EDITOR **RICCARDO MARIANI**
NVIDIA; rmariani@nvidia.com

Design Automation Standards Committee: Setting Tomorrow's Standards

Dennis Brophy, Siemens EDA

The IEEE work program for electronic design automation standards remains vibrant and its volunteers committed to continued design and verification productivity advancements.

In this article, I will focus on the work within the IEEE Design Automation Standards Committee (DASC). The group can be found online at https://www.dasc.org. IEEE DASC was formed at about the time that the electronic design automation (EDA) industry started and the U.S. Department of Defense initiated the Very-High-Speed Integrated Circuits (VHSIC) Program in 1980. The following year, a group from industry, government, and academia got together in Woods Hole, Massachusetts. VHSIC goals were discussed, and there was an imperative that a common language was needed to document design. This identified need was made at the time in which commercial EDA tools were focused on schematic capture-based systems to document logic design.

Notwithstanding current commercial technology, the U.S. government initiated a contract that was awarded to a team from Intermetrics, IBM, and Texas Instruments to develop the VHSIC Hardware Description Language (VHDL). Given the life of some hardware systems, having a way to document them in a language that could transcend current practice might be a good thing. Having a language which would be common across all engineers might even be a better thing.

VHDL 7.2 was completed in August 1985 and was the final version of the language under the government contract. It was moved to what was then known as the DASC and IEEE for ongoing maintenance and revision. In 1987, IEEE Standard 1087-1987 was completed and, a year later, was recognized as an ANSI standard.

The story does not end there—rather, it was the beginning of a vibrant standards-development group to foster design and verification productivity. As IEEE moved to form the IEEE Standards Association (SA) and put in place common rules and procedures by which to develop standards, IEEE DASC updated it rules and

Digital Object Identifier 10.1109/MC.2021.3076375
Date of current version: 1 July 2021

processes to stay aligned. As IEEE SA fostered tighter bonds with other international standards organizations, like the International Organization for Standardization and the International Electrotechnical Commission (IEC), IEEE DASC had to work to use the IEEE/IEC dual-logo process to reduce the friction of fully international standards recognition for its work.

IEEE SA has expanded its process to include standards development in a way that brought us VHDL in the first place with the embrace of entity-based standards. For IEEE DASC, the development of SystemVerilog was the first entity-based project done in collaboration with the IEEE SA Corporate Advisory Group, with content originating in Accellera, a recognized member of the IEEE SA Industry Affiliate Network. Much like in VHDL's initial development, industry came together in Accellera to define SystemVerilog and handed it off to IEEE DASC, which worked on it to make IEEE Standard 1800-2005.

IEEE DASC has been a strong adopter of the entity process as it has aligned with its history of standards formation and development. IEEE DASC-sponsored working groups make the decision to use an entity or individual expert process. Both continue to be used. But, just in the case of the industry backing of the early definition of VHDL, SystemVerilog found that bringing industry together promoted the rapid adoption of SystemVerilog.

SUSTAINED INDUSTRY COLLABORATION MEETS MARKET NEEDS

The sustained relationship IEEE DASC and IEEE SA has had with Accellera has also been a positive one. Accellera has been a feeder into IEEE DASC to handle ongoing maintenance and updates to the standards it launched. It has also worked with IEEE SA to have many of the EDA standards available without cost to the end users. The IEEE GET program has made it easier for all users of SystemVerilog, as an example, to gain access to this standard. For the academic community, it has made it easier for students to download a standard as they use it in their course work. Today, IEEE GET Design Automation Standards feature access to the following seven standards:

- 1666-2011: *IEEE Standard for Standard SystemC Language Reference Manual*
- 1666.1-2016: *IEEE Standard for Standard SystemC(R) Analog/Mixed-Signal Extensions Language Reference Manual*
- 1685-2014: *IEEE Standard for Intellectual Property (IP)-XACT Standard Structure for Packaging, Integrating, and Reusing IP Within Tool Flows*
- 1735-2014: *IEEE Recommended Practice for Encryption and Management of Electronic Design IP*
- 1800-2017: *IEEE Standard for SystemVerilog–Unified Hardware Design, Specification, and Verification Language*
- 1800.2-2020: *IEEE Standard for Universal Verification Methodology (UVM) Language Reference Manual*
- 1801-2018: *IEEE Standard for Design and Verification of Low-Power, Energy-Aware Electronic Systems.*

IEEE DASC thanks Accellera for its support to make these design automation standards available to users globally at no charge. Table 1 presents the last 11 years of download history, starting with the first standard added to the IEEE GET program: IEEE Standard 1685 (also known as IP-XACT).

The IEEE DASC working groups continue to work with the IEEE SA to evolve standards development. Returning to DASC's first standard, VHDL, we see it has played a key role in the IEEE SA Open pilot projects to test IEEE SA's open source platform. For those who don't know the VHDL standard well, it

TABLE 1. IEEE GET download history from 2010 to February 2021. The 2017 data may not be accurate due to the IEEE download platform migration.

Standard	2010	2011	2012	2013	2014	2015	2016	2017*	2018	2019	2020	2021 YTD	Total
1685	1,495	2,934	2,262	2,152	1,411	1,411	1,542	125	562	892	1,025	145	15,811
1666			9,474	4,411	2,979	2,507	2,032	143	589	1,033	1,529	292	24,989
1800				10,500	7,182	7,856	11,653	833	4,577	6,467	7,911	1,435	58,414
1801				1,967	2,262	2,324	2,709	258	1,108	1,252	1,212	222	13,314
1735							894	80	249	429	504	69	2,225
1666.1							713	63	228	271	395	74	1,744
1800.2								230	2,081	2,592	2,486	420	7,809
Total	**1,495**	**2,934**	**11,736**	**19,030**	**13,834**	**14,098**	**19,543**	**1,732**	**9,394**	**12,936**	**15,062**	**2,657**	**124,306**

merits recognition that the general nature of the language made it difficult to write design descriptions that were completely portable so they would simulate identically using different vendor tools.

At the heart of this was VHDL's support of abstract data types while it did not address the issue of characterizing different signal strengths or commonly used simulation conditions like unknowns and high impedance. To solve the nonstandard data type issues, another standard, IEEE Standard 1164, was created and defined the VHDL standard logic package. Over time, this separate standard became part of IEEE Standard 1076, and the package code was no longer embedded in the standard or found in a side file for which each user would secure some rights to use and share from the IEEE. The VHDL working group was the first to migrate its code to an open source platform and use the IEEE SA Open platform for that purpose. For those who know VHDL well, you know there are more packages than *std_logic_1164*, such as *textio*, *float_pkg* (with variants), and so on.

As part of the IEEE SA Open pilot project, the VHDL team migrated the different packages onto the IEEE SA Open repository. Not only can the code be found and easily used (can you imagine extracting it from a printed standard?), but also an industry common permissive license is used. The IEEE Standard 1076 package code is licensed under Apache 2.0, which gives VHDL users license and rights certainty and confidence to use and share that which they create in VHDL. The VHDL open source code is available here: https://opensource.ieee.org/vasg/Packages.

IEEE SA Open offers more than just support for official IEEE standards projects; IEEE SA Open is available to anyone. You can explore current groups to see what open source projects are available. You can join or create a project. The only requirement is to have an IEEE account. IEEE membership is not required to get an IEEE account. You can find details about this at https://opensource.ieee.org/.

TODAY'S IEEE DASC STANDARDS PROGRAM

IEEE DASC sponsors several working groups. Many have recently completed projects, and others are in the middle of development. IEEE numbers their standards while those developing them often use its name. I will share a brief status of the standards by name and their associated IEEE standards numbers. You will also note that companies in the EDA industry and other industry bodies have been generous to seed the start of many IEEE DASC standards as this historical information is included.

- *VHDL (both digital and analog mixed-signal, 1076 and 1076.1):* The original IEEE DASC-sponsored standard was recently revised, and IEEE Standard 1076-2019 is now the current version. It is the first IEEE DASC standard to officially incorporate managed open source under the Apache 2.0 license using IEEE SA Open. The standard supports the design, development, verification, synthesis, and testing of hardware designs; the communication of hardware design data; and the maintenance, modification, and procurement of hardware. It is meant for digital flows, and 1076.1 (analog-mixed signal VHDL) includes support for analog designs.
- *Functional Verification Language e (IEEE Standard 1647-2019):* The e language was originally developed in 2002 by Verisity with the first standardization by the IEEE in 2005. The e functional verification language is an application-specific programming language aimed at automating the task of verifying a hardware or software design with respect to its specification. The most recent version of the standard was published in August 2019.
- *SystemC Language (IEEE Standard 1666-2011):* SystemC, originally developed by Synopsys, was initially standardized by the Open SystemC Initiative (OSCI), which merged into Accellera. In 2004, OSCI contributed SystemC to the IEEE, and it became an IEEE standard in 2005. SystemC is based on the C++ programming language and adds additional structures to C++ necessary to describe hardware designs. SystemC facilitates modeling hardware at a higher level of abstraction. The companion base class library (BCL) is an open source project in Accellera using the Apache 2.0 license.
- *IP-XACT, Standard Structure for Packaging, Integrating, and Reusing IP Within Tool Flows (IEEE Standard 1685).* IP-XACT is an electronic data book—a description of components and designs written in a standard data exchange format (XML) that is both human readable and machine processable. It was originally developed by Mentor Graphics Corporation (now Siemens EDA) and standardized in 2009 by the SPIRIT Consortium, which merged into Accellera. It is now known as IEEE Standard 1685-2009. IP-XACT is intended to enable automated design integration and configuration

IEEE SA Open offers more than just support for official IEEE standards projects; IEEE SA Open is available to anyone.

within multivendor design flows. The IEEE SA New Standards Committee just approve a project authorization request for the working group to start a revision project. The work for this is being done in conjunction with Accellera.

- *Recommended Practice for Encryption and Management of Electronic Design IP (IEEE Standard 1735):* The working group is focused on a revision of its current recommended practice to address limitations and known exploits. The recommended practice offers guidance on technical protection measures to those who produce, use, process, or standardize the specifications of electronic design IP. The distribution of IP creates a risk of unsanctioned use and dilution of the investment in its creation. Measures in the recommended practice include protection through encryption, specification, management of use rights that have been granted by the producers of electronic designs, and methods for integrating license verification for granted rights.
- *SystemVerilog (IEEE Standard 1800-2017):* The second major IEEE DASC language standard was Verilog (or IEEE Standard 1364). IEEE Standard 1800 was built on IEEE Standard 1364, and in its second revision merged IEEE Standard 1364 into and absorbed the Verilog working group to form the SystemVerilog working group. The Verilog Hardware Description Language was originally developed by Gateway Design Automation and its descendent, Cadence Design Systems. Cadence contributed Verilog for standardization with the formation Open Verilog International (OVI), which subsequently contributed Verilog to the IEEE. It was first standardized in 1995. SystemVerilog, which extended Verilog to higher abstraction levels and added assertions for improved formal verification support and testbench constructs to support improved verification, was developed by Accellera in 2002. Accellera contributed SystemVerilog to the IEEE in 2004, and it became a standard in 2005.
- *UVM 1800.2 (IEEE Standard 1800.2-2020):* This is a set of application programming interfaces that define a BCL definition used to develop modular, scalable, and reusable components for functional verification environments based on the SystemVerilog standard. UVM is a combination of technology donations from Mentor Graphics Corporation (now Siemens EDA), Cadence Design Systems, and Synopsys to Accellera. Accellera merged these contributions to form UVM and contributed that to the IEEE. Verification components and environments had been created in different forms, making interoperability among verification tools and/or geographically dispersed design environments time-consuming to develop and error prone. The results of the UVM standardization effort improved interoperability and reduced the cost of repurchasing and rewriting IP for each new project or EDA tool and made it easier to reuse verification components. The companion open source implementation in SystemVerilog continues to be maintained by Accellera under the Apache 2.0 license.
- *Unified Power Format (UPF)—Design and Verification of Low-Power, Energy-Aware Electronic Systems (IEEE Standard 1801-2018):* The standard defines the syntax and semantics of a format used to express power intent in energy-aware electronic system design. Power intent includes the concepts and information required for specification and validation, implementation and verification, and modeling and analysis of power-managed electronic systems. It also defines the relationship between the power intent captured in this format and design intent captured via other formats, like VHDL and SystemVerilog. The initial version of the standard was developed by Accellera and known as the *UPF*. Updates to the original standard include enhanced concepts to model power states and transitions at all levels of aggregation, enhanced support for methodologies, such as successive refinement and bottom-up implementation, and a detailed information model that serves as the basis for enhanced-package UPF functions and query functions. The current version also provides support for component power modeling for system-level power analysis in virtual prototyping applications.
- *Large-Scale Integration (LSI)-Package-Board (LBP) Interoperable Design (IEEE Standard 2401-2019):* The standard defines a common interoperable format that is used

> IEEE SA has expanded its process to include standards development in a way that brought us VHDL in the first place with the embrace of entity-based standards.

for the design of LSI, the package for such LSI, and printed circuit board on which the packaged LSIs are interconnected. Collectively, such designs are referred to as *LBP designs*. The format provides a common way to specify information and data about the project management, netlists components, design rules, and geometries used in LBP designs. The second revision of the standard was completed in 2019. The Japan Electronics and Information Technology Industries Association was instrumental in the creation of this standard.

- *Power Modeling to Enable System-Level Analysis (IEEE Standard 2416-2019)*: The standard describes a parameterized and abstracted power model to enable system, software, and hardware IP-centric power analysis and optimization. The standard defines concepts for the development of parameterized, accurate, efficient, and complete power models for systems and hardware IP block usable for system power analysis and optimization. The concepts include things like process, voltage, and temperature independence; power and thermal management interface; and workload and architecture parameterization. These models are suitable for use in software development and hardware design flows. The standard gains leverage from technologies and specifications developed and released by Si2 that include Si2's *Leakage Power Contributor Modeling, Liberty Mode Extensions for Atomic Power Modeling, Standards for Efficient System Level Power Analysis*, and *Multi-level Power Modeling*.
- *Exchange and interoperability format for safety analysis and safety verification of IP, system on chip (SoC), and mixed-signal integrated circuit (IC) (IEEE Standard P2851)*: One of the newest projects to start standardization in DASC has set its target to define a data format with results of safety analysis (such as FMED, FMEDA, FMECA, FTA) and related safety-verification activities (such as fault injection) executed for IPs, SoCs, and mixed-signal ICs exchanged and made available to system integrators. The format will define languages, data fields, and parameters with which the result of those analyses and verification activities can be represented in a technology-independent way. The scope includes items such as systems and software (SW). Artificial intelligence is also a key part of the activity. The goal is for IEEE Standard P2851 to become a family of standards (P2851.1, 0.2, 0.3, and so on) that covers broader functional safety topics, such as system- and SW-level safety analyses and formal/semiformal representations of assumption of use that can extend into adjacent domains, such as cybersecurity analyses and related verification methodologies.

KEEPING PACE FOR TOMORROW

There is more to the history of IEEE DASC standards. There are many completed IEEE DASC standards that continue to be used with limited or no continued enhancement. They are good as is and remain available on IEEE *Xplore*. Yet, tomorrow calls for the creation and adoption of new standards to address the pressing needs of next-generation design and verification. Autonomous systems have shined a spotlight on the needs of standards to be put in place for functional safety solutions. The IEEE DASC and other standards-setting organizations are tackling this and more now.

Tomorrow calls for the creation and adoption of new standards to address the pressing needs of next-generation design and verification.

The globalization of the electronics market has led to projects that are globally dispersed that require design and verification teams to express, evaluate, verify, and package design content in clear and unambiguous ways. Multinational corporations with intracompany design projects look to standards to help their internal needs to facilitate their design and verification productivity as much as to promote more frictionless collaboration with their own customers. Small- and medium-sized companies gain leverage in global markets via standards as well. Those companies build tools to international standards that can be adopted globally.

IEEE DASC continues to provide the necessary standards to foster a smoothly functioning global electronics industry. If there is more we can do, we invite your participation to help shape the future of standards so tomorrow is more than a dream but a reality.

DENNIS BROPHY is the director of strategic business development at Siemens EDA, Wilsonville, Oregon, 97070-7777, USA, and the chair of the IEEE Design Automation Standards Committee. Contact him at dennis.brophy@siemens.com.