Index

A

aliasing, array, 42, 93
ALIGN, compiler directive, 80
alignment, array, 42, 77
alignment, default, 78
argument passing, array, 130
array aliasing, 42, 93
array alignment, 42, 77
array argument passing, 130
array descriptor, 46
array descriptor, CMRTS, 172, 178
array geometries, 46, 49
array geometry, CMRTS, 173, 179
array layout, 42, 43
array shape, 43
array types, CM Fortran, 146
array, subgrid, 45
array_geometry, **CMRT_desc_t** structure slot, 178
arrays in CMRTS, 172
assembler source code, examining, 32
assembly code, SPARC and DPEAC, 8
axis masks, 65
axis ordering, 52
axis weighting, 62
axis_descriptors, CMRTS machine geometry slot, 183
axis_length, CMRTS axis descriptor slot, 184
-axisreorder
  axis specification, 60
  compiler switch, 70

B

:block :pdesc, axis specification, 63
:block :procs, axis specification, 63
block layout, 45
block move, communication operator, 125
breaking code blocks, 107
busy time, 24

C

-c, compiler switch, 17, 18
canonical array layout, 45, 55
canonical layout algorithm, 44
CM busy time, 24
CM elapsed time, 24
CM Fortran
  execution model, 9
  program execution, 11
CM Fortran compiler (**cmf**), 12
CM Run-Time System (CMRTS), 171
CM Run-Time System (CMRTS), 8
cm_location, **CMRT_desc_t** structure slot, 178
CM_timer facility, 23, 25
CM-5, 3
  system, 3
-cm5, compiler switch, 17, 18
CMCOM layer, of CMRTS, 171
CMCOM_axis_descriptor, CMRTS data structure, 184
CMCOM_machine_geometry_t, CMRTS data structure, 173, 182
CMCOM_pe_allocate_stack_space, CMRTS memory allocation function, 189
CMCOM_pe_get_stack_pointer, CMRTS memory allocation function, 189
CMCOM_pe_set_stack_pointer, CMRTS memory allocation function, 189
-cmdebug, compiler switch, 17, 21
CMF_ALLOCATE_DETAILED_ARRAY, 63
CMF_allocate_detailed_array, 70

Copyright © 1994 Thinking Machines Corporation

CMF_ALLOCATE_LAYOUT_ARRAY, 59
CMF_describe_array, 46, 51
CMF_GET_AXIS_ORDERS, 75
CMF_GET_GLOBAL_LOCATION, 75
CMF_GET_LOCAL_EXTENTS, 75
CMF_GET_LOCAL_STRIDES, 75
CMF_GET_PHYSICAL_LOCATION, 75
CMF_GET_PROCESSOR_MASKS, 75
CMF_my_send_address, 56
CMFPROF_N_USER_TIMERS, 32
CMPROFILING, 27
CMPROFILING_DATA_FILENAME, 27
CMPROFILING_EXECUTABLE_FILENAME, 27
CMF_SERIAL, axis specification, 59
CMIP layer, of CMRTS, 171
-cmprofile, compiler switch, 17, 21, 23, 27
CMRT layer, of CMRTS, 171
CMRT_allocate_heap_field, CMRTS memory allocation function, 188
CMRT_allocate_physical_heap_field, CMRTS memory allocation function, 187
CMRT_allocate_physical_stack_field, CMRTS memory allocation function, 187
CMRT_allocate_stack_field, CMRTS memory allocation function, 188
CMRT_array_geometry_t, CMRTS data type, 173, 179
CMRT_cm_location_t, CMRTS parallel memory location, 174
CMRT_deallocate_heap_field, CMRTS memory allocation function, 188
CMRT_deallocate_physical_heap_field, CMRTS memory allocation function, 187
CMRT_deallocate_physical_stack_through, CMRTS memory allocation function, 187
CMRT_deallocate_stack_field_through, CMRTS memory allocation function, 188
CMRT_desc_get_cm_location, CMRTS accessor function, 178
CMRT_desc_get_element_size, CMRTS accessor function, 178
CMRT_desc_get_geometry, CMRTS accessor function, 178
CMRT_desc_get_lower_bound, CMRTS accessor function, 181
CMRT_desc_get_number_of_elements, CMRTS accessor function, 181
CMRT_desc_get_rank, CMRTS accessor function, 181
CMRT_desc_get_subgrid_dimension, CMRTS accessor function, 186
CMRT_desc_get_subgrid_size, CMRTS accessor function, 183
CMRT_desc_get_upper_bound, CMRTS accessor function, 181
CMRT_desc_t, CMRTS array descriptor, 172, 178
CMRTS, 8
CMSSL routines, 161
code block, startup costs, 101
code blocks, 13, 99
  code that breaks, 107
  compiler optimization of, 107
  counting, 33
  counting Flops, 104
  serial axis optimizations, 109
  timing, 31, 103
comm-comp temporary arrays, 147
common block arrays, CM Fortran array type, 146
common subexpression temporary arrays, 150
compile-time shape checking, 91
compile-time context, 121
compile-time geometry, 121
compiler temporaries, 147
Control Network, 3
counting Flops, in code blocks, 104
cross-geometry move (VP move), communication operator, 124
CSE temporary arrays, 150
cycles, SPARC, estimating, 33
cyclic layout, 45

Version 2.1, January 1994
Copyright © 1994 Thinking Machines Corporation

D

Data Network, 3
default alignment, 78
dependence analysis, 113
descriptor, array, in CMRTS, 172, 178
detailed array layouts, 63
dispatch loop, 11
DPEAC assembly code, 8
dynamically allocated arrays, CM Fortran array type, 146

E

elapsed time, 24
element_size, **CMRT_desc_t** structure slot, 178
execution model, CM Fortran, 9
extents, CMRTS array geometry slot, 180

F

Flops, estimating, 34
FORALL, optimizations, 117, 131
-forall_print, compiler switch, 17, 20, 140

G

-g, compiler switch, 17, 21
  affect on timing results, 23
garbage data, in CMRTS arrays, 173
garbage element padding, 48
garbage mask, 48
garbage_mask, CMRTS array geometry slot, 180
geometries, array, 46
geometries, CMRTS, 173
geometry object, 46
get, communication operator, 125
global/local programming model, 165
grid, physical, 45

H

heap, parallel, 144
highest-axis-varies-fastest ordering, 54

I

I/O, parallel, timing, 31, 37
instruction FIFO, between PM and nodes, 11
intrinsics
  and parallel communication, 127
  that generate communication, 129

L

LAYOUT, compiler directive, 59, 63
layout, array, 42, 43
-list, compiler switch, 17, 19
listing file, 19
local arrays, CM Fortran array type, 146
location analysis, 120
location equivalence, 121
location, preferred, 122
loops, nested and flattened, 112
lower_bounds, CMRTS array geometry slot, 180
lowest-axis-varies-fastest ordering, 54

M

machine geometry, 49
machine geometry, CMRTS, 173, 182
machine_geometry, CMRTS array geometry slot, 180
memory allocation
  in CMRTS, 187
  under CMOST, 158
memory errors, 155
memory management, 42, 143
memory mapping, of CM-5, 144

N

Network Interface, 5
news, communication operator, 124
no-motion detection, 110
-noaxisreorder, compiler switch, 17, 21, 54, 61, 71
nodal programming model, 165
-nopadding, compiler switch, 17, 21, 50, 61

Copyright © 1994 Thinking Machines Corporation

-nopadding, compiler switch, 13
number_of_elements, CMRTS array geometry slot, 180

O

-O, compiler switch, 17, 20
-o, compiler switch, 17
off_chip_length, CMRTS axis descriptor slot, 186
off_chip_mask, CMRTS axis descriptor slot, 186
off_chip_positions, CMRTS axis descriptor slot, 186
ordinary local arrays, CM Fortran array type, 146
out-of-memory errors, handling, 155

P

padding. *See* garbage element padding
parallel code, 12
parallel code stream, 15
parallel communication, 14, 42, 119
parallel computation, 14, 42, 99
parallel data, 9
parallel heap, 144
parallel instructions, 9
parallel memory, 7, 143
parallel memory allocation, in CMRTS, 187
parallel stack, 144
partition manager (PM), 3, 5
partition, CM-5, 3
:pdesc, axis specification, 63
PE code block. *See* code blocks
permanent local arrays, CM Fortran array type, 146
physical address, 65
physical axis ordering, 52
physical axis sequence, 52
physical grid, 45
PN kernel, 145
  shadow in memory, 145
power_of_two, CMRTS axis descriptor slot, 184
preferred location, 122
Prism programming environment, 23, 27
processing element (PE), 7
processing elements, 174
processing node (PN), 3, 5
processor masks. *See* axis masks
:procs, axis specification, 63
product_subgrid_lengths, CMRTS machine geometry slot, 183
program execution, CM Fortran, 11

R

rank
  CMRTS array geometry slot, 180
  CMRTS machine geometry slot, 182
reductions, and parallel communication, 128
RTS_ENABLE_STACK_DEALLOCATION, 159
RTS_MIN_HEAP/STACK_MEMORY, 159
RTS_PM_STACK_GROWTH_QUANTUM, 159
RTS_PM_STACK_SHRINK_QUANTUM, 159
Run-Time System (CMRTS), 171

S

-S, compiler switch, 17, 18
scalar code, 12
scalar code stream, 15
scalar data, 9
scalar instructions, 9
scalar memory, 143
send, communication operator, 125
send address, 55
:serial, axis specification, 59, 61, 109
serial axes, 61, 109
  how the compiler recognizes, 109
serial axis optimizations, 109
serial sections, 114
shadow, of PN kernel in memory, 145
shape, array, 43
shifts, and parallel communication, 128
single-precision optimizations, 118
-sparc, compiler switch, 17
-sparc, compiler switch, 10
SPARC assembly code, 8
SPARC cycles, estimating, 33

Version 2.1, January 1994
Copyright © 1994 Thinking Machines Corporation

spreads, and parallel communication, 127
stack, parallel, 144
startup costs, of code blocks, 101
subgrid
  in CMRTS arrays, 176
  of parallel array, 10
subgrid axis ordering, 53
subgrid axis sequence, 53
subgrid loop, 13, 99
subgrid size, 13
subgrid, array, 45
subgrid_axis_increment, CMRTS axis descriptor slot, 185
subgrid_bits_length, CMRTS axis descriptor slot, 186
subgrid_bits_mask, CMRTS axis descriptor slot, 186
subgrid_bits_position, CMRTS axis descriptor slot, 186
subgrid_length, CMRTS axis descriptor slot, 184
subgrid_orthogonal_length, CMRTS axis descriptor slot, 185
subgrid_outer_count, CMRTS axis descriptor slot, 185
subgrid_outer_increment, CMRTS axis descriptor slot, 185

T

temporary arrays, compiler, 147
timesharing, 24
timesharing daemon, 24
timing, code blocks, 103
total_off_chip_length, CMRTS machine geometry slot, 182
ts-daemon, 24

U

upper_bounds, CMRTS array geometry slot, 180
using Prism and CM timers together, 32

V

-veccode, compiler switch, 17, 18, 32
vector length, of VUs, 49
vector length padding, 49
vector unit (VU) accelerators, 3, 6
vector unit (VU) chips, 6
VP Move (cross-geometry move), communication operator, 124
-vu, compiler switch, 17, 18

W

wallclock time, 24
weighting of axes, 62

Z

zero-based axis numbering, in CMRTS, 47

Copyright © 1994 Thinking Machines Corporation