

Panfrost

A reverse engineered FOSS driver for Mali Midgard and Bifrost GPUs

Contributors

- Alyssa Rosenzweig
 - Most of the Midgard RE/Driver development
- Connor Abbott
 - Initial Midgard ISA RE
 - Most of the Bifrost ISA RE
- Lyude Paul & BiOpen
 - Helped with Bifrost utilities (assembler+panwrap)

Panfrost

- Aiming to support the following lines of ARM GPUs
 - Midgard (Mali Txxx)
 - Bifrost (Mali Gxx)
- Reverse engineered from tracing ARM's userspace 3D drivers and the open source kernel driver from ARM
- 3D only! Most of the variants of display hardware used with Mali GPUs have mainline drivers

Midgard

Midgard

- Codename for ARM's Mali Txxx line of embedded GPUs
- In many, many devices
- Especially Chromebooks

Midgard

What's done?

- ISA reverse engineered
 - NIR-based compiler supporting simple shaders
- ES 2.0 command stream reverse-engineered
 - Prototype Mesa driver

Midgard ISA Architecture

- Three unit types:
 - ALU
 - Load/store unit
 - Texture unit
- Limited parallelism

Midgard ALU unit

- Both scalar and vector (SIMD) units
 - VMUL, VADD, SMUL, SADD, VLUT
- Pipelined, VLIW-packed, some parallelism
 - VMUL, SADD concurrent
 - VADD, SMUL, VLUT concurrent
- Various opcodes allowed on a given unit
- Scheduler goal: minimize number of bundles and register pressure

Midgard

Job chains

- Unlike many other GPUs, Midgard+ use “job chains” instead of true command streams
- Chains contain pointers to each job’s dependencies
- Essentially acts as a HW level GPU scheduler

Midgard

Job chains

- All configuration in nested memory structures
 - Shader core descriptor, fragment descriptor, etc
- Ideally, entire frame in memory and submitted at once
- Tiled rendering
- Types of jobs:
 - Vertex: vertex shaders
 - Tiler: sorts triangles into tiles, passes down to fragment shader
 - Fragment: final rasterization over passed in tiles
- Hardware internals generally hidden from driver

Bifrost

Bifrost

- Codename for ARM's Mali Gxx line of embedded GPUs
- Latest gen, not in many devices yet
- Shader core completely redesigned since Midgard
- Shares a kernel driver with Midgard

Bifrost

What's done?

- Much of the ISA has been reverse engineered and documented
- Cmdstream partially RE'd, still more work to do
- No mesa stub yet!

Bifrost

Basic architecture

- Clause based
- Scheduler chooses clauses to run, not instructions
- Clauses contain instructions and immediates, unpacked by GPU at runtime
- Scheduling uses scoreboard mechanism
- No high latency instructions

Bifrost

Basic architecture

- Instructions have 3 stages
 - Register read/write
 - FMA
 - ADD
- Have 2 read ports, 1 read/write, 1 write, and one const port
- Results of FMA/ADD can be passed through to next instruction (skips register file, less power and spilling)

Bifrost

Basic architecture

- Has four other units:
 - Varying interpolation unit
 - Attribute fetching unit
 - Load/store unit
 - Texture unit
- Execution unit interacts with other units through special variable-latency instructions
- Only one special instruction per clause
 - Bypasses fixed-latency mechanism used for registers

Bifrost

Basic architecture

- Clause packing is **intense**
- 12 different formats used for instructions in clauses, 13 in total
- Instructions usually end up split between multiple quadwords
- Constants sometimes packed in instruction quadwords

Current Tools

- Panloader: main repo for utilities such as
 - Panwrap: provides userspace tracing/recording/replay
 - Midgard&Bifrost assembler
 - Shader runner
- ShaderProgramDisassembler
 - Midgard&Bifrost disassembler

Links

- <https://gitlab.freedesktop.org/panfrost>
- Build instructions for T8xx:
<https://panfrost.freedesktop.org/building-panfrost-mesa.html>

Demo time!