

OMRON

**SYSTMAC**  
always in control

## Sysmac® オートメーションプラットフォーム による制御システム開発

オムロン株式会社  
インダストリアルオートメーションビジネスカンパニー  
オートメーションシステム統轄事業部  
コントロール事業部 開発センタ ソフトウェア開発部  
浪江 正樹

**SYSTMAC**  
always in control

# 内容

- 会社紹介
- IEC61131-3の概要
- Sysmac オートメーションプラットフォームの概要と  
特長
- Sysmacにおける制御プログラムの実行動作
- Simulink® とSysmacによるモデルベースデザイン  
(デモンストレーション含む)
- まとめ

OMRON

# 会社紹介

**SYSTMAC**  
always in control

## 会社概要

- 1933年、レントゲン写真撮影用タイマの製造を開始し、立石電機製作所を創業しました。以来、創業者が唱えた「機械にできることは機械にまかせ、人間はより創造的な分野での活動を楽しむべきである」という企業哲学の下、工場や社会インフラにおいて、革新的な数々の自動化を実現して来ました。世界初となる、無接点近接スイッチ、無人駅システム、オンライン現金自動支払機などはその一例です。

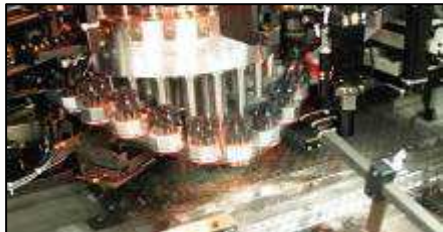
会社概要	
社名	オムロン株式会社 OMRON Corporation
本社所在地	京都市下京区塩小路通堀川東入
代表者	代表取締役社長 山田 義仁
創業	1933年5月10日
設立	1948年5月19日
資本金	641億円(2012年3月31日)
売上高(連結)	2011年度 6,195億円

# 事業別売上高構成比

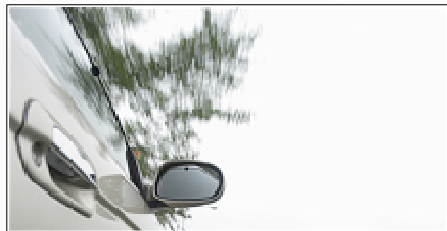
IAB(産業機器)



EMC(電子・メカニカル機器)



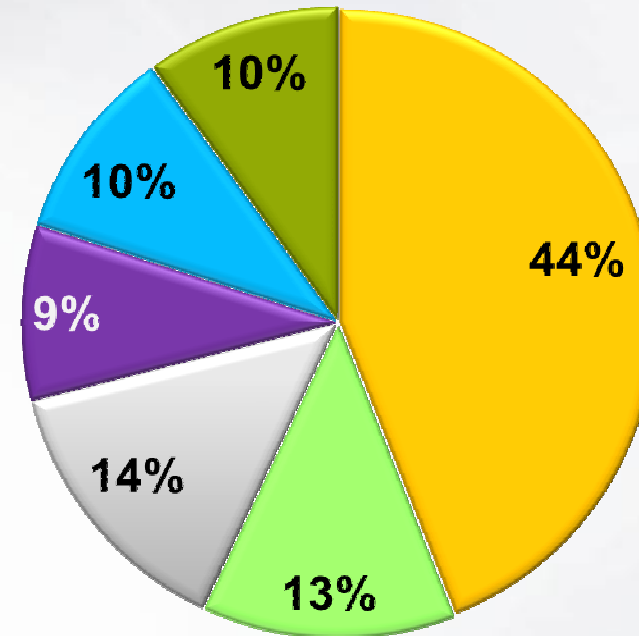
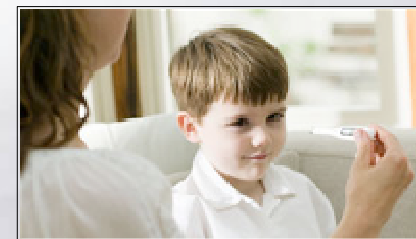
AEC(車載用電子機器)



SSB(社会システム)



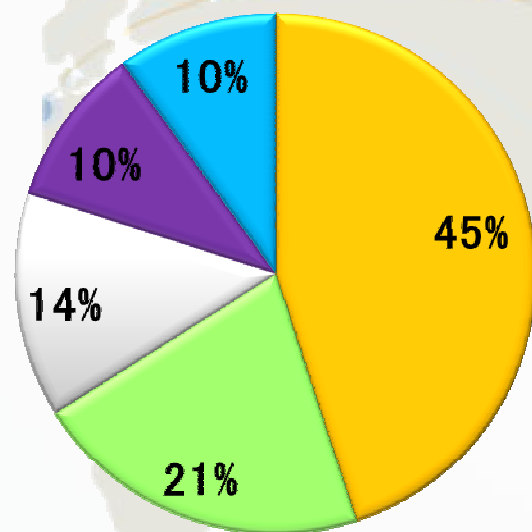
HCB(健康機器)



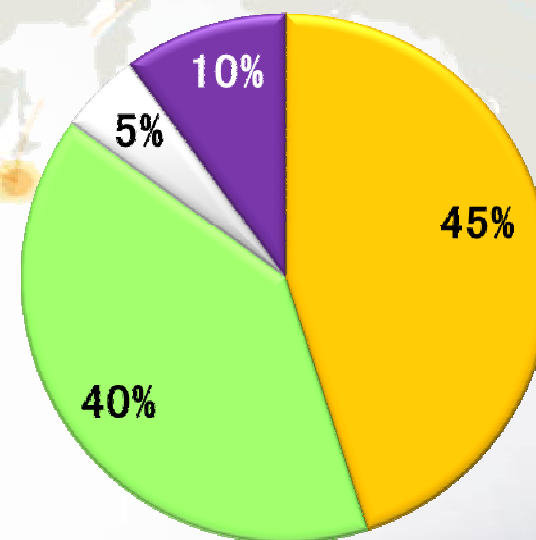
# IAB(産業事業)のエリア別・事業別売上

## エリア別

## 事業別



- 日本
- 欧州
- 中国
- 北米
- アジア



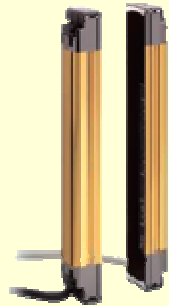
- 産業機器
- オートメーションシステム
- 検査システム
- その他

# オートメーションシステム事業の主な商品

## 進化するお客様の生産課題解決をサポートする商品群



セーフティ  
コントローラ



セーフティ  
ライトカーテン



非常停止スイッチ

### 見せる・操る



HMI

### 計る・認識する



視覚センサ



視覚センサ



### 計算する・順序だてる・指示する



マシンコオートメーション  
コントローラ



PLC

### 動かす・止める



サーボモータ

インバータ

# オートメーションセンタ(ATC)

オープンネットワーク採用による、豊富な制御機器の接続性の確保と積極的な拡充・推進  
 オムロン機器間、オムロンと第三者機器とのつなぎ検証

機械のオートメーションをお客様と一緒に検討する場

The diagram features a world map with four callout bubbles pointing to specific regions: ATC 欧州 (Europe), ATC 中国 (China), ATC 日本 (Japan), and ATC アメリカ (America). Below the map, the EtherNet/IP logo is shown with a blue and orange globe icon, and the EtherCAT logo is shown with a red arrow pointing right. At the bottom left, the SYSTMAC logo is displayed with the tagline "always in control".





# グローバル・ネットワーク／サービス・サポート

40カ国・150拠点に及ぶグローバルネットワーク



# IEC61131-3の概要

# PLC関連の技術動向の概要

## ➤ PCアーキテクチャ

ASIC + RISCマイコンの構成に代えて、Intel MPU を採用する流れが広がりつつある。また、オペレーティングシステムには各種 Real-time OS が採用されている。MPUの高速性能とその進化スピードの恩恵を享受できることが最大のメリットである。

## ➤ Ethernetベースのフィールドネットワーク

情報系だけでなく、フィールド系のネットワークも Ethernet ベースに移行している。例えば、次のネットワークがIEC規格化されている。

EtherCAT, EtherNet/IP, PROFINET, Modbus TCP/IP, ETHERNET Powerlink, CC-Link IE, etc.

## ➤ IEC 61131-3 (PLCプログラミング言語)

ソフトウェア開発生産性の向上(資産の流用性向上、養育コストの低減など)への期待を背景に、日本においてもユーザの関心が高まっている。PLCopen が普及を推進している。

## ➤ セーフティ(機能安全)

IEC 61508(電気・電子・プログラマブル電子安全関連系の機能安全)に基づいた、安全PLC規格 IEC 61131-6 の制定作業が最終段階まで進んでいる。

## ➤ セキュリティ

2010年に発生したStuxnetにより、制御システムにおけるセキュリティ問題が表面化した。

日本では2012年3月に、技術研究組合制御システムセキュリティセンタ(CSSC)が設立された。今後、PLCにおいてもセキュリティ強化が期待される。

# IEC61131-3の普及状況

- 地域により特徴があるが、世界的に認知・使用され始めている。また、各国で規格のローカライズの動向あり。



# IEC61131-3 共通要素(主なもの)

## ➤ データ型

- ✓ 基本データ型 : BOOL, SINT, DINT, REAL, TIME, DATE, BYTE, WORD, など
- ✓ 派生データ型 : 配列、構造体、列挙、など

## ➤ 変数

- ✓ VAR, VAR\_INPUT, VAR\_OUTPUT, VAR\_IN\_OUT, VAR\_GLOBAL, など

## ➤ プログラム構成ユニット(POU)

- ✓ ファンクション、ファンクションブロック、プログラム

## ➤ SFC要素

- ✓ ステップ、トランジション、アクション

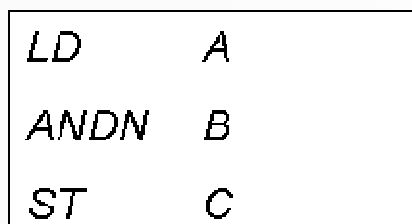
## ➤ コンフィギュレーション要素

- ✓ タスク、グローバル変数、など

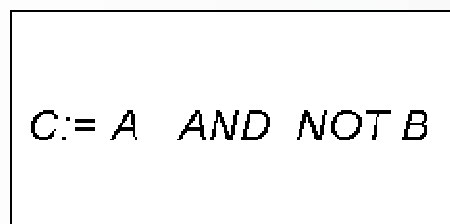
# IEC61131-3 言語の種類

- 4言語+1要素
- スキルや経験にあった言語を選択可能

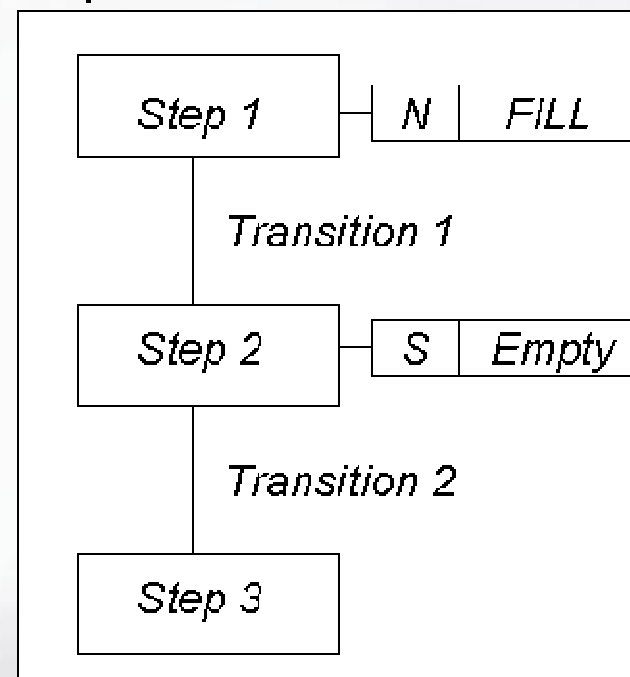
Instruction List



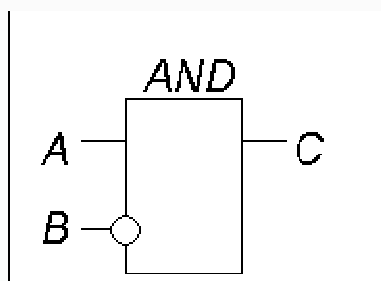
Structured Text



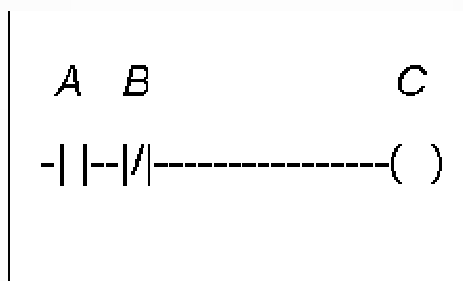
Sequential Function Chart



Function Block Diagram



Ladder Diagram



図の出典:「PLCにおける国際規格の普及と技術動向」(PLCopen Japan)

# IEC61131-3 各言語の適性

- 用途に応じた使い分けが可能
- LD言語とST言語の2つができれば、あらゆる用途に対応可能

表. プログラミング言語の得手・不得手

主な使用状況	LD言語	IL言語	ST言語	FBD言語	SFC言語
単純なリレーシーケンス処理	◎	×	△	△	×
数式演算処理	△	×	◎	○	×
状態せん移に基づく順序制御 (ステップシーケンス処理)	△	×	○	×	◎
連続的なアナログ信号処理	△	×	○	◎	×
複雑な情報処理	△	×	◎	△	×
プログラムメモリ制約の厳しい場合	○	◎	△	△	×
最も高速に性能を求められる場合	○	◎	○	○	×
運転方案と対応がとりやすい表現	×	×	○	○	◎
動作を視覚的に確認したい場合	○	×	×	◎	○
注記 記号の意味は、次による。 ◎：最も適している，○：適している，△：困難な場合もある，×：適さない					

# Sysmac

## オートメーションプラットフォーム の概要と特長



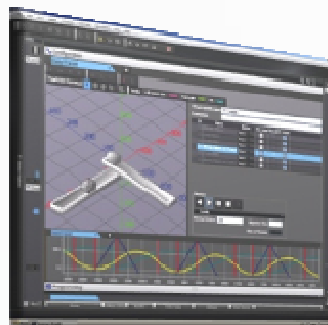
# Sysmac - 統合型オートメーションプラットフォーム

## ONE Controller



シーケンス制御 / モーション制御 / 視覚センサ

## ONE Software



機械とのウィンドウ

## ONE Connection



# Sysmac が提供を目指す価値

## ➤ 高速・高精度な制御性能

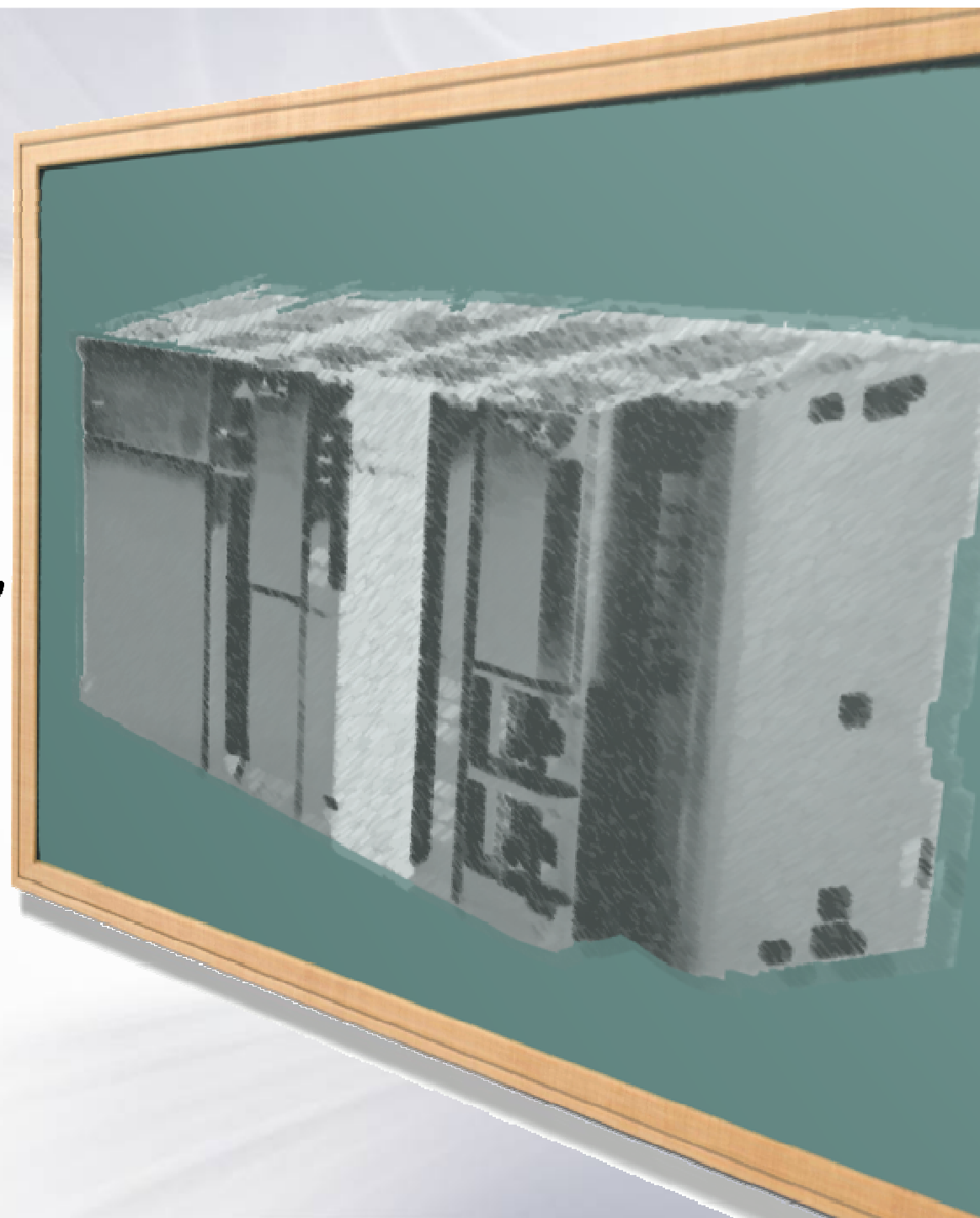
- ✓ PCアーキテクチャの採用による高速制御性能
- ✓ ロジック制御とモーション制御の統合化による制御設計の容易性
- ✓ 産業用イーサネットEtherCAT®の採用による、コントローラ・I/Oデバイス間の高速通信と高精度同期性能

## ➤ ソフトウェア開発生産性の向上

- ✓ PLC(Programmable Logic Controller)用プログラミング言語の国際規格IEC 61131-3採用による、高いソフトウェア開発生産性

# NJシリーズ

マシンオートメーションコントローラ



# NJシリーズ

業界最速

最新のインターフェース技術を搭載

Ethernetベースの産業用ネットワーク  
搭載

最先端高速MPU採用による超高速  
演算性能

モーション機能とPLC機能を融合

PLCクラスのRAS機能を搭載



OMRON



EtherCAT



IEC 61131-3 規格準拠



インテル® Atom™ プロセッサ  
(1.6 GHz)

OMRON

100% OMRON  
シーケンス & モーションを  
融合した制御ランタイムエンジン



PLCopen モーションファンクション  
ブロック



SDカードスロット



USBポート

SYSTMAC  
always in control

\*インテル、Intel ロゴ、インテル Atomは、米国およびその他の国におけるインテル コーポレーションの商標です。

# 圧倒的な高速化で高速・高精度な制御を実現

インテル® Atom™ プロセッサ採用による超高速な演算性能

Machine  
Automation  
Controller

命令実行 従来比 8~230倍の高速化

**業界最速**

- LD=1.9ns~
- 倍精度浮動小数点演算=26.0ns~ (加算命令)

モーション機能の統合と変数対応の命令実行を含め、  
従来PLCの当社比 4倍の高速化を実現

## 演算時間 <sup>\*1</sup>

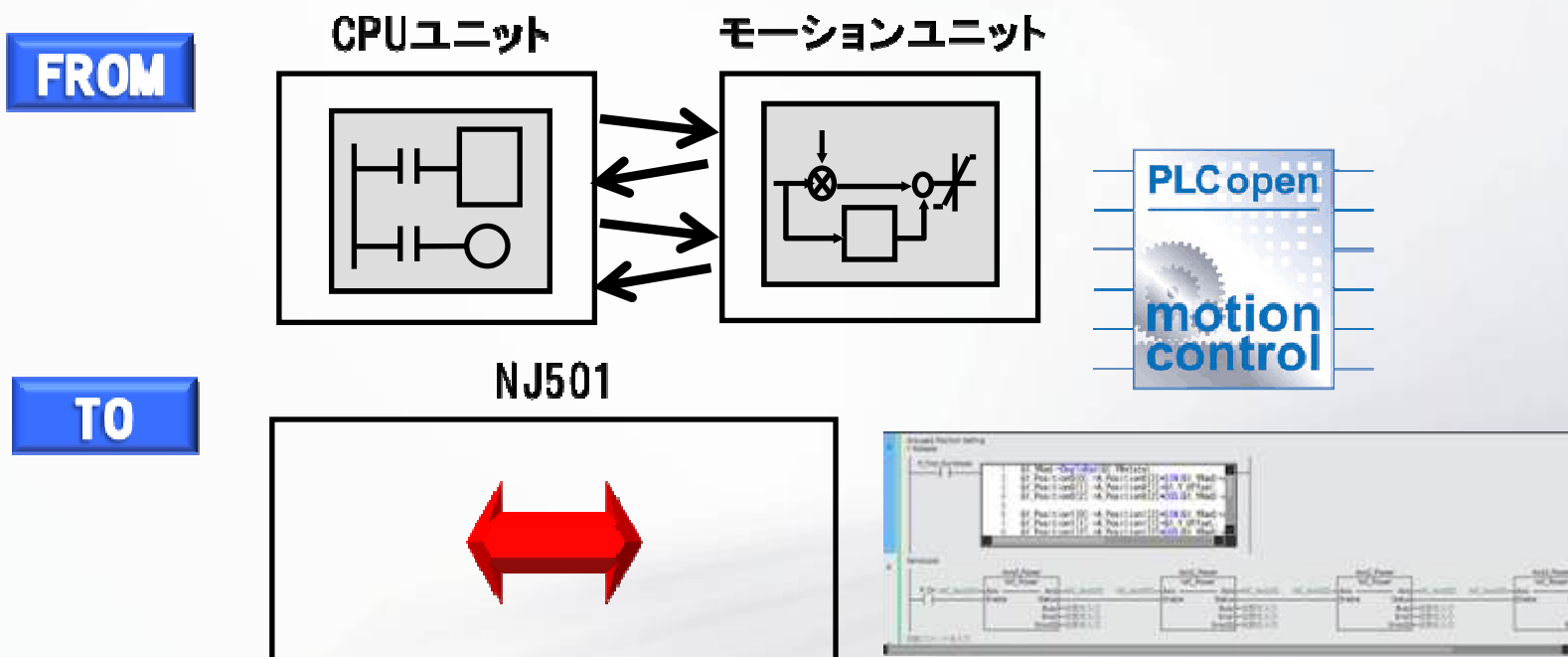


\*1  
プログラム ロジック部 50kステップ (基本、応用命令を含む)  
モーション部 60kステップ (倍精度浮動小数点演算などを含む)  
軸 32軸 (PTP 13軸, 同期制御 10軸, 補間制御 9軸)  
I/O VO512点, AD/DA32点, CT8点

# モーション機能とPLC機能を融合

同一タスク上でPLCエンジンとモーションエンジンを完全に同期演算実行ができます

動きながら加工する同期動作や複数品種の段取り変更など、お客様の高速な新商品投入、変種変量生産でのタクトアップ、精度向上を実現



# 信頼性とロバスト性の実現

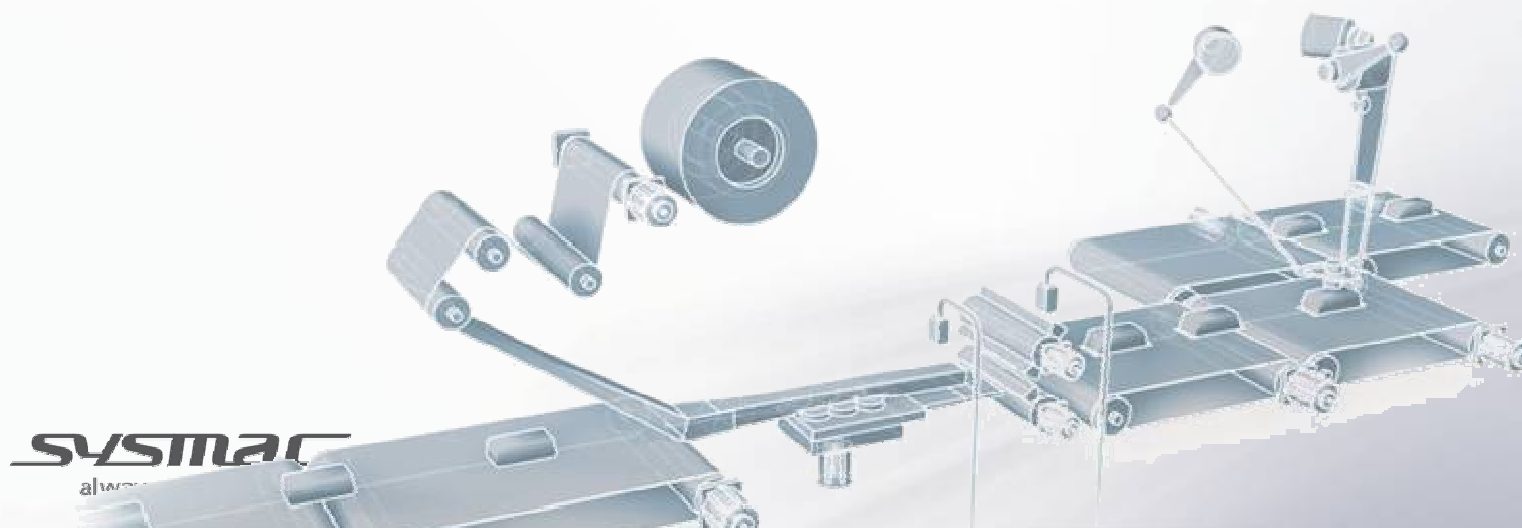
信頼性

産業用コントローラに求められる状態監視機能を搭載

- パソコンや最新の情報機器に使われる最新MPUや通信技術上で、メモリの状態管理やウォッチドッグタイマを実現


高速な電源立ち上がり時間

高いノイズ耐性など生産現場でのロバスト性を確保



# EtherCAT

マシン制御用ネットワーク

A 3D-rendered logo for EtherCAT is displayed on a green chalkboard with a wooden frame. The logo consists of the word "EtherCAT" in a white, sans-serif font, followed by a stylized white arrow pointing to the right. The arrow has a thick, blocky body and a sharp, triangular head. The entire logo is rendered with a slight shadow, giving it a three-dimensional appearance as if it were a physical object on the chalkboard. The background of the chalkboard is a solid green color, and the wooden frame is a light brown color. The overall scene is set against a light blue and white background with a subtle grid pattern.

EtherCAT®



# EtherCAT

NJ501  
EtherCAT

32軸/0.5msの高速制御を実現\*

最大192ノード、内64軸のモーション制御を実現

視覚センサを統合



サーボドライバ

インバータ

I/O

視覚センサ

**SYSTMAC**  
always in control

\*プログラム、サーボロック、正転・逆転を繰り返すプログラムのみを実行、シングルタスクで動作  
軸 32 軸 全軸動作、I/O 64

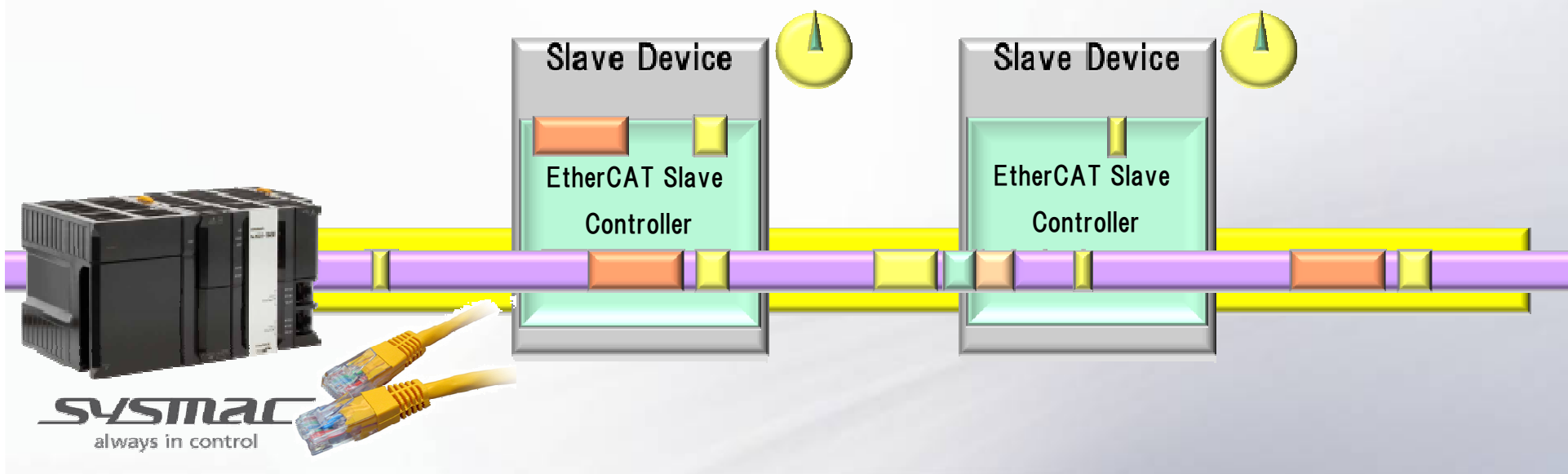
# EtherCAT

NJ501  
EtherCAT

NJ501の演算周期は、EtherCATの周期に完全に同期

スレーブ間同期1  $\mu$ sを実現  
ディストリビューテッドクロック機能

分岐ユニットにより、ツリー配線やスター配線が可能



OMRON

# Sysmac Studio

オートメーションソフトウェア



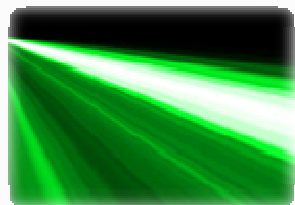
# Sysmac Studio の特長



簡単な修得

IEC 61131-3に準拠

ユーザインタフェースを統一



高速開発

ラダー、ST、ファンクションブロックのサポート

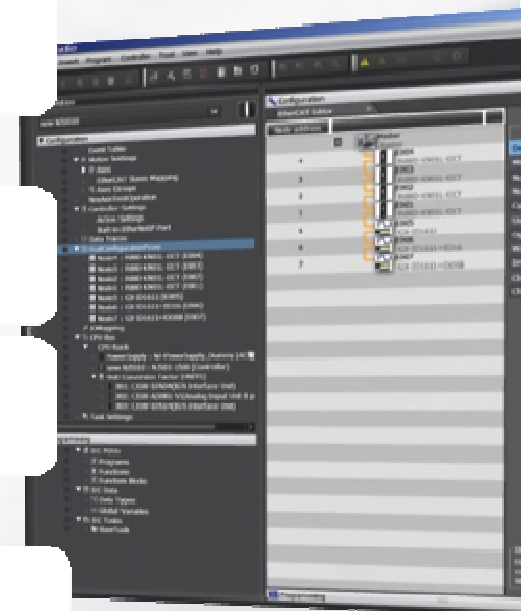
カムエディタの搭載



統合テスト環境

3D対応のモーション軌跡シミュレータ

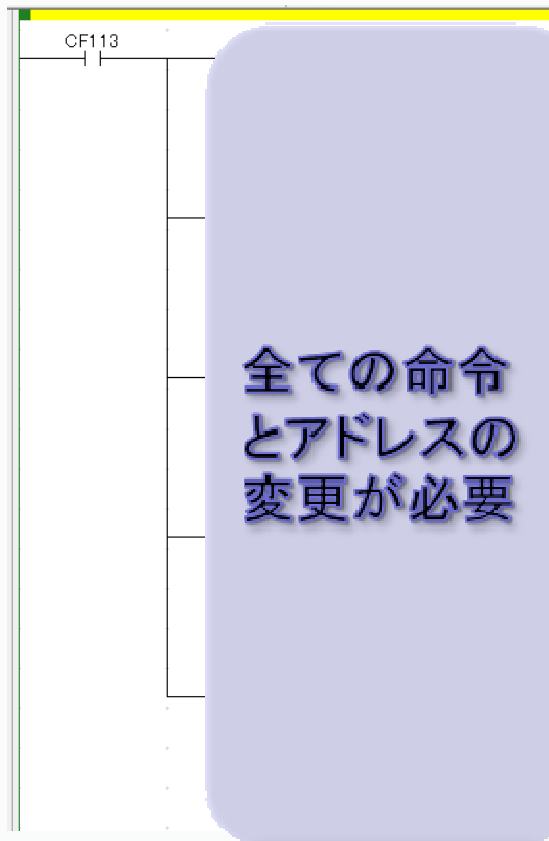
モーション統合シミュレーション、デバッグ機能



# 変数によるハードウェアとソフトウェアの完全分離

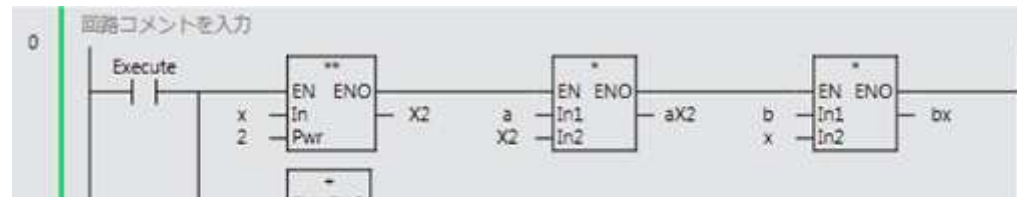
- プログラムの独立性、再利用性を確保し、ユーザプログラムの資産価値が向上します

事例) 16Bitから32bitに  
データサイズを変更



全ての命令  
とアドレスの  
変更が必要

## NJ501の場合



データ型の変更だけで、プログラムの  
変更はありません

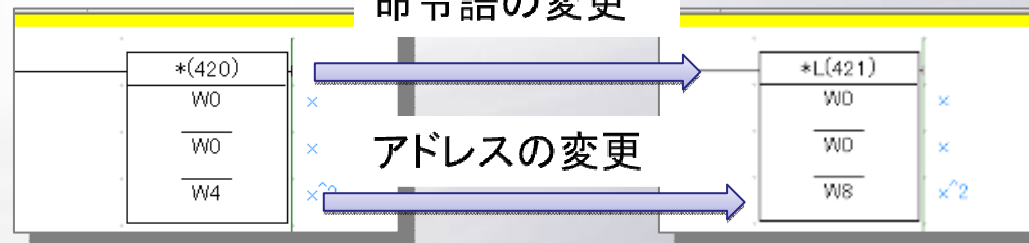
内部変数	名称	データ型
外部変数	Execute	BOOL
	a	INT
	b	INT
	c	INT
	x	INT
	y	INT

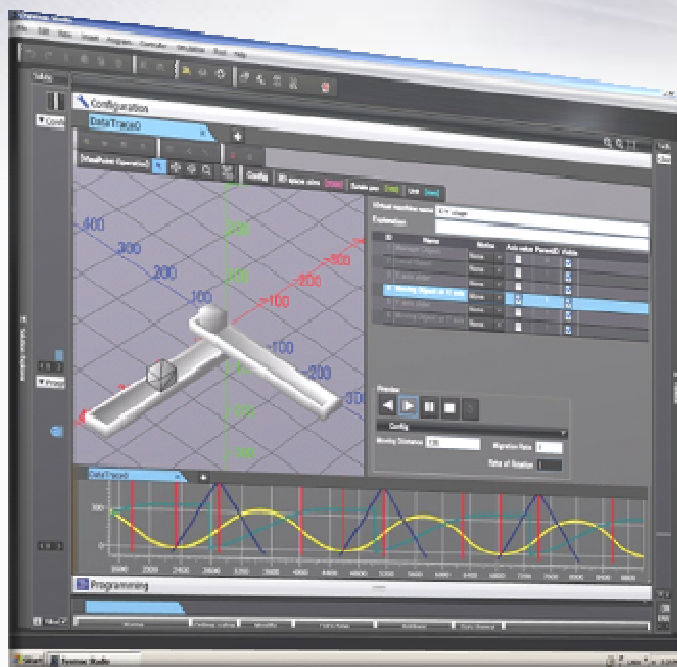
DINT
DINT
DINT
DINT
DINT

命令語の変更

アドレスの変更



# シミュレーション・デバッグ機能



- シミュレーション
  - ✓ ステップ実行、ブレークポイント
  - ✓ 3Dモーショントレース、2D軌跡表示
  - ✓ タスク実行時間予測
  
- データトレース
  - ✓ トリガトレース、連続トレース
  
- モニタリング機能
  - ✓ ラダー/STエディタ
  - ✓ ウォッチウィンドウ、I/Oマップ

```

IF (CurrentDegree >= INT#10#360) THEN
    CurrentDegree := 0
END_IF;
ResultSIN := SIN (DegToRad (CurrentDegree));
ResultCOS := COS (DegToRad (CurrentDegree));
ResultTAN := TAN (DegToRad (CurrentDegree));
    
```

CurrentDegree = 90 \*

ResultSIN = 1 \*



# Sysmac Studio 対応商品

サーボ



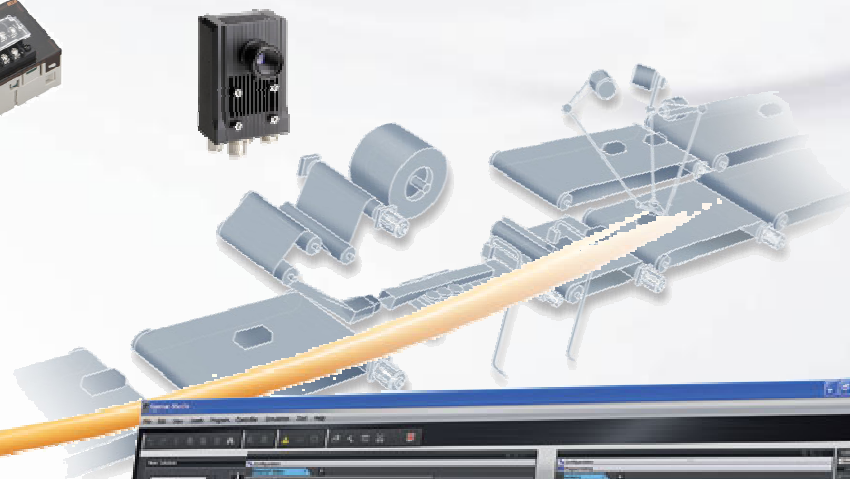
インバータ



I/O



視覚センサ

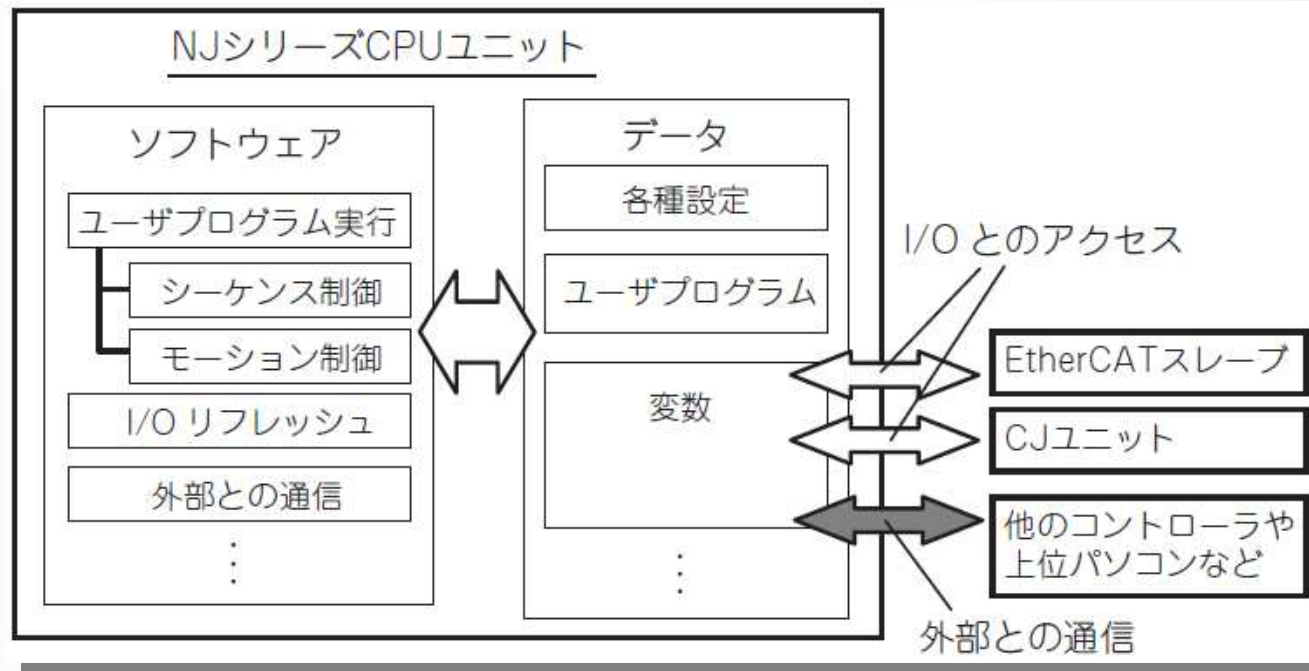


# Sysmacにおける 制御プログラムの実行動作



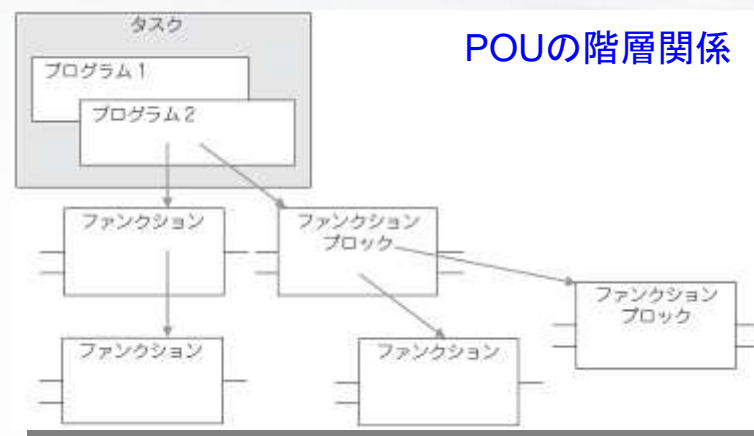
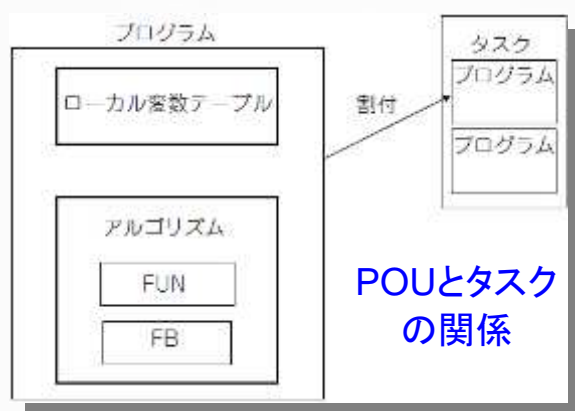
# CPUユニットの動作

- CPU ユニットは、各種設定、ユーザプログラム、変数などのデータを持ち、これらのデータを使って処理を実行する。
- CPU ユニットとI/O のアクセス、および外部との通信には、変数を使用する。
- CPU ユニットは、ソフトウェアおよび、変数を使ったI/O へのアクセスにより、シーケンス制御とモーション制御の両方を実行する。



# POUとタスクの関係

- プログラムは、ローカル変数テーブルとアルゴリズム部で構成される。
- アルゴリズム部には、ファンクションかファンクションブロックを記述できる。



項目		POU 種類		
		プログラム	ファンクションブロック (FB)	ファンクション (FUN)
起動方法		割り付けられたタスクの起動により起動	プログラム、別のFBからの呼び出し	プログラム、FB、FUNからの呼び出し
アルゴリズム内	すべての命令	可	可	不可
	ユーザ定義 FUN	可	可	不可
	ユーザ定義 FB	可	可	不可
実行条件		常時実行	常時実行 実行条件は入力変数によって指定	EN 入力で指定

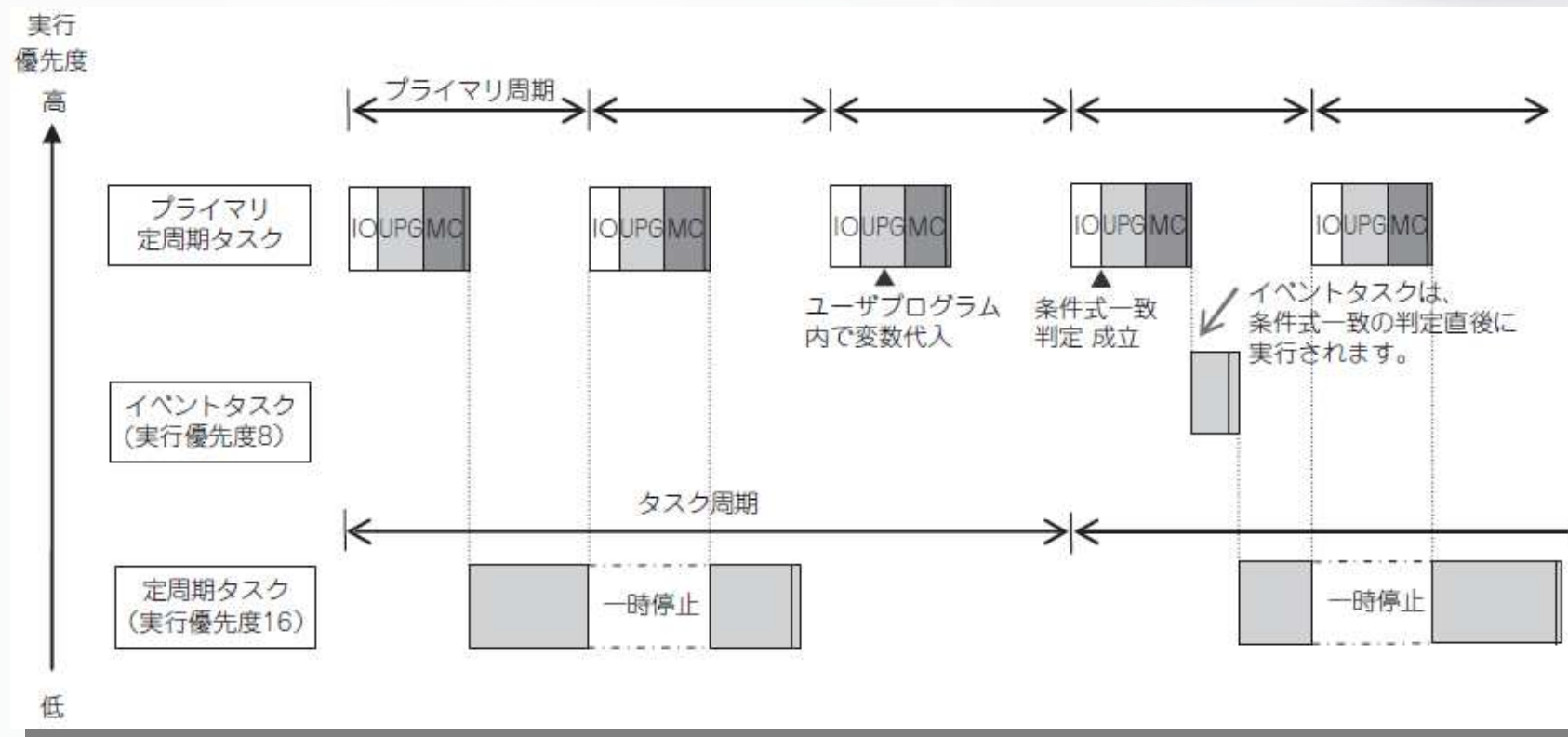
# タスクの種類

- 3種類のタスクがあり、プライマリ定周期タスクを最優先で実行する。
- 定周期タスクとイベントタスクの優先関係は指定できる。
- 1タスクあたり、最大128のプログラムを割り当てることができる。

タスク種類	タスク数	実行優先度	主な処理内容	設定可能周期
プライマリ定周期タスク	1	4	EtherCAT通信 モーション制御 I/Oリフレッシュ ユーザプログラム	0.5ms, 1ms, 2ms, 4ms
定周期タスク	0 ~ 3	16 17 18	I/Oリフレッシュ (優先度16のみ) ユーザプログラム	プライマリ定周期タスクの整数倍で、かつ1ms ~ 100msの範囲内の予め決められている選択肢から指定する。
イベントタスク	0 ~ 32	8 48	ユーザプログラム	-

# タスクの動作

- ▶ プライマリ定周期タスクと定周期タスクは、指定周期で繰り返し実行する。
- ▶ イベントタスクは、予め指定した条件が成立したときに1回だけ実行する。
- ▶ システムサービスは、すべてのタスクの空き時間に実行する。





# Simulink® と Sysmac による モデルベースデザイン (デモンストレーション含む)

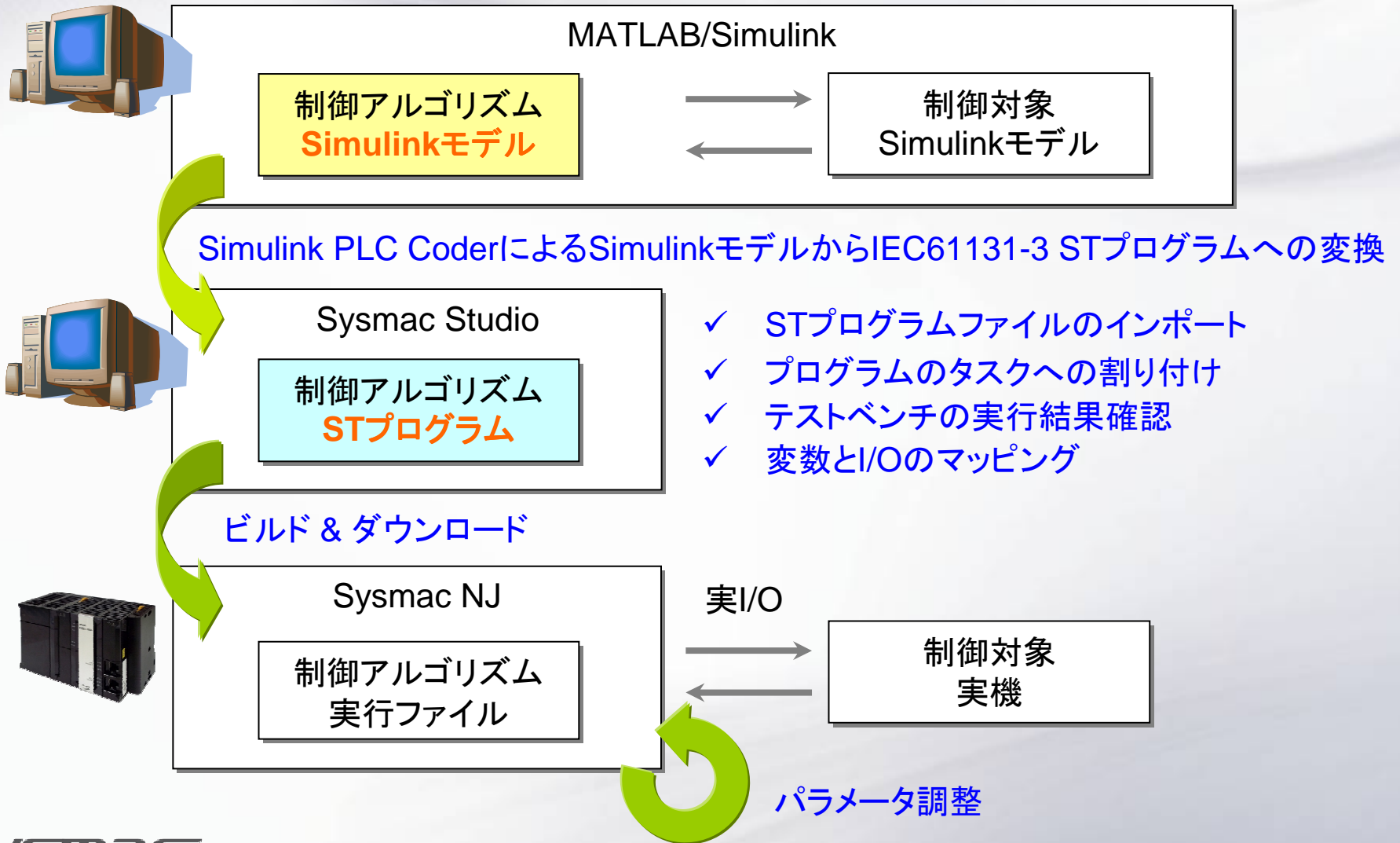
# Simulink® PLC Coder™ の概要

- IEC61131-3のStructured Text コードを自動生成
- Simulinkをサポート
  - ✓ 再利用可能なサブシステム、PIDコントローラ、ルックアップテーブルを含む
- Stateflowをサポート
  - ✓ 状態遷移図、真理値表を含む
- Embedded MATLAB関数をサポート
  - ✓ IF文、ループ構造、算術演算を含む
- 各社PLCのIEDをサポート
- テストベンチ生成

詳しくは、MathWorks社Webサイトなどを参照ください。

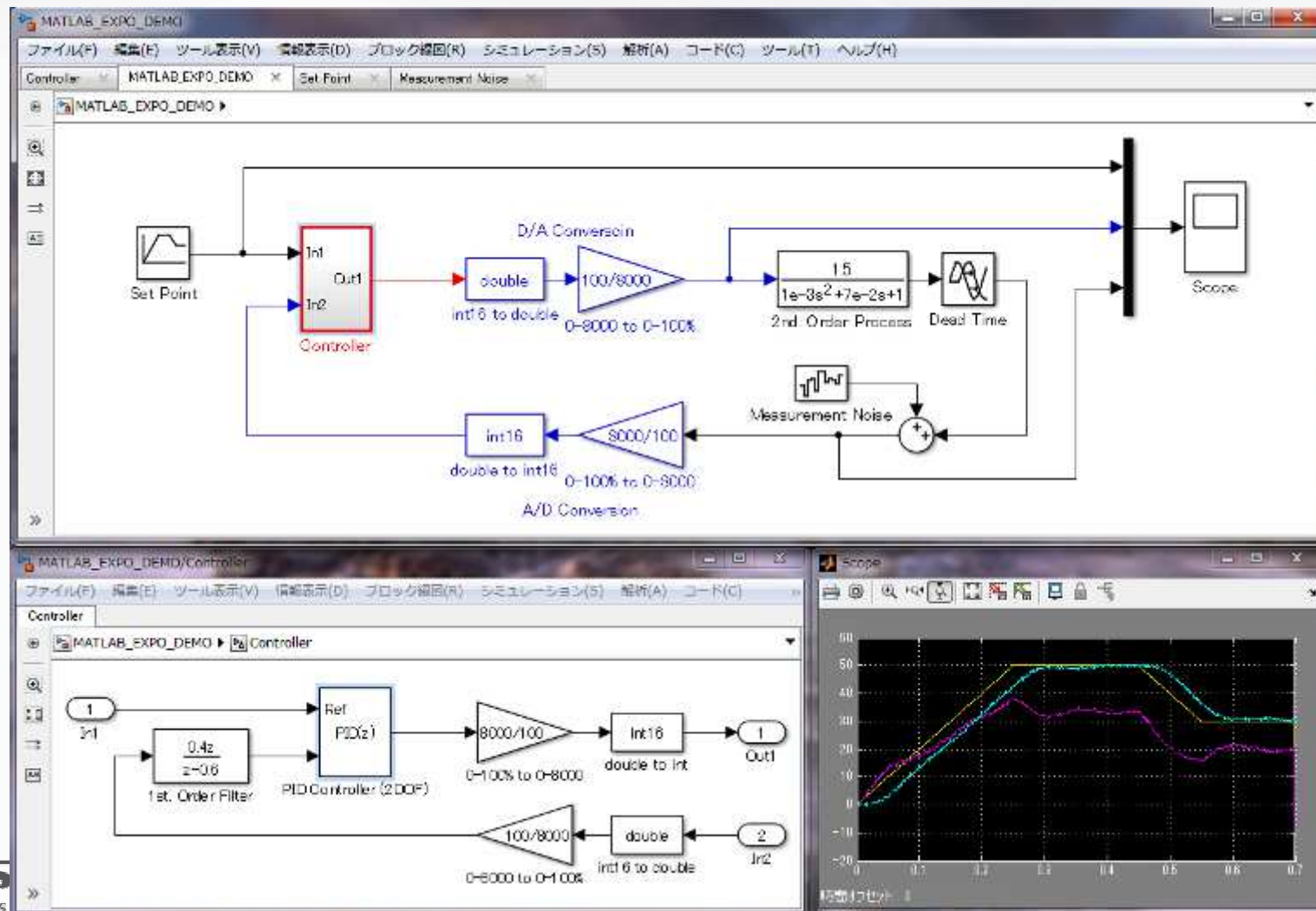
<http://www.mathworks.co.jp/products/sl-plc-coder/>

# Simulink と Sysmac によるモデルベースデザイン



# 操作デモ: Simulinkモデルの例

- コード生成対象は、サブシステム' Controller'





# 操作デモ：生成コードの例（一部抜粋）

## ➤ ヘッダ部分

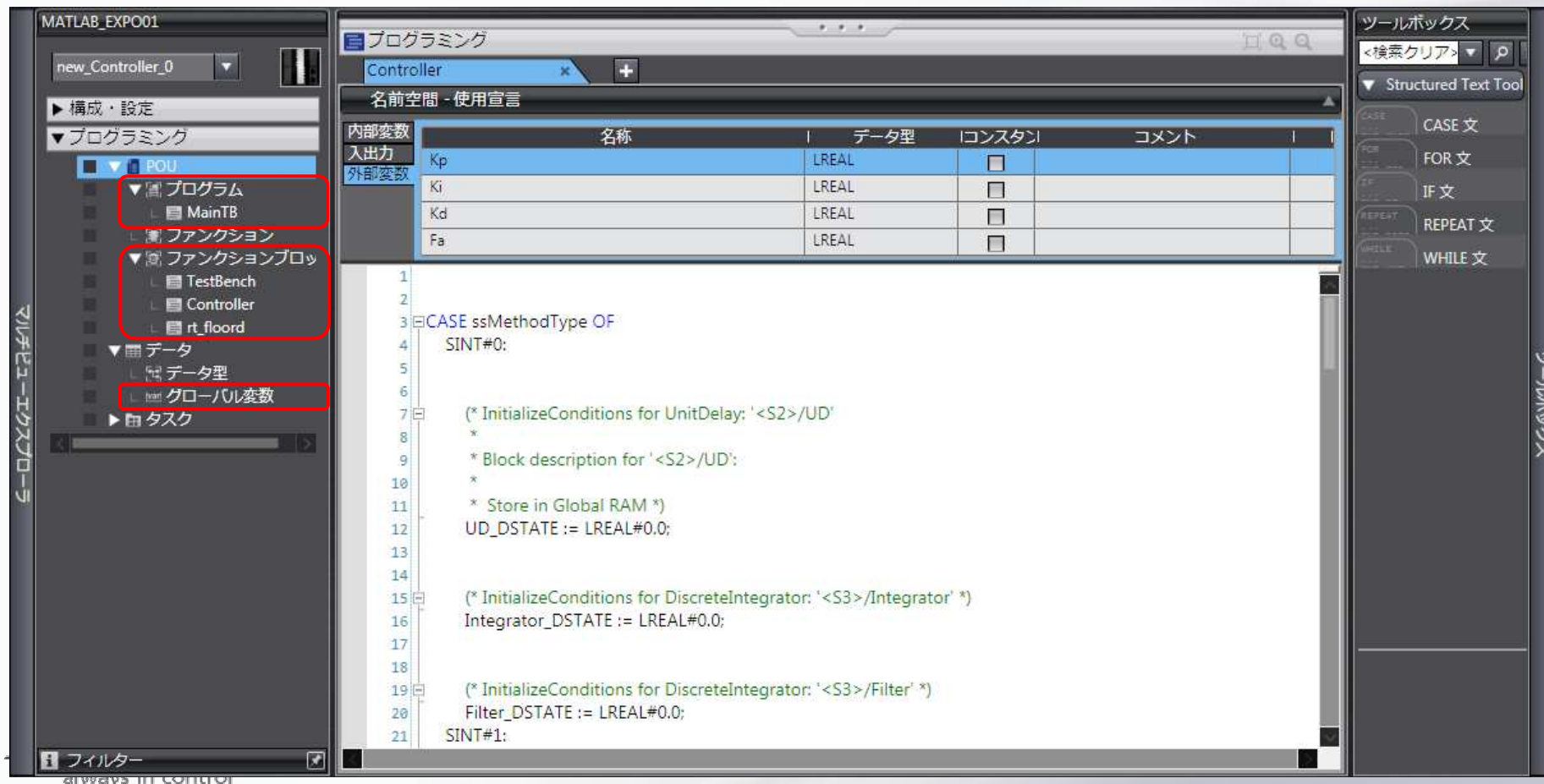
```
* File: MATLAB_EXPO_DEMO.xml
*
* IEC 61131-3 Structured Text (ST) code generated for subsystem "MATLAB_EXPO_DEMO/Controller"
*
* Model name                : MATLAB_EXPO_DEMO
* Model version              : 1.48
* Model creator              : NAMIE
* Model last modified by    : NAMIE
* Model last modified on    : Thu Oct 25 12:40:02 2012
* Model sample time         : 0.001s
* Subsystem name            : MATLAB_EXPO_DEMO/Controller
* Subsystem sample time     : 0.001s
* Simulink PLC Coder version : 1.4 (R2012b) 29-Jun-2012
* ST code generated on      : Thu Oct 25 12:40:50 2012
*
* Target IDE selection      : OMRON Sysmac Studio
* Test Bench included       : Yes
```

## ➤ 出力変数計算部分

```
(* Output: '<Root>/Out1' incorporates:
* DataTypeConversion: '<S1>/double to int'
* DiscreteIntegrator: '<S3>/Integrator'
* Gain: '<S1>/0-100% to 0-8000'
* Gain: '<S3>/Proportional Gain'
* Gain: '<S3>/Setpoint Weighting (Proportional)'
* Inport: '<Root>/In1'
* Sum: '<S3>/Sum'
* Sum: '<S3>/Sum1' *)
i0_rt_floord(u := (((((Rb * In1) - rtb_Integrator) * Kp) + Integrator_DSTATE) + rtb_FilterCoefficient) * LREAL#80.0);
temp1 := i0_rt_floord.y;
Out1 := DINT_TO_INT(TRUNC(LREAL_TO_REAL(temp1)));
```

# 操作デモ: Sysmac Studioへのインポート後画面の例

- STプログラムファイルより、以下をインポートした後の画面を示す。
  - ✓ プログラム: MainTB、      ファンクションブロック: Testbench, Controller, rt\_floor
  - ✓ グローバル変数: Kp, Ki, Kd, Fa



# 操作デモ: テストベンチ機能による検証

- テストベンチプログラム内の次の2つの変数値を確認すればOK
  - ✓ testVerify が 'True' である。
  - ✓ testCycleNum が入出力データ数に一致している。

The screenshot displays the GX Developer software interface for a test bench program. The main window shows a ladder logic program with the following code:

```

1  (* TEST_CYCLE_COUNT = 701 *)
2
3
4  IF need_init [False] THEN
5    testVerify [True] := BOOL#TRUE;
6    testCycleNum [701] := DINT#0;
7    need_init [False] := BOOL#FALSE;
8  END_IF;
9
10
11 -IF testVerify [True] THEN
12
13  IF testCycleNum [701] < DINT#701 THEN
14    (* TEST_CYCLE_SETUP *)
15    cycle_In1 [0] := tb_In1[testCycleNum];
16    cycle_In2 [2441] := tb_In2[testCycleNum];
17    cycle_Out1 [-814] := tb_Out1[testCycleNum];
18
19  IF testCycleNum [701] = DINT#0 THEN

```

The variable declaration table at the top of the main window lists the following variables:

内部変数	名称	データ型	初期値	割付先	保持	コネクション	コメント
入出力	tb_In1	ARRAY[0..700] OF...	[LREAL#0.0,LR...		<input type="checkbox"/>	<input type="checkbox"/>	
外部変数	cycle_In1	LREAL			<input type="checkbox"/>	<input type="checkbox"/>	
	tb_In2	ARRAY[0..700] OF...	[INT#9,INT#6...		<input type="checkbox"/>	<input type="checkbox"/>	
	cycle_In2	INT			<input type="checkbox"/>	<input type="checkbox"/>	
	tb_Out1	ARRAY[0..700] OF...	[INT#-8,INT#...		<input type="checkbox"/>	<input type="checkbox"/>	

The interface also shows a project tree on the left with 'TestBench' selected, and a toolbar on the right with various simulation and control options.

# 操作デモ: 変数とI/Oのマッピング

➤ I/Oマップにおいて、入出力変数と実I/Oをマッピングする。

The screenshot displays the 'I/Oマップ' (I/O Map) configuration window. The table below represents the data shown in the interface:

位置	ポート	説明	R/W	データ型	変数	変数コメント	変数種別
▼ CPU・増設ラック							
CF	CPUラック 0						
▼ EtherCATネットワークコンフィ							
Eth	Master						
Nc	▼ GX-AD0471						
	CH1 Input 16bit	入力1 変換データ	R	INT	PV100		Control_Program 内部変数
	CH2 Input 16bit	入力2 変換データ	R	INT			
	CH3 Input 16bit	入力3 変換データ	R	INT			
	CH4 Input 16bit	入力4 変換データ	R	INT			
	▼ Sysmac Error Status	Sysmac 異常ステータス	R	BYTE			
	Observation	監視情報レベルの異常情	R	BOOL			
	Minor Fault	軽度フォールトレベルの	R	BOOL			
Nc	▼ GX-DA0271						
	CH1 Output 16bit	出力1 設定値	W	INT	MV100		Control_Program 内部変数
	CH2 Output 16bit	出力2 設定値	W	INT			
	▼ Sysmac Error Status	Sysmac 異常ステータス	R	BYTE			
	Observation	監視情報レベルの異常情	R	BOOL			
	Minor Fault	軽度フォールトレベルの	R	BOOL			

# PLCターゲットにおいて C言語を使用するモデルベースデザインへの優位点

- PLC Coder は各社PLCのIDEに対して専用コードを生成するため、コードの手直し不要でIDEにインポートできる。  
⇒ 品質確保と工数削減の効果が大きい。
- IEC61131-1 ST言語はタスクやPOUといった、制御システムに適したプログラム実行のベースが用意されている。  
⇒ C言語に比べて自由度は低いが、反面、保守性が高い。
- IEC61131-3の他の言語との親和性が高い。  
例:ラダープログラム中に、SimulinkからインポートしたFBを配置可能。  
⇒ PLCユーザにとっては理解しやすく、導入障壁が低い。
- PLC Coderはテストベンチ機能を持つ。  
⇒ PLC上での計算精度の検証が容易にできる。

## まとめ

- Sysmac オートメーションプラットフォームの特長
  - ✓ PCアーキテクチャとEtherCATの採用による高速・高精度の制御性能
  - ✓ IEC61131-3をベースとした統合開発環境による高い開発生産性
  - ✓ 産業コントローラとしてPLCの良さ(対PC)を継承
    - 高い信頼性(耐環境性能、長寿命)
    - 高い保守性(オンラインエディット、長期供給)
    - 制御用I/Oの豊富な品揃え、コンパクト
  
- Simulink PLC Coder との連携による新たな価値
  - ✓ モデルベースデザインによる高い開発生産性

お客様のさらなるご要望にお応えするべく、  
Sysmacを進化させてまいります

**SYSTMAC**  
always in control