

Message Objects

Introduction

The **Message** module manages text strings associated with the error/warning code return values that can be returned from MPI methods. Each method return value has a unique value which can be decoded using the Message methods and macros.

For example, a return value of `MEISynqNetMessagePLL_ERROR` has a defined value of `0x192E`. Passing this value to the `mpiMessage(...)` function returns the following text string:

```
"SynqNet: node PLL unable to lock with drive"
```

If `MEISynqNetMessagePLL_ERROR` was the last return value received from an MPI method, the string returned from `mpiMessage(...)` would also contain extended message information. The extended message information provides greater insight into the problem that has just occurred. In this case, the extended message information would indicate which node had the PLL locking problem:

```
"SynqNet: node PLL unable to lock with drive : Node 3"
```

Notice that the extended message information is delimited from the more generic message by a colon surrounded by spaces " : ":

Extended message information may not always be returned. Extended information is only available for the very last return value from the MPI -- if a second (non-zero) return value has been returned between the time when the first return value was returned and the call to `mpiMessage(...)`, then no extended information would be returned. In addition, not all messages return extended error information.

Methods

Configuration and Information Methods

[mpiMessage](#)

[mpiMessageFunction](#)

Associate message text with a function

Data Types

[MPIMessageFunction](#)

Macros

[mpiMessageID](#)

[mpiMessageMODULE](#)

[mpiMessageNUMBER](#)

MPIMessageFunction

MPIMessageFunction

```
typedef const char *(*MPIMessageFunction)(long);
```

Description

MessageFunction is the type definition for the callback function used by `mpiMessage(...)`. A default callback function is provided internally to all MPI/MEI modules, but an application can also be written to override it and provide a custom message function instead.

See Also

[mpiMessage](#) | [mpiMessageFunction](#)

mpiMessageID

Declaration

```
#define mpiMessageID(module number) \  
((long)((((module) & MPIModuleIdMAX) << 8) | \ ((number) & 0xFF)))
```

Required Header stdmpi.h

Description **MessageID** converts the message module value and number to a unique message identification value.

See Also [mpiMessage](#)

mpiMessageMODULE

Declaration

```
#define mpiMessageMODULE(messageId) \  
(((messageId) & (MPIModuleIdMAX << 16)) >> 16)
```

Required Header `stdmpi.h`

Description **MessageMODULE** converts the message identification value to the message module value.

See Also [mpiMessage](#)

mpiMessageNUMBER

Declaration

```
#define mpiMessageNUMBER(messageId) ((messageId) & 0xFF)
```

Required Header `stdmpi.h`

Description `MessageNUMBER` converts the message identification value to the message number.

See Also [mpiMessage](#)