The LPGPU2 Project
Ben Juurlink, TU Berlin
LPGPU2 Consortium

- LPGPU2 = Low-Power Parallel Processing on GPUs

[Logos of TU Berlin, Codeplay, Think Silicon, Samsung, and Spin Digital]
LPGPU2 Objectives

1. To improve the power efficiency of compute and graphics LPGPU2 applications running on mobile GPUs for existing and emerging APIs.

2. To build a reference monitoring architecture, a reference API, and reference metrics for quantifying the power efficiency of a wide-range of applications across third party mobile GPUs.

3. To increase the productivity and portability of LPGPU2 applications by standardizing the software hardware interfaces in mobile GPUs.

4. To reduce the hardware, software, and device driver design and development cycles of mobile GPUs.

5. To increase the technology readiness level (TRL), productize, and commercialize technologies developed in LPGPU FP7 STREP project.
Capitalizing on LPGPU

- LPGPU2 is successor of LPGPU project
- LPGPU was a very successful FP7 R&I project, but we need to bring the technology closer to the market (from lab-to-market)
- LPGPU2 is an innovation action and builds upon the most promising technologies developed within LPGPU
- LPGPU helped us understand GPU power consumption
- LPGPU2 focuses on supporting ordinary software developers to understand the power consumption of their graphical software
Work Packages

- WP1: Project Management
- WP2: Commercialization, standardization, dissemination, and exploitation
  - WP3: Applications
  - WP4: Power measurement, estimation, and modeling
  - WP5: Reference platforms
  - WP6: Performance measurement and visualization
  - WP7: Integration and validation

Src: https://balance-global.com/shop/project-management-templates/prince2-work-package-template/
WP3: Applications

- WP3 Objectives
  - to develop applications for or port applications to low-power GPUs
  - to optimize the applications using latest GPU APIs (OpenCL 2.x, SYCL, Vulkan, OpenGL ES 3.1, Compute Shaders)
  - to use these applications as benchmarks for the power and performance analysis framework

- WP3 Tasks
  - H.265 video player
  - VR rendering
  - Font rendering
  - Augmented reality
  - Real-time ISP camera processing
Next Generation Immersive Media Applications

- Emerging Video Quality Enhancements
  - UHD: 4K & 8K
  - HFR: 120 fps
  - HDR: 10- and 12-bit video
  - WCG: BT.2020 color space

- Emerging Application Domains
  - UHD TV
  - VR video

- Quality enhancements = very high bitrates
  - HEVC needed for efficient video compression & delivery
  - High-performance and low-power HEVC implementations needed for next generation media applications
H.265-enabled Media Player

- The complete application is a media player not just a video decoder
- The target architecture is a heterogeneous platform CPU+GPU
- The problem is mapping the entire application to the architecture for:
  - High performance
  - Power efficiency
- Select the most appropriate languages and APIs for each module: C++, SIMD intrinsics, OpenGL, Vulkan, OpenCL, CUDA

![Diagram showing the components of the H.265-enabled Media Player]

- Input reader
- Demuxer
- Video Decoder
  - Video Post-processing
  - Video Rendering
- Audio Decoder
  - Audio Post-processing
  - Audio Rendering
- Display Device
- Audio Device
Mapping the Application to the Architecture

- **HEVC decoder: mapped to the CPU**
  - Highest performance (8K in PC, 4K in mobile)
  - Multithreading: C++11 threads
  - SIMD (SSE, AVX, Neon): compiler intrinsics

- **Video rendering and post-processing: mapped to the GPU**
  - Highly parallel kernels: good match architecture – application
  - OpenGL: several limitations do not allow high performance rendering
  - DirectX-12 and Vulkan: allow multithreading and low-level memory optimizations
H.265 Media Player: Current Results

- PC solution
  - Intel multicore: up to 2x22 cores
  - Support for UHD: 4K & 8K resolutions
  - Support for HDR and WCG
  - Working on 8K @ 120 fps

- Mobile solution
  - ARM multicore: up to 4 cores
  - Support for FHD and 4K resolutions

- Ultra high performance HEVC decoder
- High performance multi-API video renderer
  - DX12: PC-based systems
  - Vulkan (in progress): PC + Mobile

- Media player for next generation media
VR Rendering

Applications
• Virtual scene
  • Bluetooth controller to move around
• Video viewer
  • Render cube map & equirectangular 360 videos

Features
• Uses OpenGL ES 3.1
• 2 thread rendering(lens correction+ application)
• Only on Samsung device

Two threads rendering
1. Game/video rendering
2. Lens correction
GPU Font Rendering

**Application**
Using GPU to directly rendering font

**Features**
- Uses OpenGL ES 3.1 fragment shader
- Avoid CPU path rendering
- Works on all android devices
- Fonts can easily be scaled as compare to CPU version

![GPU vs CPU rendering comparison](image)
Augmented Reality

- Augmented reality
  - Using android Camera framework, SLAM & OpenGL ES 3.1 on android

- Real-time SLAM for positional tracking
  - Testing in VR environments (less sensitive to tracking accuracy)
Real-time ISP Camera Processing

- Complete (13-stages) **programmable ISP-pipeline** developed & ported in Nema eGPUs
  - Closed-loop, reconfigurable behavior based on image statistics & user input → easily adapted to new cameras
- Pipeline stages (image filters) selected (& modified) based on memory access patterns and complexity vs. image quality trade-offs
- Implementation suitable for **throughput-optimized eGPUs**
- **Current state:** Use **Vulkan compute shaders** & optimize them using **LPGPU2 tool** (codename: Think Silicon NemaISP)
WP4: Power Measurement, Estimation & Modeling

- WP4 objectives
  - Develop a **power model for heterogeneous architectures** used in mobile systems
  - Derive **min. set of HW performance counters** needed for accurate power estimations
  - Create a **power measurement testbed** for mobile systems
  - Develop a **microbenchmark suite** for obtaining empirical power models of different mobile SOC components
  - Create a **power model for a configurable GPU** based on detailed circuit simulations
  - Build a GPU driver guided **DVFS controller**

- WP4 tasks
  - Power measurement testbed
  - Performance counter-based power estimation model
  - Microbenchmark suite
  - **VLSI-level configurable power models**
  - Profiling-driven DVFS decisions
Power Measurement Testbed

- New flexible power measurement testbed for embedded SoCs
- Modular
- High Sample-rate and Resolution
- Much closer to a product than LPGPU testbed
Performance counter-based power estimation model

- CMOS dynamic power consumption depends on switching activity
- Switching activity depends on processed data
- Current performance counter-based power models do not capture this
- RTL and gate-level power tools are accurate but very slow
ALU.Power Model

• Parameters: several Hamming distances and population counts
• Linear model with 7 coefficients
  \[E(a_0,b_0,a_1,b_1) = c_0 + c_1 \text{HD}(a_0,a_1) + c_2 \text{HD}(b_0,b_1) + c_3 \text{HD}(o_0,o_1) + c_4 \text{HD}(a_0,b_0) + c_5 \text{HD}(a_1,b_1) + c_6 (\text{POPC}(a_0)+\text{POPC}(a_1)+\text{POPC}(b_0)+\text{POPC}(b_1))\]
• Linear least square fit to estimate coefficient
Power Prediction Accuracy

- Model works extremely well for logic instructions and IADD
- Almost perfect prediction
- Accuracy of AND, OR, XOR and IADD limited by measurement noise
- Good accuracy for other instructions as well
- On average 86% more accurate than previous models
- Mean Error 2.7 pJ
- Very high correlation (98%) of predictions with actual measurements
Android Power Model

- Predicts power consumption from performance counters
- Trained by microbenchmarks
- Collects various performance counters from standard Android and Linux event sources and vendor-specific APIs
Going one step further in GPU Power Modelling

• Highly accurate GPU power models built and verified at VLSI (netlist) level
  • High accuracy due to
    • No noise from the measurement device
    • Ability to concentrate on specific “problematic” HW components
    • No limitations from the existing set of performance counters
• Key Distinction Points
  • Power model parameterized from different GPU configurations
    • Number of cores, HW threads, pipeline parameters, process technologies, Voltage/frequency levels (Nema GPU: IP from Think Silicon)
    • Ability to extract a different set of performance counters for each GPU configuration
  • Build the power model and performance monitoring architecture in tandem
    • Ability to increase/decrease the accuracy of the model (in an almost predefined way) by increasing/decreasing the number of the required performance counters
Approach

- Internal operation
  - Navigate up/down in RTL hierarchy until requested accuracy is met
  - Navigation is performed in a divide & conquer
Power Model Accuracy vs. Performance Monitoring HW Overheads

- Power model training using selected testbenches; Evaluation using ISP-post camera processing applications
- Results for TSMC LP @ 40nm; Similar results for FDSOI @28nm
WP5: Reference Platforms

• WP5 objectives
  • to build the data collection and data capture part of the tool suite
  • to build the internal part of the tool
  • to extract the GPU-CPU performance monitoring counters
  • to build the necessary run-time system
  • to extend the GPU driver with the above features

• WP5 tasks
  • Performance Monitoring Hardware
  • Target data collection
  • Data capture and parsing
  • Run-time system
  • SPIR-V support
WP5: Reference Platforms

• Work in this WP has just started and will be presented at PEGPUM 2018 (!?)
WP6: Performance Analysis and Visualization

- WP6 objectives
  - Produce the components of the tool that will analyse and visualise the performance data from WP5.
  - Enable users of the tool to obtain a detailed understanding of the performance of their application.
  - Enable users of the tool to obtain a detailed understanding of the power requirements of their application.
  - Enable users of the tool to investigate how to improve their applications performance and power requirements.
  - Allow expert developers to optimise their applications more quickly.
  - Allow non-expert developers to gain insights into their performance problems.
  - Detect and suggest possible fixes to common performance or power problems.

- WP6 tasks
  - Low-level Performance Visualisation and Analysis
  - High-level Performance Visualisation and Analysis
  - Feedback
LPGPU2 Tool Suite

• The LPGPU2 tool suite is a fork of AMD’s CodeXL which augments its support for low-power GPU devices.
  • CodeXL is an open source software development tool suite that includes a GPU debugger, a GPU profiler, a CPU profiler, Graphics frame analyzer and a static shader/kernel analyzer.
• Support for tracing of SYCL and OpenGL runtime API calls has been added.
• A code coverage mode for evaluating test suites for heterogeneous programming models is being worked on.
• Power & performance data will be correlated with source code to guide optimization of power usage.
LPGPU2 Profiler

- Actual memory transfer on OpenCL
- Actual kernel execution on OpenCL
- SYCL command group (read, write, execute)
- Total execution time for SYCL
WP7: Integration and Validation

• WP7 objectives
  • To ensure that all components of the produced tool suite integrate successfully.
  • To validate the LPGPU2 tool suite as a means of optimizing applications.
  • To validate the LPGPU2 tool suite as a means of driving hardware design decisions.
  • To provide multiple, strong demonstrations of the tool suite’s effectiveness.
  • To produce optimized versions of the applications from WP3.

• WP7 tasks
  • Technical Oversight of Centralised Plug-in Framework
  • Power Estimation Integration
  • Assessment of DVFS Decisions
  • Autotuning: GPU customization
  • Optimizing Applications Using the Tool Suite
  • Optimizing Applications Using the Tool Suite (external)
Optimization of External Applications Using the Tool Suite

• We’re interested in others using our tool suite
• EU is interested in “leadership in next generation open and interoperable platforms that any business can use to make its products, processes or services ready for the digital age”

➢ This is an external task that will be subcontracted. In this task an external subcontractor will optimize a relevant but to be selected application using our toolset and provide detailed feedback to improve the toolset. This will be another real-life test of the platform and will increase its popularity.

➢ You can participate in the tender that will be launched 2nd half of 2017
Conclusions

Among others LPGPU2 will produce

• novel advanced applications on mobile platforms
• power measurement testbed close to a product
• very accurate power models based on as few performance counters as possible
• a tool suite that allows debugging, profiling, and estimating and improving the performance and power consumption of applications on low-power GPUs for various existing and emerging APIs

• Follow us at http://lpgpu.org