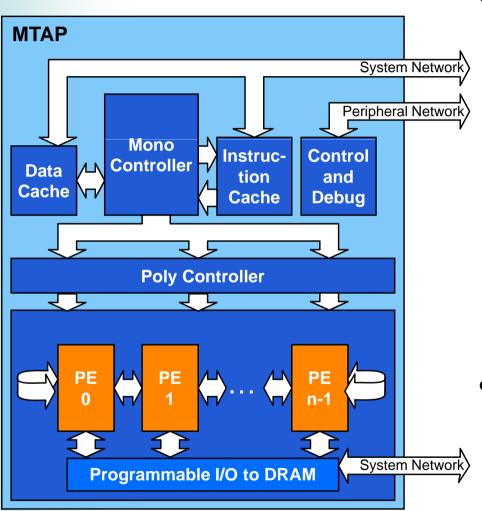
ClearSpeed

"A next-generation many-core processor with reliability, fault tolerance and adaptive power management features optimized for embedded and high performance computing applications."

Simon McIntosh-Smith, VP of Applications simon@clearspeed.com

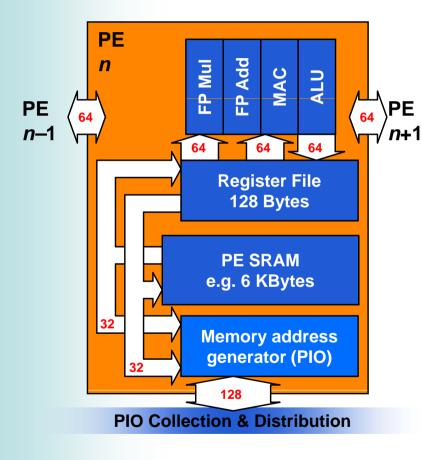
- ClearSpeed Federal Systems based in San Jose, CA, founded in 2008
- ClearSpeed Technology HQ in Bristol, UK, founded in 2002
- Deliver the world's most power efficient, high-performance processors
- Over 100 patents on the core technology
- Solutions for defence, embedded systems and high performance computing
- Licensed IP to BAE Systems for use in future space-based systems
- Partnering with HP, SGI, IBM, and Sun

ClearSpeed's "Smart SIMD" MTAP processor core



- "Smart SIMD" Multi-Threaded Array Processing:
 - Multi-threading for asynchronous, overlapped I/O with compute
 - Scalable array of many Processor Elements (PEs)
 - Coarse-grained data parallel processing
 - Supports redundancy and resiliency
- Programmed in an extended version of ANSI C called Cⁿ:
 - Rich expressive semantics
 - Single "poly" data type modifier

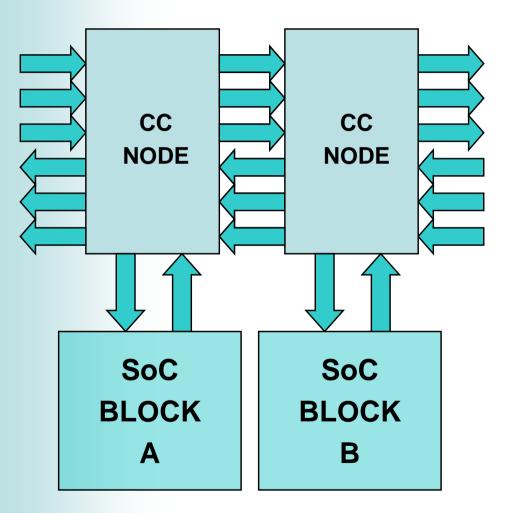
Smart SIMD Processing Elements (PEs)



- Multiple execution units per PE
 - Floating point adder

- 32 & 64-bit IEEE 754
- Floating point multiplier
- Fixed-point MAC 16x16 \rightarrow 32+64
- Integer ALU with shifter
- Load/store
- Fast inter-PE communication path
- Closely coupled, ECC-protected SRAM for data
- High bandwidth per PE DMA (PIO)
- Per PE address generators
- Platform design enables PE variants

The ClearConnect[™] "Network on Chip" system bus



- Scalable, System on Chip (SoC) platform interconnect
- Used to connect together all the major blocks on a chip:
 - Multiple MTAP smart SIMD cores
 - Multiple memory controllers
 - On-chip memories
 - System interfaces, e.g. PCI Express
- Unified memory architecture
- Distributed arbitration
- Scalable bandwidths
- Low power design

ClearSpeed's product range

- Processors
 - CSX700, CSX600
- Boards
 - e720, e710, e620, X620
- Systems
 - CATS-700 1TFLOP in 1U
- Software
 - Compiler
 - Libraries
 - Heterogeneous profiler
 - GDB-based debugger





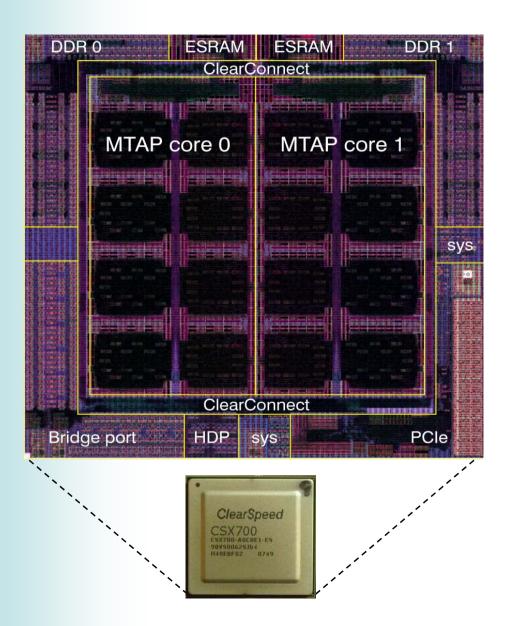








The CSX700 Processor



- Includes dual MTAP cores:
 - 96 GFLOPS peak (32 & 64-bit)
 - 48 GMACS peak (16x16 \rightarrow 32+64)
 - 10 power consumption
 - 250MHz clock speed
 - 192 Processing Elements (2x96)
 - 8 spare PEs for resiliency
 - ECC on all internal memories
- Dual integrated 64-bit DDR2
 memory controllers with ECC
- Integrated PCI Express x16
- CCBR chip-to-chip bridge port
- 2x128 KBytes of on-chip SRAM
- IBM 90nm process
- 266 million transistors

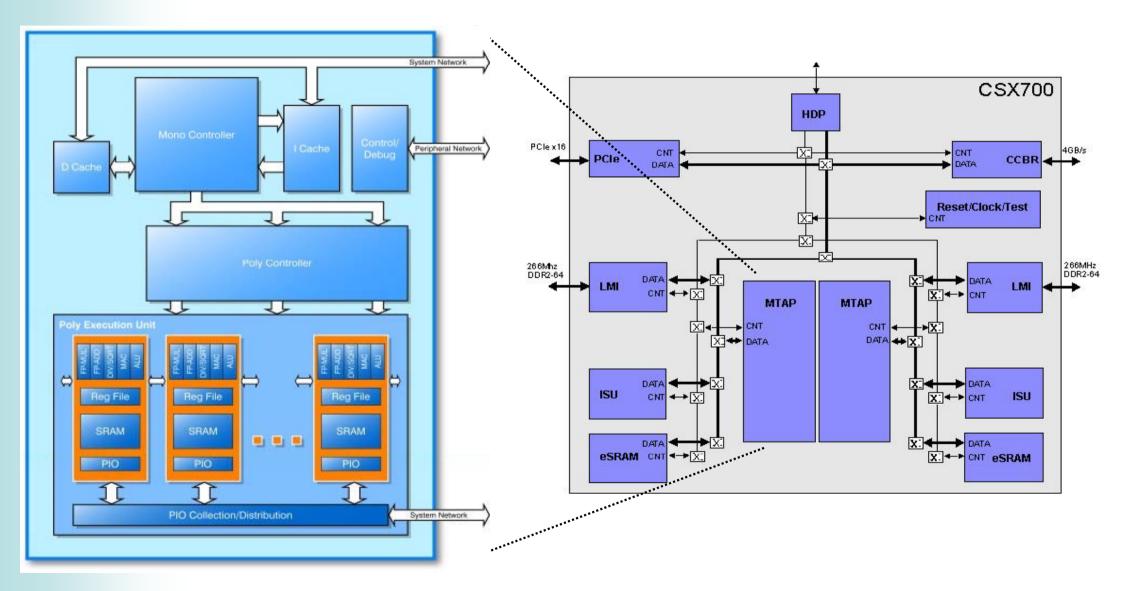
Reliability and fault tolerance

- 8 spare PEs can be kept in reserve
- Error Correcting Codes (ECC) on all on-chip memories
- ECC with memory scrubber on all external memories
- On-die temperature sensors
- Low power and modest clock speed design
- Programming model supports graceful degradation

Adaptive power management

- Ability for software to modify clock speed dynamically, even while an application is running
- On-die temperature sensors allow for thermal ceilings to be set by software
- Can also throttle clock speed based on power consumption

CSX700 – Internal Architecture



Powerful software development environment

- Version 3.11 launched in June 08
- Binary compatible across all ClearSpeed products
- ANSI C-based optimising compiler Cⁿ
- GDB-based debugger



- Profiling with heterogeneous, system-wide capabilities
- Libraries (BLAS, RNG, FFT, more...) & High level APIs (CSPX)
- ECLIPSE-based IDE



Standard Eclipse graphical debug interface for CSX processor debugging.

CSX processor provides full hardware debugging of application code.

Provides a seamless view of many processor cores in parallel with their associated state.

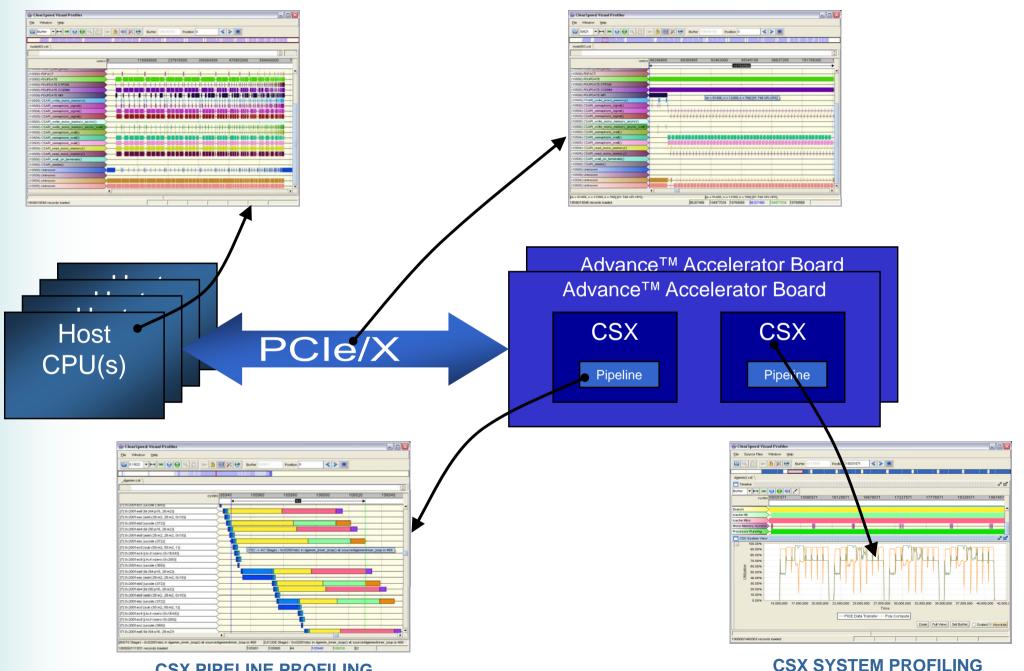
Allows full symbolic debug of the C^n language.

Enhanced views for CSX specific information.

Debug - example1_csx/src/example1_csx.cn - Eclipse SDK							
<u>F</u> ile <u>E</u> dit Refac <u>tor N</u> avigate Se <u>a</u> rch <u>P</u> roject <u>R</u> un <u>W</u> indow <u>H</u> elp							
- 	<u>.</u>		*5 ~			😭 🏇 Debug	»
🕸 Debug 🛿 🛛 🐐 🤣 🕩 💷 🔳 🔐 💽 🖓 🔐 🧮 🖬 🛒 🔽 🦳 🖬				oles 🔀 💊 Breakpoints	👬 Regist	ters 🛋 Modules	- 8
▽ 🕵 example1_csx Debug [CSX Application]						約 🕫 🖻 💕 💥	×
▽ 🖓 ClearSpeed Debugger (16/08/07 17:30) (Suspended)			Name			Value	
🗢 📫 Thread [1] (Suspended)			(×)= nu	mpes		96	
2 main() /mounts/vdrive/mikew/runtime-New_configuration/example 1_csx/src/example				slice		960000	
= 1_start() 0x02000c94				_start_val		[00]	
D 🔐 Thread [2] (Suspended)			(×)=O		0		
🛁 csgdb (16/08/07 17:30) 📲 /mounts/vdrive/mikew/runtime-New_configuration/example1_csx/Debug/example1_csx.csx (1)=1	9 60000		
)=2	1920000		
			(×)= 3	2880000		
			(×)= 4	3840000		
			(×)= 5	4800000 5760000		
)= 6			
				• ///			<u> </u>
•		>	4			· · · · · · · · · · · · · · · · · · ·	*
🖻 example1_csx.cn 🕱 💽 lib_ext.h 🛛 💽 swazzle.h	尾 get_swaz	zle_low.h				🗉 Outline 🔀	- 0
<pre>int main() { poly double series = 0; poly double series = 0; poly short penum = get_penum(); mono int numpes = get_nmm.pes(); mono int p_slice = TMES/numpes; poly int p_start_val = pe_slice*penum; mono double result = 0; mono int i = 0; mono int i = 0; mono int start, end; start = get_cycles(); for (i = 0; i < pe_slice; i++) { series += 1.0 / ((pe_start_val+i+1.0) * (pe_start_val+i+1.0)); } end = get_cycles(); result = pm_add(series); Pi = sort(6.0 * result); /// </pre>				<pre># TIMES # TIMES # math.h # mathp.h # lib_ext.h #</pre>			
📮 Console 🖉 Tasks 🕱 🛛 🖹 Problems						d 🗶 🔅	
items						1 A A A A A A A A A A A A A A A A A A A	
✓ ! Description	Resource	Path		Location			
. Description	Resource	rau		Location			
□◆]		

ClearSpeed profiler for heterogeneous multi-processor systems

HOST/BOARD INTERACTION PROFILING



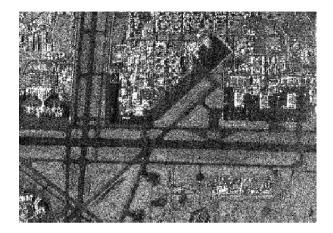
HOST CODE PROFILING

CSX PIPELINE PROFILING

Defense focus

Processor technology with a signal-processing focus

- Radar, Sonar
- Imaging
- Target discrimination
- Electronic warfare
- Communications intelligence



Engineered for optimal Size, Weight, and Power

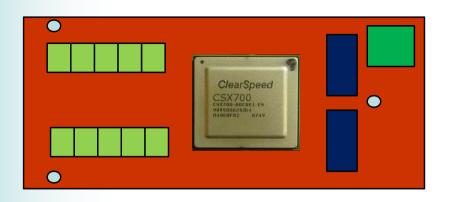
• Best performance per watt in the industry

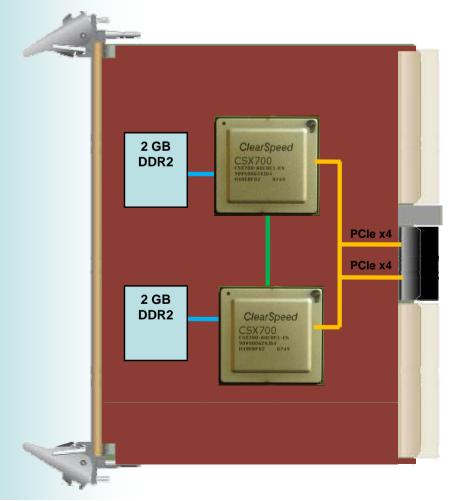
Deployable across harsh environments

- Air, land, and sea
- Commercial, rugged, conduction cooled configurations
- Embedded form factors



Embedded/Defense Concept XMC and VXS modules





XMC card

- Performance
 - 96 GFLOPS, 48 GMACS
 - 192 GBytes/s peak on-chip bandwidth
 - 8 GBytes/s peak off-chip
- High bandwidth via PCI Express I/O
 - Up to 2 GBytes/sec via PCIe x2,x4,x8
 - Up to 4 GBytes/sec via PCIe x16
- 15 watts typical, 25 watts max

Vita 41 (VME Switched Serial - VXS) Module

- Performance
 - 192 GFLOPS, 96 GMACS
 - 384 GBytes/s peak on-chip bandwidth
 - 16 GBytes/s peak off-chip

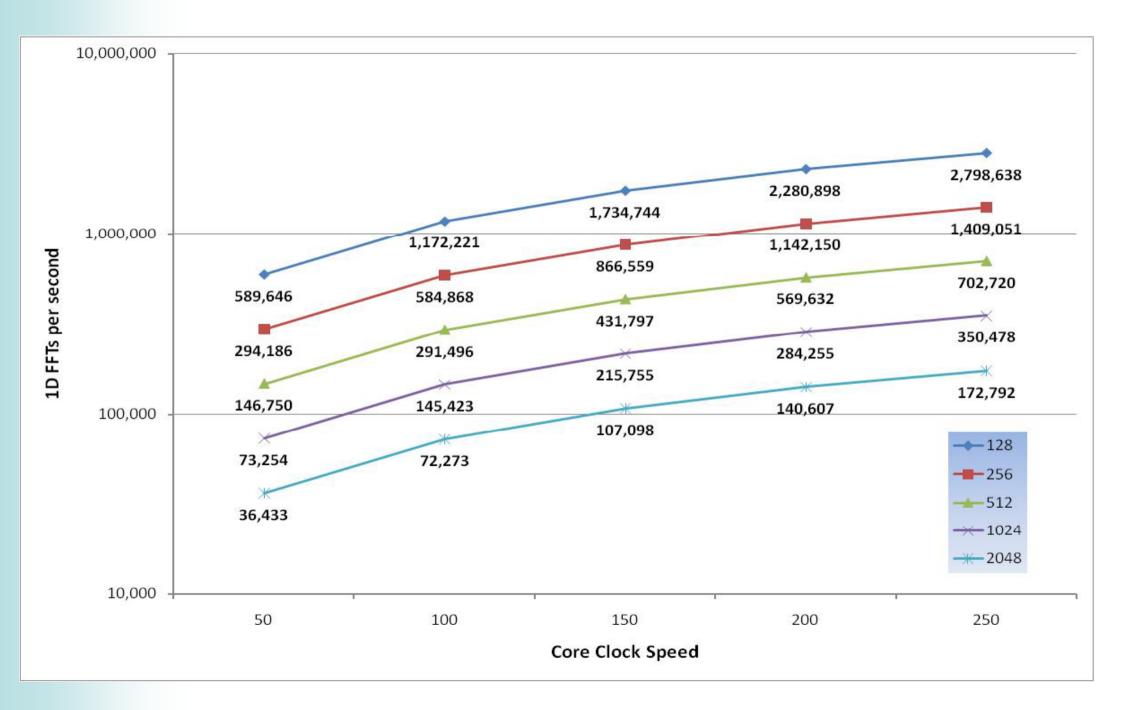
High bandwidth

- VME interface (P1 & P2) PCIe x4
- ClearConnect CCBR for chip-to-chip
- (2) 4x PCIe links off-board (Vita 41.4)
- 30 watts typical, 50 watts max

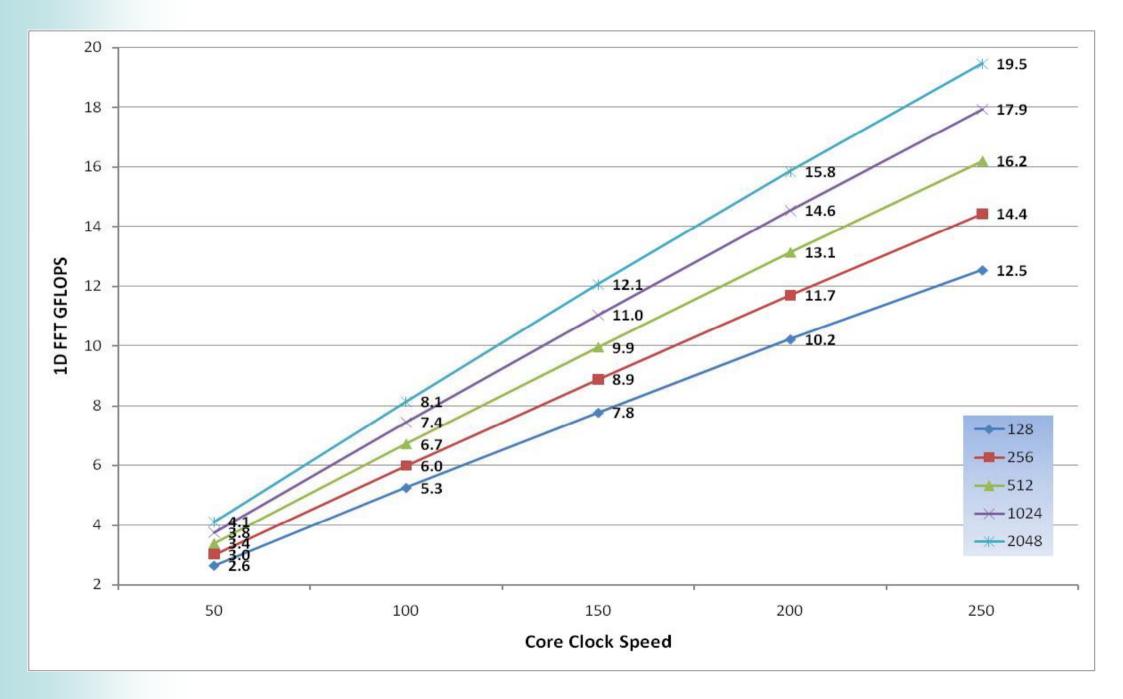
CONCEPTS, NOT COMMITTED PRODUCTS

- Power consumption figures are for a single CSX700 running at 0.9V and 42°C
- For 2D FFTs, timing and power consumption measurements include the corner turn
- All data always starts and finishes in the off-chip DRAM
- FFTs were run simultaneously on both MTAP cores of the CSX700

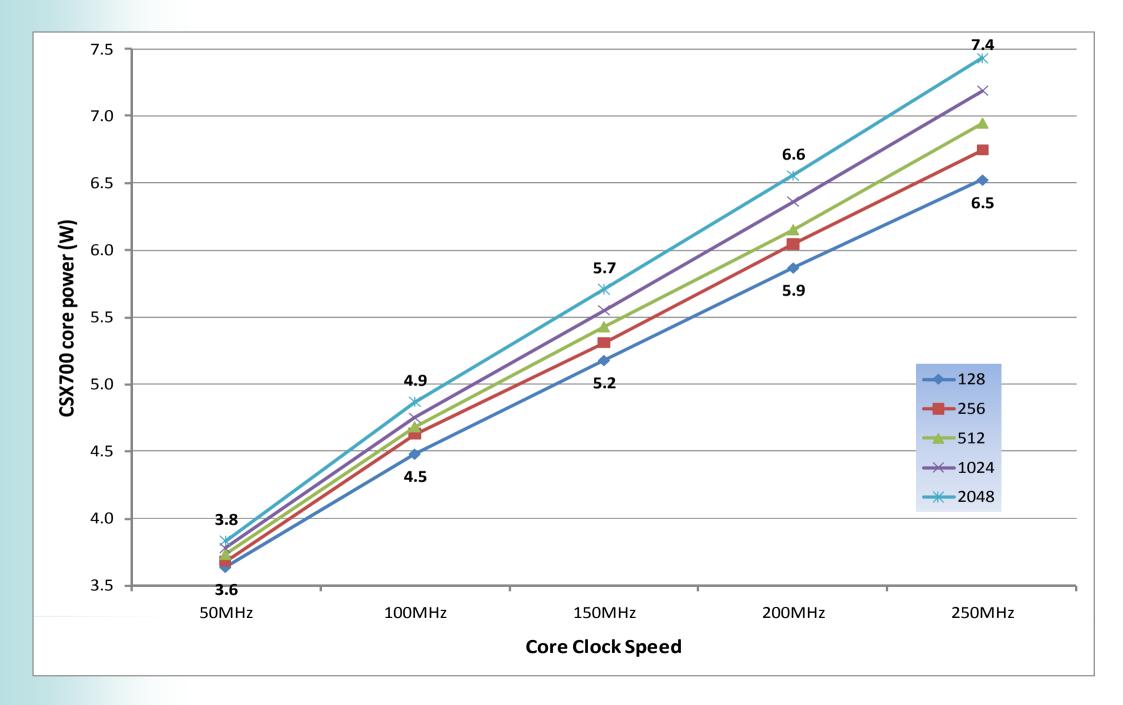
CSX700 1D FFTs per second



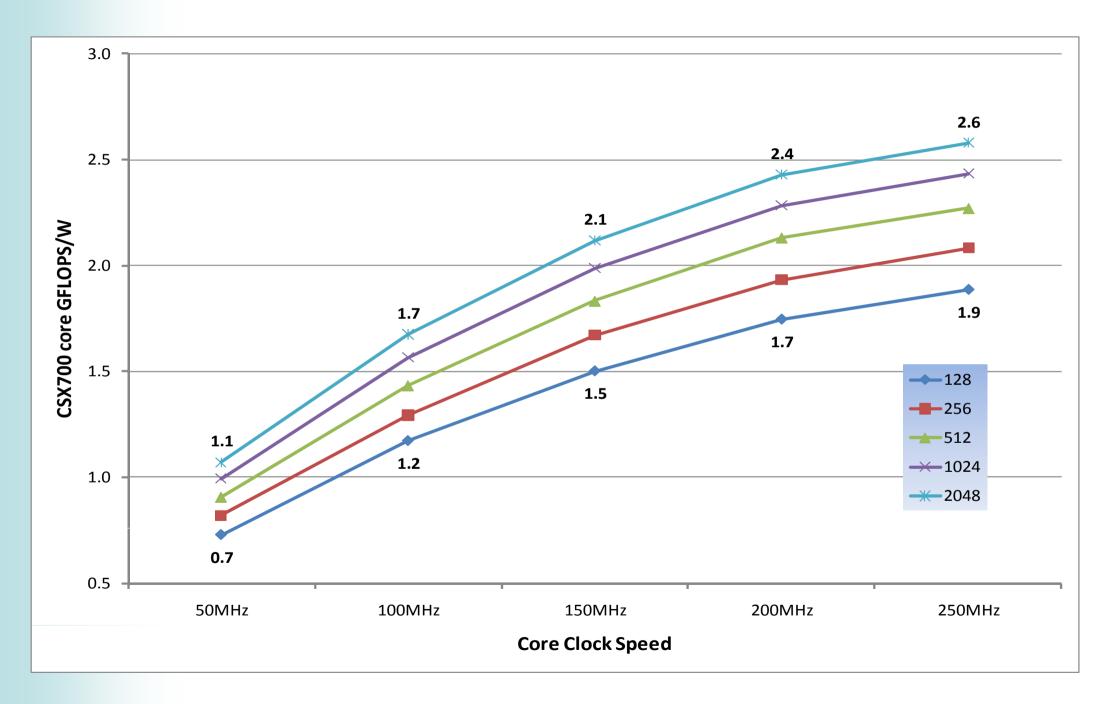
CSX700 1D FFT GFLOPS



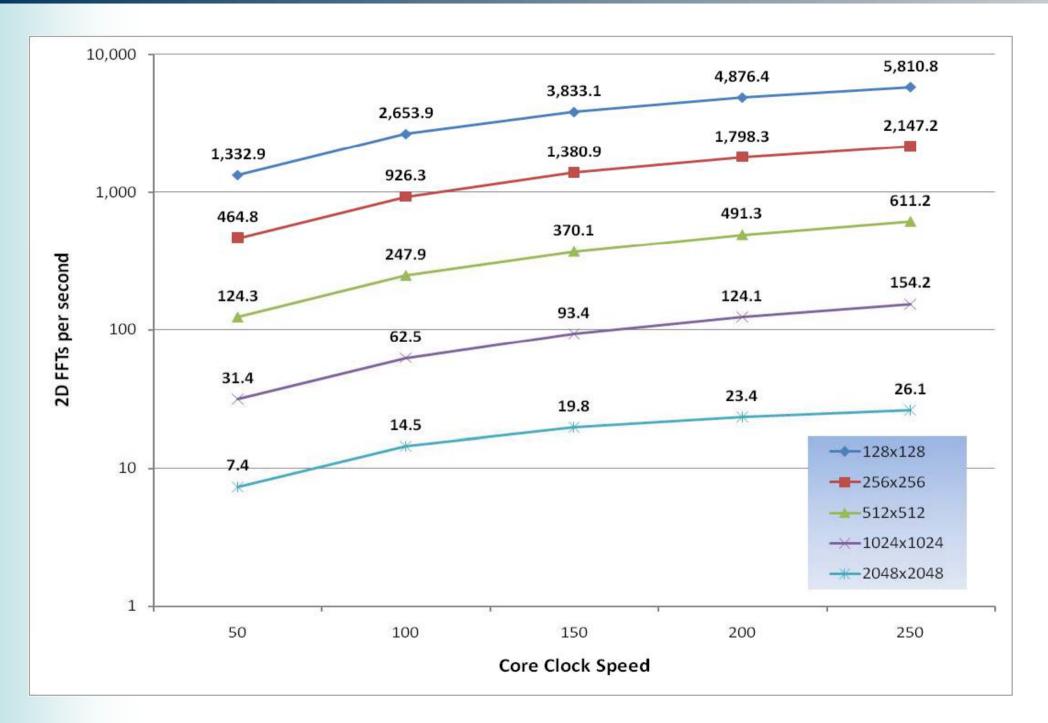
CSX700 1D FFT core power



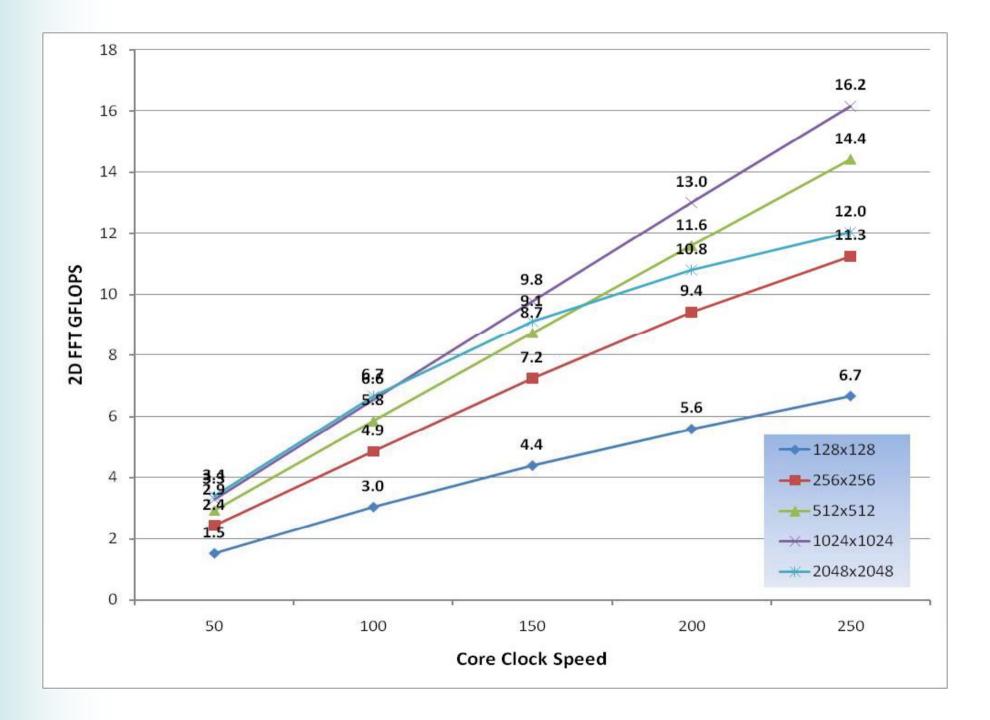
CSX700 1D FFT core power GFLOPS per watt



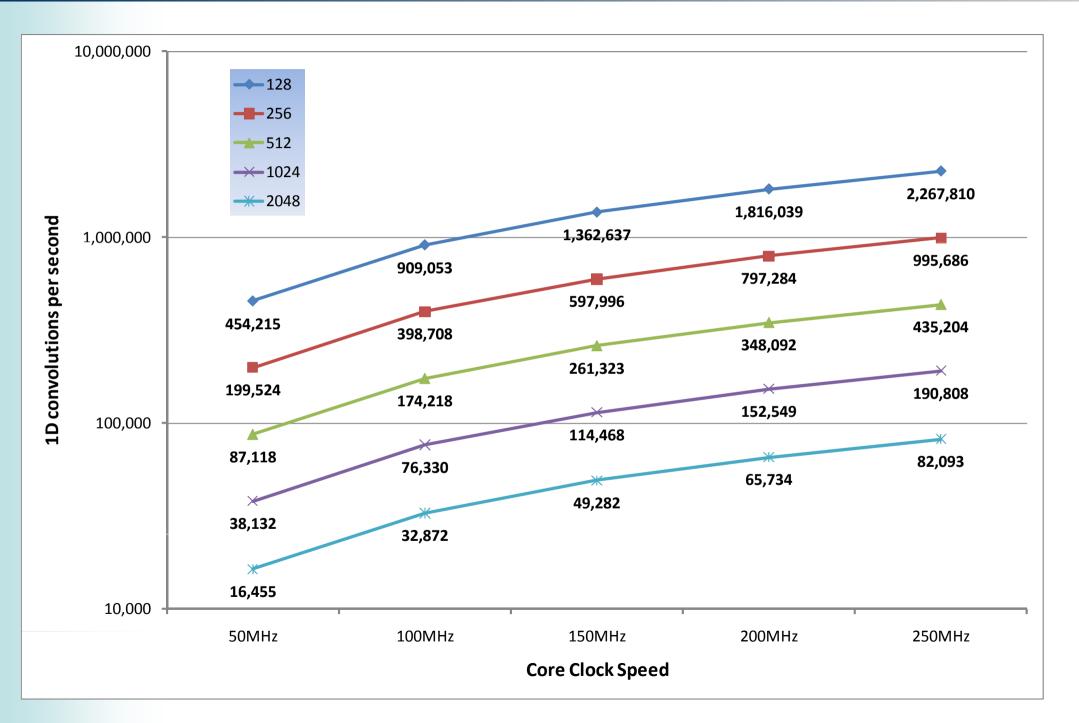
CSX700 2D FFTs per second



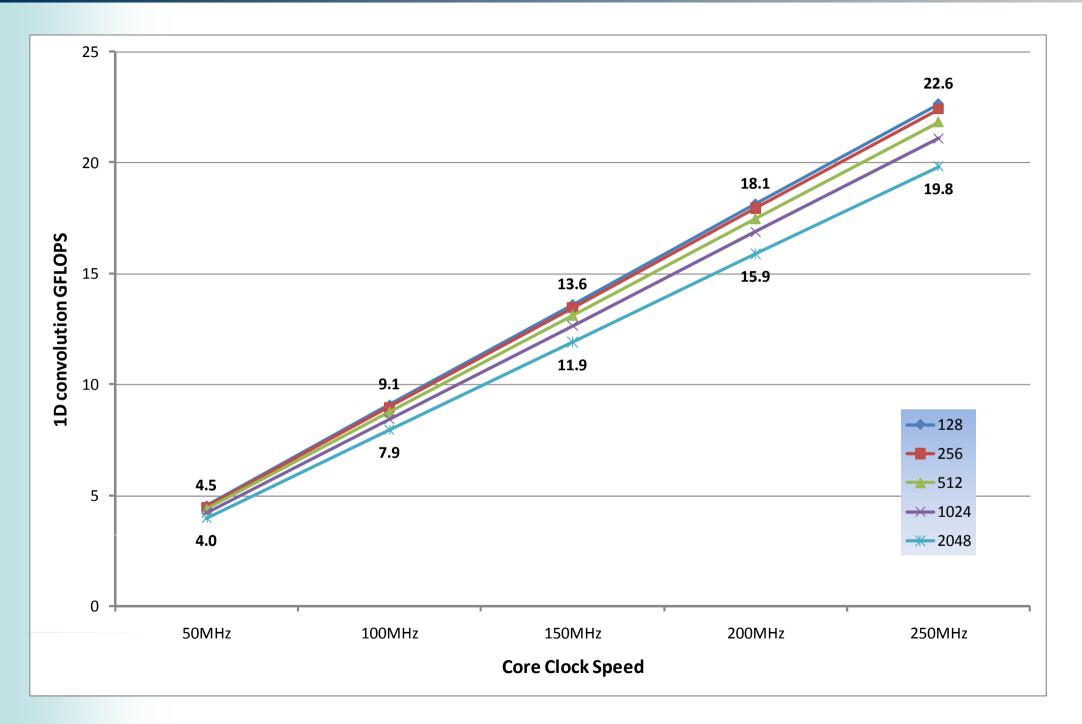
CSX700 2D FFT GFLOPS



CSX700 convolutions per second



CSX700 convolution GFLOPS



CATS-700 FFT performance



With 12 CSX700 processors the CATS-700 can achieve:

- 4.2 million 1024 point 1D FFTs per second (215 GFLOPS)
- 26,000 1024x1024 2D FFTs per second (194 GFLOPS)
- 25 billion 32-bit inverse square roots
- 14 billion 32-bit sines
- 12 billion 32-bit exponentials
- An aggregate STREAM benchmark of ~80 GBytes/s

CSX700 Transcendental Function Performance

