A 10Gbps Ethernet TCP/IP Processor

HOT Chips 15 - August 17, 2003

Jianping Xu, Nitin Borkar, Vasantha Erraguntla, Yatin Hoskote, Tanay Karnik, Sriram Vangal, Justin Rattner

Microprocessor Research, Intel Labs

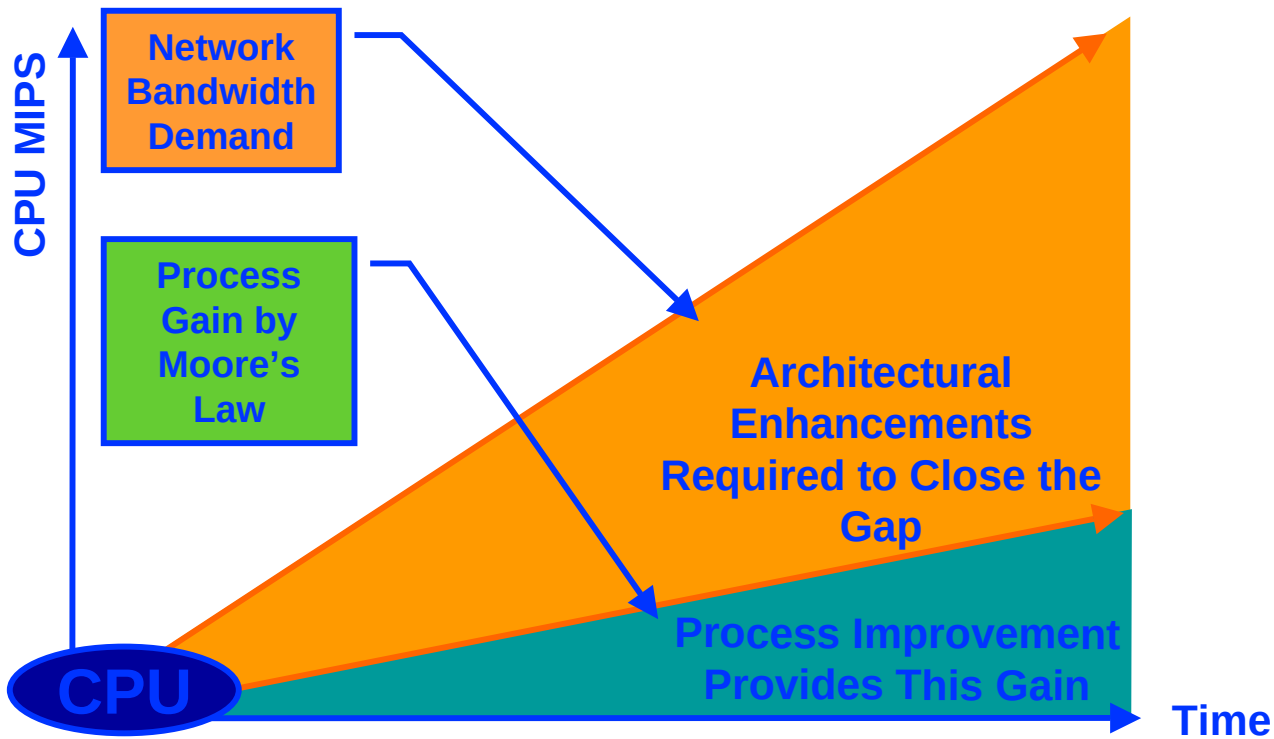
Outline

- TCP/IP processing challenges
- TCP/IP protocol processor (TIPP) features
- TIPP microarchitecture
- TIPP implementation
- Experimental chip performance
- Summary

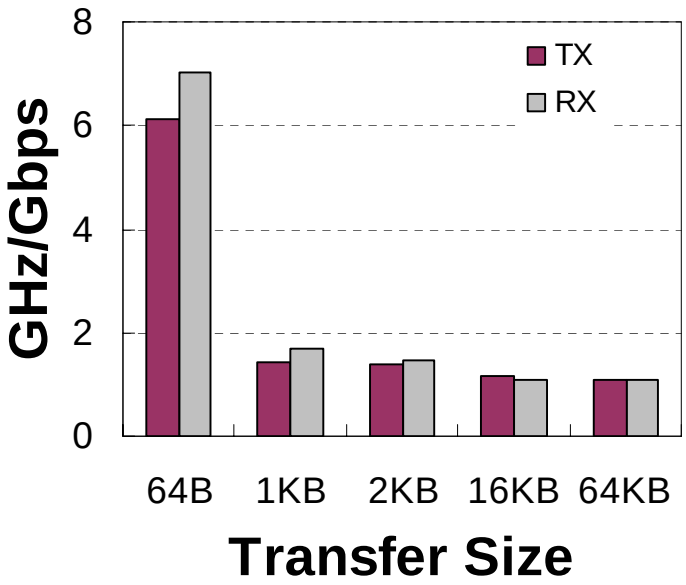
TCP/IP Challenges

100MbE	148 k pkts/sec	6.72 ms
1GbE	1.48M pkts/sec	672 ns
10GbE	14.8M pkts/sec	67.2 ns
40GbE	59.5M pkts/sec	16.8ns

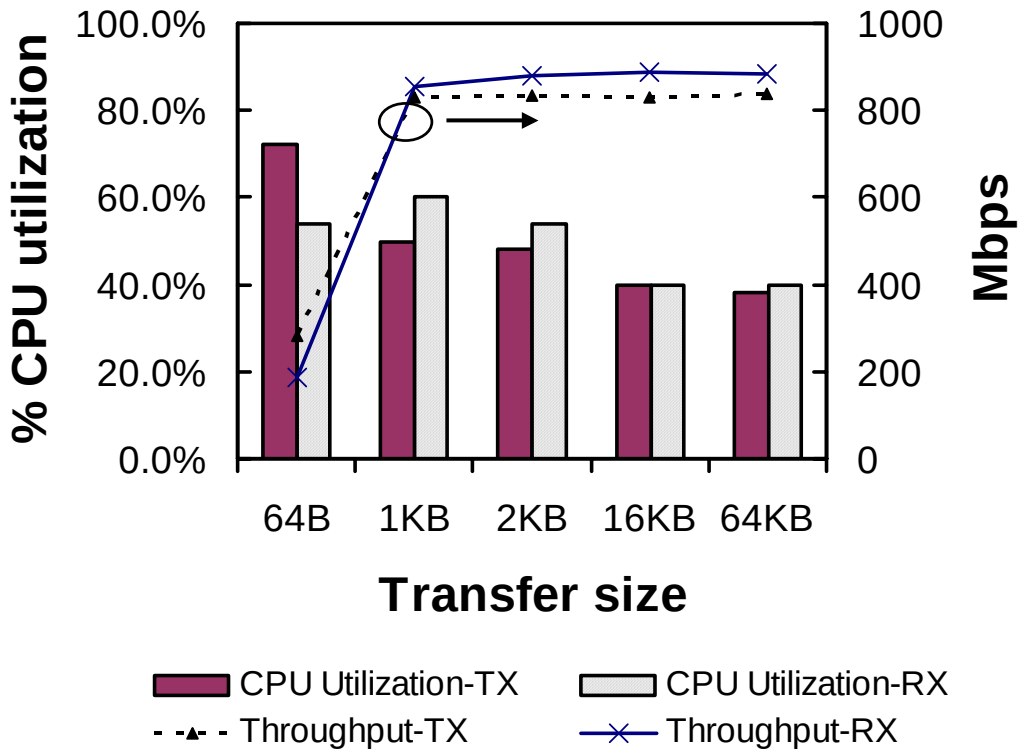
General processor cannot keep up.



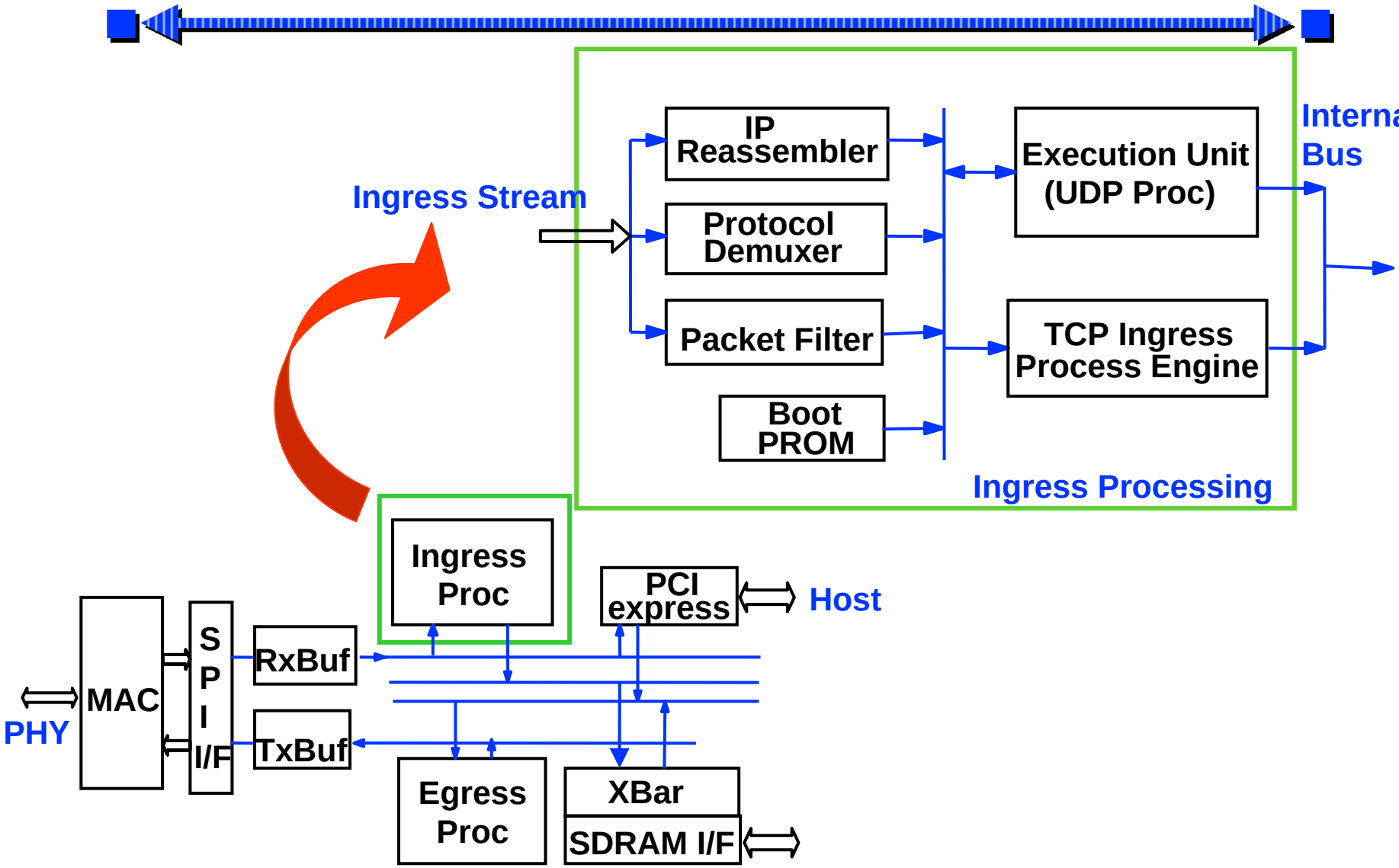
CPU Requirements



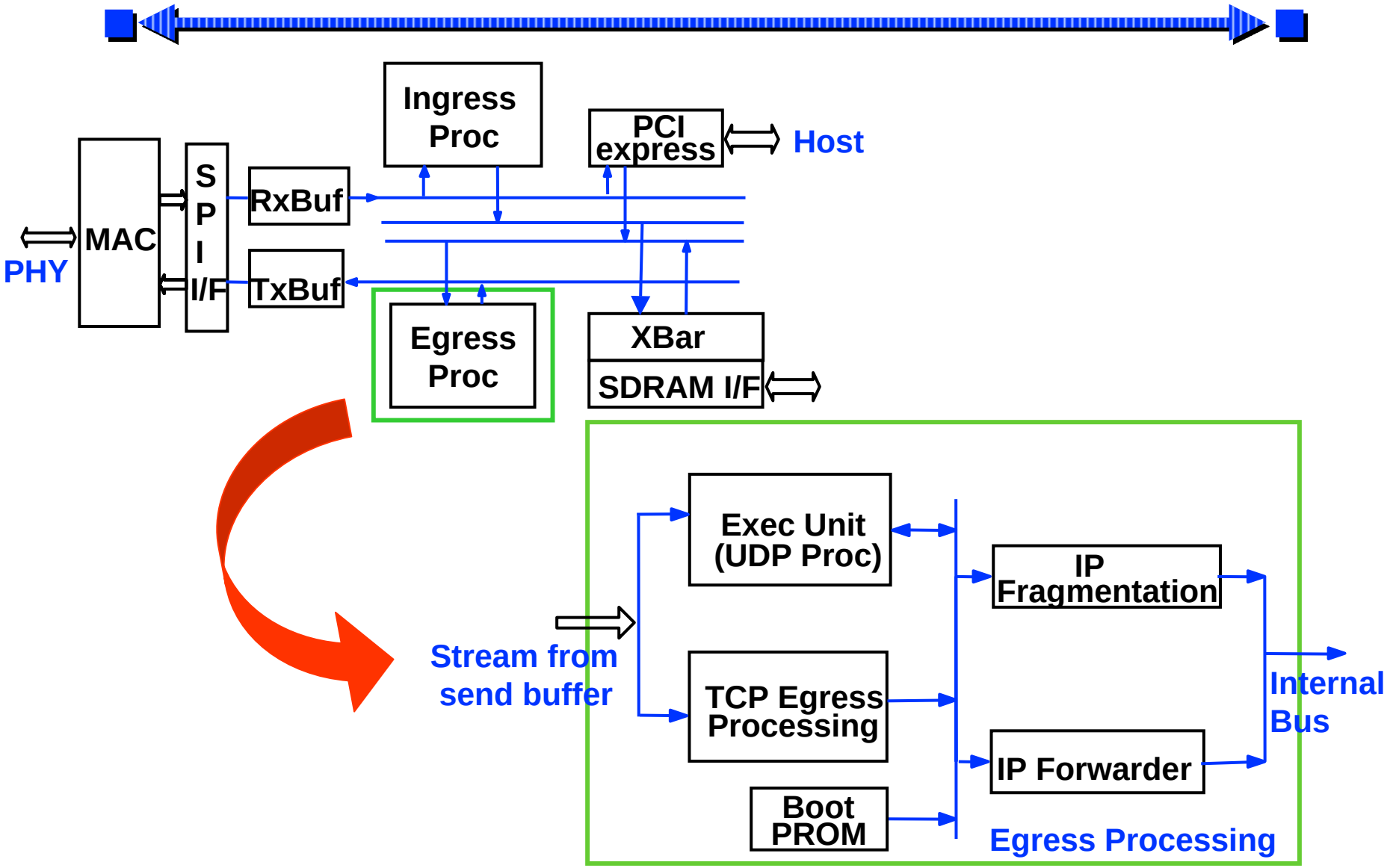
Throughput & CPU Utilization (2.4GHz)



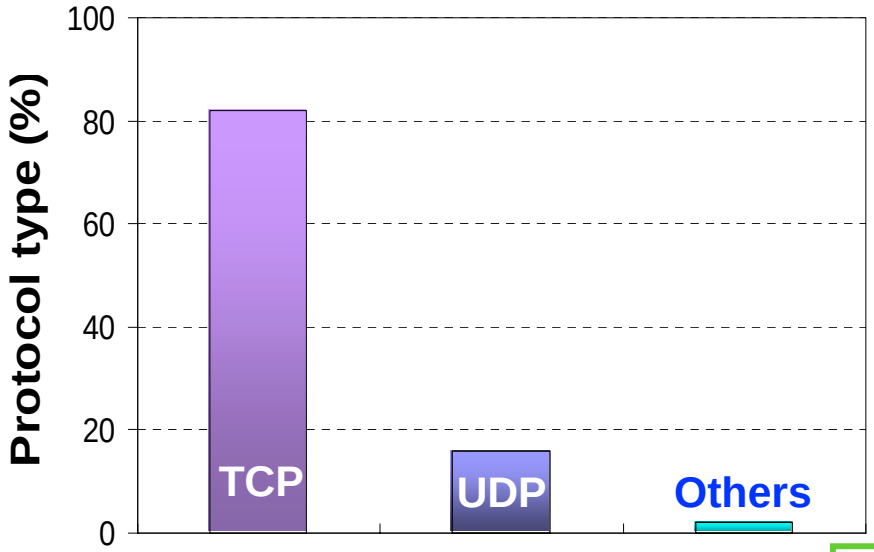
Ingress Processing



Egress Processing

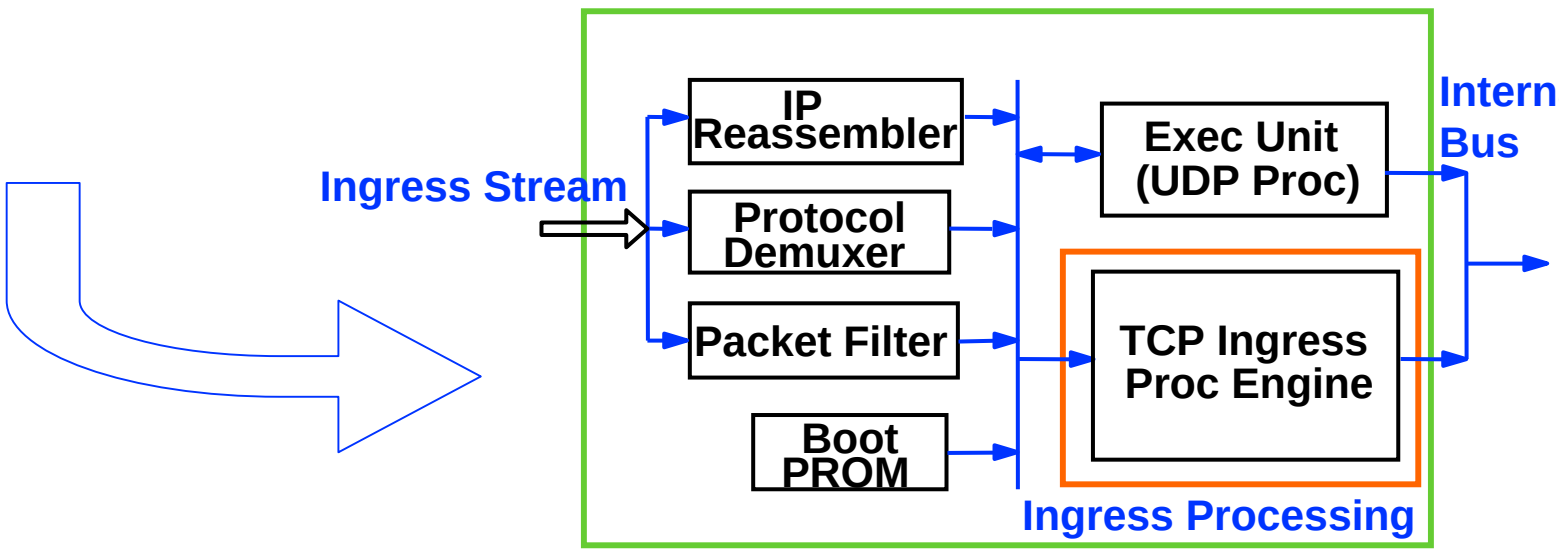


TCP Offload



Why TCP processing?

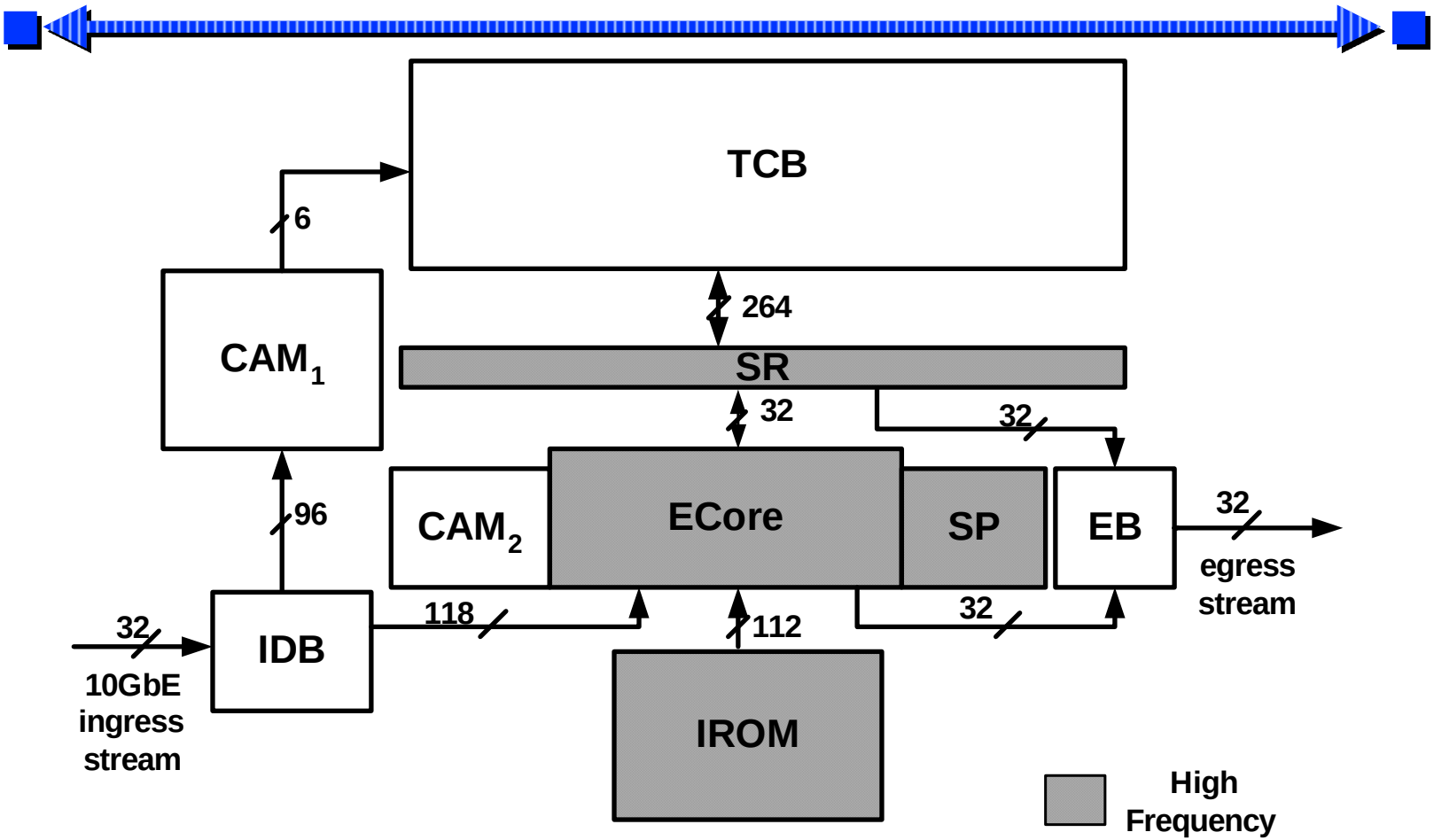
- **82% of all traffic is TCP**
- **TCP processing is the bottleneck**
  - **complex**
  - **compute intensive**



Key Features of TIPP

- **Special purpose processor**
  - **Dual frequency, low latency, buffer-free design**
  - **High frequency execution core**
  - **Accelerated context lookup and loading**
- **Programmability for up-to-date protocols**
  - **Programmable design with special instructions**
  - **Rapid validation and debug**
- **Scalable solution**
  - **Across bandwidth and packet sizes**
  - **Extendable to multi-core solution**

TIPP Organization



CAM — Content Addressable Memories
TCB — Transmission Control Block
ECore — Execution Core
IROM — Instruction ROM
IDB — Ingress Dispatcher Block
SR — State Register
SP — Scratch Pad
EB — Egress Buffer

Instruction Set

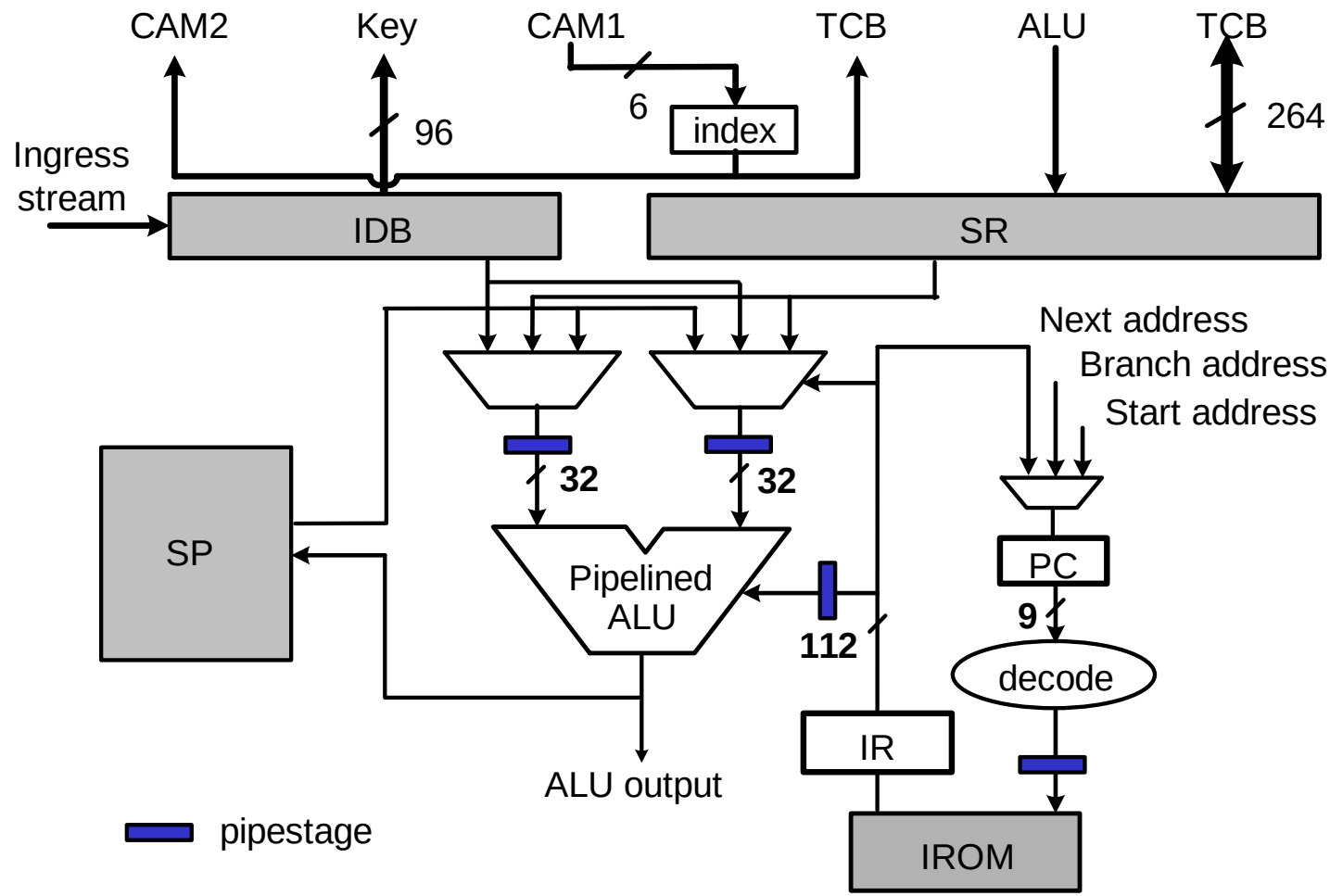
General purpose instructions

1	LOAD	A ← data
2	MOV	A → B
3	AND	A B → cond
4	OR	A B → cond
5	ADD	A B → C
6	SUB	A B → C
7	CMP	A B → cond
8	EQUAL	A B →cond
9	NOT	A → C
10	BREQZ / BRNEQZ	label
11	JMP	label
12	SHL2	A
13	NOP	

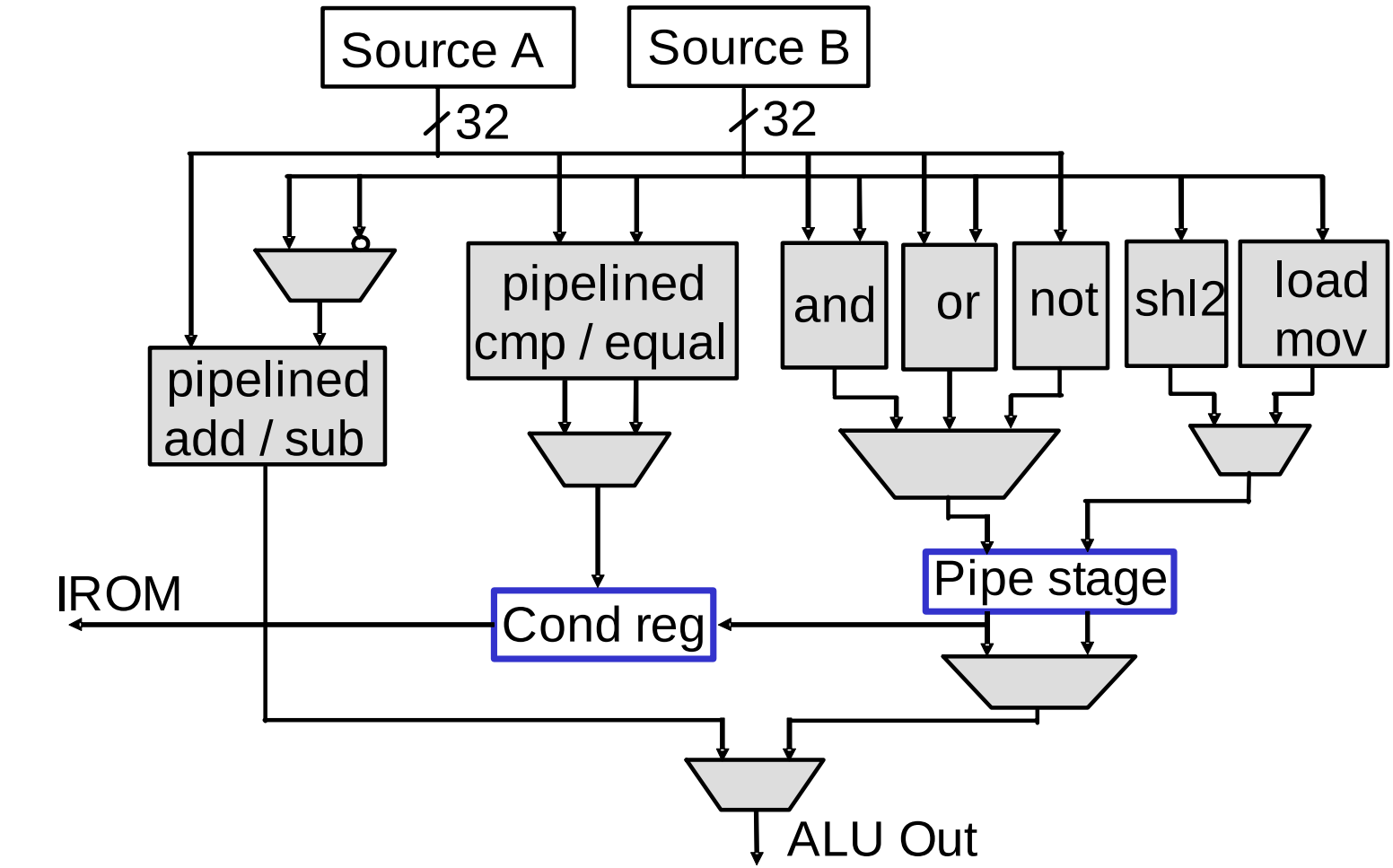
Special purpose instructions

1	CAM1CLR	index
2	CAM2CLR	index
3	CAM1LKUP	key → index, data
4	CAM2LKUP	key → index, data
5	CAM2WR	key ← data
6	CAM2EMPTY	→ cond
7	TCBWR	index ← data
8	TCBRD	index → data

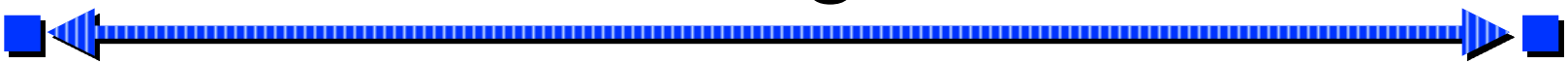
High Speed Execution Core



ALU


Source A
32
Source B
32
pipelined
add / sub
pipelined
cmp / equal
and
or
not
shl2
load
mov
IROM
Cond reg
Pipe stage
ALU Out

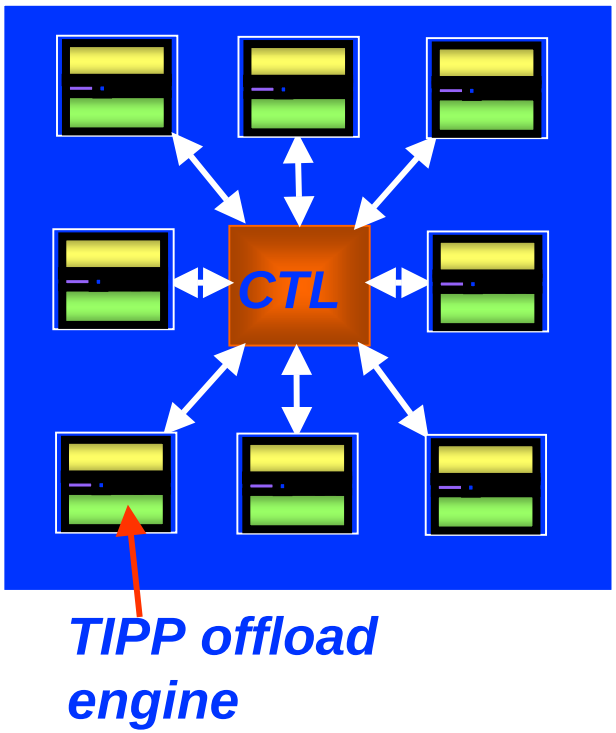
Packets Reordering



- Packets arrive in random order
- Conventional software solution uses sorting to reorder packets
- Hardware sorting is expensive and cumbersome
- Use CAM to eliminate sorting
  - Store and lookup of packets in CAMs by sequence number
  - Combine adjacent packets in CAMs

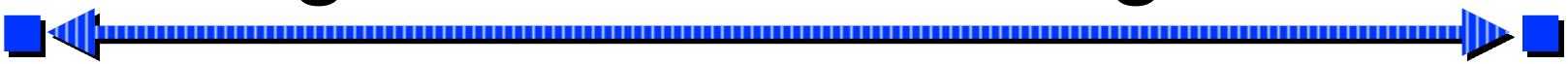
Critical for wire speed processing

System Scalability



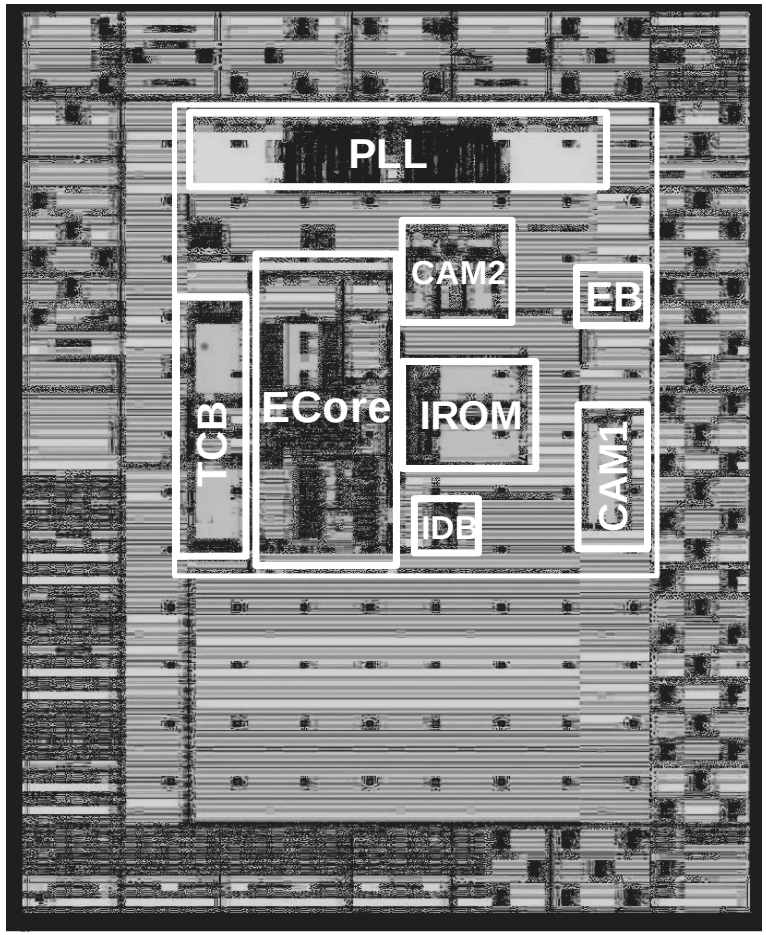
- **Multi-core solution with each core supporting up to 8K connections**
- **Large number of connections:**
  - **> 32K to 64K**
- **4 signal handshake between CTL and each TIPP**

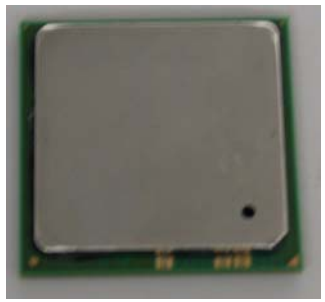
Enabling Circuit Technologies

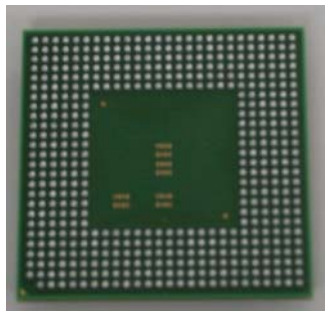


- Quaternary tree adder architecture
- Dual-V_T design
- Semi-dynamic flip-flops
- High speed instruction ROM
- High bandwidth, leakage tolerant register file
- Adaptive body bias
- Multiple clock domain synchronization

Experimental Chip

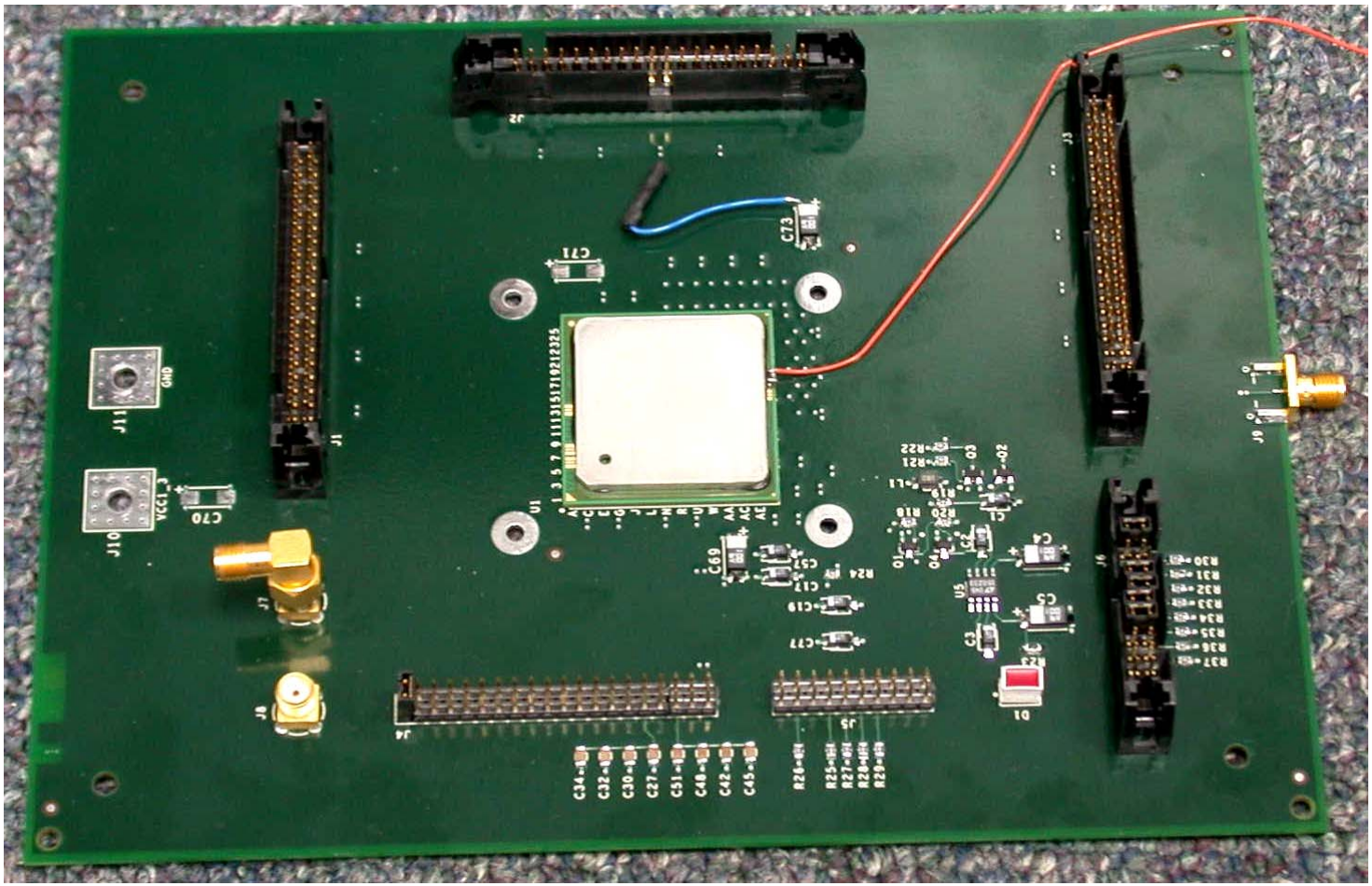




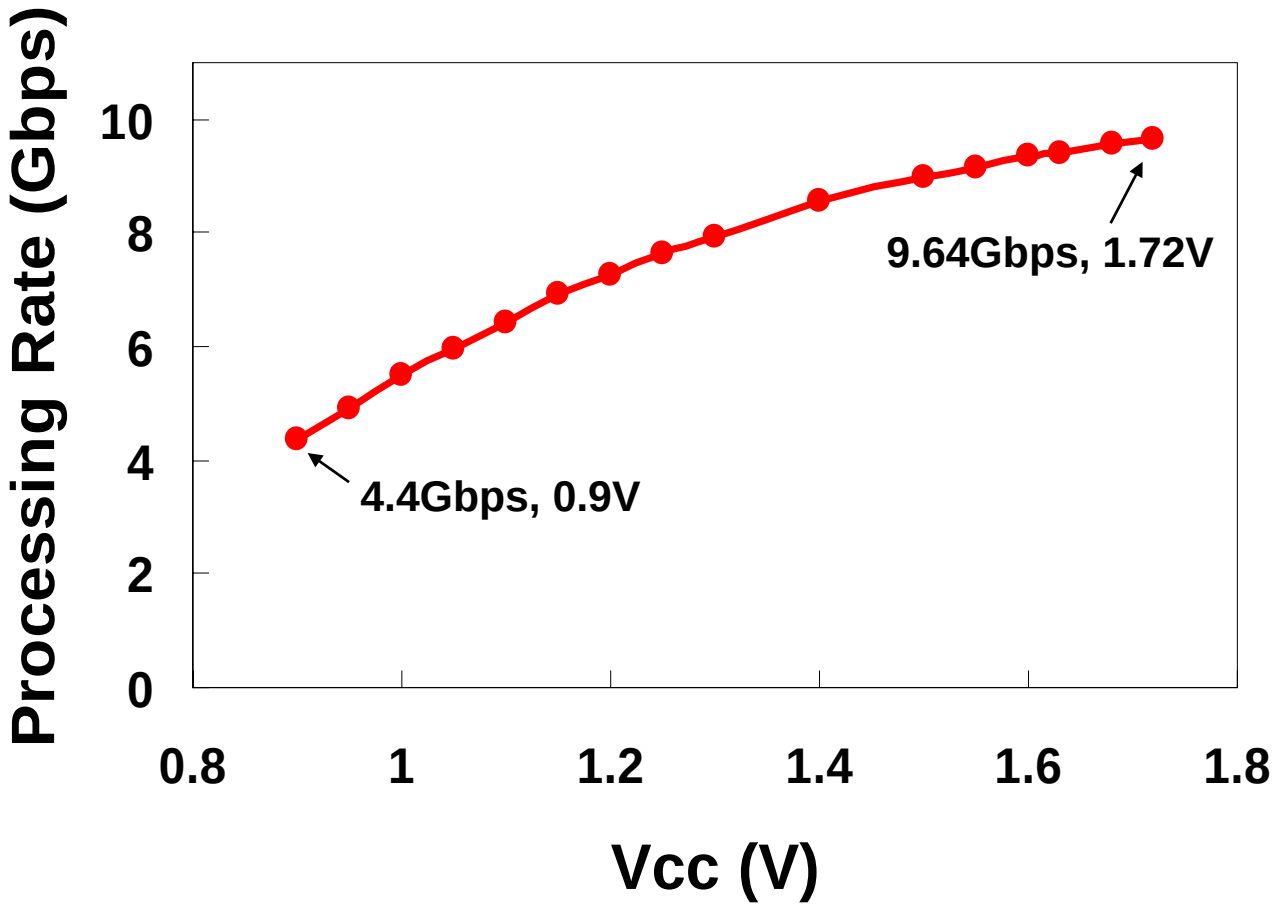


Chip Area	2.23 x 3.54mm^2
Process	90nm comm. CMOS
Interconnect	1 poly, 7 metal
Transistors	460K
Pad count	306

Evaluation Board

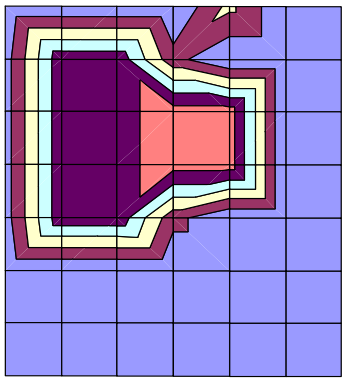


Measured Processing Performance



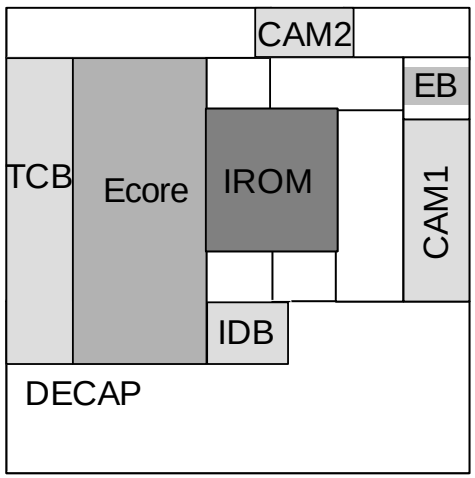
Power Consumption

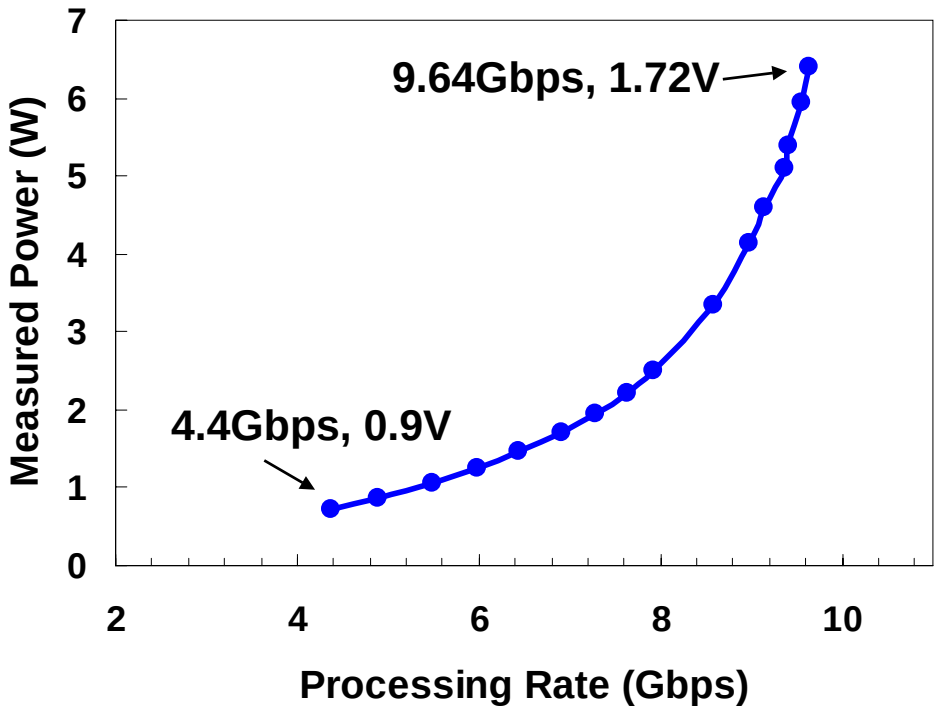
Simulated Power Density



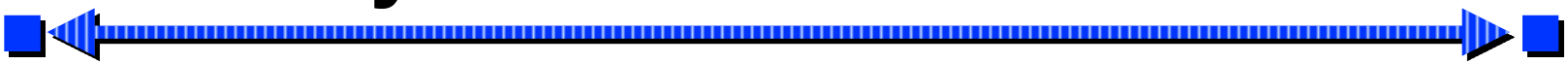
Watts/sq.cm.
@1.2V, 100C

- 250-300
- 200-250
- 150-200
- 100-150
- 50-100
- 0-50





Summary



- Programmable hardware engine
  - Line-speed TCP ingress processing
  - Support minimum packet size
- Dual frequency and buffer-free design
  - High speed execution core
  - 9.64Gbps processing at 1.72V, 6.39W in 90nm communication CMOS process
  - Extendable to large number of connections
- Dynamic reordering of packets in hardware

Acknowledgments

- B. Bloechel, G. Dermer, D. Finan, J. Howard, K. Ikeda, D. Klowden, S. Narendra, H. Nguyen, C. Parsons, G. Ruhl, D. Somasekhar, S. Tang, K. Truong, J. Tschanz, V. Veeramachaneni, H. Wilson for the TIPP experimental chip contributions
- A. Foong, F. Hady for simulation contributions
- S. Borkar, M. Haycock for encouragement and support