ioctl VIDIOC_CREATE_BUFS

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-create-bufs.rst, line 2)

Unknown directive type "c:namespace".

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

Name

VIDIOC_CREATE_BUFS - Create buffers for Memory Mapped or User Pointer or DMA Buffer I/O

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-create-bufs.rst, line 18)

Unknown directive type "c:macro".

```
.. c:macro:: VIDIOC_CREATE_BUFS
```

```
int ioctl(int fd, VIDIOC_CREATE_BUFS, struct v4l2_create_buffers *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-create-bufs.rst, line 26); *backlink*

Unknown interpreted text role "c:func".

`argp`

Pointer to struct :c:type:`v4l2_create_buffers`.

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-create-bufs.rst, line 29); *backlink*

Unknown interpreted text role "c:type".

Description

This ioctl is used to create buffers for :ref:`memory mapped <mmap>` or :ref:`user pointer <userp>` or :ref:`DMA buffer <dmabuf>` I/O. It can be used as an alternative or in addition to the :ref:`VIDIOC_REQBUFS` ioctl, when a tighter control over buffers is required. This ioctl can be called multiple times to create buffers of different sizes.

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-create-bufs.rst, line 34); *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-create-bufs.rst, line 34); *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-create-bufs.rst, line 34); *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-create-bufs.rst, line 34); *backlink*

Unknown interpreted text role "ref".

To allocate the device buffers applications must initialize the relevant fields of the struct :c:type:`v4l2_create_buffers` structure. The `count` field must be set to the number of requested buffers, the `memory` field specifies the requested I/O method and the `reserved` array must be zeroed.

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-create-bufs.rst, line 41); *backlink*

Unknown interpreted text role "c:type".

The `format` field specifies the image format that the buffers must be able to handle. The application has to fill in this struct :c:type:`v4l2_format`. Usually this will be done using the :ref:`VIDIOC_TRY_FMT <VIDIOC_G_FMT>` or :ref:`VIDIOC_G_FMT <VIDIOC_G_FMT>` ioctls to ensure that the requested format is supported by the driver. Based on the format's `type` field the requested buffer size (for single-planar) or plane sizes (for multi-planar formats) will be used for the allocated buffers. The driver may return an error if the size(s) are not supported by the hardware (usually because they are too small).

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-create-bufs.rst, line 47); *backlink*

Unknown interpreted text role "c:type".

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-create-bufs.rst, line 47); *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-create-bufs.rst, line 47); *backlink*

Unknown interpreted text role "ref".

The buffers created by this ioctl will have as minimum size the size defined by the `format.pix.sizeimage` field (or the corresponding fields for other format types). Usually if the `format.pix.sizeimage` field is less than the minimum required for the given format, then an error will be returned since drivers will typically not allow this. If it is larger, then the value will be used as-is. In other words, the driver may reject the requested size, but if it is accepted the driver will use it unchanged.

When the ioctl is called with a pointer to this structure the driver will attempt to allocate up to the requested number of buffers and store the actual number allocated and the starting index in the `count` and the `index` fields respectively. On return `count` can be smaller than the number requested.

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-create-bufs.rst, line 73)

Unknown directive type "c:type".

```
.. c:type:: v4l2_create_buffers
```

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-create-bufs.rst, line 75)

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-create-bufs.rst, line 77)

Unknown directive type "flat-table".

```
.. flat-table:: struct v4l2_create_buffers
    :header-rows:  0
    :stub-columns: 0
    :widths:       1 1 2

    * - __u32
      - ``index``
      - The starting buffer index, returned by the driver.
    * - __u32
      - ``count``
      - The number of buffers requested or granted. If count == 0, then
        :ref:`VIDIOC_CREATE_BUFS` will set ``index`` to the current number of
        created buffers, and it will check the validity of ``memory`` and
        ``format.type``. If those are invalid -1 is returned and errno is
        set to ``EINVAL`` error code, otherwise :ref:`VIDIOC_CREATE_BUFS` returns
        0. It will never set errno to ``EBUSY`` error code in this particular
        case.
    * - __u32
      - ``memory``
      - Applications set this field to ``V4L2_MEMORY_MMAP``,
        ``V4L2_MEMORY_DMABUF`` or ``V4L2_MEMORY_USERPTR``. See
        :c:type:`v4l2_memory`
    * - struct :c:type:`v4l2_format`
      - ``format``
      - Filled in by the application, preserved by the driver.
    * - __u32
      - ``capabilities``
      - Set by the driver. If 0, then the driver doesn't support
        capabilities. In that case all you know is that the driver is
        guaranteed to support ``V4L2_MEMORY_MMAP`` and *might* support
        other :c:type:`v4l2_memory` types. It will not support any other
        capabilities. See :ref:`here <v4l2-buf-capabilities>` for a list of the
        capabilities.

        If you want to just query the capabilities without making any
        other changes, then set ``count`` to 0, ``memory`` to
        ``V4L2_MEMORY_MMAP`` and ``format.type`` to the buffer type.

    * - __u32
      - ``flags``
      - Specifies additional buffer management attributes.
        See :ref:`memory-flags`.

    * - __u32
      - ``reserved``\ [6]
      - A place holder for future extensions. Drivers and applications
        must set the array to zero.
```

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-create-bufs.rst, line 128); *backlink*

Unknown interpreted text role "ref".

ENOMEM

No memory to allocate buffers for :ref:`memory mapped <mmap>` I/O.

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-create-bufs.rst, line 133); *backlink*

Unknown interpreted text role "ref".

EINVAL

The buffer type (`format.type` field), requested I/O method (`memory`) or format (`format` field) is not valid.