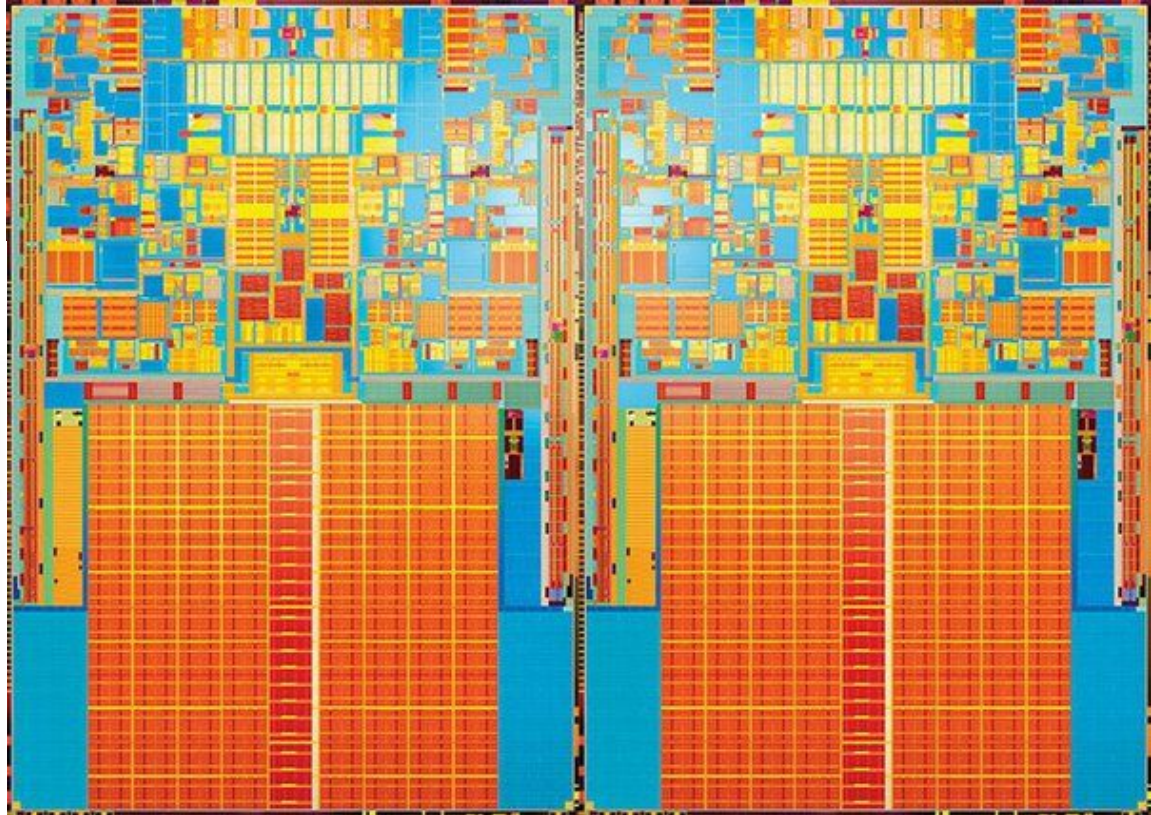
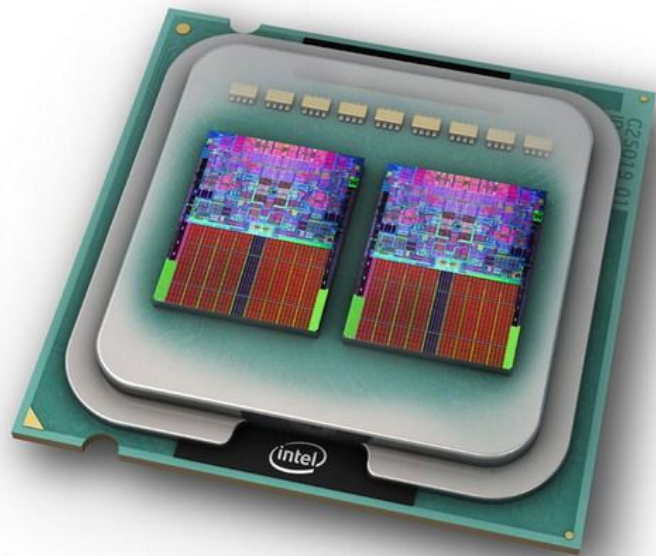


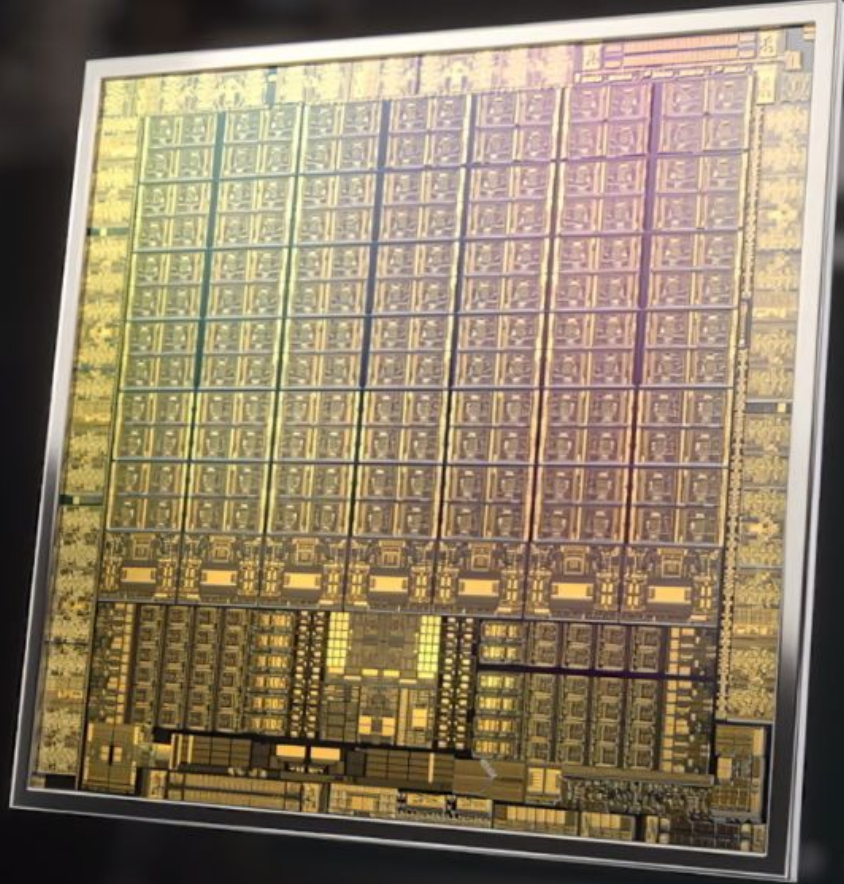
CUDA ядра

Царюк Артем ИТ-МО 21

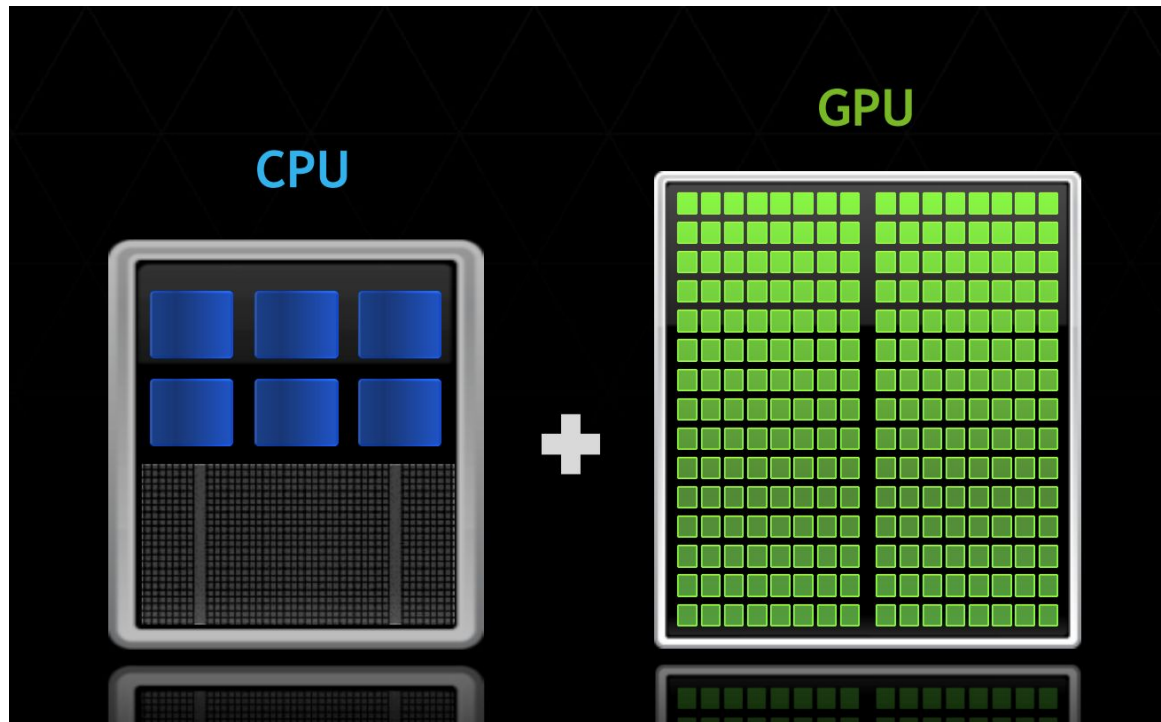
CPU



GPU

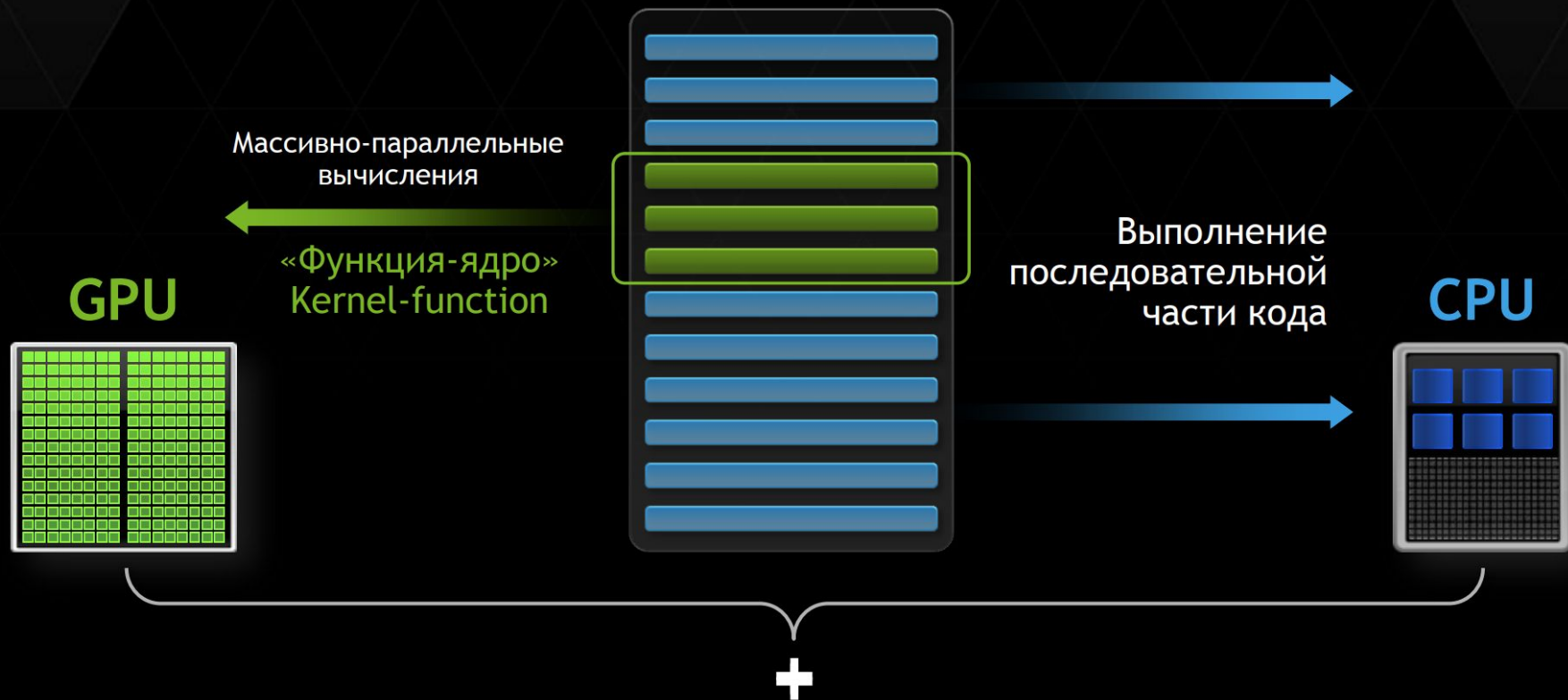


Гибридное решение



https://academy.hpc-russia.ru/files/lecture_1.pdf

Код программы



Код исполняемый на CUDA ядрах

```
5      #define N (4096*4096)
6      #define CORES (1024)
7
8      __global__ void kernel(float* dA) {
9          int idx = blockIdx.x * blockDim.x + threadIdx.x;
10         float x = 2.0f * 3.1415926f * (float) idx / (float) N;
11         dA[idx] = sinf(sqrtf(x));
12     }
```

CPU ВЫЗОВ

```
float *hA, *dA;  
hA = (float*) malloc(N * sizeof(float));  
cudaMalloc((void**) &dA, N * sizeof(float));  
kernel <<< N / CORES, CORES >>> (dA);  
cudaMemcpy(hA, dA, N * sizeof(float), cudaMemcpyDeviceToHost);
```

```
function<<<blocks, threads>>>(args...);
```

Сравнение скорости вычисления

```
PS C:\Users\func\Desktop\hodograph> .\main.exe
```

```
Renderer: NVIDIA GeForce GTX 1060 6GB/PCIe/SSE2
```

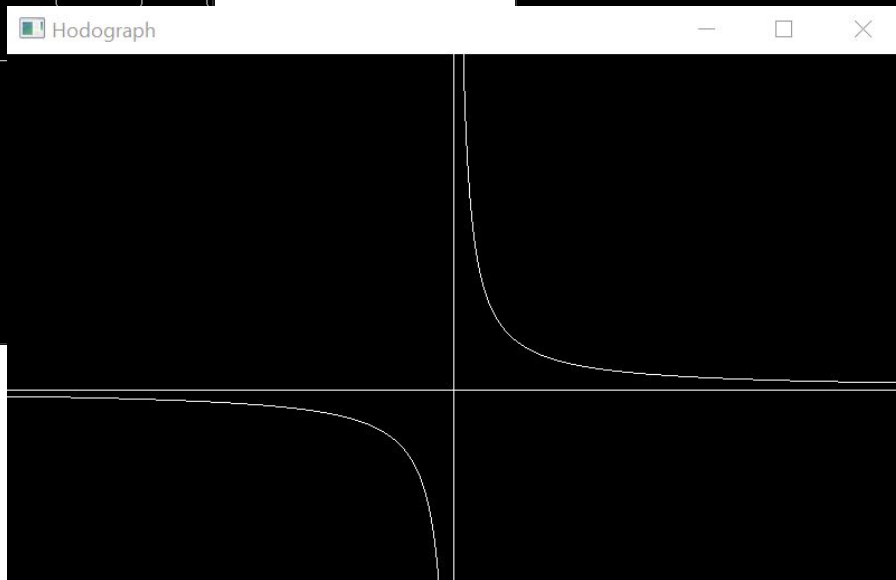
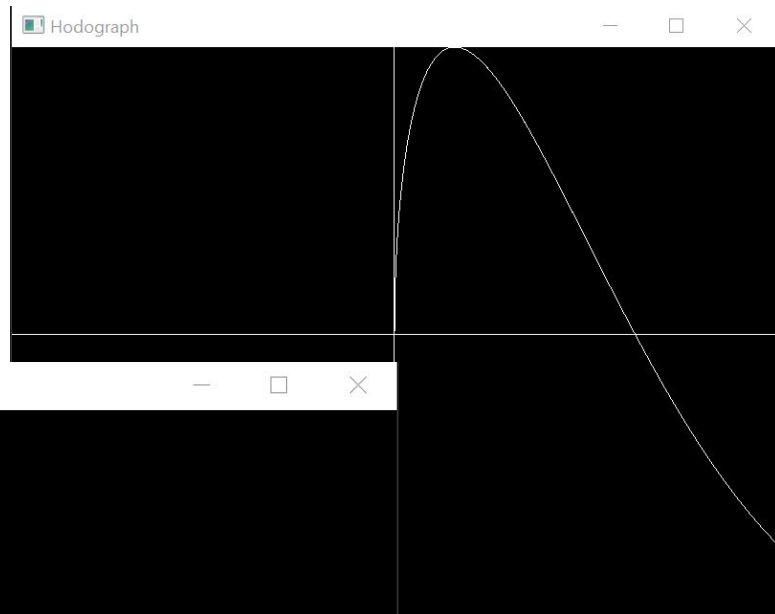
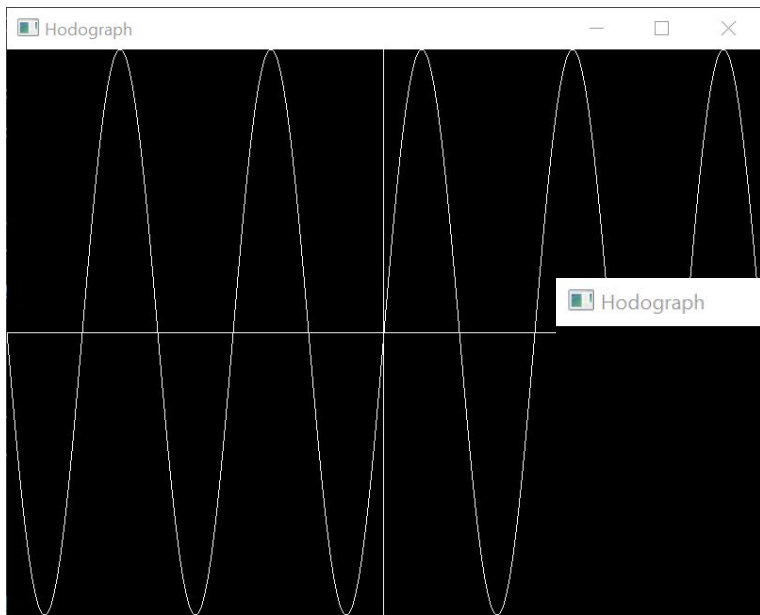
```
OpenGL version: 4.6.0 NVIDIA 531.14
```

```
GPU calculation time: 17.664127 ms
```

```
CPU calculation time: 424.837860 ms
```

```
Rate: 24.050882x
```


OpenGL + Windows.h + CUDA



Исходный код

