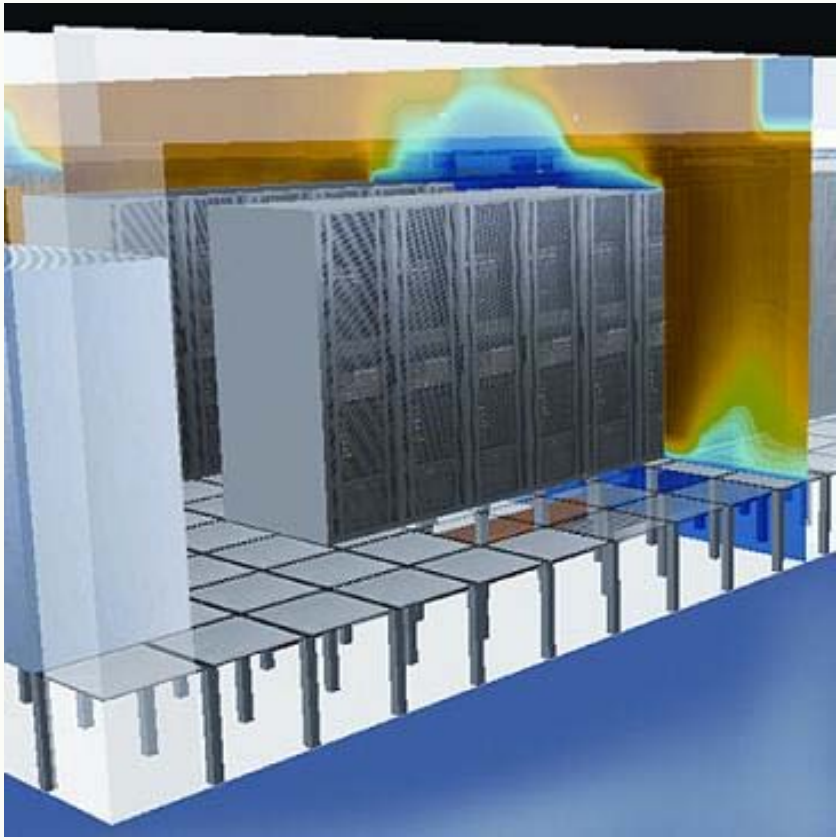
Hybrid-core Architectures for Energy-Efficient Computing

THE WORLD'S FIRST HYBRID-CORE COMPUTER.

ENA-HPC Conference, Hamburg, 16-17.9.2010

Agenda

- Energy challenges – the problem at hand
- Hybrid-core computing
- Examples



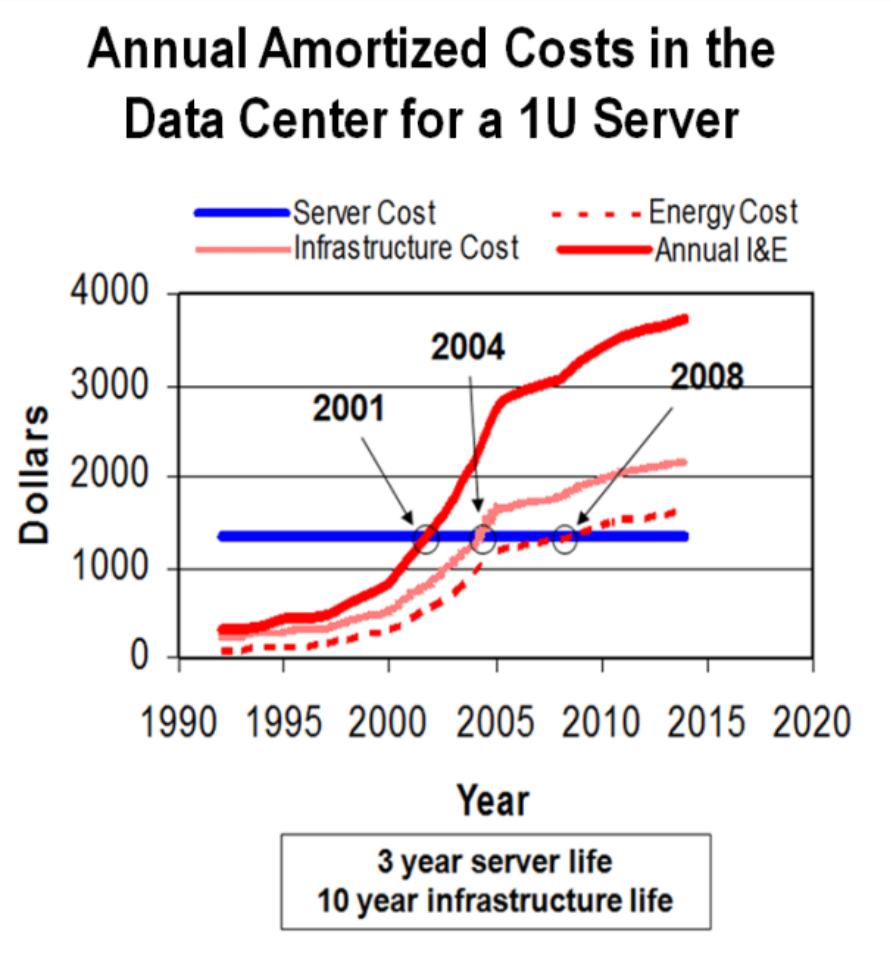
Courtesy Mentor Graphics

Energy Challenges

- **Electrical power constraints:**
  - Power into a data center has a limit
  - Energy costs are consuming ~half the operating budget
- **Cooling constraints:**
  - For every watt to power a server, up to another watt is required to cool the system
  - Cooling equipment often takes as much space as the computers themselves
- **Space constraints:**
  - Floor space becomes a hard limiter as racks of systems and the necessary cooling equipment proliferate
  - A modern data center can cost $200M

Data Center Economics

- **Energy constraints:**
  - Power to a data center has a limit
  - Energy costs are consuming half the IT operating budget
- **Cooling constraints:**
  - For every watt to power a server, another watt required for cooling
  - High capital costs
- **Space constraints:**
  - Floor space has become a limiter
  - Modern data center cost $200M



Belady, C., "In the Data Center, Power and Cooling Costs More than IT Equipment it Supports", Electronics Cooling Magazine (Feb 2007)

CONVEY computer™

Exascale Computing using Today's Technology

- **Highest-ranked system on the "Green500™ List" is 773 MFLOPS/Watt***
  - "Hybrid" x86/IBM Cell system: Forschungszentrum Juelich (FZJ) QPACE SFB TR Cluster, PowerXCell 8i, 3.2 GHz, 3D-Torus
  - Exascale system would be ~1.3×10^9 W (=1,29 Gigawatts or about 1/12 the residential electricity consumption of Germany)**
- **A "pure" x86 system is ~135 MFLOPS/Watt**
  - Exascale system would be ~7.4×10^9 W (=7,4 Gigawatts or almost 1/2 the residential electricity consumption of Germany!)

Processor Family	Count	Share	Total Power (kW)	Average MFLOPS/W	Average Green500 ranking
AMD x86_64	49	9.8%	53,420	138.96	250
Intel EM64T	401	80.2%	178,028	131.10	256
Intel IA-64	5	1%	5,433	54.61	398
NEC	1	0.2%	2,400	51.00	423
Power	42	8.4%	30,124	243.94	172
Sparc	2	0.4%	1,284	108.31	262

*http://www.green500.org/

**International Energy Agency; http://www.iea.org/stats/electricitydata.asp?COUNTRY_CODE=DE

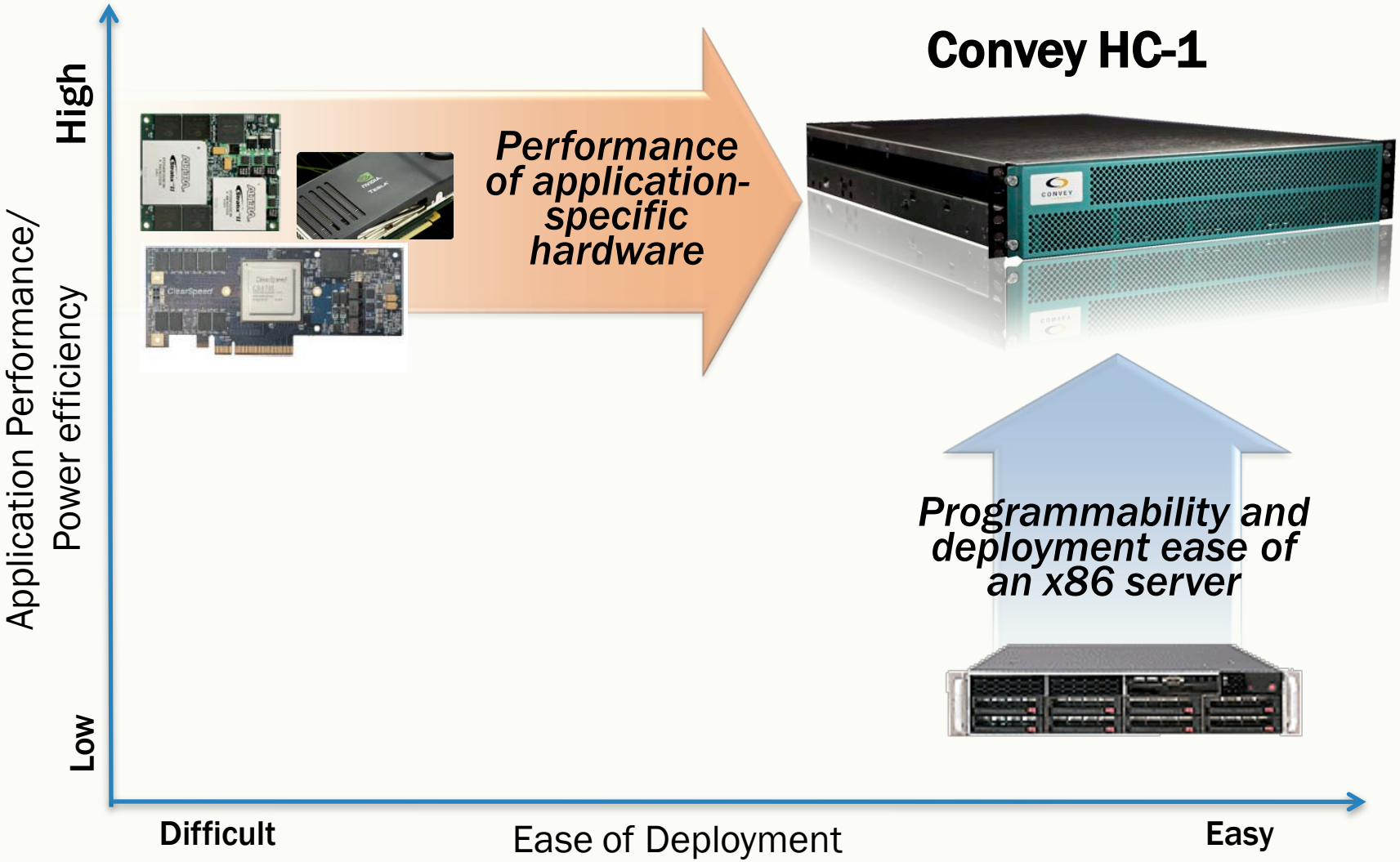
Observations

- **We must reduce facilities costs**
  - It now costs as much to power and cool a server for a year as its purchase price
  - Data center floorspace is limited
- **We cannot get to exascale with current technology**
  - Major processor vendors are doing the best they can
    - Lower clock rates, more cores/socket
    - Elegant and exotic architecture features to minimize power
  - But watts/m^2 still high
  - And space is still a problem
- **Very difficult to leave the x86 development/runtime ecosystem**
  - Billions of dollars in existing applications
  - Developers don't want to learn new languages/extensions/dialects
  - Mainstream massively parallel programming is a l-o-n-g way off
- **Ideal solution provides both**
  - Orders of magnitude better operations/watt
  - Minimizes impact to application developers

CONVEY computer™

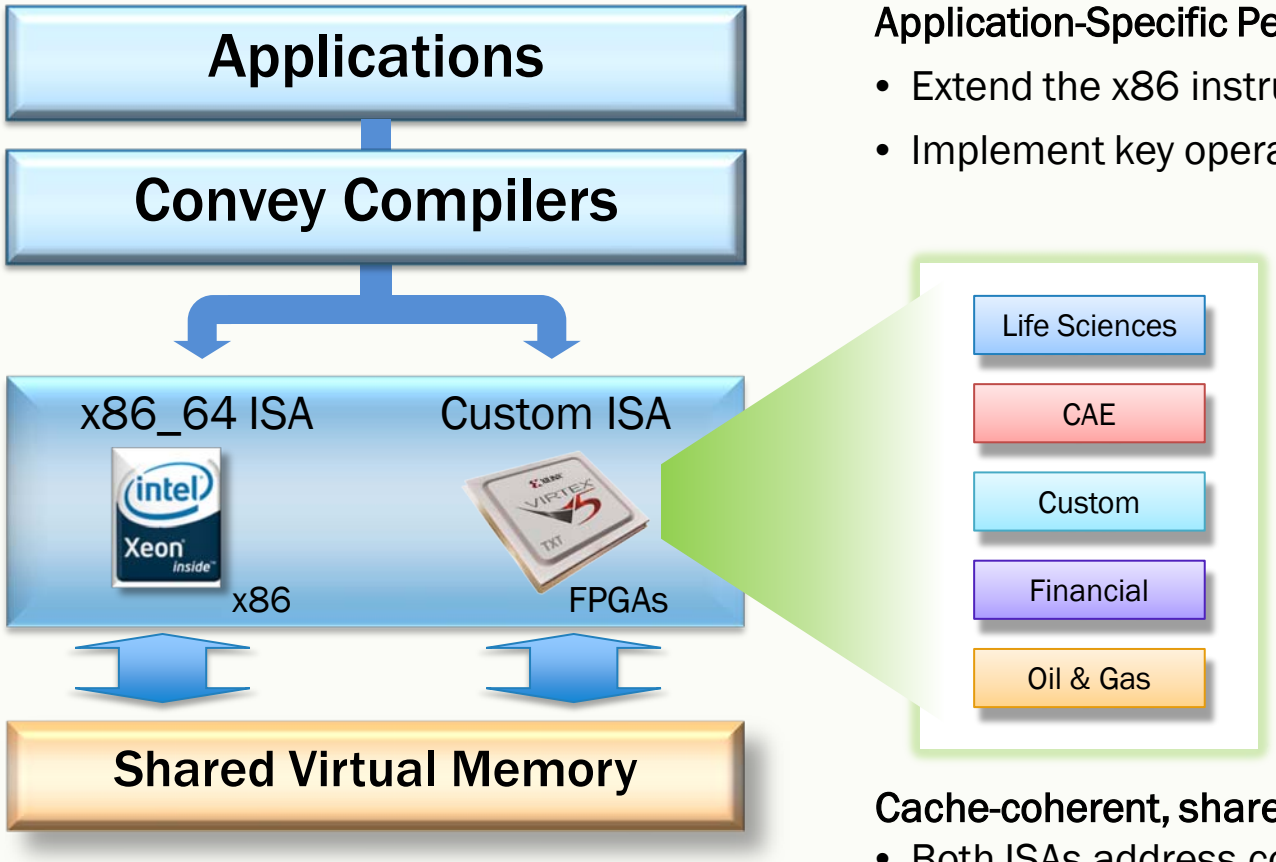
HYBRID-CORE COMPUTING

Hybrid-Core Computing



CONVEY computer™

Hybrid-Core Computing



Application-Specific Personalities

- Extend the x86 instruction set
- Implement key operations in FPGAs

Cache-coherent, shared memory

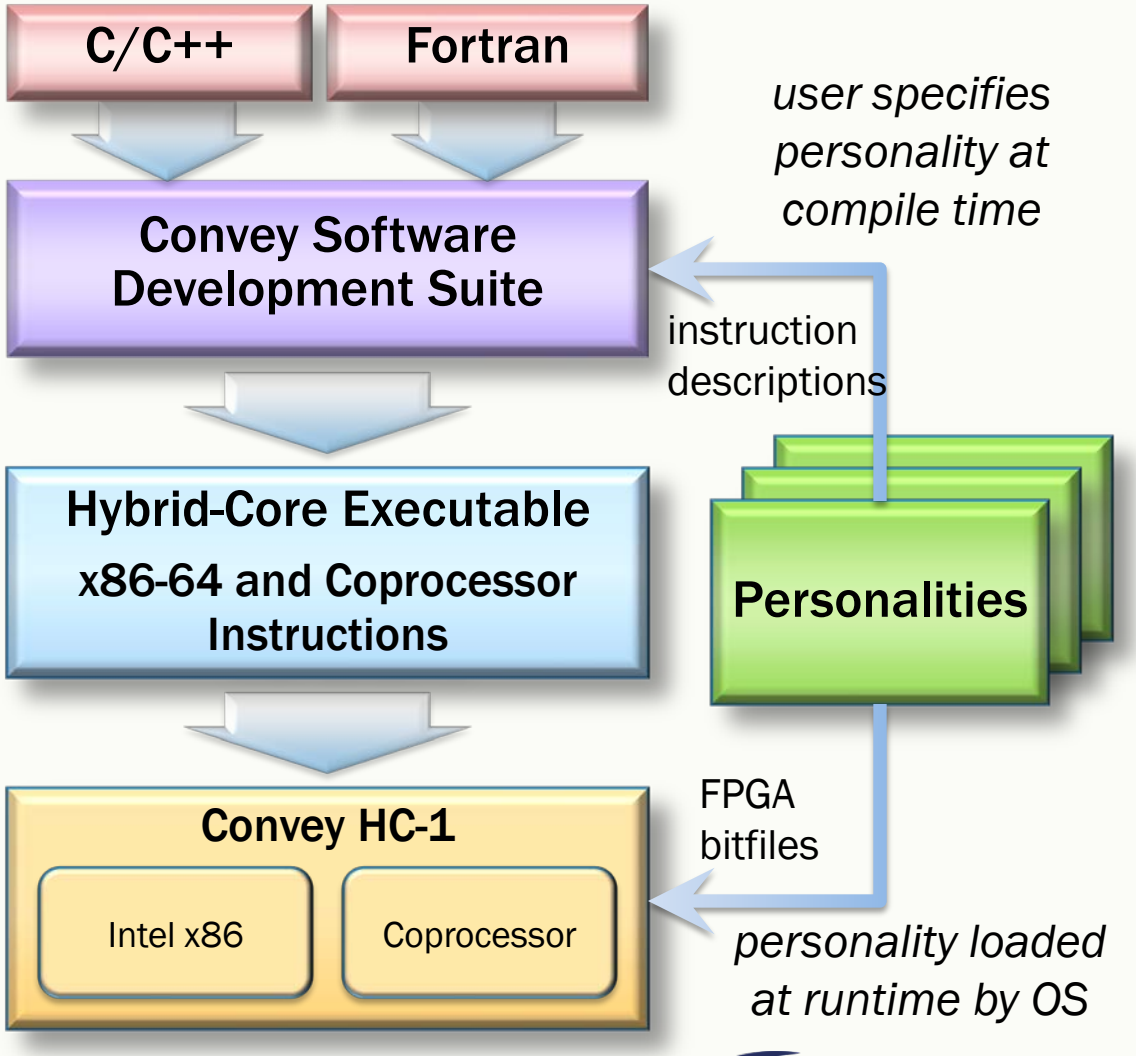
- Both ISAs address common memory

*ISA: Instruction Set Architecture

CONVEY computer™

Using Personalities

- Personalities are reloadable instruction sets
- Compiler generates x86 and coprocessor instructions from ANSI standard C/C++ & Fortran
- Executable can run on x86 nodes or Convey Hybrid-Core nodes



CONVEY computer™

Hybrid-Core Computing

- **General purpose processors solve problems using streams of fixed-functionality instructions and fixed instruction sets**
  - Frequently an algorithm may not map efficiently to the instructions provided by a processor
  - In these cases, the problem being solved takes a significant number of instructions to solve the problem
- **An FPGA uses programmable logic resources with flexible-functionality instructions and flexible instruction sets (personalities)**
  - The programmable logic is defined in a way to maximize the efficiency of the electronics (gates) involved to solve the problem

Case study:
Logical operations on 32-bit data types

- Real-world application example (albeit a special case)
- The following 'C' code performs a generic 4-input logical operation on 32-bit operands.

```
uint32 Log4(uint32 F, uint32 A, uint32 B,
            uint32 C, uint32 D) {
    uint32 R = 0;
    for (int i = 0; i < 32; i += 1) {
        uint32 a = (A >> i) & 1;
        uint32 b = (B >> i) & 1;
        uint32 c = (C >> i) & 1;
        uint32 d = (D >> i) & 1;
        uint32 e = (a << 3) | (b << 2)
                 | (c << 1) | d;
        R |= ((F >> e) & 1) << i;
    }
    return R;
}
```

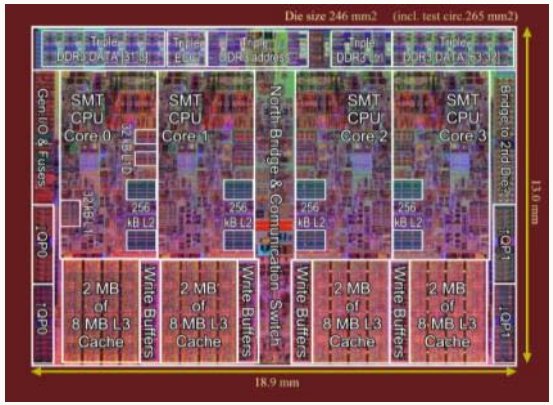
"C" Code of 4-input logical operation

```
00401006  xor   edx,edx
00401008  mov   ecx,esi
0040100A  shr   edx,cl
0040100C  and   edx,1
0040100F  lea   edi,[edx+edx]
00401012  or    edi,edx
00401014  add   edi,edi
00401016  or    edi,edx
00401018  mov   edx,1
0040101D  shr   edx,cl
0040101F  add   edi,edi
00401021  and   edx,1
00401024  or    edi,edx
00401026  mov   ecx,edi
00401028  mov   edx,5
0040102D  shr   edx,cl
0040102F  mov   ecx,esi
00401031  add   esi,1
00401034  and   edx,1
00401037  shl   edx,cl
00401039  or    eax,edx
0040103B  cmp   esi,20h
0040103E  jl    Log4+6 (401006h)
00401040  pop   edi
00401041  pop   esi
```

Assembly instructions (without loop logic)

General CPU Approach

- 23 x86_64 assembly instructions are executed per bit of the result
- Result is 32-bits wide, requiring (23x32) = 736 instructions to calculate the result



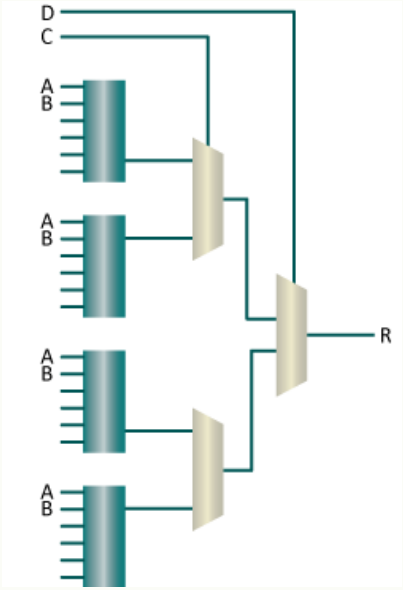
- State of the art processor runs ~3 Ghz. With one instruction performed per clock:
  - 736 instructions executes in 245 ns
- State of the art processor has 4 cores per die. One of these cores consumed for 245 ns performing this operation
- A processor consumes around 100 Watts of power when all four cores are active. One core would consume 25 Watts for 245 ns performing this operation
  - This is 6.1 x 10^{-9} Joules

Hybrid-core Approach

- An FPGA implements an algorithm using a set of logic elements
- The C code would be translated into the following hardware resources:

```
uint32 Log4(uint32 F, uint32 A, uint32 B,
            uint32 C, uint32 D) {
    uint32 R = 0;
    for (int i = 0; i < 32; i += 1) {
        uint32 a = (A >> i) & 1;
        uint32 b = (B >> i) & 1;
        uint32 c = (C >> i) & 1;
        uint32 d = (D >> i) & 1;
        uint32 e = (a << 3) | (b << 2)
                 | (c << 1) | d;
        R |= ((F >> e) & 1) << i;
    }
    return R;
}
```

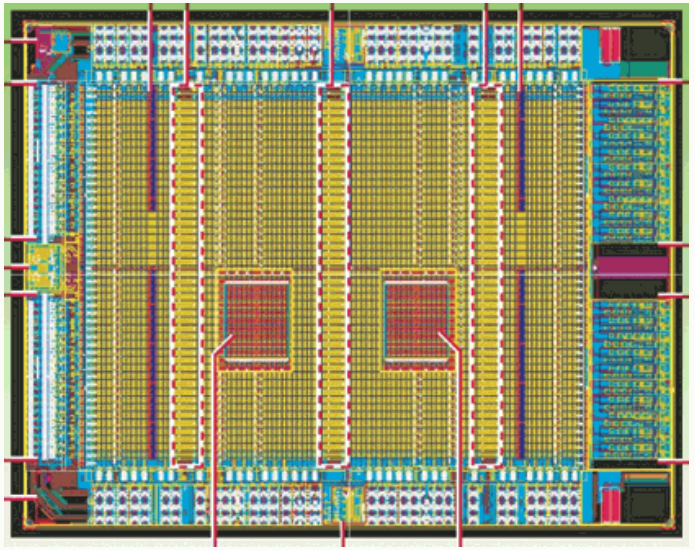
"C" Code of 4-input logical operation



FPGA Logic of 4-input logical operation

CONVEY computer™

FPGA Implementation

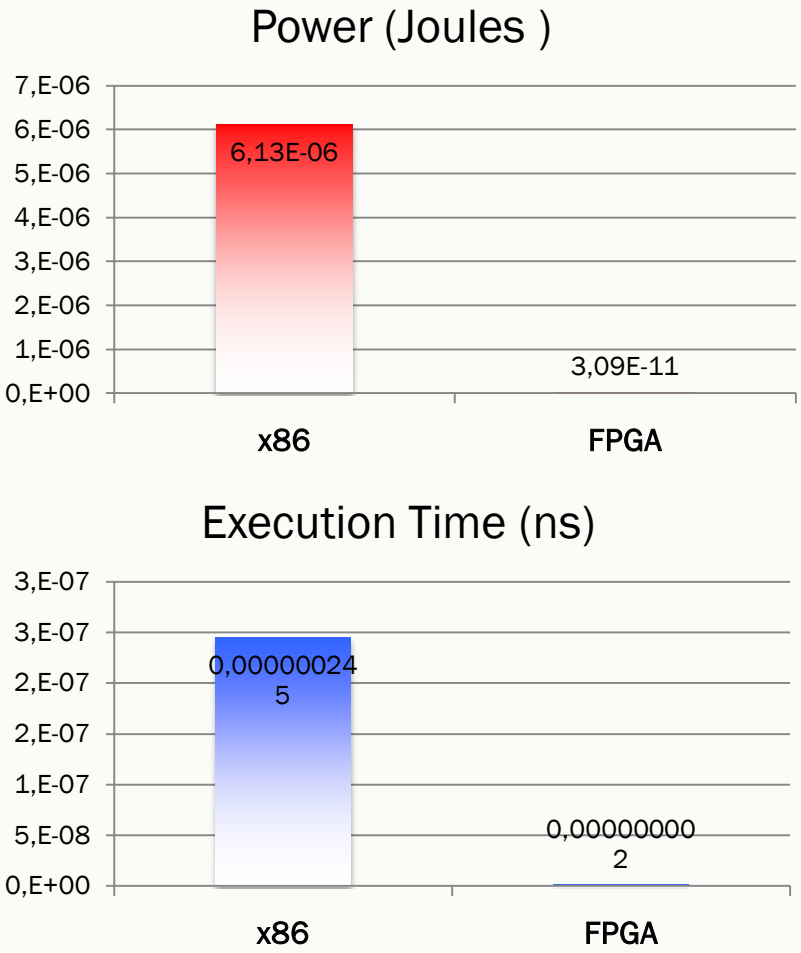


- FPGA resources for the C routine consume 4 logic tables per bit of result. For 32 result bits, FPGA uses 128 logic tables
- Xilinx 330LX FPGA consumes ~25 Watts and has 207,000 logic tables
- The FPGA resources to produce the 32-bit result consume 25 W * 128 / 207,000 or 0.0155 Watts. The FPGA logic table solution would take ~2ns to produce the result. This is $3.1x10^{-11}$ Joules

	Chip (W)	Proc time (ns)	Energy to Solution
x86	25	0,000000245	6,13E-06
FPGA	0,0155	0,000000002	3,1E-11
Ratio		122,5	198105,47

Energy/performance comparison

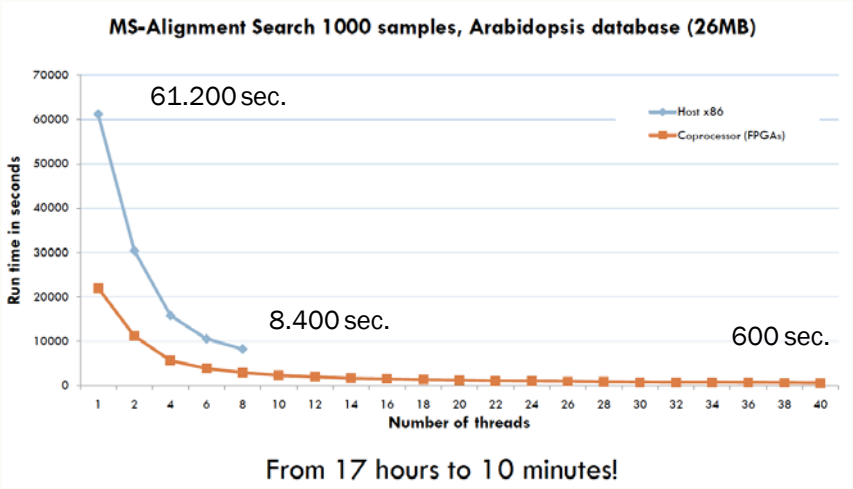
- The FPGA implementation is ~200,000 times more power efficient
- (By the way, it's also 122 times faster)



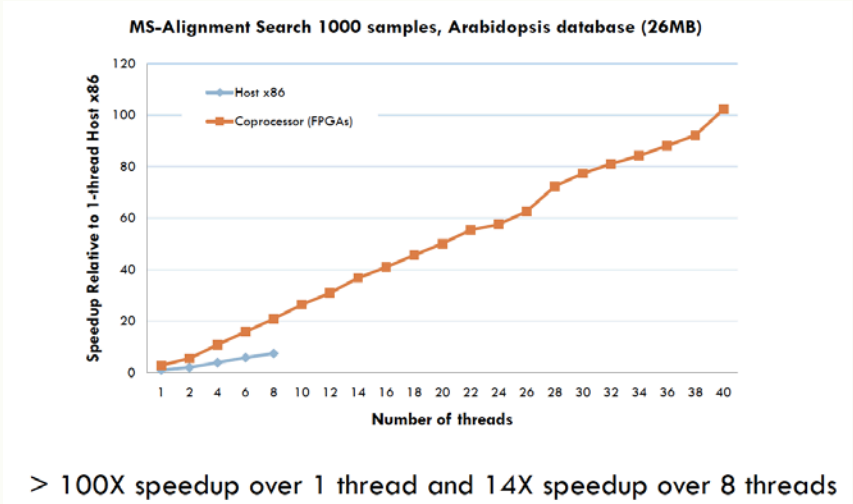
CONVEY computer™

Application Example
UCSD InsPect

- **Bioinformatics personality for protein sequencing**
  - University of California San Diego InsPect Application
  - Core of application implemented as hybrid-core "personality"
- **HC-1 produces results 14x faster than an equivalent 8-core x86 based system**



From 17 hours to 10 minutes!



> 100X speedup over 1 thread and 14X speedup over 8 threads

CONVEY computer™

UCSD InsPect

Energy comparison for equivalent performance

1 HC-1 > 14 x 2-socket 3GHz x86 servers

1 Rack (16 nodes)	Convey =	91 MW-h/yr
8 racks (224 nodes)	x86 =	880 MW-h/yr

1 Year Electricity costs[1] (@ $0.25/kWh)

Convey	$45.6 K/yr
X86	$441.5 K/yr

1 Year Infrastructure costs[2]

Convey	$15.2 K/yr
X86	$147.2 K/yr

3-Year TCO[3]

Convey	$619 K
X86	$3,426 K

224 x 1U 2-socket servers

16 x 2U Convey HC-1

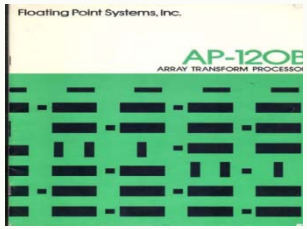
Reduction in space	86%
Reduction in datacenter watts	90%
Reduction in 3 yr TCO	73%
Reduction in 3 yr TCO with Rebate	81%

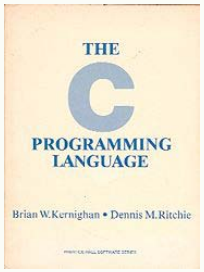
[1]Includes datacenter power/cooling costs (2x); excludes any "Green" rebates
[2]Includes prorated 10-year UPS & datacenter floorspace
[3]Includes purchase, h/w maintenance, power, infrastructure, and "Green" rebate

CONVEY computer™

Observations:

- **Heterogeneous computing is inevitable**
  - More performance using less power (more efficient use of transistors)
  - Application-specific logic is the most efficient
- **Successful performance enhancements are tightly integrated with the processor**
  - integrated vector processors vs. array processors
  - common address space & data types
- **Single compiler & programming environment**
  - industry standard source (no new languages or dialects)
  - leverage of existing applications and algorithms
- **Systems that are simpler to program win**





CONVEY computer™

Summary

- Challenge to reduce power without sacrificing performance
- Continuing flattening processor clock rates
- Challenge to fully utilize multi-core implementations (Software?)
- FPGA technology is NOT subject to Moore's Law today
- Reducing power and floor space can also mean more performance
- Or: