**Report 4**

**Introduction to CUDA and OpenCL**

**SM, memory prefetching and nvprof.**

**Dariusz Biela**

**Karol Sawicki**

1. **Introduction**

At the time of our fourth lab classes we were trying to gain more performance with using Unified Memory. To do this we had to learn about memory prefetching mechanism.

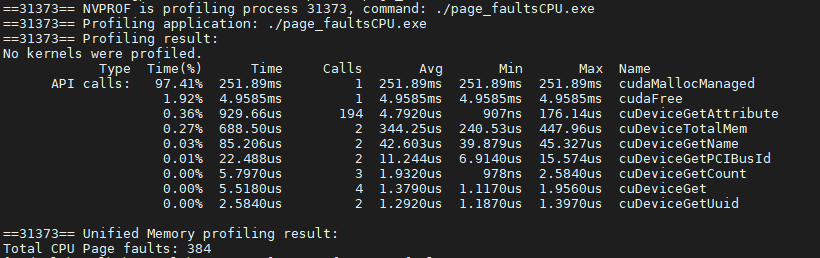
Memory prefetching is asynchronous mechanism that sends data from the device to the CPU before it needs the data. This allows to do operations quicker, because when CPU needs the data it already is available. Memory prefetching works in the opposite direction too.

Memory prefetching basically allows hardware to avoid page fault problems. Page fault is a type of hardware exception, that can be occurred when program tries to access memory that is not currently mapped.

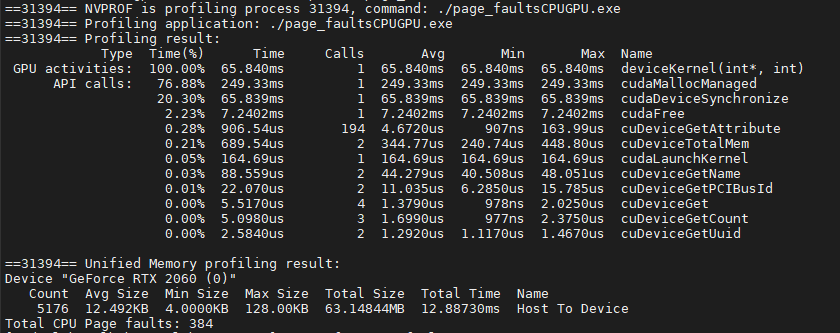
As an addition we learned about SM – Streaming Multiprocessors – every nvidia GPU chip is built of them.

1. **Calculations and measurements**

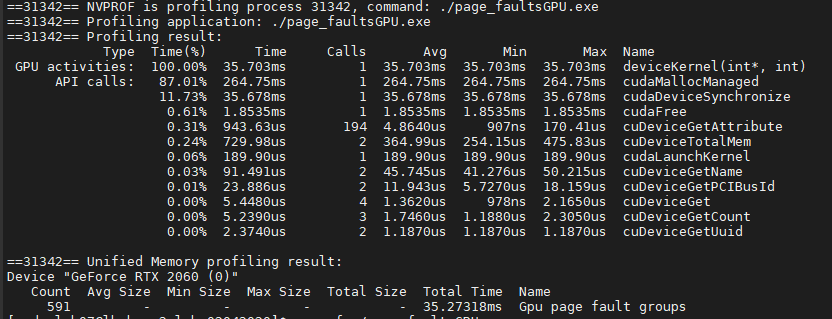
To start investigations we tried to use nvprof tool with the sample codes. We decided to show and discuss our results:



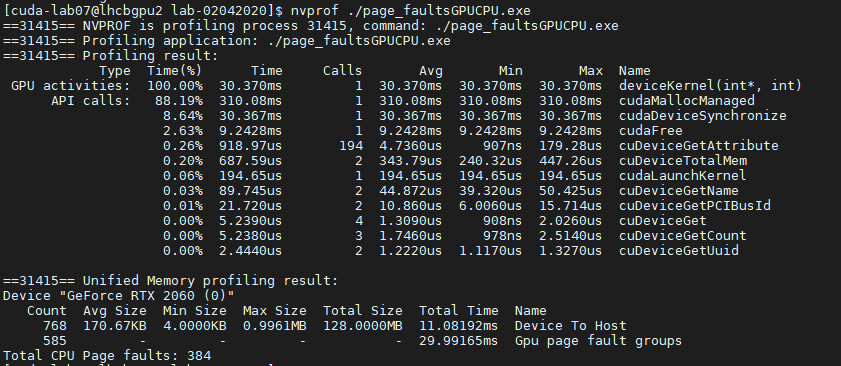
Picture 1 CPU only page faults



Picture 2 CPU to GPU page faults



Picture 3 GPU only page faults



Picture 4 GPU to CPU page faults

Pictures from *1* to *4* are describing nvprof results when exemplary code was profiled. We can see that CPU page faults are always the same – when they are. The more interesting are GPU page faults – in each case number of page fault groups is different and every time we see time that we are losing with these page faults. This lost time could be saved by good memory prefetching use.

1. **Conclusions**

Our nvprof tool shows us that page faults problems are real world problems and – when they are – we could lose so much time. The *Picture 3* shows that we lost *35.3 ms* – that may seem really small amount of lost time, but when we look at the kernel execution time - *35.7 ms* – we know that it is really problematic, especially when we have to execute kernel hundreds of thousands times.

Usage of memory prefetching can be helpful to avoid page faults, especially when we know when data is needed by the host or nvidia device that we are using. In the past, with older GPU chips every operation was ended with memory synchronization even if that was not necessary. Today we can use prefetching with unified memory only when we really have to synchronize our data and avoid page faults, that cause time loss.