Raspberry Pi 5: Challenges and Solutions in Bringing up an OpenGL/Vulkan Driver for a New GPU

EOSS EMBEDDED OPEN SOURCE SUMMIT

igalia

Who I am

- Alejandro Piñeiro Iglesias (apinheiro@igalia.com)
- Working at the Igalia graphics team on the Raspberry Pi graphics stack for 4 years, and previously on Intel GPU

Mesa

- Open Source implementation of OpenGL, Vulkan, and other graphics API specifications.
- Implements several pieces that can be reused for different drivers
  - OpenGL and Vulkan API translation
  - GLSL and SPIR-V parsing, compiler
  - etc

Mesa - Drivers

- Intel (i965, i915, Iris, Anvil, Crocus)
- AMD (radeonsi, r600, radv)
- NVIDIA (nouveau, nvk)
- Vivante (etnaviv)
- Qualcomm Adreno (freedreno, turnip)
- Arm mali (lima, panfrost, panvk)
- **Broadcom (vc4, v3d, v3dv)**

Raspberry Pi

- Raspberry Pi is a series of small single-board computers
- Originally leaned toward promotion of computer science teaching
- Become really popular and expanded outside its target market, now being used in diverse uses
- ARM processor, VideoCore Broadcom GPU

Raspberry Pi – timeline

- Raspberry Pi – 2012
- Raspberry Pi 2 – 2015
- Raspberry Pi 3 – 2016
- Raspberry Pi 4 – 2019
- Raspberry Pi 5 – 2023

Raspberry Pi - drivers

HW	Kernel Driver	Mesa Driver	GPU
Raspberry Pi 1/2/3	vc4 (display+render)	vc4 (GL/ES)	VideoCore4

Raspberry Pi 4	vc4(display)
	v3d(render)
		v3d(GL/ES)
		v3dv(Vulkan)
			VideoCore6

Raspberry Pi 5	vc4(display)
	v3d(render)
		v3d(GL/ES)
		v3dv(Vulkan)
			VideoCore7

Raspberry Pi - Igalia

- Igalia started to work with the Broadcom mesa drivers in 2018, slightly after RPI4 announcement
- VC4 and V3D were already working, and available upstream on Mesa
- Initial work centered on mature existing driver and extend functionality (like extensions)
- RPI4 has been OpenGL ES 3.1 conformant since 2020

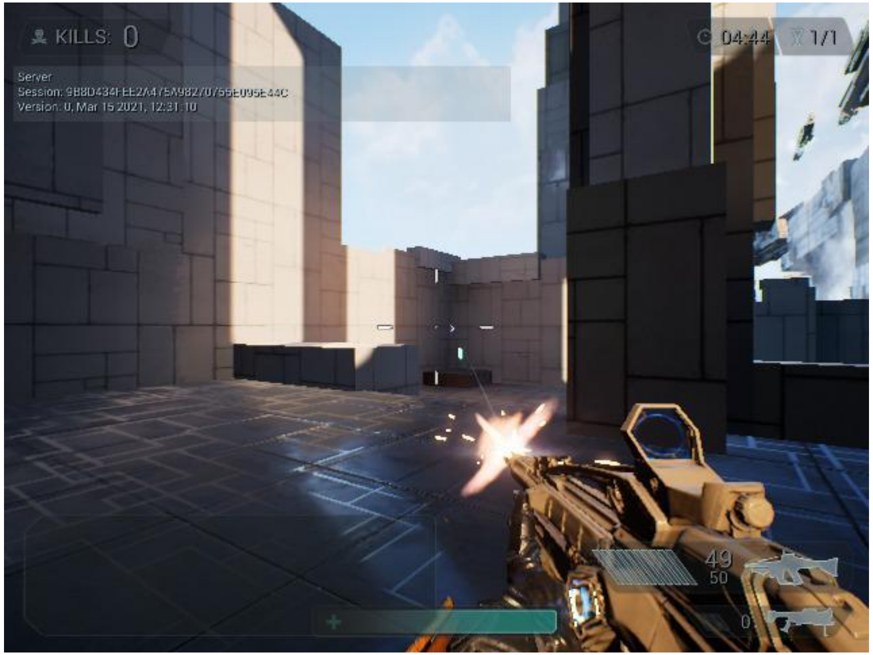
Raspberry Pi – v3dv (I)

- Vulkan driver for the rpi4
- “From scratch”: re-use as much as possible
  - Mesa frontend
  - Several v3d components were re-used and refactored to become common: HW command definitions, ISA definition, compiler, simulator

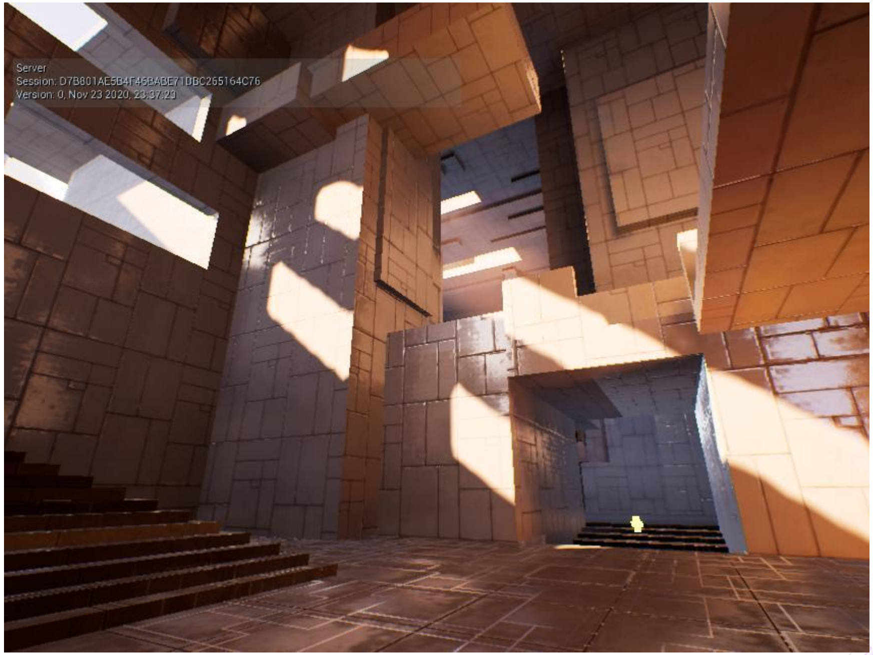
Raspberry Pi – v3dv (II)

- November 2019: starting
- January 2020: Triangle!
- August 2020: Vulkan 1.0 mandatory
- October 2020: Vulkan 1.0 conformant
- December 2020: Performance work
- December 2021: Vulkan 1.1 conformant
- July 2022: Vulkan 1.2 conformant

Raspberry Pi – v3dv (III)



```
Server
Session: 9B8D434FEE2A475A98270755E095E44C
Version: 0, Mar 15 2021, 12:31:10
```



```
Server
Session: D7B801AE5B4F46BABE71DBC265164C76
Version: 0, Nov 23 2020, 23:37:23
```

RPI 5 – description (I)

- GPU Broadcom V3D 7.1.6, same VideoCore architecture
  - Higher clock rate
  - Up to 8 RTs
  - Better support for subgroup operations
  - Better instruction-level parallelism (but a bit more register pressure!)...

RPI 5 – description (II)

- GPU Broadcom V3D 7.1.6, same VideoCore architecture
  - More image format supported
  - New assembly instructions (like packing)
  - HW support for several features
    - fullDrawIndexUint32
    - depthClamp and depthBounds, etc

But before thinking on working on all those new features …

RPI 5 – a lot of changes

- The new features added significant changes on the HW packets
- New assembly instructions, better parallelism, etc brought several changes on how to deal with the GPU set of instructions, registers, and signals
- Started work under the assumption of not needing a new driver, but v3d/v3dv with more codepaths

RPI 5 – pre work

- v3d codebase was already organized in a way that supported several HW generations
- v3dv was not, so this work was added and upstreamed
- During all development we tried to upstream any change coming from the rpi5 support work if possible
  - At least one rebase against mesa main per week

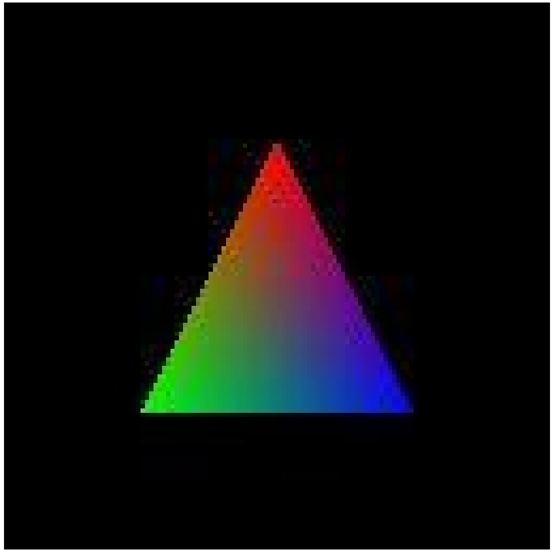
RPI 5 – simulator (I)

- Work started without HW available, using a simulator
  - Not feasible for real apps. Not a problem as ...
- Even regular tests (CTS) were too complex at the beginning
- Started with just one driver (v3dv)
- Started with the more simple tests possible
- Hard to parallelize task: one developer first

RPI 5 – simulator (II)

- Simpler tests only required some HW packets updates
- Most of initial work done on the compiler
- Focus on getting something working
  - Initially all optimizations (instruction merge, using immediates, etc) were disabled

RPI 5 – simulator (III)



The classic triangle is back!

RPI 5 – simulator (IV)

- Moved to have two people working
- Still first with small tests, then moved to use CTS as reference
  - Got most of the Vulkan CTS tests passing
- Port the changes to v3d
- Re-implement compiler optimizations to RPI5
- Adding features not supported by RPI4

RPI 5 – real device (I)

- We got the real device!
- Bootable kernel, but without GPU support
- Surprises:
  - Changes on some IRQs
  - Difference on how handle compute shaders
  - Incomplete documentation (mostly typos)

RPI 5 – real device (II)

- Testing OpenGL / OpenGL ES using the local built driver was trickier
- Required an X server running, and using it
- Required building X and some dependencies ourselves, and starting the server manually

RPI 5 – finally out

- 28 September 2023: Public announcement, MR created on Upstream Mesa
- 14 October 2023: code merged
- 23 October 2023: available to buy
- 2 November 2023: Vulkan 1.2 and OpenGL ES 3.1 Khronos conformant.

RPI 5 – current work

- Maintenance work
  - Handle issues created by users
- Keep adding features
  - Really near of Vulkan 1.3
- Keep working on performance work

Q&A

Join Us!

https://www.igalia.com/jobs