

inmos[®]

INMOS Dx305 occam 2 Toolset Master Index



INMOS is a member of the SGS-THOMSON Microelectronics Group

Master index

Key

This master index covers four manuals belonging to the Toolset Documentation set; the notation used to refer to individual documents is as follows:

UG indicates the 'occam 2 Toolset User Guide' 72-TDS-366-01.

TR indicates the 'occam 2 Toolset Reference Manual' 72-TDS-367-01.

LR indicates the 'occam 2 Language and Libraries Reference Manual' 72-TDS-368-01.

PN indicates the document: 'Performance Improvement with the Dx305 occam 2 Toolset' 72-TDS-379-00.

Symbols

.STATIC: *UG* 248, 292

.VSPTR: *UG* 248, 292

.WSSIZE: *UG* 248, 292

!, idebug: *TR* 101, 104, 111, 125

::, idebug: *TR* 115

#

idebug: *TR* 85

idump: *TR* 143

isim: *TR* 274

#alias: *TR* 188

#COMMENT: *UG* 8; *TR* 15; *LR* 139

#define, linker directive: *TR* 189

#IMPORT: *UG* 8; *TR* 14; *LR* 139

#INCLUDE: *UG* 8, 51, 52, 53; *TR* 13; *LR* 139
in configuration language: *UG* 263

#include, linker directive: *TR* 189

#mainentry: *TR* 189; *PI* 13

#OPTION: *UG* 8; *TR* 16; *LR* 139;
PI 6, 7

#PRAGMA: *UG* 8; *TR* 17; *LR* 139

COMMENT: *TR* 18

EXTERNAL: *UG* 200; *TR* 18

LINKAGE: *UG* 43; *TR* 18, 190;
LR 148; *PI* 13, 32

PERMITALIASES: *TR* 19; *PI* 7

SHARED: *TR* 19; *LR* 165; *PI* 7

TRANSLATE: *UG* 201; *TR* 14, 20

#pragma

IMS_linkage: *TR* 190

IMS_nolink: *UG* 211

IMS_translate: *UG* 201

#reference: *TR* 189

#SECTION: *UG* 43

#section: *TR* 190; *LR* 148; *PI* 13

#USE: *UG* 8, 52, 55; *TR* 14; *LR* 3, 4, 139
in configuration language: *UG* 263

\$

idebug: *TR* 85

idump: *TR* 143


%

idebug: *TR* 85, 116

imap: *TR* 241

© INMOS Limited 1993. This document may not be copied, in whole or in part, without prior written consent of INMOS.

 **inmos**, IMS, occam and DS-Link are trademarks of INMOS Limited.

 **SGS-THOMSON** is a registered trademark of the SGS-THOMSON Microelectronics Group.

INMOS Limited is a member of the SGS-THOMSON Microelectronics Group.

INMOS document number: 72 TDS 378 00

isim: *TR 274*

@, iserver: *TR 261*

+, idebug: *TR 126*

++, idebug: *TR 125*

***, idebug:** *TR 92, 116, 121, 123, 128*

**** , idebug:** *TR 123, 128*

Numbers

2D block move: *LR 5*

A

Abbreviation

checking: *LR 164*

configuration language: *UG 79, 266*

loop unrolling: *PI 21*

of **PLACED** objects: *PI 24*

of variables: *PI 20*

Abort: *UG 120*

interrupt: *UG 45*

link communication: *UG 258*

program: *UG 45*

Accuracy of floating point arithmetic: *LR 22*

ACOS: *LR 38, 57*

Action strings, in makefiles: *TR 235*

ADDRESSOF: *UG 292*

Alias: *UG 293*

Alias check: *UG 48, 293; TR 5, 16*
disable: *TR 10, 19*

Alias checking: *LR 157; PI 6*

arrays: *LR 158*

disable: *PI 6*

effect of disabling: *LR 159*

rules: *LR 157*

warning messages: *UG 49*

Alignment: *LR 151*

word: *UG 240*

Allocating

channels to links: *UG 242*

specific workspace locations: *UG 241*

ALOG: *LR 27, 45*

ALOG10: *LR 28, 47*

ALT: *TR 9; LR 154; PI 29*

Analyse: *UG 105, 135, 241, 293; TR 74, 82, 83, 85*

use when debugging: *UG 107*

ANSI C toolset: *UG 68*

ANSI screen protocol: *LR 103*

ANSI-IEEE standard 754: *LR 21*

Apollo: *LR 80*

append.char: *LR 116*

append.hex.int: *LR 117*

append.hex.int64: *LR 117*

append.int: *LR 116*

append.int64: *LR 116*

append.real32: *LR 117*

append.real64: *LR 117*

append.text: *LR 116*

ARC: *UG 69, 78, 85, 264*

Areg: *UG 134, 281, 291*

Argument reduction: *LR 22*

Arithmetic functions

floating point support: *LR 13*

IEEE behavior: *LR 6*

occam: *LR 6*

Array

alias checking: *LR 158*

as argument: *UG 157*

channel: *LR 152*

constant: *LR 155*

counted input: *LR 144*

occam: *UG 276*

of channels: *UG 242*

of pointers: *LR 152*

passing between languages: *UG 203*

placing on-chip: *PI 31*

retyping: *LR 145*

subranges: *TR 115, 125*

subscripts: *PI 29*

unknown size: *LR 155*

usage checking: *LR 162*

ASIN: *LR 37, 56*

ASM: *UG 245, 246; TR 16, 34; LR 139, 142*

channel use: *UG 292*

examples: *UG 247*

predefined names: *UG 248*

syntax: *UG 289*

Assembly code: *LR 142*

direct instructions: *UG 246*

indirect instructions: *UG 246*

insertion into occam: *UG 245*

instruction set: *UG 281*

operands: *UG 246*

prefix instructions: *UG 246*

primary operations: *UG 246*

ASSERT: *UG 48; TR 9, 32; LR 18; PI 29*

Assigning code to transputers: *UG 14*

ATAN: *LR 39, 58*

ATAN2: *LR 40, 59*

Automated program building: *UG 97*

B

B004: *UG 44, 106; TR 285*

B008: *UG 107; TR 285*
PC motherboard: *UG 105*

B014, VME motherboard: *UG 105*

B016, VME motherboard: *UG 105*

[BACKTRACE]: *TR 112*

Backtrace: *UG 157, 174, 177, 293*

Backus-Naur Form, configuration language: *UG 263*

Benchmarks: *PI 33*

Big endian: *UG 293*

binary. See *output.format*

Binary byte stream: *LR 68*

Binary lister: *TR 205*

command line: *TR 206*

errors: *TR 219*

Binary output, *ieprom:* *UG 235; TR 170*

Bit manipulation: *LR 5, 10*

BITCOUNT: *LR 10*

BITREVNBITS: *LR 10*

BITREWORD: *LR 10*

Block CRC library: *LR 126*

Block mode, *ieprom:* *TR 171*

Block move: *PI 24*

BNF. See Backus-Naur Form

Boards

boot from link: *UG 105*

boot from ROM: *UG 105*

connections: *UG 105*

IMS B004: *UG 106*

IMS B008: *UG 105*

IMS B014: *UG 105*

IMS B016: *UG 105*

types: *UG 106*

wiring: *TR 74*

BOOL: *UG 276; LR 152*

BOOLTOSTRING: *LR 123*

Boot from link: *TR 145*

boards: *UG 105*

collector memory map: *TR 61, 64*

default collector output: *TR 52*

loading mechanism: *UG 104*

Boot from ROM: *TR 52, 59, 145, 163*

boards: *UG 105*

code, debugging: *UG 120*

configurer options: *TR 30*

Bootable code: *UG 44, 293; TR 27, 47*

creating: *UG 38*

bootable.file: *TR 166*

Bootstrap: *UG 293*
 alternatives: *TR 60*
 example: *TR 347*
 loaders: *TR 60, 348*

Bptr0: *UG 134*

Bptr1: *UG 134*

BREAK key: *UG 45; TR 255, 288*

Breakpoint: *UG 146; TR 90, 276*
 clearing: *UG 175*
 commands: *TR 90*
 hardware support: *UG 125*
 menu: *TR 90*
 phantom: *UG 155*
 setting and clearing: *UG 128*

Breakpoint debugging
 See also Debugging; Interactive debugging
 methods: *TR 75*

Breg: *UG 134, 282, 291*

Buffering processes: *UG 113*

Buffers: *LR 95*

Build files, library: *UG 296*

Building libraries: *TR 179*

BYTE: *UG 276; LR 152*

byte.select: *TR 168*

C

C run time library: *LR 65*

C.ENTRY: *UG 30*

C.ENTRYD: *UG 29*

C.ENTRYD.RC: *UG 29*

callc.lib: *UG 213, 214*

callc.lnk: *UG 212*

Capability: *UG 294; TR 255, 259*
 specific host: *TR 261*

Caplin QT0: *LR 81*

CASE: *LR 154*
 debugging occam: *UG 148*

CAT: *LR 81*

CAUSEERROR: *UG 260; LR 18*

centry.lib: *TR 15*

CHAN: *UG 276*

CHAN OF ANY: *TR 9, 33; LR 142*

CHAN OF SP: *UG 41, 111*

Change control: *UG 54*

CHANGE FILE: *TR 113*

Change processor, debugging: *TR 103*

CHANNEL: *TR 110*

Channel: *UG 4*
 See also **CHAN OF**
 array: *UG 242; LR 152*
 array constructors: *UG 239, 243; LR 141*
 checking: *UG 49*
 configuration: *UG 68*
 direct: *UG 85*
 edge: *UG 85*
 fault handling: *UG 101, 257*
 hard: *UG 296*
 implementation: *LR 152*
 initialize: *UG 102*
 optimize: *UG 187*
 place: *LR 143*
 place at address: *UG 240*
 placement: *UG 85, 91, 242*
 protocols: *LR 141*
 reinitialize: *UG 102, 258*
 reliable: *UG 101*
 reserved: *UG 224*
 reset: *UG 102, 259*
 retyping: *UG 243; LR 140*
 soft: *UG 85, 299*
 usage: *UG 49*
 usage checking: *UG 49; LR 162*
 use within **ASM**: *UG 292*
 virtual: *UG 85, 86*

channel.h: *UG 276*

char.pos: *LR 114*

Character identification: *LR 112*

Check
 alias: *UG 48*

channel: *UG 48*
 configuration: *UG 102*
 usage: *UG 48*

Checking
 a network: *TR 91*
 occam code: *UG 246*

Clearing
 breakpoint: *UG 175*
 error flag: *UG 108, 126; TR 133, 289*

clibs.lnk: *UG 31, 232*

clibsrld.lnk: *UG 31, 213, 232*

CLIP2D: *LR 10*

Clock: *UG 134*
 displayed on Monitor page: *UG 136*

Clock rate: *LR 155*

Clock0: *UG 134*

Clock1: *UG 134*

CMS: *LR 81*

cnonconf.lnk: *UG 29*

Code
 allocation in memory: *TR 34; LR 147*
 using **PLACE** statement: *UG 239*
 insertion: *UG 239, 245; TR 16*
 listing: *TR 210*
 placement: *UG 90, 181*
 placing on-chip: *PI 32*
 position in memory: *UG 88, 181; TR 32, 53, 55*

Collecting, simple program: *UG 37*

Collector: *UG 18*
 command line: *TR 48*
 command line options: *PI 15*
 error messages: *TR 67*
 information: *PI 9*
 input files: *TR 51*
 output files: *TR 51*
 non-bootable: *TR 58*

Command line: *TR 293*
 server: *LR 76, 77*

Command line options
icollect: *TR 50*
idebug: *TR 77*
iemit: *TR 146*
ieprom: *TR 165*
ilibr: *TR 176*
ilink: *TR 187*
ilist: *TR 207*
imakef: *TR 226*
imap: *TR 241*
iserver: *TR 253*
isim: *TR 272*
iskip: *TR 286*
oc: *TR 7*
occonf: *TR 30*
 specify transputer target: *TR 308*

COMMENT pragma: *TR 18*

Comments
 in EPROM control files: *TR 165*
 in object code: *TR 15, 18*

Communicating Sequential Processes: *UG 4, 294, 305*

Communication. See Channel

Compare memory, debugging: *TR 91*

compare.strings: *LR 113*

Compatibility
 error modes: *TR 8*
 with previous toolsets: *TR 32*

Compilation: *UG 42*
 error modes: *TR 8*
 information: *UG 43*
 order of: *TR 11*
 separate: *UG 52*
 targets: *TR 7*
 unit: *UG 52*

Compiler
 command line: *TR 4*
 command line options: *TR 5, 7; PI 11*
 diagnostics, implementation data: *TR 299*
 directive: *UG 43*

directives: *TR* 12; *LR* 139
 syntax: *TR* 13
 error messages: *TR* 20
 file names: *TR* 7
 implementation restrictions: *LR* 155
 information: *PI* 9
 language keywords: *LR* 139
 libraries: *LR* 5
 disabling: *TR* 6
 introduction: *UG* 9
 occam: *UG* 212, 219, 294
 user functions: *LR* 5
 memory allocation: *LR* 147
 memory map: *TR* 11
 occam: *TR* 3
 optimizations: *PI* 17
 in debugging: *UG* 158
 selective loading of libraries: *TR* 178
 warning messages: *TR* 20
 warnings, enable/disable: *TR* 9
 Compiling: *UG* 17
 for a range of transputers: *TR* 302
 for debugging: *UG* 117
 for other transputer types: *UG* 39
 introduction: *UG* 42
 simple example: *UG* 35
 Concurrency, hardware support: *UG* 2
 CONFIG: *UG* 69, 264, 267
 Configuration: *UG* 67, 294
 channels: *UG* 85
 checking: *UG* 102
 code & data placement: *UG* 10
 debugging considerations: *UG* 100, 101, 156
 description: *UG* 67, 70; *TR* 27
 multiple transputer example: *UG* 94
 error modes: *TR* 31
 examples: *UG* 72, 73, 80
 multiple transputer: *UG* 93
 simple: *UG* 36
 virtual routing: *UG* 194
 hardware description: *UG* 68, 74
 host connection: *UG* 79
 language: *UG* 69
 abbreviations: *UG* 79
 constraints: *UG* 269
 introduction: *UG* 10
 optimizing memory: *PI* 14
 predefinitions: *UG* 265
 syntax: *UG* 263
 libraries of linked units: *UG* 82
 mapping: *UG* 68
 channels: *UG* 87
 description: *UG* 83
 processes: *UG* 84
 mixed language: *UG* 67
 model: *UG* 68
 OCCAM scope rules: *UG* 68
 reliable channels: *UG* 101, 257
 single transputer program: *UG* 38
 software description: *UG* 68, 81
 summary: *UG* 100
 using *imakef*: *UG* 97
 virtual routing: *UG* 86
 warning messages, enable/disable: *TR* 32
 Configurer: *UG* 18, 294; *TR* 27
 command line: *TR* 28
 error messages: *TR* 35
 memory map: *TR* 34
 options: *TR* 29, 30
 producing debuggable programs: *UG* 118
 search paths: *TR* 30
 CONNECT: *UG* 69, 264
 Connecting
 boards: *UG* 105
 links: *UG* 69, 264
 subnetworks: *UG* 105
 Connection database: *TR* 260
 example: *TR* 263
 format: *TR* 262
 Connection manager: *UG* 294
 Constant arrays: *LR* 155
 merging: *PI* 17
 Constants: *LR* 4
 cached in table: *PI* 17

folding: *PI* 17
 include files: *UG* 110
 sharing: *UG* 52
 CONTINUE FROM: *TR* 111
 Conventions
 command line options: *TR* 293
 command line syntax: *TR* 293
 error messages: *TR* 299
 filenames: *TR* 294
 imakef file extensions: *TR* 298
 search paths: *TR* 294
 standard file extensions: *TR* 295
 convert.lib: *LR* 3, 119
 Core dump: *UG* 294; *TR* 74, 143, 279
 listing: *TR* 218
 COS: *LR* 34, 53
 COSH: *LR* 42, 61
 CRC functions: *LR* 11, 126
 crc.lib: *LR* 3
 CRCBYTE: *LR* 12
 CRCFROMLSB: *LR* 127
 CRCFROMMSB: *LR* 127
 CRCWORD: *LR* 12
 Creg: *UG* 134, 283, 291
 CSP: *UG* 4, 294, 305
 cstartrd.lnk: *UG* 29, 212
 cstartup.lnk: *UG* 29, 212
 Current location, in debugger: *TR* 112
 Cursor positioning: *TR* 341

D
 DACOS: *LR* 38, 57
 DALOG: *LR* 27, 45
 DALOG10: *LR* 28, 47
 DASIN: *LR* 37, 56
 Data, listing all: *TR* 216
 DATAN: *LR* 39, 58
 DATAN2: *LR* 40, 59
 Date: *LR* 94
 dblmath.lib: *LR* 3, 20, 26
 DCOS: *LR* 34, 53
 DCOSH: *LR* 42, 61
 Dead code elimination: *PI* 17
 deadfix.occ: *UG* 150
 Deadlock: *UG* 148, 294
 deadlock.occ: *UG* 149
 Debug, support functions: *TR* 97, 111
 Debug library: *UG* 142
 DEBUG.ASSERT: *LR* 130
 debug.lib: *LR* 3, 130
 DEBUG.MESSAGE: *LR* 130
 DEBUG.STOP: *LR* 130
 DEBUG.TIMER: *UG* 150; *LR* 131
 Debuggable programs: *UG* 116
 Debugger: *UG* 19; *TR* 73
 command line: *TR* 75
 environment variables: *TR* 78
 errors: *TR* 133
 hints: *UG* 147
 information: *PI* 9
 kernel: *UG* 123
 monitor commands
 definitions: *TR* 89–108
 editing functions: *TR* 86
 mapped by ITERM: *TR* 86
 summary: *TR* 86–88
 monitor page
 commands: *TR* 85
 scroll keys: *TR* 88
 symbolic commands: *TR* 88
 program hangs: *TR* 133
 quitting: *UG* 175
 scroll keys: *TR* 85
 symbolic functions: *TR* 108

Debugging: *UG 115*
 See also Interactive debugging;
 Monitor page; Post-mortem
 debugging
 abusing hard links: *UG 151*
 arrays as arguments: *UG 157*
 B004 boards: *TR 82*
 boot from ROM code: *UG 120*
 breakpoint: *UG 172*
 catching concurrent processes:
UG 155
 commands, only available in inter-
 active mode: *UG 128*
 compiler optimizations: *UG 158*
 confidence check: *UG 153*
 configured programs: *UG 118,*
156
 current location: *TR 112*
 data: *TR 5*
 deadfix.occ: *UG 150*
 deadlock.occ: *UG 149*
 direct channel functions: *UG 118*
 error modes: *UG 118*
 errors: *UG 158*
 examining the active network: *UG*
152
 example, C: *UG 159*
 goto process: *UG 178*
 hard parity errors: *UG 120, 122*
 important points: *UG 151*
 information: *UG 117*
 INSPECT: *UG 152*
 inspecting channels: *UG 177; TR*
110
 inspecting memory: *TR 125*
 inspecting variables: *UG 176*
 interactive: *UG 116, 123, 172;*
TR 196; PI 4
 disabling: *UG 118*
 INTERRUPT key: *UG 153*
 invalid pointers: *UG 147*
 large shift values: *UG 157*
 library: *UG 142*
 library functions, in absence of
 idebug: *UG 145*
 loading programs: *UG 106*
 low level: *UG 132*
 memory size: *UG 156*
 mixed language: *UG 116*
 Monitor page: *UG 132*
 options, for different boards: *TR*
84
 post-mortem: *UG 115, 120, 175*
 program crashes: *UG 154*
 program hangs: *UG 154*
 program termination: *TR 79*
 root transputer: *UG 123*
 seterr: *UG 156*
 single step: *TR 280*
 soft configuration channels: *UG*
147
 tracing processes: *UG 177*
 TRAMs: *TR 82*
 undetected program crashes: *UG*
154
 use of isim: *UG 116*
 virtual links: *UG 152*
 Debugging support library: *LR 130*
 DEC VAX: *LR 80*
 Default
 command line arguments: *UG 27*
 error mode: *UG 42, 118*
 memory map: *UG 181; TR 34*
 transputer type: *UG 42*
 DELETE: *TR 228*
 delete.string: *LR 115*
 DEXP: *LR 29, 48*
 Direct channels: *UG 85, 101*
 Direct instructions: *UG 246, 281*
 Directives, linker: *TR 188; PI 13*
 Directory path: *TR 294*
 Disable
 alias checking: *UG 49; TR 19*
 configurer warnings: *TR 32*
 error detection: *TR 8*
 interactive debugging: *UG 48,*
118; TR 16, 33
 range checking: *UG 47*
 range checks: *TR 9, 16, 31*
 run-time checks: *UG 46; TR 9,*
16, 31
 separate vector space: *TR 16*

usage checking: *UG 49; TR 16,*
19
 vector space: *UG 50*
 virtual routing: *UG 48, 117; TR*
32
 warning messages: *TR 9*
 Disassemble memory: *TR 92*
 Display
 debugger help page: *UG 129*
 debugging messages: *TR 105*
 memory: *UG 137*
 memory in hex: *TR 95*
 memory map: *UG 136*
 Monitor page: *UG 133*
 object code: *UG 44; TR 205*
 process queues: *UG 178*
 processes: *TR 105*
 reference: *TR 216*
 run queues: *TR 104*
 source code: *UG 130*
 timer queues: *TR 105*
 DO: *UG 69, 268*
 DOS: *UG 111; LR 81*
 specific library: *LR 131*
 dos.call.interrupt: *LR 133*
 dos.port.read: *LR 134*
 dos.port.write: *LR 135*
 dos.read.regs: *LR 134*
 dos.receive.block: *LR 132*
 dos.send.block: *LR 132*
 Down, subsystem wiring: *UG 105*
 DPOWER: *LR 30, 49*
 DRAM timing parameters: *TR 155*
 DRAN: *LR 44, 63*
 DRAW2D: *LR 9*
 DRX-11: *LR 81*
 DSIN: *LR 32, 51*
 DSINH: *LR 41, 60*
 DTAN: *LR 35, 54*
 DTANH: *LR 43, 62*
 Dynamic code loading: *UG 239,*
250
 examples: *UG 253*
 file format: *TR 58*
 listing files: *TR 218*
 procedures: *LR 15*
 E
 Early write: *TR 153*
 Echoed keyboard input: *LR 84*
 EDGE: *UG 69, 77, 264*
 Edge
 channel: *UG 85, 86*
 declaring: *UG 77*
 Editing functions: *TR 86*
 Editing makefiles: *TR 235*
 Elementary functions: *LR 20, 25,*
45
 EMI: *TR 145*
 See also External Memory Inter-
 face
 clock period: *TR 153*
 END OF FILE: *TR 89, 113*
 End of file: *LR 73*
 end.offset: *TR 168*
 ENTER FILE: *TR 113*
 Entry point: *TR 20*
 Entry points
 C.ENTRY: *UG 30*
 C.ENTRYD: *UG 29*
 C.ENTRYD.RC: *UG 29*
 Environment variables: *UG 26, 34;*
TR 342; LR 78
 accessing through iserver: *TR*
328
 IBOARDSize: *TR 53*
 ICOLLECTARG: *TR 51*
 ICONDB: *TR 254, 261*
 ILIBRARG: *TR 176*
 ILINKARG: *TR 187*
 ILISTARG: *TR 208*

ISESSION: TR 254
ISIMBATCH: TR 281
ITERM: TR 82, 273
TRANSPUTER: TR 254, 260
 used by **idebug:** TR 78

EPROM: TR 30, 59
 code layout: TR 168
 devices: TR 172

EPROM program convertor: TR 163
 binary output: TR 170
 block mode: TR 171
 command line: TR 164
 control file: TR 165
 errors: TR 174
 hex dump: TR 170
 Intel extended hex format: TR 171
 Intel hex format: TR 171
 Motorola S-record format: TR 170
 output files: TR 170

EPROM programming: UG 21, 233; TR 163
 collecting: UG 236
 configuring: UG 236
 tools, introduction: UG 21

eprom.space: TR 166

eqstr: LR 114

Error: UG 105, 241, 295

Error
 detection, disable: UG 46; TR 8, 31
 handling: TR 299
 modes: UG 45, 295; TR 8, 191
 behavior: TR 31
 compatibility: TR 8, 31
 configurer: TR 31
 default: UG 42
 HALT: UG 45, 295; TR 8; PI 5
 in debugging: UG 118
 selective loading of libraries: TR 178
 STOP: UG 45, 295; TR 8; PI 5

UNDEFINED: UG 46; TR 9, 17, 31; PI 5
UNIVERSAL: UG 46, 295; TR 8; PI 5
 reporting: UG 25
 runtime: TR 300
 severities: TR 299

Error flag
 clearing in a network: UG 108, 126; TR 133, 289
 detection in interactive debugging: TR 83
 displayed on Monitor page: UG 134, 135
 of a subsystem: UG 105
 setting: UG 239, 260

Error messages
 format: TR 299
icollect: TR 67
idebug: TR 133
idump: TR 144
iemit: TR 159
ieprom: TR 174
ilibr: TR 182
ilink: TR 199
ilist: TR 219
imakef: TR 235
imap: TR 249
iserver: TR 266
 additional: TR 268
isim: TR 282
iskip: TR 289
oc: TR 20
occonf: TR 35
 runtime memory initialization: TR 56

Escape characters: LR 139

Ethernet: UG 295; TR 251

Event: UG 141

Event: UG 295; TR 99, 279

Examples
#COMMENT: TR 16
#IMPORT: TR 15
#OPTION: TR 17
#PRAGMA EXTERNAL: TR 18
#PRAGMA LINKAGE: TR 19

#PRAGMA TRANSLATE: TR 20
ASM: UG 247
 bootstrap loader: TR 347
 collecting: UG 37
 configuration: UG 36
 configuration mapping: UG 92
 connection database: TR 263
 deadlock: UG 149
 debugging C: UG 159
 debugging in post-mortem mode: UG 175
 debugging occam: UG 168
 dynamic code loading: UG 253
facts.c: UG 160
facts.occ: UG 169
ieprom control file: TR 173
imakef: TR 228
 mixed languages: TR 233
 occam: TR 231
INLINE: LR 144
 linking equivalent occam process: UG 232
 multiple transputer: UG 93
 multiplexing to host: UG 113
 network description: UG 80
oc: TR 5
occonf: TR 28
 optimized filter program: UG 194
 passing C parameters: UG 273
 phantom breakpoints: UG 155
 pipeline sorter: UG 57
 placing channels: UG 242
 resetting B004: UG 241
 retyping channels: UG 244
 running a program: UG 37
 simple program: UG 34
simple.occ: UG 34
 single transputer program: UG 38
 skip load: UG 107
 skipping a single processor: TR 287
 skipping multiple transputers: TR 287
 software description: UG 83
sorter.occ: UG 62
 type 1 interface: UG 227
 type 2 interface: UG 229

type 3 interface: UG 231
 use of debugging library: UG 144
 virtual routing: UG 196

Executable code: UG 17

EXIT FILE: TR 113

EXP: LR 29, 48

Exported names, listing: TR 212

Expressions
 conditional: PI 28
 pre-evaluate: PI 28

Extended data types: LR 5
 occam: UG 295

Extensions
 file: UG 22; TR 295
 required by **imakef:** TR 222, 298
 language: TR 3
 to occam: LR 139

External memory interface: UG 295; TR 145

EXTERNAL pragma: TR 18

External references, listing: TR 219

extintel. See **output.format**

Extraction of library modules: TR 192

Extraordinary link handling library: LR 128

Extraordinary use of links: UG 239, 256

F

facts.c: UG 160
 compiling and loading: UG 164

facts.occ: UG 169
 compiling and loading: UG 171

FALSE: PI 28

Farm. See **Processor farm**

File
 access: LR 66
 access errors: LR 74

deletion: *LR 74*
 extensions: *UG 22; TR 295*
imakef: *UG 23*
imap source files: *TR 241*
 required by *imakef*: *TR 222, 298*
 identification: *TR 217, 295*
 name
 conventions: *TR 294*
 length: *LR 156*
 output: *LR 89*
 positioning: *LR 73*
 renaming: *LR 75*
 streams: *UG 110*
 types read by *idebug*: *TR 77*

FINISH: *TR 114*

Floating point
 arithmetic: *LR 5*
 compiler optimization: *LR 148*
 instructions: *UG 285*
 registers: *UG 249*
 representation: *LR 21*

FMUL: *LR 45*

FORTRAN: *UG xix, 120*

FPError: *UG 134*

Fptr0: *UG 134*

Fptr1: *UG 134*

FRACMUL: *LR 13*

Free memory: *UG 42; TR 35*

Free variables: *UG 295*

FUNCTION: *PI 20, 27*

Function keys, *idebug*: *TR 82*

G

Gateway: *UG 295*

Generated error: *LR 22, 24*

GET ADDRESS: *TR 113*

Getting started: *UG 33*

Global static base: *UG 209, 213*

Go to process: *TR 95*

GOTO LINE: *TR 113*

Grid, network topology: *UG 3*

GUY: *TR 9, 33; LR 139, 146*

H

HALT error mode: *UG 45; TR 8, 31; PI 5*
 in debugging: *UG 118*

HaltOnError: *UG 45, 134; PI 5*

Hard channels: *UG 296*

Hardware configuration description:
UG 74, 264

Hardware support
 for breakpointing: *UG 125*
 for concurrency: *UG 2*

Heap area: *TR 54*
 mixed language programs: *UG 213, 225*
 position in memory: *TR 53*

HELIOS: *LR 81*

HELP: *TR 88, 108, 113*

Help, page in debugger: *UG 129*

hex. See *output.format*

HEX16TOSTRING: *LR 120*

HEX32TOSTRING: *LR 121*

HEX64TOSTRING: *LR 121*

Hexadecimal
 arguments to *idump*: *TR 143*
 listing: *TR 212*

Hexadecimal format
 for environment variables: *UG 27*
 for EPROM: *UG 235; TR 170*
 syntax: *UG 27*

HEXTOSTRING: *LR 120*

HOST: *UG 79, 265*

Host: *UG 296*
 access: *LR 76*
 access to services: *UG 109*
 channel protocols: *UG 111*
 communications: *UG 109*

connection, in configuration
 language: *UG 79*
 dependencies: *UG 25*
 command line syntax: *UG 25*
 filenames: *UG 26*
 search paths: *UG 26*
 environment variables: *UG 26*
 for capability: *TR 261*
 system call: *LR 79*
 versions: *UG xvii; TR xix; LR vii; PI v*

Host file server: *UG 296; TR 251*
 file streams: *UG 110*
 introduction: *UG 109*
 library: *LR 65*
 terminating: *TR 288*

Hostio library: *UG 110*

hostio.inc: *UG 110*

hostio.lib: *UG 110; LR 3*

Hyperbolic functions: *LR 41*

I

IBM 370: *LR 80*

IBM PC: *UG 7, 111*
 386: *UG 25*

IBoARDSIZE: *UG 27, 34, 42; TR 53, 79*
 errors: *TR 56*

icc: *UG 9*

ICCARG: *UG 28*

icconf: *UG 118*

ICCONFARG: *UG 28*

icollect: *UG 18*
 command line: *TR 48*
 command line options: *TR 50*
 environment variables: *TR 51, 53*
 errors: *TR 67*

ICOLLECTARG: *UG 28; TR 51*

ICONDB: *UG 27, 34; TR 254, 261*

idebug: *UG 19; TR 73*
 command line: *TR 75*
 options: *TR 77*
 environment variables: *TR 78*
 errors: *TR 133*
 help page: *UG 129*
 interactive mode: *TR 81*
 post-mortem debugging: *TR 79*
 restarting: *TR 81*

IDEBUGSIZE: *UG 27; TR 79*
 errors: *TR 133*

idump: *UG 20; TR 74, 143, 255, 285*
 errors: *TR 144*

IEEE floating point arithmetic: *LR 5*

iemit: *UG 21; TR 145*
 command line: *TR 146*
 DRAM timing parameters: *TR 155*

errors: *TR 159*
 index page: *TR 148*
 input parameters: *TR 150*
 memory read cycle: *TR 156*
 memory write cycle: *TR 157*
 timing information: *TR 154*

ieprom: *UG 21, 233, 235; TR 163*
 command line: *TR 164*
 control file: *TR 165*
 errors: *TR 174*

IF: *UG 69, 98, 266; TR 9*
 debugging occam: *UG 148*

ilibr: *UG 20; TR 175, 177*
 command line: *TR 176*
 command line options: *TR 176*
 error messages: *TR 182*

ILIBRARG: *UG 28; TR 176*

ilink: *UG 17; TR 185*
 command line: *TR 186*
 indirect files: *TR 187*

ILINKARG: *UG 28; TR 187*

ilist: *UG 20; TR 15, 205; LR 4*
 command line: *TR 206*
 command line options: *TR 207*
 errors: *TR 219*

ILISTARG: *UG 28; TR 208*
imakef: *UG 20, 39, 54, 97; TR 3, 7, 13, 198, 221*
 command line: *TR 225*
 command line options: *TR 226*
 deleting intermediate files: *TR 228*
 errors: *TR 235*
 examples: *TR 228*
 file extensions: *TR 222, 298*
 file formats: *TR 234*
 linker indirect files: *TR 225, 227*
 mixed language example: *TR 233*
 occam examples: *TR 231*
 target files: *TR 222*
imap: *UG 20; TR 239; PI 10*
 command line: *TR 240*
 command line options: *TR 241*
 errors: *TR 249*
 output file structure: *TR 243*
Implementation
 channels: *LR 152*
 compiler diagnostics: *TR 299*
 occam: *LR 147*
 restrictions: *LR 155*
Importing C functions: *UG 213*
IMS B004: *TR 285; LR 81*
IMS B008: *TR 285; LR 81*
IMS B010: *LR 81*
IMS B011: *LR 81*
IMS B014: *LR 81*
IMS B015: *LR 81*
IMS B016: *LR 81*
IMS B404: *TR 83*
IMS D700: *LR 98*
IMS T800: *UG 135*
IN: *LR 139*
Include file: *UG 110, 296*
 oconf.inc: *UG 70, 187, 192*
Indirect instructions: *UG 246, 282*
inexact.NaN: *LR 21*
Infinity: *LR 21*
INFO: *TR 110*
Information, provided by toolset: *PI 9*
init.heap: *UG 214*
init.static: *UG 214*
Initialize
 channel: *UG 102*
 link: *UG 258*
 parity checked memory: *UG 126*
INLINE: *LR 139, 143; PI 27*
 disadvantage: *PI 31*
InputOrFail.c: *UG 102; LR 129*
InputOrFail.t: *UG 101; LR 128*
INQUEST: *UG xix; TR xxi, 34*
insert.string: *LR 115*
INSPECT: *UG 176; TR 109*
Inspect memory: *TR 96*
Instruction pointer: *UG 134*
 invalid: *UG 147*
Instruction prefixing: *PI 1*
Instruction set: *UG 281*
INT: *UG 276; LR 152*
INT16: *UG 276; LR 152*
 on 32-bit transputers: *PI 29*
INT16TOSTRING: *LR 120*
INT32: *UG 276; LR 152; PI 29*
INT32TOSTRING: *LR 120*
INT64: *UG 276; LR 152*
INT64TOSTRING: *LR 120*
intel. See *output.format*
Intel extended hex format: *UG 235*
 ieprom: *TR 170*
Intel hex format: *UG 235*
 ieprom: *TR 170*
Interactive debugging: *UG 116, 123, 129; PI 4*
 See also *Debugging*

addresses of variables: *UG 166*
 backtracing: *UG 166, 174*
 backtracing to `main()`: *UG 167*
 breakpoint commands: *UG 132*
 browsing source code: *UG 130*
 clearing a breakpoint: *UG 175*
 collector option: *TR 66*
 compiler option: *UG 48*
 compiler support: *TR 10*
 configurer option: *TR 33*
 detecting the error flag: *TR 83*
 disabling: *UG 48, 118; TR 33*
 entering `#include` files: *UG 168*
 inspecting by expression: *UG 167*
 inspecting variables: *UG 131, 166, 173*
 invocation: *TR 81*
 jumping down a channel: *UG 167, 174*
 jumping down channels: *UG 131*
 locating to code: *UG 130*
 methods: *TR 75*
 modifying a variable: *UG 167, 174*
 modifying variables: *UG 132*
 program loading: *UG 126*
 program termination: *UG 128*
 quitting: *UG 168, 175*
 resuming program: *UG 174*
 runtime kernel: *UG 123*
 setting breakpoints: *UG 165, 173*
 starting a program: *UG 166, 173*
 tracing procedure calls: *UG 131*
INTERRUPT: *TR 111*
Interrupt, program: *UG 45*
 in debugging: *UG 153*
INTTOSTRING: *LR 120*
Invalid pointers: *UG 147*
lptr: *UG 134, 281*
lptrIntSave: *UG 134*
iq systems: *UG 102, 108, 126*
is.digit: *LR 113*
is.hex.digit: *LR 113*
is.id.char: *LR 113*
is.in.range: *LR 112*
is.lower: *LR 113*
is.upper: *LR 113*
ISEARCH: *UG 27, 34; TR 30, 294; LR 69*
iserver: *UG 18, 103, 109; TR 251, 285*
 accessing transputers: *TR 260*
 capability: *TR 255*
 command line: *TR 252*
 command line options: *TR 252*
 connection manager: *TR 265*
 environment variables: *TR 254*
 error codes: *TR 266*
 error messages: *TR 266*
 exit codes: *TR 266*
 functions: *TR 251*
 halt system error mode: *TR 255*
 loading programs: *TR 254*
 new features: *TR 265*
 passing parameters to a program: *TR 255*
 protocol: *TR 309*
 ALSYS – Alsys call: *TR 335*
 CommandArgs – get command line arguments: *TR 333*
 CommandLine – get server command line: *TR 329*
 Core – read peeked memory: *TR 330*
 Exit – exit the server: *TR 329*
 Fclose – close a file: *TR 312*
 Feof – test for end of file: *TR 321*
 Ferror – get file error status: *TR 322*
 FerrStat – Get file error status: *TR 324*
 Fflush – flush a stream: *TR 316*
 FGetBlock – read and return success: *TR 314*
 FGetRec – read a record: *TR 319*
 Fgets – read a line: *TR 315*
 file commands: *TR 311*
 FileExists: *TR 324*
 Fopen – open a file: *TR 311*

FopenRec – open record file:
TR 317

FPutBlock – write and return
success: TR 315

FPutEOF – write end-of-file: TR
320

FPutRec – write a record: TR
319

Fputs – write a line: TR 316

Fread – read block of data: TR
313

Fseek – set position in a file: TR
320

Ftell – find position in a file: TR
321

Fwrite – write block of data: TR
313

Getenv – get host variable: TR
326

GetInfo – get host and server
info: TR 332

Getkey – get keystroke: TR 325

host commands: TR 325

Isatty – terminal connect status:
TR 323

KPAR – Kpar call: TR 336

MSDOS – MS-DOS command:
TR 334

packets: TR 309

Pollkey – test for key press: TR
325

record structured file
commands: TR 317

record structured file format: TR
336

Remove – delete a file: TR 322

Rename – Rename a file: TR
323

Requestkey – request keyboard
event: TR 326

reserved commands: TR 334

server commands: TR 329

SocketA – socket library call:
TR 335

SocketM – socket library call:
TR 335

System – run a command: TR
327

termination codes: TR 337

Time – get the time of day: TR
327

Translate – translate an environ-
ment variable: TR 328

Version – server info: TR 331

record structured files: TR 266

session manager: TR 252, 256,
265

customizing interface: TR 258

specifying the transputer to use:
TR 255

stream identifier validation: TR
266

subsystem reset: TR 254

terminating: TR 255
on error: TR 255

user interrupt: TR 265

ISESSION: UG 27; TR 254, 256

isim: UG 21, 65, 146; TR 271

command line: TR 271

command line options: TR 272

errors: TR 282

ISIMBATCH: UG 27; TR 281

iskip: UG 18, 103; TR 74, 285

command line: TR 286

command line options: TR 286

errors: TR 289

ispy: UG 102, 108, 126; TR 133,
289

ITERM: UG 27; TR 79, 82, 273,
342

ITERM file
example listing: TR 344

format: TR 339

keyboard: TR 342

screen: TR 340

use by simulator: TR 273, 274

version: TR 340

J

JEDEC, symbol: TR 154, 156

Jump, in **ASM** code: UG 248

Jump instructions, in ROM: TR 169

Jump into program: TR 97

Jumping down a channel: UG 131,
167, 174; TR 110

K

KERNEL.RUN: UG 250; LR 15

Keyboard
definitions: TR 342
input: LR 82, 100
polling: LR 83

Keystream
input: LR 104
protocol: LR 98

ks: LR 98, 99

ks.keystream.sink: LR 101

ks.keystream.to.scrstream:
LR 101

ks.read.char: LR 104

ks.read.int: LR 105

ks.read.int64: LR 105

ks.read.line: LR 104

ks.read.real32: LR 105

ks.read.real64: LR 105

L

Label, in **ASM** code: UG 248

LAN: UG 296

Language extensions: LR 139

Large programs: UG 53

Large shift values: UG 157

Late write: TR 153

Lexical levels: LR 156

LFF files, listing: TR 218

libc.lib: TR 15

Librarian: UG 20; TR 175
command line: TR 176
concatenated input: TR 175

linked object input: TR 177
options: TR 176

Library: UG 296; LR 3
block CRC: LR 126
build files: UG 296
building: UG 55; TR 179
building optimized: TR 179

C runtime
full: UG 232
reduced: UG 232

compilation: TR 10

compiler: UG 9; LR 5

debugging: UG 142

debugging support: LR 130

displaying: LR 4

DOS specific: LR 131

extraction of modules: TR 192

extraordinary link handling: LR
128

host file server: LR 65

index: TR 175, 178

indirect files: TR 175, 177

imakef: TR 225

linking supplied libraries: UG 28;
TR 188

listing index: TR 214

maths: UG 9; LR 20

modules: TR 177

occam: UG 9, 221

of linked units: UG 82

optimized T4 series: LR 20, 45

selective loading of: TR 178

streamio: LR 98

string handling: LR 111

type conversion: LR 119

usage files: UG 296; TR 178

imakef: TR 225

using: UG 54

LINE DOWN: TR 89

Line parsing: LR 117

LINE UP: TR 88

Link: UG 3, 296
addresses: UG 243
debugging: TR 99
debugging simulated: TR 279
failure: UG 256; LR 128

introduction: *UG 2*
 long messages: *PI 37*
 map: *TR 197*
 optimization: *PI 35*
 prioritization: *PI 36*
 recovery from failure: *UG 258*
 reinitialize: *UG 258*
 virtual: *UG 86*

link: *UG 74, 265*

Link handling library: *LR 128*

linkaddr.inc: *UG 243*

LINKAGE pragma: *TR 18*

Linker: *UG 17, 297; TR 185*
 command line: *TR 186*
 command line options: *PI 13*
 compatible transputer classes:
TR 190
 directives: *TR 188; PI 13*
 errors: *TR 199*
 extraction of library modules: *TR 192*
 indirect files: *UG 28; TR 187*
 imakef: *TR 225, 227*
 information: *PI 9*
 LFF output: *TR 191*
 selective loading of libraries: *TR 178*
 startup files: *UG 28*
 clibs.lnk: *UG 232*
 clibsrd.lnk: *UG 232*
 TCOFF output: *TR 191*

Linking
 introduction: *UG 43*
 mixed language programs: *UG 212*
 simple example: *UG 35*
 transputer targets: *TR 301*

Linkops: *UG 297*

linkquota: *UG 75, 92, 190, 265*

Lister: *UG 20*
 See also *ilist*
 information: *PI 9*

Little endian: *UG 297*

LOAD.BYTE.VECTOR: *UG 250; LR 15, 17*

LOAD.INPUT.CHANNEL: *UG 250; LR 15, 16*

LOAD.INPUT.CHANNEL.VECTOR: *UG 250; LR 15, 16*

LOAD.OUTPUT.CHANNEL: *UG 250; LR 15, 17*

LOAD.OUTPUT.CHANNEL.VECTOR: *UG 250; LR 15, 17*

Loader: *UG 297*

Loading programs: *UG 44, 103*
 for breakpoint debugging: *UG 106*
 for debugging: *UG 106*
 for interactive debugging: *UG 126*
 introduction: *UG 18*
 iserver: *TR 251*
 iskip: *TR 288*
 methods: *UG 104*
 onto boards and subnetworks:
UG 104
 tools: *UG 103*

LoadStart: *UG 181; TR 34, 35, 61, 63*

localhost: *TR 261*

Location, in debugger: *TR 112*

location: *PI 8*

location.code: *UG 75, 90, 184, 265; TR 32; PI 14*

location.vs: *UG 75, 90, 184, 265; TR 32; PI 14*

location.ws: *UG 75, 90, 184, 265; TR 32; PI 14*

Logical name: *TR 342*

Logical processor: *UG 69*

Long reals: *LR 26*

Loop unrolling: *PI 21*

Lower case: *LR 113, 116*

Low-level programming: *UG 239*

M

Macros, in makefiles: *TR 234*

Main entry point: *TR 195*

MAIN.ENTRY: *UG 222*
 procedure interface: *UG 226*

MAKE: *UG 66*

Make programs: *TR 221*
 Borland: *TR 221*
 Gnu: *TR 221*
 Microsoft: *TR 221*
 UNIX: *TR 221*

Makefile generator: *UG 20, 39, 54; TR 221*
 command line: *TR 225*
 errors: *TR 235*

Makefiles: *UG 297*
 delete rule: *TR 235*
 editing: *TR 235*
 formats: *TR 234*
 macros: *TR 234*

MAP: *UG 69, 268*

MAPPING: *UG 69, 264, 267*

Mapping
 channels: *UG 87*
 description: *UG 83*
 examples: *UG 92*
 in configuration description: *UG 267*
 processes: *UG 84*
 with MAPPING: *UG 87*
 without MAPPING: *UG 88*

Master transputer, of a system: *UG 105*

Maths
 functions: *LR 6*
 libraries: *LR 20*
 introduction: *UG 9*

MemConfig: *TR 145*

MemnotWrD0: *TR 145*

Memory
 allocation: *UG 50; LR 147; PI 8*
 configuration
 ASCII output: *TR 148*
 customized: *TR 145*
 file: *TR 160*
 in PAL: *TR 145*
 in ROM: *TR 145, 168*
 PostScript output: *TR 148*
 standard: *TR 145, 153*
 table: *TR 158*
 configurer: *TR 145*
 command line: *TR 146*
 default configuration: *TR 148*
 errors: *TR 159*
 input parameters: *TR 150*
 interactive operation: *TR 148*
 output files: *TR 148*
 disassembly: *TR 276*
 Hex display: *TR 95*
 initializing: *UG 126, 225*
 inspecting: *TR 278*
 interface, configurable, T4 and T8
 series: *TR 145*
 map: *UG 181*
 mapper: *TR 239*
 command line: *TR 240*
 errors: *TR 249*
 on-chip: *UG 1; TR 18, 35*
 ordering code: *UG 88*
 parity-checked: *TR 57*
 placing code: *UG 90, 181*
 read cycle: *TR 156*
 reserved words
 lprIntSave: *UG 134*
 WdescIntSave: *UG 134*
 reserving: *UG 89*
 write cycle: *TR 157*

Memory dump: *UG 123*
 example: *UG 176*

Memory dumper: *UG 20; TR 143*
 command line: *TR 143*
 error messages: *TR 144*

Memory map: *UG 181; TR 100, 279*
 boot from link (network): *TR 64*
 boot from link (single processor):
TR 61
 boot from ROM: *TR 65*
 collector output: *TR 60*
 configurer: *TR 34*
 displayed on monitor page: *UG 136*

memory.configuration: *TR* 166

memsize: *UG* 74, 265

MemStart: *LR* 147

MemStart: *UG* 89, 135; *TR* 35, 61

MemWait: *TR* 153, 157
connection error: *TR* 159

Message length: *PI* 37

Mixed language programming: *UG* 199

#IMPORT directive: *TR* 14

example: *UG* 273

heap area: *UG* 213

importing C code: *UG* 213

introduction: *UG* 11

linking: *UG* 212

occam libraries: *UG* 221

reduced runtime library: *UG* 220

static area: *UG* 213

TRANSLATE pragma: *TR* 20

use of imakef: *TR* 233

vector space: *UG* 221

workspace: *UG* 221

MODIFY: *TR* 111

Module data, listing: *TR* 213

MONITOR: *TR* 114

Monitor page: *UG* 132
See also Debugging

breakpoint commands: *UG* 139

command format: *UG* 137

commands: *TR* 85

data displayed: *UG* 134

default address: *TR* 85

display virtual links: *TR* 107

Enter post-mortem: *TR* 106

examining memory: *UG* 137

exit: *TR* 106

locating processes: *UG* 137

selecting process: *UG* 138

simulator: *TR* 273

specifying process: *UG* 138

startup display: *UG* 133

switching processor: *UG* 138

Monitoring the error status: *TR* 288

MOSTNEG INT: *UG* 240, 247; *TR* 85; *LR* 148

MOSTPOS INT: *UG* 247; *LR* 147, 148, 156

Motorola S-record format: *UG* 235

ieprom: *TR* 171

MOVE2D: *LR* 9

Moving code and data areas: *UG* 88

Moving the cursor: *LR* 109

MS-DOS: *UG* 7, 25, 26, 27; *TR* 293

library: *LR* 131

introduction: *LR* 3

Multidimensional array, null element: *LR* 145

Multiplexing: *UG* 10

examples: *UG* 113

processes: *UG* 111

Multiplexors: *LR* 95, 96

Multiprocessor, optimization: *PI* 35

N

NaN: *LR* 21

NEC PC: *LR* 80

NETWORK: *UG* 69, 264

Network: *UG* 297

configuration: *UG* 67

description: *UG* 75

examples: *UG* 80

dump: *TR* 101

listing: *TR* 218

grid: *UG* 3

optimization: *PI* 35

pipeline: *UG* 3

Tree: *UG* 3

Next error: *TR* 93

next.int.from.line: *LR* 118

next.word.from.line: *LR* 118

nfix: *UG* 246

NODE: *UG* 68, 69, 264
attributes: *UG* 74

nodebug: *UG* 75, 265

Non-bootable files, format: *TR* 58

Non-configured programs. See icollect

NOP: *UG* 291

noprofile: *UG* 75, 265

Not a number. See NaN

notMemRd: *TR* 152

notMemS0: *TR* 152

notMemS4: *TR* 152

notMemWrB: *TR* 152

NotProcess: *UG* 135

Numerical parameters, interpretation by isim: *TR* 274

O

Object code: *UG* 297
displaying: *UG* 44; *TR* 205

Object file: *TR* 4
format: *UG* 7, 17

oc: *TR* 3

command line options: *TR* 7

error messages: *TR* 23

memory map: *TR* 11

syntax: *TR* 4

warning messages: *TR* 20

occam

array: *UG* 276

compiler libraries: *UG* 9, 294

configuration language: *UG* 263

equivalent process: *UG* 222

extended data types: *UG* 295

function return values: *UG* 272, 277

implementation: *LR* 147

interface code: *UG* 222; *TR* 52

language extensions: *LR* 139

libraries: *UG* 221; *LR* 3

low-level programming: *UG* 239

maths libraries: *UG* 9

mixing with C code: *UG* 199

obsolescent features: *LR* 145

programming model: *UG* 8

programs: *UG* 41

standard libraries: *UG* 9

occam 2 toolset, introduction: *UG* 7

occam2.lnk: *UG* 30, 43, 212

occam8.lnk: *UG* 30, 43, 212

occam.lnk: *UG* 30, 43, 212

occonf: *UG* 10, 67, 87, 118; *TR* 27

command line options: *TR* 29, 30

error messages: *TR* 35

interaction with idebug: *UG* 100

syntax: *TR* 28

warning messages: *TR* 36

occonf.inc: *UG* 70, 92, 186, 187, 192

On-chip memory: *UG* 1
use for program stack: *UG* 225

On-chip RAM: *UG* 43, 50; *TR* 18, 35

Operating systems

command lines: *UG* 25

dependencies: *UG* 25

MS-DOS: *UG* 25

SunOS: *UG* 25

UNIX: *UG* 25

VMS: *UG* 25

opr: *UG* 246

Optimization

code placement: *UG* 181; *TR* 18

code size: *PI* 31

links: *PI* 35

multiprocessor: *PI* 35

of source code: *PI* 19

performed by compiler: *PI* 17

space versus time: *PI* 3

use of occam: *PI* 26

virtual routing: *UG* 187

Options

in occam source: *TR* 16

prefix: *UG 25*
 specify transputer target: *TR 308*
 standard: *TR 293*
 unsupported: *UG 31; TR 294*

order.code: *UG 75, 88, 185, 265; TR 32; PI 8, 14*

order.vs: *UG 75, 88, 185, 265; TR 32; PI 8, 14*

order.ws: *UG 88, 185, 265; TR 32; PI 8, 14*

Out of memory errors, idebug: *TR 133*

output.address: *TR 168*

output.all: *TR 167*

output.block: *TR 167*

output.format: *TR 167*

OutputOrFail.c: *UG 102; LR 129*

OutputOrFail.t: *UG 101; LR 129*

P

PAGE DOWN: *TR 89*

PAGE UP: *TR 89*

PAR: *UG 81; TR 9; LR 149, 155*

Parallel processing
 example: *UG 62*
 introduction: *UG 4*

Parameter passing: *UG 271, 275*

Parameters
 access to: *PI 28*
 GSB: *UG 213*
 occam and C equivalents: *UG 271, 275*
 passing by reference: *UG 201*
 passing by value: *UG 201*
 TIMER: *UG 203*
 to **KERNEL.RUN**: *UG 251*

Parity checked memory, initializing:
UG 126

Parity error registers, displayed on
 Monitor page: *UG 136*

Parity errors, post-mortem debug-
 ging: *UG 120, 122*

ParityAddr: *UG 134*

ParityError: *UG 134*

Parsing command line: *LR 77*

Path searching: *TR 294*

Peek: *UG 298*

Performance improvement: *PI 1*

PERMITALIASES pragma: *TR 19*

prefix: *UG 246, 291*

Phantom breakpoints: *UG 155*

Physical processor: *UG 69*

Pipeline, network: *UG 3*

Pipeline sorter, example configura-
 tion: *UG 93*

Pipelining processes: *UG 113*

PLACE: *UG 51, 239; LR 143, 152, 155*

channels on links: *UG 242*

examples: *UG 240*

syntax: *LR 143*

then abbreviate: *PI 24*

when not to use: *PI 23*

PLACE name IN VECSPACE: *PI 6, 22*

PLACE name IN WORKSPACE: *PI 6, 22*

PLACED PAR: *UG 81*

Placement

array of channels: *LR 143*
 at address: *UG 239; LR 142*
 channels: *UG 85, 240; LR 143*
 code: *UG 88, 90, 181*
 in vector space: *LR 143*
 in workspace: *LR 143*
 variable in workspace: *UG 242*
 variables: *LR 143*

Pointer to channel: *UG 242; LR 152*

Poke: *UG 298*

PORT: *UG 276*

Port: *UG 241; TR 22*
 place at address: *UG 239; LR 142*

Post-mortem debugging: *UG 115, 120*

See also debugging
 communication on channels: *UG 141*

communication on links: *UG 141*
 communication on virtual links:
UG 141

dummy network: *TR 75*

from core dump: *TR 75*

hard parity errors: *UG 120, 122*

invocation: *TR 79*

locating procedures and func-
 tions: *UG 142*

outline of method: *UG 139*

R-mode programs: *TR 74*

stopped process: *UG 142*

stopped process location: *UG 140*

T-mode programs: *TR 74*

waiting on run queue: *UG 140*

waiting on timer queue: *UG 140*

PostScript: *UG 298*

POWER: *LR 30, 49*

Pragmas. See **#PRAGMA**

Preamble: *UG 298*

Prefixing instructions: *UG 246, 281; PI 1*

PRI ALT: *LR 154*

PRI PAR: *LR 149, 154, 155*

nested: *LR 155*

replicated: *LR 155*

Primary operations: *UG 246*

Priority: *UG 298; TR 104; LR 154*
 links: *PI 36*

PROC: *PI 20, 27*

PROC.ENTRY: *UG 223*
 procedure interface: *UG 227*

PROC.ENTRY.RC: *UG 223*

procedure interface: *UG 230*

ProcClockOut: *TR 152, 153*

Procedural interface data, listing:
TR 215

Procedure parameters: *UG 251*

Process: *UG 4, 298*

descriptor: *UG 134*

invalid: *UG 147*

memory map: *TR 105*

pointers, in debugging: *UG 135*

queue: *UG 135, 140; TR 280*

displaying: *TR 104*

scheduling: *UG 260*

PROCESSOR: *UG 68, 69, 267*

Processor

farms: *PI 39*

names: *TR 99*

types: *TR 301*

Program building, automated: *UG 66*

Program development, introduction:
UG 13

Program hangs, debugging: *UG 154*

Program termination, interactive
 debugging: *UG 128*

Programmable memory interface:
UG 1

Programs, loading: *UG 103*

Propagated error: *LR 23, 25*

PROTOCOL: *LR 156*

Protocol: *UG 298*

in debugging: *UG 123*

include files: *UG 110*

iserver: *UG 104; TR 309*

sharing: *UG 52*

SP: *UG 104, 111*

tag: *LR 151*

used by standard libraries: *UG 123*

Pseudo operations: *UG 247, 291*

Q

Queue
 process: UG 140, 178; TR 104, 280
 run: UG 135, 138, 140; TR 104, 280
 timer: UG 135, 138, 140; TR 280

Quit
 debugger: TR 104
 simulator: TR 279

R

R-mode programs: TR 74

RAM: UG 234; TR 18, 30, 59, 65
 on-chip
 improve use of: PI 1, 3
 not enough: PI 8
 placing arrays in: PI 31
 placing code in: PI 32

RAN: LR 44, 63

Random number generation: LR 44

Range reduction: LR 22

Read, strobe: TR 152

REAL: LR 65

Real numbers: LR 65

Real-time programming: UG 3

REAL32: UG 276

REAL32TOSTRING: LR 121

REAL64: UG 276

REAL64TOSTRING: LR 122

Reduced library: UG 232

REFRESH: TR 88, 108

Refresh period: TR 152

Registers
 Areg: UG 134, 281
 assigning value: TR 280
 Bptr0: UG 134
 Bptr1: UG 134

Breg: UG 134, 282

Clock0: UG 134

Clock1: UG 134

Creg: UG 134, 283

displayed on Monitor page: UG 135

Error: UG 134

FPErr0r: UG 134

Fptr0: UG 134

Fptr1: UG 134

HaltOnError: UG 134

lptr: UG 134, 281

memory dump: TR 144

ParityAddr: UG 134

ParityError: UG 134

Tptr0: UG 134

Tptr1: UG 134

Wdesc: UG 134

Wreg: UG 281

Reinitialise: UG 102, 258; LR 129

Reinitialize
 channels: UG 102
 link: UG 258

RELOCATE: TR 88, 107, 112

Replicated PAR: LR 149, 155

RESCHEDULE: UG 260; LR 18; PI 30

reserved: UG 75, 89, 183, 265; TR 35; PI 8, 14

Reserved channels, in occam
 equivalent processes: UG 224

Reserving memory: UG 89, 183

Reset: UG 105, 241, 298; TR 82
 use when debugging: UG 107

Resetting links: LR 128

Resource: TR 255

Restrictions, implementation: LR 155

RESUME: TR 88, 108, 111

Resume program
 from debugger: TR 98
 from simulator: TR 278

RETRACE: TR 88, 107, 112

Retry - server: TR 255

RETYPES: LR 145, 151

Retyping
 arrays: LR 145
 channels: UG 239, 243
 to a byte array: PI 25

ROM: TR 30, 59, 65, 163

ROM bootable code: UG 233
 processing configurations: UG 234

romsize: UG 74, 265

root: UG 74, 265

Root transputer: UG 298
 debugging: UG 106; TR 73
 loading over: UG 107; TR 285

root.processor.type: TR 166

ROUNDSN: LR 14

routecost: UG 75, 91, 188, 265

Run queue: UG 135, 140

Run queues, displaying: TR 104, 280

Run-time, checks: PI 31

Running programs: UG 44
 dynamically loaded: UG 250
 introduction: UG 18
 simple example: UG 37
 using isim: UG 38

S

Scalar channels: LR 152

Scalar workspace: TR 59

Scheduling: UG 239; PI 30
 occam processes: UG 260

Scheduling lists. See Process queue; Run queue

Scope rules: UG 147

Scoping of variables: PI 26

Screen definitions: TR 340

Screen size: TR 340

Screenstream
 output: LR 105
 protocol: LR 98

SEARCH: TR 113

Search path: UG 26; LR 69
 configurer: TR 30
 conventions: TR 294

search.match: LR 114

search.no.match: LR 114

Secondary operations: UG 246

Select process: TR 102

Select source file: TR 93

Selective linking: TR 196

Selective loading, libraries: UG 55; TR 178

Separate compilation: UG 52, 298; TR 10

Separate vector space: UG 50, 251; TR 6, 16

SEQ: TR 9

Sequential programming: UG 4

Serial links: UG 1

Server: UG 18, 299
 command line: LR 76, 77
 termination: LR 79

Session manager: UG 299; TR 252, 256
 configuration file: TR 254

SET: UG 69, 264

seterr: UG 156, 260

SHARED pragma: TR 19

Show debugging messages: TR 105

Simulator: UG 21; TR 271
 batch command files: TR 281
 batch commands: TR 281
 batch mode: TR 281
 booting program: TR 279

command
 definitions: *TR 275–281*
 summary: *TR 275*
 command line: *TR 271*
 commands: *TR 274*
 errors: *TR 282*
 options: *TR 272*
 starting a program: *TR 277*
 use in debugging: *UG 146*
SIN: *LR 32, 51*
 Single step execution: *UG 147*
SINH: *LR 41, 60*
SKIP: *UG 268; TR 32*
 Skip load
 example: *UG 107*
 in debugging: *UG 123*
 Skip loader: *UG 18; TR 285*
 command line: *TR 286*
 command line options: *TR 286*
 errors: *TR 289*
snglmath.lib: *LR 3, 20, 26*
so: *LR 98*
so.ask: *LR 84*
so.buffer: *UG 113; LR 96*
so.close: *LR 70*
so.commandline: *LR 76*
so.core: *LR 79*
so.date.to.ascii: *LR 94*
so.eof: *LR 73*
so.exit: *UG 111; LR 79*
so.ferror: *LR 74*
so.flush: *LR 72*
so.fwrite.char: *LR 90*
so.fwrite.hex.int: *LR 91*
so.fwrite.hex.int32: *LR 92*
so.fwrite.hex.int64: *LR 92*
so.fwrite.int: *LR 91*
so.fwrite.int32: *LR 91*
so.fwrite.int64: *LR 91*
so.fwrite.nl: *LR 90*
so.fwrite.real32: *LR 92*
so.fwrite.real64: *LR 92*
so.fwrite.string: *LR 90*
so.fwrite.string.nl: *LR 90*
so.getenv: *LR 78*
so.getkey: *LR 83*
so.gets: *LR 71*
so.keystream.from.file: *LR 100*
so.keystream.from.kbd: *LR 100*
so.keystream.from.stdin: *LR 100*
so.multiplexor: *UG 112; LR 96*
so.open: *LR 68*
so.open.temp: *LR 69*
so.overlapped.buffer: *UG 114; LR 96*
so.overlapped.multiplexor: *UG 112; LR 97*
so.overlapped.pri.multiplexor: *LR 97*
so.parse.command.line: *LR 77*
so.pollkey: *LR 83*
so.popen.read: *LR 69*
so.pri.multiplexor: *LR 97*
so.puts: *LR 72*
so.read: *LR 70*
so.read.echo.any.int: *LR 85*
so.read.echo.hex.int: *LR 85*
so.read.echo.hex.int32: *LR 85*
so.read.echo.hex.int64: *LR 85*
so.read.echo.int: *LR 84, 85*

so.read.echo.int32: *LR 84*
so.read.echo.int64: *LR 85*
so.read.echo.line: *LR 84*
so.read.echo.real32: *LR 86*
so.read.echo.real64: *LR 86*
so.read.line: *LR 83*
so.remove: *LR 74*
so.rename: *LR 75*
so.scrstream.to.ANSI: *LR 103*
so.scrstream.to.file: *LR 101*
so.scrstream.to.stdout: *LR 102*
so.scrstream.to.TVI920: *LR 103*
so.seek: *LR 73*
so.system: *LR 79*
so.tell: *LR 73*
so.test.exists: *LR 75*
so.time: *LR 93*
so.time.to.ascii: *LR 94*
so.time.to.date: *LR 94*
so.today.ascii: *LR 95*
so.today.date: *LR 94*
so.version: *LR 80*
so.write: *LR 71*
so.write.char: *LR 87*
so.write.hex.int: *LR 88*
so.write.hex.int32: *LR 88*
so.write.hex.int64: *LR 88*
so.write.int: *LR 87*
so.write.int32: *LR 87*
so.write.int64: *LR 87*
so.write.nl: *LR 87*
so.write.real32: *LR 88*
so.write.real64: *LR 88*
so.write.string: *LR 87*
so.write.string.nl: *LR 87*
 Soft channels: *UG 85, 299*
 Software, virtual routing: *UG 86*
 Software description: *UG 81, 267*
 example: *UG 82*
sortconf.pgm: *UG 93*
sortsw.inc: *UG 93*
 Source level debugging: *UG 129*
SP: *UG 111*
srecord. *See output.format*
ss: *LR 99*
ss.beep: *LR 109*
ss.clear.eol: *LR 108*
ss.clear.eos: *LR 108*
ss.del.line: *LR 110*
ss.delete.chl: *LR 109*
ss.delete.chr: *LR 109*
ss.down: *LR 109*
ss.goto.xy: *LR 108*
ss.ins.line: *LR 110*
ss.insert.char: *LR 109*
ss.left: *LR 109*
ss.right: *LR 109*
ss.scrstream.copy: *LR 103*
ss.scrstream.fan.out: *LR 103*
ss.scrstream.from.array: *LR 102*
ss.scrstream.multiplexor: *LR 104*
ss.scrstream.sink: *LR 101*
ss.scrstream.to.array: *LR 102*
ss.up: *LR 109*
ss.write.char: *LR 106*

ss.write.endstream: LR 107
ss.write.hex.int: LR 107
ss.write.hex.int64: LR 108
ss.write.int: LR 107
ss.write.int64: LR 107
ss.write.nl : LR 107
ss.write.real32: LR 108
ss.write.real64: LR 108
ss.write.string: LR 107
ss.write.text.line: LR 107
Stack: UG 225; PI 6
 See also Workspace
 overflow detection: UG 225
 placing in on-chip RAM: UG 225
 position in memory: TR 53
stack.buffer: TR 56
Standard error: UG 110, 299
Standard input: UG 110, 299
Standard memory configuration:
 TR 153
Standard output: UG 110, 299
Standards
 command options: TR 293
 command syntax: TR 293
 file extensions: TR 295
start.offset: TR 168
Static area: UG 213
 pointer: UG 213
 position in memory: TR 53
 requirement: UG 213
Static data: TR 54
Static variables, memory map: TR
 239
STOP: UG 268
STOP error mode: UG 45; TR 8,
 31; PI 5
 debugging: UG 118
stopp: UG 260
str.shift: LR 115
Streamio library: UG 110; LR 98
streamio.inc: UG 110
streamio.lib: UG 110; LR 3
Streams: LR 66
String handling
 comparison: LR 113
 editing: LR 115
 library: LR 111
 introduction: LR 3
 searching: LR 114
string.pos: LR 114
STRINGTOBOOL: LR 125
STRINGTOHEX: LR 124
STRINGTOHEX16: LR 124
STRINGTOHEX32: LR 124
STRINGTOHEX64: LR 124
STRINGTOINT: LR 123
STRINGTOINT16: LR 123
STRINGTOINT32: LR 123
STRINGTOINT64: LR 123
STRINGTOREAL32: LR 124
STRINGTOREAL64: LR 125
Subsystem: UG 105, 299
 connecting: TR 82
 reset: TR 286
 wiring: UG 105; TR 287
Sun, host types: LR 80
Sun 4: UG 7, 25
SunOS: UG 7, 25; LR 81
Suspending programs: UG 45
Symbol data, listing: TR 209
Symbolic debugging: UG 129; TR
 108
 See also Debugging
 compiling for: UG 117
Synchronized communication: UG
 4
System call: LR 79
System services: UG 105

T

T-mode programs: TR 74
T4 series
 configurable memory interface:
 TR 145
 optimized library: LR 20, 45
T8 series
 configurable memory interface:
 TR 145
 maths libraries: LR 26
Tabs, in occam source: LR 140
TAN: LR 35, 54
TANH: LR 43, 62
Target files, for imakef: TR 222
Target transputer: UG 8, 299; TR
 301; PI 1, 4
 command line options: TR 308
TB: LR 45
tbmaths.lib: LR 3, 20, 45
TCOFF: UG 7, 17; TR 4
 listing files: TR 218
TDS: LR 98
Temporaries, elimination of: PI 17
Terminate
 on error: TR 255
 the server: TR 255
terminate.heap.use: UG 214
terminate.static.use: UG
 214
Text
 reading: LR 71
 stream: LR 68
 writing: LR 72
Text files, listing: TR 218
Time: LR 93, 94
 See also Clock; Date
 transputer clock: LR 155
Timeout: UG 256; LR 128
 channel input: UG 101; LR 128
 channel output: UG 101; LR 129
 on links: UG 257
TIMER: UG 276
 channels: LR 155
 parameters: UG 203
Timer: LR 153
 See also Clock
Timer queue: UG 135, 140
 displaying: TR 105, 280
TIMES: PI 25
Timing data: TR 154
Tm: TR 152
to.lower.case: LR 116
to.upper.case: LR 116
TOGGLE BREAK: TR 111
TOGGLE HEX: TR 114
tolerance: UG 75, 91, 189, 266
Toolset
 development cycle: UG 13
 documentation: UG xviii; TR xx;
 LR vii, viii; PI v
 conventions: UG xix; TR xxi;
 LR ix; PI vii
 file extensions: UG 22
 program development: UG 13
 standards and conventions: TR
 293
 summary: UG 12
TOP: TR 88, 107, 112
TOP OF FILE: TR 89, 113
Tptr0: UG 134
Tptr1: UG 134
Traceback information, in ROM:
 TR 170
TRAM: UG 106, 241, 300; TR 285
TRANSLATE pragma: TR 14
TRANSPUTER: UG 27, 33, 104; TR
 254, 255, 260
Transputer
 accessing: TR 252
 on a remote host: TR 261

on the local host: *TR 261*
 architecture: *UG 2*
 classes: *TR 7*
 clock: *UG 134, 136*
 in real-time programming: *UG 3*
 instruction set: *UG 281*
 introduction: *UG 1*
 loading: *UG 103*
 master: *UG 105*
 module: *UG 300*
 networks: *UG 3*
 operation codes: *UG 282*
 products: *UG 4*
 root: *UG 298*
 scheduler: *PI 30*
 simulator: *TR 271*
 targets: *UG 299; TR 4, 7, 301;*
PI 1, 4
 command line options: *TR 308*
 timer: *UG 134; LR 153*

Transputer Development System:
LR 98

Tree, network topology: *UG 3*

Trigonometric functions: *LR 27*

TRUE: *PI 28*

TVI920: *LR 103*

type: *UG 74, 265*

Type conversion library: *LR 119*

Type mapping: *LR 151*

U

UART: *UG 240*

UDPlink: *LR 81*

UNDEFINED error mode: *UG 46,*
47; TR 9, 17, 31; PI 5

undefined.NaN: *LR 21*

UNIVERSAL error mode: *UG 46;*
TR 8, 31; PI 5
 debugging: *UG 118*

UNIX: *UG 25; TR 293*

UNPACKSN: *LR 13*

Unresolved references: *TR 196*

unstable.NaN: *LR 21*

Unsupported options: *UG 31; TR*
294

Up, subsystem wiring: *UG 105*

Update registers: *TR 105*

Upper case: *LR 113, 116*

Usage check: *UG 48, 300; TR 6,*
16
 disable: *TR 19*

Usage checking: *LR 161; PI 7*
 arrays: *LR 162*
 channels: *LR 162*
 disable: *PI 7*
 disabled: *LR 164*

Usage files, libraries: *UG 296*

User link: *UG 300; TR 252, 259*

V

VAL: *UG 69, 276*

Variable

non-local, access to: *PI 27*
 place at address: *LR 142*
 place in memory: *LR 143*
 place in workspace: *UG 242*
 scoping: *PI 26*
 unused - elimination of: *PI 17*

VAX/VMS: *UG 7, 25, 26, 27, 111*

VECSPACE: *UG 51; LR 139, 143*

Vector space: *UG 50, 51, 300; TR*
18, 59; LR 147, 150; PI 5, 22
 default: *TR 34*
 disabling: *UG 50*
 disadvantage: *PI 31*
 in mixed language programming:
UG 221
 position in memory: *UG 181; TR*
55

Virtual channel: *UG 85*
 disable: *UG 87*

Virtual link: *UG 85, 152*

Virtual memory: *TR 195*

Virtual routing: *UG 86; PI 5*
 controlling: *UG 91*
 disable: *UG 48, 87*
 disabling: *TR 10, 32*
 introduction: *UG 10*
 optimization techniques: *UG 187*
 software: *UG 86*

VME bus, motherboard: *UG 105*

VMS: *UG 25, 27; TR 293, 342;*
LR 81

VSSIZEOF: *LR 15, 17*

W

Wait

connection: *TR 153*
 race: *TR 153*
 error: *TR 159*
 states: *TR 153*

Warnings. See Error messages

Waveform diagrams: *TR 156*

Wdesc: *UG 134*

WdescIntSave: *UG 134*

Wired down: *UG 105; TR 82, 287*

Wired subs: *UG 105; TR 82, 287*

Word alignment, placed objects:
UG 240

Word length, independence: *UG*
240

WORKSPACE: *UG 51, 241; LR 139,*
143

Workspace: *UG 300; TR 18; LR*
147, 148; PI 6

See also Stack

allocation: *PI 17, 19*

default: *TR 34*

in ASM code: *UG 247*

in dynamic loading: *UG 251*

in mixed language programming:
UG 221

position in memory: *UG 181*

Worm: *UG 300*

Wreg: *UG 281*

Write

mode: *TR 153*

strobe: *TR 152*

to memory, in idebug: *TR 106*

WSSIZEOF: *LR 15, 17*

X

xlink.lib: *UG 256; LR 3, 128*

Z

z, command line option: *UG 31;*
TR 294

