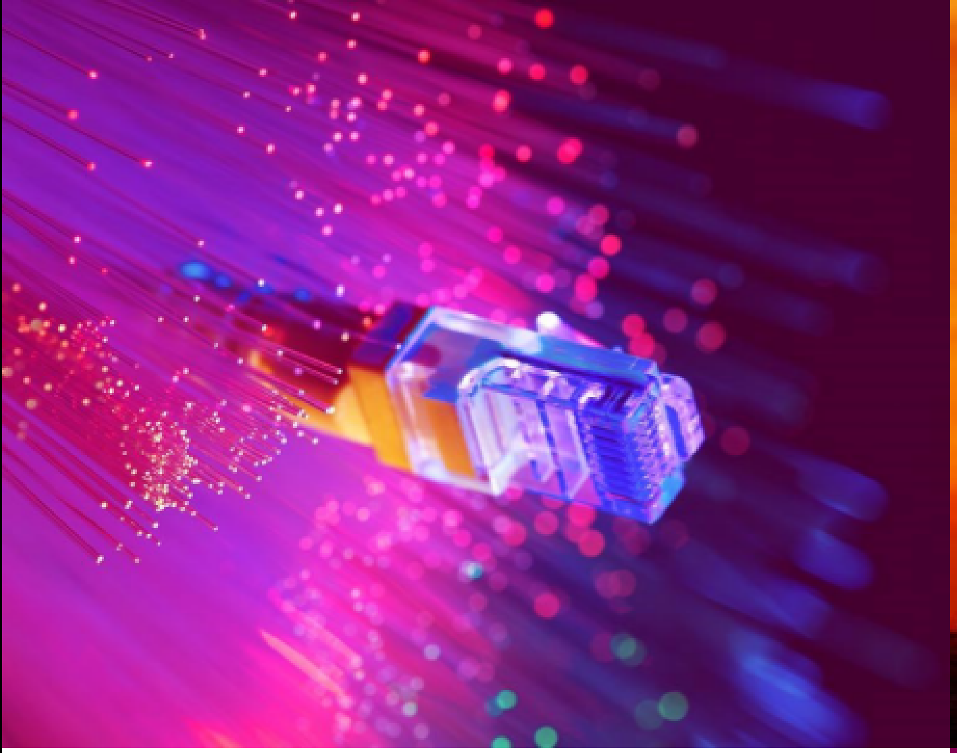
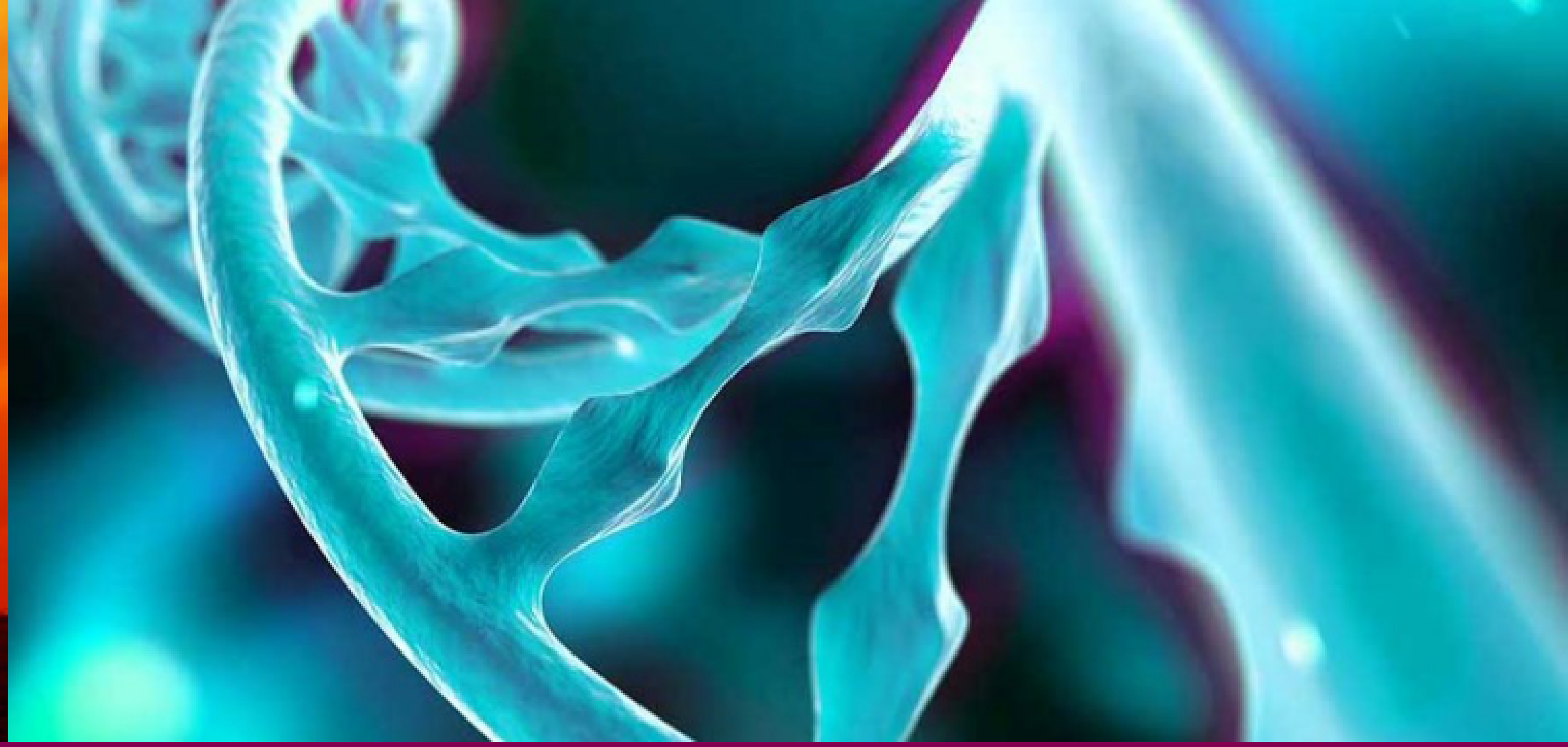




CSC

GPU Execution Model

High-Level GPU Programming

2024-02

CSC Training

CSC – Finnish expertise in ICT for research, education and public administration

GPU Execution Model

Heterogeneous Programming Model

- GPUs are co-processors to the CPU
- CPU controls the work flow:
    - *offloads* computations to GPU by launching *kernels*
    - allocates and deallocates the memory on GPUs
    - handles the data transfers between CPU and GPUs
- CPU and GPU can work concurrently
    - kernel launches are normally asynchronous

Example: axpy

Serial cpu code of `y=y+a*x`:

- have a loop going over the each index

```c
void axpy_(int n, double a, double *x, double *y)
{
    for(int id=0;id<n; id++) {
        y[id] += a * x[id];
    }
}
```

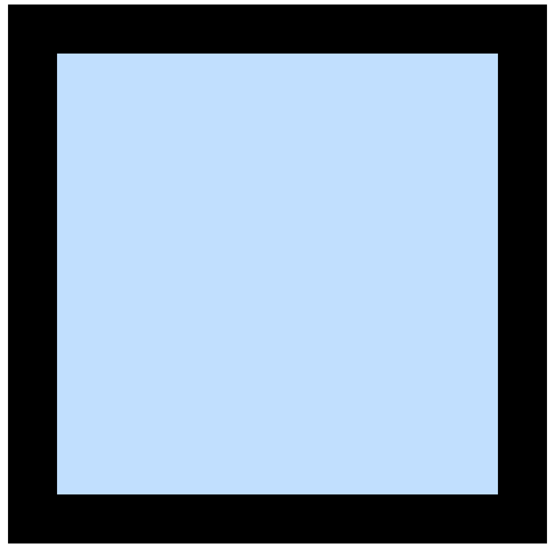
On an accelerator:

- no loop
- we create instances of the same function, **kernels**

```c
GPU_K void axpy_(int n, double a, double *x, double *y, int id)
{
        y[id] += a * x[id]; // id<n
}
```

Work-items

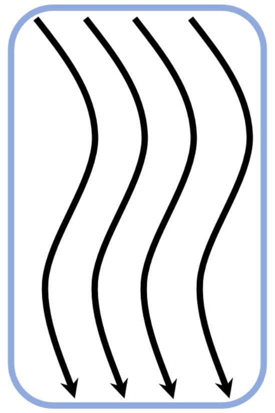
A work-item is running on a simd lane



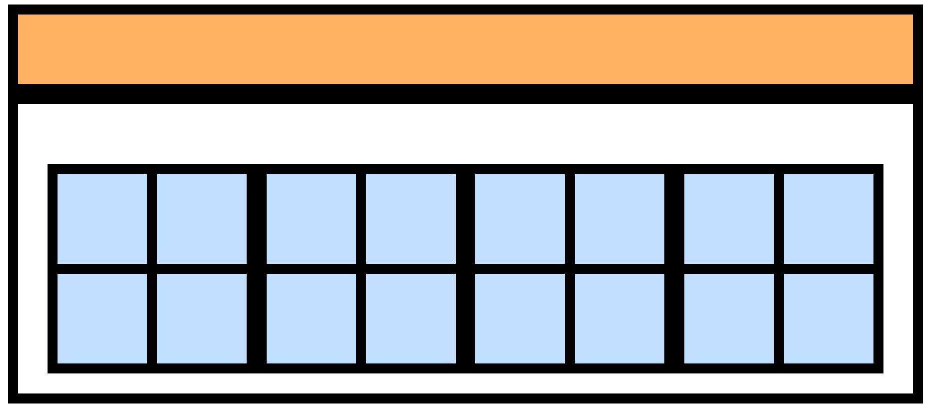
The smallest computational element in a GPU.

- the work-items are very light execution contexts.
- contain all information needed to execute a stream of instructions.
- for each work-item there is an instance of the **kernel**.
- each work-item processes different elements of the data (SIMD).

Sub-Group



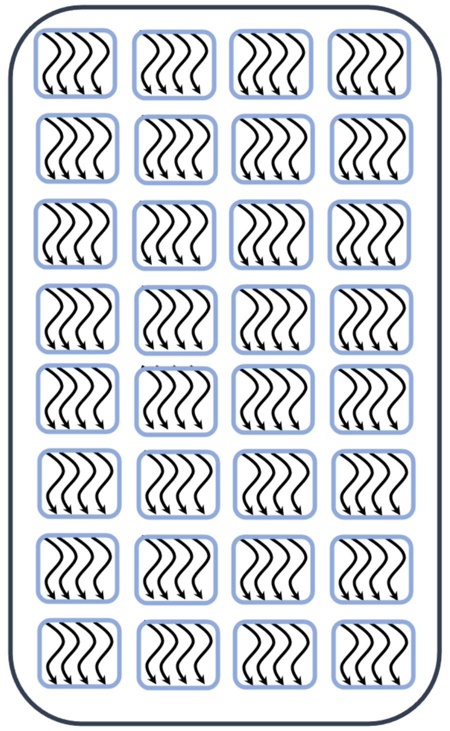
Execution is done per sub-groups.



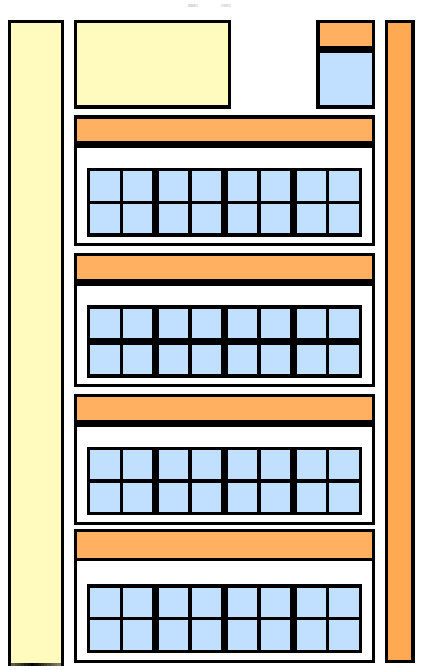
Scheme of a SIMD unit in an AMD GPU.

- the work-items are physically locked into sub-groups
- the size is locked by hardware, currently 64 for AMD and 32 for Nvidia.
- an instruction is executed by all items in the sub-group.
- in the case of branching, each branch has to be handled separetely.
- memory accesses are done per sub-group.

Work-Group



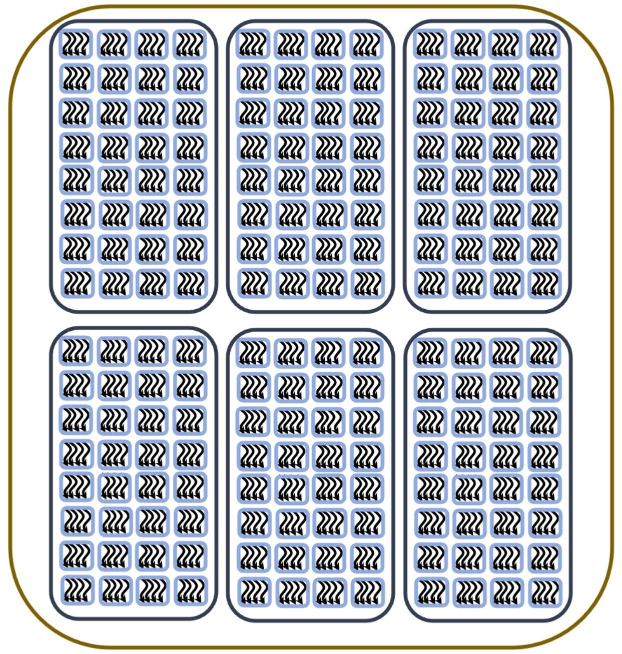
Work-groups of work-items.



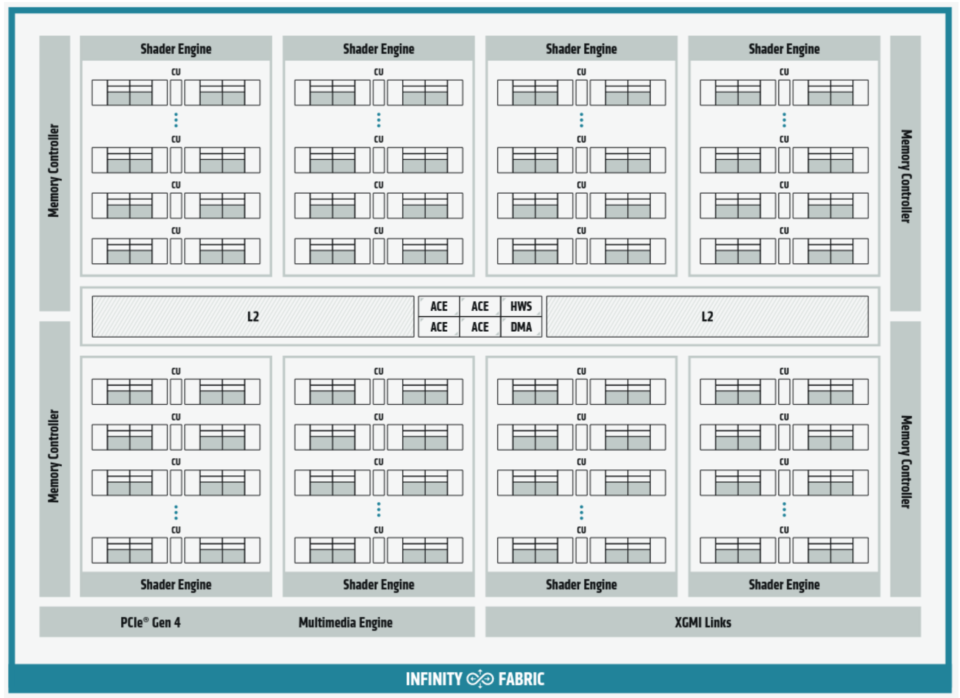
Compute Unit in an AMD GPU.

- the work-items are divided in groups of fixed size.
- size limited by hardware, 1024 for some GPUS or 8912 for some CPUs.
- each work-group is assign to a Compute Unite (2) and it can not be split.
- synchronization and data exchange is possible inside a group.

Grid of Work-Items



A grid of work-groups executing the same **kernel**



AMD Instinct MI100 architecture (source: AMD)

- a grid of threads is created on a specific device to perform the work.
- each work-item executes the same kernel
- each work-item typically processes different elements of the data.
- there is no global synchronization or data exchange.

Summary

- GPUs are hardware with high degree of parallelism.
- many threads execute the same instruction (SIMD).
- there is a hierarchy of the work-items (*work-groups*, *sub-groups*).
- all items in the sub-group execute the same instruction.
- branching in a *sub-group* should be avoided
- memory accesses are done per *sub-group*.