

**NAME**

`ibv_reg_mr`, `ibv_dereg_mr` – register or deregister a memory region (MR)

**SYNOPSIS**

```
#include <infiniband/verbs.h>
```

```
struct ibv_mr *ibv_reg_mr(struct ibv_pd *pd, void *addr,
                          size_t length, int access);
```

```
int ibv_dereg_mr(struct ibv_mr *mr);
```

**DESCRIPTION**

**ibv\_reg\_mr()** registers a memory region (MR) associated with the protection domain *pd*. The MR's starting address is *addr* and its size is *length*. The argument *access* describes the desired memory protection attributes; it is either 0 or the bitwise OR of one or more of the following flags:

**IBV\_ACCESS\_LOCAL\_WRITE** Enable Local Write Access

**IBV\_ACCESS\_REMOTE\_WRITE** Enable Remote Write Access

**IBV\_ACCESS\_REMOTE\_READ** Enable Remote Read Access

**IBV\_ACCESS\_REMOTE\_ATOMIC** Enable Remote Atomic Operation Access (if supported)

**IBV\_ACCESS\_MW\_BIND** Enable Memory Window Binding

**IBV\_ACCESS\_ZERO\_BASED** Use byte offset from beginning of MR to access this MR, instead of a pointer address

**IBV\_ACCESS\_ON\_DEMAND** Create an on-demand paging MR

If **IBV\_ACCESS\_REMOTE\_WRITE** or **IBV\_ACCESS\_REMOTE\_ATOMIC** is set, then **IBV\_ACCESS\_LOCAL\_WRITE** must be set too.

Local read access is always enabled for the MR.

To create an implicit ODP MR, **IBV\_ACCESS\_ON\_DEMAND** should be set, *addr* should be 0 and *length* should be **SIZE\_MAX**.

**ibv\_dereg\_mr()** deregisters the MR *mr*.

**RETURN VALUE**

**ibv\_reg\_mr()** returns a pointer to the registered MR, or NULL if the request fails. The local key (**L\_Key**) field **lkey** is used as the **lkey** field of struct `ibv_sge` when posting buffers with `ibv_post_*` verbs, and the remote key (**R\_Key**) field **rkey** is used by remote processes to perform Atomic and RDMA operations. The remote process places this **rkey** as the **rkey** field of struct `ibv_send_wr` passed to the `ibv_post_send` function.

**ibv\_dereg\_mr()** returns 0 on success, or the value of `errno` on failure (which indicates the failure reason).

**NOTES**

**ibv\_dereg\_mr()** fails if any memory window is still bound to this MR.

**SEE ALSO**

`ibv_alloc_pd(3)`, `ibv_post_send(3)`, `ibv_post_recv(3)`, `ibv_post_srq_recv(3)`

**AUTHORS**

Dotan Barak <dotanba@gmail.com>