Status Update of the VKMS DRM driver

Maíra Canal and Melissa Wen

A Coruña, XDC 2023

igalia

Virtual Kernel Mode Setting

VKMS is a software-only model of a KMS driver

- useful for testing and for running X (or similar) on headless machines;
- aims to enable a virtual display with no need of a hardware display capability;

DRM community project. GPU-vendor agnostic.

Performance Improvement

- Performance has been a historical problem for VKMS
- We were able to improve the performance (~30%)
  - All planes are blended at once.
  - The blend calculus is done as per line instead of per pixel.
- Memory consumption also improved: from *width * height* to *2 * width*
  - We don't need the whole image at once.
- No more IGT timeouts :)
- Thanks Igor Torrente and Pekka Paalanen!

Composition

- The composition was executed on top of the primary plane.
  - The primary plane needed to be enabled, visible and have the same size and position as the CRTC.
- Now, the composition is performed on top of a black background with the size of the CRTC.
  - The primary plane doesn't have to be visible, or even enabled.
  - It can be positioned.
- Full alpha-blending on all planes.
- More IGT tests are passing!

New features

- Support for multiple overlay planes when *enable_overlay=1*
  - Thanks José Exposito!
- VKMS used to support only ARGB8888 and XRGB8888.
  - Format support was really attached to blending.
- Now, VKMS has support for ARGB8888, XRGB8888, ARGB16161616, XRGB16161616, and RGB565.
  - Thanks Igor Torrente!

New features

- Support for plane rotation
  - *rotate-0*, *rotate-90*, *rotate-180*, *rotate-270*, *reflect-x* and *reflect-y.*
- Support a 1D gamma LUT with interpolation for each color channel.
  - Thanks Arthur Grillo!
- More IGT tests are passing!

Ongoing

- Synchronization issues
  - Deadlock between *dev->event_lock* and *timer*
  - Race-condition between flushing and destroying
  - Race-condition between the hrtimer and the atomic commit
- Implement blend mode properties
- Enable Virtual Hardware support
- Support for multi-planar formats (e.g. NV12)
  - Thanks Arthur Grillo!

Incoming work

- Support for multiple CRTCs
  - Thanks Marius Vlad, Brandon Pollack and Jim Shargo!
- Planes / CRTCs / Connectors setup via ConfigFS
  - Thanks Brandon Pollack and Jim Shargo!
- Addition of VKMS to the DRM CI pipeline
  - Thanks Helen Koike!
- Extending the color management support
  - Thanks Harry Wentland!

TO-DO

- https://dri.freedesktop.org/docs/drm/gpu/vkms.html#todo
- Unit tests for the composition procedure
- Fix failing tests on IGT detected by the DRM CI

VKMS ❤ DRM/KMS API

Testing Requirements for Userspace API

Anything that constitutes an API change should have driver-agnostic test cases in IGT for that feature.

What if support on VKMS became a requirement for new KMS/generic APIs?

Do you want to contribute?

- Getting started: https://docs.kernel.org/gpu/vkms.html
- We need more reviewers!
- Increase coverage of IGT tests


igalia