NAME

rmvb – remove a message block from a message

SYNOPSIS

#include <sys/stream.h>

mblk_t *rmvb(mblk_t ***mp*****, mblk_t** ***bp*****);**

INTERFACE LEVEL

Architecture independent level 1 (DDI/DKI).

ARGUMENTS

mp Message from which a block is to be removed. **mblk_t** is an instance of the **msgb**(9S) structure.

bp Message block to be removed.

DESCRIPTION

rmvb( ) removes a message block (*bp*) from a message (*mp*), and returns a pointer to the altered message. The message block is not freed, merely removed from the message. It is the module or driver's responsibility to free the message block.

RETURN VALUES

If successful, a pointer to the message (minus the removed block) is returned. The pointer is **NULL** if *bp* was the only block of the message before **rmvb( )** was called. If the designated message block (*bp*) does not exist, **-1** is returned.

CONTEXT

rmvb( ) can be called from user or interrupt context.

EXAMPLE

This routine removes all zero-length **M_DATA** message blocks from the given message. For each message block in the message, save the next message block (line 10). If the current message block is of type **M_DATA** and has no data in its buffer (line 11), then remove it from the message (line 12) and free it (line 13). In either case, continue with the next message block in the message (line 16).

```
 1 void
 2 xxclean(mp)
 3    mblk_t *mp;
 4 {
 5        mblk_t *tmp;
 6        mblk_t *nmp;
 7
 8        tmp = mp;
 9        while (tmp) {
10                nmp = tmp->b_cont;
11                if ((tmp->b_datap->db_type == M_DATA) &&
                     (tmp->b_rptr == tmp->b_wptr)) {
12                        (void) rmvb(mp, tmp);
13                        freeb(tmp);
14                }
15                tmp = nmp;
16        }
17 }
```

SEE ALSO

freeb(9F), **msgb**(9S)

Writing Device Drivers
STREAMS Programming Guide