Contribution ID: **401** Type: **Oral presentation**

Two FPGA Case Studies Comparing High Level Synthesis and Manual HDL for HEP applications

Tuesday, 12 June 2018 16:30 (20 minutes)

Real time data acquisition systems in nuclear science often rely on high-speed logic designs to reach the fast data rate requirements. They are mostly coded in a hardware description language (HDL). However, in recent years, so-called high level synthesis (HLS) software has appeared, some with the notable advantage that they rely on the widespread C/C++ syntax. This paper's aim is to outline differences between HDL and C++ HLS based designs for two real time data acquisition modules used in nuclear science. The first module is a real time crystal identification module, and the second is a compact event timestamp sorting module.
For the crystal identification module, both HDL and HLS versions have the same event processing interval, and the HLS implementation consumes twice as many lookup tables and flip flops as the HDL version. On the other hand, the HLS version took half the time to write and debug. For the sorter module, the HLS version requires about 3 to 4 times more logic resources, with a slightly longer interval. The challenge for this module is that some pipeline shortcuts cannot be automatically inferred from simple C++ and must be explicitly written in. This second HLS module was also completed in half the time compared to the original HDL code. While not intuitively applicable to all problem types, HLS is nonetheless a compelling alternative to custom HDL or Verilog implementations for real time systems in nuclear and plasma science.

Description

HDL code

Institute

Harvard

Speaker

Marc-André Tétrault

Country

USA

Minioral

Yes

Primary author: Dr TÉTRAULT, Marc-André (Harvard Medical School)

Presenter: Dr TÉTRAULT, Marc-André (Harvard Medical School)

Session Classification: Emerging Technologies