

NAME

edid-decode - Decode EDID data in human-readable format

SYNOPSIS

edid-decode <options> [**in** [**out**]]

DESCRIPTION

edid-decode decodes EDID monitor description data in human-readable format. If [**in**] is not given, or [**in**] is '-', then the EDID will be read from standard input. If [**out**] is given then the EDID that was read from [**in**] is written to [**out**] or to standard output if [**out**] is '-'. By default the output is written as a hex dump when writing to standard output or a raw EDID if written to a file.

If [**out**] is given then edid-decode only does the conversion, it will skip the decoding step.

Input files may be raw binaries or ASCII text. ASCII input is scanned for hex dumps; heuristics are included to search for hexdumps in **edid-decode(1)** output (as long as the initial hex dump was included), **xrandr(1)** property output and **Xorg(1)** log file formats, otherwise the data is treated as a raw hexdump. EDID blocks for connected monitors can be found in `/sys/class/drm/*/edid` on modern Linux systems with kernel modesetting support.

All timings are shown in a short format, for example:

```
VIC 16: 1920x1080 60.000 Hz 16:9 67.500 kHz 148.500 MHz (native)
VIC 5: 1920x1080i 60.000 Hz 16:9 33.750 kHz 74.250 MHz
VIC 39: 1920x1080i 50.000 Hz 16:9 31.250 kHz 72.000 MHz
```

Each format starts with a timings type prefix, the resolution, an optional interlaced indicator ('i'), the frame rate (field rate for interlaced formats), the picture aspect ratio, the horizontal frequency, the pixelclock frequency and optionally additional flags between parenthesis.

Note that for interlaced formats the frame height is given, not the field height. So each field in a 1920x1080i format has 540 lines.

Detailed timings have another 2-3 lines of data:

```
VIC 16: 1920x1080 60.000 Hz 16:9 67.500 kHz 148.500 MHz (native)
      Hfront 88 Hsync 44 Hback 148 Hpol P
      Vfront 4 Vsync 5 Vback 36 Vpol P
VIC 5: 1920x1080i 60.000 Hz 16:9 33.750 kHz 74.250 MHz
      Hfront 88 Hsync 44 Hback 148 Hpol P
      Vfront 2 Vsync 5 Vback 15 Vpol P Vfront +0.5 Odd Field
      Vfront 2 Vsync 5 Vback 15 Vpol P Vback +0.5 Even Field
VIC 39: 1920x1080i 50.000 Hz 16:9 31.250 kHz 72.000 MHz
      Hfront 32 Hsync 168 Hback 184 Hpol P
      Vfront 23 Vsync 5 Vback 57 Vpol N Both Fields
```

These describe the horizontal and vertical front porch, sync, backporch and sync polarity values. For interlaced formats there are two lines for the vertical information: one for the Odd Field (aka Field 1) and one for the Even Field (aka Field 2). The vertical front porch of the Odd Field is actually 2.5 (hence the 'Vfront +0.5' at the end of the line), and the back porch of the Even Field is actually 15.5 (hence the 'Vback +0.5' at the end of the line).

There is a special 'VIC 39' interlaced format where both fields have the same vertical timings, in that case this is marked with 'Both Fields'.

The following timing types can be shown:

DMT #: Discrete Monitor Timing (see DMT 1.3 standard). The number is the DMT ID in hexadecimal.

CVT: Coordinated Video Timings (formula-based, see CVT 1.2 standard)

GTF: Generalized Timing Formula (formula-based, see GTF 1.1 standard)

IBM: Old IBM Timings

Apple: Old Apple Timings

VIC #: Video Identification Code (see CTA-861 standard). The number is the actual VIC code.

HDMI VIC #: HDMI-specific Video Identification Code (see HDMI 2.1 standard). The number is the actual HDMI VIC code.

DTD #: Detailed Timings Descriptor (see EDID standard). Also used for DisplayID Video Timing Modes Types I, II, VI, VII, VIII and X. The number denotes that this is the Nth DTD in the Base Block and CTA Extension Blocks.

VTDB #: 20-byte DTD or 6- or 7-byte CVT descriptor in a CTA Extension Block. The number denotes that this is the Nth such timing in the CTA Extension Blocks.

RID #@#: A CTA-861.6 Video Format Descriptor with the given Resolution ID (first number) at the given framerate (second number).

By default DTDs are shown in the long format while others are just shown in the short format. With the option **--short-timings** all timings are shown in short format only. With the option **--long-timings** all timings are shown in long format.

Alternate formats for long timings can be chosen via the **--xmodeline** or **--fbmode** options.

STANDARDS

The following EDID standards are supported by edid-decode:

EDID 1.3: VESA Enhanced Extended Display Identification Data Standard, Release A, Revision 1

EDID 1.4: VESA Enhanced Extended Display Identification Data Standard, Release A, Revision 2

DisplayID 1.3: VESA Display Identification Data (DisplayID) Standard, Version 1.3

DisplayID 2.1a: VESA DisplayID Standard, Version 2.1a

DI-EXT: VESA Display Information Extension Block Standard, Release A

LS-EXT: VESA Enhanced EDID Localized String Extension Standard, Release A

VTB-EXT: VESA Video Timing Block Extension Data Standard, Release A

DTCDB: VESA Display Transfer Characteristics Data Block Standard, Version 1.0

DDDB: VESA Display Device Data Block (DDDB) Standard, Version 1

HDMI 1.4b: High-Definition Multimedia Interface, Version 1.4b

HDMI 2.1b: High-Definition Multimedia Interface, Version 2.1b

HDCP 1.4: High-bandwidth Digital Content Protection System, Revision 1.4

HDCP 2.3: High-bandwidth Digital Content Protection System, Mapping HDCP to HDMI, Revision 2.3

CTA-861-I: A DTV Profile for Uncompressed High Speed Digital Interfaces

CTA-861.7: Improvements to CTA-861-I

SPWG Notebook Panel Specification, Version 3.5

EPI Embedded Panel Interface, Revision 1.0

Microsoft EDID extension for head-mounted and specialized monitors, Version 3

The following related standards are also used by edid-decode:

DMT 1.3: VESA and Industry Standards and Guidelines for Computer Display Monitor Timing (DMT), Version 1.0, Rev. 13

CVT 2.1: VESA Coordinated Video Timings (CVT) Standard, Version 2.1

CVT 1.2: VESA Coordinated Video Timings (CVT) Standard, Version 1.2

CVT 1.2: VESA CVT v1.2 Errata E2

GTF 1.1: VESA Generalized Timing Formula Standard, Version: 1.1

HDA 1.0a: High Definition Audio Specification, Version 1.0a

OPTIONS

-h, --help

Prints the help message.

-o, --output-format <fmt>

If [out] is specified, then write the EDID in format <fmt>.

The output format can be one of:

hex: hex numbers in ascii text (default for stdout)

raw: binary data (default unless writing to stdout)

carray: c-program struct

xml: XML data

-c, --check

Check if the EDID conforms to the standards. Warnings and failures are reported at the end.

-C, --check-inline

Check if the EDID conforms to the standards. Warnings and failures are reported as they happen.

-n, --native-resolution

Report the native resolution at the end. There may be multiple native resolution reports depending on whether the Source only parses Block 0 (e.g. DVI outputs) or Block 0 and the CTA-861 Extension Blocks (HDMI), or just the DisplayID Extension Blocks (typical for DisplayPort). If all blocks contain the same native resolution, then only that resolution is reported. For older displays there may be two separate native resolutions: progressive and interlaced.

-p, --preferred-timings

Report the preferred timings at the end. There may be multiple preferred timing reports depending on whether the Source only parses Block 0 (e.g. DVI outputs), or Block 0 and the CTA-861 Extension Blocks (HDMI), or Block 0 and the DisplayID Extension Blocks (typical for DisplayPort).

-I, --infoframe <file>

Parse the given InfoFrame file. Read data from stdin if '-' was passed as a filename. This option can be used multiple times to parse multiple InfoFrames. If the EDID of the display to which these InfoFrames are transmitted is also given, then the conformity checks will take that EDID into account.

If the first byte in the InfoFrame is 0x80 or higher, then it is assumed to be an InfoFrame that starts with the HDMI header and has a checksum, as per the HDMI Specification. Otherwise it is assumed to be a regular CTA-861 InfoFrame without a checksum.

Note: this is still work-in-progress, specifically for the AVI and HDMI InfoFrames.

-E, --eld *<file>*

Parse the given EDID-Like Data (ELD) file. This option can be used multiple times to parse multiple ELD files. Read data from stdin if '-' was passed as a filename. If the EDID of the display to which these ELD files are generated is also given, then the conformity checks will take that EDID into account.

On Linux systems ELD can be extracted via the amixer command (copy all hex after the 'values='):

```
$ amixer -c 0 controls | grep ELD
numid=6,iface=PCM,name='ELD',device=3
numid=12,iface=PCM,name='ELD',device=7
numid=18,iface=PCM,name='ELD',device=8
numid=24,iface=PCM,name='ELD',device=9
$ amixer -c 0 cget iface=PCM,name=ELD,device=3
numid=6,iface=PCM,name='ELD',device=3
; type=BYTES,access=r--v---,values=95
: values=0x10,0x00,0x08,0x00,0x6d,0x10,0x00,0x01,0x00,0x02,0x00,0x00,0x00,0x00,
0x00,0x00,0x30,0xae,0xf1,0x61,0x4c,0x45,0x4e,0x20,0x54,0x33,0x32,0x68,0x2d,0x32,
0x30,0x0a,0x20,0x09,0x7f,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
```

--diagonal *<inches>*

Specify the diagonal of the display in inches. This will enable additional checks for the image size, checking if it corresponds to the diagonal. This assumes square pixels.

-P, --physical-address

Just report the HDMI Source Physical Address and nothing else. Reports f.f.f.f if the EDID could not be parsed, or if there was no CTA-861 Vendor-Specific Data Block with OUI 00-0C-03. Otherwise it reports the Source Physical Address as provided in that Data Block. This can be used as input to HDMI CEC utilities such as the `linux cec-ctl(1)` utility.

-S, --short-timings

Report all video timings in a short format.

-L, --long-timings

Report all video timings in a long format.

-N, --ntsc

Report the video timings with values suitable for NTSC-based video. E.g., this will show refresh rates of 29.97 Hz instead of 30 Hz. This is only done for timings with refresh rates that are a multiple of 6.

-X, --xmodeline

Report all long video timings in the ModeLine format as defined in `xorg.conf(5)`. This ModeLine can be used in the `xorg.conf` file or passed to `xrandr(1)` with the `xrandr --newmode` option.

-F, --fbmode

Report all long video timings in the video mode format as defined in fb.modes(5).

-V, --v4l2-timings

Report all long video timings in the video mode format as defined in the linux header `v4l2-dv-timings.h` for use with the `V4L2 VIDIOC_S_DV_TIMINGS` ioctl.

-s, --skip-hex-dump

Skip the initial hex dump of the EDID.

-H, --only-hex-dump

Only show the hex dump of the EDID, then exit.

--skip-sha

Don't show the SHA hash. Normally edid-decode will show the SHA, i.e. the hash of the git commit used to compile edid-decode. This uniquely identifies the version of edid-decode that is used to generate the warnings and failures. But it will also change the output of edid-decode for every new commit in the git repository, even if nothing else changed in the edid-decode output. Use this option to avoid including the SHA in the edid-decode output.

-u, --utf8

Convert embedded EDID strings to UTF-8, using Code Page 437 for the base block and ISO 8859-1 for the DisplayID block.

--hide-serial-numbers

Hide any serial numbers in the human readable output by '...'. Note that they are still easily extracted from the EDID hex dump at the start.

--replace-unique-ids

Replaces any unique IDs in the EDID by fixed values. Serial numbers will be replaced by '123456', Container IDs by all zeroes and the 'Made in' date by the year 2000. This will also update any checksums in the EDID and update the EDID hex dump at the start of the output. Note that since this will update checksums, any checksum errors present in the original EDID will no longer be detected.

Serial numbers can appear in the Base Block, CTA-861 Extension Blocks, DisplayID Extension Blocks and Localized String Extension Blocks. Container IDs can appear in the DisplayID and CTA-861 Extension Blocks.

The 'Made in' date appears in the Base Block.

--version

Show the SHA hash and the last commit date.

I2C/DDC OPTIONS

The following options read the DDC bus directly, provided the DDC bus is exposed by linux to **/dev/i2c-X** as an i2c adapter device.

This can be used to read the EDID and HDCP information directly from the sink and parse it.

-a, --i2c-adapter <dev>

Use the **<dev>** device to access the DDC lines. If **<dev>** starts with a digit, then **/dev/i2c-<dev>** is used.

--i2c-edid

Read and parse the EDID from the i2c adapter.

--i2c-hdcp

Read and parse the HDCP data from the i2c adapter.

--i2c-hdcp-ri <t>

Every **<t>** seconds read and report the HDCP Ri value from the i2c adapter.

--i2c-test-reliability [duration=<secs>][,sleep=<msecs>]

Read the EDID continuously for **<secs>** seconds (default is 0, meaning forever), with a sleep of **<msecs>** milliseconds (default value is 50 ms) in between each read. Report a FAIL if there are mismatches between EDIDs. This tests the i2c communication towards the display.

TIMING OPTIONS

The following options report the timings for DMT, VIC and HDMI VIC codes and calculate the timings for CVT or GTF timings, based on the given parameters. The EDID will not be shown, although it can be used with the **--gtf** option in order to read the secondary curve parameters.

--std <byte1>,<byte2>

Show the standard timing represented by these two bytes.

--dmt <dmt>

Show the timings for the DMT with the given DMT ID.

--vic <vic>

Show the timings for this VIC.

--hdmi-vic <hdmivic>

Show the timings for this HDMI VIC.

--cvt w=<width>,h=<height>,fps=<fps>[,rb=<rb>][,interlaced][,overscan]
[,alt][,hblank=<hblank>][,vblank=<vblank>][,early-vsync]

Calculate the CVT timings for the given format.

<width> is the width in pixels, <height> is the frame (not field!) height in lines.

<fps> is frames per second for progressive timings and fields per second for interlaced timings.

<rb> can be 0 (no reduced blanking, default), or 1-3 for the reduced blanking version.

If **interlaced** is given, then this is an interlaced format.

If **overscan** is given, then this is an overscanned format. I.e., margins are required.

If **alt** is given and <rb>=2, then report the timings optimized for video: $1000 / 1001 * \text{<fps>}$.

If **alt** is given and <rb>=3, then the horizontal blanking is 160 instead of 80 pixels.

If **hblank** is given and <rb>=3, then the horizontal blanking is <hblank> pixels (range of 80-200 and divisible by 8), overriding **alt**.

If **vblank** is given and <rb>=3, then the vertical blanking time is <vblank> microseconds (460 minimum, values > 705 might not be supported by all RBv3 timings compliant source devices).

If **early-vsync** is given and <rb>=3, then select an early vsync timing.

--gtf w=<width>,h=<height>,fps=<fps>[,horfreq=<horfreq>][,pixclk=<pixclk>]
[,interlaced][,overscan][,secondary][,C=<c>][,M=<m>][,K=<k>][,J=<j>]

Calculate the GTF timings for the given format.

<width> is the width in pixels, <height> is the frame (not field!) height in lines.

<fps> is frames per second for progressive timings and fields per second for interlaced timings.

<horfreq> is the horizontal frequency in kHz.

<pixclk> is the pixel clock frequency in MHz. Only one of **fps**, **horfreq** or **pixclk** must be given.

If **interlaced** is given, then this is an interlaced format.

If **overscan** is given, then this is an overscanned format. I.e., margins are required.

If **secondary** is given, then the secondary GTF is used for reduced blanking, where <c>, <m>, <k> and <j> are parameters for the secondary curve. If none of the secondary curve parameters were set, and an EDID file is passed as command line option, then the secondary curve parameters are read from that EDID.

The default secondary curve parameters are 40 for <c>, 600 for <m>, 128 for <k> and 20 for <j>. These values correspond to the normal curve that GTF uses.

--ovt (rid=<rid>|w=<width>,h=<height>),fps=<fps>

Calculate the OVT timings for the given format. Either specify a <rid> or specify <width> and <height>. <fps> is frames per second.

--list-established-timings

List all known Established Timings.

--list-dmts

List all known DMTs.

--list-vics

List all known VICs.

--list-hdmi-vics

List all known HDMI VICs.

--list-rids

List all known CTA-861 RIDs.

--list-rid-timings <rid>

List all timings for the specified <rid> or all known RIDs if <rid> is 0.

NOTES

Not all fields are decoded, or decoded completely. **edid-decode** does attempt to validate its input against the relevant standards, but its opinions have not been double-checked with the relevant standards bodies, so they may be wrong. Do not rely on the output format, as it will likely change in future versions of the tool as additional fields and extensions are added.

SEE ALSO

Xorg(1), xrandr(1), cec-ctl(1), xorg.conf(5), fb.modes(5)

AUTHORS

edid-decode was written by Adam Jackson, with contributions from Eric Anholt, Damien Lespiau, Hans Verkuil and others.

Bug reports or questions about this utility should be sent to the linux-media@vger.kernel.org mailinglist.