



SMART

プログラマブルコントローラー

Modicon M241 / M251シリーズ

簡単プログラミング

はじめての **EcoStruxure Machine Expert**  
テキスト



MEMO

## はじめに

このたびは、自己学習テキスト[はじめてのEcoStruxure Machine Expertテキスト](以下本テキストと言います)をご購読いただきまして、誠にありがとうございます。

本テキストは、シュナイダーエレクトリック社製プログラマブルコントローラー Modicon M241/M251シリーズ（以下[M241][M251]と言います）の基本的なラダープログラムの作成およびPro-face製表示器との連携方法を体感いただくためのテキストです。

弊社カスタマーケアセンターへのお問合せや実際に現場で使われている例をもとに、最も一般的かつご要望の高いプログラムおよび作画方法をユーザー様自身で効率良く、体感いただけるよう構成されています。

なお、実際のご使用にあたっては、弊社製品マニュアルをよくお読みいただき、正しい取り扱いと機能を、十分にご理解いただきますようお願い申し上げます。

### ■ 対象ソフトウェア

プログラミングツール[EcoStruxure Machine Expert]

画面作成ソフトウェア[GP-Pro EX]

### ■ 解説している実施環境

プログラマブルコントローラー : Modicon M241シリーズ(TM241CEC24U)

プログラマブル表示器 : GP-4501T(PFXGP4501TAA)

パソコン : Windows10が動作するパソコン

※本テキストは、[EcoStruxure Machine Expert バージョン1.1][GP-Pro EX バージョン4.08]で作成されています。ソフトウェアのバージョンにより、表記方法等が異なる場合があります。

本テキストは、以下の方を対象にしております。

- ・PLCに対して知識のある方
- ・表示器の画面作成の経験のある方

## おことわり

- (1) 本テキストは、シュナイダーエレクトリックホールディングス株式会社(旧：株式会社デジタル)の著作物です。文章の一部または全部の転載利用はお断りします。
- (2) 本テキストの内容については万全を期して作成しておりますが、万一お気づきの点がありましたら、シュナイダーエレクトリックカスタマーケアセンターまでご連絡ください。
- (3) 本テキストを運用した結果の影響及び第三者のいかなる請求にも、シュナイダーエレクトリックホールディングス株式会社は一切責任を負いません。
- (4) 製品の改良のため、本テキストの記述とソフトウェアとの間に異なった部分が生じることがあります。最新の説明は、各製品のマニュアルをご参照いただくか、シュナイダーエレクトリックカスタマーケアセンターまでお問い合わせください。
- (5) 本テキストで使用する製品が、記録・表示する情報の中にシュナイダーエレクトリックホールディングス株式会社及び第三者が権利を有する無体財産権、知的所有権に関わる内容を含む場合がありますが、これはシュナイダーエレクトリックホールディングス株式会社これらの権利の再利用について、ユーザー及びその他の第三者に、何らかの保証や許諾を与えるものではありません。

## 安全にご使用いただくために

### 設計上の危険事項

・外部電源の故障やM241/M251本体の故障およびM241/M251の誤作動の場合、システム全体が安全側に働くようにM241/M251の外部で安全回路を設けてください。誤出力/誤動作により、事故の恐れがあります。

(1) 非常停止回路、保護回路、正転/逆転などの相反する動作のインターロック回路、位置決めの上限/下限/走行限など機械の破損防止のインターロック回路などは、M241/M251の外部で回路構成してください。

(2) M241/M251で検出できない入出力制御部分などの異常時は、入出力部の動作について予想のできない動きとなる可能性があります。このとき、機械の動作が安全側に働くよう、M241/M251の外部でフェールセーフ回路を構成したり、機構を設けてください。

(3) 出力ユニットのリレーやトランジスタなどの故障によっては、出力がONのままになったり、OFFのままになったりすることがあります。重大な事故につながるような出力信号については、外部で監視する回路を設けてください。

・M241/M251の本体の電源を立ち上げる前に、M241/M251本体に接続されているI/Oユニットおよび負荷制御電源を投入するように回路を構成してください。M241/M251のプログラムがRUNした後に負荷制御用電源が立ち上がると、誤出力や誤動作により事故の恐れがあります。

・人的損害や物的損害をもたらす可能性があるスイッチは、絶対にプログラマブル表示器上に作らないでください。本体、ユニット、ケーブル等の故障により、意図しない出力信号が出て重大な事故につながる可能性があります。重大な動作を行うスイッチはプログラマブル表示器本体以外の装置より行うようにシステム設計をしてください。

・通信異常で機械が誤動作しないようにシステム設計を行ってください。人体に傷害を負ったり、物的損害の恐れがあります。

・M241/M251は航空機器、航空宇宙機器、幹線通信機器、原子力制御機器、生命の維持に関わる医療機器などの極めて高度な信頼性・安全性が求められる用途への使用を想定しておりません。これらの用途には使用できません。

・M241/M251を運送機器（列車、自動車、船舶等）、防災防犯装置、各種安全装置、生命の維持に関わらない医療機器などの、機能・精度において高い信頼性・安全性が求められる用途で使用する場合は、組み込まれるシステム機器全般として、冗長設計、誤動作防止設計等の安全設計を施す必要があります。

## お問い合わせ

本テキストの内容に関するご質問は、「シュナイダーエレクトリックカスタマーケアセンター」までご連絡ください。

シュナイダーエレクトリックホールディングス株式会社(旧：株式会社デジタル)

シュナイダーエレクトリックカスタマーケアセンター：0570-056-800

<http://jpn.proface.co.jp/customer/contact.html>

平日 9:00～19:00

土日祝日(弊社指定の休業日を除く) 9:00～12:00、13:00～17:00

## **取り付け上の警告事項**

- ・M241/M251の解体は絶対に行わないでください。高電圧部分がM241/M251内部にあり、M241/M251を解体すると感電の恐れがあります。
- ・M241/M251は改造しないでください。火災、感電の恐れがあります。
- ・可燃性ガスのあるところでは、使用しないでください。爆発の恐れがあります。

## **配線上の警告事項**

- ・取り付け、配線などは、必ず電源が供給されていないことを確認してから行ってください。感電や機器の破損の恐れがあります。
- ・配線後は必ず付属の端子台カバーを取り付けてください。端子台カバーを取り付けないと感電の恐れがあります。
- ・マニュアルに記載された仕様以外での環境で使用しないでください。仕様の範囲外で使用すると、感電、火災、誤動作や製品の破損の恐れがあります。

## **立ち上げ・保守時の警告事項**

- ・通電中に端子に触れないでください。感電の恐れや誤動作の原因になります。
- ・清掃や端子ネジの増し締めは、通電されていないことを確認してから行ってください。通電中に行くと感電の恐れがあります。
- ・M241/M251は時計のバックアップのためにリチウム電池を内蔵しています。電池を誤って交換すると、電池が爆発する恐れがありますので、交換は行わないでください。
- ・交換が必要な場合には、お買い求めの代理店またはシュナイダーエレクトリックカスタマーケアセンターまでご連絡ください。

## **設計上の注意事項**

- ・入出力信号線の配線は動力回路のケーブルとは、別ダクトにしてください。ノイズにより、誤作動の原因になります。

## **取り付け上の注意事項**

- ・ケーブルは、コネクタに確実に装着してください。接触不良により、誤入力や誤出力の恐れがあります。

## **配線上の注意事項**

- ・FG端子は、M241/M251専用のD種接地工事を行ってください。感電や誤動作の恐れがあります。
- ・M241/M251への配線は、定格電圧および端子配列を確認した上で正しく行ってください。定格と異なった電源の接続や誤った配線を行うと火災や故障の恐れがあります。
- ・端子ネジは規定のトルクで締め付けてください。端子ネジの締め付けがゆるいと短絡、火災や誤動作の恐れがあります。
- ・M241/M251内に、切粉や配線くずなどの異物が入らないように注意してください。火災、故障や誤動作の恐れがあります。

## **立ち上げ・保守時の注意事項**

- ・運転中のプログラム変更、強制変更、RUN、STOP、PAUSE等の操作はマニュアルおよびオンラインヘルプを熟読し、十分に安全を確認してから行ってください。操作ミスにより機械の破損や事故の原因になります。
- ・I/Oユニットの脱着は、通電されていないことを確認してから行ってください。通電中に行くとI/Oユニットの故障や誤動作の原因になります。

## **廃棄時の注意事項**

- ・M241/M251を廃棄する時は、産業廃棄物として扱ってください。

序章	プログラマブルコントローラーModicon M241/251シリーズ	
	Ⅰ. テキストのご紹介	序-1
	Ⅱ. Modicon M241/M251シリーズ/EcoStruxure Machine Expertのご紹介	序-2
	1. Modicon M241/M251シリーズの特徴	序-2
	2. プログラミングソフトウェア[EcoStruxure Machine Expert]	序-6
第1章	プログラミングツール EcoStruxure Machine Expertの基礎知識	
	Ⅰ. 開発環境を整えよう	1-1
	1. 開発環境	1-1
	Ⅱ. EcoStruxure Machine Expertの基礎知識	1-2
	1. プロジェクトの作成方法	1-2
	Ⅲ. まずは、これだけ！基本操作！	1-4
	1. プログラミングの基本操作	1-4
	2. POUの追加	1-5
	3. 命令の割付け	1-7
	4. 変数の割付け	1-8
	5. ビルドから運転まで	1-10
第2章	【ラダープログラム】基本回路をマスターしよう	
	Ⅰ. 【やってみよう】自己保持回路	2-1
	1. 自己保持回路とは	2-1
	2. 【解説】自己保持回路	2-2
	3. ビルドから運転まで	2-8
第3章	【FBD】基本回路をマスターしよう	
	Ⅰ. 【やってみよう】タイマー回路	3-1
	1. タイマー回路とは	3-1
	2. 【解説】タイマー回路	3-3
	3. ビルドから運転まで	3-8
第4章	【ST】基本回路をマスターしよう	
	Ⅰ. 【やってみよう】比較条件回路	4-1
	1. タイマー回路とは	4-1
	2. 【解説】比較条件回路	4-2
	3. ビルドから運転まで	4-5
第5章	ビジュアライゼーション機能	
	Ⅰ. 【やってみよう】ビジュアライゼーション	5-1
	1. ビジュアライゼーション	5-1
	2. 【解説】ビジュアライゼーション機能	5-3
第6章	Pro-face製表示器との接続	
	Ⅰ. Modicon M241シリーズとPro-face製表示器	6-1
	1. 3つの接続方法	6-1
	Ⅱ. Pro-face製表示器との接続方法	6-2
	1. CoDeSysイーサネット接続の流れ	6-2
	2. EcoStruxure Machine Expertの設定	6-3
	3. GP-Pro EXの設定	6-5
	Ⅲ. 【やってみよう】Pro-face製表示器との接続	6-8
	1. Pro-face製表示器との接続	6-8
	2. 【解説】Pro-face製表示器との接続	6-9
付録	さらに便利に使おう	
	1. ファームウェアのアップデー(Controllor Assistant)	付-1
	2. ソースのダウンロードとアップロード	付-3
	3. シミュレーション(ソフトウェアでのデバッグ)	付-4
	4. デバイスアドレス(%MW)	付-7
	5. 変数の種類	付-8
	6. メモリ情報の確認	付-10
	7. ビジュアライゼーション機能の便利な使い方	付-11
	8. Webサーバー	付-12

# 序章

## プログラマブルコントローラー Modicon M241/251シリーズ

I . テキストのご紹介	序-1
II . Modicon M241/M251シリーズ/EcoStruxure Machine Expertのご紹介	序-2
1. Modicon M241/M251シリーズの特徴	序-2
2. プログラミングソフトウェア[EcoStruxure Machine Expert]	序-6

## I テキストのご紹介

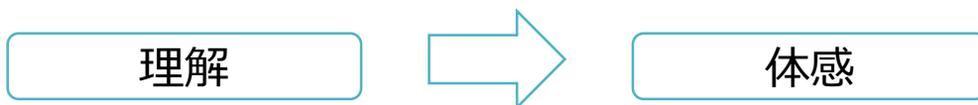
### (1) テキストについて

本テキストは、「EcoStruxure Machine Expert」の基本的なプログラム作成方法を体感いただくためのテキストです。プログラム作成をユーザー様ご自身で効率良く行なっていただけるよう構成されています。また、第7章にはPro-face製表示器との連携方法もあわせて解説しています。



### (2) テキストの使い方

本テキストは、序章から付録まで構成されています。各章ごとにテーマを設け、[EcoStruxure Machine Expert][Modicon M241/251シリーズ]のさまざまな機能や設定方法を「体感」できる内容になっています。まずは、序章と第1章で概要を理解し、第2章から第6章で各機能を体感します。



### (3) テキストの構成

本テキストでは、各章ごとに用途別のプログラムを作成します。

#### **序章：プログラマブルコントローラ [Modicon M241/251シリーズ]**

[M241/M251シリーズ]の特徴やメリットを紹介してします。

#### **第1章：プログラミングツール [EcoStruxure Machine Expert] の基礎知識**

[EcoStruxure Machine Expert]を使った[M241シリーズ]のプログラミング作成方法の基礎知識を説明しています。

#### **第2章：[ラダープログラム] 基本回路をマスターしよう**

[EcoStruxure Machine Expert]を使ったラダープログラムの作成方法を体感できます。

#### **第3章：[FBD] 基本回路をマスターしよう**

[EcoStruxure Machine Expert]を使ったファンクションブロックダイアグラムの作成方法を体感できます。

#### **第4章：[ST] 基本回路をマスターしよう**

[EcoStruxure Machine Expert]を使ったストラクチャードテキストの作成方法を体感できます。

#### **第5章：ビジュアル化機能**

Webブラウザから、M241/M251にアクセスし、遠隔操作するビジュアル化機能を体感できます。

#### **第6章：Pro-face製表示器との接続**

Pro-face製表示器との接続方法を解説してします。

#### **付録：さらに便利に使おう**

EcoStruxure Machine Expertの便利な使い方やツールを紹介してします。

## II Modicon M241/M251シリーズ/EcoStruxure Machine Expertのご紹介

### 1. Modicon M241/M251シリーズの特徴

Modicon M241/M251シリーズは、IoTに対応した小型で高性能なプログラマブルコントローラーです。M241とM251のCPU性能やサポートされる基本機能は同じで、M241はCPU本体とI/Oが一体になったモデルです。

M251はI/Oを装備してなく、ネットワーク接続に特化した製品で、イーサネットIFを標準で装備しています(モジュールを増設することでI/Oを付加することも可能です)

主な仕様は、プログラム容量は128Kステップで、CPUの処理速度は22nsです。内部で使用できる固有のデータメモリーは60,000ワードになりますが、それ以外にも、変数で自由にデータを宣言することができます。

現場のIoTソリューションを実現するために、OPC-UA、ModbusTCP、EtherNet/IPなどのIoTのトレンド規格に対応しています。また、ネットワーク上のセキュリティにもAchilles認証を取得しているので安心です。

機械災害の防止のためのセーフティ拡張モジュールを採用することでSIL2/SIL3の安全レベルを実現できます。

#### 【CPUの主な仕様】

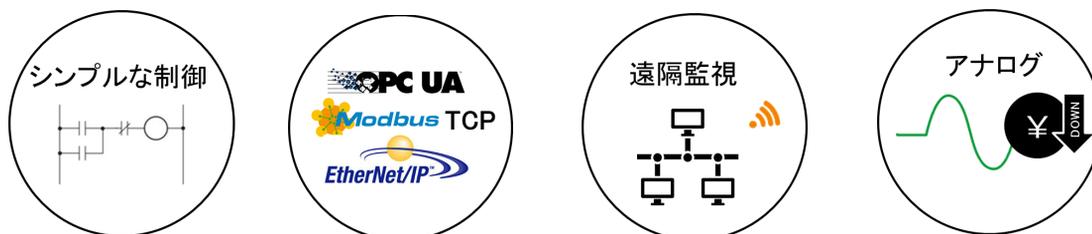
	CPUスピード	プログラム量	ワード数	RAM容量	FEPROM容量
M241/M251	22 ns/ブール命令	128Kステップ	60,000ワード※	64MB	128MB

※変数の宣言を行えば、数百万以上の変数を使用することができます。

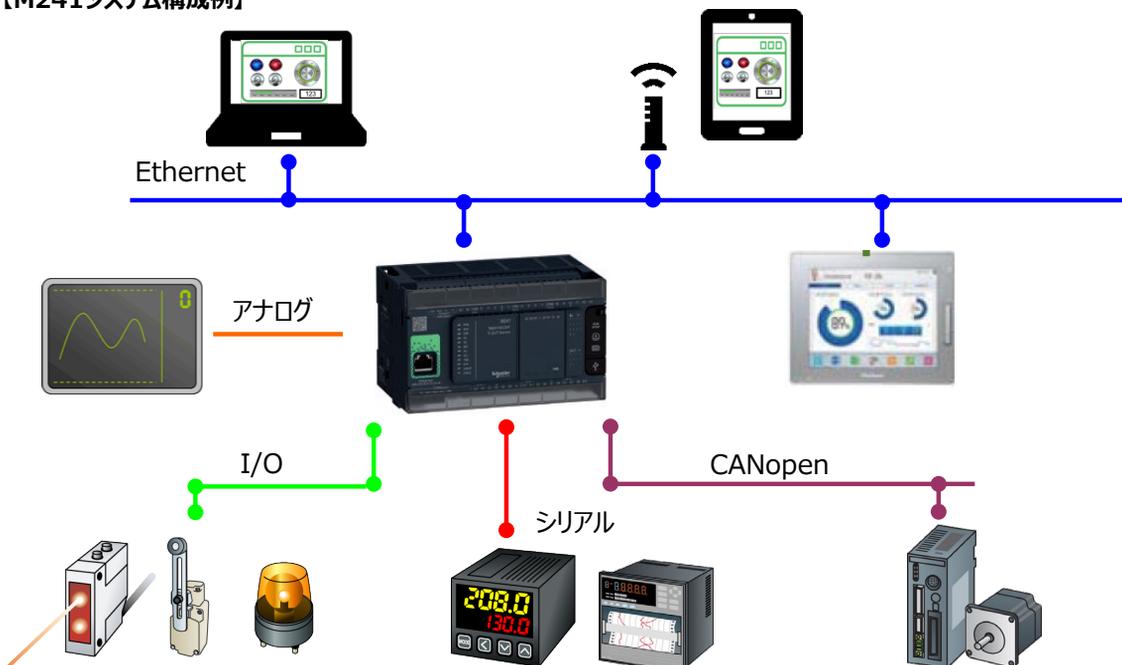
#### (1) Modicon M241シリーズとは

M241は、組み込み入出力点数が24点と40点とあり、出力タイプはリレー、トランジスタ(シンクまたはソース)から選択できます。イーサネット仕様では、Webサーバー/ビジュアルゼーション機能を使用することができます。

専用オプションの拡張カートリッジを本体に組み込むことで、外形を変えずに簡単にアナログなどの増設もできます。

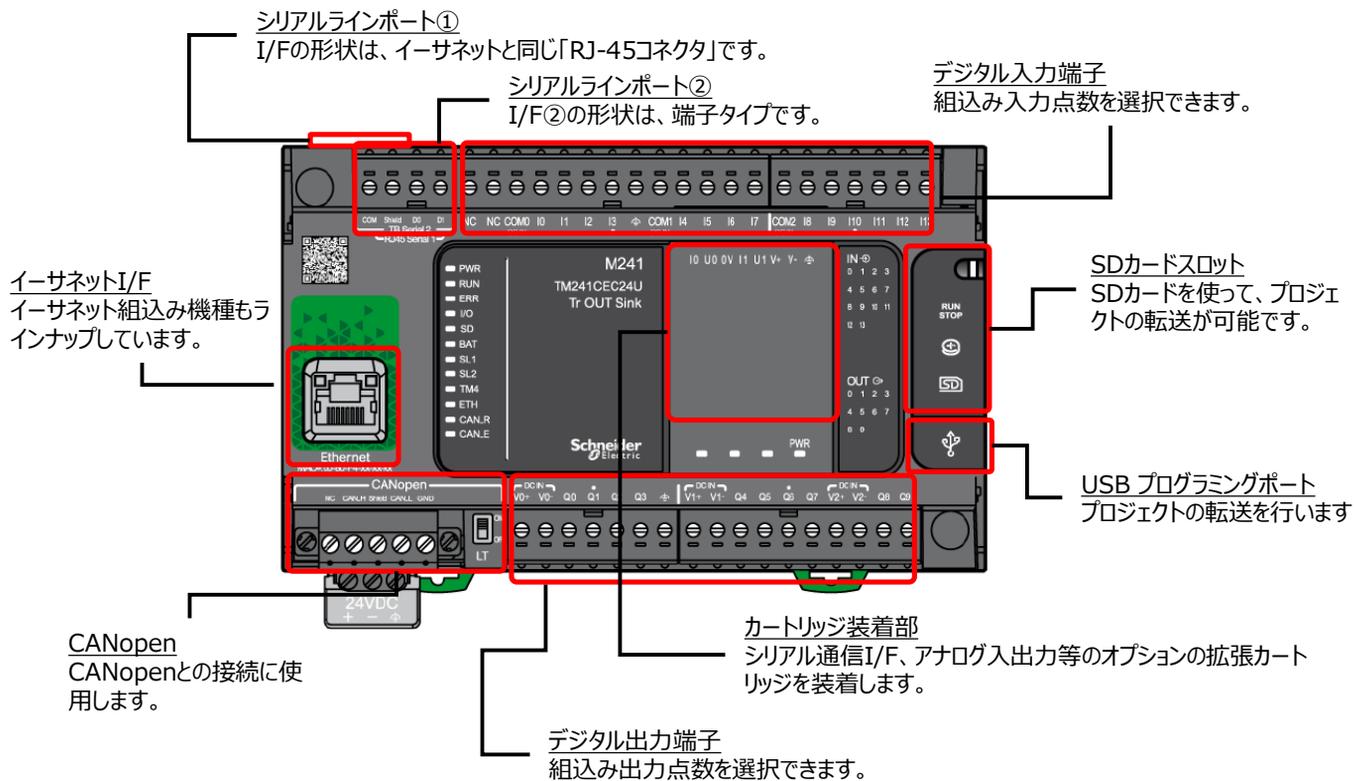


#### 【M241システム構成例】



## (2)Modicon M241のインターフェイス

M241にはさまざまなインターフェイスを備えており、拡張性に優れております。



※TM241CEC24U



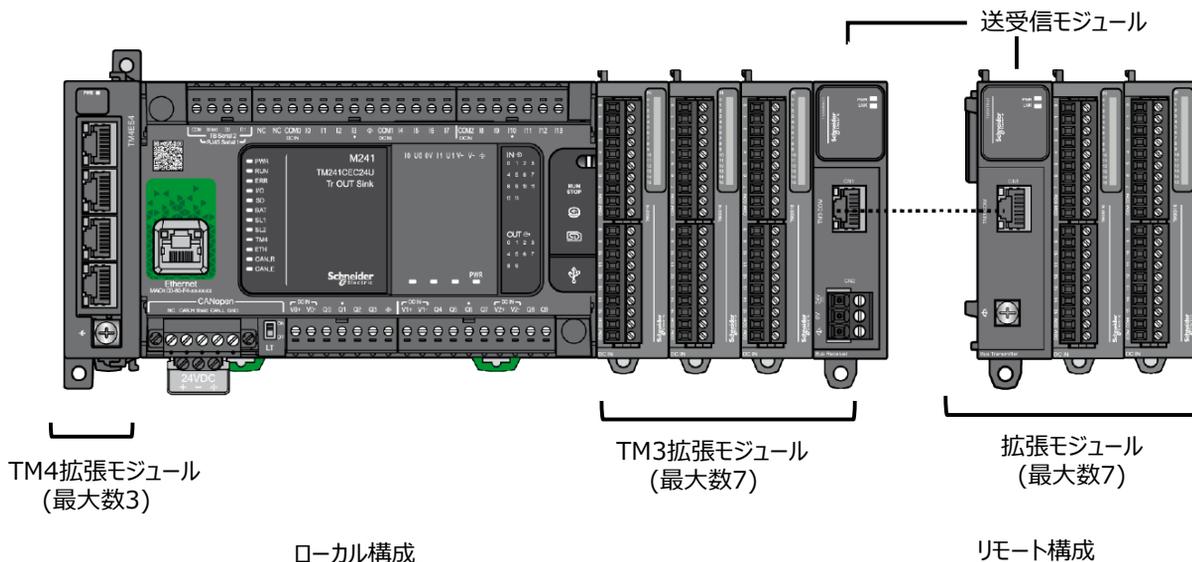
### 拡張モジュール

M241シリーズは、下記の拡張モジュールを追加することでシステムを拡張することができます。また、バス拡張モジュールを追加することでリモートの構成も可能です。

TM4拡張モジュール：イーサネットやPROFIBUS通信用のモジュールです。

TM3拡張モジュール：入出力/セーフティ、送受信用のモジュールです。

TM2拡張モジュール：入出力用のモジュールです。

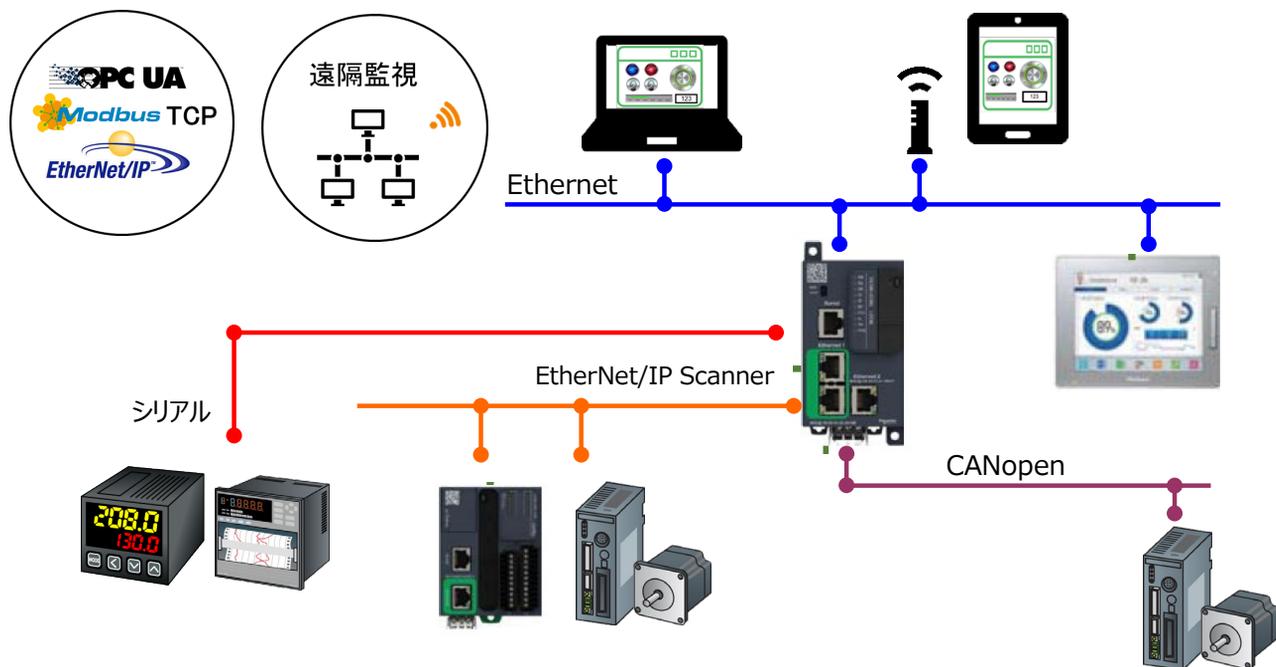


### (3)Modicon M251シリーズとは

M251は、I/Oを標準装備していないネットワークに特化してPLCです。各種機器との接続はEthernetやシリアル通信で接続します。情報端末として各種制御機器の情報を取りまとめ、Webサーバー/ビジュアルゼーション機能を使用し、上位PCと共有できます(モジュールを増設することでI/Oを付加することも可能です)

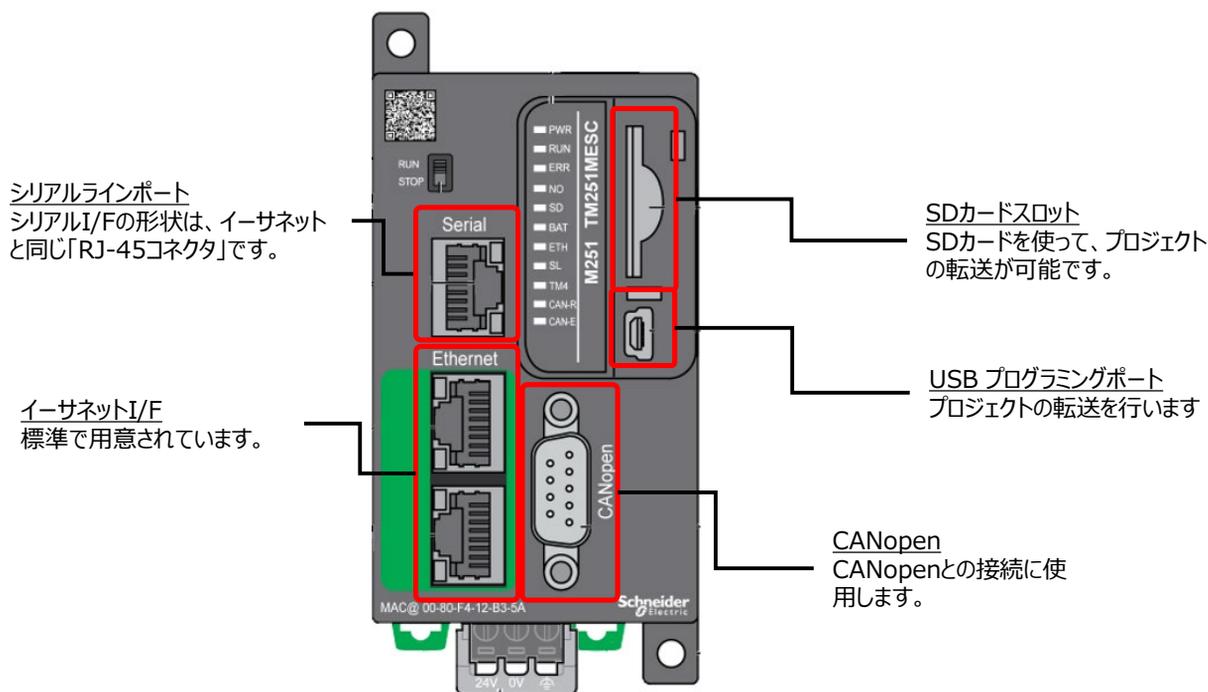
M251シリーズはCANopen対応モデルと未対応モデルの2機種がラインナップされています。

#### 【M251システム構成例】



### (4)Modicon M251のインターフェイス

M251にはさまざまなインターフェイスを備えており、拡張性に優れております。



※TM251MESC

CHECK!

## Modicon M241/M251シリーズのラインナップ

## 【Modicon M241シリーズ】

M241シリーズは、I/O点数(入出力)24点モデルと40点モデルに大別されます。また、出力はリレー/シンク/ソースから選択できます。ネットワーク機能を搭載したイーサネットモデルが得にお勧めです。



24点タイプ



40点タイプ

デジタル入力	出力		シリアル通信	電源	イーサネット無	イーサネット有	イーサネット/ CANopen有
	トランジスター出力	リレー出力					
14点	4点ソース	6点リレー	2ch	AC100-240V	TM241C24R	TM241CE24R	TM241CEC24R
	10点ソース	-		DC24V	TM241C24T	TM241CE24T	TM241CEC24T
	10点シンク	-		DC24V	TM241C24U	TM241CE24U	TM241CEC24U
24点	4点ソース	12点リレー		AC100-240V	TM241C40R	TM241CE40R	-
	16点ソース	-		DC24V	TM241C40T	TM241CE40T	-
	16点シンク	-		DC24V	TM241C40U	TM241CE40U	-

## 【Modicon M251シリーズ】

M251シリーズは、CANopen通信を搭載したモデルとそうでないモデルの2機種がラインナップされています。外形サイズはいずれも同じです。



イーサネット2点タイプ

イーサネット1点+CANopen  
タイプ

デジタル入力	出力		シリアル通信	電源	イーサネット	CANopen	型式
	トランジスター出力	リレー出力					
-	-	-	1ch	DC24V	2ch	-	TM251MESE
-	-	-			1ch	1ch	TM251MESC

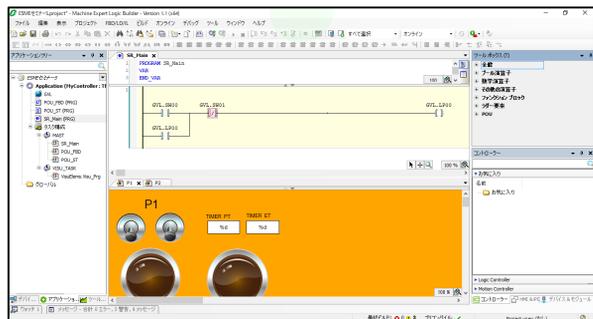
## 2. プログラミングソフトウェア[EcoStruxure Machine Expert]

### (1) EcoStruxure Machine Expertとは

EcoStruxure Machine Expertは、M241/M251専用プログラミングソフトウェアです。

#### 【EcoStruxure Machine Expertの特徴】

- 直感的な操作でプログラム作成が可能
- IEC 61131-3の5言語に対応
- IoT化に有効なネットワーク機能を搭載
- ビジュアライゼーション機能搭載で、遠隔監視/操作が可能



#### 【EcoStruxure Machine Expert インストール推奨仕様】

- OS :
- Windows 7 Professional 32bit/64bit
  - Windows 8 Professional 32bit/64bit
  - Windows 8.1 32bit/64bit
  - Windows 10
- CPU: Intel Core i7相当  
 HDD : 15 GB hard disk  
 メモリー : 8 GB RAM  
 推奨ディスプレイ解像度 : 1680 x 1050

### (2) 5つのプログラミング言語に対応

[IEC 61131-3]とは、国際電気標準会議(IEC)がPLC用のプログラム言語を定義した世界共通の標準規格です。

この規格では、日本で普及しているラダーダイアグラムを含む5種類の言語が規定されており、使用者のスキルや目的に合った言語が使うことができます。

EcoStruxure Machine Expertは、[IEC 61131-3]に準拠していることにより、制御システムの開発において最も普及しているラダー方式だけでなく、IL(Instruction List)やSFC(Sequential Function Chart)にも対応しています。

	名称	図	説明
グラフィック言語	LD (ラダーダイアグラム)		日本で最も普及しているプログラミング言語です。ビット単位の処理には向いていますが、システムが大規模かつ複雑になるほど機能単位でのモジュール化が難しくなってきます。
	FBD (ファンクションブロックダイアグラム)		電子部品(ファンクションと呼ばれる箱)と接続する配線により、あたかも電子回路を設計するようにプログラムを記述でき、データの流れわかりやすいです。計装分野を中心に使われています。
	SFC (シーケンシャルファンクションチャート)		製造ラインなどの状態遷移を記述するのに適したグラフィック言語です。
テキスト言語	IL (インストラクションリスト)	<pre>LD      BVar1 ST      tonInst1.IN CAL      tonInst1(           PT:=t1,           ET=&gt;tOut2) LD      toninst1.Q JMPC    mark1 ST      tonInst2.IN</pre>	一連の命令をリストとして記述する方法で、コンピュータのアセンブリ言語によく似ています。アプリケーションの小型化や高速化に有効ですが、他の言語と比べてメンテナンス性に劣ると言われています。
	ST (ストラクチャードテキスト)	<pre>1 2  IF GVL.TEMP &gt; 50 THEN 3      GVL.ALARM:= 1; //アラームが発報 4  ELSE 5      GVL.ALARM:= 0; //アラームが復旧 6  END_IF 7</pre>	マイコンボードの開発者やC、C++系のエンジニアに向いています。特に数値演算式やデータ処理など、LDが苦手としている用途で効果を発揮します。



## PLCの機種選定時のポイント

項目		確認すること	説明
I/O点数 (デジタル入出力点数)		<ul style="list-style-type: none"> <li>• 入力点数</li> <li>• 出力点数</li> </ul>	接続する入出力機器の数をI/O点数と呼びます。PLCの選定では、最大入出力点数を確認します。実際は、入出力の追加も生じるので、10～30%上乗せしたI/O点数で選定します。
アナログ入出力	アナログ入力	<ul style="list-style-type: none"> <li>• 点数 (c h)</li> <li>• 入力方式 (0-10V / 4-20mA 等)</li> </ul>	電圧や電流などのアナログ信号をデジタル信号に変換し、PLCに取り込みます。アナログ入力ユニットを用いることが多く、一般的には、1つのユニットに複数チャンネルの接続ができます (アナログ入力1点を、チャンネル (c h) と呼びます) また、入力方式は、電圧入力 (0-10V)、電流入力 (4-20mA) が多く使われています。
	アナログ出力	<ul style="list-style-type: none"> <li>• 点数 (c h)</li> <li>• 出力方式 (0-10V / 4-20mA 等)</li> <li>• 出力先の機器</li> </ul>	アナログ出力は、インバータなどの速度調整やバルブなどの開度指示などアナログ量を制御する場合に使われます。アナログ入力ユニット同様、アナログ出力は電圧信号 (0～10V) と電流信号 (4～20mA) で出力されるものが多く使われています。
その他の入出力	高機能入出力 (特殊入出力)	<ul style="list-style-type: none"> <li>• 入出力先の機器</li> <li>• 必要なユニット (通信方式)</li> </ul>	PLCを用いての制御は、応用の範囲が大幅に広がっています。そのため、入出力ユニットに特別な機能を持たせた特殊ユニットが豊富にあります。位置決め/高速カウンタ/パルス入力/PID制御等のユニットの使用の有無。また、PCやHMIと接続する場合は、機器名や通信ユニット、通信方式 (シリアル/イーサ等) を確認します。
	リモートI/O	<ul style="list-style-type: none"> <li>• リモート入力の点数</li> <li>• リモート出力の点数</li> <li>• リモートI/Oの種別</li> </ul>	PLCから数百m離れた場所に入出力機器がある場合、機器の点数分 (多量の) ケーブルを延々と引き回すと工事費がかさみますし、さまざまなトラブルの元になります。リモートI/Oユニットを使うと、PLCと機器間のケーブルは一本から数本で済みます。
メモリ	ユーザー (プログラム) メモリ	<ul style="list-style-type: none"> <li>• 必要ステップ数</li> </ul>	ユーザーが作成したラダープログラムを記憶します。記憶できる量をプログラム容量といい、H/Wの物理的な容量 (ワード数)、プログラムの大きさ (ステップ数) で表します。必要なプログラム容量は最大入出力点数とほぼ比例し、また、制御内容が複雑になればプログラム容量も増加します。
	データメモリ	<ul style="list-style-type: none"> <li>• 必要ワード数</li> </ul>	入出力からのデータ、演算結果を記憶する領域です (具体的には、補助リレー、タイマ・カウンタ、データレジスタ等) 多くの場合、ワード数で表します。プログラムメモリと同様で、入出力点数、制御の複雑さなどを考慮して選定する必要があります。
処理機能	処理速度	※最終的に、実機でデバッグをして適正かを判断するケースが多い。	PLCの処理速度は、各PLCメーカーにより、指標がまちまちです。主に下記の3つが使われます。 <ul style="list-style-type: none"> <li>• 基本命令の実行速度：接点やコイル等の基本的な命令の実行時間です。</li> <li>• スキャンタイム：1Kステップあたりのスキャンタイム</li> <li>• PLC MIX：1μsで実行する命令数です。数値が大きいほど処理速度が速いです</li> </ul>
	演算機能	<ul style="list-style-type: none"> <li>• 特殊な命令が必要か</li> </ul>	一般的な演算機能 (命令) はどのPLC/プログラミングツールでも持っていますが、ユーザーによっては、特殊な命令を必要としている場合があります。
プログラミング (ソフトウェア)	言語	<ul style="list-style-type: none"> <li>• 使用しているプログラム言語</li> </ul>	日本ではラダープログラム (LD) が一般的ですが、FBD、SFC、IL、STの全5種類の言語があり、スキルや目的に合った言語を使います。
	プログラミングツール	<ul style="list-style-type: none"> <li>• 使用しているソフトウェア</li> </ul>	プログラミングやデバッグを行うソフトウェアの使い勝手も選定の要素です。また、ソフトウェアによっては、モニタリングやドキュメント作成等の運用のサポートもできます。
入出力電源タイプ	入力電源タイプ	<ul style="list-style-type: none"> <li>• 入力タイプ</li> </ul>	一般的には、AC100～200V、DC12～24Vがあります。
	出力電源タイプ	<ul style="list-style-type: none"> <li>• 出力タイプ (シンク or ソース or リレー)</li> </ul>	外部機器への出力タイプです。一般的にはトランジスタ出力 (シンク/ソース) とリレー出力があります。
規格対応	<ul style="list-style-type: none"> <li>• 使用地域 (国)</li> <li>• 必要な規格</li> </ul>	特に海外への輸出する場合は、海外規格取得機種を選定します。	
拡張性	<ul style="list-style-type: none"> <li>• 拡張の予定</li> </ul>	ビルディングブロックタイプは将来の拡張に対応しやすいです。	
環境性	<ul style="list-style-type: none"> <li>• 使用場所は特殊な環境か</li> </ul>	PLCが使われる使用環境が特殊な場合は、その環境を確認します (使用周囲温度、使用周囲湿度、塵埃、腐食性ガス)	

# 第1章

## プログラミングツール EcoStruxure Machine Expertの 基礎知識

I . 開発環境を整えよう	1-1
1. 開発環境	1-1
II . EcoStruxure Machine Expertの基礎知識	1-2
1. プロジェクトの作成方法	1-2
III . まずは、これだけ！基本操作！	1-4
1. プログラミングの基本操作	1-4
2. POUの追加	1-5
3. 命令の割付け	1-7
4. 変数の割付け	1-8
5. ビルドから運転まで	1-10

# I 開発環境を整えよう

## 1. 開発環境

M241の開発には、下記の4つをご用意ください。



EcoStruxure  
Machine Expert



Modicon  
M241シリーズ



転送用ケーブル



Pro-face製表示器

### (1) EcoStruxure Machine Expert

Modicon M241/M251専用のプログラミング。ソフトウェアです。直感的な操作で初心者でも安心してお使いいただくことができます。



### (2) Modicon M241シリーズ

Modicon M241は、CPUや入出力などを一体化したオールインワンタイプでコストメリットの高いシンプルで小型/高性能のPLCです。ラインナップは、入出力40点タイプと20点タイプがあります。また、イーサネットを標準で搭載しているタイプは特にコストメリットが高くお勧めです。

M241の詳細は弊社HPにてご確認ください。

[https://www.proface.com/ja/product/schneider-plc/Modicon\\_M221/lineup](https://www.proface.com/ja/product/schneider-plc/Modicon_M221/lineup)

### (3) 転送用ケーブル

プロジェクトの転送方法は3つあります。



USBケーブル  
(A-MiniB)

市販のUSBケーブル(Type A to mini B)を用いて転送することができます。



LANケーブル

LANケーブル(イーサネットケーブル)を用いて転送することができます。



SDカード

現場にパソコンや専用ツールを持ち込まなくても、SDカードでプロジェクトの更新ができます。

### (4) Pro-face製表示器

GP4000シリーズ/SP5000シリーズ等のPro-face製表示器と接続することができます。

**CHECK!**



GP4000シリーズ/SP5000は全機種に標準でイーサネットのI/Fが付いています。



Pro-face製表示器の詳細は弊社HPにてご確認ください。

<https://www.proface.com/ja/product/hmi/top>

## II EcoStruxure Machine Expertの基礎知識

### 1. プロジェクトの作成方法

#### (1)プロジェクトとは

プロジェクトとはM241を稼働させるための[プログラム][通信プロトコル][システム]を一つのまとめたものです。プロジェクトは指定したフォルダ内に[. project]の拡張子で保存されます。

EcoStruxure Machine Expertで作ったプロジェクトをM241にダウンロードすることで、M241はシーケンス制御やI/O入出力を行ないます。

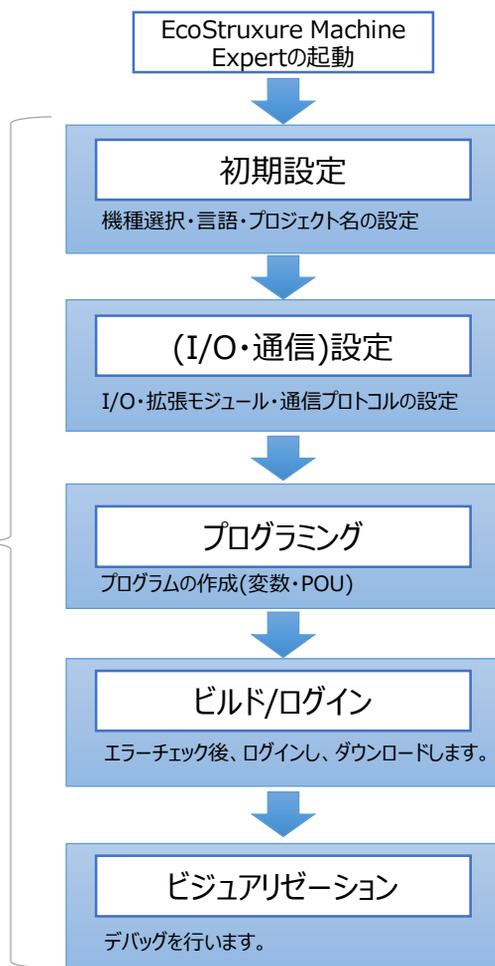
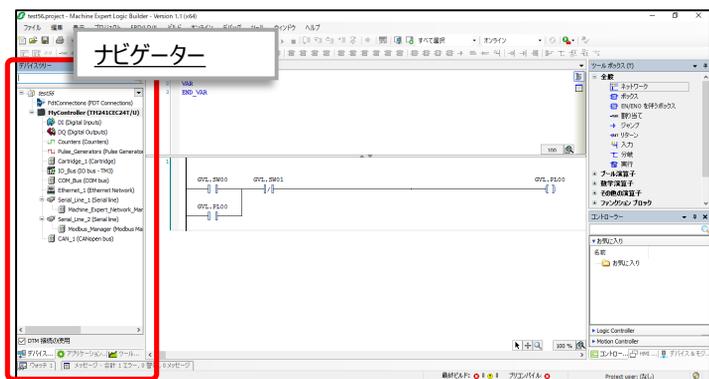


000.project

#### (2)プロジェクトの作成の流れ

プロジェクト作成の手順は、[EcoStruxure Machine Expert]の起動後、初期設定を行った後に、ナビゲーターで設定を切替ながら開発を行います。

作成したプロジェクトは、ビジュアライゼーション機能で動作確認後、M241に転送します。



CHECK!

#### 拡張子

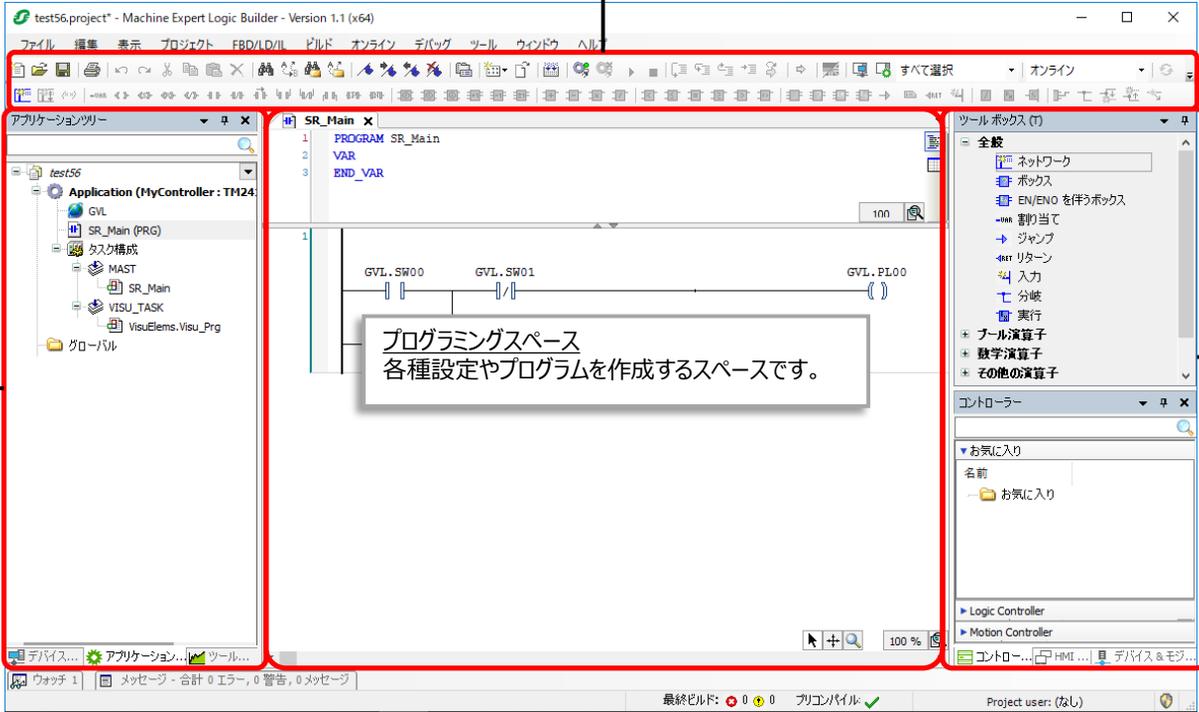
EcoStruxure Machine Expertでは、プロジェクトファイル(. project)以外にも下記の拡張子のファイルが生成されます。

.xlm : 変数情報を出したファイルです。Pro-face製表示器と通信させる際に使用します。

### (3)メインウィンドウ

メインウィンドウは使い勝手に応じて、カスタマイズを行うことができます。

ツールバー  
各種の機能をアイコン化し、ワンクリックで使用できます。



プログラミングスペース  
各種設定やプログラムを作成するスペースです。

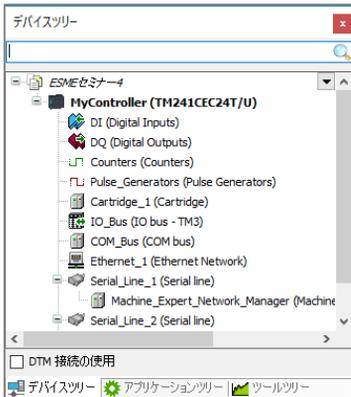
ナビゲーター  
プロジェクト作成の開始点です。ここからハード設定、プログラム作成などのすべての設定を開始します。

ワークスペース  
命令やオブジェクトの配置を行います。また、選択したオブジェクト類の属性を表示します。

### (4)ナビゲーター

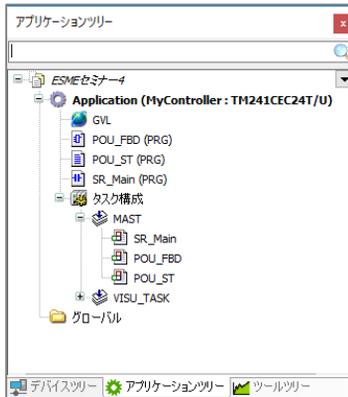
ナビゲーターとはプロジェクトを作成するための開始点です。[デバイスツリー][アプリケーションツリー][ツールツリー]の3つから構成されています。

【デバイスツリー】



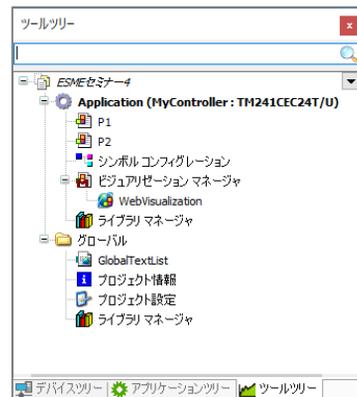
I/O設定や拡張モジュール、通信に関する設定を行います。

【アプリケーションツリー】



変数の作成、プログラミングを行います。

【ツールツリー】



ビジュアライゼーション機能等の設定を行います。

## Ⅱ まずは、これだけ！基本操作！

### 1. プログラミングの基本操作

#### (1) 代表的な基本操作

プログラミングでは、以下の順番に沿って行います。

- ① 命令の配置…………… プログラム内に、命令/接点を配置します。
- ② 変数の割付け…………… 変数を命令に割り付けます(変数は命令に割り付けることで自動宣言されます)
- ③ ビルト…………… プログラムをエラーチェック/コンパイルします。
- ④ ログイン…………… PLC本体にログインし、プロジェクトをダウンロードします。

ラダープログラムでは、上から下、左から右に向かって、信号が流れます。信号が命令に到達すると、命令内容に従い、処理を行います(変数を変化させます)

また、接点では、割当てた変数を変化させることで、接点をON/OFFします。

The screenshot displays the software interface for programming a PLC. On the left, the 'Application (MyController: ...)' tree shows a 'タスク構成' (Task Structure) folder containing 'MAST', 'SR\_Main', 'POU\_FBD', and 'POU\_ST'. Below this is a 'グローバル' (Global) folder containing 'VISU\_TASK' and 'POU'. A red box highlights the 'タスク構成' folder, with a callout box stating 'タスク内のPOUが実行されます。' (POUs within the task are executed). A bracket labeled 'ネットワーク' (Network) spans the ladder logic rungs. The main window shows the 'SR\_Main' program with the following variable declaration:

```

PROGRAM SR_Main
VAR
  Switch00: BOOL;
  Switch01: BOOL;
  Switch02: BOOL;
  Switch03: BOOL;
  CW: BOOL;
  CCW: BOOL;
END_VAR
  
```

The variable declaration is enclosed in a red box labeled '変数一覧' (Variable List). The ladder logic below shows two rungs. The first rung has inputs Switch00 (NO), Switch01 (NC), and CCW (NC) in series, leading to an output coil CW (NO). The second rung has inputs Switch02 (NO) and Switch03 (NC) in series, leading to an output coil CCW (NO). Red boxes highlight the variable names in the ladder logic, and a callout box labeled '変数' (Variable) points to the CW coil. Another callout box labeled '命令' (Command) points to the CCW coil.

CHECK!

#### タスク/POU/ネットワーク

EcoStruxure Machine Expertのプログラムは、[タスク][POU][ネットワーク]で構成されています。

- ・タスク：タスクに登録されたPOUが実行されます。
- ・POU(Program Organization Unit)：プログラムの実行単位です。POU内はネットワーク単位で区分されています。
- ・ネットワーク：入力から出力で構成される命令の集まり(行)です。

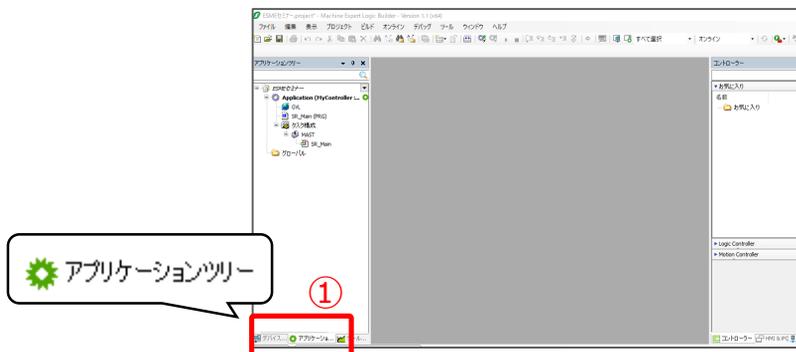
The screenshot shows the 'アプリケーションツリー' (Application Tree) on the left. It lists 'Application (MyController: TM241CEC)' with sub-items: 'GVL', 'POU\_01 (PRG)', 'POU\_02 (PRG)', and 'SR\_Main (PRG)'. Below these is a 'タスク構成' (Task Structure) folder containing 'MAST', 'SR\_Main', 'POU\_01', and 'POU\_02'. A red box highlights the 'タスク構成' folder, and a callout box labeled 'タスク' (Task) points to it. The 'グローバル' (Global) folder at the bottom contains 'グローバル'.

## 2. POUの追加

### (1)新規POUの追加

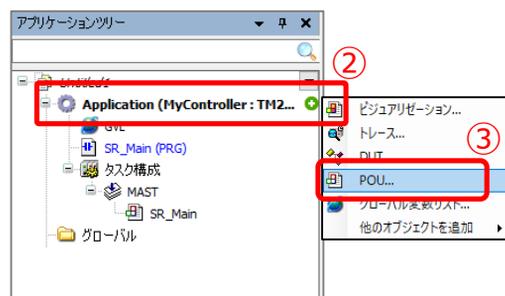
POU(Program Organization Unit)とはプログラムの実行単位です。プログラミングを行う際はPOUを追加し、プログラミングを行います。プログラミング後はPOUをタスクに移動します。なお、初期状態ではタスクには、[SR\_Main]という名称のPOUが登録されています。

①[アプリケーションツリー]に移動します。



②[Application]を選択し、 をクリックします。

③[POU]をクリックします。



④[POUの追加]でPOUの名称や記述言語を指定します。

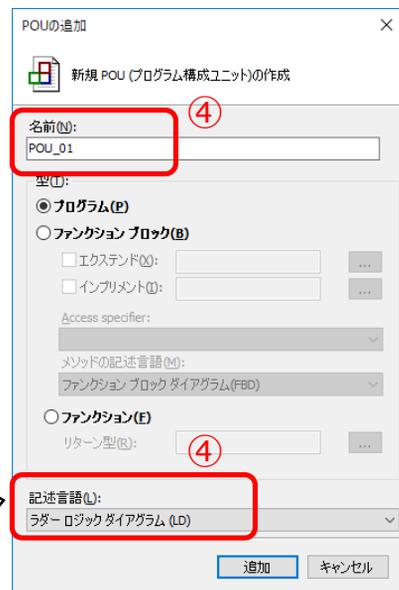
#### 【POUの追加】

名前: 任意(半角英数字のみ)

型: プログラム

記述言語: 任意

コンティニユアスファンクションチャート (CFC)  
 コンティニユアスファンクションチャート (CFC) - ページ指向  
 シーケンシャル ファンクション チャート (SFC)  
 ファンクション ブロック ダイアグラム (FBD)  
 ラダー ロジック ダイアグラム (LD)  
 構造化テキスト (ST)



⑤[追加]をクリックすると、アプリケーションツリーにPOUが表示されます。



⑥プログラム作成後は、作成したPOUをタスクにドラッグ & ドロップで移動します。  
 ※タスク内のPOUが実行されます。

CHECK!

## タスク

POUはタスクに登録することで実行されます。作成したPOUはドラッグ & ドロップで、タスク内に移動します(SR\_Mainは初期設定で用意されているPOUです)  
また、タスクには全部で4つの種類が用意されています。



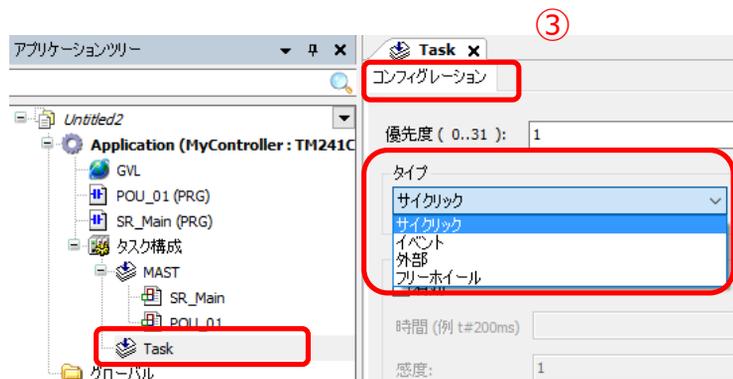
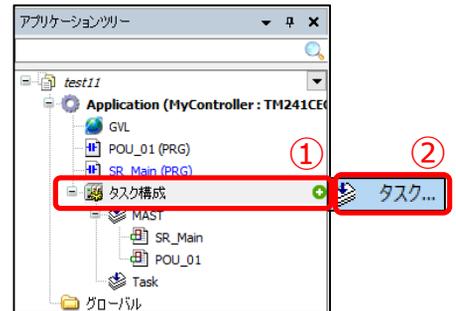
## タスクの種類

タスクには実行条件の違う4つが用意されています。

- ・サイクリック：設定周期毎に実行されます。
- ・フリーホイール：各処理の完了毎に実行されます。
- ・Event：データ変化などイベント発生時に実行されます。
- ・External event：通信などの外部イベント発生時に実行されます。

## タスクの追加方法

- ① [タスク構成]を選択し、 をクリックします。
- ② [タスク]をクリックします。
- ③ [コンフィグレーション]でタイプ(タスクの種類)を指定します



※新しいタスクが追加されました。

## 3. 命令の割付け

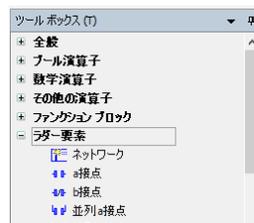
### (1) 命令とは

EcoStruxure Machine Expertは、さまざまな命令語を用意しています。命令に信号が導通すると、個々の役割に応じて動作します。各命令は、ツールバーにアイコン化されており、ネットワーク内に配置します。また、ツールボックスからの配置もできます。

#### 【ツールバーの命令アイコン】



#### 【ツールボックス】



### (2) 代表的な命令

【代表的な命令】 下記以外にも様々な命令を用意しています。

カテゴリ	命令の呼称	命令表記
接点	a接点	a接点
	b接点	b接点
	立上り接点	
	立下がり接点	
出力	コイル	コイル
	セットコイル	セット コイル
	リセットコイル	リセット コイル
制御	ジャンプ	ジャンプ
	リターン	リターン
タイマ命令	オンディレータイマ(TON)	TON
	オフディレータイマ(TOF)	TOF
カウンタ命令	加算カウンタ(CTU)	CTU
	減算カウンタ(CTD)	CTD
転送	転送(MOVE)	MOVE
演算	加算(ADD)	ADD
	減算(SUB)	SUB
	乗算(MUL)	MUL
	除算(DIV)	DIV
比較命令	比較(=)(EQ)	EQ
	比較(<>)(NE)	NE
	比較(<)(LT)	LT
	比較(<=)(LE)	LE
	比較(>)(GT)	GT
	比較(>=)(GE)	GE

## 4. 変数の割付け

### (1)変数とは

変数とはデータの格納場所(入れ物)です。PLCプログラムのデバイスアドレスにあたります。従来のPLCでは、デバイスアドレスというPLCベンダー特有の名称(記号や番号)で扱われますが、EcoStruxure Machine Expertではデバイスアドレスの代わりに変数を使用します。

変数は任意の名前を付けてプログラム上で使用します。変数には用途にあわせていくつかの変数タイプ(BOOL、INTなど)が用意されています。

それぞれの変数に自由に名前が付けられることで、プログラムの管理がしやすくなり、プログラム全体が読みやすくなるというメリットがあります。

#### 【EcoStruxure Machine Expertの変数と従来PLCのデバイスアドレス】

デバイス種別	外部出力	内部ルー	タイマ	データレジスタ
PLCメーカーA	X001	M100	T200	D0001
PLCメーカーB	01	1001	TIM000	DM0000
EcoStruxure Machine Expert	Start	Lamp	OperatingTime	Tank_Value

### (2)グローバル変数とローカル変数

変数は、プロジェクト全体で共有するグローバル変数(GVLに登録)と、POU毎に登録されるローカル変数に分かれます。グローバル変数は変数名の最初が[GVL.]になります。

	特徴	表記
グローバル変数	すべてのPOUで使用することができる	GVL.○○○ 例：GVL.SW00
ローカル変数	POU毎で使用する	○○○ 例：PL00

#### 【グローバル変数リスト(GVL)】

```
GVL x
1 {attribute 'qualified_only'}
2 VAR_GLOBAL
3 SW00: BOOL;
4 SW01: BOOL;
5 LP00: BOOL;
6 TON00: TON;
7 TON_ET: TIME;
8 TON_PT: TIME := TIME#5s0ms;
9 LP01: BOOL;
10 TEMP: INT;
11 ALARM: BOOL;
12 END_VAR
```

### (3)変数の割付方法

変数は利用したい変数タイプを自由に選択して作成できます。

取り扱うデータの種類	変数タイプ
0または1	BOOL
整数	INT、UINT、DINT、UDINT、LINT、ULINT、SINT、USINT等
文字列	STRING、WSTRING

```
BOOL
BYTE
DATE
DATE_AND_TIME
DINT
DWORD
INT
LINT
LREAL
LTIME
LWORD
REAL
SINT
STRING
TIME
TIME_OF_DAY
UDINT
UINT
ULINT
USINT
WORD
WSTRING
```

#### 【よく使う変数】

BOOL：ブール変数

0または1の値を格納します。

PLCの接点アドレスと同様に、0または1(OFFまたはON)を取り扱いたい場合に使用します。

INT(Signed INT)/符号あり整数変数

整数(-32768～32767)を格納します。PLCのデータレジスタを取り扱いたい場合に使用します。

この他にも実数を扱うREALや時刻を扱うTIMEなどの変数型も用意しています。

※変数の詳細は[付-8]をご参照ください。

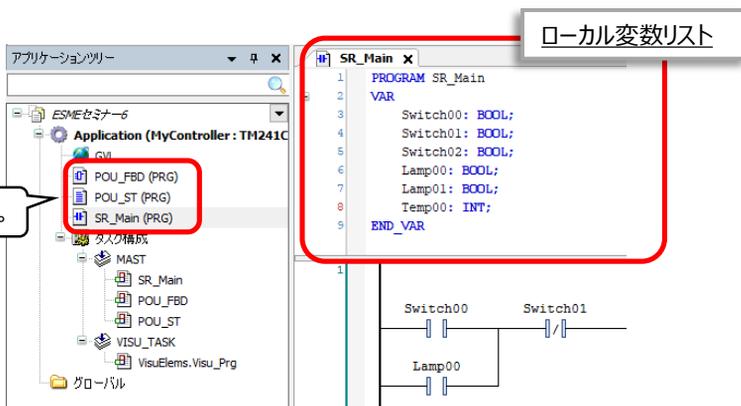
## (4)変数の確認方法

変数にはプロジェクト全体で共有するグローバル変数(GVLに登録)と、POU毎に登録されるローカル変数があります。それぞれの変数リストで確認することができます。

### 【ローカル変数リスト】

[アプリケーションツリー]のPOUをクリックします。  
ワークスペースの上部が変数リスト、下部がプログラムエディタになっています。

POUをクリックします。

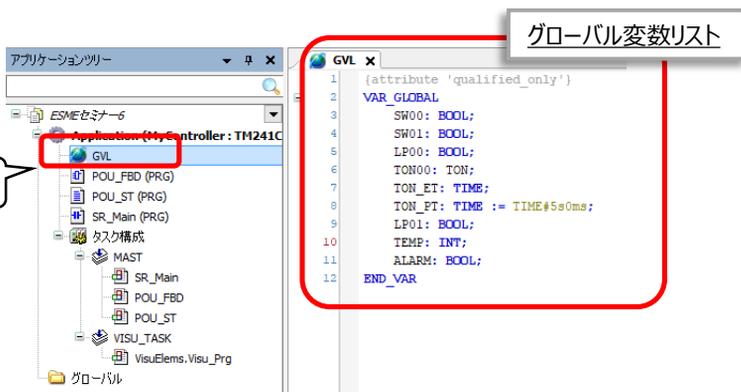


ローカル変数リスト

### 【グローバル変数リスト】

[アプリケーションツリー]の[GVL]をクリックします。

GVLをクリックします。



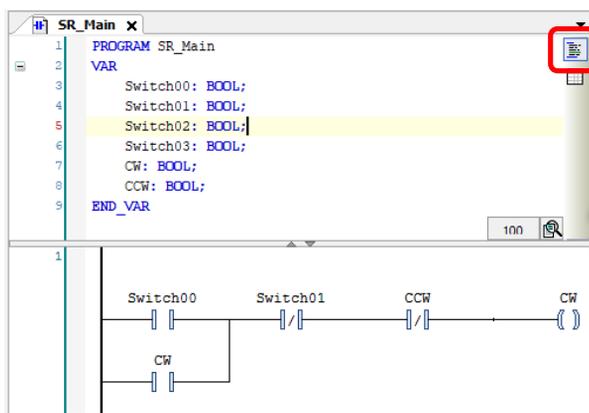
グローバル変数リスト

CHECK!

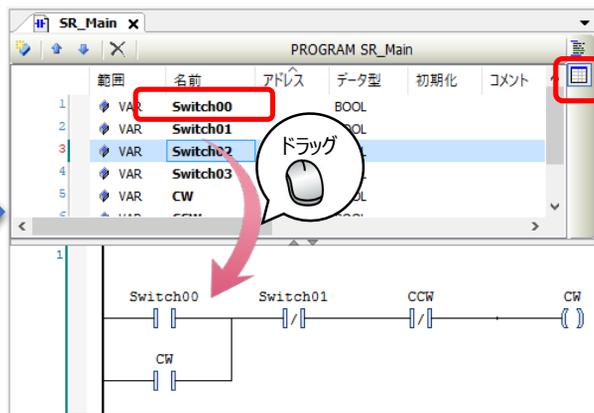
### 変数リスト

変数リストは、アイコンをクリックすることでテキスト形式と表形式を切り替えることができます。表形式ではドラッグ&ドロップで変数を割り付けることができます。

#### 【テキスト形式】



#### 【表形式】



コピー＆ペーストなどが容易で複数の変数を作成するのに便利です。

- ・ドラッグ&ドロップで変数を割り付けることができます。
- ・変数ごとの編集が容易です。
- ・配列変数等の作成が容易です。

## 5. ビルドから運転まで

ここではプログラム作成後の手順について説明します。  
プログラム作成後はビルドを行い、PLC本体にログインし、プログラムを書き込みます。

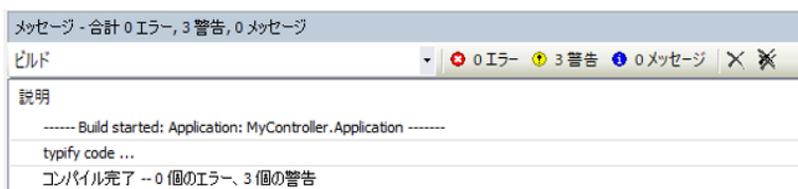


### (1)ビルド(エラーチェック)

プログラム作成後はビルドで、エラーチェックを行います。

エラー：ダウンロードを行うことができません。

警告：ダウンロードを行うことができます(警告内容は必ず確認してください)



### (2)ログイン

ログインを行うことで、PLC本体と論理的に接続され、プロジェクトがダウンロードされます。

① [デバイスツリー] をクリックします。

② [MyController] をクリックします。

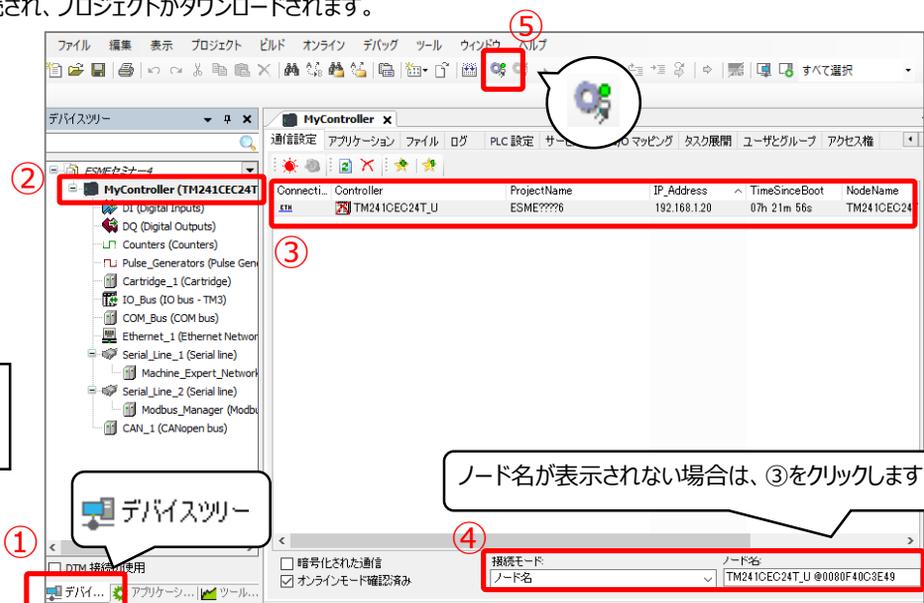
③ 接続するPLC をクリックします。

④ 接続モードを指定します。

接続モード：ノード名  
ノード名：TM241CEC24T\_U@○○○  
(自動入力)

⑤ をクリックします。

(または[Alt + F8]をタッチします)



#### 接続モード

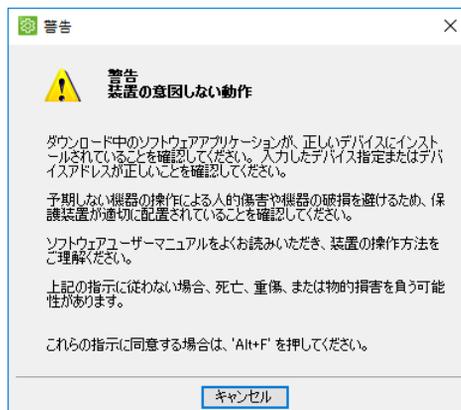
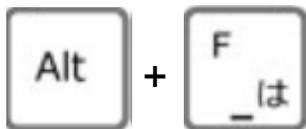
PLCとログイン(通信)するためには、接続モード(方法)を指定する必要があります。接続モードをノード名にすると右側にノード名が表示されます。

#### 接続モード

- ・ノード名：ノード名は、PLC固有のもので、USB接続/イーサネット接続時のどちらでも使用することができます。
- ・IPアドレス：イーサネット接続時に使用することができます。あらかじめ、IPアドレスを設定しておく必要があります。



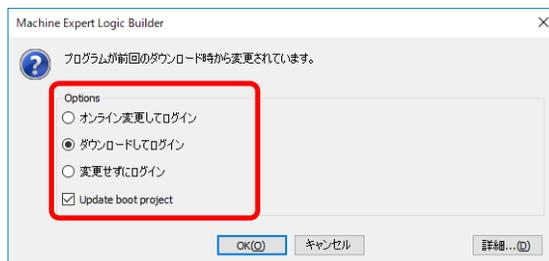
⑥ [警告]が表示されたら、[Alt+F]をタッチします。



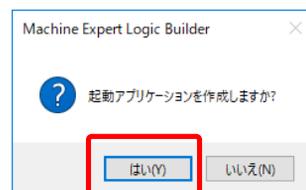
### オンラインチェンジ = RUN中書込み

運転(RUN)中にプログラムの変更を行う場合は、一度ログアウトされた後に変更部分を一括書込みします。その際下記のダイアログボックスが表示されます。一般的には安全面を考慮して、[ダウンロードしてログイン]を選択します。

- ・(プログラムを)オンラインチェンジして、ログインします。  
 > 運転は停止せず、そのままRUNします。
- ・(プログラムを)ダウンロードしてログインします。  
 > 運転は停止します。
- ・(プログラムを)変更せずにログインします。  
 > プログラムのダウンロードは行わず、ログインのみ行います。  
 > 運転は停止せず、そのままRUNします。



右記のダイアログが表示された場合は [OK]をクリックします。

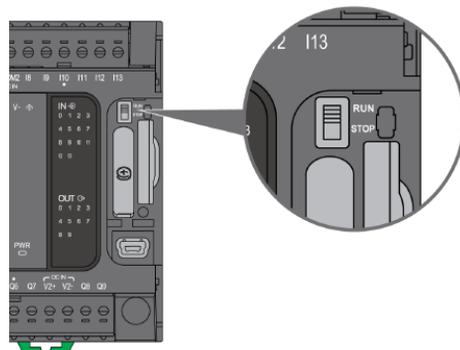


[ポスト設定警告]で[OK]をクリックします。



### M241本体でプログラムをRUNさせるには

M241には運転/ 停止ハードウェアスイッチが備っており、コントローラーを運転または停止状態にすることができます。



### (3) 運転

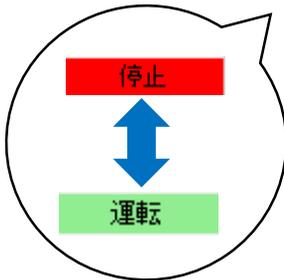
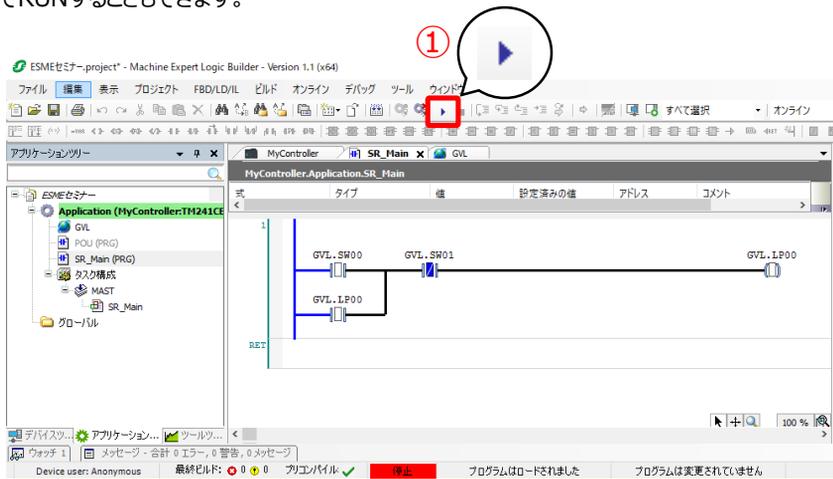
ログインしたら、プログラムをRUNします。なお、PLC本体でRUNすることもできます。

①  をクリックします(またはF5をタッチします)



②  をクリックすると運転を停止します。

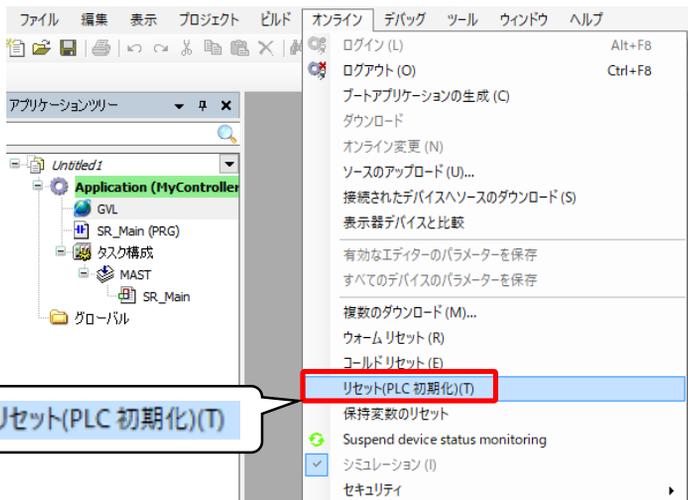
※運転/停止は画面下部で確認します。



CHECK!

#### PLCの初期化

PLCの状態(変数やデバイスアドレス)を初期状態に戻すには、メニューバー[オンライン][リセット(PLC初期化)]をクリックします。





# 第2章

## 【ラダープログラム】 基本回路をマスターしよう

I. 【やってみよう】自己保持回路	2-1
1. 自己保持回路とは	2-1
2. 【解説】自己保持回路	2-2
3. ビルドから運転まで	2-8

# I 【やってみよう】自己保持回路

## 1. 自己保持回路とは

自己保持回路とは、「入力がONした状態を自ら保つ回路」のことです。要求仕様を参考にして、ラダープログラムを作成しましょう。また、作成後はビジュアライゼーション機能(第5章)でデバッグを行います。

### (1) 要求仕様とプログラム例

下記は、ラダープログラムを使った自己保持回路の作成例です。まずは要求仕様と作成例を参考にプログラミングを行いましょう。

【PLC】  
 シリーズ：Modicon M241  
 機種：TM241CEC24U  
 IPアドレス：192.168.1.20  
 サブネット：255.255.255.0



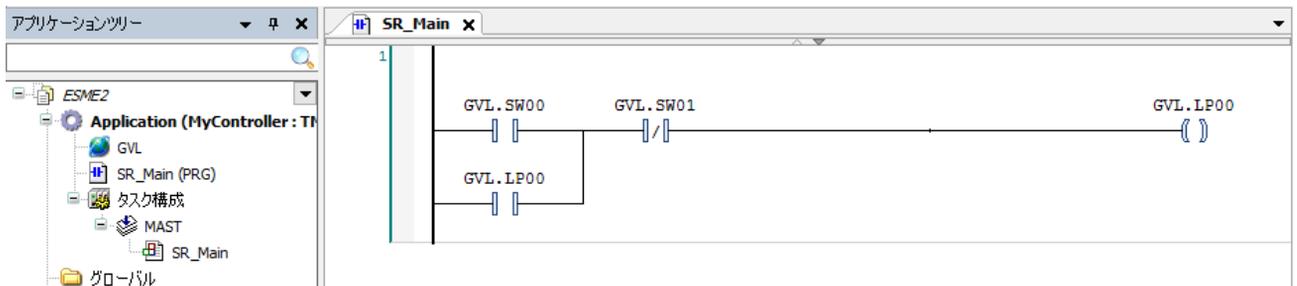
IPアドレス：192.168.1.20  
 サブネット：255.255.255.0

#### 【要求仕様】

下記の2つの要求を満たすラダープログラム作成しましょう(新規プロジェクトで作成してください/完成例を参考にしてください)

- ① GVL.SW00をONすると、コイルが出力します(GVL.LP00)  
 (GVL.SW00をOFFしても、GVL.LP00はONした状態を保持します)
- ② GVL.SW01をONすると、GVL.LP00はOFFします。

#### 【ラダープログラム作成例】



#### 【使用する命令】



a接点



b接点

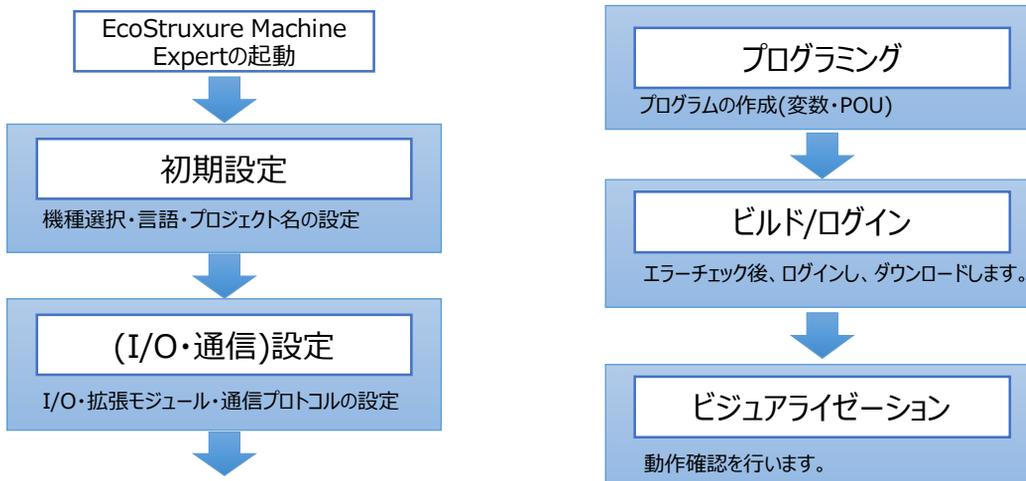


コイル出力

#### 【使用する変数】

変数名	データ型
GVL.SW00	BOOL
GVL.SW01	BOOL
GVL.LP00	BOOL

### (2) プロジェクト作成の流れ



## 2. 【解説】自己保持回路

### (1) 新規プロジェクトの作成

新規プロジェクトの作成は、最初にハードウェア側の設定(PLCの指定・拡張モジュール・通信プロトコル)を行い、その後[プログラミング]を行います。ここでは、以下の環境を例に新規プロジェクトを作成します。

#### 【PLC】

シリーズ：Modicon M241  
機種：TM241CEC24U  
IPアドレス：192.168.1.20  
サブネット：255.255.255.0

#### 【EcoStruxure Machine Expertの起動】

①デスクトップ上の[Machine Expert]のアイコンをダブルクリックします。



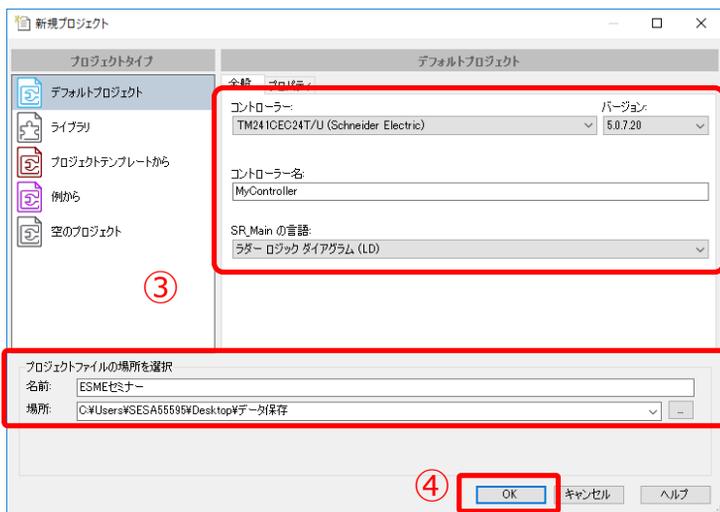
②  をクリックします。



③[新規プロジェクト]で下記の設定を行います。

コントローラ：TM241CEC24U  
コントローラ名：MyController(デフォルト)  
SR-Mainの言語：ラダーロジックダイアグラム

プロジェクトファイル  
名前：ESMEセミナー(任意)  
場所：C:\Users\〇〇〇\Desktop\データ保存(任意)



④[OK]をクリックします。

#### CHECK!

#### よいプログラムの作り方

よいプログラムとは、プログラムの作成者以外が内容を容易に把握できる見やすいプログラムのことです。  
(機能追加や改修時に現状の理解が早くなる、作業効率が上がります)



##### ①コメントを入れる。

EcoStruxure Machine Expertでは、POUにコメントを入れることができます。各POUの動作をコメントにすると、トラブル発生時や変更時に現状理解を素早く行うことができます。

##### ②ファンクションブロックを使って回路を分ける。

同様の処理を繰り返す場合はファンクションブロックとしてモジュール化すると、読みやすくなり、変更時のメンテナンスも楽になります。また、必然的にメインプログラムは小さくなり、スキャンタイムもアップします。

##### ③プログラムの配置を標準化する。

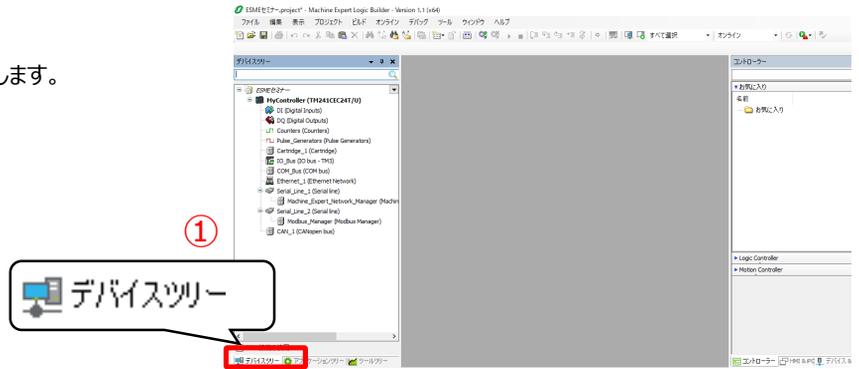
回路ごとに、プログラムの配置を標準化します。例えば、現位置確認を最初に配置し、運転条件確認・リセット回路・手動操作回路・自動運転回路・出力回路と続け最後に表示器制御回路をまとめる。こうすればプログラムのどこに目的の回路があるのかが判りやすくなります。変数名もルール付けしておくとうわりやすいです。

## (2)デバイスツリー(IPアドレスの設定)

ここでは、M241本体のIPアドレスの設定を行います。

### 【IPアドレスの設定】

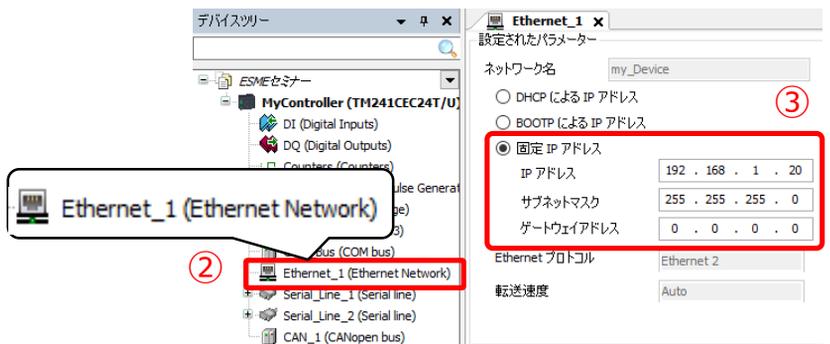
①[ナビゲーター]の左下の[デバイスツリー]をクリックします。



②デバイスツリーから[Ethernet1]をクリックします。

③M241自身のIPアドレスを下記のように指定します。

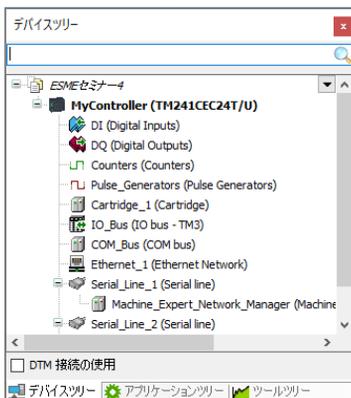
固定IPアドレス  
IPアドレス：192.168.1.20  
サブネットマスク：255.255.255.0



### ナビゲーター

ナビゲーターとはプロジェクトを作成するための開始点です。[デバイスツリー][アプリケーションツリー][ツールツリー]の3つから構成されています。

#### 【デバイスツリー】



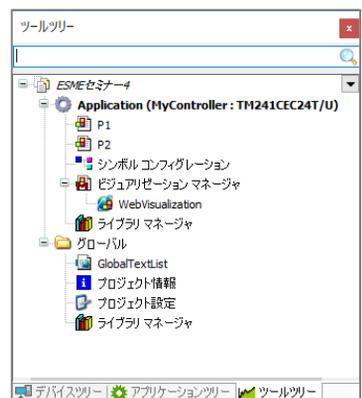
I/O設定や拡張モジュール、通信に関する設定を行います。

#### 【アプリケーションツリー】



変数の作成、プログラミングを行います。

#### 【ツールツリー】

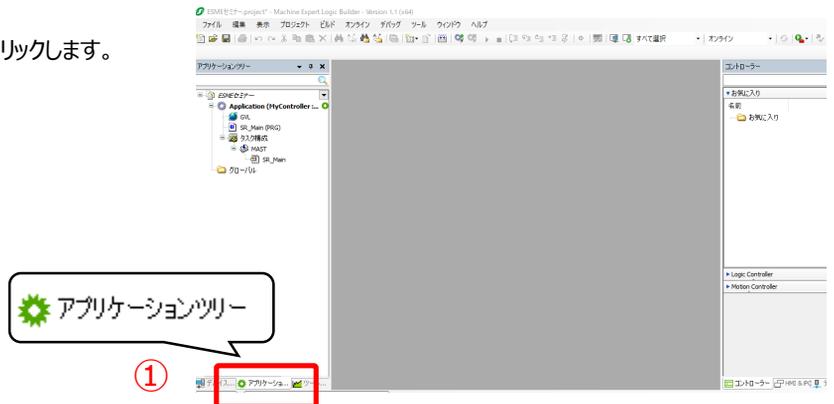


ビジュアルリゼーション機能等の設定を行います。

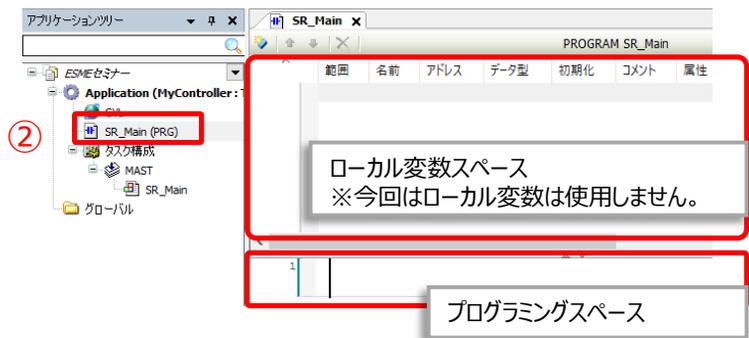
### (3)プログラミング(自己保持回路)

ラダープログラムで自己保持回路を作成します。

①[ナビゲーター]の左下の[アプリケーションツリー]をクリックします。



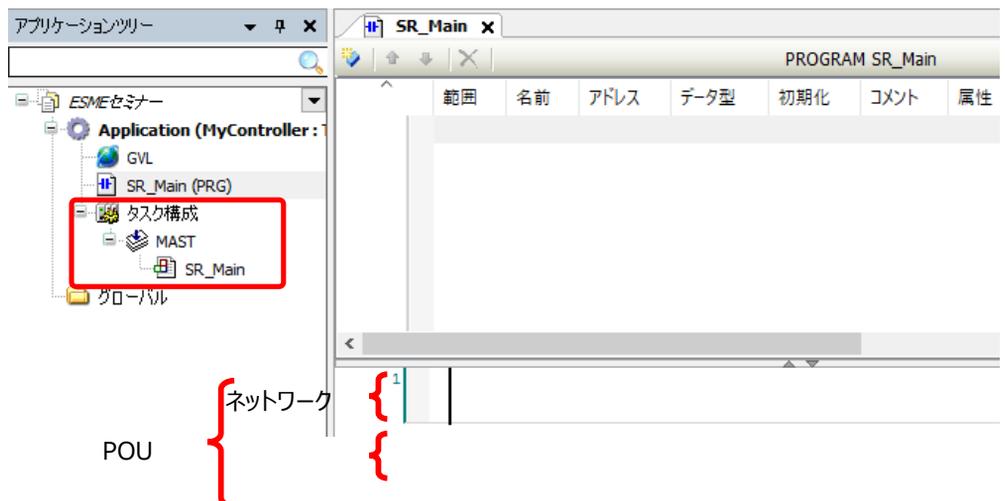
②[SR\_Main]をクリックします。



#### SR\_Main(POU)

[SR\_Main]とは、あらかじめ用意されているPOUです。初期状態でタスク(MAST)に登録されているので、プログラムを作成すれば、スキャンごとに実行されます。

- ・POU (Program Organization Unit) : プログラムの実行単位です。POU内はネットワーク単位で区分されています。
- ・ネットワーク : 入力から出力で構成される命令の集まり(行)です。



【a接点の配置とグローバル変数の割り付け】

① [ツールボックス]の[a接点]をプログラミングスペースにドラッグ&ドロップで配置します。



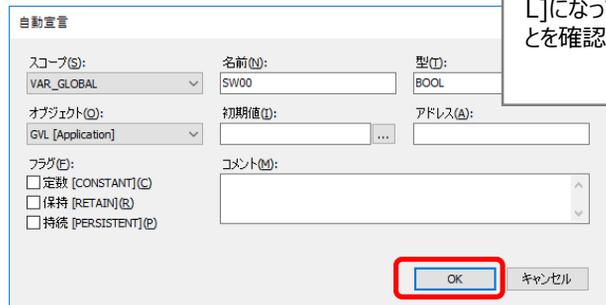
② 配置した[a接点]の[???]に[GVL.SW00]と入力します。

GVL.SW00



③ [自動宣言] で[OK]をクリックします。

スコープ : VAR\_GLOBAL  
名前 : SW00  
型 : BOOL  
オブジェクト : GVL[Application]



**CHECK!**  
スコープが [VAR\_GLOBAL] になっていることを確認しましょう。

③



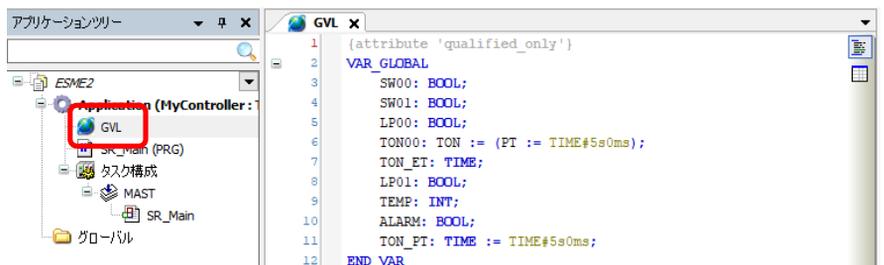
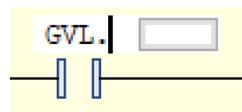
グローバル変数

グローバル変数とはプロジェクト上のすべてのPOUで、共有して使用することができる変数です。変数作成時に[GVL.]と入力します。アプリケーションツリーの[GVL]で確認できます(GVL変数リスト)

また、実I/O上にマッピングすることも可能です。

グローバル変数は[GVL変数リスト]で確認できます。また、[GVL変数リスト]では確認だけでなく、変数の作成もできます。

グローバル変数は変数名に[GVL.]と入力します。



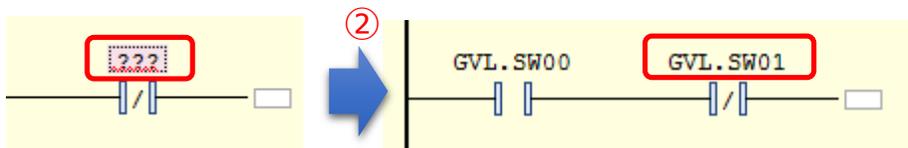
【b接点の配置とグローバル変数の割り付け】

①[ツールボックス]の[b接点]をプログラミングスペースにドラッグ&ドロップで配置します。



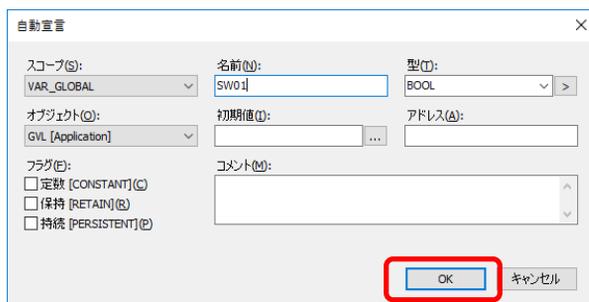
②配置した[b接点]の[???]に[GVL.SW01]と入力します。

GVL.SW01



③[自動宣言]で[OK]をクリックします。

スコープ : VAR\_GLOBAL  
名前 : SW01  
型 : BOOL  
オブジェクト : GVL[Application]



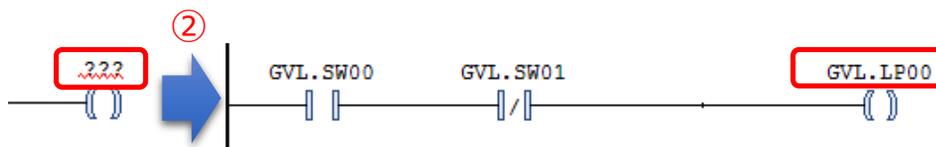
【コイルの配置とグローバル変数の割り付け】

①[ツールボックス]の[コイル]をプログラミングスペースにドラッグ&ドロップで配置します。



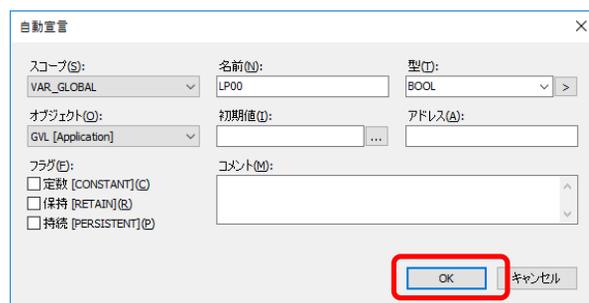
②配置した[コイル]の[???]に[GVL.LP00]と入力します。

GVL.LP00



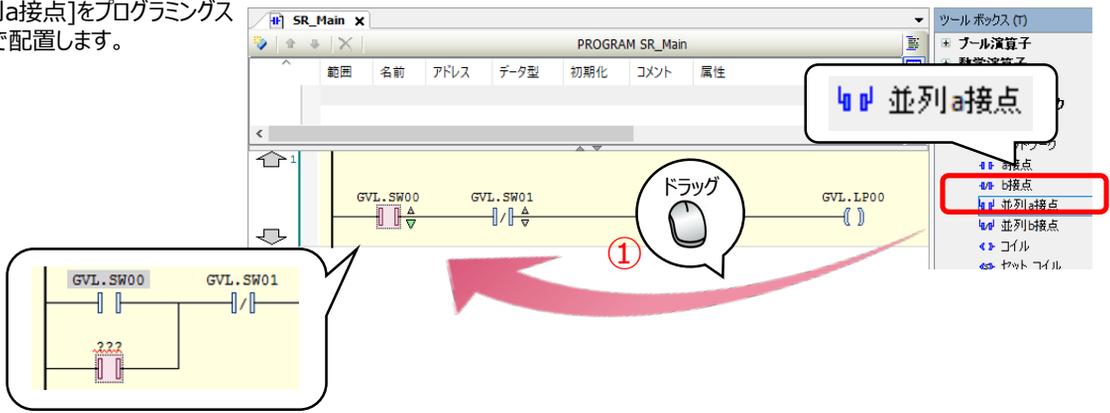
③[自動宣言]で[OK]をクリックします。

スコープ : VAR\_GLOBAL  
名前 : LP00  
型 : BOOL  
オブジェクト : GVL[Application]



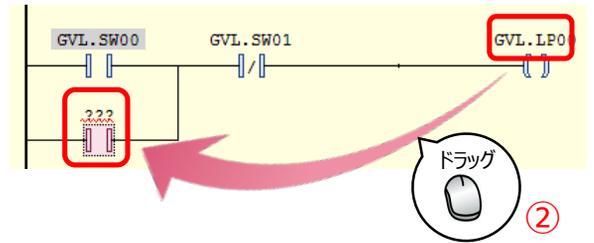
【分岐の挿入】

① [ツールボックス]の[並列a接点]をプログラミングスペースにドラッグ&ドロップで配置します。

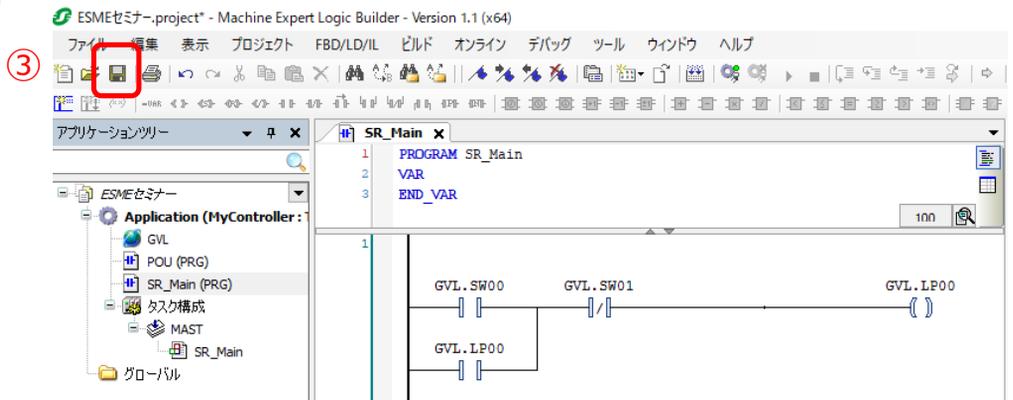


② 分岐上のa接点に[GVL.LP00]を割り当てます。  
※コイルの割り当てた[GVL.LP00]をドラッグ&ドロップします。

GVL.LP00

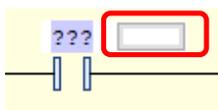


③ プログラムが完成したら  をクリックし、上書き保存します。



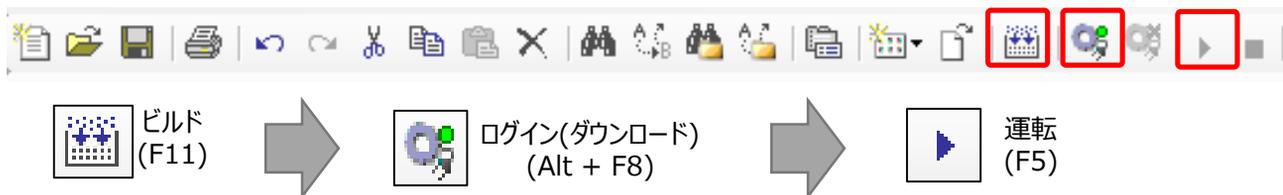
変数の割り付け方

作成済みの変数を割り付けるには、変数名の右に表示される  をクリックするか、または[F2キー]をタッチします。[入力アシスタント]を使用して変数を割り付けることができます。



### 3. ビルドから運転まで

ここではプログラム作成後の手順について説明します。  
プログラム作成後はビルドを行い、PLC本体にログインし、プログラムを書き込みます。

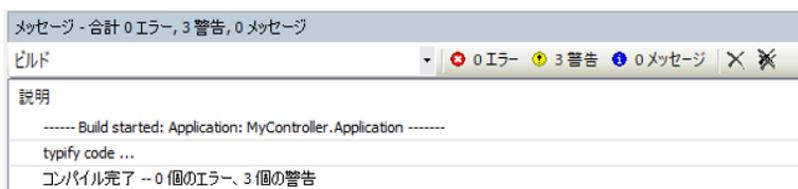


#### (1)ビルド(エラーチェック)

プログラム作成後はビルドで、エラーチェックを行います。

エラー：ダウンロードを行うことができません。

警告：ダウンロードを行うことができます(警告内容は必ず確認してください)



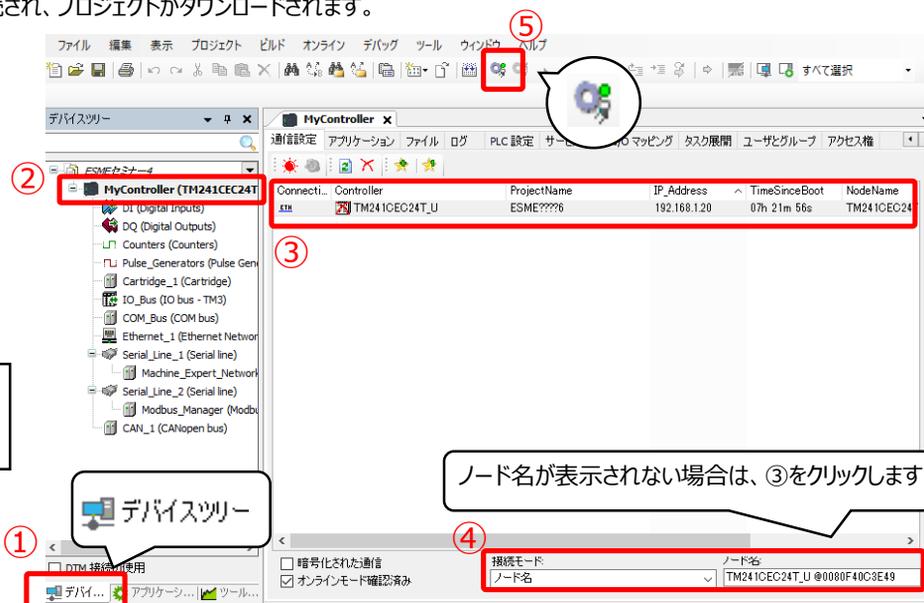
#### (2)ログイン

ログインを行うことで、PLC本体と論理的に接続され、プロジェクトがダウンロードされます。

- ① [デバイスツリー] をクリックします。
- ② [MyController] をクリックします。
- ③ 接続するPLC をクリックします。
- ④ 接続モードを指定します。

接続モード：ノード名  
ノード名：TM241CEC24T\_U@○○○  
(自動入力)

- ⑤ をクリックします。  
(または[Alt + F8]をタッチします)



CHECK!

#### 接続モード

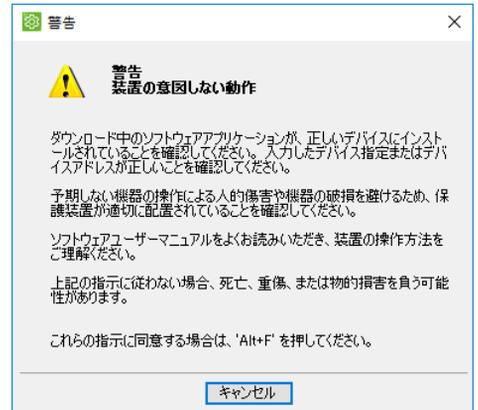
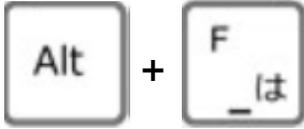
PLCとログイン(通信)するためには、接続モード(方法)を指定する必要があります。接続モードをノード名にすると右側にノード名が表示されます。

#### 接続モード

ノード名：ノード名は、PLC固有のもので、USB接続、イーサネット接続時のどちらでも使用することができます。  
IPアドレス：イーサネット接続時に使用することができます。あらかじめ、IPアドレスを設定しておく必要があります。

接続モード: ノード名  
ノード名: TM241CEC24T\_U@0080F40C3E45

⑥ [警告]が表示されたら、[Alt+F]をタッチします。

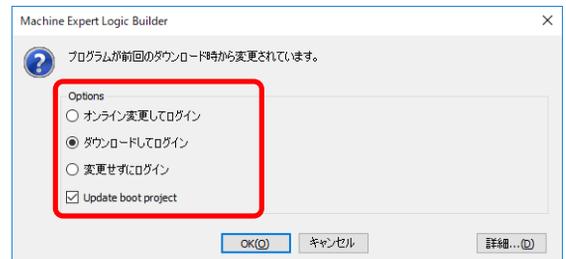


CHECK!

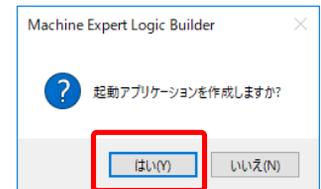
### オンラインチェンジ = RUN中書込み

運転(RUN)中にプログラムの変更を行う場合は、一度ログアウトされた後に変更部分を一括書込みします。その際下記のダイアログボックスが表示されます。一般的には安全面を考慮して、[ダウンロードしてログイン]を選択します。

- ・(プログラムを)オンラインチェンジして、ログインします。  
> 運転は停止せず、そのままRUNします。
- ・(プログラムを)ダウンロードしてログインします。  
> 運転は停止します。
- ・(プログラムを)変更せずにログインします。  
> プログラムのダウンロードは行わず、ログインのみ行います。  
> 運転は停止せず、そのままRUNします。



右記のダイアログが表示された場合は [OK]をクリックします。

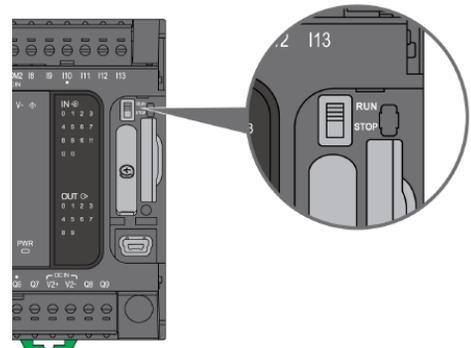
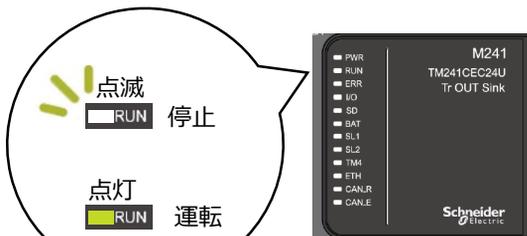


[ポスト設定警告]で[OK]をクリックします。



### M241本体でプログラムをRUNさせるには

M241には運転/ 停止ハードウェアスイッチが備っており、コントローラーを運転または停止状態にすることができます。



### (3) 運転

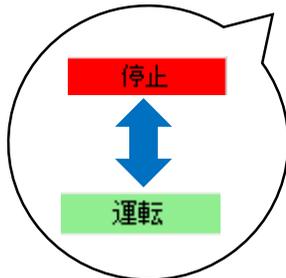
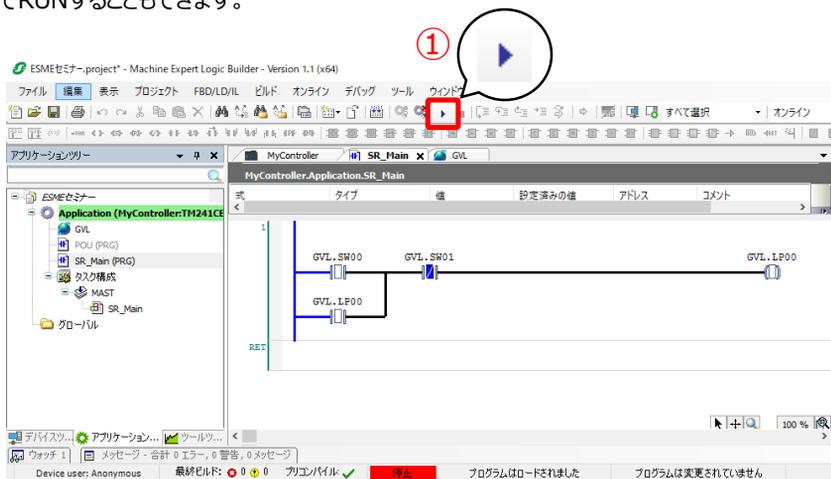
ログインしたら、プログラムをRUNします。なお、PLC本体でRUNすることもできます。

①  をクリックします(またはF5をタッチします)



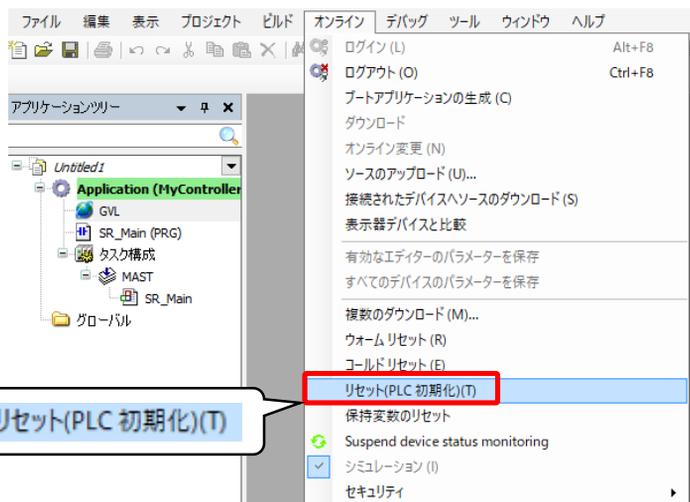
②  をクリックすると運転を停止します。

※運転/停止は画面下部で確認します。



#### PLCの初期化

PLCの状態(変数やデバイスアドレス)を初期状態に戻すには、メニューバー[オンライン][リセット(PLC初期化)]をクリックします。



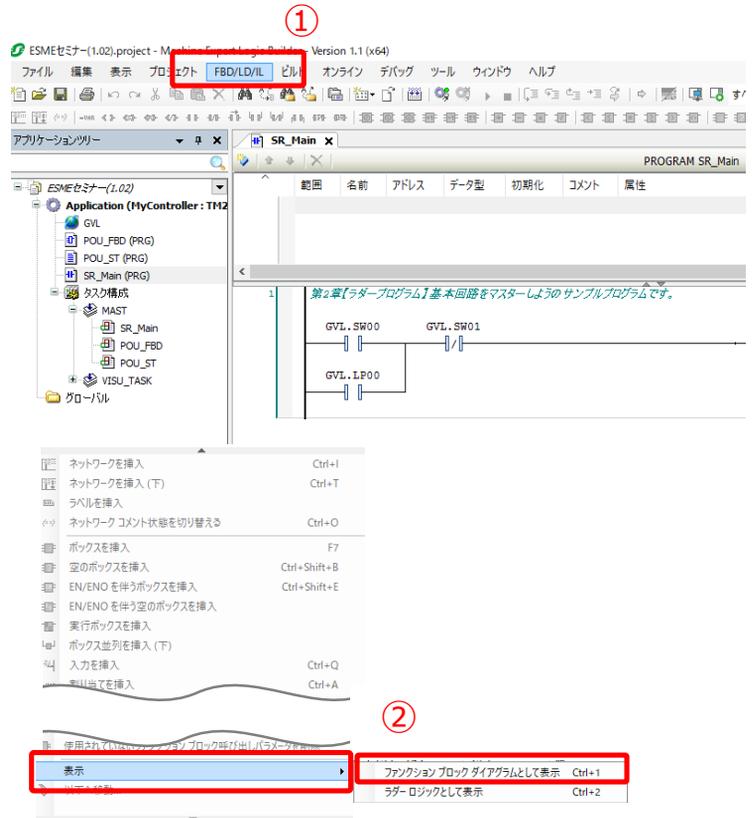
CHECK!

## LDとFBDの切り替え表示

LD(ラダープログラム)とFBD(ファンクションブロックダイアグラム)は切り替えることができます。  
※ここではLDからFBDに切り替え表示します。

①メニューバー[FBD/LD/IL]をクリックします。  
を選択します。

② [表示]-[ファンクションブロックダイアグラムとして表示]をクリックします。  
※メニューの最下部にあります。



LD(ラダープログラム)からFBD(ファンクションブロックダイアグラム)に切り替わりました。



# 第3章

## 【FBD】

### 基本回路をマスターしよう

I. 【やってみよう】タイマー回路	3-1
1. タイマー回路とは	3-1
2. 【解説】タイマー回路	3-3
3. ビルドから運転まで	3-8

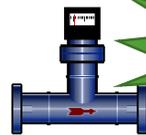
# I 【やってみよう】タイマー回路

## 1. タイマー回路とは

ここでは、EcoStruxure Machine Expertのファンクションブロックダイアグラム(FBD)を使った「タイマー」回路の実習を行います。要求仕様を参考にして、プログラムを作成しましょう。また、作成後はビジュアルリゼーション機能(第5章)でデバッグを行います。

タイマー回路とは、設定時間後に接点が動作する回路のことです。ここでは一定時間遅れてONするオンディレイタイマーを使用します。

例)機器の動作時間の監視  
一定時間以上ONしていたら警報を出すなど



バルブが開  
きっぱなし

### (1)要求仕様とプログラム例

下記は、FBDを使ったオンディレイタイマーの作成例です。まずは要求仕様と作成例を参考にプログラミングを行いましょう。

#### 【要求仕様】

下記の要求を満たすファンクションブロックダイアグラムを作成しましょう。  
(第2章で作成したプロジェクトを継続して使用します/完成例を参考にしてください)

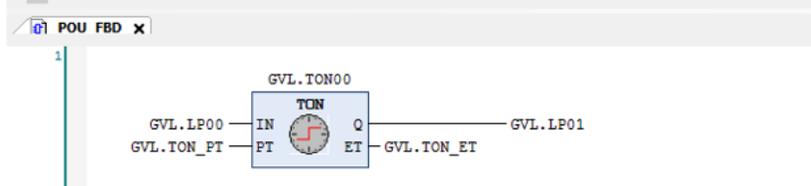
#### 【第1章で作成したラダープログラム】(作成済み)

①GVL.SW00をONすると、コイルが出力します  
(GVL.LP00)

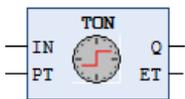


#### 【プログラム作成例】

②GVL.LP00がONすると、5秒間遅れて  
GVL.LP01がONします。



#### 【使用する命令】



オンディレイタイマー

**VAR** 割り当て

割り当て

#### 【使用する変数】

変数名	データ型
GVL.LP00	BOOL
GVL.LP01	BOOL
GVL.TON00	TIME
GVL.TON_PT	INT
GVL.TON_ET	INT

### CHECK!

#### タイマー変数(構造体変数)

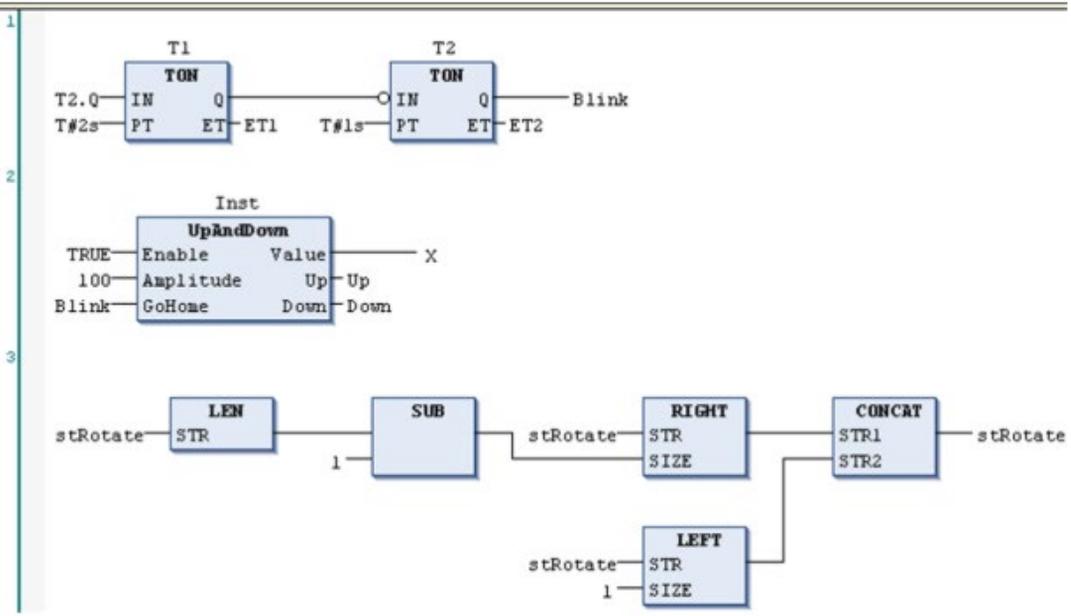
タイマー命令にはタイマー変数を割り付けます。タイマー変数は、それぞれの役割を持った要素変数の集まり(構造体変数)です。要素変数の役割は下記になります。

要素変数名	タイプ	役割
○○.IN	BOOL	タイマー計測時にON(タイマーアドレスに信号が導通するとONします)
○○.Q	BOOL	タイマー計測完了時にON(計測が完了するとONします)
○○.PT	INT	タイマーの設定値
○○.ET	INT	タイマーの現在値



### FBD(ファンクションブロックダイアグラム)

FBD言語は、電子部品(ファンクションと呼ばれる箱)をそれらを接続する配線で構成されており、電子回路を設計するようにプログラムを作成することができます。そのため、[データの流れがわかりやすい]というメリットがあります。ファンクションの左側にある変数は入力値で右側の変数が演算結果である出力値になります。

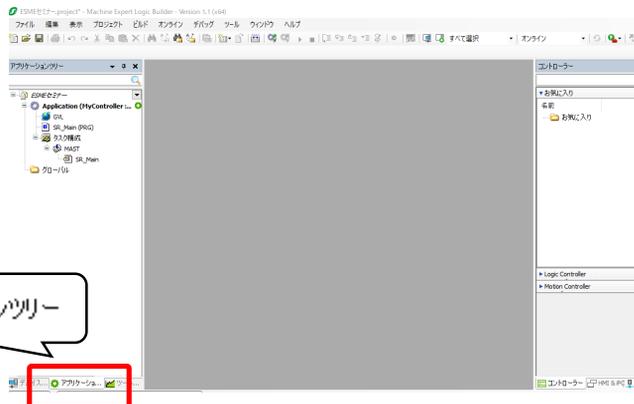


## 2. 【解説】タイマー回路

### (1)POUの追加

プログラミングはPOU単位で作成するため、最初にFBD用のPOUを作成します。

①[ナビゲーター]の左下の[アプリケーションツリー]をクリックします。



②[Application]を選択し、+ をクリックします。

③[POU]をクリックします。



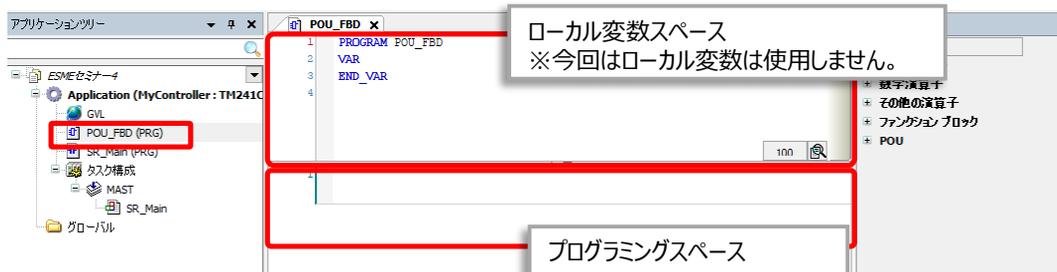
④[POUの追加]でPOUの名称や記述言語を指定します。

【POUの追加】  
名前： POU\_FBD  
型：プログラム  
記述言語：ファンクションブロックダイアグラム(FBD)



ファンクションブロックダイアグラム(FBD)

⑤[追加]をクリックすると、アプリケーションツリーにPOUが表示されます。

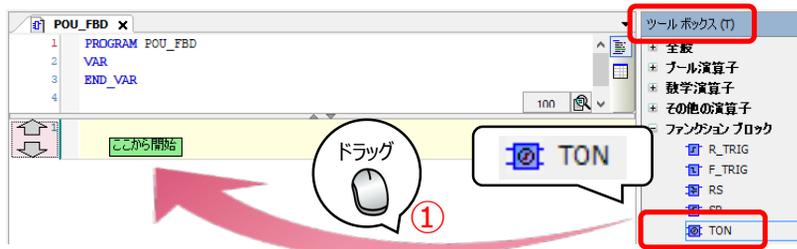


## (2)プログラミング(オンディレイタイマー)

FBDでオンディレイタイマー回路を作成します。

### 【タイマー命令の配置】

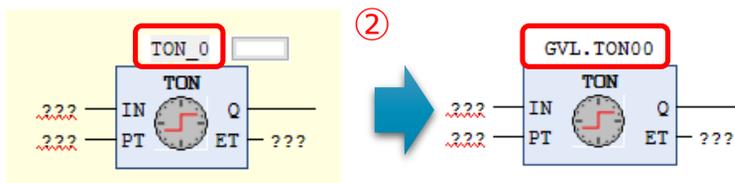
① [ツールボックス]の[a接点]をプログラミングスペースにドラッグ&ドロップで配置します。



### 【グローバル変数と作成と割り付け】

タイマー命令に変数を割り付けます(タイマー変数)

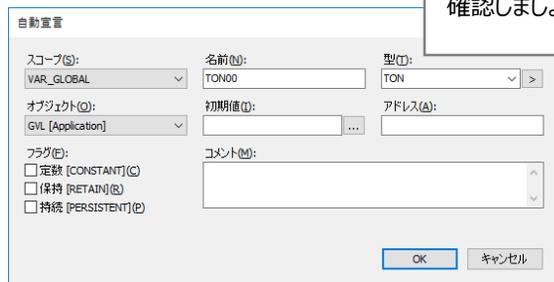
② 配置した[オンディレイタイマー]の[TON\_0]を削除し、[GVL.TON00]と入力します。



GVL.TON00

③ [自動宣言] で[OK]をクリックします。

スコープ : VAR\_GLOBAL  
名前 : TON00  
型 : TON  
オブジェクト : GVL[Application]



**CHECK!**

型が[TON]になっていることを確認しましょう。

### 【グローバル変数の割り付け(IN)】

[.IN]に [GVL.LP00](作成済み)を割り付けます。

①  をクリックします。

GVL.LP00



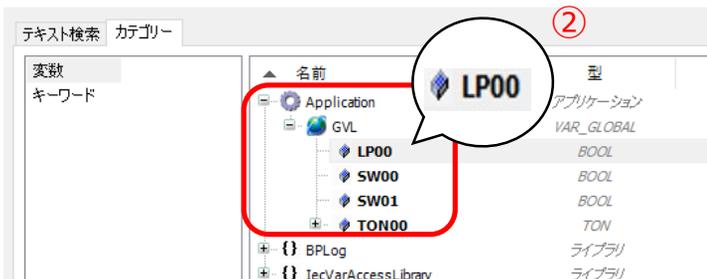
② [入力アシスタント]から [Application][GVL][LP00]をクリックします。

**CHECK!**

入力アシスタントはF2キーをタッチしても表示されます。

**F2**

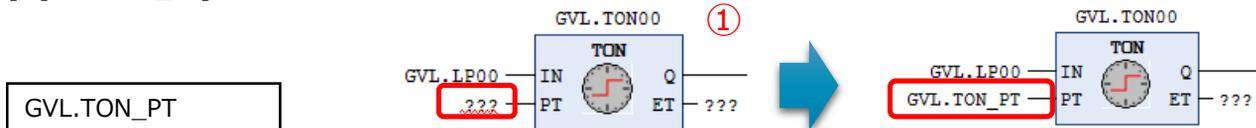
入力アシスタント



【グローバル変数の作成と割り付け(PT)】

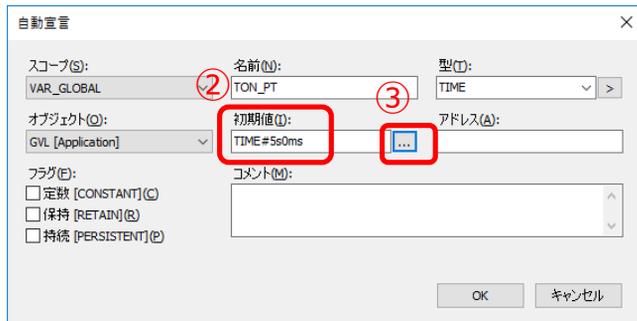
[.PT]とはタイマー設定値です。[.PT]にグローバル変数[GVL.TON\_PT]を作成し、割り付けます。また、初期値として[5000ms]を設定します。

① [???]に[GVL.TON\_PT]と入力します。



② [自動宣言]で「初期値」の設定を行います。  
※初期値は [...] から設定すると簡単です。

初期値 : TIME#5000ms  
(TIME#5s0ms)



③ [...] をクリックします。

④ [初期値]で[TON\_PT]を選択します。

⑤ リストに初期値を入力します。

初期値 : TIME#5000ms  
(TIME#5s0ms)



⑥ [値を選択行に適用]をクリックします。

⑦ [OK]をクリックします。

CHECK!

初期値の設定方法

変数の初期値は、変数リストに直接、入力することもできます。特に表形式は初期値の入力欄があるため、簡単です。

初期値設定例

- BOOL変数 : TRUE or FALSE
- INT変数 : 整数
- TIME変数 : T#5S (5秒)

【テキスト形式】

```

1 {attribute 'qualified_only'}
2 VAR_GLOBAL
3 SW00: BOOL;
4 SW01: BOOL;
5 LP00: BOOL := TRUE;
6 TON00: TON;
7 TON_ET: TIME;
8 TON_PT: TIME := T#5S;
9 LP01: BOOL;
10 TEMP: INT := 50;
11 ALARM: BOOL;
12 END_VAR
    
```

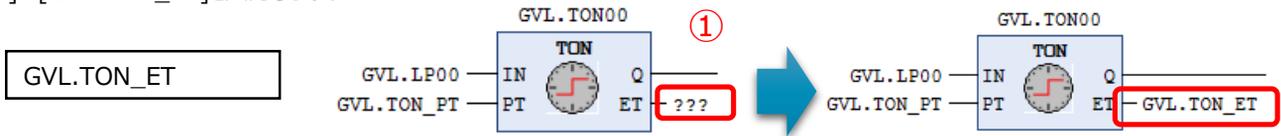
【表形式】

範囲	名前	アドレス	データ型	初期化	コメント
1	VAR_GLOBAL SW00		BOOL		
2	VAR_GLOBAL SW01		BOOL		
3	VAR_GLOBAL LP00		BOOL	TRUE	
4	VAR_GLOBAL TON00		TON		
5	VAR_GLOBAL TON_ET		TIME		
6	VAR_GLOBAL TON_PT		TIME	T#5S	
7	VAR_GLOBAL LP01		BOOL		
8	VAR_GLOBAL TEMP		INT	50	
9	VAR_GLOBAL ALARM		BOOL		

### 【グローバル変数の作成と割り付け(ET)】

[.ET]とはタイマー現在値です。[.ET]にグローバル変数[GVL.TON\_ET]を作成し、割り付けます。

① [???]に[GVL.TON\_ET]と入力します。



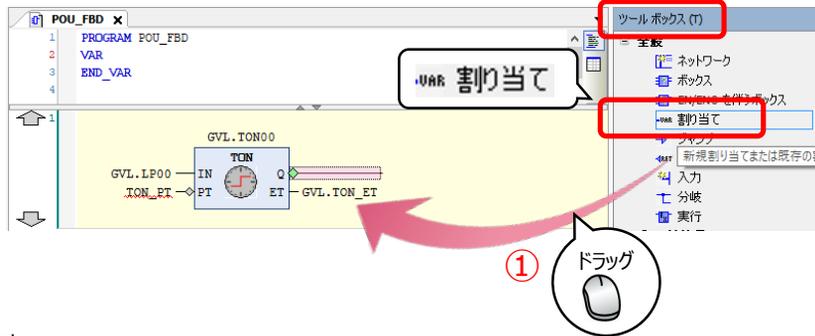
② [自動宣言]で[OK]をクリックします。

スコープ : VAR\_GLOBAL  
 名前 : TON\_ET  
 型 : TIME  
 オブジェクト : GVL[Application]

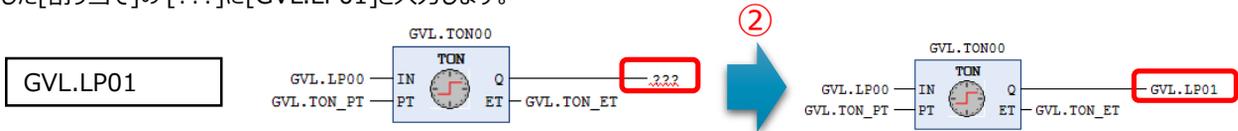
**CHECK!** 型が[TIME]になっていることを確認しましょう。

### 【[割り当て]命令の配置】

① [ツールボックス]の[割り当て]をプログラミングスペースにドラッグ&ドロップで配置します。



② 配置した[割り当て]の [???]に[GVL.LP01]と入力します。



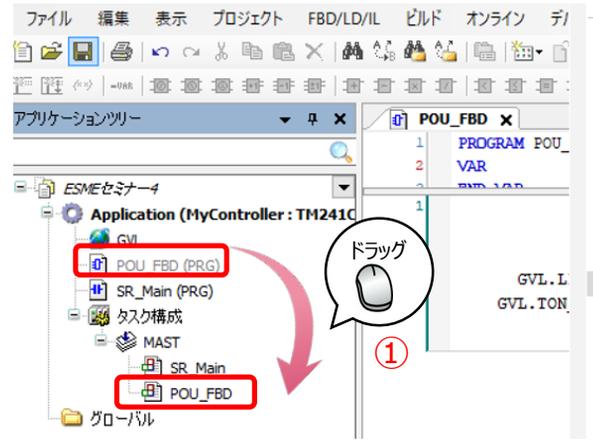
③ [自動宣言]で[OK]をクリックします。

スコープ : VAR\_GLOBAL  
 名前 : LP01  
 型 : BOOL  
 オブジェクト : GVL[Application]

【POUの移動】

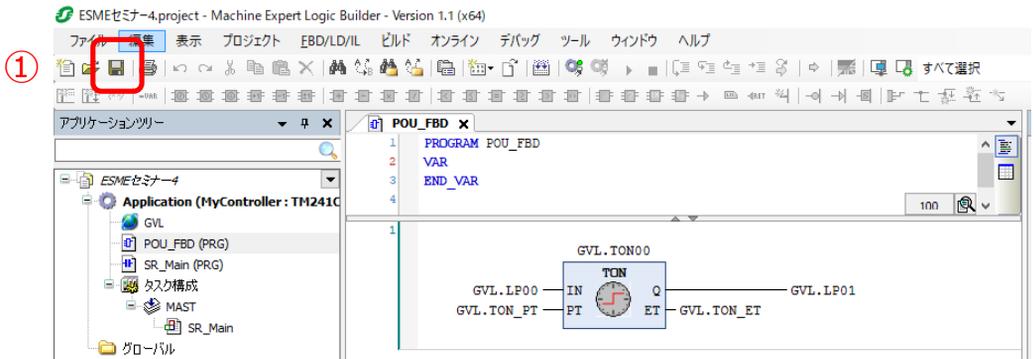
作成した[POU\_FBD]をタスク(Mast)に移動します。  
 ※タスクに移動しないと動作しません。

① [POU\_FBD(PRG)]をタスクに移動します。



【プロジェクトの保存】

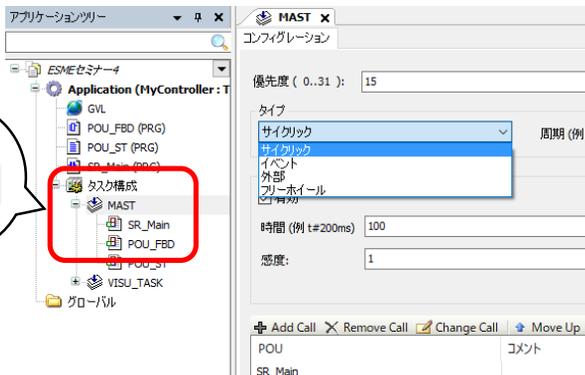
① プログラムが完成したら  をクリックし、上書き保存します。



CHECK!

タスク

POUはタスクに登録することで実行されます。作成したPOUはドラッグ&ドロップで、タスク内に移動します。  
 タスクには実行条件の違う4つが用意されています。MASTをクリックすると、コンフィグレーションで確認することができます。



- ・サイクリック： 設定周期毎に実行されます。
- ・イベント： データ変化などイベント発生時に実行されます。
- ・外部： 通信などの外部イベント発生時に実行されます。
- ・フリーホイール： 各処理の完了毎に実行されます。

### 3. ビルドから運転まで

ここではプログラム作成後の手順について説明します。  
プログラム作成後はビルドを行い、PLC本体にログインし、プログラムを書き込みます。

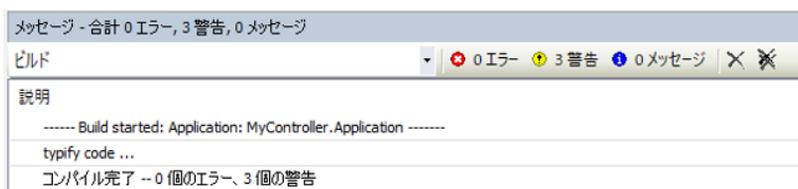


#### (1)ビルド(エラーチェック)

プログラム作成後はビルドで、エラーチェックを行います。

エラー：ダウンロードを行うことができません。

警告：ダウンロードを行うことができます(警告内容は必ず確認してください)



#### (2)ログイン

ログインを行うことで、PLC本体と論理的に接続され、プロジェクトがダウンロードされます。

① [デバイスツリー] をクリックします。

② [MyController] をクリックします。

③ 接続するPLCをクリックします。

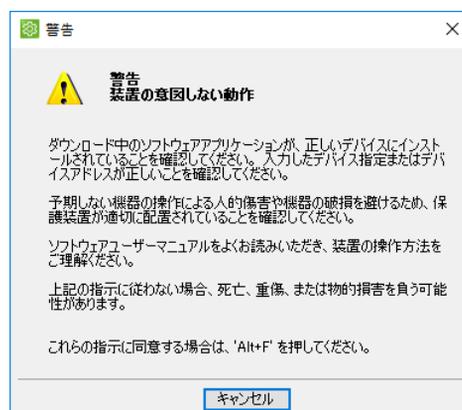
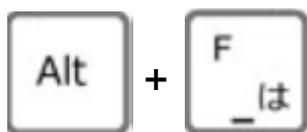
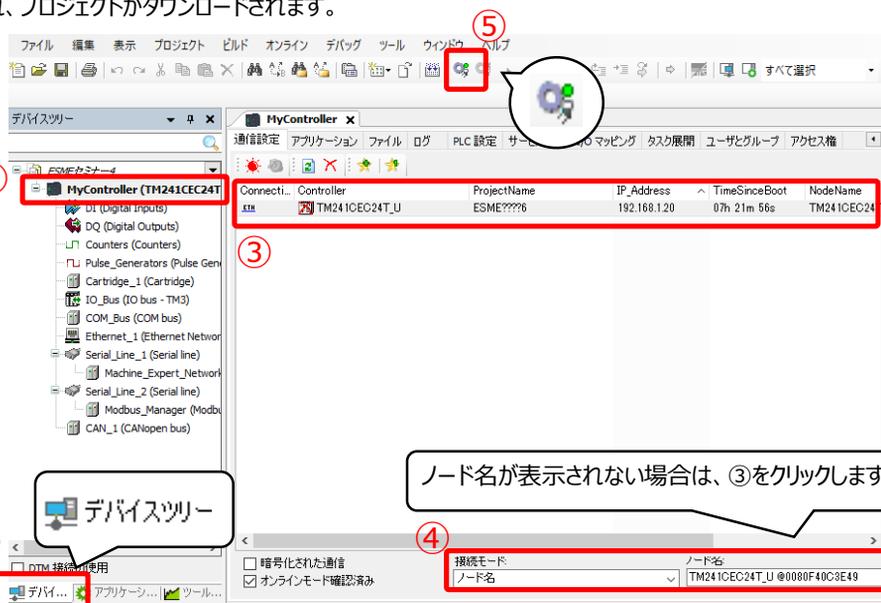
④ 接続モードを指定します。

接続モード：ノード名  
ノード名：TM241CEC24T\_U@○○○  
(自動入力)

⑤ をクリックします。

(または[Alt + F8]をタッチします)

⑥ [警告]が表示されたら、[Alt+F]をタッチします。



### (3) 運転

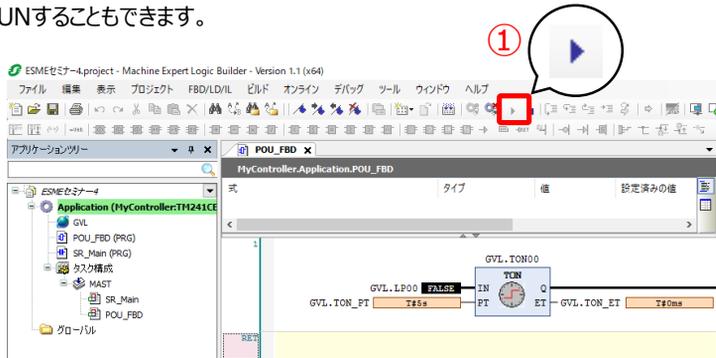
ログインしたら、プログラムをRUNします。なお、PLC本体でRUNすることもできます。

①  をクリックします(またはF5をタッチします)



②  をクリックすると運転を停止します。

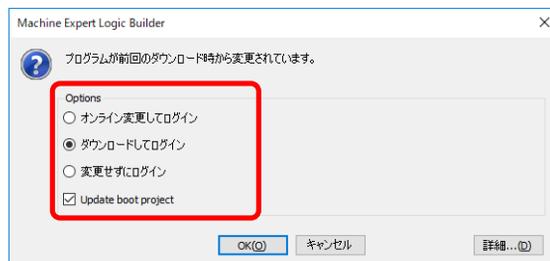
※運転/停止は画面下部で確認します。



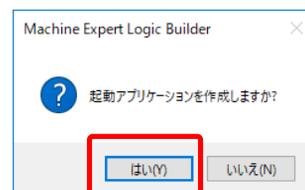
#### オンラインチェンジ = RUN中書込み

運転(RUN)中にプログラムの変更を行う場合は、一度ログアウトされた後に変更部分を一括書込みします。その際下記のダイアログボックスが表示されます。一般的には安全面を考慮して、[ダウンロードしてログイン]を選択します。

- ・(プログラムを)オンラインチェンジして、ログインします。  
 > 運転は停止せず、そのままRUNします。
- ・(プログラムを)ダウンロードしてログインします。  
 > 運転は停止します。
- ・(プログラムを)変更せずにログインします。  
 > プログラムのダウンロードは行わず、ログインのみ行います。  
 > 運転は停止せず、そのままRUNします。



右記のダイアログが表示された場合は [OK] をクリックします。

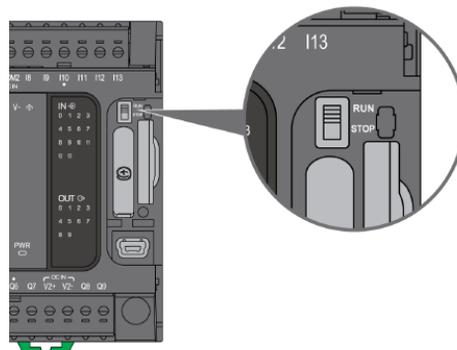
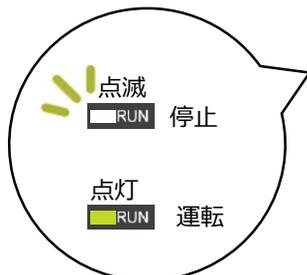


[ポスト設定警告]で[OK]をクリックします。



#### M241本体でプログラムをRUNさせるには

M241には運転/停止ハードウェアスイッチが備っており、コントローラーを運転または停止状態にすることができます。



CHECK!

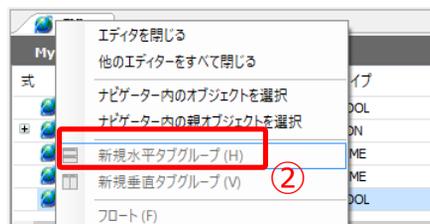
## 画面レイアウトの変更

EcoStruxure Machine Expertは使い勝手に応じて、画面レイアウトを変更することができます。特にワークスペースを水平方法に分割表示すると使いやすくなります。

- ① ワークスペースでタブを右クリックします。
- ② 新規水平タブグループをクリックします。

タブを右クリック

①



②

式	タイプ	値	設定済みの値	アドレス	コメント
SW00	BOOL	FALSE			
SW01	BOOL	FALSE			
LP00	BOOL	FALSE			
TON00	TON				
TON_ET	TIME	T#0ms			

変	タイプ	値	設定済みの値	アドレス	コメント
GVL.LP00	FALSE				
GVL.TON_PT	T#9s				
GVL.TON00	TON				
GVL.TON_ET	T#0ms				

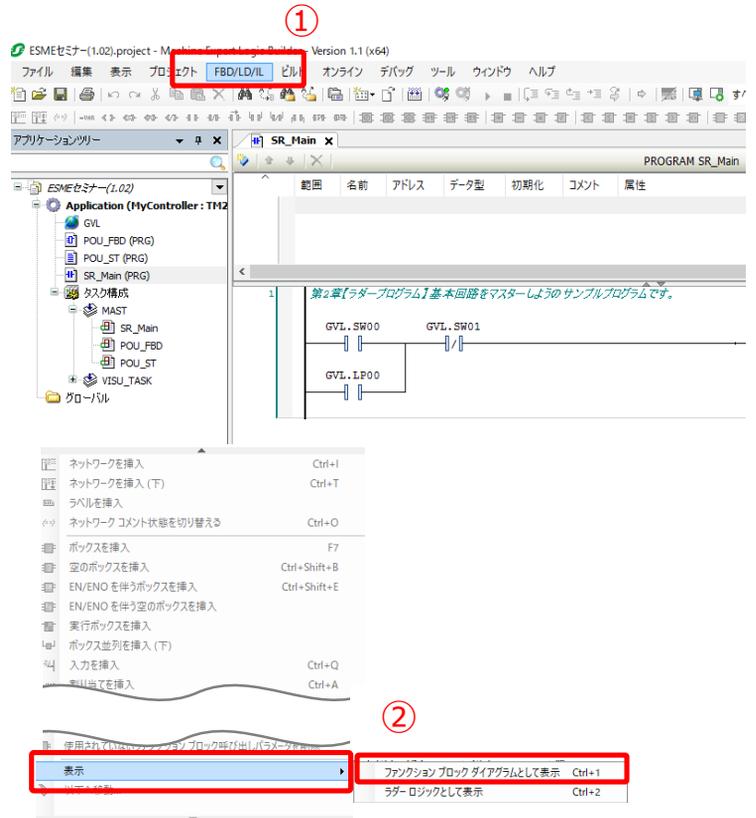

**CHECK!**

### LDとFBDの切り替え表示

LD(ラダープログラム)とFBD(ファンクションブロックダイアグラム)は切り替えることができます。  
※ここではLDからFBDに切り替え表示します。

①メニューバー[FBD/LD/IL]をクリックします。  
を選択します。

② [表示]-[ファンクションブロックダイアグラムとして表示]をクリックします。  
※メニューの最下部にあります。



LD(ラダープログラム)からFBD(ファンクションブロックダイアグラム)に切り替わりました。



# 第4章

【ST】

## 基本回路をマスターしよう

I. 【やってみよう】比較条件回路	4-1
1. タイマー回路とは	4-1
2. 【解説】比較条件回路	4-2
3. ビルドから運転まで	4-5

# I 【やってみよう】比較条件回路

## 1. 比較条件回路とは

ここでは、EcoStruxure Machine Expertのストラクチャードテキスト(ST)を使った「比較条件」回路の実習を行います。要求仕様を参考にして、プログラムを作成しましょう。また、作成後はデバッグ機能(第5章)でデバッグを行います。

比較条件回路とは、複数の変数や定数を比較し、条件を満たせば、決められた動作をする回路です。ここでは変数の値が50を超えるとアラームが発報する回路を作成します。

例)温度や容量の監視  
指定した温度や容量を超えると警報を出すなど



### (1) 要求仕様とプログラム例

下記は、STを使った比較条件回路の作成例です。まずは要求仕様と作成例を参考にプログラミングを行いましょう。

#### 【要求仕様】

下記の要求を満たすストラクチャードテキストを作成しましょう  
(第3章で作成したプロジェクトを継続して使用します。)

TEMP(INT変数)が50を超えると、ALARM(BOOL変数)がONします。

#### 【プログラム作成例】

- ① GVL.TEMPが50を超えると
- ② GVL.ALARMがONします。
- ③ (GVL.TEMPが50以下なら)  
GVL.ALARMがOFFします。

```
IF GVL.TEMP > 50 THEN
  GVL.ALARM := 1;
ELSE
  GVL.ALARM := 0;
END_IF
```

#### 【使用する関数】

IF-ELSE文

```
IF 条件 THEN
  命令1;
ELSE
  命令2;
END_IF
```

#### 【使用する変数】

変数名	データ型
GVL.TEMP	INT
GVL.ALARM	BOOL

CHECK!

#### ストラクチャードテキスト

ST言語の“ST”は“Structured Text (ストラクチャードテキスト)”の略で、日本語では「構造化テキスト」とも呼ばれます。STは、C言語やBASICによく似た記述で、数式表現を多く使う場合やループ処理を行う場合に便利です。

ST言語の代表的な文

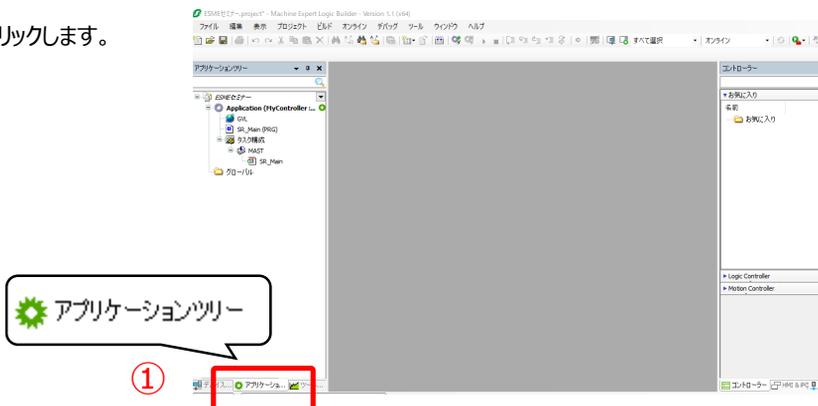
- IF文：条件分岐させたい場合に使用します。
- FOR文：指定した変数が加算し、終端値に達するまで、繰り返し実行します。
- CASE文：ある式の値に応じて、多分岐を行う場合に実行します。
- WHILE文：条件が成立している間、繰り返し実行します。
- REPEAT文：条件が成立するまで、繰り返し実行します。

## II 【解説】比較条件回路

### (1)POUの追加

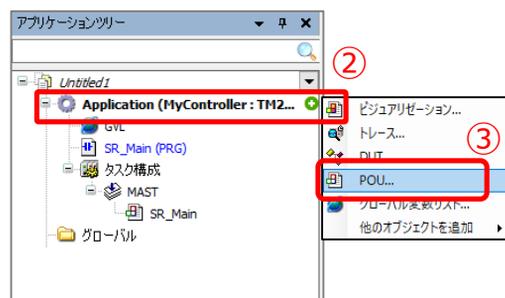
プログラミングはPOU単位で作成するため、最初にST用のPOUを作成します。

①[ナビゲーター]の左下の[アプリケーションツリー]をクリックします。



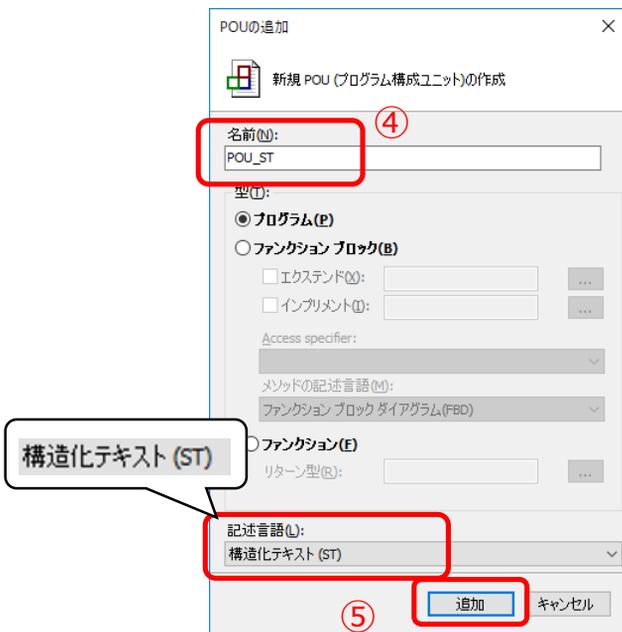
②[Application]を選択し、+ をクリックします。

③[POU]をクリックします。

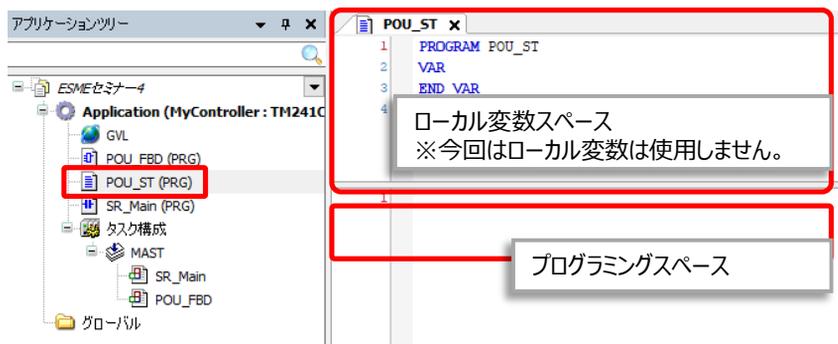


④[POUの追加]でPOUの名称や記述言語を指定します。

【POUの追加】  
名前： POU\_ST  
型：プログラム  
記述言語：構造化テキスト(ST)



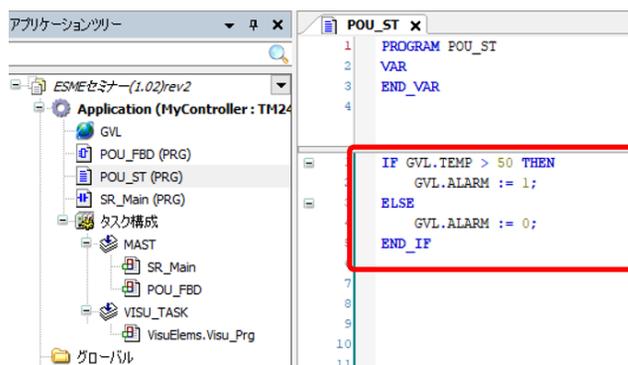
⑤[追加]をクリックすると、アプリケーションツリーにPOUが表示されます。



## (2)プログラミング(オンディレイタイマー)

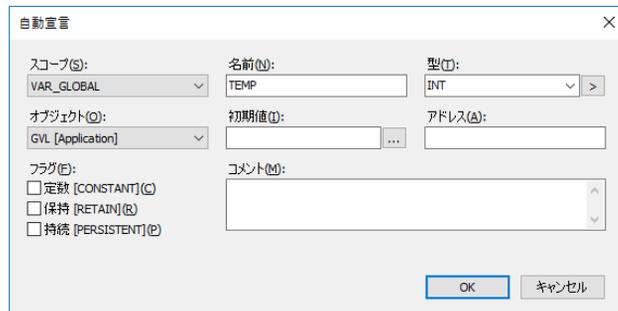
①下記の文章をプログラミングスペースに記述します。

```
IF GVL.TEMP > 50 THEN
  GVL.ALARM := 1;
ELSE
  GVL.ALARM := 0;
END_IF
```

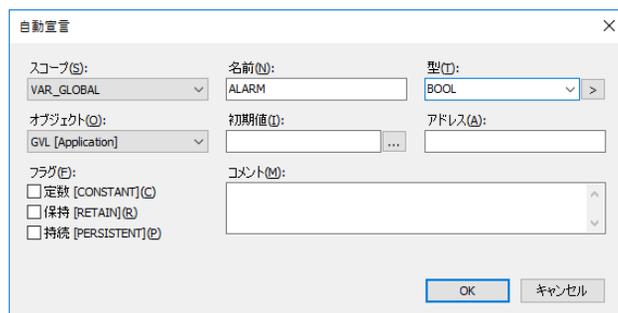


②変数作成時に、右記の[自動宣言]が表示されたら、[OK]をクリックします。

```
スコープ : VAR_GLOBAL
名前 : TEMP
型 : INT
```



```
スコープ : VAR_GLOBAL
名前 : ALARM
型 : BOOL
```



**CHECK!**

### コメント

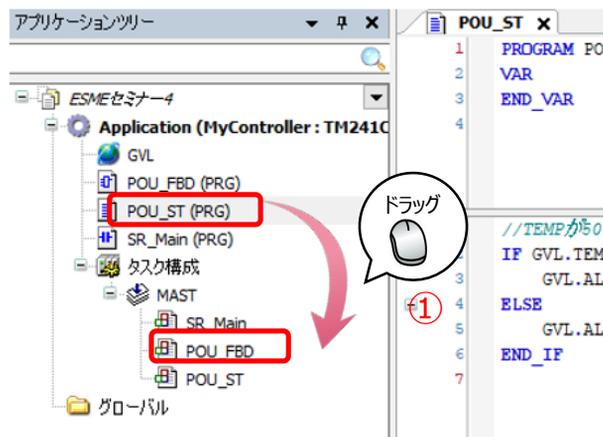
プログラム中にコメントを入力する場合は、ソースコードのあとに[ // ]と入力し、記述します。コメント部分はPLCで処理を行う際はないものとして無視されるため、自由に文を挿入することができます。

```
1 //TEMPが50を超えるとアラームが発報します。
2 IF GVL.TEMP > 50 THEN
3   GVL.ALARM:= 1; //アラームが発報
4 ELSE
5   GVL.ALARM:= 0; //アラームが復旧
6 END_IF
7
8
```

## 【POUの移動】

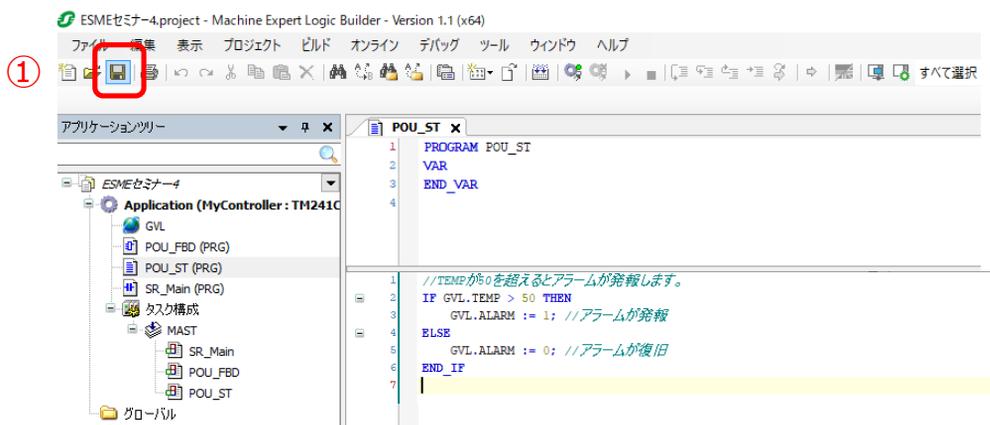
作成した[POU\_FBD]をタスク(Mast)に移動します。  
 ※タスクに移動しないと動作しません。

- ① [POU\_FBD(PRG)]をタスクに移動します。



## 【プロジェクトの保存】

- ① プログラムが完成したら  をクリックし、上書き保存します。



### 3. ビルドから運転まで

ここではプログラム作成後の手順について説明します。  
プログラム作成後はビルドを行い、PLC本体にログインし、プログラムを書き込みます。

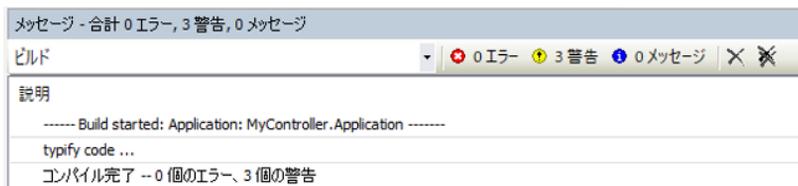
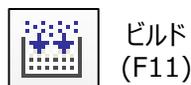


#### (1)ビルド(エラーチェック)

プログラム作成後はビルドで、エラーチェックを行います。

エラー：ダウンロードを行うことができません。

警告：ダウンロードを行うことができます(警告内容は必ず確認してください)



#### (2)ログイン

ログインを行うことで、PLC本体と論理的に接続され、プロジェクトがダウンロードされます。

①[デバイスツリー]をクリックします。

②[MyController]をクリックします。

③接続するPLCをクリックします。

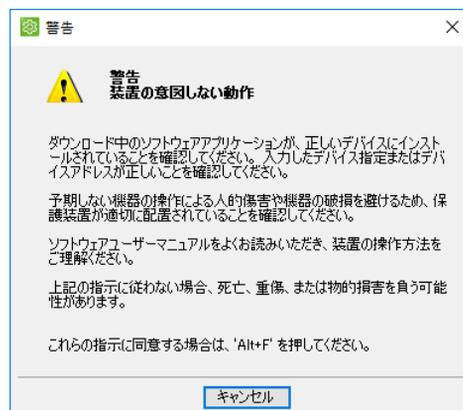
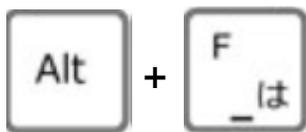
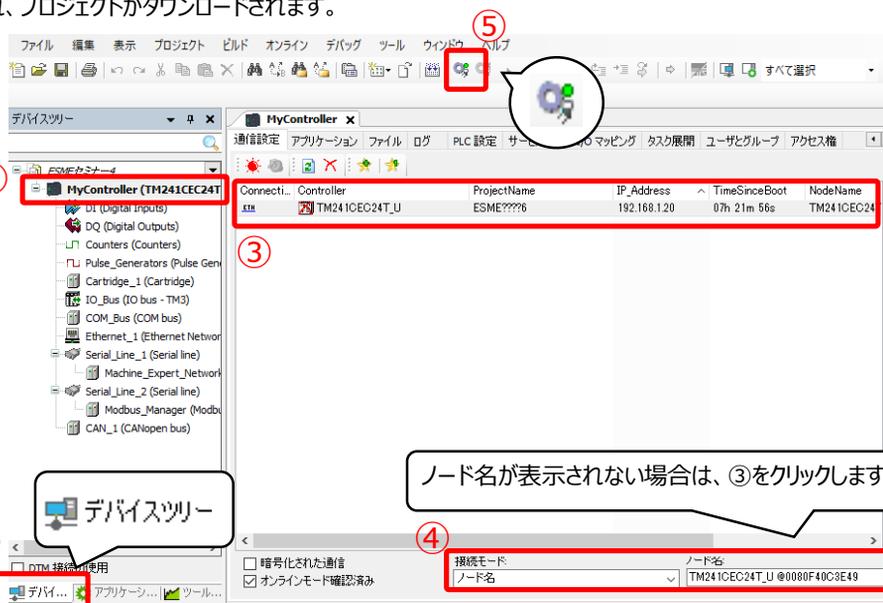
④接続モードを指定します。

接続モード：ノード名  
ノード名：TM241CEC24T\_U@○○○  
(自動入力)

⑤ をクリックします。

(または[Alt + F8]をタッチします)

⑥[警告]が表示されたら、[Alt+F]をタッチします。



### (3) 運転

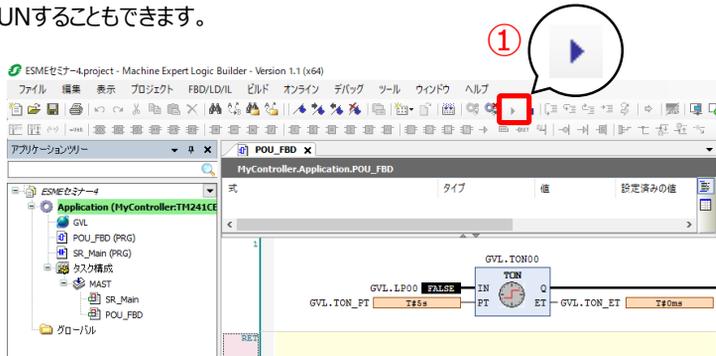
ログインしたら、プログラムをRUNします。なお、PLC本体でRUNすることもできます。

①  をクリックします(またはF5をタッチします)



②  をクリックすると運転を停止します。

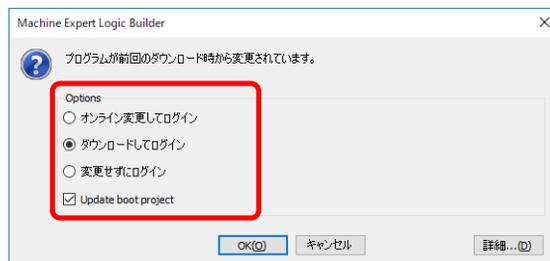
※運転/停止は画面下部で確認します。



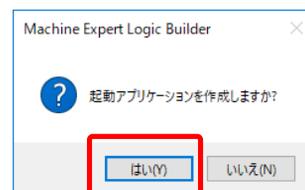
#### オンラインチェンジ = RUN中書込み

運転(RUN)中にプログラムの変更を行う場合は、一度ログアウトされた後に変更部分を一括書込みします。その際下記のダイアログボックスが表示されます。一般的には安全面を考慮して、[ダウンロードしてログイン]を選択します。

- ・(プログラムを)オンラインチェンジして、ログインします。  
 > 運転は停止せず、そのままRUNします。
- ・(プログラムを)ダウンロードしてログインします。  
 > 運転は停止します。
- ・(プログラムを)変更せずにログインします。  
 > プログラムのダウンロードは行わず、ログインのみ行います。  
 > 運転は停止せず、そのままRUNします。



右記のダイアログが表示された場合は [OK] をクリックします。

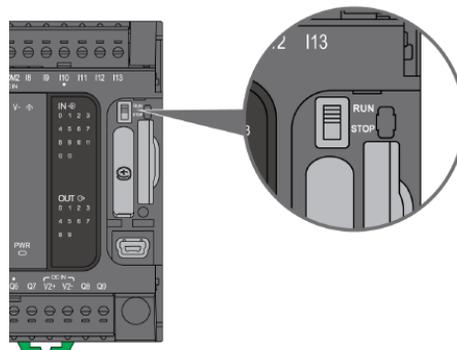


[ポスト設定警告]で[OK]をクリックします。



#### M241本体でプログラムをRUNさせるには

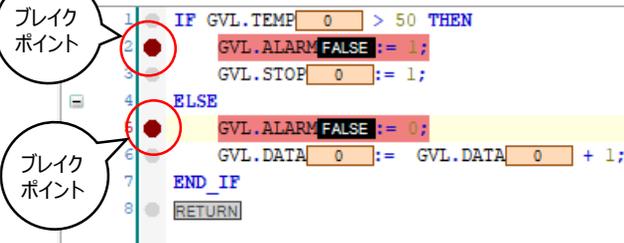
M241には運転/停止ハードウェアスイッチが備っており、コントローラーを運転または停止状態にすることができます。



CHECK!

## ブレークポイント

ブレークポイントはEcoStruxure Machine Expertに実装されているデバッグ機能です。ブレークポイントでは、指定した箇所でプログラム処理を一時停止させ、変数の値を確認したり、1命令ずつ実行することで素早いデバッグや処理のトレースを行うことができます。  
※ブレークポイントの設定はログインした状態でを行います。

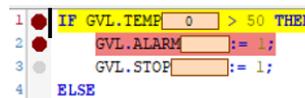


ブレークポイントは設定した場所で処理を一時停止させる機能です。

※ブレークポイントを有効にした行は色が反転します。

### 【ブレークポイントの設定と解除】

ブレークポイントを設定したい行にカーソルを移動させ、[F9]をタッチします。  
ブレークポイントを解除したい場合は[F9]を再度タッチします。

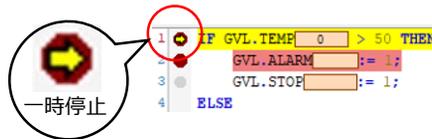


F9

### 【運転(デバッグ)開始】

 をクリックします(またはF5をタッチします)

※プログラムを実行すると設定した箇所に処理が到達した時点でプログラムが一時停止します。



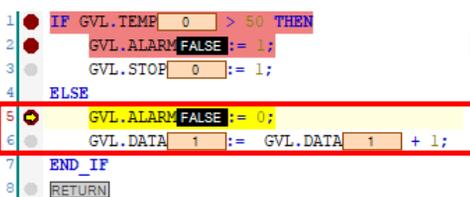
F5

## ステップ実行

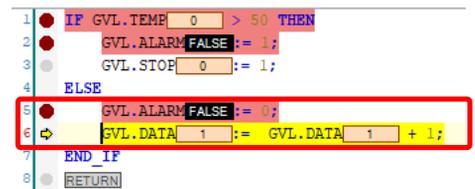
ステップ実行はブレークポイントで停止した状態から処理を進める機能です。[ステップオーバー][ステップイン][ステップアウト]の3つがあります。

### 【ステップオーバー】

[F10]をタッチ：停止中のブレークポイントの命令を1つだけ実行し、実行後は矢印が移動します。  
※ステップする命令文が関数でも、ブレークポイントを設定している関数内の次の命令に移動します。



F10



### 【ステップイン】

[F8]をタッチ：停止中のブレークポイントの命令を1つだけ実行し、実行後は矢印が移動します。  
※ステップする命令文が関数の場合は、その関数にジャンプして処理を進めます。

F8

### 【ステップアウト】

[Shift+F10]をタッチ：停止中のブレークポイントから次のブレークポイントの命令までの処理を行います。

Shift+F10

# 第5章

## ビジュアリゼーション機能

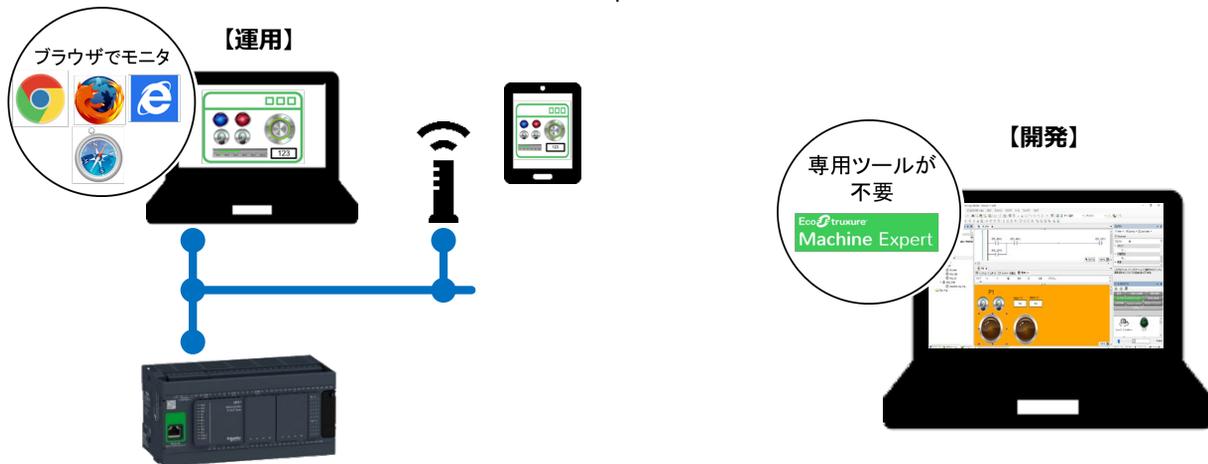
I. 【やってみよう】ビジュアリゼーション	5-1
1. ビジュアリゼーション	5-1
2. 【解説】ビジュアリゼーション機能	5-3

# I 【やってみよう】ビジュアライゼーション

## 1. ビジュアライゼーション

### (1) ビジュアライゼーション機能とは

ビジュアライゼーション機能を使用すれば、イーサネット経由でPLCのデータをパソコンやタブレット上のブラウザで表示/操作することができます。また、画面作成も専用ソフトを使わず、EcoStruxure Machine Expertで行うことができます。

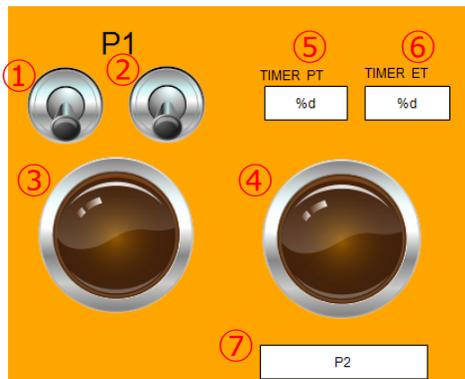


### (2) ビジュアライゼーション画面の作成例

この章では2章から4章で作成したプログラムを表示/操作するためのビジュアライゼーション画面を2つ作成します。

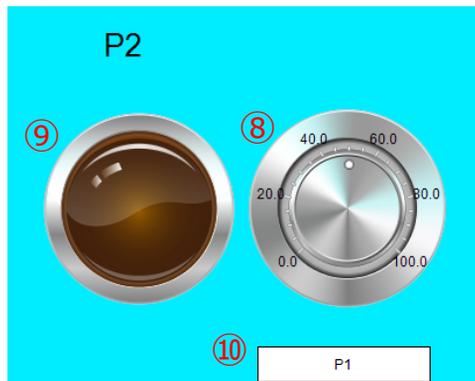
#### 【ビジュアライゼーション画面】

・P1



- ①スイッチをONすると③ランプが点灯します。
- ②スイッチをONすると③ランプが消灯します。
- ③ランプが点灯後、5秒後に④ランプが点灯します。
- ⑤タイマー設定値(5000ms)が表示されます。
- ⑥タイマー現在値が表示されます。
- ⑦スイッチで画面切替(P2)ができます。

・P2

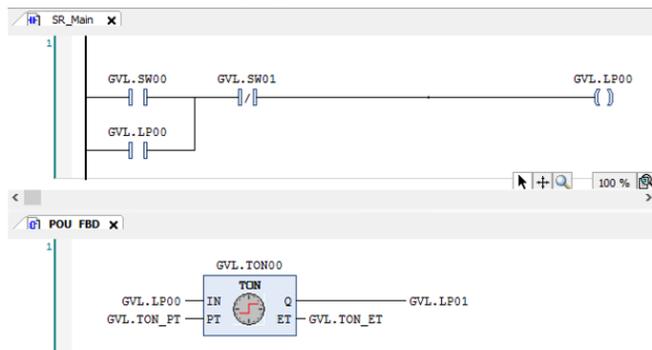


- ⑧ポジションメーターで50を超えると、⑨ランプが点灯します。
- ⑩スイッチで画面切替(P1)ができます。

#### 【プログラム】

ビジュアライゼーション画面は、2章から4章で作成したプログラムにあわせたものになっています。

・2章/3章のプログラム(P1)



・4章のプログラム(P2)

```
//TEMPが50を超えるとアラームが発報します。
IF GVL.TEMP > 50 THEN
    GVL.ALARM := 1; //アラームが発報
ELSE
    GVL.ALARM := 0; //アラームが復旧
END_IF
```

### (3) ビジュアライゼーション機能の設定手順

ビジュアライゼーション機能は、下記の手順に沿って行います。設定は[ツールツリー]から行います。

【設定の流れ】

プログラミング

#### ① ビジュアライゼーションマネージャの追加

ビジュアライゼーション機能全般の設定を行います(スタイル/セキュリティ等)

#### ② ビジュアライゼーション(画面)の追加と作成

表示用の画面を追加し、作成します。

#### ③ Webビジュアライゼーションの設定

Webブラウザで表示するための基本設定を行います。

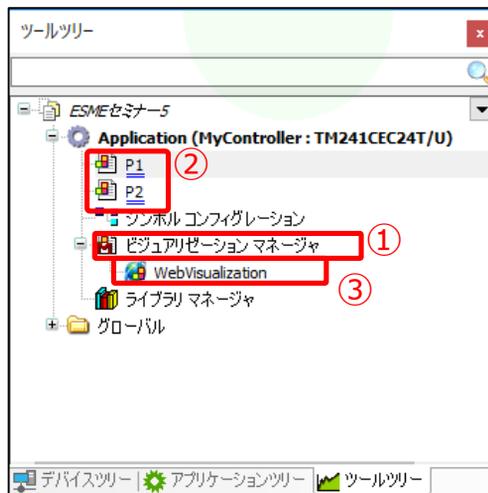
ビルド/ログイン(ダウンロード)

#### ④ Webビジュアライゼーションの起動

WebブラウザにURLを入力し、Webビジュアライゼーションの起動します。

例) <http://192.168.1.20:8080/webvisu.htm>

【ツールツリー】



## 2. 【解説】ビジュアライゼーション機能

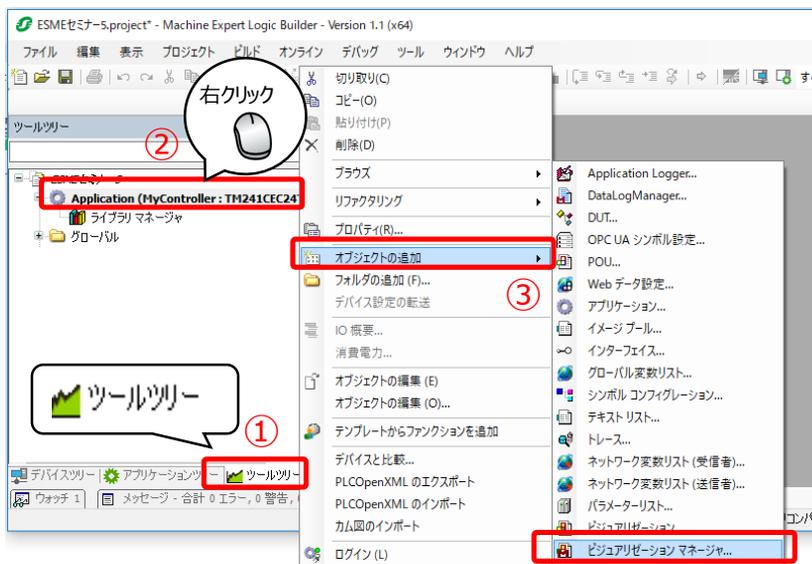
### (1)ビジュアライゼーションマネージャーの追加

ビジュアライゼーション機能全般の設定を行います。ビジュアライゼーション機能を使用する際は、最初にビジュアライゼーションマネージャーを追加します。

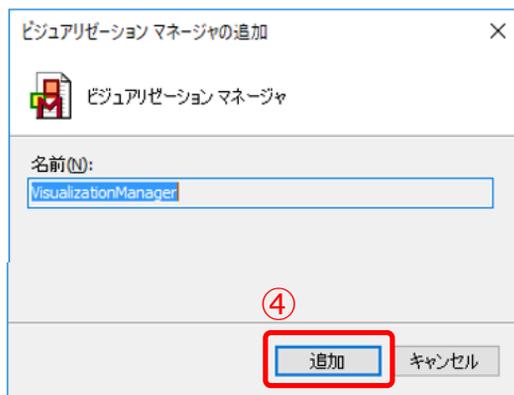
①[ナビゲーター]の左下の[ツールツリー]をクリックします。

②[Application]を右クリックします。

③[オブジェクトの追加][ビジュアライゼーションマネージャー]をクリックします。

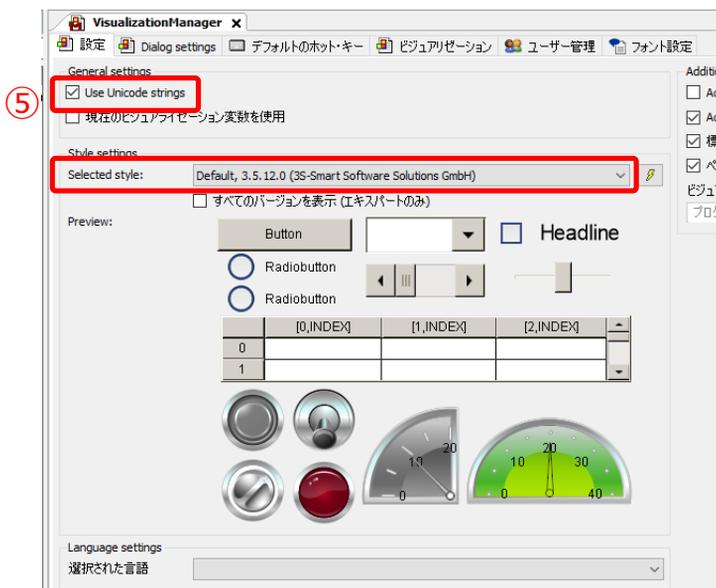


④[ビジュアライゼーションマネージャー]で[追加]をクリックします。



⑤ビジュアライゼーションマネージャーで基本設定を行います。

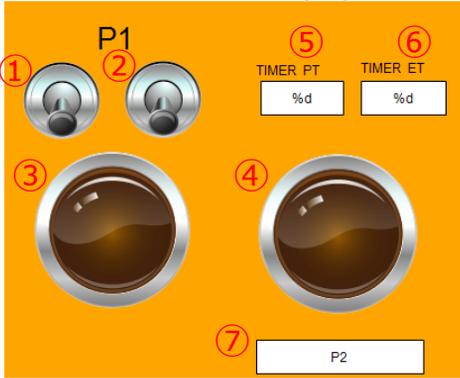
- (チェック) : Use Unicode strings  
※チェックを入れると、日本語表示ができます。
- Selected Style : 任意



## (2)ビジュアライゼーション画面の追加と作成(P1)

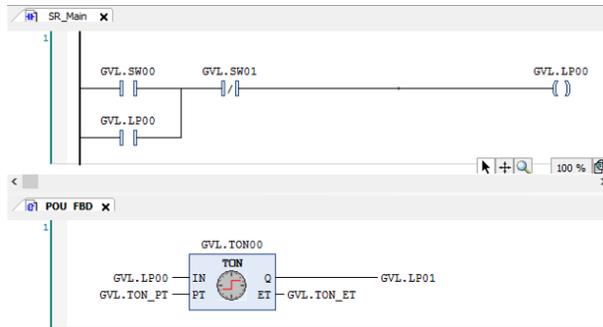
ここではWebブラウザで表示する画面の作成を行います。ビジュアライゼーション(画面)を追加し、ツールツリーを使い、画面作成を行います。

### 【完成例】ビジュアライゼーション画面(P1)



- ①スイッチをONすると③ランプが点灯します。
- ②スイッチをONすると③ランプが消灯します。
- ③ランプが点灯後、5秒後に④ランプが点灯します。
- ⑤タイマー設定値(5000ms)が表示されます。
- ⑥タイマー現在値が表示されます。
- ⑦スイッチで画面切替(P2)ができます。

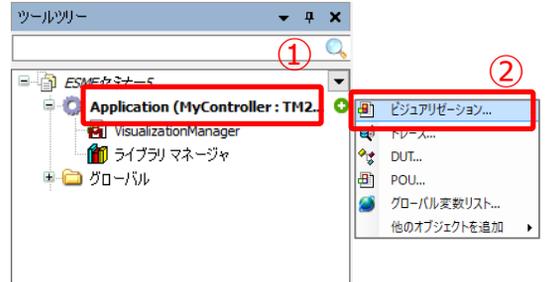
P1は2章、3章で作成したプログラムに対応しています。



### 【P1画面作成】

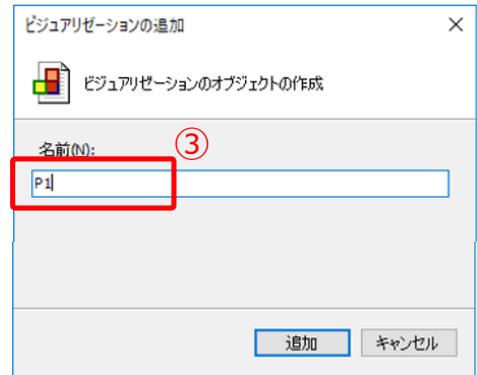
①[Application]を選択し、 をクリックします。

②[ビジュアライゼーション]をクリックします。



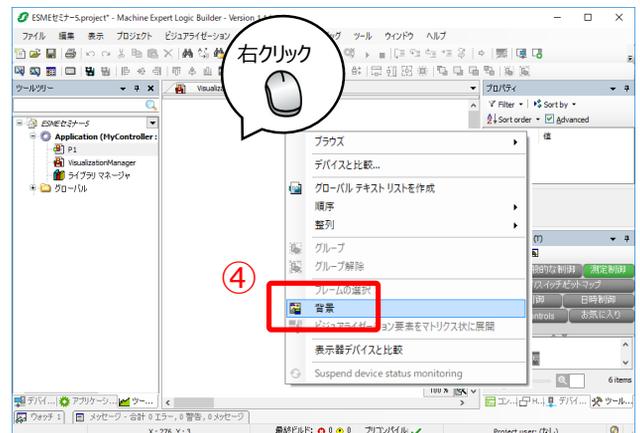
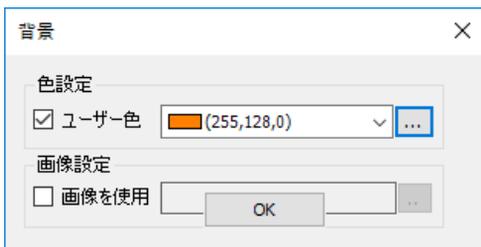
③[ビジュアライゼーションの追加]で画面の名称を入力します。

名前: P1



④ビジュアライゼーション画面が追加されたら、右クリックし、背景色を選択します。

背景色: 任意

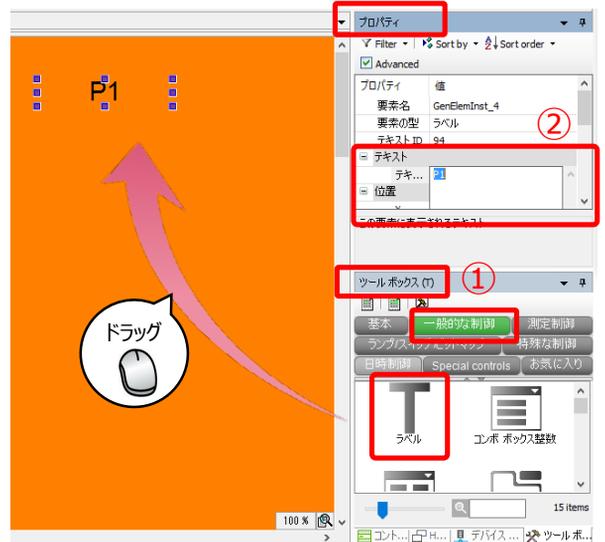


【ラベルの配置】

①[ツールボックス]から[一般的な制御]を選択し、[ラベル]をワークスペースにドラッグ&ドロップします。

②[プロパティ]から[テキスト][テキスト]で文字列を入力します。

テキスト : P1



【スイッチ1の配置】

①[ツールボックス]から[ランプ/スイッチ/ビットマップ]を選択し、[ディップスイッチ]をワークスペースにドラッグ&ドロップします。

②[プロパティ]で変数を指定します。

変数 : GVL.SW00



変数の割り付け方

作成済みの変数を割り付けるには、変数名の右に表示される  をクリックするか、または[F2キー]をタッチします。[入力アシスタント]を使用して変数を割り付けることができます。

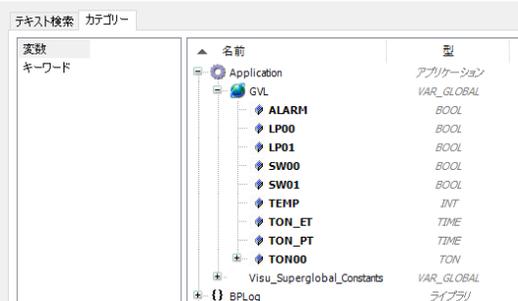
変数



F2



入力アシスタント



## 【スイッチ2の配置】

① [ツールボックス]から[ランプ/スイッチ/ビットマップ]を選択し、[ディップスイッチ]をワークスペースにドラッグ&ドロップします。

② [プロパティ]で変数を指定します。

変数 : GVL.SW01

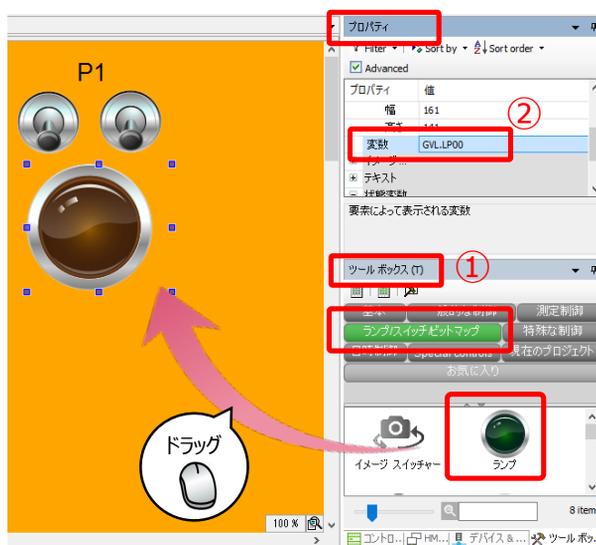


## 【ランプの配置】

① [ツールボックス]から[ランプ/スイッチ/ビットマップ]を選択し、[ランプ]をワークスペースにドラッグ&ドロップします。

② [プロパティ]で変数を指定します。

変数 : GVL.LP00





### モーメンタリースイッチ

モーメンタリースイッチを作成する場合は、[ツールボックス][一般的な制御]の[ボタン]を使用します。

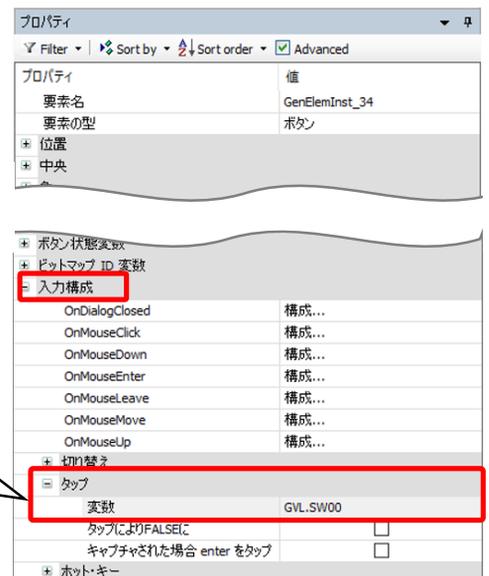
※ここではGVL.SW00(BOOL変数)をON/OFFします。

- ①[ツールボックス]から[一般的な制御]を選択し、[ボタン]をワークスペースにドラッグ&ドロップします。



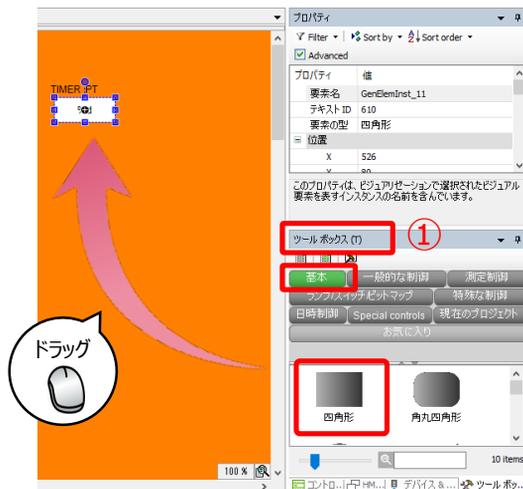
- ②[プロパティ]の[入力構成]で下記の設定を行います。

入力構成  
 タップ  
 ・変数 : GVL.SW00



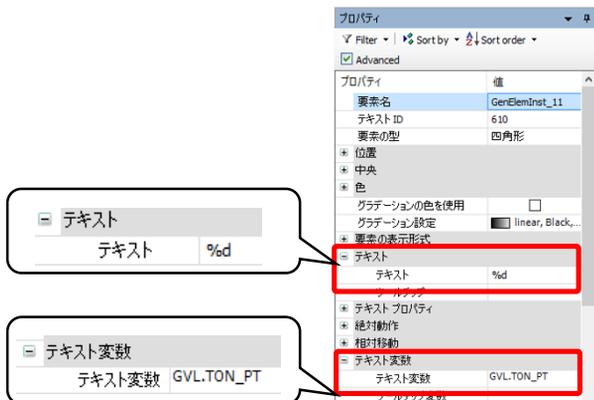
【四角形の配置とデータ表示①】

①[ツールボックス]から[基本]を選択し、[四角形]をワークスペースにドラッグ&ドロップします。



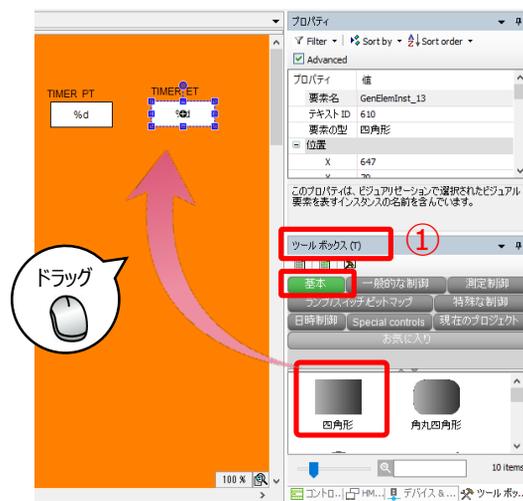
②[プロパティ]で下記の設定を行います。

テキスト : %d  
 テキスト変数 : GVL.TON\_PT



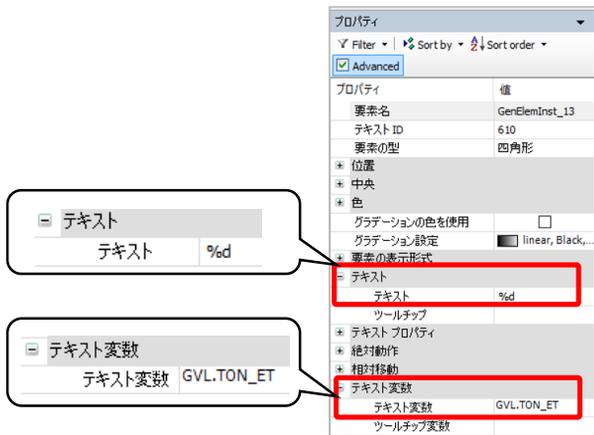
【四角形の配置とデータ表示②】

①[ツールボックス]から[基本]を選択し、[四角形]をワークスペースにドラッグ&ドロップします。



②[プロパティ]で下記の設定を行います。

テキスト : %d  
 テキスト変数 : GVL.TON\_ET

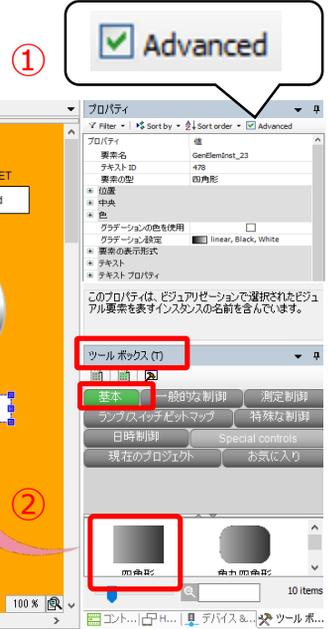




【画面切替スイッチの配置】

[四角形]を使って画面切替スイッチを作成します。設定を行う前に必ず、[Advanced]にチェックを入れます。

① [Advanced]にチェックを入れます。

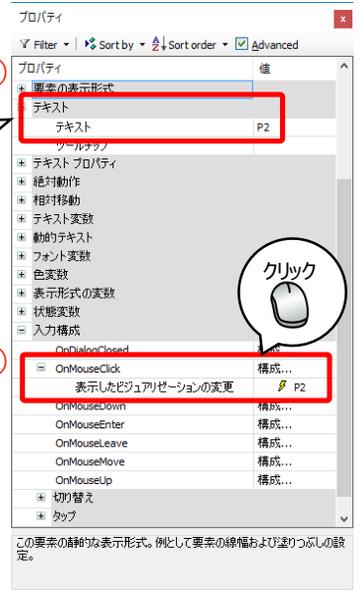


② [ツールボックス]から[基本]を選択し、[四角形]をワークスペースにドラッグ&ドロップします。



③ [プロパティ]の[テキスト]で下記の設定を行います。

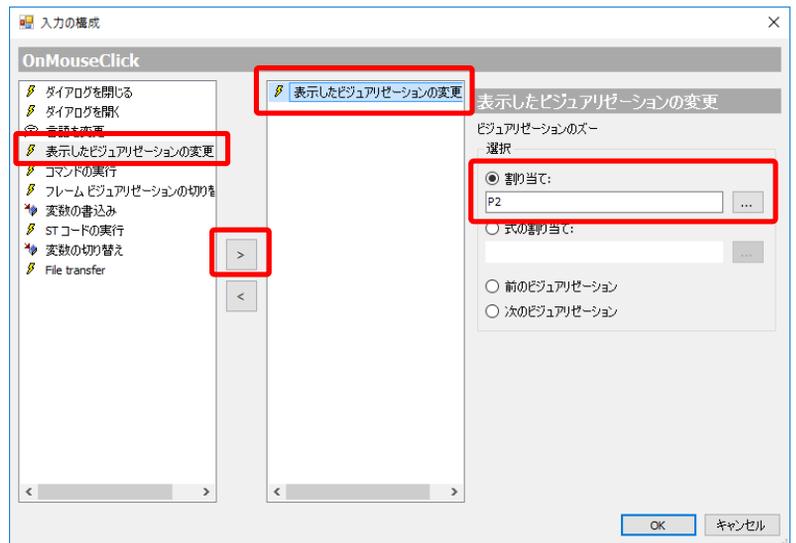
テキスト : P2



④ [プロパティ]の[入力構成]で下記の設定を行います。

入力構成 : OnMouseClicked  
表示したビジュアルセッションの変更  
P2

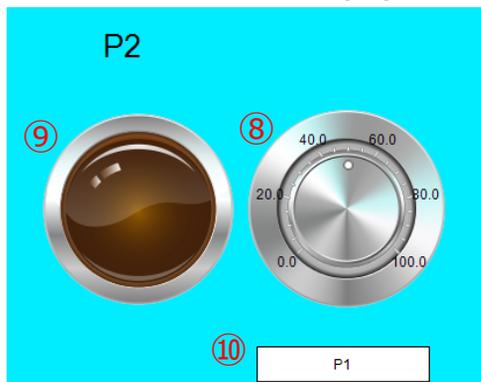
※[入力の構成]では下記のダイアログを使用すると設定が簡単です。



### (3) ビジュアライゼーション画面の追加と作成(P2)

ここではWebブラウザで表示する画面の作成を行います。ビジュアライゼーション(画面)を追加し、ツールツリーを使い、画面作成を行います。

#### 【完成例】ビジュアライゼーション画面(P2)



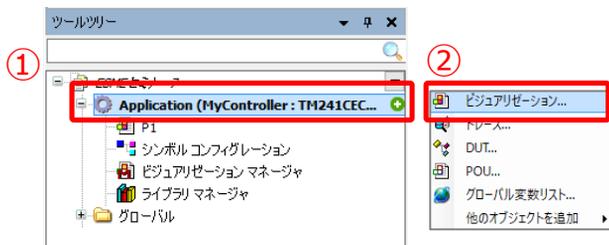
- ⑧ポジションメーターで50を超えると、⑨ランプが点灯します。
- ⑩スイッチで画面切替(P1)ができます。

P2は4章で作成したプログラムに対応しています。

```
//TEMPが50を超えるとアラームが発報します。
IF GVL.TEMP > 50 THEN
    GVL.ALARM := 1; //アラームが発報
ELSE
    GVL.ALARM := 0; //アラームが復旧
END_IF
```

#### 【P1画面作成】

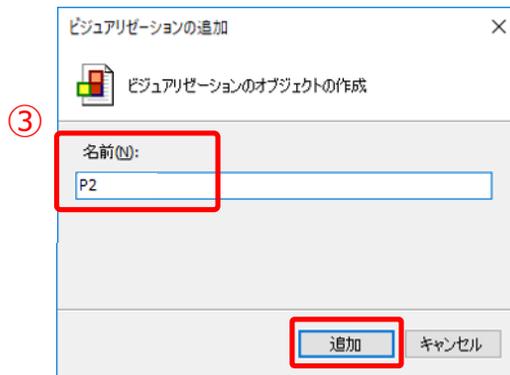
①[Application]を選択し、 をクリックします。



②[ビジュアライゼーション]をクリックします。

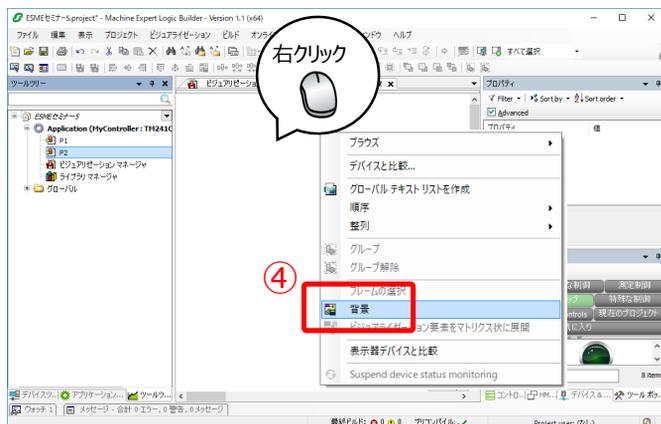
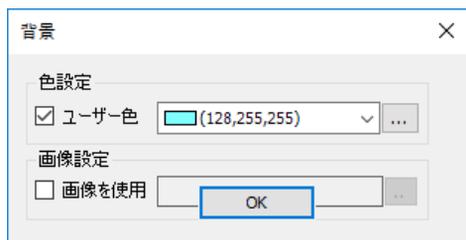
③[ビジュアライゼーションの追加]で画面の名称を入力します。

名前: P2



④ビジュアライゼーション画面が追加されたら、右クリックし、背景色を選択します。

背景色: 任意

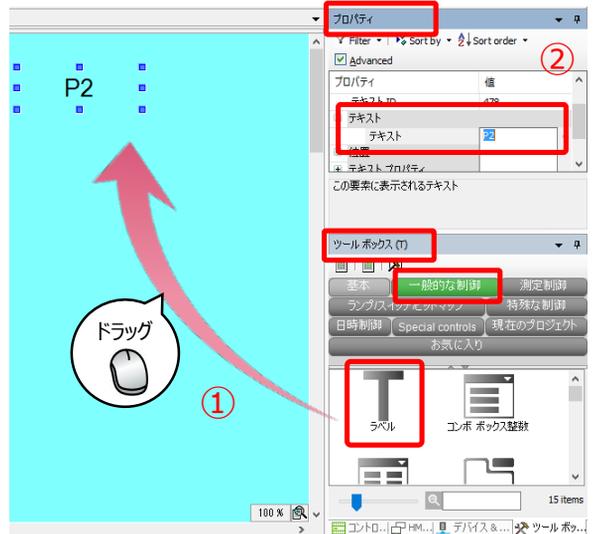


【ラベルの配置】

① [ツールボックス]から[一般的な制御]を選択し、[ラベル]をワークスペースにドラッグ&ドロップします。

② [プロパティ]から[テキスト][テキスト]で文字列を入力します。

テキスト : P2

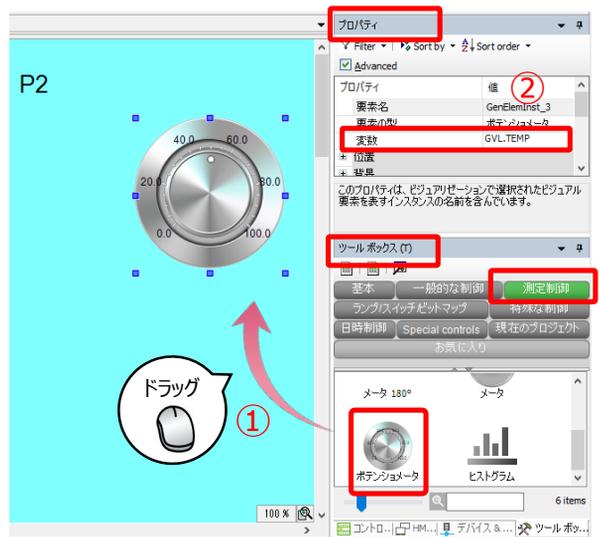


【ポジションメーターの配置】

① [ツールボックス]から[測定制御]を選択し、[ポジションメーター]をワークスペースにドラッグ&ドロップします。

② [プロパティ]で変数を指定します。

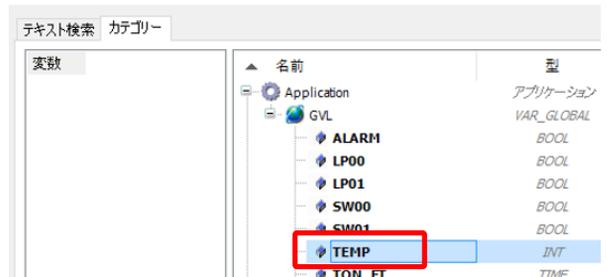
変数 : GVL.TEMP



※変数の指定は[入力アシスタント]を使えば、簡単です。



入力アシスタント

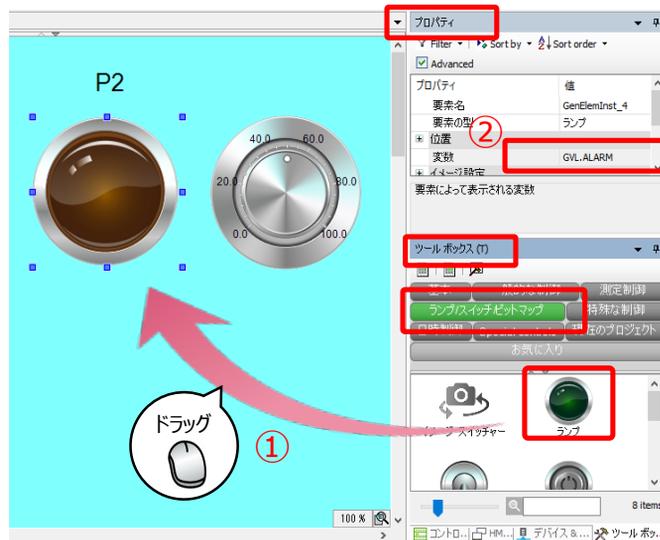


## 【ランプの配置】

① [ツールボックス]から[ランプ/スイッチ/ビットマップ]を選択し、[ランプ]をワークスペースにドラッグ&ドロップします。

② [プロパティ]で変数を指定します。

変数 : GVL.ALARM



【画面切替スイッチの配置】

[四角形]を使って画面切替スイッチを作成します。設定を行う前に必ず、[Advanced]にチェックを入れます。

① [Advanced]にチェックを入れます。

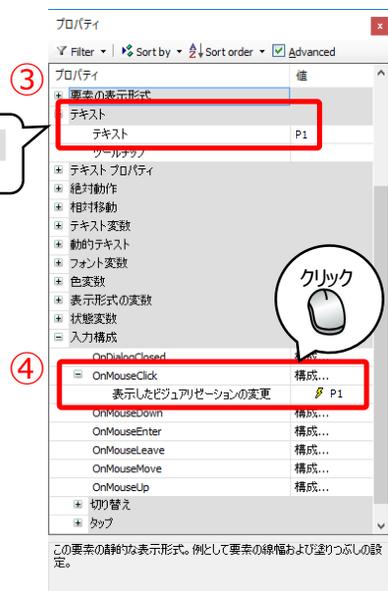
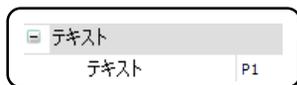


② [ツールボックス]から[基本]を選択し、[四角形]をワークスペースにドラッグ&ドロップします。



③ [プロパティ]の[テキスト]で下記の設定を行います。

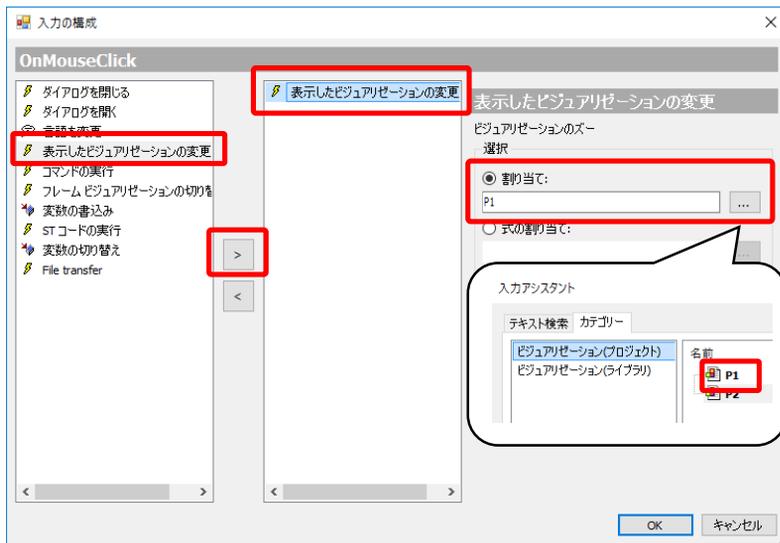
テキスト : P1



④ [プロパティ]の[入力構成]で下記の設定を行います。

入力構成 : OnMouseClicked  
表示したビジュアルセッションの変更  
P1

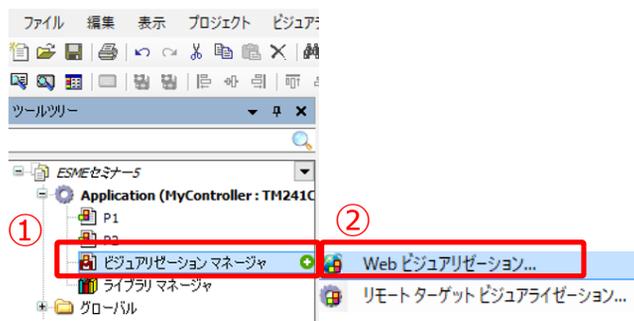
※[入力の構成]では下記のダイアログを使用すると設定が簡単です。



## (4) Webビジュアライゼーションの設定

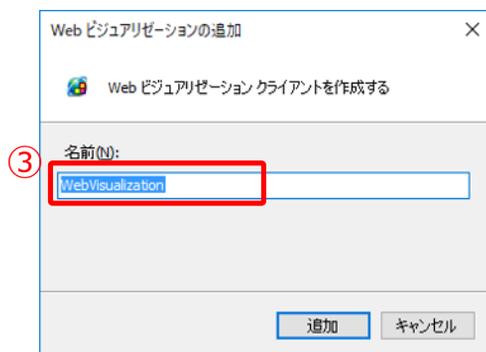
Webブラウザで表示するための基本設定を行います。

- ① [ビジュアライゼーションマネージャ] を選択し、 をクリックします。
- ② [Webビジュアライゼーション] をクリックします。



- ③ [Webビジュアライゼーションの追加] で名称を入力します。

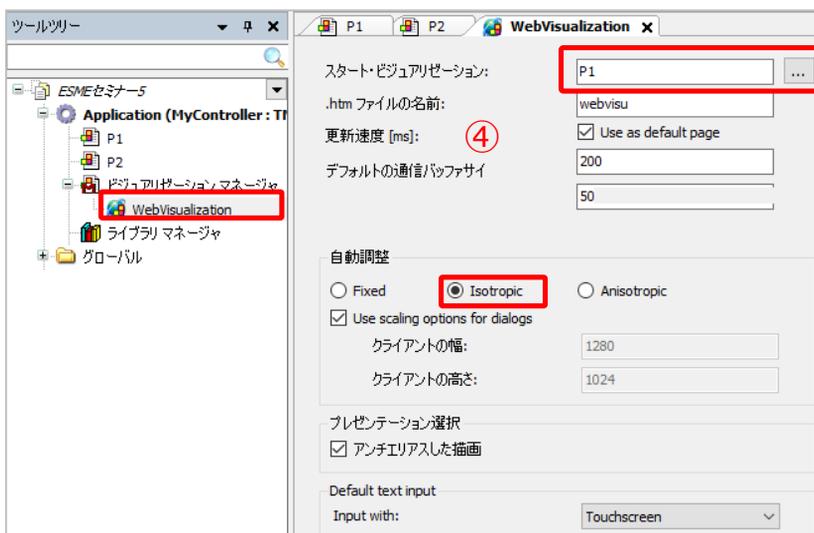
名前: WebVisualization(任意)



- ④ [WebVisualization] で下記の設定を行います。

スタート・ビジュアライゼーション: P1  
※初期画面の設定

自動調整: Isotropic  
※表示モードの設定

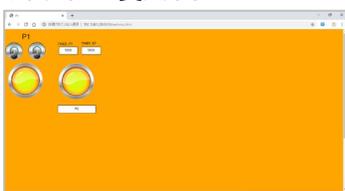


**CHECK!**

### 自動調整(WebVisualization)

WebVisualizationの表示モードには下記の3つが用意されています。

**fixed:**  
ブラウザサイズに関わらず、オブジェクトサイズは変更なし



**Isotropic**  
ブラウザサイズに合わせて、オブジェクトサイズを等倍



**Anisotropic**  
ブラウザサイズに合わせて、オブジェクトサイズを非等倍

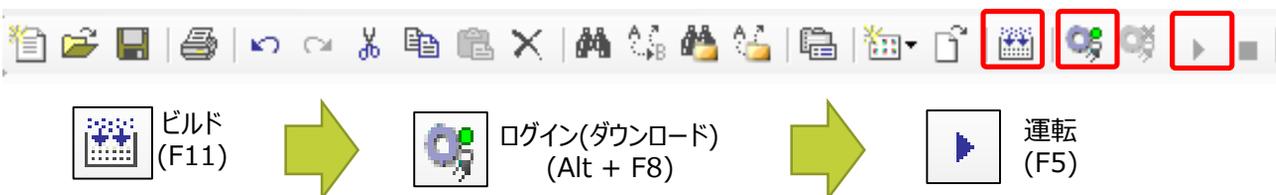


## (5) Webビジュアリゼーションの起動

WebブラウザにURLを入力し、Webビジュアリゼーションの起動します。

### ビルドから運転まで

Webビジュアリゼーションの設定終了後は、ビルド後、ログイン/運転を行います。



① Webブラウザに下記のURLを入力します。

<http://192.168.1.20:8080/webvisu.htm>

PLC本体のIPアドレス  
(2-3参照)

ポート番号  
(固定)

Webビジュアリゼーションの設定で変  
更することができます。



### CAUTION

- WebビジュアリゼーションはChrome、FireFox、Safari などHTML5対応のブラウザをお使いください。
- Internet Explorer11では正常に表示しないことがあります。



Chrome



Safari



Firefox



Microsoft Edge



# 第6章

## Pro-face製表示器との接続

I . Modicon M241シリーズとPro-face製表示器	6-1
1. 3つの接続方法	6-1
II . Pro-face製表示器との接続方法	6-2
1. CoDeSysイーサネット接続の流れ	6-2
2. EcoStruxure Machine Expertの設定	6-3
3. GP-Pro EXの設定	6-5
III . 【やってみよう】Pro-face製表示器との接続	6-8
1. Pro-face製表示器との接続	6-8
2. 【解説】Pro-face製表示器との接続	6-9

# I. Modicon M241シリーズとPro-face製表示器

## 1. 3つの接続方法

M241とPro-face製表示器との接続方法には、[CoDeSysイーサネット] [Modbus] [EtherNet/IP]の3つがあります。特に[CoDeSysイーサネット]で接続すれば、EcoStruxure Machine Expertの変数をGP-Pro EXで使用することができます。ここでは[CoDeSysイーサネット]を使用したPro-face製表示器との接続方法について解説します。

### 【Pro-face製表示器との接続方法】

CoDeSysイーサネット

Modbus

EtherNet/IP

※イーサネット接続できるのは、M241のイーサネット組込み機種のみです。

## (1) CoDeSysイーサネット接続

M241とPro-face製表示器との接続には、[CoDeSysイーサネット]ドライバーを用いると便利です。EcoStruxure Machine Expertで作成した変数情報をPro-face製画面ソフトGP-Pro EXに取り組みことにより、簡単に設定を行うことができます。

### 【CoDeSysイーサネット接続】



M241シリーズ

イーサネット通信

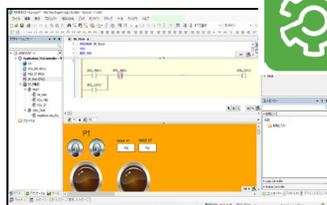


Pro-face製表示器 SP5000シリーズ

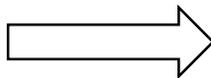
※インターフェイス：イーサネット

### 【変数情報のインポート/エクスポート】

EcoStruxure Machine Expertで作成した変数情報をPro-face製画面ソフトGP-Pro EXにインポート/エクスポートすることができます。



変数情報



MyController.Application.xml



## (2) Modbus接続

ModbusとはModicon(現シュナイダーエレクトリック)が開発したFA機器間の通信プロトコルです。仕様は一般に公開されており、汎用的に採用されています。

Modbusプロトコルは、(Modbus)シリアル接続と(Modbus TCP)イーサネット接続があり、Pro-face製表示器とはどちらを使っても接続できます。

## (3) EtherNet/IP接続

EtherNet/IPは、Ethernetを使用した産業用のネットワークです。M241とPro-face製表示器との接続に用いることができます。



CoDeSys

CoDeSysとは、PLCのプログラム開発が可能なIEC 61131-3準拠の統合開発環境およびランタイムソフトウェアです。EcoStruxure Machine Expertは、このCoDeSysをベースとして作成されています。

## II. Pro-face製表示器との接続方法

### 1. CoDeSysイーサネット接続の流れ

#### (1) CoDeSysイーサネット接続の流れ

CoDeSysイーサネット接続する場合は、下記の手順に沿って行います。設定はEcoStruxure Machine Expertでは変数情報を含んだXMLファイルを作成します。GP-Pro EXでは通信ドライバーの設定とXMLファイルのインポートを行います。

##### 【設定の流れ】

①シンボルコンフィグレーションの追加  
シンボルコンフィグレーションを追加し、エクスポートする変数を指定します。



②XMLファイルの作成  
変数情報を含んで、XMLファイルを作成します。



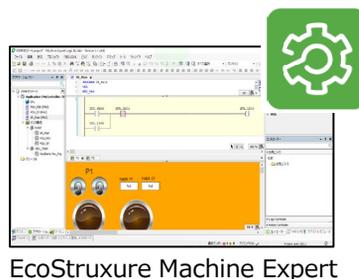
③CoDeSys V3 Ethernetドライバーの設定  
GP-Pro EXで通信ドライバーの設定を行います。



④XMLファイルのインポート  
②で作成したXMLファイルをGP-Pro EXにインポートします。



⑤変数を使った作画  
GP-Pro EXで作画および各種設定を行います。



EcoStruxure Machine Expert



GP-Pro EX



MEMO

## 2. EcoStruxure Machine Expertの設定

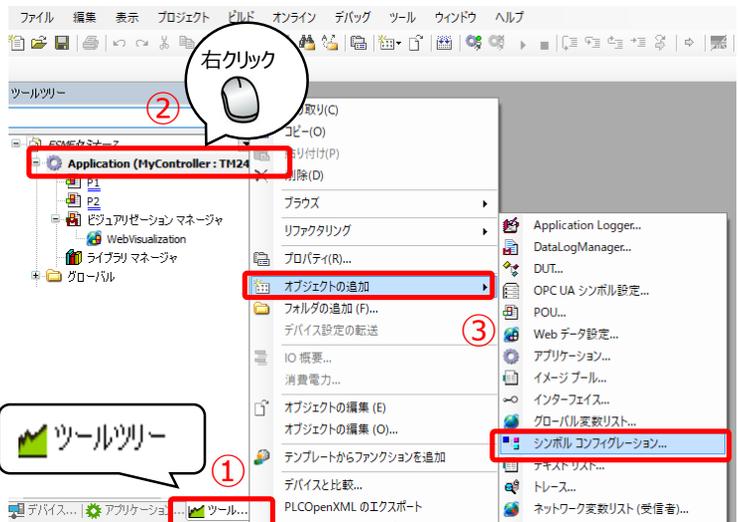
### (1)シンボルコンフィグレーションの追加

変数情報をエクスポートするためには、[シンボルコンフィグレーション]を追加します。

①[ナビゲーター]の左下の[ツールツリー]をクリックします。

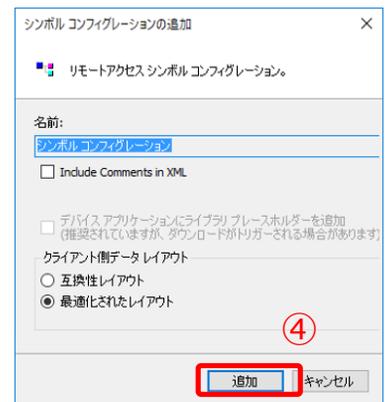
②[Application]を右クリックします。

③[オブジェクトの追加][シンボルコンフィグレーション]をクリックします。



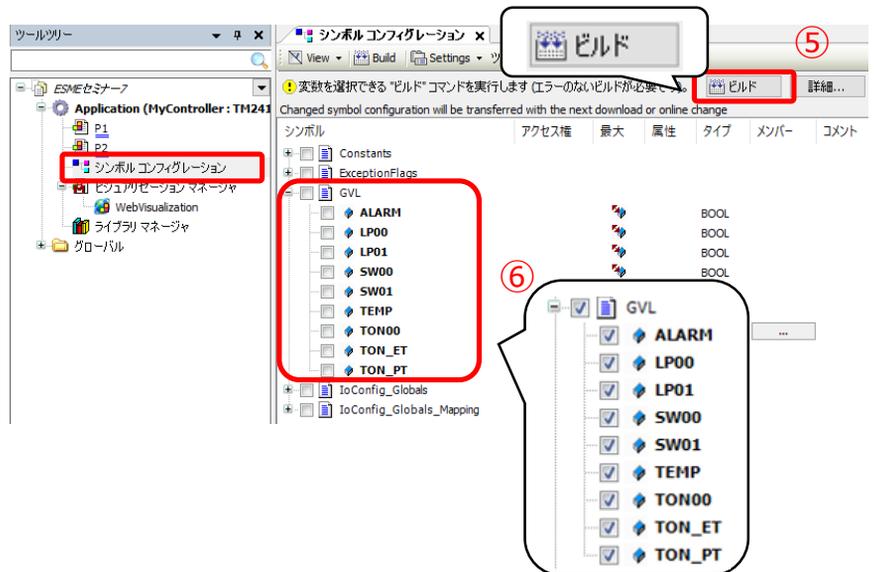
④[シンボルコンフィグレーション]で[追加]をクリックします。

- ・名前：任意
- ・クライアント側データレイアウト：最適化されたレイアウト



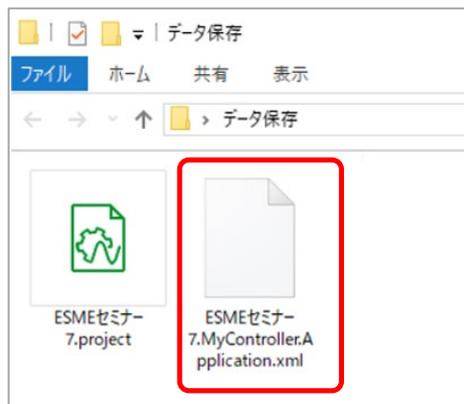
⑤[ビルド]をクリックします。

⑥エクスポートしたい変数にチェックを入れます。



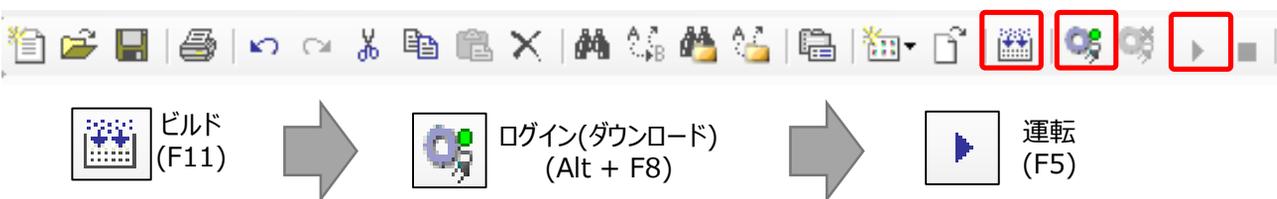
## (2) XMLファイルの作成

変数情報を含んだXMLファイルは[シンボルコンフィグレーション]の設定を行った後、ログインするとプロジェクトを保存している場所に生成されます。



### ビルドから運転まで

Webビジュアライゼーションの設定終了後は、ビルド後、ログイン/運転を行います。



CHECK!

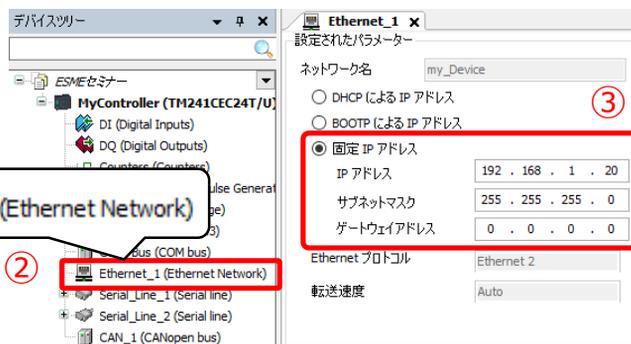
### IPアドレスの設定

CoDeSysイーサネット接続では、イーサネットを使用して接続するため、IPアドレスの設定が必要です。IPアドレスが未設定の場合は下記のように接続します。

- ①[ナビゲーター]の左下の[デバイスツリー]をクリックします。



- ②デバイスツリーから[Ethernet1]をクリックします。



- ③M241自身のIPアドレスを下記のように指定します。

固定IPアドレス  
IPアドレス : 192.168.1.20  
サブネットマスク : 255.255.255.0

## 3. GP-Pro EXの設定

### (1)CoDeSys V3 Ethernetドライバーの設定

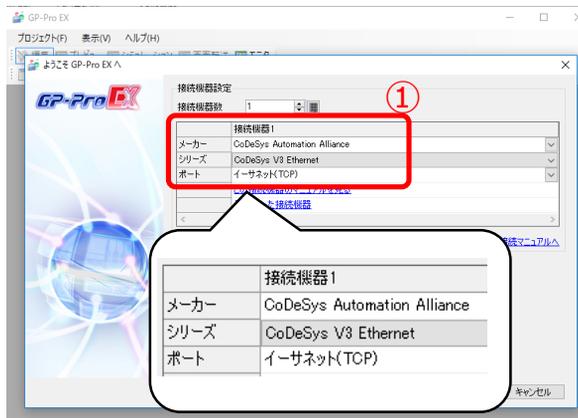
GP-Pro EXでは、[接続機器の指定][通信設定]の2つを行います。[接続機器の指定]では通信ドライバーの指定します。また[通信設定]では変数情報のインポートを行います。

#### 【接続機器の指定(通信ドライバー)】

①GP-Pro EX起動時の接続機器設定で下記の設定を行います。

##### 接続機器1

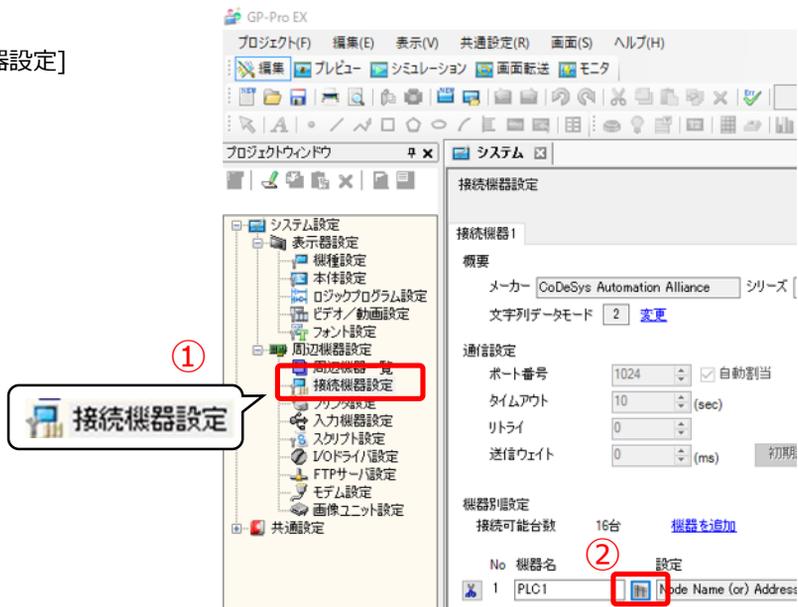
- ・メーカー：CoDeSys Automation Alliance
- ・シリーズ：CoDeSys V3 Ethernet
- ・ポート：イーサネット(TCP)



#### 【通信設定(変数情報のインポート)】

①メニューバー[プロジェクト][システム設定][接続機器設定]をクリックします。

②をクリックします。



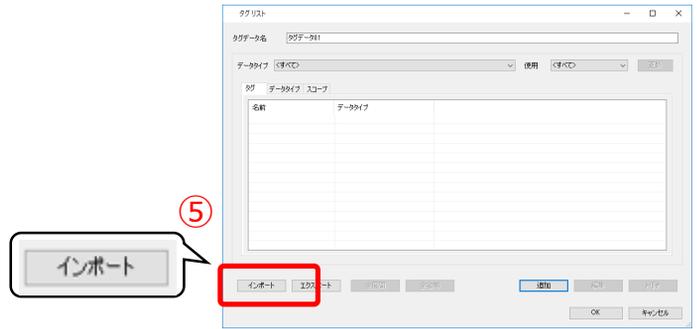
③[個別機器設定]で下記の設定を行います。

- ・Node Name Address：※EcoStruxure Machine Expertで設定したもの
- ・ (チェック)タグデータを使用する

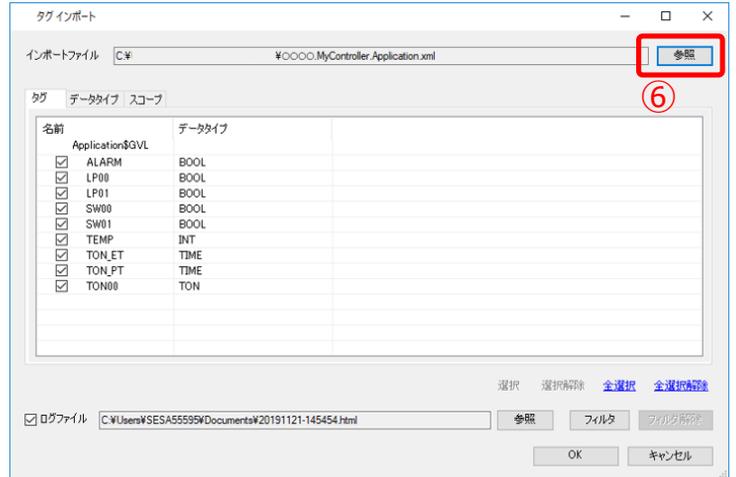
④[新規]をクリックします。



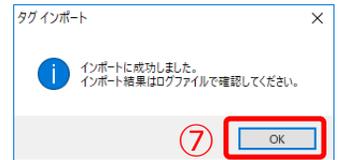
⑤ [タグリスト]で[インポート]をクリックします。



⑥ [タグインポート]の[参照]をクリックします。  
※EcoStruxure Machine Expertで作成したXMLファイル  
を指定します。

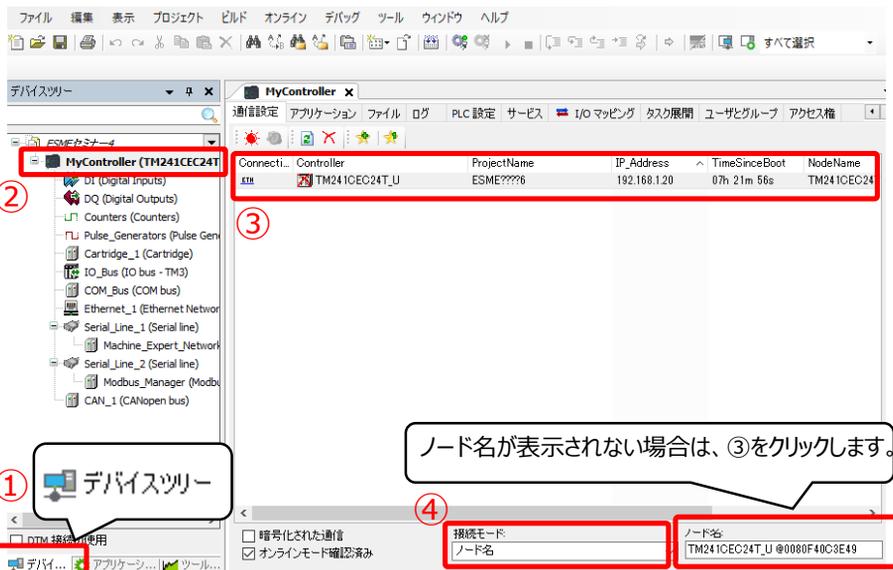


⑦ [OK]をクリックします。  
※その他のダイアログも閉じます。



ノード名

GP-Pro EXの個別機器設定で指定する[ Node Name Address]は、EcoStruxure Machine Expertのデバイスツリーで確認できます。  
ノード名はコピー & ペーストすれば、簡単に行うことができます。

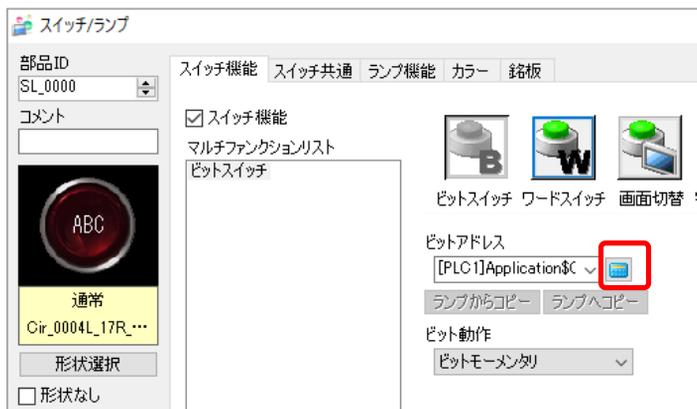


- ① [デバイスツリー]をクリックします。
- ② [MyController]をクリックします。
- ③ 接続するPLCをクリックします。
- ④ 接続モードで[ノード名]を選択します。
- ⑤ ノード名が表示されます。

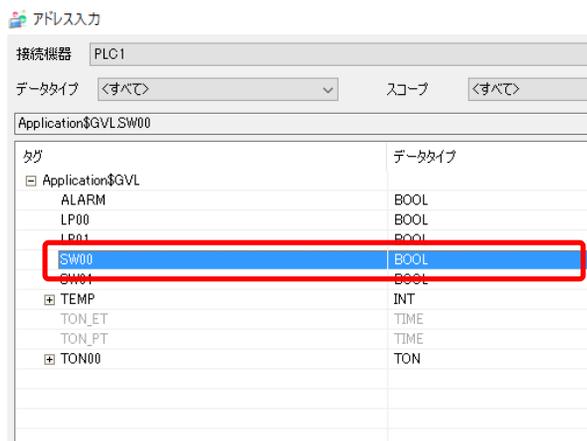
## (2)変数の割り付け

GP-Pro EXの部品に変数を割り付けるには、各部品のアドレスリストから変数を指定します。  
ここではスイッチに変数(SW00)を割り付けています。

①[スイッチ/ランプ]の  をクリックします。



②[アドレス入力]で変数(SW00)を指定します。



MEMO

### Ⅲ.【やってみよう】Pro-face製表示器との接続

ここではM241とPro-face製表示器を接続し、2章から4章までのプログラムを表示/操作する表示器画面を作成します。要求仕様を参考にし、表示器画面を作成しましょう。

#### 1. Pro-face製表示器との接続

下記の環境を想定して設定を行います。

##### ■ソフトウェア

プログラミングツール[EcoStruxure Machine Expert]

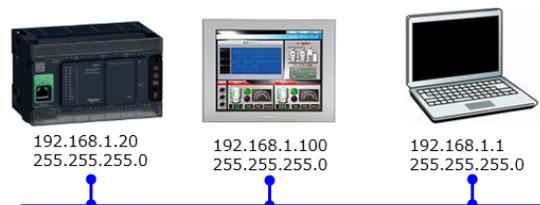
画面作成ソフトウェア[GP-Pro EX]

##### ■ハードウェア

プログラマブルコントローラ : Modicon M241シリーズ(TM241CEC24U)

プログラマブル表示器 : GP-4501T(PFXGP4501TAA)

パソコン : Windows10が動作するパソコン



#### (1)表示器画面の作成例

この章では2章から4章で作成したプログラムを表示/操作するための表示器画面を2つ作成します。



##### 【要求仕様】

- ①スイッチをONすると③ランプが点灯します。
- ②スイッチをONすると③ランプが消灯します。
- ③ランプが点灯後、5秒後に④ランプが点灯します。
- ⑤タイマー設定値(5000ms)が表示されます。  
また、キーボードから設定値の変更ができます。
- ⑥タイマー現在値が表示されます。
- ⑦スイッチで画面切替(P2)ができます。



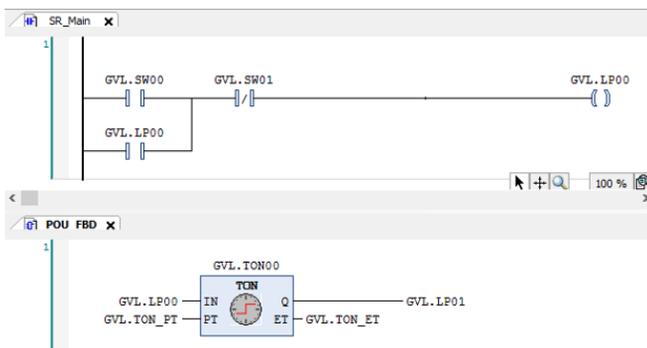
##### 【要求仕様】

- ⑧温度データが表示されます。
- ⑨タッチするたびに温度データが1ずつ加算されます。
- ⑩タッチするたびに温度データが1ずつ減算されます。
- ⑧温度データが50を超えると⑪ランプが点灯します。
- ⑫スイッチで画面切替(P1)ができます。

#### 【プログラム】

表示機器画面は、2章から4章で作成したプログラムにあわせたものになっています。

##### ・2章/3章のプログラム(P1)



##### ・4章のプログラム(P2)

```
//TEMPが50を超えるとアラームが発報します。
IF GVL.TEMP > 50 THEN
  GVL.ALARM:= 1; //アラームが発報
ELSE
  GVL.ALARM:= 0; //アラームが復旧
END_IF
```

## 2. 【解説】Pro-face製表示器との接続

### (1)GP-Pro EXの設定(接続機器設定/通信設定)

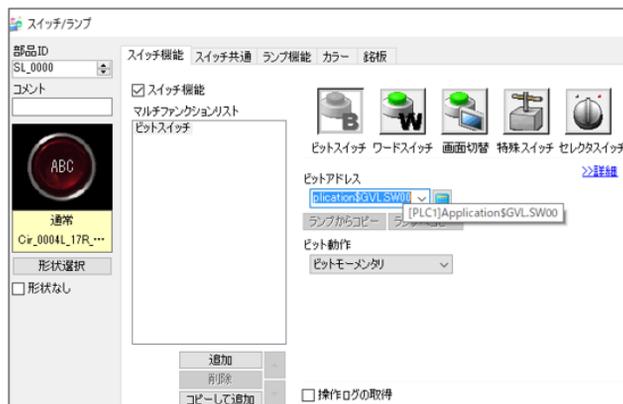
M241とPro-face製表示器を接続するには、接続機器設定と通信設定を行います。  
 接続機器設定では通信ドライバーを [CoDeSys V3 Ethernet]に設定します。通信設定では変数情報のインポートを行います。  
 詳細は6-5から6-6をご参照ください。

### (2)GP-Pro EXの設定(P1左)

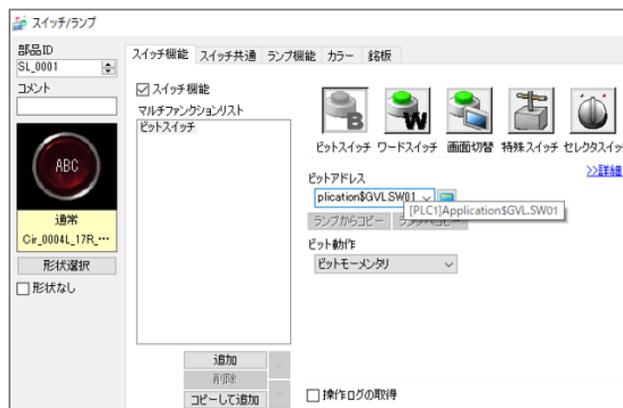
ここではP1の①から③のパーツの設定を解説します。



①スイッチ(ビットスイッチ)  
 ビットアドレス：GVL.SW00  
 ビット動作：モーメンタリー



②スイッチ(ビットスイッチ)  
 ビットアドレス：GVL.SW01  
 ビット動作：モーメンタリー



③ランプ  
 ビットアドレス：GVL.LP00



### (3)GP-Pro EXの設定(P1右)

ここではP1の⑤から⑦のパーツの設定を解説します。



①スイッチ  
ビットアドレス：GVL.SW00  
ビット動作：モーメンタリー

④ランプ  
ビットアドレス：GVL.LP01



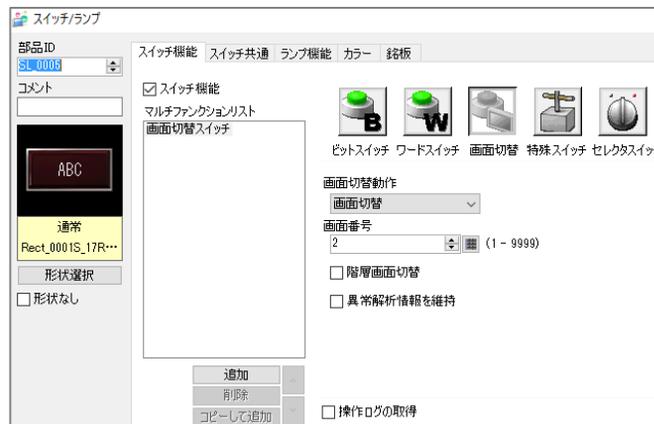
⑤データ表示器  
モニタワードアドレス：GVL.TON\_PT  
入力許可：チェックあり



⑥データ表示器  
モニタワードアドレス：GVL.TON\_ET  
入力許可：チェックなし



⑦スイッチ(画面切替)  
画面切替動作：画面切替  
画面番号：2



## (5)GP-Pro EXの設定(P2)

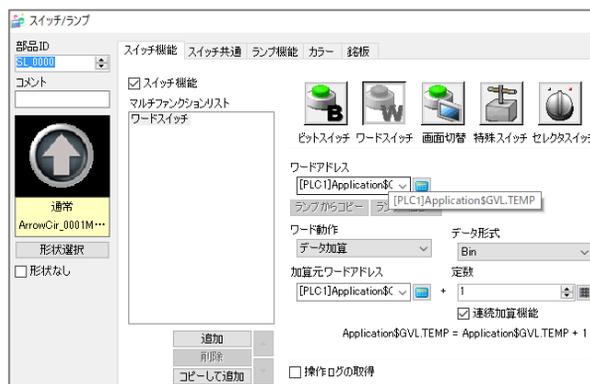
ここではP1の⑤から⑫のパーツの設定を解説します。



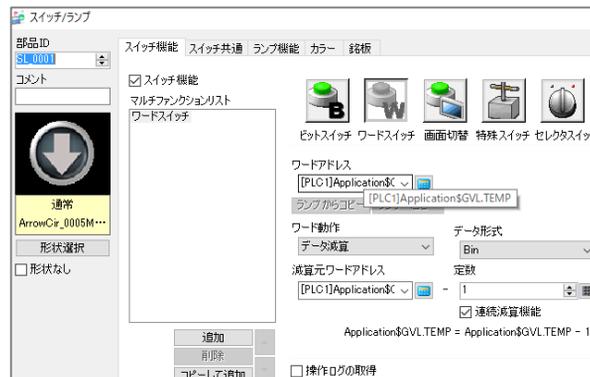
⑤データ表示器  
モニタワードアドレス：GVL.TEMP  
入力許可：チェックあり



⑨スイッチ(ワードスイッチ)  
ワードアドレス：GVL.TEMP  
ワード動作：データ加算  
加算元ワードアドレス：GVL.TEMP  
定数：1



⑩スイッチ(ワードスイッチ)  
ワードアドレス：GVL.TEMP  
ワード動作：データ減算  
減算元ワードアドレス：GVL.TEMP  
定数：1



⑪ランプ  
ビットアドレス：GVL.ALARM



⑫スイッチ(画面切替)  
画面切替動作：画面切替  
画面番号：1



# 付録

## さらに便利に使おう

1.ファームウェアのアップデート(Controller Assistant).....	付-1
2.ソースのダウンロードとアップロード.....	付-3
3.シミュレーション(ソフトウェアでのデバッグ).....	付-4
4.デバイスアドレス(%MW) .....	付-7
5.変数の種類.....	付-8
6.メモリ情報の確認.....	付-10
7.ビジュアライゼーション機能の便利な使い方.....	付-11
8.Webサーバー.....	付-12

# 1. ファームウェアのアップデート(Controller Assistant)

## (1)ファームウェアとは

ファームウェアとは、PLCのROM(読み出し専用メモリ)に記憶されている基本的な制御を行うプログラムです。ハードウェアとソフトウェアの中間的な存在になるため、ファーム(固定)ウェアと呼ばれています。

ハードウェアに最も近いソフトウェアとしてPLCの基本的な制御を行うため、他のソフトウェアのように簡単に更新することができません。

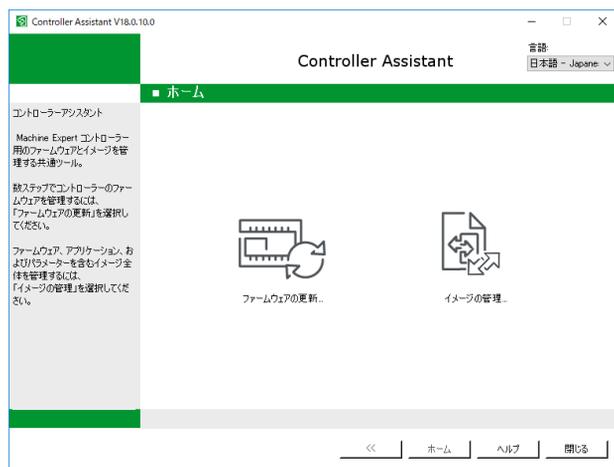
M241で、ファームウェアをアップデートするには、[Controller Assistant]を使用します。ファームウェアをアップデートすることでより快適に使用することができます。



PLC本体とEcoStruxure Machine Expertのファームウェアに相違があると、プロジェクトの転送ができません。特にPLC本体を新規に購入されたときはご注意ください。多くの場合は[Controller Assistant]を使用してファームウェアのアップデートが必要になります。

## (2)Controller Assistantとは

[Controller Assistant]とはPLC本体のファームウェアを管理するツールです。



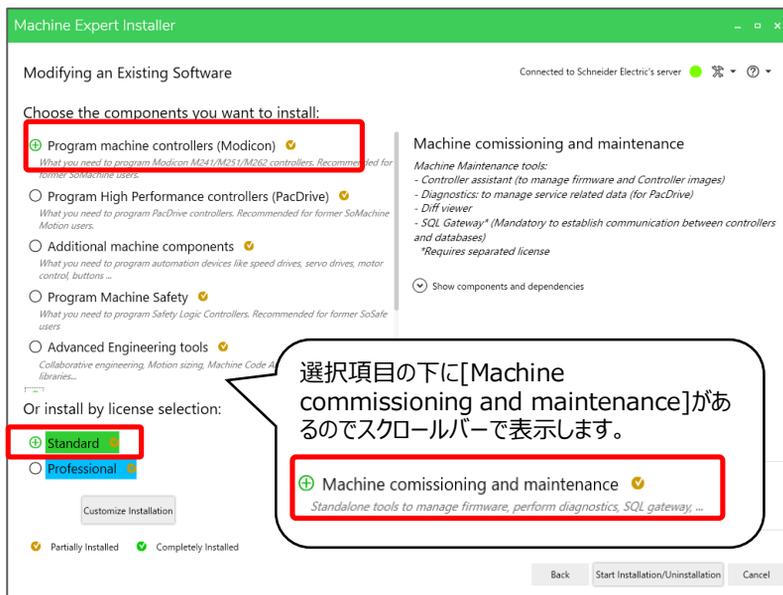
## (3)Controller Assistantのインストール方法

[Controller Assistant]をインストールするには、[Machine Expert Installer]で[Machine commissioning and maintenance]を選択し、インストールします。※自動でインストールされないので注意してください。

### Machine Expert Installer

下記の3項目に必ずチェックを入れてインストールしてください。

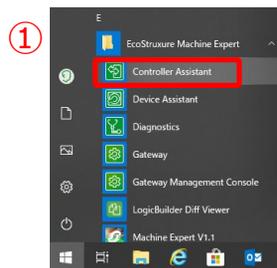
- Program machine controllers(Modicon)
- Machine commissioning and maintenance
- Standard



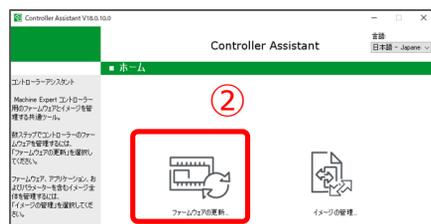
## (4)ファームウェアのアップデート方法

ファームウェアをアップデートするには、PLC本体とパソコンをUSBケーブルで接続し、作業を行います。  
 ※LANケーブルではファームウェアのアップデートはできないので注意してください。

①スタートボタンから[EcoStruxure Machine Expert] [Controller Assistant]をクリックします。



②[ファームウェアの更新]をクリックします。



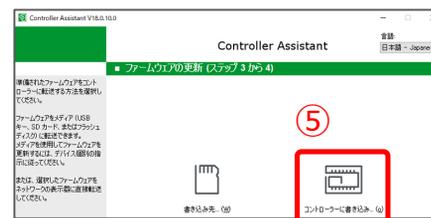
③該当する項目にチェック入れ、[次へ]をクリックします。



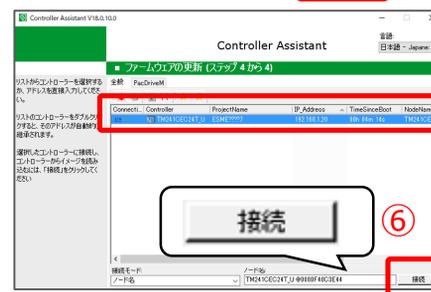
④IPアドレスを指定する場合は入力し、[次へ]をクリックします。  
 ※必須ではありません。



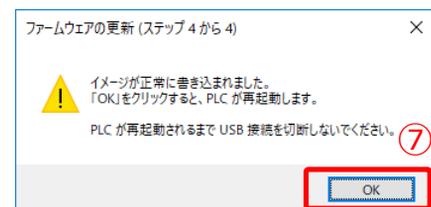
⑤[コントローラに書き込む]をクリックします。



⑥PLCを選択し、[接続]をクリックします。



⑦[ファームウェアの更新]で[OK]をクリックします。



## 2. ソースのダウンロードとアップロード

### (1) ソースとは

M241に書き込まれているデータをアップロードするためには、プロジェクトとは別にソースをあらかじめダウンロードしておきます。ソースとはアーカイブされたプロジェクトです。

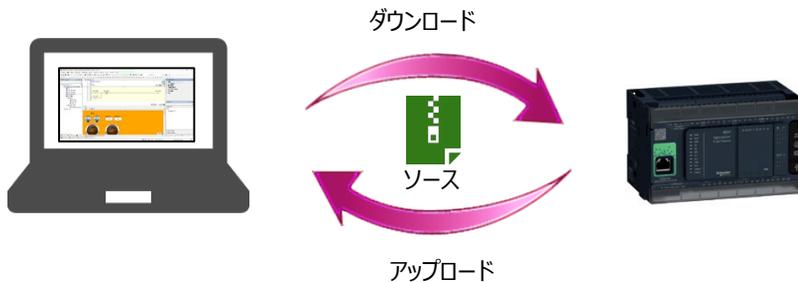
#### 【プロジェクトのダウンロード】

プロジェクトとは [プログラム][通信プロトコル][システム]を一つのまとめたものです。プロジェクトをM241にダウンロードすることで稼働させることができます。プロジェクトはダウンロードのみ可能です。プロジェクトをアップロードする場合はあらかじめソース(アーカイブしたプロジェクト)をダウンロードします。



#### 【ソースのダウンロードとアップロード】

ソースとはアーカイブされたプロジェクトです。プロジェクトをアップロードする場合はプロジェクトとは別にソースをあらかじめダウンロードしておきます。なお、M241内部ではプロジェクトとソースは別のデータとして扱われますので、プロジェクトを更新/ダウンロードしても、ソースは自動で更新されないで注意してください。



### (2) ソースのダウンロードとアップロード方法

ソースのダウンロードとアップロードはメニューバー[ファイル]から行います。

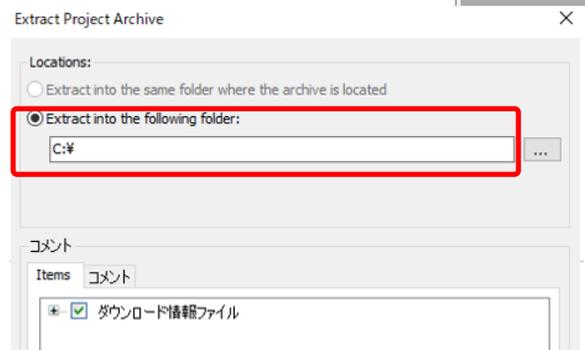
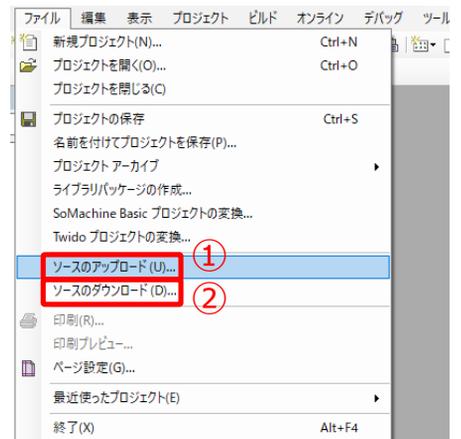
#### 【ダウンロード】

① [ソースのダウンロード]をクリックします。

#### 【アップロード】

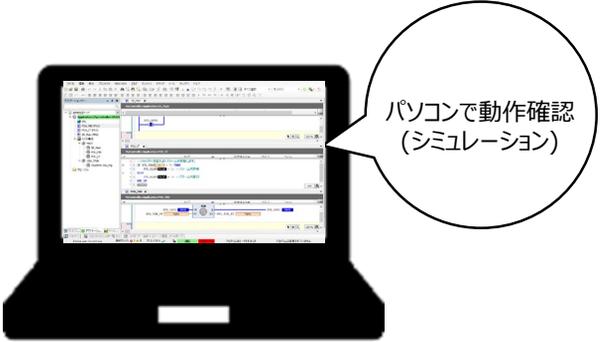
② [ソースのアップロード]をクリックします。

③ PLCとの接続方法とソースのアップロード先を指定します。



### 3. シミュレーション(ソフトウェアでのデバッグ)

シミュレーション機能を使えば、EcoStruxure Machine Expertで作成プロジェクトを、PLC本体に転送せずに、パソコン上で動作確認できます。



#### (1)シミュレーションの起動

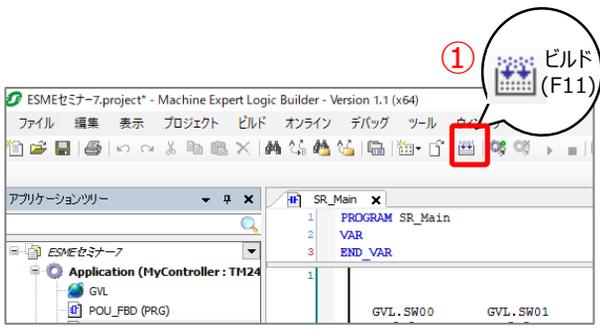
シミュレーションを行うには、ビルド(エラーチェック)後、メニューバー[オンライン][シミュレーション]にチェックを入れ、ログインします。シミュレーション時は画面下部に[シミュレーション]と表示されます。シミュレーションでは変数の状態やプログラムの動きを確認することができます。

【シミュレーションの流れ】



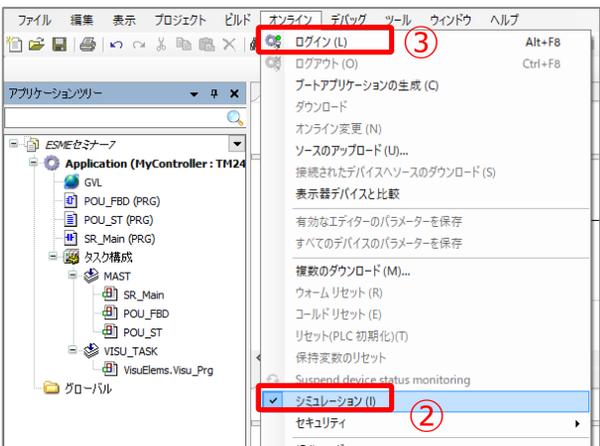
①プログラム作成後はビルドで、エラーチェックを行います。

エラー：ダウンロードを行うことができません。  
警告：ダウンロードを行うことができます(警告内容は必ず確認してください)

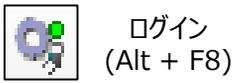


②メニューバー[オンライン][シミュレーション]にチェックを入れます。

※画面下部に **シミュレーション** と表示され、シミュレーションモードに切り替わります。



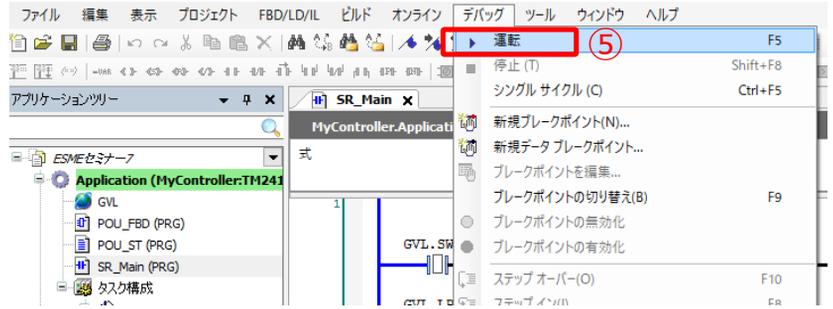
③メニューバー[オンライン][ログイン]をクリックします。(または、 をクリックか、[Alt + F8]をタッチします)



④[はい]をクリックします。

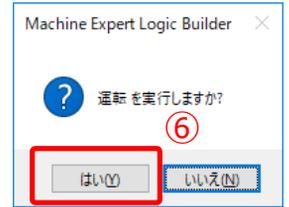


⑤メニューバー[デバッグ][運転]をクリックします。  
(または、▶ をクリックか、[F5]をタッチします)



⑥[はい]をクリックします。

※画面下部に **運転** と表示され、運転モードに切り替わります。



**シミュレーションが終了したら**

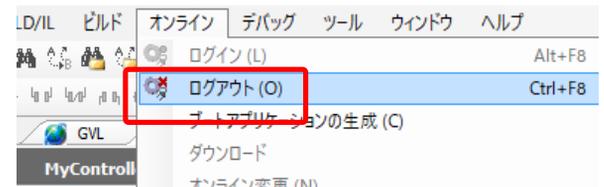
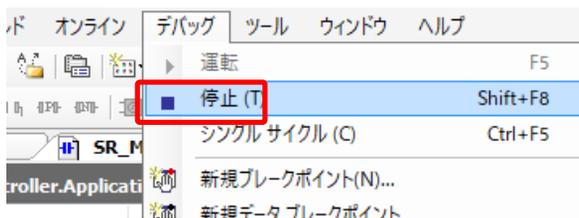
シミュレーションによるデバッグ作業が終了したら、運転を停止し、ログアウトします。

**①運転の停止**

メニューバー[デバッグ][停止]をクリックします。  
(または、■ をクリックか、[Shift+F8]をタッチします)

**②ログアウト**

メニューバー[オンライン][ログアウト]をクリックします。  
(または、❌ をクリックか、[Ctrl+F8]をタッチします)



## (2)シミュレーションの操作

シミュレーション起動時に変数を変化させるにはプログラム上で行う方法と変数一覧から行う方法があります。いずれも設定値を決め、右クリック[Write all values of ~]で書き込みます。



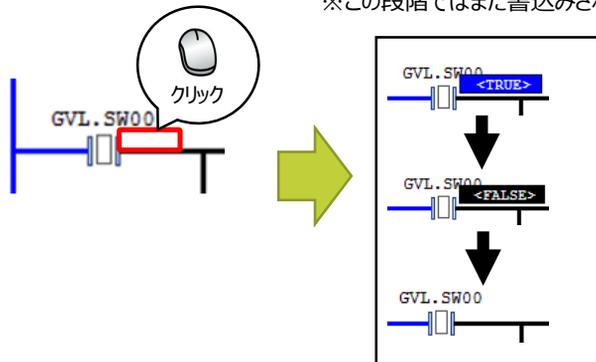
変数への書き込みはキーボード操作からでもできます。

# Ctrl + F7

### 【プログラム上で変数を変化させる方法】

ここでは、BOOL変数(GVL.SW00)をON/OFFします。

- ①変数の右下をクリックします。
- ②クリックでTRUE/FALSEが切り替えます。  
※この段階ではまだ書き込みされていません。



- ③右クリックし、[Write all values of ~]を選択します。②で設定した値を書き込みます。



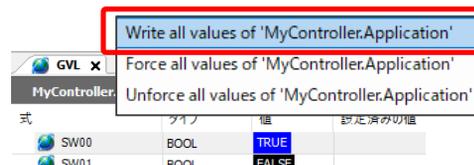
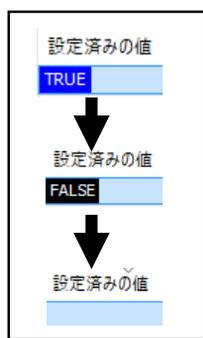
### 【変数一覧で変数を変化させる方法】

ここでは、BOOL変数(GVL.SW00)をON/OFFします。

- ①[設定済みの値]をクリックします。
- ②クリックでTRUE/FALSEが切り替えます。  
※この段階ではまだ書き込みされていません。
- ③右クリックし、[Write all values of ~]を選択します。②で設定した値を書き込みます。

式	タイプ	値	設定済みの値
SW00	BOOL	FALSE	
SW01	BOOL	FALSE	
LP00	BOOL	TRUE	
TON00	TON		
TON_ET	TIME	T#5s	
TON_PT	TIME	T#5s	
LP01	BOOL	TRUE	
TEMP	INT	0	

値は現在の変数の状態です。



CHECK!

### 強制書き込み(F7)と強制解除(Alt+F7)

変数の値を強制書き込みしたい場合は、[F7]を用います。強制書き込みした変数には **F** が表示されます。

- ①F7をタッチすると強制書き込みされます。

- ①[Alt+F7]をタッチすると強制書き込みが解除されます。

式	タイプ	値	設定済みの値
SW00	BOOL	<b>F</b> TRUE	
SW01	BOOL	FALSE	

式	タイプ	値	設定済みの値
SW00	BOOL	TRUE	
SW01	BOOL	FALSE	

## 4. デバイスアドレス(%MW)

### (1) デバイスアドレス(%MW)

EcoStruxure Machine Expertは、変数を使用してプログラミングを行うように設計されていますが、60,000ワードのデバイスアドレスも用意されています。

使用できるデバイスアドレス：**%MW0～%MW59999**

#### 【デバイスアドレスの指定方法】

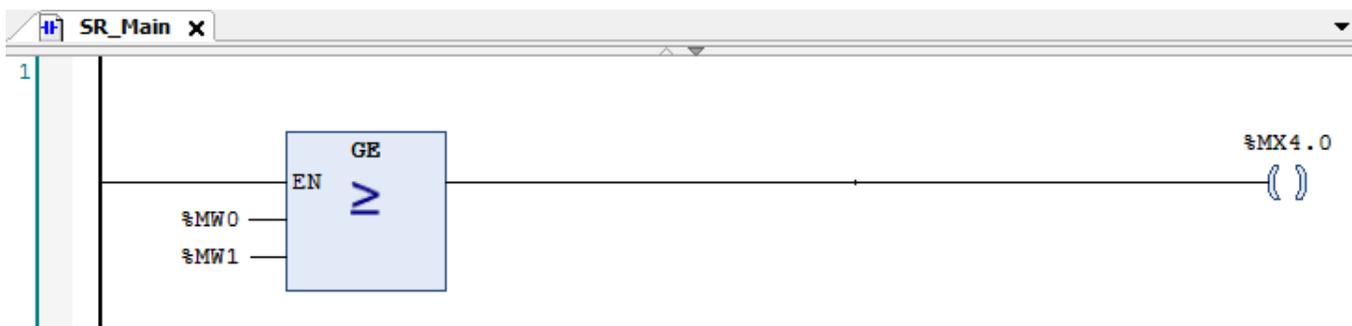
デバイスアドレスはワード単位/バイト単位/ビット単位での使用が可能です。ビット、バイト、ワード単位で分割して使用できるため、メモリの節約になります。各単位の指定方法は下記になります。

ワード指定	バイト指定	ビット指定①	ビット指定②
%MW0	%MB0	%MX0.7 ~ %MX0.0	%MX0.15 ~ %MX0.0
	%MB1	%MX1.7 ~ %MX1.0	
%MW1	%MB2	%MX2.7 ~ %MX2.0	%MX2.15 ~ %MX2.0
	%MB3	%MX3.7 ~ %MX3.0	
%MW2	%MB4	%MX4.7 ~ %MX4.0	%MX4.15 ~ %MX4.0
	%MB5	%MX5.7 ~ %MX5.0	

• • • • •

### (2) デバイスアドレスの使用例

デバイスアドレスをプログラムで使用する際は、直接デバイスアドレスを入力します。



## 5.変数の種類

### (1)変数の型

EcoStruxure Machine Expertの変数には下記の型があります。それぞれの変数には格納できる値の下限値/上限値があります

#### 【変数の型】

型	下限値	上限値	容量
BOOL	FALSE	TRUE	1 Bit
BYTE	0	255	8 Bit
WORD	0	65,535	16 Bit
DWORD	0	4,294,967,295	32 Bit
LWORD	0	$2^{64}-1$	64 Bit
SINT	-128	127	8 Bit
USINT	0	255	8 Bit
INT	-32,768	32,767	16 Bit
UINT	0	65,535	16 Bit
DINT	-2,147,483,648	2,147,483,647	32 Bit
UDINT	0	4,294,967,295	32 Bit
LINT	-263	$2^{63}-1$	64 Bit
ULINT	0	$2^{64}-1$	64 Bit
REAL	1.18E-38	3.40E+38	32 Bit
STRING	1 character	255 characters	1 character = 1 byte
WSTRING	1 character	255 characters	1 character = 1 word
TIME	-	-	32 Bit

### (2)配列変数

配列変数とは同じ変数タイプの入れ物を複数個一括登録したものです。ひとつの変数で一度に複数の入れ物が確保されるため、同種、同サイズの変数を一括作成するのに便利です。

ひとつひとつの入れ物のことは「要素」と呼び、先頭から通し番号で何番目か決めます。

#### 【スイッチが5つあり、それぞれのスイッチの変数を作成する場合】

##### ■ 配列を使用しない場合

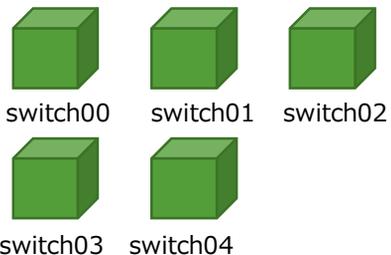
```

1 PROGRAM SR_Main
2 VAR
3   switch00: BOOL;
4   switch01: BOOL;
5   switch02: BOOL;
6   switch03: BOOL;
7   switch04: BOOL;
8 END_VAR

```

5つのBOOL変数をいちいち登録するのは面倒です。また登録ミスの恐れもあります。

##### BOOL変数のイメージ



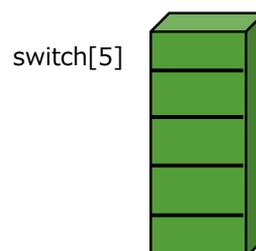
##### ■ 配列を使用する場合

```

1 PROGRAM SR_Main
2 VAR
3   switch: ARRAY[0..4] OF BOOL;
4 END_VAR

```

##### 配列変数のイメージ



【配列変数の作成方法】

変数リストで下記のように入力します。

**switch: ARRAY[0..4] OF BOOL;**

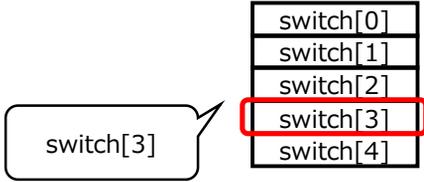
変数名                      要素数                      型

```
SR_Main x
1 PROGRAM SR_Main
2 VAR
3     switch: ARRAY[0..4] OF BOOL;
4 END_VAR
```

【配列変数の使用方法】

各要素の指定は変数名の後に [] に指定します。

例：変数名[\*]



```
SR_Main x
1 PROGRAM SR_Main
2 VAR
3     switch: ARRAY[0..4] OF BOOL;
4     lamp: ARRAY[0..4] OF BOOL;
5 END_VAR
```



配列変数の簡単な作成方法

配列変数は、変数一覧を表形式にするとウィザードで作成できるので簡単です。

【テキスト形式】

```
SR_Main x
1 PROGRAM SR_Main
2 VAR
3     switch: ARRAY[0..4] OF BOOL;
4 END_VAR
```

クリックで切替え

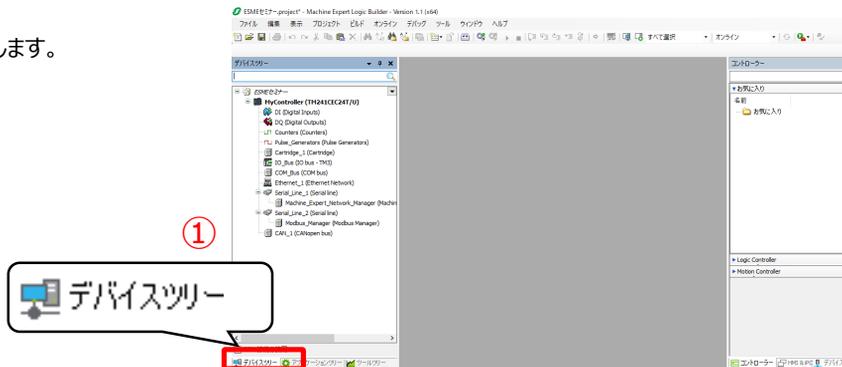
【表形式】

## 6. メモリ情報の確認

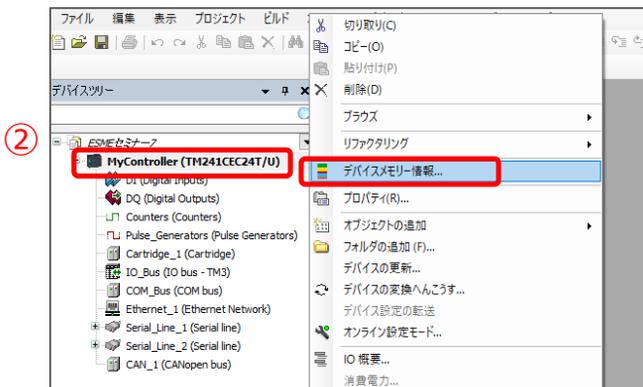
### (1) デバイスメモリ情報の確認方法

PLC本体メモリを使用状況は、[デバイスメモリ情報]で確認します。

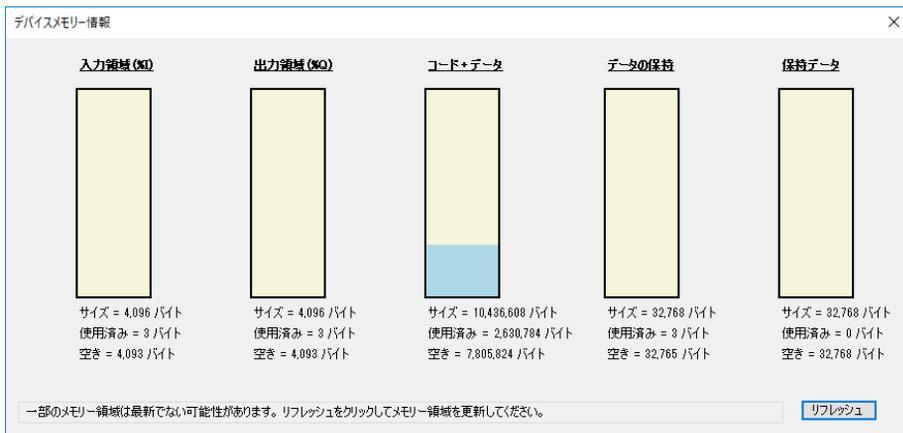
① [ナビゲーター]の左下の[デバイスツリー]をクリックします。



② デバイスツリーから[MyController]を選択し、[デバイスメモリ情報]をクリックします。



③ [デバイスメモリ情報]で使用状況を確認します。



## 7. ビジュアライゼーション機能の便利な使い方

### (1) イメージ配置

ビジュアライゼーション機能では、JPEGファイルやBMPファイル等のイメージ画像を取り込み、表示することができます。企業ロゴや装置の写真などの表示に便利です。

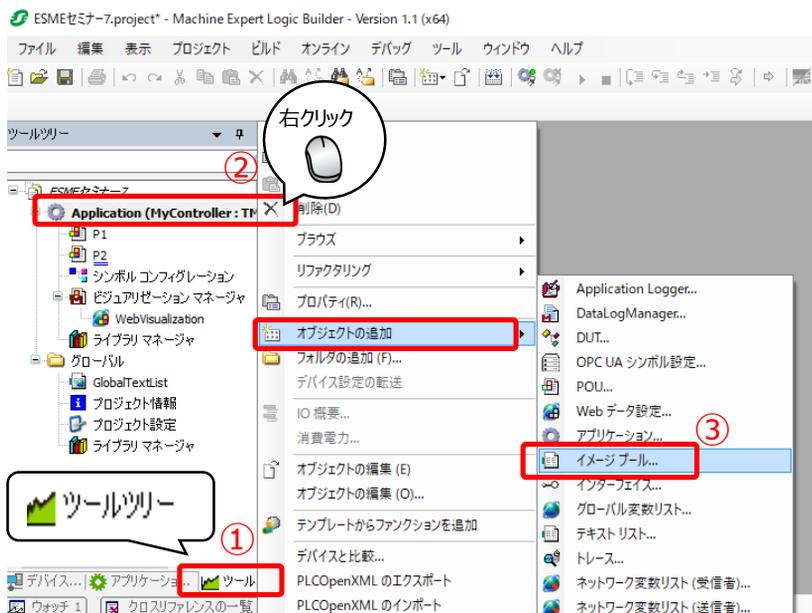
表示するには、イメージ画像をイメージプールに取り込みます。そして、ビジュアライゼーション画面で取り込んだイメージを貼り付けます。

#### 【イメージプールにイメージ画像を取り込む方法】

① [ナビゲーター]の左下の[ツールツリー]をクリックします。

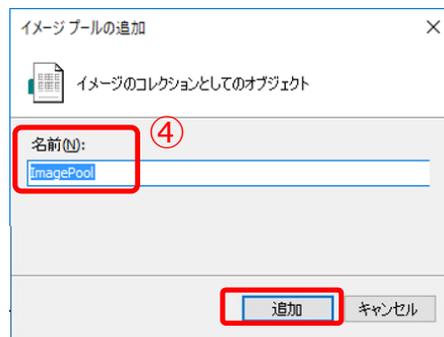
② [Application]を右クリックします。

③ [オブジェクトの追加][イメージプール]をクリックします。



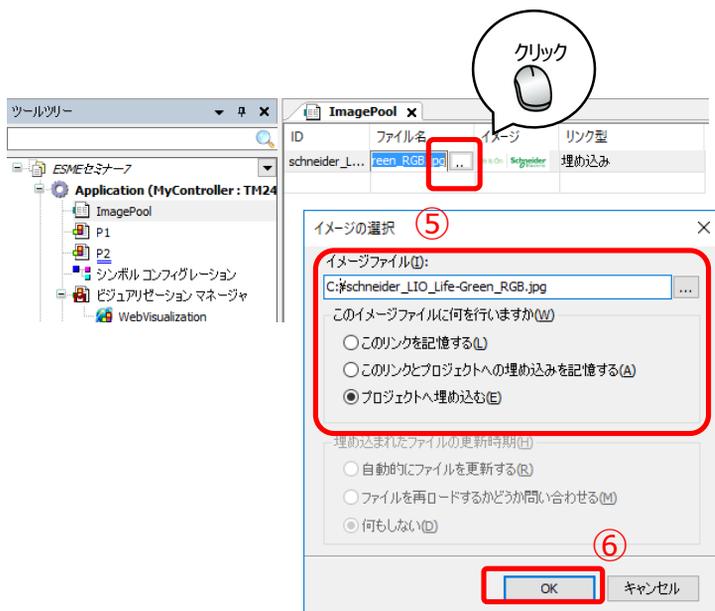
④ [イメージプールの追加]で名前をつけて[追加]をクリックします。

・名前：任意



⑤ [ImagePool]のファイルで登録するイメージを選択します。

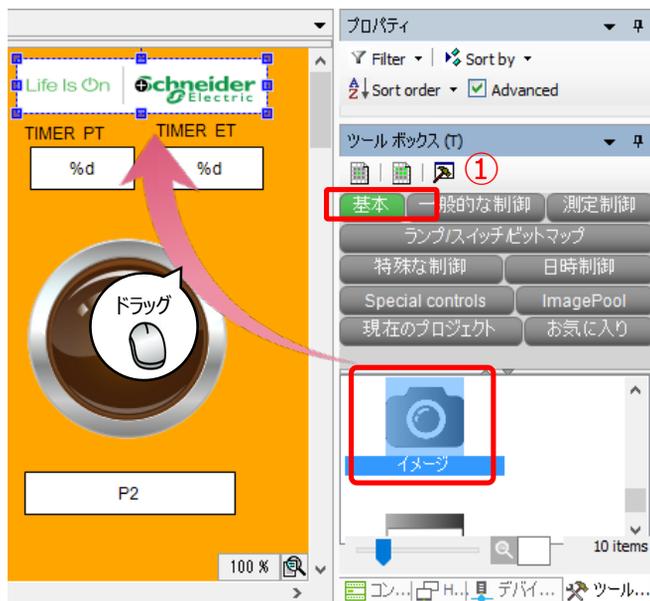
・イメージファイル：任意  
・プロジェクトに埋め込む



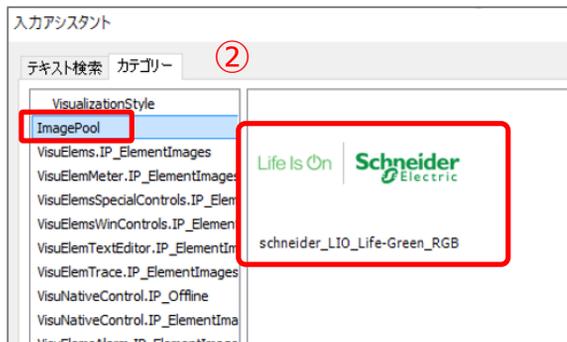
⑥ [OK]をクリックします。

## 【イメージの配置】

① [ツールボックス]から[基本]を選択し、[イメージ]をワークスペースにドラッグ&ドロップします。



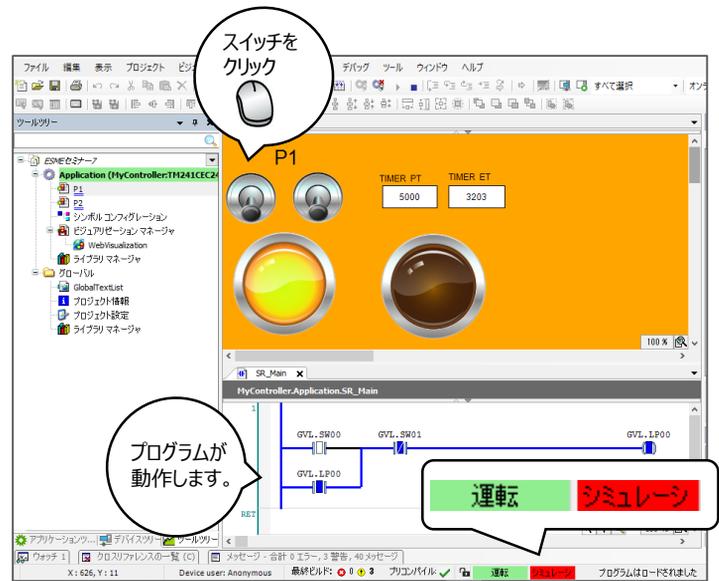
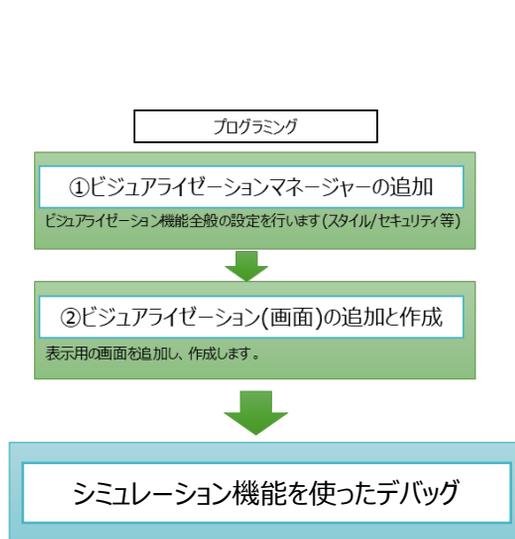
② [入力アシスタント]で[ImagePool]を選択し、表示するイメージをクリックします。



## (2)ビジュアライゼーションとシミュレーション

ビジュアライゼーション機能はシミュレーション時でも動作します。また、Webビジュアライゼーション(Webブラウザ)を使用せずに、EcoStruxure Machine Expertのみでデバッグを行うことができます。

メニューバー[オンライン]で[シミュレーション]にチェックを入れ、[ログイン]します。[運転]を開始すると、ビジュアライゼーション画面を操作することができます。



# 8. Webサーバー

## (1)Webサーバーとは

Webサーバーとは、Webブラウザを使ってPLCのデータの読み出しや簡単なメンテナンスを行うことができる機能です。WebブラウザにIPアドレスを入力するだけですぐに利用できます。

IPアドレスの  
入力のみ  
例)192.168.1.20

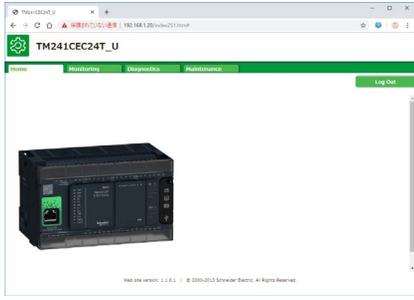


192.168.1.1  
255.255.255.0

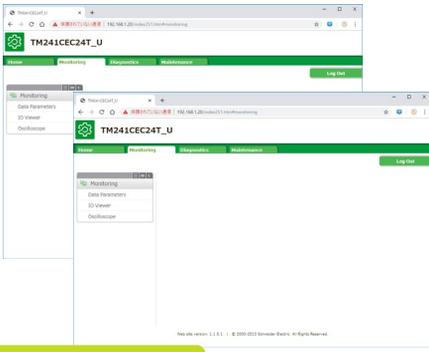
192.168.1.20  
255.255.255.0

### 【Webサーバー表示例】

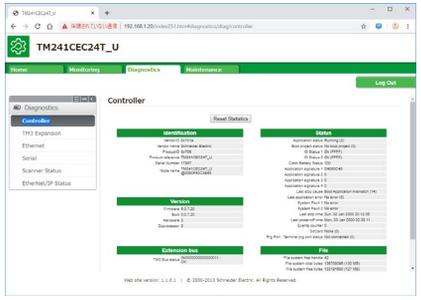
ホーム



変数値の確認/変更



PLC情報



## (2)Webサーバーの使用方法

Webサーバーを使用するには、パソコンでWebブラウザを起動し、PLCのIPアドレスを入力します。

- ①Webブラウザを起動し、IPアドレスを入力し、Enterキーをタッチします。  
※PLCで設定したIPアドレスを入力します。

IPアドレスを入力し、Enterキーをタッチ



- ②[User][Password]を入力し、[Login]をクリックします。

User : Anonymous  
Password : (空白)



- ③[Webサーバー]の初期画面が表示されます。

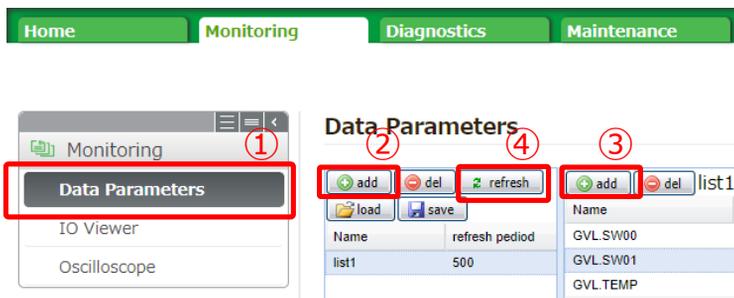


### (3)変数値の確認/変更

Webサーバー機能を使えば、変数の値の読み書きを行うことができます。また、データの推移を折れ線グラフで表示することもできます。変数値の確認/変更は[Monitoring]タブから行います。

#### 【変数値の確認/変更】

- ① [Monitoring]タグで [Data Parameters] をクリックします。
- ②  **add** をクリックし、リストを作成します。
- ③  **add** をクリックし、監視する変数を入力します。
- ④  **refresh** をクリックして、モニタリングを開始します。



Name	refresh period
list1	500

#### 【折れ線グラフ】

- ① [Monitoring]タグで [Oscilloscope] をクリックします。
- ② Item0に監視する変数を指定します。
- ③  **refresh** をクリックして、モニタリングを開始します。






発行 : 2020年1月(1.04)  
発行元 : Pro-faceトレーニングセンター  
(シュナイダーエレクトリックホールディングス株式会社)