NAME

modesetting − video driver for framebuffer device

SYNOPSIS

Section "Device"
Identifier "*devname*"
Driver "modesetting"
BusID "pci:*bus***:***dev***:** *func*"
...
EndSection

DESCRIPTION

modesetting is an Xorg driver for KMS devices. This driver supports TrueColor visuals at framebuffer depths of 15, 16, 24, and 30. RandR 1.2 is supported for multi-head configurations. Acceleration is available through glamor for devices supporting at least OpenGL ES 2.0 or OpenGL 2.1. If glamor is not enabled, a shadow framebuffer is configured based on the KMS drivers' preference (unless the framebuffer is 24 bits per pixel, in which case the shadow framebuffer is always used).

SUPPORTED HARDWARE

The **modesetting** driver supports all hardware where a KMS driver is available. modesetting uses the Linux DRM KMS ioctls and dumb object create/map.

CONFIGURATION DETAILS

Please refer to xorg.conf(5) for general configuration details. This section only covers configuration details specific to this driver.

For this driver it is not required to specify modes in the screen section of the config file. The **modesetting** driver can pick up the currently used video mode from the kernel driver and will use it if there are no video modes configured.

For PCI boards you might have to add a BusID line to the Device section. See above for a sample line.

The following driver **Options** are supported:

Option "SWcursor" "*boolean*"
Selects software cursor. The default is **off.**

Option "kmsdev" "*string*"
The framebuffer device to use. Default: /dev/dri/card0.

Option "ShadowFB" "*boolean*"
Enable or disable use of the shadow framebuffer layer. Default: on.

Option "DoubleShadow" "*boolean*"
Double-buffer shadow updates. When enabled, the driver will keep two copies of the shadow framebuffer. When the shadow framebuffer is flushed, the old and new versions of the shadow are compared, and only tiles that have actually changed are uploaded to the device. This is an optimization for server-class GPUs with a remote display function (typically VNC), where remote updates are triggered by any framebuffer write, so minimizing the amount of data uploaded is crucial. This defaults to enabled for ASPEED and Matrox G200 devices, and disabled otherwise.

Option "AccelMethod" "*string*"
One of "glamor" or "none". Default: glamor.

Option "PageFlip" "*boolean*"
Enable DRI3 page flipping. The default is **on.**

Option "VariableRefresh" "*boolean*"
Enables support for enabling variable refresh on the Screen's CRTCs when an suitable application is flipping via the Present extension.
The default is **off.**

Option "AsyncFlipSecondaries" "*boolean***"**
Use async flips for secondary video outputs on multi-display setups. If a screen has multiple displays attached and DRI3 page flipping is used, then only one of the displays will have its page flip synchronized to vblank for tear-free presentation. This is the display that is used for presentation timing and timestamping, usually the one covering the biggest pixel area of the screen. All other displays ("Secondaries") will not synchronize their flips. This may cause some tearing on these displays, but it prevents a permanent or periodic slowdown or irritating judder of animations if not all video outputs are running synchronized with each other and with the same refresh rate. There is no perfect solution apart from perfectly synchronized outputs, but this option may give preferrable results if the displays in a multi-display setup mirror or clone each other. The default is **off.**

Option "ZaphodHeads" "*string***"**
Specify the RandR output(s) to use with zaphod mode for a particular driver instance. If you use this option you must use this option for all instances of the driver.
For example: **Option "ZaphodHeads" "LVDS,VGA-0"** will assign xrandr outputs LVDS and VGA-0 to this instance of the driver.

Option "UseGammaLUT" "*boolean***"**
Enable or disable use of the GAMMA_LUT property, when available. When enabled, this option allows the driver to use gamma ramps with more entries, if supported by the kernel. By default, GAMMA_LUT will be used for kms drivers which are known to be safe for use of GAMMA_LUT.

SEE ALSO
Xorg(1), xorg.conf(5), Xserver(1), X(7)

AUTHORS
Authors include: Dave Airlie