Lock_Map_Lab

December 6, 2023

Massive parallel programming on GPUs and applications, by Lokman ABBAS TURKI

1 14. Locked memory and mapped memory on the host

1.1 14.1 Objective

The main purpose of this lab is to show additional options for memory allocation on the host memory. We already know the usual pageable memory space allocated with malloc and the unified memory that, depending on the example, can be located on the host memory. The benefits of using locked and mapped memory allocations on the host depend mainly on the considered applications. However, there are at least two situations for which locked and mapped are difficult to replace: (i) asynchronous data transfer using locked memory, (ii) virtually extending the device RAM using mapped memory. In the following exercise, we see what is the syntax associated with each one of these options.

Students need to use the CUDA documentation, in particular:

1) the specifications of CUDA API functions within the CUDA_Runtime_API.
2) the examples of how to use the CUDA API functions in CUDA_C_Programming_Guide

1.2 14.2 Content

Compile MemComp.cu using

[ ]:
```
!nvcc MemComp.cu -o MemC
```

Execute MemC using (on Windows machine ./ is not needed)

[ ]:
```
!./MemC
```

As long as you did not include any additional instruction in the file MemC.cu, the execution above is supposed to return

Processing time when using malloc CPU2GPU: 0.044589 s
Processing time when using malloc GPU2CPU: 0.049532 s
Processing time when using cudaHostAlloc CPU2GPU: 0.000000 s
Processing time when using cudaHostAlloc GPU2CPU: 0.000000 s
Processing time for mapped memory: 0.000000 s

Of course, the execution time changes at each call and depends on the machine's performance.

1.2.1 14.2.1 Locked memory allocation

a) Explain the choice of threads and of block «<(size+127)/128,128»> in lines 50 and 52.

b) Do we need to launch kernels to compare data transfer of locked to pageable memory?

c) Use cudaHostAlloc, cudaFreeHost, and complete the syntax of hostAlloc_trans.

d) Compare the execution time of hostAlloc_trans to malloc_trans.

1.2.2 14.2.2 Mapped memory

a) Using cudaDeviceGetAttribute, check if you can map the host memory.

b) Do we need to launch kernels to compare data transfer of locked and pageable to mapped memory?

c) Call cudaSetDeviceFlags(cudaDeviceMapHost) at the beginning of the main function.

d) cudaHostAlloc should be called with the option cudaHostAllocMapped.

e) Get the GPU pointer using cudaHostGetDevicePointer.

f) How to check that the result is correct?

g) Compare the three solutions.

[ ]: