4.2.1 MCAPI_ENDPOINT_CREATE

NAME

`mcapi_endpoint_create` - Create an endpoint.

SYNOPSIS

```
#include <mcapi.h>


mcapi_endpoint_t mcapi_endpoint_create(
     MCAPI_IN mcapi_port_t port_id,
     MCAPI_OUT mcapi_status_t* mcapi_status
);
```

DESCRIPTION

`mcapi_endpoint_create()` is used to create an endpoint on the local node with the specified port_id. A `port_id` of `MCAPI_PORT_ANY` is used to request the next available endpoint on the local node.

MCAPI supports a simple static naming scheme to create endpoints based on global tuple names, <domain_id, node_id, port_id>. Other nodes can access the created endpoint by calling `mcapi_endpoint_get()` and specifying the appropriate domain, node and port id.
Static naming allows the programmer to define an MCAPI communication topology at compile time. This facilitates simple initialization. Section 7.1~~7.1~~ illustrates an example of initialization and bootstrapping using static naming. Creating endpoints using `MCAPI_PORT_ANY` provides a convenient method to create endpoints without having to specify the `port_id`.

There are three types of endpoints, message, packet channel and scalar channel. The endpoint type defines certain aspects of the endpoint's behavior. The endpoint type is used to manage avoidance of messages being sent to connected endpoints. The type is set with mcapi_endpoint_set_attribute(). An endpoint is always created as a message type, which is the default type.

RETURN VALUE

On success, an endpoint is returned and `*mcapi_status` is set to `MCAPI_SUCCESS`. On error, `MCAPI_NULL` (or 0) is returned and `*mcapi_status` is set to the appropriate error defined below. MCAPI_NULL (or 0) could be a valid endpoint value so status has to be checked to ensure correctness.

ERRORS

`MCAPI_ERR_PORT_INVALID` The parameter is not a valid port. There may be no more available ports or the port may be reserved.

`MCAPI_ERR_ENDP_EXISTS` The endpoint is already created.

`MCAPI_ERR_NODE_NOTINIT` The node is not initialized.