Parallel Architectures (COM3012)

Assignment 2 – Parallel Solution Development

Due: 20 May 2014, 16:00

Dr Johann A. Briffa

Instructions

- This assignment is to be attempted individually. It is essential that the work you submit and present consists only of your own work; use of copied material will be treated as plagiarism. Discussion is only permitted on general issues, and it is absolutely forbidden to discuss specific details with anyone.
- An individual SVN repository is made available, and is used for submitting your work and receiving feedback. You are also strongly encouraged to make use of your repository to facilitate development.
- Your assignment submission is considered to be the contents of the folder `/submissions/assignmentX` relative to your repository root, where `X` is the assignment number (1 or 2). This folder should:
  - Contain all relevant Eclipse projects and any other requested material.
  - Contain only the sources and project settings (i.e. remove all binaries and build folders).

  Other methods of submission will not be considered.
- Each solution requested should be presented as a separate Eclipse project.
- The name of each Eclipse project must start with your username. This may be followed by a dash and any text you wish.
- Textual answers to any questions must be included in a *single* PDF format file named `report.pdf` at the root of the assignment folder. Each answer must clearly identify the question number to which it refers.
- In your submission, include only files and projects that *directly* answer the questions asked. Submission of irrelevant material may lead to a reduction in the grade obtained.
- Any submitted Eclipse projects should work *without modification* on NVIDIA Nsight Eclipse Edition as installed on the lab computers, with default settings (i.e. must not depend on any settings specific to your login). If your projects do not work in this way, it may lead to a reduction in the grade obtained.
- It is important to follow the above guidelines on file and folder naming *exactly* as an automated script is used to collect assignment submissions for assessment.
- This assignment is to be submitted (checked in) on SVN by the above deadline; late submissions will be penalized according to the rules in the undergraduate handbook.
- If there are extenuating circumstances which do not allow you to complete the assignment on time, you are required to follow the procedure specified in the undergraduate programme handbook.

Practical Exercise

This exercise asks you to port an implementation of a scientific computing algorithm to a parallel architecture using CUDA. You are also expected to optimize and test your implementation. Use the serial implementation developed in the first assigment as a reference implementation. The questions to be answered start below.

1. Port your serial implementation to a parallel architecture using CUDA, with the objective of making the solution as fast as possible. In doing so:

   - You may make use of whatever parallelization facilities the architecture provides, including advanced features.
   - The objective is to optimize the algorithm; time to load and save any data sets from disk should be disregarded.
   - You are strongly advised to make use of code profiling to guide your optimization strategy.
   - Document your design decisions as necessary. It may be useful to include sample input and output data or other formatted content.

   Your solution will be assessed according to the following criteria, as evidenced by the solution and its documentation:

   (a) Appropriate consideration of feedback from first assignment, updating problem statement accordingly. **[10 marks]**

   (b) Project setup to allow proper compilation with access to the parallel platform's facilities. Your setup must include a debug build that disables compiler optimization and includes debug symbols and assertions, as well as a release build that enables compiler optimizations and excludes assertions. **[10 marks]**

   (c) Division of the problem in a way that is suitable for the target platform, with rationale for the choice made. **[20 marks]**

   (d) Effective use of advanced facilities on the target platform, with rationale for the design choice. Advanced features include, but are not limited to, effective use of the hierarchical memory model in CUDA. **[20 marks]**

   (e) Optimization of code guided by profiling tools. Include annotated profiler output with optimization improvements made if necessary. **[10 marks]**

2. Practical results.

   (a) Determine the speedup obtained with parallelization, for a range of data set sizes. Include at least two sets: one that exhibits the expected properties of a small (but not trivial) data set, and one that can be considered large (but still within the capability of the available facilities). For this comparison, the reference implementation should be built for the Intel architecture. Timing tests for CUDA should be done on the Tesla C2075. **[10 marks]**

   (b) Compare this speedup with the potential additional computational speed after parallelization. **[10 marks]**

   (c) Discuss the difficulties in achieving full utilization for this algorithm. **[10 marks]**

Notes:

- The report should be written in double-column IEEE transactions style, with a font size of not less than 11 pt. Templates for LaTeX and Word can be found in the IEEE Digital Author Toolbox site [1].
- The report should not exceed six pages in length, including any title, figures, references, and appendices. If a longer report is submitted, only the first six pages will be assessed.
- The grade achieved is not proportional to the report length. It is not *expected* that reports should be any specific length, and as long as the questions are answered fully, the maximum mark is achievable.

References

[1] "IEEE author digital toolbox," Jan. 2012. [Online]. Available: http://www.ieee.org/publications_standards/publications/authors/authors_journals.html

A CPU Information

This is the output obtained with `cat /proc/cpuinfo` on `penguin01`.

```
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 23
model name      : Intel(R) Core(TM)2 Duo CPU     E8400  @ 3.00GHz
stepping        : 10
cpu MHz         : 2000.000
cache size      : 6144 KB
physical id     : 0
siblings        : 2
core id         : 0
cpu cores       : 2
apicid          : 0
initial apicid  : 0
fpu             : yes
fpu_exception   : yes
cpuid level     : 13
wp              : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36
   clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc
   arch_perfmon pebs bts rep_good aperfmperf pni dtes64 monitor ds_cpl vmx
   smx est tm2 ssse3 cx16 xtpr pdcm sse4_1 xsave lahf_lm tpr_shadow vnmi
   flexpriority
bogomips        : 5983.75
clflush size    : 64
cache_alignment : 64
address sizes   : 36 bits physical, 48 bits virtual
power management:

processor       : 1
vendor_id       : GenuineIntel
cpu family      : 6
model           : 23
model name      : Intel(R) Core(TM)2 Duo CPU     E8400  @ 3.00GHz
stepping        : 10
cpu MHz         : 2000.000
cache size      : 6144 KB
physical id     : 0
siblings        : 2
core id         : 1
cpu cores       : 2
apicid          : 1
initial apicid  : 1
fpu             : yes
fpu_exception   : yes
cpuid level     : 13
wp              : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36
   clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc
   arch_perfmon pebs bts rep_good aperfmperf pni dtes64 monitor ds_cpl vmx
   smx est tm2 ssse3 cx16 xtpr pdcm sse4_1 xsave lahf_lm tpr_shadow vnmi
   flexpriority
bogomips        : 5984.96
clflush size    : 64
cache_alignment : 64
address sizes   : 36 bits physical, 48 bits virtual
power management:
```

B Memory Information

This is the output obtained with `sudo dmidecode -t memory` on `penguin01`.

```
# dmidecode 2.9
SMBIOS 2.5 present.

Handle 0x1000, DMI type 16, 15 bytes
Physical Memory Array
        Location: System Board Or Motherboard
        Use: System Memory
        Error Correction Type: None
        Maximum Capacity: 8 GB
        Error Information Handle: Not Provided
        Number Of Devices: 4

Handle 0x1100, DMI type 17, 27 bytes
Memory Device
        Array Handle: 0x1000
        Error Information Handle: Not Provided
        Total Width: 64 bits
        Data Width: 64 bits
        Size: 2048 MB
        Form Factor: DIMM
        Set: None
        Locator: DIMM_1
        Bank Locator: Not Specified
        Type: DDR2
        Type Detail: Synchronous
        Speed: 800 MHz (1.2 ns)
        Manufacturer: 7F7F7F0B00000000
        Serial Number: C4EB062B
        Asset Tag: 0D0910
        Part Number: NT2GT64U8HD0BY-AD

Handle 0x1101, DMI type 17, 27 bytes
Memory Device
        Array Handle: 0x1000
        Error Information Handle: Not Provided
        Total Width: 64 bits
        Data Width: 64 bits
        Size: No Module Installed
        Form Factor: DIMM
        Set: None
        Locator: DIMM_3
        Bank Locator: Not Specified
        Type: DDR2
        Type Detail: Synchronous
        Speed: 800 MHz (1.2 ns)
        Manufacturer: FFFFFFFFFFFFFFFF
        Serial Number: FFFFFFFF
        Asset Tag: FFFFFF
        Part Number:

Handle 0x1102, DMI type 17, 27 bytes
Memory Device
        Array Handle: 0x1000
        Error Information Handle: Not Provided
        Total Width: 64 bits
        Data Width: 64 bits
        Size: 2048 MB
        Form Factor: DIMM
        Set: None
        Locator: DIMM_2
        Bank Locator: Not Specified
        Type: DDR2
        Type Detail: Synchronous
        Speed: 800 MHz (1.2 ns)
        Manufacturer: 7F7F7F0B00000000
        Serial Number: 6FEB062F
        Asset Tag: 0D0910
        Part Number: NT2GT64U8HD0BY-AD

Handle 0x1103, DMI type 17, 27 bytes
Memory Device
        Array Handle: 0x1000
        Error Information Handle: Not Provided
        Total Width: 64 bits
        Data Width: 64 bits
        Size: No Module Installed
        Form Factor: DIMM
        Set: None
        Locator: DIMM_4
        Bank Locator: Not Specified
        Type: DDR2
        Type Detail: Synchronous
        Speed: 800 MHz (1.2 ns)
        Manufacturer: FFFFFFFFFFFFFFFF
        Serial Number: FFFFFFFF
        Asset Tag: FFFFFF
        Part Number:
```

C GPU Information

C.1 GeForce GT520

This is the output obtained with `deviceQueryDrv` on `penguin01`.

```
CUDA Device Query (Driver API) statically linked version
Detected 1 CUDA Capable device(s)

Device 0: "GeForce GT 520"
  CUDA Driver Version:                           5.0
  CUDA Capability Major/Minor version number:    2.1
  Total amount of global memory:                 1023 MBytes (1072889856 bytes)
  ( 1) Multiprocessors x ( 48) CUDA Cores/MP:    48 CUDA Cores
  GPU Clock rate:                                1620 MHz (1.62 GHz)
  Memory Clock rate:                             897 Mhz
  Memory Bus Width:                              64-bit
  L2 Cache Size:                                 65536 bytes
  Max Texture Dimension Sizes                    1D=(65536) 2D=(65536,65535) 3D=(2048,2048,2048)
  Max Layered Texture Size (dim) x layers        1D=(16384) x 2048, 2D=(16384,16384) x 2048
  Total amount of constant memory:               65536 bytes
  Total amount of shared memory per block:       49152 bytes
  Total number of registers available per block: 32768
  Warp size:                                     32
  Maximum number of threads per multiprocessor:  1536
  Maximum number of threads per block:           1024
  Maximum sizes of each dimension of a block:    1024 x 1024 x 64
  Maximum sizes of each dimension of a grid:     65535 x 65535 x 65535
  Texture alignment:                             512 bytes
  Maximum memory pitch:                          2147483647 bytes
  Concurrent copy and kernel execution:          Yes with 1 copy engine(s)
  Run time limit on kernels:                     Yes
  Integrated GPU sharing Host Memory:            No
  Support host page-locked memory mapping:       Yes
  Concurrent kernel execution:                   Yes
  Alignment requirement for Surfaces:            Yes
  Device has ECC support:                        Disabled
  Device supports Unified Addressing (UVA):      Yes
  Device PCI Bus ID / PCI location ID:           1 / 0
  Compute Mode:
     < Default (multiple host threads can use ::cudaSetDevice() with device simultaneously) >
```

C.2 Tesla C2075

This is the output obtained with `deviceQueryDrv` on `tesla01`.

```
CUDA Device Query (Driver API) statically linked version
Detected 1 CUDA Capable device(s)

Device 0: "Tesla C2075"
  CUDA Driver Version:                           5.0
  CUDA Capability Major/Minor version number:    2.0
  Total amount of global memory:                 5375 MBytes (5636292608 bytes)
  (14) Multiprocessors x ( 32) CUDA Cores/MP:    448 CUDA Cores
  GPU Clock rate:                                1147 MHz (1.15 GHz)
  Memory Clock rate:                             1566 Mhz
  Memory Bus Width:                              384-bit
  L2 Cache Size:                                 786432 bytes
  Max Texture Dimension Sizes                    1D=(65536) 2D=(65536,65535) 3D=(2048,2048,2048)
  Max Layered Texture Size (dim) x layers        1D=(16384) x 2048, 2D=(16384,16384) x 2048
  Total amount of constant memory:               65536 bytes
  Total amount of shared memory per block:       49152 bytes
  Total number of registers available per block: 32768
  Warp size:                                     32
  Maximum number of threads per multiprocessor:  1536
  Maximum number of threads per block:           1024
  Maximum sizes of each dimension of a block:    1024 x 1024 x 64
  Maximum sizes of each dimension of a grid:     65535 x 65535 x 65535
  Texture alignment:                             512 bytes
  Maximum memory pitch:                          2147483647 bytes
  Concurrent copy and kernel execution:          Yes with 2 copy engine(s)
  Run time limit on kernels:                     Yes
  Integrated GPU sharing Host Memory:            No
  Support host page-locked memory mapping:       Yes
  Concurrent kernel execution:                   Yes
  Alignment requirement for Surfaces:            Yes
  Device has ECC support:                        Enabled
  Device supports Unified Addressing (UVA):      Yes
  Device PCI Bus ID / PCI location ID:           1 / 0
  Compute Mode:
     < Default (multiple host threads can use ::cudaSetDevice() with device simultaneously) >
```