

SoMachine Basic

オペレーティングガイド

12/2017

本書の情報には本書に記載された製品についての一般的説明および性能の技術特性が含まれません。本書は、お客様の特定の用途に対する本製品の適合性または信頼性を確約するために作成されたものではありません。お客様またはインテグレーター様は自らの責任で、関連する特定の用途またはその使用に関する本製品のリスク分析、評価、および試験を完全かつ適切に行なってください。シュナイダーエレクトリック社あるいは系列会社（以下、シュナイダーエレクトリックと称します）は、本書に記載された情報の誤用に対して一切の責任を負いかねますので、あらかじめご了承ください。本書の内容について改善点や修正点の提案がある場合、また何らかの誤りを発見した場合には、弊社までご連絡ください。

媒体の如何を問わず本書の内容の一部およびすべてを、シュナイダーエレクトリックの書面の明示による許可なしに、個人または非商業的使用以外の目的で複製することを禁じます。また、本書およびその内容へリンクを張ることを禁じます。シュナイダーエレクトリックは、使用者自身の責任において「現状有姿」のまま閲覧する非独占的権利を除き、本書およびその内容の個人または非商業的使用に対して、いかなる権利またはライセンスを許諾しません。その他著作権も所有しており、無断複写、転載を禁じます。

本製品を設置して使用する際には、関連する州、地域、地区の安全規定をすべて順守する必要があります。安全のため、また、記録されたシステムデータの適合性を確保するため、部品の修理は製造業者にお任せください。

装置を技術的な安全要件がある用途に使用する場合、関連する指示に従ってください。

シュナイダーエレクトリックのハードウェア製品には必ず、シュナイダーエレクトリック製のソフトウェアまたは承認されたソフトウェアをご使用ください。この指示に従わない場合、人的損害、物的損害、また不適切な動作が生じる可能性があります。

この情報に従わない場合、人的損害や装置の損傷を招くおそれがあります。

Copyright © 2017 Schneider Electric Japan Holdings Ltd. All Rights Reserved.



	安全に関する使用上の注意	7
	本書について	9
第 I 部	SoMachine Basic 入門	15
第 1 章	SoMachine Basic の概要	17
1.1	システム要件と対応デバイス	18
	システム要件	19
	対応デバイス	20
	サポートされているプログラミング言語	21
1.2	SoMachine Basic ユーザーインタフェース	22
	SoMachine Basic によるプログラム作成	23
	SoMachine Basic によるプログラム開発	24
	SoMachine Basic のナビゲーション	25
	オペレーティングモード	26
第 2 章	SoMachine Basic の使いはじめに	27
2.1	開始ページ	28
	開始ページの概要	29
	SoMachine Basic の登録	30
	プロジェクトウィンドウ	31
	接続ウィンドウ	33
	アプリケーションの直接ダウンロード	37
	メモリー管理	38
	プロジェクトプレートウィンドウ	39
	ヘルプウィンドウ	40
第 II 部	SoMachine Basic のアプリケーションの開発	41
第 3 章	SoMachine Basic ウィンドウ	43
3.1	SoMachine Basic ウィンドウの概要	44
	ツールバーのボタン	45
	ステータスエリア	47
	システム設定	49
	レポートの印刷	51
第 4 章	プロパティ	53
4.1	プロパティウィンドウの概要	54
	プロパティウィンドウ	55
	プロジェクトプロパティ	56
第 5 章	設定	59
5.1	構成ウィンドウの概要	60
	設定ウィンドウの概要	61
	設定の構築	62
第 6 章	プログラミング	63
6.1	プログラミングワークスペースの概要	64
	プログラミングワークスペースの概要	64

6.2	特殊機能	65
	オブジェクト	66
	シンボルアドレス指定	67
	メモリーの割り当て	69
	ラダー/リスト可逆性	70
6.3	プログラムの動作とタスクの設定	74
	アプリケーションの動作	75
	タスクとスキャンモード	78
6.4	POU の管理	80
	POU	81
	タスクと関連した POU を管理する	82
	ラングの管理	84
	グラフセ (SFC) POU の管理	87
	Free POU	89
6.5	ユーザー定義ファンクション	92
	ユーザー定義ファンクションの作成	93
	ユーザー定義ファンクションの定義	94
	ユーザー定義ファンクションの管理	97
6.6	ユーザー定義ファンクションブロック	99
	ユーザー定義ファンクションブロックの作成	100
	ユーザー定義ファンクションブロックの定義	101
	ユーザー定義ファンクションブロックの管理	103
6.7	マスタータスク	105
	マスタータスクの説明	106
	マスタータスクの設定	107
6.8	文字列	108
	ワード型定数の文字列設定	109
	ワードメモリー内の文字列の割り当て	110
	文字列の管理	111
6.9	周期タスク	114
	周期タスクの作成	115
	周期タスクのスキャン期間設定	117
6.10	イベントタスク	118
	イベントタスクの概要	119
	イベントソース	120
	イベント優先度	121
	イベントタスクの表示	122
6.11	ツールを使う	125
	メッセージ	126
	アニメーションテーブル	128
	メモリーオブジェクト	133
	システムオブジェクト	136
	I/O オブジェクト	137
	ネットワークオブジェクト	138
	ソフトウェアオブジェクト	139
	PTO オブジェクト	140
	ドライブオブジェクト	141

	通信オブジェクト	142
	検索と置換	143
	クロスリファレンス	145
	シンボルリスト	146
	メモリー消費量の表示	150
6.12	ラダー言語のプログラミング	152
	ラダー図の概要	153
	ラダー図プログラミングの原則について	155
	ラングのカラーコード	157
	ラダープログラムのグラフィック要素	158
	比較ブロック	163
	演算ブロック	164
	コメントの追加	167
	プログラミングの推奨方法	168
6.13	命令リスト (IL) 言語のプログラミング	171
	IL プログラムの概要	172
	リスト命令の動作	174
	リスト言語の命令	175
	括弧の使用	178
6.14	グラフセ (リスト) プログラミング	180
	グラフセ (リスト) プログラミングの詳細	181
	グラフセ (リスト) プログラム構成	182
	SoMachine Basic プログラムでのグラフセ (リスト) の使い方	185
6.15	グラフセ (SFC) プログラミング	187
	グラフセ (SFC) プログラミングの概要	188
	グラフセ (SFC) グラフィカルエディターの使用	191
	分岐	194
	プログラミング推奨方法	198
6.16	オンラインモードでのデバッグ	199
	トレースウィンドウ	200
	値の変更	202
	値の強制	203
	オンラインモードでの変更	204
第 7 章	通信	209
7.1	通信ウィンドウの概要	210
	通信ウィンドウの概要	210
7.2	ロジックコントローラーとの接続	211
	ロジックコントローラーとの接続	212
	アプリケーションのダウンロードとアップロード	217
7.3	コントローラーの更新	221
	コントローラーのファームウェアの更新	221
7.4	メモリー管理	222
	ロジックコントローラーのメモリー管理	222
7.5	コントローラー情報	227
	コントローラー情報	227
7.6	RTC の管理	229
	RTC の管理	229

第 8 章	シミュレーター	231
	SoMachine Basic シミュレーターの概要	232
	SoMachine Basic シミュレーター I/O マネージャーウィンドウ	234
	SoMachine Basic シミュレーター時間管理ウィンドウ	236
	SoMachine Basic シミュレーターを使った値の変更	238
	SoMachine Basic シミュレーターの使い方	242
	Vijeo-Designer でのシミュレーターの起動	243
第 9 章	プロジェクトの保存と SoMachine Basic の終了	245
	プロジェクトの保存	246
	プロジェクトをテンプレートとして保存	247
	SoMachine Basic の終了	248
	付録	249
付録 A	Twido プロジェクトを SoMachine Basic に変換します	251
	Twido プロジェクトを変換 SoMachine Basic	251
付録 B	SoMachine Basic ショートカットキー	261
	SoMachine Basic キーボードショートカット	261
用語集	265
索引	269

安全に関する使用上の注意



重要情報

お断り

本書をよくお読みいただき、装置の正しい取り扱いと機能を十分ご理解いただいた上で、設置、操作、保守を行ってください。本書および装置には以下の表示が使われています。これらは潜在的な危険を警告したり、手順を明確化あるいは簡素化する情報について注意を呼びかけるものです。



この記号が「危険」または「警告」安全ラベルに追加されると、電気的な危険が存在し、指示に従わないと人身傷害の危険があることを示します。



安全警告記号です。人的傷害の危険性があることを警告します。
この記号の後に記載された安全に関する情報に従って、人的傷害や死亡の危険性を回避してください。

⚠ 危険

危険は、危険が生じる可能性のある状況を示します。回避しないと、死亡や重傷を招きます。

⚠ 警告

警告は、危険が生じる可能性のある状況を示します。回避しないと、死亡や重傷を招くおそれがあります。

⚠ 注意

注意は、危険が生じる可能性のある状況を示します。回避しないと、軽傷を招くおそれがあります。

注記

この表示は、指示に従わないと物的損害を負う可能性があることを示します。

注意

電子機器の設置、操作、整備は必ず資格のある人物が行ってください。Schneider Electric は、本資料の使用に起因するいかなる結果についても責任を負わないものとします。

資格のある人物とは、電子機器の構造、操作、設置に関する技術および知識を有し、かつ電子機器に伴う危険性を理解しこれを回避するための安全研修を受けた人物を指します。



概要

本書の適用範囲

本書では、SoMachine Basic を使用してロジックコントローラーのアプリケーションを設定、プログラムおよび通信する方法について説明します。

有効性に関する注意

本書は、SoMachine Basic 製品を対象として書かれています。

本書は、SoMachine Basic V1.6 のリリース時に更新されました。

本書に記載された機器の技術特性は、オンラインページにも表示されています。この情報にオンラインでアクセスするには、以下を実行します。

ステップ	アクション
1	シュナイダーエレクトリックのホームページに移動します： www.schneider-electric.com 。
2	検索 ボックスに製品の参照番号または製品ライン名を入力します。 <ul style="list-style-type: none">● 参照番号または製品ライン名にはスペースを含めないようにしてください。● 類似するモジュールのグループに関する情報を表示するには、アスタリスク (*) を使用します。
3	参照番号を入力した場合は、 製品データシート 検索結果に移動して目的の参照番号をクリックします。 製品ラインを入力した場合は、 製品ライン 検索結果に移動して目的の製品ラインをクリックします。
4	製品 検索結果に複数の結果が表示された場合は、目的の参照番号を選んでクリックします。
5	画面サイズによっては、データシート全体を表示するには画面をスクロールダウンしなければなりません場合があります。
6	データシートを .pdf ファイルとして保存または印刷するには、 XXX 製品のデータシートをダウンロード をクリックします。

シュナイダーエレクトリックでは、本マニュアル内に記載された製品特性とオンラインページの記載内容が一致するよう務めていますが、継続的改善を目指す当社の方針に従い、情報をより明確かつ正確なものにするため内容を改訂させていただく場合があります。マニュアルとオンラインページの情報が一致していない場合は、オンラインページの情報を参照してください。

関連マニュアル

マニュアルタイトル	参照番号
SoMachine Basic Generic Functions - Library Guide	<u>EIO0000001474 (ENG)</u> <u>EIO0000001475 (FRE)</u> <u>EIO0000001476 (GER)</u> <u>EIO0000001477 (SPA)</u> <u>EIO0000001478 (ITA)</u> <u>EIO0000001479 (CHS)</u> <u>EIO0000001480 (POR)</u> <u>EIO0000001481 (TUR)</u>
Modicon M221 Logic Controller Advanced Functions - Library Guide	<u>EIO0000002007 (ENG)</u> <u>EIO0000002008 (FRE)</u> <u>EIO0000002009 (GER)</u> <u>EIO0000002010 (SPA)</u> <u>EIO0000002011 (ITA)</u> <u>EIO0000002012 (CHS)</u> <u>EIO0000002013 (POR)</u> <u>EIO0000002014 (TUR)</u>
Modicon M221 Logic Controller プログラミングガイド	<u>EIO0000001360 (ENG)</u> <u>EIO0000001361 (FRE)</u> <u>EIO0000001362 (GER)</u> <u>EIO0000001363 (SPA)</u> <u>EIO0000001364 (ITA)</u> <u>EIO0000001365 (CHS)</u> <u>EIO0000001368 (POR)</u> <u>EIO0000001369 (TUR)</u>
Modicon M221 Logic Controller ハードウェアガイド	<u>EIO0000001384 (ENG)</u> <u>EIO0000001385 (FRE)</u> <u>EIO0000001386 (GER)</u> <u>EIO0000001387 (SPA)</u> <u>EIO0000001388 (ITA)</u> <u>EIO0000001389 (CHS)</u> <u>EIO0000001370 (POR)</u> <u>EIO0000001371 (TUR)</u>
Modicon TMC2 Cartridge - Programming Guide	<u>EIO0000001782 (ENG)</u> <u>EIO0000001783 (FRE)</u> <u>EIO0000001784 (GER)</u> <u>EIO0000001785 (SPA)</u> <u>EIO0000001786 (ITA)</u> <u>EIO0000001787 (CHS)</u> <u>EIO0000001788 (POR)</u> <u>EIO0000001789 (TUR)</u>
Modicon TMC2 Cartridge - Hardware Guide	<u>EIO0000001768 (ENG)</u> <u>EIO0000001769 (FRE)</u> <u>EIO0000001770 (GER)</u> <u>EIO0000001771 (SPA)</u> <u>EIO0000001772 (ITA)</u> <u>EIO0000001773 (CHS)</u> <u>EIO0000001774 (POR)</u> <u>EIO0000001775 (TUR)</u>

マニュアルタイトル	参照番号
Modicon TM3 Expansion Modules Configuration - Programming Guide	<u>EIO0000001396 (ENG)</u> <u>EIO0000001397 (FRE)</u> <u>EIO0000001398 (GER)</u> <u>EIO0000001399 (SPA)</u> <u>EIO0000001400 (ITA)</u> <u>EIO0000001401 (CHS)</u> <u>EIO0000001374 (POR)</u> <u>EIO0000001375 (TUR)</u>
Modicon TM3 Digital I/O Modules - Hardware Guide	<u>EIO0000001408 (ENG)</u> <u>EIO0000001409 (FRE)</u> <u>EIO0000001410 (GER)</u> <u>EIO0000001411 (SPA)</u> <u>EIO0000001412 (ITA)</u> <u>EIO0000001413 (CHS)</u> <u>EIO0000001376 (POR)</u> <u>EIO0000001377 (TUR)</u>
Modicon TM3 Analog I/O Modules - Hardware Guide	<u>EIO0000001414 (ENG)</u> <u>EIO0000001415 (FRE)</u> <u>EIO0000001416 (GER)</u> <u>EIO0000001417 (SPA)</u> <u>EIO0000001418 (ITA)</u> <u>EIO0000001419 (CHS)</u> <u>EIO0000001378 (POR)</u> <u>EIO0000001379 (TUR)</u>
Modicon TM3 Expert Modules - Hardware Guide	<u>EIO0000001420 (ENG)</u> <u>EIO0000001421 (FRE)</u> <u>EIO0000001422 (GER)</u> <u>EIO0000001423 (SPA)</u> <u>EIO0000001424 (ITA)</u> <u>EIO0000001425 (CHS)</u> <u>EIO0000001380 (POR)</u> <u>EIO0000001381 (TUR)</u>
Modicon TM3 Safety Modules - Hardware Guide	<u>EIO0000001831 (ENG)</u> <u>EIO0000001832 (FRE)</u> <u>EIO0000001833 (GER)</u> <u>EIO0000001834 (SPA)</u> <u>EIO0000001835 (ITA)</u> <u>EIO0000001836 (CHS)</u> <u>EIO0000001837 (POR)</u> <u>EIO0000001838 (TUR)</u>
Modicon TM3 Transmitter and Receiver Modules - Hardware Guide	<u>EIO0000001426 (ENG)</u> <u>EIO0000001427 (FRE)</u> <u>EIO0000001428 (GER)</u> <u>EIO0000001429 (SPA)</u> <u>EIO0000001430 (ITA)</u> <u>EIO0000001431 (CHS)</u> <u>EIO0000001382 (POR)</u> <u>EIO0000001383 (TUR)</u>

マニュアルタイトル	参照番号
Modicon TM2 Expansion Modules Configuration - Programming Guide	<u>EIO0000000396 (ENG)</u> <u>EIO0000000397 (FRE)</u> <u>EIO0000000398 (GER)</u> <u>EIO0000000399 (SPA)</u> <u>EIO0000000400 (ITA)</u> <u>EIO0000000401 (CHS)</u>
Modicon TM2 Digital I/O Modules - Hardware Guide	<u>EIO0000000028 (ENG)</u> <u>EIO0000000029 (FRE)</u> <u>EIO0000000030 (GER)</u> <u>EIO0000000031 (SPA)</u> <u>EIO0000000032 (ITA)</u> <u>EIO0000000033 (CHS)</u>
Modicon TM2 Analog I/O Modules - Hardware Guide	<u>EIO0000000034 (ENG)</u> <u>EIO0000000035 (FRE)</u> <u>EIO0000000036 (GER)</u> <u>EIO0000000037 (SPA)</u> <u>EIO0000000038 (ITA)</u> <u>EIO0000000039 (CHS)</u>
SR2MOD02 and SR2MOD03 Wireless Modem - User Guide	<u>EIO0000001575 (ENG)</u>

ソフトウェアマニュアルなど、本製品に関連するマニュアルは、弊社サポート専用サイトからダウンロードできます。<http://www.schneider-electric.com/en/download>

警告

制御不能

- 制御手法の設計者は制御バスの障害モードが発生するおそれを考慮する必要があり、特定の重要制御機能については、バス障害の最中および終了後に安全な状態を実現するための方策を準備しておく必要があります。重要制御機能の例としては、緊急停止、オーバートラベル停止、停電、および再起動があります。
- 重要な制御機能に対しては、別のまたは冗長性のある制御バスを用意してください。
- システム制御バスには、データ通信が含まれることがあります。予期しないデータの転送遅れや障害について考慮する必要があります。
- あらゆる事故防止規制および地域の安全性ガイドライン¹を遵守してください。
- 運用を開始する前に、各実装について、正しく動作するかどうかを個別に十分にテストする必要があります。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

¹ 詳細は、NEMA ICS 1.1 (最新版)、"Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control"、および NEMA ICS 7.1 (最新版)、"Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems"、または該当地域での同等のガイドラインを参照してください。

警告

装置の意図しない動作

- 本装置には、Schneider Electric 認定のソフトウェアのみ使用してください。
- ハードウェアの設定を変更した場合は、必ずアプリケーションプログラムも更新してください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

標準から派生した用語

技術用語、専門用語、シンボル、本書の記述、また本製品での表示は、国際規格用語および定義に由来しています。

安全機能システム、ドライブ、一般オートメーションにおいて、用語は、**安全性、安全機能、安全状態、異常、異常リセット、誤動作、障害、エラー、エラーメッセージ、危険等**を含みますが、それに限定されません。

特に以下の規格が含まれます。

規格	詳細
EN 61131-2: 2007	プログラマブルコントローラー、第 2 部：機器要件、および試験
ISO 13849-1: 2008	機械類の安全性：制御システムの安全関連部設計の一般原則
EN 61496-1: 2013	機械類の安全性：電氣的検知保護装置 第 1 部：一般要件、および試験
ISO 12100: 2010	機械類の安全性 - 設計の一般原則 - リスク評価とリスク低減
EN 60204-1: 2006	機械類の安全性 - 機械の電気装置 - 第 1 部：一般要件
EN 1088: 2008 ISO 14119: 2013	機械類の安全性 - ガードと共同するインターロック装置 - 設計、および選択のための原則
ISO 13850: 2006	機械類の安全性 - 非常停止 - 設計原則
EN/IEC 62061: 2005	機械類の安全性 - 安全関連の電気・電子・プログラマブル電子制御システムの機能安全
IEC 61508-1: 2010	電気・電子・プログラマブル電子安全関連系の機能安全：一般要求事項
IEC 61508-2: 2010	電気・電子・プログラマブル電子安全関連系の機能安全：電気・電子・プログラマブル電子安全関連系に対する要求事項
IEC 61508-3: 2010	電気・電子・プログラマブル電子安全関連系の機能安全：ソフトウェア要求事項
IEC 61784-3: 2008	計測制御用デジタルデータ通信：機能安全フィールドバス
2006/42/EC	機械指令
2014/30/EU	電磁両立性指令
2014/35/EU	低電圧指令

本書で使われている用語には下記の規格も含まれています。

規格	詳細
IEC 60034 シリーズ	回転電気機械
IEC 61800 シリーズ	可変速電気駆動システム
IEC 61158 シリーズ	計測制御用デジタルデータ通信 - 産業制御システム用のフィールドバス

動作領域は特定の危険性記述と併せて使われる場合があり、**機械指令 (2006/42/EC)** と **ISO 12100: 2010** の **危険区域** と同様に定義されています。

注記：前述の規格は、本書記載の特定の機器には適用されない場合があります。本書に記載されている製品の適用規格についての詳細は製品の特徴が記載された表を参照してください。

第 I 部

SoMachine Basic 入門

このパートについて

このパートには次の章が含まれています。

章	章タイトル	参照ページ
1	SoMachine Basic の概要	17
2	SoMachine Basic の使いはじめに	27

第 1 章

SoMachine Basic の概要

この章について

この章には次のセクションが含まれています。

セクション	項目	参照ページ
1.1	システム要件と対応デバイス	18
1.2	SoMachine Basic ユーザーインターフェース	22

1.1

システム要件と対応デバイス

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
システム要件	19
対応デバイス	20
サポートされているプログラミング言語	21

システム要件

概要

SoMachine Basic ソフトウェアをインストールする PC の最小システム要件は次のとおりです。

- プロセッサー Intel Core 2 Duo 以上
- RAM 1 GB
- 表示解像度 1280 x 768 ピクセル以上
- 以下のオペレーティングシステムの 32 ビット、または 64 ビット版
 - Microsoft Windows 7
 - Microsoft Windows 8
 - Microsoft Windows 8.1
 - Microsoft Windows 10

対応デバイス

M221 ロジックコントローラー

M221 ロジックコントローラー設定の詳細は、以下のプログラミングガイド、およびハードウェアガイドを参照してください。

ロジックコントローラータイプ	ハードウェアガイド	プログラミングガイド
M221 ロジックコントローラー	Modicon M221 Logic Controller ハードウェアガイド	Modicon M221 Logic Controller プログラミングガイド

TM3 拡張モジュール

モジュール設定の詳細は、各拡張モジュールタイプのプログラミングガイド、およびハードウェアガイドを参照してください。

拡張モジュールタイプ	ハードウェアガイド	プログラミングガイド
TM3 デジタル I/O 拡張モジュール	TM3 Digital I/O Expansion Modules Hardware Guide	TM3 Expansion Modules Programming Guide
TM3 アナログ I/O 拡張モジュール	TM3 Analog Modules Hardware Guide	
TM3 エキスパート I/O 拡張モジュール	TM3 Expert I/O Modules Hardware Guide	
TM3 セーフティモジュール	TM3 Safety Modules Hardware Guide	
TM3 送受信機モジュール	TM3 Transmitter and Receiver Modules Hardware Guide	

TM2 拡張モジュール

モジュール設定の詳細は、各拡張モジュールタイプのプログラミングガイド、およびハードウェアガイドを参照してください。

拡張モジュールタイプ	ハードウェアガイド	プログラミングガイド
TM2 デジタル I/O モジュール	TM2 Digital I/O Modules Hardware Guide	TM2 Expansion Modules Programming Guide
TM2 アナログ I/O モジュール	TM2 Analog I/O Modules Hardware Guide	

TMC2 カートリッジ

カートリッジ設定の詳細は、以下のプログラミングガイド、およびハードウェアガイドを参照してください。

カートリッジタイプ	ハードウェアガイド	プログラミングガイド
TMC2 カートリッジ	TMC2 Cartridges Hardware Guide	TMC2 Cartridges Programming Guide

TMH2GDB リモートグラフィック表示

リモートグラフィック表示のインストール、互換性、設定、および操作についての詳細は、以下のガイドを参照してください。

表示器タイプ	ユーザーガイド
リモートグラフィック表示	TMH2GDB リモートグラフィック表示 User Guide

サポートされているプログラミング言語

概要

プログラマブル ロジック コントローラーは、制御プログラムに基づいて入力の読み出し、出力の書き込み、およびロジックの解決を行います。サポートされているプログラミング言語を使用して、ロジックコントローラーの制御プログラムを作成します。

SoMachine Basic は、以下の IEC-61131-3 プログラミング言語をサポートしています。

- ラダー図言語
- 命令リスト (IL) 言語
- グラフセ (リスト)
- グラフセ (SFC)

1.2 SoMachine Basic ユーザーインターフェース

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
SoMachine Basic によるプログラム作成	23
SoMachine Basic によるプログラム開発	24
SoMachine Basic のナビゲーション	25
オペレーティングモード	26

SoMachine Basic によるプログラム作成

概要

SoMachine Basic は、ロジックコントローラー用のプログラムを容易に設定、開発、および通信するために設計されたグラフィカル プログラミング ツールです。

重要な用語

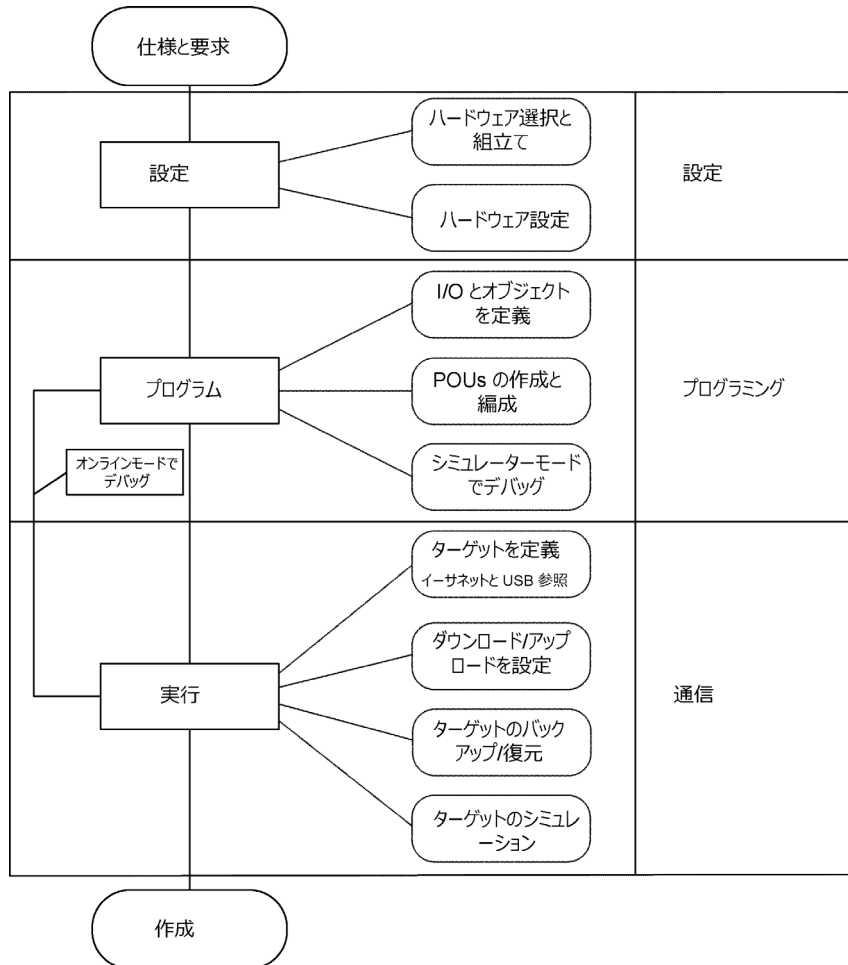
SoMachine Basic は以下の用語を使用します。

- **プロジェクト** : SoMachine Basic のプロジェクトは、プロジェクトの開発者と目的、プロジェクトのロジックコントローラーや拡張モジュールの構成、プログラム ソース コード、シンボル、コメント、ドキュメント、およびその他すべての関連情報を含みます。
- **アプリケーション** : ロジックコントローラーにダウンロードされたプロジェクトのすべてを含みます。コンパイルされたプログラム、ハードウェア設定情報、および非プログラムデータ (プロジェクトのプロパティ、シンボル、およびドキュメント) など。
- **プログラム** : ロジックコントローラーで実行されるコンパイルされたソースコード。
- **POU** (program organization unit プログラム構成単位) : 変数宣言とプログラムで使用される一連の命令を含む再利用可能なオブジェクト。

SoMachine Basic によるプログラム開発

はじめに

SoMachine Basic (**設定**、**プログラミング**、および**通信**タブ) で一般的なプロジェクトの開発をする段階を次の表に示します。



SoMachine Basic のナビゲーション

開始ページ

SoMachine Basic を起動すると**開始ページ**ウィンドウが表示されます。このウィンドウから、SoMachine Basic の登録、ロジックコントローラーの接続管理、およびプロジェクトの作成や選択を行います。

モジュールエリア

SoMachine Basic でプロジェクトを選択するとメインウィンドウが表示されます。

メインウィンドウの上部にはツールバー (45 ページ参照) があります。ツールバーにはよく使用するタスクのアイコンがあり、**開始ページ**ウィンドウに戻るアイコンも含まれます。

ツールバーの隣にあるステータスバー (47 ページ参照) には、ロジックコントローラーとの接続状況が表示されます。

その下でメインウィンドウはモジュールの数に分割されています。各モジュールはそれぞれの開発段階をコントロールしています。モジュールエリア上部のタブをクリックして各モジュールにアクセスします。モジュールを左から順に追ってアプリケーションを開発します。

- **プロパティ** (53 ページ参照)
プロジェクトのプロパティを設定します。
- **設定** (59 ページ参照)
コントローラーロジックと拡張モジュールのハードウェア設定を定義します。
- **プログラミング** (63 ページ参照)
サポートされているプログラミング言語を使ってプログラムを開発します。
- **表示** (Modicon M221, ロジックコントローラー, プログラミングガイド参照)
TMH2GDB リモートグラフィック表示モジュールのオペレータインタフェースを構築します。
- **通信** (209 ページ参照)
SoMachine Basic とロジックコントローラーの接続管理、アプリケーションのアップロード/ダウンロード、テスト、および通信を行います。

オペレーティングモード

はじめに

オペレーティングモードでは、コントローラーが SoMachine Basic に接続されているかどうかにかかわらず、アプリケーションの開発、デバッグ、監視、および変更が可能です。

SoMachine Basic は以下のモードで動作します。

- オフラインモード
- オンラインモード
- シミュレーターモード

オフラインモード

ロジックコントローラーと物理的に接続されていない場合、SoMachine Basic はオフラインモードで動作します。

オフラインモードの SoMachine Basic で使用するハードウェアコンポーネントを設定し、アプリケーションの開発をします。

オンラインモード

ロジックコントローラーが PC に物理的に接続されている場合、SoMachine Basic はオンラインモードで動作します。

オンラインモードでは、アプリケーションをロジックコントローラーにダウンロードできません (アプリケーションはシミュレーターロジックコントローラーに直接保存されるので、シミュレーターモードでアプリケーションのダウンロード、およびアップロードはできません)。SoMachine Basic で PC 上のアプリケーションとロジックコントローラーに保存されたバージョンの同期をとるので、アプリケーションのデバッグ、監視、および変が可能になります。

オンラインモードではプログラムの特定の要素を変更することができます。例えば、ラングの追加や削除、特定のファンクションブロックパラメーターの値を変更できます。

注記： オンラインプログラムの変更は事前に定義された設定に依存します。メモリー管理 (38 ページ参照) を参照してください。詳細は、オンラインモードでのデバッグ (199 ページ参照) を参照してください。

シミュレーターモード

シミュレートされたロジックコントローラーに接続されている場合、SoMachine Basic はシミュレーターモードで動作します。シミュレーターモードでは、ロジックコントローラーへの物理的な接続は確立されません。SoMachine Basic はロジックコントローラーと拡張モジュールへの接続をシミュレートし、プログラムを実行してテストします。

詳細は、SoMachine Basic シミュレーター (232 ページ参照) を参照してください。

第 2 章

SoMachine Basic の使いはじめに

2.1 開始ページ

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
開始ページの概要	29
SoMachine Basic の登録	30
プロジェクトウィンドウ	31
接続ウィンドウ	33
アプリケーションの直接ダウンロード	37
メモリー管理	38
プロジェクトテンプレートウィンドウ	39
ヘルプウィンドウ	40

開始ページの概要

概要

SoMachine Basic を起動すると開始ページウィンドウが最初に表示されます。

開始ページウィンドウには以下のウィンドウがあります。

- **レジスター** (30 ページ参照)
SoMachine Basic の登録、およびライセンス詳細の表示をします。
- **プロジェクト** (31 ページ参照)
新規プロジェクトの作成、または既存プロジェクトを開きます。
- **接続** (33 ページ参照)
ロジックコントローラーへの接続、アプリケーションのコントローラーへのアップロード / ダウンロード、コントローラーメモリーのバックアップ / 復元、および接続されたロジックコントローラーの LED の点滅を行います。
- **テンプレート** (39 ページ参照)
サンプルプロジェクトをテンプレートとして新規プロジェクトを作成します。
- **ヘルプ** (40 ページ参照)
オンラインヘルプ、関連ドキュメント、トレーニング資料、およびチュートリアルを表示します。
- **バージョン情報**
SoMachine Basic の情報を表示します。
- **終了**
SoMachine Basic を終了します。

SoMachine Basic の登録

概要

ソフトウェアの登録が必要になる前に、SoMachine Basic は 30 日間使用できます。登録するとソフトウェアを使用するための認証コードが送付されます。

SoMachine Basic を登録するとテクニカルサポートとソフトウェアアップデートが受けられます。

登録

SoMachine Basic の登録

手順	手順内容
1	開始ページウィンドウの 今すぐ登録 ボタンをクリックします。
2	登録ウィザードの手順に従って進みます。 ヘルプ ボタンをクリックすると詳細が表示され ます。

PC にインストールされたライセンスキーの詳細を表示するには、**開始ページウィンドウのバージョン情報**ボタンをクリックします。

プロジェクトウィンドウ




概要

プロジェクトウィンドウを使って、SoMachine Basic の新規プロジェクトを作成、既存の SoMachine Basic、TwidoSoft、または TwidoSuite のプロジェクトを開きます。

プロジェクトウィンドウの右側には役立つ情報へのリンクがあります。

SoMachine Basic のプロジェクトファイルを開く

次の手順で、プロジェクトファイルを開きます。

手順	手順内容
1	開始ページウィンドウのプロジェクトをクリックします。
2	以下のひとつを行います。 <ul style="list-style-type: none">● 最近使ったプロジェクトリストからプロジェクトをクリックします。● 新規プロジェクトの作成をクリックします。● 既存のプロジェクトを開くをクリックし、既存の SoMachine Basic のプロジェクトファイル (*.smbp)、またはサンプル プロジェクト ファイル (*.smbe) を選択します。
3	<p>ケース 1</p> <p>エラーウィンドウが表示された場合は、選択したプロジェクトが以前のバージョンの SoMachine Basic でダウンロード専用としてパスワード保護されていることを意味します。</p> <ol style="list-style-type: none">1. OK をクリックします。 結果：プロパティ タブが表示されます。2. プロジェクトの保護をクリックします。3.  をクリックして、プロジェクトのパスワードを入力します。4. プロジェクトの保護を解除する場合は、無効を選択し、適用をクリックします。 プロジェクトの保護を維持するには暗号化パスワードを入力し、表示とダウンロードを選択して適用をクリックします。 <p>ケース 2</p> <p>プロパティタブにエラーのアイコンが表示された場合は、選択したプロジェクトが以前のバージョンの SoMachine Basic で表示とダウンロードのみとしてパスワード保護されていることを意味します。</p> <ol style="list-style-type: none">1. プロパティタブ → プロジェクトの保護をクリックします。2. プロパティタブの  をクリックします。3. プロジェクトを暗号化するためのパスワードを入力します。 プロジェクトを保存するには、暗号化してください。4. 適用をクリックします。 <p>ケース 3</p> <p>パスワードの入力を求めるウィンドウが表示された場合、プロジェクトがパスワード保護されていることを意味します。</p> <ol style="list-style-type: none">1. 暗号化パスワードを入力します。2. 適用をクリックします。3. プロジェクトを変更するには：<ol style="list-style-type: none">a. プロパティタブの  をクリックします。 結果：パスワードの入力を求めるウィンドウが表示されます。b. 変更用パスワードを入力します。c. 適用をクリックします。 <p>結果：プロジェクトファイルが開き設定タブが表示されます。</p>

TwidoSuite、または TwidoSoft のプロジェクトファイルを開く

SoMachine Basic では、Twido プログラマブルコントローラー用に作成されたアプリケーションを開き、SoMachine Basic のプロジェクトファイルに変換することができます。

次の手順で、TwidoSuite、または TwidoSoft のプロジェクトファイルを開きます。

手順	手順内容
1	開始ページウィンドウのプロジェクトをクリックします。
2	<p>既存のプロジェクトを開くをクリックし、ファイルタイプリストから次のいずれかを選択し、以下の拡張子を持つ既存プロジェクトを選択します。</p> <ul style="list-style-type: none"> ● TwidoSuite プロジェクトファイル (*.xpr) ● Twido アーカイブプロジェクトファイル (*.xar) ● TwidoSoft プロジェクトファイル (*.twd) <p>選択した Twido プロジェクトファイルが TwidoSoft で開いている場合、TwidoSoft はプロジェクトファイルをロックするので、SoMachine Basic で開くことはできません。SoMachine Basic で開く前に、TwidoSoft でプロジェクトを閉じます。</p> <p>結果: 選択したプロジェクトファイルが開き設定タブが表示されます。</p>
3	<p>変換レポートウィンドウが表示されます。コントローラープラットフォームの変換に起因する異常が存在しないか、変換結果を慎重に調べます。異常を調整する方法については、Twido プロジェクトを SoMachine Basic に変換する (251 ページ参照) を参照してください。</p>

注記: TwidoSuite は %I0.0.1 (または %I0.0.7) を超高速カウンター (FC) (%VFC) ファンクションブロックのパルス入力として使用します。SoMachine Basic では、同等の高速カウンター (HSC) (%HSC) ファンクションブロックは %I0.0 (または %I0.6) を使用します。変換後にアプリケーションを適切に修正します。

一般に他のコントローラープラットフォームから M221 Logic Controller、および SoMachine Basic のプラットフォームへの変換は、プラットフォームの違いの限界まで対応しています。必然的に、前述の注記で説明されているような違いを調整する必要があります。

警告

装置の意図しない動作

- アプリケーションプログラムが変換前と同じように動作し、必要とされる正しい設定、パラメーター、パラメーターの値、ファンクション、およびファンクションブロックがあることを必ず確認してください。
- 以前の動作と一致するように、必要に応じてアプリケーションを変更してください。
- アプリケーションを運用する前に、新しくコンパイルされたバージョンを十分にテスト、および検証してください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

接続ウィンドウ

接続されたデバイス

接続ウィンドウには 2 つのデバイスリストが表示されます。

1. ローカルデバイス

PC に接続しているデバイスがすべて表示されます。PC がロジックコントローラーへのアクセスを提供します。

- PC の COM ポート経由 (例: COM1)
- USB ケーブル経由
- 仮想 COM ポート (USB/ シリアルコンバーター、または Bluetooth ドングル) 経由
- このリストに手動で追加したモデムとその電話番号経由

注記: COM ポートを選択し **Modbus ドライバーパラメーターを保持** チェックボックスを選択すると、Modbus ドライバーで定義されたパラメーターで通信が確立されます。

2. Ethernet デバイス

SoMachine Basic を実行している PC と同じ Ethernet サブネット上でアクセス可能なすべてのロジックコントローラーを表示します。ルーター、または UDP ブロードキャストをブロックするデバイスの背後にあるデバイスはリストされません。

SoMachine Basic で自動検知されたロジックコントローラー、および手動で追加したコントローラーがリストされます。

自動的に追加されるロジックコントローラー (**自動検出プロトコルを有効** オプションを選択時) に対してのみ **LED 点滅開始** ボタンを使用できます。




コントローラーを手動で追加

次の手順で、**Ethernet デバイス** リストにロジックコントローラーを追加します。

手順	手順内容
1	リモート参照フィールドに追加するロジックコントローラーの IP アドレス (例: 12.123.134.21) を入力します。
2	追加をクリックしてデバイスを Ethernet デバイス リストに追加します。

モデム接続の追加

以下に、ローカルデバイスリストにモデム接続を追加する手順を示します。

手順	手順内容
1	 モデム設定の追加ボタンをクリックします。 結果：モデム設定ウィンドウが表示されます。
2	<p>ドロップダウンリストからモデムの COM ポート を選択します。</p> 
3	<p>通信パラメーターを設定します。 モデムの設定パラメーターについての詳細は、以下の表を参照してください。</p>
4	<p>適用をクリックします。</p> <p>注記： このボタンはすべての設定が定義されると有効になります。</p> <p>結果： モデム接続は、ローカルデバイスリストに追加されます (例：COM2@0612345678、GenericModem)。</p>
5	<p>通信ウィンドウ (210 ページ参照) で、スレーブデバイスで定義されたアドレス (<i>Modicon M221, ロジックコントローラー, プログラミングガイド参照</i>) に一致する ユニット ID を入力します。</p>
6	<p>必要に応じてモデム設定を変更できます。変更するモデムをローカルデバイスリストで選択し、 モデム設定を変更ボタンをクリックします。</p>




モデム設定パラメーター

モデム設定のプロパティを次の表に示します。

プロパティ	値	初期値	説明
COM ポート	COMx	-	ドロップダウンリストからモデムの COM ポートを選択します。
デバイス	-	-	モデム名
電話番号	-	-	ロジックコントローラーに接続しているモデムの電話番号を入力します。 テキストフィールドですべての文字が 32 文字まで入力できます。設定を有効にするには、最低 1 文字の入力が必要です。
モデム初期化コマンド	-	AT&D0	モデムの AT 初期化命令を編集します。 AT 初期化命令はオプションです (フィールドが空白の場合、文字列 AT が送信されます)。
エスケープ文字	-	+	切断処理用のエスケープ文字を編集します。
ボーレート	1200 2400 4800 9600 19200 38400 57600 115200	19200	モデムのデータ転送速度を選択します。
パリティ	無し 偶数 奇数	偶数	エラー検知用に転送されたデータのパリティを選択します。
データビット	7 8	8	データビット数を選択します。
ストップビット	1 2	1	ストップビット数を選択します。
タイムアウト (ms)	0...60000	15000	タイムアウト (ms) を指定します。
文字間のタイムアウト (ms)	0...10000	10	フレーム間のタイムアウト (ms) を指定します。 自動チェックボックスを選択すると値は自動的に計算されます。

コントローラーに接続する

次の手順で、コントローラーを SoMachine Basic に接続します。

手順	手順内容
1	 (デバイスを更新ボタン) をクリックすると、接続されているデバイスのリストが更新されます。
2	<p>ローカルデバイス、または Ethernet デバイスリストからロジックコントローラーを 1 つ選択します。コントローラーが PC と同じネットワークに Ethernet 接続されている場合は、コントローラーの IP アドレスがリストに表示されます。リストで IP アドレスを選択すると  (IP アドレス設定ボタン) が有効になります。このボタンをクリックしてコントローラーの IP アドレスを変更します。</p> <p>注記： ポスト設定ファイルに書き込むチェックボックスを有効にすると、Ethernet パラメーターはポスト設定ファイルで変更され電源を切っても保持されます。</p>
3	<p>必要に応じて  (LED 点滅開始ボタン) をクリックし、選択したコントローラーの LED を点滅させコントローラーを識別します。このボタンを再度クリックすると LED 点滅が停止します。</p> <p>注記： 自動的に追加されるロジックコントローラー (自動検出プロトコルを有効オプションを選択時) に対してのみ LED 点滅開始ボタンを使用できます。</p>
4	<p>ログインをクリックしてコントローラーにログインします。ロジックコントローラーがパスワード保護されている場合は、パスワードの入力を要求されます。パスワードを入力し OK ボタンをクリックして接続します。</p> <p>結果： 接続の進行状況を示すステータスバーが表示されます。</p>
5	<p>接続が確立されると、ロジックコントローラーの詳細がウィンドウの選択されたコントローラーに表示され以下のボタンが有効になります。</p> <ul style="list-style-type: none"> ● コントローラーにアプリケーションをダウンロード： アプリケーションを SoMachine Basic で開かずにロジックコントローラーにダウンロードします。アプリケーションの直接ダウンロード (37 ページ参照) を参照してください。 ● メモリー管理： コントローラーメモリーを PC にバックアップ (222 ページ参照)、または PC から復元 (223 ページ参照) します。メモリー管理 (38 ページ参照) を参照してください。 ● アプリケーションをコントローラーからアップロードする： 接続されたロジックコントローラーのアプリケーションをアップロードして SoMachine Basic の新規プロジェクトを作成します。アプリケーションのアップロード (219 ページ参照) を参照してください。
6	<p>ログアウトボタンをクリックして接続しているコントローラーからログアウトします。</p>

アプリケーションの直接ダウンロード

概要

SoMachine Basic でプロジェクトを開かずに、プロジェクトファイル内のアプリケーションをロジックコントローラーにダウンロードします。プロジェクトが暗号化されていてパスワードなしではプロジェクトを開くことができない場合、この機能はとても便利です。

この方法はダウンロードのみ可能です。アプリケーションをロジックコントローラーから SoMachine Basic へアップロードするには、アプリケーションのアップロード (219 ページ参照) を参照してください。

アプリケーションの直接ダウンロード

アプリケーションを直接ロジックコントローラーにダウンロードする手順を示します。

手順	手順内容
1	SoMachine Basic を実行している PC をシリアルケーブル、USB、または Ethernet ケーブルを使用してロジックコントローラーに接続します。
2	開始ページウィンドウの 接続 タブを選択します。
3	ローカルデバイス 、または Ethernet デバイス リストからロジックコントローラーを選択し、 ログイン をクリックします。 結果 ：SoMachine Basic とロジックコントローラーの接続が設立されます。
4	コントローラーにアプリケーションをダウンロード をクリックします。
5	プロジェクトファイル フィールドで参照ボタンをクリックし、ダウンロードする SoMachine Basic プロジェクトファイル (*.smbp) を選択し 開く をクリックします。 選択したプロジェクト情報がウィンドウの 情報 エリアに表示されます。 <ul style="list-style-type: none">● プロジェクトファイルの暗号化およびパスワード保護の有無。● プロジェクトファイルに含まれる構成情報。検出されたロジックコントローラーシステムの構成が、選択したプロジェクトに含まれる構成と互換性があるかなど。
6	選択したプロジェクトファイルのアプリケーションは、SoMachine Basic でコンパイルされます。コンパイル中に検出されたエラーは、 コンパイルエラー に表示されます。SoMachine Basic でコンパイルエラーが検出された場合、アプリケーションはダウンロードできません。SoMachine Basic でプロジェクトを開きエラーを修正してからもう一度やり直します。
7	ダウンロードをする前に、以下ののボタンをクリックして現在のロジックコントローラーの状態を制御できます。 <ul style="list-style-type: none">● コントローラーの停止● コントローラーの開始● コントローラーの初期化
8	PC からコントローラー (ダウンロード) をクリックします。 結果 ：SoMachine Basic から接続したロジックコントローラーにアプリケーションがダウンロードされます。

メモリー管理

概要

接続ウィンドウの**メモリー管理**をクリックしてロジックコントローラーメモリーのバックアップ、または復元をします。

以下を選択します。

- PC にバックアップする (222 ページ参照)
- PC から復元する (223 ページ参照)

プロジェクトテンプレートウィンドウ

概要

サンプルプロジェクトを使用して SoMachine Basic の新規プロジェクトを作成できます。

プロジェクトテンプレートを開く

次の手順で、プロジェクトテンプレートを使用して新規プロジェクトを作成します。

手順	手順内容
1	開始ページ ウィンドウの テンプレート タブを選択します。
2	<p>ウィンドウの右上コーナーにあるテンプレートを検索の文字列フィールドを使ってプロジェクトの検索をします。文字を入力すると、SoMachine Basic はプロジェクト名、ウィンドウ下部にあるプロジェクトの説明文、およびプロジェクトプロパティを検索します。一致するプロジェクトのリストが、文字を入力すると同時に表示されます。</p> <p>プロジェクトリストにあるプロジェクトテンプレートファイル (*.smbe) を選択しテンプレートを開くをクリックします。</p> <p>結果：選択したテンプレートのコピーが新規プロジェクトとして作成されます。</p> <p>プロジェクトテンプレートにヘルプファイルがリンクされているプロジェクトは、関連しているヘルプを開くボタンをクリックして関連しているヘルプを開くを開きます。関連ヘルプがある場合は、プロジェクトリストの下にハイライトされます。</p> <p>注記：SoMachine Basic には、Vijeo-Designer アプリケーションファイル、およびサンプルプロジェクト用のシステムユーザーガイドも用意されています。プロジェクトにそれらのファイルが提供されているか、選択したプロジェクトの説明エリアを読んで確認します。ファイルが提供されている場合は、関連があるフォルダーを開くをクリックし、Windows エクスプローラーでプロジェクトテンプレートファイル (*.smbe) や Vijeo-Designer アプリケーションファイル (*.vdz) を参照します。</p>

ヘルプウィンドウ

概要

このウィンドウには SoMachine Basic の追加資料へのリンクが含まれます。

- SoMachine Basic オンラインヘルプシステム
- システムユーザーガイド (SUGs)、トレーニング資料、インストラクションシート、サンプルアプリケーションの説明など、関連する PDF ドキュメント
- e-Learning トレーニング資料
- チュートリアル
- Twido アプリケーションを SoMachine Basic 使うために変換する情報

第 II 部

SoMachine Basic のアプリケーションの開発

このパートについて

このパートには次の章が含まれています。

章	章タイトル	参照ページ
3	SoMachine Basic ウィンドウ	43
4	プロパティ	53
5	設定	59
6	プログラミング	63
7	通信	209
8	シミュレーター	231
9	プロジェクトの保存と SoMachine Basic の終了	245

第 3 章

SoMachine Basic ウィンドウ

3.1 SoMachine Basic ウィンドウの概要

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
ツールバーのボタン	45
ステータスエリア	47
システム設定	49
レポートの印刷	51















ツールバーのボタン




はじめに

SoMachine Basic のウィンドウ上部にツールバーが表示され、頻繁に使用されるファンクションにアクセスできます。

ツールバー

ツールバーには以下のボタンがあります。

アイコン	説明
	新規プロジェクトの作成 (CTRL+N)
	既存プロジェクトを開く (CTRL+O)
	現在のプロジェクトの保存 (CTRL+S) 下矢印をクリックすると保存オプションのメニューが表示されます。
	レポートの印刷 (CTRL+P) 下矢印をクリックして印刷 (51 ページ参照) するレポートを選択、またはレポートの内容と書式 (52 ページ参照) を設定します。
	切り取り (CTRL+X)
	コピー (CTRL+C)
	貼り付け (CTRL+V)
	元に戻す (CTRL+Z) 一度クリックすると、プログラムエディターで直前に実行した操作を元に戻すことができます。 下矢印をクリックして、すべての操作、どの操作まで、または選択した操作を元に戻すかリストから選択します。 最大 10 個の操作を元に戻せます。
	やり直す (CTRL+Y) 一度クリックすると、直前に実行した元に戻す操作をキャンセルできます。 下矢印をクリックして、すべての操作、どの操作まで、または選択した操作をやり直すかリストから選択します。 最大 10 個の操作をやり直せます。
	システム設定 (49 ページ参照) ウィンドウを表示します。
	下矢印をクリックし、リストからアクションを選択します。オンラインヘルプまたはコンテキストヘルプの表示、テンプレート、リリースノート、チュートリアルまたは e-Learning の閲覧、Schneider Electric テクニカルサポートへのお問い合わせができます。
	ロジックコントローラーを開始 (CTRL+M) オンラインモードでコントローラーが RUNNING ステート時以外に使用可能です。
	ロジックコントローラーを停止 (CTRL+L) オンラインモードでコントローラーが RUNNING ステート時に使用可能です。
	ロジックコントローラーを初期化します。オンラインモードでのみ使用可能です。

アイコン	説明
	プログラムをコンパイルします。
	選択したコントローラーにログイン (CTRL+G)、またはログアウト (CTRL+H) します。 注記 ： 選択されたコントローラーはこのボタンの左側に表示されます。
	SoMachine Basic シミュレーター (232 ページ参照) を起動 (CTRL+B)、または停止 (CTRL+W) します。

ステータスエリア

概要

メインウィンドウ上部のステータスエリアに現在のシステムステータスの情報が表示されます。



- 1 プログラムステータス**
プログラムに検出されたエラーの有無を表示します。
- 2 接続ステータス**
SoMachine Basic とロジックコントローラー、またはシミュレートされたロジックコントローラーとの接続状態を表示します。
- 3 コントローラステータス**
ロジックコントローラーの現在のステータス (RUNNING、STOPPED、HALTED など) を表示します。
- 4 スキャンタイム**
前回のスキャンタイムを表示します。
- 5 検出された最終コントローラーエラー**
直前に検出されたエラーを示します。ロジックコントローラーが STOPPED、または HALTED ステートの場合にシステムビット、およびシステムワードから情報が抽出されます。

ステータス エリア メッセージ

ステータスエリアに次のメッセージが表示されます。

メッセージタイプ	メッセージ	説明
プログラムステータス	[エラーなし]	プログラムに検出されたエラーはありません。
	[プログラムアドバイザーが検出されました]	プログラムは不完全です。
	[プログラムエラーを検出しました]	プログラムが存在しないか、プログラムにエラーがあります。
接続ステータス	[接続されていません]	SoMachine Basic はオフラインモードで実行されています。
	[オンライン]	SoMachine Basic はオンラインモードで実行されています。
コントローラステータス (オンラインモードのみ)	[接続されていません]	コントローラーは SoMachine Basic に接続されていません。
	[異常停止]	コントローラーは HALTED ステートです。 アプリケーションエラーが検出されたためコントローラーが停止しました。
	[停止]	コントローラーは STOPPED ステートです。 コントローラーにある有効なアプリケーションが停止しました。
	[実行]	コントローラーは RUNNING ステートです。 アプリケーションを実行しています。
	[Powerless]	コントローラーは POWERLESS ステートです。 コントローラーの電源供給は USB ケーブルからのみで、 USB でファームウェアのダウンロード / アップロードが可能です。

メッセージタイプ	メッセージ	説明
コントローラステータス (オンラインモードのみ)	[ファームウェアダウンロード]	ファームウェアをダウンロードしています。
	[ファームウェアエラー]	ファームウェアエラーを検出しました。コントローラーにダウンロード中のファームウェアのバージョンは現在のバージョンよりも古いです。
	[アプリケーションがありません]	コントローラーにアプリケーションがありません。
	[電源投入]	コントローラー起動中です (BOOTING)。
スキャンタイム (オンラインモードのみ)	[スキャンタイム 0 μs]	最新のスキャンタイム (マイクロ秒単位)
コントローラーが最後に 検出したエラー (オンラインモードのみ)	[エラーなし]	コントローラーに検出されたエラーはありません。
	[コントローラーは、 RUNNING のステートに 切り替えられませんでした]	コントローラーは実行できません。
	[バッテリー残量が低下]	コントローラーのバッテリー残量が低いです。
	[運転/停止入力]	運転/停止入力命令のためコントローラーが停止しました。
	[停止命令]	停止命令のためコントローラーは停止しました。
	[ソフトウェアエラーが検 出されました (コントロー ラーのスクリーンタイムを超えてい ます)。]	ソフトウェアがエラーを検出したためコントローラーが異常停止しました。コントローラーのスクリーンタイムのオーバーシュート。コントローラーのスクリーンタイムがユーザープログラムで設定された期間よりも長いです。
	[ハードウェアエラーが検 出されたため停止]	ハードウェアがエラーを検出したためコントローラーが停止しました。
	[停電]	停電のためコントローラーは停止しました。
	[コントローラーは '開始 停止' モードに設定されて います]	開始動作の設定により、コントローラーは自動アプリケーション実行モードで起動します。
	[Init コマンド]	コールドスタートの Init。
[不明な停止理由: {0}]	停止原因不明	

システムビットとシステムワードの全リストについては、ロジックコントローラーのプログラミングガイドを参照してください。

システム設定

概要

このウィンドウでは、SoMachine Basic の言語設定や、ラダーエディターのカスタマイズ、新規プロジェクトの作成で**設定**タブに表示されるデフォルトのロジックコントローラーの選択ができます。

ユーザーインターフェース言語の変更

次の手順で、ユーザーインターフェースの言語を変更します。

手順	手順内容
1	システム設定ウィンドウから システム設定 → 一般 を選択します。
2	言語リストから使用する言語を選択します。 既定の言語は英語です。
3	適用 をクリックして システム設定 ウィンドウを閉じます。
4	SoMachine Basic を再起動すると新しい言語でユーザーインターフェースが表示されます。



ヘルプのショートカットキーの変更

次の手順で、コンテキストヘルプまたは一般ヘルプにアクセスするためのショートカットキーを変更します。

手順	手順内容
1	システム設定ウィンドウから システム設定 → 一般 を選択します。
2	コンテキストヘルプは、 F1 または Shift + F1 を使用します。 一般的なヘルプ のショートカットキーは自動的に更新されます。

ラダーエディターのカスタマイズ

次の手順で、ラダーエディターをカスタマイズします。

手順	手順内容
1	システム設定ウィンドウから システム設定 → ラダーエディター を選択します。
2	ラダーエディターの グリッドラインのスタイル を選択します。 <ul style="list-style-type: none">● 点 (既定値)● 破線● 実線
3	ラダーエディターのセルの 列数 (11...30) を設定します。 セルの列数の既定値は 11 です。 詳細は、ラダーダイアグラムのプログラミング原則 (155 ページ参照) を参照してください。
4	ツール選択の保護 から以下を選択します。 <ul style="list-style-type: none">● 選択したツールを保持 (既定値): ラングにグラフィック要素を選択して配置すると、最後に選択されたグラフィック要素が選択された状態を保持します。これにより、同じ要素を再度選択することなくラングに配置することができます。ESC キーを押すか、ラングの空のセルを右クリックするとポインターツール  を選択できます。● ポインターにリセットする: ラングに A 接点、またはコイルを配置したあとに、ポインターツール  が自動的に選択されます。再度同じ A 接点、またはコイル要素を挿入するには、ツールバーから選択します。

手順	手順内容
5	ラダーエディターのショートカットとツールバーのスタイル設定を選択します。 <ul style="list-style-type: none">● SoMachine Basic セット (既定値)● Asian set 1● Asian set 2● European set● American set 選択したスタイルに対して、表示されている各ツールバーボタンのキーボードショートカットのリストを次の表に示します。
6	適用をクリックしシステム設定ウィンドウを閉じて、ラダーエディターの変更を表示します。

デフォルトのロジックコントローラーの選択

次の手順で、デフォルトのロジックコントローラーを選択します。

手順	手順内容
1	システム設定ウィンドウからシステム設定 → 設定を選択します。
2	優先コントローラーをクリックし、リストからデフォルトのロジックコントローラーを選択します。
3	適用をクリックしてシステム設定ウィンドウを閉じます。
4	SoMachine Basic を再起動し、新規プロジェクト作成すると設定タブに新しいデフォルトロジックコントローラーが表示されます。

レポートの印刷

情報




カスタマイズ可能なレポートを作成し、印刷、または PC に PDF 形式で保存することができます。

印刷ボタンには以下のオプションがあります。

- **プロジェクトレポートの印刷**では、ハードウェアコンポーネントの一覧、アプリケーション構成、およびプロジェクト、プログラム、アプリケーションの内容を含むカスタマイズされたレポートを印刷をします。
- **部品リストの印刷**では、プロジェクト設定で使用されているハードウェアコンポーネントの一覧を印刷します。
- **設定**では、プロジェクトレポートをカスタマイズし、レポートに含む要素とページレイアウトを選択できます。




プロジェクトレポートの印刷

プロジェクトレポートを印刷します。

手順	手順内容
1	ツールバーの印刷ボタン  の右にある下矢印をクリックし、 プロジェクトレポートの印刷 メニューコマンドを選択するか、CTRL+P を押します。 印刷プレビューウィンドウが表示されます。
2	<ul style="list-style-type: none">● 印刷プレビューウィンドウのツールバーの  をクリックして、プロジェクトレポートを印刷します。● 印刷プレビューウィンドウのツールバーの  をクリックして、プロジェクトレポートを PC に PDF ファイルとして保存します。


部品リストの印刷

部品リストを印刷します。

手順	手順内容
1	ツールバーの印刷ボタン  の右にある下矢印をクリックし、 部品リストの印刷 メニューコマンドを選択します。 印刷プレビューウィンドウが表示されます。
2	<ul style="list-style-type: none">● 印刷プレビューウィンドウのツールバー  をクリックして、部品リストを印刷します。● 印刷プレビューウィンドウのツールバーの  をクリックして、部品リストを PC に PDF ファイルとして保存します。

プロジェクトレポートのカスタマイズ

プロジェクトレポートに含む要素と、ページレイアウトを選択します。

手順	手順内容
1	<p>ツールバーの印刷ボタン  の右にある下矢印をクリックし、設定メニューコマンドを選択します。</p> <p>設定ウィンドウが表示されます。</p>
2	<p>レポートをクリックして、レポートの書式 (紙のサイズ、余白、および向き) を設定します。</p>
3	<p>プロジェクトレポートに含める要素を選択します。</p> <ul style="list-style-type: none"> ● 説明は、プロジェクト情報ウィンドウにあるようなプロジェクトの説明です。 ● 部品リストは、プロジェクト設定で使われているハードウェアコンポーネントの一覧です。 ● ハードウェア設定は、設定に使われているハードウェアデバイスの一覧です。 <ul style="list-style-type: none"> ○ IO Bus は、使われている I/O 拡張モジュールの一覧です。 ○ カートリッジは、使われているカートリッジの一覧です。 ● ソフトウェア設定では、以下を含める、または除外します。 <ul style="list-style-type: none"> ○ ワード型定数は、プロジェクトで使われているワード型定数 (%KW) オブジェクトの一覧です。 ○ ネットワークオブジェクトは、Ethernet/IP、または Modbus TCP デバイスの通信のために使われるオブジェクトの一覧です。 ○ ソフトウェアオブジェクトは、プログラムで使われる、タイマー、およびカウンターのようソフトウェアオブジェクトの一覧です。 ○ PTO オブジェクトは、プログラムで使われる PTO ファンクションブロックの一覧です。 ○ 通信オブジェクトは、プログラムで使われる通信オブジェクトの一覧です。 ● プログラムでは、以下を含める、または除外します。 <ul style="list-style-type: none"> ○ 動作は、動作ウィンドウで設定された設定です。 ○ メモリー消費量は、アプリケーション、プログラム、および関連するユーザーデータによって使われるコントローラーメモリー量です。 ○ アプリケーション構成は、マスタートスク、および周期タスクウィンドウで設定された設定です。 ○ POU は、プログラムで使われる POU の一覧です。 ● 表示は、リモートグラフィック表示 についての情報を含むレポートセクションです。 <ul style="list-style-type: none"> ○ 一般プロパティは、表示タブに表示される一般的なパラメーターです。レポートにオブジェクトでパスワードを印刷できます。 ○ アラームビューは、トリガーされたアラームの一覧が表示されます。 ○ ページは、表示タブに作成されたオペレーターインターフェースの一覧です。 ● シンボルは、すべてのシンボル、またはプロジェクトで使われるシンボルの一覧です。 ● クロスリファレンスは、使われているアドレス、オブジェクト、ラング、およびそれらが使われているコードを含むテーブルです。 ● アニメーションテーブルは、プロジェクトでアニメーションテーブルに追加されたオブジェクトを含むテーブルです。
4	<p>ウィンドウを閉じます。</p>

第 4 章

プロパティ

4.1 プロパティウィンドウの概要

このセクションについて

このセクションには次の項目が含まれています。

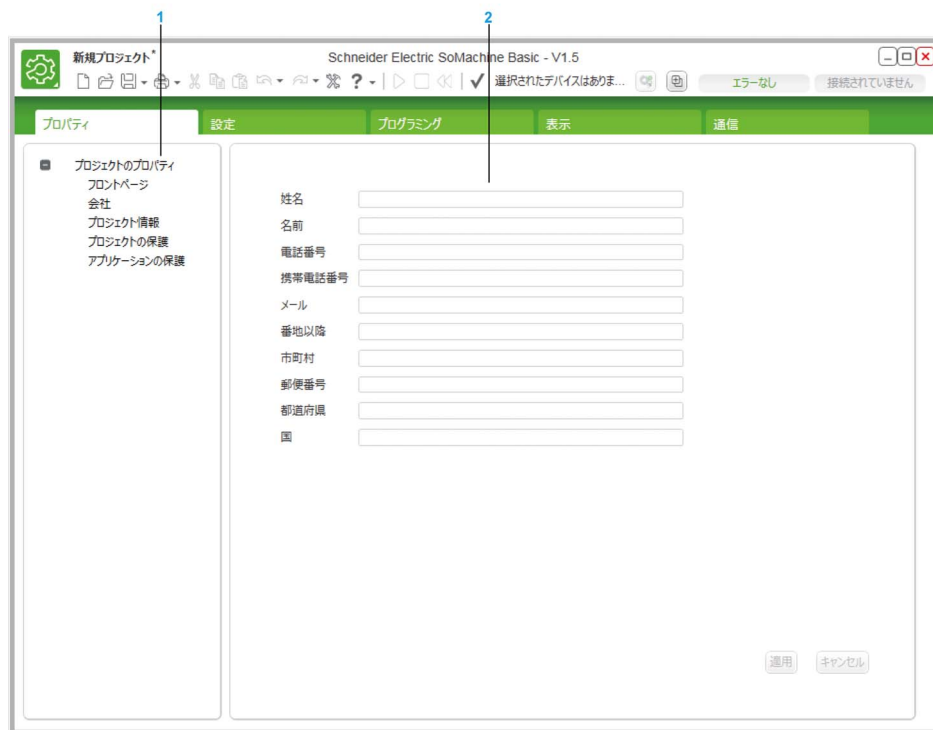
項目	参照ページ
プロパティウィンドウ	55
プロジェクトプロパティ	56

プロパティウィンドウ

概要

プロパティタブでプロジェクトに関する情報とパスワード保護を指定できます。

- プロジェクトの開発者、および開発企業の詳細
- プロジェクトの情報
- プロジェクトをパスワード保護する場合は、SoMachine Basic でプロジェクトを開くためのパスワードを正しく入力します。
- ロジックコントローラーにあるアプリケーションをパスワード保護する場合には、アプリケーションを SoMachine Basic プロジェクトにアップロードするためのパスワードを正しく入力します。



- 1 左側にプロパティの一覧が表示されます。
- 2 左側で選択した項目のプロパティが右側に表示されます。

プロジェクトプロパティ

概要

プロパティウィンドウを使用して、SoMachine Basic のユーザー、アプリケーションの開発をしている会社名、およびプロジェクトの詳細を提供します。ロジックコントローラーに格納されたプロジェクトファイルやアプリケーションをパスワードで保護することもできます。

アプリケーション開発者プロパティの設定

アプリケーション開発者プロパティを設定します。

手順	手順内容
1	プロパティタブを表示し、プロジェクトのプロパティ → フロントページをクリックします。
2	情報を入力します。
3	適用をクリックします。

注記： この情報は、SoMachine Basic のプロジェクトファイルを右クリックすると Windows Explorer のプロパティウィンドウに表示されます。

会社プロパティの設定

会社プロパティを設定します。

手順	手順内容
1	プロパティタブを表示し、プロジェクトのプロパティ → 会社をクリックします。
2	情報を入力します。 会社のロゴイメージをアップロードするには、変更をクリックしアップロードするファイルを選択します。削除をクリックすると設定していた画像イメージを取り消します。
3	適用をクリックします。

プロジェクト情報の設定

プロジェクト情報を設定します。

手順	手順内容
1	プロパティタブを表示し、プロジェクトのプロパティ → プロジェクト情報をクリックします。
2	情報を入力します。 写真や CAD 画像などのイメージをアップロードするには、変更をクリックしアップロードするファイルを選択します。削除をクリックすると設定していた画像イメージを取り消します。
3	適用をクリックします。

プロジェクトのパスワード保護

プロジェクトファイルを暗号化し、パスワード保護することができます。

プロジェクトが暗号化されている場合は、プロジェクトを開くたびに暗号化パスワードの入力を求められます。

プロジェクトが書き込み禁止の場合、初期設定ではプロジェクトの閲覧のみができます。プロジェクトを変更するには、変更用パスワードを入力します。

次の手順で、プロジェクトファイルの暗号化およびパスワード保護します。

手順	手順内容
1	プロパティタブを表示し、プロジェクトのプロパティ → プロジェクトの保護をクリックします。
2	有効を選択します。必須項目には、アスタリスク (*) が付いています。
3	プロジェクトを暗号化するためのパスワードを入力した後、確認のために再度入力します。
4	必要に応じて変更用パスワード入力し、プロジェクトを書き込み禁止にします。
5	適用をクリックします。

プログラムを読み取り専用モードにする場合は、コントローラーイメージを作成して、コントローラーに復元してください。(223 ページ参照)

プロジェクトのパスワード保護の解除

次の手順で、プロジェクトのパスワード保護を解除します。

手順	手順内容
1	プロパティタブを表示し、プロジェクトのプロパティ → プロジェクトの保護をクリックします。
2	無効を選択します。
3	適用をクリックします。 注記： 無効の設定が適用される前に現在の変更用パスワードの入力を求められた場合は、変更用パスワードを入力し適用をクリックします。

アプリケーションのパスワード保護

SoMachine Basic では、ロジックコントローラーに格納されたアプリケーションをパスワードで保護することができます。アプリケーションをロジックコントローラーから SoMachine Basic のプロジェクトへアップロードする際、パスワードでコントロールします。

次の手順で、アプリケーションをパスワード保護します。

手順	手順内容
1	プロパティタブを表示し、プロジェクトのプロパティ → アプリケーションの保護をクリックします。
2	アプリケーションの保護レベルを選択します。 <ul style="list-style-type: none">● 有効を選択し、パスワードを空白に残しロジックコントローラーから PC へのアプリケーションのアップロードを不可能にします。● 有効を選択し、同じパスワードをパスワードフィールドと確認フィールドに入力しアプリケーションの保護をします。ロジックコントローラーから PC へアプリケーションをアップロードする際に、このパスワードを入力します。
3	適用をクリックします。

アプリケーションのパスワード保護の解除

次の手順で、アプリケーションのパスワード保護を解除します。

手順	手順内容
1	プロパティタブを表示し、プロジェクトのプロパティ → アプリケーションの保護をクリックします。
2	無効を選択します。
3	適用をクリックします。 注記： 無効の設定が適用される前に現在のパスワードの入力を求められた場合は、パスワードを入力し適用をクリックします。

第 5 章

設定

5.1 構成ウィンドウの概要

このセクションについて

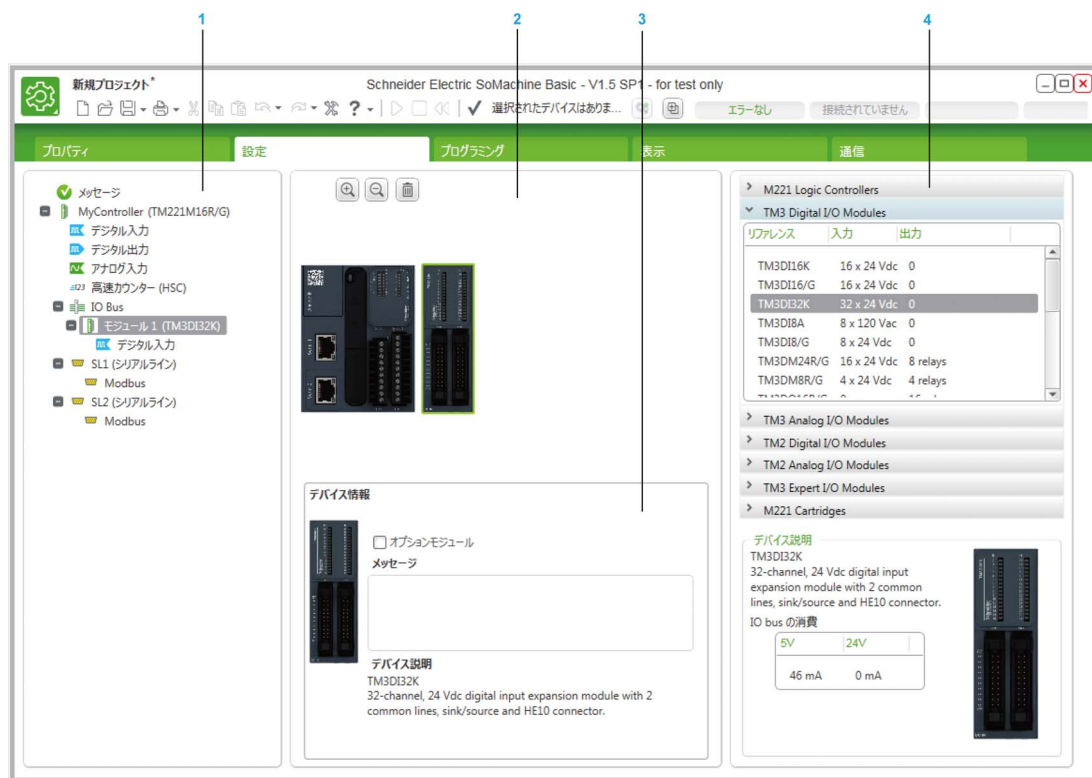
このセクションには次の項目が含まれています。

項目	参照ページ
設定ウィンドウの概要	61
設定の構築	62

設定ウィンドウの概要

はじめに

設定ウィンドウで、プログラムが対象とするロジックコントローラーと拡張モジュールのハードウェア設定をします。



- 1 ハードウェアツリー - ハードウェア設定の構造化表示
- 2 設定 - ロジックコントローラーと拡張モジュール
- 3 ロジックコントローラーと拡張モジュールのハードウェアコンポーネントのカタログリファレンス。コンポーネントをハードウェア設定に追加するには、ドラッグアンドドロップします。
- 4 設定で選択されているコンポーネント、またはハードウェアツリーで選択されているアイテムのプロパティ。

設定の構築

デフォルトのロジックコントローラーの置き換え

新規の SoMachine Basic プロジェクトを作成すると、設定ウィンドウの中央にロジックコントローラーのリファレンスが表示されます。

手順	手順内容
1	設定タブを開きます。
2	右側にあるカタログのロジックコントローラーカテゴリが表示されていない場合は展開します。
3	ロジックコントローラーリファレンスを選択します。ロジックコントローラーの物理的プロパティの簡単な説明が デバイス説明 に表示されます。
4	ロジックコントローラーリファレンスをウィンドウの中央にある既存のロジックコントローラーのイメージの上にドラッグ アンド ドロップします。
5	ロジックコントローラーリファレンスの置き換えを確認するメッセージが表示されたらはいをクリックします。

注記： デフォルトのコントローラーリファレンスは**システム設定**ウィンドウ (49 ページ参照) で指定されています。

ロジックコントローラーの設定

設定ウィンドウを使用してロジックコントローラーを設定します。

詳細は、設定で使用されているロジックコントローラーの**プログラミングガイド**を参照してください。

拡張モジュールの設定

設定ウィンドウを使用して拡張モジュールを設定します。

詳細は、設定で使用されている拡張モジュールの**プログラミングガイド**を参照してください。

第 6 章

プログラミング

この章について

この章には次のセクションが含まれています。

セクション	項目	参照ページ
6.1	プログラミングワークスペースの概要	64
6.2	特殊機能	65
6.3	プログラムの動作とタスクの設定	74
6.4	POU の管理	80
6.5	ユーザー定義ファンクション	92
6.6	ユーザー定義ファンクションブロック	99
6.7	マスタートask	105
6.8	文字列	108
6.9	周期タスク	114
6.10	イベントタスク	118
6.11	ツールを使う	125
6.12	ラダー言語のプログラミング	152
6.13	命令リスト (IL) 言語のプログラミング	171
6.14	グラフセ (リスト) プログラミング	180
6.15	グラフセ (SFC) プログラミング	187
6.16	オンラインモードでのデバッグ	199

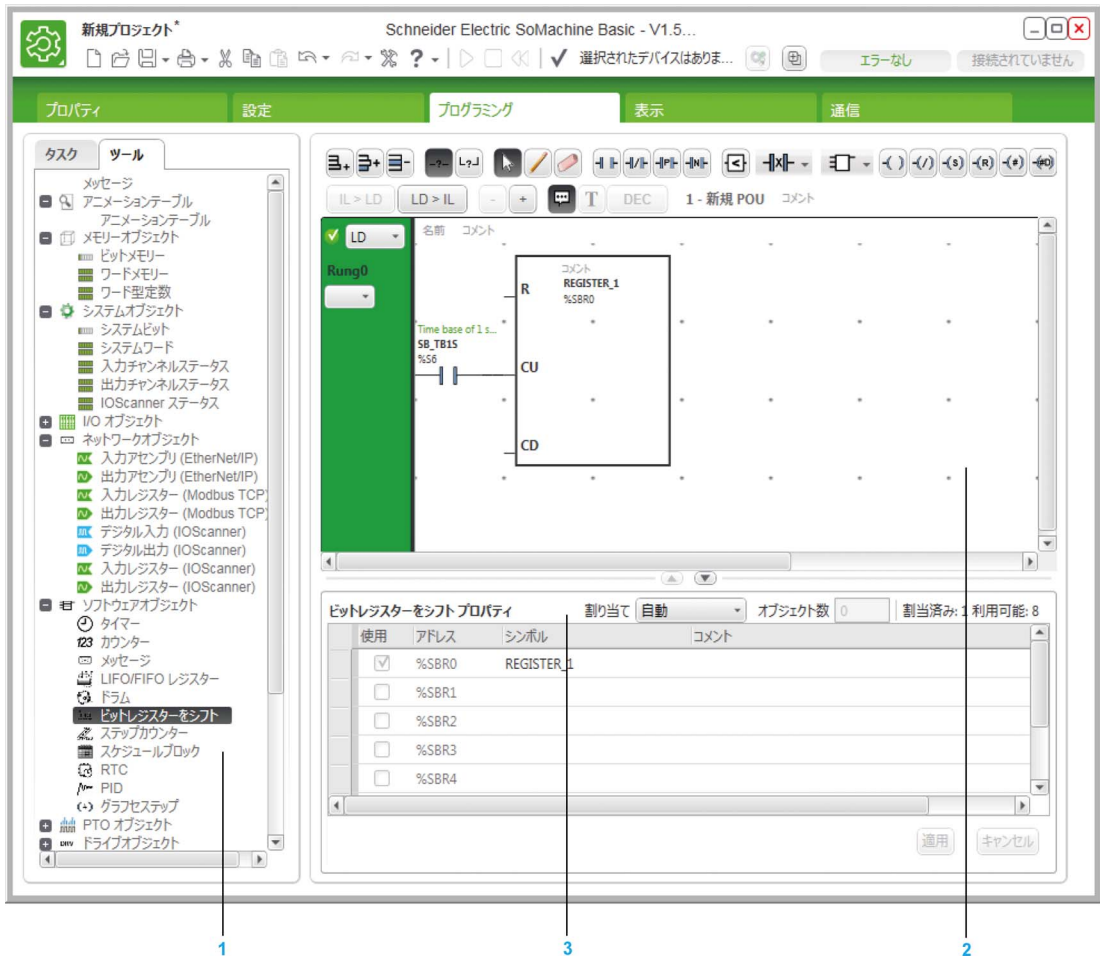
6.1

プログラミングワークスペースの概要

プログラミングワークスペースの概要

概要

プログラミングタブは3つの主要エリアに分かれています。



- 1 プログラミングツリーでは、プログラムやオブジェクトのプロパティ、関数、およびプログラムの監視とデバッグに使用できる多数のツールを選択できます。
- 2 中央上部はプログラムのソースコードを入力するプログラミングワークスペースです。
- 3 中央下部では、プログラミングワークスペース、またはプログラミングツリーで選択した項目のプロパティの表示、および設定ができます。

6.2 特殊機能

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
オブジェクト	66
シンボルアドレス指定	67
メモリーの割り当て	69
ラダー / リスト可逆性	70

オブジェクト

概要

SoMachine Basic では、オブジェクトは、アプリケーションで使用するために確保したロジックコントローラーのメモリーエリアです。オブジェクトは以下のような形態をとることができます。

- ビットメモリーやワードメモリーなどの、ソフトウェアの変数
- デジタルまたはアナログの入出力のアドレス
- システムワードやシステムビットなどの、コントローラーの内部変数
- タイマーやカウンタなどの、定義済みシステムファンクションやファンクションブロック

コントローラーメモリーは、事前に特定のオブジェクトタイプに割り当てられるか、ロジックコントローラーにアプリケーションをダウンロードするときに、自動的に割り当てられます。

メモリーが割り当てられると、プログラムでオブジェクトのアドレス指定ができます。オブジェクトのアドレス指定には、先頭に % を付けます。例えば、%MW12 はワードメモリーのアドレス、%Q0.3 は標準デジタル出力のアドレス、%TM0 は タイマーファンクションブロックのアドレスです。

シンボルアドレス指定

はじめに

SoMachine Basic は、言語オブジェクトのシンボルアドレス指定、すなわち、名前によるオブジェクトの間接的アドレス指定に対応しています。シンボルを使用することによって、迅速なプログラムロジックのテストと分析を行うことができ、アプリケーションの開発とテストを大幅に簡素化することができます。

例

例えば、WASH_END が洗浄サイクルの終わりを表す タイマーファンクションブロックのインスタンスを指定するシンボルとします。このような名前の目的を思い出すことは、%TM3 のようなプログラムアドレスの役割を覚えるより簡単です。



プロパティウィンドウでシンボルを定義

プロパティウィンドウでシンボルを定義

手順	手順内容																
1	プログラミングウィンドウの左側にある ツールタブ を選択します。																
2	シンボルを定義するオブジェクトタイプ、例えば I/O オブジェクト → デジタル入力 、を選択してデジタル入力のプロパティを表示します。 プログラミングウィンドウの中央下部に、そのオブジェクトタイプのプロパティウィンドウが表示されます。																
3	プロパティテーブルの シンボル 列をダブルクリックして、特定の項目に定義するシンボルを入力します。例：入力 %I0.2 に対して Input_1。  <table border="1"><caption>デジタル入力 プロパティ</caption><thead><tr><th>使用</th><th>アドレス</th><th>シンボル</th><th>コメント</th></tr></thead><tbody><tr><td><input type="checkbox"/></td><td>%I0.0</td><td></td><td></td></tr><tr><td><input checked="" type="checkbox"/></td><td>%I0.1</td><td>INPUT_1</td><td></td></tr><tr><td><input type="checkbox"/></td><td>%I0.2</td><td></td><td></td></tr></tbody></table>	使用	アドレス	シンボル	コメント	<input type="checkbox"/>	%I0.0			<input checked="" type="checkbox"/>	%I0.1	INPUT_1		<input type="checkbox"/>	%I0.2		
使用	アドレス	シンボル	コメント														
<input type="checkbox"/>	%I0.0																
<input checked="" type="checkbox"/>	%I0.1	INPUT_1															
<input type="checkbox"/>	%I0.2																
4	適用 をクリックします。																

ラダーエディター内でシンボルを定義

ラダーエディターでシンボルを定義します。

手順	手順内容																
1	ラダーエディターで、グラフィック要素の シンボルライン をクリックします。例：ラッチ、またはファンクションブロック。カーソルが表示されます。 																
2	Input_1 のように使用するシンボルを入力して、 Enter を押します。シンボルには以下のルールがあります。 <ul style="list-style-type: none"> ● 最大 32 文字 ● 英文字 (A-Z)、数字 (0-9)、またはアンダーバー (_) ● 最初の文字は必ず英文字パーセント記号 (%) は使えません。 ● シンボルは大文字、小文字を区別しません。例えば、Pump1 と PUMP1 は同じシンボルであり、オブジェクトに対して一意的にのみ使うことができます。すなわち、同じシンボルを異なるオブジェクトに割り当てることはできません。 																
3	グラフィック要素がまだオブジェクトに関連付けられていない場合、 注釈 ウィンドウが表示されます。シンボルと関連付けるオブジェクトを選択して、 OK をクリックします。もしくは、シンボルとオブジェクトを関連付ける指示が出たときに、 はい をクリックします。																
4	プロパティウィンドウの シンボル 列にシンボルを表示させるには、グラフィック要素のシンボル、またはオブジェクトをダブルクリックします。  <table border="1" data-bbox="322 1052 1145 1207"> <caption>デジタル入力 プロパティ</caption> <thead> <tr> <th>使用</th> <th>アドレス</th> <th>シンボル</th> <th>コメント</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>%I0.0</td> <td></td> <td></td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>%I0.1</td> <td>INPUT_1</td> <td></td> </tr> <tr> <td><input type="checkbox"/></td> <td>%I0.2</td> <td></td> <td></td> </tr> </tbody> </table>	使用	アドレス	シンボル	コメント	<input type="checkbox"/>	%I0.0			<input checked="" type="checkbox"/>	%I0.1	INPUT_1		<input type="checkbox"/>	%I0.2		
使用	アドレス	シンボル	コメント														
<input type="checkbox"/>	%I0.0																
<input checked="" type="checkbox"/>	%I0.1	INPUT_1															
<input type="checkbox"/>	%I0.2																

定義済みシンボルの表示

すべての定義済みシンボルの一覧を表示 (146 ページ参照) するには、**ツール** → **シンボル一覧** を選択します。

シンボルの保存

シンボルは非プログラムデータです。シンボルは、SoMachine Basic アプリケーションの一部としてロジックコントローラに保存されます。

メモリーの割り当て

はじめに

SoMachine Basic では、メモリーオブジェクト (ワードメモリー、ワード型定数)、およびソフトウェアオブジェクト (ファンクションブロック) を含む、プログラムで使用する特定のオブジェクトタイプをロジックコントローラーのメモリーブロックに、あらかじめ割り当て (予約) することができます。

割り当てモード

オフラインモードでは、各オブジェクトタイプにメモリー割り当てモードを指定することができます。オブジェクトを設定する際に (**プログラミング** → **ツール**)、オブジェクト一覧の上部に次のウィンドウが表示されます。

Allocation No. of Objects Allocated: 1, Available: 1024

使用するメモリー割り当てモードを選択します。

- **自動**: オフセット 0 から、プログラム、または関連付けられたシンボルで使われている最上位のメモリーアドレスまで、すべてのプロジェクトが自動的にロジックコントローラーメモリーに割り当てられます。例: プログラムでワードメモリー %MW20 が使用されている場合、%MW0 から %MW20 のすべてのオブジェクト (21 個のオブジェクト) が自動的にメモリーに割り当てられます。

その後オンラインモードに切り替えると、オンラインにする以前に使用していた最上位アドレスより上位のアドレスを持つ新規メモリーオブジェクトを割り当てることはできません。

- **手動**: **オブジェクト数**ボックスに、メモリーを割り当てるオブジェクトの番号を指定します。オンラインモードに切り替えると、ロジックコントローラーからのログアウト、プログラムの変更、ログイン、またはアプリケーションを再度ダウンロードすることなく、プログラムに接点、コイル、または方程式を新規追加することができます (割り当てられたメモリーの上限まで)。

SoMachine Basic に指定したオブジェクトの数が表示されます。

SoMachine Basic に、**割り当てられたメモリーオブジェクトの合計数**と、ロジックコントローラーで**利用可能なメモリーオブジェクトの数**が表示されます。

オブジェクトの数を指定した場合、そのオブジェクトのみが表に表示されます。

マルチオペランド命令を使用するには、%MW が 20 必要です。周期タスクを使用している場合、追加で %MW が 20 必要です。

ラダー / リスト可逆性

はじめに

SoMachine Basic ではラングをラダー図から IL (インストラクションリスト)、および IL (インストラクションリスト) からラダー図への変換に対応しています。プログラムの可逆性と呼ばれます。

SoMachine Basic では、ラングをプログラミング言語間でいつでも切り替えることができます。一部のラングをラダー図で表示し、その他のラングを IL (インストラクションリスト) で表示できます。

注記：ラダーと IL (インストラクションリスト) プログラムをグラフセ (SFC) に、グラフセ (SFC) プログラムをラダー、または IL (インストラクションリスト) に、またはグラフセ (IL) からグラフセ (SFC) に変換することはできません。

可逆性の理解

プログラムの可逆性を理解するポイントは、ラダー図のラングと対応する IL (インストラクションリスト) のラングの関係性を調べることです。

- **ラダー図ラング：**論理式を構成するラダー図命令。
- **リストシーケンス：**ラダー図命令に対応し、同じ論理式を表す IL (インストラクションリスト) プログラミング命令。

次の図は共通のラダー図ラング、およびそれに相当するプログラムの論理式を一連の IL (インストラクションリスト) 命令として表示しています。



対応する IL (インストラクションリスト) 命令

名前	コメント
0000	LD %I0.5
0001	OR %I0.4
0002	ST %Q0.4

ラダー図、または IL (インストラクションリスト) 言語として書かれたかどうかに関わらず、プログラムは内部的に IL (インストラクションリスト) 命令として格納されます。SoMachine Basic は、2つの言語のプログラム構造の類似性を利用して、プログラムの内部 IL (インストラクションリスト) イメージを使用して、IL (インストラクションリスト) 言語、またはラダー図として表示します。

可逆性に必要な命令

可逆性に対応した IL (インストラクションリスト) 言語内のファンクションブロックの構造には、次の命令が必要です。

- BLK はブロックの開始を表し、ラングの始まりとブロックへの入力の開始を定義します。
- OUT_BLK はブロックの出力の開始を表します。
- END_BLK はブロック、およびラングの終了を表します。

すべての IL (インストラクションリスト) プログラムに対して、可逆性に対応したファンクション命令を使用する必要はありません。

プログラミングの状況、および IL / ラダー可逆性

アドバイザーやエラーが発生するプログラムの状況を次の表に示します。修正されない場合、ラダーまたは IL (インストラクションリスト) の可逆性ができない可能性があります。

状況	IL	ラダー	ラングの可逆性
未定義のラベルにジャンプ	エラー	エラー	可
未定義のサブルーチンの呼び出し	エラー	エラー	可
未定義のグラフセステップの有効化 / 無効化	エラー	エラー	可
括弧に囲まれた命令にジャンプ	エラー	-	不可
括弧に囲まれたラベル	エラー	-	不可
括弧に囲まれたサブルーチン	エラー	-	不可
33 階層以上入れ子にした括弧	エラー	-	不可
括弧開きが無い括弧閉じ	エラー	-	不可
予約済み	-	-	-
括弧の不釣り合い	エラー	-	不可
END_BLK が無い BLK	エラー	-	不可
BLK が無い OUT_BLK、または END_BLK	エラー	-	不可
LD、または BLK が続かないラベル定義	エラー	-	不可
LD、または BLK が続かないサブルーチン定義	エラー	-	不可
予約済み	-	-	-
12 階層以上入れ子にした MPS	エラー	-	不可
MPS が無い MRD	エラー	-	不可
MPS が無い MPP	エラー	-	不可
POST でグラフセ命令を使用	エラー	エラー	可
LD、または BLK が続かないグラフセ定義	エラー	-	不可
スタック命令の不釣り合い	エラー	-	不可
予約済み	-	-	-
ラベルの重複	エラー	エラー	LD->IL のみ
サブルーチンの重複	エラー	エラー	LD->IL のみ
グラフセステップの重複	エラー	エラー	LD->IL のみ
予約済み	-	-	-
POST の重複	エラー	エラー	LD->IL のみ
入れ子になった FB	エラー	-	不可
BLK、および END_BLK の間にある OUT_BLK	エラー	-	不可
LD が続かない BLK	エラー	-	不可

状況	IL	ラダー	ラングの可逆性
OUT_BLK に FB 出力の LD が無い	エラー	-	不可
FB 構成外で使用される FB 出力	エラー	-	不可
繰り返されている、または使用不可の FB 出力	エラー	-	不可
FB 入力が OUT_BLK の前の BLK に無い	エラー	-	不可
対応する FB 構成外で使用される FB 入力	エラー	-	不可
繰り返されている、または使用不可の FB 入力	エラー	-	不可
BLK で宣言されたラベル	エラー	-	不可
BLK で宣言されたサブルーチン	エラー	-	不可
BLK で宣言されたグラフセステップ	エラー	-	不可
OUT_BLK の非 FB 出力の LD の試行	エラー	-	不可
BLK、および END_BLK の間で使用される FB 出力	エラー	-	不可
入れ子になったサブルーチン	エラー	エラー	不可
MPS、および MPP の間でサブルーチンの呼び出し	エラー	エラー	不可
括弧の間でサブルーチンの呼び出し	エラー	-	不可
予約済み	-	-	
プログラムの最初の命令が、ラング区切り記号では無い	エラー	-	不可
MPS、および MPP の間のジャンプ命令	エラー	エラー	不可
構文エラーを含むラング	エラー	-	不可
予約済み	-	-	-
予約済み	-	-	-
JMP、または END に続くプログラムの命令	エラー	-	不可
LD 命令で始まるラングが、条件付きアクション命令で終了していない	アドバイザー	-	不可
括弧に囲まれたアクション命令	エラー	-	不可
括弧に囲まれたスタック命令	エラー	-	不可
FB 用直接アクセス命令 (例: ""CU %C0"")	アドバイザー	-	不可
FB 入力セクション内にアクション命令	エラー	-	不可
END_BLK の後に命令	エラー	-	不可
AND、および OR 命令と共に使用される FB 出力	アドバイザー	-	不可
FB 出力内の OR 命令が括弧で囲まれていない	アドバイザー	-	不可
MRD、または MPP の前の命令が、条件付きアクション命令、またはスタック命令関連でない	アドバイザー	-	不可
MPS と MPP の間の OR が入れ子になっていない	アドバイザー	-	不可
アクション命令後の OR	アドバイザー	-	不可
MPS、MRD、または MPP の後の OR	アドバイザー	-	不可
予約済み	-	-	
JMPC、またはサブルーチンの呼び出しが、ラングの最後のアクション命令でない	アドバイザー	エラー	不可
iTwido の場合 7x11 セル、SoMachine Basic の場合 256 x 30 セルを超える標準ラング	アドバイザー	-	不可
BLK と END_BLK の間の無条件アクション命令	エラー	-	不可

状況	IL	ラダー	ラングの可逆性
OUT_BLK に有効な FB 出力の LD、または END_BLK が続いている	エラー	-	不可
最初のセルを FB は占有できません	-	-	可
ラングの先頭の FB が、セルを占有する項目を置換	-	-	可
FB の上部、または下部にロジックがない	-	エラー	不可
1 列目に XOR	-	エラー	不可
最後の列の A 接点、および水平コネクタ	-	エラー	不可
最後の列、または最後の行にダウンコネクタ	-	エラー	不可
有効サブルーチン 0 から 63 のみ許可	-	エラー	不可
有効ラベル 0 から 63 のみ許可	-	エラー	不可
演算ブロックに無効な演算式	-	エラー	不可
比較ブロックに無効な比較式	-	エラー	不可
接点、およびコイルに無効なアドレス、またはシンボル	-	エラー	不可
無効なオペランド、またはラダー命令の式	-	エラー	不可
出力アクションの無いラング	-	エラー	不可
左右のパワーバー間の途切れ	-	エラー	不可
ラダーラング未完結	-	エラー	不可
コネクタを使用し短絡した命令を含むラダーラング	-	エラー	不可
ブール論理項目のみを含む分岐は、逆順で収束する必要があります。	-	エラー	不可
関連する入力の無い FB	-	エラー	不可
FB 出力ピンは共に接続できません	-	エラー	不可
パワーバーに接続された XOR	-	エラー	不可
最後の出力アクション命令でないサブルーチン呼び出し、およびジャンプ	アドバイザー	エラー	不可
最後の行に FB の一部として FB を含んだ標準ラング	-	-	不可
iTwido の場合 7x11 セル、SoMachine Basic の場合 256 x 30 セルを超える標準ラング	アドバイザー	エラー	不可
サブネットワークの左ノードに接続された OPEN、および SHORT	-	エラー	不可
サブネットワークの左ノードに接続された XOR	-	エラー	不可
ラダーラングを表す LIST 文が 1 つも存在しない	-	エラー	不可

6.3

プログラムの動作とタスクの設定

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
アプリケーションの動作	75
タスクとスキャンモード	78

アプリケーションの動作

該当

次の状況で、アプリケーションがロジックコントローラーにどのように作用するかを設定できます。

- **起動** (75 ページ参照)
- **ウォッチドッグ** (76 ページ参照)
- **フォールバック動作** (76 ページ参照)
- **ファンクションレベル** (77 ページ参照)

アプリケーション動作の設定

次の手順で、アプリケーションの動作を設定します。

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	動作 を選択します。 結果: プログラミングウィンドウ の中央下部に 動作プロパティ が表示されます。
3	プロパティを必要に応じて変更します。
4	適用 をクリックして変更を保存します。

起動

ロジックコントローラー再起動後のプログラム動作を指定します。

- **以前の状態で開始**: ロジックコントローラーは停止する前の状態で開始します。
- **停止状態で開始**: ロジックコントローラーは自動的にアプリケーションを実行しません。
- **実行状態で開始** (初期値): ロジックコントローラーは、電池の有無および充電済み等のアプリケーション実行基準が満たされると自動的にアプリケーションの実行を開始します。
- **無条件に開始**: ロジックコントローラーは、電池がないまたは充電されていない場合でも自動的にアプリケーションの実行を開始します。

実行状態で開始機能を使用する場合、機器に電源が入ったときにコントローラーはプログラムロジックを実行します。出力の自動再起動が、制御しているプロセスや機器にどのように影響するかをあらかじめ把握してください。実行状態で開始機能を制御するように運転/停止入力を設定してください。また、運転/停止入力はリモートの実行コマンドよりもローカルの制御を優先するよう設計されています。ローカルで SoMachine によってコントローラーが停止された後に、リモートで実行コマンドが実行され予期しない結果が生じる可能性がある場合、そのような状態を制御するために運転/停止入力の設定および配線をしてください。

警告

装置の意図しない始動

- 実行状態で開始機能を使用する前に、出力の自動再起動により意図しない結果を生じないことを確認してください。
- 実行状態で開始機能を制御し、および遠隔操作による意図しない運転開始を防ぐために、運転/停止入力を使用してください。
- 運転/停止入力またはそのスイッチに触る前、または実行コマンドをリモート送信する前に、機器またはプロセスのセキュリティの状態を確認してください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

⚠ 警告

装置またはプロセスの意図しない始動

- 運転/停止入力に電源を入れる前に、機器またはプロセスのセキュリティの状態を確認してください。
- 実行状態で開始する機能を制御、および遠隔操作による意図しない運転開始を防ぐために、運転/停止入力を使用してください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

無条件に開始機能を使用する場合、以前に停止された理由がどうであれ機器に電源が入ったときにコントローラーはプログラムロジックを実行しようとします。電池残量が無いかまたは電池自体がない場合でもそのように動作します。そのため、メモリーの値が0、または初期値に戻った状態でコントローラーがスタートします。例えば短時間の停電の際などにコントローラーが再起動を試みると、停電発生時にメモリーに入っていた値が失われたり、メモリーの値を保持する電池がないために機器の再起動が意図しない結果を生じることがあります。無条件に開始機能が制御されるプロセスや機器にどのように影響するかを事前に把握することは不可欠です。無条件に開始機能を制御するように運転/停止入力を設定してください。

⚠ 警告

装置の意図しない動作

- コントローラーに無条件に開始を設定することによる影響を判断するために、すべての条件下でのリスク分析を行ってください。
- 必要のない無条件な再起動を避けるために、運転/停止入力を使用してください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

ウォッチドッグ

ウォッチドッグはプログラムで割り当てられたスキャンタイムを超過しないように使われる特別なタイマーです。

ウォッチドッグタイマーの初期値は 250 ms です。ウォッチドッグスキャンタスクの時間を指定します。有効範囲は、10...500 ms です。

フォールバック動作

何らかの理由で STOPPED または例外状態になったときに使用するフォールバック動作を指定します。

2 種類のフォールバック動作があります。

- **フォールバック値**を選択すると、標準ロジックコントローラーおよび拡張モジュール出力の設定プロパティで定義したフォールバック値に出力を設定します。これは初期値です。出力に対するフォールバック値の設定については、ロジックコントローラーまたは拡張モジュールの *プログラミングガイド* を参照してください。設定されたアラームステータス、PTO および FREQGEN 出力には個別のフォールバック値は定義できません。オブジェクトのフォールバック値は 0 で、変更できません。
- **値を保持**を選択すると、ロジックコントローラーが STOPPED または例外状態になったときの出力を保持します。このモードでは、ロジックコントローラーおよび拡張モジュール出力に設定されたフォールバック値は無視され、最後の出力とみなされる値が設定されます。**値を保持**の動作は、高速出力 (HSC 反射出力、PLS、PWM、PTO および FREQGEN) には適用されません。オブジェクトのフォールバック値は 0 です。

ファンクションレベル

システムによりファームウェアのバージョンが異なるロジックコントローラーが含まれていることがあるため、ファンクションレベルも異なります。SoMachine Basic はアプリケーションのファンクションレベルを制御する、ファンクションレベル管理に対応しています。

SoMachine Basic をロジックコントローラーに接続すると、以下のファンクションレベルを読み込みます。

- ロジックコントローラーのファームウェア。SoMachine Basic アプリケーションをロジックコントローラーにダウンロードする承認をします。アプリケーションで選択されたファンクションレベルは、ロジックコントローラーでサポートされている最大ファンクションレベル以下にしてください。そうでない場合は、ファームウェアを更新するか、アプリケーションファンクションレベルを手動でダウングレードするようにメッセージが表示されます (ファンクションレベルリストからレベルを選択します。下記参照)。
- ロジックコントローラーに組み込まれたアプリケーション。ロジックコントローラーアプリケーションを SoMachine Basic 動作中の PC へアップロードする許可をするかを決定します。アプリケーションのアップロードを承認するには、ロジックコントローラーのファンクションレベルがインストールされている SoMachine Basic のバージョンでサポートされている最大ファンクションレベル以下にしてください。そうでない場合は、アップロードする前に SoMachine Basic を最新バージョンにアップグレードしてください。

通信ウィンドウには、SoMachine Basic アプリケーションおよび接続されたロジックコントローラーに埋め込まれたアプリケーションのファンクションレベルを表示します。

ファンクションレベル一覧からレベルを選択します。

- **Level 6.0:** Modbus Serial I/O Scanner、ユーザー定義ファンクション、ユーザー定義ファンクションブロック、SD カード上のデータロギング、文字列管理、構造体ラダーブロック要素、立ち上がりおよび立ち下がりファンクション。
- **Level 5.1:** セキュリティ技術の変更が含まれています。
- **Level 5.0:** Modbus Serial I/O Scanner、ドライブおよび RTC ファンクションブロック、マルチオペランド命令を含みます。
- **Level 4.1:** オンラインモード拡張機能および SL2 のモデム対応を含みます。
- **Level 4.0:** シンクトランジスター出力コントローラー対応、グラフセ (SFC)、周波数発生器、保持タイマー、メモリー管理、進化したリモートグラフィック表示を含みます。
- **Level 3.3:** 拡張機能 (PTO モーションタスク、進化した HSC) を含みます。
- **Level 3.2:** オプションモジュール機能、EtherNet/IP adapter および %SEND_RECV_SMS ファンクションブロックに対応する拡張機能を含みます。
- **Level 3.1:** 拡張機能 (無条件に開始機能) を含みます。
- **Level 3.0:** ソフトウェアおよびハードウェアの以前のレベルに対応した拡張機能 (通信、モデム、リモートグラフィック表示) を含みます。
- **Level 2.0:** ソフトウェアおよびファームウェアの以前のレベルに対する修正および拡張機能を含みます。例えば、パルス列出力 (PTO) に対応するには、このファンクションレベル以上を選択してください。
- **Level 1.0:** 最初にリリースされた、SoMachine Basic ソフトウェアおよびそれに準拠したファームウェアバージョンの組み合わせ。

タスクとスキャンモード

概要

SoMachine Basic には、マスタータスクに対して以下のスキャンモードがあります。

- **連続モード**
連続スキャンモード (フリーホイールモード)。スキャンが完了した後すぐに新しいスキャンが開始されます。
- **周期モード**
周期的スキャンモード。設定されたスキャン時間を超過した後に新しいスキャンが開始されます。そのため、すべてのスキャン期間は同じになります。

SoMachine Basic には以下のタスクタイプがあります。

- **マスタータスク** : アプリケーションのメインタスク。
マスタータスクは、連続スキャン (連続スキャンモード)、または周期的スキャンモードのスキャン周期を 1...150 ms (初期値 100 ms) で指定することによって制御されます。
- **周期タスク** : 周期的に処理される短期サブルーチン。
周期的タスクは、スキャン周期 1...255 ms (初期値 255 ms) を指定することによって設定されます。
- **イベントタスク** : アプリケーションの応答時間を短縮するための、非常に短期のサブルーチン。
イベントタスクは、物理的入力、または HSC ファンクションブロックによってトリガーされます。これらのイベントは、標準デジタル入力 (%I0.2...%I0.5) (立上がり接点、立下り接点、および両方)、または高速カウンタ (HSC) (%HSC0、および %HSC1) (カウントが高速カウンタ (HSC) の閾値に達したとき) に関連付けられています。HSC ファンクションブロックごとに 2 つのイベントを設定できます。

タスクの優先度

タスクタイプと優先度を次の表に示します。

タスクタイプ	スキャンモード	トリガー条件	設定範囲	タスクの最大数	優先度
マスター	連続	連続	適用外	1	最低
	周期	ソフトウェアタイマー	1...150 ms ¹		
周期	周期	ソフトウェアタイマー	1...255 ms ¹	1	マスタータスクより高く、イベントタスクより低い
イベント	周期	物理的入力	%I0.2...%I0.5	4	最高
		%HSC ファンクションブロック	%HSC オブジェクトにつき 2 イベント		

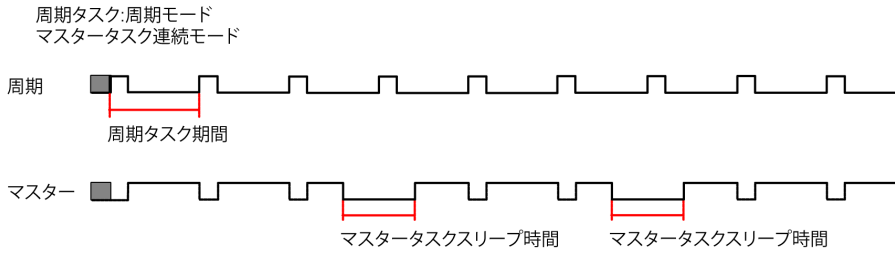
¹ 最小値 1 ms を設定するには、アプリケーションがレベル 5.0 以上のファンクションレベル (77 ページ参照) で設定されている必要があります。それ以外の場合、最小値は 2 ms です。

イベント優先度

イベント優先度とキュー (121 ページ参照) を参照してください。

連続スキャンモードのマスタータスク

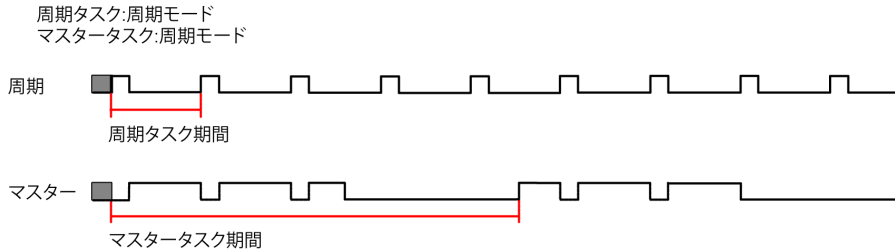
マスタータスクが連続キャンモードで設定されている場合の、マスタータスクと周期タスク実行の関係を下図に示します。



注記：マスタータスクのスリープ時間は、少なくとも合計サイクル時間の30%、最低1ミリ秒です。

周期スキャンモードのマスタータスク

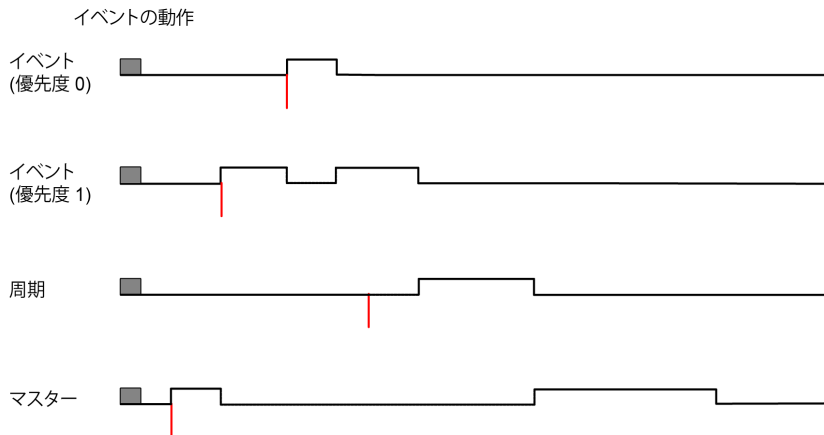
マスタータスクが周期キャンモードで設定されている場合の、マスタータスクと周期タスクの関係を下図に示します。



マスタータスク、および周期タスクのイベント優先度

イベント優先度は、イベントタスク、マスタータスク、および周期タスクの関係を制御します。イベントタスクは、マスタータスクと周期タスクの実行に割り込みます。

周期モードでのイベントタスク、マスタータスク、および周期タスクの関係を下図に示します。



イベントタスクにタスクイベントを送信するハードウェア割り込みによって、トリガーされます。

6.4 POU の管理

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
POU	81
タスクと関連した POU を管理する	82
ラングの管理	84
グラフセ (SFC) POU の管理	87
Free POU	89

POU

概要

プログラム構成単位 (POU) はプログラムで使われる再利用可能なオブジェクトです。各 POU は、対応しているプログラミング言語のソースコード内で、変数宣言、および一連の命令で構成されています。

POU は必ず 1 つあり、プログラムのマスタートスクにリンクされています。この POU は、プログラムが起動するたびに自動的に呼び出されます。


ファンクション、またはファンクションブロックなどの、他のオブジェクトを含む POU を追加で作成することも可能です。

POU を作成すると、初めは以下のいずれかになります。

- タスクに関連付けられる (82 ページ参照)
- Free POU (89 ページ参照)。Free POU は特定のタスク、またはイベントには関連付けされません。例えば、Free POU には、メインプログラムとは独立して管理されているライブラリーファンクションを含めることができます。Free POU は、サブルーチン、またはジャンプとしてプログラムで呼び出されます。周期タスク (115 ページ参照) は、Free POU として実装されるサブルーチンです。

タスクと関連した POU を管理する

タスクと関連した POU を追加する

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	以下のいずれかの方法で新規 POU を追加します。 <ul style="list-style-type: none"> ● マスタータスクを右クリックし、コンテキストメニューから POU の追加を選択します。 ● マスタータスクを選択し、タスクタブの上部にあるツールバーの  (POU の追加) をクリックします。 <p>結果：新しい POU がマスタータスクのプログラム構造、デフォルト / 最後の POU の直ぐ下に追加されます。デフォルト名は <i>n</i>-新規 POU です。<i>n</i> は POU が作成されるたびに増加する整数です。</p>
3	マスタータスク で POU の位置を変更するには、POU を選択して タスク タブの上部にある上、または下ボタンをクリックして、POU をプログラム構造内で上下に移動かします。

新規 POU を挿入する

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	POU を挿入する既存の POU を選択します。
3	選択した POU を右クリックし、コンテキストメニューから POU の挿入 を選択します。
4	マスタータスク で POU の位置を変更するには、POU を選択して タスク タブの上部にある上、または下ボタンをクリックして、POU をプログラム構造内で上下に移動かします。

タスクに関連した既存の POU をコピーペーストする

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	POU をひとつ、または複数選択します。 <ul style="list-style-type: none"> ● マスタータスクから既存の POU をひとつ選択します。 ● CTRL キーを押しながらマスタータスクの POU それぞれ選択します。
3	マスタータスク で選択した POU のひとつを右クリックし、コンテキストメニューから POU のコピー を選択します。
4	マスタータスク を右クリックし、コンテキストメニューから POU の貼り付け を選択します。 結果 ：ひとつ、または複数の POU がコピーした POU と同じ名前で、 マスタータスク のプログラム構造、選択された POU の直ぐ下に追加されます。

POU、または Free POU をエクスポートする

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	マスタータスク から POU、または Free POU をひとつ、または複数選択します。
3	マスタータスク で選択した POU、または Free POU を右クリックし、コンテキストメニューから POU のエクスポート を選択します。
4	POU ファイル (*.smbf) を エクスポートフォルダー に保存します。

POU、または Free POU をインポートする

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	マスタータスク から POU、または Free POU をひとつ、または複数選択します。
3	マスタータスク で選択した POU、または Free POU を右クリックし、コンテキストメニューから POU のインポート を選択します。
4	POU ファイル (*.smbf) をフォルダーから選択します。 注記 ：Free POU が最大数に達した場合、またはファイルが壊れている（無効なフォーマットの）場合、エラーメッセージが表示され Free POU はインポートされません。

POU の名前の変更



手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	次のいずれかの方法で POU 名を変更します。 <ul style="list-style-type: none">● 選択した POU を右クリックし、コンテキストメニューから POU の名前の変更を選択します。● POU をダブルクリックします。● POU を選択しプログラミングワークスペースで POU 名をダブルクリックします。● POU を選択し F2 キーを押します。
3	POU の新しい名前を入力し Enter キーを押します。

POUs を削除する


手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	POUs をひとつ、または複数選択します。 <ul style="list-style-type: none">● マスタータスクから既存の POU を選択します。● CTRL キーを押しながらマスタータスクの POU それぞれ選択します。
3	選択した POU を削除します。 <ul style="list-style-type: none">● マスタータスクで選択した POU を右クリックし、コンテキストメニューから POU の削除を選択します。● Delete キーを押します。

ラングの管理

ラングの作成

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	<p>以下の方法で、POU にラングを追加します。</p> <ul style="list-style-type: none"> POU を右クリックして、表示されたコンテキストメニューからラングの追加を選択します。 POU を選択し、タスクタブの上部にあるツールバーの  (ラングの追加ボタン) をクリックします。 POU を選択し、プログラミングワークスペースの上部にあるツールバーの  (新規ラングの作成ボタン) をクリックします。 <p>結果: 新規ラングは、プログラム構造の最後のラングのすぐ下に追加されます。</p>
3	POU のラングを再配置するにはラングを選択し、 タスクタブ の上部にあるツールバーの UP または DOWN ボタンをクリックして、プログラム構造内で選択したラングを上下に移動します。
4	ラングには Rung0 のようなシーケンス識別子が付けられます。ラングヘッダーをクリックすると、ラングを識別するためのラングコメントを追加することもできます。
5	初期設定のプログラミング言語は、 LD (Ladder) です。ラングに異なるプログラミング言語を選択するには、 LD をクリックしてプログラミング言語を選択します。
6	<p>ラングを JUMP 命令で呼び出す場合は、シーケンス識別子 Rungx (x は、POU のラング番号) の下のドロップダウンボタンをクリックしてラベルをラングに割り当て、一覧から %L を選択します。</p> <p>結果: ラングに、%Ly (y はラベル番号) というラベルが付けられます。ボタン上に %L と表示され、ラベル番号 y はボタンの接尾辞として表示されます。</p> <p>注記: ラベル番号は、次のラベルを定義するときに 1 ずつ増加します。</p> <p>ラベル番号を変更するには、ラングのラベル番号をダブルクリックして新規番号を入力し、Enter キーを押します。</p>

既存ラングの上にラングを挿入

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	プログラミングワークスペースの既存ラングを選択します。
3	<p>プログラミングワークスペースの上部にあるツールバーの  (新規ラングの挿入ボタン) をクリックします。</p> <p>結果: 選択したラングの上に新規ラングが表示されます。</p>
4	ラングには Rung0 のようなシーケンス識別子が付けられます。ラングヘッダーをクリックすると、ラングを識別するためのラングコメントを追加することもできます。
5	初期設定のプログラミング言語は、 LD (Ladder) です。ラングに異なるプログラミング言語を選択するには、 LD をクリックして、プログラミング言語を選択します。

手順	手順内容
6	<p>ラングを JUMP 命令で呼び出す場合は、シーケンス識別子 Rungx (x は、POU のラング番号) の下のドロップダウンボタンをクリックしてラベルをラングに割り当て、一覧から %L を選択します。</p> <p>結果: ラングに、%Ly (y はラベル番号) というラベルが付けられます。ボタン上に %L と表示され、ラベル番号 y はボタンの接尾辞として表示されます。</p> <p>注記: ラベル番号は、次のラベルを定義するときに 1 ずつ増加します。</p> <p>ラベル番号を変更するには、ラングのラベル番号をダブルクリックして新規番号を入力し、Enter キーを押します。</p>

ラングのコピー

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	<p>1 つまたは複数のラングを選択します。</p> <ul style="list-style-type: none"> ● 既存ラングを選択します。 ● CTRL キーを押したまま、各ラングを選択。
3	<p>選択したラングの 1 つを右クリックしてコピーし、次のいずれかの方法を実行します。</p> <ul style="list-style-type: none"> ● 表示されるコンテキストメニューから、選択したラングをコピーを選択します。 ● CTRL + C キーを押します。
4	<p>ラングを右クリックし、次のいずれかの方法を実行します。</p> <ul style="list-style-type: none"> ● 表示されるコンテキストメニューから、ラングを貼り付けを選択します。 ● CTRL + V キーを押します。 <p>結果: ラングのコピーは、元のラングと同じラベルで挿入されます。必要に応じてラベルを編集します。</p>



注記: プログラミングウィンドウでもラングのコピーと貼り付けができます。

手順	手順内容
1	コピーをするには、ラングを右クリックして、 選択したラングのコピー を選択します。
2	プログラミングワークスペースで右クリックして、 ラングの貼り付け を選択します。

ラングの名前の変更


手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	<p>以下の方法でラング名を変更します。</p> <ul style="list-style-type: none"> ● ラングを右クリックして、表示されたコンテキストメニューからラングの名前の変更を選択します。 ● ラングをダブルクリックします。 ● プログラミングワークスペースでラングを選択し、ラング名または名前の文字をダブルクリックします。 ● ラングを選択して、F2 キーを押します。
3	新しい名前を入力して、Enter キーを押します。

ラングの削除

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	<p>以下の方法でラングを削除します。</p> <ul style="list-style-type: none"> ● ラングを右クリックして、表示されたコンテキストメニューからラングの削除を選択します。 <p>● ラングを選択し、タスクタブの上部にあるツールバーの  (ラングの削除ボタン) をクリックします。</p> <p>● ラングを選択し、プログラミングワークスペースのタスクタブの上部にあるツールバーの  (ラングの削除ボタン) をクリックします。</p> <ul style="list-style-type: none"> ● プログラミングワークスペースのラングを右クリックして、表示されたコンテキストメニューから選択したラングの削除を選択します。 ● ラングを選択して、削除キーを押します。
3	ラングが空でない場合、ラングの削除を確認する指示がでます。

グラフセ (SFC) POU の管理

グラフセ POU の作成

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	次のいずれかの方法で新規グラフセ POU を追加します。 <ul style="list-style-type: none">● マスタートaskを右クリックして、グラフセ POU の追加をコンテキストメニューから選択します。● タスクタブの上部にあるツールバーの  (グラフセ POU の追加)のボタンをクリックします。 結果 : n - グラフセ ノードは マスタートask ノードの下に表示されます。 n はグラフセ POU が作成されると増加する整数です。

新規グラフセ POU の挿入

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	新規グラフセ POU をすぐ上に挿入するのに既存のグラフセ POU を選択します。
3	選択した POU を右クリックし、 グラフセ POU の挿入 をコンテキストメニューから選択します。
4	グラフセ POU を マスタートask の中で位置を変更するには、グラフセ POU を選択して タスクタブ の上部のツールバーにある、上に、または下にのボタンをクリックし、プログラム構成内で選択したグラフセ POU を上、または下に移動します。

グラフセ POU のコピーと貼り付け

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	1つ、または複数のグラフセ POU を選択します。 <ul style="list-style-type: none">● 既存のグラフセ POU をマスタートask内で選択します。● CTRL キーを押したまま、各グラフセ POU をマスタートask内で選択します。
3	マスタートask 内で、選択したグラフセ POU を右クリックし POU の削除 をコンテキストメニューから選択します。
4	マスタートask を右クリックし、 POU の貼り付け をコンテキストメニューから選択します。 結果 : 1つ、または複数のグラフセ POU は、 マスタートask 内で選択したグラフセ POU の下に、コピーしたグラフセ POU と同じ名前前でプログラムの構成に追加されます。

グラフセ POU 名の変更

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	次のいずれかの方法でグラフセ POU 名を変更します。 <ul style="list-style-type: none"> ● 選択した POU を右クリックし、POU の名前の変更をコンテキストメニューから選択します。 ● グラフセ POU をダブルクリックします。 ● グラフセ POU を選択し F2 キーをおします。
3	新規グラフセ POU ノードの名前を入力して ENTER を押します。

グラフセ POU の削除

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	1つ、または複数のグラフセ POU を選択します。 <ul style="list-style-type: none"> ● 既存のグラフセ POU をマスタートスク内で選択します。 ● CTRL キーを押したまま、各グラフセ POU をマスタートスク内で選択します。
3	選択されたグラフセ POU を削除します。 <ul style="list-style-type: none"> ● マスタートスク内で、右クリックでグラフセ POU を選択し POU の削除をコンテキストメニューから選択します。 ● DELETE キーを押します。

Free POU

はじめに

SoMachine Basic では、Free POU は明示的にはタスクに関連付けされていない特別なタイプの POU です。

- [-] Free POUs
 - [-] Free POU_0 (SR2)
 - Rung0
 - Rung1
 - [-] Free POU_1 (SR3)
 - Rung0
 - [-] Free POU_2 (SR4)
 - Rung0
 - Rung1
 - Rung2

各 Free POU はサブルーチンとして実装され、ラダー、または IL プログラミング言語で書かれたラングで構成されています。

注記： グラフセ POU を Free POU にすることはできません。

Free POU は以下のときに使われます。

- プログラムラングから、サブルーチンコール (SRi) を使って呼び出されたとき。
- 周期タスクとして設定されたとき。
- イベントタスクとして設定されたとき。例えば、高速カウンター (HSC) ファンクションブロックの閾値 0 (%HSCi.TH0) のサブルーチン。

周期タスク、またはイベントタスクとして使われると、Free POU サブルーチンは**タスク**ウィンドウの **Free POU** から、**周期タスク**、または**イベント**にそれぞれ自動的に移動します。

周期タスク、またはイベントタスクとして使われなくなったとき、サブルーチンは **Free POU** に戻り、他のタスク、またはイベントにより使用することが可能になります。

新規 Free POU の作成

新規 Free POU を作成するには、以下の手順で行います。

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	Free POU を右クリックして、表示されたコンテキストメニューから Free POU の追加 を選択します。 結果： デフォルト名 "Free POU_0" の新規 POU、およびデフォルトサブルーチン番号 "SR0" が Free POU ブランチの下に表示され、新規ラングが プログラミング ワークスペースに表示されます。
3	必要に応じて、新規 POU を右クリックし、 POU の名前の変更 を選択した後、新しい名前を入力して Enter キーを押します。 Free POU の名前は、 プログラミング ワークスペースに表示されているラングも更新されます。
4	任意で、Free POU に関連付けるコメント (167 ページ参照) を入力します。
5	コメントボックスの右側にある サブルーチン番号 を選択して、一覧からサブルーチン番号を選択してください。 結果： Free POU 一覧にある POU の説明が、選択したサブルーチン番号で更新されます。例："SR11"
6	ラダー、または IL プログラミング言語で、Free POU/Free グラフセ POU のラング / ステップ、またはソースコードを作成します。

既存 Free POU のコピーと貼り付け

Free POU を作成するためにタスクに関連付けされた既存 POU のコピーと貼り付けをするには、以下の手順で行います。

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	1つ、または複数の Free POU を選択します。 <ul style="list-style-type: none"> ● 既存 Free POU を選択。 ● CTRL キーを押したまま、各 Free POU を選択。
3	選択した Free POU の1つを右クリックして、表示されたコンテキストメニューから POU のコピー を選択します。
4	右クリックして、表示されたコンテキストメニューから POU の貼り付け を選択します。 結果 : Free POU_x という名前 (x は次に使用できる Free POU の番号) の新規 Free POU、およびデフォルトのサブルーチン番号 SRx (x は次に使用できるサブルーチンの番号) が Free POU の下に表示されます。POU のすべてのラングが、新規 Free POU サブルーチン番号に自動的に関連付けられます。

タスクに関連付けられた既存 POU のコピーと貼り付け

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	1つ、または複数の POU を選択します。 <ul style="list-style-type: none"> ● マスタータスクの既存 POU を選択。 ● CTRL キーを押したまま、マスタータスクの各 POU を選択。
3	マスタータスク で選択した POU の1つを右クリックして、表示されたコンテキストメニューから POU のコピー を選択します。
4	マスタータスク を右クリックして、表示されたコンテキストメニューから POU の貼り付け を選択します。 結果 : コピーした POU と同じ名前の POU が、 マスタータスク のプログラム構造で選択した POU のすぐ下に追加されます。

Free POU のエクスポート

手順	手順内容
1	プログラミングウィンドウの左上にある タスク タブを選択します。
2	マスタータスク にある1つ、または複数の既存 Free POU を選択します。
3	マスタータスク の選択した Free POU の1つを右クリックして、表示されたコンテキストメニューから POU のエクスポート を選択します。
4	表示されたエクスポートフォルダーに、エクスポートした Free POU ファイル (*.smbf) を保存します。

Free POU のインポート

手順	手順内容
1	プログラミングウィンドウの左上部にある タスク タブを選択します。
2	マスタータスクにある 1 つ、または複数の既存 Free POU を選択します。
3	マスタータスクの選択した Free POU の 1 つを右クリックして、表示されたコンテキストメニューから POU のインポート を選択します。
4	表示されたフォルダーから Free POU ファイル (*.smbf) を選択し、 開く をクリックします。 注記 : Free POU の数が最大に達したり、ファイルが壊れている (無効なフォーマット) の場合、エラーメッセージが表示され Free POU はインポートされません。

Free POU の削除

Free POU を削除するには、以下の手順で行います。

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	1 つ、または複数の Free POU を選択します。 <ul style="list-style-type: none">● 既存 Free POU を選択。● CTRL キーを押したまま、各 Free POU を選択。
3	選択した Free POU を削除します。 <ul style="list-style-type: none">● 選択した Free POU を右クリックして、表示されたコンテキストメニューから POU の削除 を選択します。● DELETE キーを押します。

注記 : 削除する前に、タスクから Free POU の割り当て解除 (116 ページ参照) します。

Free POU をイベント、または周期タスクに割り当て

初期設定では Free POU、およびサブルーチンは、イベント、またはタスクに関連付けされていません。

Free POU の周期タスクへの関連付けについては、周期タスクの作成 (115 ページ参照) を参照してください。

Free POU のイベントへの関連付けについては、イベントタスクの作成 (122 ページ参照) を参照してください。

6.5

ユーザー定義ファンクション

概要

ユーザー定義ファンクションにより、入力パラメーター、ローカル変数および戻り値を使用した新しいファンクションを作成できます。ユーザー定義ファンクションは、SoMachine Basic プロジェクトの一部として保存されます。

ユーザー定義ファンクションは、次の場所で呼び出すことができます。

- マスタータスク
- 周期タスク
- イベント
- Free POU

注記：ユーザー定義ファンクションに対応させるには、アプリケーションの ファンクションレベル (77 ページ参照) を **6.0** 以上に設定してください。

このセクションについて


このセクションには次の項目が含まれています。

項目	参照ページ
ユーザー定義ファンクションの作成	93
ユーザー定義ファンクションの定義	94
ユーザー定義ファンクションの管理	97

ユーザー定義ファクションの作成

新しいユーザー定義ファクションの追加

プロジェクトには最大 64 個のユーザー定義ファクションを含めることができます。

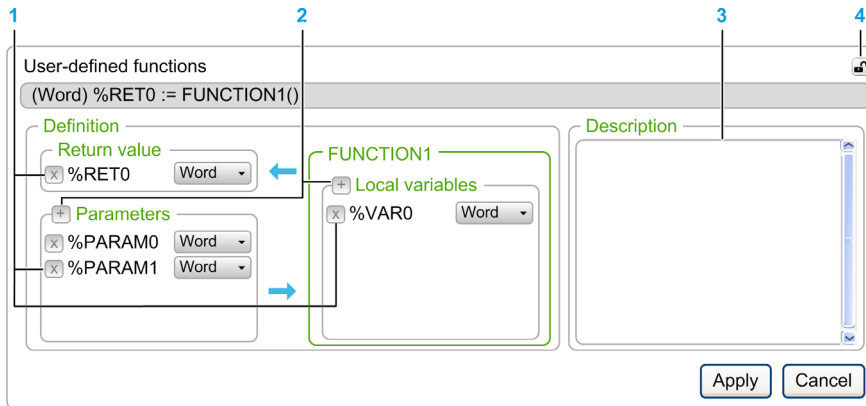
手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	次のいずれかの方法を使用して、ユーザー定義ファクションを作成します。 <ul style="list-style-type: none">● ユーザー定義ファクションを右クリックし、コンテキストメニューからユーザー定義ファクションの追加を選択します。● ユーザー定義ファクションを選択し、タスクタブの上部にあるツールバーの  (ユーザー定義ファクションの追加) をクリックします。● 既存のユーザー定義ファクションを選択し、右クリック、ユーザー定義ファクションを挿入を選択します。 <p>結果: 新しいユーザー定義ファクションがリストの一番下のプログラム構造体に追加されます。ユーザー定義ファクションを挿入した場合、新しいユーザー定義ファクションは選択したファクションの上に挿入されます。名前の初期値は、FUNCTIONn です。ここで、n はユーザー定義ファクションが作成されるたびにインクリメントされる整数です。</p>
3	必要に応じて、ユーザー定義ファクション名を変更します。ユーザー定義ファクション名の変更 (98 ページ参照) を参照してください。
4	ユーザー定義ファクションを定義します。ユーザー定義ファクションの定義 (94 ページ参照) を参照してください。

通常の POU のラングと同じ方法で、ユーザー定義ファクションでラングの作成および管理ができます。ラングの管理 (84 ページ参照) を参照してください。

ユーザー定義関数の定義

プレゼンテーション

ユーザー定義関数のプロパティ表示で使用可能な操作を下図に示します。



- 1 戻り値、入力パラメーター、ローカル変数を削除します。
- 2 戻り値、入力パラメーター、ローカル変数を追加します。
- 3 必要に応じて、ユーザー定義関数の目的についての説明を記述します。演算ブロックのユーザー定義関数を使用するとき、この説明がツールチップに表示されます。
- 4 プロパティビューを分離させます。

ユーザー定義関数のプログラム

ユーザー定義関数をプログラムします。

手順	手順内容																					
1	新しいユーザー定義関数を追加します。ユーザー定義関数の追加 (93 ページ参照) を参照してください。																					
2	戻り値、入力パラメーター およびローカル変数を定義して、ユーザー定義関数のインターフェースを定義します。ユーザー定義関数の定義 (95 ページ参照) を参照してください。																					
3	適用をクリックします。																					
4	<p>1 つ以上のラダー /IL ラング (84 ページ参照) のユーザー定義関数の機能を定義します。</p> <ol style="list-style-type: none"> 1. ラダー構造体の要素を挿入します。 2. ユーザー定義関数をプログラムします。 <p>例：</p> <div style="border: 1px solid black; padding: 5px;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 20px;">LD</th> <th style="width: 100px;">name</th> <th style="width: 100px;">Comment</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">✓</td> <td></td> <td>%VAR0 := %PARAM0 + %PARAM1</td> </tr> <tr> <td style="text-align: center;">Rung0</td> <td></td> <td>%VAR0 := %PARAM0 + %PARAM1</td> </tr> <tr> <td></td> <td></td> <td>...</td> </tr> <tr> <td></td> <td></td> <td>%RET0 := %VAR0 / 2</td> </tr> <tr> <td></td> <td></td> <td>%RET0 := %VAR0 / 2</td> </tr> <tr> <td></td> <td></td> <td>...</td> </tr> </tbody> </table> </div>	LD	name	Comment	✓		%VAR0 := %PARAM0 + %PARAM1	Rung0		%VAR0 := %PARAM0 + %PARAM1			...			%RET0 := %VAR0 / 2			%RET0 := %VAR0 / 2			...
LD	name	Comment																				
✓		%VAR0 := %PARAM0 + %PARAM1																				
Rung0		%VAR0 := %PARAM0 + %PARAM1																				
		...																				
		%RET0 := %VAR0 / 2																				
		%RET0 := %VAR0 / 2																				
		...																				

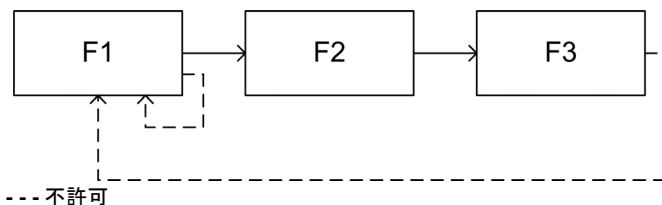
また、IL エディターでユーザー定義ファクションを直接プログラムすることもできます。

IL	name	Comment
Rung0	0000	LD 1
	0001	[%VAR0 := %PARAM0 + %PARAM1]
	0002	[%RET0 := %VAR0 / 2]

ユーザー定義ファンクションを実装するラング内の他のユーザー定義ファンクションを呼び出すことができます。

注記：ユーザー定義ファンクションは、再帰的に使用することはできません。ユーザー定義ファンクションは、それ自体を直接的または間接的に呼び出すことはできません。

例：



ユーザー定義ファクションは、サブルーチンと呼び出すことはできませんが、サブルーチンは、ユーザー定義ファクションと呼び出すことができます。

変数とグローバル変数

次の 3 つの変数は、ユーザー定義ファクションを実装するラングでのみ使用できます。

- %RET0
- %PARAMn
- %VARn

グローバル変数は、ユーザー定義関数のラングを含む SoMachine Basic プログラムで使用できる変数です。

ユーザー定義ファクションのインタフェースの定義

ユーザー定義関数を使用するには、オブジェクトとそのデータ型を定義してください。

オブジェクト	データ型	詳細
戻り値 %RET0	ワード ダブル フロート	ユーザー定義ファクションによって返される値。 ユーザー定義ファクションのラングでのみ使用できます。
パラメーター %PARAMn ⁽¹⁾		ユーザー定義ファクションのパラメーター。 ユーザー定義ファクションのラングでのみ使用できます。 パラメーターアドレスの初期値は変更できません。 パラメーターをアニメーションテーブルに追加できません。 オンラインモードでは、パラメーターの現在値は IL / ラダーエディターに表示されません。
ローカル変数 %VARn ⁽¹⁾		ユーザー定義ファクション内にデータ値を格納するために使用される変数。 ユーザー定義ファクションのラングでのみ使用できます。 ローカル変数アドレスの初期値は変更できません。 ローカル変数をアニメーションテーブルに追加できません。 オンラインモードでは、ローカル変数の現在値は IL / ラダーエディターに表示されません。



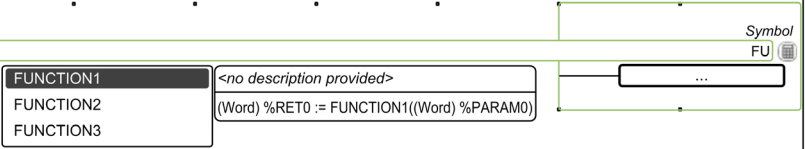

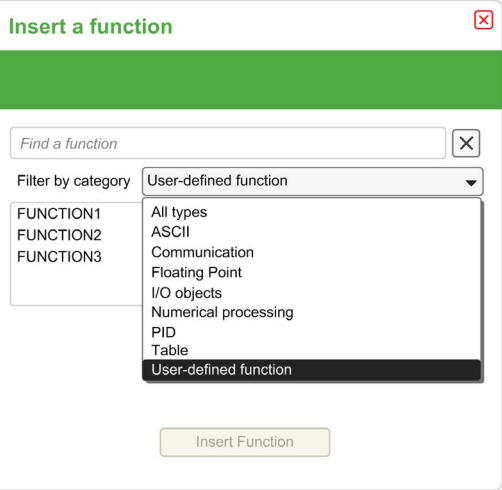
⁽¹⁾ n は、パラメーターまたはローカル変数が作成されるたびにインクリメントされる整数です。

このオブジェクトはオプションです。

ユーザー定義ファンクションの使用

ユーザー定義ファンクションを定義した後は他のファンクションと同様に**演算ブロック**を使用して、プログラム内の任意の場所で使用することができます。

ラダーエディターで、以下の手順に沿って進めます。

手順	手順内容
1	ツールバー上の 演算ブロック  ボタンをクリックします。
2	ラングのアクションゾーン (155 ページ参照) をクリックして、 演算ブロック を挿入します。
3	ツールバー上の 選択モード  ボタンをクリックします。
4	<p>演算式ラインを、ダブルクリックします。 次のいずれかの操作を行います。</p> <ul style="list-style-type: none"> ● ユーザー定義ファンクションの名前を入力します。例えば、名前が「FUNCTION1」の場合は「FU」と入力すると、「FU」で始まるすべてのユーザー定義ファンクションの名前が表示されます。  <ul style="list-style-type: none"> ● スマートコーディングを (165 ページ参照) 使用します。 <ul style="list-style-type: none"> a. スマートコーディング  ボタンをクリックします。 b. カテゴリ別フィルター、ユーザー定義ファンクションを順に選択します。 c. ユーザー定義ファンクションを選択します。 
5	ファンクションの挿入 をクリックします。
6	ユーザー定義ファンクションのインターフェースの定義 (95 ページ参照) で定義されている戻り値とパラメーターを入力して、ユーザー定義ファンクションの定義を完了します。

ユーザー定義ファンクションの管理

オフラインおよびオンラインモードでのユーザー定義ファンクション

ユーザー定義ファンクションはオフラインモードで管理することができます。

オンラインモードでは、既存のユーザー定義ファンクションにラングを追加できます。

既存のユーザー定義ファンクションのコピー、切り取り、貼り付け

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	1つ以上のユーザー定義ファンクションを選択します。 <ul style="list-style-type: none">● ユーザー定義ファンクションをクリックして選択します。● CTRL キーを押しながら複数のユーザー定義ファンクションを選択します。
3	右クリックで表示されるコンテキストメニューから ユーザー定義ファンクションのコピー または ユーザー定義ファンクションの切り取り を選択します。
4	ユーザー定義ファンクション を右クリックし、コンテキストメニューから ユーザー定義ファンクションの貼り付け を選択します。 結果 :1つ以上のユーザー定義ファンクションが、 ユーザー定義ファンクションプログラム構造体 の最後に追加されます。SoMachine Basic は自動的に、コピーされたユーザー定義ファンクションに新しい名前を割り当てます。

ユーザー定義ファンクションのエクスポート

ユーザー定義ファンクションは、プロジェクトの一部として保存されます。別のプロジェクトでユーザー定義ファンクションを使用する場合は、そのファンクションをエクスポートしてから、別のプロジェクトにインポートしてください。

SoMachine Basic インスタンス間でコピー、ペーストすることができます。

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	1つ以上のユーザー定義ファンクションを選択します。 <ul style="list-style-type: none">● ユーザー定義ファンクションをクリックして選択します。● CTRL キーを押しながら複数のユーザー定義ファンクションを選択します。
3	ユーザー定義ファンクション のユーザー定義ファンクションを右クリックし、表示されるコンテキストメニューから ユーザー定義ファンクションのエクスポート を選択します。
4	ユーザー定義ファンクションファイル (*.smbf) を エクスポートフォルダー に保存します。

ユーザー定義ファンクションのインポート


ユーザー定義ファンクションは、プロジェクトの一部として保存されます。別のプロジェクトでユーザー定義ファンクションを使用する場合は、そのファンクションをエクスポートしてから、別のプロジェクトにインポートしてください。

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	ユーザー定義ファンクション のユーザー定義ファンクションを選択します。
3	ユーザー定義ファンクション のユーザー定義ファンクションを右クリックし、表示されるコンテキストメニューから ユーザー定義ファンクションのインポート を選択します。
4	ユーザー定義ファンクションファイル (*.smbf) を含むフォルダーに移動し、ユーザー定義ファンクションを選択します。

ユーザー定義ファンクション名の変更

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	以下の方法で名前の変更をします。 <ul style="list-style-type: none"> ● ユーザー定義ファンクションを右クリックし、コンテキストメニューからユーザー定義ファンクション名の変更を選択します。 ● プログラミングワークスペースでユーザー定義ファンクションをダブルクリックします。 ● ユーザー定義ファンクションを選択し F2 キーを押します。
3	ユーザー定義ファンクションの新しい名前を入力し ENTER キーを押します。 使用できる文字は、A..Z, 0..9, _ です。名前は一意なものにします。そうでない場合は、名前は変更されません。

ユーザー定義ファンクションの削除

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	CTRL キーを押しながら 1 つ以上のユーザー定義ファンクションを選択します。
3	次のいずれかの方法を使用して、選択したユーザー定義ファンクションを削除します。 <ul style="list-style-type: none"> ● ユーザー定義ファンクションで選択したユーザー定義ファンクションを右クリックし、表示されるコンテキストメニューからユーザー定義ファンクションの削除を選択します。 ● DELETE キーを押します。 <ul style="list-style-type: none"> ● タスクタブの上部にあるツールバーの  をクリックします。

6.6

ユーザー定義ファンクションブロック

概要

ユーザー定義のファンクションブロックを使用して、1つ以上の入力パラメーター、出力パラメーターおよびローカル変数を含む新しいファンクションブロックを作成できます。ユーザー定義ファンクションブロックは、SoMachine Basic プロジェクトの一部として保存されます。

ユーザー定義ファンクションブロックは、次の場所で呼び出すことができます。

- マスタータスク
- 周期タスク
- イベント
- Free POU

注記：ユーザー定義ファンクションブロックを使用する際は、アプリケーションのファンクションレベル (77 ページ参照) を **6.0** 以上に設定してください。


このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
ユーザー定義ファンクションブロックの作成	100
ユーザー定義ファンクションブロックの定義	101
ユーザー定義ファンクションブロックの管理	103

ユーザー定義ファンクションブロックの作成

新しいユーザー定義ファンクションブロックの追加

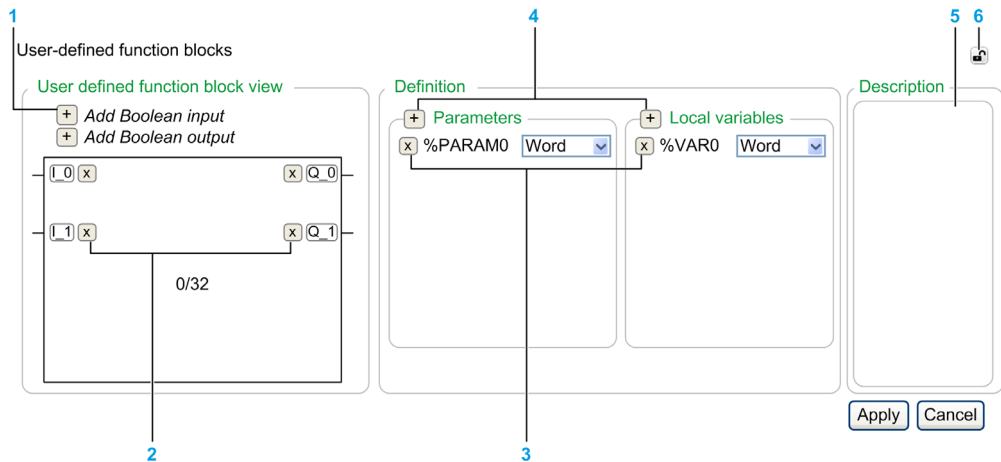
手順	手順内容
1	プログラミングウィンドウの左側にあるタスクタブを選択します。
2	<p>次のいずれかの方法を使用して、ユーザー定義ファンクションブロックを作成します。</p> <ul style="list-style-type: none"> ● ユーザー定義ファンクションブロックを右クリックし、コンテキストメニューからユーザー定義ファンクションブロックの追加を選択します。 ● ユーザー定義ファンクションブロックを選択し、タスクタブの上部にあるツールバーの  (ユーザー定義ファンクションブロックの追加) をクリックします。 ● 既存のユーザー定義ファンクションブロックを選択し、右クリック、ユーザー定義ファンクションブロックを挿入を選択します。 <p>結果: 新しいユーザー定義ファンクションブロックがリストの一番下のプログラム構造体に追加されます。ユーザー定義ファンクションブロックを挿入すると、新しいユーザー定義ファンクションブロックは選択したファンクションの上に挿入されます。名前の初期値は、UDFB です。ここで、n はユーザー定義ファンクションブロックが作成されるたびにインクリメントされる整数です。</p>
3	必要に応じて、ユーザー定義ファンクションブロック名を変更します。ユーザー定義ファンクションブロック名の変更 (104 ページ参照) を参照してください。
4	ユーザー定義ファンクションブロックを定義します。ユーザー定義ファンクションブロック名の定義 (101 ページ参照) を参照してください。

通常の POU のラングと同じ方法で、ユーザー定義ファンクションブロックでラングの作成および管理ができます。ラングの管理 (84 ページ参照) を参照してください。

ユーザー定義ファンクションブロックの定義

プレゼンテーション



次の図は、ユーザー定義ファンクションブロックのプロパティを示しています。



- 1 入力または出力を追加する
- 2 入力または出力を削除する
- 3 パラメーターまたはローカル変数を削除する
- 4 パラメーターまたはローカル変数を追加する
- 5 ユーザー定義ファンクションブロックの目的の詳細 (オプション)。この詳細は、演算ブロックでユーザー定義ファンクションブロックを使用するときにツールチップに表示されます。
- 6 プロパティビューを分離する

ユーザー定義ファンクションブロックのプログラミング

ユーザー定義ファンクションブロックをプログラムします。

手順	手順内容
1	新しいユーザー定義ファンクションブロックを追加します。ユーザー定義ファンクションブロックの追加 (100 ページ参照) を参照してください。
2	入力パラメーターおよびローカル変数を定義して、ユーザー定義ファンクションブロックのインターフェースを定義します。ユーザー定義ファンクションブロックの定義 (102 ページ参照) を参照してください。
3	適用 をクリックします。
4	1つ以上のラダー /IL ラング (84 ページ参照) のユーザー定義ファンクションブロックの機能を定義します。 <ol style="list-style-type: none">1. ツールバー上のファンクションブロック  ボタンをクリックします。2.  を選択し、次に挿入するユーザー定義ファンクションブロックを選択します。3. ラングのアクションゾーン (155 ページ参照) をクリックします。4. ユーザー定義ファンクションブロックをプログラムします。

また、IL エディターでユーザー定義ファンクションブロックをプログラムすることはできません。

ローカル変数とグローバル変数

ローカル変数は、ユーザー定義ファンクションブロックを実装するラングでのみ使用できます。

- %PARAMn
- %VARn

グローバル変数は、ユーザー定義ファンクションブロックのラングを含む SoMachine Basic プログラムで使用できるその他すべての変数です。



ユーザー定義ファンクションブロックのインタフェースの定義

ユーザー定義ファンクションブロックを使用するには、入力、出力、オブジェクトおよびそれらのデータ型を定義してください。

オブジェクトタイプ	データ型	詳細
パラメーター %PARAMn ⁽¹⁾	ワード ダブル フロート	ユーザー定義ファンクションおよびユーザー定義ファンクションブロックのラングでのみ使用できます。 パラメーターアドレスの初期値は変更できません。 パラメーターをアニメーションテーブルに追加できません。
ローカル変数 %VARn ⁽¹⁾		ユーザー定義ファンクションおよびユーザー定義ファンクションブロックのラングでのみ使用できます。 ローカル変数アドレスの初期値は変更できません。 ローカル変数をアニメーションテーブルに追加できません。
⁽¹⁾ n は、パラメーターまたはローカル変数が作成されるたびにインクリメントされる整数です。		

ユーザー定義ファンクションブロックの使用

ユーザー定義ファンクションブロックをラングに挿入するには：

手順	手順内容
1	ツールバー上の ファンクションブロック  ボタンをクリックします。
2	 を選択し、次に挿入するユーザー定義ファンクションブロックを選択します。
3	ラングのアクションゾーン (155 ページ参照) をクリックします。
4	必要に応じて、演算ブロックを使用して、ファンクションブロックのパラメーターをの読み込み、および書き込みをします。 構文は %<UDFB name><instance number>.PARAMn、ここで n はパラメーター番号に対応する整数です。 例： <ul style="list-style-type: none"> ● パラメーターが %PARAM 0 の MY_FB という名前のユーザー定義ファンクションブロックを定義します。 ● このユーザー定義ファンクションブロックのインスタンスは、マスタータスクに配置され、インスタンス番号 0 が割り当てられます。 結果： オブジェクト %MY_FB0.PARAM0 はどのタスクでも使用できます。

ユーザー定義ファンクションブロックの管理

オフラインおよびオンラインモードでのユーザー定義ファンクションブロック

ユーザー定義ファンクションブロックはオフラインモードで管理することができます。

オンラインモードでは、既存のユーザー定義ファンクションブロックにラングを追加できます。

既存のユーザー定義ファンクションブロックのコピー、切り取り、貼り付け

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	1つ以上のユーザー定義ファンクションブロックを選択します。 <ul style="list-style-type: none">● ユーザー定義ファンクションブロックをクリックして選択します。● CTRL キーを押しながら複数のユーザー定義ファンクションブロックを選択します。
3	右クリックで表示されるコンテキストメニューから ユーザー定義ファンクションブロックのコピー または ユーザー定義ファンクションブロックの切り取り を選択します。
4	ユーザー定義ファンクションブロック を右クリックし、コンテキストメニューから ユーザー定義ファンクションブロックの貼り付け を選択します。 結果 :1つ以上のユーザー定義ファンクションブロックが、 ユーザー定義ファンクションブロックプログラム構造体 の最後に追加されます。SoMachine Basic は、自動的にコピーされたユーザー定義ファンクションブロックに新しい名前を割り当てます。

ユーザー定義ファンクションブロックのエクスポート

ユーザー定義ファンクションブロックは、プロジェクトの一部として保存されます。別のプロジェクトでユーザー定義ファンクションブロックを使用する場合は、そのファンクションをエクスポートしてから、別のプロジェクトにインポートしてください。

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	1つ以上のユーザー定義ファンクションブロックを選択します。 <ul style="list-style-type: none">● ユーザー定義ファンクションブロックをクリックして選択します。● CTRL キーを押しながら複数のユーザー定義ファンクションブロックを選択します。
3	ユーザー定義ファンクションブロック のユーザー定義ファンクションブロックを右クリックし、表示されるコンテキストメニューから ユーザー定義ファンクションブロックのエクスポート を選択します。
4	ユーザー定義ファンクションブロックファイル (*.smbf) を エクスポートフォルダー に保存します。

ユーザー定義ファンクションブロックのインポート



ユーザー定義ファンクションブロックは、プロジェクトの一部として保存されます。別のプロジェクトでユーザー定義ファンクションブロックを使用する場合は、そのファンクションをエクスポートしてから、別のプロジェクトにインポートしてください。

手順	手順内容
1	プログラミングウィンドウの左側にあるタスクタブを選択します。
2	ユーザー定義ファンクションブロックのユーザー定義ファンクションを選択します。
3	ユーザー定義ファンクションブロックのユーザー定義ファンクションを右クリックし、表示されるコンテキストメニューからユーザー定義ファンクションブロックのインポートを選択します。
4	ユーザー定義ファンクションブロックファイル (*.smbf) を含むフォルダーに移動し、ユーザー定義ファンクションを選択します。

ユーザー定義ファンクションブロック名の変更

手順	手順内容
1	プログラミングウィンドウの左側にあるタスクタブを選択します。
2	以下の方法で名前の変更をします。 <ul style="list-style-type: none"> ● ユーザー定義ファンクションブロックを右クリックし、コンテキストメニューからユーザー定義ファンクションブロック名の変更を選択します。 ● プログラミングワークスペースでユーザー定義ファンクションブロック名をダブルクリックします。 ● ユーザー定義ファンクションブロックを選択し F2 キーを押します。
3	ユーザー定義ファンクションブロックの新しい名前を入力し ENTER キーを押します。使用できる文字は、A..Z, 0..9, _ です。名前は一意なものにします。そうでない場合は、名前は変更されません。

ユーザー定義ファンクションブロックの削除

手順	手順内容
1	プログラミングウィンドウの左側にあるタスクタブを選択します。
2	CTRL キーを押しながら 1 つ以上のユーザー定義ファンクションブロックを選択します。
3	次のいずれかの方法を使用して、選択したユーザー定義ファンクションブロックを削除します。 <ul style="list-style-type: none"> ● ユーザー定義ファンクションブロックのユーザー定義ファンクションブロックを右クリックし、表示されるコンテキストメニューからユーザー定義ファンクションブロックの削除を選択します。 ● DELETE キーを押します。 <div style="text-align: center; margin: 10px 0;">  </div> <ul style="list-style-type: none"> ● タスクタブの上部にあるツールバーの  をクリックします。

6.7

マスタートスク

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
マスタートスクの説明	106
マスタートスクの設定	107

マスタータスクの説明

概要

マスタータスクはアプリケーションプログラムのメインタスクです。これは必須であり、初期設定で作成されます。マスタータスクは、プログラム構成単位 (POU) の中でセクションとサブルーチンで構成されています。マスタータスクの各 POU は、対応しているプログラミング言語でプログラムできます。

手順

目的	参照
マスタータスクに新規 POU を作成	タスクに関連付けられた新規 POU のを作成 (82 ページ参照)
マスタータスクの POU の名前を変更	POU の名前を変更 (83 ページ参照)
マスタータスクから POU を削除	POU をの削除 (83 ページ参照)

マスタータスクの設定

手順

次の手順で、マスタータスクを設定します。

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ を選択します。
2	マスタータスクを選択します。 結果：マスタータスクプロパティが SoMachine Basic ウィンドウの中央下部に表示されます。
3	プロパティを必要に応じて変更します。
4	適用 をクリックして変更を保存します。

マスタータスクプロパティ

スキャンモード

プログラムに使用するスキャンモードを選択します。

- **連続**: ロジックコントローラーが連続 (フリーホーリング) スキャンモードの時は、スキャンが終わると直ちに次のスキャンが開始します。
- **周期**: ロジックコントローラーが周期スキャンモードの時は、設定されたスキャンタイムが経過してから次のスキャンが開始します。すべてのスキャンは同じ期間です。
周期スキャンモードの**周期**を 2...150 ms に設定します。

マスタータスクを制御するシステムビットシステムワード

システムビット (%S) とシステムワード (%SW) でマスタータスクを制御します。

システムビットを次の表に示します。

システムビット	説明
%S11	ウォッチドッグオーバーフロー
%S19	スキャン周期オーバーラン (周期スキャンモード)

システムワードを次の表に示します。

システムワード	説明
%SW0	ロジックコントローラー スキャン周期 (周期スキャンモード)
%SW30, %SW70	最後のスキャンタイム。最後のコントローラー スキャンサイクルの実行時間です。マスタータスクのスキャンサイクルの開始 (入力の取得) から終了 (出力の更新) までに経過した時間を示します。%SW30 はミリ秒、%SW70 はマイクロ秒を示します。 スキャンタイムが 2.250 ms の場合、%SW30 = 2 と %SW70 = 250 になります。
%SW31, %SW71	最大スキャンタイム。ロジックコントローラーの最後のコールドスタート以降、最長のコントローラー スキャン時間の実行時間を示します。%SW31 はミリ秒、%SW71 はマイクロ秒を示します。 スキャンタイムが 2.250 ms の場合、%SW31 = 2 と %SW71 = 250 になります。
%SW32, %SW72	最小スキャンタイム。ロジックコントローラーの最後のコールドスタート以降、最短のコントローラー スキャン時間の実行時間を示します。%SW31 はミリ秒、%SW71 はマイクロ秒を示します。 スキャンタイムが 2.250 ms の場合、%SW32 = 2 と %SW72 = 250 になります。

システムビットとシステムワードの詳細は、使用しているハードウェアプラットフォームの**プログラミングガイド**を参照してください。

6.8 文字列

概要

文字列は、次のメモリーオブジェクトに格納できる ASCII 文字を含む連続するバイトです。

- ワードメモリー %MW
- ワード型定数 %KW

1 ワードは 2 バイトです。

文字列をプログラムする構文を次に示します。

```
%MWx:L
```

x メモリーオブジェクトのインデックス

L 文字列に使用するワード数は 1 ~ 255 にしてください。

対応しているすべてのコントローラーはリトルエンディアンアーキテクチャーです。最下位バイトから最上位バイトの順で格納されます。

次の表は、文字列 *Basic* のバイトの格納例を示しています。

メモリーオブジェクト	16 進数	ASCII
%MW0 または %KW0	6142	aB
%MW1 または %KW1	6973	is
%MW2 または %KW2	0D63	\rc ⁽¹⁾

(1) \r は文字列の最後の文字を表すマーカーです。このマーカーは、文字列を処理するときに使用されます。

使用できる文字数は最大 509 個です。

注記：メモリーオブジェクトは、変数または文字列として使用されます。メモリーオブジェクトに文字列を設定した場合は、変数としてえ r 設定しないでください。


このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
ワード型定数の文字列設定	109
ワードメモリー内の文字列の割り当て	110
文字列の管理	111

ワード型定数の文字列設定

文字列の入力

手順	手順内容
1	プログラミングウィンドウで、ツール → メモリーオブジェクト → ワード型定数をクリックします。
2	ワード型定数のプロパティの %KW をクリックします。
3	設定したいワード型定数の設定列の  ボタンをクリックします。 ワード型定数がすでに設定されている場合、確認ウィンドウが表示されます。OK をクリックして値を上書き、または キャンセル をクリックしてください。 結果：定数文字列アシスタントウィンドウが表示されます。
4	文字列を入力します。 結果：必要な定数の範囲により、文字列に使用されるワード型定数が定義されます。
5	適用をクリックします。

結果：入力された文字は、対応する必要定数変数に適用されます。文字は逆になります。このセクションの概要 (108 ページ参照) を参照してください。

ワードメモリー内の文字列の割り当て

構文

IL 構文は次のとおりです。演算ブロックグラフィカル要素を使用して、ラダーダイアグラムのラングに IL 演算と割り当ての命令を挿入 (164 ページ参照) できます。

ワードメモリーに文字列を割り当てるには、次の構文を使用します。Op1 := "Your string"

例：

```
%MW10:20 := "This is a SoMachine Basic string."
```

必要なメモリー容量を無視する場合、次のように入力します。%MWx:? := "Your string"

使用規則

文字列を割り当てる場合

- 重複がないことを確認してください。別の文字列で文字列を消去することができます。
- “ は使用できません。

文字列の管理

はじめに

次のファンクションを使用すると、以下のことが可能です。

- 文字列のコピー。
- 文字列の長さを取得。
- 2つの文字列を連結。
- 2つの文字列を比較。

構文

IL 構文は次のとおりです。**演算ブロック**グラフィカル要素を使用して、ラダーダイアグラムのラングに IL 演算と割り当ての命令を挿入 (164 ページ参照) できます。

文字列のコピー

文字列をコピーするには、次の構文を使用します。Op1 := Op2.

次の表は、Op1 および Op2 に許可されているメモリーオブジェクトを示します。

パラメーター	詳細
Op1	%MWx:L
Op2	%MWy:L または %KWy:L
x, y はメモリーオブジェクトのインデックスです。 L は Op1 および Op2 で同じものを使用します。	

直文字列 (文字列リテラル) は使用できません。

文字列の長さを取得

文字列の長さを取得するには、次の構文を使用します。Op1 := LENGTH(Op2).

次の表は、Op1 および Op2 に許可されているメモリーオブジェクトを示します。

パラメーター	詳細
Op1	%MWx
Op2	%MWy:L または %KWy:L
x, y はメモリーオブジェクトのインデックスです。	

直文字列 (文字列リテラル) は使用できません。

2 つの文字列を連結

連結された 2 つの文字列をコピーするには、次の構文を使用します。Op1 := CONCAT(Op2, Op3).

次の表は、Op1、Op2 および Op3 に許可されているメモリーオブジェクトを示します。

パラメーター	詳細
Op1	%MWx:L
Op2	%MWy:A または %KWy:A
Op3	%MWz:B または %KWz:B
x, y, z はメモリーオブジェクトのインデックスです。	
SoMachine Basic は、L、A、B を比較しません。	

直文字列 (文字列リテラル) は使用できません。

次の表は、連結の処理を示しています。

手順	詳細
1	アプリケーションは Op2 を Op1 にコピーします。
2	コピーは次の場合に停止します。 <ul style="list-style-type: none"> ● Op2 の終了文字に達した場合。 ● Op2 に割り当てられたメモリー領域がコピーされ、%S28 が生成された場合。システムビット詳細を参照してください。 ● Op1 のメモリーがいっぱいで、%S28 が生成された場合。
3	Op1 のメモリーに空きがある場合、アプリケーションは Op3 を Op1 にコピーすることによって続きます。
4	コピーは次の場合に停止します。 <ul style="list-style-type: none"> ● Op3 の終了文字に達した場合。 ● Op3 に割り当てられたメモリー領域がコピーされ、%S28 が生成された場合。 ● Op1 のメモリーがいっぱいで、%S28 が生成された場合。
5	アプリケーションは、Op1 が終了文字で終わるようにします。メモリーがいっぱいになると、Op1 の最後の文字が終了文字に置き換えられます。

2つの文字列を比較

連結された2つの文字列を比較するには、次の構文を使用します。Op1 := EQUAL_STR(Op2, Op3).

次の表は、Op1、Op2 および Op3 に許可されているメモリーオブジェクトを示します。

パラメーター	詳細
Op1	%MWx
Op2	%MWy:A または %KWy:A
Op3	%MWz:B または %KWz:B
x, y, z はメモリーオブジェクトのインデックスです。	
SoMachine Basic は、A と B を比較しません。	

アプリケーションが差異のある文字を検出すると、Op1 は差異のある文字のインデックス位置と等しくなります。

次の表に、文字列の比較結果の例を示します。

Op2	Op3	差異のある文字	Op1
azerty	qwerty	1 文字目	0
123456	124356	3 文字目	2
SoMachine Basic	SoMachine Basic	-	-1

次の表は、文字列比較のプロセスを示しています。

IF	And IF	Then
アプリケーションが Op2 の終了文字に到達。	Op2 = Op3	Op1 := -1
	Op2 ≠ Op3	Op1 は差異のある文字の位置に等しい。
アプリケーションは Op2 または Op3 の終了文字に到達する前に差異のある文字を検出します。	-	Op1 は差異のある文字の位置に等しい。
Op2 または Op3 に割り当てられたメモリーに空きがない。	A ≠ B	Op1 は差異のある文字の位置に等しく、%S28 が生成されます。システムビットの説明を参照してください。
	A = B	Op1 := -1 および %S28 が生成される。

6.9 周期タスク

このセクションについて

このセクションには次の項目が含まれています。



項目	参照ページ
周期タスクの作成	115
周期タスクのスキャン期間設定	117

周期タスクの作成


概要

周期タスクとは、周期的に処理される短期間のサブルーチンのことです。SoMachine Basic では、このサブルーチンは Free POU (89 ページ参照) として実装されています。サブルーチンは、SoMachine Basic で対応しているプログラミング言語で書くことができます。

サブルーチンの周期タスクへの割り当て

手順	手順内容
1	周期タスクサブルーチンを含む新規 Free POU を作成 (89 ページ参照) します。
2	プログラミングウィンドウの左側にある タスク タブを選択します。
3	<p>以下の方法で、サブルーチンを周期タスクに割り当てます。</p> <ul style="list-style-type: none">● 周期タスクを選択し、タスクタブの上部にあるツールバーの  (Free POU の割当てボタン) をクリックします。● 周期タスクを右クリックして、表示されたコンテキストメニューから Free POU の割当てを選択します。 <p>結果 : Free POU を選択ウィンドウが表示されます。</p> <div data-bbox="353 813 872 1251"></div> <p>注記 : Free POU を周期タスクに直接追加することもできます。周期タスクを右クリックして、表示されたコンテキストメニューから Free POU の追加を選択します。この場合、Free POU が作成され、周期タスクに割り当てられます。</p>
4	<p>周期タスクに割り当てる Free POU を選択し、OK をクリックします。</p> <p>結果 : 選択されたサブルーチンは、周期タスクに割り当てられ、タスクタブの Free POU ブランチにはありません。</p> <p>例えば、サブルーチン SR4 を含む Free POU、“Free POU_0” が周期タスクに割り当てられている場合、Free POU_0 (%SR4) サブルーチンは、タスクタブの Free POU ブランチから周期タスク ブランチへ移動します。</p>

周期タスクからサブルーチンを削除

手順	手順内容
1	<p>プログラミングウィンドウの左側にあるタスクタブをクリックします。</p>
2	<p>以下の方法で、周期タスクからサブルーチンを削除します。</p> <ul style="list-style-type: none"> ● 周期タスクを選択し、タスクタブの上部にあるツールバーの  (Free POU の解除ボタン) をクリックします。 ● 周期タスクを右クリックして、表示されたコンテキストメニューから Free POU の解除を選択します。 <p>結果：選択されたサブルーチンは、周期タスクから削除され、タスクタブの Free POU ブランチで Free POU として利用可能になります。</p>

周期タスクのスキャン期間設定

設定手順

次の手順で、周期タスクのスキャン期間を設定します。

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	周期タスクアイテムを選択します。 結果：周期タスクプロパティが、SoMachine Basic の下中央部のウィンドウに表示されます。
3	プロパティを必要に応じて変更します。
4	適用 をクリックして変更を保存します。

周期タスクプロパティ

周期タスク (1...255 ms) のスキャン**周期**を指定します。初期値は 255 ms です。

6.10

イベントタスク

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
イベントタスクの概要	119
イベントソース	120
イベント優先度	121
イベントタスクの表示	122

イベントタスクの概要

はじめに

イベントタスクは：

- 規定の条件が満たされたときに実行されるプログラムの一部です (イベントソース)。
- メインプログラムより優先度が高いです。
- 応答時間が速いため、システム全体の応答時間を減らすことができます。

イベントの詳細

イベントは以下により構成されています。

- **イベントソース**: イベントがトリガーされたときにプログラムに割り込むソフトウェア、またはハードウェアの状態。
- **POU**: イベントに関連付けられている独立したプログラムのサブルーチン。
- **優先度**: イベントが実行される順序を決める、イベントに指定されている優先度。

イベントソース

概要

8つのイベントソースが利用可能です。

- 4つは、ロジックコントローラーの選択された物理的入力にリンクされています。
- その他の4つは、HSC ファンクションブロックの閾値にリンクされています。

1つのイベントソースが、常に1つのイベントに付加されます。イベントがトリガーされると、イベントはコントローラーに割り込み、コントローラーはイベントに関連付けられたサブルーチンを実行します。

ロジックコントローラーの物理的入力イベント

ロジックコントローラーの標準デジタル入力 %I0.2、%I0.3、%I0.4、および %I0.5 は、イベントソースとして設定することができます (フィルタリングは無効にします)。

これらの各イベントソースに、以下を選択できます。

- 立上がり接点、立下り接点、または両方が検出されたときに、イベントをトリガーする。
- イベントに優先度を指定する。
- イベントに関連付けられているサブルーチンを識別する。

入力イベント設定の詳細は、ロジックコントローラーの *プログラミングガイド* を参照してください。

%HSC ファンクションブロックの閾値出力イベント

%HSC ファンクションブロックの閾値出力 TH0、および TH1 はイベントソースとして使用できます。出力 TH0、および TH1 は以下のように設定されます。

- 値が閾値 S0、および閾値 S0 より小さいとき、TH0 = 0、および TH1 = 0
- 値が閾値 S0 より大きく、閾値 S0 より小さいとき、TH0 = 1、および TH1 = 0
- 値が閾値 S0、および閾値 S0 より大きいとき、TH0 = 1、および TH1 = 1

これらの各イベントソースに、以下を選択できます。

- 立上がり接点、立下り接点、または両方が検出されたときに、イベントをトリガーする。
- イベントに優先度を指定する。
- イベントに関連付けられているサブルーチンを識別する。

これらの出力の立上がり、または立下り接点で、イベント処理を有効にすることができます。

出力イベント設定の詳細は、ロジックコントローラーの *プログラミングガイド* を参照してください。

イベント優先度

イベント優先度

イベントには7(最低)から0(最高)までの8つの優先度のうちの1つが付いています。

各イベントソースで優先度を指定します。2つのイベントに同じ優先度は指定できません。実行順序は、相対的な優先度と、検出される順序に依存します。

イベントタスクは、マスタートaskと周期タスクの実行に割り込みます。詳細は、マスタートaskと周期タスクのイベント優先度(79ページ参照)を参照してください。

注記: 他のタスクの実行中イベントタスクが呼び出されたときに、メモリーのグローバル領域へ書き込み、またはI/O値に影響を与える場合は注意してください。他のタスクで使用されている値を変更すると、それらのタスクの論理結果に影響する可能性があります。

警告

装置の意図しない動作

アプリケーションを運用する前に、すべてのタスク(マスター、周期、およびイベントタスク)とその相互作用を十分にテスト、および検証してください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

イベントタスクの優先度設定については、コントローラーのプログラミングガイドを参照してください。

イベント管理

イベントソースにリンクされた割り込みがあるたびに、以下のシーケンスが開始されます。

手順	説明
1	イベント割り込みが発生
2	コンテキストを保存
3	イベントにリンクされたプログラミングセクション(SRi: サブルーチン)の実行
4	標準出力の更新
5	コンテキストの復元

イベントタスクの表示

概要

イベントタスクは、**設定** タブに表示されます。デジタル入力設定 を参照してください。

設定されたイベントソースや、イベントに付加されたサブルーチンを表示し、システムビット、およびワードを使用してイベントのステータスを確認することができます。



イベントに割り当てられたイベントソース、およびサブルーチン (Free POU) を表示します。

手順	手順内容
1	プログラミングウィンドウの左側にある タスク タブを選択します。
2	<p>イベントを選択します。</p> <pre> Events %HSC0.TH0 : %HSC0.TH1 : %I0.2 : %I0.3 : Free POU_0 Rung0 </pre> <p>注記：まだサブルーチンが割り当てられていないイベントソースは赤で表示されます。</p>

注記：イベントサブルーチンで使用できるのは、標準コントローラー入出力のみです。


Free POU のイベントソースへの割り当て

設定したイベントソースに Free POU を割り当てるには、以下の手順で行います。

手順	手順内容
1	イベントに使用するサブルーチンを含む 新規 Free POU を作成 (89 ページ参照) します。
2	プログラミングウィンドウの左側にある タスクタブ を選択します。
3	<p>以下の方法で、サブルーチンをイベントソースに割り当てます。</p> <ul style="list-style-type: none">● イベント一覧のイベントソースを選択し、タスクタブの上部にあるツールバーの  (Free POU の割り当てボタン) をクリックします。● イベント一覧のイベントソースを右クリックして、表示されたコンテキストメニューから Free POU の割り当て を選択します。 <p>結果 : Free POU を選択ウィンドウが表示されます。</p>  <p>注記 : Free POU をイベントソースに直接追加することもできます。イベント一覧のイベントソースを右クリックして、表示されたコンテキストメニューから Free POU を選択 を選択します。この場合、Free POU が作成され、イベントソースに割り当てられます。</p>
4	<p>イベントソースに割り当てる Free POU を選択し、OK をクリックします。</p> <p>結果 : 選択されたサブルーチンは、イベントソースに割り当てられ、タスクタブの Free POU ブランチには存在しません。</p> <p>例えば、サブルーチン SR1 を含む Free POU、“Free POU_0” がイベントソースに割り当てられている場合、Free POU_0 (%SR1) サブルーチンは、タスクタブの Free POU ブランチからイベントソースブランチへ移動します。</p>

イベントからサブルーチンを削除

次の手順で、サブルーチンとイベントソースの関連付けを削除します。

手順	手順内容
1	プログラミングウィンドウの左側にある タスクタブ をクリックします。
2	以下の方法で、イベントソースからサブルーチンを削除します。 <ul style="list-style-type: none"> ● イベント一覧のイベントソースを選択し、タスクタブの上部にあるツールバーの  (Free POU の解除ボタン) をクリックします。 ● イベント一覧のイベントソースを右クリックして、表示されたコンテキストメニューから Free POU の解除 を選択します。 <p>結果：選択されたサブルーチンは、イベントソース から削除され、タスクタブの Free POU プランチで Free POU として利用可能になります。</p>

システムビットとワードによるイベント確認

以下のシステムビットが、イベントの確認に使われます。

システムビット	説明
%S38	イベント処理を有効 (%S38 = 1)、または無効 (%S38 = 0) にするために使用。
%S39	イベント損失の判断に使用。

以下のシステムワードが、イベントの確認に使われます。

システムワード	説明
%SW48	ロジックコントローラーの最後のコールドスタート後に実行されたイベントの数。

コールドスタート、またはアプリケーションのロード後、%S39、および %SW48 の値は 0 にリセットされ、システムビット %S38 の値は初期ステート 1 に設定されます。ウォームスタート後は、値は変更されません。

6.11

ツールを使う

このセクションについて


このセクションには次の項目が含まれています。

項目	参照ページ
メッセージ	126
アニメーションテーブル	128
メモリーオブジェクト	133
システムオブジェクト	136
I/O オブジェクト	137
ネットワークオブジェクト	138
ソフトウェアオブジェクト	139
PTO オブジェクト	140
ドライブオブジェクト	141
通信オブジェクト	142
検索と置換	143
クロスリファレンス	145
シンボルリスト	146
メモリー消費量の表示	150

メッセージ

概要

SoMachine Basic では、プログラムを編集している間**プログラミング**タブでソースコードが分析されます。



また SoMachine Basic では、ツールバーの**コンパイル**ボタン  をクリックした時にもプログラムが分析されます。


エラーやアドバイザリーが検出された場合、クリック可能なアイコンが**プログラミング**タブに表示されます。



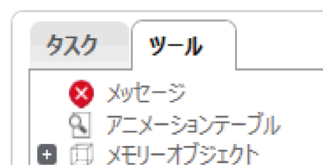
このアイコンをクリックするとメッセージウィンドウが表示されます。

表示されるアイコンは、メッセージの重大度によって異なります。

アイコン	意味
	アドバイザリー。ラングが不完全です。
	シンタックスエラーが検出されました。

アドバイザリーとエラーメッセージ両方が検出された場合、エラーアイコン  のみ表示されます。

アイコンは、**メッセージ**タブの横にある**ツール**タブにも表示されます。



メッセージを表示する

以下に、エラー、およびアドバイザリーメッセージのリストを表示する手順を示します。

手順	手順内容
1	プログラミング タブのアイコンをクリックします。 または ツール → メッセージ をクリックします。 メッセージリストが プログラミング ウィンドウの中央下部に表示されます。
2	メッセージ エリアで、 アドバイザリー ボタンをクリックしてアドバイザリーメッセージを表示し、 エラー ボタンをクリックしてエラーメッセージを表示します。ボタンをもう一度クリックするとメッセージリストが非表示になります。

ラングステータス

SoMachine Basic では、プログラムの各ラングのステータスが個別に表示されます。

ラングの構文が正しく問題がない場合は、メッセージ表示はなく緑のチェックマークが表示されます。

IL Rung0	名前	コメント
	0000	LD %I0.0
0001	ST %Q0.0	コメント

ラングが完全でない場合、例えば、END、CALL、または Jump などの最後の命令が含まれていない場合は、アドバイザーのアイコンが表示されます。

IL Rung0	名前	コメント
0000	LD %I0.0	コメント
0001	AND %I0.1	コメント

コンパイルの成功を妨げるような間違っ構文が SoMachine Basic で検出されると、エラーアイコンが表示されます。

IL Rung0	名前	コメント
0000	BLK %SBR0	コメント
0001	LD %S6	コメント
0002	CU	コメント

アドバイザーアイコンとエラーアイコンは、**タスクタブ**の各ラングの横に表示されます。

- 1 - M_ZeroPressureAccumulator
 - ✖ Rung0
 - Rung1
 - Rung2
 - Rung3 - Rung_1
 - ✖ Rung4 - Rung_3
 - Rung5
 - Rung6 - Rung_2
 - Rung7
 - ✖ Rung8

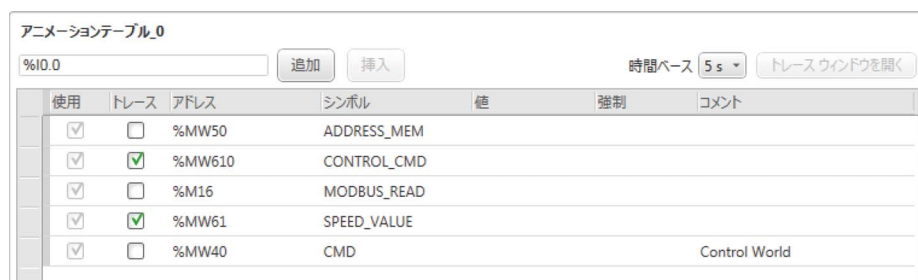
アニメーションテーブル

概要

オブジェクトをアニメーションテーブルに追加して以下を行います。

- オブジェクトのシンボルやコメントを表示します。
- ロジックコントローラーに SoMachine Basic が接続されている場合 (オンラインモード)、特定のオブジェクトタイプのリアルタイムの値を表示および編集します。
- トレースウィンドウ (200 ページ参照) に表示させるオブジェクトを選択します。

アニメーションテーブルは、SoMachine Basic アプリケーションのコンポーネントで、プログラムとともにプログラム以外のデータとしてロジックコントローラーにダウンロードされます。これにより、アプリケーションが後でロジックコントローラーからアップロードされるときに、アニメーションテーブルに格納されたオブジェクトを取り出すことができます。



存在しないオブジェクトをアニメーションテーブルに追加すると、**値**フィールドが赤枠で表示されます。例えば、%Q1.0 を追加し、それに対応するデジタル出力モジュールが設定にない場合です。



アニメーションテーブルを作成する

手順	手順内容
1	プログラミングウィンドウの左側にあるツールタブを選択します。
2	アニメーションテーブルを右クリックし、コンテキストメニューから 新規アニメーションテーブルの追加 を選択します。 結果：ツールウィンドウのアニメーションテーブルエリアに新しいアニメーションテーブルが表示されプロパティウィンドウがツールウィンドウの中央下部に表示されます。

アニメーションテーブルに個々のオブジェクトを追加

手順	手順内容
1	プログラミングウィンドウの左側にあるツールタブを選択します。
2	ツールウィンドウのアニメーションテーブルエリアで設定するアニメーションテーブルを選択します。 結果：プロパティウィンドウがウィンドウの中央下部に表示されます。

手順	手順内容
3	<p>新しいオブジェクトをアニメーションテーブルの最後に追加するには、テキストボックスにオブジェクト名を入力し Enter キーを押すか、追加をクリックします。</p> <p>以下に、アニメーションテーブルに追加できるオブジェクトを示します。</p> <ul style="list-style-type: none"> ● I/O オブジェクト ● ファンクションブロックオブジェクト。例えば、タイマーファンクションブロック %TM0、%TM0.V、%TM0.P、および %TM0.Q は、自動的にアニメーションテーブルに追加されます。 ● ビット列 (例: %Mx:L、L はビット数、8 の倍数) ● ワードテーブル (例: %MWx:L、L はワード数) ● ワードのビット (例: %MWx:X、X はビットのオフセット) ● ネットワークオブジェクト (%QWE、%IWE、%QWM、%IWM) <p>注記: ネットワークオブジェクトは、ロジックコントローラーの EtherNet/IP アダプター (Modicon M221、ロジックコントローラー、プログラミングガイド参照) が有効な場合または Modbus TCP 設定で Modbus マッピング が有効な場合にのみ使用できます。</p>
4	<p>既存オブジェクトのすぐ上に新しいオブジェクトを追加したい場合は、アニメーションテーブルの行を選択し、テキストボックスに追加するオブジェクトの名前を入力して挿入をクリックします。</p>

アニメーションテーブルの I/O オブジェクトのアドレスは、設定の変更後に自動的に変更されません。例えば、対応しているモジュールの位置が設定で変更されても %Q3.0 は自動的に %Q1.0 に変更されません。アプリケーション内の I/O メモリー割り当ての変更すべてを考慮し、それに応じて更新してください。

警告

装置の意図しない動作

設定を変更した後は、アプリケーションで使われている即時 I/O アドレスを必要に応じて確認、および変更してください。


上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

設定の変更後には必ずアニメーションテーブルを確認して更新します。

ラングのオブジェクトすべてをアニメーションテーブルに追加する

手順	手順内容
1	アニメーションテーブルが1つ以上ある場合は、ツールウィンドウのアニメーションテーブルエリアで、アニメーションテーブルを1つ選択します。 結果: アニメーションテーブル プロパティ ウィンドウがウィンドウの中央下部に表示されます。
2	タスク ウィンドウを選択します。
3	ラングを右クリックし、アニメーションテーブルにラングオブジェクトを追加をコンテキストヘルプから選択します。 結果: ラングで使用されているオブジェクトがアニメーションテーブルに追加されます。

注記:

- ラングにエラーが検出されていないことを確認します (エラーアイコン  が表示されていない)。
- ラングで使用されている最初の 64 個のオブジェクトのみが追加されます (アニメーションテーブルの最大サイズ)。
- 同じオブジェクトがラングで複数回使用されている場合は、最初のオブジェクトのみがアニメーションテーブルに追加されます。

アニメーションテーブルプロパティ

アニメーションテーブルのプロパティを次の表に示します。

プロパティ	編集可	値	詳細
使用	いいえ	True/False	オブジェクトが現在プログラムで使用されているかを示します。
トレース	可 (1)	True/False	トレースウィンドウ (200 ページ参照) でトレースするオブジェクトを選択します。
アドレス	いいえ	オブジェクトアドレス	オブジェクトのアドレスを表示します。
シンボル	いいえ	有効なシンボル	オブジェクトで使用されているシンボルの名前 (定義されている場合)。
値	可 (2)	現在値	オブジェクトの値。 オブジェクトに読み取り / 書き込みアクセスがありオンラインモード (26 ページ参照) の場合、必要に応じてダブルクリックし、新しいオブジェクトの値を入力します。ロジックコントローラーで実行されているプログラムで、オブジェクトの値がリアルタイムに更新されます。 詳細は、リアルタイムの値を変更する (202 ページ参照) を参照してください。
強制	可 (2)	0 に強制 1 に強制 強制なし	デジタル入力、デジタル出力のみの場合にのみ表示されます。オンラインモード (26 ページ参照) の場合にのみ編集可。必要に応じて入力または出力の値を 0 または 1 に強制できます。 強制なし を選択すると適用されている強制が解除されます 注記: デジタル入力または出力の値を強制してもユーザーロジックには影響しません。例えば、1 に 強制 を選択して、デジタル出力をユーザープログラムで使用する場合、ユーザープログラムの値は影響を受けません。
コメント	いいえ	有効なコメント	オブジェクトで使用されているコメント (定義されている場合)。
(1) 8 個までオブジェクトを選択できます。			
(2) オブジェクトタイプ、オンラインモードかどうかにかかわらず依存します。			

アニメーションテーブルで項目を設定する

アニメーションテーブルでオブジェクトの検索または置換するには、オブジェクトを右クリックし**検索と置換**を選択します。詳細は、**検索と置換 (143 ページ参照)** を参照してください。

アニメーションテーブルからオブジェクトを削除するには、オブジェクトを右クリックし**アニメーションテーブルから削除**を選択します。

既存のアニメーションテーブルのコピー、切り取り、貼り付け

手順	手順内容
1	プログラミングウィンドウの左側にある ツールタブ を選択します。
2	CTRL キーを押しながら 1 つ以上のアニメーションテーブルを アニメーションテーブル から選択します。
3	アニメーションテーブル で選択したアニメーションテーブルの 1 つを右クリックし、 アニメーションテーブルのコピー または アニメーションテーブルのカット を選択します。
4	アニメーションテーブルを貼り付けるには、次のいずれかを行います。 <ul style="list-style-type: none">● アニメーションテーブルを右クリックし、アニメーションテーブルの貼り付けを選択します。● 既存のアニメーションテーブルを右クリックし、アニメーションテーブルの貼り付けを選択します。 <p>結果: 確認ウィンドウが表示されます。シンボルとコメントを保持するには、チェックボックスを外して、OK をクリックします。</p> <p>結果: 1 つ以上のアニメーションテーブルが、アニメーションテーブルの終わりまたは選択したアニメーションテーブルの後に追加されます。</p> <p>アニメーションテーブルをコピー / 貼り付けをする際、SoMachine Basic が自動的に新しい名前を割り当てます。例: Animation_table_2 は、Animation_table_2_0 になります。</p>

ファクションレベル (77 ページ参照) の低いプロジェクトにアニメーションテーブルを貼り付けると、そのファクションレベルでサポートされているオブジェクト設定だけがコピーされます。

貼り付けられたアニメーションテーブルに含まれるシンボルがすでにプロジェクトで使用されている場合は、SoMachine Basic によって貼り付けられたシンボルが置き換えられます。

アニメーションテーブルの削除

手順	手順内容
1	ツールウィンドウの アニメーションテーブル エリアで、名前を変更するアニメーションテーブルを右クリックし アニメーションテーブルの削除 を選択します。

アニメーションテーブル名の変更

手順	手順内容
1	ツールウィンドウの アニメーションテーブル エリアで、名前を変更するアニメーションテーブルを右クリックし アニメーションテーブル名の変更 を選択します。
2	アニメーションテーブルの新しい名前を入力し Enter キーを押します。

アニメーションテーブルのエクスポート

手順	手順内容
1	プログラミングウィンドウの左側にあるツールタブを選択します。
2	アニメーションテーブルを選択するには、次のいずれかを行います。 <ul style="list-style-type: none"> ● アニメーションテーブルを右クリックします。 ● CTRL キーを押しながら 1 つ以上のアニメーションテーブルを選択し、右クリックします。
3	アニメーションテーブルのエクスポートをクリックします。
4	フォルダーを選択し、アニメーションテーブル (.smbf) を保存します。

アニメーションテーブルのインポート

手順	手順内容
1	プログラミングウィンドウの左側にあるツールタブを選択します。
2	アニメーションテーブルを選択するには、次のいずれかを行います。 <ul style="list-style-type: none"> ● アニメーションテーブルを右クリックします。 ● 既存のアニメーションテーブルを右クリックします。
3	アニメーションテーブルのインポートをクリックします。
4	アニメーションテーブルファイル (* .smbf) を含むフォルダーに移動します。
5	アニメーションテーブルをダブルクリックします。 結果：アニメーションテーブルがアニメーションテーブルの終わりまたは選択したアニメーションテーブルの前に追加されます。

インポートされたアニメーションテーブルに含まれるシンボルがすでにプロジェクトで使用されている場合は、SoMachine Basic によってインポートされたシンボルが置き換えられます。

トレースウィンドウを開く

手順	手順内容
1	アニメーションテーブルのトレース列で、8 個までのオブジェクトを選択します。
2	ロジックコントローラーに接続 (215 ページ参照) するかシミュレーターを起動 (232 ページ参照) します。
3	時間ベースリストから値を選択します。選択された値がトレースウィンドウ (200 ページ参照) の更新頻度 (秒) になります。
4	トレースをクリックします。 トレースウィンドウが表示されます。

メモリーオブジェクト

概要

メモリーオブジェクトは以下を含みます。

- ビットメモリー
- ワードメモリー
- ワード型定数

メモリー割り当てモードの選択

メモリーオブジェクトのプロパティの表示または更新の前にメモリー割り当てモード (69 ページ参照) を使用できるように選択します。

ビットメモリーのプロパティ

ビットメモリーのプロパティを次の表に示します。

プロパティ	編集可	値	初期値	詳細
使用	いいえ	True/False	False	ビットメモリーが現在プログラムで使用されているかを示します。
アドレス	いいえ	ビットオブジェクトを参照してください。	N/A	ビットメモリーのアドレスを表示します。x はロジックコントローラーで対応しているビットメモリーの番号です。
シンボル	可	有効なシンボル	なし	シンボルをビットメモリーに関連付けることができます。
値	可	ビットオブジェクトを参照	0	ビットメモリーの値
コメント	可	有効なコメント	なし	コメントをビットメモリーに関連付けることができます。

ワードメモリーのプロパティ

ワードメモリー プロパティ %MW %MD %MF

最初にプロパティを表示するワードメモリーのタイプを選択します。

- %MW ワードメモリー
- %MD ダブルワード
- %MF 浮動小数点ワード

ワードメモリーのプロパティを次の表に示します。

プロパティ	編集可	値	初期値	詳細
使用	いいえ	True/False	False	ワードメモリーが現在プログラムで使用されているかを示します。
重複使用	いいえ	True/False	False	重複使用。ワードメモリーのメモリー領域の一部が現在使われているかを示します。オブジェクト間の重複の可能性 (<i>SoMachine Basic</i> , <i>Generic Functions Library Guide</i> 参照) を参照してください。
アドレス	いいえ	ワードオブジェクト (<i>SoMachine Basic</i> , <i>Generic Functions Library Guide</i> 参照) を参照	N/A	ワードメモリーのアドレスを表示します。

プロパティ	編集可	値	初期値	詳細
シンボル	可	有効なシンボル	なし	シンボルをワードメモリーに関連付けることができます。
値	可	ワードオブジェクト (<i>SoMachine Basic, Generic Functions Library Guide</i> 参照) を参照	0	ビットメモリーの値
コメント	可	有効なコメント	なし	コメントをワードメモリーに関連付けることができます。

ワード型定数のプロパティ

ワード型定数 プロパティ %KW %KD %KF

初めにプロパティを表示するワード型定数のタイプを選択します。

- %KW ワード型定数
- %KD ダブルワード型定数
- %KF 浮動小数点ワード型定数

ワード型定数のプロパティを次の表に示します。



プロパティ	編集可	値	初期値	詳細
使用	いいえ	True/False	False	ワード型定数が現在プログラムで使用されているかを示します。
重複使用	いいえ	True/False	False	重複使用。ワード型定数のメモリー領域の一部が現在使われているかを示します。オブジェクト間の重複の可能性 (<i>SoMachine Basic, Generic Functions Library Guide</i> 参照) を参照してください。
アドレス	いいえ	ワードオブジェクト (<i>SoMachine Basic, Generic Functions Library Guide</i> 参照) を参照	N/A	ワード型定数のアドレスを表示します。
シンボル	可	有効なシンボル	なし	シンボルをワード型定数に関連付けることができます。
10 進数	可	値の 10 進数表示ワードオブジェクト (<i>SoMachine Basic, Generic Functions Library Guide</i> 参照) を参照	0	ワード型定数の 10 進数
2 進数	可	値の 2 進数表示ワードオブジェクト (<i>SoMachine Basic, Generic Functions Library Guide</i> 参照) を参照	2#0000000000000000	ワード型定数の 2 進数
16 進数	可	値の 16 進数表示ワードオブジェクト (<i>SoMachine Basic, Generic Functions Library Guide</i> 参照) を参照	16#0000	ワード型定数の 16 進数

プロパティ	編集可	値	初期値	詳細
ASCII	可	値の ASCII 表示ワードオブジェクト (<i>SoMachine Basic, Generic Functions Library Guide</i> 参照) を参照	無意味	ワード型定数の ASCII
コメント	可	有効なコメント	なし	コメントをワード型定数に関連付けることができます。



ワード型定数プロパティのエクスポート/インポート

オフラインまたはオンラインモードで、CSV ファイルにエクスポートできます。また、アドレス、シンボル、値およびコメントのプロパティをインポートできます。

ワード型定数のプロパティのエクスポート

手順	手順内容
1	プログラミングウィンドウの左側にあるツールタブを選択します。
2	メモリーオブジェクト → ワード型定数をクリックします。
3	ワード型定数のプロパティの エクスポート をクリックします。 結果: 定数をエクスポートウィンドウが表示されます。
4	定数をエクスポートウィンドウで、次の手順に従います。 1. エクスポートタイプ を選択。 2.  をクリックしてファイルパスを選択します。 3. 次に示す事項を入力します。 ○ ファイル名 、 ○ 最初のインデックス (数値)、 ○ 最後のインデックス (数値)、 最初のインデックスは、最後のインデックス以下に設定してください。
5	エクスポートパラメーターを変更するには、  エクスポートオプション をクリックします。 1. ヘッダーの名前を表示する場合は、 ヘッダー を選択します。 2. セミコロン または カンマ を区切り記号として選択します。
6	エクスポート をクリックします。

ワード型定数のプロパティのインポート

手順	手順内容
1	プログラミングウィンドウの左側にあるツールタブを選択します。
2	メモリーオブジェクト → ワード型定数をクリックします。
3	ワード型定数のプロパティの インポート をクリックします。 結果: 定数のインポートウィンドウが表示されます。
4	 をクリックし、ファイル (*.csv) を含むフォルダーに移動し、ファイルをダブルクリックします。
5	インポートパラメーターを変更するには、  インポートオプション をクリックし、.csv ファイルで使用する区切り記号 セミコロン または コンマ を選択します。
6	インポート をクリックします。

値が重複している場合は、最後の重複値がインポートされます。

システムオブジェクト

概要

システムオブジェクトはロジックコントローラー固有です。詳細は、ロジックコントローラーのプログラミングガイドを参照してください。

I/O オブジェクト

概要

以下のオブジェクトタイプはハードウェア固有であり、使われているロジックコントローラーに依存します。

- デジタル入力、および出力
- アナログ入力、および出力
- 高速カウンター (FC)、高速カウンター (HSC)、パルス出力のような高度ファンクションブロック

詳細は、ロジックコントローラーの *プログラミングガイド*、および *Advanced Functions Library Guide* を参照してください。

ネットワークオブジェクト

情報

ネットワークオブジェクトは、EtherNet/IP、Modbus TCP、または Modbus Serial IOMaster を介した通信に使われます。

EtherNet/IP 通信には、2 種類のネットワークオブジェクトがあります。

- %QWE: 入力アセンブリ
- %IWE: 出力アセンブリ

Modbus TCP 通信には、2 種類のネットワークオブジェクトがあります。

- %QWM: 入力レジスター
- %IWM: 出力レジスター

Modbus Serial IOMaster には以下のネットワークオブジェクトが使われます。

- %IN: デジタル入力 (IOMaster)
- %QN: デジタル出力 (IOMaster)
- %IWN: 入力レジスター (IOMaster)
- %QWN: 出力レジスター (IOMaster)
- %IWNS: IOMaster ネットワーク診断コード

注記: 入力と出力の参照は、EtherNet/IP マスター、または Modbus TCP クライアントの視点から見たものです。

ネットワークオブジェクトの設定の詳細は、ロジックコントローラーのプログラミングガイドを参照してください。

ソフトウェアオブジェクト

概要

SoMachine Basic は以下の汎用ソフトウェアオブジェクトに対応しています。

オブジェクト	説明
タイマー	イベントのトリガー等の動作までの時間を指定します。
カウンター	イベントのアップカウント、およびダウンカウントをします。
メッセージ	外部デバイスとの通信ができます。
LIFO/FIFO レジスター	LIFO、または FIFO モードでそれぞれ 16 ビットのワードを 16 個まで格納できるメモリーブロック。
ドラム	外部イベントに応じてステップが変わる電気機械式ドラムコントローラーに類似した原理で動作します。各ステップにおいて、カムの高点でロジックコントローラーにより実行されるコマンドが与えられます。
ビットシフトレジスター	バイナリーデータビット (0、または 1) を右、または左にシフトします。
ステップカウンター	処理が割り当てられる一連のステップを提供します。
スケジュールブロック	定義済みの月、日、時間での動作を管理します。
RTC	RTC から日付と時間の読み込み、またはロジックコントローラーの RTC をユーザー定義の日付と時間で更新します。
PID	PID (比例積分微分アルゴリズム) 制御を可能にします。
データロギング	オブジェクトや文字列のデータを永久的に保存します。
グラフセステップ	シンボルやコメントを追加、または編集するためにグラフセビットアドレス (%Xi) 変数を一覧にします。

これらのファンクションブロックの詳細は、SoMachine Basic Generic Functions Library Guide (SoMachine Basic, Generic Functions Library Guide 参照) を参照してください。

メモリー割り当てモードの選択

ソフトウェアオブジェクトのプロパティの表示、または更新の前に、メモリー割り当てモード (69 ページ参照) を使用できるように選択します。

PTO オブジェクト

概要

PTO オブジェクトは、PTO ファンクションのプログラミングに使用するファンクションブロックを提供します。PTO ファンクションブロックは、以下に分類されます。

- モーションタスクテーブル
順序付けられたシーケンスで各 PTO 動作を設定すること、および推定される動作の全体的なプロファイルを視覚化することができます。
- モーション
これらのファンクションブロックは、軸の動作を制御します。例：軸への電源、軸の動作等。
- 管理
これらのファンクションブロックは、軸の動作のステータス、および診断を制御します。例：実際の速度のステータスと値、実際の位置、エラーを検出した軸制御等。

PTO ファンクションブロックの詳細は、コントローラーの *Advanced Function Library Guide* を参照してください。

ドライブオブジェクト

概要

ドライブオブジェクトは、Modbus Serial IOScanner または Modbus Serial IOScanner 上に設定された ATV ドライブおよび他のデバイスを制御します。

Advanced Functions Library Guide を参照してください。

通信オブジェクト

概要

通信オブジェクトは、Modbus デバイスとの通信、文字モード (ASCII) メッセージの送受信、および SMS メッセージの送受信に使用されます。


詳細は、[通信オブジェクト](#) を参照してください。

検索と置換

概要

検索と置換を使用して、プログラムで使用されているすべてのオブジェクトを検索し、また別のオブジェクトと置き換えることができます。

アイテムの検索と置換

手順	手順内容																		
1	<p>プログラミングウィンドウの左側にあるツールタブを選択します。</p> <p>SoMachine Basic では、色々な場所から検索と置換機能呼び出すことができます。アニメーションテーブルのアイテム (128 ページ参照) を右クリックし、検索と置換を選択するのが一例です。</p>																		
2	<p>以下に、検索と置換ウィンドウを表示する方法を示します。</p> <ul style="list-style-type: none">● プログラミングウィンドウの ツールタブにある検索と置換をクリックします。● ラング、またはラングで選択されているアイテムを右クリックし、コンテキストメニューから検索と置換を選択します。● 任意のオブジェクトのプロパティウィンドウの 1 行を右クリックし、コンテキストメニューから検索と置換を選択します。 <p>検索と置換ウィンドウ</p>  <p>検索と置換</p> <p>検索 <input type="text" value="%Q0.0"/> <input type="button" value="検索"/></p> <p>置換 <input type="text"/> <input type="button" value="置換"/></p> <p>検索対象 <input checked="" type="checkbox"/> プログラム <input checked="" type="checkbox"/> コメント</p> <p>置き換えオプション <input checked="" type="radio"/> すべて <input type="radio"/> 選択結果</p>																		
3	<p>検索ボックスにオブジェクト名、またはシンボル名を入力します。ラングで選択されたアイテム、またはオブジェクトのプロパティウィンドウで選択されたアイテムを右クリックして検索と置換ウィンドウを開いた場合、検索フィールドはあらかじめ入力されています。</p> <p>以下のワイルドカードを使用できます。</p> <ul style="list-style-type: none">● アスタリスク (*)。検索したい文字の 0 文字、またはそれ以上の文字を置き換えて検索します。%MW1* と入力すると %MW1 と %MW101 の両方を検出します。● 疑問符 (?)。検索したい文字の 1 文字だけを置き換えて検索します。COIL?2 と入力すると COIL12 を検出し、COIL012 を検出しません。																		
4	必要に応じて、 置換 ボックスに置き換えたいオブジェクト名、またはシンボル名を入力します。																		
5	<p>プログラムを選択して、プログラムのソースコード内の検索をします。</p> <p>コメントを選択して、プログラムコメントの検索をします。</p>																		
6	<p>検索、または置換をクリックします。Enter キーを押して検索を始めることもできます。</p> <p>置換ボタンは、置換ボックスに置き換えるオブジェクト名、またはシンボル名が入力された場合のみに有効です。</p> <p>検出されたアイテムは、結果リストに表示されます。</p> <p>結果</p> <table border="1"><thead><tr><th></th><th>POU</th><th>ラング</th><th>オブジェクト</th><th>コード</th><th>シンボル</th></tr></thead><tbody><tr><td></td><td>1 - 新規 POU</td><td>Rung0</td><td></td><td>%Q0.0</td><td></td></tr><tr><td></td><td>1 - 新規 POU</td><td>Rung1</td><td></td><td>LD %Q0.0</td><td>LD %Q0.0</td></tr></tbody></table>		POU	ラング	オブジェクト	コード	シンボル		1 - 新規 POU	Rung0		%Q0.0			1 - 新規 POU	Rung1		LD %Q0.0	LD %Q0.0
	POU	ラング	オブジェクト	コード	シンボル														
	1 - 新規 POU	Rung0		%Q0.0															
	1 - 新規 POU	Rung1		LD %Q0.0	LD %Q0.0														

手順	手順内容									
7	<p>必要に応じてシンボルを表示を選択し、オブジェクトに定義されたシンボルを代わりに表示します。</p> <p>Results <input checked="" type="checkbox"/> Show symbols</p> <table border="1" data-bbox="323 305 1149 413"> <thead> <tr> <th>POU</th> <th>Rung</th> <th>Code</th> </tr> </thead> <tbody> <tr> <td>POU_0</td> <td>Rung_0</td> <td>OUTPUT</td> </tr> <tr> <td>POU_0</td> <td>Rung_1</td> <td>LD OUTPUT</td> </tr> </tbody> </table>	POU	Rung	Code	POU_0	Rung_0	OUTPUT	POU_0	Rung_1	LD OUTPUT
POU	Rung	Code								
POU_0	Rung_0	OUTPUT								
POU_0	Rung_1	LD OUTPUT								
8	リストされた任意の結果をクリックすると、プログラムのコード行に直接ジャンプできます。									

クロスリファレンス

概要

クロスリファレンスビューでは、POU に含まれるプログラムを表示できます。オブジェクトが同じ POU の別のオブジェクトに依存する場合、対応するラングが表示されます。

クロスリファレンスビューは、オフラインモードとオンラインモードの両方で使用できます。

クロスリファレンスビューの表示

クロスリファレンスビューを表示するには、**プログラミング** → **ツール** → **クロスリファレンス** をクリックし、アクションゾーン内の 1 つ以上のオブジェクトを選択します。

クロスリファレンスビュー

Cross Reference			
POU	Rung	Code view	Symbol view
ElementName - %MW4 (1 items)			
1 - My POU	Rung0	[%MW4 := 5]	[%MW4 := 5]

次の表にクロスリファレンスビューの要素を示します。

要素	詳細
POU	オブジェクトを含む POU の名前。
ラング	オブジェクトを含むラングの名前。
コードビュー	オブジェクトのコード。
シンボルビュー	オブジェクトのシンボル。

シンボルリスト

概要

プログラム内のオブジェクトに関連付けられたすべてのシンボルのリストを表示することができます。システムビット (%S) とシステムワード (%SW) に自動的に関連付けられているシンボルを除いて、シンボルを含むすべてのオブジェクトが表示されます。システムオブジェクトプロパティを使用、または独自のシンボルリスト (下記参照) をインポートして、システムビット (%S)、およびシステムワード (%SW) のシンボルやコメントを上書きできます。上書きされたシンボルがシンボルリストに表示されます。

シンボルの定義と使用 (67 ページ参照) では、シンボルを作成してプログラムで使用方法について説明します。

シンボルリストの表示

手順	手順内容
1	プログラミングウィンドウの左側にあるツールタブを選択します。
2	<p>シンボルリストをクリックします。 結果: シンボルリストウィンドウが表示されます。各アイテムの情報は以下のように表示されます。</p> <ul style="list-style-type: none"> ● 使用: シンボルが現在プログラムで使用されているかどうかを示します。 ● アドレス: シンボルが使用されているオブジェクトのアドレス。 ● シンボル: シンボル名。 ● コメント: オブジェクトで使用されているコメント (定義されている場合)。 ● シンボルを割り当て: SoMachine Basic は、使用された各オブジェクトにシンボルを自動的に割り当てます。ただし、次に挙げるタイプのシンボル以外です。%M, %MW, %MD, %MF, %S, %SW, %KW, %KD, %I, %IW, %Q, %QW ● デフォルトシンボルの削除: デフォルトシンボルの削除 ● インポート: シンボルのインポート ● エクスポート: シンボルのエクスポート

デフォルトシンボルの作成

メモリーオブジェクトのデフォルトシンボルを作成するには:

1. シンボルを割り当てをクリックします。

結果: デフォルトシンボルは、プログラムで使用されているすべてのメモリーオブジェクト (シンボルがすでに定義されていないもの) に割り当てられます。(%M, %MW, %MD, %MF, %S, %SW, %KW, %KD, %KF, %I, %IW, %Q, %QW)。

シンボルは次のように名前が付けられます。symbolname = objectname_i の、objectname は、% のないオブジェクトタイプで、i はオブジェクトのインデックスです。

例: 次のオブジェクトはプログラムで使用されますが、シンボルは定義されていません。

オブジェクト	割り当てられたシンボル
%MW0	MW_0
%MW2	MW_2
%M0	M_0

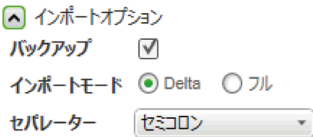

デフォルトシンボルの削除

デフォルトシンボルの削除：

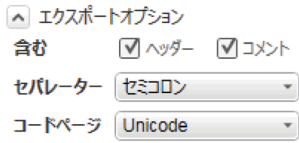
手順	手順内容
1	デフォルトシンボルの削除をクリックします。
2	表示される確認ウィンドウではいをクリックします。 結果：割り当てられたすべてのデフォルトシンボルが削除されます。

注記：デフォルトシンボルが割り当てられているオブジェクトがプログラムで使用されなくなった場合、デフォルトシンボルは保持されます。

シンボルのインポート

手順	手順内容
1	インポートボタンをクリックするか、シンボルリストで右クリックをしてシンボルのインポートを選択します。 結果：シンボルのインポートウィンドウが表示されます。
2	インポートするシンボルを含むカンマ区切り (CSV) ファイルのファイルパスを選択します。
3	必要に応じて、インポートオプションをクリックし、インポートするシンボルのフォーマット設定をします。 
4	インポートをクリックします。 結果：シンボルリストウィンドウに選択した CSV ファイルにあるすべてのシンボルが、設定したフォーマットで表示されます。 インポート中にエラーが検出された場合は、エラーリストのレポートが表示されます。 
5	保存をクリックしてテキストファイル (.txt) にレポートを出力します。

シンボルリストのエクスポート

手順	手順内容
1	エクスポートボタンをクリックするか、シンボルリストで右クリックをして シンボルのエクスポート を選択します。 シンボルのエクスポートウィンドウが表示されます。
2	ファイルパスと作成するカンマ区切り (CSV) ファイルの ファイル名 を選択します。
3	必要に応じて、 エクスポートオプション をクリックし、エクスポートするシンボルのフォーマット設定をします。 
4	エクスポート をクリックします。 結果 : CSV ファイルが、指定したフォーマットで作成されます。

SoMachine Basic プロジェクトと Vijeo-Designer プロジェクト間でシンボルを共有する

シンボルを Vijeo-Designer プロジェクトと共有する前に、共有したいシンボルはすべて SoMachine Basic プロジェクトで定義されていることを確認します。定義されていない場合は、SoMachine Basic でプロジェクトを作成、または開いてシンボル名を定義しプロジェクトを保存します。プロジェクト内のすべてのメモリーオブジェクトに Vijeo-Designer のデフォルトシンボルを作成できます。詳細は、デフォルトシンボルの作成 (146 ページ参照) を参照してください。

次の手順で、SoMachine Basic のシンボルを Vijeo-Designer プロジェクトと共有します。

手順	手順内容
1	Vijeo-Designer を起動します。
2	Vijeo-Designer でプロジェクトを作成、または開きます。
3	Navigator ウィンドウの Project タブをクリック、 IO Manager を右クリックし、 New Driver... Insert を選択します。 結果 : New Driver ウィンドウが表示されます。
4	Driver リストからドライバーを選択、 Equipment リストから機器を選択し、 OK をクリックします。 例: <ul style="list-style-type: none"> ● ドライバー: Modbus TCP/IP ● 機器: Modbus Equipment 結果 : Equipment Configuration ウィンドウが表示されます。
5	各パラメータの詳細を入力し OK をクリックします。例えば、 IP Address 、 Unit ID 、 IP Protocol などを入力します。 結果 : コントローラーとの通信を開始するための新しいドライバーが作成されます。 Navigator ウィンドウの Project タブにある IO Manager ノードに、選択したドライバーと機器が表示されます。
6	Vijeo-Designer のメニューバーで、 Variable → Link Variables をクリックします。 結果 : Link Variables ウィンドウが表示されます。
7	Files of type のフィルターで SoMachine Basic プロジェクトファイル (*.SMBP) に絞込み、 Equipment のフィルターで通信用に作成したドライバーに絞り込みます。

手順	手順内容
8	シンボルを定義した SoMachine Basic プロジェクトを選択し、 Open をクリックします。 結果 ：すべてのシンボルは自動的にプロジェクトから抽出され、作成したドライバーにリンクされます。
9	使用する変数を選択し、HMI アプリケーションに追加します。 結果 ：シンボルと同じ名前の変数が、変数リストに追加されます。変数リストは、 Navigator ウィンドウの Project タブにある Variables ノードに表示されます。

注記：既にシンボルを Vijeo-Designer のプロジェクトと共有した後に SoMachine Basic のプロジェクトで既存のシンボルを変更、または新しいシンボルを追加した場合は、Vijeo-Designer のプロジェクトでそのシンボルを更新します。

シンボルを Vijeo-Designer のプロジェクトで更新するには、まず新しいシンボルを定義、または既存のシンボルを編集し SoMachine Basic のプロジェクトを保存します。次の手順で、Vijeo-Designer のプロジェクトを開きます。

手順	手順内容
1	Navigator ウィンドウの Project タブで、 Variables を右クリックし Update Link を選択します。 結果 ：機器のドライバーと既存のシンボルが更新されます。
2	Variables を再度右クリックし、 New Variables from Equipment を選択、SoMachine Basic のプロジェクトで作成した新しい変数を選択します。 結果 ：新しい変数が SoMachine Basic のプロジェクトから、変数リストへ追加されます。追加された変数は、 Navigator ウィンドウの Project タブにある Variables ノードに表示されます。

メモリー消費量の表示

概要

アプリケーション、プログラム、およびユーザーデータで消費されているコントローラーのメモリー情報を表示することができます。

メモリー消費量の表示

この機能を使用するには、プログラムをコンパイルしエラーがない状態にします。メッセージウィンドウ (126 ページ参照) でプログラムステータスを確認します。


以下に、メモリー消費量の表示を開く手順を示します。

手順	手順内容
1	プログラミングウィンドウの左側にあるツールタブを選択します。
2	メモリー消費量をクリックします。 メモリー消費量のウィンドウが表示されます。

メモリー消費量の表示について

注記： この表示は、コンパイルがエラーなしで完了している場合のみに可能です。

メモリー消費量の表示のフィールドを次の表に示します。

フィールド	説明
最後のコンパイル	プログラムを最後にコンパイルした日時。 注記： この値は以下のタイミングで更新されます。 <ul style="list-style-type: none"> ● ツールバーのコンパイルボタン  をクリックしたとき ● コントローラーへログインしたとき ● プログラムのアップロードを開始したとき ● プログラムの修正をオンラインのコントローラーへ送信したとき ● シミュレーションを開始したとき

プログラム行	
フィールド	説明
使用	プログラムで使用されているコード行数。
残り	プログラムで使用可能な最高行数から使用されている行数をマイナスした数。
注記： プログラミングタブの、使用されているプログラム行数とラング内の IL コード行数の間に直接のリンクはありません。例えば、2 行の IL コードは 6 行のプログラムを生成する可能性があります。	

キャッシュメモリー	
フィールド	説明
周期、およびイベントタスク	周期、およびイベントタスクで占有されているキャッシュメモリー (バイト)。
システム用に確保	システム用に確保されたキャッシュメモリー (バイト)。
メモリー残量	プログラムに使用できるキャッシュメモリー (バイト)。

RAM メモリー	
フィールド	説明
マスタータスクとサブルーチン	プログラムのマスタータスクとすべてのサブルーチンで占有されている RAM メモリー (バイト)。
設定	ロジックコントローラーと拡張モジュールのハードウェア設定情報に使用されている RAM メモリー (バイト)。
メモリーオブジェクト	アプリケーションで使われているメモリーオブジェクト (ビットメモリー、ワードメモリー、およびワード型定数) で占有されている RAM メモリー (バイト)。
表示	リモートグラフィック表示 アプリケーションのサイズ (バイト)。ロジックコントローラーが リモートグラフィック表示 をサポートしていない場合は、0。
メモリー残量	プログラムに使用できる RAM メモリー (バイト)。

非プログラムデータ	
フィールド	説明
使用	プロジェクトプロパティ、シンボル、コメント、およびアニメーションテーブルで使われているメモリー。
残り	非プログラムデータに使用可能なメモリー。

6.12

ラダー言語のプログラミング

このセクションについて



このセクションには次の項目が含まれています。

項目	参照ページ
ラダー図の概要	153
ラダー図プログラミングの原則について	155
ラングのカラーコード	157
ラダープログラムのグラフィック要素	158
比較ブロック	163
演算ブロック	164
コメントの追加	167
プログラミングの推奨方法	168

ラダー図の概要

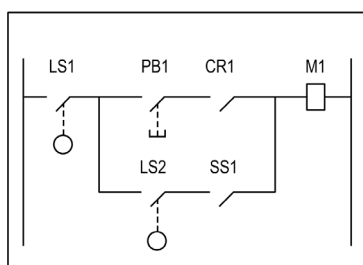
はじめに

ラダー図は、リレー制御回路を表すリレー論理図に似ています。ラダー図とリレー論理図の主な違いは、ラダー図のプログラミングにしかない以下の機能です。

- すべての入力と 2 進論理ビットは接点シンボル () で表されます。
- すべての出力と 2 進論理ビットはコイルシンボル () で表されます。
- 数値演算は、図によるラダーインスタレーションに含まれます。

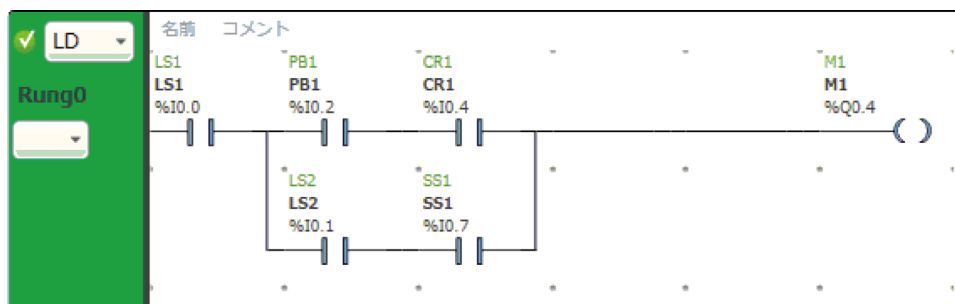
リレー回路に相当するラダー図

以下に、簡略化されたリレー論理回路の配線図を示します。



Relay logic circuit

相当するラダー図



上の図では、リレー論理図のスイッチングデバイスで使用されているすべての入力が、ラダー図に接点として示されています。リレー論理図の M1 出力コイルは、ラダー図では出力コイルシンボルで示されています。ラダー図で接点やコイルの上部にあるアドレス番号は、ロジックコントローラーへの外部入出力接続の位置を示します。

ラダー図のラング

ラダー図言語のプログラムは、2 本の垂直線の間にかかれた図によるインスタレーションからなるラングで構成されています。ラングはロジックコントローラーによってシーケンシャルに実行されます。

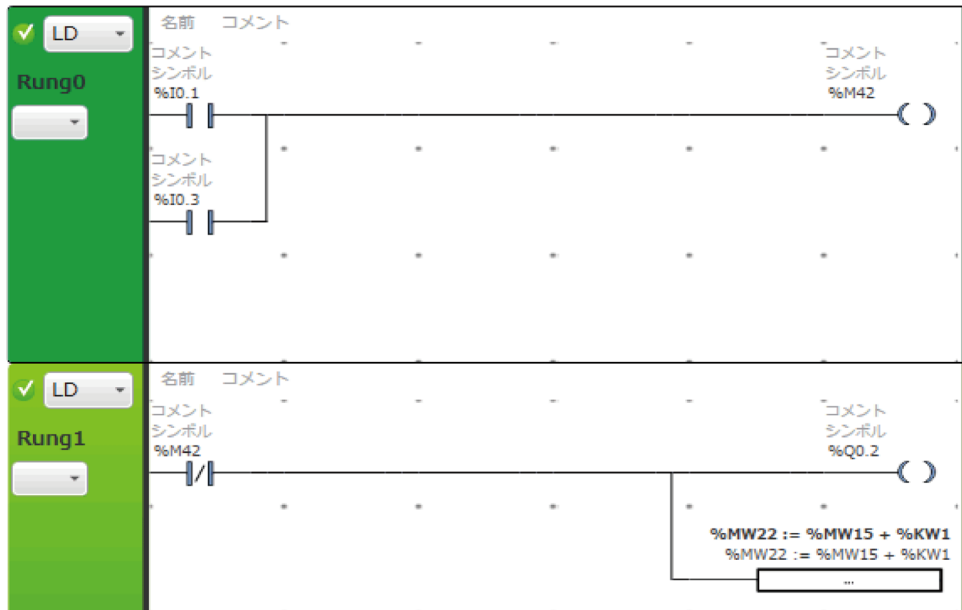
以下に、図によるインスタレーションの機能を示します。

- コントローラーの入力 / 出力 (プッシュボタン、センサー、リレー、パイロットランプ)
- コントローラーの機能 (タイマー、カウンターなど)
- 数学と論理演算 (加算、除算、AND、XOR など)
- 比較演算子、およびその他の数値演算 ($A < B$ 、 $A = B$ 、シフト、回転など)
- コントローラーの内部変数 (ビット、ワードなど)

これらの図によるインストラクションは、縦横に接続され最終的にひとつ、または複数の出力や動作をもたらします。ラングはインストラクションをリンクしたグループを複数対応できません。

ラダー図のラングの例

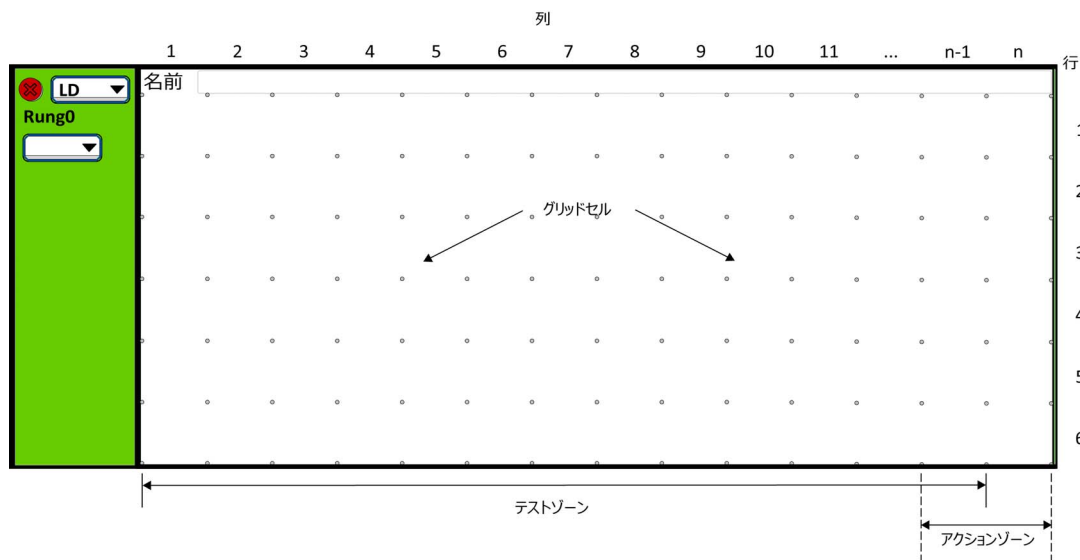
以下に、2つのラングで構成されたラダー図のプログラム例を示します。



ラダー図プログラミングの原則について

プログラミンググリッド

以下の図のように、各ラダーラングは最高 255 行、11...30 列からなり、2 つのゾーンに分かれています。



n 設定された列数 (11...30)。列数の設定についての詳細は、ラダーエディターのカスタマイズ (49 ページ参照) を参照してください。

グリッドセル

セルを使って図の要素をグリッド内に配置します。グリッド内の各セルは、4 ドットで仕切られています。

グリッドゾーン

ラダー図のプログラミンググリッドは、2 つのゾーン (初期値) に分かれています。

- テストゾーン
アクションを実行するためにテストされる条件を含みます。列 1 から n-1 (n は設定された列数) からなり、接点、ファンクションブロック、比較ブロックを含みます。
- アクションゾーン
テストゾーンでテストされた条件の結果による出力または結果によって実行されるオペレーションを含みます。列 -1 から n-1 (n は設定された列数) からなり、コイルおよびオペレーションブロックを含みます。

ラダーエディターのカスタマイズ



Ladder エディターの上にある以下のオブジェクトを使用してエディターのコンテンツをカスタマイズします。

オブジェクト	詳細
IL>LD	すべてのラングの表示を IL から Ladder に切り替えます。
LD>IL	すべてのラングの表示を Ladder から IL に切り替えます。
-	ラダーグリッドから 1 列削除します。列数が最小数 (11) になるとボタンは無効になります。
+	ラダーグリッドに 1 列追加します。列数が最大数 (30) になるとボタンは無効になります。
コメントの表示 / 非表示	クリックしてラングのメッセージの表示 / 非表示を切り替えます T を解除すると、コメントは 2 行にわたり表示されます。
T	ラングのシンボルの表示 / 非表示を切り替える際にクリックします。コメントの表示 / 非表示を解除すると、シンボルは 2 行にわたり表示されます。
DEC/HEX	オンラインモードでのみ表示されます。クリックしてラングの数値を 10 進数または 16 進数の表示に切り替えます。
1 - 新規 POU	ツール → マスタータスクエリアに表示される、デフォルトの POU 名をダブルクリックして編集します。
コメント	ダブルクリックしてテキストを入力し、この POU にコメントを関連付けます。
ズームスライダー	ラダーエディターを拡大 / 縮小します。 ショートカット Ctrl + マウスホイール を使用して拡大または縮小することができます。 プロジェクトを移動させても拡大位置は変わりません。

ラングのカラーコード

オフラインモード

選択したラングは、背景色が濃い緑に表示されます。



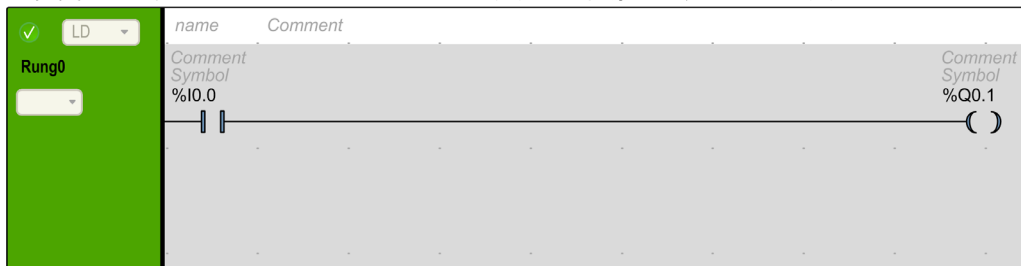
オンラインモード

オンラインモードの場合：

- ラングは背景色が緑で表示されます。
- オンラインモード中に追加、または変更されたラングはオレンジの背景色で表示されます。



- 変更可能な要素がないラングはロックされ、灰色の前景色で表示されます。




ラダープログラムのグラフィック要素

はじめに

ラダー図のインストラクションは、プログラミングワークスペースの上のツールバーからグラフィック要素をグリッドのセルにドラッグアンドドロップして挿入します。




グラフィック要素の挿入

以下に、ラングにグラフィック要素を挿入する手順を示します。

手順	手順内容
1	挿入するグラフィック要素をツールバーでクリックします。グラフィック要素がメニューの場合、グラフィックアイテムがメニューに表示されるので、挿入するメニューアイテムをクリックします。
2	マウスをグラフィック要素を挿入したいラングの位置に移動しクリックします。 メモ：要素によっては、ラングのテストゾーンまたはアクションゾーンにのどちらかに挿入してください。詳細は、各グラフィック要素の説明を参照してください。
3	必要に応じて、ツールバーの【選択モード】グラフィック要素  をクリックして選択をリセットします。



ラング

以下のグラフィック要素を使用してプログラムのラングを管理します。

グラフィック要素	名前	ファンクション
	ラングの作成 (84 ページ参照)	新しい空のラングをプログラムワークスペースで最後のラングの下に挿入します。
	ラングの挿入 (84 ページ参照)	新しい空のラングを選択されているラングのすぐ上に挿入します。
	ラングの削除 (86 ページ参照)	選択されているラングをプログラムから削除します。ラングが空ではない場合、ラングのコンテンツを削除するかどうか確認します。




分岐モード

以下のグラフィック要素を使用してラダー図の分岐を管理します。

グラフィック要素	名前	ファンクション
	正常モード	プログラミング要素 (例えば、接点、コイルなど、ファンクションブロックは除く) をワイヤーラインに沿って配置することができます。
	分岐モード	プログラミング要素 (例えば、接点、コイルなど、ファンクションブロックは除く) をワイヤーラインに沿って配置することができます。


選択とライン

以下のグラフィック要素を使用して、グラフィック要素の選択またはラインの描画を行います。

グラフィック要素	名前	ファンクション
	選択モード	選択モード。
	ラインの描画	2つのグラフィック要素の間にワイヤーラインを描画します。
	行の削除	ワイヤーラインを削除します。

接点


以下のグラフィック要素を使用して、接点 (1列 x 1行) を挿入します。

グラフィック要素	名前	IL	ファンクション
	A 接点	LD	制御ビットオブジェクトのステートが1のときに接点を通過させます。
	B 接点	LDN	制御ビットオブジェクトのステートが0のときに接点を通過させます。
	立上りを検知する接点	LDR	立上り接点。制御ビットオブジェクトの0から1への変更を検出します。
	立下りを検知する接点	LDF	立下り接点。制御ビットオブジェクトの1から0への変更を検出します。

比較ブロック


比較ブロックはプログラミンググリッドのテストゾーンに配置します。インストラクション全体がテストゾーンに収まれば、テストゾーンのどこにでも表示できます。

比較ブロックのグラフィック要素は、2セル (1列 x 2行) 必要とします。

グラフィック要素	名前	IL	ファンクション
	比較ブロック	任意の有効な比較式	比較ブロックのグラフィックシンボルを使用して、命令リストの比較式 (163 ページ参照) をラダー図のラングに挿入します。比較式は2つのオペランドを比較します。結果が確認されると出力は1に変更します。

ブール演算子


ブール演算子のグラフィック要素は、1セル(1列x1行)必要とします。

グラフィック要素	名前	演算子	ファンクション
	XOR 命令	XOR、XORN、XORR、XORF	XOR 命令は、オペランドと前の命令のブール結果との排他的論理和演算を実行します。 XORN 命令は、オペランドの逆数と前の命令のブール結果との排他的論理和演算を実行します。 XORR 命令は、オペランドの立ち上がり接点と前の命令のブール結果との排他的論理和演算を実行します。 XORF 命令は、オペランドの立下り接点と前の命令のブール結果との排他的論理和演算を実行します。

ファンクション

ファンクションブロックは、必ずラダー図のプログラミンググリッドの最初の行に表示されます。ファンクションブロックの上下にラダーインストラクション、その続きが表示されることはありません。ラダーテストインストラクションはファンクションブロックの左側に、テストインストラクションとアクションインストラクションはファンクションの右側から進みます。

ファンクションブロックのグラフィック要素はテストゾーンにのみ配置でき、2から4行x2列のセルを必要とします。

グラフィック要素	名前	ファンクション
	タイマー、カウンタ、レジスターなど	各ファンクションブロックは、他のグラフィック要素へのリンクを可能にする入力と出力を使用します。 注記： ファンクションブロックの出力同士を接続することはできません(縦短絡)。

コイル

コイルのグラフィック要素はアクションゾーンにのみ配置でき、1セル(1行x1列)必要とします。

グラフィック要素	名前	演算子	ファンクション
	コイル	ST	関連付けられたビットオブジェクトにテストゾーンの結果の値が入ります。
	反転コイル	STN	関連付けられたビットオブジェクトにテストゾーンの結果の否定値が入ります。
	セットコイル	S	テストゾーンの結果が1の場合、関連付けられたビットオブジェクトに1がセットされます。
	リセットコイル	R	テストゾーンの結果が1の場合、関連付けられたビットオブジェクトに0がセットされます。


グラフセ (リスト) 命令

以下のグラフィック要素を使用してラダー図の分岐を管理します。


グラフィック要素	名前	演算子	ファンクション
	グラフセステップ有効化 / 現在のステップの無効化	#	現在のステップを無効にし、オプションでグラフセプログラムの別のステップを有効にします。
	グラフセステップ無効化	#D	現在のステップを無効にし、さらにグラフセプログラムのステップを無効にします。

オペレーションブロック

オペレーションブロックはアクションゾーンに配置され、2 行 x 1 列必要とします。

グラフィック要素	名前	演算子	ファンクション
	演算ブロック	任意の有効なオペレーターまたは割り当て命令	オペレーションブロックのグラフィックシンボルを使用して、命令リストのオペレーションと割り当て命令 (164 ページ参照) をラダー図のラングに挿入します。

その他のラダー項目

その他のラダー項目メニュー  にはその他の命令がまとめられています。

OPEN と SHORT 命令は、ラダープログラムのデバッグおよびトラブルシューティングに使える便利な方法を提供します。これらの特殊命令がラングの連続性を短絡またはオープンすることによって、以下の表に示すように、ラングのロジックを変更します。

END/JUMP のグラフィック要素はアクションゾーンにのみ配置でき、1 セル (1 行 x 1 列) 必要とします。

グラフィック要素	名前	演算子	ファンクション
	立上がり接点	RISING $n^{(1)}$	式の立上がり接点を評価します。
	立下り接点	FALLING $n^{(1)}$	式の立下り接点を評価します。
	論理反転演算	N	オペランドの反転値を渡します。
	OPEN	LD 0 AND 0	ラングの始まり。 ラング内:最後の論理オペレーションの結果に関係なくラダーラングの途中にブレークを入れます。
	SHORT	LD 1 OR 1	ラングの始まり。 ラング内:最後の論理オペレーションの結果に関係なくラダーラングを続行させます。
	プログラムの終了	END	プログラムの終了を定義します。

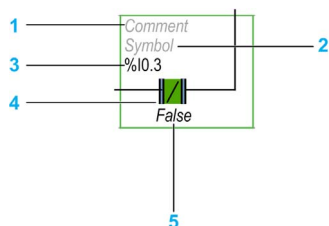
⁽¹⁾ n は、立上がりまたは立下り接点が挿入されるたびにインクリメントされる整数です。

グラフィック要素	名前	演算子	ファンクション
	条件付プログラムの終了	ENDCN	条件付きプログラムの終了を定義します。
	ジャンプまたはサブルーチンの呼び出し	JMP	アップストリームまたはダウンストリームのラベル付きラングに接続します。 注記: IL のプログラミングでは、接続はアップストリームまたはダウンストリームのラベル付きラングにします。
	条件付きジャンプまたはサブルーチンの呼び出し	JMPCN	条件付きで、アップストリームまたはダウンストリームのラベル付きラングに接続します。 注記: IL のプログラミングでは、接続はアップストリームまたはダウンストリームのラベル付きラングにします。
	条件付き要素	IF ELSE ENDIF	式の値に応じて条件付きで一連のステートメントを実行します。
	ループ要素	FOR ENDFOR	一連のステートメントを繰り返します。

(1) *n* は、立上がりまたは立下り接点が挿入されるたびにインクリメントされる整数です。

接点とコイル

接点またはコイルをセルに挿入すると、関連するオブジェクトの情報が表示されます。



凡例	項目	詳細
1	ユーザーコメント	クリックしてコメント (167 ページ参照) を追加します。
2	シンボル	クリックしてシンボル (67 ページ参照) 名を入力し、セル内のオブジェクトと関連付けます。
3	アドレス	クリックしてセル内のオブジェクトのアドレスを入力します。
4	グラフィック要素	グラフィック要素
5	リアルタイム値	オンラインモード (ロジックコントローラーに接続中でプログラム実行中) の場合、セル内のオブジェクトのリアルタイム値を表示します。


比較ブロック

ラダーダイアグラムに IL 比較式を挿入

比較ブロックのグラフィックシンボルを使って、ラダーダイアグラムのラングに IL 比較式を挿入できます。



オペランドには同じ型のオブジェクトを使用します (ワード型同士、フロート型同士など)。以下の手順に沿って進めます。

手順	手順内容
1	 比較ブロック ボタンをクリックします。
2	ラング上をクリックして、比較ブロックを挿入します。
3	比較式ラインを、ダブルクリックします。
4	IL 比較演算を入力し、ENTER を押します。 オンラインモードで式を編集できます。オンライン編集 (207 ページ参照) を参照してください。

注記：

アプリケーションがレベル 6.0 以上のファンクションレベル (77 ページ参照) で構成されている場合。

- 比較ブロックでは、最大 5 つのオペランドと 入れ子の括弧を最大 3 つまで使用できます。
- マスタータスクで複数のオペランドを使用するには、最低 20 のワードメモリー (%MW) が利用できる必要があります。周期タスクでも複数のオペランドを使用する場合は、さらに 20 個のワードメモリーを利用できる必要があります。

注記：マルチオペランド命令はイベントタスクでは使えません。

構文のヘルプを取得

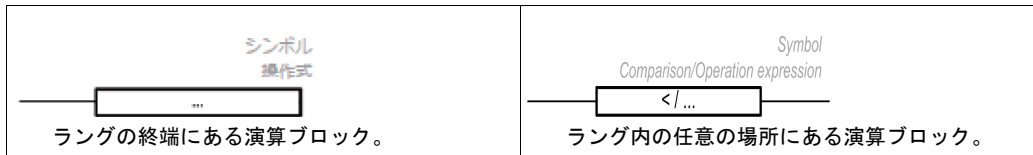
IL 比較演算の構文が正しくない場合、比較式ボックスの縁が、赤色になります。次のいずれかでヘルプを参照してください。

- マウスを比較式ラインへ移動する。
- ツール → プログラムメッセージを選択する。

演算ブロック

ラダーダイアグラムに、割り当て命令と IL 命令の挿入

演算ブロックのグラフィックシンボルを使って、ラダーダイアグラムのラングに IL 命令や割り当て命令を挿入できます。



演算ブロックのグラフィックシンボルは、1 列目以外のラダーダイアグラムの任意の位置に挿入できます (ラングの最初の接点として使用できないため)。




ラダーダイアグラムのラングで 1 つ以上の演算ブロックのグラフィックシンボルを使用する場合は、それらを直列に配置してください。演算ブロック命令を並列に使用することはできません。

注記：

アプリケーションがレベル 5.0 以上のファンクションレベル (77 ページ参照) で構成されている場合。

- 演算ブロックでは、最大 5 つのオペランドと入れ子の括弧を最大 3 つまで使用できます。オペランドには同じ型のオブジェクトを使用します (ワード型同士、フロート型同士など)。
- マスタータスクで複数のオペランドを使用するには、最低 20 のワードメモリー (%MW) が利用できる必要があります。周期タスクでも複数のオペランドを使用する場合は、さらに 20 個のワードメモリーを利用できる必要があります。

ラダーダイアグラムのラングに演算ブロックグラフィックを挿入するには

手順	手順内容
1	演算ブロック  ボタンをクリックします。
2	ラング上をクリックして、演算ブロックを挿入します。
3	ツールバー上の選択モード  ボタンをクリックします。
4	演算式ラインを、ダブルクリックします。 スマートコーディング (165 ページ参照) ボタン  が、ラインの最後に表示されます。このボタンをクリックすると、ファンクションと命令の構文が選択できます。
5	IL 命令または assignment instruction を入力し、ENTER を押します。 例 : %MF10 := ((SIN(%MF12 + 60.0) + COS(%MF13)) + %MF10) + 1.2 オンラインモードで式を編集できます。オンライン編集 (207 ページ参照) を参照してください。

注記：マルチオペランド命令はイベントタスクでは使えません。

OPER 命令の構文

OPER 命令は、ラングの任意の場所に配置された演算ブロックに対応します。

同等の OPER 命令は、IL ラングで直接使用できます。

OPER [式] は、式が有効で、最大 5 つのオペラントと入れ子の括弧が最大 3 つまで含まれません。例：

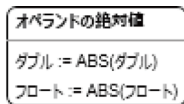
```
OPER [ %MF10 := ((SIN( %MF12 + 60.0 ) + COS( %MF13 )) + %MF10 ) + 1.2]
```

ラダーダイアグラムのスマートコーディングツールチップ


SoMachine Basic では、演算ブロックでファンクション名を入力時、ファンクションを選択するためにツールチップが表示されます。

ツールチップは、2 種類あります。

- ファンクション名のリスト。リストは入力された文字によって、動的に更新されます。例えば、“AS” と入力すると ASCII_TO_FLOAT、ASCII_TO_INT、ASIN が表示されます。
- ファンクションの構文のヘルプ。括弧開きを入力したときに表示されます。例えば、“ABS(“ と入力すると、以下が表示されます。



スマートコーディングアシスタントの使用

スマートコーディングアシスタントは、演算式ラインでスマートコーディングボタン  をクリックすると表示されます。



以下の手順に沿って進めます。

手順	手順内容
1	(オプション) ファンクションのカテゴリーでリストをフィルターします。 <ul style="list-style-type: none"> ● すべてのタイプ ● ASCII ● 浮動小数点 ● I/O オブジェクト ● 浮動小数点 ● 数値処理 ● テーブル ● PID ● ユーザー定義ファンクション
2	式に追加するファンクションを選択します。
3	ファンクションの挿入をクリックします。

構文のヘルプを取得

IL 命令の構文または assignment instruction が正しくない場合、**演算式**ボックスの縁が、赤色になります。次のいずれかでヘルプを参照してください。

- マウスを **演算式**ラインへ移動する。
- ツール → **プログラムメッセージ**を選択する。

コメントの追加

ラダー図にコメントを追加

次の手順で、ラダー図のプログラムにコメントを追加します。

手順	手順内容
1	ラングにグラフィック要素を挿入します。
2	必要であれば選択ポインターをクリックするか Esc を押します。
3	グラフィック要素の上部にあるコメントラインをダブルクリックします。
4	コメントをグラフィック要素用のコメントを入力し ENTER を押します。

ラダー図のコメントの例

以下にラダー図のラング上のコメントの例を示します。



プログラミングの推奨方法

プログラムジャンプの取り扱い

プログラムジャンプは、スキャンタイムが増加する可能性のある長いループを避けるように使用します。アップストリーム内の命令へのジャンプは避けます。

注記：アップストリーム命令ラインはプログラム内のジャンプの前に現れます。ダウンストリーム命令ラインはプログラム内のジャンプの後に現れます。

出力のプログラミング

物理的出力、および論理ビットはプログラム内で 1 回だけ変更します。物理的出力の場合、最後にスキャンされた値のみ更新時に考慮されます。

直接配線された緊急停止センサーの使用

緊急停止に直接使用されるセンサーはロジックコントローラーで処理しないでください。これらは対応する出力に直接接続します。また、地方、国内、国際的な規則に準拠させてください。

復電の取り扱い

停電後は手動で電源を復電します。インストールの自動再開は機器の意図しない動作を引き起こす可能性があります (システムビット %S0、%S1、および %S49 を使用します)。そのほかのシステムビット、およびシステムワードは停電後の再起動を管理しても差し支えないです。システムビット (%S)、およびシステムワード (%SW) (*Modicon M221, ロジックコントローラー, プログラミングガイド参照*) を参照してください。

⚠ 警告

装置の意図しない動作

機器およびソフトウェアが機能安全装置として指定され、それに適用される規格や基準に準拠していない限り、本ソフトウェアで設定およびプログラムした機器を安全上重要な機能に使用しないでください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

時間、およびスケジュール帳の管理

システムビット %S51 のステートは RTC のエラーを検出します。必ず確認してください。

文法の検証

プログラミングする際、SoMachine Basic は命令の文法、オペランド、およびそれらの関係を検証します。

括弧の使用に関する補注

割り当て命令を括弧内に書かないでください。

```
LD    %I0.0

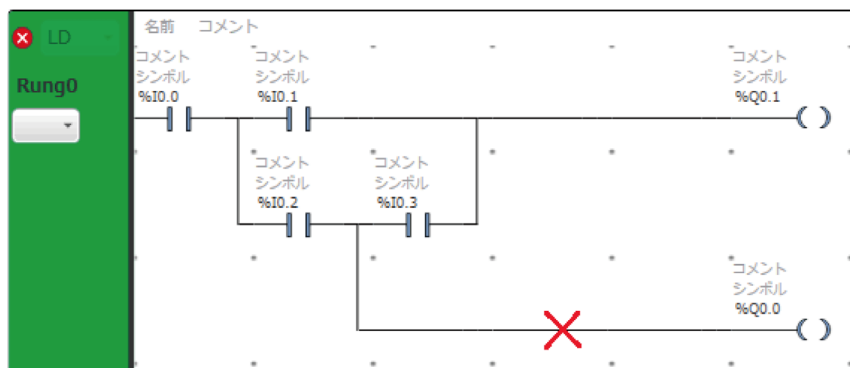
MPS

AND   %I0.1

OR(   %I0.2

)
```

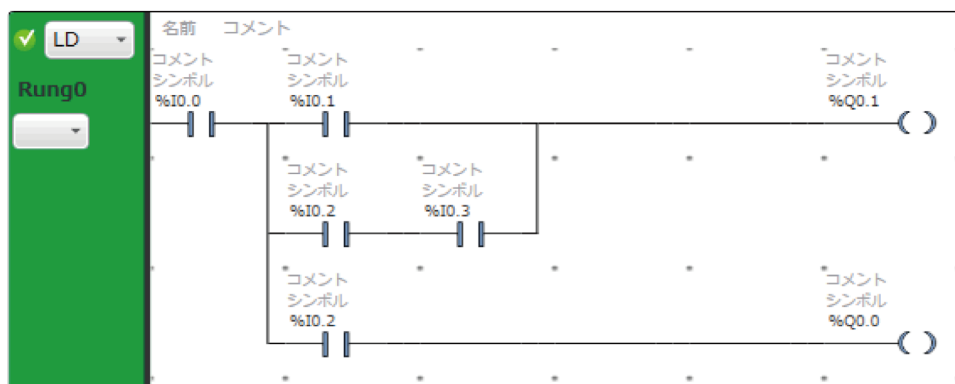

相当するラダー図は短絡エラーが発生します。



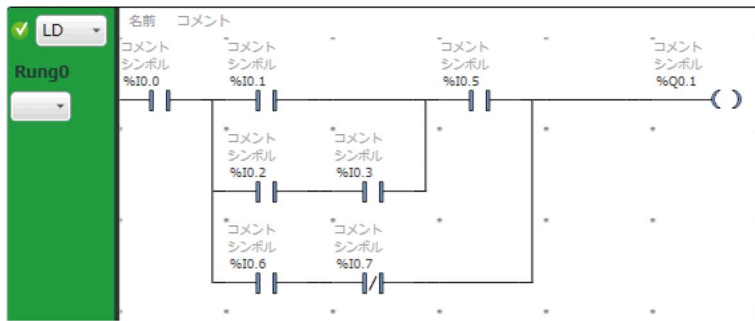
同じファンクションを実現するには、命令を次のようにプログラムします。

```
LD    %I0.0
MPS
AND ( %I0.1
OR ( %I0.2
AND %I0.3
)
)
ST   %Q0.1
MPP
AND %I0.2
ST   %Q0.0
```

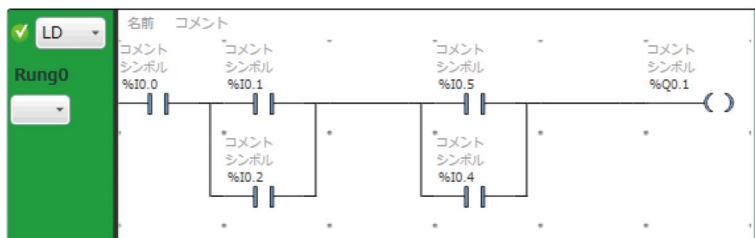
相当するラダー図



A 接点が平行になっている場合は互いに入れ子にします。



あるいは以下のように A 接点を完全に分離します。



6.13

命令リスト (IL) 言語のプログラミング

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
IL プログラムの概要	172
リスト命令の動作	174
リスト言語の命令	175
括弧の使用	178

IL プログラムの概要

はじめに

IL 言語で書かれたプログラムは、ロジックコントローラーにより順番に実行される一連の命令で構成されます。各命令は 1 行のプログラム行で表され、以下に示す部品で構成されます。

- 行番号
- 現在値 (オンラインモードのみ)
- 命令演算子
- オペランド
- オプションコメント

IL プログラムの例

以下は IL プログラムの例です。

IL	名前	コメント
0000	LD %M1	Load bit 1
0001	AND (%I0.1	Start a branch and load input bit 1
0002	OR (%I0.2	Load input bit 2
0003	ANDN %I0.3	Load input bit 3 and invert
0004)	コメント
0005)	コメント
0006	ST %Q0.0	Set out put bit 0

行番号

4 桁の行番号は新しいプログラム行とともに作成され SoMachine Basic により管理されます。

現在値

SoMachine Basic がオンラインモード (26 ページ参照) (ロジックコントローラーに接続中でプログラム実行中) の場合、SoMachine Basic は IL 編集ウィンドウ内のオブジェクトタイプの現在値を表示します。

これらのオブジェクトの表示値は更新されます。

命令演算子

命令演算子はオペランドと呼ばれるニーモニック記号であり、オペランドを使用して実行する演算を識別します。一般的な演算子はブール演算と数値演算をします。

上記のプログラムの例では LD は LOAD 演算子のニーモニックです。LOAD 命令はオペランド %M1 の値をブール値アキュムレーターと呼ばれる内部レジスターに格納します。

演算子には 2 種類あります。

- テスト演算子
アクションを行うのに必要な条件を設定、またはテストします。例えば LOAD (LD)、および AND。
- アクション演算子
これらは前段のロジックの結果としてアクションをします。例えば、STORE (ST)、および RESET (R) などの割り当て演算子。

演算子は、オペランドと共に命令を構成します。

オペランド

オペランドは、プログラムが命令内で操作できる値を表すオブジェクト、アドレス、またはシンボルです。上記のプログラムの例では、オペランド %M1 は、ロジックコントローラーの標準入力の値が割り当てられたアドレスです。命令は、命令演算子のタイプにより、0 から 3 個のオペランドをもつことができます。

オペランドは以下を表します。

- センサー、プッシュボタン、リレーなどのコントローラーの入出力
- タイマーやカウンタなどの、定義済みシステムファンクション
- 算術演算、論理演算、比較演算、および数値演算
- システムビット、システムワードなどのコントローラー内部変数

コメント

インストラクションリストプログラムにコメントを追加する

手順	手順内容
1	第 1 行目 0000 の上のラングの上部に表示されるコメントボックスをクリックして、ラング用のコメントを入力します (オプション)。
2	命令ラインの挿入
3	命令の右にある コメント エリアをクリックします。
4	コメントを入力し Enter を押します。

ラダー / IL エディターのカスタマイズ



IL エディターの上部にある以下のオブジェクトを使用してエディターのコンテンツをカスタマイズします。

オブジェクト	説明
IL>LD	すべてのラングの表示を IL から Ladder に切り替えます。
LD>IL	すべてのラングの表示を Ladder から IL に切り替えます。
-	IL グリッドから 1 列削除します。列数が最小数 (11) になるとボタンは無効になります。
+	IL グリッドに 1 列追加します。列数が最大数 (30) になるとボタンは無効になります。
コメントの表示 / 非表示	クリックするとラングのメッセージが表示、または非表示になります。
T	アドレスモード、またはシンボルモードのオブジェクトの表示をクリックで切り替えます。
DEC/HEX	オンラインモードでのみ有効です。クリックしてラングの数値を 10 進数、または 16 進数の表示に切り替えます。
1 - 新規 POU	ツール → マスタータスクエリアに表示される、デフォルトの POU 名をダブルクリックして編集します。
コメント	ダブルクリックしてテキストを入力し、この POU にコメントを関連付けます。
ズームスライダー	ラダーエディターを拡大 / 縮小します。ショートカット Ctrl + マウスホイール を使用して拡大または縮小することができます。プロジェクトを移動させても拡大位置は変わりません。

リスト命令の動作

はじめに

IL バイナリーインストラクションは通常 1 つのエクスプリシットオペランドを持ちます。もう一方はインプライドオペランドです。インプライドオペランドは、ブール値アキュムレーターの値です。例えば、命令 LD %I0.1 では、%I0.1 がエクスプリシットオペランドです。インプリシットオペランドはアキュムレーターにロードされ、アキュムレーターに入っていた値は、%I0.1 の値に上書きされます。この値は、下位命令ではインプリシットな値となります。

操作設定

IL 命令は特殊なアキュムレーターの内容とエクスプリシットオペランドに対して指定された演算を実行し、アキュムレーターの内容を結果に置き換えます。例えば AND %I1.2 の演算は、アキュムレーターの内容と入力 1.2 との論理積 (AND) を実行し、アキュムレーターの内容をこの結果と置き換えます。

ロード、格納、および否定以外のすべてのブール演算命令は 2 つのオペランドで演算します。2 つのオペランドの値は True、または False のいずれかになり、命令プログラムの実行は True、または False のどちらか 1 つの値を生成します。ロード命令は格納命令がアキュムレーターの値をオペランドに転送する間に、オペランドの値をアキュムレーターに格納します。否定命令は、エクスプリシットオペランドをもたず、単純にアキュムレーターのステートを逆転するものです。

対応するリスト命令

IL 言語の命令の一覧を次の表に示します。

命令のタイプ	例	ファンクション
ブール命令	LD %M10	アキュムレーターに内部ビット %M10 の値をロードします。
ブロック命令	IN %TM0	タイマー %TM0 を開始します。
ワード命令	[%MW10 := %MW50+100]	加算命令
プログラム命令	SR5	サブルーチン #5 を呼び出します。

リスト言語の命令

はじめに







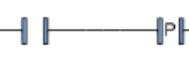

IL 言語には、次に示す命令やブロック命令があります。

- テスト命令
- アクション命令
- ファンクションブロック

リストプログラミングの命令について説明します。

テスト手順

リスト言語のテスト命令を次の表に示します。


ニーモニック	名前	対応するグラフィック要素	ファンクション
LD	Load		オペランドのブール値をアキュムレーターに読み込みます。
LDN	Load Not		オペランドのブール値を反転したものをアキュムレーターに読み込みます。
LDR	Load Rising		値が 0 から 1 に変わったときにオペランドのブール値をアキュムレーターに読み込みます (立上がり接点)。オペランドが 0 から 1 に移行するまでアキュムレーターの値は 0 が読み込まれます。
LDF	Load Falling		値が 1 から 0 に変わったときにオペランドのブール値をアキュムレーターに読み込みます (立下り接点)。オペランドが 1 から 0 に移行するまでアキュムレーターの値は 1 が読み込まれます。
AND	And		ブールの結果は、直前の命令のブールの結果 (アキュムレーターに格納されたもの)、およびオペランドの状態の、AND 論理演算結果に等しい。その命令の結果は、直前の値を上書きするアキュムレーターに暗黙的に読み込まれます。
ANDN	And Not		ブールの結果は、直前の命令のブールの結果 (アキュムレーターに格納されたもの)、およびオペランドの状態を反転したものと AND 論理演算結果に等しい。その命令の結果は、直前の値を上書きするアキュムレーターに暗黙的に読み込まれます。
ANDR	And Rising		ブールの結果は、直前の命令のブールの結果 (アキュムレーターに格納されたもの)、およびオペランドの立上がり接点の AND 論理演算結果に等しい (1 = 立上がり接点) その命令の結果は、直前の値を上書きするアキュムレーターに暗黙的に読み込まれます。
ANDF	And Falling		ブールの結果は、直前の命令のブールの結果 (アキュムレーターに格納されたもの)、およびオペランドの立下り接点の AND 論理演算結果に等しい (1 = 立下り接点) その命令の結果は、直前の値を上書きするアキュムレーターに暗黙的に読み込まれます。

ニーモニック	名前	対応するグラフィック要素	ファンクション
OR	Or		ブールの結果は、直前の命令のブールの結果、およびオペランドの状態 (アキュムレーターに格納されたもの) の OR 論理演算結果に等しい。
AND(And With		論理 AND (括弧は最大 32 階層まで)。括弧は、括弧内の命令の中間論理結果を指定します。そしてその結果はアキュムレーターの値と論理 AND されます。
OR(Or With		論理 OR (括弧は最大 32 階層まで)。括弧は、括弧内の命令の中間論理結果を指定します。そしてその結果はアキュムレーターの値と論理 OR されます。
XOR XORN XORR XORF	Ex Or Ex Or Not Ex Or Rising Ex Or Falling		エクスクルーシブ OR
MPS MRD MPP	Memory Push Store Memory Read Memory PoP		出力アクションを分岐する演算子。
N	Not		オペランドの値を反転します。

アクション命令


リスト言語のアクション命令を次の表に示します。

ニーモニック	名前	対応するグラフィック要素	ファンクション
ST	Store		関連付けられたオペランドにテストゾーンの結果の値が入ります。
STN	Store Not		関連付けられたオペランドにテストゾーンの結果を反転した値が入ります。
S	Set		テストゾーンの結果が 1 の場合、関連付けられたオペランドに 1 がセットされます。
R	Reset		テストゾーンの結果が 1 の場合、関連付けられたオペランドに 0 がセットされます。

ニーモニック	名前	対応するグラフィック要素	ファンクション
JMP	Jump		無条件にラベル付きシーケンス、アップストリーム、またはダウンストリームに接続します。
SRn	Subroutine		サブルーチンの開始部の接続 (サブルーチン呼び出し)。
END	End		プログラムの終端
ENDCN	End Conditional		ブール演算の結果が 0 という条件でプログラムを終了する。

ファンクションブロック

リスト言語のファンクションブロックを次の表に示します。

名前	対応するグラフィック要素	ファンクション
タイマー、カウンタ、レジスタなど		<p>各ファンクションブロックに対しブロックを制御する命令があります。</p> <p>構造化された形式がブロックの入力と出力を接続するのに使用されます。</p> <p>メモ: ファンクションブロックの出力同士を接続することはできません (縦短絡)。</p> <p>詳細はソフトウェアオブジェクト (<i>SoMachine Basic</i>, <i>Generic Functions Library Guide</i> 参照) を参照してください。</p>

括弧の使用

はじめに

論理演算子 AND、および OR と共に、括弧は論理命令を入れ子にするのに使用されます。その際、ラダーエディターで分岐 (分枝) を指定します。括弧は次のような指示に関連しています。

- 始め括弧は AND 演算子、または OR 演算子に対応しています。
- 終わり括弧は命令 (オペランドのない演算子) で、各始め括弧に必要なものです。

例: AND 命令の使用

AND 命令と括弧の使用例を以下に示します。

ラング	EX 命令表記
0	LD %I0.0 AND %I0.1 OR %I0.2 ST %Q0.0
1	LD %I0.0 AND(%I0.1 OR %I0.2) ST %Q0.1

注記: 可逆性の手順 (*SoMachine Basic, Generic Functions Library Guide 参照*) を参照して、相当するラダーダイアグラムを入手します。

例: OR 命令の使用

OR 命令と括弧の使用例を以下に示します。

ラング	命令
0	LD %I0.0 AND %I0.1 OR(%I0.2 AND %I0.3) ST %Q0.0

注記: 可逆性の手順 (*SoMachine Basic, Generic Functions Library Guide 参照*) を参照して、相当するラダーダイアグラムを入手します。

修飾子

括弧に割り当てられる修飾子を次の表に示します。

修飾子	ファンクション	例
N	否定	AND(N、または OR(N
F	立下り接点	AND(F、または OR(F
R	立上がり接点	AND(R、または OR(R
[比較	比較命令 を参照してください。

注記: 修飾子 '[' は、演算子として機能する他の命令と組み合わせて使用することもできます。その他の命令における '[' の使用については、数値操作の概要 を参照してください。

括弧の入れ子

括弧は 32 階層まで入れ子にできます。

括弧を入れ子にする場合、以下のルールに従ってください。

- 始め括弧には必ず対応する終わり括弧が必要です。
- ラベル (%Li:)、サブルーチン (SRi:)、JMP 命令 (JMP)、およびファンクションブロック命令は、括弧内の式に入れることはできません。
- 格納命令 (ST、STN、S、および R) は括弧内に入れることはできません。
- スタック命令 (MPS、MRD、および MPP) は括弧内に入れることはできません。

括弧の入れ子の例

括弧を入れ子にする方法を下の例に示します。

ランク	EX 命令表記
0	LD %I0.0 AND(%I0.1 OR(N %I0.2 AND %M3)) ST %Q0.0
1	LD %I0.1 AND(%I0.2 OR(%I0.5 AND %I0.6) AND %I0.4 OR(%I0.7 AND %I0.8)) ST %Q0.0

注記：可逆性の手順 (*SoMachine Basic, Generic Functions Library Guide* 参照) を参照して、相当するラダーダイアグラムを入手します。

6.14

グラフセ (リスト) プログラミング

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
グラフセ (リスト) プログラミングの詳細	181
グラフセ (リスト) プログラム構成	182
SoMachine Basic プログラムでのグラフセ (リスト) の使い方	185

グラフセ (リスト) プログラミングの詳細

はじめに

SoMachine Basic のグラフセ (リスト) プログラミングは、制御シーケンスをステップに変換する手段です。制御シーケンスをグラフセステップに変換し、そのステップをグラフセ命令を使用したプログラムで使うことができます。

グラフセステップの最大数はコントローラーに依存します。常に有効であるステップ数は、ステップの合計数によってのみ制限されます。

グラフセ命令

SoMachine Basic グラフセプログラムには以下の命令があります。

演算子	オペランド	IL 命令	命令名	図	説明
==	x	== x	INITIAL STEP		この命令は、プログラムの初期ステップを定義します。
== POST	適用外	== POST	POST PROCESSING (内在オペランド)		この命令は、シーケンス処理の終了と後処理を定義します。
-*	x	-* x	STEP		この命令は、プログラムの移行検証用ステップを定義します。
#	適用外	#	DEACTIVATE CURRENT STEP (内在オペランド)		この命令は、プログラムの現在のステップを無効にします。
#	x	# x	DEACTIVATE CURRENT STEP、および ACTIVATE STEP x		この命令は、プログラムの現在のステップを無効にし、ステップ x を有効にします。
#D	x	#D x	DEACTIVATE CURRENT STEP、および STEP x		この命令は、プログラムの現在のステップ、およびステップ x を無効にします。
S	x	S x	ACTIVATE STEP x		この命令は、プログラムのステップ x を有効にします。この処理は、他の有効なステップには影響しません。
R	x	R x	DEACTIVATE STEP x		この命令は、プログラムのステップ x を無効にします。この処理は、他の有効なステップには影響しません。

x グラフセステップ番号 (1 から始まる整数)

グラフセ (リスト) プログラム構成

はじめに

SoMachine Basic グラフセ (リスト) プログラムには以下の処理があります。

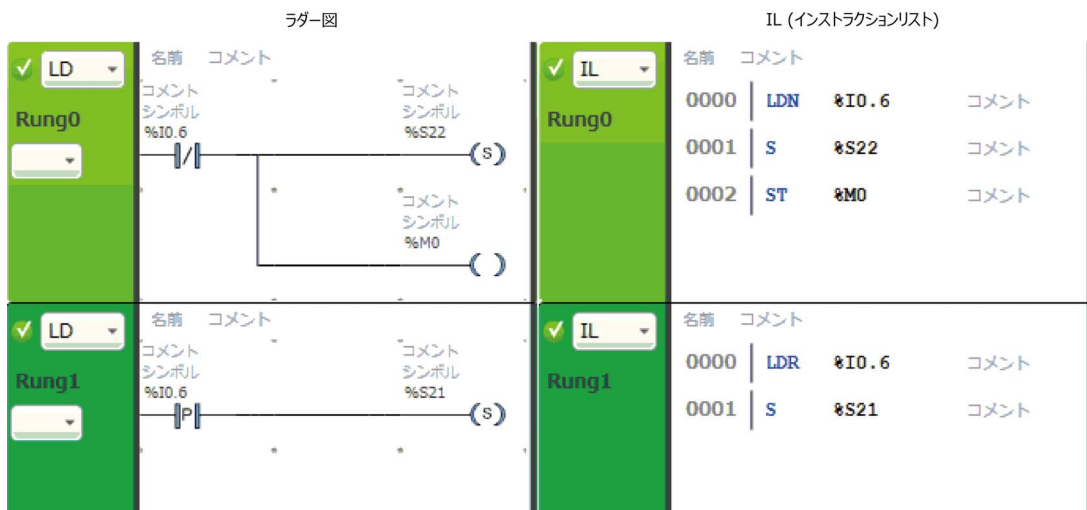
- 前処理
- シーケンス処理
- 後処理

前処理

前処理は、以下で構成されています。

- 電源回復
- エラー管理
- 操作モードの変更
- 前処理グラフセステップ
- ロジックの入力

以下の例では、システムビット %S21 は入力 %I0.6 (ラング 1) の立上がり接点で、1 に設定されます。これにより、有効なステップが無効になり、初期ステップが有効になります。



前処理はプログラムの最初の行から始まり、`=*`、または `-*` 命令が初めて発生した時に終了します。

システムビット %S21、%S22、および %S23 は、グラフセ制御専用です。これらのシステムビットは、通常は前処理で (必要に応じて) アプリケーションにより 1 に設定されます。前処理の最後に、関連するファンクションがシステムによって実行され、システムビットは 0 にリセットされます。

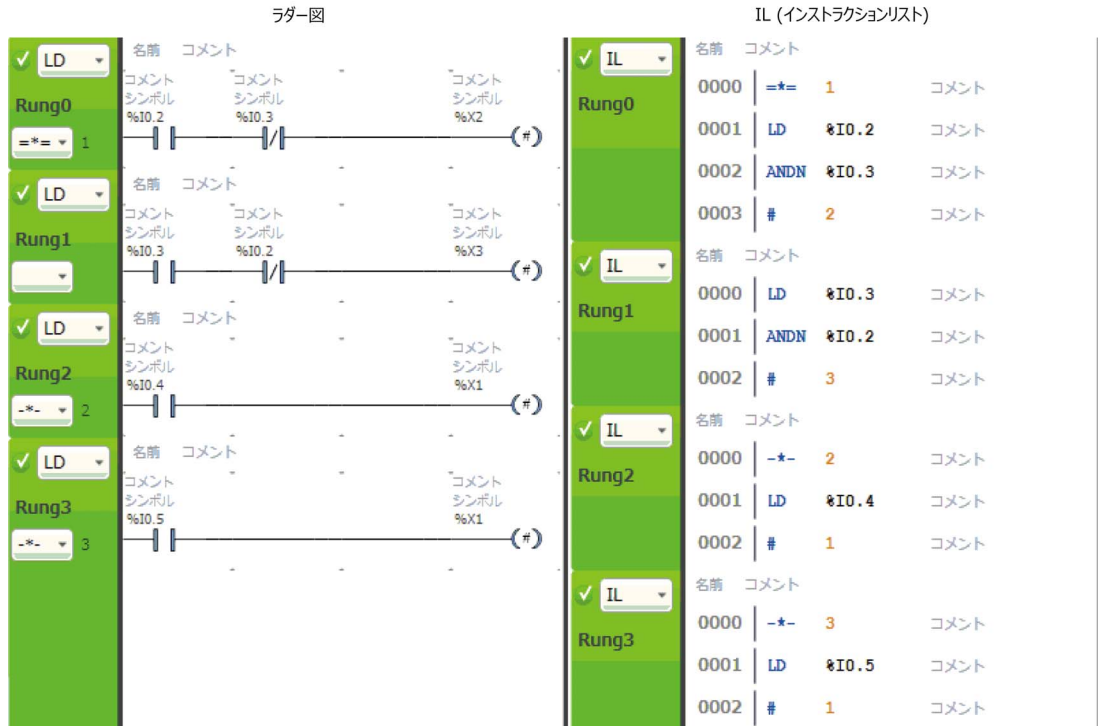
システムビット	名前	説明
%S21	グラフセ初期化	すべての有効なステップが無効化され、初期ステップが有効になります。
%S22	グラフセ再初期化	すべてのステップが無効になります。
%S23	グラフセの事前配置	%Xi オブジェクトが前処理中にアプリケーションによって明示的に書かれる場合、このビットを 1 に設定します。%Xi オブジェクトを明示的に変更しないでこのビットを 1 に保持すると、グラフセはフリーズします (更新は考慮されません)。

シーケンス処理

シーケンス処理はチャートで実行されます (チャートを表す命令)。

- ステップ
- ステップに関連するアクション
- 移行
- 移行条件

例：



シーケンス処理は、**POST** 命令の実行、またはプログラムの終了で終わります。

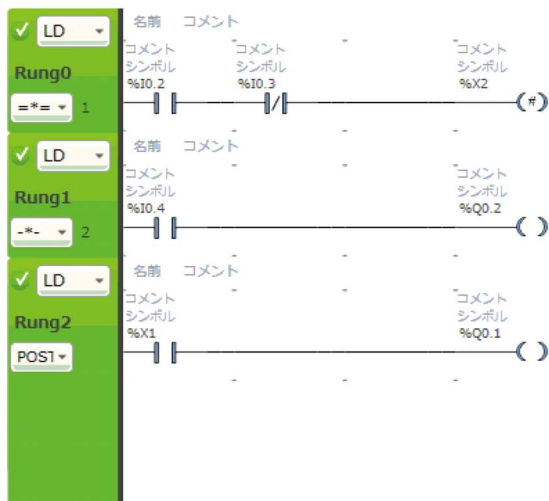
後処理

後処理は以下を含みます。

- 出力制御のためのシーケンス処理からのコマンド
- 出力に固有のインターロック

例：

ラダー図



IL (インストラクションリスト)

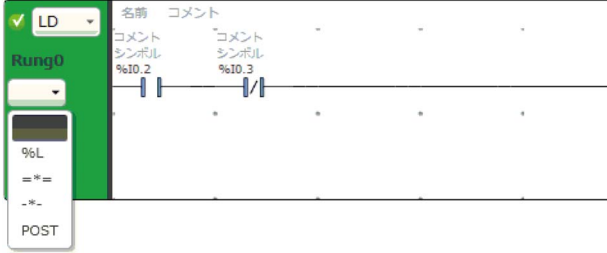
名前	コメント
Rung0	0000 == 1 コメント
	0001 LD %I0.2 コメント
	0002 ANDN %I0.3 コメント
	0003 # 2 コメント
Rung1	0000 -- 2 コメント
	0001 LD %I0.4 コメント
	0002 ST %Q0.2 コメント
Rung2	0000 == POST コメント
	0001 LD %X1 コメント
	0002 ST %Q0.1 コメント

SoMachine Basic プログラムでのグラフセ (リスト) の使い方

注記： グラフセ (リスト) 命令は、プログラムのマスタータスクでのみ使用できます。



ラダーのグラフセ (リスト) ステップの作成

次の手順で、プログラムにグラフセステップを作成します。

手順	手順内容
1	<p>POU のラングを選択し、ラングシーケンス識別子 ラング x の下のドロップダウンボタンをクリックします。x は POU のラング番号です。</p>  <p>結果： 使用できるグラフセ (リスト) 一覧のメニューが表示されます。</p>
2	<p>一覧にある命令をクリックして、ラングを初期ステップ、後処理、またはグラフセ (リスト) プログラムのステップとして定義します。</p> <p>結果： ラングがグラフセ命令として設定されます。ボタン上に命令の演算子が表示され、オペランド (ステップ番号) はボタンの接尾辞として表示されます。</p> <p>注記： ステップ番号は、次の STEP、または INITIAL STEP 命令を定義するときに 1 ずつ増加します。プログラムでは 1 つの POST 命令しか定義できないため、POST 命令にはステップ番号はありません。</p> <p>ステップ番号を変更するには、ラングのステップ番号をダブルクリックして新しい番号を入力し、ENTER を押します。</p>

ラダーのグラフセ (リスト) ステップの有効化、および無効化

次の手順で、プログラムにグラフセ (リスト) ステップを有効、または無効にします。

手順	手順内容
1	POU で、プログラムのラングを選択します。
2	 をクリック (現在のステップを無効にし、任意で特定のステップを有効にします)、または  をクリック (現在のステップ、および特定のステップを無効にします) して、ラングのアクション領域に要素を挿入します (詳細は、グラフィック要素の挿入 (158 ページ参照)) を参照してください。
3	または、ALT+A を押して ACTIVATE 命令、ALT+D を押して ACTIVATE 命令をラングで使用することもできます。 結果: ラングのアクション領域に、ラダーシンボルの有効化、または無効化が表示されます。ENTER を押して要素を挿入します。
4	プログラムラングのグラフセシンボルの有効化、または無効化の アドレス フィールドをダブルクリックして、グラフセビットアドレス (%Xi、i はステップ番号) を入力します。 例えば、%X4 はグラフセプログラムのステップ 4 を示しています。%X4 が無効化シンボルのアドレスである場合、このシンボルが使われているラングの出力が true のとき、ステップ 4 は無効化されます。 注記: どの場合でも、現在のステップは無効化されます。

6.15

グラフセ (SFC) プログラミング

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
グラフセ (SFC) プログラミングの概要	188
グラフセ (SFC) グラフィカルエディターの使用	191
分岐	194
プログラミング推奨方法	198

グラフセ (SFC) プログラミングの概要

はじめに

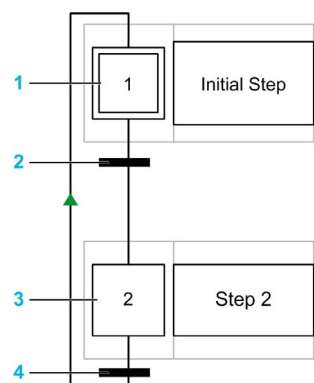
グラフセ (SFC) は、ステップとして知られている、各タスクの実行を時系列に記述するグラフィカルプログラミング言語です。ステップが実行される順序は、ステップを繋ぐ移行によって決められます。

グラフセ (SFC) POU の要素

グラフセ (SFC) POU には以下のコンポーネントがあります。

- **ステップ**: ステップは、ラダー/IL プログラミング言語で書かれたラング内で定義された一連のアクションを実行します。ステップは以下になることができます。
 - **初期ステップ**: プログラムの開始時、またはコントローラーの再起動後に実行されます。境界線が二重線のセルで表示されます。
 - **通常ステップ**: 初期ステップの実行が完了後、条件によって実行されるステップです。
- **移行**: ステップ間で評価されるブール式です。複数のステップ間のリンクです。ブール式は、ラダー/IL プログラミング言語で書かれる 1 つの移行ラングで定義されます。

グラフセ (SFC) POU の初期ステップが 1 つ、通常ステップが 1 つ、および移行が 2 つある例を下図に示します。



- 1 初期ステップ
- 2 ステップ 1 からステップ 2 への移行
- 3 通常ステップ
- 4 ステップ 2 からステップ 1 へ戻る移行リンクの上に矢印が表示され、ステップの実行順序が初期値の左から右、上から下ではないことを示しています。

グラフセ (SFC) POU のルール

グラフセ POU はプログラムのマスタータスクでのみ作成できます。

複数の グラフセ POU を作成できます。

グラフセ (SFC) 処理

グラフセ (SFC) を実行するときには、ロジックコントローラーによって以下のルールが適用されます。

- マスタータスクサイクルの開始。
- 最初のグラフセ (SFC) ステップより前にある POU が順次実行されます。
- 最初のグラフセ (SFC) ステップで**グラフセモニター**が起動します。
- **グラフセモニター**が終了すると、最後のグラフセ (SFC) ステップに続く最初の POU が呼び出されます。

グラフセモニター動作

1. ロジックコントローラーは関連する グラフセ (SFC) システムビット %S21、%S22、および %S23 を処理します。
2. ロジックコントローラーは各グラフセ (SFC) ステップの有効化ステータスを更新します。
 - 無効化とマークされたステップは、無効になります。
 - 有効化とマークされたステップは、有効になります。
 - 同時に有効化、および無効化とマークされたステップは、有効化されるか、または有効のままになります。
 - 有効化、および無効化一覧がリセットされます。
3. ロジックコントローラーがステップをスキャンします (定義されたステップ番号の最下位から最上位にループ)。スキャンされたステップが有効になると、関連するステップコードが呼び出されます。
4. 移行コードがステップを有効化、または無効化すると、このアクションは、次のタスクサイクル用有効化、または無効化一覧にそれぞれ配置されます。
5. 最後の有効なステップコードが実行されると、**グラフセモニター**が終了します。

マルチトークン動作

SoMachine Basic グラフセ POU は IEC 61131-3 には準拠していないマルチトークンです。

初期状態は、初期ステップとして定義されたステップによってによって制御されます
グラフセ POU で、複数のステップを同時に有効にすることができます。

有効な信号ステータス処理は、1 つ、または複数の移行を切り替えることによってトリガーされ、リンクの方向に従って行われます。処理の方向はリンク方向に従っていて、前のステップの下側から次のステップの上側まで実行されます。

移行が評価されるのは、直前のステップが有効な場合です。直前のステップが有効でない場合、移行は評価されません。

移行に関連する条件が満たされたとき、移行がトリガーされます。

移行のトリガーは、移行にリンクされている直前のステップを無効にし、すぐ次のステップを有効にするようにマークされています。

ステップの実際の有効化、または無効化は、各マスタートaskサイクルの始めに実行されます (**グラフセモニター** (188 ページ参照) を参照)。

一連のステップの行に条件が満たされている移行が複数ある場合、サイクルごとに 1 つのステップが処理されます。

ステップが同時に有効化、および無効化された場合、ステップは、有効化されるか、または有効のままになります。

複数の分岐を、代替分岐で有効にできます。

実行される分岐は、代替分岐に続く移行の移行条件の結果によって決められます。分岐移行は並列に処理されます。

条件の満たされた移行のある分岐がトリガーされます。

サブルーチンコールは、ステップアクションでも使えます。

グラフセ (SFC) の制御ビット

制御ビット	名前	説明
%S21	グラフセ初期化	1 に設定されている場合、グラフセ POU の初期ステップは評価されています。
%S22	グラフセリセット	1 に設定されている場合、ステップが無効化され、実行が再開されます。
%S23	グラフセのプリセットとフリーズ	1 に設定されている場合、ビットが 0 に設定されるまでグラフセ POU の実行を停止します。
%Xi	グラフセステップ	ビット %X1 から %Xi はグラフセステップに関連付けされています。ステップビット %Xi が 1 に設定されているときは、対応するステップが有効です。0 に設定されているときは、ステップが無効です。グラフセ (SFC) 使用時、このビットは書き込みできません。

詳細は、システムビット (*Modicon M221, ロジックコントローラー, プログラミングガイド 参照*) の説明を参照してください。

グラフセ (SFC) グラフィカルエディターの使用

概要

グラフセエディターは、グラフセ (SFC) でプログラミングするために使われます。

グラフセエディターを表示するには、ツリービューの n -**グラフセノード** を選択します。


グラフセエディターにはセルのグリッドが含まれます。各セルには、1つのステップ、1つの移行または両方が含まれます。

グラフセ POU の最小サイズはステップが 1 つです。

アプリケーションのステップ最大数は 96 です。

グラフセグラフィカルエディターの分離

グラフセエディターウィンドウは、メインの SoMachine Basic ウィンドウから分離させ、独立して移動およびサイズの変更をすることができます。例えば、別のモニターに移動させて、グラフセ POU と IL / ラダー POU を同時に表示することもできます。

ウィンドウを分離するにはグラフセエディターウィンドウの右上端にある  ボタンをクリックします。

ウィンドウのタイトルバーをドラッグして動かします。ウィンドウを閉じると、通常ビューに戻ります。

ステップの挿入

ステップを追加するグリッドセルをダブルクリックするか、グリッドセルを右クリックして、表示されているコンテキストメニューから**ステップの追加**を選択します。

グラフセエディターウィンドウの右上隅に**使用されているグラフセのステップ数**が表示されます。

ドラッグ&ドロップでステップを別のグリッドセルに移動できます。

ステップタイプの変更 (初期または通常)

初期設定により、グラフセエディターで最初に作成されるステップは初期ステップです。

グラフセ POU には必ず初期ステップをいれてください。複数のステップを初期ステップとして定義することもできます。

ステップタイプ (初期 / 通常) を変更するには、ステップを右クリックして**初期ステップとして設定 / 設定解除**を選択します。

ステップのコピーと貼り付け

手順	手順内容
1	コピーするステップを右クリックして、表示されたコンテキストメニューから コピー を選択します。
2	空のグリッドセルを右クリックして、 貼り付け を選択します。 結果 : ステップのコピーが表示されます。ステップに関連する ラダー / IL ラングのコピーが、ツリービューの対応する ステップサブノード の下に追加されます。

移行の作成

ステップをリンクし、ステップ実行の順序を定義します。

2つのステップ間に移行を作成します。

手順	手順内容
1	ステップの下にマウスを移動します。 結果: 緑のブロックが表示されます。
2	マウスをリンクしたいステップにドラッグします。
3	マウスボタンを離します。 結果: リンクと移行が表示されます。

ラベルの編集


ステップまたは移行の初期値のラベルを編集します。

手順	手順内容
1	グラフセ (SFC) ステップまたは移行のラベルをダブルクリックします。
2	ステップまたは移行要素に新しい名前を入力し、ENTER キーを押します 例えば、初期値の <i>Step_1</i> ラベルを <i>INIT</i> に変更します。

ステップ機能のプログラミング

ステップの機能は1つまたは複数の IL / ラダー言語のラングで定義されます。


ステップの機能を定義します。

手順	手順内容
1	以下のいずれかを行います。 <ul style="list-style-type: none"> ● グラフセエディターのステップをダブルクリックします。 ● ツリービューのステップノードを選択します。<i>n</i>はステップ番号です。 結果: グラフセエディターが閉じます。
2	選択した ステップ ノードを右クリックして、表示されたコンテキストメニューから ラングの追加 を選択します。 結果: ラングが、ツリービューウィンドウに ステップ ノードのサブノードとして表示されます。
3	ラダー言語プログラミング (152 ページ参照) または命令リストプログラミング (171 ページ参照) の説明にあるように、ラダーまたは IL プログラミング言語でラングをプログラミングし、必要に応じて追加のラングを作成します。
4	グラフセエディターを再度表示するには、以下のいずれかを行います。  アイコンをクリックする。 <ul style="list-style-type: none"> ● <i>n</i> - グラフセ POU ノードを選択します。<i>n</i> はグラフセ POU の番号です。

移行機能のプログラミング

移行の機能は、1つのIL / ラダー言語の移行ラングで定義されます。

移行ラングの機能を定義します。

手順	手順内容
1	<p>以下のいずれかを行います。</p> <ul style="list-style-type: none">● グラフセエディターの移行をダブルクリックします。● ツリービューの Transitions → Trn ノードを選択します。 <p>結果: グラフセエディターが閉じ、ラダー言語のラングが表示されます。</p>
2	<p>ラダー言語プログラミング (152 ページ参照) または命令リストプログラミング (171 ページ参照) の説明にあるように、ラダーまたは IL プログラミング言語でラングをプログラミングします。</p> <p>ビットシフトレジスタまたはステップカウンターなどのように出力がないものを除き、ファンクションブロックで移行ラングを使用することができます。</p> <p>ファンクションブロックを使用する場合は、END_BLK 命令は ENDT 命令の直後におきます。以下は例です。</p> <pre>Tr1 Comment 0000 BLK %TM2 0001 LD 0 0002 IN 0003 OUT_BLK 0004 LD Q 0005 ENDT 0006 END_BLK</pre> <p>注記: ラングは ENDT (移行の終了) 命令で終わります。この命令は選択または変更できません。また、ラングに FB が含まれていない限りラングの最後の命令にします。</p>
3	<p>グラフセエディターを再度表示するには、以下のいずれかを行います。</p> <ul style="list-style-type: none">●  アイコンをクリックする。● n - グラフセ POU ノードを選択します。n は グラフセ POU の番号です。

元に戻す / やり直す

保存されている最大 10 個のアクションに対して、ツールバーの **元に戻す** または **やり直す** ボタンが使えます。

ステップまたは移行の削除

手順	手順内容
1	<p>グラフセエディターで、以下を行います。</p> <ul style="list-style-type: none">● ステップまたは移行を選択して、DELETE キーを押します。● ステップまたは移行を右クリックして、コンテキストメニューの 選択した項目を削除 を選択します。 <p>結果: 選択したステップまたは移行が削除されます。</p> <p>注記: ツリービューからはステップまたは移行は削除できません。</p>

分岐

はじめに

グラフセ (SFC) POU は分岐を使えます。

2 種類の分岐があります。

- 並列分岐 : 前の移行が真の場合、2 つ以上のステップが同時に処理されます。
- 代替分岐 : 前の移行条件 (マルチトークン動作) の結果に応じて、1 つ以上の代替ステップが処理されます。

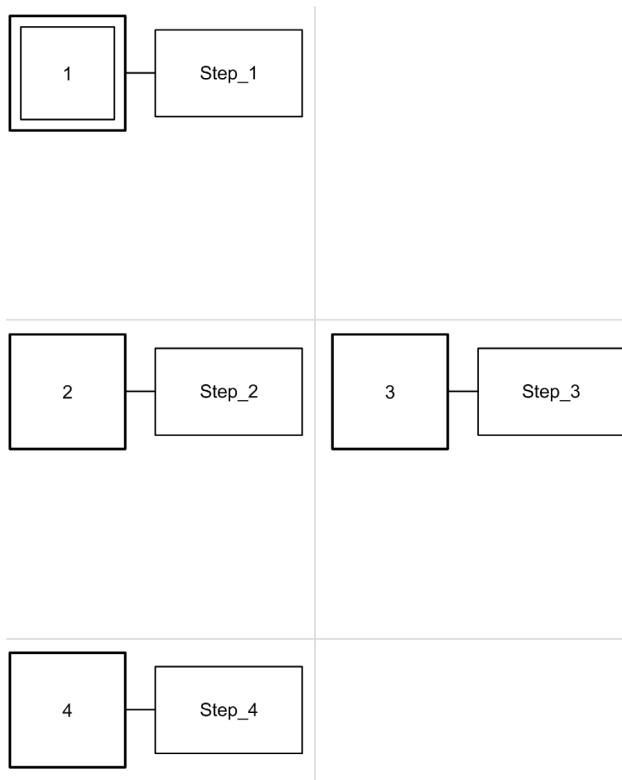
並列分岐

並列分岐は 1 ステップから複数のステップに移行できます。

並行分岐を行い、その後にはステップを実行する必要があります。

並列分岐には、入れ子になった代替分岐、またはその他の並列分岐を含むことができます。

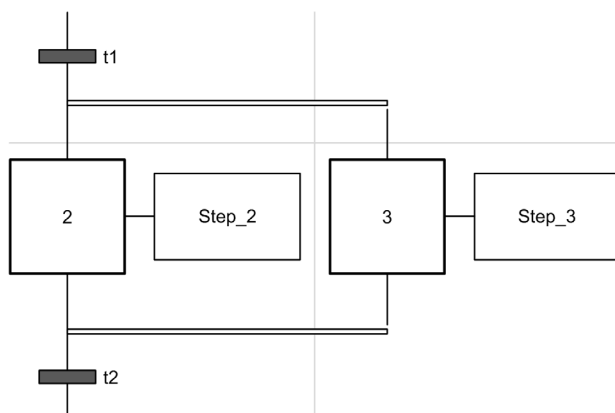
並列分岐を作成する前に 4 ステップのグラフセ POU の例を次の表に示します。



ステップ 2 と 3 の並行分岐を作成するには：

手順	手順内容
1	ステップ 1 とステップ 2 の移行を作成：ステップ 1 の一番下にマウスを移動し、ステップ 2 にドラッグしてマウスボタンを離します。新しいリンクと移行表示されます。
2	ステップ 3 と移行の間にリンクを作成：ステップ 3 の一番上にマウスを移動し、移行にドラッグしてマウスボタンを離します。 結果 ：既存のトランジションの下に横の二重線が表示されます（次の図を参照）。 注記 ：移行と POU の上部にあるステップの間にリンクを作成するには、ステップのリンクから始めて、移行にドラッグします。
3	メインの処理分岐を使用して分岐に復帰させるには、ステップ 2 とステップ 4 の間に移行を作成します。
4	ステップ 3 と移行の間に新規移行を作成：ステップ 3 の一番下にマウスを移動し、移行にドラッグしてマウスボタンを離します。 結果 ：既存のトランジションの上に横の二重線が表示されます（次の図を参照）。

並列分岐を作成した後のグラフセ POU を次の表に示します。



分岐した領域の前後の水平ラインは二重線です。

代替分岐

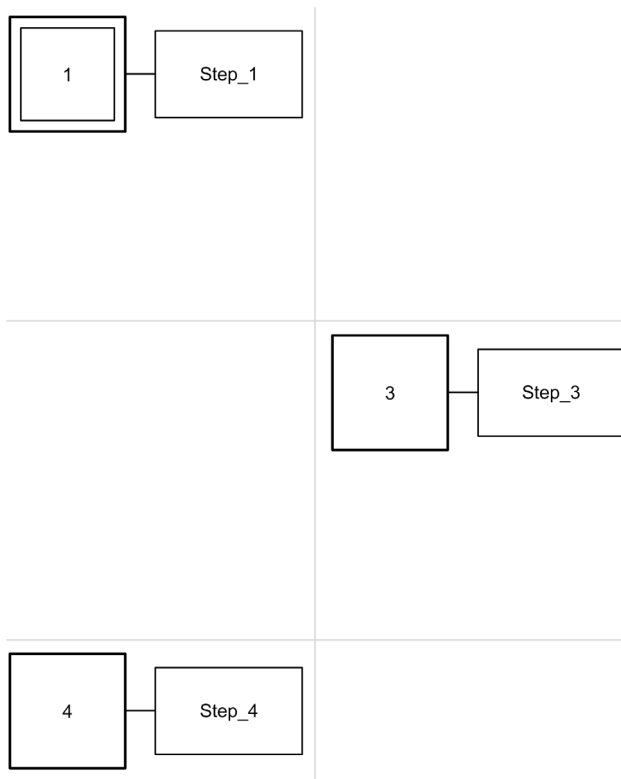
代替分岐は、移行で開始、および終了します。

代替分岐は、入れ子になった並列分岐、または他の代替分岐を含むことができます。

マルチトークン動作では、移行から複数の並列分岐を作成できます。実行される分岐は、代替分岐に続く移行の移行条件の結果によって決められます。分岐の移行が処理されます。条件の満たされた移行のある分岐がトリガーされます。

代替ブランチを排他的に切り替える必要がある場合（モノトークン動作）、移行コードで明示的に定義します。

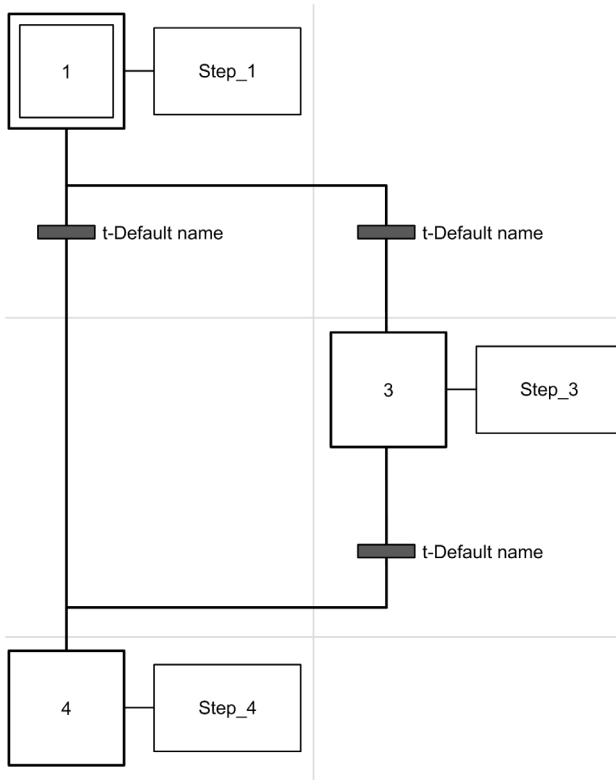
ステップ 3、およびステップ 4 の代替分岐作成前の、3 ステップの グラフセ POU の例を下図に示します。



代替分岐を作成します。

手順	手順内容
1	ステップ 1 とステップ 4 の間に移行を作成します。 結果: 新しいリンクと移行表示されます。
2	ステップ 1 とステップ 3 の間に移行を作成します。ステップ 1 の下にマウスを移動し、ステップ 3 にドラッグしてマウスボタンを離します。 結果: 新規リンク、および移行が既存移行の上の分岐に表示されます (下図を参照してください)。
3	ステップ 3 とステップ 4 の間に移行を作成します。 結果: 新規リンク、および移行がステップ 1 とステップ 4 の間の既存移行の下の分岐に表示されます (下図を参照してください)。

代替分岐作成後の グラフセ POU を下図に示します。

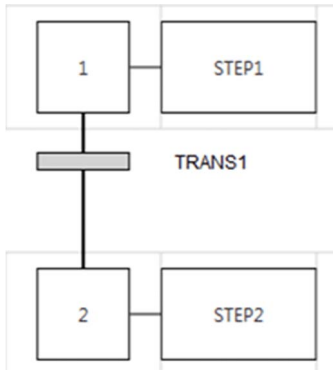


分岐した領域の前後の水平ラインは単線です。

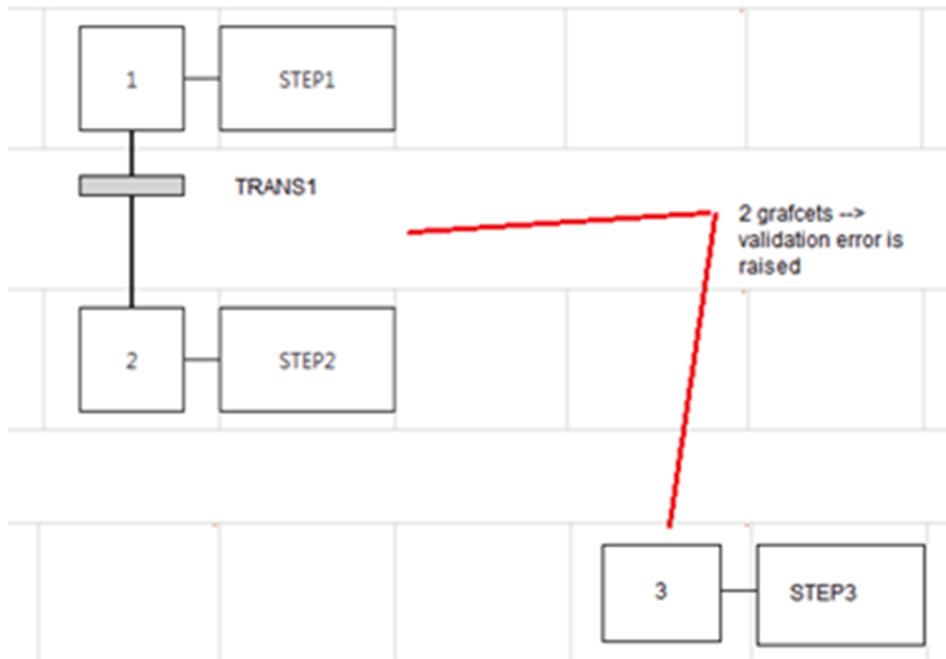
プログラミング推奨方法

グラフセ (SFC) ルール

- ステップは移行で接続します：



- 同じグラフセエディターの中では、1つのグラフセ POU しか追加できません。



クロスリンク

次の理由でクロスリンクを使用できます。

- 代替 (論理 OR) 分岐 (フォーク、またはジャンクション)
- セルのグリッドにスペースを保存します。線が交差する場合、線間の相互作用はなく、シンボリック表現にのみに使われます。

6.16

オンラインモードでのデバッグ

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
トレースウィンドウ	200
値の変更	202
値の強制	203
オンラインモードでの変更	204

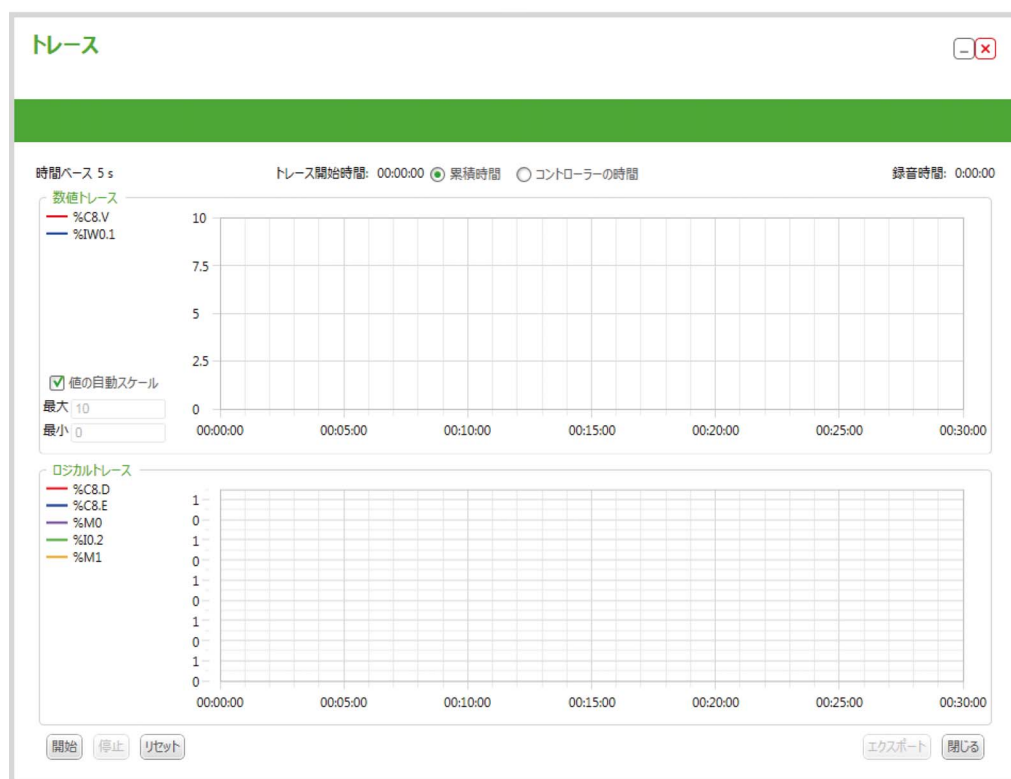
トレースウィンドウ

概要

トレースウィンドウでは、特定のアナログおよびデジタル変数の値をグラフ形式で表示できます（連続記録は 12 時間に制限されています）。各アニメーションテーブルは、常に 1 つのトレースを含むことができます。トレースには、最大 8 個のオブジェクトを追加できます。分析のために、データをファイルにエクスポートすることもできます。

注記：トレースに設定可能な最小リフレッシュ時間は 1 秒 (128 ページ参照) です。例えば、マスタートaskサイクル間のブール変数の値の変化はトレースできません。

トレースウィンドウ情報



累積時間を選択して、トレース開始時間を 00:00:00 に設定するかまたは**コントローラーの時間**を選択して、トレースの開始時間としてロジックコントローラーの時間と日付を使ってください。

トレースウィンドウには、アニメーションテーブルでトレース用に選択した各データタイプのグラフが個別に表示されます。

- **数値トレース領域**に、整数値および実数値が表示されます。
すべての数値がグラフ上の同じ目盛りに表示されます。
値の自動スケールを選択すると、すべての値が表示されるように自動的に垂直軸を調整します。そうでない場合、**最大**および**最小**の値を入力して、固定された範囲の値を表示します。
注記：**最大**および**最小**には、整数値または実数値を入力できます。
- **ロジカルトレース領域**に、2進数の値が表示されます。
2進数の値は、それぞれ個別に0と1の目盛りでトレースされます。

トレースの開始、一時停止およびリセット

開始をクリックすると、変数のトレースを開始します。

停止をクリックすると、リアルタイムトレースを一時停止します。

リセットをクリックすると、グラフから以前にトレースしたすべてのデータを削除し、**録音時間の値**を0にリセットします。

トレースをエクスポート

エクスポートをクリックすると、すべてのトレースされたデータがPCのファイルにエクスポートされます。

データはカンマ区切り値 (CSV) 形式で保存されます

値の変更

はじめに

オンラインモードのとき、SoMachine Basic で特定のオブジェクトタイプの値を変更することができます。

オブジェクトに読み込み / 書き込みアクセスがあるときにのみ、オンライン更新が可能です。

例：

- アナログ入力の値は変更できません。
- Timer ファンクションブロックの Preset パラメーター (%TM0.P オブジェクト) の値は更新可能です。

オブジェクトタイプによる読み込み / 書き込みアクセスの有無については、お使いのプラットフォームの SoMachine Basic *Generic Functions Library Guide*、または *プログラミングガイド* のオブジェクトの詳細を参照してください。

オブジェクトの値を変更するには、その値をアニメーションテーブル (130 ページ参照) に追加し、必要に応じてそのプロパティを設定します。

値の強制

概要

オンラインモードでは、特定のブール型オブジェクトの値を False (0)、または True (1) に強制することができます。アドレスを特定の値に設定し、プログラムロジック、または外部システムがその値を変更してしまうことを防ぐことができます。この機能はプログラムのデバッグや微調整に使われます。

オンラインモードのときにブール型オブジェクトの値を強制する方法を以下に示します。

- アニメーションテーブル (128 ページ参照) の使用
- ラダー (LD) 編集で直接ブール型オブジェクトの変更 (205 ページ参照)

以下のとき、デジタル入力、および出力は強制できません。

- 入力が運転 / 停止入力として使われているとき
- 高速カウンター (FC) 入力として設定されているとき
- 高速カウンター (HSC) 入力として設定されているとき
- 反射出力として設定されているとき

オンラインモードでの変更

概要

以下により、オフラインモードでもプログラムを変更することができます。

- ラングの追加 (204 ページ参照)
- ラングの変更 (204 ページ参照)
- ラダーのブール値の変更 (205 ページ参照)
- ファンクションブロックパラメーターの変更 (206 ページ参照)
- ワード型定数の変更 (207 ページ参照)
- 演算ブロックと比較ブロックのオブジェクト値の変更 (207 ページ参照)
- ラングの削除 (208 ページ参照)
- 変更の送信 (208 ページ参照)

変更後に、その変更をロジックコントローラーに送ります (208 ページ参照)。

ラングの追加

オンラインモード中に、プログラムに新規ラングを追加 (84 ページ参照) できます。

注記： オンラインモードで新規ラングを追加するには、アプリケーションがレベル 4.1 以上のファンクションレベル (77 ページ参照) で設定されている必要があります。

新規ラングがロジックコントローラーに正常に送信されるまで、以下の制限があります。

- エラー (✖) を含むラングはロジックコントローラーに送信できません。
- ラングはラダー言語で書かれ、正常にコンパイルされるまで IL へ変換することはできません。
- ラングにはグラフセ (リスト) ステップを含めることはできません。
- ラベルはラングに追加できません。

ラングの変更

オンラインモードでは、命令リスト (IL) およびラダー (LD) エディターの両方でプログラムのラングを変更できます。ただし、グラフセ (SFC) はオンラインでは使用できません。変更したラングは、オレンジ色の背景 (157 ページ参照) で表示されます。

ロジックコントローラーが RUNNING または STOPPED ステートであるかにより、実行できる変更のタイプおよび変更できる命令に制限があります。これらの制限は、コントローラーのステートとプログラムの整合性保護に役立ちます。

オンラインモードでも、命令リスト (IL) およびラダー (LD) 間のラング表示を切り替えることができます。

変更が可能な状態を次の表に示します。

操作	IL の STOPPED	IL の RUNNING	ラダーの STOPPED	ラダーの RUNNING
イベントタスク内容	可	不可	可	不可
マスター / 周期タスク内容	可	可	可	可
Free POU 内容	可	可	不可	不可
ラベル付きラング	可	不可	不可	不可
ラングと終了、ジャンプ、サブルーチンまたはラベルの呼び出し	不可	不可	不可	不可
ラングとグラフセ命令	不可	不可	不可	不可
ラベルの追加 / 変更	不可	不可	不可	不可

操作	IL の STOPPED	IL の RUNNING	ラダーの STOPPED	ラダーの RUNNING
複数のオペランド (演算ブロックと比較ブロック)	不可	不可	可	可

注記： この表では、オンラインモードでは許可されていないプログラム構造の変更は考慮していません。

ラダーのブール値の変更

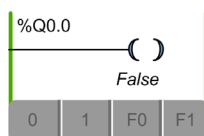
ラダー言語で表示されたラングの場合、特定のブール型オブジェクトタイプの値は 1/0 に書き込まれるか、1/0 に強制されるかまたは強制されません。

変更可能なブール型オブジェクトタイプを以下に示します。

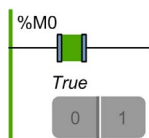
オブジェクトタイプ	1/0 の書き込み	1/0 に強制または強制しない
デジタル入力 (%Ix.y)	N/A	可
デジタル入力 (%Qx.y)	可	可
システムビット (%Si) ¹	可	N/A
ビットメモリー (%Mi)	可	N/A
ワードメモリービット (%MWi :Xj)	可	N/A
アナログ出力ビット (%QWi :Xj)	可	N/A
システムワードビット (%SWi :Xj) ¹	可	N/A
入力アセンブリビット (%QWEi :Xj)	可	N/A

¹ システムビットまたはシステムワードがユーザープログラムによって書き込まれる場合。

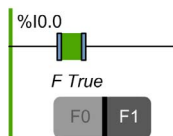
ラダーエディターのオブジェクト上にマウスカーソルを移動します。オブジェクトが 1/0 に書き込まれ、また 1/0 に強制される場合、グラフィック要素の下に以下のボタンが表示されます。



オブジェクトは 1/0 に書き込まれるが、強制はされない場合、以下のボタンが表示されます。



オブジェクトは 1/0 に強制はされるが、書き込まれない場合、以下のボタンが表示されます。



ボタンをクリックすると、オブジェクトのリアルタイム値が変更されます。

- 00 を書き込み
- 11 を書き込み
- F00 に強制
- F11 に強制

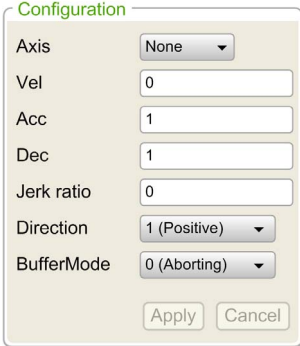
オブジェクトの現在のステータスに対応するボタンは濃いグレーで表示されます (上の例では F1)。

強制を解除するには、以下のいずれか行ってください。

- 再度 F0/F1 ボタンをクリックする。
- アニメーションテーブル (128 ページ参照) を使う。

ファンクションブロックパラメーター値の変更

オンラインモードでファンクションブロックのパラメーターを変更するには、以下の手順で行います。

手順	手順内容
1	<p>プログラミングウィンドウで、マウスカーソルをラダーエディタのファンクションブロックに移動します。</p> <p>結果: 設定ツールチップが表示されます。</p> <p>次の図は、設定ツールチップの例を示しています。</p> 
2	変更する値をクリックします。
3	値を入力します。
4	<p>作成した変更を検証するには、次のいずれかの方法を実行します。</p> <ul style="list-style-type: none"> ● 適用をクリックします。 ● 設定ツールチップの外側をクリックします。結果: 質問ウィンドウが表示されます。OK をクリックします。

ワード型定数の変更

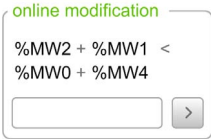

ワード型定数 (%KW) の設定値および実行データ値、ダブルワード型定数 (%KD) および浮動小数点型定数 (%KF) のオブジェクトはオンラインモードで変更が可能です。プロパティグリッドの **10 進数**、**2 進数**、**16 進数** および **ASCII** の列が変更可能です。

Constant word properties		%KW	%KD	%KF				
Used	Equ Used	Address	Symbol	Decimal	Binary	Hexadecimal	ASCII	
<input type="checkbox"/>	<input type="checkbox"/>	%KW0		0	2#00000000000000000000	16#0000	no meaning	
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%KW1		0	2#00000000000000000000	16#0000	no meaning	

オンラインモードでワード型定数または浮動小数点の値を変更するには、以下の手順で行います。

手順	手順内容
1	プログラミングウィンドウのツールタブのメモリーオブジェクト → ワード型定数を選択します。
2	変更する定数のタイプ、%KW、%KD または %KF を選択します。
3	必要に応じて値を変更します。 定数値をインポートできます。ワード型定数のプロパティのインポート (135 ページ参照) を参照してください。
4	適用をクリックします。 結果: 変更された値がロジックコントローラーに送信されます。

演算ブロックと比較ブロックのオブジェクト値の変更

手順	手順内容
1	プログラミングウィンドウで、マウスカーソルをラダーエディタの演算ブロックまたは比較ブロックに移動します。 結果: オンライン変更ツールチップが表示されます。 
2	変更するオブジェクトまたはシンボルをクリックします。
3	値を入力します。
4	検証するには、次のいずれかの方法を実行します。 <ul style="list-style-type: none"> ●  をクリックします。 ● Enter 押します。 間違った値を入力すると、値は変更されません。

ラングの削除

オンラインモード中に、プログラムからラングを削除できます。

注記： オンラインモードでラングを削除するには、アプリケーションが**レベル 4.1**以上のファンクションレベル (77 ページ参照) で設定されている必要があります。

以下の制限があります。

- ラングはラダー言語で表示されている必要があります。
- ラングは、POU または Free POU の中で、ラングのみにはできません。この制限は、グラフセ POU には適用されません。
- ラングはグラフセ (リスト) ステップを含んでいない、ラングはサブルーチンラングではないまたは
 - JMP
 - JMPC
 - JMPCN
 - END
 - ENDC
 - ENDCN
 - G7
- 一度に削除できるラングは、1 つのみです。

変更の送信

IL では、変更が許可されているとき、変更された IL ラインは検証後に自動的にロジックコントローラーに送られます。変更が許可されていない場合には、メッセージが表示されます。

ラダーでは、変更は自動的に送信されません。オンラインモードのときは、ボタンバーが表示されます。



送信をクリックすると、変更がロジックコントローラーに送信されます。このボタンはプログラムがオンラインモードで変更され、エラーがない場合のみ有効です。

ロールバックをクリックすると、オンラインモードでの変更が破棄され、元のラング (ロジックコントローラーに保存されているバージョン) が復元されます。ラングの背景がオレンジから緑に変わります。このボタンはプログラムがオンラインモードで変更された場合のみ有効です。

プログラム以外のデータをダウンロードをクリックすると、非プログラムデータ (プロジェクトプロパティ、シンボル、コメントおよびアニメーションテーブル等) の更新をロジックコントローラーにダウンロードします。このボタンは、非プログラムデータが PC とロジックコントローラー間で同期していない場合にのみ有効です。例えば、オンラインモードに入る前に、アニメーションテーブルが変更された場合など。

バックアップをクリックすると、ロジックコントローラーのフラッシュメモリーと RAM メモリーの内容を同期します。このステータスは、コントローラー情報ウィンドウ (227 ページ参照) に表示されます。バックアップ中は、進行中の Ethernet 通信 (例えば、Modbus TCP の使用または EXCH3 命令) が一時的に中断されます。

注記： 複製を作成する前には必ず、オンライン変更がフラッシュメモリーに保存されていることを確認します。

変更されたラングは、コントローラーが RUNNING ステートであるかまたは STOPPED ステートであるかの状況に応じた有効性を評価されます。ランタイムエラーを起こすまたはプログラムメモリーの構造が変わる変更は、オンラインモードでは拒否されます。

第 7 章

通信

この章について

この章には次のセクションが含まれています。

セクション	項目	参照ページ
7.1	通信ウィンドウの概要	210
7.2	ロジックコントローラーとの接続	211
7.3	コントローラーの更新	221
7.4	メモリー管理	222
7.5	コントローラー情報	227
7.6	RTC の管理	229

7.1

通信ウィンドウの概要

通信ウィンドウの概要

はじめに

通信ウィンドウでは以下のことが可能です。

- ロジックコントローラーのログイン、またはログアウト。
- ロジックコントローラーのファームウェアのアップグレード (、またはダウングレード)。
- ロジックコントローラーのメモリー管理 (例: バックアップ、および復元操作)。
- 接続しているロジックコントローラー、拡張モジュール (リファレンス、TM3 拡張モジュール、ファームウェアバージョン)、およびカートリッジに関する情報表示。
- ロジックコントローラーのリアルタイムクロック (RTC) の管理。

The screenshot shows a software interface for managing a logic controller. On the left, there is a table with the following data:

見つかりました:	
リファレンス	ファームウェア
コントローラ TM221CE40U	1.5.0.0
モジュール 1 TM3AI4-	-
モジュール 2 TM3AQ4-	-
カートリッジ TMC2AI2	
カートリッジ TMC2AQ2V	

In the center, a green-bordered box contains a green checkmark icon and the text: "PC とコントローラー アプリケーションは同一です 接続が確立されました".

On the right side, there are several buttons:

- PCからコントローラー (ダウンロード)
- コントローラーから PC (アップロード)
- コントローラーの停止
- コントローラーの開始
- シミュレーターを起動
- シミュレーターを停止

注記: TM3 アナログ拡張モジュールのファームウェアバージョンを表示するには、アプリケーションのファンクションレベル (77 ページ参照) を少なくともレベル 5.0 に設定する必要があります。

7.2

ロジックコントローラーとの接続

このセクションについて

このセクションには次の項目が含まれています。

項目	参照ページ
ロジックコントローラーとの接続	212
アプリケーションのダウンロードとアップロード	217

ロジックコントローラーとの接続

概要

通信ウィンドウの接続をクリックしてロジックコントローラーとの接続を管理します。

接続済みコントローラー

ロジックコントローラーは2つのリストに表示されます。

1. ローカルデバイス

PC に接続されたすべてのロジックコントローラーを表示します。

- PC の COM ポートで接続されているもの (例えば COM1)。
- USB ケーブルで接続されているもの。
- バーチャル COM ポート (USB - シリアル変換器、または Bluetooth ドングル使用) で接続されているもの。
- 手動で追加したモデムで接続されているもの。SoMachine Basic、およびロジックコントローラー間のモデム接続は監視のみに使用します。

注記 : COM ポート、および **Modbus ドライバーパラメーターを保持**のチェックボックスが選択されている場合、Modbus ドライバーで定義されたパラメーターで通信が確立されます。

2. Ethernet デバイス

Ethernet からアクセス可能なすべてのロジックコントローラーを表示します (同じサブネットにあり、ルーター下でない、または UDP ブロードキャストをブロックするデバイス上) このリストには手動で追加したコントローラーに加え、自動的に SoMachine Basic により検出されたロジックコントローラーも含まれます。

手動で Ethernet コントローラーを追加

ロジックコントローラーを手動で **Ethernet デバイス** リストに追加する手順を示します。

手順	手順内容
1	リモート参照フィールドに追加するロジックコントローラーの IP アドレス (例 : 12.123.134.21) を入力します。
2	追加をクリックして Ethernet デバイス リストにデバイスを追加します。


手動でモデム接続を追加

モデム使用要件

- モデムが PC にインストールされていない場合、このボタンは無効です。
- モデムがインストールされている Windows **コントロールパネルの電話、およびモデムオプション**を確認し、テストを実行します (**モデムタブ**でテストするモデムをクリックし、**プロパティ** → **診断** → **クエリーモデム**をクリックします)。モデムの応答は有効である必要があります。
- COM ポートに接続されている外部モデムの場合、下記の通信設定が同じか確認します。
 - モデムアドバンスドパラメーター
 - 通信ポートパラメーター
 - Modbus ドライバーパラメーター

SR2MOD03 モデムの設定、およびインストールに関する詳細は SR2MOD02 and SR2MOD03 Wireless Modem User Guide (10 ページ参照) を参照してください

モデム接続を手動で **Ethernet デバイス** リストに追加する手順を示します。




手順	手順内容
2	 (モデム設定の追加 ボタン) をクリックし モデム設定 ウィンドウを開きます。 結果 : モデム設定 ウィンドウが表示されます。
3	ドロップダウンリストからモデムの COM ポートを選択します。 
4	通信パラメーターを設定します。 モデム設定パラメーターの詳細情報を次の表に示します。
5	適用 をクリックします。 注記 : すべての設定が正しくされている場合のみ、このボタンが有効になります。 結果 : モデム接続は、 ローカルデバイス リストに追加されます (例 : COM2@0612345678、GenericModem) 。

モデム設定のプロパティを次の表に示します。

プロパティ	値	初期値	説明
ポート	COMx	-	ドロップダウンリストからモデムの COM ポートを選択します。
機器	-	-	モデム名が表示されます。
電話番号	-	-	ロジックコントローラーに接続しているモデムの電話番号を入力します。 このテキストフィールドは合計 32 文字以内のすべての文字を入力できます。このフィールドには、設定を適用するために文字が 1 つ以上含まれている必要があります。
AT init cmd	-	AT&D0	モデムの AT 初期化コマンドを編集できます。 AT 初期化コマンドは必須ではありません (フィールドが空の場合 AT スtringが送信されます)。
エスケープ文字	-	+	電話を切る手続用のエスケープ文字を編集できます。
ボーレート	1200 2400 4800 9600 19200 38400 57600 115200	19200	モデムのデータ転送レート (ビット / 秒) を選択できます。
パリティ	なし 偶数 奇数	偶数	エラー検出用に送信されるデータのパリティを選択できます。
データビット	7 8	8	データビットの数を選択できます。
ストップビット	1 2	1	ストップビットの数を選択できます。
タイムアウト (ms)	0...60000	15000	転送タイムアウト (ms) を指定できます。
Break タイムアウト (ms)	0...10000	10	インターフレームタイムアウト (ms) を指定できます。 自動 チェックボックスがチェックされている場合この値は自動的に計算されます。

ロジックコントローラーとの接続

ロジックコントローラーへログインする手順を示します。

手順	手順内容
1	 (デバイスを更新 ボタン) をクリックし接続している Ethernet デバイスのリストを更新します。
2	<p>ローカルデバイス、または Ethernet デバイス リストのロジックコントローラーを 1 つ選択します。</p> <p>コントローラーがご使用の PC と同じネットワークに Ethernet で接続されている場合、コントローラーの IP アドレスがリストされます。リストから IP アドレスを選択すると、 (IP アドレス設定 ボタン) が有効になります。ボタンをクリックしコントローラーの IP アドレスを変更します。</p> <p>注記： ポスト設定ファイルの書き込みチェックボックスを選択すると、ポスト設定ファイル内の Ethernet パラメーターが変更され、電源を切っても保持されます。</p>
3	<p>必要に応じて  (LED 点滅開始 ボタン) をクリックし選択したコントローラーの LED を点滅させ、コントローラーの識別に利用できます。ボタンを再度クリックし LED の点滅を停止します。</p> <p>注記： 自動検出プロトコルを有効オプションを選択して自動的に追加されたロジックコントローラーの LED 点滅開始 ボタンのみ使用できます。</p>
4	<p>ログインをクリックして選択したコントローラーにログインします。</p> <p>ロジックコントローラーがパスワード保護されている場合、パスワードの入力を求められます。パスワードを入力し OK ボタンをクリックして接続します。</p> <p>結果： 接続状況を示すステータスバーが表示されます。</p>
5	<p>接続すると、ロジックコントローラーに保存されているアプリケーションの保護ステータスが、ウィンドウの選択されたコントローラーエリアに表示されます。</p> <p>接続が正常に確立すると、ロジックコントローラーの詳細がウィンドウの選択されたコントローラーエリアに表示されます。</p> <ul style="list-style-type: none">● ファームウェアのリビジョン● ロジックコントローラーのリファレンス番号● ロジックコントローラーに接続しているすべての拡張モジュールのリファレンス番号● SoMachine Basic とロジックコントローラーの現在の接続状態。
6	<p>SoMachine Basic はロジックコントローラーのハードウェア設定と現在のプロジェクトの設定に互換性があるかどうかを検証します。</p> <p>互換性があればアプリケーションをコントローラーにダウンロードできます。PC からコントローラー (ダウンロード) ボタンが有効になり、アプリケーションのダウンロード (217 ページ参照) に進むことができます。</p> <p>SoMachine Basic は、ロジックコントローラーに格納されている非プログラムデータ (シンボル、コメント、アニメーションテーブルなど) が現在のアプリケーションと同じであるかどうかを確認します。異なる場合はアドバイザリーメッセージが表示されます。</p> <p>SoMachine Basic は最新のファームウェアバージョンが利用可能な場合、ファームウェアアップグレードを開始できるリンクを表示します。</p>


接続時にプロジェクトを比較

SoMachine Basic アプリケーションとロジックコントローラー内のアプリケーションを比較できます。差異が表示されるので、それらを評価、および考慮することができます。

ダウンロード、およびアップロードが許可され、PC とロジックコントローラーのアプリケーションが異なる場合、**通信**ウィンドウにメッセージが表示されます。



以下の手順に沿って進めます。

手順	手順内容
1	メッセージ内の PC とコントローラーのアプリケーションを比較 をクリックします。 結果 ：ポップアップウィンドウが表示され、比較結果の確認の前にロジックコントローラーの接続を切るように通知します。
2	OK をクリックし、ロジックコントローラーの接続を切ります。
3	アプリケーションの比較 ウィンドウが表示されます。  <p>The screenshot shows the 'Application Comparison' dialog box. It has a title bar 'アプリケーションの比較' and a close button. Below the title bar, there are three colored circles representing different comparison elements: green for 'PC 上でのみの要素' (Elements only on PC), blue for 'PC/コントローラー' (PC/Controller), and red for 'コントローラー上でのみの要素' (Elements only on controller). There are also radio buttons for 'ILコード' (IL code) comparison: '差分のみを表示' (Show only differences) and 'すべてを表示' (Show all). Below this, there are tabs for 'マスタータスク', '同期タスク', 'イベント', and 'Free POU'. The 'ILコード' section is expanded, showing a comparison between 'PC 上のILコード' and 'コントローラー上のILコード'. The PC code shows '1 POU' with 'ラング 0' containing LD, OR, and ST instructions, and 'ラング 1' containing BLK, LDR, R, LDN, AND, CU, and OUT instructions. The controller code is currently empty. At the bottom, there are expandable sections for 'ハードウェア設定' (Hardware settings), 'ソフトウェア設定' (Software settings), and 'アプリケーションの表示' (Application display). There are also buttons for 'Excelファイルへのエクスポート' (Export to Excel file) and '閉じる' (Close).</p>
4	Excel ファイルへのエクスポート をクリックしスプレッドシート形式で比較結果を保存します (オプション)。


設定とアプリケーションの次のエリアで比較結果の確認ができます。

- IL コード
- ハードウェア設定
- ソフトウェア設定
- アプリケーションの表示

アプリケーションのダウンロードとアップロード

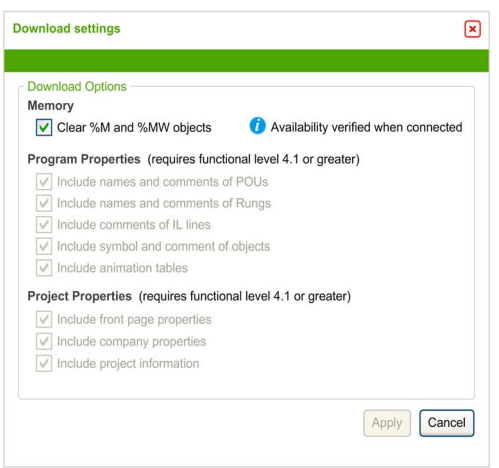
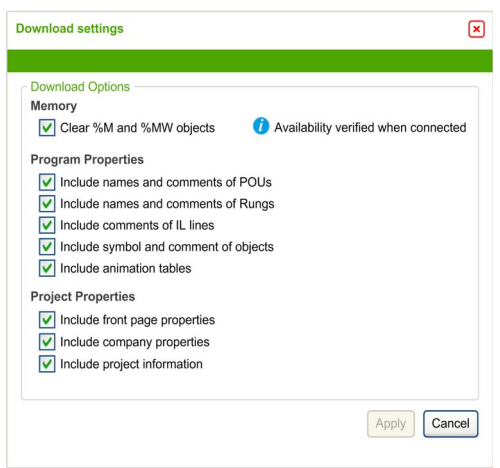
アプリケーションのダウンロード

次の手順で、SoMachine Basic で開いているアプリケーションをロジックコントローラーへダウンロードします。

手順	手順内容
1	通信ウィンドウの通信ツリーで 接続 をクリックします。
2	ローカルデバイス または Ethernet デバイス リストからロジックコントローラーを 1 つ選択します。
3	ログイン をクリックしてコントローラーにログインします。 ロジックコントローラーがパスワード保護されている場合、パスワードを入力し、 OK をクリックして接続します。
4	必要に応じて、  ダウンロード設定 をクリックします。 ダウンロード後、ワードメモリー (%MW) およびビットメモリー (%M) をリセットしたくない場合、 メモリーのリセット オプションを解除します。 注記 ：このメモリーオプションは、ファームウェアバージョン 1.3.3.y 以上のロジックコントローラーでのみ使用できます。 この プログラムプロパティ および プロジェクトプロパティ のオプションは、ファームウェアバージョン 1.4.1.y 以上のロジックコントローラーでのみ使用できます。
5	PC からコントローラー (ダウンロード) をクリックします。 PC からコントローラーへ (ダウンロード) ボタンが使用できない場合、以下を確認します。 <ul style="list-style-type: none">● ロジックコントローラーに格納されているアプリケーションが、SoMachine Basic アプリケーションと一致している。● ロジックコントローラーシステムのハードウェア設定が、SoMachine Basic アプリケーションの設定と互換性がない。
6	アプリケーションの非プログラムデータが、コントローラーに格納されているデータと異なる場合、非プログラムデータのみコントローラーにダウンロードされます。
7	アプリケーションが、 実行状態で開始 に設定されている場合、危険メッセージが表示され、アプリケーションの設定を確認する指示があります。 OK をクリックしてアプリケーションをダウンロードするか、 キャンセル をクリックして、設定を変更します。
8	OK をクリックして転送を続行し、ロジックコントローラーのアプリケーションを上書します。 結果 ：接続ステータスを示すステータスバーが表示されます。
9	ダウンロードしたアプリケーションを実行するには、 コントローラーの実行 をクリックした後、 OK をクリックします。 操作モードの変更ができないというメッセージが表示された場合、 閉じる をクリックし、ロジックコントローラーの運転/停止スイッチまたはコントローラーが RUNNING 状態になる妨げになっている可能性のある運転/停止入力を確認します。詳細は、ロジックコントローラーのハードウェアガイドを参照してください。

ダウンロードオプションの設定

ダウンロードオプションを表示するには、 **通信ウィンドウのダウンロード設定をクリック**します。

ファンクションレベル ≤ 4.1	ファンクションレベル ≥ 4.1
	
設定はアップロードされません	設定がアップロードされました。

初期設定では、各設定が選択された状態です。オンラインモードでオプションを選択または解除するには、**PC からコントローラーへ (ダウンロード)** をクリックして変更をダウンロードします。

オンラインモードでは、POU、ラングまたは IL ラインの名前またはコメントを変更し、**ダウンロード設定**でこれらに対応するオプションがで選択されている場合は、ダウンロードが自動的に行われます。

初期設定で、**メモリーのリセット**オプションは選択されています。このオプションは、オフラインおよびオンラインモードで使用できます。

メモリーのリセットが選択されている場合、アプリケーションがダウンロードされると、ワードメモリーおよびビットメモリーは 0 にリセットされます。

メモリーのリセットが選択されていない場合、ワードメモリーおよびビットメモリーの値は保持されます。

アプリケーションに割り当てられたメモリー容量が、ロジックコントローラーと PC で違う場合、メモリーは次のように管理されます。

- ロジックコントローラーのアプリケーションで割り当てられた %MWx が、PC メモリー上のアプリケーションで割り当てられた %MWx よりも大きい場合、PC での割り当てが使用され、追加ワード %MWx に、0 が設定されます。
- ロジックコントローラーのアプリケーションで割り当てられた %MWx が、PC メモリー上のアプリケーションで割り当てられた %MWx よりも小さい場合、追加ワード %MWx はメモリー領域から削除されます。
- ロジックコントローラーにアプリケーションがない場合、%MW に、0 が設定されます。同じルールが %M に適用されます。ダウンロード設定はプロジェクトに依存し、プロジェクトと共に保存されます。

アプリケーションのアップロード

次の手順で、ロジックコントローラーに格納されたアプリケーションを、SoMachine Basic にアップロードします。

手順	手順内容
1	通信ウィンドウの通信ツリーで 接続 をクリックします。
2	ローカルデバイス または Ethernet デバイス リストからロジックコントローラーを 1 つ選択します。
3	ログイン をクリックしてコントローラーにログインします。 ロジックコントローラーがパスワードで保護されている場合、パスワードを入力し、 OK をクリックして接続します。
4	コントローラーから PC へ (アップロード) をクリックします。 コントローラーから PC へ (アップロード) ボタンが使用できない場合、ロジックコントローラーに格納されているアプリケーションが、SoMachine Basic アプリケーションと一致していることを確認します。
5	OK をクリックして、ロジックコントローラーからアップロードします。 結果 ：接続ステータスを示すステータスバーが表示されます。転送が完了すると、ロジックコントローラーから、SoMachine Basic へアプリケーションをアップロードします。

注記：アプリケーションをアップロードした場合、**メモリーのリセット オプション**の値は保存されません。

パスワードで保護されたアプリケーションのダウンロードまたはアップロード

下位バージョンの SoMachine Basic のパスワードで保護されたアプリケーションをダウンロードまたはアップロードする場合、実行できるアクションまたは実行する必要があるアクションは、バージョンによって異なります。

操作	SoMachine Basic バージョン	アプリケーションのファンク ションレベル:	ファームウェア バージョン	アクション
ダウンロード				
	≤1.5	≤5.0	≤1.5	ダウンロードプロセスでは、最新のセキュリ ティー技術は使用されません。
			≥ 1.5.1	ダウンロードプロセスは可能ですが、パス ワードが可視状態になる可能性があります。
		≥ 5.1	–	ダウンロードできません。
	≥ 1.5 SP1	≤5.0	≤ 1.5	アプリケーションがパスワードで保護されて いる場合、ダウンロードできません。
			≥ 1.5.1	次のいずれかの方法を実行します。 ● ファンクションレベルを 5.0 にアップグ レードする。 ● パスワードを空白にする。 ● アプリケーション保護を無効にする。
		≥ 5.1	≤ 1.5	ダウンロードできません。
		≥ 1.5.1	ダウンロードプロセスでは、最新のセキュリ ティー技術が使用されます。	
アップロード				
	≤1.5	≤5.0	≤1.5	アップロードプロセスでは、最新のセキュリ ティー技術は使用されません。
			≥ 1.5.1	次のいずれかの方法を実行します。 ● ファームウェアのバージョンをダウング レードする。 ● SoMachine Basic バージョンをアップグ レードする。
		≥ 5.1	–	アップロードできません。
	≥ 1.5 SP1	≤5.0	≤1.5	アップロードプロセスでは、最新のセキュリ ティー技術は使用されません。
			≥ 1.5.1	アップロードプロセスでは、最新のセキュリ ティー技術が使用されます。
		≥ 5.1	≤1.5	アップロードできません。
		≥ 1.5.1	アップロードプロセスでは、最新のセキュリ ティー技術が使用されます。	

7.3

コントローラーの更新

コントローラーのファームウェアの更新

概要

ファームウェアの更新データをロジックコントローラーに、直接 SoMachine Basic から、または SD カードを使用してダウンロードできます。

ファームウェアの更新データをロジックコントローラーへダウンロード

ファームウェアの更新を実行すると、コントローラー内のアプリケーションプログラム (不揮発性メモリーのブートアプリケーションを含む) が保持されます。

次の手順で、ロジックコントローラーにファームウェア更新データをダウンロードします。

手順	手順内容
1	ファームウェア更新をする際は、ロジックコントローラーに接続していないことを確認します。
2	更新 → コントローラーの更新をクリックします。
3	更新をクリックします。 エグゼクティブローダー (OS ロード) のウィザードが表示されます。

アプリケーションの転送中にデバイスの電源を切ったり、停電または通信中断が起きた場合、デバイスが動作不能になる場合があります。通信中断または停電が起きた場合、転送を再試行してください。ファームウェアの更新中に停電または通信中断が起きた場合、または不正なファームウェアが使用されている場合、デバイスが動作不能になります。そのような場合有効なファームウェアを使用するかファームウェアの更新を再試行してください。

注記

装置の動作不能

- アプリケーションプログラムやファームウェア変更の転送が開始されたら中断しないでください。
- 転送が中断された場合、どんな理由であっても、転送をはじめからやり直してください。
- 転送が完了するまで、機器 (ロジックコントローラー、モーションコントローラー、HMI コントローラー、およびドライブ) の運転を開始しないでください。

上記の指示に従わないと、物的損害を負う可能性があります。

7.4 メモリー管理

ロジックコントローラーのメモリー管理

概要

SoMachine Basic では、接続しているロジックコントローラーの要素のバックアップ、復元および削除ができます。

バックアップ、復元および削除はオンラインモードでのみ利用可能です。

警告

装置の意図しない動作

- 消去、または復元操作を実行する前に、接続しているコントローラーが対象のターゲットであることを確認してください。
- リモートで消去、または復元操作を実行する前には、機器、または処理環境の安全状態を確認してください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

PC またはコントローラー SD カードへバックアップ

次の手順で、ロジックコントローラーのメモリーを PC またはコントローラー SD カードにバックアップします。

手順	手順内容
1	ロジックコントローラーへのログインします。
2	通信ウィンドウの左側にある メモリー管理 を選択します。
3	アクションリスト から コントローラーのバックアップ を選択します。
4	PC にバックアップする場合、 出力先 より PC を選択します。参照ボタンをクリックし、バックアップファイルを書き込むフォルダーを選択します。 または SD カードにバックアップする場合、 出力先 より コントローラー SD カード フォルダーを選択します。SD カードをロジックコントローラーの SD カードスロットに挿入します。 注記 : クローン作成またはスクリプトの実行を避けるため (<i>Modicon M221, ロジックコントローラー, プログラミングガイド</i> 参照) SD カードに script.cmd ファイルを入れたり、空にしないでください。
5	オプションを選択して、バックアップする要素を選択します。 <ul style="list-style-type: none"> ● ファームウェアのバックアップ ● プログラムのバックアップ ● メモリーの値をバックアップ ● ログファイルをバックアップ ● ポスト設定ファイルをバックアップ PC のバックアップで メモリーの値をバックアップ を選択している場合、 最初のメモリービット 、 最後のメモリービット 、 最初のワードメモリー および 最後のワードメモリー をバックアップに含めるかどうかを指定します。

手順	手順内容
6	<p>コントローラーのバックアップをクリックしてバックアップを開始します。</p> <p>指定された PC 上のフォルダーまたは SD カードに SD カードイメージ (.smbk) として、その要素が保存されます。</p> <p>レポートウィンドウが表示されバックアップに関する情報や検出されたエラーに関するメッセージが表示されます。</p>

注記：メモリー値のバックアップ選択した場合、ロジックコントローラーが RUNNING 状態のときにバックアップを開始することができます。ただし、バックアップに含まれるメモリー変数の数によっては、ロジックスキャン間でバックアップが実行されないことがあります。結果として、メモリー変数の値はスキャンごとに変更するため、バックアップは必ずしも一貫しているとは限りません。変数の値が一貫したセットが必要な場合、最初にロジックコントローラーを STOPPED 状態にしてください。

復元

次の手順で、PC からロジックコントローラーの要素を復元します。

手順	手順内容
1	ロジックコントローラーへのログインします。
2	通信 ウィンドウの左側にある メモリー管理 を選択します。
3	アクション リストから コントローラーの復旧 を選択します。
4	バックアップファイルを含むソースフォルダーを PC から選択します。
5	ロジックコントローラーに復元したい要素を選択します。
6	<p>コントローラーの復旧をクリックし復元を始めます。</p> <p>レポートウィンドウが表示され復元に関する情報や検出されたエラーに関するメッセージが表示されます。</p>

データファイル、アプリケーションファイル、およびファームウェアファイルなどのファイル転送が不完全な場合、機器、またはコントローラーに重大な影響を与える可能性があります。ファイル転送中に電源の切断、停電、または通信中断があった場合、機器が動作不能になる、またはアプリケーションが破損したデータファイルで動作する可能性があります。転送が中断した場合、再度転送してください。必ず破損したデータファイルの影響をリスク分析に含めてください。

警告

装置の意図しない動作、データ損失、またはファイルの破損

- 進行中のデータ転送を中断しないでください。
- 如何なる理由でも転送が中断した場合は、転送を初めからやりなおしてください。
- 破損したファイルのリスク分析がされ、ファイル転送の失敗による重大な影響がでることを防ぐための適切な措置をしている場合を除き、ファイル転送が完全に終わるまで機器を運用しないでください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

コントローラーの SD カードからバックアップを復元する場合は、ロジックコントローラーの **プログラミングガイド**を参照してください。

ロジックコントローラーの要素の消去

次の手順で、コンピューターからロジックコントローラーの要素を消去します。

手順	手順内容
1	通信ウィンドウの左側にある メモリー管理 を選択します。
2	アクションリスト から コントローラーの消去 を選択します。
3	ロジックコントローラーから消去したい要素を選択します。 ポスト設定ファイルの削除オプション を選択した場合、ポスト設定ファイルは コントローラーの消去 をクリックすると消去されます。現存する Ethernet 接続を保持するために、次に示す Ethernet の再初期化に伴うイベントの後にファイルが消去されます。 <ul style="list-style-type: none"> ● Ethernet ケーブルの抜き差し ● コントローラーの初期化 ● コントローラーの電源切断
4	コントローラーの消去 をクリックし消去を始めます。 レポートウィンドウが表示され消去に関する情報や検出されたエラーに関するメッセージが表示されます。

ロジックコントローラーイメージの作成と読み込み

ロジックコントローラーイメージには、コントローラーファームウェア、プログラムおよびポスト設定ファイルが含まれます。スクリプトを使用して、それらの要素をロジックコントローラーに転送できます。

ロジックコントローラーイメージを作成するとき出力先として SD カード を選択すると、ロジックコントローラーで SD カードを使用できるようになります。

ロジックコントローラーイメージの作成

オフラインモードで以下の処理によりスクリプトを生成し、次の要素を PC または SD カードにコピーするために必要なファイルをコピーできます。

- インストールした SoMachine Basic ソフトウェアに含まれるファームウェア
- 現在開かれているプロジェクトのプログラム
- ポスト設定ファイル

次の手順で、ロジックコントローラーイメージを作成します。

手順	手順内容
1	ロジックコントローラーに接続している場合は、 通信ウィンドウ のログアウトをクリックします。
2	通信ウィンドウの左側にある メモリー管理 を選択します。
3	アクションリスト から コントローラーイメージの作成 を選択します。
4	出力先 → PC の参照ボタンをクリックして、イメージファイルを書き込むフォルダーに移動します。 PC に接続されている SD カードを出力先として選択できます。
5	次の項目を選択して、コピーする要素を選択します。 <ul style="list-style-type: none"> ● ファームウェアを含める ● プログラムを含める
6	ポスト設定ファイルを上書きする場合は、 ポスト設定ファイルの消去 を選択します。
7	コントローラーイメージの作成 をクリックします。 結果 ：次のフォルダーとファイルが作成されます。 <ul style="list-style-type: none"> ● script.cmd ● usr/app/*.smbk ● sys/os/*.mfw
8	PC にコントローラーイメージを作成した場合、そのファイルを SD カードにコピーします。

次の図は、設定の例を示しています。

通信

- 接続
- コントローラーの更新
- メモリ管理**
- コントローラー情報
- RTC の管理

コントローラーのメモリ管理

アクション

- コントローラーのバックアップ
- コントローラーの復旧
- コントローラーの消去
- コントローラーイメージの作成
- イメージの読み込み

出力先

- コントローラー SD カード
- PC

含む: ファームウェア

含む: プログラム

含む: モーリーの値

最初のメモリビット 最後のメモリビット

最初のメモリワード 最後のメモリワード

含む: ログファイル

含む: ポスト設定ファイル

ロジックコントローラーイメージの読み込み

オフラインモードでは、この操作により SoMachine Basic の .smbk イメージファイルをプロジェクトとして開くことができます。

注記：開くイメージは、あらかじめ**コントローラーイメージの作成**または**コントローラーからバックアップ** (224 ページ参照) 操作で作成されたものです。

次の手順で、ロジックコントローラーイメージを読み込みます。

手順	手順内容
1	ロジックコントローラーに接続している場合は、 通信 ウィンドウのログアウトをクリックします。
2	通信 ウィンドウの左側にある メモリ管理 を選択します。
3	アクション リストから イメージの読み込み を選択します。
4	コピー元 → PC の参照ボタンをクリックして、イメージファイル (.smbk) のあるフォルダーに移動します。 プログラムの読み込み は初期設定として選択されています。画像ファイルを読み込むには、それを選択してください。
5	イメージの読み込み をクリックして、プログラムを読み込みプロジェクトを開きます。

次の図は、設定の例を示しています。

■ 通信

- 接続
- コントローラーの更新
- メモリー管理**
- コントローラー情報
- RTC の管理

コントローラーのメモリー管理

アクション

- コントローラーのバックアップ
- コントローラーの復旧
- コントローラーの消去
- コントローラーイメージの作成
- イメージの読み込み

コピー元

- コントローラー SD カード
- PC 

- 読み込み ファームウェア
- 読み込み プログラム
- 読み込み メモリーの値

最初のメモリービット 最後のメモリービット

最初のメモリーワード 最後のメモリーワード

- 読み込み ログファイル
- 読み込み ポスト設定ファイル

7.5

コントローラー情報

コントローラー情報

概要

通信ウィンドウの左側にある**コントローラー情報**をクリックすると、ロジックコントローラーの現在のステートに関する情報が表示されます。

- **実行可能な RAM:** このオプションは、有効なアプリケーションがロジックコントローラーの RAM メモリーに格納されているか確認します。システムワード %SW7 (*Modicon M221, ロジックコントローラー, プログラミングガイド参照*) のビット 14 を確認することで、プログラムからも取得できます。
- **保護された RAM:** ロジックコントローラーの RAM メモリーにあるアプリケーションがパスワード保護されている場合に、このオプションにチェックが入ります。システムワード %SW7 (*Modicon M221, ロジックコントローラー, プログラミングガイド参照*) のビット 8 を確認することで、プログラムからも取得できます。
- **強制された I/O:** ロジックコントローラーのデジタル入力、または出力が特定の値に強制 (130 ページ参照) されている場合に、このオプションにチェックが入ります。この場合、システムビット %S14 (*Modicon M221, ロジックコントローラー, プログラミングガイド参照*) (I/O 強制の有効) は 1 に設定されます。
- **RAM とフラッシュメモリーを同期:** 不揮発性メモリーに格納されているアプリケーションが RAM メモリーのアプリケーションと異なる場合、このオプションにチェックが入ります。以下のいずれかの場合、このオプションにはチェックが入りません。
 - オンラインでのアプリケーションへの変更がまだロジックコントローラーに送信されていない場合 (プログラミングタブの**バックアップ**ボタンをクリックしていません)。
 - 変更が行われてから、ロジックコントローラーが初期化されていません (ツールバーの**コントローラーの初期化**ボタンをクリックしていません)。
- **ステータス:** ロジックコントローラーの現在のステート。
システムワード %SW6 を確認することで、プログラムからも取得できます。コントローラーのステートについては、ロジックコントローラーの**プログラミングガイド**を参照してください。
- **最終停止箇所:** ロジックコントローラーが最後に停止したときの日時 (STOPPED、HALTED など)。
システムワード %SW54 から %SW57 を確認することで、プログラムからも取得できます。
- **最終停止理由:** ロジックコントローラーの最後の停止理由を表示します。
これは、システムワード %SW58 を確認することで、プログラムからも取得できます。
- **スキャンタイム (µs):** 以下のスキャンタイムがあります。
 - **最小 (ミリ秒):** ロジックコントローラーの電源が最後に入ってから、最短のスキャン時間。
システムワード %SW32 を確認することで、プログラムからも取得できます (ミリセカンド)。
 - **現在値 (ミリ秒):** 現在のスキャン時間
システムワード %SW30 を確認することで、プログラムからも取得できます (ミリセカンド)。
 - **最大 (ミリ秒):** ロジックコントローラーの電源が最後に入ってから、最長のスキャン時間。
システムワード %SW31 を確認することで、プログラムからも取得できます (ミリセカンド)。

- **コントローラーの時間** : ロジックコントローラーにリアルタイムクロック (RTC) があるときのみ、以下が表示されます。
 - **日付 (DD/MM/YYYY)** : ロジックコントローラーに格納されている現在の日付。
システムワード %SW56、および %SW57 を確認することで、プログラムからも取得できます。
 - **時間 (HH:MM:SS)** : ロジックコントローラーに格納されている現在の時間。
システムワード %SW54、および %SW55 を確認することで、プログラムからも取得できます。日付と時刻は、PC に指定されたものと同じ書式で表示されます。
- **Ethernet 情報** : ロジックコントローラーに標準 Ethernet 接続があるときのみ、以下が表示されます。
 - **IP アドレス** : ロジックコントローラーの IP アドレス。
これは、システムワード (*Modicon M221, ロジックコントローラー, プログラミングガイド 参照*) %SW33、および %SW34 を確認することで、プログラムからも取得できます。
 - **サブネットマスク** : ロジックコントローラーのサブネットマスク。
システムワード %SW35、および %SW36 を確認することで、プログラムからも取得できます。
 - **ゲートウェイアドレス** : ロジックコントローラーのゲートウェイアドレス。
システムワード %SW37、および %SW38 を確認することで、プログラムからも取得できます。
- **SL1 のポスト設定ステータス** : チェックボックスが有効になっているパラメーターは、ポスト設定ファイルで定義されます。システムワード %SW98 (*Modicon M221, ロジックコントローラー, プログラミングガイド 参照*) を確認することで、プログラムからも取得できます。
- **SL2 のポスト設定ステータス** : チェックボックスが有効になっているパラメーターは、ポスト設定ファイルで定義されます。システムワード %SW99 (*Modicon M221, ロジックコントローラー, プログラミングガイド 参照*) を確認することで、プログラムからも取得できます。
- **ETH のポスト設定ステータス** : チェックボックスが有効になっているパラメーターは、ポスト設定ファイルで定義されます。システムワード %SW100 (*Modicon M221, ロジックコントローラー, プログラミングガイド 参照*) を確認することで、プログラムからも取得できます。

7.6

RTC の管理

RTC の管理

概要

RTC 管理ウィンドウでロジックコントローラーのリアルタイムクロック (RTC) を設定できます。この設定は、SoMachine Basic が RTC に対応しているロジックコントローラーに接続されているときのみ可能です。

RTC の更新

手順	手順内容
1	通信ウィンドウの左側にある RTC 管理 オプションを選択します。
2	オンラインモードの場合、 現在のコントローラーの時刻 が表示されます。 ロジックコントローラーの時間設定のモードを選択します。 <ul style="list-style-type: none">● 手動: このモードでは、日時が表示され、手動でロジックコントローラーの日付と時間を設定できます。● 自動: このモードでは、ロジックコントローラーの時間が SoMachine Basic がインストールされている PC の現在時刻に設定されます。
3	適用 をクリックします。

第 8 章

シミュレーター

この章について

この章には次の項目が含まれています。

項目	参照ページ
SoMachine Basic シミュレーターの概要	232
SoMachine Basic シミュレーター I/O マネージャーウィンドウ	234
SoMachine Basic シミュレーター時間管理ウィンドウ	236
SoMachine Basic シミュレーターを使った値の変更	238
SoMachine Basic シミュレーターの使い方	242
Vijeo-Designer でのシミュレーターの起動	243

SoMachine Basic シミュレーターの概要

はじめに

SoMachine Basic シミュレーターでは、次のことが実行できます。

- PC、ロジックコントローラー、および拡張モジュール間の接続シミュレーション。
- PC に接続されるロジックコントローラーや拡張モジュールなしの、プログラムの実行やテスト。


シミュレーターは、ロジックコントローラーの動作を複製した SoMachine Basic の仮想ロジックコントローラーです。

注記：シミュレーターを使用すると、セキュリティパラメーター (*Modicon M221, ロジックコントローラー, プログラミングガイド 参照*) は、適用されません。

シミュレーターを起動すると、ロジックコントローラーに実際に接続しているときに行える、接続、実行、停止、およびその他の操作を行うことができます。

注記：シミュレーターは、最大 2 つの接続に対応しています。1 つは、SoMachine Basic 用で、もう 1 つは、データ用です (例えば、HMI 接続)。

SoMachine Basic シミュレーターへのアクセス

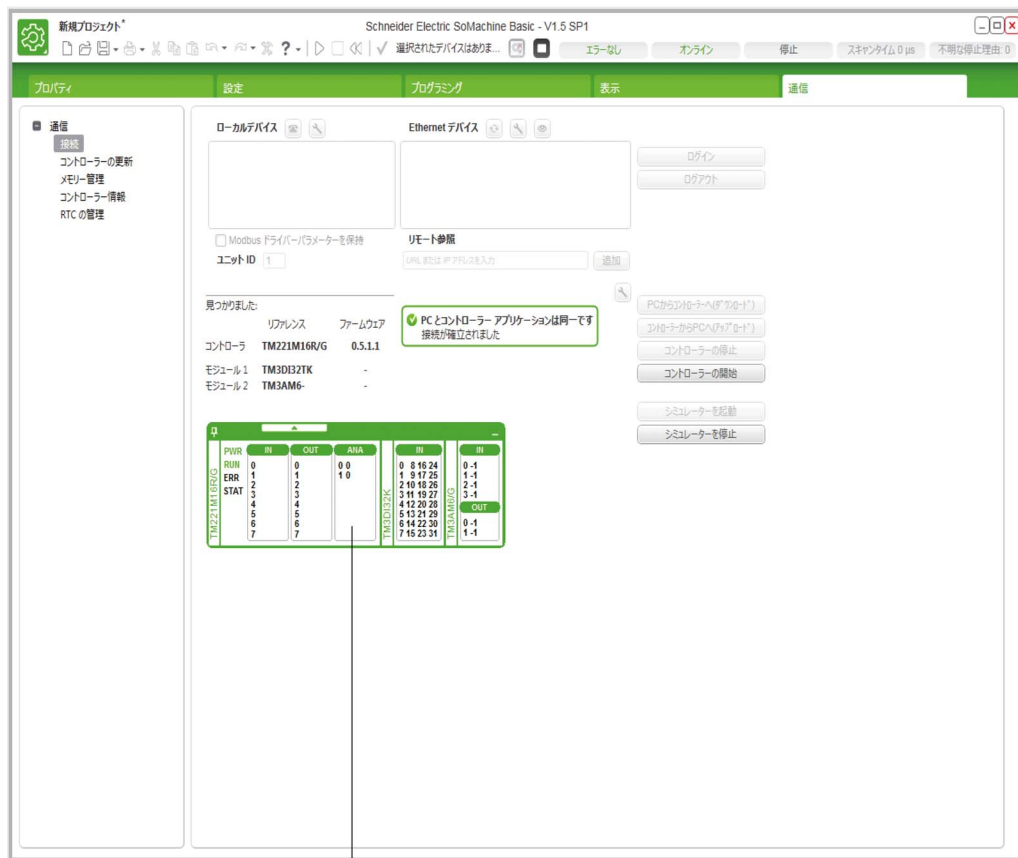
手順	手順内容
1	プログラムが有効であることを確認します。有効でない場合、コンパイルエラーが検出され、シミュレーターが中断、画面上にメッセージが表示されます。
2	次のいずれかの方法で、シミュレーターを起動します。 <ul style="list-style-type: none"> ● 通信タスクのシミュレーターを起動をクリックします。 ● 通信ウィンドウで、CTRL+B を押します。 ● SoMachine Basic のツールバーの  (シミュレーターを起動ボタン) をクリックします。

SoMachine Basic シミュレーターウィンドウ

SoMachine Basic のシミュレーターには、次の 2 つのウィンドウがあります。

- **シミュレーター時間管理ウィンドウ**
コントローラーの RTC を制御して、時間の経過と RTC の変化に対するロジックコントローラーへの影響をシミュレーションできます。
- **シミュレーター I/O マネージャーウィンドウ**
拡張モジュールとコントローラーの入力 / 出力のステートを管理できます。

PC と仮想ロジックコントローラーの接続が確立されると、SoMachine Basic シミュレーター I/O マネージャーウィンドウが画面に表示されます (SoMachine Basic シミュレーターの使い方 (242 ページ参照) を参照してください)。



1 シミュレーター I/O マネージャーウィンドウ (234 ページ参照)

SoMachine Basic シミュレーター I/O マネージャーウィンドウ

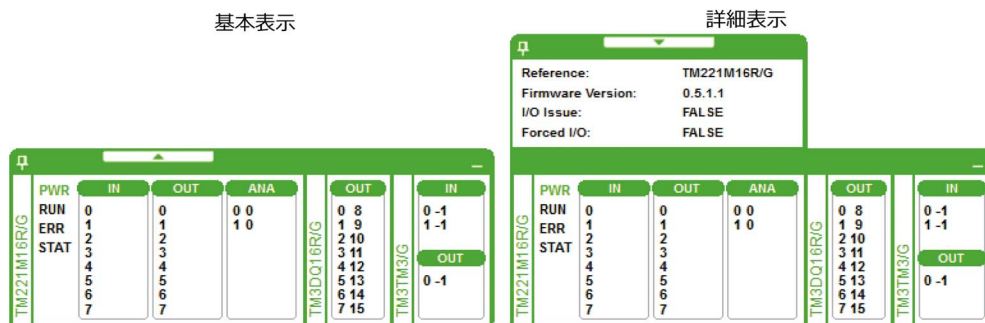
概要

シミュレーター I/O マネージャーウィンドウには以下のコンポーネントがあります。

- LED ステータス
シミュレーションされたコントローラーの LED ステータスを監視します。
- 入力 / 出力 ステータス
プログラム実行中の入力、および出力を制御します。

シミュレーター I/O マネージャーウィンドウ

以下の図はシミュレーター I/O マネージャーウィンドウです。



ウィンドウの左上部にあるピン記号をクリックすると、ウィンドウが前面に固定、または固定が解除されます。

ウィンドウの右上部にある最小化記号をクリックすると、タスクバーのウィンドウが最小化されます。

LED ステータス

PWR、RUN、ERR、および STAT LED は接続されたベースコントローラーに表示されるように、SoMachine Basic シミュレーター I/O マネージャーウィンドウでシミュレーションされます。

シミュレーションされたロジックコントローラーのシミュレーター I/O マネージャーウィンドウに表示される LED ステータスを以下に示します。

LED	ステータス情報
PWR	シミュレーションされたロジックコントローラーの電源が入っているかを示します。
RUN	シミュレーションされたロジックコントローラーの RUN ステートを示します。
ERR	シミュレーションされたロジックコントローラー ERR ステートを示します。
STAT	STAT LED の操作はユーザーロジックによって定義されます。

入力 / 出力 ステータス

シミュレーター I/O マネージャーウィンドウではプログラム実行中のコントローラー、および拡張モジュールの I/O を制御、および監視できます。

入力、および出力は、番号の一覧で表示されます。この一覧は、選択したコントローラー、および拡張モジュールの I/O によって異なります。例えば、コントローラーに n 点のデジタル入力がある場合、番号一覧は $0 \dots (n-1)$ になります。各番号はそれぞれの入力チャンネルに対応しています。

コントローラーで表示される I/O は以下になります。

- **IN**: デジタル入力
- **OUT**: デジタル出力
- **ANA**: アナログ入力

拡張モジュールで表示される I/O は以下になります。

- **IN**: デジタル / アナログ 入力
- **OUT**: デジタル / アナログ 出力

注記 : アナログ I/O はアナログ入力番号の右側に現在値とともに表示されます。

デジタル I/O ステータスは、I/O 番号の文字の色によって識別されます。

- 緑: I/O は 1 に設定
- 黒: I/O は 0 に設定

アナログ I/O ステータスは、以下の値によって識別されます。

- - (ハイフン): I/O は設定されていません。
- 数字: I/O の現在値

SoMachine Basic シミュレーター時間管理ウィンドウ

概要

シミュレーター **Time Management** ウィンドウには、次のコンポーネントがあります。

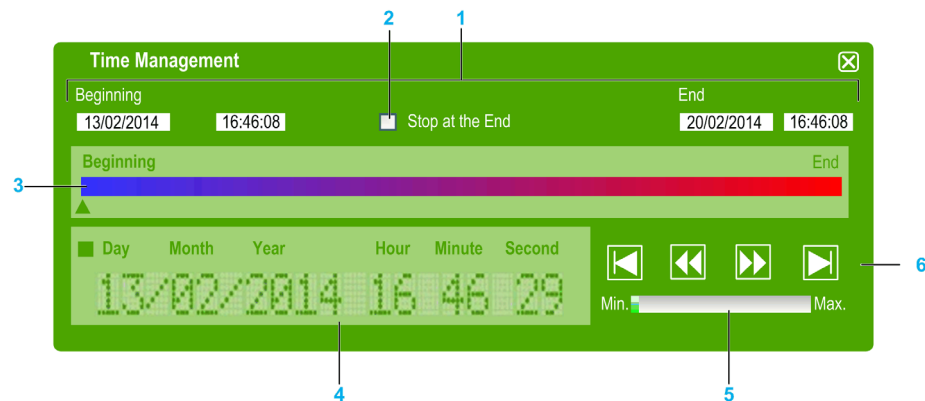
- シミュレーターでプログラムを実行するための、日付 / 時間のシミュレーション範囲
 - **Beginning** 日時
 - **End** 日時
 - **Stop at the End** チェックボックス (End 日時になるとプログラムの実行を停止します)。
- 時間制御スクロールバー
 - 時間の経過のシミュレーションを手動で前後に移動します
- 日付と時間の表示
 - シミュレーションされた RTC の日付と時間
- 制御ボタン
 - RTC に関連する時間管理を、前方へジャンプ、後方へジャンプ、リセット、または停止します
- インクリメントバー
 - リアルタイムに対して、シミュレーションされる時間の経過速度を変更します

次の手順で、シミュレーター Time Management ウィンドウ

Time Management ウィンドウを表示します。

手順	手順内容
1	シミュレーター I/O マネジャーウィンドウの上部のバーで、右クリックします。
2	Time Management を選択します。

シミュレーター Time Management ウィンドウ



- 1 日付 / 時間シミュレーション範囲 (Beginning – End)
- 2 Stop at the end (日付 / 時間の範囲) チェックボックス
- 3 時間制御スクロールバー
- 4 RTC の日付と時間
- 5 インクリメントバー
- 6 経過時間制御ボタン

シミュレーター、日付 / 時間シミュレーションの範囲

シミュレーション範囲を使用すると、シミュレーターの RTC の設定と制御ができます。シミュレーターを RUN ステートにすると、RTC に **Beginning** 日時が設定されます。**End** 日時には、シミュレーションの終了を設定します。**Stop at the End** チェックボックスを選択すると、シミュレーション範囲の終了時にシミュレーターを STOP ステートにします。チェックを外すと、SoMachine Basic のシミュレーターを手動で停止するまで、シミュレーターと RTC は実行し続けます。

時間制御スクロールバー

時間制御スクロールバーを使用すると、シミュレーション範囲の日付と時間を手動で操作できます。バーの下の矢印をポイントし、マウスの右ボタンをクリックしたまま右に移動すると、RTC の時間と日付が進みます。同じようにマウスを左に移動すると、RTC の時間と日付が戻ります。

RTC の日付と時間





RTC の日付と時間ゾーンには、進行中のシミュレーションに付随する RTC の値が表示されます。シミュレーションが RUN ステートになると、RTC の初期時間に **Beginning** 日時が設定されます。それ以降、シミュレーターで進行中の RTC の値が表示されます。RTC は、時間制御スクロールバー、または時間経過速度制御ボタンで変更できます。

インクリメントバー

インクリメントバーでは、経過時間制御ボタンを使用して RTC の値を前後へジャンプするために、相対的な増分を設定できます。バーをクリックすると設定したシミュレーション範囲を基準にしたインクリメントの設定ができます。

経過時間制御ボタン

経過時間制御ボタンを使用して RTC の値を変更すると、次のようにシミュレーターで実行中のプログラムを操作することができます。

グラフィック要素	コマンド	説明
	初期化	開始日時フィールドに設定されている日付と時間に戻ります。
	進む	インクリメントバーで設定された現在の日付と時間を進めます。
	戻る	インクリメントバーで設定された現在の日付と時間を戻します。
	終了	終了日時フィールドに設定されている日付と時間に移動します。

SoMachine Basic シミュレーターを使った値の変更

概要

オンラインモードでは、SoMachine Basic シミュレーター I/O マネージャーウィンドウで以下を行うことができます。

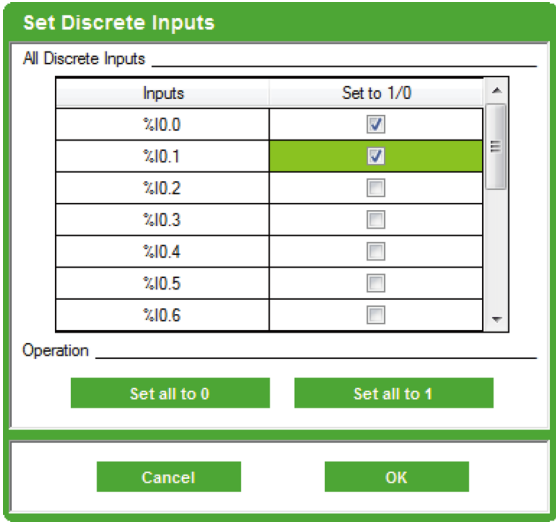
- 入力値の変更
- 出力のトレース

デジタル入力値の変更

次の手順で、ワンクリック操作によるデジタル入力値を変更します。

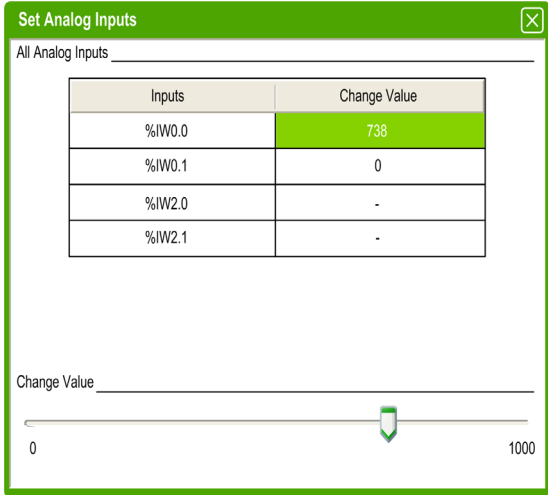
手順	手順内容
1	シミュレーター I/O マネージャーウィンドウのデジタル入力番号をクリックして、離散入力値を変更します。 結果 ：入力番号の色が変わります。デジタル入力値は、文字の色で識別されます。 <ul style="list-style-type: none"> ● 緑：I/O は 1 に設定 ● 黒：I/O は 0 に設定
2	同じ入力番号を再度クリックすると値が切り替わります。

次の手順で、デジタル入力値を一括操作でまとめて変更します。

手順	手順内容
1	シミュレーター I/O マネージャーウィンドウのデジタル入力番号をダブルクリックします。 結果 ：画面に、すべてのデジタル入力が一覧になっている Set Discrete Inputs ウィンドウが表示されます。 
2	Set Discrete Inputs ウィンドウの Operation で、以下をクリックします。 <ul style="list-style-type: none"> ● Set all to 0: すべての入力値を 0 に設定します。 ● Set all to 1: すべての入力値を 1 に設定します。 結果 ：チェックボックスが選択されている場合、入力値が 1 に設定されます。選択されていない場合、入力値は 0 に設定されます。
3	または、 Set Discrete Inputs ウィンドウの All Discrete Inputs で、入力に対応するチェックボックスをクリックして値を個別に変更します。
4	OK をクリックして、変更を保存、および Set Discrete Inputs ウィンドウを終了します。

アナログ入力の I/O 値の変更

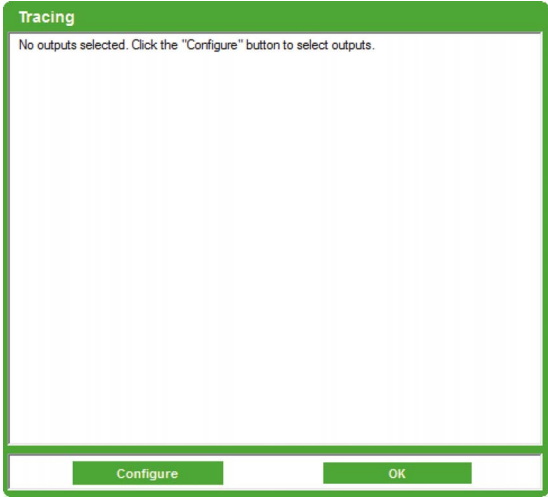
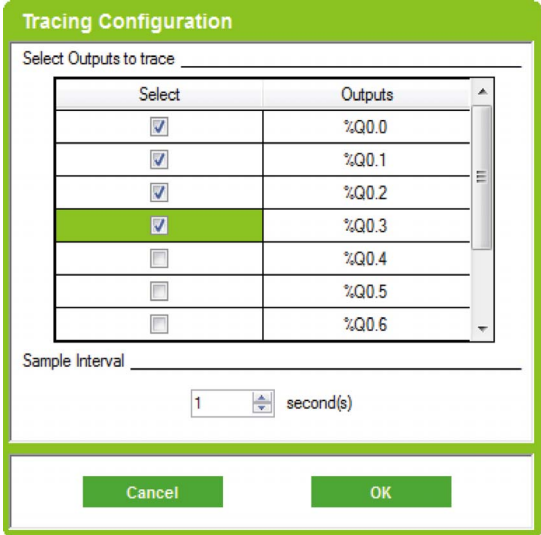
次の手順で、アナログ入力値を変更します。


手順	手順内容
1	<p>シミュレーター I/O マネージャーウィンドウのアナログ入力番号をダブルクリックします。 結果：画面に、すべてのアナログ入力が一覧になっている Set Analog Inputs ウィンドウが表示されます。</p> 
2	<p>Set Analog Inputs ウィンドウの All Analog Inputs で、変更する入力の Change Value フィールドをダブルクリックします。</p>
3	<p>値を 0...1023 の範囲で入力し、ENTER キーを押します。</p>
4	<p>または、Set Analog Inputs ウィンドウの Inputs 一覧から入力を選択し、Change Value にあるスライダーを動かして入力値を 0...1023 に調整します。 スライダーを左から右に動かすと値が増加し、左に動かすと減少します。</p>
5	<p>OK をクリックして変更を保存し、Set Analog Inputs ウィンドウを終了します。</p>

出力のトレース

出力値はプログラムによって異なりますので、値は変更することができませんが、SoMachine Basic シミュレーターでデジタル、およびアナログ出力をトレースすることができます。

次の手順で、アナログ入力値を変更します。

手順	手順内容
1	<p>シミュレーター I/O マネージャーウィンドウの出力番号をダブルクリックします。 結果: 画面に Tracing ウィンドウが表示されます。</p> 
2	<p>Configure ボタンをクリックして、トレースする出力を選択します。 結果: 画面に Tracing Configuration ウィンドウが表示されます。</p> 
3	<p>Select チェックボックス列の、トレースする出力に対応するチェックボックスをクリックします。</p>

手順	手順内容
4	ドロップダウンメニューから Sample Interval を選択し、トレースする出力のサンプル時間間隔を設定します。 <ul style="list-style-type: none"> ● 1 秒 ● 5 秒 ● 10 秒 ● 20 秒
5	<p>Ok をクリックして保存し、Tracing Configuration ウィンドウを終了します。</p> <p>結果：シミュレーションされた出力値のトレースが表示される Tracing ウィンドウに、選択した出力が追加されます。</p> 
6	<p>OK をクリックして Tracing ウィンドウを終了します。</p>

SoMachine Basic シミュレーターの使い方

手順

次の手順で、プログラムをテストするための SoMachine Basic シミュレーターを起動します。

手順	手順内容
1	ステータスエリアのステータスメッセージをチェックして、有効なプログラムがあることを確認します (詳細については、ステータスエリア (47 ページ参照) を参照してください)。プログラムのステータスは エラーなし にします。 プログラムのステータスが アドバイス の場合、SoMachine Basic シミュレーターを実行できます。
2	シミュレーターを起動します (シミュレーターへのアクセス (232 ページ参照) を参照してください)。
3	コントローラーを実行します。 通信 ウィンドウで通信ツリーの 接続 を選択し、通信タスクの コントローラーの実行 ボタンをクリックします。
4	シミュレーターのメインウィンドウで、プログラムを制御します (制御ボタン (237 ページ参照) を参照してください)。
5	シミュレーターのメインウィンドウで、LED ステータスを確認します (LED 表示 (235 ページ参照) を参照してください)。
6	シミュレーター I/O マネジャーウィンドウで、入力 / 出力ステータスを確認をします (入力 / 出力ステータス (235 ページ参照) を参照してください)。
7	シミュレーター I/O マネジャーウィンドウで、LED ステータスを確認します (LED ステータス (234 ページ参照) を参照してください)。
8	必要に応じて、I/O 値の変更します (シミュレーターを使った値の変更 (238 ページ参照) を参照してください)。
9	必要に応じて、出力をトレースします (出力のトレース (240 ページ参照) を参照してください)。
10	コントローラーを停止します。 通信 ウィンドウで、通信ツリーの 接続 を選択し、通信タスクの コントローラーの停止 ボタンをクリックします。
11	シミュレーターを停止します。 通信 ウィンドウで、通信ツリーの 接続 を選択し、通信タスクの コントローラーの停止 ボタンをクリック、またはシミュレーターで CTRL+W を押して終了します。

Vijeo-Designer でのシミュレーターの起動

手順

Vijeo-Designer の HMI シミュレーターを起動する前に、SoMachine Basic (232 ページ参照) のロジックコントローラーシミュレーターを起動します。

次の手順で、Vijeo-Designer のシミュレーターを起動します。

手順	手順内容
1	Vijeo-Designer を開始します。
2	SoMachine Basic プロジェクトのシンボルを含む Vijeo-Designer プロジェクトを開きます。 注記: Vijeo-Designer プロジェクトが存在しない場合には、Vijeo-Designer でプロジェクトを作成し、SoMachine Basic プロジェクトとシンボルを共有します。詳細は、SoMachine Basic プロジェクトと Vijeo-Designer プロジェクト間のシンボルの共有 (148 ページ参照) を参照してください。
3	Navigator ウィンドウの Project タブをクリックし、 IO Manager の下の機器を右クリックして Configuration を選択します。 結果: Equipment Configuration ウィンドウが開きます。
4	IP Address を入力し、 OK をクリックします。 注記: IP アドレスは、ローカルホストアドレス、または PC のローカルアドレスにします。 例: 127.0.0.1
5	Device Simulation Tool を起動します。
6	Variables タブをクリックし、シミュレーションする変数のチェックボックスを選択します。 注記: View All アイコンを選択すると、 Simulation タブで選択されている Variables タブのすべての変数が表示されます。
7	Simulation タブをクリックします。
8	変数を選択し、変数に対する操作を選択してから、 Active チェックボックスを選択します。 注記: 1 回につき、変数に適用できるシミュレーション操作は 1 つのみです。
9	変数シミュレーション操作のパラメーターを定義します。
10	Simulation アイコンをクリックして、シミュレーションを開始します。
11	シミュレーション中、必要に応じて変数の値を変更します。 <ul style="list-style-type: none">● スライダー操作では、スライダーを動かしたり、マウスのホイールを動かしたり、キーボードの矢印キーを押すことによって値を変更することができます。● 切り替え操作では、Set、または Reset をクリックして変数に対応する文字列を書き込みます。
12	Simulation アイコンを再度クリックして、シミュレーションを停止します。
13	CTRL+Z を押して、 Device Simulation Too を終了します。

第 9 章

プロジェクトの保存と SoMachine Basic の終了

この章について

この章には次の項目が含まれています。

項目	参照ページ
プロジェクトの保存	246
プロジェクトをテンプレートとして保存	247
SoMachine Basic の終了	248


プロジェクトの保存

概要


SoMachine Basic プロジェクトはローカル PC 上にファイルとして保存できます。ファイルの拡張子は *.smbp で、以下のものを含まます。

- プログラムのソースコード。 **プログラミング**タブにあります。
- 現在のハードウェア設定。 **設定**タブにあります。
- SoMachine Basic プロジェクト内の設定とオプション設定。

プロジェクトの保存

手順	手順内容
1	ツールバー上の 保存  をクリック、または Ctrl-S を押します。
2	プロジェクトを保存するのが初めての場合、プロジェクトファイルを保存するフォルダを選択します。
3	プロジェクトファイルの名前を入力し、 保存 をクリックします。

名前を付けてプロジェクトを保存


手順	手順内容
1	ツールバー上の 保存 ボタン  をクリックし、 名前を付けて保存 を選択します。
2	プロジェクトファイルを保存するフォルダを選択します。
3	プロジェクトファイルの名前を入力し、 保存 をクリックします。

プロジェクトをテンプレートとして保存

概要

SoMachine Basic プロジェクトはテンプレートとして保存できます。テンプレートとして保存したプロジェクトは開始ページ (39 ページ参照) のテンプレートタブにリスト表示されます。そのプロジェクトを新規プロジェクトの起点として使用できます。

プロジェクトをテンプレートとして保存

手順	手順内容
1	ツールバーの保存ボタン  の隣にあるメニュー矢印をクリックし、テンプレートとして保存を選択します。 初期設定ではテンプレートは次のフォルダーに保存されます。 C:\Users\Public\SoMachine Basic\Examples
2	プロジェクト名を入力します。
3	ファイルタイプとしてサンプルプロジェクトファイル (*.smbe) を選択し、保存をクリックします。

SoMachine Basic の終了

概要

SoMachine Basic を終了するには、SoMachine Basic ウィンドウの右上の**閉じる**ボタンをクリックします。

開始ページウィンドウの**終了**ボタンをクリックして終了することもできます。

付録



付録について

付録には次の章が含まれています。

章	章タイトル	参照ページ
A	Twido プロジェクトを SoMachine Basic に変換します	251
B	SoMachine Basic ショートカットキー	261

付録 A

Twido プロジェクトを SoMachine Basic に変換します

Twido プロジェクトを変換 SoMachine Basic

概要

TwidoSoft または TwidoSuite (32 ページ参照) プロジェクトを開くと、自動的に SoMachine Basic プロジェクトに変換されます。SoMachine Basic の同等機能に自動的に変換できなかったものすべては変換レポートにリストアップされます。

変換処理情報の詳細を次に示します。

警告

装置の意図しない動作

- アプリケーションプログラムが変換前と同じように動作し、必要とされる正しい設定、パラメーター、パラメーターの値、ファンクション、およびファンクションブロックがあることを必ず確認してください。
- 以前の動作と一致するように、必要に応じてアプリケーションを変更してください。
- アプリケーションを運用する前に、新しくコンパイルされたバージョンを十分にテスト、および検証してください。

上記の指示に従わないと、死亡、重傷、または物的損害を負う可能性があります。

手動で適応が必要な Twido プログラムタイプ

M221 Logic Controller に同等機能が存在しない機能を含む Twido プロジェクトを SoMachine Basic プロジェクトに変換するための対処方法を、プログラムタイプ別に次の表に示します。

Twido プログラムタイプ	対処方法	詳細
CANopen を使用したプログラム	Ethernet ネットワークを使用するプログラムへの変換を検討してください。	M221_with_LXM32_Modbus_TCP と M221_with_ILx2T_Modbus_TCP テンプレート (テンプレートウィンドウで検索を実行します (39 ページ参照)) を参照してください。
Twido Macro Comm を使用したプログラム	Twido コードは EXCH 命令を使用するように自動的に変換されません。通信ファンクションブロック (SoMachine Basic, Generic Functions Library Guide 参照) を使用するプログラムへの変更を検討してください。	通信ファンクションブロックを使用するプログラムの変更方法については、xSample_twido_macro_COMM_Conversion プロジェクトテンプレートと関連するマニュアル (テンプレートウィンドウで検索を実行します (39 ページ参照)) を参照してください。
Twido drive macros を使用したプログラム	Twido コードの一部はラダー言語コードに自動変換できません。	ドライバー管理機能の適応方法については、xSample_ATV Modbus SL_M221 または xSample_Twido_Macro_Drive_Conversion プロジェクトテンプレートを参照してください。

Twido プログラムタイプ	対処方法	詳細
Twido Extreme TWDLEDCK1 プロジェクト	このプロジェクトタイプは自動変換できません。	プログラムの一部を取得するには、 <ul style="list-style-type: none"> ● TwidoSuite プロジェクトのコントローラーを TWDLEDCK1 から別の Twido コントローラーへ 変更します。 ● 更新プロジェクトを変換します
Remote Link プロトコルを使用したプログラム	M221 Logic Controller の次の機能を使用するプログラムの変更を検討してください。 <ul style="list-style-type: none"> ● Ethernet ネットワーク上の Modbus TCP マッピング ● 通信ファンクションブロックを使用した Modbus シリアルプロトコル (<i>SoMachine Basic</i>, <i>Generic Functions Library Guide</i> 参照) 	The Remote Link プロトコルは、シリアルラインのリモート I/O モジュールと同じように Twido コントローラーを使用することができます。

変換レポートに表示されるメッセージ

変換レポートで参照する特定のメッセージ ID に関する追加情報を次の表に示します。

メッセージ ID	メッセージ	説明 / 対処方法
エラーメッセージ		
TC-001	Twido プロジェクトを読み込めません。	Twido プロジェクトファイルを SoMachine Basic で開くことができませんでした。
TC-002	Twido 情報のあるフォルダー (.xpr ファイルと同じ名前と同じ場所) が見つかりませんでした。	指定したフォルダーが見つかりませんでした。
TC-003	Twido ファイル <filename> のフォーマットが正しくありません。	Twido プロジェクトのフォーマットが正しくないため、変換できません。
TC-004	Twido ファイル <filename> はフォーマットが違います。	Twido プロジェクトは不完全です。変換できません。
TC-005	デバイス <device> は対応していません。	Twido 型式 <device> は対応していません。変換できません。
TC-006	CANopen マクロは IL に変換されていません。	M221 Logic Controller は CANopen に対応していないため Twido CANopen マクロも対応されません。
アドバイザーメッセージ		
TC-101	シリアルライン 2 の物理媒体は RS485 に変更されています。	TM221M*** 型式では、シリアルライン 2 を RS232 で設定することができません。代わりに外部デバイスで RS485 を設定することを検討します。または外付けの RS232/RS485 アダプターを追加する、ロジックコントローラーを TM221C 型式に置き換えるまたは RS232 をサポートする TMC2 カートリッジを追加します。

メッセージ ID	メッセージ	説明 / 対処方法
TC-102	シリアルラインのリモートリンク設定は Modbus プロトコルに変更されています。	Remote Link プロトコルは M221 Logic Controller では対応していません。 Modbus の通信ファンクションブロックを使用するまたは Ethernet 付き M221 Logic Controller を使用している場合は Modbus マッピングテーブルを使用するなどで解決できます。手動で適応が必要な Twido プログラムタイプ (251 ページ参照) も参照してください。
TC-103	TWDXCPODC 拡張モジュールは SoMachine Basic では対応していません。インポートされませんでした。	TWDXCPODC は SoMachine Basic では対応していない表示器用の拡張モジュールです。 M221 Logic Controller 用にオペレーターインターフェイスのアプリケーション提供している TMH2GDB リモートグラフィック表示を使用できます。
TC-104	TWDXCPODM 拡張モジュールは SoMachine Basic では対応していません。インポートされませんでした。	TWDXCPODM は SoMachine Basic では対応していない表示器用の拡張モジュールです。 M221 Logic Controller 用にオペレーターインターフェイスのアプリケーション提供している TMH2GDB リモートグラフィック表示を使用できます。
TC-105	新規ロジックコントローラーはパルス (PLS) またはパルス幅変調 (PWM) に対応していません。	Twido Compact Base 40 I/O、240 Vac コントローラーには 2 つのトランジスタ高速出力がありました。 M221 Logic Controller では 24 Vdc 電源のコントローラーにのみトランジスタ出力があります。M221 Vac 電源のコントローラーにはリレー出力のみがあります。コントローラーを置き換える場合、24 Vdc 電源の M221 Logic Controller を選択します。
TC-106	CANopen 通信拡張モジュールは SoMachine Basic では対応していません。インポートされませんでした。	CANopen は M221 Logic Controller では対応していません。 CANopen が必要な場合は、Modicon M241 Logic Controller を使用してください。または通信バスを、シリアルラインでは Modbus に、Ethernet では Modbus TCP に置きかえてください。
TC-107	AS-Interface マスター拡張モジュールは SoMachine Basic では対応していません。インポートされませんでした。	M221 Logic Controller AS-Interface マスターモジュールはありません。 Ethernet-AS Interface ゲートウェイの使用。または、Modbus シリアル、Modbus TCP または EtherNet/IP でリモート I/O を使用することを検討してください。
TC-108	TM200 HSC 拡張モジュールは SoMachine Basic では対応していません。インポートされませんでした。	M221 Logic Controller は高速カウンター (HSC) で使用できる 4 つの高速入力があります。
TC-109	TWD PTO 拡張モジュールは SoMachine Basic では対応していません。インポートされませんでした。	リレー出力なしの M221 Logic Controller はパルス列出力で使用できる 2 つまたは 4 つの高速出力があります。
TC-110	TM2 VCM 拡張モジュールは SoMachine Basic では対応していません。インポートされませんでした。	TM2 VCM 拡張モジュールは SoMachine Basic では対応していません。
TC-111	Timer 3 “Adjustable” パラメーターは SoMachine Basic では対応していません。常に True に設定されます	SoMachine Basic ファンクションブロックはこのパラメーターは対応していません。

メッセージ ID	メッセージ	説明 / 対処方法
TC-112	%QA ASi 出力は SoMachine Basic では対応していません。	これらのアドレスは AS-Interface リモート I/O の管理用に確保されています。AS-Interface マスターモジュールと同様、これらのアドレスは M221 Logic Controller では対応していません。
TC-113	PID の自動調整は変更されました。PID 自動調整 (AT) タブに新規パラメーター AT トリガが追加設定されています。パラメーター '出力セットポイント' は無視されます。	SoMachine Basic で PID 自動調整設定をします。
TC-114	HSCn (Twido では : VFCn) で使われている入力は <input1> から <input2> に変更されました。	割り当てられている入力がプログラムで使われているか確認します。
TC-115	HSCn で使用されている入力 (<input1> と <input2>) は、Twido VFCn と逆の順序になっています。	Twido VFC コントローラーの HSC 入力 <input1> と <input2> は SoMachine Basic で逆の順序になっています。入力の順序をアプリケーションで戻してください。
TC-116	Free POU <x> はすでに <y> に割り当てられています。<z> イベントはこの Free POU では使用できません。	Free POU は他のイベントに割り当ててください。
TC-117	新規コントローラーで <x> Twido オブジェクトが <y> に移動されました。整合性を維持するため、プログラムを更新してください。	オブジェクトは同等の機能をもつ SoMachine Basic オブジェクトに変換されました。<y> はシステムビット、システムワードまたは %IWS のようなオブジェクトタイプにできます。
TC-118	新規コントローラーで <x> Twido オブジェクトが変更されました。プログラムと整合しているか確認してください。	オブジェクトは変換されましたが SoMachine Basic での機能は異なる場合があります。プログラムを変更するには、オンラインヘルプを参照してください。
TC-119	新規コントローラーで <x> Twido オブジェクトは対応していません。新規ファンクションを使用して、プログラムを更新してください。	SoMachine Basic でオブジェクトと同等の機能はありません。プログラムを変更するには、オンラインヘルプを参照してください。
TC-120	ソースコントローラーは 24 Vdc 供給されていますが、目的のコントローラー <型式> は 100...240 Vac が供給されています。	変換された M221 Logic Controller は同じ電源ではありませんが、アプリケーションには影響ありません。
TC-121	トランジスタと出力リレー付きソースコントローラー <型式 1> は出力リレー付き <型式 2> に変更されています。	変換された M221 Logic Controller 同じ出力タイプではありません。変換してもアプリケーションは変更されません。
TC-122	<y> に関連付けられたシンボル <x> の構成が無効です。	指定したシンボルの構文を修正します。
TC-123	<y> に関連付けられたシンボル '<x>' は予約ワードのため、 '<z>' に変更されています。	SoMachine Basic には TwidoSuite/TwidoSoft にはない新規命令があります。予約ワードは <z> に変換されました。
TC-124	<x> で時間ベースで設定されている <w> は <y> に変更されています。アプリケーションの <w>.<z> と設定のプリセットを調整してください。	このメッセージは PLS または PWM を使用しているアプリケーションを変換する場合に発生します。Twido ではハードウェアタイムベースは 0.142 ms と 0.57 ms です。M221 Logic Controller ではハードウェアタイムベースはそれぞれ 0.1 ms と 1 ms です。PLS と PWM ファンクションブロックの信号の周期は、プリセット値 (PLS.P、PLS.PD または PWM.P) を乗算したタイムベースです。プリセット値 (.P または .PD) は設定とプログラムの両方で調整する必要があります場合があります。

メッセージ ID	メッセージ	説明 / 対処方法
TC-125	<y> で設定されている <x> は <z> に変更されています。	M221 Logic Controller はダウンカウントモードで HSC に対応していません。SoMachine Basic では標準カウントモード (アップカウント) に設定を変換しました。
TC-126	<x> の複数の閾値に同じ値を設定できないため変更されています。	SoMachine Basic では複数の閾値に同じ値を設定できません。Twido アプリケーションが閾値 (関連付けられたイベントや反射の設定がない) を使用しない場合、値は設定エラーを避けるために変更されます。
TC-127	<x> の複数の閾値が同じ値だと、エラーになります。	SoMachine Basic では複数の閾値に同じ値を設定できません。Twido アプリケーションが閾値を使用している場合、何も変更されず、結果として設定エラーになります。エラーを修正するためにアプリケーションを変更します。
TC-128	<x> は SoMachine Basic の Twido プロジェクトで運転 / 停止とイベントトリガーが設定されているため、運転 / 停止機能の設定が解除されています。	SoMachine Basic では同時に 2 つの異なる機能を同じ入力に設定できません。
TC-129	Ethernet モジュールが標準 Ethernet ポートの Twido 型式で検出されました。Ethernet モジュール設定は無視されます。	SoMachine Basic では 2 つの Ethernet リンクは使用できません。
TC-130	Twido マクロはサブルーチンから呼び出しできません。SRn ラング <x> から呼び出されるマクロは変換されていません。	SoMachine Basic ではマクロをサブルーチンから呼び出しできません。
TC-131	すべてのイベント優先順位を変換できません。手動調整が必要です。	変換処理ですべてのイベント優先順位を設定できませんでした。
TC-132	マクロ <macro> を変換できません: 最大数のサブルーチンが使用されています。	Twido プロジェクトはすでに最大数のサブルーチンが使用されており、Free POU に変換されました。マクロ変換処理は追加の Free POU が必要な場合があります。
TC-133	Twido アプリケーションのパスワードは大文字で入力するしてください。	Twido パスワードは、TwidoSuite または TwidoSoft によって大文字で保存されています。
メッセージ情報		
TC-201	コントローラー <reference1> は <reference2> に変更されています。	SoMachine Basic で、デフォルト設定のコントローラーで置き換えをしました。必要な特性と一致しない場合、他の型式に置き換えてください。
TC-202	モジュール <reference1> は <reference2> に変更されています。	SoMachine Basic は TM2 モジュールを同等の TM3 モジュールに変換します。
TC-203	Ethernet モジュールが検出されました。コントローラーは、Ethernet ポート以上の基準に変換されています。	499TWD01100 モジュールが TwidoSuite で設定されている場合、標準 Ethernet ポート付きの M221 Logic Controller リファレンスを選択します。
TC-204	NAC シリアルラインオプションが検出されました。シリアルラインカートリッジが設定に追加されました。	シリアルラインカートリッジ TMC2SL1 は Compact Twido に 3 つある TWDNAC シリアルアダプターの 1 つで対応しています。設定とケーブルの接続を確認します。
TC-205	NOZ シリアルラインオプションが検出されました。SL2 で設定されています。	シリアルラインカートリッジ TMC2SL1 は モジュール Twido に 3 つある TWDNOZ シリアル拡張モジュールの 1 つで対応しています。設定とケーブルの接続を確認します。

メッセージ ID	メッセージ	説明 / 対処方法
TC-206	<device> は汎用モデムに変更されました。	TD-33/V90 モデムは SoMachine Basic では対応していません。
TC-207	SL2 で設定された <device> は削除されました。SL1 モデムのみ承認されています。	M221 Logic Controller のシリアルライン SL2 でモデムを設定することはできません。モデムをシリアルライン SL1 に追加します。
TC-208	プロジェクトのファンクションレベルは <x> に設定されました。	指定したファンクションレベルがロジックコントローラーの機能設定に対応しているか確認します。
TC-209	<x> の優先順位は <y> から <z> へ変更されています。	イベントに割り当てられた優先度レベルを確認します。
TC-210	POU <y> のマクロ <x> - ラング <z> は同等のコード POU <a> - ラング に変更されています。	変換されたコードの機能を確認します。
TC-211	POU <y> のマクロ <x> - ラング <z> は同等のコード Free POU <a> に変更されています。	変換された Free POU のコード機能を確認します。

システムビット

M221 Logic Controller では対応していないまたは変更されている Twido システムビットを次の表に示します。

Twido システムビット	詳細	M221 Logic Controller システムビット	詳細
%S8	配線試験	非対応	M221 Logic Controller では対応していません
%S24	操作画面を固定にします	非対応	リモートグラフィック表示器で対応しています
%S25	操作表示器で表示モードの選択	非対応	リモートグラフィック表示器で対応しています
%S26	表示で符号付きまたは符号なしの値を選択	非対応	リモートグラフィック表示器で対応しています
%S31	イベントのマスク	非対応	M221 Logic Controller では対応していません
%S66	Brick 40 で BAT LED 表示の有効 / 無効	非対応	M221 Logic Controller では対応していません
%S69	ユーザー STAT LED 表示	非対応	M221 Logic Controller にはユーザー STAT LED はありません。
%S95	ワードメモリーの復元	%S94 で対応	非揮発性メモリーに保存されているデータを復元するには、このビットを 1 に設定します。
%S97	%MW の保存	%S92 で対応	不揮発性メモリーに保存されている変数 %MW
%S100	TwidoSuite 通信ケーブルの接続	非対応	M221 Logic Controller は USB ケーブルを使用します
%S110	リモートリンクの交換	変更	シリアルライン 1 の Modbus Serial IOScanner がリセットされます。
%S111	単一リモートリンクの交換	変更	シリアルライン 2 の Modbus Serial IOScanner がリセットされます。
%S112	リモートリンクの接続	非対応	M221 Logic Controller にはリモートリンク機能に対応していません
%S113	リモートリンクの設定 / 操作	変更	シリアルライン 1 の Modbus Serial IOScanner がリセットされます。
%S118	リモート I/O エラー	非対応	M221 Logic Controller にはリモートリンク機能に対応していません
%S120	入力 PWM0 オーバーフロー (%IW0.7) (Twido Extreme)	非対応	M221 Logic Controller には PWM の入力はありません
%S121	入力 PWM1 オーバーフロー (%IW0.8) (Twido Extreme)	非対応	M221 Logic Controller には PWM の入力はありません

詳細はシステムビット %S を参照してください。

システムワード

M221 Logic Controller には同等のものがないまたは異なる目的で使われている Twido システムワードを次の表に示します。

Twido システムワード	詳細	M221 Logic Controller システムワード	詳細
%SW6	コントローラーのステータス	変更	コントローラーステート
%SW7	コントローラーステート	変更	コントローラーのステータス
%SW20...%SW27	CANopen スレーブモジュールのステータス	非対応	CANopen バスは M221 Logic Controller では使用できません
%SW49...%SW53	RTC ファンクション: 日付と時間の値 (BCD 形式) を含むワード	変更	RTC ファンクション: 日付と時間の値 (BCD 形式) を含むワード
%SW58	停止原因をコードで表示	変更	停止原因をコードで表示します
%SW59	現在の日付調整	変更	現在の日付調整
%SW60	RTC 補正值	非対応	RTC 補正值は使えません。
%SW67	コントローラーのファンクションとタイプ	変更	コントローラーの機能とタイプ
%SW68	2 ライン操作表示器に表示される要素	非対応	M221 Logic Controller に表示器は内蔵されていません。リモートグラフィック表示器で対応しています。
%SW69	2 ライン操作表示器に表示される要素	非対応	M221 Logic Controller に表示器は内蔵されていません。リモートグラフィック表示器で対応しています。
%SW73	AS-Interface システムステート	非対応	ASI バスは M221 Logic Controller では使用できません。
%SW74	AS-Interface システムステート	非対応	ASI バスは M221 Logic Controller では使用できません。
%SW80	Base I/O ステータス	変更	標準アナログ入力ステータス
%SW81...%SW87	I/O 拡張モジュールの 1 から 7 のステータス	%IWS、%QWS へ移動	アナログ入力またはアナログ出力ステータス用システムオブジェクト
%SW96	アプリケーションプログラムと %MW の保存 / 復元ファンクション用のコマンドおよび診断	変更	プログラムおよび %MW の保存 / 復元機能の診断 (詳細は保持変数 (Modicon M221, ロジックコントローラー, プログラミングガイド 参照) を参照してください)。
%SW96:X0	不揮発性メモリーに保存するワードメモリーの指定	%S93	%SW96:X0 を M221 Logic Controller に書き込むことはできません。プログラムで %SW96:X0 を %S93 に置換します。 システムビット %S95 を %S94 と置き換え、%S97 を %S92 と置き換えます。システムワード %SW97 を %SW48 と置き換えます。 システムワード %SW96 の他のビットの使用を確認します。
%SW97	保存 / 復元機能のコマンドまたは診断	%SW148 に対応	保持変数の数 (最大 2000 個の変数)
%SW111	リモートリンクのステータス	非対応	M221 Logic Controller にはリモートリンク機能に対応していません
%SW112	リモートリンクの設定 / 操作エラーコード	非対応	M221 Logic Controller にはリモートリンク機能に対応していません

Twido システムワード	詳細	M221 Logic Controller システムワード	詳細
%SW113	リモートリンクの設定	非対応	M221 Logic Controller にはリモートリンク機能に対応していません

詳細は システムワード %SW (Modicon M221, ロジックコントローラー, プログラミングガイド参照) 参照してください。

付録 B

SoMachine Basic ショートカットキー

SoMachine Basic キーボードショートカット

キーボードショートカッターリスト

修飾キー	キー	コマンド	ビュー	条件
CTRL	C	コピー	テキストボックス	-
CTRL	V	貼り付け	テキストボックス	-
CTRL	X	切り取り	テキストボックス	-
ALT	Left	前のタブに移動	すべて	-
ALT	Right	次のタブに移動	すべて	-
	F1	ヘルプまたはコンテキストヘルプの表示	すべて	システム設定 → 一般で選択
SHIFT	F1			
ALT	F4	SoMachine Basic の終了	すべて	-
CTRL	B	シミュレーターを起動	すべて	-
CTRL	G	ログイン	すべて	-
CTRL	H	ログアウト	すべて	-
CTRL	L	コントローラーの停止	すべて	-
CTRL	M	コントローラーの実行	すべて	-
CTRL	N	新規プロジェクト	すべて	-
CTRL	O	プロジェクトを開く	すべて	-
CTRL	P	プロジェクトレポートの印刷	すべて	-
CTRL	Q	SoMachine Basic の終了	すべて	-
CTRL	S	プロジェクトの保存	すべて	-
CTRL	W	シミュレーターを停止	すべて	-
CTRL	J	ダウンロード	通信	-
CTRL	K	アップロード	通信	-
	ALT	ラダーショートカットの表示	プログラミング	-
	Del	削除	プログラミング	項目が選択されている
CTRL	D	プログラムのすべてのラングをラダーに変換	プログラミング	-
CTRL+ALT	D	プログラムのすべてのラングをILに変換	プログラミング	-
CTRL	F	検索	プログラミング	-
CTRL	I	選択したラングの前に新規ラングを挿入	プログラミング	-
CTRL	Y	やり直し	プログラミング	-
CTRL	Z	元に戻す	プログラミング	-

修飾キー	キー	コマンド	ビュー	条件
CTRL	矢印キー	ラインの描画	ラダーラング	描画ツールが選択されている
CTRL	矢印キー	ラインの削除	ラダーラング	消去ツールが選択されている
CTRL	矢印キー	次のラダーセルの選択 / 解除 (セルごと)	ラダーラング	選択ツールが選択されている
SHIFT	矢印キー	次のラダーセルの選択 / 解除 (領域で選択)	ラダーラング	選択ツールが選択されている
	ESC	選択ツールへのポインターをリセット	ラダーラング	選択されたツールが配線の描画、または消去ではないか、ドラッグされている項目はない、または開かれているポップアップがない
	ESC	保留中のラインをキャンセル	ラダーラング	描画中
	ESC	ラインの消去をキャンセル	ラダーラング	消去中
	ESC	選択した項目の移動をキャンセル (初期位置に戻す)	ラダーラング	ラダー項目がドラッグされている
	ESC	提案一覧を閉じる	ラダーラング	提案一覧が開いている (接点の有効な記述子等)
	ESC	ラダーツールバーのメニュー項目を閉じる	ラダーラング	ラダーツールバーのメニューが開いている (ファンクションブロック等)
	ENTER	ラダー要素移動の開始 / 停止	ラダーラング	1 つ以上のセルが選択されている
	矢印キー	浮動セルの移動	ラダーラング	セルの移動が開始されている
	矢印キー	現在のセルを変更	ラダーラング	初期設定
	F5	A 接点を開く	ラダーラング	Asian set 1 ラダーツールバー
	F6	分岐開始	ラダーラング	Asian set 1 ラダーツールバー
SHIFT	F5	A 接点を閉じる	ラダーラング	Asian set 1 ラダーツールバー
SHIFT	F6	分岐終了	ラダーラング	Asian set 1 ラダーツールバー
	F7	コイル	ラダーラング	Asian set 1 ラダーツールバー
CTRL	F7	反転コイル	ラダーラング	Asian set 1 ラダーツールバー
CTRL	F5	セットコイル	ラダーラング	Asian set 1 ラダーツールバー
CTRL	F6	リセットコイル	ラダーラング	Asian set 1 ラダーツールバー
	F8	アプリケーション命令	ラダーラング	Asian set 1 ラダーツールバー
	F9	水平ラインの描画	ラダーラング	Asian set 1 ラダーツールバー
	F10	垂直ラインの描画	ラダーラング	Asian set 1 ラダーツールバー
CTRL	F9	水平ラインの削除	ラダーラング	Asian set 1 ラダーツールバー
CTRL	F10	垂直ラインの削除	ラダーラング	Asian set 1 ラダーツールバー
SHIFT	F7	立上がりパルス接点	ラダーラング	Asian set 1 ラダーツールバー
SHIFT	F8	立下りパルス接点	ラダーラング	Asian set 1 ラダーツールバー
ALT	F7	立上がりパルス分岐開始	ラダーラング	Asian set 1 ラダーツールバー
ALT	F8	立下りパルス分岐開始	ラダーラング	Asian set 1 ラダーツールバー
CTRL+SHIFT	O	比較ブロック	ラダーラング	Asian set 1 ラダーツールバー
	X	XOR ブロック	ラダーラング	Asian set 1 ラダーツールバー
	F	ファンクションブロック	ラダーラング	Asian set 1 ラダーツールバー

修飾キー	キー	コマンド	ビュー	条件
	A	ステップを有効にする	ラダーラング	Asian set 1 ラダーツールバー
	D	ステップを無効にする	ラダーラング	Asian set 1 ラダーツールバー
CTRL+ALT	F10	操作結果を取り消す	ラダーラング	Asian set 1 ラダーツールバー
	O	他のラダー項目	ラダーラング	Asian set 1 ラダーツールバー
ALT	F10	自由描画ライン	ラダーラング	Asian set 1 ラダーツールバー
ALT	F9	自由描画ラインの削除	ラダーラング	Asian set 1 ラダーツールバー
	C	新規 A 接点	ラダーラング	Asian set 2 ラダーツールバー
	/	新規 B 接点	ラダーラング	Asian set 2 ラダーツールバー
	W	新規 A 接点 OR	ラダーラング	Asian set 2 ラダーツールバー
	X	新規 B 接点 OR	ラダーラング	Asian set 2 ラダーツールバー
CTRL+SHIFT	F4	立上がり接点	ラダーラング	Asian set 2 ラダーツールバー
CTRL+SHIFT	F5	立下り接点	ラダーラング	Asian set 2 ラダーツールバー
CTRL+SHIFT	O	比較ブロック	ラダーラング	Asian set 2 ラダーツールバー
ALT	X	XOR ブロック	ラダーラング	Asian set 2 ラダーツールバー
	F10	新規垂直ライン	ラダーラング	Asian set 2 ラダーツールバー
ALT	I	新規水平ライン	ラダーラング	Asian set 2 ラダーツールバー
	O	新規コイル	ラダーラング	Asian set 2 ラダーツールバー
	Q	新規閉コイル	ラダーラング	Asian set 2 ラダーツールバー
CTRL	F9	セットコイル	ラダーラング	Asian set 2 ラダーツールバー
CTRL+SHIFT	F9	リセットコイル	ラダーラング	Asian set 2 ラダーツールバー
	A	ステップを有効にする	ラダーラング	Asian set 2 ラダーツールバー
	D	ステップを無効にする	ラダーラング	Asian set 2 ラダーツールバー
	I	新規命令	ラダーラング	Asian set 2 ラダーツールバー
	F	新規ファンクションブロック	ラダーラング	Asian set 2 ラダーツールバー
ALT	O	他のラダー項目	ラダーラング	Asian set 2 ラダーツールバー
	F2	分岐モードを無効にする	ラダーラング	European、または American ラダーツールバー
SHIFT	F2	分岐モードを有効にする	ラダーラング	European、または American ラダーツールバー
SHIFT	F3	A 接点	ラダーラング	European ラダーツールバー
SHIFT	F4	B 接点	ラダーラング	European ラダーツールバー
CTRL+SHIFT	F4	立上がり接点	ラダーラング	European ラダーツールバー
CTRL+SHIFT	F5	立下り接点	ラダーラング	European ラダーツールバー
CTRL+SHIFT	6	演算ブロック	ラダーラング	European ラダーツールバー
CTRL+SHIFT	O	比較ブロック	ラダーラング	European ラダーツールバー
	X	XOR ブロック	ラダーラング	European ラダーツールバー
SHIFT	F7	割り当て	ラダーラング	European ラダーツールバー
CTRL+SHIFT	F9	反転コイル	ラダーラング	European ラダーツールバー
	F9	セットコイル	ラダーラング	European ラダーツールバー
SHIFT	F9	リセットコイル	ラダーラング	European ラダーツールバー

修飾キー	キー	コマンド	ビュー	条件
	A	ステップを有効にする	ラダーラング	European ラダーツールバー
	D	ステップを無効にする	ラダーラング	European ラダーツールバー
SHIFT	F5	ファンクションブロック	ラダーラング	European ラダーツールバー
CTRL+SHIFT	F6	演算ブロック	ラダーラング	European ラダーツールバー
	F3	直線	ラダーラング	European ラダーツールバー
	F3	配線ラインの描画	ラダーラング	European ラダーツールバー
	F4	配線ラインの削除	ラダーラング	European ラダーツールバー
	O	他のラダー項目	ラダーラング	European ラダーツールバー
SHIFT	F2	分岐モードを有効にする	ラダーラング	SoMachine ラダーツールバー
	F2	分岐モードを無効にする	ラダーラング	SoMachine ラダーツールバー
	F3	配線ラインの描画	ラダーラング	SoMachine ラダーツールバー
SHIFT	F3	配線ラインの削除	ラダーラング	SoMachine ラダーツールバー
	F4	A 接点	ラダーラング	SoMachine ラダーツールバー
SHIFT	F4	B 接点	ラダーラング	SoMachine ラダーツールバー
CTRL	F9	コイル	ラダーラング	SoMachine ラダーツールバー
CTRL+SHIFT	F9	反転コイル	ラダーラング	SoMachine ラダーツールバー
	F9	セット出力コイル	ラダーラング	SoMachine ラダーツールバー
SHIFT	F9	リセット出力コイル	ラダーラング	SoMachine ラダーツールバー
CTRL+SHIFT	F4	立上がり接点	ラダーラング	SoMachine ラダーツールバー
CTRL+SHIFT	+F5	立下り接点	ラダーラング	SoMachine ラダーツールバー
CTRL+SHIFT	{6, 7}	演算ブロック	ラダーラング	SoMachine ラダーツールバー
CTRL+SHIFT	{O, P}	比較ブロック	ラダーラング	SoMachine ラダーツールバー
X、または ALT+X		XOR ブロック	ラダーラング	SoMachine ラダーツールバー
O、または ALT+O		他のラダー項目	ラダーラング	SoMachine ラダーツールバー
A、または ALT+A		ステップを有効にする	ラダーラング	SoMachine ラダーツールバー
D、または ALT+D		ステップを無効にする	ラダーラング	SoMachine ラダーツールバー



%S

IEC 規格により、%S はシステムビットを表します。

%SW

IEC 規格により、%S はシステムワードを表します。

アニメーションテーブル

入力ビットおよびメモリーワードなどのオブジェクトの今現在の値をソフトウェア上の表に示します。SoMachine Basic がロジックコントローラーに接続されている場合、アニメーションテーブル上の特定のオブジェクトタイプの値を強制的に任意の値にすることができます。アニメーションテーブルは SoMachine Basic アプリケーションの一部として保存されます。

アプリケーション

設定データ、シンボル、ドキュメントを含むプログラム。

インストラクションリスト言語

コントローラーにより順に実行される一連のテキストベースの命令で書かれたプログラム。各命令は、ライン番号、命令コードおよびオペランドを含んでいる (IEC 61131-3 を参照してください)。

ウォッチドッグ

ウォッチドッグは特殊なタイマーで、プログラムが決められたスキャンタイムを超過して実行されないようにします。ウォッチドッグタイマーは通常スキャンタイムより長い時間に設定され、スキャン周期の最後に 0 にリセットされます。ウォッチドッグタイマーが既定値に達した場合、例えばプログラムが無限ループに入った場合は、エラーとなりプログラムが停止します。

コントローラー

産業プロセスを自動化する (プログラマブルロジックコントローラーまたはプログラマブルコントローラーとして知られる)。

シンボリックアドレス指定

物理的入力および出力を含むメモリーオブジェクトをアドレス指定する間接的手法。プログラミング命令にオペランドおよびパラメーターとして、これらのプログラミング命令と関連したシンボルを最初に定義する。

即時アドレス指定とは対照的に、プログラム設定が変更された場合にシンボルが新しい関連アドレスに自動的に更新されるので、この手法は推奨されています。それとは逆に、オペランドまたはパラメーターとして使用される即時アドレスは更新されません (詳細は *即時アドレス指定* 参照)。

シンボル

アルファベットで始まる最大 32 文字 (アルファベットおよび数字) の文字列。これによりコントローラーのオブジェクトをパーソナライズして、アプリケーションの保守性を向上できます。

フラッシュメモリー

上書き可能な不揮発性メモリーです。特殊な EEPROM 上に記憶し、消去および書き換えが可能です。

プログラム

アプリケーションのコンポーネント。コンパイルされたソースコードで構成され、ソースコードはロジックコントローラーのメモリーにインストール可能。

ポスト設定

(*post configuration*、*ポストコンフィギュレーション*) アプリケーションを変更することなく、アプリケーションのパラメーターを変更できるオプション。ポスト設定パラメーターは、コントローラーに保存されているファイルで定義されます。それらは、アプリケーションの設定パラメーターを多重定義しています。

マスタータスク

プログラミングソフトウェアを介して実行されるプロセッサタスク。マスタータスクの2つのセクションを次に示します。

- **IN:** マスタータスクの実行前に、入力はこの IN セクションにコピーされます。
- **OUT:** マスタータスクの実行後に、出力はこの OUT セクションにコピーされます。

ユーザー定義ファンクション

1つまたは複数の入力パラメーター、ローカル変数、および戻り値を使用して独自のファンクションを作成できます。操作ブロックでユーザー定義ファンクションを呼び出すことができます。ユーザー定義ファンクションはプロジェクトの一部として保存され、アプリケーションの一部としてロジックコントローラーにダウンロードされます。

ユーザー定義ファンクションブロック

1つまたは複数の入力パラメーター、ローカル変数、および戻り値を使用して独自のファンクションブロックを作成できます。ユーザー定義のファンクションブロックは、プロジェクトの一部として保存されます。

ラダーダイアグラム言語

コントローラープログラムの命令を表す図。コントローラーで順次実行される一連のラングにある接点、コイル、およびブロックのシンボルを含む (IEC 61131-3 を参照してください)。

拡張バス

拡張 I/O モジュールとコントローラー間の電子通信用バス。

設定

システム内のハードウェアコンポーネントの配置と接続、およびシステムの動作特性を決めるハードウェアおよびソフトウェアパラメーターの設定。

非プログラムデータ

プロジェクトプロパティ、シンボルおよびコメントなどのプログラムに直接使用されていない SoMachine Basic アプリケーション内のデータ。

EtherNet/IP

(*Ethernet industrial protocol*、*産業用イーサネットプロトコル*)。産業システムにおける生産自動化ソリューション向けの、オープン通信プロトコル。EtherNet/IP は、上位層に産業共通プロトコルを実装したネットワークのファミリーに属します。サポート組織の ODVA は、グローバルな適応性とメディアの独立性を得るため、EtherNet/IP を規定しています。

Free POU

programmable object unit、プログラマブルオブジェクトユニット (POU)。プログラムのマスタータスクとは別にプログラムおよび更新が可能。一般的にライブラリー関数を含みます。フリー POU をプログラムのサブルーチンまたはジャンプとして呼び出し可能です。例えば定期的なタスクはサブルーチンで、フリー POU としてインプリメントされています。

GRAFNET

一連のオペレーションがどのように機能しているかを、構造化されたグラフィック形式で表したもの。

シーケンス制御システムを、アクション、トランジションおよび状態と関連する一連のステップに分割する分析的手法。

I/O

(入力/出力)

POU

(*program organization unit*, プログラムオーガニゼーションユニット) ソースコードおよび対応する命令セット内の変数宣言。POUs はソフトウェアプログラム、ファンクション、およびファンクションブロックのモジュラー化した再利用を容易にします。一度宣言すると POUs がもう一方でも利用可能となります。

RTC

(*real-time clock*, リアルタイムクロック) 電池の寿命の間、コントローラーに給電されていない時でも継続して動作する電池バックアップ式日時およびカレンダークロック。

TCP

(*transmission control protocol*, 伝送制御プロトコル) データの同時双方向伝送を提供する接続ベースのトランスポート層プロトコル。TCP は TCP/IP プロトコルスイートの一部です。



- (%HSC の) 閾値出力
 - イベントソースとして設定, 120
- end/jump
 - グラフィック要素, 161
- Ethernet
 - ポスト設定ファイルを使用した設定, 227
- EXCEPTION ステート
 - フォールバック動作, 76
- Free POU
 - イベントソースに割り当て, 123
 - イベントに割り当て, 91
- Free POU
 - 作成, 89
- Free POU
 - 削除, 91
 - 周期タスクに割り当て, 91
 - 周期タスク用, 115
 - 概要, 81
- IL (インストラクションリスト)
 - コメント, 173
- LI (リストインストラクション), 175
- POU
 - Free, 115
 - コピー, 90
 - タスクと管理する, 82
 - 概要, 81
 - 貼り付け, 90
- PTO オブジェクト, 140
- RAM メモリー
 - アプリケーションを含む実行可能ファイル, 227
- RAM メモリー、消費量, 150
- RTC
 - コントローラーで更新, 229
 - システムビットの管理, 168
 - 日付と時間の表示, 227
- SD カード
 - メモリー管理と, 222
- SoMachine Basic の登録, 30
- STOPPED ステート
 - フォールバック動作, 76
- TH0、TH1
 - イベントソースとして設定, 120
- Twido プロジェクト、SoMachine Basic に変換, 251
- Twido プロジェクトを SoMachine Basic に変換, 251
- XOR
 - グラフィック要素, 160
- アキュムレーター, 174
- アクションゾーン, 155
- アップロード
 - パスワードで防止, 57
 - ロジックコントローラーからアプリケーション, 219
- アドレス指定
 - シンボル, 67
- アニメーションテーブル, 128
- アプリケーション
 - コントローラーヘダダウンロード, 217
 - パスワードで保護, 57
 - パスワード保護, 55
 - パスワード保護の有無, 227
 - ロジックコントローラーからのアップロード, 219
 - 動作、設定, 75
 - 定義, 23
- アプリケーションのパスワード保護, 55
- イベント
 - サブルーチンをトリガー, 120
 - 最後のコールドスタート以降, 124
- イベントソース
 - サブルーチンの割り当て, 123
 - タイプの, 120
- イベントタスク
 - 概要, 119
 - 管理, 122
 - 設定, 78
- イベントのソース, 120
- イベントの優先度, 119
- インポート
 - シンボルリスト, 147
- ウォッチドッグタイマー、設定, 76
- エクスポート
 - シンボルリスト, 148
 - トレース, 201
- オブジェクト
 - アニメーションテーブルでトレースする, 128
 - ネットワーク, 138
 - リアルタイムで値を更新する, 128
 - レポート表示, 51
 - 定義, 66
- オフラインモード
 - ステータスエリアに表示, 47
 - 概要, 26
- オペランド, 174
- オペレーションブロック
 - グラフィック要素, 161
- オペレーティング、モード, 26
- オペレーティングモード, 26

- オンラインモード, 69
 - RTC, 229
 - アニメーションテーブル, 128
 - アニメーションテーブルで値を編集する, 130
 - ステータスエリアに表示, 47
 - デバッグ, 199
 - 概要, 26
- オンラインモードでのデバッグ, 199
- カスタマイズ、ラダーエディター, 49
- カタログ, 61
 - ロジックコントローラーをリファレンスで置き換える, 62
- キーボードショートカット, 261
- キーボードのショートカット, 49
- キャッシュメモリー、消費量, 150
- グラフィック要素
 - ラダープログラム, 158
- グラフセ, 181
 - グラフィカルエレメント, 161
 - シーケンス処理, 183
 - プログラムの構成, 182
 - 前処理, 182
 - 命令, 181
 - 命令の使い方, 185
 - 後処理, 184
- グラフセ (SFC)
 - グラフセグラフィカルエディター, 191
- グラフセ POU
 - のコピーと貼り付け, 87
 - の削除, 88
 - 作成, 87
 - 名の変更, 88
- グリッド線、ラダーエディター内のスタイル, 49
- コイル
 - グラフィック要素, 160
 - 出力の図による表示, 153
- コピーと貼り付け
 - POU, 90
 - グラフセ POU, 87
- コメント
 - IL への追加, 173
 - ラダー図の追加, 167
- コントローラーのメモリー割り当て, 69
- コントローラー時間、トレースに表示, 200
- コンパイル、最後の日付と時間, 150
- サブルーチン
 - POU として実装, 81
 - イベントで実行をトリガー, 120
 - タスクへの割り当て, 122
 - マスタータスクの, 106
 - 周期タスクの割り当て, 115
- システムビット / ワード
 - シンボルリスト内, 146
- システムオブジェクト, 136
- システムビット
 - %S0, 168
 - %S11, 107
 - %S14, 227
 - %S19, 107
 - %S31, 124
 - %S38, 124
 - %S39, 124
 - %S49, 168
 - %S51, 168
- システムビット / ワード
 - イベントの管理, 124
- システムワード
 - %SW0, 107
 - %SW27, 107
 - %SW30, 107
 - %SW30...%SW32, 227
 - %SW31, 107
 - %SW32, 107
 - %SW35...%SW38, 227
 - %SW48, 124
 - %SW54...%SW57, 227
 - %SW58, 227
 - %SW6, 227
- システム要件, 19
- シミュレーター, 232
 - I/O マネージャーウィンドウ, 234
 - アナログ入力値の変更, 239
 - シミュレーターウィンドウ, 232
 - シミュレーターへアクセスする, 232
 - デジタル入力値の変更, 238
 - モード、概要, 26
 - 使い方, 242
 - 値の変更, 238
 - 出力のトレース, 240
 - 時間管理, 236
- シリアルライン
 - ポスト設定ファイルを使用した設定, 227
- シンボル
 - アドレス指定, 67
 - プロパティウィンドウで定義, 67
 - ラダーエディターのグラフィック要素を定義, 68
 - リスト、使用された, 146
 - ロジックコントローラーに保存, 68
- シンボルアドレス指定, 67
- シンボルリスト
 - Vijeo Designer プロジェクトと共有する, 148
 - インポート, 147
 - エクスポート, 148
 - 表示する, 146
- シンボルを共有する
 - Vijeo Designer と, 148

- スキャンタイム
 - 最小、ステータスエリアに表示, 47
 - 最小、最大、現在の表示, 227
- スキャンタスク、ウォッチドッグ設定, 76
- スキャンモード, 78, 107
- ステータスエリア, 47
- ステート
 - コントローラーの、表示, 227
 - ロジックコントローラーの初期設定, 75
- セクション
 - イベントの, 119
 - マスタータスクの, 106
- ソフトウェアオブジェクト, 139
- タイマー、ウォッチドッグ, 76
- ダウンロード
 - アプリケーションをコントローラーへ, 217
 - ファームウェアの更新, 221
- ダウンロードする
 - アプリケーションを直接コントローラーに, 37
- タスク
 - イベント, 119
 - 周期, 115
 - 設定, 78
- ツール
 - PTO オブジェクト, 140
 - アニメーションテーブル, 128
 - システムオブジェクト, 136
 - シンボルリスト, 146
 - ソフトウェアオブジェクト, 139
 - ドライブオブジェクト, 141
 - ネットワークオブジェクト, 138
 - メモリーオブジェクト, 133
 - メモリー消費量, 150
 - 使う, 125
 - 入力 / 出力 オブジェクト, 137
 - 検索と置換, 143
 - 通信オブジェクト, 142
- ツールバーのボタン, 45
- デジタル入力
 - イベントソースとして設定, 120
- テストゾーン, 155
- テンプレート
 - プロジェクト, 39
 - 名前を付けてプロジェクトファイルを保存, 247
- ドライブオブジェクト, 141
- トレース
 - PDF にエクスポート, 201
 - オブジェクトを選択する, 128
 - 時間ベース を選択する, 132
 - 表示, 200
- ネットワークオブジェクト, 138, 138
- ハードウェアコンポーネント、設定, 61
- ハードウェアツリー, 61
- パスワード
 - アプリケーションから解除, 58
 - アプリケーションの保護, 57
 - アプリケーション保護の有無, 227
 - プロジェクトから解除, 57
 - プロジェクトを開くために必要, 57
- パスワード保護の解除, 57, 58
- パルス幅 (TON), 107
- ブール演算子
 - アキュムレーター, 174
 - グラフィック要素, 160
- ファームウェア、更新データのコントローラーへのダウンロード, 221
- ファームウェアの更新, 221
- ファンクションブロック
 - グラフィック要素, 160
- ファンクションレベル, 77
- フォールバック
 - 値, 76
 - 動作、指定, 76
- フォールバック値の保持, 76
- プログラミング
 - グリッド, 155
 - ワークスペース, 64
 - 推奨方法, 168
 - 言語、サポートされている, 21
- プログラム
 - definition of, 23
 - コンパイル, 45
 - ジャンプ, 168
 - 行数の表示, 150
- プログラム、フォールバック動作の設定, 76
- プログラム以外のデータ, 128
- プログラム変更の送信, 208
- プログラム構成単位 (POU), 81
- プログラム開発、ステージ, 24, 24
- プログラム開発の段階, 25
- プロジェクト
 - テンプレート, 39
 - テンプレートとして保存, 247
 - パスワード付きプロジェクト, 57
 - プロパティの設定, 55
 - 作成, 23
 - 保存, 246
 - 定義, 23
- プロジェクトを作成する, 23
- プロパティ, 55
- ヘルプ
 - ショートカットキーの変更, 49
- ポスト設定
 - からの Ethernet パラメーターの使用, 227
 - からのシリアルラインパラメーターの使用, 227
- ポスト設定ファイルを投稿、Ethernet パラ

- メーターを書き込む, 36
- ボタン、ツールバー, 45
- マスタータスク
 - POU の割り当て, 81
 - システムビットとワード制御, 107
 - 設定, 78, 106
- メモリーオブジェクト, 133
- メモリーの割り当て, 69
- メモリー内容のバックアップ, 208
- メモリー消費量、表示する, 150
- メモリー管理と SD カード, 222
- モジュールエリア, 25
- モデム
 - 接続、追加する, 34
- モデムス
 - テータスの表示, 227
- ユーザーインターフェース
 - 言語の設定, 49
- ユーザー定義ファンクション
 - プログラム, 94
 - 管理, 97
- ユーザー定義ファンクションブロック
 - プログラミング, 101
 - 定義, 101
 - 管理, 103
- ライフサイクルステート
 - ロジックコントローラーの, 47
- ライン
 - グラフィック要素, 159
- ラダー / リスト可逆性, 70
- ラダーエディター
 - カスタマイズする, 49
 - シンボルを定義, 68
 - 挿入後にポインターをリセットする, 49
- ラダープログラム
 - グラフィック要素, 158
- ラダー図
 - IL (インストラクションリスト) に戻す, 70
 - コメント, 167
 - はじめに, 153
 - プログラミングの原則, 155
 - ラング, 153
 - 括弧の使用, 168
- ラング
 - グラフィック要素, 158
 - コピー, 85
 - 作成, 84
 - 削除, 86
 - 名前の変更, 85
 - 挿入, 84
 - 管理, 84
- リスト言語
 - 概要, 172
- リレー回路、ラダー図での表示, 153
- レポート
 - エクスポート, 51
 - 印刷, 51
- レポートの印刷, 51
- ロールバック変更, 208
- ロジックコントローラー
 - RTC の更新, 229
 - アプリケーションの直接ダウンロード, 37
 - ステートの表示, 227
 - についての情報の表示, 227
 - ファームウェアの更新, 221
 - 対応タイプ, 20
 - 最終停止時の日付と時間, 227
 - 現在の設定を置き換え, 62
 - 起動時ステート、設定, 75
- ロジックコントローラーとの接続, 212
- ロジックコントローラーの起動ステート, 75
- 一般設定, 49
- 作成
 - Free POU, 89
 - グラフセ POU, 87
- 値の強制
 - I/O の, 227
- 値を強制する
 - アニメーションテーブルで, 128
- 停止センサー、配線, 168
- 停止センサーの配線, 168
- 入力
 - イベントソースとして設定, 120
 - 変更, 168
- 入力 / 出力 オブジェクト, 137
- 共有
 - シンボルリスト, 148
- 出力
 - 変更, 168
- 分岐モード
 - グラフィック要素, 158
- 削除
 - Free POU, 91
 - グラフセ POU の削除, 88
- 割り当てモード, 69
- 割り当て命令
 - ラダーダイアグラムのラングに挿入, 164
- 可逆性
 - の概要, 70
- 名の変
 - グラフセ POU 名の変更, 88
- 周期
 - スキャン周期, 117
 - タスク, 115
- 周期タスク
 - Free POU の割り当て, 91
 - 期間の設定, 117

周期的
スキャンモード, 107

周期的タスク
設定, 78

命令
アップストリーム / ダウンストリーム,
168

対応デバイス, 20

拡張モジュール
対応デバイス, 20

括弧
プログラム内での使用, 178
ラダー図の使用, 168
修飾子, 178
入れ子, 179

挿入
新規グラフセ POU, 87

接点
グラフィック要素, 159
入力の図による表示, 153

操作
ラダーダイアグラムのラングに挿入,
164

新規グラフセ POU
挿入, 87

時間ベース (トレース用), 132

最小システム要件, 19

期間、スキャン, 107

検索と置換, 143

比較ブロック
IL 式を挿入, 163
グラフィック要素, 159

比較式
ラダーダイアグラムのラングに挿入,
163

演算ブロック
割り当て命令の挿入, 164

置き換え
ロジックコントローラーの設定, 62

言語
ユーザーインターフェース, 49

設定
アプリケーションの動作, 75
タスクとスキャン, 78
プロジェクトプロパティ, 55
マスタータスク, 106
ロジックコントローラーの置き換え, 62
一般, 49
周期タスクの期間, 117
現在の, 61

設定する
設定ウィンドウのハードウェアコン
ポーネント, 61

通信, 25
ロジックコントローラーとの接続, 212
通信ウィンドウ, 210

通信オブジェクト, 142
通常スキャンモード, 107
選択
グラフィック要素, 159
部品表 (BOM)、印刷, 51
開始ページ, 25
開発段階, 25
非プログラムデータ, 23
ダウンロード, 208
非プログラムデータのダウンロード, 208

