

NAME

xcb_randr_get_crtc_gamma -

SYNOPSIS

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

Request function

```
xcb_randr_get_crtc_gamma_cookie_t xcb_randr_get_crtc_gamma(xcb_connection_t *conn,
    xcb_randr_crtc_t crtc);
```

Reply datastructure

```
typedef struct xcb_randr_get_crtc_gamma_reply_t {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t size;
    uint8_t pad1[22];
} xcb_randr_get_crtc_gamma_reply_t;
```

Reply function

```
xcb_randr_get_crtc_gamma_reply_t *xcb_randr_get_crtc_gamma_reply(xcb_connection_t *conn,
    xcb_randr_get_crtc_gamma_cookie_t cookie, xcb_generic_error_t **e);
```

Reply accessors

```
uint16_t *xcb_randr_get_crtc_gamma_red(const xcb_randr_get_crtc_gamma_request_t *reply);
```

```
int xcb_randr_get_crtc_gamma_red_length(const xcb_randr_get_crtc_gamma_reply_t *reply);
```

```
xcb_generic_iterator_t xcb_randr_get_crtc_gamma_red_end(const xcb_randr_get_crtc_gamma_reply_t
    *reply);
```

```
uint16_t *xcb_randr_get_crtc_gamma_green(const xcb_randr_get_crtc_gamma_request_t *reply);
```

```
int xcb_randr_get_crtc_gamma_green_length(const xcb_randr_get_crtc_gamma_reply_t *reply);
```

```
xcb_generic_iterator_t xcb_randr_get_crtc_gamma_green_end(const
    xcb_randr_get_crtc_gamma_reply_t *reply);
```

```
uint16_t *xcb_randr_get_crtc_gamma_blue(const xcb_randr_get_crtc_gamma_request_t *reply);
```

```
int xcb_randr_get_crtc_gamma_blue_length(const xcb_randr_get_crtc_gamma_reply_t *reply);
```

```
xcb_generic_iterator_t xcb_randr_get_crtc_gamma_blue_end(const  
    xcb_randr_get_crtc_gamma_reply_t *reply);
```

REQUEST ARGUMENTS

conn The XCB connection to X11.

crtc TODO: NOT YET DOCUMENTED.

REPLY FIELDS

response_type The type of this reply, in this case *XCB_RANDR_GET_CRTC_GAMMA*. 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).

size TODO: NOT YET DOCUMENTED.

DESCRIPTION

RETURN VALUE

Returns an *xcb_randr_get_crtc_gamma_cookie_t*. Errors have to be handled when calling the reply function *xcb_randr_get_crtc_gamma_reply*.

If you want to handle errors in the event loop instead, use *xcb_randr_get_crtc_gamma_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 for corrections and improvements.