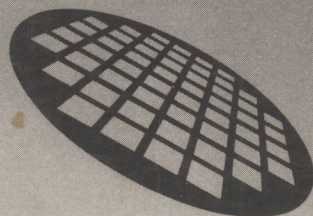
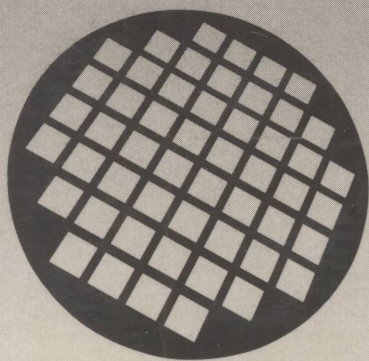




TRANSPUTER

DEVELOPMENT

SYSTEM



TP3
T772

8963662

TRANSPUTER DEVELOPMENT SYSTEM

INMOS Limited



E8963662



Prentice Hall

New York London Toronto Sydney Tokyo

8983883

First published 1988 by
Prentice Hall International (UK) Ltd,
66 Wood Lane End, Hemel Hempstead,
Hertfordshire, HP2 4RG
A division of
Simon & Schuster International Group



© 1988 INMOS Limited

INMOS reserves the right to make changes in specifications at any time and without notice. The information furnished by INMOS in this publication is believed to be accurate, however no responsibility is assumed for its use, nor for any infringement of patents or other rights of third parties resulting from its use. No licence is granted under any patents, trademarks or other rights of INMOS.

●, **inmos**, **IMS** and **OCCAM** are trademarks of the INMOS Group of Companies.

INMOS document number: 72 TRN 011 00

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission, in writing, from the publisher. For permission within the United States of America contact Prentice Hall Inc., Englewood Cliffs, NJ 07632.

Printed and bound in Great Britain
at the University Press, Cambridge

CIP data are available

2 3 4 5 92 91 90 89

ISBN 0-13-928995-X

Other titles in this series

TRANSPUTER DEVELOPMENT SYSTEM

Transputer Reference Manual

Transputer Development System

Communicating Process Architecture

Transputer Technical Notes

Transputer Installation Set: a compiler writer's guide

Digital Signal Processing

Other titles in this series

Transputer Reference Manual
Transputer Development System
Communicating Process Architecture
Transputer Technical Notes
Transputer Instruction Set: a compiler writer's guide
Digital Signal Processing

Contents overview

1	<i>How to use the manual</i>	Describes the layout of the manual.
2	<i>Introduction</i>	Introductory explanation of the transputer and the TDS.

The user guide

3	<i>Directories</i>	Describes the directories set up and used by the TDS.
4	<i>The editing environment</i>	Describes the editor and its facilities.
5	<i>Compiling and linking occam programs</i>	Describes how to use the compiler and its associated utilities.
6	<i>Running programs within the TDS</i>	Describes how to prepare and use programs on the TDS.
7	<i>Configuring programs and loading transputer networks</i>	How to prepare programs for and run them on networks.
8	<i>Standalone transputer programs</i>	Describes how to prepare programs to run independently of the TDS.
9	<i>Debugging</i>	An introduction, with a worked example, to the debugger.
10	<i>EPROM programming</i>	Describes how to prepare programs for EPROMs.
11	<i>Low level programming</i>	Describes how to use low level programming facilities.

The reference manual

12	<i>The development environment</i>	Lists and describes the keys and messages.
13	<i>Utilities</i>	Describes in detail all the utilities.
14	<i>Libraries</i>	Describes in detail the functions and procedures of all libraries.
15	<i>Tools</i>	Describes in detail all the software tools e.g. debugger.
16	<i>System interfaces</i>	Describes in detail the interfaces to the system.

The appendices

A	<i>Keyboard layouts</i>	Shows how the keys are mapped to functions.
B	<i>Summary of standard utilities</i>	Shows how the utilities are grouped.
C	<i>Names defined by the software</i>	Lists all the names defined by the software.
D	<i>System constant definitions</i>	Lists the constants used by the system.
E	<i>Error numbers</i>	Lists the error numbers.
F	<i>Fold attributes</i>	Lists the values that the fold attributes can take.
G	<i>File formats</i>	Describes the various file formats.
H	<i>Transputer instruction support</i>	List the transputer instructions that are supported.
I	<i>Bibliography</i>	Lists literature worth referring to.
J	<i>Glossary</i>	A glossary of terms used to describe the features of the TDS.
	THE INDEX	A comprehensive index.

Preface

This manual describes the Transputer Development System, an integrated programming environment developed by INMOS to support the programming of transputer networks in OCCAM. The Transputer Development System comprises an integrated editor, file manager, compiler and debugging system.

The Transputer Development System runs on a transputer board; for example it runs on an INMOS IMS B004 board containing an IMS T414 32-bit processor and 2 MBytes of memory. This board is installed inside an IBM PC/AT or similar computer, which provides a means of interfacing keyboard, screen and disks to the transputer.

The Transputer Development System allows OCCAM programs to be written, compiled and then run from within the development system. Programs may also be configured to run on a target network of transputers; these may range from a single transputer on an evaluation board to networks of several hundred transputers. The code for a transputer network may be loaded directly from the Transputer Development System, through a link connecting the Transputer Development System transputer to the target network. Programs may also be placed into a file separate from the Transputer Development System, or into a ROM (Read-Only Memory), and used to load a network.

A post-mortem debugger allows programs running in the Transputer Development System environment or on a transputer network to be examined after they have been interrupted or have stopped as a result of an error. The line of source corresponding to a program error on one of the processors can be displayed, and the values of variables may be examined. The state of other currently active processes on this processor, and on other processors in the network, can also be examined.

The Transputer Development System software includes the interactive programming environment, the compilation utilities and other programming tools, a number of libraries to support program development (such as mathematical functions and I/O libraries), and an extensive set of examples in source form.

This manual is divided into two major parts: the *User Guide*, which introduces the system and takes the reader through the steps needed to write, compile and run programs, and the *Reference Manual*, which contains detailed reference information on the editor, utilities, tools, libraries and system interfaces.

The instructions on installing the software and a detailed list of the components of the release are contained in a separate *Delivery manual*, supplied with the software.

This manual corresponds to the IMS D700D (IBM PC) and IMS D800D (NEC PC) releases of the Transputer Development System.

Contents

Contents	v
Contents overview	xvii
Preface	xix
1 How to use the manual	1
1.1 Introduction	1
1.2 User guide	1
1.3 Reference manual	1
1.4 Appendices	1
2 Introduction	3
2.1 Overview	3
Transputers and occam	3
2.2 System design rationale	4
2.2.1 Programming	5
2.2.2 Hardware	5
2.2.3 Programmable components	5
2.3 occam model	5
2.4 Program development	7
2.4.1 Logical behaviour	7
2.4.2 Performance measurement	7
2.4.3 The transputer development system	7
The user guide	9
3 Directories	11
4 The editing environment	13
4.1 Introduction	13
4.1.1 Folding	13
4.1.2 Files as folds	15
4.2 Starting and finishing the system for the first time	15
4.2.1 Starting the system	15
4.2.2 The TDS2 command	16
4.2.3 Problems starting the system	16
4.2.4 Keyboard layout	17
4.2.5 Repainting the screen	17
4.2.6 Ending the session	17
4.2.7 Interrupting and rebooting the TDS	17
4.2.8 Suspending the TDS	17
4.3 Tutorial file	18
4.4 The editor interface	18
4.4.1 Editor's view of a document	18
4.4.2 The screen display	18
4.4.3 Line types	19
4.5 Editor functions	20
4.5.1 Overview of editor functions	20
4.5.2 Editor modes	21
4.5.3 Moving the cursor	21

4.5.4	Scrolling the screen	21
4.5.5	Fold browsing operations	22
	Opening and closing folds	22
	Fold information	22
	Browsing mode	22
4.5.6	Inserting and deleting characters	23
	Insertion	23
	Deletion	23
4.5.7	Fold creation and removal	23
4.5.8	Filed folds	24
	Storage of files in memory	25
	File extensions	25
	Writing back files	25
4.5.9	Deleting lines	26
4.5.10	Moving and copying lines	26
4.5.11	Defining a keystroke macro	27
4.6	Utilities and programs	27
4.6.1	The toolkit fold	28
4.6.2	Loading utilities and programs	29
4.6.3	Loading code from the toolkit fold	29
4.6.4	Running a utility	30
4.6.5	Supplying parameters to utilities	30
4.6.6	When a utility finishes	31
4.6.7	Running executable programs	31
4.7	File handling utilities	32
4.8	Searching and replacing	33
4.9	Listing programs	33
4.9.1	The lister and unlist programs	34
4.10	Transferring TDS files between computers	34
5	Compiling and linking occam programs	35
5.1	Introduction	35
5.2	The compiler utility set	35
5.3	Preparing a program for compilation	36
5.3.1	Creating a compilation fold	36
5.3.2	Comment folds	37
5.4	Using the compiler utilities	37
5.4.1	Checking occam programs	37
5.4.2	Compiling occam programs	37
5.4.3	Linking occam programs	38
5.5	Compiling a simple example program	39
5.5.1	Getting the compiler utilities	39
5.5.2	Making an EXE fold	40
5.5.3	Checking and compiling the example program	40
5.5.4	Running the example program	41
5.5.5	Compilation information	41
5.6	Separate compilation and libraries	42
5.6.1	Separate compilation	42
5.6.2	Libraries	43
5.6.3	Compiling and linking large programs	43
5.6.4	Changing and recompiling programs	44
5.6.5	The implementation of change control	44
5.7	Compiler parameters	45

5.7.1	The parameter fold	45
5.7.2	Error modes of compilation	45
5.7.3	Alias and usage checking	46
5.7.4	Using the separate vector space	47
5.8	Creating and using libraries	48
5.8.1	Creating libraries	48
5.8.2	Using libraries	49
5.8.3	Using protocols with separate compilation	50
5.8.4	How the library system works	50
5.8.5	The library logical names fold	51
5.8.6	Library compaction	52
5.9	Changing and recompiling libraries	53
5.9.1	Change control	53
5.9.2	Library dependencies	53
5.9.3	Recompiling mixed libraries	53
5.9.4	Compacting recompiled libraries	54
5.10	The pipeline sorter example	54
5.10.1	The 'header.tsr' library fold	56
5.10.2	The 'problem.tsr' library fold	57
5.10.3	The 'monitor.tsr' library fold	57
	The keyboard handler	58
	The screen handler	59
5.11	The implementation of occam	59
5.11.1	The transputer implementation of occam	60
5.11.2	Memory allocation by the compiler	61
5.11.3	Implementation of usage checking	62
	Usage rules	62
	Checking of non-array elements	62
	Checking of arrays of variables and channels	62
	Arrays as procedure parameters	63
	Abbreviating variables and channels	63
	Problems with replicators	64
6	Running programs within the TDS	65
6.1	Loading and running an executable program	65
6.2	The interface for user programs	65
6.3	The channel parameters and their protocols	67
6.3.1	The keyboard and screen	67
6.3.2	Communicating with the user filer	67
6.3.3	The fold manager	68
6.3.4	Communicating with the filer	68
6.3.5	The kernel channels	68
6.4	Memory usage within the TDS	69
6.5	The occam input/output procedures	70
6.5.1	The input/output models	71
6.5.2	TDS terminal and file I/O	71
6.5.3	Reading and writing a DOS file	71
6.6	The pipeline sorter example	72
6.7	Example programs using the I/O libraries	73
6.7.1	Keyboard and screen example	74
6.7.2	Example showing input from file	75

7	Configuring programs and loading transputer networks	79
7.1	Introduction	79
7.2	The transputer configuration and loading utilities	79
7.3	The configuration description	79
7.4	Configuring a program	82
7.5	Connecting a network to the TDS	82
7.6	Loading a network	83
7.7	Using the transputer network tester	83
7.8	Running the pipeline sorter on a target transputer	84
7.8.1	Creating a PROGRAM fold	84
7.8.2	Monitoring the target with an EXE	85
7.8.3	Configuring and running the example	86
7.9	Running the pipeline sorter on a four transputer network	87
7.9.1	A PROGRAM for four transputers	87
7.9.2	The root transputer	88
7.9.3	The three other transputers	89
7.9.4	Configuration for four transputers	89
8	Standalone transputer programs	91
8.1	Introduction	91
8.2	Using the host file server	91
8.3	Creating a standalone program	92
8.4	The pipeline sorter	92
9	Debugging	97
9.1	Using the debugger	97
9.2	Debugger facilities	97
9.2.1	Symbolic facilities	98
9.2.2	Lower level facilities	98
9.3	Debugging a program running on a network of transputers	99
9.4	Debugging a program running within the TDS	99
9.5	Debugging a standalone program	99
9.6	A worked example	100
9.6.1	Running the example program	102
9.6.2	Creating a core dump	102
9.6.3	Using the debugger	102
9.6.4	Inspecting variables	103
9.6.5	Jumping down channels	103
9.6.6	Retrace and Backtrace	104
9.6.7	Process Queues	104
9.6.8	Display occam	104
9.6.9	Read/Write and Finish	105
9.6.10	Other options	105
9.6.11	More information	105
9.7	How the debugger works	106
9.7.1	How the debugger accesses the network	106
9.7.2	Debugging information generated by the compiler	106
9.7.3	How the symbolic facilities work	107
9.7.4	Backtracing	107
9.7.5	Inspecting variables	107
9.7.6	Jumping down channels	107
9.7.7	Analysis of deadlock	107

9.7.8	occam scope rules	109
10	EPROM programming	111
10.1	Introduction	111
10.2	How to create the fold bundle	112
10.3	Creating the ROM file	113
10.4	Burning the ROM	113
10.5	Execution from ROM instead of RAM	114
10.6	ROMs which load from a host computer	115
10.7	Adding a memory configuration to the EPROM	116
11	Low level programming	117
11.1	Allocation	117
11.2	Code insertion	119
11.2.1	Using the code insertion mechanism	119
11.2.2	Labels and jumps	120
11.3	Dynamic code loading	120
11.3.1	The call	121
11.3.2	Loading parameters	122
11.3.3	Examples	122
11.3.4	Code format	124
	Extracted code format	124
11.4	Extraordinary use of links	125
11.4.1	Clarification of requirements	125
11.4.2	Programming concerns	125
11.4.3	Input and output procedures	126
11.4.4	Recovery from failure	126
11.4.5	Example: a development system	127
	The reference manual	129
12	The development environment	131
12.1	Keys	131
12.2	Messages	140
12.2.1	Development environment messages	140
12.2.2	TDS server termination messages	144
13	Utilities	147
13.1	occam program development package	147
13.1.1	CHECK	147
13.1.2	COMPILATION INFO	147
	Compilation information	148
	Configuration information	148
	Library information	149
13.1.3	COMPILE	149
13.1.4	Compiler messages	152
13.1.5	Library logical name fold errors	152
13.1.6	Program errors	153
13.1.7	Implementation limits	153
13.1.8	Compiler errors	154
13.1.9	Configurer error messages	154
13.1.10	EXTRACT	156

13.1.11	LOAD NETWORK	159
13.1.12	MAKE COMMENT	160
13.1.13	MAKE FOLDSET	161
13.1.14	RECOMPILE	161
13.1.15	REPLACE	162
13.1.16	SEARCH	163
13.2	File handling package	165
13.2.1	ATTACH	165
13.2.2	COMPACT LIBRARIES	166
13.2.3	COPY ATTACH	166
13.2.4	COPY IN	167
13.2.5	COPY OUT	168
13.2.6	DETACH	169
13.2.7	READ HOST	170
13.2.8	RENAME FILE	171
13.2.9	WRITE ENABLE	172
13.2.10	WRITE HOST	172
13.2.11	WRITE PROTECT	173
14	Libraries	175
14.1	Multiple length integer arithmetic functions	177
14.2	Floating point functions	178
14.3	IEEE arithmetic functions	179
14.4	2D block move library	179
	MOVE2D	179
	DRAW2D	180
	CLIP2D	180
14.5	Bit manipulation and CRC library	180
	BITCOUNT	180
	CRCWORD	180
	CRCBYTE	181
	BITREVNBITS	181
	BITREVWORD	181
14.6	Code execution	181
14.7	Arithmetic instruction library	183
14.8	Single length and double length elementary function library	184
14.8.1	Introduction	186
14.8.2	Inputs and Outputs	186
14.8.3	Accuracy	187
	Range Reduction	187
	Generated Error	187
	Propagated Error	188
	Test Procedures	188
14.8.4	Symmetry	188
14.8.5	The Function Specifications	189
	Terms used in the Specifications	189
	Specification of Ranges	190
	Abbreviations	190
14.8.6	ALOG	190

14.8.7	ALOG10	191
14.8.8	EXP	192
14.8.9	POWER	193
14.8.10	SIN	194
14.8.11	COS	195
14.8.12	TAN	197
14.8.13	ASIN	198
14.8.14	ACOS	199
14.8.15	ATAN	199
14.8.16	ATAN2	200
14.8.17	SINH	201
14.8.18	COSH	202
14.8.19	TANH	203
14.8.20	RAN	203
14.9	IMS T414 elementary function library	204
14.9.1	ALOG	205
14.9.2	ALOG10	206
14.9.3	EXP	207
14.9.4	POWER	208
14.9.5	SIN	209
14.9.6	COS	210
14.9.7	TAN	212
14.9.8	ASIN	213
14.9.9	ACOS	214
14.9.10	ATAN	214
14.9.11	ATAN2	215
14.9.12	SINH	216
14.9.13	COSH	217
14.9.14	TANH	218
14.9.15	RAN	218
14.10	Basic type i/o conversion library ioconv	219
14.11	Extra type i/o conversion library extrio	220
14.12	String handling library strings	222
14.12.1	Character handling functions	223
14.12.2	String comparison functions	224
14.12.3	String editing procedures	225
14.12.4	String searching functions	226
14.12.5	String append functions	226
14.13	General purpose i/o procedure library userio	228
14.13.1	The simple input and output procedures	232
	Simple output procedures	233
	Simple input procedures	233
	Output to and input from the folded file store	234
14.13.2	Simple output	235
14.13.3	Control codes to the terminal screen	237
14.13.4	Simple input	238
14.13.5	Long integers and reals	240
14.13.6	Write folded stream	242
14.13.7	Read folded stream	243
14.14	Interface procedure library interf	249
14.15	Block transfer procedure library slice	254
14.16	Low level user filer interface support library ufiler	255
14.17	TDS server channel support library msdos	258

14.18	Byte stream i/o library <code>derivio</code>	260
14.19	Afserver low level protocol library <code>afio</code>	264
14.20	Afserver command library <code>afiler</code>	265
14.21	Afserver protocol interface and multiplexor <code>afinterf</code>	268
14.22	Transputer board support library <code>t4board</code>	269
14.23	IMS B006 support library <code>t2board</code>	271
14.24	Extraordinary link handling library <code>reinit</code>	271
14.25	Block CRC library <code>blockcrc</code>	273
15	Tools	275
15.1	Debugger	275
15.1.1	Debugging a PROGRAM	275
	What the debugger does	277
15.1.2	Debugging an EXE	277
	Start up procedure for an EXE	277
15.1.3	Debugging an SC	278
	Start up procedure for an SC	278
15.1.4	Symbolic facilities	278
	Debugging an SC	282
	Invalid Wdesc	282
15.1.5	Monitor page	282
15.1.6	Monitor page options	284
15.1.7	Hints	292
	Invalid pointers	292
	Failure to communicate	292
	Default addresses	292
	IF and CASE	292
	ALT	292
	CASE input	293
	Deadlocks	293
15.1.8	Creating a core dump file	294
15.1.9	OCCAM run time errors	295
15.2	Transputer network tester	297
15.2.1	What the network tester does	297
15.2.2	Using the network test program	298
	A note on matching	298
	Limitations of use	298
15.2.3	Options available	299
15.2.4	Interpretation of loading data	300
15.2.5	Description of network	301
15.2.6	Error messages	302
15.2.7	Testing specifications	304
15.2.8	Stages of loading	306
15.3	Memory interface program	307
15.3.1	Capabilities	307
15.3.2	Using the program	307
15.3.3	Input	308
15.3.4	Output	309
	Numeric output	309
	Waveform output	310
15.3.5	Storing and retrieving parameters and pages	310
15.3.6	Examples	311
15.3.7	Caveats	313

15.3.8	Error and warning messages	314
15.4	EPROM hex program	314
15.4.1	Using the program	314
15.4.2	What the EPROM hex program does	315
	Error messages	318
15.5	Hex to programmer program	319
15.5.1	Using the program	319
15.5.2	Adapting the Hex to programmer program	320
16	System interfaces	323
16.1	Terminal interfaces	323
16.1.1	Input from the keyboard channel	323
16.1.2	Screen stream protocol	324
	Outputting characters to the screen	326
	Cursor movement	326
	Clearing the screen	326
	Character operations	326
	Line operations	327
	Other operations	327
	Initialising	327
	Changing the way keyboard input is processed	328
	Termination, claim and release	328
16.2	User filer interfaces	329
16.2.1	User filer protocol	329
16.2.2	Selecting a fold for access	330
16.2.3	User filer channels	330
16.2.4	User filer modes	331
16.2.5	Commands in user filer command mode	332
	Definitions of <code>uf.</code> commands	332
	Example showing use of a <code>uf.</code> command	336
	Opening a fold for reading	336
	Opening a fold for writing	337
16.2.6	Communications in file stream modes	338
	Introduction to file stream modes	338
	Syntax of valid sequences of communications	338
	Data stream modes	339
	Folded stream modes	340
	Reading a fold stream from the system sender	344
	Writing a fold stream to the system receiver	345
16.3	Host file server	347
16.3.1	Afserver command syntax	347
16.3.2	Afserver command line options	348
16.3.3	Afserver protocol	349
	Introduction to the afserver protocol	349
	Overview of afserver operations	351
	Server termination operations	352
	Stream opening operations	353
	Server information operations	355
	Operating system support operations	358
	Stream information operations	361
	Stream reading and writing operations	363
	Low level DOS operations	365
16.3.4	Afserver error messages	369

	16.3.5	Summary of afserver protocol	371
		Basic protocol	371
		Parameter and result types	371
		Operations protocol	372
16.4		TDS file server	375
	16.4.1	Server description	375
	16.4.2	The SERVER command	376
	16.4.3	Server overview	377
	16.4.4	Communicating with the server	378
		Channel multiplexing	378
		Sequence of communications	381
	16.4.5	The terminal channels	382
		Protocol to the terminal	382
		Protocol from the terminal	385
		Differences in the server and TDS terminals	387
	16.4.6	The filer channels	387
	16.4.7	The kernel channels	415
	16.4.8	The serial channels	417
	16.4.9	Summary of TDS server protocol	418
		Appendices	425
A		Keyboard layouts	427
	A.1	IBM PC function keys	427
	A.2	IBM PC keyboard layout	428
	A.3	NEC PC keyboard layout	430
B		Summary of standard utilities	433
C		Names defined by the software	435
D		System constant definitions	447
	D.1	MATHVALS	447
	D.2	USERHDR	448
	D.3	FILERHDR	450
	D.4	KRNLHDR	454
	D.5	USERVALS	458
	D.6	AFHDR	458
E		Error numbers	461
	E.1	File server errors	461
	E.2	DOS errors	462
	E.3	Filer errors	462
	E.4	File streamer errors	463
F		Fold attributes	465
	F.1	Fold attributes in the TDS	465
		F.1.1 Fold type	465
		F.1.2 Fold contents	465
		F.1.3 Fold indent	466
	F.2	Attribute constant values	466
	F.3	Attributes of common fold types	467