Manual Pages

F

This appendix lists the STREAMS related manual pages. The manual pages are supplied in the SunOS 5.x Reference Manuals and are listed here for reference.

- `strchg`(1) - change or query Stream configuration
- `autopush`(1M) - configure automatically pushed STREAMS modules
- `fdetach`(1M) - detach a name from a STREAMS-based file descriptor
- `strace`(1M) - print STREAMS trace messages
- `strclean`(1M) - STREAMS error logger cleanup program
- `strerr`(1M) - STREAMS error logger daemon
- `close`(2) - close a file descriptor
- `getmsg`(2) - get next message off a Stream
- `ioctl`(2) - control device
- `open`(2) - open for reading or writing
- `poll`(2) - STREAMS input/output multiplexing
- `putmsg`(2) - send a message on a Stream
- `read`(2) - read from file
- `write`(2) - write in a file
- `fattach`(3C) - attach a STREAMS file descriptor to a filename
- `fdetach`(3C) - disassociate a file name from a named Stream

F

- grantpt(3C) - grant access to the slave pseudo-terminal device
- iastream(3C) - determine if *fd* is associated with a STREAMS device
- ptsname(3C) - get the slave pseudo-terminal device name
- unlockpt(3C) - unlock a pseudo-terminal master/slave pair
- audio(7) - telephone-quality audio device
- clone(7) - open a major/minor device a STREAMS driver
- connld(7) - gain a unique, non-multiplexed connection to a server
- console(7) - STREAMS based console interface
- ldterm(7) - standard STREAMS terminal line discipline module
- log(7) - interface to STREAMS error logging and event tracing
- pckt(7) - push a PCKT module (packet mode) on the master side
- ptem(7) - process terminal ioctl messages
- sad(7) - STREAMS Administrative Driver
- streamio(7) - STREAMS ioctl commands
- timod(7) - Transport Interface cooperating STREAMS module
- tirdwr(7) - Transport Interface read/write interface STREAMS module
- ttcompat(7) - V7, 4BSD and Xenix STREAMS compatibility module
- zs(7) - Zilog 8530 SCC serial communications driver

DDI/DKI Entry Points

- close(9E) - relinquish access to a device
- fini - prepare a loadable module for loading
- info - provide information about a loadable module
- init(9E) - initialize a loadable module
- open(9E) - gain access to a device
- put(9E) - receive messages from the preceding queue
- srv(9E) - service queued messages

DDI/DKI Functions

- `OTHERQ(9F)` - get pointer to queue's partner queue
- `RD(9F)`- get pointer to the read queue
- `SAMESTR`(9F) - test if next queue is in the same stream
- `WR(9F)` - get pointer to the write queue for this module or driver
- `adjmsg`(9F) - trim bytes from a message
- `allocb`(9F) - allocate a message block
- `backq`(9F) - get pointer to the queue behind the current queue
- `bcanput`(9F) - test for flow control in specified priority band
- `bufcall`(9F) - call a function when a buffer becomes available
- `canput`(9F) - test for room in a message queue
- `canputnext`(9F) - test for room in a message queue
- `copyb`(9F) - copy a message block
- `copymsg`(9F) - copy a message
- `datamsg`(9F) - test whether a message is a date message
- `dupb`(9F) - duplicate a message block descriptor
- `dupmsg`(9F) - duplicate a message
- `enableok`(9F) - reschedule a queue for service
- `esballoc`(9F) - allocate a message block using a caller-supplied buffer
- `esbbcall`(9F) - call function when buffer is available
- `flushband`(9F) - flush messages for a specified priority band
- `flushq`(9F) - remove messages from a queue
- `freeb`(9F) - free a message block
- `freemsg`(9F) - free all message blocks in a message
- `freezestr`(9f) - freeze and unfreeze the state of a stream
- `getq`(9F) - get the next message from a queue
- `insq`(9F) - insert a message into a queue

F

- `linkb`(9F) - concatenate two message blocks
- `msgdsize`(9F) - return the number of bytes in a message
- `noenable`(9F) - prevent a queue from being scheduled
- `otherq`(9F) - get pointer to queue's partner queue
- `pullupmsg`(9F) - concatenate bytes in a message
- `putbq`(9F) - place a message at the head of a queue
- `putctl`(9F) - send a control message to a queue
- `putctl1`(9F) - send a control message with a one-byte parameter to a queue
- `putnext`(9F) - send a message to the next queue
- `putnext`ctl(9F) - send a control message to a queue
- `putq`(9F) - put a message on a queue
- `qenable`(9F) - enable a queue
- `qprocsoff`(9F)/`qprocson`(9F) - disable/enable STREAMS put and service procedures
- `qreply`(9F) - send a message on a stream in the reverse direction
- `qsize`(9F) - find the number of messages on a queue
- `rd`(9F) - get pointer to the read queue
- `rmvb`(9F) - remove a message block from a message
- `rmvq`(9F) - remove a message from a queue
- `samestr`(9F) - test if next queue is in the same stream
- `strlog`(9F) - submit messages to the log driver
- `strqget`(9F) - get information about a queue or band of the queue
- `strqset`(9F) - permits updating the value of certain streams' queues parameters
- `testb`(9F) - check for an available buffer
- `unlinkb`(9F) - remove a message block from the head of a message
- `wr`(9F) - get pointer to the write queue for this module or driver

DDI/DKI Data Structures

- `List`(9S) - List of structures
- `datab`(9S) - STREAMS message data structure
- `free_rtn`(9S) - structure that specifies a driver's message freeing routine
- `module_info`(9S)- STREAMS driver identification and limit value structure
- `msgb`(9S) - STREAMS message block structure
- `qband`(9S) - STREAMS queue flow control information structure
- `qinit`(9S) - STREAMS queue processing procedures structure
- `queue`(9S) - STREAMS queue structure
- `streamtab`(9S) - STREAMS entity declaration structure

F