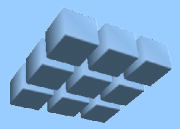
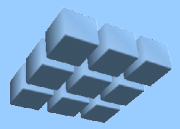


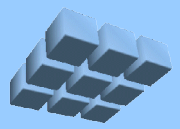
Microprocessors: The new LUT

Steven A. Guccione
Cmpware, Inc.



Digital System Design (2005)

- **The Design Gap**: circuit technology outpacing design productivity
- **Verification**: 70% of design effort
- **Power**: Power consumption becoming main limitiation in circuit design
- **The Performance Gap**: stark choices on hardware vs. software implementations
- **Fixed costs rising**: expect to spend $50M on ASIC design effort + mask sets

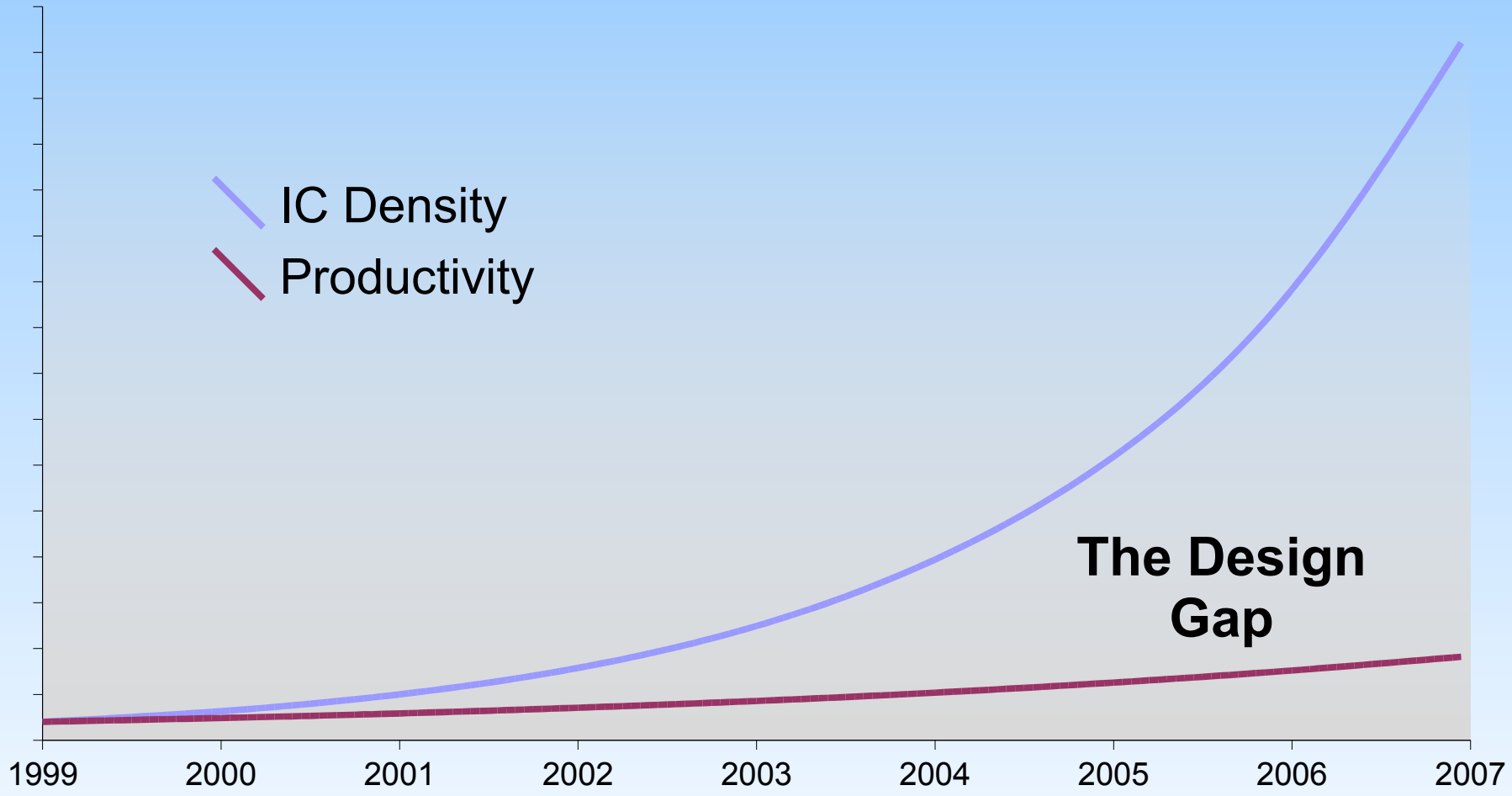


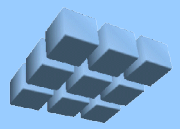
Design Productivity

- Logic density growth rate: 58% per year
- Productivity growth rate: 21% per year
- Increasingly difficult / expensive to take advantage of available circuit densities

The Design Gap: difference between circuit design productivity growth and actual circuit size growth.

The Design Gap

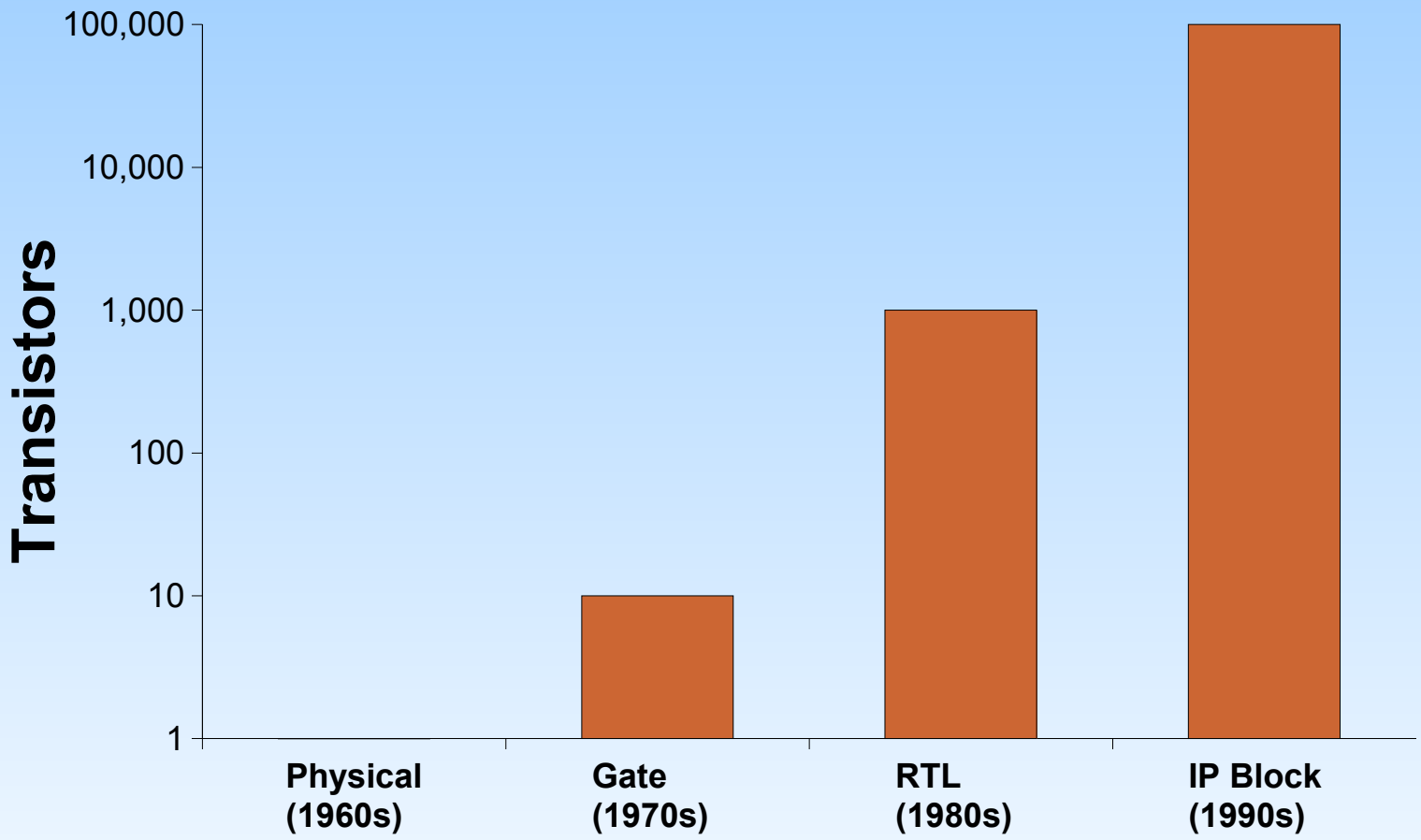




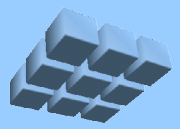
Design Abstraction Level

- Increasing circuit densities (“Moore's Law”) require periodic shifts to manage complexity
- Continually increasing level of abstraction:
  - 1960s: Physical / Transistor
  - 1970s: Gate
  - 1980s: RTL / Synthesis
  - 1990s: Intellectual Property (IP)
  - 2000s: ???

Design Abstraction Levels



* Log scale

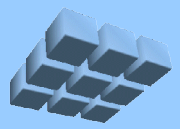


Verification

- Cost of verification increasing
- 70% of design effort now verification

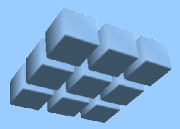
  ... even with large amounts of re-used IP

  ... and 50% of designs need a re-spin
- A re-spin usually involves
  - Re-design
  - Re-verification
  - A new mask set (as much as $2M)



Power Consumption

- Power has become the limiting design factor
- Large designs now have 'power budgets'
- Large devices can generate 100W+, more than can effectively be dissipated
- Power increasingly important for handhelds
- Heat generation a problem for data and communication centers
- All other technology trends make power problems worse

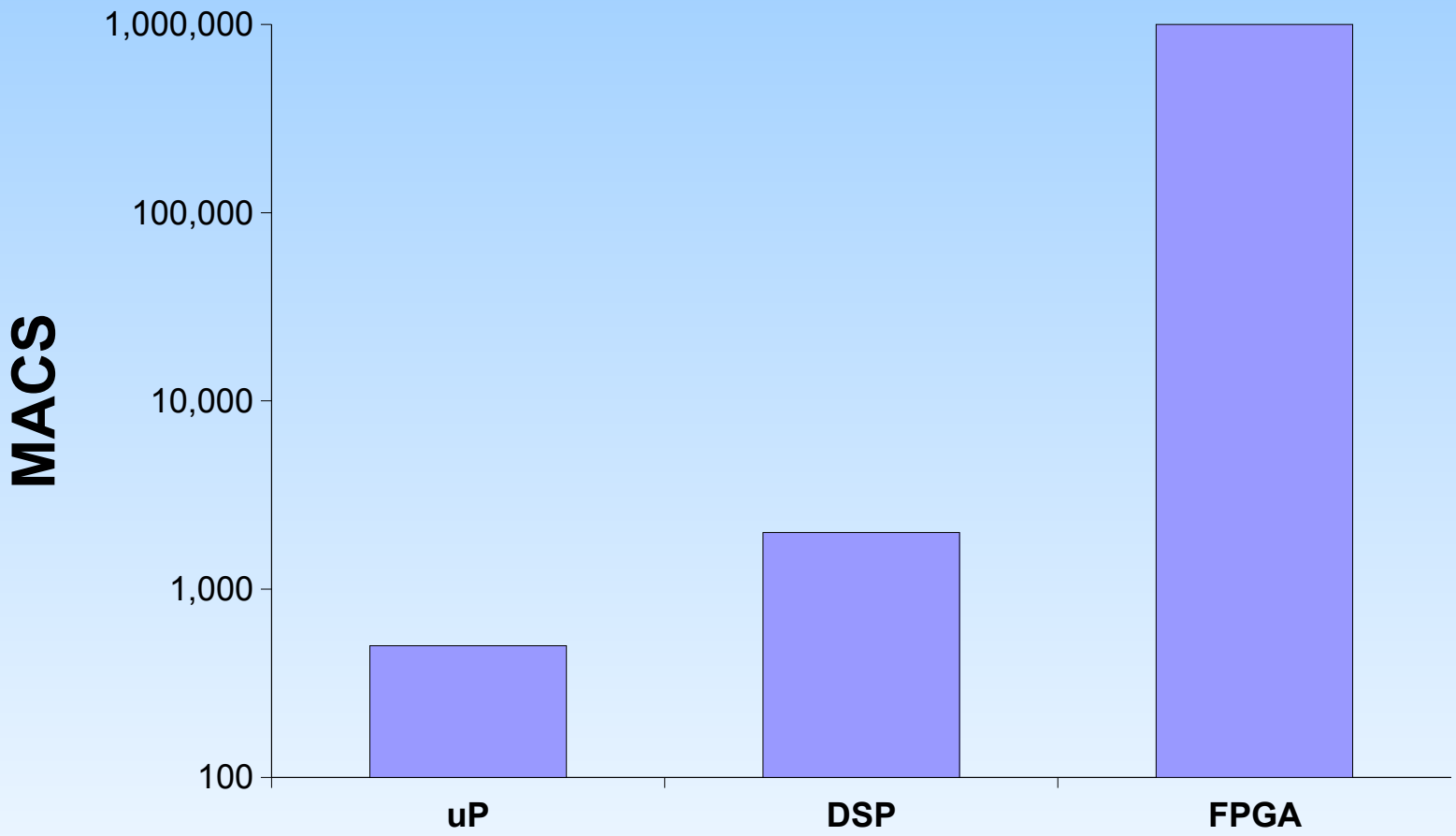


The Performance Gap

- HW typically 1,000x faster than SW
  - Custom circuitry: "billions and billions" of MACS
  - FPGA (*Xilinx Virtex II Pro*): 1 *trillion*+ MACS
  - DSP (*T.I. TMS320C6000*): 2 billion MACS
  - Microprocessor: 0.5 billion MACS
- Massive HW performance available
- No "middle ground" between HW and SW

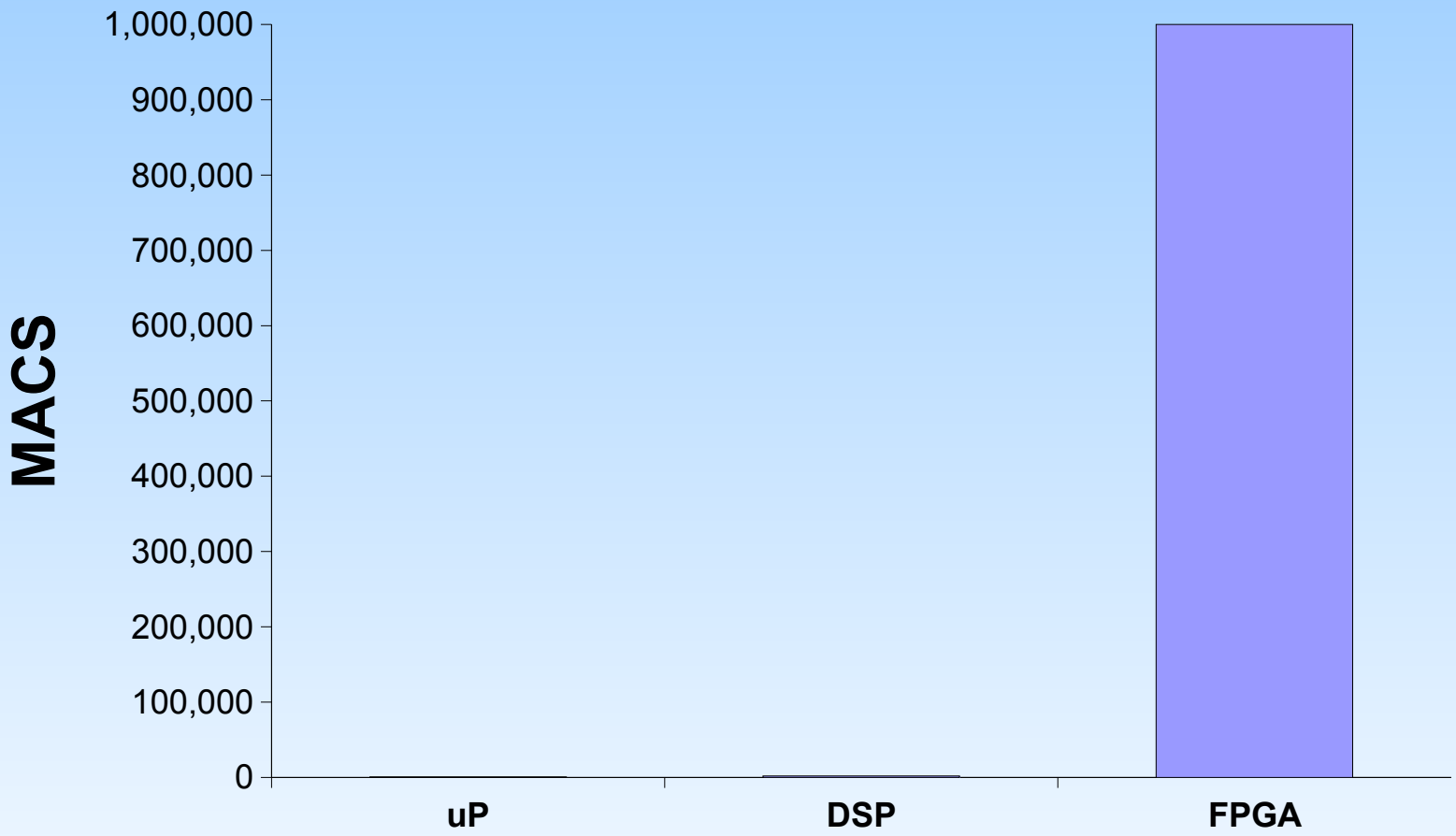
==> Suppose you only need 2x (or 10x)?

The Performance Gap*

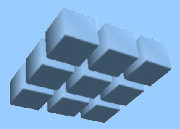


* Logarithmic scale

The Performance Gap*



* Linear scale

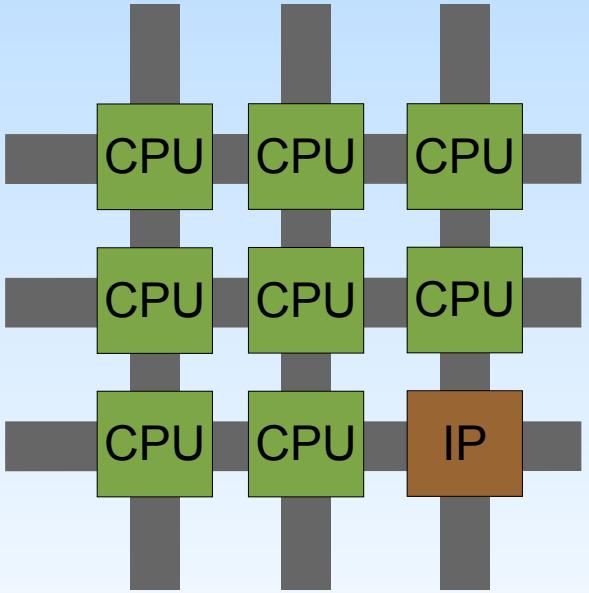


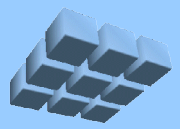
Emerging Design Solutions

- Requirements:
  - Manage 1B+ transistors
  - High degree of flexibility
  - Universality (a solution for everyone)
- Candidates:
  - Engineering System Level (ESL) design tools
  - Programmable Fabrics (FPGA / ALU Array)
  - Multiprocessors

Configurable Multiprocessing

- Thousands of CPUs on a die possible
- High performance: Millions of MIPS
- Simplifies HW design
  - Uses existing CPU cores
  - Minimizes custom circuitry
- Simplifies verification
  - All IP pre-verified
  - Simple interfaces
- Power efficient





Who is using CMP?

- CPU:
  - IBM Power PC 907MP
  - Sun SPARC (Rock, Niagara, Gemini)
  - Intel Pentium 4 HT ("*Hyperthreaded*")
- FPGAs:
  - Xilinx Virtex II Pro
  - Multiple soft CPU cores
- ASICs:
  - *Averaging* 6 CPU cores per design
  - Hundreds of CPUs reported

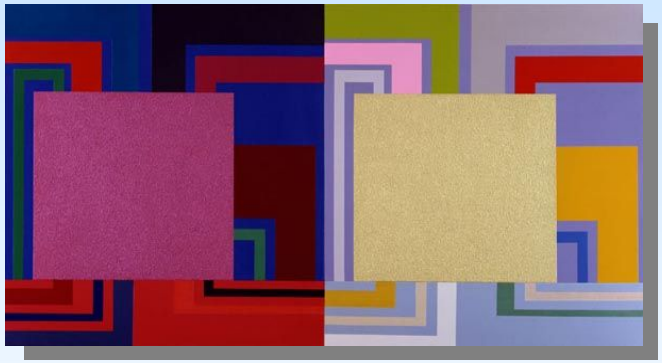
CMP Software

- Burden shifted from HW to SW
  - CMP hardware relatively simple
  - CMP software / tools more difficult
- High bandwidth on-chip interconnect
  - Rapid inter-processor communication (1 cycle)
  - Wide data paths
  - Rapid processor synchronization
- Can exploit sub-task level parallelism

==> *Resembles HW design effort*

Converting HW to SW

- SW cheaper and easier to produce than HW
- SW more flexible than HW
- More existing software IP
- SW can be ***upgraded***, even in fielded units
  - Fix design errors (bugs)
  - Increase performance
  - Lower power consumption
  - Enhance functionality



Hardware Reuse

- Fixed circuits require HW for each function

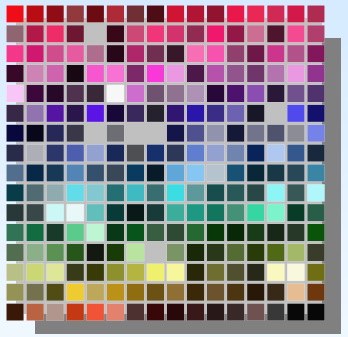
  ... even when they aren't being used

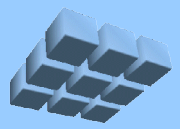
- CMP CPUs can be re-programmed
  - Unused functions swapped out
  - Example: MP3 ==> WMA ==> MPEG4

==> *Reprogrammable HW may be smaller than fixed HW for same application*

Reconfigurable Computing

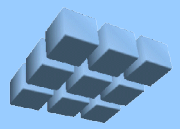
- Reconfigurable Architecture building block:
  - 1980s: LUTs (Altera, Xilinx, etc.)
  - 1990s: ALUs (Chameleon, PACT, Elixent)
  - 2000s: Microprocessors
- Configurable Multiprocessing:
  - HLL programmable (C/C++, Java, etc.)
  - Quickly and easily reprogrammed
  - High performance
  - Even 'fast context switchable'





The Cmpware Approach

- CPU + Compiler = programmable “black box”
  - “Pluggable” processor core model
  - Uses variety of processors (even mixed)
- Simulation-based
  - Early SW development
  - No waiting for HW development to complete
- Simulate processors, not gates
  - Simplified modeling
  - Approx. 10,000x faster simulations

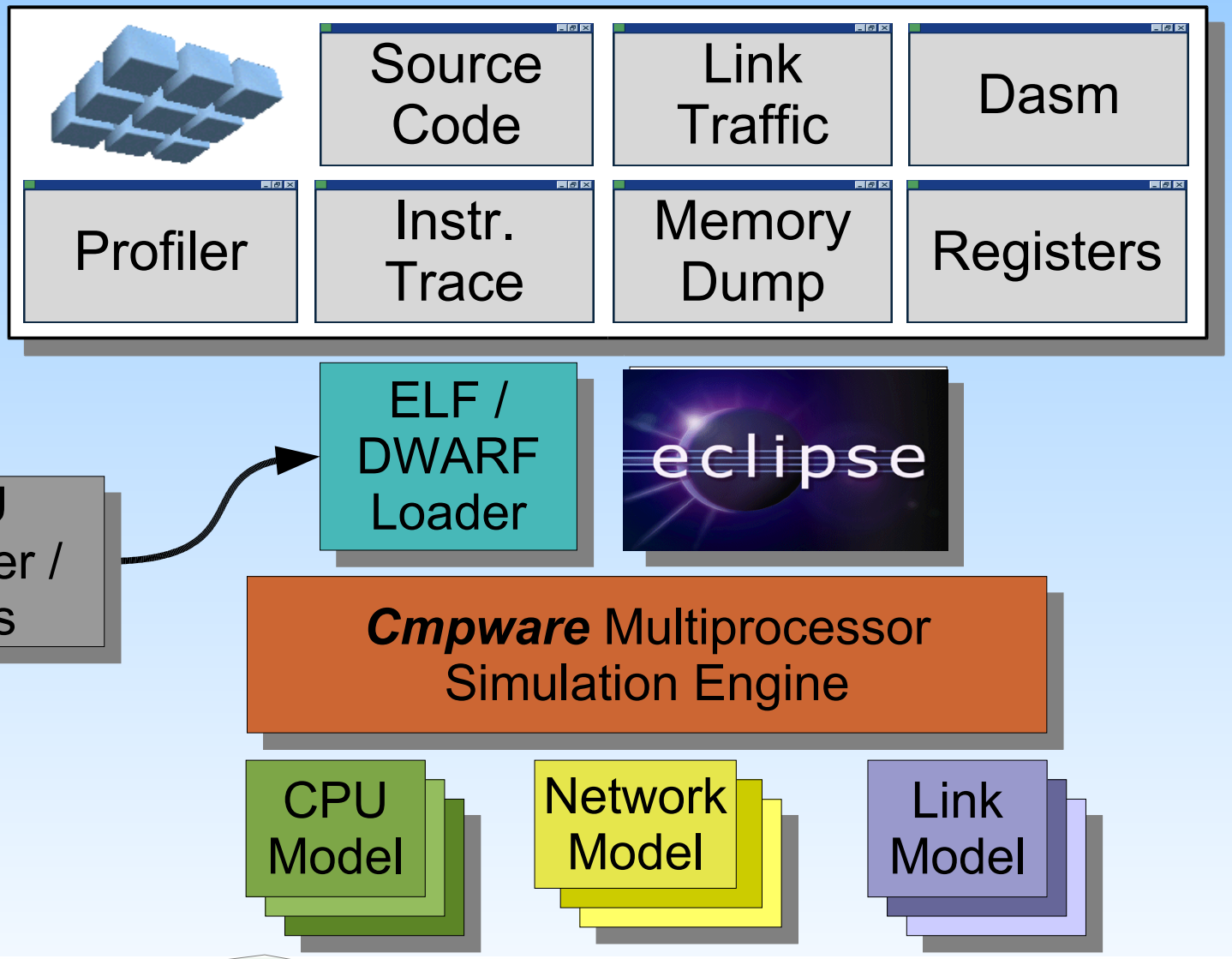


The Cmpware Toolkit

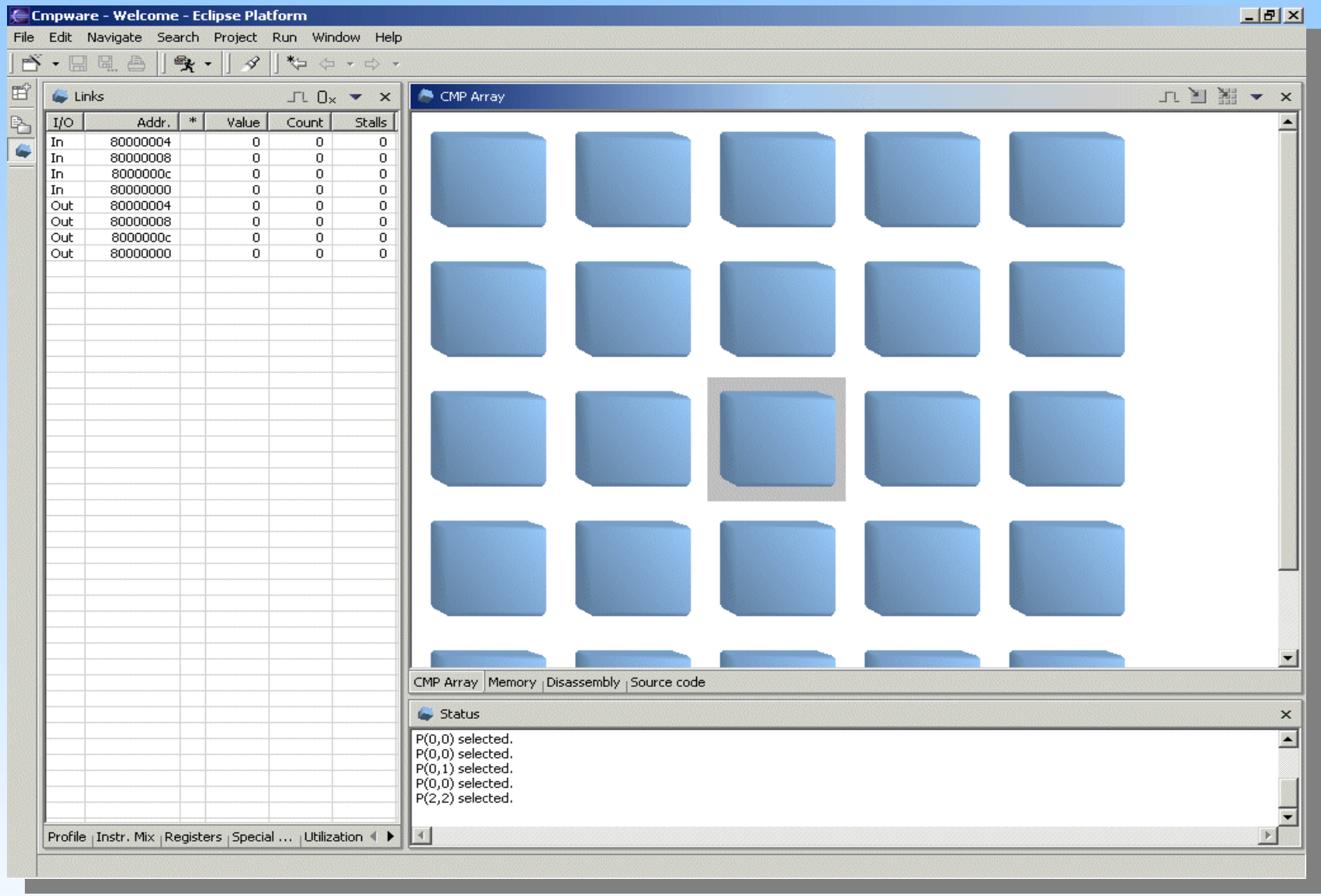
- Models / interfaces for:
  - Processors
  - Network
  - Links
  - Hardwired logic
- Common displays
  - Execution profile
  - Instruction trace
  - Source code display
  - Disassembly
  - Memory Display, Etc.

==> *enables early high-quality architecture and software development*

The Cmpware Toolkit

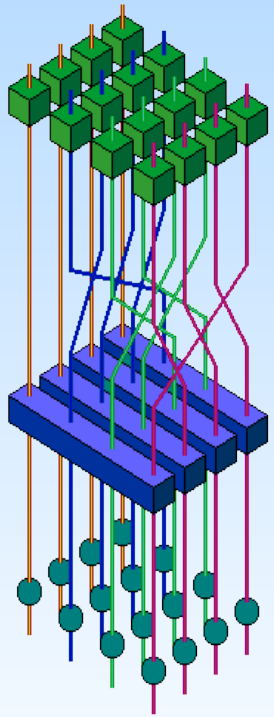


The Cmpware CMP-DK



Example: AES

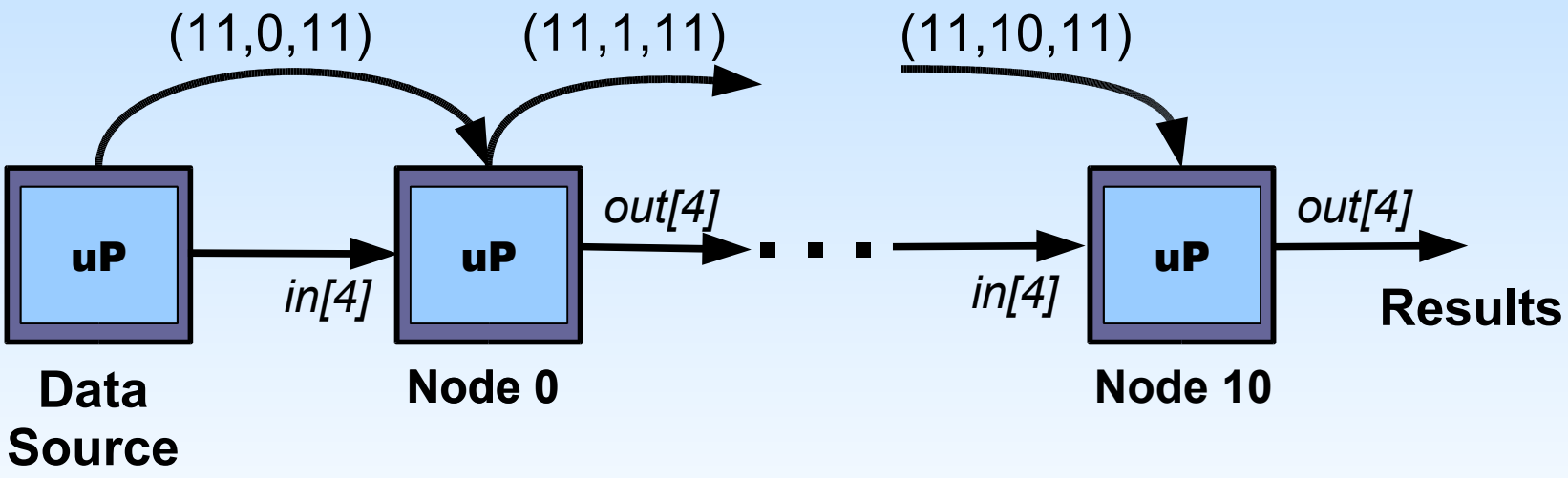
- AES: Advanced Encryption Standard
- US NIST program to replace the Data Encryption Standard (DES)
- Selected 'Rijndael' algorithm in 2000
- Efficient HW and SW implementations
- 128, 192 or 256 bit keys
- Operates on 128, 192 or 256 bits of data

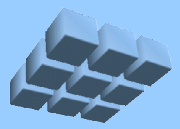


AES Implementation

- 128 bit key / 11 'rounds' / <N> processors
- Parameters: (*processors*, *thisNode*, *rounds)*
- Interface:

```
int round(int r, uint32 in[4], uint32 out[4]);
```

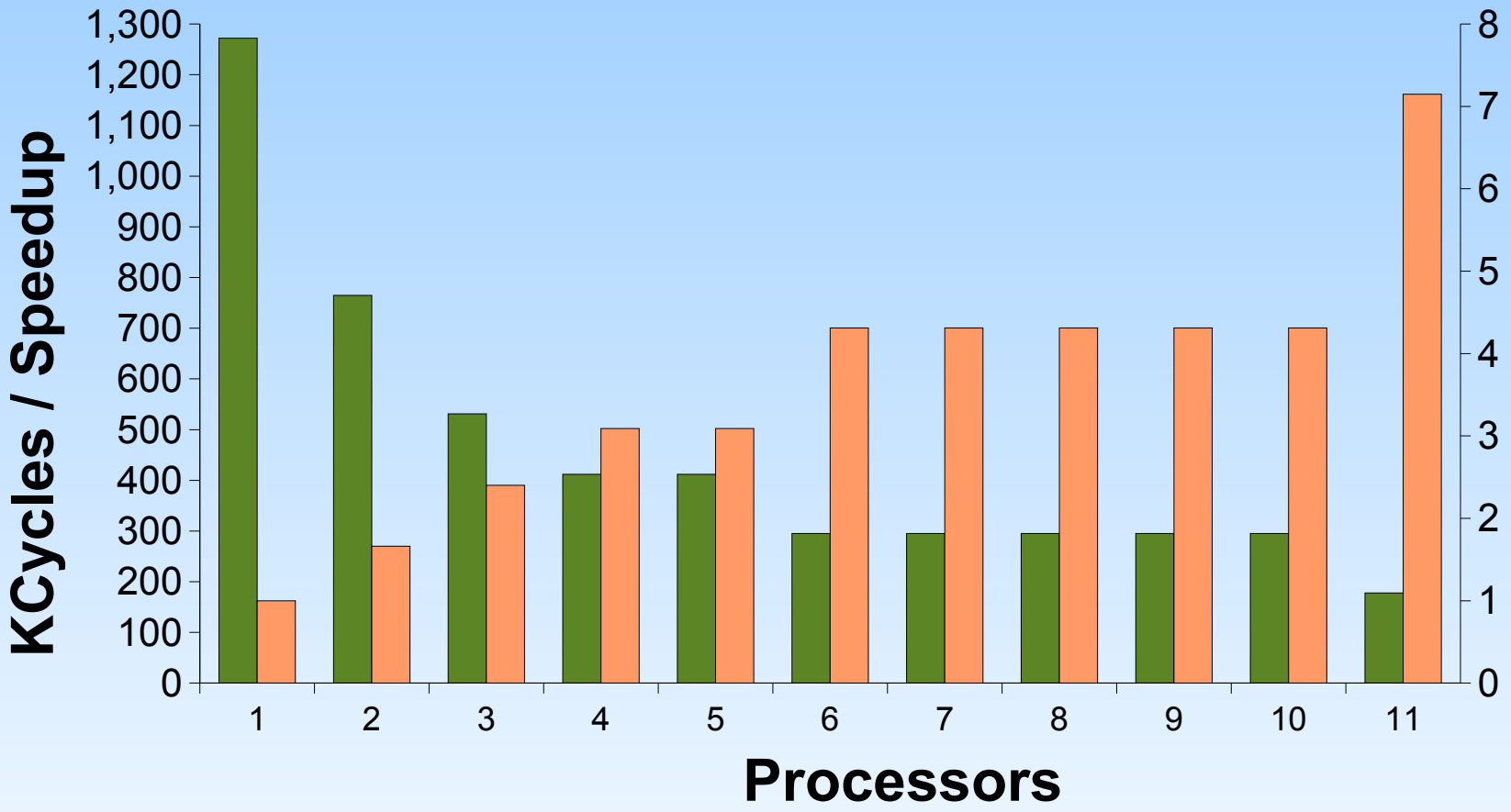




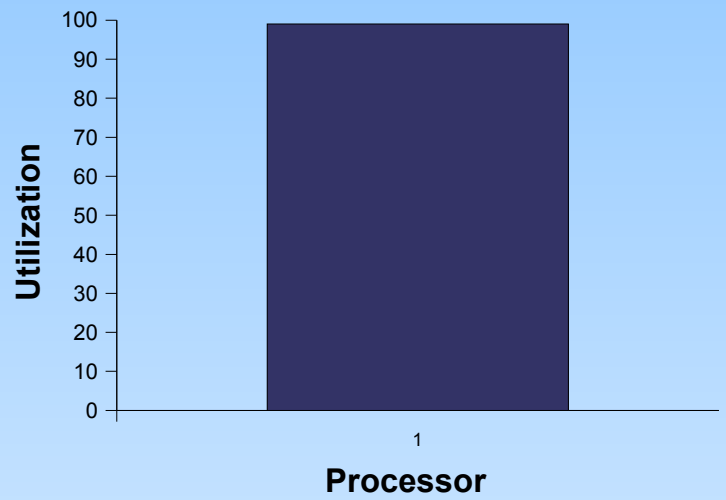
AES CMP Characteristics

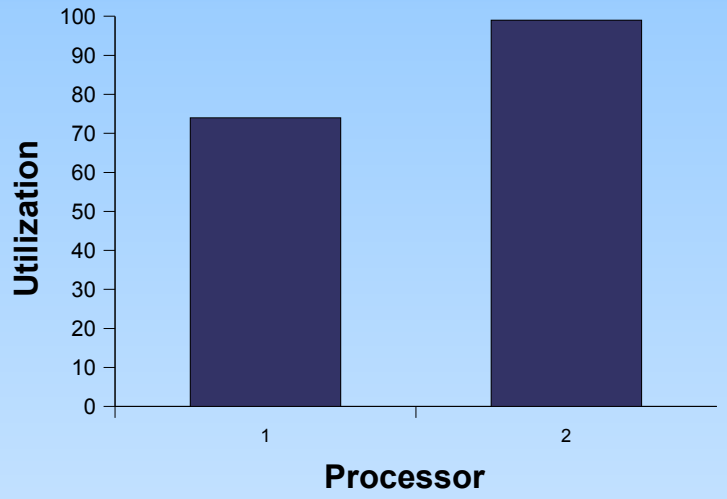
- Fully parameterized on number of processors
  - Runs on 1 to 11 nodes
  - Slightly modified from standard AES distribution
- Relatively large granularity
  - Approximately one node per round
  - Possible uneven work distribution
  - Effects utilization, but not power / energy
- Sub-round parallelism exploitation possible

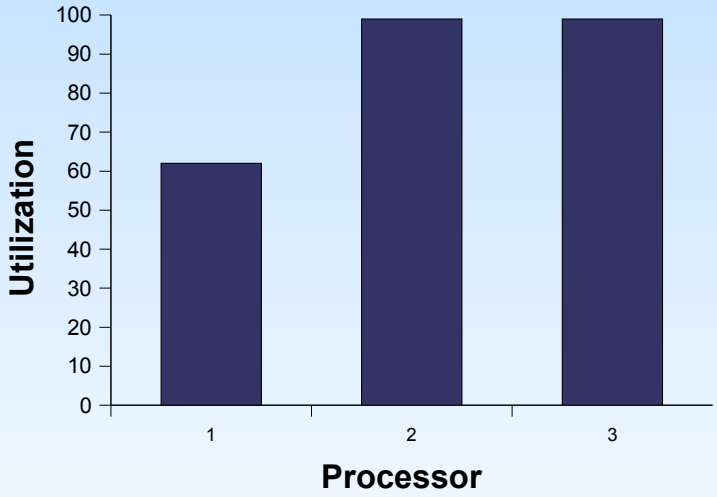
AES CMP Performance / Speedup

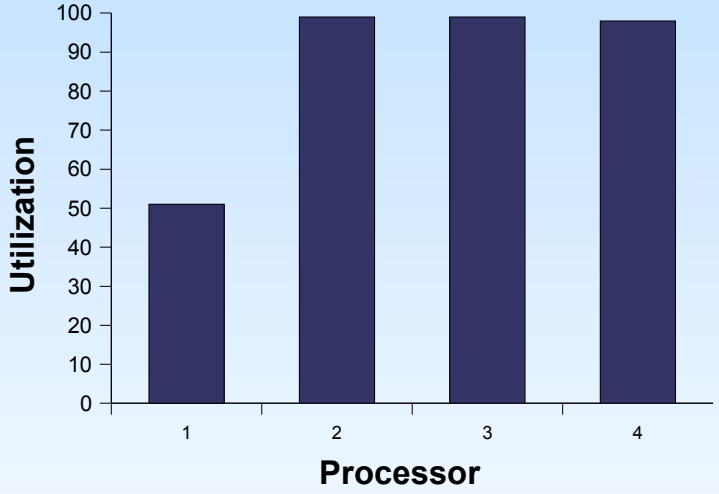


AES CMP Utilization Details

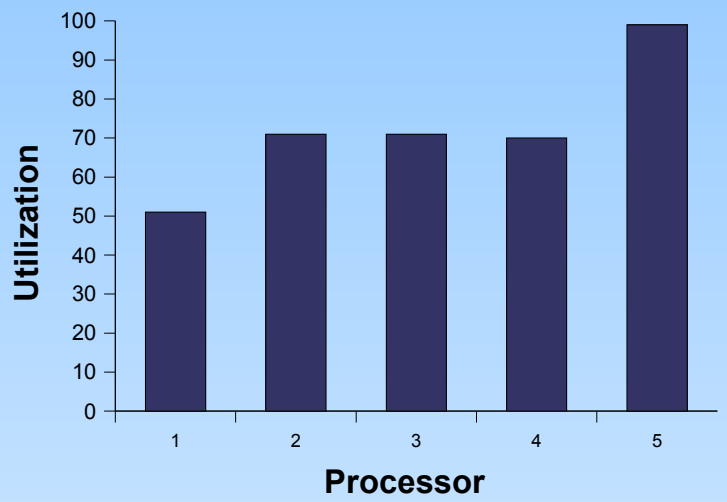


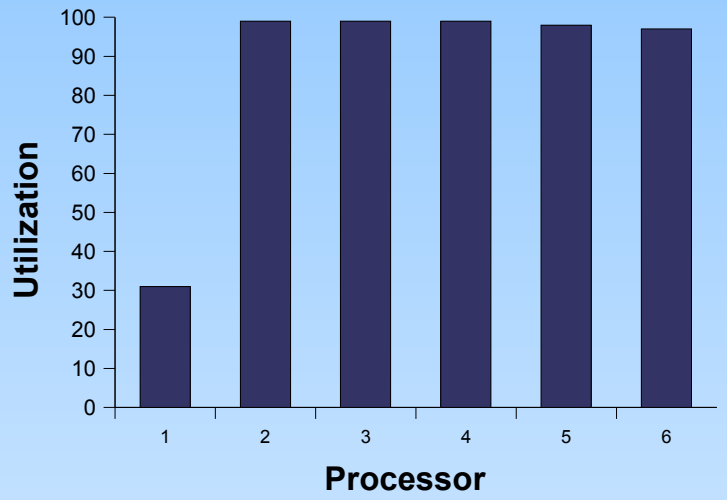


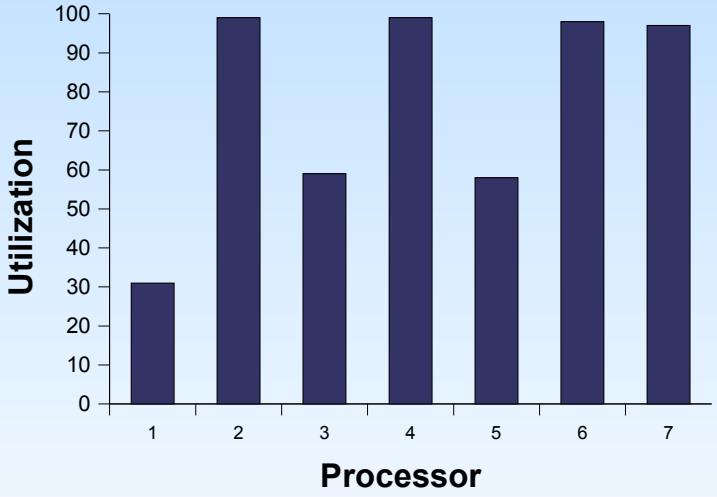


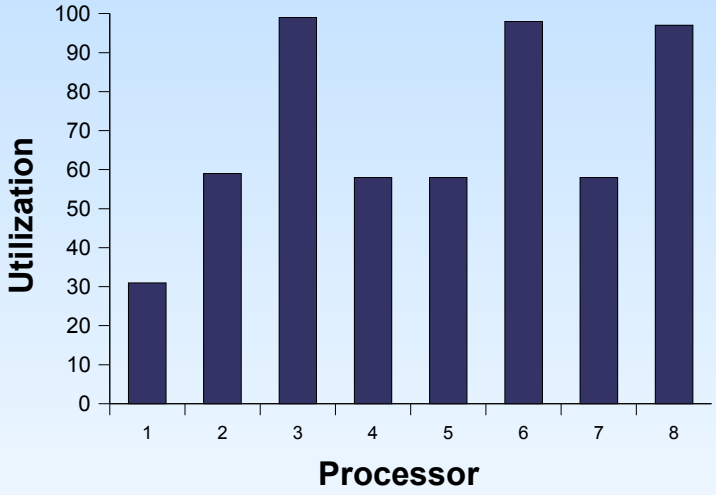


AES CMP Utilization Details (cont.)

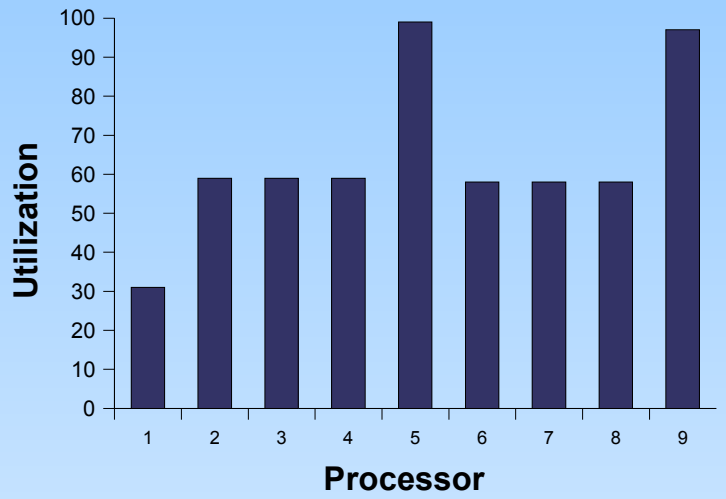


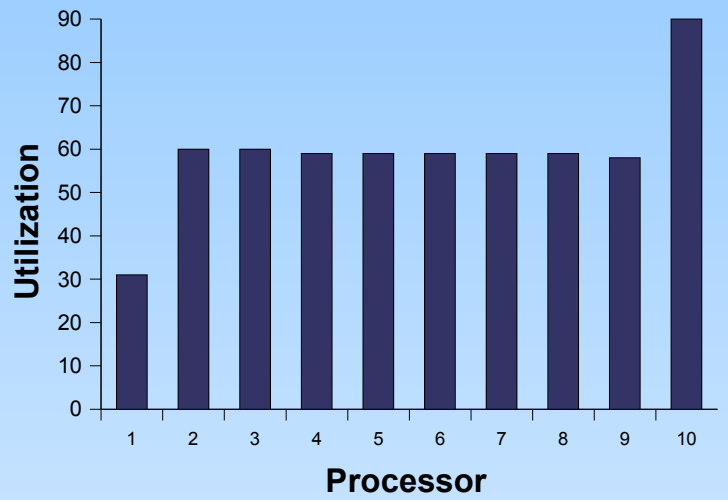


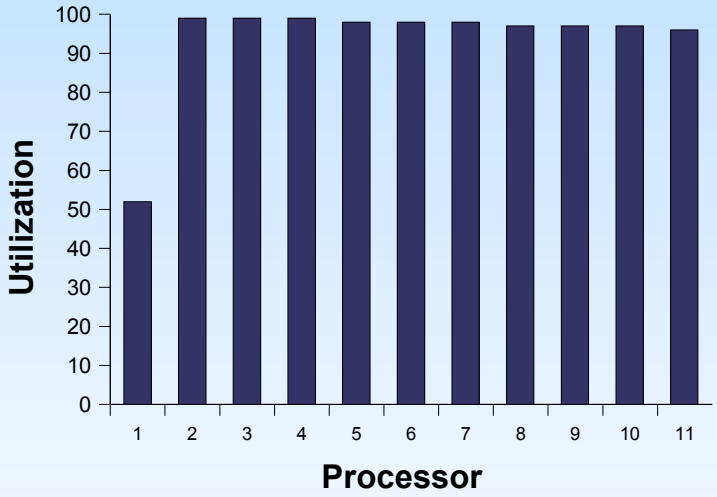




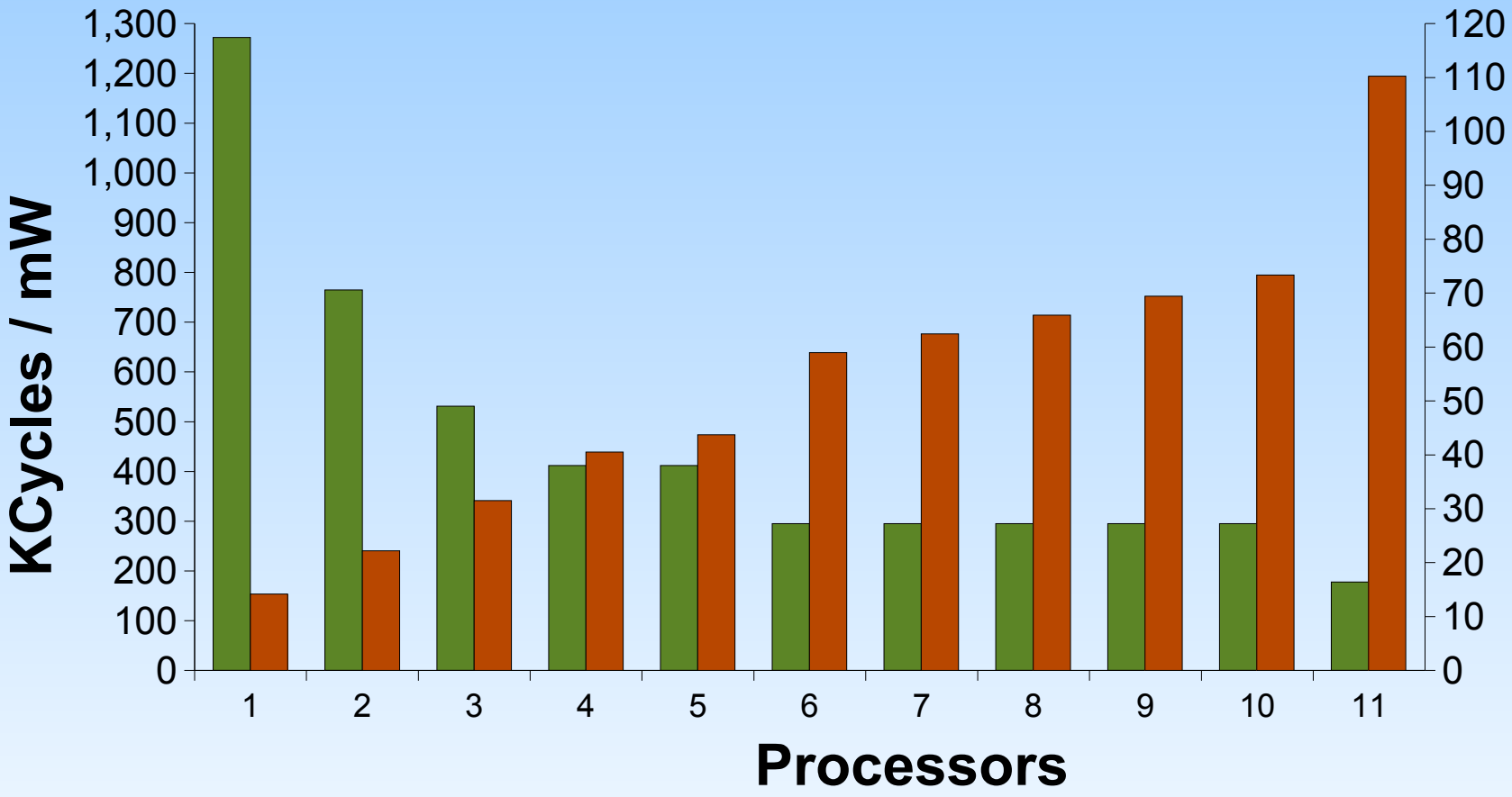
AES CMP Utilization Details (cont.)

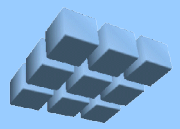






AES CMP Performance / Power

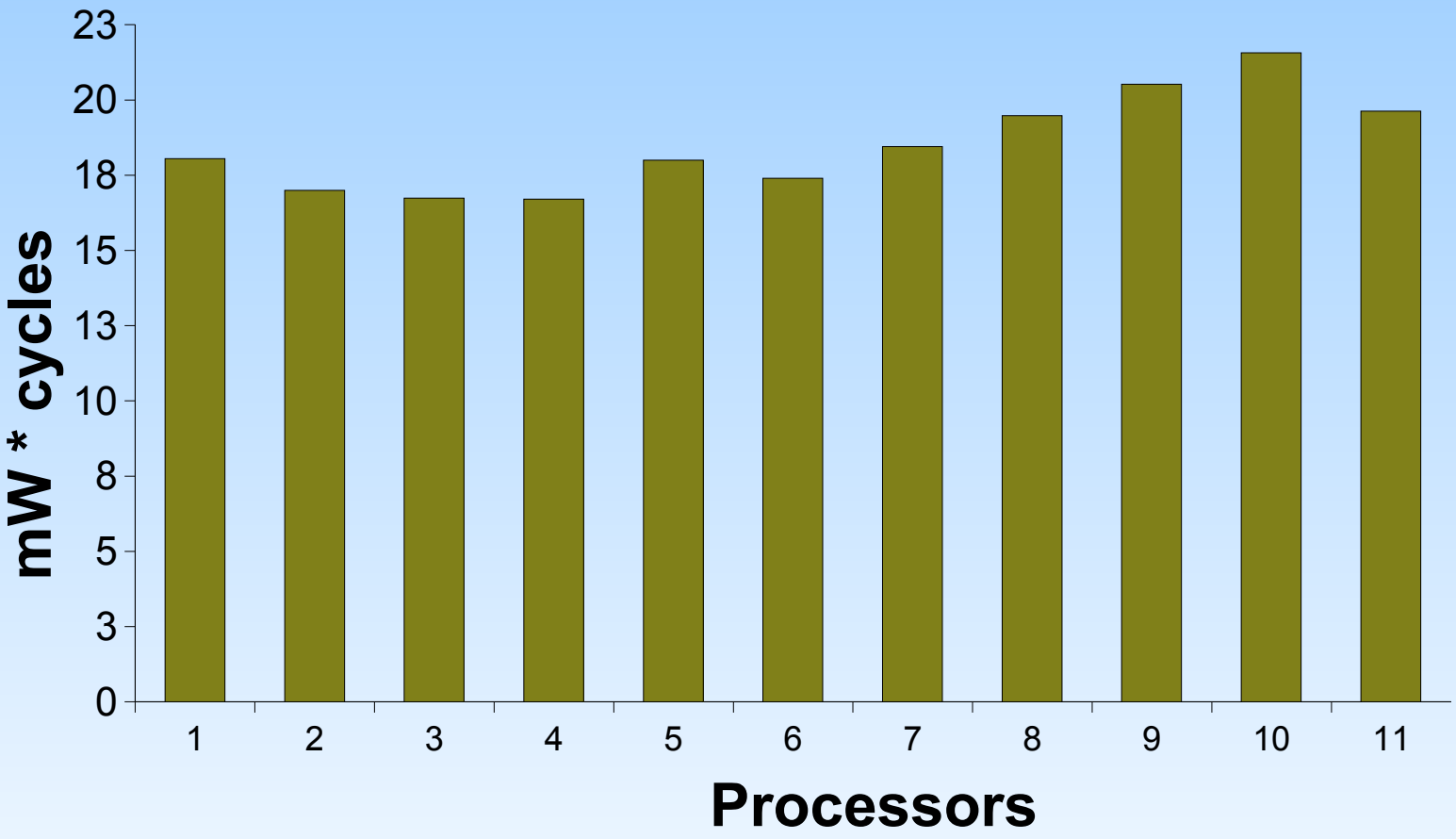


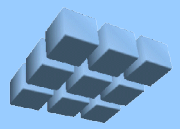


Energy = Power x Time

- Total energy consumed == battery life
- Total energy constant (+/- 10%)
  - More processors ==> faster result
  - Fewer processors ==> slower result
  - Little penalty for idle processors
- Provides a power / performance continuum
  - Can trade performance for power
  - Useful in hand held / portable environments

AES CMP (Power x Time)



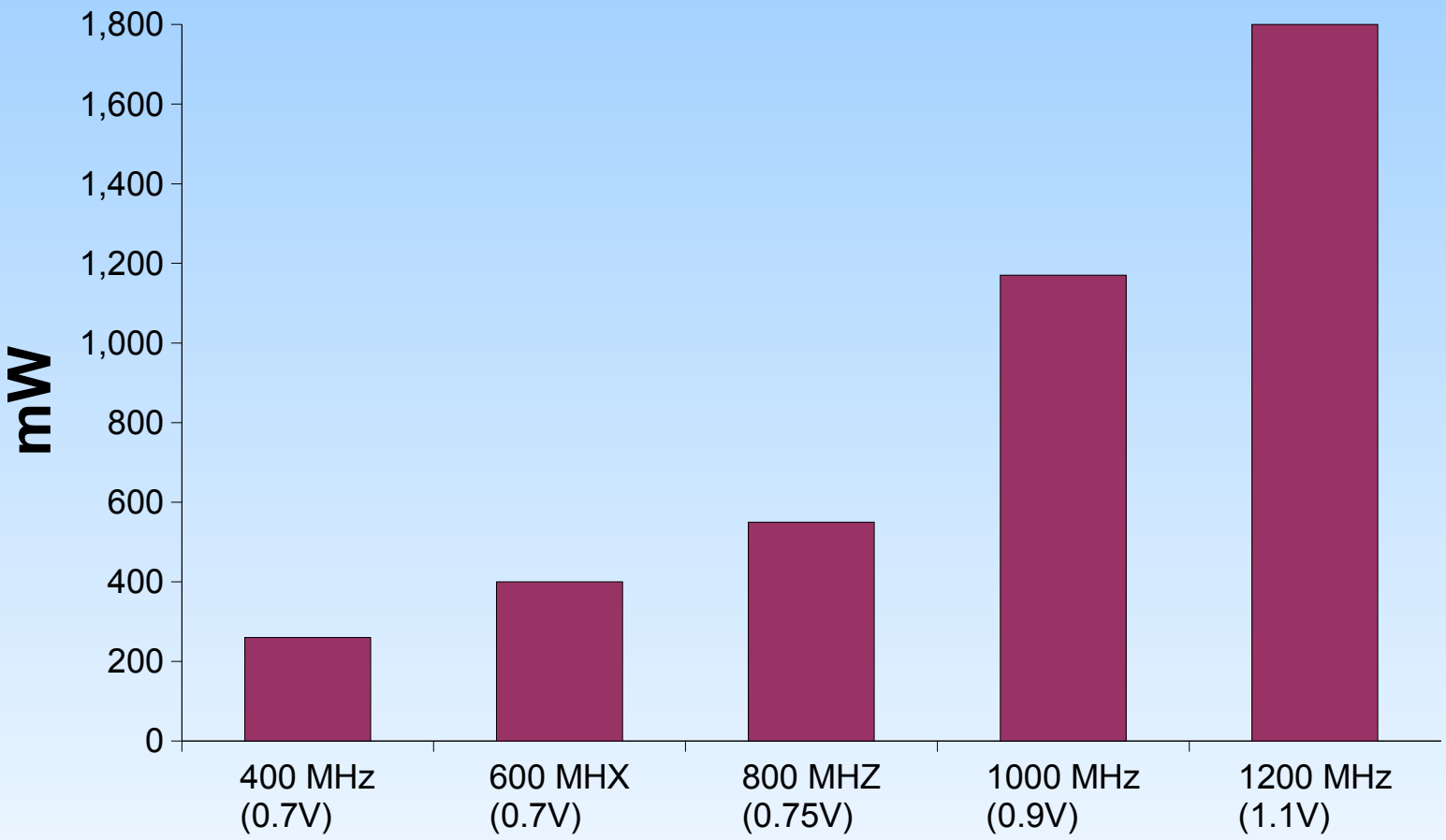


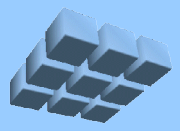
Power and Voltage Scaling

- Multiprocessors scales power / performance
  - Two processors == 2x performance
  - Two processors == 2x power == ½ clock speed
- Lower clock speed ==> lower voltage
- Power proportional to voltage squared (v^2)
- ARM1020E (Samsung “Halla”, 0.13 um)
  - 400 MHz / 0.7 V ==> 260 mW
  - 1200 Mhz / 1.1V ==> 1800 mW

  ==> *3x performance = approx. 7x power*

Power and Voltage Scaling

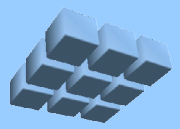




Add Processors, Save Power

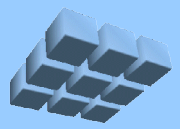
- Run hardware at lower voltage / clock speed
- Multiple processors meet performance goals
  - 1 x 1200 Mhz = ***1800 mW***
  - 3 x 400 Mhz = 3 x 260 mW = ***780 mW***
- One vs. Three ARM1020E
  - 1 processor ==> 3 processors
  - Identical raw performance (1200 MIPS)
  - **57%** total power savings

  ==> *Add processors and save power (!)*



AES CMP Results

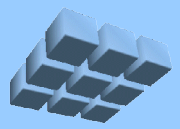
- Easily exploits round-level parallelism
- Linear performance / power tradeoffs
- Handles irregular / large grained computation
  - Utilization may suffer
  - Power / performance won't
- Uses existing software
  - Data read from IO ports, not memory



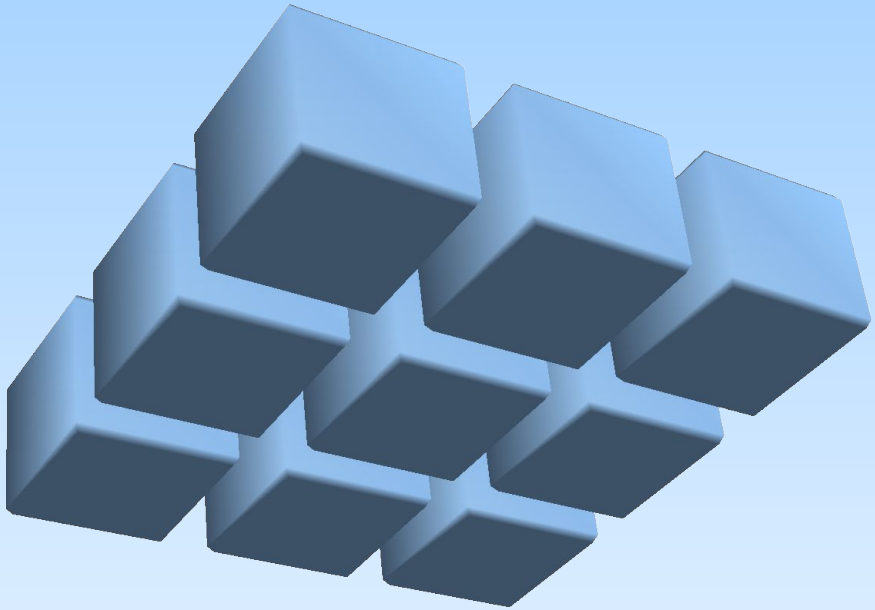
Conclusions

- CMP being used in ASICs, CPUs and FPGAs
- Solves problems in system design
  - Complexity, verification, power, flexibility, performance, programmability, etc.
- Delivers on the (largely unfulfilled) promises of *Reconfigurable Computing*
- Software / tools challenges still exist

... but progress is being made



Extra Slides



AES CMP Utilization