The Tip of Iceberg in Open-Source Hardware GPU
Blaise Tine, Ruobing Han, Hyesoon Kim
Georgia Tech
Outline

● Motivations
● Vortex GPU Platform
● Compiler Support
● Driver Support
● Software Stack
● Simulation Stack
● Debugging Stack
● The Future of OpenGPU
Motivations

Conventional GPU Architecture Research

• Focus on cycle-level simulation
• ISAs are proprietary
• No Full-system open-source GPU

The true cost of open-GPU research

• RTL is a smaller challenge
• An ISA extension is costly
• Compiler changes
• Software support
• Simulation support
• Debugging support

[Diagram showing Hardware, Simulation, Compiler Software, Testing, Debugging]
Motivations – Four Pillars

Simulation
- Pre-RTL evaluation
- Design-space exploration

Compiler
- Enabling language support
- Device-specific optimizations

Software
- User Applications
- OS driver support

Hardware
- RTL implementation
- FPGA Prototype
- ASIC fabrication
Vortex GPU Platform

<table>
<thead>
<tr>
<th>Supports OpenCL API</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Target FPGA:</td>
</tr>
<tr>
<td>• Stratrix10 Intel FPGA</td>
</tr>
<tr>
<td>• 64 cores (1024 H/W threads)</td>
</tr>
<tr>
<td>• @250 MHz, 16 GB/s BW</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Key Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>• PCIe-based Host communication</td>
</tr>
<tr>
<td>• High-bandwidth Cache sub-system</td>
</tr>
<tr>
<td>• Multi-channel memory system</td>
</tr>
<tr>
<td>• Design scaling &amp; configuration</td>
</tr>
<tr>
<td>• Pipeline elasticity</td>
</tr>
</tbody>
</table>

Vortex: Extending the RISC-V ISA for GPGPU and 3D-Graphics Research

Blaise Tine, Fares Elsabbagh, Krishna Yalamarthy, Hyesoon Kim – MICRO21
ISA Extension for GPGPU

Threading model
- Thread clustering: Wavefront

Memory model
- Global / shared memory
- Texture / constants memory

Register File
- Per-thread registers

Thread scheduling
- Wavefront activation
- Thread mask

Flow control
- Split, Join

Synchronization
- Wavefront barrier

Implementing Hardware Extensions for Multicore RISC-V GPUs
Blaise Tine, Fares, Hyesoon Kim – CARRV22
Compiler Support

Assembler/Disassembler
- Toolchain integration
- Code dump debugging

Automating Code Translation
- Identify code pattern
- Insert new instruction
- Code restructuration
- SW fallback

Sample Applications
- Split/join insertion
- Barrier insertion
- Wspawn insertion
- Texture sampler insertion
Driver Support

GPU Driver Roles

- Interface between SW and HW
- Low-level OS abstraction
  - Kernel API
  - I/O drivers
- Low-level HW abstraction
  - DRAM controller
  - PCIe controller
  - GPIO controller
  - JTAG controller
Software Stack - OpenCL

OpenCL Compiler
- Use POCL Compiler framework
- Added Vortex kernel runtime pass
  - Work items => Vortex threads
  - Wavefront invocations

OpenCL Runtime
- Use POCL Runtime framework
- Added new device target for Vortex
- FPGA Driver uses Intel OPAE API
Software Stack - CUDA

NVVM-IR to SPIR-IR Translation
• Device-independent instructions
• Build-in function
• Meta-data selection

SPIR-V to OpenCL
• Use POCL Compiler support
• Use POCL Runtime support
Simulation Stack

Cross-Platform Simulation

- FPGA
  - Device (Intel FPGA)
  - ASE SIM (Intel ASE)
- RTL Simulation
  - RTLSIM (Processor only)
  - VLSIM (Processor + command processor)
- Cycle-Level Simulation
  - SimX

- A common driver API
  - Same application runs anywhere
Simulation Stack (2)

SimX Emulator
- Full ISA emulation
- Trace generation

SimX Timing Simulation Engine
- Event-base
- Communication ports
- Template abstraction

SimX Timing Simulator
- Full GPU Pipeline simulation
- High-Bandwidth Caches
- Ramulator-based Memory System
Debugging Stack

Basic console output
- No console access
- Multiple threads
- Override system calls

RISC-V Debug Extension
- Per-warp-thread registers
- Hardware support
  - Debug module
- Software support
  - Customized for GPU
- Simulation support
Proposed Solutions

Configurable toolchain

- Facilitate compiler support for new extensions
  - e.g. GCC insn syntax
- Configurable driver APIs
  - e.g. OPAE
- Configurable Software APIs
  - e.g. POCL
The Future of OpenGPU

Hardware Extensions
- 3D Graphics
- Graphics Analytics
- Ray Tracing
- Custom Extension

Software Support
- CUDA Native Support
- Vulkan Graphics API

Cross-platform Synthesis
- Altera, Xilinx
- ASIC

SoC Integration
- ESP – Open SoC
Thank You!

Project Website
vortex.cc.gatech.edu