



Dynamic Hardware Plugins: Exploiting Reconfigurable Hardware for High-Performance Programmable Routers

David Taylor, Jonathan Turner, John Lockwood
Applied Research Lab, Washington University in St.
Louis

Outline

- Why are programmable routers needed?
- Related Works
- Architecture
 - Programmable Router
 - Port Processor
 - Dynamic Hardware Plugins (DHP)
- Implementation and Limitations
- Current Work



Motivation

- Routers need to be able to handle flow-specific processing at optical line speeds without high per-port costs
- Software processing is flexible while ASIC solutions offer high performance
- Applications that use low data rates could be run in software but many applications would perform better in hardware
- FPGA / software solution could offer the desired platform

A teal circular graphic is partially visible on the left side of the slide, overlapping the edge.

Case for Using FPGAs

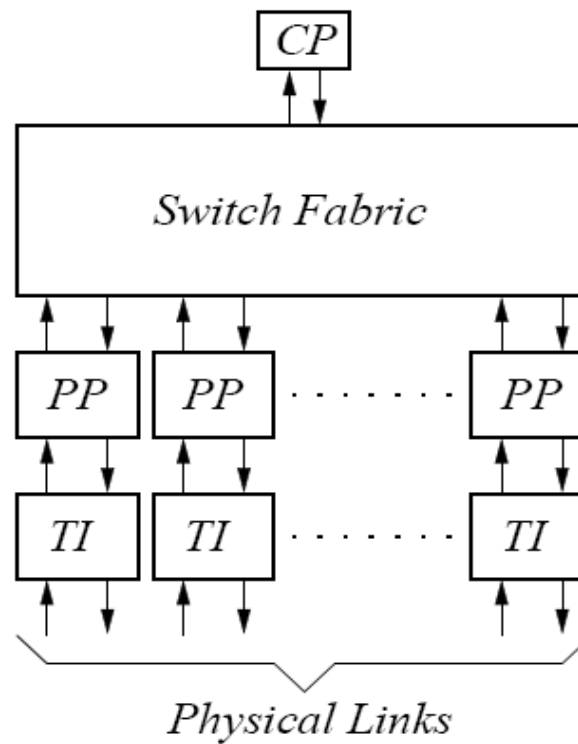
- Over 1 million logic gates
- Internal clock speeds of 200 MHz
- 100KB on-chip (SRAM) memory
- Partial Reconfiguration



Previous Work

- *Wolf, T., Turner, J.: "Design Issues for High Performance Active Routers"* – Proposes Software environment for this platform
- Other work in reconfigurable network hardware does not provide memory resources or scalability to multi-port routers

Generic Programmable Router





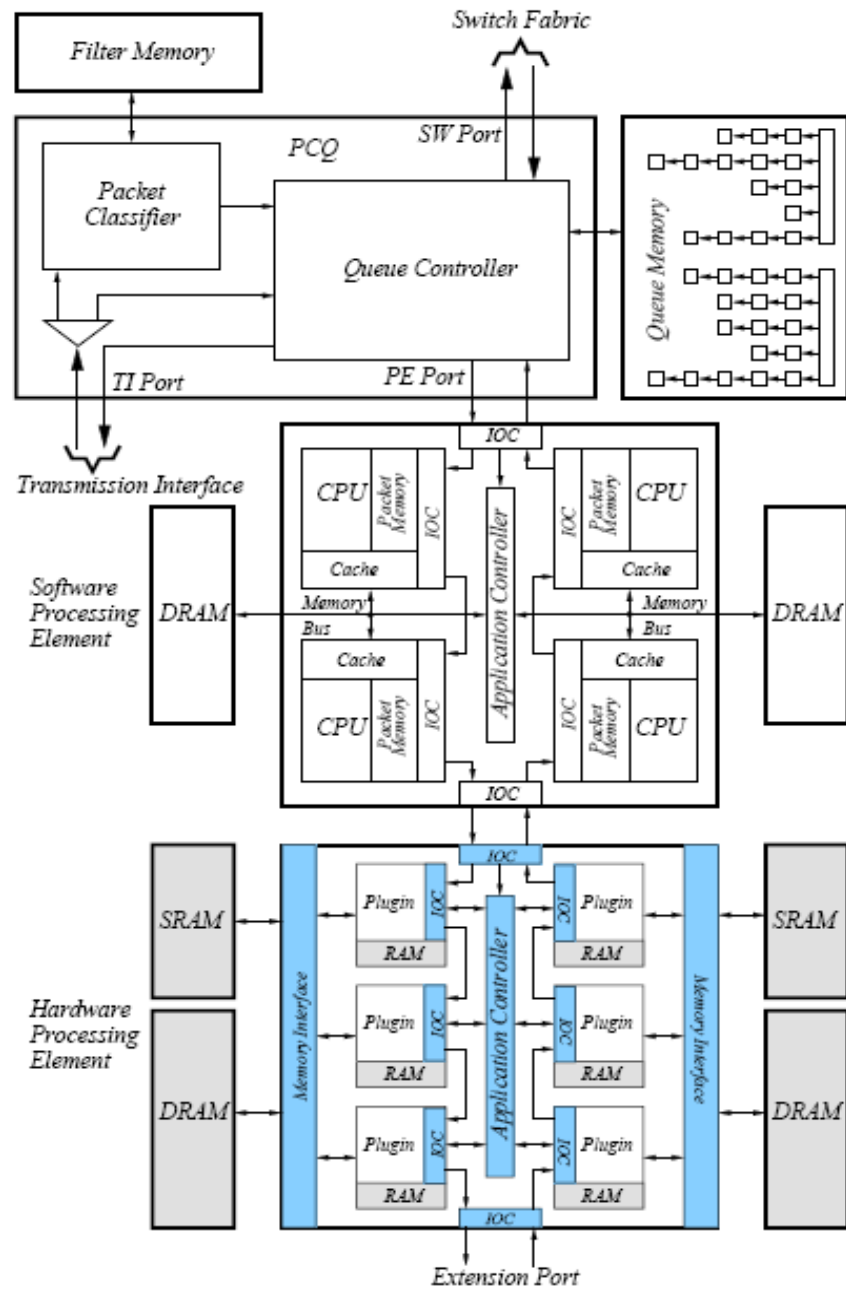
Programmable Router Components

- Switching Fabric – ten to 1000s of ports with link rates of 2.4 Gb/s
- Transmission Interface (TI) – converts data to standard router input format
- Port Processor (PP) – Performs all flow classification, processing, etc...
- Control Processor (CP) – External control, manages flows on Port Processors; can be shared-memory multiprocessor



Port Processor Architecture

- Packet Classification and Queuing
- Software Processing Element
- Dynamic Hardware Plugin (DHP)
 - Infrastructure
 - Hardware Plugins



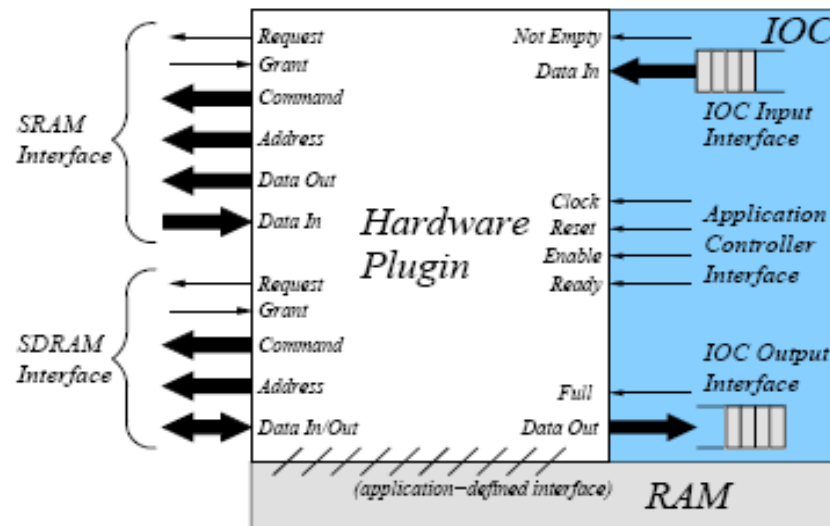


DHP Architecture

- Infrastructure
 - Data I/O and Flow Control
 - Application Controller
- Application Controller
- Memory Interfaces

Hardware Plugin Interface

- Uses standardized API
- IOC, SRAM, DRAM, Application Controller Interface



Implementation and Testing

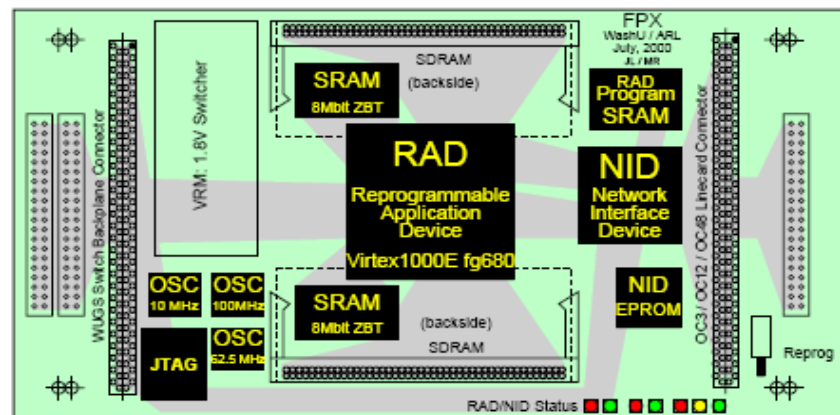
- AES encryption using Rijndael algorithm
- Software approach – 31.64 Mb/s
- ASIC approach – 5.16 Gb/s
- FPGA (Virtex 3200E)
 - Iterative approach – 353 Mb/s (20% resources used)
 - 5-stage partial pipeline – 1.94 Gb/s (40 % resources used)

Results

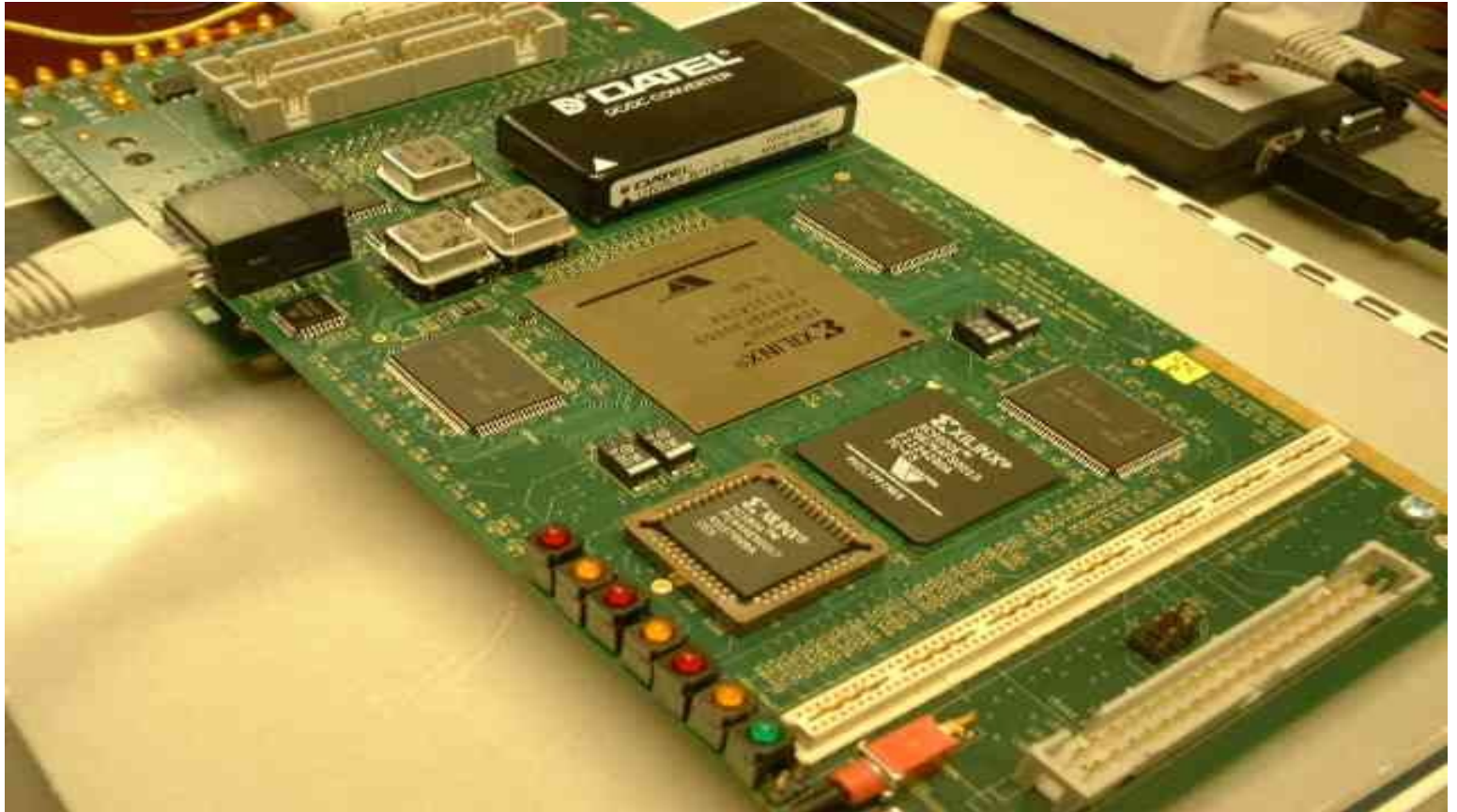
- Even the largest FPGAs cannot match ASIC performance
- Dynamic partial reconfiguration does not work (as of 2000)
- Preliminary results show promise for processing streaming data
- Using a mixed ASIC/FPGA platform could improve throughput

Prototype Testing

- Field Programmable Port Extender Contains 2 FPGAS – the NID and RAD
- WUGS 20 8 port ATM switch is used for switching fabric (2.4 Gb/s link rates)

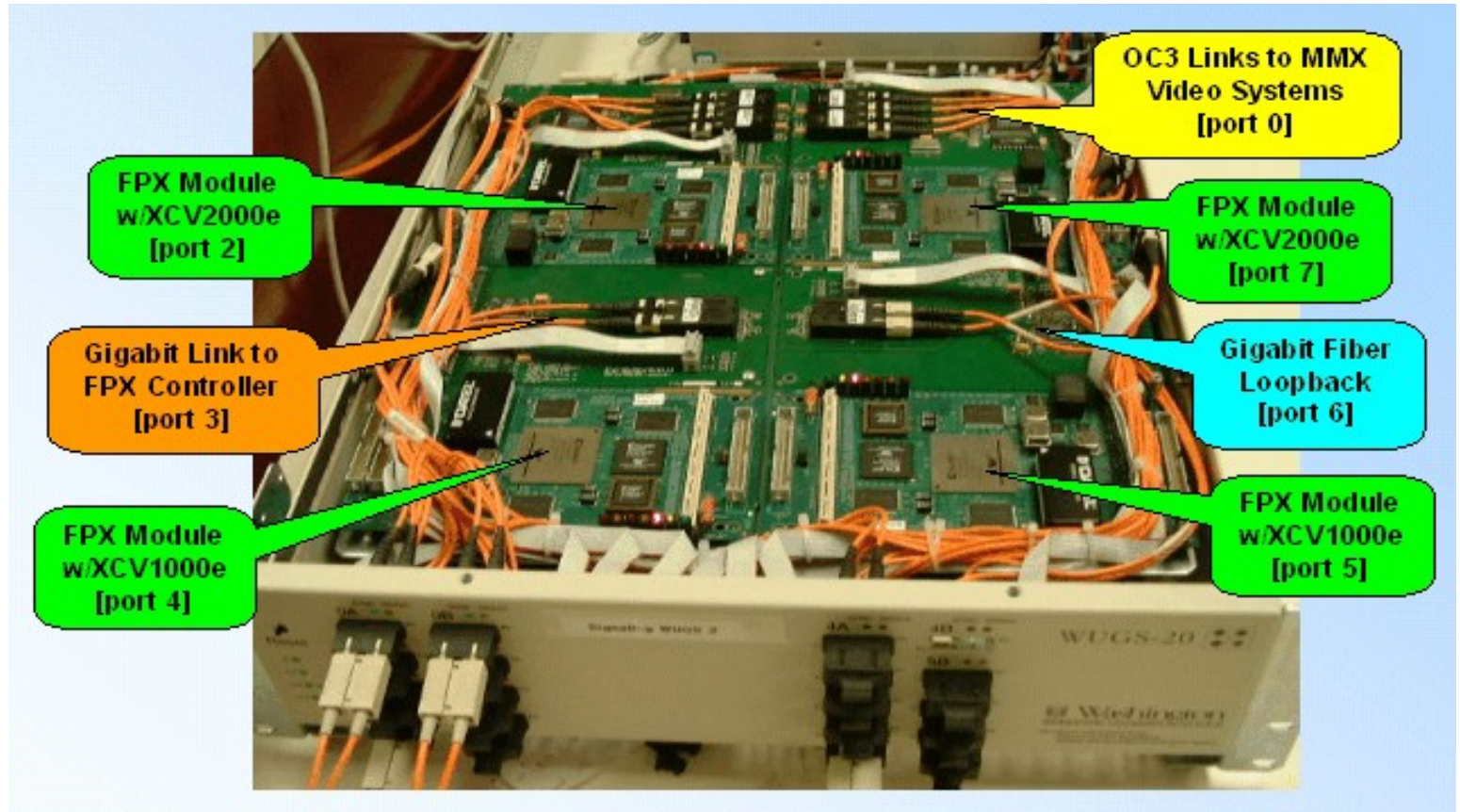


FPX in Action



<http://www.arl.wustl.edu/projects/fpx/reconfig.htm>

FPX in WUSG switch





Current Research

- DAC 2002 - Dynamic Hardware Plugins in an FPGA with Partial Runtime Reconfiguration
- 85 FXP boards fabricated and tested with compression, encryption, video and image processing