AutoMapper – An Automated Tool for Optimal Hardware Resource Allocation for Networking Applications on FPGA*

Swapnil Haria#, Viktor Prasanna
BITS, Pilani USC

Introduction

- Ethernet/IP based packet forwarding
  - Complex sequences of lookup operations
  - High throughput, low latency and power consumption desired
  - Need efficient resource utilization of hardware

- Field Programmable Gate Arrays
  - Ideal choice for high-performance networking applications
  - Parallelism, reconfigurability and the abundant on-chip resources
  - Supports efficient implementations of packet lookup engines.

Background

- Lookup Scheme Representation
  - Lookup Flow Graph
  - Sequential arrangement of field nodes
  - Field nodes depict distinct lookup operations
  - Decision nodes enforce conditions leading to different paths

Challenges

- Mapping Problem
  - Generation of a linear pipeline architecture
  - Incorporating all lookup operations in a LFG
  - Optimizing power, throughput or latency
- Ineffective Manual Organization Techniques
  - No guarantee of optimality
  - Time-consuming
  - Wasteful allocation of target resources
  - Difficult to calculate resource costs and latency

Tool Features

- Lookup Schemes
  - Implementation choices

AutoMapper

- Generates input LFG structure
- Identifies distinct paths and location of decision nodes
- Computes the resource costs of implementation choices
- Creates an ILP formulation, corresponding to the mapping problem
- Interprets the optimal solution
- Generates the high-level mapping

Experiments

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Table Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>48 bits, 15,000 entries</td>
</tr>
<tr>
<td>II</td>
<td>64 bits, 20,000 entries</td>
</tr>
<tr>
<td>III</td>
<td>64 bits, 10,000 entries</td>
</tr>
<tr>
<td>IV</td>
<td>64 bits, 1000 entries</td>
</tr>
<tr>
<td>V</td>
<td>64 bits, 10,000 entries</td>
</tr>
</tbody>
</table>

# In terms of clock cycles
* As fraction of the dynamic power in the latency-optimized case

Conclusion

✓ Automapper - automated tool
- Optimally maps complex lookup schemes onto FPGAs
- Maps on to a linear pipelined architecture
- Optimizes for latency, power or throughput

Future Work

- Extend the tool
  - Generate a synthesized implementation for the mapped pipeline in HDL
  - Incorporate mapping of multiple LFCs simultaneously

* Supported by U.S. National Science Foundation under grant CCR-1018801.
# Participation sponsored by the Viterbi-India 2012 Program.

Email: swapnilster@gmail.com, prasanna@usc.edu