Boosting Simulation Performance of UVM Registers in High Performance Systems

Ahmed Yehia

The proposed implementations of `do_bus_write()` and `do_bus_read()` below avoid chopping data in bus width chunks and let the adapter’s `reg2bus()` method decide the amount of data it is going to consume in a bus transaction according to its bus capabilities. There is some coding required to workaround the `reg2bus()` prototype which returns only one sequence item.

A more efficient prototype of `reg2bus()` is to return an array of sequence items.

Appendix B in the full paper shows a way that can be adopted by the UVM register library to improve the interface of `reg2bus()` to boost simulation performance and still maintains backward compatibility.

The current implementations of `do_bus_write()` and `do_bus_read()` insert undesired bottleneck when performing on high performance buses, by chopping data in bus-width chunks, generating a simple transaction for each chunk, and avoiding making use of bus powerful features when found. Maximizing the number of transactions generated maximizes context switching in simulation, which may have severe consequences on simulation performance imagining a test performing hundreds of these operations.

1. www.uvmworld.org
2. www.verificationacademy.com/uvm-ovm
3. AMBA AXI reference, infodenter.arm.com

Or get a copy of the full paper...