

**NAME**

xcb\_randr\_get\_crtc\_info -

**SYNOPSIS**

```
#include <xcb/randr.h>
```

**Request function**

```
xcb_randr_get_crtc_info_cookie_t xcb_randr_get_crtc_info(xcb_connection_t *conn,
    xcb_randr_crtc_t crtc, xcb_timestamp_t config_timestamp);
```

**Reply datastructure**

```
typedef struct xcb_randr_get_crtc_info_reply_t {
    uint8_t    response_type;
    uint8_t    status;
    uint16_t   sequence;
    uint32_t   length;
    xcb_timestamp_t timestamp;
    int16_t    x;
    int16_t    y;
    uint16_t   width;
    uint16_t   height;
    xcb_randr_mode_t mode;
    uint16_t   rotation;
    uint16_t   rotations;
    uint16_t   num_outputs;
    uint16_t   num_possible_outputs;
} xcb_randr_get_crtc_info_reply_t;
```

**Reply function**

```
xcb_randr_get_crtc_info_reply_t *xcb_randr_get_crtc_info_reply(xcb_connection_t *conn,
    xcb_randr_get_crtc_info_cookie_t cookie, xcb_generic_error_t **e);
```

**Reply accessors**

```
xcb_randr_output_t *xcb_randr_get_crtc_info_outputs(const xcb_randr_get_crtc_info_request_t
    *reply);
```

```
int xcb_randr_get_crtc_info_outputs_length(const xcb_randr_get_crtc_info_reply_t *reply);
```

```
xcb_generic_iterator_t xcb_randr_get_crtc_info_outputs_end(const xcb_randr_get_crtc_info_reply_t
```

```

    *reply);

xcb_randr_output_t *xcb_randr_get_crtc_info_possible(const xcb_randr_get_crtc_info_request_t
    *reply);

int xcb_randr_get_crtc_info_possible_length(const xcb_randr_get_crtc_info_reply_t *reply);

xcb_generic_iterator_t xcb_randr_get_crtc_info_possible_end(const xcb_randr_get_crtc_info_reply_t
    *reply);

```

## REQUEST ARGUMENTS

*conn*            The XCB connection to X11.

*crtc*            TODO: NOT YET DOCUMENTED.

*config\_timestamp*  
                  TODO: NOT YET DOCUMENTED.

## REPLY FIELDS

*response\_type* The type of this reply, in this case *XCB\_RANDR\_GET\_CRTC\_INFO*. This field is also present in the *xcb\_generic\_reply\_t* and can be used to tell replies apart from each other.

*sequence*        The sequence number of the last request processed by the X11 server.

*length*          The length of the reply, in words (a word is 4 bytes).

*status*          TODO: NOT YET DOCUMENTED.

*timestamp*        TODO: NOT YET DOCUMENTED.

*x*                TODO: NOT YET DOCUMENTED.

*y*                TODO: NOT YET DOCUMENTED.

*width*            TODO: NOT YET DOCUMENTED.

*height*          TODO: NOT YET DOCUMENTED.

*mode*            TODO: NOT YET DOCUMENTED.

*rotation*            TODO: NOT YET DOCUMENTED.

*rotations*            TODO: NOT YET DOCUMENTED.

*num\_outputs*        TODO: NOT YET DOCUMENTED.

*num\_possible\_outputs*  
                      TODO: NOT YET DOCUMENTED.

## DESCRIPTION

### RETURN VALUE

Returns an *xcb\_randr\_get\_crtc\_info\_cookie\_t*. Errors have to be handled when calling the reply function *xcb\_randr\_get\_crtc\_info\_reply*.

If you want to handle errors in the event loop instead, use *xcb\_randr\_get\_crtc\_info\_unchecked*. See **xcb-requests(3)** for details.

### ERRORS

This request does never generate any errors.

### SEE ALSO

### AUTHOR

Generated from randr.xml. Contact [xcb@lists.freedesktop.org](mailto:xcb@lists.freedesktop.org) for corrections and improvements.