Master Informatics Eng.

2018/19

A.J.Proença

Data Parallelism 2 (*SIMD++, NVidia GPUs...*)

(most slides are borrowed)

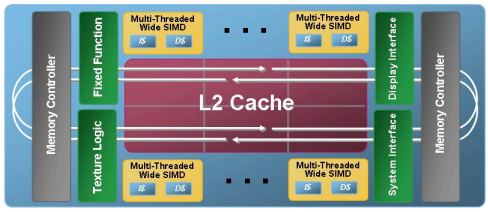
Beyond Vector/SIMD architectures

- Vector/SIMD-extended architectures are hybrid approaches
  - mix (super)**scalar + vector** op capabilities on a single device
  - **highly pipelined** approach to reduce memory access penalty
  - **tightly-closed access to shared memory**: lower latency
- Evolution of Vector/SIMD-extended architectures
  - **PU (Processing Unit) cores with wider vector units**
    - x86 many-core: **Intel** MIC / Xeon KNL
    - ...
  - **coprocessors (require a host scalar processor): accelerator devices**
    - on disjoint physical memories (e.g., **Xeon KNC** with PCI-Expr, ...)
    - ...
  - **heterogeneous PUs in a SoC: multicore PUs with GPU-cores**
    - ...

Intel MIC: Many Integrated Core

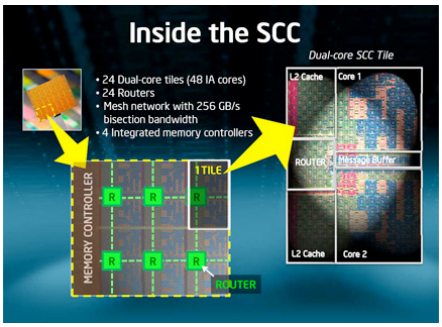
Intel evolution, from:

- Larrabee (80-core GPU)



& SCC

Single-chip
Cloud
Computer,
24x
dual-core tiles

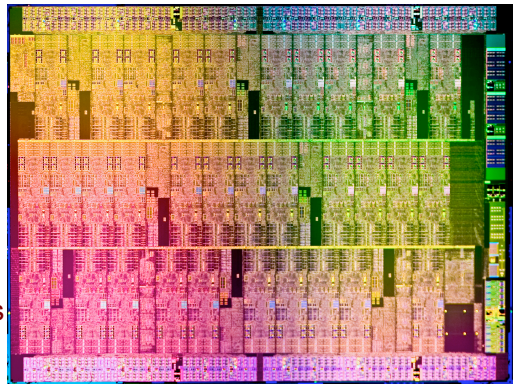


to MIC:

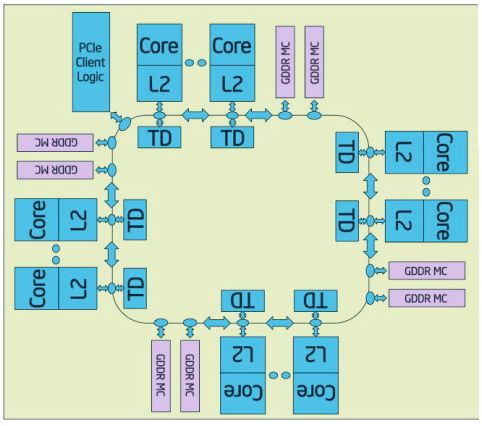
- Knights Ferry (pre-production, Stampede)
- Knights Corner →

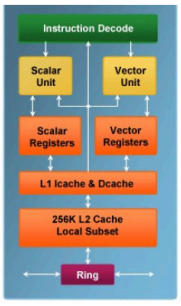
  Xeon Phi co-processor up to 61 Pentium cores
- Knights Landing (~~& Knights Mill...~~)

  Xeon Phi full processor up to 36x dual-core Atom tiles

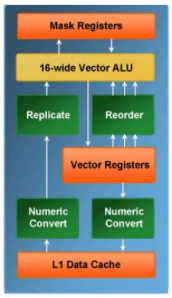


Intel Knights Corner architecture

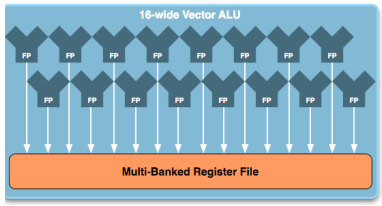


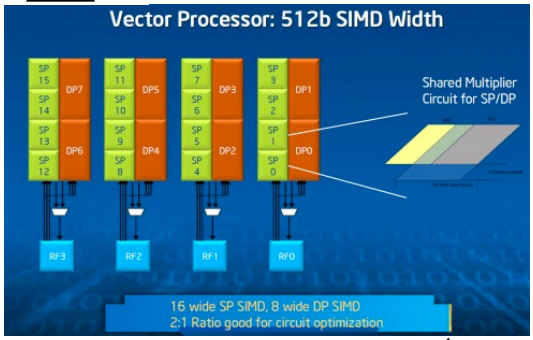


Processor Core

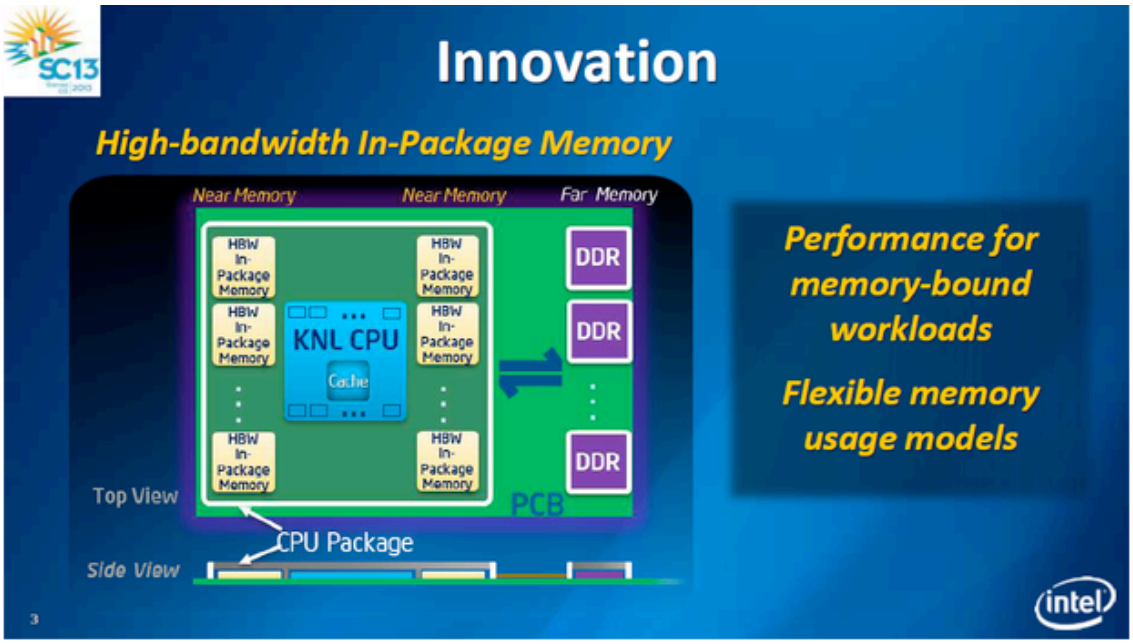


Vector Unit

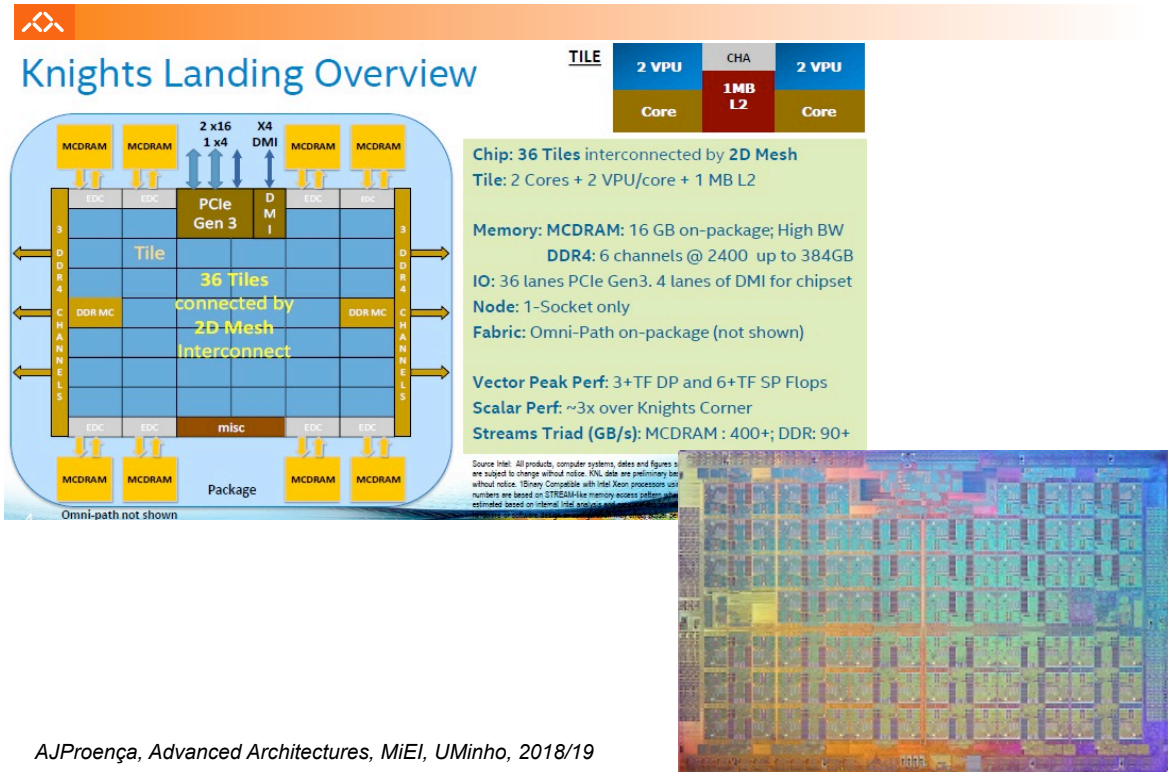




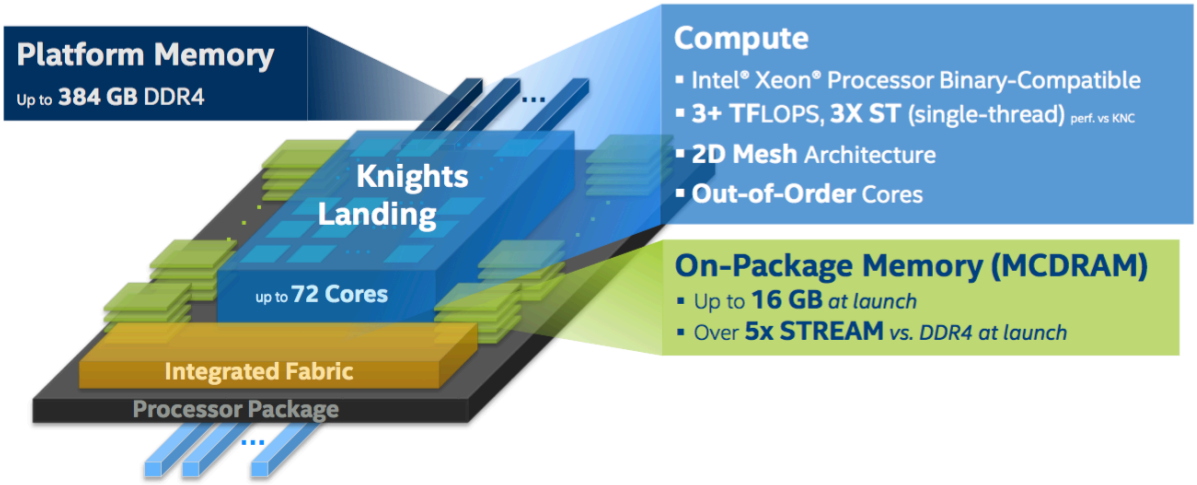
The new Knights Landing architecture



Intel Knights Landing in 2016: Xeon Phi com 72 active cores



INTEL® XEON PHI™ X200 PROCESSOR OVERVIEW



More details in a later set of slides...

Beyond Vector/SIMD architectures

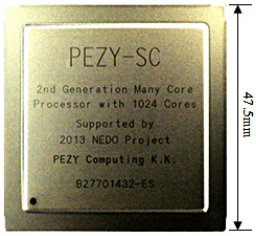
- Vector/SIMD-extended architectures are hybrid approaches
  - mix (super)**scalar + vector** op capabilities on a single device
  - **highly pipelined** approach to reduce memory access penalty
  - **tightly-closed access to shared memory**: lower latency
- Evolution of Vector/SIMD-extended architectures
  - **PU (Processing Unit) cores with wider vector units**
    - x86 many-core: **Intel** MIC / Xeon KNL (*more slides later*)
    - ...
  - **coprocessors (require a host scalar processor): accelerator devices**
    - on disjoint physical memories (e.g., **Xeon KNC** with PCI-Expr, **PEZY-SC**)
    - ...
  - **heterogeneous PUs in a SoC: multicore PUs with GPU-cores**
    - ...

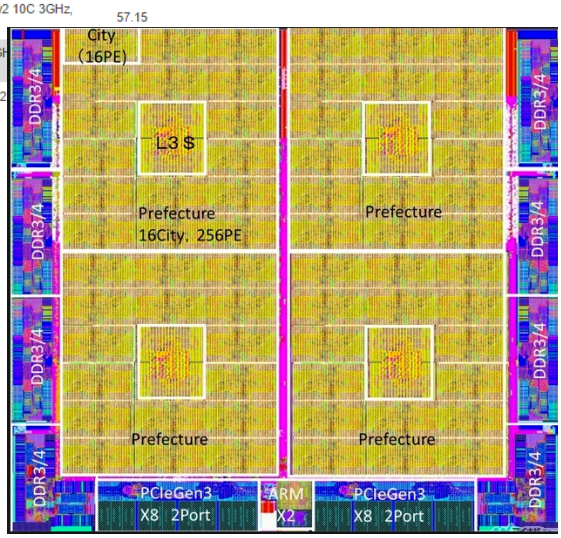
PEZY-SC: Peta_Exa_Zetta_Yotta-SuperComputer: a 1024-core many-core processor chip

Green500 Rank	MFLOPS/W	Site*	Computer*	Total Power (kW)
1 (94)	6,673.84	Advanced Center for Computing and Communication, RIKEN	Shoubu - ZettaScaler-1.6, Xeon E5-2618Lv3 8C 2.3GHz, Infiniband FDR, PEZY-SCnp	149.99
2 (486)	6,195.22	Computational Astrophysics Laboratory, RIKEN	Satsuki - ZettaScaler-1.6, Xeon E5-2618Lv3 8C 2.3GHz, Infiniband FDR, PEZY-SCnp	46.89
3 (1)	6,051.30	National Supercomputing Center in Wuxi	Sunway TaihuLight - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway	15,371.00
4 (440)	5,272.09	GSI Helmholtz Center	ASUS ESC4000 FDR/G2S, Intel Xeon E5-2690v2 10C 3GHz, Infiniband FDR, AMD FirePro S9150	57.15
5 (446)	4,778.46	Institute of Modern Physics (IMP), Chinese Academy of Sciences	Sugon Cluster W780I, Xeon E5-2640v3 8C 2.6GH QDR, NVIDIA Tesla K80	
6 (122)	4,112.11	Stanford Research Computing Center	XStream - Cray CS-Storm, Intel Xeon E5-2680v2 Infiniband FDR, Nvidia K80	

Top500 Rank

Green500 list
June'2016

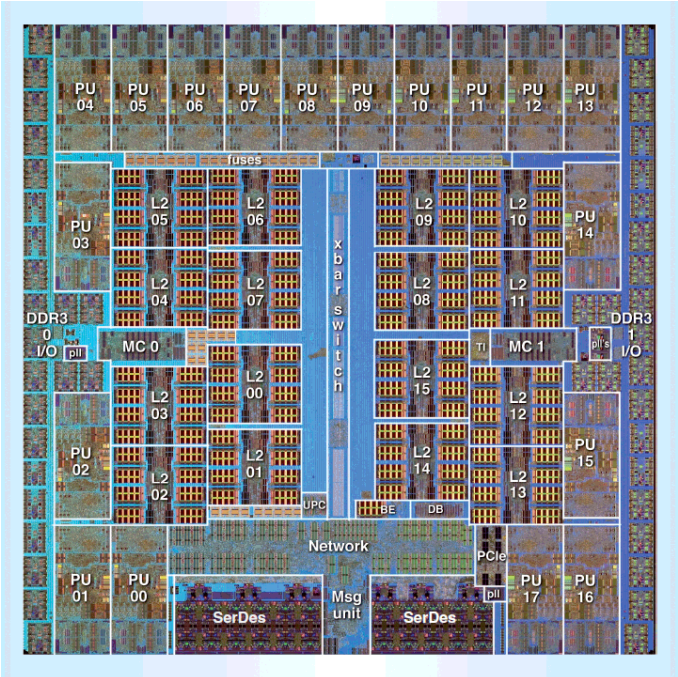




Beyond Vector/SIMD architectures

- Vector/SIMD-extended architectures are hybrid approaches
  - mix (super)**scalar + vector** op capabilities on a single device
  - **highly pipelined** approach to reduce memory access penalty
  - **tightly-closed access to shared memory**: lower latency
- Evolution of Vector/SIMD-extended architectures
  - **PU (Processing Unit) cores with wider vector units**
    - x86 many-core: **Intel** MIC / Xeon KNL (*more slides later*)
    - other many-core: **IBM** Power BlueGene/Q Compute, **ShenWay** 260
  - **coprocessors (require a host scalar processor): accelerator devices**
    - on disjoint physical memories (e.g., **Xeon KNC** with PCI-Expr, **PEZY-SC**)
    - ...
  - **heterogeneous PUs in a SoC: multicore PUs with GPU-cores**
    - ...

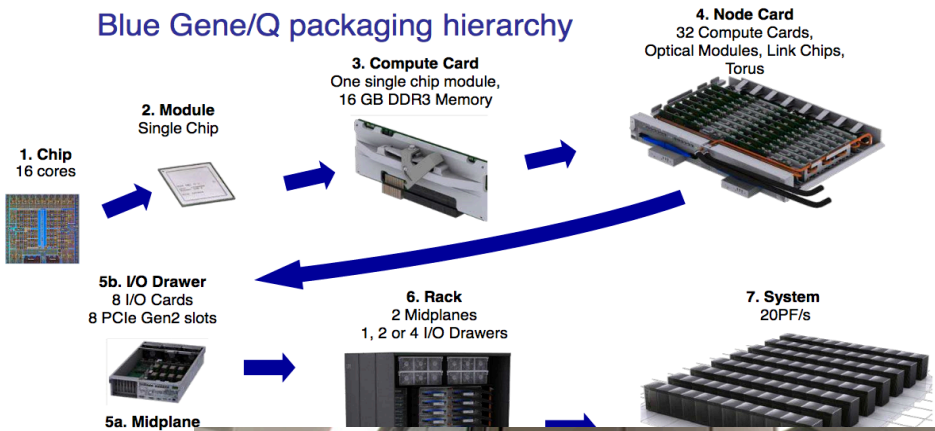
IBM Power BlueGene/Q Compute *(chip)*



Features:

- launched in 2010/11
  (TOP500: #1 in Jun12, #4 in Jun16)
- 18-cores
  - 16 compute,
    1 OS support, 1 redundant
  - 64 bits PowerISA
  - 1.6 GHz
  - L1 I/D cache => 16 kB / 16 kB
  - **each core: quad-FPU**
    (4-wide double precision SIMD)
  - each core: 4-way multi-threaded
- shared L2 cache: 32 MB
- dual memory controller
- IBM ended development of BlueGene project in 2015...

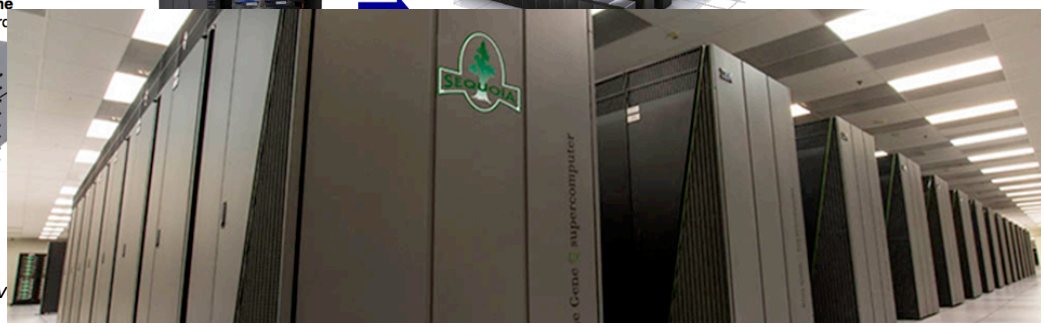
IBM Power BlueGene/Q Compute *(Sequoia system)*



Ref: SC2010

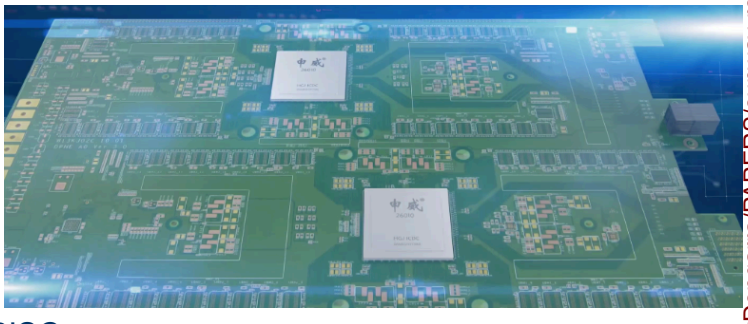
TOP500 The List.

Jun'12: #1
Nov'12: #2
Jun'13: #3
Nov'13: #3
Jun'14: #3
Nov'14: #3
Jun'15: #3
Nov'15: #3
Jun'16: #4
Nov'16: #4
Jun'17: #5

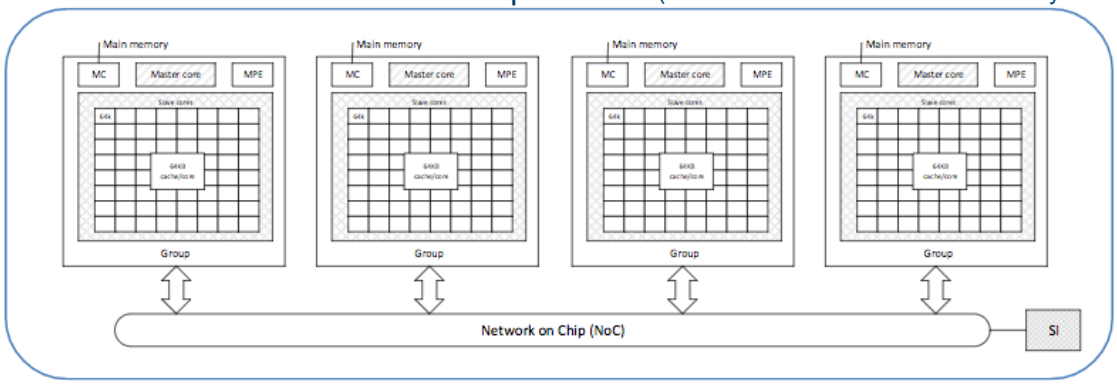


The Sunway TaihuLight

One card with two nodes
(two SW26010 chips)



SW26010: the 4x64-core 64-bit RISC processor (w/ **256-bit vector** instructions & only cache L1)



3

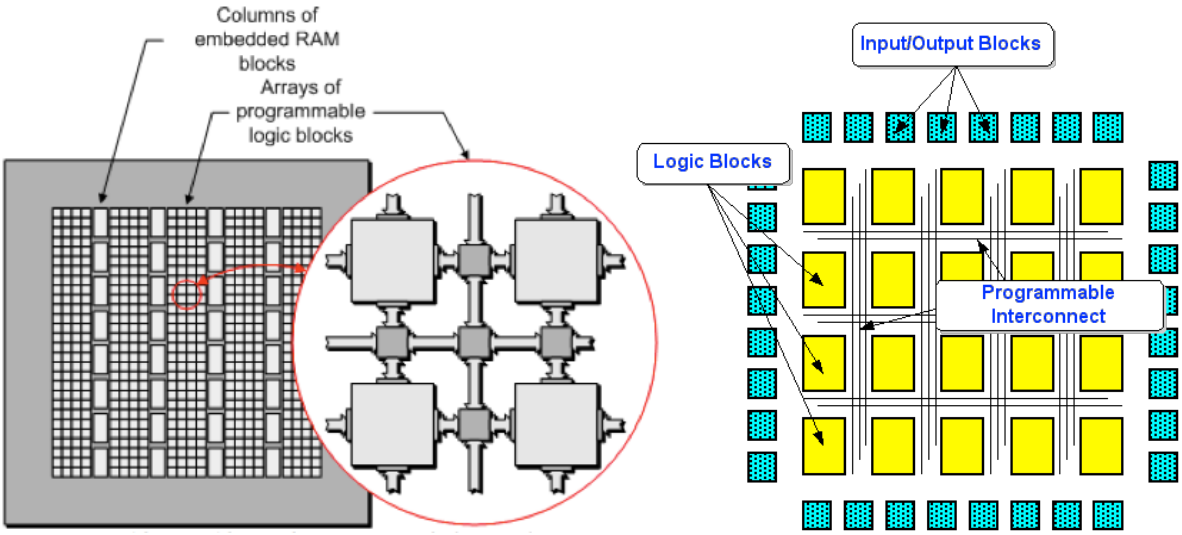
http://www.netlib.org/utk/people/JackDongarra/PAPERS/sunway-report-2016.pdf

Beyond Vector/SIMD architectures

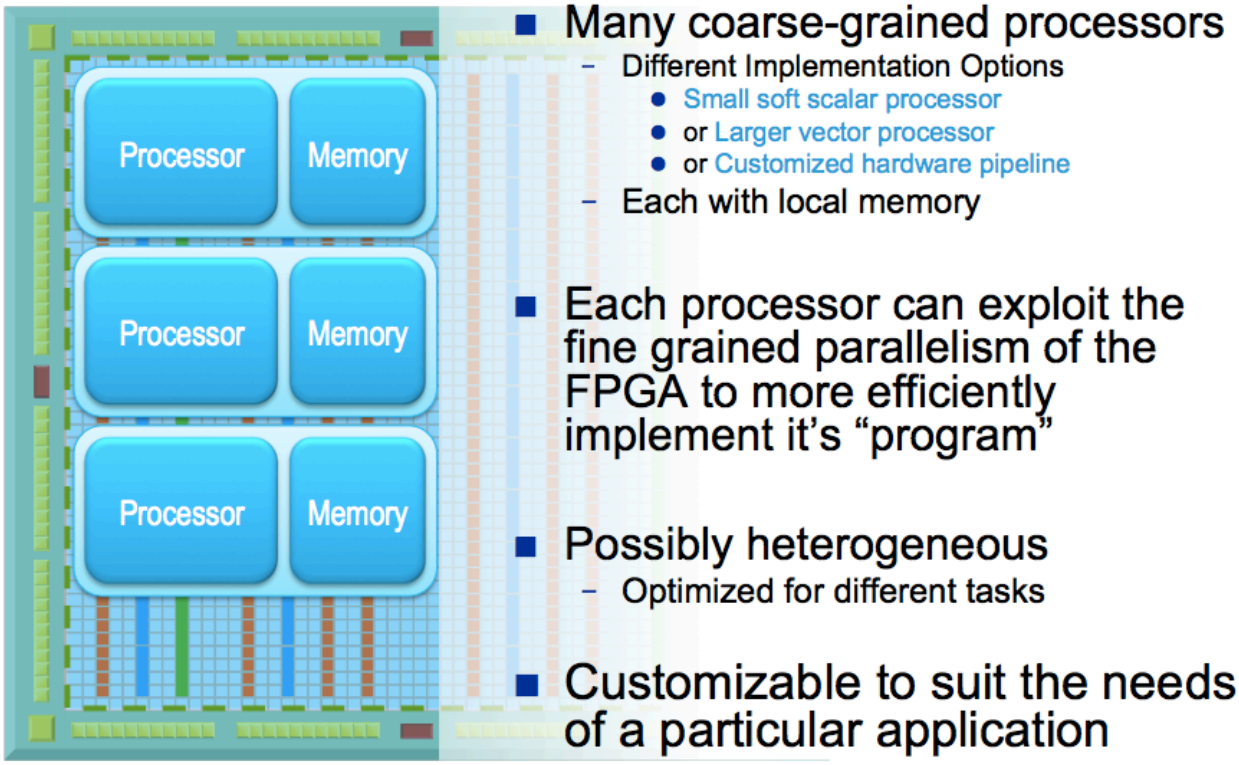
- Vector/SIMD-extended architectures are hybrid approaches
  - mix (super)**scalar + vector** op capabilities on a single device
  - **highly pipelined** approach to reduce memory access penalty
  - **tightly-closed access to shared memory**: lower latency
- Evolution of Vector/SIMD-extended architectures
  - **PU (Processing Unit) cores with wider vector units**
    - x86 many-core: **Intel** MIC / Xeon KNL (*more slides later*)
    - other many-core: **IBM** BlueGene/Q Compute, **ShenWay** 260
  - **coprocessors (require a host scalar processor): accelerator devices**
    - on disjoint physical memories (e.g., **Xeon KNC** with PCI-Expr, **PEZY-SC**)
    - ISA-free architectures, code compiled to silica: **FPGA**
    - ...
  - **heterogeneous PUs in a SoC: multicore PUs with GPU-cores**
    - ...

Field-Programmable Gate Arrays (FPGA)

A fabric with 1000s of simple configurable logic cells with LUTs, on-chip SRAM, configurable routing and I/O cells

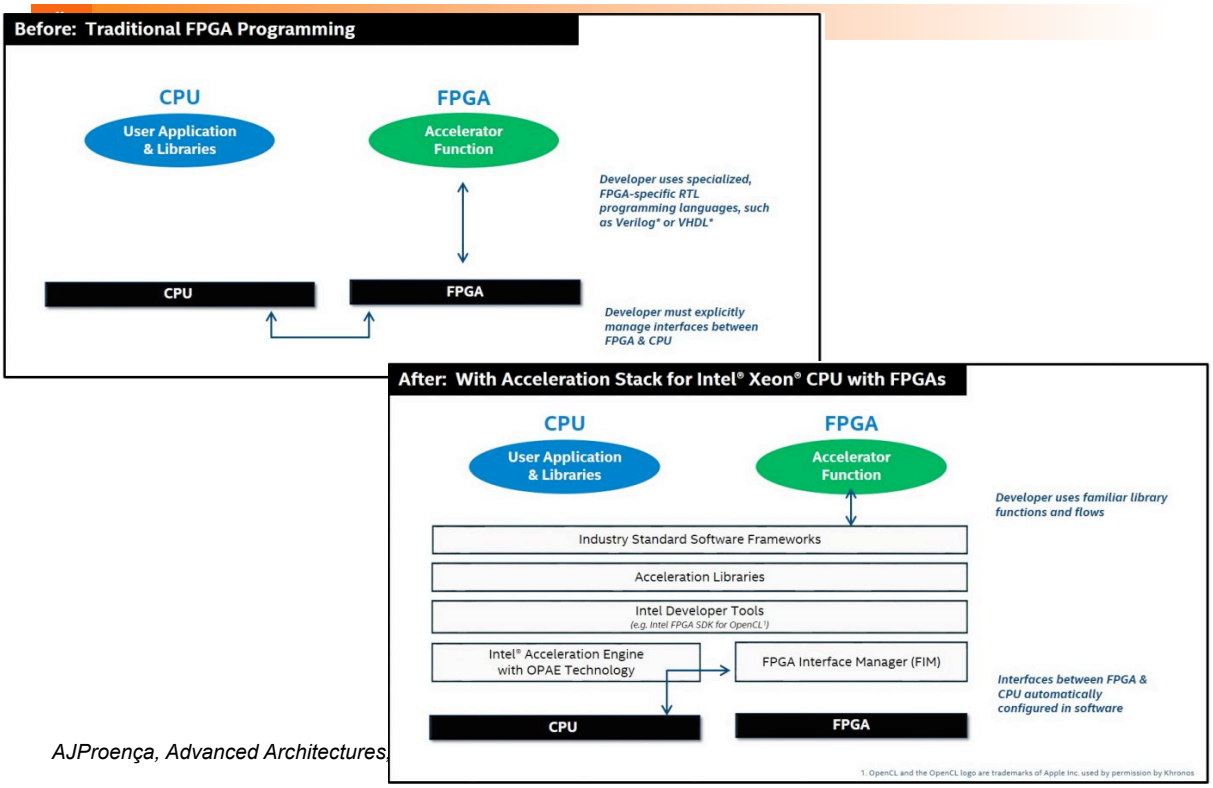


FPGA as a multiple configurable ISA



- Many coarse-grained processors
  - Different Implementation Options
    - Small soft scalar processor
    - or Larger vector processor
    - or Customized hardware pipeline
  - Each with local memory
- Each processor can exploit the fine grained parallelism of the FPGA to more efficiently implement it's "program"
- Possibly heterogeneous
  - Optimized for different tasks
- Customizable to suit the needs of a particular application

FPGA as a computing accelerator



https://builders.intel.com/blog/fpga-in-the-data-center-programming-for-all/

The Intel Programmable Acceleration Card

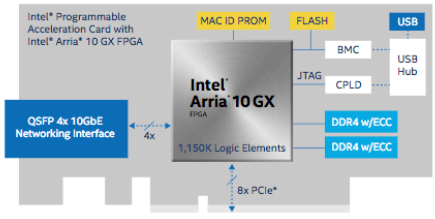
PRODUCT BRIEF

intel

Intel® Programmable Acceleration Card (Intel® PAC) with Intel® Arria® 10 GX FPGAs

Introduction

This PCIe-based FPGA acceleration card for data centers offers both inline and lookaside acceleration. It provides the performance and versatility of FPGA acceleration and is one of several platforms supported by the Acceleration Stack for Intel® Xeon® CPUs with FPGAs. This acceleration stack provides a common developer interface for both application and accelerator function developers, and includes drivers, application programming interfaces (APIs), and an FPGA interface manager. Together with acceleration libraries and development tools, the acceleration stack saves developer's time and enables code re-use across multiple Intel FPGA platforms. The card can be deployed in a variety of servers with its low-profile form factor, low-power dissipation, and passive heat sink.

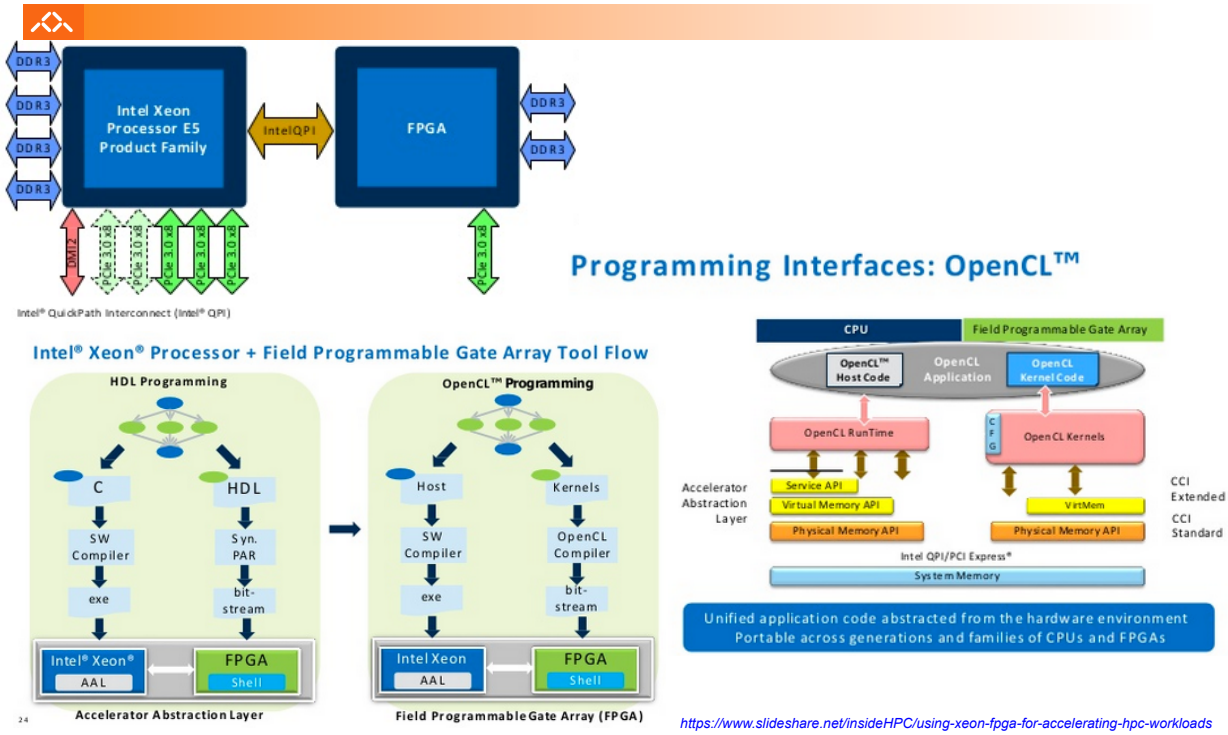


Targeted Workloads

- Big data analytics
- Artificial intelligence
- Video transcoding
- Cyber security
- High-performance computing (HPC), such as genomics and oil and gas
- Financial technology, or FinTech

https://builders.intel.com/blog/fpga-in-the-data-center-programming-for-all/

Faster integration of programmable acceleration cards at Intel



https://www.slideshare.net/insideHPC/using-xeon-fpga-for-accelerating-hpc-workloads

Beyond Vector/SIMD architectures

- Vector/SIMD-extended architectures are hybrid approaches
  - mix (super)**scalar + vector** op capabilities on a single device
  - **highly pipelined** approach to reduce memory access penalty
  - **tightly-closed access to shared memory**: lower latency
- Evolution of Vector/SIMD-extended architectures
  - **PU (Processing Unit) cores with wider vector units**
    - x86 many-core: **Intel** MIC / Xeon KNL (*more slides later*)
    - other many-core: **IBM** BlueGene/Q Compute, **ShenWay** 260
  - **coprocessors (require a host scalar processor): accelerator devices**
    - on disjoint physical memories (e.g., **Xeon KNC** with PCI-Expr, **PEZY-SC**)
    - ISA-free architectures, code compiled to silica: **FPGA**
    - focus on SIMT/SIMD to hide memory latency: **GPU**-type approach
    - …
  - **heterogeneous PUs in a SoC: multicore PUs with GPU-cores**
    - …

Graphical Processing Units

- Question to GPU architects:
  - *Given the hardware invested to do graphics well, how can we supplement it to improve the performance of a wider range of applications?*
- Key ideas:
  - Heterogeneous execution model
    - CPU is the *host*, GPU is the *device*
  - Develop a C-like programming language for GPU
  - Unify all forms of GPU parallelism as *CUDA_threads*
  - Programming model follows SIMT: *"Single Instruction Multiple Thread"*

cores/processing elements in several devices

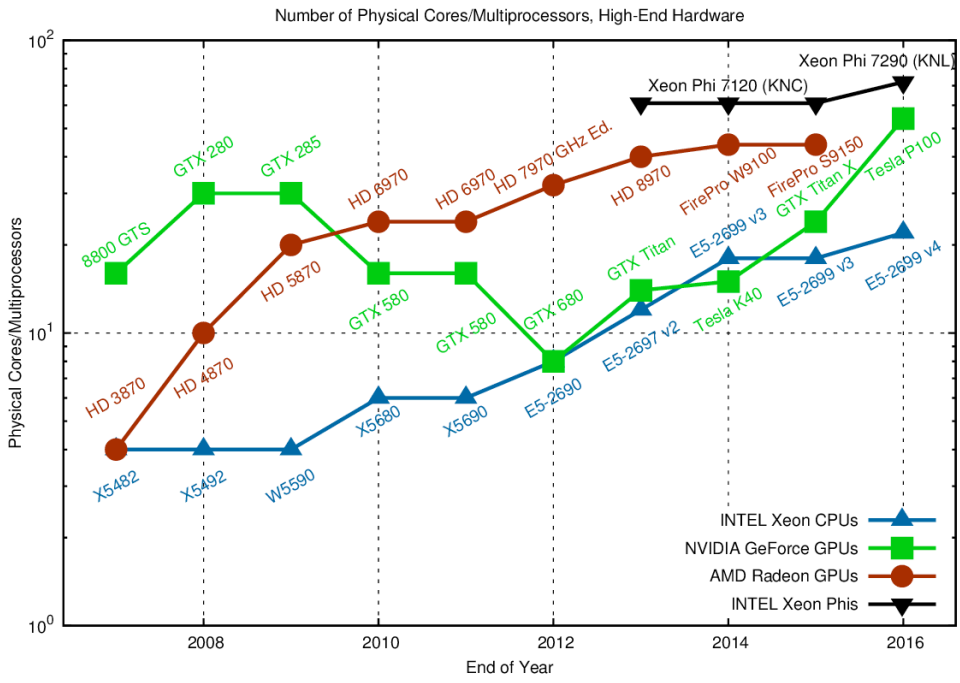
Key question: what is a **core**?

a) IU+FPU? *GPU-type...*

b) A SIMD processor? *CPU-type..*

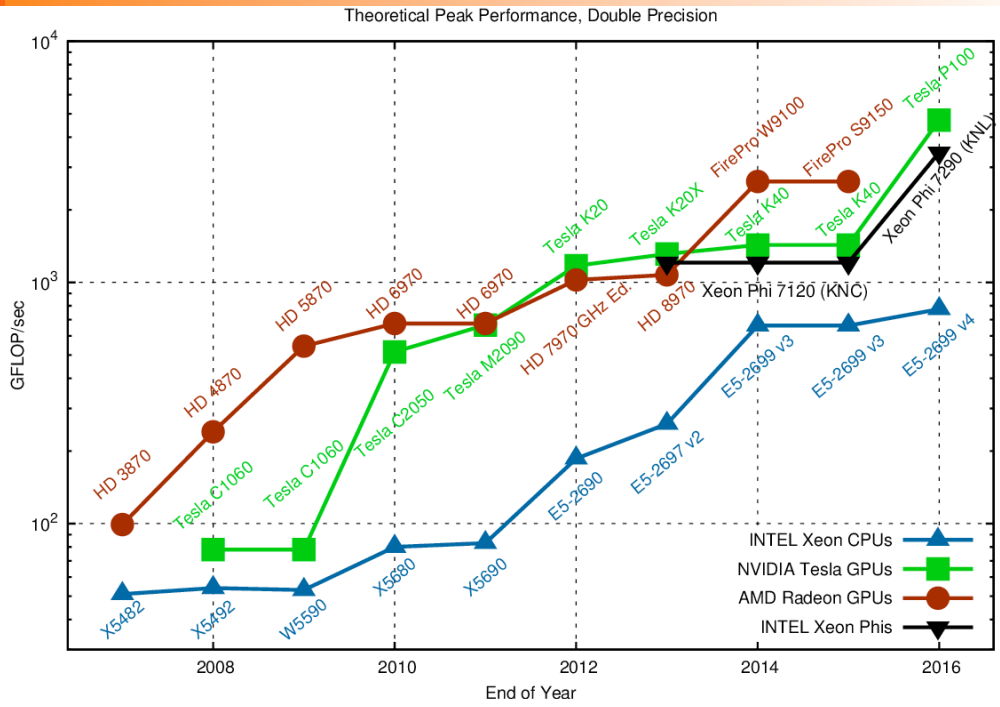
This updated slide and in this course: - **b)**

Note: the web link with these plots was updated in Aug'16



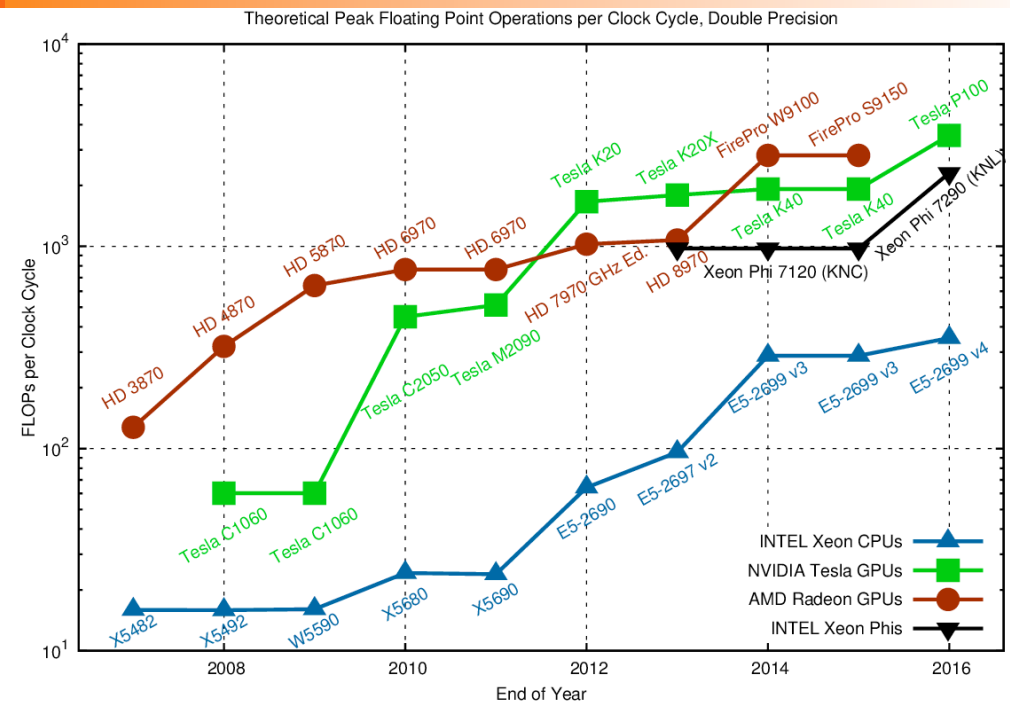
http://www.karlrupp.net/2013/06/cpu-gpu-and-mic-hardware-characteristics-over-time/

Theoretical peak performance in several computing devices (DP)



http://www.karlrupp.net/2013/06/cpu-gpu-and-mic-hardware-characteristics-over-time/

Theoretical peak FP Op's per clock cycle in several computing devices (DP)



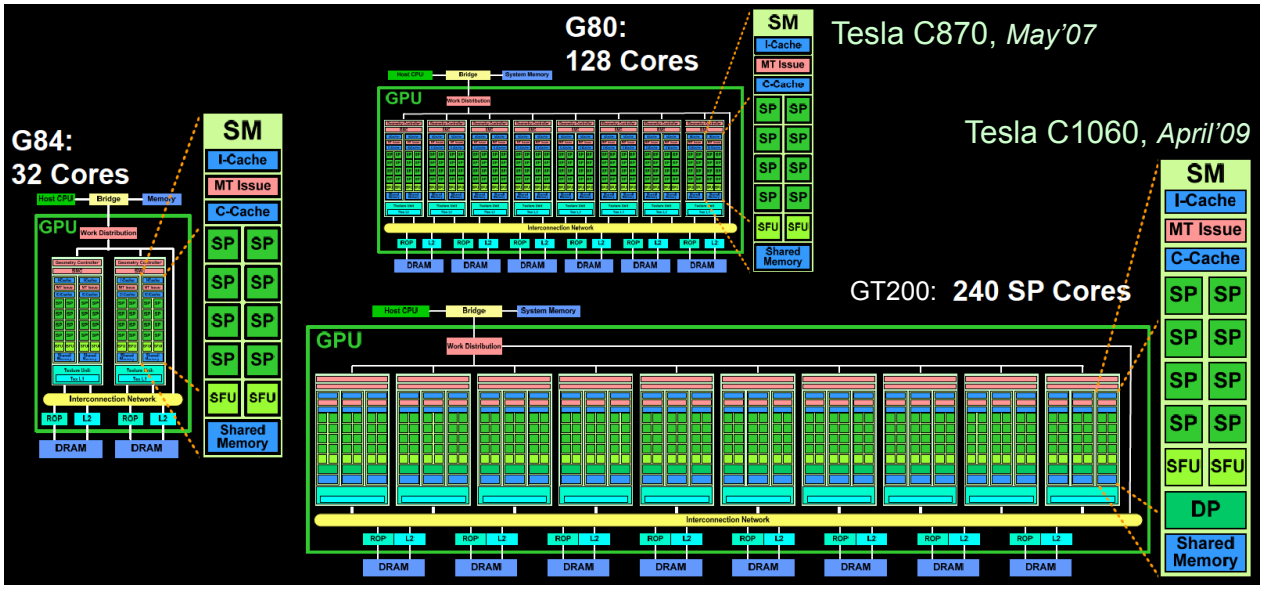
http://www.karlrupp.net/2013/06/cpu-gpu-and-mic-hardware-characteristics-over-time/

NVIDIA GPU Architecture

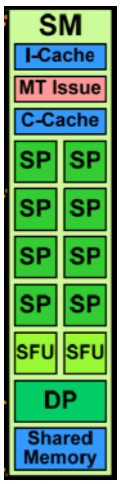
- Similarities to vector machines:
  - Works well with data-level parallel problems
  - Scatter-gather transfers
  - Mask registers
  - Large register files

- Differences:
  - No scalar processor
  - Uses multithreading to hide memory latency
  - Has many functional units, as opposed to a few deeply pipelined units like a vector processor

Early NVidia GPU Computing Modules



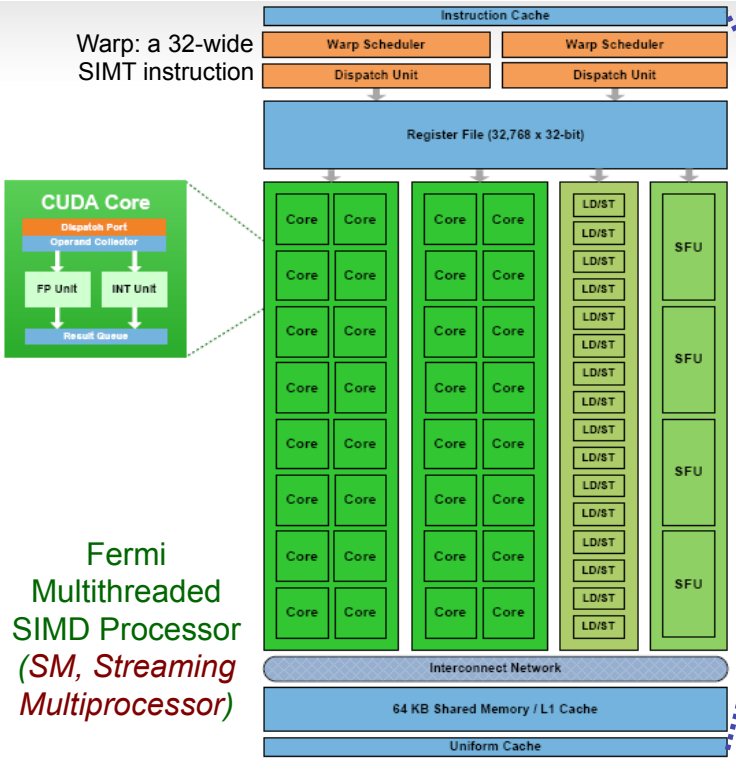
NVIDIA GPU Memory Structures



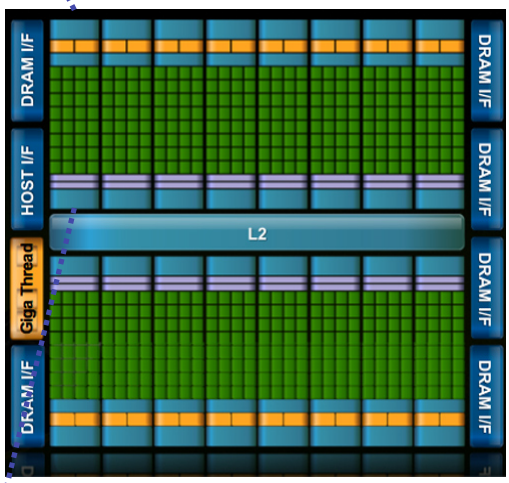
- Each SIMD Lane has private section of **off-chip DRAM**
  - "Private memory" *(Local Memory)*
  - Contains stack frame, spilling registers, and private variables
- Each multithreaded SIMD processor *(SM)* also has local memory *(Shared Memory)*
  - Shared by SIMD lanes / threads within a block
- Memory shared by SIMD processors *(SM)* is GPU Memory, off-chip DRAM *(Global Memory)*
  - Host can read and write GPU memory

The NVidia Fermi architecture

Warp: a 32-wide SIMT instruction



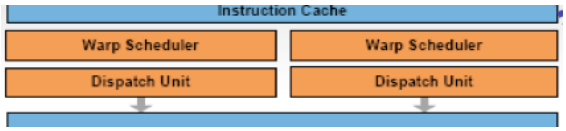
Fermi Multithreaded SIMD Processor *(SM, Streaming Multiprocessor)*



Fermi Architecture: GF110: 512 CUDA-cores *July'11*

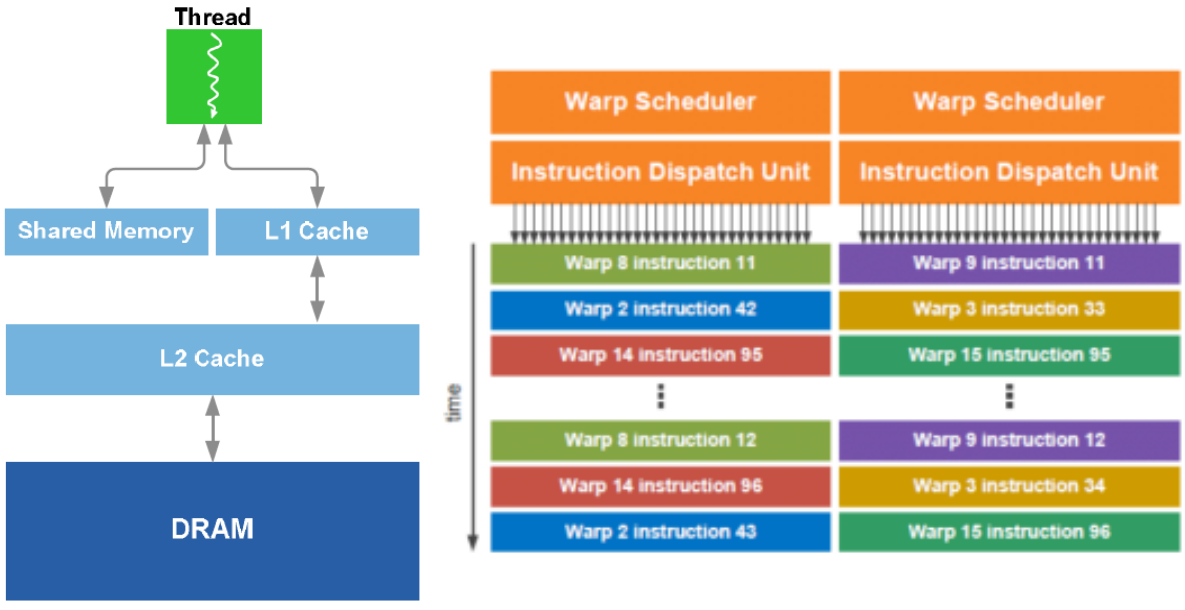
Fermi Architecture Innovations

- Each SIMD processor has
  - Two SIMD thread schedulers, two instruction dispatch units
  - 16 SIMD lanes (SIMD width=32, chime=2 cycles), 16 load-store units, 4 special function units
  - Thus, two threads of SIMD instructions are scheduled every two clock cycles



- Fast double precision
- Caches for GPU memory (16/64KB_L1/SM and global 768KB_L2)
- 64-bit addressing and unified address space
- Error correcting codes
- Faster context switching
- Faster atomic instructions

Fermi: Multithreading and Memory Hierarchy



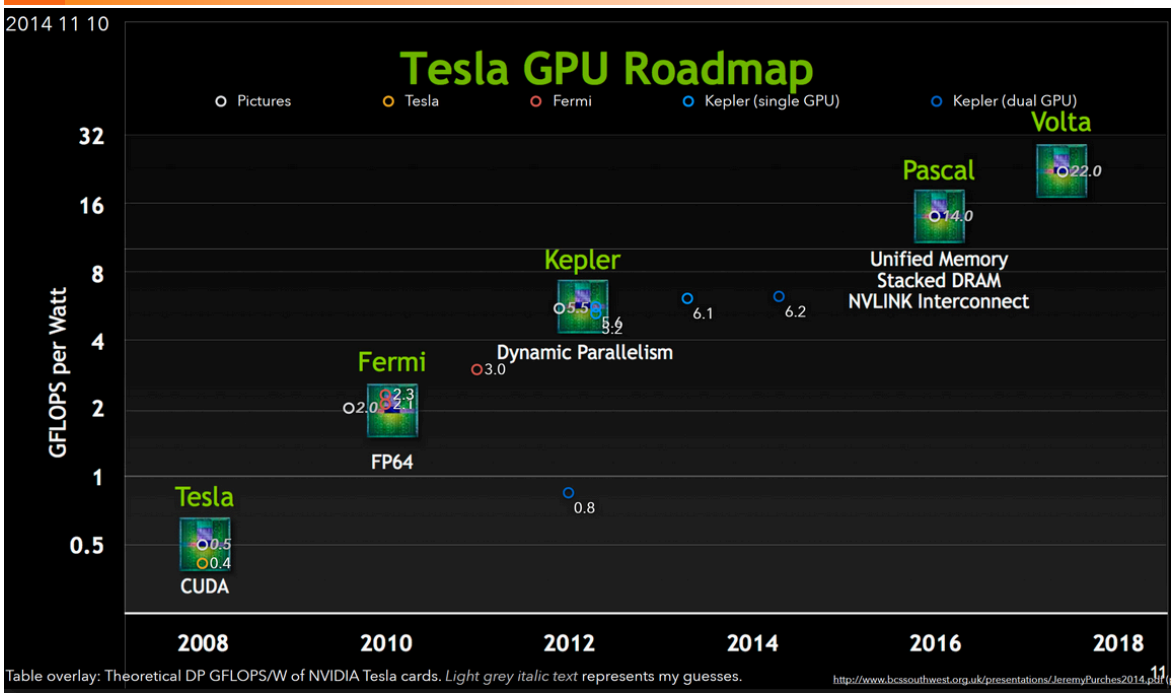
TOP500 list in November 2010: 3 systems in the top4 use Fermi GPUs

TOP 500
The List.

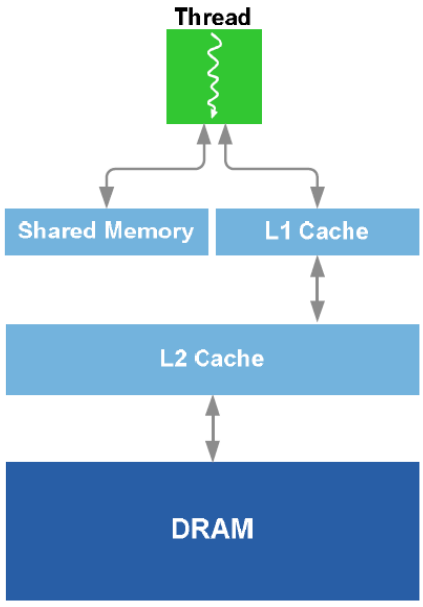
HIGHLIGHTS: NOVEMBER 2010

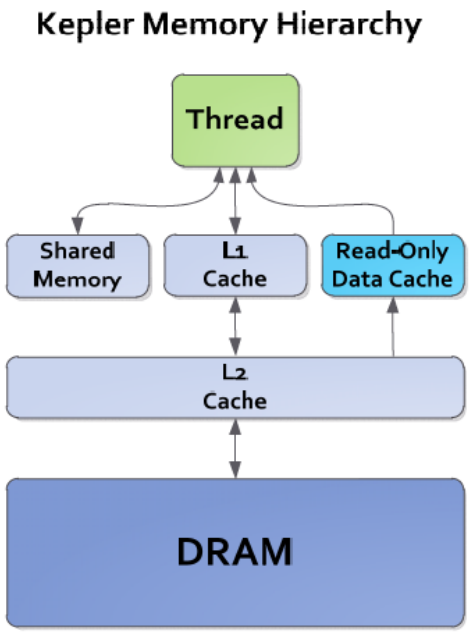
- The Chinese Tianhe-1A system is the new No. 1 on the TOP500 and clearly in the lead with 2.57 petaflop/s performance.
- No. 3 is also a Chinese system called Nebulae, built from a Dawning TC3600 Blade system with Intel X5650 processors and NVIDIA Tesla C2050 GPUs
- There are seven petaflop/s systems in the TOP10
- The U.S. is tops in petaflop/s with three systems performing at the petaflop/s level
- The two Chinese systems and the new Japanese Tsubame 2.0 system at No. 4 are all using NVIDIA GPUs to accelerate computation and a total of 28 systems on the list are using GPU technology.

Families in NVidia Tesla GPUs

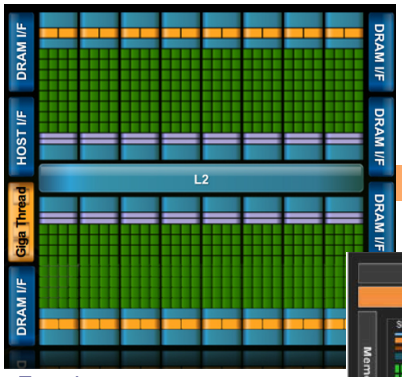


From Fermi into Kepler: The Memory Hierarchy





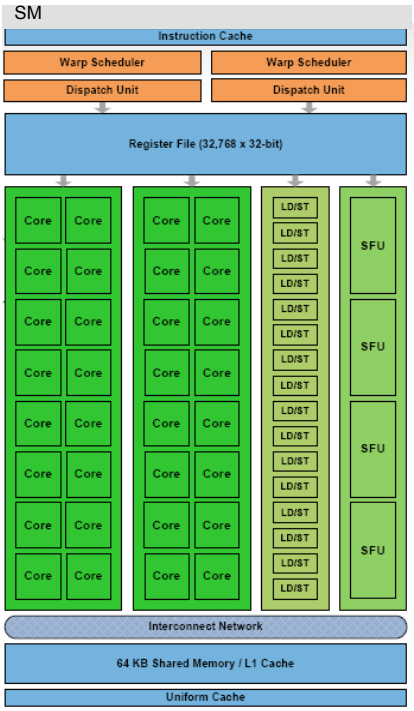
From the GF110 to the GK110 Kepler Architecture



Fermi:
512 CUDA-cores
July'11

Kepler:
2880 CUDA-cores
October'13

From Fermi to Kepler core: SM and the SMX Architecture

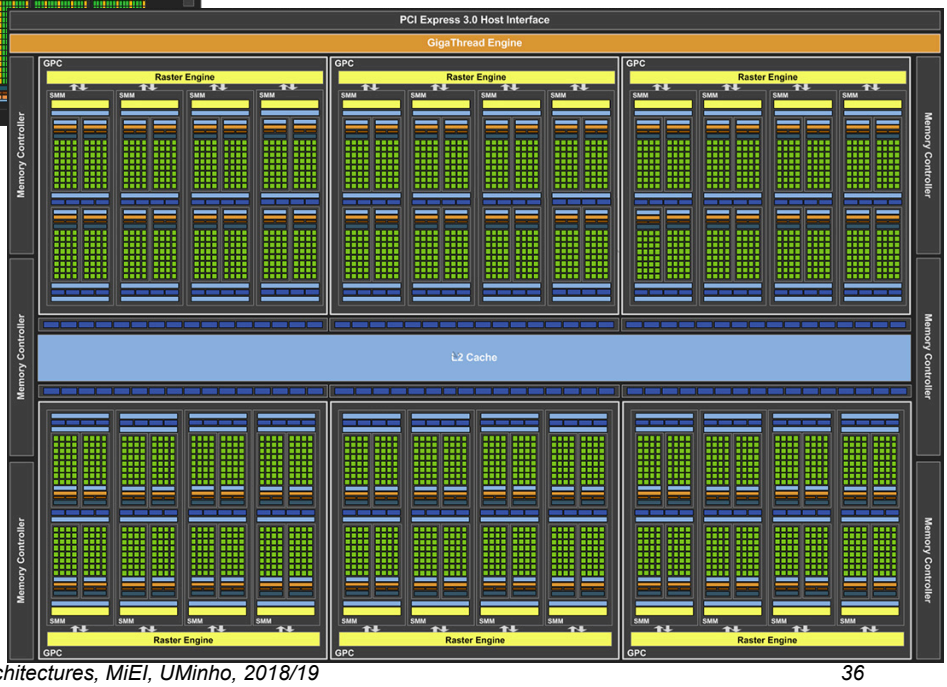


SMX:
192 CUDA-cores

Ratio **DP**unit : **SP**unit —> **1 : 3**

From the GK110 to the GM200 Maxwell Architecture

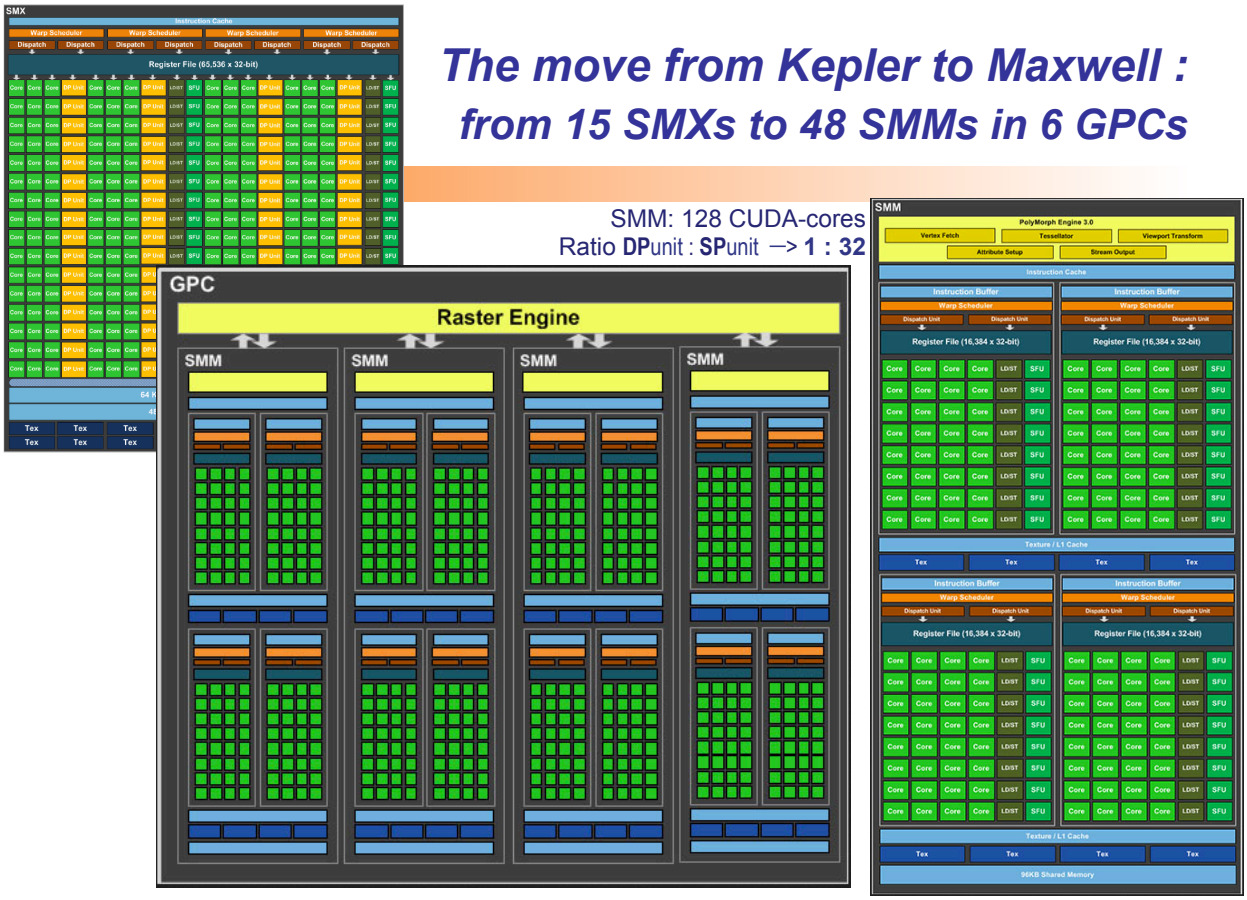
Kepler:
2880 CUDA-cores
October'13



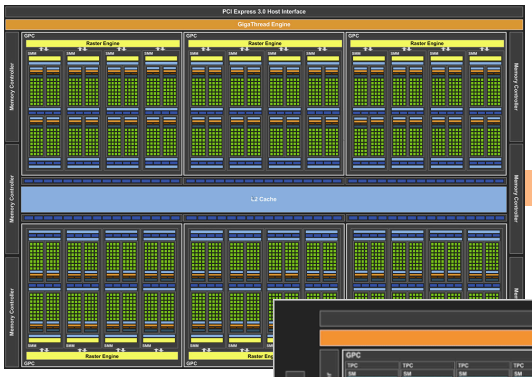
Maxwell:
3072 CUDA-cores
November'15

The move from Kepler to Maxwell : from 15 SMXs to 48 SMMs in 6 GPCs

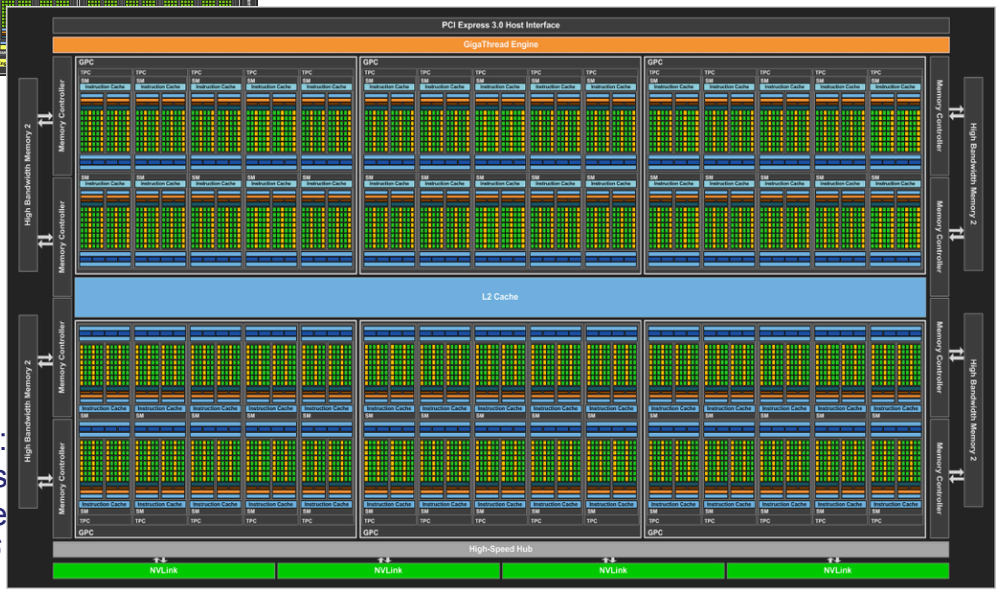
SMM: 128 CUDA-cores
Ratio **DP**unit : **SP**unit –> **1 : 32**



From the M200 to the GP100 Pascal Architecture

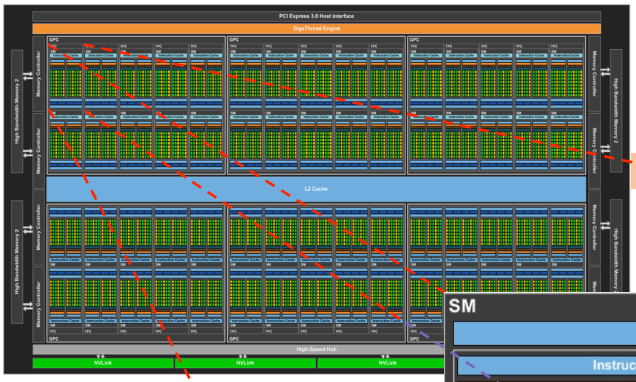


Maxwell:
3072 CUDA-cores
November'15



Pascal:
3584 CUDA-cores
HBM on-package
September'16

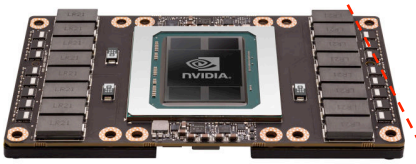
Pascal Architecture: 6x GPCs, 60 SMs



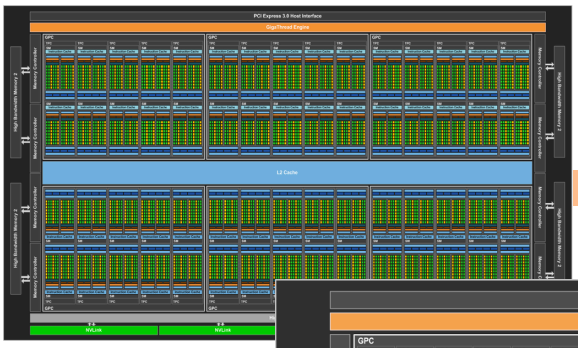
Pascal SM:
64 CUDA-cores

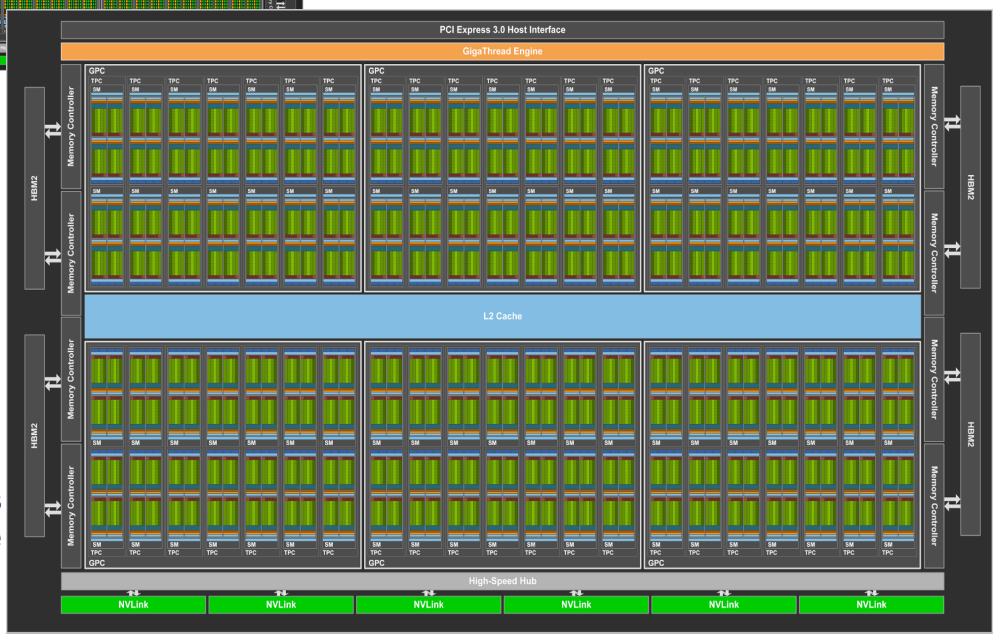
Ratio **DP**unit : **SP**unit —> **1 : 2**

Pascal P100 w/ 16GB HBM2



From the GP100 to the GV100 Volta Architecture

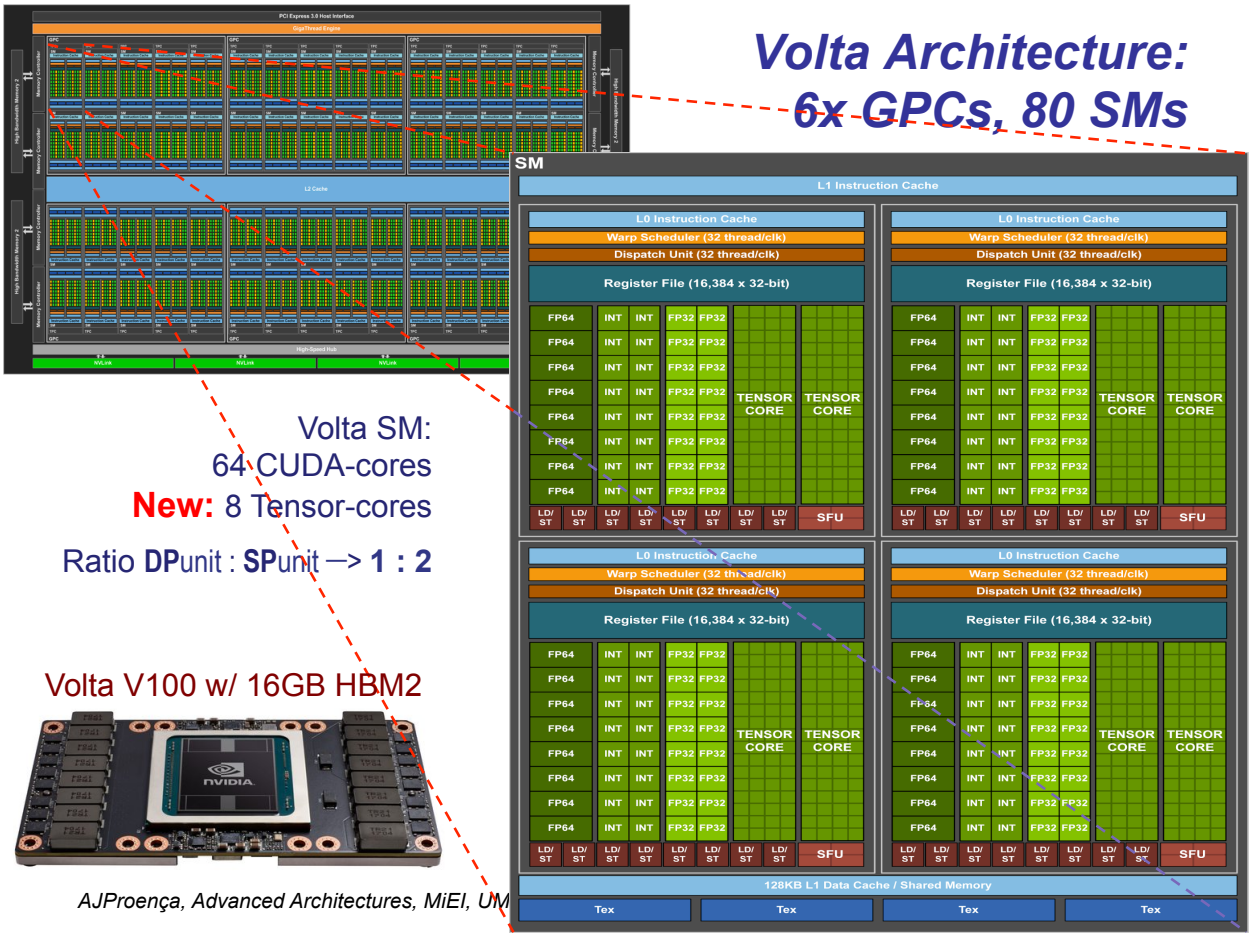




Pascal:
3584 CUDA-cores
November'15

Volta:
5120 CUDA-cores
HBM on-package
June'17

Volta Architecture: 6x GPCs, 80 SMs



Volta SM:
64 CUDA-cores
New: 8 Tensor-cores

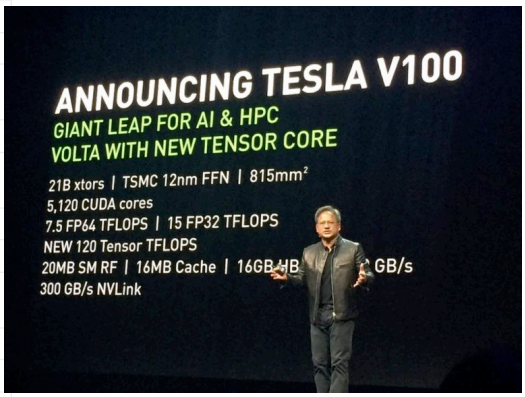
Ratio **DP**unit : **SP**unit —> **1 : 2**

Volta V100 w/ 16GB HBM2

AJProença, Advanced Architectures, MiEI, UM

Tesla accelerators: recent evolution

Tesla Product	Tesla K40	Tesla M40	Tesla P100	Tesla V100
GPU	GK180 (Kepler)	GM200 (Maxwell)	GP100 (Pascal)	GV100 (Volta)
SMs	15	24	56	80
TPCs	15	24	28	40
FP32 Cores / SM	192	128	64	64
FP32 Cores / GPU	2880	3072	3584	5120
FP64 Cores / SM	64	4	32	32
FP64 Cores / GPU	960	96	1792	2560
Tensor Cores / SM	NA	NA	NA	8
Tensor Cores / GPU	NA	NA	NA	640
GPU Boost Clock	810/875 MHz	1114 MHz	1480 MHz	1530 MHz
Peak FP32 TFLOP/s*	5.04	6.8	10.6	15.7
Peak FP64 TFLOP/s*	1.68	.21	5.3	7.8
Peak Tensor Core TFLOP/s*	NA	NA	NA	125
Texture Units	240	192	224	320
Memory Interface	384-bit GDDR5	384-bit GDDR5	4096-bit HBM2	4096-bit HBM2
Memory Size	Up to 12 GB	Up to 24 GB	16 GB	16 GB
L2 Cache Size	1536 KB	3072 KB	4096 KB	6144 KB
Shared Memory Size / SM	16 KB/32 KB/48 KB	96 KB	64 KB	Configurable up to 96 KB
Register File Size / SM	256 KB	256 KB	256 KB	256KB
Register File Size / GPU	3840 KB	6144 KB	14336 KB	20480 KB
TDP	235 Watts	250 Watts	300 Watts	300 Watts
Transistors	7.1 billion	8 billion	15.3 billion	21.1 billion
GPU Die Size	551 mm²	601 mm²	610 mm²	815 mm²
Manufacturing Process	28 nm	28 nm	16 nm FinFET+	12 nm FFN



https://devblogs.nvidia.com/parallelforall/inside-volta/