ioctl VIDIOC_G_EDID, VIDIOC_S_EDID, VIDIOC_SUBDEV_G_EDID, VIDIOC_SUBDEV_S_EDID

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 2)

Unknown directive type "c:namespace".

```
.. c:namespace:: V4L
```

Name

VIDIOC_G_EDID - VIDIOC_S_EDID - VIDIOC_SUBDEV_G_EDID - VIDIOC_SUBDEV_S_EDID - Get or set the EDID of a video receiver/transmitter

Synopsis

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 18)

Unknown directive type "c:macro".

```
.. c:macro:: VIDIOC_G_EDID
```

```
int ioctl(int fd, VIDIOC_G_EDID, struct v4l2_edid *argp)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 22)

Unknown directive type "c:macro".

```
.. c:macro:: VIDIOC_S_EDID
```

```
int ioctl(int fd, VIDIOC_S_EDID, struct v4l2_edid *argp)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 26)

Unknown directive type "c:macro".

```
.. c:macro:: VIDIOC_SUBDEV_G_EDID
```

```
int ioctl(int fd, VIDIOC_SUBDEV_G_EDID, struct v4l2_edid *argp)
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 30)

Unknown directive type "c:macro".

```
.. c:macro:: VIDIOC_SUBDEV_S_EDID
```

```
int ioctl(int fd, VIDIOC_SUBDEV_S_EDID, struct v4l2_edid *argp)
```

Arguments

`fd`

File descriptor returned by :c:func:`open()`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 38); *backlink*

Unknown interpreted text role "c:func".

`argp`

Pointer to struct :c:type:`v4l2_edid`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 41); *backlink*

Unknown interpreted text role "c:type".

Description

These ioctls can be used to get or set an EDID associated with an input from a receiver or an output of a transmitter device. They can be used with subdevice nodes (/dev/v4l-subdevX) or with video nodes (/dev/videoX).

When used with video nodes the `pad` field represents the input (for video capture devices) or output (for video output devices) index as is returned by :ref:`VIDIOC_ENUMINPUT` and :ref:`VIDIOC_ENUMOUTPUT` respectively. When used with subdevice nodes the `pad` field represents the input or output pad of the subdevice. If there is no EDID support for the given `pad` value, then the `EINVAL` error code will be returned.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 51); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 51); *backlink*

Unknown interpreted text role "ref".

To get the EDID data the application has to fill in the `pad`, `start_block`, `blocks` and `edid` fields, zero the `reserved` array and call :ref:`VIDIOC_G_EDID <VIDIOC_G_EDID>`. The current EDID from block `start_block` and of size `blocks` will be placed in the memory `edid` points to. The `edid` pointer must point to memory at least `blocks` * 128 bytes large (the size of one block is 128 bytes).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 59); *backlink*

Unknown interpreted text role "ref".

If there are fewer blocks than specified, then the driver will set `blocks` to the actual number of blocks. If there are no EDID blocks available at all, then the error code `ENODATA` is set.

If blocks have to be retrieved from the sink, then this call will block until they have been read.

If `start_block` and `blocks` are both set to 0 when :ref:`VIDIOC_G_EDID <VIDIOC_G_EDID>` is called, then the driver will set `blocks` to the total number of available EDID blocks and it will return 0 without copying any data. This is an easy way to discover how many EDID blocks there are.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 73); *backlink*

Unknown interpreted text role "ref".

Note

If there are no EDID blocks available at all, then the driver will set `blocks` to 0 and it returns 0.

To set the EDID blocks of a receiver the application has to fill in the `pad`, `blocks` and `edid` fields, set `start_block` to 0 and zero the `reserved` array. It is not possible to set part of an EDID, it is always all or nothing. Setting the EDID data is only valid for receivers as it makes no sense for a transmitter.

The driver assumes that the full EDID is passed in. If there are more EDID blocks than the hardware can handle then the EDID is not written, but instead the error code `E2BIG` is set and `blocks` is set to the maximum that the hardware supports. If `start_block` is any value other than 0 then the error code `EINVAL` is set.

To disable an EDID you set `blocks` to 0. Depending on the hardware this will drive the hotplug pin low and/or block the source from reading the EDID data in some way. In any case, the end result is the same: the EDID is no longer available.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 101)

Unknown directive type "c:type".

```
.. c:type:: v4l2_edid
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 103)

Unknown directive type "tabularcolumns".

```
.. tabularcolumns:: |p{4.4cm}|p{4.4cm}|p{8.5cm}|
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 105)

Unknown directive type "flat-table".

```
.. flat-table:: struct v4l2_edid
    :header-rows:  0
    :stub-columns: 0
    :widths:       1 1 2

    * - __u32
      - ``pad``
      - Pad for which to get/set the EDID blocks. When used with a video
        device node the pad represents the input or output index as
        returned by :ref:`VIDIOC_ENUMINPUT` and
        :ref:`VIDIOC_ENUMOUTPUT` respectively.
    * - __u32
      - ``start_block``
      - Read the EDID from starting with this block. Must be 0 when
        setting the EDID.
    * - __u32
      - ``blocks``
      - The number of blocks to get or set. Must be less or equal to 256
        (the maximum number of blocks as defined by the standard). When
        you set the EDID and ``blocks`` is 0, then the EDID is disabled or
        erased.
    * - __u32
      - ``reserved``\ [5]
      - Reserved for future extensions. Applications and drivers must set
        the array to zero.
    * - __u8 *
      - ``edid``
      - Pointer to memory that contains the EDID. The minimum size is
        ``blocks`` * 128.
```

Return Value

On success 0 is returned, on error -1 and the `errno` variable is set appropriately. The generic error codes are described at the :ref:`Generic Error Codes <gen-errors>` chapter.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\(linux-master)(Documentation)(userspace-api)(media)(v4l)vidioc-g-edid.rst, line 138); *backlink*

Unknown interpreted text role "ref".

ENODATA
: The EDID data is not available.

E2BIG
: The EDID data you provided is more than the hardware can handle.