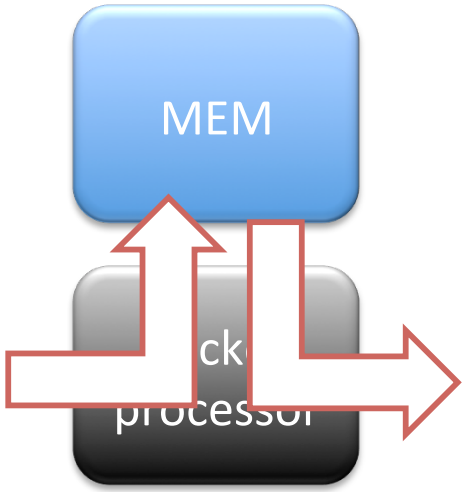
Memory, multi-100Gbps interfaces and their impact on network design

Rafal Szarecki

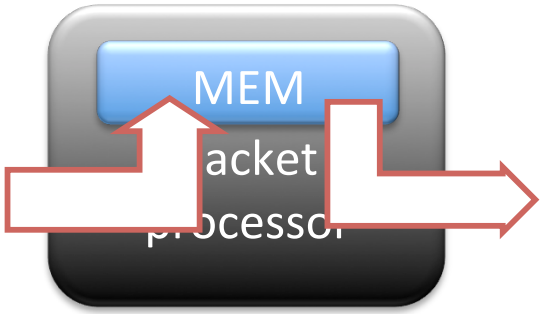
Juniper Networks

Problem

- 100GE is here. We are heading 400GE
  - 250+ Mpps sequential processing for lookup/match. Multiple memory access per packet.
  - 10's Gbps Read lookup result.
  - 800Gbps Memory (netto + ECC) for Packet Buffer. (WR+RD)
- What is that memory?



On-Chip memory

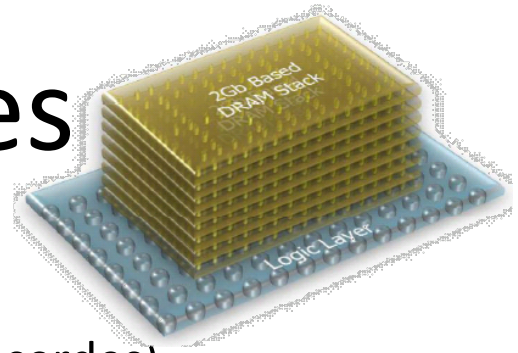


- Fast, SRAM-type
- Small in size
  - Small FIB (100k-200k entries)
  - Shallow DBB – ~100 micro-seconds or less
- No need for power board lines
- Simple system design – no signal integrity issues.

Off-chip - DDR4

- 2.4Gbps (today, 3.2 max in future) per pin.
- To make 800Gbps → 336 pins /42 Bytes wide – not power of 2 → 512 pin (64 Bytes wide bus).
- Each pin drain power
- Each pin adds complexity to signal integrity and board real estate design.
- Wide bus is not good for lookup memory – many of memory access to return only pointer 2-4B of usefull data out of 64B

Off-chip - alternatives



- Hybrid Memory Cubes (HMC)
  - Proprietary
  - Fewer serdes running at high data rate (1.28Tbps agg – 48 serdes)
  - Much less issues with SI, space, traces routes. Less power.
  - Production - 2014
- High Bandwidth Memory
  - Wide interface (a lot of parallel lines) – SI, power and board real-estate issue.
  - To be packaged together w/ processor (ASIC). Use TSV for massive parallel connections
  - Production - 2015
- Both are “3D” memory – DRAM die stacked one on the another.
- Both are expensive and not in economy of scale camp.
- So fare limited capacity

Impact on network gear

- Type A
  - Low scale FIB,
  - shallow buffer (10s of uSec)
  - Low cost
  - Made base on packet processors w/ on-chip memory
- Type B
  - Full FIB (2-4M+ as today)
  - Deep DBB (10s of mSec). QoS queus, DiffServ Scheduling, etc
  - Significantly higher cost.
  - Made base on packet processors w/ off-chip memory

Network Design (1)

- Knowledge what traffic is on top of your network is key.
  - OK for limited FIB?
  - OK to relay on application level to deal with losses?
    - SLA KPI defined at application level – user experience.
    - Control application and/or OS (e.g. TCP tunning, App loss detection and retransmission)
  - OK to overprovisioning Capacity?
    - Physical media (FO) and interface (-SR laser) need to be cheap.
    - Runs well below link speed even during failure. Prevent Micro-burst do fill shallow buffer.
    - All traffic is premium – QoS scheduling not so efficient – individual queue is just few packets.
    - 50% link fill rule not applicable – should be less.
- IF yes then type A is fine.

Network Design (2)

- Otherwise Type B
- It will cost you more on Network device
  - Memory is going to be expensive.
  - It will drain more power (then type A)
  - More chips is more chips
- Save on media acquisition - FO rental.
  - Can drive each link to 100%+ during failure.
  - QoS Scheduling allows to manage and mitigate impact of packet loss and delay.
  - 50% link fill rule not applicable – could and should be more for better economics.