NAME

rmvq – remove a message from a queue

SYNOPSIS

```
#include <sys/stream.h>

void rmvq(queue_t *q, mblk_t *mp);
```

INTERFACE LEVEL

Architecture independent level 1 (DDI/DKI).

ARGUMENTS

q	Queue containing the message to be removed.
mp	Message to remove.

DESCRIPTION

rmvq( ) removes a message from a queue. A message can be removed from anywhere on a queue. To prevent modules and drivers from having to deal with the internals of message linkage on a queue, either **rmvq( )** or **getq**(9F) should be used to remove a message from a queue.

CONTEXT

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

EXAMPLE

This code fragment illustrates how one may flush one type of message from a queue. In this case, only **M_PROTO T_DATA_IND** messages are flushed. For each message on the queue, if it is an **M_PROTO** message (line 8) of type **T_DATA_IND** (line 10), save a pointer to the next message (line 11), remove the **T_DATA_IND** message (line 12) and free it (line 13). Continue with the next message in the list (line 19).

```
1  mblk_t *mp, *nmp;
2  queue_t *q;
3  union T_primitives *tp;
4
5       freezestr(q);
6       mp = q->q_first;
7       while (mp) {
8               if (mp->b_datap->db_type == M_PROTO) {
9                       tp = (union T_primitives *)mp->b_rptr;
10                      if (tp->type == T_DATA_IND) {
11                              nmp = mp->b_next;
12                              rmvq(q, mp);
13                              freemsg(mp);
14                              mp = nmp;
15                      } else {
16                              mp = mp->b_next;
17                      }
18              } else {
19                      mp = mp->b_next;
20              }
21      }
22      unfreezestr(q);
```

SEE ALSO

freemsg(9F), **freezestr**(9F), **getq**(9F), **insq**(9F), **unfreezestr**(9F)

Writing Device Drivers
STREAMS Programming Guide

WARNINGS

Make sure that the message *mp* is linked onto *q* to avoid a possible system panic.

NOTES

The stream must be frozen using **freezestr**(9F) before calling **rmvq( ).**