ACCELERATOR DESIGN WITH OPENCL

(ATHENS WEEK 19-24 MARCH, 2018)

WHAT DO WE KNOW SO FAR ?

- There are three types of parallelism
  - Task Parallelism
  - Data Parallelism
  - Pipeline
- The Amdahl's Law
- We saw the reasons for memory stalls and latency.
- The techniques to hide latency through Caching.
- The Virtual Memory.

WHAT DO WE KNOW SO FAR ?

- We saw the evolution of processors from
  - Uniprocessor to ...
  - Multicores with Simultaneous Multi-Threading.
- And we said hello to the world from our GPU (Mali-T628).

WHAT DO WE KNOW SO FAR ?

- We saw the GPU architecture evolve to a Multi-Threaded SIMD Mulitprocessor.
- We saw a SoC archietcture with and some deatils about Mali T628.
- We know how to launch a OpenCL kernel
  - Platfrom-> Device-> Context-> Command Queue -> compile kernel
  - Create Buffers-> Pass Buffers to GPU
  - Launch Kernel-> readback result buffers

WHAT DO WE KNOW SO FAR ?

- We know how to express parallelism using:
  - get_global_id()
- We know how to express dependence using:
  - get_local_id(), get_group_id()
- We learned about synchronization functions:
  - fence, barrier
- We saw the use of Mali Graphics Degugger

LAB WORK 1

- Vector addition with size N
- Calculate speedup with varying N.
- Measure Flops/s.
- Calculate the average of a vector.
- Calculate the average of a vector using workgroups.
- Measure speedup.

LAB WORK 2

- Write a Matrix multiplication routing with two matrices of size M x K, K x N.
- where M=K=N
- measure speed up
- use streamline to see various statistics about Cache/TLB miss.
- Measure Flops/S.

PROJECT: VIDEO FILTERING

- checkout the videofilter directory under tpt39/GPU
- we do two filters
  - GaussianBlur
  - Sobel Edge Detection
- opencv docs:
  - https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html?highlight=gaussianblur#gaussianblur
  - https://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/sobel_derivatives/sobel_derivatives.html

PROJECT: USEFUL FUNCTIONS

- to copy opencv mat to an array:
  - memcpy(cameraFrame.data, input, 3*ROWS*COLS*sizeof(char));
- to pad:
  - copyMakeBorder(src, dst, top, bottom, left, right, borderType, value);
  - https://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/copyMakeBorder/copyMakeBorder.html

PROJECT: USEFUL FUNCTIONS

- Mat data type:
  - https://docs.opencv.org/2.4/modules/core/doc/basic_structures.html#mat
- Mat Convert data type:
  - https://docs.opencv.org/2.4/modules/core/doc/basic_structures.html#mat-convertto

DEBUGGER: MGD

- in a405-xx.enst.fr (desktop) clone the git depot.
- source init.sh > /dev/null
- module load mali/4.4
- mgd
  - in odroid
  - source init_odroid.sh
  - mgddaemon
  - make debug

PERFORMANCE MONITOR: STREAMLINE

- run start_gator.sh in tpt39/
  - cd tpt39; ./start_gator.sh&
- in a405-XX.enst.fr
  - $ source init.sh
  - $ module load mali/4.4
  - $ streamline

DOMAIN SPECIFIC ARCHITECTURE