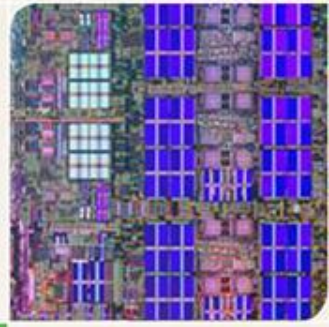
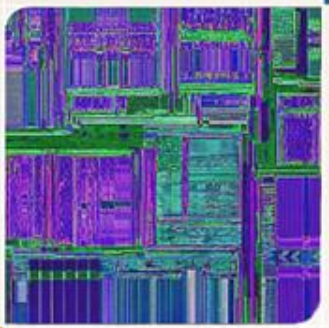
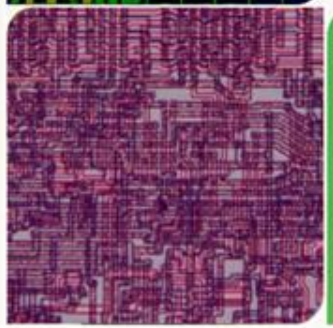



CS4803DGC Design and Programming of Game Console

Spring 2011

Prof. Hyesoon Kim

Georgia Tech College of Computing

Smart Phone ~= Game Devices

- More and more phone devices become game devices
- Game characteristics of phone/tablet games (report from Nokia)
  - Many of them are for time killing (snack games)
  - Still core mania
  - Less gender biased



Mobile Games

- Snack games
  - Just casual games
- Cross-platform
- Location-based games (GPS)
  - BotFighter
- Online games (multi-player)
- Gambling



Technology Issues

- Platform fragmentation
  - Different platforms: different screen sizes, keypad, software environment
  - Less a problematic in smartphones
- New interactions technologies:
  - Camera, GPUS, accelerators
- Connectivity:
  - More options for connecting internet
  - 2G: 300ms to one second
  - 3G: improve data rates but latency increases are marginal
  - 4G: could replace Wi-Fi



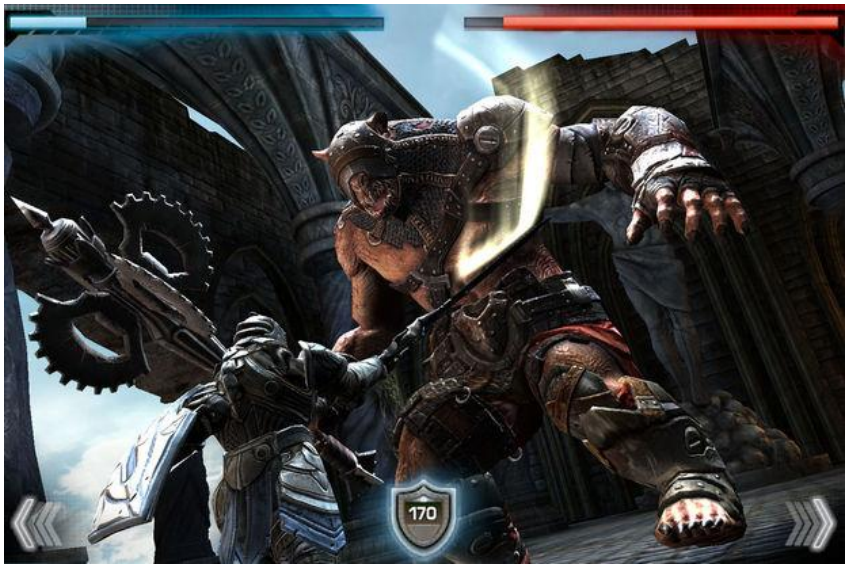
- Peripherals
  - Connect mobile phones with all other electronics
  - Use phone for generic peripherals
- Security
  - Copy the game across multiple platforms
  - App stores



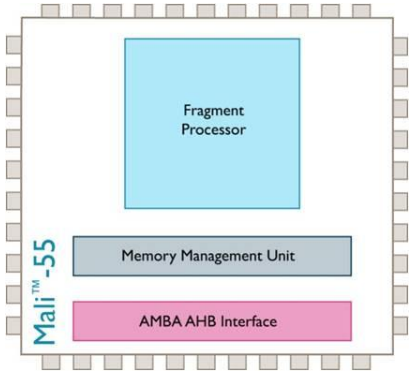
Mobile GPUs

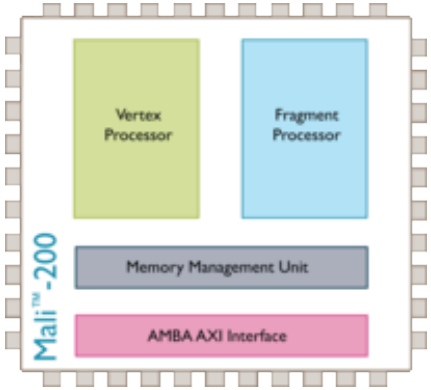
- 3 Major companies
  - ARM GPUs
  - Imagination
  - Nvidia
- Other companies
  - Freescale,
- So far still low performance than high-end GPUs
- But the performance & power issues are getting bigger

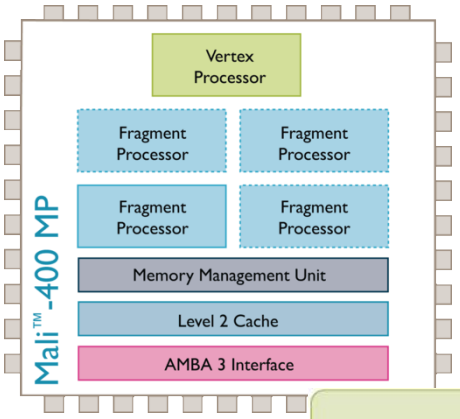
High Performance



ARM Mali Graphics Processors







- Mali-55: smallest graphics processors
- Mali-200
- Mali-400 MP
- Mali-T604

http://www.arm.com/products/multimedia/mali-graphics-hardware/mali-400-mp.php

Imagination: PowerVR SGX Core

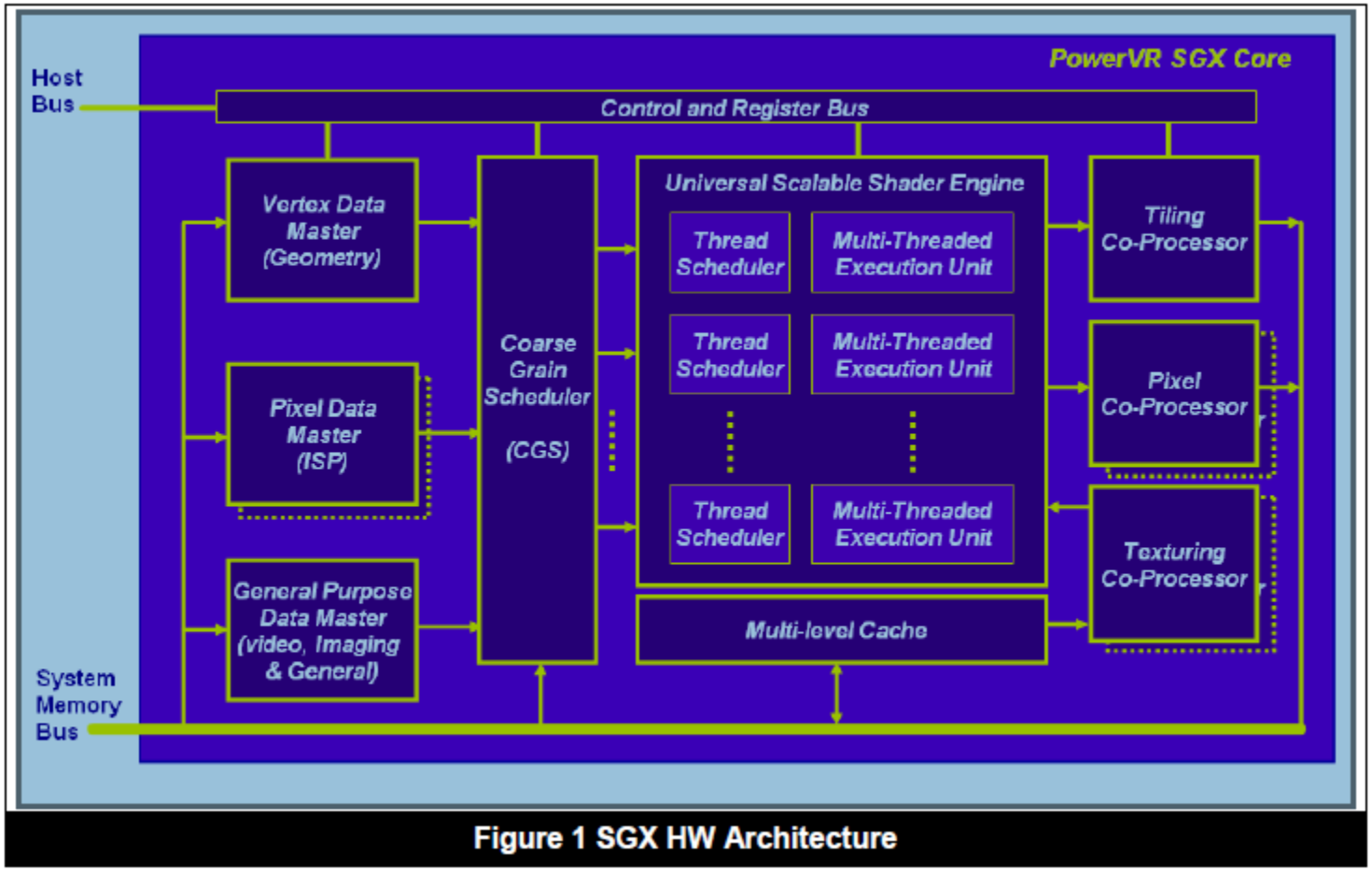
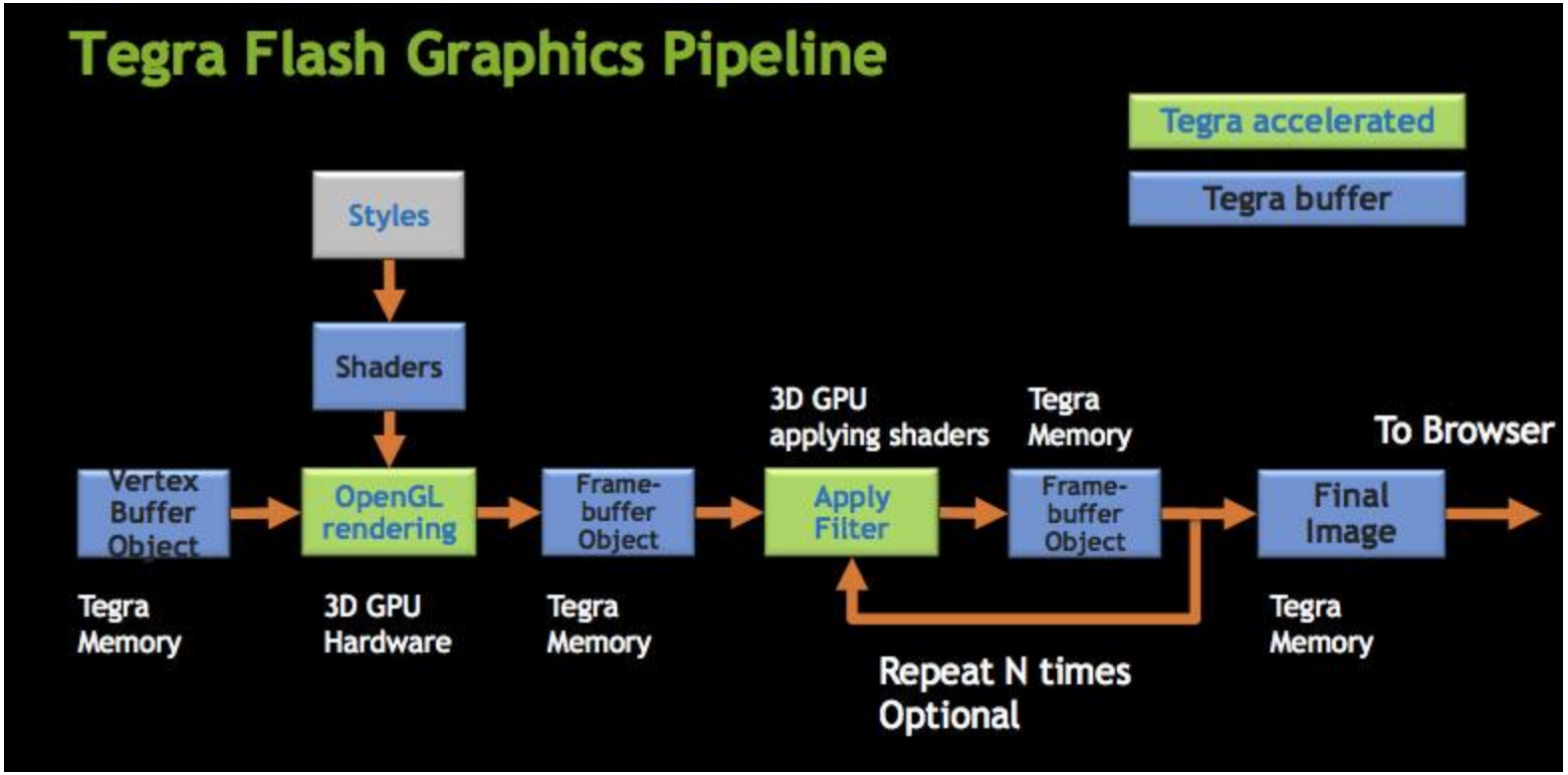


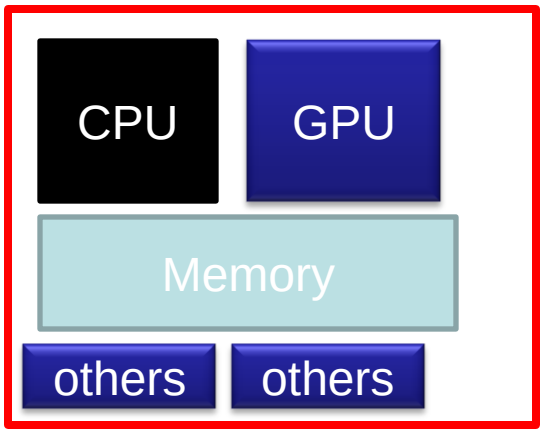
Figure 1 SGX HW Architecture

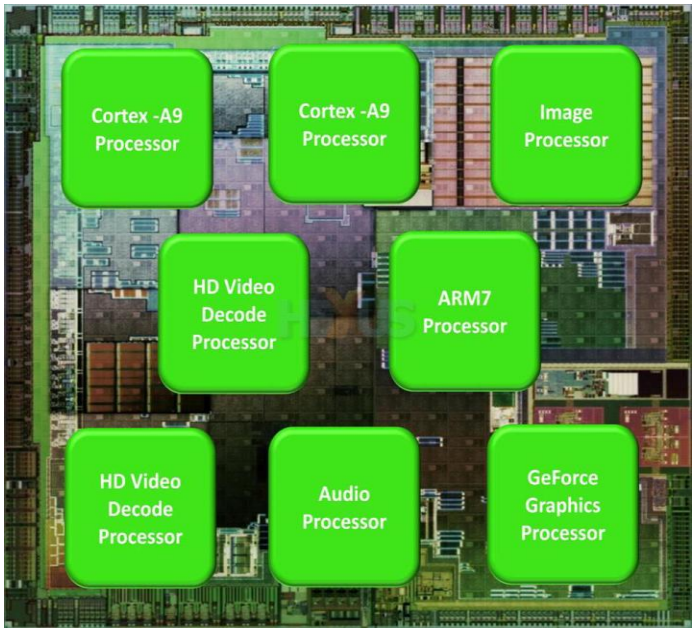
NVIDIA Tegra



SoC Designs

- Mobile platforms are SoC (System-On-Chip) designs





- New challenges
  - Sharing the memory bandwidth



Bandwidth Issues

- Scenario #1
  - CPU wants to run a job and needs data but GPUs use the memory
- Scenario #2
  - Opposite of Scenario #1
- Scenario #3
  - CPU and GPU split the work but need communications
  - Sending data from GPU to CPU and receiving data from CPU to GPU



Power Problems in SoC

- CPU and GPU are sharing the power
- CPU and GPU cannot use the peak power for long period.
- Cooling, power gating, clock gating etc.



Mobile OpenCL

- Currently each vendor has their own implementation versions
- Performance/Power consumption issues
- Imagination, Nokia
- Nokia's report:
  - Unintelligent scheduling algorithm makes CPU+GPU be worse performance/more power consumption than GPU only
  - Double data copy between CPUs and GPUs



MOBILE PROGRAMMING



Web Apps vs. Native Apps

- Web apps: websites
  - HTML, CSS, JavaScript
- Native apps: run it on locally, access to the hardware
  - iPhone/iPad: use Object-C , MacOS environment , SDK, Java
  - Android: Android SDK, Java



HTML/CSS

- CSS stands for Cascading Style Sheets
- Styles define how to display HTML elements
- Styles were added to HTML 4.0 to solve a problem
- Currently HTML 4
  - Html 5 (still evolving)
  - GPU-powered HTML5 : more graphics components



JavaScript

- Not Java !!
- A script language that can be added to an HTML page to make it more interactive and convenient
- More similar to functional languages
- Variables, arrays, objects, and control structures (if, while, for etc.)

Final Project Presentation (4/27)

- Submit your executable code into the google group before the presentation
- Students can play the games …
- 4-5 min presentation + 2-3 min demo
- Presentation contents:
  - Overall game architectures
  - New technologies that you have to use
- Report due: 4/29
- introduction, descriptions of the work (game architecture, CUDA algorithms. etc.) , output results (screen shots or bar graphs), challenges or difficulties, contribution from each team member , conclusion , references