Contents

List of Examples ix

List of Figures xi

List of Tables xiii

Introduction xv
Organization xv
Additional Reading xvi
Typographical Conventions xvii

1. Compiling, Linking, and Running Programs 1
Compiling and Linking 1
- Drivers 1
- Compilation 2
- Compiling Multilanguage Programs 3
- Linking Objects 5
- Specifying Link Libraries 6

Driver Options 7
- Compiling Simple Programs 7
- Specifying Source File Format 8
- Specifying Compiler Input and Output Files 8
- Using a Defaults Specification File 9
- Specifying Target Machine Features 9
- Specifying Memory Allocation and Alignment 10
- Specifying Debugging and Profiling 10
- Specifying Optimization Levels 11
- Controlling Compiler Execution 12

Object File Tools 13
Archiver 13

Run-Time Considerations 14
Invoking a Program 14
Maximum Memory Allocations 14
File Formats 17
Preconnected Files 18
File Positions 18
Unknown File Status 19
Quad-Precision Operations 19
Run-Time Error Handling 19
Floating Point Exceptions 20

2. Storage Mapping 21
Alignment, Size, and Value Ranges 21
Access of Misaligned Data 24
Accessing Small Amounts of Misaligned Data 25
Accessing Misaligned Data Without Modifying Source 25

3. Fortran Program Interfaces 27
How Fortran Treats Subprogram Names 27
Working with Mixed-Case Names 28
Preventing a Suffix Underscore with $ 28
Naming Fortran Subprograms from C 29
Naming C Functions from Fortran 29
Testing Name Spelling Using *nm* 29
Correspondence of Fortran and C Data Types 30
Corresponding Scalar Types 30
Corresponding Character Types 31
Corresponding Array Elements 31
How Fortran Passes Subprogram Parameters 33
Normal Treatment of Parameters 33
Calling Fortran from C 34
Calling Fortran Subroutines from C 34
Calling Fortran Functions from C 37

Calling C from Fortran 39
Normal Calls to C Functions 39
Using Fortran COMMON in C Code 41
Using Fortran Arrays in C Code 41
Calls to C Using LOC%, REF% and VAL% 42
Making C Wrappers with *mkf2c* 44
Using *mkf2c* and *extcentry* 48
Makefile Considerations 49

4. System Functions and Subroutines 51
Library Functions 51
Extended Intrinsic Subroutines 59
DATE 60
IDATE 60
ERRSNS 60
EXIT 61
TIME 61
MVBITS 62
Extended Intrinsic Functions 62
SECNDS 63
RAN 63

5. Fortran Enhancements for Multiprocessors 65
Overview 65
Directives 66
Parallel Loops 66
Writing Parallel Fortran 68
C$DOACROSS 68
C$& 74
C$ 75
C$MP_SCHEDTYPE and C$CHUNK 75
Nesting C$DOACROSS 76
Analyzing Data Dependencies for Multiprocessing 76
Breaking Data Dependencies 82

Work Quantum 87
Cache Effects 88
Performing a Matrix Multiply 89
Understanding Trade-Offs 90
Load Balancing 91
Reorganizing Common Blocks To Improve Cache Behavior 93
Advanced Features 93
mp_block and mp_unblock 93
mp_setup, mp_create, and mp_destroy 94
mp_blocktime 94
mp_numthreads, mp_set_numthreads 95
mp_suggested_numthreads 95
mp_my_threadnum 95
Environment Variables: MP_SET_NUMTHREADS, MP_BLOCKTIME, MP_SETUP 96
Environment Variables: MP_SUGNUMTHD, MP_SUGNUMTHD_MIN, MP_SUGNUMTHD_MAX, MP_SUGNUMTHD_VERBOSE 97
Environment Variables: MP_SCHEDTYPE, CHUNK 97
mp_setlock, mp_unsetlock, mp_barrier 98
Local COMMON Blocks 98
Compatibility With sproc 99
DOACROSS Implementation 100
Loop Transformation 100
Executing Spooled Routines 101
PCF Directives 102
Parallel Region 104
PCF Constructs 104
Restrictions 114
A Few Words About Efficiency 115

Synchronization Intrinsics 116
Atomic fetch-and-op Operations 117
Atomic op-and-fetch Operations 117
Atomic BOOL Operation 118
Atomic synchronize Operation 119
Atomic lock and unlock Operations 119
Example of Implementing a Pure Spin-Wait Lock 120

6. Parallel Programming on Origin2000 121
Performance Tuning of Parallel Programs on Origin2000 122
Improving Program Performance 122
Choosing Between Multiple Options 125
New Directives for Performance Tuning on Origin2000 126
Data Distribution Directives 127
Nested Doacross Directive 129
Affinity Scheduling 129
Data Affinity 130
Thread Affinity 132
Specifying Processor Topology With the ONTO Clause 132
Types of Data Distribution 133
Regular Data Distribution 133
Data Distribution With Reshaping 134
Query Intrinsics for Distributed Arrays 136
Implementation of Reshaped Arrays 137
Regular vs. Reshaped Data Distribution 139
Explicit Placement of Data 140
Optional Environment Variables and Compile-Time Options 141
Examples 142
Distributing Columns of a Matrix 143
Using Data Distribution and Data Affinity Scheduling 144
Parameter Passing 145
Redistributed Arrays 146
Irregular Distributions and Thread Affinity 147

7. Compiling and Debugging Parallel Fortran 149
Compiling and Running 149
Using the **–static** Option 150
Examples of Compiling 150
Profiling a Parallel Fortran Program 151
Debugging Parallel Fortran 152
General Debugging Hints 152

A. Run-Time Error Messages 155

Index 163