|  |
| --- |
| NetSpeed Gemini & Orion 16.04 Release Notes  Version: GEMINI-16.04  April 15, 2016 |

NetSpeed Gemini & Orion 16.04 Release Notes

About This Document

This document lists the release notes for NetSpeed Gemini & Orion. Using NetSpeed NocStudio, users can define NoC architectures, describe specifications and requirements, optimize the NoC design and finally generate the NoC IP files such as RTL, testbench, synthesis scripts, NoC IP documentation etc.

Audience

This document is intended for users of NocStudio:

* NoC Designers
* NoC Architects
* SoC Architects

Prerequisite

Before proceeding, you should generally understand:

* Basics of NetSpeed Gemini IP Technology

Related Documents

The following documents can be used as a reference to this document.

* NetSpeed NocStudio User Manual

Customer Support

For technical support about this product, please contact [support@netspeedsystems.com](mailto:support@netspeedsystems.com)

For general information about NetSpeed products refer to: [www.netspeedsystems.com](http://www.netspeedsystems.com)

**Contents**

[About This Document 2](#_Toc449893366)

[Audience 2](#_Toc449893367)

[Prerequisite 2](#_Toc449893368)

[Related Documents 2](#_Toc449893369)

[Customer Support 2](#_Toc449893370)

[1 Deliverables 6](#_Toc449893371)

[2 Installation 7](#_Toc449893372)

[3 Feature Updates 8](#_Toc449893373)

[3.1 Fine-Grain Clock Gating improvments 8](#_Toc449893374)

[3.2 Coarse-Grain Clock Gating for Gemini 8](#_Toc449893375)

[3.3 16 Layer NOC support 8](#_Toc449893377)

[3.4 Shared Interface Master Bridge 8](#_Toc449893380)

[3.5 Shared Interface Bridge and Port Checking 9](#_Toc449893381)

[3.6 Programmable Relocation Registers And Hash Functions 9](#_Toc449893382)

[3.7 Combined Relocation and Hashing Functions 10](#_Toc449893383)

[3.8 Reset Bypass for DFT 10](#_Toc449893384)

[3.9 Priority Address Map 10](#_Toc449893389)

[3.10 Reg Bus Master Guarantee of Forward Progress 10](#_Toc449893391)

[3.11 AHB Slave Byte Enable Support 11](#_Toc449893392)

[3.12 AHB-Lite Slave Area Reduction 11](#_Toc449893396)

[3.13 APB Master Interface for Reg Bus 11](#_Toc449893398)

[3.14 Multi-Clock Reg Bus 11](#_Toc449893401)

[3.15 Separate Interrupt Pins at the NOC Level Option 11](#_Toc449893403)

[3.16 Centralized Clock Crossing on the Links 12](#_Toc449893408)

[3.17 Adding Pipeline Registers into the DEF file 13](#_Toc449893411)

[3.18 Variable Size Grid Support 13](#_Toc449893413)

[3.19 Dynamic Priority Support for Isochronous Traffic 13](#_Toc449893416)

[3.20 Multi-Voltage Low Power Support 13](#_Toc449893419)

[3.21 Exposing Virtual Channels at the Slave Brige 13](#_Toc449893420)

[3.22 RTL Grouping for High Fanout Nets and Pipeline Registers 14](#_Toc449893425)

[3.23 Original ID Propagation 14](#_Toc449893432)

[3.24 CR/CD to R Latency Improvement 14](#_Toc449893433)

[3.25 Speculative Fetch Control for ACE-Lite Agents 14](#_Toc449893434)

[3.26 Multiple Split Sizes for Gemini 15](#_Toc449893435)

[3.27 Early Write Response at CCC 15](#_Toc449893436)

[3.28 CCC Provides Read Response without Waiting for Memory 15](#_Toc449893437)

[3.29 LLC Scratchpad RAM mode 15](#_Toc449893438)

[3.30 LLC Way Allocation Controls 16](#_Toc449893439)

[3.31 CCC Directory now supports Parity 16](#_Toc449893440)

[3.32 Coherency Connect/Disconnect 16](#_Toc449893441)

[3.33 Hashing or Slicing for CCC, LLC, CACHE and SLV groups 17](#_Toc449893442)

[3.34 IMG2 Sideband Option 17](#_Toc449893443)

[4 EDA Tool Compatibility 18](#_Toc449893444)

[5 Protocol & HW Restrictions 19](#_Toc449893445)

[6 Errata 20](#_Toc449893446)

[6.1 Orion Low Power Support 20](#_Toc449893447)

[6.2 Gemini Low Power Support 20](#_Toc449893453)

[6.3 Router Checker 21](#_Toc449893454)

[6.4 AHB 21](#_Toc449893455)

[6.5 Priority Address Map 21](#_Toc449893465)

[6.6 Spyglass 21](#_Toc449893472)

[6.7 User Reg Bus Limitations 21](#_Toc449893473)

[7 Changes to Commands and Properties 22](#_Toc449893474)

[7.1 Command Changes 22](#_Toc449893475)

[7.2 Default Property Changes 23](#_Toc449893476)

[7.3 Mesh Property Changes 23](#_Toc449893477)

[7.4 Bridge Property Changes 24](#_Toc449893478)

[7.5 Host Proprerty Changes 24](#_Toc449893479)

[7.6 Link Property Changes 24](#_Toc449893490)

[7.7 Router Property Changes 25](#_Toc449893491)

# Deliverables

* NetSpeed NocStudio Package, N7 version supporting 16 layers and 256 bridges.
* NocStudio executable with interactive GUI.
* Verification checkers to be used in the DV environment.
* Sanity Test Bench.
* Documentation
  1. NocStudio User Manual: The User Guide describes how to set up a system using NocStudio and how to use it to generate NetSpeed IP.
  2. IP Integration Spec: The Integration Manual describes how to integrate a configured network into a larger subsystem.
  3. Technical Reference Manual: The Technical Reference Manual describes how the functionality of the various NoC elements, the features and functions available, and how to dynamically change the functions using the programmers mode.

# Installation

* NocStudio uses FlexLM based licensing.
  + Copy over the license file emailed separately into a folder, and point LM\_LICENSE\_FILE environment variable to this license file before launching NocStudio.
  + NOTE: When untarring Linux files, ensure it is done on a Linux machine. Untarring Linux files on a Windows machine causes problems with symbolic links.
  + The executable requires Linux CentOS 5.5 or higher.
* The release makes use of Qt libraries covered under LGPL:
  + <http://qt-project.org/downloads>

# Feature Updates

## Fine-Grain Clock Gating improvments

Support for fine grained clock gating has been enaled for pipeline registers. Since the valid signals need to control the clock-gater, this will consume additional time in the cycle for the fanout and gating. The distance that can be travelled in a cycle will be reduced accordingly. Consequently, there is a potential tradeoff between power and latency. A new link property is created to enable this fine-grain clock gating support. The default property is set to enable this fine-grain clock-gating. This can be modified by changing the prop\_default:

prop\_default enable\_fine\_grain\_clock\_gating no

Alternatively, each link can be controlled individually using the link property, such as:

link\_prop R0/18E.in enable\_fine\_grain\_clock\_gating no

## Coarse-Grain Clock Gating for Gemini

Gemini IP components (CCC, DVM, IOCB, and LLC) all support coarse-grain clock gating now. These components look for idle conditions and turn of the clock for the entire component. New requests may be delayed a cycle while the clock turns back on. These modules are controlled by hysteresis counters just like the other NoC components. The hysteresis count can be controlled per agent.

## 16 Layer NOC support

Maximum number of layers supported by NocStudio has been increased from 8 to 16 layers. This will enable even more sophistated and complex designs to be designed through NocStudio.

## Shared Interface Master Bridge

Support for a new bridge type called shared interface master bridge (SIB) has been added. Using this bridge, up to 4 AMBA hosts can share a master bridge on the NoC. A SIB contains a full master bridge and a port aggregator.

Here are the requirements for using the SIB:

* Single clock domain for all hosts
* Support for AXI4 and AXI3 protocols
* Maximum of 4 hosts aggregated
* Maximum of 16 ports aggregated

## Shared Interface Bridge and Port Checking

The Shared Interface Bridge enables port-checking capability. Port-checking is used with two equivalent masters that run in lock-step. Port-checking compares all traffic going in an out ports for equivalence. When the two masters make a request, only one is treated as real within the NoC. When the response is returned, it is returned to both. If they ever make requests or transmit data that is not equivalent and on the same cycle, the Shared Interface Bridge will report an error.

The command to enable port-checking is:

set\_sib\_port\_check\_pairs -name <name> -pairs <m0>:<m1> [<m2>:<m3>]+

## Programmable Relocation Registers And Hash Functions

The support to program the relocation registers and hash functions is now available. The hash functions can be made programmable using the add\_hash\_function command.

For reloc registers, they can be enabled as follows.

In the add\_range command, when –reloc flag is used, an address offset is set to determine how the address will be presented to the slave. To make that Slave Address programmable, the add\_range command now has a prog\_reloc\_slv flag that can be set to 0 or 1.

In the add\_range\_to\_master command, the –base flag is used to indicate that the Master Address is different from the System Address, then reloc\_sys register will inherit the programming from the original add\_range programmability option. So if the range was created as programmable, and an add\_range\_to\_master is used, the reloc\_sys will be a programmable register. In other words, since the add\_range specified that the System Sddress was programmable, the reloc\_sys, if used, will be programmable.

More details about the usage of these feature can be found in the NetSpeed Gemini Technical Reference Manual.

## 

All hashing functions are performed on the System Address, and not the Master Address or Slave Address. This enables a consistent hashing for each address range.

## Reset Bypass for DFT

The ability to bypass reset for DFT is now available. This is needed for testability functionality. New inputs to the interconnect are provided to control the reset bypass.

## Priority Address Map

The support for priority address map is now available. In the add\_range command, a new argument called “background” can be used to create a background/default range. Requests first check the normal foreground ranges to determine if there is an address match. If they miss, they will look up the background ranges and can be sent there.

One use case would be to create a large background range that goes to memory, with smaller ranges that carve out smaller portions of that address range for other devices. A 4MB background address range with a 4KB carveout would only need two address ranges. If these were implemented as normal ranges, requiring power-of-2 sizes, it would require many ranges to fill in all of the gaps created by the 4K region.

This feature must be used with care. If some agents don’t have access to one of the foreground ranges, they will end up hitting on the background range and will send there requests to that slave, instead of getting a decode error.

More details about the usage of these feature can be found in the NetSpeed Gemini Technical Reference Manual.

## Reg Bus Master Guarantee of Forward Progress

The Reg Bus Master now guarantees forward progress to the dedicated port when the tunnel is used. If the main NoC layer goes down, it can potentially prevent responses being returned from the tunnel. In prior releases, this backpressure could affect the dedicated port. In this release, dedicated resources enable the dedicated port to make forward progress even if the tunnel is locked.

## AHB Slave Byte Enable Support

The AHB Slave byte enable support is now available. This provides the ability to handle packets where not all bytes are valid. AHB protocol does not have write strobe (byte enable).  Therefore sparse writes need to be conveyed using HADDR and HSIZE.

## AHB-Lite Slave Area Reduction

The AHB-lite slave bridge has a new property that disabled write bursts and sends writes as a sequence of single-beat writes. This eliminates a significant data structure within the AHB-lite bridge, reducing area substantially. This reduction is enabled by default, but can be modified with the following NocStudio property:

bridge\_prop ahbls\_no\_write\_bursts\_enable yes/no

## APB Master Interface for Reg Bus

The APB Master Interface for reg bus is now available. When RegBus tunnel is used, the additional port can be configured to use APB. This allows an APB connection to the regbus layer for debug, as an example. The NoC registers have been modified to enable 32-bit access so the APB port can be used to access all registers.

The NocStudio property to enable this feature is:

host\_prop tunnel\_slv1\_apb yes

## Multi-Clock Reg Bus

The Multi-Clock reg bus is now available. The NocStudio property to enable this feature is:

prop\_default use\_dedicated\_regbus\_clk no

With this setting, in mapping NocStudio will use the node clock domain for regbus routers and ring master (rbs) instances. Just as with other layers, the clock domain assignments for regbus routers may be further tuned with router\_prop clock\_domain commands.

## Separate Interrupt Pins at the NOC Level Option

The ability to have separate interrupt pins at the NOC level is now available.

This gives 2 options for interrupt signal pin aggregation:

* All interrupt signals are OR'd together into a single pin at the ns\_soc\_ip interface.
* All interrupt signals are brought directly out to pins on ns\_soc\_ip from each NoC element source.

The interrupt aggregation can be controlled by the following NocStudio Mesh Prop:

mesh\_prop single\_interrupt\_for\_noc yes/no

The default is set to yes, which will combine the interrupts into a single signal.

## Centralized Clock Crossing on the Links

Support for clock crossings on Noc links is now available. Previously all clock-domain crossings were done in noc bridges or routers. Now, for cases where it is desirable to have a clock-domain crossing on a noc link, either between two routers or between a bridge and a router, NocStudio will enable you to instantiate an asynchronous fifo on a link at a position you choose, and the clock crossing will occur there. This might be helpful, for instance, at the boundary between layout partitions, where the clocks must be kept internal to their respective partitions.

NocStudio provides two ways to instantiate clock crossings on links. The first is using a link\_prop:

link\_prop <link\_name> domain\_crosser\_pos <position>

For example, “link\_prop R0.7S domain\_crosser\_pos 7” will place an asynchronous clock cross fifo on link R0.7S at position 7. The second way to create asynchronous crossings links is to use the following NocStudio command:

create\_async\_crossers\_between\_groups

This command will create asynchronous link clock-domain crossing fifos on all links that go between rtl groups defined in NocStudio.

When either method is used, NocStudio will auto-configure the pre- and post-domain crossing pipeline depths and credits.

## Adding Pipeline Registers into the DEF file

The DEF file now includes recommended placement information for pipeline registers.

## Variable Size Grid Support

The variable size grid support is now available. You can define the dimension of every column and row with different values. This gives the ability to model the floorplan more accurately. New commands to enable this feature are:

set\_row\_height <Row no> <value>

set\_column\_width <Column no> <value>

## Dynamic Priority Support for Isochronous Traffic

The dynamic priority support for isocronous traffic is now available. This provides the capability for the priority to be dynamically changed for real time traffic such as audio or video traffic in order to meet the real time requirements.

Dynamic priority allows traffic classes to be specified with two priority levels. A side-band input will be created for each traffic class with alternative priority levels. The input will change the behavior of all bridges and routers to use the alternative priority. This will affect the behavior of all outstanding and new requests in that traffic class.

One intended use of this feature is to support isochronous traffic classes, such as display traffic. Display traffic would be given its own traffic class and two priorities. It should be given a low priority as its default value. When the display engine starts falling behind in its data prefetching, it can change the dynamic priority select control to switch to the alternative priority, which should be set to a high priority. This will allow the traffic to start as low priority but switch to high priority as the real-time requirement approaches.

## Multi-Voltage Low Power Support

The Multi-Voltage Low Power Support is now available, allowing a voltage crossing to be present between the host and the bridges of the NoC.

## Exposing Virtual Channels at the Slave Brige

The support for Exposing Virtual Channels at the Slave Bridge is now available.

The number of virtual interfaces on a slave can be controlled by bridge properties axi4s\_ar\_num\_virtual\_ifces (to control the number of virtual channels on the AR interface) and axi4s\_aw\_num\_virtual\_ifces (to control the number of virtual channels on the AWW interface).

Here are the new properties to enable this feature:

bridge\_prop axi4\_ar\_num\_virtual\_ifces host/bridge <num>

bridge\_prop axi4\_aw\_num\_virtual\_ifces host/bridge <num>

## 

## RTL Grouping for High Fanout Nets and Pipeline Registers

The support for RTL grouping for high fanout nets and pipeline registers is now available.

## 

## Original ID Propagation

The AxID present to a slave interface will consist of a transaction ID plus the ID of the initiating master. However, when multi-hop traffic exists, such as a coherent read that goes to CCC, then LLC, and then to memory, the Master ID that arrives at the destination is the ID of the last master. In this case, the LLC.

It may be useful to know where the request originated. NocStudio allows a port to add an additional bus to the AR and AW channels, called the Original ID bus. This bus is separate from the ARID or AWID bus, and provides no ordering restrictions to the agent. This is just an information bus.

To enable the Original ID additional to the AR and AW bus, a slave may set the bridge properties as follows:

bridge\_prop slave/port axi4\_ar\_org\_id\_enb yes/no

bridge\_prop slave/port axi4\_aw\_org\_id\_enb yes/no

## CR/CD to R Latency Improvement

CCC has reduced latency between a snoop response to the read response waiting for the snoop.

## Speculative Fetch Control for ACE-Lite Agents

Speculative fetch is the ability of the CCC to issue a read to LLC/memory before the directory has been accessed or the snoop responses returned. In the prior release, the speculative fetch programmability was limited to ACE agents and the IOCB. All agents talking through IOCB (ACE-lite, ACE-lite+DVM, and AXI->ACE-lite converted) used the IOCB property. Now each agent can individually control whether to perform speculative fetch. The default value can be specified in NocStudio using a bridge property. This value can be reprogrammed.

bridge\_prop master/port cc\_axi4m\_speculative\_fetch yes/no

## Multiple Split Sizes for Gemini

The split sizes controlled at each master was forced to be 64B when the bridge accessed one of the Gemini IPs (CCC, IOCB, LLC). This is no longer required. The bridge determines if a request is to one of these destinations and will force a split of the request to 64B boundary if needed. But if the request targets another IP, the programmed split size will be used to determine whether to split and on what boundary.

## Early Write Response at CCC

CCC now has the ability to provide an early write response instead of waiting for the write to propagate to LLC or memory.

## CCC Provides Read Response without Waiting for Memory

If the CCC issues a speculative fetch for a cache line, the snoop response may return with sufficient data to process the request immediately. If a read to memory was issued speculatively, the CCC used to wait for that to complete as well before satisfying the read. It will now return immediately.

## LLC Scratchpad RAM mode

The LLC can now be configured or programmed to convert parts of the cache into a scratchpad RAM. This can be controlled on a way-group basis (4 ways). A range must be created that is the size of the LLC rounded up to the nearest power of 2. Then two properties can be set to determine the reset condtion of the cache:

host\_prop llc llc\_waygroup\_ram\_mode\_enable bit\_vector

host\_prop llc llc\_waygroup\_ram\_mode\_secure bit\_vector

The first property determines which of the waygroups is enabled by default as RAM. The second property determines the security requirements for each waygroup, since the scratchpad RAM acts as a backing store. A zero indicates non-secure access. A one indicates secure access.

## LLC Way Allocation Controls

LLC now support way allocation controls. A new property called LLC Allocation Class is added to each master bridge that can talk to the LLC. This can be set to one of 8 allocation classes. The LLC has a control register for each allocation class to determine which of the waygroups a line can allocation into by that agent. This can allow soft-partitioning of associativity in the LLC. The LLC Allocation Class is controlled by a bridge property.

bridge\_prop master/port llc\_allocation\_class #class\_num

## CCC Directory now supports Parity

CCC directory already had the ability to be protected with ECC. It now includes the option to have parity protection as a lower-area solution. This can be enabled using the following CCC host property:

host\_prop ccc cc\_directory\_parity\_enabled yes/no

## Coherency Connect/Disconnect

Gemini now supports a coherency connect/disconnect protocol for ACE and ACE-lite+DVM agents. When connected to coherency, they are able to issue coherent or DVM requests and receive snoops. When disconnected, no snoops will be sent to them and they can be safely reset or powered off. The connect/disconnect protocol can either be controlled by I/O pins to the agent (called SYSCOREQ and SYSCOACK), or they can be controlled through registers. If controlled through registers, an agent can be configured to be disconnected or connected after reset.

To control the connect default or whether to use pins or registers, use the following bridge property:

bridge\_prop master/port cc\_coherency\_connect <io\_pins/register\_connected/register\_disconnected>

## Hashing or Slicing for CCC, LLC, CACHE and SLV groups

The add\_ccc\_group, add\_llc\_group,add\_cache\_group, and add\_slv\_group commands now support either address slicing or address hashing to determine which of the agents in the group should receive a request.

## 

# EDA Tool Compatibility

* Cadence EDA tools were used for verification and synthesis of this product.
* Compatibility testing has been done with VCS. Issues if any might be seen in the verification IP for specific configurations. The NetSpeed Orion AMBA IP Integration specification lists the various defines to be used to enable / disable Verification IP. NetSpeed support will be available to resolve any issues.

# Protocol & HW Restrictions

* Regbus master bridge implements a customized AXI protocol with 32b data width, AxLEN of 0, 1 to support 32b, 64b register accesses over a 32b down-sizeable NoC layer interface.
* AXI error handling has the following limitation
  + For reads responses for narrow transfers, if RRESP for the same response has mixture of different error types per beat, end-to-end checker may flag a false RRESP mismatch.    This is a pathological case and not expected to be seen in normal usage.

# Errata

## Orion Low Power Support

* User regbus bridges are not currently fully supported in low power mode
* Multi-voltage: AHBLM master bridge, and IMG2bus, APB & AHB slave bridges do not implement multi-voltage support.
* For regbus, the following restrictions apply:
  + The regbus feature has been tested only with limited directed tests where the power profiles were static.
  + The regbus tunnel feature has been tested only with limited directed tests where the power profiles were static.
  + The regbus tunnel local host interface is not supported in low power configurations.
  + Rebug tunnel power\_domain assignments: NocStudio does not automatically keep power\_domain\_host of rbm/s instance in sync with rbm/s power\_domain and power\_domain\_host/power\_domain of rbm/m. User should take care to ensure all these properties are set the same for regbus tunnel deployments.
  + The host regbus feature is not supported in low power configurations.
* Bridge and host power domains: if left unassigned by the user, NocStudio may assign inappropriate power domains. Users should explicitly specify host and bridge power domains in their configurations.
* QACTIVE idle status indications may be incomplete for host and bridge power domains for AHBLM and SIB aggregated hosts and ports. It may also be incomplete for bridge and host power domains where there is a voltage crossing. However, proper fencing and draining functionality supporting safe power transitions is maintained in all cases.
* Support has not been implemented for:
  + DVFS.
  + UPF power intent format
    - Impacts LP simulation and synthesis with Synopsys tools

## Gemini Low Power Support

Gemini Low Power configurations have some restrictions. While voltage domain crossings can exist within a link or at the interface between a host and a bridge, Gemini does not currently support the voltage crossing between the host and bridge for Gemini IP (CCC, DVM, IOCB, LLC) or ACE bridges (ACE, ACE-lite, ACE-lite+DVM).

Gemini Low Power also currently expects that the coherent components are part of auto-wake power domains. If they are added to decode-error power domains, some coherent requests may fail. DVM requests, for instance, cannot handle decode errors.

Gemini Low Power support is an Alpha Release. While there are no known bugs, verification is still in progress.

## Router Checker

The router checker has been removed from this release.

## AHB

There could be a deadlock between AHB master bridge and the AHB master if master is waiting for HREADY to be asserted before removing the BUSY command. Workaround: remove BUSY as soon as the next command is available. This will be addressed in the next release.

## Priority Address Map

As noted in the Feature Updates section, the Priority Address Map has a potential issue when some agents do not have access to a slave in a foreground range. Instead of getting a decode error when they attempt to access those ranges, they can hit against the background range and send the request to that slave.

## Spyglass

Please see the Waiver document about the spylass waivers.

## User Reg Bus Limitations

For user reg bus, below are the limitations of the support:

1. Transaction size needs to equal the interface size:
   1. If you have a 32 bit rb native interface, 32 bit user reg bus accesses are supported
   2. If you have a 64 bit rb native interface, 64 bit user reg bus accesses are supported. 32 bit access are not supported.
2. Host errors returned from the rb native interface are not supported.
3. Async interface at rb native is not supported.
4. Write strobes (byte enables) are not supported. This is similar to what we support for the NOC internal registers.

# Changes to Commands and Properties

## Command Changes

|  |  |
| --- | --- |
| **Command Name** | **Comment** |
| add\_bridge | New to add bridge after add\_host command has been specified |
| add\_sib | New command to add a shared interface bridge |
| create\_async\_crossers\_between\_groups | New command to auto configure async crossing properties based on RTL grouping |
| del\_bridge | New to delete bridge after add\_host command has been specified |
| json\_dump | New command to output a JSON structure of NoC |
| list\_sibs | New command to list shared interface bridges within a design |
| list\_vifce\_aid\_entries | New command that lists the number of AID entries for each virtual interface of a given interface |
| list\_vifce\_class | New command that lists virtual interfaces associated with each class value for a given interface |
| list\_vifce\_credits | New command that lists number of credits for each virtual interface for a given interface |
| set\_auto\_pd\_prefix | New command to set prefix to automatically created power domain. |
| set\_column\_width | New command to set per column grid cell size |
| set\_row\_height | New command to set per row grid cell size |
| set\_vifce\_aid\_entries | New command to set AID entries for each virtual interface |
| set\_vifce\_class | New command to set traffic class for each virtual interface |
| set\_vifce\_credits | New command to set number of credits for each virtual interface |
| show\_layers | Command to show specified number of layers in GUI |
| add\_ccc\_group | Command syntax has been changed to accommodate hash functions |
| add\_llc\_group | Command syntax has been changed to accommodate hash functions |
| add\_cache\_group | Command syntax has been changed to accommodate hash functions |

## Default Property Changes

|  |  |  |
| --- | --- | --- |
| **Property Name** | **Default Value** | **Comment** |
| archive\_project\_dir | yes | New property to disable automatic archiving of project directories |
| axi4\_ar\_org\_id\_enb | no | New properties that indicates whether original ID is enabled for read requests from all AXI4/ACEL slave bridges |
| axi4\_aw\_org\_id\_enb | no |
| axi4m\_prog\_addrs\_range | no | Property has been deprecated |
| credit\_flop\_to\_output\_ns | 0.8 | New properties that set default values for the credit return path |
| credit\_pipeline\_depth | 0 |
| enable\_fine\_grain\_clock\_gating | yes | New property that enables fine grained clock gating withing each link in design |
| tag\_project\_name | no | This property will not be supported for Orion and Gemini designs in this release |
| use\_dedicated\_regbus\_clk | yes | New property that enables/disables the use of a dedicated regbus clock |

## Mesh Property Changes

|  |  |  |
| --- | --- | --- |
| **Property Name** | **Default Value** | **Comment** |
| def\_blockage\_top\_routing\_layer | 0 | New property that indicates the top routing layer in a def file |
| def\_scale | 1 | New property to control the scaling factor of generated box size in DEF. |
| enable\_detailed\_logging | yes | New property, when disabled, will speed up performance simulation runtime |
| enable\_sib\_autoconnect | no | New property that indicates if compatible masters can be automatically connected to a shared interface bridge |
| ipxact\_compact\_version | no | New property to skip memory map section in IPXACT generation. This property is visible only if default property ipxact\_enable is set to yes. |

## Bridge Property Changes

|  |  |
| --- | --- |
| **Property Name** | **Comment** |
| axi4\_ar\_num\_virtual\_ifces | New bridge property for supporting virtual AXI interface |
| axi4\_ar\_org\_id\_enb | New prooperty that indicates whether original ID is enabled for read requests from a bridge |
| axi4\_aw\_num\_virtual\_ifces | New bridge property for supporting virtual AXI interface |
| axi4m\_prog\_addrs\_range | This bridge property has been deprecated |
| cc\_axi4m\_speculative\_fetch | Support has been added to acel amd aceldm slaves |
| cc\_coherency\_connect | New bridge property for coherency connect signals |
| llc\_allocation\_class | New property for setting LLC allocation class |
| rw\_full\_throughput | Property has been enabled for Gemini configurations |

## Host Proprerty Changes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Property Name** | | **Comment** | | | |
| llc\_waygroup\_ram\_mode\_enable | | Bit vector that specifies which of the LLC waygroups work as a RAM | | | |

## Link Property Changes

|  |  |
| --- | --- |
| **Property Name** | **Comment** |
| credit\_flop\_to\_output\_ns | New properties to set a link’s credit return path |
| credit\_pipeline\_depth |
| enable\_fine\_grain\_clock\_gating | New property to enable / disable fine grained clock gating on a link |
| domain\_crosser\_fifo\_depth | New properties to set on a link with ILDC (In link domain crossing). Please see NocStudio help menu for usage details.  New properties to set on a link with ILDC (In link domain crossing). Please see NocStudio help menu for usage details |
| domain\_crosser\_output\_register |
| domain\_crosser\_pos |
| domain\_crosser\_credit\_output\_register |
| post\_domain\_crosser\_credit\_pipeline\_depth |
| post\_domain\_crosser\_pipeline\_depth |

## Router Property Changes

None.

2670 Seely Ave

Building 11

San Jose, CA 95134

(408) 914-6962

<http://www.netspeedsystems.com>