NetSpeed Gemini

Release Notes

Version: GEMINI-16.12

January 31, 2017

NetSpeed Gemini 16.12 Release Notes

About This Document

This document lists the release notes for NetSpeed Gemini. 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

For general information about NetSpeed products refer to: www.netspeedsystems.com

Contents

1 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
    a. NocStudio User Manual: The User Guide describes how to set up a system using NocStudio and how to use it to generate NetSpeed IP.
    b. IP Integration Spec: The Integration Manual describes how to integrate a configured network into a larger subsystem.
    c. 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 programmer's mode

2 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

3 Feature Updates: System Interconnect

3.1 PER NODE CLOCK DELAY

With today's scalable SoCs comes the challenge of physical design especially in routing and managing the clock skews across the floorplan. Balancing the clocks across a chip is not an easy task and NocStudio eases this burden by providing the ability for system architects and physical designers to assign a clock delay at each node, to adjust pipeline depths for links between different nodes. The relative difference in clock delay is used to adjust propagation delay, allowing unbalanced clock trees to be handled without clock crossers. The feature also helps in reducing the physical design iteration by allowing a way to import some of the clocking characteristics back into NocStudio and providing an RTL that is already tailored to meet those specifications.

3.1.1 GUI Enhancement

In the clock domain view, the clock delays of each node are displayed in the bottom left corner of each node. The clock delay of a node can be changed by clicking on the clock delay and entering the new value in the text box that appears. There is a check box in the top left corner of the clock domain view that allows toggling between showing and hiding the clock delays of the nodes. The value of the clock delay can also be changed from the console through the command *set_node_clock_delay*.

3.2 CONTROL OF WARNING MESSAGES

Depending on the development phase or the debug requirement users need control over the generation of warning messages, to focus on other part of the design. NocStudio has added two commands for user to choose the level at which warning message can be hidden and ignored. The *hide_warning* allows fine-grained control over whether various warning messages are printed, while the *set_debug_level* allows 'all warnings and info messages' or 'all info messages' to be not printed.

3.3 USER DEFINED COMMAND

NocStudio adds support for user define commands by allowing creation of values that can be used later, either

- as tokens in an existing command or
- as an entire command or
- sequence of commands.

This is similar to a #define to help user create commands specific to their system requirements and reusing them. Please refer of the User Manual or the NocStudio Help for details description of these commands.

3.4 DOMAIN CROSSER

It is now possible to skip the routers which have async ports, and instead have Domain Crossers be present across the async link. This is done through the prop_default *keep_routers_for_async*, which is set to "yes" by default, meaning that async routers are not skipped. When this prop is turned "no", the async routers are skipped and instead, there are Domain Crossers added on the async links.

3.5 ROUTER ARBITRATION

Within a QoS weighting round, the round-robin arbitration switches between the different ports. This distributes bandwidth within the round, and reduces the worst-case latency for a request, since it doesn't have to wait for all of the requests to be issued from the prior ports. Disabling the round-robin arbitration has the advantage that the arbitration within the QoS round will have a static prioritization, which will tend to keep accesses from the same agent selected together. This can increase memory controller efficiency by keeping accesses to the same DRAM page clustered together in the network so that they arrive together at the memory controller. Disabling round - robin will result in some agents taking longer to get their requests to memory. If QoS weighting is disabled in the NoC, the round-robin arbitration should not be disabled or starvation can occur for some agents. The QoS weighting takes effect regardless of whether the round - robin arbitration is enabled, so disabling it will not affect overall bandwidth allocation. It only affects the order of requests within a QoS round.

4 Feature Updates: Non-Coherent Components

4.1 FUNCTIONAL SAFETY

NocStudio has augmented its existing robust Functional Safety features, with additional support for parity in CSR (configuration and status registers). The parity is implemented at a byte granularity, and also provides support for interrupt generation to communicate errors. These interrupts are also maskable depending on specific system requirements to ignore some of the non-critical registers.

For more details, please refer to Chapter "Safety and Reliability" in the TRM.

4.2 CONSOLIDATED RESET

Previous releases had two sets of reset signals, the *reset_pd_n_<PD>* and the *reset_n_<PD>*, per power domain. Effort has been made to consolidate and simplify the reset structure. With this release, there is a single *reset_n_<PD>*.

4.3 EASE INTEGRATION OF POWER SUPERVISOR

NetSpeed has added multiple capabilities to ease the integration of the NetSpeed Power Supervisor logic. This includes

- creating independent RTL group *nsps_grp* within each power domain
- explicit naming of interface signals with a tag "async"
- controllability of the position of each *nsps_grp* module
- independent control of the clock for each *nsps_grp*

5 Feature Updates: Coherent Components

5.1 NOC IMPROVEMENTS

5.1.1 Interface Isolation

In previous releases, the command map is used to map the specified traffic on 3 layers (1 layer for read requests and write responses, 1 layer for read responses and write requests and 1 layer for acknowledgement messages). For Gemini, the behavior of map has been modified to use 1 layer per interface type to provide better isolation. Thus the "*map*" command will map Gemini traffic over 8 layers. The similar behavior can be achieved by using the *–isolate_interfaces* argument with *map_opt*. This will allow *map_opt* to try an iteration (in addition to those specified in the *map_opt* command) with traffic from interfaces separated on layers like *map*.

5.1.2 Bandwidth intent of *add_traffic_c*

In certain configurations, the computation of average bandwidth in *add_traffic_c* has been improved and tailored to better match the user's intent. As a consequence, some configurations may see different specified bandwidths on the sources and destinations compared to the previous releases. The specified bandwidth in all add traffic commands is applied on a per source to destination pair basis. Hence if a bandwidth of 2 GBps is specified for a traffic command with 3 sources and 4 destinations, the bandwidth seen by each master is 8 GBps and the bandwidth seen by each slave is 6 GBps. If the "*split*" keyword is specified in the add traffic command, the specified bandwidth is divided across all the source to destination pairs.

5.2 QOS CONTROL FOR DERIVATIVE TRANSACTIONS

Coherency requires extra points for checks and balances in the life of a transaction, to ensure that the coherent intent of the transaction is respected and maintained. As a result, derivative transactions are injected to send messages and or transfer data. The QoS requirements for these derivative transactions are not specified by the user, and is usually not explicitly available. Depending on the system usecase incorrect QoS values could make or break the performance of the coherent system. NocStudio adds support to allow the user to control the QoS for all the derivative coherent transaction based on the system needs. This allows a system to define varied QoS for derivative and flow through commands giving the user an intelligent way to control the sharing of available bandwidth.

5.3 DEADLOCK AVOIDANCE UNIT

With today's systems supporting different standard protocols, it is vital to make sure the interoperability and dependencies are taken care of. PCIe introduce some specific requirements that have to be designed for to avoid unforeseen dependencies with other protocols. The deadlock-avoidance unit is designed for this specific purpose and provides decoupling of protocols necessary to allow interfacing between a coherent subsystem with cores using the AMBA-ACE protocol and a PCIe root complex.

Please note that this IP block requires a special license key for access

5.4 INTEGRATED CCC

There are several advantages and motivations to having the CCC and LLC as distinct modules, so as to have better and varied controllability of their organization. There are also other systems that benefit from the fact of having them combined into one module. Specifically, in systems that have a one-to-one mapping between the CCC and the LLC. For these systems, NocStudio has added support for an Integrated CCC (iCCC) that reduces the latency for such tightly coupled configurations. Also in today's multi-level coherency requirements, there is a need to marry the CCC and LLC into one component. Please refer to the new command *add_iccc* for more details on the usage.

Please note that this IP block requires a special license key for access

5.5 CR CHANNEL ON SIDEBAND

Latency is one of key careabouts in systems, especially with the addition of coherency requirements among multiple masters. One of the key components is the snoop response latency from the masters back to the coherency controller. NocStudio provides an enhancement, by which, snoop response (CR) as a sideband field of the packet carrying snoop data CD. This reduces the latency of snoop responses from masters to coherency controller. Area reduction can also be expected for agents with large data width.

5.6 AUTOMATED CCC AND LLC SIZING

The specified CCC size values (capacity, associativity, index width, index bits, tag bits) and LLC size values (index bits, tag bits) specified before *setup_coherency* are now used in the calculation of the unspecified size values during *setup_coherency*. In previous releases, *setup_coherency* computed all the CCC and LLC size values (both previously specified and unspecified). Custom values were needed to be specified after *setup_coherency*. This is no longer required.

6 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 IP Integration specification lists the various defines to be used to enable / disable Verification IP. NetSpeed support will be available to resolve any issues.

7 Errata: System Interconnect

NTR

8 Errata: Non-Coherent Components

8.1 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.

8.2 PRIORITY ADDRESS MAP

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.

9 Errata: Coherent Components

NTR

10 Changes to Commands and Properties

10.1 COMMAND CHANGES

Command Name	Comment
defcmd	New command to define a new NocStudio command based on existing commands.
define	New command to define a constant that can be referred to in later commands.
add_dau	New command to add a deadlock avoidance unit.
show_direct_host_connection_info	New command to check if 2 bridges can be optimized out with their hosts directly connected.
set_node_clock_delay	New command to set the clock delay for each node in the mesh.
list_node_clock_delay	New command to list the clock delay for all nodes in the mesh.
assign_regbus_node_clock_domain	New command to set the clock domain that will be used by regbus elements.
reset_regbus_node_clock_domain	New command to reset the clock domain settings of all regbus elements.
set_debug_level	New command to change the level of messages that are printed.
show_noc_density	New command to print a report of the wire and flop counts.
reset_prop_defaults	New command to reset the values of default properties to their original value.
set_traffic_props	New command to modify the class, layers or VC's of existing traffic.
add_iccc	New command to add an integrated CCC and LLC block.
add_iccc_group	New command to group ICCCs.

del_iccc_group	New command to delete ICCC groups.
list_iccc_groups	New command to list ICCC groups.
set_power_domain_nsps_clk	New command to change the NSPS clock of existing power domains.
set_power_domain_nsps_pos	New command to change the NSPS position of existing power domains.
set_route_type	This command now accepts source and destination interfaces as arguments instead of source and destination bridges.

10.2 DEFAULT PROPERTY CHANGES

Property Name	Default Value	Comment
gen_dep_graph	no	New default property to enable to generation of all_dependency_graph.gv.
keep_routers_for_async	yes	New default property that can be set to no to enable the use of in-link domain crossers instead of routers for clock crossing.
between_underscore_token	<none>	New default property to inject a string between double-underscores used for RTL.
enable_round_robin	yes	New default property to enable or disable round robin arbitration between different ports of a bridge or router.
gen_all_ifce_deps	no	New default property to enable the writing of a file that contains all interface level dependencies after mapping.

10.3 MESH PROPERTY CHANGES

Property Name	Default Value	Comment

register_parity_enabled	no	New mesh property to enable the parity register.

10.4 BRIDGE PROPERTY CHANGES

Property Name	Comment
enable_round_robin	New bridge property to enable or disable round robin arbitration between different ports of the bridge.

10.5 HOST PROPERTY CHANGES

Property Name	Comment
dau_cmd_sram	New property on DAU hosts that allows command write queue to be implemented as a RAM instead of flops.
dau_data_sram	New property on DAU hosts that allows data write queue to be implemented as a RAM instead of flops.
dau_data_width	New property on DAU hosts that sets the data width of its master and slave bridges.
dau_aw_mem_in_width	New property on DAU hosts to specify the number of info bits needed as input for write commands to DAU.
dau_aw_mem_out_width	New property on DAU hosts to specify the number of info bits needed as output for write commands to DAU.
dau_w_mem_in_width	New property on DAU hosts to specify the number of info bits needed as input for write data to DAU.
dau_w_mem_out_width	New property on DAU hosts to specify the number of info bits needed as output for write data to DAU.
cc_ovrd_arqos_val	New property on CCC hosts to specify the QoS override value for read requests from the CCC.
cc_ovrd_wrqos_val	New property on CCC hosts to specify the QoS override value for write requests from the CCC.
cc_ovrd_snqos_val	New property on CCC hosts to specify the QoS override value for snoop requests from the CCC.

cc_ovrd_direvtqos_val	New property on CCC hosts to specify the QoS override value for directory evicts from the CCC.
cc_ovrd_arqos_enb	New property on CCC hosts to specify the overriding of QoS for read requests from the CCC.
cc_ovrd_wrqos_enb	New property on CCC hosts to specify the overriding of QoS for write requests from the CCC.

10.6 INTERFACE PROPERTY CHANGES

None

10.7 LINK PROPERTY CHANGES

None

10.8 ROUTER PROPERTY CHANGES

Property Name	Comment
enable_round_robin	New property that enables round robin arbitration between different ports of the router.

10.9 VC PROPERTY CHANGES

None

2870 Zanker Road,

Suite 210,

San Jose, CA 95134

(408) 617-5209

http://www.netspeedsystems.com