Real Time Executive Interface Definition January 22, 1988

3.7.5 RN\_DELETE

NAME

rn\_delete – “Delete a Region”

SYNOPSIS

#include < memory.h >

uint rn\_delete ( rnid )

uint rnid; /\* region id as returned by rn\_create or rn\_ident \*/

DESCRIPTION

This directive deletes the specified region, provided that none of its segments is still allocated. After this directive has successfully executed, the executive will reject any rn\_getseg and rn\_retseg directives for the region.

RETURN VALUE

If rn\_delete successfully deleted the region, then 0 is returned.

If the region was not successfully deleted, an error code is returned.

ERROR CONDITIONS

Invalid rnid.

Cannot delete - outstanding segments.

NOTES

Not callable from ISR.

Will not cause a pxeempt.

January 22, 1988 Real Time Executive Interface Definition

3.7.6 RN\_GETSEG

NAME

rn\_getseg – “Get a Segment”

SYNOPSIS

#include < memory.h >

uint rn\_getseg ( rnid, size, flags, timeout, &segaddr )

uint rnid; /\* region id as returned by rn\_create or rn\_ident \*/

uint size; /\* segment size in bytes \*/

uint flags; /\* directive options \*/

uint timeout; /\* number of ticks to wait for memory \*/

/\* 0 indicates forever \*/

char \*segaddr; /\* segment address – returned by this call \*/

The flag field values are defined as follows:

NOWAIT set if the task is to return immediately

clear if the task is to wait for memory

DESCRIPTION

This directive allocates a variable size segment from the region specified by the rnid. The address of the segment is returned to the caller in segaddr.

The actual segment length is a multiple of the region pagesize. Thus, the segment allocated may be larger than the requested size.

RETURN VALUE

if rn\_getseg successfully allocated the segment, the address of the segment is returned in segaddr and 0 is returned.

If the call was not successful, an error code is returned.

ERROR CONDITIONS

Invalid rnid.

No memory available (no-wait only).

Timeout occurred before memory was available (wait with timeout).

Region has been deleted.

NOTES

Real Time Executive Interface Definition January 22, 1988

Not callable from ISR.

Requester will be blocked when the wait option is selected and the memory is not available.

Real Time Executive Interface Definition January 22, 1988

3.7.7 RN\_RETSEG

NAME

rn\_retseg – “Return a Segment”

SYNOPSIS

#include < memory.h >

uint rn\_retseg ( rnid, segaddr )

uint rnid; /\* region id as returned by rn\_create or rn\_ident \*/

char \*segaddr /\* segment address as returned by rn\_getseg \*/

DESCRIPTION

This directive returns a segment to its region. If possible, the segment is merged with neighboring segments. The resulting segment then becomes available for subsequent allocation, or allocation to tasks already waiting.

RETURN VALUE

If rn\_retseg successfully returned the segment, then 0 is returned.

If the call was not successful, an error code is returned.

ERROR CONDITIONS

Invalid rnid.

Segment not from specified region.

NOTES

Not callable from ISR.

May cause a preempt if a task waiting for memory becomes ready as a result of this call and has a higher priority than the running task, and the preempt mode is in effect.

Real Time Executive Interface Definition January 22, 1988

3.7.8 PT\_CREATE

NAME

pt\_create – “Create a Partition”

SYNOPSIS

#include < memory.h >

uint pt\_create ( name, paddr, length, bsize, flags, &ptid, &bnum )

uint name; /\* used defined 4-byte partition name \*/

char \*paddr; /\* physical start address of partition \*/

uint length; /\* physical length in bytes \*/

uint bsize; /\* size of buffers in bytes \*/

uint flags; /\* partition attributes \*/

unit ptid; /\* partition id – returned by this call \*/

unit bnum; /\* number of buffers in partition – returned by this call \*/

Flags field values:

GLOBAL set to indicate the partition is

a multiprocessor global resource

clear to indicate the partition is local

DESCRIPTION

This directive allows the user to create a partition of fixed size buffers from a contiguous memory area. The partition id will be returned in ptid by the executive to use for pt\_getbuf and pt\_retbuf directives for the partition. The number of buffers created by the executive will be returned in bnum.

The partition physical start address specified in paddr will be long­word aligned by the executive. In systems with an MMU, the partition physical start address must be on the pagesize boundary.

The executive may use memory within the partition for partition and buffer data structures. Therefore, the product of the buffer count and buffer size will be slightly less than the length of the partition.

By setting the GLOBAL value in the flags field, the ptid will be sent to all processors in the system, to be entered into a global resource table. The system is defined as the collection of interconnected processors.

The maximum number of partitions that may exist at any one time is a configuration parameter.

RETURN VALUE

If pt\_create successfully created the partition, the ptid and bnum are filled in and 0 in returned.

January 22, 1988 Real Time Executive Interface Definition

If the call was not successful, an error code in returned.

ERROR CONDITIONS

Too many partitions.

NOTES

Not callable from ISR.

Will not cause a preempt.

Real Time Executive Interface Definition January 22, 1988

1.7.9 PT\_IDENT

NAME

pt\_ident – “Obtain id of a Partition”

SYNOPSIS

#include < memory.h >

uint pt\_ident ( name, node, &ptid )

uint name; /\* used defined 4-byte partition name \*/

uint node; /\* node identifier \*/

/\* 0 indicates any node \*/

uint ptid; /\* partition id – returned by this call \*/

DESCRIPTION

This directive allows a task to identify a previously created partition by name and obtain the ptid to use for pt\_getbuf and pt\_retbuf directives for the partition.

If the partition name is not unique, the ptid returned may not correspond to the partition named in this call.

The partition may have been created by the local processor or any remote processor in a multiprocessor configuration, as long as the partition was created with the GLOBAL flags value set (see pt\_create). If the partition name is not unique within the multiprocessor configuration, a non-zero node identifier must be specified in the node field.

RETURN VALUE

If pt\_ident directive succeeds, the ptid will be filled in and 0 is returned.

If the call was not successful, an error code is returned.

ERROR CONDITIONS

Named partition does not exist.

Invalid node identifier.

NOTES

Can be called from within an ISR.

Will not cause a preempt.

January 22, 1988 Real Time Executive Interface Definition

3.7.10 PT\_DELETE

NAME

pt\_delete – “Delete a Partition”

SYNOPSIS

#include < memory.h >

uint pt\_delete ( ptid )

uint ptid; /\* partition id as returned by pt\_create or pt\_ident \*/

DESCRIPTION

This directive removes a partition, provided that none of its buffers is still allocated.

After this directive has successfully executed, the executive will reject any pt\_getbuf or pt\_retbuf directives for the partition.

The partition must exist on the local processor. If the partition was created with the GLOBAL flags value set in a multiprocessor configuration, a notification will be sent to all processors in the system, so the ptid can be deleted from the global resource table.

RETURN VALUE

If pt\_delete successfully removed the partition, then 0 is returned.

If the call was not successful, an error code is returned.

ERROR CONDITIONS

Invalid ptid.

Cannot delete - some buffers in use.

Partition not created from local node.

NOTES

Not callable from ISR.

Will not cause a preempt.

Real Time Executive Interface Definition January 22, 1988

3.7.11 PT\_GETBUF

NAME

pt\_getbuf– “Get a Buffrer”

SYNOPSIS

#include < memory.h >

uint pt\_getbuf ( ptid, &bufaddr )

uint ptid; /\* partition id as returned by pt\_create or pt\_ident \*/

char \*bufaddr; /\* buffer address – returned by this call \*/

DESCRIPTION

The pt\_getbuf directive will get a buffer from a buffer partition. The buffer address will be returned in bufaddr as a result of this call.

The partition may have been created by the local processor or any remote processor in a multiprocessor configuration, as long as the partition was created with the GLOBAL flags value set (see pt\_create ).

RETURN VALUE

If pt\_getbuf successfully got a buffer, then the address of the buffer is returned in bufaddr and 0 is returned.

If the call was not successful, an error code is returned.

ERROR CONDITIONS

Invalid ptid.

Partition out of free buffers

NOTES

Can be called from within an ISR.

Will not cause a preempt.

January 22, 1988 Real Time Executive Interface Definition

3.7.12 PT\_RETBUF

NAME

pt\_retbuf– “Return a Buffrer”

SYNOPSIS

#include < memory.h >

uint pt\_retbuf ( ptid, bufaddr )

uint ptid; /\* partition id as returned by pt\_create or pt\_ident \*/

char \*bufaddr; /\* buffer start address as returned by pt\_getbuf \*/

DESCRIPTION

The pt\_retbuf directive will return a buffer to the partition from which it was originally allocated.

Buffers are not automatically released when a task is deleted.

RETURN VALUE

If pt\_retbuf successfully returned the buffer, then 0 is returned.

If the buffer was not returned, a value of -1 is returned, and errno is set to indicate the error.

ERROR CONDITIONS

Invalid ptid.

Buffer not from specified partition.

NOTES

Can be called from within an ISR.

Will not cause a preempt.