Contribution ID: **231** Type: **Oral**

Implementing Machine Learning inference on FPGAs: from software to hardware using hls4ml

Wednesday 26 October 2022 15:15 (20 minutes)

In the past few years, using Machine and Deep Learning techniques has become more and more viable, thanks to the availability of tools which allow people without specific knowledge in the realm of data science and complex networks to build AIs for a variety of research fields. This process has encouraged the adoption of such techniques: in the context of High Energy Physics, new algorithms based on ML are being tested for event selection in trigger operations, end-user physics analysis, computing metadata based optimizations, and more. Time critical applications can benefit from implementing algorithms on low-latency hardware like specifically designed ASICs and programmable micro-electronics devices known as FPGAs. The latter offers a unique blend of the benefits of both hardware and software. Indeed, they implement circuits just like hardware, providing power, area and performance benefits over software, yet they can be reprogrammed cheaply and easily to implement a wide range of tasks, at the expense of performance with respect to ASICs.

In order to facilitate the translation of ML models to fit in the usual workflow for programming FPGAs, a variety of tools have been developed. One example is the HLS4ML toolkit, developed by the HEP community, which allows the translation of Neural Networks built using tools like TensorFlow to a High-Level Synthesis description (e.g. C++) in order to implement this kind of ML algorithms on FPGAs.

This paper presents and discusses the activity started at the Physics and Astronomy department of University of Bologna and INFN-Bologna devoted to preliminary studies for the trigger systems of the Compact Muon Solenoid (CMS) experiment at the CERN LHC accelerator. A broader-purpose open-source project from Xilinx (a major FPGA producer) called PYNQ is being tested combined with the HLS4ML toolkit. The PYNQ purpose is to grant designers the possibility to exploit the benefits of programmable logic and microprocessors using the Python language. This software environment can be deployed on a variety of Xilinx platforms, from IOT devices like the ZYNQ-Z1 board, to the high performance ones, like Alveo accelerator cards and on the cloud AWS EC2 F1 instances.

Even though a rich documentation can be found on how to use hls4ml, a comprehensive description of the entire workflow from Python to FPGA is still hard to find. This work tries to fill this gap, presenting hardware and software set-up, together with performance tests on various baseline models used as benchmarks. The presence or not of some overhead causing an increase in latency will be investigated. Eventually, the consistency in the predictions of the NN, with respect to a more traditional way of interacting with the FPGA using C++ code, will be verified.

Significance

This talk would present, through examples and actual lines of code, for the first time the entire workflow needed to go from a purely software Neural Network in Python to the hardware implementation on a generic FPGA, together with the possibility of using PYNQ to run the inference on compatible boards.

References

https://pos.sissa.it/378/005/
https://indico4.twgrid.org/event/20/contributions/1119/attachments/672/775/ISGC2022_slides.pdf

https://agenda.infn.it/event/28874/contributions/169219/attachments/94335/129059/ICHEP2022_slides_Lorusso.pdf

Experiment context, if any

Compact Muon Solenoid at CERN

Primary author: LORUSSO, Marco (Universita e INFN, Bologna (IT))

Co-authors: Prof. BONACORSI, Daniele (University of Bologna / INFN); TRAVAGLINI, Riccardo (INFN, Bologna (IT))

Presenter: LORUSSO, Marco (Universita e INFN, Bologna (IT))

Session Classification: Track 1: Computing Technology for Physics Research

Track Classification: Track 1: Computing Technology for Physics Research