



Heterogeneity

The natural consequence of *free* transistors

Ben Bennett, GM
ClearSpeed Technology plc
PPAM, Gdansk 11th September 2007



Green is “the new black”

- **Design Automation Conference, San Diego, July 2007**
 - “...power is perhaps the most important criterion in multi-core. But it will be very hard to get it right.” - John Darringer, lead system researcher, IBM.
 - “...with over a thousand cores on-chip, the interconnect alone could account for 150W.” - Anant Agarwal, CTO, Tiler.
 - “...power will continue to haunt us in the next decade.” - Shekar Borkar, microprocessor research director, Intel.
- **‘Beyond Linpack’ – A new class of ‘Power-Efficient Systems’**
 - Dr. Strohmaier of Lawrence Berkeley National Laboratory at International Super-Computing Conference, Dresden June 2007

Constraints For Processor Development

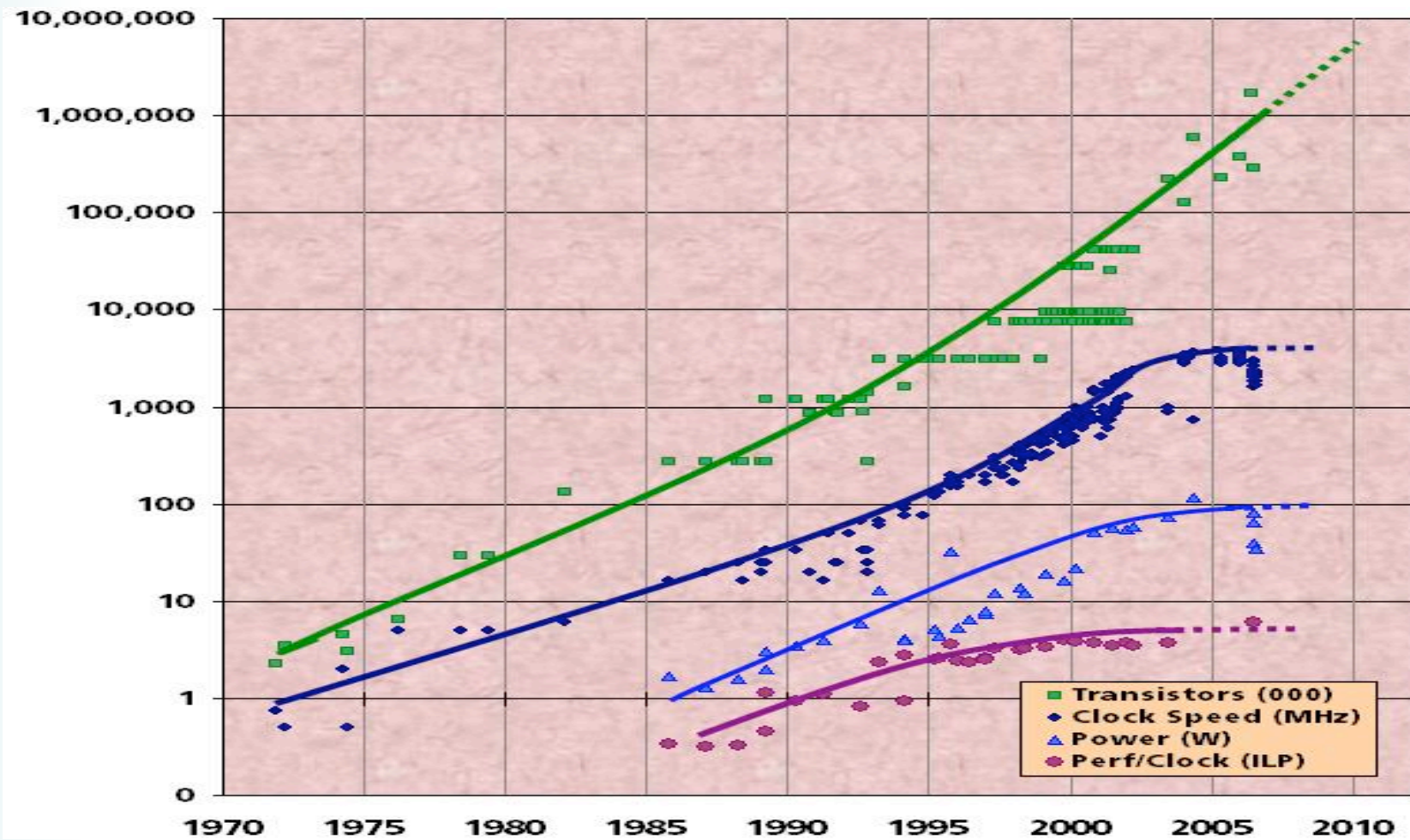


Figure courtesy of Kunle Olukotun, Lance Hammond, Herb Sutter, and Burton Smith

The Move to Acceleration & Heterogeneous Systems

- **Multi-core CPU's break dependence on clock speed and slow system's power consumption trend, but cannot solve it**
- **Accelerators can compliment CPU's to address costs of ownership and infrastructure**
- **Intel, IBM, and AMD driving interface standards to efficiently connect accelerators within system**
 - Not just the hardware
- **Heterogeneous systems will have CPU's and Accelerators:**
 - Exploit the excellent economics of standard servers
 - Address their power consumption and infrastructure impact by having special performance accelerators
- **Accelerators must therefore:**
 - Enhance current paradigm of standard server adoption
 - Address issues of power, cooling, space, weight
 - Apply across multiple vertical markets

Requirements of Mainstream Accelerators

- **Enhance Standard Servers:**
 - Significantly more performance per watt efficiency than standard CPU's
 - X86 compatible
 - A question of economics not processor religion
 - Additive not alternate to CPU performance
 - Exploit expansion slots or daughter card formats
 - Economies of scale doom socket compatibility accelerators
 - Meet expansion slot or daughter card power budgets
- **Apply Across Multiple Vertical Markets**
 - 64 Bit Floating Point - the standard of accuracy for HPC
 - Easily programmable and sustainable - unified source throughout software lifecycle
 - Truly exploit full data parallelism in application
 - Exploit standard library functions – minimise code to be ported
 - Linear Scalability

Performance & Power Efficiency

	Average Wattage	32-bit Peak GFLOPS	64-bit Peak GFLOPS
Intel Clovertown (3.6 GHz) - system	250	86	57
Nvidia Tesla	170	~500	0
Future Nvidia 64-bit	~200	~500	~62-125
ATI Radeon	~200	550	0
Cell BE	210	230	15
Future Cell HPC	220	~200	104
10 ClearSpeed Advance e620	250	~840	~840

Notes

- 1 Table uses information given by vendors at International Super Computer Conference, Dresden, June 2007
- 2 25 to 50 Watts is current expansion slot power budget, 250 Watts proposed

Looking Forward 2 – 5 Years

- **Heterogeneous systems will have moved from HPC to mainstream:**
 - Intel's Geneseo* and Larrabee*
 - AMD's Torrenza* and Fusion*
 - Multi-core CPU's will have caused much software development to increase exploitation of standard 'parallel' libraries
- **Performance per watt and ease of programming will dominate:**
 - Acceleration only makes economic sense if far more efficient than the standard CPU on performance per watt
 - Acceleration without ease of programming is no solution
- ***Accelerated systems will achieve 10PFLOPS by 2011***

* 3rd party names/brands are owned by the 3rd parties

Two years ago

In summary

- HPC is advanced indicator for the rest of the technical industry ✓
- More cores, more threads, more potential, more risk ✓
- Invest now in algorithms & software
 - Hardware is a commodity ✓
- Intel is NOT just the worlds most prolific supplier of HPC building blocks
 - Talk to us about software ✓

Talk to “us” about software

ClearSpeed's Software Development Kit (SDK)

- **Familiar development environment**
 - Industry-standard tools and languages
- **Optimizing Cⁿ compiler**
 - Cⁿ based on ANSI C
 - Standard C libraries (including parallelized versions)
- **Industry-standard GNU debugger, gdb**
 - Real time debugging on hardware
 - Extensions for CSX600 architecture
- **Visual profiling toolkit**
 - Profiles Advance cards, host cores and whole system
 - Fully supported in CSX600 hardware

Compilation

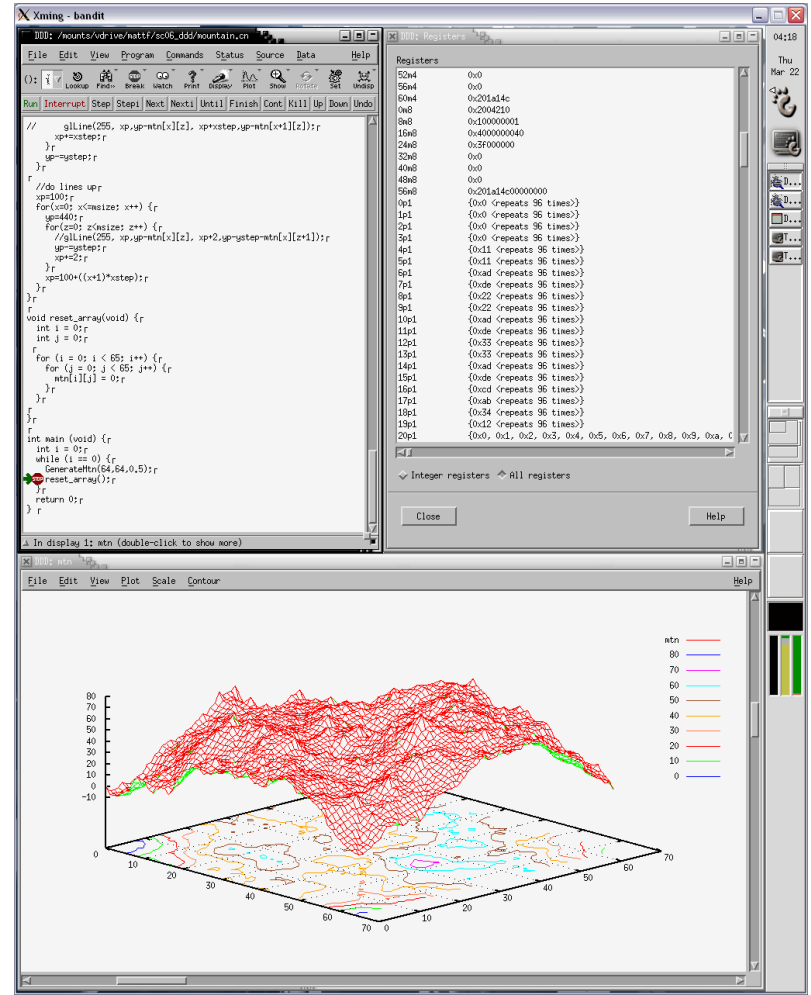
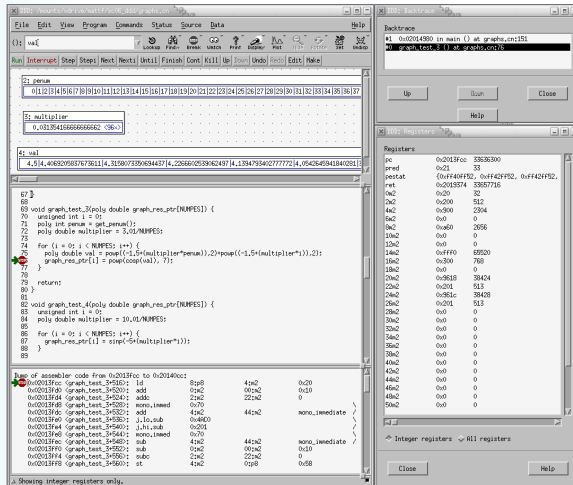
- **Easy to use**
 - Standard toolchain: preprocessor, compiler, assembler, linker, loader
- **Optimizing compiler for performance and rapid development**
- **Options for further tuning**
 - Vector math library
 - Inline functions
 - Vector data types and intrinsics
 - Pragmas for control of
 - Data alignment
 - Code placed in faster memory
 - Loop unrolling
- **Inline or macro assembler for ultimate performance**
 - Direct access to instruction set
 - Most instructions have mono and poly variants

Libraries

- **Optimized libraries for application developers**
 - For higher performance and complex math functions
- **Standard Cⁿ libraries**
 - Most of standard C library
 - Plus parallelized versions of most functions
 - Parallel random number generators (for Monte Carlo, etc)
 - Vector math library to exploit performance of FP pipeline
 - Extended library for architecture specific features (I/O, etc)
- **Application libraries**
 - FFT functions: 1D, 2D, real, complex
 - Card-side BLAS and LAPACK functions (in 3.x release)

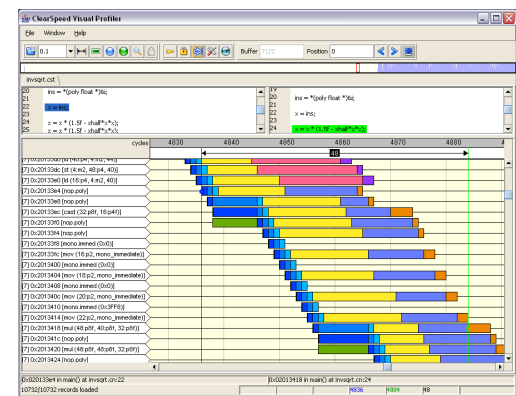
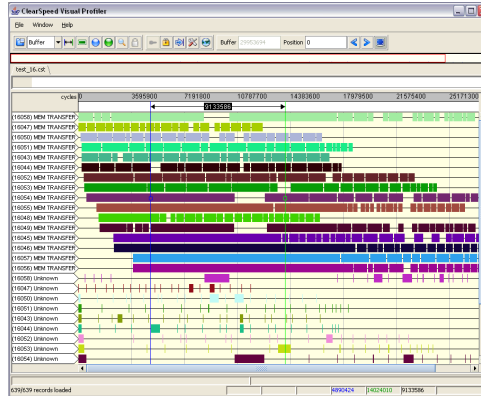
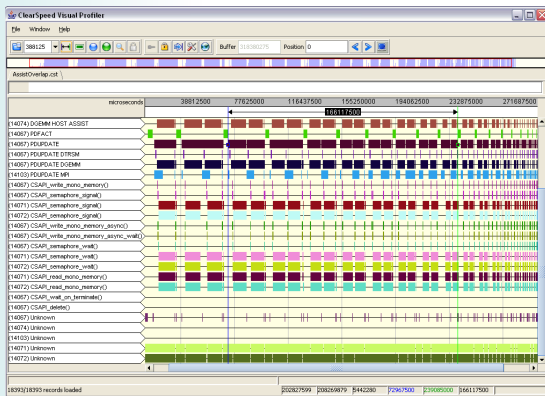
Source code debugging

- **Based on industry-standard GDB**
 - Real time debugging on hardware
 - Familiar tool
 - Cross platform
 - Range of graphical interfaces
- **Extensions for poly data type**
 - Supports debugging of CSX600 code
 - Visualize poly data across the array



ClearSpeed Visual Profiler

- **Profiles multiple x86 cores and Advance cards**
 - Profile code running on hardware
- **Profile at various levels of detail**
 - Visual display of performance bottlenecks
 - Complete system coverage
 - View Advance card activity across cluster
 - Visibility of compute and I/O overlap
 - Show overlap of host and Advance card compute
 - Drill down for detailed profiling of CSX600 execution



Er... what about the unified source?

- **Good question**

- Currently there is research activity between Intel and ClearSpeed
- Technology demonstrator was shown at IDF

Intel IDF content at:

<https://intel.wingateweb.com/us/catalog/controller/catalog>

QATS003:

Accelerator Exoskeleton: Intel® Architecture Look and Feel for Heterogeneous Cores

In Summary

- **Moore's law continues**
 - Number of cores now doubling every 18 months - 32 cores per socket by end of 2011
 - This means that the next large computer you buy will have to help develop the code to run on these 32 cores/socket
- **The world goes green**
 - Power consumption is chief concern for system architects
 - Power efficiency is the primary growing concern of consumers of computer systems
 - Transition to specialist low-power accelerators
 - Happening today for floating point
 - Security, Java, xml...
- **It's still about the software**

Thank you

ben.bennett@clearspeed.com

Cⁿ code example

Accelerated application code

Standard C code

```
double BlackScholes(...)  
{  
    ...  
}  
  
int main()  
{  
    double a[NUM_INPUTS];  
    ...  
    for (i=0; i<NUM_INPUTS; i++)  
        a[i] = BlackScholes(...);  
    ...  
}
```

Ported to Cⁿ

```
poly double BlackScholes(...)  
{  
    ...  
}  
  
int main()  
{  
    poly double a[NUM_INPUTS/96];  
    ...  
    for (i=0; i<NUM_INPUTS/96; i++)  
        a[i] = BlackScholes(...)  
    ...  
}
```

BlackScholes function: embarrassingly parallel

Standard C code

```
double CND(double X);

double BlackScholes(char CallPutFlag,
                   double S,
                   double X,
                   double T,
                   double r,
                   double v)
{
    double d1, d2, v1, v2, t1, t2, vc1, vc2, vc,
           vp1, vp2, vp;

    t1 = S * exp(-r*T);
    t2 = X * exp(-r*T);
    v1 = log(S/X) + (v*v/2)*T;
    v2 = v * sqrt(T);
    d1 = v1/v2;
    d2 = (v1/v2)-v2;
    vc1 = t1 * CND(d1);
    vc2 = t2 * CND(d2);
    vc = vc1 - vc2;
    vp1 = t1 * CND(-d1);
    vp2 = t2 * CND(-d2);
    vp = vp2 - vp1;

    return (CallPutFlag=='c') ? vc : vp;
}
```

Ported to Cⁿ

```
poly double CND(poly double X);

poly double BlackScholes(poly char CallPutFlag,
                          poly double S,
                          poly double X,
                          poly double T,
                          poly double r,
                          poly double v)
{
    poly double d1, d2, v1, v2, t1, t2, vc1, vc2, vc,
               vp1, vp2, vp;

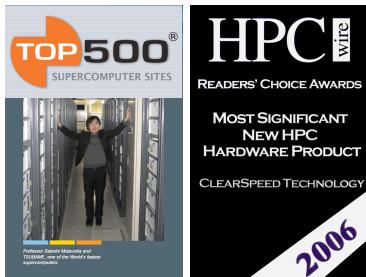
    t1 = S * exp(-r*T);
    t2 = X * exp(-r*T);
    v1 = logp(S/X) + (v*v/2)*T;
    v2 = v * sqrt(T);
    d1 = v1/v2;
    d2 = (v1/v2)-v2;
    vc1 = t1 * CND(d1);
    vc2 = t2 * CND(d2);
    vc = vc1 - vc2;
    vp1 = t1 * CND(-d1);
    vp2 = t2 * CND(-d2);
    vp = vp2 - vp1;

    return (CallPutFlag=='c') ? vc : vp;
}
```

The CND function calculates the cumulative normal distribution and is, again, almost identical in the C and Cⁿ versions

ClearSpeed

Overview

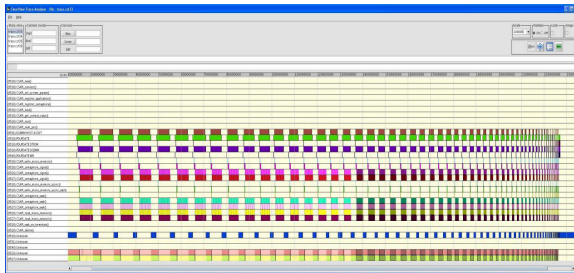
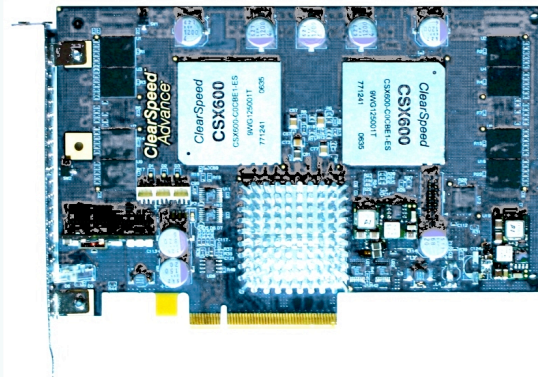
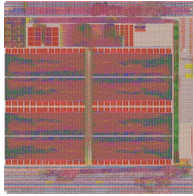


ClearSpeed company background

- **Fabless semiconductor company based in Bristol and San Jose**
 - CSX6000 processor manufactured by IBM
 - boards assembled and tested by Flextronics
- **Core Products**
 - World's highest performance, lowest power consumption processors for Double Precision Floating Point (IEEE 754 compliant*)
 - Accelerators for PCI expansion slots in servers and workstations
 - Work alongside 32 or 64 bit Intel and AMDx86 (or compatible) processors to accelerate specific functions
- **Market Focus**
 - Acceleration of High Performance Computing (HPC) applications
 - Specific focus on Financial Services, Universities & National Laboratories
 - Specialist embedded usage in consumer and military applications
- **Competitive Position**
 - Only supplier of custom designed HPC focussed acceleration products
 - Uniquely positioned to exploit growing HPC acceleration need
 - Substantial Intellectual Property base with patents granted/pending

*The CSX600 floating point units conform to the IEEE 754 standard for 64 bit double precision and 32 bit single precision numbers with the following exceptions:
Denormalized numbers are not supported, denormals are treated as zero on input.
Round to Nearest Even and Round to Zero are the only rounding modes supported.

ClearSpeed product overview

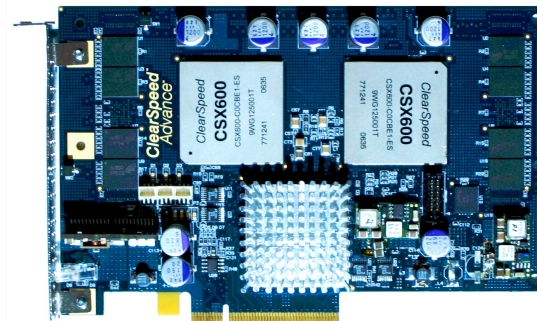


- **CSX600 Processors**
 - World's highest performance, most energy efficient processor for double precision floating point applications
 - 96 processing cores
 - 40 DP GFLOPS peak, >33 GFLOPS DGEMM
 - 10 watts (typical)
- **Advance™ PCI-X and PCIe Accelerators**
 - Exploit standard expansion slots for servers, workstations and blade expansion units
 - >66 GFLOPS DGEMM per accelerator
 - 25 – 33 watts (typical)
- **Software**
 - Linux® and Microsoft® drivers
 - ClearSpeed, Intel, and AMD standard libraries
 - Software Development Kit
 - Visual Profiler and Debugging Tools
 - Wolfram Mathematica® certification
 - Application level acceleration

The ClearSpeed Advance accelerator family



Advance X620



Advance e620

- **The only accelerator family specifically designed for HPC**
 - Industry leading energy efficiency at > 2 GFLOPS per watt
 - Advance e620
 - PCIe x8, standard height: 98 mm (3.9 in), half length: 167 mm (6.5 in)
 - Advance X620
 - PCI-X, standard height: 98 mm (3.9 in), two-thirds length: 203 mm (8.0 in)
 - Plug & Play acceleration with standard math libraries including Level 3 BLAS and LAPACK
 - Fully programmable in Cⁿ extended parallel language

Acceleration for finance and science applications

- **Finance**
 - Up to 20x speedup per accelerator for Monte Carlo based option pricing
- **Universities and National Laboratories**
 - 3.4x to 11.0x speedup for scientific applications (molecular modeling)
- **Multiple accelerators per host system**
 - Greater perf./watt for data parallel problems (MC, Amber, etc.)
- **Industry leading space and energy efficiency**

