NAME

ibv_post_srq_recv − post a list of work requests (WRs) to a shared receive queue (SRQ)

SYNOPSIS

#include <infiniband/verbs.h>

int ibv_post_srq_recv(struct ibv_srq **srq*, **struct ibv_recv_wr** **wr*,
struct ibv_recv_wr ***bad_wr*);

DESCRIPTION

ibv_post_srq_recv() posts the linked list of work requests (WRs) starting with *wr* to the shared receive queue (SRQ) *srq*. It stops processing WRs from this list at the first failure (that can be detected immediately while requests are being posted), and returns this failing WR through *bad_wr*.

The argument *wr* is an ibv_recv_wr struct, as defined in <infiniband/verbs.h>.

```
struct ibv_recv_wr {
        uint64_t            wr_id;    /* User defined WR ID */
        struct ibv_recv_wr  *next;    /* Pointer to next WR in list, NULL if last WR */
        struct ibv_sge      *sg_list; /* Pointer to the s/g array */
        int                 num_sge;  /* Size of the s/g array */
};

struct ibv_sge {
        uint64_t            addr;     /* Start address of the local memory buffer */
        uint32_t            length;   /* Length of the buffer */
        uint32_t            lkey;     /* Key of the local Memory Region */
};
```

RETURN VALUE

ibv_post_srq_recv() returns 0 on success, or the value of errno on failure (which indicates the failure reason).

NOTES

The buffers used by a WR can only be safely reused after WR the request is fully executed and a work completion has been retrieved from the corresponding completion queue (CQ).

If a WR is being posted to a UD QP, the Global Routing Header (GRH) of the incoming message will be placed in the first 40 bytes of the buffer(s) in the scatter list. If no GRH is present in the incoming message, then the first bytes will be undefined. This means that in all cases, the actual data of the incoming message will start at an offset of 40 bytes into the buffer(s) in the scatter list.

SEE ALSO

ibv_create_qp(3), **ibv_post_send**(3), **ibv_post_recv**(3), **ibv_poll_cq**(3)

AUTHORS

Dotan Barak <dotanba@gmail.com>