University Booth at DATE 2020 Demonstrator Description


Marc Benito\(^1,2\), Matina Maria Trompouki\(^3\), Leonidas Kosmidis\(^2,1\)

\(^1\) Universitat Politècnica de Catalunya, Barcelona, Spain \quad \(^2\) Barcelona Supercomputing Center (BSC)

Current critical systems require increased performance in order to support advanced functionalities such as autonomous operation. Graphics Processing Units (GPUs) can provide this required performance, however their general programming models like CUDA or OpenCL cannot be used in such systems, since they violate by design safety critical programming guidelines like the use of pointers and dynamic memory allocation. This prevents the certification of their software according to safety standards such as ISO26262 used in automotive and DO-178B used in avionics.

On the other hand, subset of graphics APIs exist which are designed for safety critical systems, such as OpenGL SC 2. However, programming general purpose algorithms on graphics -- similar to the first days of GPGPU (general purpose computations on GPU), before the appearance of CUDA and OpenCL -- is very complex and error prone. Our open source technology, Brook SC [1], which is based on our DAC 2018 Brook Auto paper [2], allows to program safety critical applications in a CUDA-like high level general purpose GPU language, Brook, which is transformed into safety critical graphics APIs, enabling the certification of the code, while increasing the developer’s productivity.

Moreover, unlike closed-source toolchains like CUDA and OpenCL with massive code-bases, the small size of Brook SC and its open source nature simplifies the collection of the appropriate evidence to support its qualification, as we show in our recent ICCD 2019 paper for ISO 26262 [3].

In our demo, an avionics application running on a realistic safety critical GPU software stack and hardware is show cased. In this Bachelor’s thesis project [4], which was awarded a 2019 HiPEAC Technology Transfer Award, an Airbus prototype application performing general-purpose computations with a safety-critical graphics API was ported to Brook SC in record time, achieving an order of magnitude reduction in the lines of code to implement the same functionality without performance penalty.

Our booth visitors will have the opportunity to see both the original version of the application as well as its Brook SC version being executed on an AMD E8860 avionics-grade GPU and on top of a commercial OpenGL SC 2 driver (ArgusCore SC2), both from CoreAVI. The application will be also showcased on other embedded GPUs, on top of their stock OpenGL ES 2 drivers. Finally, the visitors will be able to experiment with Brook SC in the latter platforms and learn more about the language, its capabilities and how to use it in their projects.

References


