V4L2 mmap()

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 2)

Unknown directive type "c:namespace".

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

Name

v4l2-mmap - Map device memory into application address space

Synopsis

```
#include <unistd.h>
#include <sys/mman.h>
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 23)

Unknown directive type "c:function".

```
.. c:function:: void *mmap( void *start, size_t length, int prot, int flags, int fd, off_t offset
```

Arguments

`start`

Map the buffer to this address in the application's address space. When the `MAP_FIXED` flag is specified, `start` must be a multiple of the pagesize and mmap will fail when the specified address cannot be used. Use of this option is discouraged; applications should just specify a `NULL` pointer here.

`length`

Length of the memory area to map. This must be the same value as returned by the driver in the struct :c:type:`v4l2_buffer` `length` field for the single-planar API, and the same value as returned by the driver in the struct :c:type:`v4l2_plane` `length` field for the multi-planar API.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 36); *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]func-mmap.rst, line 36); *backlink*

Unknown interpreted text role "c:type".

`prot`

The `prot` argument describes the desired memory protection. Regardless of the device type and the direction of data exchange it should be set to `PROT_READ` | `PROT_WRITE`, permitting read and write access to image buffers. Drivers should support at least this combination of flags.

Note

1. The Linux `videobuf` kernel module, which is used by some drivers supports only `PROT_READ` | `PROT_WRITE`. When the driver does not support the desired protection, the :c:func:`mmap()` function fails.

System Message: ERROR/3 (D:\onboarding-resources\sample-

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

Unknown interpreted text role "c:func".

2. Device memory accesses (e. g. the memory on a graphics card with video capturing hardware) may incur a performance penalty compared to main memory accesses, or reads may be significantly slower than writes or vice versa. Other I/O methods may be more efficient in such case.

`flags`

The `flags` parameter specifies the type of the mapped object, mapping options and whether modifications made to the mapped copy of the page are private to the process or are to be shared with other references.

`MAP_FIXED` requests that the driver selects no other address than the one specified. If the specified address cannot be used, :c:func:`mmap()` will fail. If `MAP_FIXED` is specified, `start` must be a multiple of the pagesize. Use of this option is discouraged.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 69); *backlink*

Unknown interpreted text role "c:func".

One of the `MAP_SHARED` or `MAP_PRIVATE` flags must be set. `MAP_SHARED` allows applications to share the mapped memory with other (e. g. child-) processes.

Note

The Linux `videobuf` module which is used by some drivers supports only `MAP_SHARED`. `MAP_PRIVATE` requests copy-on-write semantics. V4L2 applications should not set the `MAP_PRIVATE`, `MAP_DENYWRITE`, `MAP_EXECUTABLE` or `MAP_ANON` flags.

`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]func-mmap.rst, line 88); *backlink*

Unknown interpreted text role "c:func".

`offset`

Offset of the buffer in device memory. This must be the same value as returned by the driver in the struct :c:type:`v4l2_buffer` `m` union `offset` field for the single-planar API, and the same value as returned by the driver in the struct :c:type:`v4l2_plane` `m` union `mem_offset` field for the multi-planar API.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 91); *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]func-mmap.rst, line 91); *backlink*

Unknown interpreted text role "c:type".

Description

The :c:func:`mmap()` function asks to map `length` bytes starting at `offset` in the memory of the device specified by `fd` into the application address space, preferably at address `start`. This latter address is a hint only, and is usually specified as 0.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 101); *backlink*

Unknown interpreted text role "c:func".

Suitable length and offset parameters are queried with the :ref:`VIDIOC_QUERYBUF` ioctl. Buffers must be allocated with the :ref:`VIDIOC_REQBUFS` ioctl before they can be queried.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 106); *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]func-mmap.rst, line 106); *backlink*

Unknown interpreted text role "ref".

To unmap buffers the :c:func:`munmap()` function is used.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 111); *backlink*

Unknown interpreted text role "c:func".

Return Value

On success :c:func:`mmap()` returns a pointer to the mapped buffer. On error `MAP_FAILED` (-1) is returned, and the `errno` variable is set appropriately. Possible error codes 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]func-mmap.rst, line 116); *backlink*

Unknown interpreted text role "c:func".

EBADF

`fd` is not a valid file descriptor.

EACCES

`fd` is not open for reading and writing.

EINVAL

The `start` or `length` or `offset` are not suitable. (E. g. they are too large, or not aligned on a `PAGESIZE` boundary.)

The `flags` or `prot` value is not supported.

No buffers have been allocated with the :ref:`VIDIOC_REQBUFS` ioctl.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]func-mmap.rst, line 132); *backlink*

Unknown interpreted text role "ref".

ENOMEM

Not enough physical or virtual memory was available to complete the request.