The Carte++™ Application Programming Environment Advantage
SRC Computers, LLC
www.srccomputers.com

This white paper contains information which may be protected by issued and pending patents in the United States and elsewhere. No license under any SRC Computers, LLC intellectual property rights is granted by its publication.

OVERVIEW

To access the SRC system benefits of higher application performance, lower power consumption, reduced footprint and superior total cost of ownership (TCO) advantages, SRC Computers introduces the Carte++ programming environment. Carte++ builds upon and extends the proven technology of the flagship Carte™ product which was first released in 2002.

Carte++ offers application programmers a complete set of features unavailable elsewhere:

- Mature, proven code development technology.
- A single environment to compile source code for multiple processor types.
- Software programmer oriented. No hardware knowledge needed.
- Incorporation of popular open-source Clang/LLVM.
- Standard Linux libraries that remain available for microprocessor execution.
- Full support of any SRC system size or configuration.
- Ability to import legacy HDL modules.
- Unified application development environment.
- Multiple execution targets: debug, simulation, and hardware.
- Java, Python, Fortran. Any language that can call C functions.
- Standard C/C++ instead of C-like or HDL programming.
- Performance-oriented libraries for MAP processor execution.
- Run-time enables autonomous heterogeneous peer processor cooperation.
- Microprocessor and FPGA vendor agnostic.

DESIGNED FOR MAXIMUM PERFORMANCE AND FLEXIBILITY

There are two major architectures for heterogeneous reconfigurable systems available today: master processor with accelerator and autonomous peer processor. Nearly all of the existing offerings, except those offered by SRC Computers, fall into the “master processor/multiple accelerator device” category supported by OpenCL. The accelerator model depletes any potential application performance gains due to data movement and coordination overhead required between the central processor and accelerator(s).

In contrast, SRC systems fall into the “autonomous peer processor” category supported by Carte++. For maximum overall application performance, the Carte++ run-time environment allows all processors in a system to operate independently of one another or selectively cooperate as required by the application.

SRC Computers’ reconfigurable systems are comprised of a mix of microprocessor, reconfigurable processor, common memory, and crossbar switch module types. These modules may exist in any quantity and combination, defined by customer application requirements. Carte++ fully supports any combination and number of modules in an SRC reconfigurable system. Lastly, over the years, SRC Computers has remained completely agnostic with respect to microprocessors and FPGAs in these modules. Carte++ is designed to allow SRC to change microprocessors and/or FPGAs at will with minimal impact on the end users.

NO HARDWARE KNOWLEDGE REQUIRED

SRC Computers’ reconfigurable systems consist of at least two autonomous peer processors: one microprocessor and one SRC-developed reconfigurable MAP® processor. This peer-based architecture is called the IMPLICIT+EXPLICIT™ Architecture. Carte++ unifies several third-party and SRC tools required to create applications using both types of
processors. This environment enables the creation of a single executable from microprocessor and MAP processor source code for three specific targets (“modes”): debug, simulation and hardware. The first of these, debug mode, is used to establish program correctness. All source code is compiled and executed on the microprocessor so that standard Linux debugging tools may be utilized. The second, simulation mode, provides clock cycle accurate execution in the rare cases that level of debugging is required. Once all source code is compiled, the microprocessor code is executed on the microprocessor and the MAP processor code is executed on the microprocessor under the iVerilog simulation engine. The last target, hardware mode, creates the executable “bit stream” for the components on the MAP processor and combines this with compiled microprocessor code into one executable file. All of the compilation flow and details for each mode are abstracted away from the user. The application programmer simply types “make debug”, “make sim” or “make hw” at the Linux command line.

While the SRC MAP processor is based upon FPGA technology, Carte++ is designed so that programmers do not need to know anything about FPGAs. Traditionally, FPGAs have been the domain of hardware designers who use Hardware Descriptor Languages (HDLs). Instead of requiring programmers to learn idiosyncratic C-like coding (OpenCL) or HDLs (Verilog, VHDL), Carte provides widely accepted, standard software languages (C, C++) for programming both processors in the system.

**CLANG/LLVM**

Carte++ incorporates the popular, well-supported Clang/LLVM project as its C/C++ compiler. Currently, the LLVM front-end Clang supports C, C++, OpenCL, and Objective-C code compilation. There are additional LLVM front-ends under development in the open-source community, including one for Fortran. Carte++ is intentionally designed to accommodate any language supported by proven, mature LLVM-compliant compiler front-ends in the future should customer demand arise. Carte++ already provides microprocessor support for any software language that has the ability to call C functions (Java, Python, Fortran).

**PROGRAMMING LIBRARIES**

Any standard Linux library may be included for the microprocessor. In addition, Carte++ provides extensive function call libraries for the MAP processor. These MAP processor specific library functions fall into two major categories: application domain enablers and performance enablers. For the former, some applications require specialized programming (signal processing, cryptography, Ethernet network programming) and so these are provided for the MAP processor. For the latter, Carte++ provides functions designed to enable programmers to gain the desired application performance levels, despite C’s intrinsically serial nature.

Some customers require a third type of “library”. They may have legacy HDL code or IP cores created for FPGAs from other projects, and Carte++ provides the facility to wrap their HDL code into a set of parameterized C/C++ function calls.

**SUMMARY**

Carte++ is the only future-proof, comprehensive and unified application development environment utilizing standard software languages for effective microprocessor/FPGA-based computation. Furthermore, Carte++ gives application programmers full access to the high performance, low power, small footprint and superior TCO advantages inherent in SRC reconfigurable servers.

---

1 SRC Application Implementations on a Saturn 1 Processor, SRC Computers LLC, MKT-054-00, August 3, 2012.