

4.4.2 MCAPI_PKTCHAN_RECV_OPEN_I

NAME

`mcapi_pktchan_recv_open_i` – Creates a typed and directional, local representation of the channel. It also provides synchronization for channel creation between two endpoints. Opens are required on both receive and send endpoints.

SYNOPSIS

```
#include <mcapi.h>

void mcapi_pktchan_recv_open_i(
    MCAPI_OUT mcapi_pktchan_recv_hdl_t* receive_handle,
    MCAPI_IN mcapi_endpoint_t receive_endpoint,
    MCAPI_OUT mcapi_request_t* request,
    MCAPI_OUT mcapi_status_t* mcapi_status
);
```

DESCRIPTION

Opens the receive end of a packet channel. The corresponding calls are required on both sides for synchronization to ensure that the channel has been created. It is a non-blocking function, and the `receive_handle` is filled in upon successful completion. No specific ordering of calls between sender and receiver is required since the call is non-blocking. `receive_endpoint` is the local endpoint associated with the channel. The open call returns a typed, local handle for the connected channel that is used for channel receive operations. An endpoint with a previously open channel can't be opened until the previous channel is disconnected (implies both sides closed).

RETURN VALUE

On success, meaning that both sides of the channel are successfully open, `*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_CHAN_TYPE</code>	Attempt to open a packet channel on an endpoint that has been connected with a different channel type.
<code>MCAPI_ERR_CHAN_DIRECTION</code>	Attempt to open a receive handle on an endpoint that was connected as a sender.
<code>MCAPI_ERR_REQUEST_LIMIT</code>	No more request handles available.