5.1. PARTITION_CREATE

Create a partition.

Synopsis

```
partition_create( name, addr, length, block_size, options, pid )
```

Input Parameters

name	: string	user defined partition name
addr	: address	start address of partition
length	: integer	length of partition in bytes
block_size	: integer	partition block size in bytes
options	: bit_field	partition create options

Output Parameters

pid	: part_id	kernel defined partition identifier

Literal Values

option:	+GLOBAL	partition is global within the shared memory system

Completion Status

OK	partition_create operation successful
ILLEGAL_USE	operation not callable from XSR or ISR
INVALID_PARAMETER	a parameter refers to an illegal address
INVALID_ADDRESS	area defined is not within actual memory present
INVALID_BLOCK_SIZE	block_size not supported
INVALID_OPTIONS	invalid options value
TOO_MANY_PARTITIONS	too many partitions on the node
PARTITION_OVERLAP	area given overlaps an existing partition

Description

This operation declares an area of memory to be organized as a partition by the kernel. The process of formatting the memory to operate as a partition may require a memory overhead which may be taken from the new partition. It can never be assumed that all of the memory in the partition will be available for allocation. The overhead percentage will be implementation dependent.

5.2. PARTITION_DELETE

Delete a partition.

Synopsis

```
partition_delete( pid, options )
```

Input Parameters

pid	: part_id	kernel defined partition identifier
options	: bit_field	partition deletion options

Output Parameters

<none>

Literal Values

options	+ FORCED_DELETE	deletion will go ahead even though there are unreleased blocks

Completion Status

OK	partition_delete operation successful
ILLEGAL_USE	operation not callable from ISR
INVALID_PARAMETER	a parameter refers to an illegal address
INVALID_ID	partition does not exist
OBJECT_DELETED	partition specified has been deleted
INVALID OPTIONS	invalid options value
PARTITION_IN_USE	blocks from this partition are still allocated
NODE_NOT_REACHABLE	node on which task resides is not reachable

Description

Unless the FORCED_DELETE option was specified, this operation first checks whether the partition has any blocks which have not been returned. If this is the case, then the PARTITION_IN_USE completion status is returned. If not, and in any case if FORCED_DELETE was specified, then the partition is deleted from the kernel data structure

5.3. PARTITION_IDENT

Obtain the identifier of a partition on a given node with a given name.

Synopsis

```
   partition_ident( name, nid, pid, )
```

Input Parameters

name	: string	user defined partition name
nid	: node_id	node identifier

Output Parameters

pid	: part_id	kernel defined partition identifier
block_size	: integer	the partition's block size

Literal Values

nid	= LOCAL_NODE	the node containing the calling task
	= OTHER_NODES	all nodes in the system except the local node

Completion Status

OK	partition_ident operation successful
ILLEGAL_USE	operation not callable from XSR or ISR
INVALID_PARAMETER	a parameter refers to an illegal address
INVALID_NODE	node does not exist
NAME_NOT_FOUND	name does not exist on node
NODE_NOT_REACHABLE	node on which task resides is not reachable

Description

This operation searches the kernel data structure in the node(s) specified for a partition with the given name, and returns its identifier and block size if found. If OTHER_NODES is specified, the node search order is implementation dependent, but will include only those nodes in the shared memory system or subsystem containing the partition. If there is more than one partition with the same name, then the pid of the first one found is returned.

5.4. PARTITION_GET_BLK

Get a block from a partition.

Synopsis

```
   partition_get_blk( pid, blk_addr )
```

Input Parameters

pid	: part_id	kernel defined partition identifier

Output Parameters

blk_addr	: address	address of obtained block

Completion Status

OK	partition_get_blk operation successful
ILLEGAL_USE	operation not callable from ISR
INVALID_PARAMETER	a parameter refers to an illegal address
INVALID_ID	partition does not exist
OBJECT_DELETED	partition specified has been deleted
NO_MORE_MEMORY	no more blocks available in partition
NODE_NOT_REACHABLE	node on which task resides is not reachable

Description

This operation is a request for a single block from the partition's free block pool. If the kernel cannot immediately fulfil the request, it returns the error completion status NO_MORE_MEMORY, otherwise the address of the allocated block is returned. The exact allocation algorithm is implementation dependent.

5.5. PARTITION_RET_BLK

Return a block to its partition.

Synopsis

```
partition_ret_blk( pid, blk_addr )
```

Input Parameters

pid	: part_id	kernel defined partition identifier
blk_addr	: address	address of block to be returned

Output Parameters

<none>

Completion Status

OK	partition_ret_blk operation successful
ILLEGAL_USE	operation not callable from ISR
INVALID_PARAMETER	a parameter refers to an illegal address
INVALID_ID	partition does not exist
OBJECT_DELETED	partition specified has been deleted
INVALID_BLOCK	no block allocated from partition at blk_addr
NODE_NOT_REACHABLE	node on which task resides is not reachable

Description

This operation returns the given block to the given partition's free block pool. The kernel checks that the block was previously allocated from the partition and returns INVALID_BLOCK if it wasn't.