THE CENTER FOR AUTOMATA PROCESSING

UNIVERSITY of VIRGINIA

Performance and Power of Emerging Heterogeneous Architectures

Ke Wang
Mohamed Ezzat El-Hadedy Aly
Kevin Skadron

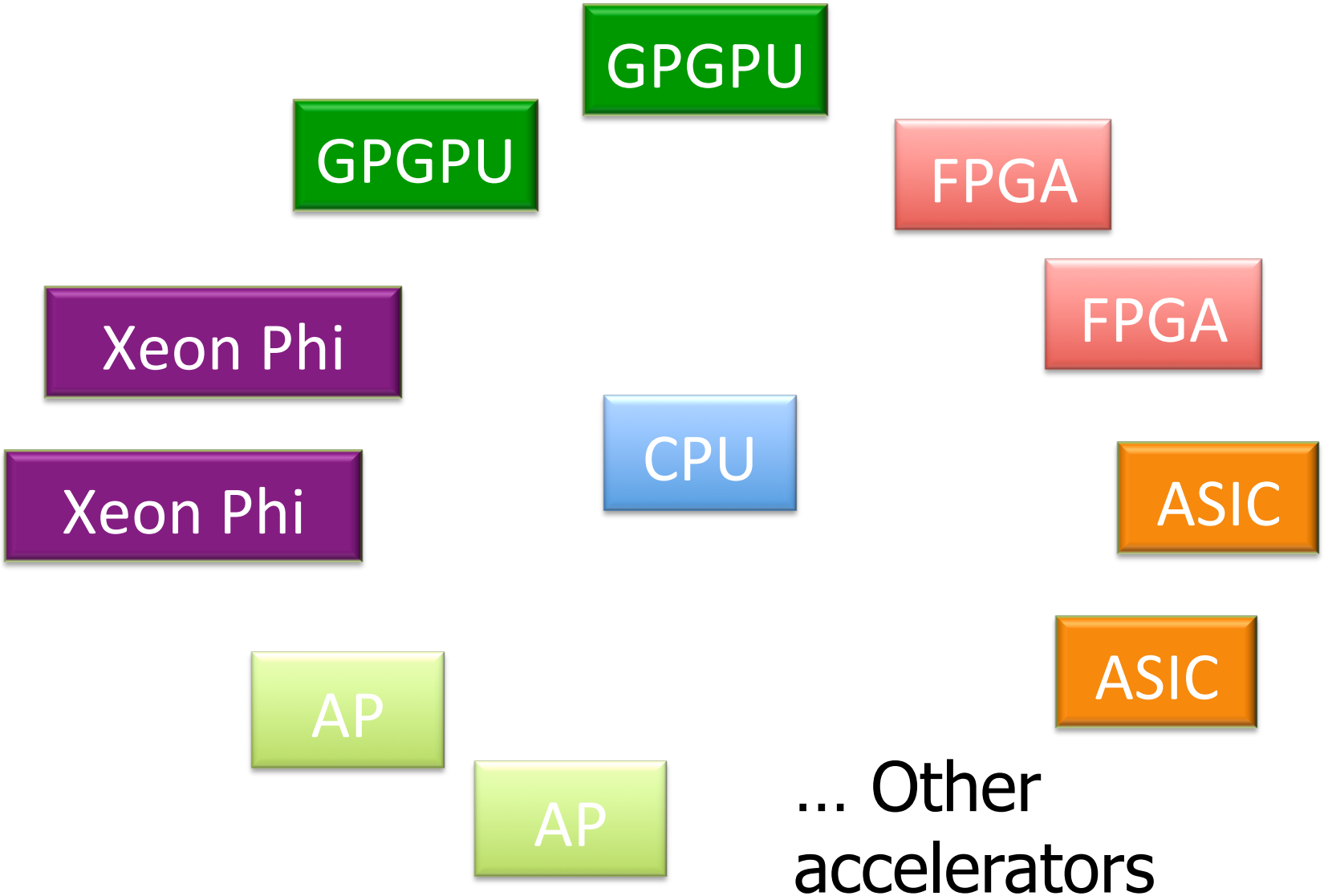
Center for Automata Processing
Department of Computer Science
University of Virginia

micron

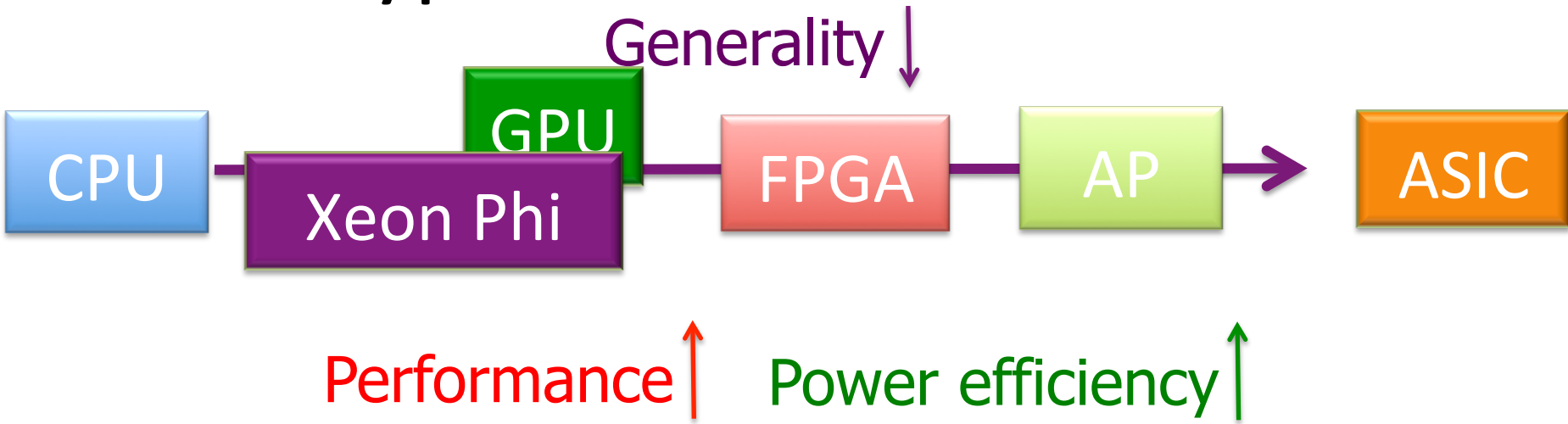
Outline

- Heterogeneous architectures
- Automata processor
- Field-programmable gate array (FPGA)
- 3D integration techniques

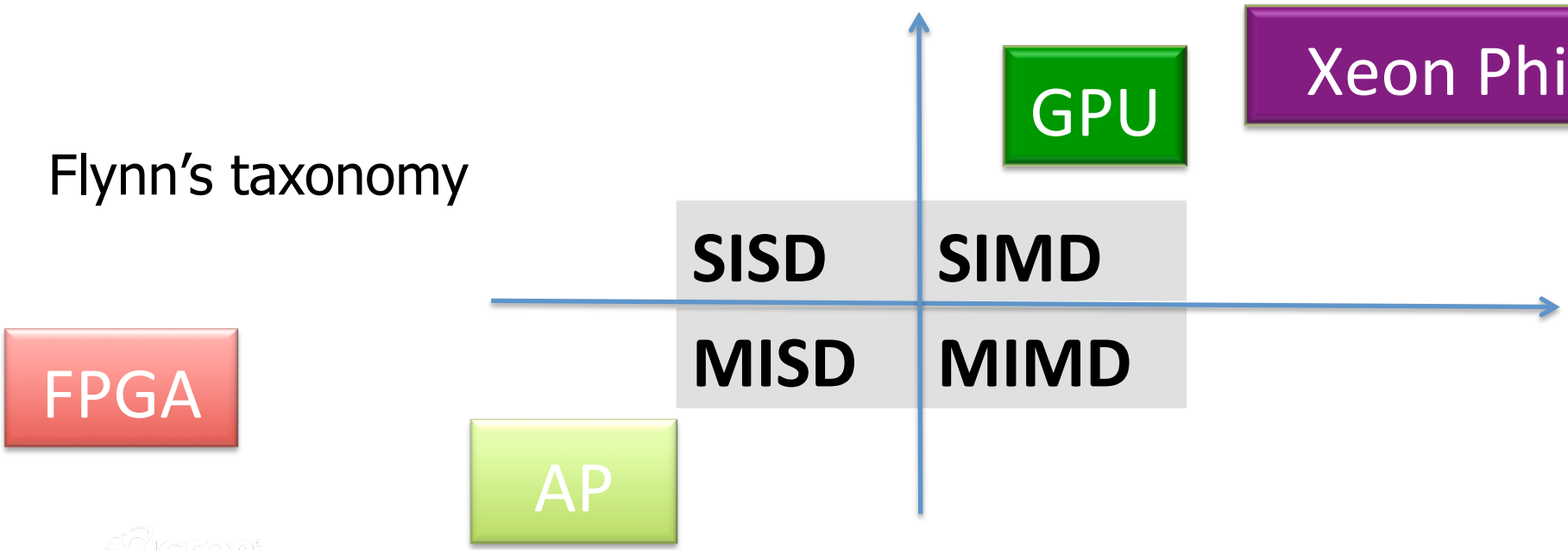
Heterogeneous Architecture



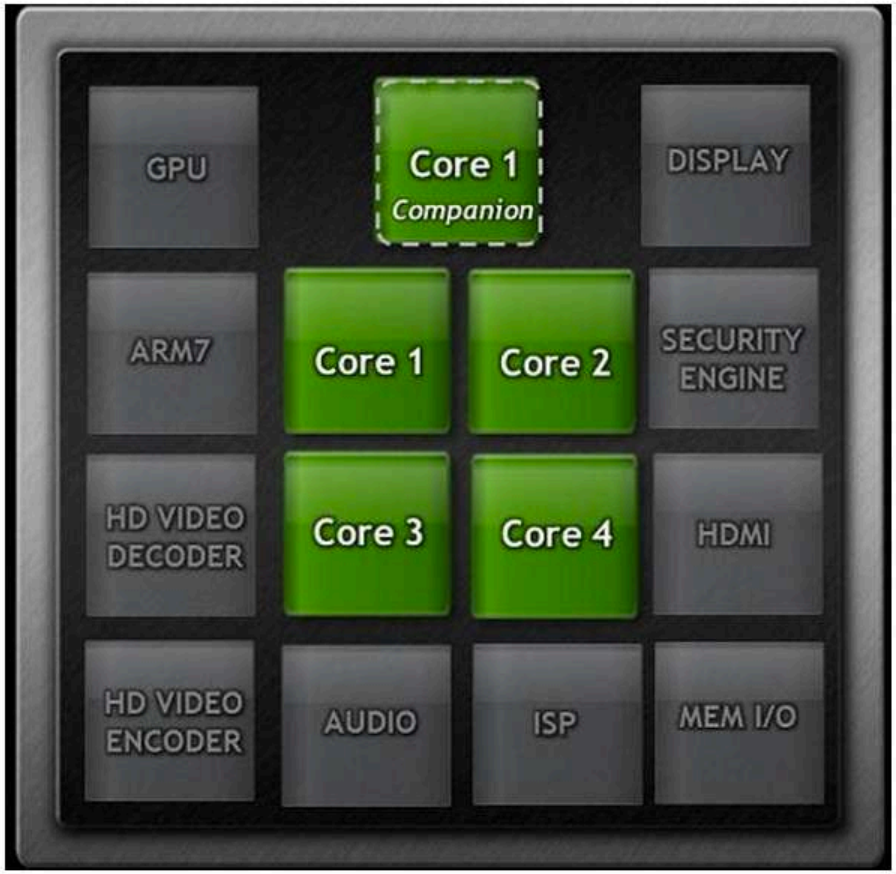
Types of Accelerators



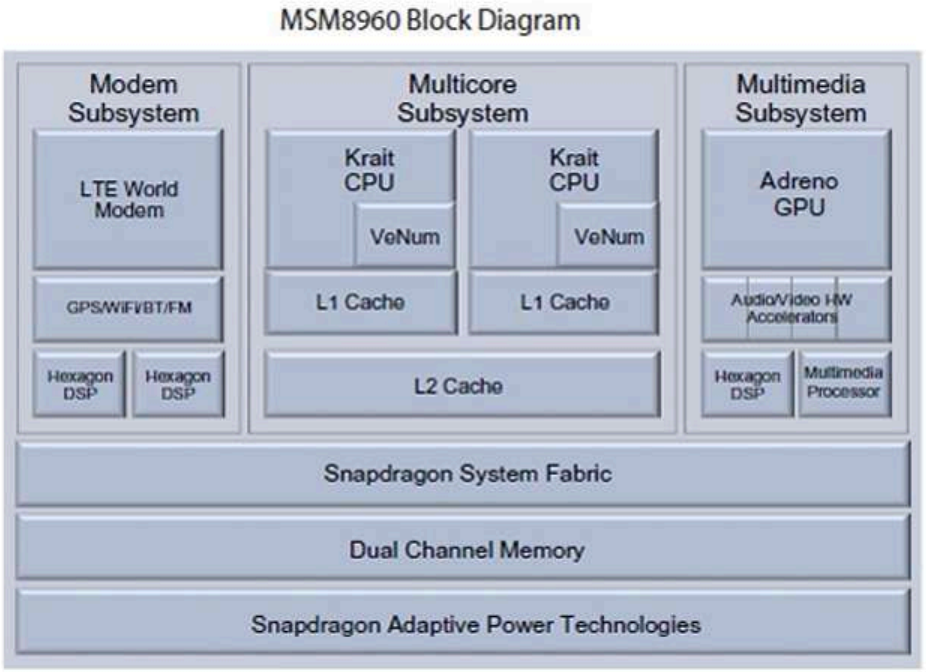
Flynn's taxonomy



Mobile -> HPC



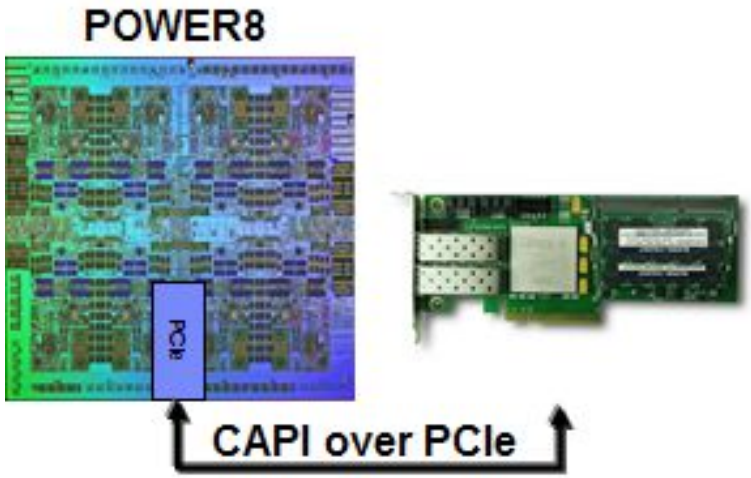
Nvidia Tegra 3 SoC

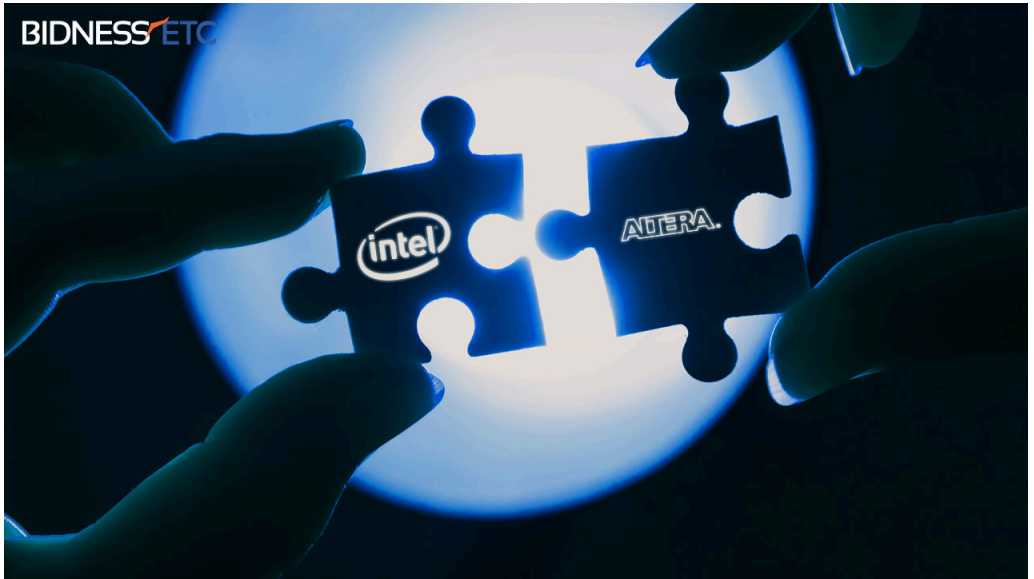


Qualcomm Snapdragon S4 SoC

Need better CPU support

- IBM's Coherent Accelerator Processor Interface (CAPI)

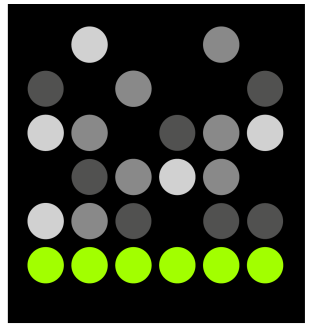




Introduction to Automata Processing

The Automata Processor (AP) is a programmable silicon device capable of performing very high-speed, symbolic pattern matching, allowing comprehensive search and analysis of complex, unstructured patterns in data streams.

- Hardware implementation of non-deterministic finite automata or NFA (plus some extra features)
- A massively parallel, scalable, reconfigurable, two dimensional fabric comprised of ~50,000 symbol processing elements per chip, each programmed to perform an independent pattern matching
- Exploits the very high and natural level of parallelism found in DRAM
- On-board FPGA allows sophisticated processing pipelines



AUTOMATA PROCESSING

Problems Aligned with the Automata Processor

- AP strengths
  - Complex/fuzzy pattern matching
  - Combinatorial search space
  - Highly parallel set of analysis steps for each input item
  - Unstructured data, unstructured communication
    - Esp. with high fan-out/fan-in
  - These challenges are common in “big data” analytics!
- AP limitations
  - No arithmetic, only counting (but on-board FPGA can help)
  - Changing the “program” requires a reconfiguration step

Problems Aligned with the Automata Processor

Applications requiring **deep analysis** of **data streams** containing **spatial** and **temporal** information are often impacted by the **memory wall** and will benefit from the **processing efficiency** and **parallelism** of the Automata Processor

Network Security:
- Millions of patterns
- Real-time results
- Unstructured data

Bioinformatics:
- Large operands
- Complex patterns
- Many combinatorial problems
- Unstructured data

Video Analytics:
- Highly parallel operation
- Real-time operation
- Unstructured data

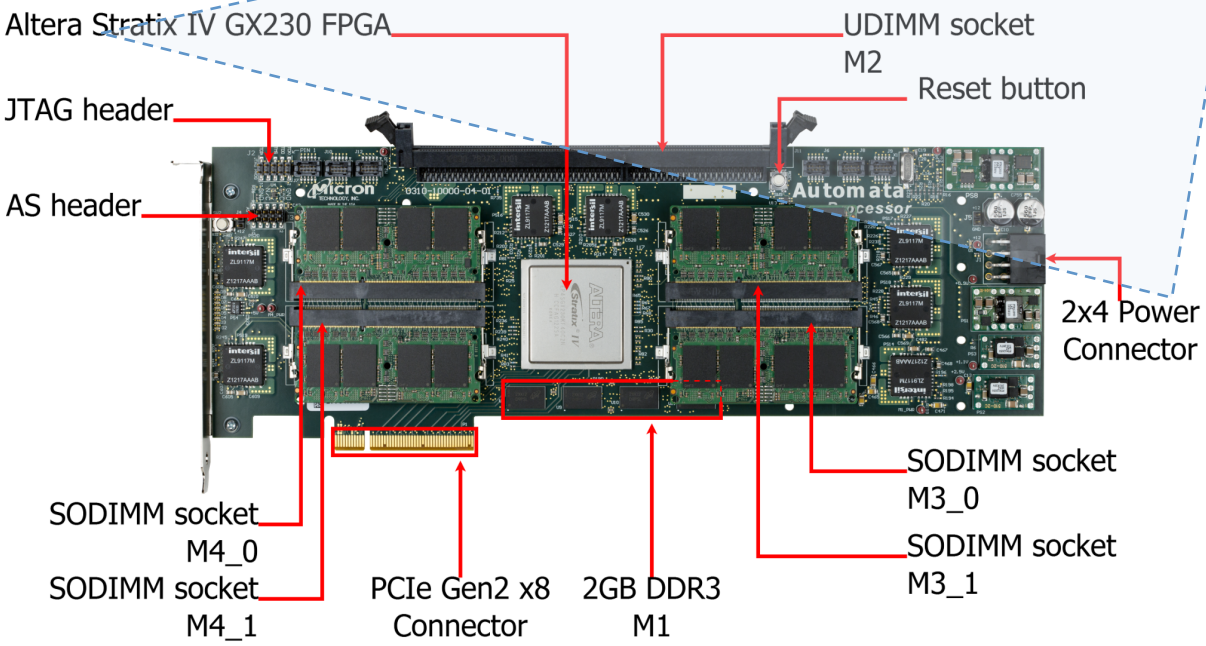
Data Analytics:
- Highly parallel operation
- Real-time operation
- Complex patterns
- Many combinatorial problems
- Unstructured data

So far: 10-100X+ speedups possible!

An overview of AP hardware

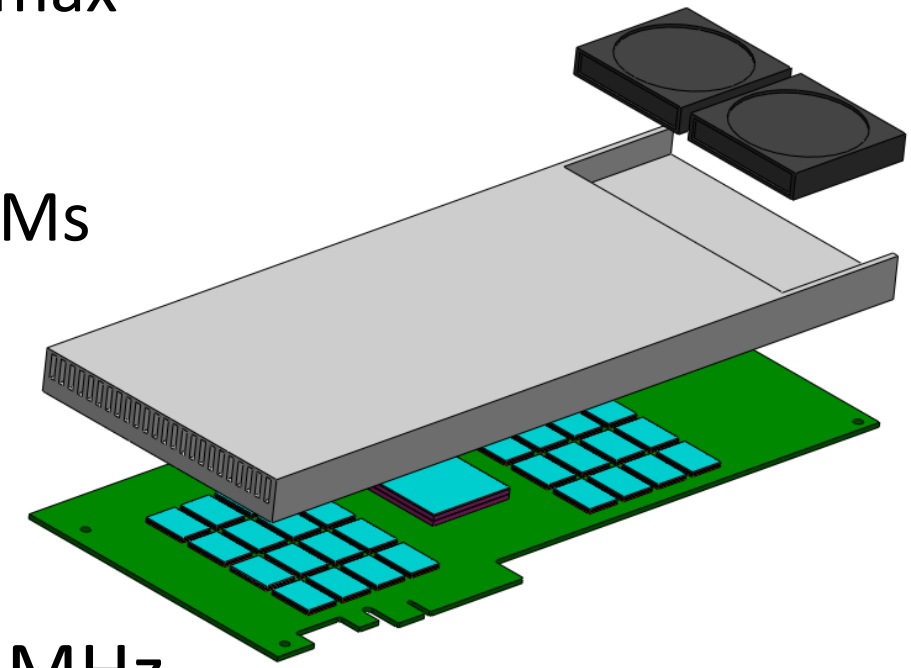
First AP Solution – Debug board

- PCIe board
- Four ranks = 32 total AP chips
- Installs into standard server

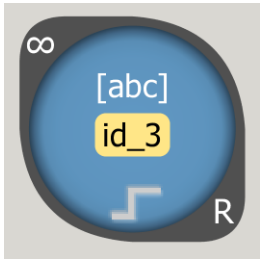
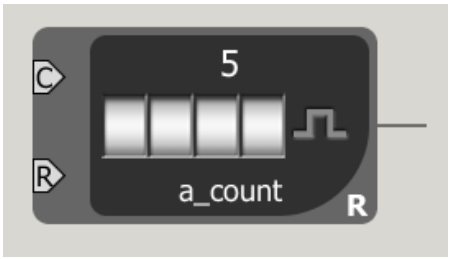
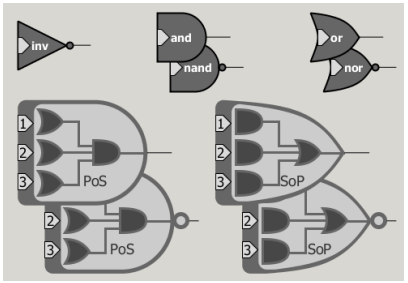


First Production Board

- PCIe Gen3 x8
  - 7.877 GBps is theoretical max
- Arria 10 GX270 FPGA
  - 270,000 LE's / 101,620 ALMs
  - 17,451 Kb Total Memory
- Four Ranks of AP
  - Soldered down
- 4GB DDR3 – 32bit @ 533 MHz

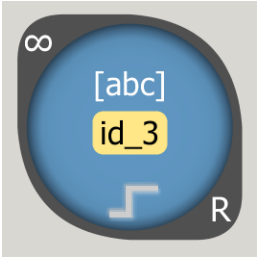


Automata Processor Hardware Functional Units

	GUI icon	per chip
State Transition Element (STE)		49,152
Counter Element		768
Boolean Logic Element		2,304
Reporting Property "virtual pins"		6,144

State elements / DRAM blocks

State Transition Element (STE)

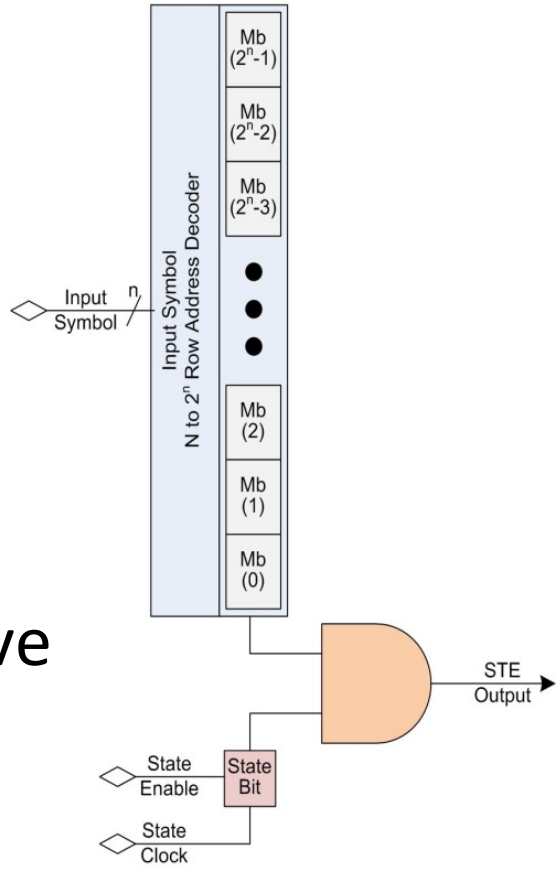


49,152/chip

- Holds state of machine
- Matches against 8-bit input symbols
- Can be programmed with multiple match values (subset of 8-bit symbol)

All STE process the input symbol – massive parallelism

Flynn's Taxonomy: MISD architecture



Reconfiguration Options

- Full-system programming takes around 50ms
- Partial-programming is available to facilitate quicker updates when only small changes are needed
  - Modifying STE match values
  - Modifying counter target values
  - Network topology remains the same

Brill Tagging on the Micron Automata Processor

- POS tagging: a task in NLP
- Grammatical tagging of words in text (corpus)
  - E.g.

    Keira loves dogs. -> POS Tagger -> Keira/noun loves/verb dogs/noun ./.
- Complicated:
  - E.g. I book tickets. -> book: Noun? Verb?
- Baseline tagging:
  - Tag each word to its most frequent tag based on training corpus
- Some standard tagsets:
  - Penn Treebank tagset [1], Brown Corpus tagset [2]

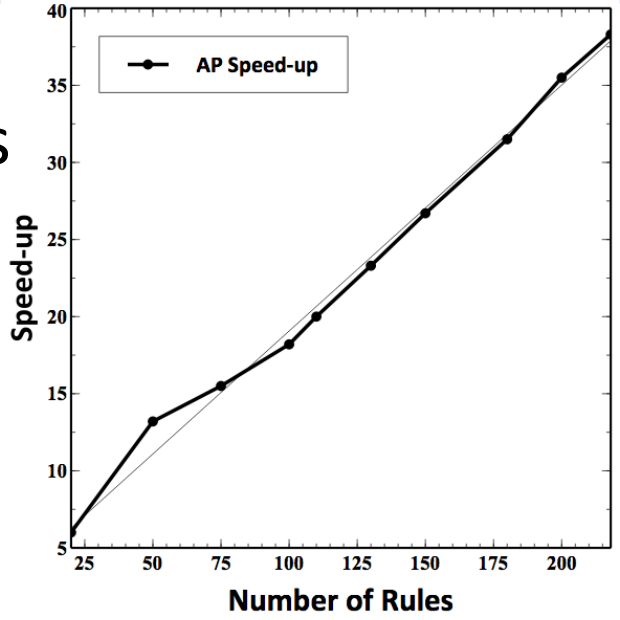
[1] Alphabetical list of part-of-speech tags used in the Penn Treebank Project: https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html
[2] Francis, W. Nelson & Kucera, H. (1979). Brown Corpus Manual. Brown University.

Brill Tagging on the Micron Automata Processor

- Comparing against C version developed by Brill [5]
- Tested on a file size 99KB (File size will NOT impact the speed-up)

(time in microsecs)	20 rules	75 rules	130 rules	180 rules	218 rules
CPU time	13687	48167	81187	113435	141810
AP time	2288	3104	3481	3601	3707
Speed-up	6.0X	15.5X	23.3X	31.5X	38.3X

- Linear Speed-up with No. of Rules
  - Processing all rules in parallel
  - Complexity: nKR for CPU vs. n for AP
- Known ruleset size: 1729
  - Projected speed-up: 276X



K. Zhou, J. J. Fox, K. Wang, D. E. Brown, and K. Skadron. "Brill Tagging on the Micron Automata Processor." ICSC '15

Association Rule Mining

Association rule mining (ARM, or frequent itemset mining, FIM):

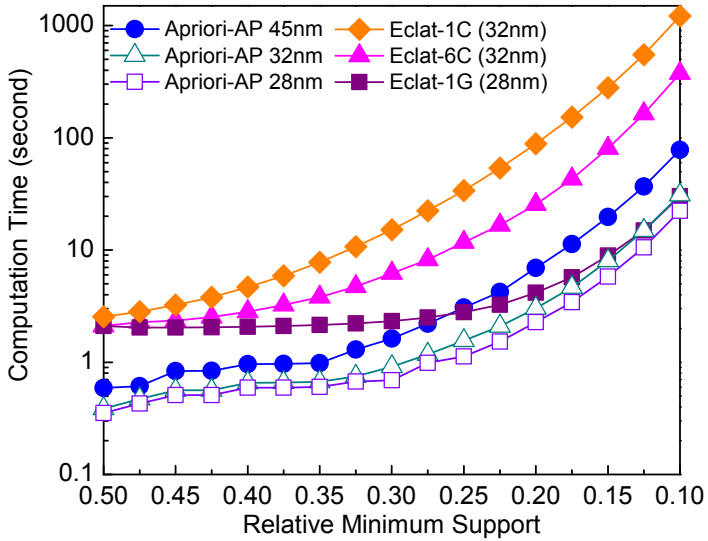
- Identify ***strong rules*** discovered in databases
- The order of items within a transaction doesn't matter

- Web usage mining
- Traffic accident analysis
- Intrusion detection
- Market basket analysis
- Bioinformatics

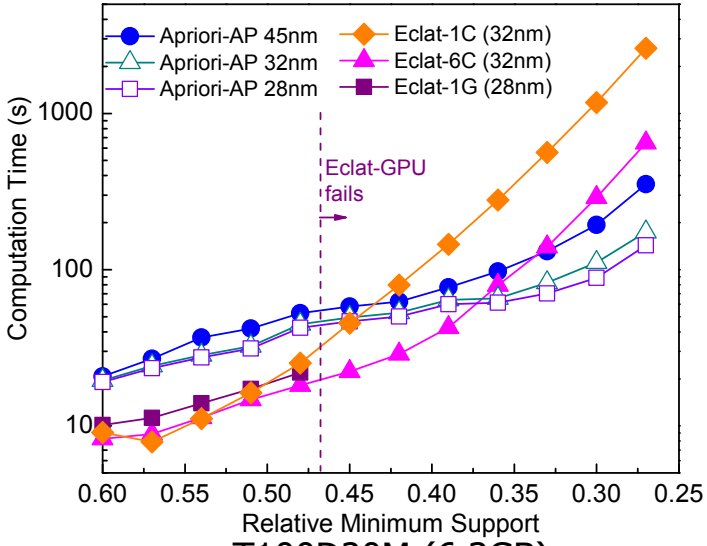
Trans.	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer, Coke
5	Bread, Milk, Diaper, Coke

Itemset
K–Itemset
Support: number of transactions which contain this itemset
sup({Diaper, Milk})= 3
Minimum Support: threshold to tell frequent or not

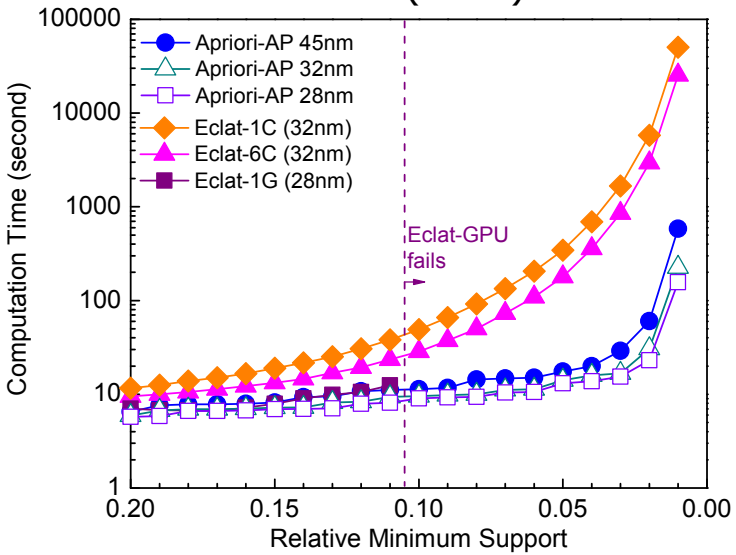
Performance Evaluation – vs. Eclat



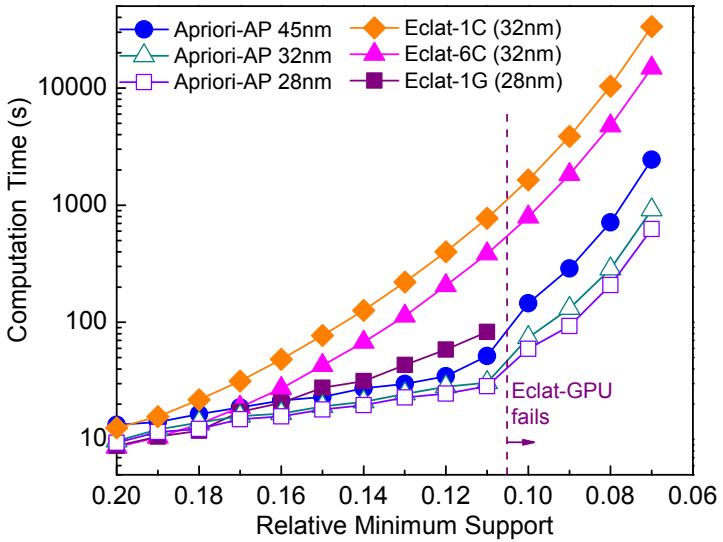
Accidents (34MB)



T100D20M (6.3GB)



ENWiki(3.0GB)



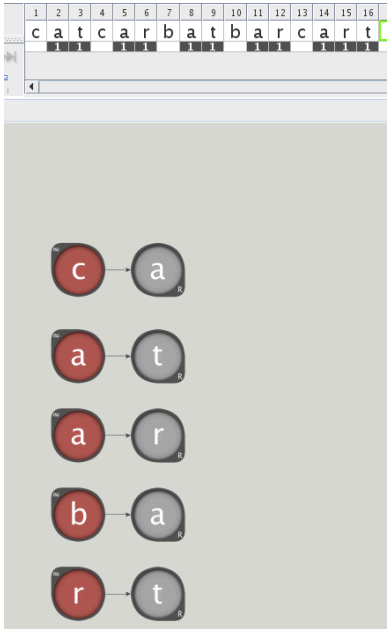
Webdocs5X (7.1GB)

K. Wang, J. Qi, J. J. Fox, M. R. Stan, and K. Skadron. "Association Rule Mining with the Micron Automata Processor." In Proc. IPDPS'15

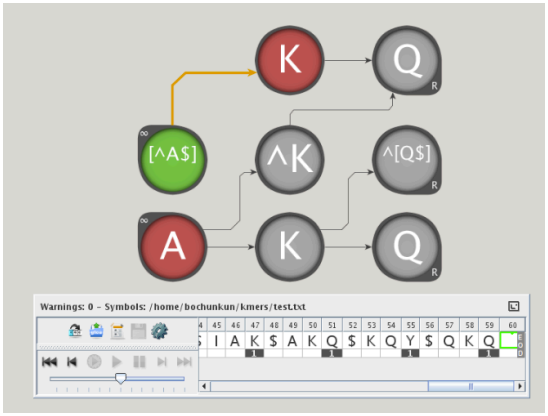
String Kernel

- String Kernel (SK) is a widely used kernel in machine learning and text mining
- Fast processing is required, especially for the testing phase
- Feature vector mapping is the current performance bottleneck, which involves a lot of pattern matching

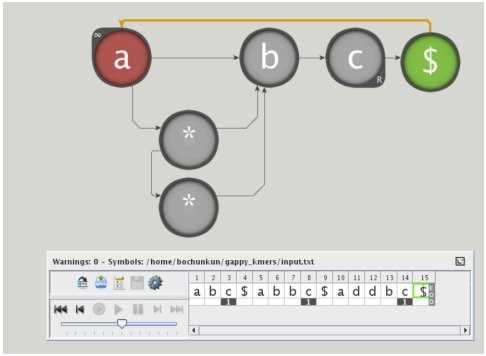
Using the AP to accelerate String Kernel Testing



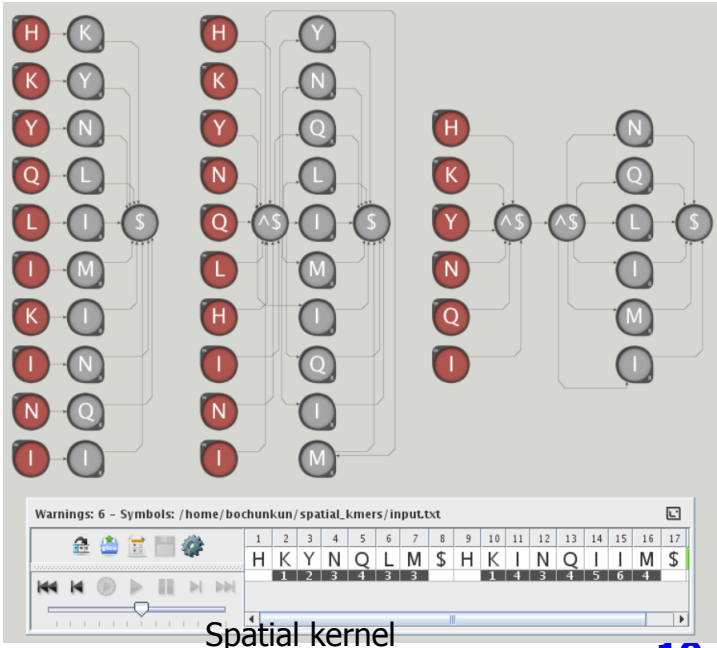
Exact kernel



Mismatch kernel



Gappy kernel

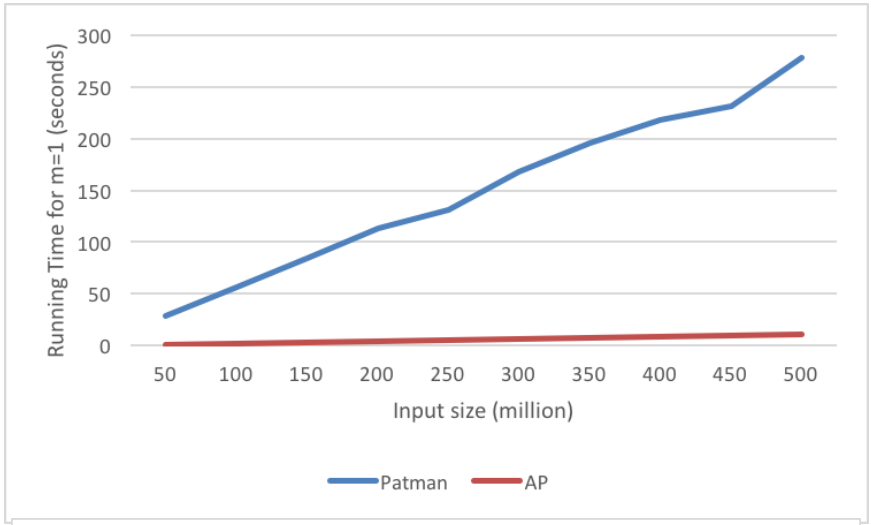


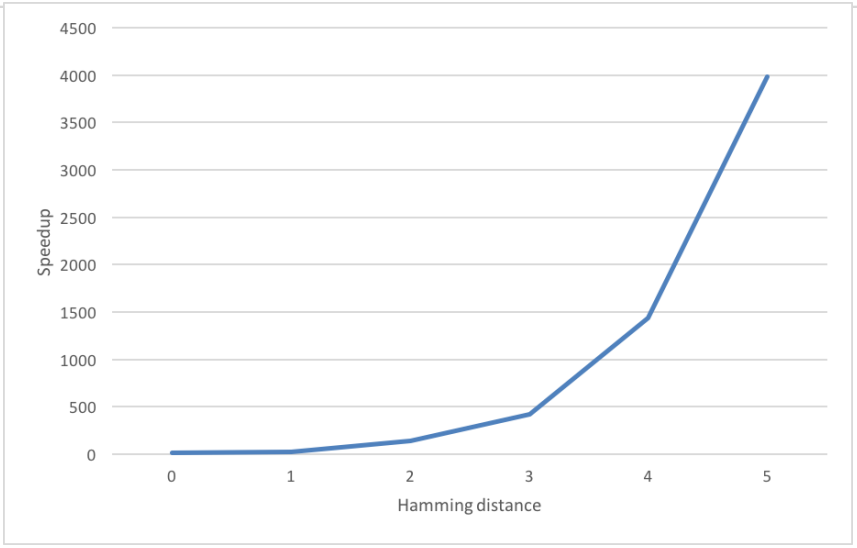
Spatial kernel

Performance Evaluation

- Both AP and PatMaN time increase linearly as input size increases
- PatMaN increases much more severely
- Different mismatch distances: similar trends
- Speedups increases exponentially

  8.5x ~ 3980x





C. Bo, K. Wang, Y. Qi and K. Skadron. "String Kernel Testing Acceleration using the Micron Automata Processor". The 1st International Workshop of Computer Architecture for Machine learning. (In conjunction with ISCA'15)

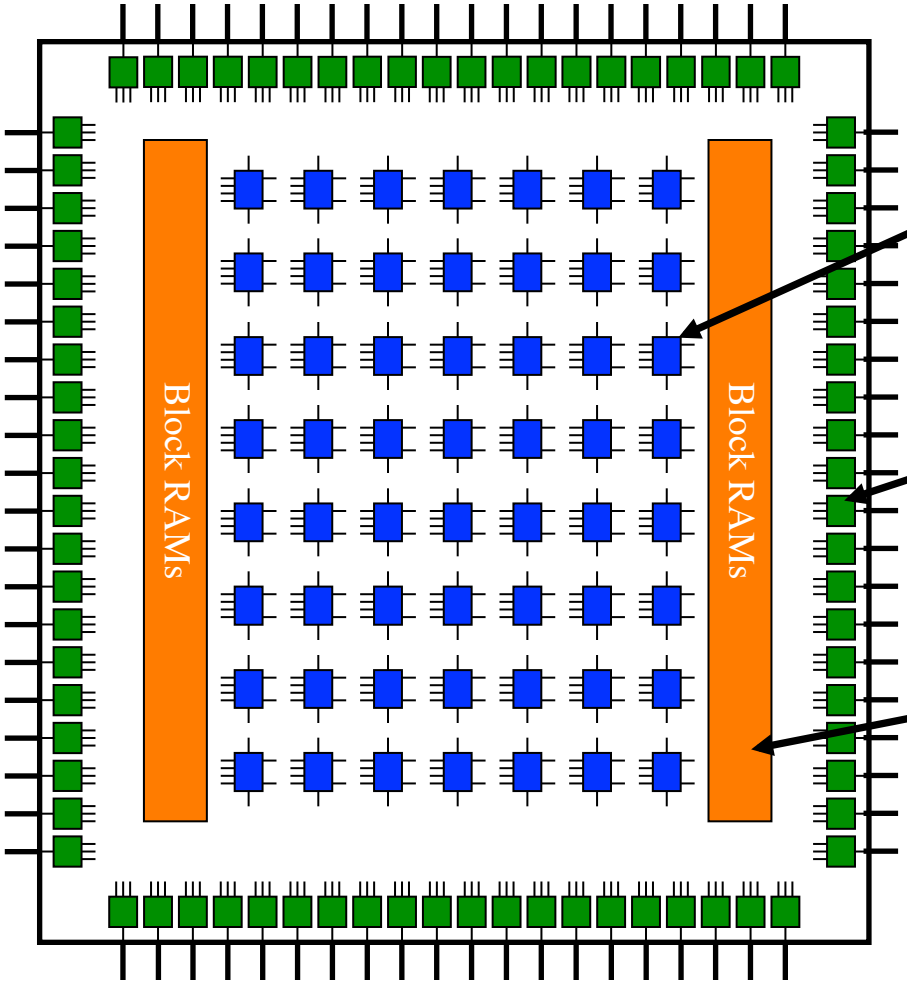
FPGA

- Field-Programmable Gate Arrays (FPGAs) are a recent kind of programmable logic devices. They allow the implementation of integrated digital electronic circuits without requiring the complex optical, chemical and mechanical processes used in a conventional chip fabrication

- FPGAs can be embedded in traditional system design flows to perform prototyping and emulation tasks. In addition, they also enable novel applications such as configurable computers with hardware dynamically adaptable to a specific problem

- FPGA: Circuits that can be modified or configured by users

FPGA structure



Configurable Logic Blocks

I/O Blocks

Block RAMs

- Applications of FPGAs

1. Aerospace and Defense
2. ASIC Prototyping (Using FPGA for proving a concept)
3. Broadcast
4. DSP (DCT, DWT, FFT, ….etc)
5. Security (Cipher)
6. Smarter Networks
7. Medical (custom processors for processing sensor data)
8. High Performance Computing
9. Embedded Systems

π-Cipher

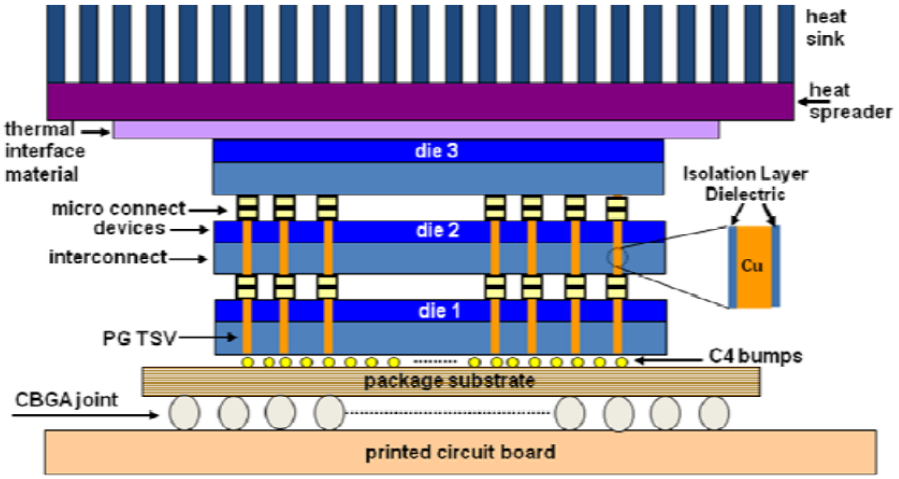
An authenticated encryption cipher with associated data

	PPE	Single Width	Double Width	Quad Width
Throughput	1.17 Gbps	3.09 Gbps	3.68 Gbps	4.22Gbps
Area (Slices)	227	445	447	634
Frequency (MHz)	250	254	243	245
Throughput/Area (Mbps/slice)	5.3	7.1	8.4	6.8

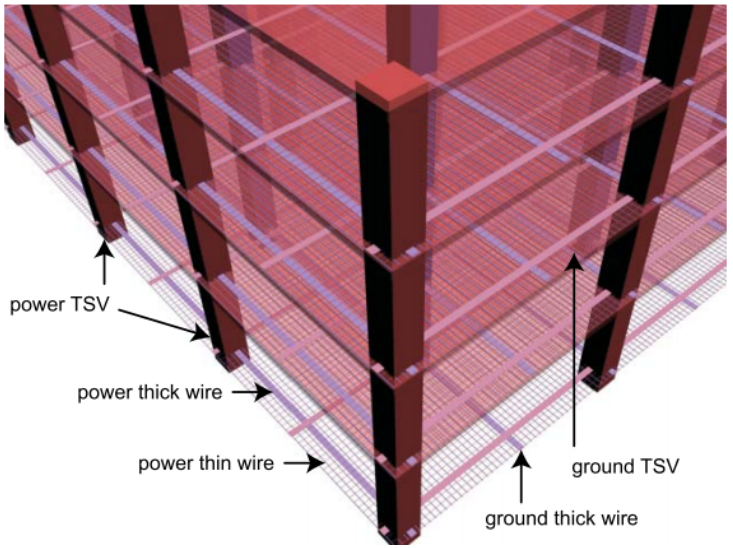
ARX Performance (64-Wide Pi-Cipher)
Xilinx Virtex 7 FPGA

- ***M. El-Hadedy, K. Skadron, H. Mihajloska, and D. Gligoroski, "Area Programmable Processing Element for Crypto-Systems on FPGAs," In Proc. HEART2015***
- ***M. El-Hadedy, K. Skadron, "Hardware Overhead Analysis of Programmability in ARX Crypto Processing", In Proc. HASP2015***

3D Integration



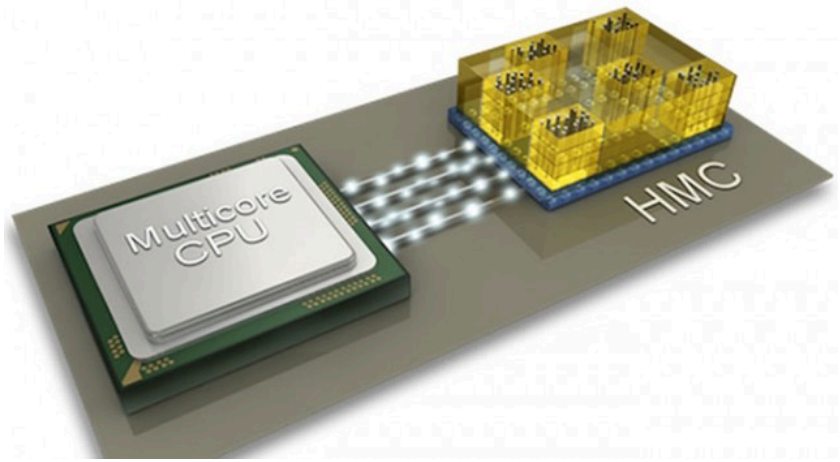
Example 3D system.
Source : N.H.Khan et al. 3DIC 2009



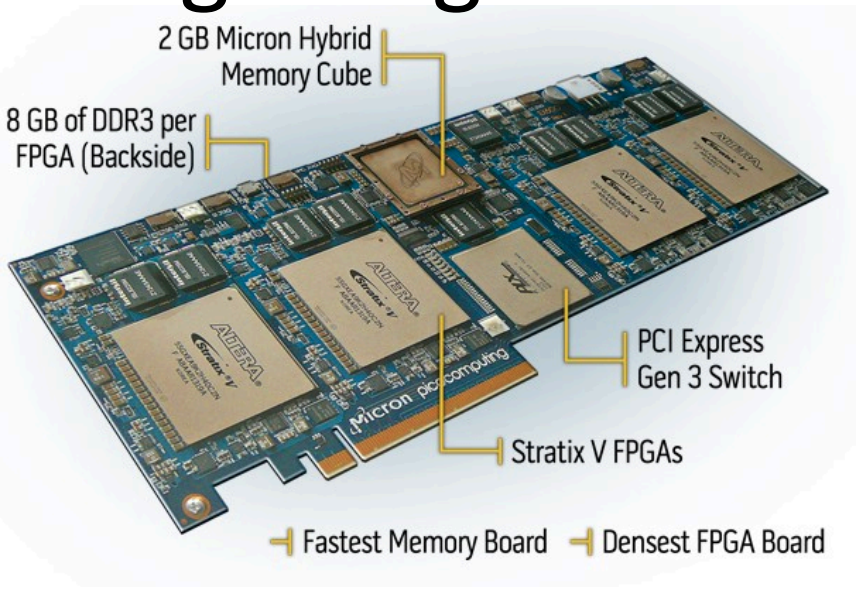
Wires and TSVs in a 3D PDN.
Source : M.B.Healy et al. ECTC 2010

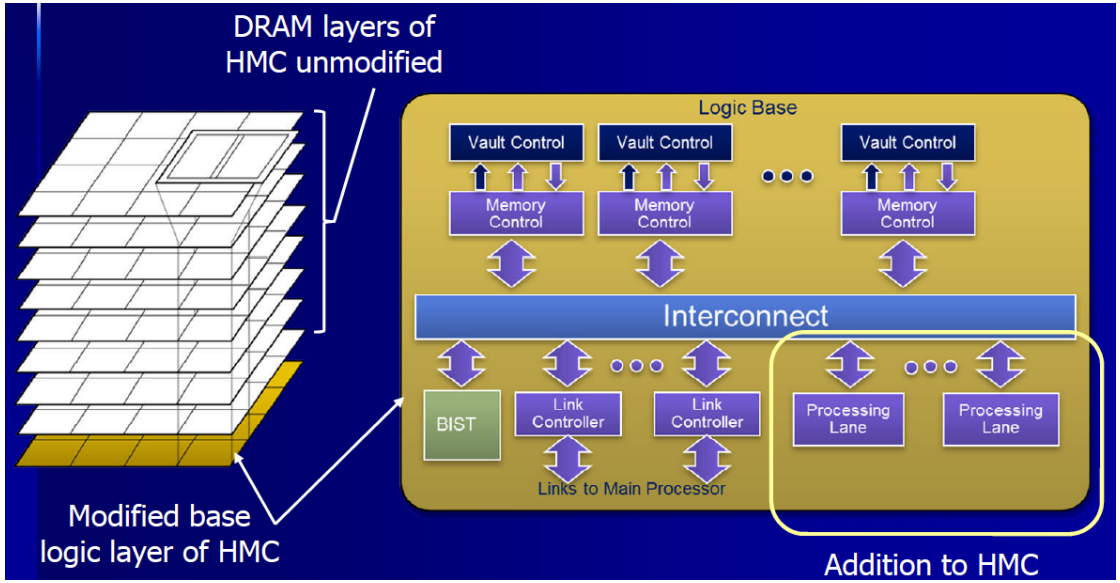
- Opportunities
  - increase bandwidth & reduce latency
  - Power efficiency
  - Heterogeneous system
- Challenges
  - Increased power density
  - Single side heat removal
  - Single side power delivery

Heterogeneous computing using 3D–IC



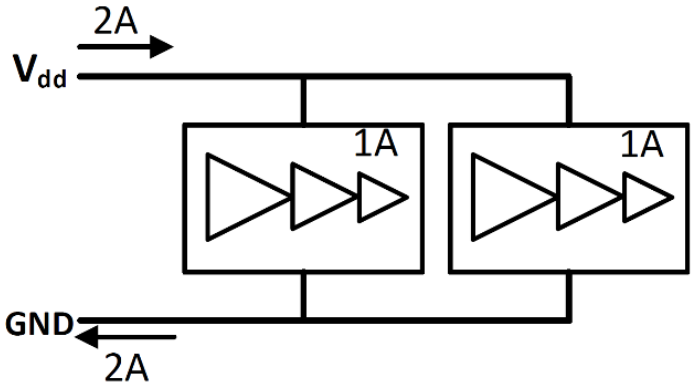
Micron's Hybrid Memory Cube



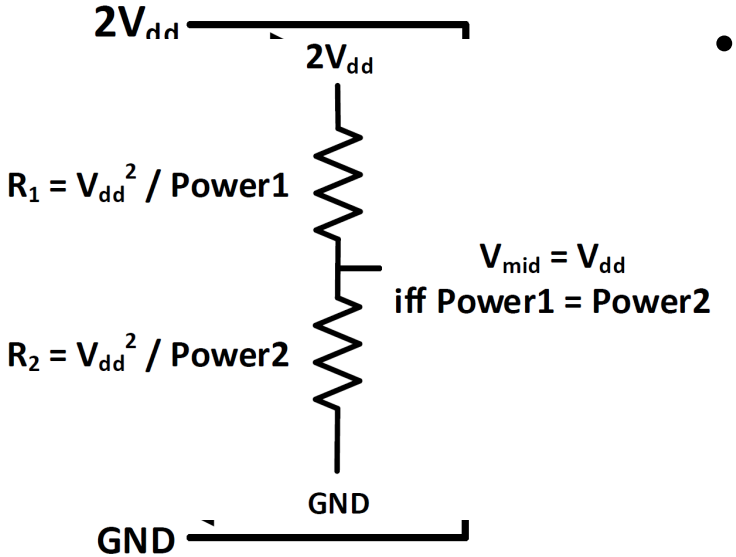


IBM's Active Memory Cube

Power-delivery for 3D-IC: Conventional vs. Voltage-Stacking



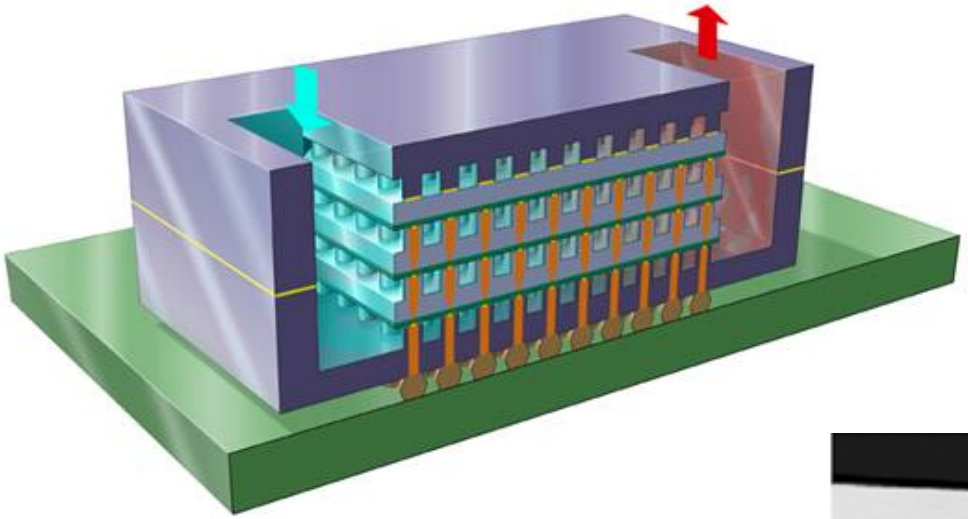
- Regular PDN (conventional)
  - Connects layers in parallel
  - Total off-chip current = Σ layer current
  - Mismatch: 3D consumption vs. 2D delivery

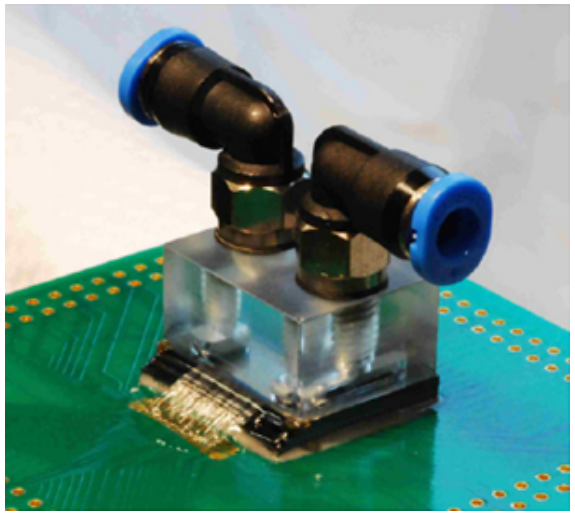


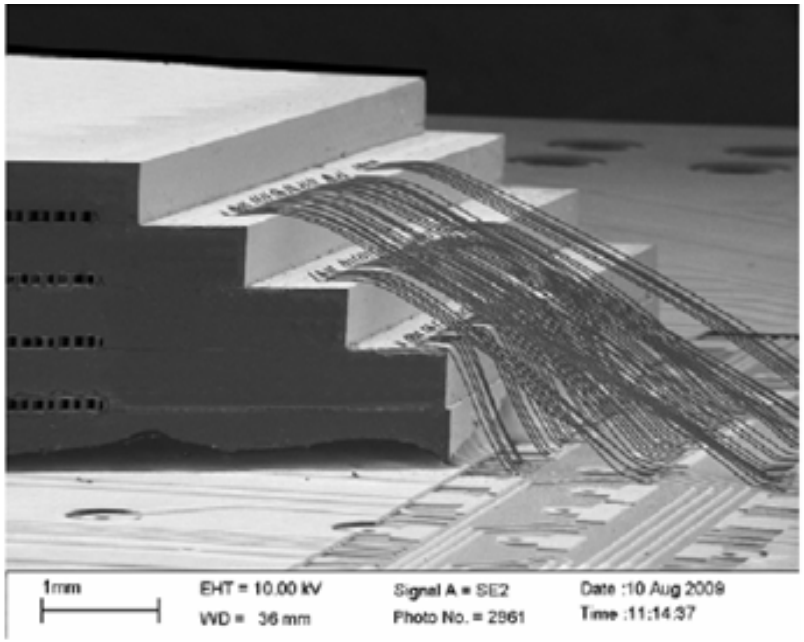
- Voltage-Stacking (V-S) PDN
  - Connects layers in series
  - Recycles current between layers
  - Supports arbitrary number of layers with constant current
  - Voltage noise when workloads are not perfectly balanced

- ***R. Zhang, K. Mazumder, B. H. Meyer, K. Wang, K. Skadron, and M. R. Stan. "Transient Voltage Noise in Charge-Recycled Power Delivery Networks for Many-Layer 3D-IC." ISLPED'15***
- ***R. Zhang, K. Mazumder, B. H. Meyer, K. Wang, K. Skadron, and M. R. Stan. "A Cross-Layer Design Exploration of Charge-Recycled Power Delivery in Many-Layer 3D-IC."***

Liquid cooling microchannel







Conclusion

- Heterogeneous architectures are promising solutions toward energy-efficient and high performance computing
- Automata processor accelerates complex pattern searching in big data
- FPGA shows broad range of application in digital data processing
- 3D−IC shows big potentials in increasing data transfer bandwidth and reducing latency
- Liquid cooling microchannel and innovative power delivery strategy migrate the thermal and power issues of 3D chips

Thanks!