Designing a Programmable Wire-Speed Regular-Expression Matching Accelerator

Jan van Lunteren, Christoph Hagleitner, Timothy Heil, Giora Biran, Uzi Shvadron, Kubilay Atasu
Objectives and Challenges

Design Objectives

- A regular-expression *accelerator* and *compiler*, supporting
  1. thousands of regular expressions
  2. high scan rates ~20 Gbit/s
  3. millions of active sessions
  4. incremental and dynamic updates

Design challenges

- Efficient use of available *memory capacity* and *bandwidth*
  - “DFA state explosion” problem
  - high main memory latency (~ 400 cycles)
  - single-cycle cache access required (for high single-stream scan rates)
Solution and Results

Programmable RegX Accelerator

- Architecture comprising fast *programmable state machines*, simple *processing elements* and a software-managed *L1 cache*
  - Functionality available as a set of single-cycle primitive functions that can be programmed by a sophisticated compiler in a *RISC* fashion
  - Instruction *dispatch* and *execution options* allow maximum scan rate to be sustained without back pressure and independently of input characteristics

- Compiler stack

PowerEN™ RegX Implementation

- *SOC* design in *45-nm* SOI, clocked at 2.3 GHz
- RegX: 15.4 mm² (chip size: 410 mm²)
- Theoretical peak scan rate: *73.6 Gbit/s*
- Measured: *15 to 40 Gbit/s* for typical workloads