

Linux Fast-STREAMS Porting Guide

Version 0.7a Edition 4
Updated 2006-01-04
Package streams-0.7a.4

Brian Bidulock <bidulock@openss7.org> for
The OpenSS7 Project <<http://www.openss7.org/>>

Copyright © 2001-2005 OpenSS7 Corporation <<http://www.openss7.com/>>
Copyright © 1997-2000 Brian F. G. Bidulock <bidulock@openss7.org>
All Rights Reserved.

Published by OpenSS7 Corporation
1469 Jefferys Crescent
Edmonton, Alberta T6L 6T1
Canada

This is texinfo edition 4 of the Linux Fast-STREAMS documentation, and is consistent with streams 0.7a. This manual was developed under the **OpenSS7 Project** and was funded in part by **OpenSS7 Corporation**.

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the same conditions as for modified versions.

Short Contents

Acknowledgements	1
1 Porting	3
2 Development	21
A Copying	25
B Documentation License	31
Indices	39

Table of Contents

Acknowledgements	1
Sponsors	1
Contributors	1
Introduction	1
Concepts	1
1 Porting	3
1.1 Base Functions	3
1.1.1 Message Functions	3
1.1.2 UP Queue Functions	4
1.1.3 MP Queue Functions	5
1.1.4 DDI/DKI Functions	5
1.1.5 Some Common Extension Functions	6
1.1.6 Some Internal Functions	7
1.1.7 Some Oddball Functions	7
1.2 Porting from SVR 4.2 MP	8
1.2.1 Differences from SVR 4.2 MP	8
1.2.1.1 Priority Levels	8
1.2.1.2 Basic Locks	8
1.2.2 Commonalities with SVR 4.2 MP	8
1.2.3 Compatibility functions for SVR 4.2 MP	9
1.2.4 Configuration ala SVR 4.2 MP	11
1.3 Porting from AIX 5L Version 5.1	12
1.3.1 Differences from AIX 5L Version 5.1	12
1.3.2 Commonalities with AIX 5L Version 5.1	12
1.3.3 Compatibility Functions for AIX 5L Version 5.1	12
1.3.4 Configuration ala AIX 5L Version 5.1	12
1.4 Porting from HP-UX 11.0i v2	13
1.4.1 Differences from HP-UX 11.0i v2	13
1.4.2 Commonalities with HP-UX 11.0i v2	13
1.4.3 Compatibility Functions for HP-UX 11.0i v2	13
1.4.4 Configuration ala HP-UX 11.0i v2	13
1.5 Porting from OSF/1 1.2/Digital UNIX	14
1.5.1 Differences from OSF/1 1.2/Digital UNIX	14
1.5.2 Commonalities with OSF/1 1.2/Digital UNIX	14
1.5.3 Compatibility Functions for OSF/1 1.2/Digital UNIX	14
1.5.4 Configuration ala OSF/1 1.2/Digital UNIX	14
1.6 Porting from UnixWare 7.1.3 (OpenUnix 8)	15
1.6.1 Differences from UnixWare 7.1.3 (OpenUnix 8)	15
1.6.2 Commonalities with UnixWare 7.1.3 (OpenUnix 8)	15
1.6.3 Compatibility Functions for UnixWare 7.1.3 (OpenUnix 8)	15
1.6.4 Configuration ala UnixWare 7.1.3 (OpenUnix 8)	15

1.7	Porting from Solaris 9/SunOS 5.9	16
1.7.1	Differences from Solaris 9/SunOS 5.9	16
1.7.2	Commonalities with Solaris 9/SunOS 5.9	16
1.7.3	Compatibility Functions for Solaris 9/SunOS 5.9	16
1.7.4	Configuration ala Solaris 9/SunOS 5.9	17
1.8	Porting from SUPER-UX	18
1.8.1	Differences from SUPER-UX	18
1.8.2	Commonalities with SUPER-UX	18
1.8.3	Compatibility Functions for SUPER-UX	18
1.8.4	Configuration ala SUPER-UX	18
1.9	Porting from UXP/V	19
1.9.1	Differences from UXP/V	19
1.9.2	Commonalities with UXP/V	19
1.9.3	Compatibility Functions for UXP/V	19
1.9.4	Configuration ala UXP/V	19
1.10	Porting from Linux STREAMS (LiS) 2.18.1	20
1.10.1	Differences from LiS 2.18.1	20
1.10.2	Commonalities with LiS 2.18.1	20
1.10.3	Compatibility Functions for LiS 2.18.1	20
1.10.4	Configuration ala LiS 2.18.1	20
2	Development	21
2.1	Portable <i>STREAMS</i> Drivers and Modules	21
2.1.1	Memory Allocation	21
2.1.2	Alignment of Message Buffers	21
2.1.3	Disabling and Enabling Queue Procedures	21
2.1.4	Freezing and Unfreezing Streams	21
2.1.5	Passing Messages from Interrupt Service Routines	21
2.1.6	Timeout Call Back and Link Identifiers	21
2.1.7	Synchronization with Timeouts and Callback Functions ...	22
2.1.8	Synchronization with Callout Functions	22
2.1.9	Synchronization of Drivers and Modules	22
2.1.10	Special <i>STREAMS</i> Message Types	22
2.1.11	Use of Message Allocation Priorities	22
2.1.12	Registration/Deregistration and Device Numbering	22
Appendix A	Copying	25
A.1	GNU General Public License	25
A.1.1	Preamble	25
A.1.2	Terms and Conditions for Copying, Distribution and Modification	26
A.1.3	How to Apply These Terms to Your New Programs	30

Appendix B Documentation License..... 31

B.1	GNU Free Documentation License.....	31
B.1.1	Preamble	31
B.1.2	Terms and Conditions for Copying, Distribution and Modification	31
B.1.3	How to use this License for your documents	37

Indices 39

Concept Index	40
Type Index	41
Function and Macro Index	42
Variable and Constant Macro Index	45
Program and File Index	46

Acknowledgements

As with most open source projects, this project would not have been possible without the valiant efforts and productive software for the *Free Software Foundation* and the *Linux Kernel Community*.

Sponsors

Funding for completion of the Linux Fast-STREAMS package was provided in part by:

- OpenSS7 Corporation

Additional funding for **The OpenSS7 Project** was provided by:

- OpenSS7 Corporation
- Lockheed Martin
- Performance Technologies
- Motorola
- HOB International
- Comverse
- Sonus Networks
- France Telecom
- SS8 Networks
- Nortel Networks
- Verisign

Contributors

The primary contributor to the OpenSS7 Linux Fast-STREAMS package is **Brian F. G. Bidulock**. The following is a list of significant contributors to **The OpenSS7 Project**:

- Per Berquist
- John Boyd
- Chuck Winters
- Peter Courtney
- Tom Chandler
- Gurol Ackman
- Kutluk Testicioglu
- Others

Introduction

Concepts

1 Porting

Although each of the manual pages of supported functions and structures provides compatibility and porting information, this document attempts to gather together pertinent information concerning porting from various UNIX operating system supporting STREAMS.

The porting information is organized by the operating system from which porting is being attempted. Note that, aside from configuration details, any system not listed here that is based on SVR 4.2 MP or on another of the implementations, should start with that implementation's portability information.

Porting information is organized into sections as follows:

1.1 Base Functions

1.1.1 Message Functions

- adjmsg(9)** trim bytes from the front or back of a *STREAMS* message
- allocb(9)** allocate a *STREAMS* message and data block
- bufcall(9)** install a buffer callback
- copyb(9)** copy a *STREAMS* message block
- copymsg(9)**
copy a *STREAMS* message
- datamsg(9)**
tests a *STREAMS* message type for data
- dupb(9)** duplicate a *STREAMS* message block
- dupmsg(9)**
duplicate a *STREAMS* message
- esballoc(9)**
allocate a *STREAMS* message and data block with a caller supplied data buffer
- freeb(9)** frees a *STREAMS* message block
- freemsg(9)**
frees a *STREAMS* message
- linkb(9)** link a message block to a *STREAMS* message
- msgdsz(9)**
calculate the size of the data in a *STREAMS* message
- msgpullup(9)**
pull up bytes in a *STREAMS* message
- pcmsg(9)** test a data block message type for priority control
- pullupmsg(9)**
pull up the bytes in a *STREAMS* message
- rmvb(9)** remove a message block from a *STREAMS* message

testb(9) test if a *STREAMS* message can be allocated

unbufcall(9)
 remove a *STREAMS* buffer callback

unlinkb(9) unlink a message block from a *STREAMS* message

1.1.2 UP Queue Functions

backq(9) find the upstream or downstream queue

bcanput(9)
 test flow control on a *STREAMS* message queue

canenable(9)
 test whether a *STREAMS* message queue can be scheduled

enableok(9)
 allow a *STREAMS* message queue to be scheduled

flushband(9)
 flushes band *STREAMS* messages from a message queue

flushq(9) flushes messages from a *STREAMS* message queue

getq(9) gets a message from a *STREAMS* message queue

inq(9) inserts a message into a *STREAMS* message queue

noenable(9)
 disable a *STREAMS* message queue from being scheduled

OTHERQ(9)
 return the other queue of a *STREAMS* queue pair

putbq(9) put a message back on a *STREAMS* message queue

putctl(9) put a control message on a *STREAMS* message queue

putctl1(9) put a 1 byte control message on a *STREAMS* message queue

putq(9) put a message on a *STREAMS* message queue

qenable(9) schedules a *STREAMS* message queue service routine

qreply(9) replays to a message from a *STREAMS* message queue

qsize(9) return the number of message on a queue

RD(9) return the read queue of a *STREAMS* queue pair

rmvq(9) remove a message from a *STREAMS* message queue

SAMESTR(9)
 test for *STREAMS* pipe or fifo

WR(9) return the write queue of a *STREAMS* queue pair

1.1.3 MP Queue Functions

- bcanputnext(9)**
test flow control on a message queue
- canputnext(9)**
test flow control on a message queue
- freezestr(9)**
freeze the state of a stream queue
- put(9)** invoke the put procedure for a *STREAMS* module or driver with a *STREAMS* message
- putnext(9)**
put a message on the downstream *STREAMS* message queue
- putnextctl1(9)**
put a 1 byte control message on the downstream *STREAMS* message queue
- putnextctl(9)**
put a control message on the downstream *STREAMS* message queue
- qprocsoff(9)**
disables *STREAMS* message queue processing for multi-processing
- qprocson(9)**
enables *STREAMS* message queue processing for multi-processing
- strqget(9)** gets information about a *STREAMS* message queue
- strqset(9)** sets attributes of a *STREAMS* message queue
- unfreezestr(9)**
thaw the state of a stream queue

1.1.4 DDI/DKI Functions

- kmem_alloc(9)**
allocate kernel memory
- kmem_free(9)**
deallocates kernel memory
- kmem_zalloc(9)**
allocate and zero kernel memory
- cmn_err(9)**
print a kernel command error
- bcopy(9)** copy byte strings
- bzero(9)** zero a byte string
- copyin(9)** copy user data in from user space to kernel space
- copyout(9)**
copy user data in from kernel space to user space

delay(9) postpone the calling process for a number of clock ticks

drv_getparm(9)
driver retrieve kernel parameter

drv_hztomsec(9)
convert kernel tick time between microseconds or milliseconds

drv_htztousec(9)
convert kernel tick time between microseconds or milliseconds

drv_msectohz(9)
convert kernel tick time between microseconds or milliseconds

drv_priv(9)
check if the current process is privileged

drv_usectohz(9)
convert kernel tick time between microseconds or milliseconds

drv_usecwait(9)
delay for a number of microseconds

min(9) determine the minimum of two integers

max(9) determine the maximum of two integers

getmajor(9)
get the internal major device number for a device

getminor(9)
get the extended minor device number for a device

makedevice(9)
create a device from a major and minor device numbers

strlog(9) pass a message to the *STREAMS* logger

timeout(9)
start a timer

untimeout(9)
stop a timer

mknod(9) make block or character special files

mount(9) mount and unmount filesystems

umount(9)
mount and unmount filesystems

unlink(9) remove a file

1.1.5 Some Common Extension Functions

linkmsg(9)
link a message block to a *STREAMS* message

putctl2(9) put a two byte control message on a *STREAMS* message queue

putnextctl2(9) put a two byte control message on the downstream *STREAMS* message queue

weldq(9) weld two (or four) queues together

unweldq(9) unweld two (or four) queues

1.1.6 Some Internal Functions

allocq(9) allocate a *STREAMS* queue pair

bcanget(9) test for message arrival on a band on a stream

canget(9) test for message arrival on a stream

freeq(9) deallocate a *STREAMS* queue pair

qattach(9) attach a module onto a *STREAMS* file

qclose(9) close a *STREAMS* module or driver

qdetach(9) detach a module from a *STREAMS* file

qopen(9) call a *STREAMS* module or driver open routine

setq(9) set sizes and procedures associated with a *STREAMS* message queue

1.1.7 Some Oddball Functions

appq(9) append one *STREAMS* message after another

esbbcall(9) install a buffer callback for an extended *STREAMS* message block

isdatabl(9) test a *STREAMS* data block for data type

isdatamsg(9) test a *STREAMS* data block for data type

kmem_zalloc_node(9)

msgsize(9) calculate the size of the message blocks in a *STREAMS* message

qcountstrm(9) add all counts on all *STREAMS* message queues in a stream

xmsgsize(9) calculate the size of message blocks in a *STREAMS* message

1.2 Porting from SVR 4.2 MP

This section captures portability information for SVR 4.2 MP based systems. If the operating system from which you are porting more closely fits one of the other portability sections, please see that section.

1.2.1 Differences from SVR 4.2 MP

Linux Fast-STREAMS has very few differences from *SVR 4.2 MP*. Not all *SVR 4.2 MP* functions are implemented in the base **Linux Fast-STREAMS** kernel modules. Some functions are included in the *SVR 4.2 MP* compatibility module, `'streams-svr4compat.o'`.

1.2.1.1 Priority Levels

Linux has a different concept of priority levels than *SVR 4.2 MP*. **Linux** has basically 4 priority levels as follows:

1. Preemptive

At this priority level, software and hardware interrupts are enabled and the kernel is executing with preemption enabled. This means that the currently executing kernel thread could preempt and sleep in favor of another thread of kernel execution.

This priority level only exists on preemptive (mostly 2.6) kernels.

2. Non-Preemptive

At this priority level, software and hardware interrupts are enabled and the kernel is executing with preemption disabled. This means that the currently executing kernel thread will only be interrupted by software or hardware interrupts.

This priority level exists in all kernels.

3. Software Interrupts Disabled

At this priority level, software interrupts are disabled and the kernel is executing with preemption disabled. This means that the currently executing kernel thread will only be interrupted by hardware interrupts.

This is the case when the executing thread is processing a software interrupt, or when the currently executing thread has disabled software interrupts.

This priority level exists in all kernels.

4. Interrupt Service Routines Disabled

At this priority level, hardware interrupts are disabled and the kernel is executing with preemption disabled. This means that the currently executing kernel thread will not be interrupted.

This is the case when the executing thread is processing a hardware interrupt, or when the currently executing thread has disabled hardware interrupts.

This priority level exists in all kernels.

1.2.1.2 Basic Locks

1.2.2 Commonalities with SVR 4.2 MP

1.2.3 Compatibility functions for SVR 4.2 MP

ATOMIC_INT_ADD(9)

add an integer value to an atomic integer

ATOMIC_INT_ALLOC(9)

allocate and initialize an atomic integer

ATOMIC_INT_DEALLOC(9)

deallocate an atomic integer

ATOMIC_INT_DECR(9)

decrement and test an atomic integer

ATOMIC_INT_INCR(9)

increment an atomic integer

ATOMIC_INT_INIT(9)

initialize an atomic integer

ATOMIC_INT_READ(9)

read an atomic integer

ATOMIC_INT_SUB(9)

subtract an integer value from an atomic integer

ATOMIC_INT_WRITE(9)

write an integer value to an atomic integer

itimeout(9)

perform a timeout at an interrupt level

lbolt(9)

time in ticks since reboot

LOCK(9) lock a basic lock**LOCK_ALLOC(9)**

allocate a basic lock

LOCK_DEALLOC(9)

deallocate a basic lock

LOCK_OWNED(9)

determine whether a basic lock is held by the caller

MPSTR_QLOCK(9)

release a queue from exclusive access

MPSTR_QRELE(9)

acquire a queue for exclusive access

MPSTR_STPLOCK(9)

acquire a stream head for exclusive access

MPSTR_STPRELE(9)

release a stream head from exclusive access

major(9) get the internal major number of a device

makedev(9) make a device number from internal major and minor device numbers

minor(9) get the internal minor number of a device

RW_ALLOC(9) allocate and initialize a read/write lock

RW_DEALLOC(9) deallocate a read/write lock

RW_RDLOCK(9) acquire a read/write lock in read mode

RW_TRYRDLOCK(9) attempt to acquire a read/write lock in read mode

RW_TRYWRLOCK(9) attempt to acquire a read/write lock in write mode

RW_UNLOCK(9) release a read/write lock

RW_WRLOCK(9) acquire a read/write lock in write mode

sleep(9) put a process to sleep

SLEEP_ALLOC(9) allocate a sleep lock

SLEEP_DEALLOC(9) deallocate a sleep lock

SLEEP_LOCK(9) acquire a sleep lock

SLEEP_LOCKAVAIL(9) determine whether a sleep lock is available

SLEEP_LOCKOWNED(9) determine whether a sleep lock is held by the caller

SLEEP_LOCK_SIG(9) acquire a sleep lock

SLEEP_TRYLOCK(9) attempt to acquire a sleep lock

SLEEP_UNLOCK(9) release a sleep lock

spl0(9) set priority level 0

spl1(9) set priority level 1

spl2(9) set priority level 2

spl3(9) set priority level 3

spl4(9) set priority level 4
spl5(9) set priority level 5
spl6(9) set priority level 6
spl7(9) set priority level 7
spl(9) set priority level
splx(9) set priority level x
SV_ALLOC(9)
 allocate a basic condition variable
SV_BROADCAST(9)
 broadcast a basic condition variable
SV_DEALLOC(9)
 deallocate a basic condition variable
SV_SIGNAL(9)
 signal a basic condition variable
SV_WAIT(9)
 wait on a basic condition variable
SV_WAIT_SIG(9)
 interruptible wait on a basic condition variable
TRYLOCK(9)
 try to lock a basic lock
UNLOCK(9)
 unlock a basic lock
vtop(9) convert virtual to physical address
wakeup(9) wake a process

1.2.4 Configuration ala SVR 4.2 MP

1.3 Porting from AIX 5L Version 5.1

1.3.1 Differences from AIX 5L Version 5.1

1.3.2 Commonalities with AIX 5L Version 5.1

1.3.3 Compatibility Functions for AIX 5L Version 5.1

putctl2(9) put a 2 byte control message on a *STREAMS*

splstr(9) set or restore priority levels

splx(9) set or restore priority levels

unweldq(9)
unweld two pairs of streams queues

weldq(9) weld together two paris of streams queues

mi_bufcall(9)
reliable alternative to **bufcall**(9)

mi_close_comm(9)
STREAMS common minor device close utility

mi_next_ptr(9)
STREAMS minor device list traversal

mi_open_comm(9)
STREAMS common minor device open utility

mi_prev_ptr(9)
STREAMS minor device list traversal

str_install(9)
install a *STREAMS* module or driver

wantio(9) perform direct I/O from a *STREAMS* driver

wantmsg(9)
provide a filter of wanted messages from a *STREAMS* module

1.3.4 Configuration ala AIX 5L Version 5.1

1.4 Porting from HP-UX 11.0i v2

1.4.1 Differences from HP-UX 11.0i v2

1.4.2 Commonalities with HP-UX 11.0i v2

1.4.3 Compatibility Functions for HP-UX 11.0i v2

putctl2(9) put a 2 byte control message on a *STREAMS* message queue

putnextctl2(9)
put a 2 byte control message on the downstream *STREAMS* message queue

unweldq(9)
unweld two pairs of streams queues

weldq(9) weld together two paris of streams queues

str_install(9)
install a *STREAMS* module or driver

str_uninstall(9)
uninstall a *STREAMS* module or driver

streams_get_sleep_lock(9)
provide access to the global sleep lock

streams_put(9)
invoke the put procedure for a *STREAMS* module or driver with a *STREAMS* message

1.4.4 Configuration ala HP-UX 11.0i v2

1.5 Porting from OSF/1 1.2/Digital UNIX

1.5.1 Differences from OSF/1 1.2/Digital UNIX

1.5.2 Commonalities with OSF/1 1.2/Digital UNIX

1.5.3 Compatibility Functions for OSF/1 1.2/Digital UNIX

unweldq(9)

unweld two pairs of streams queues

weldq(9) weld together two pairs of streams queues

lbolt(9) time in ticks since reboot

puthere(9)

invoke the put procedure for a *STREAMS* module or driver with a *STREAMS* message

streams_close_comm(9)

common minor device close utility

streams_open_comm(9)

common minor device open utility

streams_open_ocomm(9)

common minor device open utility

strmod_add(9)

add a *STREAMS* module

strmod_del(9)

delete a *STREAMS* module or driver from the kernel

time(9) (undoc)

1.5.4 Configuration ala OSF/1 1.2/Digital UNIX

1.6 Porting from UnixWare 7.1.3 (OpenUnix 8)

1.6.1 Differences from UnixWare 7.1.3 (OpenUnix 8)

1.6.2 Commonalities with UnixWare 7.1.3 (OpenUnix 8)

1.6.3 Compatibility Functions for UnixWare 7.1.3 (OpenUnix 8)

The following compatibility functions are in addition to all SVR 4.2 compatibility functions.

allocb_physreq(9)

allocate a *STREAMS* message and data block

emajor(9) get the external (real) majore device number from the device number

eminor(9) get the external extended minor device number from the device number

etoimajor(9)

convert an external major device number to an internal major device number

getemajor(9)

get the external (real) majore device number

geteminor(9)

get the external minor device number

itoemajor(9)

convert an internal major device number to an external major device number

msgphysreq(9)

cause a message block to meet physical requirements

msgpullup_physreq(9)

pull up bytes in a *STREAMS* message

msgscgth(9)

(undoc)

striocall(9)

(undoc)

1.6.4 Configuration ala UnixWare 7.1.3 (OpenUnix 8)

1.7 Porting from Solaris 9/SunOS 5.9

1.7.1 Differences from Solaris 9/SunOS 5.9

1.7.2 Commonalities with Solaris 9/SunOS 5.9

1.7.3 Compatibility Functions for Solaris 9/SunOS 5.9

_fini(9)

_info(9)

_init(9)

attach(9) attach a device to the system or resume a suspended device

ddi_create_minor_node(9)
create a minor node for this device

ddi_driver_major(9)
find the major device number associated with a driver

ddi_driver_name(9)
return normalized driver name

ddi_get_cred(9)
get a reference to the credentials of the current user

ddi_getiminor(9)

ddi_get_instance(9)
get device instance number

ddi_get_lbolt(9)
get the current value of the system tick clock

ddi_get_pid(9)

ddi_get_soft_state(9)

ddi_get_time(9)
get the current time in seconds since the epoch

ddi_remove_minor_node(9)
remove a minor node for a device

ddi_removing_power(9)

ddi_soft_state(9)

ddi_soft_state_fini(9)

ddi_soft_state_free(9)

ddi_soft_state_init(9)

ddi_soft_state_zalloc(9)

ddi_umem_alloc(9)
allocate page aligned kernel memory

ddi_umem_free(9)

detach(9) detach a device from the system or suspend a device

getinfo(9)

identify(9) determine if a driver is associated with a device

install_driver(9)

install a device driver

mod.info(9)

provides information on a loadable kernel module to the *STREAMS* executive

mod_install(9)

installs a loadable kernel module in the *STREAMS* executive

mod_remove(9)

removes a loadable module from the *STREAMS* executive

power(9) power a device attached to the system

probe(9)

qbufcall(9)

install a buffer callback

qtimeout(9)

start a timer associated with a queue

queclass(9)

qunbufcall(9)

quntimeout(9)

qwait(9) wait for a queue message

qwait_sig(9)

wait for a queue message or signal

qwriter(9)

1.7.4 Configuration ala Solaris 9/SunOS 5.9

1.8 Porting from SUPER-UX

1.8.1 Differences from SUPER-UX

1.8.2 Commonalities with SUPER-UX

1.8.3 Compatibility Functions for SUPER-UX

`lbolt(9)` time in ticks since reboot

1.8.4 Configuration ala SUPER-UX

1.9 Porting from UXP/V

1.9.1 Differences from UXP/V

1.9.2 Commonalities with UXP/V

1.9.3 Compatibility Functions for UXP/V

1.9.4 Configuration ala UXP/V

1.10 Porting from Linux STREAMS (LiS) 2.18.1

1.10.1 Differences from LiS 2.18.1

1.10.2 Commonalities with LiS 2.18.1

1.10.3 Compatibility Functions for LiS 2.18.1

lis_appq(9)

append one *STREAMS* message after another

lis_date(9)

lis_esbcall(9)

install a buffer callback for an extended *STREAMS* message block

lis_find_strdev(9)

lis_mknod(9)

make block or character special files

lis_mount(9)

mount a file system

lis_OTHER(9)

return the other queue of a *STREAMS* queue pair

lis_register_strdev(9)

register a *STREAMS* device

lis_register_strmod(9)

register a *STREAMS* module

lis_umount2(9)

unmount a file system

lis_umount(9)

unmount a file system

lis_unlink(9)

remove a file

lis_unregister_strdev(9)

unregister a *STREAMS* device

lis_unregister_strmod(9)

unregister a *STREAMS* module

lis_version(9)

lis_xmsgsize(9)

calculate the size of message blocks in a *STREAMS* message

1.10.4 Configuration ala LiS 2.18.1

2 Development

2.1 Portable *STREAMS* Drivers and Modules

In the process of creating the **Linux Fast-STREAMS** subsystem in such a way so as to facilitate portability of *STREAMS* drivers and modules from a wide range of *UNIX* operating system variants, a number of guidelines for the development of portable *STREAMS* drivers and modules have been developed. These guidelines, when adhered to, will allow the resulting driver or module to be ported to another *STREAMS* implementation with minimal effort. These portability guidelines are collected here.

2.1.1 Memory Allocation

Portable *STREAMS* modules and drivers will always allocate memory using the SVR4 memory allocators/deallocators: **kmem_alloc(9)**, **kmem_zalloc(9)** and **kmem_free(9)**.

Additional eligible allocators are:

rmallocmap(9) **rmfreemap(9)** **rmalloc(9)** **rmalloc_wait(9)** **rmfree(9)** **rminit(9)** **rmsetwant(9)** **rmwanted(9)**

Unfortunately, these resource map allocators are not available on *AIX* so, if portability to the *AIX PSE* is important, then do not use these allocators.

Additional eligible allocators are:

kmem_fast_alloc(9) **kmem_fast_free(9)**

2.1.2 Alignment of Message Buffers

2.1.3 Disabling and Enabling Queue Procedures

Portable *STREAMS* modules and drivers will always call **qprocson(9)** before returning from its queue open procedure (see **qopen(9)**).

Portable *STREAMS* modules and drivers will always call **qprocsoff(9)** upon entering its queue close procedure (see **qclose(9)**).

2.1.4 Freezing and Unfreezing Streams

2.1.5 Passing Messages from Interrupt Service Routines

2.1.6 Timeout Call Back and Link Identifiers

Although buffer callbacks identifiers (see **bufcall(9)**), timeout identifiers (see **timeout(9)**), and multiplexing driver link identifiers (see *L_LINK* and *L_PLINK* under **streamio(2)**), are often illustrated as small integer numbers, with some *STREAMS* implementations, including **Linux Fast-STREAMS**, these identifiers are kernel addresses (pointers) and are never small integer values like 1, 2, or 3.

Also, there is no guarantee that the identifier will be positive. It is guaranteed that the returned identifier will not be zero (0). Zero is used by these function as a return value to indicate an error.

Portable *STREAMS* drivers and modules will not depend upon the returned identifier from **bufcall(9)**, **timeout(9)** or **streamio(2)** as being any specific range of value. Portable drivers

and modules will save any returned identifiers in data types that will not lose the precision of the identifier.

2.1.7 Synchronization with Timeouts and Callback Functions

2.1.8 Synchronization with Callout Functions

2.1.9 Synchronization of Drivers and Modules

2.1.10 Special *STREAMS* Message Types

2.1.11 Use of Message Allocation Priorities

2.1.12 Registration/Deregistration and Device Numbering

UNIX Device Numbering

In versions of *UNIX System V* previous to *Release 4*, the major and minor device numbers were each 8 bit, and they were packed into a 16 bit word (usually a C Language *short* variable). Under *UNIX System V Release 4*, the device numbers are held in a `'dev_t'` variable, which is often implemented as a 32 bit integer. The minor device number is held as 14 bits, and a further 8 bits are used for the major device number. `'dev_t'` is often referred to as the "expanded device type", since it allows many more minor devices than before.

Many drivers were written for earlier releases, and may eventually be ported to *UNIX System V Release 4*. In earlier releases, some manufacturers got around the 256 minor device number limit by using multiple major device numbers for a device. Devices were created with different major device numbers (the external major device number) but they all mapped to the same device driver entry in the device switch tables (the internal device number). Even under this scheme, each major device could only support 256 minor devices, but the driver could support many more. This has been recognized in *UNIX System V Release 4*, and functions are provided to do this mapping; for example, the function `etoimajor()` and so on, give a machine independent interface to the device number mapping.¹

Linux Device Numbering

Versions of the **Linux** kernel in the 2.4 kernel series and prior to 2.6 also provided an 8 bit major device number and an 8 bit minor device number grouped into a 16-bit combined device number. Linux 2.6 kernels (and some patched 2.4 kernels) now have larger device numbers. These extended device numbers are 12 bits for major device number and 20 bits for minor device number, with 32 bits for the combined device number.

LiS Device Numbering

LiS prior to the 2.18.0 release was incapable of providing an internal representation of the device number and the number of minor device numbers for a device driver was restricted to

¹ *The Magic Garden Explained*

256.² Many **OpenSS7** device drivers written for *LiS* would allocate additional major device numbers if required. Good examples of devices that require more than 255 minor device numbers are INET clone devices, SCTP streams, SS7 signalling link streams, MG media channels, etc. These streams are often *LPLINK*ed under a multiplexing driver and do not even consume a system file descriptor.

Linux Fast-STREAMS Device Numbering

Linux Fast-STREAMS began with extended device numbering. The ‘**specfs**’ shadow special character device filesystem used by **Linux Fast-STREAMS** uses the ‘**inode**’ number to hold the ‘**dev_t**’ device number instead of the ‘**inode->i_rdev**’, which on older kernels is only a 16-bit *short*.

In earlier versions of **Linux Fast-STREAMS**, the internal device numbering is 16-bits for major device number and 16-bits for minor device number. This will soon be changed to 12-bits for major device number and 20-bits for minor device number to accomodate the newer **Linux** scheme.

On 2.6 **Linux** kernels that support the newer extended device numbers, external device numbers and internal device numbers will be the same. On 2.4 **Linux** kernels with the older 16-bit device numbers, external device number and internal device numbers will differ. In some situations, an internal device number can exist with no corresponding external device number (accessed only via a clone device or direct access to the mounted ‘**specfs**’ shadow special character device filesystem).

etoimajor(9)

change external to internal major device number

getemajor(9)

get external major device number

geteminor(9)

get external minor device number

itoemajor(9)

change internal to external major device number

² Actually, 255 as the kernel reserved minor device number 255 for expansion.

Appendix A Copying

A.1 GNU General Public License

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright © 1989, 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

A.1.1 Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation’s software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author’s protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors’ reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone’s free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

A.1.2 Terms and Conditions for Copying, Distribution and Modification

1. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The “Program”, below, refers to any such program or work, and a “work based on the Program” means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term “modification”.) Each licensee is addressed as “you”.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

2. You may copy and distribute verbatim copies of the Program’s source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

3. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
 - a. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
 - b. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
 - c. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program,

the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

4. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
 - a. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
 - b. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
 - c. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

5. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.
6. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore,

by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

7. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.
8. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

9. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.
10. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a

version number of this License, you may choose any version ever published by the Free Software Foundation.

11. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

12. **BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.**
13. **IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.**

END OF TERMS AND CONDITIONS

A.1.3 How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

```
one line to give the program's name and an idea of what it does.
Copyright (C) 19yy name of author
```

```
This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details
type 'show w'. This is free software, and you are welcome
to redistribute it under certain conditions; type 'show c'
for details.
```

The hypothetical commands ‘show w’ and ‘show c’ should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than ‘show w’ and ‘show c’; they could even be mouse-clicks or menu items—whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a “copyright disclaimer” for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright
interest in the program 'Gnomovision'
(which makes passes at compilers) written
by James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Appendix B Documentation License

B.1 GNU Free Documentation License

GNU FREE DOCUMENTATION LICENSE

Version 1.1, March 2000

Copyright © 2000 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

B.1.1 Preamble

The purpose of this License is to make a manual, textbook, or other written document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

B.1.2 Terms and Conditions for Copying, Distribution and Modification

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The

relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front

cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document’s license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section entitled “History”, and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled “History” in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the “History” section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. In any section entitled “Acknowledgments” or “Dedications”, preserve the section’s title, and preserve in the section all the substance and tone of each of the contributor acknowledgments and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled “Endorsements”. Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section as “Endorsements” or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled “History” in the various original documents, forming one section entitled “History”; likewise combine any sections entitled “Acknowledgments”, and any sections entitled “Dedications”. You must delete all sections entitled “Endorsements.”

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an “aggregate”, and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document’s Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that

you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

END OF TERMS AND CONDITIONS

B.1.3 How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  year  your name.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.1
or any later version published by the Free Software Foundation;
with the Invariant Sections being list their titles, with the
Front-Cover Texts being list, and with the Back-Cover Texts being list.
A copy of the license is included in the section entitled ‘‘GNU
Free Documentation License’’.
```

If you have no Invariant Sections, write “with no Invariant Sections” instead of saying which ones are invariant. If you have no Front-Cover Texts, write “no Front-Cover Texts” instead of “Front-Cover Texts being *list*”; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Indices

Concept Index

B

Base Functions	3
basic locks	8

C

commonalities with aix 5l version 5.1	12
commonalities with hp-ux 11.0i v2	13
commonalities with lis 2.18.1	20
commonalities with osf/1 1.2/digial unix	14
commonalities with solaris 9/sunos 5.9	16
commonalities with super-ux	18
commonalities with svr 4.2 mp	8
commonalities with unixware 7.1.3 (openunix 8)	15
commonalities with uxp/v	19
compatibility functions for aix 5l version 5.1	12
compatibility functions for hp-ux 11.0i v2	13
compatibility functions for lis 2.18.1	20
compatibility functions for osf/1 1.2/digial unix	14
compatibility functions for solaris 9/sunos 5.9	16
compatibility functions for super-ux	18
compatibility functions for svr 4.2 mp	9
compatibility functions for unixware 7.1.3 (openunix 8)	15
compatibility functions for uxp/v	19
concepts	1
configuration ala aix 5l version 5.1	12
configuration ala hp-ux 11.0i v2	13
configuration ala lis 2.18.1	20
configuration ala osf/1 1.2/digial unix	14
configuration ala solaris 9/sunos 5.9	17
configuration ala super-ux	18
configuration ala svr 4.2 mp	11
configuration ala unixware 7.1.3 (openunix 8)	15
configuration ala uxp/v	19
contributors	1
credits	1

D

DDI/DKI functions	5
development	21
differences from aix 5l version 5.1	12
differences from hp-ux 11.0i v2	13
differences from lis 2.18.1	20
differences from osf/1 1.2/digial unix	14

differences from solaris 9/sunos 5.9	16
differences from super-ux	18
differences from svr 4.2 mp	8
differences from unixware 7.1.3 (openunix 8)	15
differences from uxp/v	19

I

introduction	1
--------------------	---

L

license, FDL	31
license, GNU Free Documentation License	31
license, GNU General Public License	25
license, GPL	25

M

message functions	3
MP queue functions	5

P

portable <i>STREAMS</i> drivers and modules	21
porting	3
Porting from AIX 5L Version 5.1	12
Porting from HP-UX 11.0i v2	13
Porting from Linux STREAMS (LiS) 2.18.1	20
Porting from OSF/1 1.2/Digital UNIX	14
Porting from Solaris 9/SunOS 5.9	16
Porting from SUPER-UX	18
Porting from SVR 4.2 MP	8
Porting from UnixWare 7.1.3 (OpenUnix 8)	15
Porting from UXP/V	19
priority levels	8

S

some common extension functions	6
some internal functions	7
some oddball functions	7
sponsors	1

U

UP queue functions	4
--------------------------	---

Type Index

`dev_t` [22](#), [23](#)

Function and Macro Index

–		
_fini(9)	16	
_info(9)	16	
_init(9)	16	
A		
adjmsg(9)	3	
alloca(9)	3	
alloca_physreq(9)	15	
allocq(9)	7	
appq(9)	7	
ATOMIC_INT_ADD(9)	9	
ATOMIC_INT_ALLOC(9)	9	
ATOMIC_INT_DEALLOC(9)	9	
ATOMIC_INT_DECR(9)	9	
ATOMIC_INT_INCR(9)	9	
ATOMIC_INT_INIT(9)	9	
ATOMIC_INT_READ(9)	9	
ATOMIC_INT_SUB(9)	9	
ATOMIC_INT_WRITE(9)	9	
attach(9)	16	
B		
backq(9)	4	
bcanget(9)	7	
bcanput(9)	4	
bcanputnext(9)	5	
bcopy(9)	5	
bufcall(9)	3	
bzero(9)	5	
C		
canenable(9)	4	
canget(9)	7	
canputnext(9)	5	
cmn_err(9)	5	
copyb(9)	3	
copyin(9)	5	
copymsg(9)	3	
copyout(9)	5	
D		
datamsg(9)	3	
ddi_create_minor_node(9)	16	
ddi_driver_major(9)	16	
ddi_driver_name(9)	16	
ddi_get_cred(9)	16	
ddi_get_instance(9)	16	
ddi_get_lbolt(9)	16	
ddi_get_pid(9)	16	
ddi_get_soft_state(9)	16	
ddi_get_time(9)	16	
ddi_getminor(9)	16	
ddi_remove_minor_node(9)	16	
ddi_removing_power(9)	16	
ddi_soft_state(9)	16	
ddi_soft_state_fini(9)	16	
ddi_soft_state_free(9)	16	
ddi_soft_state_init(9)	16	
ddi_soft_state_zalloc(9)	16	
ddi_umem_alloc(9)	16	
ddi_umem_free(9)	16	
delay(9)	6	
detach(9)	16	
drv_getparm(9)	6	
drv_htztousec(9)	6	
drv_hztomsec(9)	6	
drv_msectohz(9)	6	
drv_priv(9)	6	
drv_usectohz(9)	6	
drv_usecwait(9)	6	
dupb(9)	3	
dupmsg(9)	3	
E		
emajor(9)	15	
eminor(9)	15	
enableok(9)	4	
esballoc(9)	3	
esbcall(9)	7	
etoimajor(9)	15, 23	
F		
flushband(9)	4	
flushq(9)	4	
freeb(9)	3	
freemsg(9)	3	
freeq(9)	7	
freezestr(9)	5	
G		
getemajor(9)	15, 23	
geteminor(9)	15, 23	
getinfo(9)	17	
getmajor(9)	6	
getminor(9)	6	
getq(9)	4	
I		
identify(9)	17	
insq(9)	4	
install_driver(9)	17	

isdatablk(9) 7
 isdatamsg(9) 7
 itimeout(9) 9
 itoemajor(9) 15, 23

K

kmem_alloc(9) 5
 kmem_free(9) 5
 kmem_zalloc(9) 5
 kmem_zalloc_node(9) 7

L

lbolt(9) 9, 14, 18
 linkb(9) 3
 linkmsg(9) 6
 lis_appq(9) 20
 lis_date(9) 20
 lis_esbbcall(9) 20
 lis_find_strdev(9) 20
 lis_mknod(9) 20
 lis_mount(9) 20
 lis_OTHER(9) 20
 lis_register_strdev(9) 20
 lis_register_strmod(9) 20
 lis_umount(9) 20
 lis_umount2(9) 20
 lis_unlink(9) 20
 lis_unregister_strdev(9) 20
 lis_unregister_strmod(9) 20
 lis_version(9) 20
 lis_xmsgsize(9) 20
 LOCK(9) 9
 LOCK_ALLOC(9) 9
 LOCK_DEALLOC(9) 9
 LOCK_OWNED(9) 9

M

major(9) 9
 makedev(9) 10
 makedevice(9) 6
 max(9) 6
 mi_bufcall(9) 12
 mi_close_comm(9) 12
 mi_next_ptr(9) 12
 mi_open_comm(9) 12
 mi_prev_ptr(9) 12
 min(9) 6
 minor(9) 10
 mknod(9) 6
 mod_info(9) 17
 mod_install(9) 17
 mod_remove(9) 17
 mount(9) 6
 MPSTR_QLOCK(9) 9
 MPSTR_QRELE(9) 9

MPSTR_STPLOCK(9) 9
 MPSTR_STPRELE(9) 9
 msgdsz(9) 3
 msgphysreq(9) 15
 msgpullup(9) 3
 msgpullup_physreq(9) 15
 msgscgth(9) 15
 msgsize(9) 7

N

noenable(9) 4

O

OTHERQ(9) 4

P

pcmsg(9) 3
 power(9) 17
 probe(9) 17
 pullupmsg(9) 3
 put(9) 5
 putbq(9) 4
 putctl(9) 4
 putctl1(9) 4
 putctl2(9) 6, 12, 13
 puthere(9) 14
 putnext(9) 5
 putnextctl(9) 5
 putnextctl1(9) 5
 putnextctl2(9) 7, 13
 putq(9) 4

Q

qattach(9) 7
 qbufcall(9) 17
 qclose(9) 7
 qcountstrm(9) 7
 qdetach(9) 7
 qenable(9) 4
 qopen(9) 7
 qprocsoff(9) 5
 qprocson(9) 5
 qreply(9) 4
 qsize(9) 4
 qtimeout(9) 17
 queclass(9) 17
 qunbufcall(9) 17
 quntimeout(9) 17
 qwait(9) 17
 qwait_sig(9) 17
 qwriter(9) 17

R

RD(9)	4
rmvb(9)	3
rmvq(9)	4
RW_ALLOC(9)	10
RW_DEALLOC(9)	10
RW_RDLOCK(9)	10
RW_TRYRDLOCK(9)	10
RW_TRYWRLOCK(9)	10
RW_UNLOCK(9)	10
RW_WRLOCK(9)	10

S

SAMESTR(9)	4
setq(9)	7
sleep(9)	10
SLEEP_ALLOC(9)	10
SLEEP_DEALLOC(9)	10
SLEEP_LOCK(9)	10
SLEEP_LOCK_SIG(9)	10
SLEEP_LOCKAVAIL(9)	10
SLEEP_LOCKOWNED(9)	10
SLEEP_TRYLOCK(9)	10
SLEEP_UNLOCK(9)	10
spl(9)	11
spl0(9)	10
spl1(9)	10
spl2(9)	10
spl3(9)	10
spl4(9)	11
spl5(9)	11
spl6(9)	11
spl7(9)	11
splstr(9)	12
splx(9)	11, 12
str_install(9)	12, 13
str_uninstall(9)	13
streams_close_comm(9)	14
streams_get_sleep_lock(9)	13
streams_open_comm(9)	14
streams_open_ocomm(9)	14
streams_put(9)	13
striocall(9)	15
strlog(9)	6

strmod_add(9)	14
strmod_del(9)	14
strqget(9)	5
strqset(9)	5
SV_ALLOC(9)	11
SV_BROADCAST(9)	11
SV_DEALLOC(9)	11
SV_SIGNAL(9)	11
SV_WAIT(9)	11
SV_WAIT_SIG(9)	11

T

testb(9)	4
time(9)	14
timeout(9)	6
TRYLOCK(9)	11

U

umount(9)	6
unbufcall(9)	4
unfreezestr(9)	5
unlink(9)	6
unlinkb(9)	4
UNLOCK(9)	11
untimeout(9)	6
unweldq(9)	7, 12, 13, 14

V

vtop(9)	11
---------------	----

W

wakeup(9)	11
wantio(9)	12
wantmsg(9)	12
weldq(9)	7, 12, 13, 14
WR(9)	4

X

xmsgsize(9)	7
-------------------	---

Variable and Constant Macro Index

I_LINK 21 I_PLINK 21, 23

Program and File Index

(Index is nonexistent)