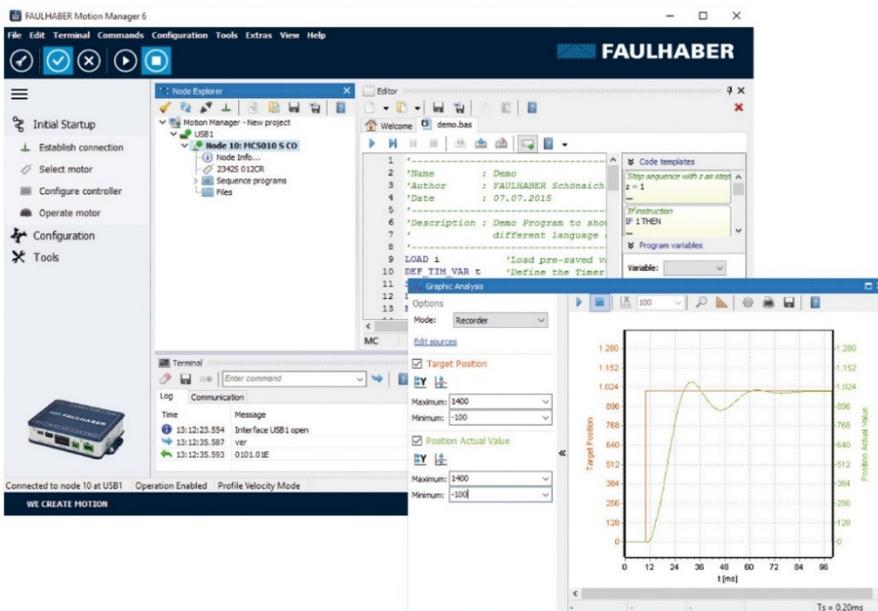


FAULHABER

ソフトウェアマニュアル

FAULHABER Motion Manager 6



JP

WE CREATE MOTION

インプリント

バージョン:

第3版、24-04-2017

Copyright

Dr. Fritz Faulhaber GmbH & Co. KG

Daimlerstr.23 / 25 · 71101 Schönaich

翻訳を含む転載禁止。

本マニュアルの全部または一部を、Dr. Fritz Faulhaber GmbH & Co. KGの明示的な書面による同意なしに、情報システムへ複製、再生、保存、あるいは他の形式に加工または転送することは禁止されています。

本マニュアルは細心の注意を払って作成されています。

ただし、Dr. Fritz Faulhaber GmbH & Co. KGは本マニュアルの誤記および誤記によって発生した結果に対して何ら責任を負わないものとします。同様に、機器の不適切な使用による直接的損失および結果的損害に関しても何ら責任を負わないものとします。

本ソフトウェアを使用する際には、安全工学および干渉抑制に関連する規制ならびに本マニュアルに記載された要件に留意し遵守する必要があります。

仕様は予告なしに変更されることがあります。

最新版のテクニカルマニュアルは、以下のFAULHABERのインターネットサイトから入手できます：
www.faulhaber.com

目次

1	本マニュアルについて	6
1.1	本マニュアルの目的	6
1.2	追加文書	6
1.3	本書の使用	7
1.4	略語一覧	7
1.5	本マニュアルで使用する記号とマーク	8
2	安全	9
2.1	使用目的	9
2.2	安全上の注意事項	10
3	製品概要	11
4	インストール	12
4.1	システム要件	12
4.2	Motion Manager 6のインストール	12
4.3	Motion Manager 6のアップデート	13
4.3.1	バージョン5.xx以前のアップデート	13
4.3.2	バージョン6.xxのアップデート	13
4.4	Motion Manager 6のアンインストール	13
4.5	Callパラメータ	14
5	ユーザインターフェース	15
5.1	メニューバー	16
5.2	ツールバー	17
5.3	クイックアクセス	17
5.4	Node Explorer	18
5.4.1	プロジェクトの管理	19
5.4.2	接続の管理	19
5.4.3	ネットワークの構成	21
5.4.4	ノードの検索	21
5.5	ターミナル	22
5.5.1	コマンドの入力	22
5.5.2	アクティビティログ	23
5.5.3	通信履歴	23
5.5.4	検索機能	23
5.6	エディタ	24
5.7	Motion Managerの構成	25
6	コントローラのコミッショニング	26
6.1	接続の確立	26
6.2	モータの選択	26
6.3	コントローラの設定	27

6.4	モータの運転	27
6.5	拡張機能	28
7	コントローラ固有の機能	29
7.1	MC V2.x / SCコントローラファミリ	29
7.1.1	モータ選択ウィザード	29
7.1.2	モーションコントローラの構成	31
7.1.2.1	駆動機能	31
7.1.2.2	コントローラパラメータ	32
7.1.2.3	接続パラメータ	33
7.1.3	スピードコントローラの構成	33
7.1.4	コントローラのチューニング	33
7.1.5	追跡機能	35
7.1.5.1	追跡設定	35
7.1.6	状況表示	36
7.1.7	シーケンスプログラム	36
7.1.8	パラメータファイル	38
7.2	MC V3.x コントローラファミリ	39
7.2.1	モータ選択ウィザード	39
7.2.2	コントローラ構成ウィザード	40
7.2.3	駆動機能の設定	41
7.2.4	コントローラパラメータの変更	41
7.2.5	接続パラメータ	42
7.2.6	モーションコックピット	42
7.2.7	コントローラのチューニング	43
7.2.8	追跡機能	45
7.2.8.1	追跡設定	46
7.2.9	状況表示	46
7.2.10	シーケンスプログラム	47
7.2.11	周波数応答の測定	48
7.3	CANopenの標準機能	49
7.3.1	NMT	49
7.3.2	機器制御	49
7.3.3	SDO/PDO	49
7.3.3.1	オブジェクトブラウザ	50
7.3.3.2	PDOマッピング	51
7.3.4	LSS	52
7.4	グラフィック解析	53
7.4.1	メインツールバー	53
7.5	マクロ機能	54
7.5.1	マクロの編集	54
8	追加機能	55
8.1	VB Scriptプログラム	55
8.1.1	スクリプトプログラムの作成	55
8.1.2	スクリプトプログラムの開始と停止	55

8.1.3	Motion Managerの一般的な機能	56
8.1.3.1	Sendコマンド	56
8.1.3.2	WaitAnswer	57
8.1.3.3	WriteToHistory	57
8.1.3.4	CloseCom	58
8.1.3.5	OpenCom	58
8.1.3.6	CmdExecute	58
8.1.3.7	UpdateWindows	59
8.1.4	オブジェクトディクショナリを使用したモーションコントローラの関数	59
8.1.4.1	GetObj	59
8.1.4.2	SetObj	60
8.1.4.3	SetStrObj	60
8.1.5	MC V2.xファミリのRSインターフェース付きモーションコントローラの関数	61
8.1.5.1	ComXonXoff	61
8.1.5.2	SendBin	61
8.1.5.3	SetBinMode	61
8.1.5.4	BinRequest	62
8.2	ファームウェアのアップデート	62
8.3	オフラインモード	63
8.3.1	モーションコントローラ	63
8.3.1.1	仮想モーションコントローラへの接続の確立	63
8.3.1.2	仮想モーションコントローラの管理	63
8.3.2	スピードコントローラ	64
9	保証	65
10	付録	66
10.1	Motion Managerコマンドリファレンス	66
10.1.1	CANopenステートマシンの制御	66
10.1.2	オブジェクトディクショナリのオブジェクトの記述	68
10.1.3	オブジェクトディクショナリのオブジェクトの読み込み	69
10.1.4	オブジェクトディクショナリでの任意のオブジェクトの書き込み	70
10.1.5	オブジェクトディクショナリでの任意のオブジェクトの読み込み	70
10.1.6	信号の送信	71
10.2	問題の解決法	72
10.2.1	ポートが存在しない	72
10.2.2	ポートを開くことができない	72
10.2.3	接続された機器に接続できない	72
10.2.4	モータが始動しない	72
10.3	エンドユーザライセンス契約	73

1 本マニュアルについて

1.1 本マニュアルの目的

本マニュアルは、FAULHABER Motion Managerのインストールおよび使用方法について記載しています。Motion Managerは、下記のコントローラに対応します。

MC V3.0ファミリ

- MC5010 S RS/CO/ET
- MC5005 S RS/CO/ET
- MC5004 P RS/CO/ET
- MCSシリーズ

MC V2.0/V2.5ファミリ

RSインターフェース (RS232) : CFインターフェース (FAULHABER CANで使用するCANopen) : COインターフェース (CANopen) :

- | | | |
|----------------|----------------|----------------|
| ■ MCDC 300x RS | ■ MCDC 300x CF | ■ MCDC 300x CO |
| ■ MCBL 300x RS | ■ MCBL 300x CF | ■ MCBL 300x CO |
| ■ MCLM 300x RS | ■ MCLM 300x CF | ■ MCLM 300x CO |
| ■ CS/CSDシリーズ | ■ CC/CCDシリーズ | ■ CO/CODシリーズ |

SCファミリ

- SCxxxx
- BX4 SCシリーズ
- BRCシリーズ
- BL Flat SCシリーズ

本マニュアルは、対応する製品の操作を許可されトレーニングを受けた専門技術者を対象としています。

本マニュアルの全ての情報は、標準バージョンの上記のコントローラを基本に説明しています。顧客固有のバージョンについては添付書類を参照してください。

1.2 追加文書

FAULHABER製品の初期設定および運転時の操作に関する追加情報は、以下のマニュアルを参照してください。マニュアルは以下のサイトからpdf形式でダウンロードできます : www.faulhaber.com/manuals/

1.3 本書の使用

本マニュアルは、オンラインヘルプ形式の電子版のドキュメントで、Motion Manager 6に同梱されています。専用のヘルプボタンやF1キーでディスプレイに表示させることができます。

- 通信を構成する前に、「安全」の章（2章 9ページ参照）を中心に本マニュアルをよくお読みください。
- 本マニュアルは、製品の使用期間にわたって、オペレータがいつでも手に取って読むことができる場所に保管してください。
- また、製品を譲渡する際には製品と一緒に本マニュアルも次の所有者に渡してください。

1.4 略語一覧

略語	意味
ASCII	情報交換用米国標準コード
CAN	コントローラエリアネットワーク
CiA	CAN in Automation e.V.
CSV	カンマ区切り
DLL	ダイナミックリンクライブラリ
LSS	レイヤー設定サービス
NMT	ネットワーク管理オブジェクト
OD	オブジェクトディクショナリ
PDO	プロセスデータオブジェクト
RxPDO	受信プロセスデータオブジェクト（ドライブから受信されたPDO）
SDO	サービスデータオブジェクト
PLC	プログラマブルロジックコントローラ
TxPDO	送信プロセスデータオブジェクト（ドライブから送信されたPDO）
USB	ユニバーサルシリアルバス
VB	ビジュアルベーシック

1.5 本マニュアルで使用する記号とマーク

注意!

損傷の危険

- ▶ 上記の状況を回避するための措置

駆動に対する操作手順、使用方法のヒントや推奨事項

- ✓ 必要な操作の前提条件

1. 必要な操作の最初の手順

-  手順の結果

2. 必要な操作の次の手順

-  操作の結果

- ▶ 単発の操作

2 安全

2.1 使用目的

FAULHABER Motion Managerは、FAULHABER製ドライブシステムの構成および設定をサポートします。本ソフトウェアは、生産運転における駆動システムの稼働を目的として設計されているわけではありません。

機種によっては、コントローラをさまざまなインターフェースを介してPCと接続できますが、一部の制御機能では、プログラミングアダプタが必要になる場合もあります。

 各駆動コントローラの操作説明書に記載されたパラメータ化および設定に関する指示に、常に順守してください。

FAULHABER Motion Managerは主に、以下の使用を前提としています。

- FAULHABER製ドライブシステムのモーションコントローラおよびスピードコントローラの構成および設定。
- 対応するインターフェースを介した機器制御への通信
- ドライブおよびコントローラの設定サポート
- グラフィカルインターフェースの、またはコマンド入力による、サポートされ動作モードでのコントローラの操作
- VB Scriptプログラムまたは統合されたマクロ機能を使用した、駆動シーケンスの作成および実行
- ドライブシステムの動きをグラフィカルに解析（追跡機能）
- コントローラデバイスで実行されるシーケンスプログラムの作成、転送および管理（デバッグオプションを含む）
- パラメータファイルのアップロードおよびダウンロード

 上記の機能は、全てのコントローラで利用できるとは限りません。コントローラの構成によっては使用できない機能もあります。

下記のインターフェースに対応しています。

- RS232 (COMx)
- CAN (対応インターフェース: IXXAT、Peak、ESD、EMS、その他ご要望に応じて対応可能)
- USB

 スピードコントローラを使用する場合は、RS232またはUSBに対応したプログラミングアダプタを使用する必要があります。

 RS232インターフェース付きのモーションコントローラは、直接、またはUSBシリアル変換アダプタを使用してUSB経由でCOMxに接続できます。

2.2 安全上の注意事項

注意!

 駆動機器が正しく設定されていない場合、モーションコントローラが損傷する可能性があります。

- ▶ ソフトウェアマニュアルの指示に従ってください。

3 製品概要

Motion Managerを使用すると、接続されたコントローラの設定やパラメータに簡単にアクセスできます。

コントローラの初期設定では、専用のウィザードが使用できます。選択したインターフェースで検出されたドライブレユニットが、ツリービューで表示されます。インターフェースや表示の設定は、プロジェクトファイルに保存することができ、ユーザのアクションおよび通信のフローが記録できます。

Motion Managerを使用すると、保存や実行のシーケンスプログラムの作成や編集ができます。作成したプログラムをコントローラに転送したり機器で実行することができます。エラー検出およびプログラムフローの監視も利用可能です。

コントローラの操作およびモーションタスクは、以下を介して実行します：

- グラフィカルなユーザインターフェース
- コマンド入力
- マクロ機能
- VBScript（ビジュアルベーシックスクリプト）を使用したシーケンスプログラミング

グラフィカルな解析機能により、制御パラメータを記録できます。追加のツールを使用して、コントローラパラメータを作成し最適化できます。

4 インストール

4.1 システム要件

- オペレーティングシステム: Microsoft Windows 7以降
- ハードディスクの空き容量: 100MB

4.2 Motion Manager 6のインストール

1. セットアップファイルを実行します。
2. 使用する言語を選択します。
3. インストールプロセスを確認し、[**Next**]をクリックします。
4. エンドユーザライセンス契約の全項目を読みます。
5. 同意する場合は、[**Next**]をクリックして確認します。
6. 必要に応じて、ソフトウェアのインストール場所を変更します。
7. [**Next**]をクリックして、インストール場所を確認します。
8. 必要に応じてプログラムリンクの条件を変更し、追加リンクを作成します。
9. [**Next**]をクリックして、選択を確認します。
10. 表示された選択の概要を確認します。
11. [**Installation**]をクリックして、インストールを開始します。
 - 👉 インストールが実行されます。
12. インストールの終了と同時にソフトウェアを起動するかどうかを選択します。
13. [**Finish**]をクリックして、インストールを終了します。
 - 👉 これで、Motion Managerがインストールされます。

4.3 Motion Manager 6のアップデート

4.3.1 バージョン5.xx以前のアップデート

Motion Manager 6のインストールは、既にインストールされたバージョン5.xxを上書きすることもなく、1つのPCに2つのバージョンを共存させることができます。

インストールプロセスは、Motion Manager 6のインストールプロセスと同じです（4.2節、12ページ参照）。

4.3.2 バージョン6.xxのアップデート

Motion Manager 6の古いバージョンがインストールされている場合は、新しいバージョンをインストールすると、旧バージョンがアップデートされます。[Extras]メニューの項目[Options]を使用すると、自動または手動のオンラインアップデート機能を有効にできます。ソフトウェアを常に最新状態にするにはインターネット接続が必要です。

4.4 Motion Manager 6のアンインストール

1. Windowsコントロールパネルの[プログラムと機能]または[ソフトウェア]を選択します。
2. 表示されたリストから[FAULHABER Motion Manager 6]を選択します。
3. [Uninstall]をクリックして選択し、[Delete]を選択して確認します。
 - 👉 Motion Manager 6のアンインストールを確認するダイアログが表示されます。
4. [YES]をクリックして、確認します。
 - 👉 これで、Motion Managerがアンインストールされます。

4.5 Callパラメータ

Moman6.exeプログラムファイルには、さまざまなcallパラメータが含まれ、コマンドラインまたはカスタマイズされたリンクによりソフトウェアを起動すると使用できます。Moman6.exeを呼び出す場合は、callパラメータを単独、または組み合わせて指定できます。

i プログラムファイルは、Motion Managerインストールファイル（作業フォルダ）から呼び出す必要があります。

パラメータ	機能
/ENG	ドイツ語がデフォルト言語として設定されている場合は、Motion Managerを英語のユーザインターフェイスで起動します。
/DEU	英語がデフォルト言語として設定されている場合は、Motion Managerをドイツ語のユーザインターフェイスで起動します。
/RUN:script.vbs	Motion Managerの起動直後に、指定されたVBScriptプログラムを実行します。
/H	/RUN:が実行されMotion Managerをバックグラウンドで展開します。
E	スクリプトが完全に実行されると同時に、/RUN:が実行されMotion Managerが終了します。

例: 英語版インターフェイスからドイツ語版インターフェイスの切り替え

Moman6.exe /DEU

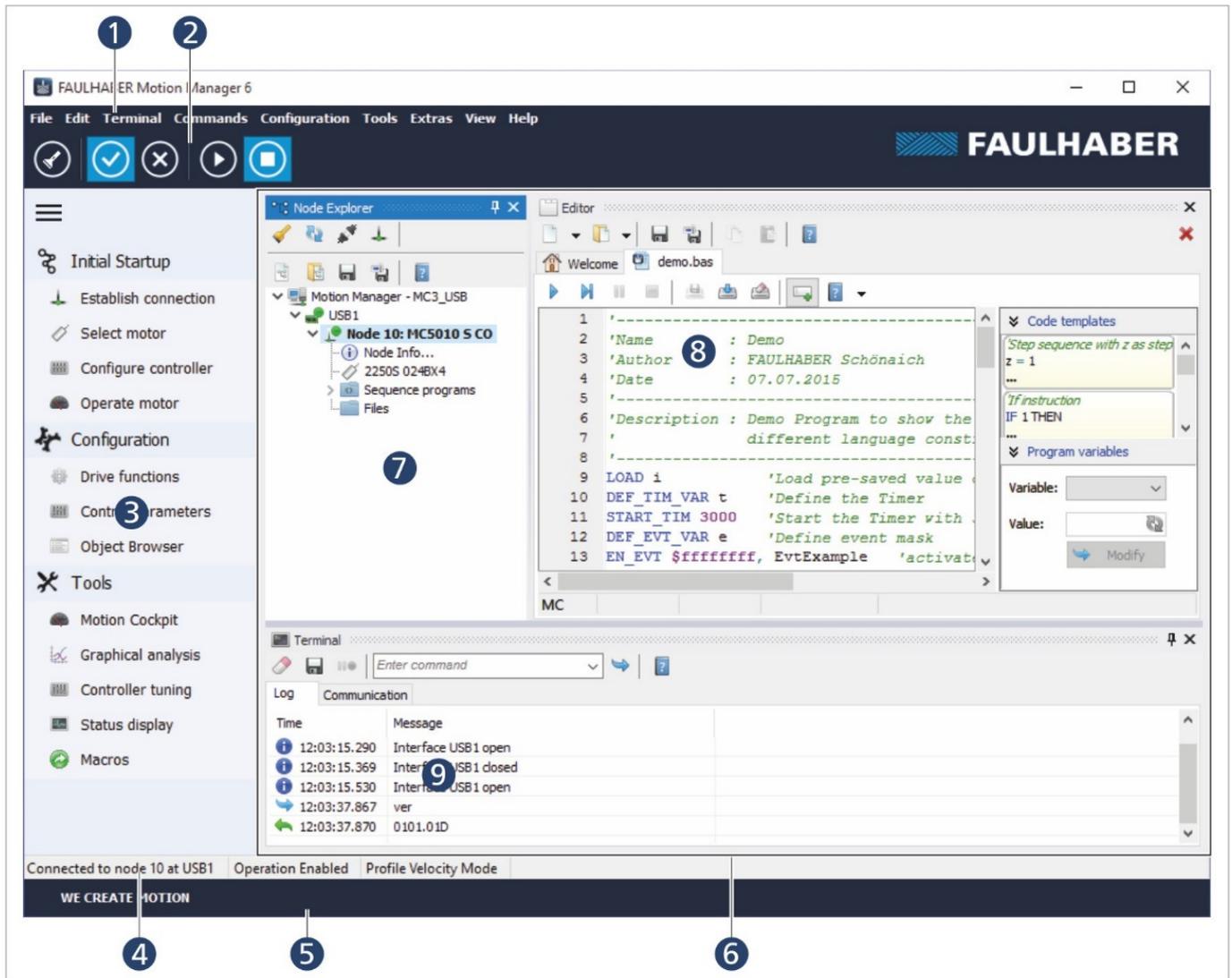
i callパラメータでリンクを作成する場合は、「Target」属性フィールドの呼び出しラインを次の形式で既述する必要があります（例）。

"...\Faulhaber\Motion Manager 6\Moman6.exe" /ENG

プログラム名とパスの前後のコーテーションマークに注意してください。

5 ユーザーインターフェース

ユーザーインターフェースは、さまざまなタスクが実行できる複数のウィンドウで構成されます。ウィンドウには、ポップアップのように独立している画面とメインウィンドウの一部として表示される画面があります。



標準的なユーザーインターフェースは、以下の部分で構成されています。

- メニュー (1) とツールバー (2) があるヘッダ
- クイックアクセス (ツリー構造) (3)
- ステータスバー (4)
- フッター (5)
- ドッキングエリア (6)

クラシックビュー (メニュー[View] - [Layout] - [Classic view]で設定します) では、ヘッダは縮小され、クラシックツールバーのメニューのみが表示されます。フッターは表示されません。これは、画面をより広く使用できることを意味します。

接続されたコントローラのタイプに関係なく、以下のウィンドウが常に使用できます。

- Node Explorer (7)
- Editor (8)
- Terminal (9)

Node Explorerで選択したコントローラのタイプに応じて、Graphical Analysis、Object Browser、Motion Cockpit など、他のウィンドウが利用できるようになりました。

各ウィンドウに表示された情報は、マウスを使用してメインウィンドウのドッキングエリアにドラッグドロップ（ドラッグ&ドック）することができます。

ドックできる全てのウィンドウは、メニュー項目[View]の下に表示され、そこから呼び出すことができます。ウィンドウは、水平または垂直に配置、またはタブとして配置することもできます。配置を変更するには、各ウィンドウをドラッグすると表示される上下左右の使用するエリアマークにウィンドウをドラッグします。

画面の端にあるドッキングエリアのウィンドウは、右端のピンのアイコンで非表示にできます。非表示になったウィンドウのタブは、ドッキングエリアのそれぞれの端に表示されます。タブをクリックするとウィンドウが開きます。このウィンドウは、マウスを使用して外部の要素を選択すると自動的に閉じます。

Motion Managerを終了すると、現在のレイアウトが保存され、再起動したときに再び同じ状態で表示されます。また、[View] - [Layout]メニューを使用して、現在の画面のレイアウトをファイルに保存し、いつでもこのファイル設定を呼び出すことができます。

5.1 メニューバー

i メニューバーには、Motion Managerの操作に必要な全ての機能およびコマンドが含まれています。内容は、現在選択されているコントローラにより異なり、コントローラに対応する機能のみが使用可能になります。

ただし、以下の機能は常にメニューバーに表示されます。

機能	説明
File	プログラムファイルおよびプロジェクトファイルを管理するデフォルト機能
Edit	ファイルを編集するデフォルト機能
Terminal	インターフェースおよび接続されたノードを管理する機能
Extras	追加機能
View	ウィンドウのレイアウトおよび表示されるウィンドウの設定
Help	オンラインヘルプおよび他のサポート機能へのアクセス

5.2 ツールバー

ツールバーは、メニューバーの下に表示されます。ツールバーには、使用頻度の高い機能に素早くアクセスできるボタンが表示されています。

ボタン

ボタン	意味	説明
	ノード検索	接続されたコントローラのノードまたはネットワークノードの指定されたインターフェースを検索します。
	有効	コントローラの出カステージをオンに切り替えます。
	無効	コントローラの出カステージをオフに切り替えます。
	実行	コントローラでシーケンスプログラムをロード、または開始します。
	停止	コントローラで実行中のシーケンスプログラムを停止します。

5.3 クイックアクセス

クイックアクセスツールバーを使用して、重要な機能にアクセスできます。クイックアクセスツールバーは畳み込むことができます。接続されたコントローラに応じて、ツリー展開も動的に構成され、以下の3つのエリアに分割されています。

エリア	内容
初期設定	駆動コントローラの初期設定ウィザードとダイアログ
構成	各駆動タスクに応じた駆動機器の設定およびパラメータの作成ができるダイアログ
ツール	起動装置の操作および解析に使用する追加のツール

5.4 Node Explorer

Node Explorerは、Motion Managerによって通信機能が確立されている全てのコントローラを表示します。Node Explorerには、プロジェクトの構造も表示されます。さらには、Node Explorerにプロジェクトを表示します。

ボタン

ボタン	意味	説明
	ノード検索	接続された駆動機器のノードまたはネットワークノードが指定されたインターフェースを検索します。
	アクティブなノードのアップデート	アクティブなノードはノードエクスプローラでアップデートされます。
	接続の確立	接続設定の変更（6.1節、26ページ参照）。
	接続の切断／確立	既存の接続を切断または再度確立します。
	新規プロジェクト	新しいプロジェクトを作成します。
	プロジェクトを開く	既存のプロジェクトを開きます。
	プロジェクトの保存	現在のプロジェクトを保存します。
	名前を付けてプロジェクト保存	現在のプロジェクトを新しく保存します。

Node Explorerは、プロジェクトが3つのレベルで分かれています。

- ポートレベル（COM、CAN、USB）
- ノードレベル（検出済みのコントローラのノードまたはネットワークノード）
- プロジェクトレベル（ノード情報および関連ファイル）

マウスを右クリックすると、コンテキストメニューが表示され追加機能が選択できます。

インターフェース（ポート）およびノードには、さまざまなステータスがあります。

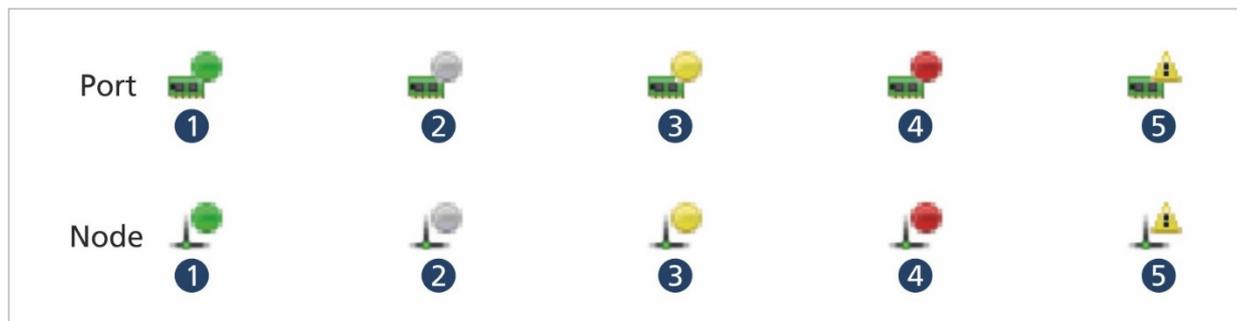


図1: ポートとノードのステータス

- | | | | |
|---|-------------|---|-------------|
| 1 | アクティブ、オンライン | 4 | 不良ステータス |
| 2 | アクティブでない | 5 | スキャンによる検出なし |
| 3 | アクティブ、オフライン | | |

ネットワークノードは、ダブルクリックすると選択され、アクティブ状態になります。ノードの属性（状態）に応じて、表示されるアイコンが変わります。デフォルト設定では、コマンドがこのノードのノード番号とともに送信されます。

[Node info]エントリをダブルクリックすると別のウィンドウが表示され、選択したノードに関する追加情報（ソフトウェアバージョンまたはシリアル番号など）が確認できます。

コントローラはFAULHABER製モータが設定されている場合、そのモータの名前が表示されます。それ以外の場合、[Select motor]エントリが表示されます。このエントリをダブルクリックすると、FAULHABER製モータの設定および選択のためのモータ選択ウィザードが表示されます。

[Files]フォルダでは、この機器に接続されたファイルへのリンクを管理します。たとえば、パラメータファイル、データシートまたはドキュメントなどです。Motion Managerのエディタで開いているファイルは、ファイルフォルダにドラッグ&ドロップできます。コンテキストメニューでファイルを追加することができます。

5.4.1 プロジェクトの管理

Node Explorerでは、プロジェクトファイルの現在のプログラム設定を管理します。

プロジェクトファイルを保存すると、次の情報が記録されます：

- 全てのアクティブ化されたインターフェースの設定（プロトコル、ポート、チャンネル、ボーレート、スキャン範囲）
- 表示されているノードの情報（ノード番号、シリアル番号、名前）
- 各ノードの追加されたファイルへのリンク
- 各ノードのプログラムの設定（トレース設定など）

プロジェクトファイルがロードされると、保存されている情報の再構築が試行されます。プロジェクトファイルに保存されたノードが検出されなかった場合は、Node Explorerに表示された対象のノードのよこに？マークが付きます。ノードは、コンテキストメニューまたは「Delete」キーを使用してNode Explorerから削除されると、プロジェクトファイルからも削除されます。

新しいプロジェクトを作成する場合は、通信インターフェースを最初に指定し、接続されているネットワークノードを検索する必要があります。必要に応じて、付加的なインターフェースを追加できます（[Wizard for establishing a connection]または[Terminal] - [Manage connection...]メニュー）。ノードエクスプローラに表示されるノードは、保存するとプロジェクトファイルに書き込まれます。

Motion Managerは常に、タイムスタンプが新しいプロジェクトファイルからスタートします。

5.4.2 接続の管理

[Manage connections]ダイアログを使用すると、接続の追加、削除および設定など、以前に設定された通信接続の管理ができます。

この機能は、[Terminal]メニューまたはNode Explorerのツールバーから呼び出すことができます。

 接続の確立ウィザードを使用すると、接続の設定がより簡単にできます。

ウィザードのウィンドウは、ツリー構造のナビゲーションエリアと入力エリアに分かれています。起動したウィザードの最初の画面には、現在の接続設定が表示されます。

入力エリアのノードのチェックボックスを選択または選択解除して、ツリーにノードを追加、または削除します。

入力エリアには、ナビゲーションエリアのアクティブなノードに応じた入力オプションが常に表示されます。

[OK]をクリックすると、現在のプロジェクトの設定が適用され、選択したインターフェースに接続された機器が自動的に検索されます。

対応インターフェース

Motion Managerが対応する全てのインターフェースの種類を示します。

- COM（シリアルRS232インターフェース）
- CAN
- USB

利用可能なインターフェースのプラグイン

選択されたインターフェースに利用できる全てのインターフェースプラグインを示します。インターフェースプラグインは、Motion Managerとインターフェースドライバとの間で接続を確立します。

デフォルト設定では、次の駆動機器の接続がインストールされます。

インターフェース	プラグイン名	プラグインファイル	機能
COM	Standard COM	Mocom.dll	標準的なシリアルCOMポートへの接続
CAN	Ixxat VCI3	Ixxat_vci3.dll	HMS-IXXAT VCI3- / VCI4 ドライバへの接続
	PEAK PCAN	Peak_pcan.dll	PEAK PCAN ドライバへの接続
	EMS CPC	Ems_cpc.dll	EMS CPC ドライバへの接続
	ESD NTCAN	Esd_ntcan.dll	ESD NTCAN ドライバへの接続
USB	FAULHABER MC V3.x USB	MC3Usb.dll	FAULHABER製MC V3.x USB ドライバへの接続
	FAULHABER SC USB	SCUsb.dll	FAULHABER製SC-USB ドライバへの接続

i COM: Motion ManagerはCOM1からCOM256のシリアルインターフェースに対応しています。システムで利用可能なインターフェースは、自動的に検出されます。USBシリアル変換アダプタは、モーションコントローラのシリアルインターフェースへのアクセスに必要で、対応するドライバもインストールする必要があります。インストールされたドライバは、メーカーのデータを使用して常に最新の状態を維持してください。

CAN: CANインターフェース付きMotion Managerを操作するには、使用するCANカードのドライバをインストールする必要があります（CANインターフェースのメーカーのマニュアルを参照してください）。

USB: USBを使用して駆動機器にアクセスするには、対応するドライバをインストールする必要がありますが、FAULHABER製のコントローラの場合は、Motion Managerのインストール時にドライバもインストールされます。

検出されたポート

選択したインターフェースプラグインを介して検出された、システムの物理ポートを示します。存在するポートが表示されない場合は、システムによって検出されていません（10.2節、72ページ参照）。

i USBシリアル変換アダプタを使用する場合、COMポート番号の割り当ては、Windowsデバイスマネージャに表示でき、必要に応じて変更できます。

選択されたポートの構成

選択されたポートの接続パラメータを設定します。

各機器の通信プロトコルの情報が記載されたDLLファイルが、プロトコルプラグインとして選択されます。

インターフェース	プロトコルプラグイン	意味
COM	MC2RS	グループMC V2.xのモーションコントローラのシリアルプロトコル
	SCRS	RS232を使用するスピードコントローラのプロトコル
	CO_RS232	RS232を使用するMC V3.xファミリのモーションコントローラのCOプロトコル
CAN	CO_CAN	CANを使用する標準的なCANopen
USB	CO_USB	USBを使用するMC V3.xファミリのモーションコントローラのCOプロトコル
	SC_USB	USBを使用するスピードコントローラのプロトコル

i プロトコルの競合を除外できる場合は、単一のインターフェースで複数のプロトコルを使用できます。ネットワークのスキャン中、記載された全てのプロトコルプラグインが正常にロードされ、対応するノードを検索します。ネットワークをスキャンする時は、サポートされているノードを検出するために、リスト化された全てのプロトコルプラグインが順番にロードされます。

転送速度

データ転送の速度（ボーレート）を設定します。設定したボーレートが、接続された全ての機器で対応できることを事前に確認してください。

スキャン範囲

接続された機器を検索するノード番号の範囲を設定します。

5.4.3 ネットワークの構成

Motion Managerを使用すると、一部の駆動機器のネットワークアドレスを個別に指定することも、ネットワークでアドレスを割り当てることもできます。ネットワークを構成する際には、コントローラの通信マニュアルを参照する必要があります。通常、次を満たしていることが必要です。

- 全てのノードに同じボーレートが設定されていること
- 全てのノードに一意のノード番号が割り当てられていること
- ▶ 前述の要件を満たしていることを確認するには、全てのコントローラを個別に構成し、相互に接続を試行してください。

5.4.4 ノードの検索

ノードの検索では、選択されたインターフェースのうち、パラメータの設定を満たす全てのインターフェースが検索されます。スキャン後、検出されない機器のノードもリストに表示されます。必要に応じて、[Del]ボタンまたはコンテキストメニューを使用して、使用しないノードをプロジェクトから削除できます。

5.5 ターミナル

手動でコマンドを入力する場合は、ターミナルを使用します。データ交換、ユーザの動作、ステータスメッセージが記録されます。

ユーザインターフェース	説明
	アクティブなタブの内容を削除します
	ファイルのアクティブなタブの内容を保存します
	アクティブなタブで記録を中断または継続します
	手動でコマンドを入力するフィールド
	入力されたコマンドを送信します

各メッセージは、タイムスタンプとマークとともに表示されます。

マーク	意味
	ノードの検索が開始されました
	ステータス情報
	エラー情報
	コマンドが送信されました
	非同期的にデータを受信します
	同期的にデータを受信します
	非同期エラーメッセージを受信しました
	同期データ交換中のエラー
	スクリプトプログラムからの出力

5.5.1 コマンドの入力

コマンド入力フィールドには、Motion Managerのコマンドを入力できます。コマンドは、対応するプロトコルに変換されコントローラに送信されます（10.1節、66ページの「コマンドリファレンス」を参照）。

インターフェースがRSまたはCFのMC V2.xファミリの機器では、指定されたコマンドをASCII文字として直接、またはCAN信号としてFAULHABERチャンネルを経由して送信します。このフィールドでは、コントローラが対応する全てのコマンドを指定できます。

ただし、SC（スピードコントローラ）機器はコマンド入力に対応していません。

コマンド入力によって送信されたコマンドは、アクティビティログに記録されます。

5.5.2 アクティビティログ

アクティビティログの[Log]タブには、ユーザの動作およびステータス情報が記録されます。

使用できるユーザインターフェース

ユーザインターフェース	機能
	アクティブなタブの内容を削除します
	ファイルのアクティブなタブの内容を保存します

5.5.3 通信履歴

通信履歴では、全てのアクティブなインターフェースを介して実行されたデータ交換の情報とステータス情報が、[Communication]タブに記録されます。

- 入力されたコマンドやデータはMotion Managerによって解釈され送受信されます。[Data]の欄には、送受信されたコマンドやデータが表示されます。
- [Telegram]欄は、16進値として送受信された各信号をデータバイトで表示します。

使用できるユーザインターフェース

ユーザインターフェース	機能
	アクティブなタブの内容を削除します
	ファイルのアクティブなタブの内容を保存します
	アクティブなタブで記録を中断または継続します

5.5.4 検索機能

検索機能を使用すると、ログおよび通信履歴の内容全体を特定の用語で検索できます。

ユーザインターフェース	機能
	キーワードを入力するフィールドキーワードとして入力できる文字に区別はありません。 <ul style="list-style-type: none">■ 「X」は、入力されたキーワードを削除します。■ 「Enter」キーを押すと、キーワードを含む直近の項目が選択されます。
	キーワードを含む前の項目または次の項目に移動できます。

一致する文字列が検索されると、矢印のアイコンの横に対象の行番号が表示されます。

5.6 エディタ

組み込みのエディタにより、さまざまなファイル形式を編集および実行できます。

■ シーケンスプログラム

コントローラ上で保存および実行されるプログラム:

- モーションコントロールファイルMC V3.x (*.bas)
- モーションコントロールファイルMC V2.x (*.mcl)

■ パラメータファイル

読み取りパラメータのセットを保存するためにコントローラに転送されるMC V2.xパラメータファイル:

- パラメータファイルMC V2.x (*.mcp)

■ VB Scriptプログラム

Motion Manager内のPCで作動するプログラム:

- VB Scriptファイル (*.vbs)

■ テキストファイル

ドキュメントなど任意の内容を含むファイル:

- テキストファイル (*.txt)

各新規ファイルまたは開いたファイルは、個別のタブに表示されます。新しいプログラムファイルが作成されると、コメントヘッダが自動的に生成されます。VBSファイルの場合、主要な機能の本文が挿入されます。プログラムファイルを含むタブには追加のツールバーがあります。このツールバーからプログラムの起動、停止、アップロード、ダウンロードなどが実行できます。ツールバーの各アイコン、プログラミングおよびデバッグの手順は、各プログラミングマニュアルのプログラミングの章を参照してください。

プログラミングファイルのツールバーの[Extras]ボタン (8.1.2項、55ページを参照) を押すと、コードテンプレートや変数の確認/変更などの追加機能が表示されます。

コードテンプレートには、プログラムコードにドラッグしてカスタマイズできる一般的なプログラム構成要素が含まれています。コードの強調表示されたセクションをエディタウィンドウからコードテンプレートのツールバーにドラッグし、新しいテンプレートを作成することもできます。追加したテンプレートは、キーボードの「Del」キーを押すと簡単に削除できます。

ボタン

ユーザインターフェース	説明
	新しい文書を作成します
	保存された文書を開きます
	文書を保存します
	文書に名前を付けて保存します
	選択した内容をクリップボードにコピーします
	内容をクリップボードから挿入します

ユーザインターフェース	説明
	状況に応じたヘルプを呼び出します
	開いた文書を閉じます

5.7 Motion Managerの構成

メニュー項目[Extras] - [Options]を選択すると、Motion Managerの全般的な設定ができます。選択できるオプションは以下の通りです。

設定	説明
全般	Motion Managerソフトウェアのオンラインアップデートに関するオプション
データ交換	メッセージフィルター（CANのみ） Motion Managerには、使用されていないノードのCAN信号を除去するメッセージフィルターがあります。デフォルト設定では、ハートビート信号以外に、使用されていないノードの全メッセージがターミナルウィンドウの通信履歴に表示されます。「 Messages from inactive CAN nodes 」の下の[Hide all]チェックボックスを選択すると、デフォルトの設定を変更することができます。特定のCOB-ID付きCAN信号を除外しないように例外を追加できます

 アクティブなノードのハートビート信号は、NMT状態が変更された場合にのみ評価されます。それ以外の場合は、メッセージフィルターにより除去されます。

6 コントローラのコミッショニング

FAULHABER製コントローラの初期設定では、Motion Managerで利用できるさまざまなウィザードおよびダイアログが使用できます。設定の各手順で使用されるMotion Managerの機能は、各節で説明しています。

6.1 接続の確立

最初に、Motion ManagerがインストールされたPCでコントローラとの接続を確立する必要があります。ウィザードを使用して、対応インターフェースの1つを介して通信接続を設定できます。新しいプロジェクトを作成すると、ウィザードが自動的に表示されます。このウィザードは、クイックアクセスツールバーの[**Establish connection**]ボタン、または[**Terminal**]メニューからいつでも呼び出すことができます。

構成の設定に使用する拡張オプションは、[**Manage terminal connections**]メニューから呼び出すことができます。

コントローラへの接続の確立後、追加機能を使用して接続された駆動機器をカスタマイズできます。

6.2 モータの選択

接続されたモータを確実に操作できるように、コントローラの駆動パラメータおよびコントローラパラメータをカスタマイズして、モータおよび使用するセンサシステムに適合させる必要があります。追加のウィザードを使用できます。クイックアクセスツールバーの[**Select motor**]ボタン、または[**Configuration - Commissioning**]メニューで呼び出すことができます。

ウィザードの全ての手順が正常に終了すると、駆動パラメータおよびモータコントローラが、使用されるモータとセンサシステムの属性に適合されます。そのため、モータはアイドル状態（追加の負荷がない状態）で運転できます。

駆動パラメータの実際の構成は、コントローラごとに異なるため、ウィザードの内容も機器ファミリに応じて異なります。

モータの選択	MC V3.x	MC V2.x	SC
モータ選択用のウィザード	✓	✓	✓

6.3 コントローラの設定

モータの選択、電流コントローラ、スピードコントローラおよび位置コントローラの基本構成の終了後、コントローラを対象のシステム特性に適応させる必要があります。このためコントローラパラメータを最適化するためにシステムの情報が必要です。各種コントローラの特性に適合させるためにウィザードやダイアログが提供されています。

コントローラパラメータの構成は、駆動機能の構成ダイアログ、またはコントローラパラメータの動的な設定用のウィンドウも使用できますが、最初にウィザードを使用して、基本設定を実行することをお勧めします。

コントローラの設定	MC V3.x	MC V2.x	SC
コントローラ構成用のウィザード	✓	–	–
モータ選択ウィザードを使用したコントローラの設定	–	✓	✓
駆動機能の構成ダイアログ	✓	✓	✓
コントローラパラメータの動的な構成	✓	✓	–
コントローラのチューニング	✓	✓	–

6.4 モータの駆動

使用目的に応じて、駆動パラメータおよびコントローラパラメータを構成した後は、モータを運転できます。Motion Managerは、さまざまな目的に使用できる多くの機能があります。

単一の操作	MC V3.x	MC V2.x	SC
モータ運転ダイアログ	✓	–	–
モーションコックピット	✓	–	–
構成ダイアログによるモータ運転	–	–	✓
コントローラのチューニング	✓	✓	–

継続した操作	MC V3.x	MC V2.x	SC
マクロ	✓	✓	–
シーケンスプログラム	✓	(✓)	–
VB Script	✓	✓	–
各種コマンド	✓	✓	–

6.5 拡張機能

FAULHABER製モーションコントローラおよびスピードコントローラの機能全体の説明は、コントローラのマニュアルに記載されています（www.faulhaber.com/manuals）。ほとんどの機能は、各コントローラの駆動パラメータの構成ダイアログを使用して構成できます。追加のダイアログおよびツールが特殊機能、および駆動機器の解析に使用できます。

構成	MC V3.x	MC V2.x	SC
接続パラメータ	✓	✓	–
駆動機能	✓	✓	✓
CANopenの標準機能	(✓)	(✓)	–
ファームウェアのアップデート	✓	✓	✓

解析	MC V3.x	MC V2.x	SC
グラフィカルな解析	✓	✓	–
ステータス表示	✓	✓	–

7 コントローラ固有の機能

コントローラ固有の機能は、コントローラが接続されている場合にのみ使用できます。メインメニューの項目、およびクイックアクセス用の項目は、接続されたコントローラに応じて拡張されます。

7.1 MC V2.x / SCコントローラファミリ

この節で説明する機能は、MC V2.xファミリのモーションコントローラおよびSCシリーズのスピードコントローラで使用できます（1.1節、6ページ参照）。

7.1.1 モータ選択ウィザード

モータ選択ウィザードは、クイックアクセスツールバーの[*Select motor*]ボタンを使用して呼び出すことができます。

このウィザードにより、外部コントローラを接続されたモータに合わせてカスタマイズできます。このためには、接続済みのFAULHABER製モータをリストから選択し、

モータのデータ、電流限界値、および追加で算出されたコントローラパラメータを選択したモータ用に設定する必要があります。本機能は、コントローラを内蔵した機器でも使用できます。

モータ選択ウィザードは、5つのセクションに分かれ、選択した構成に柔軟に対応していますが、表示された全ての構成機能が対象のコントローラで使用できるわけではありません。

手順1: モータの選択

Node Explorerで選択されたコントローラに対応する、FAULHABER製モータが選択できます。

最上部のリストには、コントローラに対応するモータの種類が表示され、リストの下の2つのフィールドに、モータのラベルに記載された製品コードが表示されます。

リストに含まれていないモータを使用する場合は、[*Create*]ボタンを使用して、リストに追加できますが、モータのデータシートの値を手動で入力する必要があります。

また、モータに使用するセンサの種類（ホールセンサ、インクリメンタルエンコーダなど）の選択も必要になります。その他に、制御に適合したセンサの種類も表示されます。

 スピードコントローラの場合、接続されたコントローラで対応するモータおよびセンサの種類を確認することができないため、想定される構成が全て選択の対象として表示されます。

モータに印加される電圧（モータ電源用に電源供給ユニットで設定された電圧値）は、自動的にコントローラから読み込む（モーションコントローラ）、または手動で入力する必要があります（スピードコントローラ）。この値は、必ずしもモータの公称電圧と同じである必要はありません。

注意!

 高すぎる電圧値を印加した場合、または不正な電圧値を入力した場合は、機器が損傷する可能性があります。

- ▶ 各コントローラのデータシートを参照し、電圧の許容範囲を確認し、確実に順守してください。

手順2: 負荷伝達

このページでは、さまざまな種類の負荷伝達（ギアボックス、スピンドルなど）を選択できます。ここでの設定は、対象の機器内のコントローラ設計に影響があります。

手順3: 慣性係数

モータのコントローラパラメータの計算では、慣性係数 K_J が使用されます。この慣性係数は、次の公式により、モータの慣性 J_M の質量モーメントおよび結合負荷の慣性 J_L の質量モーメントから計算されます。

$$K_J = \frac{J_M + J_L}{J_M}$$

モータの質量慣性モーメントは、データシートでロータ慣性モーメントとして記載されています。負荷の慣性モーメントは、決定または推定する必要があります。

この値は、スライダーにより設定、または各入力フィールドに直接入力できます。リニアモータの場合は、質量の慣性モーメントではなく、質量を入力する必要があります。

コントローラパラメータの算出用の慣性係数の上限は30です。

手順4: コントローラの設定

コントローラパラメータ算出のための事前設定は、円滑な運転または高い動力にコントローラを最適化するためのデータです。。

状況に応じて、[Next]ボタンを押した後、必要な運転速度が要求されます。

手順5: 概要

ウィザードの最後のページには、選択されたモータ/センサの組み合わせ用に決定されたコントローラパラメータの要約が表示されます。

設定が間違っている場合は、[Back]ボタンを押して訂正できます。[Finish]ボタンを押すと、モータのデータおよび決定したコントローラパラメータがコントローラに転送されます。

スピードコントローラ:

スピードコントローラの場合、データを直接送信できません。データは、構成ダイアログで新しい設定をあらかじめ割り当てるために使用されます。構成ダイアログの[Send]ボタンを押すと、新しいデータがコントローラに転送されます。

モーションコントローラ:

モーションコントローラの場合は、データを各コマンドにより直接送信できます。コントローラの電源を再度投入した後もパラメータを保持する必要がある場合は、SAVEコマンドを実行する必要があります。この場合、確認を要求するメッセージが表示されます。

アナログ式ホールセンサを搭載した新しいブラシレスモータがモーションコントローラに接続されると、モータおよびコントローラパラメータを転送した後、ホールセンサの信号を最適化する必要があります。[Configuration drive functions]ボタンまたはメニュー項目をクリックし、構成ダイアログを開きます。

[Basic setting]タブで[Optimising to the connected motor]ボタンを使用して調整することができます（7.1.2項、31ページ参照）。

7.1.2 モーションコントローラの構成

7.1.2.1 駆動機能

変更した設定のコマンドがロードされます。新しい設定は直ちに有効になり、コントローラへの電源が切断されるまで有効な状態が維持されます。

新しい設定を恒常的に保存する場合は、[EPPSAV]または[SAVEAPP]ボタンをクリックする必要があります。これにより現在のパラメータが駆動機器の不揮発性メモリーにロードされます。

CANインターフェース付きモーションコントローラ、CFシリーズ

CANインターフェース付きモーションコントローラ、CFシリーズまたはそれ以前のモーションコントローラの場合、変更した設定のパラメータコマンドは、FAULHABERチャンネルを介してPDO2に送信されます。「運転」状態の場合のみ、構成ダイアログにアクセスできます。駆動機器は、[Command Network Management]メニュー (**Start Remote Node**) を使用して、事前に適切な設定をする必要があります。

ここでは、FAULHABERモード (OPMOD-1) が選択されている場合のみ、[Basic Settings]タブを使用できます。これは、構成にCANopen標準でサポートされないFAULHABER独自の構成が使用されるためです。

 [Basic settings]タブでは、BLおよびLMコントローラにも[Optimising to the connected motor]機能を使用できます。最適化を実行する前に、必ずFAULHABERモード ([Mode]タブのOPMOD-1) に切り替えてください (本章の「接続されたモータ (MCBL/MCLM) の最適化」を参照してください)。

CANインターフェース付きモーションコントローラ、COシリーズ

CANインターフェース付きモーションコントローラ、COシリーズの場合、パラメータ化は、CANopenオブジェクトディクショナリを使用して実行されます。変更されたパラメータはSDO通信で送信されます。

接続されたモータ (MCBL/MCLM) の最適化

コントローラに新しいBLモータまたはリニアモータが接続されていた場合は、新しいモータパラメータが設定された後にホールセンサ信号を最適化する必要があります。構成ダイアログの[Basic settings]タブにはホールセンサ信号の最適化を実行するためのボタンがあります。

ホールセンサ信号が最適化されていないと、電源投入後、数秒で動作が不安定になり、精度が低下する場合があります。

接続されたモータにMCBLコントローラをさらに適合させるために、正弦転流の位相角を最適化します。位相角が最適化されていない場合、多くの電力が消費され効率の低下の原因になります。

 ホールセンサ信号および位相角の最適化中に、最初の数秒間モータはフリーであり、無負荷で動作することを確認してください。

[*Optimising to the connected motor*]ボタンをクリックすると、ガイドに従ってホールセンサ信号および位相角を自動的に最適化できます。

最適化後、SAVEコマンドを使用して決定されたシステムパラメータをコントローラに恒常的に保存する必要があります。

例えば、接続されたモータがアイドル状態で、最高速度で運転できないギアヘッドにフランジが取り付けられているなどで場合、位相角の自動設定を使用できない可能性があります。このような場合は位相角を手動で修正できるオプションがあります。

- 各画面で使用する出力電圧を設定します。
設定は100%で最も正確に実行されます（15,000 min⁻¹または駆動機器の無負荷速度に相当）。特定の接続されたドライブユニットでは、例えばギアの最大入力回転数を超えないように、値を小さくすることをお勧めします。
- その後、位相角のスライダーをいずれかの方向に動かし、電流値を観察してください。最小の電流値が表示されると、位相角は最適な角度に設定されています。
- 手動での調整の完了後、[Next]をクリックしてページを終了します。ホールセンサ信号は再度最適化する必要があります。最後に、SAVEコマンドを実行し、決定されたシステムパラメータをコントローラに恒常的に保存します。

7.1.2.2 コントローラパラメータ

コントローラパラメータの動的設定

モータ選択ウィザードおよびコントローラチューニングウィザードの他に、コントロールパラメータを設定するためのダイアログがあります。コントローラパラメータをオンラインで変更するには、メニュー項目 [Configuration] - [controller parameters] を選択し、矢印を入力フィールドに移動させ、キーボードで数値を入力します。入力フィールドの数値は、変更されると直ちに駆動機器に自動的に送信されます。キーボードを使用した入力の場合、フィールドを終了する、または[Enter]ボタンを押すまで、各フィールドはグレーで表示されます。これで、全ての数値が確実に駆動機器に送信されます。

これは、ポテンションメータを使用するのと同様の技術によってパラメータの動的な調整が可能になります。

コントローラパラメータダイアログは、非モーダルダイアログで、他のウィンドウが開いている場合でも開くことができます。したがって、コントロールパラメータはグラフィック解析が表示されている間に変更でき、速度や位置の安定性への影響を監視できます。

 コントロールパラメータの最適化の後、「SAVE」または「EEPSAVE」を実行してください。これで、次回コントローラに電源を投入した際にもこれらのパラメータを使用することができます。通信マニュアルおよび駆動機能の「コントロールパラメータの設定」を参照してください。

7.1.2.3 接続パラメータ

Motion Managerの接続パラメータ構成ダイアログと同様に、コントローラの接続パラメータ用のダイアログも使用できます。[**Configuration**]メニューからアクセスできます。このダイアログにより、ノード番号、および必要に応じて転送速度を構成できます。変更が確認されると、Motion Managerの接続パラメータはアップデートされます。CANインターフェース付きコントローラの場合は、追加入力が必要です（7.3.4項、52ページ参照）。

i ネットワークで操作する場合は、ノードアドレスが2つ以上割り当てられていないこと、および全てのノードが同じ転送速度で動作していることを確認してください。

7.1.3 スピードコントローラの構成

スピードコントローラの場合、構成はファームウェアのダウンロードによって変更されます。

変更をすぐに確認するには、ダウンロード後に[**Run**]ボタンをクリックします。駆動機器は構成モードからオペレーティングモードに切り替わります。

オペレーティングモードでは構成を変更できません。変更を行う場合は、[**Stop**]ボタンをクリックし、駆動機器を構成モードに切り替えます。

コントローラの場合は、電源の投入後のみ構成モードに切り替えることができます。構成中に接続が切断された場合は、[**Run**]/[**Stop**]ボタンをクリックして構成モードに切り替えます。

構成ダイアログが終了すると、駆動機器はオペレーティングモードに切り替わります。駆動機器は、設定された構成で直ちに作動します。駆動機器を直ちに作動させない場合は、構成の完了後電源を切断する必要があります。

7.1.4 コントローラのチューニング

コントロールチューニングウィザードは、クイックアクセスツールバーの、[**Controller tuning**]ボタンで呼び出すことができます。

このウィザードは、グラフィカルな解析を使用したステップ応答の受け入れや評価機能を提供します。これにより手動によるコントローラパラメータの最適化を可能にします。

i CANインターフェース付きモーションコントローラの場合、NMTが「**運転**」状態である場合のみグラフィカルな解析が使用できます。

ツールバーのボタンは、駆動機器を有効または無効にすることができ（出力ステージの有効化/無効化）、ステップシーケンスの間隔および目標回転域のサイズを設定することができます。

注意!

! 衝突による機器の損傷

ステップシーケンスを実行する時、駆動機器は入力された数値に従って動作します。可動範囲内に障害物がある場合は、衝突する可能性があります。

- ▶ ステップシーケンスを実行する時、駆動機器が入力された数値の範囲内で自由に動作できることを確認してください。

コントロールパラメータを最適化するために、次の手順を実行する必要があります。位置決めタスクでは、最初にスピードコントローラを最適化し、次に位置コントローラを最適化することが推奨されています。

i COシリーズのモーションコントローラでは、解析データの記録にオブジェクトTxPDO4が必要です。このオブジェクトは、記録の目的で一時的に再構成されます。ウィザードを閉じると、オブジェクト TxPDO4の元のマッピングが復元されます。

手順1: ステップ応答の記録

最初のステップで、必要な速度および位置の設定値を設定する必要があります。[Start]ボタンをクリックすると、設定値は設定された間隔でコントローラに交互に送信されます。

設定によって、選択されたサイズに対して1つまたはそれ以上のステップ応答が実行・表示されます。

連続シーケンスの場合、コントローラのゲイン用のスライダーが変更され、速度信号および位置信号に対する影響を監視できます。

手順2: ステップ応答の評価

1回以上の完全なステップ応答の記録がある場合は、[Analysis]タブを使用して、ステップ応答を評価できます。シーケンス内で複数のステップが実行された場合は、最後のステップ応答が表示されます。

「Scrolling arrow」キーを使用して、解析されたステップ応答を交互に切り替えて結果を比較することができます。[Recycle bin]ボタンを使用して、表示された解析結果をステップ応答のリストから削除できます。

表1: 解析の値

値	値
Control rise time	開始回転域から最初の目標回転域に到達するまでの時間
Control settling time	開始回転域から最後の目標回転域に到達するまでの時間
Overshoot	実際の設定値からの最大偏差

手順3: コントローラパラメータの最適化

駆動機器の動作をさらに最適化する必要がある場合は、[Speed controller]タブまたは[Position controller]タブのスライダーを使用して、各コントローラのゲインを変更し別のシーケンスを開始することができます。

新しく記録されたシーケンスの最後のステップ応答は、[Analysis]タブを使用して再度評価でき、以前に記録されたステップと比較できます。

[Apply]ボタンを使用すると、この設定を駆動機器に再度ロードできます。セキュリティクエリを確認した後、設定を恒常的に保存します。

-  コントローラの基本設定には、モータ選択ウィザードを使用してください。
位置決めタスクでは、最初にスピードコントローラを可能な範囲で最も動的に設定します。つまりオーバーシュート、立ち上がり時間および遷移時間を最も低く設定し、次に位置コントローラを最適化します。
全てのコントローラパラメータが、コントローラチューニングウィザードを使用して設定できるわけではありません。詳細な設定が必要な場合は、メニュー項目[Configuration] - [controller parameters q]を選択し、[Controller parameters]のフォームを使用してください。このフォームは、変更がコントローラの動作に与える影響を監視する目的で、グラフィカルな解析の追跡ウィンドウとともに表示できます。

7.1.5 追跡機能

追跡機能を使用すると、ロギングモードでコントローラのパラメータを2個まで記録できます。

パラメータ値は、連続的に要求され読み込まれます。CANインターフェース付きの駆動機器の場合は、PDO通信サービスが使用されます。

RSまたはCFインターフェース付きモーションコントローラ:

各曲線に対して選択リストを使用して、ログに記録するデータソースを選択できます。リストには、各コントローラが対応する指定されたデータソースが含まれています。また、コントローラが対応するパラメータ番号を追加するオプションもあります。データが選択されると、スケーリングを設定する追加オプションが使用できます。

COインターフェース付きモーションコントローラ:

ソースの選択により、コントローラの使用可能な送信PDOの1つを選択できます。PDOマッピングによってそれぞれのPDOに割り当てられたパラメータはデータソースとして使用できます。PDOマッピングダイアログを使用して、PDO（例: TxPDO4）の追跡にオブジェクトディクショナリからのマッピング可能なパラメータを割り当てることができます。その後、PDOはデータソースとして使用できます（7.3.3.2項、51ページ参照）。使用可能な全てのデータソースには独自のエリアが含まれ、スケーリングの無効化または有効化、またはスケーリングの設定ができます。

 CANインターフェース付きモーションコントローラの場合、NMTが「運転」状態である場合のみ追跡機能の実行が可能です。

表示とスケーリング:



■ スケーリングON/OFFの自動切り替え手動スケーリングの場合、Y軸の最小値および最大値の下に数値が表示されます。



■ 他のY軸との同期を有効または無効にします。軸の設定に加えた変更は、他の軸にも適用されません。



■ 中央曲線自動スケーリングが有効な場合、現在の軸の設定は入力フィールドに入力されます。それ以外の場合は、軸の最小値および軸の最大値は変化し、距離は変わらないままです。

7.1.5.1 追跡設定

[Trigger]タブ

次の設定は、[Signal recording]エリアで指定できます。

Continuous: (継続)	記録は継続して実行されます。
Single shot: (シングルショット:)	選択されたトリガソースがトリガしきい値の限度を超えると、記録は停止されません。
Types of trigger: (トリガの種類:)	トリガの指定された種類、およびオプションのトリガ遅延時間から選択できます (COシリーズのコントローラを除く)。

[Buffer]タブ

次の設定は、データ取得エリアで指定できます。

Number of data packets per request: (リクエストごとの所定の数のデータパケット:)	要求ごとに指定された数のデータパケットがコントローラによってPCに送信されます。
--	--

Time resolution: (時間分解能:)	時間分解能は、コントローラまたはPCからオプションとして指定できます。PCから指定した場合は、希望のサンプリング間隔を設定できます。それ以外の場合、使用可能な通信方法で許す限りの速度でデータ交換が行われます。
Total buffer size: (バッファ全体のサイズ:)	ここで指定される数値は、記録中に一時的に保存されます。長時間にわたる記録の場合、古い値は上書きされます (リングバッファ)。
Fixed X axis: (固定されたX軸:)	「fixed X-axis」設定が有効な場合は、一定の時間がディスプレイウィンドウに表示されます。

[Curb]タブ

次の設定は表示エリアで実行できます。

Curve: (曲線:) 曲線表示のパラメータを設定できます。

Physical unit/conversion factor: (物理単位/変換係数:) コントローラがサポートする生データを限定的な単位に変換できます。



コントローラは常に生データをサポートします。変換はPCで実行されます。

7.1.6 状況表示

状況表示は、選択されたコントローラがこの機能に対応している場合に、メニュー項目[Tools] - [Status display]を使用して開くことができます。

値が変更されると、値の前にチェックマークが付きます。変更されない値にはチェックマークが付きません。500msごとの位置で表示が更新されます。

7.1.7 シーケンスプログラム

シーケンスプログラムの保存と実行に対応するモーションコントローラは、シーケンスプログラムの編集、ロード、デバッグおよび管理する機能を提供します。

既存のプログラムシーケンスをロード:

既存のプログラムシーケンスは、[File] - [open]コマンドを使用して、File editorウィンドウにロードできます。

ファイル形式:

モーションコントロールのファイルMC V2.xには、デフォルトで拡張子「*.mcl」が付きます。mclファイルはASCII形式で保存されるため、任意のテキストエディタで読み込むことができます。

シーケンスプログラムの作成:

コンテキストメニューから新しいシーケンスプログラムを作成するには、[File - New]メニューから[Motion Control file MC V2.x]を選択します。

ここではコードを入力できます。プログラム編集モード ([Edit] - [programme file]メニュー) が有効な場合は、コマンドメニューのコマンドをプログラムコードに直接ロードできます。

[File]タブの下にあるツールバーを使用すると、シーケンスプログラムのロード、実行およびデバッグができます。

構文説明

- 各行にはコマンドが含まれています。コマンドは引数の後に数字が続きますの後に数字の形式で引数が続く場合があります（例: LA1000）。
- 行頭のスペース、およびコマンドと引数との間のスペースは無視され、英数字のみが送信されます。
- コマンド以外にコメントも入力できます。コメントは、セミコロン (;) に続けて入力され、コマンド行の最後または別の行に書き込むことができます（例: HO ;Define home position）。
- コメントは駆動機器に送信されません。コメントはPCに保存されたプログラムの単なる覚書です。
- 原則としてプログラム行は、セミコロンの記載直前の文字または数字までがモーションコントロールに送信されます。モーションコントローラは、有効なコマンドを含む各プログラム行を保存します。
- コマンド[PROGSEQ]および[END]は、[Load programme file]機能によって自動的に送信されるため、入力する必要はありません。

プログラム編集モードを終了するには、メニュー項目[Edit] - [programme file]を再度選択し、無効にします。これで、コマンドをコマンドメニューから駆動機器に直接送信できます。

シーケンスプログラムのコントローラへのロード:

[Sequence programme]の選択後、入力またはロードされたプログラムは、メニュー項目[Commands] - [Load file]を使用してコントローラに送信できます。

また、[File]タブの[Start]ボタンをクリックして、シーケンスプログラムを転送することもできます。プログラムは直ちに開始します。

シーケンスプログラムの比較:

Edit fileウィンドウのコマンドコードとコントローラに保存されたプログラムコードを比較し、一致していることを確認できます。これを実行するには、メニュー項目[Commands] - [Compare files]を使用し、次に[Sequence programme]を選択します。

デバッグツールバーを使用してロードされたシーケンスプログラムは、プログラムの開始前に関連する構文エラーの表示と照合されます。

コントローラからMotion Managerへのシーケンスプログラムのロード:

コントローラに保存されたシーケンスプログラムは、メニュー項目[Commands] - [Receive file]を使用し、さらに[Sequence programme]を選択することでMotion Managerにロードできます。プログラムコードは新しい[File]タブに表示され、編集、保存、および印刷することができます。また、再度ロードすることも可能です。

シーケンスプログラムの開始:

[ENPROG]コマンドを使用する、または[Execute sequence programme]ボタンをクリックすると、コントローラに送信されたプログラムを開始できます。

デバッグツールバーを使用してロードしたシーケンスプログラムは、自動的に開始されます。

シーケンスプログラムのデバッグ:

Motion Controlファイル形式V2.xのFile Editorウィンドウでは、追加のツールバーが使用できます。

アイコン	機能
	シーケンスプログラムのロードと実行: シーケンスプログラムをロードすると、再度読み込まれ、構文エラーの有無がチェックされます。構文エラーがある場合は、対象のコマンド行はモーションコントローラで解釈されず、エラーのプログラム行がEditorウィンドウに赤色で表示されます。ロードの後のプログラムにエラーがない場合は、File Editorウィンドウ全体がグレーで表示されます。プログラムはコントローラでこの状態のまま実行されます。デバッグモードが有効になります。
	シングルステップ: 表示されたシーケンスプログラムが、モーションコントローラにロードされ比較されます。ロードの後のプログラムにエラーがない場合は、File Editorウィンドウ全体がグレーで表示され、最初のプログラム行が緑色で表示されます。現在表示されているシーケンスプログラム行は、ボタンをクリックすると次の1行に進みます。
	シーケンスプログラムの中断: プログラムがデバッグモードで実行されている場合に、このボタンをクリックすると実行中のプログラムを中断できます。現在のプログラム行がEditorウィンドウに緑色で表示されます。
	シーケンスプログラムの停止: プログラムがデバッグモードで実行されている場合に、このボタンをクリックするとプログラムが停止し、デバッグモードが終了します。File Editorウィンドウは編集モードに戻り、プログラムコードを再度変更することができます。

 Motion Manager 6\Examplesの下のインストールフォルダには、シーケンスプログラムの例が格納されています。

7.1.8 パラメータファイル

シーケンスプログラム用のファイル機能に加え、パラメータファイルをロード、受信および比較する機能、パラメータファイルとシーケンスプログラムを完全な構成の機能があります。

既存のパラメータファイルのロード:

既存のパラメータファイル（シーケンスプログラムの有無）は、[File] - [open]コマンドを使用してFile Editorウィンドウにロードできます。

 モーションコントローラパラメータファイルはテキストファイルで、デフォルトの拡張子は「*.mcp」です。

パラメータファイルの転送:

ロードされたパラメータセットは、[Commands] - [Load parameter file]または[Commands] - [Load file]項目を使用し、さらに[Parameter file]を選択することでコントローラに送信できます。

プログラムシーケンスおよびパラメータのリストを含む構成ファイルがロードされている場合、このファイルは[Parameter file with sequence programme]を使用してロードできます。

パラメータファイルの受信:

メニュー項目[Commands] - [Receive parameter file]または[Commands] [Receive file]を使用し、さらに[Parameter file]または[Parameter file with sequence programme]を選択することで、コントローラに保存された駆動構成のイメージを作成できます。

読み込まれたパラメータ構成（シーケンスプログラムありまたはなし）が新しい[File]タブに表示され、編集、保存、および印刷することができます。また、再度ロードすることも可能です。

パラメータファイルの比較:

メニュー項目[Commands] - [Compare parameter file]または[Commands] - [Compare file]を使用し、さらに[Parameter file]または[Parameter file with sequence programme]を選択すると、ロードされた構成ファイルがコントローラの現在の構成と一致しているか確認できます。

パラメータファイルの編集:

文書化を目的として、パラメータファイルを閉じる前に、ファイルにコメントを記入することができます。

7.2 MC V3.x コントローラファミリ

この節で説明する機能は、MC V3xファミリのモーションコントローラで使用できます（1.1節、6ページ参照）。

7.2.1 モータ選択ウィザード

このウィザードでは、クイックアクセスツールバーの[**Select motor**]ボタン、または[**Configuration**] - [**Commissioning**]メニューを使用して、設定を呼び出すことができます。

ウィザードを使用すると、接続されたモータおよび使用されるセンサシステムに応じて、FAULHABER製コントローラをカスタマイズできます。構成は複数のステップに分かれ、選択した構成に柔軟に対応していますが、表示された全ての構成機能が対象のコントローラで使用できるわけではありません。

注意!



不適切なモータの選択や不正確なデータシートの値の入力は、モータが損傷する可能性があります。

- ▶ 正しいモータが選択され、正確なモータのデータが入力されていることを確認してください。

1. FAULHABER製モータのリストから使用するモータを選択します。使用するモータがリストにない場合:
 - モータのデータシートの数値を入力し、これらの数値に基づき新しいモータを作成します。
 - 新しく作成されたモータは恒常的に使用可能であり、必要に応じて編集または削除できます。
2. 使用するセンサシステムを入力します。
 - 多くの場合、モータは組込型または外付け型のセンサシステムを装備しています（モータの製品説明を参照してください）。
 - 追加のセンサシステムを使用する場合は、選択リストを使用してセンサシステムとモーションコントローラの接続関係を指定する必要があります。
3. 指定されたセンサシステムに機能を割り当てます。
4. 指定した構成を概要で確認し、コントローラにロードします。

i モータのパラメータおよびセンサシステムの構成の実行の他に、ウィザードではバックグラウンドで他の機能も構成します。

- コントローラおよびプロファイルパラメータは、駆動機器をアイドル状態で運転できるように構成されます。
- 因子グループ（速度および位置のためのユーザユニットの構成）はリセットされます。
- 速度ウィンドウと位置ウィンドウはリセットされます。
- 電流および速度の事前制御が無効になります。

5. アナログ式ホールセンサを使用している場合:

- モータが無負荷で自由に動作できるようにしてください。
- ホールセンサ信号を調整してください。

🔄 ホールセンサ信号が最適化されていないと、動作が不安定になり、精度が低下する場合があります。

7.2.2 コントローラ構成ウィザード

安定して制御するには、コントローラを使用目的に応じた構成にする必要があります。このウィザードは、クイックアクセスツールバーの[**Configure controller**]ボタン、または[**Configuration**] - [**Commissioning**]メニューを使用して、呼び出すことができます。

1. システムパラメータを決定します。

コントローラパラメータを制御するシステムに一致させるには、システムのパラメータを次の方法で決定する必要があります。

手動入力

モータのコントローラパラメータの計算では、慣性係数 K_J が使用されます。

この慣性係数は、次の公式により、モータの質量慣性モーメント J_M のおよび取り付けられた負荷の質量慣性モーメント J_L のから計算されます。

$$K_J = \frac{J_M + J_L}{J_M}$$

モータの質量慣性モーメントは、データシートにロータ慣性として記載されています。負荷の慣性モーメントは、決定または推定する必要があります。この値は、スライダーによる設定、または各入フィールドに直接入力できます。リニアモータの場合は、質量慣性モーメントではなく、質量を入力する必要があります。

自動識別

i 自動識別は、モータに速度センサが取り付けられている場合にのみ使用できます。

注意

! 自動識別が実行されると、駆動機器は動作します。可動範囲内に障害物がある場合は、衝突および損傷する可能性があります。

- ▶ 自動認識を実行する場合は、入力値の範囲内で駆動機器が自由に動作することを確認してください。

自動識別の場合、システムパラメータは特定の測定プロセスで決定されます。慣性の質量を決定する場合、駆動機器は、一定の速度まで等加速度を実現するために、電流制御モードで、モータの定格トルクで短時間動作します。最大許容速度および指定した駆動範囲は、この操作中にコンパイルされます。測定はいつでもキャンセルできます。

識別プロセス中にモータが機械的制限により拘束された場合、識別は無効になります。この場合、駆動機器を開始位置に戻し、自動識別を繰り返しおこなってください。

識別後、必要に応じて識別されたパラメータを表示および修正してください。

👉 必要なパラメータが渡された場合は、決定されたコントローラパラメータがウィザードの最後のページに表示されます。

2. コントローラパラメータをチェックし、コントローラにロードします。

7.2.3 モータの設定

接続されたコントローラに対し、包括的でグラフィカルな構成ダイアログが使用できます。

このダイアログを使用すると、簡単に駆動機器の構成およびパラメータ化ができます（クイックアクセスバーの[*Drive functions*]ボタンまたは[*Configuration*]メニュー）。

ダイアログはコントローラの機能に合わせて調整され、柔軟な構成機能を提供します。ウィンドウ左側のツリー構造を使用して、各入力ページに切り替えることができます。入力ページのヘッダにある[*Back*]ボタンで、前のページに切り替えることができます。

表示されたパラメータは、コントローラの構成可能なオブジェクトに対応します。入力項目の上にマウスポインタを置くと、オブジェクトに関する情報とともにツールチップが表示されます。

変更は、グラフィカルユーザインターフェースで直接実行し、[*Send*]ボタンをクリックして駆動機器にロードできます。新しい設定は直ちに有効になり、コントローラへの電源が切断されるまで有効な状態が維持されます。

新しい設定を恒常的に保存する場合は、[*Save*]ボタンをクリックする必要があります。これによって現在のパラメータが駆動機器の不揮発性メモリーにロードされます。

 各オブジェクトの機能については、「駆動機能」を参照してください。

7.2.4 コントローラパラメータの変更

Controller parametersウィンドウを使用して、駆動機器の運転中にコントロールパラメータ、フィルタ設定およびプロファイルパラメータを変更できます。終了するまでフォアグラウンドに残る他の構成ダイアログとは対照的に、コントロールパラメータダイアログは他のウィンドウと並行して使用できます。

1. クイックアクセスツールバーの[**Controller parameters**]ボタン、または[**Configuration**]メニューを使用して、**Controller parameters**ウィンドウを開きます。
2. 入力フィールドの矢印を使用、またはキーボードで値を入力して、必要なパラメータの設定を変更します。
 - 👉 変更された値は自動的に駆動機器に送信されます。キーボードを使用して入力する場合、入力フィールドを終了する、または[**Enter**]キーを押すまで、各入力フィールドはグレーで表示されます。

ツールバーのボタン

ボタン	機能
	現在の設定をコントローラに恒常的に保存します
	駆動機能の構成ダイアログを表示し、追加のパラメータを設定できるようにします
	現在のコントローラの構成の概要を表示します。

7.2.5 接続パラメータ

Motion Managerの接続パラメータ構成ダイアログ以外に、コントローラの接続パラメータ用のダイアログも使用できます。[**Configuration**]メニューからアクセスします。

このダイアログにより、ノード番号、および必要に応じて転送速度を構成できます。変更が確認されると、Motion Managerの接続パラメータはアップデートされます。

CANインターフェース付きコントローラの場合は、追加入力が必要です（7.3.4項、52ページ参照）。

 ネットワークで操作する場合は、ノードアドレスが2つ以上割り当てられていないこと、全てのノードが同じ転送速度で動作していることを確認してください。

7.2.6 モーションコックピット

モーションコックピットは、入力項目および表示項目が含まれ、動作の制御を容易にします。これは、クイックアクセスツールバーの[**Motion Cockpit**]ボタンまたは[**Tools**]メニューを使用して起動します。

注意

 動作コマンドが実行されると、モータは入力された値に従って動作します。可動範囲内に障害物がある場合は、衝突および損傷する可能性があります。

- ▶ 動作コマンドの実行中は、モータが入力された数値の範囲内で自由に動作できることを確認してください。

駆動機器の準備:

1. 選択リストから、必要なオペレーティングモードを選択します。
2. [**Activate**]ボタンをクリックして、オペレーティングモードを有効にします。
3. 出力ステージがオンであることを確認し、必要に応じて表示されるボタンを使用してオンにしてください。
 モータが入力された動作コマンドを実行できるように、出力ステージをオンにする必要があります。

設定値の入力:

コントローラが選択されたオペレーティングモードになると、入力エリアに設定値を入力できます。

設定値の交互の送信:

モーションコックピットのこのエリアでは、以前に入力された設定値を交互に送信できます。動作コマンド間の待ち時間は、各選択リストで設定することができます。

ツールバーのボタン

ボタン	機能
	出力ステージをオフに切り替えます（緊急停止）。
	機器パラメータを記録する Graphical analysis ウィンドウを開きます。
	機器の状態を監視する Status display ウィンドウを開きます。
	コントローラパラメータおよびプロファイルパラメータを設定する Controller parameters ウィンドウを開きます。

7.2.7 コントローラのチューニング

クイックアクセスツールバーの[**Controller tuning**]ボタン、または[**Tools**]メニューを使用すると、**Controller tuning**ツールを開くことができます。

このツールは、グラフィカルな解析を使用したステップ応答の受け入れや評価する機能を提供します。これにより手動によるコントローラパラメータの最適化を可能にします。

ツールバーのボタンを使用すると、モータを有効および無効にできます（出力ステージを有効/無効にします）。

注意!

 **ステップシーケンスの実行中、モータは入力される数値に従って動作します。可動範囲内に障害物がある場合は、衝突および損傷する可能性があります。**

- ▶ ステップシーケンスの実行中は、モータが入力された数値の範囲内で自由に動作できることを確認してください。

コントロールパラメータを最適化するために、次の手順を実行する必要があります。位置決めタスクでは、最初にスピードコントローラを最適化し、次に位置コントローラを最適化することが推奨されています。

1. ステップ応答の記録

- 速度および位置にの値を設定します。
- 適切なボタンをクリックして、ステップ応答または連続シーケンスを選択します。

ボタン	機能
One step response (ステップ応答)	各ステップ応答を記録します。
Continuous (継続)	ステップの連続的な繰り返しによるアクティブな運転で、連続シーケンスを実行してコントローラパラメータの調整を有効にします。

2. ステップ応答の評価

少なくとも1つ以上の完全なステップ応答の記録がある場合は、[**Analysis**]ページでステップ応答を評価できます。シーケンス内で複数のステップが実行された場合は、最後の完全なステップ応答のみが表示されません。

- **ステップ応答:**

記録に関する主要なデータの概要がこのエリアに表示されます。表示される目標回転域は、コントローラで設定された回転域に対応します。これはステップ応答の評価に使用されます。次の変数が評価エリアで解析されます。

表2: 解析の値

値	意味
制御の立ち上がり時間	設定値の変更に対し最初に到達するまでの時間
制御の設定時間	設定値の変更に対し到達して安定するまでの時間
オーバーシュート	実際の設定値からの最大偏差

- **比較:**

このエリアでは、結果を比較するために記録されたステップ応答を交互に切り替えることができます。[**Recycle bin**]ボタンは、表示された記録をステップ応答のリストから削除できます。[**Apply parameters**]ボタンは、下のエリアに表示されたコントローラパラメータをコントローラにロードできます。パラメータの表示は、表示されたステップ応答の記録時に有効であった構成に対応します。

i コントローラの基本設定には、コントローラ構成ウィザードを使用してください。

位置決めタスクでは、スピードコントローラを可能な範囲で最も動的に設定します。つまりオーバーシュート、立ち上がり時間および遷移時間を最も低く設定し、次に位置コントローラを最適化します。

全てのコントローラパラメータは、コントローラチューニングツールを使用して設定できるわけではありません。詳細な設定は、クイックアクセスツールバーまたは[**Configuration**]メニューの[**Controller parameters**]フォームを使用してください。このフォームは、変更がコントローラの動作に与える影響を監視する目的で、グラフィカルな解析の追跡ウィンドウとともに表示できます。(7.4章、53ページ参照)。

7.2.8 追跡機能

追跡機能を使用すると、コントローラのパラメータを4個まで記録できます。

記録の種類

2種類の記録方法が利用できます。

記録の種類	説明
ロガー	パラメータ値は、連続して要求され読み込まれます。したがって、長時間にわたる連続的なデータの記録が可能です。CANインターフェース付きの駆動機器の場合は、PDO通信サービスが使用されます。各データ要求が使用可能な通信路を通過する必要があるため、分解能は制限された状態です。
レコーダ	パラメータ値は内部装置のバッファに書き込まれ、次に読み出されます。追跡バッファは、可能な限り高い解像度でデータを保存できますが、メモリのサイズによって制限されるため、連続的な記録の実行はできません。レコーダにはトリガ機能があります。

オプション:

■ ソース:

使用可能な構成の1つをソースとして使用できます。オプションにはソース、およびロガーとレコーダの適切な構成が含まれています。[Edit settings] リンクから構成の表示および変更ができます。

■ モード:

記録の種類を選択: ロガーまたはレコーダ。レコーダに対してトリガソースが定義されている場合は、運転中にトリガモードを選択できます。

 CANインターフェース付きモーションコントローラの場合は、NMTが「運転」状態である場合のみロガーモードの実行が可能です。

■ データソース:

現在のソース選択リストの全てが割り当てられたデータソースには独自のエリアが含まれ、スケーリングを無効または有効にしてスケーリングの設定ができます。



- スケーリングON/OFFの自動切り替え。手動スケーリングの場合、Y軸の最小値および最大値の下に数値が表示されます。



- 他のY軸との同期を有効または無効にします。軸の設定を変更すると、他の軸にも適用されません。



- 中央曲線。自動スケーリングが有効な場合、現在の軸の設定は入力フィールドに入力されます。それ以外の場合は、軸の最小値および軸の最大値は変化し、距離は変わらないままです。

記録

記録が開始されると、アクティブ化されたデータソースは、さまざまな色の曲線で[Graphical analysis]ウィンドウに表示されます (7.4節、53ページ参照)。

以下の場合、記録が停止します。

- ボタンを押して記録を停止した場合。
- コントローラが応答しない場合。
- PCが素早くデータを表示できない場合。
- 記録が、記録の種類「Recorder」で終わる場合 (Single shot)
- 線形バッファが使用され、満杯な状態の場合 (7.2.8.1項、46ページ参照)。

7.2.8.1 追跡設定

記録パラメータおよびデータソースは、**Settings**ウィンドウで構成されます。複数の構成が使用可能で、自由にカスタマイズできます。設定はプロジェクトファイルに保存されます。

全般

- バッファ合計: 作業メモリーが必要以上に多くの追跡データでいっぱいにならないように、記録されるパラメータ値の最大数を制限できます。デフォルトで最大60,000のパラメータ値が設定されていますが、1msの分解能では、60秒の記録時間に相当します。
- ロガーのためにリリースされるTxPDO: ロガーモードのためにCANインターフェースを使用して、Motion Managerが使用できるTxPDOの決定に使用されます。

データソース

コントローラのオブジェクトディクショナリからの全てのマッピング可能なオブジェクトは、データソースとして使用できます。

最も重要なパラメータは[Proposed sources]タブの下に表示されます。

[Arrow]ボタンを使用すると、右側に表示されているリストから強調表示されたパラメータをソース選択またはトリガソースに割り当てることができます。パラメータは、ドラッグ&ドロップでも割り当てることができます。

[Recycle bin]ボタンまたは[Delete]キーを使用すると、強調表示されたデータソースを削除できます。

ロガー

連続的なデータの記録の場合には、サンプリング間隔を指定できます。

- 最速のサンプリング: 前のクエリが完了すると、直ちに新しいデータが要求されます。
- 固定サンプリング間隔: データは、特定の間隔でサンプリングされます。

 使用する通信インターフェースによっては、指定された間隔が適用できない場合があります。この場合は、最速のサンプリングが使用されます。

レコーダ

記録の種類が「レコーダ」である場合は、一定のサンプリング間隔で一定量のデータが記録されます。構成は、コントローラに定義された追跡オブジェクトを適合させるために実行されます（「通信マニュアル」を参照してください）。

 追跡機能に関する詳細は、コントローラの通信マニュアルを参照してください。

7.2.9 状況表示

さまざまな制御パラメータの状況を表示する状況表示は、周期的に問い合わせを受け取ります。状況表示にはコントローラのエラー履歴も表示されます。エラー履歴は、必要に応じてリセットできます。

クイックアクセスツールバーの[Status display]ボタンまたは[Tools]メニューを使用して、状況表示を開くことができます。

7.2.10 シーケンスプログラム

シーケンスプログラムの開始と停止

コントローラに保存されたシーケンスプログラムは、以下の方法で開始できます。

機器に保存されたシーケンスプログラムとは対照的に、PCに保存されたファイルには追加のコメントが含まれフォーマットが異なります。



ツールバーの[*Load sequence programme from the EEPROM and execute it*]ボタンをクリックします。
[Commands] - [sequence programme]の下の各メニュー項目を使用します。



エディタの[Start]ボタンを使用してシーケンスプログラムを実行すると、シーケンスプログラムは最初に制御RAMにロードされ、次に実行されます。このプログラムは、[Save to EEPROM]ボタンがクリックされるまで、Motion ControllerのEEPROMに恒久的に保存されません。メインメニューまたはメインツールバーからシーケンスプログラムを実行する場合は、最初に選択したプログラムを制御EEPROMからRAMにロードし、次に実行します。

シーケンスプログラムの停止

実行中のシーケンスプログラムは、以下の方法で停止できます。



ツールバーの[*Stop running sequence programme*]ボタンを使用します。
[Commands] - [sequence programme]の下の各メニュー項目を使用します。

シーケンスプログラムの編集

統合開発環境を使用して、シーケンスプログラムをエディタエリアで編集できます。シーケンスプログラムの作成、編集のみでなく、ダウンロード、実行、デバッグも実行できます。

Motion Managerを使用したシーケンスプログラムの開発、およびプログラミング言語に関する詳細情報は、プログラミングマニュアルを参照してください。

シーケンスプログラムの読み込み

コントローラに保存されたシーケンスプログラムは、以下の方法で読み込みエディタエリアに表示できます。

- [Commands] - [Sequence programme] - [Read out and show all]メニューを使用します。
- アクティブなノードの[Sequence programmes]の下のNode Explorerで、項目[Upload]をダブルクリックします。

[Sequence programmes]の下のNode Explorerは、プログラムファイルへのリンクを管理します。これらのファイルはロードされたプロジェクト内で編集およびダウンロードされます。ファイルをダブルクリックすると、エディタで直接開くことができます。

シーケンスプログラムの保存

[Save to EEPROM]ボタンをクリックすると、シーケンスプログラムがMotion ControllerのEEPROMに恒久的に保存されます。

シーケンスプログラムのMotion ControllerのEEPROMへの保存では、ファイル名およびダウンロード時間も保存されます。この情報はプロジェクトファイルにも挿入されます。ファイル名およびダウンロード時間の情報は、アップロードの際に正確性をチェックに使用されます。チェックで問題がない場合は、機器のプログラムに適したファイルがロードされ表示されます。



Motion Manager 6\Examplesの下のインストールフォルダには、シーケンスプログラムの例が格納されています。

7.2.11 周波数応答の測定

モータの周波数応答は、次のように測定してボード線図に表示できます。

- ✓ モータの選択が完了していること（7.2.1項、39ページ参照）。
- ▶ **[Tools] - [Frequency response measurement]**メニューを使用して、周波数応答を測定します。
 - ☞ モータは励磁し、ノイズ信号が発生します。これにより、オープン制御ループでコントローラの周波数応答が測定されます。信号のレベルはモータの特性に適合されます。これによりモータが損傷することはありませんが、モータの励磁がノイズ発生を増加させ、駆動軸の強い振動を発生させる可能性があります。

励磁プロセス中にモータが機械的制限により抑制された場合、測定は無効になります。この場合、提供された機能および反復されるプロセスを使用して、駆動機器の開始位置を修正する必要があります。

7.3 CANopenの標準機能

Motion Managerには、CiA 301およびCiA 402に準拠したCANopen acc.の標準機能が含まれています。CANopen対応の機器は、Node Explorerに表示され、原則として基本機能で操作することができます。

7.3.1 NMT

CANopenネットワーク管理機能は、[Network management]の下の[Commands]メニューで使用できます。NMTステートマシンの遷移に従って、CANopenノードの起動、停止、*運転前*NMT状態への切り替え、またはリセットが可能です。

メニュー項目以外に、10.1節、66ページのコマンドも使用できます。

7.3.2 機器制御

駆動コントローラ用のCANopenの機器制御機能は、[Commands]メニューの[Device Control]で使用できます。CiA 402に準拠した機器制御ステートマシンの遷移に従って、制御ワードを使用してCANopenでモータを運転できます。

メニュー項目以外に、10.1節、66ページのコマンドも使用できます。

7.3.3 SDO/PDO

[Commands] - [Object dictionary]メニューのコマンドSOBJおよびGOBJは、SDO（サービスデータオブジェクト）を使用したデータ交換に使用できます。これらはオブジェクトディクショナリ内のオブジェクトの記述や読み込みで使用できます。

このコマンドの使用方法は、10.1.4項、70ページおよび10.1.5項、70ページを参照してください。オブジェクトディクショナリの特定のオブジェクトに容易にアクセスする追加的なMotion Managerのコマンドについては、10.1.2項、68ページおよび10.1.3項、69ページを参照してください。7.3.3.1項、50ページで説明するオブジェクトブラウザは、オブジェクトディクショナリの全てのオブジェクトにアクセスできる他の便利な機能を提供します。

[Send telegram directly]ダイアログを使用して、任意のPDO（プロセスデータオブジェクト）を送信できます。ダイアログを表示するには、[Send telegram directly]の [Commands]メニューを選択します。このダイアログ以外に、10.1.6項、71ページの*TRANSMIT*コマンドも使用できます。

PDOのマッピングの表示および変更は、7.3.3.2項、51ページで説明するダイアログを使用します。

7.3.3.1 オブジェクトブラウザ

オブジェクトブラウザにより、オブジェクトディクショナリの全てのエントリを表示して変更することができます。オブジェクトブラウザは、クイックアクセスツールバーの[Object browser]ボタン、または[Configuration]メニューを使用して、開くことができます。

オブジェクトブラウザを表示するには、接続されたコントローラのファームウェアに適合するXML形式のEDSファイル（XDDファイル）がMotion Managerのインストールに含まれている必要があります。

含まれていない場合は、ブラウザの起動時に適切なファイルをロードしてインポートする必要があります。

レジスタ

- 通信: CiA 301に準拠した通信オブジェクト
- メーカー: メーカー固有のオブジェクト
- 機器: CiA 402に準拠した駆動機器プロファイルのオブジェクト

変更できない読み取り専用のパラメータは、[new value]欄の「-」で識別されます。他の全ての値は各行をダブルクリックし、新しい値を入力して変更します。

ボタン

ボタン	意味	説明
	アップデート	オブジェクトディクショナリがアップデートされ、全てのオブジェクトが機器から再度読み込まれます。
	転送の変更	変更された設定が機器にロードされます。
	オブジェクトディクショナリの保存	オブジェクトディクショナリの現在の値を機器に恒常的に保存します。
	構成のロード	前回保存されたXDC構成ファイル、または新しいXDD機器記述ファイルをロードします。
	現在の構成をファイルとして保存	現在のパラメータ構成がXDCファイルとして保存されます。
	ヘルプを開く	状況に応じたヘルプ目次が表示されます。
	フィルタオブジェクト	出力には、入力された検索テキストに一致するオブジェクトのみが表示されます。このために、[Actual value]欄および[New value]欄のエントリは無視されます。 以下の規則は、検索テキストの入力に適用されます。 <ul style="list-style-type: none">▪ 「+」を使用すると、複数の検索式を組み合わせ使用できます。▪ 検索式の後に「*」を挿入すると、検索式で開始される全てのエントリが選択されます。▪ 通常、大文字と小文字は区別されません。

 PLCおよび他のCANopenツールに組み込まれるさまざまな標準ファームウェアのEDSファイルは、\Motion Manager 6\EDSのインストールファイルに含まれています。

機器構成の読み込み

オブジェクトブラウザを起動、または[Update]ボタンをクリックすると、オブジェクトディクショナリ全体が現在の値とともにコントローラから読み込まれます。使用できるオブジェクトはXDDファイルから取得されます。

機器構成の変更

変更可能なパラメータ行をダブルクリックすると、入力ウィンドウが開きます。確認後、入力された値は[New value]行に適用されます。

[Load changes]ボタンをクリックすると、変更された全ての値がコントローラにロードされます。変更をコントローラに恒常的に保存するには、[Save object directory]ボタンをクリックします。

機器構成の保存

[Save current configuration as a file]ボタンをクリックすると、最後の設定がコントローラからXDCファイル（XML機器構成ファイル）に読み込まれます。

機器構成のロード

前回保存された機器構成が、XDCファイルの形式で使用可能な場合は、[Load configuration]ボタンを使用して、ファイルをオブジェクトブラウザにロードできます。

[Load changes]ボタンをクリックすると、変更された全ての値がコントローラにロードされます。変更をコントローラに恒常的に保存するには、[Save object directory]ボタンをクリックします。

7.3.3.2 PDOマッピング

[PDO mapping]ダイアログは、CANopenコントローラのPDOのマッピング構成に使用する便利な機能を提供します。ダイアログは、[Configuration]メニューから開くことができます。

PDOは、CANopenプロトコルのプロセスデータオブジェクトで最大長は8バイトです。

ダイアログには、コントローラに対応した受信および送信PDOのマッピングに使用するタブがあります。

- 受信PDOはコントローラ（RxPDOs）によって受信されます。
- 送信PDOはコントローラ（TxPDOs）によって送信されます。

PDOのデータ内容は、マッピングの各方向をサポートするCANopenオブジェクトディクショナリの任意のパラメータで構成できます。PDO内のパラメータ数は、PDOの最大長8バイトに依存します。

ボタン

ボタン	機能
	構造ツリー（左のウィンドウ）で強調表示されたPDOのリストからパラメータを割り当てます。パラメータはドラッグ&ドロップでも割り当てることができます。
	強調表示されたPDOを構造ツリー（左のウィンドウ）から削除します。オプションとして、[Del]キーを使用してパラメータを削除することもできます。

[Send]ボタンをクリックすると、表示されたPDOマッピングがコントローラにロードされます。設定を恒常的にコントローラに保存するには、[Save]ボタンもクリックします。

7.3.4 LSS

LSS（レイヤー設定サービス）プロトコルの機能をCiA 305に従って使用する場合は、メニュー項目 **[Configuration] [connection parameters]** を使用して、FAULHABER CANOpenノードのノード番号と転送速度を変更できます。

LSSプロトコルは、構成のオプションを2つ提供します。

構成の種類	説明
グローバルモード変更	接続された全てのLSSスレーブが構成モードに切り替えられます。LSSスレーブが1つだけ接続されている場合は、ボーレートおよびノードID（ノードアドレス）のみが設定できます。
セレクトティブモード変更	ベンダID、製品コードおよびシリアル番号が確認できるLSSスレーブ1つのみが構成モードに切り替えられます。このモードは、ネットワーク上の各駆動機器をそのシリアル番号によって構成するために使用できます。

LSSモードで使用できる専用のダイアログがあります。これにより各ノードを一括で構成、またはLSSデータ（ベンダID、製品コードおよびシリアル番号）が入力されるノードを選択して構成することができます。

- Node Explorerを使用して前回構成されたノードを再構成する（別のノード番号の割り当てなど）場合、そのLSSデータが直接表示されます。[Next]をクリックして、データを確定する必要があります。
- 構成されていないノード（ノードID = 0xFF）をネットワーク上で構成する場合は、ベンダIDおよび製品コードが事前に割り当てられ、選択されている必要があります。さらに必要なノードのシリアル番号を入力する必要があります。1つのモータが接続されている場合は、グローバルモード変更を選択でき、それ以上のデータを指定する必要はありません。

MC V3.xファミリのモーションコントローラの場合は、以下のデータが必要です。

- ベンダID: 327
- 製品コード: 48
- 改訂番号: 1.0

MC V2.xファミリのモーションコントローラの場合は、以下のデータが必要です。

- ベンダID: 327
- 製品コード: 3150
- 改訂番号: 0.0

7.4 グラフィック解析

Motion Managerのグラフィック解析（追跡）を使用すると、モータの動作を柔軟に監視して評価することができます。潜在的な使用目的の1つは、モータおよびコントローラの動作評価、またはコントローラパラメータの最適化（ステップ応答の記録）です。

グラフィック解析は、クイックアクセスツールバーの[**Graphical analysis**]ボタンまたは[**Tools**]メニューを使用して、開始することができます。

i オプションの設定はコントローラごとに異なります。各コントローラファミリの関連する章を参照してください。

- MC V2.xについては、7.1.5項、35ページを参照してください。
- MC V3.xについては、7.2.8項、45ページを参照してください。

7.4.1 メインツールバー

グラフィック解析のツールバーにより、選択した機能に直接アクセスできます。

アイコン	機能
	記録の開始と停止
	<p>X軸のスケーリング</p> <ul style="list-style-type: none"> ■ アクティブ: 記録の全期間が表示されるように、自動的にX軸をスケーリングします。 ■ 非アクティブ（デフォルト）: X軸のスケーリングは、表示エリア全体に対する入力フィールドの値（ミリ秒）に対応します。 ■ 手動スケーリングにより、ミリ秒で表示された時間窓
	<p>ズームモードの有効化／無効化。ズームモードが有効な場合は、以下のマウス機能を使用できます。</p> <ul style="list-style-type: none"> ■ マウスの左ボタンおよびマウスホイールは、選択したウィンドウのセクションの拡大／縮小に使用できます。 <ul style="list-style-type: none"> ■ ウィンドウのセクションを拡大するには、カーソルの十字を記録ウィンドウの最上部に置き、左ボタンを押した状態で十字を下にドラッグします。 ■ ウィンドウのセクションを縮小するには、カーソルの十字を記録ウィンドウの最下部に置き、左ボタンを押した状態で十字を上をドラッグします。 ■ 右ボタンを押した状態で、ウィンドウのセクションを移動できます。
	<p>2つのデータポイント間のX方向およびY方向の距離を計算できるように計算モードを有効にします。</p> <ul style="list-style-type: none"> ■ 複数のパラメータが記録された場合は、曲線を選択します。特定のY軸に対して距離を計算する必要があるためです。選択を解除するには再度右クリックします。 ■ マウスの左ボタンを使用して、曲線上のポイント、またはエリアの任意のポイントを選択します。 ■ それ以外のポイントを選択します。選択したポイント、およびX方向とY方向の距離が表示されます。 ■ 計算するには、任意のポイントをクリックします。
	追跡設定ウィンドウが開きます。
	表示されたグラフの印刷に使用するダイアログが表示されます。
	<p>オプションとして、現在表示されている記録をビットマップファイルまたはテキスト形式のCSVファイルとして保存できます。</p> <p>CSVファイルは、数学的プログラムまたは表計算プログラムに読み込み、さらに処理することができます。</p>

7.5 マクロ機能

Motion Managerのマクロ機能は、コントローラでサポートされているコマンドで自由に定義できる一連のシーケンスを提供します。コマンドは、マウスのクリック1つでコントローラに送信されます。

この機能は、[Tools]-[Macros]、またはクイックアクセスツールバーを使用して呼び出すことができます。

各モーションコントローラで使用できるマクロのセットがあります。セットは必要に応じて拡張または変更することができます。作成されたマクロはボタンに割り当てることができます。ボタンをクリックすると実行出来ます。

 ボタンの追加: 割り当てられていないボタンを追加します

 ボタンの削除: 選択されたボタンを削除します

 マクロの編集: マクロ編集のエディタを開きます

 プルダウンメニューまたはボタンのコンテキストメニューを使用し、マクロを選択してボタンに割り当てることができます。

7.5.1 マクロの編集

マクロ編集のエディタで、あらかじめ定義されたマクロを拡張および変更できます。

マクロの定義は、以下の規則に従う必要があります。

- 各マクロに、角括弧で囲まれたマクロ名が先頭に配置されたコマンドブロックが含まれていること
- 使用可能なコマンドが、使用されるコントローラに対するMotion Managerコマンドリファレンスまたはコマンドリファレンス（10.1節、66ページ）に含まれていること。
- キーワードWAITを使用して、ミリ秒単位の遅延を2つのコマンドの間に挿入できること（例: WAIT(1000)、1000msの遅延を作成します）。
- コメントの前に「;」が挿入されていること。

[Apply]ボタンを使用すると、変更が適用されてマクロファイルに保存されます。

マクロの実行中、各コマンドは1行ずつロードされます。

 事前に定義されたマクロファイルは、マクロ編集のエディタにより導入した内容を完全に削除し変更を適用すると、再生できます。

8 追加機能

8.1 VB Scriptプログラム

FAULHABER Motion Managerは、簡素化された自動化スクリプトを作成する機能、およびMotion ManagerがインストールされたPC環境でスクリプトを実行できる機能を提供します。入力されたVB Scriptのコードは、Microsoft Windowsに組み込まれたスクリプトホストを介して実行されます。

 Microsoftスクリプトホストの時間応答は厳密ではありません。そのため、使用するスクリプトプログラムは事前にテストしたプログラムのみが対象となります。

生産運転の場合は、適切なプログラミングツールを使用してください。

8.1.1 スクリプトプログラムの作成

スクリプトプログラムは、Editorウィンドウで作成し、そこからロードできます。ファイルの拡張子は「*.vbs」です。

スクリプトプログラムは、常に「SUB MAIN」で開始され「END SUB」で終了します。VB Scriptコードは、これらの区切り文字の間に書き込みます。Motion Managerには、スクリプトプログラムの作成が簡単にできるようにコードサンプルが用意されています。

 VB Scriptコマンドセットの説明は、VB ScriptのMicrosoft Helpを参照してください（[Help] - [Help for VB Script]）。

 VB Scriptプログラムサンプルが、Motion Manager 6\Examplesの下にインストールフォルダに格納されています。

8.1.2 スクリプトプログラムの開始と停止

エディタでVB Scriptファイルを開くと、以下の制御ツールが[File]タブのツールバーに表示されます。

アイコン	機能
	スクリプトプログラムの実行: スクリプトプログラムが実行されます。スクリプトプログラムの実行中、エディタのバックグラウンドはグレーで表示されます。この間はプログラムコードを変更できません。 エラーが発生すると、スクリプトプログラムが停止してエラーメッセージが表示されます。さらに、スクリプトプログラムが停止した行が赤色で協調表示されます。 実行が終了すると、編集モードは編集エリアで再起動されます。
	プログラムの終了: 実行中のスクリプトを終了し、編集エリアで編集モードを再起動します。
	Extras (追加): コードテンプレートのリストを表示および非表示にします。 コードテンプレートには、代表的なプログラム構成要素が含まれ、ドラッグ&ドロップでスクリプトプログラムに組み込むことができます。追加的なコードテンプレートは、スクリプトプログラム内のコードのエリアを強調表示し、ドラッグ&ドロップでコードテンプレートのリストに取り込み定義できます。

 スクリプトプログラムは、Motion ManagerがインストールされたPC環境で実行され、コントローラにはロードすることはできません。実行中のスクリプトとコントローラの間ではコマンドとデータのみが交換されます。

8.1.3 Motion Managerの一般的な機能

VB Scriptコマンドセットは、モーションコントローラとデータを交換するために、特殊なMotion Managerの機能が追加され拡張されています。これらの機能は常にオブジェクト識別子「MC.」を使用して呼び出す必要があります。



Motion Managerの機能に、自動統合 (**Ctrlキー+スペースキー**) を使用すると、素早くアクセスできます。

8.1.3.1 Sendコマンド

構文:

```
MC.SendCommand (command)
```

説明:

ASCIIコマンドをモーションコントローラに送信します。Motion Managerのコマンドは10.1節、66ページに一覧表示されています。



オブジェクトディクショナリでオブジェクトの読み込みまたは書き込みをするMotion Managerのコマンドが使用される場合は、その後、返信を読み込むためにMC.WaitAnswer()を呼び出す必要があります。

MC V2.xファミリのRSインターフェース付きモーションコントローラの場合、指定されたコマンドはコントローラに直接送信されます。これらのコマンドは、MC V2.xファミリのCFインターフェース付きモーションコントローラにも使用できます。この場合、コマンドはCAN信号により送信されます。

パラメータ:

command (string): モーションコントローラに送信されるコマンド

例:

```
MC.SendCommand ("V100")
```

8.1.3.2 WaitAnswer

構文:

```
answer = MC.WaitAnswer(timeout, answMode)
```

説明:

モーションコントローラからの返信を指定された時間（ミリ秒）待ちます。

パラメータ:

answer (string): Antwort, die vom Motion Controller eingelesen wurde

answer = "": Keine Antwort nach timeout

timeout (int): Ganzzahliger Wert in ms, bis Antwort eintreffen muss

answMode (int): Ganzzahliger Wert für die Interpretation der eingelesenen Daten

answMode = 0: Antwort-Daten werden als Integer interpretiert

answMode = 1: Antwort-Daten werden als String interpretiert

answMode > 255: Es wird auf eine Nachricht mit der hier angegebenen

COB-ID bzw. Knotennummer (LowWord) mit Befehls-ID (HighWord) gewartet

追加のanswModeの値は、MC V2.x グループのRSインターフェース付きモーションコントローラに使用できます。

answMode = 10: Asynchrone Nachrichten und Quittierungen werden ignoriert

answMode = 11: Quittierungen werden ignoriert

answMode = 12: Asynchrone Nachrichten werden ignoriert

answMode = 13: Asynchrone Nachrichten bis auf v, e und h werden ignoriert

例:

```
a = MC.WaitAnswer(1000,0)
```

8.1.3.3 WriteToHistory

構文:

```
MC.WriteToHistory(text)
```

説明:

指定されたテキスト行をMotion Managerのログウィンドウに書き込みます。

パラメータ:

text (String): ログウィンドウに出力されるテキスト

例:

```
MC.WriteToHistory(" Position 1 reached ")
```

8.1.3.4 CloseCom

構文:

```
MC.CloseCom
```

説明:

現在アクティブな通信インターフェースを閉じます。

8.1.3.5 OpenCom

構文:

```
ret = MC.OpenCom
```

説明:

選択された通信インターフェースを開きます。

パラメータ:

ret (int): 整数の戻り値

ret = 1: インターフェースが正常に開きました。

ret = 0: インターフェースを開く際にエラーが発生しました。

例:

```
ret = MC.OpenCom
IF ret = 1 THEN
    MsgBox(„Interface successful opened! “)
ELSEIF ret = 0 THEN
    MsgBox(„Error opening Interface! “)
END IF
```

8.1.3.6 CmdExecute

構文:

```
MC.CmdExecute (command)
```

説明:

指定されたコマンドをシステムレベルで実行します。この機能は外部プログラムの起動などに使用できます。

パラメータ:

コマンド (文字列) : 実行されるコマンドまたはアプリケーションのファイル名を、必要に応じてファイルパスとともに指定する文字列。

例:

```
MC.CmdExecute ("C:\tools\mytool.exe")
```

8.1.3.7 UpdateWindows

構文:

MC.UpdateWindows

説明:

他のMC機能を使用できない場合は、使用目的の時間の他の部分をアップロードに割り当てるために、この機能をループ内で呼び出すことができます。

8.1.4 オブジェクトディクショナリを使用したモーションコントローラの関数

8.1.4.1 GetObj

構文:

```
answer = MC.GetObj(nodeNr, index, subIndex, dataType)
```

説明:

オブジェクトディクショナリからパラメータを読み込みます。

パラメータ:

answer (string)パラメータ値を読み込みました

answer = “” : パラメータ値を読み込めませんでした

nodeNr (int):ノード番号

index (int): オブジェクトエントリのインデックス

subIndex (int): オブジェクトエントリのサブインデックス

dataType (int): 読み取るパラメータのデータ型

dataType = 0:整数

dataType = 1:文字列

例:

```
value = MC.GetObj(1, &h6067, &h00, 0)
```

8.1.4.2 SetObj

構文:

```
ret = MC.SetObj (nodeNr, index, subIndex, value, len)
```

説明:

新しい数値をオブジェクトディクショナリの整数パラメータに書き込みます。

パラメータ:

```
ret (int): 整数の戻り値  
    ret = 1: 関数が正常に実行されました。  
    ret = 0: 関数の実行中のエラー  
nodeNr (int): ノード番号  
index (int): オブジェクトエントリのインデックス  
subIndex (int): オブジェクトエントリのサブインデックス  
value (int): パラメータの新しい値  
len (int): バイトで表したパラメータのデータ長
```

例:

```
ret = MC.SetObj (1, &h6067, &h00, 30, 4)  
IF ret = 0 THEN  
    MsgBox( "Error writing Parameter" )  
END IF
```

8.1.4.3 SetStrObj

構文:

```
ret = MC.SetStrObj (nodeNr, index, subIndex, value)
```

説明:

新しい文字列の値をオブジェクトディクショナリの文字列パラメータに書き込みます。

パラメータ:

```
ret (int): 整数の戻り値  
    ret = 1: 関数が正常に実行されました。  
    ret = 0: 関数の実行中のエラー  
nodeNr (int): ノード番号  
index (int): オブジェクトエントリのインデックス  
subIndex (int): オブジェクトエントリのサブインデックス  
value (int): パラメータの新しい値
```

8.1.5 MC V2.xファミリのRSインターフェース付きモーションコントローラの関数

8.1.5.1 ComXonXoff

構文:

```
MC.ComXonXoff (on)
```

説明:

シリアルRS232インターフェース付きモーションコントローラのXon/XoffプロトコルおよびASCIIプロトコル (MC V2.x) を有効または無効にします。大量のデータが返信を待たずに連続して送信される場合は、アクティブ化が必要です。

パラメータ:

on (int): Xon/Xoffプロトコルを有効または無効にする整数の入力値
on = 1: Xon/Xoffプロトコルを有効にします。
on = 0: Xon/Xoffプロトコルを無効にします。

例:

```
MC.ComXonXoff (1)
```

8.1.5.2 SendBin

構文:

```
MC.SendBin (value)
```

説明:

2進数値をコントローラに送信します。

パラメータ:

value (int): 送信される8ビットの値
2進数値は、システムパラメータの設定のみに使用され、通常ユーザはアクセスできません。

8.1.5.3 SetBinMode

構文:

```
MC.SetBinMode (mode1, mode2)
```

説明:

以下を使用して、パラメータ1および2のバイナリモードを設定します。

- MC V2.xファミリのRSインターフェース付きモーションコントローラのバイナリインターフェース
 - MC V2.xファミリのCFインターフェース付きモーションコントローラの追跡チャンネル
- 次に、BinRequest()関数を使用して、最大2つの変数を同時に読み込むことができます。

パラメータ:

mode1 (int): 最初に読み込まれるパラメータの整数値
mode2 (int): 2番目に読み込まれるパラメータの整数値
mode1およびmode2に使用できる値は、各コントローラのマニュアルを参照してください。

8.1.5.4 BinRequest

構文:

```
ret = MC.BinRequest(timeout)
```

説明:

データ要求。現時点でSetBinMode()によって設定されたパラメータを読み込みます。結果は、以下の属性を使用して読み込むことができます。

- MC.BinVal1
- MC.BinVal2
- MC.BinTimecode

パラメータ:

ret (int): 整数の戻り値

ret = 1: データを受信しました。

ret = 0: タイムアウト時間内にデータを受信しませんでした。

timeout (int): 返信を受信するまでの整数値 (ミリ秒)

例:

```
CALL MC.SetBinMode(0,1) ' Ist- und Solldrehzahl einlesen
IF MC.BinRequest(500) THEN ' Datenanforderung
    Istdrehzahl = MC.BinVal1
    Solldrehzahl = MC.BinVal2
    Timecode = MC.BinTimecode
END IF
```

8.2 ファームウェアのアップデート

組み込まれたファームウェアのアップデート機能により、接続されたFAULHABER製コントローラのファームウェアの確認とアップデートが可能になります。始動後、メニュー項目[Extras] - [Firmware Update]から、2つの機能が選択できます。

機能	説明
アップデートの有無の確認 Check for update	機器の現在のファームウェアで使用可能なアップデートを確認します。使用可能なアップデートがある場合は、機器のファームウェアをアップデートできます。
ファームウェアファイルのロード Load firmware	FAULHABERが別途提供するファームウェアファイルをロードし、コントローラに転送できます。

i ファームウェアのアップデートには、アップデートインターフェースとして接続されたコントローラによりサポートされるインターフェースのみが使用できます。

8.3 オフラインモード

仮想デバイスは、オフラインモードで管理および有効化できます。これにより、適合するハードウェアを使用できない場合でも、Motion Managerのコントローラ固有の機能を実行することができます。

仮想機器との通信は、ターミナルで実際の状態と同じように表示されます（5.5章、22ページ参照）が、通信プロトコルの全てのサービスのサポートが、全ての仮想装置に保証されているわけではありません。

i 仮想機器は、機能の範囲が限られます。主に1組の構成を表すことを目的としています。仮想機器による構成を実際の機器に転送しないでください。

8.3.1 モーションコントローラ

8.3.1.1 仮想モーションコントローラへの接続の確立

1. メニュー[Extras] - [Offline Mode] - [Motion Controller]を使用して、使用可能な仮想モーションコントローラのリストを開きます。
2. チェックボックスを使用して、必要なモーションコントローラを有効にします。
↳ アクティブな機器のみに接続を確立できます。
3. [OK]ボタンを使用して選択を確定します。
4. モーションコントローラとの接続を確立します（6.1節、26ページ参照）。

i ポート番号が0の通信インターフェースは、仮想機器のみに使用できます。

8.3.1.2 仮想モーションコントローラの管理

[Extras] - [Offline Mode] - [Motion Controller]メニューの以下の機能は、仮想モーションコントローラの管理に使用できます。

ボタン	機能
New	新しい仮想モーションコントローラの機器ファミリー、機器名、およびノード番号を選択するウィンドウが開きます。 <ul style="list-style-type: none">▪ 仮想機器として表わされるFAULHABERモーションコントローラを選択できます。▪ 複数の機器が接続されている場合は、通信が妨害されないように一意のノード番号を使用する必要があります。
Import	新しい仮想モーションコントローラの作成に使用する構成ファイル (*.xdc) を選択するウィンドウが開きます。
Edit	選択された仮想モーションコントローラを編集するウィンドウが開きます。事前定義された機器は編集できません。
Delete	選択された仮想モーションコントローラを削除します。事前定義された機器は削除できません。

8.3.2 スピードコントローラ

FAULHABERスピードコントローラには、アクティブなコマンドインターフェースがありません。モーションコントローラとは異なった方法で設定する必要があります。

1. 任意のCOMインターフェースを使用して、スピードコントローラへの接続を確立します（6.1節、26ページ参照）。
 - 👉 記号SCxxxxが付いたノードがNode Explorerに表示されます。
2. Node Explorerでノードを選択します。
3. **[Extras] - [Offline Mode] - [Speed Controller]**メニューで、機器の記号および仮想スピードコントローラのファームウェア番号を選択します。
 - 👉 選択された種類の標準的な構成を表示できます。

9 保証

Dr. Fritz Faulhaber GmbH & Co. KGの製品は、最も近代的な製造方法を使用して製造され、厳しい品質検査を受けます。全ての販売業務および納品は、FAULHABERのホームページ www.faulhaber.com/agbで閲覧でき、このホームページからダウンロードできる、当社の販売および納品に関する一般条件に基づき実施されます。

10 付録

10.1 Motion Managerコマンドリファレンス

以下のタスクのコマンドは、Motion Managerで使用できます。

- ターミナルウィンドウのコマンド入力フィールドへの直接の入力
- マクロ内での使用
- 関数MC.SendCommand()を用いたVB Scriptプログラム内での使用

i 本章に記載されたコマンドは、Motion Manager環境内でのみ使用してください。コマンドはMotion Managerによって解釈され、各インターフェースのプロトコルに変換されます。コマンドはシーケンスプログラム内では使用できません。または駆動機器が他のホストシステムによって駆動されている場合も使用できません。

i ネットワークの任意のノードは、あらかじめ割り当てられた10進のノード番号を使用して、アドレスを指定できます。ノード番号が指定されていない場合は、Motion Managerでアクティブなノードにアドレスを指定されます。大文字と小文字は区別されません。

10.1.1 CANopenステートマシンの制御

表3: CANopen NMTステートマシンのコントローラに対するコマンド

コマンド	機能	CAN信号
START	リモートノードの開始（モーションコントローラは、 <i>運転状態</i> に切り替えられます）	Id 0x000: 0x01 Node ID
STOP	リモートノードの停止（モーションコントローラは、 <i>停止状態</i> に切り替えられます）	Id 0x000: 0x02 Node ID
PREOP	運転前の状態にする（モーションコントローラは、 <i>運転前状態</i> に切り替えられます）	Id 0x000: 0x80 Node ID
RESET	ノードのリセット	Id 0x000: 0x81 Node ID
RESETCOM	通信のリセット	Id 0x000: 0x82 Node ID
STARTALL	全リモートノードの開始（モーションコントローラは、 <i>運転状態</i> に切り替えられます）	Id 0x000: 0x02 0x00

表4: CiA 402制御ワードを制御するコマンド

コマンド	機能	CiA 402制御ワード (0x6040)
SHUTDOWN	停止（モーションコントローラは <i>Ready to Switch On</i> 状態に切り替えられません）	0x0006
SWITCHON	電源投入（モーションコントローラは <i>Switched On</i> 状態に切り替えられません）	0x0007
DISABLE	電圧が無効（モーションコントローラは <i>Switched On Disabled</i> 状態に切り替えられます）	0x0000
QUICKSTOP	クイック停止（モーションコントローラは <i>Quick Stop Active</i> 状態に切り替えられます）	0x0002
DIOP	操作が無効（モーションコントローラは <i>Switched On</i> 状態に切り替えられません）	0x0007
ENOP	操作が有効（モーションコントローラは <i>Operation Enabled</i> 状態に切り替えられます）	0x000F
FAULTRESET	不良リセット（モーションコントローラは <i>Switched On Disabled</i> 状態に切り替えられます）	0x0080
MA	絶対位置移動（PP）	0x003F
MR	相対位置移動（PP）	0x007F
HS	原点復帰開始（HM）	0x001F

例: ノード10を*運転*状態に設定します。

- ▶ 10 START
 - ↳ ノード10は*運転*状態に設定されます。

例: Motion Managerのアクティブな駆動機器を*Operation Enabled*状態に切り替えるための電源投入シーケンス

- ▶ SHUTDOWN
- ▶ SWITCHON
- ▶ ENOP
 - ↳ 駆動機器は現在、*Operation Enabled*ステートです。

i PP動作モードでは、制御ワード（新しい設定点）内のビット4の立ち上がりフランクのみに、新しい目標位置が適用されます。このため、MAまたはMRコマンドを実行する前に、ENOPコマンドなどを使用して、このビットが再度リセットされることを確認する必要があります。

10.1.2 オブジェクトディクショナリのオブジェクトの記述

引数を持つコマンドは、コマンドの後に10進数の数値が必要です。

正常に実行されると、ターミナルウィンドウまたはスクリプト関数MC.WaitAnswer()に値[OK]が返されます。実行が失敗すると、プレーンテキストのSDOエラーメッセージが返される、またはタイムアウトエラー（空の文字列）が返されます。

表5: オブジェクトディクショナリにオブジェクトを書き込むコマンド

コマンド	機能	オブジェクト
引数を持つコマンド		
OPMOD	オペレーティングモードの構成	0x6060.00（動作モード）
SPOS	実際の位置の指定（PP）	0x607A.00（目標位置）
V	目標位置の指定（PV）	0x60FF.00（目標速度）
HM	原点復帰モード（HM）の設定	0x6098.00（原点復帰モード）
U	出力電圧（出力モード）の指定	装置依存
SP	最大速度の設定	0x6081.00（プロファイル速度）
AC	加速勾配をの構成	0x6083.00（プロファイル加速）
DEC	減速勾配の構成	0x6084.00（プロファイル減速）
引数のないコマンド		
SAVE	全てのパラメータの保存	0x1010.01（全てのパラメータの保存）
SAVE_ALL	全てのパラメータの保存	0x1010.01（全てのパラメータの保存）
SAVE_COM	通信パラメータの保存	0x1010.02（全ての通信パラメータの保存）
SAVE_APP	アプリケーションパラメータの保存	0x1010.03（全てのアプリケーションパラメータの保存）
RESTORE	出荷時の全てのパラメータをロード	0x1011.01（全てのパラメータの復元）
RESTORE_ALL	出荷時の全てのパラメータをロード	0x1011.01（全てのパラメータの復元）
RESTORE_COM	出荷時の通信パラメータをロード	0x1011.02（通信パラメータの復元）
RESTORE_APP	出荷時の通信アプリケーションパラメータをロード	0x1011.03（アプリケーションパラメータの復元）

例: プロファイル位置モード（PP）を設定し、モータを相対的に10,000（または設定された位置決め単位）だけ動かします。

▶ 以下のコマンドシーケンスを送信します。

- **OPMOD 1**
- **SPOS 10000**
- **MR**

🔗 モータは、プロファイル位置モードで10,000（または設定された位置決め単位）ずつ移動されました。

10.1.3 オブジェクトディクショナリのオブジェクトの読み込み

正常に実行されると、ターミナルウィンドウまたはスクリプト関数MC.WaitAnswer()（10進形式の数値）に返信が文字列として返されます。実行が失敗すると、プレーンテキストのSDOエラーメッセージが返される、またはタイムアウトエラー（空の文字列）が出力されます。

表6: オブジェクトディクショナリのオブジェクトを読み込むコマンド

コマンド	機能	オブジェクト
GTYP	機器名の読み込み	0x1008.00（メーカーの機器名）
VER	ソフトウェアバージョンの読み込み	0x100A.00（メーカーのソフトウェアバージョン）
GSER	シリアル番号の読み込み	0x1018.04（シリアル番号）
GSW	ステータスワードの読み込み	0x6041.00（ステータスワード）
GOPMOD	設定されたオペレーティングモードの読み込み	0x6061.00（動作表示モード）
POS	実際の位置の読み込み	0x6064.00（実位置の値）
POSI	実際の位置の読み込み（内部単位）	0x6063.00（実位置の値）
TPOS	目標位置の読み込み	0x6062.00（位置要求値）
GV	目標速度の読み込み	0x606B.00（速度要求値）
GN	実際の速度の読み込み	0x606C.00（実速度値）
GRC	実際の電力消費の読み込み	0x6078.00（実電流値）
GSP	最大速度の読み込み	0x6081.00（プロファイル速度）
GAC	加速勾配の読み込み	0x6083.00（プロファイル加速）
GDEC	減速勾配の読み込み	0x6084.00（プロファイル減速）

例: 実位置を読み込みます。

▶ 以下のコマンドを送信します。

- **POS**

🔍 実位置の実際の値が読み込まれました。

10.1.4 オブジェクトディクショナリでの任意のオブジェクトの書き込み

16進数表記でインデックス (xxxx) およびサブインデックス (yy) を指定することで、オブジェクトディクショナリのオブジェクトをアドレス指定できます。入力する値は、16進引数として指定する必要があります。16進引数の長さ (データバイト数) は、書き込まれるオブジェクトの種類に必要な長さに対応する必要があります。最低値のバイトは右端に配置されます。

- Int8 = 1バイト
- Int16 = 2バイト
- Int32 = 4バイト

引数のバイトは、2つの16進数文字で表わされます (00...FF)。

正常に実行されると、ターミナルウィンドウまたはスクリプト関数MC.WaitAnswer()に値[OK]が返されます。実行が失敗すると、プレーンテキストのSDOエラーメッセージが返される、またはタイムアウトエラー (空の文字列) が出力されます。

表7: オブジェクトディクショナリに任意のオブジェクトを書き込むコマンド

コマンド	機能
SOBJ xxxx.yy	オブジェクトの書き込み (オブジェクトの設定)

例: ノード10の加速値を値500に変更します (プロファイル加速0x6083.00, Int32)。

- ▶ 以下のコマンドシーケンスを送信します。
 - 10 SOBJ 6083.00 000001F4

🔗 加速値は500に変更されました。

i SLOBJコマンドは、文字列オブジェクトに使用できます。このために、文字列のASCII文字を引数として設定し、左から右へと読む必要があります。

10.1.5 オブジェクトディクショナリでの任意のオブジェクトの読み込み

16進数表記でインデックス (xxxx) およびサブインデックス (yy) を指定することで、オブジェクトディクショナリのオブジェクトをアドレス指定できます。正常に実行されると、ターミナルウィンドウまたはスクリプト関数MC.WaitAnswer()に返信が文字列として返されます。実行が失敗すると、プレーンテキストのSDOエラーメッセージが返される、またはタイムアウトエラー (空の文字列) が出力されます。

表8: オブジェクトディクショナリに任意のオブジェクトを読み込むコマンド

コマンド	機能
GOBJ xxxx.yy	オブジェクトの読み込む (オブジェクトの取得)

例: ノード10の加速値を読み込みます。

- ▶ 以下のコマンドを送信します。
 - 10 GOBJ 6083.00

🔗 ノード10の加速値が出力されました。

i 戻りデータの中では、データ長が4バイトまでオブジェクトのエントリは、通常整数値として解釈されません。

GLOBJコマンドは、文字列オブジェクトの読み込みに使用できます。

10.1.6 信号の送信

CAN通信用のCOB-IDまたはRS232/USB通信用のノード番号 (XXX) を3桁の16進数表記で指定すると、基本的な下層のプロトコルの信号を送信できます。信号のデータバイトは、16進数値として指定し通信の位置 (左の最低値バイト) に配置する必要があります。

表9: オブジェクトディクショナリに任意のオブジェクトを読み込むコマンド

コマンド	機能
TRANSMIT xxx	信号の転送

例: RxPDO1を使用して、値 0x007Fを持つ制御ワードをノード10から送信します (COB-ID = 0x20A)。

- ▶ 以下のコマンドを送信します。
 - **TRANSMIT 20A 7F00**

10.2 問題の解決法

10.2.1 ポートが存在しない

Motion Managerは、5.4.2項、19ページに記載されたインターフェースおよびインターフェースプラグインに対応しています。各インターフェースに関する指示も記載されています。

10.2.2 ポートを開くことができない

ポートが別のアプリケーションによって使用されている場合、開くことができません。

- ▶ 必要なポートにアクセスできない場合は、開いているアプリケーションを確認し、閉じてください。
 - ↳ アプリケーションを閉じて、ポートが解除されない場合があります。
- ▶ ポートが解除されない場合は、PCを再起動してください。

10.2.3 接続された機器に接続できない

最も簡単に接続を確立するには、[Wizard for establishing a connection]を使用します。

- ▶ このウィザードを使用しても接続を確立できない場合は、以下の原因をチェックします。

考えられる原因	改善法
Motion Managerで選択したインターフェースに機器が接続されていません。	<ul style="list-style-type: none">▪ 正しいインターフェースまたは正しいポートを選択します。
機器のインターフェースがPCに正しく接続されていません。	<ul style="list-style-type: none">▪ 配線の取り付けが機器のマニュアルの指定通りであることを確認、またはマニュアル通りに接続します。
機器に十分な電圧が供給されていません。	<ul style="list-style-type: none">▪ 電圧源を確認します。
機器が、正しく構成されていないネットワークに配置されています。	<ul style="list-style-type: none">▪ ネットワークの構成を確認します（5.4.3項、21ページ参照）。
機器がbroadcastコマンドに反応しない場合は、接続を確立できません。	<ul style="list-style-type: none">▪ 接続の種類「Network」を選択し、適切な転送速度とスキャン範囲を指定します。

10.2.4 モータが始動しない

1. モータ選択ウィザードを使用して、接続されたモータに応じてコントローラを構成します。
2. その後、モータが動作しない場合は、次の考えられる原因をチェックします。

考えられる原因	改善法
モータが正しく接続されていません。	<ul style="list-style-type: none">▪ モータおよび機器のマニュアルの製品説明に従って、モータの接続を確認します。
モータに電力が供給されていません。	<ul style="list-style-type: none">▪ コントローラの電力供給がモータの電力供給から分離されているか確認します。
供給電圧が低すぎます。	<ul style="list-style-type: none">▪ コントローラの電圧範囲を確認します。
出力ステージがオンになっていません。	<ul style="list-style-type: none">▪ 出力ステージをオンにします（5.2節、17ページ参照）。
コントローラが障害状態です。	<ul style="list-style-type: none">▪ メニューコマンド[Commands] - [Device Control] - [Fault Resef]を実行します。
コントローラのオペレーティングモードが間違っています。	<ul style="list-style-type: none">▪ メニュー[Configuration] - [Drive functions]を使用して、オペレーティングモードを設定します。

10.3 エンドユーザライセンス契約

Dr. Fritz Faulhaber GmbH & Co. KGのFaulhaber Motion Managerに関する エンドユーザライセンス契約

1. Dr. Fritz Faulhaber、Gert Frech-Walter、Dr. Thomas Bertolini取締役代表されるDaimlerstraße 23、71101 SchönaichのDr. Fritz Faulhaber GmbH & Co. KG（以下「Faulhaber」と称す）

および

- 2 ユーザであるお客様（以下「ライセンシー」と称す）

間の

Dr. Fritz Faulhaber GmbH & Co. KGのFaulhaber Motion Managerに関する エンドユーザライセンス契約

序文

1. Faulhaberは、ミニチュアおよびマイクロドライブ・システムを設計し、生産する。Faulhaberは「Faulhaber Motion Manager」ソフトウェア（以下「MoMan」と称す）を開発した。これにより、Faulhaberが製造する特定のミニチュアおよびマイクロドライブ・システムの構成を変更し、パラメータを割り当てるためにこれらのシステムを制御できるようになる。詳細はMoManのプログラム記述にある。MoManは個人顧客の個別のニーズには対応していない標準的なソフトウェア製品である。FaulhaberはMoManを顧客に無償で提供する。
 2. ライセンシーは自社においてMoManを使用することを意図している。この詳細は第2条にある。
- これについて説明を受けた上で、両当事者は以下のライセンス契約を締結する。

§ 1

本契約の条件

- (1) 本契約の主題はFaulhaberがMoManの使用権をライセンシーに付与することである
- (2) MoManの構成要素は次のとおりである。
 - a) 機械可読オブジェクト・コード
 - b) プログラム記述を含めたユーザ文書

使用権の内容および範囲

- (1) Faulhaberは、MoManを自己の使用目的のために時間無制限で使用するための無制限の空間的かつ非排他的な権利をライセンシーに付与する。
- (2) 使用目的として、ライセンシーはFaulhaberが製造するミニチュアまたはマイクロドライブ・システムのサービスを提供し、構成を変更し、パラメータを設定するためだけにMoManを使用するものとする。「コミッショニング」は、Faulhaberが製造する個別のミニチュアまたはマイクロドライブ・システムの構成を変更し、パラメータを設定するためにMoManを使ってこれらのシステムを制御することを意味する。このように構成を変更しパラメータを設定する場合、個別のミニチュアまたはマイクロドライブ・システムの取扱説明書に常に気を配り、これに従うものとし、そこに示されている仕様をライセンシーは順守するものとする。ライセンシーは生産的活動においてMoManを使用しないものとする。「生産的活動」(Productive Operation)は、MoManを使ってFaulhaberが製造する個別のミニチュアまたはマイクロドライブ・システムを会社の進行中の活動において単体でまたは全体的なシステムのその他の構成要素と組み合わせて使用することを意味する。かかる使用はMoManの使用目的ではない。MoManを使ってFaulhaberが製造していないミニチュアおよびマイクロドライブ・システムを制御することにも、MoManを使ってFaulhaberが製造したがプロフラム記述には列挙されていないミニチュアおよびマイクロドライブ・システムを制御することにも同じ規定が適用される。
- (3) MoManを使用する権利は、特にFaulhaberドライブシステムなしで編集(変更)または配布する権利を含まない。この場合もMoManはあくまで無償で、かつ形式を変えずに配布されるものとし、本ライセンス契約の同封物の適用を受ける。ライセンシーは、MoManのソースコードまたはソースコード説明書を発行することをFaulhaberに要求する資格を有さない。ライセンシーは、期間限定のレンタルまたはリースという手段でソフトウェア、ドキュメンテーション、またはその一部分を第三者に譲渡する資格も有さない。第三者はライセンシー・グループ内の会社であるともみなされる。
- (4) それ以外で、MoManのさらなる使用、特に第三者への使用権の移転またはサブライセンスの付与は、Faulhaberの事前の明示的かつ書面の承諾を要するものとする。これはミニチュアおよびマイクロドライブ・システムの適切な使用にはMoManの使用が必要となることを条件として、これらの販売には適用されない。

§ 3

MoManの譲り渡し

- (1) 本契約と併せて付与される使用权を行使するために必要とされるMoManのコピーは、デジタルおよび機械可読形式でライセンシーに提供される。
- (2) ライセンシーはMoManのダウンロードによってユーザ文書のコピーも電子形式で受け取る。本ユーザ文書はプログラム記述も含む。

§ 4

業務範囲

特にFaulhaberに y 以下のサービスは本契約の主題ではない。

- a) ライセンシー敷地内でのMoManのインストール
- b) ライセンシーの要求事項に従ったMoManの変数パラメータの個別設定（カスタマイズ）
- c) ライセンシー向けの個別のプログラム拡張（個別の修正）
- d) ライセンシーのニーズに合わせたMoManインターフェースの調整
- e) ライセンシーのプログラムユーザを対象とした指導および研修
- f) MoManの保守、特に事後の新規プログラム・バージョンの供給

§ 5

権利の不備がある場合のライセンシーの請求

- (1) Faulhaberは、MoManの契約上の使用を妨げる第三者の権利なくMoManを譲渡することを約束する。
- (2) 第三者がかかる権利を主張した場合、ライセンシーは第三者によるかかる権利の主張をただちにFaulhaberに通知し、第三者により主張される権利からFaulhaberを防御するために必要な全ての法的権能および権限をFaulhaberに与えるものとする。
- (3) この場合（つまり上記（2）項の場合）、Faulhaberは自己の自由裁量により、以下のいずれかを行う権利を有する。
 - a) MoManの契約上の使用を妨げる第三者の権利またはこれらの主張を排除するため適した対策を講じる
 - b) 第三者の外部の権利が侵害されなくなるようにMoManを変更または交換する（ただし、これがMoManの保証される機能を損なわないこと）。Faulhaberがライセンシーの定める合理的な期間内にこれを行えない場合、ライセンシーは通知なくライセンス契約を終了する権利を有する。

§ 6

欠陥があった場合のライセンシーの請求

- (1) Faulhaberおよびライセンシーは、全ての適用条件においてエラーが発生しないソフトウェア・プログラムを開発するのは不可能だということに合意する。Faulhaberは、本契約が締結された時点で有効でありこれもダウンロード可能であるプログラム記述に従い、第2条に従った使用目的のための第3条によるダウンロードができるよう提供されているバージョンにおけるMoManの適合性を保証する。ただしライセンシーがサードパーティ・ソフトウェアとMoManを組み合わせた場合、Faulhaberはかかるサードパーティ・ソフトウェアとMoManの互換性に関して瑕疵担保責任を一切認めない。
- (2) プログラム記述から著しく逸脱する場合、Faulhaberはその後改良する義務を負う。契約上の使用が可能となるような妥当な期間内にFaulhaberがその後の改良を介して当該逸脱を排除または回避できない場合、またはその他の理由によりその後の改良が失敗に終わったとみなされる場合、ライセンシーは通知なくライセンス契約を終了する権利を有する。
- (3) 欠陥を理由とする請求は、ユーザ文書を含めたMoManのダウンロードから1年後に有効期限が切れる。時効期間の短縮は意図的な行為の場合には適用されない。

§ 7

賠償責任、補償

- (1) Faulhaberは基本的にライセンシーが第2条に基づいてMoManを使用目的通りに使用する場合に限り責任を負う。
- (2) Faulhaberは故意または重過失に起因する損失については無制限で責任を負う。死亡、身体障害、健康被害を引き起こす、または欠陥を故意に隠蔽したことに起因する落ち度のある損害に関する要求についても同じ規定が適用される。
- (3) Faulhaberは製造物責任法による製造物責任を負う。
- (4) Faulhaberはいわゆる「重要な契約上の義務」の違反に起因する損害について責任を負う。「重要な契約上の義務」(material contractua obligations)は、ライセンシーによる本契約締結において決定的であり、順守されると信頼できる根本的な契約義務のことである。Faulhaberが軽過失により重要な契約上の義務に違反した場合、結果生じる補償責任は本契約に関して典型的な予見可能である損失の補填に限定される。
- (5) Faulhaberは、データの損失または破壊が重過失または契約上もしくは法定上の義務の意図的な違反に起因しない場合、かかるデータの損失または破壊に対する賠償責任を一切認めない。ライセンシー敷地内でデータ損失が発生した場合、Faulhaberは最先端の標準に従って定期的にデータをバックアップしていても発生するデータ復元の典型的な費用の金額を上限として賠償責任を負う。
- (6) そうでない場合、法律上の理由にかかわらずFaulhaberによる補償責任は除外される。
- (7) 本ライセンス契約の終了後、ライセンシーはユーザ文書およびMoManのあらゆるバックアップ用コピーを含めてMoManを完全に削除するよう義務付けられている。

最終規定

- (1) 本契約の変更または追加は書面によるものとする。本条項に従わない場合、その変更または追加は無効である。書面による形式を必要とする本条項の変更にも同じ規定が適用される。
- (2) 本契約は、1980年4月11日付国際物品売買条約に関する国際連合条約（CISG）を除き、ドイツ連邦共和国の法律に準拠し、これに従って解釈されるものとする。
- (3) ライセンシーが商法の意味における経営者（businessperson）である、公法に基づく法人もしくは公法に基づく特別資産である、またはライセンシーがドイツ連邦共和国に登録事務所もしくは通常の居住地（定住所）を持たずに訴訟を起こす場合、司法上の唯一の裁判地はシュツットガルトである。
- (4) 本契約の規定が無効である、または無効になる場合、全てのその他の規定は影響を受けないままである。かかる場合、本契約当事者は法的に有効な方法で無効である規定の経済効果を最もよく達成する規定を作る作業に関わるよう義務付けられる。

〒140-0013

東京都 品川区 南大井 6-20-8

ユニゾ大森ビル 8F

新光電子株式会社

TEL. 03-6404-1003

FAX. 03-6404-1005

e-mail.motor-info@shinkoh-elecs.co.jp

www.shinkoh-faulhaber.jp

7000.05054, 3rd edition, 24-04-2017

© DR.FRITZ FAULHABER GMBH & CO.KG

仕様は予告なしに変更されることがあります。

DR.FRITZ FAULHABER

GMBH & CO.KG