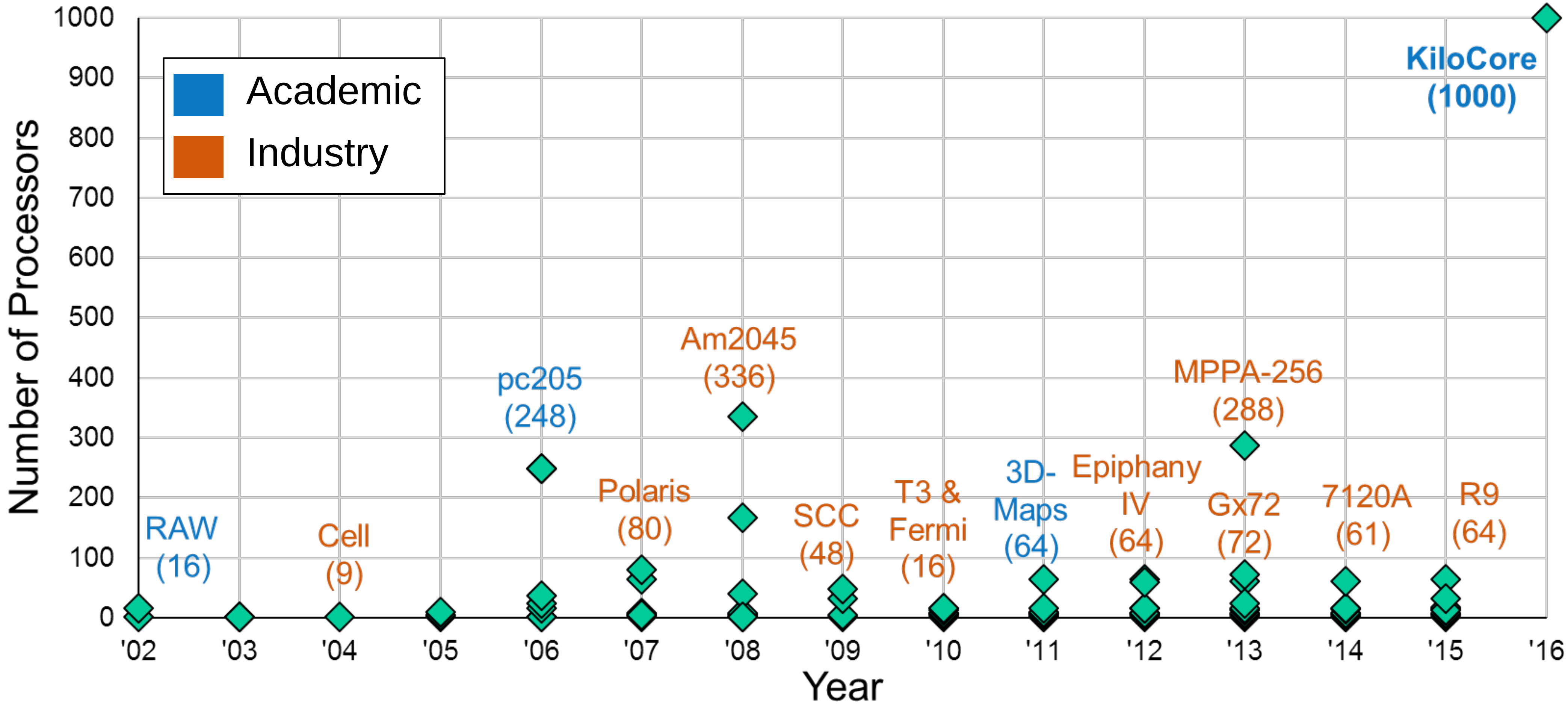
KiloCore: A 32 nm 1000-Processor Array

Brent Bohnenstiehl, Aaron Stillmaker,
Jon Pimentel, Timothy Andreas, Bin Liu,
Anh Tran, Emmanuel Adeagbo, Bevan Baas

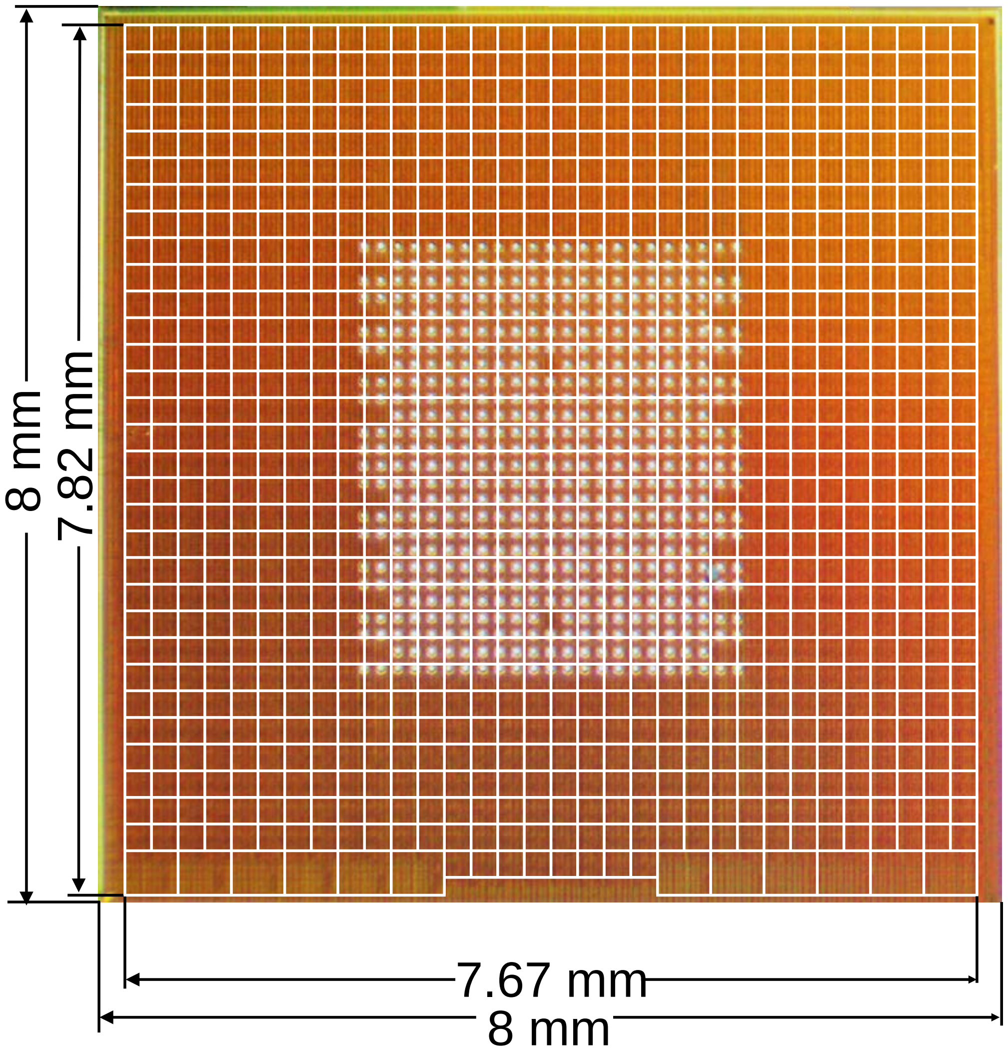
University of California, Davis
VLSI Computation Laboratory
August 23, 2016

Processors Over Time



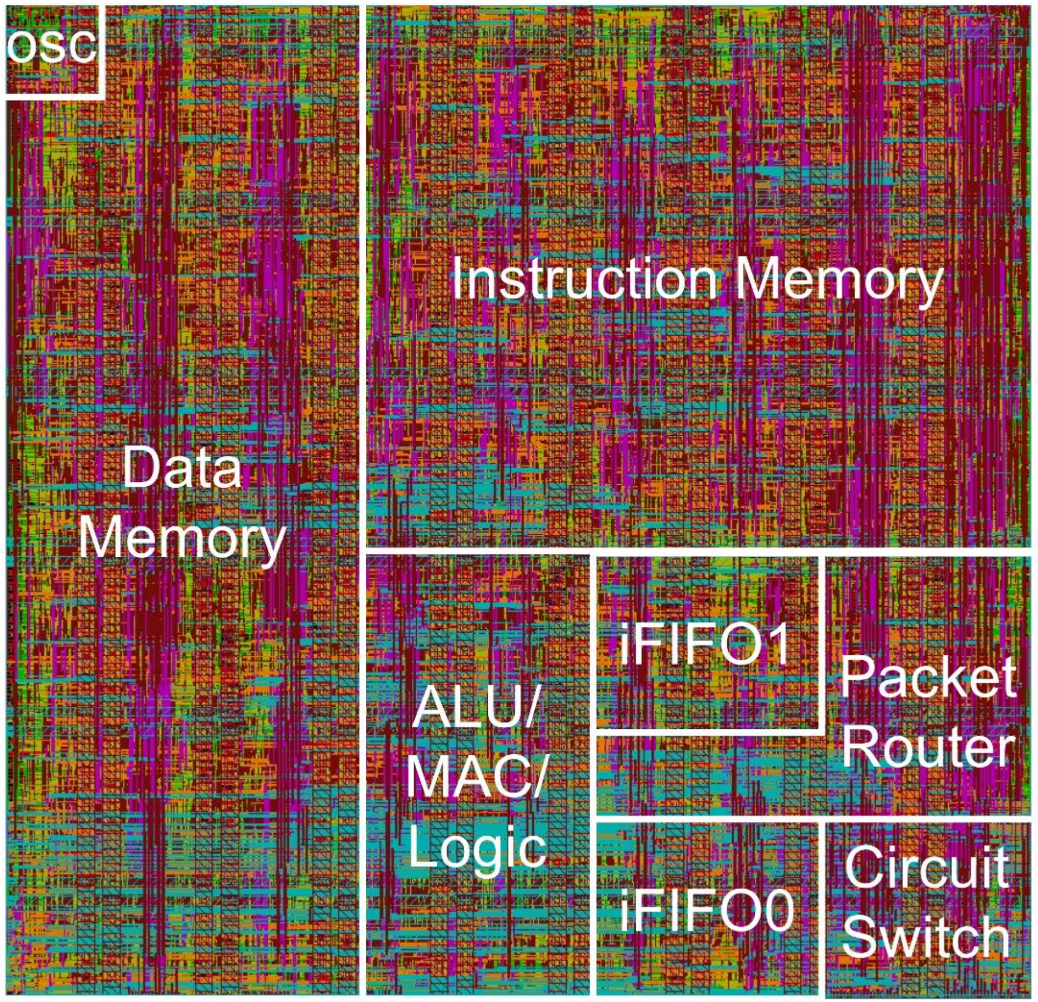
- Number of processors on single die vs. year
  - Each processor capable of independent program execution

KiloCore Chip



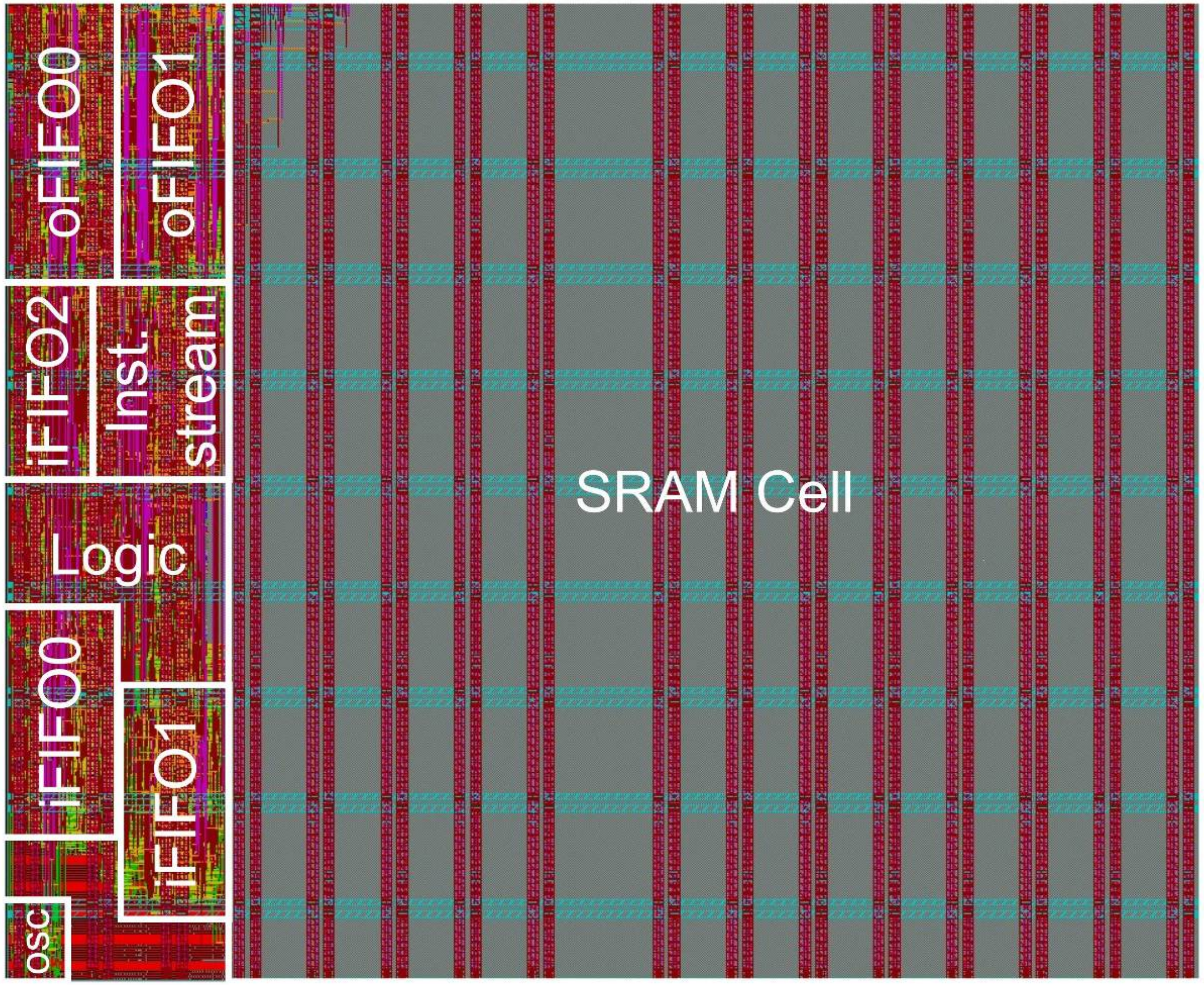
Technology	32nm IBM PDSOI CMOS
Num. Procs.	1000
Num. Mems.	12
Die Area	64 mm^2
Array Area	60 mm^2
Transistors	621 Million
C4 Bumps	564 (162 I/O)
Package	676 Pad Flip-Chip BGA

Single Processor Tile



Tile Area	0.055 mm^2
Transistors	574,733
Instruction Memory	128 x 40-bit
Data Memory	256 x 16-bit
Input FIFO Size (x2)	32 x 16-bit
Instruction Types	72

Single Memory Tile



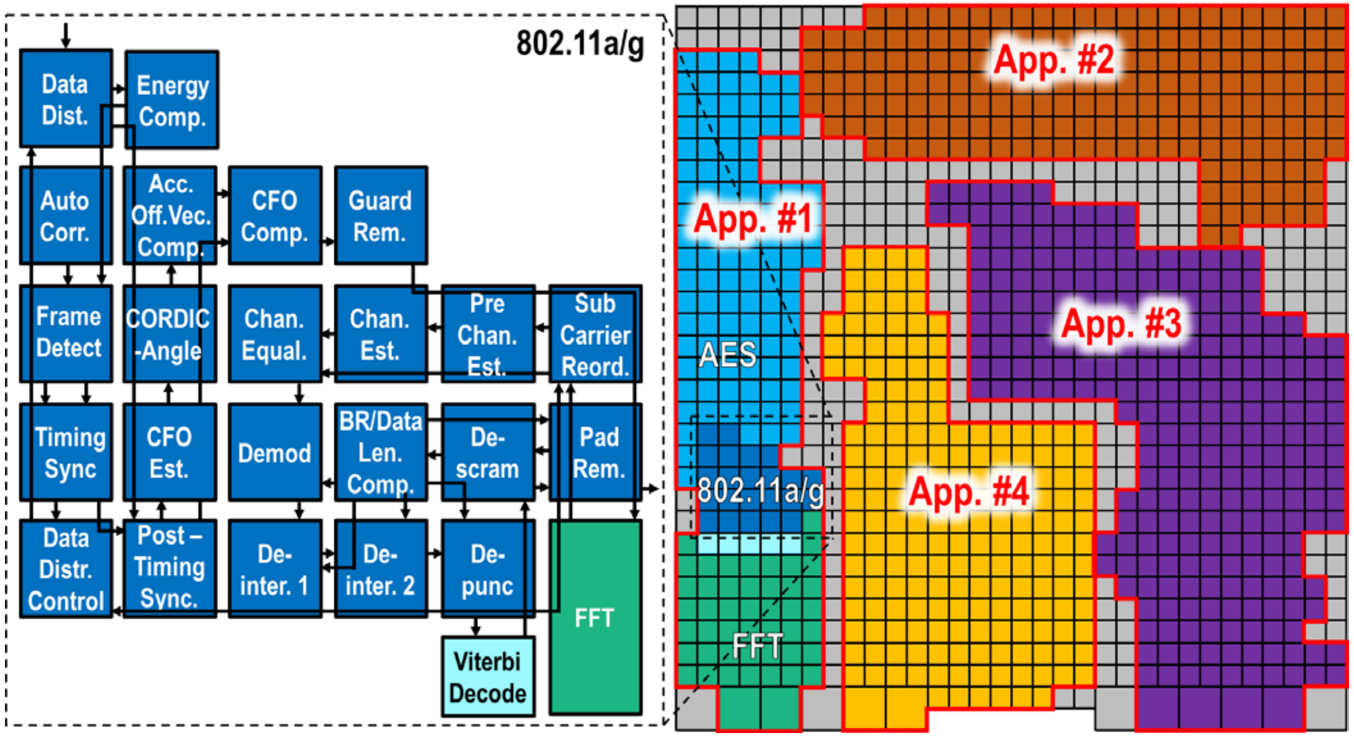
Tile Area	0.164 mm^2
Transistors	3,813,095
SRAM Size	64 kB
Input FIFO Size (x2)	32 x 18-bit
Input FIFO Size (x1)	16 x 2-bit
Output FIFO Size (x2)	32 x 16-bit

Overview

- KiloCore is best suited for computationally-intensive applications and kernels
- Each processor holds up to 128 instructions
  - 40-bits per instruction
  - Modified during application programming
  - Typically static during the run time of an application
  - Larger programs are supported for processors neighboring a memory module
- Data is passed by messages between processors
  - A pair of processors neighboring a shared memory may transfer data through that memory

Programming

- Applications are implemented as a set of suitably small programs by:
  - Organizing the application into a group of tasks
  - Partitioning task code into serial blocks
  - Replicating parallelizable code blocks
- Partitioning techniques are suitable for tool automation



Example of an application mapped onto KiloCore

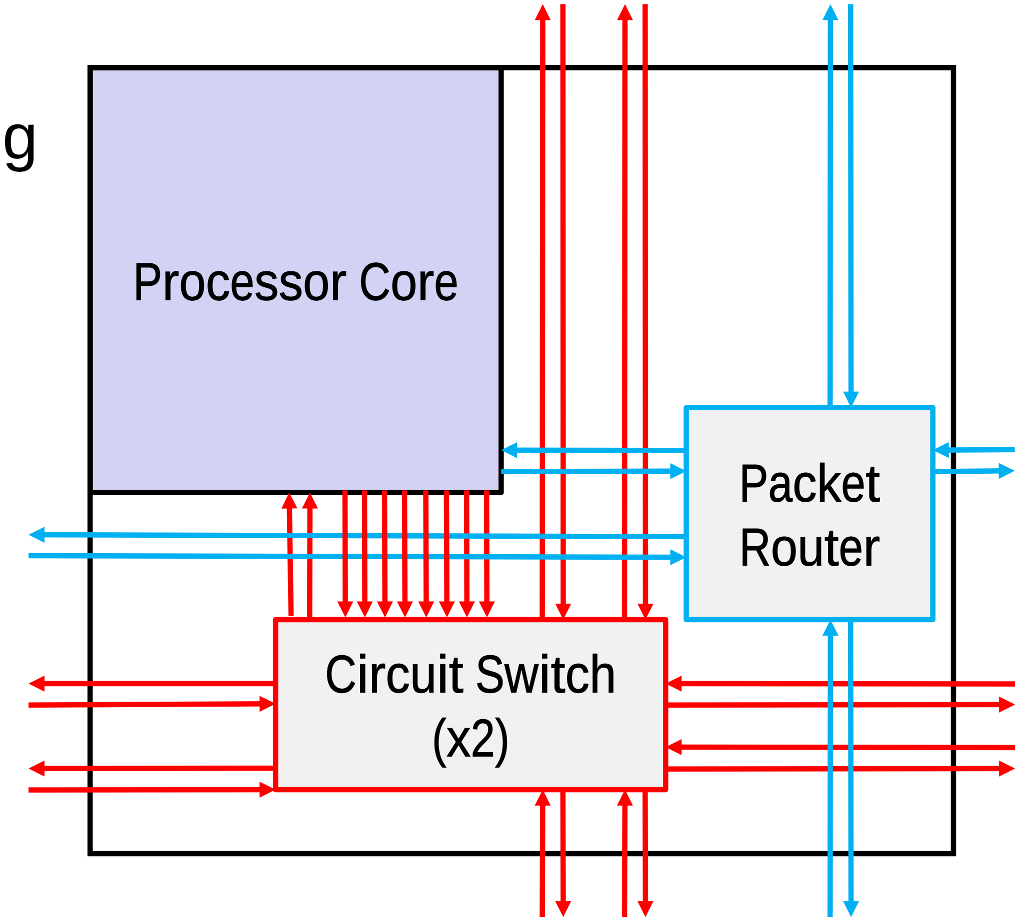
GALS Clocking

- Globally Asynchronous, Locally Synchronous Clocking
- 2012 oscillators
  - One per processor, packet router, and memory
- Oscillators may:
  - Independently change frequency
  - Halt within 1-5 clock periods when work is not available
  - Restart in less than 1 clock period
- Halted processors consume 1.1% of their typical active power
- Data is synchronized using dual clock buffers between domains

Note: Halted processor power measurement taken at 900 mV

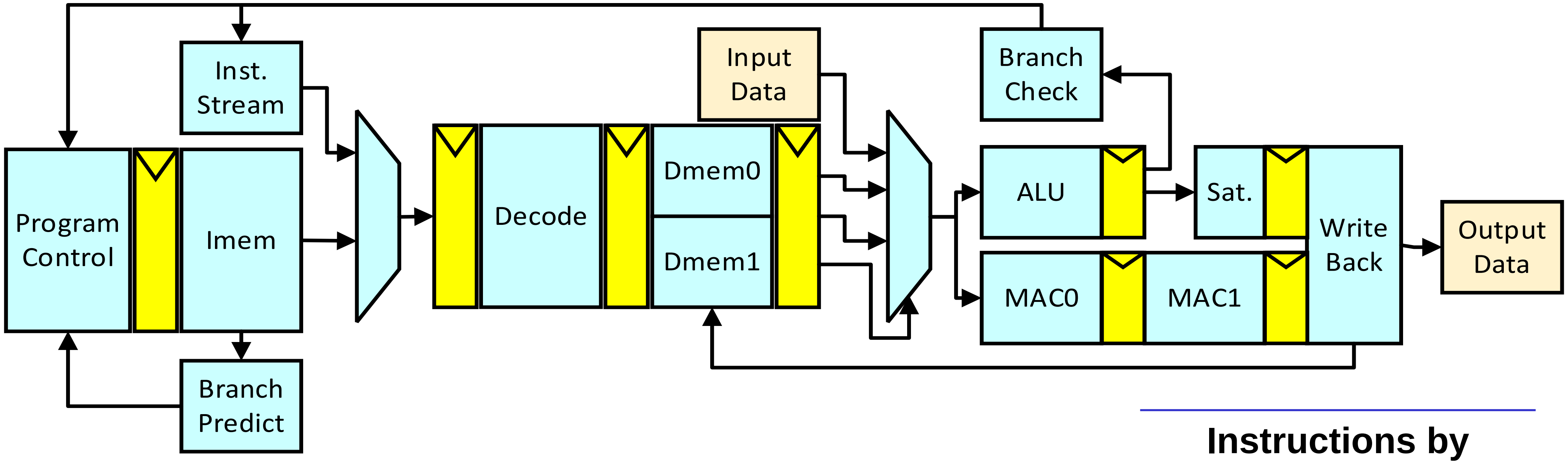
Communication Network

- Two layer circuit switched network
  - Statically configured during programming
  - Source-synchronous
  - 16-bit data width per link
  - Up to 28 Gbps per link
  - 456 Gbps total tile I/O
- Dynamic packet routing network
  - Wormhole routing
  - Source-synchronous
  - 16-bit data width per flit
  - Up to 9.1 Gbps per link



Note: bandwidth measurements taken at 1.1 V

Processor Pipeline



- 7-stage pipeline
- 16-bit, fixed-point datapath
- 40-bit, memory-to-memory instructions
- Single-issue, in-order execution

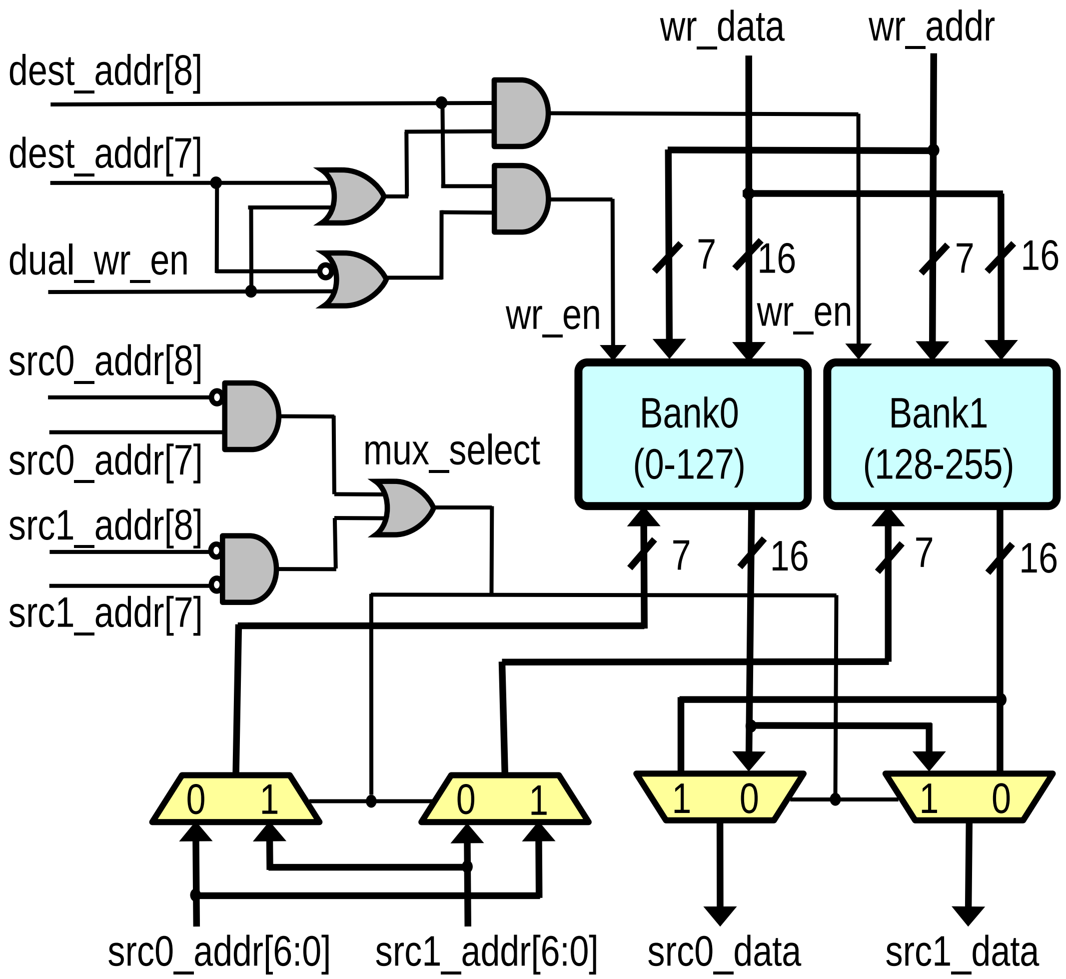
Instructions by Opcode Type	
Add/Sub	16
Logic	21
Mac	14
Branch	18
Other	3

Processor Pipeline

- Signed and unsigned operations
- Multiplier is 16-bit in, 32-bit out, with 40 bit accumulator
  - Supports one multiply per two cycles
- Predication supported for all instructions
- Automated loop hardware accelerates innermost loops
- Static branch prediction
  - Controlled by opcode selected during compilation
  - 94% of branches predicted correctly in sampled applications
    - Many branches close loops or handle special cases
    - Difficult to predict branches are often replaced with predication

Processor Data Memory

- Two data memory banks
- Instruction operands sourced one from each bank
  - Each source is assigned a default bank; if either source reads the other bank, swap banks
- Instructions optionally write back to one or both banks
  - Software selects this by setting a Dual_Write flag



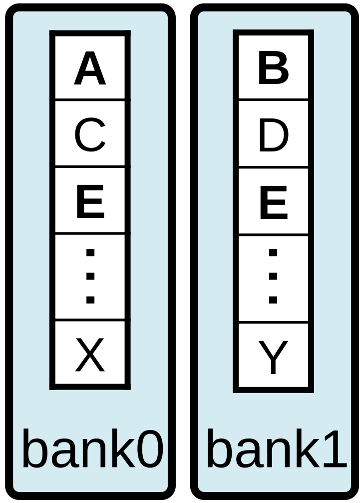
(Pipeline registers not shown)

Processor Data Memory

- The compiler will:
  - Find variables potentially read on the same cycle
  - Construct read conflict lists
  - Map variables to memory banks to avoid same-bank conflicts
    - A variable is mapped to both banks only when a conflict is otherwise unavoidable

Example of variable conflict analysis and mapping

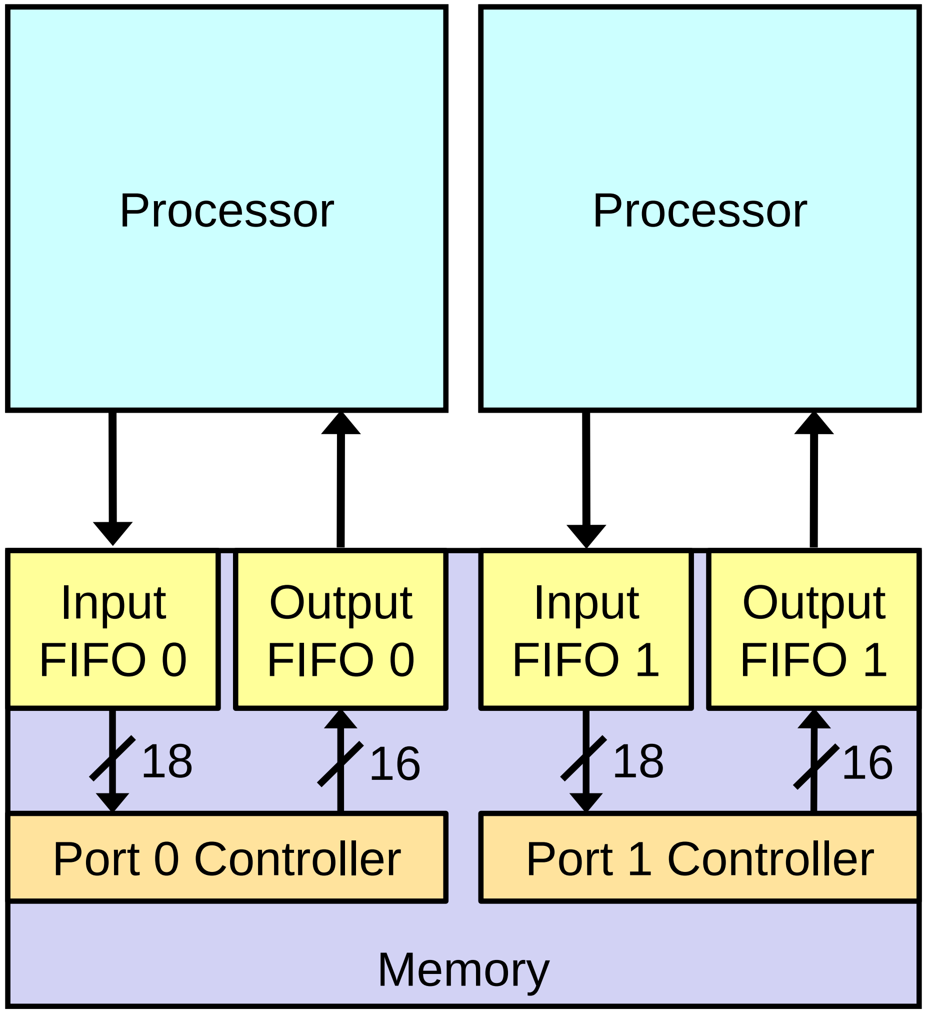
Var.	Conflicts with	Mapped to bank
A	B, E, …	0
B	A, E, …	1
E	A, B, …	0 & 1
…	…	…



Instr.	Src 0 bank	Src 1 bank	Swap read banks?	Dual write flag
C=A+B	0	1	No	0
E=D-C	1	0	**Yes**	**1**
X=**E-A**	0	0	**Yes**	0
Y=**E-B**	0	1	No	0

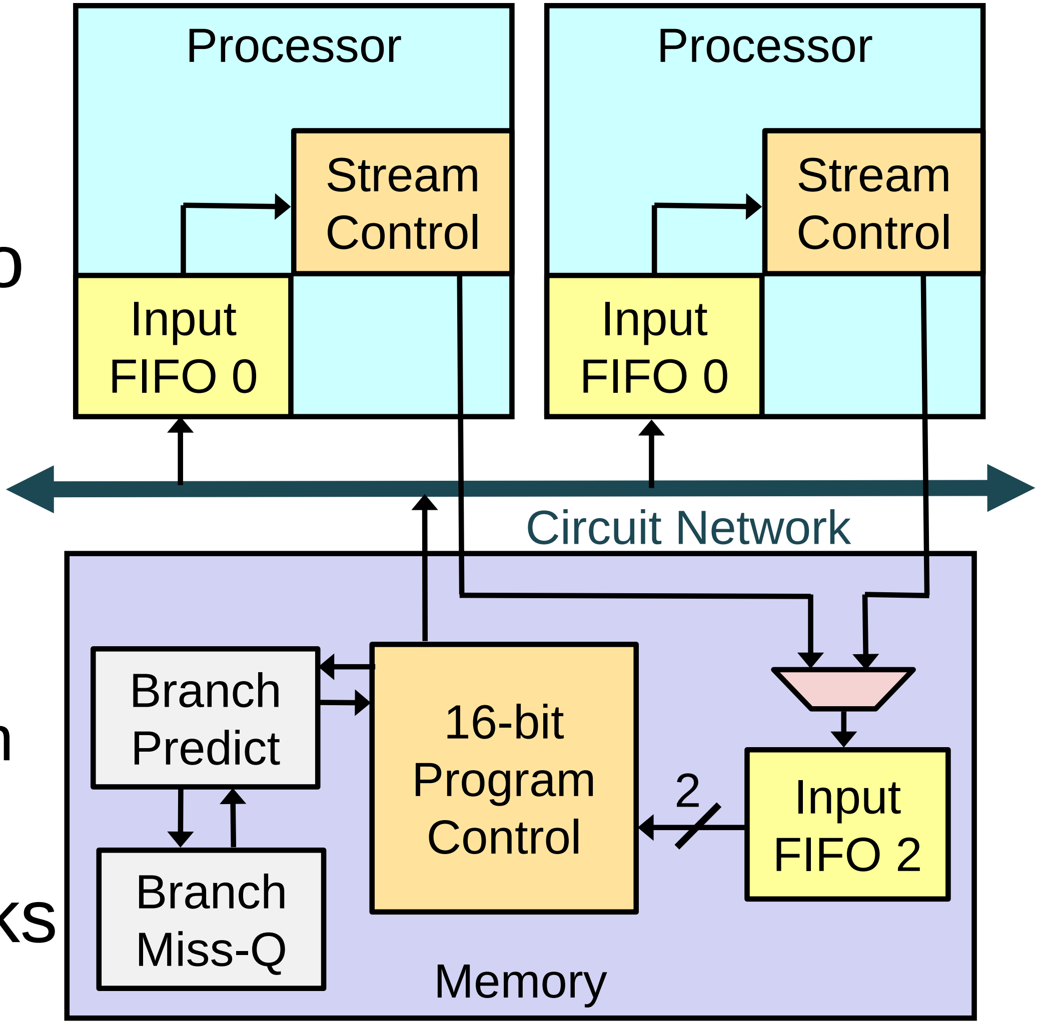
Shared Memory, Data Read/Write

- Each independent memory module connects to two neighboring processors
- Offers 64 kB of storage
  - 780 kB total across 12 memories
- Supports random and burst access modes, with programmable addressing patterns



Shared Memory, Instruction Streaming

- Memory may stream instructions to one neighboring processor
- Extends program size from 128 up to 10,922 instructions
- Program control is handled in the memory module
  - 16-bit controller
  - 8-deep branch prediction and correction queue
- Used for complex administrative tasks and highly serial, low priority tasks

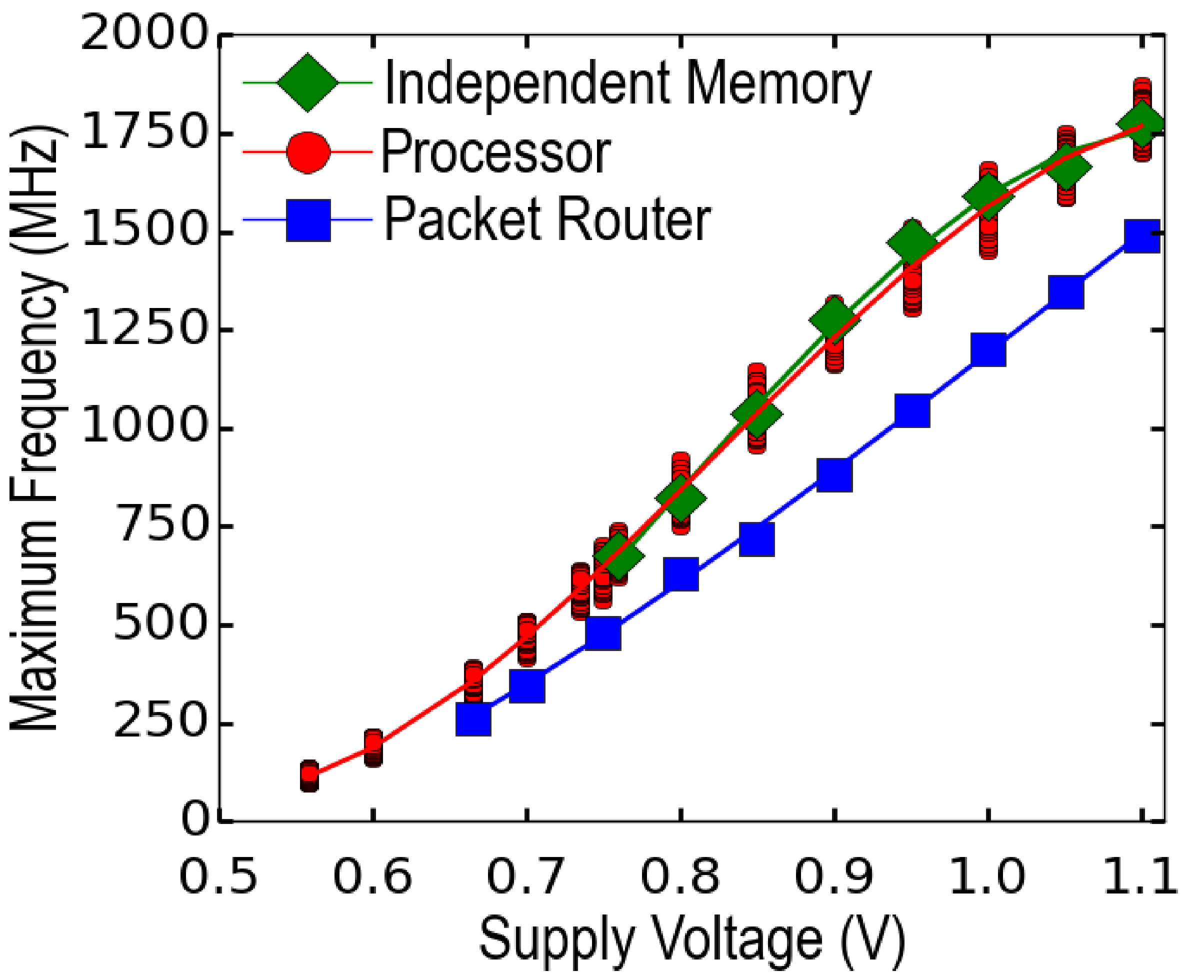


Physical Design Notes

- Tools used:
  - Design Compiler by Synopsys
  - SoC Encounter by Cadence
- 34 days between full access to design libraries and tapeout
- Chip functionality:
  - All processors, network, and shared memory are fully functional except hold time violations on some network paths
- Non-custom BGA flip-chip C4 package:
  - Indirect power delivery outside the center of the processor array leads to voltage droop in outer processors when operating at high voltage and activity

Frequency Measurements

Processor	
1.1 V	1.78 GHz
900 mV	1.24 GHz
560 mV	115 MHz
Independent Memory	
1.1 V	1.77 GHz
900 mV	1.27 GHz
760 mV	675 MHz
Packet Router	
1.1 V	1.49 GHz
900 mV	884 MHz
670 mV	262 MHz

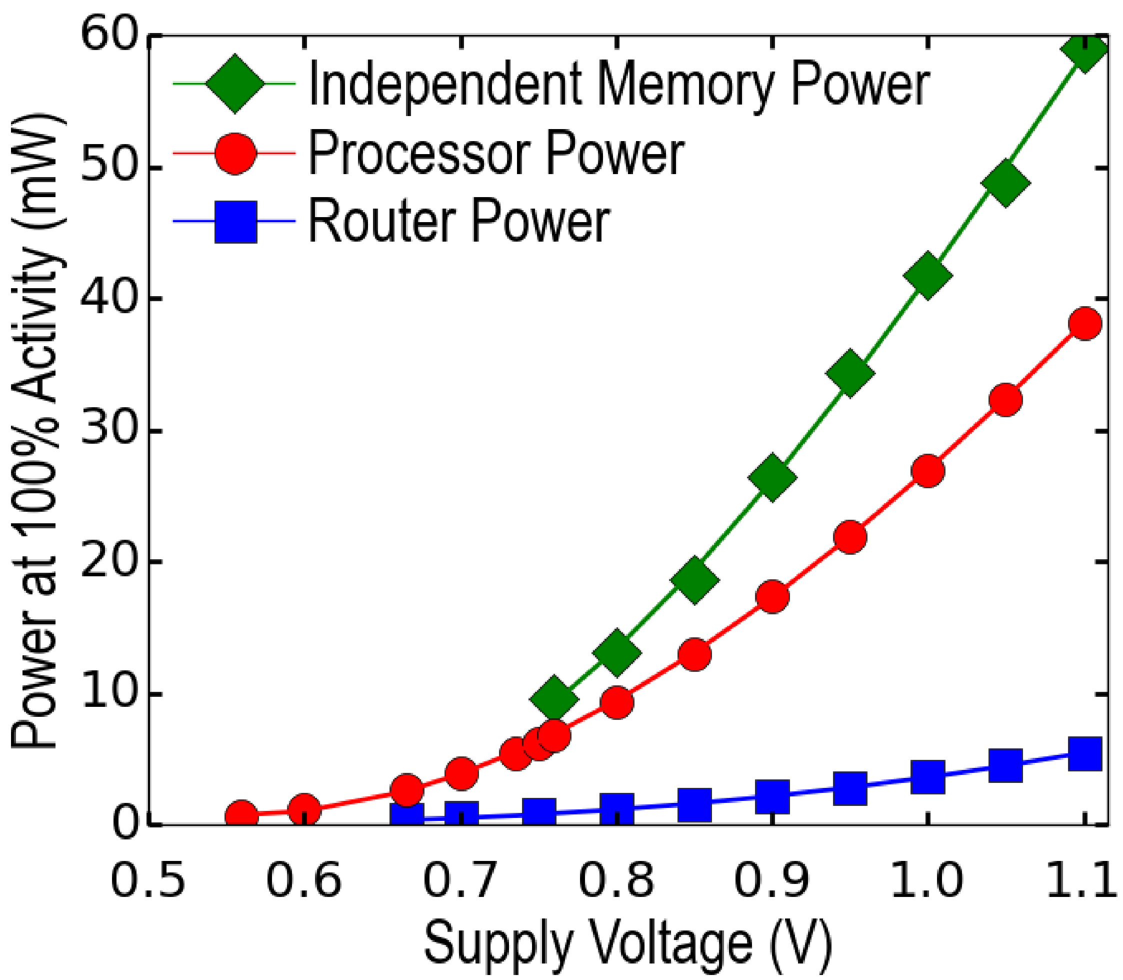


Notes:

Measurements made at 25ºC; lowest measurements are at the respective minimum operable voltages

Power Measurements

Processor	
1.1 V	38.8 mW
900 mV	17.7 mW
560 mV	0.7 mW
Memory	
1.1 V	59.0 mW
900 mV	26.5 mW
760 mV	9.5 mW
Packet Router	
1.1 V	5.5 mW
900 mV	2.1 mW
670 mV	0.4 mW



Measurements

- KiloCore has a potential maximum of 1.78 trillion instructions per second using 40 Watts
  - Assumes a custom package design
- At minimum voltage, KiloCore performs up to 115 billion instructions per second using 0.7 Watts
- Processors achieve their optimal energy times time of 11.1 (pJ x ns / instruction) at a voltage of 0.9 V
- Chip minimum voltage is constrained by any active application's usage of memories or routers
  - 760 mV if any independent memory is in use, 670 mV if the packet network is in use, 560 mV otherwise

Comparison Against Other Chips

Chip	Proc Count	Tech (nm)	Proc Area (mm^2)	Clock Freq (MHz)	Supply Voltage (V)	Energy/Op (pJ)	E x T (pJ x ns)	Bisection BW (Tb/s)
Sleepwalker [1]	1	65	0.42	25 23.6	0.4 0.375	2.6 **2.2**	104 93.2	N/A
IBM Cell [2]	9	90	14.5	**5000**	1.3	1100	220	2.46
Tilera/EZChip Gx72 [3]	72	40	-	1200	-	750	625	3.44
Intel TeraFlops [4]	80	65	3	4000 3130	1.2 1.0	70.6 49.1	17.7 15.7	2.65
Ambric Am2045 [6]	336	130	-	300	-	79.4	265	0.713
KiloCore [7]	**1000**	32	**0.055**	1782 1237 115	1.1 0.9 0.56	21.9 13.8 5.8	12.2 **11.1** 50.3	**4.24**

Academic   Industry

1. JSSC'13 2. MICRO'05 3. EZChip Product Brief 2016
4. ISSCC'07 5. JSSC'09 6. MICRO'07 7. VLSI Symp.'16

Applications

- Several applications have been implemented for KiloCore:

– Fast Fourier Transform
  - 4096 length, 16-bit fixed-point data
  - Using 980 processors, 12 memories
  - 138 thousand FFTs/s at 4.0 Watts

– Advanced Encryption Standard
  - 128-bit keys
  - Using 974 processors
  - 14.9 Gb/s at 9.1 Watts

– Low Density Parity Check
  - 4095 code length
  - Using 944 processors, 12 memories
  - 111 Mb/s at 3.4 Watts

– Record Sort
  - 100 Byte records with 10 Byte keys, 1850 records per sorted block
  - Using 1000 processors
  - 12.4 million records/s at 0.8 Watts

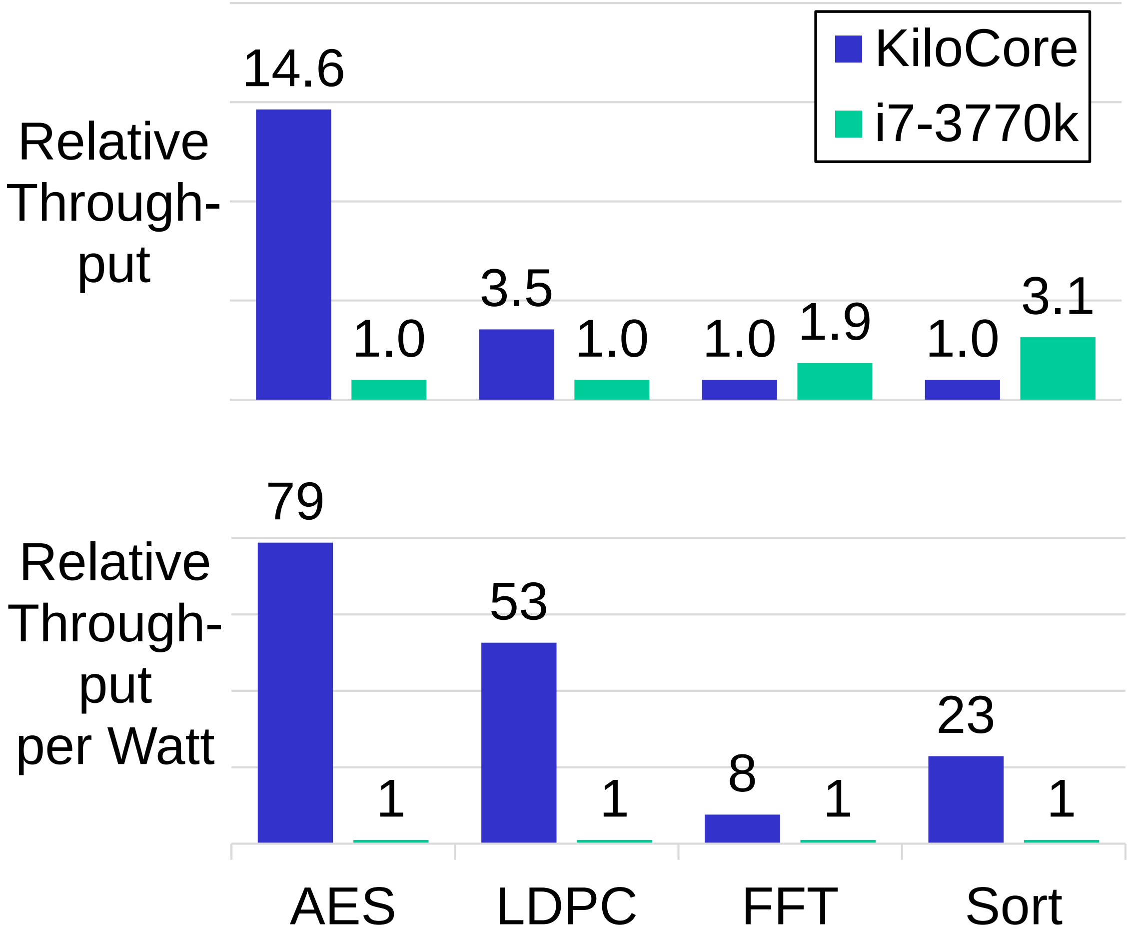
Notes:

Performance based on cycle-accurate simulations using fine-grain sub-instruction energy measurements at 900 mV.

Implementations have not been optimized.

Application Comparison

- Application implementations compared against a desktop Intel i7-3770k processor.
  - 22 nm technology, 160 mm2 die area
  - Using FFTW, C++ std::sort, open source AES C library, custom LDPC C++ implementation
    - FFT operating on single precision floating point data, not using AES specialized instructions, operating on pre-cached data, using 8 threads



Acknowledgments

- Funding and Support:
  - DoD and ARL/ARO Grant W911NF-13-1-0090
  - TAPO
  - NSF CAREER award 546907

    CCF Grant No. 430090

    CCF Grant No. 903549

    CCF Grant No. 1018972

    CCF Grant No. 1321163
  - SRC GRC Grant 1598

    CSR Grant 1659

    GRC Grant 1971

    GRC Grant 2321
  - ST Microelectronics
  - C2S2
  - Intel Corporation
  - UCD Faculty Research Grant
  - MOSIS
  - Artisan