

**導入マニュアル**

**同期モーション制御  
(PLCopen 仕様)**

# 目次

## はじめに

1) お願いと注意	1
-----------	---

## 第1章 PLCopen について

1-1 PLCopen MC 仕様とは	1-1
---------------------	-----

## 第2章 MC 対応フィールドネットワークについて

2-1 MECHATROLINK-III	2-1
2-1-1 MECHATROLINK-III とは	2-1
2-1-2 MECHATROLINK-III 構成図	2-2
2-2 EtherCAT	2-4
2-2-1 EtherCAT とは	2-4
2-2-2 EtherCAT 構成図	2-5

## 第3章 標準版と AI-Motion について

3-1 PLCopen の仕様について	3-1
3-2 開発環境	3-5
3-3 実行環境	3-5

## 第4章 INtime 初期設定

4-1 デバイスの INtime への移動	4-1
4-1-1 マスタ拡張ボードデバイスの移動	4-2
4-1-2 Ethernet デバイスの移動	4-6
4-2 INtime ノードマネジメント設定	4-15
4-2-1 System Wide 設定	4-17
4-2-2 Kernel 設定	4-19

4-2-3	Network 設定	4-20
4-2-4	AutoLoad 設定	4-22
4-3	INtime レジストリキー登録について	4-27

# 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。

- PLCopen について
  - ・ PLCopen の仕様について
  - ・ PLCopen を INtime 上で使用するための手順
  - ・ MECHATROLINK-III、EtherCAT を INtime 上で使用するための手順

EtherCAT、MECHATROLINK-III、PLCopen の制御方法および詳細は省略させていただきます。  
それぞれの通信制御用ライブラリおよびテストツールを用意していますので、そちらのマニュアルを参照してください。

**※注：AI-Motion は、(株) テクノのテクノモーションライブラリを使って、PLCopen 準拠の同期制御と通常の位置決め制御を実現しています。そのため、標準品の FB とは動作が異なる場合があります。**

- INtime 初期設定について
  - ・ AI-PLC、AI-Motion を使用する場合、INtime のコンフィグ設定を変更するための手順

INtime は TenAsys 社の登録商標です。マイクロネット社はアジア地区の総代理店となっています。  
INtime についての技術的な質問は、直接 TenAsys 社、マイクロネット社に問い合わせしないでください。  
弊社にお問い合わせください。

# 第 1 章 PLCopen について

本章では PLCopen について解説します。

## 1-1 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、日本の主要メーカーを含む世界 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 1-1-1 の 6 つの Part から構成されています。

表 1-1-1. 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 1-1-1 に示します。

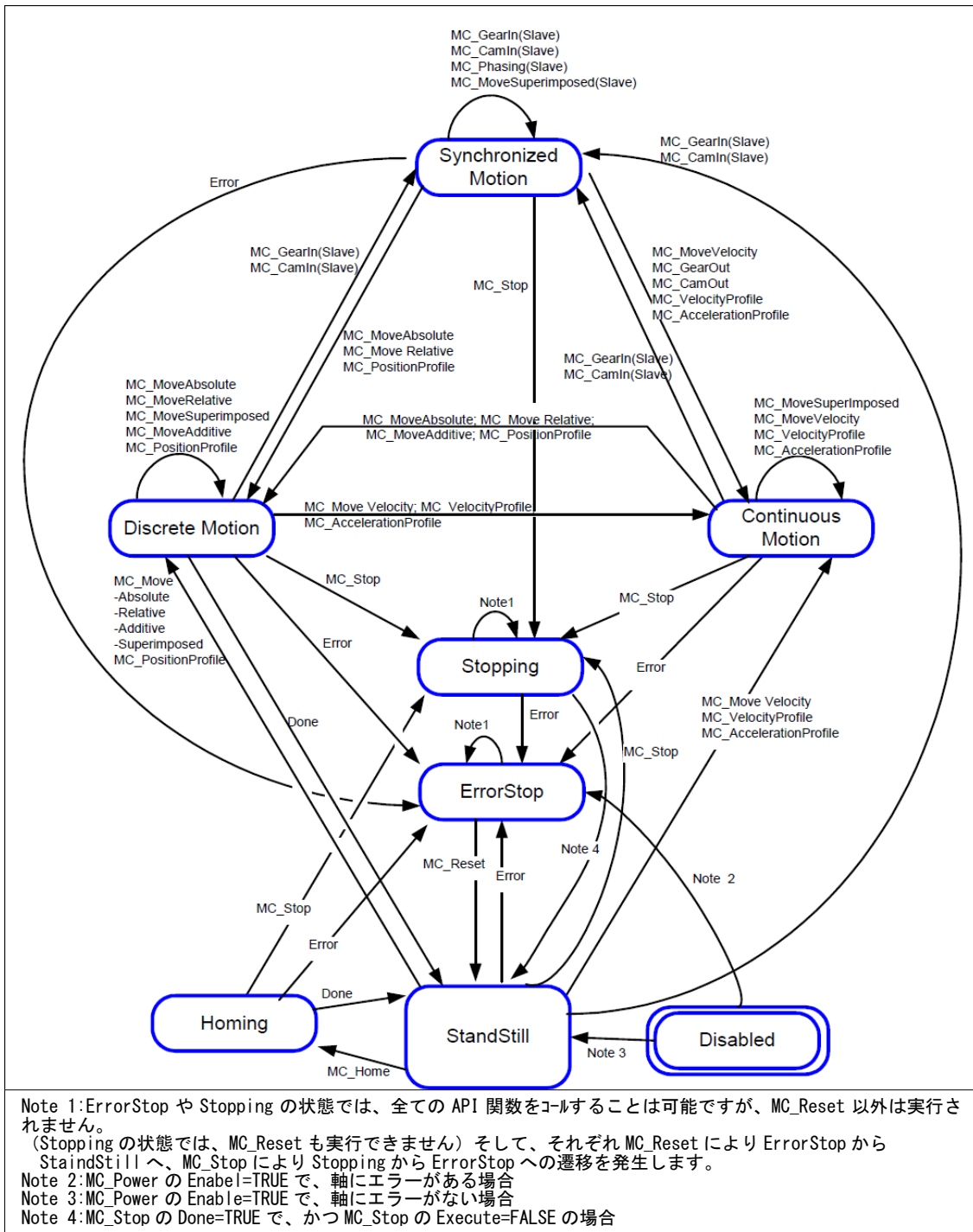


図 1-1-1. PLCopenMC 状態遷移図

図 1-1-1 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 1-1-2 に示します。

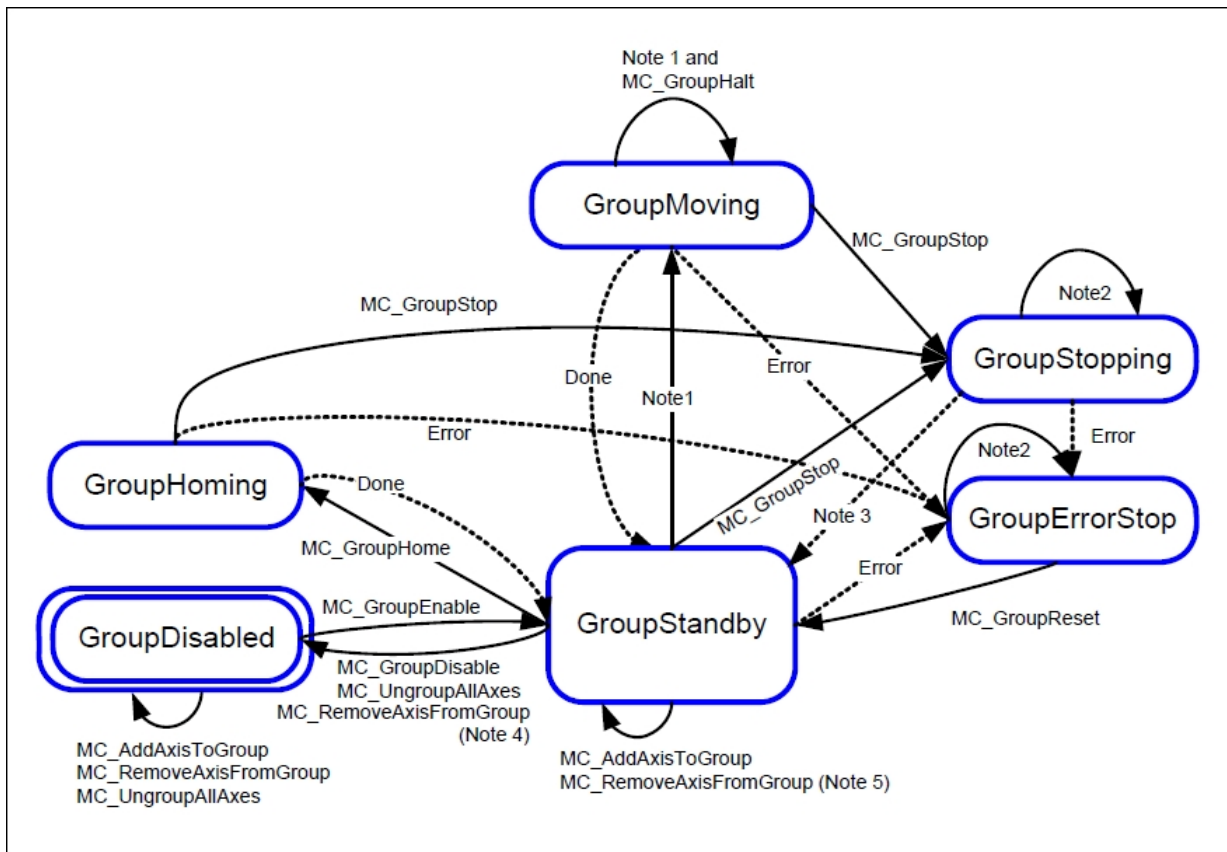


図 1-1-2. PLCopenMC Part4 状態遷移図

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 1-1-3 に示します。

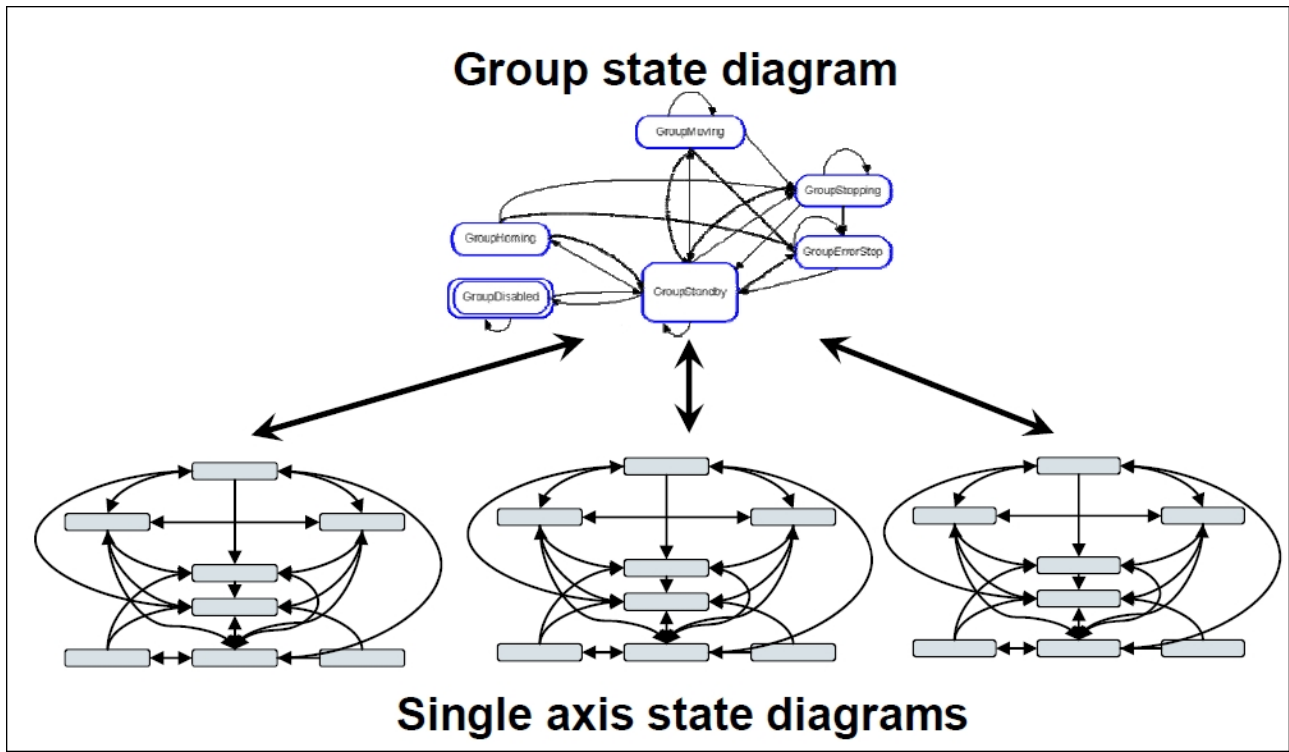


図 1-1-3. PLCopenMC Part4 単軸状態との相関図

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。



## 第2章 MC 対応フィールドネットワークについて

本章ではモーション制御に対応したフィールドネットワークについて解説します。

### 2-1 MECHATROLINK-III

本章では、MECHATROLINK-III通信について説明します。

#### 2-1-1 MECHATROLINK-IIIとは

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。

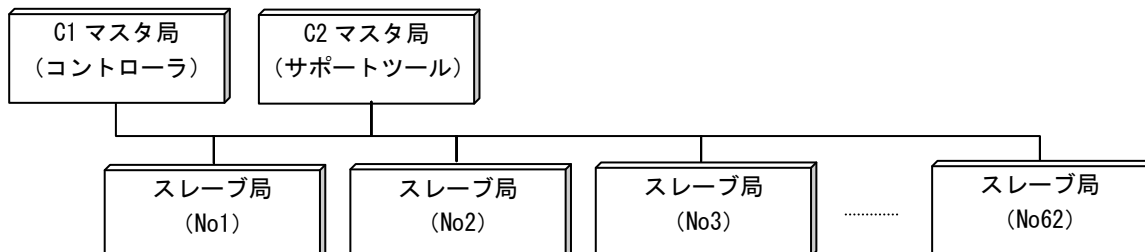


図 2-1-1-1. MECHATROLINK-III接続図

マスタ拡張ボードには、MECHATROLINK-III通信用 LSI (JL-101) が実装されており、通信制御を容易に行うことが可能です。マスタ拡張ボードを搭載することで、MECHATROLINK-IIIによる、PLCopen 準拠の同期モーション制御を行うことができます。

**※注：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。**

### 2-1-2 MECHATROLINK-III 構成図

MECHATROLINK-III 制御方法は下記の 2 種類があります。ユーザで最適な開発方法を選ぶことができます。

#### 1) 制御プログラムを C 言語で作成する場合

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団体が規格した、モーションコントロール仕様 (PLCopen 仕様 MC) です。

ユーザは PLCopen に準拠したライブラリ関数を使用し、PLCopen 仕様のプログラムを作成することで、PLCopen 準拠の制御プログラムを開発することが可能です。

ユーザアプリケーションは、Microsoft Visual Studio 統合開発環境を使って開発します。Windows 側の GUI アプリケーションとのインターフェースはユーザで設計することができます。

MECHATROLINK-III を C 言語アプリケーション+PLCopen で使用する際は、MECHATROLINK-III 版テクノモーションプロセス (RtpIML3Proc) と PLCopen 制御プロセス (PLCopenTechProc) を起動するようにしてください。

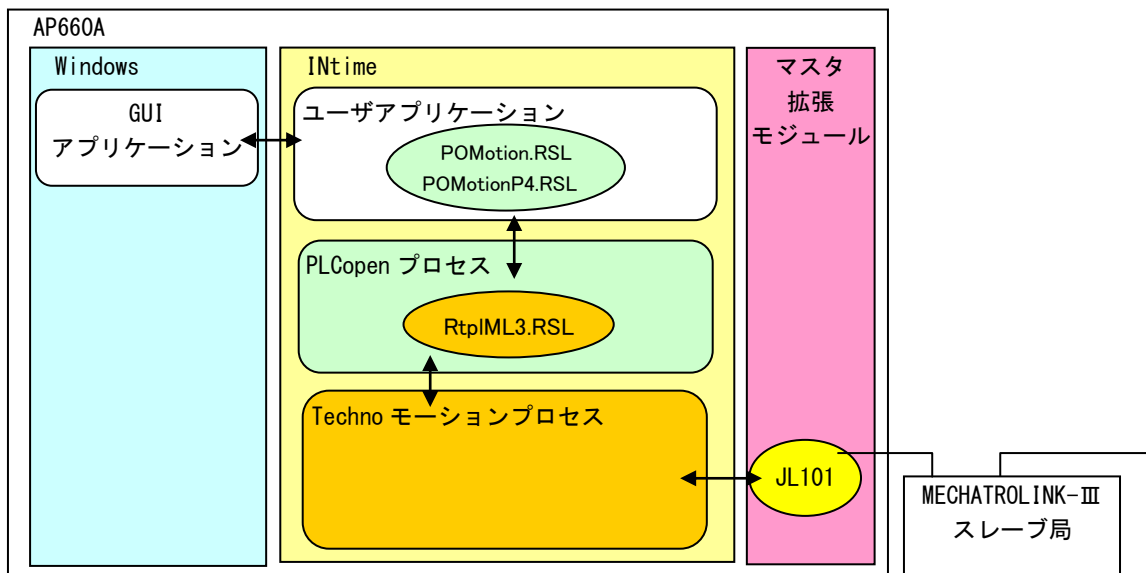


図 2-1-2-1. MECHATROLINK-III 制御構成図 (制御プログラムを C 言語で作成する場合)

※注：INtime アプリケーションを開発するためには、開発環境として、Microsoft Visual Studio 統合開発環境と INtime SDK が必要となります。

※注：INtime アプリケーション開発環境の詳細と開発方法については、「INtime 開発環境 ユーザーズマニュアル」を参照してください。

## 2) 制御プログラムをラダー言語で作成する場合

ProConOS は、INtime 上で動作する国際標準規格 IEC61131-3 に適合したソフト PLC です。専用の開発環境で開発したラダープログラムをダウンロードすることで制御を行うことができます。

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団体が規格した、モーションコントロール仕様 (PLCopen 仕様 MC) です。

ユーザはラダー言語により、PLCopen 仕様のプログラムを作成することで、MECHATROLINK-III通信を意識することなく制御プログラムを開発することが可能です。

MECHATROLINK-IIIをソフト PLC+PLCopen で使用する際は、MECHATROLINK-III版テクノモーションプロセス (RtpIML3Proc) と PLCopen 制御プロセス (PLCopenProc)、ソフト PLC プロセス (ProConOS) を起動するようにしてください。

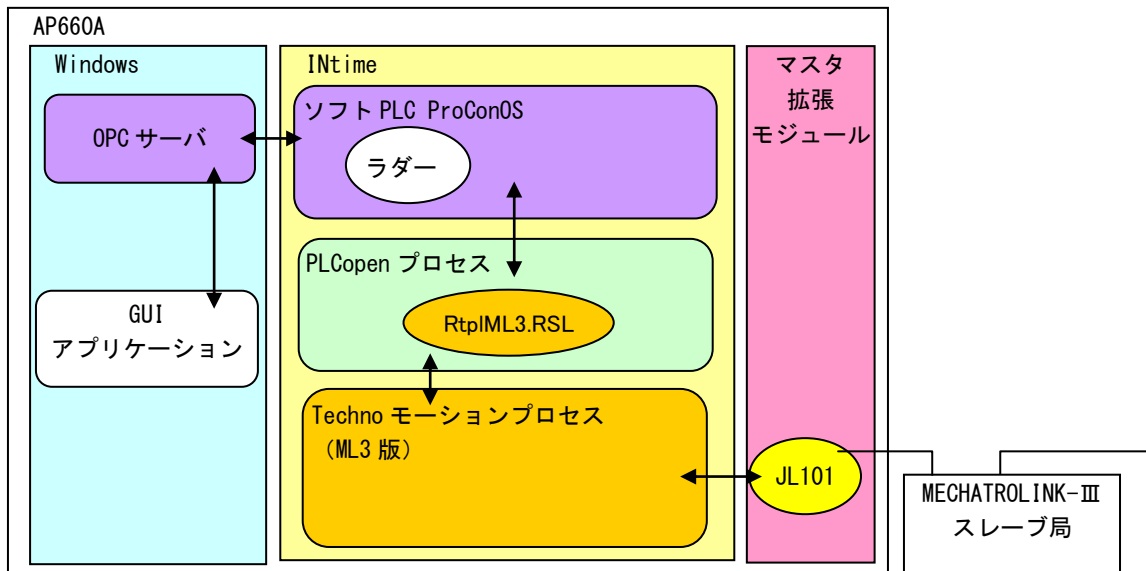


図 2-1-2-2. MECHATROLINK-III制御構成図 (制御プログラムをラダー言語で作成する場合)

**※注：ラダープログラムを開発するためには、開発環境として、Phoenix Contact (KW-Software) 製 MULTIPROG が必要となります。**

## 2-2 EtherCAT

### 2-2-1 EtherCAT とは

EtherCAT (Ethernet Control Automation Technology) は、IEEE802.3 Ethernet ベースのフィールドバスシステムです。EtherCAT はマスタとスレーブによって構成されます。

EtherCAT マスタを搭載した RT-OS 上のソフトウェアを用いて、EtherCAT スレーブの入出力を高速で制御できます。

ユーザはマスタとスレーブの通信を意識することなく、プログラミング言語、ラダー言語などからスレーブの入出力制御を行うことができます。

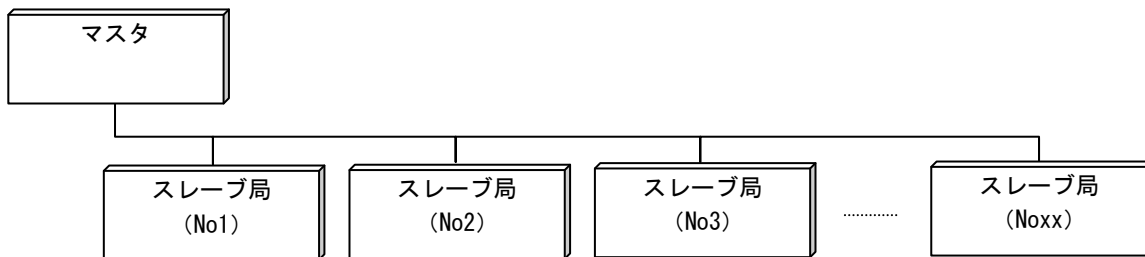


図 2-2-1-1. EtherCAT 接続図

EtherCAT を使用する場合は、INTIME 側に LAN ポートを 1 ポート専有させる必要があります。拡張ボードを別途使用する必要はありません。

**※注：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。**

### 2-2-2 EtherCAT 構成図

EtherCAT 制御方法は 2 種類の方法があります。ユーザで最適な開発方法を選ぶことができます。

#### 1) 制御プログラムを C 言語で作成する場合

EtherCAT アクセスドライバを使って、INtime 上で動作する EtherCAT 制御プロセス (ACatProc プロセス) を使い、ユーザアプリケーションから専用のライブラリ関数を呼び出すことで制御する方式です。

ユーザアプリケーションは、Microsoft Visual Studio 統合開発環境を使って開発します。Windows 側の GUI アプリケーションとのインターフェースはユーザで設計することができます。

EtherCAT 通信制御は ACatProc プロセスが行っていますので、ユーザは EtherCAT 通信を意識することなくユーザアプリケーションを開発することができます。

EtherCAT を C 言語アプリケーション+PLCopen で使用する際は、EtherCAT マスタプロセス (ACatProc)、EtherCAT 版テクノモーションプロセス (RtpIECTProc) と PLCopen 制御プロセス (PLCopenTechProc) を起動するようにしてください。

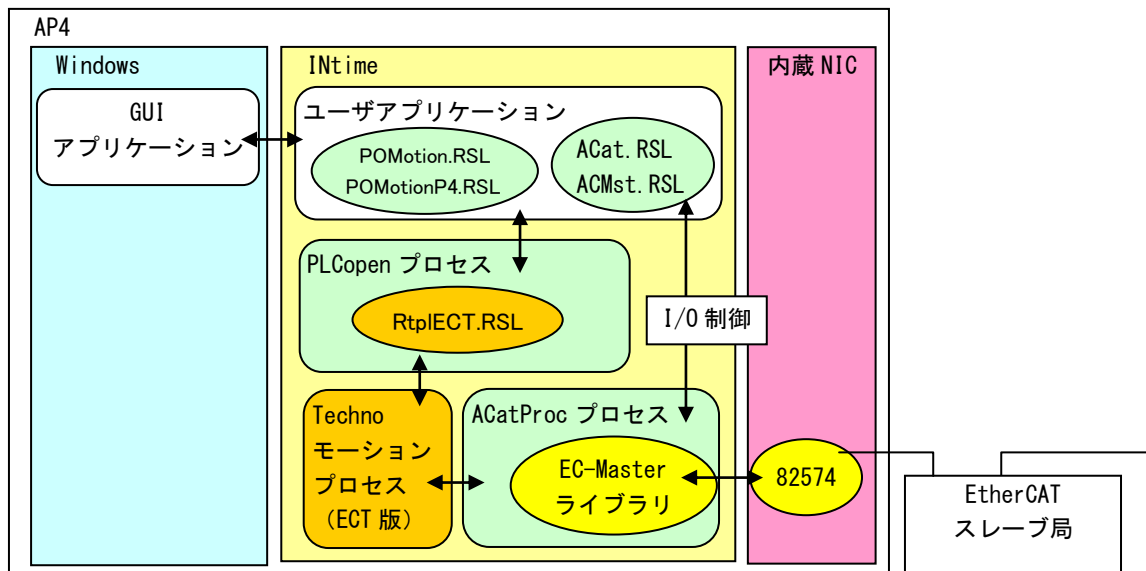


図 2-2-2-1. EtherCAT 制御構成図 (制御プログラムを C 言語で作成する場合)

※注：INtime アプリケーションを開発するためには、開発環境として、Microsoft Visual Studio 統合開発環境と INtime SDK が必要となります。

※注：INtime アプリケーション開発環境の詳細と開発方法については、「INtime 開発環境 ユーザーズマニュアル」を参照してください。

## 2) 制御プログラムをラダー言語で作成する場合

ProConOS は、INtime 上で動作する国際標準規格 IEC61131-3 に適合したソフト PLC です。専用の開発環境で開発したラダープログラムをダウンロードすることで制御を行うことができます。

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団体が規格した、モーションコントロール仕様 (PLCopen 仕様 MC) です。

ユーザはラダー言語により、PLCopen 仕様のプログラムを作成することで、EtherCAT 通信を意識することなく制御プログラムを開発することが可能です。

EtherCAT をソフト PLC+PLCopen で使用する際は、EtherCAT マスタプロセス (ACatProc) と EtherCAT 版テクノモーションプロセス (RtpIECTProc) と PLCopen 制御プロセス (PLCopenProc)、ソフト PLC プロセス (ProConOS) を起動するようにしてください。

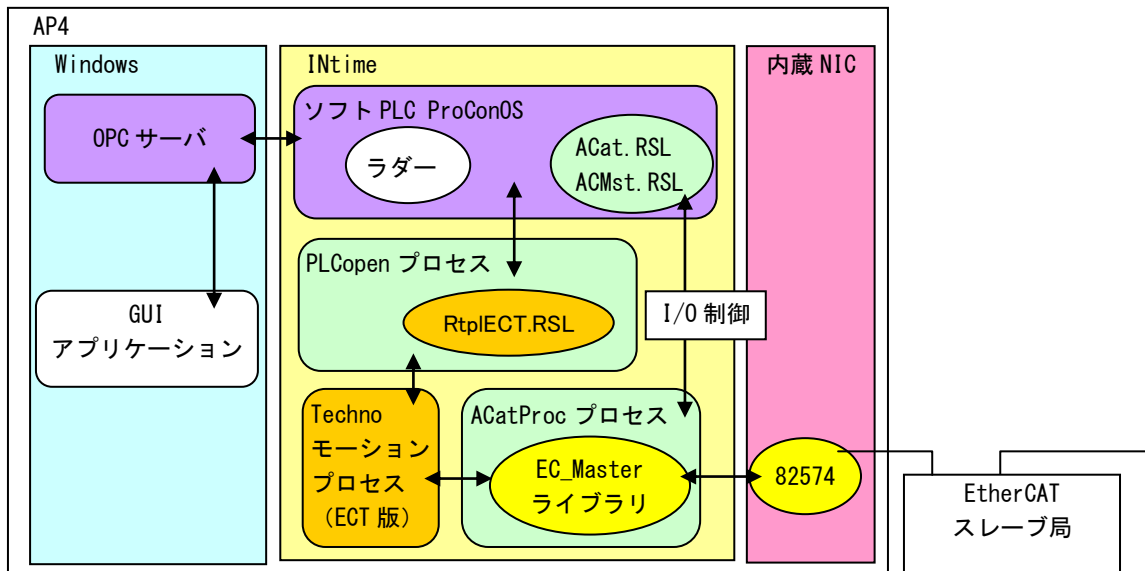


図 2-2-2-2. EtherCAT 制御構成図 (制御プログラムをラダー言語で作成する場合)

**※注：ラダープログラムを開発するためには、開発環境として、Phoenix Contact (KW-Software) 製 MULTIPROG が必要となります。**

## 第3章 標準版と AI-Motion について

本章では同期制御ができない、標準版の PLCopen と、同期制御ができる AI-Motion パッケージの違いについて説明します。

### 3-1 PLCopen の仕様について

標準版の PLCopen と AI-Motion の PLCopen について、サポートされている PLCopen の FB が違います。それぞれの FB の詳細については、リファレンスマニュアルを参照してください。

#### 1) PLCopen 仕様 MC 管理ファンクションブロック

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	AI-Motion	標準版	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

■：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

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	AI-Motion	標準版	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 本来の仕様とは違う仕様で実装されています。

## 3) PLCopen 仕様 MC 原点復帰ファンクションブロック

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	AI-Motion	標準版	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
機械的境界	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_StepFeferenceFlyingRefPulse <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

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	AI-Motion	標準版	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。

■：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。

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

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	AI-Motion	標準版	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

■：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。

### 3-2 開発環境

PLCopen 開発環境で使用できるライブラリについて、表 3-2-1 に示します。

表 3-2-1. 開発環境ライブラリ一覧

名称	ファイル名 (ディレクトリ名)	内容	AI-Motion	標準版
<b>MULTIPROG 用 FW ライブラリ</b>				
Part1, Part2, Part5 用 FWL	MP_FwLib_PlcOpenMC	位置決め用 FB のファームウェアライブラリ	○	○
Part4 用 FWL	MP_FwLib_PlcOpenMC_P4	同期制御用 FB のファームウェアライブラリ	○	—
<b>ユーザアプリケーション開発用 INtime ライブラリ</b>				
Part1, Part2, Part5 用 RSL	POMotion	位置決め用 API 関数ライブラリ	○	○
Part4 用 RSL	POMotionP4	同期制御用 API 関数ライブラリ	○	—

○ : 使用できる — : 使用できない

**※注 : AI-Motion は、(株) テクノのテクノモーションライブラリを使って、PLCopen 準拠の同期制御と通常の位置決め制御を実現しています。そのため、標準品の FB とは動作が異なる場合があります。**

### 3-3 実行環境

PLCopen 実行環境の起動プロセスの種類と違いについて表 3-3-1 に示します。

表 3-3-1. 実行環境プロセス一覧

	名称	AI-Motion プロセス名	標準版プロセス名	内容
①	MECHATROLINK-III マスタプロセス	RtpIML3Proc. RTA	MLMstProc. RTA	MECHATROLINK-III の通信制御を行うプロセス
②	EtherCAT マスタプロセス	ACatProc. RTA	ACatProc. RTA	EtherCAT の通信制御を行うプロセス
③	EtherCAT 同期モーション制御プロセス	RtpIECTProc. RTA	—	EtherCAT 版の同期モーション制御を行うプロセス
④	PLCopen 準拠モーション制御プロセス	PLCOpenTechProc. RTA	PLCOpenProc. RTA	PLCopen 準拠のモーション制御を行うプロセス
⑤	ソフト PLC プロセス	eclrIntime. rta	eclrIntime. rta	MULTIPROG で作成したプログラムをダウンロードして実行するプロセス

・ MECHATROLINK-III にて制御する場合の起動順序

①→④→⑤

・ EtherCAT にて制御する場合の起動順序

②→③→④→⑤ (標準版の場合は②→④→⑤)

## 第4章 INtime 初期設定

本章では、AI-Motion を使用する場合の INtime 初期設定方法について記述します。

### 4-1 デバイスの INtime への移動

各種省配線を INtime 上で使用するためには、それぞれのデバイスを Windows から INtime へ移動させる必要があります。

デバイス名称と省配線デバイスの対応を表 4-1-1 に示します。

表 4-1-1. デバイス名称と省配線デバイス対応表

省配線デバイス	デバイス名称	初期設定	参照先
MECHATROLINK-III	G8 MECHATROLINK Extension Board	Windows	4-1-1
EtherCAT	Intel(R) Gigabit CT Desktop Adapter	INtime	4-1-2
	Intel(R) Gigabit CT Desktop Adapter #2	Windows (INtime)	
	Intel(R) Gigabit CT Desktop Adapter #3	Windows	

初期設定で、INtime となっているデバイスは、出荷時設定ですでに INtime 側にデバイスを移動させています。EtherCAT を使用しない場合、Windows 側で LAN を 3ch 使用したい場合は、「Intel(R) Gigabit CT Desktop Adapter」を Windows へ移動させてください。

マスタ拡張ボードは、出荷時、INtime 側に移動させていません。MECHATROLINK-III 版の PLCopen 制御を使用する際は、「4-1-1 マスタ拡張ボードデバイスの移動」を参照してマスタ拡張ボードデバイスの移動を行ってください。

#### 4-1-1 マスタ拡張ボードデバイスの移動

INtime 上で、MECHATROLINK、A-Link、CUnet を使用するためには、マスタ拡張ボードのデバイスを INtime 側へ移動させる必要があります。下記の手順でデバイスを移動させます。

- ① INtime アイコンをクリックし、「INtime Configuration(C)」をクリックして、INtime Configuration を起動します。

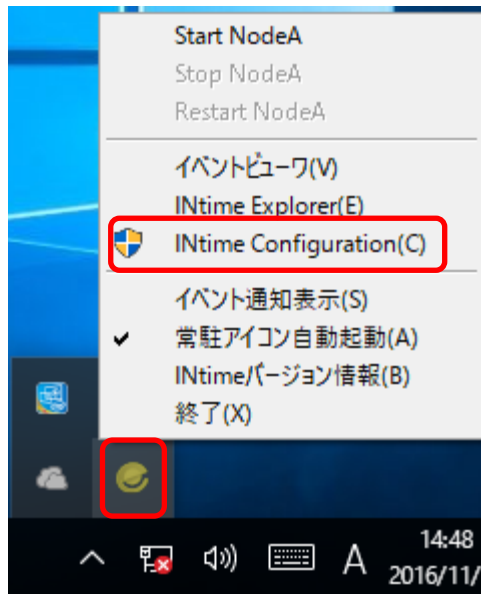


図 4-1-1-1. INtime Configuration の起動

- ② 「INtime Device Manager」をダブルクリックして、INtime Device Manager を起動します。

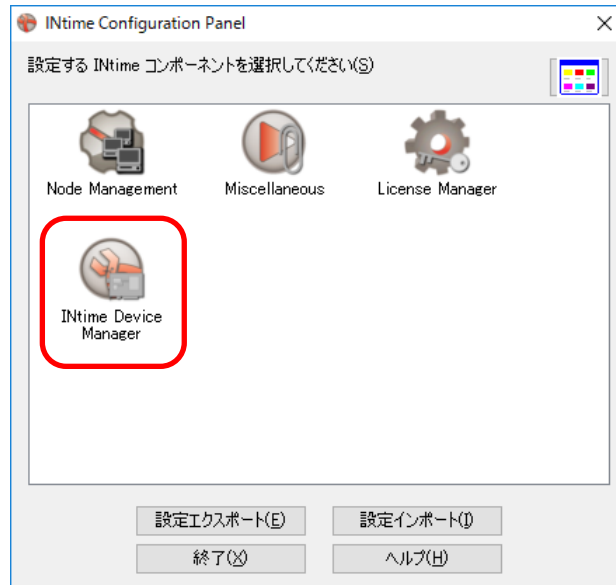


図 4-1-1-2. INtime Configuration 画面

- ③ 移動させるドライバ「G8 Option Board Driver (MSI capable)」を選択します。ドライバ名称は接続している拡張ボード型式により異なります (表 4-1-1-1 参照)。

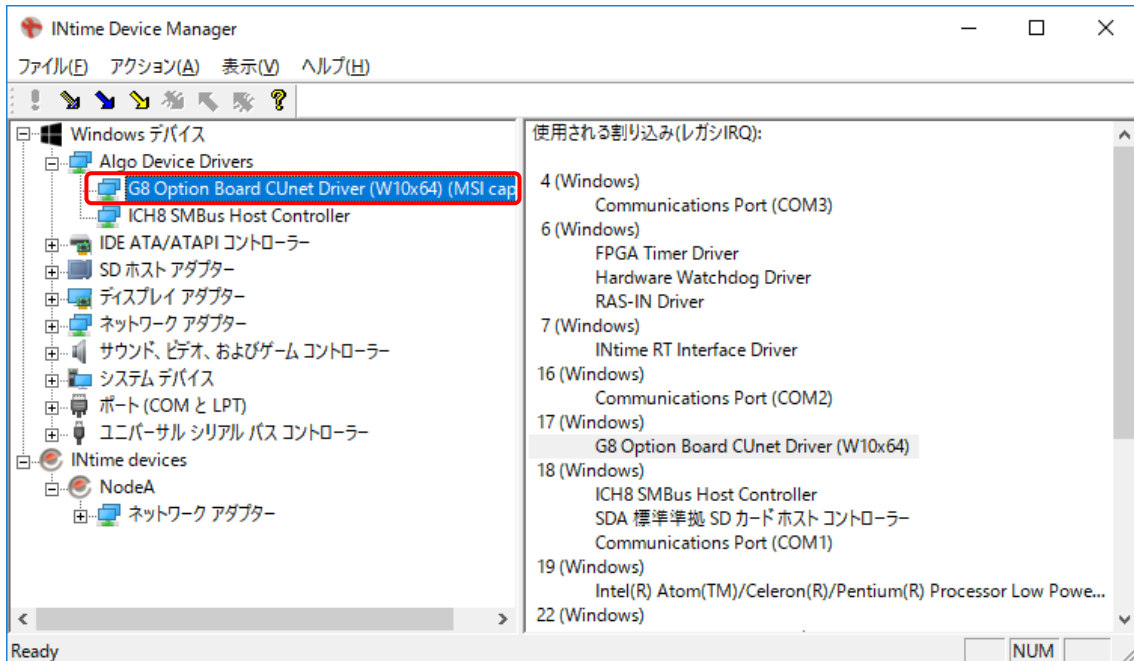


図 4-1-1-3. INtime Device Manager 画面

表 4-1-1-1. マスタ拡張ボードドライバ名称一覧

型式	省配線タイプ	ドライバ名称
AP302EX, AP303EX AP304EX	MECHATROLINK-III	G8 MECHATROLINK Extension Board (MSI capable)
	A-Link	G8 A-Link Extension Board (MSI capable)
	CUNet	G8 CU-net Extension Board (MSI capable)
AP306EX	拡張 I/O ボード	G8 Extend Board Digital I/O Driver (MSI capable)

- ④ 「アクション (A)」 → 「INtime へ移す (IRQ リソースあり)」をクリックします。

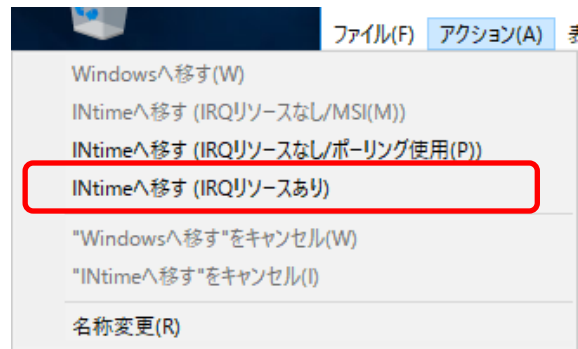


図 4-1-1-4. デバイスの移動

- ⑤ 「！」のアイコンをクリックして、デバイス移動を実行します。

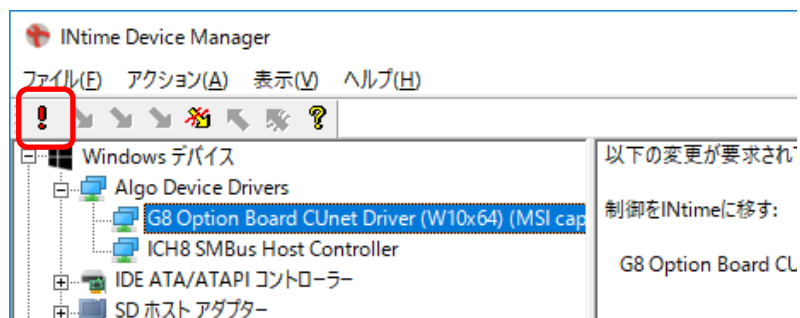


図 4-1-1-5. デバイスの移動 2

- ⑥ INtime device 側に「G8 Option Board Driver (MSI capable)」が移動されていることを確認してください。

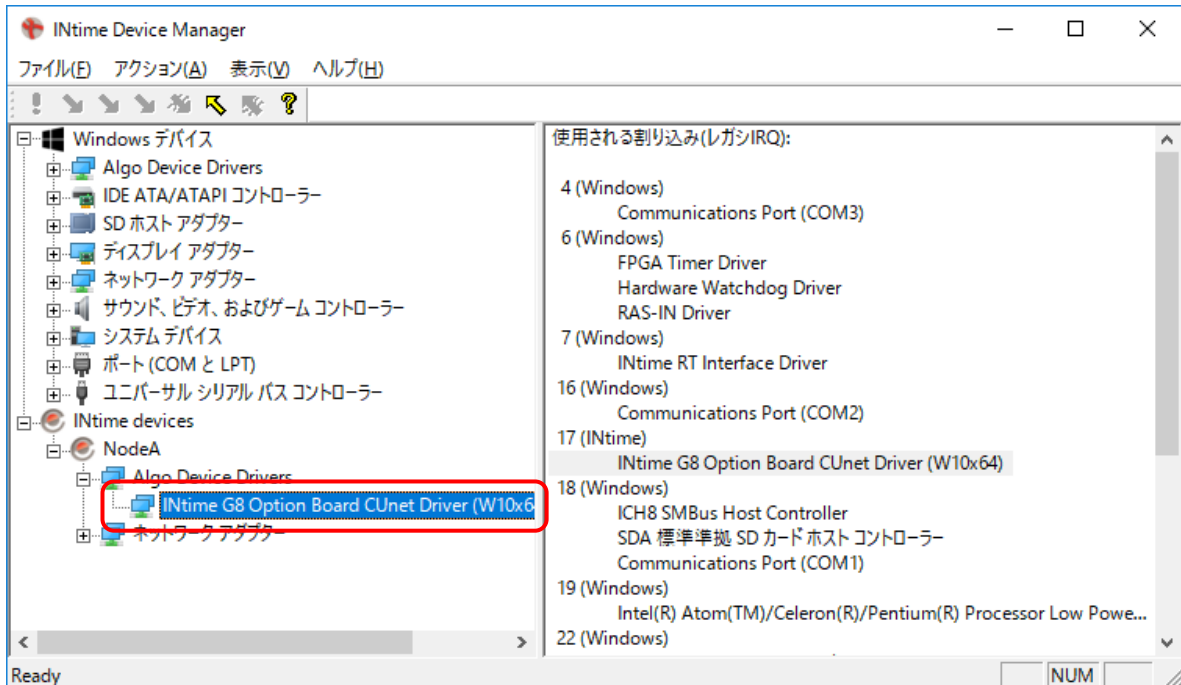


図 4-1-1-6. INtime Device Manager 画面（移動後）

※注：マスタ拡張ボードを INtime 側に移動させたとき、割り込みの競合が発生していますが、競合するデバイスはマスタ拡張ボードが接続されている PCIe デバイスのルートポートドライバなので動作に影響はありません。競合させたままで動作させます。

- ⑦ 移動後、Windows を再起動します。



### 4-1-2 Ethernet デバイスの移動

INtime 上で、EtherCAT または、Ethernet を使用するためには、LAN デバイスを INtime 側へ移動させる必要があります。

表 3-1-2 に示した弊社製 FA コンピュータには、表 4-1-2-1 に示した LAN デバイスが 3 ポート搭載されています。

表 4-1-2-1. LAN デバイス一覧

デバイス名称	ポート	備考
Intel (R) Gigabit CT Desktop Adapter	LAN1	チップセット内蔵 LAN です。出荷時設定ですでに INtime へ移動されています。
Intel (R) Gigabit CT Desktop Adapter #2	LAN2	同一の NIC を使用しているため、移動後どちらのポートが INtime 側に移ったかは実際に接続して確認する必要があります。
Intel (R) Gigabit CT Desktop Adapter #3	LAN3	AI-ImgPro が組み込まれている機種では、LAN2 は出荷時設定で INtime へ移動されています。

LAN1 ポートは出荷時状態で INtime 側に割り振られています。

LAN2 ポートは、AI-ImgPro が組み込まれている機種では INtime 側へ、それ以外では Windows 側へ割り当てられています。

LAN3 ポートは、INtime 仮想 LAN とブリッジ接続されています。

出荷時状態での、Windows で認識しているネットワークアダプタを図 4-1-2-1 に示します。

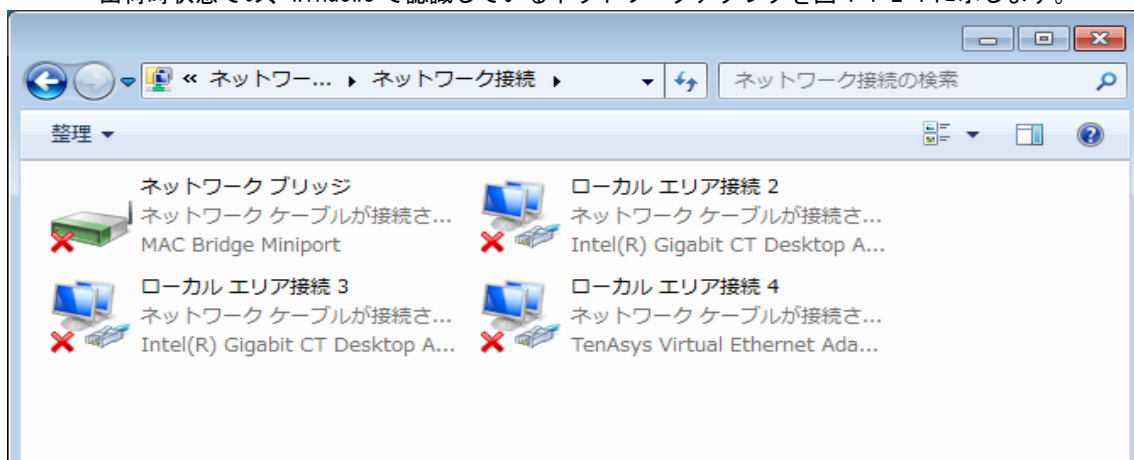


図 4-1-2-1. Windows ネットワークアダプタ

ローカルエリア接続 1 は INtime 側にネットワークデバイスが割り振られているため、Windows 側では認識できない状態になっています。

ローカルエリア接続 4 は「4-2-3 Network 設定」が有効の時、INtime カーネルが起動されたら接続されます。

ネットワークブリッジは、ローカルエリア接続 3 とローカルエリア接続 4 がブリッジ接続されています。INtime アプリケーションを別 PC 上の VisualStudio からリモートデバッグする際または、AI-PLC の ProConOS を動作させる際に使用します。

以下の手順で、ネットワークデバイスの移動を行う際は、出荷時設定が上記のようにになっていることを念頭に置いて設定してください。

## ●Windows から INtime へ移動させる場合

- ① INtime アイコンをクリックし、「INtime Configuration(C)」をクリックして、INtime Configuration を起動します。

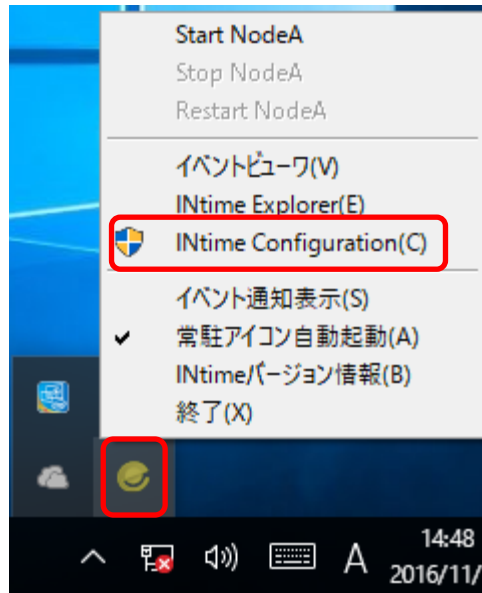


図 4-1-2-2. INtime Configuration の起動

- ② 「INtime Device Manager」をダブルクリックして、INtime Device Manager を起動します。

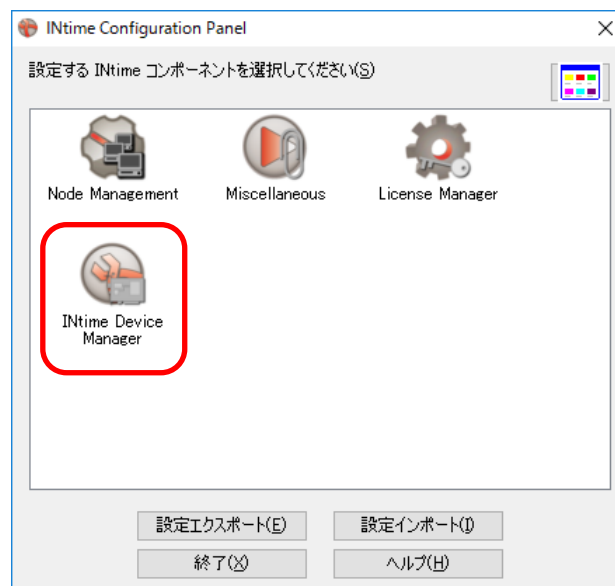


図 4-1-2-3. INtime Configuration 画面

③ INtime へ移動させたい LAN デバイスを選択します。

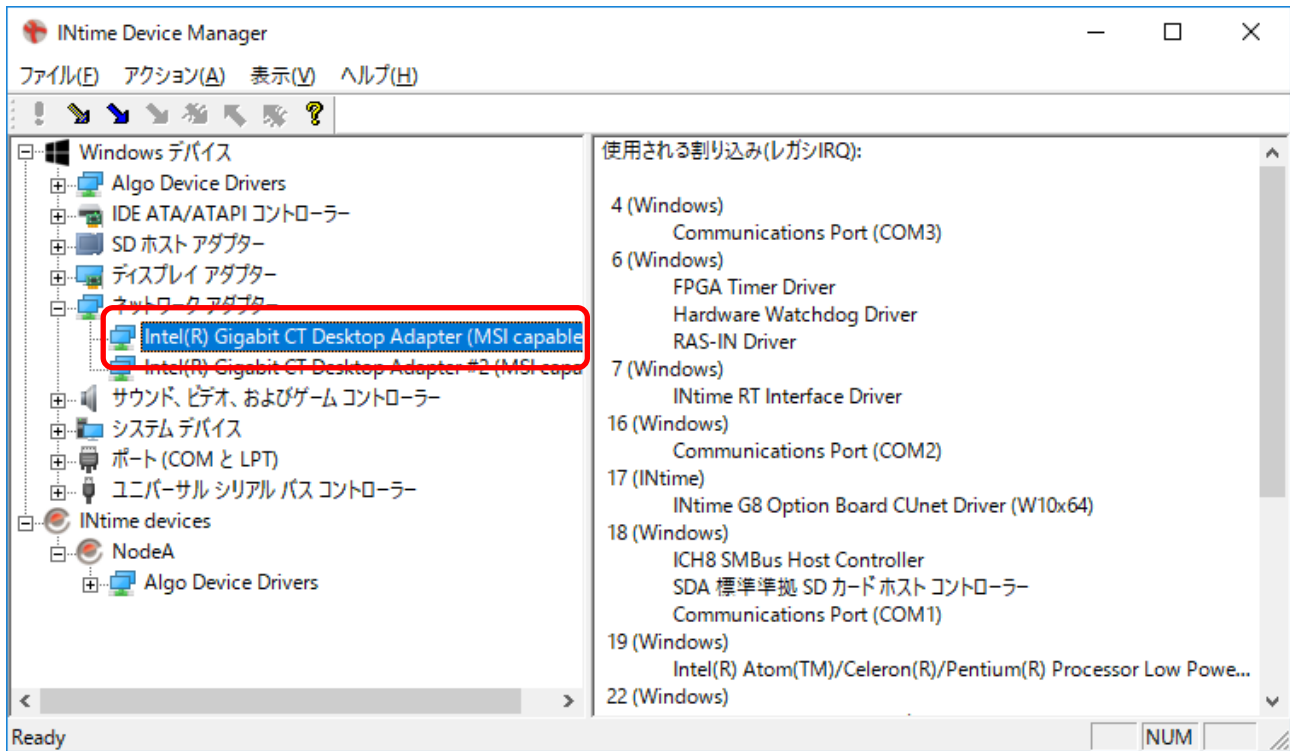


図 4-1-2-4. INtime Device Manager 画面

- ④ 「アクション (A)」 → 「INtime へ移す (IRQ リソースなし/MSI(M))」をクリックします。

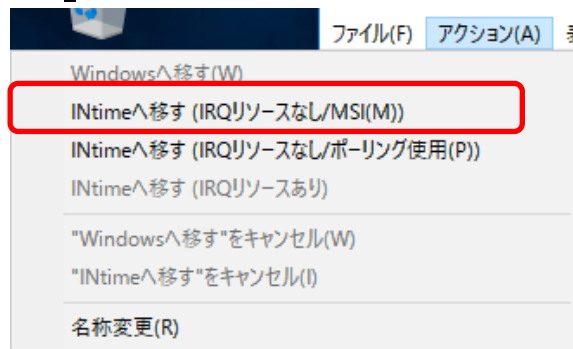


図 4-1-2-5. デバイスの移動

- ⑤ 「！」のアイコンをクリックして、デバイス移動を実行します。

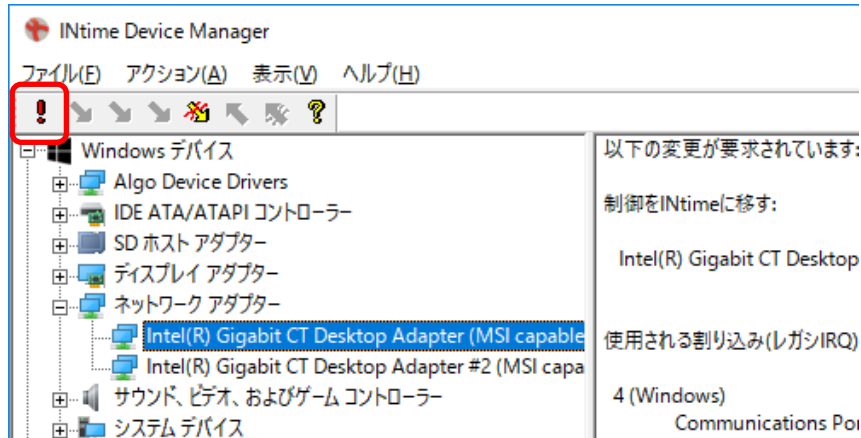


図 4-1-2-6. デバイスの移動 2

- ⑥ マスタ拡張ボードが INtime 側に移動されているとき、割り込みの競合が発生しているため、下図のような警告画面が表示されますが「はい (Y)」をクリックしてください。

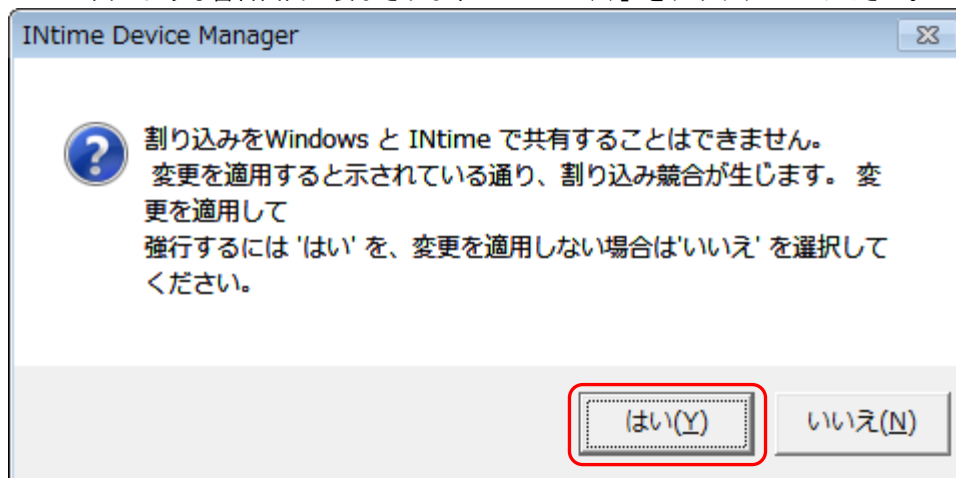


図 4-1-2-7. デバイスの移動警告画面

- ⑦ 再起動するか確認画面が起動されますので、再起動を実行してください。

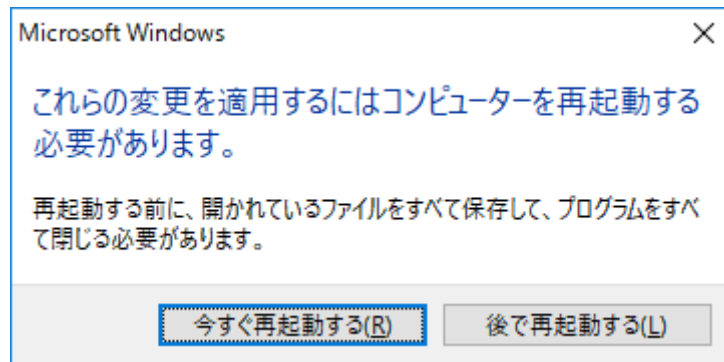


図 4-1-2-8. 再起動確認画面

- ⑧ 再起動後、再度「INtime Device Manager」を開き、INtime device 側に「Intel (R) Gigabit CT Desktop Adapter (MSI capable)」が移動されていることを確認してください。

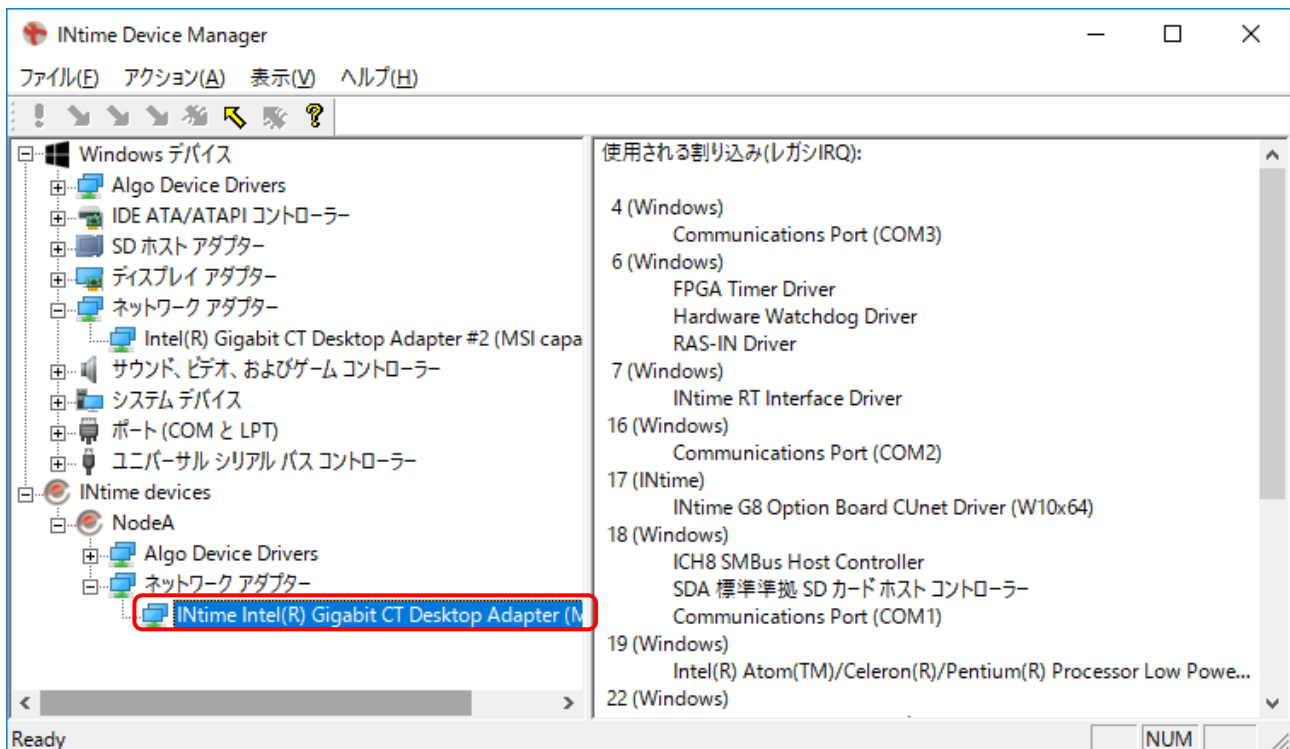


図 4-1-2-9. INtime Device Manager 画面 (移動後)

## ●INtime から Windows へ移動させる場合

- ① INtime アイコンをクリックし、「INtime Configuration(C)」をクリックして、INtime Configuration を起動します。

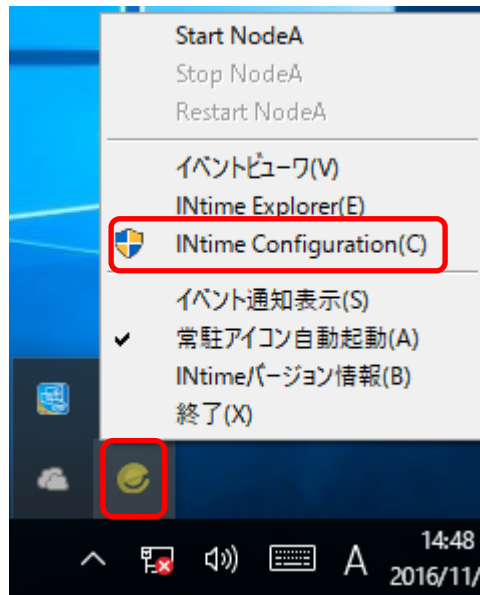


図 4-1-2-10. INtime Configuration の起動

- ② 「INtime Device Manager」をダブルクリックして、INtime Device Manager を起動します。

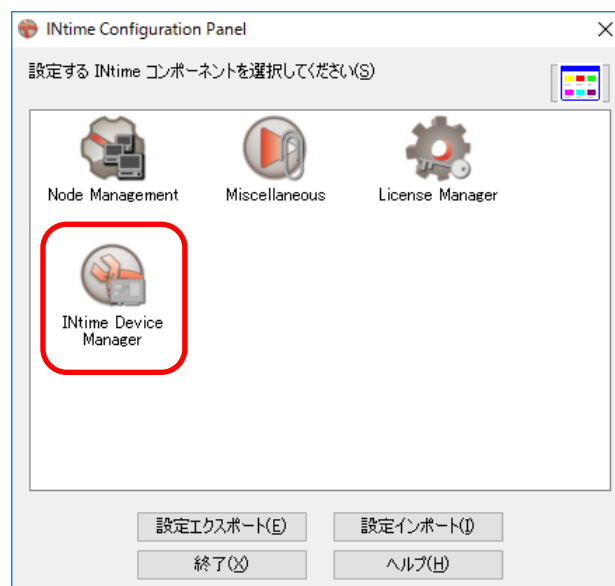


図 4-1-2-11. INtime Configuration 画面

- ③ Windows へ移動させたい LAN デバイスを選択します。

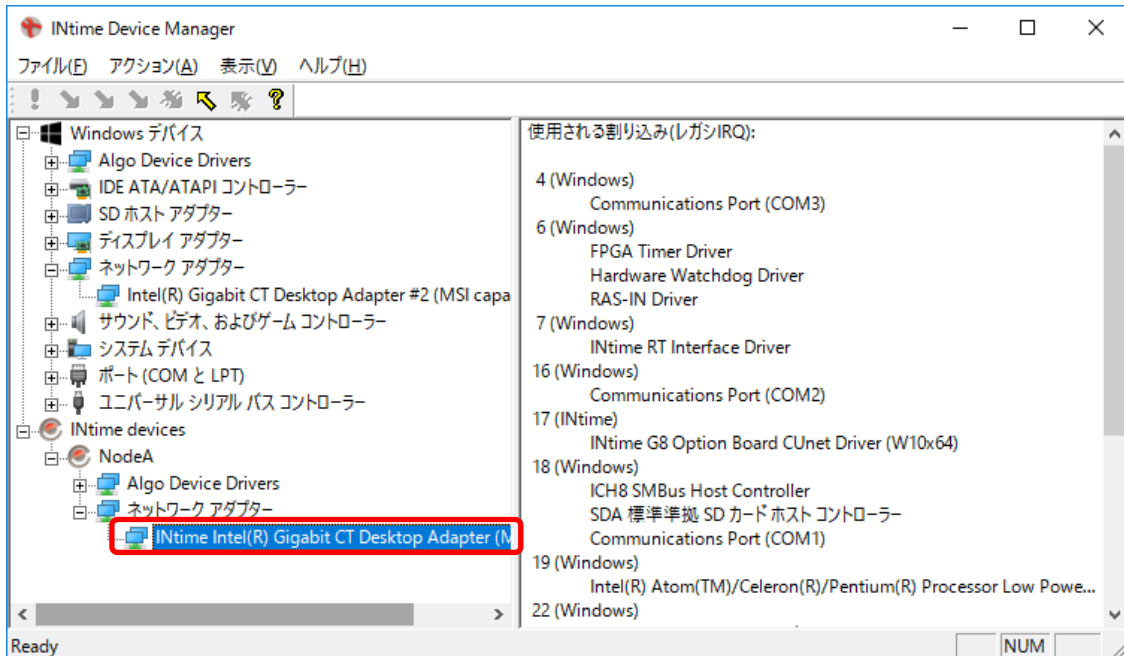


図 4-1-2-12. INtime Device Manager 画面

- ④ 下図のボタンをクリックします。

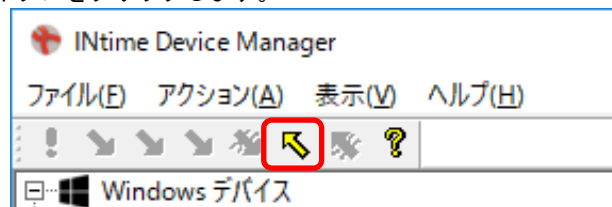


図 4-1-2-13. デバイスの移動

- ⑤ 「！」のアイコンをクリックして、デバイス移動を実行します。

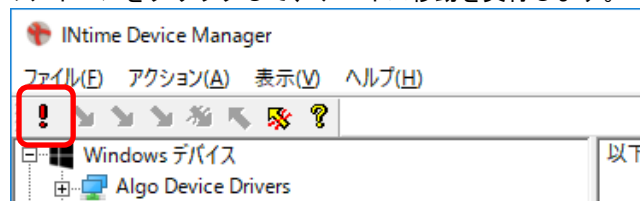


図 4-1-2-14. デバイスの移動 2

- ⑥ マスタ拡張ボードが INtime 側に移動されているとき、割り込みの競合が発生しているため、下図のような警告画面が表示されますが「はい (Y)」をクリックしてください。

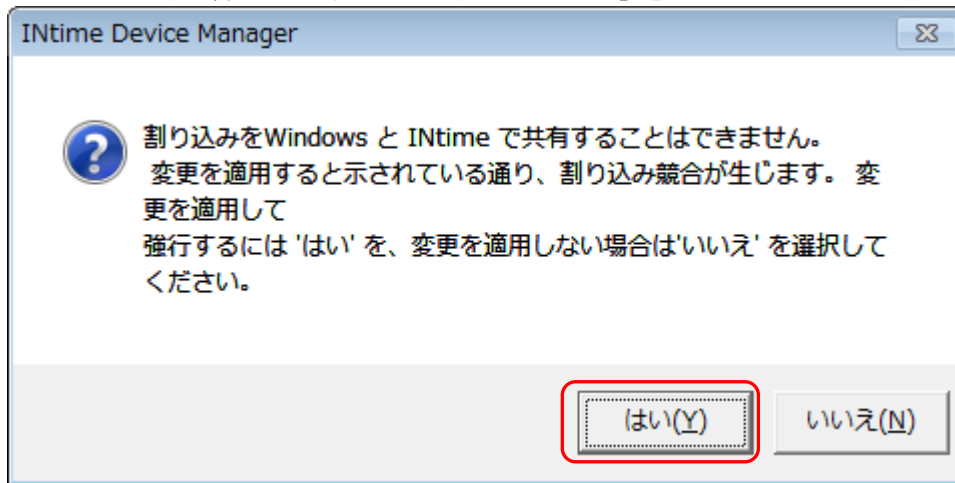


図 4-1-2-15. デバイスの移動警告画面

- ⑦ 再起動するか確認画面が起動されますので、再起動を実行してください。

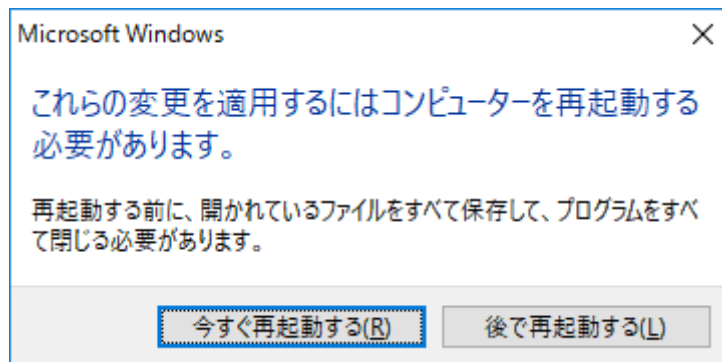


図 4-1-2-16. 再起動確認画面



- ⑧ 再起動後、再度「INtime Device Manager」を開き、Windows デバイス側に「Intel (R) Gigabit CT Desktop Adapter」が移動されていることを確認してください。

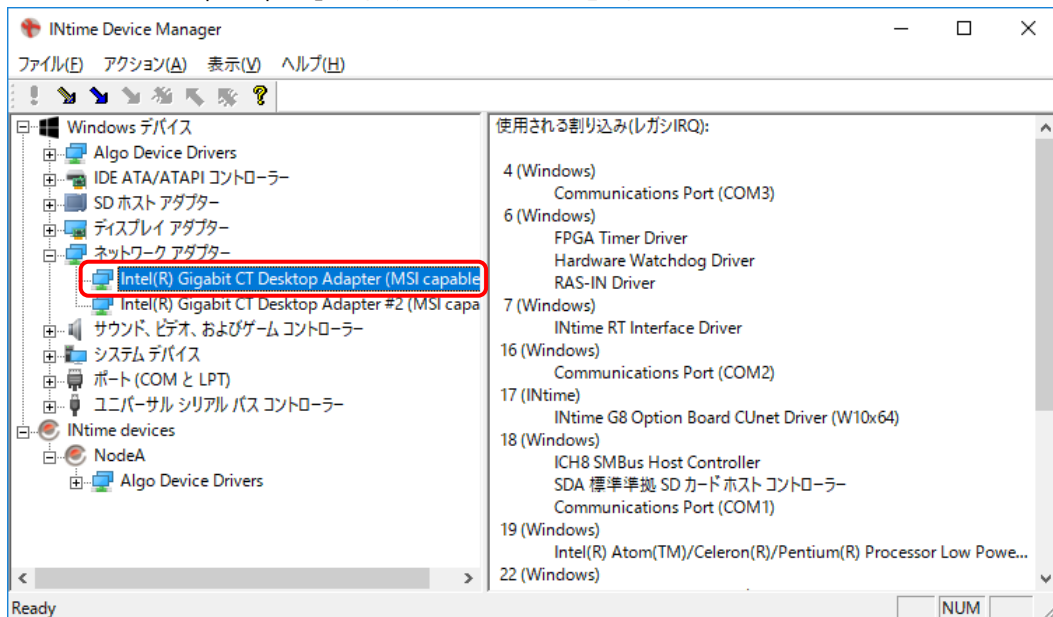


図 4-1-2-17. INtime Device Manager 画面（移動後）

※注：INtime 側にデバイスに移した後、再度 Windows 側に戻したとき、Windows は再度ドライバをインストールし直します。このとき、LAN デバイスの場合、デバイスドライバの名称に付属するナンバリングがインクリメントされます。元に戻すためには、Windows のレジストリを設定し直す必要があります。LAN のデバイス名で直接制御していない限りは、LAN 自体の動作に影響はありません。

## 4-2 INtime ノードマネジメント設定

AI3 パッケージ (AI-PLC、AI-Motion、AI-ImgPro) を使用する場合は、INtime のノードマネジメント設定値を出荷時設定から変更する必要があります。

本項では、設定変更が必要な部分の設定方法を記述します。本項にないパラメータの意味と設定方法については INtime のヘルプおよびマニュアルを参照ください。

**INtime についての技術的な質問は、直接 TenAsys 社、マイクロネット社にお問い合わせしないでください。弊社にお問い合わせください。**

INtime コンフィグレーションツールは下記の手順で起動してください。

- ① INtime アイコンをクリックし、「INtime Configuration(C)」をクリックして、INtime Configuration を起動します。

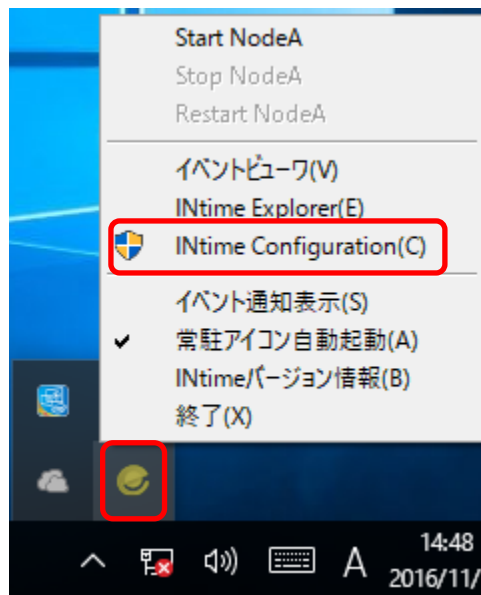


図 4-2-1. INtime Configuration の起動

- ② 「Node Management」をダブルクリックして、Node Management を起動します。

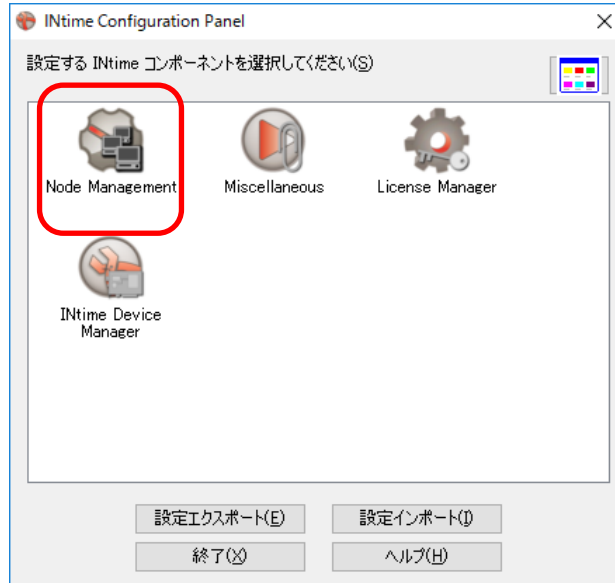


図 4-2-2. INtime Configuration 画面

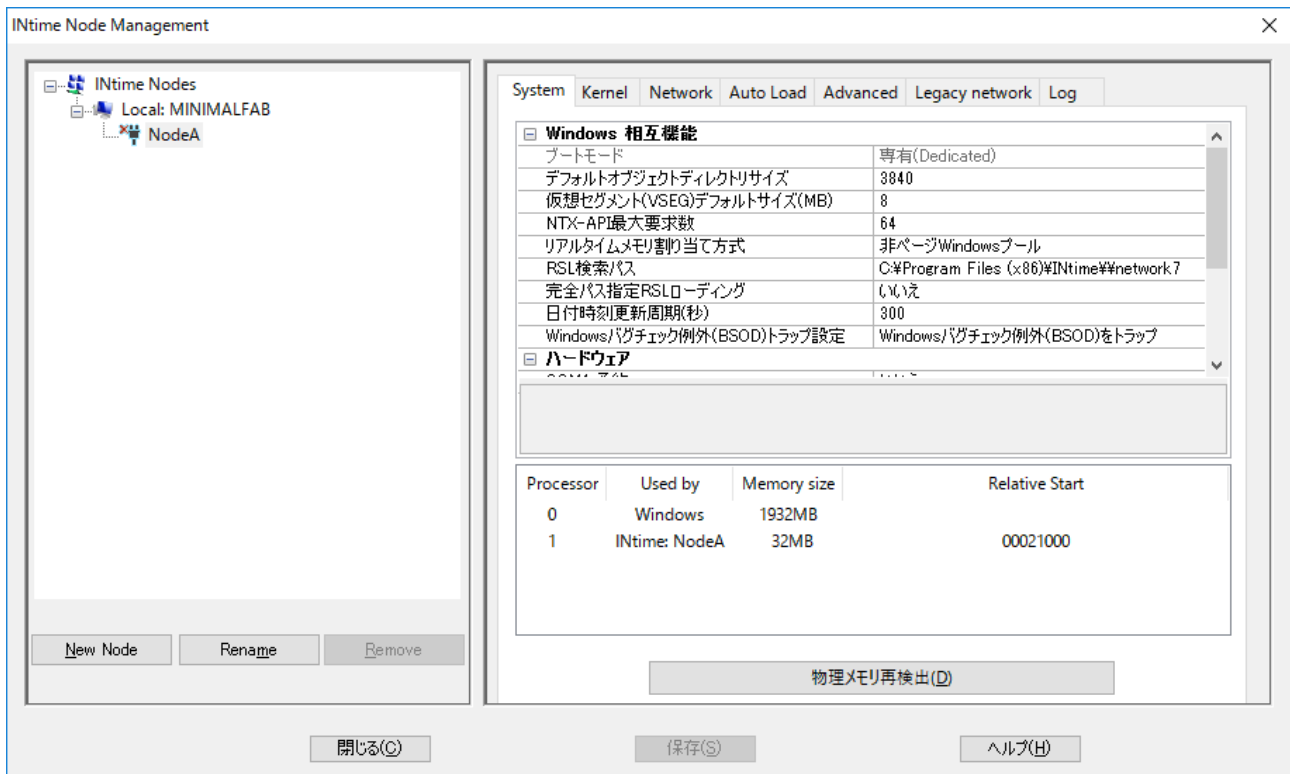


図 4-2-3. INtime Node Management 画面

### 4-2-1 System Wide 設定

System Wide 設定は Windows 相互機能設定と、ローカルハードウェアの設定を行います。設定画面は図 4-2-1-1 のようになります。

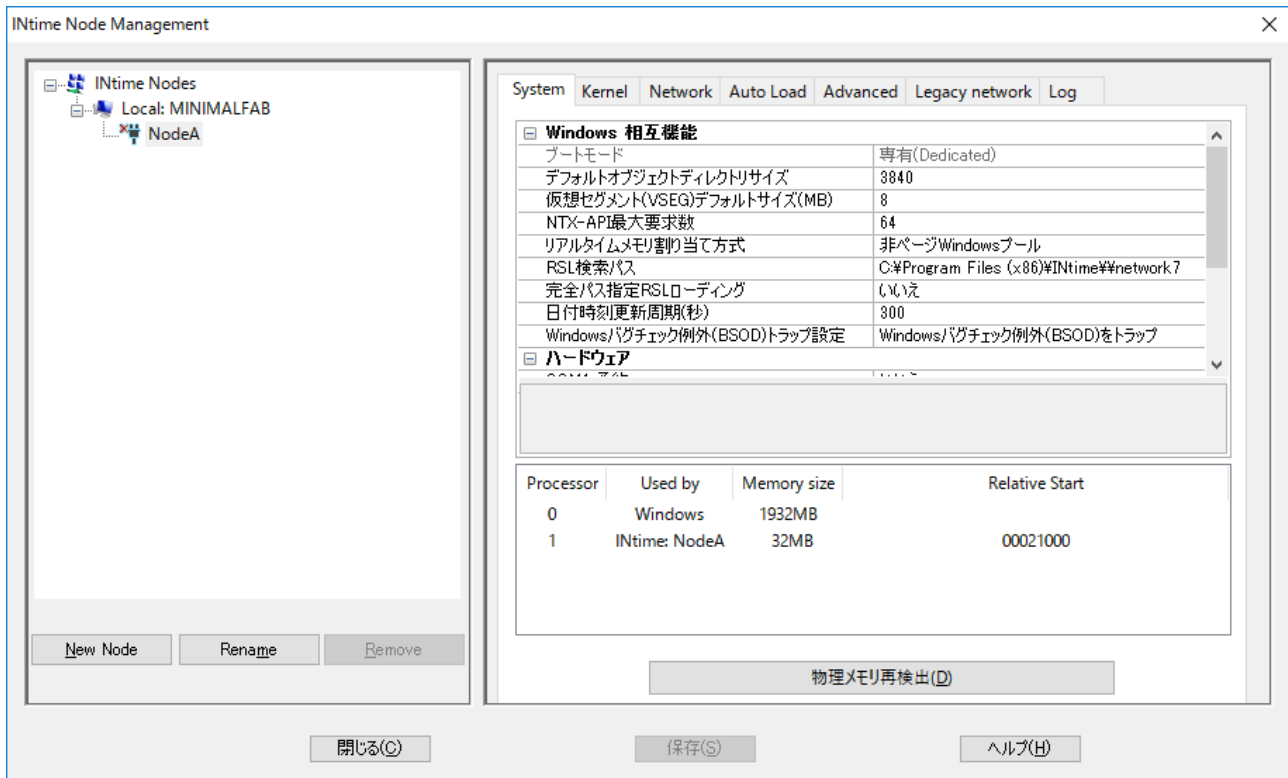


図 4-2-1-1. INtime Node Management (System Wide) 設定画面

各設定項目の出荷時設定値と、パッケージ毎の設定変更値一覧を示します。

表 4-2-1-1. INtime Node Management (System Wide) 設定表

Windows 相互機能		
名称	出荷時 設定値	備考
ブートモード	専有 (Dedicated)	
デフォルトオブジェクトディレクトリサイズ	3840	
仮想セグメントデフォルト値 (MB 値)	8	
NTX-API 最大要求数	64	
リアルタイムメモリ割り当て方式	非ページ Windows プール	
RSL 検索パス	空白	
日付時刻の同期更新周期	300	
Windows バグチェック例外 (BSOD) トラップ設定	Windows バグチェック例外 (BSOD) をトラップ	
ハードウェア		
名称	出荷時 設定値	備考
COM1 予約	いいえ	INtime 側でシリアルポートや LPT ポートを使用する場合は、はいを設定します。
COM2 予約	いいえ	
COM3 予約	いいえ	
COM4 予約	いいえ	
LPT1 予約	いいえ	

設定変更後、「保存」ボタンをクリックします。変更内容によっては、INtime Node Management 画面と、INtime Configuration Panel 画面を終了したとき再起動を要求されます。

### 4-2-2 Kernel 設定

Kernel 設定は INtime カーネルのチック周期や自動起動、カーネルで使用するメモリサイズ等の設定を行えます。設定画面は図 4-2-2-1 のようになります。

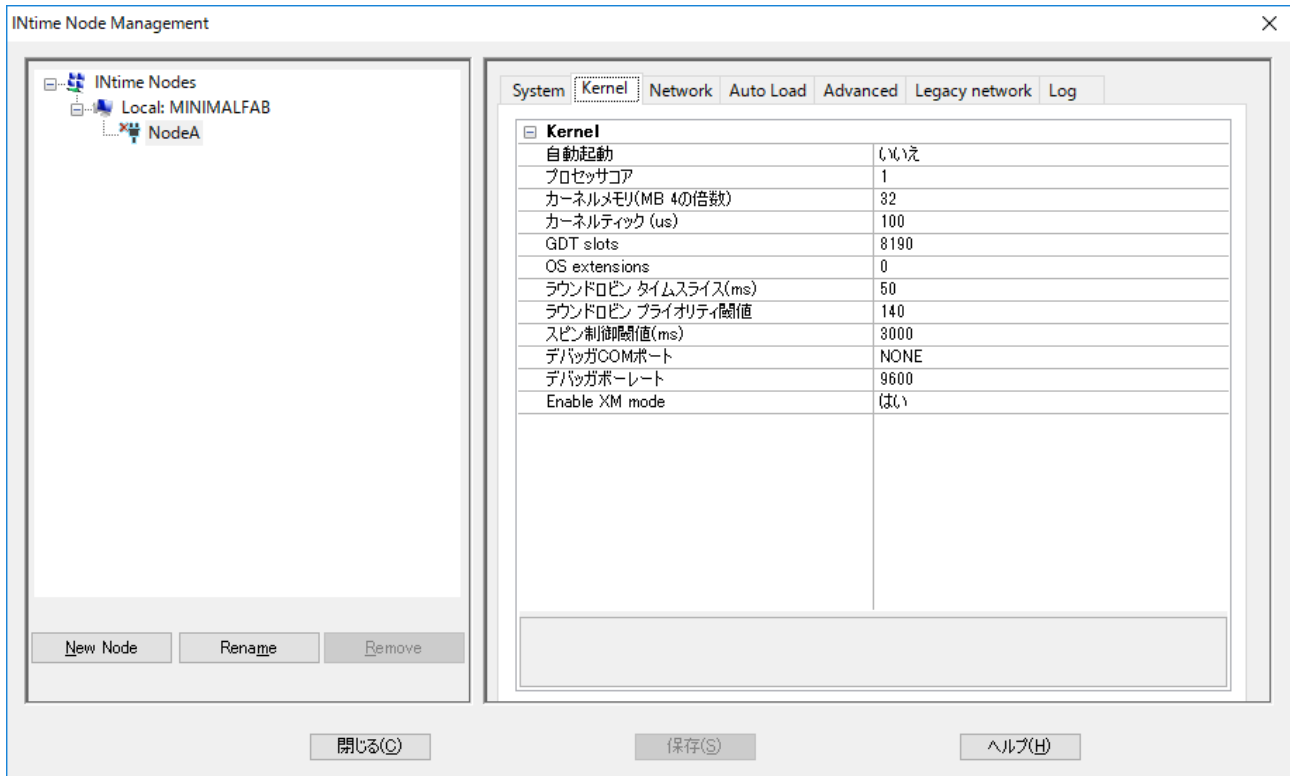


図 4-2-2-1. INtime Node Management (Kernel) 設定画面

各設定項目の出荷時設定値と、パッケージ毎の設定変更値一覧を示します。

表 4-2-2-1. INtime Node Management (Kernel) 設定表

カーネル			
名称	出荷時 設定値	備考	
自動起動	はい		
プロセッサコア	1		
カーネルメモリ (MB)	384 (504)	AI-ImgPro が組み込まれている機種では、504 が初期設定値となります。	
カーネルティック (us)	100		
GDT Slots	8190		
OS Extensions	0		
ラウンドロビン タイムスライス (ms)	50		
ラウンドロビン プライオリティ閾値	140	。	
スピン制御 閾値 (x 10 ms)	300		
デバッガ COM ポート	NONE		
デバッガボーレート	9600		

設定変更後、「保存」ボタンをクリックします。変更内容によっては、INtime Node Management 画面と、INtime Configuration Panel 画面を終了したとき再起動を要求されます。

### 4-2-3 Network 設定

Network 設定は Windows との仮想ネットワークの構築のために使用します。また、INtime アプリケーションでイーサネット機器を使用する際にも設定します。

出荷時設定では仮想 LAN のみ有効となっています。再度、仮想 LAN を有効としたい場合または、Ethernet として LAN ポートを使用したい場合は下記手順により設定してください。

- ① 「NodeA」を選択し、「Network」タブをクリックします。図 4-2-3-1 のような画面が表示されます。「ネットワーク自動起動」を「はい」に設定します。次に、「ネットワークインターフェース (NIC) リスト」項目を選択し、右端の「...」をクリックします。

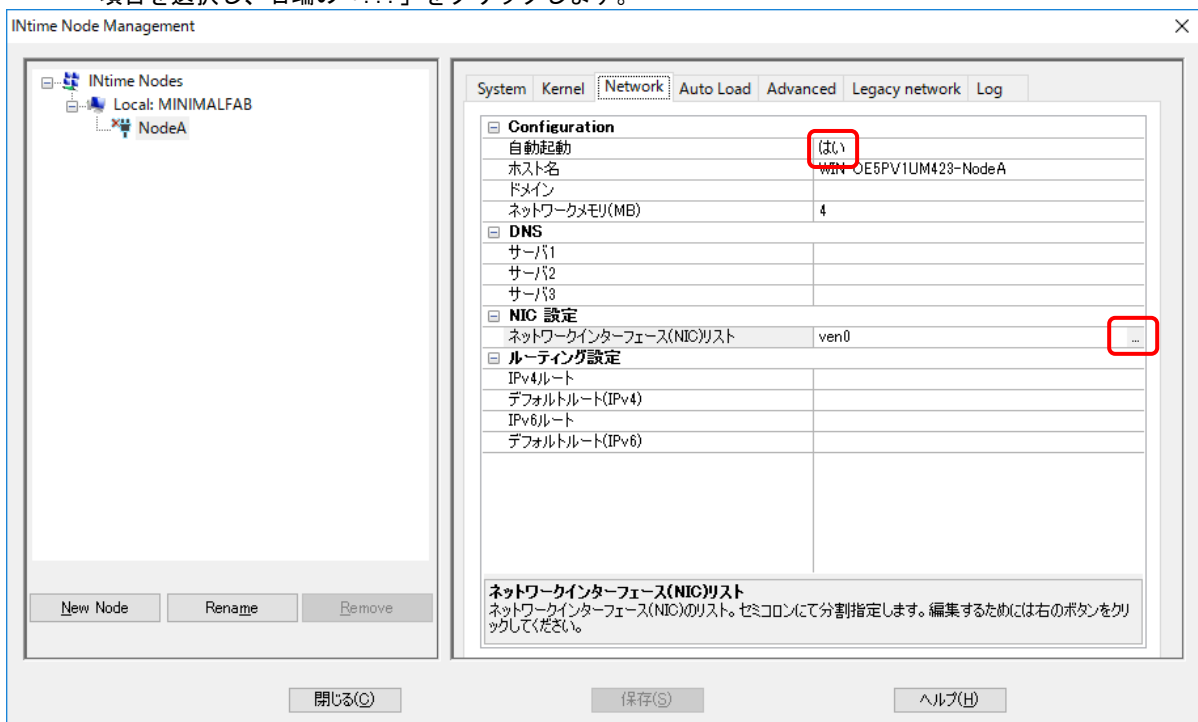


図 4-2-3-1. INtime Node Management のネットワーク設定画面

- ② 図 4-2-3-2 のような画面が起動します。仮想 LAN の設定を変更する場合は「編集」ボタンをクリックします。新規に LAN ポートを追加したい場合は「追加」ボタンをクリックします。

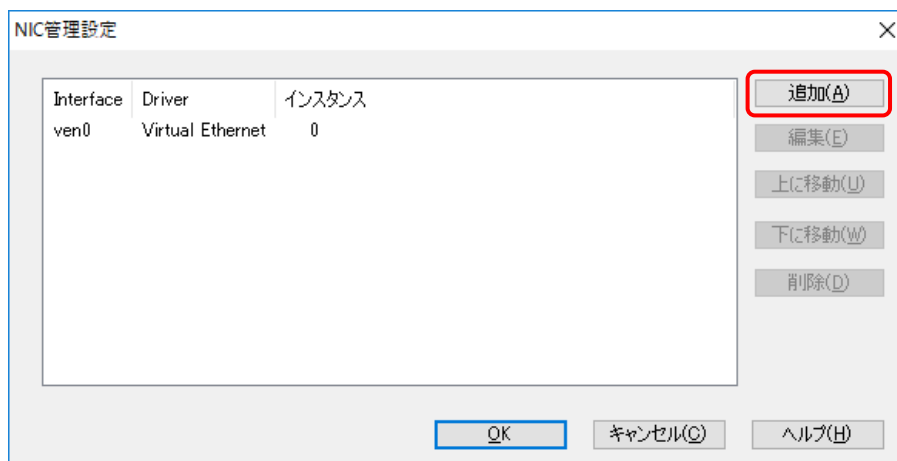


図 4-2-3-2. NIC 管理設定画面

- ③ 下図のような画面が起動します。「インターフェース種類」のドロップダウンリストから LAN デバイスを選択します。

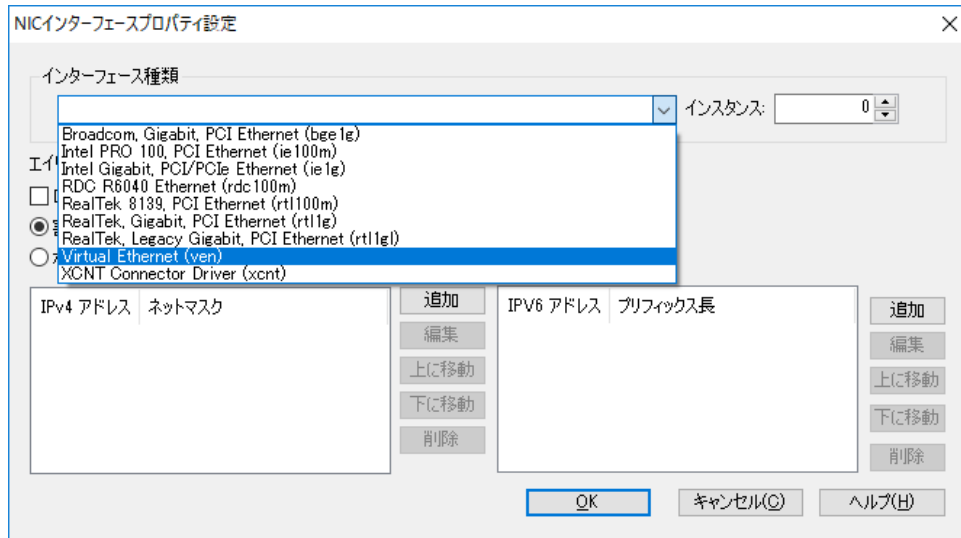


図 4-2-3-3. NIC インターフェースプロパティ設定画面

仮想 LAN を使用する場合は「Virtual Ethernet (ven)」を選択してください。LAN ポートを使用する場合は、「Intel Gigabit, PCI/PCIe Ethernet (ie1g)」を選択してください。

- ④ 「追加」または「編集」ボタンをクリックし、IPv4 の固定 IP アドレスを設定します。仮想 LAN の場合は出荷時設定で「IP アドレス : 192.168.0.10 サブネットマスク : 255.255.255.0」が設定されています。

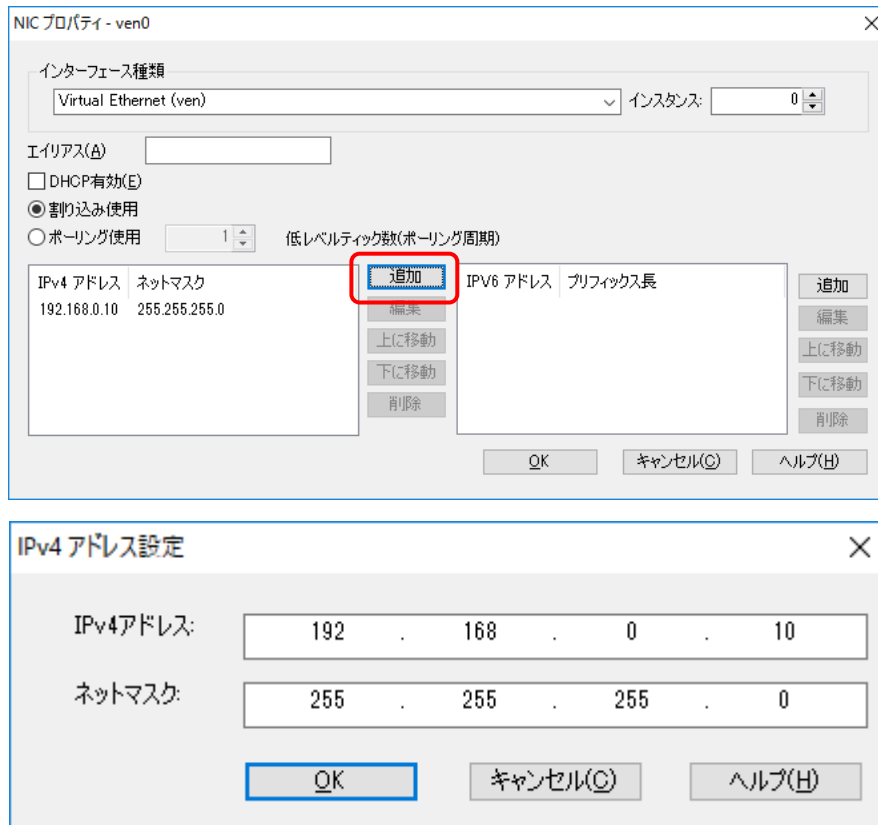


図 4-2-3-4. NIC インターフェース IPv4 アドレス設定画面



- ⑤ 「OK」ボタンをクリックし、図 4-2-3-1 の画面に戻った後、「保存」をクリックします。これでネットワークの設定は完了します。

#### 4-2-4 AutoLoad 設定

AutoLoad 設定は INTIME 上で動作するアプリケーションをカーネル起動時に自動起動するかどうかを設定します。設定画面は図 4-2-4-1 のようになります。

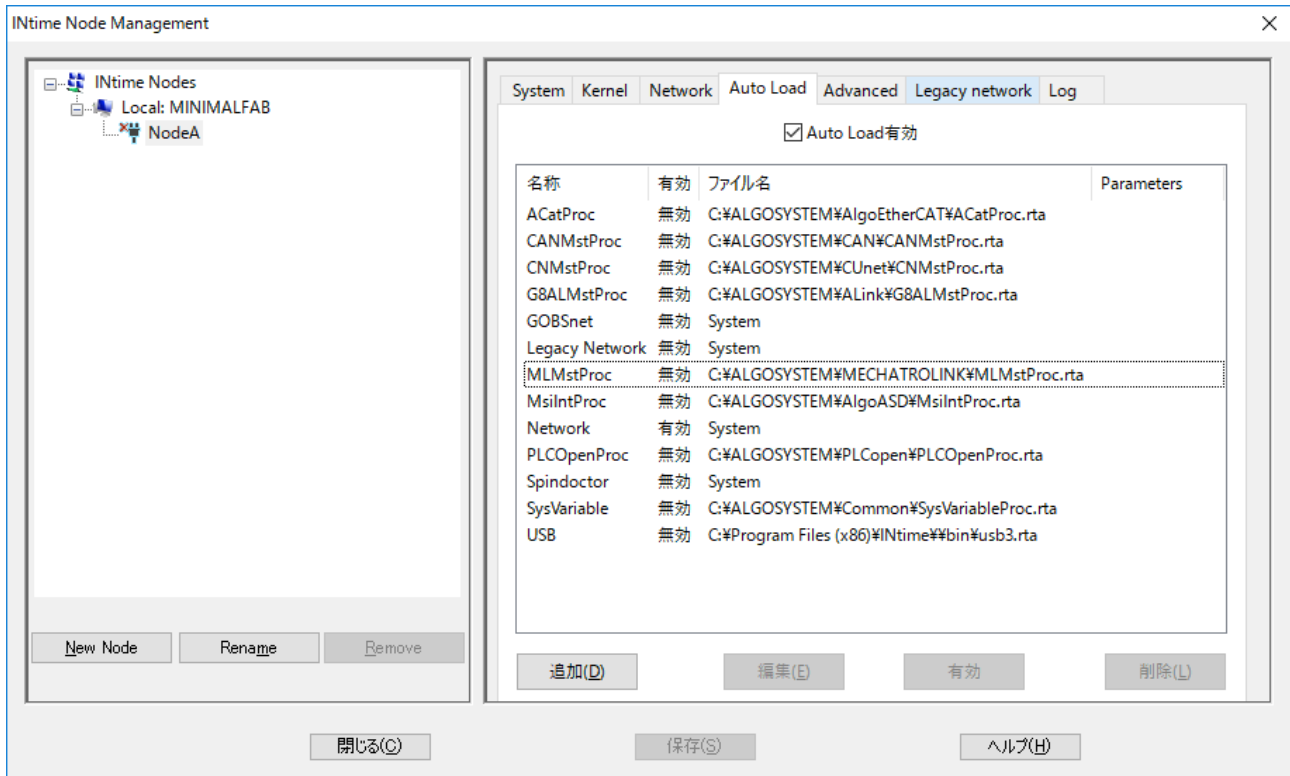


図 4-2-4-1. INTIME Node Management (AutoLoad) 設定画面

使用する省配線の種類によって、自動起動すべきプロセスと、依存関係を変更する必要があります。各設定項目の出荷時設定値と、パッケージ毎の設定変更値一覧を示します。

表 4-2-4-1. INtime Node Management (AutoLoad) 設定表

INtime カーネルプロセス				
名称	出荷時設定値			
	自動 起動	INIT 待機	依存関係	
			初期値	備考
iWin32	有効	—	—	
Legacy Network	無効	—	—	
Network	有効	—	—	
SpinDoctor	有効	—	—	
USB	無効	有効	依存無し	
各種省配線プロセス				
名称	出荷時設定値			
	自動 起動	INIT 待機	依存関係	
			初期値	備考
ACatProc (EtherCAT マスタプロセス)	無効	有効	SysVariable	
G8ALMstProc (A-Link マスタプロセス)	無効	有効	依存無し	
MLMstProc (MECHATROLINK-III マスタプロセス)	無効	有効	Network	
CNMstProc (CUnet マスタプロセス)	無効	有効	依存無し	
SysVariable (システム変数プロセス)	無効	有効	依存無し	
SramProc (電源断データ保持プロセス)	有効	有効	依存無し	
PLCOpenProc (PLCopen 制御プロセス)	無効	有効	依存無し	EtherCAT 使用時は ACatProc を追加します。 MECHATROLINK 使用時は MLMstProc を追加します。 A-Link 使用時は、G8ALMstProc を追加します。

AI-Motion プロセス				
プロセス名称	出荷時設定値			
	自動 起動	INIT 待機	依存関係	
			初期値	備考
PLCOpenTechProc (PLCopen 同期制御プロセス)	無効	有効	RtpIECTProc	
RtpIECTProc (同期制御プロセス)	無効	有効	AGatProc	
AI-PLC プロセス				
名称	出荷時設定値			
	自動 起動	INIT 待機	依存関係	
			初期値	備考
ProConOS (ソフト PLC プロセス)	無効	有効	Network SramProc	同期モーション使用時は PLCOpenTech Proc を追加します。 A-Link 使用時は、G8ALMstProc を追加し ます。 PLC 上にて Windows ハングアップ検知の 機能を使用する場合は、WHWDTRT を追加 します。 PLC と Windows 間の共有メモリを使用す る場合は、SMemProc を追加します。
WHWDTRT (Windows ハングアップ検知機能)	無効	有効	依存無し	
SMemProc (ソフト PLC⇔Windows 間 共有メモリ I/F)	無効	有効	依存無し	SMemProc.ini ファイルの設定に従い共有 メモリが準備されます。

設定を変更する場合、下記の手順で変更してください。

- ① 設定変更したいプログラムを選択し、「編集」ボタンをクリックします。新規に自動起動するアプリケーションを追加したい場合は「追加」ボタンをクリックします。

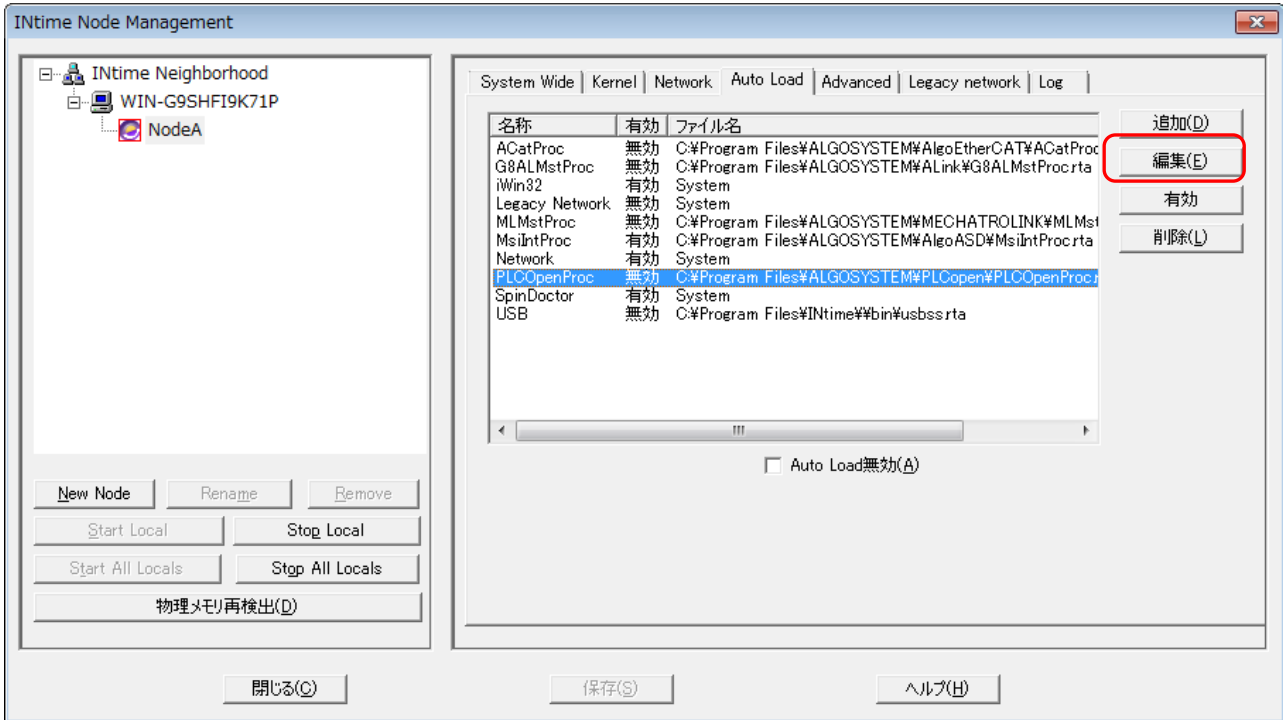


図 4-2-4-2. INtime Node Management (AutoLoad) 設定画面 2

- ② 下図のような設定画面が表示されます。それぞれの設定値の意味を表 4-2-4-2 に示します。

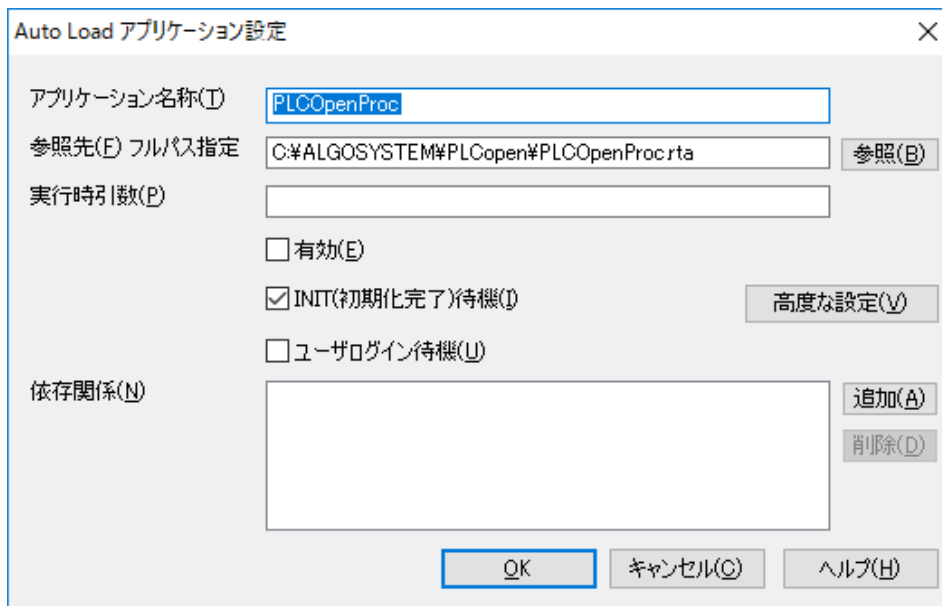


図 4-2-4-3. Auto Load アプリケーション設定画面

表 4-2-4-2. Auto Load アプリケーション設定表

#	設定項目	内容
1	アプリケーション名称	図 7-2-4-2 の一覧に表示されるアプリケーション名称です。名称は AutoLoad リストの中で一意である必要があります。
2	参照先 フルパス指定	起動する INtime アプリケーション (RTA ファイル) をフルパス指定します。
3	実行時引数	2 で指定したアプリケーションの起動時に引数が必要な場合、ここに指定します。
4	有効	自動起動を有効とするか、無効とするかを指定します。チェックが入っていると有効です。有効時は、INtime カーネルの起動時に自動実行されます。
5	INIT (初期化完了) 待機	この設定を有効とした場合、アプリケーションの初期化が完了するまで、次のプログラムの実行を待機します。 この機能を使用するためには、アプリケーションの初期化完了時に SynchronizeRtLoader 関数をコールします。
6	ユーザログイン待機	この設定を有効とした場合、自動起動処理は Windows のユーザログイン、およびデスクトップの初期化まで待機します。
7	依存関係	本アプリケーションが実行される前に、起動されるべきアプリケーションを設定します。「追加」ボタンをクリックすると、図 4-2-4-4 のような追加できるアプリケーションのリストが表示されます。 設定例として PLCOpenProc の場合は、EtherCAT を使用するときは ACatProc を指定します。MECHATROLINK を使用するときは MLMstProc を指定します。



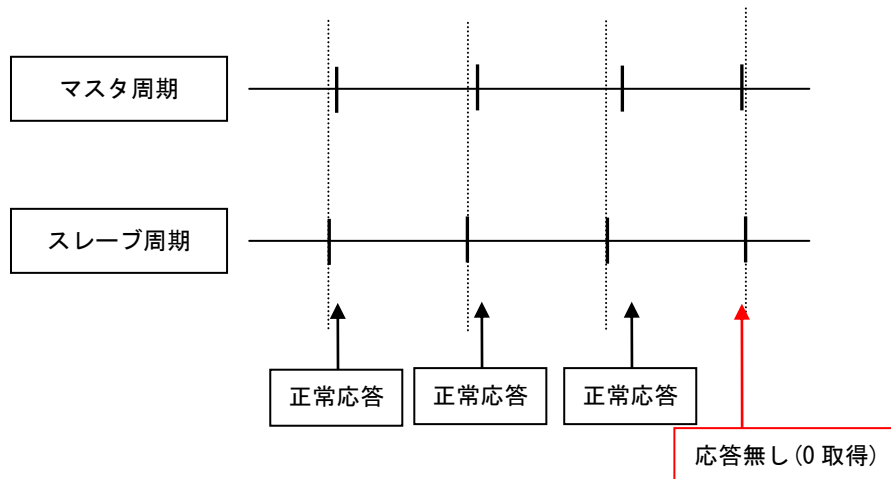
図 4-2-4-4. Auto Load 依存アプリケーション設定画面 (PLCOpenProc の場合)

- ⑥ 「OK」ボタンをクリックし、図 4-2-4-1 の画面に戻った後、「保存」をクリックします。これで Auto Load の設定は完了します。

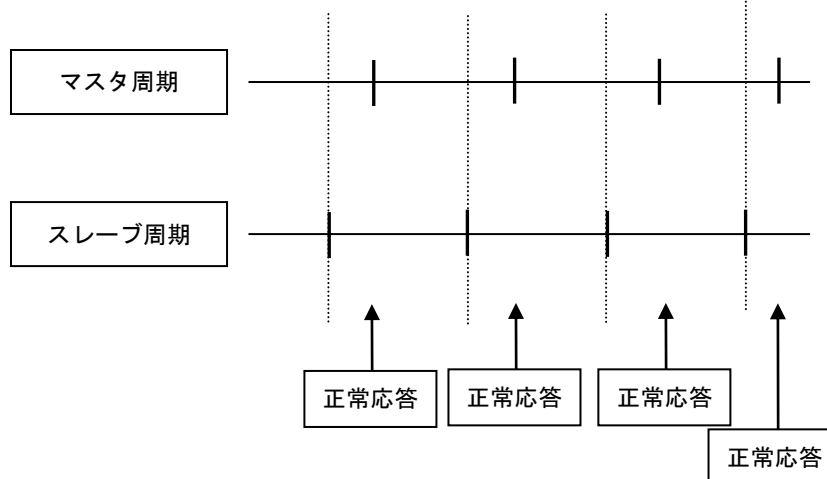
### 4-3 INtime レジストリキー登録について

EtherCAT 通信にて、DC モードと呼ばれる同期通信機能があります。これは、EtherCAT スレーブ間の制御周期を合わせるための機能です。DC モードに対応されたスレーブ同士を接続したとき、先頭のスレーブの制御周期に全スレーブが合わせられます。

DC モードでは、マスタとスレーブ間の制御周期を合わせる機能は持っていないため、合わせるためにはマスタ側がスレーブの制御周期に合わせる必要があります。この周期が合っていない場合、スレーブによっては、正常なデータが受信できない場合があります。



1) マスタ周期とスレーブ周期の更新タイミングがほぼ同じ場合



2) マスタ周期とスレーブ周期の更新タイミングが離れている場合

図 4-3-1. 制御周期のズレと応答について

図 4-3-1 はマスタとスレーブの制御周期のズレ方によって、スレーブから正常に回答されるかどうか比較した図です。

1)、2) 共に、マスタ周期とスレーブ周期は同じですが、1) ではマスタ周期の微妙なズレ (数 us 以内) がスレーブからの応答に影響を与えます。あるメーカーのサーボパックでは 1 スレーブ周期に 1 回のみ応答する仕様になっています。そのため、1) のように 1 スレーブ周期にマスタから 2 回データ更新が発生した場合、2 回目の応答はすべて 0 になります。

INtime には、図 4-3-2 に示したように Windows に「RtClockAdjust」というレジストリキーを設定することで、INtime で使用する制御周期を調整する機能が存在します。この調整により、マスタ制御周期を EtherCAT スレーブの制御周期に合わせることで、0 応答を無くすることができます。

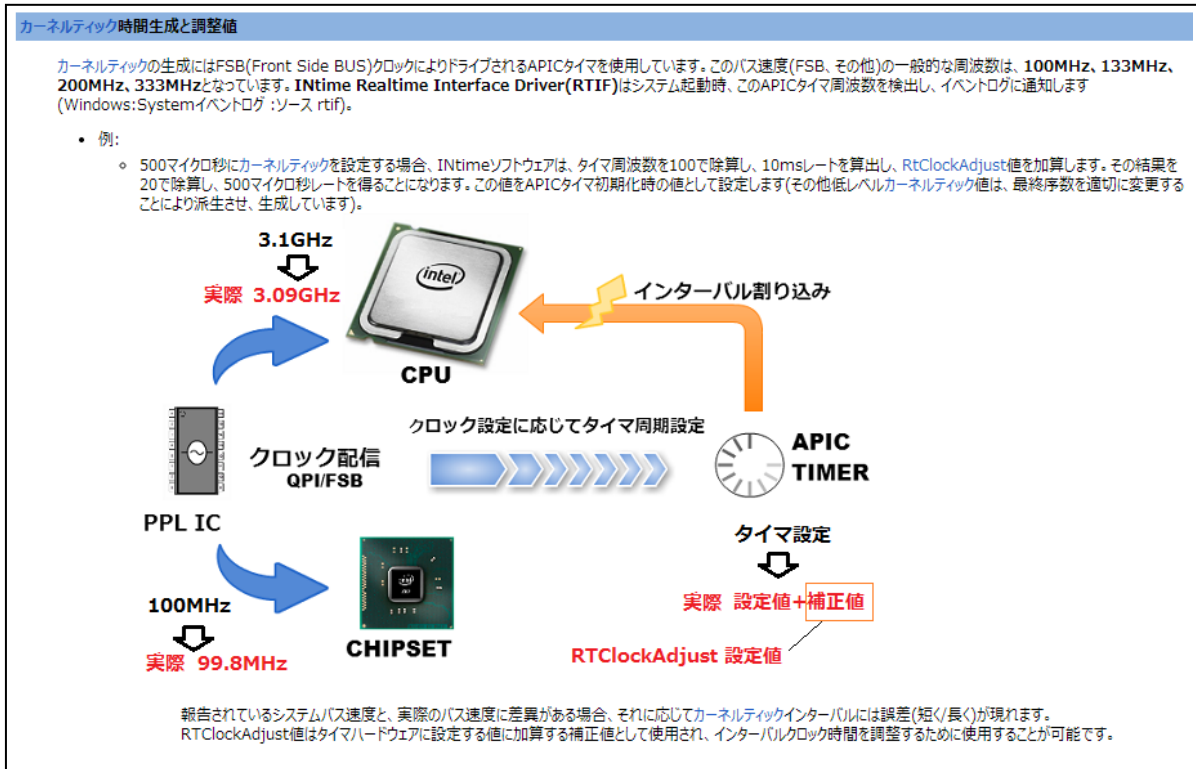


図 4-3-2. INtime カーネルティック時間生成と調整値 (マイクロネット社提供)

RtClockAdjust 設定値については、弊社で検証したところ、EtherCAT 通信周期を 1ms とした場合、複数メーカーのスレーブをどれだけ接続しても 0xFFFF702(-2300)という値を設定することで、0 応答することは無くなりました。

AI-Motion 開発環境ディスクの「レジストリ登録ファイル」に RtClockAdjust レジストリキーを 0xFFFF702 という設定値で登録するためのファイルを用意しています。下記の方法で登録してください。

- ① Windows32bit 版のときは「RtClockAdjust.reg」を Windows64bit 版のときは「RtClockAdjust64.reg」を USB メモリにコピーします。
- ② INtime がインストールされた端末上で「RtClockAdjust.reg」 or 「RtClockAdjust64.reg」をダブルクリックして実行します。
- ③ 実行すると図 4-3-3 のような画面が起動します。「はい」ボタンをクリックしてください。

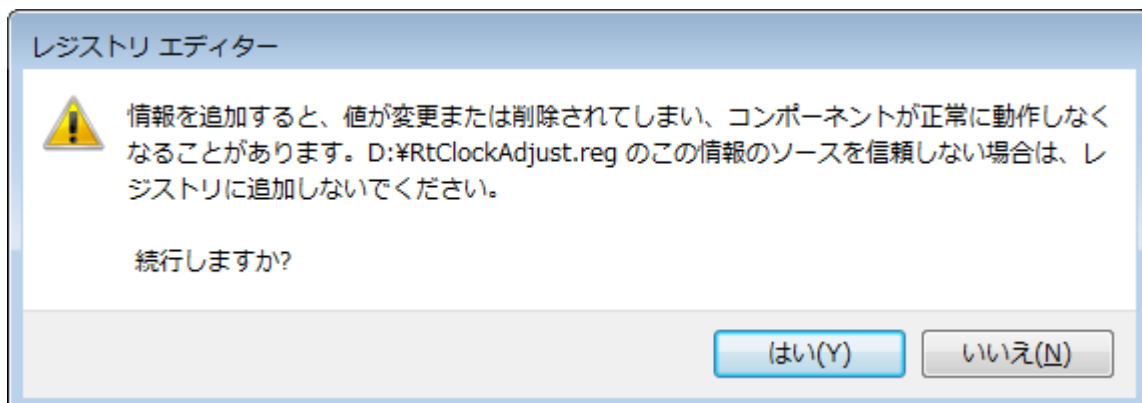


図 4-3-3. レジストリ登録警告メッセージ

- ④ 正常に登録が完了すると、図 4-3-4 のような画面が起動します。「OK」ボタンをクリックしてください。

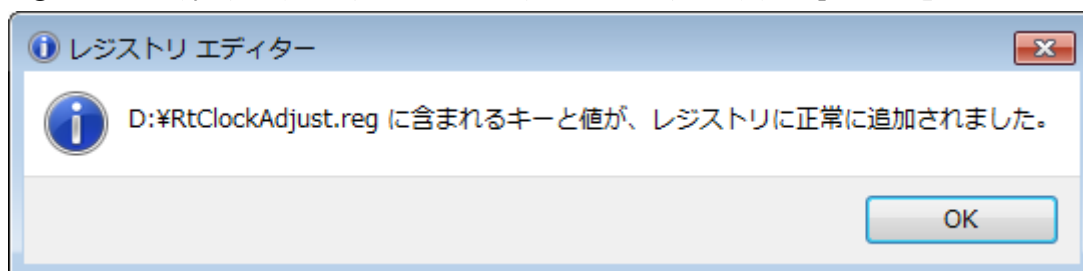


図 4-3-4. レジストリ登録完了メッセージ



## 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。

77IT20002A

2017年 6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>