

プログラマブルターミナル NA シリーズ

活用ガイド 画面作成 (Q & A 集)

形 NA5-15W□□□□

形 NA5-12W□□□□

形 NA5-9W□□□□

形 NA5-7W□□□□

Practices
Guide

■ はじめに

本誌は、NA の画面作成の際に参考となる内容を記載しており、安全上の要点などは掲載しておりません。実際のご利用にあたっては必ずプログラマブル ターミナル NA シリーズのユーザーズマニュアルをお取り寄せいただき、安全上の要点など、ご使用の際に必要な内容をお読みいただき、十分に動作確認を行った上でご使用ください。

- (1) 本書の内容の一部または全部を無断で転載、複製することは禁止されています。
- (2) 本書の内容に関しては、改良のため予告なしに仕様などを変更することがありますのでご了承ください。
- (3) 商標の使用
 - ・ Sysmac は、オムロン株式会社製 FA 機器製品の日本およびその他の国における商標または登録商標です。
 - ・ Windows、Visual Basic、Excel は、米国 Microsoft Corporation の、米国、日本およびその他の国における登録商標または商標です。
 - ・ その他、記載されている会社名と製品名などにつきましては、各社の登録商標または商標です。
 - ・ スクリーンショットはマイクロソフトの許可を得て使用しています。
 - ・ 本誌で使用している製品写真や図にはイメージ画像が含まれており、実物とは異なる場合があります。

ご承諾事項

「当社商品」について特別の合意がない場合には、お客様のご購入先にかかわらず、本ご承諾事項記載の条件を適用いたします。

● 定義

本ご承諾事項中の用語の定義は次のとおりです。

- ・「当社商品」:「当社」の FA システム機器、汎用制御機器、センシング機器、電子・機構部品
- ・「カタログ等」:「当社商品」に関する、ベスト制御機器オムロン、電子・機構部品総合カタログ、その他のカタログ、仕様書、取扱説明書、マニュアル等であって電磁的方法で提供されるものも含まれます。
- ・「利用条件等」:「カタログ等」に記載の、「当社商品」の利用条件、定格、性能、動作環境、取り扱い方法、利用上の注意、禁止事項その他
- ・「お客様用途」:「当社商品」のお客様におけるご利用方法であって、お客様が製造する部品、電子基板、機器、設備またはシステム等への「当社商品」の組み込み、または利用を含みます。
- ・「適合性等」:「お客様用途」での「当社商品」の(a)適合性、(b)動作、(c)第三者の知的財産の非侵害、(d)法令の遵守および(e)各種規格の遵守

● 記載事項のご注意

「カタログ等」の記載内容については次の点をご理解ください。

- ・定格値および性能値は、単独試験における各条件のもとで得られた値であり、各定格値および性能値の複合条件のもとで得られる値を保証するものではありません。
- ・参考データはご参考として提供するもので、その範囲で常に正常に動作することを保証するものではありません。
- ・利用事例はご参考ですので、「当社」は「適合性等」について保証いたしかねます。
- ・「当社」は、改善や当社都合等により、「当社商品」の生産を中止し、または「当社商品」の仕様を変更することがあります。

● ご利用にあたってのご注意

ご採用およびご利用に際しては次の点をご理解ください。

- ・定格・性能ほか「利用条件等」を遵守しご利用ください。
- ・お客様ご自身にて「適合性等」をご確認いただき、「当社商品」のご利用の可否をご判断ください。「当社」は「適合性等」を一切保証いたしかねます。
- ・「当社商品」がお客様のシステム全体の中で意図した用途に対して、適切に配電・設置されていることをお客様ご自身で、必ず事前に確認してください。
- ・「当社商品」をご使用の際には、(i)定格および性能に対し余裕のある「当社商品」のご利用、冗長設計などの安全設計、(ii)「当社商品」が故障しても、「お客様用途」の危険を最小にする安全設計、(iii)利用者に危険を知らせるための、安全対策のシステム全体としての構築、(iv)「当社商品」および「お客様用途」の定期的な保守、の各事項を実施してください。
- ・「当社」は DDoS 攻撃(分散型 DoS 攻撃)、コンピュータウイルスその他の技術的な有害プログラム、不正アクセスにより、「当社商品」、インストールされたソフトウェア、またはすべてのコンピュータ機器、コンピュータプログラム、ネットワーク、データベースが感染したとしても、そのことにより直接または間接的に生じた損失、損害その他の費用について一切責任を負わないものとします。お客様ご自身にて、①アンチウイルス保護、②データ入出力、③紛失データの復元、④「当社商品」またはインストールされたソフトウェアに対するコンピュータウイルス感染防止、⑤「当社商品」に対する不正アクセス防止についての十分な措置を講じてください。

・「当社商品」は、一般工業製品向けの汎用品として設計製造されています。従いまして、次に掲げる用途での使用は意図しておらず、お客様が「当社商品」をこれらの用途に使用される際には、「当社」は「当社商品」に対して一切保証をいたしません。ただし、次に掲げる用途であっても「当社」の意図した特別な商品用途の場合や特別の合意がある場合は除きます。

- (a) 高い安全性が必要とされる用途(例:原子力制御設備、燃焼設備、航空・宇宙設備、鉄道設備、昇降設備、娯楽設備、医用機器、安全装置、その他生命・身体に危険が及びうる用途)
- (b) 高い信頼性が必要な用途(例:ガス・水道・電気等の供給システム、24 時間連続運転システム、決済システムほか権利・財産を取扱う用途など)
- (c) 厳しい条件または環境での用途(例:屋外に設置する設備、化学的汚染を被る設備、電磁的妨害を被る設備、振動・衝撃を受ける設備など)
- (d) 「カタログ等」に記載のない条件や環境での用途

・上記の(a)から(d)に記載されている他、「本カタログ等記載の商品」は自動車(二輪車含む。以下同じ)向けではありません。自動車に搭載する用途には利用しないで下さい。自動車搭載用商品については当社営業担当者にご相談ください。

● 保証条件

「当社商品」の保証条件は次のとおりです。

・保証期間 ご購入後 1 年間といたします。

(ただし「カタログ等」に別途記載がある場合を除きます。)

・保証内容 故障した「当社商品」について、以下のいずれかを「当社」の任意の判断で実施します。

- (a) 当社保守サービス拠点における故障した「当社商品」の無償修理
(ただし、電子・機構部品については、修理対応は行いません。)
- (b) 故障した「当社商品」と同数の代替品の無償提供

・保証対象外 故障の原因が次のいずれかに該当する場合は、保証いたしません。

- (a) 「当社商品」本来の使い方以外のご利用
- (b) 「利用条件等」から外れたご利用
- (c) 本ご承諾事項「ご利用にあたってのご注意」に反するご利用
- (d) 「当社」以外による改造、修理による場合
- (e) 「当社」以外の者によるソフトウェアプログラムによる場合
- (f) 「当社」からの出荷時の科学・技術の水準では予見できなかった原因
- (g) 上記のほか「当社」または「当社商品」以外の原因(天災等の不可抗力を含む))

● 責任の制限

本ご承諾事項に記載の保証が、「当社商品」に関する保証のすべてです。

「当社商品」に関連して生じた損害について、「当社」および「当社商品」の販売店は責任を負いません。

● 輸出管理

「当社商品」または技術資料を、輸出または非居住者に提供する場合は、安全保障貿易管理に関する日本および関係各国の法令・規制を遵守ください。お客様が法令・規則に違反する場合には、「当社商品」または技術資料をご提供できない場合があります。

目次

ご承諾事項	3
関連マニュアル	6
1 概要	7
1-1 Q&A 一覧	7
2 Q&A	9
2-1 システム設定	9
2-2 部品共通機能	11
2-3 部品個別機能	15
2-4 その他	33
3 改定履歴	47

関連マニュアル

本資料に関連するマニュアルは以下の通りです。

Man.No.	形式	マニュアル名称
SBCA-470	形 SYSMAC-SE2□□□	Sysmac Studio Version 1 オペレーションマニュアル
SBSA-545	形 NA5-15W□□□□ 形 NA5-12W□□□□ 形 NA5-9W□□□□ 形 NA5-7W□□□□	プログラマブルターミナル NA シリーズ ユーザーズマニュアル ハードウェア編
SBSA-546	形 NA5-15W□□□□ 形 NA5-12W□□□□ 形 NA5-9W□□□□ 形 NA5-7W□□□□	プログラマブルターミナル NA シリーズ ユーザーズマニュアル ソフトウェア編
SBSA-547	形 NA5-15W□□□□ 形 NA5-12W□□□□ 形 NA5-9W□□□□ 形 NA5-7W□□□□	プログラマブルターミナル NA シリーズ ユーザーズマニュアル デバイス接続編
SBSA-548	形 NA5-15W□□□□ 形 NA5-12W□□□□ 形 NA5-9W□□□□ 形 NA5-7W□□□□	プログラマブルターミナル NA シリーズ 導入ガイド

1 概要

この章では、本テキストに記載されている、NA シリーズで画面を作成する際の疑問点と回答の一覧を記載されています。各項目の詳細については第 2 章以降を参照してください。

1-1 Q&A 一覧

項目	回答	詳細項目
オブジェクトのプロパティ・アニメーション・イベントとアクションの設定欄を表示したい	上部メニュー[表示]から表示したい項目を選択します。	2-1-2 プロパティタブの表示方法
SysmacStudio 上で各機能・部品の項目の説明を表示させたい	右クリックメニューの「説明表示」を選択することで説明の表示/非表示を切り替えることができます。	2-1-3 オブジェクトの説明欄の表示方法
ボタン等の入力部品を無効化したい(NS での制御フラグ機能)	[プロパティ]タブ[動作]-[IsEnabled]のチェックを外します。	2-2-1 入力許可設定
ボタン等の入力部品を入力許可設定を間接指定したい(NS での制御フラグ機能)	[アニメーション]タブから[Animations]-[Enable]を選択して[Expression]に間接指定条件を設定してください。	2-2-1 入力許可設定
部品の表示/非表示設定を行いたい(NS での制御フラグ機能)	[プロパティ]タブを開き、[外観]-[IsVisible]のチェックを外します。	2-2-2 表示/非表示設定
部品を点滅させたい	[アニメーション]タブから[Animations]-[Blink]を選択して[Expression]に間接指定条件を設定してください。	2-2-3 部品の点滅設定
変数を設定する欄で[Variable]と[Expression]、[Value]が混在しているが、こういった違いがあるのか	[Variable]は変数のみ設定が可能です。 [Expression]、[Value]は変数の値に加え、演算結果を表示させることができます。	2-2-4 Variable と Expression・Value の違い
NS の機能部品が NA ではどのオブジェクトに該当するかを知りたい	本テキスト 2-3-1 NS-NA 機能部品対応表 を参照してください。	2-3-1 NS-NA 機能部品対応表
画像ファイルとイメージオブジェクトとの縦横比を揃えたい	[プロパティ]タブ[外観]-[LockAspectRatio]のチェックを外します。	2-3-2 イメージオブジェクトの縦横比の自動調整
変数に対して固定値を入力する部品を作成したい(NS のワードボタン: 値セット)	[イベントとアクション]タブ[Actions]から「Set Variable」を選択し、入力する値を設定してください。	2-3-3 固定値の入力・値の加減算の設定方法
変数に対して値の加算を行いたい(NS のワードボタン: 値増減)	[イベントとアクション]タブ[Actions]から「IncreaseVariable」を選択し、加算する値を設定してください。	2-3-3 固定値の入力・値の加減算の設定方法
変数に対して値の減算を行いたい(NS のワードボタン: 値増減)	[イベントとアクション]タブ[Actions]から「DecreaseVariable」を選択し、減算する値を設定してください。	2-3-3 固定値の入力・値の加減算の設定方法
NS シリーズのマルチファンクションに該当する部品を作成したい	[イベントとアクション]に複数の項目を設定することができます。	2-3-4 マルチファンクションの設定方法
日付時刻のフォーマットについて知りたい。	Visual Basic の時刻フォーマットに準じています。ユーザーが任意で設定する事も可能です。	2-3-5 日付時刻表示のフォーマット
ボタンオブジェクトの状態変化を押下時だけでなく、アドレスの値によっても行いたい	ボタンとランプを組み合わせる必要があります。この時、ボタンは透明にして、外観はランプ側で設定します。	2-3-6 ボタンオブジェクトの状態変化
NS シリーズのメッセージ表示器に該当するオブジェクトはどれか	データランプが該当する。NS シリーズと異なり、状態毎にフォントのサイズや種類を変更することができません。	2-3-7 メッセージ表示器の作成方法
ランプ部品のグラデーションを削除したい (下図ランプ部品の右上光沢部分)	ランプ部品からグラデーションを削除することはできませんが、図形オブジェクトの[アニメーション]タブから[ColorChange]の設定を行うことでランプ機能を一部代替可能です。	2-3-8 グラデーションがかかっていないランプ部品の作成方法
数値表示部品にて小数点表示をしたい (NS でのスケール設定)	[プロパティ]タブ[動作]-[Expression]に計算式を記載してください。	2-3-9 整数型数値の小数点表示



項目	回答	詳細項目
発生したアラームイベントを選択時に表示されるポップアップを表示させないようにしたい	ユーザーアラームの「ポップアップ」のチェックを外します。	2-3-10 アラームオブジェクト
アラームがデフォルト設定では発生と解除の区別がつかない	[プロパティ]タブ[外観]-[ColumnTitles], [ColumnOrder], [ColumnWidths]に「Status」を追加します。	2-3-10 アラームオブジェクト
NS でのプロジェクトロード時マクロの設定方法	[グローバルイベント]からプロジェクト起動時条件である、「ProjectInitialization」に対して作成したサブルーチンを設定してください。	2-4-1 システム設定におけるサブルーチン(マクロ)の呼び出し方
NS でのアラームイベント発生/解除時マクロの設定方法	各ユーザーアラームのイベントとアクションに対して作成したサブルーチンを設定してください。	2-4-1 システム設定におけるサブルーチン(マクロ)の呼び出し方
NS での値変化時マクロの設定方法	[グローバルイベント]から値変化時条件である、「Condition」に対して作成したサブルーチンを設定してください。	2-4-1 システム設定におけるサブルーチン(マクロ)の呼び出し方
CJ シリーズ CPU と接続することは可能か？	接続対象の CPU を外部デバイスとして登録することで接続可能。現状は Ether 接続のみ対応(NA 本体がシリアル対応していない)	2-4-2 CJ シリーズ CPU との接続方法
PLC 側から画面を切り替えることは可能か？また、NA から現在の画面を PLC へ通知することは可能か？	NJ 側で String 型変数を用意して、その変数を現在の画面を表すシステム変数である「_HMI_CurrentPage」に割り当てます。システム変数へ割り当てた変数に移動先画面名を入力することで画面移動ができます。また、NS シリーズのようにアドレスの値と画面番号が直結している場合は、Visual Basic を記載する必要があります。	2-4-3 PLC 変数から画面を切り替える方法

2 Q&A

この章では、NA シリーズで画面を作成する際の疑問点と回答を、システム設定・部品共通設定・部品個別設定の 3 点に分類して記載しています。

2-1 システム設定

2-1-1 NS と NA の相違点

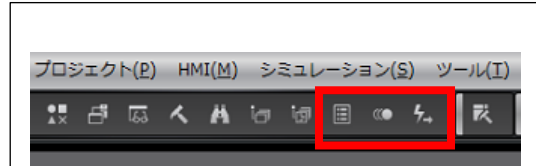
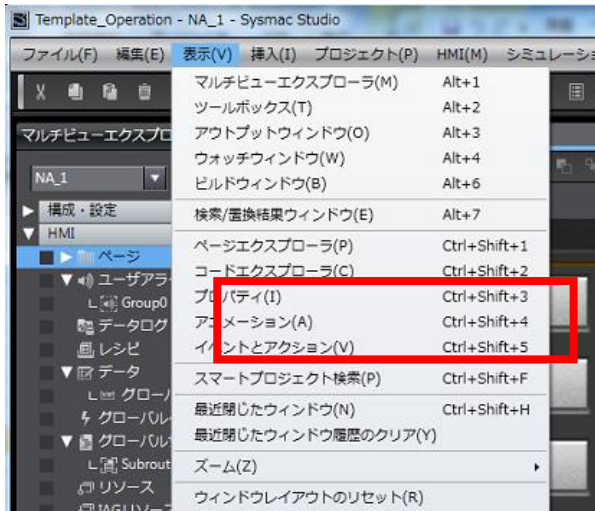
NS と NA の主な相違点を以下に記します。

項目	NS	NA
1 画面に適用できるシート画面の最大数 ※NA ではバックグラウンド画面と呼称します	10 枚	1 枚 ただし、NA ではシートを適用しているベース画面をさらにシートとして使用可能なので、結果として複数のシートを適用することができます。
PT 内部時刻の設定	システムメニューの他、日付・時刻部品やシステムメモリから入力が可能です。	日付・時刻部品から入力ができず、システムメニュー内から設定を行う必要があります。日時を表すシステム変数は用意されていますが、読込専用属性です。
システムテンキー	文字列入力テンキーでは日本語(漢字含む)が可能です。	現時点では文字列入力テンキーでは日本語の入力は不可能です。ユーザーテンキーの作成が必要です。 9 月に予定されているバージョンアップで日本語入力機能が実装されます。
ユーザー作成テンキー	コマンドボタンもしくはマルチファンクションの[キーボタン]機能を使用することで比較的容易にユーザーがテンキーを作成することが可能です。	ユーザーがテンキーを作成することは可能ですが、Visual Basic の使用が必須です。9 月に予定されているバージョンアップでシステムテンキーの編集機能が実装されます。
ポップアップ表示数	同時に最大 3 画面まで表示可能です。	1 画面のみ表示可能です。
マクロ ※NA ではサブルーチンと呼称します	NS 独自のスクリプトを使用します。	Visual Basic 2008 を採用しています。一部機能の制限及び拡張が行われています。

2-1-2 プロパティタブの表示方法

デフォルトではプロパティ等のタブは非表示となっています。これらのタブを表示させるには、ツールバーの[表示]から表示したい項目を選択する必要があります。(下図赤枠部)

また、SysmacStudio V1.11 以降からはオブジェクトをダブルクリックすることで、プロパティを表示させることができます。



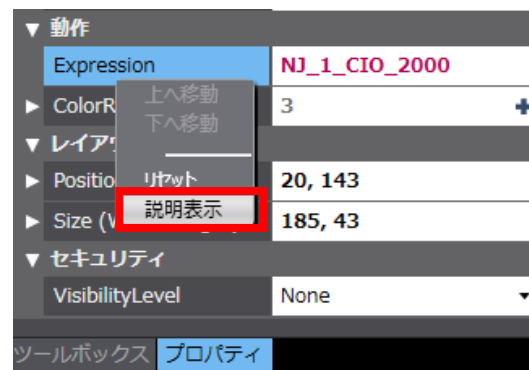
ツールバー下のアイコン(上図赤枠部)からも各タブの表示が可能です。

左からプロパティ、アニメーション、イベントとアクションを表示するアイコンです。

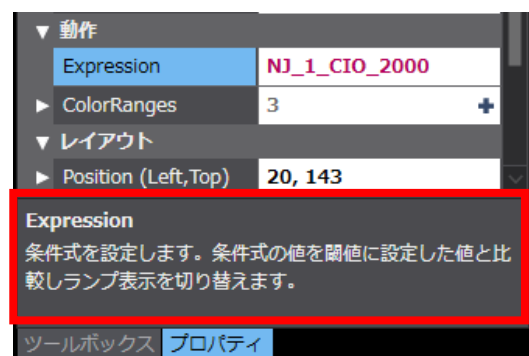
2-1-3 オブジェクトの説明欄の表示方法

デフォルトではオブジェクトの項目説明欄は非表示となっています。説明欄を表示させるには、右クリックメニューから「説明表示」を選択します。

1. 右クリックメニューから「説明表示」を選択します。



2. タブの下部の項目の説明欄が表示されます。再度手順 1 の操作を行うことで説明欄が非表示になります。

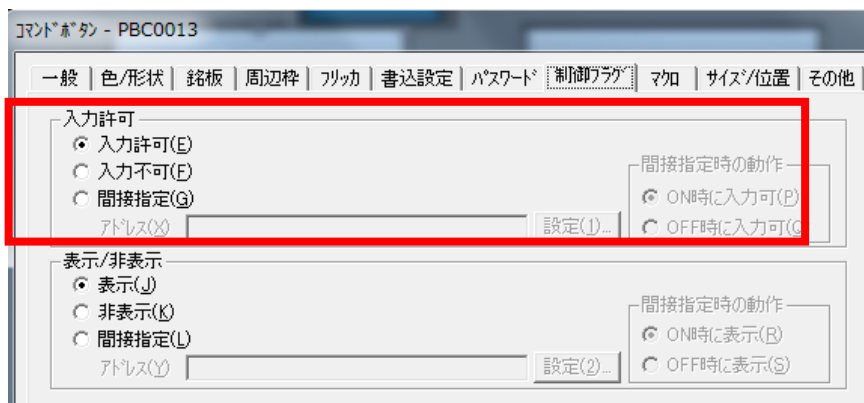


2-2 部品共通機能

本項では複数のオブジェクトに共通して存在する機能について記載します。

2-2-1 入力許可設定

NS シリーズでの入力許可の設定方法について記載します。



1. 入力許可設定を行う部品を選択して【プロパティ】タブを開き、【動作】-【IsEnabled】から入力許可/不可の設定を行います。チェック有が入力許可、チェック無が入力不可です。

▼ 動作	
Variable	Sample1
IsEnabled	☒
DoubleTouchTime	0
DelayTime	0

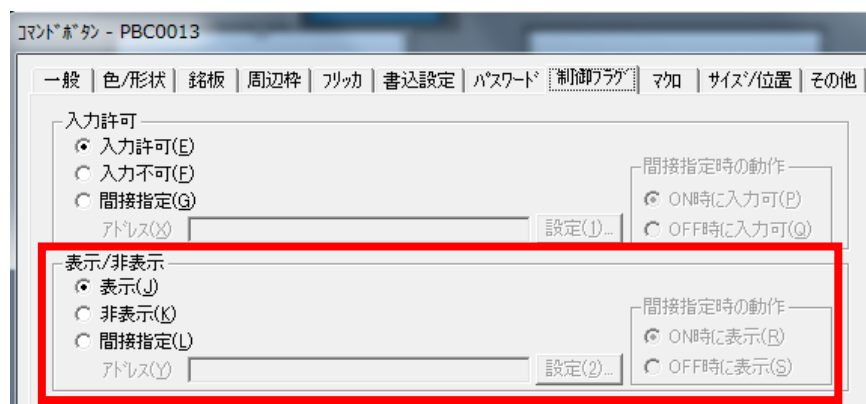
2. 変数の状態によって入力許可/不可を制御する場合は、対象の部品を選択して【アニメーション】タブを開き、【Animations】から【Enable】を選択して【Expression】に入力許可になる条件を設定します。
NS シリーズではビットアドレスの ON/OFF でしか設定ができませんでしたが、NA シリーズでは数値による条件設定も可能となっています。

アニメーション	
ToggleButton0	
▼ Animations	< 追加するアニメーションを選択します。 >
▼ [0]	Enable
Expression	Sample2=True

アニメーション	
ToggleButton0	
▼ Animations	< 追加するアニメーションを選択します。 >
▼ [0]	Enable
Expression	SampleA>=10

2-2-2 表示/非表示設定

NS シリーズでの表示/非表示の設定方法について記載します。



1. 設定を行う部品を選択して[プロパティ]タブを開き、[外觀]-[IsVisible]から表示/非表示の設定を行います。チェック有が表示、チェック無が非表示です。
NS シリーズが非表示設定にしても部品の機能が生きていたのに対し、NA シリーズでは非表示部品はタッチしても動作しない点に注意してください。

TextColorButtonUp	Black
TextColorButtonDown	Black
BackgroundColorButtonU	Linear
BackgroundColorButtonD	Linear
BorderColorButtonUp	Linear
BorderColorButtonDown	Linear
BorderThickness	2
CornerRadius (X,Y)	0, 0
IsVisible	☒

2. 非表示かつ、オブジェクトの機能も生かしておきたい場合、右図赤枠部の部品色を設定する箇所をすべて「Transparent」に設定してください。

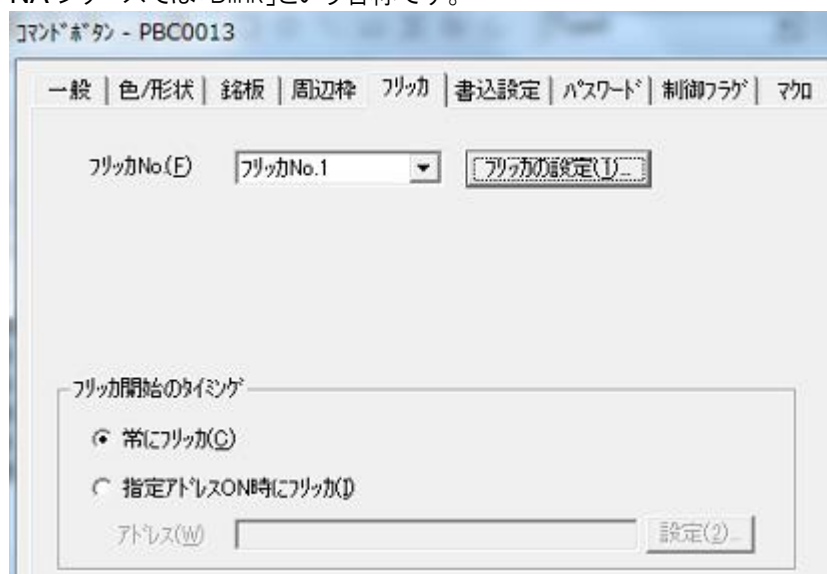
TextColorButtonUp	Transparent
TextColorButtonDown	Transparent
BackgroundColorButtonU	Transparent
BackgroundColorButtonD	Transparent
BorderColorButtonUp	Transparent
BorderColorButtonDown	Transparent

3. 変数の状態によって表示/非表示を制御する場合は、対象の部品を選択して[アニメーション]タブを開き、[Animations]から「Visibility」を選択して[Expression]に部品を表示する条件を設定します。

アニメーション	
ToggleButton1	
Animations	< 追加するアニメーションを選択します。 >
[0]	Visibility
Expression	Sample2=False

2-2-3 部品の点滅設定

NS シリーズではフリッカと呼ばれている機能の設定方法について記載します。
NA シリーズでは「Blink」という名称です。



1. 設定を行う部品を選択して[アニメーション]タブを開き、[Animations]から「Blink」を選択してください。



2. [Expression]に点滅を行う条件をを指定し、[Color]には点滅時の部品色を設定します。
NS シリーズのように点滅速度を変更することはできません。



3. 常時点滅する状態を作成する場合は、[Expression]に「True」を選択してください。



2-2-4 Variable と Expression ・ Value の違い

オブジェクトに対して変数を割り当てる欄の名称が、部品や項目によって[Variable]と表記されている項目と、[Expression]と表記されている項目、[Value]と表記されている項目の3種類あります。

[Variable]と表記されている項目は、データ入力やボタンといった、値を書込むオブジェクトで主に使用されています。本欄に設定できるのは変数のみです。

[Expression]と表記されている項目は、データ表示やランプといった、値を読み込むオブジェクトの起動条件や表示対象を設定する欄で主に使用されています。本欄には変数の他、演算式を設定することが可能です。

[Value]と表記されている項目は、値を書込む項目で固定値もしくは変数の値を書込む用途で使用されています。[Variable]が変数しか設定できないのに対して、演算式も記載できるので、変数値に固定値を加算した値を書込むといった使い方も可能です。

Variable: 変数のみ設定可能。

IsEnabled	☒
DataType	Numeric
Variable	SampleA

Expression: 変数に加え、演算式も記載可能。

DataType	Numeric	DataType	Numeric
Expression	SampleA	Expression	SampleA /10

Value: Expressionと同様に、変数に加えて演算式も記載可能。

※下図は変数 SampleD に対して、変数 SampleB と SampleC の値を加算した値を格納する。

Events	< 追加するイベントを選択します。 >
▼ [0]	Press
▼ Actions	< 追加するアクションを選択します >
▼ [0]	SetVariable
Variable	SampleD
Value	SampleB+SampleC

[Expression]は[プロパティ]タブ内の項目、[Value]は[イベントとアクション]タブ内の項目に主に使用されます。
左図はラベルオブジェクトの[イベントとアクション]タブに設定しています。

Expression や Value の演算式には四則演算の他、べき乗と剰余が使用できます。

入力値		Expression	表示結果
SampleA	####	SampleA	####
SampleB	####	SampleA/10	####
SampleC	####	SampleA*10	####
		SampleA+1	####
		SampleA-1	####
		SampleA^2 (べき乗)	####
		SampleA mod3(剰余)	####
SampleB+SampleC			
	####		

2-3 部品個別機能

本項では部品の個別機能について記載します。

2-3-1 NS - NA 機能部品対応表

NS の機能部品が、NA ではどのオブジェクトに該当するかを下表に記します。

NS 機能部品	NA オブジェクト	備考
ON/OFF ボタン (モーメンタリ)	[ボタン]-[モーメンタリボタン]	
ON/OFF ボタン (オルタネート)	[ボタン]-[トグルボタン]	
ON/OFF ボタン (セット)	[ボタン]-[セットボタン]	
ON/OFF ボタン (リセット)	[ボタン]-[リセットボタン]	
ワードボタン (値セット)	イベントとアクションから[SetVariable]を選択して入力値を設定する	ボタンに限らず、ラベルや図形といった部品に対してもイベントとアクションを設定可能です。 2-3-3 固定値の入力・値の加減算の設定方法 を参照してください。
ワードボタン (値の増減)	イベントとアクションから[IncreaseVariable]もしくは[DecreaseVariable]を選択し、値を設定する。	ボタンに限らず、ラベルや図形といった部品に対してもイベントとアクションを設定可能です。 2-3-3 固定値の入力・値の加減算の設定方法 を参照してください。
ワードボタン (ポップアップメニューの表示)	[標準コントロール]-[ドロップダウン]	
コマンドボタン (画面切替)	イベントとアクションから[ShowPage]を選択して移動先画面を決定する	ボタンに限らず、ラベルや図形といった部品に対してもイベントとアクションを設定可能です。
コマンドボタン (操作画面戻す)	イベントとアクションから[ShowPreviousPage]を選択する	ボタンに限らず、ラベルや図形といった部品に対してもイベントとアクションを設定可能です。
コマンドボタン (キーボタン)	-	部品単位では、代替手段は現状ありません。Visual Basic を使用することで一部機能の再現が可能です。
コマンドボタン (ポップアップ画面制御 -ポップアップ閉)	イベントとアクションから[ClosePage]を選択して、閉じる対象の画面名を設定する。	
ビットランプ	[ランプ]-[ビットランプ]	
ワードランプ	[ランプ]-[データランプ]	
マルチファンクション	-	特定の部品としては存在しません。イベントとアクションを複数設定することで、1 部品で複数機能が実現できます。 2-3-4 マルチファンクションの作成方法 を参照してください。
ラベル	[標準コントロール]-[ラベル] or [テキストボックス]	1 行のみならラベル。複数行表示するならテキストボックスを使用します。 イベントとアクションの設定ができるので、NS と異なり、変数に値を書込むなどの入力操作を行うことができます。
メッセージ表示器(ラベル)	[ランプ]-[データランプ]	NS と異なり、状態毎のフォント種類・サイズは変更られません。 背景色・文字色は変更可能です。 2-3-7 メッセージ表示器の作成方法 を参照してください。

NS 機能部品	NA オブジェクト	備考
数値表示入力 or 文字列表示入力(入力可能)	[標準コントロール]-[データ入力]	数値部品と文字列部品はデータ入力部品及び データ表示部品として分離・統合されました。
数値表示入力 or 文字列表示入力(入力不可)	[標準コントロール]-[データ表示]	
リスト選択	[標準コントロール]-[リストボックス]	
サムロータリスイッチ	-	部品単位では代替手段がありません。数値部品 と、ボタン内イベントとアクション等を組み合わせ ることで類似機能を作成可能です。
アナログメータ	ゲージ	
レベルメータ	ゲージ	
折れ線グラフ	現状該当部品・機能なし。次回 VerUp 対 応予定	
ビットマップ	[標準コントロール]-[イメージ]	
アラーム/イベント表示	該当部品・機能なし	
アラーム/イベントサマリ・履歴	ユーザーアラームビューワ	NS ではグループ・表示種別のみを指定して表示 することができましたが、現時点の NA ではできま せん。
日付	[標準コントロール]-[日付時刻表示]	日付のみ表示も可能です。 NS では部品から直接日付設定ができたが、NA ではシステムメニューからしか設定できません。
時間	[標準コントロール]-[日付時刻表示]	時刻のみ表示も可能です。 NS では部品から直接時刻設定ができたが、NA ではシステムメニューからしか設定できません。
データロググラフ	[HMI コントロール]-[トレンドグラフ]	
データブロックテーブル	[HMI コントロール]-[レシビューワ]	
ビデオ表示	[HMI コントロール]-[メディアプレーヤー]	
テンポラリ入力	該当部品・機能なし	
連続線描画	[図形]-[連続直線]	
コンテンツ表示	[ランプ]-[データランプ]	データランプで類似の機能を作成できます。 状態毎に文字の色やフォントを変更できない等と いった制約があります。
フレーム	現状該当部品・機能なし。次回 VerUp 対 応予定	

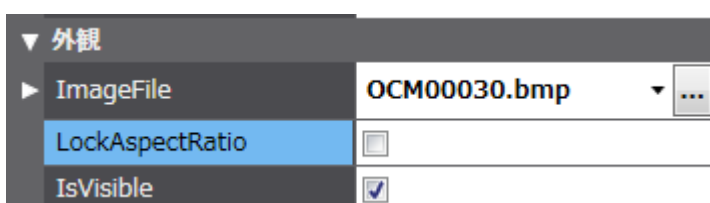
2-3-2 イメージオブジェクトの縦横比の自動調整

ボタン・ランプオブジェクトに外部画像を使用すると、部品と画像の縦横比が自動的に調整されるのに対して、イメージオブジェクトのデフォルト設定では画像のサイズがそのまま表示されます。縦横比の自動調整を行う手順を以下に記載します。

1. デフォルト設定ではイメージオブジェクトに画像を設定すると、縦のみサイズがあっている形になり、横は空白ができてしまいます。



2. イメージオブジェクトを選択して[プロパティ]タブを開き、[外観]-[LockAspectRatio]のチェックを外します。



3. イメージオブジェクトに対して画像の縦横比が一致します。



2-3-3 固定値の入力・値の加減算の設定方法

NS シリーズで変数・アドレスへ固定値の入力や値の加算/減算を行う場合はワードボタンを使用していました。

NA シリーズでは、イベントとアクションを使用します。

● 固定値の入力

1. ツールボックスから[ボタン]-[ボタン]を選択し、ページ上にドラッグ&ドロップしてください。



2. イベントとアクションから[Events]に「Click」、[Actions]に「Set Variable」をそれぞれ選択し、[Variable]に入力対象の変数、[Value]に入力値を設定します。

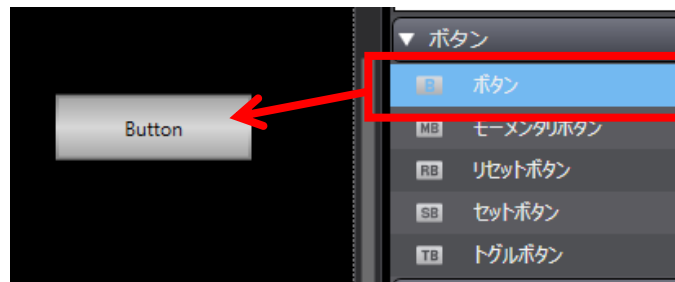
▼ Events	< 追加するイベントを選択します。 >
▼ [0]	Click
▼ Actions	< 追加するアクションを選択します。 >
▼ [0]	SetVariable
Variable	SampleA
Value	10

3. [Value]に固定値ではなく変数を入れることで、入力値を間接指定とすることも可能です。
右図の設定では、変数 SampleC の値が SampleB へ入力されます。

▼ Events	< 追加するイベントを選択します。 >
▼ [0]	Click
▼ Actions	< 追加するアクションを選択します。 >
▼ [0]	SetVariable
Variable	SampleB
Value	SampleC

● 値の加算/減算

1. ツールボックスから[ボタン]-[ボタン]を選択し、ページ上にドラッグ&ドロップしてください。



2. イベントとアクションから[Events]に「Click」、[Actions]に「IncreasetVariable」をそれぞれ選択し、[Variable]に入力対象の変数、[Value]に加算する値を設定します。

▼ Events	< 追加するイベントを選択します。 >
▼ [0]	Click
▼ Actions	< 追加するアクションを選択します。 >
▼ [0]	IncreaseVariable
Variable	SampleB
Value	1

3. 減算処理の場合は[Actions]に「DecreasetVariable」を選択し、[Value]に減算する値を設定します。

▼ Events	< 追加するイベントを選択します。 >
▼ [0]	Click
▼ Actions	< 追加するアクションを選択します。 >
▼ [0]	DecreaseVariable
Variable	SampleB
Value	1

4. [Value]に固定値ではなく変数を入れることで、加算/減算する値を間接指定とすることも可能です。

Button6	
▼ Events	< 追加するイベントを選択します。 >
▼ [0]	Click
▼ Actions	< 追加するアクションを選択します。 >
▼ [0]	IncreaseVariable
Variable	SampleB
Value	SampleC

2-3-4 マルチファンクションの作成方法

NS シリーズで利用できるマルチファンクションのような、1 つの部品で複数の機能を持つ部品を作成するには、NA シリーズでは専用のオブジェクトは用意されていません。オブジェクトに対して、イベントとアクションを複数割り当てることで実現できます。

1. ツールボックスから[ボタン]-[ボタン]を選択し、ページ上にドラッグ&ドロップしてください。



2. イベントとアクションから[Events]に「Click」、[Actions]に「SetVariable」をそれぞれ選択し、[Variable]に入力対象の変数、[Value]に入力値を設定します。

▼ Events	< 追加するイベントを選択します。 >
▼ [0]	Click
▼ Actions	< 追加するアクションを選択します。 >
▼ [0]	SetVariable
Variable	SampleB
Value	10

3. 続けて[Actions]に「SetVariable」を選択し、[Variable]に手順 2 とは別の入力対象の変数、[Value]に入力値を設定します。

イベントとアクション	
Button0	
▼ Events	< 追加するイベントを選択します。 >
▼ [0]	Click
▼ Actions	< 追加するアクションを選択します。 >
▼ [0]	SetVariable
Variable	SampleB
Value	10
▼ [1]	SetVariable
Variable	SampleC
Value	20

2-3-5 日付時刻表示のフォーマット

NS シリーズが日付部品と時刻部品がそれぞれ独立しているのに対し、NA シリーズでは日付時刻表示部品として統一されています。日付のみ、もしくは時刻のみ表示させたい場合はフォーマットを設定することで調整します。

15 種類フォーマットが用意されており、そのうちの 1 種類はユーザーが任意でフォーマットを設定します。

また、NS シリーズでは日付部品または時刻部品を入力許可設定にすることで、直接 PT 内部時刻を設定できましたが、NA シリーズでは日付時刻表示部品からの時刻設定はできません。

フォーマット名	表示例(2015 年 6 月 5 日 13 時 08 分 54 秒)
ShortDatePattern	06/05/2015
LongDatePattern	Friday,05 June 2015
FullDateShortTimePattern	Friday,05 June 2015 13:08
FullDateTimePattern	Friday,05 June 2015 13:08:54
ShortDateTimePattern	06/05/2015 13:08
ShortDateLongTimePattern	06/05/2015 13:08:54
MonthDayPattern	June 05
Rfc1123Pattern	Fri,05 June 2015 13:08:54 GMT
SortedDateTimePattern	2015-06-05T13:08:54
ShortTimePattern	13:08
LongTimePattern	13:08:54
UniversalSortedDateTimePattern	2015-06-05 13:08:54Z
UniversalSortedDateTimePattern	Friday,05 June 2015 04:08:54
YearMonthPattern	2015 June
UseCustomDateTimePattern	ユーザーの任意設定※1

※1: 下図[CustomFormat]にユーザーが任意に設定します。

・ UseCustomDateTimePattern の設定例

CustomFormat 設定内容	表示例(2015 年 6 月 5 日 13 時 08 分 54 秒)
yyyy/MM/dd HH:mm:ss	2015/06/05 13:08:54
yy-MM-dd dddd HH:mm:ss dddd	15-06-05 Friday 13:08:54
yy/M/d ddd H:m:s	15/6/5 Fri 13:8:54
MMMM dd tt hh:mm:ss	June 05 PM 01:08:54
MMM dd hh:mm:ss t	Jun 05 01:08:54 P

注意:

大文字と小文字は区別されるため、日付の箇所を YYYY/MM/DD と設定すると、年と日が正常に表示されません。

2-3-6 ボタンオブジェクトの状態変化

NA シリーズのボタンオブジェクト類は、ボタン押下時のみ状態の変化が行われ、NS シリーズのようなアドレスによる状態変化機能を持ちません。

よって、NS シリーズと同等の動作にする場合は、ランプオブジェクトとの組み合わせが必要になります。

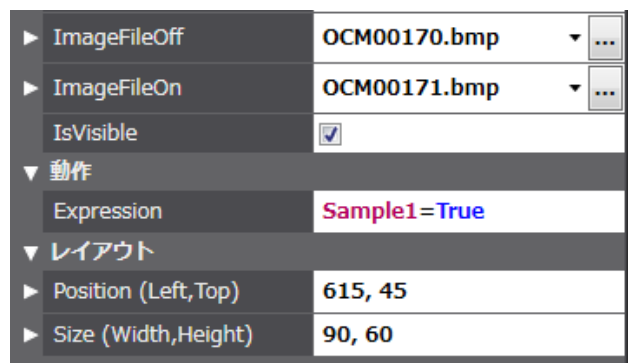
● 書込アドレスの ON/OFF による状態変化

NS シリーズでのボタンの種類(Type1)に該当します。押下時は状態変化を行わず、書込アドレスの ON/OFF でのみ状態が変化します。

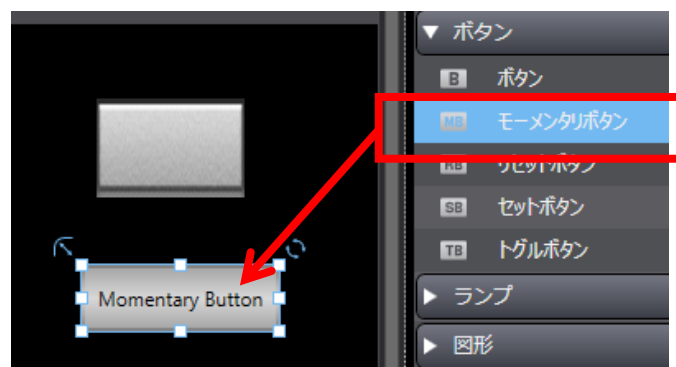
1. ツールボックスのタブを選択し、[ランプ]-[ビットランプ]を選択し、ドラッグ&ドロップします。



2. ビットランプを選択して[プロパティ]タブを開き、外観・レイアウトを任意に設定します。
また、状態変化の条件も併せて設定します。(NS シリーズでいう書込アドレスを設定)



3. ツールボックスのタブを選択し、[ランプ]-[モーメンタリボタン]を選択し、ドラッグ&ドロップします。
(トグル操作が必要なトグルボタン、セット操作が必要なセットボタンと、部品の機能に応じて作成するランプを選択してください)



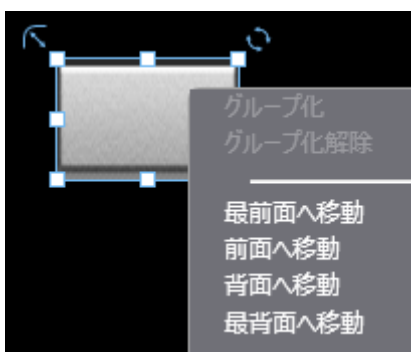
4. モーメンタリボタンの部品色を設定する箇所を、すべて「Transparent」に設定してください。

TextColorButtonUp	☐ Transparent	▼
TextColorButtonDown	☐ Transparent	▼
BackgroundColorButtonU	☐ Transparent	▼
BackgroundColorButtonD	☐ Transparent	▼
BorderColorButtonUp	☐ Transparent	▼
BorderColorButtonDown	☐ Transparent	▼

5. モーメンタリボタンの動作条件を、手順 2 で作成したビットランプに設定した変数と同じ変数を選択します。また、[Position]と[Size]を手順 1~3 で作成したビットランプと合わせます。

動作	
Variable	Sample1
IsEnabled	☒
DoubleTouchTime	0
OnDelayTime	0
OffDelayTime	0
レイアウト	
Position (Left,Top)	615, 45
Size (Width,Height)	90, 60

6. 部品が重なっている場合、背面の部品は動作しない為、ボタンが背面にある場合は部品を右クリックして前面のランプを「最背面へ移動」させてボタンを前面へ配置してください。



● 表示アドレスの ON/OFF による状態変化

NS シリーズでのボタンの種類(Type2-1)に該当します。押下時は状態変化を行わず、表示アドレスの ON/OFF でのみ状態が変化します。

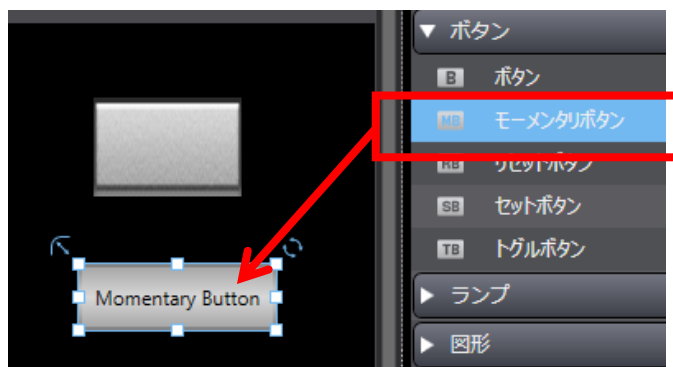
1. ツールボックスのタブを選択し、[ランプ]-[ビットランプ]を選択し、ドラッグ&ドロップします。



2. ビットランプを選択して[プロパティ]タブを開き、外観・レイアウトを任意に設定します。
また、状態変化の条件も併せて設定します。(NS シリーズでいう表示アドレスを設定)
条件(Expression)に変数名のみ記載した場合、=True と同義になります。

ImageFileOff	OCM00170.bmp
ImageFileOn	OCM00171.bmp
IsVisible	☒
動作	
Expression	Sample2=True
レイアウト	
Position (Left,Top)	615, 150
Size (Width,Height)	90, 60

3. ツールボックスのタブを選択し、[ランプ]-[モーメンタリボタン]を選択し、ドラッグ&ドロップします。
(トグル操作が必要ならトグルボタン、セット操作が必要ならセットボタンと、部品の機能に応じて作成するランプを選択してください)



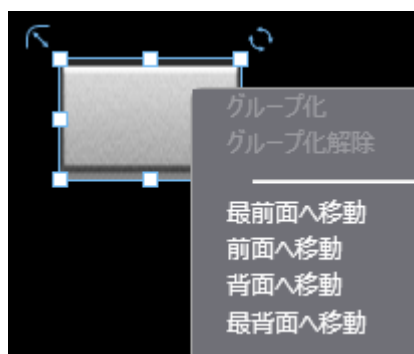
4. モーメンタリボタンの部品色を設定する箇所を、すべて「Transparent」に設定してください。

TextColorButtonUp	☐ Transparent
TextColorButtonDown	☐ Transparent
BackgroundColorButtonU	☐ Transparent
BackgroundColorButtonD	☐ Transparent
BorderColorButtonUp	☐ Transparent
BorderColorButtonDown	☐ Transparent

5. モーメンタリボタンの動作条件を、NSシリーズでいう書込アドレスを設定します。また、[Position]と[Size]を手順1~3で作成したビットランプと合わせます。

動作	
Variable	Sample1
IsEnabled	☒
DoubleTouchTime	0
OnDelayTime	0
OffDelayTime	0
レイアウト	
Position (Left,Top)	615, 45
Size (Width,Height)	90, 60

6. 部品が重なっている場合、背面の部品は動作しない為、ボタンが背面にある場合は部品を右クリックして前面のランプを「最背面へ移動」させてボタンを前面へ配置してください。



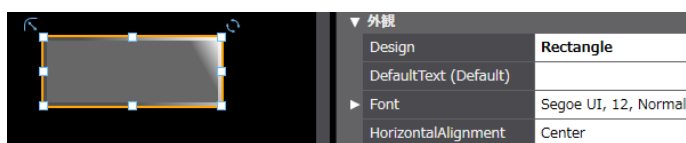
2-3-7 メッセージ表示器の作成方法

NS シリーズで利用できるメッセージ表示器のような、変数の値によって状態を変化させる機能は、NA シリーズではデータランプオブジェクトを使用します。
ただし、NS シリーズでは状態ごとにフォントの種類やサイズを設定することができましたが、NA シリーズではベース設定となる値から変えることができません。背景色や文字色の変更は従来通り行えます。

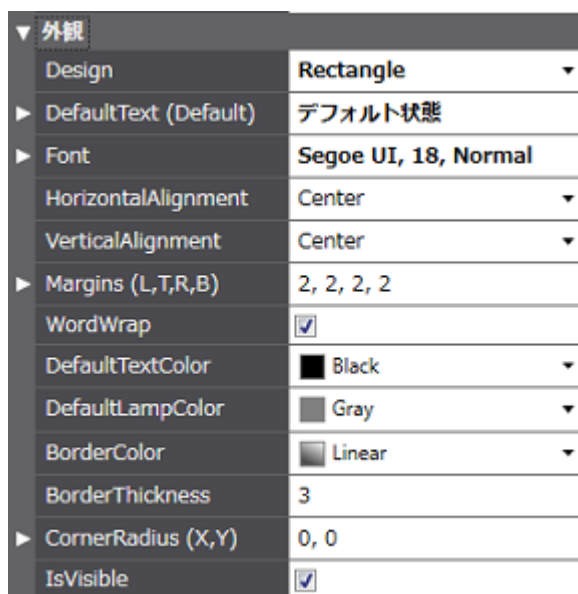
1. ツールボックスから[ランプ]-[データランプ]を選択し、ページ上にドラッグ&ドロップしてください。



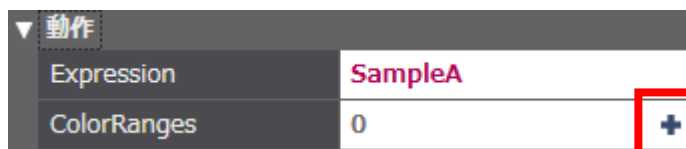
2. オブジェクトを選択して[プロパティ]タブを開き、[外観]-[Design]を「Rectangle」に設定し、部品サイズを任意に調整します。



3. [外観]からデフォルト状態のテキスト内容、フォント種類等を任意に設定します。



4. [動作]-[Expression]に状態切替を司る変数を設定し、[ColorRanges]内の+をクリックします。



5. [ColorRanges]に設定欄が追加されるので、[StartValue]に条件を設定します。また、[StartValue]の条件を満たした時のテキスト内容を[Text(Default)]、背景色を[LampColor]、テキスト色を[TextColor]にそれぞれ設定します。右図の設定では変数「SampleA」の値が 1 以上になるとテキスト内容が「状態 1」となり、背景色が白、文字色が赤に変化します。

▼ 動作	
Expression	SampleA
ColorRanges	1 +
▼ [0]	>= 1 🗑
StartValue	1
▶ Text (Default)	状態1
LampColor	☐ White ▼
TextColor	☒ Red ▼

6. さらに条件を追加する場合は、都度 [ColorRanges]内の+をクリックして設定欄を追加し、手順 5 の操作を繰り返します。

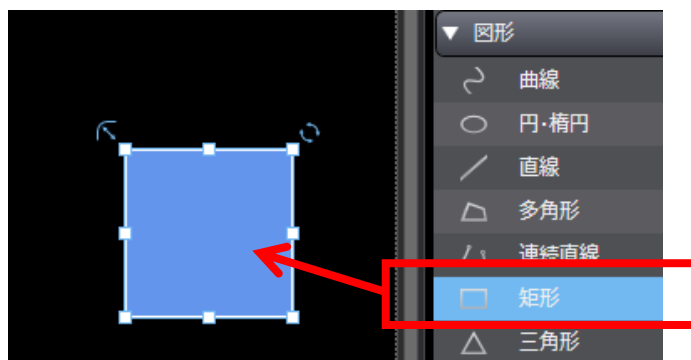
▼ 動作	
Expression	SampleA
ColorRanges	3 +
▶ [0]	>= 1 🗑
▶ [1]	>= 2 🗑
▶ [2]	>= 3 🗑

2-3-8 グラデーションがかかっていないランプ部品の作成方法

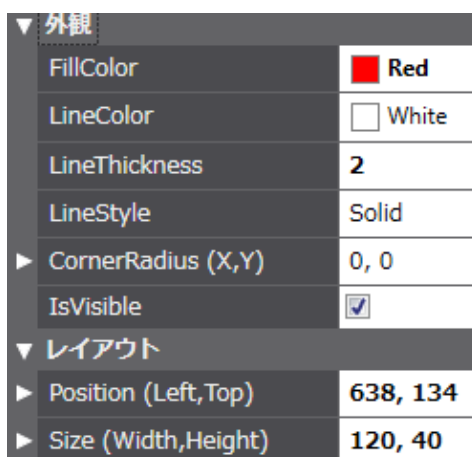
NA シリーズのランプ部品はデフォルトでグラデーションがかかっており、ボタンと同じ配色設定にしても完全に同じ外観になりません(下図参照)。ランプ部品の形状設定を「Image」にして外部画像を取り込むことで回避できますが、アニメーション機能を使用する事でランプの機能を一部代替することができます。



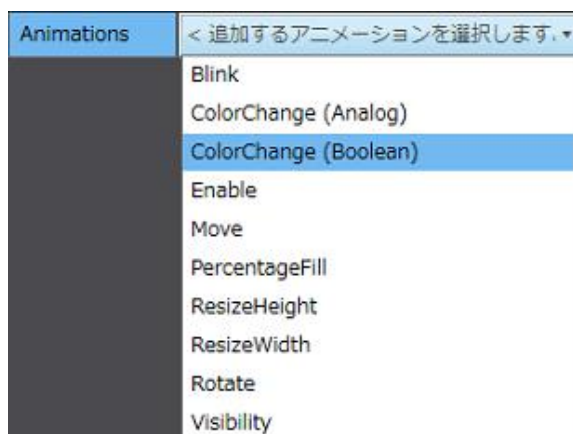
1. ツールボックスから[図形]-[矩形]を選択し、ページ上にドラッグ&ドロップしてください。



2. オブジェクトを選択して[プロパティ]タブを開き、[外観]から部品色と枠色を、[レイアウト]から部品サイズをそれぞれ任意に調整します。



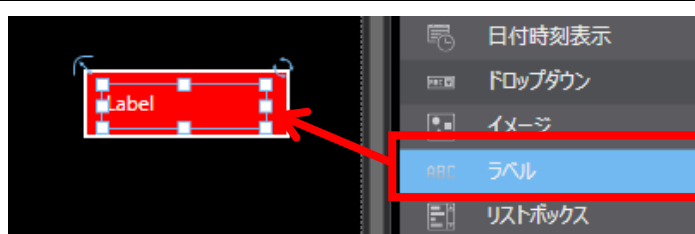
3. [アニメーション]タブを開き、[Animations]から「ColorChange(Boolean)」を選択します。



4. [Expression]に背景色を変化させる条件を設定し、[OffColor]に条件がFalse時の背景色を、[OnColor]に条件がTrue時の背景色をそれぞれ設定します。
図形部品で変更できるのは背景色のみであり、ランプのように枠の色は変更できない点に注意してください。

Animations	< 追加するアニメーションを選択します、>
▼ [0]	ColorChange (Boolean)
Expression	Sample1=True
OffColor	Red
OnColor	Blue

5. 銘板を追加する場合は、ツールボックスから[標準コントロール]-[ラベル]を選択し、手順4までに作成したオブジェクトの上にドラッグ&ドロップし、任意の文字列に設定します。
ラベル部品も矩形部品と同様に、アニメーションタブから「ColorChange」による状態変化ができるので、枠の色の変化以外は、ビットランプとほぼ同様の機能が再現できます。



6. データランプのような、変数の値に応じて状態を切り替える機能を使用する場合は、手順3で「ColorChange(Analog)」を選択します。

Animations	< 追加するアニメーションを選択します、>
	Blink
	ColorChange (Analog)
	ColorChange (Boolean)

7. [動作]-[Expression]に状態切替を司る変数を設定し、[ColorRanges]内の+をクリックします。

Animations	< 追加するアニメーションを選択 >
▼ [0]	ColorChange (Analog)
Expression	SampleA
DefaultColor	Red
ColorRanges	0 +

8. [ColorRanges]に設定欄が追加されるので、[StartValue]に条件を設定します。また、[StartValue]の条件を満たした時の背景色を[Color]に設定します。右図の設定では変数「SampleA」の値が5以上になるとテキスト内容が「状態 1」となり、背景色が白に変化します。さらに条件を追加する場合は、再度[ColorRanges]内の+をクリックして設定欄の追加を行ってください。

Animations	< 追加するアニメーションを選択 >
▼ [0]	ColorChange (Analog)
Expression	SampleA
DefaultColor	Red
▼ ColorRanges	1 +
▼ [0]	>= 5
StartValue	5
Color	White

2-3-9 整数型数値の小数点表示

NS シリーズでは[単位とスケール]から設定していた機能です。

現在、整数型の変数に対して本機能を用いて小数点表示を行うことができるのは、データ表示部品のみです。データ入力部品は演算式を設定できないので、小数を扱う場合は変数のデータ型を浮動小数点型に設定する必要があります。

下記説明は変数「SampleA」に対して倍率設定「0.1 倍」を設定する例です。

1. データ表示部品のプロパティタブを開き、右図のように設定します。倍率を[Expression]で決定し、表示形式を[DisplayFormat]で決定します。

動作	
DataType	Numeric ▼
Expression	SampleA /10
ValueFormat	Decimal ▼
DisplayFormat	###.## ▼

2. 手順 1 の設定で変数 SampleA に値「1234」を入力すると右図のようになります。

入力値
1234
表示結果
123.4

9 月に予定されているバージョンアップでデータ入力・表示部品にスケール機能が実装されます。

2-3-10 アラームオブジェクト

NS シリーズと NA シリーズのアラームオブジェクトはデフォルト設定での動作が大きく異なります。NS シリーズのアラーム設定に近づける設定方法を記載します。

● アラーム確認時ポップアップを表示しない方法

NA のアラームオブジェクトのデフォルト設定では、発生したアラームをタッチするとアラーム詳細情報が記載されたポップアップが表示されます。そのポップアップを表示させない方法を以下に記載します。

1. マルチビューエクスプローラの [HMI]-[ユーザーアラーム] を選択してください。

グループ表示名							
名前	ID	アラームコード	条件式	優先順位	メッセージ	ポップアップ	確認
Alarm1	Group0_Alarm1	NJ_1_CIO_3000_0	ユーザアラームレ		アラームメッセージ1	☒	☒
Alarm2	Group0_Alarm2	NJ_1_CIO_3000_0	ユーザアラームレ		アラームメッセージ2	☒	☒
Alarm3	Group0_Alarm3	NJ_1_CIO_3000_0	ユーザアラームレ		アラームメッセージ3	☒	☒
Alarm4	Group0_Alarm4	NJ_1_CIO_3000_0	ユーザアラームレ		アラームメッセージ4	☒	☒
Alarm5	Group0_Alarm5	NJ_1_CIO_3000_0	ユーザアラームレ		アラームメッセージ5	☒	☒

2. ポップアップを表示させたくないイベントの「ポップアップ」のチェックを外します。

ポップアップ	確認
☐	☐
☐	☐
☐	☐
☐	☐
☐	☐

● アラームの発生/解除情報を表示する方法

NA のアラームオブジェクトのデフォルト設定では、発生したアラームの表示/解除の情報が表示されません。下図では「アラームメッセージ 1～3」の発生と解除がそれぞれ記録されていますが、表示/解除の情報が無い為に分かりにくい表示となっています。

なお、NS では発生と解除を同一行に表示することができましたが、NA では強制的に別の行に分かれて表示されます。発生のみ表示、もしくは解除のみ表示させることもできません。

Date and Time	Message	Priority	Group
6/19/2015 3:19:01 PM	アラームメッセージ2	Level 1	
6/19/2015 3:18:57 PM	アラームメッセージ3	Level 1	
6/19/2015 3:18:56 PM	アラームメッセージ3	Level 1	
6/19/2015 3:18:54 PM	アラームメッセージ2	Level 1	
6/19/2015 3:18:50 PM	アラームメッセージ1	Level 1	
6/19/2015 3:18:49 PM	アラームメッセージ1	Level 1	

1. ユーザーアラームビューワを選択して[プロパティ]タブを開きます。(右図はデフォルトの設定です)

ColumnTitles	Date and Time Message Priority Group
ColumnOrder	Time Message Priority Group
ColumnWidths	130.0 300.0 80.0 200.0

2. [ColumnTitles]と[ColumnOrder]の先頭に「Status」と追記し、[ColumnWidths]では項目の幅を任意に設定します。本項目では大文字と小文字を区別するため、「Status」は必ず先頭のみ大文字とし、以降は小文字で記載してください。

ColumnTitles	Status	Date and Time Message Priority Group
ColumnOrder	Status	Time Message Priority Group
ColumnWidths	160.0	210.0 210.0 90.0 70.0

上記の設定を行ったアラームオブジェクトは以下のように表示されます。

「Status」の欄に発生/解除の情報が表示されています。

「Alarm Raised」が発生、「Alarm Cleared」が解除を表しています。

Status	Date and Time	Message	Priority	Group
Alarm Cleared	6/19/2015 3:19:01 PM	アラームメッセージ2	Level 1	
Alarm Cleared	6/19/2015 3:18:57 PM	アラームメッセージ3	Level 1	
Alarm Raised	6/19/2015 3:18:56 PM	アラームメッセージ3	Level 1	
Alarm Raised	6/19/2015 3:18:54 PM	アラームメッセージ2	Level 1	
Alarm Cleared	6/19/2015 3:18:50 PM	アラームメッセージ1	Level 1	
Alarm Raised	6/19/2015 3:18:49 PM	アラームメッセージ1	Level 1	

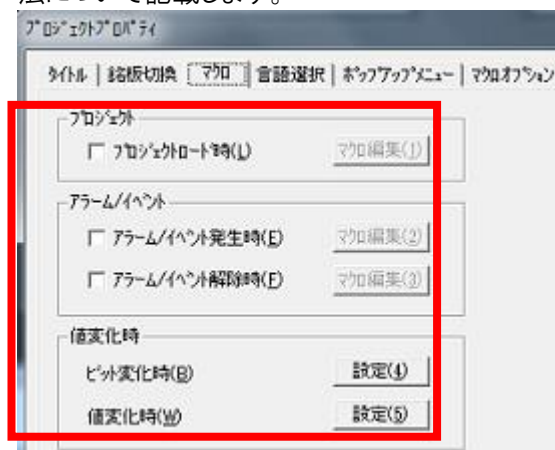
2-4 その他

2-4-1 システム設定におけるサブルーチンの設定方法

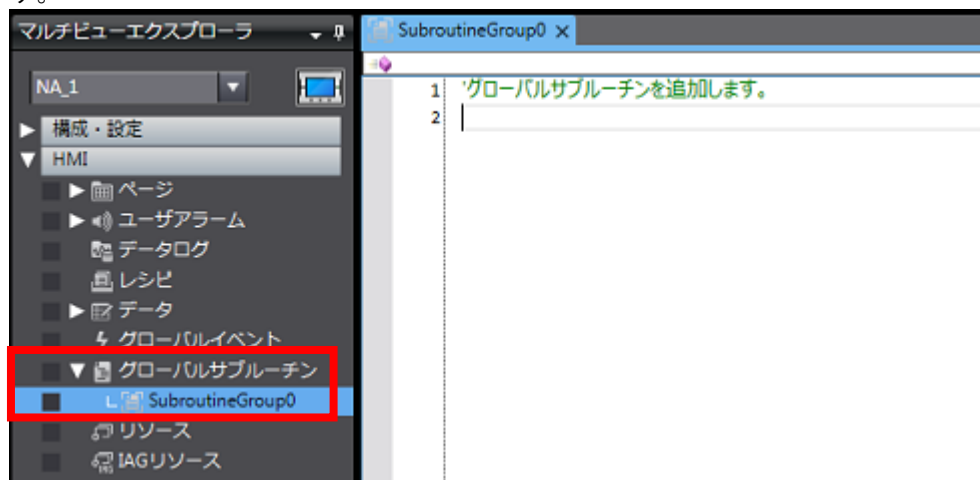
サブルーチンは NS シリーズではマクロと呼ばれている機能です。

ユーザが任意でプログラムを記述して、基本機能でサポートしていない機能を追加することができます。

本項では NS シリーズでの[システム]-[プロジェクトプロパティ]-[マクロ]内で設定できるマクロの動作方法について記載します。



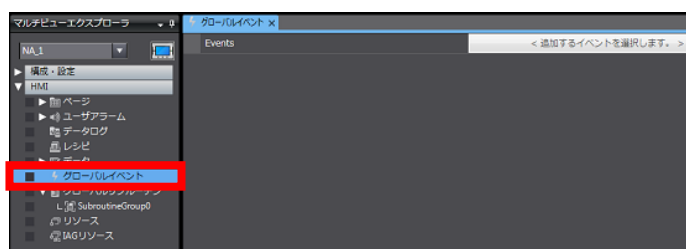
サブルーチンの内容は、マルチビューエクスプローラの[HMI]-[グローバルサブルーチン]に記載します。



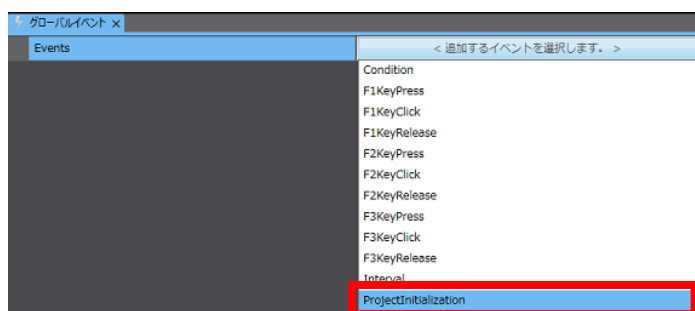
● プロジェクトロード時に起動するサブルーチンの作成方法

NA シリーズの電源が投入され、システムが起動した時に実行されるサブルーチンです。

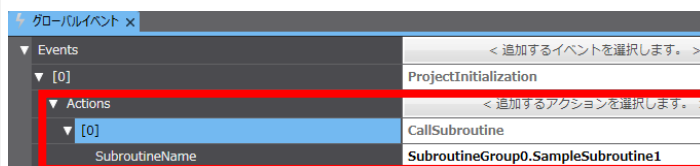
1. マルチビューエクスプローラの [HMI]-[グローバルイベント] を選択してください。



2. [Events] から「ProjectInitialization」を選択してください。



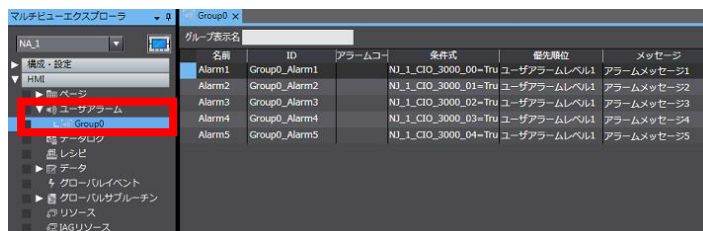
3. [Action] から「CallSubroutine」を選択し、[Subroutine] に呼び出すサブルーチン名を設定してください。



● アラームイベント発生/解除時に起動するサブルーチン

NS シリーズでは発生と解除時にマクロ起動できましたが、NA ではアラーム確認のタイミングでも起動可能です。

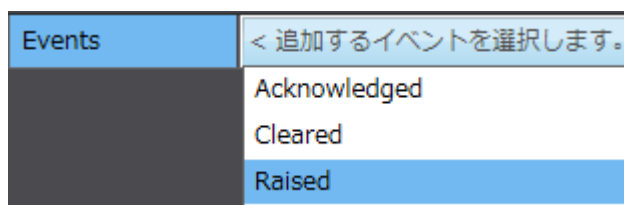
1. マルチビューエクスプローラの [HMI]-[ユーザーアラーム] を選択してください。



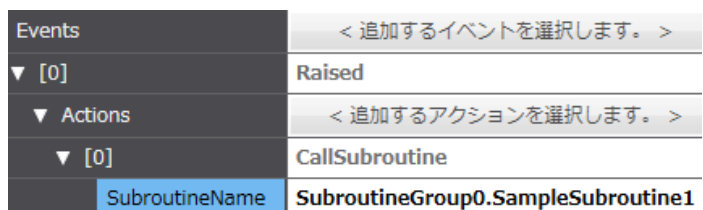
2. サブルーチンを割り当てたいアラームイベントを選択した状態で、[イベントとアクション] タブを開きます。



3. アラームイベント発生時に実行する場合は「Raised」、解除時は「Cleared」、確認時は「Acknowledged」をそれぞれ選択します。

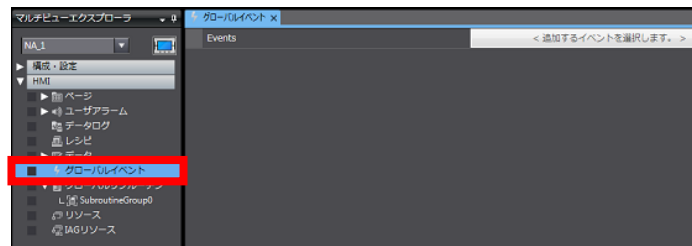


4. [Actions] から「CallSubroutine」を選択し、[Subroutine] に呼び出すサブルーチン名を設定してください。

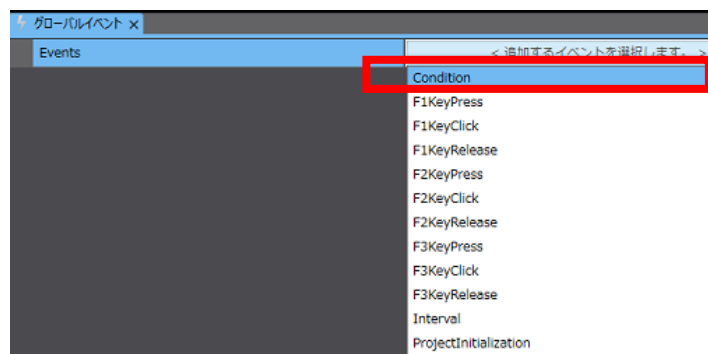


● ビット/値変化時に起動するサブルーチン

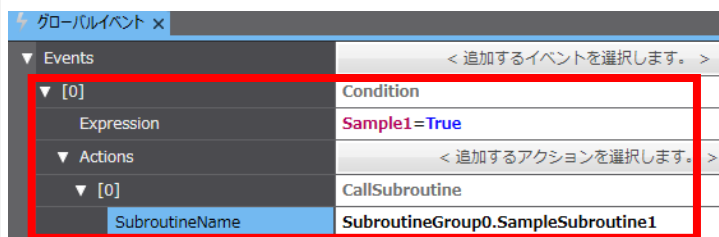
1. マルチビューエクスプローラの [HMI]-[グローバルイベント] を選択してください。



2. [Events] から「Condition」を選択してください。



3. [Expression] にサブルーチン機能条件を設定した後、[Action] から「CallSubroutine」を選択し、[Subroutine] に呼び出すサブルーチン名を設定してください。



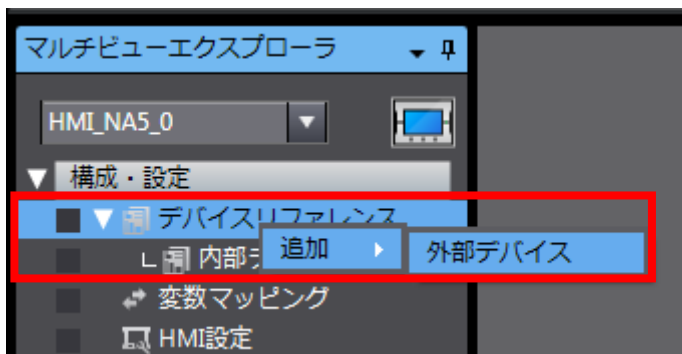
2-4-2 CJ シリーズ CPU との接続方法

NA シリーズは NJ シリーズの他、CJ シリーズとも接続が可能です。

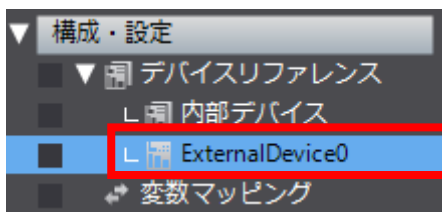
CJ シリーズの変数を NA で使用するには、CJ シリーズ PLC を接続デバイスとして設定し、CJ の変数を NA へ登録する必要があります。

CJ シリーズを接続デバイスとして登録する方法は以下の通りです。

1. マルチビューエクスプローラから[デバイスリファレンス]を右クリックして[追加]-[外部デバイス]を選択します。



2. [デバイスリファレンス]のツリーの下に追加された[ExternalDevice0]をダブルクリックします。



3. [デバイス構成]から[デバイス]を「CJ」に選択し、[接続方法]に「CIP Ethernet」もしくは「FINS Ethernet」を選択し、[通信の構成]-[IP アドレス]に接続先 PLC の IP アドレスを設定します。



外部デバイスとして登録した CJ の変数を NA へ登録する方法は以下の 3 種類あります。

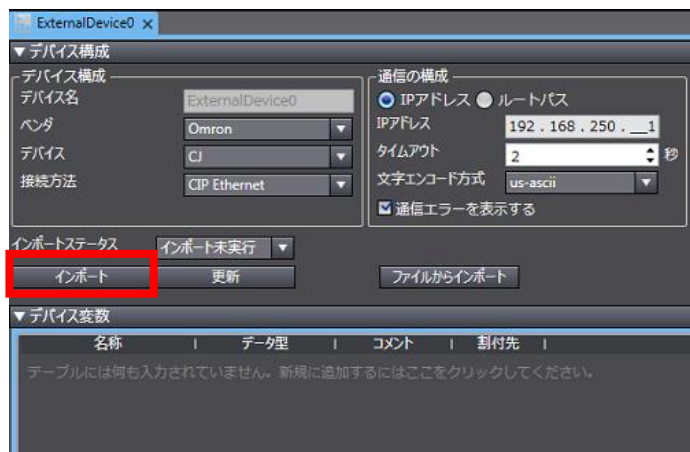
- ・デバイスから直接、変数のインポート・更新を行う
- ・変数テーブルへ直接入力する
- ・Excel 等からコピーペーストする

上記のいずれの方法でも、CJ のシステム変数をインポートすることはできません。

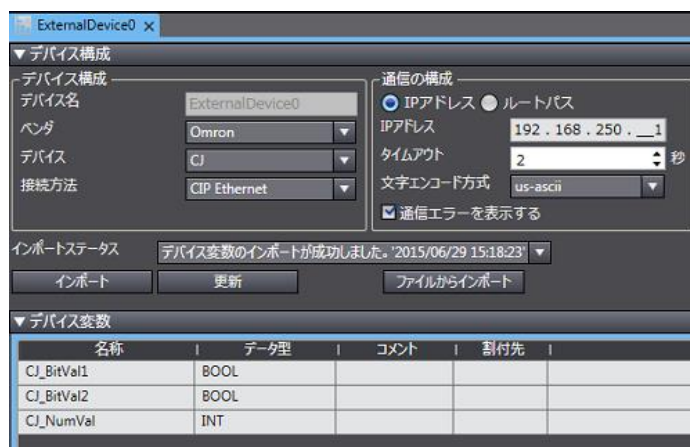
● デバイスから直接、変数のインポート・更新を行う

[接続方法]に[CIP Ethernet]を選択した時のみ実行可能です。あらかじめ CPU ユニットに、変数テーブルを転送しておく必要があります。

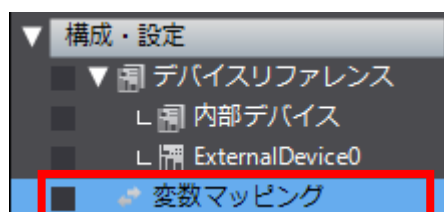
1. CJ CPU と PC とを Ether ケーブルで接続した状態で「インポート」を選択します。(この時、Sysmac Studio 側の通信設定は「USB」でもかまいません)



2. [デバイス変数]の欄に CJ に登録されているデバイス変数がインポートされます。



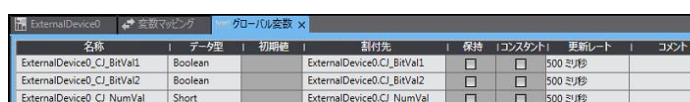
3. マルチビューエクスプローラから[変数マッピング]をダブルクリックします。



4. NJ CPU の変数を登録する場合と同様に、右クリックメニューの「デバイス変数生成」から変数の作成が行えます。



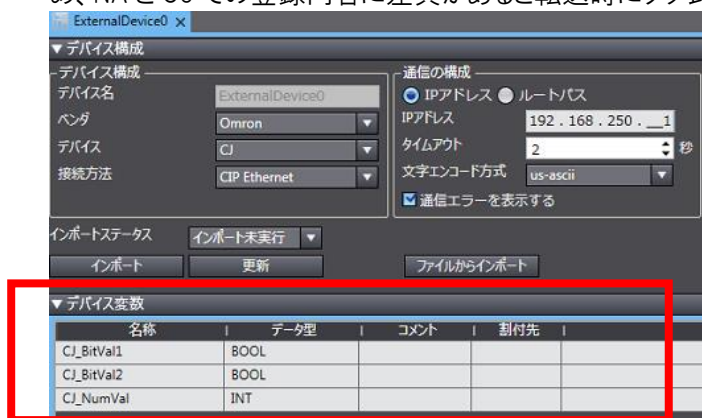
5. 自動的に[グローバル変数]に手順 4 で生成した変数が登録されます。



● 変数テーブルへ直接入力する

デバイス変数欄は直接編集することが可能です。

ただし、接続方法に「CIP Ethernet」を選択した場合は変数名やデータ型で CJ 本体との照合を行うため、NA と CJ での登録内容に差異があると転送時にタグ比較エラーが発生します。



● Excel 等からコピー・ペーストする

デバイス変数欄は直接入力の他、Excel やテキストファイルからコピー・ペーストすることも可能です。ペーストする際は、あらかじめ定められたフォーマットで記述されている必要があります。

1. CX-Programmer の変数テーブルを開き、コピーする範囲を選択して「右クリック+コピー」もしくは「Ctrl+C」で変数をコピーします。

名称	データ型	アドレス / 値	ネットワーク...	用途	コメント
P_UF	BOOL	CF010		ワーク	アンダーフロー(UF)フラグ
P_WR	WORD	A451		ワーク	WR112指定
CJ_BitVal1	BOOL	W0.00	公開のみ	ワーク	
CJ_BitVal2	BOOL	W0.01	公開のみ	ワーク	
CJ_NumVal	INT	D0	公開のみ	ワーク	

2. コピーしたデータを Excel へ張り付けます。

B3		fx C_J_BitVal1						
	A	B	C	D	E	F	G	H
1								
2								
3		C_J_BitVal1	BOOL	W0.00		0		
4		C_J_BitVal2	BOOL	W0.01		0		
5		C_J_NumVal	INT	D0		0		
6								
7								

3. ペーストしたデータの並びを以下のように変更します。
「変数名→データ型→コメント→割付先(アドレス)」

A1		fx					
	A	B	C	D	E	F	
1							
2							
3		CJ_BitVal1	BOOL		W0.00		
4		CJ_BitVal2	BOOL		W0.01		
5		CJ_NumVal	INT		D0		
6							

4. 並び替えたデータをデバイス変数欄へペーストします。この時、コメント欄が空白であっても名称から割付先までの 4 列は同時にコピー・ペーストを行ってください。

インポートステータス		インポート未実行		ファイルからインポート	
インポート		更新			
▼ デバイス変数					
名称	データ型	コメント	割付先		
CJ_BitVal1	BOOL		W0.00		
CJ_BitVal2	BOOL		W0.01		
CJ_NumVal	INT		D0		

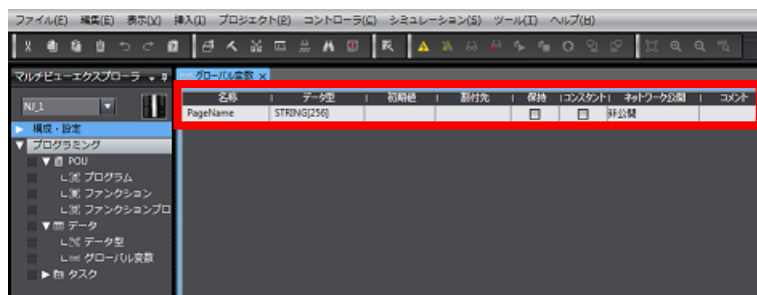
2-4-3 PLC 変数から画面を切り替える方法

「_HMI_CurrentPage」は表示中の画面名を示す NA のシステム変数です。NA の変数マッピング機能はシステム変数を割り当てることもできます。

また、NS シリーズ等のように、アドレスの値が画面番号と直結している設定の場合は Visual Basic を記載する必要があります。

● NJ 変数を NA システム変数へ割り当てる方法

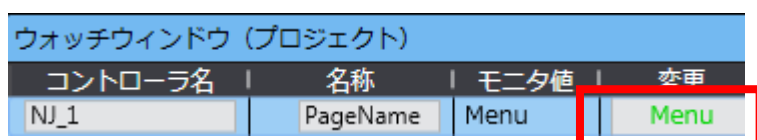
1. NJ 側で画面名を格納する String 型変数を宣言します。



2. NA 側の変数マッピング機能から、手順 1 で作成した、画面名の変数に対して NA システム変数である「_HMI_CurrentPage」を割り当てます。(この時、右クリックからの自動割当てでなく、手動でシステム変数名を入力してください)



3. 手順 1 で作成した NJ 側変数に、移動先ページ名を書込むことで該当画面へ移動します。



● CJ 変数を NA システム変数へ割り当てる方法

1. CJ 側で画面名を格納する String 型変数を宣言します。

名称	データ型	アドレス / 値	ネットワーク変数
P_UF	BOOL	CF010	
P_WR	WORD	A451	
CJ_BitVal1	BOOL	W0.00	公開のみ
CJ_BitVal2	BOOL	W0.01	公開のみ
CJ_NumVal	INT	D0	公開のみ
CJ_PageNumber	INT	D10	公開のみ
CJ_PageName	STRING(8)	D100	公開のみ

2. 手順 1 で作成した CJ 変数を NA に割り付けます。割付の方法は 2-4-2 CJ シリーズ CPU との接続方法を参照してください。

▼ デバイス変数				
名称	データ型	コメント	割付先	
CJ_BitVal1	BOOL			
CJ_BitVal2	BOOL			
CJ_NumVal	INT			
CJ_PageName	STRING(8)			
CJ_PageNumber	INT			

3. NA 側の変数マッピング機能から、手順 1 で作成した、画面名の変数に対して NA システム変数である「_HMI_CurrentPage」を割り当てます。(この時、右クリックからの自動割当てでなく、手動でシステム変数名を入力してください)

▼ ExternalDevice0		
CJ_BitVal1	BOOL	CJ_BitVal1
CJ_BitVal2	BOOL	CJ_BitVal2
CJ_NumVal	INT	CJ_NumVal
CJ_PageName	STRING(8)	_HMI_CurrentPage
CJ_PageNumber	INT	CJ_PageNumber

4. 手順 1 で作成した CJ 側変数に、移動先ページ名を書込むことで該当画面へ移動します。

新しい値を設定

アドレス:	D100	値変更(S)
現在値:	'Page0'	閉じる
値(V):	'Page1'	アドレス/型変更(E)
2進表示(B) >>		

● Visual Basic を使用する方法(Case 文)

Visual Basicコードの中で画面毎に番号を割り振り、その番号に対応した画面に移動する動作を Case 文を使用して作成します。

1. NJ 側で画面番号を格納する数値型変数を宣言します。

グローバル変数 ×						
名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開
PageName	STRING(256)			☐	☐	非公開
PageNumber	DINT			☐	☐	非公開

2. 手順 1 で宣言した変数を NA 側でデバイス変数割付を行い、画面名を格納する String 型変数と、NA 側で画面番号を格納する変数をそれぞれ宣言します。
「Page_Name」:画面名を表す変数
「Page_No」:NA 側の画面番号を表す変数

SubroutineGroup0 グローバル変数 ×						
名称	データ型	初期値	割付先	保持	コンスタント	更新レート
Always_Off	Boolean	False		☐	☒	無効
NJ_1_PageNumber	Integer		NJ_1.PageNumber	☐	☐	500 ミリ秒
Page_Name	String			☐	☐	無効
Page_No	Integer			☐	☐	無効
Sample1	Boolean			☐	☐	無効

3. グローバルサブルーチンで右図のコードを記述します。
「PageNameSelect」:
NJ の変数の値を元に NA の画面を切り替えます。
「PageNumberSelect」:
現在表示している画面に対応する番号を PLC へ通知します。
どちらの場合も移動する画面を追加する場合は、Case の条件を増やしてください。

```

3 Sub PageNameSelect
4   Select Case NJ_1_PageNumber
5     Case 0
6       Page_Name="Page0"
7     Case 1
8       Page_Name="Page1"
9     Case 2
10      Page_Name="Page2"
11   End Select
12   Page_No=NJ_1_PageNumber
13   ShowPage(Page_Name)
14 End Sub

15 Sub PageNumberSelect
16   Select Case _HMI_CurrentPage
17     Case "Page0"
18       NJ_1_PageNumber=0
19     Case "Page1"
20       NJ_1_PageNumber=1
21     Case "Page2"
22       NJ_1_PageNumber=2
23   End Select
24 End Sub

```

4. 手順 3 で作成したコードの実行条件をグローバルイベントから設定します。
①NJ 側の画面番号と NA 側の画面番号が等しくない状態を表します。
②実行するサブルーチン名を指定します。
③ページ切替が行われた時に動作します。
④実行するサブルーチン名を指定します。

Events		< 追加するイベントを選択します。 >
▼ [0]	Condition	
	Expression	NJ_1_PageNumber <> Page_No
▼ Actions	< 追加するアクションを選択します。 >	
	▼ [0]	
	SubroutineName	SubroutineGroup0.PageNameSelect
▼ [1]	Condition	
	Expression	_HMI_IsPageSwitching=False
▼ Actions	< 追加するアクションを選択します。 >	
	▼ [0]	
	SubroutineName	SubroutineGroup0.PageNumberSelect

5. NJから画面を切り替える場合は NJ 変数「PageNumber」を変更します。NAから画面を切り替える場合は NA システム変数「_HMI_CurrentPage」を変更します。

ウォッチウィンドウ (プロジェクト)			
コントローラ名	名称	モニタ値	変更
NJ_1	PageNumber	1	1

- Visual Basic を使用する方法(String 型配列とシステム変数_HMI_CurrentPage)
システム変数_HMI_CurrentPage とユーザー定義 String 型配列変数を組み合わせた方法です
画面番号と配列の要素番号を直結させ、配列変数の内容に応じた画面番号へ移動させます。

1. NJ 側で画面番号を格納する数値型変数と、String 型配列変数を宣言します。String 型配列変数の要素数は画面数以上の値を設定してください。

名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開
PageName	STRING[256]			☐	☐	非公開
PageNumber	DINT			☐	☐	非公開
PageNameAry	ARRAY[0..19] OF STRING[255]			☐	☐	非公開

2. String 型配列変数の初期値に画面名を入力します。後に画面名が変更になった場合は、本欄も変更を行う必要がある点に注意してください。

初期値設定

PageNameAry[0..19]

[0]	'Page0'
[1]	'Page1'
[2]	'Page2'
[3]	'Page3'
[4]	'Page4'
[5]	'Page5'

3. 手順 1 で宣言した変数を NA 側でデバイス変数割付を行った後、画面番号を格納する変数を宣言します。

▼ ユーザ変数		
PageName	STRING[256]	HMI_CurrentPage
PageNameAry	ARRAY[0..19] OF STRING[255]	NJ_1_PageNameAry
PageNumber	DINT	NJ_1_PageNumber

4. グローバルサブルーチンで右図のコードを記述します。
String 型配列変数の要素数の箇所に画面番号を表す変数を設定することで、画面番号に応じた画面へ移動します。

```

26 Sub PageChange
27     _HMI_CurrentPage=NJ_1_PageNameAry(NJ_1_PageNumber)
28     Page_No=NJ_1_PageNumber
29 End Sub

```

5. コードの実行条件をグローバルイベントから設定します。右図では 100 ミリ秒ごとに手順 4 で作成したコードが実行されます。

▼ [2]	Interval
Interval	100 ミリ秒
▼ Actions	< 追加するアクションを選択 >
▼ [0]	CallSubroutine
SubroutineName	SubroutineGroup0.PageChange

6. NJ 変数「PageNumber」を変更することで、手順 2 で作成した配列変数の初期値として登録した画面へ移動します。
例: 2 を入力すると Page2 へ移動

初期値設定

PageNameAry[0..19]

[0]	'Page0'
[1]	'Page1'
[2]	'Page2'
[3]	'Page3'
[4]	'Page4'
[5]	'Page5'

● Visual Basic を使用する方法(“Page”変数でページ遷移を行う)

デフォルトで画面名は「Page0」「Page1」・・・と、「Page」+数字に自動で設定されます。

これを利用して、「Page」部分を固定し数字部分を変数にして、変数値の変更で画面切り替えを行います。

ただし、画面名はデフォルトから変更している場合はこの方法は使用できません。

1. NJ 側で画面番号を格納する数値型変数を宣言します。

グローバル変数 ×						
名称	データ型	初期値	割付先	保持	「コンスタント」	ネットワーク公開
PageName	STRING(256)			☐	☐	非公開
PageNumber	DINT			☐	☐	非公開

2. NA 側で以下の変数を宣言します。
 NJ_1_PageNumber: NJ 側で設定したページ番号変数を紐付けしています。
 Page_No: ページ番号を格納します。
 Page_Str: 初期値を「Page」に設定しています。後述する Visual Basic で使用されます。
 Page_No_Str: Page の数字部分を格納する String 型変数です。

名称	データ型	初期値	割付先
NJ_1_PageNumber	Integer		NJ_1.PageNumber
Page_No	Integer		
Page_Str	String	'Page'	
Page_No_Str	String		

3. グローバルサブルーチンで右図のコードを記述します。
 画面番号を文字列変換して、「Page_Str」の初期値である「Page」と連結することで「Page0」「Page1」といった画面名に該当する文字列を作成し、画面名を表すシステム変数である「_HMI_CurrentPage」へ代入することで画面移動を行います。

```

31 Sub PageChange2
32     '.ToString : 文字列変換を行う関数
33     Page_No_Str=NJ_1_PageNumber.ToString
34     '+ : 文字列を連結する演算子
35     Page_Str=Page_Str+Page_No_Str
36     _HMI_CurrentPage=Page_Str
37     'Page_Strを初期値へ戻す
38     Page_Str="Page"
39     Page_No=NJ_1_PageNumber
40 End Sub
  
```

4. 手順 3 で作成したコードの実行条件をグローバルイベントから設定します。右図では 100 ミリ秒ごとに手順 4 で作成したコードが実行されます。

▼ [2]	Interval
Interval	100 ミリ秒
▼ Actions	< 追加するアクションを選択し
▼ [0]	CallSubroutine
SubroutineName	SubroutineGroup0.PageChange2

5. NJ 変数「PageNumber」を変更することで PLC から画面の切替を行うことができます。

ウォッチウィンドウ (プロジェクト)			
コントローラ名	名称	モニタ値	変更
NJ_1	PageNumber	1	1

● Visual Basic を使用する方法(接続先が CJ の例)

接続先 PLC が CJ でも NJ の場合と同様に、Visual Basic を使用する事で PLC からの画面移動が可能です。

1. CJ 側で画面番号を格納する数値型変数を宣言します。

名称	データ型	アドレス / 値	ネットワーク変数
・ P_UF	BOOL	CF010	
— P_WR	WORD	A451	
🔌 CJ_BitVal1	BOOL	W0.00	公開のみ
🔌 CJ_BitVal2	BOOL	W0.01	公開のみ
🔌 CJ_NumVal	INT	D0	公開のみ
🔌 CJ_PageNumber	INT	D10	公開のみ
🔌 CJ_PageName	STRING(8)	D100	公開のみ

2. 手順 1 で宣言した変数を NA 側でデバイス変数割付を行い、画面名を格納する String 型変数と、NA 側で画面番号を格納する変数をそれぞれ宣言します。
「Page_Name」:画面名を表す変数
「Page_No」:NA 側の画面番号を表す変数

名称	データ型	初期値	割付先
CJ_BitVal1	Boolean		ExternalDevice0.CJ_BitVal1
CJ_BitVal2	Boolean		ExternalDevice0.CJ_BitVal2
CJ_NumVal	Short		ExternalDevice0.CJ_NumVal
CJ_PageName	String		
CJ_PageNumber	Short		ExternalDevice0.CJ_PageNumber
Page_Name	String		
Page_No	Short		

3. グローバルサブルーチンで右図のコードを記述します。
「PageNameSelect」:
CJ の変数の値を元に NA の画面を切り替えます。
「PageNumberSelect」:
現在表示している画面に対応する番号を PLC へ通知します。
どちらの場合も移動する画面を追加する場合は、Case の条件を増やしてください。

```

3 Sub PageNameSelect
4   Select Case CJ_PageNumber
5     Case 0
6       Page_Name="Page0"
7     Case 1
8       Page_Name="Page1"
9     Case 2
10      Page_Name="Page2"
11   End Select
12   Page_No=CJ_PageNumber
13   ShowPage(Page_Name)
14 End Sub

15 Sub PageNumberSelect
16   Select Case _HMI_Currentpage
17     Case "Page0"
18       CJ_PageNumber=0
19     Case "Page1"
20       CJ_PageNumber=1
21     Case "Page2"
22       CJ_PageNumber=2
23   End Select
24 End Sub
25
26

```

4. 手順 3 で作成したコードの実行条件をグローバルイベントから設定します。
①CJ 側の画面番号と NA 側の画面番号が等しくない状態を表します。
②実行するサブルーチン名を指定します。
③ページ切替が行われた時に動作します。
④実行するサブルーチン名を指定します。

Events		< 追加するイベントを選択します。 >
▼ [0]	①	Condition
Expression		CJ_PageNumber <> Page_No
▼ Actions		< 追加するアクションを選択します。 >
▶ [0]	②	CallSubroutine("SubroutineGroup0.PageNameSelect")
▼ [1]	③	Condition
Expression		_HMI_IsPageSwitching=False
▼ Actions		< 追加するアクションを選択します。 >
▶ [0]	④	CallSubroutine("SubroutineGroup0.PageNumberSelect")

5. CJから画面を切り替える場合はCJ変数「CJ_PageNumber」を変更します。
NAから画面を切り替える場合はNAシステム変数「_HMI_CurrentPage」を変更します。

新しい値を設定

アドレス:	D10	値変更(S)
現在値:	80	閉じる
値(V):	1	アドレス/型変更(E)
0 ~ 65535 (1CH)		2進表示(B) >>

3 改定履歴

改定履歴	改定年月	改定理由・改定ページ
A	2015 年 9 月	初版
B	2018 年 12 月	ご承諾事項・関連マニュアル変更

本誌には主に機種のご選定に必要な内容を掲載し、ご使用上の注意事項等は掲載していません。
ご使用上の注意事項、ご使用の際に必要な内容につきましては、必ずユーザーズマニュアルをお読みください。

- 本誌に記載の標準価格はあくまで参考であり、確定されたユーザ購入価格を表示したものではありません。本誌に記載の標準価格には消費税が含まれておりません。
- 本誌に記載されているアプリケーション事例は参考用ですので、ご採用に際しては機器・装置の機能や安全性をご確認の上、ご使用ください。
- 本誌に記載のない条件や環境での使用、および原子力制御・鉄道・航空・車両・燃焼装置・医療機器・娯楽機械・安全機器、その他人命や財産に大きな影響が予測されるなど、特に安全性が要求される用途に使用される際には、当社の意図した特別な商品用途の場合や特別の合意がある場合を除き、当社は当社商品に対して一切保証をいたしません。
- 本製品の内、外国為替及び外国貿易法に定める輸出許可、承認対象貨物(又は技術)に該当するものを輸出(又は非居住者に提供)する場合は同法に基づく輸出許可、承認(又は役務取引許可)が必要です。

オムロン株式会社 インダストリアルオートメーションビジネスカンパニー

●製品に関するお問い合わせ先

お客様相談室



クイック オムロン
0120-919-066

携帯電話・PHS・IP電話などではご利用いただけませんので、下記の電話番号へおかけください。

電話 **055-982-5015** (通話料がかかります)

■営業時間：8:00～21:00 ■営業日：365日

●FAXやWebページでもお問い合わせいただけます。

FAX **055-982-5051** / **www.fa.omron.co.jp**

●その他のお問い合わせ

納期・価格・サンプル・仕様書は貴社のお取引先、または貴社担当オムロン販売員にご相談ください。
オムロン制御機器販売店やオムロン販売拠点は、Webページでご案内しています。

オムロン制御機器の最新情報をご覧ください。

www.fa.omron.co.jp

緊急時のご購入にもご利用ください。

オムロン商品のご用命は