NVIDIA® Tesla®

Digital Signal Processor, AAU Klagenfurt

Gernot Rischner | Udo Rußegger

What is NVIDIA Tesla?

Nvidia Tesla is Nvidia's brand name for their products targeting stream processing and general purpose GPU (GPGPU). GPGPU computing is the use of a graphics processing unit (GPU) together with a CPU to accelerate deep learning, analytics, and engineering applications.

Reference: [3]

How GPUs accelerate software application?

GPU-accelerated computing offloads compute-intensive portions of the application to the GPU, while the remainder of the code still runs on the CPU. From a user's perspective, applications simply run much faster.

Reference: [3]

GPU Applications

Today, hundreds of applications are already GPU-accelerated and the number is growing.

Reference: [4]

GPU Hardware Roadmap

At GTC 2015 NVIDIA confirmed that MAXWELL will be succeeded by PASCAL architecture in 2016. On 4th April NVIDIA released first GPU codenamed GP100, which was followed by Pascal GP104, GP106 and in July by GP102.

In 2018 NVIDIA will release new a architecture codenamed VOLTA, which will bring twice as power efficient chips as PASCAL and even higher bandwidth memory with capacities of 64 GB.

Reference: [5]

The Tesla architecture

(NOTE: If not mentioned otherwise, this section is based on the paper „*NVIDIA TESLA: A UNIFIED GRAPHICS AND COMPUTING ARCHITECTURE", see references [1].)*

The Tesla architecture is based on a scalable processor array.[1] The following picture shows the architecture of a Tesla GPU chip, here the G80.

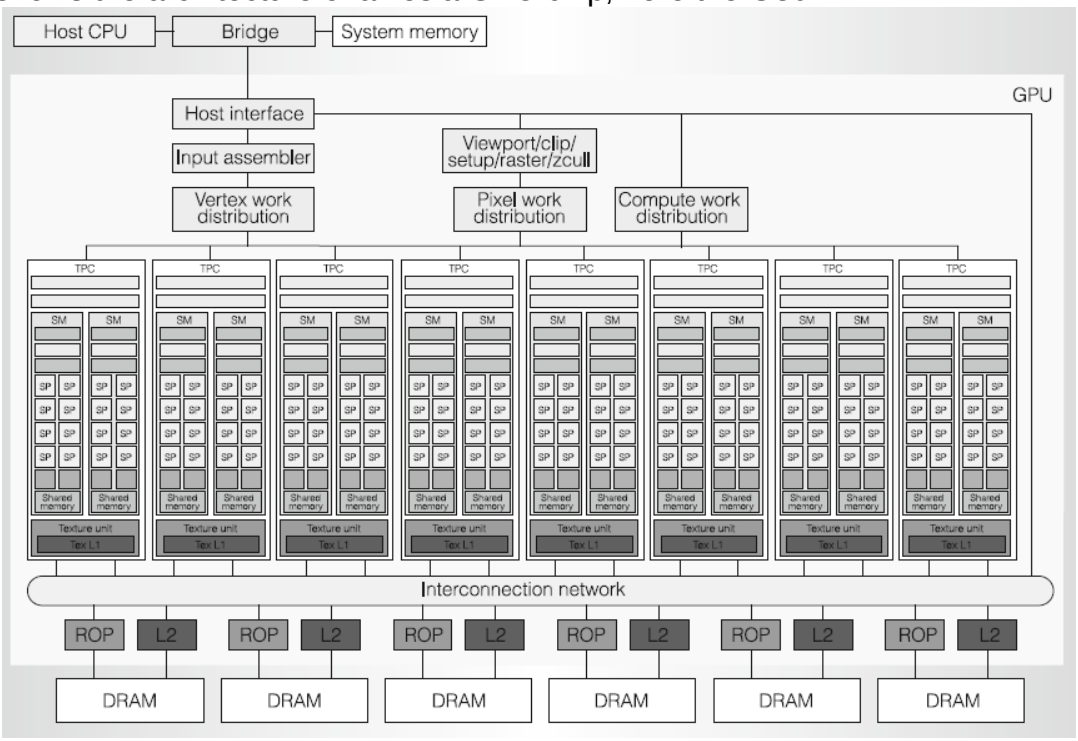

Figure 1. Tesla architecture; Chip: G80; Reference: [1]

As seen in figure 1, this architecture consists of the following components, which will be explained shortly.

NOTE: Numbers of components can vary for other chips. Here, specifications from the G80 were used, but the general principle is the same.

Here, the G80 has a total of 128 streaming processors, 8 per each streaming multiprocessor, which leads to 16 streaming multiprocessors. The streaming multiprocessors are furthermore organized in eight independent texture/processor clusters (each TPC consists of 2 SMs).

- **Host interface**: Communication between the host and the GPU (here: PCIe bus).

- **Interconnection network**: Carries computed pixel-fragment colors and depth values from the streaming processor array(SPA) to the raster operations processors(ROP).[1] Also routes texture memory read requests from the SPA to DRAM and read data from DRAM through a level-2 cache back to the SPA.[1]

- **Input assembler**: Collects vertex work from the input.

- **Vertex work distribution**: Distribute the vertex work packets from the input assembler to the TPCs. These work packets are then executed there. The resulting output is then written to on-chip buffers.
- **Viewport/clip/setup/raster/zcull**: Rasterize the results from the on-chip buffers (see 'vertex work distribution') into pixel fragments.

- **Pixel work distribution**: Distribute the pixel fragments from the 'Viewport/clip/setup/raster/zcull' to TPCs for pixel-fragment processing.

- **Compute work distribution**: Dispatches compute thread arrays to the TPCs.

- **Streaming processor – SP**: Primary thread processor in the streaming multiprocessor (SM).

  Performs fundamental floating-point operations:
  - add
  - multiply
  - multiply-add

  Also performs varieties of integer, comparison and conversion operations.

- **Geometry Controller:** Map the logical graphics vertex pipeline into recirculation on the physical SMs. Manage dedicated on-chip input and output vertex attribute storage and forwards contents as required.[1]

- **Streaming multiprocessor – SM:** A unified graphics and computing multiprocessor.

  Execute:
  - Vertex programs
  - Geometry programs
  - Pixel-fragment shader programs
  - Parallel computing programs

  Each SM consists of:
  - 8 SPs
  - 2 Special Function Units (**SFU**)
  - 1 Multithreaded instruction fetch and issue unit (**MT issue)**
  - 1 Instruction cache (**I cache)**

- 1 Read-Only constant cache(**C cache**)
- A read/write shared memory

Uses **Singe-instruction, multiple-thread – SIMT** scheduling.

- **Texture/processor cluster – TPC**

  Consists of: 1 Geometry controller
  1 SM controller – **SMC**
  2 SMs
  1 Texture unit.

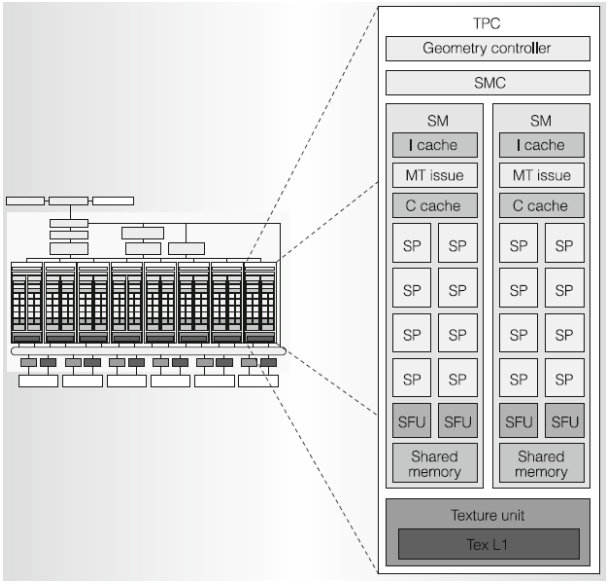

Figure 2. TCP; Reference: [1]

Figure 2 shows a more detailed look of a TPC.

- **Streaming processor array – SPA:** Performs all the GPU's programmable calculations.

  The term **'SPA'** describes the "whole" array.

  Executes: - Graphics shader thread programs.
  - GPU computing programs.

  Provide: -Thread control
  -Thread management

- **Raster operation processor – ROP:** TPCs feed data to the ROPs via the interconnection network. ROPs handle depth and stencil testing and updates and color blending and updates.[1] Performs color and depth frame buffer operations directly on memory.

- **Texture unit:** Processes one group of four threads(vertex, geometry, pixel, compute) per cycle.[1]

Comparison: Then vs. now

Comparison of some specifications from the first Tesla chip, the G80, which was introduced in 2006 and the GP100 from 2016.

	G80	**GP100**
Transistors	681 million	15300 million
SP	128	3584
SM	16	56
TPC	8	28
Base Clock	1350 MHz	1328 MHz
Process	90-nm CMOS	16-nm FinFET
TDP	170 Watts	300 Watts
Processing power (Single-presicion FMA)	345.6 GFLOPs	9519-10690 GFLOPs

References:

[1] E. Lindholm, J. Nickolls, S. Oberman, J. Montrym; *„NVIDIA TESLA: A UNIFIED GRAPHICS AND COMPUTING ARCHITECTURE“;* NVIDIA, IEEE Compute Society 2008.

[2] NVIDIA, *“NVIDIA Tesla P100”,* Whitepaper, 2016

[3] NVIDIA, WHAT IS GPU-ACCELERATED COMPUTING? http://www.nvidia.com/object/what-is-gpu-computing.html

[4] http://on-demand.gputechconf.com/gtc/2015/webinar/deep-learning-course/intro-to-deep-learning.pdf

[5] http://videocardz.com/specials/roadmaps