PACT 2005 Tutorial: Network Processor Architecture and Compilation

Luddy Harrison
Associate Professor
Department of Computer Science
University of Illinois at Urbana-Champaign
4108 Siebel Center for Computer Science
201 N. Goodwin Avenue
Urbana, IL 61801
(217) 244-2882
June 21, 2005

Abstract

Network processors are employed in networking gear (e.g., routers and switches) where they are used for packet processing. The packets that make up network traffic are examined by the network processor and properly disposed of: routed to their final destination, filtered and dropped, converted from one protocol to another, prioritized with respect to other packets, and so on. In this tutorial I will describe the architecture of several commercial network processors including the Intel IXP, the programming of these processors, and their compilers. I will give an overview of the hardware and software environment surrounding the network processor, i.e., the interfaces between the network processor and the high-speed networks that carry the network traffic they operate on, and the architecture of the telecommunications gear they are used in. I will discuss the performance challenges faced when programming a network processor for line-rate performance (that is, the ability to keep up with the steady flow of packets arriving at a maximum rate from the network). I will describe how this problem
manifests itself in the programming models for these processors, both at the machine level and in higher-level languages for packet processing. I will then describe the challenge of building a compiler for a network processor, and will describe the architecture of the Intel IXP auto-partitioning C compiler for IXP.

1 Outline

1. Overview of Networking and Packet Processing
   - The Architecture of SONET
   - The Architecture of High-Performance Routers
   - Packet Over SONET (POS) Applications

2. The Architecture of Network Processors
   - The Intel IXP
   - The Cisco Toaster
   - The AMCC nP Family

3. Performance Considerations in Packet Processing
   - Memory Bandwidth
   - Concurrency

4. Programming Models for Packet Processing
   - Machine-Level Models
   - Stream Processing Models
   - IXP C

5. Compiling for Network Processors
   - Pipelining
   - Multithreading
   - Managing Memory Latency
   - Guaranteeing Worst-Case Performance

6. The Design of the Intel Auto-Partitioning C Compiler for IXP
2 Intended Audience

This tutorial is appropriate for those with technical training in the area of computer architecture, machine language, and programming. A background in networking is not assumed.

3 About the Speaker

Luddy Harrison is an Associate Professor of Computer Science at the University of Illinois at Urbana-Champaign, where he teaches computer architecture and conducts research on the design and programming of communication and high-performance computer systems. Prior to joining the University of Illinois in 2004, Prof. Harrison worked as a manager and product architect for Intel Corporation, on compilers for signal and network processors. Prior to that, he founded and managed Connected Components Corporation, which supplied compiler technology to a number of semiconductor and communications businesses, including Motorola, Analog Devices, Cisco, and AMCC.