

# ioctl VIDIOC\_G\_FBUF, VIDIOC\_S\_FBUF

**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-fbuf.rst, line 2)**

Unknown directive type "c.namespace".

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

## Name

VIDIOC\_G\_FBUF - VIDIOC\_S\_FBUF - Get or set frame buffer overlay parameters

## 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-fbuf.rst, line 18)**

Unknown directive type "c.macro".

```
.. c:macro:: VIDIOC_G_FBUF
```

```
int ioctl(int fd, VIDIOC_G_FBUF, struct v4l2_framebuffer *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-fbuf.rst, line 22)**

Unknown directive type "c.macro".

```
.. c:macro:: VIDIOC_S_FBUF
```

```
int ioctl(int fd, VIDIOC_S_FBUF, const struct v4l2_framebuffer *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-fbuf.rst, line 30); [backlink](#)**

Unknown interpreted text role "c:func".

argp

Pointer to struct `:c:type:`v4l2_framebuffer``.

**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-fbuf.rst, line 33); [backlink](#)**

Unknown interpreted text role "c:type".

## Description

Applications can use the `:ref:`VIDIOC_G_FBUF <VIDIOC_G_FBUF>`` and `:ref:`VIDIOC_S_FBUF <VIDIOC_G_FBUF>`` `ioctl` to get and set the framebuffer parameters for a `:ref:`Video Overlay <overlay>`` or `:ref:`Video Output Overlay <osd>`` (OSD). The type of overlay is implied by the device type (capture or output device) and can be determined with the `:ref:`VIDIOC_QUERYCAP``

ioctl. One /dev/videoN device must not support both kinds of overlay.

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

Unknown interpreted text role "ref".

The V4L2 API distinguishes destructive and non-destructive overlays. A destructive overlay copies captured video images into the video memory of a graphics card. A non-destructive overlay blends video images into a VGA signal or graphics into a video signal. *Video Output Overlays* are always non-destructive.

To get the current parameters applications call the `ref:VIDIOC_G_FBUF <VIDIOC_G_FBUF>` ioctl with a pointer to a struct `:ctype:'v4l2_framebuffer'` structure. The driver fills all fields of the structure or returns an EINVAL error code when overlays are not supported.

**System Message: ERROR/3** (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\linux-master) (Documentation) (userspace-api) (media) (v4l) videioc-g-fbuf.rst, line 52); [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) videioc-g-fbuf.rst, line 52); [backlink](#)

Unknown interpreted text role "ctype".

To set the parameters for a *Video Output Overlay*, applications must initialize the `flags` field of a struct `:ctype:'v4l2_framebuffer'`. Since the framebuffer is implemented on the TV card all other parameters are determined by the driver. When an application calls `ref:VIDIOC_S_FBUF <VIDIOC_G_FBUF>` with a pointer to this structure, the driver prepares for the overlay and returns the framebuffer parameters as `ref:VIDIOC_G_FBUF <VIDIOC_G_FBUF>` does, or it returns an error code.

**System Message: ERROR/3** (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\linux-master) (Documentation) (userspace-api) (media) (v4l) videioc-g-fbuf.rst, line 57); [backlink](#)

Unknown interpreted text role "ctype".

**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-fbuf.rst, line 57); [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-fbuf.rst, line 57); [backlink](#)**

Unknown interpreted text role "ref".

To set the parameters for a *non-destructive Video Overlay*, applications must initialize the `flags` field, the `fmt` substructure, and call `ref:VIDIOC_S_FBUF <VIDIOC_G_FBUF>`. Again the driver prepares for the overlay and returns the framebuffer parameters as `ref:VIDIOC_G_FBUF <VIDIOC_G_FBUF>` does, or it returns an error code.

**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-fbuf.rst, line 66); [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-fbuf.rst, line 66); [backlink](#)**

Unknown interpreted text role "ref".

For a *destructive Video Overlay* applications must additionally provide a `base` address. Setting up a DMA to a random memory location can jeopardize the system security, its stability or even damage the hardware, therefore only the superuser can set the parameters for a destructive video overlay.

**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-fbuf.rst, line 78)**

Unknown directive type "tabularcolumns".

```
.. tabularcolumns:: |p{3.5cm}|p{3.5cm}|p{3.5cm}|p{6.6cm}|
```

**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-fbuf.rst, line 80)**

Unknown directive type "c.type".

```
.. c:type:: v4l2_framebuffer
```

**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-fbuf.rst, line 82)**

Unknown directive type "cssclass".

```
.. cssclass:: longtable
```

**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-fbuf.rst, line 84)**

Unknown directive type "flat-table".

```
.. flat-table:: struct v4l2_framebuffer
   :header-rows: 0
   :stub-columns: 0
   :widths:      1 1 1 2

   * - _u32
     - ``capability``
```

- Overlay capability flags set by the driver, see :ref:`framebuffer-cap`.
- \* - `__u32`
- `__flags``
- 
- Overlay control flags set by application and driver, see :ref:`framebuffer-flags`
- \* - void \*
- `__base``
- 
- Physical base address of the framebuffer, that is the address of the pixel in the top left corner of the framebuffer. [#fl]\_
- \* -
- 
- 
- This field is irrelevant to \*non-destructive Video Overlays\*. For \*destructive Video Overlays\* applications must provide a base address. The driver may accept only base addresses which are a multiple of two, four or eight bytes. For \*Video Output Overlays\* the driver must return a valid base address, so applications can find the corresponding Linux framebuffer device (see :ref:`osd`).
- \* - struct
- `__fmt``
- 
- Layout of the frame buffer.
- \* -
- `__u32`
- `__width``
- Width of the frame buffer in pixels.
- \* -
- `__u32`
- `__height``
- Height of the frame buffer in pixels.
- \* -
- `__u32`
- `__pixelformat``
- The pixel format of the framebuffer.
- \* -
- 
- 
- For \*non-destructive Video Overlays\* this field only defines a format for the struct :c:type:`v4l2\_window` `__chromakey`` field.
- \* -
- 
- 
- For \*destructive Video Overlays\* applications must initialize this field. For \*Video Output Overlays\* the driver must return a valid format.
- \* -
- 
- 
- Usually this is an RGB format (for example :ref:`V4L2\_PIX\_FMT\_RGB565` <V4L2-PIX-FMT-RGB565>) but YUV formats (only packed YUV formats when chroma keying is used, not including `__V4L2_PIX_FMT_YUYV`` and `__V4L2_PIX_FMT_UYVY``) and the `__V4L2_PIX_FMT_PAL8`` format are also permitted. The behavior of the driver when an application requests a compressed format is undefined. See :ref:`pixfmt` for information on pixel formats.
- \* -
- enum :c:type:`v4l2\_field`
- `__field``
- Drivers and applications shall ignore this field. If applicable, the field order is selected with the :ref:`VIDIOC\_S\_FMT` <VIDIOC\_G\_FMT> ioctl, using the `__field`` field of struct :c:type:`v4l2\_window`.
- \* -
- `__u32`
- `__bytesperline``
- Distance in bytes between the leftmost pixels in two adjacent lines.
- \* - :cspan:`3`

This field is irrelevant to \*non-destructive Video Overlays\*.

For \*destructive Video Overlays\* both applications and drivers can set this field to request padding bytes at the end of each line. Drivers however may ignore the requested value, returning `__width`` times bytes-per-pixel or a larger value required by the

hardware. That implies applications can just set this field to zero to get a reasonable default.

For *\*Video Output Overlays\** the driver must return a valid value.

Video hardware may access padding bytes, therefore they must reside in accessible memory. Consider for example the case where padding bytes after the last line of an image cross a system page boundary. Capture devices may write padding bytes, the value is undefined. Output devices ignore the contents of padding bytes.

When the image format is planar the `bytesperline` value applies to the first plane and is divided by the same factor as the `width` field for the other planes. For example the Cb and Cr planes of a YUV 4:2:0 image have half as many padding bytes following each line as the Y plane. To avoid ambiguities drivers must return a `bytesperline` value rounded up to a multiple of the scale factor.

- \* -
- `_u32`
- `sizeimage`
- This field is irrelevant to *\*non-destructive Video Overlays\**. For *\*destructive Video Overlays\** applications must initialize this field. For *\*Video Output Overlays\** the driver must return a valid format.

Together with `base` it defines the framebuffer memory accessible by the driver.

- \* -
- enum :c:type: `v4l2_colorspace`
- `colorspace`
- This information supplements the `pixelformat` and must be set by the driver, see :ref:`colorspaces`.
- \* -
- `_u32`
- `priv`
- Reserved. Drivers and applications must set this field to zero.

**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-fbuf.rst, line 210)**

Unknown directive type "tabularcolumns".

```
.. tabularcolumns:: |p{7.4cm}|p{1.6cm}|p{8.3cm}|
```

**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-fbuf.rst, line 214)**

Unknown directive type "flat-table".

```
.. flat-table:: Frame Buffer Capability Flags
:header-rows: 0
:stub-columns: 0
:widths:      3 1 4
```

- \* - `V4L2_FBUF_CAP_EXTERNOVERLAY`
- 0x0001
- The device is capable of non-destructive overlays. When the driver clears this flag, only destructive overlays are supported. There are no drivers yet which support both destructive and non-destructive overlays. Video Output Overlays are in practice always non-destructive.
- \* - `V4L2_FBUF_CAP_CHROMAKEY`
- 0x0002
- The device supports clipping by chroma-keying the images. That is, image pixels replace pixels in the VGA or video signal only where the latter assume a certain color. Chroma-keying makes no sense for destructive overlays.
- \* - `V4L2_FBUF_CAP_LIST_CLIPPING`
- 0x0004
- The device supports clipping using a list of clip rectangles.
- \* - `V4L2_FBUF_CAP_BITMAP_CLIPPING`
- 0x0008
- The device supports clipping using a bit mask.
- \* - `V4L2_FBUF_CAP_LOCAL_ALPHA`

- 0x0010
- The device supports clipping/blending using the alpha channel of the framebuffer or VGA signal. Alpha blending makes no sense for destructive overlays.
- \* - ``V4L2\_FBUF\_CAP\_GLOBAL\_ALPHA``
- 0x0020
- The device supports alpha blending using a global alpha value. Alpha blending makes no sense for destructive overlays.
- \* - ``V4L2\_FBUF\_CAP\_LOCAL\_INV\_ALPHA``
- 0x0040
- The device supports clipping/blending using the inverted alpha channel of the framebuffer or VGA signal. Alpha blending makes no sense for destructive overlays.
- \* - ``V4L2\_FBUF\_CAP\_SRC\_CHROMAKEY``
- 0x0080
- The device supports Source Chroma-keying. Video pixels with the chroma-key colors are replaced by framebuffer pixels, which is exactly opposite of ``V4L2\_FBUF\_CAP\_CHROMAKEY``

**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-fbuf.rst, line 258)**

Unknown directive type "tabularcolumns".

```
.. tabularcolumns:: |p{7.4cm}|p{1.6cm}|p{8.3cm}|
```

**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-fbuf.rst, line 262)**

Unknown directive type "cssclass".

```
.. cssclass:: longtable
```

**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-fbuf.rst, line 264)**

Unknown directive type "flat-table".

```
.. flat-table:: Frame Buffer Flags
:header-rows: 0
:stub-columns: 0
:widths:      3 1 4

* - ``V4L2_FBUF_FLAG_PRIMARY``
  - 0x0001
  - The framebuffer is the primary graphics surface. In other words, the overlay is destructive. This flag is typically set by any driver that doesn't have the ``V4L2_FBUF_CAP_EXTERNOVERLAY`` capability and it is cleared otherwise.
* - ``V4L2_FBUF_FLAG_OVERLAY``
  - 0x0002
  - If this flag is set for a video capture device, then the driver will set the initial overlay size to cover the full framebuffer size, otherwise the existing overlay size (as set by :ref:`VIDIOC_S_FMT` <VIDIOC_G_FMT>) will be used. Only one video capture driver (bttv) supports this flag. The use of this flag for capture devices is deprecated. There is no way to detect which drivers support this flag, so the only reliable method of setting the overlay size is through :ref:`VIDIOC_S_FMT` <VIDIOC_G_FMT>. If this flag is set for a video output device, then the video output overlay window is relative to the top-left corner of the framebuffer and restricted to the size of the framebuffer. If it is cleared, then the video output overlay window is relative to the video output display.
* - ``V4L2_FBUF_FLAG_CHROMAKEY``
  - 0x0004
  - Use chroma-keying. The chroma-key color is determined by the ``chromakey`` field of struct :c:type:`v4l2_window` and negotiated with the :ref:`VIDIOC_S_FMT` <VIDIOC_G_FMT> ioctl, see :ref:`overlay` and :ref:`osd`.
* - :cspan:`2` There are no flags to enable clipping using a list of clip rectangles or a bitmap. These methods are negotiated with the
```

```

:ref:`VIDIOC_S_FMT` <VIDIOC_G_FMT>` ioctl, see :ref:`overlay`
and :ref:`osd`.
* - ``V4L2_FBUF_FLAG_LOCAL_ALPHA``
  - 0x0008
  - Use the alpha channel of the framebuffer to clip or blend
    framebuffer pixels with video images. The blend function is:
    output = framebuffer pixel * alpha + video pixel * (1 - alpha).
    The actual alpha depth depends on the framebuffer pixel format.
* - ``V4L2_FBUF_FLAG_GLOBAL_ALPHA``
  - 0x0010
  - Use a global alpha value to blend the framebuffer with video
    images. The blend function is: output = (framebuffer pixel * alpha
    + video pixel * (255 - alpha)) / 255. The alpha value is
    determined by the ``global_alpha`` field of struct
    :c:type:`v4l2_window` and negotiated with the
    :ref:`VIDIOC_S_FMT` <VIDIOC_G_FMT>` ioctl, see :ref:`overlay`
    and :ref:`osd`.
* - ``V4L2_FBUF_FLAG_LOCAL_INV_ALPHA``
  - 0x0020
  - Like ``V4L2_FBUF_FLAG_LOCAL_ALPHA``, use the alpha channel of the
    framebuffer to clip or blend framebuffer pixels with video images,
    but with an inverted alpha value. The blend function is: output =
    framebuffer pixel * (1 - alpha) + video pixel * alpha. The actual
    alpha depth depends on the framebuffer pixel format.
* - ``V4L2_FBUF_FLAG_SRC_CHROMAKEY``
  - 0x0040
  - Use source chroma-keying. The source chroma-key color is
    determined by the ``chromakey`` field of struct
    :c:type:`v4l2_window` and negotiated with the
    :ref:`VIDIOC_S_FMT` <VIDIOC_G_FMT>` ioctl, see :ref:`overlay`
    and :ref:`osd`. Both chroma-keying are mutual exclusive to each
    other, so same ``chromakey`` field of struct
    :c:type:`v4l2_window` is being used.

```

## 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\v41\ (linux-master) (Documentation) (userspace-api) (media) (v41) vidioc-g-fbuf.rst, line 335); [backlink](#)

Unknown interpreted text role "ref".

### EPERM

[ref:`VIDIOC\\_S\\_FBUF` <VIDIOC\\_G\\_FBUF>](#) can only be called by a privileged user to negotiate the parameters for a destructive overlay.

**System Message: ERROR/3** (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v41\ (linux-master) (Documentation) (userspace-api) (media) (v41) vidioc-g-fbuf.rst, line 340); [backlink](#)

Unknown interpreted text role "ref".

### EINVAL

The [ref:`VIDIOC\\_S\\_FBUF` <VIDIOC\\_G\\_FBUF>](#) parameters are unsuitable.

**System Message: ERROR/3** (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v41\ (linux-master) (Documentation) (userspace-api) (media) (v41) vidioc-g-fbuf.rst, line 344); [backlink](#)

Unknown interpreted text role "ref".

[1] A physical base address may not suit all platforms. GK notes in theory we should pass something like PCI device + memory region + offset instead. If you encounter problems please discuss on the linux-media mailing list: <https://linuxtv.org/lists.php>.