

I/O MESSAGES(c)

I/O MESSAGES(c)

INTRODUCTION TO I/O PROCESS MESSAGES

Most block and record device I/O driver processes are programmed to accept messages of the following types:

IOREAD	1
IOWRITE	2
IOOPEN	3
IOCLOSE	4

However some I/O device drivers may not be programmed to accept IOOPEN and IOCLOSE messages, as they may be unnecessary for these devices. When the I/O device driver process is built using *ldp(e)* one of the specifications is whether the process will accept open and close messages. The format of a message to the device drivers is specified by the following structure:

```
struct io_msgd {
    struct    msghdr io_dhr; /* message 6-word header */
    int      msiosid;      /* ID of segment into which or from which I/O is to take place */
    int      msioba;      /* word offset into segment */
    int      msiobc;      /* number of bytes to be read */
    char     msiodev;     /* logical device number */
    char     msiob0;      /* high order byte of block number */
    char     *msiob1;     /* low order word of block number */
    char     msiopri;     /* priority of I/O */
    char     msiotry;     /* number of retries on error */
    int      msiocnt;     /* number of bytes transferred */
    int      msfill1;     /* scratch word, may be used by driver */
    int      msfill2;     /* scratch word, may be used by driver */
    int      msfill3;     /* scratch word, may be used by driver */
}
```

The IOOPEN and IOCLOSE messages need only specify the logical device number of the particular device driver and the number of times that the device driver file is open. This information is normally only available to the file manager process; the file manager thus will automatically send the open and close messages to the device drivers when the special files are opened and closed.

The IOREAD and IOWRITE messages must specify a total of 5 arguments besides the normal parameters in the message header as discussed in the introduction. These are the ID of the segment into or from which the I/O transfer is to take place, the word offset into this segment, the number of bytes to be transferred, the logical device number of the driver and the block number at which the transfer is to start. The segment may be either in supervisor or user address space, but it must be locked for I/O by the sender of the message before the message is sent. The word offset into the segment is from the beginning of the segment if it is positive, otherwise it is from the end of the segment (e.g. for stack segment). Normally the file manager will make a call to the *ioqueuem* routine to determine if the transfer is within the bounds of the segment and if the segment is locked in memory. *Ioqueuem* then sends message to the appropriate driver. This routine will convert the virtual address given by *msiosid* and *msioba* into a 22-bit physical address. The lower 16 bits are returned in *msioba* and the upper 2 bits are set in the *msustat* byte of the message status byte. The total number of bytes read or written are returned in *msiocnt*. By invoking the *messink* routine in the device driver when the I/O transfer is complete, the segment is unlocked.

I/O MESSAGES(c)

SEE ALSO

ioqueue(m)(b), open(c), close(c).

DIAGNOSTICS

An error status byte is returned if the segment is not locked for I/O, the segment does not exist, the I/O transfer would extend beyond the end of the segment or if a physical I/O error occurs in the transfer.

I/O MESSAGES(c)

However some I/O device drivers may not be programmed to accept HOBFF and LOCK messages. It is the user's responsibility to determine if the device driver supports these messages. When the I/O device driver message is built, the user must specify whether the program will accept open and close messages. The format of a message to the device driver is specified by the following structure:

Field Name	Field Description
msg	Message word, may be used by driver
msg2	Message word, may be used by driver
msg3	Message word, may be used by driver
msg4	Message word, may be used by driver
msg5	Message word, may be used by driver
msg6	Message word, may be used by driver
msg7	Message word, may be used by driver
msg8	Message word, may be used by driver
msg9	Message word, may be used by driver
msg10	Message word, may be used by driver
msg11	Message word, may be used by driver
msg12	Message word, may be used by driver
msg13	Message word, may be used by driver
msg14	Message word, may be used by driver
msg15	Message word, may be used by driver
msg16	Message word, may be used by driver
msg17	Message word, may be used by driver
msg18	Message word, may be used by driver
msg19	Message word, may be used by driver
msg20	Message word, may be used by driver
msg21	Message word, may be used by driver
msg22	Message word, may be used by driver
msg23	Message word, may be used by driver
msg24	Message word, may be used by driver
msg25	Message word, may be used by driver
msg26	Message word, may be used by driver
msg27	Message word, may be used by driver
msg28	Message word, may be used by driver
msg29	Message word, may be used by driver
msg30	Message word, may be used by driver
msg31	Message word, may be used by driver
msg32	Message word, may be used by driver
msg33	Message word, may be used by driver
msg34	Message word, may be used by driver
msg35	Message word, may be used by driver
msg36	Message word, may be used by driver
msg37	Message word, may be used by driver
msg38	Message word, may be used by driver
msg39	Message word, may be used by driver
msg40	Message word, may be used by driver
msg41	Message word, may be used by driver
msg42	Message word, may be used by driver
msg43	Message word, may be used by driver
msg44	Message word, may be used by driver
msg45	Message word, may be used by driver
msg46	Message word, may be used by driver
msg47	Message word, may be used by driver
msg48	Message word, may be used by driver
msg49	Message word, may be used by driver
msg50	Message word, may be used by driver
msg51	Message word, may be used by driver
msg52	Message word, may be used by driver
msg53	Message word, may be used by driver
msg54	Message word, may be used by driver
msg55	Message word, may be used by driver
msg56	Message word, may be used by driver
msg57	Message word, may be used by driver
msg58	Message word, may be used by driver
msg59	Message word, may be used by driver
msg60	Message word, may be used by driver
msg61	Message word, may be used by driver
msg62	Message word, may be used by driver
msg63	Message word, may be used by driver
msg64	Message word, may be used by driver
msg65	Message word, may be used by driver
msg66	Message word, may be used by driver
msg67	Message word, may be used by driver
msg68	Message word, may be used by driver
msg69	Message word, may be used by driver
msg70	Message word, may be used by driver
msg71	Message word, may be used by driver
msg72	Message word, may be used by driver
msg73	Message word, may be used by driver
msg74	Message word, may be used by driver
msg75	Message word, may be used by driver
msg76	Message word, may be used by driver
msg77	Message word, may be used by driver
msg78	Message word, may be used by driver
msg79	Message word, may be used by driver
msg80	Message word, may be used by driver
msg81	Message word, may be used by driver
msg82	Message word, may be used by driver
msg83	Message word, may be used by driver
msg84	Message word, may be used by driver
msg85	Message word, may be used by driver
msg86	Message word, may be used by driver
msg87	Message word, may be used by driver
msg88	Message word, may be used by driver
msg89	Message word, may be used by driver
msg90	Message word, may be used by driver
msg91	Message word, may be used by driver
msg92	Message word, may be used by driver
msg93	Message word, may be used by driver
msg94	Message word, may be used by driver
msg95	Message word, may be used by driver
msg96	Message word, may be used by driver
msg97	Message word, may be used by driver
msg98	Message word, may be used by driver
msg99	Message word, may be used by driver
msg100	Message word, may be used by driver

The HOBFF and LOCK messages only specify the logical device number of the particular device driver and the number of times that the device driver is open. The information is only available in the message processor. The manager thus will automatically send the open and close messages to the device files when the special files are opened and closed.

The HOBFF and LOCK messages must specify a total of 5 arguments besides the device number. The arguments are discussed in the introduction. These are the (1) the segment name of the message, (2) the word offset into the segment, the number of bytes to be transferred, the block number of the device and the block number at which the transfer is to start. The segment may be either in segment or user address space, but it must be locked (L) at the position of the message before the message is sent. The word offset into the segment is 0 on the beginning of the segment. If it is positive, otherwise it is from the end of the segment. For example, if the message will make a call to the program routine to determine the transfer is within the bounds of the segment and if the segment is locked in transfer, the routine will convert the virtual address given by the message to the appropriate device. The routine will convert the virtual address given by the message into a 32-bit physical address. The routine will also convert the physical address into a 32-bit virtual address. The total number of bytes read or written will be set in the seventh byte of the message status byte. The total number of bytes read or written will be turned in message. If leaving the system routine in the driver routine when the I/O routine is done, the segment is unlocked.