リファレンスマニュアル

EtherCAT マスタボード
Linux ライブラリ

目　次

第１章 関数一覧

第２章 関数仕様

２－１ ライブラリ使用方法 ・・・

第１章　関数一覧

１）EtherCAT マスタボード アクセス関数

関　数	機　　能
EMM_Connect()	マスタボードと接続します。
EMM_DisConnect()	マスタボードと切断します。
EMM_SmemOpen()	指定した共有メモリをオープンします。
EMM_SmemClose()	指定した共有メモリをクローズします。
EMM_SmemIsMemSize()	指定した共有メモリのサイズを取得します。
EMM_ShutdownNotify()	シャットダウン発生を通知します。
EMM_SetDateTime()	現在日時を設定します。
EMM_Reset()	マスタボードにソフトリセットを実行します。
EMM_GetVersion()	ハードバージョン、ソフトバージョンを取得します。
EMM_GetBoardNumber()	ボードナンバーを取得します。
EMM_SmemReadByte()	指定した共有メモリから 1BYTE(uint8_t)データを取得します。
EMM_SmemReadWord()	指定した共有メモリから 1WORD(uint16_t)データを取得します。
EMM_SmemReadDWord()	指定した共有メモリから 1DWORD(uint32_t)データを取得します。
EMM_SmemReadLWord()	指定した共有メモリから 1DWORDLONG(uint64_t)データを取得します。
EMM_SmemReadBytes()	指定した共有メモリから指定した長さの BYTE 列を取得します。
EMM_SmemWriteByte()	指定した共有メモリへ 1BYTE(uint8_t)データを書き込みます。
EMM_SmemWriteWord()	指定した共有メモリへ 1WORD(uint16_t)データを書き込みます。
EMM_SmemWriteDWord()	指定した共有メモリへ 1DWORD(uint32_t)データを書き込みます。
EMM_SmemWriteLWord()	指定した共有メモリへ 1DWORDLONG(uint64_t)データを書き込みます。
EMM_SmemWriteBytes()	指定した共有メモリへ指定した長さの BYTE 列を書き込みます。

第２章　関数仕様

２－１　ライブラリ使用方法

ライブラリを使用したアプリケーション開始、アプリケーション終了のフローチャートを以下に示します。

２－１－１　アプリケーション開始

ライブラリを使用したアプリケーション開始のフローチャートを以下に示します。

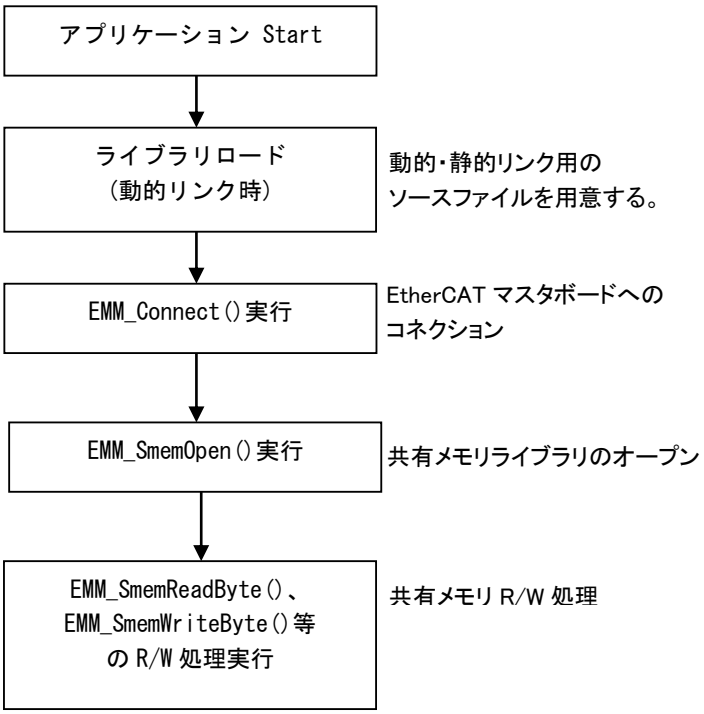


図 2-1-1. アプリケーション開始フローチャート

ライブラリロード後、コネクション、共有メモリのオープンを行うことで共有メモリへ
アクセス可能となります。

２－１－２　アプリケーション終了

ライブラリを使用したアプリケーション終了のフローチャートを以下に示します。

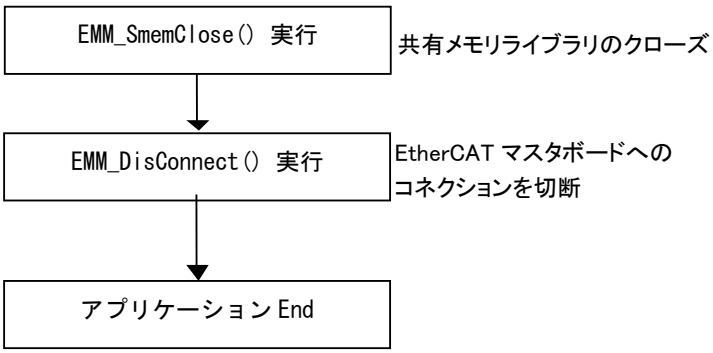


図 2-1-2. アプリケーション終了フローチャート

２－２　EtherCAT マスタボード　アクセス関数

EMM_Connect 関数

機能　マスタボードと接続します。

書式　int EMM_Connect (int Board, const char *pIpAddress, unsigned int port);

引数　Board：
　ボード番号(0-3)
pIpAddress：
　マスタボードの IP アドレスを格納したポインタ
port：
　マスタボードのポート番号

戻り値　MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_ALREADYCONNECT：
　既に接続済みです。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　本ライブラリを初期化し、指定した IP アドレス、ポート番号のマスタボードとコネクション接続します。
本関数をコール後、マスタボードとコマンドのやり取りをすることができます。

EMM_DisConnect 関数

機能　　マスタボードと切断します。

書式　　int EMM_DisConnect (int Board);

引数　　Board：
　　ボード番号(0-3)

戻り値　　MOTION_ER_OK：
　　成功。
MOTION_ER_INVALIDPARAM：
　　無効な引数。
MOTION_ER_NOTOPEN：
　　オープンされていません。

説明　　マスタボードとコネクションを切断します。
本関数をコール後、マスタボードとコマンドのやり取りを行うことができなくなります。

EMM_SmemOpen 関数

機能　マスタボードの共有メモリをオープンします。

書式　unsigned int EMM_SmemOpen (int Board, const char *pShmemName);

引数　Board：
　ボード番号(0-3)
pShmemName：
　マスタボード上の共有メモリ名を格納したポインタ

戻り値　共有メモリのハンドル(値 0 以上)：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTCONNECT：
　マスタボードと接続されていません。
ER_MEMORY_ALREADYOPEN：
　既にオープン済みです。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　共有メモリ名で指定したマスタボード上の共有メモリをオープンします。
本ライブラリが成功した場合、共有メモリのハンドルを取得することができます。
本関数をコール後、取得したハンドルを使用し、Read/Write 関数で共有メモリを読み書きすることができます。

EMM_SmemClose 関数

機能　マスタボードの共有メモリをクローズします。

書式　int EMM_SmemClose (int Board, unsigned int handle);

引数　Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル

戻り値　MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　引数の共有メモリハンドルで指定した共有メモリをクローズします。
本関数をコール後、取得した共有メモリのハンドルでは共有メモリの Read/Write は使用できません。

EMM_SmemIsMemSize 関数

機能　　共有メモリのサイズを確認します。

書式　　int EMM_SmemIsMemSize (int Board, unsigned int handle);

引数　　Board：
　　ボード番号(0-3)
handle：
　　共有メモリのハンドル

戻り値　　共有メモリのサイズ(値０以上)：
　　成功。
MOTION_ER_INVALIDPARAM：
　　無効な引数。
MOTION_ER_NOTCONNECT：
　　マスタボードと接続されていません。
MOTION_ER_NOTOPEN：
　　オープンされていません。
MOTION_ER_OPENDEVICE：
　　マスタボードの接続に失敗しました。

説明　　共有メモリのサイズを取得します。
本ライブラリが成功した場合、戻り値に引数の共有メモリハンドルで指定した
共有メモリのサイズが取得できます。

EMM_ShutdownNotify 関数

機能　　マスタボードにシャットダウンを通知します。

書式　　int EMM_ShutdownNotify(int Board);

引数　　Board：
　　ボード番号(0-3)

戻り値　　MOTION_ER_OK：
　　成功。
MOTION_ER_INVALIDPARAM：
　　無効な引数。
MOTION_ER_NOTOPEN：
　　オープンされていません。
MOTION_ER_OPENDEVICE：
　　マスタボードの接続に失敗しました。

説明　　マスタボードにシャットダウンを通知し、設定の保存を要求します。

EMM_SetDateTime 関数

機能　　システム日時を設定します。

書式　　int EMM_SetDateTime (int Board, LPSYSTEMTIME lpSystemTime);

引数　　Board：
　　ボード番号(0-3)
lpSystemTime：
　　SYSTEMTIME 構造体を格納するポインタ

戻り値　　MOTION_ER_OK：
　　成功。
MOTION_ER_INVALIDPARAM：
　　無効な引数。
MOTION_ER_NOTOPEN：
　　オープンされていません。
MOTION_ER_OPENDEVICE：
　　マスタボードの接続に失敗しました。

説明　　マスタボードのシステム日時を設定します。

EMM_Reset 関数

機能　　マスタボードをリセットします。

書式　　int EMM_Reset (int Board);

引数　　Board：
　　ボード番号(0-3)

戻り値　　MOTION_ER_OK：
　　成功。
MOTION_ER_INVALIDPARAM：
　　無効な引数。
MOTION_ER_NOTOPEN：
　　オープンされていません。
MOTION_ER_OPENDEVICE：
　　マスタボードの接続に失敗しました。

説明　　マスタボードにリセット要求をします。
成功した場合、マスタボードがリセットされます。

EMM_GetVersion 関数

機能　　バージョンを取得します。

書式　　int EMM_GetVersion (int Board, unsigned short *hver, unsigned short *sver);

引数　　Board：
　　ボード番号(0-3)
hver：
　　ハードウェアバージョンを格納するポインタ
sver：
　　ソフトウェアバージョンを格納するポインタ

戻り値　　MOTION_ER_OK：
　　成功。
MOTION_ER_INVALIDPARAM：
　　無効な引数。
MOTION_ER_NOTOPEN：
　　オープンされていません。
MOTION_ER_OPENDEVICE：
　　マスタボードの接続に失敗しました。

説明　　マスタボードのハードウェアバージョンとソフトウェアバージョンを
取得します。

EMM_GetBoardNumber 関数

機能　ボードナンバー(基板識別番号）を取得します。

書式　int EMM_GetBoardNumber (int Board, unsigned short *boardnumber);

引数　Board：
　　ボード番号(0-3)
boardnumber：
　　ボードナンバーを格納するポインタ

戻り値　MOTION_ER_OK：
　　成功。
MOTION_ER_INVALIDPARAM：
　　無効な引数。
MOTION_ER_NOTOPEN：
　　オープンされていません。
MOTION_ER_OPENDEVICE：
　　マスタボードの接続に失敗しました。

説明　マスタボードのボードナンバーを取得します。

EMM_SmemReadByte 関数

機能　共有メモリから 1BYTE(uint8_t)のデータを読み出します。

書式
```
int EMM_SmemReadByte (
    int Board,
    unsigned int handle,
    unsigned int address,
    BYTE *pdata
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
pdata：
　取得データを保存するポインタ

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address からデータを 1BYTE(uint8_t)読み出します。

EMM_SmemReadWord 関数

機能　共有メモリから 1WORD(uint16_t)のデータを読み出します。

書式
```
int EMM_SmemReadWord (
    int Board,
    unsigned int handle,
    unsigned int address,
    WORD *pdata
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
pdata：
　取得データを保存するポインタ

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address からデータを 1WORD(uint16_t)読み出します。

EMM_SmemReadDWord 関数

機能　共有メモリから 1DWORD(uint32_t)のデータを読み出します。

書式
```
int EMM_SmemReadDWord (
    int Board,
    unsigned int handle,
    unsigned int address,
    DWORD *pdata
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
pdata：
　取得データを保存するポインタ

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address からデータを 1DWORD(uint32_t)読み出します。

EMM_SmemReadLWord 関数

機能　共有メモリから 1DWORDLONG(uint64_t)のデータを読み出します。

書式
```
int EMM_SmemReadLWord (
    int Board,
    unsigned int handle,
    unsigned int address,
    DWORDLONG *pdata
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
pdata：
　取得データを保存するポインタ

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address からデータを 1DWORDLONG(uint64_t)読み出します。

EMM_SmemReadBytes 関数

機能　共有メモリから BYTE 列(uint8_t*)のデータを読み出します。

書式
```
int EMM_SmemReadBytes (
    int Board,
    unsigned int handle,
    unsigned int address,
    BYTE *pdata,
    unsigned int len
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
pdata：
　取得データを保存するポインタ
len：
　取得データ長

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address から取得データ長分、
取得データを読み出します。

EMM_SmemWriteByte 関数

機能　共有メモリへ 1BYTE(uint8_t)のデータを書き込みます。

書式
```
int EMM_SmemWriteByte (
    int Board,
    unsigned int handle,
    unsigned int address,
    BYTE data
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
data：
　書き込みデータ

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address へデータを 1BYTE(uint8_t)書き込みます。

EMM_SmemWriteWord 関数

機能　共有メモリへ 1WORD(uint16_t)のデータを書き込みます。

書式
```
int EMM_SmemWriteWord (
    int Board,
    unsigned int handle,
    unsigned int address,
    WORD data
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
data：
　書き込みデータ

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address へデータを 1WORD(uint16_t)書き込みます。

EMM_SmemWriteDWord 関数

機能　共有メモリへ 1DWORD(uint32_t)のデータを書き込みます。

書式
```
int EMM_SmemWriteDWord (
    int Board,
    unsigned int handle,
    unsigned int address,
    DWORD data
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
data：
　書き込みデータ

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address へデータを 1DWORD(uint32_t)書き込みます。

EMM_SmemWriteLWord 関数

機能　共有メモリへ 1DWORDLONG(uint64_t)のデータを書き込みます。

書式
```
int EMM_SmemWriteLWord (
    int Board,
    unsigned int handle,
    unsigned int address,
    DWORDLONG data
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
data：
　書き込みデータ

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address へデータを 1DWORDLONG(uint64_t)書き込みます。

EMM_SmemWriteBytes 関数

機能　共有メモリへ BYTE 列(uint8_t*)データを書き込みます。

書式
```
int EMM_SmemWriteBytes (
    int Board,
    unsigned int handle,
    unsigned int address,
    BYTE *pdata,
    unsigned int len
);
```

引数
Board：
　ボード番号(0-3)
handle：
　共有メモリのハンドル
address：
　共有メモリのアドレス
pdata：
　書き込みデータを保存するポインタ
len：
　書き込みデータ長

戻り値
MOTION_ER_OK：
　成功。
MOTION_ER_INVALIDPARAM：
　無効な引数。
MOTION_ER_NOTOPEN：
　オープンされていません。
MOTION_ER_OPENDEVICE：
　マスタボードの接続に失敗しました。

説明　ハンドルで指定した共有メモリの指定した address へ取得データ長分、
BYTE 列データを書き込みます。

このリファレンスマニュアルについて

（1）本書の内容の一部または全部を当社からの事前の承諾を得ることなく、無断で複写、複製、掲載することは固くお断りします。
（2）本書の内容に関しては、製品改良のためお断りなく、仕様などを変更することがありますのでご了承下さい。
（3）本書の内容に関しては万全を期しておりますが、万一ご不審な点や誤りなどお気付きのことがございましたらお手数ですが巻末記載の弊社もしくは、営業所までご連絡下さい。その際、巻末記載の書籍番号も併せてお知らせ下さい。

７７Ａ８２００００５Ａ　　２０２１年　１月　初版

株式会社アルゴシステム

本社
〒５８７－００２１　大阪府堺市美原区小平尾６５６番地
ＴＥＬ（０７２）３６２－５０６７
ＦＡＸ（０７２）３６２－４８５６

ホームページ　http://www.algosystem.co.jp/