Index

<arpa/inet.h>....**210**
<fcntl.h> **178**
<netatm/atm.h> 405
<netdb.h> **211**
<netinet/in.h>....**213**
<sys/socket.h>....**179**, 405
<sys/stat.h> **183**
<sys/uio.h>....**184**
<sys/un.h>....**185**
<unistd.h>....**214**
<xti.h> 51, 77, 281, 405
<xti_atm.h>....405
<xti_mosi.h>....317
_XOPEN_SOURCE....3
abortive release....24, 111, 415
accept 222
accept( )....**130**
accept1 ....33, 37
accept2 ....33, 37
accept3 ....33, 37
additions to <sys/socket.h> for ATM 405
additions to <xti.h> for ATM 405
address....11-12, 19-21, 26, 29, 55
....59-60, 63-66, 71, 74, 77
....81-82, 100-101, 116-117, 215
address Class....353
application 11-14, 245, 280
applications....51
portability 245, 279
applications portability....11, 280
association-related 220
association-related options....39
asynchronous 38, 79
asynchronous events 14
asynchronous mode 14, 93, 415
ATM addresses....364
ATM levels 405
ATM options....369
ATM transport headers 405
ATM transport protocol for sockets 395
ATM transport protocol for XTI 363
ATM transport provider....363
bind....11, 28, 33, 36, 59, 128
bind( )....**132**
broadcast 353
buffer....57, 69, 91
byte order of Internet address....353
C language
Issue 4 environment....2
Call structure 286
caller....29, 64, 77
can 1
character string....11
checksum check 231
child process 12
CL....295
Class of address 353
close 28, 36-37, 62
close( ) **134**
closed....33, 36-37
cluster of local networks 353
cmsghdr....**179**
CO 295
Common Usage C....2
compatibility
future....58
compilation environment **3**
connect( ) **135**
connect1....33, 37
connect2....33, 37
connection....11, 63, 91, 93, 111, 113
connection establishment 19, 21, 63-64
....93, 229, 241, 415
connection indication 35, 59-60, 77, 95
connection mode 216
connection release....19, 24-25, 229, 241, 415
connection request....54, 77, 93, 111
connection-mode 13, 19, 21, 28, 37, 241-242, 415
connection-mode service 37
connectionless 224
connectionless-mode 13, 26, 28, 36, 100
....102, 116, 220, 244, 415
constants....51
control message protocol....353
create
transport endpoint 20
current event 31, 36, 79
current state 12, 31, 36, 76, 125
data 91, 95, 100, 108, 111, 114, 116
data transfer....19, 22, 26, 36-37, 241, 415
data unit....23, 26, 98, 100, 116
discarded....27

datagram 13, 27, 415
datagram structure 286
de-initialisation 19-20, 26, 36
default 81, 245
device 245
device driver 280
discarded data unit 27
discon 98, 114
disconnect 15, 95
  indication 33
disconnection 19, 38, 95
  request 111
disconnection structure 286
dup 11-12, 125-126
duplex 22
EBADF
  in recvmsg( ) 157
EM 246, 295
end-to-end significance 39
endhostent( ) **188**
endnetent( ) **190**
endprotoent( ) **191**
endservent( ) **192**
enqueue 21, 29
errmsg **66**
errno 66
errnum 124
error 282
error code 102, 124
error codes 281-282
  TACCES 281
  TADDRBUSY 282
  TBADADDR 281
  TBADDATA 281
  TBADF 281
  TBADFLAG 281
  TBADNAME 282
  TBADOPT 281
  TBADQLEN 282
  TBADSEQ 281
  TBUFOVFLW 281
  TFLOW 281
  TINDOUT 282
  TLOOK 281
  TNOADDR 281
  TNODATA 281
  TNODIS 281
  TNOREL 281
  TNOSTRUCTYPE 282
  TNOTSUPPORT 281
  TNOUDERR 281
  TOUTSTATE 281
  TPROTO 282
  TPROVMISMATCH 282
  TQFULL 282
  TRESADDR 282
  TRESQLEN 282
  TSTATECHNG 282
  TSYSERR 281
error handling 13
error indication 102
error message 66, 124
error number 124
error numbers 6
established
  connection 242
ETSDU 23, 81-82, 91, 105, 108-110, 295
event 31, 36, 244
  current 31, 36, 79
event management 16-17
Event Management 246
events 247, 282
  accept1 ....33, 37
  accept2 ....33, 37
  accept3 ....33, 37
  bind 33, 36
  closed 33, 36-37
  connect1 ....33, 37
  connect2 ....33, 37
  incoming 34
  listen 34, 37, 77
  opened 33, 36
  optmgmt 33, 36
  outgoing 33
  pass_conn 34, 37
  rcv 34, 37, 91
  rcvconnect 34, 37
  rcvdis1 ....34, 37
  rcvdis2 ....34, 37
  rcvdis3 ....34, 37
  rcvrel 34, 37
  rcvudata 34, 36
  rcvuderr 34, 36
  snd 33, 37
  snddis1 ....33, 37
  snddis2 ....33, 37
  sndrel 33, 37
  sndudata 33, 36
  T_CONNECT 282
  T_DATA 282
  T_DISCONNECT 282
  T_EXDATA 282

T_GODATA 282
T_GOEXDATA 282
T_LISTEN 282
T_ORDREL 282
T_UDERR 282
unbind 33, 36
events and t_look 15
example 249
example for select 257
exec 125-126
execution mode 14, 23
expedited data 19, 23, 82, 91, 105, 108, 216, 224, 226, 229, 290, 415
expedited transport service data unit 415
ETSDU 82, 91
family of protocols 353
fcntl 11, 14, 64, 77-78, 91-93, 100-101, 109, 116-117
fcntl( ) **138**
fcntl.h 81
fd 11, 33
features 28-29
fgetpos( ) **139**
file descriptor 11, 62, 71, 81, 125
file.c 51
flag 81, 84, 91
flags 81, 84, 91, 224, 282, 285
T_CHECK 282
T_CURRENT 282
T_DEFAULT 282
T_EXPEDITED 282
T_FAILURE 282
T_MORE 282
T_NEGOTIATE 282
T_NOTSUPPORT 282
T_PARTSUCCESS 282
T_PUSH 282
T_READONLY 282
T_SUCCESS 282
flow control 27
fork 11-12, 125-126
fsetpos( ) **140**
ftell( ) **141**
full duplex 22
F_GETOWN 138, 178
F_SETOWN 138, 178
General purpose defines 287
gethostbyaddr( ) **188**, 194
gethostbyname( ) **188**
gethostent( ) **188**
gethostname( ) **195**
getnetbyaddr( ) **190**, 196
getnetbyname( ) **190**
getnetent( ) **190**
getpeername( ) **142**
getprotobyname( ) **191**
getprotobynumber( ) **191**, 197
getprotoent( ) **191**
getservbyname( ) **192**
getservbyport( ) **192**, 198
getservent( ) **192**
getsockname( ) **143**
getsockopt( ) **144**
headers
<xti.h> 281
host byte order 415
host part of address 353
hostent **211**
htonl( ) **200**
htons( ) **200**
h_errno **199**
h_errno( ) **188**
ICMP 353
implementation-dependent 1
INADDR_ANY 213, 353
INADDR_BROADCAST 213
incoming events 34
inet_addr( ) **201**
inet_lnaof( ) **201**
inet_makeaddr( ) **201**
inet_netof( ) **201**
inet_network( ) **201**
inet_ntoa( ) **201**
initialisation 19-20, 26, 36, 81
initiator 19, 415
interfaces
implementation 2
use 2
Internet address
byte order 353
Internet protocol 353
Internet Protocol family 353
Internet protocol family 353
Internet protocol-specific information 229
in_addr 210, 213
in_addr_t 210-211
in_port_t 210-211, 213
IOV_MAX 283
IP 353
IP-level Options 239, 292
IPPORT_RESERVED 211

IPPROTO_ macros
defined in <netinet/in.h> 213
IP_TOS type of service 240, 293
ISO 215, 225, 288, 295
priorities 225, 288
protection levels 225, 288
transport classes 225, 288
ISO C 2
language-dependent 66
legacy 1, 415
library functions 283
library structure 57
linger **180**
listen 60, 77, 222
listen( ) **147**
listener application 12
little-endian 353
lseek( ) **148**
management options 218, 221, 227, 290
mandatory features 245
maximum size
address 82
address buffer 59, 74
buffer 64, 77, 93, 100, 102
ETSDU 82, 110
TSDU 24, 82, 110, 237
may 1
memory
allocate 57, 69
mode
asynchronous 14
connection 19, 21, 28, 37, 216, 241-242
connectionless 26, 28, 36, 100, 102
....116, 220, 244
record-oriented 24
stream-oriented 24
synchronous 14, 79
modes of service 13
mosi Header File 317
msghdr **179**
MSG_ macros
defined in <sys/socket.h> **181**
multiple options 45
must 1
name space
X/Open 3
native ATM services 363, 395
netbuf structure 40, 57, 84
netent **211**
network byte order 416
network host database 416
network net database 416
network protocol database 416
network service database 416
next state 36
ntohl( ) **200**, 203
ntohs( ) **200**
NULL 59
null
call 111
null pointer 58, 60, 66, 69, 83, 95, 102
ocnt **33**
open 81
opened 33, 36
option
value 46
option management 287
option negotiation
initiate 42
response 43
option values 216
options
association-related 39
connection mode 216
connectionless-mode 220
expedited data 216
format 48
generalities 39
illegal 41
ISO-specific 225-226, 288-289
management 218
multiple 45
privileged 45
quality of service 216
read-only 45
retrieving information 44
TCP-level 230
transport endpoint 84
transport level 9
transport provider 64
T_IP-level 232
T_UDP-level 231
unsupported 42
XTI-level 288
Options management structure **285**
options with end-to-end significance 39
optmgmt 28, 33, 36
orderly release 24, 97, 113, 416
OSI 295
transport classes 223
outgoing events 33
outstanding connection indications 35, 60, 95

O_NONBLOCK flag 14
pass_conn 34, 37
poll 79
poll( ) **149**
polling 15
portability 50
portable 11, 245, 279
precedence levels
  IP 239, 293
primitives 14-15
process 12
program 51
programs
  multiple protocol 245
protocol ....11, 29, 39, 59, 63, 71 ....74, 81, 84, 102, 215, 245
protocol independence 72, 82, 245
protocol-specific servicelimits 284
protocols in Internet family 353
protoent **211**
quality of service 216, 220, 226, 290
queue 21, 29, 78
rate 216
rate structure **225**, 289
rcv 34, 37
rcvconnect 34, 37
rcvdis1 ....34, 37
rcvdis2 ....34, 37
rcvdis3 ....34, 37
rcvrel 34, 37
rcvreldata 34
rcvudata 34
rcvuderr 34
rcvvudata 284
read( ) **150**
readv( ) **150**
reason
  disconnection 95
receipt 97
receive 91, 98, 100
Receiving Data 22, 27
record-oriented 24
recv( ) **151**
recvfrom( ) **153**
recvmsg( ) **156**
release 19, 24, 37, 97, 111, 113
reliable 13
remote user ...16, 21, 24-25, 62, 64-65, 111, 242, 244
reqvalue 216, 226, 289
reqvalue structure **226**, 289
resfd **33**
responder 19, 416
safety 6
sa_family_t **179**
select( ) **159**
send( ) **160**
Sending Data 23, 27
sendmsg( ) **162**
sendto( ) **165**
servent **211**
server program 248, 257
service definition
  ISO 24, 215, 222
  TCP 24
service type defines 285
sethostent( ) **188**, 204
setnetent( ) **190**, 205
setprotoent( ) **191**, 206
setservent( ) **192**, 207
setsockopt( ) **168**
should 1
shutdown( ) **171**
snd 33, 37-38, 215
snddis1 ....33, 37
snddis2 ....33, 37
sndrel 33, 37-38
sndreldata 33
sndudata 33, 36, 38, 215
sockaddr **179**
sockaddr_in **213**
sockaddr_un **185**
socket 235-236, 416
socket( ) **172**
socketpair( ) **174**
SOCK_DGRAM 353
SOCK_RAW 353
SOCK_STREAM 353
SO_ macros
  defined in <sys/socket.h> **180**
standard error 66
state 31-32, 36, 287
  current 31, 36, 76, 125
  next 36
  T_DATAXFER **32**, 287
  T_IDLE **32**, 287
  T_INCON **32**, 287
  T_INREL **32**, 287
  T_OUTCON **32**, 287
  T_OUTREL **32**, 287
  T_UNBIND **32**
  T_UNBND 287
  T_UNIT **32**

state table....36-37, 244
status
  connection....64
  connection request....22, 93
stream-oriented....24
strerror(3C)....66
struct netbuf....285
struct rate....225, 289
struct reqvalue....226, 289
struct thrpt....226, 289
struct transdel....226, 289
struct t_bind....285
struct t_call....286
struct t_discon....286
struct t_info....284
struct t_kpalive....238, 292
struct t_linger....288
struct t_opthdr....285
struct t_optmgmt....285
struct t_uderr....286
struct t_unitdata....286
structure types....286
  T_BIND....286
  T_CALL....286
  T_DIS....286
  T_INFO....286
  T_OPTMGMT....286
  T_UDERROR....286
  T_UNITDATA....286
subnet part of address....353
SVID....295
synchronise....125
synchronous mode....14, 79, 93, 416
t-opthdr....216
TACCES....**281**
TADDRBUSY....**282**
TBADADDR....**281**
TBADDATA....**281**
TBADF....**281**
TBADFLAG....**281**
TBADNAME....**282**
TBADOPT....**281**
TBADQLEN....**282**
TBADSEQ....**281**
TBUFOVFLW....**281**
TC....295
TCP....24, 295, 353
TCP-level options....230, 238, 291
terminated
  connection....242
terminology....**1**
TFLOW....27, 247, 281
thread cancellation point....6
thread safety....6
threads....6
thrpt....216, 226, 289
thrpt structure....**226**, 289
TINDOUT....**282**
TLI....279-280, 295
TLOOK....15, 23, 38, 281
TNOADDR....**281**
TNODATA....**281**
TNODIS....**281**
TNOREL....**281**
TNOSTRUCTYPE....**282**
TNOTSUPPORT....**281**
TNOUDERR....**281**
TOS precedence levels....239, 293
TOUTSTATE....**281**
TPDU lengths....225, 289
TPROTO....**282**
TPROVMISMATCH....**282**
TQFULL....**282**
transdel....216
transdel structure....226, 289
Transmission Control Protocol....353
transport address....11, 215, 416
transport classes....223, 225, 288
transport connection....11, 21, 62, 71, 113, 416
transport endpoint....11, 32-33, 46, 59, 62-63
  ....79, 81, 83-84, 125, 128, 416
Transport Level Interface (TLI)....279-280
transport level options....9
transport provider....11, 19, 31-32, 36, 40
  ....71, 81, 125, 215, 241, 285
transport provider identifier....**11**, 19, 81, 416
transport service....9, 215, 241
transport service access point....416
  TSAP....12
transport service data unit....416
  TSDU....16, 82, 91, 245
transport service provider....417
transport service user....11, 19, 21, 24
  ....31-32, 63, 241, 417
transport user actions....35
TRESADDR....**282**
TRESQLEN....**282**
TSAP....12, 295
TSDU....16, 23, 81-82, 91, 108-110
  ....116, 215, 224, 229, 295
TSTATECHNG....**282**
TSYSERR....13, 23, 66, 281

T_ABSREQ **287**
t_accept 28, 54, 222, 229, 235, 283
t_accept( ) **54**
T_ACTIVEPROTECT **225**, 289
t_addleaf( ) **385**
T_ADDR **287**
T_ALIGN **287**
T_ALL **287**
t_alloc 28, 57, 69, 283, 286
t_alloc( ) **57**
T_ALLOPT **287**
t_atm_sap structure 365
t_bind 11, 28, 59, 128, 222, 235, 283
T_BIND **286**
t_bind( ) **59**
t_call 54
T_CALL **286**
T_CHECK **282**
T_CLASS0 ..... **225**, 288
T_CLASS1 ..... **225**, 288
T_CLASS2 ..... **225**, 288
T_CLASS3 ..... **225**, 288
T_CLASS4 ..... **225**, 288
t_close 28, 62, 222, 241, 283
t_close( ) **62**
T_CLTS **285**
T_CONNECT 15
t_connect 15
T_CONNECT 16
t_connect 28, 63, 93, 222, 235
T_CONNECT 247, 282
t_connect 283
t_connect( ) 44, 63
T_COTS **285**
T_COTS_ORD **285**
T_CRITIC_ECP **239**, 293
T_CURRENT **282**
T_DATA 15-16, 26, 37-38, 247, 282
T_DATAXFER **32**, 37, 287
T_DEFAULT **282**
T_DIS 15, 38, 286
T_DISCONNECT 15, 17, 23, 38, 247, 282
t_errno 13, 68, 124, 282
t_error 13, 28, 66, 124, 283
t_error( ) **66**
T_EXDATA 15-16, 236, 247, 282
T_EXPEDITED 23, 91, 105, 282
T_FAILURE **282**
T_FLASH **239**, 293
t_free 28, 69, 283
t_free( ) **69**
t_getinfo 28, 71, 222, 283
t_getinfo( ) **71**
t_getprotaddr 28, 74, 283
t_getprotaddr( ) **74**
t_getstate 28, 76, 283
t_getstate( ) **76**
T_GODATA 15, 17, 26, 247, 282
T_GOEXDATA 15, 17, 247, 282
T_HIREL **240**, 293
T_HITHRPT **240**, 293
T_IDLE **32**, 36-37, 63, 287
T_IMMEDIATE **239**, 293
T_INCON **32**, 37, 287
T_INETCONTROL **239**, 293
T_INET_IP **239**, 292
T_INET_TCP 238, 291
T_INET_UDP 238, 292
T_INFINITE **287**
T_INFO **286**
T_INREL **32**, 37, 287
T_INVALID **287**
T_IP-level options 232
T_IP_BROADCAST 232, 239, 293
T_IP_DONTROUTE 232, 239, 293
T_IP_OPTIONS 232, 239, 292
T_IP_REUSEADDR 233, 239, 293
T_IP_TOS 233, 239, 292
T_IP_TTL 234, 239, 292
T_ISO_TP **226**, 289
t_kpalive **238**, 292
T_LDELAY **240**, 293
T_LISTEN 15
t_listen 15
T_LISTEN 16
t_listen 28
T_LISTEN 38
t_listen 38, 77, 223, 229, 236
T_LISTEN 247, 282
t_listen 283
t_listen( ) 44, 77
T_LOCOST **240**, 293
t_look 15, 28, 79, 236, 283
t_look( ) **79**
T_LTPDUDFLT **225**, 289
T_MORE 23, 91, 105, 108, 282
T_MORE flag 229
T_NB_ABORT 275
T_NB_BCAST_NAME 269
T_NB_CLOSED 275
T_NB_GROUP 269
T_NB_LOCAL 269

T_NB_NAMELEN....269
T_NB_NOANSWER....275
T_NB_OPREJ....275
T_NB_UNIQUE....269
T_NEGOTIATE....**282**
T_NETCONTROL....**239**, 293
T_NEXTHDR....**287**
T_NO....**287**
T_NOPROTECT....**225**, 288
T_NOTOS....**240**, 293
T_NOTSUPPORT....**282**
T_NULL....**287**
t_open....11, 28, 81, 223, 236, 283
t_open( )....**81**
T_OPT....**287**
t_optmgmt....28, 84, 283
T_OPTMGMT....**286**
t_optmgmt( )....45, 84
T_OPT_DATA....**287**
T_OPT_FIRSTHDR....**287**
T_ORDREL....15, 17, 38, 247, 282
T_ORDRELDATA....**285**
T_OUTCON....**32**, 37, 287
T_OUTREL....**32**, 37, 287
T_OVERRIDEFLASH....**239**, 293
T_PARTSUCCESS....**282**
T_PASSIVEPROTECT....**225**, 289
T_PRIDFLT....**225**, 288
T_PRIHIGH....**225**, 288
T_PRILOW....**225**, 288
T_PRIMID....**225**, 288
T_PRIORITY....**239**, 293
T_PRITOP....**225**, 288
T_PUSH....109, 119, 282
t_rcv....15, 28, 38, 91, 224, 236, 283
t_rcv( )....**91**
t_rcvconnect....28, 38, 93, 224, 236
t_rcvconnect( )....44, 93, 283
t_rcvdis....15, 28, 95, 224, 236, 283
t_rcvdis( )....**95**
t_rcvleafchange( )....**389**
t_rcvrel....15, 38, 97, 284
t_rcvrel( )....**97**
t_rcvreldata....**98**, 284
t_rcvreldata( )....**98**
t_rcvudata....28, 38, 100, 224, 284
t_rcvudata( )....44, 100
t_rcvuderr....15, 28, 102, 224, 284
t_rcvuderr( )....45, 102
t_rcvv....28, 284
t_rcvv( )....**104**
t_rcvvudata....28
t_rcvvudata( )....**106**
T_READONLY....**282**
t_removeleaf( )....**387**
T_ROUTINE....**239**, 293
T_SENDZERO....**285**
T_SNA_CONNECTION_OUTAGE....325
T_SNA_CONNECTION_SETUP_FAILURE....325
T_SNA_MAX_LU_LEN....321
T_SNA_MAX_NETID_LEN....321
T_SNA_MAX_TPN_LEN....321
T_SNA_SYSTEM_DISCONNECT....325
T_SNA_TIMEOUT....325
T_SNA_USER_DISCONNECT....325
t_snd....15, 28, 108, 224, 236-237, 284
t_snd( )....**108**
t_snddis....28, 111, 224, 237, 284
t_snddis( )....**111**
t_sndrel....**113**, 284
t_sndrel( )....**113**
t_sndreldata....**114**, 284
t_sndreldata( )....**114**
t_sndudata....28, 116, 224, 237, 284
t_sndudata( )....**116**
t_sndv....28, 284
t_sndv( )....**118**
t_sndvudata....28, 284
t_sndvudata( )....**121**
t_strerror....28, 124, 284
t_strerror( )....**124**
T_SUCCESS....**282**
t_sync....12, 28, 125, 284
t_sync( )....**125**
t_sysconf....28, 284
t_sysconf( )....**127**
T_TCL_CHECKSUM....**227**, 291
T_TCL_PRIORITY....**227**, 290
T_TCL_PROTECTION....**227**, 290
T_TCL_RESERRORRATE....**227**, 290
T_TCL_TRANSDEL....**227**, 290
T_TCO_ACKTIME....**227**, 290
T_TCO_ALTCLASS1....**227**, 291
T_TCO_ALTCLASS2....**227**, 291
T_TCO_ALTCLASS3....**227**, 291
T_TCO_ALTCLASS4....**227**, 291
T_TCO_CHECKSUM....**227**, 291
T_TCO_CLASS....**227**, 291
T_TCO_CONNRESIL....**227**, 290
T_TCO_ESTDELAY....**226**, 290
T_TCO_ESTFAILPROB....**226**, 290
T_TCO_EXPD....**227**, 290

T_TCO_EXTFORM **227**, 291
T_TCO_FLOWCTRL **227**, 291
T_TCO_LTPDU **227**, 290
T_TCO_NETEXP **227**, 291
T_TCO_NETRECPTCF **227**, 291
T_TCO_PRIORITY **227**, 290
T_TCO_PROTECTION **227**, 290
T_TCO_REASTIME **227**, 290
T_TCO_RELDELAY **227**, 290
T_TCO_RELFAILPROB **226**, 290
T_TCO_RESERRORRATE **226**, 290
T_TCO_THROUGHPUT **226**, 290
T_TCO_TRANSDEL **226**, 290
T_TCO_TRANSFFAILPROB **226**, 290
T_TCP 229
T_TCP_KEEPALIVE 230, 238, 291
T_TCP_MAXSEG 231, 238, 291
T_TCP_NODELAY 231, 238, 291
T_UDATA **287**
T_UDERR 15, 17, 26, 38, 247, 282
T_UDERROR **286**
T_UDP 229
T_UDP-level options 231
T_UDP_CHECKSUM 231, 239, 292
t_unbind 28
T_UNBIND **32**
t_unbind **128**, 284
t_unbind( ) **128**
T_UNBND 36-37, 287
T_UNIT **32**
T_UNITDATA **286**
T_UNSPEC 48, 287
T_YES **287**
UDP 295, 353
UDP-level options 239, 292
unbind 28, 33, 36, 38
undefined 1
unitdata 100, 116
Unitdata error structure 286
UNIX
  process 12
  versions 279
unspecified 1
user application 19, 26, 417
user data 95, 114
User Datagram Protocol 353
User-Network Interface 395
will 2
write( ) **176**
writev( ) **176**
X/Open name space **3**
XEM 295
XTI 9, 295
  applications 51
  features 28-29
  library 51
XTI error return 282
XTI level 288
XTI-level options 88, 288
XTI_DEBUG **288**
XTI_GENERIC **288**
XTI_LINGER **288**
XTI_RCVBUF **288**
XTI_RCVLOWAT **288**
XTI_SNDBUF **288**
XTI_SNTLOWAT **288**
Zero-length TSDUs and TSDU fragments 72
..... 82, 108-109, 215, 224