

VIDEO_GET_EVENT

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\drivers\staging\media\av7110\linux-master) (drivers) (staging) (media) (av7110) video-get-event.rst, line 2)

Unknown directive type "c:namespace".

```
.. c:namespace:: DTV.video
```

Name

VIDEO_GET_EVENT

Attention!

This ioctl is deprecated.

Synopsis

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\drivers\staging\media\av7110\linux-master) (drivers) (staging) (media) (av7110) video-get-event.rst, line 20)

Unknown directive type "c:macro".

```
.. c:macro:: VIDEO_GET_EVENT
```

```
int ioctl1(fd, VIDEO_GET_EVENT, struct video_event *ev)
```

Arguments

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\drivers\staging\media\av7110\linux-master) (drivers) (staging) (media) (av7110) video-get-event.rst, line 27)

Unknown directive type "flat-table".

```
.. flat-table::
   :header-rows: 0
   :stub-columns: 0

   - .. row 1
     - int fd
     - File descriptor returned by a previous call to open().

   - .. row 2
     - int request
     - Equals VIDEO_GET_EVENT for this command.

   - .. row 3
     - struct video_event \*ev
     - Points to the location where the event, if any, is to be stored.
```

Description

This ioctl is for Digital TV devices only. To get events from a V4L2 decoder use the V4L2 [ref`VIDIOC_DQEVENT`](#) ioctl instead.

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

```
master\drivers\staging\media\av7110\ (linux-master) (drivers) (staging) (media)
(av7110) video-get-event.rst, line 52); backlink
```

Unknown interpreted text role "ref".

This ioctl call returns an event of type `video_event` if available. If an event is not available, the behavior depends on whether the device is in blocking or non-blocking mode. In the latter case, the call fails immediately with `errno` set to `EWOULDBLOCK`. In the former case, the call blocks until an event becomes available. The standard Linux `poll()` and/or `select()` system calls can be used with the device file descriptor to watch for new events. For `select()`, the file descriptor should be included in the `exceptfds` argument, and for `poll()`, `POLLPRI` should be specified as the wake-up condition. Read-only permissions are sufficient for this ioctl call.

```
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-
master\drivers\staging\media\av7110\ (linux-master) (drivers) (staging) (media)
(av7110) video-get-event.rst, line 66)
```

Unknown directive type "c:type".

```
.. c:type:: video_event
```

```
struct video_event {
    __s32 type;
#define VIDEO_EVENT_SIZE_CHANGED      1
#define VIDEO_EVENT_FRAME_RATE_CHANGED 2
#define VIDEO_EVENT_DECODER_STOPPED  3
#define VIDEO_EVENT_VSYNC             4
    long timestamp;
    union {
        video_size_t size;
        unsigned int frame_rate; /* in frames per 1000sec */
        unsigned char vsync_field; /* unknown/odd/even/progressive */
    } u;
};
```

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\drivers\staging\media\av7110\ (linux-master) (drivers) (staging) (media)
(av7110) video-get-event.rst, line 87); backlink
```

Unknown interpreted text role "ref".

```
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-
master\drivers\staging\media\av7110\ (linux-master) (drivers) (staging) (media)
(av7110) video-get-event.rst, line 91)
```

Unknown directive type "flat-table".

```
.. flat-table::
    :header-rows: 0
    :stub-columns: 0

    - .. row 1
      - ``EWOULDBLOCK``
      - There is no event pending, and the device is in non-blocking mode.

    - .. row 2
      - ``EOVERFLOW``
      - Overflow in event queue - one or more events were lost.
```