadjust_10 • rsadjust_6p5 • rsadjust_3 • rsadjust_m3 • rsadjust_m6 • rsadjust_m9 • rsadjust_m12	disabled	TODO
RSHIFT_RDOWN_DIS	Bool	t/f	f	TODO
RSHIFT_RUP_DIS	Bool	t/f	f	TODO
RSMULT_VAL	Mux	• disabled • rsmult_1 • rsmult_2 • rsmult_3 • rsmult_4 • rsmult_5 • rsmult_6 • rsmult_7 • rsmult_10	rsmult_1	TODO
RTADJUST_VAL	Mux	• disabled • rtadjust_2p5v • rtad- just_1p5_1p8v	disabled	TODO
RTMULT_VAL	Mux	• disabled • rtmult_1 • rtmult_2 • rtmult_3 • rtmult_4 • rtmult_5 • rtmult_6	rtmult_1	TODO
SCANEN_INV	Bool	t/f	f	TODO
TEST_0_EN	Bool	t/f	f	TODO
TEST_1_EN	Bool	t/f	f	TODO
TEST_4_EN	Bool	t/f	f	TODO
TEST_5_EN	Bool	t/f	f	TODO
USER_OCT_INV	Bool	t/f	f	TODO
VREFH_LEVEL	Mux	• vref_m • vref_l • vref_h	vref_m	TODO

2.3.18 PMA3

The PMA3 blocks control triplets of channels used with the HSSI.

TODO: everything

Name	Instance	Type	Values	Default	Documentation
FPLL_DRV_EN		Bool	t/f	t	TODO
FPLL_REFCLK_SEL_IQ_TX_RX_CLK		Mux	• iq_tx_rx_clk0 • iq_tx_rx_clk1 • iq_tx_rx_clk2 • iq_tx_rx_clk3 • iq_tx_rx_clk4 • iq_tx_rx_clk5 • pd	pd	TODO
FPLL_SEL_IQ_TX_RX_CLK		Mux	• iq_tx_rx_clk0 • iq_tx_rx_clk1 • iq_tx_rx_clk2 • pd	pd	TODO
FPLL_SEL_REF_IQCLK		Mux	• ffpll_top • ref_iqclk0 • ref_iqclk1 • ref_iqclk2 • ref_iqclk3 • ffpll_bot • pd	pd	TODO
FPLL_SEL_RX_IQCLK		Mux	• rx_iqclk0 • rx_iqclk1 • rx_iqclk2 • rx_iqclk3 • pd	pd	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
HCLK_TOP_OUT_DRIVER		Mux	• tristate • up_en • down_en	down_en	TODO
SEGMENTED_0_UP_MUX_SEL		Mux	• other_segmented • pd_1 • ch0_txpll	ch0_txpll	TODO
X6_DRIVER_EN		Bool	t/f	f	TODO
AUTO_NEGOTIATION	1-2	Bool	t/f	f	TODO
CDR_PLL_ATB	0-2	Ram	0-f	0	TODO
CDR_PLL_BBPD_CLK0_OFFSET	0-2	Mux	• delta_0 • delta_1_left • delta_2_left • delta_3_left • delta_4_left • delta_5_left • delta_6_left • delta_7_left • delta_1_right • delta_2_right • delta_3_right • delta_4_right • delta_5_right • delta_6_right • delta_7_right	delta_0	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
CDR_PLL_BBPD_0_CLK180_OFFSET	CDR_PLL_BBPD_0	Mux	• delta_0 • delta_1_left • delta_2_left • delta_3_left • delta_4_left • delta_5_left • delta_6_left • delta_7_left • delta_1_right • delta_2_right • delta_3_right • delta_4_right • delta_5_right • delta_6_right • delta_7_right	delta_0	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
CDR_PLL_BBPD_0_CLK270_OFFSET	C12	Mux	• delta_0 • delta_1_left • delta_2_left • delta_3_left • delta_4_left • delta_5_left • delta_6_left • delta_7_left • delta_1_right • delta_2_right • delta_3_right • delta_4_right • delta_5_right • delta_6_right • delta_7_right	delta_0	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
CDR_PLL_BBPD_0_CLK90_OFFSET	CDR_PLL	Mux	• delta_0 • delta_1_left • delta_2_left • delta_3_left • delta_4_left • delta_5_left • delta_6_left • delta_7_left • delta_1_right • delta_2_right • delta_3_right • delta_4_right • delta_5_right • delta_6_right • delta_7_right	delta_0	TODO
CDR_PLL_BBPD_0_SEL	CDR_PLL	Mux	• normal • testmux	normal	TODO
CDR_PLL_CGB_CLK_EN	CDR_PLL	Bool	t/f	f	TODO
CDR_PLL_CLOCK_EN	CDR_PLL	Bool	t/f	f	TODO
CDR_PLL_COUNTER_PD_CLK_DIS	CDR_PLL	Bool	t/f	f	TODO
CDR_PLL_CPUMP_CURRENT_TEST	CDR_PLL	Mux	• normal • disable • test_down • test_up	normal	TODO
CDR_PLL_CP_RGLA_BYPASS_EN	CDR_PLL	Bool	t/f	f	TODO
CDR_PLL_DIAG_REV_LOOPBACK	CDR_PLL	Bool	t/f	f	TODO
CDR_PLL_FAST_LOCK_MODE_EN	CDR_PLL	Bool	t/f	t	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
CDR_PLL_FB_SEL	0-2	Mux	• vco_clk • external_clk	vco_clk	TODO
CDR_PLL_FREF_PPM_DIV2_EN	0-2	Bool	t/f	f	TODO
CDR_PLL_GPON_DETECTION_EN	0-2	Bool	t/f	f	TODO
CDR_PLL_IGNORE_PHASELOCK_EN	0-2	Bool	t/f	f	TODO
CDR_PLL_LEVSHIFT_POWER_TAP	0-2	Ram	0-3	1	TODO
CDR_PLL_L_COUNTER	0-2	Num	• 1-2 • 4 • 8	1	TODO
CDR_PLL_M_COUNTER	0-2	Num	• 0 • 4-5 • 8 • 10 • 12 • 16 • 20 • 25 • 32 • 40 • 50	20	TODO
CDR_PLL_ON	0-2	Bool	t/f	f	TODO
CDR_PLL_PCIE_FREQ_MHZ	0-2	Num	• 100 • 125	100	TODO
CDR_PLL_PD_CP_CURRENT_UA	0-2	Num	• 5 • 10 • 20 • 30 • 40	5	TODO
CDR_PLL_PD_L_COUNTER	0-2	Num	• 1-2 • 4 • 8	1	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
CDR_PLL_PFD_CHARGEPUMP_CURRENT_UA	0-2	Num	• 5 • 10 • 20 • 30 • 40 • 50 • 60 • 80 • 100 • 120	20	TODO
CDR_PLL_REF_CLK_DIV	0-2	Num	• 1-2 • 4 • 8	1	TODO
CDR_PLL_REGULATOR_INC_PCT	0-2	Mux	• p0 • p5 • p10 • p15 • p20 • p25 • disabled	p5	TODO
CDR_PLL_REPLICA_BIAS_DIS	0-2	Bool	t/f	f	TODO
CDR_PLL_RESERVE_LOOPBACK_EN	0-2	Bool	t/f	f	TODO
CDR_PLL_RIPPL_CAP_CTRL_EN	0-2	Bool	t/f	f	TODO
CDR_PLL_RXPLL_PD_BW_CTRL	0-2	Num	• 170 • 240 • 300 • 600	300	TODO
CDR_PLL_RXPLL_PFD_BW_CTRL	0-2	Num	• 1600 • 3200 • 4800 • 6400	3200	TODO
CDR_PLL_TXPLL_HCLK_DRIVER_EN	0-2	Bool	t/f	f	TODO
CDR_PLL_VCO_AUTO_RESET_EN	0-2	Bool	t/f	t	TODO
CDR_PLL_VCO_OVERANGE_REF	0-2	Ram	0-3	2	TODO
CDR_PLL_VLOCK_MONITOR	0-2	Mux	• mon_clk • mon_data	mon_clk	TODO
CVP_EN	0-2	Bool	t/f	f	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
DPRIO_REG_PLD_PMA_IF_BADDR	0-2	Ram	000-7ff		TODO
FORCE_MDIO_DIS	0-2	CSR_END	Bool	t/f	f	TODO
HCLK_PCS_DRIVER_EN	0-2	Bool	t/f	f	TODO
INT_EARLY_EIOSEL	0-2	Mux	• pcs • core	pcs	TODO
INT_FFCLK_EN	0-2	Bool	t/f	f	TODO
INT_LTR_SEL	0-2	Mux	• pcs • core	pcs	TODO
INT_PCIE_SWITCH	0-2	SEL	Mux	• pcs • core	pcs	TODO
INT_TXDERECTRX	0-2	SEL	Mux	• pcs • core	pcs	TODO
INT_TX_ELEC_IDLE	0-2	SEL	Mux	• pcs • core	pcs	TODO
IQ_CLK_TO_CH2_SEL	0-2	Mux	• ffpll_top • ffpll_bot • ref_clk0 • ref_clk1 • ref_clk2 • ref_clk3 • rx_clk0 • rx_clk1 • rx_clk2 • rx_clk3 • pd_pma	pd_pma	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
IQ_TX_RX_CLK_A_SEL	0-2	Mux	• a_pma_rx_b_pma_rx • a_pcs_rx_b_pcs_rx • a_pma_tx_b_pma_rx • a_pcs_tx_b_pcs_tx • a_tri_b_pcs_rx • a_tri_b_pcs_tx • a_pcs_tx_b_tri • tristate	tristate	TODO
IQ_TX_RX_TO_CB_FB	0-2	Mux	• clk0 • clk1 • clk2 • pd	pd	TODO
PCLK0_SEL	0-2	Ram	0-7	0	TODO
PCLK1_SEL	0-2	Ram	0-7	0	TODO
PCLK_SEL	0-2	Mux	• a_pma_rx_b_pma_rx • a_pcs_rx_b_pcs_rx • a_pma_tx_b_pma_rx • a_pcs_tx_b_pcs_tx • a_tri_b_pcs_rx • a_tri_b_pcs_tx • a_pcs_tx_b_tri • tristate	tristate	TODO
RX_BIT_SLIP_BYPASS_EN	0-2	Bool	t/f	t	TODO
RX_BUF_RX_ATB	0-2	Ram	0-f	0	TODO
RX_BUF_SD_3DB_GAIN_EN	0-2	Bool	t/f	f	TODO
RX_BUF_SD_CDRCLK_TO_CGB_EN	0-2	Bool	t/f	f	TODO
RX_BUF_SD_DIAG_LOOPBACK	0-2	Bool	t/f	f	TODO
RX_BUF_SD_EN	0-2	Bool	t/f	f	TODO
RX_BUF_SD_HALF_BW_EN	0-2	Bool	t/f	f	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RX_BUF_SD_OFF	0-2	Mux	• divrx_1 • divrx_2 • divrx_3 • divrx_4 • divrx_5 • divrx_6 • divrx_7 • divrx_8 • divrx_9 • divrx_10 • divrx_11 • divrx_12 • divrx_13 • divrx_14 • reserved_off_1 • reserved_off_2 • off_on_tx_divrx_1 • off_on_tx_divrx_2 • off_on_tx_divrx_3 • off_on_tx_divrx_4 • off_on_tx_divrx_5 • off_on_tx_divrx_6 • off_on_tx_divrx_7 • off_on_tx_divrx_8 • off_on_tx_divrx_9 • off_on_tx_divrx_10 • off_on_tx_divrx_11 • off_on_tx_divrx_12 • off_on_tx_divrx_13 • off_on_tx_divrx_14	divrx_2	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RX_BUF_SD_ON	0-2	Mux	• pulse_4 • pulse_6 • pulse_8 • pulse_10 • pulse_12 • pulse_14 • pulse_16 • pulse_18 • pulse_20 • pulse_22 • pulse_24 • pulse_26 • pulse_28 • pulse_30 • reserved_on_1 • reserved_on_2 • force_on	pulse_6	TODO
RX_BUF_SD_RX_ACGAIN_A	0-2	Mux	• v0 • v0p5 • v0p75 • v1	v0	TODO
RX_BUF_SD_RX_ACGAIN_V	0-2	Mux	• v0 • v0p5 • v0p75 • v1	v1	TODO
RX_BUF_SD_RX_CLK_DIV2_EN	0-2	Bool	t/f	f	TODO
RX_BUF_SD_RX_REFCLK_EN	0-2	Bool	t/f	f	TODO
RX_BUF_SD_TERM_SEL	0-2	Mux	• external • r150ohm • r120ohm • r100ohm • r85ohm	r100ohm	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RX_BUF_SD_THRESHOLD_MV	0-2	Num	• 15 • 20 • 25 • 30 • 35 • 40 • 45 • 50	30	TODO
RX_BUF_SD_VCM_SEL	0-2	Mux	• tristated1 • tristated2 • tristated3 • tristated4 • v0p35 • v0p50 • v0p55 • v0p60 • v0p65 • v0p70 • v0p75 • v0p80 • pull_down_strong • pull_down_weak • pull_up_strong • pull_up_weak	v0p80	TODO
RX_BUF_SX_PDB_EN	0-2	Bool	t/f	f	TODO
RX_BUF_VCM_CURRENT_ADD	0-2	Ram	0-3	1	TODO
RX_DESER_CLK_SEL	0-2	Mux	• or_cal • lc • pld	or_cal	TODO
RX_DESER_REVERSE_LOOPBACK	0-2	Mux	• rx • cdr	rx	TODO
RX_EN	0-2	Bool	t/f	f	TODO
RX_MODE_BITS	0-2	Num	• 8 • 10 • 16 • 20	8	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
RX_SDCLK_EN	0-2	Bool	t/f	f	TODO
RX_VCO_BYPASS	0-2	Mux	• clklow • fref • normal • normal_dont_care	normal	TODO
TX_BUF_CML_EN	0-2	Bool	t/f	f	TODO
TX_BUF_COMMON_MODE_DRIVER_SEL	0-2	Mux	• grounded • pull_down • pull_up • pull_up_vccela • tristated1 • tristated2 • tristated3 • tristated4 • v0p35 • v0p50 • v0p55 • v0p60 • v0p65 • v0p70 • v0p75 • v0p80	v0p65	TODO
TX_BUF_DFT_SEL	0-2	Mux	• vod_en_lsb • vod_en_msb • po1_en • disabled • pre_en_po2_en	pre_en_po2_en	TODO
TX_BUF_DRIVER_RESOLUTION_CTRL	0-2	Mux	• combination • disabled • offset_main • offset_po1	offset_main	TODO
TX_BUF_EN	0-2	Bool	t/f	f	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
TX_BUF_FIR_COEF_SEL	0-2	Mux	• ram • dynamic	ram	TODO
TX_BUF_LOCAL_OB_CTL	0-2	Mux	• r49ohm • r29ohm • r42ohm • r22ohm	r29ohm	TODO
TX_BUF_LST_ATB	0-2	Ram	0-f	0	TODO
TX_BUF_RX_DET_MODE	0-2	Ram	0-f	0	TODO
TX_BUF_RX_DET_PDB_EN	0-2	Bool	t/f	f	TODO
TX_BUF_SLEW_RATE_CTRL	0-2	Num	• 15 • 30 • 50 • 90 • 160	30	TODO
TX_BUF_SWING_BOOST_DIS	0-2	Bool	t/f	f	TODO
TX_BUF_TERM_SEL	0-2	Mux	• r150ohm • r120ohm • r100ohm • r85ohm • external	r100ohm	TODO
TX_BUF_VCM_CURRENT_ADD	0-2	Ram	0-3	1	TODO
TX_BUF_VOD_BOOST_DIS	0-2	Bool	t/f	f	TODO
TX_BUF_VOD_SW_1ST_POST_TAP	0-2	Ram	00-1f	0	TODO
TX_BUF_VOD_SW_MAIN_TAP	0-2	Ram	00-3f	0	TODO
TX_CGB_CLK_MUTE	0-2	Mux	• disable • enable_mute • enable_mute_master_channel	disable	TODO
TX_CGB_COUNTER_RESET_EN	0-2	Bool	t/f	f	TODO
TX_CGB_ENABLE	0-2	Bool	t/f	f	TODO
TX_CGB_FREF_VCO_BYPASS	0-2	Bool	t/f	f	TODO
TX_CGB_MUX_POWER_DOWN	0-2	Bool	t/f	f	TODO
TX_CGB_PCIE_RESET	0-2	Mux	• normal • pcie	normal	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
TX_CGB_RX_IQCLK_SEL	0-2	Mux	• cgb_x1_m_div • rx_output • tristate	tristate	TODO
TX_CGB_SYNC	0-2	Mux	• normal • sync_rst	sync_rst	TODO
TX_CGB_X1_CLOCK_SOURCE_SEL	0-2	Mux	• up_segmented • down_segmented • ffpll • ch1_txpll_t • ch2_txpll_b • same_ch_txpll • hf-clk_xn_up • hf-clk_cn1_x6_dn • hf-clk_xn_dn • hf-clk_ch1_x6_up	up_segmented	TODO
TX_CGB_X1_DIV_SEL	0-2	Num	• 1-2 • 4 • 8	1	TODO
TX_CGB_XN_CLOCK_SOURCE_SEL	0-2	Mux	• xn_up • ch1_x6_dn • xn_dn • ch1_x6_up • cgb_x1_m_div	cgb_x1_m_div	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
TX_MODE_BITS	0-2	Num	• 8 • 10 • 16 • 20 • 80	8	TODO
TX_SER_CLK_DIV2_TX_DESKEW	0-2	Ram	0-f	0	TODO
TX_SER_DUTY_CYCLE_TIME	0-2	Ram	0-7	3	TODO
TX_SER_FORCED_DATA_MODE_EN	0-2	Bool	t/f	f	TODO
TX_SER_POST_TAP_1_EN	0-2	Bool	t/f	f	TODO
TX_VREF_ES_TAP	0-2	Mux	• vref_10r_ov_18r • vref_11r_ov_19r • vref_12r_ov_20r • vref_13r_ov_21r • vref_14r_ov_22r	vref_12r_ov_20r	TODO
REF_IQCLK_BUF_EN	0-2	Bool	t/f	f	TODO
RX_IQCLK_BUF_EN	0-2	Bool	t/f	f	TODO
FF-PLL_IQTXRXCLK_DIRECTION	0-5	Mux	• tristate • up • down	tristate	TODO
FF-PLL_IQCLK_DIRECTION	0-1	Mux	• tristate • up • down		TODO
CLK-BUF_DIV2_EN		Bool	t/f	f	TODO
CLK-BUF_LVPECL_DIS		Bool	t/f	t	TODO
CLK-BUF_TERM_DIS		Bool	t/f	t	TODO
CLK-BUF_VCM_PUP		Mux	• tristate • vcc	tristate	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
SEGMENTED_0_DOWN_MUX_SEL		Mux	• ch2_txpll • other_segmented • pd_1	pd_1	TODO
SEGMENTED_1_DOWN_MUX_SEL		Mux	• fpllin • mux1 • ch0_txpll • pd_2	pd_2	TODO
SEGMENTED_1_UP_MUX_SEL		Mux	• fpllin • mux1 • ch2_txpll • pd_2 • ch1_txpll_bot • ch1_txpll_top	ch1_txpll_top	TODO
XN_DN_SEL		Mux	• xn_dn • x6_up • x6_dn • pd_xn_dn	pd_xn_dn	TODO
XN_UP_SEL		Mux	• xn_up • x6_up • x6_dn • pd_xn_up	pd_xn_up	TODO
CLK-BUF_DIV2_EN		Bool	t/f	f	TODO
CLK-BUF_LVPECL_DIS		Bool	t/f	t	TODO
CLK-BUF_TERM_DIS		Bool	t/f	t	TODO
CLK-BUF_VCM_PUP		Mux	• tristate • vcc	tristate	TODO
SEGMENTED_0_DOWN_MUX_SEL		Mux	• ch2_txpll • other_segmented • pd_1	pd_1	TODO

continues on next page

Table 11 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
SEGMENTED_1_DOWN_MUX_SEL		Mux	• ch1_txpll_bot • ch1_txpll_top • fpllin • mux2 • ch0_txpll • pd_2	pd_2	TODO
SEGMENTED_1_UP_MUX_SEL		Mux	• fpllin • mux2 • pd_2 • ch2_txpll	ch2_txpll	TODO

2.3.19 HMC

The Hardware memory controller controls sets of GPIOs to implement modern SDR and DDR memory interfaces. In the sx dies one of them is taken over by the HPS. They can be bypassed in favor of direct access to the GPIOs.

What triggers the bypass is unclear, but the default configuration is in bypass mode. When bypassed a direct connection is extablished between two pnodes with the same coordinates and only a different port type. The source ports DDIOPHYDQDIN are connected to IOINTDQDIN, routing the inputs to the chip, while the source ports IOINT* are connected to the corresponding PHYDDIO* ports.

TODO: everything

Name	Instance	Type	Values	Default	Documentation
AC_DELAY_EN		Ram	0-3	0	TODO
ADDR_ORDER		Mux	• chip_row_bank_col • chip_bank_row_col • row_chip_bank_col	chip_row_bank_col	TODO
ATTR_COUNTER_ONE_MASK		Ram	64 bits	0	TODO
ATTR_COUNTER_ONE_MATCH		Ram	64 bits	0	TODO
ATTR_COUNTER_ONE_RESET		Ram	0-1	0	TODO
ATTR_COUNTER_ZERO_MASK		Ram	64 bits	0	TODO
ATTR_COUNTER_ZERO_MATCH		Ram	64 bits	0	TODO
ATTR_COUNTER_ZERO_RESET		Ram	0-1	0	TODO
ATTR_DEBUG_SELECT_BYTE		Ram	32 bits	0	TODO
ATTR_STATIC_CONFIG_VALID		Bool	t/f	f	TODO
A_CSR_ATPG_EN		Bool	t/f	f	TODO

continues on next page

Table 12 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
A_CSR_LPDDR_DIS		Bool	t/f	f	TODO
A_CSR_PIPELINEGLOBALENABLE		Bool	t/f	f	TODO
A_CSR_RESET_DELAY_EN		Bool	t/f	f	TODO
A_CSR_WRAP_BC_EN		Bool	t/f	f	TODO
CAL_REQ		Bool	t/f	f	TODO
CFG_BURST_LENGTH		Num	• 0 • 2 • 4 • 8 • 16	0	TODO
CFG_INTERFACE_WIDTH		Num	• 0 • 8 • 16 • 24 • 32 • 40	0	TODO
CFG_SELF_RFSH_EXIT_CYCLES		Num	• 0 • 37 • 44 • 52 • 59 • 74 • 88 • 200 • 512	0	TODO
CFG_STARVE_LIMIT		Ram	00-3f	0	TODO
CFG_TYPE		Mux	• ddr • ddr2 • ddr3 • lpddr • lpddr2	ddr	TODO
CLR_INTR		Bool	t/f	f	TODO
CTL_ECC_ENABLED		Bool	t/f	f	TODO
CTL_ECC_RMW_ENABLED		Bool	t/f	f	TODO
CTL_REGDIMM_ENABLED		Bool	t/f	f	TODO
CTL_USR_REFRESH		Bool	t/f	f	TODO
DATA_WIDTH		Num	• 16 • 32 • 64	16	TODO

continues on next page

Table 12 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
DBE_INTR		Bool	t/f	f	TODO
DDIO_ADDR_EN		Ram	0000-ffff	0	TODO
DDIO_BA_EN		Ram	0-7	0	TODO
DDIO_CAS_N_EN		Bool	t/f	f	TODO
DDIO_CKE_EN		Ram	0-3	0	TODO
DDIO_CS0_N_EN		Ram	0-3	0	TODO
DDIO_DM_EN		Ram	00-1f	0	TODO
DDIO_DQSB_EN		Ram	00-1f	0	TODO
DDIO_DQSLOGIC_EN		Ram	00-1f	0	TODO
DDIO_DQS_EN		Ram	00-1f	0	TODO
DDIO_DQ_EN		Ram	45 bits	0	TODO
DDIO_MEM_CLK_EN		Bool	t/f	f	TODO
DDIO_MEM_CLK_N_EN		Bool	t/f	f	TODO
DDIO_ODT_EN		Ram	0-3	0	TODO
DDIO_RAS_N_EN		Bool	t/f	f	TODO
DDIO_RESET_N_EN		Bool	t/f	f	TODO
DDIO_WE_N_EN		Bool	t/f	f	TODO
DELAY_BONDING		Ram	0-3	0	TODO
DFX_BYPASS_ENABLE		Bool	t/f	f	TODO
DISABLE_MERGING		Bool	t/f	f	TODO
DQA_DELAY_EN		Ram	0-3	0	TODO
DQSLOGIC_DELAY_EN		Ram	0-3	0	TODO
DQ_DELAY_EN		Ram	0-3	0	TODO
ENABLE_ATPG		Bool	t/f	f	TODO
ENABLE_BONDING_WRAPBACK		Bool	t/f	f	TODO
ENABLE_BURST_INTERRUPT		Bool	t/f	f	TODO
ENABLE_BURST_TERMINATE		Bool	t/f	f	TODO
ENABLE_DQS_TRACKING		Bool	t/f	f	TODO
ENABLE_ECC_CODE_OVERWRITES		Bool	t/f	f	TODO
ENABLE_INTR		Bool	t/f	f	TODO
ENABLE_NO_DM		Bool	t/f	f	TODO
ENABLE_PIPELINEGLOBAL		Bool	t/f	f	TODO
EXTRA_CTL_CLK_ACT_TO_ACT		Ram	0-f	0	TODO
EXTRA_CTL_CLK_ACT_TO_ACT_DIFF_BANK		Ram	0-f	0	TODO

continues on next page

Table 12 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
EXTRA_CTL_CLK_ACT_TO_PCH		Ram	0-f	0	TODO
EXTRA_CTL_CLK_ACT_TO_RDWR		Ram	0-f	0	TODO
EXTRA_CTL_CLK_ARF_PERIOD		Ram	0-f	0	TODO
EXTRA_CTL_CLK_ARF_TO_VALID		Ram	0-f	0	TODO
EXTRA_CTL_CLK_FOUR_ACT_TO_ACT		Ram	0-f	0	TODO
EXTRA_CTL_CLK_PCH_ALL_TO_VALID		Ram	0-f	0	TODO
EXTRA_CTL_CLK_PCH_TO_VALID		Ram	0-f	0	TODO
EXTRA_CTL_CLK_PDN_PERIOD		Ram	0-f	0	TODO
EXTRA_CTL_CLK_PDN_TO_VALID		Ram	0-f	0	TODO
EXTRA_CTL_CLK_RD_AP_TO_VALID		Ram	0-f	0	TODO
EXTRA_CTL_CLK_RD_TO_PCH		Ram	0-f	0	TODO
EXTRA_CTL_CLK_RD_TO_RD		Ram	0-f	0	TODO
EXTRA_CTL_CLK_RD_TO_RD_DIFF_CHIP		Ram	0-f	0	TODO
EXTRA_CTL_CLK_RD_TO_WR		Ram	0-f	0	TODO
EXTRA_CTL_CLK_RD_TO_WR_BC		Ram	0-f	0	TODO
EXTRA_CTL_CLK_RD_TO_WR_DIFF_CHIP		Ram	0-f	0	TODO
EXTRA_CTL_CLK_SRF_TO_VALID		Ram	0-f	0	TODO
EXTRA_CTL_CLK_SRF_TO_ZQ_CAL		Ram	0-f	0	TODO
EXTRA_CTL_CLK_WR_AP_TO_VALID		Ram	0-f	0	TODO
EXTRA_CTL_CLK_WR_TO_PCH		Ram	0-f	0	TODO
EXTRA_CTL_CLK_WR_TO_RD		Ram	0-f	0	TODO
EXTRA_CTL_CLK_WR_TO_RD_BC		Ram	0-f	0	TODO
EXTRA_CTL_CLK_WR_TO_RD_DIFF_CHIP		Ram	0-f	0	TODO
EXTRA_CTL_CLK_WR_TO_WR		Ram	0-f	0	TODO

continues on next page

Table 12 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
EXTRA_CTL_CLK_WR_TO_WR_DIFF_CHIP		Ram	0-f	0	TODO
GANGED_ARF		Bool	t/f	f	TODO
GEN_DBE		Ram	0-1	0	TODO
GEN_SBE		Ram	0-1	0	TODO
IF_DQS_WIDTH		Num	• 0-5	0	TODO
INC_SYNC		Num	• 2-3	2	TODO
LOCAL_IF_CS_WIDTH		Num	• 0-4	0	TODO
MASK_CORR_DROPPED_INTR		Bool	t/f	f	TODO
MEM_AUTO_PD_CYCLES		Ram	0000-ffff	0	TODO
MEM_CLK_ENTRY_CYCLES		Ram	0-f	0	TODO
MEM_IF_AL		Num	• 0-10	0	TODO
MEM_IF_BANKADDR_WIDTH		Num	• 0 • 2-3	0	TODO
MEM_IF_COLADDR_WIDTH		Num	• 0 • 8-12	0	TODO
MEM_IF_ROWADDR_WIDTH		Num	• 0 • 12-16	0	TODO
MEM_IF_TCCD		Num	• 0-4	0	TODO
MEM_IF_TCL		Num	• 0 • 3-11	0	TODO
MEM_IF_TCWL		Num	• 0-8	0	TODO
MEM_IF_TFAW		Num	• 0-32	0	TODO

continues on next page

Table 12 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
MEM_IF_TMRD		Num	• 0 • 2 • 4	0	TODO
MEM_IF_TRAS		Num	• 0-29	0	TODO
MEM_IF_TRC		Num	• 0-40	0	TODO
MEM_IF_TRCD		Num	• 0-11	0	TODO
MEM_IF_TREFI		Ram	0000-1fff	0	TODO
MEM_IF_TRFC		Ram	00-ff	0	TODO
MEM_IF_TRP		Num	• 0 • 2-10	0	TODO
MEM_IF_TRRD		Num	• 0-6	0	TODO
MEM_IF_TRTP		Num	• 0-8	0	TODO
MEM_IF_TWR		Num	• 0-12	0	TODO
MEM_IF_TWTR		Num	• 0-6	0	TODO
MMR_CFG_MEM_BL		Num	• 2 • 4 • 8 • 16	2	TODO
OUTPUT_REGD		Bool	t/f	f	TODO
PDN_EXIT_CYCLES		Mux	• disabled • fast • slow	disabled	TODO
POWER_SAVING_EXIT_CYCLES		Ram	0-f	0	TODO

continues on next page

Table 12 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PRIORITY_REMAP		Mux	• disabled • priority_0 • priority_1 • priority_2 • priority_3 • priority_4 • priority_5 • priority_6 • priority_7	disabled	TODO
READ_ODT_CHIP		Mux	• disabled • read_chip0_odt0_chip1 • read_chip0_odt1_chip1 • read_chip0_odt01_chip1 • read_chip0_chip1_odt0 • read_chip0_odt0_chip1_odt0 • read_chip0_odt1_chip1_odt0 • read_chip0_odt01_chip1_odt0 • read_chip0_chip1_odt1 • read_chip0_odt0_chip1_odt1 • read_chip0_odt1_chip1_odt1 • read_chip0_odt01_chip1_odt1 • read_chip0_chip1_odt01 • read_chip0_odt0_chip1_odt01 • read_chip0_odt1_chip1_odt01 • read_chip0_odt01_chip1_odt01	disabled	TODO
REORDER_DATA		Bool	t/f	f	TODO
SBE_INTR		Bool	t/f	f	TODO
TEST_MODE		Bool	t/f	f	TODO
USER_ECC_EN		Bool	t/f	f	TODO

continues on next page

Table 12 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
WRITE_ODT_CHIP		Mux	• disabled • write_chip0_odt0_chip1 • write_chip0_odt1_chip1 • write_chip0_odt01_chip1 • write_chip0_chip1_odt0 • write_chip0_odt0_chip1_odt0 • write_chip0_odt1_chip1_odt0 • write_chip0_odt01_chip1_odt0 • write_chip0_chip1_odt1 • write_chip0_odt0_chip1_odt1 • write_chip0_odt1_chip1_odt1 • write_chip0_odt01_chip1_odt1 • write_chip0_chip1_odt01 • write_chip0_odt0_chip1_odt01 • write_chip0_odt1_chip1_odt01 • write_chip0_odt01_chip1_odt01	disabled	TODO
INST_ROM_DATA	0-127	Ram	20 bits	0	TODO
AC_ROM_DATA	0-39	Ram	30 bits	0	TODO
AUTO_PCH_ENABLE		Bool	t/f	f	TODO
CLOCK_OFF	0-5	Bool	t/f	f	TODO
CPORT_RDY_ALMOST_FULL	0-5	Bool	t/f	f	TODO
CPORT_RFIFO_MAP	0-5	Ram	0-3	0	TODO
CPORT_TYPE	0-5	Mux	• disabled • write • read • bi_direction	disabled	TODO
CPORT_WFIFO_MAP	0-5	Ram	0-3	0	TODO
CYC_TO_RLD_JAM	0-5	Ram	00-ff	0	TODO
ENABLE_BONDING	0-5	Bool	t/f	f	TODO

continues on next page

Table 12 – continued from previous page

Name	Instance	Type	Values	Default	Documentation
PORT_WIDTH	0-5	Num	• 32 • 64 • 128 • 256	32	TODO
RCFG_STATIC_WEIGHT	0-5	Ram	00-1f	0	TODO
RCFG_USER_PRIORITY	0-5	Ram	0-7	0	TODO
THLD_JAR1	0-5	Ram	00-3f	0	TODO
THLD_JAR2	0-5	Ram	00-3f	0	TODO
RFIFO_CPORT_MAP	0-3	Num	• 0-5	0	TODO
SINGLE_READY	0-3	Mux	• concatenate • separate	concatenate	TODO
SYNC_MODE	0-3	Mux	• asynchronous • synchronous	asynchronous	TODO
USE_ALMOST_EMPTY	0-3	Bool	t/f	f	TODO
WFIFO_CPORT_MAP	0-3	Num	• 0-5	0	TODO
WFIFO_RDY_ALMOST_FULL	0-3	Bool	t/f	f	TODO
RCFG_SUM_WT_PRIORITY	0-7	Ram	00-ff	0	TODO

Port Name	Instance	Port bits	Route node type	Documentation
AFICTLLONGIDLE		0-1	GIN	TODO
AFICTLREFRESHDONE		0-1	GIN	TODO
AFISEQBUSY		0-1	GOUT	TODO
AVLADDRESS		0-15	GOUT	TODO
AVLREAD			GOUT	TODO
AVLREADDATA		0-31	GIN	TODO
AVLRESETN			GOUT	TODO
AVLWAITREQUEST			GIN	TODO
AVLWRITE			GOUT	TODO
AVLWRITEDATA		0-31	GOUT	TODO
BONDINGIN	0-2	0-5	GOUT	TODO
BONDINGOUT	0-2	0-5	GIN	TODO
CTLCALREQ			GIN	TODO
GLOBALRESETN			GOUT	TODO
IAVSTCMDDATA	0-5	0-41	GOUT	TODO
IAVSTCMDRESETN	0-5		GOUT	TODO

continues on next page

Table 13 – continued from previous page

Port Name	Instance	Port bits	Route node type	Documentation
IAVSTRDCLK	0-3		DCMUX	TODO
IAVSTRDREADY	0-3		GOUT	TODO
IAVSTRDRESETN	0-3		GOUT	TODO
IAVSTWRACKREADY	0-5		GOUT	TODO
IAVSTWRCLK		0-3	DCMUX	TODO
IAVSTWRDATA	0-3	0-89	GOUT	TODO
IAVSTWRRESETN	0-3		GOUT	TODO
IOINTADDRACLR		0-15	GOUT	TODO
IOINTADDRDOUT		0-63	GOUT	TODO
IOINTAFICALFAIL			GIN	TODO
IOINTAFICALSUCCESS			GIN	TODO
IOINTAFIRLAT		0-4	GIN	TODO
IOINTAFIWLAT		0-3	GIN	TODO
IOINTBAACLR		0-2	GOUT	TODO
IOINTBADOUT		0-11	GOUT	TODO
IOINTCASNACLR			GOUT	TODO
IOINTCASNDOUT		0-3	GOUT	TODO
IOINTCKDOUT		0-3	GOUT	TODO
IOINTCKEACLR		0-1	GOUT	TODO
IOINTCKEDOUT		0-7	GOUT	TODO
IOINTCKNDOUT		0-3	GOUT	TODO
IOINTCSNACLR		0-1	GOUT	TODO
IOINTCSNDOUT		0-7	GOUT	TODO
IOINTDMDOUT		0-19	GOUT	TODO
IOINTDQDIN		0-31, 36-67, 72-103, 108-139, 144-175	GIN	TODO
IOINTDQDOUT		0-31, 36-67, 72-103, 108-139, 144-175	GOUT	TODO
IOINTDQOE		0-15, 18-33, 36-51, 54-69, 72-87	GOUT	TODO
IOINTDQSBDOUT		0-19	GOUT	TODO
IOINTDQSBOE		0-9	GOUT	TODO
IOINTDQSDOUT		0-19	GOUT	TODO
IOINTDQSLOGICACLRFIFOCTRL		0-4	GOUT	TODO
IOINTDQSLOGICACLRPSTAMBLE		0-4	GOUT	TODO
IOINTDQSLOGICDQSENA		0-9	GOUT	TODO
IOINTDQSLOGICFIFORESET		0-4	GOUT	TODO
IOINTDQSLOGICINCRDATAEN		0-9	GOUT	TODO
IOINTDQSLOGICINCWRPTR		0-9	GOUT	TODO
IOINTDQSLOGICOCT		0-9	GOUT	TODO
IOINTDQSLOGICRDATAVALID		0-4	GIN	TODO
IOINTDQSLOGICREADLATENCY		0-24	GOUT	TODO
IOINTDQSOE		0-9	GOUT	TODO
IOINTODTACLR		0-1	GOUT	TODO
IOINTODTDOUT		0-7	GOUT	TODO
IOINTRASNACLR			GOUT	TODO
IOINTRASNDOUT		0-3	GOUT	TODO
IOINTRESETNACLR			GOUT	TODO
IOINTRESETNDOUT		0-3	GOUT	TODO
IOINTWENACLR			GOUT	TODO
IOINTWENDOUT		0-3	GOUT	TODO
LOCALDEEPPOWERDNACK			GIN	TODO

continues on next page

Table 13 – continued from previous page

Port Name	Instance	Port bits	Route node type	Documentation
LOCALDEEPPOWERDNCHIP		0-1	GOUT	TODO
LOCALDEEPPOWERDNREQ			GOUT	TODO
LOCALINITDONE			GIN	TODO
LOCALPOWERDOWNACK			GIN	TODO
LOCALREFRESHACK			GIN	TODO
LOCALREFRESHCHIP		0-1	GOUT	TODO
LOCALREFRESHREQ			GOUT	TODO
LOCALSELFRFSHACK			GIN	TODO
LOCALSELFRFSHCHIP		0-1	GOUT	TODO
LOCALSELFRFSHREQ			GOUT	TODO
MMRADDR		0-9	GOUT	TODO
MMRBE			GOUT	TODO
MMRBURSTBEGIN			GOUT	TODO
MMRBURSTCOUNT		0-1	GOUT	TODO
MMRCLK			DCMUX	TODO
MMRRDATA		0-7	GIN	TODO
MMRRDATAVALID			GIN	TODO
MMRREADREQ			GOUT	TODO
MMRRESETN			GOUT	TODO
MMRWAITREQUEST			GIN	TODO
MMRWDATA		0-7	GOUT	TODO
MMRWRITEREQ			GOUT	TODO
OAMMREADY		0-5	GIN	TODO
ORDAVSTDATA	0-3	0-79	GIN	TODO
ORDAVSTVALID	0-3		GIN	TODO
OWRACKAVSTDATA	0-5		GIN	TODO
OWRACKAVSTVALID	0-5		GIN	TODO
PHYRESETN			GIN	TODO
PLLLOCKED			GOUT	TODO
PORTCLK	0-5		DCMUX	TODO
SCADDR		0-9	GOUT	TODO
SCANEN			GOUT	TODO
SCBE			GOUT	TODO
SCBURSTBEGIN			GOUT	TODO
SCBURSTCOUNT		0-1	GOUT	TODO
SCCLK			DCMUX	TODO
SCRDATA		0-7	GIN	TODO
SCRDATAVALID			GIN	TODO
SCREADREQ			GOUT	TODO
SCRESETN			GOUT	TODO
SCWAITREQUEST			GIN	TODO
SCWDATA		0-7	GOUT	TODO
SCWRITEREQ			GOUT	TODO
SOFTRESETN			GOUT	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
	0-4		>	DQS16	TODO
			>	LVL	TODO

continues on next page

Table 14 – continued from previous page

Port Name	Instance	Port bits	Dir	Remote port	Documentation
DDIOPHYDQDIN		0-31, 36-67, 72-103, 108-139, 144-175	<	GPIO:DATAOUT	TODO
PHYDDIOADDRACLR		0-15	>	GPIO:ACLR	TODO
PHYDDIOADDRDOUT		0-63	>	GPIO:DATAIN	TODO
PHYDDIOBAACLR			>	GPIO:ACLR	TODO
PHYDDIOBADOUT		0-11	>	GPIO:DATAIN	TODO
PHYDDIOCASNACLR			>	GPIO:ACLR	TODO
PHYDDIOCASNDOUT		0-3	>	GPIO:DATAIN	TODO
PHYDDIOCKDOUT		0-3	>	GPIO:DATAIN	TODO
PHYDDIOCKEACLR		0-1	>	GPIO:ACLR	TODO
PHYDDIOCKEDOUT		0-7	>	GPIO:DATAIN	TODO
PHYDDIOCKNDOUT		0-3	>	GPIO:DATAIN	TODO
PHYDDIOCSNACLR		0-1	>	GPIO:ACLR	TODO
PHYDDIOCSNDOUT		0-7	>	GPIO:DATAIN	TODO
PHYDDIODMDOUT		0-19	>	GPIO:DATAIN	TODO
PHYDDIODQDOUT		0-31, 36-67, 72-103, 108-139, 144-175	>	GPIO:DATAIN	TODO
PHYDDIODQOE		0-15, 18-33, 36-51, 54-69, 72-87	>	GPIO:OEIN	TODO
PHYDDIODQSBDOUT		0-19	>	GPIO:DATAIN	TODO
PHYDDIODQSBOE		0-9	>	GPIO:OEIN	TODO
PHYDDIODQSDOUT		0-19	>	GPIO:DATAIN	TODO
PHYDDIODQSOE		0-9	>	GPIO:OEIN	TODO
PHYDDIOODTACLR		0-1	>	GPIO:ACLR	TODO
PHYDDIOODTDOUT		0-7	>	GPIO:DATAIN	TODO
PHYDDIORASNACLR			>	GPIO:ACLR	TODO
PHYDDIORASNDOUT		0-3	>	GPIO:DATAIN	TODO
PHYDDIORESETNACLR			>	GPIO:ACLR	TODO
PHYDDIORESETNDOUT		0-3	>	GPIO:DATAIN	TODO
PHYDDIOWENACLR			>	GPIO:ACLR	TODO
PHYDDIOWENDOUT		0-3	>	GPIO:DATAIN	TODO

2.3.20 HPS

The interface between the FPGA and the Hard processor system is done through 37 specialized blocks of 28 different types.

TODO: everything. GOUT/GIN/DCMUX mapping is done except for HPS_CLOCKS.

HPS_BOOT

Port Name	Instance	Port bits	Route node type	Documentation
BOOT_FROM_FPGA_ON_FAILURE			GOUT	TODO
BOOT_FROM_FPGA_READY			GOUT	TODO
BSEL		0-2	GOUT	TODO
BSEL_EN			GOUT	TODO
CSEL		0-1	GOUT	TODO
CSEL_EN			GOUT	TODO

HPS_CLOCKS

Name	Instance	Type	Values	Default	Documentation
RIGHT_CLOCK_SEL	0-8	Ram	0-3	3	TODO
TOP_CLOCK_SEL	0-8	Ram	0-3	3	TODO

Port Name	Instance	Port bits	Dir	Remote port	Documentation
CLKOUT	0	0-3	>	CMUXHG:PLLIN	HPS clock output to clock mux
CLKOUT	0	0-8	>	CMUXHR:PLLIN	HPS clock output to clock mux
CLKOUT	1	5-8	>	CMUXVG:PLLIN	HPS clock output to clock mux
CLKOUT	1	0-8	>	CMUXVR:PLLIN	HPS clock output to clock mux

HPS_CLOCKS_RESETS

Port Name	Instance	Port bits	Route node type	Documentation
F2H_COLD_RST_REQ_N			GOUT	TODO
F2H_DBG_RST_REQ_N			GOUT	TODO
F2H_PENDING_RST_ACK			GOUT	TODO
F2H_PERIPH_REF_CLK			DCMUX	TODO
F2H_SDRAM_REF_CLK			DCMUX	TODO
F2H_WARM_RST_REQ_N			GOUT	TODO
H2F_PENDING_RST_REQ_N			GIN	TODO
PTP_REF_CLK			DCMUX	TODO

HPS_CROSS_TRIGGER

Port Name	Instance	Port bits	Route node type	Documentation
ASICCTL		0-7	GIN	TODO
CLK			DCMUX	TODO
CLK_EN			GOUT	TODO
TRIG_IN		0-7	GOUT	TODO
TRIG_INACK		0-7	GIN	TODO
TRIG_OUT		0-7	GIN	TODO
TRIG_OUTACK		0-7	GOUT	TODO

HPS_DBG_APB

Port Name	Instance	Port bits	Route node type	Documentation
DBG_APB_DISABLE			GOUT	TODO
P_ADDR		0-17	GIN	TODO
P_ADDR_31			GIN	TODO
P_CLK			DCMUX	TODO
P_CLK_EN			GOUT	TODO
P_ENABLE			GIN	TODO
P_RDATA		0-31	GOUT	TODO
P_READY			GOUT	TODO
P_RESET_N			GIN	TODO
P_SEL			GIN	TODO
P_SLV_ERR			GOUT	TODO
P_WDATA		0-31	GIN	TODO
P_WRITE			GIN	TODO

HPS_DMA

Port Name	Instance	Port bits	Route node type	Documentation
ACK	0-7		GIN	TODO
REQ	0-7		GOUT	TODO
SINGLE	0-7		GOUT	TODO

HPS_FPGA2HPS

Port Name	Instance	Port bits	Route node type	Documentation
ARADDR		0-31	GOUT	TODO
ARBURST		0-1	GOUT	TODO
ARCACHE		0-3	GOUT	TODO
ARID		0-7	GOUT	TODO
ARLEN		0-3	GOUT	TODO
ARLOCK		0-1	GOUT	TODO
ARPROT		0-2	GOUT	TODO
ARREADY			GIN	TODO
ARSIZE		0-2	GOUT	TODO
ARUSER		0-4	GOUT	TODO
ARVALID			GOUT	TODO
AWADDR		0-31	GOUT	TODO
AWBURST		0-1	GOUT	TODO
AWCACHE		0-3	GOUT	TODO
AWID		0-7	GOUT	TODO
AWLEN		0-3	GOUT	TODO
AWLOCK		0-1	GOUT	TODO
AWPROT		0-2	GOUT	TODO
AWREADY			GIN	TODO
AWSIZE		0-2	GOUT	TODO

continues on next page

Table 15 – continued from previous page

Port Name	Instance	Port bits	Route node type	Documentation
AWUSER		0-4	GOUT	TODO
AWVALID			GOUT	TODO
BID		0-7	GIN	TODO
BREADY			GOUT	TODO
BRESP		0-1	GIN	TODO
BVALID			GIN	TODO
CLK			DCMUX	TODO
PORT_SIZE_CONFIG		0-1	GOUT	TODO
RDATA		0-127	GIN	TODO
RID		0-7	GIN	TODO
RLAST			GIN	TODO
RREADY			GOUT	TODO
RRESP		0-1	GIN	TODO
RVALID			GIN	TODO
WDATA		0-127	GOUT	TODO
WID		0-7	GOUT	TODO
WLAST			GOUT	TODO
WREADY			GIN	TODO
WSTRB		0-15	GOUT	TODO
WVALID			GOUT	TODO

HPS_FPGA2SDRAM

Port Name	Instance	Port bits	Route node type	Documentation
BONDING_OUT	0-1	0-3	GIN	TODO
CFG_AXI_MM_SELECT		0-5	GOUT	TODO
CFG_CPORT_RFIFO_MAP		0-17	GOUT	TODO
CFG_CPORT_TYPE		0-11	GOUT	TODO
CFG_CPORT_WFIFO_MAP		0-17	GOUT	TODO
CFG_PORT_WIDTH		0-11	GOUT	TODO
CFG_RFIFO_CPORT_MAP		0-15	GOUT	TODO
CFG_WFIFO_CPORT_MAP		0-15	GOUT	TODO
CMD_DATA	0-5	0-59	GOUT	TODO
CMD_PORT_CLK	0-5		DCMUX	TODO
CMD_READY	0-5		GIN	TODO
CMD_VALID	0-5		GOUT	TODO
RD_CLK	0-3		DCMUX	TODO
RD_DATA	0-3	0-79	GIN	TODO
RD_READY	0-3		GOUT	TODO
RD_VALID	0-3		GIN	TODO
WRACK_DATA	0-5	0-9	GIN	TODO
WRACK_READY	0-5		GOUT	TODO
WRACK_VALID	0-5		GIN	TODO
WR_CLK	0-3		DCMUX	TODO
WR_DATA	0-3	0-89	GOUT	TODO
WR_READY	0-3		GIN	TODO
WR_VALID	0-3		GOUT	TODO

HPS_HPS2FPGA

Port Name	Instance	Port bits	Route node type	Documentation
ARADDR		0-29	GIN	TODO
ARBURST		0-1	GIN	TODO
ARCACHE		0-3	GIN	TODO
ARID		0-11	GIN	TODO
ARLEN		0-3	GIN	TODO
ARLOCK		0-1	GIN	TODO
ARPROT		0-2	GIN	TODO
ARREADY			GOUT	TODO
ARSIZE		0-2	GIN	TODO
ARVALID			GIN	TODO
AWADDR		0-29	GIN	TODO
AWBURST		0-1	GIN	TODO
AWCACHE		0-3	GIN	TODO
AWID		0-11	GIN	TODO
AWLEN		0-3	GIN	TODO
AWLOCK		0-1	GIN	TODO
AWPROT		0-2	GIN	TODO
AWREADY			GOUT	TODO
AWSIZE		0-2	GIN	TODO
AWVALID			GIN	TODO
BID		0-11	GOUT	TODO
BREADY			GIN	TODO
BRESP		0-1	GOUT	TODO
BVALID			GOUT	TODO
CLK			DCMUX	TODO
PORT_SIZE_CONFIG		0-1	GOUT	TODO
RDATA		0-127	GOUT	TODO
RID		0-11	GOUT	TODO
RLAST			GOUT	TODO
RREADY			GIN	TODO
RRESP		0-1	GOUT	TODO
RVALID			GOUT	TODO
WDATA		0-127	GIN	TODO
WID		0-11	GIN	TODO
WLAST			GIN	TODO
WREADY			GOUT	TODO
WSTRB		0-15	GIN	TODO
WVALID			GIN	TODO

HPS_HPS2FPGA_LIGHT_WEIGHT

Port Name	Instance	Port bits	Route node type	Documentation
ARADDR		0-20	GIN	TODO
ARBURST		0-1	GIN	TODO
ARCACHE		0-3	GIN	TODO
ARID		0-11	GIN	TODO
ARLEN		0-3	GIN	TODO
ARLOCK		0-1	GIN	TODO
ARPROT		0-2	GIN	TODO
ARREADY			GOUT	TODO
ARSIZE		0-2	GIN	TODO
ARVALID			GIN	TODO
AWADDR		0-20	GIN	TODO
AWBURST		0-1	GIN	TODO
AWCACHE		0-3	GIN	TODO
AWID		0-11	GIN	TODO
AWLEN		0-3	GIN	TODO
AWLOCK		0-1	GIN	TODO
AWPROT		0-2	GIN	TODO
AWREADY			GOUT	TODO
AWSIZE		0-2	GIN	TODO
AWVALID			GIN	TODO
BID		0-11	GOUT	TODO
BREADY			GIN	TODO
BRESP		0-1	GOUT	TODO
BVALID			GOUT	TODO
CLK			DCMUX	TODO
RDATA		0-31	GOUT	TODO
RID		0-11	GOUT	TODO
RLAST			GOUT	TODO
RREADY			GIN	TODO
RRESP		0-1	GOUT	TODO
RVALID			GOUT	TODO
WDATA		0-31	GIN	TODO
WID		0-11	GIN	TODO
WLAST			GIN	TODO
WREADY			GOUT	TODO
WSTRB		0-3	GIN	TODO
WVALID			GIN	TODO

HPS_INTERRUPTS

Port Name	Instance	Port bits	Route node type	Documentation
H2F_CAN_IRQ	0-1		GIN	TODO
H2F_CLKMGR_IRQ			GIN	TODO
H2F_CTI_IRQ_N	0-1		GIN	TODO
H2F_DMA_ABORT_IRQ			GIN	TODO
H2F_DMA_IRQ	0-7		GIN	TODO
H2F_EMAC_IRQ	0-1		GIN	TODO
H2F_FPGA_MAN_IRQ			GIN	TODO
H2F_GPIO_IRQ	0-2		GIN	TODO
H2F_I2C_EMAC_IRQ	0-1		GIN	TODO
H2F_I2C_IRQ	0-1		GIN	TODO
H2F_L4SP_IRQ	0-1		GIN	TODO
H2F_MPUWAKEUP_IRQ			GIN	TODO
H2F_NAND_IRQ			GIN	TODO
H2F_OSC_IRQ	0-1		GIN	TODO
H2F_QSPI_IRQ			GIN	TODO
H2F_SDMMC_IRQ			GIN	TODO
H2F_SPI_IRQ	0-3		GIN	TODO
H2F_UART_IRQ	0-1		GIN	TODO
H2F_USB_IRQ	0-1		GIN	TODO
H2F_WDOG_IRQ	0-1		GIN	TODO
IRQ		0-63	GOUT	TODO

HPS_JTAG

Port Name	Instance	Port bits	Route node type	Documentation
NENAB_JTAG			GIN	TODO
NTRST			GIN	TODO
TCK			GIN	TODO
TDI			GIN	TODO
TMS			GIN	TODO

HPS_LOAN_IO

Port Name	Instance	Port bits	Route node type	Documentation
INPUT_ONLY		0-13	GIN	TODO
LOANIO_IN		0-70	GIN	TODO
LOANIO_OE		0-70	GOUT	TODO
LOANIO_OUT		0-70	GOUT	TODO

HPS_MPU_EVENT_STANDBY

Port Name	Instance	Port bits	Route node type	Documentation
EVENTI			GOUT	TODO
EVENTO			GIN	TODO
STANDBYWFE		0-1	GIN	TODO
STANDBYWFI		0-1	GIN	TODO

HPS_MPU_GENERAL_PURPOSE

This block provides one input and one output 32 bits port directly accessible from the arm cores at 0xff706010 (arm to fpga) and 0xff706014 (fpga to arm).

Port Name	Instance	Port bits	Route node type	Documentation
GP_IN		0-31	GOUT	Port from fpga to arm
GP_OUT		0-31	GIN	Port from arm to fpga

HPS_PERIPHERAL_CAN

(2 blocks)

Port Name	Instance	Port bits	Route node type	Documentation
RXD			GOUT	TODO
TXD			GIN	TODO

HPS_PERIPHERAL_EMAC

(2 blocks)

Port Name	Instance	Port bits	Route node type	Documentation
CLK_RX_I			DCMUX	TODO
CLK_TX_I			DCMUX	TODO
GMII_MDC_O			GIN	TODO
GMII_MDI_I			GOUT	TODO
GMII_MDO_O			GIN	TODO
GMII_MDO_O_E			GIN	TODO
PHY_COL_I			GOUT	TODO
PHY_CRS_I			GOUT	TODO
PHY_RXDV_I			GOUT	TODO
PHY_RXD_I		0-7	GOUT	TODO
PHY_RXER_I			GOUT	TODO
PHY_TXD_O		0-7	GIN	TODO
PHY_TXEN_O			GIN	TODO
PHY_TXER_O			GIN	TODO
PTP_AUX_TS_TRIG_I			GOUT	TODO
PTP_PPS_O			GIN	TODO
RST_CLK_RX_N_O			GIN	TODO
RST_CLK_TX_N_O			GIN	TODO

HPS_PERIPHERAL_I2C

(4 blocks)

Port Name	Instance	Port bits	Route node type	Documentation
OUT_CLK			GIN	TODO
OUT_DATA			GIN	TODO
SCL			DCMUX	TODO
SDA			GOUT	TODO

HPS_PERIPHERAL_NAND

Port Name	Instance	Port bits	Route node type	Documentation
ADQ_IN		0-7	GOUT	TODO
ADQ_OE			GIN	TODO
ADQ_OUT		0-7	GIN	TODO
ALE			GIN	TODO
CEBAR		0-3	GIN	TODO
CLE			GIN	TODO
RDY_BUSY		0-3	GOUT	TODO
REBAR			GIN	TODO
WEBAR			GIN	TODO
WPBAR			GIN	TODO

HPS_PERIPHERAL_QSPI

Port Name	Instance	Port bits	Route node type	Documentation
MI	0-3		GOUT	TODO
MO	0-3		GIN	TODO
N_MO_EN		0-3	GIN	TODO
N_SS_OUT		0-3	GIN	TODO

HPS_PERIPHERAL_SDMMC

Port Name	Instance	Port bits	Route node type	Documentation
CARD_INTN_I			GOUT	TODO
CCLK_OUT			GIN	TODO
CDN_I			GOUT	TODO
CLK_IN			GOUT	TODO
CMD_EN			GIN	TODO
CMD_I			GOUT	TODO
CMD_O			GIN	TODO
DATA_EN		0-7	GIN	TODO
DATA_I		0-7	GOUT	TODO
DATA_O		0-7	GIN	TODO
PWR_ENA_O			GIN	TODO
RSTN_O			GIN	TODO
VS_O			GIN	TODO
WP_I			GOUT	TODO

HPS_PERIPHERAL_SPI_MASTER

(2 blocks)

Port Name	Instance	Port bits	Route node type	Documentation
RXD			GOUT	TODO
SSI_OE_N			GIN	TODO
SS_IN_N			GOUT	TODO
SS_N	0-3		GIN	TODO
TXD			GIN	TODO

HPS_PERIPHERAL_SPI_SLAVE

(2 blocks)

Port Name	Instance	Port bits	Route node type	Documentation
RXD			GOUT	TODO
SCLK_IN			DCMUX	TODO
SSI_OE_N			GIN	TODO
SS_IN_N			GOUT	TODO
TXD			GIN	TODO

HPS_PERIPHERAL_UART

(2 blocks)

Port Name	Instance	Port bits	Route node type	Documentation
CTS			GOUT	TODO
DCD			GOUT	TODO
DSR			GOUT	TODO
DTR			GIN	TODO
OUT_N	0-1		GIN	TODO
RI			GOUT	TODO
RTS			GIN	TODO
RXD			GOUT	TODO
TXD			GIN	TODO

HPS_PERIPHERAL_USB

(2 blocks)

Port Name	Instance	Port bits	Route node type	Documentation
CLK			DCMUX	TODO
DATAIN		0-7	GOUT	TODO
DATAOUT		0-7	GIN	TODO
DATA_OUT_EN		0-7	GIN	TODO
DIR			GOUT	TODO
NXT			GOUT	TODO
STP			GIN	TODO

HPS_STM_EVENT

Port Name	Instance	Port bits	Route node type	Documentation
STM_EVENT		0-27	GOUT	TODO

HPS_TEST

Port Name	Instance	Port bits	Route node type	Documentation
CFG_DFX_BYPASS_ENABLE			GOUT	TODO
DFT_IN_FPGA_ATPG_EN			GOUT	TODO
DFT_IN_FPGA_AVSTCMDPORTCLK_TESTEN		0-5	GOUT	TODO
DFT_IN_FPGA_AVSTRDCLK_TESTEN		0-3	GOUT	TODO
DFT_IN_FPGA_AVSTWRCLK_TESTEN		0-3	GOUT	TODO
DFT_IN_FPGA_BISTEN			GOUT	TODO
DFT_IN_FPGA_BIST_CPU_SI			GOUT	TODO
DFT_IN_FPGA_BIST_L2_SI			GOUT	TODO
DFT_IN_FPGA_BIST_NRST			GOUT	TODO
DFT_IN_FPGA_BIST_PERI_SI	0-2		GOUT	TODO
DFT_IN_FPGA_BIST_SE			GOUT	TODO

continues on next page

Table 18 – continued from previous page

Port Name	Instance	Port bits	Route node type	Documentation
DFT_IN_FPGA_CANTESTEN	0-1		GOUT	TODO
DFT_IN_FPGA_CFGTESTEN			GOUT	TODO
DFT_IN_FPGA_CTICLK_TESTEN			GOUT	TODO
DFT_IN_FPGA_DBGATTESTEN			GOUT	TODO
DFT_IN_FPGA_DBGTESTEN			GOUT	TODO
DFT_IN_FPGA_DBGTMTESTEN			GOUT	TODO
DFT_IN_FPGA_DBGTRTESTEN			GOUT	TODO
DFT_IN_FPGA_DDR2XDQSTESTEN			GOUT	TODO
DFT_IN_FPGA_DDRDQSTESTEN			GOUT	TODO
DFT_IN_FPGA_DDRDQTESTEN			GOUT	TODO
DFT_IN_FPGA_DLLNRST			GOUT	TODO
DFT_IN_FPGA_DLLUPDWNEN			GOUT	TODO
DFT_IN_FPGA_DLLUPNDN			GOUT	TODO
DFT_IN_FPGA_DQSUPDTEN		0-4	GOUT	TODO
DFT_IN_FPGA_ECCBYP			GOUT	TODO
DFT_IN_FPGA_EMACTESTEN	0-1		GOUT	TODO
DFT_IN_FPGA_F2SAXICLK_TESTEN			GOUT	TODO
DFT_IN_FPGA_F2SPCLKDBG_TESTEN			GOUT	TODO
DFT_IN_FPGA_FMBHNIOTRI			GOUT	TODO
DFT_IN_FPGA_FMCSREN			GOUT	TODO
DFT_IN_FPGA_FMNIOTRI			GOUT	TODO
DFT_IN_FPGA_FMPLNIOTRI			GOUT	TODO
DFT_IN_FPGA_GPIODBTESTEN			GOUT	TODO
DFT_IN_FPGA_HIOCLKIN0			GOUT	TODO
DFT_IN_FPGA_HIOSCANCLK_TESTEN			GOUT	TODO
DFT_IN_FPGA_HIOSCANEN			GOUT	TODO
DFT_IN_FPGA_HIOSCANIN		0-1	GOUT	TODO
DFT_IN_FPGA_HIOSCLR			GOUT	TODO
DFT_IN_FPGA_IPSCCLK			GOUT	TODO
DFT_IN_FPGA_IPSCENABLE		0-11	GOUT	TODO
DFT_IN_FPGA_IPSCIN			GOUT	TODO
DFT_IN_FPGA_IPSCUPDATE			GOUT	TODO
DFT_IN_FPGA_L3MAINTESTEN			GOUT	TODO
DFT_IN_FPGA_L3MPTESTEN			GOUT	TODO
DFT_IN_FPGA_L3SPTESTEN			GOUT	TODO
DFT_IN_FPGA_L4MAINTESTEN			GOUT	TODO
DFT_IN_FPGA_L4MPTESTEN			GOUT	TODO
DFT_IN_FPGA_L4SPTESTEN			GOUT	TODO
DFT_IN_FPGA_LWH2FAXICLK_TESTEN			GOUT	TODO
DFT_IN_FPGA_MEM_CPU_SI			GOUT	TODO
DFT_IN_FPGA_MEM_L2_SI			GOUT	TODO
DFT_IN_FPGA_MEM_PERI_SI	0-2		GOUT	TODO
DFT_IN_FPGA_MEM_SE			GOUT	TODO
DFT_IN_FPGA_MPUL2RAMTESTEN			GOUT	TODO
DFT_IN_FPGA_MPUPERITESTEN			GOUT	TODO
DFT_IN_FPGA_MPUTESTEN			GOUT	TODO
DFT_IN_FPGA_MPU_SCAN_MODE			GOUT	TODO
DFT_IN_FPGA_MTESTEN			GOUT	TODO
DFT_IN_FPGA_NANDTESTEN			GOUT	TODO

continues on next page

Table 18 – continued from previous page

Port Name	Instance	Port bits	Route node type	Documentation
DFT_IN_FPGA_NANDXTESTEN			GOUT	TODO
DFT_IN_FPGA_OCTCLKENUSR			GOUT	TODO
DFT_IN_FPGA_OCTCLKUSR			GOUT	TODO
DFT_IN_FPGA_OCTENSERUSER			GOUT	TODO
DFT_IN_FPGA_OCTNCLRUSR			GOUT	TODO
DFT_IN_FPGA_OCTS2PLOAD			GOUT	TODO
DFT_IN_FPGA_OCTSCANCLK			GOUT	TODO
DFT_IN_FPGA_OCTSCANEN			GOUT	TODO
DFT_IN_FPGA_OCTSCANIN			GOUT	TODO
DFT_IN_FPGA_OCTSERDATA			GOUT	TODO
DFT_IN_FPGA_OSC1TESTEN			GOUT	TODO
DFT_IN_FPGA_PIPELINE_SE_ENABLE			GOUT	TODO
DFT_IN_FPGA_PLLBYPASS			GOUT	TODO
DFT_IN_FPGA_PLLBYPASS_SEL			GOUT	TODO
DFT_IN_FPGA_PLLTEST_INPUT_EN			GOUT	TODO
DFT_IN_FPGA_PLL_ADVANCE			GOUT	TODO
DFT_IN_FPGA_PLL_BG_PWRDN	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_BG_RESET	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_BWADJ		0-11	GOUT	TODO
DFT_IN_FPGA_PLL_CLKF		0-12	GOUT	TODO
DFT_IN_FPGA_PLL_CLKOD		0-8	GOUT	TODO
DFT_IN_FPGA_PLL_CLKR		0-5	GOUT	TODO
DFT_IN_FPGA_PLL_CLK_SELECT	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_ENSAT			GOUT	TODO
DFT_IN_FPGA_PLL_FASTEN			GOUT	TODO
DFT_IN_FPGA_PLL_OUTRESET	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_OUTRESETALL	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_PWRDN	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_REG_EXT_SEL			GOUT	TODO
DFT_IN_FPGA_PLL_REG_PWRDN	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_REG_RESET	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_REG_TEST_DRV			GOUT	TODO
DFT_IN_FPGA_PLL_REG_TEST_OUT			GOUT	TODO
DFT_IN_FPGA_PLL_REG_TEST_REP			GOUT	TODO
DFT_IN_FPGA_PLL_REG_TEST_SEL	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_RESET	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_STEP			GOUT	TODO
DFT_IN_FPGA_PLL_TEST	0-2		GOUT	TODO
DFT_IN_FPGA_PLL_TESTBUS_SEL		0-4	GOUT	TODO
DFT_IN_FPGA_PSTDQSENA			GOUT	TODO
DFT_IN_FPGA_QSPITESTEN			GOUT	TODO
DFT_IN_FPGA_S2FAXICLK_TESTEN			GOUT	TODO
DFT_IN_FPGA_SCANIN		0-389	GOUT	TODO
DFT_IN_FPGA_SCAN_EN			GOUT	TODO
DFT_IN_FPGA_SDMMCTESTEN			GOUT	TODO
DFT_IN_FPGA_SPIMTESTEN			GOUT	TODO
DFT_IN_FPGA_TEST_CKEN			GOUT	TODO
DFT_IN_FPGA_TEST_CLK			DCMUX	TODO
DFT_IN_FPGA_TEST_CLKOFF			GOUT	TODO

continues on next page

Table 18 – continued from previous page

Port Name	Instance	Port bits	Route node type	Documentation
DFT_IN_FPGA_TPIUTRACECLKIN_TESTEN			GOUT	TODO
DFT_IN_FPGA_USBMPTESTEN			GOUT	TODO
DFT_IN_FPGA_USBULPICLK_TESTEN		0-1	GOUT	TODO
DFT_IN_FPGA_VIOSCANCLK_TESTEN			GOUT	TODO
DFT_IN_FPGA_VIOSCANEN			GOUT	TODO
DFT_IN_FPGA_VIOSCANIN			GOUT	TODO
DFT_IN_HPS_TESTMODE_N			GOUT	TODO
DFT_OUT_FPGA_BIST_CPU_SO			GIN	TODO
DFT_OUT_FPGA_BIST_L2_SO			GIN	TODO
DFT_OUT_FPGA_BIST_PERI_SO	0-2		GIN	TODO
DFT_OUT_FPGA_DLLLOCKED			GIN	TODO
DFT_OUT_FPGA_DLLSETTING		0-6	GIN	TODO
DFT_OUT_FPGA_DLLUPDWNCORE			GIN	TODO
DFT_OUT_FPGA_HIOCDATA3IN		0-44	GIN	TODO
DFT_OUT_FPGA_HIODQSOUT		0-4	GIN	TODO
DFT_OUT_FPGA_HIODQSUNGATING		0-4	GIN	TODO
DFT_OUT_FPGA_HIOOCTRT		0-4	GIN	TODO
DFT_OUT_FPGA_HIOSCANOUT		0-1	GIN	TODO
DFT_OUT_FPGA_IPSCOUT		0-4	GIN	TODO
DFT_OUT_FPGA_MEM_CPU_SO			GIN	TODO
DFT_OUT_FPGA_MEM_L2_SO			GIN	TODO
DFT_OUT_FPGA_MEM_PERI_SO	0-2		GIN	TODO
DFT_OUT_FPGA_OCTCLKUSRDFT			GIN	TODO
DFT_OUT_FPGA_OCTCOMPOUT_RDN			GIN	TODO
DFT_OUT_FPGA_OCTCOMPOUT_RUP			GIN	TODO
DFT_OUT_FPGA_OCTSCANOUT			GIN	TODO
DFT_OUT_FPGA_OCTSERDATA			GIN	TODO
DFT_OUT_FPGA_PLL_TESTBUS_OUT		0-2	GIN	TODO
DFT_OUT_FPGA_PSTTRACKSAMPLE		0-4	GIN	TODO
DFT_OUT_FPGA_PSTVFIFO		0-4	GIN	TODO
DFT_OUT_FPGA_SCANOUT_100_126		0-26	GIN	TODO
DFT_OUT_FPGA_SCANOUT_131_250		0-119	GIN	TODO
DFT_OUT_FPGA_SCANOUT_15_83		0-68	GIN	TODO
DFT_OUT_FPGA_SCANOUT_254_264		0-10	GIN	TODO
DFT_OUT_FPGA_SCANOUT_271_389		0-118	GIN	TODO
DFT_OUT_FPGA_SCANOUT_2_3		0-1	GIN	TODO
DFT_OUT_FPGA_VIOSCANOUT			GIN	TODO
DFX_IN_FPGA_T2_CLK			GOUT	TODO
DFX_IN_FPGA_T2_DATAIN			GOUT	TODO
DFX_IN_FPGA_T2_SCAN_EN_N			GOUT	TODO
DFX_OUT_FPGA_DATA		0-17	GIN	TODO
DFX_OUT_FPGA_DCLK			GIN	TODO
DFX_OUT_FPGA_OSC1_CLK			GIN	TODO
DFX_OUT_FPGA_PR_REQUEST			GIN	TODO
DFX_OUT_FPGA_S2F_DATA		0-31	GIN	TODO
DFX_OUT_FPGA_SDRAM_OBSERVE		0-4	GIN	TODO
DFX_OUT_FPGA_T2_DATAOUT			GIN	TODO
DFX_SCAN_CLK			GOUT	TODO
DFX_SCAN_DIN			GOUT	TODO

continues on next page

Table 18 – continued from previous page

Port Name	Instance	Port bits	Route node type	Documentation
DFX_SCAN_DOUT			GIN	TODO
DFX_SCAN_EN			GOUT	TODO
DFX_SCAN_LOAD			GOUT	TODO
F2S_CTRL			GOUT	TODO
F2S_JTAG_ENABLE_CORE			GOUT	TODO

HPS_TPIU_TRACE

Port Name	Instance	Port bits	Route node type	Documentation
TRACECLKIN			DCMUX	TODO
TRACECLK_CTL			GOUT	TODO
TRACE_DATA		0-31	GIN	TODO

2.4 Options

Name	Type	Values	Default	Documentation
ALLOW_DEVICE_WIDE_OUTPUT_ENABLE_DIS	Bool	t/f	f	TODO
COMPRESSION_DIS	Bool	t/f	f	TODO
CRC_DIVIDE_ORDER	Num	• 0-8	0	TODO
CRC_ERROR_DETECTION_EN	Bool	t/f	f	TODO
CVPCIE_MODE	Ram	0-3	0	TODO
CVP_CONF_DONE_EN	Bool	t/f	f	TODO
DEVICE_WIDE_RESET_EN	Bool	t/f	f	TODO
DRIVE_STRENGTH	Ram	0-3	0	TODO
IDCODE	Ram	00-ff		TODO
IOCSR_READY_FROM_CSR_DONE_EN	Bool	t/f	f	TODO
JTAG_ID	Ram	32 bits		TODO
NCEO_DIS	Bool	t/f	f	TODO
OCT_DONE_DIS	Bool	t/f	f	TODO
OPT_A	Ram	0000-ffff		TODO
OPT_B	Ram	64 bits		TODO
RELEASE_CLEARS_BEFORE_TRISTATES_DIS	Bool	t/f	f	TODO
RETRY_CONFIG_ON_ERROR_EN	Bool	t/f	f	TODO
START_UP_CLOCK	Ram	00-ff	40	TODO

CHAPTER

THREE

CYCLONEV LIBRARY USAGE

3.1 Library structure

The library provides a CycloneV class in the mistral namespace. Information is provided to allow to choose a CycloneV::Model object which represents a sold FPGA variant. Then a CycloneV object can be created from it. That object stores the state of the FPGA configuration and allows to read and modify it.

All the types, enums, functions, methods, arrays etc described in the following paragraph are in the CycloneV class.

3.2 Packages

```
enum package_type_t;

struct CycloneV::package_info_t {
  int pin_count;
  char type;
  int width_in_pins;
  int height_in_pins;
  int width_in_mm;
  int height_in_mm;
};

const package_info_t package_infos[5+3+3];
```

The FPGAs are sold in 11 different packages, which are named by their type (Fineline BGA, Ultra Fineline BGA or Micro Fineline BGA) and their width in mm.

Enum	Type	Pins	Size in mm	Size in pins
PKG_F17	f	256	16x16	17x17
PKG_F23	f	484	22x22	23x23
PKG_F27	f	672	26x26	27x27
PKG_F31	f	896	30x30	31x31
PKG_F35	f	1152	34x34	35x35
PKG_U15	u	324	18x18	15x15
PKG_U19	u	484	22x22	19x19
PKG_U23	u	672	28x28	23x23
PKG_M11	m	301	21x21	11x11
PKG_M13	m	383	25x25	13x13
PKG_M15	m	484	28x28	15x15

3.3 Model information

```
enum die_type_t { E50F, GX25F, GT75F, GT150F, GT300F, SX50F, SX120F };

struct Model {
  const char *name;
  const variant_info &variant;
  package_type_t package;
  char temperature;
  char speed;
  char pcie, gxb, hmc;
  uint16_t io, gpio;
};

struct variant_info {
  const char *name;
  const die_info &die;
  uint16_t idcode;
  int alut, alm, memory, dsp, dpll, dll, hps;
};

struct die_info {
  const char *name;
  die_type_t type;
  uint8_t tile_sx, tile_sy;
  // ...
};

const Model models[];
CycloneV *get_model(std::string model_name);
```

A Model is built from a package, a variant and a temperature/speed grade. A variant selects a die and which hardware is active on it.

The Model fields are:

- name - the SKU, for instance 5CSEBA6U23I7
- variant - its associated variant_info
- package - the packaging used
- temperature - the temperature grade, 'A' for automotive (-45..125C), 'I' for industrial (-40..100C), 'C' for commercial (0..85C)
- speed - the speed grade, 6-8, smaller is faster
- pcie - number of PCIe interfaces (depends on both variant and number of available pins)
- gxb - ??? (same)
- hmc - number of Memory interfaces (same)
- io - number of i/os
- gpio - number of fpga-usable gpios

The Variant fields are:

- name - name of the variant, for instance se120b

- die - its associated die_info
- idcode - the IDCODE associated to this variant (not unique per variant at all)
- alut - number of LUTs
- alm - number of logic elements
- memory - bits of memory
- dsp - number of dsp blocks
- dpll - number of plls
- dll - number of delay-locked loops
- hps - number of arm cores

The Die usable fields are:

- name - name of the die, for instance sx120f
- type - the enum value for the die type
- tile_sx, tile_sy - size of the tile grid

The limits indicated in the variant structure may be lower than the theoretical die capabilities. We have no idea what happens if these limits are not respected.

To create a CycloneV object, the constructor requires a Model *. Either choose one from the models array, or, in the usual case of selection by sku, the CycloneV::get_model function looks it up and allocates one. The models array ends with a nullptr name pointer.

The get_model function implements the alias "ms" for the 5CSEBA6U23I7 used in the de10-nano, a.k.a MiSTer.

3.4 pos, rnode and pnode

```
using pos_t = uint16_t;          // Tile position

static constexpr uint32_t pos2x(pos_t xy);
static constexpr uint32_t pos2y(pos_t xy);
static constexpr pos_t xy2pos(uint32_t x, uint32_t y);
```

The type pos_t represents a position in the grid. xy2pos allows to create one, pos2x and pos2y extracts the coordinates.

```
using rnode_t = uint32_t;        // Route node id

enum rnode_type_t;
const char *const rnode_type_names[];
rnode_type_t rnode_type_lookup(const std::string &n) const;

constexpr rnode_t rnode(rnode_type_t type, pos_t pos, uint32_t z);
constexpr rnode_t rnode(rnode_type_t type, uint32_t x, uint32_t y, uint32_t z);
constexpr rnode_type_t rn2t(rnode_t rn);
constexpr pos_t rn2p(rnode_t rn);
constexpr uint32_t rn2x(rnode_t rn);
constexpr uint32_t rn2y(rnode_t rn);
constexpr uint32_t rn2z(rnode_t rn);
```

(continues on next page)

(continued from previous page)

```
std::string rn2s(rnode_t rn);
```

A rnode_t represents a note in the routing network. It is characterized by its type (rnode_type_t) and its coordinates (x, y for the tile, z for the instance number in the tile). Those functions allow to create one and extract the different components. rnode_types_names gives the string representation for every rnode_type_t value, and rnode_type_lookup finds the rnode_type_t for a given name. rn2s provides a string representation of the rnode (TYPE.xxx.yyy.zzzz).

The rnode_type_t value 0 is NONE, and a rnode_t of 0 is guaranteed invalid.

```
using pnode_t = uint64_t;        // Port node id

enum block_type_t;
const char *const block_type_names[];
block_type_t block_type_lookup(const std::string &n) const;

enum port_type_t;
const char *const port_type_names[];
port_type_t  port_type_lookup (const std::string &n) const;

constexpr pnode_t pnode(block_type_t bt, pos_t pos, port_type_t pt, int8_t bindex, int16_
↪t pindex);
constexpr pnode_t pnode(block_type_t bt, uint32_t x, uint32_t y, port_type_t pt, int8_t␣
↪bindex, int16_t pindex);
constexpr block_type_t pn2bt(pnode_t pn);
constexpr port_type_t  pn2pt(pnode_t pn);
constexpr pos_t        pn2p (pnode_t pn);
constexpr uint32_t     pn2x (pnode_t pn);
constexpr uint32_t     pn2y (pnode_t pn);
constexpr int8_t       pn2bi(pnode_t pn);
constexpr int16_t      pn2pi(pnode_t pn);

std::string pn2s(pnode_t pn);
```

A pnode_t represents a port of a logical block. It is characterized by the block type (block_type_t), the block tile position, the block number instance (when appropriate, -1 when not), the port type (port_type_t) and the bit number in the port (when appropriate, -1 when not). pn2s provides the string representation BLOCK.xxx.yyy(.instance):PORT(.bit)

The block_type_t value 0 is BNONE, the port_type_t value 0 is PNONE, and pnode_t 0 is guaranteed invalid.

```
rnode_t pnode_to_rnode(pnode_t pn) const;
pnode_t rnode_to_pnode(rnode_t rn) const;
```

These two methods allow to find the connections between the logic block ports and the routing nodes. It is always 1:1 when there is one.

```
std::vector<pnode_t> p2p_from(pnode_t pn) const;
pnode_t p2p_to(pnode_t pn) const;
```

These two methods allow to find the direct connections between logic port nodes of different logic blocks. The connections being 1:N the p2p_from method can give multiple results while p2p_to only answers one node or the value 0.

3.5 Routing network management

```
void rnode_link(rnode_t n1, rnode_t n2);
void rnode_link(pnode_t p1, rnode_t n2);
void rnode_link(rnode_t n1, pnode_t p2);
void rnode_link(pnode_t p1, pnode_t p2);
void rnode_unlink(rnode_t n2);
void rnode_unlink(pnode_t p2);
```

The method rnode_link links two nodes together with n1 as source and n2 as destination, automatically converting from pnode_t to rnode_t when needed. rnode_unlink disconnects anything connected to the destination n2.

There are two special cases. DCMUX is a 2:1 mux which selects between a data and a clock signal and has no disconnected state. Unlinking it puts in in the default clock position. Most SCLK muxes use a 5-bit vertical configuration where up to 5 inputs can be connected and the all-off configuration is not allowed. Usually at least one input goes to vcc, but in some cases all five are used and unlinking selects the 4th input (the default in that case).

```
std::vector<std::pair<rnode_t, rnode_t>> route_all_active_links() const;
std::vector<std::pair<rnode_t, rnode_t>> route_frontier_links() const;
```

route_all_active_links gives all current active connections. route_frontier_links solves these connections to keep only the extremities, giving the inter-logic-block connections directly.

3.6 Logic block management

```
const std::vector<pos_t> &lab_get_pos()    const
[etc]

cosst std::vector<block_type_t> &pos_get_bels(pos_t pos) const
```

The numerous xxx_get_pos() methods gives the list of positions of logic blocks of a given type. The known types are lab, mlab, m10k, dsp, hps, gpio, dqs16, fpll, cmuxc, cmuxv, cmuxh, dll, hssi, cbuf, lvl, ctrl, pma3, serpar, term and hip. A vector is empty when a block type doesn't exist in the given die.

In the hps case the 37 blocks can be indexed by hps_index_t enum.

Alternatively the pos_get_bels() method gives the (possibly empty) list of logic blocks present in a given tile.

```
enum { MT_MUX, MT_NUM, MT_BOOL, MT_RAM };

enum bmux_type_t;
const char *const bmux_type_names[];
bmux_type_t bmux_type_lookup(const std::string &n) const;

struct bmux_setting_t {
  block_type_t btype;
  pos_t pos;
  bmux_type_t mux;
  int midx;
  int type;
  bool def;
  uint32_t s; // bmux_type_t, or number, or bool value, or count of bits for ram
```

(continues on next page)

(continued from previous page)

```
  std::vector<uint8_t> r;
};

int bmux_type(block_type_t btype, pos_t pos, bmux_type_t mux, int midx) const;
bool bmux_get(block_type_t btype, pos_t pos, bmux_type_t mux, int midx, bmux_setting_t &
↪s) const;
bool bmux_set(const bmux_setting_t &s);
bool bmux_m_set(block_type_t btype, pos_t pos, bmux_type_t mux, int midx, bmux_type_t s);
bool bmux_n_set(block_type_t btype, pos_t pos, bmux_type_t mux, int midx, uint32_t s);
bool bmux_b_set(block_type_t btype, pos_t pos, bmux_type_t mux, int midx, bool s);
bool bmux_r_set(block_type_t btype, pos_t pos, bmux_type_t mux, int midx, uint64_t s);
bool bmux_r_set(block_type_t btype, pos_t pos, bmux_type_t mux, int midx, const␣
↪std::vector<uint8_t> &s);

std::vector<bmux_setting_t> bmux_get() const;
```

These methods allow to manage the logic blocks muxes configurations. A mux is characterized by its block (type and position), its type (bmux_type_t) and its instance number (0 if there is only one). There are four kinds of muxes, symbolic (MT_MUX), numeric (MT_NUM), booolean (MT_BOOL) and ram (MT_RAM).

bmux_type looks up a mux and returns its MT_* type, or -1 if it doesn't exist. bmux_get reads the state of a mux and returns it in s and true when found, false otherwise. The def field indicates whether the value is the default. The bmux_set sets a mux generically, and the bmux_*_set sets it per-type.

The no-parameter bmux_get version returns the state of all muxes of the FPGA.

3.7 Inverters management

```
struct inv_setting_t {
  rnode_t node;
  bool value;
  bool def;
};

std::vector<inv_setting_t> inv_get() const;
bool inv_set(rnode_t node, bool value);
```

inv_get() returns the state of the programmable inverters, and inv_set sets the state of one. The field def is currently very incorrect.

3.8 Pin/package management

```
enum pin_flags_t : uint32_t {
  PIN_IO_MASK    = 0x00000007,
  PIN_DPP        = 0x00000001, // Dedicated Programming Pin
  PIN_HSSI       = 0x00000002, // High Speed Serial Interface input
  PIN_JTAG       = 0x00000003, // JTAG
  PIN_GPIO       = 0x00000004, // General-Purpose I/O
```

(continues on next page)

(continued from previous page)

```
  PIN_HPS        = 0x00000008, // Hardware Processor System

  PIN_DIFF_MASK  = 0x00000070,
  PIN_DM         = 0x00000010,
  PIN_DQS        = 0x00000020,
  PIN_DQS_DIS    = 0x00000030,
  PIN_DQSB       = 0x00000040,
  PIN_DQSB_DIS   = 0x00000050,

  PIN_TYPE_MASK  = 0x00000f00,
  PIN_DO_NOT_USE = 0x00000100,
  PIN_GXP_RREF   = 0x00000200,
  PIN_NC         = 0x00000300,
  PIN_VCC        = 0x00000400,
  PIN_VCCL_SENSE = 0x00000500,
  PIN_VCCN       = 0x00000600,
  PIN_VCCPD      = 0x00000700,
  PIN_VREF       = 0x00000800,
  PIN_VSS        = 0x00000900,
  PIN_VSS_SENSE  = 0x00000a00,
};

struct pin_info_t {
  uint8_t x;
  uint8_t y;
  uint16_t pad;
  uint32_t flags;
  const char *name;
  const char *function;
  const char *io_block;
  double r, c, l, length;
  int delay_ps;
  int index;
};

const pin_info_t *pin_find_pos(pos_t pos, int index) const;
const pin_info_t *pin_find_pnode(pnode_t pn) const;
```

The pin_info_t structure describes a pin with:

- x, y - its coordinates in the package grid (not the fpga grid, the pins one)
- pad - either 0xffff (no associated gpio) or (index << 14) | tile_pos, where index indicates which pad of the gpio is connected to the pin
- flags - flags describing the pin function
- name - pin name, like A1
- function - pin function as text, like “GND”
- io_block - name of the I/O block for power purposes, like 9A
- r, c, l - electrical characteristics of the pin-pad connection wire
- length - length of the wire

- delay_ps - usual signal transmission delay is ps
- index - pin sub-index for hssi_input, hssi_output, dedicated programming pins and jtag

The pin_find_pos method looks up a pin from a gpio tile/index combination. The pin_find_pos method looks up a pin from a gpio or hmc pnode.

3.9 Options

```
struct opt_setting_t {
  bmux_type_t mux;
  bool def;
  int type;
  uint32_t s; // bmux_type_t, or number, or bool value, or count of bits for ram
  std::vector<uint8_t> r;
};

int opt_type(bmux_type_t mux) const;
bool opt_get(bmux_type_t mux, opt_setting_t &s) const;
bool opt_set(const opt_setting_t &s);
bool opt_m_set(bmux_type_t mux, bmux_type_t s);
bool opt_n_set(bmux_type_t mux, uint32_t s);
bool opt_b_set(bmux_type_t mux, bool s);
bool opt_r_set(bmux_type_t mux, uint64_t s);
bool opt_r_set(bmux_type_t mux, const std::vector<uint8_t> &s);

std::vector<opt_setting_t> opt_get() const;
```

The options work like the block muxes without a block, tile or instance number. They're otherwise the same.

3.10 Bitstream management

```
void clear();
void rbf_load(const void *data, uint32_t size);
void rbf_save(std::vector<uint8_t> &data);
```

The clear method returns the FPGA state to all defaults. rbf_load parses a raw bitstream file from memory and loads the state from it. rbf_save generats a rbf from the current state.

3.11 HMC bypass

```
pnode_t hmc_get_bypass(pnode_t pn) const;
```

The hmc_get_bypass method gives the associated HMC port to a given one when in bypass mode. Specifically, to find the rnode corresponding to a given GPIO port connected to the HMC in bypass mode do:

- Get the port(s) connected to the GPIO with p2p_to (when look for a GOUT) or p2p_from (when looking for a GIN). There should be only one even in the p2p_from case.
- Get the associated node when in bypass mode with hmc_get_bypass (the method is direction-independant)

- Get the associated routing node with pnode_to_rnode.

CHAPTER

FOUR

THE MISTRAL-CV COMMAND-LINE PROGRAM

The mistral-cv command line program allows for a minimal interfacing with the library. Calling it without parameters shows the possible usages.

4.1 models

```
mistral-cv models
```

Lists the known models with their SKU, IDCODE, die, variant, package, number of pins, temperature grade and speed grade.

4.2 routes

```
mistral-cv routes <model> <file.rbf>
```

Dumps the active routes in a rbf.

4.3 routes2

```
mistral-cv routes <model> <file.rbf>
```

Dumps the active routes in a rbf where a GIN/GOUT/etc does not have a port mapping associated.

4.4 cycle

```
mistral-cv cycle <model> <file.rbf> <file2.rbf>
```

Loads the rbf in file1.rbf and saves is back in file2.rbf. Useful to test if the framing/unframing of oram/pram/cram works correctly.

4.5 bels

```
mistral-cv bels <model>
```

Dumps a list of all the logic elements of a model (only depends on the die in practice).

4.6 decomp

```
mistral-cv decomp <model> <file.rbf> <file.bt>
```

Decompiles a bitstream into a compilable source. Only writes down what is identified as not being in default state.

4.7 comp

```
mistral-cv comp <file.bt> <file.rbf>
```

Compiles a source into a bitstream. The source includes the model information.

4.8 diff

```
mistral-cv diff <model> <file1.rbf> <file2.rbf>
```

Compares two rbf files and identifies the differences in terms of oram, pram and cram. Useful to list mismatches after a decomp/comp cycle.

CHAPTER

FIVE

MISTRAL CYCLONEV LIBRARY INTERNALS

5.1 Structure

A large part of the library is generated code from information in the data directory and generated compressed per-die binary data that is embedded in the library. The source code generation is currently done with python programs (tools directory) and the binary data through the routes-to-bin executable.

5.2 Routing data

The routing data is stored in bzip2-compressed text files named <die>-r.txt.bz2. Each line describes a routing mux.

A mux description looks like that:

```
H14.000.032.0003 4:0024_2832 0:GIN.000.032.0005 1:GIN.000.032.0004 2:GIN.000.032.0001␣
↪3:GIN.000.032.0000
```

That line describes the mux for the rnode H14.000.032.0003. It uses the pattern 4 as position (24, 2832) and has four inputs connected to four GIN rnodes.

The chip uses a limited number of mux types, with a specific bit pattern in the cram controlling a fixed number of inputs and of bit set/unset values selecting them. There is a total of 70 different patterns, currently only described as C++ code in cv-rpats.cc. An additional 4 are added to store the variations of pattern 6 where the default is different.

The special case of pattern 6 looks like:

```
SCLK.014.000.0025 6.3:1413_0638 0:GCLK.000.008.0009 1:RCLK.000.004.0011 4:RCLK.000.004.
↪0003
```

The ".3" indicates that the default is on slot 3, e.g. value 0x08 or pattern 70+3.

5.3 Block muxes

The lists of block muxes and options muxes are independant of the dies. They're in the block-mux.txt files. Each mux is described in these files using the following syntax:

```
g dft_mode m:3 21.42 20.40 20.43
  0 off
  1 on !
  7 dft_pprog
```

"g" indicates the subtype of mux, which is block-dependant, here "global". 'm' indicates a symbolic mux, 3 is the number of bits. It is followed by the bits coordinates, LSB first. Here it's an inner block, so the coordinates are 2D. Options are also 2D, and peripheral blocks are 1D.

In such a case of symbolic mux it is followed by the indented possible values of the mux (in hex) with the exclamation point indicating the default.

A numeric mux is similar but the type is 'n' and labels on the right have to be numeric.

Boolean muxes look like this:

```
g clk0_inv                b-   6.45
```

The 'b' indicates boolean, and '-' indicates the default is false, otherwise it is '+' for true. The boolean can be multi-bits, such as in the following example. Then all bits are set or unset.

```
g pr_en              b-:2 0.61 0.67
```

Finally ram muxes look like:

```
g cvpcie_mode                      r-:2    2.21 2.22
g clkin_0_src                      r2:4  760 761 762 763
```

In the second case the '2' between r and : indicates that the default value is 2.

Instanciated muxes can take two forms. For instance in fpll muxes of subtype 'c' are instanciated on the counter number, hence have 9 values. The mux is written as:

```
c cnt_in_src                       r2:2  600 601 | 602 603 | 604 605 | 606 607 | 608 609␣
↪| 610 611 | 612 613 | 614 615 | 616 617
c dprio0_cnt_hi_div                r1:8
  * 8 9 10 11 12 13 14 15
  * 24 25 26 27 28 29 30 31
  * 40 41 42 43 44 45 46 47
  * 56 57 58 59 60 61 62 63
  * 72 73 74 75 76 77 78 79
  * 88 89 90 91 92 93 94 95
  * 104 105 106 107 108 109 110 111
  * 120 121 122 123 124 125 126 127
  * 136 137 138 139 140 141 142 143
```

Either the bits are indicated on the same line separated by '|', or they're set as one set per line start with an indented '*'.

The lab, mlab, m10k, mlab and hps_clocks target bits in the 2D cram by offsetting from a base position computed from the tile position (see the method pos2bit). opt targets bits in the oram. All the others with the exception of pma3-c target bits in the pram from a position found in <die>-pram.txt. pma3-c targets bits in the cram from the tables in pma3-cram.txt

mux_to_source.py enum <datadir> generates the file cv-bmuxtypes.ipp while mux_to_source.py mux <datadir> generates the file cv-bmux-data.cc. mkmux.sh does both calls.

5.4 Logic blocks

Blocks come from two sources, the files <die>-pram.txt indicates all the peripheral blocks with their pram address. The files <die>-<block>.txt where bock is cmux, ctrl, fpll, hmc, hps or iob has the information of the connections between the blocks and neighbouring blocks and the routing grid.

blocks_to_source.py generates the cvd-<die>-blk.cc file for a given die, abd mkblocks.sh calls it for every die.

5.5 Inverters

The list of inverters, their cram position and their default value (always 0 at this point) is in <die>-inv.txt. inv_to_source.py/mkinv.sh takes care of generating the cvd-<die>-inv.cc files.

5.6 Forced-1 bits

Five of the seven dies seem to have bits always set to 1. They are listed in the files <die>-1.txt. blocks_to_source.py takes care of it.

5.7 Packages

The file <die>-pkg.txt lists the packages and the pins of each package for each die. pkg_to_source.py/mkpkg.sh take cares of generating the cvd-<die>-pkg.cc files.

5.8 Models

models.txt includes all the information on variants and models. The cv-models.cc file is generated by models_to_source.py called by mkmodels.sh.

5.9 Binary data

5.9.1 Generation and embedding

The binary blocks are accessible as individual files as <chip>-r.bin in the libmistral build subdirectory. They're embedded into object files and linked in the library where they're accessed through symbols _binary_<chip>_r_bin_start and _binary_<chip>_r_bin_end.

The .bin files are generated with the routes-to-bin executable:

```
routes-to-bin mistral/data <chip> build/libmistral
```

The decompressed data starts by a header and is followed by a number of data blocks.

5.9.2 Header

```
uint32_t off_rnode
uint32_t off_rnode_end
uint32_t off_rnode_hash
uint32_t size_rnode_hash
uint32_t count_rnode
```

- off_rnode: offset from the start of the data of the routing node information block
- off_rnode: offset from the start of the data of the end of the routing node information block
- off_rnode_hash: offset from the start of the data of the routing node hash block
- size_rnode_hash: number of entries in the routing node hash block
- count_rnode: number of routing nodes

5.9.3 Routing node information block

This block consists of a sequence of variable-length records, one per node. The non-variable part is in the structure rnode_base.

```
rnode_t node
uint32_t pattern
uint32_t fw_pos
rnode_t sources[]
```

- node: id of the routing node
- pattern: pattern number of the mux
- fw_pos: position of the mux in the firmware as x + y*width
- sources[]: array of sources, size = rmux_patterns[pattern].span

The position of the end of the block is available in the global header to know when to stop when scanning. The class method rnode_next allows to go from one rnode_base to the next. The class method rnode_sources provides a pointer to the start of the sources array from the rnode_base object.

5.9.4 Routing node hash

The block is an array of rnode_hash_entry structures.

```
uint32_t rnode_offset
uint32_t next
```

- rnode_offset: offset in bytes of the rnode_base from the start of the routing node information block. 0xffffffff if not present
- next: index of the next entry for the same hash index in the routing node hash array, 0 if none

The initial hash index for a given rnode id is id % size_rnode_hash. The size is hand-tuned to have a maximum depth of 3 and have less than 10k nodes with depth>1.

The method rnode_lookup does the hash lookup and provides a pointer to the rnode_base if the node exists.