Appendix B

CMMD Function Summary: Fortran Syntax

Note: In this summary, <array> is declaration of array arguments; it stands for any named type, which is the data type of the F77 data contained in the array. Stated more formally,

<array> : : = **`INTEGER`** | **`REAL`** | **`REAL*8`** | **`LOGICAL`**

B.1 Point-to-Point Functions

B.1.1 Cooperative Message Passing

INTEGER FUNCTION **`CMMD_send_block(`***dest_node, tag, buffer*
buffer_desc`)`
integer *dest_node*, *tag*, **`buffer_desc`**
<array> *buffer*

INTEGER FUNCTION **`CMMD_send_block_v(`***dest_node, tag, buffer,*
elem_len, stride, elem_count`)`
integer *dest_node, tag,*
<array> *buffer*
integer *elem_len, stride, elem_count*

INTEGER FUNCTION **`CMMD_receive_block(`***source_node, tag, buffer,*
buffer_desc`)`
integer *source, tag*
<array> *buffer*
integer *buffer_desc*

INTEGER FUNCTION **`CMMD_receive_block_v(`***source_node, tag, buffer, elem_len, stride, elem_cnt***`)`**

integer *source, tag*
<array> *buffer*
integer *elem_len, stride, elem_count*

INTEGER FUNCTION **`CMMD_send_and_receive(`***source_node, source_tag, inbuffer, inbuf_desc, dest_node, dest_tag, outbuffer, outbuf_desc***`)`**

integer *source_node, source_tag*
<array> *inbuffer*
integer *inbuf_desc*
integer *dest_node, dest_tag*
<array> *outbuffer*
integer *outbuf_desc*

INTEGER FUNCTION **`CMMD_send_and_receive_v(`***source_node, source_tag, inbuffer, inelem_len, in_stride, in_elem_count, dest_node dest_tag, outbuffer, out_elem_len, out_stride, out_elem_count***`)`**

integer *source_node, source_tag*
<array> *inbuffer*
integer *in_elem_len, in_stride, in_elem_count*
integer *dest_node, dest_tag*
<array> *outbuffer*
integer *out_elem_len, out_stride, out_elemen_count*

INTEGER FUNCTION **`CMMD_swap(`***partner, inbuffer, inbuf_desc, outbuffer, outbuf_desc***`)`**

integer *partner*
<array> *inbuffer*
integer *inbuf_desc*
<array> *outbuffer*
integer *outbuf_desc*

INTEGER FUNCTION **CMMD_swap_v(***partner, inbuffer, in_elem_len, in_stride, in_elem_count, outbuffer, out_elem_len, out_stride, out_elem_count***)**

integer *partner*
<array> *inbuffer*
integer *in_elem_len*, *in_stride*, *in_elem_count*
<array> *outbuffer*
integer *out_elem_len*, *out_stride*, *out_elem_count*

B.1.2 Asynchronous Message Passing

INTEGER FUNCTION **CMMD_send_async(***dest_node, tag, buffer, buffer_desc, handler, mcb, handler_arg***)**

integer *dest_node*, *tag*
<array> *buffer*
integer *buffer_desc*
SUBROUTINE *handler* (*mcb, handler_arg*)
    integer *mcb*
    <array> *handler_arg*

INTEGER FUNCTION **CMMD_receive_async(***source_node, tag, buffer, buffer_desc, handler, mcb, handler_arg***)**

integer *source_node*, *tag*
<array> *buffer*
integer *buffer_desc*
SUBROUTINE *handler* (*mcb, handler_arg*)
    integer *mcb*
    <array> *handler_arg*

INTEGER FUNCTION **CMMD_send_noblock(***dest_node, tag, buffer, buffer_desc***)**

integer *dest_node*, *tag*
<array> *buffer*
integer *buffer_desc*

B.2 Cooperative and Asynchronous Auxiliary Functions

B.2.1 Processor Information Functions

INTEGER FUNCTION **CMMD_self_address()**
INTEGER FUNCTION **CMMD_partition_size()**

B.2.2 Cooperative Message-Passing Information

INTEGER FUNCTION **CMMD_msg_pending(***node, tag***)**
  integer *node*, *tag*
INTEGER FUNCTION **CMMD_bytes_received()**
INTEGER FUNCTION **CMMD_bytes_sent()**
INTEGER FUNCTION **CMMD_msg_sender()**
INTEGER FUNCTION **CMMD_msg_tag()**

B.2.3 Asynchronous Message-Passing Information

INTEGER FUNCTION **CMMD_mcb_pending(***node, tag***)**
  integer *node*, *tag*

INTEGER FUNCTION **CMMD_mcb_node(***mcb***)**
INTEGER FUNCTION **CMMD_mcb_source(***mcb***)**
INTEGER FUNCTION **CMMD_mcb_dest(***mcb***)**
INTEGER FUNCTION **CMMD_mcb_tag(***mcb***)**
INTEGER FUNCTION **CMMD_mcb_bytes(***mcb***)**
  integer *mcb*
INTEGER FUNCTION **CMMD_mcb_buffer(***mcb***)**
INTEGER FUNCTION **CMMD_msg_done(***mcb***)**
  integer *mcb*
INTEGER FUNCTION **CMMD_all_msgs_done()**
INTEGER FUNCTION **CMMD_all_sends_done()**
INTEGER FUNCTION **CMMD_all_receives_done()**

SUBROUTINE **CMMD_msg_wait (***mcb***)**
  integer *mcb*
SUBROUTINE **CMMD_all_msgs_wait()**

SUBROUTINE **CMMD_free_mcb (***mcb***)**
integer *mcb*

B.3 Virtual Channels

INTEGER FUNCTION **CMMD_open_send_channel**
(*dest_node, tag, buffer, buffer_desc***)**
integer *dest_node, tag, buffer_desc*
<array> *buffer*
INTEGER FUNCTION **CMMD_open_receive_channel**
(*source_node, tag, buffer, buffer_desc***)**
integer *source_node, tag, buffer_desc*
<array> *buffer*

INTEGER FUNCTION **CMMD_send_channel_status(***scd***)**
integer *scd*
INTEGER FUNCTION **CMMD_receive_channel_status(***rcd***)**
integer *rcd*

INTEGER FUNCTION **CMMD_is_channel_writable(***scd***)**
integer *scd*
INTEGER FUNCTION **CMMD_is_channel_readable(***rcd***)**
integer *rcd*

INTEGER FUNCTION **CMMD_write_channel(***scd***)**
integer *scd*
INTEGER FUNCTION **CMMD_reset_channel(***rcd***)**
integer *rcd*

INTEGER FUNCTION **CMMD_close_send_channel(***scd***)**
integer *scd*
INTEGER FUNCTION **CMMD_close_receive_channel(**int *rcd***)**
integer *rcd*

INTEGER FUNCTION **CMMD_all_channels_free()**
SUBROUTINE **CMMD_all_channels_free_wait()**

B.4 Global Functions

B.4.1 Broadcast

SUBROUTINE **`CMMD_bc_to_nodes(`***buffer, buffer_desc***`)`**
<array> *buffer*
integer *buffer_desc*

SUBROUTINE **`CMMD_receive_bc_from_node(`***buffer, buffer_desc***`)`**
<array> *buffer*
integer *buffer_desc*

B.4.2 Reduce, Scan, and Concatenate

INTEGER FUNCTION **`CMMD_reduce_int(`***value***`,`** *combiner***`)`**
integer *value*
integer *combiner*

REAL*8 FUNCTION **`CMMD_reduce_float(`***value***`,`** *combiner***`)`**
real*4 *value*
integer *combiner*

REAL*8 FUNCTION **`CMMD_reduce_double(`***value***`,`** *combiner***`)`**
real*8 *value*
integer *combiner*

SUBROUTINE **`CMMD_reduce_v(`***dest, source, combiner, stride, elem_count***`)`**
<array> *dest, source*
integer *combiner, stride, elem_count*

INTEGER FUNCTION **`CMMD_scan_int(`***value, combiner, direction, smode, sbit, inclusion***`)`**
integer *value, combiner, direction*
integer *smode, sbit, inclusion*

REAL*8 FUNCTION **`CMMD_scan_float(`***value, combiner, direction, smode, sbit, inclusion***`)`**
real*4 *value*
integer *combiner, direction*
integer *smode, sbit, inclusion*

REAL*8 FUNCTION **CMMD_scan_double(***value, combiner, direction, smode, sbit, inclusion***)**
real*8 *value*
integer *combiner, direction*
integer *smode, sbit, inclusion*

SUBROUTINE **CMMD_scan_v(***dest, source, combiner, direction, smode, sbit, inclusion, stride, elem_count***)**
<array> *dest, source*
integer *combiner, direction*
integer *smode, sbit, inclusion*
integer *stride, elem_count*

SUBROUTINE **CMMD_concat_with_nodes(***element, buffer, elem_length***)**
<array> *element, buffer*
integer *elem_length*

B.4.3 Global Synchronization

SUBROUTINE **CMMD_sync_with_nodes()**

B.4.4 Asychronous Global OR

INTEGER FUNCTION **CMMD_set_global_or(***value***)**
integer *value*

INTEGER FUNCTION **CMMD_get_global_or()**

B.5 Interrupts and Polling

INTEGER FUNCTION **CMMD_enable_interrupts**
INTEGER FUNCTION **CMMD_disable_interrupts**
INTEGER FUNCTION **CMMD_interrupts_are_enabled**
INTEGER FUNCTION **CMMD_inside_interrupt_handler**
SUBROUTINE **CMMD_poll**

B.6 I/O Functions

INTEGER FUNCTION **`CMMD_set_open_mode(`***io_mode***`)`**
    integer *io_mode*

INTEGER FUNCTION **`CMMD_get_open_mode()`**

INTEGER FUNCTION **`CMMD_set_io_mode(`***unit, io_mode***`)`**
INTEGER FUNCTION **`CMMDD_fset_io_mode(`***unit*`,` *io_mode***`)`**
    integer *unit*
    integer *io_mode*

INTEGER FUNCTION **`CMMD_set_append_mode(`***unit***`)`**
    integer *unit*

INTEGER FUNCTION **`CMMD_get_io_mode(`***unit***`)`**
INTEGER FUNCTION **`CMMD_fget_io_mode(`***unit***`)`**
    integer *unit*

REAL* FUNCTION **`dseek(`***unit, offset, whence***`);`**
    integer *unit*
    real*8 *offset*
    integer *whence*
REAL*8 FUNCTION **`dtell(`***unit***`)`**
    integer *unit*
REAL*8 FUNCTION **`dlength(`***unit***`)`**
    integer *unit*
INTEGER FUNCTION **`dftruncate(`***unit, length***`)`**
    integer *unit*
    real*8 *length*

INTEGER FUNCTION **`CMMD_physical_blocking`**(*unit*)
    integer *unit*

INTEGER FUNCTION **`CMMD_physical_write(`***unit, buffer, length***`)`**
    integer *unit*
    <array> *buffer*
    integer *length*

INTEGER FUNCTION **`CMMD_physical_read(`***unit, buffer, length***`)`**
    integer *unit*
    <array> *buffer*
    integer *length*

B.7 Timing Functions

INTEGER FUNCTION **CMMD_node_timer_clear(***timer***)**
INTEGER FUNCTION **CMMD_node_timer_start(***timer***)**
INTEGER FUNCTION **CMMD_node_timer_stop(***timer***)**
REAL*8 FUNCTION **CMMD_node_timer_elapsed(***timer***)**
REAL*8 FUNCTION **CMMD_node_timer_busy(***timer***)**
REAL*8 FUNCTION **CMMD_node_timer_idle(***timer***)**

B.8 Changing Defaults

SUBROUTINE **CMMD_set_msg_matching_fcn(***func***)**
INTEGER FUNCTION *func* **(***req_node, req_tag, req_length, node, tag, length***)**
integer *req_node, req_tag, req_length*
integer *send_node, send_tag, send_length*
INTEGER FUNCTION **CMMD_reset_partition_size**
integer *new_size*
SUBROUTINE **CMMD_enable_safety()**
SUBROUTINE **CMMD_disable_safety()**

B.9 Host/Node Environment Functions

SUBROUTINE **CMMD_enable()**
SUBROUTINE **CMMD_enable_host()**

INTEGER FUNCTION **CMMD_host_node(**)

SUBROUTINE **CMMD_sync_host_with_nodes()**
SUBROUTINE **CMMD_sync_with_host()**

SUBROUTINE **CMMD_bc_from_host(***buffer, buffer_desc***)**
SUBROUTINE **CMMD_receive_bc_from_host(***buffer, buffer_desc*)
<array> *buffer*
integer *buffer_desc*

SUBROUTINE **CMMD_bc_from_host_v(***vector, elem_bytes, stride, elem_count***)**

SUBROUTINE **CMMD_receive_bc_from_host_v(***vector, elem_bytes stride, elem_count***)**
<array> *vector*
integer *elem_bytes, stride, elem_count*

SUBROUTINE **CMMD_distrib_to_nodes(***buffer, length***)**
SUBROUTINE **CMMD_receive_element_from_host(***buffer, length***)**
<array> *buffer*
integer *length*

SUBROUTINE **CMMD_gather_from_nodes(***buffer, length***)**
SUBROUTINE **CMMD_concat_elements_to_host(***element, length***)**
<array> *buffer*
integer *length*

INTEGER FUNCTION **CMMD_reduce_from_nodes_int(***value, combiner***)**
integer *value, combiner*

REAL*8 FUNCTION
CMMD_reduce_from_nodes_float(*value, combiner***)**
real*4 *value*
integer *combiner*

REAL*8 FUNCTION
CMMD_reduce_from_nodes_double(*value, combiner***)**
real*8 *value*
integer *combiner*

INTEGER FUNCTION **CMMD_reduce_to_host_int(***value, combiner***)**
integer *function*, *combiner*

REAL*8 FUNCTION
CMMD_reduce_to_host_float(*value, combiner***)**
real*4 *value*
integer *combiner*

REAL*8 FUNCTION
CMMD_reduce_to_host_double(*value, combiner***)**
real*8 *value*
integer *combiner*

SUBROUTINE **CMMD_reduce_from_nodes_v(***dest, source, combiner, stride, elem_count***)**
SUBROUTINE **CMMD_reduce_to_host_v(***dest, source, combiner, stride, elem_count***)**
<array> *dest, source*
integer *combiner, stride_bytes, elem_count*

SUBROUTINE **CMMD_service_loop()**
SUBROUTINE **CMMD_global_suspend_servers()**
INTEGER FUNCTION **CMMD_poll_for_services()**