УДК 004.9

Saladkou A, Verkhov A. Supervisor Alioshyna N.

Using GPUs for General Purpose Computing

Belarusian National Technical University
Minsk, Belarus

In the article we will discuss the GPGPU technology. The GPU is the Graphic Processor Unit. It is designed to solve graphics problems. Letters GP stand for General-Purpose. Putting it all together, GPGPU stands for General-Purpose Computing on Graphic Processor Unit.

In 2000, when the clock speed of CPUs grew, Intel predicted that by 2010 the clock speed of a single processor core would reach 10 gigahertz. We now know that these expectations were not met.

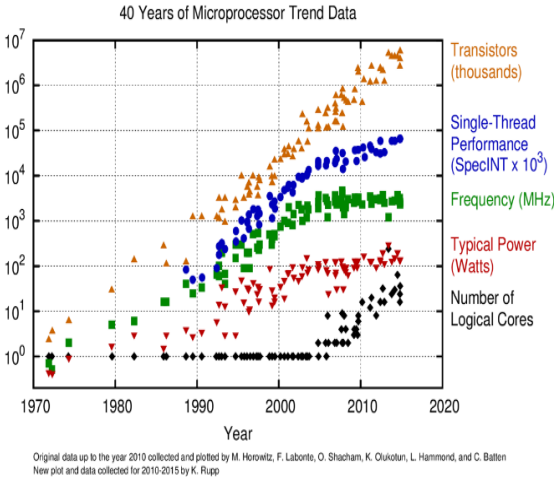

Fig. 1 – Processor growth chart

Fig. 1 shows that the growth was indeed very good, but somewhere around 2004-2005 it stopped abruptly. And then the processors even began to lose a little in power. The development of central processors eventually went by increasing the number of cores. Nowadays, in powerful computers, the CPU can have 16 cores. But the core clock speed hasn't increased over the last 10 years.

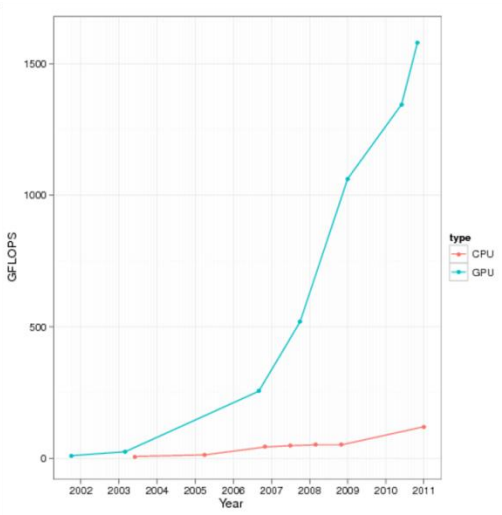

Fig. 2 – Comparative graph of GPU and CPU development

If you look at Fig. 2 that compares CPU and GPU, you can see that the processors in video cards have intercepted this trend. The graphics processor continued to increase its power and, in many ways, outperformed the central processors in terms of performance. It is worth saying that the processor in the video card performs much more floating-point operations per second than the main central processor.

The GPU is good for massive parallelism, where a big number of very similar computations are running at the same

time. On one side of the scale, there is latency, on the other - FLOPS - the number of operations. For example, if the operation takes a second, it is very long. But if you can do 100 billion of these operations at the same time, you have 100 billion operations per second. For the CPU this is pretty bad, because we need everything to work in sequence. And for the GPU it is fine. The main thing is not the delay, but the result. The numbers in this example are of course greatly exaggerated, the real GPU parallelism is on the order of several thousand.

Examples of massively parallel tasks: ray tracing, bitcoin mining, neural network training.

Many scientists use the GPU in their tasks, because there is often a lot of scientific data, they are processed for a long time, and the GPU is sometimes useful here.

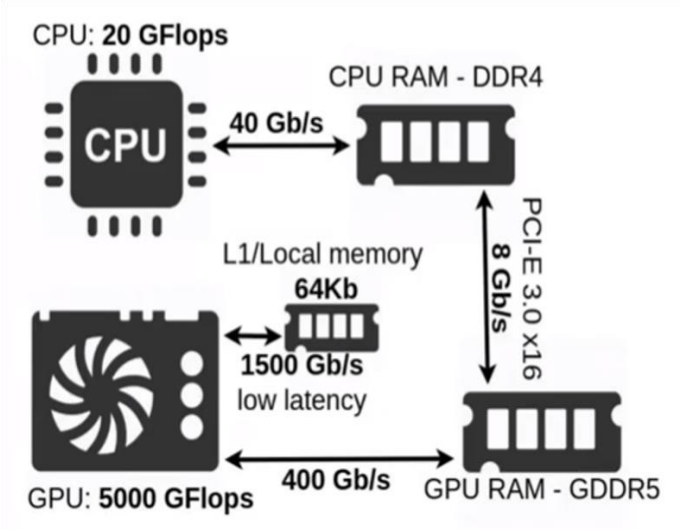

Fig. 3 – General PC architecture scheme

It can be seen that the CPU can have a performance of 20 gigaflops, and the GPU - 5000 gigaflops. But there is one

bottleneck - the PCI-E bus. It connects RAM and GRAM and passes only 8 Gb/s. Suppose the task is to add two arrays of numbers of 10 billion: the first is added to the first, the second to the second, and so on. You need to get the sum, the third set of numbers.

It is a task of massive parallelism. But there is one point. In order to add these numbers to the GPU, they must first be transferred to the GRAM over a narrow 8 Gb/s bridge. The video card will calculate everything very quickly. But then you will need to transfer this data across the bridge back. And it turns out that it's much faster to transfer over a larger channel to the CPU, and with its 20 gigaflops it will add the numbers faster than if they were sent over the bus to the GPU.

GPGPU technology did not appear quickly. Prior to this, the graphics adapter had been used exclusively for its intended purpose. But when people realized, how powerful GPUs are, they wanted to use that power for their non-graphical tasks. Then they had to get out and somehow disguise a non-graphic task as a graphic one. The graphics card thought it was drawing triangles, but it was actually calculating and processing scientific data. Luckily, graphics card developers have seen this problem and are moving forward. This is how OpenCL, CUDA and other technologies appeared.

References:

1. CPU and GPU trends over time [Electronic resource] / Mode of access: https://www.r-bloggers.com/2011/01/cpu-and-gpu-trends-over-time/. – Date of access: 23.04.2022.
2. Bases of GPU optimization [Electronic resource]. – Mode of access: http://my-it-notes.com/2013/06/bases-of-gpu-optimisation/. – Date of access: 23.04.2022.