Status of Broadcom's vc4 and vc5 Drivers

Eric Anholt

2017-09-21

Quick vc4 history

- Broadcom released BSD-licensed 21553 vc4 driver and spec in February 2014
- Started Mesa/DRM vc4 project in June 2014
- Merged to Mesa in August, kernel in October

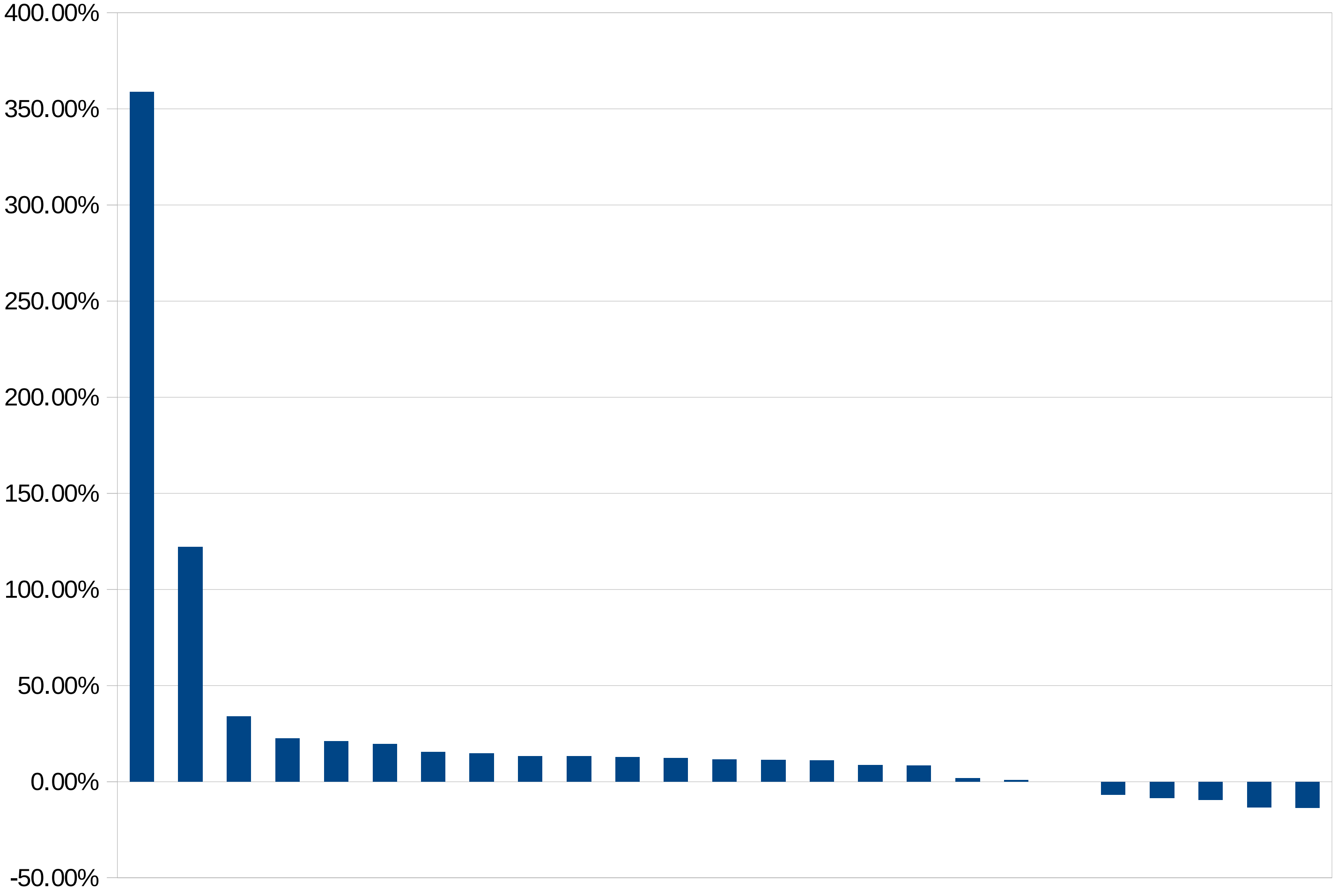
3D support

- Exposes GLESv2 and GL 2.1
- Supports BCM283x (Raspberry Pi) and Cygnus
- Does not support Capri, other V3D 2.x platforms

3D difficulties

- No MMU means shader and command list validation for security
- Missing compatibility features for GL 2.1
- Loops are hard and slightly broken (dEQP failures)
- No register spilling means your allocator had better be good (dEQP failures)

3D performance: glmark2 on vc4 vs closed stack



VC4 Display

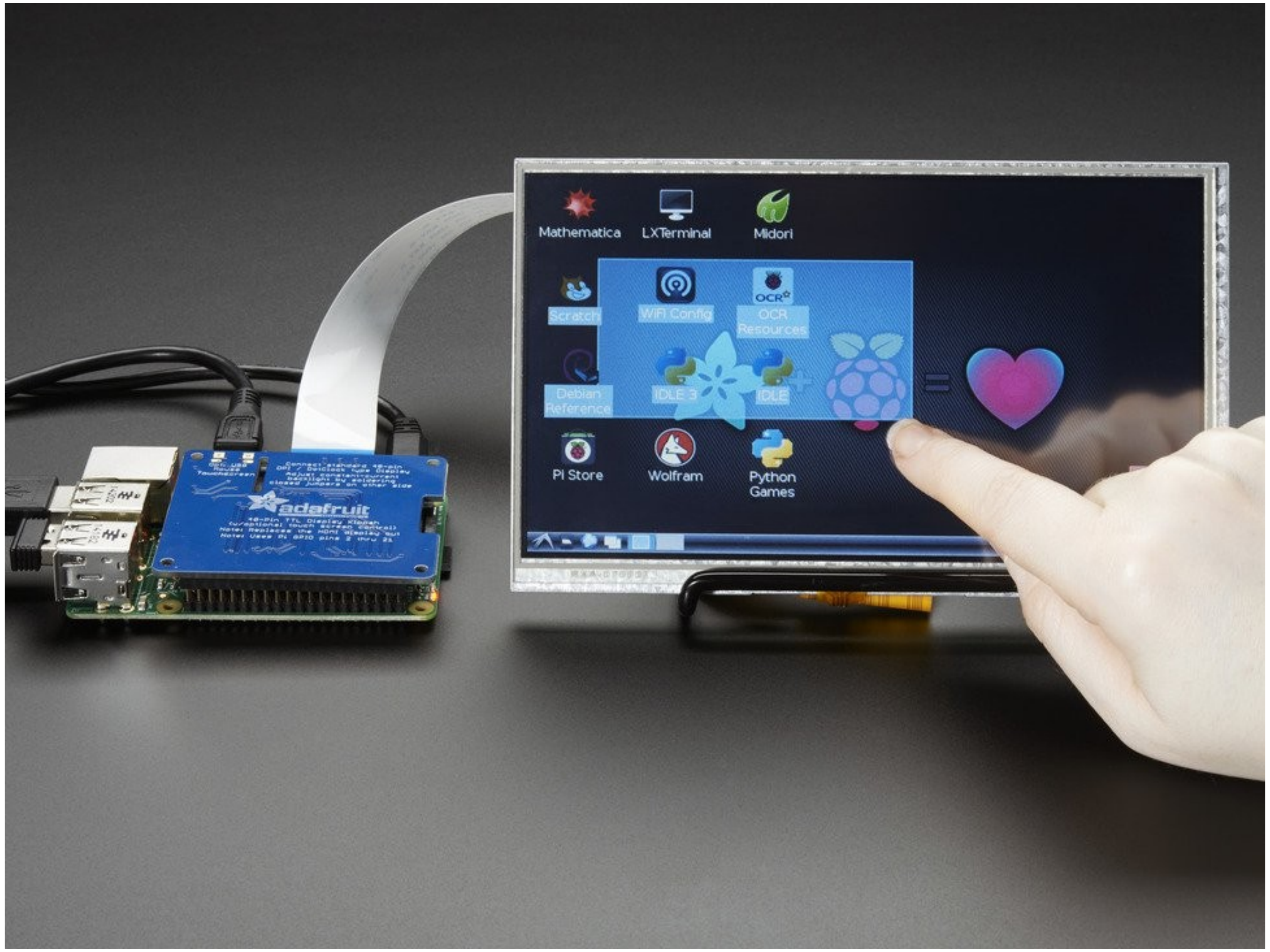
Display support: HDMI

- HDMI Audio
- CEC
- Power management

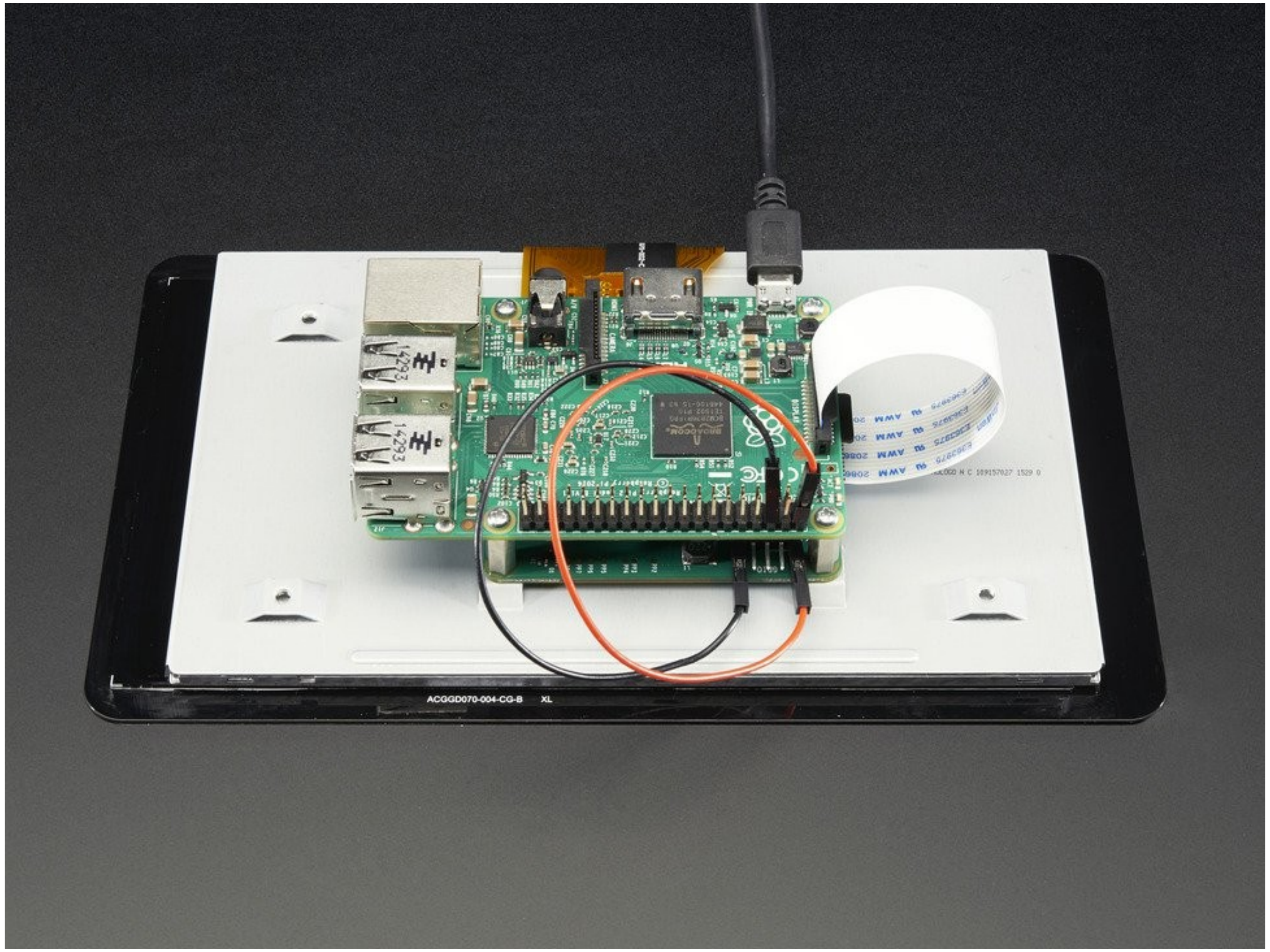
Display support: SDTV

- PAL and NTSC modes

Display support: DPI

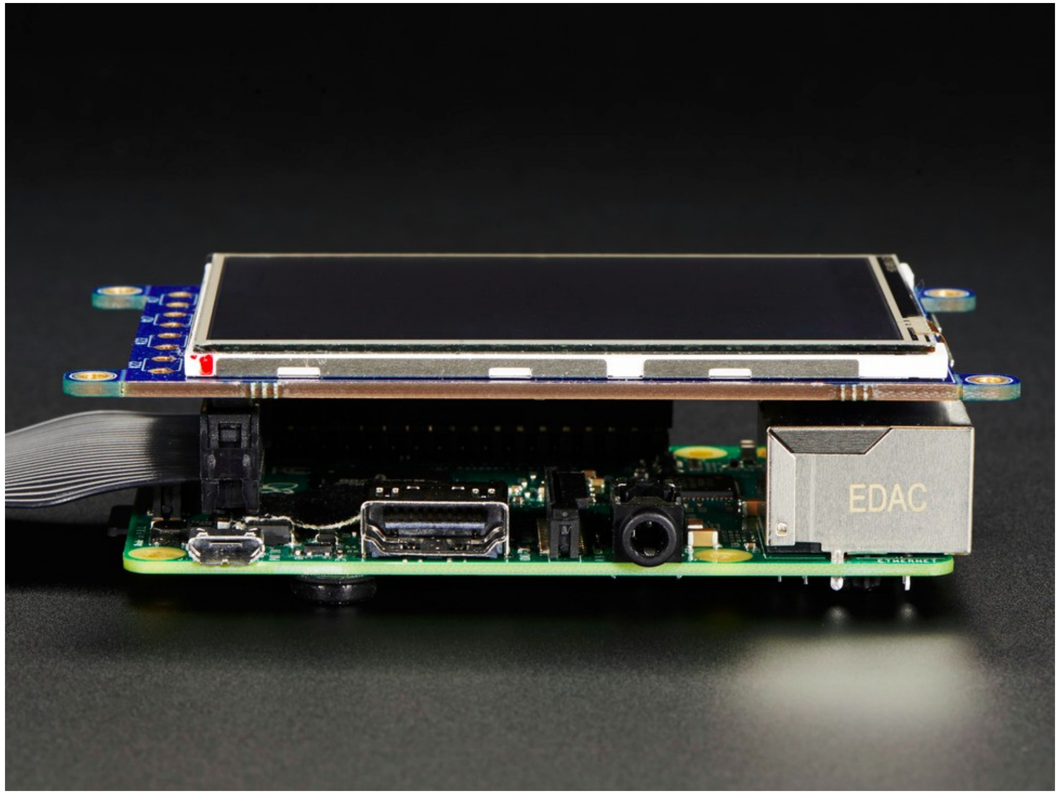


Display support: DSI



Display support: SPI?

- Not directly part of the VC4 display stack
- Supported by Noralf's tinydrm
- Build a vc4+tinydrm renderonly driver?
- DMA directly from VC4 HVS to SPI?



Display support: HVS (Plane compositing)

- Supported by DRM atomic interfaces
- No fixed limit, just structures in a bit of RAM
- Scaling, rotation, YUV conversion, blending
- Currently exposing 8 planes

Display support: Future work

- Use planes in X11
- Support SAND-modifier YUV planes from media decode
- Rotation
- Upstream 7” DSI panel driver
- Writeback connector
- Plane resource limit calculations (pixels/second and memory bandwidth)

VC4 status in distributions

- Fedora supports Raspberry Pi 2/3 with gnome-shell
  - Full KMS driver implementation, vc4 3d driver
- Raspbian has 3 options:
  - Closed-source GL and display stack on fbdev (default)
  - Full KMS implementation with vc4
  - Closed source display stack but vc4 3D
- Debian
  - Kernel and userspace have vc4 support
  - No official installer with RPi support
- Others
  - Generally closed source GL and display stack

Getting Raspbian switched to vc4

- firmwarekms mode is the first target
  - Keeps existing config.txt settings working, just adds vc4 3D
  - CMA memory management is a struggle
- Full KMS mode is the eventual goal
  - Opens the possibility of eliminating most of the closed source firmware
  - Needs equally reliable HDMI
  - Needs i2c input driver for DSI panel
  - Needs overlays and panel drivers for misc DPI panels

VC5

What’s new in VC5

- V3D 3.3 in BCM7268 (Set top box platform)
- GLES 3.1 and Vulkan HW support
- MMU means we no longer need CMA!
- GMP lets us mask buffers between clients
- Better shader ISA (no more register banks!)
- So much FP16

VC5 kernel interface

- Client gets an offset within the MMU at buffer create/import time
- Client gets uses that offset however it feels like in command lists
- Command list submission takes a list of buffers that must be in the MMU, without relocation information
- Shared 4MB (contiguous) MMU page table
- 8KB GMP table per client for which of the buffers in the MMU they can see
- ABI still unstable
  - need to add explicit fence support
  - need to add TFU (texture upload/download unit)

VC5 OpenGL

- Currently building a gallium driver for vc5
- Uses genxml based on the i965 driver for emitting state
  - XML describes hardware structs to generate C pack/unpack code
  - Prepack state structs at CSO creation time
  - OR together structs from CSOs at draw time
- Compiles NIR to VC5 QPU instructions
  - Based on vc4 compiler
  - Replaced most of the IR with unpacked QPU instruction
  - Minimal FP16 support so far
- Almost ready to merge

VC5 Vulkan (“bcmv”)

- Started in early September
- Forking from the anv driver, using some shared code from radv as well
- Uses shared compiler with vc5 gallium driver
- Separate surface layout
- No blitter library yet
- Not even linking, be patient

Thank You

BROADCOM®