# sc2 C-to-FPGA Compiler

Maya Gokhale, Jan Stone, Jan Frigo and Christine Ahrens
Los Alamos National Laboratory
Los Alamos, NM 87545
maya@lanl.gov, jmstone@nerc.com, jfrigo@lanl.gov, cahrens@lanl.gov
505-665-9095(tel) 505-665-5722 (fax)

## Abstract

*sc2 is a new, open source implementation of the Streams-C language and compiler [1] that uses the Stanford SUIF 1.3 compiler infrastructure [2]. sc2 has been improved through various standard compiler optimizations and retargetted to Xilinx Virtex technology. The sc2 compiler passes are freely available for non-commercial use in source form from Los Alamos National Laboratory. Please refer to the Streams-C web site, http://rcc.lanl.gov/Tools/Streams-C/ for download information.*

*The Streams-C programming model is that of communicating processes. A system consists of a collection of processes that communicate using streams and signals. Processes can run either in software on conventional processors (SP) or in hardware on FPGA processors (HP). The sc2 compiler is used to compile FPGA processes in hardware. The compiler translates a subset of C into Register-Transfer-Level (RTL) VHDL that is synthesizable on FPGAs. The sc2 compiler synthesizes hardware circuits for one or more FPGAs as well as a set of communicating processes on conventional processors. The language extensions allow pipelined stream computation, so that the generated hardware/software is capable of pipelining a computation across multiple FPGAs and the conventional processor.*

*In the Streams-C approach, we target an intermediate level of expression. Our compiler processes a subset of C suitable for automatic synthesis to FPGAs. Our programming model is targeted at stream-oriented FPGA applications. Characteristics of stream-oriented computing include high-data-rate flow of one or more data sources, fixed size, small stream payload (one byte to one word), compute-intensive operations, usually low precision fixed point on the data stream, access to small local memories holding coefficients and other constants, and occasional synchronization between computational phases.*

*A software library using POSIX threads provides concurrent processes and stream support in software. Thus the software libraries support a dual function: when all processes are mapped to software, our system provides a functional simulation environment for the parallel program. When processes are mapped to a combination of software and hardware, the software libraries are used for communication among software processes and between software and hardware processes. Hardware libraries, for the Annapolis Micro Systems (AMS) Firebird board, which contains one Xilinx Virtex-E FPGA on a 64-bit PCI bus, are used for communication among hardware processes and for the hardware side of communication to software processes.*

*We compare the sc2 approach to other compiler projects that target FPGAs such as SA-C [3], Handel-C[4], and DeFacto [5].*

*We report results for two application written in Streams-C and compiled with the sc2 compiler to the AMS Firebird board. The applications mapped to reconfigurable hardware are a polyphase filter bank[7] and a K-means clustering[8] algorithm.*

*Multi-rate filter banks have been employed to help detect RF signals in noisy environments. By decomposing a signal into various frequency subbands, filter banks enhance many algorithms because they make it easier to identify pertinent material on a band by band basis. The polyphase implementation[1] is a multi-rate filter structure combined with a Fast Fourier Transform (FFT) designed to extract subbands from an input signal[7]. A filter bank of four prototype filters is implemented in hardware. The stream input data from the host is unsigned, fixed point, 8 bit data. The coefficents are unsigned, fixed point 12 bit values. The sc2 pipelined design utilized 9% of the area of the chip at 52 MHz and delivers a result every clock cycle. Design-to-implementation took one day.*

[1] The filter structure was developed in collaboration with Prof. John Villesenor's team at UCLA.

*The K-means clustering algorithm performs classification of features in multi- and hyper-spectral imagery. The input data (pixel and class centers) comes to the FPGA board from the host via a stream. One processor per this algorithm[6] is implemented, utilizing 8% of the area on the chip at a speed of 56 MHz. For this pipelined Streams-C implementation of the kmeans algorithm, we achieve an output every clock cycle. The application took one day of development time.*

*We will compare area, frequency, and development time for these algorithms between the Streams-C approach to manually created hardware implementations, as well as to software implementations.*

*We will also suggest new research problems that we invite other researchers to investigate in the open source sc2 framework.*

## References

[1] M. B. Gokhale, J. Frigo and J. Stone. Parallel C Programming of Reconfigurable Computers: the Streams-C Approach In *HPEC 2000*, Sept. 2000.

[2] Stanford University Intermediate Format. http://suif.stanford.edu.

[3] W. Bohm, J. Hammes, B. Draper, M. Chawathe, C. Ross, W. Najjar. “Mapping a Single Assignment Programming Language to Reconfigurable Systems” In *Journal of Supercomputing*, v. 21, pp. 117–130, 2002.

[4] Celoxica, Inc. www.celoxica.com.

[5] Mary Hall et al. “Defacto: A design environment for adaptive computing technology.” *Proceedings of the 6th Reconfigurable Architectures Workshop (RAW’99)*, 1999.

[6] Jan Frigo, Maya Gokhale, and Dominique Lavenier. “Evaluation of the Streams-C C-to-FPGA Compiler: An Applications Perspective.” In *FPGA 01*, February 2001.

[7] Joseph Arrowood. “Comparison of filter banks for signal detection.” In *LAUR Number 99-4551*, Los Alamos, NM, March 2000.

[8] Miriam Leeser. “Applying reconfigurable hardware to segmentation for multispectral imagery.” In *HPEC 2000*, Boston, MA, September 2000.