SystemC 3.0 Library Update

Laurent Maillet-Contoz, SystemC LWG Chair
Andy Goodrich, SystemC LWG Co-Chair
Outline

• SystemC 3.0.0 library
• Datatypes performance improvements
• SystemC non regression tests
• Windows DLL support
• Overall cleanup
• Summary
SystemC 3.0.0 library

• Implementation aligned to IEEE 1666-2023 revision
  – See previous presentations
    • Sept’23 SystemC Evolution Fika (https://systemc.org/events/scef202309/)
    • SystemC Evolution Day ‘23

• Progressive alignment to C++17 standard

• This is a major release
  – Some APIs have changed
  – Some implementation–specific functions (not part of the IEEE standard) no longer available
sc_biguint & sc_bigint Performance Improvements

• Storing 32 bits rather than 30 bits per word
• Math performed using twos-complement instead of sign-magnitude
• Bit access
• Bit range reads and writes
• Operator and method invocation overhead
• Expanding use of template information in operator and method bodies
• Initialization overhead
• Data storage allocation/deallocation overhead

*Up to 20x performance improvement!*
SystemC non-regression tests

• So far maintained in a separate repo (to be archived)
  – https://github.com/OSCI-WG/systemc-regressions

• From SystemC 3.0.0 final version, integrated in the main repo
  – Under systemc/tests/
  – Including Cmake support

• Preparing for continuous integration through GitHub actions
  – As soon as pushed on https://github.com/accellera-official/systemc
Windows DLL support

- Windows DLL build now supported for MSVC17 and beyond
- `cmake`
  - Specify build system generator e.g. “Visual Studio 16 2019” and compiler “msvc”
  - Add option `-DBUILD_SHARED_LIBS=ON`
- Visual Studio
  - Open solution file (*.sln)
  - Select configuration ReleaseDLL or DebugDLL
- Minor open issue
  - Linker warning LNK4006 (duplicate symbols) can be ignored
- Note
  - Do not forget to add the `systemc-<version>.dll` to your Windows Environment Path
General cleanup

- About 40 Pull Requests merged for the final release related to
  - Fix of UTF-8 issues reported by some compilers
  - `sc_string` (`sc_string_old` removed, `sc_string` now remapped to `std::string`)
  - Removal of unused meta programming helpers: No longer needed thanks to C++17 migration
  - Removal of various preprocessor macros, either
    - Temporarily introduced to help with the adoption of the IEEE 1666-2011 standard
    - Now irrelevant with C++17 alignment
    - Now unconditionally enabled throughout library, examples, and tests
  - `sc_vector` implementation updates to use native C++17 features
  - QuickThreads: drop legacy coroutine stack allocation based on malloc to use `mmap`
  - Windows support: Mingw64, MSVC 2017
  - Tests cleanup
  - Removal of embedded Boost headers
    - Note: move of `sc_bind` from macros to functions not propagated to IEEE 1666-2023
  - More to come on CMake-based flow improvements
Summary

• SystemC 3.0: a significant step forward
  – C++17 alignment
  – Datatypes performance improvements
  – Streamlining of SystemC regression tests in main repo
  – Windows DLL support
  – General cleanup
Q&A