CCDU, ClkDist

Integration Guide

IP Rev. 0.1

2014ww46

Intel Restricted Secret

Copyright © 2014, Intel Corporation. All rights reserved.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

\* Other names and brands may be claimed as the property of others.

This document contains information on products in the design phase of development.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT, OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED OR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked “reserved” or “undefined.” Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your Intel account manager or distributor to obtain the latest specifications and before placing your product order.

Copies of documents that have an order number and are referenced in this document or in other Intel literature can be obtained from your Intel account manager or distributor.

Contents

[1 Introduction 7](#_Toc379365542)

[1.1 Audience 7](#_Toc379365543)

[1.2 Supported Projects 7](#_Toc379365544)

[1.3 Terminology 7](#_Toc379365545)

[1.4 Related Documents 8](#_Toc379365546)

[1.5 Opens, Risks, and Assumptions 8](#_Toc379365547)

[1.6 Contact Information 8](#_Toc379365548)

[1.7 Document Revision History 8](#_Toc379365549)

[2 Quick Start 10](#_Toc379365550)

[2.1 Downloading Sub IP 10](#_Toc379365551)

[2.2 Integrity Checks for Standalone IP 10](#_Toc379365552)

[3 Overview 12](#_Toc379365553)

[3.1 IP Block Diagram 12](#_Toc379365554)

[3.2 Functional Top-Level Signals 12](#_Toc379365555)

[4 Design Information for Integration 13](#_Toc379365556)

[4.1 RTL Directory Structure 13](#_Toc379365557)

[4.2 Clock, Power and Reset Domains 13](#_Toc379365558)

[4.2.1 Clock Domain Diagram 13](#_Toc379365559)

[4.3 Embedded Building Blocks/Custom Logic 13](#_Toc379365560)

[4.4 RTL Configuration Parameters 13](#_Toc379365561)

[4.4.1 Mandatory Parameters 13](#_Toc379365562)

[4.4.2 Boundary Scan Parameters 14](#_Toc379365563)

[4.4.3 Test Data Register Parameters 14](#_Toc379365564)

[4.5 Testbench Parameters 14](#_Toc379365565)

[4.6 IP Straps 14](#_Toc379365566)

[4.7 Fuses 14](#_Toc379365567)

[4.8 Power Information 14](#_Toc379365568)

[4.8.1 Power Supply 14](#_Toc379365569)

[4.8.2 Bumps and Their Power Domains 15](#_Toc379365570)

[4.9 Power-up Requirements 15](#_Toc379365571)

[4.10 Macros used by IP 15](#_Toc379365572)

[4.11 Other Design Considerations 15](#_Toc379365573)

[4.12 DFx Considerations 15](#_Toc379365574)

[4.12.1 DFx Top-Level Signals 15](#_Toc379365575)

[4.12.2 DFx Clock Definition 15](#_Toc379365576)

[4.12.3 Clock Crossings 15](#_Toc379365577)

[4.12.4 VISA 15](#_Toc379365578)

[4.12.5 Debug Registers 16](#_Toc379365579)

[4.12.6 Scan – Clock Gating in RTL 16](#_Toc379365580)

[4.12.7 Scan – Reset Override 16](#_Toc379365581)

[4.12.8 TAP and Associated Registers 16](#_Toc379365582)

[4.13 System Startup 16](#_Toc379365583)

[4.13.1 Power-up Sequence 16](#_Toc379365584)

[4.13.2 Initialization Sequence 16](#_Toc379365585)

[4.13.3 Device Configuration 16](#_Toc379365586)

[4.13.4 Header for Windows Boot 16](#_Toc379365587)

[4.14 Security Considerations 16](#_Toc379365588)

[4.14.1 Security Threats 16](#_Toc379365589)

[4.14.2 Security Tests 16](#_Toc379365590)

[4.14.3 Interface Signals Implemented for Security 17](#_Toc379365591)

[4.15 RTL Design Libraries 17](#_Toc379365592)

[4.16 RTL Uniquification 17](#_Toc379365593)

[4.17 Emulation Support 17](#_Toc379365594)

[5 Verification Information for Integration 18](#_Toc379365595)

[5.1 IP Testbench Overview 18](#_Toc379365596)

[5.2 Reusable IP Testbench Components 18](#_Toc379365597)

[5.2.1 Test Island 18](#_Toc379365598)

[5.2.2 Collage or Sandbox Files 18](#_Toc379365599)

[5.2.3 IP Environment 18](#_Toc379365600)

[5.2.4 Sequences 19](#_Toc379365601)

[5.2.5 Miscellaneous 20](#_Toc379365602)

[5.3 IP-Level Information Required for Sequence Writing 21](#_Toc379365603)

[5.4 Environment Settings and Files 21](#_Toc379365604)

[5.4.1 Base Test 21](#_Toc379365605)

[5.4.2 Configuration Object 21](#_Toc379365606)

[5.4.3 API 21](#_Toc379365607)

[5.5 Description of Reusable Tests 21](#_Toc379365608)

[5.6 Description of Reusable Automation Scripts 21](#_Toc379365609)

[5.7 Supported Compiler Options for Simulation 21](#_Toc379365610)

[5.8 Reusable Simulation RUNMODEs 22](#_Toc379365611)

[5.9 RTL Verification Libraries 22](#_Toc379365612)

[6 Tools and Methodology for Integration 23](#_Toc379365613)

[6.1 Supported Tools 23](#_Toc379365614)

[6.2 Environment Variables 23](#_Toc379365615)

[6.3 HIP Libraries Included in Release 23](#_Toc379365616)

[6.3.1 Register Files or SRAM 23](#_Toc379365617)

[6.3.2 M-PHY and Related Libraries 24](#_Toc379365618)

[6.4 Directory Structure 24](#_Toc379365619)

[6.5 Ace 24](#_Toc379365620)

[6.6 Lintra 24](#_Toc379365621)

[6.7 Synthesis 24](#_Toc379365622)

[6.7.1 Clocks 24](#_Toc379365623)

[6.7.2 Clock Diagram 25](#_Toc379365624)

[6.7.3 Constraint Files 25](#_Toc379365625)

[6.7.4 BKMs 25](#_Toc379365626)

[6.7.5 Scan Insertion 25](#_Toc379365627)

[6.7.6 Latches 25](#_Toc379365628)

[6.8 Formal Verification 25](#_Toc379365629)

[6.9 CDC 25](#_Toc379365630)

[6.10 Scan 25](#_Toc379365631)

[7 Physical Integration 26](#_Toc379365632)

[8 Integration Test Plan 27](#_Toc379365633)

About This Template

How to Use This Template

Do not remove any headings from this document. If you do not need the headings to describe your IP, enter “Not applicable” under the heading. This lets the reader know that you did not overlook this topic.

In the main document that follows, add new headings that you need to fully describe the integration of this IP. Add them in the appropriate chapters.

Most red text in this document contains instructions for filling out the section where it appears. The tag for most of this red text is called “Gaps.” You should replace this text with the content appropriate for that section, ensuring that the text is tagged appropriately (for example, with the BodyText or List Bullet style). If a section is not relevant, do not remove it; instead just replace the “Gap” text with “Not applicable” and apply the BodyText style.

Goal of This Document

This document should contain all information an integration team would need to accomplish the task without needing to seek help from another source. Try not to refer to other documents for required information; do so only if you include specific instructions for obtaining those documents, and only if you are sure your audience has access to them. Verify all links. This should be a self-contained guide for integration.

# Introduction

## Audience

The information in this document is intended for:

* Subsystem integration teams integrating agent logic to clocking infrastructure delivered by this IP.
* Chassis integration teams integrating their IP to clocking infrastructure delivered by this IP.
* SoC integration teams integrating subsystems to clocking infrastructure delivered by this IP.

## Supported Projects

This document supports the following projects at the listed RTL maturity level. Type “NA” if this IP is not included in a specific project, or remove those project names from the table.

|  |  |
| --- | --- |
| Project Name | IP Maturity Level |
| CNX |  |
| KNH |  |
| ICX |  |
| ERL |  |
|  |  |
|  |  |
|  |  |

## Terminology

The table below defines uncommon terms used in this document.

|  |  |
| --- | --- |
| Term | Definition |
| CDU | Cluster DFx Unit |
| CCDU | Cluster clock distribution Unit |
| SSS | Scan subsystem |
| DOP | Drop of point |
|  |  |

## Related Documents

If you need more information on this IP, you may find these documents helpful.

|  |  |
| --- | --- |
| Document Title | Location |
| HAS or EAD |  |
| Product Brief |  |
| Release Notes | $IP\_ROOT/doc/10nmCLKDIST\_Release\_Notes\_0p5.docx |
| Signal List |  |
| Other |  |

## Opens, Risks, and Assumptions

|  |  |  |  |
| --- | --- | --- | --- |
| Item # | Description | Comment | Status (Open or Closed)/Date |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Contact Information

If you need additional help, use the contact information below.

|  |  |  |
| --- | --- | --- |
| Function | Name | Email |
| IP Architecture | Richard Gammack | richard.gammack@intel.com |
| IP Verification | N/A |  |
| IP Integration | Pratik Bhatt | pratik.r.bhatt@intel.com |
| Doc Template Owner | Susann Flowers | susann.flowers@intel.com |
|  |  |  |
|  |  |  |

## Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision Number | Description of Change | Date | Revised By |
| 0.1 | Update for drop D | 2014ww46 | Richard Gammack Pratik Bhatt |
| 0.2 | Update for 0.3A drop | 2015ww10 | Richard Gammack Pratik Bhatt |
| 0.3 | Removal of CCDU and clkdist modules. No Connections should be made to these blocks. Parameterized pccdu and pclkdist are new blocks to support clock distribution. | 2015ww21 | Richard Gammack Pratik Bhatt |
| 0.4 | Update for clkdist mux and repeater | 2015ww37 | Richard Gammack Pratik Bhatt |

# Quick Start

## Downloading Sub IP

Not applicable

## Integrity Checks for Standalone IP

Following are steps for running standalone integrity checks of this IP. It is assumed that the environment variable IP\_ROOT is set to the path of the IP collateral.

1. Run the environment script:

source /p/hdk/rtl/hdk.rc -cfg shdk74

setenv MODEL\_ROOT <release\_path>

1. Run Lintra:

bman –mc model\_name +s lintra

3.Compile the model:  
Clone the globalclk repo: git clone $GIT\_REPOS/gk/globalclk-srvr10nm <your repo>

bman –mc model\_name

bman –dut globalclk ( builds all models defined in globalclk)

# Overview

## IP Block Diagram

Clkdist, CCDU, and Aggregator IP’s are connected as shown below to deliver clocks to agent IP. For simplicity a minimal set of clocks are shown. Typical agent IP’s require more clocks as illustrated by more detailed diagrams in the appendix.

Blue connectivity is delivered by the Subsystem integration team.  
Red connectivity is delivered by the Chassis or SoC integration team.



Global clocks and usyncs are generated and distributed to all subsystems by the Clk Dist IP’s in CCI-CLK and the Subsystem. Subsystems receive the global clocks in CCDU IP’s which provide clock gating and scan clock muxing functions, before delivering the clocks to the agent IP’s. PMA and CDU control the CCDU. Agents use PMCI and/or clkreq/clkack signaling to the PMA to enable clock gating. The AGG IP aggregates clkreq/clkack.

The agent phy layer may contain a source PLL that generates a clock that is distributed through the agent clock dist IP in CCI-CLK.

Detailed descriptions af these interfaces are provided in the following tables.

## Functional Top-Level Signals

### CCDU

The CCDU is a parameterized module which generates a set of gridded clock drivers (DOPs) for a partition. Each clock driver DOP has a programmable divisor, allowing the DOP output (secondary domain) to be a divided down version of the primary grid input (primary domain). Dividers must be synchronized within a primary domain, so the preclk\_div\_sync input periodically resets these dividers to guarantee alignment of the secondary clocks derived from

a primary domain.

For each primary domain input, the module will also generate a free running copy of

the primary clock for use by the DFx and PMA subsystems. This module is parameterized to support any number of primary grid domain inputs (NUM\_OF\_GRID\_PRI\_CLKS) and generate any number of secondary domain

outputs (NUM\_OF\_GRID\_SCC\_CLKS). Two parameters control the association of the DOP outputs to the primary grid source and divisors:

GRID\_SCC\_PRICLK\_MATRIX - defines which primary domain clock is the source for each DOP

GRID\_SCC\_DIVISOR\_MATRIX - defines the divisor value for each DOP

Detailed information about these parameters is included in the comments below.

It is possible to define clocks in any order, however, it is advised to group entries by primary domain within the matrices, as shown in the examples below.

Scan support is included for the non-free running postdop clocks. Each secondary clock has a scan clock input which drives the DOP clock output during scan shift and slow speed capture.

parameter [NUM\_OF\_GRID\_SCC\_CLKS\*GRID\_PRICLK\_BITS-1:0] GRID\_SCC\_PRICLK\_MATRIX =

{4'd2, // dopclk[5] <- driven by prigrid[2]

4'd1, // dopclk[4] <- driven by prigrid[1]

4'd1, // dopclk[3] <- driven by prigrid[1]

4'd0, // dopclk[2] <- driven by prigrid[0]

4'd0, // dopclk[1] <- driven by prigrid[0]

4'd0}, // dopclk[0] <- driven by prigrid[0]

Matrix to set the divisor for each DOP clock from the corresponding primary grid. The layout of this sliced parameter is as follows:

parameter [NUM\_OF\_GRID\_SCC\_CLKS\*GRID\_DIVISOR\_BITS-1:0] GRID\_SCC\_DIVISOR\_MATRIX =

{4'd1, // dopclk[5] <- driven by prigrid[2] / 1

4'd2, // dopclk[4] <- driven by prigrid[1] / 2

4'd1, // dopclk[3] <- driven by prigrid[1] / 1

4'd4, // dopclk[2] <- driven by prigrid[0] / 4

4'd2, // dopclk[1] <- driven by prigrid[0] / 2

4'd1} // dopclk[0] <- driven by prigrid[0] / 1



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Facing | Src | Pin Name | Dest | Pin Name | Description |
| AGENT CLOCK INTERFACE (HIGH SPEED, SCANNED) | | | | | | |
| 1 | Fabric | <agent>\_clkdist | fdop\_preclk\_grid | PCCDU | fdop\_preclk\_grid | Input clk for scanned domain, from clkdist |
| NUM\_OF\_GRID\_SCC\_CLKS-1 |  | SSS | fdop\_scan\_clk | PCCDU | fdop\_scan\_clk | Scan clk to pccdu |
| NUM\_OF\_GRID\_SCC\_CLKS-1 |  | SSS | fscan\_dop\_clken | PCCDU | fscan\_dop\_clken | Clock enable |
| NUM\_OF\_GRID\_SCC\_CLKS-1 |  | SSS | fdop\_preclk\_div\_ | PCCDU | fdop\_preclk\_div\_sync | Sync from SSS |
| NUM\_OF\_GRID\_SCC\_CLKS-1 | Agent | PCCDU | adop\_postclk[] | Agent | <agent clk name> | Output clocks, for scanned clock domain |
| NUM\_OF\_GRID\_PRI\_CLKS-1 | Agent | PCCDU | adop\_postclk\_free | PMA, Agent | <agent clk name> | Output free running clock |
| 1 | Fabric | refclkdist | ck\_sb\_clk | CCDU | x4clk\_in | Optional input non-scanned clock |
| 1 | Fabric | refclkdist | Ck\_sb\_clk\_sync | CCDU | X4clk\_in\_sync | Optional input usync for non-scanned clock |
| 1 | Agent | CCDU | x4clk\_out | Agent | <agent clk name> | Optional output non-scanned clock |
| 1 | Agent | CCDU | x4clk\_out\_sync | Agent | <agent clk name> | Optional output usync for non-scanned clock |

### Clock Req/Ack Aggregator

Clock req/ak aggregator is used to aggregate clocks reqs from chassis IPs within agnets and sent it to local PMA unit to hadld. PMA unit process this aggregated clock reqs and sent ack back to aggregator which further passed down to requested chassis IPs in agent . Number of clock reqs/acks handled by aggregator is defined by RTL parameter CLKREQ\_CNT.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Src | Pin Name | Dest | Pin Name | Description |
| 1 | PMA | pm\_ip\_side\_rst\_b | Clkreqaggr | rst\_b | **Functional Reset #:** When asserted, reset all logic in the component. This reset is assumed to be synchronized to iclk outside of the component. |
| 1 | CCDU | x4clk\_out | Clkreqaggr | iclk | **Clock from Upstream:** Input clock from external |
| 1 | Clkreqaggr | Oclkmreq | PMA | cdu\_ctrl\_clk\_req | **Upstream Clock Request:** Clock request upstream |
| 1 | PMA | cdu\_ctrl\_clk\_ack | Clkreqaggr | iclkmack | **Upstream Clock Acknowledge:** Clock acknowledge from upstream |
| CLKREQ\_CNT-1 | Agent | Clkreq | Clkreqaggr | Clkreqaggr | **Downstream Clock Request/s:** Clock request/s from downstream (From IP/s). |
| CLKREQ\_CNT-1 | Clkreqaggr | Oclkack | Agent | clkack | **Downstream Clock Acknowledge/s:** Clock acknowledge/s to downstream (To IP/s or sub-units). |

### Refclkdist

Refclkdist distributes the reference clock from filter PLL to subsystems. It contains dividers at the endpoints to divide the x4 clock down to x1clk (refclk, 100MHz) and x3clk (133MHz).



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | Collage Interface Port | Width | Src | Pin Name | Dest | Pin Name | Description |
| 1 |  | 1 | Filter PLL | ckpll\_clkout1 | refclkdist | x12clk\_in | Input clk from filter PLL |
|  |  | 1 | Filter PLL | pll\_usync12\_h | refclkdist | x12clk\_in\_sync | usync from filter pll |
| 2 | X1CLK\_IN | 1 | refclkdist | ckpll\_ref\_clk | CCDU | x1clk\_in | x1clk output to CCDU |
|  |  | 1 | refclkdist | ckpll\_ref\_sync | CCDU | x1clk\_in\_sync | x1sync to CCDU |
|  | X3CLK\_IN | 1 | refclkdist | ckx3clk | CCDU | x3clk\_in | x3clk to CCDU |
|  |  | 1 | refclkdist | ckx3clk\_sync | CCDU | x3clk\_in\_sync | x3sync to CCDU |
|  | X4CLK\_IN | 1 | refclkdist | x4clk\_out | CCDU | x4clk\_in | x4clk to CCDU, FIVR, PMA |
|  | X4clk\_in\_sync | 1 | refclkdist | x4clk\_out\_\_sync | CCDU | x4clk\_in\_sync | x4sync to CCDU |
|  | x12clk\_out | 1 | Refclkdist | x12clk\_out | CCDU | x12clk\_in | x12 clk to CCDU |
|  | x12clk\_out\_sync | 1 | Refclkdist | X12clkout\_sync | CCDU | x12clkin\_sync | x12 clk sync to CCDU |

### Mesh\_clkdist

Mesh\_clkdist distributes the mesh clock from mesh PLL to subsystems.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | Collage Interface Port | Width | Src | Pin Name | Dest | Pin Name | Description |
| 1 |  | 1 | Mesh PLL | ckpll\_clkout1 | mesh\_clkdist | clkspine\_in | Input clk from mesh PLL |
|  |  | 1 | Mesh PLL | pll\_usync12\_h | mesh\_clkdist | pll\_sync\_in | usync from mesh pll |
| 2 |  | 1 | mesh\_clkdist | ckpredop | <agent>\_mesh\_clkdist | clkspine\_in | uclk to agent mesh distributions |
|  |  | 1 | mesh\_clkdist | pll\_sync\_out | <agent>\_mesh\_clkdist | pll\_sync\_in | Uclk usync to agent mesh distributions |

### pclkdist



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | Collage Interface Port | Width | Src | Pin Name | Dest | Pin Name | Description |
| 1 |  | 1 to NUM\_OF\_PRIM\_CLKS - 1 | Agent PLL | ckpll\_clkout | pclkdist | clk\_in | Input clk from Agent PLL |
| 2 |  | 1 to NUM\_OF\_PRIM\_CLKS - 1 | pclkdist | clk\_out | PCCDU | Clk\_outfdop\_preclk\_grid | clock to Agent CCDU |

### Psyncdist



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | Collage Interface Port | Width | Src | Pin Name | Dest | Pin Name | Description |
| 1 |  | 1 to NUM OF PRI CLKS-1 | PLL/global distribution | sync\_in | <agent>\_mesh\_clkdist | Sync\_in | Input sync from PLL/global distribution |
| 2 | dop\_preclk\_grid | 1 to NUM OF PRI CLKS-1 | pclkdist | Sync\_out | SSS/PMA/agent IP | Sync\_out | Mesh clock to SSS/IPs |
| 3 |  | 1 | Clkdist\_mus | x12clk\_in | psyncdist | X12clk\_in | x12clk from clkdist\_mux |
| 4 |  | 1 | Psyncdist | x4clk\_sync\_out | PMA | X4clk\_sync\_out | Sync aligned with x4 (400 Mhz) clk. |
| 5 |  | 1 | Psyncdist | x1clk\_sync\_out | PLL | x1clk\_sync\_out | Sync aligned with x1 (100 Mhz) clk. |
| 6 |  | 1 | Psyncdist | x3clk\_sync\_out | PLL ( MCDDR SS only) | x3clk\_sync\_out | Sync aligned with x3 (133 Mhz) clk. Used in MCDDR SS only. |

### 2.4.7 Clkdist\_mux



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Width | Src | Pin Name | Dest | Pin Name | Description |
| 1 | 1 | PLL/global distribution | clk\_in | Clkdist\_mux | clk\_in | Input clock from PLL/global distribution |
| 2 | 1 | scan | fscan\_ovrd\_clk | Clkdist\_mux | fscan\_ovrd\_clk | Scan clock input |
| 3 | 1 | scan | fscan\_ovrd\_clk\_sel | Clkdist\_mux | fscan\_ovrd\_clk\_sel | Scan clock override |
| 4 | 1 | Clkdist\_mux | clk\_out | Clkdist | Clk\_out | Output clock to clkdist |

### 2.4.7 Clkdist\_repeater



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | Collage Interface Port | Width | Src | Pin Name | Dest | Pin Name | Description |
| 1 |  | NUM\_OF\_PRIM\_CLKS - 1 | PCCDU | adop\_postclk\_free | Clkdist\_repeater | adop\_postclk\_free | free running clk |
| 2 |  | NUM\_OF\_PRIM\_CLKS – 1 | SSS/repeater | clk\_en | Clkdist\_repeater | clk\_en | Clk\_en from SSS or o/p from another repater stage |
| 3 |  | NUM\_OF\_PRIM\_CLKS – 1 | SSS/repeater | clk\_div\_sync | Clkdist\_repeater | clk\_div\_sync | Clk\_div\_sync from SSS or o/p from another repater stage |
| 4 |  | NUM\_OF\_PRIM\_CLKS – 1 | Clkdist\_repeater | clk\_en\_out | CCDU/Clkdist\_repeater | clk\_en\_out | Clk\_en from clkdist\_rpeater to CCDU |
| 5 |  | NUM\_OF\_PRIM\_CLKS – 1 | Clkdist\_repeater | clk\_div\_sync\_out | CCDU/Clkdist\_repeater | clk\_div\_sync\_out | Clk\_div\_sync\_out from clkdist\_rpeater to CCDU |

### 2.4.8 Divsync\_gen



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Width | Src | Pin Name | Dest | Pin Name | Description |
| 1 | 1 | PLL/global distribution | clk\_free\_in | Divsync\_gen | clk\_free\_in | Input free running clock from PLL/global distribution |
| 2 | 1 | EBC | Reset\_b | Divsync\_gen | Reset\_b | Ealy reset ( from EBC) |
| 3 | 1 | EBC | Clk\_en\_in | Divsync\_gen | Clk\_en\_en | Clock enable coming from EBC |
| 4 | 1 | PLL/global distribution | Usync\_in | Divsync\_gen | Usync\_in | Reference Usync to generate early Usync at o/p |
| 5 | 1 | Parameter value | MUsyncdelay | Divsync\_gen | MUsyncdelay | Tied to 1’b0 |
| 6 | 1 | Parameter value | NUsyncdelay | Divsync\_gen | NUsyncdelay | Tied to 1’b0 |
| 7 | 1 | Divsync\_gen | Clk\_en\_out | Clkdist\_repeater | Clk\_en\_out | Used to gate clocks  at o/p of CCDU. |
| 8 | 1 | Divsync\_gen | Div\_reset\_out | Clkdist\_repeater | Div\_reset\_out | Early Usync  Pulse used as reset to dividers inside CCDU. |
| 9 | 1 | Divsync\_gen | Usync\_out | IP/agent | Usync\_out | This is 50% duty cycle early Usync |

# Design Information for Integration

This chapter is primarily targeted to the IP integration team responsible for integrating this IP into an SoC.

## RTL Directory Structure

RTL follows standard directory structure.

## Clock, Power and Reset Domains

Clock IP interface’s inherent the clock, power, and reset domains from their context.

### Clock Domain Diagram

Not applicable

## Embedded Building Blocks/Custom Logic

|  |  |  |
| --- | --- | --- |
| Name | Library | Synthesis exchange? |
|  |  |  |
|  |  |  |
|  |  |  |

## RTL Configuration Parameters

The following tables list all RTL configuration parameters for this IP. If the parameter is derived, it must not be changed by the user.

#### CCDU Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Derived? | Range | Default | Descriptions (including interdependencies) |
| NUM\_OF\_PRIM\_CLKS | No | 1..N-1 | 3 | number of primary clocks in a given CCDU. [0]- Primary clock 0 [1]-  Primary clock 1 [N-1]- Primary clock N-1 |
| GRID\_DIVISOR\_BITS | No | 4 | 4 | number of bits needed to specify divisors used by DOPs |
| GRID\_SCC\_PRICLK\_MATRIX | YES | 1 to NUM\_OF\_GRID\_SCC\_CLKS\*GRID\_PRICLK\_BITS-1 | 24’h211000 | Matrix to map each DOP clock to the appropriate primary grid |
| GRID\_SCC\_DIVISOR\_MATRIX | YES | 1 to NUM\_OF\_GRID\_SCC\_CLKS\*GRID\_DIVISOR\_BITS-1 | 24’h121421 | Matrix to set the divisor for each DOP clock from the corresponding primary grid |
|  |  |  |  |  |

#### Pclkdist parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Derived? | Range | Default | Descriptions (including interdependencies) |
| NUM\_OF\_PRIM\_CLKS | NO | 0 to N | 2 | Number of primary clocks. |

#### Psyncdist parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Derived? | Range | Default | Descriptions (including interdependencies) |
| NUM\_OF\_PRIM\_CLKS | NO | 0 to N | 2 | Number of primary clocks |

#### Clkreqaggr Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Derived? | Range | Default | Descriptions (including interdependencies) |
| CLKREQ\_CNT | NO | 2 to N | 2 | **# of Downstream Requests:** Number of clock requests from downstream agents. |

#### Clkdist\_repeater Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Derived? | Range | Default | Descriptions (including interdependencies) |
| NUM\_OF\_SCC\_CLKS | NO | 1 to N | 1 | Number of scan clocks to be controlled by clk\_en and clk\_div\_sync |
| NUM\_OF\_RPTRS | NO | 1 to N | 1 | Number of repeater stages for clk\_en and clk\_div\_sync between SSS and CCDU. |

#### Divsync\_gen parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Derived? | Range | Default | Descriptions (including interdependencies) |
| INPUT\_SYNC\_GCLK\_BEFORE\_GAL\_SYNC | NO | 0 to 143 | 1 | Set the number of clock cycles after input Usync. |
| OUTPUT\_SYNC\_GCLK\_BEFORE\_X\_SYNC | NO | 0 to 143 | 0 | Set the how many clock cycles early div\_reset\_out and usync\_out  generated. |
| MAX\_RATIO\_WIDTH | NO | n/a | 12 | Set to 12. |
| RO\_RATIO\_WIDTH | NO | n/a | 1 | Not used. |

### Boundary Scan Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Derived? | Range | Default | Descriptions (including interdependencies) |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

### Test Data Register Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameter Name | Derived? | Range | Default | Descriptions (including interdependencies) |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## Testbench Parameters

The following table lists all testbench configuration parameters for this IP.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | Range | Default | Descriptions (including interdependencies) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## IP Straps

|  |  |
| --- | --- |
| Strap | Purpose |
|  |  |
|  |  |
|  |  |

## Fuses

Not applicable.

## Power Information

### Power Supply

Not applicable.

### Static Clock Gating

Not applicable.

### Power Gating

Not applicable.

### Bumps and Their Power Domains

Not applicable

## Power-up Requirements

Not applicable.

## Macros used by IP

Not applicable.

## Other Design Considerations

Not applicable

## DFx Considerations

Not applicable.

### DFx Top-Level Signals

Not applicable

### DFx Clock Definition

### Clock Crossings

Not applicable

### N/ADebug Registers

### Scan – Clock Gating in RTL

Document the override signal that makes clocks free-running in scan mode.

### Scan – Reset Override

Describe the mechanism to override reset during scan mode.

### TAP and Associated Registers

## System Startup

### Power-up Sequence

Not applicable

### Initialization Sequence

Not applicable

### Device Configuration

Not applicable

### Header for Windows Boot

Not applicable

## Security Considerations

Not applicable

### Security Threats

Not applicable.

### Security Tests

Not applicable.

### Interface Signals Implemented for Security

Not applicable.

## RTL Design Libraries

|  |  |  |
| --- | --- | --- |
| Library | Version | Special usage |
| pccdu\_clkdist\_rtl\_lib | 0.3 | N/A |
| refclkdist\_rtl\_lib | 0.3 | N/A |
| mesh\_clkdist\_rtl\_lib | 0.3 | N/A |
| Clkreqaggr\_rtl\_lib | 0.3 | N/A |
| Pclkdist\_rtl\_lib | 0.3 | N/A |
| Psyncdist\_rtl\_lib | 0.3 | N/A |
| Globalclk\_rtl\_lib | 0.5 | Includes all hdls used in globalclk |

## RTL Uniquification

To unquify ip please run following script from IP\_ROOT area

scripts/unquifyme <prefix>

## Emulation Support

Not applicable

# Verification Information for Integration

## IP Testbench Overview

## IP does not use any unit level test bench or test island.Reusable IP Testbench Components

### Test Island

Document any defines that are referenced by the test island or environment that can be overridden during integration. List files that may be needed for connecting non-standard signals and the associated defines.

Following are details of interfaces that are to be connected at the SoC level.

|  |  |  |
| --- | --- | --- |
| Signal | Connect to | Description |
|  |  |  |
|  |  |  |

### Collage or Sandbox Files

$MODEL\_ROOT/tools/collage/buld/builder.pccdu\_clkdist.tcl

$MODEL\_ROOT/tools/collage/buld/builder.pclklkdist.tcl

$MODEL\_ROOT/tools/collage/buld/builder.psyncdist.tcl

$MODEL\_ROOT/tools/collage/buld/builder.clkreqaggr.tcl

### IP Environment

Not applicable

#### Configuring the IP Environment

Not applicable

#### Saola Environment Walkthrough

Following are the components of RAL:

|  |  |
| --- | --- |
| File | Description |
| N/A | N/A |
|  |  |
|  |  |

#### Saola/RAL Components

|  |  |  |  |
| --- | --- | --- | --- |
| SAOLA components | Description | SoC recommendations | Required? |
| N/A |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| RAL components | Description | SoC recommendations | Required? |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

#### System Manager

Not applicable

### N/A. IP does not has any fuse requirements.Sequences

Sequences are located here: N/A

#### Sequence for Bringing up the IP

#### N/ABFM Sequences

|  |  |  |  |
| --- | --- | --- | --- |
| Sequence Name | Description | Parameters | Saola Phase |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

#### IOSF Primary/Sideband BFM Sequences

|  |  |  |  |
| --- | --- | --- | --- |
| Sequence Name | Description | Parameters | Saola Phase |
| N/A |  |  |  |
|  |  |  |  |
|  |  |  |  |

#### Other Reusable Sequences

|  |  |  |  |
| --- | --- | --- | --- |
| Sequence Name | Description | Parameters | Saola Phase |
| N/A |  |  |  |
|  |  |  |  |
|  |  |  |  |

#### IP Test Sequences

|  |  |  |
| --- | --- | --- |
| Test Sequence Name | Parameters | Function |
| N/A |  |  |
|  |  |  |
|  |  |  |

#### SoC Requirements for Sequence Reuse

#### Sequence File Dependencies

### Ip does not used any sequences or extended sequences.Miscellaneous

#### Using the Runtime or Post-Processing Checkers

Not applicable

#### Environment Files

Not applicable

#### Coverage

Not applicable.

## Environment Settings and Files

### Base Test

### N/AConfiguration Object

Not applicable

### API

## N/ADescription of Reusable Tests

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Name | Runcmd | Range | Transaction | Source |
| N/A |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## Description of Reusable Automation Scripts

## N/ASupported Compiler Options for Simulation

The table below summarizes the supported options.

|  |  |  |
| --- | --- | --- |
| Argument | Input | Example |
| N/A |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Reusable Simulation RUNMODEs

|  |  |
| --- | --- |
| RUNMODE | Description |
| N/A |  |
|  |  |
|  |  |

## RTL Verification Libraries

|  |  |  |
| --- | --- | --- |
| Library | Version | Special usage |
| N/A |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Tools and Methodology for Integration

## Supported Tools

The following tools are used in the integration of this IP. For versions supported by each release, see Release Notes in the "doc" directory of the release package.

* VCSMX
* OVM
* Ace
* SaolaLintra
* Design Compiler
* Conformal
* 0-In

## Environment Variables

Set the following environment variables as listed.

|  |  |
| --- | --- |
| Variable | Description |
| N/A |  |
|  |  |
|  |  |

## HIP Libraries Included in Release

|  |  |  |
| --- | --- | --- |
| Library | Version | Location |
| N/A |  |  |
|  |  |  |
|  |  |  |

### Register Files or SRAM

Not applicable

### M-PHY and Related Libraries

|  |  |  |
| --- | --- | --- |
| Library | Version | Date |
| N/A |  |  |
|  |  |  |
|  |  |  |

## Directory Structure

Doc

Cfg

Bin

Src

Tools

Target

## Ace

Paths to acerc: $MODEL\_ROOT/cfg/globalclk.acerc

Location of udf file:$MODEL\_ROOT/cfg/globalclk\_hdl.udf

Model to use when importing libraries:globalclk

Elaboration options not exported:N/A

Required content in sip\_shared\_libs:N/A

## Lintra

Lintra Version: 14.1p9\_shOpt64

Lintra location : tools/lintra/

Location of waiver files:

tools/lint/waivers/pccdu\_w.xml

tools/lintra/waivers/psyncdist\_w.xml

tools/lintra/waivers/clkreqaggr\_w.xml

tools/lintra/waivers/pclkdist\_w.xml

Location of Lintra patches & configuration: N/A

Location of Lintra report file for warnings and errors:

$MODEL\_ROOT/target/lint/<model\_name> /<model\_name>.log

## Synthesis

Not applicable

### Clocks

1. Primary Clocks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Clock name | Clock period | Clock waveform | Clock source |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1. Generated Clocks

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Clock Name | Master Clock Name | Master Clock Source | Edges | Source |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### Clock Diagram

### Constraint Files

* + 1. **Single Domain SIP, UPF exists. Do not use it in integration;**

**/tools/syn/gclk\_pccdu/inputs/gclk\_pccdu.upf**

**/tools/upf/pccdu.upf**

**/tools/syn/gclk\_psyncdist/inputs/gclk\_psyncdist.upf**

**tools/upf/psyncdist.upf**

**tools/syn/gclk\_clkreqaggr/inputs/gclk\_clkreqaggr.upf**

**tools/upf/clkreqaggr.upf**

Not applicable

### Scan Insertion

Not applicable

## Formal Verification

Not applicable

## CDC

Not applicable

# Physical Integration

This chapter is intended to capture the aspect ratio requirements and any fixed size impact, etc., of memories that will be used in the IP.  It is not intended to be “accurate” so much as an indication of what the impact and limitations might be.  As this information will be based on the current memories, it would be only as accurate as the current design.

|  |  |
| --- | --- |
| Array type and number of instances | N/A |
| Functional usage (how many bits are used) |  |
| Highest functional clock frequency |  |
| Floorplan details |  |
| Security requirements |  |
| IP power draw limitations for array testing |  |

# Integration Test Plan

Not applicable

# Appendix

## Subsystem connectivity details