



OpenCL 开发教程

V1.0

修定历史记录:

日期	版本	说明	作者
2017.12	Version<1.0>	文件创建	Wang

Note: 任何修改操作请在上述文档中备注说明。

www.jiang-niu.com

目录

一 OpenCL 概述.....	4
二 OpenCL 实例讲解.....	4
2.1 platform 实例.....	4
2.2 向量相加实例.....	6
2.3 矩阵转置实例.....	7

一 OpenCL 概述

OpenCL 是一个为异构平台编写程序的框架，此异构平台可由 CPU、GPU 或其他类型的处理器组成。OpenCL 是由编写 kernels（在 OpenCL 设备上运行的函数）的语言（基于 C99）和一组用于定义并控制平台的 API 组成。OpenCL 提供了基于任务分割和数据分割的并行计算机制。

OpenCL 类似于另外两个开放的工业标准 OpenGL 和 OpenAL，这两个标准分别用于三维图形和计算机音频。OpenCL 扩展了 GPU 图形生成之外的能力。OpenCL 由非盈利性技术组织 Khronos Group 维护管理。

二 OpenCL 实例讲解

编译 OpenCL 实例程序，在 Ubuntu PC 终端运行如下命令

```
cd ~/ti-processor-sdk-linux-rt-am57xx-evm-04.03.00.05/JiangNiu-demo/OpenCL  
make
```

```
building [matrix_transposition]  
building [platform]  
building [vectoradd]
```

2.1 platform 实例

实例源码位于 JiangNiu-demo/OpenCL/platform/platform.c

2.1.1 实例运行步骤

- 1 参考板卡开机登录章节，启动板卡。
- 2 在板卡上运行如下命令，启动 platform 实例

```
root@am57xx-evm:~# cd /opt/JiangNiu-demo/  
root@am57xx-evm:~# ./platform
```

实例运行结果，如下图 2-1-1 所示：

```
root@am57xx-evm:~# ./platform
[ 3743.226556] omap-iommu 41501000.mmu: 41501000.mmu: version 3.0
[ 3743.232472] omap-iommu 41502000.mmu: 41502000.mmu: version 3.0
[ 3743.246817] omap-iommu 40d01000.mmu: 40d01000.mmu: version 3.0
[ 3743.252724] omap-iommu 40d02000.mmu: 40d02000.mmu: version 3.0
platform name [TI AM57x]
platform vendor [Texas Instruments, Inc.]
platform version [OpenCL 1.1 TI product version 01.01.12.0 (Mar 29 2017 19:33:03)]
platform profile [FULL_PROFILE]
device type [accelerator]
device name [TI Multicore C66 DSP]
device global mem [160]MB
device global mem cache [128]KB
device local mem [128]KB
device max mem alloc [144]MB
device max clock frequency [750]MHZ
device max compute unit [2]
root@am57xx-evm:~#
```

图 2-1-1

2.1.2 实例分析

platform 程序主要用来检索 OpenCL device 的特性数据，包括执行频率，内存大小，cache 大小等等。

详细过程如下图 2-1-2 所示：

```
clGetDeviceInfo(device_id[j], CL_DEVICE_NAME, 512, strings, NULL);
printf("device name [%s]\n", strings);

clGetDeviceInfo(device_id[j], CL_DEVICE_GLOBAL_MEM_SIZE,
    sizeof(cl_ulong), &tmp, NULL);
printf("device global mem [%lld]MB\n", tmp/1024/1024);

clGetDeviceInfo(device_id[j], CL_DEVICE_GLOBAL_MEM_CACHE_SIZE,
    sizeof(cl_ulong), &tmp, NULL);
printf("device global mem cache [%lld]KB\n", tmp/1024);

clGetDeviceInfo(device_id[j], CL_DEVICE_LOCAL_MEM_SIZE,
    sizeof(cl_ulong), &tmp, NULL);
printf("device local mem [%lld]KB\n", tmp/1024);

clGetDeviceInfo(device_id[j], CL_DEVICE_MAX_MEM_ALLOC_SIZE,
    sizeof(cl_ulong), &tmp, NULL);
printf("device max mem alloc [%lld]MB\n", tmp/1024/1024);

clGetDeviceInfo(device_id[j], CL_DEVICE_MAX_CLOCK_FREQUENCY,
    sizeof(cl_ulong), &tmp, NULL);
printf("device max clock frequency [%lld]MHZ\n", tmp);

clGetDeviceInfo(device_id[j], CL_DEVICE_MAX_COMPUTE_UNITS,
    sizeof(cl_ulong), &tmp, NULL);
printf("device max compute unit [%lld]\n", tmp);
```

图 2-1-2

2.2 向量相加实例

实例源码位于 JiangNiu-demo/OpenCL/vectoradd/vectoradd.c

2.2.1 实例运行步骤

- 1 参考板卡开机登录章节，启动板卡。
- 2 在板卡上运行如下命令，启动 vectoradd 实例

```
root@am57xx-evm:~# cd /opt/JiangNiu-demo
```

```
root@am57xx-evm:~# ./vectoradd
```

实例运行结果，如下图 2-2-1 所示：

```
root@am57xx-evm:~# ./vectoradd
[ 4500.395511] omap-iommu 41501000.mmu: 41501000.mmu: version 3.0
[ 4500.401447] omap-iommu 41502000.mmu: 41502000.mmu: version 3.0
[ 4500.418757] omap-iommu 40d01000.mmu: 40d01000.mmu: version 3.0
[ 4500.424658] omap-iommu 40d02000.mmu: 40d02000.mmu: version 3.0
0 1 2 3 4 5 6 7 8 9
9 8 7 6 5 4 3 2 1 0
9 9 9 9 9 9 9 9 9 9
root@am57xx-evm:~#
```

图 2-2-1

2.2.2 实例分析

vectoradd 是可执行程序，vectoradd.cl 是核函数代码。vectoradd 程序将 A，B 向量相加，值存在 C 向量，OpenCL 核函数会根据用户提供的维数，将向量分解成多个任务分发给 DSP 计算，最后进行打印。

2.3 矩阵转置实例

实例源码位于 JiangNiu-demo/OpenCL/matrix_transposition

2.3.1 实例运行步骤

- 1 参考板卡开机登录章节，启动板卡。
- 2 在板卡上运行如下命令，启动 matrix_transposition 实例

```
root@am57xx-evm:~# cd /opt/JiangNiu-demo
```

```
root@am57xx-evm:~# ./matrix_transposition
```

实例运行结果，如下图 2-3-1 所示：

```
root@am57xx-evm:~# ./matrix_transposition
[ 4824.524716] omap-iommu 41501000.mmu: 41501000.mmu: version 3.0
[ 4824.530632] omap-iommu 41502000.mmu: 41502000.mmu: version 3.0
[ 4824.544771] omap-iommu 40d01000.mmu: 40d01000.mmu: version 3.0
[ 4824.550675] omap-iommu 40d02000.mmu: 40d02000.mmu: version 3.0
0 1 2 3
4 5 6 7
8 9 10 11
12 13 14 15
0 4 8 12
1 5 9 13
2 6 10 14
3 7 11 15
root@am57xx-evm:~#
```

图 2-3-1

2.3.2 实例分析

matrix_transposition 是可执行程序，matrix_transposition.cl 是核函数。
matrix_transposition 程序主要作用是将矩阵的行和列进行转换，即第 n 行变成第 n 列。