assorted projects

Robert Olsson

NAPI

NAPI. Robust driver API

Overall Effect

Inelegant handling of heavy net loads

System collapse

Scalabiity affected

System and number of NICS

A single hogger netdev can bring the system to its knees and deny service to others

March 15 report on lkml

Thread: "How to optimize routing perfomance"
reported by Marten.Wikstron@framsfab.se
- Linux 2.4 peaks at 27Kpps
- Pentium Pro 200, 64MB RAM

NAPI

Work with the legends Alexey Kuznetsov, Jamal Hadi Salim (2001-11-10).

Paper: "Beyond softnet for usenix.

Three years of work, paper written in three hours

Kernel inclusion major impact...
Today ALL hi-perf drivers are NAPI-based.

https://en.wikipedia.org/wiki/New_API

NAPI

First NAPI driver Alexey
Tulip (DEC) 100 Mbps

Second (Intel) e1000 GIGE by me..
MIT Click-Os challenge.
Different design wrt of interrupt acking. UFO.
“use HW interrupt just as signals” - softirqd

Result: Very good connections with Intel.

fib_trie

One of the most researched areas in Internet and networking. Many proposals.

IPv4 routing algo in Linux. Algo orginally by Stefan Nilsson, Gunnar Karlsson.

fib_trie can be own seminar and was at UU and CSC (Olof Hagsand)

Now the major lookup, no fib_hash, no route cache just fib_trie.

fib_trie

Garner group 1 Billion Android mobiles. 2014
So between 1-2 billion users of this work.

Intel Academic Award 2005. with co-authtors Jens Låås and Hans Liss.

Now more work is needed. We can discuss...

pktgen

In kernel testing tool...
Has improved network performance.

Challanged by netmap dpdk.

Pktgen recently improved the driver API
Superior small pkts TX performance. Including pktgen performance. Wirespeed with small pkts at 10G Intel 82599 NIC.

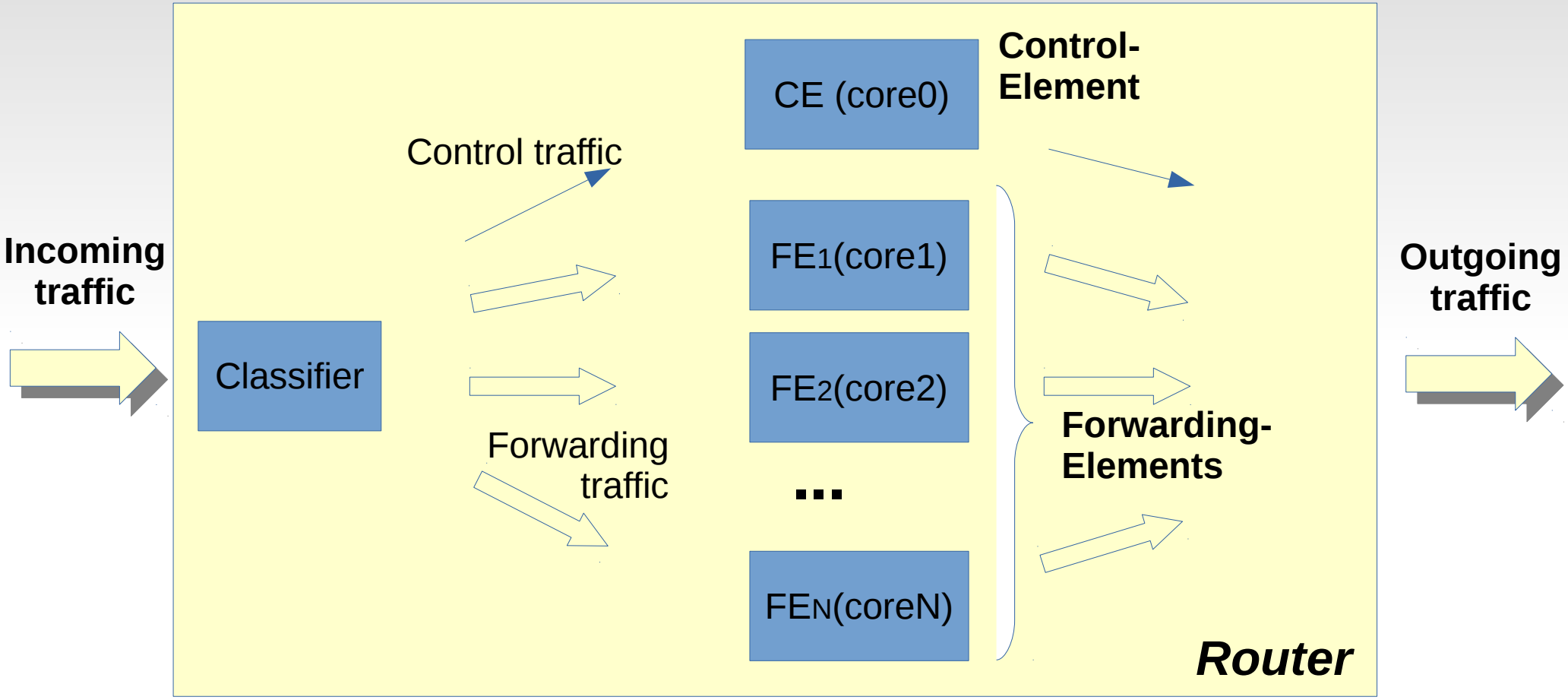
Jesper @ REDHAT (Some times at Bifrost workshop)

Linux hi-pref routing

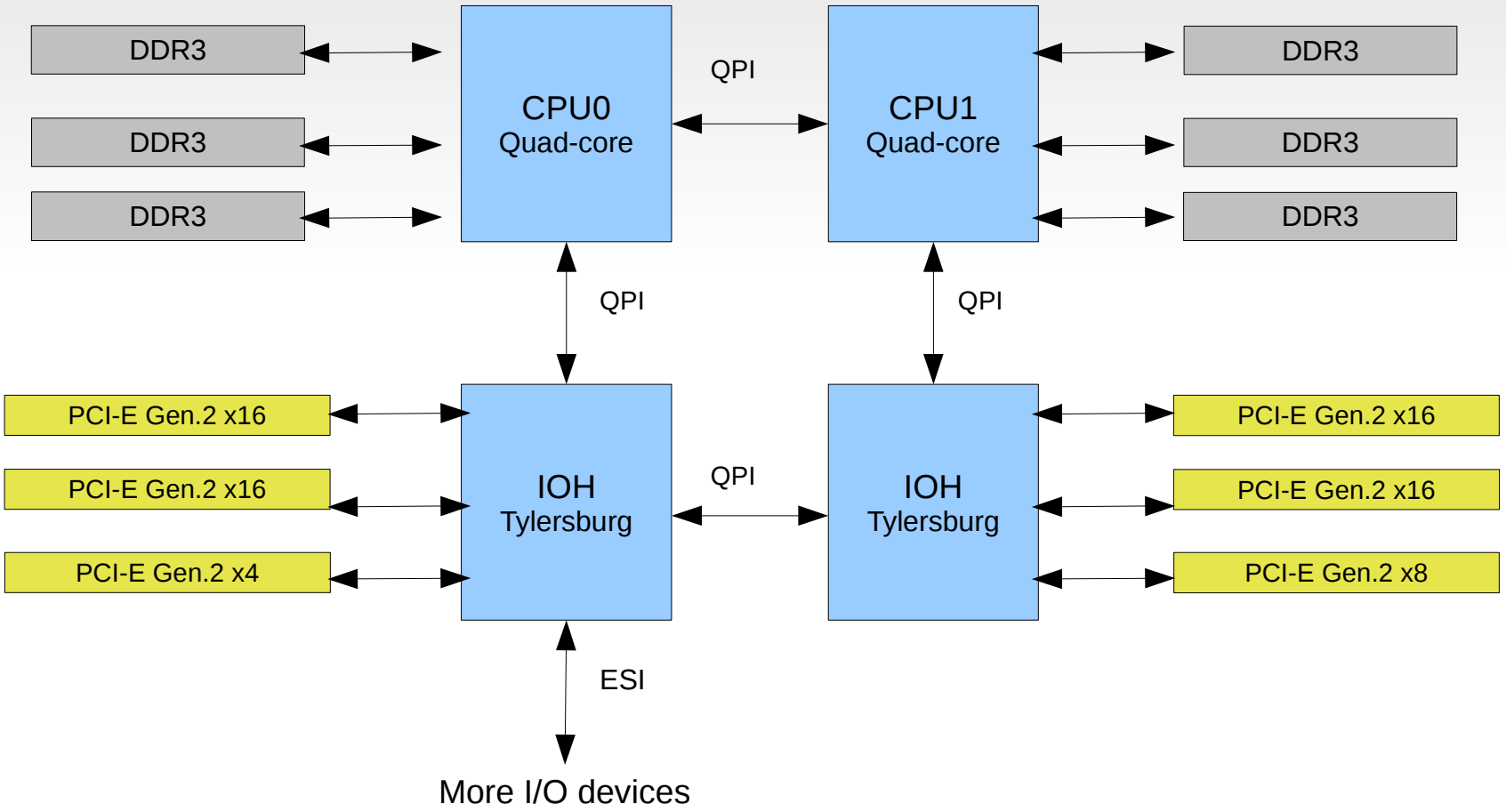
HW classifier in NIC (netchannels)
Multi-queue (virtualization)
RSS Receiver Side Scaling
MSI Message Signaled Interrupts
    PIC, APIC, IOAPIC, MSI
Flow Director
Interrupt Affinity

Flow separation in box
Minimize cache impact
Minimize lock contention
Eliminate packet reordering

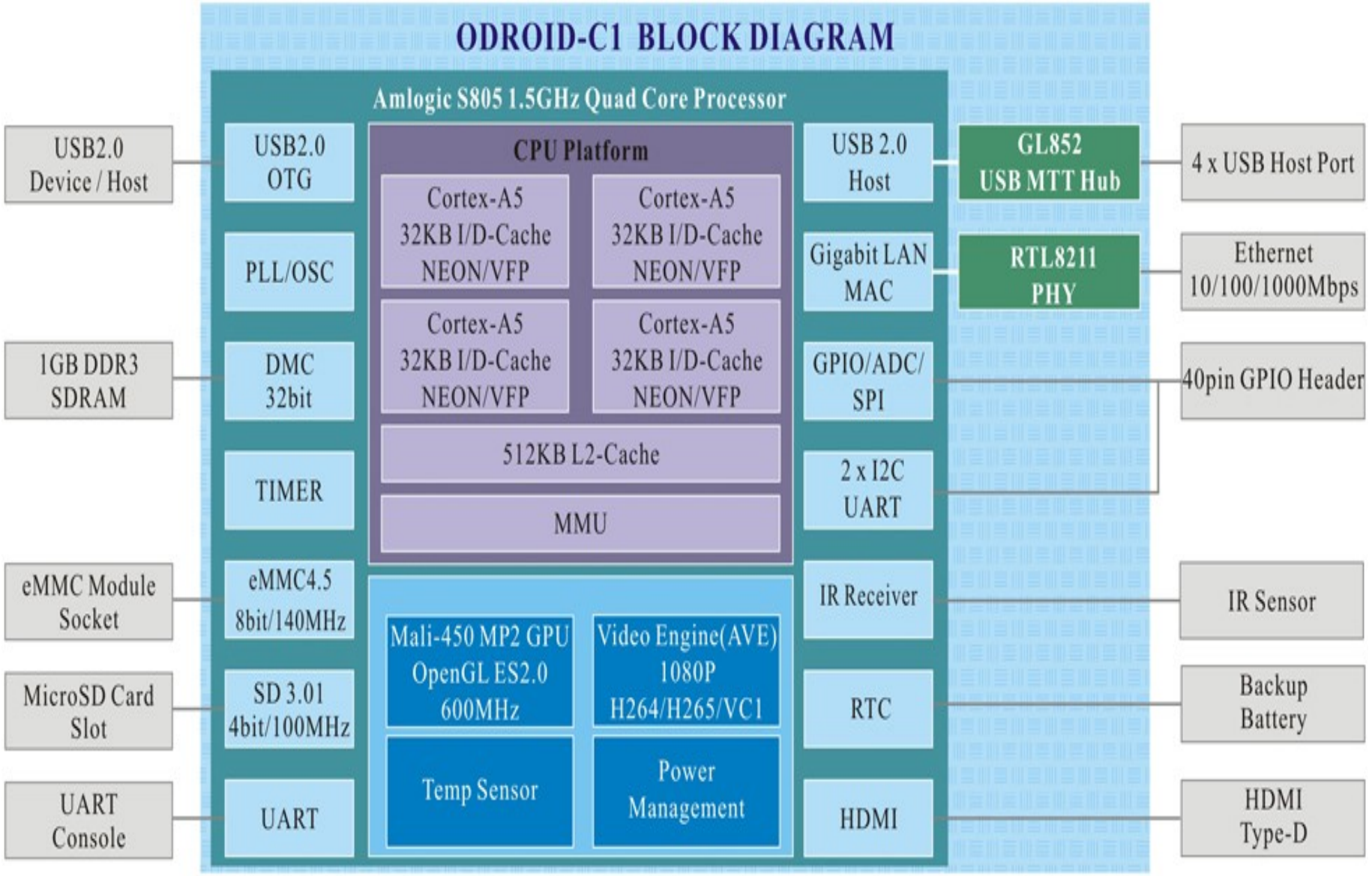
Control-plane separation on a multi-core



Block hardware structure



Odroid C1 arch



N-tuple or Flowdirector

ethtool -K eth0 ntuple on

ethtool -U eth0 flow-type tcp4 src-ip 0x0a0a0a01 src-ip-mask
0xFFFFFFFF dst-ip 0 dst-ip-mask 0 src-port 0 src-port-mask 0
dst-port 0 dst-port-mask 0 vlan 0 vlan-mask 0 user-def 0
user-def-mask 0 action 0

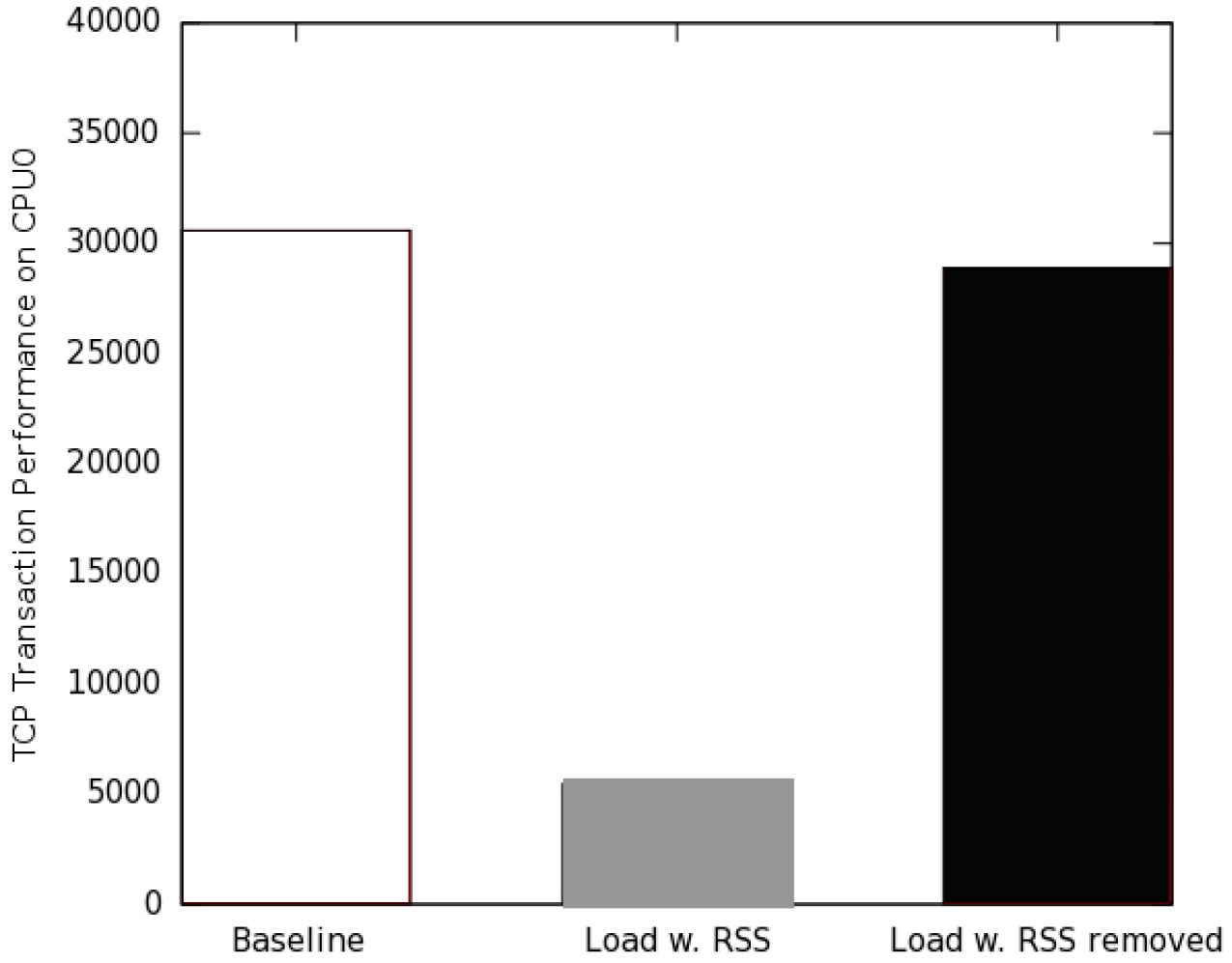
ethtool -u eth0

N-tuple is supported by SUN Niu and Intel ixgbe driver.

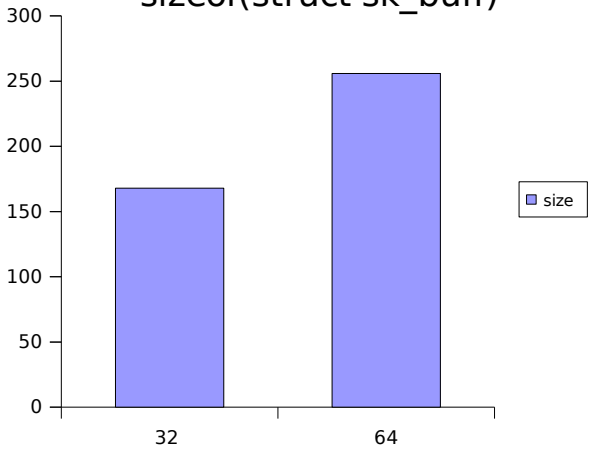
Actions are: 1) queue 2) drop

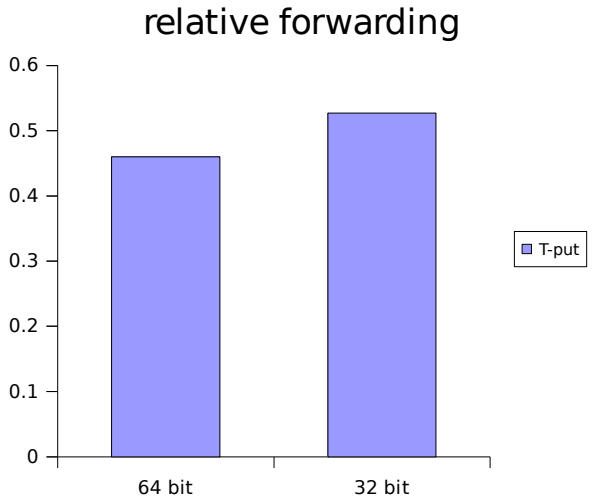
But we were lazy and patched ixgbe for ssh and BGP to use CPU0

Transaction latency using flow separation



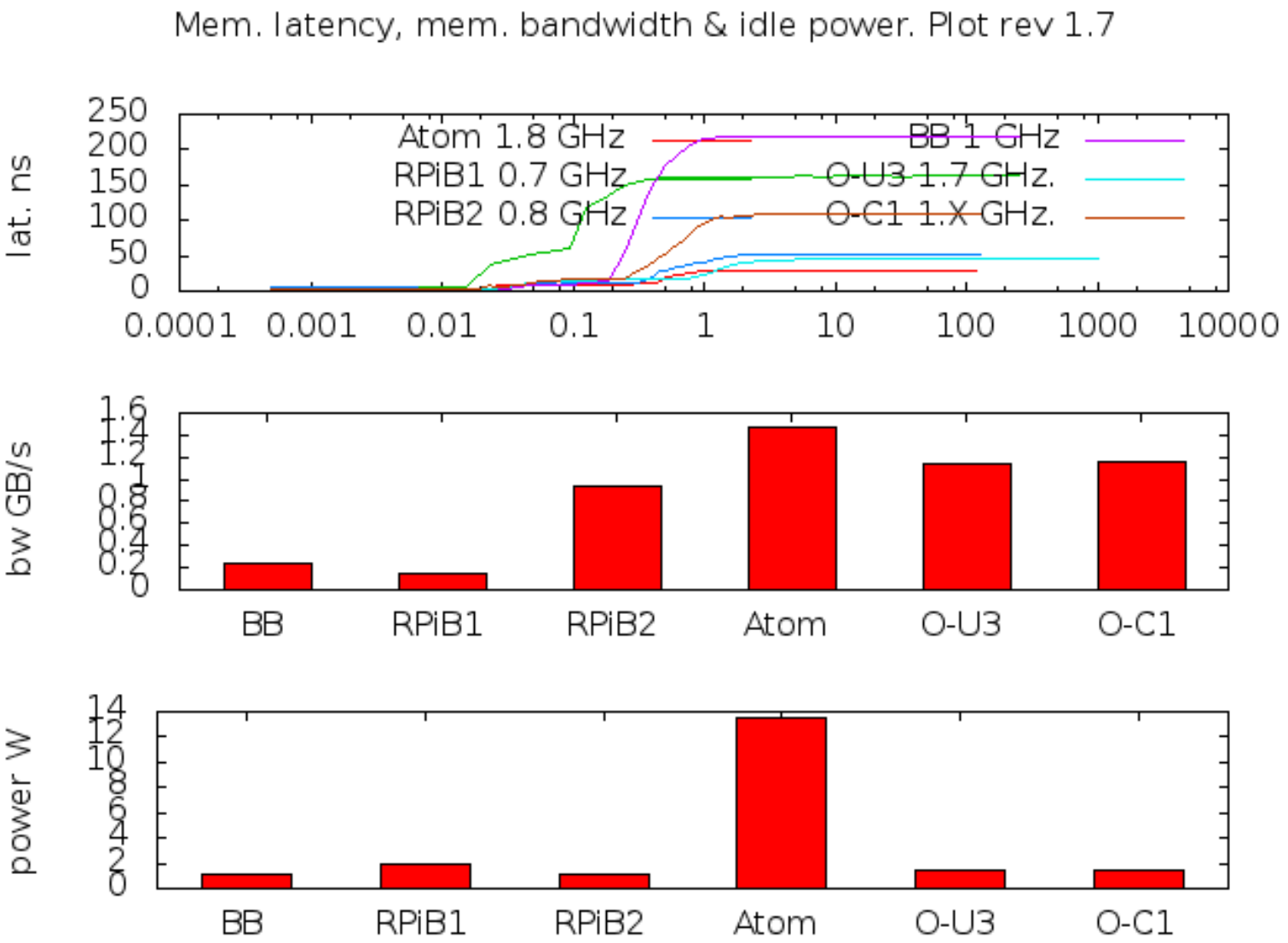
32/64 bit || sizeof(sk_buff)





Gcc 3.4 x86_64 vs i686 on same HW

Performance & Efficiency



Broadwell X10SDV-TLN4F 45W

Intel® Xeon® processor D-1540,

Single socket FCBGA 1667;

8-Core, 45W 2. System on Chip 3. Up to 128GB ECC RDIMM DDR4

2133MHz or 64GB ECC/non-ECC

UDIMM in 4 sockets 4. Expansion slot: 1x PCIe 3.0 x16

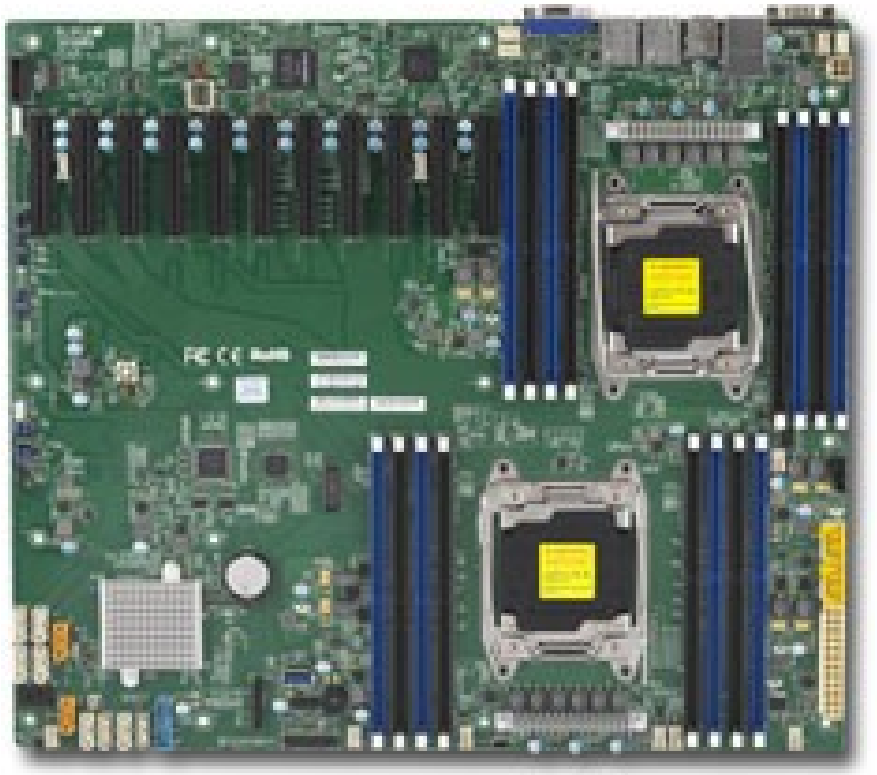
M.2 PCIe 3.0 x4, M Key 2242/2280 5. Dual 10GbE LAN and Intel® i350-AM2

dual port GbE LAN 6. 6x SATA3 (6Gbps) ports via SoC 7. 2x USB 3.0 ports (rear)

4x USB 2.0 ports (via headers) 8. 1x SuperDOM, 1x COM, TPM 1.2

header, GPIO and SMbus headers 9. 12V DC input and ATX Power Source

X10DRX 10 PCIx8 (3.0) slot



1. Dual socket R3 (LGA 2011) supports Intel® Xeon® processor E5-2600 v3 family; QPI up to 9.6GT/s
2. Intel® C612 chipset
3. Up to 1TB ECC DDR4 2133MHz; 16x DIMM slots
4. Expansion slots: 10 PCI-E 3.0 x8 and 1 PCI-E 2.0 x4 (in x8) slot
5. Intel® i350 Dual port GbE LAN
6. 10x SATA3 (6Gbps); RAID 0, 1, 5, 10
7. Integrated IPMI 2.0 and KVM with Dedicated LAN
8. 5x USB 3.0 ports, 4x USB 2.0 ports
9. 2x SuperDOM ports, TPM 1.2 header

TRASH – Trie HASH

Flow lookup algo. with StefanNilsson (fib_trie)
Common lookup in Linux

Implemented, Deployed in UU routers and KTH
Wire tapping.

IEEE High Performance Switching and Routing, 2007. HPSR '07.

Cache: Core i7 Xeon 5500 Series

Data Source Latency (approximate)

L1 CACHE hit, ~4 cycles
L2 CACHE hit, ~10 cycles
L3 CACHE hit, line unshared ~40 cycles
L3 CACHE hit, shared line in another core ~65 cycles
L3 CACHE hit, modified in another core ~75 cycles
remote L3 CACHE ~100-300 cycles

Local Dram ~60 ns
Remote Dram ~100 ns

TLB: Core i7 Xeon 5500 Series

TLB is Translation Look Aside Buffer

The TLB is a seperate very small cache of the virtual address to physical address mappings.

Effect needs to studied.

IO bus latency – huge

mmio_test -- A simple NIC latency tool

Abstract
This is code to measure latency from various NIC's. It maps chip registers and reads latency. Of course bus latency atc are included.

https://github.com/herjulf/mmio_test

Method

Successive approximation

Lab test
Profile
Read code

Current netdev work

Speed up memory subsystem including packet memory

SLAB (from SUN)
SLUB (fom IBM)

Private pools for recycling. Avoid if possible.

Thanks!

Questions?

Bifrost

Workshop. Host?

Our lab facilities

Now at Uppsala University

UU now low activity