Accellera SystemC Standards Update

Martin Barnasconi
Accellera Technical Committee Chair

www.accellera.org
Outline

• Accellera Systems Initiative
• SystemC ecosystem
• Accellera SystemC Working Groups
  – SystemC Language Working Group
  – SystemC Analog/Mixed-Signal Working Group
  – SystemC Configuration, Control & Inspection Working Group
  – SystemC Synthesis Working Group
  – SystemC Verification Working Group
• IEEE related Working Groups
• Advancing SystemC Standards Together
Accellera Systems Initiative

Our Mission

To provide a platform in which the electronics industry can collaborate to innovate and deliver global standards that improve design and verification productivity for electronics products.
Accellera Membership - Broad Industry Support

<table>
<thead>
<tr>
<th>Corporate Members</th>
<th>Associate Members</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMD</td>
<td>aedvices</td>
</tr>
<tr>
<td>arm</td>
<td>ALDEC</td>
</tr>
<tr>
<td>cadence</td>
<td>AMIQ</td>
</tr>
<tr>
<td>Ericsson</td>
<td>amUniv</td>
</tr>
<tr>
<td>Intel</td>
<td>ANALOG DEVICES</td>
</tr>
<tr>
<td>Mentor (A Siemens Business)</td>
<td>BOSCH (Invented for life)</td>
</tr>
<tr>
<td>NXP</td>
<td>BREKER</td>
</tr>
<tr>
<td>Qualcomm</td>
<td>CLIQ Soft</td>
</tr>
<tr>
<td>Synopsys</td>
<td>COSEDA</td>
</tr>
<tr>
<td>ST</td>
<td>DEFACTO</td>
</tr>
<tr>
<td></td>
<td>dialog</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DOULOS</td>
</tr>
<tr>
<td></td>
<td>Fraunhofer IS</td>
</tr>
<tr>
<td></td>
<td>IBM</td>
</tr>
<tr>
<td></td>
<td>infineon</td>
</tr>
<tr>
<td></td>
<td>IRT</td>
</tr>
<tr>
<td></td>
<td>magillem</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MARVELL</td>
</tr>
<tr>
<td></td>
<td>Methodics</td>
</tr>
<tr>
<td></td>
<td>NVIDIA</td>
</tr>
<tr>
<td></td>
<td>onespinc</td>
</tr>
<tr>
<td></td>
<td>Semifore</td>
</tr>
<tr>
<td></td>
<td>SiFive</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sigasi</td>
</tr>
<tr>
<td></td>
<td>Texas Instruments</td>
</tr>
<tr>
<td></td>
<td>Tortuga Logic</td>
</tr>
<tr>
<td></td>
<td>Vayavya Labs Pte. Ltd.</td>
</tr>
<tr>
<td></td>
<td>Verific</td>
</tr>
<tr>
<td></td>
<td>Xilinx</td>
</tr>
</tbody>
</table>

Start-Up and University

|                     | Ethernovia        |
|                     | University of Maryland |
|                     | UPMC              |
|                     |                  |

© Accellera Systems Initiative
Accellera Standards Developments

System/Design – Analog & Digital
- SystemC
- TLM/CCI/Synthesis
- SystemC-AMS
- SystemVerilog
- SV-AMS/V-AMS

Verification – Analog & Digital
- UVM
- UVM-AMS
- Portable Stimulus
- Multi-Language
- UCIS
- OVL

Integration – Infrastructure
- IP Security Assurance
- Functional Safety
- IP-XACT
- SCE-MI
- IP Tagging
- OCP
- SystemRDL
SystemC ecosystem

- SystemC is a C++-based language standard, widely used for:
  - System-level modeling, design and verification
  - Architectural exploration, performance modeling
  - Analog/mixed signal modeling
  - High-level Synthesis
  - Software development
- Defined by Accellera, ratified as IEEE Std. 1666-2011 (SystemC) and IEEE Std. 1666.1-2016 (SystemC AMS)
Accellera SystemC Working Groups

- **SystemC Language Working Group (LWG)**
  - Chair: Laurent Maillet-Contoz (ST), Vice-chair: Andy Goodrich (Allied member)
  - Subgroups
    - Transaction-Level Modeling (TLMWG), Chair: Bart Vanthournout (Synopsys)
    - Datatypes (SDTWG), Chair: Frederic Doucet (Qualcomm)
    - Common Practices (CPS): Chair: Mark Burton (IRT Saint-Exupery), Vice-chair: Joachim Geishauser (NXP)

- **SystemC Analog/Mixed-Signal Working Group (AMSWG)**
  - Chair: Martin Barnasconi (NXP), Vice-chair: Christoph Grimm (TU Kaiserslautern)

- **SystemC Configuration, Control & Inspection Working Group (CCIWG)**
  - Chair: Ola Dahl (Ericsson), Vice-chair: Bart Vanthournout (Synopsys)

- **SystemC Synthesis Working Group (SWG)**
  - Chair: Andres Takach (Mentor), Vice-chair: Mike Meredith (Cadence)

- **SystemC Verification Working Group (VWG)**
  - Chair: Stephan Gerth (Bosch), Vice-chair: Bas Arts (NXP)
SystemC Language Working Group

- The SystemC Language Working Group is responsible for the definition and development of the SystemC core language, the foundation on which all other SystemC libraries and functionality are built.

- **Current status**
  - SystemC IEEE Std. 1666-2011 made available by Accellera under the [IEEE GET Program](https://www.ieee.org/get-program)
  - SystemC/TLM 2.3.4 public release available at [Github](https://github.com/accellera/systemc)
  - Currently refining proposals and handover of language updates to IEEE P1666 WG

- **Future plans & directions**
  - Define industry common practice aiming at interoperability using SystemC TLM and CCI extensions for commonly used bus interfaces
  - Alignment and consolidation on SystemC Datatypes to enhance HLS flows
SystemC Analog/Mixed-Signal WG

• The SystemC AMS Working Group is responsible for the standardization of the SystemC AMS extensions, defining and developing the language, methodology and class libraries for **analog, mixed-signal and RF modeling** in SystemC.

• **Current status**
  – SystemC AMS IEEE Std. 1666.1-2016 made available by Accellera under the [IEEE GET Program](#)
  – Developing SystemC AMS regression test suite for release later this year

• **Future plans & directions**
  – Definition of new SystemC AMS language extensions as preparation for next IEEE P1666.1 revision
SystemC Analog/Mixed-Signal WG

- SystemC AMS defines 3 additional models of computation focusing on efficient AMS system-level modeling concepts
  - Timed Data Flow (TDF)
  - Linear Signal Flow (LSF)
  - Electrical Linear Networks (ELN)
- Practical SystemC AMS User’s Guide and application examples explaining the language constructs and execution semantics in detail
SystemC Configuration, Control & Inspection WG

• The SystemC Configuration, Control and Inspection WG is responsible for developing standards that allow tools to interact with models in order to perform activities such as setup, debug and analysis.

• Current status
  – Configuration, Control & Inspection Language Reference Manual released in 2018
  – Availability of a Reference implementation and a collection of examples to demonstrate the use and value of the SystemC CCI 1.0 standard

• Future plans & directions
  – Review of checkpointing (save/restore) capabilities based on a contribution of Intel
  – Define Register Introspection API
  – Evaluation of the use of the CCI configuration mechanism as basis for the Common Practices WG to enable interoperability of TLM extensions
CCI 1.0 covers standardized interfaces for parameters
• Contribution under review enabling checkpointing (save/restore)
SystemC Synthesis WG

• The SystemC Synthesis Working Group is responsible for the SystemC synthesis subset, to enable synthesis of digital hardware from high-level specifications.

• Current status
  – Working on second version of the SystemC Synthesis Subset standard

• Future plans & directions
  – Update and finalize support of modern C++ language features defined in C++11/14/17
  – Gather and evaluate additional requirements
  – Alignment and consolidation on SystemC Datatypes to enhance HLS flows
• The SystemC Verification WG is responsible for defining verification extensions to the SystemC standard and reference implementation by offering an add-on libraries to ease the deployment of a verification methodology based on SystemC.

• **Current Status**
  – SystemC Verification Library (SCV) maintenance released in 2017
  – UVM-SystemC Library 1.0-beta3 released in July 2020

• **Future plans & directions**
  – Objective to release UVM-SystemC library 1.0 later this year
  – Introduce Constrained Randomization engine based on CRAVE contribution from University of Bremen
  – Introduce Functional Coverage based on FC4SC contribution of AMIQ Consulting
SystemC Verification Working Group

- The UVM-SystemC library enables the creation of a modular, scalable, configurable and reusable testbenches
  - Following the principles of the Universal Verification Methodology (UVM)
  - Implemented in C++/SystemC, offering flexibility and reuse across verification and validation domains
- Additional verification-specific features such as constrained randomization and functional coverage will be addressed by supporting add-on libraries such as CRAVE and FC4SC
IEEE related Working Groups

• P1666
  – Latest version: IEEE 1666-2011, published 2012-01-09
  – Chair: Jerome Cornet (ST Microelectronics)
  – P1666 WG currently active

• P1666.1
  – Latest version: IEEE 1666.1-2016, Published 2016-04-06
  – Chair: Martin Barnasconi (NXP)
  – P1666.1 WG not active
Advancing SystemC Standards Together

• Become an Accellera Working Group member
  – Join Accellera and participate in the Accellera working groups
  – Direct access to the latest standardization proposals and development tree

• Become a member of the IEEE Standards Association
  – Join IEEE-SA to participate in the ongoing standardization in the P1666 (SystemC) working group

• Share your experiences
  – Visit www.accellera.org and join the community forums at forums.accellera.org
  – Report your issues and/or create pull requests on the public SystemC Github repository

• Help us to grow the SystemC footprint and community
  – Participate in community events such as the SystemC Evolution Day
  – Promote the use of the SystemC standard in complex system simulation tasks
Thank You

Q&A