Photonic Many-Core Architecture Study

Nadya Bliss[1], Krste Asanović[2], Keren Bergman[3], Luca Carloni[3], Jeremy Kepner[1], Sanjeev Mohindra[1], Vladimir Stojanović[4]

[1]MIT Lincoln Laboratory, [2]University of California Berkeley, [3]Columbia University, [4]MIT Research Laboratory of Electronics

September 23rd, 2008

PM: Jagdeep Shah

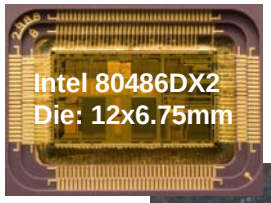
This work is sponsored by DARPA under Air Force contract FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

Outline

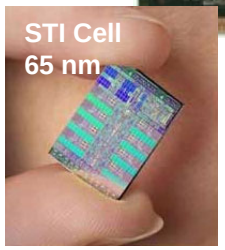
- **Introduction**
- Logical Architecture Abstraction
- Modeling and Mapping
- Experiments and Results
- Summary

Emerging Device Trends

Feature Size Reduction

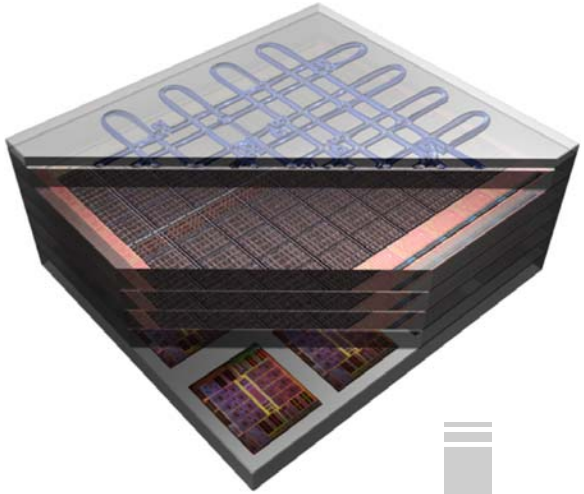


1970s

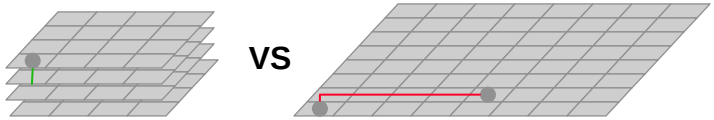


2008

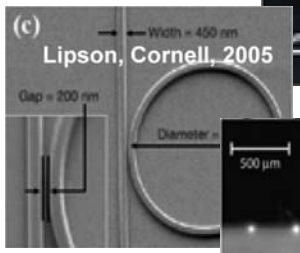
3D Fabrication

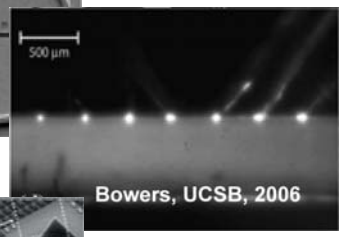


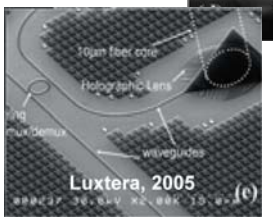
Reduced path length for accesses across the memory hierarchy



Photonic Interconnects





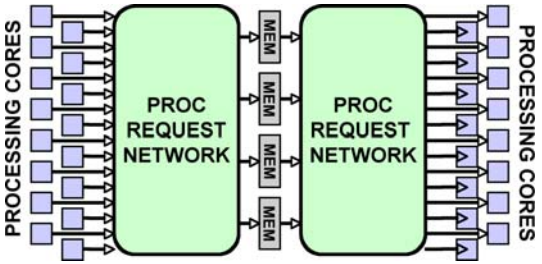


Emerging device technologies create a large parameter space of possible future architectures

Benefits of Photonic Interconnects

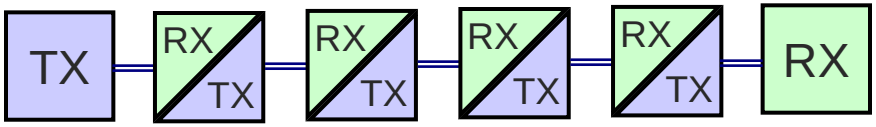
ELECTRONICS

TO MEMORY



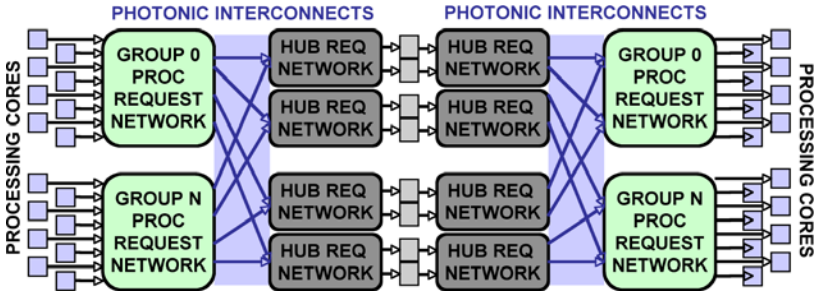
- Communication to memory banks is chip power and pin/wire density limited
- Poor scaling of on-chip mem controllers with cores
- At most 3-6 Tb/sec in the next few years

CORE-TO-CORE

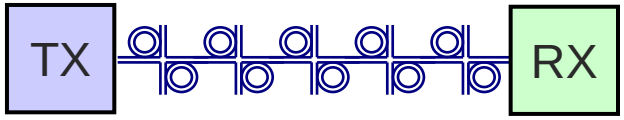


- Buffer, receive and re-transmit at every switch
- Power dissipation grows with data rate

OPTICS



- Use optical network as an efficient global crossbar
- Better scaling with N groups
- Expected performance - 40-80 Tb/sec



- Modulate/receive data once per communication
- Scalable, low power switch fabric
- Balanced communication and computation

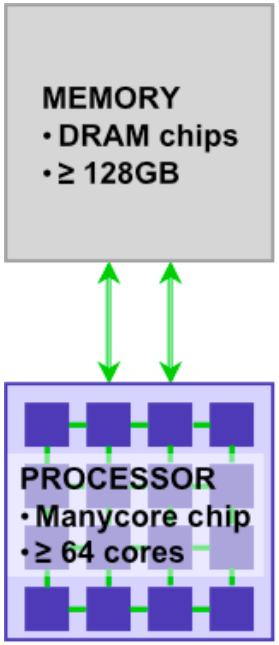
Photonics can provide high bandwidth, low latency communication while meeting power requirements of embedded systems.

System Level View

-Photonic Many-core Architecture Network: PhotoMAN-

Selecting a system level architecture allows the parameter space to be narrowed while meeting requirements of DoD applications.

- **Manycore processor chip**
  - **64-256 cores (in 22nm node)**
- **Off-chip memory**
  - **a set of DRAM chips**
  - **minimum capacity - 128 GB (at 22nm)**
- **Evaluate interaction of the photonic network and memory hierarchy**
- ***Board power limit 500 W***
  - **Consistent with power constraints of medium-sized UAV**



To evaluate the architecture develop

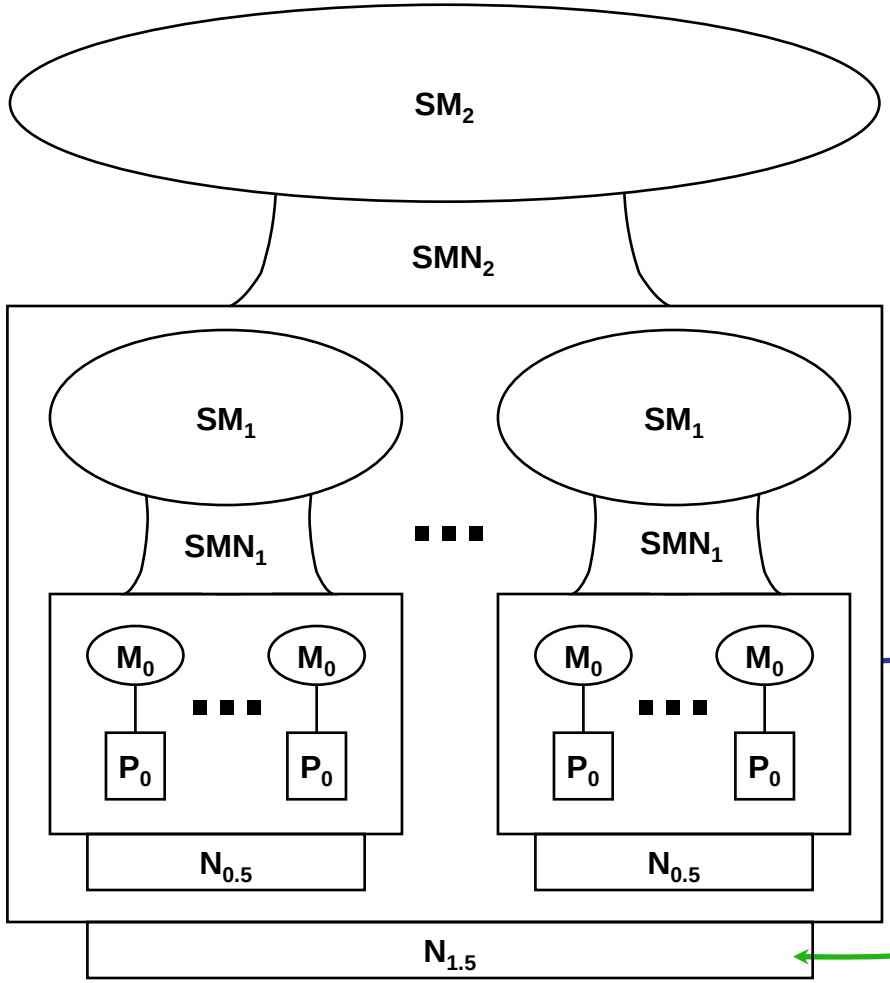
1. **Expressive logical abstraction**
2. **Modeling and mapping framework**

Outline

- Introduction
- **Logical Architecture Abstraction**
- Modeling and Mapping
- Experiments and Results
- Summary

Logical Abstraction

-Kuck* Memory Hierarchy-



2-LEVEL HIERARCHY EXAMPLE

Legend:
- P - processor
- N - inter-processor network
- M - memory
- SM - shared memory
- SMN - shared memory network

Subscript indicates hierarchy level

x.5 subscript for N indicates indirect memory access

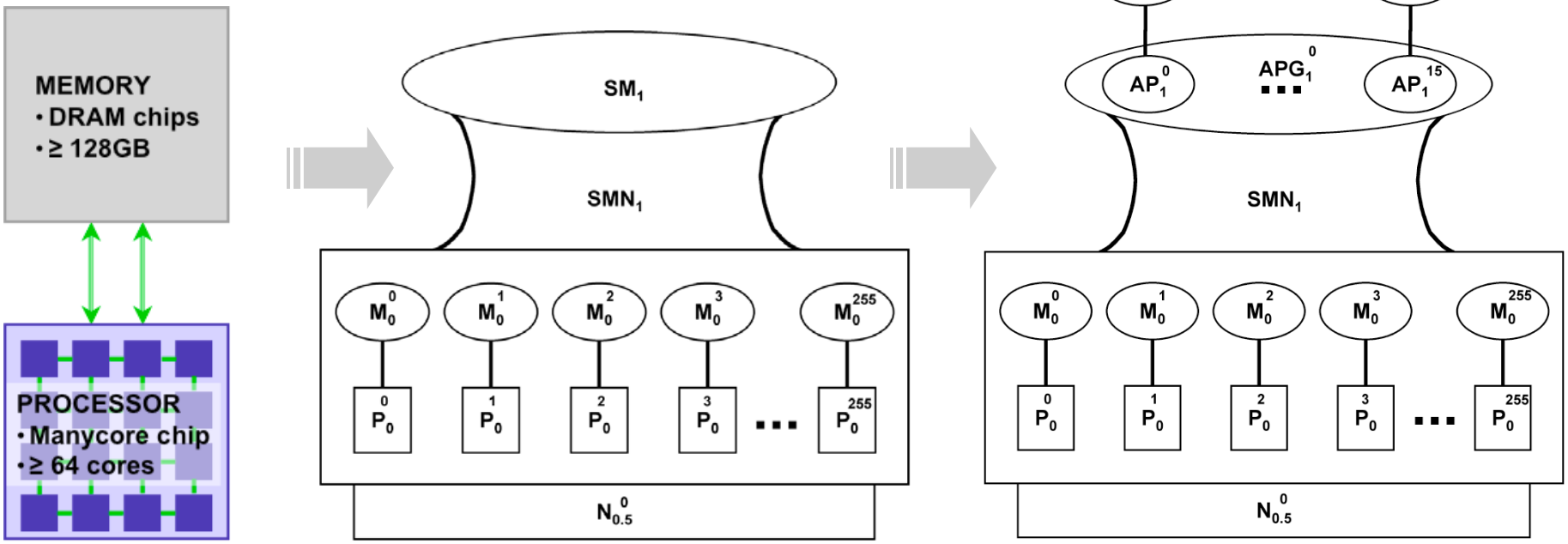
The Kuck notation provides a clear way of describing a hardware architecture along with the memory and communication hierarchy

**High Performance Computing: Challenges for Future Systems*, David Kuck, 1996

PhotoMAN Logical Representation

-MIT/UCB 1 Group Memory Configuration-

System-Level	High-Level	Detailed

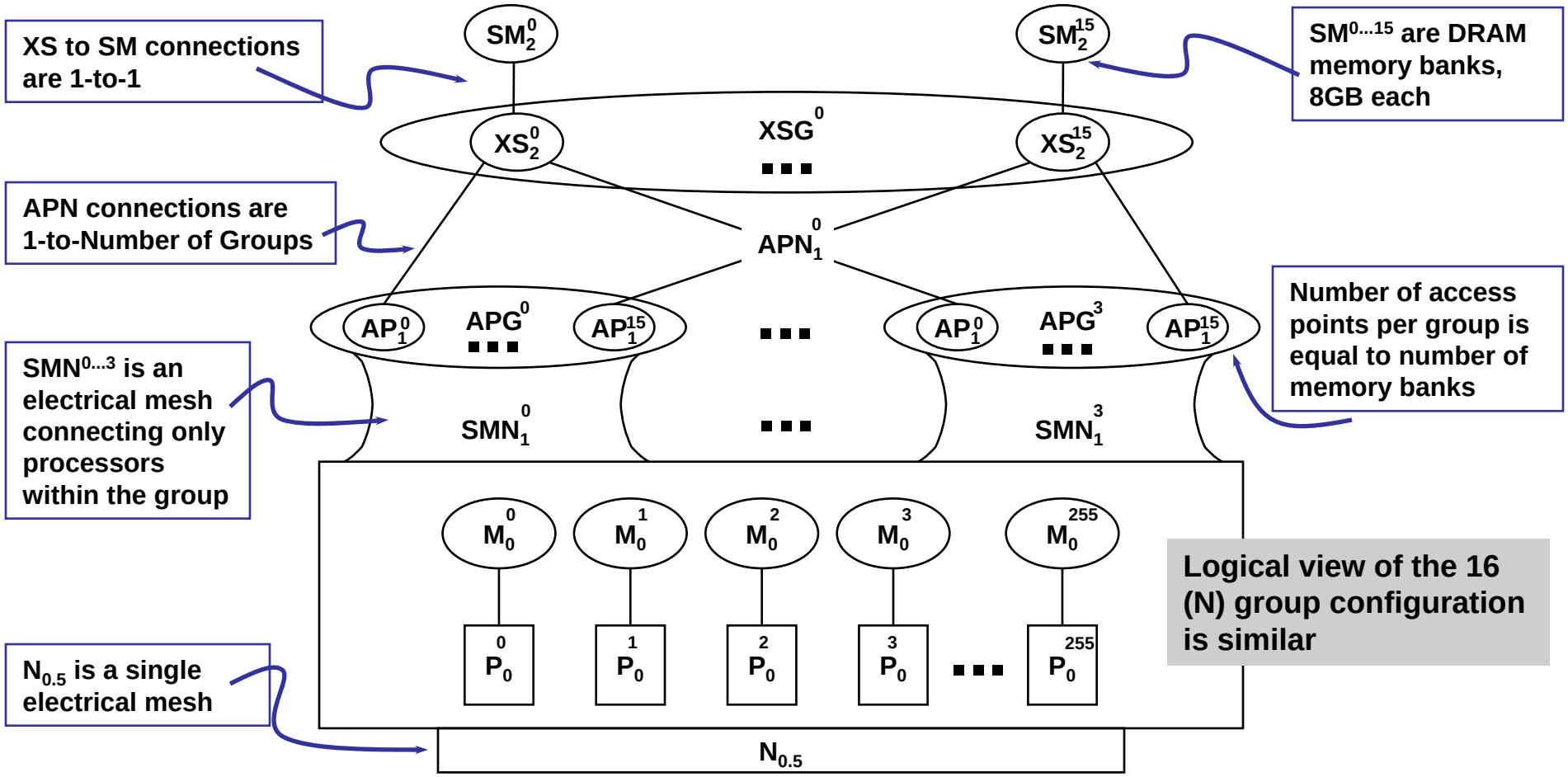


The Kuck notation is suitable for both high-level and detailed physical descriptions of the architecture, such as groups and access points.

Legend:
- **AP - access point**
- **APG - access point group**

PhotoMAN Logical Representation

-MIT/UCB 4 Group Memory Configuration-



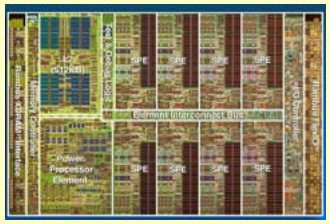
While the Kuck representation is flexible, the PhotoMAN study is focused on 1, 4, and 16 group memory configurations.

Legend:
- APN - access point network
- XS - cross bar
- XSG - cross bar group

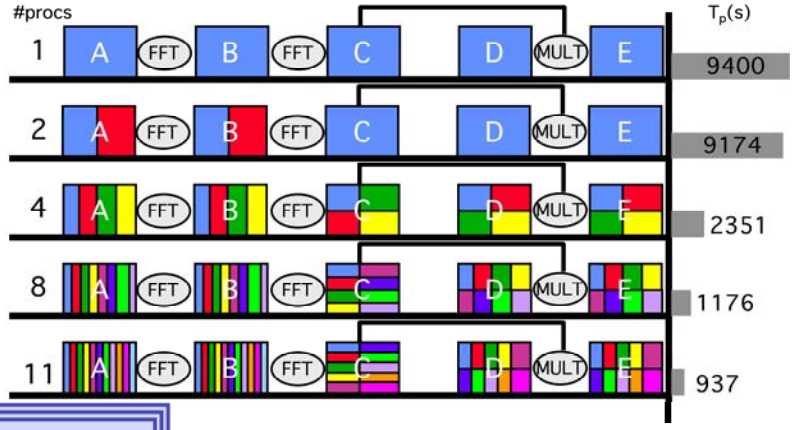
Outline

- Introduction
- Logical Architecture Abstraction
- **Modeling and Mapping**
- Experiments and Results
- Summary

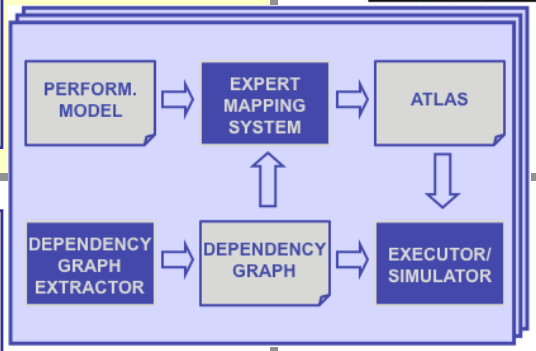
pMapper: Modeling and Mapping



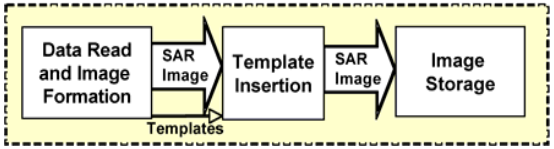
Machine description together with an abstraction layer is used to generate a performance model



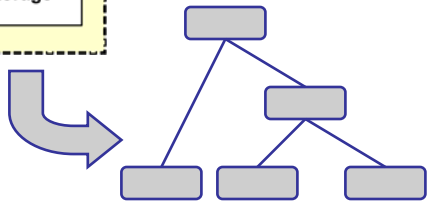
Maps (distribution specifications) are generated for the application



Application specification (MATLAB) is used to generate a signal flow graph



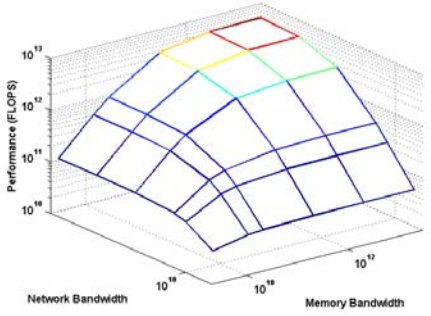
APPLICATION



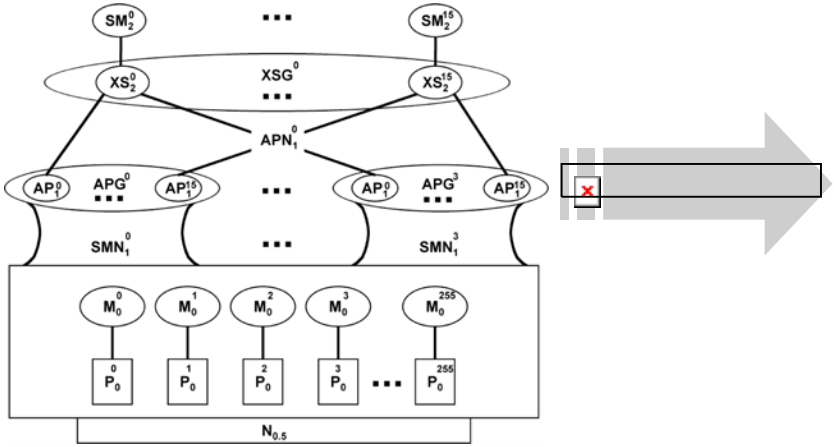
SIGNAL FLOW GRAPH

Results can be used to predict application performance and architecture parameters

pMapper performs
- application to architecture mapping
- application on architecture simulation



PhotoMAN Machine Description



Parameter	Symbol	Unit
Number of processing cores	N_P	N/A
P_0^i speed	R_P	(FL)OPs/second
P_0^i latency	L_P	second
P_0^i efficiency on operation k	$E_{P_{i,k}}$	N/A
M_0^i, local memory capacity of P_0^i	C_{local}	bytes
Number of memory banks	N_{MB}	N/A
Locations of access points (core indices)	$\mathbf{A}$	N/A
Number of groups	N_G	N/A
SM_1, on-chip shared memory capacity	$C_{on-chip}$	bytes
Size of a single data element of data type T	S_T	bytes
$N_{0.5}$, inter-core network bandwidth	$\mathbf{R_N}$	bytes/second
$N_{0.5}$, inter-core network latency	$\mathbf{L_N}$	second
SMN_1, on-chip memory network bandwidth	$\mathbf{R_{M_{on}}}$	bytes/second
SMN_1, on-chip memory network latency	$\mathbf{L_{M_{on}}}$	second
Access point to on-chip memory bandwidth	$\mathbf{R_{A_{M_{on}}}}$	bytes/second
Access point to on-chip memory latency	$\mathbf{L_{A_{M_{on}}}}$	second
Access point to crossbar bandwidth	$\mathbf{R_{A_{XS_{on}}}}$	bytes/second
Access point to crossbar latency	$\mathbf{L_{A_{XS_{on}}}}$	second
Crossbar to on-chip memory bandwidth	$\mathbf{R_{XS_{M_{on}}}}$	bytes/second
Crossbar to on-chip memory latency	$\mathbf{L_{XS_{M_{on}}}}$	bytes/second
SM_2, off-chip shared memory capacity	$C_{off-chip}$	bytes
SMN_2, off-chip memory network bandwidth	$\mathbf{R_{M_{off}}}$	bytes/second
SMN_2, off-chip memory network latency	$\mathbf{L_{M_{off}}}$	second

Given a hardware model *H* and a program parse tree *T*, pMapper finds maps *M* that minimize execution latency:

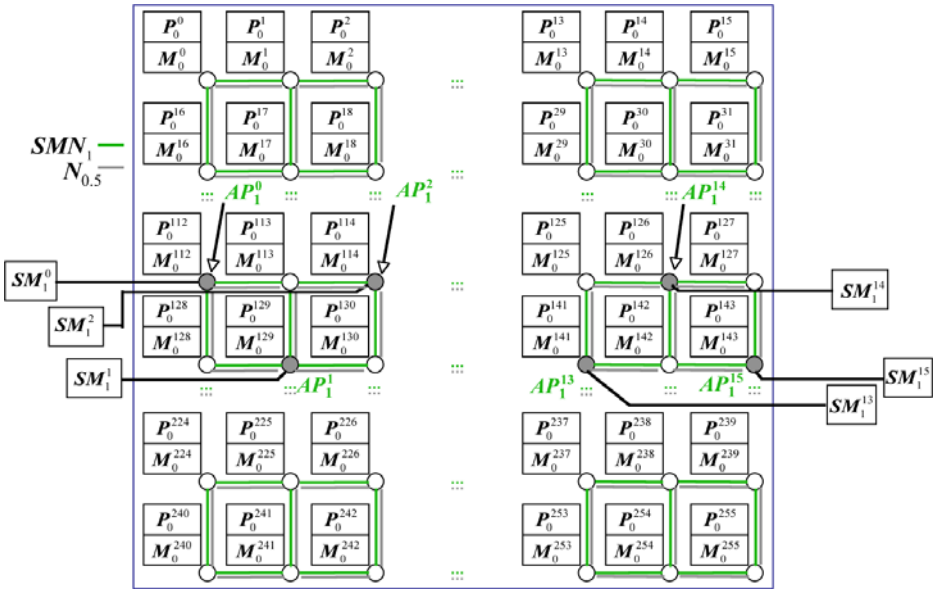
$$argmin_M f(T, H, M)$$

Focus of the PhotoMAN study

Memory Hierarchy Formulation

-MIT/UCB 1 Group Memory Configuration-

PHYSICAL VIEW



- Bandwidth and latency matrices have the same pattern of non-zeros
- Topology for $N_{0.5}$ and SMN_1 is the same for the 1-Group configuration
- Diagonal entries encode
  - R_N - bandwidth to local store
  - R_{Mon} - whether P^i is an access point

CORE-TO-CORE NETWORK, $N_{0.5}$

$$\mathbf{R_N} = \begin{array}{c} \\ P^0 \\ P^1 \\ P^2 \\ P^3 \\ \dots \\ P^{16} \\ \dots \\ P^{255} \end{array} \begin{array}{c} \begin{array}{cccccccc} P^0 & P^1 & P^2 & P^3 & \dots & P^{16} & \dots & P^{255} \end{array} \\ \begin{bmatrix} r_{M_0^0} & r_{N_{0.5}} & 0 & 0 & \dots & r_{N_{0.5}} & \dots & 0 \\ r_{N_{0.5}} & r_{M_0^1} & r_{N_{0.5}} & 0 & \dots & 0 & \dots & 0 \\ 0 & r_{N_{0.5}} & r_{M_0^2} & r_{N_{0.5}} & \dots & 0 & \dots & 0 \\ 0 & 0 & r_{N_{0.5}} & r_{M_0^3} & \dots & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ r_{N_{0.5}} & 0 & 0 & 0 & \dots & r_{M_0^{16}} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & \dots & \dots & r_{M_0^{255}} \end{bmatrix} \end{array}$$

SHARED MEMORY NETWORK, SMN_1

$$\mathbf{R_{M_{on}}} = \begin{array}{c} \\ P^0 \\ P^1 \\ P^2 \\ P^3 \\ \dots \\ P^{16} \\ \dots \\ P^{112} \\ P^{113} \\ \dots \\ P^{255} \end{array} \begin{array}{c} \begin{array}{cccccccccccc} P^0 & P^1 & P^2 & P^3 & \dots & P^{16} & \dots & P^{112} & P^{113} & \dots & P^{255} \end{array} \\ \begin{bmatrix} 0 & r_{SMN_1} & 0 & 0 & \dots & r_{SMN_1} & \dots & 0 & 0 & \dots & 0 \\ r_{SMN_1} & 0 & r_{SMN_1} & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & r_{SMN_1} & 0 & r_{SMN_1} & \dots & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & r_{SMN_1} & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ r_{SMN_1} & 0 & 0 & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & \dots & 1^{112} & r_{SMN_1} & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & \dots & r_{SMN_1} & 1^{113} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix} \end{array}$$

AP-to-SM

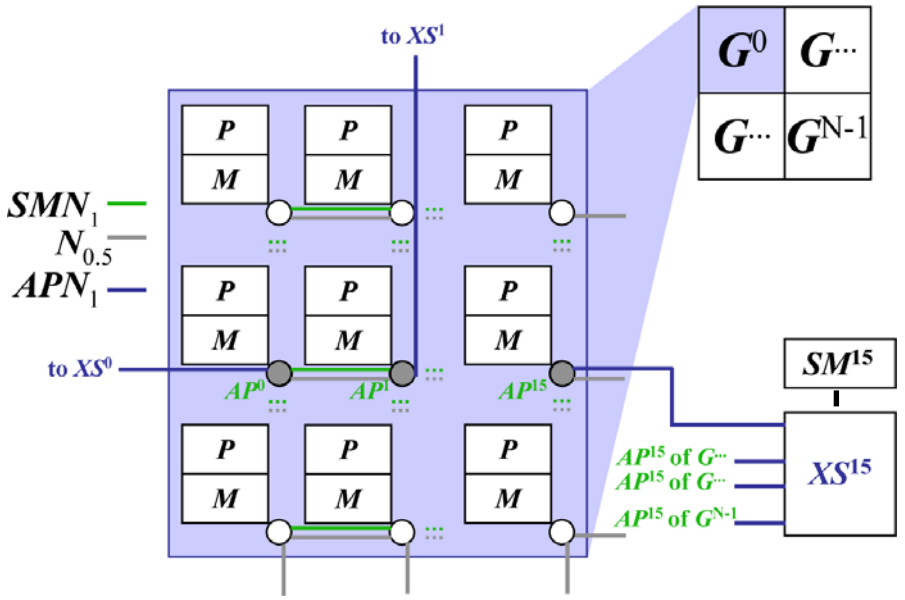
$$\mathbf{R_{A_{M_{on}}}} = r_{A_{M_0 n}} * I_{N_{MB}}$$

ACCESS POINTS

Memory Hierarchy Formulation

-MIT/UCB N_G Group Memory Configuration-

PHYSICAL VIEW



- Core-to-core network not shown and is the same as in 1 group case
- While memory access requires one additional transfer, the topology is represented with a single matrix - R_{AXSon}

SHARED MEMORY NETWORK, SMN_1

$$\mathbf{R}_{\mathbf{M}_{on}} = \begin{array}{c} \\ P^0 \\ P^1 \\ P^2 \\ P^3 \\ \dots \\ P^{16} \\ \dots \\ P^{48} \\ \dots \\ P^{64} \\ \dots \\ P^{255} \end{array} \begin{array}{c} \begin{array}{cccccccccccc} P^0 & P^1 & P^2 & P^3 & \dots & P^{16} & \dots & P^{48} & \dots & P^{64} & \dots & P^{255} \end{array} \\ \begin{bmatrix} 0 & r_{SMN_1} & 0 & 0 & \dots & r_{SMN_1} & \dots & 0 & \dots & 0 & \dots & 0 \\ r_{SMN_1} & 0 & r_{SMN_1} & 0 & \dots & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ 0 & r_{SMN_1} & 0 & r_{SMN_1} & \dots & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ 0 & 0 & r_{SMN_1} & 0 & \dots & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ r_{SMN_1} & 0 & 0 & 0 & \dots & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & \dots & 1^{48} & \dots & r_{SMN_1} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & \dots & r_{SMN_1} & \dots & 1^{64} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 0 & \dots & 0 & \dots & 0 & \dots & 0 \end{bmatrix} \end{array}$$

ACCESS POINTS

AP-XS-MEMORY NETWORK

$$\mathbf{R}_{\mathbf{APN}} = \begin{array}{c} \\ APG^0 \\ APG^1 \\ APG^2 \\ APG^3 \\ XSG^0 \end{array} \begin{array}{c} \begin{array}{cccccc} APG^0 & APG^1 & APG^2 & APG^3 & XSG^0 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & r_{A_{XS_{on}}} \\ 0 & 0 & 0 & 0 & r_{A_{XS_{on}}} \\ 0 & 0 & 0 & 0 & r_{A_{XS_{on}}} \\ 0 & 0 & 0 & 0 & r_{A_{XS_{on}}} \\ 0 & 0 & 0 & 0 & r_{XS_{M_{on}}} \end{bmatrix} \end{array}$$

AP-XS BANDWIDTH

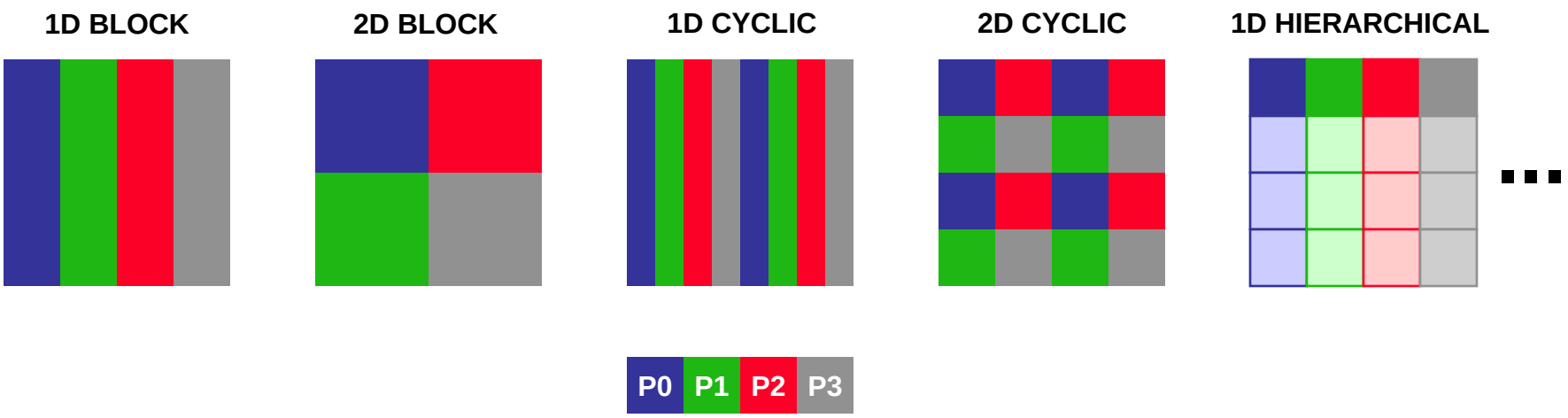
XS-MEMORY BANDWIDTH

$$\mathbf{R}_{\mathbf{A_{XS_{on}}}} = \mathbf{R}_{\mathbf{APN}} \otimes \mathbf{I}_{\mathbf{N_{MB}}}$$

Outline

- Introduction
- Logical Architecture Abstraction
- Modeling and Mapping
- **Experiments and Results**
- Summary

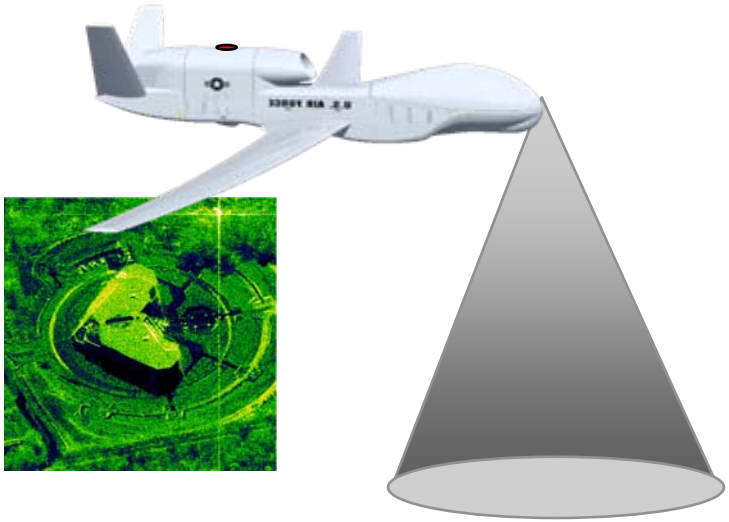
Maps



- High programmability is a desirable architecture characteristic
- Complexity of mapping chosen to optimize performance (minimize execution time) provides insight into programmability of hardware
  - The higher complexity of the mapping, the lower programmability

Synthetic Aperture Radar (SAR)

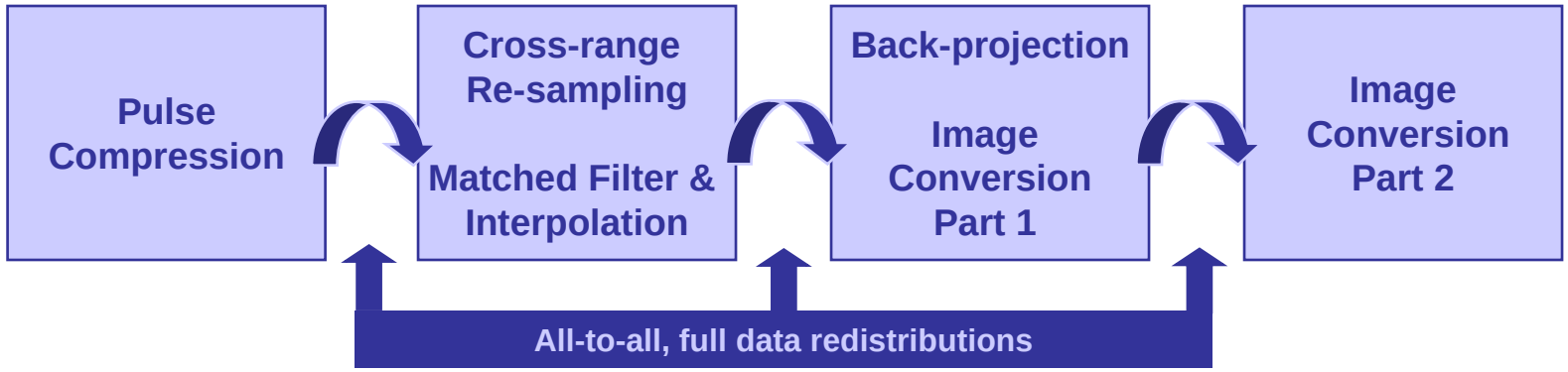
SAR processing chain is common to many defense application and requires significant amount of both computation and communication.



Typical application
- SAR spotlight mode
- Collect raw SAR data
- Processing chain produces an image
- Image can then be analyzed

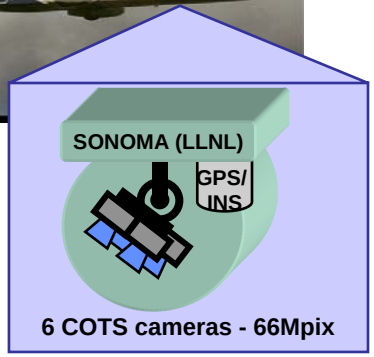
Processing chain simulated
- FFTs, IFFTs, and data-reorganization

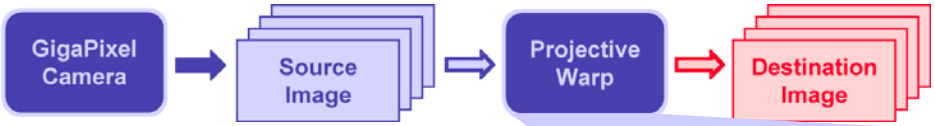
HPC Challenge relevance: FFT



Airborne Video Surveillance

Georegistration is a key computational kernel in airborne video surveillance and other image processing algorithms.

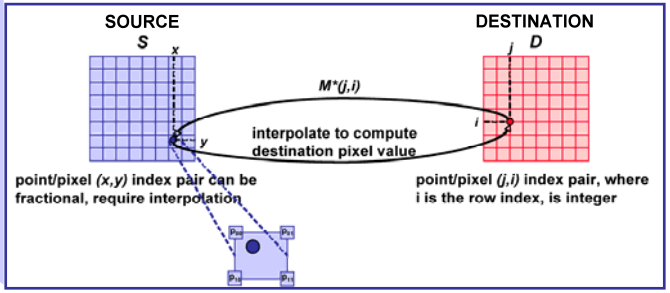




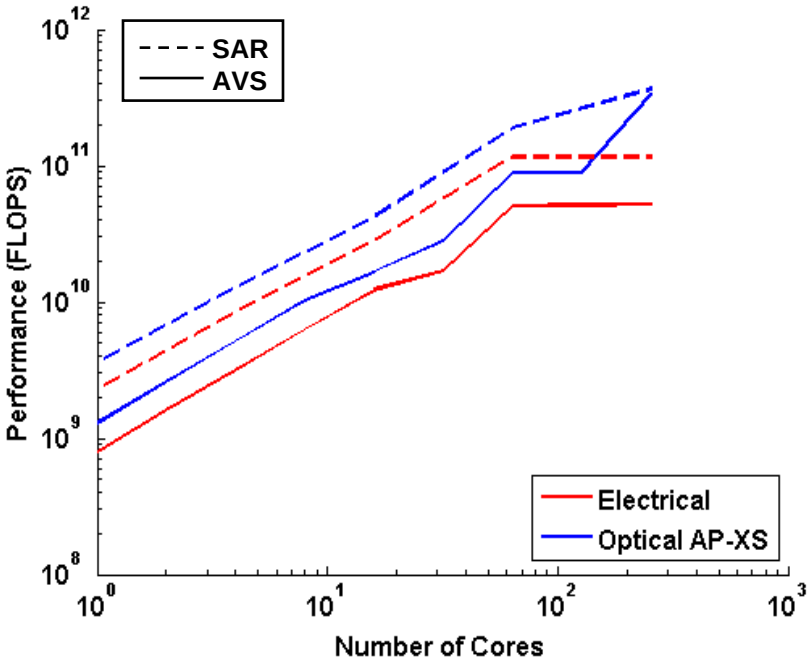
Typical application
- **High data rate imaging sensor**
- **Collect data**
- **Georegister data**
- **Analyze activity**

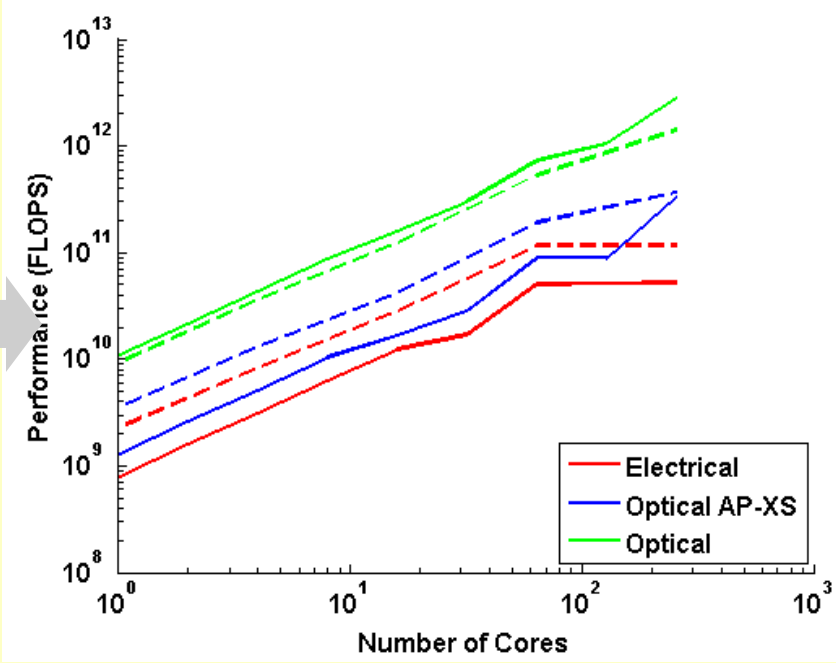
Processing chain simulated
- **projective transform with bilinear interpolation for each pixel**

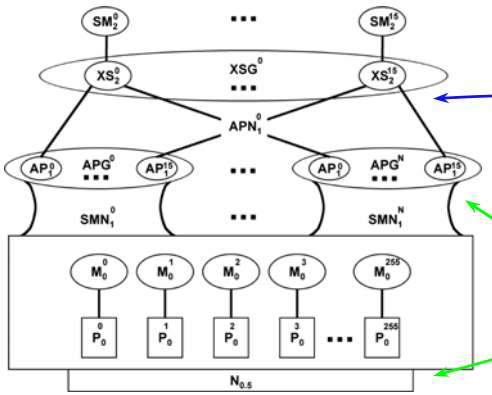
HPC Challenge relevance: STREAM and Random Access



PhotoMAN Performance





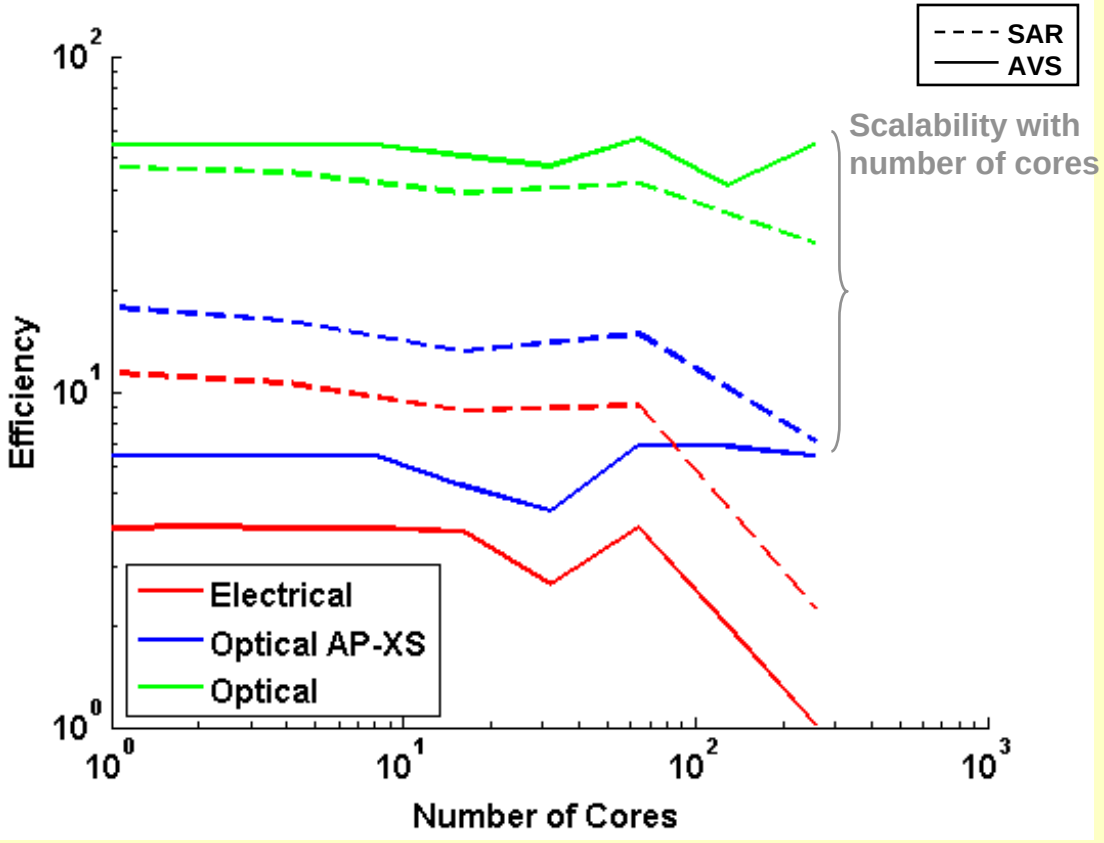


OPTICAL TO MEMORY BANKS (MIT/UCB)

OPTICAL MESH (COLUMBIA)

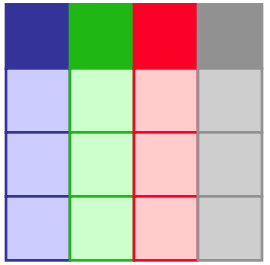
Optical (photonic) interconnects both to memory and between cores yield best performance

PhotoMAN Programmability



Maps selected:
- **1D Block**
- **Hierarchical**
- **Smallest block: fits into core's local store**

1D HIERARCHICAL



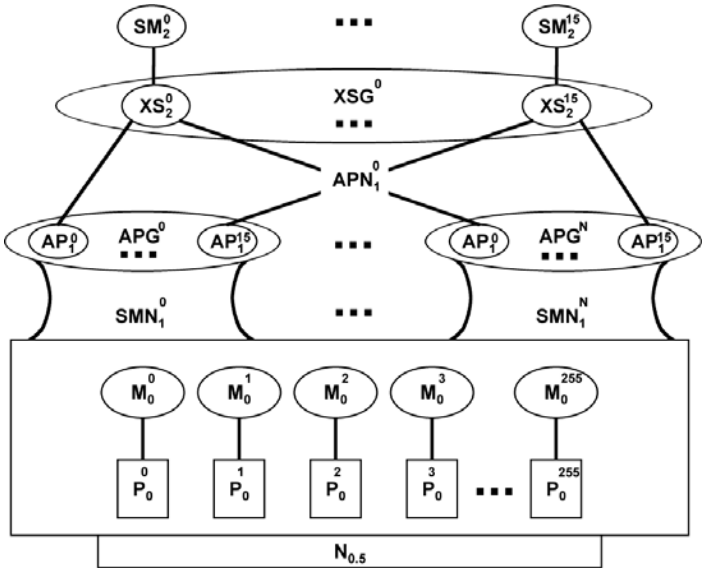
-
- **Requires hierarchical maps**
- ***Can be improved with cache architecture***

+
- **Architecture is well-balanced**
- **Maps with maximum number of cores are chosen (optical to memory and optical)**

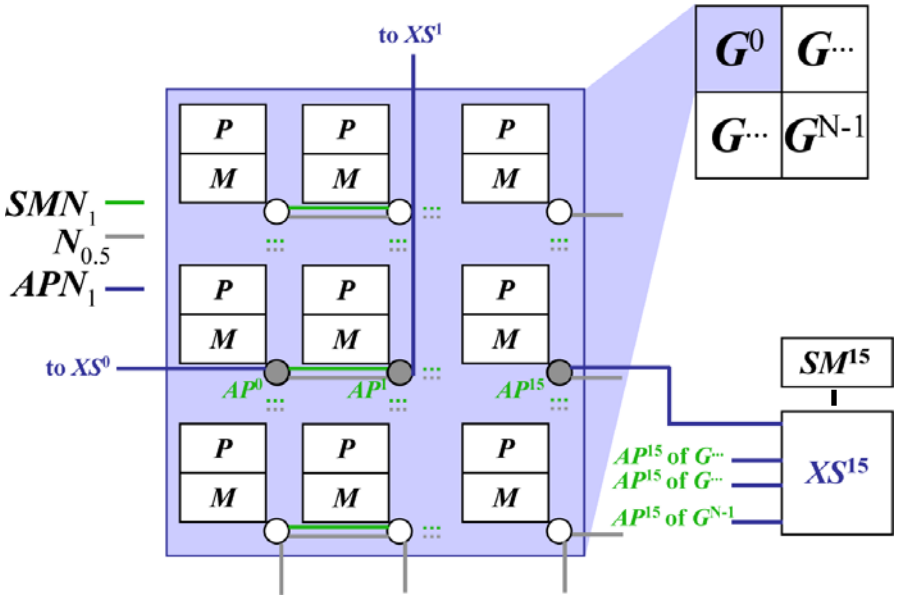
See J. Kepner and N. Bliss, "Evaluating the Productivity of a Multicore Architecture"

Best Performing Architecture

LOGICAL VIEW



PHYSICAL VIEW



- 16 groups
- Optical to memory
- Optical mesh
- 256 cores

Current/future research

- Network topology
- Power optimization
- Processor characteristics
- Cache architecture
- Hierarchical mapping

Summary

- Emerging device trends are motivating the need for logical architecture abstractions and robust modeling, mapping and simulation environments
- PhotoMAN study focus: photonic networks
- Kuck diagrams provide an expressive logical abstraction
- Detailed hardware model describes the mapping and modeling optimization space explored by pMapper and allows for architecture evaluation
- Initial results show over an order of magnitude improvement in *application* performance with photonics, while maintaining scalability