4.4.6 MCAPI_PKTCHAN_RECV_I

NAME

`mcapi_pktchan_recv_i` – receives a (connected) packet on a channel.

SYNOPSIS

```
#include <mcapi.h>

void mcapi_pktchan_recv_i(
     MCAPI_IN mcapi_pktchan_recv_hndl_t receive_handle,
     MCAPI_OUT void** buffer,
     MCAPI_OUT mcapi_request_t* request,
     MCAPI_OUT mcapi_status_t* mcapi_status
);
```

DESCRIPTION

Receives a packet on a connected channel. It is a non-blocking function, and returns immediately. `receive_handle` is the local representation of the handle used to receive packets. When the receive operation completes, the `buffer` parameter is filled with the address of a system-supplied buffer containing the received packet. After the receive request has completed and the application is finished with `buffer`, `buffer` must be returned to the system by calling `mcapi_pktchan_release()`. `request` is the identifier used to determine if the receive operation has completed and `buffer` is ready for use; the `mcapi_test()` , `mcapi_wait()` or `mcapi_wait_any()` function will return the actual size of the received packet. Furthermore, this method will abandon the receive and return `MCAPI_ERR_MEM_LIMIT` if sufficient memory space is not available.

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

`MCAPI_ERR_NODE_NOTINIT`	The node is not initialized.
`MCAPI_ERR_CHAN_INVALID`	Argument is not a valid channel handle.
`MCAPI_ERR_MEM_LIMIT`	No memory available
`MCAPI_ERR_REQUEST_LIMIT`	No more request handles available.
`MCAPI_ERR_TRANSMISSION`	Transmission failure. This error code is optional, and if supported by an implementation, its functionality shall be described.
`MCAPI_ERR_PARAMETER`	Incorrect `buffer` or `request` parameter.