

4.6.3 MCAPI_WAIT_ANY

NAME

`mcapi_wait_any` – waits for any non-blocking operation in a list to complete.

SYNOPSIS

```
#include <mcapi.h>

mcapi_uint_t mcapi_wait_any(
    MCAPI_IN size_t number,
    MCAPI_IN mcapi_request_t* requests,
    MCAPI_OUT size_t* size,
    MCAPI_IN mcapi_timeout_t timeout,
    MCAPI_OUT mcapi_status_t* mcapi_status
);
```

DESCRIPTION

Wait until any non-blocking operation of a list has completed. It is a blocking function and returns the index into the `requests` array (starting from 0) indicating which of any outstanding operations has completed (successfully or with an error). `number` is the number of requests in the array. `requests` is the array of `mcapi_request_t` identifiers for the non-blocking operations. The call only waits for the completion of an operation and doesn't affect any messages/packets/scalars. The `size` parameter is set to number of bytes that were either sent or received by the non-blocking transaction that completed (`size` is irrelevant for non-blocking connect and close calls). The `mcapi_wait_any()` call will return `MCAPI_RETURN_VALUE_INVALID` if any of the requests are cancelled by calls to `mcapi_cancel()` (during the waiting). 0 could be a valid index, so status has to be checked to ensure correctness. The returned status will indicate that a request was cancelled. The units for `timeout` are implementation defined. If a timeout occurs the `mcapi_status` parameter will indicate that a timeout occurred. A value of 0 for the `timeout` parameter indicates that the function will return without blocking with failure or success and a value of `MCAPI_INFINITE` for the `timeout` parameter indicates no timeout is requested, i.e. the function will block until it is unblocked because of failure or success. In regards to timeouts `mcapi_wait_any()` is non destructive, i.e. the request is not cleared in the case of a timeout and can be waited upon multiple times. Multiple threads attempting to wait on the same request will result in an error.

RETURN VALUE

On success, the index into the `requests` array of the `mcapi_request_t` identifier that has completed or has been canceled is returned and `*mcapi_status` is set to `MCAPI_SUCCESS`. On error or timeout `MCAPI_RETURN_VALUE_INVALID` is returned and `*mcapi_status` is set to the appropriate error/status defined below for parameter errors for the `mcapi_wait_any()` call or errors from the requesting functions, as defined in those respective functions.

ERRORS

<code>MCAPI_ERR_NODE_NOTINIT</code>	The node is not initialized.
-------------------------------------	------------------------------