NAME

openchrome − video driver for VIA Unichromes

SYNOPSIS

Section "Device"
Identifier "*devname***"**
Driver "openchrome"
...
EndSection

DESCRIPTION

openchrome is an Xorg driver for VIA chipsets that have an integrated Unichrome graphics engine.

The **openchrome** driver supports the following chipsets: CLE266, KM400/KN400/KM400A/P4M800, CN400/PM800/PN800/PM880, K8M800, CN700/VM800/P4M800Pro, CX700, P4M890, K8M890, P4M900/VN896/CN896, VX800, VX855 and VX900. The driver includes 2D acceleration and Xv video overlay extensions. Flat panel, TV, and VGA outputs are supported, depending on the hardware configuration.

3D direct rendering is available using experimental drivers from Mesa (www.mesa3d.org). There is also an XvMC client library for hardware acceleration of MPEG1/MPEG2 decoding (not available on the KM/N400) that uses the Direct Rendering Infrastructure (DRI). The XvMC client library implements a non-standard "VLD" extension to the XvMC standard. The current Direct Rendering Manager (DRM) kernel module is available at dri.sourceforge.net.

The driver supports free modes for Unichrome Pros (K8M800/K8N800, PM800/PN800, and CN400). For plain Unichromes (CLE266, KM400/KN400), it currently supports only a limited number of dotclocks, so if you are using X modelines you must make sure that the dotclock is one of those supported. Supported dotclocks on plain Unichromes are currently (in MHz): 25.2, 25.312, 26.591, 31.5, 31.704, 32.663, 33.750, 35.5, 36.0, 39.822, 40.0, 41.164, 46.981, 49.5, 50.0, 56.3, 57.284, 64.995, 65.0, 65.028, 74.480, 75.0, 78.8, 81.613, 94.5, 108.0, 108.28, 122.0, 122.726, 135.0, 148.5, 155.8, 157.5, 161.793, 162.0, 175.5, 189.0, 202.5, 204.8, 218.3, 229.5. On top of this, bandwidth restrictions apply for both Unichromes and Unichrome Pros.

CONFIGURATION DETAILS

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

The following driver **options** are supported:

Option "AccelMethod" "*string***"**
The driver supports "XAA" and "EXA" acceleration methods. The default method is XAA, since EXA is still experimental. Contrary to XAA, EXA implements acceleration for screen uploads and downloads (if DRI is enabled) and for the Render/Composite extension.

Option "ActiveDevice" "*string***"**
Specifies the active device combination. Any string containing "CRT", "LCD", "DFP", "TV" should be possible. "CRT" represents anything that is connected to the VGA port, "LCD" is for laptop panels (not TFT screens attached to the VGA port), "DFP" is for screens connected to the DVI port, "TV" is self-explanatory. The default is to use what is detected. The driver is currently unable to use LCD and TV simultaneously, and will favour the LCD. The DVI port is not properly probed and needs to be enabled with this option.

Option "AGPMem" "*integer***"**
Sets the amount of AGP memory that is allocated at X server startup. The allocated memory will be "integer" kB. This AGP memory is used for the AGP command buffer (if the option "EnableAGPDMA" is set to "true"), for DRI textures, and for the EXA scratch area. The driver will allocate at least one system page of AGP memory, or −− if the AGP command buffer is used −− at least 2 MB plus one system page. If there is no room for the EXA scratch area in AGP

space, it will be allocated from VRAM. If there is no room for DRI textures, they will be allocated from the DRI part of VRAM (see the option "MaxDRIMem"). The default amount of AGP is 32768 kB. Note that the AGP aperture set in the BIOS must be able to accommodate the amount of AGP memory specified here. Otherwise no AGP memory will be available. It is safe to set a very large AGP aperture in the BIOS.

Option "Center" "*boolean*"
Enables image centering on DVI displays. The default is disabled.

Option "DisableIRQ" "*boolean*"
Disables the vertical blank IRQ. This is a workaround for some mainboards that have problems with IRQs coming from the Unichrome engine. With IRQs disabled, DRI clients have no way to synchronize their drawing to Vblank. (IRQ is disabled by default on the KM400 and K8M800 chipsets.)

Option "DisableVQ" "*boolean*"
Disables the use of the virtual command queue. The queue is enabled by default.

Option "EnableAGPDMA" "*boolean*"
Enables the AGP DMA functionality in DRM. This requires that DRI is enabled and will force 2D and 3D acceleration to use AGP DMA. The XvMC DRI client will also make use of this on the CLE266 to consume much less CPU. (This option is enabled by default, except on the K8M890 and P4M900.)

Option "ExaNoComposite" "*boolean*"
If EXA is enabled (using the option "AccelMethod"), this option enables acceleration of compositing. Since EXA, and in particular its composite acceleration, is still experimental, this is a way to disable a misbehaving composite acceleration.

Option "ExaScratchSize" "*integer*"
Sets the size of the EXA scratch area to "integer" kB. This area is used by EXA as a last place to look for available space for pixmaps. Too little space will slow compositing down. This option should be set to the size of the largest pixmap used. If you have a screen width of over 1024 pixels and use 24 bpp, set this to 8192. Otherwise you can leave this at the default 4096. The space will be allocated from AGP memory if available, otherwise from VRAM.

Option "LCDDualEdge" "*boolean*"
Enables the use of dual-edge mode to set the LCD. The default is disabled.

Option "MaxDRIMem" "*integer*"
Sets the maximum amount of VRAM memory allocated for DRI clients to "integer" kB. Normally DRI clients get half the available VRAM size, but in some cases it may make sense to limit this amount. For example, if you are using a composite manager and you want to give as much memory as possible to the EXA pixmap storage area.

Option "MigrationHeuristic" "*string*"
Sets the heuristic for EXA pixmap migration. This is an EXA core option, and starting from Xorg server version 1.3.0 this defaults to "always". The openchrome driver performs best with "greedy", so you should really add this option to your configuration file. The third possibility is "smart".

Option "NoAccel" "*boolean*"
Disables the use of hardware acceleration. Acceleration is enabled by default.

Option "NoAGPFor2D" "*boolean*"
Disables the use of AGP DMA for 2D acceleration, even when AGP DMA is enabled. The default is enabled.

Option "NoXVDMA" "*boolean*"
If DRI is enabled, Xv normally uses PCI DMA to transfer video images from system to framebuffer memory. This is somewhat slower than direct copies due to the limitations of the PCI bus, but on the other hand it decreases CPU usage significantly, particularly on computers with fast

processors. Some video players are buggy and will display rendering artifacts when PCI DMA is used. If you experience this, or don't want your PCI bus to be stressed with Xv images, set this option to "true". This option has no effect when DRI is not enabled.

Option "PanelSize" "*string*"
Specifies the size (width x height) of the LCD panel attached to the system. The sizes 640x480, 800x600, 1024x768, 1280x1024, and 1400x1050 are supported.

Option "RotationType" "*string*"
Enabled rotation by using RandR. The driver only support unaccelerated RandR rotations "SWRandR". Hardware rotations "HWRandR" is currently unimplemented.

Option "Rotate" "*string*"
Rotates the display either clockwise ("CW"), counterclockwise ("CCW") and upside-down ("UD"). Rotation is only supported unaccelerated. Adding option "Rotate", enables RandR rotation feature. The RandR allows clients to dynamically change X screens.

Option "ShadowFB" "*boolean*"
Enables the use of a shadow frame buffer. This is required when rotating the display, but otherwise defaults to disabled.

Option "SWCursor" "*boolean*"
Enables the use of a software cursor. The default is disabled: the hardware cursor is used.

Option "TVDeflicker" "*integer*"
Specifies the deflicker setting for TV output. Valid values are "0", "1", and "2". Here 0 means no deflicker, 1 means 1:1:1 deflicker, and 2 means 1:2:1 deflicker.

Option "TVDotCrawl" "*boolean*"
Enables dot-crawl suppression. The default is disabled.

Option "TVOutput" "*string*"
Specifies which TV output to use. The driver supports "S-Video", "Composite", "SC", "RGB", and "YCbCr" outputs. Note that on some EPIA boards the composite-video port is shared with audio-out and is selected via a jumper.

Option "TVPort" "*string*"
Specifies TV port. The driver currently supports "DVP0", "DVP1", "DFPHigh" and "DFPLow" ports.

Option "TVType" "*string*"
Specifies TV output format. The driver currently supports "NTSC" and "PAL" timings only.

Option "VBEModes" "*boolean*"
Enables the use of VBE BIOS calls for setting the display mode. This mimics the behaviour of the vesa driver but still provides acceleration and other features. This option may be used if your hardware works with the vesa driver but not with the openchrome driver. It may not work on 64-bit systems. Using "VBEModes" may speed up driver acceleration significantly due to a more aggressive hardware setting, particularly on systems with low memory bandwidth. Your refresh rate may be limited to 60 Hz on some systems.

Option "VBESaveRestore" "*boolean*"
Enables the use of VBE BIOS calls for saving and restoring the display state when the X server is launched. This can be extremely slow on some hardware, and the system may appear to have locked for 10 seconds or so. The default is to use the driver builtin function. This option only works if option "VBEModes" is enabled.

Option "VideoRAM" "*integer*"
Overrides the VideoRAM autodetection. This should never be needed.

TV ENCODERS

Unichromes tend to be paired with several different TV encoders.

VIA Technologies VT1621

Still untested, as no combination with a Unichrome is known or available. Supports the following normal modes: "640x480" and "800x600". Use "640x480Over" and "800x600Over" for vertical overscan. These modes are made available by the driver; modelines provided in xorg.conf will be ignored.

VIA Technologies VT1622, VT1622A, VT1623

Supports the following modes: "640x480", "800x600", "1024x768", "848x480", "720x480" (NTSC only) and "720x576" (PAL only). Use "640x480Over", "800x600Over", "1024x768Over", "848x480Over", "720x480Over" (NTSC) and "720x576Over" (PAL) for vertical overscan. The modes "720x480Noscale" (NTSC) and "720x576Noscale" (PAL) (available on VT1622 only) provide cleaner TV output (unscaled with only minimal overscan). These modes are made available by the driver; modelines provided in xorg.conf will be ignored.

SEE ALSO

Xorg(1), xorg.conf(5), Xserver(1), X(7), EXA(5), Xv(5)

AUTHORS

Authors include: ...