

orphan:

MSM Crash Dump Format

Following a GPU hang the MSM driver outputs debugging information via `/sys/kernel/dri/X/show` or via `devcoredump (/sys/class/devcoredump/dcdX/data)`. This document describes how the output is formatted.

Each entry is in the form `key: value`. Sections headers will not have a value and all the contents of a section will be indented two spaces from the header. Each section might have multiple array entries the start of which is designated by a `(-)`.

Mappings

kernel

The kernel version that generated the dump (`UTS_RELEASE`).

module

The module that generated the crashdump.

time

The kernel time at crash formatted as `seconds.microseconds`.

comm

Comm string for the binary that generated the fault.

cmdline

Command line for the binary that generated the fault.

revision

ID of the GPU that generated the crash formatted as `core.major.minor.patchlevel` separated by dots.

rbbm-status

The current value of `RBBM_STATUS` which shows what top level GPU components are in use at the time of crash.

ringbuffer

Section containing the contents of each ringbuffer. Each ringbuffer is identified with an id number.

id

Ringbuffer ID (0 based index). Each ringbuffer in the section will have its own unique id.

iova

GPU address of the ringbuffer.

last-fence

The last fence that was issued on the ringbuffer

retired-fence

The last fence retired on the ringbuffer.

rptr

The current read pointer (rptr) for the ringbuffer.

wptr

The current write pointer (wptr) for the ringbuffer.

size

Maximum size of the ringbuffer programmed in the hardware.

data

The contents of the ring encoded as `ascii85`. Only the used portions of the ring will be printed.

bo

List of buffers from the hanging submission if available. Each buffer object will have a unique iova.

iova

GPU address of the buffer object.

size

Allocated size of the buffer object.

data

The contents of the buffer object encoded with `ascii85`. Only Trailing zeros at the end of the buffer will be skipped.

registers

Set of registers values. Each entry is on its own line enclosed by brackets `{ }`.

offset

Byte offset of the register from the start of the GPU memory region.

value

Hexadecimal value of the register.

registers-hlsq

(5xx only) Register values from the HLSQ aperture. Same format as the register section.