Contents

About This Manual ix
Customer Support xi

Chapter 1 Introduction 1

1.1 Developing a C* Program 1
1.2 Performing I/O in a C* Program 1
1.3 Compiling a C* Program 2
1.4 Executing a C* Program 2
1.5 Debugging a C* Program 2

Chapter 2 Developing a C* Program 3

2.1 C* `.cs` Files 3
2.1.1 C* Keywords 4
2.1.2 Reserved Identifiers 4
2.2 Header Files 4
2.2.1 The `<math.h>` File 5
2.2.2 The `<stdlib.h>` File 6
2.2.3 The `<string.h>` File 6
2.2.4 Header Files and C* Keywords 7
2.3 Timing a Program 7
2.3.1 Hints on Using the Timing Utility 9
2.4 Calling CM Libraries 10
2.4.1 CMX11 11
2.4.2 CM/AVS 11
2.4.3 CMMD 11
2.5 Calling PN Functions 11
2.6 Calling CM Fortran 13
2.6.1 Overview 13
2.6.2 In Detail 13
Include File 13
Calling the Subroutine 14

Copyright © 1990–1995 Thinking Machines Corporation

What Kinds of Variables Can You Pass? 14
Passing Parallel Variables 15
Passing Scalar Variables 15
Return Type 16
Freeing the Descriptors 16
An Example 16
2.7 Writing Functions that Are Callable from CM Fortran 17
2.7.1 Names 17
2.7.2 Shapes 18
2.7.3 Return Type 18
2.7.4 Parallel Arguments 18
2.7.5 Scalar Arguments 19
2.7.6 An Example 21

Chapter 3 Performing I/O 25
3.1 Which Method to Use 25
3.2 Using Parallel Versions of `fread` and `fwrite` 26
3.2.1 Performance 29
Parallel Files 29
Parallel Sockets 30
I/O Involving the Partition Manager 30
3.2.2 Limitations 31
3.2.3 Examples 31
Parallel Reads and Writes 31
Socket I/O 33
3.3 Creating and Using an NMO Buffer 40
3.3.1 Allocating and Freeing an NMO Buffer 41
3.3.2 Writing to and Reading from an NMO Buffer 42
Parallel Data 42
Scalar Data 44
3.3.3 Examples 45
3.4 Using the CMFS Library 49
3.5 Performing Physical I/O 52

Chapter 4 Compiling a C* Program 55
4.1 The Compilation Process 55
4.2 Basic Options 58
4.2.1 Choosing the Compiler: The `-cm2`, `-cm200`, `-cmsim`, and `-cm5` Options 58

Version 7.2, May 1995
Copyright © 1990–1995 Thinking Machines Corporation

4.2.2 Getting Help: The **-help** Option 59
4.2.3 Printing the Version Number: The **-version** Option 60
4.2.4 Compiling for CM-5s with or without Vector Units: The **-vu**, **-vecunit**, and **-sparc** Options 60
4.3 Options in Common with **cc**: The **-c**, **-D**, **-E**, **-g**, **-I**, **-l**, **-L**, **-o**, **-P**, and **-U** Options 60
4.4 Advanced Options 61
4.4.1 Using Another C Compiler: The **-cc** Option 61
4.4.2 Debugging Your Program: The **-cmdebug** Option 61
4.4.3 Obtaining Performance Data: The **-cmprofile** Option 61
4.4.4 Displaying the **bin**, **lib**, **include**, and **temp** Directories: The **-dirs** Option 62
4.4.5 Displaying Compilation Steps: The **-dryrun** Option 62
4.4.6 Putting .c Files through the C* Compiler: The **-force** Option 63
4.4.7 Keeping an Intermediate File: The **-keep** Option 63
4.4.8 Simplifying Linking with Other CM Software Libraries: The **+l** Option 63
4.4.9 Compiling Local Functions: The **-local** Option 65
4.4.10 Don't Link with ANSI Library: The **-noansilibs** Option . . . 65
4.4.11 Using CMMD: The **-node** and **-cmmd_root** Options 66
4.4.12 Don't Link with the **printf** Library with 64-bit **int** Support: The **-noprintflibs** Option 66
4.4.13 Displaying Names of Overloaded Functions: The **-overload** Function 66
4.4.14 Displaying Warnings for Noncompliant Code: The **-pedantic** Option 67
4.4.15 Creating Assembly Source Files: The **-S** Option 67
4.4.16 Changing the Location of Temporary Files: The **-temp** Option 67
4.4.17 Turning On Verbose Compilation: The **-v** or **-verbose** Option 67
4.4.18 Turning Off Warnings: The **-warn** Option 67
4.4.19 Turning On Warnings about Undeclared Functions: The **-Wimplicit** Option 68
4.4.20 Specifying Options for Other Components: The **-Z** Option 68
4.5 Linking 68
4.5.1 Names of C* Run-Time Libraries 68
4.5.2 Intermediate Files 69
4.5.3 Creating Libraries 70

Copyright © 1990–1995 Thinking Machines Corporation

4.6 Compiling and Linking a C* Program that Calls CM Fortran 70
4.7 Compiling CM Fortran Programs that Call C* 72
4.8 Symbols 72
4.9 Using **make** 73

Chapter 5 Executing a C* Program 75

5.1 Executing the Program Directly 75
5.2 Executing in Batch Mode 76
5.2.1 UNIX Batch Commands 76
5.2.2 DJM (Distributed Job Manager) 76
5.2.3 The NQS Batch System 78
Specifying Output Files 78
5.3 Executing a C* Program on a Sun-4 79

Chapter 6 Debugging a C* Program 81

6.1 Compiling for Prism 81
6.2 Starting Prism 81
6.2.1 Graphical Prism 82
6.2.2 Commands-Only Prism 82
6.2.3 Node Prism 83
6.3 Using Prism 83
6.4 Loading and Executing Programs 84
6.5 Debugging 84
6.6 Visualizing Data 85
6.6.1 Visualizing Parallel Objects 87
6.7 Analyzing a Program's Performance 87
6.8 Node Prism 88
6.9 Using Prism with Global/Local Programs 88

Appendix A Man Pages 91

cs 93
bitop.h 100
cscomm.h 102
math.h 106
stdarg.h 110

Version 7.2, May 1995
Copyright © 1990–1995 Thinking Machines Corporation

stdlib.h 112
string.h 114

Index **117**

Copyright © 1990–1995 Thinking Machines Corporation