opencl

PROGRAMMING LANGUAGE SURVEY ASSIGNMENT

Presented by V satya siddardha
CB.EN.U4CYS22063
TIFAC-CORE in Cyber Security
Amrita Vishwa Vidyapeetham, Coimbatore Campus



Outline

opencl in 2008 introduction

2 what is opencl?

Bibliography



Introduction

- OpenCL was initially developed by Apple.inc., which holds trade mark rights, and refined into an initial proposal in collaboration with technical teams at AMD, IBM, Qualcomm, intel, and Nvidia. Apple submitted this initial proposal to the Khronos Group
- This group worked for five months to finish the technical details of the specification for OpenCL 1.0 by November 18, 2008.
- This technical specification was reviewed by the Khronos members and approved for public release on December 8, 2008.



WHAT IS OPENCL [applications and uses]

- Low level language for high performance heterogeneous data parallel computation.
- Access to compute devices in your system: CPUs GPUs Accelerators (eg:, CELL...but that only exists on ps3 now)
- Open standard Good industry support
- Present we are using opencl3.0
- Based on C99.
- Good (familiar)
- Bad (ancient, low level language)

What is replacing OpenCL?

Vulkan is intended to replace OpenGL. It's a graphics API, whereas OpenCL is a compute API. Vulkan and OpenGL communicate only with graphics hardware.



APPlication

- OpenCL developers use C or C++- based kernel language to code programs that are passed through a device compiler for parallel execution on accelerator devices.
- It defines a C-like language for writing programs. Functions executed on an OpenCL device are called kernels
- OpenCL adopts C/C++-based languages to specify the kernel computations performed on the device with some restrictions and additions to facilitate efficient mapping to the heterogeneous hardware resources of accelerators.
- C++ for OpenCL supports most of the features (syntactically and semantically) from OpenCL C except for nested parallelism and blocks.
- ullet However, there are minor differences in some supported features mainly related to differences in semantics between C++ and C
- ullet C++ for OpenCL is supported by clang and its documentation can be found here. It enables developers to use most C++17 features in OpenCL kernels.
- All CPUs support OpenCL 1.2 only. NVIDIA: NVIDIA GeForce 8600M GT, GeForce 8800 GT, and likely more are supported. Apple (MacOS X only is supported) Host CPUs as compute devices are supported.

DATA TYPES

- Types of opencl kernels —.types of appications
- int cl-int
- float cl-float
- char cl-char
- var cl-var
- long cl-long
- void cl-void



opencl use cases

- What do real-world use cases for OpenCL look like?
- OCR: Optical character recognition, or OCR, is a compute-intensive task. OpenCL can significantly improve the performance of OCR applications, provided that the applications take advantage of parallelization.
- Image recognition. Identifying patterns or objects in images for purposes such as facial recognition is another compute-intensive task for which OpenCL can come in handy.
- Bitcoin mining. If you know much about Bitcoin, you know that the process for mining Bitcoin requires immense computing power — so much, in fact, that mining Bitcoin using standard CPUs is not practical today. For that reason, a variety of OpenCL Bitcoin- mining applications have been created.
- Image resizing. Resizing images is another compute-intensive operation that can benefit from OpenCL



AN OPENCL PROGRAM

- setup
- 1.get the divice
 2.create a context(for sharing a devices)
 3.create command queues(for submitting work)
- Compilation
- 1.create a program2.build the program(compile)3.create kernels
- Create memory objects
 Enqueue writes to copy data to the gpu
 Set the kernel arguments
 Enqueue kernel executions



HELLO WORLD

```
Int main()
printf("hello, world!");
return 0;
```

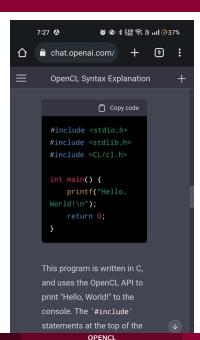


advantages and disadvantages of opencl

- Open Computing Language (OpenCL) serves as an independent, open standard for crossplatform parallel programming.
- OpenCL provides abstract memory and portability, due to its run-time execution model.
- The OpenCL kernel can run on any supported software implementation.
- OpenCL is used to accelerate supercomputers, cloud servers, PCs, mobile devices, and embedded platforms.
- A drawback of OpenCL is, that it does not support dynamic memory handling.
- This is required by typical PIC or hybrid approaches to dynamically remove or insert particles at every step of the simulation.
- OpenCL is not just for GPUs (like CUDA) but also for CPUs, FPGAs... In addition, OpenCL was developed by multiple companies, as opposed to NVIDIA's CUDA.



helloworld





```
#include "opencl.hpp"
    int main() {
       const Device device(select device with most flops()); // compile OpenCL C code for the fastest available device
       const uint N = 16u; // size of vectors
       Memory<char> buf(device, N); // allocate memory on both host and device
       const Kernel HelloWorld(device, N. "HelloWorld", buf); // kernel that runs on the device
       HelloWorld.run(); // run add kernel on the device
       buf.read_from_device(); // copy data from device memory to host memory
       println(buf.data()):
10
13 #include "kernel.hpp" // note: string literals can't be arbitrarily long, so periodically interrupt with )+R(
    kernel void HelloWorld(global char* data) {
       data[0] = 'H';
       data[1] = 'e':
       data[2] = '1';
       data[3] = '1';
       data[4] = 'o':
       data[5] = 32; // spaces are wrongly converted with stringification macro, so use ascii code here instead of ' '
       data[6] = 'W';
       data[7] = 'o';
       data[8] = 'r';
       data[9] = '1';
       data[10] = 'd';
       data[11] = '!':
       data[12] = '\n';
30 }
```

Figure: opencl



Thank you

Thank you.



References

```
data: https://en.wikipedia.org/wiki/OpenCL
data: https://developer.nvidia.com/opencl#:~:text=OpenCLU data:
https://www.khronos.org/opencl/
```



