“SoC and Multi-Core Debug: Are Design for Debug (DFD) features that are put in reuse cores sufficient for Silicon Debug?”

Sankaran Menon
Intel Corporation

Abstract

With the advent of System-on-Chip (SoC) technology and integration of multiple cores on a single die resulting in Multi-Core chips, complexity of integrated circuits has compounded. This has resulted in additional complexity in debugging Silicon after manufacturing. With some of the SoCs and MultiCores integrating analog cores, the task of debugging deeply embedded analog cores is becoming even more a difficult task.

For faster time to market, SoCs and MultiCores instantiate reuse cores. Reuse cores could be soft-cores or hard-cores, sometimes developed internally within the same organization or cores procured externally from external vendors. Design for Debug needs to be planned right from the beginning while developing the reuse cores for ease of debug at the SoC as well as at the MultiCore level, in order to avoid challenges related to Silicon Debug. This panel will address the issue related to “Are these debug features that are incorporated in reuse cores sufficient for SoCs and MultiCores that are being designed today?”

1. Challenges associated with Multi-Core and SoC Debug

With the advent of System-on-Chip (SoC) technology and integration of multiple cores on a single die resulting in Multi-Core chips, complexity of integrated circuits has increased considerably. This has resulted in additional complexity in debugging Silicon after manufacturing. Some of the challenges associated with debug of SoCs and MultiCore are related to the following:

1. Lack of observability of nodes in deeply embedded cores
2. Debug of deeply embedded arrays
3. Functional debug of deeply embedded cores
4. Speed path debug of deeply embedded cores
5. Debug of interface between cores
6. Debug related to multiple JTAG TAP controllers
7. Debug of analog cores within SoCs and MultiCore
8. Low power debug

Some of the above debug challenges are attributed to the complexity involved with integration of the SoCs and MultiCores, with additional complexity caused with the use of digital as well as analog cores. In certain cases, RF cores are also integrated into the same SoC/MultiCores, compounding debug challenges. Classic Design for Debug (DFD) techniques, such as bringing out test points, as well as making internal nodes visible outside for the purpose of debug work well for stand-alone ICs. But, for deeply embedded cores, such techniques are clearly not sufficient.

Speed-path related bugs become very difficult to debug due to inability to observe the cause of speed path at the output. Special speed-path debug techniques need to be incorporated early in the design stage, in the deeply embedded core for ease of silicon debug. Other important areas that cause debug challenges are in the area of debug of deeply embedded arrays as well as debug related to multiple JTAG TAP controllers at the SoC/MultiCore level.

Current day SoCs are also used widely in many applications such as handhelds, cell-phones, network processors, wireless applications etc., and have unique requirements of multiple voltage domains, multiple power domains, multiple
synchronous clock domains and additionally asynchronous clock domains as well. Asynchronous interface debug is a challenging task. Additionally, there are situations where the part functions well but the power drawn by the device is above the spec. Additional debug hooks may be required at the reuse core level or needs to be incorporated by the SoC/MultiCore integrator for ease of power debug as well as for debug of all the above issues.

This panel will try to address several important questions related to, “What debug hooks do we need to plan early on at the reuse core level to make debug at the SoC/MultiCore level easy?” Since bringing up first silicon is extremely important for time to market, this is definitely an important question that this panel will try to address and come to a consensus.

This panel session will have four panelists with each of the panelists discussing one or more of the challenges involved with SoC and MultiCore debug with their position statements and presentations, after which we will open up for questions from the audience to find the verdict on “SoC and Multi-Core Debug: Are Design for Debug (DFD) features that are put in reuse cores sufficient for Silicon Debug?”

Coordinates:

Organizer: Sankaran Menon, Intel Corporation
Intel Corporation, MS: AN1-3
1501 S. Mopac Expressway
Austin, TX 78746
E-mail: Sankaran_Menon@intel.com
Tel: (512)-314-0573
Fax: (512)-314-1103

Moderator: Yervant Zorian, Virage Logic

Panelist 1: Teresa McLaurin, ARM
E-mail: Teresa.McLaurin@arm.com

Panelist 2: Bob Gottlieb, Intel
E-mail: bob.gottlieb@intel.com

Panelist 3: Mike Genden, IBM
E-mail: Mgenden@us.ibm.com

Panelist 4: Eric Rentschler, AMD
E-mail: Eric.Rentschler@amd.com