INDEX

A

Abort a Print Job 101,102
Aborting Commands 76
Access Functions 251
Add New Keyboards to the List 310
add-form *(Function)* 299
ADD.BORDER.TO.BITMAP *(Function)* 65
ADD.PROCESS *(Function)* 266
Adding FileBrowser Commands 92
Address space of the internetwork 261
ADJUSTCOLORMAP *(Function)* 37
Administrator Commands for NSMaintain 206
ADVISE *(Function)* 283
Aliases 203
ALLOCATE.ETHERPACKET *(Function)* 270
Analyzing user Functions 157
Application layer 261
ARPANET 253
Array Functions 193
Array operations 29
Assignments 189

B

BACKWARD FIND *(Command)* 288
BEGINDST *(Variable)* 199
BIN *(Function)* 222
BIT.IN.COLUMN *(Function)* 65
BIT.IN.ROW *(Function)* 65
BITBLT *(Function)* 116
BITMAPCREATE WIDTH HEIGHT BITSPERPIXEL *(Function)* 38
BITSPERPIXEL BITMAP *(Function)* 38
BOUT *(Function)* 222
BREAK *(Function)* 283
Broadcast address 262
Browser 111
    describe window 12
    printout window 12
BROWSER window 74
BROWSERBOXING *(Variable)* 12
BROWSERFORMAT *(Variable)* 12

C

C-Shell 303
C4 printed circuit board 1
CACHE/NODE/LABEL/BITMAPS *(Variable)* 116
Call graph 233
Call stack 233
Calling the Keyboard Editor 150
CALLS *(Function)* 175
CALLSCCODE *(Function)* 175
Cash-File 15
Centronics 17
    cable 17
    port 1
CENTRONICS.RESET *(Function)* 17
CH.DEFAULT.DOMAIN *(Variable)* 206
CH.DEFAULT.ORGANIZATION *(Variable)* 206
Changing an Existing Keyboard 150
Character framing errors 224
CharCodeTables 19
Chat 21,303
CHAT *(Function)* 22,303
Chat connections 22
Chat Menu 23
Chat window 146
CHAT.ALLHOSTS *(Variable)* 25
CHAT.DISPLAYTYPES *(Variable)* 24
CHAT.EMACSCOMMANDS *(Variable)* 25
CHAT.FONT *(Variable)* 25
CHAT.IN.EMACS? *(Variable)* 26
CHAT.INTERRUPTS *(Variable)* 24
CHAT.KEYACTIONS *(Variable)* 24
CHAT.PROTOCOLTYPES *(Variable)* 26
CHAT.TTY.PROCESS *(Variable)* 25
CHAT.WINDOW.REGION *(Variable)* 25
CHAT.WINDOW.SIZE *(Variable)* 25
Chatting to a host 258
CHECKSAMEDIR *(Function)* 231
Clean-Up After Copying Files 46
Clearinghouse 26
Clearinghouse service 203
CLOSE *(Function)* 305
close-hash-file *(Function)* 134
CLOSECHATWINDOWFLG *(Variable)* 25
CLOSEF *(Function)* 9,222
CLOSEHASHFILE *(Function)* 128
CmlFloatArray 29
CMYCOLORMAP *(Function)* 36
COLLECTINUSE *(Function)* 108
COLLISIONS *(Function)* 108
COLORDISPLAY *(Function)* 33
COLORDISPLAYP *(Function)* 33
COLORIZEBITMAP *(Function)* 44
COLORLEVEL *(Function)* 37
COLORMAPCOPY *(Function)* 36
COLORMAPCREATE *(Function)* 36
COLORNAMES *(Association List)* 34
COLORNUMBERP *(Function)* 35
COLORSCREEN *(Function)* 39
COLORSCREENBITMAP *(Function)* 38
COMMAND menu 74
Common Lisp 288
COMPILE *(Command)* 86
Conjunctions of Sets 168
Connecting to a host 258
Control-E *(Command)* 76
Control-F *(Command)* 53
Control-O *(Command)* 139
Control-P *(Command)* 280
Control-Q *(Command)* 75,87
Control-S *(Command)* 53
Control-Z *(Command)* 53
Conversation Mode 45
convert-loaded-files *(Variable)* 292
COPY *(Command)* 78
Copy-Selection 53, 77
COPYFILE *(Function)* 46,267
CopyFiles 45
COPYFILES *(Function)* 45
COPYHASHFILE *(Function)* 130
CPE FP 17
CPE-FP upgrade 1
Create a Key Pad for Repeated Use 216
CREATE-PROCESS-STREAM *(Function)* 305
CREATE.NUMBERPAD.READER *(Function)* 216
CREATEHASHFILE *(Function)* 128

CREATEW *(Function)* 41

Creating
    4045XLP Master Files 4
    a graph 112
    a key pad 215
    a new bitmap 64
    a new keyboard 149
    a TableBrowser 247
    horizontal rules 139
    HOST.TXT file 254
    new keyboard configurations 153
    objects 206
    the local IP.INIT file 256
Creation dates 46
Cumulative mode 238
CURSORPOSITION *(Function)* 44
CURSORSCREEN *(Function)* 44
Customizing
    Chat 24
    FileBrowser 89
Cyclic graphs 111

D

DataBaseFns 49
Databases 313
    using 313,314
Datamedia 2500 21
Daughter 114
Debuggee 279
Debugger 279
DECREASING.FONT. LIST *(Variable)* 125
DEdit 51,307
    Command menu 54
    Functions 54
    idioms 59
    parameters 57
    window 52
DEDITLINGER *(Variable)* 57
DEDITTYPEINCOMS *(Variable)* 57
DEFAULT-CLEANUP-COMPILER *(Variable)* 86
DEFAULT.GRAPH.WINDOWSIZE *(Variable)* 117
DEFAULTCHATHOST *(Variable)* 22,25
DEFAULTPRINTERTYPE *(Variable)* 2
DEFAULTPRINTINGHOST *(Variable)* 2,213
defdefiner-macros *(Variable)* 292
DEFINEKEYBOARD *(Function)* 311
Defining a Virtual Keyboard 310
DEGREES-TO-RADIANS *(Function)* 196
DELETE *(Command)* 77
Deleting objects 206
Device Errors 17
Differences between TExec and TEdit 287
Digital VT100 21
Directed acyclic graph 113
DIRECTORIES *(Variable)* 2,258
Directory-only lines 75
DIRECTORYNAME *(Function)* 231
Display-Only a Keyboard 309
Displayer 233
DISPLAYFONTDIRECTORIES *(Variable)* 100
DISPLAYGRAPH *(Function)* 117
Displaying
    a graph 116
    a stack 280
Displaystream graphics 288
DM2500 Chat 27
DODIP.HOSTP *(Function)* 272

Domain 203
manipulating 208
DOMAIN.GRAPH *(Function)* 276
DOMAIN.INIT *(Function)* 276
DOMAIN.LOOKUP *(Function)* 276
DOMAIN.LOOKUP ADDRESS *(Function)* 276
DOMAIN.LOOKUP NAMESERVER *(Function)* 276
DOMAIN.TRACE *(Function)* 276
DRAWBETWEEN *(Function)* 43
DRAWLINE *(Function)* 43
DRAWTO *(Function)* 43
DSPBACKCOLOR *(Function)* 42
DSPCOLOR *(Function)* 42
DT.EDITMACROS *(Variable)* 58
DUMPDATABASE *(Function)* 177
DUMPDB *(Function)* 50
DUMPGRAPH *(Function)* 121
Dvorak keyboard 311
Dynamic structure 233

E

E-30 option kit 1
E30 option 219
EDIT *(Command)* 84
Edit buffer window 52
EDIT.BITMAP *(Function)* 61
EditBitMap 61
sub-menu 62
window 62
EDITBM *(Function)* 61
EDITCOLORMAP *(Function)* 37
EDITCONFIGURATION *(Function)* 154
EDITEMBEDTOKEN *(Variable)* 57
EDITGRAPH *(Function)* 121
EDITGRAPHMENU 119
EDITGRAPHMENUCOMMANDS *(Variable)* 119
Editing
a bitmap in a document 64
editing a graph 117
existing bitmap 63
keyboard configuration 154
EDITKEYBOARD *(Function)* 150
EDITMODE *(Function)* 52
EDITMODE *(Variable)* 84
Effecting MasterScope Analysis 171
Element patterns 185
Embedding and extracting 59
ENDDST *(Variable)* 199
Entering DEdit Commands 54
Environmental and Lisp mappings 95
EOLCONVENTION *(Variable)* 272
Error Condition Reporting 224
ERRORPUP *(record)* 70
ERRORXIP *(record)* 69
Establishing a Connection 143
et FX-80 Fast Mode 101
ETHERHOSTNUMBER *(Variable)* 4,280
Ethernet 260
Ethernet packet 264
EtherRecords 69
EVALUATE-POLYNOMIAL *(Function)* 31
expr definitions of Functions 171
EXPUNGE *(Command)* 86
Extend the selection 75
Extended Processor board 17
Extended processor option 29
External Communication Service 27

F

FASSOC *(Function)* 121
Fast mode FX-80 101
FASTFX80-DEFAULT-DESTINATION *(Variable)* 100
FB *(Command)* 90
FB.ALLOW.ABORT *(Function)* 94
FB.BROWSERFONT *(Variable)* 92
FB.DEFAULT.EDITOR *(Variable)* 84,92
FB.DEFAULT.INFO *(Variable)* 87,90
FB.DEFAULT.NAME.WIDTH *(Variable)* 91
FB.FETCHFILENAME *(Function)* 93
FB.HARDCOPY.DIRECTORY.FONT *(Variable)* 92
FB.HARDCOPY.FONT *(Variable)* 92
FB.ICONFONT *(Variable)* 91
FB.INFO.FIELDS *(Variable)* 91
FB.INFO.MENU.ITEMS *(Variable)* 91
FB.MENU.ITEMS *(Variable)* 92
FB.PROMPTFONT *(Variable)* 92
FB.PROMPTLINES *(Variable)* 92
FB.PROMPTW.FORMAT *(Function)* 93
FB.PROMPTWPRINT *(Function)* 93
FB.PROMTFORINPUT *(Function)* 94
FB.SELECTEDFILES *(Function)* 93
FB.TABLEBROWSER *(Function)* 93
File Manager functions, changed 313
File name translation Function 269
File transfer using RS232 226
FILEBROWSE *(Command)* 84
FileBrowser 71
Functions 89
Variables 90
FILEBROWSER *(Function)* 89
filecom-specifier *(Function)* 299
Filing operations 267
FILING.ENUMERATION.DEPTH *(Variable)* 92
FIND-ARRAY-ELEMENT-INDEX *(Function)* 31
FINDPOINTER *(Function)* 109
FINDPOINTERS.OF.TYPE *(Function)* 109
FLIPNODE *(Function)* 120
Floating-point vector 29
FNT.DISPLOOK *(Function)* 96
FNT.DISPTBLE *(Function)* 96
FNT.MAKEBOOK *(Function)* 95
Font mappings 95
FontSample 95
FORCEOUTPUT *(Function)* 222
Forest 113
form-specifier *(Function)* 299
FORWARD FIND *(Command)* 288
FREEVARS *(Function)* 175
FROM.SCREEN.BITMAP *(Function)* 66
FTP service 268
FTPDEBUG *(Function)* 268
FTPserver 97
FTPSERVER *(Function)* 4,97
FTPSERVER.DEFAULT.HOST *(Variable)* 97
Function-calling structures 111
Functions for saving work 283
Functions for writing routines 176
FX Printer Compatibility 102
FX-80
DIP Switch Settings 100
driver 99
family 99
serial Interface 99
FX80-PRINT *(Function)* 102

G

GapTelnet 26
Garbage collector 284
Gateway Access Protocol 26
GCHax 105
General Purpose Records 69
Get, Set Parameters via Inspector Window 7
get-cash-file *(Function)* 15
GETBOXSCREENPOSITION *(Function)* 40
GETBOXSCREENREGION *(Function)* 41
GETFILEINFO *(Function)* 267
GETHASHTEXT *(Function)* 129
GETHASHVILE *(Function)* 129
GETRELATION *(Function)* 176
GETSCREENPOSITION *(Function)* 40
GETTEMPLATE *(Function)* 174
GETTERMTABLE *(Function)* 288
Getting Hardcopy Directory Listings 77
Ghost boxes 238
Graph 11
    data structure 120
    nodes 111
GRAPH *(record)* 121
GRAPH/HARDCOPY/FORMAT *(Variable)* 117
Grapher 111,233
Grapher image objects 120
GRAPHEROBJ *(Function)* 120
GRAPHERPROP *(Function)* 121
Graphics interface 99
GRAPHNODE *(record)* 122
GRAPHREGION *(Function)* 120
GraphZoom 125
GRAYCOLORMAP *(Function)* 36
Groups 207

H

HANZON Universal Card 99
HARDCOPY *(Command)* 81
HARDCOPYGRAPH *(Function)* 117
Hash tables 178
Hash-File 133
HASHFILEDEFAULTSIZE *(Variable)* 130
HASHFILEDTBL *(Variable)* 130
HASHFILENAME *(Function)* 129
HASHFILEP *(Function)* 129
HASHFILEPLST *(Function)* 130
HASHFILEPROP *(Function)* 129
HASHLOADFACTOR *(Variable)* 130
HFGROWTHFACTOR *(Variable)* 131
HLS *(Record)* 35
HLSP *(Function)* 35
HOST&DIRECTORYFIELD *(Function)* 231
HOSTS.TEXT.DIRECTORIES *(Variable)* 276
HOSTS.TXT files, parsing 276
HPRINT *(Function)* 121
HQFX80-DEFAULT-DESTINATION *(Variable)* 100
HQFX80-FONT-DIRECTORIES *(Variable)* 100
HRule 139
HRULE.CREATE *(Function)* 139
HTE.READ.FILE *(Function)* 276
Hue–Lightness–Saturtion Triples 35

I

I/O processor 226
IDENTITY-3-BY-3 *(Function)* 194
IDENTITY-4-BY-4 *(Function)* 194
il:typesof *(Function)* 313
il:whereis *(Function)* 313
Imagestream 5
Include all files, both deleted and undeleted 75
Individual moe 239

INFO menu 74
Info Options window 87
Input conventions for FileBrowser commands 75
Inserting a segment 59
install-form *(Function)* 299
INTENSITIESFROM *(Function)* 36
INTERACT&ADD.BORDER.TO.BITMAP *(Function)* 66
INTERACT&SHIFT.BITMAP.DOWN *(Function)* 66
INTERACT&SHIFT.BITMAP.LEFT *(Function)* 66
INTERACT&SHIFT.BITMAP.RIGHT *(Function)* 66
Interactive File Transfers With Kermit or Modem 146
Interactive Terminal Service 27
Internal fields 241
Internet layer 260
INTERPRESSFONTDIRECTORIES *(Variable)* 19
Interpreter 239
Interrupt character 234
INVERT.BITMAP.B/W *(Function)* 65
INVERT.BITMAP.DIAGONALLY *(Function)* 65
INVERT.BITMAP.HORIZONTALLY *(Function)* 65
INVERT.BITMAP.VERTICALLY *(Function)* 65
Invisible characters 288
IP
    addresses 262
    networks primer 261
    packet building 274
    packet sending 275
    socket access 272
IP.ADD.PROTOCOL *(Function)* 273
IP.APPEND.BYTE *(Function)* 275
IP.APPEND.CELL *(Function)* 275
IP.APPEND.STRING *(Function)* 275
IP.APPEND.WORD *(Function)* 275
IP.CLOSE.SOCKET *(Function)* 274
IP.DELETE.PROTOCOL *(Function)* 274
IP.OPEN.SOCKET *(Function)* 274
IP.SETUPIP *(Function)* 275
IP.TRANSMIT *(Function)* 275
IPHOSTNAME *(Function)* 273
IPINIT *(Function)* 272
IPTRACE *(Function)* 273
Iterative statement operator 177

J

join-comments *(Variable)* 292

K

Kermit 144,226
Kermit menu 146
KERMIT.RECEIVE *(Function)* 145
KERMIT.SEND *(Function)* 144
KEYACTION *(Function)* 153
Keyboard editor menus 150

KeyboardEditor 149

L

Landscape mode 1
Landscape printing 103
Lattices 111
Laying out a graph for display 112
LAYOUTGRAPH *(Function)* 112
LAYOUTSEXPR *(Function)* 115
LDESHELL *(Variable)* 303
Library module changes 5-9
    summary 5-9
Library module dependencies 9
Linguistic tree 120
Link layer 260
Lisp interrupts 22
LISPUSERSDIRECTORIES *(Variable)* 258
LISPXREAD *(Function)* 53
List of nodes 112
List structure editor 51
LISTFILES *(Function)* 3,213
LOAD *(Command)* 84
LOAD *(Function)* 49
Load New Keyboards 310
load-textmodule *(Function)* 290
LOADDB *(Function)* 50
LOADDBFLG *(Variable)* 50
LOADFROM *(Function)* 49,85
Loading TCP 258
LOOKUPHASHFILE *(Function)* 129
Lost characters 224
Low-level TCP Functions 272

M

Macro 177
Macro Expansion 170
MAINSCREEN *(Function)* 39
Maintenance panel halt 279
make-cash-file *(Function)* 15
make-hash-file *(Function)* 133
MAKE-HOMOGENEOUS-3-BY-3 *(Function)* 193
MAKE-HOMOGENEOUS-3-VECTOR *(Function)* 193
MAKE-HOMOGENEOUS-4-VECTOR *(Function)* 194
MAKE-HOMOGENEOUS-N-BY-3 *(Function)* 193
MAKE-HOMOGENEOUS-N-BY-4 *(Function)* 194
make-specifier *(Function)* 297
make-textmodule *(Function)* 291
MAKEFILE *(Function)* 49,78,231
MAKEFILEFORMS *(Function)* 231
Making a sysout that contains TCP-IP 259
Manipulating
    domains 208
    groups 207
    remote Vmem 284
MAP-ARRAY *(Function)* 29
MAPGC *(Function)* 110
MAPHASHFILE *(Function)* 130
MAPOFACOLOR *(Function)* 37
MAPRELATION *(Function)* 176
MASTERSCOPE *(Function)* 175
MasterScope 11,49,157
    Commands 158
    database 157
    entries 175
    relations 161
    set specifications 164
    templates 164
Match 183
MatMult 193
MATMULT-133 *(Function)* 195
MATMULT-144 *(Function)* 195
MATMULT-331 *(Function)* 195
MATMULT-333 *(Function)* 195
MATMULT-441 *(Function)* 195
MATMULT-444 *(Function)* 196
MATMULT-N33 *(Function)* 195
MATMULT-N44 *(Function)* 196
Matrix Creation Functions 193
Matrix multiplication 193
Matrix Multiplication Functions 195
Merged node 237
MIGRATIONS *(Variable)* 231
MiniServe 199
mlFloatArray 196
Modem 145,226
MODEM.RECEIVE *(Function)* 145
MODEM.SEND *(Function)* 145
Moving an expression 59
MSMACROPROPS *(Variable)* 170
MSMARKCHANGED *(Function)* 177
MSNEEDUNSAVE *(Function)* 176
Multiple DEdit commands 58
Multiple streams 9
MY.NSHOSTNUMBER *(Variable)* 200

N

Network addresses 261
Network protocols 21,26
NETWORKOSTYPES *(Variable)* 257
NEW INFO FileBrowser Command 87
NODECREATE *(Function)* 112
Notecards 9
Noticing changes that require recompiling 177
NS characters 213
NS Chat 26
NS Records 69
NS.TO.PUP.ALIST *(Variable)* 200
NS.TO.PUP.FILE *(Variable)* 200
NSHOSTNUMBER *(record)* 69
NSMaintain 203
NSMAINTAIN *(Function)* 204
NSTIMESERVER *(Function)* 199
NTERACT&SHIFT.BITMAP.UP *(Function)* 66
NUMBERPAD.READ *(Function)* 216

O

Objects 203
    creating 206
    deleting 206
Obtaining information in NSMaintain 204
Obtaining network addresses 254
open-hash-file *(Function)* 133
OPENHASHFILE *(Function)* 128
OPENIMAGESTREAM *(Function)* 9,101
Opening
    a 4045 Stream 7
    a Centronics Stream 17
    a Chat Connection 22
OPENP *(Function)* 222
OPENSTREAM *(Function)* 222

OPENWINDOWS *(Function)* 42
Operations on Multiple Items 249
Organization 203
OVERFLOWS *(Function)* 108

P

Packages 124, 286
Parents 114
parity errors 224
PARSERELATION *(Function)* 176
PATLISPCHECK *(Variable)* 184
Pattern
    elements 185
    match compiler 184
    match expressions 185
PATVARDEFAULT *(Variable)* 185
PERSPECTIVE-4-BY-4 *(Function)* 195
Place Markers 189
Portrait mode 1
PORTSTRING *(Function)* 4,280
PPTCB *(Function)* 266
Press 213
PRESSFONTWIDTHFILES *(Variable)* 213
Primitive relationship 178
PRIMTERMTABLE *(Variable)* 288
PRIN3 *(Function)* 124
Print
    a file 101,102
    source code in high-quality mode 103
    source code on fast FX-80 103
    TEdit file in fast FX-80 mode 103
    TEdit file in high-quality mode 104
    text and graphics in high-quality mode 103
print-filecom *(Function)* 299
Printer drivers 99
Printer's point 139
Printing
    in fast mode 101
    in high-quality mode 102
    source or TEdit Files 3
    speed 9
    via FTPserver 4
    windows 4
Programmer's Assistant 288
PROJECT-AND-FIX-3-VECTOR *(Function)* 196
PROJECT-AND-FIX-4-VECTOR *(Function)* 196
PROJECT-AND-FIX-N-BY-3 *(Function)* 196
PROJECT-AND-FIX-N-BY-4 *(Function)* 196
PROMPT window 73
Properties 203
Protocol number 273
PUP *(record)* 70
PUP
    Chat 26
    FTP 97,232
    ID service 200
    records 70
    time service 200
PUP.ID.SERVER *(Function)* 199
PUPADDRESS *(record)* 70
PUPNUMBER *(Variable)* 200
PUPTIMESERVER *(Function)* 199
PUTHASHFILE *(Function)* 128
PUTHASHTEXT *(Function)* 129

Q

QABLEITEM *(record)* 69
Query Mode 45
Queue multiple arguments 60
Quick-scrolling 74
Quitting the FileBrowser 76

R

RANDACCESSP *(Function)* 272
READ-BYTE *(Function)* 219
READ-CHAR *(Function)* 219
READGRAPH *(Function)* 121
Reading the Remote Vmem 284
ReadNumber 215
ReadNumber window 66
READP *(Function)* 222
READSYS *(Function)* 284
REALFRAMEP *(Function)* 239
RECOMPILEDEFAULT *(Variable)* 178
RECOMPUTE *(Command)* 86
Reconstruction 190
Record declaration 177
Recursive loads 97
Redefine Existing Keyboards 310
REDUCE-ARRAY *(Function)* 30
Red–Green–Blue Triples 34
REFCNT *(Function)* 108
Reference counting 284
REHASHFILE *(Function)* 130
REHSHGAG *(Variable)* 131

Relations between sets 157
RELDRAWTO *(Function)* 43
Remote
    Kermit in Server Mode 144
    Kermit not in Server Mode 144
    system administration 27
    system executive 27
Removing Keyboards From the Men 310

RENAME *(Command)* 81
REPACKFILENAME.NEW.TRANSLATION *(Function)* 269
REPACKFILENAME.OSTYPE.TABLE *(Variable)* 269
REPACKFILENAME.STRING *(Function)* 268
Replace All Known Keyboards 310
Replacements 190
RESET/NODE/BORDER *(Function)* 120
RESET/NODE/LABELSHADE *(Function)* 120
RESETDEDIT *(Function)* 54
Resetting 4045XLPstream 8
Restarting MiniServe 201
RGB *(Record)* 34
RGBCOLORMAP *(Function)* 36
RGBP *(Function)* 35
RNUMBER *(Function)* 215
Ross referencing user programs 157
ROTATE-3-BY-3 *(Function)* 194
ROTATE-4-BY-4-ABOUT-X *(Function)* 194
ROTATE-4-BY-4-ABOUT-Y *(Function)* 194
ROTATE-4-BY-4-ABOUT-Z *(Function)* 194
ROTATE.BITMAP.LEFT *(Function)* 65
ROTATE.BITMAP.RIGHT *(Function)* 65
ROTATECOLORMAP *(Function)* 37
RS232 143,219

RS232
    Chat 27
    port 219
RS232C port 1
RS232C.CLOSE-STREAM *(Function)* 222
RS232C.DEFAULT.INIT.INFO *(Variable)* 220
RS232C.ERROR.STREAM *(Variable)* 224
RS232C.GET.PARAMETERS *(Function)* 221
RS232C.INIT *(Function)* 220
RS232C.OTHER.STREAM *(Function)* 222
RS232C.OUTPUT.PACKET.LENGTH *(Function)* 222
RS232C.READP.EVENT *(Function)* 223
RS232C.REPORT.STATUS *(Function)* 224
RS232C.SET.PARAMETERS *(Function)* 220
RS232C.SHUTDOWN *(Function)* 221
RS232CHAT 225
RS232CMENU 225,228
RS232MODEMCONTROL *(Function)* 223
RS232MODEMHANGUP *(Function)* 224
RS232MODEMSTATUSP *(Function)* 223
RS232MODIFYMODEMCONTROL *(Function)* 223
RS232SENDBREAK *(Function)* 223
RS232TRACE 224

S

S-expression 115
SameDir 231
Sampler 233
SAVEDBFLG *(Variable)* 50
Scale factors 9
SCALE-3-BY-3 *(Function)* 194
SCALE-4-BY-4 *(Function)* 195
SCALEDBITBLT *(Function)* 9
SCREENBITMAP *(Function)* 39
SCREENCOLORMAP *(Function)* 36
SCREENPOSITION *(Record)* 39
SCREENREGION *(Record)* 39
Scroll bar 74
SEE *(Command)* 82
Segment Patterns 186
Segment selection and manipulation 59
Selecting Files 74
Selecting Objects and Lists 52
Selection stack 53
Send text output to fast FX-80 103
SEND.FILE.TO.PRINTER *(Function)* 3,213
Serial interface card 99
Server mode 144
SET DEPTH *(Command)* 88
Set
    destination 102
    determiners 167
    FX-80 Destination 101
    FX-80 Page Size 101
    HQ Mode 102
    page Size 102
    specifications by Blocktypes 167
    specifications by Relation 166
    types 167
SETCOLORINTENSITY *(Function)* 37
SETSYNONYM *(Function)* 175
SETTEMPLATE *(Function)* 174
SETTIME *(Function)* 199
SHIFT.BITMAP.DOWN *(Function)* 65
SHIFT.BITMAP.LEFT *(Function)* 66
SHIFT.BITMAP.RIGHT *(Function)* 66
SHIFT.BITMAP.UP *(Function)* 66
SHOW PATHS 180
SHOW PATHS *(Command)* 169
SHOW PATHS *(Function)* 11
SHOW.CLOSED.WINDOWS *(Function)* 110
SHOWCIRCULARITY *(Function)* 110
SHOWCOLORTESTPATTERN *(Function)* 44
SHOWCOMMONCSETS *(Function)* 20
SHOWCSET *(Function)* 20
SHOWCSETLIST *(Function)* 20
SHOWCSETRANGE *(Function)* 20
SHOWGC *(Function)* 107
SHOWGRAPH *(Function)* 116,125
SHOWZOOMGRAPH *(Function)* 125
Simple item operations 248
SINGLEFILEINDEX *(Function)* 10
Sketch 10
SORT *(Command)* 89
Special characters 307
specifiers *(Variable)* 297
Specifying the files to browse 72
Spy 233
SPY.BORDERS *(Variable)* 239
SPY.BUTTON *(Function)* 234
SPY.END *(Function)* 234
SPY.FONT *(Variable)* 239
SPY.FREQUENCY *(Variable)* 239
SPY.LEGEND *(Function)* 239
SPY.MAXLINES *(Variable)* 239
SPY.NOMERGEFNS *(Variable)* 239
SPY.START *(Function)* 234
SPY.TOGGLE *(Function)* 234
SPY.TREE *(Function)* 235
SPY.TREE *(Variable)* 239
Stack 280
Stack architecture 58
Stacking several rules in a single object 140
Starting FileBrowser 71
Starting TExec 287
STARTMINISERVE *(Function)* 199
Static structure 233
stderr 305
stdin 305
stdout 305
STOPIP *(Function)* 272
STORAGE *(Function)* 105
Store keyboards 309
SUBNETMASK *(Variable)* 262
Subnetworks 262
Sun,TCP to a 254
SunOS, interface to 305
Switch & Display a Keyboard 308
Switch A-2 2
Switch Keyboards 308
Switch settings 2
SWITCHKEYBOARDS *(Function)* 311
SysEdit 241
SYSHASHFILE *(Variable)* 131
SYSHASHFILELST *(Variable)* 131
System sources 241

T

TableBrowser 243
TABLEBROWSER *(record)* 244
TABLEITEM *(record)* 243
Tally window 73
Target-source-Command 59

TB.BROWSER.BUSY *(Function)* 248
TB.CLEAR.LINE *(Function)* 249
TB.COLLECT.ITEMS *(Function)* 250
TB.DELETE.ITEM *(Function)* 248
TB.FIND.ITEM *(Function)* 250
TB.FINISH.CLOSE *(Function)* 248
TB.INSERT.ITEM *(Function)* 249
TB.ITEM.DELETED? *(Function)* 251
TB.ITEM.SELECTED? *(Function)* 251
TB.MAKE.BROWSER *(Function)* 247
TB.MAP.DELETED.ITEMS *(Function)* 250
TB.MAP.ITEMS *(Function)* 250
TB.MAP.SELECTED.ITEMS *(Function)* 250
TB.NORMALIZE.ITEM *(Function)* 249
TB.NTH.ITEM *(Function)* 250
TB.NUMBER.OF.ITEMS *(Function)* 249
TB.REDISPLAY.ITEMS *(Function)* 249
TB.REMOVE.ITEM *(Function)* 249
TB.REPLACE.ITEMS *(Function)* 248
TB.SELECT.ITEM *(Function)* 248
TB.SET.FONT *(Function)* 248
TB.UNDELETE.ITEM *(Function)* 248
TB.USERDATA *(Function)* 251
TB.WINDOW *(Function)* 251
TCP
    and directory enumeration 254
    Chat 27
    debugging aids 277
    segment 264
TCP-IP 253
    protocol 47
    protocol layers 260
TCP.BYE *(Function)* 268
TCP.CLOSE.SENDER *(Function)* 265
TCP.DEFAULT.RECEIVE.WINDOW *(Variable)* 265
TCP.DEFAULT.USER.TIMEOUT *(Variable)* 265
TCP.DEFAULTFILETYPE *(Variable)* 267
TCP.ECHO.SERVER *(Function)* 266
TCP.ECHOTEST *(Function)* 266
TCP.FAUCET *(Function)* 267
TCP.INIT *(Function)* 265
TCP.OPEN *(Function)* 264
TCP.OTHER.STREAM *(Function)* 265
TCP.SINK.SERVER *(Function)* 266
TCP.STOP *(Function)* 265
TCP.URGENT.EVENT *(Function)* 265
TCP.URGENT.MARK *(Function)* 265
TCP.URGENTP *(Function)* 265
TCPCHAT.TELNET.TTY.TYPES *(Variable)* 269
TCPCHAT.TRACEFILE *(Variable)* 269
TCPCHAT.TRACEFLG *(Variable)* 269
TCPFTP.DEFAULT.FILETYPES *(Variable)* 267
TCPFTP.EOL.CONVENTION 267
TCPFTP.INIT *(Function)* 268
TCPFTP.SERVER *(Function)* 268
TCPFTP.SERVER.USE.TOPS20.SYNTAX *(Variable)* 268
TCPTRACE *(Function)* 266
TEdit 9,21,139,287,307
TEdit Chat 28
TeleRaid 279
TELERAID *(Function)* 280
TeleRaid Commands 280
Teletype editor 51
TELNET protocol 261,269
Ten-key calculator pad 215
Terminal emulators 21
Testing the connection between two Xerox Lisp machines 228
TESTRELATION *(Function)* 176
TExec 287
TEXEC *(Function)* 287
TEXTMODULES 289
TFTP.CLOSEFILE *(Function)* 272
TFTP.GET *(Function)* 272
TFTP.OPENFILE *(Function)* 272
TFTP.PUT *(Function)* 271
TFTP.SERVER *(Function)* 272
Thumbing 74
TIMEZONECOMP *(Variable)* 200
Tracing and test Functions 265
Trailer encapsulation 277
Transferring files 143,258
TRANSLATE-3-BY-3 *(Function)* 195
TRANSLATE-4-BY-4 *(Function)* 195
Translating between the file-naming conventions 268
Transport layer 260
Transport control protocol 264
Trees 111
TRIM.BITMAP *(Function)* 66
Trivial file transfer protocol 271
Troubleshooting Problems With FileBrowser 94
TTY port 1,219
TTY.DEFAULT.INIT.INFO *(Variable)* 226
TTY.GET.PARAMETERS *(Function)* 228
TTY.INIT *(Function)* 226
TTY.SET.PARAMETERS *(Function)* 227
TTY.SHUTDOWN *(Function)* 228
TTYCHAT 228
Typing Characters to DEdit 53

U

UDP.APPEND.BYTE *(Function)* 271
UDP.APPEND.CELL *(Function)* 271
UDP.APPEND.STRING *(Function)* 271
UDP.APPEND.WORD *(Function)* 271
UDP.CLOSE.SOCKET *(Function)* 270
UDP.EXCHANGE *(Function)* 271
UDP.GET *(Function)* 270
UDP.INIT *(Function)* 270
UDP.OPEN.SOCKET *(Function)* 270
UDP.SEND *(Function)* 271
UDP.SETUP *(Function)* 270
UDP.SOCKET.EVENT *(Function)* 270
UDP.SOCKET.NUMBER *(Function)* 270
UDP.STOP *(Function)* 270
UNCOLORIZEBITMAP *(Function)* 44
UNDELETE *(Command)* 77
Unechoed input mode 288
UNIX 232,272
UNIX-STREAM-CLOSE *(Function)* 305
UNIXChat 303
UNIXComm 305
Unread 54
UNSAVEFNS *(Function)* 177
UPDATECHANGED *(Function)* 177
UPDATEFN *(Function)* 177
Updating the MasterScope Data Base 174
upgrade-semicolon-comments *(Variable)* 292
User Commands for NSMaintain 204
User datagram protocol 270
USERWORDS *(Function)* 288
Using

4045XLP stream 8
4045 as a default printing host 3
FileBrowser window 73
Keyboard editor 150
modems 223
RS232 streams 222
TExec 288
TTY port 226
TTY streams 228

V

Verifying TCP connections 258
Version Control 46
VFIND.PACKAGE *(Function)* 286
VFIND.SYMBOL *(Function)* 286
VGETBASE0 *(Function)* 285
VGETDEFN *(Function)* 285
VGETPROPLIST *(Function)* 285
VGETTOPVAL *(Function)* 285
VGETVAL *(Function)* 285
Viewing
an existing bitmap 64
frames from a stack 281
system stack 282
Violation of the IP standard 262
Virtual graph nodes 113
Virtual memory 279
Virtual terminal I/O 269
VirtualKeyboards 149,307
VKBD.CONFIGURATIONS *(Variable)* 154
VKBD.KNOWN-KEYBOARDS *(Variable)* 311
VLOADFNS *(Function)* 283
VLOADVAR *(Function)* 283
VPUTBASE0 *(Function)* 286
VRAID *(Function)* 284
VSAVEWORK *(Function)* 283
VSETTOPVAL *(Function)* 285
VT100 Chat 28
VTYPENAME *(Function)* 285
VUNSAVEDEF *(Function)* 283
VVAG2 *(Function)* 285
VYANKDEF *(Function)* 283
V\COPY *(Function)* 285
V\UNCOPY *(Function)* 285

W

Weitek floating-point chip set 29
When To Copy 46
Where-Is 313
Wild cards 72
WINDOWPROP *(Function)* 42
WORPCURSOR *(Function)* 43
Wrapper Functions 193

X

XCVR interface cable 253
Xerox 2700-II laser printer 1
Xerox 4045 Laser CP 1
Xerox Character Codes 19
Xerox/Diablo 630 1
XIP *(record)* 69
XNS host number 200
XNS Time Service 199

(

(GETSCREENREGION *(Function)* 40
(MAKE-HOMOGENEOUS-4-BY-4 *(Function)* 194

4

4045 configuration cartridge 2
4045 Emulation Mode Selection 2
4045 Fonts 9
4045 Parameter Names and Values 5
4045 Port Initialization 3
4045 Port Selection 2
4045 PROM and Software Compatibility 1
4045XLP.CHANGE.MODE *(Function)* 5
4045XLP.DEFAULTS *(Variable)* 5
4045XLP.GET.PARAMETERS *(Function)* 7
4045XLP.PARAMETERS *(record)* 5
4045XLP.SET.PARAMETERS *(Function)* 6
4045XLPStream 1
4045XLPstream Options 5
4045XLPSTREAM.VERSION *(Variable)* 1