

4.3.3 MCAPI_MSG_RECV_I

NAME

`mcapi_msg_rcv_i` – receives a (connectionless) message from a receive endpoint.

SYNOPSIS

```
#include <mcapi.h>

void mcapi_msg_rcv_i(
    MCAPI_IN mcapi_endpoint_t  receive_endpoint,
    MCAPI_OUT void*  buffer,
    MCAPI_IN size_t  buffer_size,
    MCAPI_OUT mcapi_request_t* request,
    MCAPI_OUT mcapi_status_t* mcapi_status
);
```

DESCRIPTION

Receives a (connectionless) message from a receive endpoint. It is a non-blocking function, and returns immediately. `receive_endpoint` is a local endpoint identifying the receive endpoint. `buffer` is the application provided buffer, and `buffer_size` is the buffer size in bytes. `request` is the identifier used to determine if the receive operation has completed (all the data is in the buffer). Furthermore, this method will abandon the receive and return `MCAPI_ERR_MEM_LIMIT` if the system cannot either wait for sufficient memory to become available or allocate enough memory.

RETURN VALUE

On success, `*mcapi_status` is set to `MCAPI_SUCCESS` if completed and `MCAPI_PENDING` if not yet completed. On error, `*mcapi_status` is set to the appropriate error defined below.

ERRORS

<code>MCAPI_ERR_NODE_NOTINIT</code>	The node is not initialized.
<code>MCAPI_ERR_ENDP_INVALID</code>	Argument is not a valid local endpoint descriptor.
<code>MCAPI_ERR_MSG_TRUNCATED</code>	The message size exceeds the <code>buffer_size</code> .
<code>MCAPI_ERR_TRANSMISSION</code>	Transmission failure. This error code is optional, and if supported by an implementation, its functionality shall be described.
<code>MCAPI_ERR_REQUEST_LIMIT</code>	No more request handles available.
<code>MCAPI_ERR_MEM_LIMIT</code>	No memory available.
<code>MCAPI_ERR_PARAMETER</code>	Incorrect <code>buffer</code> and/or <code>request</code> parameter.