



Contribution ID: 63

Type: **not specified**

## Standardising Linux DRM drivers implementations by interfacing DRM Bridge as a single API

*Tuesday, 14 November 2023 09:30 (45 minutes)*

Display and graphic drivers in Linux are part of the Linux DRM subsystem and are using DRM resources like memory management, interrupt handling, and DMA via Kernel Mode Settings (KMS) that act as an abstraction layer to provide uniform access to applications.

Encoders are one of the key KMS components that takes pixel data from a CRTC and converts it to a format suitable for an attached connector. Early Linux-4.0 encoders play a crucial role in connecting display hardware attributes to KMS however, for new-age display solutions like Bridges, Converters, and Switches it becomes hard for the encoder to handle these topologies in order to support various functionalities.

Linux-v4.0 has introduced a DRM bridge, a linear link structure of objects always attached to a single encoder at a time and connected directly or chain of bridges to the connector of a given KMS pipeline. Supporting bridge chains to new-age display solutions will end up with an encoder becoming a dumb encoder without any operations limited.

This progressive change of moving the DRM drivers from the encoder to the bridge will standardize the single API, So it becomes simple and clear for implementing a drm drivers for new-age display solutions without touching the existing KMS pipeline.

This talk explains how the DRM drivers are converted from encoder to bridge in order to standardize a single API by considering real-time solutions submitted to Mainline Linux on Samsung DSIM IP with a conclusion that are encoders replaced or removed from the drm stack?

**Primary author:** Mr TEKI, Jagan (U-Boot Allwinner, SPI, SPI Flash Maintainer, Linux DRM Bridge Contributor and Maintainer, AI/ML-enthusiast)

**Presenter:** Mr TEKI, Jagan (U-Boot Allwinner, SPI, SPI Flash Maintainer, Linux DRM Bridge Contributor and Maintainer, AI/ML-enthusiast)

**Session Classification:** LPC Refereed Track

**Track Classification:** LPC Refereed Track