

Design Automation for Streaming Systems

Ph.D. Dissertation
Computer Science Division, U.C. Berkeley

Eylon Caspi
eylon@cs.berkeley.edu

12/16/05

Abstract

RTL design methodologies are struggling to meet the challenges of modern, large system design. Their reliance on manually timed design with fully exposed device resources is laborious, restricts reuse, and is increasingly ineffective in an era of Moore's Law expansion and growing interconnect delay. We propose a new hardware design methodology rooted in an abstraction of communication timing, which provides flexibly timed module interfaces and automatic generation of pipelined communication. Our core approach is to replace inter-module wiring with streams, which are FIFO buffered channels. We develop a process network model for streaming systems (TDFPN) and a hardware description language with built in streams (TDF). We describe a complete synthesis methodology for mapping streaming applications to a commercial FPGA, with automatic generation of efficient hardware streams and module-side flow control. We use this methodology to compile seven multimedia applications to a Xilinx Virtex-II Pro FPGA, finding that stream support can be relatively inexpensive. We further propose a comprehensive, system-level optimization flow that uses information about streaming behavior to guide automatic communication buffering, pipelining, and placement. We discuss specialized stream support on reconfigurable, programmable platforms, with intent to provide better results and compile times than streaming on generic FPGAs. We also show how streaming can support an efficient abstraction of area, allowing an entire system to be reused with automatic performance improvement on larger, next generation devices.

Keywords: Streaming, Design Reuse, Concurrency, Hardware Compilation, Reconfigurable, FPGA, System Architecture.

Full text: <http://www.cs.berkeley.edu/~eylon/phd/>