

**リファレンスマニュアル**

**MULTIPROG 用  
同期モーション制御  
ファンクションブロック  
(PLCopen 仕様)**

# 目次

## はじめに

- 1) ……お願いと注意 …………… 1
- 2) ……データタイプ …………… 1

## 第1章 MULTIPROG 用 PLCopen MC ライブラリ

- 1-1 全体構成 …………… 1-1
- 1-2 MULTIPROG とは …………… 1-1
- 1-3 PLCopen MC 仕様とは …………… 1-1
- 1-4 MECHATROLINK-III とは …………… 1-5
  - 1-4-1 概要 …………… 1-5
- 1-5 EtherCAT 通信とは …………… 1-6
  - 1-5-1 概要 …………… 1-6
  - 1-5-2 導入手順 …………… 1-7

## 第2章 開発環境

- 2-1 インストール方法 …………… 2-1
- 2-2 I/O グループ設定方法 …………… 2-1
  - 2-2-1 Input 設定 …………… 2-1
  - 2-2-2 Output 設定 …………… 2-4
  - 2-2-3 ダミー変数の登録 …………… 2-7

## 第3章 ファンクションブロック

- 3-1 機能概要 …………… 3-1
- 3-2 使用方法 …………… 3-6
- 3-3 ファンクションブロックリファレンス …………… 3-11
  - 3-3-1 PLCopen 仕様 管理ファンクションブロック …………… 3-11
    - MC\_Power 関数 …………… 3-12
    - MC\_ReadStatus 関数 …………… 3-14

MC_ReadAxisError 関数	3-16
MC_ReadParameter 関数	3-17
MC_ReadBoolParameter 関数	3-18
MC_ReadByteParameter 関数	3-19
MC_ReadWordParameter 関数	3-20
MC_ReadDwordParameter 関数	3-21
MC_WriteParameter 関数	3-22
MC_WriteBoolParameter 関数	3-23
MC_WriteByteParameter 関数	3-24
MC_WriteWordParameter 関数	3-25
MC_WriteDwordParameter 関数	3-26
MC_ReadActualPosition 関数	3-27
MC_ReadActualVelocity 関数	3-28
MC_ReadActualTorque 関数	3-29
MC_Reset 関数	3-31
MC_CamTableSelect 関数	3-32
<b>3-3-2 PLCopen 仕様 動作ファンクションブロック</b>	<b>3-33</b>
MC_MoveAbsolute 関数	3-34
MC_MoveRelative 関数	3-36
MC_MoveAdditive 関数	3-38
MC_MoveSuperimposed 関数	3-40
MC_MoveVelocity 関数	3-41
MC_TorqueControl 関数	3-43
MC_Home 関数	3-45
MC_Stop 関数	3-50
MC_PositionProfile 関数	3-52
MC_VelocityProfile 関数	3-53
MC_AccelerationProfile 関数	3-54
MC_CamIn 関数	3-55
MC_CamOut 関数	3-56
MC_GearIn 関数	3-57
MC_GearOut 関数	3-59
MC_Phasing 関数	3-60
<b>3-3-3 PLCopen 仕様 原点サーチファンクションブロック</b>	<b>3-61</b>
MC_StepAbsSwitch 関数	3-62
MC_StepLimitSwitch 関数	3-65
MC_StepBlock 関数	3-68
MC_FinishHoming 関数	3-69
MC_StepRefPulse 関数	3-71
MC_StepDirect 関数	3-73
MC_StepAbsolute 関数	3-75
MC_StepReferenceFlyingSwitch 関数	3-77
MC_StepReferenceFlyingRefPulse 関数	3-78
MC_AbortPassiveHoming 関数	3-79
<b>3-3-4 PLCopen 仕様 MC Part4 管理ファンクションブロック</b>	<b>3-80</b>
MC_AddAxisToGroup 関数	3-81
MC_RemoveAxisFromGroup 関数	3-82
MC_UngroupAllAxes 関数	3-83
MC_GroupReadConfiguration 関数	3-84
MC_GroupEnable 関数	3-85

MC_GroupDisable 関数	3-86
MC_SetKinTransform 関数	3-87
MC_SetCartesianTransform 関数	3-88
MC_SetCoordinateTransform 関数	3-89
MC_ReadKinTransform 関数	3-90
MC_ReadCartesianTransform 関数	3-91
MC_ReadCoordinateTransform 関数	3-92
MC_GroupSetPosition 関数	3-93
MC_GroupReadActualPosition 関数	3-95
MC_GroupReadActualVelocity 関数	3-96
MC_GroupReadActualAcceleration 関数	3-97
MC_GroupReadStatus 関数	3-98
MC_GroupReadError 関数	3-100
MC_GroupReset 関数	3-101
MC_PathSelect 関数	3-102
MC_GroupSetOverride 関数	3-103
MC_SetDynCoordTransform 関数	3-104
<b>3-3-5 PLCopen仕様 MC Part4 動作ファンクションブロック</b>	<b>3-105</b>
MC_GroupHome 関数	3-106
MC_GroupStop 関数	3-107
MC_GroupHalt 関数	3-109
MC_GroupInterrupt 関数	3-111
MC_GroupContinue 関数	3-112
MC_MoveLinearAbsolute 関数	3-113
MC_MoveLinearRelative 関数	3-116
MC_MoveCircularAbsolute 関数	3-119
MC_MoveCircularRelative 関数	3-122
MC_MoveDirectAbsolute 関数	3-125
MC_MoveDirectRelative 関数	3-126
MC_MovePath 関数	3-127
MC_SyncAxisToGroup 関数	3-128
MC_SyncGroupToAxis 関数	3-129
MC_TrackConveyorBelt 関数	3-130
MC_TrackRotaryTable 関数	3-131
<b>3-3-6 PLCopen仕様範囲外 特殊ファンクションブロック</b>	<b>3-132</b>
MC_IntelligentStop 関数	3-133
<b>3-4 モーション制御機能</b>	<b>3-136</b>
3-4-1 動作ファンクションブロックの多重起動	3-136
3-4-2 Part4動作ファンクションブロックの多重起動	3-145

## 第4章 モーション制御パラメータ

4-1 概要	4-1
4-2 PLCopenパラメータ一覧	4-2
4-2-1 共通パラメータ	4-2

4-2-2	軸毎パラメータ	4-3
4-2-3	サーボバックパラメータ	4-7
4-3	ini ファイルによるパラメータ初期値設定方法	4-9
4-3-1	POpenSetting.ini ファイル	4-10
4-3-2	TechnoML3Setting.ini ファイル	4-12
4-3-3	TechnoECTSetting.ini ファイル	4-15
4-4	エラーコード一覧	4-17
4-4-1	通信異常	4-17
4-4-2	機器異常	4-17
4-4-3	コマンド異常	4-17
4-4-4	FB インスタンス異常	4-18
4-4-5	通信コマンド異常	4-18
4-4-6	EtherCAT 通信コマンド異常	4-24

## 第5章 付録

5-1	サンプルプロジェクト	5-1
5-1-1	MULTIPROG 用 PLCopen サンプルプロジェクト	5-1
5-1-2	サンプルプロジェクト使用方法	5-1
5-2	参考文献	5-3

# はじめに

この度は、アルゴシステム製品をお買い上げ頂きありがとうございます。

弊社製品を安全かつ正しく使用していただくために、お使いになる前に本書をお読みいただき、十分に理解していただくようお願い申し上げます。

## 1) お願いと注意

本書では、下記の方法について説明します。

- ・ MULTIPROG への PLCopen 仕様の MC 機能の登録方法
- ・ PLC プログラミング用 PLCopen 仕様 MC ライブラリの使用方法

MULTIPROG や PLC プログラミングについての詳細は省略させていただきます。MULTIPROG および PLC プログラミングに関する資料および文献と併せて本書をお読みください。

## 2) データタイプ

本書で使用するデータタイプとその範囲を表 1 に示します。

表 1. データタイプ

分類	データ型名	サイズ	値の範囲	記述方法
BOOL	BOOL	1bit	TRUE or FALSE	TRUE or FALSE
整数	BYTE	1byte	0~255	2進表現：先頭に「2#」を付けます。 8進表現：先頭に「8#」を付けます。 10進表現：先頭に「10#」を付けます。 16進表現：先頭に「16#」を付けます。 先頭に何も付けなければ、10進数として解釈されます。 (例) 2進表現 2#1111_0000 2#1010_0101 8進表現 8#345 8#567 10進表現 -345 0 +456 10#1234 16進表現 16#AA 16#aaaaaaaa 16#0005
	SINT		-128~128	
	USINT		0~255	
	INT8		-128~128	
	UINT8		0~255	
	WORD	2byte	0~65535	
	INT		-32768~32767	
	UINT		0~65535	
	INT16		-32768~32767	
	UINT16		0~65535	
	DWORD	4byte	0~4294967295	
	DINT		-2147483648~2147483647	
	UDINT		0~4294967295	
	INT32		-2147483648~2147483647	
UINT32	0~4294967295			
浮動小数点	LREAL	8byte	-1.79769313486231e+308 ~ -2.22507385850720e-308、 0、 2.22507385850720e-308 ~ 1.79769313486231e+308、 + ∞ / - ∞	(符号)+整数部+小数点+小数部+(指数部) で記述します。 ( ) 付きは、省略可 (例) -678.0 0.0 0.1234 100000.0 -2.34E-6

# 第 1 章 MULTIPROG 用 PLCopen MC ライブラリ

本章では KW-Software 社製 PLC プログラミング統合開発環境『MULTIPROG』と、MECHATROLINK-III 通信のサーボプロファイルまたは、EtherCAT 通信のモーションコントロールのデバイスプロファイル (CiA402 デバイスプロファイル) を用いた『PLCopen 仕様 MC ファンクションブロック』の、基本的な仕様、構成について説明します。

## 1-1 全体構成

本ライブラリを使用した場合の全体構成図を説明します。

### ◆ EtherCAT を使用した場合

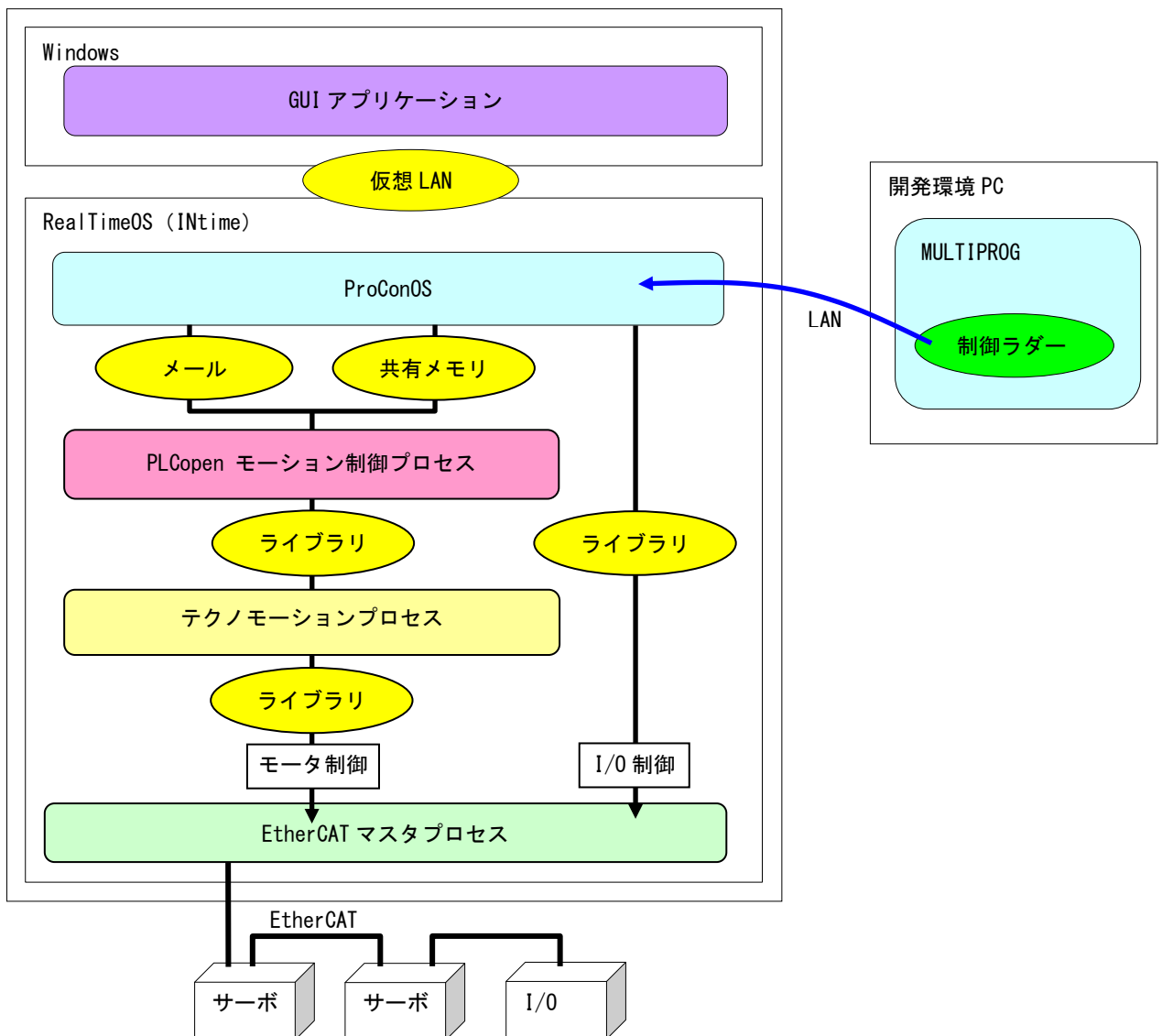


図 1-1-1. 全体構成図 (EtherCAT 版)

MULTIPROG で開発したプログラムを、RealTimeOS 上で動作している ProConOS にダウンロードし、実行します。ProConOS から、PLCopen モーション制御プロセスへ命令が伝達されます。PLCopen モーション制御プロセスはテクノモーションプロセスを通じてサーボパックを制御します。I/O ユニットは ProConOS から直接制御します。EtherCAT 通信は EtherCAT マスタプロセスにて行います。

## ◆ MECHATROLINK-Ⅲ を使用した場合

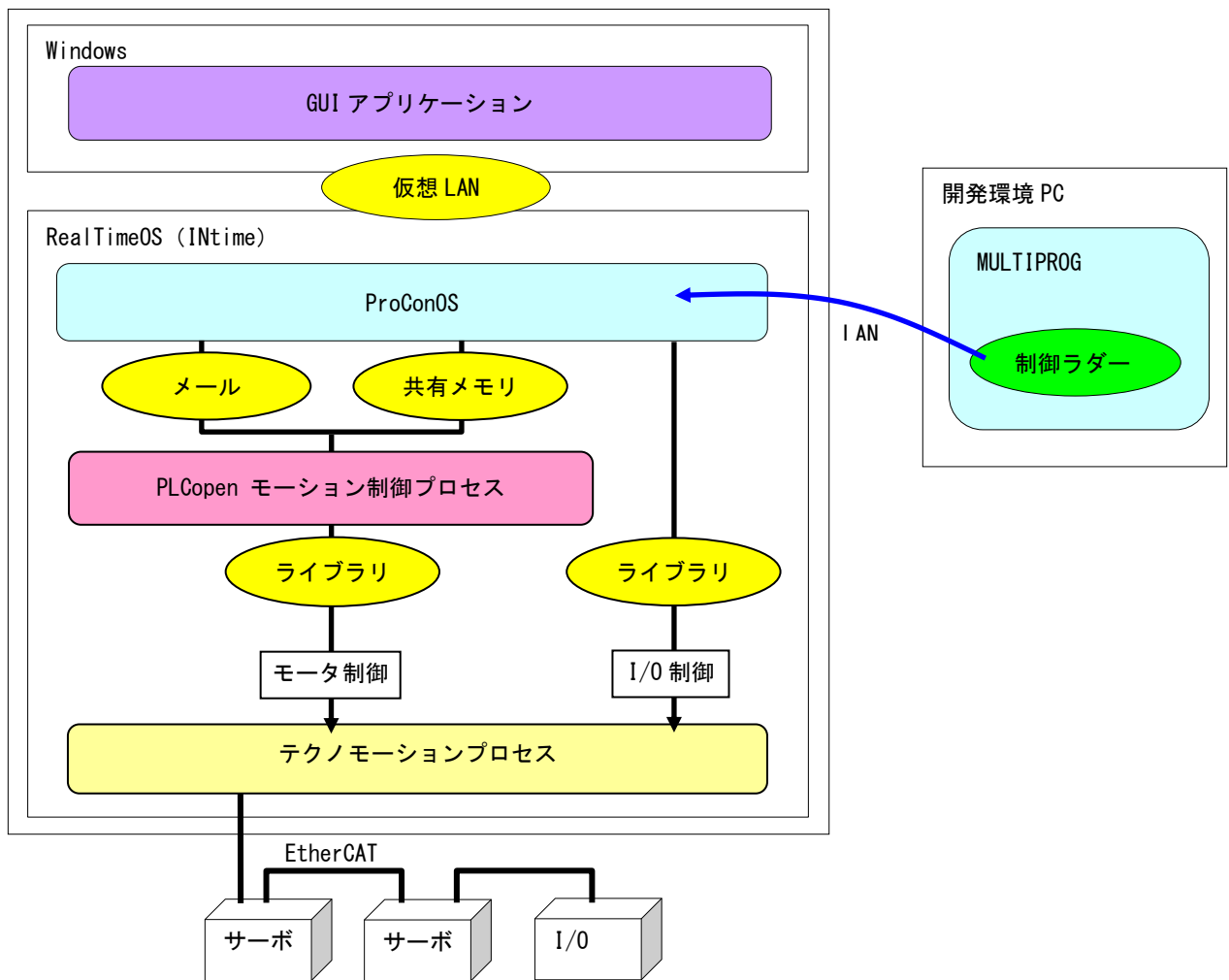


図 1-1-2. 全体構成図 (MECHATROLINK-Ⅲ)

MULTIPROG で開発したプログラムを、RealTimeOS 上で動作している ProConOS にダウンロードし、実行します。ProConOS から、PLCopen モーション制御プロセスへ命令が伝達されます。PLCopen モーション制御プロセスはテクノモーションプロセスを通じてサーボパックを制御します。I/O ユニットは ProConOS から直接制御します。MECHATROLINK-Ⅲ通信はテクノモーションプロセスにて行います。



## 1-2 MULTIPROG とは

MULTIPROG とは、KW-Software 社が開発した、PLC プログラミング統合開発環境です。プログラミング言語とプロジェクト構造は、国際標準規格 IEC61131-3 に適合しています。

MULTIPROG のプログラミングシステムにはインストラクションリスト (IL)、ストラクチャードテキスト (ST)、ラダーダイアグラム (LD)、ファンクションブロックダイアグラム (FBD)、シーケンシャルファンクションチャート (SFC) の 5 言語があります。

**※注：MULTIPROG のタイプにより、使える言語は異なります。**

MULTIPROG の主な機能は下記の通りです。

- ・ プロジェクトの操作
- ・ 制御アプリケーションの作成
- ・ 制御パラメータの生成とコンフィグレーションコードの生成
- ・ ロジックアナライザ、ブレークポイント、アドレスデバッグ、ステップ実行、変数の上書き等のデバッグとコミッシング機能
- ・ プログラムコードとタスクプロパティの修正などのオンライン変更

MULTIPROG についての詳細な操作方法については、MULTIPROG のヘルプ等を参照してください。

## 1-3 PLCopen MC 仕様とは

PLCopen とはプログラマブルコントローラ (PLC) のプログラミングの国際標準規格である IEC 61131-3 の普及促進・標準化推進団体であり、日本の主要メーカを含む世界 PLC 関連企業 46 社を含む 100 社以上が参加するワールド・ワイドな会員組織です。

この団体の規定するモーションコントロール仕様を PLCopen 仕様 MC と呼んでいます。

PLCopen の MC 仕様は動作仕様だけでなく、FB の起動方法や状態まで定義しています。これにより、ハードウェアへの依存性を低減しています。

PLCopen の MC 仕様は表 1-3-1 の 6 つの Part から構成されています。

表 1-3-1. PLCopenMC 技術仕様の種類と状況

	仕様	状況	内容
Part1	基本仕様	◎	単軸、多軸制御および、管理用の命令を準備
Part2	拡張仕様	◎	Part1 に対して拡張機能を追加 (Ver2.0 にて Part1 と Part2 は統合化される)
Part3	ユーザガイドライン	○	ユーザのモーションをガイドラインとして準備
Part4	補間制御仕様	○	補間制御および、管理用の命令を準備
Part5	原点復帰追加仕様	○	原点復帰に関して、Part1 への追加仕様を準備
Part6	流体パワー拡張仕様	△	流体パワーデバイスやシステムとのプログラミングの統合化のために規定

◎：英語、日本語仕様公開済み ○：英語仕様公開済み △：仕様策定中

PLCopen の MC 仕様 Part1、Part2、Part5 では状態遷移が規定されており、この状態毎に実行可能なファンクションブロックが決まっています。この状態遷移図を図 1-3-1 に示します。

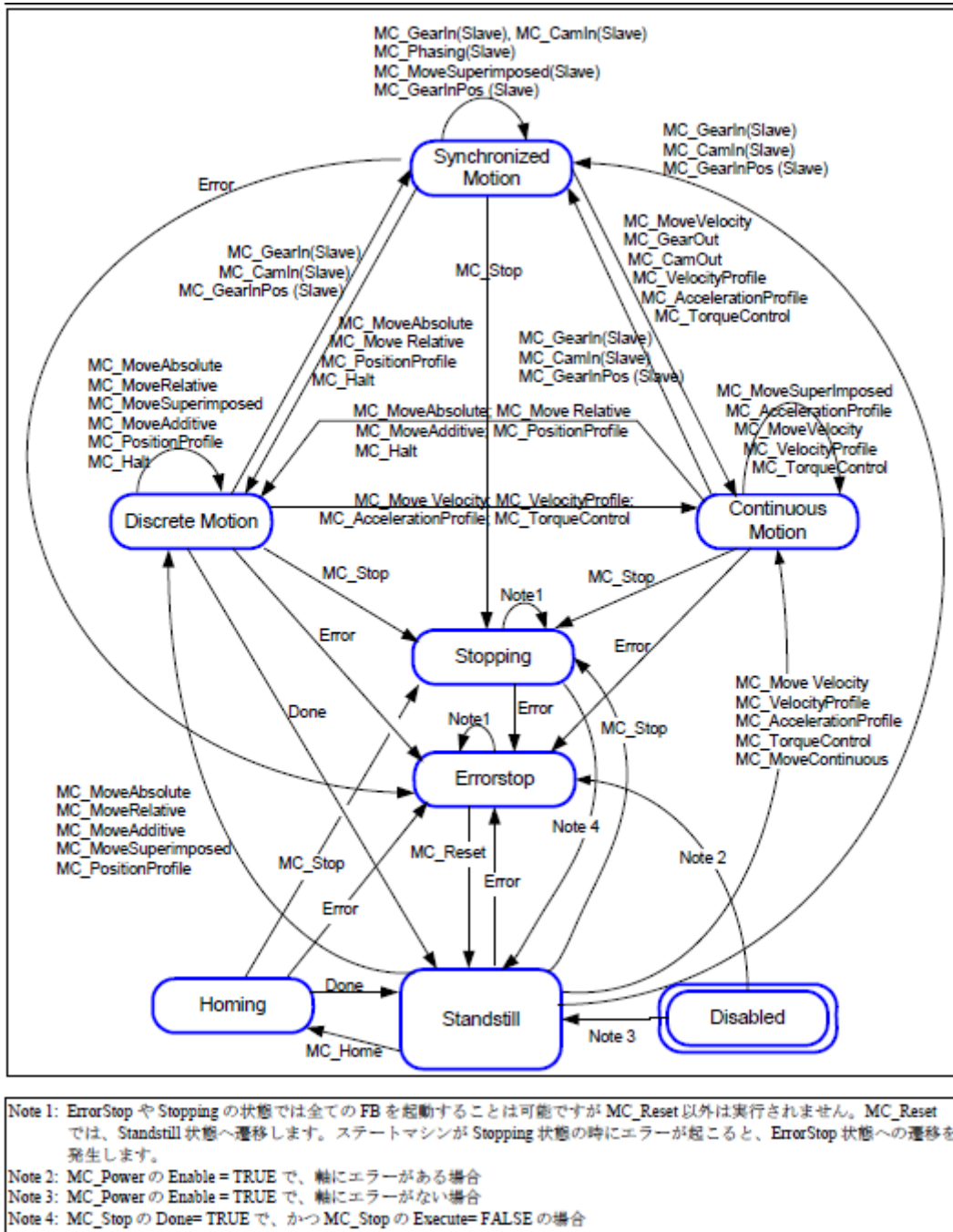


図 1-3-1. PLCopenMC Part1、Part2、Part5 状態遷移図

図 1-3-1 に記載されていない以下のファンクションブロックについては、軸の状態に影響しないため、状態を変化させずに実行する事が出来ます。

MC\_ReadStatus; MC\_ReadAxisError; MC\_ReadParameter; MC\_ReadBoolParameter; MC\_ReadByteParameter;  
 MC\_ReadWordParameter; MC\_ReadDwordParameter; MC\_WriteParameter; MC\_WriteBoolParameter;  
 MC\_WriteByteParameter; MC\_WriteWordParameter; MC\_WriteDwordParameter; MC\_ReadActualPosition;  
 MC\_ReadActualVelocity; MC\_ReadActualTorque; MC\_CamTableSelect.

その他の詳細については、「技術仕様書 PLCopen - Technical Committee 2- 専門委員会 モーションコントロール用ファンクションブロック パート 1-Basics Version 1.1」と「技術仕様書 PLCopen - Technical

Committee 2- 専門委員会 モーションコントロール用ファンクションブロック パート2-Extensions Version 1.0]、「Technical Paper PLCopen Technical Committee 2 – Task Force Function Blocks for motion control: Part 5 – Homing Version 0.99」を参照してください。

同期制御（直線補間や円弧補間）を行うためには、Part4 で規定されている仕様を満足する必要があります。Part4 の状態遷移図を図 1-3-2 に示します。

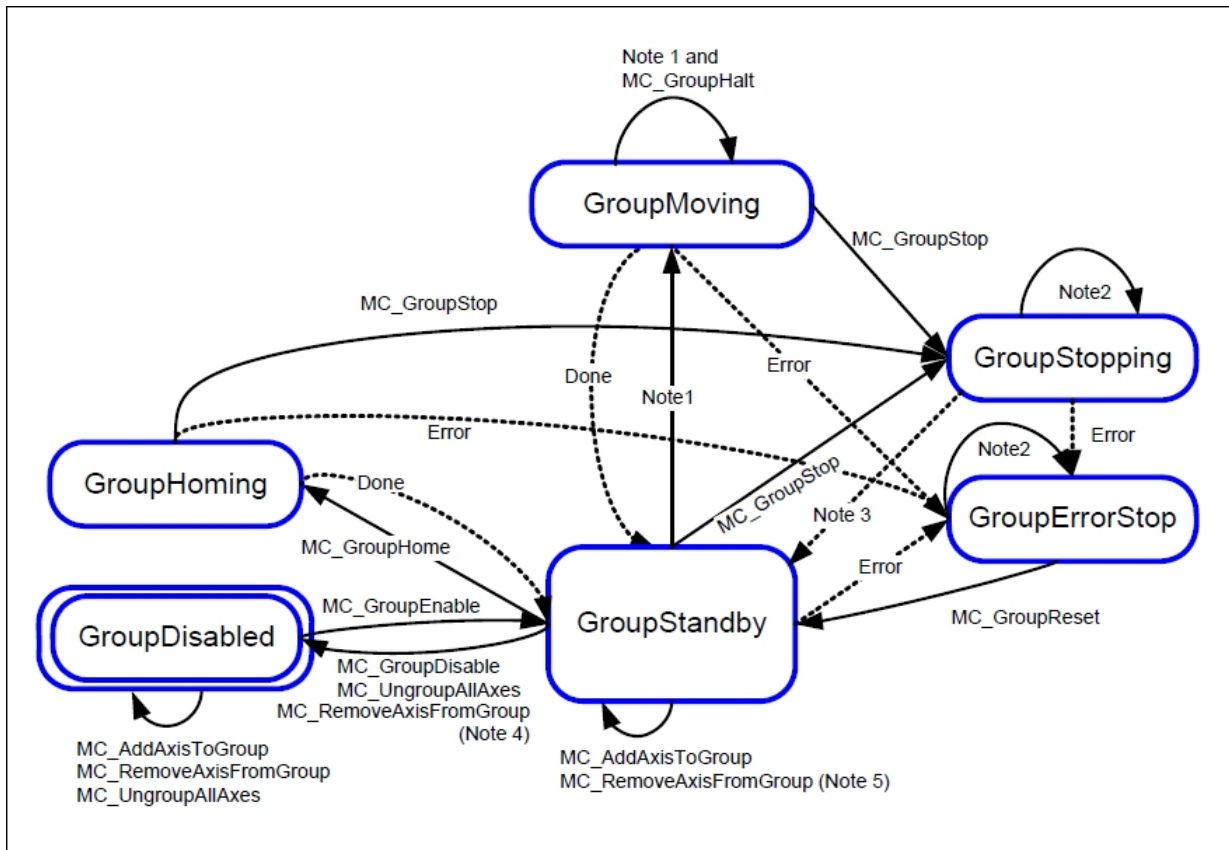


図 1-3-2. PLCopenMC Part4 状態遷移図

Note1: 全ての動作系ファンクションブロックに適用されます。

Note2: MC\_GroupReset によるエラー解除成功以外、全てのファンクションブロックに適用されます。

Note3: MC\_GroupStop.DONE=TRUE かつ MC\_GroupStop.EXECUTE=FALSE の時に遷移します。

Note4: 最後の軸が Remove された際に遷移します。

Note5: グループが空ではない時に遷移します。

また、単軸の状態遷移図と Part4 状態との相関図を図 1-3-3 に示します。

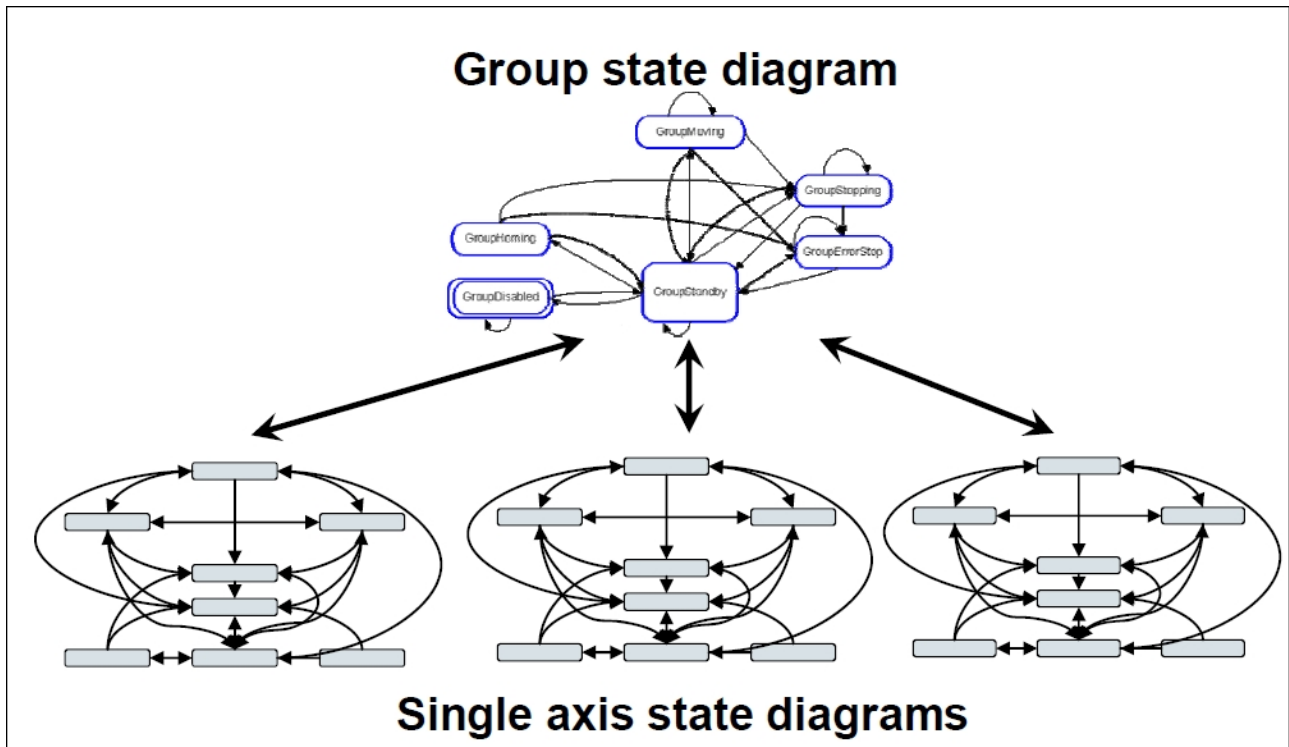


図 1-3-3. PLCopenMC Part4 単軸状態との相関図

動作系ファンクションブロックについて、軸グループに登録済み (MC\_AddAxisToGroup 実行済み) の単軸に対して動作指令を実行すると単軸の動作指令はエラーとなります。(この時、軸グループの動作は停止せず、単軸エラーもエラー状態に遷移しません。)

その他の詳細については、「Technical Paper PLCopen Technical Committee 2 – Task Force Function Blocks for motion control:Part 4 –Coordinated Motion Version 1.0」を参照してください。

## 1-4 MECHATROLINK-III とは

### 1-4-1 概要

MECHATROLINK-III通信とは、MECHATROLINK 協会の提唱するオープンな高速フィールドネットワークです。1台のコントローラで、複数のユニットを分散制御することが可能です。

MECHATROLINK-IIIの特徴は下記の通りです。

- ・ サイクリック伝送による同期通信
- ・ 100Mbps での高速伝送
- ・ 伝送周期は接続局数、伝送データ量で最適値を選択可能（伝送周期 31.25us~64ms）
- ・ 接続方法をカスケード形/スター形/Point to Point 形と装置に合わせた形で自由に構成可能
- ・ MECHATROLINK 協会製「伝送 LSI」が、誤り検出と伝送周期内再送制御を含む伝送制御を行うため、FA コントローラの負荷低減が可能
- ・ マスタとなるコントローラの他にサポートツールを接続可能

MECHATROLINK-IIIの接続形態は、C1 マスタ局が 1 局、スレーブ局が最大 62 局の Ethernet 接続によるネットワークシステムです。必要に応じて C2 マスタ局を 1 局接続できます。

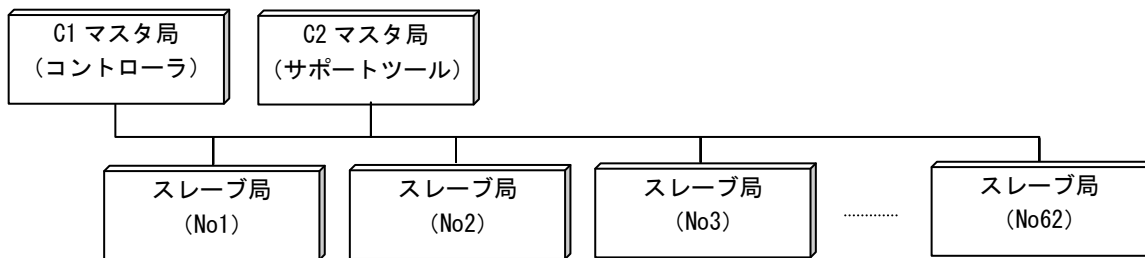


図 1-4-1-1. MECHATROLINK-III 接続図

モーション同期制御を実現するために、(株)テクノが開発したモーションライブラリを組み込んでいます。同期制御を組み込んでいないものと一部仕様が異なる部分があります。

## 1-5 EtherCAT 通信とは

### 1-5-1 概要

EtherCAT (Ethernet Control Automation Technology) は、Beckhoff 社により開発され、現在では EtherCAT Technology Group (ETG) により管理されています。

EtherCAT 接続は、新しいリアルタイムイーサネットを用いたネットワーク通信で、ツイストペア、または光ファイバケーブルで接続ができるとともに、ライン、ツリー、デジチェーン、ドロップラインをサポートします。

EtherCAT 転送方法はマスタから送信されたフレームがスレーブ通過時に出カデータを取り出し、入力データを挿入します。Ethernet プロトコルは、IEEE802.3 に準拠した標準のイーサネットプロトコルが維持されていますので、新たにサブバスの構築は必要ありません。

EtherCAT プロトコルはプロセスデータ向けに最適化されています。EtherType により Ethernet フレーム内で直接転送されます。いくつかのサブ・テレグラムを構成しているかもしれませんが、それぞれ 4GB 容量までのロジック・プロセス・イメージを特定のメモリ・エリアに提供します。

EtherCAT は Ethernet をベースとしたネットワークの基本的な通信構造が定義されている IEC61158 の Section12 に定義されており、EtherCAT 通信プロファイルの EtherCAT ステートマシーン (ESM)、フィールドメモリ管理ユニット (FMMU) によるプロセスデータ通信方式、MailBox による CoE サービスチャンネル、シンクマネージャ (SM)、同期クロック方式による同期構造が説明されています。

ドライブおよびモーションコントロールのデバイスプロファイル (CiA402 デバイスプロファイル) は、サーボドライブ、正弦波インバータ、およびステッピングモーター用コントローラの機能動作を定義します。このプロファイルでは、複数の動作モードと対応する設定パラメータも規定されます。

この仕様には、状態ごとの内部および外部動作を規定する有限状態オートマトン (Finite State Automaton: FSA) も含まれます。受領されるコマンドや高出力を適用するかどうかは、ドライブの状態によって決まります。

状態はホストコントローラから受け取るコントロールワードで変更されます。また、内部イベントによって変更することもできます。現在の状態はステータスワードで示されます。コントロールワードと各種コマンド値 (速度など) はデフォルトの RxPDO (レシーブ PDO) にマッピングされます。ステータスワードと各種実査値 (位置など) は TxPDO (トランスミット PDO) にマッピングされます。この規格には、すべてのドライブで利用できる汎用のデフォルト PDO と特定のドライブ (サーボドライブ、正弦波インバータ、ステッピングモーターなど) でのみ使用できるデフォルト PDO が用意されています。

オプション機能やパラメータが多いため、CiA 402 に準拠するデバイスは交換できない場合があります。

CiA 402 デバイスプロファイルは IEC 61800-7-201 および IEC 61800-7-301 (いずれも IEC から入手可能) で国際標準として定められています。

モーション同期制御を実現するために、(株) テクノが開発したモーションライブラリを組み込んでいます。同期制御を組み込んでいないものと一部仕様が異なる部分があります。

## 1-5-2 導入手順

EtherCAT 版サーボパックを使用して、PLCopen による同期制御を実現するためには、下記のような手順を踏む必要があります。

- 1) INtime カーネル設定  
↓
- 2) EtherCAT マスタスタック設定  
↓
- 3) ENI ファイル作成  
↓
- 4) アプリケーションの作成

### 1) INtime カーネル設定

EtherCAT を使用するために、INtime カーネルを設定する必要があります。設定方法については、『同期モーション導入マニュアル』(77IT10020\*) を参照してください。

### 2) EtherCAT マスタスタック設定

EtherCAT 通信は、図 1-1-1 に書かれている EtherCAT マスタプロセスによって通信が開始されます。このとき、EtherCAT で使用する NIC の選択と通信周期等を、設定ファイル「ACMst.ini」で設定します。この設定ファイルの詳細については、『設定マニュアル ACMst.ini』(76DLH049\*) を参照してください。

### 3) ENI ファイル作成

EtherCAT の通信は ESI (EtherCAT スレーブ情報) ファイルをもとにして、各スレーブがどのように接続されているかマッピングするための、ENI (EtherCAT ネットワーク情報) ファイルを生成する必要があります。

EtherCAT マスタは、生成された ENI ファイルを元にして通信を開始します。そのため、ENI ファイルに記述されているスレーブ構成と実際に接続されているスレーブ構成が異なる場合、正常に通信することができません。

ENI ファイルを作成するためには BECKHOFF 社製「ET9000 | EtherCAT Configurator」を使用します。

ENI ファイルの作成方法を下記に記します。

#### <用意するもの>

- A) 「ET9000」 | 「EtherCAT Configurator」(Beckhoff Automation より入手) がインストールされた Windows PC
- B) 接続したい EtherCAT スレーブ群
- C) 接続する各スレーブの ESI ファイル  
ESI ファイルは各 EtherCAT スレーブベンダーから提供されています。
- D) PC と各 EtherCAT を接続する CAT5 以上の Ethernet ケーブル
- E) 生成される ENI ファイルを保存する USB メモリ等の記録媒体

## ① Ethernet ケーブルの接続

A)、B)、D)を図 1-5-2-1 のように EtherCAT スレーブをデジチェーン接続します。

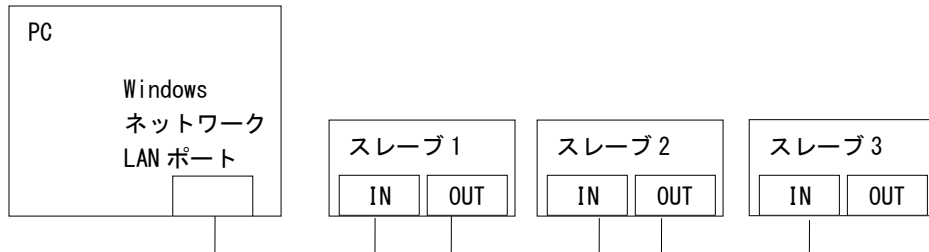


図 1-5-2-1. EtherCAT 接続図

## ② ESI ファイルの適用

各スレーブの ESI ファイルを、EtherCAT Configurator インストールフォルダの「EtherCAT」フォルダにコピーしてください。

【例】デフォルトインストールの場合

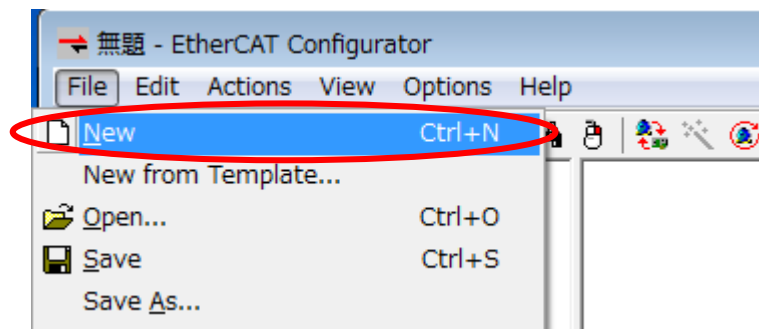
“ C:\Program Files\EtherCAT Configuration\EtherCAT ”

## ③ EtherCAT Configurator の起動

「スタートメニュー」→「すべてのプログラム」→「EtherCAT Configurator」→「EtherCAT Configurator」をクリック

## ④ 新規プロジェクト作成

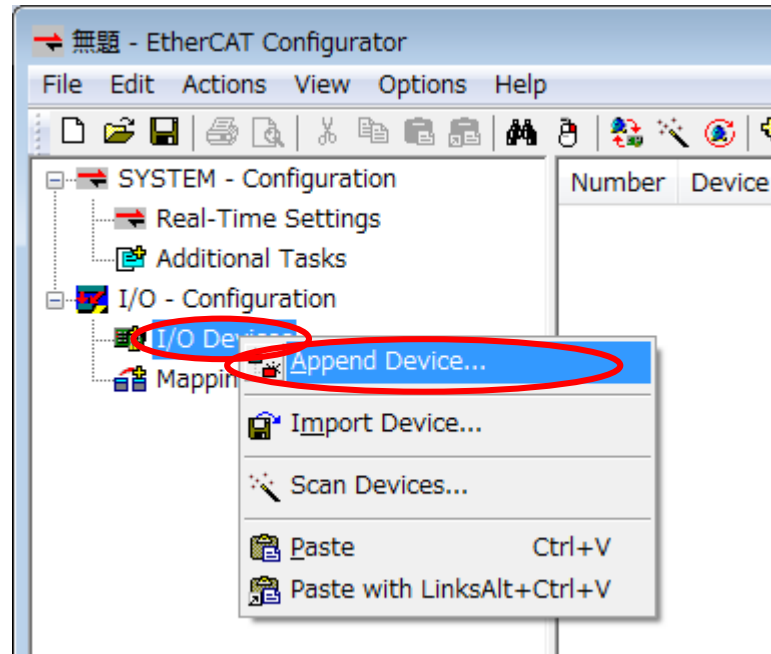
メニューバーより、[File]→[New] をクリックし、新規プロジェクトを作成します。



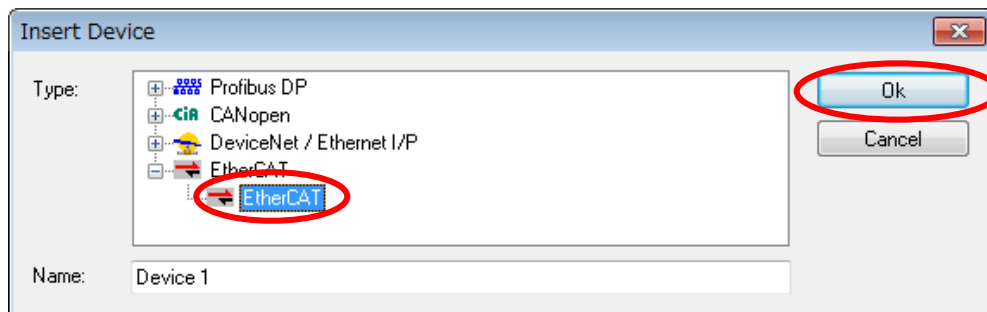


## ⑤ EtherCAT Device の追加

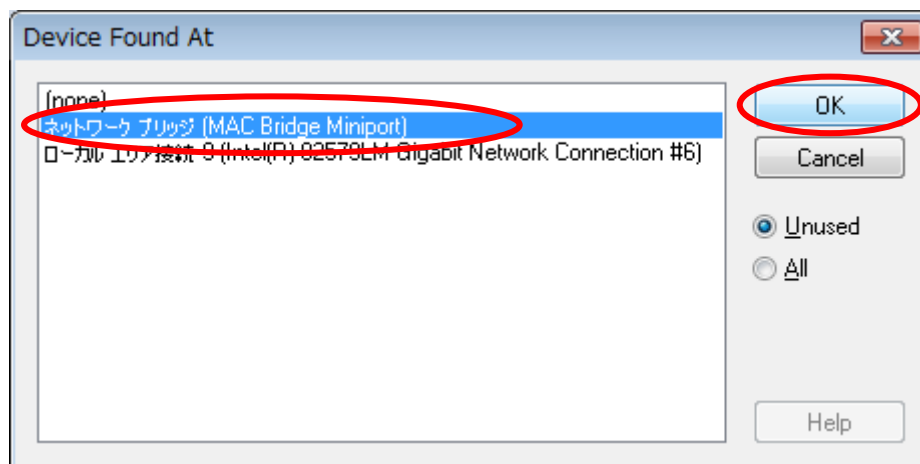
左ウィンドウから [I/O Devices] を右クリックし、ポップアップメニューの [Append Device...] をクリックします。



[Insert Device] 画面が開きますので、[Type:] のツリーから [EtherCAT] → [EtherCAT] を選択し、[OK] を押してください。



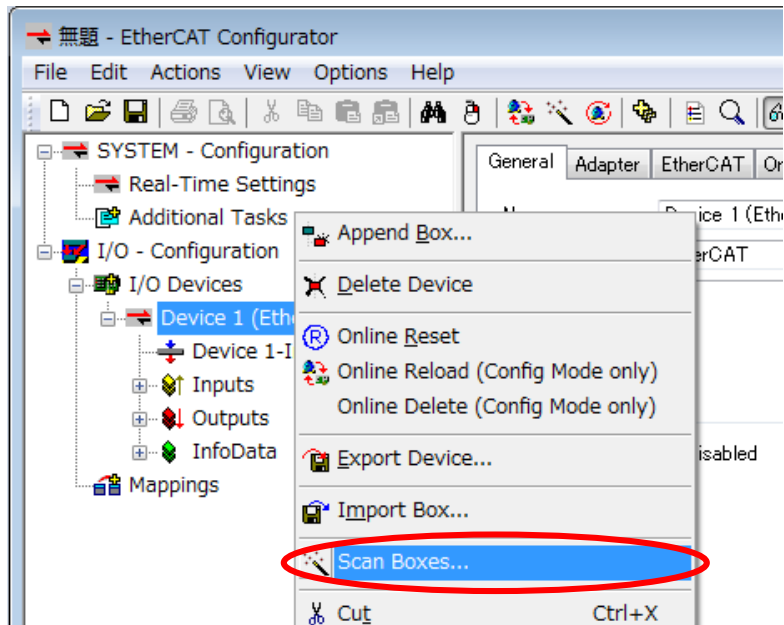
続いて [Device Found At] 画面から、接続した LAN ポートを選択し [OK] を押します。



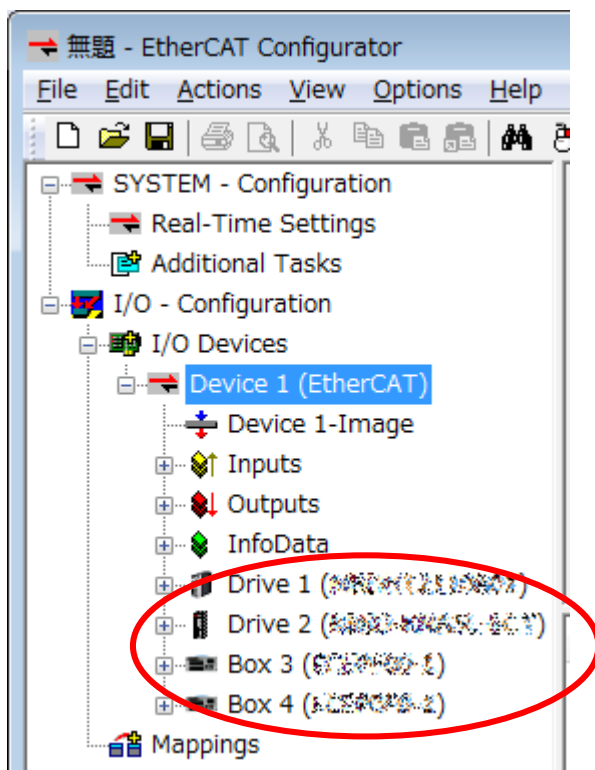
EtherCAT Device が追加されます。

## ⑥ スレーブのスキャン実行

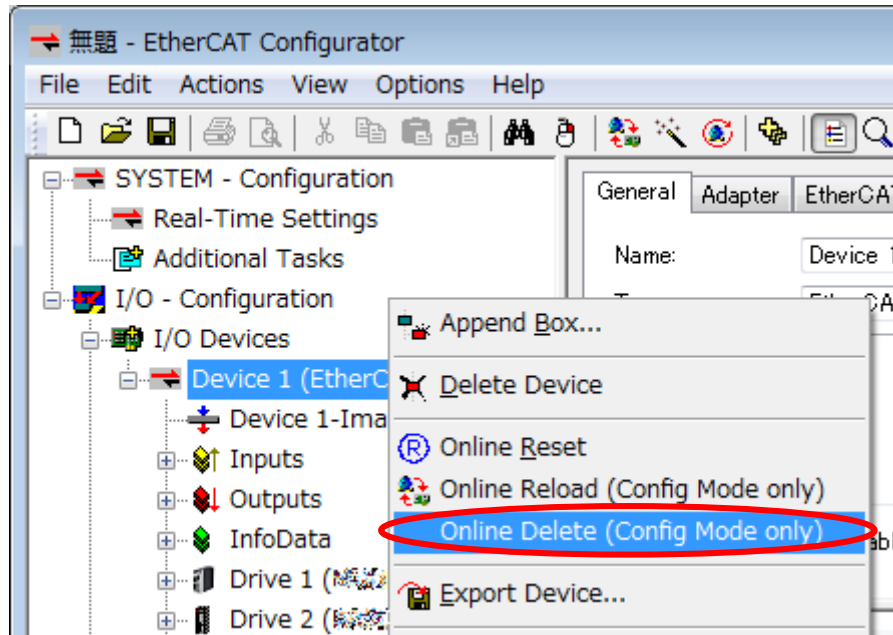
左ウィンドウから [Device 1] を右クリックし、ポップアップメニューの [Scan Boxes...] をクリックし、EtherCAT 通信を開始させスレーブの検出を行います。



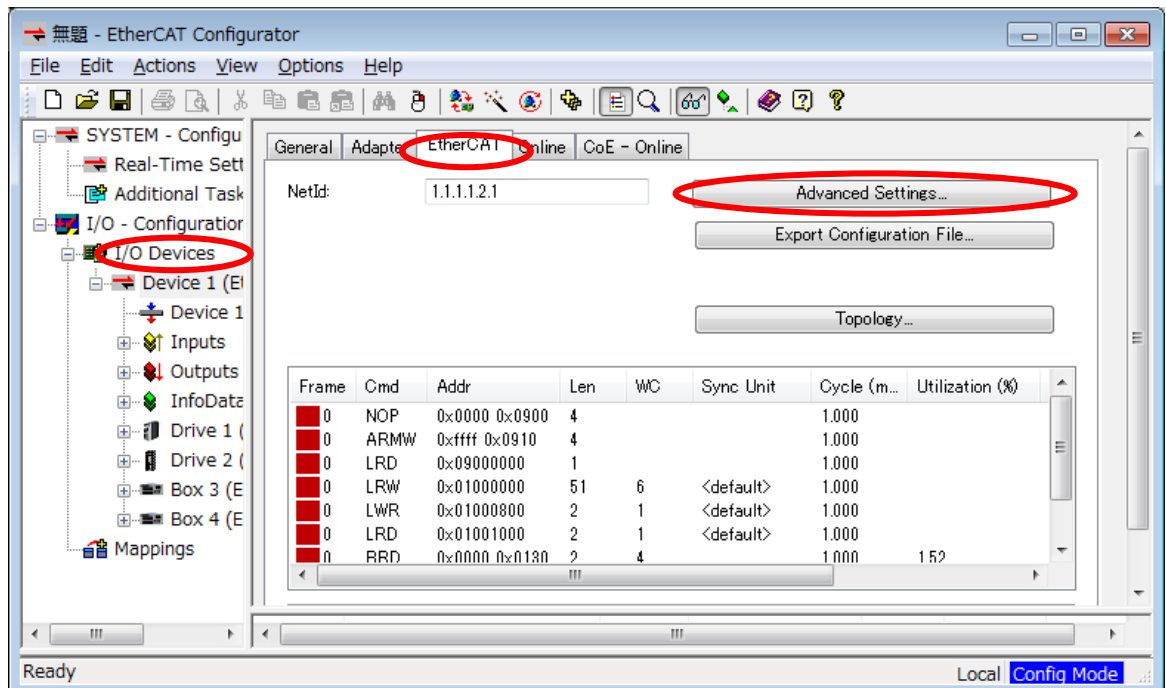
スレーブの検出が成功すると、以下のように [Device 1] の配下にスレーブが追加されます。



- ⑦ Online Delete 実行  
「Device1」を右クリックし、「Online Delete」をクリックして EtherCAT 通信を停止させます。

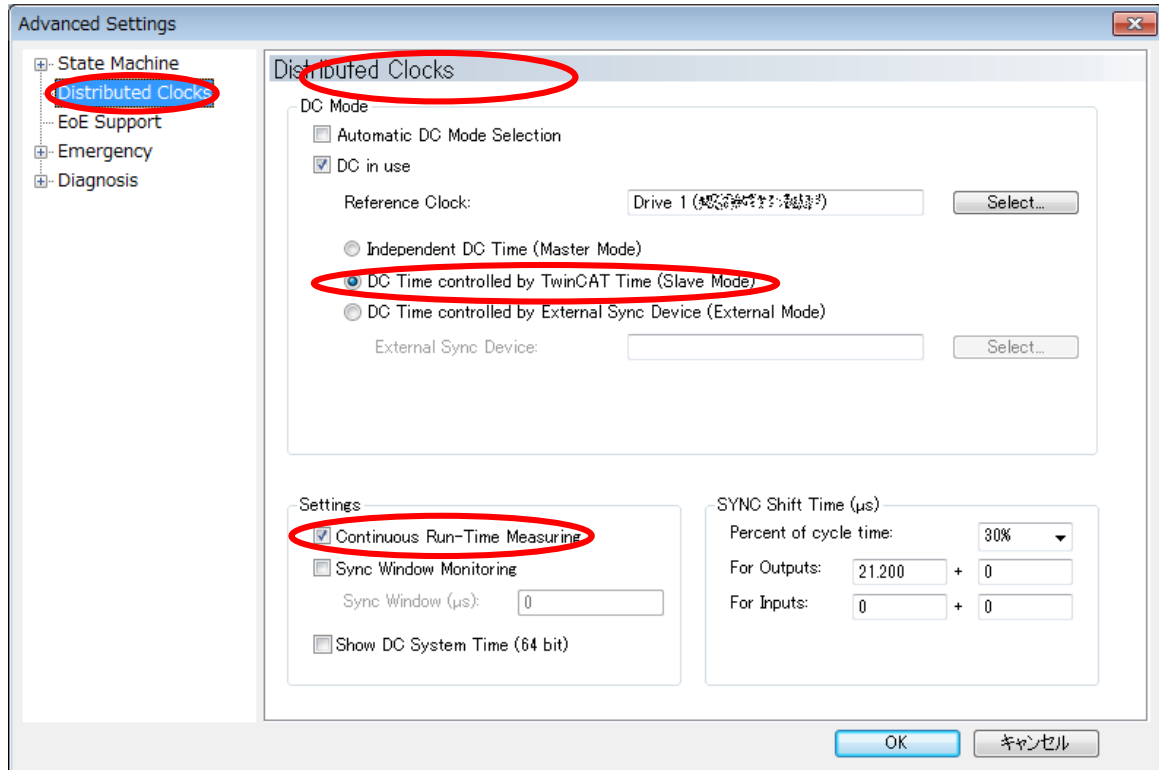


- ⑧ MasterDC (Distributed Clocks) 設定  
左ウィンドウから [Device 1] を選択後、右ウィンドウにある [EtherCAT] タブを選択します。  
[Advanced Settings...] を押してください。[Advanced Settings] 画面が開きます。



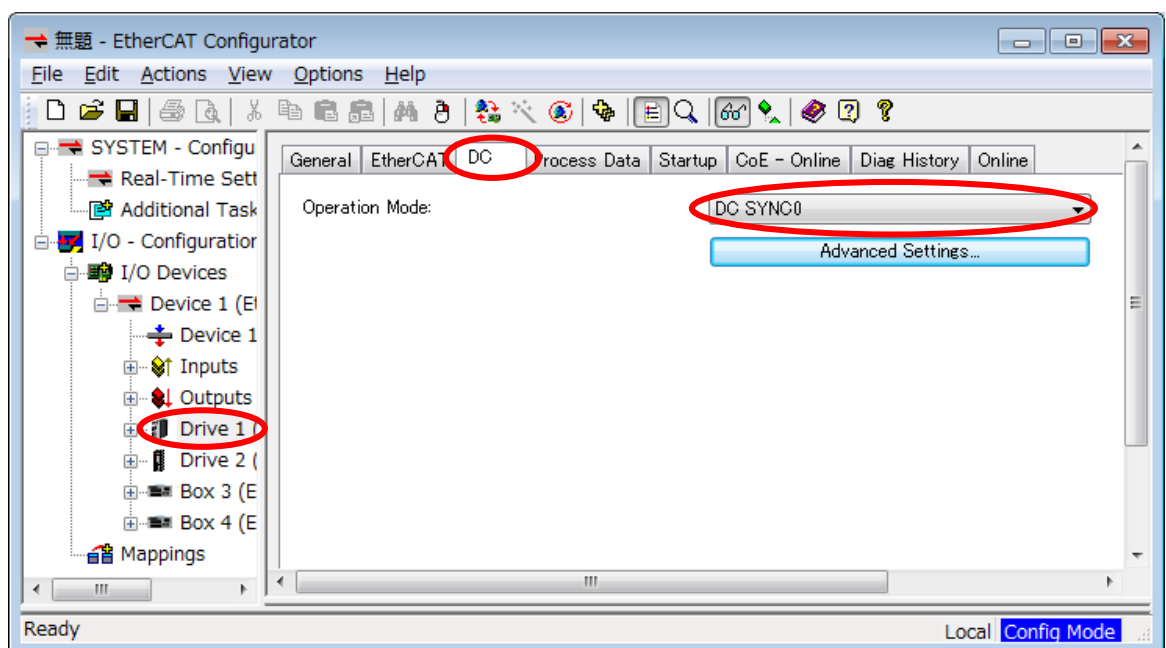
[Advanced Settings] では、左ウィンドウから [Distributed Clocks] を選択後、右ウィンドウの項目を設定します。

[Automatic DC Mode Selection]のチェックを外してください。  
 [DC Time controlled by TwinCAT Time (Slave Mode)]を選択してください。  
 [Settings]の[Continuous Run-Time Measuring]にチェックを入れてください。



### ⑨ SlaveDC(Distributed Clocks)設定

左ウィンドウから設定する Slave を選択後、右ウィンドウにある [DC] タブを選択します。  
 [Operation mode]のプルダウンから DC を選択してください。  
 ([Operation mode]選択肢は Slave により異なります。“DC”、“DC0”、“DC SYNC0”等、Distributed Clocks を意味する選択肢を選んでください。)

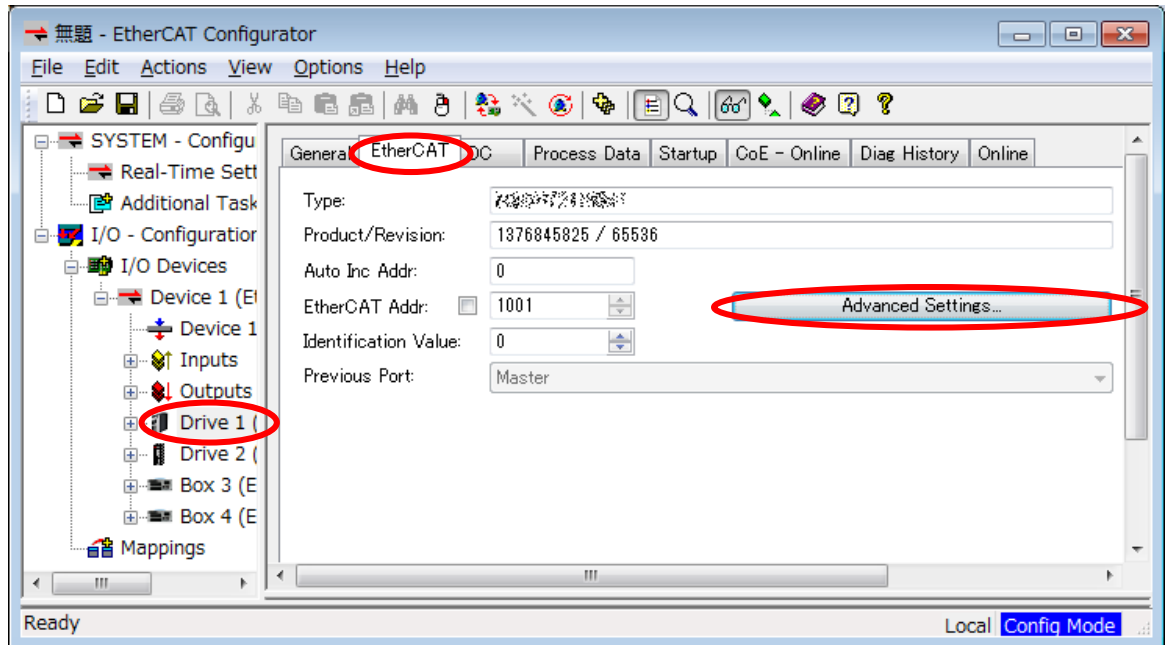


※注：DC に対応していないスレーブは設定の必要ありません。

⑩ Alias の設定

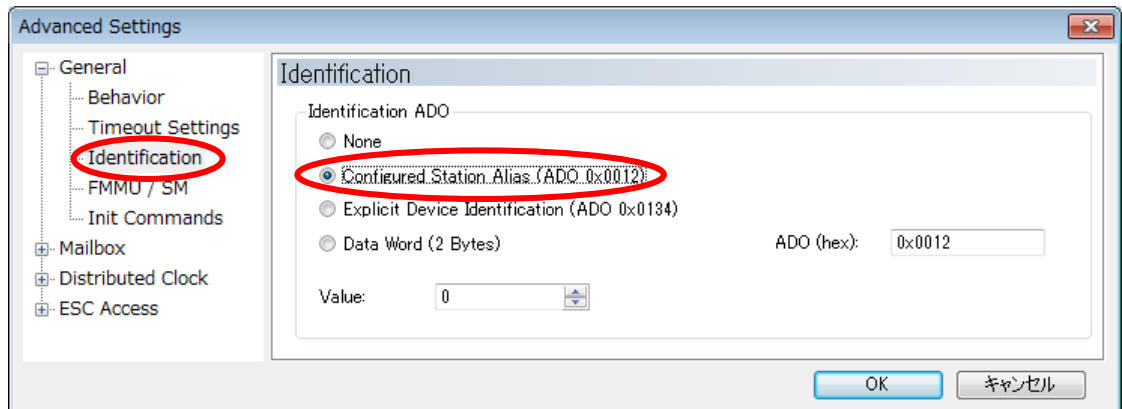
左ウィンドウから設定する Slave を選択後、右ウィンドウにある [EtherCAT] タブを選択します。

[Advanced Settings...] を押してください。[Advanced Settings] 画面が開きます。



[Advanced Settings] では、左ウィンドウから [Identification] を選択後、右ウィンドウの項目を設定します。

[Identification ADO] の [Configured Station Alias (ADO 0x0012)] を選択してください。



## ⑪ PDO マッピングの確認

同期モーションを実現するためには下記のオブジェクトが PDO にマッピングされている必要があります。

## ● 必須項目 (サイクリック位置モードに必須のオブジェクト)

## 【Receive PDO】

- ・ STATUS\_WORD (6041h, 00h)
- ・ POSITION\_ACTUAL\_VALUE (6064h, 00h)

## 【Transmit PDO】

- ・ CONTROL\_WORD (6040h, 00h)
- ・ TARGET\_POSITION (607Ah, 00h)

## ● トルク指令

## 【Receive PDO】

- ・ MODES\_OF\_OPERATION\_DISPLAY (6061h, 00h)
- ・ TORQUE\_ACTUAL\_VALUE (6077h, 00h)

## 【Transmit PDO】

- ・ MODES\_OF\_OPERATION (6060h, 00h)
- ・ TARGET\_TORQUE (6071h, 00h)

## ● ラッチ位置決め - C 相及び Probe1 を ON エッジでラッチする場合

## 【Receive PDO】

- ・ TOUCH\_PROBE\_STATUS (60B9h, 00h)
- ・ TOUCH\_PROBE\_POS1\_POS\_VALUE (60BAh, 00h)

## 【Transmit PDO】

- ・ TOUCH\_PROBE\_FUNCTION (60B8h, 00h)

## ● ラッチ位置決め - C 相及び Probe1 を OFF エッジでラッチする場合

## 【Receive PDO】

- ・ TOUCH\_PROBE\_STATUS (60B9h, 00h)
- ・ TOUCH\_PROBE\_POS1\_NEG\_VALUE (60BBh, 00h)

## 【Transmit PDO】

- ・ TOUCH\_PROBE\_FUNCTION (60B8h, 00h)

## ● ラッチ位置決め - Probe2 を ON エッジでラッチする場合

## 【Receive PDO】

- ・ TOUCH\_PROBE\_STATUS (60B9h, 00h)
- ・ TOUCH\_PROBE\_POS2\_POS\_VALUE (60BCh, 00h)

## 【Transmit PDO】

- ・ TOUCH\_PROBE\_FUNCTION (60B8h, 00h)

## ● ラッチ位置決め - Probe2 を OFF エッジでラッチする場合

## 【Receive PDO】

- ・ TOUCH\_PROBE\_STATUS (60B9h, 00h)
- ・ TOUCH\_PROBE\_POS2\_NEG\_VALUE (60BDh, 00h)

## 【Transmit PDO】

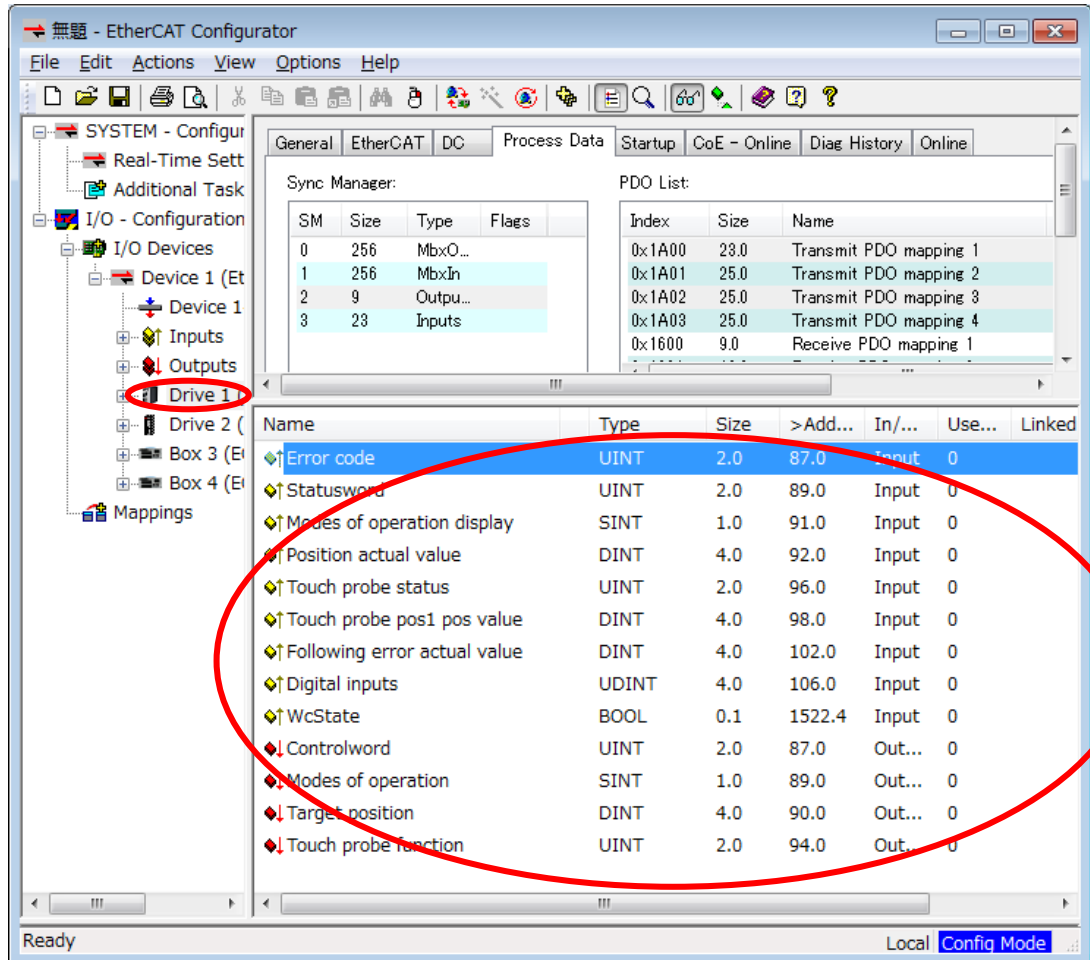
- ・ TOUCH\_PROBE\_FUNCTION (60B8h, 00h)

## ● ラッチ位置決め - 原点信号, +0T, -0T でラッチする場合

## 【Receive PDO】

- ・ DIGITAL\_INPUTS (60FDh, 00h)

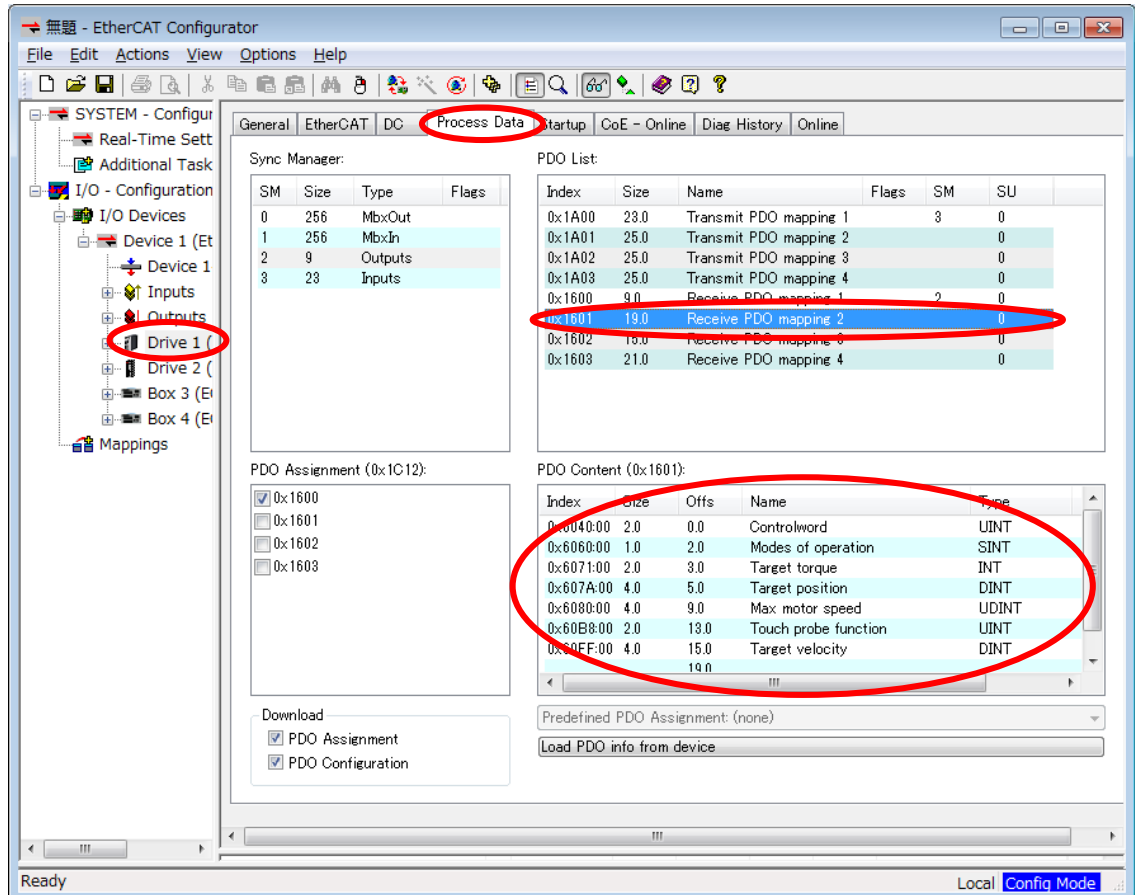
左ウィンドウから設定する Slave を選択すると、右下のウィンドウにマップされているオブジェクトが表示されます。  
ここに必要なオブジェクトが無い場合は「PDO マッピングの選択方法」の操作方法で PDO マッピングを変更してください。ここに必要なオブジェクトがそろっている場合は、「ENI ファイルの出力」に進んでください。



## ⑫ PDO マッピングの選択方法

必要なオブジェクトを含む PDO マッピングがあるかどうかは各スレーブのマニュアルで確認できますが、以下のように『EtherCAT Configurator』の画面で確認することもできます。まず、左ウィンドウから設定する Slave を選択後、右ウィンドウにある [Process Data] タブを選択します。

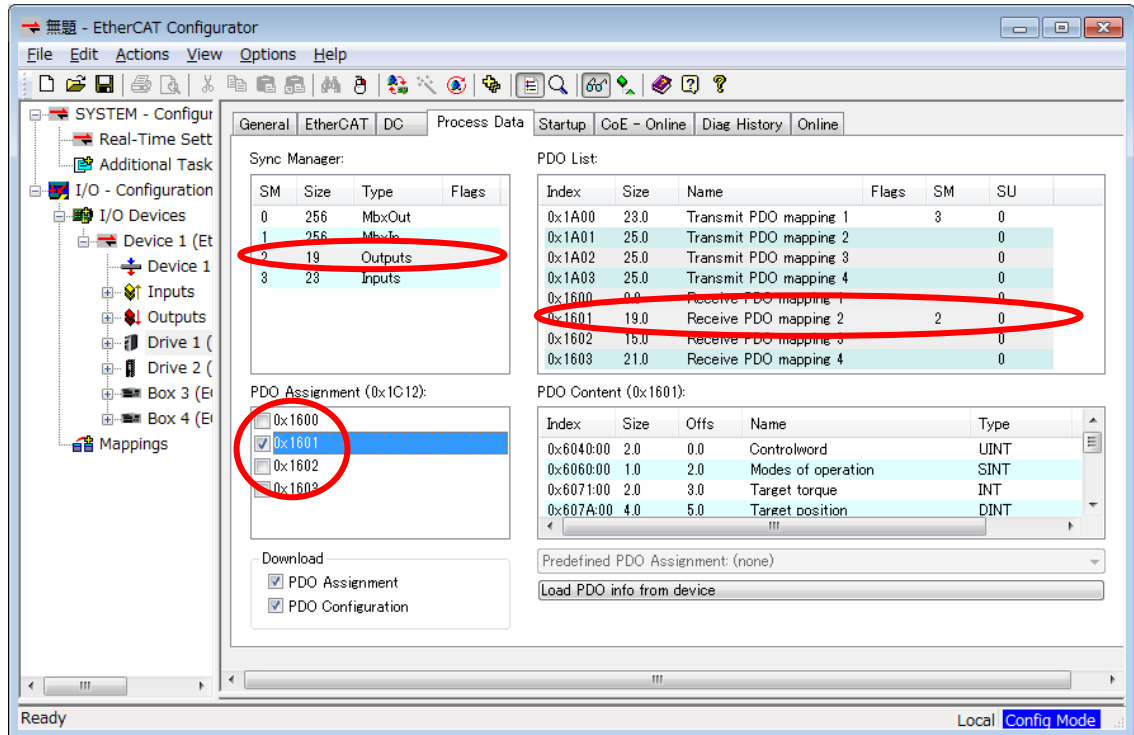
[PDO List] から PDO マッピングオブジェクトをクリックし選択すると、下の [PDO Content] にマップされるオブジェクトのリストが表示されます。



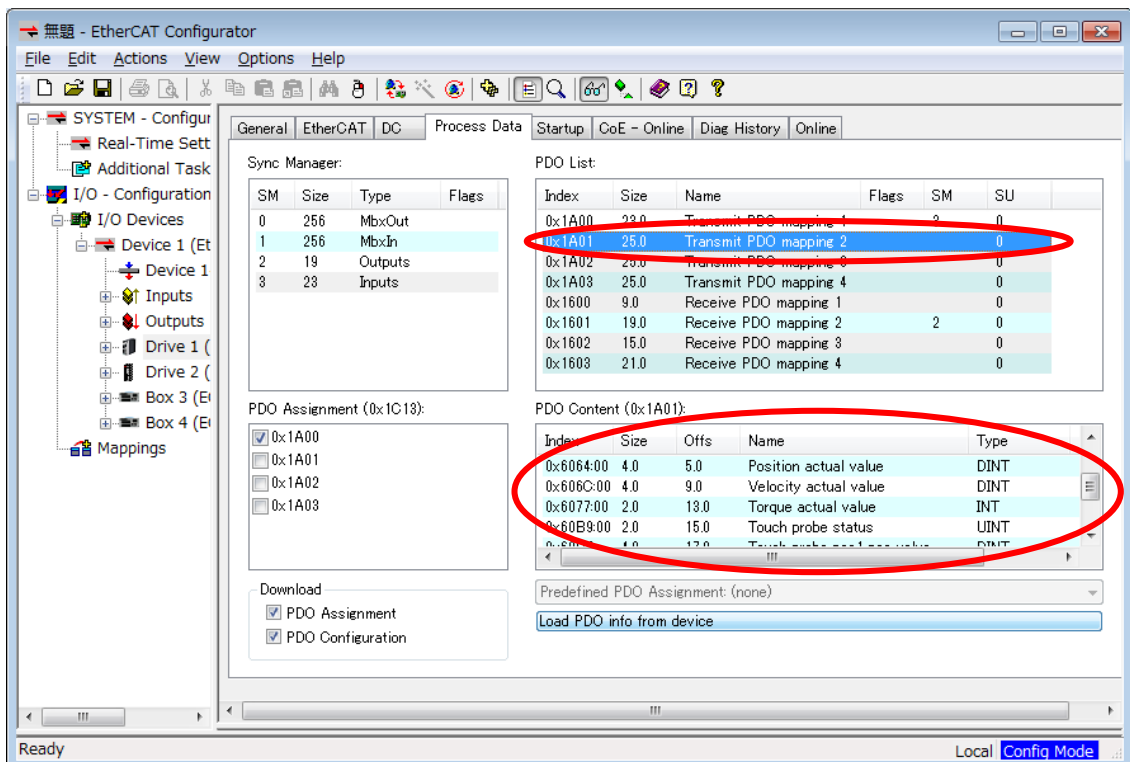
例えば、上記画面では Receive PDO mapping の [0x1601] に必要なオブジェクトがそろっているようなので、[Sync Managers] から [SM] 列が「2」の行 (Type:Outputs) を選択します。



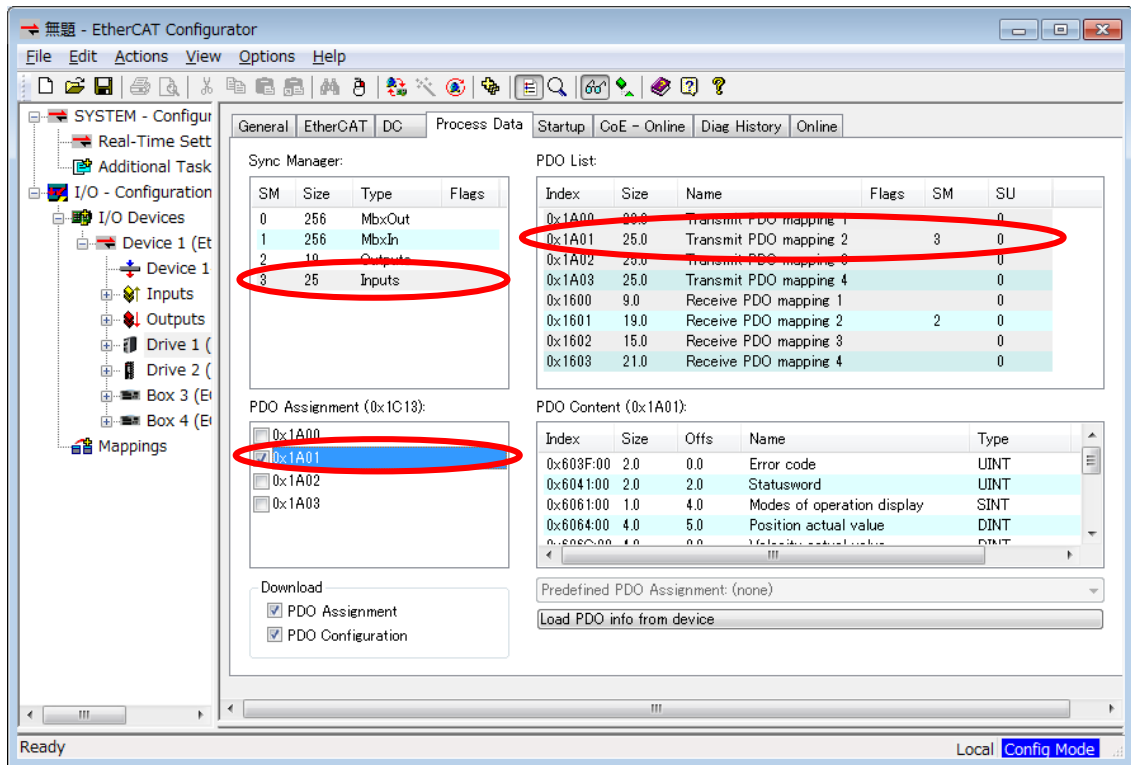
下の [PDO assignment] のデフォルト設定の [0x1600] からチェックを外し、[0x1601] にチェックをします。  
 チェックをすると [PDOs configuration] の [0x1601] の [SM] 列に「2」が表示されます。



続いて、Transmit PDO mapping についても、[PDOs configuration] に登録されています。Receive PDO mapping と同様を選択すると、下の [PDO Content] にマップされるオブジェクトのリストが表示されます。



上記画面では Transmit PDO mapping の [0x1A01]に必要なオブジェクトがそろっているようなので、[Sync Managers]から [SM]列が「3」の行 (Type:Inputs) を選択します。下の [PDO assignment]のデフォルト設定の [0x1A00]からチェックを外し、[0x1A01]にチェックをします。  
 チェックをすると [PDOs configuration]の [0x1A01]の [SM]列に「3」が表示されます。



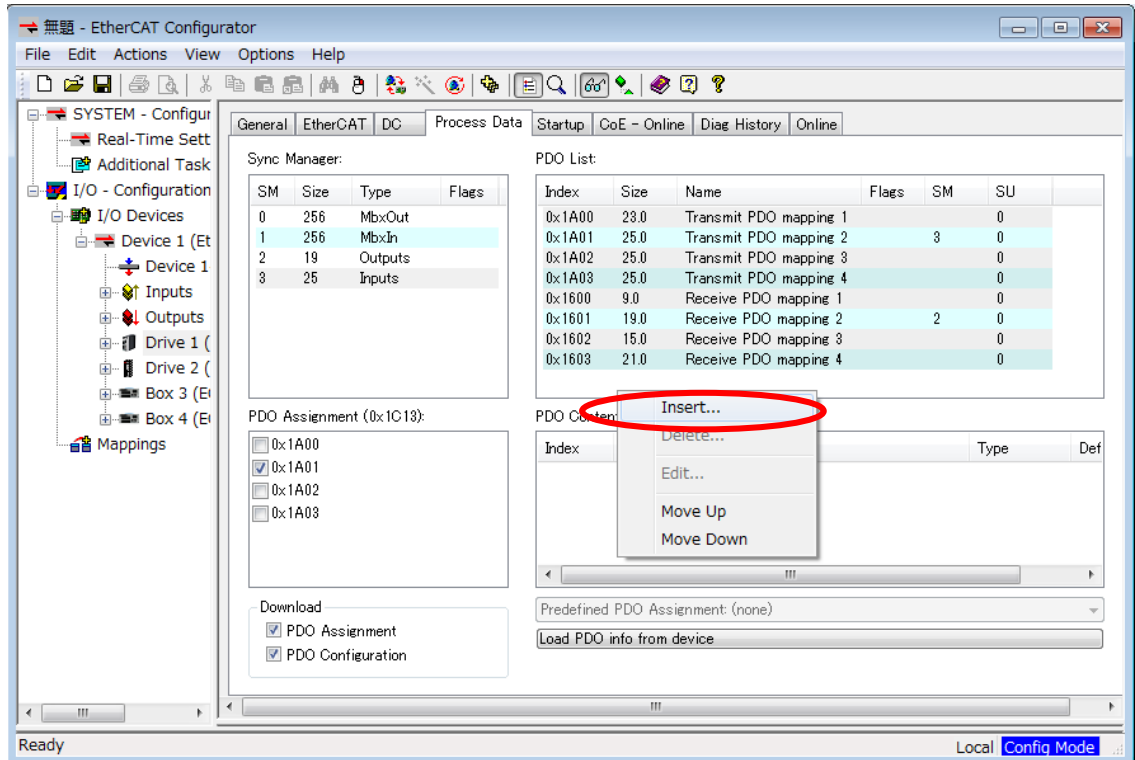
ここに必要なオブジェクトが無い場合は「PDO マッピングへのオブジェクトの編集方法」の操作方で PDO マッピングの内容を変更してください。  
 ここに必要なオブジェクトがそろっている場合は、「ENI ファイルの出力」に進んでください。

## ⑬ PDO マッピングのオブジェクト編集方法

まずは、PDO マッピングオブジェクトを追加します。

PDO マッピングオブジェクトを追加できるかどうかはスレーブのマニュアルでご確認ください。登録できない場合は、既存のマッピングオブジェクトに項目を追加しますので、本ページは飛ばして次のページに進んでください。

[PDO List]内を右クリックし、ポップアップメニューから[Insert...]をクリックします。



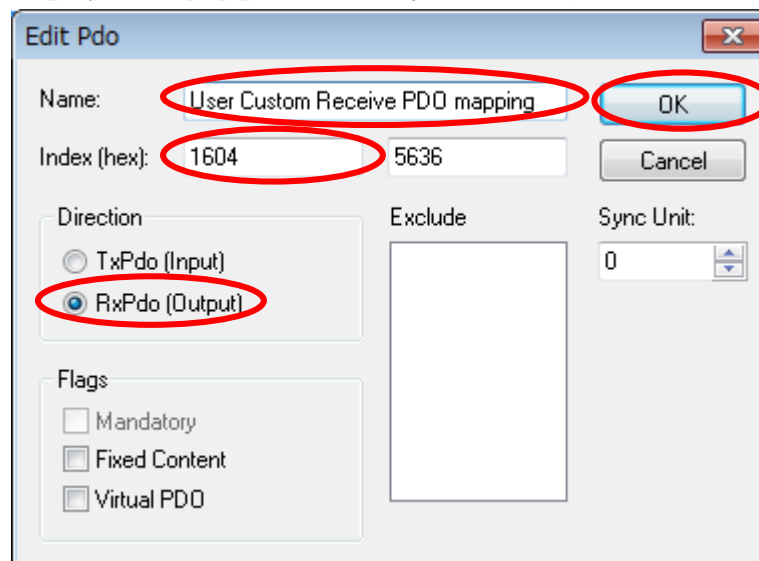
以下のような登録画面が表示されます。

[Name:] 任意の名称を付けてください。

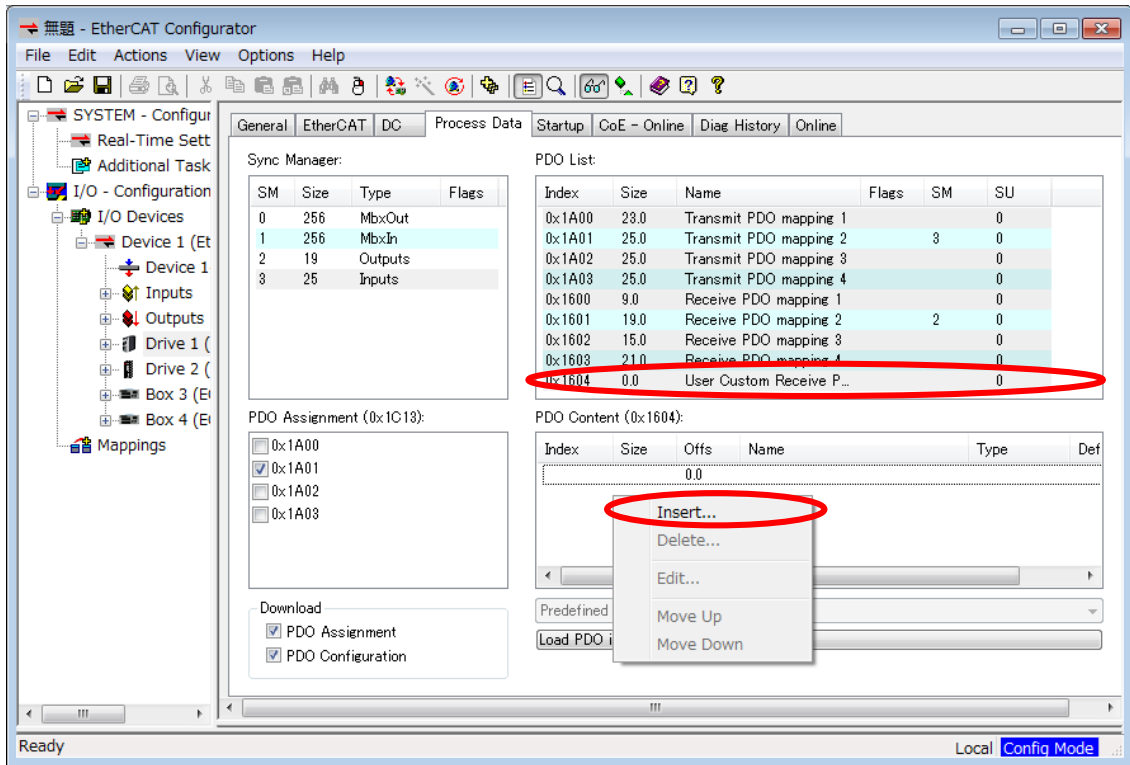
[Index (hex) :] 未登録の値を設定してください。登録可能な Index の範囲はスレーブのマニュアルで確認してください。

[Direction:] 登録するマッピングで適切な方を選択してください。

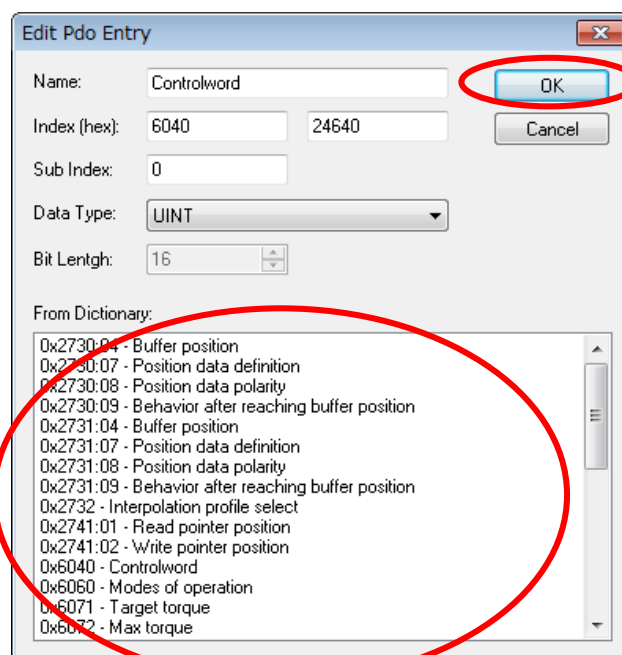
以上を選択したら [OK] をクリックし登録してください。



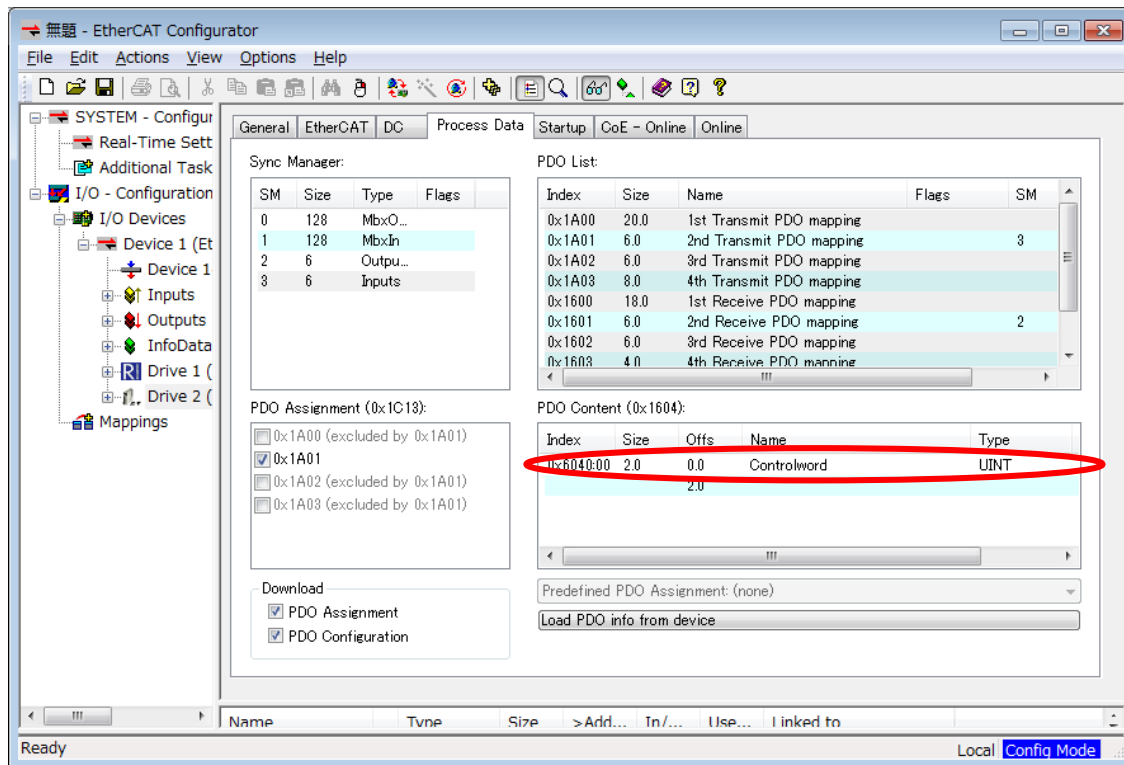
続いて、PDO マッピングオブジェクトにマップするオブジェクトを追加していきます。  
 [PDO List]で追加するマッピングオブジェクトを選択してください。  
 以下では、先ほど登録した PDO マッピングオブジェクトを選択します。  
 選択したら、[PDO Content]内を右クリックし、ポップアップメニューの[Insert...]を押してマッピングするオブジェクトを追加します。



[Insert...]をクリックすると以下のように[Edit PDO Entry]画面が表示されます。  
 [From dictionary:]にマッピング可能なオブジェクトがリストアップされますので、このリストの中から必要なオブジェクトを選択して[OK]を押してください。

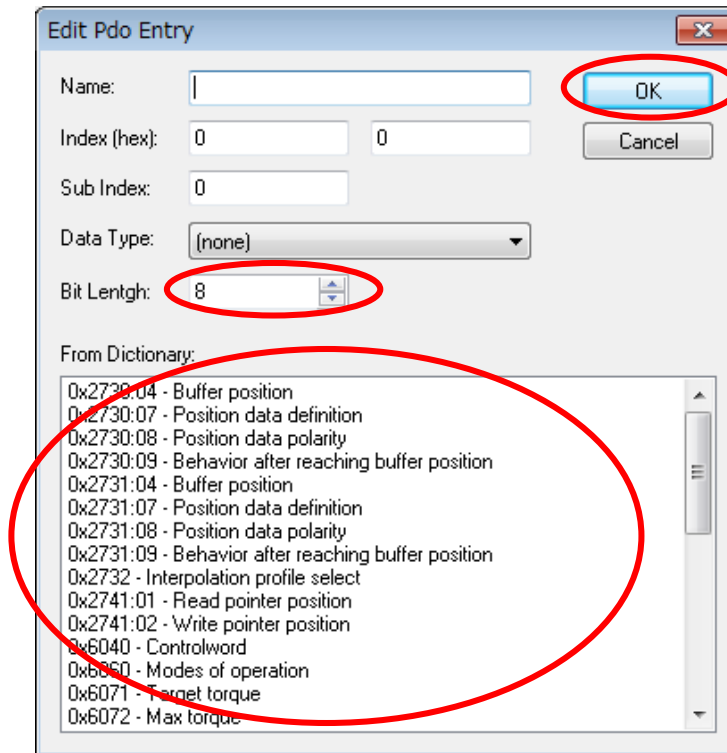


[Edit PDO Entry]画面で追加すると、以下のように[PDO Content]にオブジェクトが追加されます。  
この操作を必要なオブジェクトの数だけ繰り返してください。



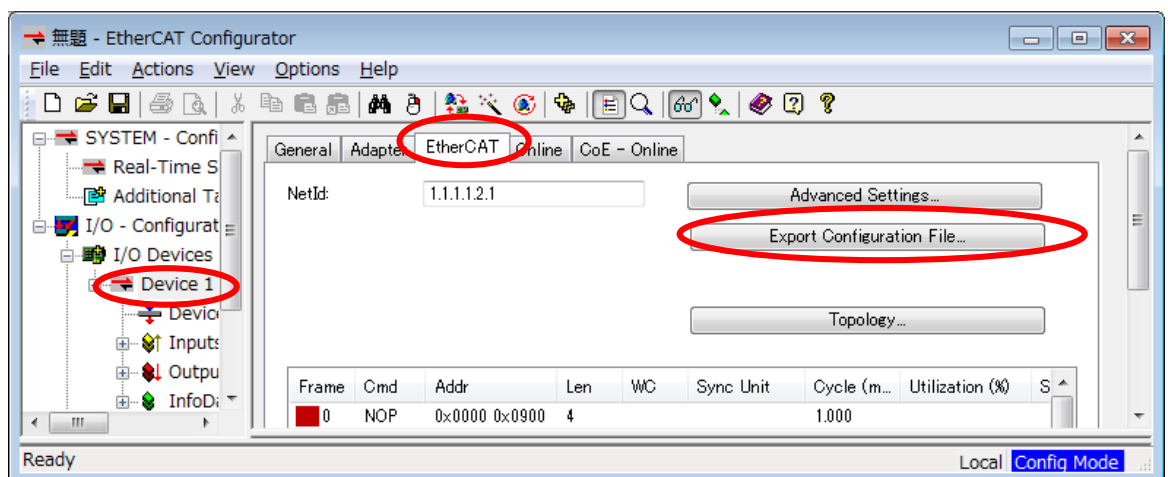
必要なオブジェクトの登録が完了したら「PDO マッピングの選択方法」の操作により、そのマッピングオブジェクトを選択してください。

※注：アライメントの登録が必要な場合があります。  
 テクノの動作確認済みのスレーブにおいて、内部が 2 バイト境界のスレーブがありました。（テクノで動作確認済みのスレーブについては、HP でご確認ください。）  
 その場合サイズが 1 バイトのオブジェクトを登録すると 1 バイトのアライメントをマップする必要があります。アライメントを登録する場合は [Edit PDO Entry] 画面で [Bit size] のみ入力して [OK] を押ししてください。

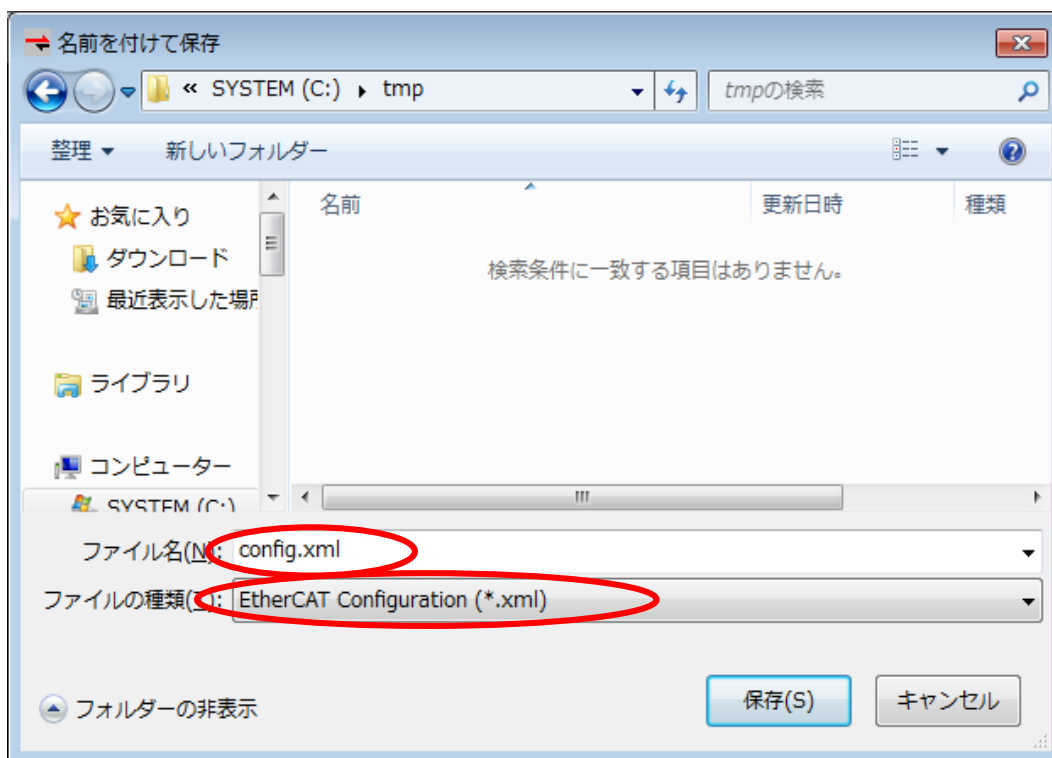


#### ⑭ ENI ファイルの出力

左ウィンドウから [Device 1] を選択後、右ウィンドウにある [EtherCAT] タブを選択します。  
 [Export Configuration File...] を押ししてください。ファイルの保存ダイアログが開きます。



ファイル名を [config.xml] として保存してください。



⑮ ENI ファイルの配置

保存したファイル [config.xml] を、INTIME 実行端末にコピーします。コピー先は下記のフォルダです。

“C:\Program Files\ALGOSYSTEM\AlgoEtherCAT\config.xml”

出荷時にデフォルトファイルを配置しておりますので、それを上書きする形になります。

4) アプリケーションの作成

「第 2 章 開発環境」を参照して、FBD プログラムを作成してください。作成したプログラムを ProConOS へダウンロードすることで、同期モーション制御プログラムが実現できます。

## 第2章 開発環境

本章では、MULTIPROGにPLCopen MC ライブラリをインストールする方法や、使用するために必要な各種設定方法について説明します。

### 2-1 インストール方法

<DVD>%setup.exe を実行してください。

PLCopen MC ライブラリを使用するためのファンクションブロックライブラリがインストールされます。

MULTIPROG のインストール先を変更している場合は、インストール先を合わせてください。

### 2-2 I/O グループ設定方法

ハードウェアの I/O グループ (IEC-61131 規格のアドレス) 宣言を行います。

本項で設定した I/O のアドレスは、変数に割り当てる事で使用可能となります。

#### 2-2-1 Input 設定

①MULTIPROG のプロジェクトから「IO\_Configuration」をダブルクリックしてください。

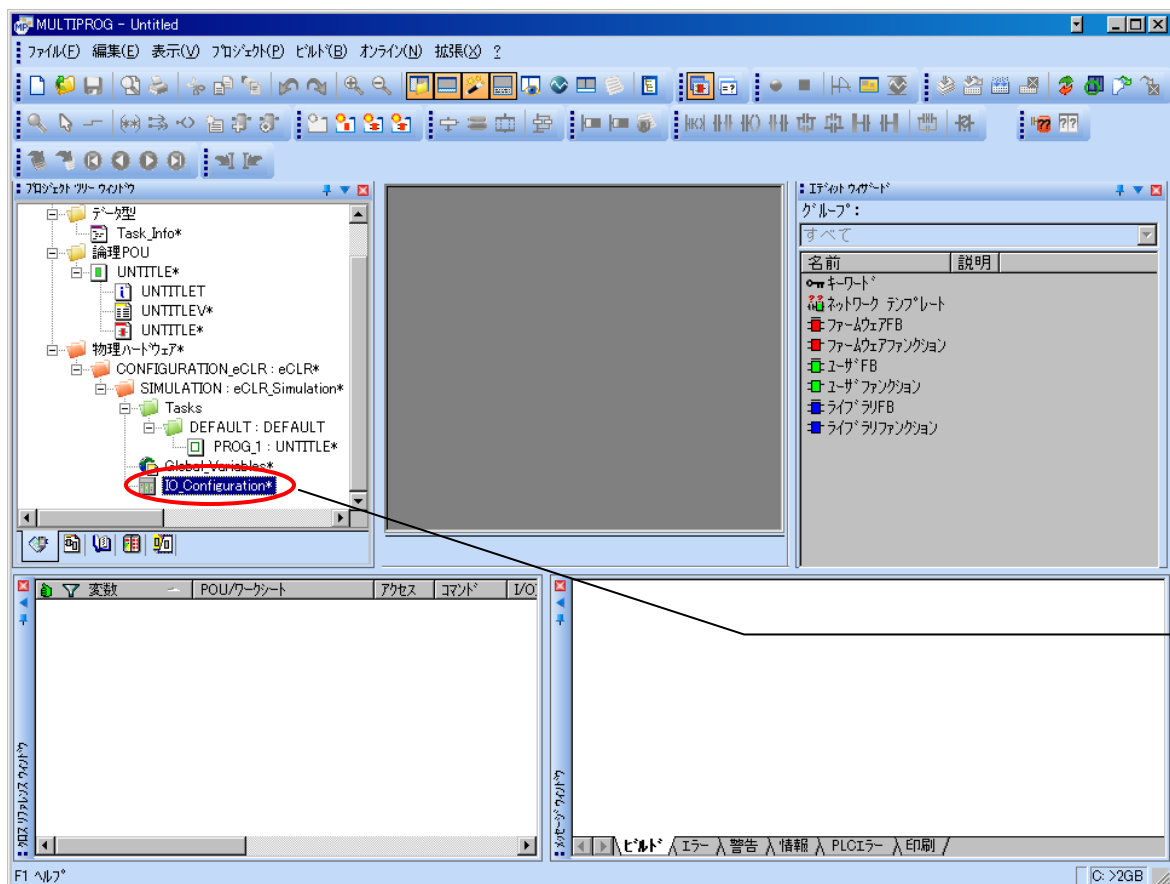


図 2-2-1-1. MULTIPROG メイン画面



②デフォルトで登録されている I/O グループを選択し、「プロパティ (P)」ボタンをクリックします。



図 2-2-1-2. I/O コンフィグレーション INPUT 登録画面

③名前 (N), タスク (T), 開始アドレス (S), 長さ (L), リフレッシュ, デバイスを設定し、ボード/IO モジュール (O) を選択してください。設定する値は、表 2-2-1-1 を参考に変更してください。

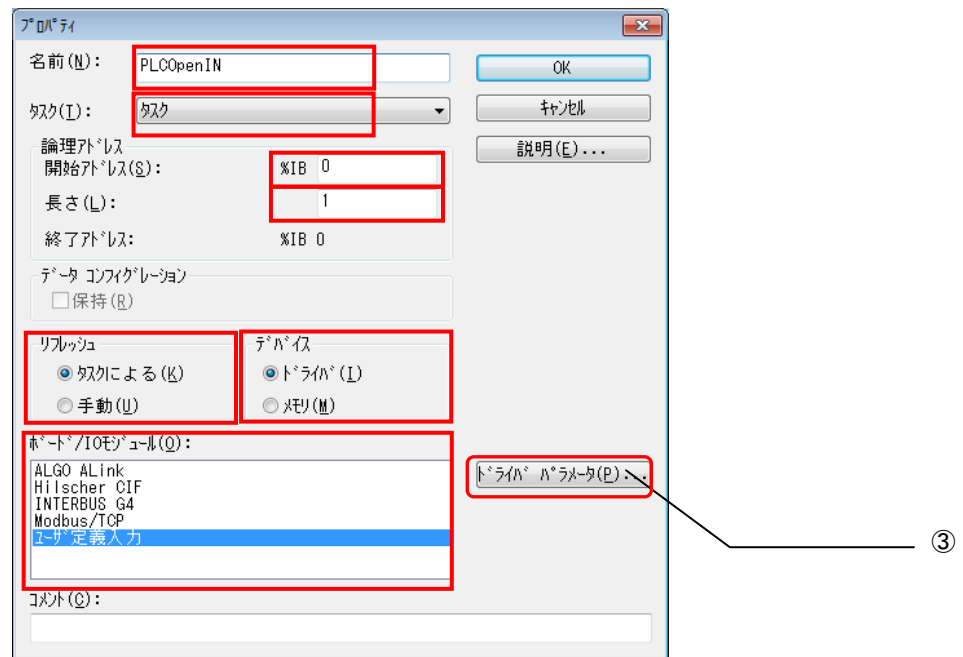


図 2-2-1-3. I/O コンフィグレーション INPUT プロパティ画面

表 2-2-1-1. I/O コンフィグレーションプロパティ

パラメータ名	説明	設定値
名前 (N)	I/O グループ名称	任意
タスク (T)	全入出力を行うタスク	任意
開始アドレス (S)	IEC61131 規格の開始アドレス	変更無し
長さ (L)	I/O ドライバを使用するために最小値を設定します。	1
リフレッシュ	-	タスクによる (K)
デバイス	-	ドライバ (I)
ボード/IO モジュール (O)	接続モジュール名称	ユーザー定義入力

設定完了後、「ドライバパラメータ (P)」ボタンをクリックします。

- ④ ドライバパラメータを設定します。  
設定する値は表 2-2-1-2 を参考にしてください。

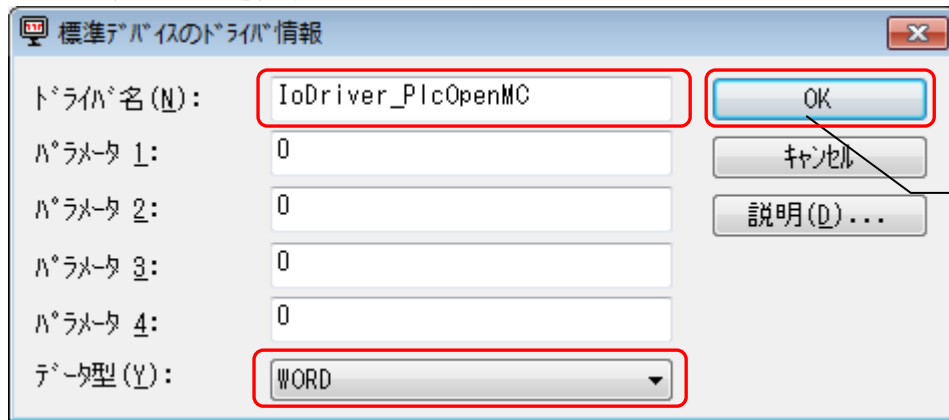


図 2-2-1-4. I/O コンフィグレーション ドライバパラメータ画面

表 2-2-1-2. ドライバパラメータ

パラメータ名	説明	設定値
ドライバ名 (N)	-	IoDriver_PlOpenMC
パラメータ 1	-	0
パラメータ 2	-	0
パラメータ 3	-	0
パラメータ 4	-	0
データ型 (Y)	-	WORD

※ ドライバパラメータに設定する値は、Input/Output どちらにも同じ値を設定してください。

設定完了後、「OK」ボタンをクリックします。

- ⑤ 以上で設定は完了になります。  
図 2-2-1-5 のように、設定した値が表示されている事を確認してください。

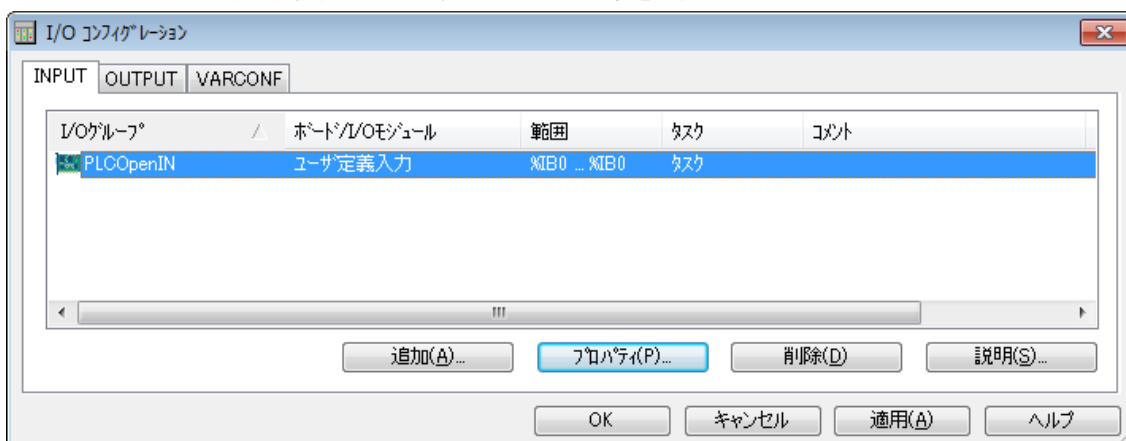


図 2-2-1-5. I/O コンフィグレーション INPUT 設定完了画面

## 2-2-2 Output 設定

①MULTIPROG のプロジェクトから「IO\_Configuration」をダブルクリックしてください。

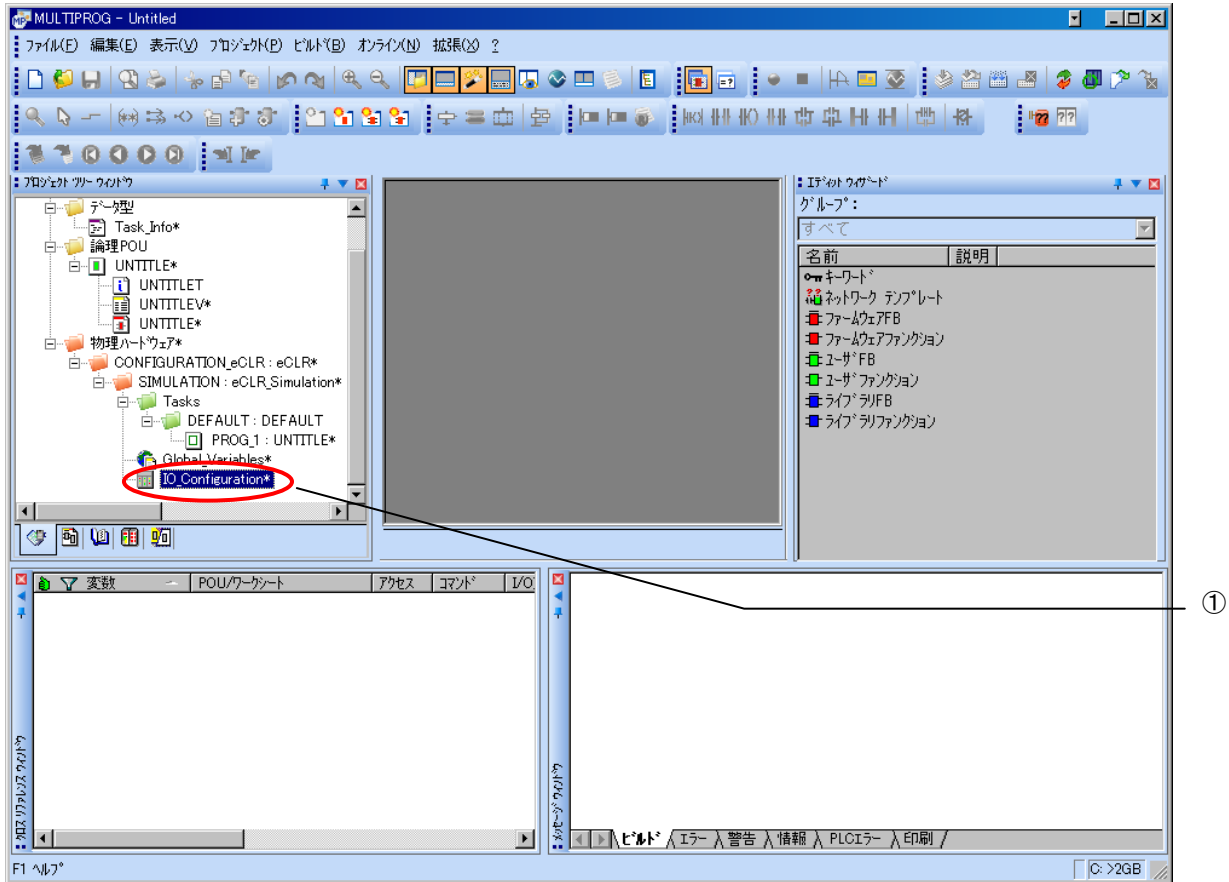


図 2-2-2-1. MULTIPROG メイン画面

②デフォルトで登録されている I/O グループを選択し、「プロパティ (P)」ボタンをクリックします。



図 2-2-2-2. I/O コンフィグレーション OUTPUT 登録画面

- ③名前(N), タスク(T), 開始アドレス(S), 長さ(L), リフレッシュ, デバイスを設定し、ボード/I/O モジュール(O)を選択してください。設定する値は、表 2-2-2-1 を参考に更新してください。

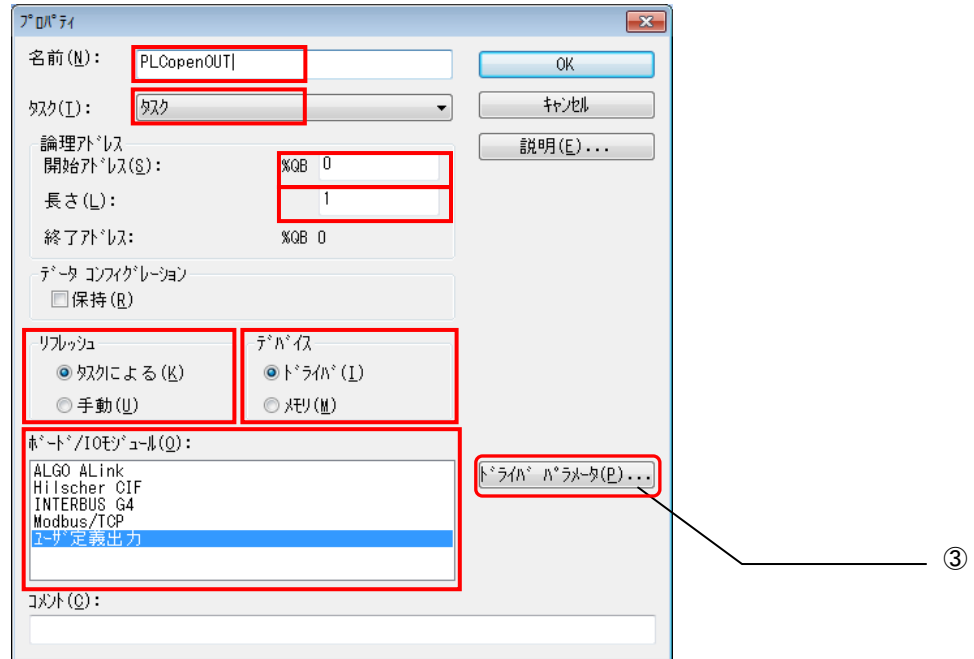


図 2-2-2-3. I/O コンフィグレーション OUTPUT プロパティ画面

表 2-2-2-1. I/O コンフィグレーションプロパティ

パラメータ名	説明	設定値
名前(N)	I/O グループ名称	任意
タスク(T)	全入出力を行うタスク	任意
開始アドレス(S)	IEC61131 規格の開始アドレス	変更無し
長さ(L)	I/O ドライバを使用するために最小値を設定します。	1
リフレッシュ	-	タスクによる(K)
デバイス	-	ドライバ(I)
ボード/I/O モジュール(O)	接続モジュール名称	ユーザー定義入力

設定完了後、「ドライバパラメータ(P)」ボタンをクリックします。

- ④ ドライバパラメータを設定します。  
設定する値は表 2-2-2-2 を参考にしてください。

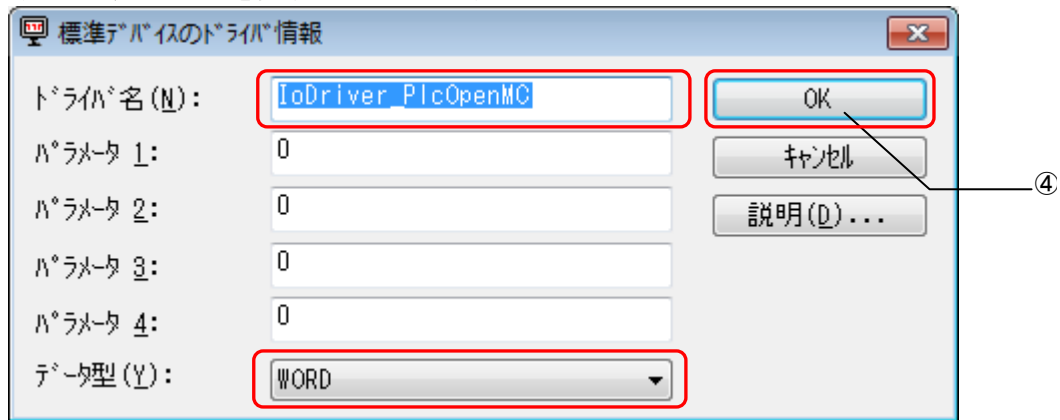


図 2-2-2-4. I/O コンフィグレーション ドライバパラメータ画面

表 2-2-2-2. ドライバパラメータ

パラメータ名	説明	設定値
ドライバ名(N)	-	IoDriver_PlcOpenMC
パラメータ 1	-	0
パラメータ 2	-	0
パラメータ 3	-	0
パラメータ 4	-	0
データ型(Y)	-	WORD

※ ドライバパラメータに設定する値は、Input/Output どちらにも同じ値を設定してください。

設定完了後、「OK」ボタンをクリックします。

- ⑤ 以上で設定は完了になります。  
図 2-2-2-5 のように、設定した値が表示されている事を確認してください。

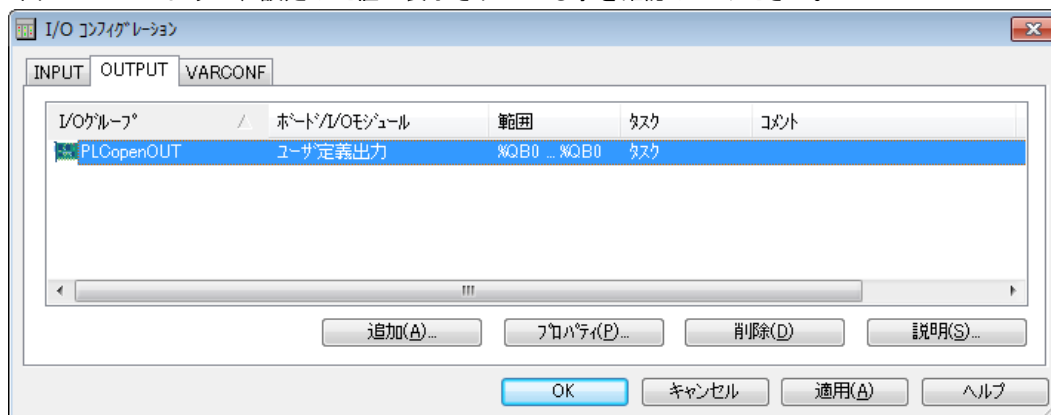


図 2-2-2-5. I/O コンフィグレーション OUTPUT 設定完了画面

### 2-2-3 ダミー変数の登録

「2-2-1 Input 設定」、「2-2-2 Output 設定」で登録した PLCopen 用の I/O ドライバを機能させるためには、登録した変数をプロジェクトに組み込み使用する必要があります。

下記の手順で、ダミー変数の登録を行ってください。

- ① MULTIPROG のプロジェクトから「Global\_Variables」をダブルクリックしてください。

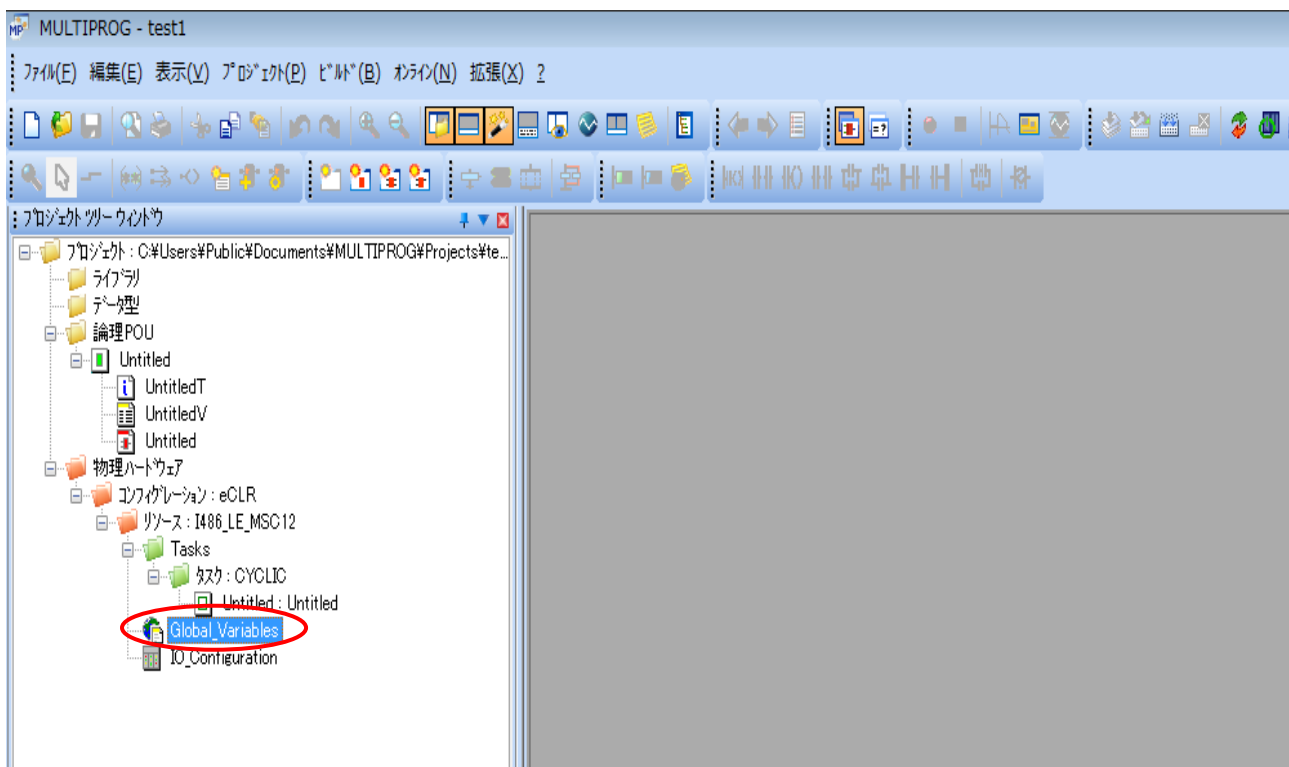


図 2-2-3-1. MULTIPROG メイン画面

- ② 新しいグループを追加します。(既存のグループに追加する場合は②、③の手順は必要ありません。④へ進んでください。)
- 赤枠の中で右クリックし、下図のメニューを開き、「グループの挿入」をクリックします。

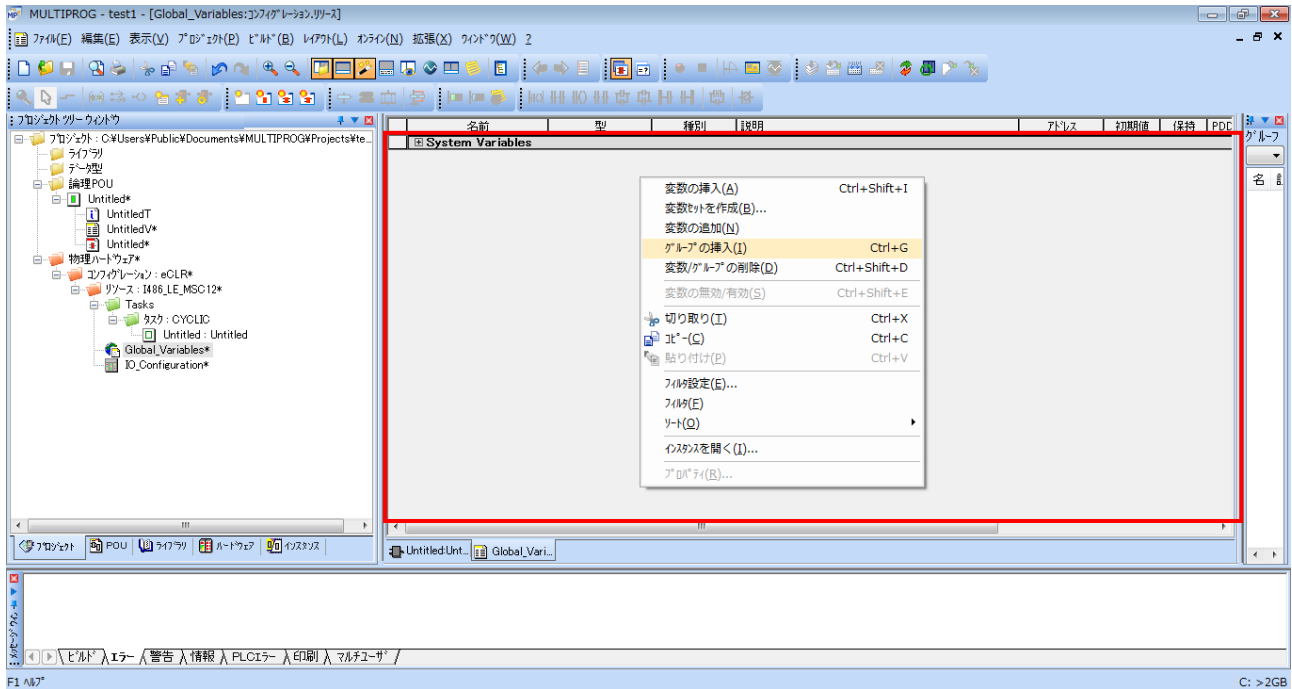


図 2-2-3-2. グローバル変数右クリックメニュー

- ③ 新しいグループが作成されます。必要に応じて、グループ名を変更してください。

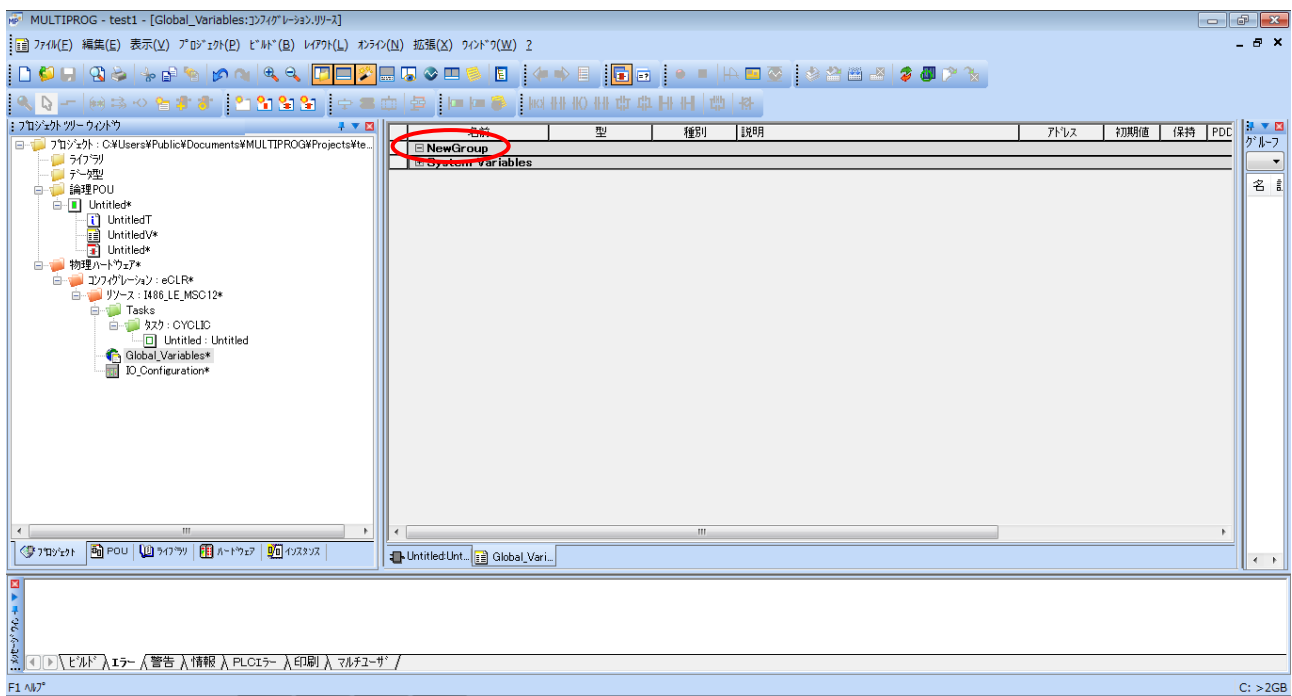


図 2-2-3-3. NewGroup 追加後

- ④ ダミー変数を追加します。追加したいグループの上で右クリックし、下図のメニューを開き、「変数の挿入」をクリックします。IN用とOUT用の2つのダミー変数を登録する必要があるため、2回実行します。

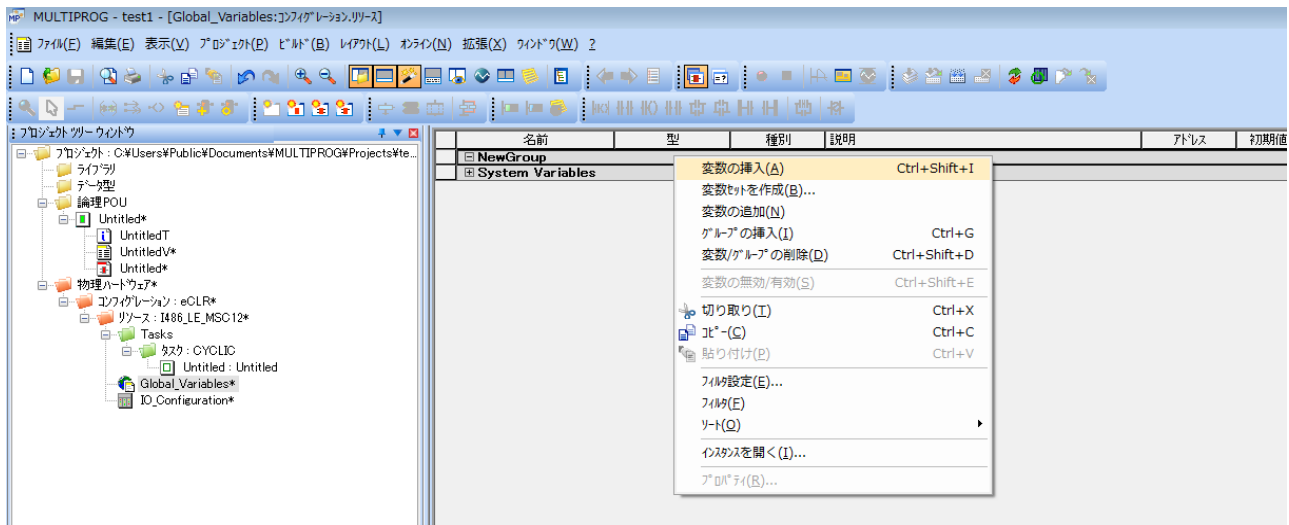


図 2-2-3-4. グローバル変数右クリックメニュー

- ⑤ ダミー変数を設定します。「型」と「アドレス」の項目の設定を行います。

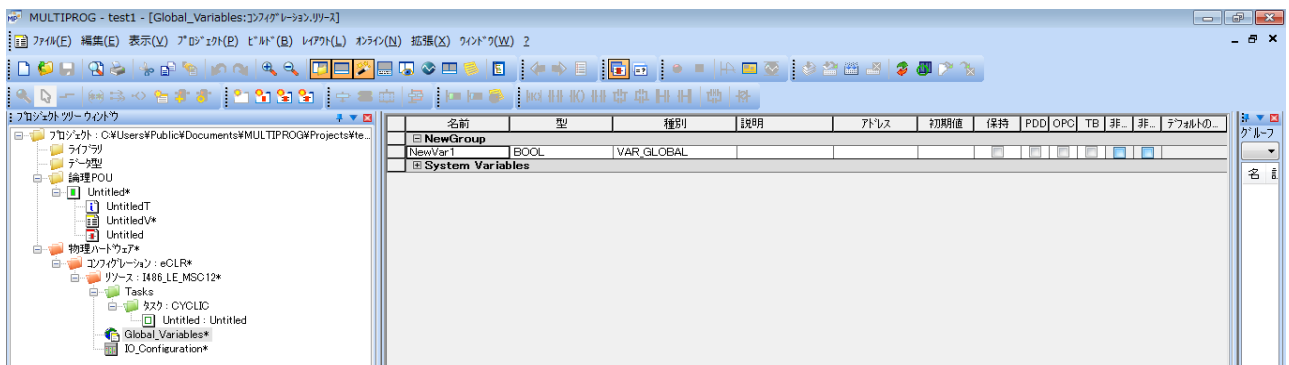


図 2-2-3-5. 新規変数追加画面

- ⑥ 表 2-2-3-1 と表 2-2-3-2 の設定を行います。表以外の項目については変更する必要はありません。

表 2-2-3-1. ダミー変数設定リスト

設定項目	設定値	備考
名前	NewVar1	名前は任意に変更してください。
型	BYTE	
アドレス	%QB0	

表 2-2-3-2. IN用ダミー変数設定リスト

設定項目	設定値	備考
名前	NewVar2	名前は任意に変更してください。
型	BYTE	
アドレス	%IB0	

名前	型	種別	説明	アドレス	初期値	保持	PDD	OPC	TB
NewGroup									
NewVar2	BYTE	VAR_GLOBAL		%IB0		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
NewVar1	BYTE	VAR_GLOBAL		%QB0		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

図 2-2-3-4. ダミー変数設定後

**※注：必ず IN 用と OUT 用 2 種類のダミー変数を登録してください。**



## 第3章 ファンクションブロック

本章では、PLCopen 仕様のモーションコントロールファンクションブロックを使用するために必要な内容について説明します。

### 3-1 機能概要

MULTIPROG 上で PLCopen に準拠する制御を実現するために、ファンクションブロック形式のライブラリを用意しました。各種コマンド毎のファンクションブロックを使用することで、PLCopen 仕様モーションコントロールが可能となります。

ファンクションブロックは管理用、動作用、原点復帰用、同期制御管理用、同期制御動作用の5種類に分類されます。

#### 1) PLCopen 仕様 MC 管理ファンクションブロック

PLCopen 仕様で定義されている MC ファンクションブロックの内、管理ファンクションブロックの一覧を表 3-1-1 に示します。詳細は『3-3-1 PLCopen 仕様 管理ファンクションブロック』を参照してください。

表 3-1-1. PLCopen 仕様 MC 管理ファンクションブロック一覧

機能概略	ファンクションブロック	サポート	制御軸
サーボ ON/OFF	MC_Power	○	単軸
状態遷移ステータス読み込み	MC_ReadStatus	○	単軸
軸エラー読み込み	MC_ReadAxisError	○	単軸
軸パラメータ (LREAL 型) 読み込み	MC_ReadParameter	○	単軸
軸パラメータ (BOOL 型) 読み込み	MC_ReadBoolParameter	○	単軸
軸パラメータ (BYTE 型) 読み込み	MC_ReadByteParameter	○	単軸
軸パラメータ (WORD 型) 読み込み	MC_ReadWordParameter	○	単軸
軸パラメータ (DWORD 型) 読み込み	MC_ReadDwordParameter	○	単軸
軸パラメータ (LREAL 型) 書き込み	MC_WriteParameter	○	単軸
軸パラメータ (BOOL 型) 書き込み	MC_WriteBoolParameter	○	単軸
軸パラメータ (BYTE 型) 書き込み	MC_WriteByteParameter	○	単軸
軸パラメータ (WORD 型) 書き込み	MC_WriteWordParameter	○	単軸
軸パラメータ (DWORD 型) 書き込み	MC_WriteDwordParameter	○	単軸
現在位置読み込み	MC_ReadActualPosition	○	単軸
現在速度読み込み	MC_ReadActualVelocity	○	単軸
現在トルク読み込み	MC_ReadActualTorque	○	単軸
エラーリセット	MC_Reset	○	単軸
CAM 動作定義データ選択	MC_CamTableSelect	×	多軸

○：仕様通りにサポート △：機能を限定してサポート ×：サポートしない  
色が変わっているファンクションブロックは、AI-Motion で追加されているものです。

## 2) PLCopen 仕様 MC 動作ファンクションブロック

PLCopen 仕様で定義されている MC ファンクションブロックの内、動作ファンクションブロックの一覧を表 3-1-2 に示します。詳細は『3-3-2 PLCopen 仕様 動作ファンクションブロック』を参照してください。

表 3-1-2. PLCopen 仕様 MC 動作ファンクションブロック一覧

名称	ファンクションブロック	サポート	制御軸
絶対位置決め	MC_MoveAbsolute	○	単軸
相対位置決め	MC_MoveRelative	○	単軸
加算位置決め	MC_MoveAdditive	○	単軸
位置決め中割り込み位置決め	MC_MoveSuperimposed	×	単軸
定速動作	MC_MoveVelocity	○	単軸
トルク制御	MC_TorqueControl	○	単軸
原点サーチ	MC_Home	△	単軸
軸停止	MC_Stop	○	単軸
位置データによる繰り返し	MC_PositionProfile	×	単軸
速度データによる繰り返し	MC_VelocityProfile	×	単軸
加速度データによる繰り返し	MC_AccelerationProfile	×	単軸
主軸に対しての CAM 同期	MC_CamIn	×	多軸
主軸からの CAM 同期解除	MC_CamOut	×	多軸
主軸に対しての GEAR 同期	MC_GearIn	△	多軸
主軸からの GEAR 同期解除	MC_GearOut	△	多軸
主軸との位相同期	MC_Phasing	×	多軸

○：仕様通りにサポート △：機能を限定してサポート ×：サポートしない

■：ファンクションブロックは、AI-Motion で追加されているものです。

■：ファンクションブロックは、PLCopen 本来の仕様と異なる仕様で作成されています。

## 3) PLCopen 仕様 MC 原点復帰ファンクションブロック

PLCopen 仕様で定義されている MC ファンクションブロックの内、原点復帰用のファンクションブロックの一覧を表 3-1-3 に示します。詳細は『3-3-3 PLCopen 仕様 原点復帰ファンクションブロック』を参照してください。

\*) 原点復帰を行うためにはサーボパックへの以下の信号入力と設定が必要になります。

リミットセンサ信号を接続しない場合、MC\_StepAbsSwitch, MC\_StepLimitSwitch は使用出来ません。個々のパラメータ設定方法は各サーボパックマニュアルを参照してください。

## 必要な信号

正転リミットセンサ信号 (MC\_StepAbsSwitch, MC\_StepLimitSwitch)  
 逆転リミットセンサ信号 (MC\_StepAbsSwitch, MC\_StepLimitSwitch)  
 原点センサ信号 (MC\_StepAbsSwitch)

## 必要な設定

正転リミットセンサ (P-OT) 有効  
 逆転リミットセンサ (N-OT) 有効  
 外部ラッチ信号 1 有効 (外部ラッチ信号 1 への原点センサ信号入力設定)

リミットセンサ入力時の停止方法設定 (減速停止)  
 非常停止トルク

表 3-1-3. PLCopen 仕様 MC 原点サーチファンクションブロック一覧

名称	ファンクションブロック	サポート	制御軸
リミット/原点センサ使用	MC_StepAbsSwitch	○	単軸
リミットセンサ使用	MC_StepLimitSwitch	○	単軸
機械的限界	MC_StepBlock	×	単軸
原点サーチ終了	MC_FinishHoming	○	単軸
Z 相検知による原点復帰	MC_StepRefPulse	○	単軸
現在位置を指定して変更	MC_StepDirect	○	単軸
現在位置を 0 位置に変更	MC_StepAbsolute	○	単軸
位置決め中の SW 入力位置変更	MC_StepReferenceFlyingSwitch <MC_StepRefFlyingSwitc>	×	単軸
位置決め中の Z 相検知位置変更	MC_StepFferenceFlyingRefPulse <MC_StepRefFlyingRefPulse>	×	単軸
原点サーチ中断	MC_AbortPassiveHoming	×	単軸

○ : 仕様通りにサポート △ : 機能を限定してサポート × : サポートしない

<> : 本来の PLCopen 仕様の FB 名称が長いため、本ライブラリでは<>内の名称を使用しています。

## 4) PLCopen仕様 MC Part4 管理ファンクションブロック

PLCopen仕様で定義されているMCファンクションブロックの内、管理ファンクションブロックの一覧を表3-1-4に示します。詳細は『3-3-4 PLCopen仕様 MC Part4 管理ファンクションブロック』を参照してください。

表 3-1-4. PLCopen仕様 MC Part4 管理ファンクションブロック一覧

機能概略	ファンクションブロック名	サポート	制御軸
軸追加	MC_AddAxisToGroup	○	同期
軸削除	MC_RemoveAxisFromGroup	○	同期
軸グループ解除	MC_UngroupAllAxes	○	同期
軸グループ設定読出	MC_GroupReadConfiguration <MC_GroupReadCfg>	○	同期
軸グループ有効	MC_GroupEnable	○	同期
軸グループ無効	MC_GroupDisable	○	同期
運動学的変換設定	MC_SetKinTransform <MC_SetKinTrans>	×	同期
直交座標変換設定	MC_SetCartesianTransform <MC_SetCartesianTrans>	×	同期
座標変換設定	MC_SetCoordinateTransform <MC_SetCoordinateTrans>	×	同期
運動学的変換読出	MC_ReadKinTransform <MC_ReadKinTrans>	×	同期
直交座標変換読出	MC_ReadCartesianTransform <MC_ReadCartesianTrans>	×	同期
座標変換読出	MC_ReadCoordinateTransform <MC_ReadCoordinateTrans>	×	同期
軸グループ現在位置変更	MC_GroupSetPosition <MC_GroupSetPos>	○	同期
軸グループ現在位置読出	MC_GroupReadActualPosition <MC_GroupReadActualPos>	○	同期
軸グループ現在速度読出	MC_GroupReadActualVelocity <MC_GroupReadActualVel>	○	同期
軸グループ現在加速度読出	MC_GroupReadActualAcceleration <MC_GroupReadActualAcc>	×	同期
軸グループステータス読出	MC_GroupReadStatus	○	同期
軸グループエラー読出	MC_GroupReadError	○	同期
軸グループエラーリセット	MC_GroupReset	○	同期
経路選択	MC_PathSelect	×	同期
軸グループオーバーライド値設定	MC_GroupSetOverride	○	同期
動的座標変換設定	MC_SetDynCoordTransform <MC_SetDynCoordTrans>	×	同期

○：仕様通りにサポート △：機能を限定してサポート ×：サポートしない

◇：本来のPLCopen仕様のFB名称が長い為、本ライブラリでは◇内の名称を使用しています。

## 5) PLCopen 仕様 MC Part4 動作ファンクションブロック

PLCopen 仕様で定義されている MC ファンクションブロックの内、動作ファンクションブロックの一覧を表 3-1-5 に示します。詳細は『3-3-5 PLCopen 仕様 MC Part4 動作ファンクションブロック』を参照してください。

表 3-1-5. PLCopen 仕様 MC Part4 動作ファンクションブロック一覧

機能概略	ファンクションブロック名	サポート	制御軸
軸グループ原点復帰	MC_GroupHome	×	同期
軸グループ強制停止	MC_GroupStop	○	同期
軸グループ停止	MC_GroupHalt	○	同期
軸グループ一時停止	MC_GroupInterrupt	○	同期
軸グループ一時停止解除	MC_GroupContinue	○	同期
絶対値直線補間	MC_MoveLinearAbsolute	○	同期
相対値直線補間	MC_MoveLinearRelative	○	同期
絶対値円弧補間	MC_MoveCircularAbsolute	○	同期
相対値円弧補間	MC_MoveCircularRelative	○	同期
軸グループ絶対値位置決め	MC_MoveDirectAbsolute	×	同期
軸グループ相対値位置決め	MC_MoveDirectRelative	×	同期
指定経路移動	MC_MovePath	×	同期
グループへの単軸同期動作	MC_SyncAxisToGroup	×	同期
単軸へのグループ同期動作	MC_SyncGroupToAxis	×	同期
コンベヤ追従動作	MC_TrackConveyorBelt	×	同期
ロータリテーブル追従動作	MC_TrackRotaryTable	×	同期

○：仕様通りにサポート △：機能を限定してサポート ×：サポートしない

## 6) PLCopen 仕様範囲外の特殊FB

当社オリジナルの PLCopen 仕様範囲外の FB の一覧を表 3-1-6 に示します。FB によっては、EtherCAT (ETC) 版のみサポートする場合と MECHATROLINK-III (ML3) のみサポートする FB があります。詳細は「3-3-6 PLCopen 仕様範囲外 特殊ファンクションブロック」を参照してください。

表 3-1-6. PLCopen 仕様範囲外ファンクションブロック一覧

機能概略	ファンクションブロック	制御軸	ETC	ML3
トリガによる複数軸一斉停止	MC_IntelligentStop	複数軸	○	×

○：サポートしている ×：サポートしていない

### 3-2 使用方法

MULTIPROG のプロジェクトでユニット毎のファンクションブロックを使用するためには、プロジェクト毎に登録が必要となります。

本項では登録方法について説明します。

- ① MULTIPROG 画面の左ペインにある「ライブラリ」を選択します。

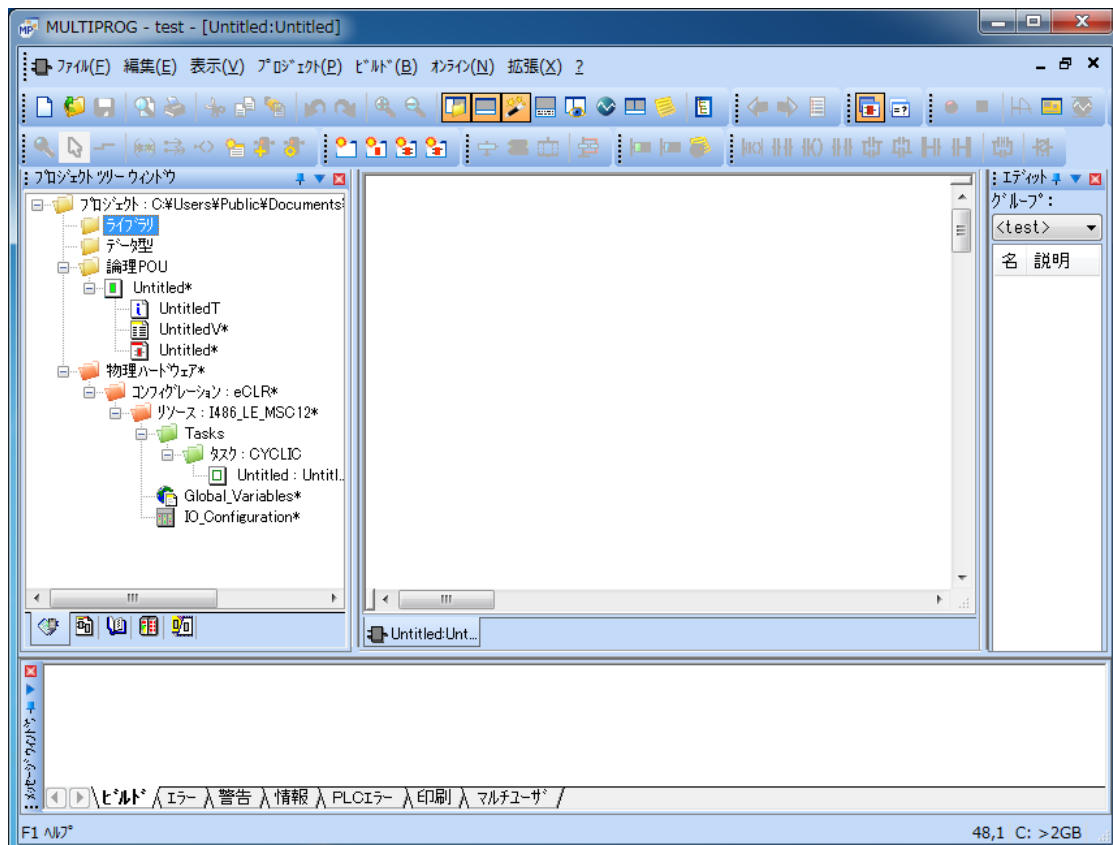


図 3-2-1. ライブラリ選択画面

- ② 「ライブラリ」を右クリックし「挿入(I)」→「ファームウェアライブラリ(F)」を選択します。

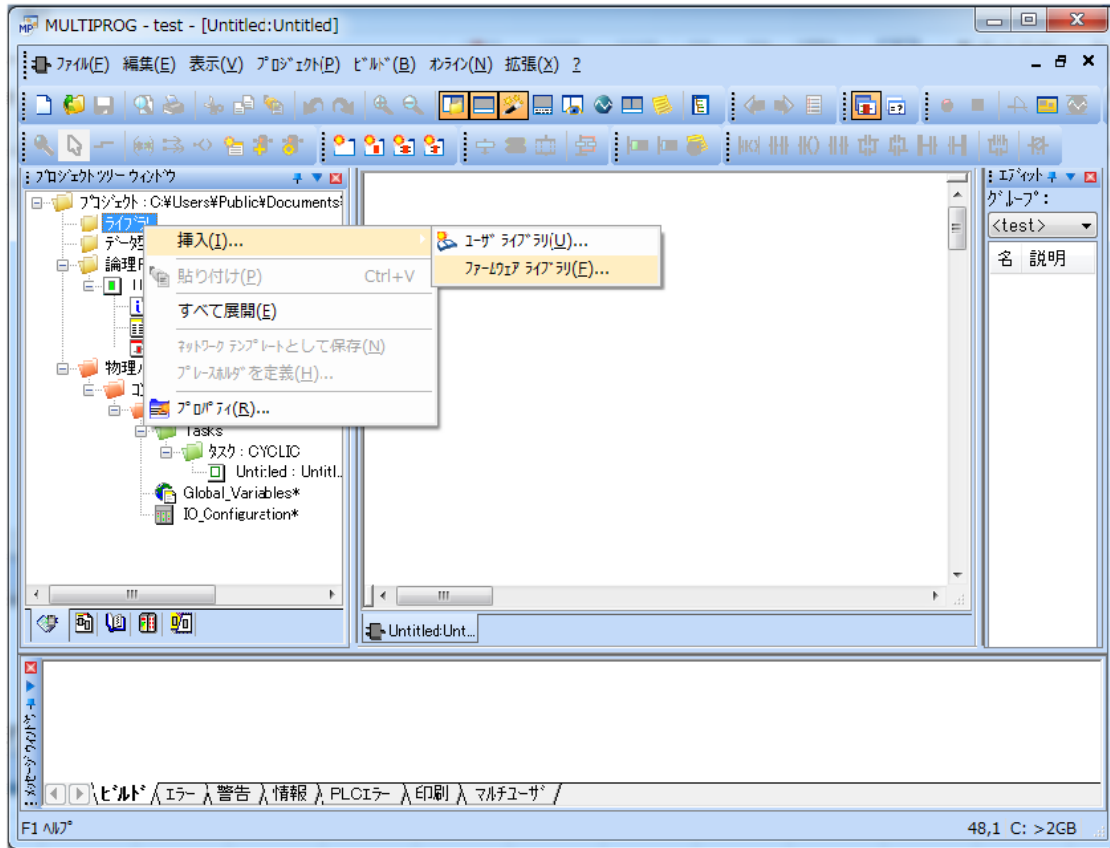


図 3-2-2. ライブラリ挿入画面

- ③ 図 3-2-3 の画面が表示されますので、「MP\_FwLib\_PlcOpenMC」という名前のディレクトリを選択してください。

ファームウェアライブラリのインストール先はデフォルトの場合は下記の間所となります。

「C:\Program File\KW-Software\MULTIPROG 5.35 Build XXX\plc\FW\_LIB」

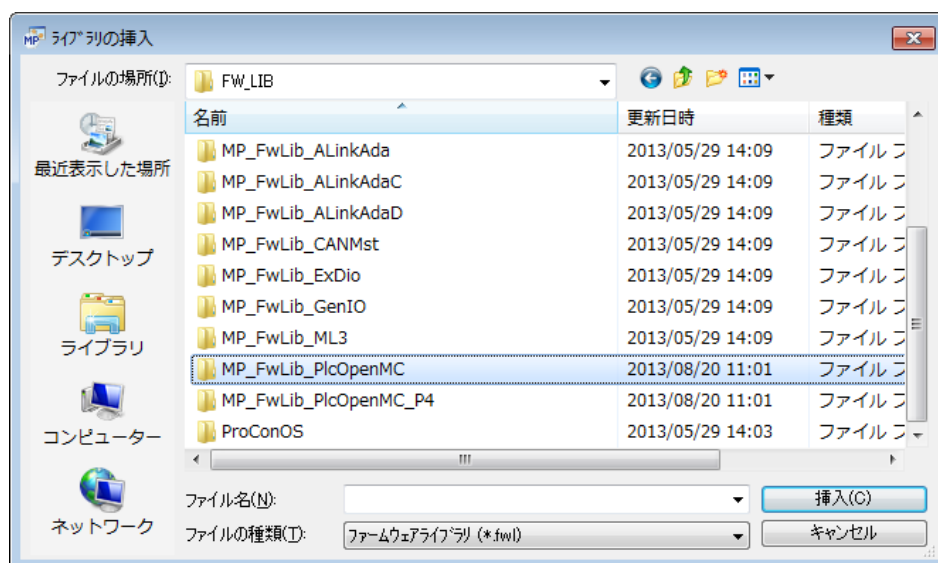


図 3-2-3. 挿入ライブラリディレクトリ選択

- ④ 挿入する FWL ファイルを選択して、「挿入(C)」ボタンを押すことで、登録が完了します。

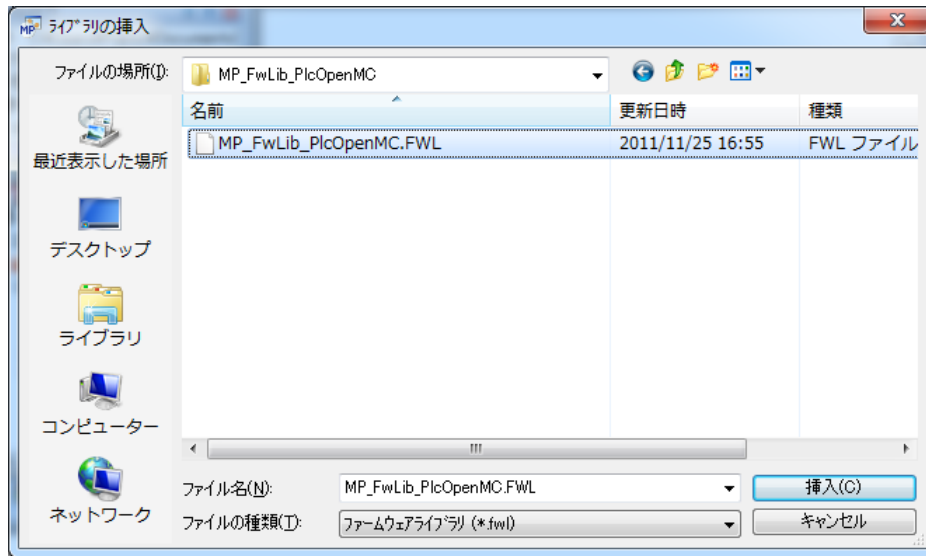


図 3-2-4. 挿入ライブラリファイル選択

- ⑤ 挿入した、FWL の FB を使用するには、右ペインのグループで「MP\_FwLib\_PlcOpenMC」を選択してください。登録されている FB の一覧が表示されます。

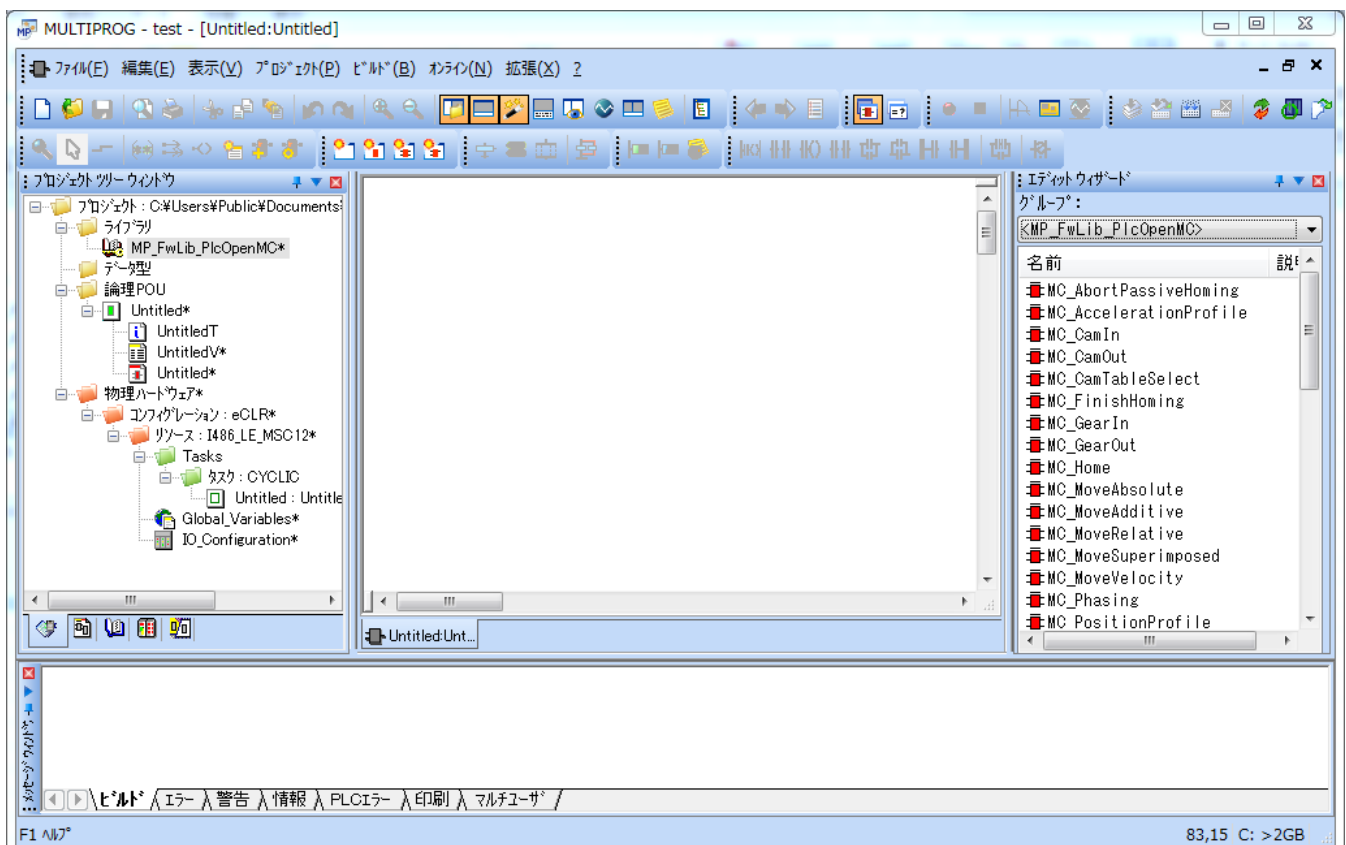


図 3-2-5. ライブラリ挿入後画面



- ⑥ Part4のFBライブラリを追加するには、図3-2-6の画面が表示されますので、「MP\_FwLib\_PlcOpenMC\_P4」という名前のディレクトリを選択してください。  
ファームウェアライブラリのインストール先はデフォルトの場合は下記の場合となります。  
「C:\Program File\KW-Software\MULTIPROG 5.35 Build XXX\plc\FW\_LIB」

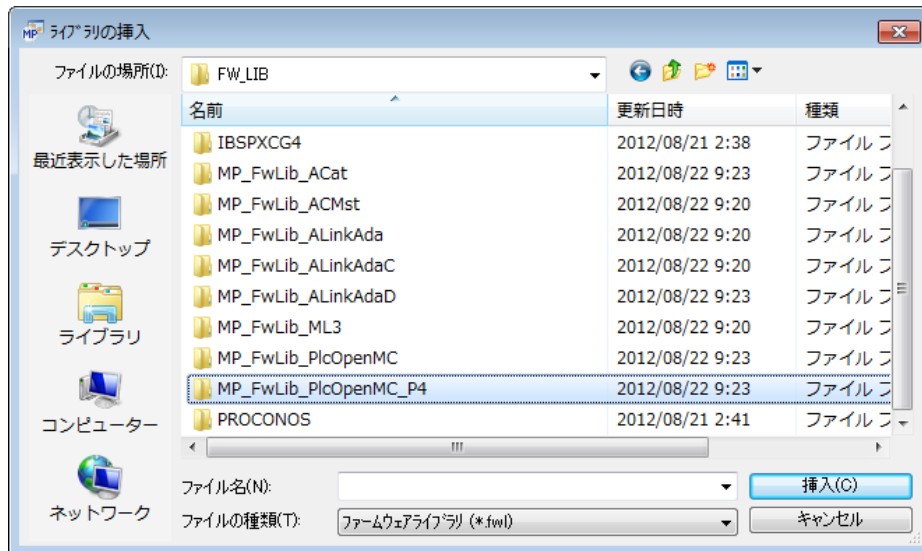


図 3-2-6. 挿入ライブラリディレクトリ選択

- ⑦ 挿入する FWL ファイルを選択して、「挿入(C)」ボタンを押すことで、登録が完了します。

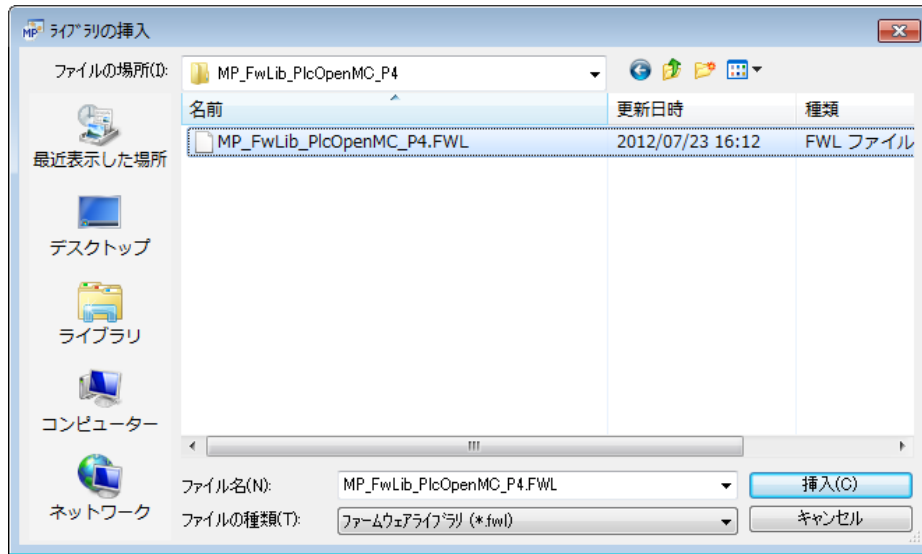


図 3-2-7. 挿入ライブラリファイル選択

- ⑧ 挿入した、FWL の FB を使用するには、右ペインのグループで「MP\_FwLib\_PlcOpenMC\_P4」を選択してください。登録されている FB の一覧が表示されます。

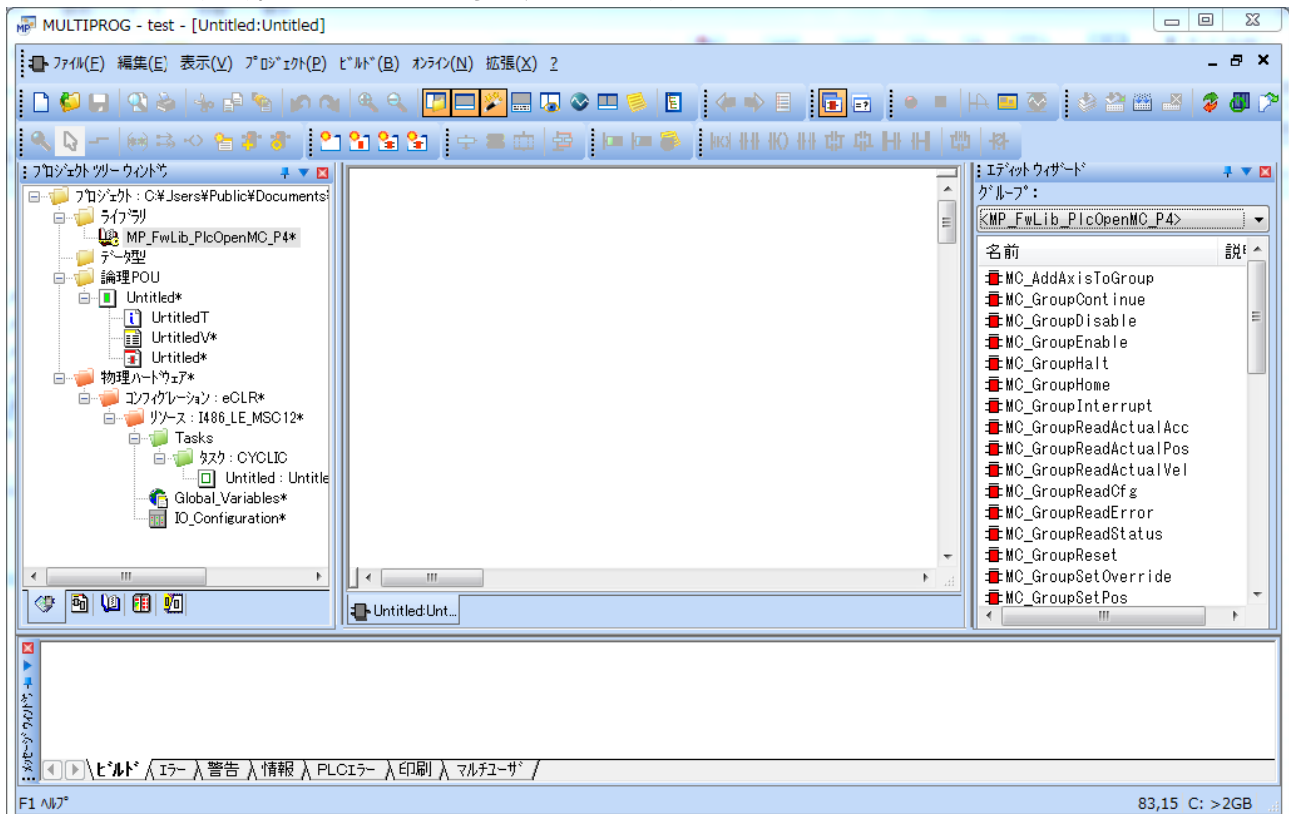


図 3-2-8. ライブラリ挿入後画面

### 3-3 ファンクションブロックリファレンス

本項では MULTIPROG 用に用意した PLCopen 仕様 MC ファンクションブロックを使用するために必要な通信設定と各 MC ファンクションブロックについて説明します。各 MC ファンクションブロックは PLCopen 仕様に従い作成されていますので、詳細については PLCopen が発行している技術仕様書「モーションコントロール用ファンクションブロック」なども参考にしてください。

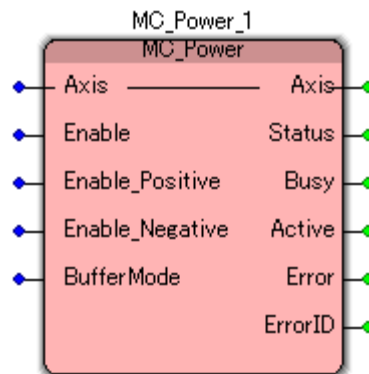
#### 3-3-1 PLCopen 仕様 管理ファンクションブロック

本項では PLCopen MC 使用に定義されている管理系のファンクションブロックについて説明します。PLCopen 仕様の MC では状態遷移が定義されていますが、管理系のファンクションブロックの多くは状態遷移上の状態に関わらず実行する事が可能になっています。

## MC\_Power 関数

**機能** 運転の可否を制御します。

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 運転可能状態 FALSE: 運転可能状態を解除
Enable_Positive	正転駆動許可	BOOL	TRUE or FALSE	FALSE	未サポート
Enable_Negative	逆転駆動許可	BOOL	TRUE or FALSE	FALSE	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Status	運転可	BOOL	TRUE or FALSE	運転可能状態になったとき TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中のときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

本ファンクションブロックは、Enable（有効）を TRUE にすると、Axis で指定された軸がサーボ ON され運転可能状態となります。

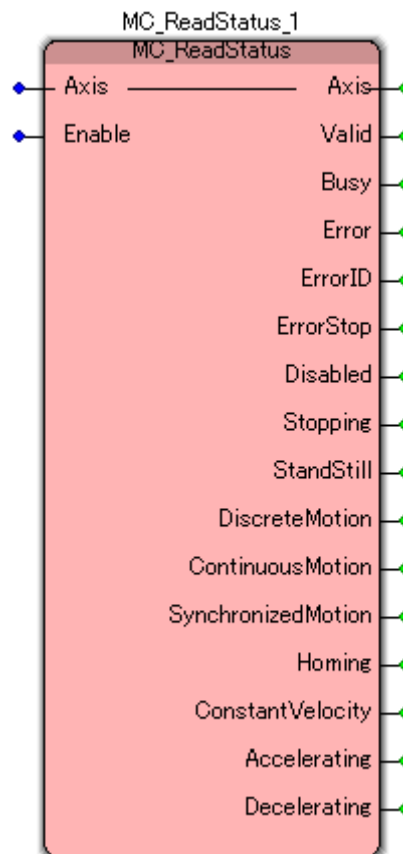
Enable（有効）を FALSE にすると、Axis で指定された軸がサーボ OFF され運転可能状態を解除します。運転可能状態を解除した場合、軸は動作指令を受け付けず、軸制御ができません。

運転可能状態でない軸に対して、動作指令を実行した場合エラーとなります。  
ただし、運転可能解除状態でも、MC\_Power と MC\_Reset は実行可能です。

## MC\_ReadStatus 関数

**機能**

現在処理中のモーション動作に対する軸の詳細ステータスを取得します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し

## 出力

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力
ErrorStop	ErrorStop 状態	BOOL	TRUE or FALSE	現在の軸状態を出力 いずれかの状態のみが TRUE となり、2つ以上が同時に TRUE になることはありません。
Disabled	Disabled 状態	BOOL	TRUE or FALSE	
Stopping	Stopping 状態	BOOL	TRUE or FALSE	
StandStill	StandStill 状態	BOOL	TRUE or FALSE	
Discrete Motion	Discrete Motion 状態	BOOL	TRUE or FALSE	
Continuous Motion	Continuous Motion 状態	BOOL	TRUE or FALSE	
Synchronized Motion	Synchronized Motion 状態	BOOL	TRUE or FALSE	
Homing	Homing 状態	BOOL	TRUE or FALSE	
ConstantVelocity	定速動作中	BOOL	TRUE or FALSE	軸が定速動作中のとき TRUE
Accelerating	加速中	BOOL	TRUE or FALSE	軸が加速中のとき TRUE
Decelerating	減速中	BOOL	TRUE or FALSE	軸が減速中のとき TRUE

※1：エラーコード一覧を参照

## 説明

現在処理中のモーション動作に対する軸の状態遷移ステータスを出力します。  
Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。

ConstantVelocity、Accelerating、Decelerating については MC\_ReadActualVelocity と同じルーチンで、現在速度を読み出し、システム周期毎に現在速度と前回の速度を比較して確認しています。

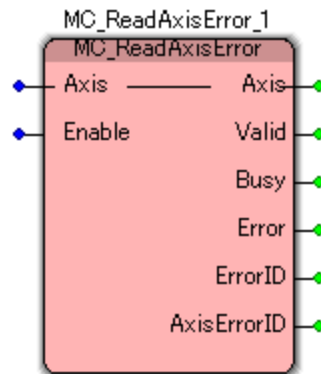
MC\_ReadActualVelocity の読み出し速度が安定していない場合、誤動作する可能性がありますので、位置ループゲイン等のサーボパック固有パラメータの調整をしてください。

## MC\_ReadAxisError 関数

## 機能

一般的な軸エラーを取得します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し

## 出力

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力
AxisErrorID	サーボパック エラー番号	DWORD	※1	サーボパック側でエラーが発生した場合はサーボパックエラーを出力

※1 : エラーコード一覧を参照

## 説明

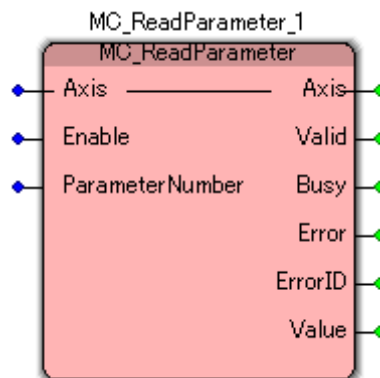
軸のエラーコードを取得します。

Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。



## MC\_ReadParameter 関数

**機能** LREAL 型パラメータを取得します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し
ParameterNumber	パラメータ番号	DWORD	※1	0	読み込みパラメータ番号

※1 : パラメーター一覧を参照

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力
Value	読み出し パラメータ	LREAL	-	読み出した値

※2 : エラーコード一覧を参照

**説明**

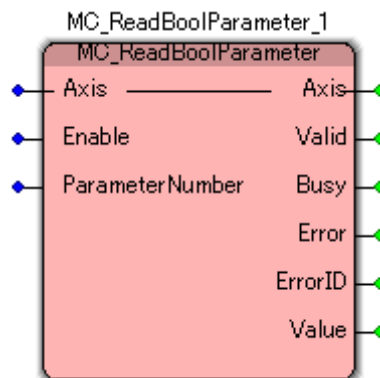
ParameterNumber で指定された、パラメータ (LREAL 型) の値を取得します。  
Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。

## MC\_ReadBoolParameter 関数

## 機能

BOOL 型パラメータを取得します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し
ParameterNumber	パラメータ番号	DWORD	※1	0	読み込みパラメータ番号

※1 : パラメーター一覧を参照

## 出力

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力
Value	読み出し パラメータ	BOOL	TRUE or FALSE	読み出した値

※2 : エラーコード一覧を参照

## 説明

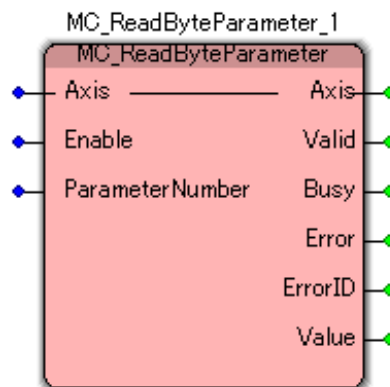
ParameterNumber で指定された、パラメータ (BOOL 型) の値を取得します。  
Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。

## MC\_ReadByteParameter 関数

## 機能

BYTE 型パラメータを取得します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し
ParameterNumber	パラメータ番号	DWORD	※1	0	読み込みパラメータ番号

※1 : パラメータ一覧を参照

## 出力

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力
Value	読み出しパラメータ	BYTE	-	読み出した値

※2 : エラーコード一覧を参照

## 説明

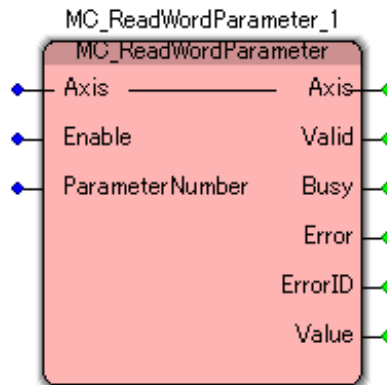
ParameterNumber で指定された、パラメータ (BYTE 型) の値を取得します。  
Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。

## MC\_ReadWordParameter 関数

## 機能

WORD 型パラメータを取得します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し
ParameterNumber	パラメータ番号	DWORD	※1	0	読み込みパラメータ番号

※1 : パラメータ一覧を参照

## 出力

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力
Value	読み出しパラメータ	WORD	-	読み出した値

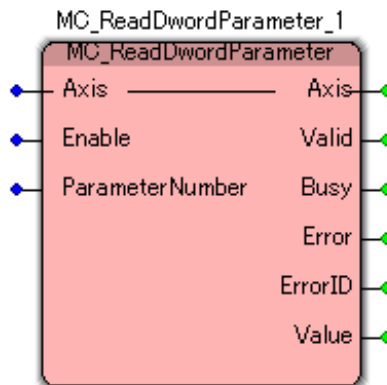
※2 : エラーコード一覧を参照

## 説明

ParameterNumber で指定された、パラメータ (WORD 型) の値を取得します。  
Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。

## MC\_ReadDwordParameter 関数

**機能**      DWORD 型パラメータを取得します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し
ParameterNumber	パラメータ番号	DWORD	※1	0	読み込みパラメータ番号

※1 : パラメータ一覧を参照

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力
Value	読み出しパラメータ	DWORD	-	読み出した値

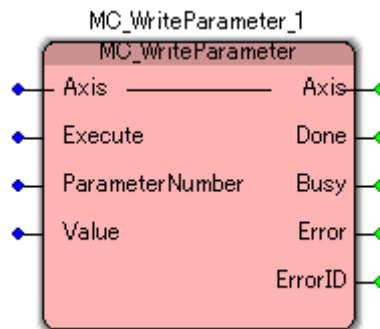
※2 : エラーコード一覧を参照

**説明**

ParameterNumber で指定された、パラメータ (DWORD 型) の値を取得します。  
Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。

## MC\_WriteParameter 関数

**機能** LREAL 型パラメータを書き込みます。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
ParameterNumber	パラメータ番号	DWORD	※1	0	書き込みパラメータ番号
Value	設定値	LREAL	-	0	書き込みパラメータ値

※1：パラメータ一覧を参照

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力

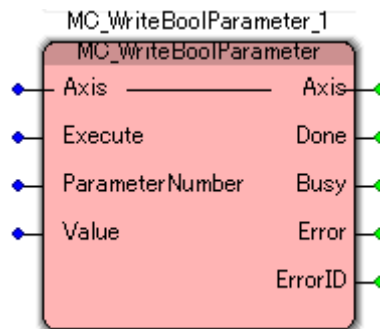
※2：エラーコード一覧を参照

**説明**

ParameterNumber で指定された、パラメータ (LREAL 型) の値を、Execute の立ち上がりエッジで書き込みます。

## MC\_WriteBoolParameter 関数

**機能** BOOL 型パラメータを書き込みます。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
ParameterNumber	パラメータ番号	DWORD	※1	0	書き込みパラメータ番号
Value	設定値	BOOL	TRUE or FALSE	0	書き込みパラメータ値

※1：パラメータ一覧を参照

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力

※2：エラーコード一覧を参照

**説明**

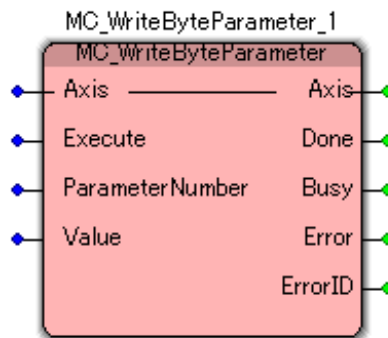
ParameterNumber で指定された、パラメータ (BOOL 型) の値を、Execute の立ち上がりエッジで書き込みます。

## MC\_WriteByteParameter 関数

## 機能

BYTE 型パラメータを書き込みます。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
ParameterNumber	パラメータ番号	DWORD	※1	0	書き込みパラメータ番号
Value	設定値	BYTE	-	0	書き込みパラメータ値

※1：パラメータ一覧を参照

## 出力

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力

※2：エラーコード一覧を参照

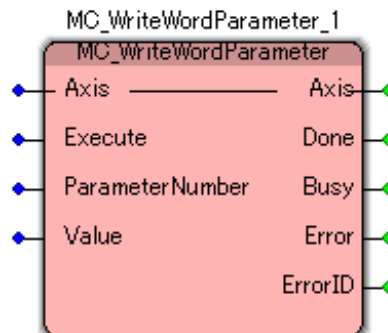
## 説明

ParameterNumber で指定された、パラメータ (BYTE 型) の値を、Execute の立ち上がりエッジで書き込みます。



## MC\_WriteWordParameter 関数

**機能** WORD 型パラメータを書き込みます。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
ParameterNumber	パラメータ番号	DWORD	※1	0	書き込みパラメータ番号
Value	設定値	WORD	-	0	書き込みパラメータ値

※1：パラメータ一覧を参照

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力

※2：エラーコード一覧を参照

**説明**

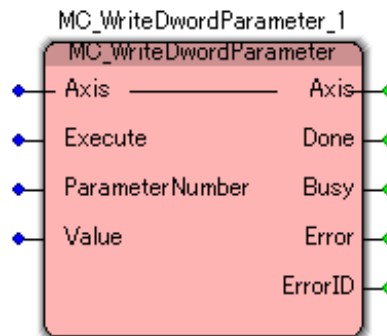
ParameterNumber で指定された、パラメータ（WORD 型）の値を、Execute の立ち上がりエッジで書き込みます。

## MC\_WriteDwordParameter 関数

## 機能

DWORD 型パラメータを書き込みます。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
ParameterNumber	パラメータ番号	DWORD	※1	0	書き込みパラメータ番号
Value	設定値	DWORD	-	0	書き込みパラメータ値

※1 : パラメータ一覧を参照

## 出力

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※2	異常が発生したときのエラーコードを出力

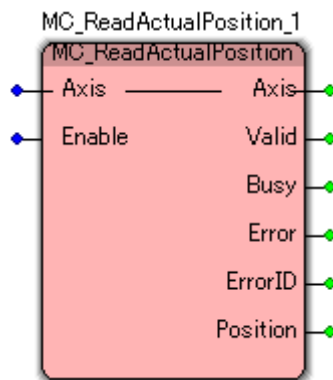
※2 : エラーコード一覧を参照

## 説明

ParameterNumber で指定された、パラメータ (DWORD 型) の値を、Execute の立ち上がりエッジで書き込みます。

## MC\_ReadActualPosition 関数

**機能** 現在位置を取得します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力
Position	現在位置	LREAL	倍精度実数値 【指令単位】	現在位置を絶対座標で出力

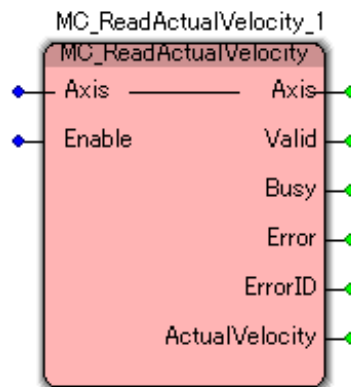
※1 : エラーコード一覧を参照

**説明**

軸の現在位置を絶対座標で取得します。  
Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。

## MC\_ReadActualVelocity 関数

**機能** 現在速度を取得します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力
ActualVelocity	現在速度	LREAL	倍精度実数値 【指令単位/s】	現在速度を出力

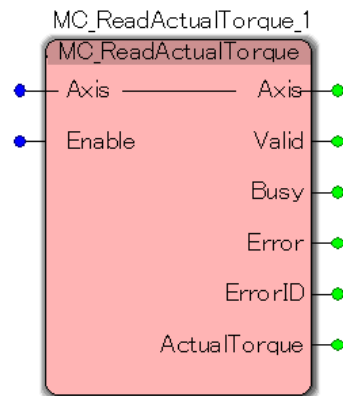
※1 : エラーコード一覧を参照

**説明**

軸の現在速度を取得します。正の値なら正転、負の値なら逆転、0なら停止中となります。Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。

## MC\_ReadActualTorque 関数

**機能** 現在トルクを取得します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 継続読み出し FALSE: 実行無し

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	以下の出力値が有効の場合 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力
ActualTorque	現在トルク	LREAL	倍精度実数値 【N.m】 or 【%】	現在トルクを出力 単位はサーボパックにより異なる。<※2>

※1 : エラーコード一覧を参照

※2 : MECHATROLINK-III の場合 : サーボパラメータ「トルク単位選択 (0x47)」の値により変わります。

EtherCAT の場合 : サーボパックマニュアルの CiA402 パラメータ「目標トルク (0x6071)」または、「内部指令トルク (0x6074)」の単位を参照してください。

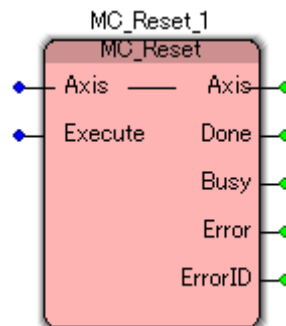
**説明**

軸の現在トルクを取得します。正の値なら正転、負の値なら逆転、0なら停止中となります。Enable が TRUE の間、ソフトウェア PLC のタスク周期で値を更新します。



## MC\_Reset 関数

**機能** 指定した軸に関するエラーをリセットします。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	リセット完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

軸でエラーが発生し、ErrorStop 状態に移行したとき、本FBを実行することで、StandStill 状態へ復帰します。

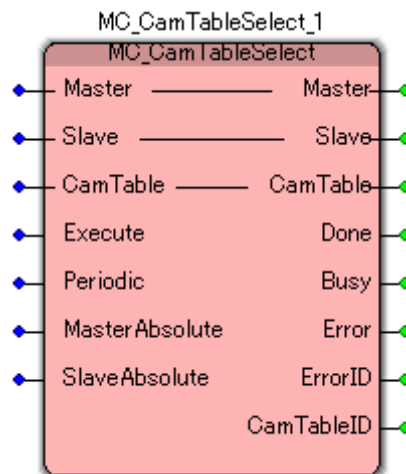
サーボパックで発生したエラーについては、エラーリセット処理が実行されます。

通信異常等の外的要因で発生したエラーについては、解除できない場合があります。エラー内容からエラー要因を取り除いた上で実行してください。

## MC\_CamTableSelect 関数

**機能** 未サポート

**書式**



**説明** 未サポート



3-3-2 PLCopen仕様 動作ファンクションブロック

本項では PLCopen MC 使用に定義されている動作系のファンクションブロックについて説明します。本項で説明しますファンクションブロックは、PLCopen に定義されている状態遷移に従い動作を行います（図 3-3-2-1 参照）。

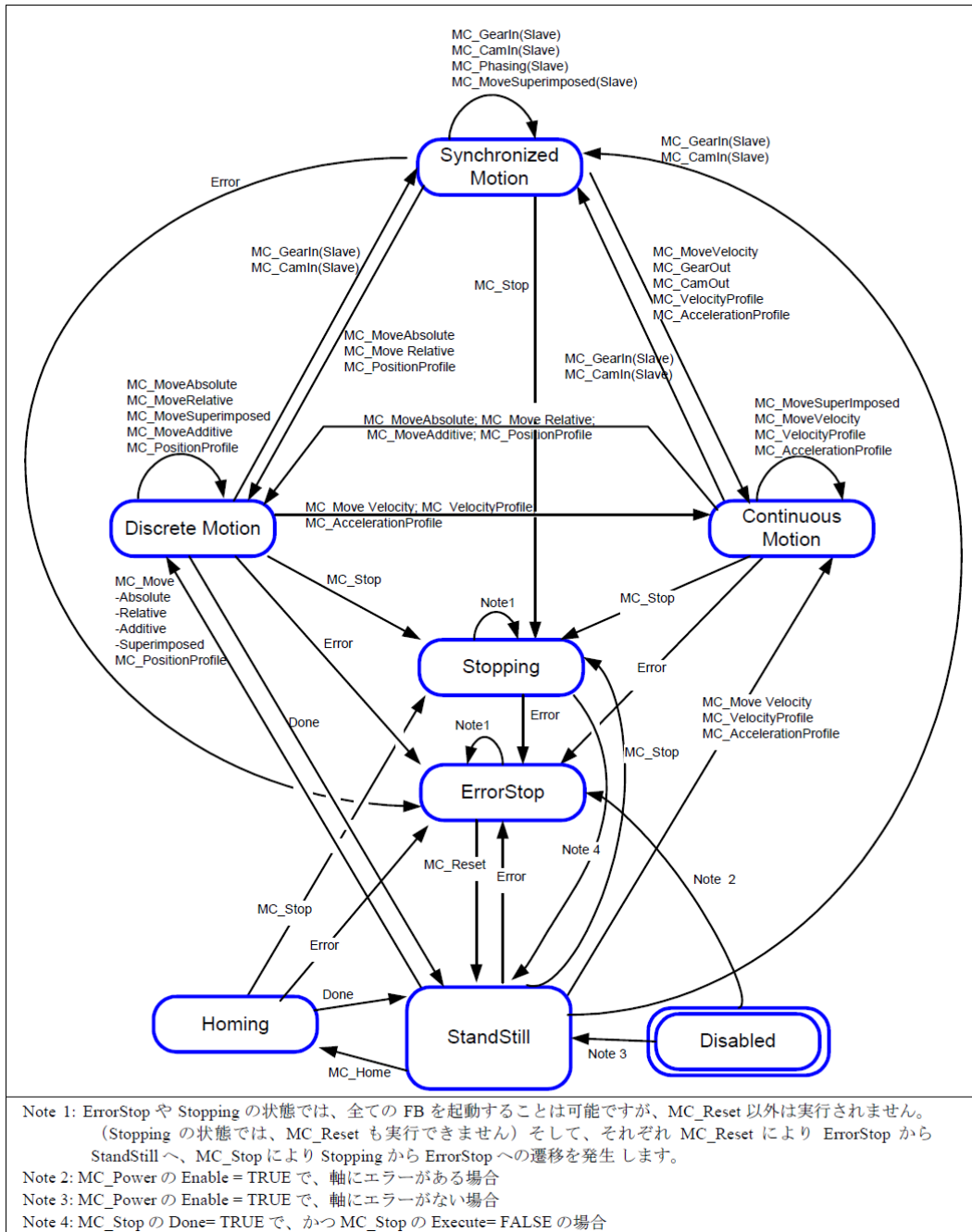


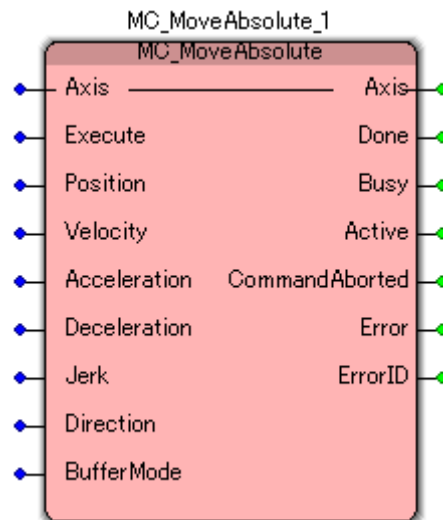
図 3-3-2-1. PLCopenMC 状態遷移図

## MC\_MoveAbsolute 関数

## 機能

絶対位置による位置決めを実行します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Position	目標位置	LREAL	倍精度実数値 【指令単位】	0	位置決め目標位置を絶対位置で指定
Velocity	移動速度	LREAL	0, 倍精度実数値正数 【指令単位/s】	0	位置決め時の速度
Acceleration	加速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	位置決め時の加速度
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	位置決め時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
Direction	動作方向	MC_Direction	-	-	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定

型名	型	値	モード	説明
MC_Direction	UINT16	—	—	—
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	位置決め完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

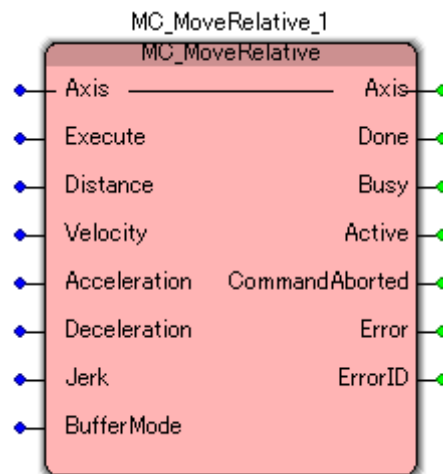
絶対位置による位置決めを実行します。位置決め完了は、目標位置に対して設定された位置決め完了幅の範囲に到達する事で完了します。

## MC\_MoveRelative 関数

## 機能

相対位置による位置決めを実行します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Distance	移動量	LREAL	倍精度実数値 【指令単位】	0	位置決め目標位置を相対位置で指定
Velocity	移動速度	LREAL	0, 倍精度実数値正数 【指令単位/s】	0	位置決め時の速度
Acceleration	加速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	位置決め時の加速度
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	位置決め時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	位置決め完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

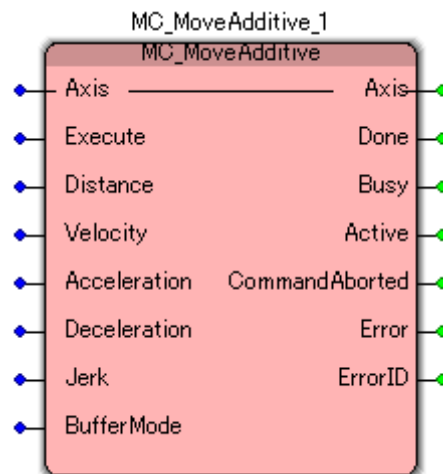
相対位置による位置決めを実行します。位置決め完了は、目標位置に対して機器に設定された位置決め完了幅の範囲に到達する事で完了します。

## MC\_MoveAdditive 関数

## 機能

直前に実行された位置決めに対して、本ファンクションブロックで指定した相対位置を加算した位置に対して位置決めを実行します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Distance	加算移動量	LREAL	倍精度実数値 【指令単位】	0	加算移動量を相対位置で指定
Velocity	移動速度	LREAL	0, 倍精度実数値正数 【指令単位/s】	0	位置決め時の速度
Acceleration	加速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	位置決め時の加速度
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	位置決め時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	位置決め完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

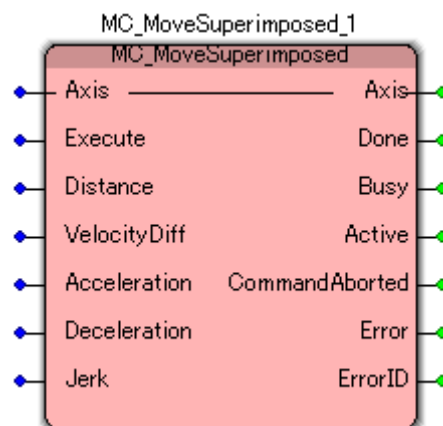
**説明**

直前のコマンドによる目標位置に、指定された相対位置を付加して移動します。位置決め完了は、目標位置に対して機器に設定された位置決め完了幅の範囲に到達する事で完了します。本ファンクションブロックを単体で実行した場合の動作は、MC\_MoveRelative と同等です。

## MC\_MoveSuperimposed 関数

**機能** 未サポート

**書式**

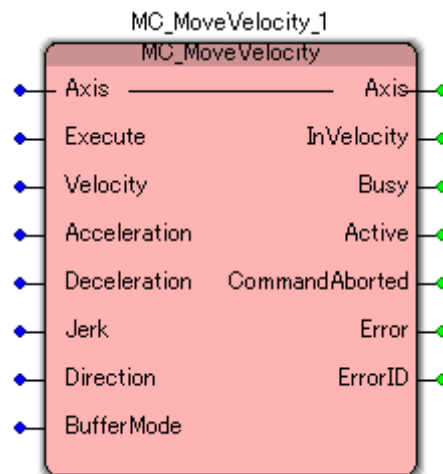


**説明** 未サポート



## MC\_MoveVelocity 関数

**機能** 指定速度による定速駆動を実行します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Velocity	指令速度	LREAL	0, 倍精度実数値正数 【指令単位/s】	0	速度制御時の動作速度
Acceleration	加速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	速度制御時の加速度
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	速度制御時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
Direction	動作方向	MC_Direction	0~3	0	動作方向を指定
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定

型名	型	値	モード	説明
MC_Direction	UINT16	0	正方向	正転方向に動作
		1	近回り	本 FB では常に正方向に動作
		2	逆方向	逆転方向に動作
		3	現在の方向	動作中の場合、同方向に動作。 停止中の場合は正転方向へ動作
MC_BufferMode	UINT16	0	Aborting	実行中の FB に対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
InVelocity	速度到達通知	BOOL	TRUE or FALSE	指令速度に到達で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

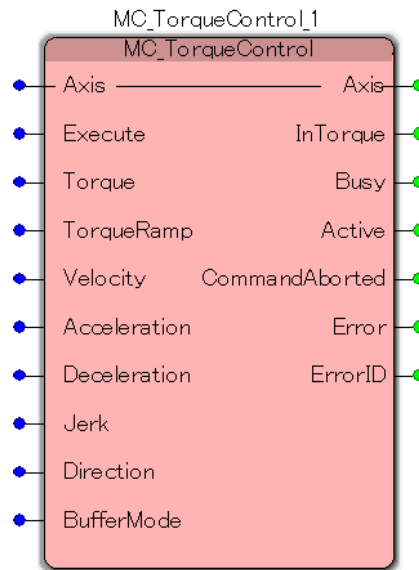
**説明**

指定された速度での永久動作を命令します。

本ファンクションブロックによる動作を停止させるには、別のファンクションブロックによる指令を行う必要があります。

## MC\_TorqueControl 関数

**機能** 指定トルクによるトルク制御を実行します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Torque	指令トルク	LREAL	0, 倍精度実数値正数 【N.m】 or 【%】	0	トルク制御時の動作トルク 単位はサーボパックにより異なる。〈※1〉
TorqueRamp	トルク傾斜度	LREAL	-	-	未サポート
Velocity	指令速度	LREAL	-	-	未サポート
Acceleration	加速度	LREAL	-	-	未サポート
Deceleration	減速度	LREAL	-	-	未サポート
Jerk	加加速度	LREAL	-	-	未サポート
Direction	動作方向	MC_Direction	0~3	0	動作方向を指定
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定

※1 : MECHATROLINK-IIIの場合 : サーボパラメータ「トルク単位選択 (0x47)」の値により変わります。  
EtherCAT の場合 : サーボパックマニュアルの CiA402 パラメータ「目標トルク (0x6071)」または、「内部指令トルク (0x6074)」の単位を参照してください。

型名	型	値	モード	説明
MC_Direction	UINT16	0	正方向	正転方向に動作
		1	近回り	本FBでは常に正方向に動作
		2	逆方向	逆転方向に動作
		3	現在の方向	動作中の場合、同方向に動作。 停止中の場合は正転方向へ動作
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
InTorque	トルク到達通知	BOOL	TRUE or FALSE	指令トルクに到達で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

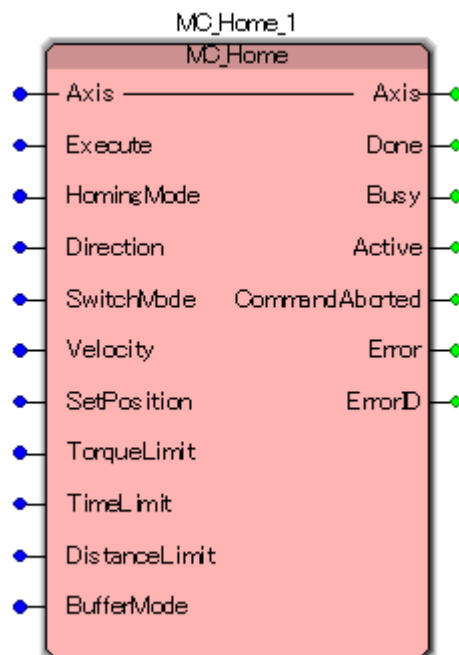
指定されたトルクでの永久動作を命令します。  
本ファンクションブロックによる動作を停止させるには、別のファンクションブロックによる指令を行う必要があります。

## MC\_Home 関数

## 機能

原点サーチシーケンスを実行します。本ファンクションブロックの機能は「3-3-3 PLCopen仕様 原点サーチファンクションブロック」に分割して実装しています。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
HomingMode	原点復帰モード	MC_HomingMode	0~3	0	原点復帰モード
Direction	動作方向	MC_HomeDir	0~3	0	動作方向を指定
SwitchMode	センサモード	MC_SwitchMode	0~5	0	原点復帰完了センサモード
Velocity	移動速度	LREAL	0, 倍精度実数値正数 【指令単位/s】	0	位置決め時の速度
SetPosition	原点更新位置	LREAL	倍精度実数値 【指令単位】	0	原点サーチ完了後に更新される原点オフセット値
TorqueLimit	トルクリミット	LREAL	-	-	未サポート
TimeLimit	タイマリミット	LREAL	0~655 【s】	0	原点復帰タイムアウト時間 0の場合は無制限

DistanceLimit	移動量リミット	LREAL	0, 倍精度実数値正数 【指令単位】	0	移動量のリミット値 0の場合は無制限
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみ サポート</b>

型名	型	値	モード	説明
MC_HomingMode	UINT16	0	StepAbsSwのみ	StepAbsSwを実行して原点復帰完了
		1	StepAbsSw + StepRefPulse	StepAbsSwを実行後、StepRefPulseを実行して原点復帰完了
		2	StepLimitSwのみ	StepLimitSwを実行して原点復帰完了
		3	StepLimitSw + StepRefPulse	StepLimitSwを実行後、StepRefPulseを実行して原点復帰完了
MC_HomeDir	UINT16	0	MC_Positive	正方向へ開始 (未サポート)
		1	MC_Negative	負方向へ開始 (未サポート)
		2	MC_SwitchPositive	MC_HomingMode=0or1 実行開始時に原点信号がONしていたら負方向へ開始、OFFしていたら正方向へ開始 MC_HomingMode=2or3 動作方向を指定
		3	MC_SwitchNegative	MC_HomingMode=0or1 実行開始時に原点信号がONしていたら正方向へ開始、OFFしていたら負方向へ開始 MC_HomingMode=2or3 動作方向を指定
MC_SwitchMode	UINT16	0	MC_On	センサがON状態なら原点復帰完了 (未サポート)
		1	MC_Off	センサがOFF状態なら原点復帰完了 (未サポート)
		2	MC_EdgeOn	センサがOFF→ONのエッジで原点復帰完了
		3	MC_EdgeOff	センサがON→OFFのエッジで原点復帰完了
		4	MC_EdgeSwitchPositive	動作方向により原点復帰完了のエッジが変わる (未サポート)
		5	MC_EdgeSwitchNegative	動作方向により原点復帰完了のエッジが変わる (未サポート)
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

※ 注：MC\_EdgeOffについては、サーボバックが原点信号OFFエッジでの原点復帰を対応していないと使用できません。

※ 注：MC\_HomingMode=2or3を実行するときは、MC\_EdgeOFFのみ有効となります。

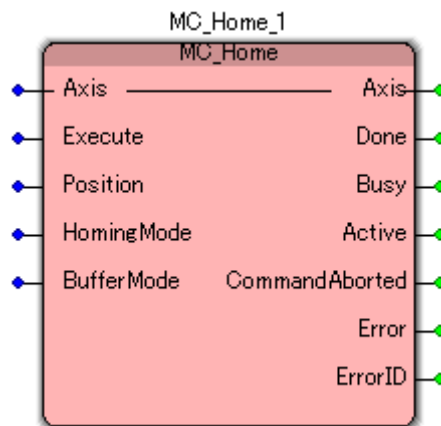
**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	原点復帰完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

PLCopenMC の Part1 に定義されている本来の FB は下記のパラメータのみとなっています。速度や原点信号のタイプ等については、FB とは別に設定する必要があります。



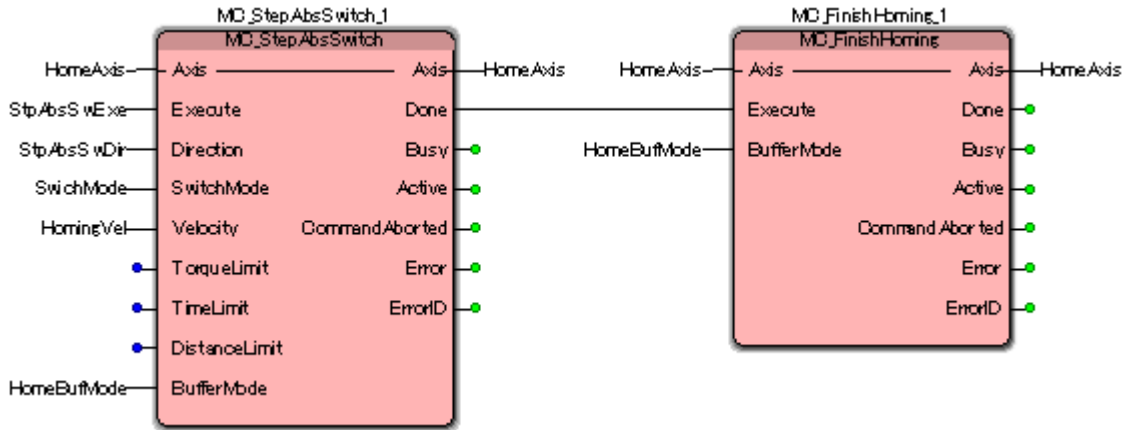
Part1 で定義されている原点サーチ仕様では、定義が曖昧なため、PLCopenMC は Part5 で詳細な原点サーチ FB を追加定義しました。

Part5 で定義されている原点サーチ FB を組み合わせて、目的に添った原点サーチシーケンスを組み込むことができます。

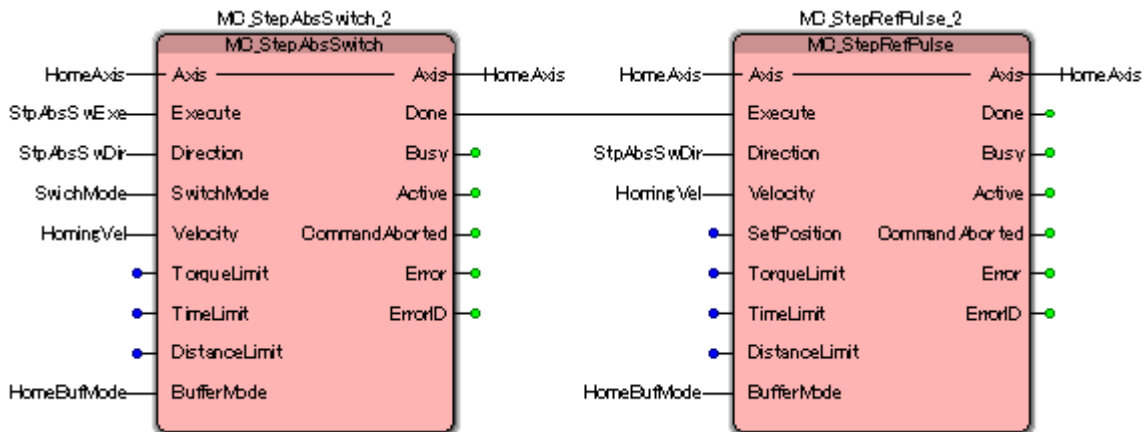
本 FB では、本来の MC\_HOME 仕様をやめ、Part5 で定義されている MC\_StepAbsSw と MC\_StepLimitSw、MC\_StepRefPulse を連続して実行できるように4つの HomingMode を定義しました。

HomingMode 毎の動作は下図のFB と同義です。

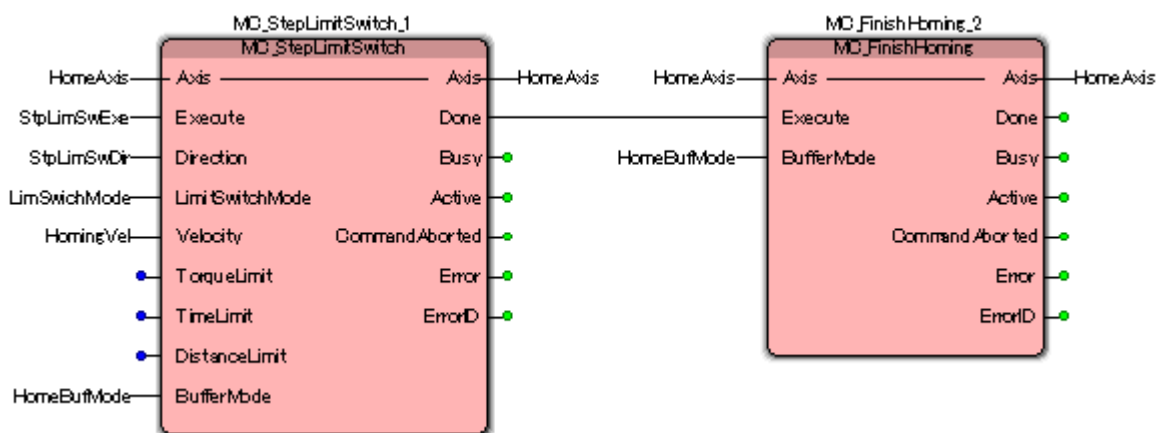
MC\_HomingMode=0



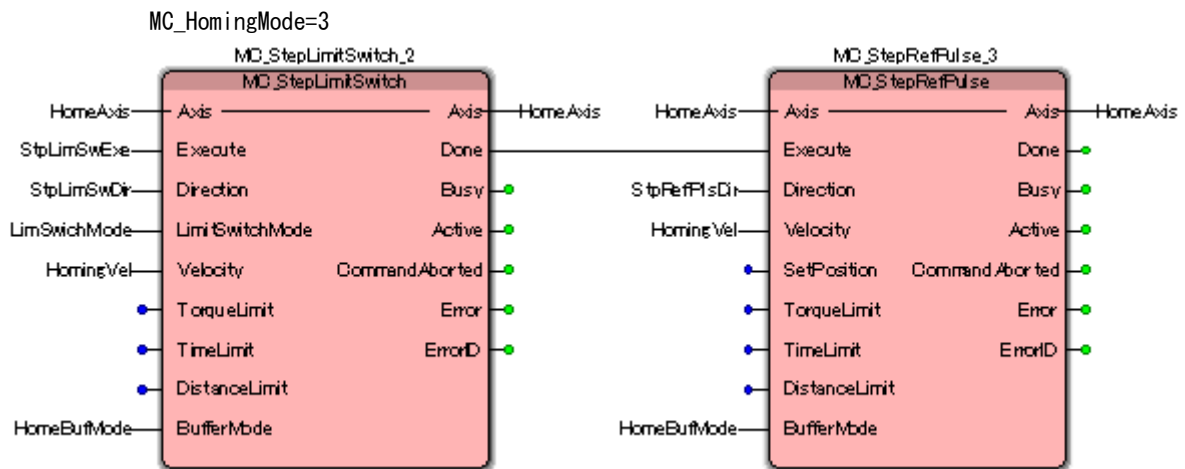
MC\_HomingMode=1



MC\_HomingMode=2



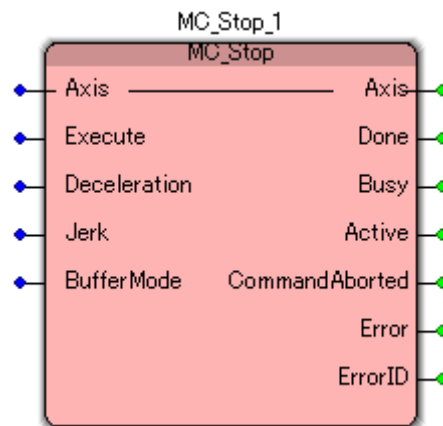




※注：MC\_HomingMode=3 のときは、StepRefPulse の動作方向は、StepLimitSwitch で指定した動作方向と逆になります。

## MC\_Stop 関数

**機能** 位置決め実行中の軸動作を停止させます。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	停止時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	ゼロ速度到達で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

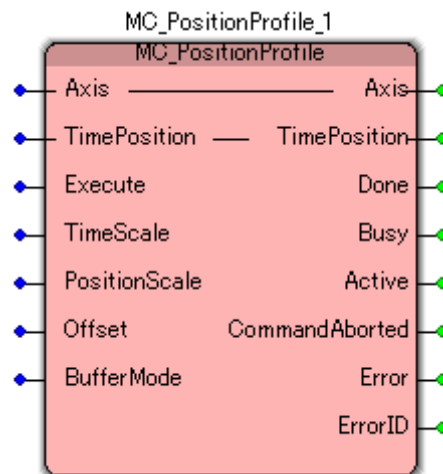
**説明**

軸の制御動作を停止させ、Stopping 状態に遷移します。軸停止後、Done 出力がセットされますが、Execute 入力が TRUE の間は Stopping 状態のままになります。Done 出力セット後に Execute 入力が FALSE になる事で StandStill 状態に遷移します。

## MC\_PositionProfile 関数

**機能** 未サポート

**書式**

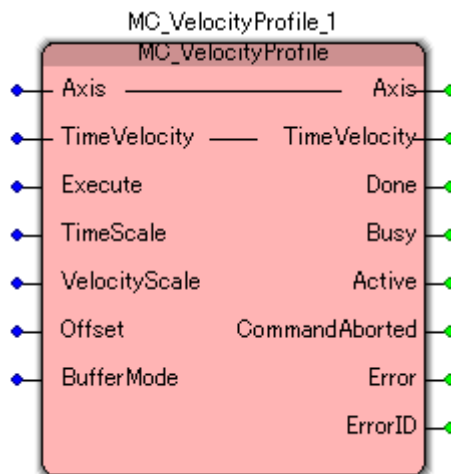


**説明** 未サポート

## MC\_VelocityProfile 関数

**機能** 未サポート

**書式**

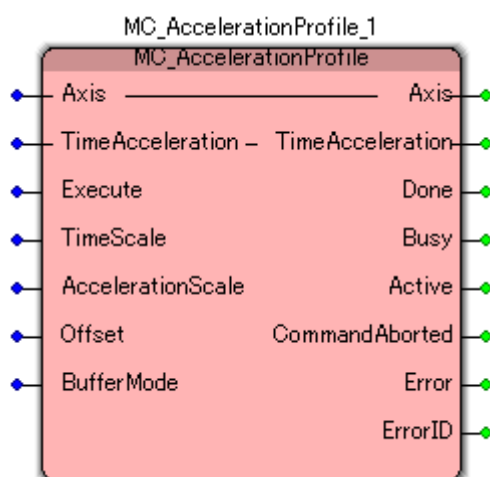


**説明** 未サポート

## MC\_AccelerationProfile 関数

**機能**

未サポート

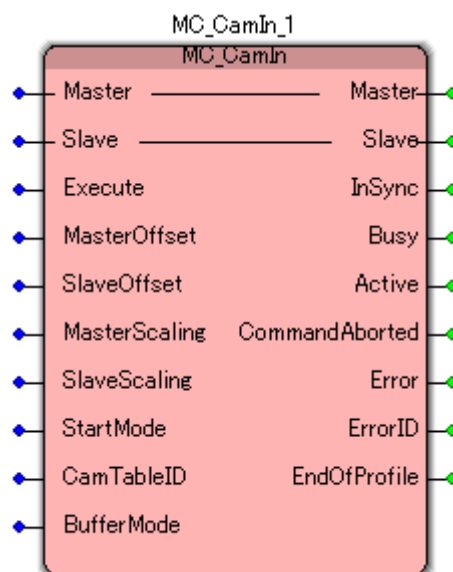
**書式****説明**

未サポート

## MC\_CamIn 関数

**機能** 未サポート

**書式**

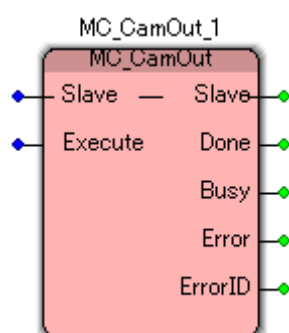


**説明** 未サポート

## MC\_CamOut 関数

**機能** 未サポート

**書式**



**説明** 未サポート

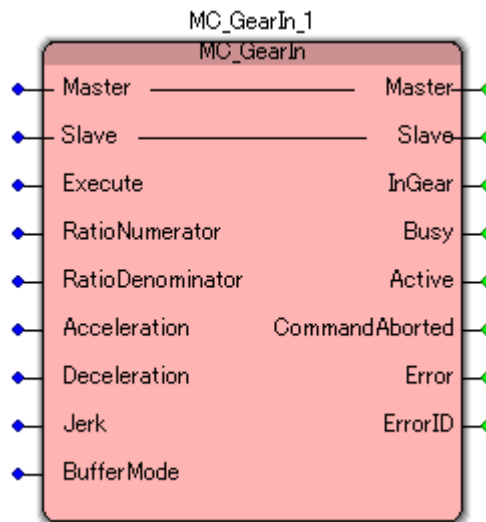


## MC\_GearIn 関数

## 機能

マスタ軸とスレーブ軸間の連動比率を設定します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Master	マスタ軸	AXIS_REF (UINT)	1~62	-	マスタとなる論理軸番号を指定
Slave	スレーブ軸	AXIS_REF (UINT)	1~62	-	スレーブとなる論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
RatioNumerator	ギア比 (分子)	INT	1	1	本FBでは1固定
RatioDenominator	ギア比 (分母)	INT	1	1	本FBでは1固定
Acceleration	加速度	LREAL	-	-	未サポート
Deceleration	減速度	LREAL	-	-	未サポート
Jerk	加加速度	LREAL	-	-	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
InGear	同期比率到達	BOOL	TRUE or FALSE	設定した比率に到達したとき TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

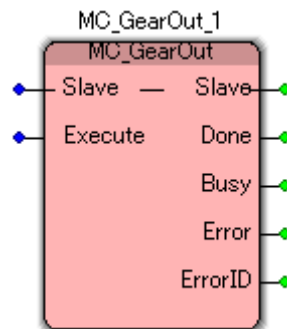
指定したスレーブ軸を指定したマスタ軸に連動させます。スレーブ軸は SynchronizedMotion 状態に遷移され、マスタ軸が動作すると連動してスレーブ軸も動作します。

本FBの制約として下記の3点があります。

- ① マスタ軸とスレーブ軸が停止しているときのみ実行可能です。
- ② 設定できる比率は 1:1 のみです。
- ③ GearOut が実行されるまで、他の動作コマンドは受け付けません。

## MC\_GearOut 関数

**機能** マスタ軸とスレーブ軸の連動を解除します。

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
Slave	スレーブ軸	AXIS_REF (UINT)	1~62	-	スレーブとなる論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	連動解除で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

マスタ軸とスレーブ軸の連動動作を解除します。マスタ軸とスレーブ軸が停止している状態でのみ実行可能です。

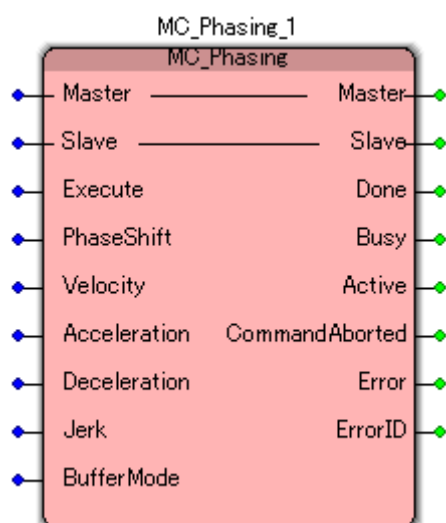
本FBの制約として下記の2点があります。

- ① マスタ軸とスレーブ軸が停止しているときのみ実行可能です。
- ② 実行完了後、スレーブの状態は StandStill 状態へ遷移します。

## MC\_Phasing 関数

**機能** 未サポート

**書式**



**説明** 未サポート

### 3-3-3 PLCopen 仕様 原点サーチファンクションブロック

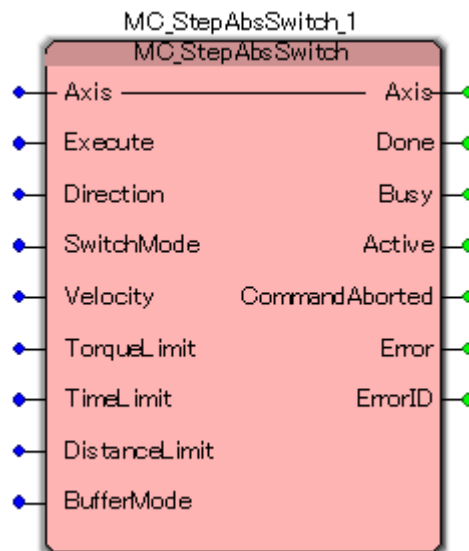
本項では PLCopen MC 使用に定義されている原点サーチ系のファンクションブロックについて説明します。本項で説明しますファンクションブロックは、PLCopen の技術仕様書「Technical Paper PLCopen Technical Committee 2 – Task Force Function Blocks for motion control Part 5 – Homing」に定義されている仕様に従ったファンクションブロックになっています。これらのファンクションブロックは MC\_Home に相当する機能を個別のファンクションブロックとして定義されたものになっており、実行時には Homing 状態に状態遷移します。

## MC\_StepAbsSwitch 関数

## 機能

機械的に設置されたリミット SW、原点 SW を使用する事で原点サーチを実行します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Direction	動作方向	MC_HomeDir	0~3	0	動作方向を指定
SwitchMode	センサモード	MC_SwitchMode	0~5	0	原点復帰完了センサモード
Velocity	移動速度	LREAL	0, 倍精度実数値正数 【指令単位/s】	0	位置決め時の速度
TorqueLimit	トルクリミット	LREAL	-	-	未サポート
TimeLimit	タイマリミット	LREAL	0~655 【s】	0	原点復帰タイムアウト時間 0の場合は無制限
DistanceLimit	移動量リミット	LREAL	0, 倍精度実数値正数 【指令単位】	0	移動量のリミット値 0の場合は無制限
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本 FB では、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_HomeDir	UINT16	0	MC_Positive	正方向へ開始 (未サポート)
		1	MC_Negative	負方向へ開始 (未サポート)
		2	MC_SwitchPositive	実行開始時に原点信号が ON していたら負方向へ開始、OFF していたら正方向へ開始
		3	MC_SwitchNegative	実行開始時に原点信号が ON していたら正方向へ開始、OFF していたら負方向へ開始
MC_SwitchMode	UINT16	0	MC_On	センサが ON 状態なら原点復帰完了 (未サポート)
		1	MC_Off	センサが OFF 状態なら原点復帰完了 (未サポート)
		2	MC_EdgeOn	センサが OFF→ON のエッジで原点復帰完了
		3	MC_EdgeOff	センサが ON→OFF のエッジで原点復帰完了
		4	MC_EdgeSwitchPositive	動作方向により原点復帰完了のエッジが変わる (未サポート)
		5	MC_EdgeSwitchNegative	動作方向により原点復帰完了のエッジが変わる (未サポート)
MC_BufferMode	UINT16	0	Aborting	実行中の FB に対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

※注：MC\_EdgeOff については、サーボパックが OFF エッジでの原点復帰に対応していないと使えません。

### 出力

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	原点復帰完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

### 説明

原点サーチが実行可能な状態のとき、Execute 入力の立ち上がりエッジにより原点サーチを開始し、状態を Homing 状態に遷移します。本 FB では、停止位置の位置情報を更新しません。本 FB は、正常終了時、StandStill 状態へ遷移しません。MC\_FinishHoming を実行し、StandStill 状態へ遷移させてください。

異常終了時は ErrorStop 状態へ遷移します。

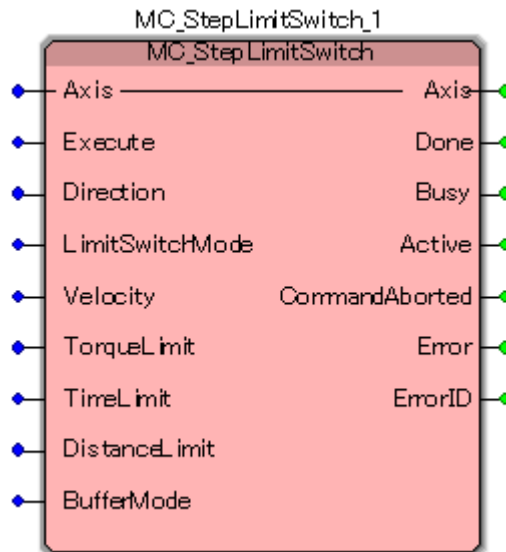
動作方向	センサモード	動作内容
MC_SwitchPositive	MC_EdgeOn	
	MC_EdgeOff	
MC_SwitchNegative	MC_EdgeOn	
	MC_EdgeOff	



MC\_StepLimitSwitch 関数

**機能** 機械的に設置されたリミット SW を使用して原点サーチを行います。

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Direction	動作方向	MC_HomeDir	0~1	0	動作方向を指定
LimitSwitchMode	センサモード	MC_SwitchMode	0~3	0	原点復帰完了センサモード
Velocity	移動速度	LREAL	0, 倍精度実数値正数 【指令単位/s】	0	位置決め時の速度
TorqueLimit	トルクリミット	LREAL	-	-	未サポート
TimeLimit	タイマリミット	LREAL	0~655 【s】	0	原点復帰タイムアウト時間 0の場合は無制限
DistanceLimit	移動量リミット	LREAL	0, 倍精度実数値正数 【指令単位】	0	移動量のリミット値 0の場合は無制限
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_HomeDir	UINT16	0	MC_Positive	正方向へ開始
		1	MC_Negative	負方向へ開始
MC_SwitchMode	UINT16	0	MC_On	センサが ON 状態なら原点復帰完了 (未サポート)
		1	MC_Off	センサが OFF 状態なら原点復帰完了 (未サポート)
		2	MC_EdgeOn	センサが OFF→ON のエッジで原点復帰完了 (未サポート)
		3	MC_EdgeOff	センサが ON→OFF のエッジで原点復帰完了
MC_BufferMode	UINT16	0	Aborting	実行中の FB に対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	原点復帰完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

原点サーチが実行可能な状態のとき、Execute 入力の立ち上がりエッジにより原点サーチを開始し、状態を Homing 状態に遷移します。本ファンクションブロックでは、停止位置の位置情報を更新しません。本FBは、正常終了時、StandStill 状態へ遷移しません。MC\_FinishHoming を実行し、StandStill 状態へ遷移させてください。

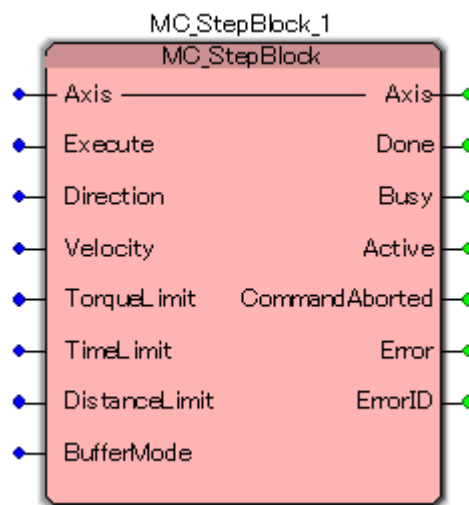
異常終了時は ErrorStop 状態へ遷移します。

動作方向	センサモード	動作内容
MC_Positive	MC_EdgeOff	
MC_Negative	MC_EdgeOff	

## MC\_StepBlock 関数

**機能** 未サポート

**書式**



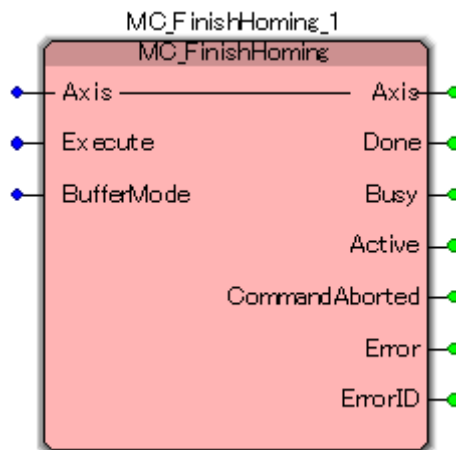
**説明** 未サポート

## MC\_FinishHoming 関数

## 機能

指定された軸の状態を Homing 状態から StandStill 状態に遷移させます。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本 FB では、0:Aborting のみサポート</b>

## 出力

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	原点復帰完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**説明**

指定された軸の状態を Homing 状態から StandStill 状態に遷移させます。  
軸は動作しません。

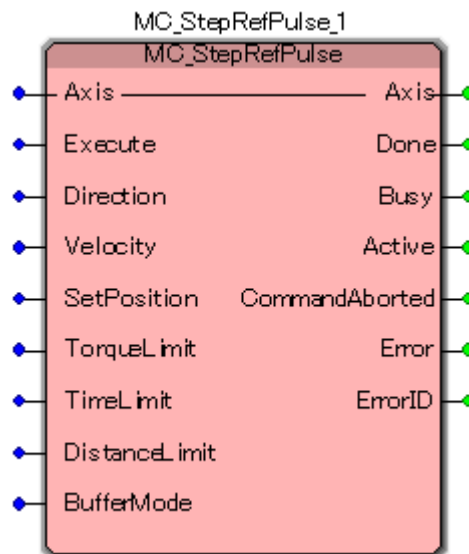
MC\_StepAbsSwitch と MC\_StepLimitSwitch を実行した後に、本FBを実行します。

## MC\_StepRefPulse 関数

## 機能

エンコーダからの Zero パルス（マーカ、またはリファレンスパルスとも呼びます。）を参照しながら原点サーチを行います。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
Direction	動作方向	MC_HomeDir	0~1	0	動作方向を指定
Velocity	移動速度	LREAL	0, 倍精度実数値正数 【指令単位/s】	0	位置決め時の速度
SetPosition	原点更新位置	LREAL	倍精度実数値 【指令単位】	0	原点サーチ完了後に更新される原点オフセット値
TorqueLimit	トルクリミット	LREAL	-	-	未サポート
TimeLimit	タイマリミット	LREAL	0~655 【s】	0	原点復帰タイムアウト時間 0の場合は無制限
DistanceLimit	移動量リミット	LREAL	0, 倍精度実数値正数 【指令単位】	0	移動量のリミット値 0の場合は無制限
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_HomeDir	UINT16	0	MC_Positive	正方向へ開始
		1	MC_Negative	負方向へ開始
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	原点復帰完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

エンコーダからの Zero パルス（マーカ、またはリファレンスパルスとも呼びます。）を参照しながら原点サーチを行います。

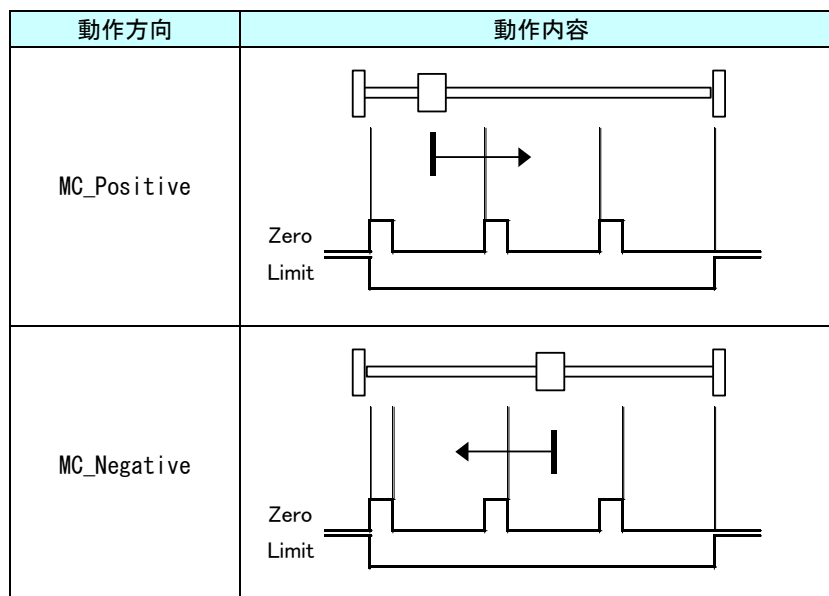
リファレンスパルスはエンコーダ1回転に1度発生します。

原点サーチにリファレンスパルスを使用する利点は伝統的な光学センサや磁気センサよりも高い精度を出す事が出来ることです。

実行開始時、状態が Homing 状態ではなければ Homing 状態に遷移します。

原点復帰完了位置を SetPosition で指定された位置に更新します。

正常終了時、Homing 状態から StandStill 状態に遷移します。



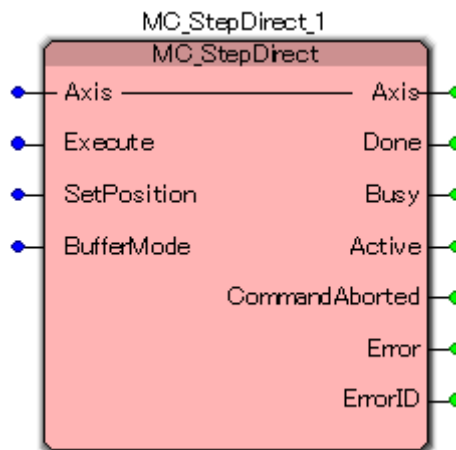


## MC\_StepDirect 関数

## 機能

SetPosition 入力値を現在停止位置にセットすることで原点サーチを完了します。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
SetPosition	原点更新位置	LREAL	倍精度実数値 【指令単位】	0	原点サーチ完了後に更新される 原点オフセット値
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみ サポート</b>

## 出力

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	原点復帰完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**説明**

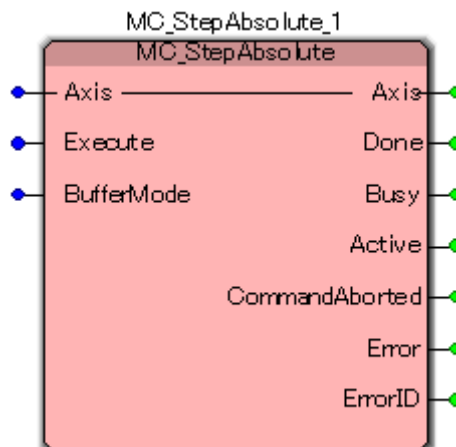
SetPosition 入力値を現在停止位置にセットすることで原点サーチを完了します。  
本ファンクションブロックでは軸は動作しません。  
正常終了時、Homing 状態から StandStill 状態に遷移します。

## MC\_StepAbsolute 関数

## 機能

アブソリュートエンコーダに現在位置を原点位置としてセットします。

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	立ち上がり時に命令を実行
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

## 出力

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	原点復帰完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-1 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**説明**

アブソリュートエンコーダに現在位置を原点位置としてセットします。

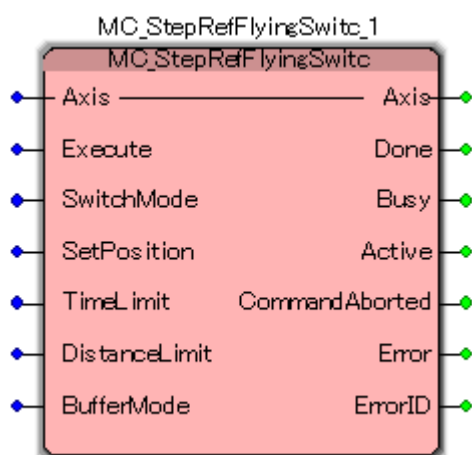
軸動作は行われません。

正常終了時、Homing 状態から StandStill 状態に遷移します。

## MC\_StepReferenceFlyingSwitch 関数

**機能** 未サポート

**書式**

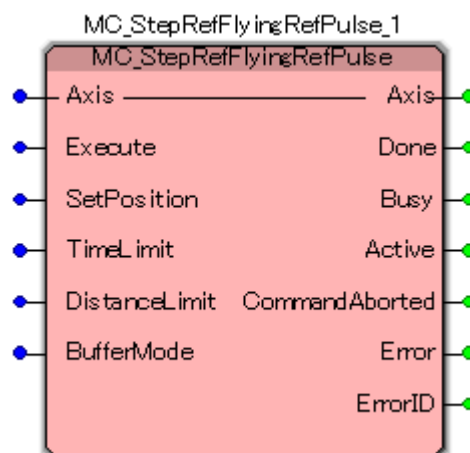


**説明** 未サポート

## MC\_StepFeferenceFlyingRefPulse 関数

**機能** 未サポート

**書式**

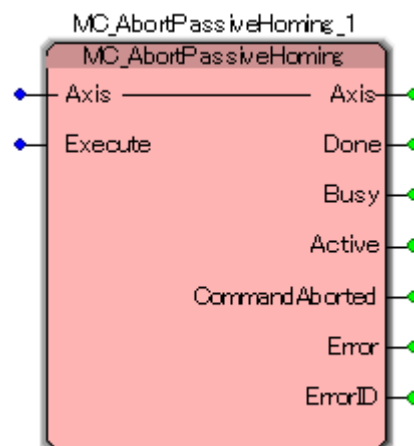


**説明** 未サポート

## MC\_AbortPassiveHoming 関数

**機能** 未サポート

**書式**



**説明** 未サポート

#### 3-3-4 PLCopen仕様 MC Part4 管理ファンクションブロック

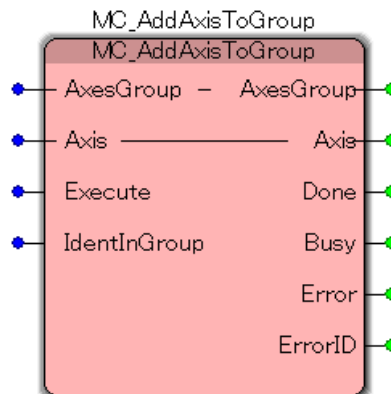
本項ではPLCopen MC Part4 仕様に定義されている同期制御を実現するため、グループ管理系のファンクションブロックについて説明します。グループに登録された軸は、3-3-1~3-3-3 までのファンクションブロックのほとんどは実行できなくなります。



## MC\_AddAxisToGroup 関数

**機能** 軸追加

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定
Axis	軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE: 実行無し
IdentInGroup	軸識別情報	IDENT_IN_GROUP_REF (UINT)	1~16	-	グループに追加する軸の 識別番号を指定

**出力**

変数	名称	データ型	範囲	内容
Done	完了	BOOL	TRUE or FALSE	正常終了時に TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを 出力

※1 : エラーコード一覧を参照

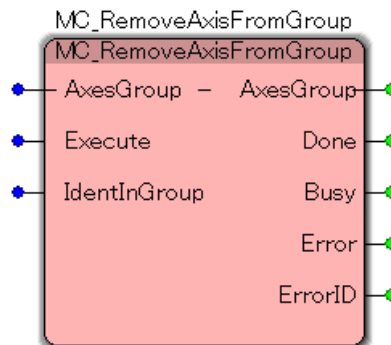
**説明**

指定した軸グループに対して軸を追加します。  
本ファンクションブロックにより軸グループへの登録が完了した軸は、単軸の動作を行う事は出来ません。  
単軸制御を行う場合は、MC\_RemoveAxisFromGroup()により軸グループから軸を削除してください。

## MC\_RemoveAxisFromGroup 関数

**機能** 軸削除

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE: 実行無し
IdentInGroup	軸識別情報	IDENT_IN_GROUP_REF (UINT)	1~16	-	グループに追加する軸の 識別番号を指定

**出力**

変数	名称	データ型	範囲	内容
Done	完了	BOOL	TRUE or FALSE	正常終了時に TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを 出力

※1: エラーコード一覧を参照

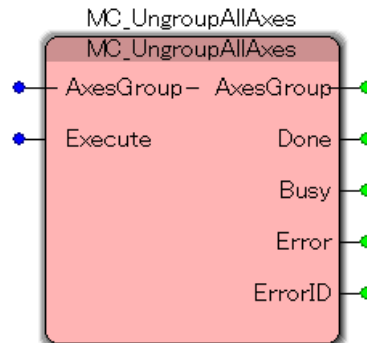
**説明**

指定した軸グループから軸を削除します。  
削除後に軸グループから軸が無くなった場合、グループの状態を GroupDisabled 状態に変更  
します。

## MC\_UngroupAllAxes 関数

**機能** 軸グループ解除

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し

**出力**

変数	名称	データ型	範囲	内容
Done	完了	BOOL	TRUE or FALSE	正常終了時に TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1 : エラーコード一覧を参照

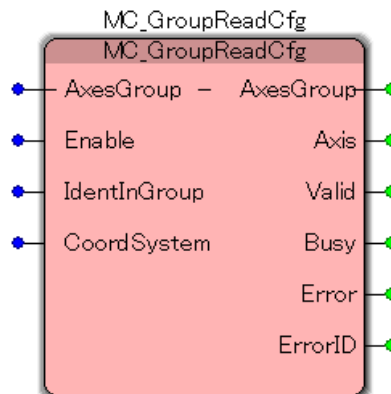
**説明**

指定した軸グループから軸を全て削除します。  
削除完了後、グループの状態を GroupDisabled 状態に変更します。

## MC\_GroupReadConfiguration 関数

**機能** 軸グループ設定読出

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し
IdentInGroup	軸識別情報	IDENT_IN_GROUP_REF (UINT)	1~16	FALSE	グループに追加する軸の 識別番号を指定
CoordSystem	座標系選択	ENUM (UINT)	-	-	未サポート

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	各読み込みデータが有効の間 TRUE
Axis	軸	AXIS_REF (UINT)	1~62	論理軸番号
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを 出力

※1：エラーコード一覧を参照

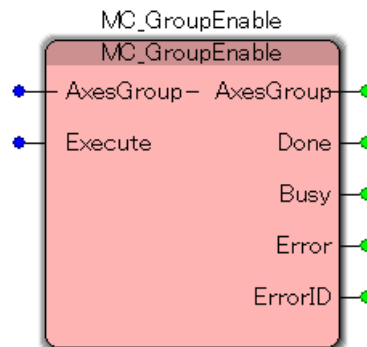
**説明**

指定した軸グループに追加されている軸番号を読み出します。

## MC\_GroupEnable 関数

**機能** 軸グループ有効

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE: 実行無し

**出力**

変数	名称	データ型	範囲	内容
Done	完了	BOOL	TRUE or FALSE	正常終了時に TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1 : エラーコード一覧を参照

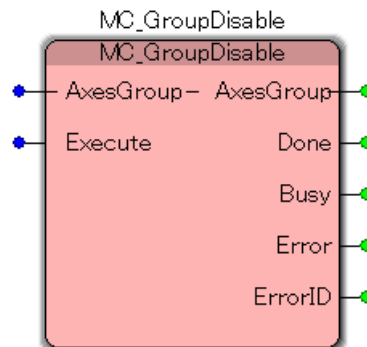
**説明**

軸グループの状態を GroupDisabled から GroupStandby に遷移します。  
本ファンクションブロックが正常終了後、動作ファンクションブロックを実行する事ができます。

## MC\_GroupDisable 関数

**機能** 軸グループ無効

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し

**出力**

変数	名称	データ型	範囲	内容
Done	完了	BOOL	TRUE or FALSE	正常終了時に TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1 : エラーコード一覧を参照

**説明**

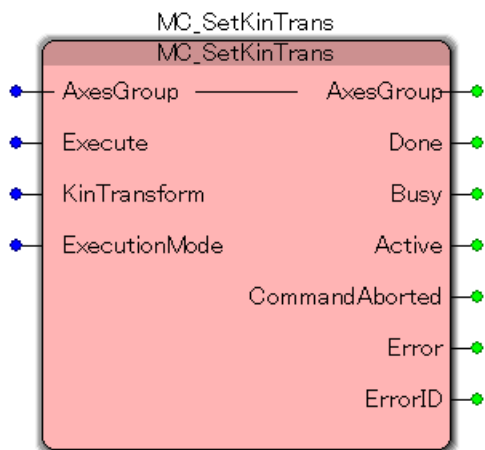
軸グループの状態を GroupDisabled に遷移します。  
本ファンクションブロック実行後は動作ファンクションブロックは実行できません。

### MC\_SetKinTransform 関数

---

**機能** 運動学的変換設定

**書式**

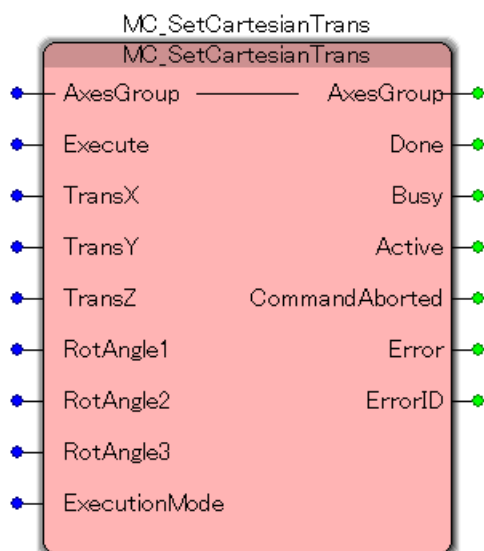


**説明** 未サポート

## MC\_SetCartesianTransform 関数

**機能** 直交座標変換設定

**書式**



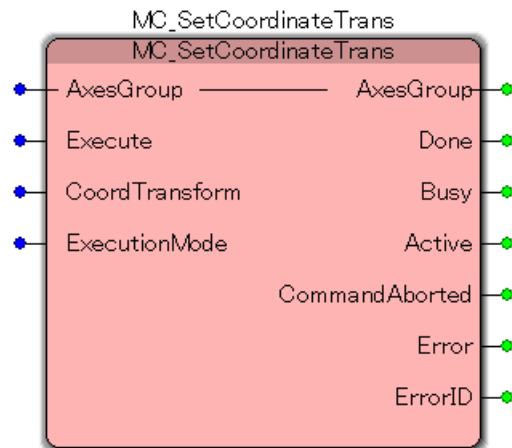
**説明** 未サポート



## MC\_SetCoordinateTransform 関数

**機能**

座標変換設定

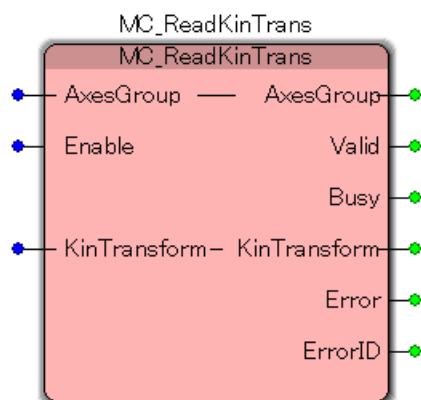
**書式****説明**

未サポート

## MC\_ReadKinTransform 関数

**機能**

運動学的変換読出

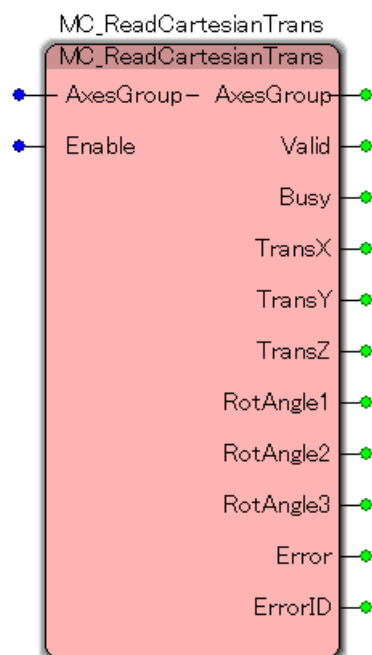
**書式****説明**

未サポート

## MC\_ReadCartesianTransform 関数

**機能**

直交座標変換読出

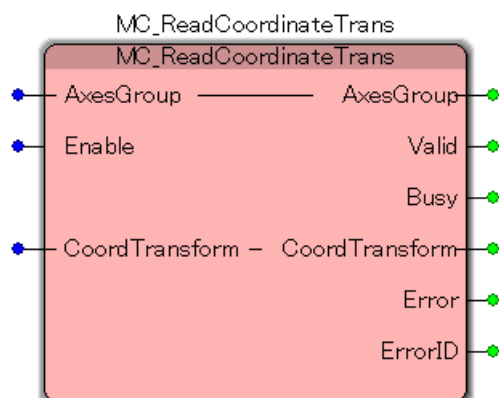
**書式****説明**

未サポート

## MC\_ReadCoordinateTransform 関数

**機能**

座標変換読出

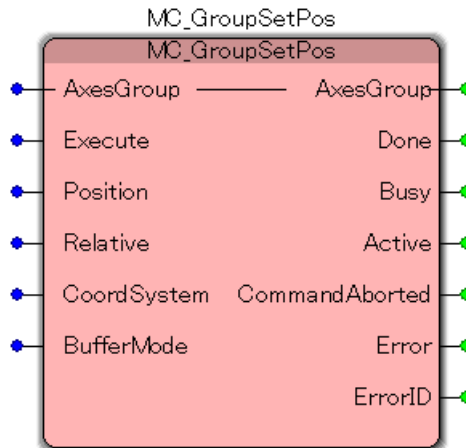
**書式****説明**

未サポート

MC\_GroupSetPosition 関数

**機能** 軸グループ現在位置変更

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し
Position	目標座標配列	ARRAY [1..16] of LREAL	倍精度実数値 【指令単位】	-	目標位置を指定 (位置モードに従う)
Relative	位置モード	BOOL	TRUE or FALSE	FALSE	TRUE :相対位置指令 FALSE:絶対位置指令
CoordSystem	座標系選択	ENUM (UINT)	-	-	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

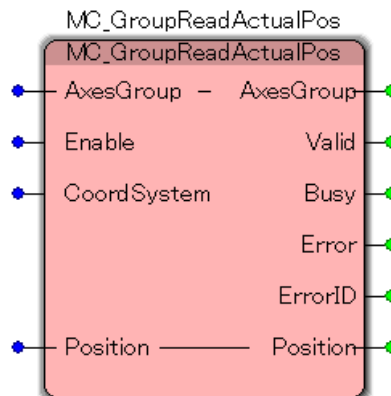
**説明**

軸グループの現在位置を変更します。

## MC\_GroupReadActualPosition 関数

**機能** 軸グループ現在位置読出

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定
Position	現在座標配列	ARRAY [1..16] of LREAL	倍精度実数値【指令単位】	-	絶対座標による現在位置

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE : 実行無し
CoordSystem	座標系選択	ENUM (UINT)	-	-	未サポート

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	各読み込みデータが有効の間 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

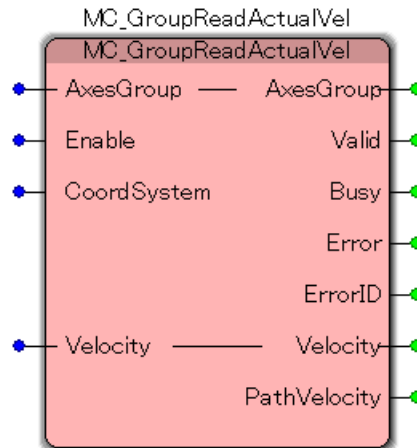
※1 : エラーコード一覧を参照

**説明** 軸グループの現在位置を読み出します。

## MC\_GroupReadActualVelocity 関数

**機能** 軸グループ現在速度読出

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定
Velocity	現在速度配列	ARRAY [1..16] of LREAL	倍精度実数値【指令単位】	-	軸毎の移動速度

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し
CoordSystem	座標系選択	ENUM (UINT)	-	-	未サポート

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	各読み込みデータが有効の間 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力
PathVelocity	経路速度	LREAL	倍精度実数値【指令単位】	経路上の移動速度

※1：エラーコード一覧を参照

**説明**

軸グループの現在速度を読み出します。  
読み出す現在速度は、グループ内の軸毎の現在速度と、同期のための合成速度になります。

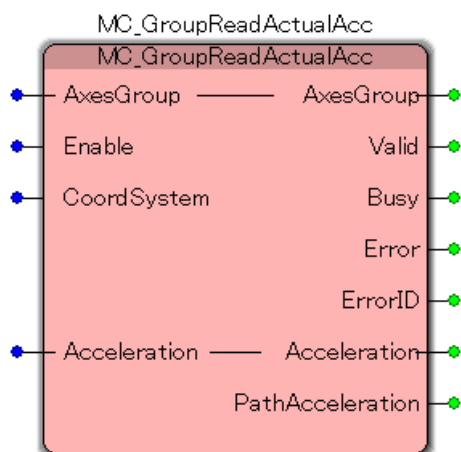


## MC\_GroupReadActualAcceleration 関数

## 機能

軸グループ現在加速度読出

## 書式



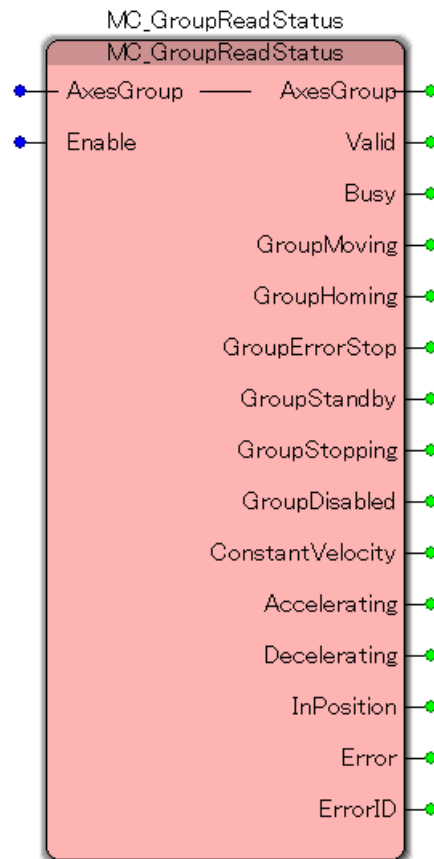
## 説明

未サポート

## MC\_GroupReadStatus 関数

**機能** 軸グループステータス読出

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	各読み込みデータが有効の間 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
GroupMoving	Group Moving 状態	BOOL	TRUE or FALSE	現在の軸状態を出力 いずれかの状態のみが TRUE となり、2 つ以上が同時に TRUE になることはありません。
GroupHoming	Group Homing 状態	BOOL	TRUE or FALSE	
GroupErrorStop	Group ErrorStop 状態	BOOL	TRUE or FALSE	
GroupStandby	Group Standby 状態	BOOL	TRUE or FALSE	
GroupStopping	Group Stopping 状態	BOOL	TRUE or FALSE	
GroupDisabled	Group Disabled 状態	BOOL	TRUE or FALSE	
ConstantVelocity	定速動作中	BOOL	TRUE or FALSE	グループが定速動作中のとき TRUE
Accelerating	加速中	BOOL	TRUE or FALSE	グループが加速中のとき TRUE
Decelerating	減速中	BOOL	TRUE or FALSE	グループが減速中のとき TRUE
InPosition	位置範囲内	BOOL	TRUE or FALSE	グループが目標位置範囲内の時 TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

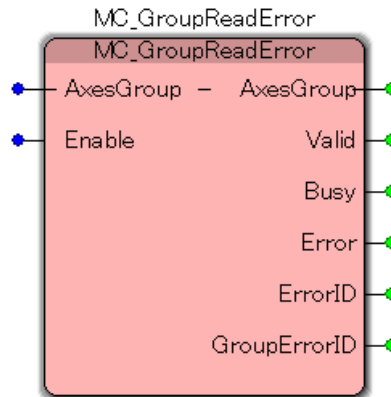
**説明**

指定したグループの状態遷移図上の状態を読み出します。

## MC\_GroupReadError 関数

**機能** 軸グループエラー読出

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し

**出力**

変数	名称	データ型	範囲	内容
Valid	出力有効	BOOL	TRUE or FALSE	各読み込みデータが有効の間 TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力
GroupError ID	グループエラー番号	DWORD	※1	グループ内のエラーコードを出力

※1 : エラーコード一覧を参照

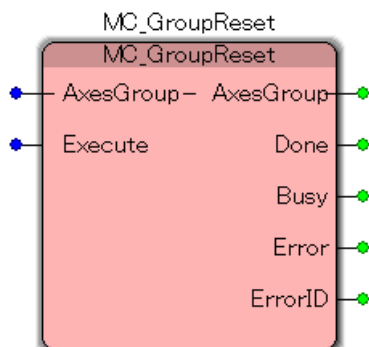
**説明**

軸グループに発生しているエラーを読み出します。  
軸毎に発生しているエラーについては、MC\_ReadError をご使用ください。

## MC\_GroupReset 関数

**機能** 軸グループエラーリセット

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し

**出力**

変数	名称	データ型	範囲	内容
Done	完了	BOOL	TRUE or FALSE	正常終了時に TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1 : エラーコード一覧を参照

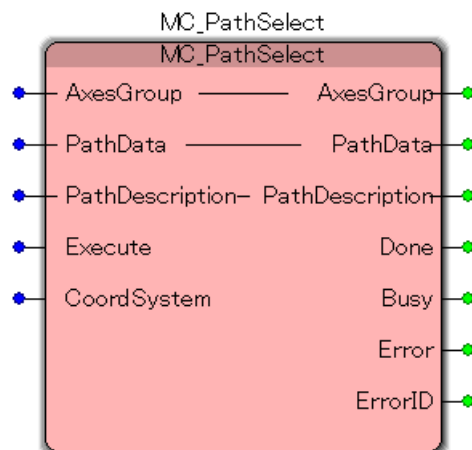
**説明**

軸グループに発生しているエラーをリセットします。

## MC\_PathSelect 関数

**機能**

経路選択

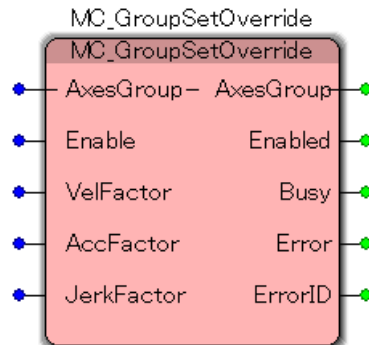
**書式****説明**

未サポート

## MC\_GroupSetOverride 関数

**機能** 軸グループオーバーライド値設定

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE: 実行無し
VelFactor	速度倍率	LREAL	倍精度実数値正数 【0.0<値<=2.0】	-	現在の速度に対する 倍率を指定
AccFactor	加速度倍率	LREAL	倍精度実数値正数 【0.0<値<=2.0】	-	現在の加速度に対する 倍率を指定
JerkFactor	加加速度倍率	LREAL	倍精度実数値正数 【0.0<値<=2.0】	-	未サポート

**出力**

変数	名称	データ型	範囲	内容
Enabled	有効化完了	BOOL	TRUE or FALSE	正常終了時に TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを 出力

※1 : エラーコード一覧を参照

**説明**

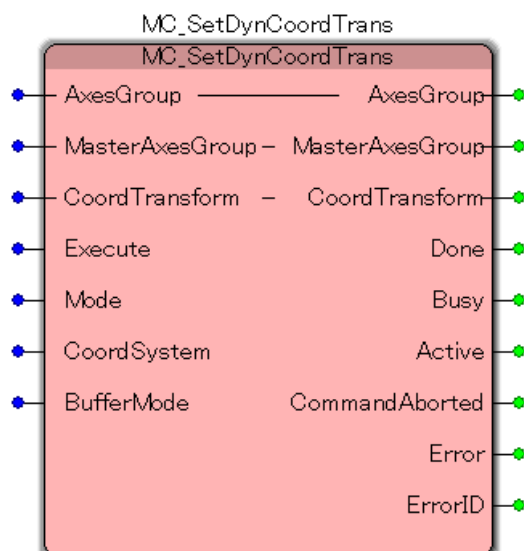
実行中の位置決め動作の、速度・加速(減速)度に対して倍率指定によりパラメータを変更します。

速度倍率に対して 0.0 を指定した場合、状態遷移を発生させずに軸を停止します。

## MC\_SetDynCoordTransform 関数

**機能**

動的座標変換設定

**書式****説明**

未サポート



### 3-3-5 PLCopen仕様 MC Part4 動作ファンクションブロック

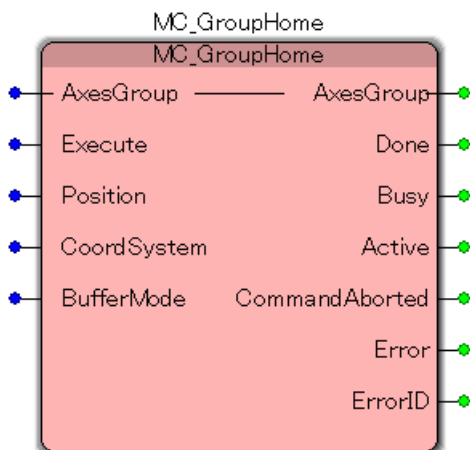
本項ではPLCopen MC Part4 仕様に定義されている、同期制御動作系のファンクションブロックについて説明します。

### MC\_GroupHome 関数

---

**機能** 軸グループ原点復帰

**書式**

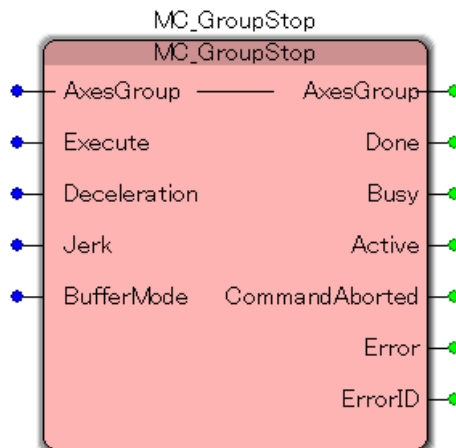


**説明** 未サポート

## MC\_GroupStop 関数

**機能** 軸グループ強制停止

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	-	未サポート
Jerk	加加速度	LREAL	-	-	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

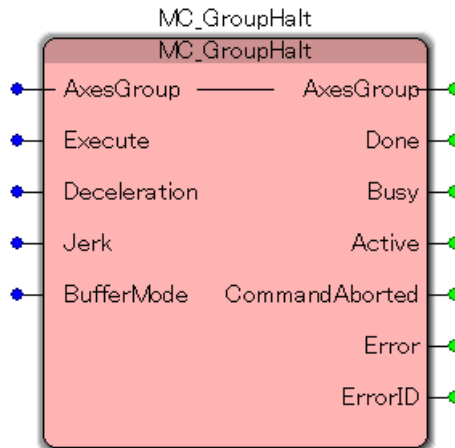
**説明**

軸グループの位置決めを停止します。  
実行後 GroupStopping 状態に遷移し、停止完了後に GroupStandby 状態に遷移します。

MC\_GroupHalt 関数

**機能** 軸グループ停止

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	-	未サポート
Jerk	加加速度	LREAL	-	-	未サポート
BufferMode	動作モード	MC_BufferMode	0~5	0	モーション命令 多重起動命令動作指定 <b>本FBでは、0:Abortingのみサポート</b>

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

軸グループの位置決めを停止します。

本ファンクションブロックは GroupStopping 状態への遷移を発生させずに位置決めを停止します。

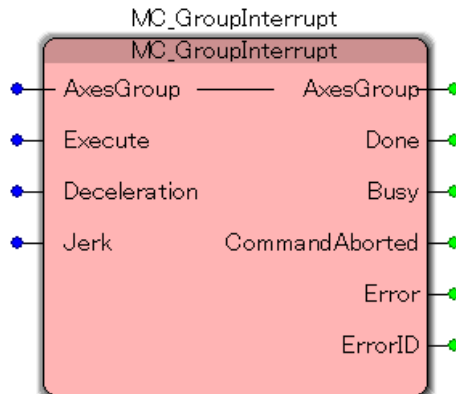
停止後に GroupStandby 状態に遷移します。

GroupStopping 状態へ遷移しないため、本ファンクションブロック実行中に位置決めファンクションブロックの起動が可能です。

## MC\_GroupInterrupt 関数

**機能** 軸グループ一時停止

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	-	未サポート
Jerk	加加速度	LREAL	-	-	未サポート

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1 : エラーコード一覧を参照

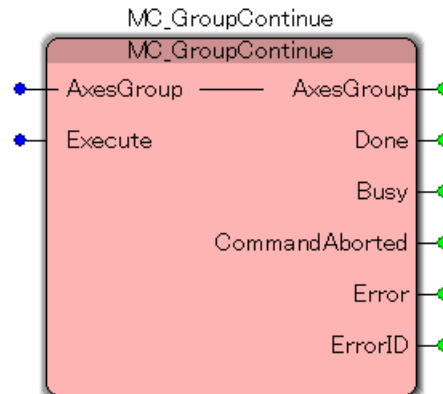
**説明**

軸グループの位置決めを一時停止します。  
一時停止中、停止後も状態遷移は発生しません。  
本ファンクションブロック後に MC\_GroupContinue を実行する事で、続きの動作を行うことができます。  
また、一時停止前とは違う位置決め関数の実行もできます。

## MC\_GroupContinue 関数

**機能** 軸グループ一時停止解除

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE :実行 FALSE:実行無し

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1 : エラーコード一覧を参照

**説明**

MC\_GroupInterrupt により一時停止された動作を再開することができます。

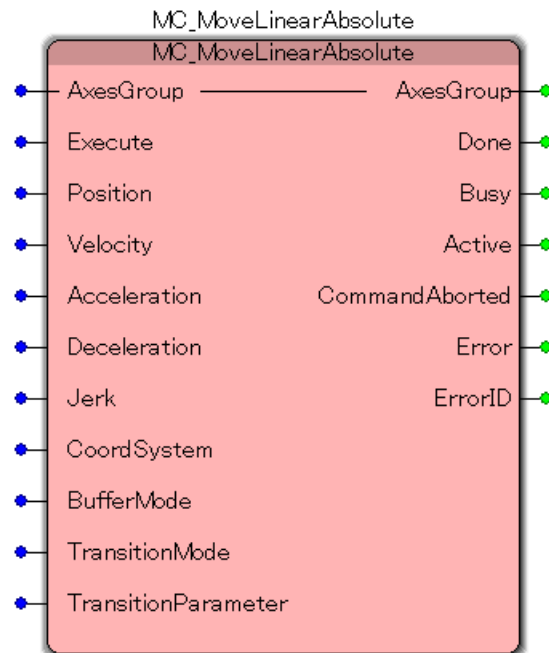


## MC\_MoveLinearAbsolute 関数

## 機能

絶対値直線補間

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE: 実行無し
Position	目標座標配列	ARRAY [1..16] of LREAL	倍精度実数値 【指令単位】	-	目標位置を指定 (位置モードに従う)
Velocity	速度	LREAL	倍精度実数値正数 【指令単位/s】	-	位置決め時の速度
Acceleration	加速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	-	位置決め時の加速度
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	位置決め時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
CoordSystem	座標系選択	ENUM (UINT)	-	-	未サポート
BufferMode	動作モード	MC_BufferMode (UINT16)	0~5	0	モーション命令 多重起動命令動作指定 <b>0: Aborting</b> <b>1: Buffered</b> <b>3: BlendingPrevious のみ</b>
Transition Mode	トランジションモード	MC_TransitionMode (UINT16)	0 or 10	0	FB 同士を接続したときの 遷移カーブを決定します。 <b>0: TMNone</b> <b>10: TMSmoothing のみ</b>
Transition Parameter	トランジションパラメータ	ARRAY [1..4] of LREAL	-	-	未サポート

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
MC_TransitionMode	UINT16	0	TMNone	重ねて次の動作を行うときの速度および軌跡の遷移カーブを指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1~9	PLCopen で予約	
		10	TMSmoothing	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

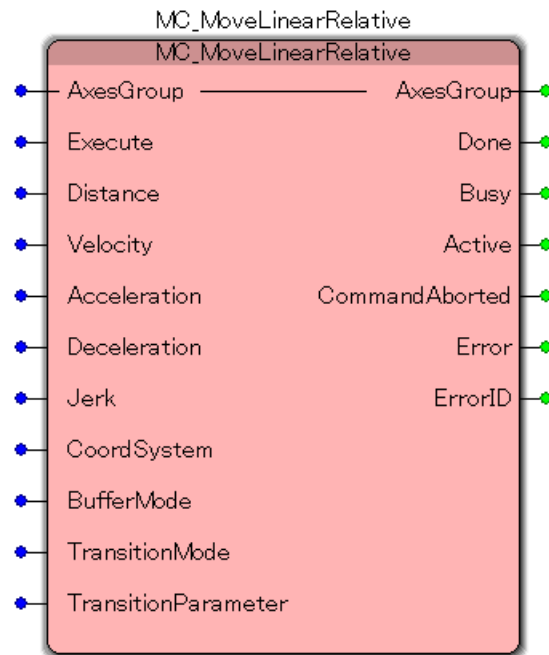
絶対座標指令による直線補間を行います。

## MC\_MoveLinearRelative 関数

## 機能

相対値直線補間

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE: 実行無し
Distance	目標座標配列	ARRAY [1..16] of LREAL	倍精度実数値 【指令単位】	-	目標位置を指定 (位置モードに従う)
Velocity	速度	LREAL	倍精度実数値正数 【指令単位/s】	-	位置決め時の速度
Acceleration	加速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	-	位置決め時の加速度
Deceleration	減速度	LREAL	倍精度実数値正数 【指令単位/s <sup>2</sup> 】	0	位置決め時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
CoordSystem	座標系選択	ENUM (UINT)	-	-	未サポート
BufferMode	動作モード	MC_BufferMode (UINT16)	0~5	0	モーション命令 多重起動命令動作指定 <b>0: Aborting</b> <b>1: Buffered のみ。</b>
Transition Mode	トランジションモード	MC_TransitionMode (UINT16)	0 or 10	0	FB 同士を接続したときの 遷移カーブを決定します。 <b>0: TMNone</b> <b>10: TMSmoothing のみ</b>
Transition Parameter	トランジションパラメータ	ARRAY [1..4] of LREAL	-	-	未サポート

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
MC_TransitionMode	UINT16	0	TMNone	重ねて次の動作を行うときの速度および軌跡の遷移カーブを指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1~9	PLCopen で予約	
		10	TMSmoothing	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

**説明**

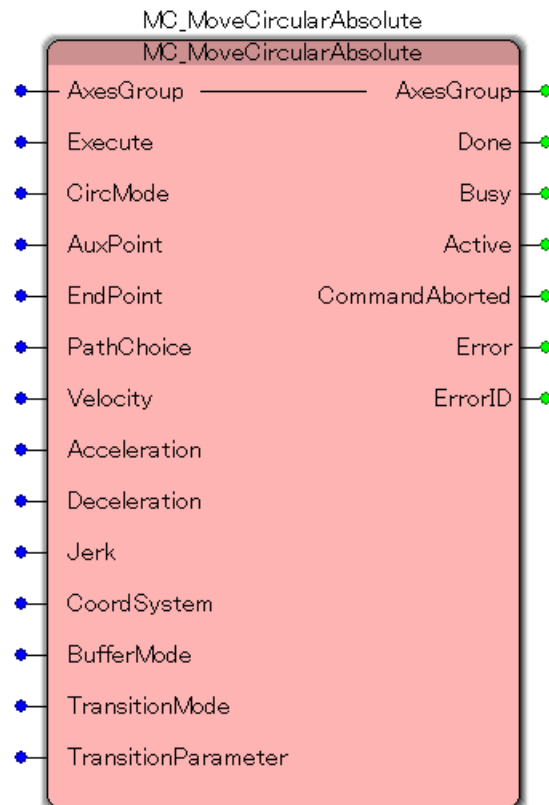
現在位置に対する相対座標指定による直線補間を行います。

## MC\_MoveCircularAbsolute 関数

## 機能

絶対値円弧補間

## 書式



## 入出力

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE: 実行無し
CircMode	円弧補間モード	MC_CircMode (UINT)	0~2	0	AuxPoint, EndPoint の意味を決定
AuxPoint	補助点	ARRAY [1..16] of LREAL	倍精度実数値【指令単位】	-	CircMode に従う。 絶対位置指定
EndPoint	終点	ARRAY [1..16] of LREAL	倍精度実数値【指令単位】	-	CircMode に従う。 絶対位置指定
PathChoice	経路選択	MC_PathChoice (UINT)	0~1	0	移動方向
Velocity	速度	LREAL	倍精度実数値正数【指令単位/s】	-	位置決め時の速度
Acceleration	加速度	LREAL	倍精度実数値正数【指令単位/s <sup>2</sup> 】	-	位置決め時の加速度
Deceleration	減速度	LREAL	倍精度実数値正数【指令単位/s <sup>2</sup> 】	0	位置決め時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
CoordSystem	座標系選択	ENUM (UINT)	-	-	未サポート
BufferMode	動作モード	MC_BufferMode (UINT)	0~5	0	モーション命令 多重起動命令動作指定 <b>0: Aborting</b> <b>1: Buffered のみ。</b>
Transition Mode	トランジションモード	MC_TransitionMode (UINT16)	0 or 10	0	FB 同士を接続したときの 遷移カーブを決定します。 <b>0: TMNone</b> <b>10: TMSmoothing のみ</b>
Transition Parameter	トランジションパラメータ	ARRAY [1..4] of LREAL	-	-	未サポート

型名	型	値	モード	説明
MC_CircMode	UINT16	0	BORDER	「AuxPoint」はスタート位置から、「EndPoint」で構成される円周上の点。
		1	CENTER	「AuxPoint」は中心点。
		2	RADIUS	「AuxPoint」は、右母指の規則による円平面の垂直の先頭ポイントを定義。円の半径はベクトルの長さ。
MC_PathChoice	UINT16	0	時計回り (CW)	スタート位置から「EndPoint」へ時計回りに円弧補間を実行します。
		1	反時計回り (CCW)	スタート位置から「EndPoint」へ反時計回りに円弧補間を実行します。



型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	
MC_TransitionMode	UINT16	0	TMNone	重ねて次の動作を行うときの速度および軌跡の遷移カーブを指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1~9	PLCopen で予約	
		10	TMSmoothing	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

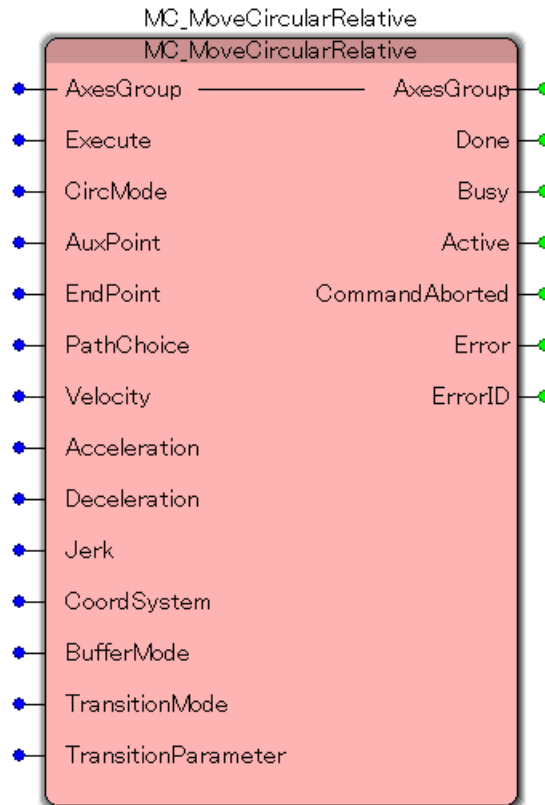
**説明**

絶対座標指令による2軸円弧補間を行います。2軸以上登録されているグループに対して実行した場合はエラーとなります。

MC\_MoveCircularRelative 関数

**機能** 相対値円弧補間

**書式**



**入出力**

変数	名称	データ型	範囲	初期値	内容
AxesGroup	軸グループ	AXES_GROUP_REF (UINT)	1~32	-	論理グループ番号を指定

## 入力

変数	名称	データ型	範囲	初期値	内容
Execute	起動トリガ	BOOL	TRUE or FALSE	FALSE	TRUE : 実行 FALSE: 実行無し
CircMode	円弧補間モード	MC_CircMode (UINT)	0~2	0	AuxPoint, EndPoint の意味を決定
AuxPoint	補助点	ARRAY [1..16] of LREAL	倍精度実数値【指令単位】	-	CircMode に従う。相対位置指定
EndPoint	終点	ARRAY [1..16] of LREAL	倍精度実数値【指令単位】	-	CircMode に従う。絶対位置指定
PathChoice	経路選択	MC_PathChoice (UINT)	0~1	0	移動方向
Velocity	速度	LREAL	倍精度実数値正数【指令単位/s】	-	位置決め時の速度
Acceleration	加速度	LREAL	倍精度実数値正数【指令単位/s <sup>2</sup> 】	-	位置決め時の加速度
Deceleration	減速度	LREAL	倍精度実数値正数【指令単位/s <sup>2</sup> 】	0	位置決め時の減速度
Jerk	加加速度	LREAL	-	-	未サポート
CoordSystem	座標系選択	ENUM (UINT)	-	-	未サポート
BufferMode	動作モード	MC_BufferMode (UINT)	0~5	0	モーション命令多重起動命令動作指定 <b>0: Aborting</b> <b>1: Buffered のみ。</b>
Transition Mode	トランジションモード	MC_TransitionMode (UINT16)	0 or 10	0	FB 同士を接続したときの遷移カーブを決定します。 <b>0: TMNone</b> <b>10: TMSmoothing のみ</b>
Transition Parameter	トランジションパラメータ	ARRAY [1..4] of LREAL	-	-	未サポート

型名	型	値	モード	説明
MC_CircMode	UINT16	0	BORDER	「AuxPoint」はスタート位置から、「EndPoint」で構成される円周上の点。
		1	CENTER	「AuxPoint」は中心点。
		2	RADIUS	「AuxPoint」は、右母指の規則による円平面の垂直の先頭ポイントを定義。円の半径はベクトルの長さ。
MC_PathChoice	UINT16	0	時計回り (CW)	スタート位置から「EndPoint」へ時計回りに円弧補間を実行します。
		1	反時計回り (CCW)	スタート位置から「EndPoint」へ反時計回りに円弧補間を実行します。

型名	型	値	モード	説明
MC_BufferMode	UINT16	0	Aborting	実行中のFBに対して、重ねて次の動作指示を出す場合の制御方法を指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1	Buffered	
		2	BlendingLow	
		3	BlendingPrevious	
		4	BlendingNext	
		5	BlendingHigh	
MC_TransitionMode	UINT16	0	TMNone	重ねて次の動作を行うときの速度および軌跡の遷移カーブを指定 詳細は「3-4-2 Part4 動作ファンクションブロックの多重起動」を参照してください。
		1~9	PLCopen で予約	
		10	TMSmoothing	

**出力**

変数	名称	データ型	範囲	内容
Done	実行完了通知	BOOL	TRUE or FALSE	書き込み完了で TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	軸制御中に TRUE
CommandAborted	実行中断	BOOL	TRUE or FALSE	命令が中止されたときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1：エラーコード一覧を参照

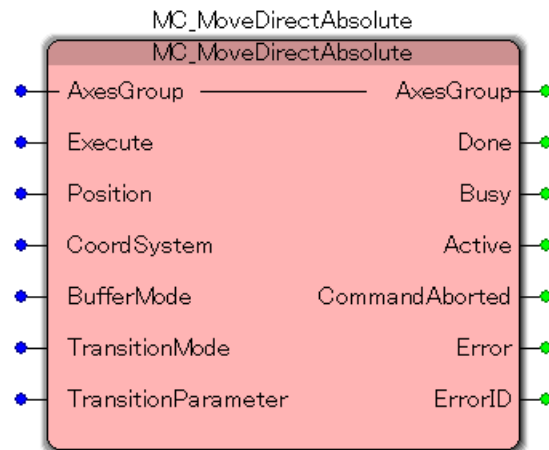
**説明**

現在位置に対する相対座標指定による円弧補間を行います。2 軸以上登録されているグループに対して実行した場合はエラーとなります。

## MC\_MoveDirectAbsolute 関数

**機能**

軸グループ絶対値位置決め

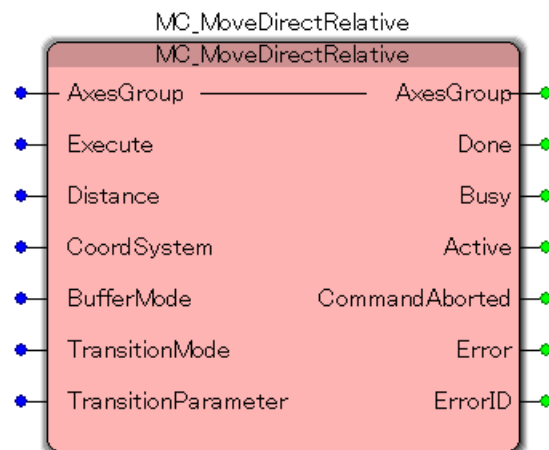
**書式****説明**

未サポート

## MC\_MoveDirectRelative 関数

**機能** 軸グループ相対値位置決め

**書式**

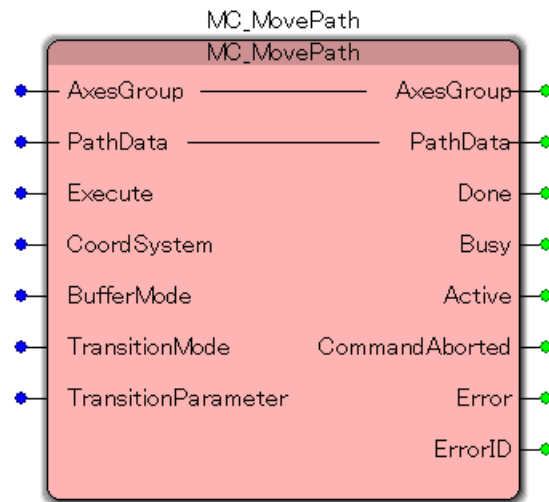


**説明** 未サポート

## MC\_MovePath 関数

**機能**

指定経路移動

**書式****説明**

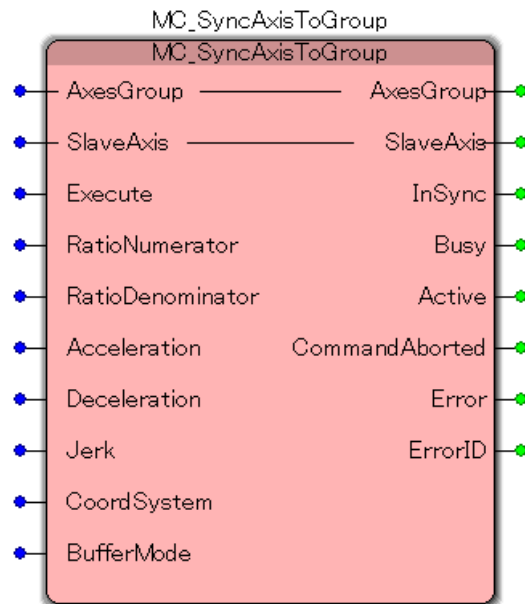
未サポート

## MC\_SyncAxisToGroup 関数

## 機能

グループへの単軸同期動作

## 書式



## 説明

未サポート

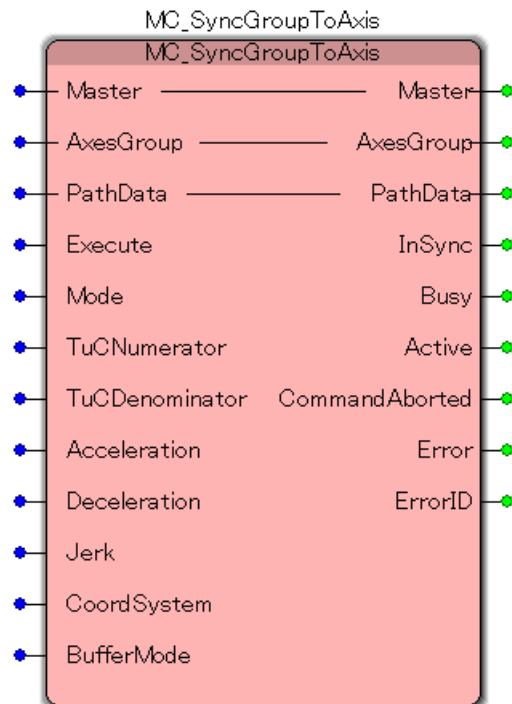


## MC\_SyncGroupToAxis 関数

## 機能

単軸へのグループ同期動作

## 書式



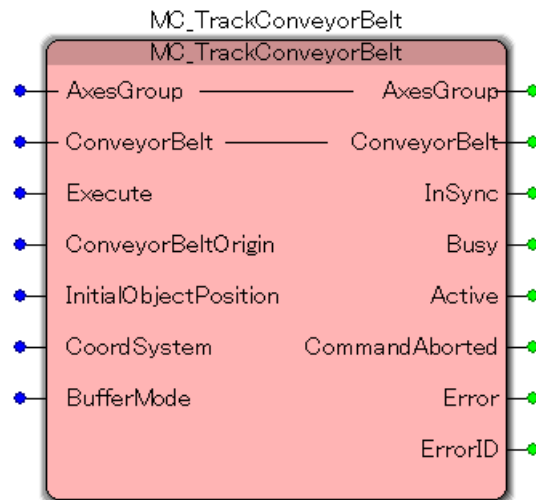
## 説明

未サポート

## MC\_TrackConveyorBelt 関数

**機能**

コンベヤ追従動作

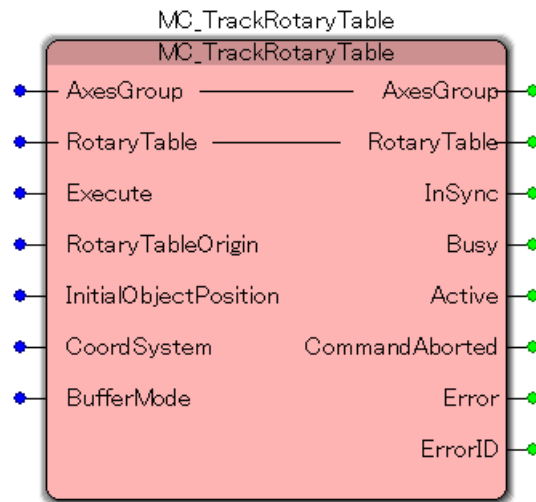
**書式****説明**

未サポート

## MC\_TrackRotaryTable 関数

**機能**

ロータリテーブル追従動作

**書式****説明**

未サポート

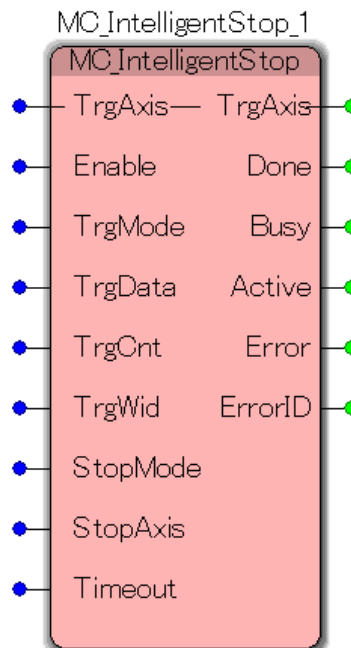
### 3-3-6 PLCopen 仕様範囲外 特殊ファンクションブロック

本項では当社オリジナルのファンクションブロックについて説明します。状態遷移については、PLCopen の仕様に準拠します。

## MC\_IntelligentStop 関数

**機能**

指定したトリガによって、複数の軸を同時に停止させることができます。  
EtherCAT 版のみ有効

**書式****入出力**

変数	名称	データ型	範囲	初期値	内容
TrgAxis	トリガ軸	AXIS_REF (UINT)	1~62	-	論理軸番号を指定

**入力**

変数	名称	データ型	範囲	初期値	内容
Enable	有効	BOOL	TRUE or FALSE	FALSE	TRUE : トリガ監視開始 FALSE: トリガ監視解除
TrgMode	トリガモード	UINT	0 : 位置 1 : 速度 2 : トルク	0	トリガ監視するパラメータを選択
TrgData	トリガデータ	LREAL	倍精度実数値 位置【指令単位】 速度【指令単位/s】 トルク【N・m】 or 【%】	-	現在位置がこの値を超えた場合、現在速度 or トルクがこの値と一致したときに停止コマンドを発行
TrgCnt	トリガカウント	UINT	0~65535【回】	0	設定した回数連続してトリガ一致したときに即停止を発行
TrgWid	トリガー致幅	LREAL	倍精度実数値 位置【指令単位】 速度【指令単位/s】 トルク【N・m】 or 【%】	0	トリガー致の幅を設定
StopMode	停止モード	UINT	0 : 即停止 1 : 減速停止	0	停止モードを設定
StopAxis	停止軸 ID	UINT[16]	0 : 無効 1~62 : 論理軸番号	0	停止させる軸を最大 16 軸指定
Timeout	タイムアウト	DWORD	0 : 無限待ち 1~4294967295【ms】	0	トリガ監視開始してから、トリガー一致するまでのタイムアウト時間

トルク値の単位 : サーボパックマニュアルの GiA402 パラメータ「目標トルク (0x6071)」または、「内部指令トルク (0x6074)」の単位を参照してください。

**出力**

変数	名称	データ型	範囲	内容
Done	停止完了	BOOL	TRUE or FALSE	停止完了したときに TRUE
Busy	実行中	BOOL	TRUE or FALSE	命令を受け付けたときに TRUE
Active	軸制御中	BOOL	TRUE or FALSE	トリガー致して停止を実行のときに TRUE
Error	エラー発生中	BOOL	TRUE or FALSE	異常が発生したとき TRUE
ErrorID	エラー番号	DWORD	※1	異常が発生したときのエラーコードを出力

※1 : エラーコード一覧を参照

**説明**

本ファンクションブロックは、Enable（有効）を TRUE にすると、TrgAxis で指定した軸の現在位置 or 現在速度 or 現在トルクを監視します。監視を開始すると Busy が TRUE になります。

監視するトリガは TrgMode で指定します。トリガ一致条件はそれぞれ下記のようになります。

<TrgMode=0 : 位置>

移動方向が+の場合

現在位置 - TrgWid > TrgData

移動方向が-の場合

現在位置 + TrgWid < TrgData

<TrgMode=1 : 速度>

現在速度（絶対値） + TrgWid >= TrgData

かつ

現在速度（絶対値） - TrgWid <= TrgData

<TrgMode=2 : トルク>

現在トルク（絶対値） + TrgWid >= TrgData

かつ

現在トルク（絶対値） - TrgWid <= TrgData

連続して TrgCnt 回数だけ条件が一致したら StopMode で指定した方式で、StopAxis に格納された軸に対して停止コマンドを実行します。

停止コマンドを実行すると、Active が TRUE になります。

停止コマンドを実行した全軸が停止したことを確認すると、Done が TRUE になります。

Timeout に 0 以外の値を設定したとき、Busy が TRUE になってから、条件が一致するまでの時間【ms】が Timeout を超えたとき、Error が TRUE になり、タイムアウトエラーとなります。

### 3-4 モーション制御機能

本項では、PLCopen で規定されている、モーション制御の特殊な使い方について説明します。

#### 3-4-1 動作ファンクションブロックの多重起動

いくつかの FB は「BufferMode」と呼ばれる入力を持ちます。この入力により、FB は「Aborting mode」（デフォルト動作）と「Buffered mode」の両方で動作が可能です。これらのモードの相違点は、それらの動作がいつ開始されるかというものです。

- ・ 非バッファリングモードでのコマンドは、他の動作を中断してでもすぐに動作します。
- ・ バッファリングモードでのコマンドは、現在の FB が自身の「Done」（または「InPosition」や「InVelocity」）出力をセットするまで待ちます。

バッファリングモードには、いくつかのオプションがあります。この入力は MC\_BUFFERMODE の ENUM 型です。表 3-4-1-1 に各バッファリングモードの一覧を示します。

表 3-4-1-1. バッファリングモード一覧

値	モード	内容
0	Aborting	バッファリングしないデフォルトモード。次の FB は、実行中の動作を中断し、コマンドは直ちに軸に影響します。
1	Buffered	次の FB は、以前の動作が「Done」になると、直ちに軸に影響します。
2	BlendingLow	次の FB は、以前の FB が完了した後に軸を制御しますが、2つの動作間で軸は停止しません。 FB1 の終了位置で FB1 と FB2 の低い速度とします。
3	BlendingPrevious	次の FB は、以前の FB が完了した後に軸を制御しますが、2つの動作間で軸は停止しません。 FB1 の終了位置で FB1 の速度とします。
4	BlendingNext	次の FB は、以前の FB が完了した後に軸を制御しますが、2つの動作間で軸は停止しません。 FB1 の終了位置で FB2 の速度とします。
5	BlendingHigh	次の FB は、以前の FB が完了した後に軸を制御しますが、2つの動作間で軸は停止しません。 FB1 の終了位置で FB1 と FB2 の高い速度とします。

以下の例は、これらモードの動作の相違を記述しています。



- 連続した2つの絶対位置移動の標準動作

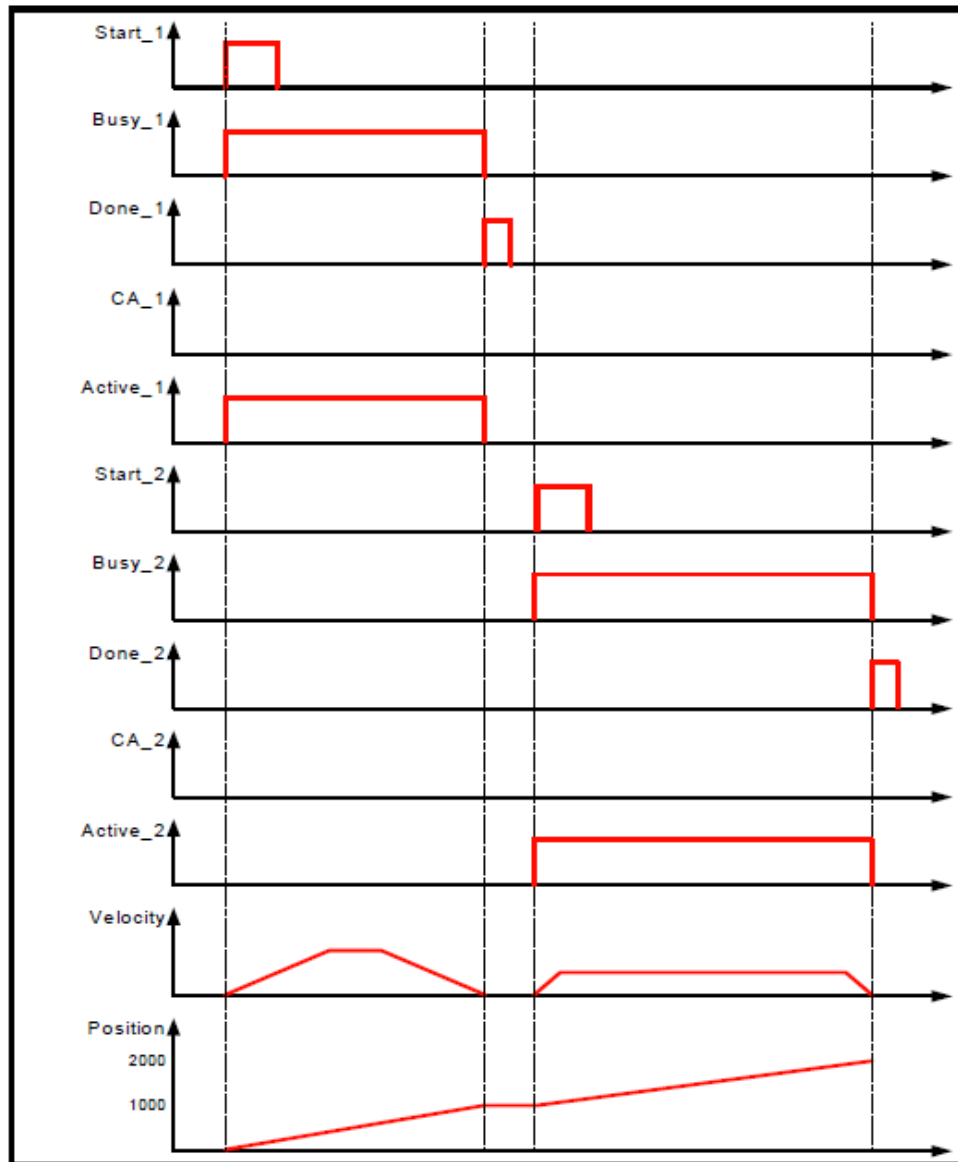
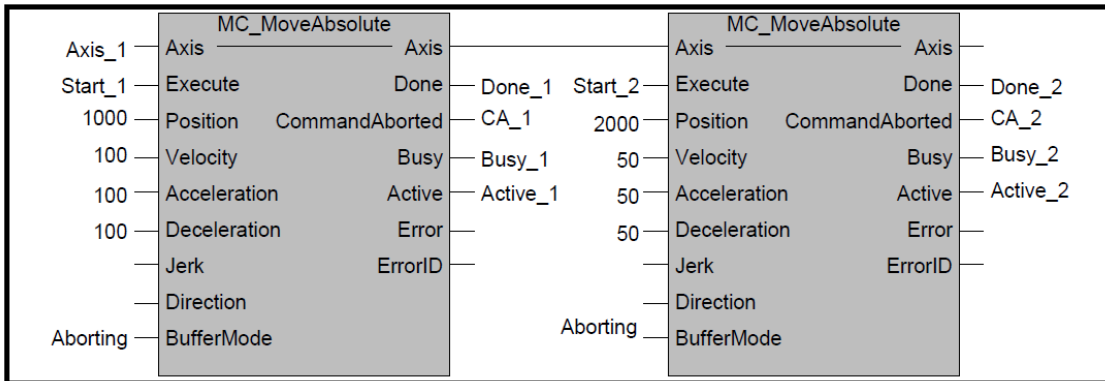


図 3-4-1-1. 上の例で FB1 と FB2 の間に干渉が無い場合でのタイムチャート (Aborting Mode)

● Aborting mode での動作

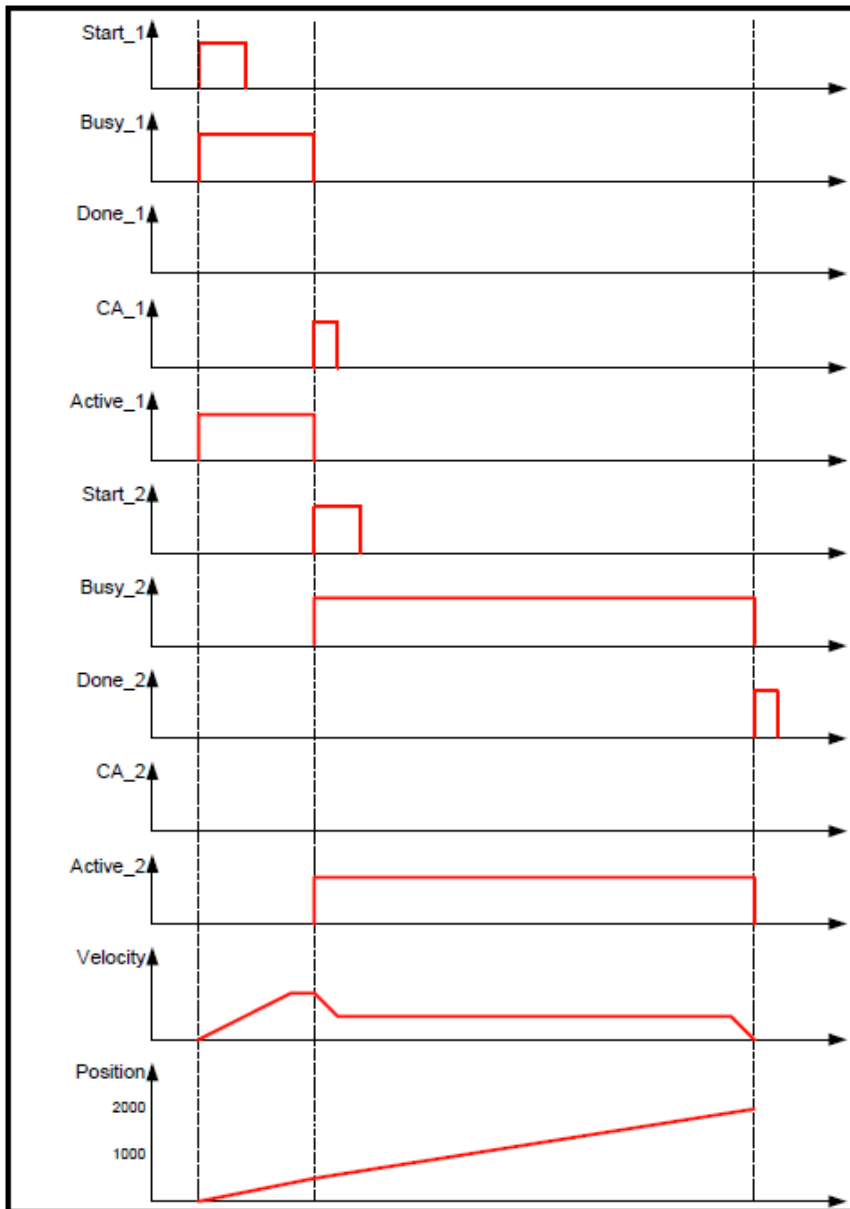
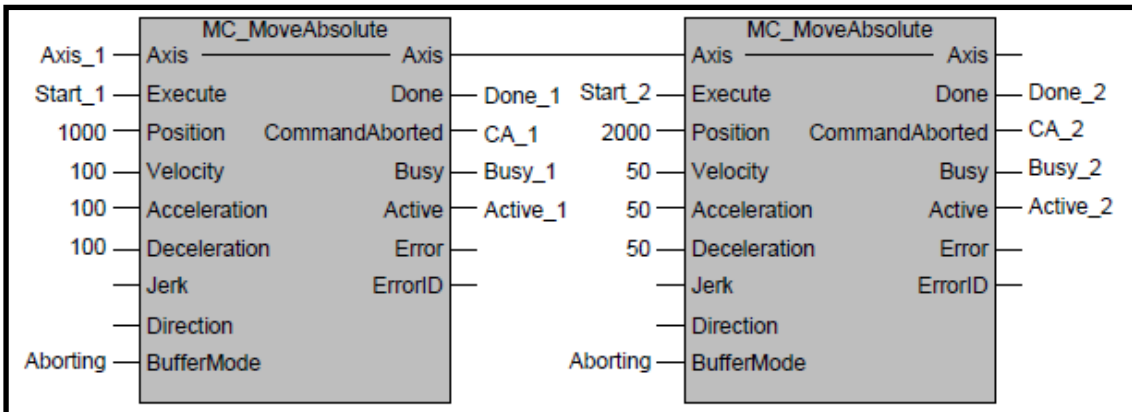


図 3-4-1-2. 上の例の FB2 が FB1 に割り込む場合でのタイムチャート (Aborting Mode)

● Buffered mode での動作

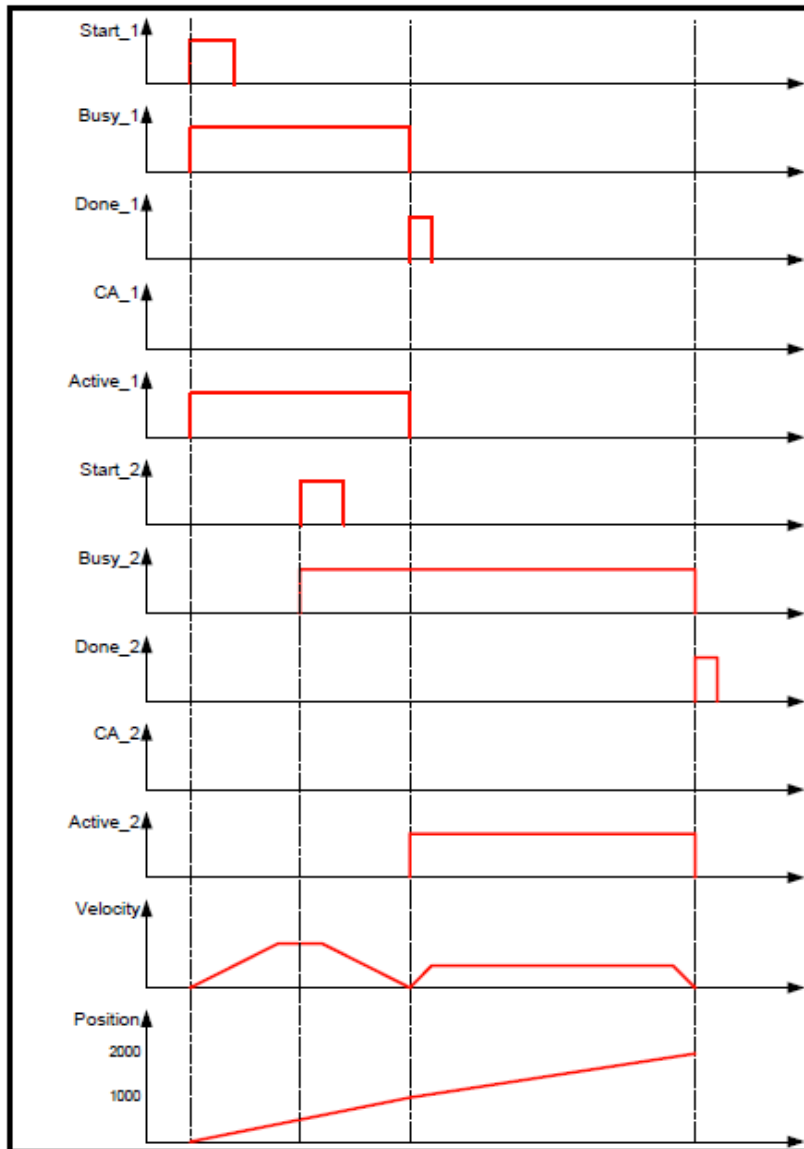
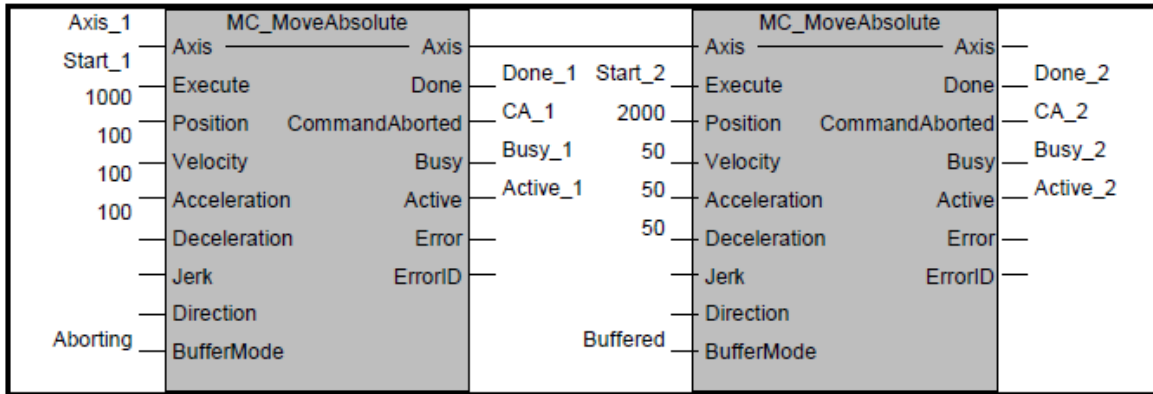


図 3-4-1-3. 上の例の Buffered Mode でのタイムチャート  
(速度 0 で停止し、遅延なくその位置で FB2 を開始する)

● BlendingLow mode での動作

Axis_1	MC_MoveAbsolute				Axis	MC_MoveAbsolute				Axis	MC_MoveAbsolute				Axis
Start_1	Execute	Done	Done_1	Start_2	Execute	Done	Done_2	Start_3	Execute	Done	Done_3	Start_3	Execute	Done	Done_3
1000	Position	CommandAborted	CA_1	2000	Position	CommandAborted	CA_2	3000	Position	CommandAborted	CA_3	3000	Position	CommandAborted	CA_3
100	Velocity	Busy	Busy_1	50	Velocity	Busy	Busy_2	100	Velocity	Busy	Busy_3	100	Velocity	Busy	Busy_3
100	Acceleration	Active	Active_1	50	Acceleration	Active	Active_2	100	Acceleration	Active	Active_3	100	Acceleration	Active	Active_3
100	Deceleration	Error		50	Deceleration	Error		100	Deceleration	Error		100	Deceleration	Error	
	Jerk	ErrorID			Jerk	ErrorID			Jerk	ErrorID			Jerk	ErrorID	
	Direction				Direction				Direction				Direction		
Aborting	BufferMode		BlendingLow		BufferMode		BlendingLow		BufferMode		BlendingLow		BufferMode		BlendingLow

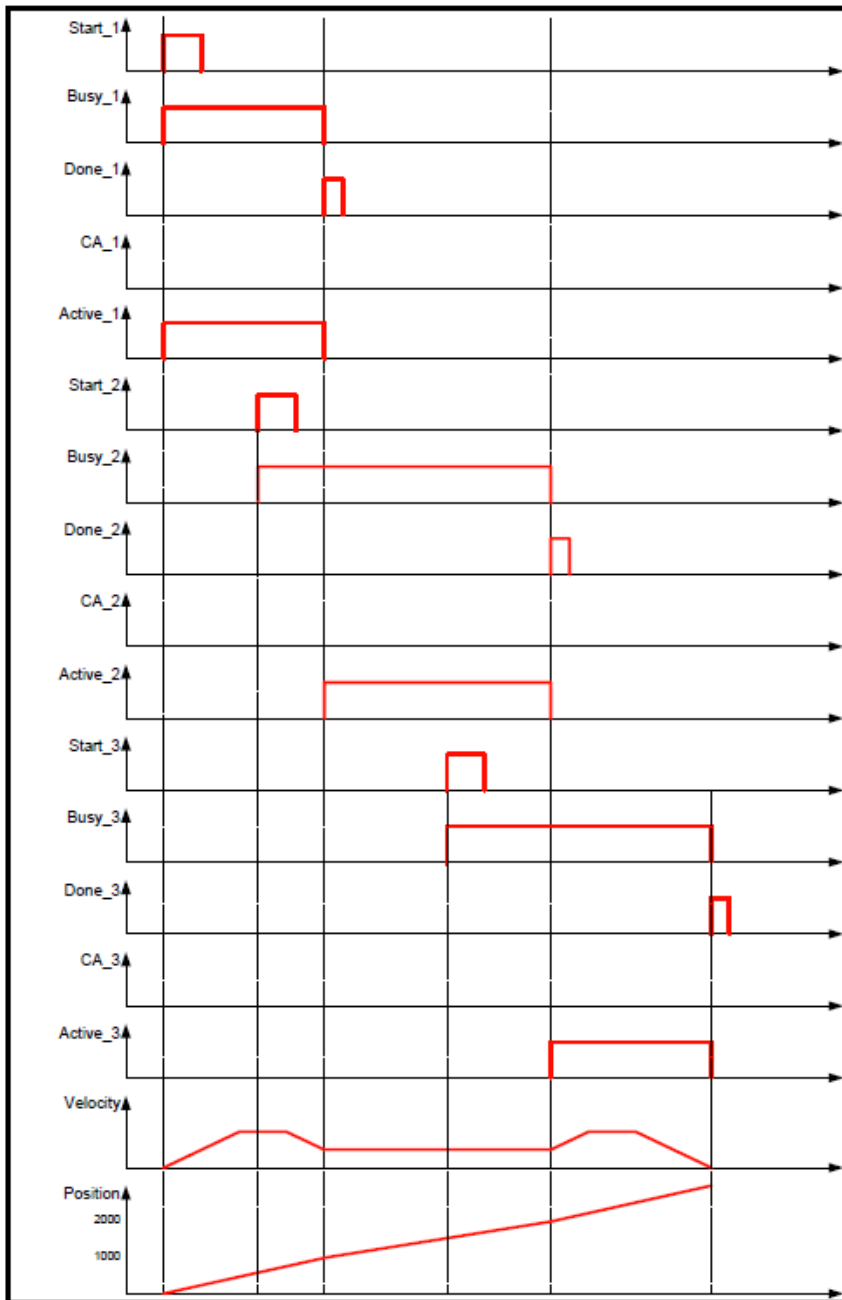


図 3-4-1-4. 上の例の BlendingLow Mode でのタイムチャート  
 (FB1 の最終位置から FB2 の最終位置まで低い方の速度 (velocity2) を使用)

● BlendingPrevious mode での動作

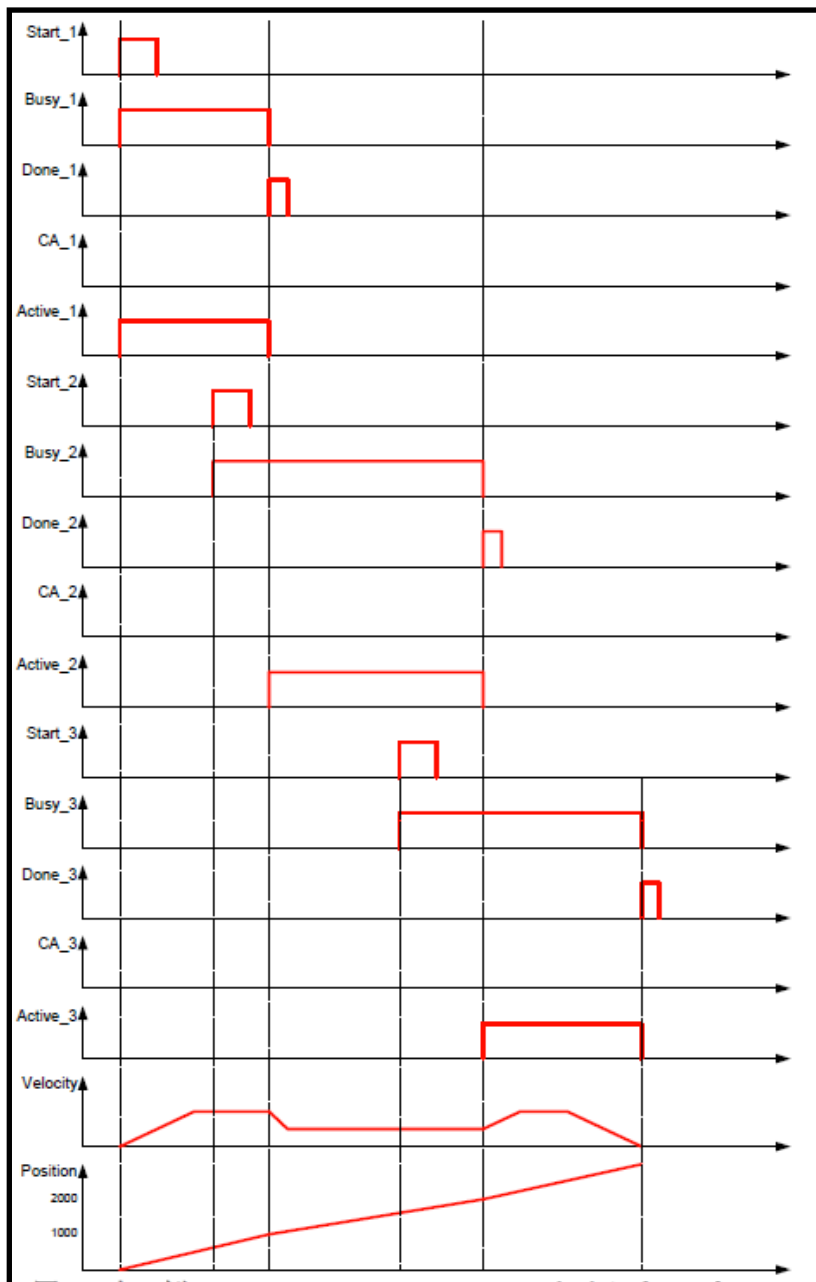
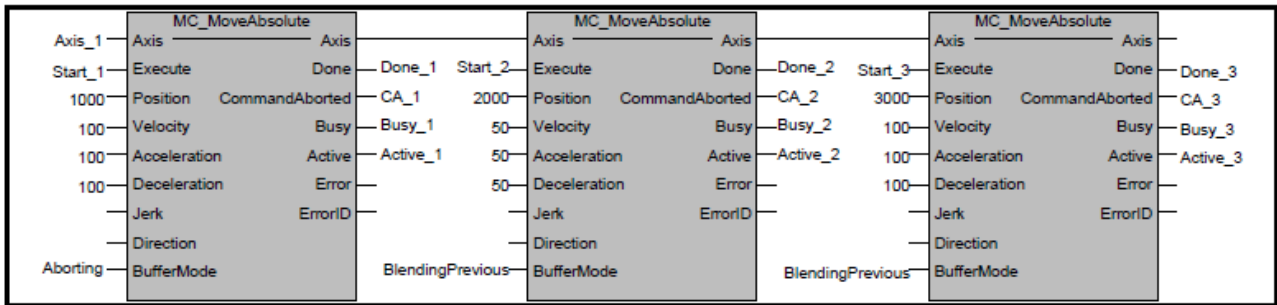


図 3-4-1-5. 上の例の BlendingPrevious mode でのタイムチャート  
(FB1 の最終位置では、FB1 の速度を使用)

● BlendingNext mode での動作

MC_MoveAbsolute				MC_MoveAbsolute				MC_MoveAbsolute			
Axis_1	Axis	Axis	Axis	Axis	Axis	Axis	Axis	Axis	Axis	Axis	Axis
Start_1	Execute	Done	Done_1	Start_2	Execute	Done	Done_2	Start_3	Execute	Done	Done_3
1000	Position	CommandAborted	CA_1	2000	Position	CommandAborted	CA_2	3000	Position	CommandAborted	CA_3
100	Velocity	Busy	Busy_1	50	Velocity	Busy	Busy_2	100	Velocity	Busy	Busy_3
100	Acceleration	Active	Active_1	50	Acceleration	Active	Active_2	100	Acceleration	Active	Active_3
100	Deceleration	Error		50	Deceleration	Error		100	Deceleration	Error	
	Jerk	ErrorID			Jerk	ErrorID			Jerk	ErrorID	
	Direction				Direction				Direction		
Aborting	BufferMode		BlendingNext	BufferMode		BlendingNext	BufferMode		BufferMode		BlendingNext

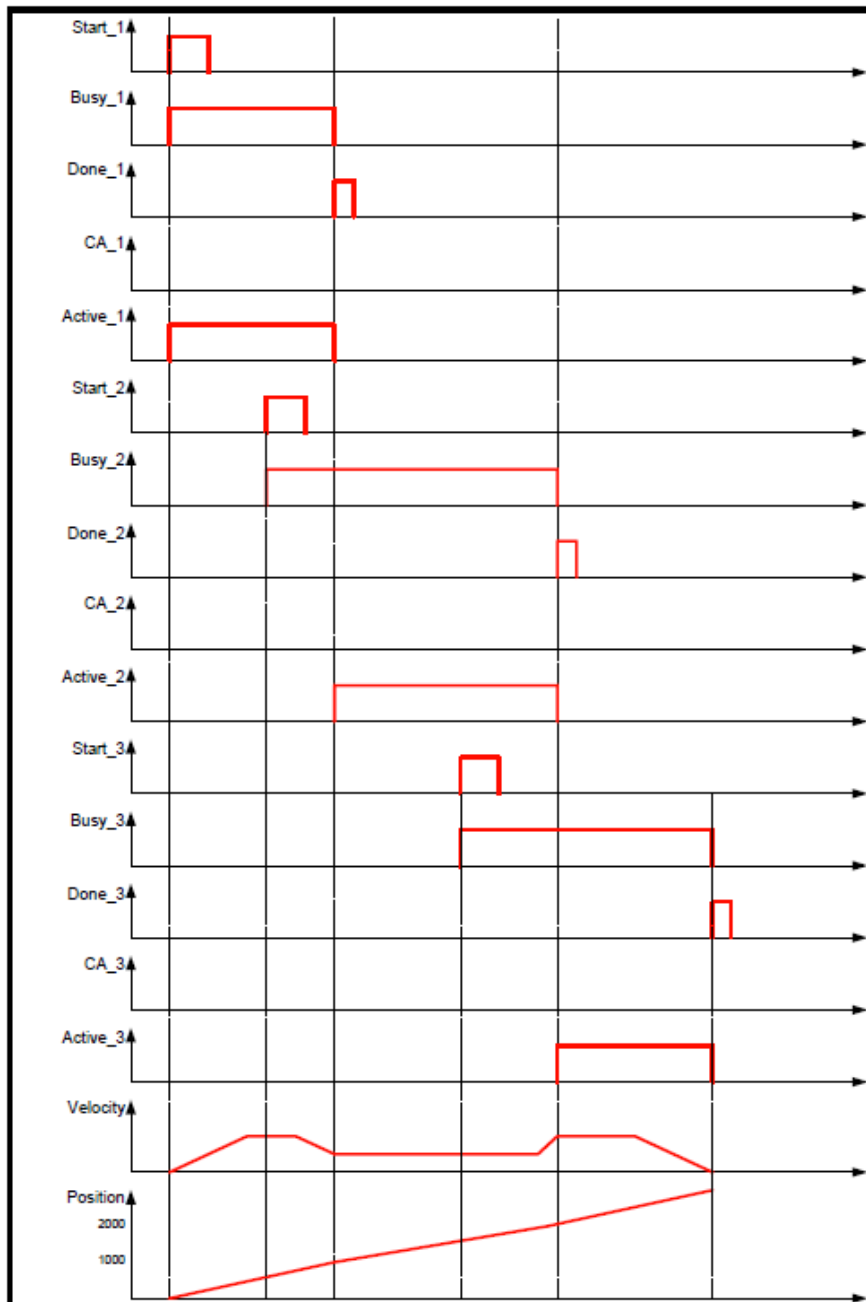


図 3-4-1-6. 上の例の BlendingNext mode でのタイムチャート  
(前回の FB 終了時点では、次の FB の速度を使用)

● BlendingHigh mode での動作

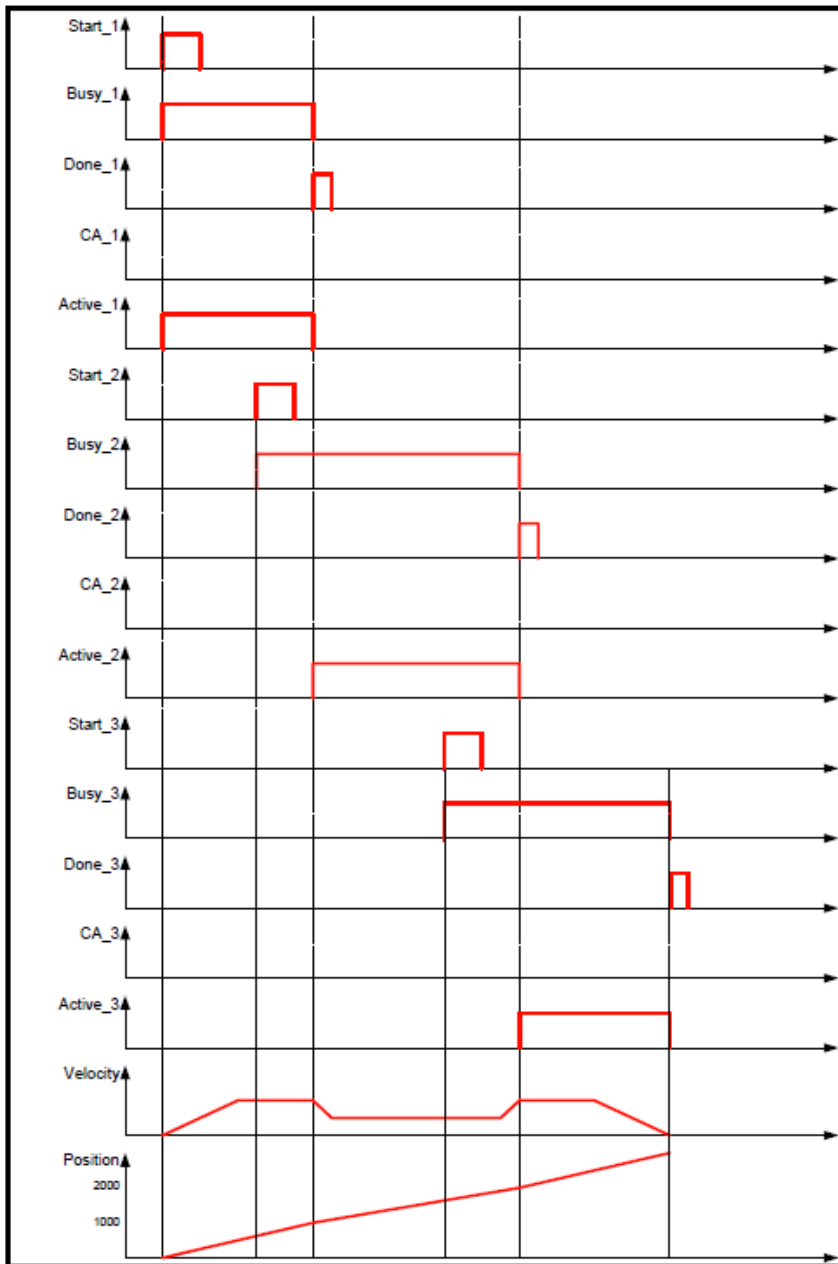
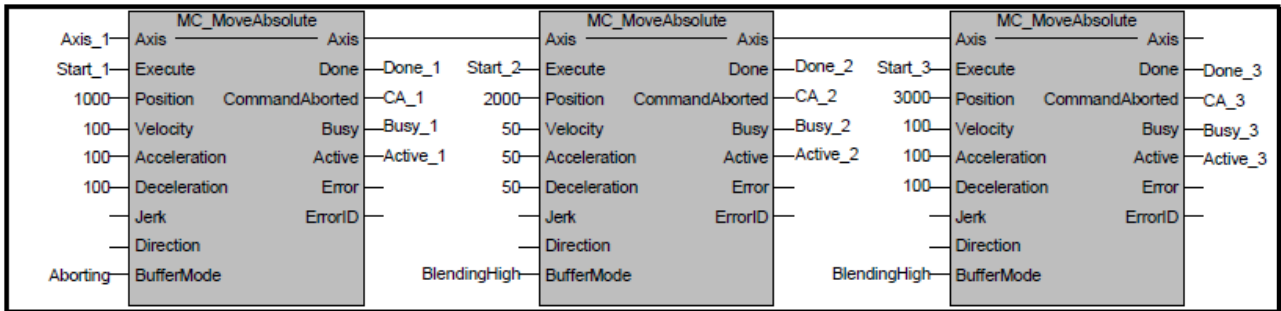


図 3-4-1-7. 上の例の BlendingHigh mode でのタイムチャート  
 (FB1 最終位置では FB1 の速度を使用、FB2 の最終位置では FB3 の速度を使用)

Buffered Modeのうち、ブレンディング系 (BlendingLow、BlendingPrevious、BlendingNext、BlendingHigh) の動作をさせる場合いくつか注意事項があります。

1. ブレンディングさせる動作の動作方向は同一方向としてください。反転動作となる指定をされた場合はエラーとなります。
2. MC\_MoveVelocity の速度制御系の FB から、MC\_MoveAbsolute 等の位置制御系の FB へのブレンディング動作は、すべて Buffered 動作となります。MC\_MoveVelocity の InVelocity が ON した位置からの位置制御となります。  
MC\_MoveVelocity で加減速中に MC\_MoveAbsolute をブレンディングモードで実行した場合で、動作方向が逆転する場合は、InVelocity が ON した後、減速し一度速度が 0 になってから反転動作します。
3. 制御中の FB で、目標位置到達前に速度を変更する場合、指定された加減速度では、目標位置を越える場合は、加減速度値を急激にして目標位置到達時点で次の速度値になるようにします。



### 3-4-2 Part4 動作ファンクションブロックの多重起動

いくつかのファンクションブロックは「BufferMode」と呼ばれる入力を持ちます。この入力により、ファンクションブロックは「Aborting mode」（デフォルト動作）と「Buffered mode」の両方で動作が可能です。これらのモードの相違点は、多重起動を行った際に後から実行した動作の開始タイミングです。

- ・ 非バッファリングモードでのコマンドは、他の動作を中断してでもすぐに動作します。
- ・ バッファリングモードでのコマンドは、現在のファンクションブロックが自身の「Done」出力をセットするまで待ちます。

バッファリングモードには、いくつかのオプションがあります。この入力は MC\_BUFFERMODE の ENUM 型です。表 3-4-2-1 に各バッファリングモードの一覧を示します。

表 3-4-2-1. バッファリングモード一覧

値	モード	内容
0	Aborting	バッファリングしないデフォルトモード。次の FB は、実行中の動作を中断し、コマンドは直ちに実行され、次の FB の動作になります。
1	Buffered	次の FB は、以前の動作が「Done」になると、次の動作を実行します。1度軸は停止します。
2	BlendingLow	次の FB は、以前の FB が完了した後に軸を制御しますが、2つの動作間で軸は停止しません。 合成速度の接続はトランジションモードにより決まります。 弊社 Part4 仕様ではサポートしていません。
3	BlendingPrevious	次の FB は、以前の FB が完了した後に軸を制御しますが、2つの動作間で軸は停止しません。 合成速度の接続はトランジションモードにより決まります。
4	BlendingNext	次の FB は、以前の FB が完了した後に軸を制御しますが、2つの動作間で軸は停止しません。 合成速度の接続はトランジションモードにより決まります。 弊社 Part4 仕様ではサポートしていません。
5	BlendingHigh	次の FB は、以前の FB が完了した後に軸を制御しますが、2つの動作間で軸は停止しません。 合成速度の接続はトランジションモードにより決まります。 弊社 Part4 仕様ではサポートしていません。

また、位置決め動作を伴うファンクションブロックには、BufferMode 適用時の曲線を指定するために「TransitionMode」と「TransitionParameter」という入力を持ちます。

「TransitionParameter」は「TransitionMode」により Parameter の意味が変わります。

表 3-4-2-2. トランジションモード一覧

値	Transition Mode	Transition Parameter	Parameter 内容
0	TMNone	TMNone	TransitionParameter 入力は評価されません。
1	TMStartVelocity	TMStartVelocity	弊社 Part4 仕様ではサポートしていません。
2	TMConstantVelocity	TMConstantVelocity	弊社 Part4 仕様ではサポートしていません。
3	TMCornerDistance	TMCornerDistance	弊社 Part4 仕様ではサポートしていません。
4	TMMaxCornerDeviation	TMMaxCornerDeviation	弊社 Part4 仕様ではサポートしていません。
5~9	PLCopen で予約	—	—
10	TMSmoothing	TMSmoothing	TransitionParameter 入力は評価されません。 内部で自動的に速度合成が実行されます。

トランジションモードとバッファリングモードの対応表を表 3-4-2-3 に示します。

表 3-4-2-3. トランジションモードとバッファリングモード対応表

値	TransitionMode	Aborting	Buffered	BlendingPrevious
0	TMNone	○	○	Buffered の動作
10	TMSmoothing	○	Previous の動作	○

以下の例は、バッファリング・トランジションモードの組み合わせによる動作の相違を記述しています。

- Part4 BufferMode “Aborting”

バッファリングモード「Aborting」では、実行中の動作を即座に停止し、新しい動作を開始します。

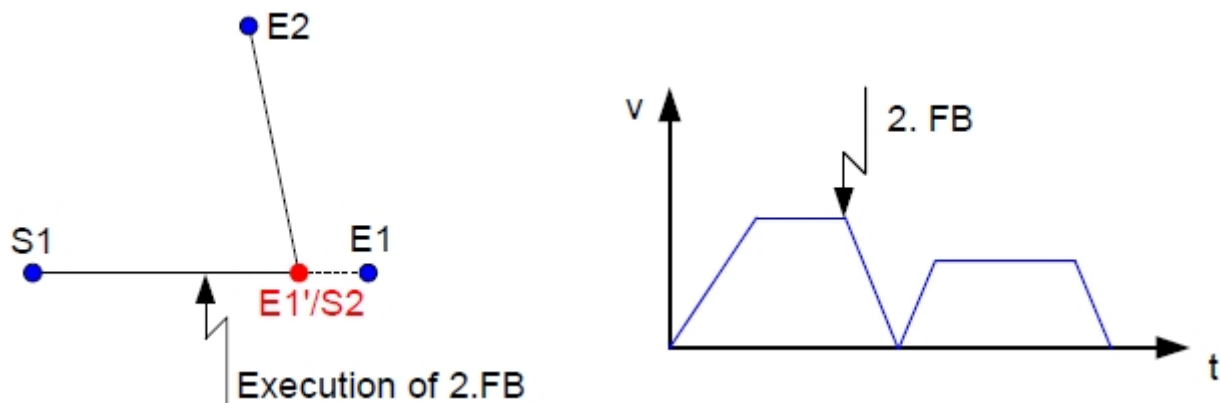


図 3-4-2-1. Aborting 速度合成図

● Part4 BufferMode “Buffered”

バッファリングモード「Buffered」では、実行中の動作完了後に新しい動作を開始します。

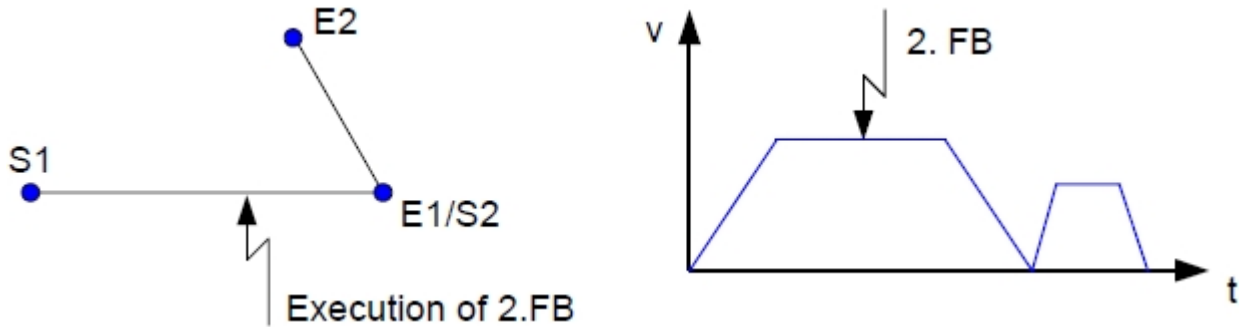


図 3-4-2-2. Buffered 速度合成図

● Part4 BufferMode “BlendingPrevious”

補間機能の設定により、補間の速度合成によるつなぎ動作を行うことができます。つなぎ機能を有効にすると補間のブロック間で停止せずに速度がつながれるため、サイクルタイムの短縮が可能です。直線補間または円弧補間実行中に補間機能のつなぎ機能を有効に設定した直線補間または円弧補間を指令することで、つなぎ動作が行われます。つなぎ動作が行われた場合、最初の補間の動作完了は速度合成終了時に発行されます。

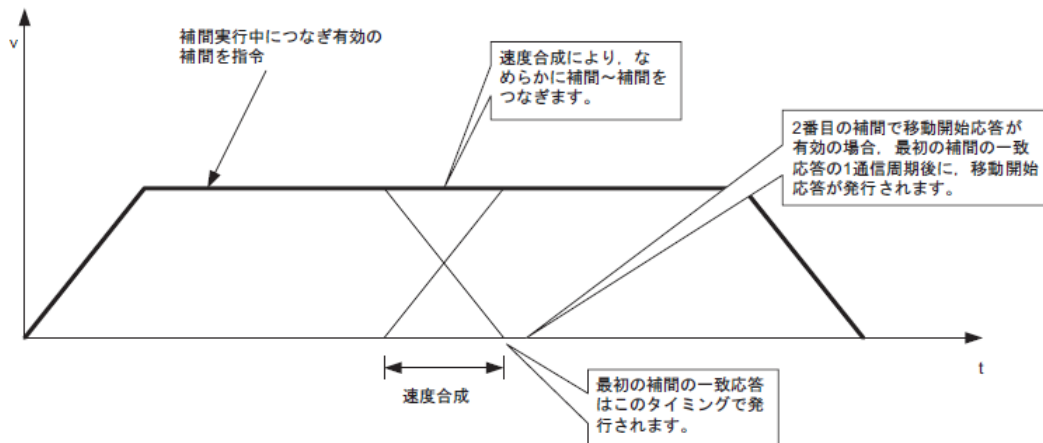


図 3-4-2-3. BlendingPrevious 速度合成図（速度一定、加速度一定）

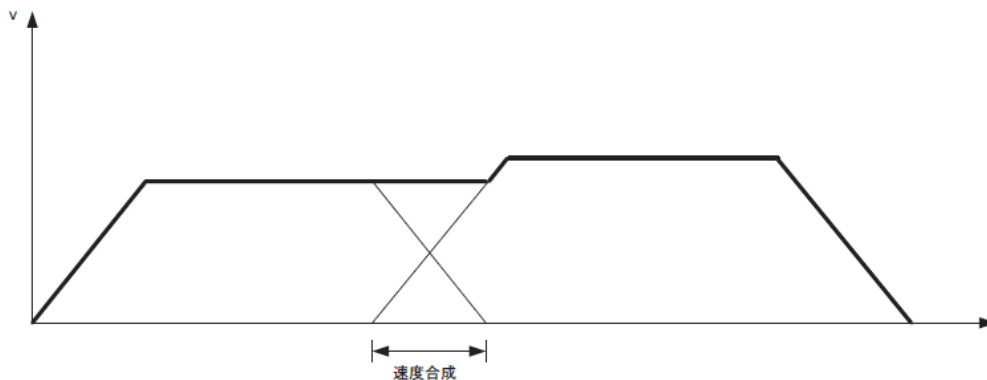


図 3-4-2-4. BlendingPrevious 速度合成図（速度上昇、加速度一定）

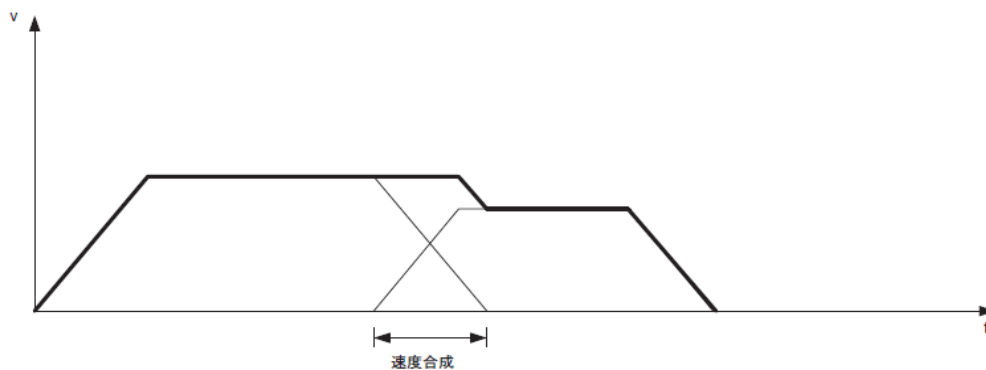


図 3-4-2-5. BlendingPrevious 速度合成図 (速度下降、加速度一定)

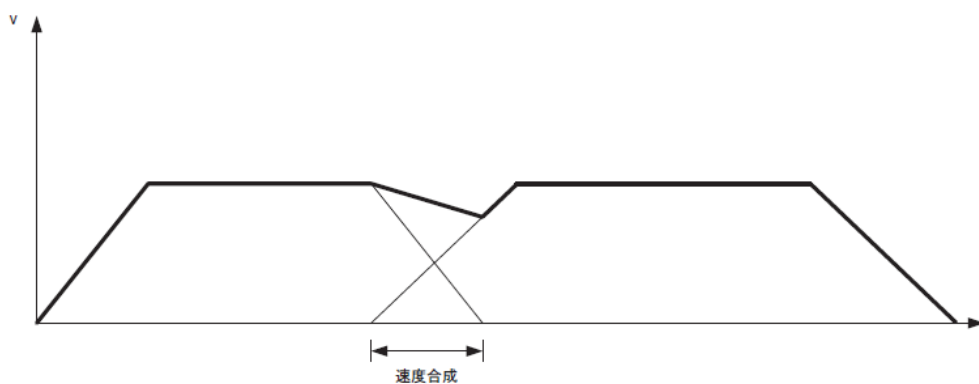


図 3-4-2-6. BlendingPrevious 速度合成図 (速度一定、加速度減少)

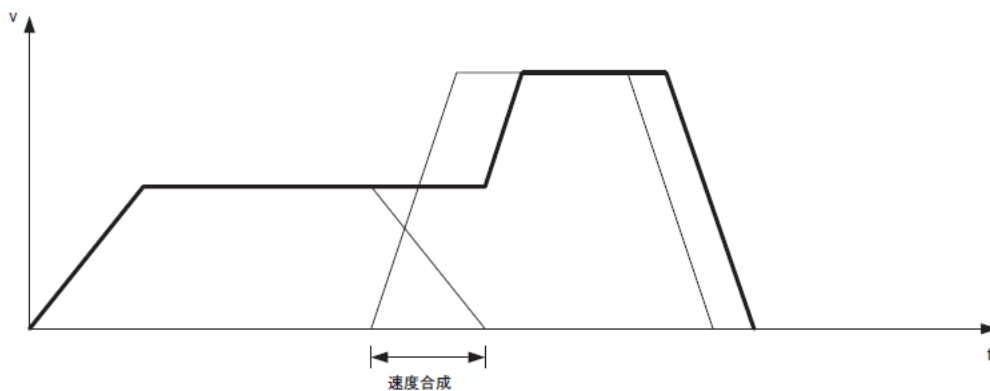


図 3-4-2-7. BlendingPrevious 速度合成図 (速度上昇、加速度上昇)

## 第4章 モーション制御パラメータ

本章では、PLCopen 仕様のモーション制御で使用する軸パラメータについて説明します。

### 4-1 概要

モーション制御パラメータとしては、PLCopen プロセス内で使用している PLCopen パラメータとサーボパックパラメータの2種類があります。

サーボパックのパラメータについては、各メーカーのサーボパックのマニュアルを参照してください。

PLCopen プロセス内で使用しているパラメータについては、ini ファイルで初期値を設定することができます。

各パラメータをリード・ライトするときは、パラメータ型に合った FB を使用してください。パラメータリード・ライトで使用する FB の一覧を表 4-1-1 に示します。

表 4-1-1. パラメータリード・ライトファンクションブロック一覧

FB 名	パラメータ型例	サイズ	内容
MC_ReadParameter	LREAL	8byte	浮動小数点パラメータ読み出し
MC_ReadBoolParameter	BOOL	1bit	BOOL パラメータ読み出し
MC_ReadByteParameter	BYTE SINT USINT INT8 UINT8	1byte	1byte パラメータ読み出し
MC_ReadWordParameter	WORD INT UINT INT16 UINT16	2byte	2byte パラメータ読み出し
MC_ReadDwordParameter	DWORD DINT UDINT INT32 UINT32	4byte	4byte パラメータ読み出し
MC_WriteParameter	LREAL	8byte	浮動小数点パラメータ書き込み
MC_WriteBoolParameter	BOOL	1bit	BOOL パラメータ書き込み
MC_WriteByteParameter	BYTE SINT USINT INT8 UINT8	1byte	1byte パラメータ書き込み
MC_WriteWordParameter	WORD INT UINT INT16 UINT16	2byte	2byte パラメータ書き込み
MC_WriteDwordParameter	DWORD DINT UDINT INT32 UINT32	4byte	4byte パラメータ書き込み

※注：パラメータ型と範囲については、『はじめに 2データタイプ』を参照してください。

## 4-2 PLCopen パラメータ一覧

PLCopen プロセスで定義されているパラメータは、共通パラメータと各軸毎のパラメータがあります。

### 4-2-1 共通パラメータ

型名	TPOPEN_SMEM_CONTROL		説明	共通管理領域	
メンバ名	型	パラメータNo	説明		
UseAxis	UINT32	0x00000001	制御軸数	範囲	1~62
				初期値	1
EtherCATAddrKind	UINT16	0x00000002	0 : EtherCATスレーブはノードアドレスで管理 1 : EtherCATスレーブはDipSwで管理 <b>※注 : AI-Motionでは使用しません。</b>	範囲	0 or 1
				初期値	0
UseGroup	UINT32	0x00000003	制御グループ数	範囲	1~31
				初期値	0

#### ● MECHATROLINK-Ⅲの場合

MECHATROLINK-Ⅲ通信を使った PLCopen 仕様のスレーブ指定方法は下記ようになります。

- ・ DipSw と TechnoML3Setting.ini 設定ファイルで管理された論理 ID

MECHATROLINK-Ⅲ通信の場合は、上記表中の「EtherCATAddrKind」は設定不要です。

#### ● EtherCAT の場合

EtherCAT 通信を使った PLCopen 仕様のスレーブ指定方法は下記ようになります。

- ・ DipSw と TechnoECTSetting.ini 設定ファイルで管理された論理 ID

AI-Motion での EtherCAT 通信の場合は、上記表中の「EtherCATAddrKind」は設定不要です。

## 4-2-2 軸毎パラメータ

型名	AXIS_REF_CFG		説明	軸基本設定			
メンバ名	型	パラメータNo	説明				
AxisEnable	UINT16	0x00000100	0: この軸は使用しない 1: この軸は使用する	範囲	0 or 1		
				初期値	0		
AxisType	UINT16	0x00000101	この軸で使用するユニットタイプを指定する			範囲	0x00 ~ 0x20
						初期値	0
			値	タイプ			
			0x00	EtherCAT : サーボ			
			0x01	EtherCAT : ALGOSYSTEM モーションコントロールユニット			
			0x02	EtherCAT : YASKAWA Σ-5シリーズ			
			0x03	EtherCAT : SANYO SANMORTION-Rシリーズ			
			0x04	EtherCAT : KOLLMORGEN AKDシリーズ			
			0x05	EtherCAT : OMRON シリーズ			
			0x06	EtherCAT : PANASONIC シリーズ			
			0x07- 0x0F	EtherCAT : リザーブ			
			0x10	MECHATROLINK-Ⅲ通信サーボパック			
			0x11- 0x1F	MECHATROLINK-Ⅲ : リザーブ			
0x20	仮想軸						
NodeAddr	UINT16	0x00000102	EtherCATの場合 : TechnoECTSetting. ini設定ファイルで設定した論理ID MECHATROLINK-Ⅲの場合 : TechnoML3Setting. ini設定ファイルで設定した論理ID	範囲	—		
				初期値	—		
NodeSubCh	UINT16	0x00000103	1ノードで複数軸動作できるユニットの場合の内部チャンネル指定	範囲	0~7		
				初期値	0		

これらの値は、PLCopen プロセスが起動する前に設定されている必要があります。そのため、ini ファイルにより初期設定値を設定できるようにしてあります。これらの値を変更された後は、INTime のノードを再起動してください。

ini ファイル設定方法は、『4-3 ini ファイルによるパラメータ初期値設定方法』を参照ください。

ini ファイルでは AxisType に 0x00 (EtherCAT) か、0x10 (MECHATROLINK-Ⅲ) か、0x20 (仮想軸) かを選択してください。EtherCAT のサーボパックメーカーの判別は初期化時に自動で行います。初期化後、AxisType パラメータを読み出すことで、メーカーを判別できます。

型名	AXIS_REF_SCALE		説明	単位変換設定	
メンバ名	型	パラメータNo	説明		
Num	UINT32	0x00000200	モータ1回転のパルス数	範囲	0~ 4294967295 【pulse】
				初期値	0
Den	LREAL	0x00000201	モータ1回転の移動量	範囲	実数値 【指令単位】
				初期値	0
Units	UINT16	0x00000202	指令単位を指定。	範囲	0~5
				初期値	0
			値	指令単位	
			0	パルス【pulse】	
			1	ミリメートル【mm】	
			2	マイクロメートル【um】	
			3	ナノメートル【nm】	
4	度【degree】				
5	インチ【inch】				

モーション制御で使用する際、指令単位とパルス単位の間係を設定するために電子ギアを使用します。電子ギア比の計算は下記のようになります。

- 電子ギア比（単位変換の式）

指令位置【pulse】 = 指令位置【指令単位】 × 電子ギア比

電子ギア比 = モータ1回転のパルス数【pulse】 / モータ1回転の移動量【指令単位】

モーション制御命令では目標位置や速度、加減速度をLREAL型で指定しますが、電子ギア比を使って、パルス単位系に変換しています。変換した後の値がサーボパックで設定できる範囲を超える場合は、命令で異常が発生します。

#### 【設定例】

モータ1回転のパルス数 = 1048576【pulse】

ボールネジピッチ = 6【mm】

減速比 = 1/3（モータ1回転でボールネジは1/3回転する）

モータ1回転の移動量 = ボールネジピッチ × 減速比 = 6 × 1/3 = 2【mm】

上記の構成の場合、設定値は下記のようになります。

Num = 1048576 Den = 2 Units = 1

この設定で、モーション制御命令での指令単位は1【mm】となります。123.4【mm】の位置へ絶対位置移動するときは、MC\_MoveAbsoluteのPositionに123.4を設定します。

上記の例で減速比が1/7の場合、モータ1回転の移動量は  $6 \times 1/7 = 0.857142857\cdots$ 【mm】となります。この場合、モータ1回転の移動量の設定値を四捨五入したりすると誤差が発生し、目的の位置になりません。モータ1回転の移動量が割り切れない場合は、モータ1回転のパルス数とモータ1回転の移動量に同じ係数を掛けた値を設定します。

Num = 1048576 × 7 = 7340032 Den = 6 × 1/7 × 7 = 6 Units = 1



型名	AXIS_REF_MOVE		説明	軸動作設定	
メンバ名	型	パラメータNo	説明		
MaxSpeed	LREAL	0x00000300	最高速度設定 最高速度設定値を超える速度を設定された場合、この設定速度で動作します。	範囲	実数正数値 【指令単位/s】
				初期値	40000000
FollowingErrorWindow	LREAL	0x00000301	位置偏差カウンタオーバーフロー値 位置要求値に相対的に許容可能な位置範囲を設定します。	範囲	実数値正数 【指令単位】
				初期値	5000000
PositionWindow	LREAL	0x00000302	位置決め完了範囲 ターゲット位置到達として許容可能な範囲を設定します。	範囲	実数値正数、0 【指令単位】
				初期値	100
VelocityWindow	LREAL	0x00000303	速度到達範囲 ターゲット速度到達として許容可能な範囲を設定します。	範囲	実数値正数 【指令単位/s】
				初期値	50
HomeOffset	LREAL	0x00000304	ホームオフセット メカ原点位置をホームオフセット値で正規化します。	範囲	実数値 【指令単位】
				初期値	0
CwSoftLimit	LREAL	0x00000305	正方向ソフトリミット値 正方向側のソフトリミット値を設定	範囲	実数値 【指令単位】
				初期値	2147483647
CcwSoftLimit	LREAL	0x00000306	負方向ソフトリミット値 負方向側のソフトリミット値を設定	範囲	実数値 【指令単位】
				初期値	-2147483648
ZeroSearchSpeed	LREAL	0x00000307	原点復帰ゼロ点サーチスピード 原点復帰時の低速スピードを設定	範囲	実数値正数 【指令単位/s】
				初期値	100000
HomeAcceleration	LREAL	0x00000308	原点復帰加減速度 原点復帰時の加減速度を設定	範囲	実数値正数 【指令単位/s】
				初期値	10000000
PositionReadProfileNo	UINT16	0x00000309	ポジション読み出しプロファイル番号 ブレンディングモード時の計算時に位置監視の対象パラメータとして使用。 ReadActualPosition時にも使用	範囲	0x6063 or 0x6064
				初期値	0x6064
VelocityReadProfileNo	UINT16	0x0000030A	速度読み出しプロファイル番号 ブレンディングモード時の計算時に速度監視の対象パラメータとして使用。 ReadActualVelocity時にも使用	範囲	0x606B or 0x606C
				初期値	0x606B
TorqueWindow	LREAL	0x0000030B	トルク到達範囲 ターゲットトルク到達として許容可能な範囲を設定します。	範囲	実数値正数 【N.m】 or 【%】
				初期値	50

型名	AXIS_REF_COUNT		説明	軸カウンタ設定	
メンバ名	型	パラメータNo	説明		
CountMode	UINT16	0x00000400	カウンタモード 0:リニアモード(有限長) 1:ロータリモード(無限長)	範囲	0 or 1
				初期値	0
MinPosLimit	LREAL	0x00000401	リングカウンタ下限設定値 カウンタモードロータリモードとしたときのリングカウンタ下限値を設定	範囲	実数値 【指令単位】
				初期値	-2147483648
MaxPosLimit	LREAL	0x00000402	リングカウンタ上限設定値 カウンタモードロータリモードとしたときのリングカウンタ上限値を設定	範囲	実数値 【指令単位】
				初期値	2147483647

型名	AXIS_REF_STATUS		説明	軸ステータス																																					
メンバ名	型	パラメータNo	説明																																						
CtrlStatus	UINT16	0x00010000	<table border="1"> <thead> <tr> <th>BIT</th> <th>名称</th> <th>内容</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>原点確定</td> <td>0:原点未確定 1:原点確定</td> </tr> <tr> <td>1</td> <td>指令速度飽和</td> <td>0:指令速度は無制限 1:指令速度が最大速度で制限中</td> </tr> <tr> <td>2</td> <td>動作方向</td> <td>0:正方向 1:負方向</td> </tr> <tr> <td>3</td> <td>サーボReady</td> <td>0:サーボ準備未完 1:サーボ準備完了</td> </tr> <tr> <td>4</td> <td>主回路電源</td> <td>0:主回路電源OFF状態 1:主回路電源ON状態</td> </tr> <tr> <td>5</td> <td>サーボON</td> <td>0:サーボOFF 1:サーボON</td> </tr> <tr> <td>6</td> <td>+EL</td> <td>0:正方向エンドリミットOFF 1:正方向エンドリミットON</td> </tr> <tr> <td>7</td> <td>-EL</td> <td>0:負方向エンドリミットOFF 1:負方向エンドリミットON</td> </tr> <tr> <td>8</td> <td>ORG</td> <td>0:原点信号OFF 1:原点信号ON</td> </tr> <tr> <td>9</td> <td>ドライブアラーム</td> <td>0:ドライブアラーム無し 1:ドライブアラーム発生中</td> </tr> <tr> <td>10</td> <td>ドライブワーニング</td> <td>0:ドライブワーニング無し 1:ドライブワーニング発生中</td> </tr> </tbody> </table>	BIT	名称	内容	0	原点確定	0:原点未確定 1:原点確定	1	指令速度飽和	0:指令速度は無制限 1:指令速度が最大速度で制限中	2	動作方向	0:正方向 1:負方向	3	サーボReady	0:サーボ準備未完 1:サーボ準備完了	4	主回路電源	0:主回路電源OFF状態 1:主回路電源ON状態	5	サーボON	0:サーボOFF 1:サーボON	6	+EL	0:正方向エンドリミットOFF 1:正方向エンドリミットON	7	-EL	0:負方向エンドリミットOFF 1:負方向エンドリミットON	8	ORG	0:原点信号OFF 1:原点信号ON	9	ドライブアラーム	0:ドライブアラーム無し 1:ドライブアラーム発生中	10	ドライブワーニング	0:ドライブワーニング無し 1:ドライブワーニング発生中	範囲	—
				BIT	名称	内容																																			
				0	原点確定	0:原点未確定 1:原点確定																																			
				1	指令速度飽和	0:指令速度は無制限 1:指令速度が最大速度で制限中																																			
				2	動作方向	0:正方向 1:負方向																																			
				3	サーボReady	0:サーボ準備未完 1:サーボ準備完了																																			
				4	主回路電源	0:主回路電源OFF状態 1:主回路電源ON状態																																			
				5	サーボON	0:サーボOFF 1:サーボON																																			
				6	+EL	0:正方向エンドリミットOFF 1:正方向エンドリミットON																																			
				7	-EL	0:負方向エンドリミットOFF 1:負方向エンドリミットON																																			
				8	ORG	0:原点信号OFF 1:原点信号ON																																			
				9	ドライブアラーム	0:ドライブアラーム無し 1:ドライブアラーム発生中																																			
				10	ドライブワーニング	0:ドライブワーニング無し 1:ドライブワーニング発生中																																			
初期値	—																																								

### 4-2-3 サーボパックパラメータ

本項では、サーボパックで規定されているパラメータをアクセスする方法について説明します。

● MECHATROLINK-Ⅲの場合

MECHATROLINK-Ⅲ通信のサーボパックでは、共通パラメータ／機器パラメータと呼ばれるパラメータが存在し、これらのパラメータはRAM領域／不揮発メモリ領域を選択してR/Wする事が出来ます。

PLCopen仕様MCファンクションブロックのMC\_ReadParameterやMC\_WriteParameter等のリード・ライト系のFBでは下記のようなパラメータ番号を入力することで、それぞれのサイズ毎にパラメータの読み書きを行います。

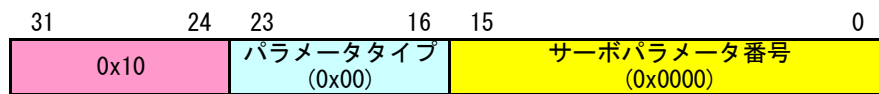


図 4-2-3-1. R/Wパラメータ番号

パラメータタイプは以下のタイプがあります。

表 4-2-3-1. サーボパラメータタイプ

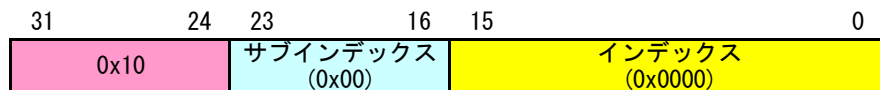
共通パラメータ	RAM領域	0x00
	不揮発メモリ領域	0x01
機器パラメータ	RAM領域	0x10
	不揮発メモリ領域	0x11

共通パラメータ／機器パラメータの詳細については、各サーボパックのマニュアルを参照してください。

- EtherCAT の場合

EtherCAT 通信のサーボパックでは、CiA402 で規定されているパラメータ (0x6000 番台) の他に、メーカー独自のパラメータ等が規定されています。

EtherCAT の場合は、インデックス番号 (16bit) + サブインデックス番号 (8bit) でアクセスすることができます。MC\_ReadParameter や MC\_WriteParameter 等のリード・ライト系の FB では下記のようなパラメータ番号を入力することで、それぞれのサイズ毎にパラメータの読み書きを行います。

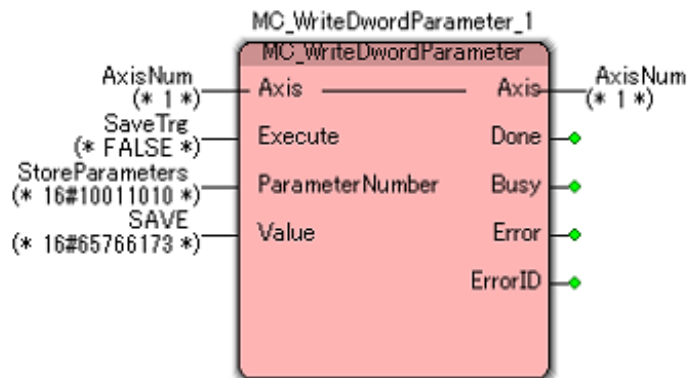


- 設定例

全サーボパラメータを EEPROM へ保存する場合

インデックス番号 = 0x1010

サブインデックス番号 = 0x01



SaveTrg を True にすると、サーボパックに設定されているすべてのパラメータが EEPROM へ保存されます。インデックス番号 (0x1010) の使用方法については、各メーカーのサーボパックマニュアルを参照してください。

### 4-3 iniファイルによるパラメータ初期値設定方法

P0penSetting.ini ファイルは「INtime 版 PLCopen プロセス PLCOpenProc. RTA」を使用する際に必要な設定ファイルです。

本設定ファイルにより「INtime 版 PLCopen プロセス PLCOpenProc. RTA」で使用する軸数や、各軸のタイプ設定を変更することができます。本項では設定ファイル P0penSetting.ini の設定法について解説します。

設定ファイル P0penSetting.ini の構成図は図 4-3-1 のようになります。

TechnoML3Setting.ini ファイルは「MECHATROLINK-Ⅲ用テクノモーションプロセス RtpIML3Proc. RTA」を使用する際に必要な設定ファイルです。本設定ファイルにより、MECHATROLINK-Ⅲの通信構成を構築し、各種通信設定を変更して動作させることができます

TechnoECTSetting.ini ファイルは「EtherCAT 用テクノモーションプロセス RtpIECTProc. RTA」を使用する際に必要な設定ファイルです。本設定ファイルにより、EtherCAT の通信構成を構築し、各種通信設定を変更して動作させることができます

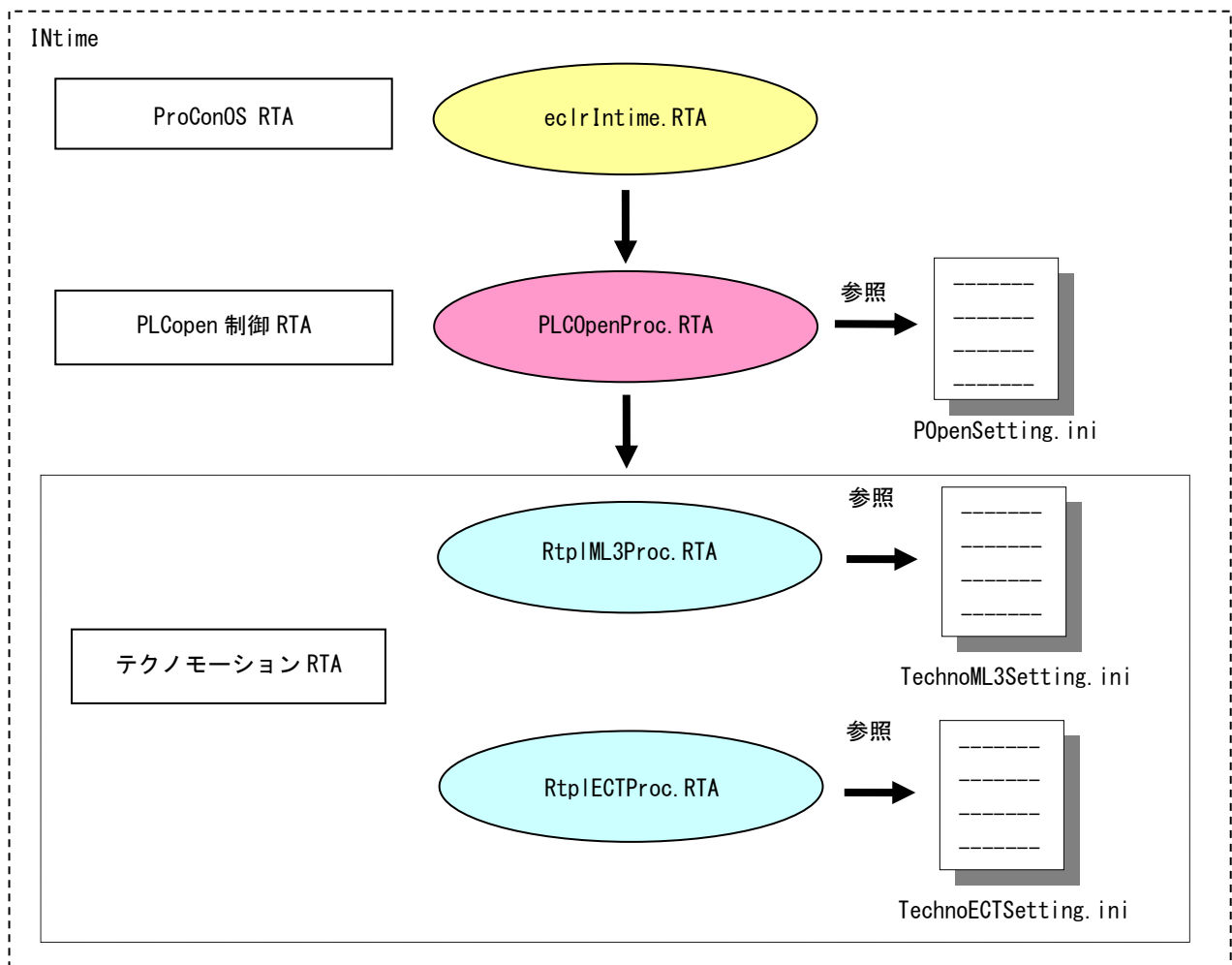


図 4-3-1 INtime 版 PLCopen プロセス構成図

## 4-3-1 POpenSetting.ini ファイル

PLCOpenProc.rta は「POpenSetting.ini」ファイルを設定することによって軸タイプおよび通信設定を変更して動作させることができます。I/O ユニットの制御は ProConOS にて直接行いますので、あくまで、制御軸数を設定してください。

設定する項目は表 4-3-1-1 のようになります。

表 4-3-1-1 INI ファイル項目

セクション名	キー名	備 考
CONTROL	UseAxis	TPOPEN_SMEM_CONTROL パラメータ参照
	EtherCATAddrKind	
	UseGroup	
AXIS_n n : 1, 2, ...62 (軸番号)	RefCfg_AxisEnable	AXIS_REF_CFG パラメータ参照
	RefCfg_AxisType	
	RefCfg_NodeAddr	
	RefCfg_NodeSubCh	
	RefScale_Num	AXIS_REF_SCALE パラメータ参照
	RefScale_Den	
	RefScale_Units	
	RefScale_A_Units	
	RefMove_MaxSpeed	AXIS_REF_MOVE パラメータ参照
	RefMove_FollowingErrorWindow	
	RefMove_PositionWindow	
	RefMove_VelocityWindow	
	RefMove_HomeOffset	
	RefMove_CwSoftLimit	
	RefMove_CcwSoftLimit	
	RefMove_ZeroSearchSpeed	
	RefMove_HomeAcceleration	
	RefMove_PositionReadProfileNo	
	RefMove_VelocityReadProfileNo	
	RefCount_CountMode	
RefCount_MinPosLimit		
RefCount_MaxPosLimit		

POpenSetting.ini ファイルの例をリスト 4-3-1-1 に示します。

リスト 4-3-1-1 INI ファイル例

```
[CONTROL]
UseAxis=2
UseGroup=1
EtherCATAddrKind=0

[AXIS_1]
RefCfg_AxisEnable=0
```

```
RefCfg_AxisType=0
RefCfg_NodeAddr=1001
RefCfg_NodeSubCh=0
RefScale_Num=1
RefScale_Den=1
RefScale_Units=0
RefScale_A_Units=1
RefMove_MaxSpeed=40000000
RefMove_FollowingErrorWindow=5000000
RefMove_PositionWindow=100
RefMove_VelocityWindow=50
RefMove_HomeOffset=0
RefMove_CwSoftLimit=0
RefMove_CcwSoftLimit=0
RefMove_ZeroSearchSpeed=100000
RefMove_HomeAcceleration=10000000
RefMove_PositionReadProfileNo=24676
RefMove_VelocityReadProfileNo=24683
RefCount_CountMode=0
RefCount_MinPosLimit=-2147483648
RefCount_MaxPosLimit=2147483647
```

[AXIS\_2]

```
RefCfg_AxisEnable=0
RefCfg_AxisType=0
RefCfg_NodeAddr=1002
RefCfg_NodeSubCh=0
RefScale_Num=1
RefScale_Den=1
RefScale_Units=0
RefScale_A_Units=1
RefMove_MaxSpeed=40000000
RefMove_FollowingErrorWindow=5000000
RefMove_PositionWindow=100
RefMove_VelocityWindow=50
RefMove_HomeOffset=0
RefMove_CwSoftLimit=0
RefMove_CcwSoftLimit=0
RefMove_ZeroSearchSpeed=100000
RefMove_HomeAcceleration=10000000
RefMove_PositionReadProfileNo=24676
RefMove_VelocityReadProfileNo=24683
RefCount_CountMode=0
RefCount_MinPosLimit=-2147483648
RefCount_MaxPosLimit=2147483647
```

値はすべて 10 進数で指定してください。16 進数での指定は無効です。

## 4-3-2 TechnoML3Setting.ini ファイル

RTPLML3PROC.RSL は「TechnoML3Setting.ini」ファイルを設定することによって MECHATROLINK-III 通信設定を行い、スレーブのタイプを設定することができます。

サーボパックだけでなく、I/O ユニットを使用する場合もこちらで設定します。PLCOpen MC として制御するのはサーボパックのみです。I/O ユニットについては、ProConOS により直接制御します。

設定する項目は表 4-3-2-1、表 4-3-2-2 のようになります。

表 4-3-2-1 INI ファイル項目 (マスタ設定)

セクション名	キー名	タイプ	備考
SYSTEM	CommMode	16 進数	これらの設定値の詳細については、下記を参照してください。
	MaxSlave	10 進数	
	CycleTime	10 進数	
	IntOffset	10 進数	
	C2MstDly	10 進数	
	CommProtocol	10 進数	
	MaxRetry	10 進数	
	HostWdt	10 進数	

**CommMode** : 通信モード設定 (初期値 : 0x8002)

CommMode (論理和)	内容
SYS_MOD_TYPE_C1MST *1	0x00000002 : 動作タイプ C1MST
SYS_MOD_INTLV_PLS	0x00000800 : INTOL 信号をパルス出力設定
SYS_MOD_ESYNC	0x00001000 : RTCIL 信号入力に同期
SYS_MOD_INT_FR *1	0x00008000 : ハード同期有効

\*1 必ず指定する必要があります。

**MaxSlave** : 最大接続スレーブ数 (初期値 : 1)  
[ 1 ~ 62 ] 単位 : 【局】

**CycleTime** : 伝送周期 (初期値 : 50000)  
[ 3125(31.25us) ~ 6400000(64ms) ] 単位 : 【10ns】

**IntOffset** : 割込遅延時間 (初期値 : 25000)  
[ 0 ~ 伝送周期設定値-500(5us) ] 単位 : 【10ns】

**C2MstDly** : C2 マスタ送信開始時間 (初期値 : 0)  
[ 0 : C2 マスタ使用しない ]  
[ 1 ~ 伝送周期設定値-500(5us) ] 単位 : 【10ns】

**CommProtocol** : 通信プロトコル選択 (初期値 : 0)  
[ 0 : サイクリック通信 ]  
[ 1 : イベントドリブン通信 ]

**MaxRetry** : 最大リトライ回数 (初期値 : 1)  
[ 0 ~ 62 ] 単位 : 【回】

**HostWdt** : ホスト監視用 WDT 設定 (初期値 : 16384)  
[ 0 : 機能無効 ]  
[ 1(8us) ~ 16384(131072us) ] 単位 : 【8us】



表 4-3-2-2 INI ファイル項目 (スレーブ設定)

セクション名	キー名	タイプ	備考
SLAVE_n (n = 1 ~ 62)	SlaveID	10 進数	これらの設定値の詳細については、下記を参照してください。
	Cd_Rd_Len	10 進数	
	ResponseTime	10 進数	
	Com_Mod	16 進数	
	Com_Tim	10 進数	
	Profile_Type	16 進数	

**SlaveID** : 局アドレス (初期値 : 3)  
 [ 0 : スレーブなし ]  
 拡張アドレス (上位バイト) [ 0x00 ~ 0x3D ]  
 局アドレス (下位バイト) [ 0x03 ~ 0xEF ]

**Cd\_Rd\_Len** : 送受信データ長 (初期値 : 48)  
 [ 8/16/32/48/64 ] 単位【Byte】

**ResponseTime** : 応答監視時間 (初期値 : 5000)  
 [ 500(5us) ~ 伝送周期設定値 ] 単位【10ns】

**Com\_Mod** : 通信モード (初期値 : 0x82)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SUBCMD	0	0	0	0	0	SYNCMODE	0

SYNCMODE:同期設定 [ 0:非同期通信 1:同期通信 ]

SUBCMD:サブコマンド設定 [ 0:サブコマンド無効 1:サブコマンド有効 ]

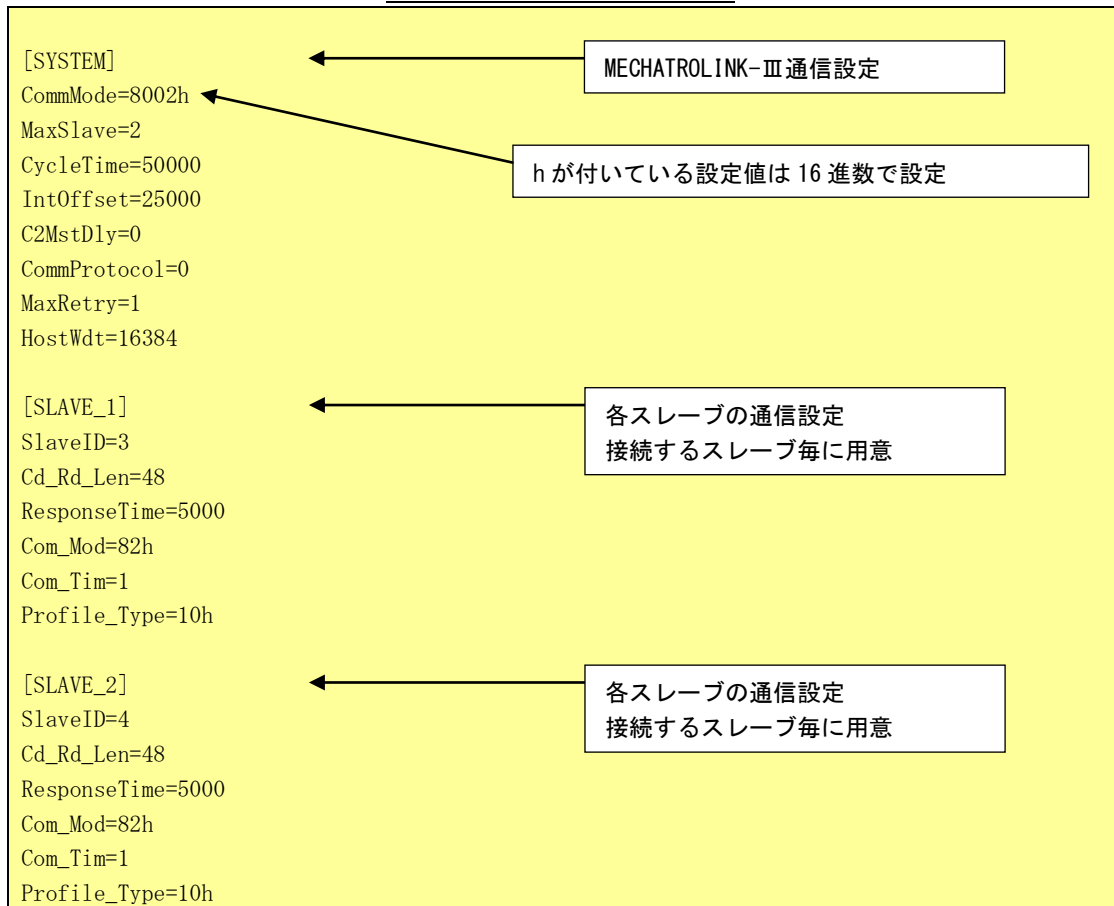
**Com\_Tim** : 通信周期 (通信周期 = 設定値 × 伝送周期) (初期値 : 1)  
 [ 1 ~ 255 ] 単位【倍】

**Profile\_Type** : プロファイルタイプ

Profile_Type	内容
SLV_PROFILETYPE_SRV	0x10 : サーボパックプロファイルタイプ (初期値)
SLV_PROFILETYPE_IO	0x30 : I/O プロファイルタイプ

TechnoML3Setting.ini ファイルの例をリスト 4-3-2-1 に示します。

リスト 4-3-2-1 INI ファイル例



## 4-3-3 TechnoECTSetting.ini ファイル

RTPLECTPROC.RSL は「TechnoECTSetting.ini」ファイルを設定することによって EtherCAT 通信設定を行い、スレーブのタイプを設定することができます。

サーボパックだけでなく、I/O ユニットを使用する場合もこちらで設定します。PLCOpen MC として制御するのはサーボパックのみです。I/O ユニットについては、ProConOS により直接制御します。

設定する項目は表 4-3-3-1、表 4-3-3-2 のようになります。

表 4-3-3-1 INI ファイル項目 (マスタ設定)

セクション名	キー名	タイプ	備考
SYSTEM	MaxSlave	10 進数	これらの設定値の詳細については、下記を参照してください。

**MaxSlave** : 最大接続スレーブ数 (初期値 : 1)  
[ 1 ~ 62 ] 単位 : 【局】

表 4-3-3-2 INI ファイル項目 (スレーブ設定)

セクション名	キー名	タイプ	備考
SLAVE_n (n = 1 ~ 62)	StationAlias	10 進数	これらの設定値の詳細については、下記を参照してください。
	SlaveType	10 進数	

**StationAlias** : 局アドレス (初期値 : 1)  
[ 0 : スレーブなし ]  
[ 0x01 ~ 0xEF ]

**SlaveType** : スレーブタイプ (初期値 : 0)  
[ 0 : CiA402 準拠サーボパック ]  
[ 0 以外 : その他の EtherCAT スレーブ ]

TechnoECTSetting.ini ファイルの例をリスト 4-3-3-1 に示します。

リスト 4-3-3-1 INI ファイル例

[SYSTEM] MaxSlave=3	←	EtherCAT 通信設定
[SLAVE_1] StationAlias=1 SlaveType=0	←	各スレーブの通信設定 接続するスレーブ毎に用意 (サーボ)
[SLAVE_2] StationAlias=2 SlaveType=0	←	各スレーブの通信設定 接続するスレーブ毎に用意 (サーボ)
[SLAVE_3] StationAlias=3 SlaveType=1	←	各スレーブの通信設定 接続するスレーブ毎に用意 (I/O)

PLCopen から制御できるのは、サーボパックだけです。

## 4-4 エラーコード一覧

ファンクションブロック出力の ErrorID、MC\_ReadAxisError で読み出せるエラー番号の詳細を示します。

### 4-4-1 通信異常

エラー番号	エラー名	内容
0x10000001	リンクオフエラー	通信断が発生しています。 PLCopen は MECHATROLINK-III or EtherCAT 共に、正常に通信されていることが前提で動作します。 外的要因により通信断が発生した場合は、通信を再接続させた上で、MC_Power を実行してください。

### 4-4-2 機器異常

エラー番号	エラー名	内容
0x2000XXXX	サーボパック内部エラー	サーボパック内部で発生したエラーです。 下位 16bit がサーボパック側のエラー番号となります。 エラー内容については、各社サーボパックマニュアルを参照してください。

### 4-4-3 コマンド異常

エラー番号	エラー名	内容
0x30000000	スレッド生成エラー	スレッドの生成に失敗しました。
0x30000001	CPOpenMCPart1 生成エラー	PLCopen Part1 制御クラスの生成に失敗しました。
0x30000002	CPOpenMCPart5 生成エラー	PLCopen Part5 制御クラスの生成に失敗しました。
0x30000101	MC_Power エラー	予約。本エラーは発生しません。
0x30000102	MC_Stop エラー	予約。本エラーは発生しません。
0x30000103	MC_Home エラー	予約。本エラーは発生しません。
0x30000201	Homing 異常停止	原点復帰が異常停止しました。
0x30000202	Homing タイムアウト	原点復帰が指定された時間以内に完了しませんでした。
0x30000203	Homing ポジションリミット	原点復帰が指定された移動量以内で完了しませんでした。
0x30000204	Homing 未サポート	指定された原点復帰モードは、サーボパック側でサポートされていません。
0x30000301	Buffered 方向異常	FB 多重起動のブレンディング動作時に、もともと動作していたFBの動作方向と、多重起動されたFBの動作方向が違います。
0x30000302	±Limit 信号停止	±Limit 信号により停止しました。
0x30000303	位置決め完了タイムアウト	モータ動作停止後、位置決め完了幅範囲内に収まらなかった。
0x30000401	型タイプ異常	パラメータの型がありません。
0x30000801	グループへの軸登録未完	補間移動グループに軸が登録されていません。
0x30000802	単軸動作中	補間移動グループに登録しようとしている軸が単軸で動作しています。
0x30000803	単軸エラー発生中	補間移動グループに登録しようとしている軸がエラー状態です。
0x30000804	グループ登録軸数オーバ	グループに登録できる軸数を越えています。直線補間時は 16 軸まで、円弧補間時は 2 軸まで登録できます。
0x30001001	MC_IntelligentStop タイムアウト	トリガ条件がタイムアウト時間以内に一致しませんでした。

## 4-4-4 FB インスタンス異常

エラー番号	エラー名	内容
0x40000001	Buffered 未対応	仕様範囲外の Buffered 命令
0x40000002	ステータス異常	FB 呼び出しが仕様範囲外の状態で実行された
0x40000003	FB 入力値異常	FB 入力値が異常です。
0x40000004	同時読み込み	パラメータ読み込み FB が同時に複数実行された
0x40000005	同時書き込み	パラメータ書き込み FB が同時に複数実行された
0x40000006	前 FB 異常	Buffered で前 FB の完了待機時に前 FB で異常が発生した。
0x40000801	グループへ登録済み	すでにグループに登録されています。
0x4FFFFFFE	I0 ドライバ初期化異常	I0 ドライバ設定ミス、マスタプロセス未起動
0x4FFFFFFF	FB 機能未サポート	未サポート FB を使用した

## 4-4-5 通信コマンド異常

エラー番号	エラー名	内容
0x5000XXXX	テクノモーションライブラリ 内部エラー	テクノモーションライブラリ内部で発生したエラーです。 下位 16bit がテクノモーションライブラリ側のエラー番号となります。 エラー番号は下表を参照してください。
0x5001XXXX	MECHATROLINK-III ワーニング	MECHATROLINK-III 通信コマンドでワーニングが発生しています。 下位 16bit が MECHATROLINK-III 通信レスポンスデータの CMD_STAT の情報です。

## ● MECHATROLINK-III

エラー番号	エラー名	内容
1	引数エラー	
2	共通データ不正エラー	
3	MECHATROLINK-III データ不正エラー	
100	プロテクト認証エラー	
101	ライブラリ初期化済	
102	ライブラリ未初期化	
103	スレッドリソース初期化済	
104	スレッドリソース未初期化	
105	スレッド数フル	
106	マスタ初期化済	
107	マスタ未初期化	
108	スレーブ未接続	
109	プロファイル違いエラー	
110	サーボ OFF エラー	
111	指令有りエラー	
112	ガントリ軸設定済	
113	ガントリ軸未設定	
114	ガントリスレーブ軸への指令	
115	ガントリサポート対象外指令	
116	レスポンスフルエラー	
200	バッファフルエラー	
201	バッファ空エラー	
202	コマンド違いエラー	

203	稼働済移動グループ登録エラー	
204	登録済移動グループ登録エラー	
205	バッファグループ即実行コマンド登録エラー	
206	応答待ち時タイムアウトエラー	
300	非移動中エラー	
301	一時停止無視エラー	
302	加減速フィルターバッファフル	
303	加速時定数オーバーエラー	
304	減速時定数オーバーエラー	
305	S字加減速時定数オーバーエラー	
306	円弧指令回転方向違いエラー	
307	円弧半径オーバーフローエラー	
308	円弧中心点違いエラー	
309	円弧中心点計算エラー	
400	サーボ主電源 OFF	
401	+方向ソフトリミット	
402	-方向ソフトリミット	
403	+方向オーバートラベル	
404	-方向オーバートラベル	
500	ML コマンドタイムアウトエラー	
501	ML 通信エラー	
502	ML WDT エラー	
503	ML データ受信無しエラー	
504	ML アラーム	
505	ML ワーニング	
1000	INtime 未起動	
1001	INtime プロセス無し	
1002	位置決めLIB 未起動	
1003	パラメータエラー	
1004	空き通信ハンドル無し	
1005	無効通信ハンドル	
1006	多重ハンドル	
1007	通信ビジー	
1008	通信データサイズエラー	
1100	通信デバイス未初期化	
1101	通信タイムアウト	
1102	リトライオーバー	
1103	多重リトライ	
1104	通信シーケンスエラー	
1999	通信中不明エラー	
9999	不明エラー	
-1	その他のエラー	

## ● EtherCAT

エラー番号	エラー名	内容
1	引数エラー	各種 API 呼び出し時、引数の値が範囲外の場合に発生します
2	共通データ不正エラー	各種 API 呼び出し時、共通データ構造体に指定した値が範囲外の場合に発生します
4	EtherCAT 設定データ不正エラー	マスタ初期化 API 呼び出し時に、API で指定したパラメータと ENI ファイルの内容に相違がある場合に発生します。
100	プロテクト認証エラー	ライセンス認証に不備のある状態で各種 API を呼び出すと発生します
101	ライブラリ初期化済	RtpIectInitializeLib でライブラリ初期化正常終了後、再度、同 API を呼び出すと発生します
102	ライブラリ未初期化	RtpIectInitializeLib 実行前に各種初期化 API を呼び出すと発生します
103	スレッドリソース初期化済	RtpIectInitializeAPIThread をスレッドリソース初期化済みの時に呼び出すと発生します
104	スレッドリソース未初期化	RtpIectInitializeAPIThread 正常終了前に、各種 API を呼び出すと発生します
105	スレッド数フル	RtpIectInitializeAPIThread でリソース確保済のスレッドが最大数に達している時に RtpIectInitializeAPIThread を呼び出すと発生します
106	マスタ初期化済	RtpIectInitializeMaster 正常終了後、再度、RtpIectInitializeMaster を呼び出すと発生します
107	マスタ未初期化	RtpIectInitializeMaster の成功前に、各種 API を呼び出すと発生します
108	スレーブ未接続	スレーブが未接続の時に、各種 API を呼び出すと発生します
109	プロファイル違いエラー	スレーブ接続時に指定したプロファイルに未対応の関数を呼び出すと発生します。
110	サーボ OFF エラー	移動系 API を、サーボ OFF の時に呼び出すと発生します
111	指令有りエラー	ガントリマスター軸、ガントリスレーブ軸いずれかが動作中、又は、コマンドバッファにコマンドが登録されている時に RtpIectSetGantryAxis を呼び出すと発生します
112	ガントリ軸設定済	ガントリ設定済の軸に対し、再度、RtpIectSetGantryAxis を呼び出すと発生します
113	ガントリ軸未設定	ガントリマスター軸以外の軸に、RtpIectResetGantryAxis を呼び出すと発生します
114	ガントリスレーブ軸への指令	ガントリスレーブ軸に対して移動系 API を呼び出すと発生します
115	ガントリサポート対象外指令	ガントリマスター軸にガントリ対応していない移動系 API を呼び出すと発生します
116	レスポンスフルエラー	レスポンスキューが最大値の場合に各種 API を呼び出すと発生します。レスポンスキューの最大値は、スレッド毎・各スレーブ毎に 64 個。システム全体として 1000 個となります。



200	バッファフルエラー	コマンドバッファに空きが無い時に各種 API を呼び出すと発生します。コマンドバッファは各スレーブ 8 個です。
201	バッファ空エラー	移動系 API を、目標位置変更設定で呼び出した時、又は、実行時に、更新対象の移動コマンドがコマンドバッファに無い時に発生します
202	コマンド違いエラー	グループ ID を指定して移動系 API を呼び出した時に、既に同じグループ ID が別の移動系 API で使用されている場合に発生します
203	稼働済移動グループ登録エラー	グループ ID を指定して移動系 API を呼び出した時に、既に同じグループ ID のコマンドが動作中の場合に発生します
204	登録済移動グループ登録エラー	グループ ID を指定して移動系 API を呼び出した時に、既に同じ軸にそのグループ ID のコマンドが登録されていた場合に発生します
205	バッファグループ即実行コマンド登録エラー	コマンドバッファ上に未アクティブな移動系 API が登録されている時に、その同じグループ ID でアクティブな移動系 API を呼び出すと発生します
206	応答待ち時タイムアウトエラー	RtpIcTResWait を呼び出し後、指定したタイムアウト以上時間が経過しても応答が無い場合に発生します
300	非移動中エラー	移動系 API の実行中では無いときに、移動中断、一時停止、一時停止解除等の API を実行すると発生します
301	一時停止無視エラー	一時停止に対応していない移動系 API の実行中に一時停止 API を実行すると発生します
302	加減速フィルターバッファフル	補間コマンド実行時に、加減速フィルターが不足した場合に発生します。加速時定数より短い時間内に、連続で補間コマンドを実行すると発生します
303	加減速時定数オーバーエラー	補間コマンドの加速時定数に 2000 制御周期より長い値を指定すると、補間コマンド呼び出し時に発生します。 加速時定数は下記の計算で求められます。 加減速時定数 = (速度 / 加減速度) x 1000 速度のほうが加減速度の 2 倍より高い値を設定した場合にエラーとなります。
305	S 字加減速時定数オーバーエラー	補間コマンドの S 字加減速時定数に 2000 制御周期より長い値を指定すると、補間コマンド呼び出し時に発生します
306	円弧指令回転方向違いエラー	通過点指定の円弧補間コマンドにおいて、通過点から求めた回転方向と、引数の回転方向が異なると、円弧補間コマンド実行時に発生します
307	円弧半径オーバーフローエラー	中心点指定の円弧補間コマンドにおいて、算出した半径が半径の上限値 (2147483647) を超えていた場合に発生します。
308	円弧中心点違いエラー	中心点指定の円弧補間コマンドにおいて、移動開始位置から中心点までの距離と、目標位置から中心点までの距離が大きく異なる場合 (1000 [指令単位] 以上) に発生します。

309	円弧中心点計算エラー	半径指定の円弧補間コマンドにおいて、中心点を求められない場合に発生します。例) 目標位置までの距離が直径より長い場合半径が求まらない。
310	位置制御不可エラー	PDO マッピングが不足し csp(サイクリック位置決め)モードでの制御が不可の場合に発生します。
311	ラッチ方法選択不可エラー	PDO マッピングが不足しラッチ位置決めで指定したラッチ方法が実行できない場合に発生します。
312	トルク制御不可エラー	PDO マッピングが不足しトルク制御が実行できない場合に発生します。
313	ポジションプリセット不可エラー	CiA402 準拠サーボパックが原点復帰方法 35, 37 両方に対応していない場合に発生します。
400	サーボ主電源 OFF	StatusWord(6041h)の bit4:voltage enabled が 0 の時に発生します
403	＋方向オーバートラベル	Digital inputs(60FDh)の bit1:positive limit switch (POT)のビットが1の時に発生します。但し、ラッチ位置決めにおいて、＋方向の OT 信号をラッチする場合は POT ビットが1になってもエラーになりません。
404	－方向オーバートラベル	Digital inputs(60FDh)の bit0:negative limit switch (NOT)のビットが1の時に発生します。但し、ラッチ位置決めにおいて、－方向の OT 信号をラッチする場合は NOT ビットが1になってもエラーになりません。
600	ENI ファイルとスレーブ情報が異なる	ライブラリ初期化時にマスタースタックの API ファイル(*.rs1)が検出できない場合に発生します
601	未対応動作ステータスエラー	マスタ初期化時に ENI ファイルに構成されているスレーブ数と、API のパラメータのスレーブ数が異なる場合に発生します
602	未対応動作モード設定エラー	CiA402 スレーブの未対応制御モードを設定しようとした時に発生します
603	未対応 PDO の呼び出しエラー	PDO にマッピングされていないオブジェクトにアクセスしようとした時に発生します
604	モード変更失敗エラー	制御モードの変更に失敗した場合に発生します
605	コマンドタイムアウト	サーボドライブ状態を変更するコマンドにおいて API に指定したタイムアウト時間内に処理が終了しなかった場合に発生します
606	アラーム発生	サーボドライブ状態において FAULT となった時に発生します
607	ワーニング発生	サーボドライブ状態において WARNING となった時に発生します
608	即停止中指令エラー	サーボドライブ状態を変更するコマンドにおいて即停止状態(QUICK STOP)になった場合に発生します
609	マスタスタックエラー	RTPL-EC 内でマスタースタック API を実行した時にエラーとなった場合に発生します。このエラー発生時には、ステータスマニタで直前に発生したマスタのエラーコードを確認できます
610	原点復帰エラー	hm(Homing mode)を使用する API において、原点復帰実行時にスレーブが Homing Error となった場合に発生します

611	未対応原点復帰方法エラー	GiA402 スレーブの未対応の原点復帰方法を実行しようとした時に発生します
1000	INtime 未起動	
1001	INtime プロセス無し	
1002	位置決めLIB 未起動	
1003	パラメータエラー	
1004	空き通信ハンドル無し	
1005	無効通信ハンドル	
1006	多重ハンドル	
1007	通信ビジー	
1999	通信中不明エラー	
9999	不明エラー	
-1	その他のエラー	

## 4-4-6 EtherCAT 通信コマンド異常

エラー番号	エラー名	内容
0x80000001	ACAT_NOTIFY_ER_CYCCMD_WKC	サイクルコマンドエラー (WKC)
0x80000002	ACAT_NOTIFY_ER_MASTER_INITCMD_WKC	マスター初期化エラー (WKC)
0x80000003	ACAT_NOTIFY_ER_SLAVE_INITCMD_WKC	スレーブ初期化エラー (WKC)
0x80000005	ACAT_NOTIFY_ER_COE_MBXRCV_WKC	CoE メールボックス受信エラー (WKC)
0x80000008	ACAT_NOTIFY_ER_COE_MBXSND_WKC	CoE メールボックス送信エラー (WKC)
0x8000000A	ACAT_NOTIFY_ER_FRAME_RESPONSE	Ethernet フレームのレスポンスがない
0x8000000B	ACAT_NOTIFY_ER_SLAVE_INITCMD_RESPONSE	スレーブ初期化のレスポンスがない
0x8000000C	ACAT_NOTIFY_ER_MASTER_INITCMD_RESPONSE	マスター初期化のレスポンスがない
0x8000000D	ACAT_NOTIFY_ER_CMD_MISSING	受信した Ethernet フレームの EtherCAT コマンドが間違えている
0x8000000E	ACAT_NOTIFY_ER_MBSLAVE_INITCMD_TIMEOUT	初期化コマンドがタイムアウト
0x8000000F	ACAT_NOTIFY_ER_NOT_ALL_DEVICES_OPERATIONAL	全スレーブが Operational にならない
0x80000010	ACAT_NOTIFY_ER_ETH_LINK_NOT_CONNECTED	Ethernet のリンクが途切れた
0x80000012	ACAT_NOTIFY_ER_RED_LINEBRK	Cable Redundancy は未サポート
0x80000013	ACAT_NOTIFY_ER_STATUS_SLAVE	スレーブの AL-Status が異常
0x80000014	ACAT_NOTIFY_ER_SLAVE_STATUS_INFO	スレーブの Status Information が異常
0x80000015	ACAT_NOTIFY_ER_SLAVE_NOT_ADDRESSABLE	スレーブのステーションアドレスが異常
0x80000016	ACAT_NOTIFY_ER_MBSLAVE_COE_SDO_ABORT	SDO Abort 異常
0x80000017	ACAT_NOTIFY_ER_CLIENTREGISTRATION_DROPPED	Client registration dropped
0x80000018	ACAT_NOTIFY_ER_RED_LINEFIXED	Cable Redundancy は未サポート
0x80000019	ACAT_NOTIFY_ER_MBXRCV_INVALID_DATA	メールボックスの無効なデータ
0x8000001A	ACAT_NOTIFY_ER_PDIWATCHDOG	PDI ウォッチドッグエラー
0x8000001B	ACAT_NOTIFY_ER_SLAVE_NOTSUPPORTED	スレーブ未サポート
0x8000001C	ACAT_NOTIFY_ER_SLAVE_UNEXPECTED_STATE	スレーブの予期しない State
0x8000001D	ACAT_NOTIFY_ER_ALL_DEVICES_OPERATIONAL	全スレーブが Operational になった
0x8000001E	ACAT_NOTIFY_ER_EEPROM_CHECKSUM	EEPROM のチェックサムエラー
0x80000101	ACAT_NOTIFY_ER_HC_DETECTADGROUPS	HotConnect は未サポート
0x80000102	ACAT_NOTIFY_ER_HC_PROBEALLGROUPS	HotConnect は未サポート
0x80000103	ACAT_NOTIFY_ER_HC_TOPOCHGDONE	HotConnect は未サポート
0x80000104	ACAT_NOTIFY_ER_HC_SLAVE_PART	HotConnect は未サポート
0x80000105	ACAT_NOTIFY_ER_HC_SLAVE_JOIN	HotConnect は未サポート
0x90000001	ACAT_ER_ALREADYOPEN	既にオープンしている
0x90000002	ACAT_ER_NOTOPEN	オープンされていない
0x90000003	ACAT_ER_INVALIDPARAM	無効な引数
0x90000004	ACAT_ER_OPENDEVICE	デバイスドライバー起動失敗
0x90000005	ACAT_ER_CREATETHREAD	スレッド作成失敗
0x90000006	ACAT_ER_CREATESEMAPH	セマフォ作成失敗
0x90000007	ACAT_ER_CREATEMAP	マップトファイル作成失敗
0x90000008	ACAT_ER_CREATEMAIL	メールスロット作成失敗
0x90000009	ACAT_ER_INICONFIG	設定ファイルの記述エラー
0x9000000A	ACAT_ER_ALREADYSTART	スタートしている
0x9000000B	ACAT_ER_NOTSTART	スタートしていない
0x9000000C	ACAT_ER_DEVICE	デバイスアクセスエラー
0x9000000D	ACAT_ER_LOADDEVICE	デバイスロードエラー
0x9000000E	ACAT_ER_MAILCODE	メール受信コードエラー
0x9000000F	ACAT_ER_NOTFOUND_SLAVE	スレーブ検出失敗

0x9000010	ACAT_ER_ETC_COMLINK	EC マスタ通信エラー
0x9000011	ACAT_ER_NOTSERVOON	サーボ ON ではない
0x9000012	ACMST_ER_MOVING_MODECHANGE	動作中のオペレーションモード設定変更
0x9000013	ACAT_ER_TIMEOUT	コマンドタイムアウト
0x9000014	ACMST_ER_NOTPDOADDR	PDO 割付にない
0x9000015	ACAT_ER_BUSY	SDO がビジー状態
0x9000016	ACAT_ER_STATE_FAULT	ステートが FAULT 状態
0x9000017	ACAT_ER_NOTCIA402	CiA402 デバイスではない
0x90000101	ACAT_ER_MST_NOTSUPPORT	未サポートエラー
0x90000102	ACAT_ER_MST_INVALIDINDEX	無効なインデックス
0x90000103	ACAT_ER_MST_INVALIDOFFSET	無効なオフセット
0x90000104	ACAT_ER_MST_CANCEL	キャンセルエラー
0x90000105	ACAT_ER_MST_INVALIDSIZE	無効なサイズ
0x90000106	ACAT_ER_MST_INVALIDDATA	無効なデータ
0x90000107	ACAT_ER_MST_NOTREADY	準備ができていない
0x90000108	ACAT_ER_MST_BUSY	BUSY 状態
0x90000109	ACAT_ER_MST_ACYC_FRM_FREEQ_EMPTY	EtherCAT コマンドが送信できない
0x9000010A	ACAT_ER_MST_NOMEMORY	メモリ不足
0x9000010B	ACAT_ER_MST_INVALIDPARM	無効なパラメータ
0x9000010C	ACAT_ER_MST_NOTFOUND	EtherCAT マスタが見つからない
0x9000010E	ACAT_ER_MST_INVALIDSTATE	無効なステート
0x9000010F	ACAT_ER_MST_TIMER_LIST_FULL	タイマーリストにスレーブを追加できない
0x90000110	ACAT_ER_MST_TIMEOUT	タイムアウト
0x90000111	ACAT_ER_MST_OPENFAILED	オープンできない
0x90000112	ACAT_ER_MST_SENDFAILED	送信できない
0x90000113	ACAT_ER_MST_INSERTMAILBOX	MailBox に追加できない
0x90000114	ACAT_ER_MST_INVALIDCMD	無効な MailBox コマンド
0x90000115	ACAT_ER_MST_UNKNOWN_MBX_PROTOCOL	未定義な MailBox プロトコル
0x90000116	ACAT_ER_MST_ACCESSDENIED	アクセス拒否
0x9000011A	ACAT_ER_MST_PRODKEY_INVALID	無効なプロダクトキー
0x9000011B	ACAT_ER_MST_WRONG_FORMAT	コンフィグファイルフォーマット警告
0x9000011C	ACAT_ER_MST_FEATURE_DISABLED	特色無効
0x9000011D	ACAT_ER_MST_SHADOW_MEMORY	間違ったモードで要求されたメモリ
0x9000011E	ACAT_ER_MST_BUSCONFIG_MISMATCH	バス設定が誤り
0x9000011F	ACAT_ER_MST_CONFIGDATAREAD	コンフィグファイル読み込みエラー
0x90000121	ACAT_ER_MST_XML_CYCCMDS_MISSING	サイクリックコマンド送信失敗
0x90000122	ACAT_ER_MST_XML_ALSTATUS_READ_MISSING	ALSTATUS 読み込み失敗
0x90000123	ACAT_ER_MST_MCSM_FATAL_ERROR	McSm の致命的エラー
0x90000124	ACAT_ER_MST_SLAVE_ERROR	スレーブエラー
0x90000125	ACAT_ER_MST_FRAME_LOST	フレームロスト, IDX ミスマッチ
0x90000126	ACAT_ER_MST_CMD_MISSING	受信フレームで1つの EtherCAT コマンドが失敗
0x90000128	ACAT_ER_MST_INVALID_DCL_MODE	無効な DC ラッチモード
0x90000129	ACAT_ER_MST_AI_ADDRESS	オートインクリメントアドレスの加算異常(スレーブ異常)
0x9000012A	ACAT_ER_MST_INVALID_SLAVE_STATE	無効なステート(スレーブ異常)
0x9000012B	ACAT_ER_MST_SLAVE_NOT_ADDRESSABLE	無効なアドレス(スレーブ異常)
0x9000012C	ACAT_ER_MST_CYC_CMD_S_OVERFLOW	コンフィグファイルで多くのサイクリックコマンドが実行された
0x9000012D	ACAT_ER_MST_LINK_DISCONNECTED	Ethernet ケーブルが切断されました

0x9000012E	ACAT_ER_MST_MASTERCORE_INACCESSIBLE	EtherCAT マスタにアクセスできません
0x9000012F	ACAT_ER_MST_COE_MBXSNDR_WKC_ERROR	送信 MailBox のワーキングカウンタエラー
0x90000130	ACAT_ER_MST_COE_MBXRCV_WKC_ERROR	受信 MailBox のワーキングカウンタエラー
0x90000131	ACAT_ER_MST_NO_MBX_SUPPORT	MailBox 未サポート
0x90000132	ACAT_ER_MST_NO_COE_SUPPORT	CoE 未サポート
0x90000133	ACAT_ER_MST_NO_EOE_SUPPORT	EOE 未サポート
0x90000134	ACAT_ER_MST_NO_FOE_SUPPORT	FoE 未サポート
0x90000135	ACAT_ER_MST_NO_SOE_SUPPORT	SoE 未サポート
0x90000136	ACAT_ER_MST_NO_VOE_SUPPORT	VoE 未サポート
0x90000137	ACAT_ER_MST_EVAL_VIOLATION	評価の構成に違反がありました
0x90000138	ACAT_ER_MST_EVAL_EXPIRED	評価は時間リミットに達しました
0x90000201	ACAT_ER_MST_SDO_ABORTCODE_TOGGLE	SDO Abort(トグルビットは交替されませんでした)
0x90000202	ACAT_ER_MST_SDO_ABORTCODE_TIMEOUT	SDO Abort(SDO プロトコル タイムアウト)
0x90000203	ACAT_ER_MST_SDO_ABORTCODE_CCS_SCS	SDO Abort(Client/Server Commandが無効)
0x90000204	ACAT_ER_MST_SDO_ABORTCODE_BLK_SIZE	SDO Abort(無効なブロックサイズ)
0x90000205	ACAT_ER_MST_SDO_ABORTCODE_SEQNO	SDO Abort(無効なシーケンス番号)
0x90000206	ACAT_ER_MST_SDO_ABORTCODE_CRC	SDO Abort(CRC エラー)
0x90000207	ACAT_ER_MST_SDO_ABORTCODE_MEMORY	SDO Abort(メモリ範囲外)
0x90000208	ACAT_ER_MST_SDO_ABORTCODE_ACCESS	SDO Abort(未サポートアクセス)
0x90000209	ACAT_ER_MST_SDO_ABORTCODE_WRITEONLY	SDO Abort(WriteOnly エリアを読み出した)
0x9000020A	ACAT_ER_MST_SDO_ABORTCODE_READONLY	SDO Abort(ReadOnly エリアに書き込んだ)
0x9000020B	ACAT_ER_MST_SDO_ABORTCODE_INDEX	SDO Abort(無効な Index 番号)
0x9000020C	ACAT_ER_MST_SDO_ABORTCODE_PDO_MAP	SDO Abort(PDO マッピングできないオブジェクト)
0x9000020D	ACAT_ER_MST_SDO_ABORTCODE_PDO_LEN	SDO Abort(PDO Length が間違えている)
0x9000020E	ACAT_ER_MST_SDO_ABORTCODE_P_INCOMP	SDO Abort(一般パラメータの不一致)
0x9000020F	ACAT_ER_MST_SDO_ABORTCODE_I_INCOMP	SDO Abort(内部情報の不一致)
0x90000210	ACAT_ER_MST_SDO_ABORTCODE_HARDWARE	SDO Abort(ハードウェア)
0x90000211	ACAT_ER_MST_SDO_ABORTCODE_DATA_SIZE	SDO Abort(データタイプ: パラメータミスマッチ)
0x90000212	ACAT_ER_MST_SDO_ABORTCODE_DATA_SIZE1	SDO Abort(データタイプ: パラメータ too long)
0x90000213	ACAT_ER_MST_SDO_ABORTCODE_DATA_SIZE2	SDO Abort(データタイプ: パラメータ too short)
0x90000214	ACAT_ER_MST_SDO_ABORTCODE_OFFSET	SDO Abort(サブインデックス)
0x90000215	ACAT_ER_MST_SDO_ABORTCODE_DATA_RANGE	SDO Abort(書込み: パラメータ範囲外)
0x90000216	ACAT_ER_MST_SDO_ABORTCODE_DATA_RANGE1	SDO Abort(書込み: パラメータ上限超)
0x90000217	ACAT_ER_MST_SDO_ABORTCODE_DATA_RANGE2	SDO Abort(書込み: パラメータ下限以下)
0x90000218	ACAT_ER_MST_SDO_ABORTCODE_MINMAX	SDO Abort(最大値が最小値よりも低い)
0x90000219	ACAT_ER_MST_SDO_ABORTCODE_GENERAL	SDO Abort(一般エラー)
0x9000021A	ACAT_ER_MST_SDO_ABORTCODE_TRANSFER	SDO Abort(データの転送/保存はできない)
0x9000021B	ACAT_ER_MST_SDO_ABORTCODE_TRANSFER1	SDO Abort(ローカル制御からデータの転送/保存はできない)
0x9000021C	ACAT_ER_MST_SDO_ABORTCODE_TRANSFER2	SDO Abort(現在の ESM ではデータの転送/保存はできない)
0x9000021D	ACAT_ER_MST_SDO_ABORTCODE_DICTIONARY	SDO Abort(オブジェクトディクショナリエラー)
0x9000021E	ACAT_ER_MST_SDO_ABORTCODE_UNKNOWN	SDO Abort(unknown code)
0x90000301	ACAT_ER_MST_CFGFILENOTFOUND	コンフィグファイルが見つからない
0x90000302	ACAT_ER_MST_EEPROMREADERROR	EEPROM 読み込みエラー
0x90000303	ACAT_ER_MST_EEPROMWRITEERROR	EEPROM 書込みエラー
0x90000304	ACAT_ER_MST_XML_CYCCMDS_SIZEMISMATCH	サイクリックコマンドサイズ不一致
0x90000305	ACAT_ER_MST_XML_INVALID_INP_OFF	サイクリックコマンド、無効入力オフセット
0x90000306	ACAT_ER_MST_XML_INVALID_OUT_OFF	サイクリックコマンド、無効出力オフセット

0x90000307	ACAT_ER_MST_PORTCLOSE	ポートクローズ失敗
0x90000308	ACAT_ER_MST_PORTOPEN	ポートオープン失敗
0x90000309	ACAT_ER_MST_SLAVE_NOT_PRESENT	スレーブはBusに出席していません
0x9000030A	ACAT_ER_MST_NO_FOE_SUPPORT_BS	FoE プロトコルは Boot Strap をサポートしていません
0x9000030B	ACAT_ER_MST_EEPROMRELOADERROR	EEPROM ReLoad エラー
0x9000030C	ACAT_ER_MST_SLAVECTRLRESETERROR	スレーブコントローラリセットエラー
0x9000030D	ACAT_ER_MST_SYSDRIVERMISSING	ドライバをオープンできません
0x9000030E	ACAT_ER_MST_BUSCONFIG_TOPOCHANGE	changed Topology のバス設定を検出できません
0x9000030F	ACAT_ER_MST_EEPROMASSIGNERROR	EEPROM の割り当てに失敗しました
0x90000310	ACAT_ER_MST_MBX_ERROR_TYPE	MailBox 受信エラー
0x90000311	ACAT_ER_MST_REDLINEBREAK	Redundancy line break
0x90000312	ACAT_ER_MST_XML_INVALID_CMD_WITH_RED	Redundancy の無効な EtherCAT コマンドを受信
0x90000313	ACAT_ER_MST_XML_PREV_PORT_MISSING	<PreviousPort>-tag 失敗
0x90000314	ACAT_ER_MST_XML_DC_NOT_ALLOWED_WITH_RED	Redundancy の DC アクセスを許さない
0x90000315	ACAT_ER_MST_DLSTATUS_IRQ_TOPOCHANGED	changed Topology の DL Status 割込み
0x9000031C	ACAT_ER_MST_DC_REF_CLOCK_SYNC_OUT_UNIT_DISABLED	DC リファレンスクロック無効
0x9000031D	ACAT_ER_MST_DC_REF_CLOCK_NOT_FOUND	DC リファレンスクロックが見つからない
0x9000031E	ACAT_ER_MST_XML_DC_REF_CLOCK_NOT_FIRST	最初のスレーブからリファレンスクロックが有効にならない
0x9000031F	ACAT_ER_MST_MBX_CMD_WKC_ERROR	MailBox コマンドのワーキングカウンタエラー
0x90000401	ACAT_ER_MAX_BUS_SLAVES_EXCEEDED	Bus Slaves が最大数を越えています
0x90000402	ACAT_ER_MBX_SYNTAX	MailBox ヘッダーが間違えています
0x90000403	ACAT_ER_MBX_UNSUPPORTEDPROTOCOL	MailBox プロトコルがサポートされていません
0x90000404	ACAT_ER_MBX_INVALIDCHANNEL	無効なチャンネルが選択されています
0x90000405	ACAT_ER_MBX_SERVICENOTSUPPORTED	MailBox プロトコルヘッダーが間違えています
0x90000406	ACAT_ER_MBX_INVALIDHEADER	MailBox プロトコルヘッダーが間違えています
0x90000407	ACAT_ER_MBX_SIZETOOSHORT	受信した MailBox データ長が Short ではありません
0x90000408	ACAT_ER_MBX_NOMOREMEMORY	MailBox プロトコルを受け取るリソースがありません
0x90000409	ACAT_ER_MBX_INVALIDSIZE	無効なサイズです
0x90000501	ACAT_ER_DCM_NOTINITIALIZED	DC モード初期化に失敗しました
0x90000502	ACAT_ER_DCM_MAX_CTL_ERROR_EXCEED	DC 同期通信に失敗しました
0x90000503	ACAT_ER_DCM_NOMEMORY	十分なメモリが確保できません
0x90000504	ACAT_ER_DCM_INVALID_HWLAYER	無効なハードウェアレイヤー
0x90000505	ACAT_ER_DCM_TIMER_MODIFY_ERROR	ハードウェアレイヤーエラー (タイマ補正)
0x90000506	ACAT_ER_DCM_TIMER_NOT_RUNNING	ハードウェアレイヤーエラー (タイマ未動作)
0x90000507	ACAT_ER_DCM_WRONG_CPU	ハードウェアレイヤーエラー (Wrong CPU)
0x90000508	ACAT_ER_DCM_INVALID_SYNC_PERIOD	無効な同期ピリオド
0x90000509	ACAT_ER_DCM_INVALID_SETVAL	SetVal が小さい
0x9000050A	ACAT_ER_DCM_DRIFT_TO_HIGH	タイマとリファレンスクロックの間のドリフトが高い
0xFFFFFFFF	ACAT_NOTIFY_ILLEGAL_CODE	イリーガルコード

## 第5章 付録

### 5-1 サンプルプロジェクト

#### 5-1-1 MULTIPROG 用 PLCopen サンプルプロジェクト

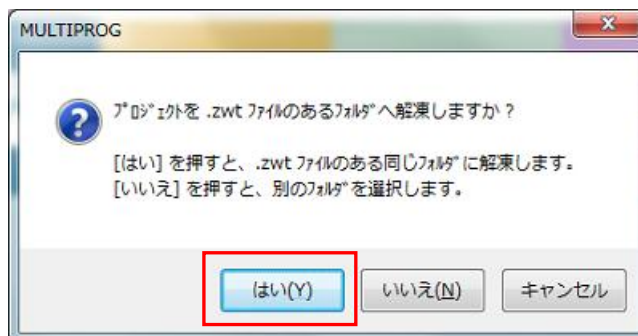
「AI-Motion-SDK 開発環境 CD-ROM」に MULTIPROG 用 AI-Motion 版 PLCopenFB を使用したサンプルプロジェクトを用意しています。サンプルコードは MULTIPROG のプロジェクトです。MULTIPROG の開発環境で使用することが可能です。DVD に含まれるサンプルプロジェクトの内容を表 5-1-1-1 に示します。

表 5-1-1-1. MULTIPROG 用 PLCopen サンプルプロジェクト

DVD-ROM のフォルダ	内容
¥SDK¥SAMPLE¥PLCopen¥5.35¥PLCopen_MC_P4¥MP_PLCopen_MC_Sample	AI-Motion 版 PLCopen サンプルプロジェクトです。(MULTIPROG5.35 用です)
¥SDK¥SAMPLE¥PLCopen¥5.50¥PLCopen_MC_P4¥MP_PLCopen_MC_Sample	AI-Motion 版 PLCopen サンプルプロジェクトです。(MULTIPROG5.50 用です)

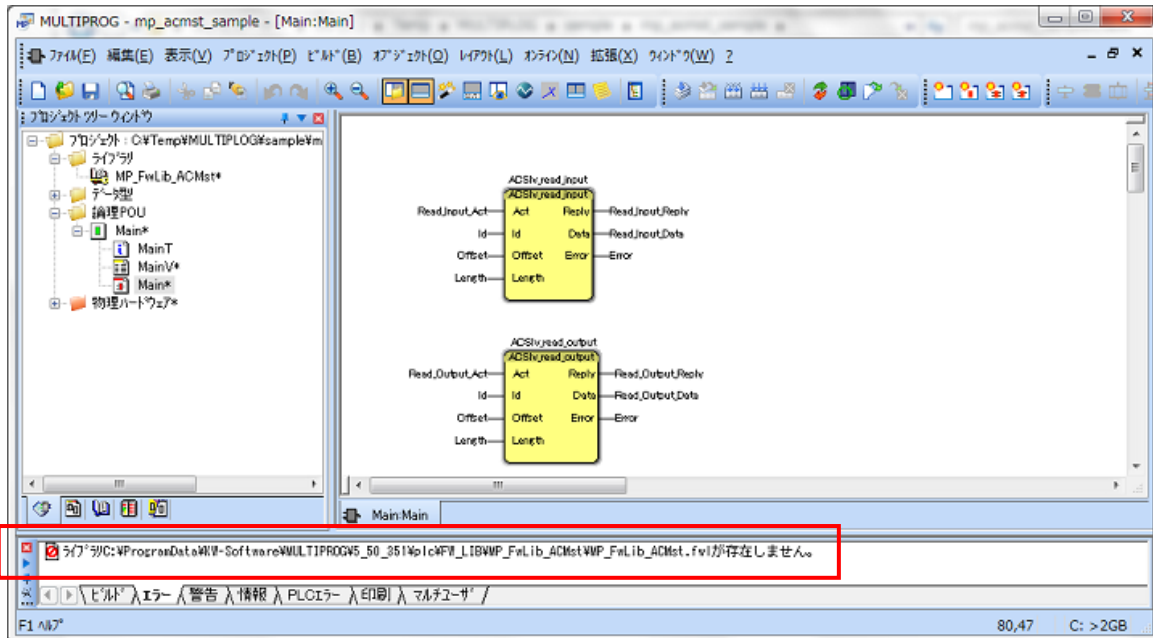
#### 5-1-2 サンプルプロジェクト使用方法

- ① マスタ制御関数のサンプルプロジェクト「mp\_\*\*\*\*\_sample.zwt」をダブルクリックしてください。下記、ダイアログが表示されますので、フォルダを指定し、「はい」を押下してください。

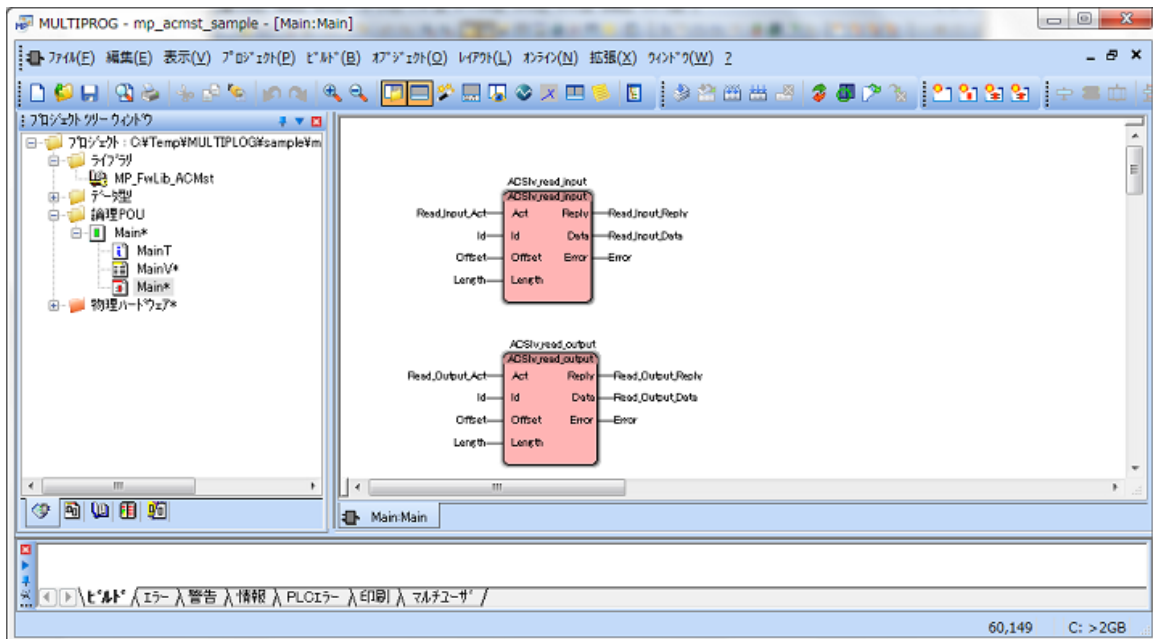




- ② サンプルプロジェクト解凍後、ファームウェアライブラリのライブラリパスが違うパスを指し示しています。  
「3-2 使用方法」を参考に再度、ファームウェアライブラリの登録してください。  
(ライブラリパスが違くとファンクションブロックが黄色になります。)



- ③ ライブラリパスを再設定後、MULTIPROG を再起動すると、ファンクションブロックが正常に表示されます。



## 5-2 参考文献

- 「IEC61131-3 を用いた PLC プログラミング」

著者	K.-H. John / M. Tiegelkamp
監訳者	PLCopen Japan
発行者	深田 良治
発行所	シュプリンガー・フェアラーク東京株式会社
発行年	2006 年

本 DVD には KW-Software 社提供の MULTIPROG に関するマニュアルも収録しております。

MULTIPROG の使用方法に関する詳細などはそちらを参照してください。

各マニュアルは<DVD>¥doc¥に収録されています。

また、サンプルコードも<DVD>¥sample¥に収録されています。こちらも参考にしてください。

## このユーザーズマニュアルについて

---

- (1)本書の内容の一部又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ください。
- (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づ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までご連絡ください。その際、巻末記載の書籍番号も併せてお知らせください。

77KW10021F  
77KW10021A

2016年 9月 第6版  
2013年 9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>