



EasyBuilder Pro

Ver. 6.04.02

1.	EasyBuilder Pro のインストールについて	1-1
1.1.	動作環境	1-2
1.2.	インストール手順	1-3
2.	Utility Manager	2-1
2.1.	概要	2-2
2.2.	HMI IP 及びパスワードの設定	2-5
2.3.	編集ツール	2-6
2.3.1.	SD カード及び USB ディスクに使用するダウンロードデータを作成する	2-6
2.3.2.	USB ディスク/SD カードでプロジェクトを HMI にダウンロードする手順	2-6
2.4.	転送	2-7
2.4.1.	ダウンロード	2-7
2.4.2.	アップロード	2-8
2.5.	シミュレーション	2-10
2.5.1.	オフラインシミュレーションとオンラインシミュレーション	2-10
2.6.	パススルー	2-11
3.	プロジェクトを作成する	3-1
3.1.	概要	3-2
3.2.	新規プロジェクトを作成する	3-2
3.3.	プロジェクトを保存/コンパイルする	3-4
3.4.	オフラインシミュレーションとオンラインシミュレーション	3-5
3.5.	cMT Viewer	3-6
3.6.	プロジェクトを HMI にダウンロードする	3-7
3.6.1.	EasyBuilder Pro から設定する	3-7
3.6.2.	HMI 名を使用する	3-10
3.6.3.	USB ケーブルでダウンロード	3-11
3.6.4.	USB ディスク/SD カードを使用する	3-12
3.7.	HMI からプロジェクトをアップロードする	3-13
4.	ハードウェア設定	4-1
4.1.	概要	4-2
4.2.	I/O ポート	4-2
4.3.	LED インジケーター	4-2
4.4.	システム設定	4-3
4.5.	システムツールバー	4-4
4.5.1.	システム設定	4-4
4.5.2.	システム情報	4-12
5.	システムパラメータ設定	5-1
5.1.	概要	5-2

5.2. 装置	5-2
5.2.1. ローカル PLC をコントロールする仕方.....	5-2
5.2.2. 遠隔地にある装置(リモート装置)をコントロールする仕方	5-7
5.2.3. 遠隔地にある HMI(リモート HMI)をコントロールする仕方	5-9
5.3. HMI 属性	5-11
5.4. 一般的な属性	5-14
5.5. システム	5-20
5.6. リモート	5-23
5.7. ユーザーパスワード	5-25
5.7.1. 一般モード	5-26
5.7.2. 高度セキュリティモード	5-26
5.8. 拡張メモリー	5-29
5.9. プリンター/バックアップサーバー.....	5-30
5.10. 時刻同期/夏時間.....	5-32
5.11. E メール	5-34
5.12. セルラーネットワーク	5-36
6. ウィンドウ	6-1
6.1. 概要	6-2
6.2. ウィンドウの種類	6-2
6.2.1. ベースウィンドウ	6-2
6.2.2. ファーストセレクションウィンドウ	6-2
6.2.3. 共有ウィンドウ	6-3
6.2.4. システムメッセージウィンドウ	6-4
6.3. ウィンドウの作成、設定と削除	6-5
6.3.1. ウィンドウの作成と設定	6-6
6.3.2. ウィンドウをオープンする、閉じる、削除する	6-8
6.4. ウィンドウの透明度	6-8
6.5. 編集ティップス	6-11
6.5.1. 非表示エリア、選択ツール、手のひらツール	6-11
6.5.2. 定規	6-12
6.5.3. クリックコピー機能	6-13
7. イベントログ	7-1
7.1. 概要	7-2
7.2. イベントログの管理	7-2
7.2.1. eMT、iE、XE、mTV、iP シリーズ.....	7-3
7.2.2. cMT シリーズ	7-4
7.2.3. Excel 編集	7-7
7.2.4. 迅速に無効なイベントを閲覧する	7-8

7.3. イベントログを新規作成する	7-8
8. データサンプリング	8-1
8.1. 概要	8-2
8.2. データサンプリングの管理	8-2
8.3. 新規データサンプリングオブジェクトを作成する	8-2
8.3.1. 自動中止の例	8-9
8.3.2. カスタマイズファイル管理	8-11
8.4. cMT Viewer のデータを外部装置またはデータベースに同期して保存する	8-13
8.5. cMT Viewer の特定の日付またはファイルの履歴データを調べる	8-13
9. オブジェクトの一般的な属性	9-1
9.1. 概要	9-2
9.2. 装置を選択し、読み書きアドレスを設定する	9-2
9.3. 図形ライブラリと画像ライブラリを使用する	9-3
9.3.1. 画像	9-4
9.3.2. 図形	9-4
9.3.3. 図形管理	9-6
9.4. テキストラベルの内容を設定する	9-7
9.5. 輪郭を調整する	9-10
10. ユーザーパスワードとオブジェクトセキュリティ	10-1
10.1. 概要	10-2
10.2. ユーザーパスワードと操作可能なオブジェクトのクラスの設定	10-2
10.2.1. 一般モード	10-2
10.2.2. 高度セキュリティモード	10-3
10.2.3. LDAP モード	10-4
10.3. 高度セキュリティモードのコントロールアドレス	10-4
10.3.1. コントロールアドレスの使用説明	10-5
10.3.2. コマンド機能の説明	10-6
10.3.3. コマンド実行結果の送信	10-9
10.4. 高度セキュリティモードの応用	10-9
10.4.1. ユーザーアカウントをインポートする	10-9
10.4.2. USB セキュリティキーでログインする	10-11
10.4.3. USB セキュリティキーで自動的にログイン/ログアウトする	10-13
10.4.4. 高度セキュリティモードにオプションリストオブジェクトを合わせる	10-14
10.4.5. LDAP モード	10-15
10.5. オブジェクトセキュリティ設定	10-21
10.5.1. 安全コントロール	10-21
10.5.2. 有効/無効にする	10-22
10.5.3. ユーザー制限	10-26

10.6.	オブジェクトセキュリティの例	10-27
10.7.	ユーザー・パスワード保護	10-30
10.8.	複数オブジェクトのセキュリティ属性設定を一括変更	10-31
11.	インデックスレジスタ	11-1
11.1.	概要	11-2
11.2.	インデックスレジスタの使用例	11-4
12.	キーボードのデザインと使用	12-1
12.1.	概要	12-2
12.2.	ポップアップキーボードをデザインする	12-2
12.3.	直接ウインドウでキーボードをデザインする	12-5
12.4.	固定キーボードを作成する	12-6
12.5.	Unicode キーボードを作成する	12-6
13.	オブジェクト	13-1
13.1.	ビットランプ	13-3
13.1.1.	概要	13-3
13.1.2.	設定	13-3
13.2.	ワードランプ(マルチ状態ランプ)	13-5
13.2.1.	概要	13-5
13.2.2.	設定	13-5
13.3.	ビット状態設定	13-10
13.3.1.	概要	13-10
13.3.2.	設定	13-10
13.4.	ワード設定(マルチ状態設定)	13-13
13.5.	ファンクションキー	13-21
13.5.1.	概要	13-21
13.5.2.	設定	13-21
13.6.	トグルスイッチ	13-27
13.6.1.	概要	13-27
13.6.2.	設定	13-27
13.7.	マルチ状態切替スイッチ	13-30
13.7.1.	概要	13-30
13.7.2.	設定	13-30
13.8.	スライドスイッチ	13-34
13.8.1.	概要	13-34
13.8.2.	設定	13-34
13.8.3.	複合型設定	13-37
13.9.	数値	13-40
13.9.1.	概要	13-40

13.9.2. 設定	13-40
13.10. アスキイ	13-59
13.10.1. 概要	13-59
13.10.2. 設定	13-59
13.11. 間接ウインドウ	13-62
13.11.1. 概要	13-62
13.11.2. 設定	13-62
13.12. 直接ウインドウ	13-71
13.12.1. 概要	13-71
13.12.2. 設定	13-71
13.13. 移動/回転図形	13-78
13.13.1. 概要	13-78
13.13.2. 設定	13-78
13.14. アニメーション	13-83
13.14.1. 概要	13-83
13.14.2. 設定	13-83
13.15. 棒グラフ	13-87
13.15.1. 概要	13-87
13.15.2. 設定	13-87
13.15.3. 複合型設定	13-93
13.16. メーター	13-96
13.16.1. 概要	13-96
13.16.2. 設定	13-96
13.17. トレンド図	13-104
13.17.1. 概要	13-104
13.17.2. 設定	13-104
13.18. 履歴データ表示	13-124
13.18.1. 概要	13-124
13.18.2. 設定	13-124
13.19. データグループ表示	13-132
13.19.1. 概要	13-132
13.19.2. 設定	13-132
13.20. XY プロット	13-140
13.20.1. 概要	13-140
13.20.2. 設定	13-140
13.21. アラームバーとアラーム表示	13-149
13.21.1. 概要	13-149
13.21.2. 設定	13-149

13.22. イベント表示	13-156
13.22.1. 概要	13-156
13.22.2. 設定	13-156
13.23. データ転送	13-168
13.23.1. データ転送(ウインドウ)	13-168
13.23.2. データ転送(背景)時間ベース型	13-171
13.23.3. データ転送(背景)ビットトリガー型	13-174
13.24. バックアップ	13-176
13.24.1. 概要	13-176
13.24.2. 設定	13-176
13.25. メディアプレーヤー	13-184
13.25.1. 概要	13-184
13.25.2. 設定	13-184
13.26. BACnet Schedule	13-191
13.26.1. 概要	13-191
13.26.2. 設定	13-191
13.27. PLC コントロール	13-195
13.27.1. 概要	13-195
13.27.2. 設定	13-195
13.28. スケジューラー	13-203
13.28.1. 概要	13-203
13.28.2. 設定	13-203
13.29. オプションリスト	13-214
13.29.1. 概要	13-214
13.29.2. 設定	13-214
13.30. タイマー	13-223
13.30.1. 概要	13-223
13.30.2. 設定	13-223
13.31. 映像入力	13-228
13.31.1. 概要	13-228
13.31.2. 設定	13-228
13.32. システムメッセージ	13-235
13.32.1. 概要	13-235
13.32.2. 設定	13-235
13.33. レシピ閲覧	13-237
13.33.1. 概要	13-237
13.33.2. 設定	13-237
13.34. フローブロック	13-243

13.34.1. 概要	13-243
13.34.2. 設定	13-243
13.35. 操作ログ	13-248
13.35.1. 操作ログ設定	13-248
13.35.2. 操作ログ閲覧	13-251
13.35.3. 操作ログ印刷	13-254
13.36. 複合ボタン	13-261
13.36.1. 概要	13-261
13.36.2. 設定	13-261
13.37. 近似曲線円グラフ	13-269
13.37.1. 概要	13-269
13.37.2. 設定	13-269
13.38. 画像閲覧	13-278
13.38.1. 概要	13-278
13.38.2. 設定	13-278
13.39. ファイルブラウザ	13-280
13.39.1. 概要	13-280
13.39.2. 設定	13-280
13.40. インポート/エクスポート	13-283
13.40.1. 概要	13-283
13.40.2. 設定	13-283
13.41. 円グラフ	13-288
13.41.2. 設定	13-288
13.41.3. 複合型設定	13-289
13.42. バーコード表示	13-292
13.42.1. 二次元バーコード表示	13-292
13.42.2. WeChat バーコード表示	13-293
13.42.3. バーコードスキャナー(Android カメラ)	13-302
13.43. ストリングテーブル	13-308
13.43.1. 概要	13-308
13.43.2. 設定	13-308
13.44. データベース	13-310
13.44.1. データベースサーバー	13-310
13.44.2. SQL クエリ	13-315
13.44.3. SQL クエリ結果ビューア	13-323
13.45. 動的スケール	13-325
13.45.1. 概要	13-325
13.45.2. 設定	13-325

13.46. 動的作画	13-328
13.46.1. 概要	13-328
13.46.2. 設定	13-328
13.47. PDF リーダー	13-334
13.47.1. 概要	13-334
13.47.2. 設定	13-334
13.48. テーブル	13-336
13.48.1. 概要	13-336
13.48.2. 設定	13-336
13.49. VNC Viewer	13-338
13.49.1. 概要	13-338
13.49.2. 設定	13-338
13.50. 連絡先エディター	13-342
13.50.1. 設定	13-342
13.51. イベントバーチャート	13-345
13.51.1. イベントログ	13-345
13.51.2. イベントバーチャート	13-347
13.52. 動作トリガー	13-350
13.52.1. 概要	13-350
13.52.2. 設定	13-350
13.53. カレンダー	13-356
13.53.1. 概要	13-356
13.53.2. 設定	13-356
13.54. タッチジェスチャー	13-358
13.54.1. 概要	13-358
13.54.2. 設定	13-358
14. 図形ライブラリ及び画像ライブラリの作成	14-1
14.1. 概要	14-2
14.2. 図形ライブラリの作成	14-2
14.2.1. 図形管理	14-3
14.2.2. 図形ライブラリを作成する手順	14-5
14.3. 画像ライブラリの作成	14-10
14.3.1. 画像管理	14-10
14.3.2. 画像ライブラリを新規作成する手順	14-12
14.3.3. クリップボードから画像をインポートする	14-14
14.4. 図形ライブラリ/画像ライブラリの即時調整及び管理	14-17
14.4.1. 画像の置換	14-17
14.4.2. 色の置換	14-18

14.4.3. 付加機能	14-19
14.4.4. ウィンドウ位置の調整	14-20
15. ラベルタグライブラリ及び多言語の使用	15-1
15.1. 概要	15-2
15.2. ラベルタグライブラリの管理	15-2
15.3. ラベルタグライブラリの作成	15-3
15.4. ラベルタグライブラリの使用	15-4
15.5. 言語とフォント	15-5
15.5.1. 言語	15-5
15.5.2. 非 ASCII フォント	15-5
15.5.3. フォント管理	15-7
15.6. 多言語の使用	15-9
16. アドレスタグライブラリの作成及び使用	16-1
16.1. 概要	16-2
16.2. アドレスタグライブラリの作成	16-2
16.3. アドレスタグライブラリの使用	16-5
17. レシピデータの転送	17-1
17.1. 概要	17-2
17.2. イーサネットまたは USB ケーブルでレシピデータを更新する	17-2
17.3. SD カードまたは USB ディスクでレシピデータを更新する	17-3
17.4. レシピデータの転送	17-5
17.5. レシピデータの保存メカニズム	17-5
18. マクロの説明	18-1
18.1. 概要	18-2
18.2. マクロエディターの使用説明	18-2
18.3. マクロの構成	18-8
18.4. マクロのシンタックス	18-9
18.4.1. 定数と変数	18-9
18.4.2. 演算子	18-11
18.5. ステートメント	18-15
18.5.1. 定義ステートメント	18-15
18.5.2. 代入ステートメント	18-15
18.5.3. 論理演算ステートメント	18-15
18.5.4. 選択ステートメント	18-17
18.5.5. 反復ステートメント	18-19
18.6. サブ関数	18-21
18.7. 組み込み関数機能	18-23
18.7.1. 関数一覧表	18-23

18.7.2.	PLC	18-27
18.7.3.	Free protocol.....	18-36
18.7.4.	マクロ制御	18-40
18.7.5.	データ操作	18-41
18.7.6.	データ変換	18-46
18.7.7.	ストリング処理	18-56
18.7.8.	数学	18-76
18.7.9.	統計	18-81
18.7.10.	レシピデータベース	18-83
18.7.11.	データサンプリング/イベントログ	18-86
18.7.12.	パリティ	18-90
18.7.13.	他の関数	18-92
18.8.	マクロの作成、及び実行の方法について	18-94
18.8.1.	マクロの作成方法について	18-94
18.8.2.	マクロを実行する	18-99
18.9.	ユーザー定義関数機能	18-100
18.9.1.	関数ライブラリファイルをインポートする	18-101
18.9.2.	マクロ関数ライブラリの利用方法	18-102
18.9.3.	関数ライブラリ管理インターフェース	18-104
18.10.	マクロを使用する際の注意点	18-115
18.11.	フリー通信プロトコルで装置をコントロールする	18-116
18.12.	コンパイルエラーメッセージ	18-123
18.13.	マクロコマンドのサンプルプロジェクト	18-130
18.14.	マクロ TRACE 関数	18-135
18.15.	ストリング操作関数の使用方法	18-139
18.16.	マクロパスワード保護	18-147
18.17.	CANbus アドレスの読み取り/書き込みに変数を使用	18-149
19.	HMI を MODBUS 装置に設定する方法	19-1
19.1.	概要	19-2
19.2.	MODBUS Server 装置を作成する	19-2
19.3.	一個の MODBUS Server 装置をアクセスする	19-5
19.4.	オンラインで MODBUS Server ステーション番号を変更する	19-8
19.5.	各 MODBUS アドレスの説明	19-8
20.	バーコードスキャナーを使用する方法	20-1
20.1.	概要	20-2
20.2.	バーコードスキャナーを接続する手順	20-2
21.	イーサネット通信及びマルチ接続	21-1
21.1.	概要	21-2

21.2.	HMI と HMI の間の通信	21-2
21.3.	PC と HMI の間の通信	21-3
21.4.	他の HMI に接続している PLC をコントロール.....	21-4
21.4.1.	eMT/ iE / XE / mTV / iP シリーズの設定方法	21-5
21.4.2.	cMT シリーズの設定方法	21-5
22.	レジスタ	22-1
22.1.	概要	22-3
22.2.	ローカル HMI のアドレス範囲	22-4
22.2.1.	ビットアドレス	22-4
22.2.2.	ワードアドレス	22-5
22.3.	システムレジスタ	22-5
22.3.1.	HMI 時間	22-5
22.3.2.	HMI 操作	22-7
22.3.3.	タッチ位置	22-9
22.3.4.	ローカル HMI ネットワーク情報	22-9
22.3.5.	プロジェクト情報	22-12
22.3.6.	保存領域の管理	22-13
22.3.7.	レシピ及び拡張メモリー	22-14
22.3.8.	データサンプリング	22-15
22.3.9.	イベントログ	22-16
22.3.10.	ステーション番号変数	22-19
22.3.11.	インデックスレジスタ	22-21
22.3.12.	MODBUS Server 通信	22-23
22.3.13.	通信パラメータの設定	22-25
22.3.14.	装置(COM)との通信状態及びコントロール	22-28
22.3.15.	装置(イーサネット)との通信状態及びコントロール	22-30
22.3.16.	装置(USB)との通信状態及びコントロール	22-39
22.3.17.	装置(CAN Bus)との通信状態及びコントロール	22-40
22.3.18.	リモート HMI の通信状態及びコントロール	22-41
22.3.19.	リモート装置との通信状態及びコントロール	22-46
22.3.20.	ローカル/リモートコントロールの制限	22-49
22.3.21.	通信エラーコード	22-49
22.3.22.	ドライバ ID	22-52
22.3.23.	DLT645 コントローラ	22-52
22.3.24.	[Device No Response] ウィンドウのコントロール	22-52
22.3.25.	[ファーストセレクション] ウィンドウのコントロール	22-53
22.3.26.	EasyAccess	22-54
22.3.27.	EasyAccess2.0	22-54

22.3.28.	リモートプリンター/バックアップサーバー	22-55
22.3.29.	パススルー通信の設定	22-56
22.3.30.	VNC のコントロール	22-57
22.3.31.	HMI 識別キーとプロジェクト識別キー	22-58
22.3.32.	USB セキュリティキー	22-58
22.3.33.	ユーザー名とパスワード	22-59
22.3.34.	マクロ	22-60
22.3.35.	入力オブジェクトの機能	22-61
22.3.36.	時刻同期/夏時間	22-62
22.3.37.	セルラーネットワーク	22-63
22.3.38.	Wi-Fi 設定	22-65
22.3.39.	OPC UA サーバー	22-67
22.3.40.	E メール	22-67
22.3.41.	cMT Viewer と診断器	22-69
22.3.42.	CODESYS 情報	22-71
22.3.43.	他の機能	22-71
23.	HMI がサポートするプリンターの種類	23-1
23.1.	サポートするプリンターの種類	23-2
23.2.	一台のプリンターを新規追加し、プリントをトリガーする方法	23-5
23.3.	cMT HMI でネットワークプリンターをインストールする	23-8
23.4.	cMT HMI で PPD ファイルをインストールする	23-11
24.	Recipe Editor	24-1
24.1.	概要	24-2
24.2.	レシピデータ/拡張メモリーエディタ設定	24-2
24.3.	レシピ記録の設定	24-4
25.	EasyConverter	25-1
25.1.	概要	25-2
25.2.	データサンプリングファイルを Excel にエクスポートする	25-2
25.3.	イベントログファイルを Excel にエクスポートする	25-4
25.4.	操作ログファイルを Excel にエクスポートする	25-5
25.5.	マルチファイル変換	25-6
25.6.	スケーリング機能	25-7
25.7.	バッチファイルの規則	25-8
26.	EasyPrinter	26-1
26.1.	概要	26-2
26.2.	EasyPrinter をプリンターPrinter サーバーにする	26-3
26.2.1.	EasyPrinter 設定手順	26-3
26.2.2.	EasyBuilder Pro の設定手順	26-4

26.3.	EasyPrinter をバックアップサーバーにする	26-7
26.3.1.	EasyPrinter の設定手順	26-7
26.3.2.	EasyBuilder Pro の設定手順	26-8
26.4.	EasyPrinter の操作説明	26-11
26.4.1.	ウインドウインターフェース	26-11
26.4.2.	操作ガイド	26-12
26.5.	変換バッチファイル	26-16
26.5.1.	変換バッチファイルのデフォルト	26-17
26.5.2.	特定した基準	26-18
26.5.3.	変換バッチファイルのフォーマット	26-18
26.5.4.	実行する順序	26-19
27.	EasySimulator	27-1
27.1.	概要	27-2
27.2.	EasySimulator を設定する手順	27-2
28.	EasySystemSetting	28-1
28.1.	EasySystemSetting	28-2
28.2.	操作手順	28-5
29.	パススルー機能	29-1
29.1.	概要	29-2
29.2.	COM ポートパススルーのイーサネットモード	29-2
29.2.1.	仮想 COM ポートドライバをインストールする手順	29-2
29.2.2.	仮想 COM ポート番号を変更する手順	29-3
29.2.4.	仮想 COM ポートドライバを更新する手順	29-4
29.2.5.	イーサネットモード設定	29-5
29.3.	COM ポートパススルーの COM ポートモード	29-7
29.3.1.	COM ポート設定	29-7
29.3.2.	Utility Manager を使用する	29-7
29.3.3.	システムレジスタを使用する	29-9
29.4.	パススルー通信のコントロール	29-10
29.5.	SIEMENS S7-200 PPI と S7-300 MPI パススルー機能設定	29-10
29.5.1.	EasyBuilder Pro 設定	29-10
29.5.2.	S7-200 PPI の接続方式	29-11
29.5.3.	S7-300 MPI の接続方式	29-11
29.5.4.	SIEMENS パススルーに関連するレジスタ	29-14
29.6.	イーサネットパススルー	29-14
30.	プロジェクト保護機能	30-1
30.1.	概要	30-2
30.2.	EXOB パスワード	30-2

30.3.	デコンパイル禁止	30-3
30.4.	EXOB ファイルをアップロード禁止機能.....	30-3
30.5.	識別キー	30-3
30.6.	EMTP パスワード	30-4
31.	cMT Diagnoser.....	31-1
31.1.	概要	31-2
31.2.	cMT 診断器の使用方法	31-2
31.3.	cMT 診断器のユーザーインターフェース	31-4
31.3.1.	オブジェクト設定	31-4
31.3.2.	装置設定	31-7
31.3.3.	パケット設定	31-7
31.3.4.	マクロ設定	31-10
31.3.5.	MQTT 設定	31-17
31.4.	補足説明	31-19
32.	FTP サーバーの運用	32-1
32.1.	概要	32-2
32.2.	FTP サーバーにログインする手順	32-2
32.3.	履歴データのバックアップ及びレシピデータの更新	32-3
33.	EasyDiagnoser.....	33-1
33.1.	概要	33-2
33.2.	設定手順	33-2
33.3.	EasyDiagnoser 設定	33-3
33.3.1.	メインメニュー	33-3
33.3.2.	通信記録エリア	33-4
33.3.3.	ポーリングパッケージ	33-6
33.3.4.	装置	33-7
33.3.5.	出力(Macro debug)	33-8
33.4.	エラーコード	33-8
33.5.	ウインドウ調整	33-10
34.	Rockwell Ethernet / IP Free Tag Names.....	34-1
34.1.	概要	34-2
34.2.	ユーザー定義 AB Tag CSV ファイルを EasyBuilder Pro にインポートする	34-2
34.3.	データ型を新規追加する手順	34-4
34.4.	貼り付け機能を実行する手順	34-7
34.5.	他の機能	34-8
34.6.	Module-Defined.....	34-9
35.	EasyWatch.....	35-1
35.1.	概要	35-2

35.2. 設定	35-3
35.2.1. 基本機能	35-3
35.2.2. クイックセレクションツール	35-4
35.3. 監視オブジェクト設定	35-5
35.3.1. 監視オブジェクトを新規追加する	35-5
35.3.2. 監視オブジェクト設定	35-5
35.3.3. 監視オブジェクトを新規追加する手順	35-6
35.4. マクロオブジェクト設定	35-9
35.4.1. マクロオブジェクトを新規追加する	35-9
35.4.2. マクロオブジェクト設定	35-9
35.4.3. マクロ設定を新規追加する	35-9
35.5. HMI マネージャ	35-10
35.5.1. HMI 設定をオープンする	35-10
35.5.2. HMI マネージャ	35-11
35.6. オブジェクト表示リスト	35-12
35.6.1. オブジェクト表示欄	35-12
35.6.2. ページ設定	35-13
36. 管理者ツール	36-1
36.1. 概要	36-2
36.2. ユーザーアカウント	36-2
36.2.1. ユーザーアカウントの紹介	36-2
36.2.2. ユーザーアカウント設定	36-4
36.2.3. EasyBuilder Pro でアカウントをインポートする	36-5
36.3. USB セキュリティキー	36-6
36.3.1. USB セキュリティキーの紹介	36-6
36.3.2. USB セキュリティキー設定	36-7
36.3.3. EasyBuilder Pro で USB セキュリティキーを設定する	36-8
36.4. E メール SMTP サーバー	36-9
36.4.1. E メール SMTP サーバー設定	36-10
36.5. E メール連絡先	36-11
36.5.1. E メール連絡先の紹介	36-11
36.5.2. E メール連絡先設定	36-12
36.5.3. EasyBuilder Pro を使用し、E メール設定と連絡先をインポートする	36-13
37. MODBUS TCP / IP ゲートウェイ機能	37-1
37.1. 概要	37-2
37.2. アドレスマッピングテーブルを作成する方法	37-2
37.3. アドレスマッピング設定での注意点	37-4
38. EasyDownload	38-1

38.1. 概要	38-2
38.2. 設定	38-2
39. データセキュリティ	39-1
39.1. 概要	39-2
39.2. 設定	39-2
39.2.1. ワードアドレス属性設定	39-3
39.2.2. ビットアドレス属性設定	39-4
40. ウェブストリーミング	40-1
40.1. 概要	40-2
40.2. 設定	40-2
40.3. 操作方式	40-2
40.4. 対応機種	40-3
41. エネルギー管理	41-1
41.1. エネルギー需要量設定	41-2
41.1.1. 概要	41-2
41.1.2. 設定	41-2
41.2. エネルギー需要量表示	41-5
41.2.1. 概要	41-5
41.2.2. 設定	41-5
42. IIoT	42-1
42.1. MQTT	42-2
42.1.1. 概要	42-2
42.1.2. 設定	42-2
42.2. OPC UA サーバー	42-26
42.2.1. 概要	42-26
42.2.2. 設定	42-26
42.2.3. 装置統計資料	42-33
42.2.4. サポート及び制限	42-34
付録 A. 各シリーズ HMI のソフトウェアの相違点	A-1
付録 B. cMT Viewer 使用制限	B-1
付録 C. PC シミュレーション使用制限	C-1
付録 D. 64-bit データ型の使用においての注意事項	D-1

Windows®は米国 Microsoft Corporation の、米国およびその他の国における登録商標または商標です。

Android は Google Inc.の、米国およびその他の国における登録商標または商標です。

iOS は、米国およびその他の国における Cisco 社の商標または登録商標です。

本ドキュメントに記載されている各社名、製品名及び商標は、一般に各開発メーカーの登録商標あるいは商標です。

本ドキュメントの記載内容は、予告なく変更する場合があります。

1. EasyBuilder Pro のインストールについて

本章では、EasyBuilder Pro のインストール手順について説明します。

- | | | |
|------|----------------|-----|
| 1.1. | 動作環境 | 1-2 |
| 1.2. | インストール手順 | 1-3 |

1.1. 動作環境

ソフトウェアソース：

Weintekのオフィシャルウェブサイト <http://www.weintek.com> からソフトウェアをダウンロードしてください。利用できる言語バージョン(中国語(簡体字)、中国語(繁体字)、英語、日本語、ドイツ語、イタリア語、韓国語、スペイン語、ロシア語、フランス語、ポーランド語とトルコ語バージョン)及び最新ソフトウェアの更新情報をダウンロードすることができます。

OS(OperatingSystem):

Windows® 7 SP1 (32bit / 64bit)

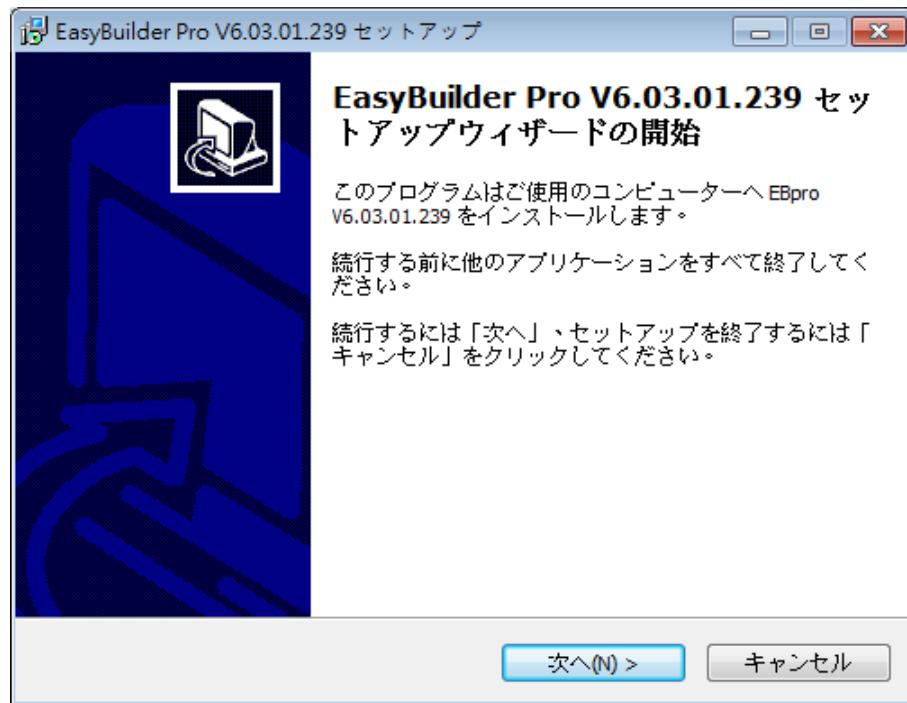
Windows® 8 (32bit / 64bit)

Windows® 8.1 (32bit / 64bit)

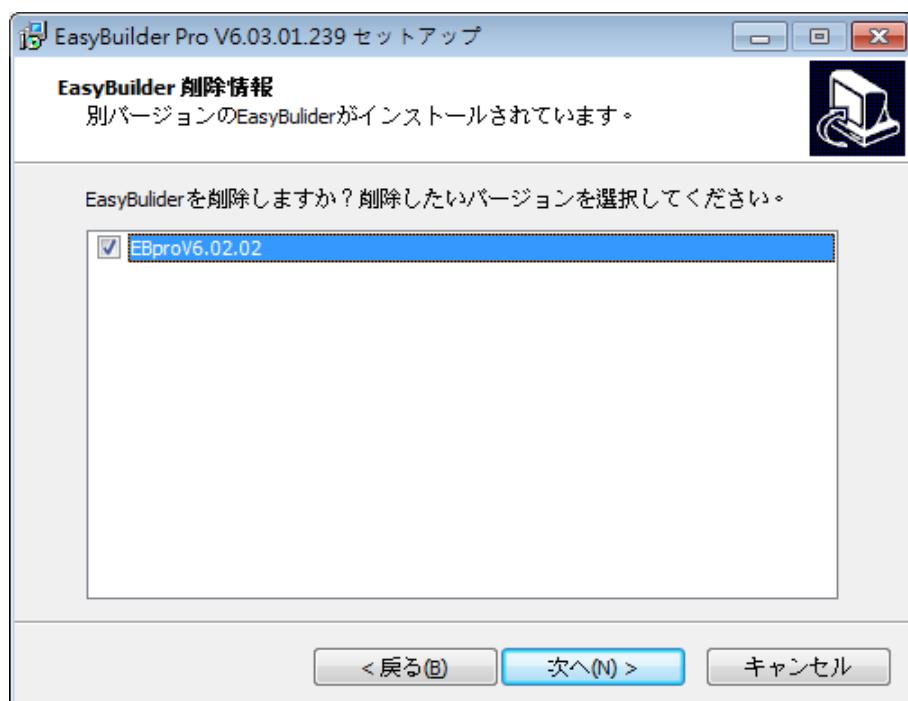
Windows® 10 (32bit / 64bit)

1.2. インストール手順

- Weintek のオフィシャルウェブサイトから EasyBuilder Pro インストールファイルをダウンロードした後、[Install] をクリックし、言語バージョンを選択して、[次へ] をクリックします。

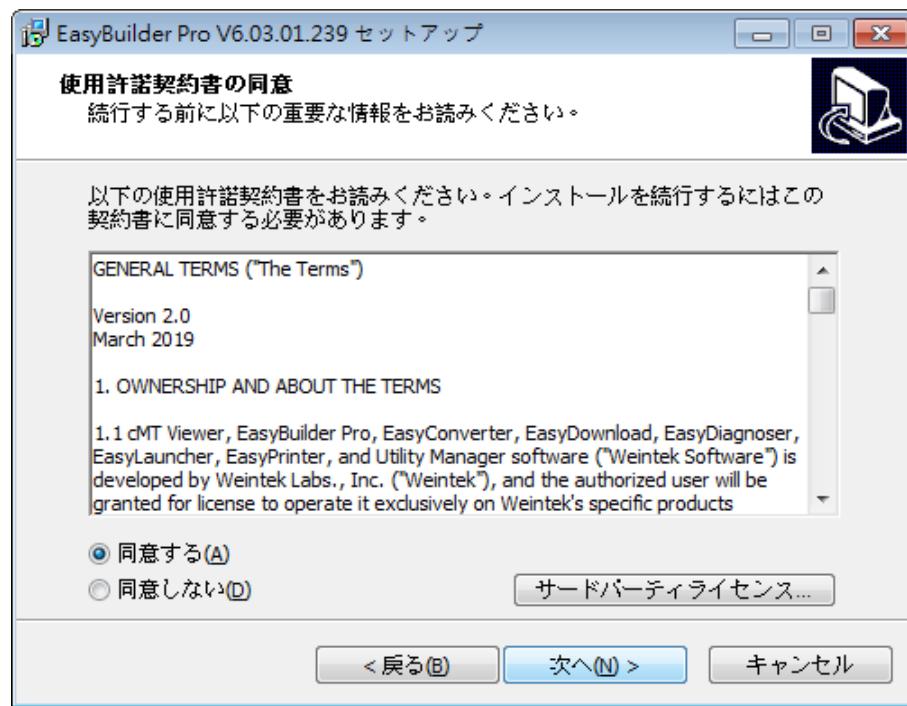


- OS はインストールした EasyBuilder Pro バージョンを削除しますかとユーザーに聞きます。当方の要求により選択し、[次へ] をクリックします。

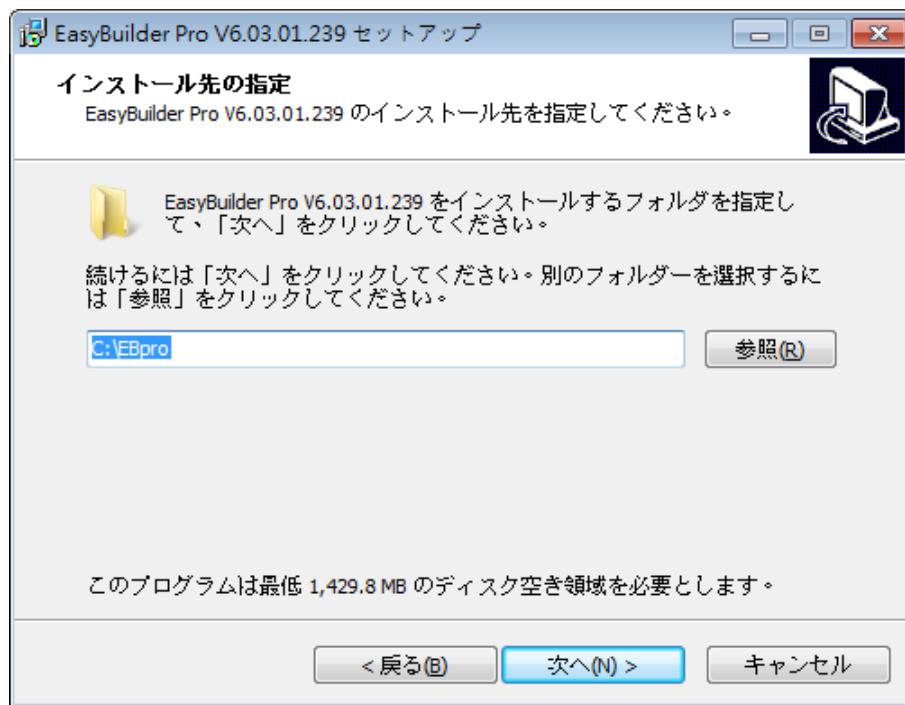


- 続いて「使用許諾契約書の同意」確認ウインドウがポップアップされます。内容をご一読

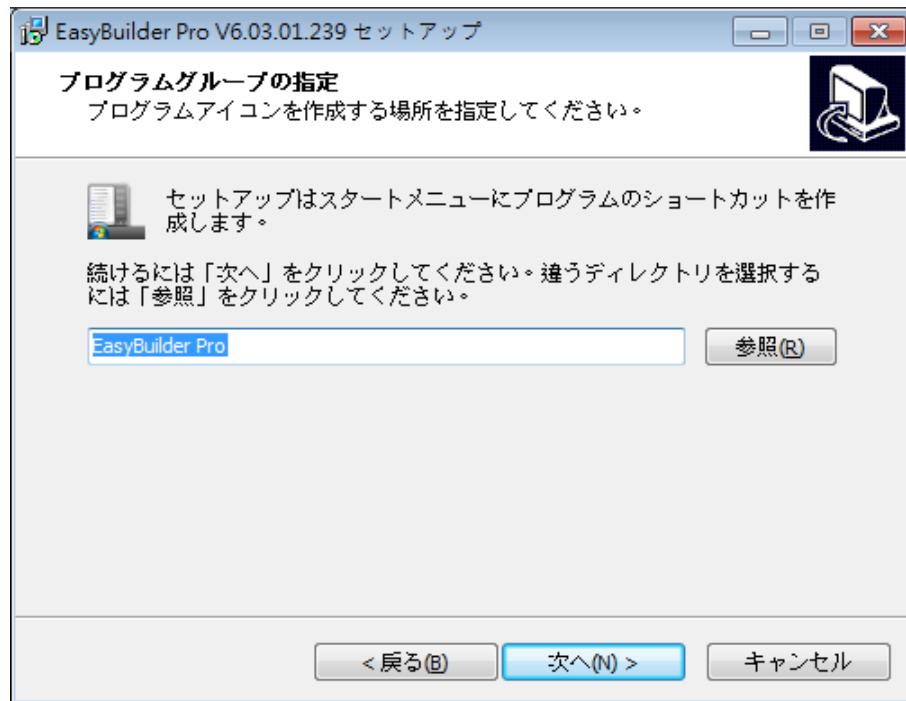
の上、[同意する]を選択し、[次へ]をクリックしてください。



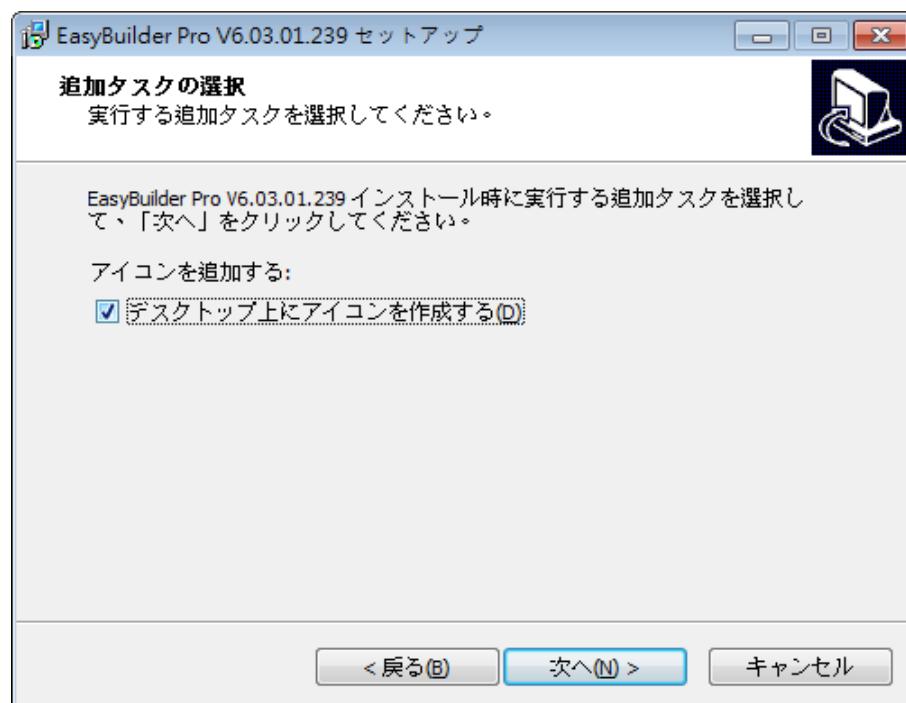
4. EasyBuilder Pro に資料をインストールするため、新たなフォルダを指定してください。或いはそのまま推奨フォルダを使用し、[次へ]をクリックします。



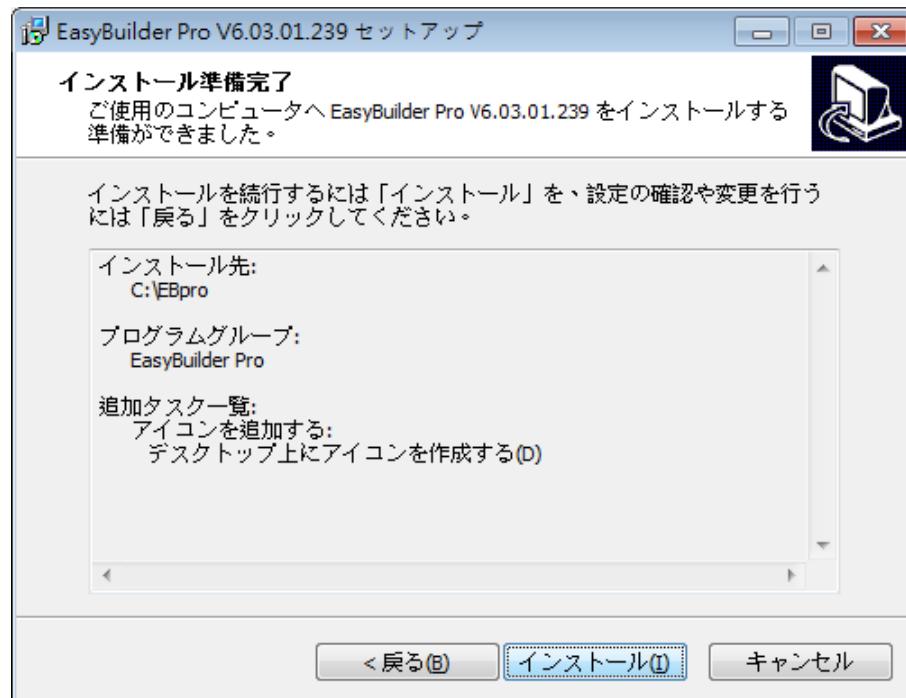
5. OS はスタートメニューのフォルダを指定し、ショートカットを作成します。[参照]をクリックして、インストール先フォルダを選択します。或いは推奨のフォルダを使用し、[次へ]をクリックして進みます。



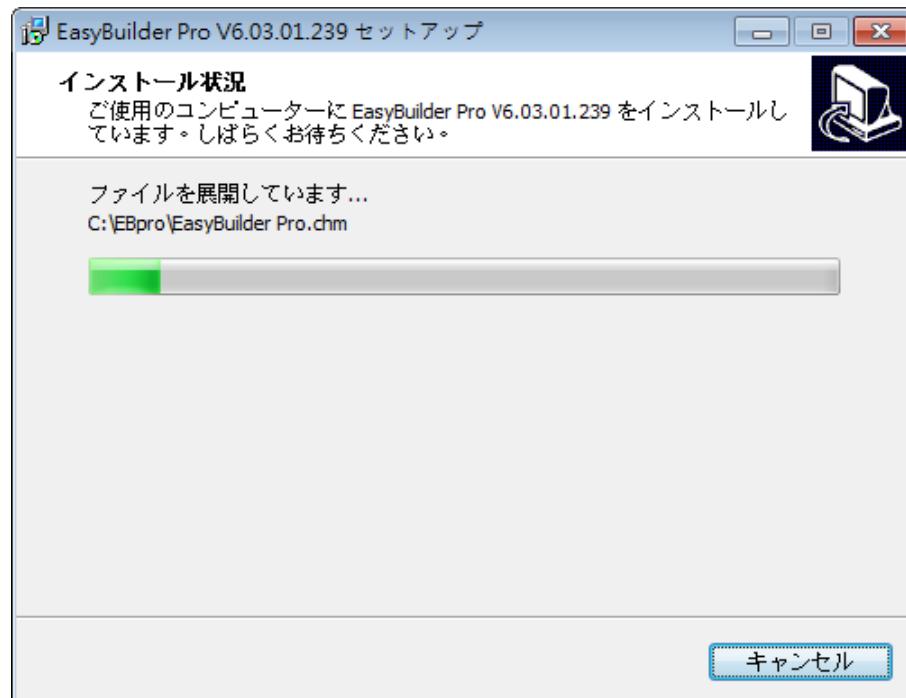
6. OS は追加オプションを選択しますかと聞きます。例えば[デスクトップにショートカットを作成]、必要な場合は選択してください。そして[次へ]をクリックして進みます。



7. ここまででは全ての設定を完了しました。間違ったところがあるのかを検査してください。やり直したいところがありましたら、[戻る]をクリックしてください。間違いがない場合は[インストール]をクリックしてインストールし始めます。



8. インストールしています。



9. インストールが完了したら、[完了]をクリックしてください。



10. [スタートメニュー] » [全てのプログラム] » [EasyBuilder Pro] のメニューに EasyBuilder の各機能のショートカットが見えます。

各ソフトウェアメニューの意味は以下のとおりです:

項目	記述
Administrator Tool	[アカウント]、[USB セキュリティキー]、[E メール SMTP サーバー]、[E メール連絡先]四種類のデータを外部記憶装置(SD カード/USB ディスク)に保存してから、HMI (プログラマブル表示器) にインポートすることができ、データの可搬性及び利便性を向上します。
cMT Viewer	ネットワークを通し、cMT シリーズに接続し、PC で cMT シリーズの HMI を監視、操作できます。
EasyAccess	HMI HMI インターネットを通じて EasyAccess サーバーにログインし、リモートでの HMI 監視・操作を実現できます。
EasyBuilder Pro	HMI がイーサネットを通じ、データをリモート PC のサーバーに転送し、リモート印刷/バックアップを実行する機能です。
EasyConverter	サンプリングデータとイベントログデータのファイル変換ツールです。

EasyDiagnoser	HMI で操作するオンライン監視プログラム及び通信デバッガツールです。
EasyPrinter	HMI がイーサネットを通じ、データをリモート PC のサーバーに転送し、リモート印刷/バックアップの機能を実現します。
EasySimulator	シミュレーションを実行します。
EasySystemSetting	SD カードまたは USB ディスクを使用し、ハードウェアシステムの設定を更新することができます。
EasyWatch	PC で監視、または HMI と PLC のアドレス値を設定し、同時にマクロの使用によりデバッグや遠隔監視も可能になります。
Recipe Editor	レシピデータのファイルフォーマットを設定し、レシピデータ及び外部メモリー内のデータをオープンすることができます。
Release Note	ソフトウェアの最新バージョンと関連情報を記載します。
Utility ManagerEx	EasyBuilder Pro の統括管理ソフトウェアです。



Note

- 部分の HMI は USB ケーブルでプロジェクトをダウンロード/アップロードすることをサポートします。EasyBuilder Pro をインストール完了した後、USB ドライバーが自動的にインストールされます。もし“ドライバー ソフトウェアの発行者を検証できません”メッセージが出たら、インストールを続行するのを選択してください。インストール完了後、[PC マネージャー] » [装置マネージャー] で USB ドライバーをインストールしたかを確認してください。

2. Utility Manager

本章では、Utility Manager の使用方法について説明します。

2.1.	概要	2-2
2.2.	HMI IP 及びパスワードの設定	2-5
2.3.	編集ツール	2-6
2.4.	転送	2-7
2.5.	シミュレーション	2-10
2.6.	パススルー	2-11

2.1. 概要

EasyBuilder Pro のソフトウェアをインストール完了した後、PC デスクトップの UtilityManager ショートカットをダブルクリックして起動します。これは EasyBuilder Pro ソフトウェアの総合管理ソフトです。独立プログラムとして操作することができます。



設定	記述
モデルを選択する	モデルを選択してください。正しく選択していない場合、一部の機能が正常に動作できない場合があるので、ご注意ください。
デザイン	EasyBuilder Pro: EasyBuilder Pro を起動して、プロジェクトを編集します。 EasyAddress Viewer: 各 PLC のアドレス範囲とフォーマットを閲覧します。 シミュレーション: PC でプロジェクトの実行状況をシミュレーションします。
分析・テストツール	EasyDiagnoser: HMI と PLC の間の通信状況。 詳細は《33 EasyDiagnoser》をご参照ください。 EasyWatch: PC で HMI と PLC のアドレス及び数値を監視・設定することができます。



詳細は《35 EasyWatch》をご参照ください。

HMI を再起動:HMI を初期状態に復元します。

パススルー通信設定:PC のアプリケーションソフトが HMI を通して PLC にリンクすることを許可します。



詳細は《29 Pass-through》をご参照ください。

転送

ダウンロード:イーサネットまたは USB ケーブルでプロジェクトファイルを HMI にダウンロードします。

アップロード:イーサネットまたは USB ケーブルでプロジェクトファイルを HMI にアップロードします。

SD カードと USB ディスクに使用するダウロードデータを作成する:作ったデータは SD カード/USB ディスクで HMI にダウンロードすることができます。cMT シリーズは本機能をサポートしていません。

メンテナンス

EasyPrinter, Backup Server:リモートプリンター/バックアップサーバーを起動します。

管理者ツール:[アカウント]、[USB セキュリティキー]、[SMTP サーバー]、[E メール連絡先]四種類のデータを USB デバイスに保存します。cMT シリーズはこの機能をサポートしていません。



詳細は《36 管理者ツール》をご参照ください。

cMT Viewer:CMT シリーズに接続し、CMT のプロジェクトやデータを表示します。

データ/イベントログ情報:USB ケーブルやイーサネットで HMI 内の履歴データファイル数をチェックします。cMT シリーズはこの機能をサポートしていません。

EasyAccess1.0:LAN やインターネットを通して HMI をリモートコントロールすることができます。詳細は www.ihmi.net をご参照ください。

データ変換

レシピデータベースエディター:レシピデータを編集することができます。



このアイコンをクリックし、レシピデータベースの関連資料をダウンロードしてください。

EasyConverter: HMI が取ったデータサンプリングフ

ファイル或いはイベントログファイルを読み取り、そして Excel ファイル(.xls)のフォーマットに変換できます。

 詳細は《25 EasyConverter》をご参照ください。
レシピデータ/拡張メモリーエディター:HMI に使用されるレシピデータを作成でき、既存のレシピデータをオープンや編集することもできます。

 詳細は《24 RecipeEditor》をご参照ください。



Utility Manager のダイアログボックスを最小化します。



Utility Manager を閉じます。



常に使用するツールをダイアログボックス下部のツールバーに追加します。

Run

ツールバーで選択したプログラムを実行します。

Edit

ツールバーで選択したプログラムを削除します。

2.2. HMI IP 及びパスワードの設定

設定

イーサネットや USB ケーブルを通じて HMI を操作する場合、HMI が不正アクセスに侵入されないように、パスワードを設定してください。

パスワード :	*****	<input checked="" type="checkbox"/> マスクする
---------	-------	---

ダウンロード

ダウンロードパスワードを設定してください。[非表示パスワード]を選択すると、パスワードを入力する際に“*”で示します。



- 必ずパスワードをお控えください。パスワードを忘れ、工場出荷時の設定に復元する場合には、HMI 内のすべてのプロジェクトファイルやデータが削除されます。

HMI を再起動する

電源を切断しなくとも HMI を再起動することができ、初期状態に復元します。イーサネットで HMI を再起動する場合は正しい HMI IP アドレスを設定してください。

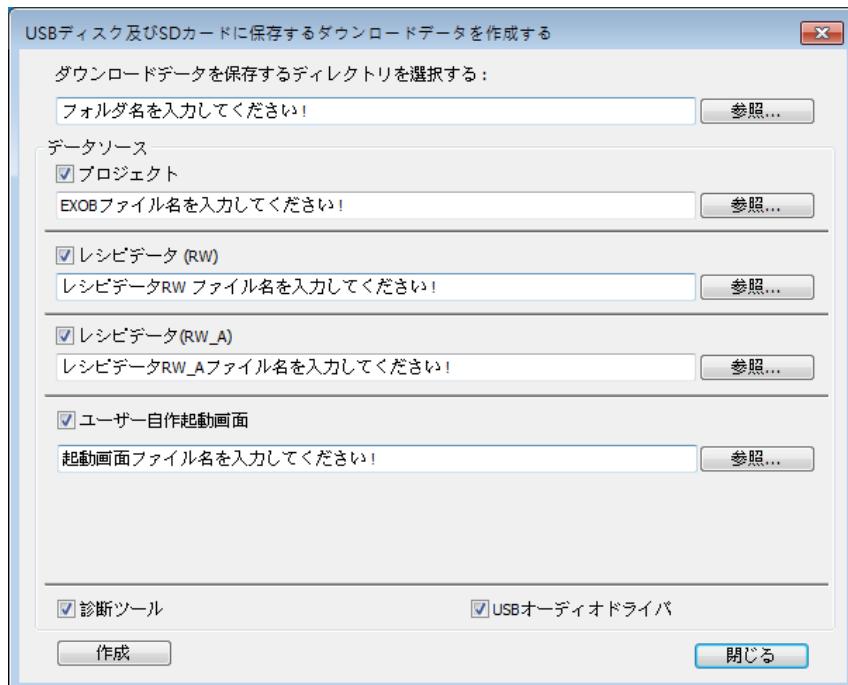
データ/イベントログファイル

接続形式を設定してから、HMI に接続し、履歴ファイルの数をチェックすることができます。



2.3. 編集ツール

2.3.1. SD カード及び USB ディスクに使用するダウンロードデータを作成する



1. SD カードまたは USB ディスクを PC に接続します。
2. データの保存先を指定します。
3. データソースを指定します。
4. [作成]を選択します。

選択したデータが指定された外部記憶装置に書き込まれ、ユーザーはその装置を通してプロジェクトを HMI にダウンロードすることができます。イーサネットもしくは USB ケーブルでダウンロードする必要がなくなります。

2.3.2. USB ディスク/SD カードでプロジェクトを HMI にダウンロードする手順

例として、仮に USB ディスクにあるダウンロードしたいデータを“123”(K:¥123)と名づけた場合

1. USB ディスク(既にプロジェクトを含まれている)を HMI に接続します。
2. [Download/Upload]が表示され、[Download]を選択します。
3. ダウンロードパスワードを入力します。
4. [Download Settings]で[Download project files]及び[Download history files]を選択します。
5. [OK]をクリックします。
6. [Pick a Directory]でダウンロードルート : usbdisk\disk_a_1\123 を選択します。
7. [OK]をクリックします。

プロジェクトが自動的に更新されます。

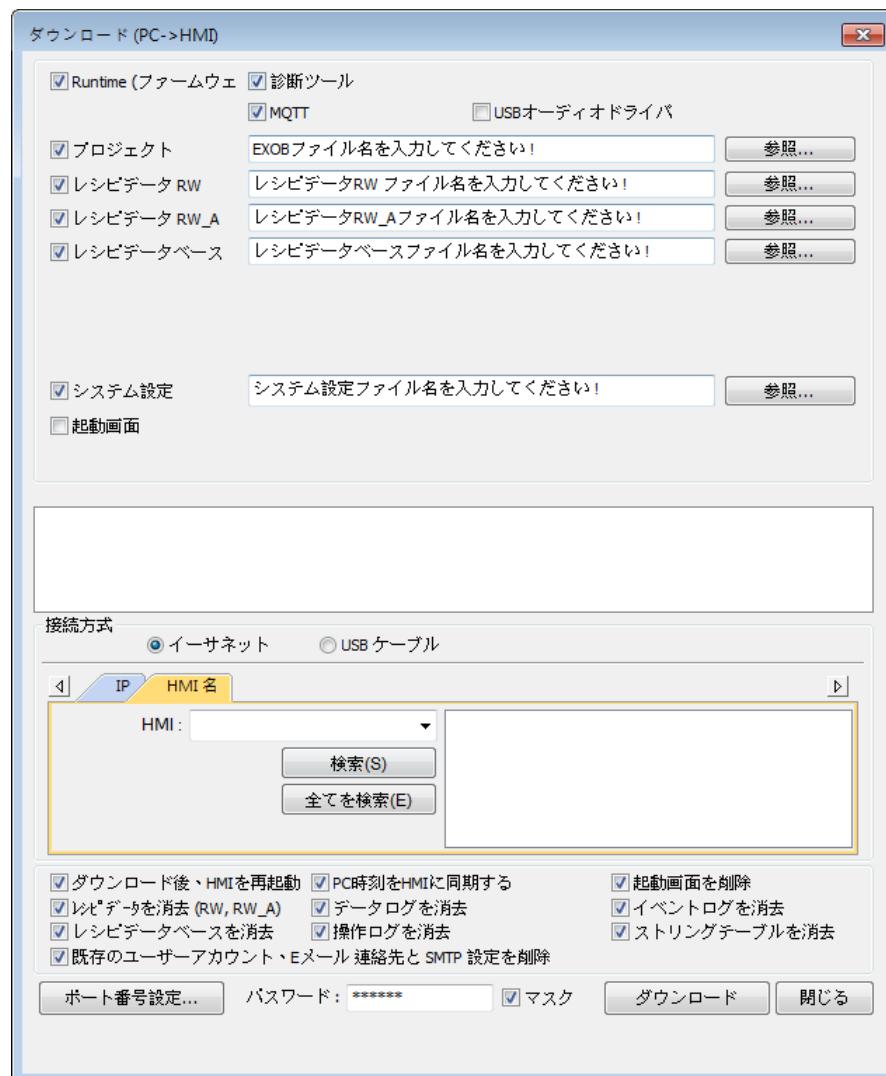
 Note

- プロジェクトをダウンロードしないで履歴データだけをダウンロードする場合、手動で HMI を再起動しデータを更新する必要があります。

2.4. 転送

2.4.1. ダウンロード

本機能を通じ、イーサネットまたは USB ケーブルでデータを HMI にダウンロードすることができます。



設定

ファームウェア

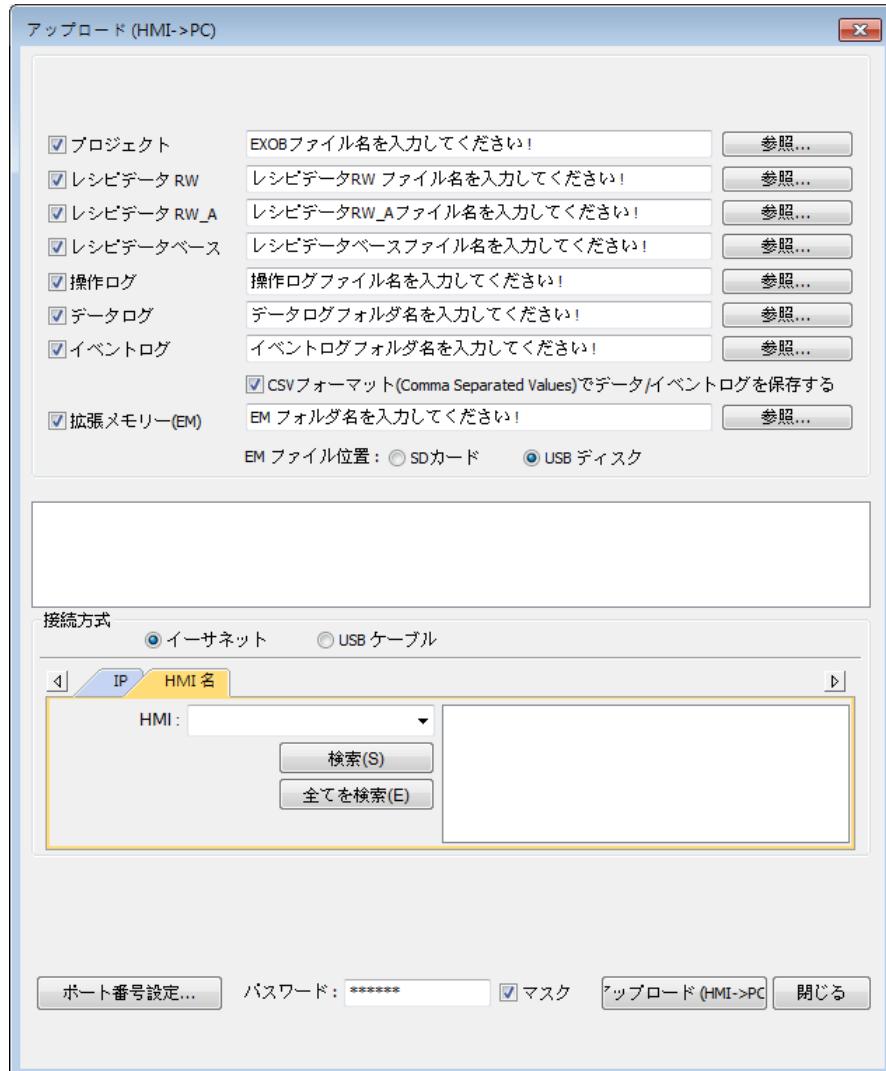
記述

本項を選択すると、HMI の全てのカーネルログ

	ラムを更新することになります。初めてデータを HMI にダウンロードする際に、必ずファームウェアをダウンロードすること。
MQTT	プロジェクト内で MQTT を使用し、それに HMI をブローカーとして使用する場合、ダウンロードした後で MQTT を使用するには、本項にチェックマークを入れる必要があります。
プロジェクト	.exob, .cxob プロジェクトファイルを選択します。
RW/RW_A	.rcp レシピファイルを選択します。
レシピデータベース	.db レシピデータベースファイルを選択します。
システム設定	HMI .conf フォーマットのシステム設定ファイルを選択します。
起動画面	指定した.bmp 画像を HMI にダウンロードしたら、HMI を再起動する時、この画像が起動画面になります。
ダウンロード完了後、HMI を再起動	本項を選択すると、HMI がデータをダウンロード完了後、自動的に再起動します。
ポート番号設定	イーサネットでプロジェクトのダウンロード用ポート番号を設定します。
HMI 時刻を PC と同期	HMI 時刻を PC と同期します。
レシピデータ/レシピデータベース/イベントログ/データログ/ストリングテーブル/操作ログ/起動画面/既存のユーザーアカウント、E メール連絡先、SMTP 設定を削除する	データをダウンロードする前に、先に選択したデータを削除します。

2.4.2. アップロード

本機能を通じて、イーサネットまたは USB ケーブルで HMI のファイルを PC にアップロードできます。アップロードする前、ファイルを保存する場所を指定する必要があります。[参照]をクリックし、保存場所を指定します。

**設定****記述****イベントログ**

HMI 内の.event データを PC にアップロードします。

拡張メモリー (EM)

HMI の SD カード、USB ディスク内の.emi データを PC にアップロードします。

 [プロジェクト]、[RW/RW_A]、[レシピデータベース]、[データログ]に関しては本章《2.4.1 ダウンロード》をご参照ください。

 **Note**

- プロジェクトを PC にアップロードする場合、アップロードされたファイルは.exob, .cxob ファイルであるため、ユーザーはファイルを.emtp, .cmtp ファイルにデコンパイルする必要があります。デコンパイルした場合のみ、EasyBuilder Pro で編集することができるようになります。

2.5. シミュレーション

2.5.1. オフラインシミュレーションとオンラインシミュレーション

オフラインシミュレーション-他の装置と接続しなく、PCのみでプロジェクトの運転をシミュレーションします。

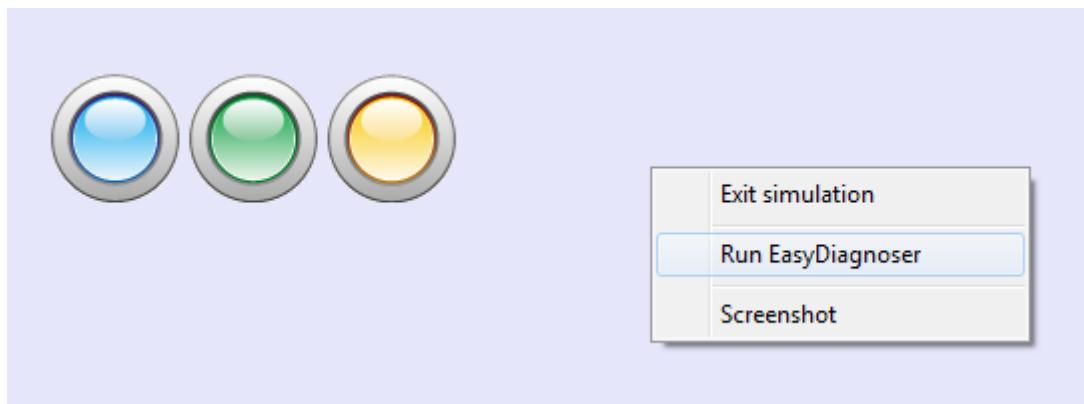
オンラインシミュレーション-PLCと接続したまま、PCでプロジェクトの運転をシミュレーションします。

Note

- PCで[オンラインシミュレーション]をする時、目標装置はローカルPCに接続したPLCである場合、シミュレーション時間は10分間が限界です。

オフラインシミュレーション/オンラインシミュレーションを実行する前、.exobファイルのソースを選択すること。

オフラインシミュレーション/オンラインシミュレーションを実行する時、右クリックしたら下記機能を実行できます：



設定	記述
Exitsimulation	シミュレーションを中止します。
Run EasyDiagnoser	EasyDiagnoserを実行し、通信状態を監視します。
Screenshot	現在のシミュレーション画面を画像ファイルでセーブし、Screenshotフォルダに保存します。

2.6. パススルー

パススルー機能はPC内のアプリケーションソフトがHMIを通してPLCに直接に接続することを可能にする機能です。本機能を使用する際に、HMIの役割はコネクタに似ています。



パススルー機能は[COMポート]モードと[イーサネット]モードがあります。

[イーサネット]モードを使用する前、仮想 COM ポートドライバをインストールしてください。

詳細は《29 パススルー機能》をご参照ください。

3. プロジェクトを作成する

本章では、プロジェクトの作成手順について説明します。

3.1.	概要	3-2
3.2.	新規プロジェクトを作成する	3-2
3.3.	プロジェクトを保存/コンパイルする	3-4
3.4.	オフラインシミュレーションとオンラインシミュレーション	3-5
3.5.	cMT Viewer	3-6
3.6.	プロジェクトを HMI にダウンロードする	3-7
3.7.	HMI からプロジェクトをアップロードする	3-13

3.1. 概要

新規プロジェクトを作成する基本的な手順は以下のとおりです：

1. 新規プロジェクトを作成する。
2. プロジェクトを保存/コンパイルする。
3. オンラインシミュレーション/オフラインシミュレーションを実行する。
4. プロジェクトを HMI にダウンロードする。

以下は、各手順について説明します。

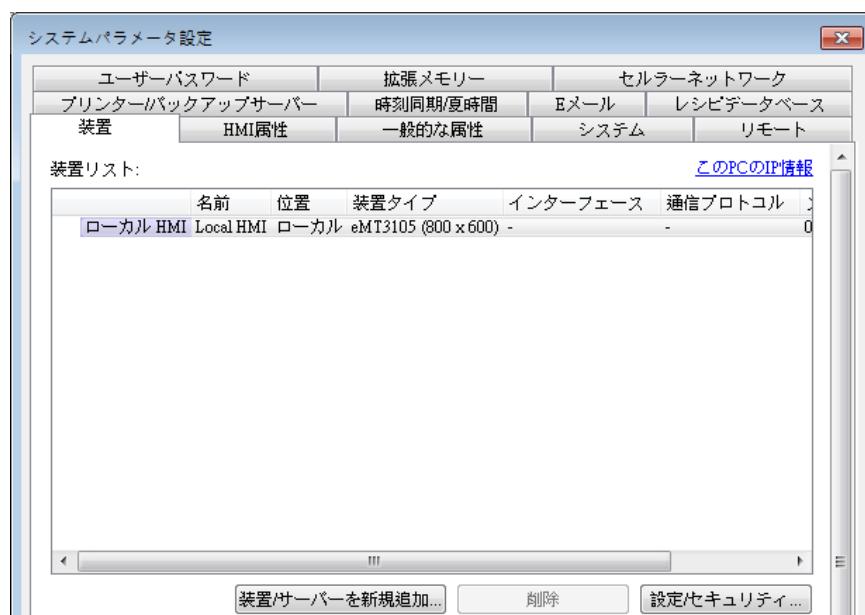
このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

3.2. 新規プロジェクトを作成する

1. EasyBuilder Pro を起動し、新しいファイルをオープンします。
2. [モデル]を選択し、[テンプレートを使用する]にチェックマークを入れてください。

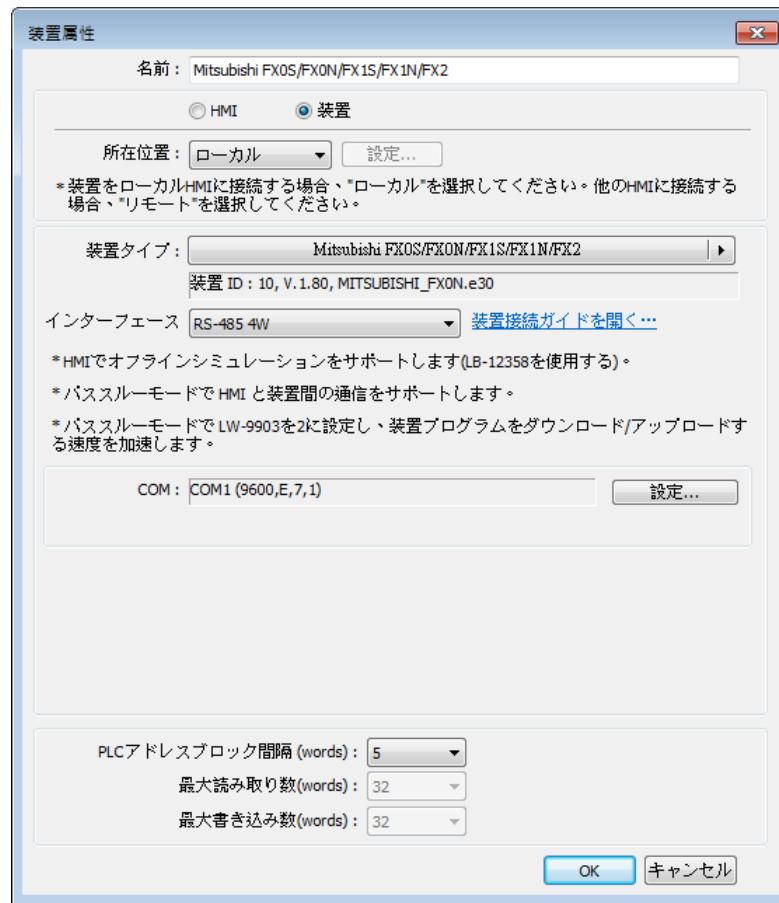


3. [装置]タブで[装置サーバーを新規追加]をクリックし、接続する装置を設定します。



プロジェクトを作成する

4. パラメータを設定します。

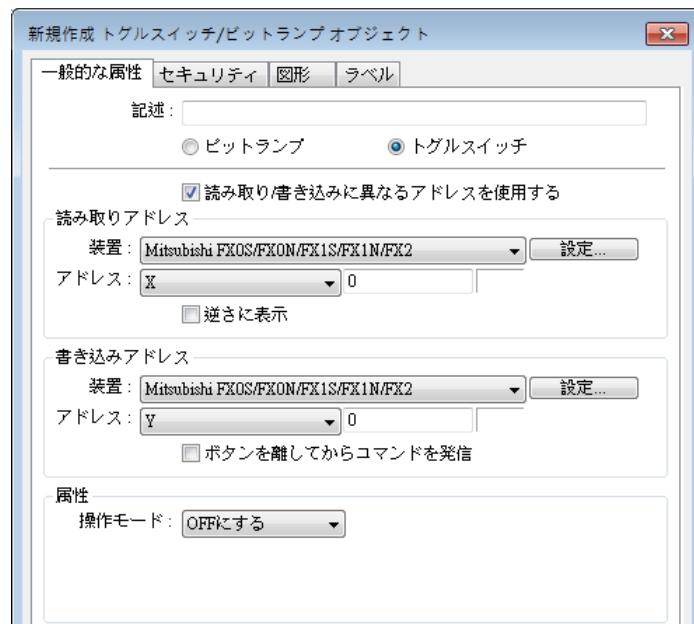


5. [装置リスト]タブに新しい装置が追加されました。

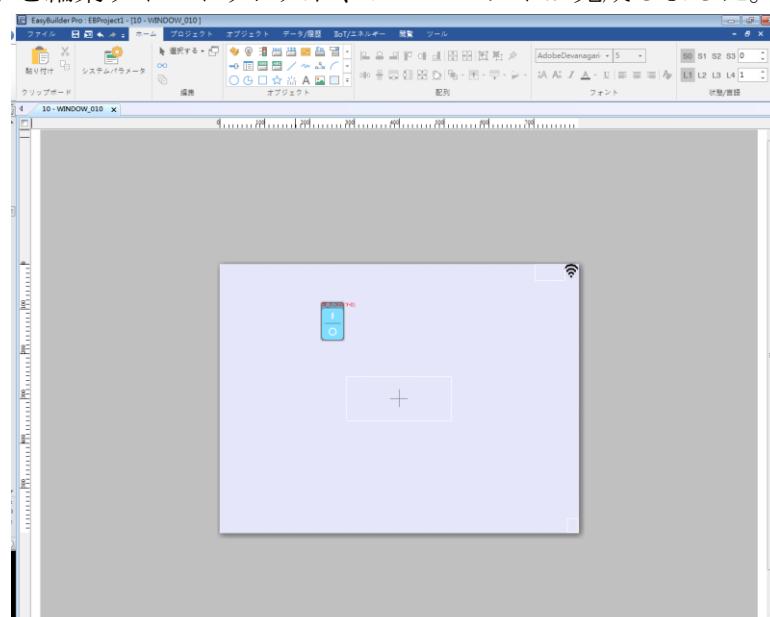


6. 新規オブジェクトを作成します。[トグルスイッチ]を例として、アドレスを設定します。

プロジェクトを作成する



7. オブジェクトを編集ウインドウに入れ、プロジェクトが完成しました。



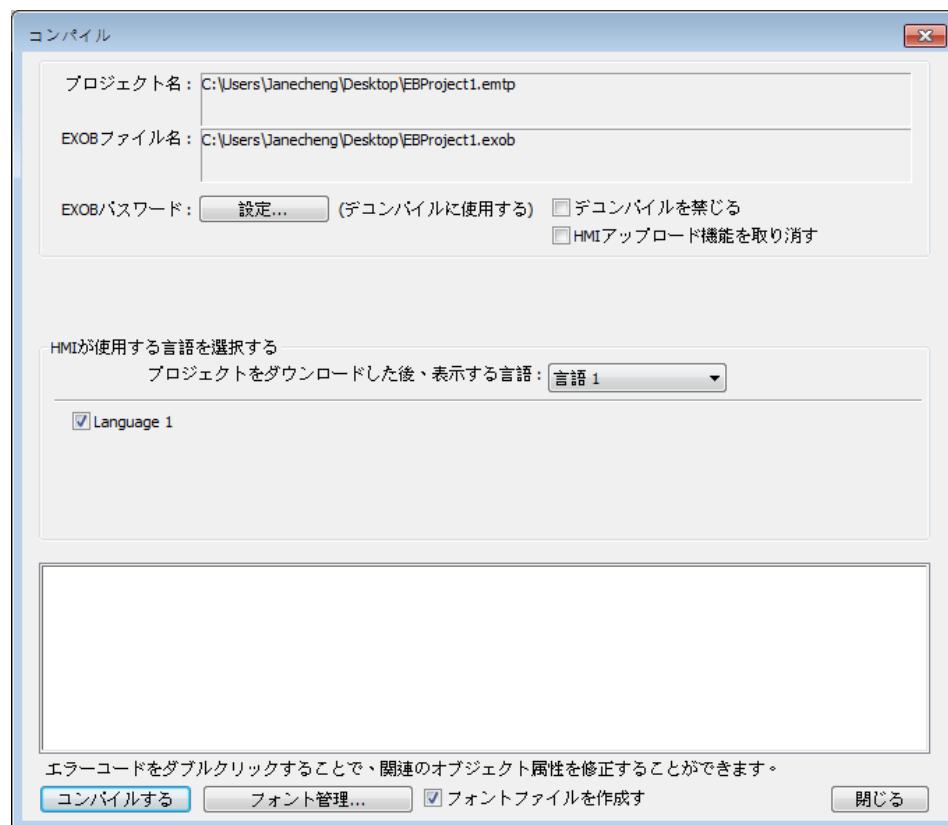
3.3. プロジェクトを保存/コンパイルする

1. EasyBuilder Pro のツールバーで [ファイル] » [保存] をクリックし、.emtp ファイルに保存します。
2. EasyBuilder Pro のツールバーで [プロジェクト] » [コンパイル] をクリックして、.exob ファイルにコンパイルします。このファイルを HMI にダウンロードし、プロジェクトが正常に運転できるのかを検査します。

Note

- cMT シリーズで保存するファイルの拡張子は.cmtp ファイルで、コンパイルするファイルの拡張子は.cxob ファイルです。

3. 多言語機能を使用するには、ユーザーが事前にラベルタグライブラリに多言語を設定してから、プロジェクトが必要とする言語を選択して HMI にダウンロードする必要があります。コンパイルに成功した画面は下記のように示されます：



3.4. オフラインシミュレーションとオンラインシミュレーション



オフラインシミュレーション：他の装置と接続しなく、PC のみでプロジェクトの運転をシミュレーションします。



オンラインシミュレーション：PLC と接続したまま、PC でプロジェクトの運転をシミュレーションします。パラメータを設定してください。プロジェクトを HMI にダウンロードする必要がありません。

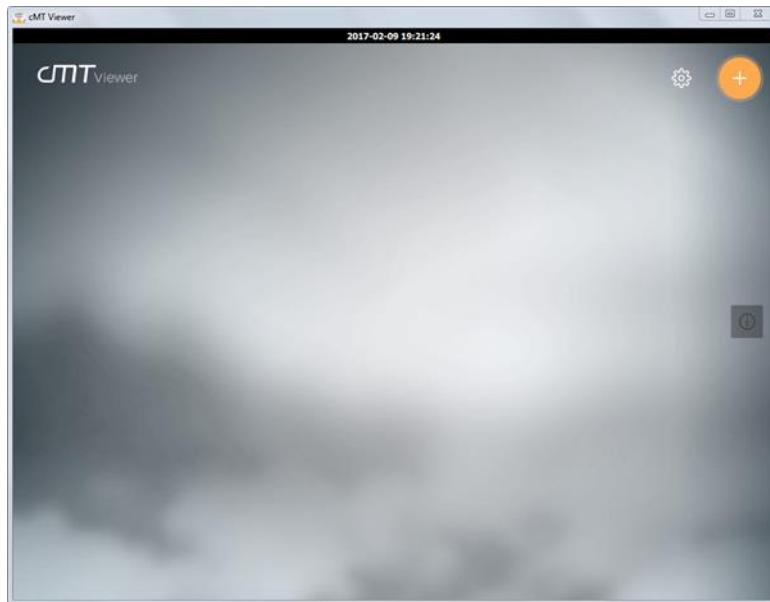
Note

- PC で[オンラインシミュレーション]をする時、目標装置はローカル PC に接続した PLC である場合、シミュレーション時間は 10 分間が限界です。

3.5. cMT Viewer

cMT Viewer はネットワークを利用し cMT シリーズの HMI に接続できます。このプログラムを実行したい場合、インストールディレクトリで cMTViewer.exe を実行するか、または EasyBuilder Pro のツールバーで[ツール] » [cMT Viewer]をクリックします。

▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。





3.6. プロジェクトを HMI にダウンロードする

以下はプロジェクトを HMI にダウンロードする 4 種類の方法を紹介します。

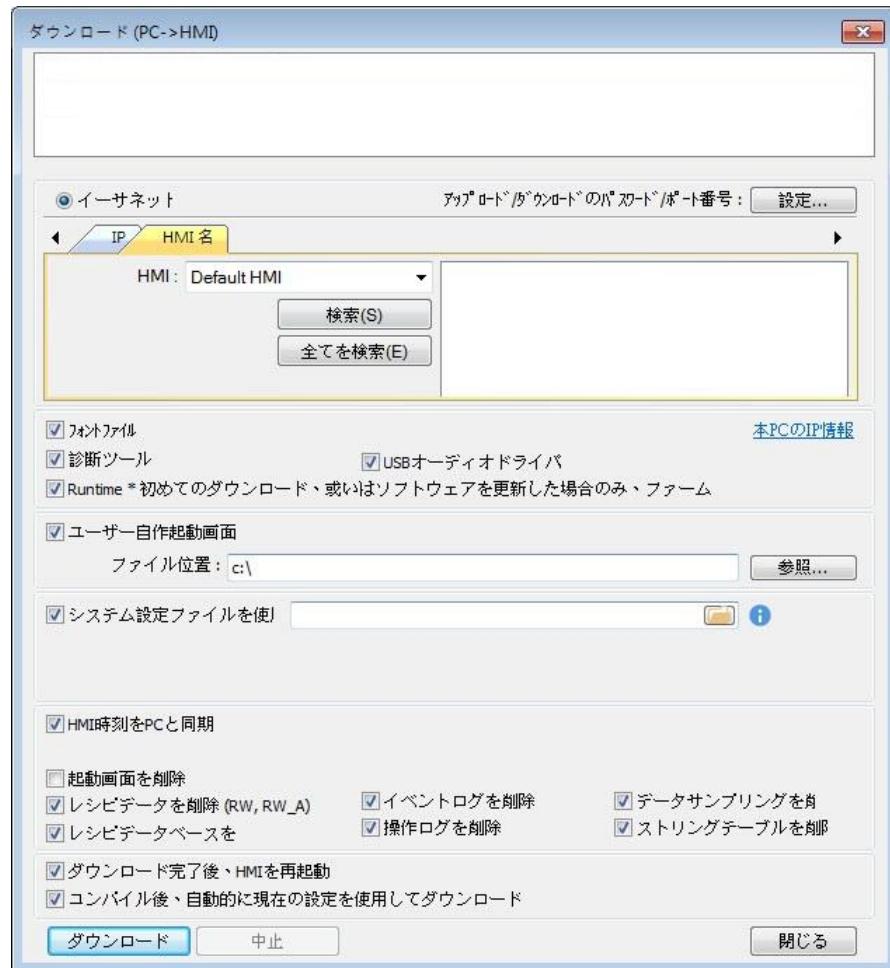
Note

- cMT-SVR は 3.6.1 のダウンロード方法のみ適用します。
- mini USB ケーブルでプロジェクトを HMI にダウンロードする時、PLC のノイズを避けるため、PLC を同時に接続しないでください。
- ▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

3.6.1. EasyBuilder Pro から設定する

1. EasyBuilder Pro のツールバーで、[プロジェクト] » [ダウンロード]をクリックします。全ての設定が正確しているかを先に確認してください。
2. [イーサネット]を選択し、[パスワード]を設定して[HMI IP]を指定します。

プロジェクトを作成する



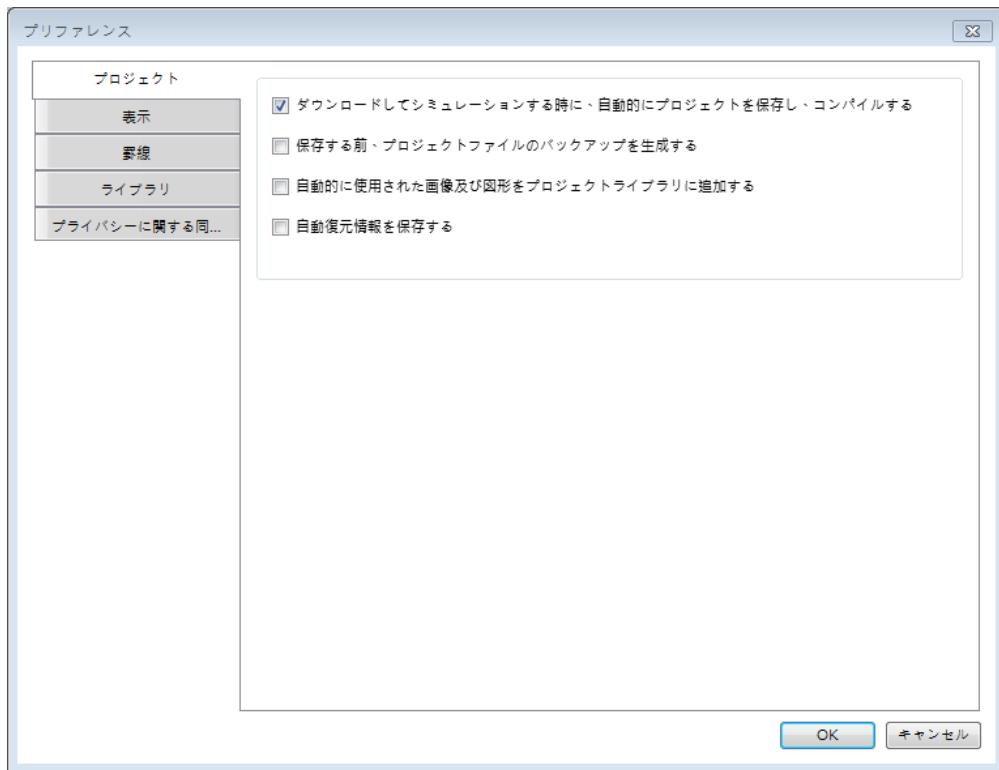
設定	記述
フォントファイル	プロジェクトで使用するフォントを HMI にダウンロードします。
ファームウェア (Runtime)	本項目を選択すると、HMI の全てのカーネルプログラムを更新することになります。初めてデータを HMI にダウンロードする際に、必ずファームウェアをダウンロードすること。
EasyAccess 2.0	EasyAccess 2.0 のドライバを HMI にダウンロードします。(MT8000iE のみ本項目があり、他のモデルは自動的にダウンロードします)
ユーザー自作起動画面	指定した bmp 画像を HMI にダウンロードし、起動画面として使用します。
システム設定ファイルを使用	システム設定ファイルをダウンロードし、それを使ってハードウェア設定を更新します。
HMI 時刻を PC に同期する	プロジェクトをダウンロードしている時、HMI の時刻を PC に同期します。

既存のユーザー アカウント、E メール連絡先と SMTP 設定を削除	本項にチェックマークを入れると、プロジェクトをダウンロードする前に、HMI 内既存のユーザー アカウント、E メール連絡先と SMTP 設定が削除されます。[システムパラメータ設定] » [ユーザー パスワード] » [高度セキュリティモード] で [HMI で既存のユーザー アカウントを使用] を選択するか、或いは [E メール] で [HMI で既存の連絡先設定を使用] を選択する場合のみ、本項が有効になります。
レシピデータ / レシピデータベース / イベントログ / データサンプリング / 操作ログ / ストリングテーブル / 起動画面を削除	本項を選択すると、プロジェクトをダウンロードする前に、選択されたデータが削除されます。
ダウンロード完了後、HMI を再起動	本項を選択すると、ダウンロード完了後、HMI が自動的に再起動されます。
コンパイル後、自動的に現在の設定を使用してダウンロード	本項を選択すると、次回から [ダウンロード] を選んだら、EasyBuilder Pro が自動的にプロジェクトをコンパイルし、前回ダウンロードの保存先である HMI にダウンロードします。詳細は下記をご参照。

[コンパイル後、自動的に現在の設定を使用してダウンロード]

本項を選択すると、次回から [ダウンロード] を選んだら、EasyBuilder Pro が自動的にプロジェクトをコンパイルし、前回ダウンロードの保存先である HMI にダウンロードします。

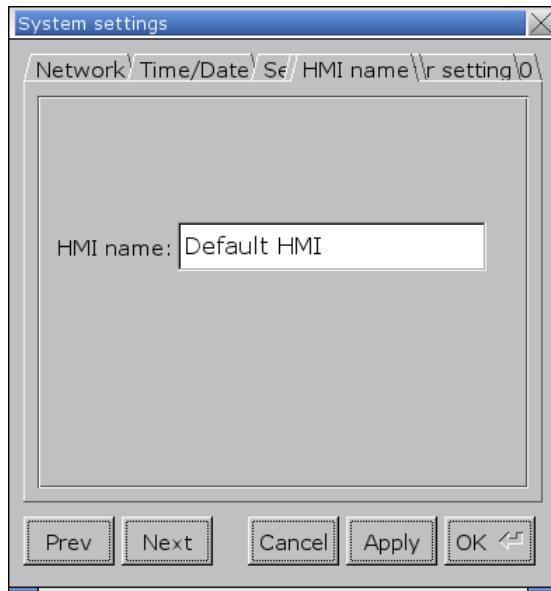
1. EasyBuilder Pro のツールバーで [ファイル] » [プリファレンス] を選択します。
2. [ダウンロードとシミュレーションする前、自動的にプロジェクトをセーブし、コンパイルする] にチェックマークを入れます。



3. EasyBuilder Pro のツールバーで[保存]をクリックしプロジェクトを保存してから、[ダウンロード]をクリックします。
4. ダイアログボックス[ダウンロードとシミュレーションする前、自動的にプロジェクトをセーブし、コンパイルする]にチェックマークを入れます。
5. [ダウンロード]をクリックします。
6. 上記設定を完了したら、次回から[ダウンロード]を選択すると、EasyBuilder Pro が自動的にプロジェクトをコンパイルし、前回ダウンロードの保存先である HMI にダウンロードします。

3.6.2. HMI 名を使用する

1. HMI の System settings で HMI name を設定します。



2. PC で設定された HMI 名を選択してダウンロードし始めます。[検索]を使用する場合、[HMI 名]で特定した HMI 名を検索します。[全てを検索]を選択すると、当ネットワークにある全部の HMI を検索します。



3.6.3. USB ケーブルでダウンロード

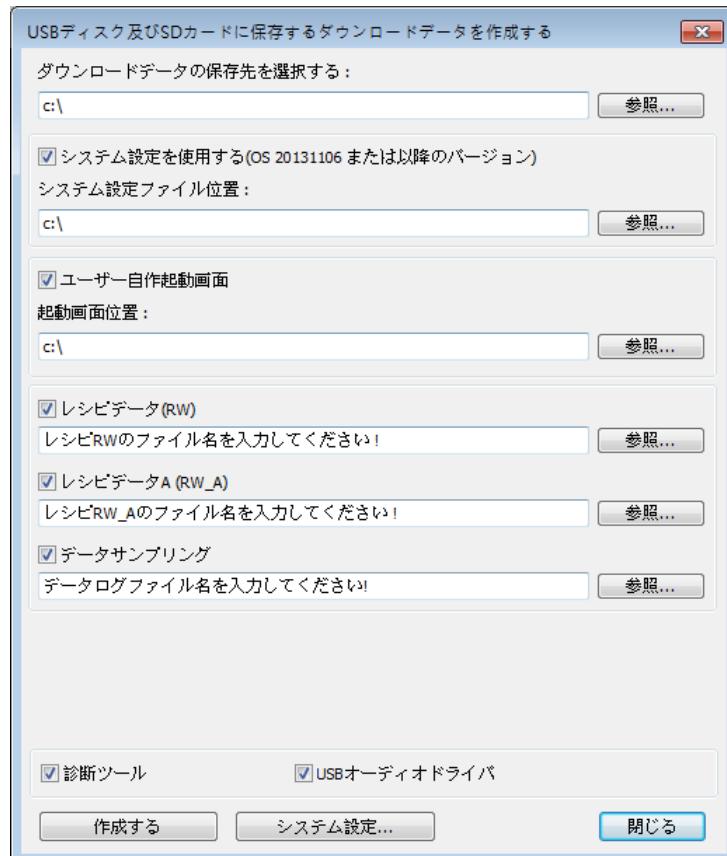


USB ケーブルの使用を選択し、プロジェクトをダウンロードします。他の設定については《3.6.1EasyBuilder Pro から設定する》と同様です。USB ケーブルでプロジェクトを転送する前、[コンピュータマネジメント] » [デバイスマネージャ]で USB デバイスドライバがインストールされたかを確認してください。もしまだインストールしていない場合、[インストール手順](#)を参照し手動的にインストールしてください。

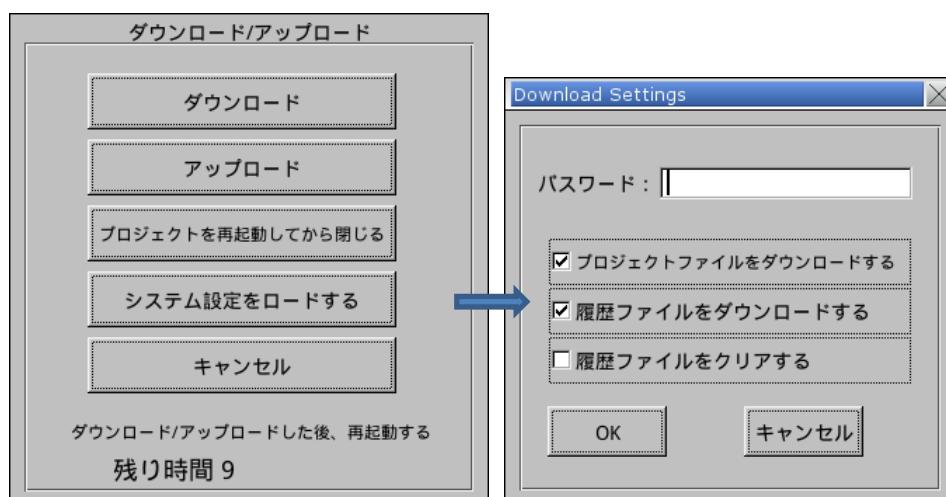
3.6.4. USB ディスク/SD カードを使用する

以下は USB ディスクまたは SD カードでプロジェクトをダウンロードする手順を説明します。

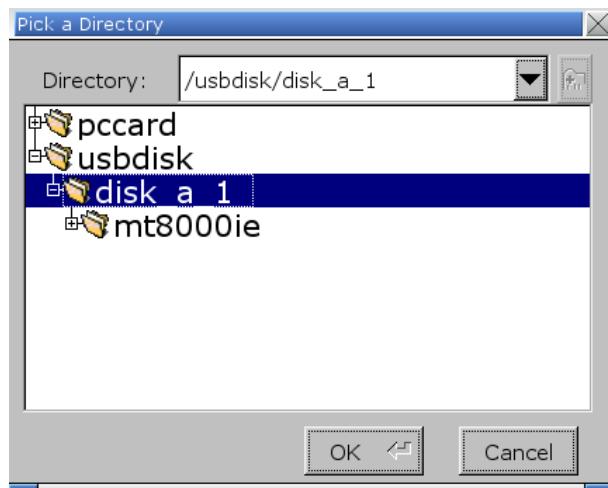
- EasyBuilder Pro のツールバーで、[プロジェクト] » [USB ディスク及び SD カードに保存するダウンロードデータを作成する]を選択します。[参照]でダウンロードしたいプロジェクトを選択してから、[作成する]をクリックしてプロジェクトを外部記憶装置に保存します。



- 外部記憶装置 HMI に接続します。
- HMI で[Download]を選択し、パスワードを入力します。



- パスワードが確認されたら、外部記憶装置のディレクトリが表示されます。(pccard:SD カード ; usbdisk:USB ディスク)



5. プロジェクトの保存先を選択し、[OK]をクリックするとダウンロードが始められます。

Note

- データをダウンロードする際に、保存先を記憶装置のルートディレクトリを選択すること。上記を例として、**mt8000ie**ではなく、**disk_a_1**を選択すること。
 - [システム設定]は EasyBuilder Pro ソフトウェアで HMI ハードウェアの各パラメータを設定完了後、USB ディスク/SD カードで設定を保存し、HMI にダウンロードすることを指します。詳細は《4 ハードウェア設定》をご参照ください。
- このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

3.7. HMI からプロジェクトをアップロードする

1. EasyBuilder Pro のツールバーで、[ファイル] » [アップロード]をクリックします。
2. HMI の IP、モデル、プロジェクト名を設定した後、[アップロード]をクリックすればいいです。

プロジェクトを作成する



4. ハードウェア設定

本章では、ハードウェアの関連設定について説明します。

4.1.	概要	4-2
4.2.	I/O ポート	4-2
4.3.	LED インジケーター	4-2
4.4.	システム設定	4-3
4.5.	システムツールバー	4-4

4.1. 概要

本章は各シリーズのハードウェア関連設定を主として説明します。

このアイコンをクリックし、cMT-SVR シリーズに関する情報をダウンロードしてください。
ダウンロードする前、既にインターネットケーブルを接続しているのを確認してください。

4.2. I/O ポート

HMI がサポートするインターフェースは機種によって異なります。詳細な仕様は各機種の仕様書をご参照ください。

- SD カードスロット : SD カードがレシピデータ、イベントログ、データログなどを含めるプロジェクトのアップロード・ダウンロード機能を提供します。履歴データのコピーや記録もできます。
- COM ポート : PLC または他の装置に接続します。RS-232/RS-4852W/RS-4854W/CAN Bus があります。
- イーサネット : レシピデータ、イベントログ、データログなどを含めるプロジェクトのアップロード・ダウンロード機能を提供します。また、イーサネット通信機能を有する装置：PLC、PC などにも接続できます。
- USB ホスト(Host) : 各種の USB インターフェース装置をサポートします。例えば：マウス、キーボード、USB ディスク、プリンター、バーコードスキャナーなどです。
- USB クライアント(Client) : レシピデータ、イベントログ、データログなどを含めるプロジェクトのアップロード・ダウンロード機能を提供します。履歴データのバックアップまたは記録もできます。

初めて HMI を操作する時、HMI に下記のシステム設定を完成する必要があります。設定完了後、EasyBuilder Pro HMI 編集ソフトウェアでプロジェクトを作成できます。

4.3. LED インジケーター

LED インジケーターが HMI の操作状態を表示します：

機種:MT8121XE, MT8150XE, MT8121iE, MT8150iE

LED	記述
PWR(オレンジ)	パワー状態を表示します。
CPU(緑)	フラッシュメモリーを読み取る/書き込む時には点滅します。
COM(青)	COM ポート通信状態を表示します。毎回通信時には点滅します。通信状況が良好の場合、LED ランプが点灯を維持する可能性もあります。(ネットワーク通信を含めない)

他の機種：

LED	記述
PWR(オレンジ)	パワー状態を表示します。
CPU(緑)	CPU 状態を表示します。点滅や消灯になったら、CPU 異常と示します。
COM(青/赤)	通信状態を表示します。毎回通信時には点滅します。通信状況が良好の場合、LED ランプが点灯を維持する可能性もあります。

4.4. システム設定

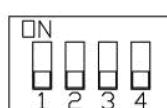
一部の HMI の背面にはリセットボタンとディップスイッチがあります。モード切替の時に使用されます。HMI のシステム設定パスワードを紛失した場合、ディップスイッチで工場出荷時の設定に復元できます。詳細な設定手順は以下のとおりです：

1. DIP Switch1 を ON にし、残りのディップスイッチを OFF のままにしてから、HMI を再起動します。その場合、HMI がタッチ精度補正モードに入ります。
2. モニターに“+”カーソルが表示されます。タッチペンまたは指で“+”カーソルの中心点をタッチし、5 点補正をします。全ての“+”が正確にタッチされたら、“+”が消えます。タッチ精度パラメータがシステムに保存されます。
3. 補正完了後、システムはユーザーに「HMI のシステム設定パスワードを工場出荷時の設定に復元しますか」を聞きます。[Yes]を選択してください。
4. HMI のシステム設定パスワードを工場出荷時の設定に復元するのかを再度確認します。[yes]を入力して[OK]を押すと、HMI 内全てのシステムデータと履歴資料が削除されます。(工場出荷時の[Local Password]初期設定パスワードは 111111；しかし、ダウンロードとアップロードに使用するパスワードを含め、他のパスワードについて、工場出荷時の初期設定に復元したら、改めて設定する必要があります)。

以下は各機種のディップスイッチ設置です。詳細は関連インストール説明書をご参照ください。

cMT/eMT/iE

Dip Switch



SW1	SW2	SW3	SW4	モード
ON	OFF	OFF	OFF	タッチ精度補正
OFF	ON	OFF	OFF	ツールバーを隠す
OFF	OFF	ON	OFF	ブートローダー
OFF	OFF	OFF	ON	予約済み
OFF	OFF	OFF	OFF	ノーマル

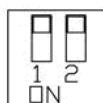
Note

- 各 HMI の Dip Switch4 スイッチの位置が違う可能性があります。Dip Switch4 スイッチが工場出荷時既にカットされた場合、この HMI の Dip Switch4 は ON に設定する必要があると示します。Dip Switch4 スイッチが予約された場合、この HMI の Dip Switch4 は OFF に設定する必要があると示します。

ハードウェア設定

mTV

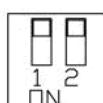
Dip Switch



SW1	SW2	モード
ON	ON	工場出荷時の設定に復元する
ON	OFF	システムツールバーを隠す
OFF	ON	ブートローダー
OFF	OFF	ノーマル

cMT-SVR

Dip Switch



SW1	SW2	モード
ON	ON	工場出荷時の設定に復元する
ON	OFF	イーサネット IP 設定を復元する
OFF	ON	ブートローダー
OFF	OFF	ノーマル

4.5. システムツールバー

HMI を起動後、モニター下部の[ツールバー]でシステムを設定することができます。一般的な状況ではこのツールバーが自動的に隠されているが、ユーザーがモニター右下隅の矢印アイコンを押すと、ツールバーが下記図示のようにポップアップします。左から右は：システム設定、システム情報、文字キーボード、数字キーボードです。



HMI システム設定ツールバーを隠す方法：

- Dip Switch2 を ON にすることで、システム設定ツールバーを隠します。OFF にすると、システム設定ツールバーが表示され、コントロールできます。HMI を再起動することで、本機能を有効/無効にします。
- mTV シリーズの場合、Dip Switch1 を ON にすることでシステム設定ツールバーを隠します。
- また、システムレジスタ [LB-9020] でシステム設定ツールバーを表示/隠蔽できます。
[LB-9020]を ON にする場合、当ツールバーが表示され、OFF にしたら、隠蔽されます。

4.5.1. システム設定

cMT シリーズ

HMI の各システムパラメータを設定・変更したい場合、セキュリティのため、パスワードの確認が必要です。デフォルトパスワードは 111111 です。

ハードウェア設定



CODESYS

CODESYS をアクティベートするには、アクティベーションコードを入力してください。

一般

ここで FTP 設定及び他の一般設定を設定します。

[表示方向]

0 度、90 度、180 度或いは 270 度が選べます。

[明るさ]

モニターの明るさを調整します。

[ダウンロードウインドウをポップアップ]

有効にした後、ダウンロードウインドウをポップアップできます。

[Start button を表示]

Start ボタンを表示するには、本項を有効にする必要があります。



ネットワーク

イーサネットでプロジェクトを HMI にダウンロードするには、正確に操作対象 (HMI) の IP アドレスを設定しなければなりません。自動的に IP アドレスを取得するか、または自分で IP アドレスを入力することができます。Email 及び EasyAccess2.0 を使用したい場合、正確に DNS address を設定する必要があります。

ハードウェア設定



時刻/日付

HMI が表示する現地の日付、時刻を設定します。



HMI名

プロジェクトをダウンロード/アップロードする時に判別用で、HMI 名を設定します。



履歴を消去する

HMI に保存した履歴データを消去します。



プロジェクト管理

再起動ボタンをクリックすれば、HMI を再起動することができます。

ハードウェア設定



システムパスワード

HMI のパスワードです。初期設定パスワードは 111111 で、自分用のパスワードを設定してください。完成したら、自主設定のパスワードが使用できます。



高度セキュリティコード

ユーザー アカウントを新規追加・削除し、異なるユーザーの権限を管理できます。



EasyAccess 2.0

EasyAccess 2.0 をアクティベートします。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。



プリンター

設定の詳細は CH 23.3 をご参照ください。

ハードウェア設定



OS を更新する

SD カードまたは USB ディスク内の OS ファイルでシステムのファームウェアを更新します。更新している時、電源を切らないでください。詳細な更新ステップ及び注意事項に関しては、各モデルの OS 更新説明をご参照ください。

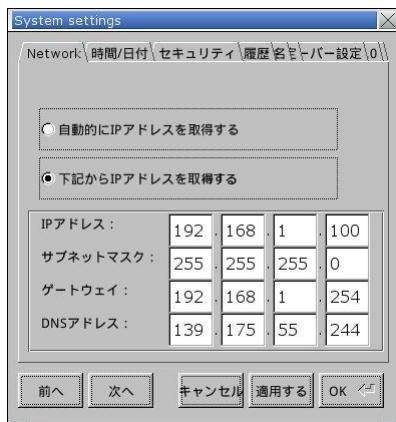


リセットオプション

HMI を工場出荷時の設定にリセットします。

非 cMT シリーズ

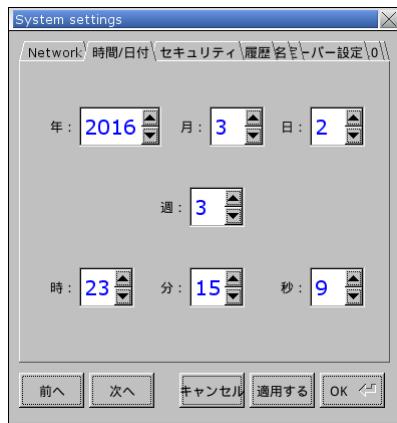
HMI の各システムパラメータを設定・変更したい場合、セキュリティのため、パスワードの確認が必要です。工場出荷時のデフォルトパスワードは 111111 です。



Network

イーサネットでプロジェクトを HMI にダウンロードするには、正確に操作対象 (HMI) の IP アドレスを設定しなければなりません。自動的に IP アドレスを取得するか、または自分で IP アドレスを入力することができます。Email 及び EasyAccess2.0 を使用したい場合、正確に DNS address を設定する必要があります。

ハードウェア設定



時刻/日付

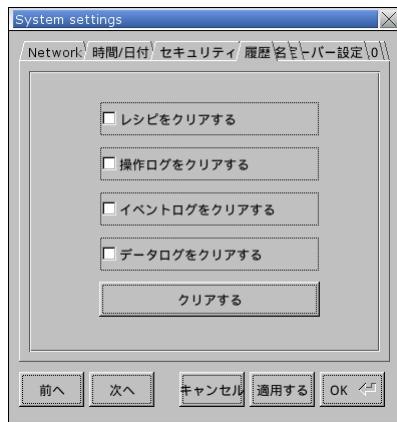
HMI が表示する現地の日付、時刻を設定します。



セキュリティ

HMI のパスワード保護です。デフォルトパスワードは 111111 で、下記ボタンを押し自分用のパスワードを設定してください。完成したら、自主設定のパスワードが使用できます。

- [システム設定に入るパスワード]
- [プロジェクトアップロードパスワード]
- [プロジェクトダウンロードパスワード]
- [履歴データアップロードパスワード]



History

HMI に保存した履歴データをクリア :

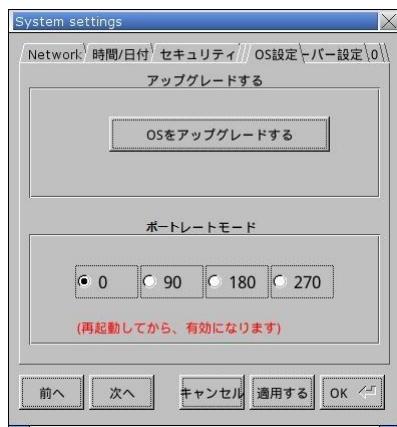
- [レシピデータをクリア]
- [操作ログをクリア]
- [イベントログをクリア]
- [データログをクリア]



HMI 名

プロジェクトをダウンロード/アップロードする時に判別用で、HMI 名を設定します。

ハードウェア設定



OS 設定

[OS をアップグレードする]でシステムのファームウェアを更新します。更新する時、電源を切らないでください。詳細な更新ステップ及び注意事項に関しては、各モデルの OS 更新説明をご参照ください。

[ポートレートモード]で表示モードを調整します。表示モードを調整した後、HMI のプラグを一度抜き、そして再度にプラグを挿して改めて電源を入れてから有効になります。垂直な表示モードを選択すれば(90 度と 270 度)、プロジェクトも垂直なプロジェクトでなければなりません。さもないと、正確に表示できません。



VNC サーバー設定

起動すると、VNC を用いてイーサネットを通して HMI を遠隔操作することができます。

[VNC シングル接続を開始する]

一台の VNC クライアント装置に接続することを許可します。

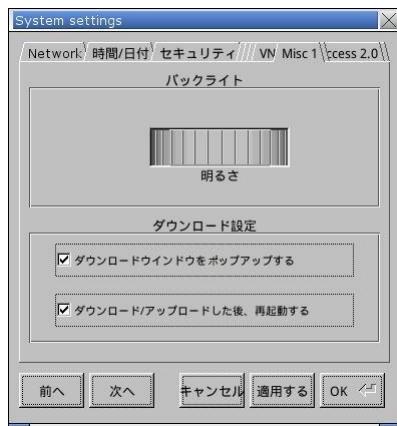
[VNC マルチ接続を開始する]

複数の VNC クライアント装置に接続することを許可します。

同時に多くの VNC クライアントに接続すると、HMI の通信速度に影響が出る可能性があります。

設定手順は次のページをご参照ください。

このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。



Miscellaneous

ロータリースイッチで LCD 画面の明るさを調整します。

[ダウンロードウィンドウをポップアップする]

起動して USB ディスクまたは SD カードを接続後、アップロード/ダウンロードウィンドウを表示します。

[ダウンロード/アップロードした後、再起動する]

起動し、プロジェクトをアップロード/ダウンロード完了後、HMI を再起動します。

**Misc2**

[マウスカーソルを隠す]

マウスカーソルを隠します。

[精度補正モードに[工場出荷時の設定に復元]ボタンを有効にする]

有効にすると、HMI起動時にモニターを長く押せば、精度補正モードに入ることができます。補正し終えると、[工場出荷時の設定に復元]が表示されます。

[FTPクライアントはUSB/SDデータを修正可能]

有効にすると、FTPサーバーを通じてUSB/SDカード内のデータを修正できます。

**EasyAccess2.0**

EasyAccess2.0をアクティベーションします。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

以下はVNCサーバーを設定する手順です。

1. HMIのVNCサーバーをオープンし、ログインパスワードを設定します。
2. Java IEまたはVNC viewerをPCにインストールします。
3. Java IEをインストール完了したら、ウェブサーバーを通して遠隔地にあるHMIのIPアドレスを入力できます。

もしくはVNC viewerを通じて遠隔地にあるHMIのIPアドレスとパスワードを入力します。



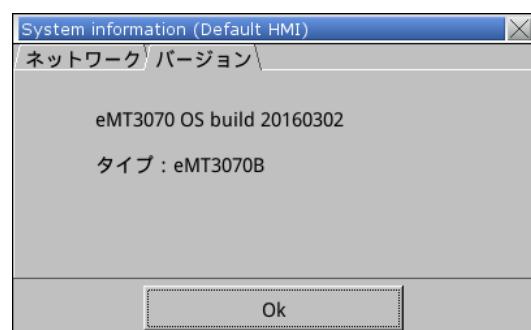
4.5.2. システム情報

ネットワーク:ネットワークの情報を表示します。

(HMI の IP アドレスなどを含む)。



Version:HMI のファームウェアバージョン及び機種情報を表示します。



5. システムパラメータ設定

本章では、EasyBuilder Pro の各システムパラメータの設定について説明します。

5.1.	概要	5-2
5.2.	装置	5-2
5.3.	HMI 属性	5-11
5.4.	一般的な属性	5-14
5.5.	システム	5-20
5.6.	リモート	5-23
5.7.	ユーザーパスワード	5-25
5.8.	拡張メモリー	5-29
5.9.	プリンター/バックアップサーバー	5-30
5.10.	時刻同期/夏時間	5-32
5.11.	E メール	5-34
5.12.	セルラーネットワーク	5-36

5.1. 概要

EasyBuilder Pro のメインメニューで[ホーム] » [システムパラメータ]を選択したら、[システムパラメータ設定]ダイアログボックスが表示されます。以下は各タブについて説明します。

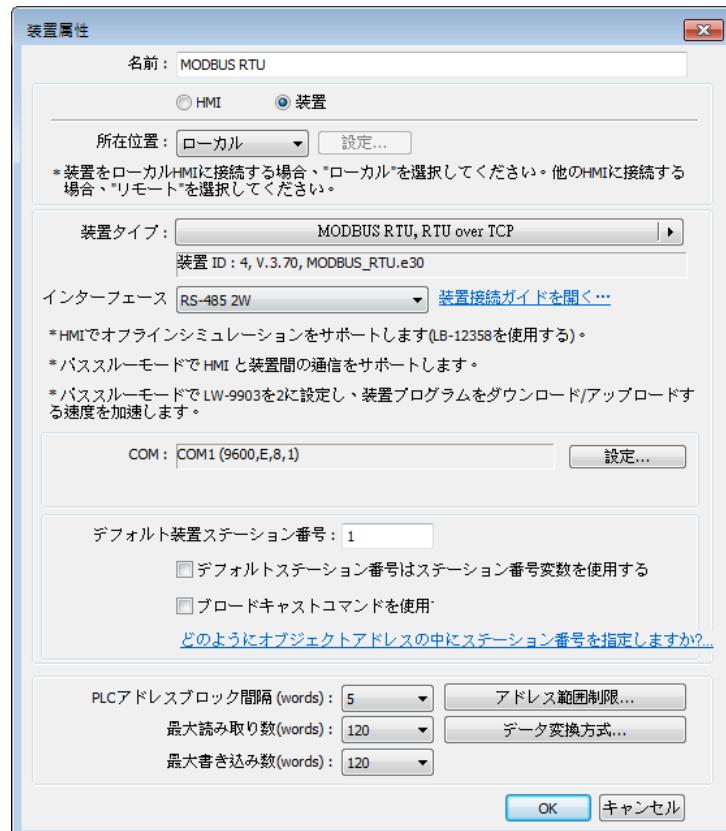
5.2. 装置

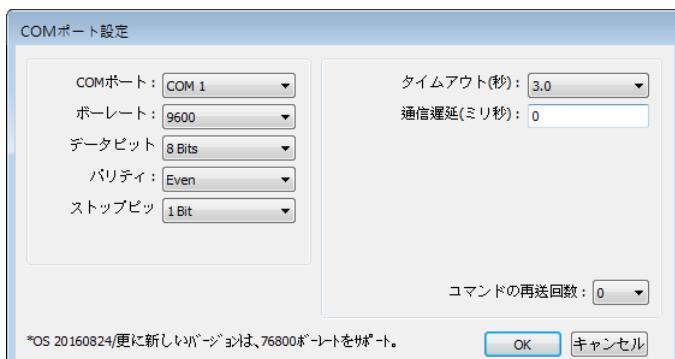
HMI に接続する装置の属性を設定します。これらの装置はローカル HMI、リモート HMI、或いは装置などを含めています。新しいプロジェクトをオープンする時に、初期には機種判別用途で“ローカル HMI”装置が設定されます。

5.2.1. ローカル PLC をコントロールする仕方



“ローカル装置”というのは、ローカル HMI と直接に接続する装置のことを指します。ローカル装置をコントロールしたい時、こういうタイプの装置を追加する必要があります。[装置リスト] タブの[装置/サーバーを新規追加]をクリックしたら、[装置属性] ウィンドウがオープンされます。以下は MODBUS RTU をローカル装置とします：



設定	記述
名前	装置の名前を表示します。
HMI/装置	装置を接続する装置ので、[装置]を選択します。
所在位置	[ローカル]または[リモート]が選択できます。装置がローカル HMI に接続しているので、[ローカル]を選択します。
装置タイプ	装置のタイプを選びます。
インターフェース	装置が使用するインターフェースのタイプです。[RS-232]、[RS-4852W]、[RS-4854W]、[イーサネット]、[USB]及び[CAN Bus]があります。
	<ul style="list-style-type: none"> ● インターフェースのタイプは[RS-232]、[RS-4852W]、[RS-4854W]である場合、[装置属性]ダイアログボックスの[設定]をクリックしたら、[通信ポート設定]ダイアログボックスが開かれ、通信パラメータを設定することができます。
	

タイムアウト

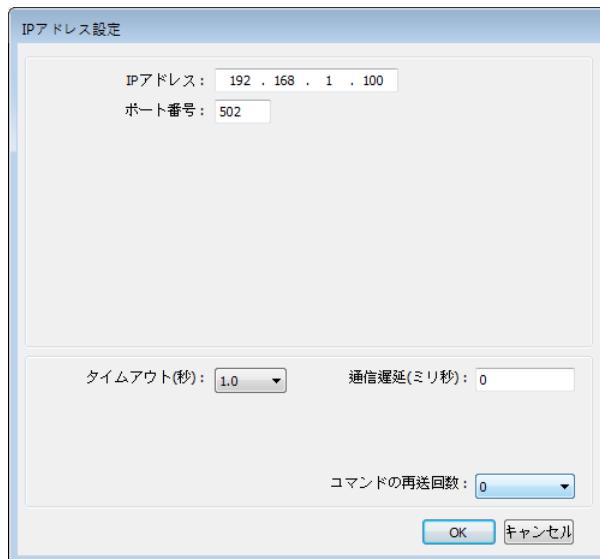
通信の中断時間がこの設定値(単位：秒)を超えたら、HMI が 5 番ウインドウ“Device No Response”をポップアップし注意します。

通信遅延

HMI は次のコマンドを装置に送信する前、この設定値(単位：ミリ秒)を遅延してからコマンドを出します。この設定値は HMI と装置の間の通信効率を低下にするので、特定なニーズがなっかたら、“0”に設定すればいいです。

ただし、使用する装置は SIEMENS S7-200 シリーズである場合、この設定をしなければなりません。[通信遅延]を“5”、[ACK 信号遅延]を“30”に設定することがお勧めです。

- インターフェースタイプが[イーサネット]の場合、[装置属性]ダイアログボックスの[設定]をクリックすると、[IP アドレス設定]ダイアログボックスが表示します。装置の IP アドレスとポート番号を正確に設定してください。



- インターフェースタイプは[USB]の場合、再度に設定する必要がありません。[装置属性]の各設定値が正確であるのかを確認すればいいです。
- インターフェースタイプは[CAN Bus]の場合、《装置接続ガイド》で“CANopen”についての説明を参照し、edsファイルをインポートしてください。

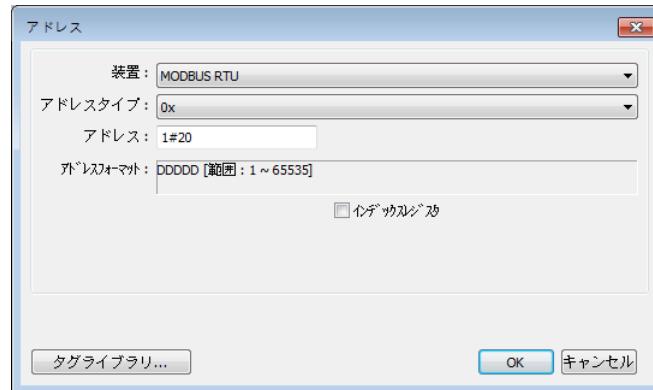
装置デフォルトステーション番号

装置アドレスが使用するデフォルトステーション番号を設定します。アドレス内容にステーション番号が含まれていない場合、この設定値を使用し装置のステーション番号とします。

また装置ステーション情報を直接にアドレス内容に設定することもできます。その場合、アドレスのフォーマットはABC#Addrになります。

ABCはPLCが使用するステーション番号を示し、数値が0~255の範囲にある必要があります。Addrは装置のアドレスを示し、両数値の間に "#" で区切りします。下図のように、装置ステーション1の0x-20アドレスの内容を読み込むと示します。



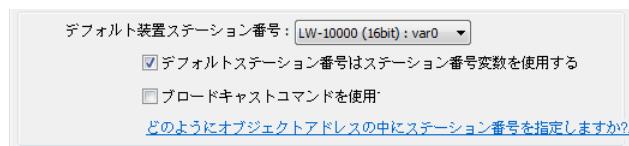


デフォルトステーション番号はステーション番号変数を使用する

ステーション番号変数を装置のデフォルトステーション番号として使用することができます。

LW-10000~LW-10015(var0~var15)を利用してステーション変数を設定します。ステーション番号変数を使用する場合、もし装置アドレスに使用するステーション番号を指定していなかったら、ステーション番号変数で決定します。

例として、装置デフォルトステーション番号に var3 を選ぶと：



下記にて数例を用いて説明します。

- 装置ステーション番号を 5 にする場合：



- 装置ステーション番号を var7(LW-10007)で決定する場合：



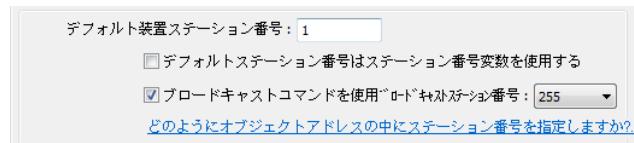
- 装置アドレスが"111"で、装置ステーション番号を指定しなくても、既にステーションデフォルト番号を var3 に指定したので、装置ステーション番号は var3(LW-10003)で決定します。



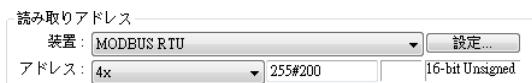
このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

ブロードキャストコマンドを使用する

[ブロードキャストコマンドを使用する]にチェックマークを入れた後、装置が定義したブロードキャストステーション番号を[ブロードキャストコマンドが使用するステーション]に記入してください。HMI がブロードキャストステーション番号を用いてコマンドを送信する場合、装置はただコマンドを受信するだけで、HMI に返信しません。



下図のように示します：



仮にブロードキャストステーション番号を 255 に設定する場合、HMI がコマンドを 255#200 というアドレスに発信する時、全ての装置がこのコマンドを受信するが、HMI に返信しません。

ブロードキャストコマンドをサポートする装置のみ適用します。

装置アドレス ブロック間隔 (words)

異なるコマンドの読み取りアドレスの間隔がこの設定値より小さい場合、これらのコマンドを一つのコマンドに合併することができます。設定値を“0”にすると、コメントを合併する機能が取り消されます。

設定値を“5”にする場合、それぞれ LW-3 から 1 個の word、と LW-6 から 2 個の word データ(即ち LW-6 と LW-7 の内容)を読み取る際に、LW-3 と LW-6 のアドレス間隔の差が 5 より小さいので、両コマンドを一つのコマンドに合併することができます。合併後のコマンド内容は LW-3 から連続の 5 個の word データ(LW-3~LW-7 を読み取り)です。注意すべきなのは、合併できるコマンドの読み取りサイズは[最大読み取り数(words)]より大きくなれません。

最大読み取り 数(words)

一回で装置から読み取られるデータの最大量です。単位は word です。

最大書き込み 数(words)

一回で装置に書き込めるデータの最大量です。単位は word です。

上記の各設定を完了後、[装置リスト]では“ローカル装置 1”という装置が表示されます。



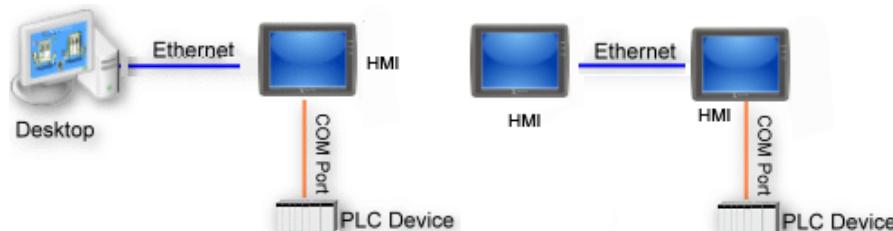
Note

- cMT を使用する時、[システムパラメータ設定] ダイアログボックスで “ローカル HMI” を選択し、[装置を新規追加] ボタンを押すと、下図のように “ローカル HMI” の下に “ローカル装置 1” が追加されます。



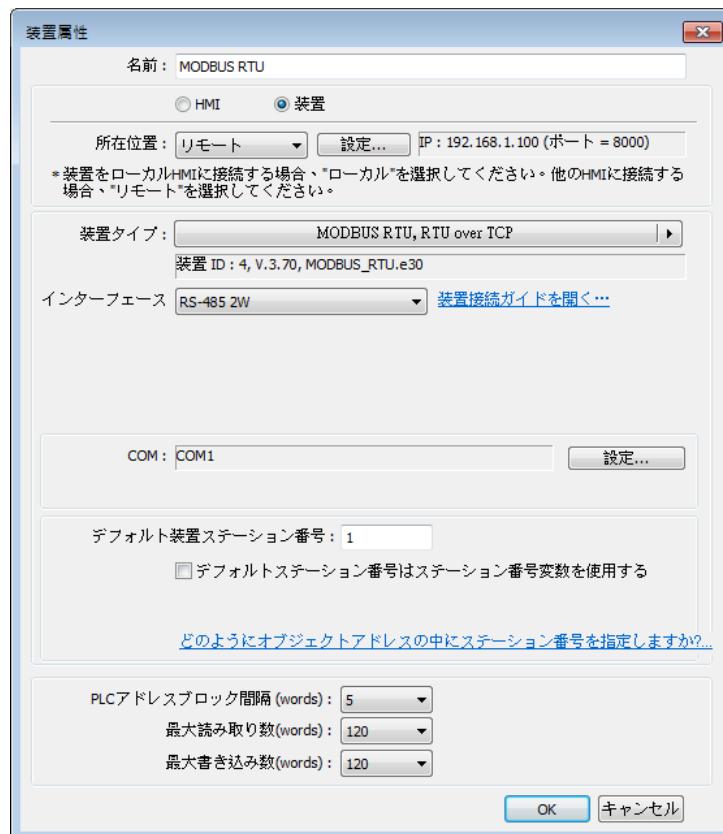
このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

5.2.2. 遠隔地にある装置(リモート装置)をコントロールする仕方



“リモート装置”というのは、遠隔地にある HMI に接続する装置のことです。リモート装置をコントロールしたい場合、まずはこのタイプの装置を追加する必要があります。[装置リスト] タブの [新規追加] をクリックすれば、[装置属性] ダイアログボックスで各属性を設定することができます。

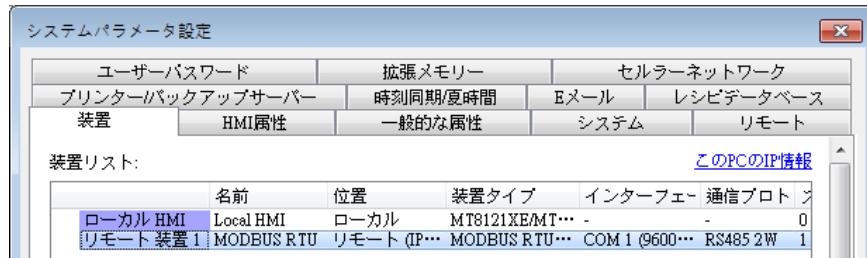
以下は MODBUS RTU を PLC とします。



設定	記述
HMI/装置	装置を接続するので、[装置]を選択します。
所在位置	[ローカル]または[リモート]が選択できます。装置が遠隔地の HMI に接続しているので、[リモート]を選択し、遠隔地の HMI の IP アドレス及びポート番号を入力します。[装置属性] ダイアログボックスで[所在位置]隣の[設定]を押してください。
装置タイプ	リモート装置のタイプを選択します。
インターフェース	リモート装置が使用するインターフェースタイプを選択します。リモート装置が COM ポートを使用している場合、インターフェースは[RS-232]、[RS-4852W]、[RS4854W]のどれかを選択します。
COM	リモート装置がリモート HMI に接続する際に使用するシリアルポートです。正確に設定する必要があります。
装置デフォル	リモート装置が使用するステーション番号。

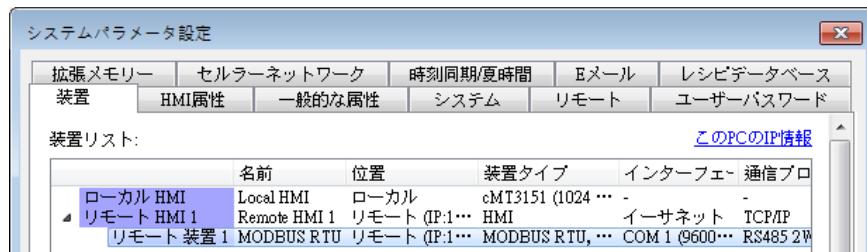
トステーショ ン番号

上記の各設定を終えたら、[装置リスト]に“リモート装置 1”という装置が追加されます。



Note

- cMT シリーズを使用する時、[装置リスト]タブで設置された“リモート HMI1”を選択し[PLC を新規追加]ボタンを押すと、下図のように“リモート HMI1”的下に“リモート装置 1”が追加されます。

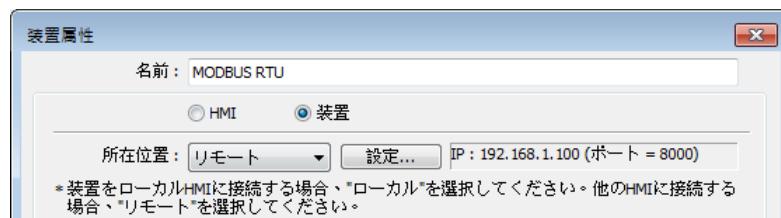


このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

5.2.3. 遠隔地にある HMI(リモート HMI)をコントロールする仕方



“リモート HMI”というのは、ローカル HMI 以外、他の全ての HMI のことを指します。PC もリモート HMI の一種と視されます。リモート HMI をコントロールしたい場合、まずはこのタイプの装置を増加する必要があります。[装置リスト]タブの[新規追加]をクリックすれば、[装置属性]ダイアログボックスで各属性を設定できます。



設定

記述

HMI/装置

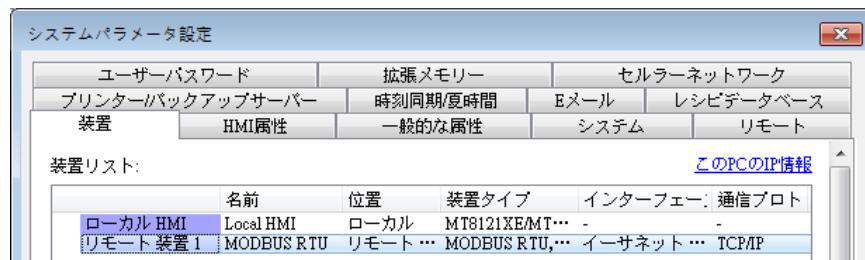
接続装置は HMI なので、[HMI]を選択します。

システムパラメータ設定

所在位置 [ローカル]または[リモート]が選択できます。遠隔地にある HMI を使用するので、[リモート]を選択し、遠隔地の HMI の IP アドレス及びポート番号を設定します。[装置属性] ダイアログボックスでの[所在位置]の隣の[設定]を押してください。

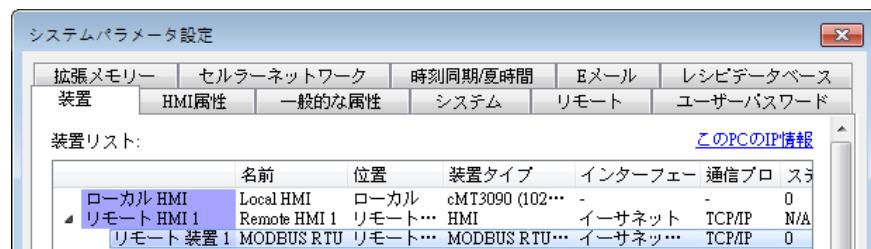


上記の各設定を終えたら、[装置リスト]に“リモート装置 1”が追加されます。



Note

- cMT シリーズを使用する時、[装置リスト]タブで[HMI を新規追加]ボタンを押すと、下図のように“リモート HMI1”が追加されます。



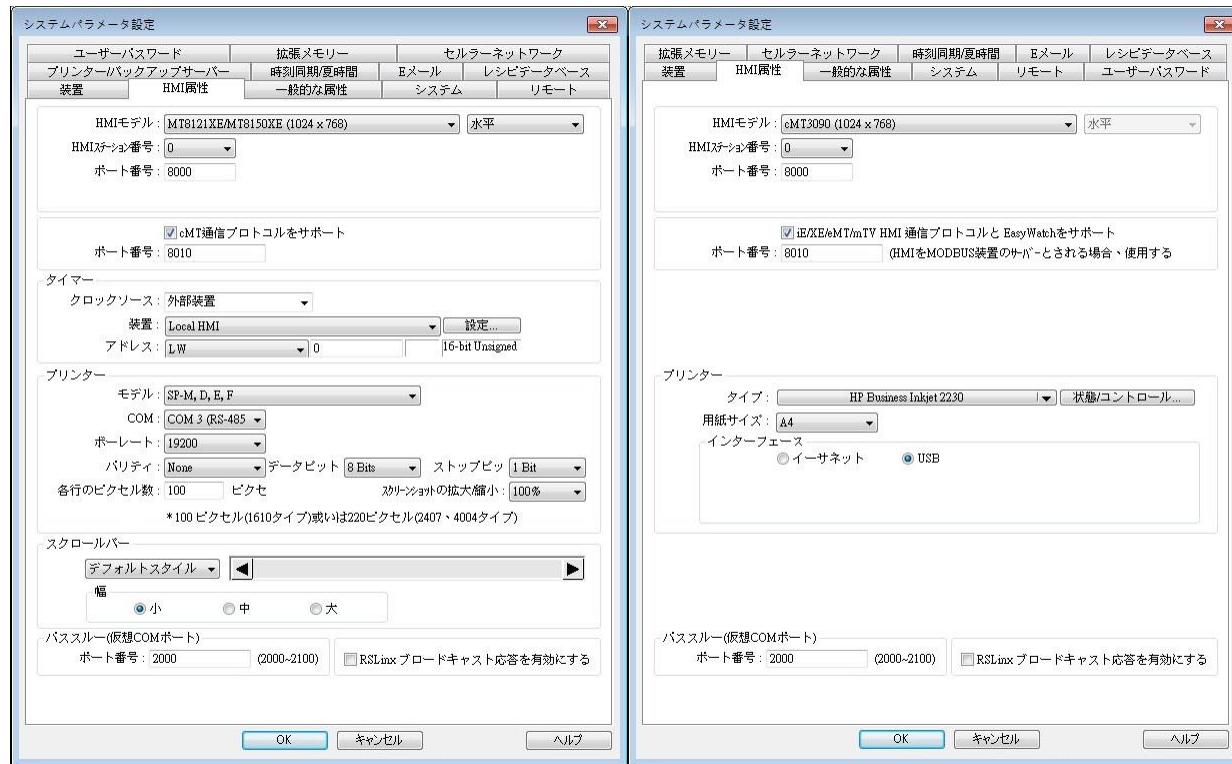
▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

5.3. HMI 属性

[HMI]設定タブで[HMI モデル]、[タイマー]、[プリンター]、[スクロールバー]などを設定することができます。

eMT、iE、XE、mTV シリーズ

cMT シリーズ



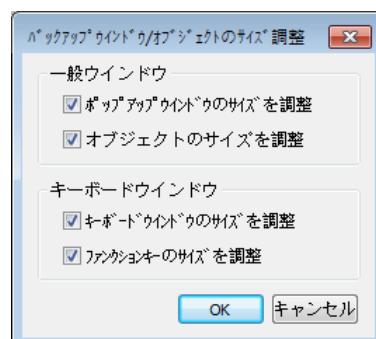
設定

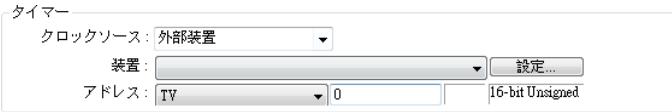
記述

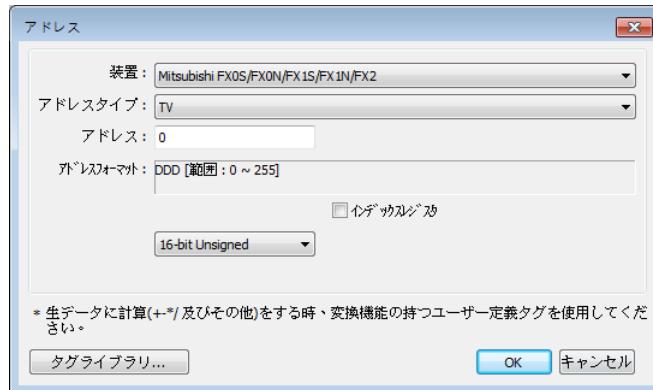
HMI モデル

使用する HMI モデルを選択します。

ユーザーが他のモデルの HMI に変更し確認ボタンを押す時、変更したモデルの解像度が本来のモデルと異なる場合、下図のように[ポップアップウインドウ/オブジェクトのサイズ調整]ダイアログボックスがポップアップされます。確認を押すと、HMI モデルの変更が完了します。



	水平/垂直モード プロジェクトが HMI での表示モードを設定します。
HMIステーション番号	HMI が使用するステーション番号を選択します。特別な目的がなかったら、デフォルト値を使用すればいいです。
ポート番号	HMI が使用するポート番号を選択します。特別な目的がなかったら、デフォルト値を使用すればいいです。
cMT通信プロトコルをサポート	cMT シリーズの HMI と通信することをサポートします。cMT シリーズのプロジェクトも同じの通信ポートを設定する必要があります。
タイマー	クロックソース タイマーの時刻信号のソースを設定します。タイマーの時刻は[データサンプリング]、[イベントログ]など時刻表記が必要のオブジェクトに運用されます。 <ul style="list-style-type: none"> ● [HMI RTC]を選択すると、時刻信号のソースは HMI の内蔵タイマーとなります。 ● [外部装置]を選択すると、時刻信号のソースは外部装置となります。この場合、正確に時刻信号のソースアドレスを設定しなければなりません。下図の設定を例として、時刻信号のソースは“ローカル PLC”の TV アドレスであることを示し、この時、アドレス TV-0 をはじめ、6 個連続のワードレジスタが保存する情報は以下のとおりです： <p style="margin-left: 20px;">TV-0 → 秒(制限範囲 : 0~59) TV-1 → 分(制限範囲 : 0~59) TV-2 → 時(制限範囲 : 0~23) TV-3 → 日(制限範囲 : 1~31) TV-4 → 月(制限範囲 : 1~12) TV-5 → 年(制限範囲 : 1970~2037)</p> 



**プリンター
(eMT、iE、XE、
mTV シリーズ)**

モデル

現在サポートしているプリンタータイプを表示します。その中、HP PCL シリーズは USB インターフェースを使用し接続する必要があります、他のタイプのプリンターは COM ポートで接続します。

☞ 詳細は《23HMI がサポートするプリンタータイプ》をご参照。

COM ポートで接続するプリンターは正確に通信パラメータを設定する必要があります。プリンターのモデルが[SP-M, D, E, F]の場合、[毎行のピクセル数]を設定する必要があります。この設定値はプリンターが毎行に印刷できるピクセル数を超えてはいけません。さもなければ、印刷が失敗になります。

**プリンター
(cMT シリーズ)**

HMI の上にプリンターのデバイスドライバをインストールすることができます。

モデル

プリンターのモデルを選択します。

状態アドレス

プリンターの運転状態を表示します。

LW-	状態
0	無し
1	プリンターのデバイスドライバーをインスール中
2	プリンターが準備完了になった
3	印刷中
4	印刷ジョブが無効された
LW-n+1	エラー
0	無し
1	プリンターが見かれないと
2	未知のエラー

コントロールアドレス

システムパラメータ設定

プリンターの接続パラメータを設定します。

LW-n	コマンド
0	無し
1	プリンターの接続パラメータを更新する
2	プリンタードライバを再起動する
LW-n+1	インターフェースタイプ
0	イーサネット
1	USB
LW-n	IP アドレス (合計 4 ワード)
LW-+6	接続ポート (デフォルトは 9100)

スクロールバー

スクロールバーのサイズを設定できます。オブジェクトのサイズが不足の場合、スクロールバーが当オブジェクトの上に表示されます。本機能は[アラーム表示]、[イベント表示]、[履歴データ表示]、[オプションリスト]などのオブジェクトに利用されます。

パススルー(仮想 COM ポート)

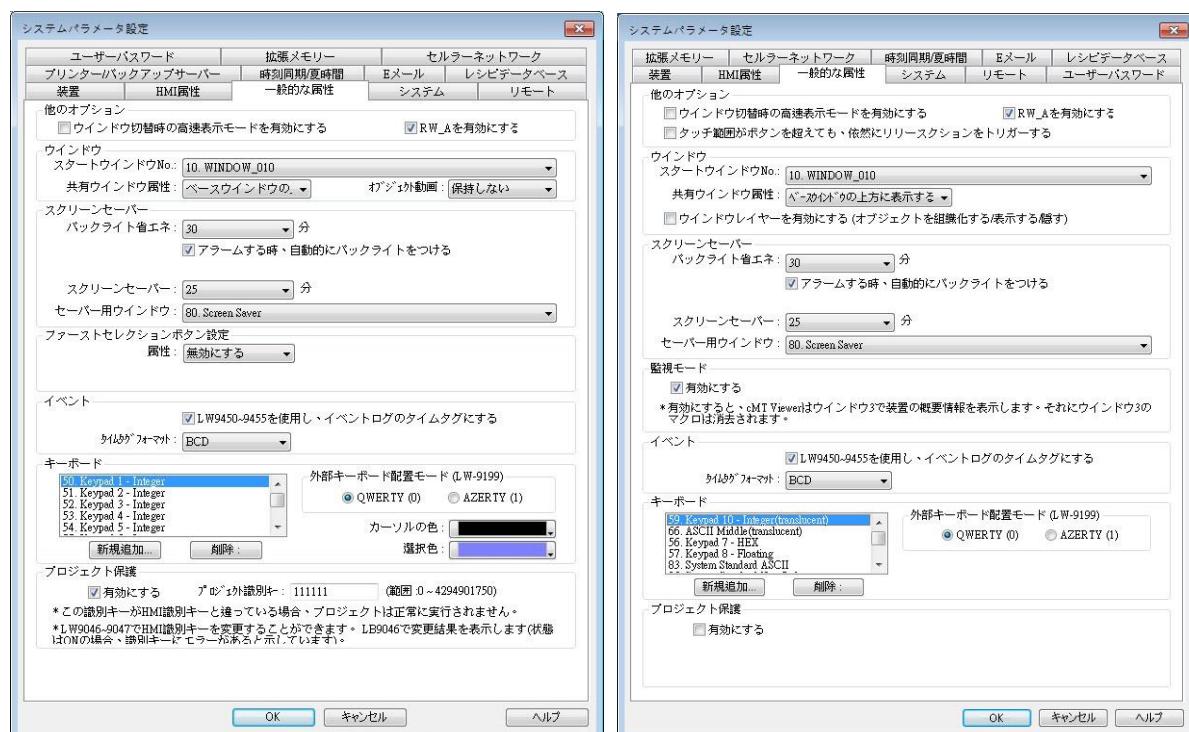
パススルーする時に、使用するポート番号を自分で設定することができます。

5.4. 一般的な属性

[一般的な属性]ページで表示画面と関連する各属性を設定します。

eMT、iE、XE、mTV シリーズ

cMT シリーズ



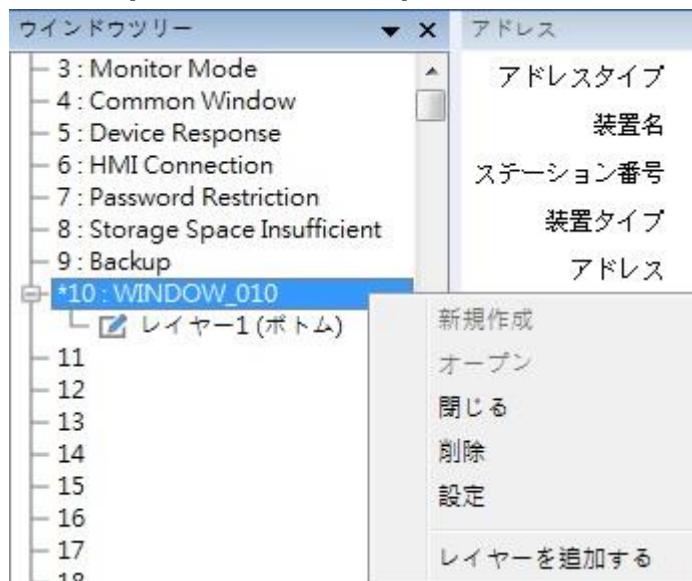
設定	記述
オプション	<p>ウインドウ変更時の高速表示モードを有効にする 本機能を選択すると、ページを切り替えた後、オブジェクトは前回正確に読み取ったデータを使用し、予めオブジェクトの内容を表示させ、より速い切替速度が求められます。</p> <p>RW_A を有効にする レシピデータ RW_A を有効にするかを選択します。RW_A を有効にしてから、オブジェクトが RW_A の内容を操作できるようになります。RW_A のサイズは 64K です。</p> <p>タッチ範囲がボタンを外れても、依然にアップアクションをトリガーする 本項にチェックマークを入れると、ボタンを押した後、たとえ指がボタンから外れた場所に移動してから離れても、指がボタンを離したと視され、アップアクションがトリガーされます。</p> <p>入力フォーカスを外す際に数値を設定する 本項にチェックマークを入れると、入力用オブジェクト(例えば数値オブジェクト)で数値を入力完了後、Enter キーを押す必要がなくそのまま入力されます。例：入力欄にフォーカスした後、また他の入力オブジェクトをクリックした場合、システムは入力が完成したと視し、数値を設定します。入力した内容が不正だった場合、ウォーニングメッセージがポップアップされます。</p>
ウインドウ	<p>スタートアップウインドウ No. HMI 起動後のスタートページを選択します。</p> <p>共有ウインドウ属性 共有ウインドウ(4 番ウインドウ)のオブジェクトが全部のベースウインドウに表示されます。本項は共有ウインドウ内のオブジェクトがベースウインドウの上方もしくは下方に表示するのを設定することができます。</p> <p>オブジェクト動画 [保持する]を選択すると、操作する時、[アニメーション]と[移動/回転図形]オブジェクトが他のオブジェクトの上に表示され、作成順番とは関係ありません。[保持しない]を選択すると、オブジェクトの表示順番は作成順番に基づき、先に作成したものが先に表示されます。</p> <p>ウインドウレイヤーを有効にする(オブジェクトを組織化)</p>

する/表示する/隠す)

有効にすると、ウインドウツリーの下での同一のウインドウに異なるレイヤーを生成できます。

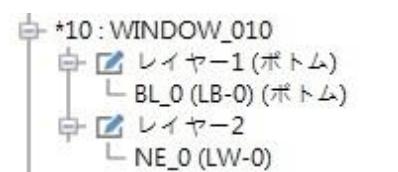
一個のウインドウでは最大 10 個のレイヤーに分けられます。

下図に示されたとおり、ウインドウ番号でマウスを右クリックし、[レイヤーを追加する]を選択します。

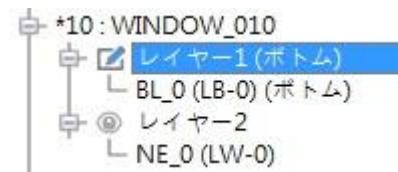


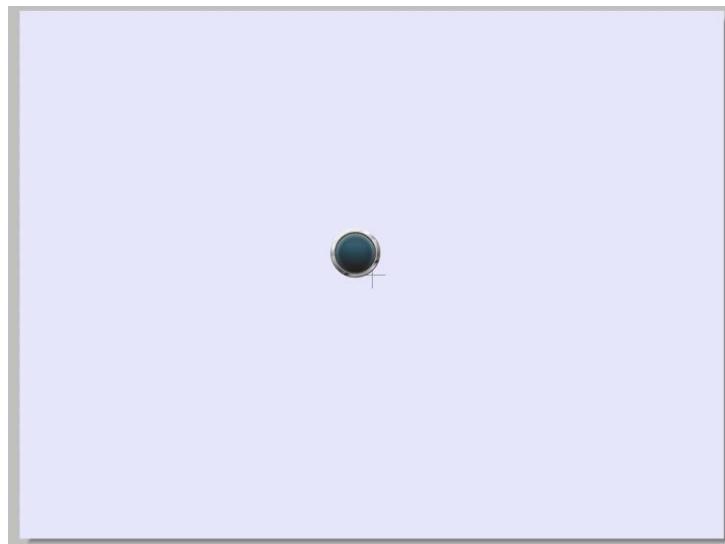
以下は使用例です:

レイヤー1 に 1 個のビットランプをオブジェクト置き、レイヤー2 に 1 個の数値オブジェクトを置きます。

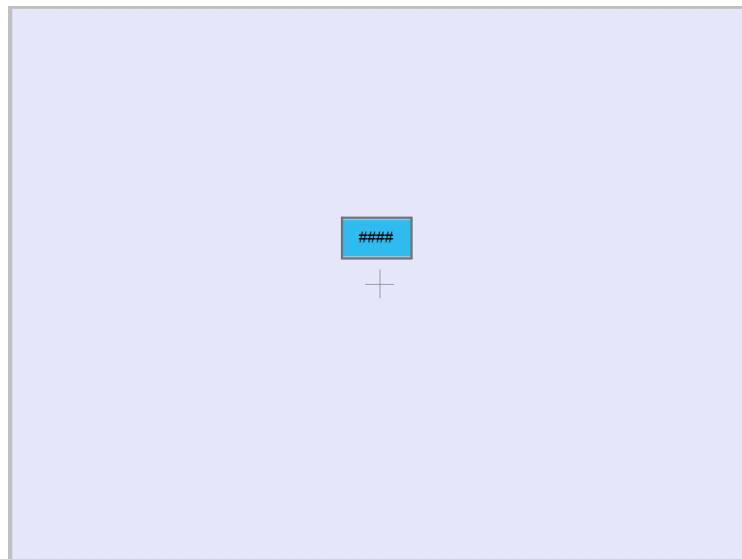
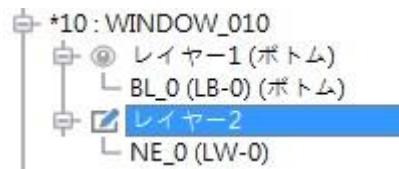


レイヤー1 を選択した場合、ビットランプオブジェクトだけが表示されます。

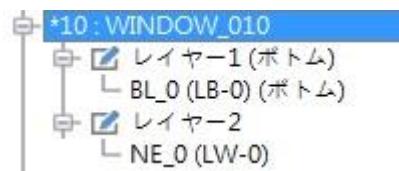


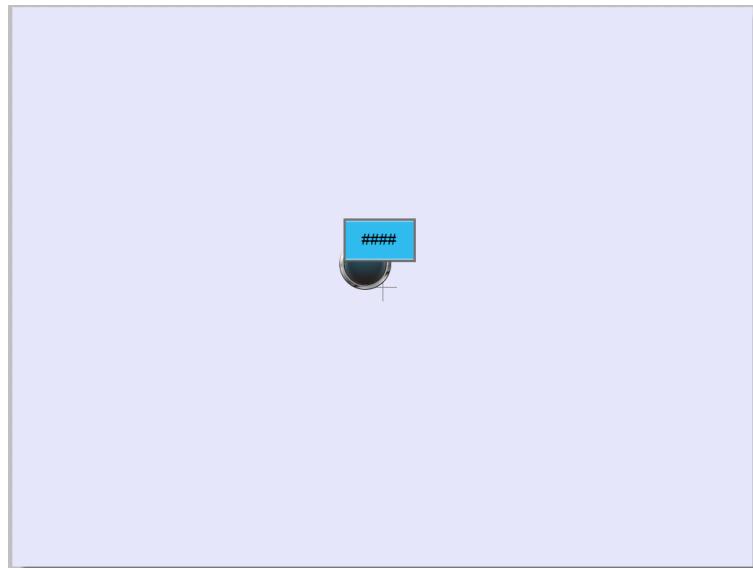


レイヤー2を選択した場合、数値オブジェクトだけが表示されます。



ウインドウ10を選択した場合、全てのオブジェクトが表示されます。





スクリーンセーバー

バックライト省エネ(時間)

スクリーンをタッチしていない時間が本設定値に相当する時、バックライトが消されます。時間単位は分です。バックライトが消された後、スクリーンにタッチしたら、バックライトがつけられます。設定値で[無]を選択する時、HMIはバックライト省エネ機能を使用しません。

スクリーンセーバー(時間)

スクリーンをタッチしていない時間が本設定値に相当する時、自動的にスクリーンセーバーモードに切り替えます。時間単位は分です。設定値で[無]を選択する時、HMIはスクリーンセーバーの機能を使用しません。

セーバー用ウインドウ

スクリーンセーバーを実行時、表示するウインドウを指定します。

ファーストセレクションウインドウ設定

eMT、iE、XE、mTV、iP シリーズのみに適します。

ファーストセレクションウインドウ(3番ウインドウ)の各属性を設定します。ファーストセレクションウインドウを使用する前、3番ウインドウを作成する必要があります。

属性

ファーストセレクションウインドウ使用するかを選択します。[使用する]を選択したら、[設定する]機能を選択し、表示する色と文字を含め、ファーストセレクションボウンドウの各属性を設定します。

位置

ファーストセレクションウインドウが現れる位置を選択します。[左]を選択すると、ウインドウは画面の左下隅に現

れます；[右]を選択すると、ウインドウは画面の右下隅に現れます。

設定

ファーストセレクションウインドウの画像及び文字のパラメータを設定します。

HMI 起動時、ボタンを隠す

チェックマークを入れると、HMI を起動した際、ファーストセレクションウインドウ設定ボタンは隠されます。ファーストセレクションウインドウを呼び出すには、LB-9013~LB-9015 で呼び出さなければなりません。

監視モード

cMT シリーズのみに適用します。

有効にすると、cMT Viewer プロジェクトのプレビュー画面はプロジェクト内のウインドウ 3(No.3 ウインドウ)になり、リアルタイムにウインドウ 3 のデータ変化を監視できます。3x3 及び 5x4 の表示形式を提供しており、最大 50 台の HMI 画面を表示できます。

イベント

LW9450~9455 を使用し、イベントログのタイムタグにする本項にチェックマークを入れると、イベントログは以下のタイムタグの時間をイベントのトリガー時間にします。

- [LW-9450: イベントログのタイムタグ-秒](制限範囲: 0 ~ 59)
- [LW-9451: イベントログのタイムタグ-分](制限範囲: 0 ~ 59)
- [LW-9452: イベントログのタイムタグ-時](制限範囲: 0 ~ 23)
- [LW-9453: イベントログのタイムタグ-日](制限範囲: 1 ~ 31)
- [LW-9454: イベントログのタイムタグ-月](制限範囲: 1 ~ 12)
- [LW-9455: イベントログのタイムタグ-年](制限範囲: 1970 ~ 2037)

注意: LW-9450 ~ LW-9455 内の数値が制限範囲を超えると、システムは本機能を使用できません。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

キーボード

キーボードウインドウのリストを表示します。これらのウインドウには[数値]と[アスキー]オブジェクトを使用する際に、選べるキーボードタイプを示します。最大 32 個のキーボードを追加することができます。ユーザーが新規キーボードを追加したい場合、既存のウインドウで使用するキ

ボードを作成し、[新規追加]機能でウインドウを選んでリストに追加すればいいです。

 詳細は《12 キーボードのデザインと使用》をご参照。

外部キーボード配置モード

QWERTY 及び AZERTY は USB キーボードのアルファベット配列方式です。HMI では LW-9199 でキーボードモードを切替できます。

カーソルの色・色を選択

eMT、iE、XE、mTV、iP のみに適用します。

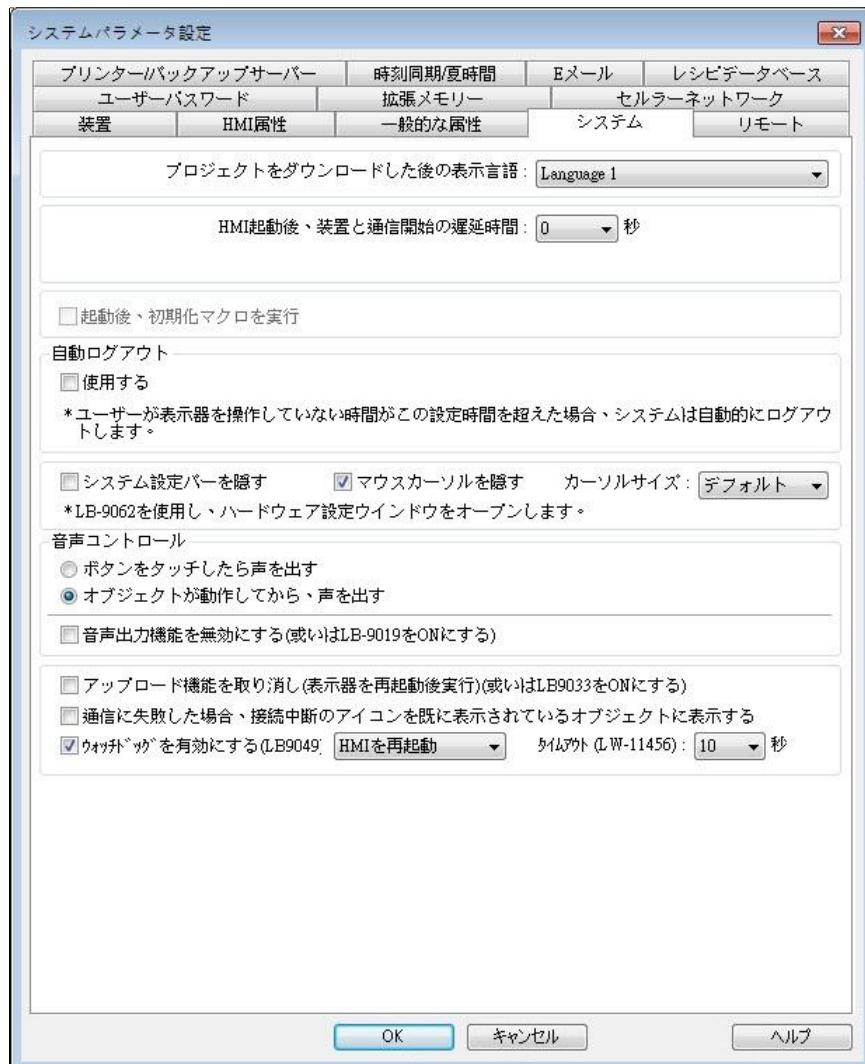
[数値]或いは[アスキー]オブジェクトを使用する時、入力欄に表示する[カーソルの色]及び[色を選択]を設定します。

プロジェクト 保護	ユーザーのプロジェクトは特定した HMI だけで使用できる ように設定できます。
--------------	---

 詳細は《30 プロジェクト保護機能》をご参照。

5.5. システム

[システム]で EasyBuilder Pro の各種の機能を設定します。



部分の機能はシステムレジスタからコピーしています。例えば：[システム設定ツールバーを隠す(LB-9020)]、[マウスカーソルを隠す(LB-9018)]、[音声出力機能を無効にする(LB-9019)]、及び[アップロード機能を取り消し(LB-9033)]などです。ユーザーはシステムタグでこれらの機能を使用することができます。

システムタグを選択するには、ユーザーがオブジェクトを選択する際に、設定タブの[システムタグ]にチェックマークを入れ、[アドレスタイプ]を選択します。

全てのシステムタグを閲覧したい場合、EasyBuilder Pro で[プロジェクト] » [タグライブラリ] » [システムタグ]の順で選択します。

設定	記述
プロジェクトをダウンロードした後の表示言語	プロジェクトをダウンロードし、HMI を起動する時に表示される言語を選択します。
HMI 起動後、装置と通信開始の遅延時	一部の PLC 装置の起動速度がより遅いので、通信当初に、HMI が PLC 装置の起動速度が遅いから通信エラー

間	を出すことを避けるため、本機能を利用して HMI 起動された後、装置と通信し始める時点を設定できます。 HMI 起動後、初期化マクロの実行を遅延することを含む 有効にすると、[HMI 起動時、一回実行する]に設定されたマクロの実行も遅延されます。
起動後、初期化マクロを実行	HMI を起動後、実行するマクロを指定します。
自動ログアウト	HMI をタッチしていない時間がこの設定値を超えたら、セキュリティがあるオブジェクトが使用できなくなり、使用するにはもう一度ユーザーID 及びパスワードを入力する必要があります。
システム設定バーを隠す	右下隅にあるシステム設定バーを隠します。
マウスカーソルを隠す	HMI 上のマウスカーソルを隠します。
カーソルサイズ	マウスカーソルのサイズを調整できます。
音声コントロール	ボタンをタッチしたら声を出す：ボタンをタッチしたら声が出されます。 オブジェクトが動作してから、声を出す：オブジェクトが動作してから声が出されます。オブジェクトに [最小押し時間] が設定されたら、オブジェクトをタッチしてから、オブジェクトが動作するまでには、時間差があるので、オブジェクトが動作してから声が出されます。 音声出力機能を無効にする：HMI の音声を閉じます。 (システム設定のボタン音が含まれません)。
アップロード機能を取り消し(HMI を再起動後実行)(或いは LB9033 を ON にする)	プロジェクトをアップロードする機能を閉じます。本項を選択しプロジェクトをダウンロードしたら、HMI を再起動してからアップロードを取り消します。
装置との通信が失敗した場合、接続中断のアイコンを関連オブジェクトに表示する	装置との通信が失敗した場合、接続中断のアイコンを関連オブジェクトに表記するかを選択します。  

断のアイコンがオブジェクトの右下隅に表示されます。

接続中断のアイコンは一度装置との接続に成功したが、後に失敗したオブジェクトでのみ表示されます。

ウォッチドッグを

有効にする

フォントの事前ロードを有効にする
(パフォーマンスを最適化するが、HMIの起動速度が遅くなるかもしれません)

HMI が指定した時間を超えても正常に作動できない場合、自動的にシステムを再起動します。

本機能は cMT HMI にのみサポートされます。本項にチェックマークを入れると、文字が多いページでの切り替え速度が向上しますが、HMI の起動速度が数秒遅延される可能性があります。

5.6. リモート

本タブで遠隔接続関連の機能を設定します。部分の機能はシステムレジスタと同じです。



設定	記述
リモート HMI との接続を禁 止	リモート HMI がローカル HMI への接続を禁止し、LB-9044 でリモート接続を制御することもできます。
遠隔でのパス ワード読み取 りを禁止(或 いは LB9053 を ON にする)	プロジェクトとユーザーpasswordなどの情報を含め、リモート HMI がローカル HMI のpasswordを読み取るのを禁止します。
遠隔でのパス ワード書き込 みを禁止(或 いは LB9054 を ON にする)	プロジェクトとユーザーpasswordなどの情報を含め、リモート HMI がローカル HMI のpasswordを書き込むのを禁止します。
VNC サーバー	パスワード無しでログイン：VNC を通じて HMI を接続する時、passwordを設定する必要がありません。

[監視モード]: 使用したら、VNC を通じて HMI に接続する場合、操作ができなく、監視のみできます。

プロジェクトからパスワードを設定する: VNC サーバーにログイン用のパスワードを設定します。

EasyAccess サーバー

EasyAccess 1.0 サーバーにログイン :

本機能を通し、その場で HMI を操作しているように、ユーザーがインターネットを通じて接続中の HMI を監視し、直接に PC で操作できます。

さらに特別なのは、EasyAccess は更新した画像を転送する必要がなく、即時の資料だけを転送します。それにより、転送の速度や効率が向上できます。詳細は《EasyAccess》をご参照ください。

EasyAccess 2.0 サーバー位置

グローバル、または中国が選べます。

 このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

診断器

有効にする : 有効にすると、cMT Viewer を通じて cMT に接続する場合、診断器をオープンできます。LB-12656 も診断器を制御することに用いられます。

パスワードが不要 : 有効にすると、診断器にログインしたい時、パスワードを入力する必要がありません。LB-12657 もパスワードを入力するかを制御することに用いられます。

パスワード : 診断器にログイン用のパスワードを設定します。LW-11756 (4 words) もパスワードの設定に用いられます。

cMT Viewer

最大接続数

cMT HMI が cMT Viewer に接続される最大の接続数を設定します。

5.7. ユーザーパスワード

[ユーザーパスワード]設定タブでユーザーのパスワードと関連オブジェクトを設定します。認証モードは 2 種類あります：
一般モードと高度セキュリティモード。

 詳細情報は、《10 オブジェクトセキュリティ》をご参照ください。

5.7.1. 一般モード



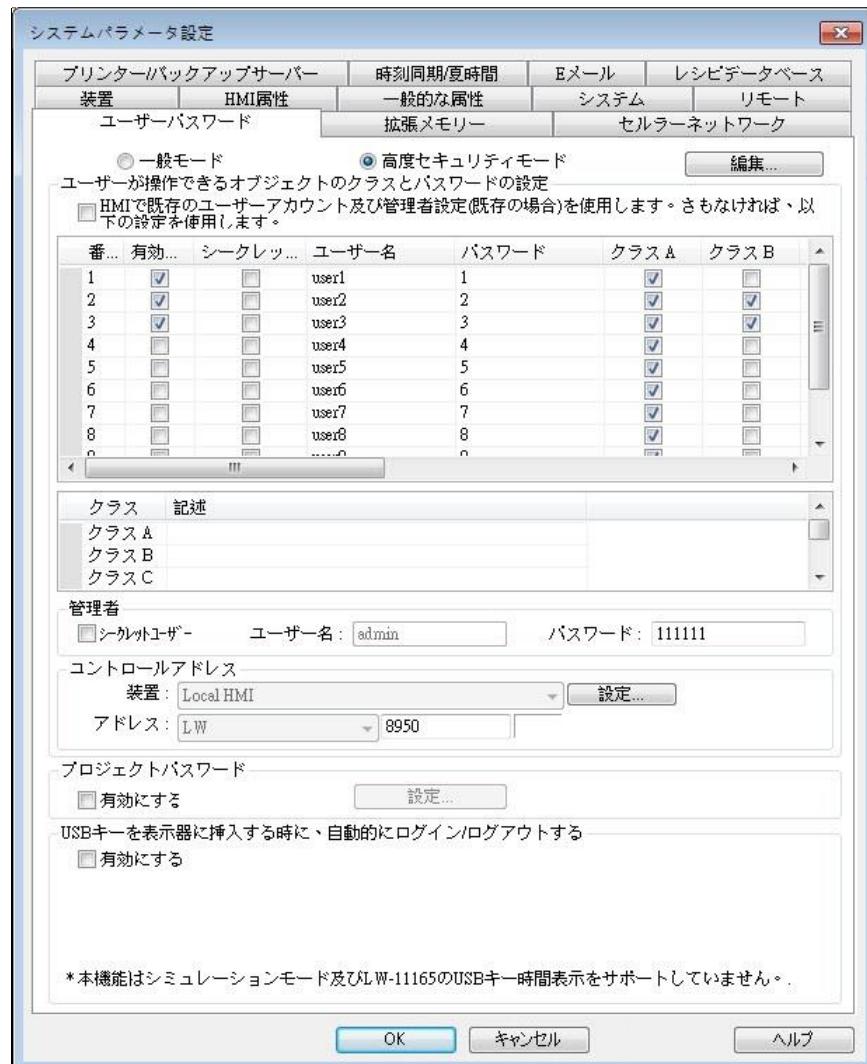
12 セットのユーザーを設定することができます。パスワードは非負整数であること。ユーザーがパスワードを入力した後、システムはユーザーの設定内容に基づき、操作できるオブジェクトのクラスを決めます。オブジェクトのクラスは[クラス A]から[クラス F]と合計 6 クラスあります。

クラスが[無]に属した場合、全てのユーザーが使用できます。

“ユーザー3”の設定内容が上記の場合、このユーザーは“無”と A、B、C クラスのオブジェクトのみ使用できます。

5.7.2. 高度セキュリティモード

高度セキュリティモードで設置できるユーザー数は 11 セットで、また[管理者]モードを提供しています。管理者は最大な権限を持ち、どのクラスのオブジェクトも操作できます。ユーザー パスワードはアルファベット或いは数字で組み合わせられ、また各ユーザーごとが操作できるオブジェクトを[クラス A]～[クラス L]と 12 クラスに分けられます。

**設定****記述****編集**

このページに対する変更を禁止するためのパスワードを設定します。



[読み取り専用を有効にする]：有効にすると、ユーザーはこのページの設定を変更することができません。

[パスワードをマスクする]：有効にすると、パスワードは*印で表示されます。

ユーザーが操作できるオブ

[HMI で既存のユーザーアカウントを使用する]を選択した場合、プロジェクトを HMI にダウンロードすると、HMI での

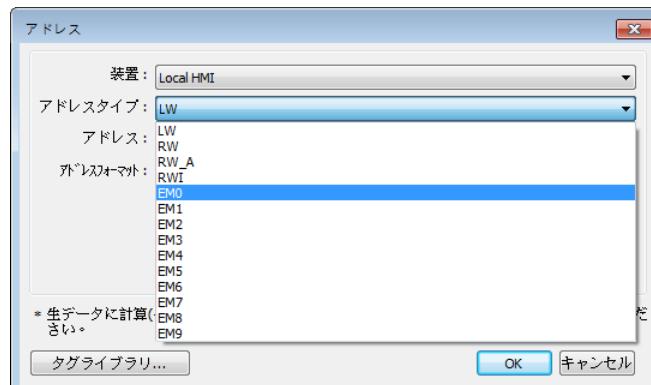
オブジェクトのクリア	既存のユーザー アカウントは消去されません。
パスワードの設定	
管理者	デフォルト管理者アカウントであり、削除不可です。全ての権限を持ち、変更することができません。 高度セキュリティモードは[オプションリスト]に合わせ、アカウント名と権限を表示することができます。 [アカウントを隠す]を選択すると、アカウント名と権限などの資料は[オプションリスト]オブジェクトに表示されません。
コントロールアドレス	高度セキュリティモードはユーザーのログインとアカウント管理のため、[コントロールアドレス]を提供します。
プロジェクトパスワード	ユーザーは[プロジェクトパスワード]を設定することができます。このパスワードが設置された場合、ユーザーがプロジェクトを編集したかったら、パスワードを入力する必要があります。 [有効にする]にチェックマークを入れて[設定]を押すと、パスワードを設定することができます。 設定が完了後、プロジェクトを編集する前に、パスワードを入力するというリクエストが出ます。入力したパスワードが正確だった場合だけ、プロジェクトに入ることができます。
USBキーをHMIに挿入する時に、自動的にログイン/ログアウトする	本項にチェックマークを入れたら、USBキーで自動的にログイン/ログアウトでき、それにログイン/ログアウトの状態は自動的に指定した[状態アドレス]に書き込みます。USBディスクを挿入すると、自動的にログインします。USBディスクを外すと、自動的にログアウトします。状態コードの意味は 0x00:無動作 0x01:ログイン成功 0x04:ログイン失敗 0x08:ログアウト成功 0x10:ログアウト失敗となります。

5.8. 拡張メモリー

[拡張メモリー]設定タブでメモリーを拡張する保存位置を設定します。



拡張メモリーは EM0~EM9 があり、使用方式は他の装置と似ています(LW や RW アドレスと類似している)。ユーザーはオブジェクト作成時に[装置タイプ]で EM0~EM9 を指定すれば使用できます。一個の拡張メモリーは最大 2G word の資料を保存することができます。



拡張メモリー内のデータはファイル形式で[SD カード]、[USB ディスク]に保存されます。

[EM0]~[EM9]が使用するデフォルトファイル名は em0.emi~em9.emi で、ユーザーは自分でファイルをリネームでき、そして RecipeEditor.exe でこれらのファイルをオープンし編集できます。

拡張メモリー内のデータは HMI の電源中断で消えることがありません。即ち、次回起動後、拡張メモリー内のデータは電源が切れた前の状態に復元します。レシピデータ(RW、RW_A)に類似しています。

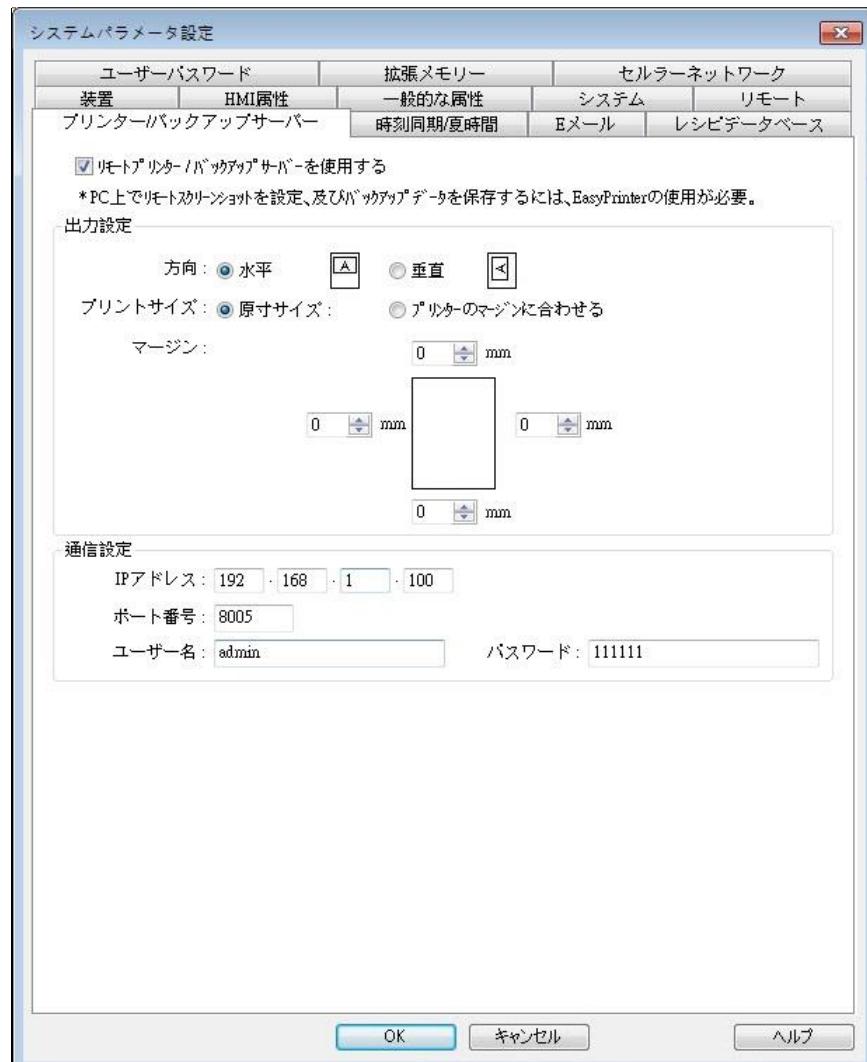
拡張メモリーとする装置が存在しない場合、拡張メモリー内のデータを読み取ると、内容は全部“0”と表示されます。拡張メモリーとする装置が存在しない場合、データを拡張メモリーに書き込むと、HMI には“Device No Response”を表示します。

HMI の電源が中断していない状況下、隨時に外部装置を挿入、または外すことができます。ユーザーはこの特性を用いて、拡張メモリー内のデータを更新または取得することができます。

 このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

5.9. プリンター/バックアップサーバー

この設定タブにリモートプリント/バックアップサーバーの関連設定を設定します。



設定	記述
出力設定	<p>方向 図面/文字のプリントアウト方向を[水平]または[垂直]を選択します。</p> <p>プリントサイズ [原寸サイズ]または[プリンターのマージンに合わせる]を選択します。</p> <p>マージン 上下左右のマージンの幅を設定します。</p>
通信設定	<p>IP アドレス ネットワークを通してサーバーのIPアドレスを設定します。</p> <p>ポート番号、ユーザー名、パスワードを設定 プリンターのログイン情報を指定します。</p> <p>ポート番号は1~65535に設定することができます。</p> <p>ユーザー名またはパスワードは最長12ワードを設定するこ</p>

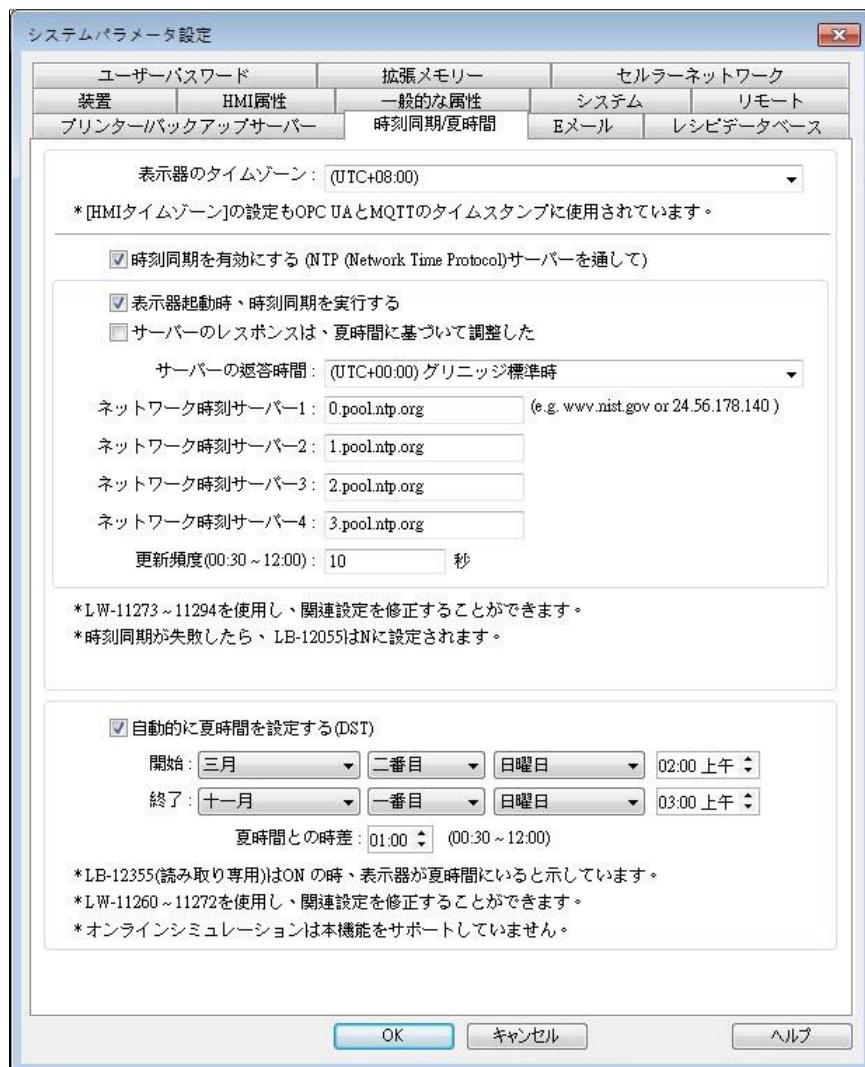
とができます。

 詳細は《26 EasyPrinter》をご参照ください。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

5.10. 時刻同期/夏時間

この設定タブで HMI の時間を NTP サーバーと一致にします。



設定 記述

表示器タイムゾーン	HMI のタイムゾーンを設定します。
-----------	--------------------

時刻同期を実行する	表示器起動時、時刻同期を実行する
-----------	------------------

行する	<p>HMI を起動する際に、まずは指定した NTP サーバーに自動同期させます。</p> <p>HMI のタイムゾーン</p> <p>HMI のタイムゾーンを設定します。</p> <p>サーバーの返答時間</p> <p>NTP サーバーのタイムを設定します。</p> <p>ネットワーク時刻サーバー</p> <p>四個のネットワークタイムサーバーフィールドを提供します。一番目のサーバーと時刻同期に失敗した場合、二番目のサーバーと時刻を取ってみます。これに沿って順番で同期を試します。HMI が全てのサーバーにも同期が失敗してしまった時、システムレジスタ LB-12055 は ON に設定されます。</p> <p>更新頻度</p> <p>NTP サーバーと時刻同期を取る頻度を設定します。頻度範囲は 10~86400 秒です。</p>
自動的に夏時間(DST)を設定する	<p>開始/終了</p> <p>夏時間の開始と終了時間を設定します。週の欄の[最後の]とは当月の最終の週で、当月の五番か六番目の週に当たる可能性があります。</p> <p>夏時間との時差</p> <p>夏時間との時差を設定します。</p>

Note

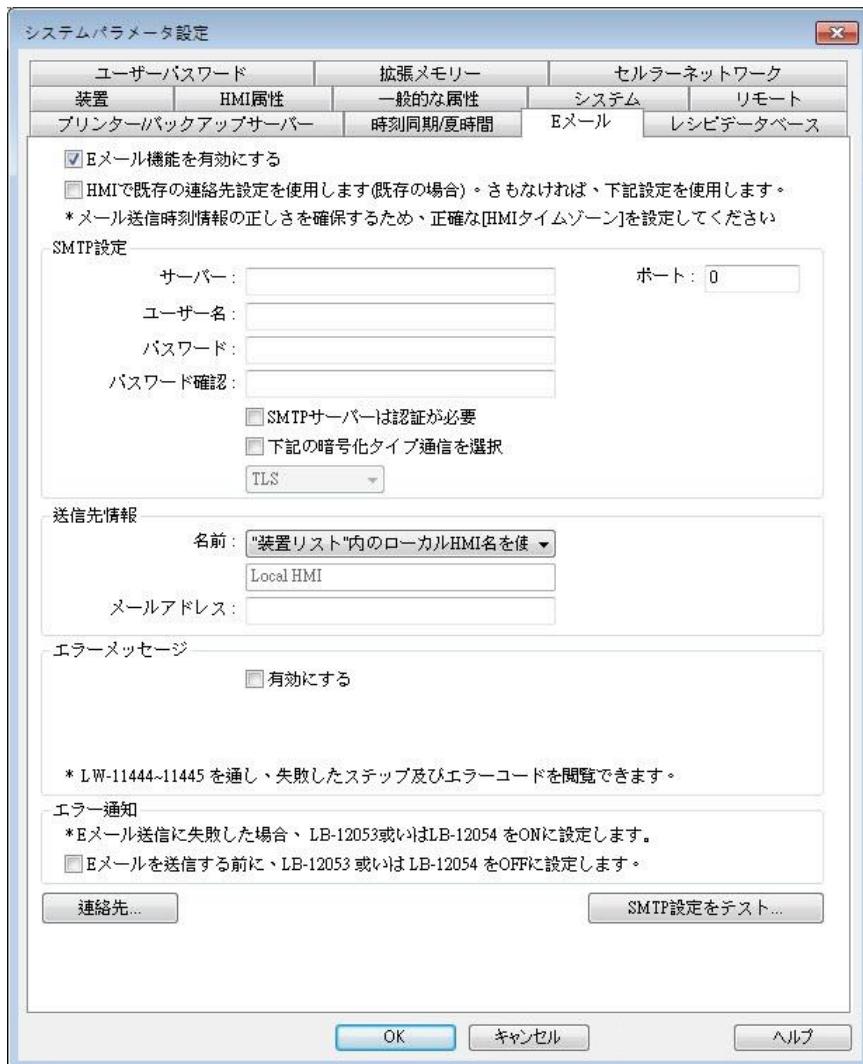
- 夏時間が終了すると、時刻は前へ調整され、暫く時間が重ねる時期になります。この時、トレンド図オブジェクトのリアルタイムモードは画面の更新を停止するが、履歴データは影響されません。
- 夏時間が終了すると、HMI の時刻が調整されても(手動修正またはネットワークで校正するなど)、システムは夏時間を起動しません。LB-12355 は OFF に維持します。
- 夏時間が終了すると、HMI の時刻は夏時間に戻るように見えるが、システムは夏時間を起動しません。LB-12355 は OFF に維持します。
- まだ夏時間に入っていない時、先に HMI の時刻を夏時間に調整したら、夏時間に入ると視し、LB-12355 は ON にします。この時、HMI は調整された時刻を基準とし、[夏時間との時差]を加えません。
- 夏時間になった後、HMI 時刻を夏時間から外す場合、夏時間が終了すると視し、LB-12355 は OFF にします。この時、HMI は調整された時刻を基準とし、[夏時間との時差]を加えません。
- 本機能は南半球の夏時間もサポートしていません。

 夏時間の関連レジスタを《22 アドレスレジスタ》をご参照ください。

5.11. E メール

[E メール]設定タブで E メールの関連データを設定します。

[HMI で既存の連絡先を使用する]にチェックマークを入れたら、プロジェクトを HMI にダウンロードした時、優先に HMI での既存の連絡先の設定を使用します。存在しない場合、以下の設定を使用します。



設定	記述
SMTP 設定	<p>サーバー : E メールサーバーを設定します。</p> <p>ポート : ポートを設定します。</p> <p>ユーザー名 : メールアドレスを設定します。</p> <p>パスワード : E メールのパスワードを設定します。</p> <p>パスワードを確認 : 設定されたパスワードを確認します。</p> <p>SMTP サーバーに認証が必要 : E メールにログインする際、セキュリティパスワード認証が必要です。</p>

下記の暗号化タイプ通信を選択：メールを発送する際、暗号化通信が必要かを選択します。(TLS、SSL)

送信先情報	名前 HMI 名を使用、または送信者名前を入力します。 メールアドレス メールアドレスを設定します。
エラーメッセージ	E メール送信に異常が出た時、SMTP サーバーが送信したエラーメッセージを指定したレジスタアドレスに表示させます。
	 E メール送信の関連レジスタに関して、《22 システムレジスタ》をご参照ください。
エラー通知	有効にすると、E メールを送信する前、まずは LB-12053 または LB-12054 を Off に設定するようになります。さもないと、E メールの送信が失敗になる場合、LB-12053 または LB-12054 は常時 ON に維持されます。
SMTP 設定をテスト	[テスト]を選択した後、EasyBuilder Pro は自動的に一通のメールを指定した宛先グループに送信し、SMTP サーバーの設定が正しいかを検証します。

[連絡先...]ボタンを押すと：



設定	記述
連絡先リスト	連絡先リストを追加/削除します。
グループ情報	複数の連絡先をグループに設定します。
グループ数量	グループ数量を設定します。グループ名は数量によって順番で A~P と命名します。最大 16 グループまでに設定できます。

カレントグループ

現在、リスト内にいる連絡先が所属しているグループを表示します。

記述

ユーザーは当グループに関する記述を入力することができます。



イベントログの送信については《7 イベントログ》をご参照ください。



Eメール送信の関連レジスタ情報については《22.3.40 システムレジスタ》をご参照ください。



このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

5.12. セルラーネットワーク

[セルラーネットワーク]設定タブでセルラーネットワークの関連内容を設定します。

対応機種：

cMT-SVR(OS バージョン：20151127 以降)

cMT 3072/ cMT 3090/ cMT 3103/ cMT 3151(OS バージョン：20180723 以降)

cMT-HDM/cMT-FHD (OS バージョン：20190130 以降)

3G/4G ドングルを USB ポートに挿入すれば、インターネットに接続できます。

**設定****記述****HMIで現行設定
を使用する**

本項を選択すると、セルラーネットワークは HMI の既存パラメータを使用します。

備考：特別なネットワークカード以外、一般的なセルラーネットワークの[PIN コード] は 0000 で、[APN]は internet です。[ユーザー名]、[パスワード]、[電話番号]を設定する必要がありません。

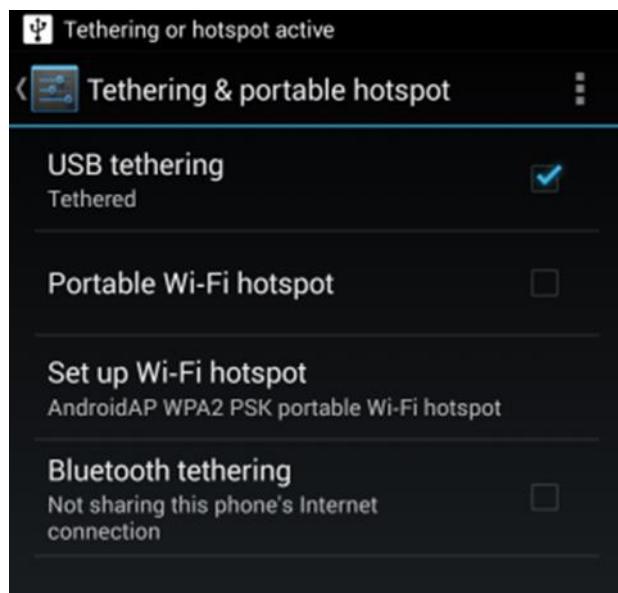
関連するシステムレジスタ：

- LW-11297: SIM カードの PIN コード(セルラーネットワーク)
- LW-11313: アクセスポイントネーム APN (セルラーネットワーク)
- LW-11329: ユーザー名(セルラーネットワーク)
- LW-11345: パスワード(セルラーネットワーク)
- LW-11361: 電話番号(セルラーネットワーク)
- LW-11377: 接続を停止(0に設定)/起動する(1に設定)(セルラーネットワーク)
- LW-11378: 最終エラーコード(0:成功, 1:間違った PIN コード, 2: SIM

カードがない, 3:装置がない, 4:puk がロックされた, 5:その他)(セルラーネットワーク)
接続状態(0:設備がない, 1:接続が中断された, 2:接続中, 3:
接続された)(セルラーネットワーク)

USB テザリング

Android 携帯電話の USB テザリング機能を通じ、HMI はインターネットに接続することが可能になります。USB テザリング機能を使用したい場合、micro USB ケーブル、或いは任意の携帯電話用 USB データ転送ケーブルさえあれば、Android 携帯電話の USB テザリング機能で HMI と Android を接続することができます。例は下記をご参照ください。



そして、下記のシステムレジスタで状態を監視します：

LW-11380: 接続を停止(0 に設定)/起動する(1 に設定)(USB テザリング)

LW-11381: 接続状態(0:装置がない, 1:接続が切断された, 2:接続された, 3:失敗, 4:OS がサポートしない, 5:HMI がサポートしない)(USB テザリング)

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

6. ウィンドウ

本章では、ウィンドウの種類、及び作成と使用の仕方について説明します。

6.1.	概要	6-2
6.2.	ウィンドウの種類	6-2
6.3.	ウィンドウの作成、設定と削除	6-5
6.4.	ウィンドウの透明度	6-8
6.5.	編集ティップス	6-11

6.1. 概要

ウインドウは EasyBuilder Pro 編集ソフトの中に最も大切な組成の一つであります。各種の HMI に表示するオブジェクト、画像、文字などを表示するのは、ウインドウを通さなければなりません。EasyBuilder Pro はウインドウ 1997 個を内蔵していて、その番号はウインドウ 3～ウインドウ 1999 になります。

6.2. ウインドウの種類

機能と使用の仕方の違いで、下記四種類に分けられます：

- ベースウインドウ
- ファーストセレクションウインドウ
- 共有ウインドウ
- システムメッセージウインドウ

6.2.1. ベースウインドウ

ベースウインドウは一番よく使われるウインドウタイプです。メイン画面に使用する以外、下記用途にも使われます：

- ベース画面として、他のウインドウの背景画面に用いられます。
- キーボードウインドウ。
- ファンクションキーオブジェクトが使用するポップアップウインドウ。
- 間接ウインドウと直接ウインドウオブジェクトが使用するポップアップウインドウ。
- スクリーンセーバーウインドウ画面。

Note

- ベースウインドウのサイズは HMI のスクリーンサイズと同じにする必要があるので、解像度の設定も HMI と一致しなければなりません。

6.2.2. ファーストセレクションウインドウ

3番ウインドウはデフォルトのファーストセレクションウインドウで、このウインドウはベースウインドウと同時に存在することができ、一般には常に使われる仕事ボタンを置く用に使用されています。位置は画面の左下隅或いは右下隅にあります。ファーストセレクションウインドウを使用するには、先に 3番ウインドウを作成する以外、ファーストセレクションウインドウボタンの各属性を設定する必要があります。ファーストセレクションウインドウボタンの各設定は[システムパラメータ設定] » [一般的な属性]タブにあります。ファーストセレクションウインドウボタンで表示/隠すの切り替えができる以外、システムレジスタもファーストセレクションをコントロール用で下記アドレスを提供しています：

[LB-9013] ファーストセレクションウインドウを隠す(ON に設定する)/表示する(OFF に設定する)

ウインドウ

[LB-9014] ファーストセレクションボタンを隠す(ON に設定する)/表示する(OFF に設定する)

[LB-9015] ファーストセレクションウインドウ/ボタンを隠す(ON に設定する)/表示する(OFF に設定する)

Note

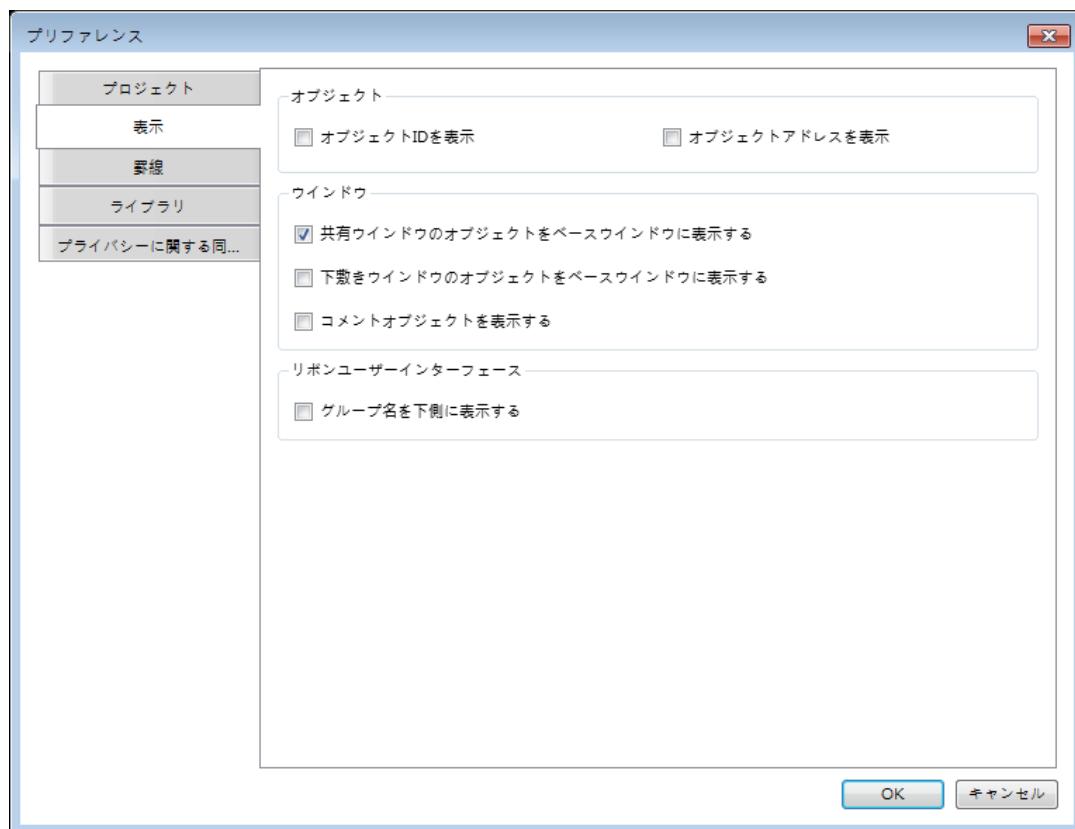
- cMT シリーズはファーストセレクションウインドウ機能をサポートしていません。

6.2.3. 共有ウインドウ

4 番ウインドウはデフォルトの共有ウインドウで、このウインドウ中のオブジェクトも他のベースウインドウに現れるが、ポップアップウインドウだけに現れません。よりまして、通常では各ウインドウが共通しているオブジェクトを共有ウインドウに置きます。

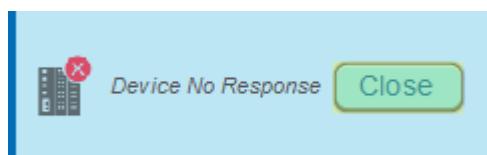
HMI を操作する時、ファンクションキーの[共有ウインドウを切り替える]モードで、オンラインで共有ウインドウのソースを変更することができます。

» [プリファレンス]ではプロジェクトを編集する際に、共有ウインドウ内のオブジェクトがベースウインドウに表示するかを決められます。それにより、プロジェクトを編集している時に、ベースウインドウのオブジェクトが共有ウインドウのオブジェクトに重ねることが避けられます。



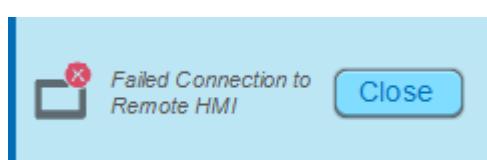
6.2.4. システムメッセージウインドウ

5、6、7、8番ウインドウはデフォルトのシステムメッセージウインドウです：



5番ウインドウは Device Response ウィンドウ

HMI と装置との通信が中断した時、システムは自動的に Device No Response のメッセージウインドウをポップアップします。システムレジスタでポップアップするのを禁止することができます。



6番ウインドウは HMI Connection ウィンドウ

ローカル HMI がリモート HMI に接続できない場合、システムは自動的に Failed connection to remote HMI のメッセージウインドウをポップアップします。



7番ウインドウは Password Restriction ウィンドウ

ユーザーがあるオブジェクトを操作する権限がない場合、Password Protected! Access Denied! のメッセージウインドウをポップアップするのかを決められます。



8番ウインドウは Storage Space Insufficient ウィンドウ

HMI 内蔵メモリー、USB ディスク或いは SD カード内の空き領域が不足になった場合、システムが自動的に Storage Space Insufficient のメッセージウインドウをポップアップします。(メモリーが 4MB 以下と検知された時)

下記のシステムレジスタは HMI、USB ディスク或いは SD カードで現在での空き領域を表示します：

[LW-9072]現在、HMI 内の空き領域(Kbytes)

[LW-9074]現在、SD カード内の空き領域(Kbytes)

[LW-9076]現在、USB ディスク内の空き領域(Kbytes)

また、下記のシステムレジスタを通じて記憶装置の空き領域が足りるのかを確認することができます。メモリーが 4MB 以下になったと検知された場合、ON に設定されます：

[LB-9035]HMI 内の空き領域が不足アラーム(状態は ON の場合)

[LB-9036]SD カード内の空き領域が不足アラーム(状態は ON の場合)

[LB-9037]USB ディスク内の空き領域が不足アラーム(状態は ON の場合)

詳細は《22 アドレスレジスタ》をご参照ください。

ウインドウ 5~ウインドウ 8 の内容提示は、実際の需要で変更することができ、操作人員が故障情報をより理解、判別しやすくなるようにします。

Note

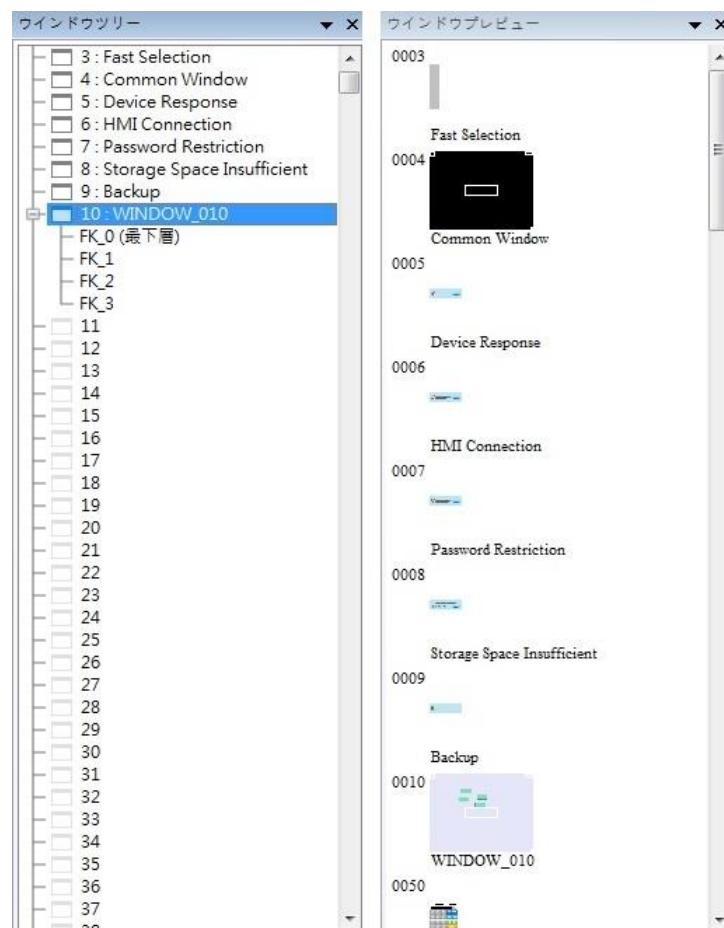
- eMT、iE、XE、mTV シリーズはシステムメッセージウインドウ、直接ウインドウや間接ウインドウを含め、同時に最大24個ポップアップウインドウを表示することができます。cMT シリーズはポップアップウインドウの数量制限がありません。
- システムは一個のベースウインドウの上で、2 個の直接(または間接)ウインドウを使用し同一のウインドウをポップアップすることを禁じます。
- ウインドウ 3~ウインドウ 9 はシステムに使用されるもので、ウインドウ 10~1999 はユーザーに使用、編集されるものです。

6.3. ウインドウの作成、設定と削除

[ウインドウツリー]で作成したウインドウを確認します。

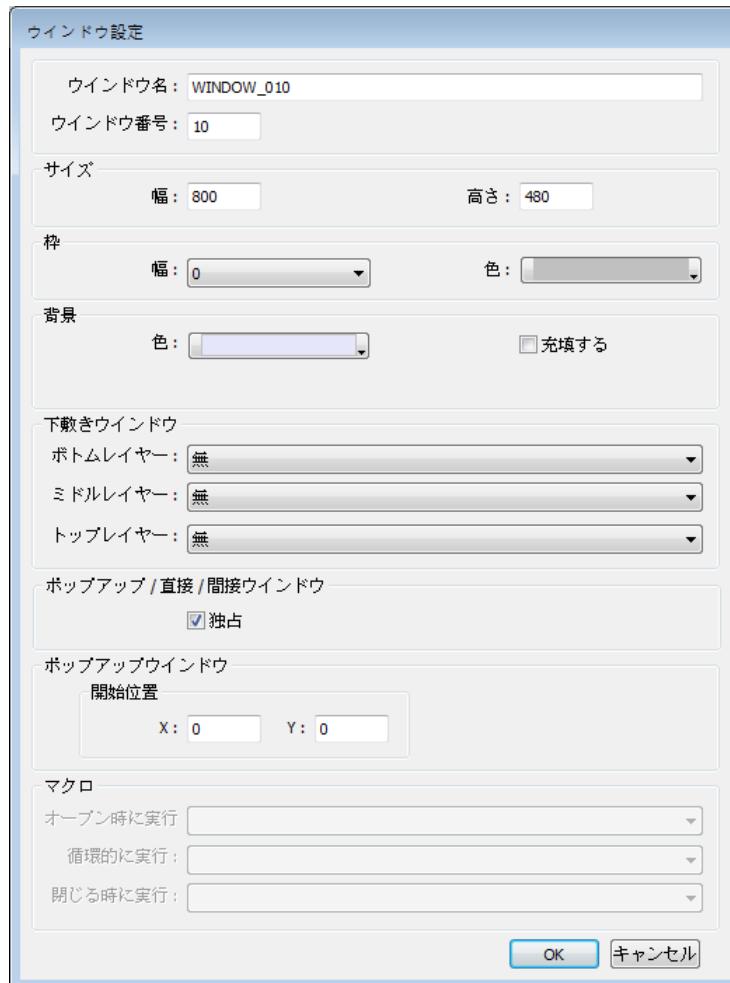
[ウインドウ]ではウインドウの番号が表示され、定義されたウインドウ名は表示されます。現在編集中のウインドウの番号の前には□アイコンがあり、番号のそばにある(+)印を押すと、ウインドウに使用されたオブジェクトが見られ、オブジェクト ID、アドレスと記述などが含まれています。

[ウインドウプレビュー]ではウインドウ外見のサムネイルをプレビューすることができます。



6.3.1. ウインドウの作成と設定

ウインドウツリーで、作成したいウインドウ番号を選択してから、マウスを右クリックし、[新規作成]を選択します。



設定	記述
ウインドウ名	入力した名前はウインドウのタイトルバーに表示され、ウインドウツリーにも表示されます。
ウインドウ番号	3~1999 があります。
サイズ	ウインドウのサイズを設定します。一般の場合、ウインドウの解像度を HMI の解像度と一致にします。
下敷きウインドウ	下敷きウインドウ機能はもう一つの共有ウインドウに視してもいいです。プロジェクトを設計する段階で、同一のオブジェクトが全てのウインドウではなかったが、複数のウインドウに用いられる可能性があり、その場合、こういう下敷きウインドウが適します。 一個のベースウインドウは最大三個の下敷きウインドウを背

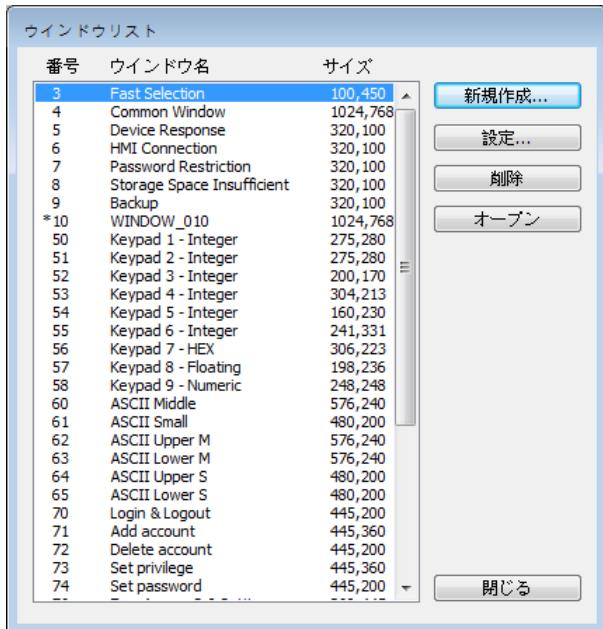
景にすることができます。[トップレイヤー]から[ボトムレイヤー]まで、下敷きウインドウ内のオブジェクトが順番に現されます。

ポップアップ ウインド ウ	ベースウインドウもポップアップウインドウにすることができます。[X]と[Y]を使用し、ベースウインドウが画面にポップアップされる座標位置を設定します。座標の原点は画面の左上隅にあります。
独占	本項を選択すると、当該ウインドウの操作を最優先にします。例えば、ある独占ウインドウがポップアップされた場合、他のポップアップウインドウと背景ウインドウの操作は完全に中止され、独占ウインドウを閉めてから、他のウインドウを操作することができるようになります。ベースウインドウがキーボードウインドウとされる場合、自動的に独占ウインドウになります。
タイトルバー	システムメッセージウインドウがポップアップされた時に、タイトルバーを含むかを選択します。 5~8番ウインドウのみ本機能を設定することができます。

Note

- 下敷きウインドウにあるオブジェクトは、それらオブジェクトを表示するベースウインドウの中で編集することができません。下敷きウインドウの中のオブジェクトを編集する場合、当該ウインドウ内で編集しなければなりません。
 - ベースウインドウが使用する下敷きウインドウの番号が、ポップアップしたいウインドウの番号と同じの場合、当該ウインドウをポップアップアップすることができなくなります。
 - ベースウインドウとポップアップウインドウが同一の下敷きウインドウを使用する場合、ポップアップされたウインドウは下敷きウインドウ上のオブジェクトを表示することができません。
-  このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。
- 或いは[閲覧] » [ウインドを開く]で[新規作成]を選択し、作成したいウインドウのタイプを選択します。

ウインドウ



[ウインドウ設定]ダイアログボックスを呼び出すには下記方法があります：

- ウィンドウツリーにウインドウ番号を選択し、マウスを右クリックし[設定]を選択します。
- [閲覧] » [ウインドを開く]で設定したいウインドウを選択した後、[設定]を選択します。
- 当ウインドウの中に、オブジェクトを選択していない状態でマウスを右クリックし、[属性]を選択します。

6.3.2. ウィンドウをオープンする、閉じる、削除する

既存のウインドウをオープンするには下記方法があります：

- ウィンドウツリー内のウインドウ番号をダブルクリックします。
- ウィンドウツリーにオープンしたいウインドウ番号を選択した後、マウスを右クリックし[オープン]を選択します。
- [閲覧] » [ウインドを開く]でオープンしたいウインドウ番号を選択した後、[オープン]を選択します。

既存のウインドウを閉じる/削除するには下記方法があります：

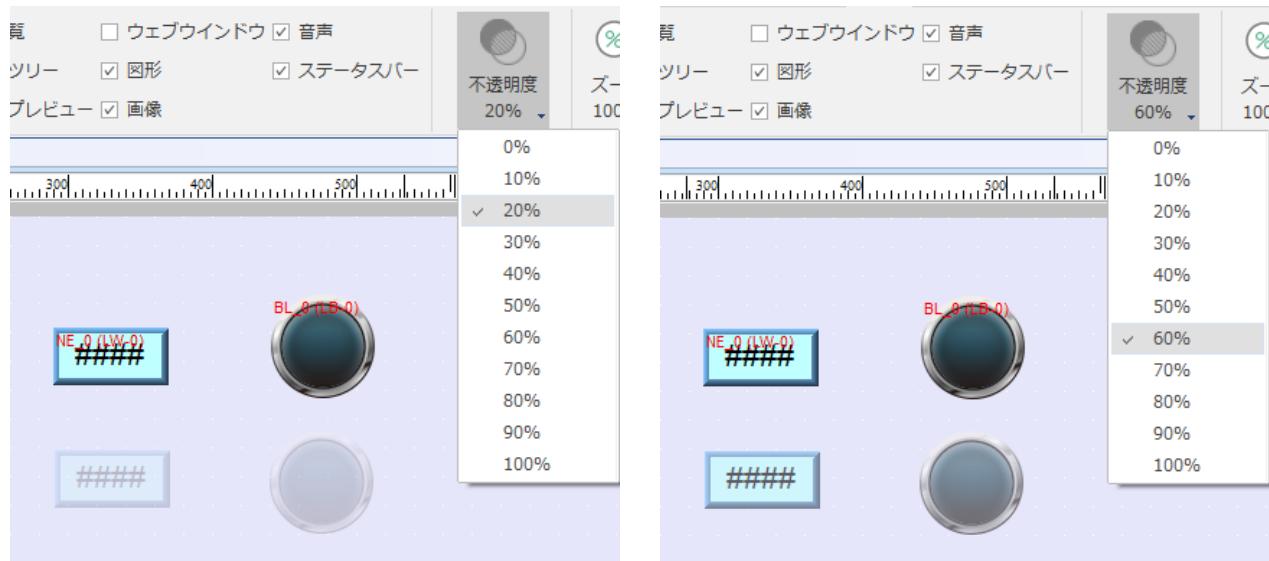
- ウィンドウツリーに閉じたい/削除したいウインドウ番号を選択した後、マウスを右クリックし[閉じる]/[削除]を選択します。
- [閲覧] » [ウインドを開く]で削除したいウインドウ番号を選択した後、[削除]を選択します。

6.4. ウィンドウの透明度

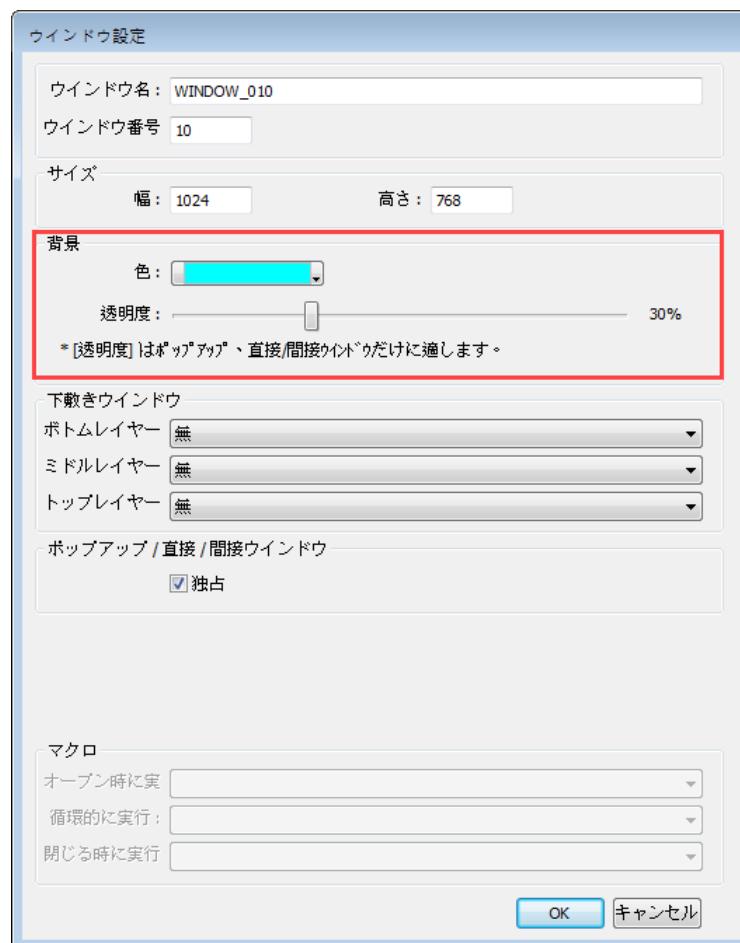
EasyBuilder Pro でプロジェクトを編集する際に、ウィンドウの[不透明度]を設定でき、ウィンドウレイヤー、共有ウィンドウ、下敷きウィンドウに適用しています。

下図に示されたとおり、[不透明度]が大きければ大きいほど、前述のウィンドウ内のオブジェクトがよりはっきり見えます。

ウインドウ



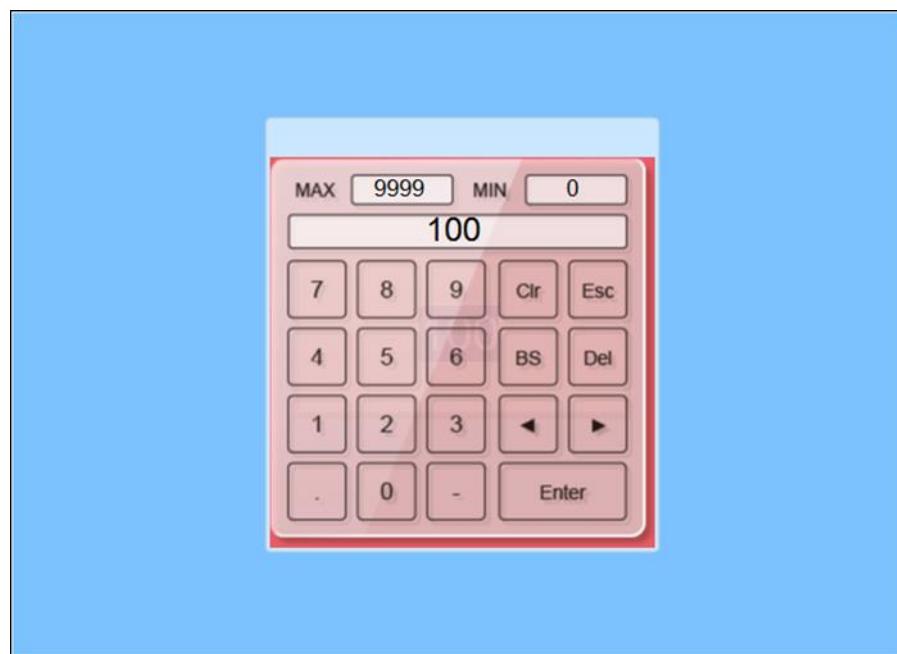
また、cMT シリーズのウインドウは[透明度]を設定でき、パップアップウインドウ、直接/間接ウインドウ及びキーボードに適用しています。下図に示す通り、[透明度]の値が大きければ大きいほど、ベースウインドウにあるパップアップウインドウに重ねられたオブジェクトがよりはっきりと見えるようになります。



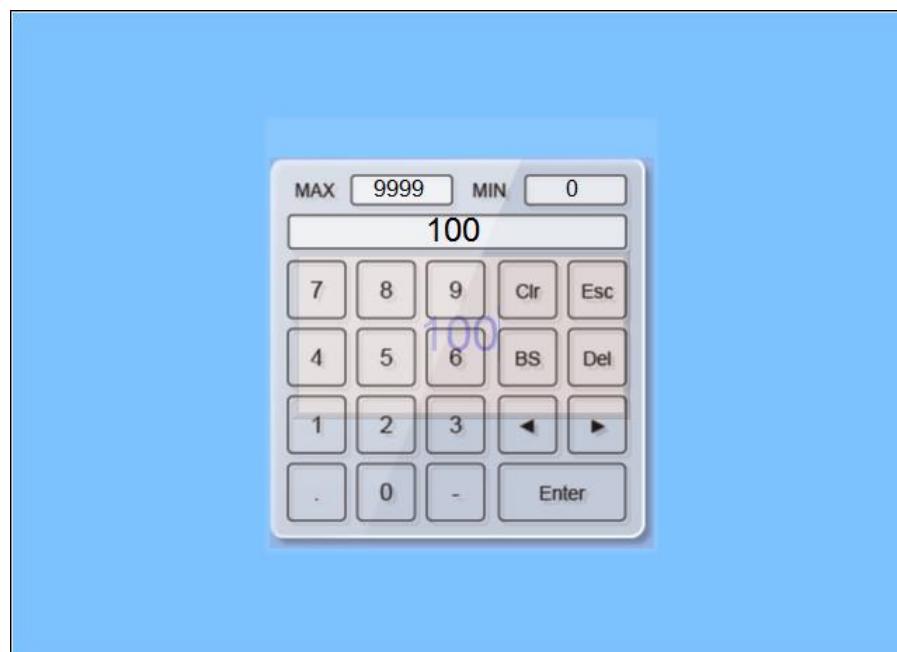
例

ウインドウ

[透明度]を 40%に設定した場合：



[透明度]を 90%に設定した場合：

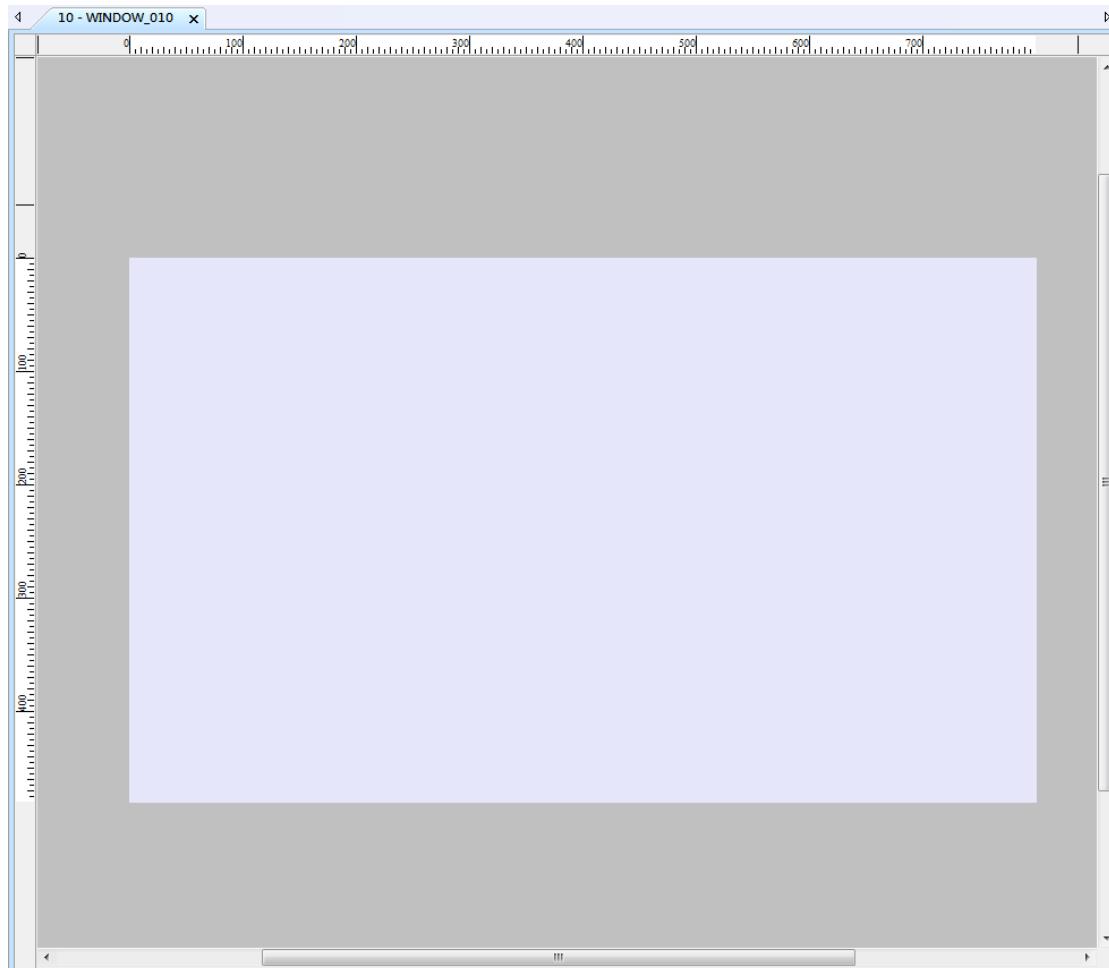


6.5. 編集ティップス

EasyBuilder Pro は定規、オブジェクトのクイックコピーなどの機能を有し、ユーザーにフレンドリーなインターフェースを提供します。以下は数点の編集ティップスです。これらのティップスを使用することで、編集の効率を向上できます。

6.5.1. 非表示エリア、選択ツール、手のひらツール

下図に示した通り、編集ウインドウではバイオレット色に表示された表示エリアと、灰色に表示された非表示エリアに分けられます。バイオレット色のエリアは表示エリアで、ここに置かれたオブジェクトは HMI のモニターに表示されます。灰色のエリアは非表示エリアで、ここに置かれたオブジェクトは HMI のモニターに表示されないが、依然に装置と通信します。非表示オブジェクト(ビット/ワード設定、データ送信など)を非表示エリアに置き、または文字オブジェクトでコメントをすることで、編集画面のきれいさを維持できます。

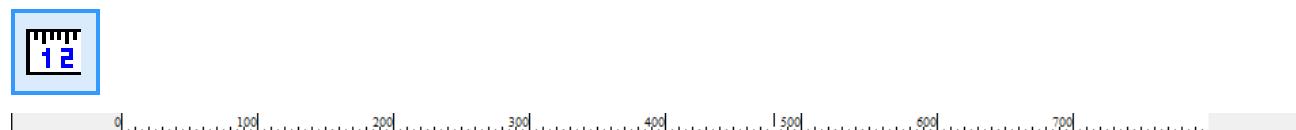


選択ツール

選択ツールを使用する場合、マウスを左クリックしたままドラッグすると、一回で複数のオブジェクトを選択することができます。

	<p>手のひらツール 手のひらツールを使用する場合、マウスを左クリックしたままカーソルを移動すると、PDF ファイルを閲覧するように直接に画面を移動することができます。</p>
---	---

6.5.2. 定規



[閲覧] » [定規]を選択すれば、編集ウインドウの上端と左側に定規が現れ、定規の上に右クリックしてガイドを新規追加することができます。ガイドはオブジェクトの整列に用いられます。オブジェクトをドラッグして移動、または拡大/縮小する時に、オブジェクトは自動的にガイドに揃えます。

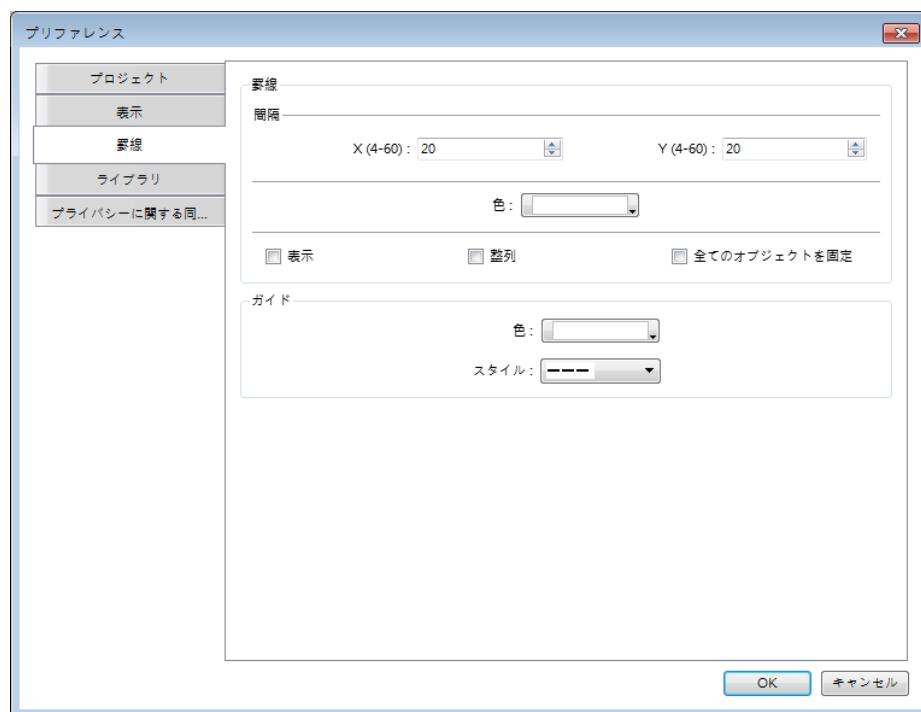


設定	記述
削除する	ガイドを削除します。
編集	 手動でガイドの位置を編集します。

設定

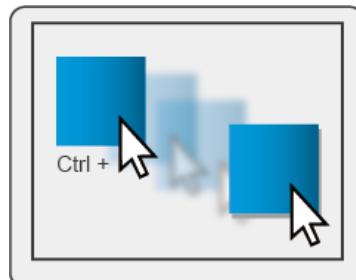
ここで複数のガイドを新規追加、または削除できます。

また、メインメニューで [ファイル] » 「プリファレンス」 » 「罫線」ウインドウでガイドの色とスタイルを変更することができます。



6.5.3. クリックコピー機能

オブジェクトが選択されている状態で Ctrl キーを押し続ける同時に、コピーを置きたい場所でカーソルをドラッグすれば、オブジェクトが指定した場所でコピーされます。



7. イベントログ

本章では、イベントログの設定及び使用の仕方について説明します。

7.1.	概要	7-2
7.2.	イベントログの管理	7-2
7.3.	イベントログを新規作成する	7-8

7.1. 概要

イベントログ機能を使用するには、基本的な手順は以下のとおりです：

1. イベントをトリガーする条件とその内容を定義する。
2. 条件に基づいてイベントをトリガーする。
3. イベントを指定した位置に保存する。
4. オブジェクトでイベントの処理プロセスを確認できる。

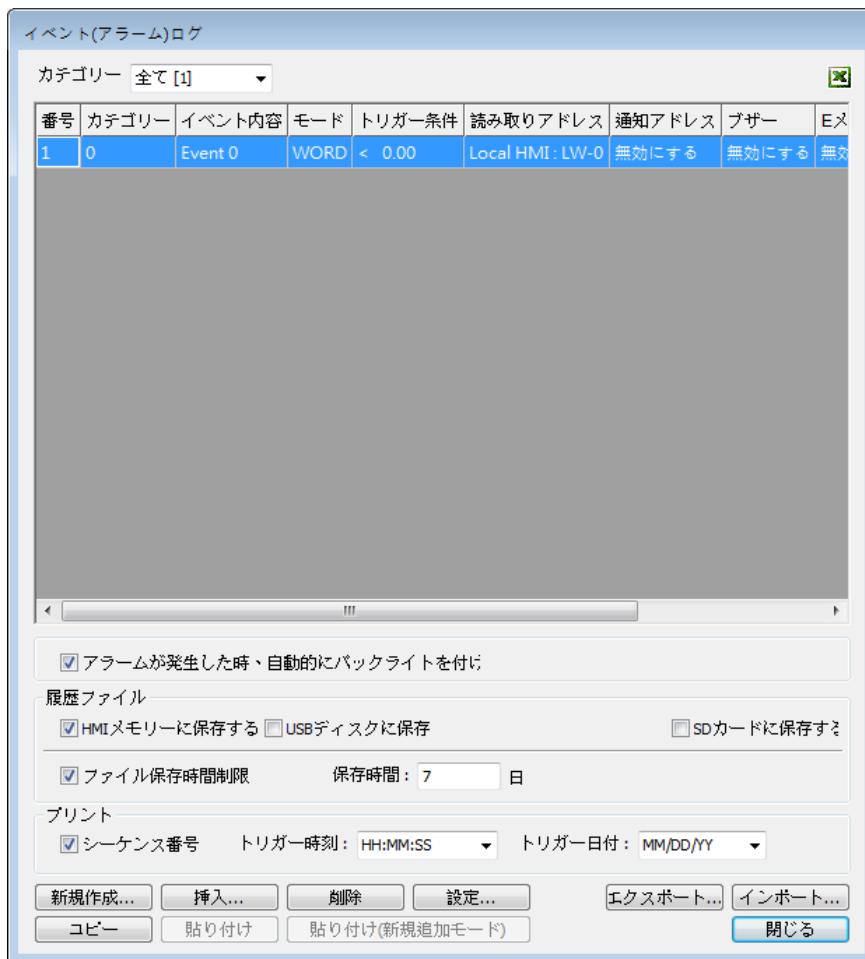
本章はイベントログの設定及び使用の仕方について説明します。

このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

7.2. イベントログの管理

アラームバー 、アラーム表示 、イベント表示 などのオブジェクトを通じて、イベントが発生→処理を待つ→アラームが解除までの時間が分かるようになります。まずはイベントの内容を定義する必要があります。イベントログの最大レコード数は 10000 レコードです。

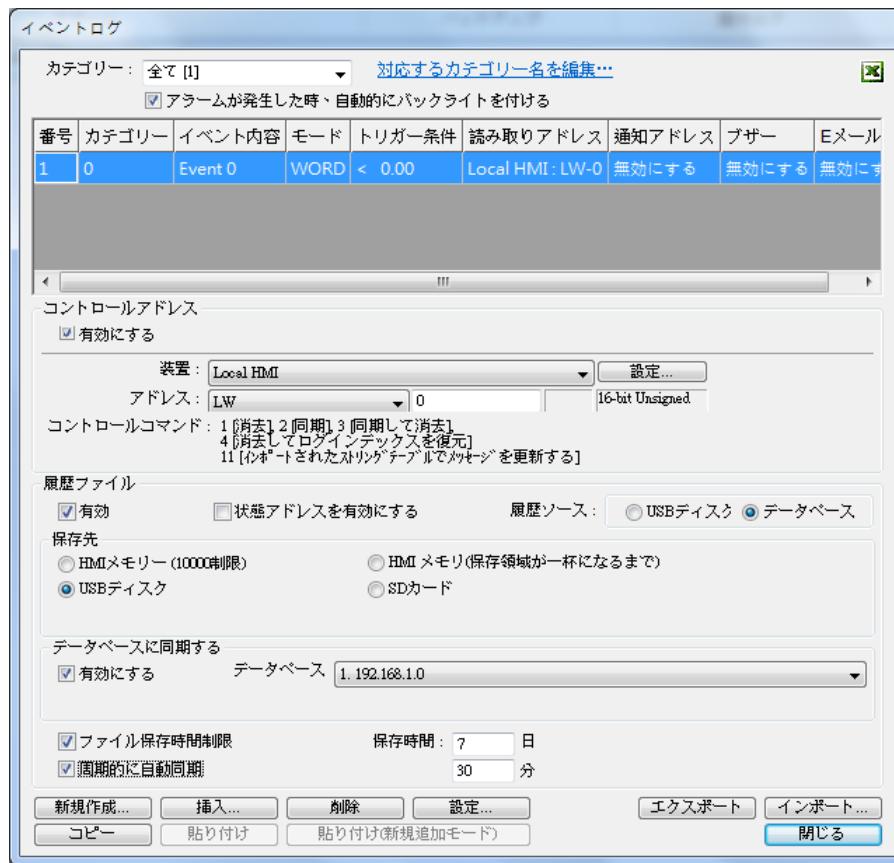
7.2.1. eMT、iE、XE、mTV、iP シリーズ



設定	記述
カテゴリ	イベントを分類する機能を提供し、0~255種類に分類でき、一つの種類を取り出し、イベント情報を入力/表記できます。[]の中では本種類のイベント情報の数を表示します。
履歴ファイル	イベントログの保存場所を指定します。イベントが発生すると、HMIはすぐに履歴ファイルを保存します。しかし、オンライン/オフラインシミュレーションを使用する場合、ファイルは一切 HMI_memory/SD_card/USB に保存することになります。
データ保存時間制限	本設定でイベントログファイルが保存される最大数量を決めます(当日を含まない)。つまり、仮に[保存時間]を二日に設定する場合、システムは本日以外の最新の 2 イベントファイルのみ保存し、その前のファイルが自動的に削除されることで、空き領域を確保します。
プリント	前もって[システムパラメータ設定] » [HMI 属性]タブでプリンターのモデルを選択した場合のみ、イベントがトリガーされた

	際でのプリントフォーマットが選択できます。
コピー	選択した項目をコピーします。
貼り付け	貼り付けると、コピーしたデータを選択したイベントに上書きします。上書きする前には、上書きしますかとの確認メッセージがポップアップされます。
貼り付け (新規追加 モード)	貼り付けると、コピーしたデータをリストの末尾に置きます。

7.2.2. cMT シリーズ



設定	記述
カテゴリー	イベントを分類する機能を提供し、0~255種類に分類でき、一つの種類を取り出し、イベント情報を入力/表記できます。[]の中では本種類のイベント情報の数を表示します。
対応するカテゴリーを編集	: 各対応するカテゴリーの名前を編集できます。
コントロールアドレス	[コントロールアドレス]と[履歴ファイル]の[有効にする]にチェックマークを入れたら、ユーザーはコントロールアドレスで数值を入力することによって、装置内のイベントログを同期、あるいは消去することができます。

コントロールアドレスに特定の数値を書き込むと、関連コマンドをトリガーできます。

数値	コマンド
1	cMT HMI 内のデータを消去します。
2	データを外部装置に同期します。
3	まずはデータを外部装置に同期してから、cMT HMI 内のデータを消去します。
4	HMI を変更した後、USB ディスクまたは SD カードに保存された履歴データを使用する
6	イベントログ(*.db)用に予約されている未使用の保存領域を解放します。履歴データの保存先を[HMI メモリ(保存領域が一杯になるまで)]に設定した場合、このコマンドを使用できます。
11	イベントログメッセージの内容をインポートされた[ストリングテーブル]に更新します。

ユーザーが手動でコントロールアドレスに上記数値を入力していない場合、システムは下記[履歴ファイル]のデータ同期規則の説明文に基づき、イベントログを保存します。

状態/エラーアドレス(cMT シリーズ) コントロールアドレスは LW-n の場合、4 個の連続したレジスタ (LW-n+1 ~ LW-n+4) が状態及びエラーを表示することに用いられます。詳細はイベントログ設定ダイアログボックス内の記述を参考してください。



数値	状態アドレス : LW-n+1 及び LW-n+3
0	外部装置またはデータベースに接続していない
1	外部装置またはデータベースに接続中
2	外部装置またはデータベースに接続済み
3	イベントログを外部装置またはデータベースに保存中。完了後、数値 2 に復元する
数値	エラーアドレス : LW-n+2 及び LW-n+4
0	エラー無し

1	未知のエラー
2	外部装置またはデータベースへの接続に失敗した
3	操作権限が不足
4	不正確なデータベース名
5	データフォーマットが一致していない
6	テーブルをオープンできない
7	テーブルを作成できない
8	テーブルを書き込めない
9	データベースが開けない
10	データベースが破損した

履歴ファイル

イベントログデータの保存場所です。SD カード、USB ディスク、データベースに同期できます。同期の規則は以下のとおりです：

- イベントログが 10000 レコードに達していない場合、暫く cMT HMI に保存されます。
- イベントログが 10000 レコードに達し、それに[HMI メモリ (10000 レコード制限)]を選択した場合、システムは自動的にイベントログを指定した外部保存装置に接続し、それに cMT HMI 内の最も古い 1000 レコードのデータを消去します。
- イベントログが 10000 レコードに達し、それに[HMI メモリ(一杯になるまで)]を選択した場合、システムは継続にデータを HMI のメモリに保存し、削除しません。データベースに同期することができません。
- 外部保存装置内にすでにイベントログがある場合、毎回同期するたびに、新しいイベントログが本来のデータに付加されます。
- 外部保存装置を cMT HMI から外し、またはデータベースサーバーとの接続を切断した場合、新規追加されたイベントログが 9000 レコード未満のうちに再び外部装置を挿入/接続を再開すると、イベントは依然に cMT HMI に保存され、消去はされません。もし接続が中断された間に、イベントログが 9000 レコードに達したら、より古いデータが消去され、その後に接続を再開しても、消去されたデータが復元できません。

周期的に自動同期

有効にすると、HMI は[履歴ファイル]の同期規則を参照する上、データを設定された周期で外部装置に保存することになります。

単位：分間。

数値：1 ~ 1440。

ファイル 保存時間 制限	イベントファイルが HMI メモリー内で保存される最大日数です(当日を含まない)。イベントログの同期を実行する時のみ、ファイルの時間を検査して古いファイルを削除します。 [保存日数]を二日に設定する場合、システムは当日以外の最近二日のイベントログファイルだけを保存します。その前のファイルが自動的に削除されるようになります。
データベース内の履歴情報を表示する	本項目を選択すると、イベントログにはデータベースから読み取った履歴データを表示します。
コピー	選択した項目をコピーします。
貼り付け	貼り付けると、コピーしたデータを選択したイベントに上書きします。貼り付ける前には、上書きしますかとの確認メッセージがポップアップされます。
貼り付け (新規追加 モード)	貼り付けると、コピーしたデータをリストの末尾に置きます。



Note

- ユーザーが SD カード/USB ディスクを外す、またはデータベースとの接続を切断する必要がある場合、先にコントロールアドレスでイベントログデータを同期することができます。
- cMT Viewer で複数 HMI を監視した場合、操作している HMI の Home ボタンが点滅したら、他の HMI にイベントが発生したことを示しています。これによって、HMI を切替してイベントを確認するように注意喚起の効果が得られます。

7.2.3. Excel 編集

イベントログウインドウ右上隅の Excel アイコンをクリックすると、Excel テンプレートをオープンし、編集時の見本にすることができます。

本テンプレートはインストールディレクトリでの EventLogExample.xls ファイルで、テンプレートにはデザインされた検証機構とドロップダウンメニューがあります。

A	B	C	D	E	F	G	H	I	J	K
1 Category	Priority level	Address type	PLC name	Device type	System tag	User-defined tag	Address	Index	Data Format	Enable
2 0	Middle	Word	Local HMI	LW	False	False	100	null	32-bit Signed	True
3 1	Low	Bit	Local HMI	LB-9009	True	False	9009	IDX 5	16-bit BCD	▼ lse
4									16-bit BCD	
5									32-bit BCD	
6									16-bit Unsigned	
7									16-bit Signed	
									32-bit Unsigned	
									32-bit Signed	
									32-bit Float	

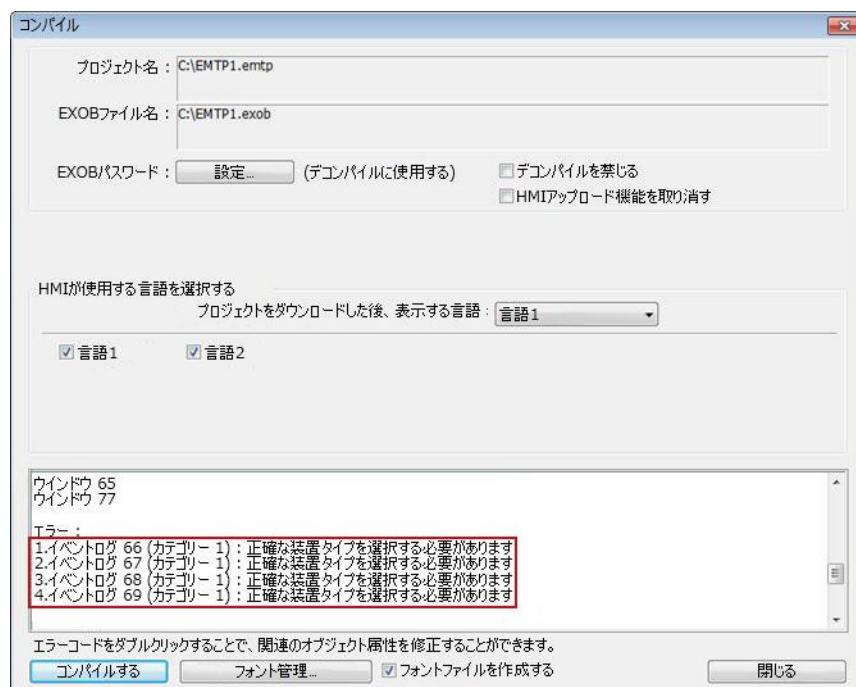


Note

- [System tag] と [User-defined tag] は同時に true にしないでください。さもないと、システムは [System tag] を認め、[User-defined tag] を false に視します。[Device type] で [User-defined tag] を入力した場合、[System tag] を false にしてください。
- Excel 表で [User-defined tag] を true にしても、システムが [Device type] をシステム内のユーザ一定義 tag を合わせてみると、対応する tag が見つけられない場合、自動的にイベントログ内の [User-defined tag] を false にします。
- [Color] フォーマットは R:G:B で、0~255 範囲内の整数でなければなりません。
- ラベルタグライブラリ或いはサウンドライブラリをインポートする前、システムには対応する名前が存在していることを確認してください。

7.2.4. 迅速に無効なイベントを閲覧する

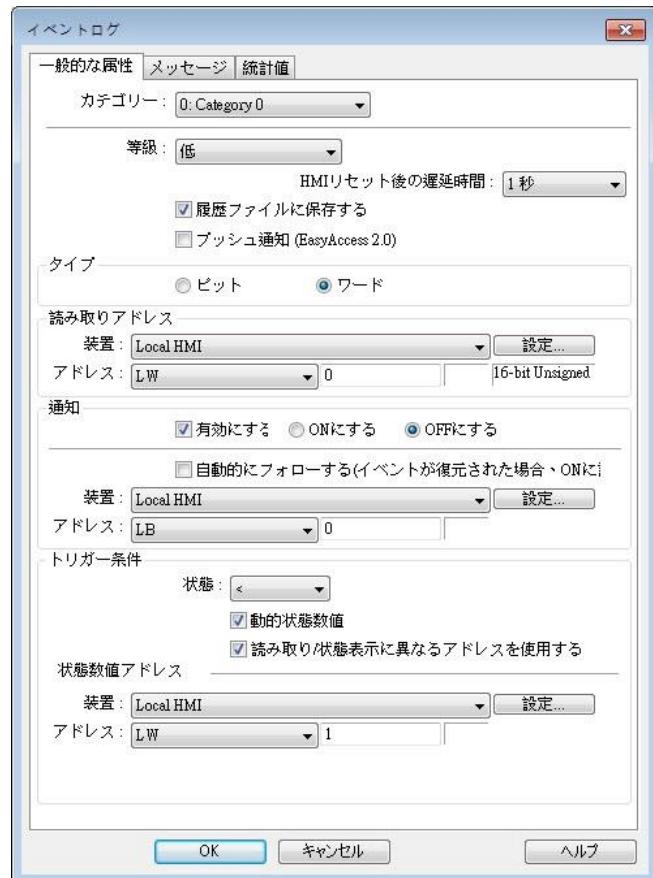
プロジェクトをコンパイルする時、無効なイベントログがあれば、コンパイルダイアログボックスの中に表示されます。エラーのある項目をダブルクリックと、「イベントログ」オブジェクトが自動的にオープンされ、それにエラーのある項目に指向します。



7.3. イベントログを新規作成する

一般的な属性の設定

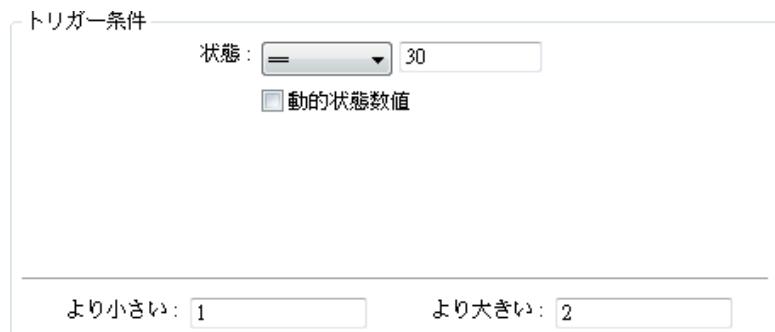
[イベントログ] ウィンドウの [新規作成] をクリックすると、[一般的な属性] という設定タブが出ます。



設定	記述
カテゴリ ー	イベントの種類を選択します。範囲は 0~255 です。
等級	発生したイベントの数がシステムの許可する最大値(デフォルト : 1000)に達すると、重要さがより低いイベントが削除され、新しいイベントが入れられます。 アラーム表示オブジェクトに表示されるイベントはまず等級によって配列されてから、時刻によって配列されます。
履歴ファ イルに保 存する	イベントログのメイン設定タブで、[履歴ファイル]オプションの保存機能を選択すると、各イベントを履歴ファイルに保存するのかを設定できます。
プッシュ 通知 (EasyAccess 2.0)	本項にチェックマークを入れると、イベントが発生した時、iOS/Android 装置で EasyAccess2.0 を通じて、イベントのプッシュ通知を受け取ることができます。
HMI リセ ット後の 遅延時間	[動的状態数値]と合わせて使用します。HMI を起動した時に、動的状態数値がまだ設定していない可能性があり、[HMI リセット後の遅延時間]機能を使用すると、イベントを読み取る時間を遅延できます。本機能は HMI を起動する際に一回だけ実

	行します。
読み取りアドレス	システムは本アドレスのデータを読み取ることによって、イベントがトリガー条件に満足しているのかを検査します。
通知	本項を選択すると、システムはイベントが発生する際に、指定したレジスタを[ON]または[OFF]にします。 自動的にフォローする アラーム状態が正常に戻ったら、[通知]アドレスの状態を復帰します。例：アラームがトリガーされた時、通知機能で指定ビットを ON にするが、[自動的にフォローする]を選択した場合、正常に戻ったたびに、指定ビットが OFF に設定されます。
トリガーアクション	ビットを選択する場合、イベントログは一個のビットアドレスの状態を検査します。 動的状態数値 ワードを選択する場合、イベントログは一個のワードアドレスの数値が一個の特定数値にイコール/より大きい/より小さいなどのかを検査します。下記例 1、例 2 をご参照ください。
	読み取り/状態表示に異なるアドレスを使用する 本機能を使用すると、自分でトリガー条件の参照ソースアドレスを設定することができます。

例 1



上記内容の意味は：

[読み取りアドレス]内の数値 $\geq 29 (=30-1)$

または $\leq 31 (=30+1)$ の場合、イベントがトリガーされます。つまり、イベントのトリガー条件は：

$29 \leq [\text{読み取りアドレス}] \leq 31$

イベントがトリガーされた後、[読み取りアドレス]内の数値 $> 32 (=30+2)$ または $< 28 (=30-2)$ の場合、

システムは正常状態に戻ります。つまり、システムが正常状態に戻る条件は：

[読み取りアドレス]内の数値 <28 または [読み取りアドレス]内の数値 >32

例 2



上記内容の意味は：

[読み取りアドレス]内の数値 $<29 (=30-1)$

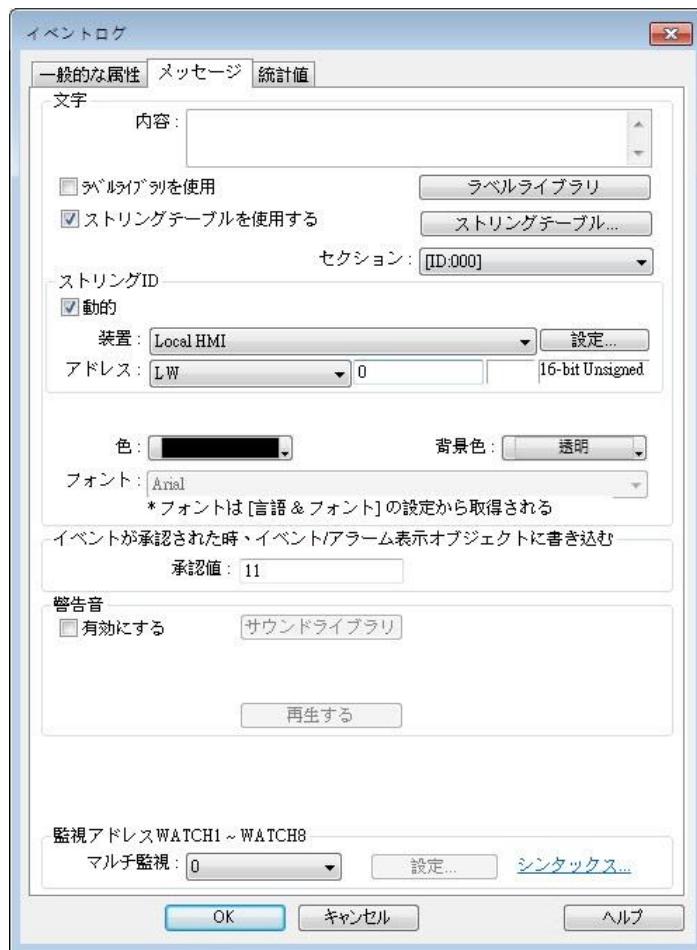
または $>31 (=30+1)$ の場合、イベントがトリガーされます。つまり、イベントのトリガー条件は：

[読み取りアドレス]内の数値 <29 または [読み取りアドレス]内の数値 >31

イベントがトリガーされた後、[読み取りアドレス]内の数値が $\geq 28 (=30-2)$ または $\leq 32 (=30+2)$ の場合、システムは正常状態にも戻ります。つまり、システムが正常状態に戻る条件は：

$28 \leq$ [読み取りアドレス]内の数値 ≤ 32

メッセージ設定



設定

記述

内容

イベントログが[アラームバー]、[アラーム表示]と[イベント表示]オブジェクトで表示する情報内容です。また内容の中で[監視アドレス]WATCH1~WATCH8の資料を使用することもできます。文字内容のソースはラベルタグライブラリ、またはストリングテーブルでもいいです。

フォント/ 色/背景色

一個のイベントは分別にフォント、色、背景色を設定できます。[アラームバー]にはフォントと色の設定ができます。[アラーム表示]と[イベント表示]にはフォント、色と背景色の設定ができます。履歴モード下の[イベント表示]には、上記の設定ができません。.

イベント が承認さ れた時に 書き込む

[イベント表示]と[アラーム表示]オブジェクトでのイベントが承認された際に、本数値を当オブジェクトが指定した[書き込みアドレス]に書き込みます。



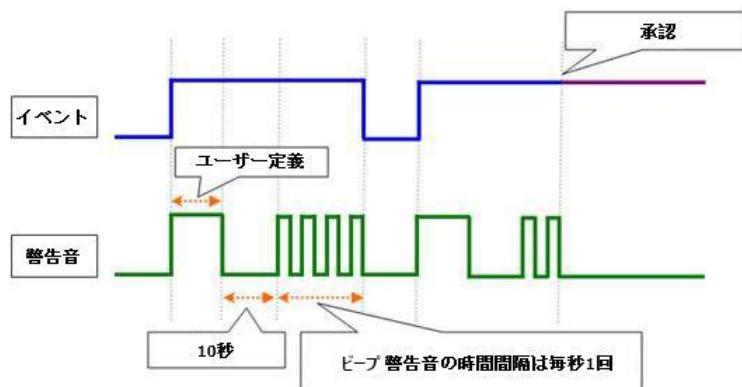
このアイコンをクリックし、サンプルプロジェクトをダウ

ンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

警告音

有効にすると、イベントが発生した時は指定した警告音を放送します。またイベントが承認された/正常状態に戻った前、継続に警告音を放送することが選択できます。

継続に警告音を放送するにチェックマークを入れた後、アラームが発生した時に、指定した時間を経ってから、ビープ警告音を出すことが選択できます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

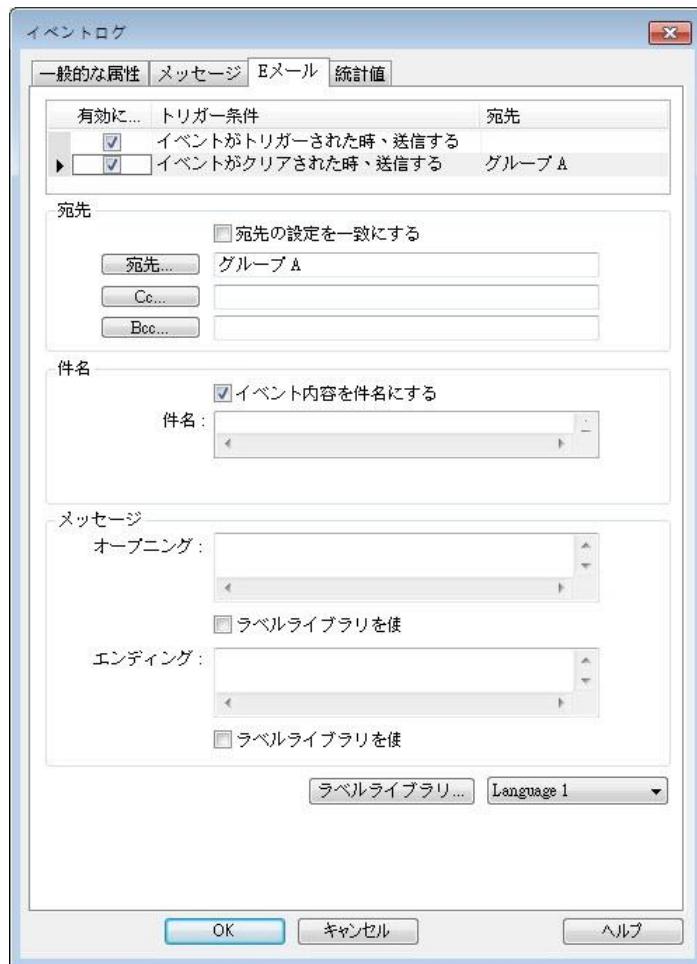
監視アドレス

イベントがトリガーされた場合、監視アドレスレジスタ内の数値を表示するように設定することができます。同時に最大8個のアドレスを監視できます。[シンタックス]をクリックすれば、イベントログで表示される監視アドレス内の数値のシンタックスを確認できます。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

Eメール設定

お先に[システムパラメータ設定] » [Eメール]で本機能を選択してください。

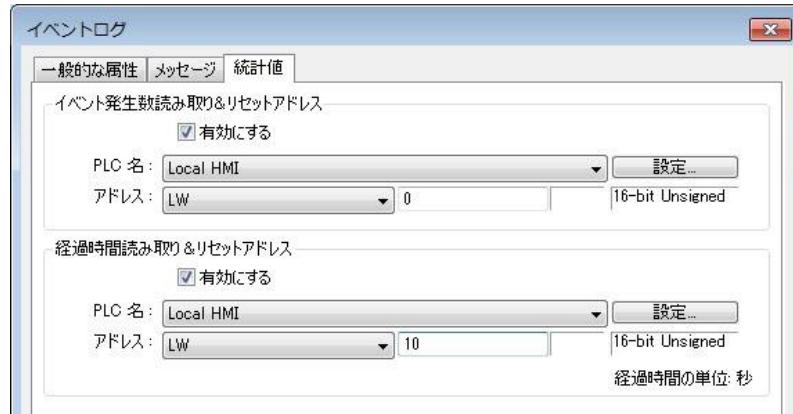


設定	記述
宛先	個別に宛先、カーボンコピー、プライドカーボンコピーの欄で宛先を選択することができます。
件名	メールのテーマを表示できます。
メッセージ	メールが表示する始めと終わりの内容を入力できます。
添付ファイル	本項にチェックマークを入れると、指定したウインドウのスクリーンショットを選択し添付することができます。

Note

- メールの差出人が定義したイベントの等級は、メールの重要度に影響します。イベント等級が緊急である場合、メールの重要度が高；等級が高と正常である場合、重要度が一般；等級が低である場合、重要度が低と等しいです。

統計値設定



設定

記述

イベント

発生数読み取り&リセット

有効にすると、HMI を起動してから発生したイベントの発生数をワードアドレスに書き込みます。

セットアドレス

このワードアドレスを読み取る/書き込むことができます。

経過時間読み取り&リセット

アドレス

有効にすると、当該イベントが発生してから正常に戻った時までに累積された秒数をワードアドレスに書き込みます。このワードアドレスを読み取る/書き込むことができます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

8. データサンプリング

本章では、データサンプリングの設定及び使用の仕方について説明します。

8.1.	概要	8-2
8.2.	データサンプリングの管理	8-2
8.3.	新規データサンプリングオブジェクトを作成する	8-2
8.4.	cMT Viewer のデータを外部装置またはデータベースに同期して保存する	8-13
8.5.	cMT Viewer の特定の日付またはファイルの履歴データを調べる	8-13

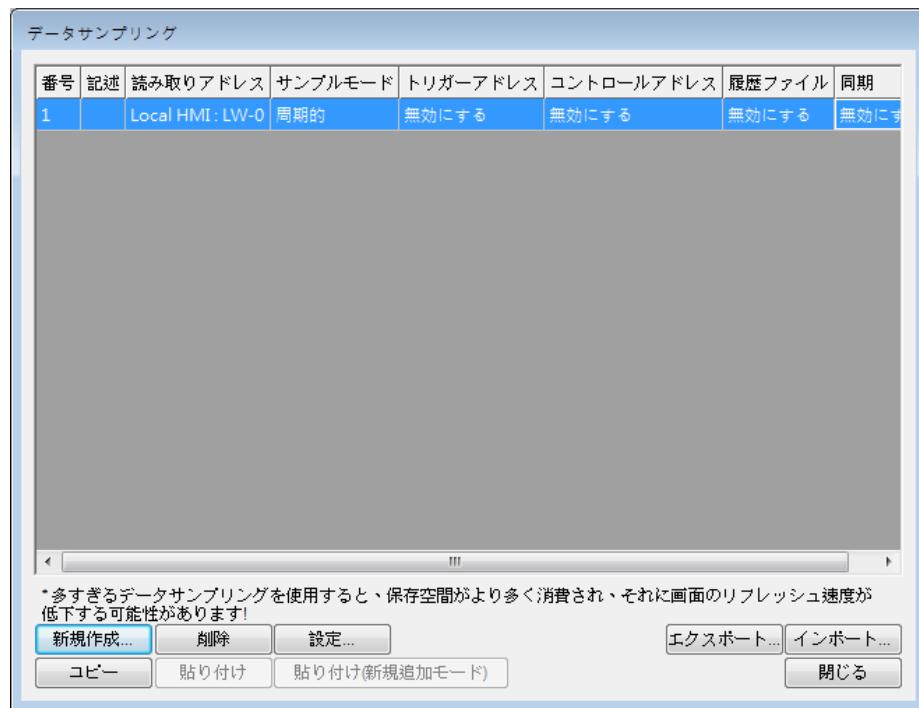
8.1. 概要

「データサンプリング」のサンプリング方法を定義します。例えば：サンプリング時間、サンプリングアドレス、及びデータの長さを定義した後、取ったサンプリングデータを指定した場所(例：HMI メモリー、SD カード、USB ディスク)に保存することができます。データサンプリング機能はトレンド図や履歴データ表示オブジェクトに合わせ、サンプリング内容を検査することができます。

8.2. データサンプリングの管理

新規のデータサンプリングを作成するには、下記手順に従ってください。

1. オプションリストの[データ/履歴]をクリックし、[データサンプリング]を選択してください。
2. [新規作成]をクリックして、下記のように関連設定を始めます。

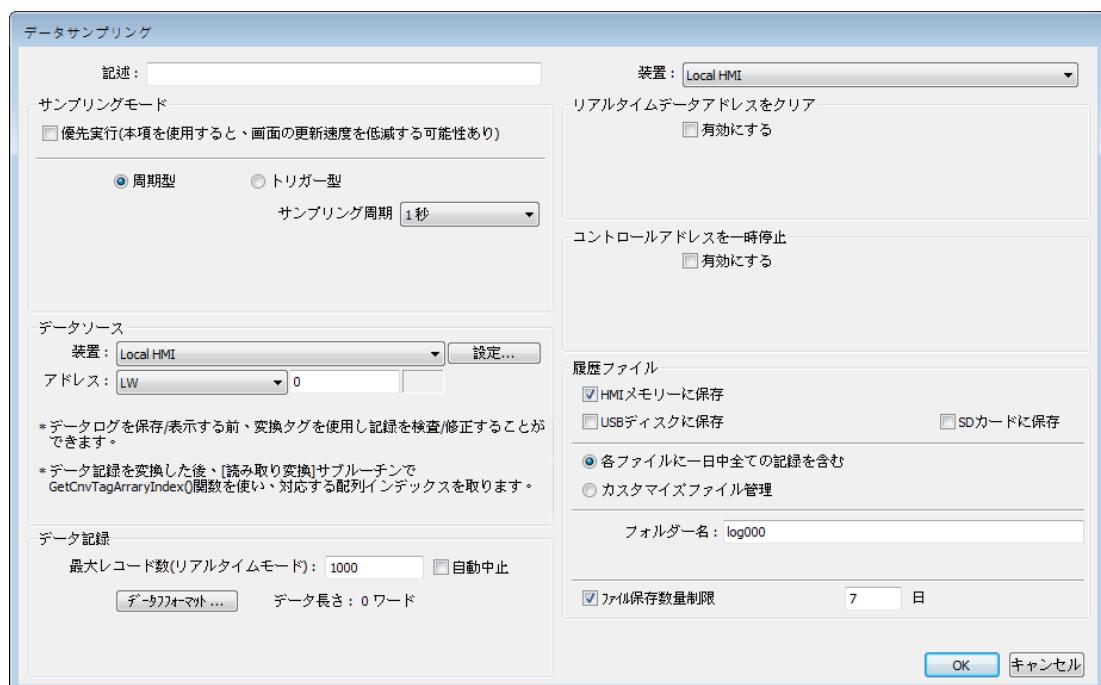


8.3. 新規データサンプリングオブジェクトを作成する

以下は一個の新規データサンプリングオブジェクトを作成・設定する仕方を紹介します。

eMT、iE、XE、mTV、iP シリーズ

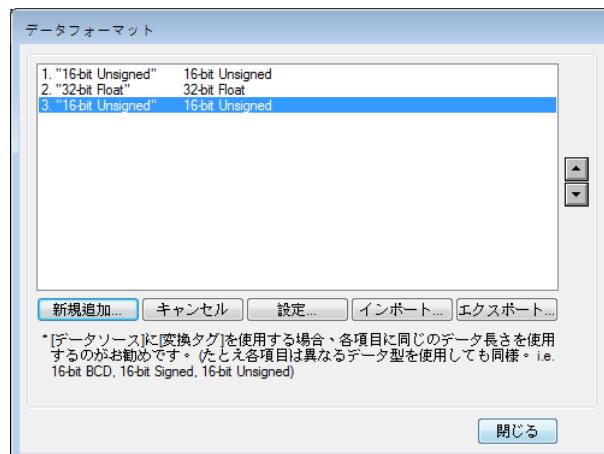
データサンプリング



設定	記述
サンプリングモード	優先実行
ド	データサンプリング機能を優先に実行します。注意すべきなのは、HMIはデータサンプリングを優先に実行するので、HMI画面でのオブジェクトの更新速度が低下します。優先実行の設定が多くなると、更新速度は遅くなるので、多すぎた優先実行の設定を避けてください。
周期型	固定した時間頻度でデータサンプリングを行います。[サンプリング周期]は0.1秒~120分範囲内で設定できます。
トリガー型	一個の特定ビットアドレスの状態で、サンプリング動作をトリガります。
[モード]は：	<p>[OFF->ON]指定アドレスの状態がOFFからONになると、データサンプリングをトリガーします。</p> <p>[ON->OFF]指定アドレスの状態がONからOFFになると、データサンプリングをトリガーします。</p> <p>[OFF<->ON]一旦指定アドレスの状態が変わったら、データサンプリングをトリガーします。</p>
サンプリング完了後、ON/OFFに設定する	本項にチェックマークを入れると、データサンプリングがトリガーされた後、システムはトリガービットをON/OFFに復帰します。

データサンプリング

データソース	サンプリングソースとして、一個の装置アドレスを選択します。データを計算する必要がある場合、[読み取り変換]機能があるユーザー定義タグを使用しなければなりません。配列アドレスには GetCnvTagArrayIndex 関数を使用して対応する配列インデックス入手してから計算することができます。
データ記録(リアルタイムモード)	リアルタイムモードで、もし[自動停止]にチェックマークを入れない場合、最大 86400 レコードのサンプリングデータを保存できます。86400 レコードを超えたたら、一番古いレコードから削除されます。 データフォーマット 複数の異なるフォーマットの連続アドレス内のデータを読み取るようく設定できます。例えば、アドレスやフォーマットを LW-0(16-bitUnsigned)、LW-1(32-bitFloat)、LW-3(16-bitUnsigned) に設定します。上限は 1000 レコードです。

**自動中止**

データサンプリングが最大数に達したら、サンプリングを停止します。

詳細は《8.3.1 自動中止の説明》をご参照。

リアルタイムデータアドレスをクリア	指定したビットアドレスの状態が [OFF->ON] または [ON->OFF] になる時、トレンド図の [リアルタイムモード] でサンプリングされたデータを消去し、サンプリングデータの数量もゼロに戻されます。ただし、履歴データ内のサンプリングデータは影響されません。
コントロールアドレスを一時停止	指定したビットアドレスの状態が ON 或いは OFF に設定されると、復帰するまでサンプリングが一時停止されます。
履歴ファイル	HMI メモリーに保存 データサンプリングを HMI に保存します。データが 10 秒ごとに一

回保存されます。10秒未満の場合、システムレジスタ[LB-9034]で強制的に保存することができます。(LB-9034の使用制限に関しては、下記 Note 6 項をご参照ください)

SD カード/USB ディスクに保存

データサンプリングを指定した外部記憶装置に保存します。

各ファイルに一日中全ての記録を含む

データサンプリングは一日を単位とし、日付順番でファイルを指定したファイル名のフォルダに保存します。ファイル名のフォーマットは *yyyymmdd.dtl* です。

フォルダ一名

サンプリングフォルダの名前を設定します。全部 ASCII 文字で構成する必要があります。

保存先 : [保存場所]¥[サンプリングフォルダ名]¥[ファイル名]

ファイル保存数量制限

本項の設定値でデータサンプリングファイルが保存される数量を決めます(当時動作している dtl ファイルを含まない)。つまり、保存日数を二日に設定した場合、システムは現在サンプリングしているファイル以外の最新の 2 ファイルを保存します。

カスタマイズファイル管理	ユーザー自分でデータサンプリングファイル名(*.dtl)とファイルの区分方式を設定する機能を提供します。
	詳細は《8.3.2 カスタマイズファイル管理》をご参照。

Note

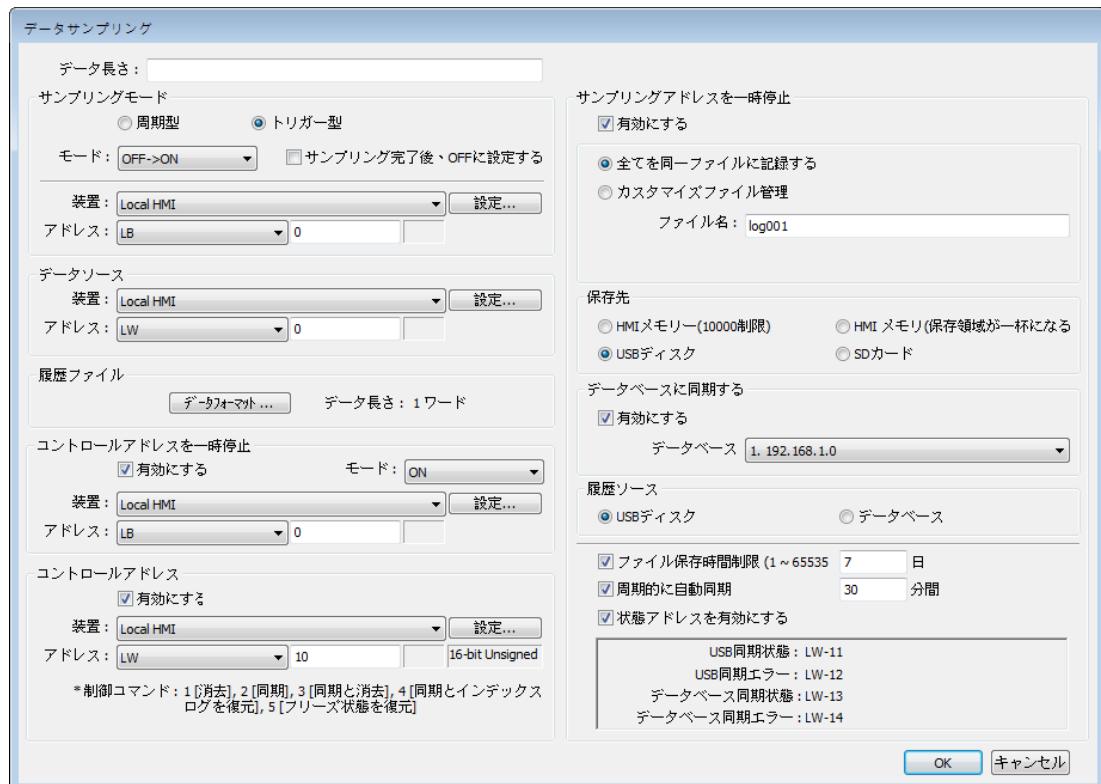
- 1 個のサンプリングデータには 1 タイプ以上のデータを含む可能性があります。データサンプリングは同時に異なるタイプのデータをサンプリングするのが可能です。ご自分でサンプリングデータの内容を定義することができます。例えば：3 タイプのデータで、合計長さが 4 words に定義された場合、毎回サンプリングを実行する際に、システムは指定したソースアドレスに長さ 4 words のデータを 1 サンプリングの内容としてサンプリングします。
- [各ファイルに一日中全ての記録を含めている]機能を使用し、また[ファイル保存数量制限]を 2 ファイルと設定した場合、システムは昨日と一昨日のデータサンプリングファイルのみ保存し、この時間範囲外のファイルは自動的に削除され、保存領域が不足になる恐れを避けます。
- [カスタマイズファイル管理]機能を使用し、また[ファイル保存数量制限]を 2 ファイルと設定した場合、システムは現在サンプリング中のファイル以外、別途に最新の 2 ファイルを保存するので、合計 3 ファイルになります。他のファイルは自動的に削除され、保存領域が不足になる恐れを避けます。
- PC でシミュレーション機能を使用する時、データサンプリングファイルは全部インストー

ルディレクトリの「保存場所」内の **datalog** サブディレクトリに保存します。ルートは：
C:¥EBpro¥[保存位置]¥datalog です。この場合、データサンプリングのフォーマットを変更したかったら、システムが誤って古いレコードを読み取るのを避けるため、先にインストールディレクトリでの古いデータサンプリングレコードを削除する必要があります。

- ファイルを USB ディスク或いは SD カードに保存する時、一個の **FAT32** フォルダに保存できるファイル数はファイル名の長さによります。ファイル名が長ければ長いほど、保存できるファイル数が少なくなります。
 - LB-9034 のトリガー間隔は最速 2 秒で 1 回実行されます。即ち、LB-9034 をトリガーした後、2 秒経つてから再度にトリガーされます。
- ▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

データサンプリング

cMT シリーズ(cMT シリーズで一部の設定は eMT、iE、XE、mTV、iP シリーズと同じなので、特別な説明がなかったら、直接に参照すればいいです。)



設定

記述

データ記録 複数の異なるフォアーマットの連続したアドレスのデータを読み取るように設定できます。上限は 1000 レコードです。

コントロールアドレス コントロールアドレスでコマンド数値を入力する際に、履歴データに特定したコマンドを送信します。

数値	コマンド
1	HMI 内のデータを消去する
2	データを外部装置/データベースに同期する
3	まずデータを外部装置/データベースに同期してから、HMI 内のデータを消去する
4	HMI を変更した後、USB ディスクまたは SD カードに保存された履歴データを使用する
5	HMI の空き領域が不足になった場合、データサンプリングが一時停止され、空き領域が出たらこのコマンドを使用すると、データサンプリングが再開される
6	データログ(*.db)用に予約されている未使用の保存領域を解放します。履歴データの保存先を[HMI メモリ(保存領域が一杯になるまで)]に設定した場合、このコマンドを使用できます。

状態/エラーアドレ

コントロールアドレスは LW-n の場合、4 個の連続したレジスタ (LW-n+1 ~ LW-n+4) が状態及びエラーを表示することに用いられま

ス
(cMT シリ
ーズ)

す。 詳細はイベントログ設定ダイアログボックス内の記述を参考して下さい。



数値	状態アドレス : LW-n+1 及び LW-n+3
0	外部記憶装置またはデータベースに接続していない
1	外部記憶装置またはデータベースに接続中
2	外部記憶装置またはデータベースに接続済み
3	イベントログを外部記憶装置またはデータベースに保存中。完了後、数値 2 に復元する
数値	エラー アドレス : LW-n+2 及び LW-n+4
0	エラーレコード
1	未知のエラー
2	外部記憶装置またはデータベースへの接続が失敗した
3	操作権限不足
4	不正確なデータベース名
5	データフォーマットが一致していない
6	テーブルをオープンできない
7	テーブルを作成できない
8	テーブルに書き込めない
9	データベースが開けない
10	データベースが破損した

履歴ファ
イル

履歴データの保存先は USB ディスク、SD カード、データベースサーバーが選べます。cMT のサンプリングデータが 10000 レコードに達し、それに[HMI メモリ(10000 レコード制限)]を選択すると、自動的にサンプリングしたデータを指定した外部記憶装置に保存し、最初の 1000 レコードのデータを削除します。[HMI メモリ(一杯になるまで)]を選択した場合、システムは継続にデータを HMI のメモリに保存し、削除しません。データベースに同期することができません。データベースサーバーを保存先に選択する場合、データベースサーバーが所在する PC の IP アドレスを設定する必要があります。

👉 同期規則は《8.4 外部記憶装置を cMT Viewer にデータを同期する》をご参照ください。

👉 データベースサーバーの使用方法については、《13.44 データベースサーバー》をご参照ください。

カスタマ
イズファ
イル管
理

ユーザー自分でデータサンプリングファイル名(*.dtl)とファイルの分割方式を設定する機能を提供します。現在サンプリング中のファイルは全部 HMI メモリーに保存されます。USB ディスク/SD カード

に同期するのを選択する場合、下記 2 点をご留意ください：

1. データサンプリングを保存する db ファイルの名前が変更された場合、本来の db ファイルは USB ディスク/SD カードに同期されます。
2. 新しい db ファイルが生成された時、USB ディスク/SD カードが HMI に接続していなかったら、古い db ファイルは直接に削除されます。例えば：現在、サンプリングされたデータが 20161218.db に保存されているが、もしこの時、20161219.db が生成され、それに USB ディスク/SD カードが HMI に接続していない場合、20161218.db は直接に削除されます。

 詳細は《8.3.2 カスタマイズファイル管理》をご参照。

周期的に自動同期

有効にすると、HMI は[履歴ファイル]の同期規則を参照する上、データを設定した周期で外部装置に保存することになります。

注意すべきなのは、周期的に自動同期のカウンターは、コントロールアドレスが使用された時にリセットされます。

 《8.4. cMT Viewer のデータを外部装置またはデータベースに同期して保存する》をご参考ください。

ファイル保存制限

[全てを同一ファイルに記録する]を使用する

同期目的地が SD カードまたは USB ディスクの場合、ファイルの保存期限を 1~65535 に設定できます。

[カスタマイズファイル管理]を使用する

同期目的地が SD カードまたは USB ディスクの場合、ファイルの保存数を 1~65535 個に設定できます。同期目的地が HMI メモリーの場合、ファイルの保存数を 1~1000 個に設定できます。

履歴ファイルソース

履歴ファイルを読み取る時、ファイルソースを USB ディスク、SD カードまたはデータベースが選べます。

8.3.1. 自動中止の例

異なるオブジェクトに組み合わせることによって、[自動中止]の作用が異なります。下記のように例を挙げます：(最大データ数を n にする場合)

オブジェクト	[自動中止]を選択しない	[自動中止]を選択する
トレンド図-リアルタイムモード	より古いサンプリングデータを削除し、取ったばかりの第 n レコードのデータをトレンド図に表示します。下記図解をご参照ください。	第 n レコードになると、動作を停止します。
トレンド図-履歴モード	データは継続にサンプリングされ、全ての履歴データをトレンド図に表示します。	第 n レコードになると、動作を停止します。
履歴データ表示	データは継続にサンプリングされ、全ての履歴データを履歴データ表示に表示しま	第 n レコードになると、動作を停止します。

す。

データサンプリング

継続に新データをサンプリングします。

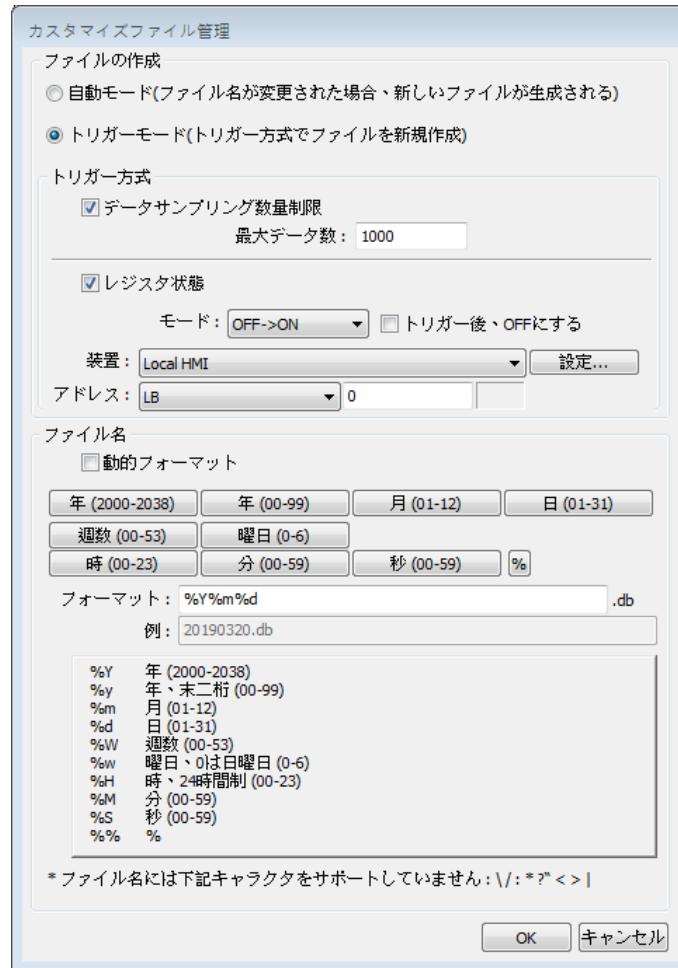
第 n レコードになると、動作を停止します。

例：データ数を 10 個に設定する場合、11 番目のデータが発生した時、[自動中止]を選択していない場合、一番古いデータが削除され、最新のデータ記録を記入します。下図をご参照ください。

レコード数	データ	選択しない (自動停止)
1	101	102
2	102	103
3	103	104
4	104	105
5	105	106
6	106	107
7	107	108
8	108	109
9	109	110
10	110	111
11	111	

8.3.2. カスタマイズファイル管理

ユーザー自分でデータサンプリングファイル名(*.dtl, *db)とファイルの区分方式を設定する機能を提供します。



設定	記述
ファイルの作成	自動モード ファイル名が変更されたら、新しいファイルが作成されます。
トリガーモード	[トリガー方式]の設定によって、新しいファイルが作成されます。
データサンプリング数量制限	サンプリング数量が[最大データ数]の設定値に達した場合、新しいファイルが作成されます。
レジスタ状態	一個の特定ビットアドレスの状態変化によって新しいファイルを作成することを指定します。状態が[モード]での設定に合った時、新しいファイルが作成されます。
トリガー後、ON/OFFに設定する	

指定したビットアドレスの状態が変化し、新しいファイルが作成される時、本来の状態に復帰します。

ファイル名 デフォルトの制御コード、英数字、一部の半角文字を使用することができます。

動的フォーマット

1 セットのワードアドレスでファイル名を設定することができ、時間ボタンのコントロールコードを入力することにより、システム時刻を加えることもできます。

注意：キャラクタの長さは 1~25 以内で、半角文字¥/:*?“<>|をサポートしていません。



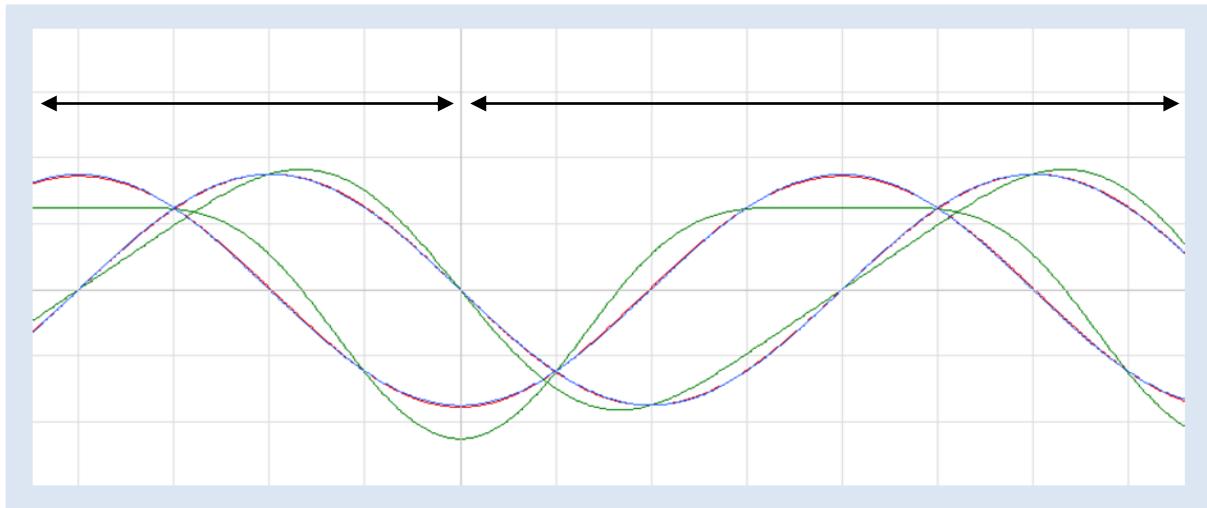
Note

- ファイル名が[動的フォーマット]を使用し、またトリガーモードの[データサンプリング数量制限]を選択する場合、HMI を起動する前、予めに動的フォーマットのソースレジスタにファイル名を書き込む必要があります。さもないと、指定したサンプリング数量に達せないし、サンプリングファイルも作成されません。
- 新規ファイルが作成された場合、システムはまず新規ファイルのファイル名が既に存在しているのかを検知します。既に存在している場合、サンプリングデータがそのファイルに加えられます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

8.4. cMT Viewer のデータを外部装置またはデータベースに同期して保存する



従来、サンプリングしたデータが[トレンド図]オブジェクトで表示する時、リアルタイムモード/履歴モードに設定する必要があり、両モードが1つのオブジェクトに共存することができません。cMT シリーズは、両者を融合し、全てのサンプリングデータが同一の[トレンド図]または[履歴データ表示]オブジェクトで表示するようにし、画面も自動的に更新します。なお、外部記憶装置でデータを同期することもできます。

同期の規則：

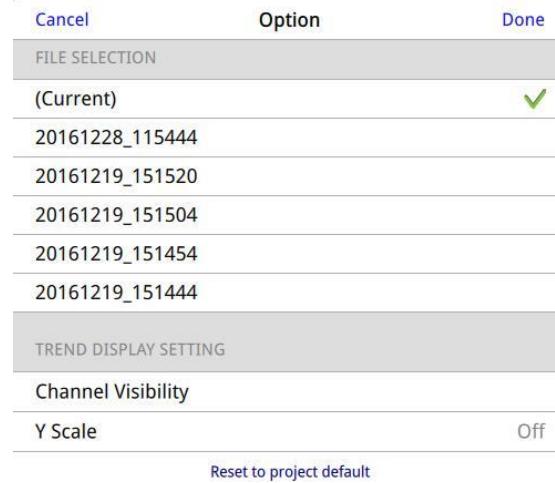
1. サンプリングしたデータが 10000 レコードに達した時、HMI は自動的にデータを外部記憶装置に書き込み、そして HMI 内の最初の 1000 レコードのデータを削除します。
2. もし外部記憶装置を HMI から外した後、新規作成されたサンプリングデータが 9000 レコード未満の際に再び外部装置を接続する場合、外部装置が外された期間、サンプリングデータは依然に HMI に保存され、削除していません。一方、外部装置が外された期間で、サンプリングしたデータが 9000 レコードを超えた場合、より古いデータが削除され、その後で外部記憶装置を再度接続しても、削除されたデータに同期できません。
3. もし外部記憶装置に既にサンプリングデータが存在している場合、毎回同期する時、新しいサンプリングデータは既存のデータに加えられます。

8.5. cMT Viewer の特定の日付またはファイルの履歴データを調べる

履歴データを調べるには、下記手順をご参照ください（トレンド図を例として）：

1. トレンド図の右上隅にある  ボタンをクリックします。

2. トレンド図の設定ダイアログボックスがポップアップされます。



3. 調べたい日付またはファイルを指定します。



4. [完成]をクリックし、設定を完了します。

9. オブジェクトの一般的な属性

本章では、オブジェクトの基本的な設定について説明します。

9.1.	概要	9-2
9.2.	PLC 装置を選択し、読み書きアドレスを設定する	9-2
9.3.	図形ライブラリと画像ライブラリを使用する	9-3
9.4.	テキストラベルの内容を設定する	9-7
9.5.	輪郭を調整する	9-9

9.1. 概要

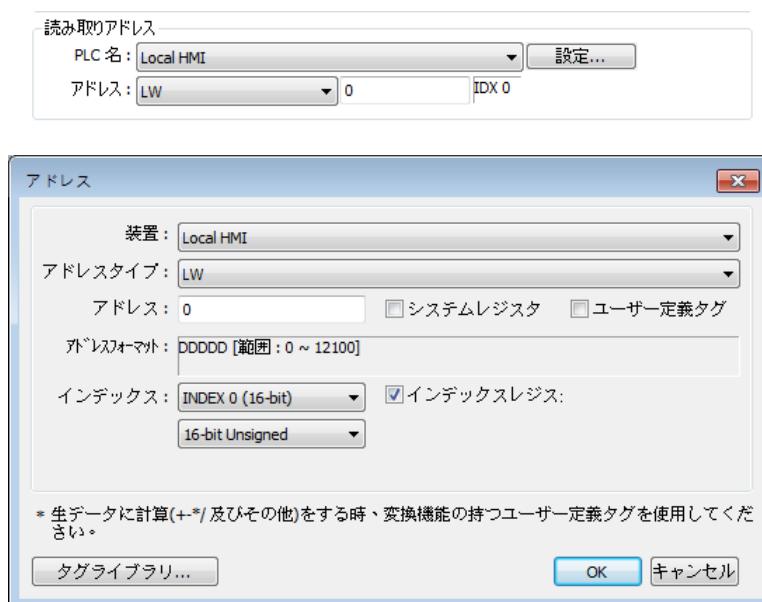
一般の場合、一個のオブジェクトを作成する手順は以下のとおりです：

1. 装置を選択し、読み書きアドレスを設定する。
2. 図形ライブラリまたは画像ライブラリを使用する。
3. テキストラベルの内容を設定する。
4. 輪郭を調整する。

本章は、オブジェクトの一般的な属性の内容について説明します。

9.2. 装置を選択し、読み書きアドレスを設定する

一部のオブジェクトを使用するには、操作対象の装置を選択しなければなりません。[装置]は操作対象である装置のことを指し、これらの装置は[システムパラメータ]内の[装置リスト]の内容に基づいて表示されます。



設定	記述
装置	装置のモデルを選択します。
アドレスタ イプ	装置タイプを選択します。装置モデルが異なる場合、異なる装置タイプが現れます。
アドレス	読み書きアドレスを設定します。
システムレジスタ	アドレスタグには[システムレジスタ]と[ユーザー定義タグ]が含まれています。本項は、[システムレジスタ]を使用するのかを選択します。システムレジスタはシステムが特殊用途として予約済みのアドレスで、bit アドレスレジスタと word アドレスレジスタに分かれています(LB または LW)。[シス

オブジェクトの一般的な属性

ムレジスタ]を選択したら、[アドレスタイプ]に選択したタグを表示する以外、[アドレス]にも選択したタグを表示します。

インデック
スレジスタ

[インデックスレジスタ]を使用するかを選択します。



システムレジスタの詳細情報について、《22 アドレスレジスタ》をご参照ください。



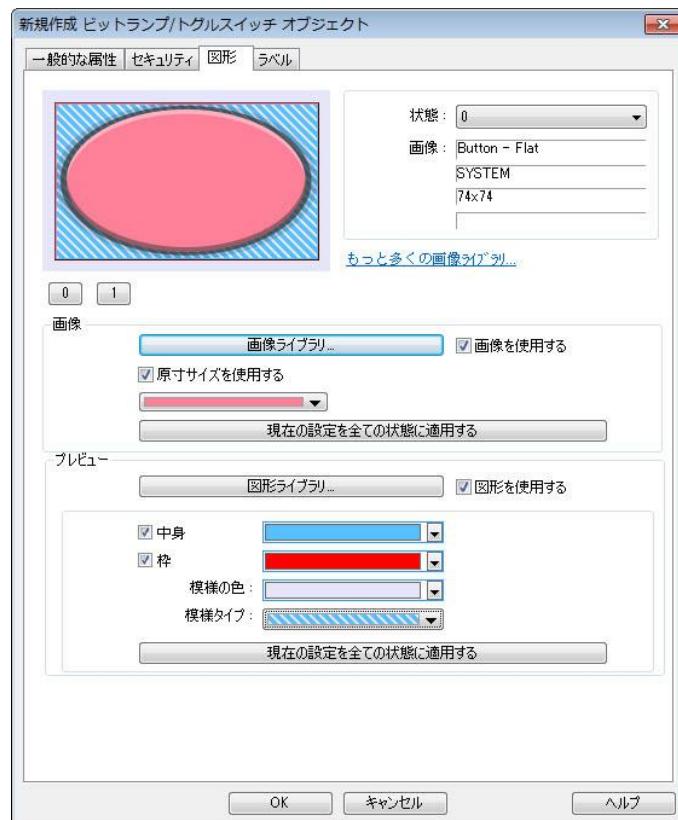
インデックスレジスタの詳細情報について、《11 インデックスレジスタ》をご参照ください。



アドレスタグライブラリの詳細情報について、《16 アドレスタグライブラリ》をご参照ください。

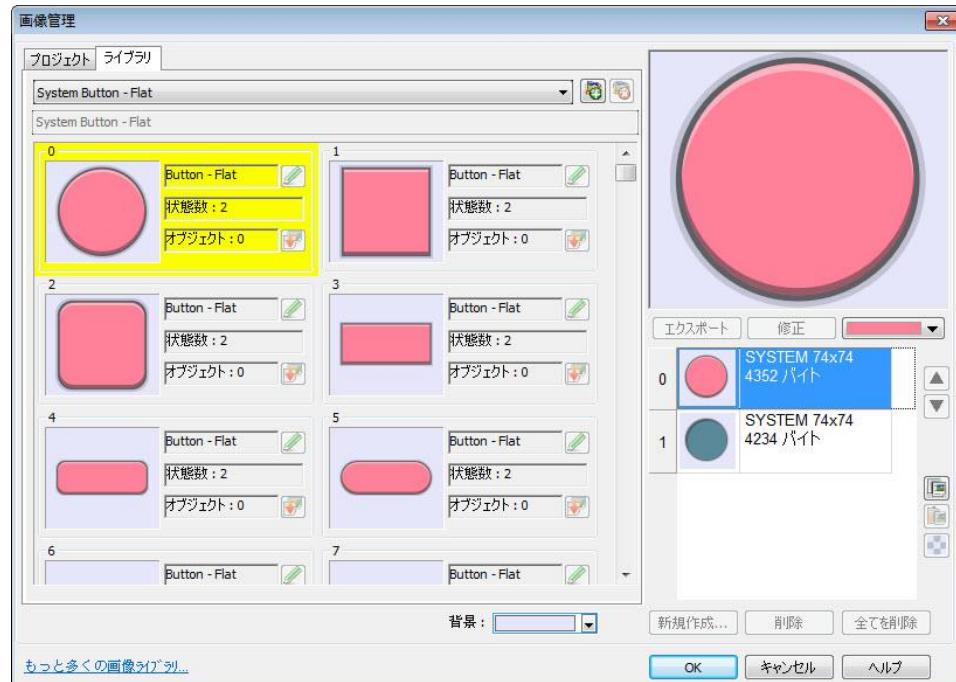
9.3. 図形ライブラリと画像ライブラリを使用する

部分のオブジェクトの設定ダイアログボックスには[図形]タブがあり、[図形ライブラリ]と[画像ライブラリ]の図形を使用し、オブジェクトの視覚効果を豊富にすることができます。図形ライブラリと画像ライブラリの使用は、下図のようにオブジェクトの属性設定タブの[図形]タブで設定します。



9.3.1. 画像

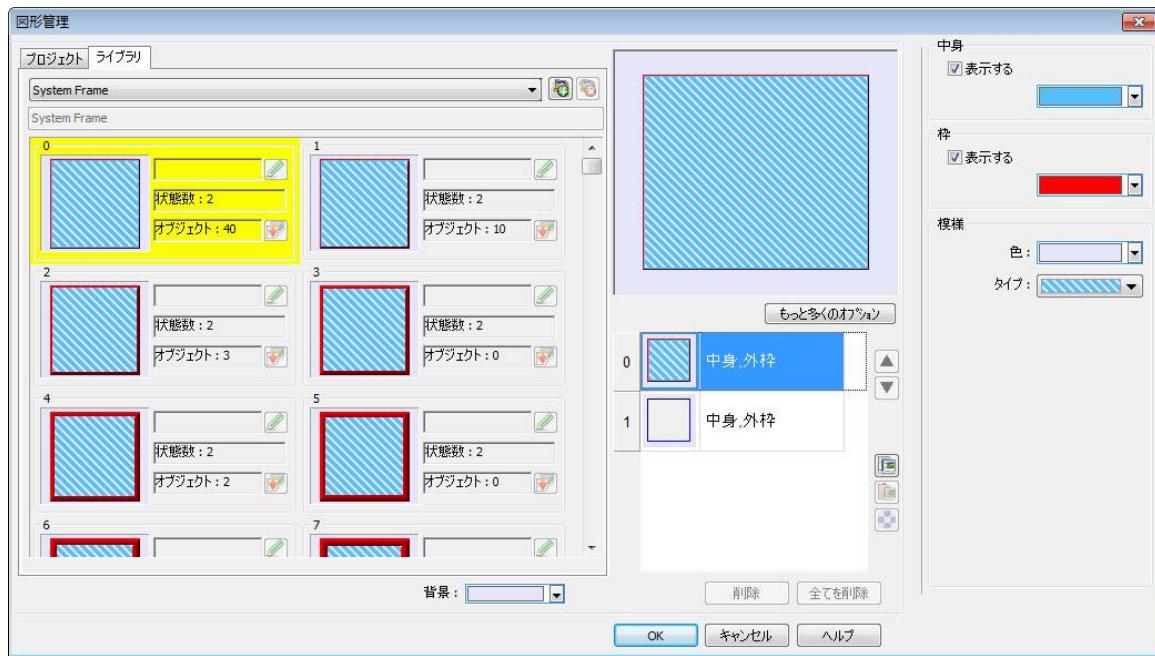
[画像を使用する]にチェックマークを入れ、そして[画像ライブラリ]ボタンをクリックすると、下記のような[画像管理]ダイアログボックスが現れ、その中から画像を選択することができます。



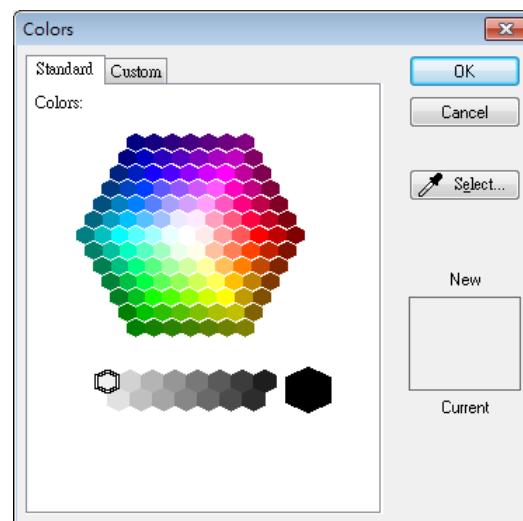
設定	記述
画像ライブラリ	[図形]タブの[図形ライブラリ]ボタンをクリックすると、画像ライブラリ管理ダイアログボックスが現れます。
原寸サイズを使用する	本項にチェックマークを入れると、EasyBuilderは画像を原寸サイズに調整して表示速度を上げます。
現在の設定を全ての状態に適用する	本項はシステム画像を使用する場合のみ使用可能です。[現在の設定を全ての状態に適用する]をクリックした後、現在選択している画像の各設定が他の状態に設置されます。

9.3.2. 図形

[図形を使用する]にチェックマークを入れ、そして[図形ライブラリ]ボタンをクリックすると、下記のような[図形管理]ダイアログボックスが現れ、その中から図形を選択することができます。

**設定****記述****中身**

本項目を選択し、図形の中身の色を選擇します。色指定のドロップダウンボタンをクリックすると、現れた[色の設定]ダイアログボックスで中身の色を設定することができます。

**枠**

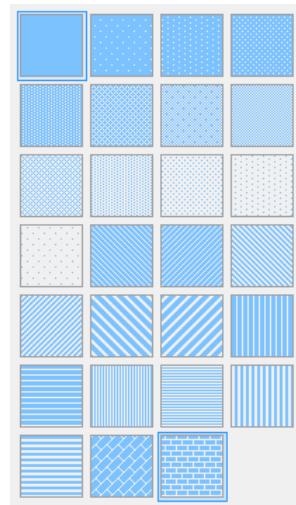
図形の枠を選択します。色指定のドロップダウンボタンをクリックすると、現れた[色の設定]ダイアログボックスで枠の色を選択することができます。

模様色

図形の中身に充填する模様の色を選択します。

模様スタイル

図形の中身に充填する模様を選択します。

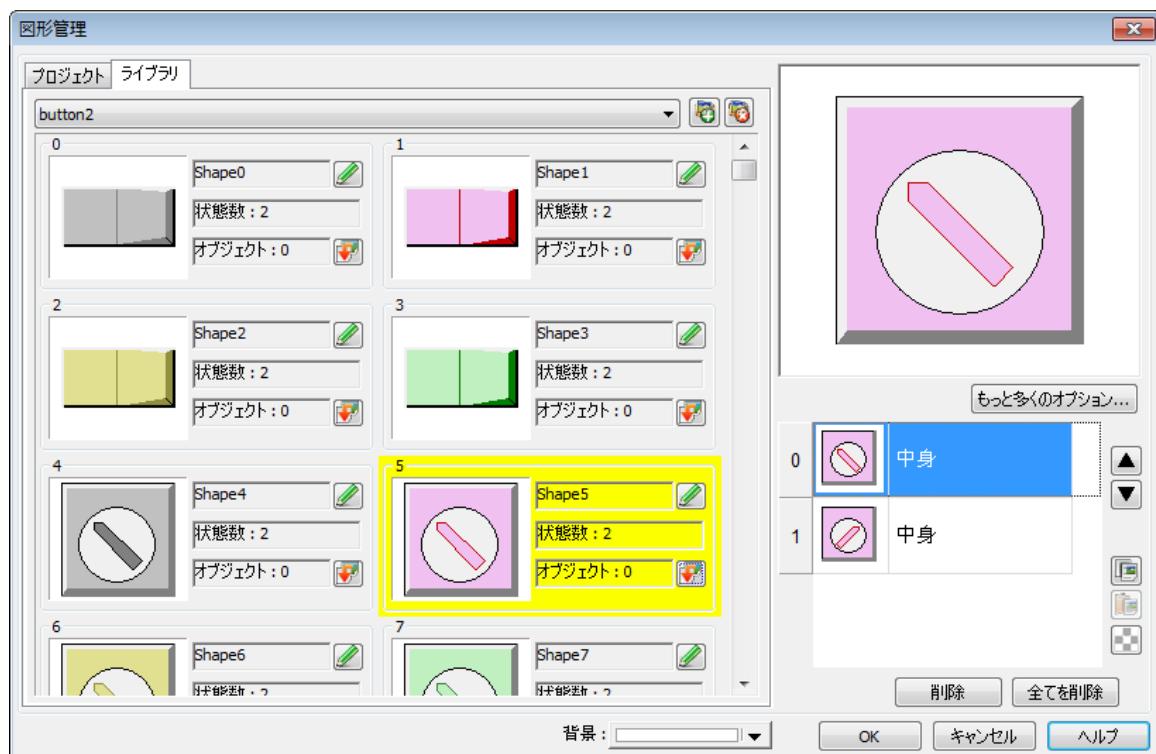
**現在の設定****を全ての状態に適用する**

現在の各設定を他の状態に設置します。

る

9.3.3. 図形管理

[図形ライブラリ]ボタンをクリックすると、下記のような[図形管理]ダイアログボックスが現れます。ダイアログボックスでは、現在選択しているスタイルが黄色ベースに表示されているのが見られます。



上記図面は図形ライブラリ内のあるスタイルの情報を表示しています。これらの情報の意味は

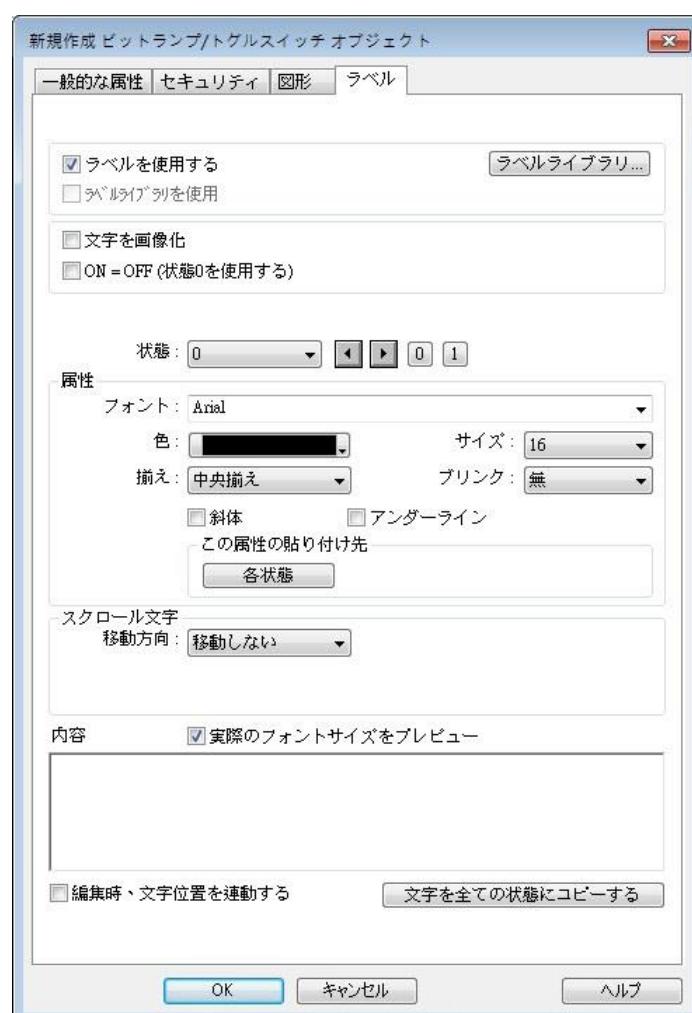
オブジェクトの一般的な属性

以下のとおりです：

Shape5 この図形の名前
 状態数:2 この図形の状態の数
 オブジェクト:1 この図形は1個のオブジェクトに使用されている
 この図形の状態0と状態1は中身だけがあり、枠はありません。

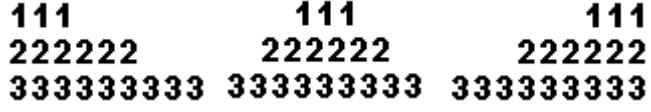
 図形ライブラリと画像ライブラリの詳細情報に関しては《14 図形ライブラリと画像ライブラリの作成》をご参照ください。

9.4. テキストラベルの内容を設定する



設定	記述
ラベルを使用する	本項にチェックマークを入れ選択してから、テキストラベルが使用可能になります。
ラベルライブラリを使用する	本項にチェックマーク入れると、ラベルはラベルライブラリから取得することになります。

オブジェクトの一般的な属性

テキスト画像化	本項にチェックマークを入れると、テキストは bitmap 図形で表示します。
ラベルライブラリ	ラベルライブラリの内容を閲覧します
	 詳細は《15 ラベルライブラリと多言語の使用》をご参照ください。
フォント	使用するフォントを選択します。EasyBuilder Pro は WINDOWS の true-font フォントをサポートしています。
色	文字の色を選択します。
サイズ	文字の大きさを選択します。
揃え	多行文字の配列方式です。ここで設定する以外、下記のショートカットキーを使用して配列することもできます。この配列とは、文字がラベル内での配列のことと、文字がオブジェクト内での配列ではありません。
	 [左揃え]  [中央揃え]  [右揃え]
	
プリント	文字のプリント方式を選択します。プリントしたくない場合は[無]を選択でき、またはプリントする間隔を[1秒]或いは[0.5秒]を選択することができます。
斜体	斜体を使用します。 <i>Italic Label</i>
アンダーライン	文字にアンダーラインをつけます。
	<u>Underline Label</u>
スクロール文字	移動方向 スクロール文字の表示効果と移動方向を選択します。[移動しない]、[左へ]、[右へ]、[上へ]、[下へ]を選択することができます。 継続に移動する 本項にチェックマークを入れていないと、下記図示のように文字が全部消えてから次の文字が現れるようになります。
	

オブジェクトの一般的な属性

本項にチェックマークを入れ選択すると、下記図示のように文字が連続に現れます。



速度

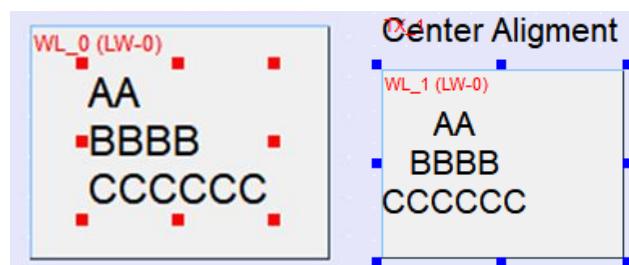
文字の移動速度を選択します。

内容	文字の内容設定します。[ラベルライブラリ]を使用する場合、文字の内容はラベルライブラリから取得します。
編集時、文字位置を連動する	本項にチェックマークを入れると、ある状態の文字を移動すると、他の状態の文字が連動されることになります
文字を全ての状態にコピーする	現在の文字内容を他の状態にコピーします。



Note

- [文字/コメント]オブジェクトで[コメント]機能を使用する場合、サポートする文字内容はコメントとして使用され、HMIでは表示しません。
- ホットキー を使用して配列の属性を調整し、オブジェクト内の文字列(赤枠)を選択する必要があります。オブジェクト(青枠)のみ選択した場合、ホットキーを使用すると、文字列がオブジェクト内での位置を調整することになり、文字列自体の配列属性が変わりません。



- 使用したいフォントが編集画面と HMI で垂直方向の位置ずれが発生した場合、%appdata%\Weintek\EasyBuilder フォルダ内の FontRules.xml ファイルでフォントを調整します。

FontRules.xml にフォント名を追加することにより、事前に特定したフォントに垂直指標(vertical metrics)を施すことができ、位置ずれを防げます。

```
<?xml version="1.0"?>
<FontRules>
    <Version data="1.1"/>
    <Rule data="[\w]*algerian[\.\w]*"/>
    <Rule data="[\w]*castellar[\.\w]*"/>
    <Rule data="^Wingdings$"/>
    <Rule data="fix_vertical_metrics = ^Gotham Pro Black$"/>
    <Rule data="fix_vertical_metrics = ^Gotham Pro Bold$"/>
    <Rule data="fix_vertical_metrics = ^Gotham Pro$"/>
</FontRules>
```

9.5. 輪郭を調整する

オブジェクトを編集ウインドウに置いてから、オブジェクトをクリックすると、[輪郭]設定タブが現れ、オブジェクトのサイズを調整することができます。



設定	記述
位置	ピン留め 本項にチェックマークを入れると、オブジェクトの位置や大きさを変更することができなくなります。 [X]、[Y] この座標はオブジェクトが編集ウインドウ中の位置を表示しています。
寸法	オブジェクトの[幅]と[高さ]を設定します。

10. ユーザーパスワードと オブジェクトセキュリティ

本章では、EasyBuilder Pro でユーザーパスワードとユーザーのセキュリティクラス、及びパスワードの設定について説明します。

10.1.	概要	10-2
10.2.	ユーザーパスワードと操作可能なオブジェクトのクラスの設定	10-2
10.3.	高度セキュリティモードのコントロールアドレス	10-4
10.4.	高度セキュリティモードの応用	10-9
10.5.	オブジェクトセキュリティ設定	10-21
10.6.	オブジェクトセキュリティの例	10-27
10.7.	ユーザーパスワード保護	10-30
10.8.	複数オブジェクトのセキュリティ属性設定を一括変更	10-31

10.1. 概要

EasyBuilder Pro でユーザーのセキュリティクラス及びパスワードを設定するには、以下のモードがあります：

- 一般モード
- 高度セキュリティモード

また、cMT シリーズは LDAP でログインすることを提供します。

オブジェクトセキュリティでは、下記 2 設定が必要です：

1. ユーザーパスワードと操作可能なオブジェクトのクラスの設定
2. オブジェクト操作セキュリティの設定

一個のオブジェクトは 1 クラスのセキュリティのみ所属されます。或いはセキュリティクラスを“無”に設定することで、誰でも当オブジェクトを操作できるようにします。

10.2. ユーザーパスワードと操作可能なオブジェクトのクラスの設定

[システムパラメータ設定] » [ユーザーパスワード] タブでは関連パラメータを設定することができます。

10.2.1. 一般モード

一般モードでは、最大 12 セットのユーザーを設定することができ、それぞれ異なるユーザーパスワードを設定します。パスワードは非負整数である必要があり、各ユーザーが操作できるオブジェクトを“A~F”との 6 クラスに分けます。

HMI が運転する時、ユーザーが正確にパスワードを入力次第、システムは設定した内容に基づきユーザーが操作できるクラスを決めます。下図のように、“ユーザー1”はクラス“A、B、C”的オブジェクトのみ操作できます。



- cMT シリーズはユーザーパスワードでの一般モードをサポートしていません。

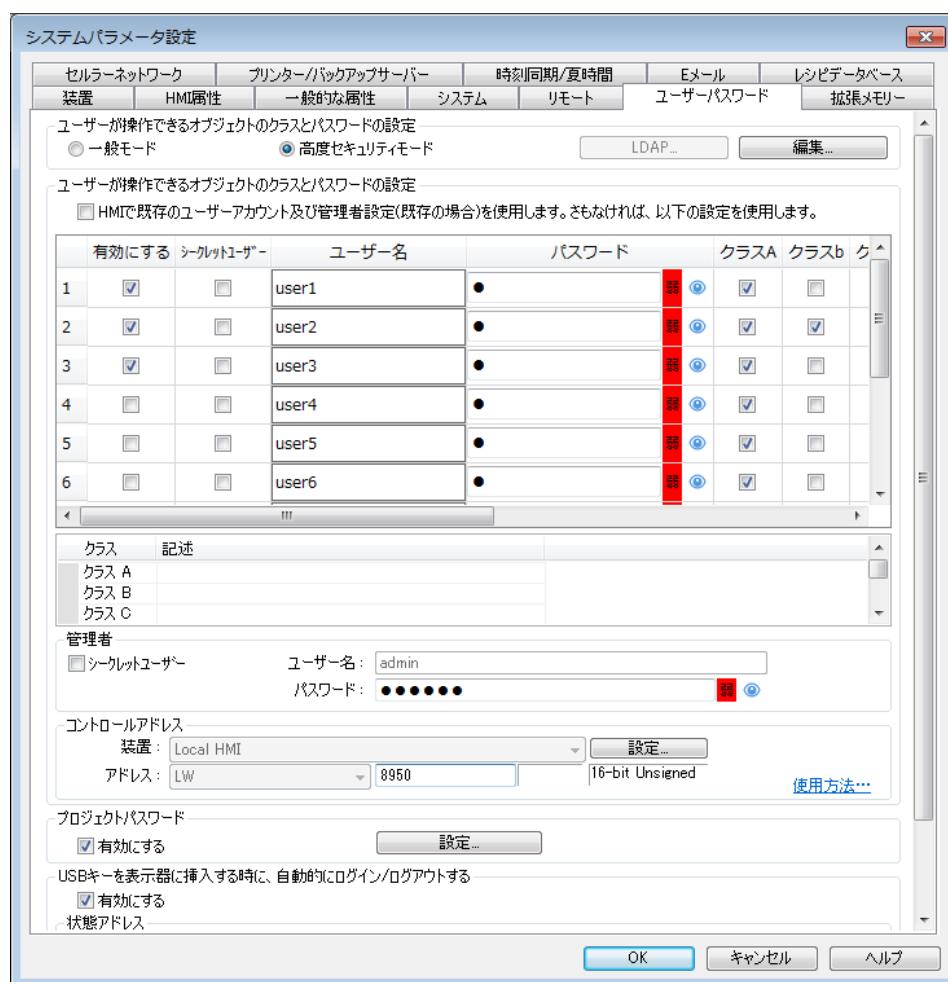


このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

10.2.2. 高度セキュリティモード

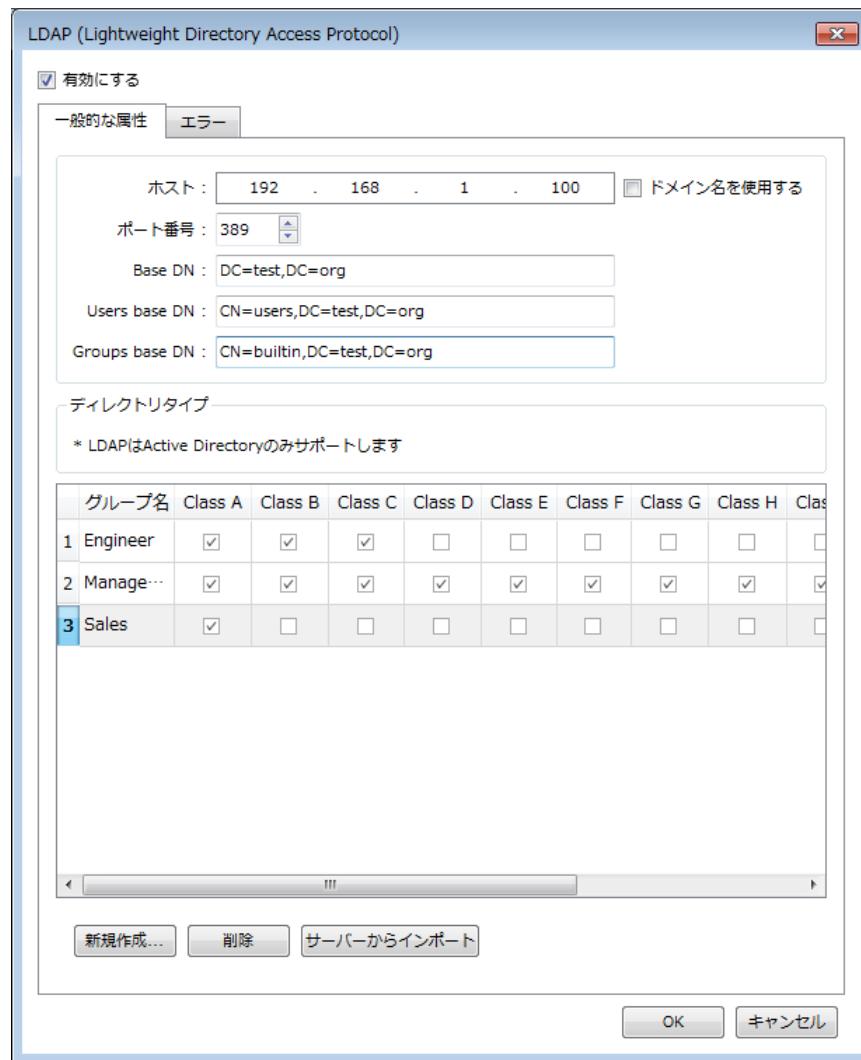
高度セキュリティモードで設定できるユーザー数は 11 セットで、また[管理者]モードを提供しています。管理者は最大な権限を持ち、どのクラスのオブジェクトも操作できます。ユーザー パスワードはアルファベット或いは数字で組み合わせられ、また各ユーザーごとが操作できるオブジェクトを[クラス A]～[クラス L]との 12 クラスに分けられます。(管理者ツールを使用する場合、最大 127 セットのユーザーを設定することができます。本機能は《10.4 高度セキュリティモードの応用》に紹介されます。)

高度セキュリティモードはユーザーにログインとアカウントの管理用で、1 セットの[コントロールアドレス]を提供します。詳細は、《10.3 高度セキュリティモードのコントロールアドレス》での説明をご参考ください。或いは、USB セキュリティキーで自動的にログインすることもできます。接続した USB ディスクには USB セキュリティキーを含んでいる場合、自動的に指定したアカウントにログインします。詳細は、《10.4.3 USB セキュリティキーでログイン/ログアウト》の説明をご参考ください。



10.2.3. LDAP モード

LDAP (Lightweight Directory Access Protocol)はディレクトリサービスに接続するために使用される一種の軽量な通信プロトコルです。ディレクトリサービスは情報を集中保存するようにデータベースに似た仕組みを提供し、主にアカウントの統合、検証に用いられ、ユーザーを集中管理します。LDAP モードを使用する場合、アカウントの管理は HMI(EBPro)側で行う必要がなく、LDAP サーバーで行われ、HMI は LDAP プロトコルを通じてアカウントを検証します。HMI でただディレクトリサーバーの位置、組織単位などの必要情報を設定し、及び各グループが使用できる権限クラスを設定すればいいので、個別のユーザーアカウント・パスワードについて管理する必要がありません。



10.3. 高度セキュリティモードのコントロールアドレス

コントロールアドレスはログインとアカウント管理に適用し、このコントロールアドレスのワードアドレスソースはローカル HMI の LW アドレスのみ使用でき、それに連続 20 個のアドレスをパラメータとして使用します。[コントロールアドレス]にログインする場合、[ユーザー名]もしくは[ユーザーインデックス]でログインする必要があります。

[ユーザー名]及び[パスワード]は、先に[システムパラメータ設定]»[ユーザーパスワード]»[高度セキュリティモード]で設定する必要があります。

10.3.1. コントロールアドレスの使用説明

コントロールアドレスを LW-n に設定した場合(n は任意の数字)、下記アドレスを使用することになります：

コントロールアドレス	ラベル名前	記述
LW-n(1 ワード)	コマンド	各コマンドをコントロールします。(例:ログイン、ログアウト、アカウントを新規作成/変更/削除など)。
LW-n+1(1 ワード)	コマンド実行結果	コマンドを実行した結果を表示します。
LW-n+2(1 ワード)	ユーザーインデックス	アカウントのインデックス(オプションリストオブジェクトを合わせて使用する)。
LW-n+3(1 ワード)	ユーザー権限	権限値(Level A=bit0, Level B=bit1...など)。
LW-n+4(8 ワード)	ユーザー名	アカウント名(英数字を使用可能。大文字・小文字を区別する)。
LW-n+12(8 ワード)	パスワード	アカウントパスワード(英数字、符号を使用可能。大文字・小文字を区別する)。

コントロールアドレスを設定した後、[アドレスラベルライブラリ]»[ユーザー定義タグ]で他の関連機能のアドレスを調べることができます。

例えば、[コントロールアドレス]は LW-0 の場合：

LW-0→[コマンド]
 LW-1→[コマンド実行結果]
 LW-2→[ユーザーインデックス]
 LW-3→[ユーザー権限]
 LW-4~LW-11→[ユーザー名]
 LW-12~LW-20→[パスワード]

Note

- cMT シリーズで、ユーザーパスワードの高度セキュリティモードを使用する場合、コントロールアドレスのワードアドレスソースはローカルHMIのPLWアドレスのみ使用できます。

10.3.2. コマンド機能の説明

[コマンド] -> LW-n に特定[数値]を入力する場合、操作できる機能は以下のとおり：

設定数値	コマンド	対応アドレス
1	ユーザー名でログインする	事前に[ユーザー名]と[パスワード]を定義する必要があり、ユーザー名とパスワードを入力した後、システムはそれが[システムパラメータ設定] » [ユーザーパスワード] » [高度セキュリティモード]での設定に合っているかどうかを確認します。
2	ユーザーインデックスでログインする	事前に[ユーザーインデックス]と[パスワード]を定義する必要があります、《10.4.4 高度セキュリティモードにオプションリストオブジェクトを合わせる》をご参照。
3	ログアウト	
4	ログインしているアカウントのパスワードを変更する	事前に[ユーザー名]と[パスワード]を定義する必要があります、[ユーザー名]に本来のパスワードを入力し、[パスワード]に新しいパスワードを入力する必要があります。
5	アカウントを新規作成する	事前に[ユーザー名]、[パスワード]や[ユーザー権限]を定義する必要があります。
6	臨時アカウントを作成する(單位：分間)	事前に[ユーザー名]、[パスワード]、[ユーザー権限]や[ユーザーインデックス]を定義する必要があります、[ユーザーインデックス]で有効時間(単位:分)を指定します。0は永久的に有効と示しています。
7	ユーザー名で既存アカウントを削除する	事前に[ユーザー名]を定義する必要があります。
8	ユーザーインデックスで既存アカウントを削除する	事前に[ユーザーインデックス]を定義する必要があります。
9	ユーザー名で既存アカウントの権限を設定する	事前に[ユーザー名]と[ユーザー権限]を定義する必要があります。

10	ユーザーインデックスで既存アカウントの権限を設定する	事前に[ユーザーインデックス]と[ユーザー権限]を定義する必要があります。
11	ユーザー名で既存アカウントのパスワードを設定する	事前に[ユーザー名]と[パスワード]を定義する必要があります。
12	ユーザーインデックスで既存アカウントのパスワードを設定する	事前に[ユーザーインデックス]と[パスワード]を定義する必要があります。
13	ユーザー名で既存アカウントの権限を読み取る	事前に[ユーザー名]を定義する必要があり、実行に成功したら、アカウントの権限は[ユーザー権限]に表示されます。
14	ユーザーインデックスで既存アカウントの権限を読み取る	事前に[ユーザーインデックス]を定義する必要があり、実行に成功したら、アカウントの権限は[ユーザー権限]に表示されます。
15	臨時アカウントを新規作成する(単位: 日数)	事前に[ユーザー名]、[パスワード]、[ユーザー権限]及び[ユーザーインデックス]を定義する必要があります、その中、[ユーザーインデックス]は有効日数の指定に用いられます。0は永久に有効と示しています。
16	期間限定アカウントを新規作成する(単位: 分間)	事前に[ユーザー名]、[パスワード]、[ユーザー権限]及び[ユーザーインデックス]を定義する必要があります、その中、[ユーザーインデックス]は有効時間の指定に用いられ、それに0にしてはいけません。
17	期間限定アカウントを新規作成する(単位: 日数)	事前に[ユーザー名]、[パスワード]、[ユーザー権限]及び[ユーザーインデックス]を定義する必要があります、その中、[ユーザーインデックス]は有効日数の指定に用いられ、それに0にしてはいけません。
18	ユーザー名読み取りの残り時間(分間)	事前に[ユーザー名]を定義する必要があります、実行に成功したら、残り時間は[ユーザーインデックス]に表示されます。

19	ユーザーインデックス読み取りの残り時間(分間)	事前に[ユーザーインデックス]を定義する必要があり、実行に成功したら、残り時間は[ユーザーインデックス]に表示されます。
20	ユーザー名読み取りの残り日数	事前に[ユーザー名]を定義する必要があり、実行に成功したら、残り日数は[ユーザーインデックス]に表示されます。
21	ユーザーインデックス読み取りの残り日数	事前に[ユーザーインデックス]を定義する必要があり、実行に成功したら、残り日数は[ユーザーインデックス]に表示されます。

 Note

- 臨時/期間限定アカウントを作成する：臨時アカウント及び期間限定アカウントと一般アカウントとの違いは、臨時/期間限定アカウントはシステムに保存されていないので、電源が切れたら無効になります。それに、臨時/期間限定アカウントは有効期限を越えると、自動的に削除されます。
- 既存アカウントを削除する：現在ログイン中のアカウントを削除してはいけません。
- オフラインシミュレーション/オンラインシミュレーション：プロジェクト内のアカウントで設定します。シミュレーションをする時、ユーザーがアカウントに変更した内容は次回のシミュレーションに残しません。
- admin：デフォルトの管理者アカウントであり、削除不可です。全ての権限を持ち、変更することができません。
- システムレジスタ PLW-10754：現在ログイン中のユーザー名を表示します。(cMT シリーズにのみサポートされています)。
- [ユーザー権限]には現在ログイン中のユーザーの権限を表示できません。現在ログイン中のユーザーの権限を表示させるには、システムレジスタ LW-9222 で表示させなければなりません。
- LDAP モードは[ユーザーインデックス]方式でログインすることに対応しません。

 このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

10.3.3. コマンド実行結果の送信

コマンドを実行したら、システムが自動的に結果コードをコントロールアドレスの LW-n+1 アドレスに送信します。下記の結果コードは 16 進数です：

結果コード	意味
(0x001)	コマンドの実行に成功した
(0x002)	コマンドエラー
(0x004)	アカウントが既存している(アカウントを新規作成する場合に発生する)
(0x008)	アカウントが存在しない
(0x010)	パスワードエラー
(0x020)	現在、コマンドは実行されない
(0x040)	不正なアカウント名
(0x080)	パスワードには無効なキャラクタが含まれている
(0x100)	データのインポートが無効になる
(0x200)	有効期限を超えている(USB セキュリティキーでログインする時、有効期限を管理者ツールに設定することができます)

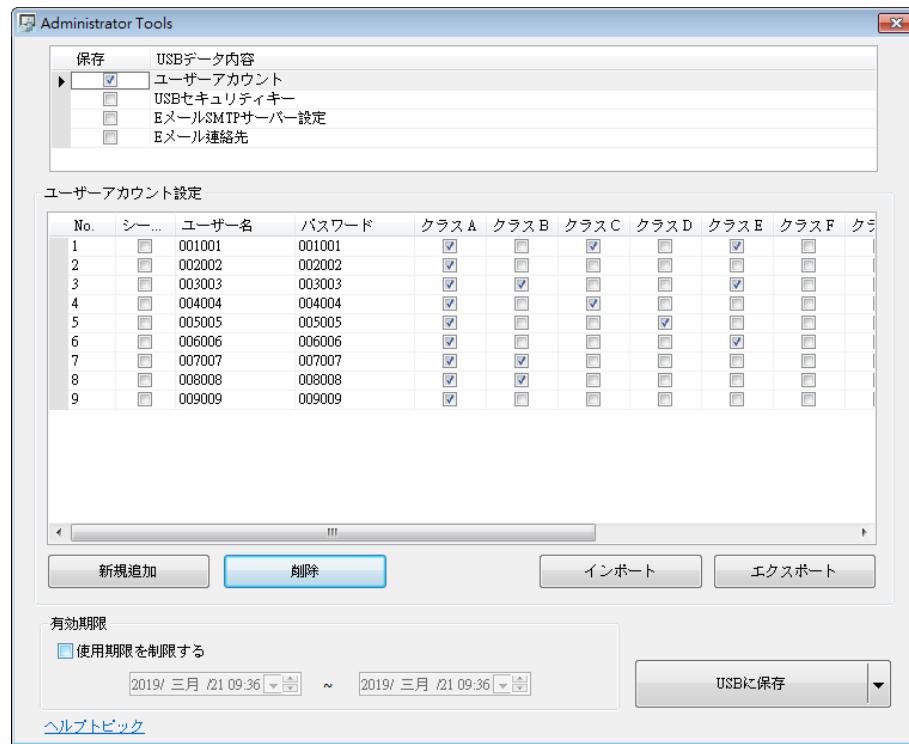
Note

- ユーザーは[コマンド実行結果]LW-n+1 や結果コードの対応意味を事前にイベントログオブジェクトに設定し、そしてイベント表示オブジェクトで直接にコマンドの実行結果を表示することができます。

10.4. 高度セキュリティモードの応用

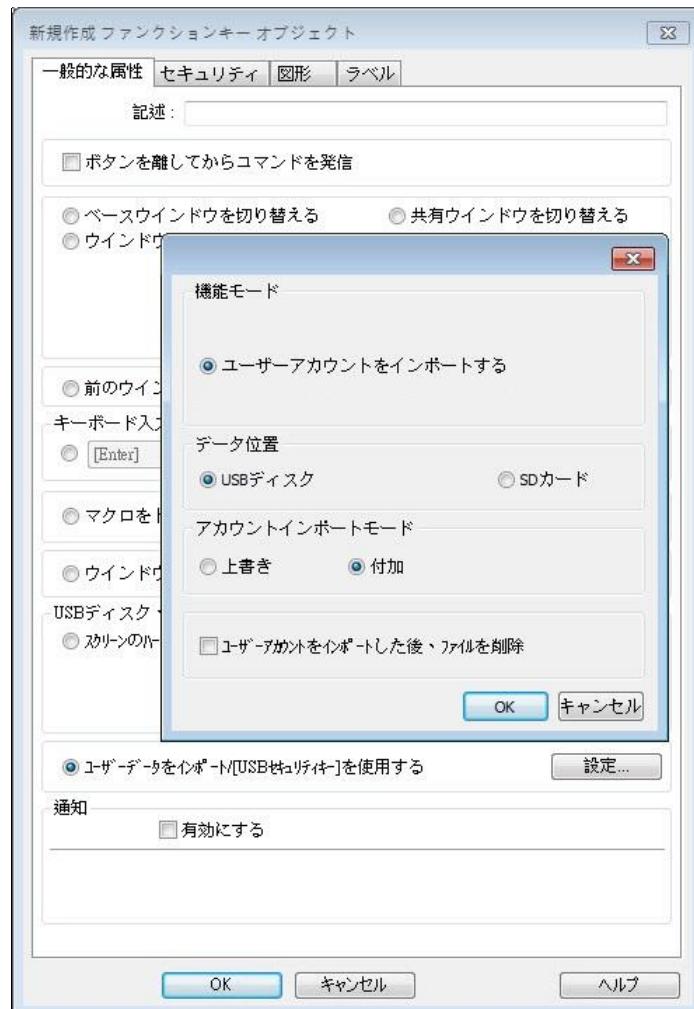
10.4.1. ユーザーアカウントをインポートする

[システムパラメータ設定] » [ユーザーパスワード]タブでユーザーアカウントを設定する方法以外、EasyBuilder Pro インストールディレクトリで管理者ツールをオープンし、[ユーザーアカウント]にチェックマークを入れて選択することで、ユーザーアカウントを設定するのもできます。管理者ツールを使用したら、最大 127 セットのアカウントを作成できます。操作画面は下記のように示します：



☞ 管理者ツールの詳細情報に関して、《36 管理者ツール》をご参照ください。

アカウントの作成を完了した後、USB ディスクまたは SD カードに保存し、それに下図のように予めプログラムで[ファンクションキー] » [ユーザー情報をインポートする]を設定することができます。



設定完了後、外部記憶装置を HMI に挿入すると、本ファンクションキーでユーザー アカウントをインポートすることができます。

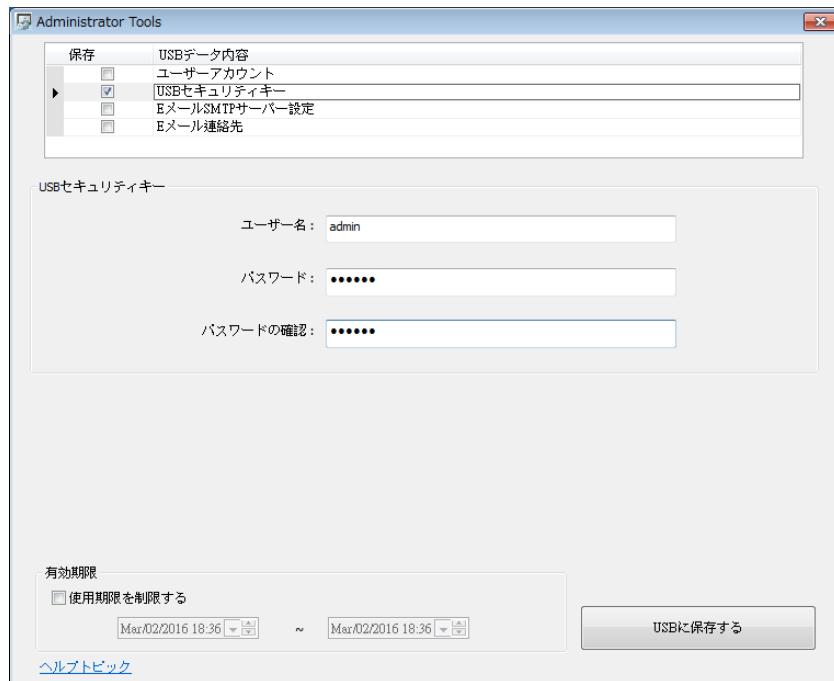
アカウントのインポート方式に[上書き]を選択した場合、インポートする前には、システム内の全てのアカウントがクリアされ、インポート完了後、ログアウトします。

もし[ユーザー アカウントをインポートした後、ファイルを削除]を選択した場合、アカウントをインポートし終えると、システムは自動的に外部装置に保存しているインポートデータを削除します。

もし管理者ツールで[有効期限]を指定した場合、ユーザーは指定した期限内のみインポートできます。インポートしたアカウントは有効期限を超えた原因で削除されることはありません。

10.4.2. USB セキュリティキーでログインする

手動でアカウント及びパスワードを入力することでログインする以外、ユーザーもワンクリックログインでログインできます。EasyBuilder Pro インストールディレクトリで管理者ツールをオープンし、[USB セキュリティキー]にチェックマークを入れると、関連設定を行えます。事前に設定したユーザーのログイン情報を通し、USB セキュリティキーで直接にアカウントにログインすることができます。設定画面は以下のとおりです：

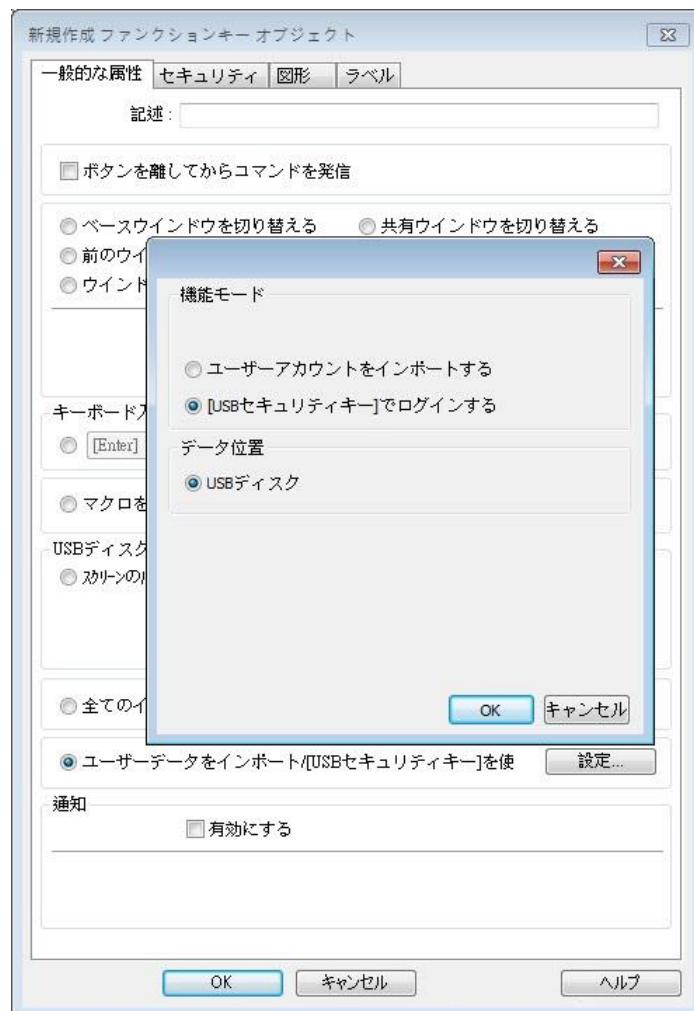


Note

■ USB セキュリティキーを設定したいユーザー アカウントは、既に HMI プログラムに存在しているユーザー アカウントでなければなりません。

☞ 管理者ツールの詳細情報に関しては、《36 管理者ツール》をご参照ください。

USB セキュリティキーの設定を完了した後、USB ディスクまたは SD カードに保存することができます。プログラムで[ファンクションキー] » [USB セキュリティキー]を使用する]を作成します。設定画面は以下のとおりです：



設定完了後、外部記憶装置を HMI に挿入すると、本ファンクションキーでワンクリックで USB セキュリティキーにログインすることができます。

もし管理者ツールで[有効期限]を指定した場合、ユーザーは指定した期限内のみログインできます。また、システムは有効期限を超えた後、自動的にログアウトします。

10.4.3. USB セキュリティキーで自動的にログイン/ログアウトする

下記図示のように、[システムパラメータ設定] » [ユーザーパスワード]タブで[有効にする]にチェックマークを入れると、USB セキュリティキーで自動的にログイン/ログアウトできます：



使用した後、USB セキュリティキーを含めている外部装置を挿入したら、直接にログインできます。装置を外すと、自動的にログアウトします。ログイン/ログアウト状態は自動的に指定し

た[状態アドレス]に書き込みます。表示される数値の意味は：

- 0x00:無動作
- 0x01:ログインに成功した
- 0x04:ログインに失敗した
- 0x08:ログアウトに成功した
- 0x10:ログアウトに失敗した

 USB セキュリティキーを設定する方法について、《36 管理者ツール》をご参照ください。

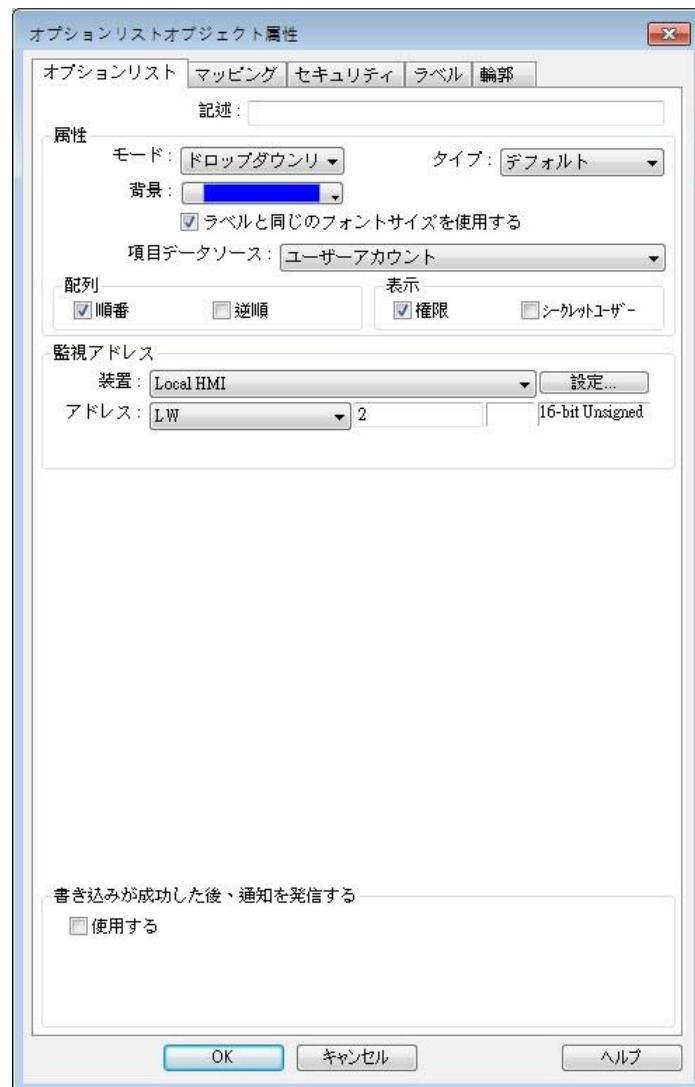
Note

- [USB セキュリティキーで自動的にログイン/ログアウトする]を使用する時に、[ファンクションキー]でログインすることができません。但し依然にコントロールアドレスでログイン/ログアウトできます。
- PC シミュレーションモード下で、本機能は使用できません。
- 本機能は USB ディスクに保存している USB セキュリティキーのみで使用できます。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。本サンプルプロジェクトは USB セキュリティキーで自動的にログイン/ログアウトする方法を説明します。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

10.4.4. 高度セキュリティモードにオプションリストオブジェクトを合わせる

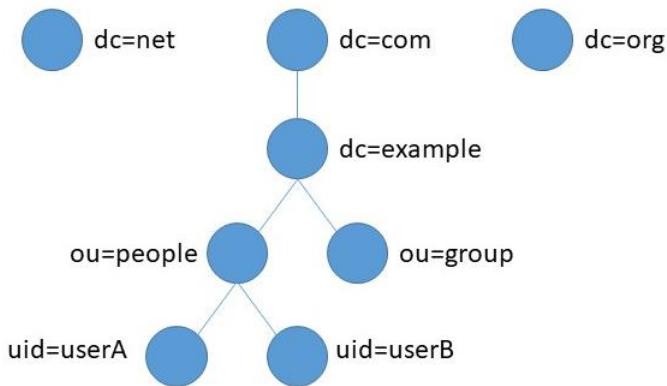
高度セキュリティモードのコントロールアドレス LW-n+2 をユーザーインデックスとして使用するには、オプションリストオブジェクトのユーザー アカウントを合わせてアカウント名と権限を表示することができます。ユーザーはオプションリストオブジェクトでアカウント権限を表示するやユーザーを隠すのかを選択できます。ユーザーを隠すとは、[システムパラメータ設定] » [ユーザーパスワード] » [高度セキュリティモード] でユーザー アカウント名を隠せ、安全性を向上することです。例えば、高度セキュリティモードのコントロールアドレスを LW-0 に設定した場合、監視アドレスは LW-2 となります。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。本サンプルプロジェクトはより一層の高度セキュリティモードの使用方法を説明します。
サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

10.4.5. LDAP モード

LDAP (Lightweight Directory Access Protocol)はディレクトリサービスに接続するために使用される一種の軽量な通信プロトコルです。ディレクトリサービスは情報を集中保存するようにデータベースに似た仕組みを提供し、主にアカウントの統合、検証に用いられ、ユーザーを集中管理します。LDAP モードを使用する場合、アカウントの管理は HMI(EBPro)側で行う必要がなく、LDAP サーバーで行われ、HMI は LDAP プロトコルを通じてアカウントを検証します。HMI でただディレクトリサーバーの位置、組織単位などの必要情報を設定し、及び各グループが使用できる権限クラスを設定すればいいので、個別のユーザー アカウント・パスワードについて管理する必要がありません。



LDAP が使用するコントロールアドレスは高度セキュリティモードと同じです。コントロールアドレスの使用方法については《10.3.高度セキュリティモードのコントロールアドレス》を参照してください。オプションリストオブジェクトが LDAP のユーザー名を取得できないので、[ユーザーインデックス]でアカウントにログインすることに対応しません。

Note

- 複数のグループ内で同じのユーザーが存在する場合、当該ユーザーは各グループの権限を得ます。下図に示されたとおり、仮に Engineer と Sales グループの中に同じくユーザー Angela が存在する場合、Angela は Class A ~ F 権限を得ます。

グループ名	Class A	Class B	Class C	Class D	Class E	Class F
1 Engineer	✓	✓	✓	□	□	□
2 Sales	□	□	□	✓	✓	✓

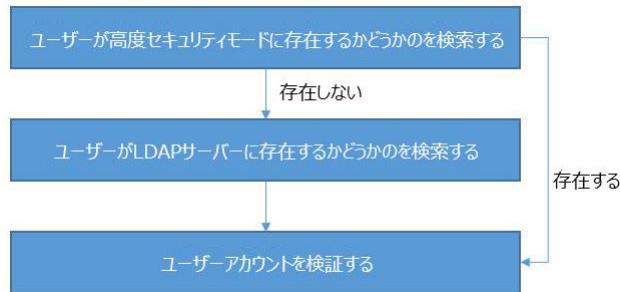
- 高度セキュリティモードのユーザー・アカウント・パスワードは LDAP モードで使用される検証に用いられます。しかしながら、高度セキュリティモード内のユーザー名が同時に LDAP サーバーに存在する場合、ユーザー・アカウントを検証するには、高度セキュリティモード内のアカウントで検証し、LDAP サーバー内のアカウントを使用しません。下表で示されたとおり、ユーザー Angela が同時に LDAP サーバー及び高度セキュリティモードに存在し、この場合、HMI は高度セキュリティモードのアカウントで検証します。

LDAP サーバーユーザー・アカウント

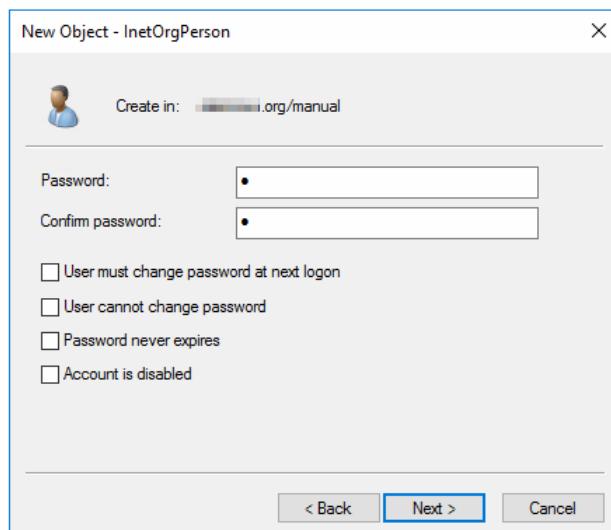
Name	Type
Angela	User
Bella	User
Cindy	User
Dora	User
Elly	User
Fanny	User

高度セキュリティユーザー・アカウント

有効にする	シークレットユーザー	ユーザー名	パスワード	クラスA
1 ✓	□	Angela	●	④ ✓
2 ✓	□	Gina	●	④ ✓
3 ✓	□	Helen	●	④ ✓

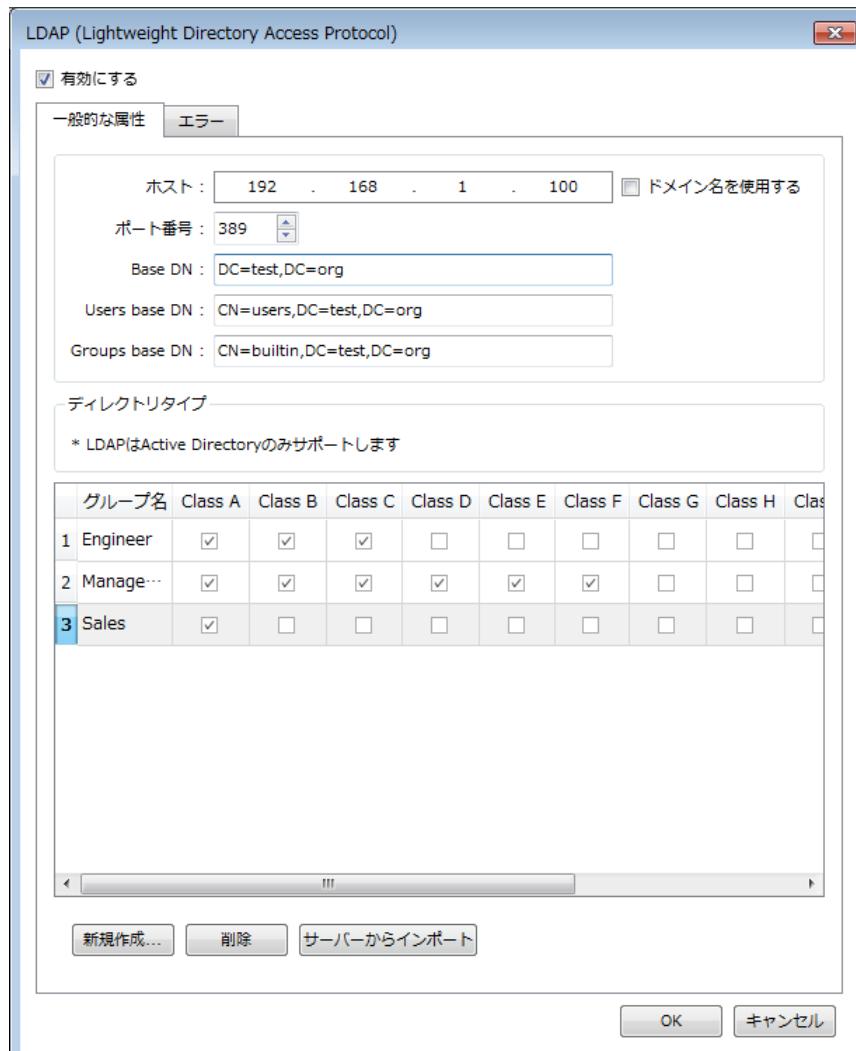


- LDAP モードは[ユーザーインデックス]でログインすることに対応しません。
- LDAP は現在、Active Directory にのみ対応しています。
- 現在、HMI でユーザーのパスワードを変更できないため、LDAP サーバーでユーザーを作成する場合、[User must change password at next logon]でチェックを外してください。



10.4.5.1. 一般的な属性

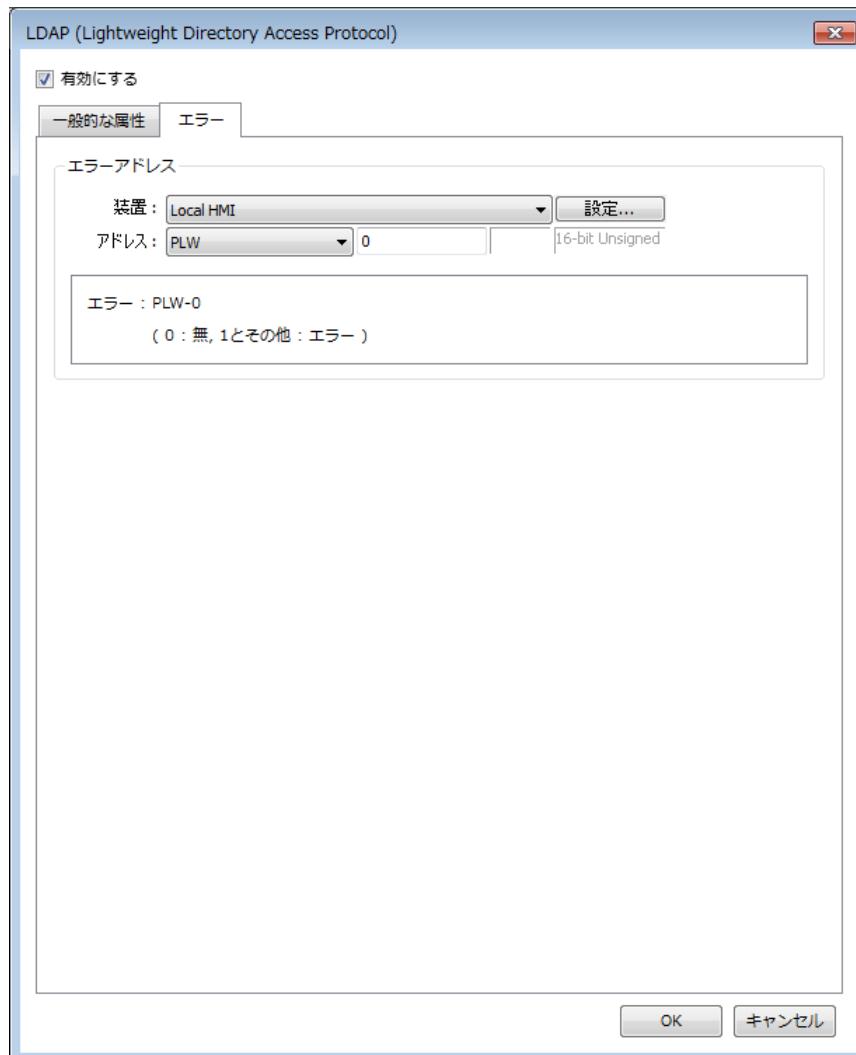
LDAP サーバーの接続情報及び使用可能な権限クラスを設定します。



設定	記述
ローカル	LDAP サーバーのローカル名または IP アドレスです。
ポート	デフォルトは 389 です。
Base DN	LDAP サーバーのドメイン名です(Domain Component, DC)。
Users base DN	ユーザー情報の組織単位(OU)です。
Group base DN	グループ情報の組織単位(OU)です。
新規追加	一個のグループを新規追加します。
削除	一個のグループを削除します。
サーバーから インポート	アカウントパスワードで LDAP サーバーにログインし、全ての接続可能なグループを取得します。
グループ名及 びクラス	各グループが使用できるオブジェクトの権限クラスです。 グループ名の長さは最大 64 ワードまで使用でき、英字(大文字小文字を区別する)/数字/記号/Unicode を使用可能です。

10.4.5.2. エラー

LDAP サーバに接続できない場合、エラーコードを表示するアドレスを設定します。



設定	記述												
エラーアドレス	ログインコマンドの実行結果は本アドレスに出力されます。												
<table border="1"><thead><tr><th>数値</th><th>記述</th></tr></thead><tbody><tr><td>0</td><td>エラー無し</td></tr><tr><td>1</td><td>LDAP サーバ側に異常があり、またはパスワードが入力されていなかった</td></tr><tr><td>2</td><td>未知のエラー</td></tr><tr><td>257</td><td>リモート LDAP サーバに接続できなかった</td></tr><tr><td>258</td><td>ログインアカウントまたはパスワードが正しくなかった</td></tr></tbody></table>		数値	記述	0	エラー無し	1	LDAP サーバ側に異常があり、またはパスワードが入力されていなかった	2	未知のエラー	257	リモート LDAP サーバに接続できなかった	258	ログインアカウントまたはパスワードが正しくなかった
数値	記述												
0	エラー無し												
1	LDAP サーバ側に異常があり、またはパスワードが入力されていなかった												
2	未知のエラー												
257	リモート LDAP サーバに接続できなかった												
258	ログインアカウントまたはパスワードが正しくなかった												

10.4.5.3. LDAP([サーバーからインポート]設定タブ)

グループ情報を LDAP サーバーから取得します。



設定	記述								
ユーザー名	LDAP サーバーにログインするためのユーザー アカウントです。								
パスワード	LDAP サーバーにログインするためのパスワードです。								
全てのグループを取得する	LDAP サーバー内の指定した DN の全てのグループを取得します。								
	<table border="1"> <thead> <tr> <th>数値</th> <th>記述</th> </tr> </thead> <tbody> <tr> <td>Can't contact LDAP server</td> <td>LDAP サーバーに接続できなかった</td> </tr> <tr> <td>Invalid Credentials</td> <td>LDAP サーバーへのログインアカウントまたはパスワードが正しくなかった</td> </tr> <tr> <td>Unknown</td> <td>LDAP サーバーが異常、またはパスワードが入力されなかった</td> </tr> </tbody> </table>	数値	記述	Can't contact LDAP server	LDAP サーバーに接続できなかった	Invalid Credentials	LDAP サーバーへのログインアカウントまたはパスワードが正しくなかった	Unknown	LDAP サーバーが異常、またはパスワードが入力されなかった
数値	記述								
Can't contact LDAP server	LDAP サーバーに接続できなかった								
Invalid Credentials	LDAP サーバーへのログインアカウントまたはパスワードが正しくなかった								
Unknown	LDAP サーバーが異常、またはパスワードが入力されなかった								

Note

- 最大 128 個のグループまで使用できます。インポートする時、システムは LDAP サーバーのグループ数を確認し、128 個を超えた場合、インポートは失敗します。
- インポートされたグループの名前が重複した場合、当該グループの元の権限クラスは消去されません。

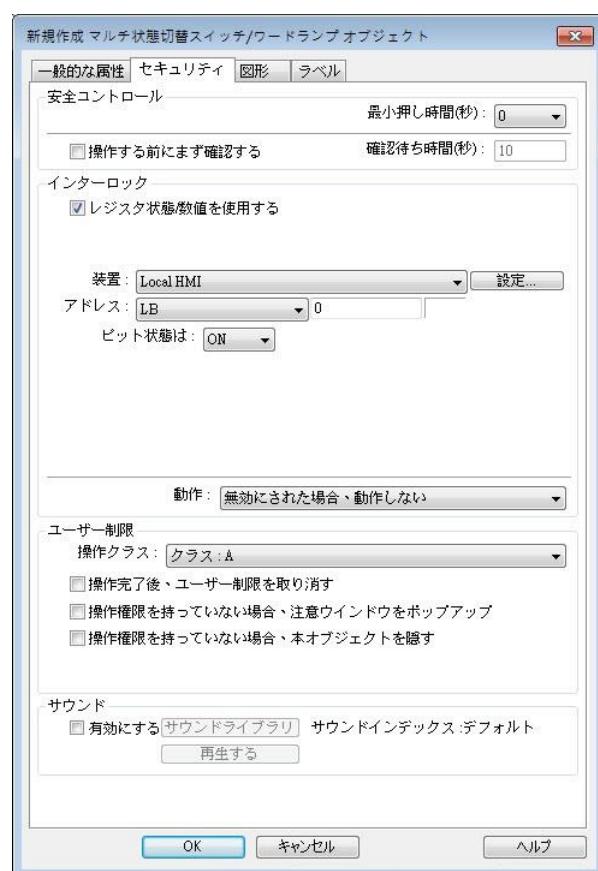
10.5. オブジェクトセキュリティ設定

オブジェクトの[セキュリティ]設定タブでの設定により、特定の条件が満たされた場合のみ、オブジェクトを操作できるように制限することができます。また、本設定タブでオブジェクトの操作音を設定できます。

cMT シリーズ

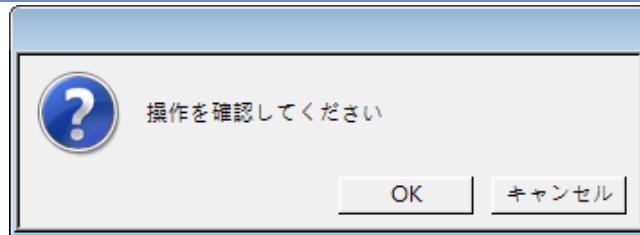


eMT、iE、XE、mTV シリーズ



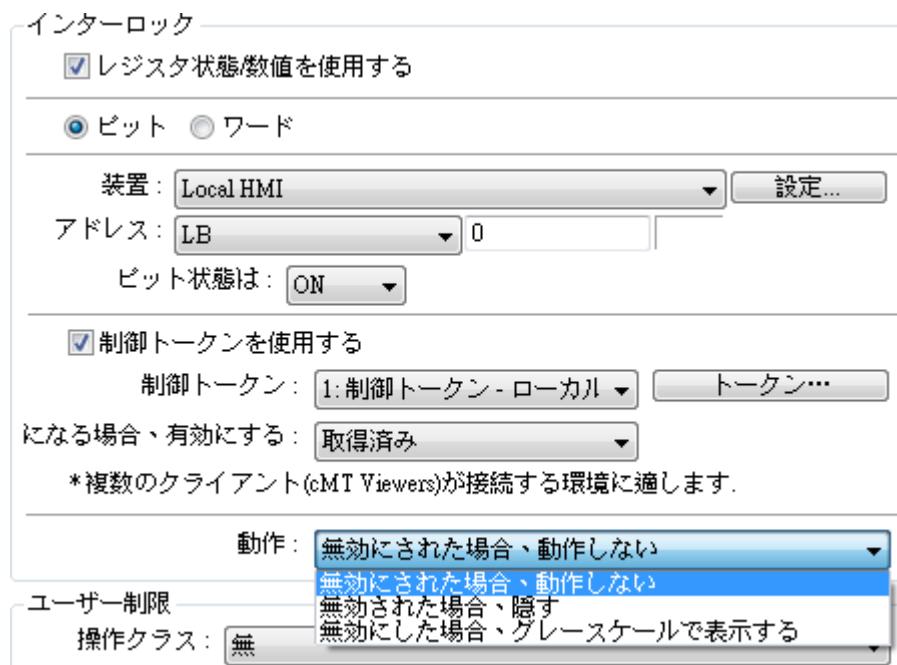
10.5.1. 安全コントロール

設定	記述
最小押し時間	継続的にオブジェクトを押す時間がこの設定値を超えてから、操作できるようになります。
操作する前に まず確認する	一個のオブジェクトを押した後、確認ダイアログボックスが現れ、OKをクリックして操作するかを確認します。もし確認の待ち時間が[確認待ち時間]を超えたたら、ダイアログボックスは自動的に消え、操作を取り消します。



10.5.2. 有効/無効にする

本項目にチェックマークを入れると、このオブジェクトを操作できるのかはある指定ビットアドレスの状態によります。下図に示されたように、LB-0 の状態が ON それに“2: Control Token”を取得した場合のみ、このオブジェクトを操作できます。



下表では、トークンが取得されていない時にオブジェクトが実行するアクションを示しています。

設定	記述
無効にされた場合、動作しない	オブジェクトが操作されない場合でも、依然にオブジェクトを表示します。
無効された場合、隠す	オブジェクトが操作されない場合、隠されます。
無効にした場合、グレースケールで表示する	オブジェクトが操作されない場合、ラベル文字がグレースケールで表示します。

[ワード設定]、[数値]オブジェクトの[インターロック]ビットアドレスの設定に関して、13.4 及び 13.9 をご参照ください。

10.5.2.1. レジスタの状態/数値を使用する

オブジェクトはビット/ワードの数値により操作されるかを決めます。

設定	記述
ビット	指定のビットアドレスが On/Off 状態にいる場合のみ、オブジェクトが操作されます。
ワード	[レジスタの状態/数値を使用する]にチェックマークを入れ、それに[ワード]を選択すれば、本オブジェクトが操作されるかどうかは、指定のワードアドレスの[状態]によります。 状態 指定のワードアドレスの条件を設定できます。>、<、==、<>、>= または <= が選べます。その中、== と <> には[許容誤差]を設定できます。 例：



指定のワードアドレスの数値が 11 より大きいか等しい、または 9 より小さいか等しいの場合、オブジェクトが操作されません。

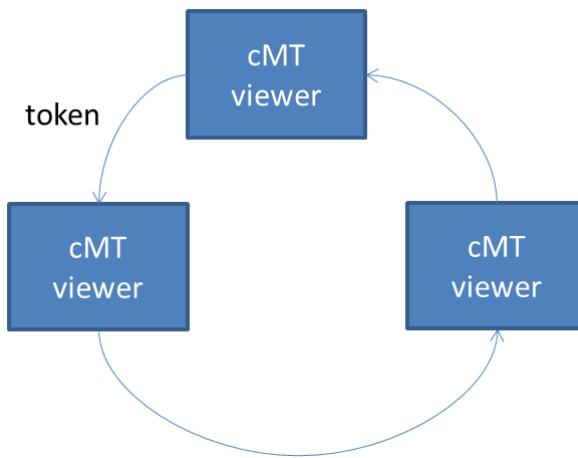
Note

現在はワード設定、数値オブジェクトだけがワードアドレスの状態により有効/無効にする機能を行うことができます。

Note

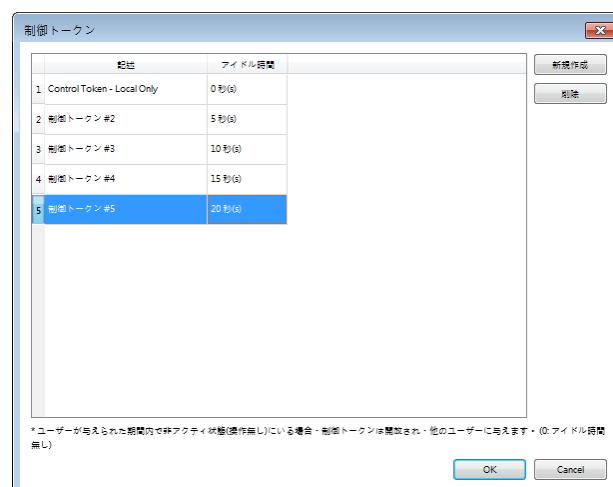
- cMT シリーズに[ワード]をサポートするオブジェクトはワード設定、数値、アスキー、複合ボタンオブジェクトがあります。iE/XE/eMT/mTV シリーズに[ワード]をサポートするオブジェクトはワード設定、数値オブジェクトがあります。

10.5.2.2. 制御トークン(cMT シリーズ)



cMT シリーズでは同時に複数の cMT Viewer の操作が許されるので、同一時間で一台以上の装置が同じのオブジェクトを操作することで作業の流れを影響することを避けるため、制御トークンを使用してオブジェクトを保護できます。制御トークンを使用する時、同一時間では一台の装置のみ制御トークンを取得でき、他の装置は順序によって制御トークンの取得を待つ必要があります。

設定	記述
制御トークン	制御トークンを設定します。
制御トークン...	制御トークンのを新規追加/削除できます。
制御トークン - ローカル専用	本トークンを使用する場合、cMT Viewer を内蔵したローカル側のみオブジェクトを操作できます。(例えば : cMT3072、cMT3090、cMT3151、cMT-HDM...)



アイドル時間

一人のユーザーが制御トークンを取得し操作し終えた後、

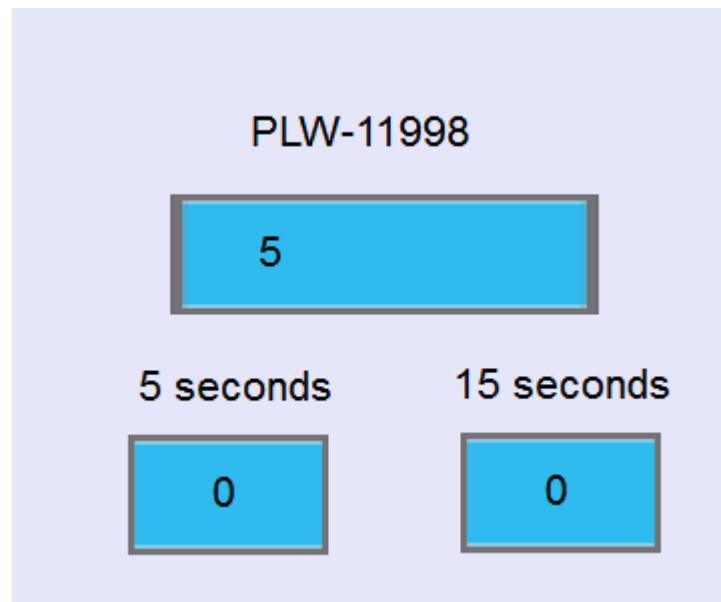
制御トークンを自動的に解放しなくて他のユーザーが操作できなくなるようなことを避けるため、[アイドル時間]機能を利用できます。ユーザーが一個の制御トークンを取得した後、指定した時間内で操作していない場合、システムはこの制御トークンを解放します。アイドル時間の範囲は 0 ~ 86400 秒です。制御トークン - ローカル専用はアイドル時間が無く、固定に 0 に設定されます。

トークンが次の状態になる場合、有効にする

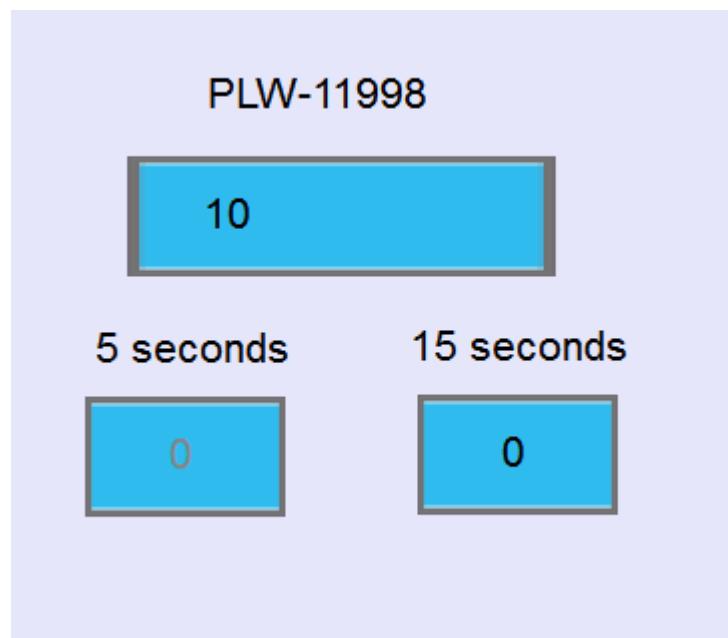
[取得された]を使用する場合、制御トークンを取得した装置がオブジェクトを操作できます。[未取得]を使用する場合、制御トークンを取得していない装置がオブジェクトを操作できます。

Note

- cMT Viewer がウインドウに入る時、そのウインドウ内にある全てのオブジェクトのまだ取得されていない制御トークンを取得します。ウインドウから離れると、制御トークンが解放されます。
- 一台の cMT Viewer には同時に複数の制御トークンを所有することができます。制御トークンの取得機制は先にはいる人が先に取得するのです。それに 1 個の制御トークンは 1 台の cMT Viewer 装置にのみ取得されます。残りの装置は取得しようとした時点によりキューに入れられ、制御トークンの取得を順番待ちにします。
- ウインドウを切り替える場合、切り替えた前後でも同じの制御トークンが設定されたオブジェクトが存在しても、新しいウインドウに入った後、改めて制御トークンを取得する必要があります。
- もし制御トークンがあるオブジェクトを下敷きウインドウまたは共有ウインドウに置き、それにウインドウを切り替えた前後でも同じの下敷きウインドウを使用する場合、ウインドウを切り替えた後、改めて制御トークンを取得する必要がありません。
- cMT Viewer 装置とサーバーとの接続が切断された時、切断された時間が 30 秒以内の場合、制御トークンを改めて取得する必要がありません。切断された時間が 30 秒超えの場合、制御トークンを改めて取得する必要があります。
- 操作していないまま[アイドル時間]を超えた場合、制御トークンが解放され、再度操作するには改めて制御トークンを取得する必要があります。
- アイドル時間の残り時間は[PLW-11998 (8 words): 制御トークン解放カウントダウン]とアスキーオブジェクトと合わせて確認できます。当該アドレスはアイドル時間の残り時間をカウントダウンします。2 個以上の制御トークンがある場合、まずは残り時間が少ないほうからカウントダウンしてから、多いほうのカウントダウンを始めます。以下の例をご参照ください:
アスキーオブジェクトで PLW-11998 を表示し、アイドル時間の残り時間をカウントダウンします。2 個の数値オブジェクトはそれぞれ異なる制御トークンを設定していて、一個のアイドル時間は 5 秒で、もう一個は 15 秒です。シミュレーションを実行する時、PLW-11998 はまず 5 秒のアイドル時間からカウントダウンします。

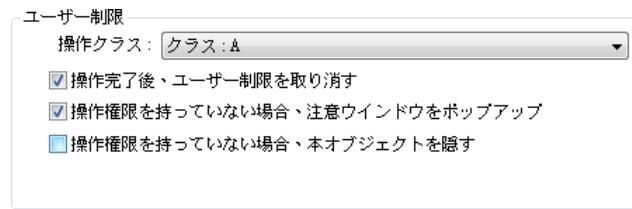


5秒が経った後、一個の制御トークンは解放され、当該オブジェクトはグレーアウトになります。もう一個のアイドル時間が15秒の制御トークンはあと10秒で解放されるため、この時、PLW-11998は10秒からカウントダウンします。



10.5.3. ユーザー制限

オブジェクトの操作クラスを設定します。クラスが一致するユーザーのみ当該オブジェクトを操作できます。

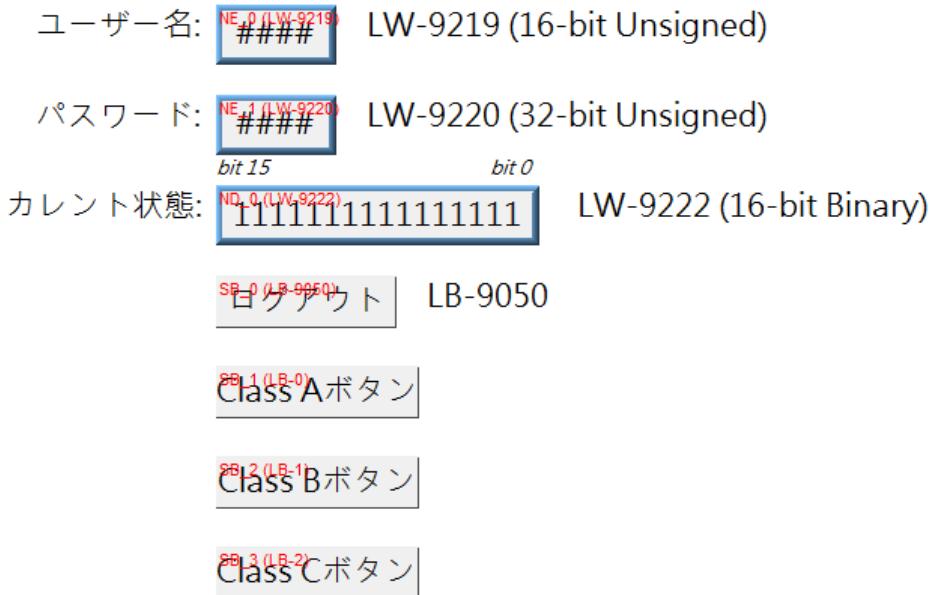


設定	記述
操作クラス	“無”はどのユーザーでも操作できることを示しています。 “管理者”は admin アカウントのみが操作できることを示しています。
操作完了後、ユーザー制限を取り消す	一旦あるユーザーにあるオブジェクトへの操作を許可した場合、システムは二度とそのオブジェクトのセキュリティクラスを確認しません。つまり、そのユーザーが操作し終えたら、他のユーザーでもそのオブジェクトを自由に操作することができるようになります。
操作権限を持っていない場合、注意ウィンドウをポップアップ	ユーザーの操作権限が本オブジェクトのセキュリティクラスに満たしていない場合、注意ウィンドウ 7 番をポップアップします。ユーザーは自分でウィンドウで表示する注意言葉を設定することができます。
操作権限を持っていない場合、本オブジェクトを隠す	ユーザーの操作権限が本オブジェクトのセキュリティクラスに満たしていない場合、オブジェクトが隠されます。

10.6. オブジェクトセキュリティの例

一般モードでのオブジェクトセキュリティの使用例を下記にて挙げます：

- プロジェクトを新規作成し、[システムパラメータ設定] » [ユーザーパスワード] » [一般モード]で三個のユーザーを設定します。例：
 ユーザー1=操作できるオブジェクトはクラス A
 ユーザー2=操作できるオブジェクトはクラス A, B
 ユーザー3=操作できるオブジェクトはクラス A, B, C
- ウインドウ 10 での設計は下図のように示します:



2 個の[数値]オブジェクトを作成する :

[LW-9219]ユーザー番号(1~12)、長さ=1 word

[LW-9220]ユーザーパスワードを入力、長さ=2 words

1 個の[数値表示]オブジェクトを作成する :

[LW-9222]現在ユーザーの状態を表示する。フォーマット : 16-bit Binary

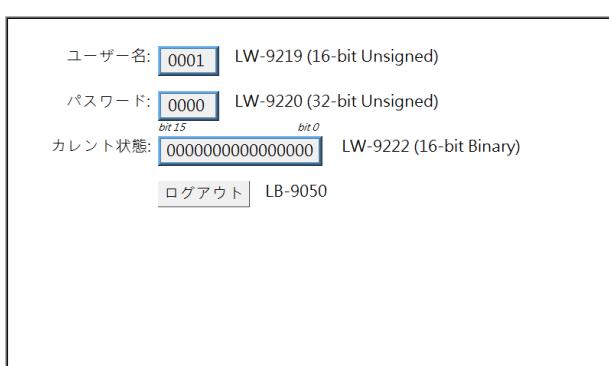
1 個の[ビット状態設定]オブジェクトを作成する :

[LB-9050]ログアウト

3 個の[ビット状態設定]オブジェクトを作成する :

異なるクラスを選択し、全部[操作権限を持っていない場合、本オブジェクトを隠す]を設定する。

- 上記の各設定を完了し、セーブやコンパイルした後、オフラインシミュレーション機能を実行することができます。下図はシミュレーション機能の開始画面です。



ユーザー名:	<input type="text" value="0001"/>	LW-9219 (16-bit Unsigned)
パスワード:	<input type="text" value="0111"/>	LW-9220 (32-bit Unsigned)
カレント状態:	<input type="text" value="0000000000000001"/> bit 15 bit 0	LW-9222 (16-bit Binary)
<input type="button" value="ログアウト"/> LB-9050		
<input type="button" value="Class Aボタン"/>		

“ユーザー1”のパスワード(111)を入力する：
“ユーザー1”がクラスAのオブジェクトを操作できると設定したので、[Class A ボタン]オブジェクトが現れ、操作が許可されます。[LW-9222]の bit0 が“1”になると、この時ユーザーがクラス“A”的オブジェクトのみ操作できると示します。

ユーザー名:	<input type="text" value="0003"/>	LW-9219 (16-bit Unsigned)
パスワード:	<input type="text" value="0333"/>	LW-9220 (32-bit Unsigned)
カレント状態:	<input type="text" value="0000000000000111"/> bit 15 bit 0	LW-9222 (16-bit Binary)
<input type="button" value="ログアウト"/> LB-9050		
<input type="button" value="Class Aボタン"/>		
<input type="button" value="Class Bボタン"/>		
<input type="button" value="Class Cボタン"/>		

“ユーザー3”的パスワード(333)を入力：
“ユーザー3”がクラスA, B, Cのオブジェクトを操作できると設定したので、[LW-9222]の bit0~bit2 の数値が全部“1”になり、この時ユーザーがクラス“A~C”的オブジェクトを操作できると示します。

ユーザー名:	<input type="text" value="0003"/>	LW-9219 (16-bit Unsigned)
パスワード:	<input type="text" value="0000"/>	LW-9220 (32-bit Unsigned)
カレント状態:	<input type="text" value="0000000000000000"/> bit 15 bit 0	LW-9222 (16-bit Binary)
<input type="button" value="ログアウト"/> LB-9050		

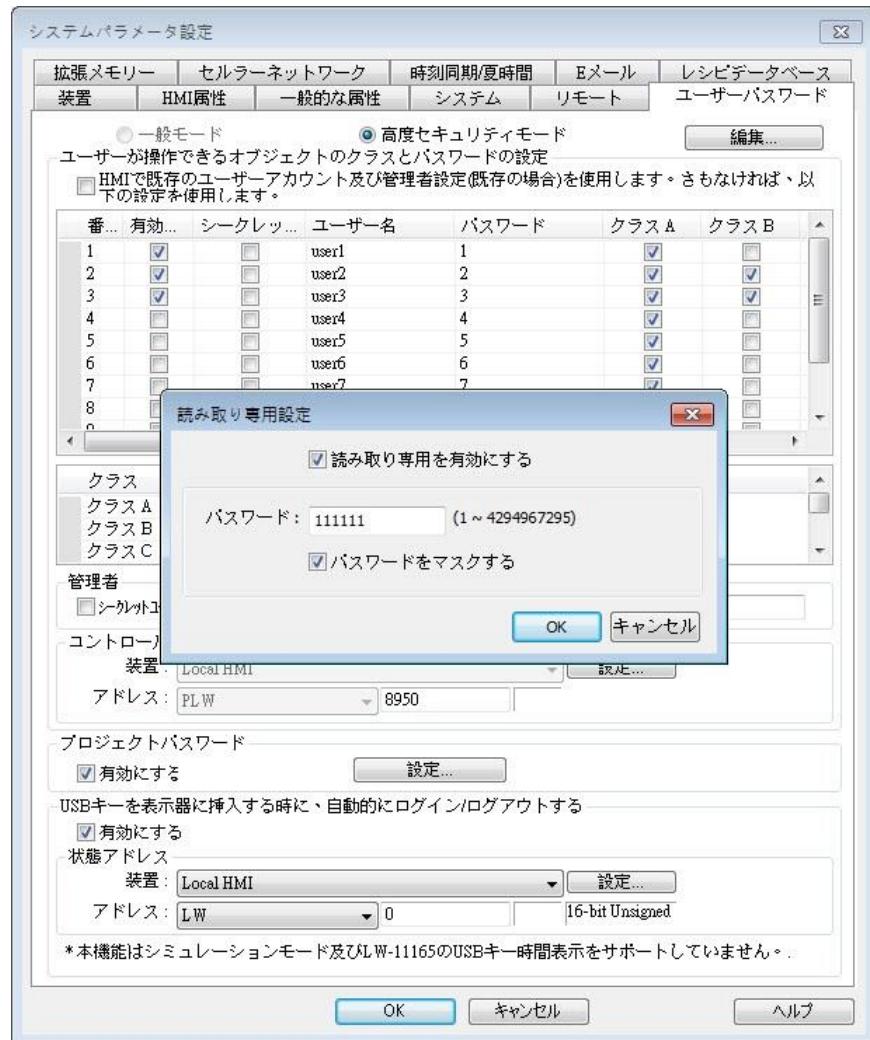
この時、[ログアウト]を押して強制的にログアウトすれば、システムは初期状態に戻り、操作クラスを“無”に設定したオブジェクトのみ操作できるようになります。

Note

- パスワード入力：パスワードの入力が間違った場合、[LB-9060]の状態はON状態に設定されます；パスワードの入力に成功した場合、[LB-9060]の状態が自動的にOFF状態に戻されます。全てのユーザー(ユーザー1~ユーザー12)のパスワードは、システムレジスタ[LW-9500]~[LW-9522]合計24wordsの内容を読み取って取得することができます。
- オンラインでパスワードを変更する：[LB-9061]の状態をONに設定した時、システムは[LW-9500]~[LW-9522]内の数値を読み取り、ユーザーのパスワードを更新します。今後は新規パスワードを使用することになります。この際にユーザーが操作できるオブジェクトのクラスは、パスワード変更により変更することはありません。

10.7. ユーザーパスワード保護

プロジェクトを他人に編集させる必要があり、ただしユーザーパスワードを知らせたくない場合、[編集]ボタンを押し、ユーザーパスワード設定部を読み取り専用にすることができます。



[読み取り専用を有効にする]を選択したら、ユーザーパスワードを変更するには、パスワードを入力する必要があります。

[パスワードをマスクする]を選択したら、パスワードは * 印で表示されます。

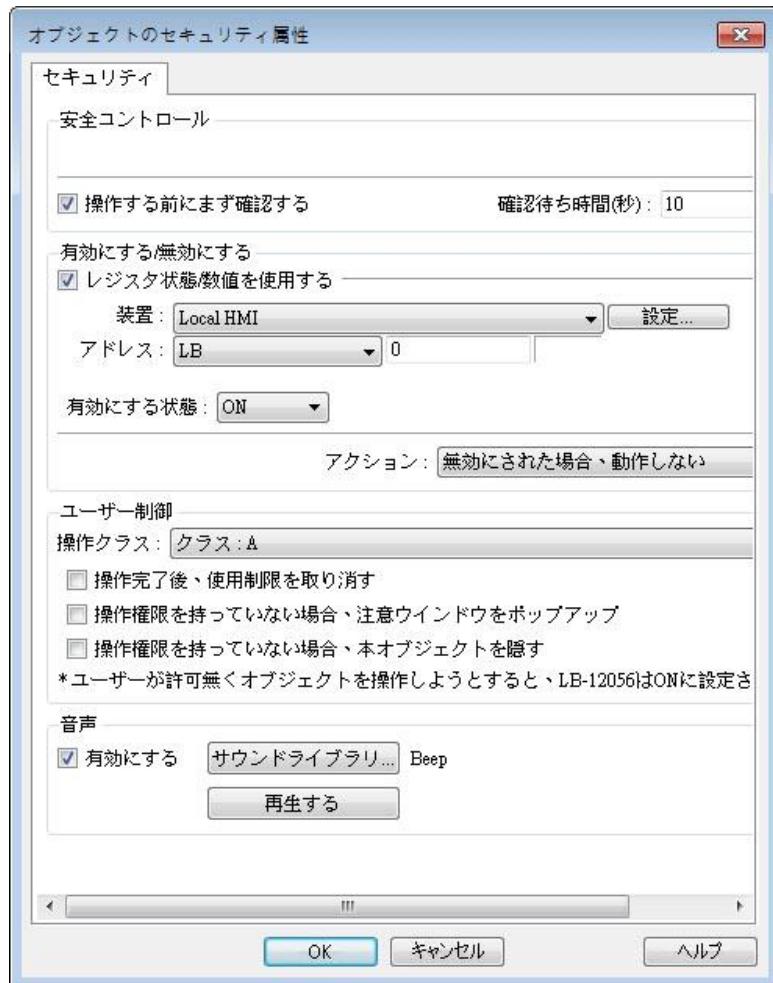


Note

- パスワードを忘れた場合、プロジェクトが既に暗号化されたので、たとえメーカーでも解読できません。パスワードを大切に保管してください。

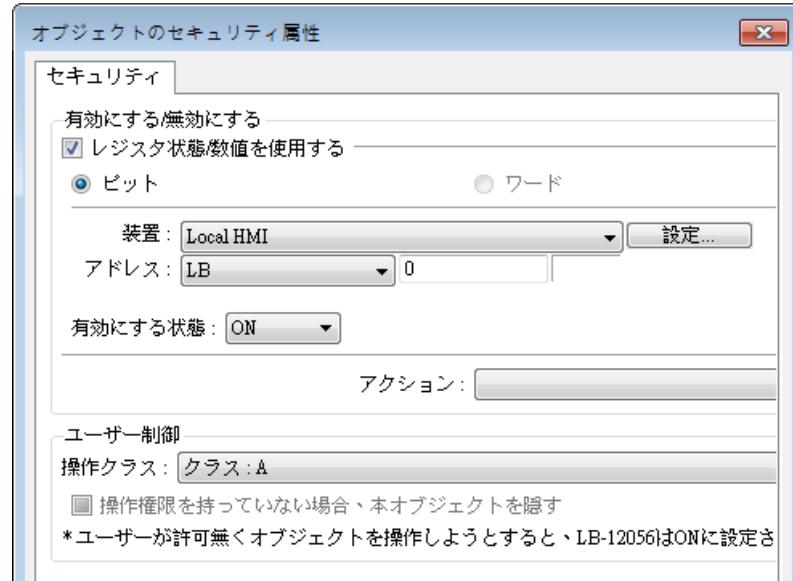
10.8. 複数オブジェクトのセキュリティ属性設定を一括変更

複数のオブジェクトを選択してから右クリックしてセキュリティ設定を選択すると、オブジェクトのセキュリティ属性設定ウインドウに入ることができます。単一のオブジェクトの設定ウインドウと全く同じで、複数のオブジェクトのセキュリティ属性を一括で変更することができます。



Note

- 選択したオブジェクトには異なるセキュリティ属性のインターフェースがある場合、一括変更を行う時に、セキュリティ設定インターフェースが少し変わります。下図を例に挙げると、同時にビットランプと数値オブジェクトを選択した場合、セキュリティインターフェースには設定可能な部分だけが現れ、他の部分は隠され、またはグレーアウトされます。



11. インデックスレジスタ

本章では、インデックスレジスタの使用方法について説明します。

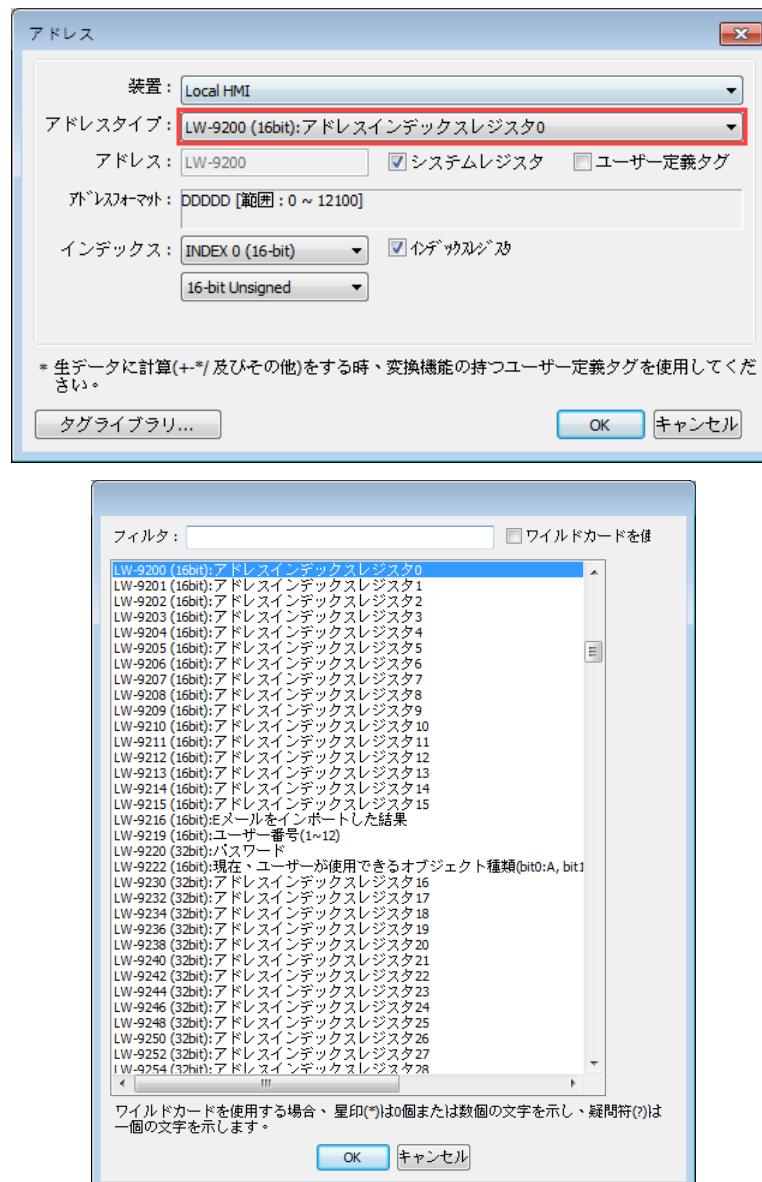
11.1. 概要	11-2
11.2. インデックスレジスタの使用例	11-4

11.1. 概要

インデックスレジスタは EasyBuilder Pro がアドレスの変更用に提供するレジスタです。

インデックスレジスタがあると、ユーザーはオブジェクトアドレスを変更しないまま、HMI で直接にオブジェクトの読み書きアドレスを変更することができます。

EasyBuilder Pro は 32 セットのインデックスレジスタを提供しており、16 セットの 16-bit インデックスレジスタと 16 セットの 32-bit インデックスレジスタがあります。



16-bit アドレスインデックスレジスタ 0~15 に対応したアドレスは LW-9200(16bit)~LW-9215(16bit) で、その最大オフセット量は 65536 words です。

32-bit アドレスインデックスレジスタ 16~31 に対応したアドレスは

LW-9230(32bit)~LW-9260(32bit) で、その最大オフセット量は 4294967296 words です。

[インデックスレジスタ]を使用した後、使用する[装置タイプ]のアドレスは下記公式で決定します：

“設定した定数アドレス+選択されるインデックスレジスタ内の数値”



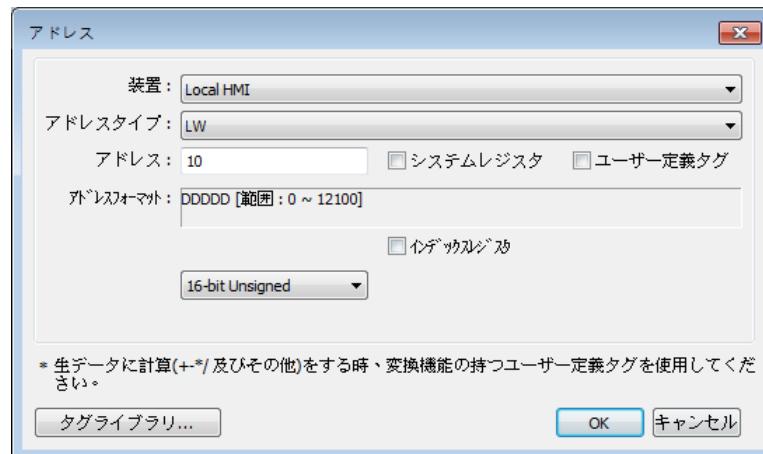
Note

- インデックスレジスタは全ての装置内のワードアドレスレジスタに使用されます。もしビットアドレスレジスタに使用されると、インデックスレジスタ内の数値が 1 を変更したら、16 個のビットアドレスがオフセットされます。

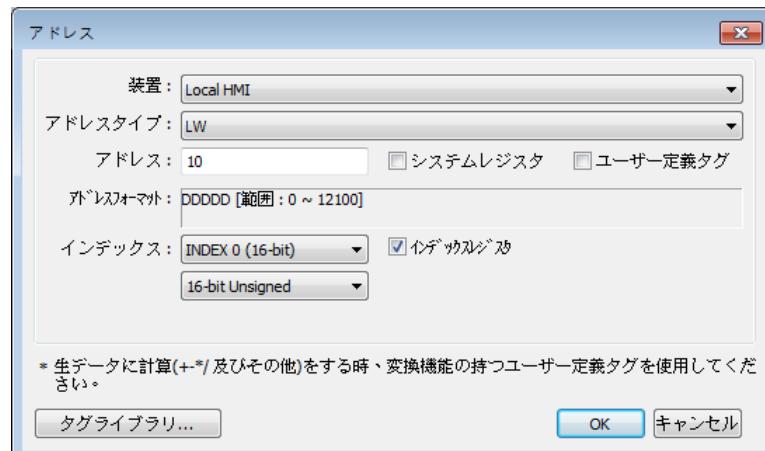
11.2. インデックスレジスタの使用例

以下は実際にアドレスをアクセスする過程です：

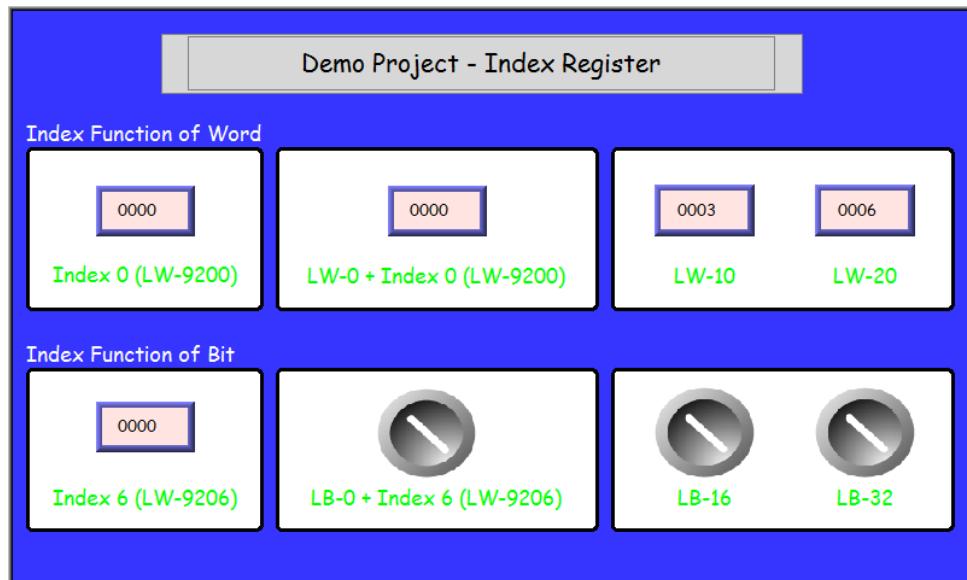
もし[インデックスレジスタ]にチェックマークを入れなく、またアドレスを LW-10 に設定する場合、システムはこのアドレスに対し読み取り/書き込みを行います。



もし[インデックスレジスタ]にチェックマークを入れ、また[インデックス]に[INDEX0(16bit)]を選択した場合、アクセスアドレスは[LW-(10+アドレスインデックスレジスタ 0 の数値)]になります。例：[LW-9200]アドレス内の数値は“5”的場合、公式で算出した実際のアクセスアドレスは[LW-(10+5)]で、即ち[LW-15]になります。

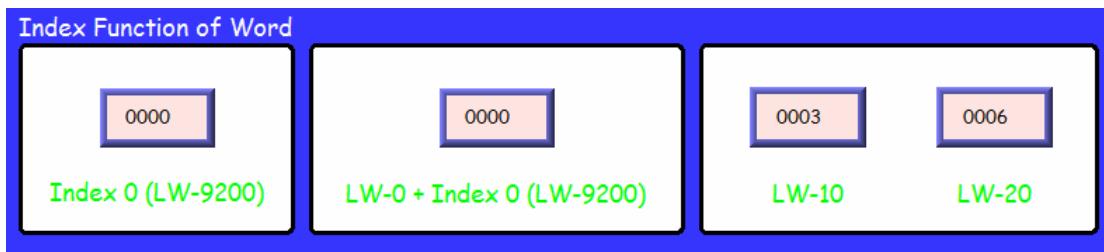


実例でさらなる説明をします：

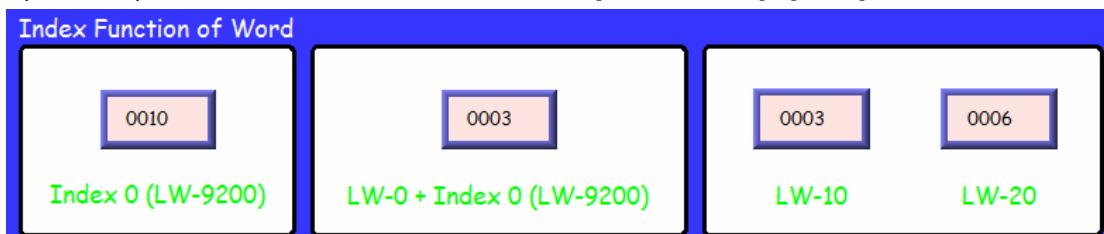


例 1

下図はインデックスレジスタを使用しているワードアドレスレジストの状態を示しています。仮に[LW-0]の数値は“0”、[LW-10]の数値は“3”、[LW-20]の数値は“6”的場合、結果は以下のとおりです：



Index0(LW-9200)アドレス内のデータは“0”的場合、[LW-0+Index0]=[LW-0]を読み取る。



Index0(LW-9200)アドレス内のデータは“10”的場合、[LW-0+Index0]=[LW-10]を読み取る=“3”。

例 2

下図はインデックスレジスタを使用するビットアドレスレジスタの状態を示しています。1個のワード=16個のビットなので、インデックスレジスタ数値に1を変更すると、16個のビットがオフセットされます。仮に[LB-16]がON、[LB-32]がOFFの場合、結果は：



もし Index6(LW-9206)アドレス内のデータを“1”にした場合、スイッチ[LB-0+Index6]は LB-16 アドレスの状態、即ち ON の状態を読み取ります。



もし Index6(LW-9206)アドレス内のデータを“2”にした場合、スイッチ[LB-0+Index6]は LB-32 アドレスの状態、即ち OFF の状態を読み取ります。

Note

- インデックスレジスタをビットアドレスレジスタに使用している時、設定されるビットアドレスは 16 個のビットアドレスを一個のオフセット単位にします。例として、仮に LB-0 がインデックスレジスタを使用すれば、もしインデックスレジスタ内の数値は 1 の場合、LB-16 が稼動します。もしインデックスレジスタ内の数値は 2 の場合、LB-32 が稼動します。



サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

12. キーボードのデザインと 使用

本章では、キーボードのデザイン及び使用する方法を説明します。

12.1. 概要	12-2
12.2. ポップアップキーボードをデザインする	12-2
12.3. 直接ウインドウでキーボードをデザインする	12-5
12.4. 固定キーボードを作成する	12-6
12.5. Unicode キーボードを作成する	12-6

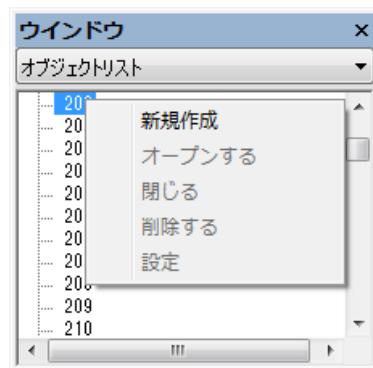
12.1. 概要

数値やアスキーオブジェクトには、キーボードを入力ツールとして使用する必要があり、それに数字キーボードとアスキーキーボードは全部、ファンクションキーオブジェクトで作成されているのもです。EasyBuilder Pro が提供した内蔵キーボードを利用する以外、ご自分でキーボードをデザインすることができます。キーボードの種類は下記に分けられます：

- ポップアップキーボード(ウインドウタイトルバーを使用可能)
- 固定キーボード
- Unicode 文字キーボード

12.2. ポップアップキーボードをデザインする

1. キーボードに作成する予定のウインドウをオープンし作成します。ここではウインドウ 200 を例にします。

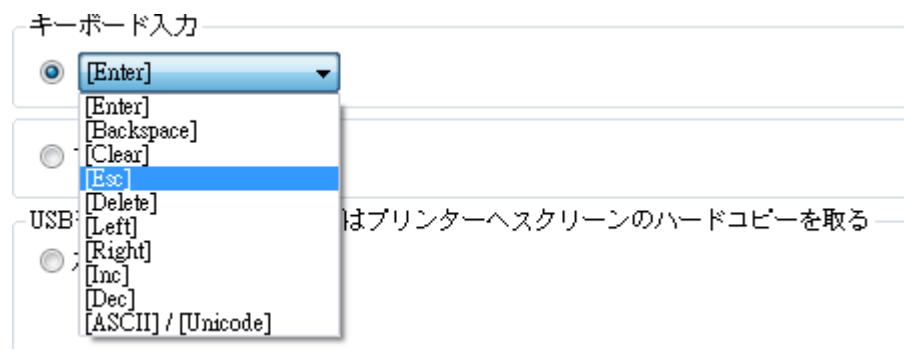


2. ウィンドウ 200 の長さと幅を調整し、[キーボード入力]で各ファンクションキーオブジェクトを設定します。

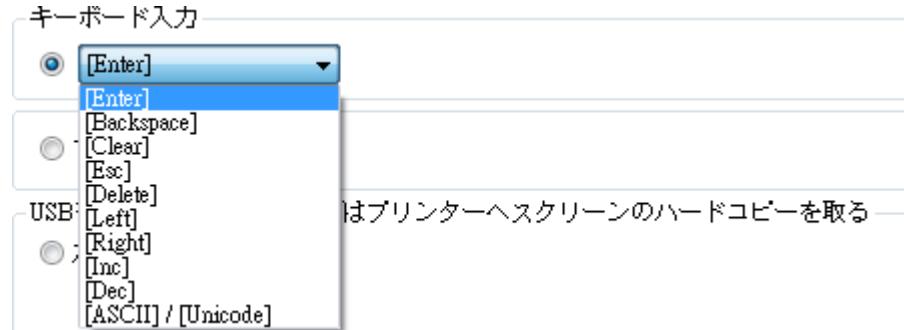


その中に 1 つのファンクションキーを取り消し信号[ESC]をトリガーする用途にします。

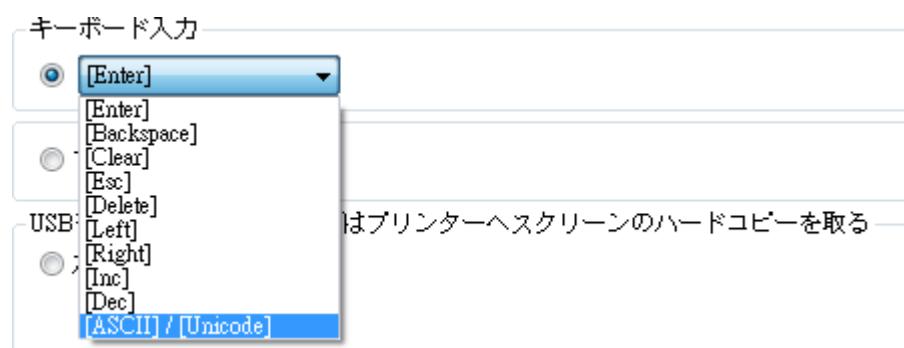
キーボードのデザインと使用



もう 1 つのファンクションキーを入力信号[Enter]をトリガーする用途にします。



他の大部分のファンクションキーを数値入力信号をトリガーする用途にします。例えば、数値 1 の入力信号をトリガーします。



3. 最後はファンクションキーに適する画像を選んで表示させます。



キーボードのデザインと使用

4. [システムパラメータ設定] » [一般的な属性] » [キーボード]設定で、[新規作成]を押して[ウインドウ 200]を追加します。最大 32 個のキーボードウインドウを新規作成することができます。



5. 上記の全てのステップを完了したら、数値やアスキオオブジェクトの設定タブを使用する時に、[数値入力] » [キーボード]設定の[ウインドウ番号]では“200. キーボード”が見つかれます。[キーボードのポップアップ位置]ではキーボードが HMI に現れる位置を選択することができ、下図のように HMI モニターが 9 箇所に分けられます。

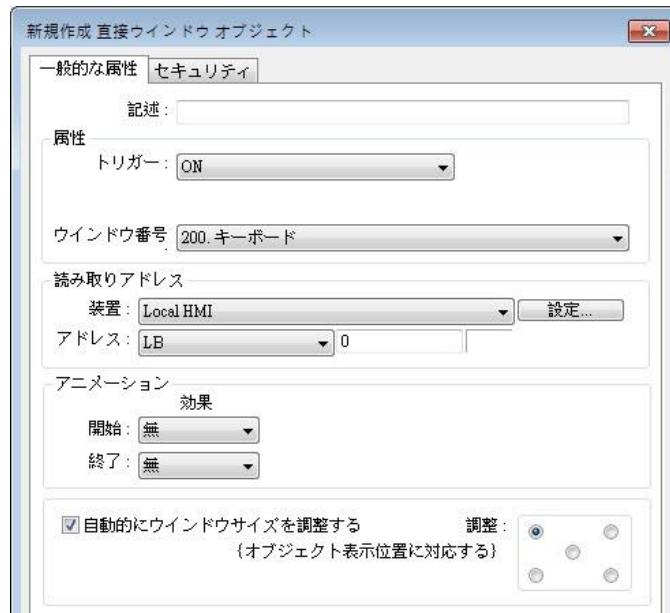


6. [200. キーボード]を選択した後、数値またはアスキオオブジェクトを使用する時に、ウインドウ 200 が自動ポップアップされ、キーボードのファンクションキーを押せば数値を入力することができます。



12.3. 直接ウインドウでキーボードをデザインする

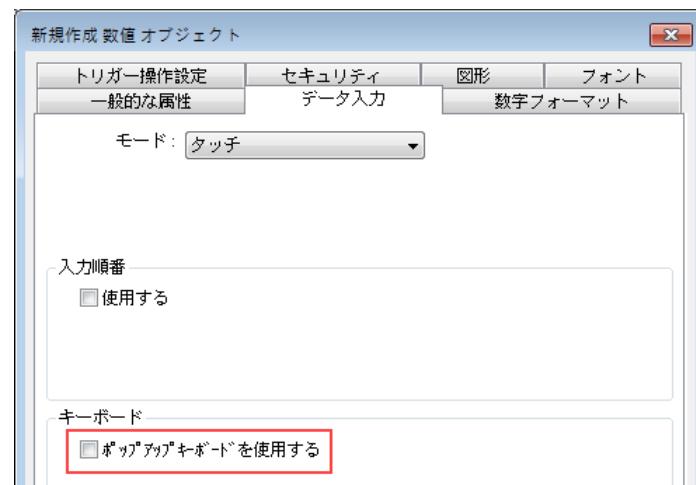
- 直接ウインドウオブジェクトを新規作成し、アドレス LB-0 を読み取ることによって直接ウインドウを起動するのを設定します。[属性]で[ウインドウタイトルバーを隠す]と選択し、[ウインドウ番号]を設定します。



- [輪郭]タブでオブジェクトのサイズをキーボードウインドウに一致させます。



- 数値オブジェクトを新規作成し、データ入力タブ上の[ポップアップキーボードを使用する]にチェックマークを入れないでください。



4. ビット状態設定オブジェクトを設定し、書き込みアドレスを LB-0、スイッチタイプを[ONにする]にして、それに数値オブジェクトの上に重ねます。数値オブジェクトを押した同時に、キーボードウインドウが開かれます。



5. キーボードの[Enter]ファンクションキーと[ESC]ファンクションキーに、それぞれ 1 個のビット状態設定オブジェクトを置き、書き込みアドレスを LB-0、スイッチタイプを[OFF にする]に設定します。この 2 つキーのいずれを押せば、キーボードウインドウが閉じられます。

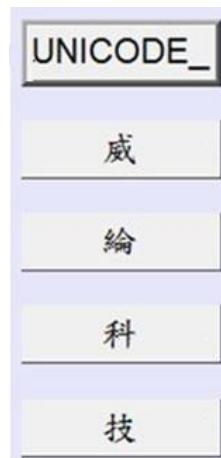
12.4. 固定キーボードを作成する

ポップアップキーボードまたは直接ウインドウを使用しない場合、固定キーボードを使用してもいいです。但し、このキーボードを移動または閉じることができません。

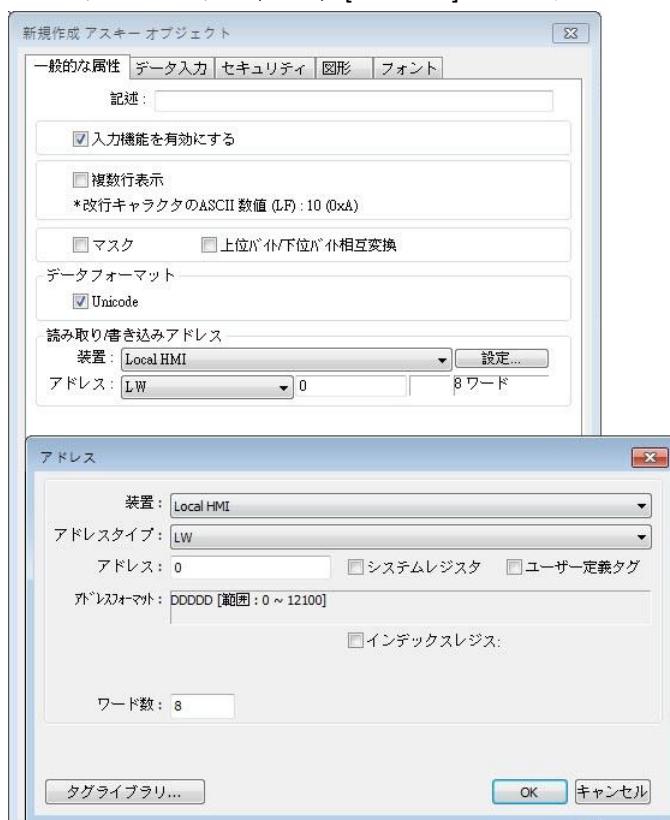
1. 数値オブジェクトを新規作成し、[数値入力] » [キーボード]属性で[ポップアップキーボードを使用]にチェックマークを入れないでください。
2. ファンクションキーオブジェクトでキーボードをデザイン完了後、ウインドウに置くと使用することができます。
3. 数値オブジェクトを押したら、キーボードのファンクションキーで数値を入力することができます。

12.5. Unicode キーボードを作成する

本節では、ファンクションキーオブジェクトで Unicode キーボードを作成する方法を説明します：

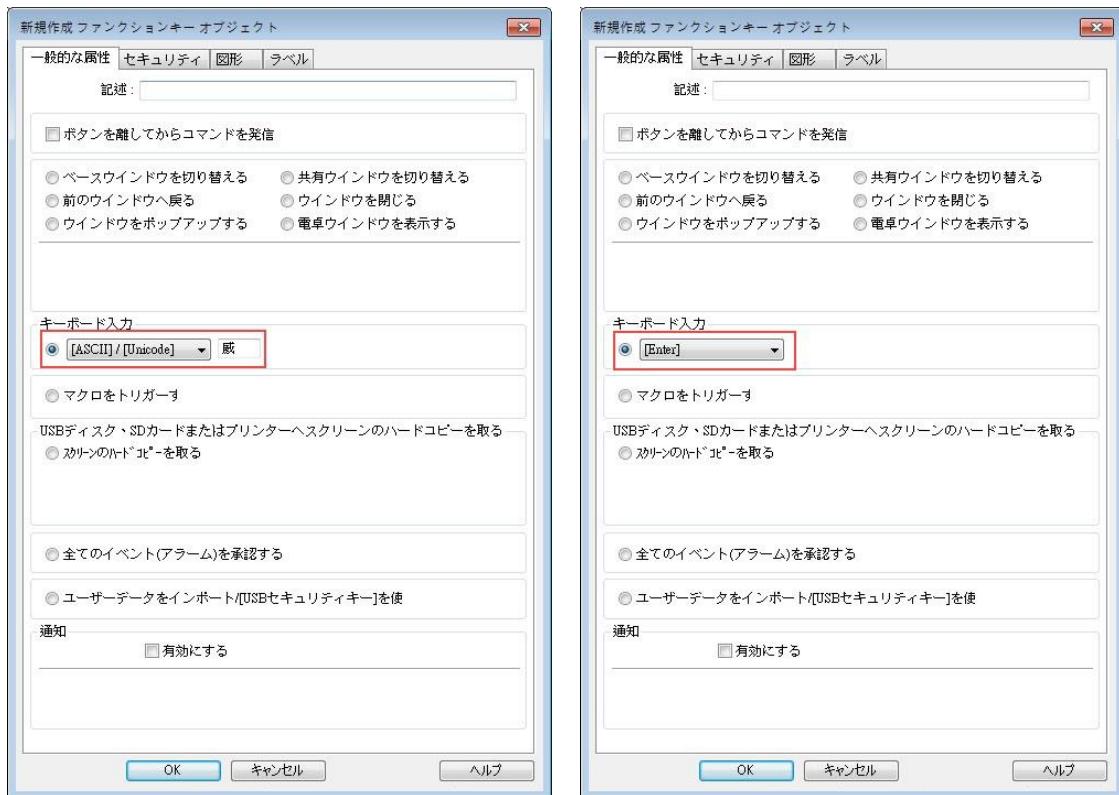


1. アスキーオブジェクトをウインドウに置き、[Unicode]にチェックマークを入れます。



2. 「威」、「綸」、「科」、「技」との4個のアスキーアンクションキーを作成し、また1個の[Enter]入力アンクションキーを作成したら、簡単な Unicode キーボードが完成されます。

キーボードのデザインと使用



Note

- 今後の使用により便利になるように、作成したキーボードを[グループ画像]と設定し、[グループ画像ライブラリ]に追加することができます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13. オブジェクト

本章では、各種オブジェクトのデザイン及び使用方法について説明します。

13.1.	ビットランプ	13-3
13.2.	ワードランプ(マルチ状態ランプ).....	13-5
13.3.	ビット状態設定	13-10
13.4.	ワード設定(マルチ状態設定).....	13-13
13.5.	ファンクションキー	13-21
13.6.	トグルスイッチ	13-27
13.7.	マルチ状態切替スイッチ	13-30
13.8.	スライドスイッチ	13-34
13.9.	数値	13-40
13.10.	アスキー	13-59
13.11.	間接ウインドウ	13-62
13.12.	直接ウインドウ	13-71
13.13.	移動/回転図形.....	13-78
13.14.	アニメーション	13-83
13.15.	棒グラフ	13-87
13.16.	メーター	13-96
13.17.	トレンド図	13-104
13.18.	履歴データ表示	13-124
13.19.	データグループ表示	13-132
13.20.	XYプロット	13-140
13.21.	アラームバーとアラーム表示	13-149
13.22.	イベント表示	13-156
13.23.	データ転送	13-168
13.24.	バックアップ	13-176
13.25.	メディアプレーヤー	13-184
13.26.	BACnet Schedule.....	13-191
13.27.	PLCコントロール	13-195
13.28.	スケジューラー	13-203
13.29.	オプションリスト	13-214
13.30.	タイマー	13-223
13.31.	映像入力	13-228
13.32.	システムメッセージ	13-235

13.33. レシピ閲覧	13-237
13.34. フローブロック	13-243
13.35. 操作ログ	13-248
13.36. 複合ボタン	13-261
13.37. 近似曲線円グラフ	13-269
13.38. 画像閲覧	13-278
13.39. ファイルブラウザ	13-280
13.40. インポート/エクスポート	13-283
13.41. 円グラフ	13-288
13.42. バーコード表示	13-292
13.43. ストリングテーブル	13-308
13.44. データベース	13-310
13.45. 動的スケール	13-325
13.46. 動的作画	13-328
13.47. PDF リーダー	13-334
13.48. テーブル	13-336
13.49. VNC Viewer	13-338
13.50. 連絡先エディター	13-342
13.51. イベントバーチャート	13-345
13.52. 動作トリガー	13-350
13.53. カレンダー	13-356
13.54. タッチジェスチャー	13-358

13.1. ビットランプ

13.1.1. 概要

[ビットランプ]オブジェクトはビットレジスタの状態を表示するものです。状態 0 はビットの状態が OFF ; 状態 1 はビットの状態が ON と示しています。

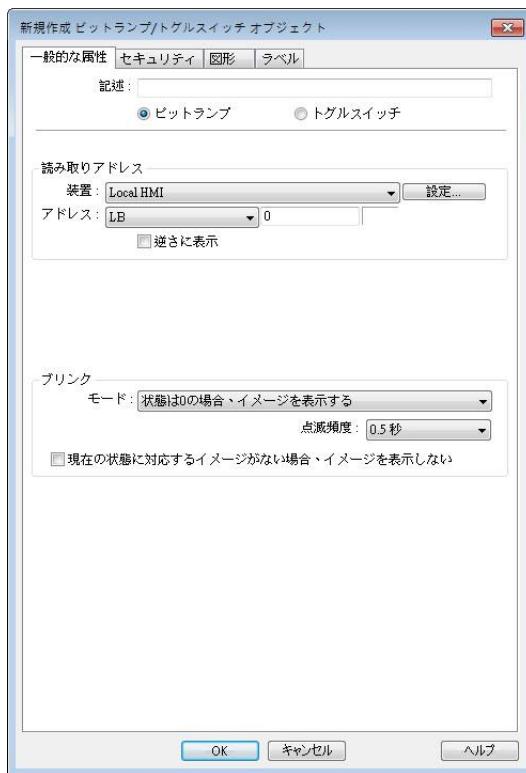


13.1.2. 設定



ツールバー上の[オブジェクト]»[ビットランプ]ボタンを押せば、[ビットランプ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[ビットランプ]オブジェクトが新規作成されます。

一般的な属性の設定



設定

記述

記述	ユーザーはこのオブジェクトに関連情報を記述できます。 ビットランプ/トグルスイッチ [トグルスイッチ]機能と相互に切り替えることができます。
読み取りアドレス	[設定]をクリックした後、ビットアドレス装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、ビットランプオブジェクトをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。
逆さに表示	読み取りの状態を逆さに表示できます。例えば、ビットの実際状態は OFF だったが、[逆さに表示]にチェックマークを入れたら ON に表示します。
プリント	オブジェクトのビット状態が ON または OFF のを表示します。 モード： 無 点滅しません。 状態は 0 の場合、イメージを表示する ビットは OFF の時に、状態 0 のイメージ及び状態 1 のイメージが交互に表示します。 状態は 1 の場合、イメージを表示する ビットは ON の時に、状態 0 のイメージ及び状態 1 のイメージが交互に表示します。 状態は 0 の場合、点滅する ビットは OFF の時に、状態 0 のイメージは点滅します。 状態は 1 の場合、点滅する ビットは ON の時に、状態 1 のイメージは点滅します。 現在の状態に対応するイメージがない場合、イメージを表示しない 本項にチェックマークを入れると、イメージの数量が不足で全ての状態を表示しきれない場合、イメージを表示しません。さもなければ、最後の状態を表示します。

 Note

- ラベルタグで[ON=OFF(使用状態)]にチェックマークを入れると、状態 0 と 1 は全部状態 0 の属性設定を使用します。

13.2. ワードランプ(マルチ状態ランプ)

13.2.1. 概要

[ワードランプ(マルチ状態ランプ)]オブジェクトはワードレジスタ内の数値を使用し、対応した状態のイメージを表示します(最大 256 種類の状態表示をサポート)。レジスタ内の数値は 0 の場合、[状態 0]を表示し、数値は 1 の場合、[状態 1]を表示します。このように類推します。

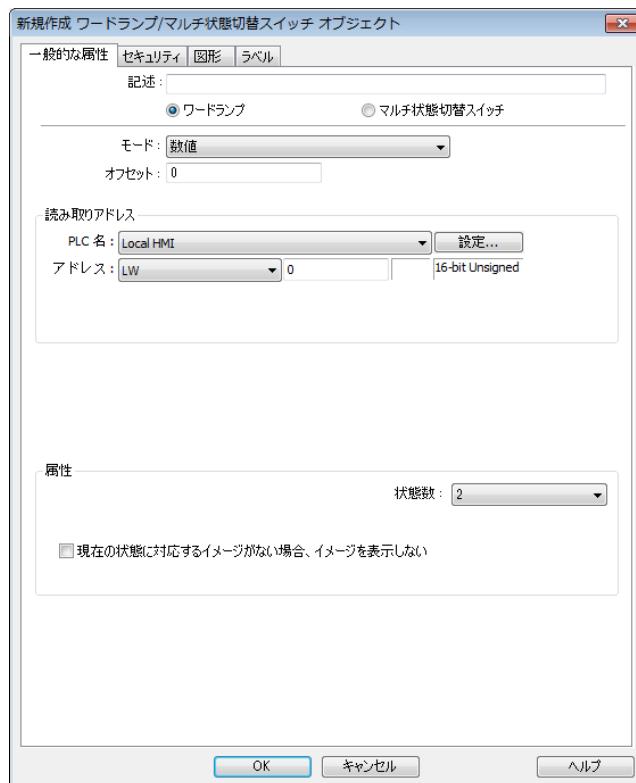


13.2.2. 設定

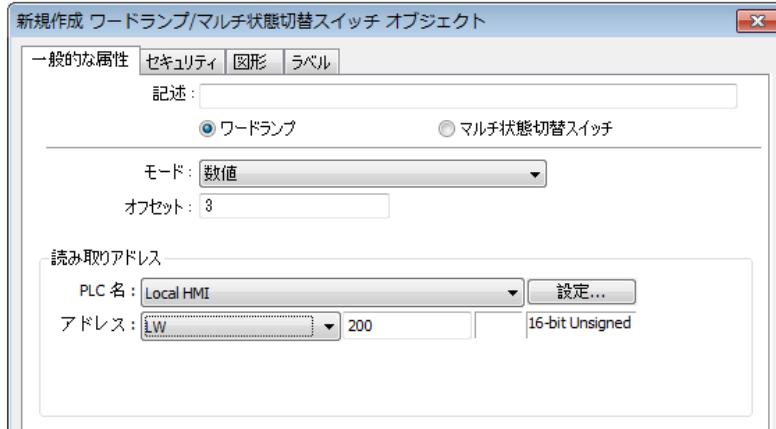


ツールバーの[ワードランプ]ボタンを押せば、[ワードランプ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[ワードランプ]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述
記述	ユーザーはこのオブジェクトに関連情報を記述できます。 ワードランプ/マルチ状態切替スイッチ [マルチ状態切替スイッチ]機能と相互に切り替えることができます。
モード/オフセット量	[ワードランプ]オブジェクトは下記三種類の表示モードを提供しています。 数値 直接にレジスタ内の数値に[オフセット量]を加減した結果をオブジェクトの現在状態とします。下図のように、数値 3 をレジスタアドレス: LW-200 に書き込んだ際に、オフセット量 3 があるので、アドレス LW-200 のオブジェクトは状態 0(数値 3-オフセット量 3)の画像を表示します。



LSB

本モードはまずレジスタ内の数値を2進数にし、続いて0でない最低ビットでオブジェクトの現在状態を決めます。下記リストを例に挙げます：

十進数	二進数	表示する内容
0	00000000000000000000	全bitは0の場合、0を表示する
1	00000000000000000001	0でない最小ビットはbit0の場合、状態1を表示する
2	00000000000000000010	0でない最小ビットはbit1の場合、状態2を表示する
3	00000000000000000011	0でない最小ビットはbit0の場合、状態1を表示する
4	000000000000000000100	0でない最小ビットはbit2の場合、状態3を表示する
7	000000000000000000111	0でない最ビットはbit0の場合、状態1を表示する
8	00000000000000001000	0でない最小ビットはbit3の場合、状態4を表示する
16	000000000000000010000	0でない最小ビットはbit4の場合、状態5を表示する
32	0000000000000000100000	0でない最小ビットはbit5の場合、状態6を表示する
64	00000000000000001000000	0でない最小ビットはbit6の場合、状態7を表示する
128	000000000000000010000000	0でない最小ビットはbit7の場合、状態8を表示する
256	0000000000000000100000000	0でない最小ビットはbit8の場合、状態9を表示する
512	00000000000000001000000000	0でない最小ビットはbit9の場合、状態10を表示する

	1024	000001000000000000	0でない最小ビットはbit 10の場合、状態11を表示する
	2048	000010000000000000	0でない最小ビットはbit 11の場合、状態12を表示する
	4096	000100000000000000	0でない最小ビットはbit 12の場合、状態13を表示する
	8192	001000000000000000	0でない最小ビットはbit 13の場合、状態14を表示する
	16384	010000000000000000	0でない最小ビットはbit 14の場合、状態15を示す
	32768	100000000000000000	0でない最小ビットはbit 15の場合、状態16を表示する

ビット組み合わせ

ランプの状態はビットの状態表示によります。属性の状態数を変更すれば、異なる装置の読み取りアドレスを増加することができます。装置1は最下位と示し、装置2はその次で、このように類推します。最大は4アドレスで合計16状態数を読み取ることができます。

周期的に状態を変換する

オブジェクトの状態は固定した頻度で順次に状態を変換します。ユーザーも[頻度]で状態の変換頻度を変更することができます。

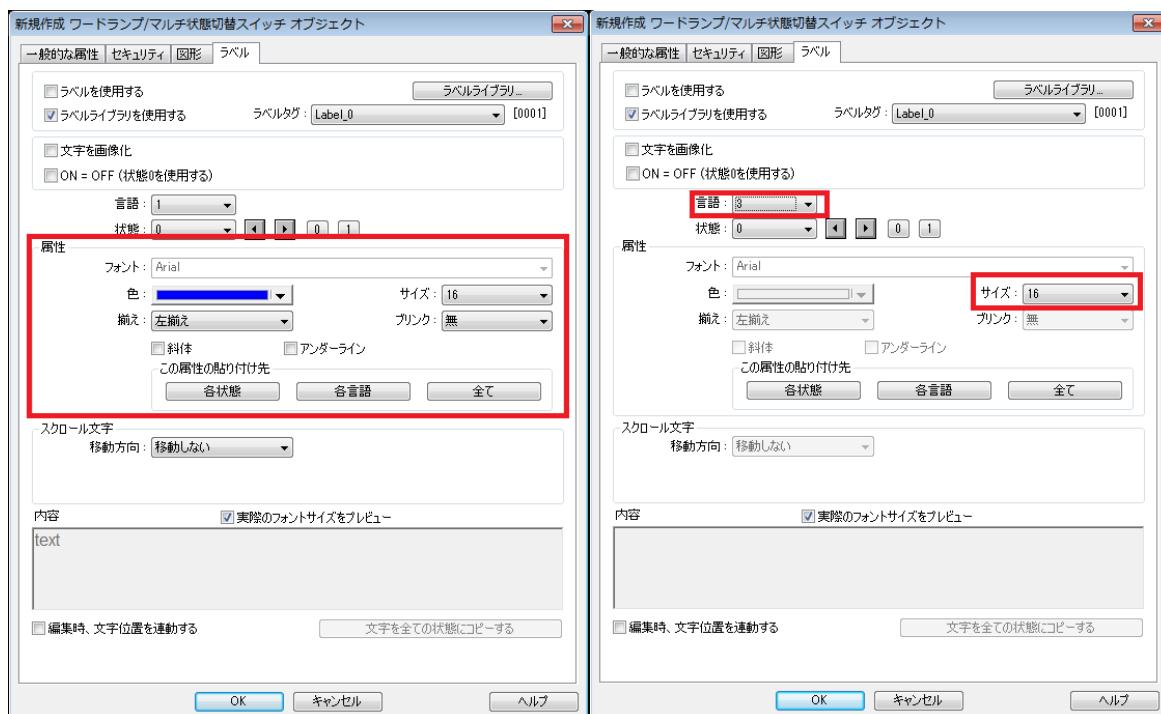
読み取りアド レス	[設定]をクリックした後、ワードレジスタ装置タイプの[装置]、[アド レス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ] を選択し、ワードランプオブジェクトをコントロールします。ユー ザーも[一般的な属性]タブでアドレスを設定することができます。
--------------	--

属性	状態数
	オブジェクトが表示する状態の数です。状態は0から番号付けし、表示できる最大の状態番号は設定した[状態数]-1で、設定した数より超えた数を要求した場合、システムは最後の状態を表示します。例えば[状態数]は8の場合、表示する状態は順番で0、1、2、...、7で、状態8(含)以上の状態を表示するように要求した時、状態7のみ表示されます。 現在の状態に対応する画像/図形がない場合、イメージを表示しない本項にチェックマークを入れると、イメージの数量が不足で全ての状態を表示しきれない場合、イメージを表示しません。さもなければ、最後の状態を表示します。

Note

- ラベルタグ内の言語1はフォントの関連属性の設定を変更することができます。ただし、言語2~8は文字のサイズだけ変更でき、他の属性設定は皆、言語1と同じにされています。

オブジェクト



13.3. ビット状態設定

13.3.1. 概要

[ビット状態設定]はビットレジスタの状態を設定するオブジェクトです。このオブジェクトは手動操作と自動実行との二種類の操作モードを提供しています。手動操作モードを使用する場合、本ボタンを押せばレジスタの状態を ON、または OFF することができます。

自動実行モードを使用すると、ある特定条件下では自動的に指定した動作を実行します。この操作モードを使用すれば、[ビット状態設定]オブジェクトを押しても何の影響も与えません。

13.3.2. 設定



ツールバー上の[オブジェクト]»[ビット状態設定]ボタンを押せば、[ビット状態設定]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[ビット状態設定]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述	
書き込みアドレス	[設定]をクリックした後、ビットレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、ビット状態設定オブジェクトをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。	
ボタンを離してからコマンドを発信	本設定を使用すると、オブジェクトを押した後、完全にボタンから離してから、オブジェクトが定義した操作モードが実行されます。もし本設定を使用しなかったら、一旦ボタンを押すと、オブジェクトはすぐに実行されます。もしモーメンタリモードを使用する場合、本機能をサポートしていません。	
属性	スイッタイプ	記述
	ON にする	このオブジェクトを押すと、指定したレジスタの状態が ON に設定されます。
	OFF にする	このオブジェクトを押す、指定したレジスタの状態が OFF に設定されます。
	トグル	このオブジェクトを押すと、指定したレジスタの状態が逆さまに表示されます。即ち ON→OFF, OFF→ON になります。
	モーメンタリ	このオブジェクトを押すと、指定したレジスタの状態はまず ON に設定されたが、手を放せば、状態は OFF に設定されます
	周期切替スイッチ	指定したレジスタの状態が ON/OFF で周期的に切替えます。本モードは自動的に実行されます。設定できる周期は0.1~25.5 秒です。
	ウインドウがオープンされた時、ON にする	オブジェクトが所在しているウインドウがオープンされた場合、指定したレジスタの状態は自動的に ON に設定されます。
	ウインドウがオープンされた時、OFF にする	オブジェクトが所在しているウインドウがオープンされた場合、指定したレジスタの状態は自動的に OFF に設定されます。
	ウインドウが閉じられた時、ON にする	オブジェクトが所在しているウインドウが閉じられた場合、指したレジスタの状態は自動的に ON に設定されます。
	ウインドウが閉じられた時、OFF にする	オブジェクトが所在しているウインドウが閉じられた場合、定したレジスタの状態

		は自動的にOFFに設定されます。
バックライトがついた時、ONにする(cMT-SVRをサポートしていない)		バックライトがつく場合、指定したレジスタの状態は自動的にONに設定されます。
バックライトがついた時、OFFにする(cMT-SVRをサポートしていない)		バックライトがつく場合、指定したレジスタ状態は自動的にOFFに設定されます。
バックライトが消された時、ONにする(cMT-SVRをサポートしていない)		バックライトが消された場合、指定したジスタの状態は自動的にONに設定されます。
バックライトが消された時、OFFにする(cMT-SVRをサポートしていない)		バックライトが消された場合、指定したレジスタの状態は自動的にOFFに設定されます。

マクロ

[ビット状態設定]オブジェクトはマクロを合わせて使用することができます。本機能を使用する前、予めにマクロを作成する必要があります。

 マクロコマンドを作成するには《18 マクロの説明》をご参照ください。

トリガーモード

オブジェクトの操作モードで[トグルスイッチ]を選択した時、マクロを実行する条件を設定します。状態が OFF→ON または ON→OFF になった時点でのマクロコマンドを実行するように設定でき、或いは状態が変更したたびに(OFF<->ON)、すぐマクロを実行するように設定できます。

 **Note**

- ラベルタグで[ON=OFF(使用状態)]にチェックマークを入れたら、状態0と1は全部状態0が設定した属性内容を使用するようになります。
- cMT-SVRはビット設定オブジェクトの[周期切替スイッチ]属性がPLBとPLW_Bit以外のアドレスを使用することをサポートしません。

13.4. ワード設定(マルチ状態設定)

13.4.1. 概要

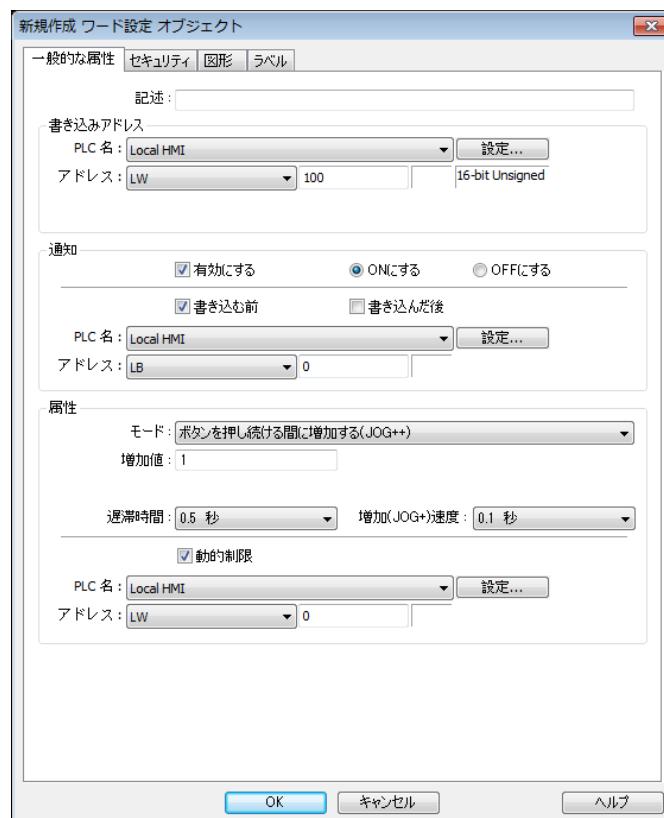
[ワード設定(マルチ状態設定)]はワードレジスタの数値を設定するオブジェクトです。このオブジェクトは手動操作と自動実行との二種類の操作モードを提供しています。手動操作モードを使用する場合、本ボタンを押せばレジスタでの数値を設定することができます。自動実行モードを使用すると、ある特定条件下では自動的に指定した動作を実行します。この操作モードを使用すれば、[マルチ状態設定]ボタンを押しても何の影響も与えません。

13.4.2. 設定



ツールバー上の[オブジェクト]»[ワード設定]ボタンを押せば、[ワード設定]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[ワード設定]オブジェクトが新規作成されます。

一般的な属性の設定



設定

記述

書き込みアドレス	[設定]をクリックした後、ワードレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、ワード設定オブジェクトをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。
ボタンを離してからコマンドを発信	本設定を使用すると、オブジェクトを押した後、完全にボタンから離してから、オブジェクトが定義した操作モードが実行されます。もし本設定を使用しなかったら、一旦ボタンを押すと、オブジェクトはすぐに実行されます。
通知	本設定を使用すると、手動モードでの動作が完了後には、連帶に本項が指定したレジスタの状態を設定することができ、[ON]と[OFF]で状態を設定します。
[書き込む前]/[書き込んだ後]	書き込みの前/後で指定したレジスタの状態を設定します。

属性	モード
	オブジェクトの動作モードを選択します。選択できるモードを下記例 2 をご参照ください。
	動的制限

指定のレジスタで上下限を設定することができます。下記例 1 をご参考ください。

例 1

指定のレジスタで上下限を設定することができます。書き込みアドレスが LW-n である場合、上/下限は下記規則で自動的に設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMT のみ)
動的制限アドレス	LW-n	LW-n	LW-n
下限	LW-n	LW-n	LW-n
上限	LW-n+1	LW-n+2	LW-n+4

下記を例として、[レジスタアドレス]が LW-100 である場合、上/下限は自動的に以下のとおりに設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMT のみ)
動的制限アドレス	LW-100	LW-100	LW-100
下限	LW-100	LW-100	LW-100
上限	LW-101	LW-102	LW-104

例 2

選択できるモードは以下のとおりです：

- 定数を書き込む

定数を書き込む機能です。オブジェクトを一回押すごとに、[定数を設定する]での設定値は指定したレジスタに書き込まれます。定数のタイプは 16-bit BCD、32-bit BCD、...、64-bit Double などです。下図を例として、本ボタンを押したら、数値 12 を指定したレジスタに書き込みます。



- 増加する(JOG+)

数値を増加する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値に[増加値]で設定された数値を加えます。但し、増加した結果は[上限値]の設定値を超えることはありません。

下図を例として、本ボタンを一回押すごとに、指定したレジスタでの数値を +1 にし、上限値 10 に達するまで増加されます。



- 減少する(JOG-)

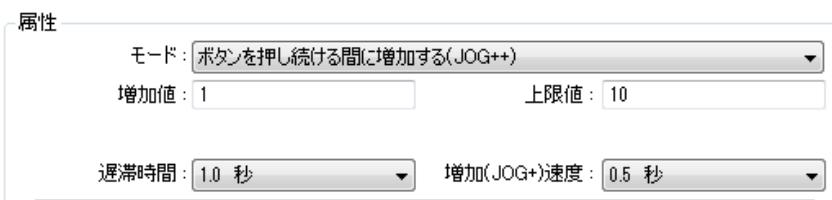
数値を減少する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値に[減少値]で設定された数値を引きます。但し、減少した結果は[下限値]の設定値を超えることはありません。

下図を例として示す通り、本ボタンを一回押すごとに、指定したレジスタ内の数値を -1 にし、下限値 0 に達するまで減少されます。



- ボタンを押し続ける間に増加する(JOG++)

ボタンを押し続ける間に数値を増加する機能です。押し時間が[遅延時間]の設定値を超えた後、指定したレジスタ内の数値に[増加速度]で設定した速度で、毎回ごとに[増加値]で設定された数値を加えます。但し、増加した結果は[上限値]の設定値を超えることはありません。下図を例として、本ボタンを一回押すごとに、指定したレジスタ内の数値を +1 にし、本ボタンを 1.0 秒以上に押し続けると、0.5 秒ごとに +1 をし続け、上限値 10 に達するまで増加されます。

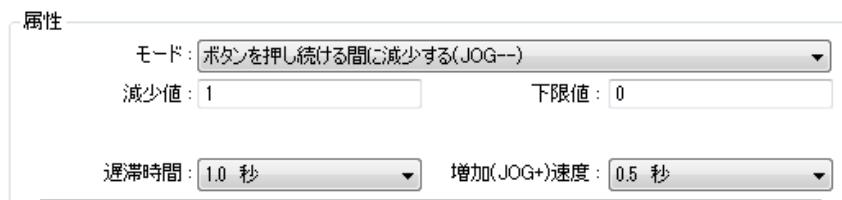


- ボタンを押し続ける間に減少する(JOG--)

ボタンを押し続ける間に数値を減少する機能です。押し時間が[遅延時間]の設定値を超えた後、

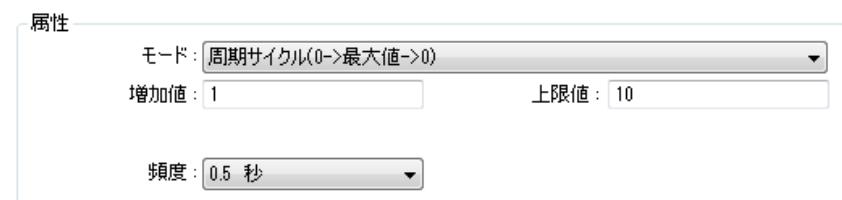
オブジェクト

指定したレジスタ内の数値に[増加速度]で設定した速度で、毎回ごとに[減少値]で設定された数値を引きます。但し、減少した結果は[下限値]の設定値を超えられません。下図を例として、本ボタンを一回押すごとに、指定したレジスタでの数値を-1 にし、本ボタンを 1.0 秒以上に押し続けると、0.5 秒ごとに-1 をし続け、下限値 0 に達するまで減少されます。



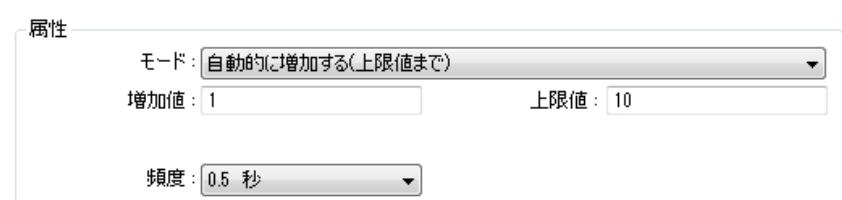
● 周期サイクル(0->最大値->0)

周期的に数値を増加する機能です。[マルチ状態設定]オブジェクトは[頻度]で設定した周期と[増加値]で設定した増加値を基づいて、自動的に指定したレジスタ内の数値を増加します。但し、増加した結果は[上限値]の設定値を超えることはありません。下図を例として、システムはレジスタ内の数値に 0.5 秒ごとに+1 をし続け、上限値 10 に達するまで増加された後、数値は 0 に戻し、改めて+1 をし続けます。



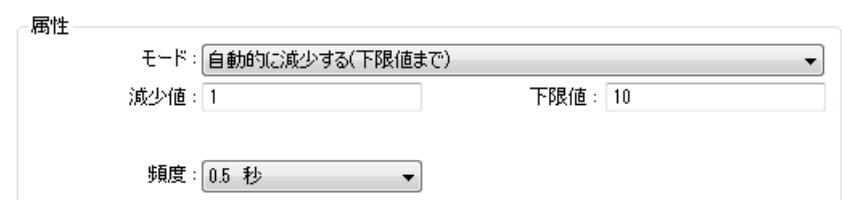
● 自動的に増加する(上限値まで)

周期的に数値を増加する機能です。[マルチ状態設定]オブジェクトは[頻度]で設定した周期と[増加値]で設定した増加値を基づいて、自動的に指定したレジスタ内の数値を増加し、結果が[上限値]に達すると、自動的に停止します。下図を例として、システムはレジスタ内の数値に 0.5 秒ごとに+1 をし続け、上限値 10 に達すると停止します。



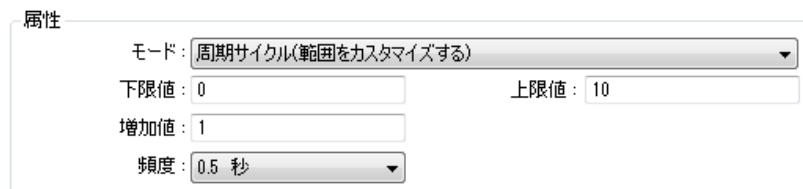
● 自動的に減少する(下限値まで)

周期的に数値を減少する機能です。[マルチ状態設定]オブジェクトは[頻度]で設定した周期と[減少値]で設定した減少値を基づいて、自動的に指定したレジスタ内の数値を減少し、結果が[下限値]に達すると、自動的に停止します。下図を例として、システムはレジスタ内の数値に 0.5 秒ごとに-1 をし続け、下限値 0 に達すると停止します。



- 周期サイクル(範囲をカスタマイズする)

週期的に増加/減少をサイクルで実行する機能です。[マルチ状態設定]オブジェクトは[頻度]で設定した周期と[増加値]で設定した設定値を基づいて、自動的に指定したレジスタ内の数値を[上限値]に等しいまで増加します。続いて同じ周期を用いて、レジスタ内の数値を[下限値]に等しいまで減少します。下図を例として、システムは自動的に指定したレジスタ内の数値を 0.5 秒ごとの頻度で+1 にし、上限値 10 に達するまで増加します。そして同じ周期を用いて、レジスタ内の数値を下限値 0 に等しいまで減少し、それを循環し続けます。



- 周期的に増加する(小→大)

遞増機能です。[ワード設定]オブジェクトは[頻度]で設定した周期と[増加値]で設定した設定値を基づいて、自動的に指定したレジスタ内の数値を[最大値]に等しいまで増加します。続いてレジスタ内の数値を[最小値]に復帰し、それに前の動作を繰り返して数値が動的な変化を保つようにさせます。下図を例として、システムは自動的に指定したレジスタ内の数値を 0.5 秒ごとの頻度で+1 にし、上限値 10 に達するまで増加します。そしてまた下限値 0 に復帰し改めて増加し、それを循環にし続けます。



- 周期的減少する(大→小)

遞減機能です。[マルチ状態設定]オブジェクトは[頻度]で設定した周期と[減少値]で設定した設定値を基づいて、自動的に指定したレジスタ内の数値を[最小値]に等しいまで減少します。続いてレジスタ内の数値を[最大値]に復帰し、それに前の動作を繰り返して数値が動的な変化を保つようにさせます。下図を例として、システムは自動的に指定したレジスタ内の数値を 0.5 秒ごとの頻度で-1 にし、下限値 0 に達するまで減少します。そしてまた上限値 10 に復帰し改めて減少し、それを循環にし続けます。



- ウィンドウがオープンされた時、設定する/ウィンドウが閉じられた時、設定するオブジェクトの所在ウィンドウがオープンされた s /閉じれた際に、[定数を設定する]での設定値を自動的に指定したレジスタに書き込みます。[定数を設定する]を 5 に設定した場合、当ペー

ジのウインドウがオープンされた/閉じられた際に、システムは自動的に数値 5 をレジスタに書き込みます。

- バックライトがついた時、設定する/バックライトが閉じられた時、設定する(cMT をサポートしていません)

バックライトがもともとオフ/オンにいる状態で、一旦逆の状態になると、[定数を設定する]での設定値を自動的に指定したレジスタに書き込みます。[定数を設定する]を 5 に設定した場合、当ページのバックライト状態が変わり次第、システムは自動的に数値 5 をレジスタに書き込みます。

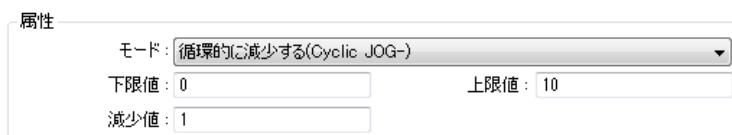
- 循環的に増加する(Cyclic JOG+)

数値を増加する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値に[増加値]で設定された数値を増加します。数値が上限値に達したら、また下限値に復帰し改めて増加します。下図を例として、オブジェクトを一回押すごとに、指定したレジスタ内の数値に+1 にし、上限値 10 に達したら自動的に 0 に復帰し、改めて増加します。



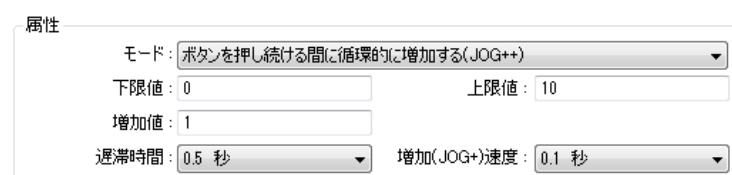
- 循環的に減少する(Cyclic JOG-)

数値を減少する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値に[減少値]で設定された数値を減少します。数値が下限値に達したら、また上限値に復帰し改めて減少します。下図を例として、オブジェクトを一回押すごとに、指定したレジスタ内の数値に-1 にし、下限値 10 に達したら自動的に 10 に復帰し、改めて減少します。



- ボタンを押し続ける間に循環的に増加する(JOG++)

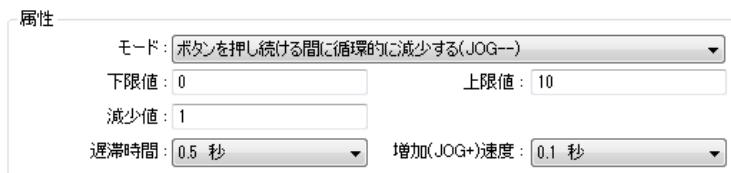
ボタンを押し続ける間に数値を増加する機能です。押し時間が[遅滞時間]の設定値を超えたたら、指定したレジスタ内の数値は[増加速度]で設定した速度で、毎回ごとに[増加値]で設定された数値を上限値に達するまで増加し、その後、下限値に復帰します。下図を例として、本ボタンを一回押すごとに、指定したレジスタ内の数値を+1 にし、本ボタンを 0.5 秒以上に押し続けると、0.1 秒ごとに+1 をし続け、上限値 10 に達するまで増加され、そしてまた 0 に復帰し、改めて増加します。



- ボタンを押し続ける間に循環的に減少する(JOG--)

ボタンを押し続ける間に数値を減少する機能です。押し時間が[遅滞時間]の設定値を超えたたら、指定したレジスタ内の数値は[増加速度]で設定した速度で、毎回ごとに[減少値]で設定された数値を下限値に達するまで減少し、その後、上限値に復帰します。下図を例として、本ボタンを

一回押すごとに、指定したレジスタでの数値を-1にし、本ボタンを0.5秒以上に押し続けると、0.1秒ごとに-1をし続け、下限値0に達するまで減少され、そしてまた10に復帰し、改めて減少します。



Note

- cMT-SVRはマルチ状態設定オブジェクトの[周期サイクル]、[自動的に増加する]、[自動的に減少する]、[周期的に増加する]、[周期的に減少する]などの属性がPLW以外のアドレスタイプを使用するのをサポートしません。

セキュリティ設定



設定	記述
インターロック機能を使用する	本項にチェックマークを入れ、そして[ワード]を選択すると、このオブジェクトを操作できるかどうかのは、一個の指定したワードアドレスの[状態]によります。上図に示されたように、LW-1の数値が1より大きい場合のみ、本オブジェクトを操作できます。
無効の場合、オブジェクトを隠す	指定したワードアドレス内の数値が[状態]に合致したい場合、オブジェクトは隠されます。
無効の場合、グレースケールで文字を表示する	

オブジェクトのラベル文字は指定したワードアドレス内の数値が[状態]に合致していない場合に、グレースケールで表示されます。

button

状態

指定したワードアドレスをトリガーする状態を設定します。トリガー条件は、`>`、`<`、`==`、`<>`、`>=` 或いは`<=`が設定できます。その中、`==` と `<>` は[許容誤差]の設定に用いられます。

例えば：

装置 :	Local HMI	設定...	
アドレス :	LW	0	16-bit Unsigned
状態 :	==	10	許容誤差 : 1

動作 : 無効された場合、隠す

指定したワードアドレスの数値が大きいイコール 11、或いは小さいイコール 9 の場合、オブジェクトが無効されて隠されます。

13.5. ファンクションキー

13.5.1. 概要

[ファンクションキー]オブジェクトはウインドウの切り替え、キーボードの作成、マクロの実行およびスクリーンのハードコピーなどの機能を提供するオブジェクトです。一方、USB セキュリティキーの設定にも使われます。

cMT シリーズの HMI を使用すれば、cMT Viewer で[スクリーンのハートコピーを取る]、[ユーザ情報をインポート/USB セキュリティキーを使用する]のファンクションキーオブジェクトを閲覧・使用できません。

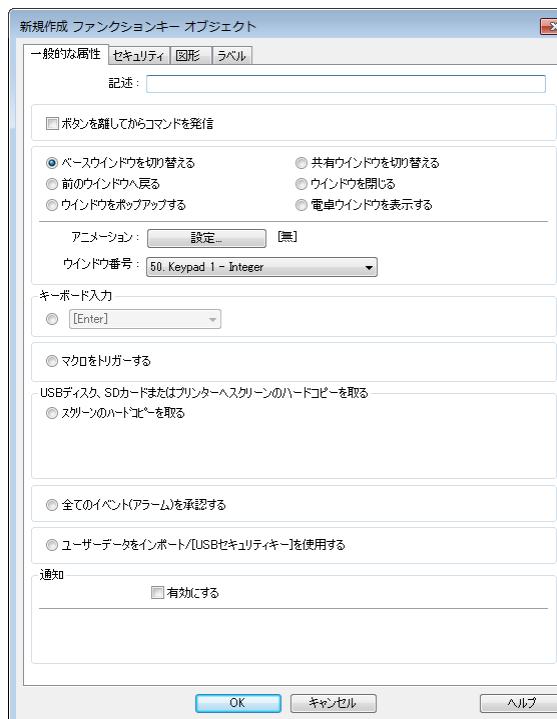
13.5.2. 設定



ツールバー上の[オブジェクト]»[ファンクションキー]ボタンを押せば、[ファンクションキー]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[ファンクションキー]オブジェクトが新規作成されます。

一般的な属性の設定

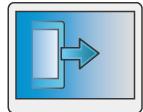
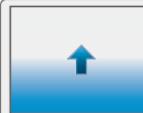
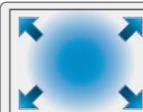
cMT シリーズ



eMT, iE, XE, mTV シリーズ



設定	記述						
ボタンを離してからコマンドを発信	本項目を選択すると、オブジェクトを押した後、完全にボタンから離してから、選択した動作が実行されます。もし本項目を選択しなかつたら、一旦ボタンを押すと、オブジェクトはすぐに実行されます。						
ウインドウを切り替える	<p>ベースウインドウを切り替える：ベースウインドウを切り替えます。</p> <p>共有ウインドウを切り替える：共有ウインドウを切り替えます。</p> <p>ウインドウをポップアップする：他のウインドウを呼び出します。</p> <p>呼び出されたウインドウは必ずベースウインドウの上にあります。</p> <p>本機能を使用すると、下図のように[親ウインドウが閉じられた時、ポップアップウインドウを閉じる]を使用するかを選択できます。本項目を選択すると、呼び出されたウインドウはウインドウを切り替える際に自動的に消えます。本項目を選択しない場合、呼び出されたウインドウを閉じるには、そのウインドウの上で[ウインドウを閉じる]ファンクションキーを作成しなければなりません。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <input checked="" type="radio"/> ベースウインドウを切替える <input checked="" type="radio"/> ウインドウをポップアップする <input checked="" type="checkbox"/> 親ウインドウが閉じられた時、ポップアップウインドウを閉じる スタイル : <input type="button" value="タイトルバーを表示する"/> ウインドウ番号 : <input type="button" value="11. ウインドウ_011"/> </div> <p>アニメーション設定(cMT シリーズ) : cMT シリーズでファンクションキーを使用し、他のウインドウを呼び出す場合、アニメーション効果を設定できます。[アニメーション設定]を押すと、ウインドウをポップアップする際のアニメーション(トランジション)効果を選択できます。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <input checked="" type="radio"/> ウインドウをポップアップする <input checked="" type="radio"/> 電卓ウインドウを表示する アニメーション : <input type="button" value="設定..."/> [無, 無] ウインドウ番号 : <input type="button" value="50. Keypad 1 - Integer"/> <input type="button" value="スタイル/位置..."/> </div> <p>ウインドウがポップアップされる時の[開始]と[終了]のアニメーション効果を設定できます。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="background-color: #668dce;">効果</th> <th style="background-color: #668dce;">スタイル</th> </tr> </thead> <tbody> <tr> <td>フェードイン</td> <td></td> </tr> <tr> <td>飛んでくる</td> <td></td> </tr> </tbody> </table>	効果	スタイル	フェードイン		飛んでくる	
効果	スタイル						
フェードイン							
飛んでくる							

浮かび上がる	
ワイプ	
スプリット	
サイクル	
クロック	
ズーム	
ターン	
プッシュ	

[表示期間]でウインドウの開始/終了アニメーションが完了するまでの時間を設定できます。

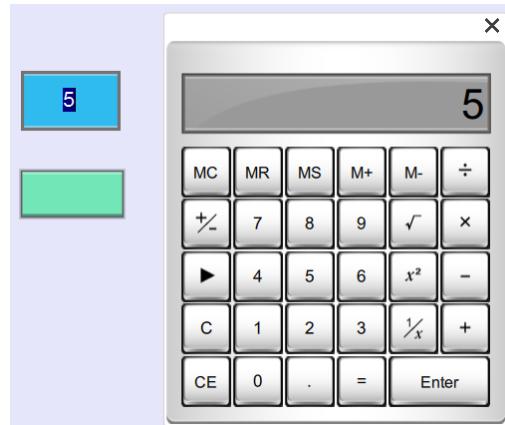
[表示方向]でウインドウの開始/終了アニメーションの移動方向を設定できます。

前のウインドウへ戻る:前のベースウインドウに戻ります。例えば、"ウインドウ 10"から"ウインドウ 20"に切り替える際に、本機能を使用すれば再度"ウインドウ 10"に戻ることができます。本機能はベースウインドウのみ適用します。

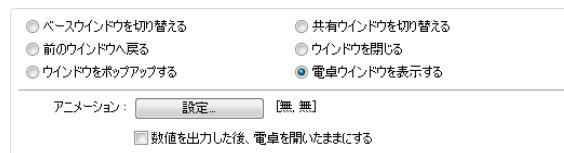
ウインドウを閉じる:ベースウインドウの上で呼び出されたウインドウを閉じます。(メッセージウインドウを含む)。

このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

電卓ウインドウを表示する(cMT シリーズ) :直接に固定した外見の電卓オブジェクトを呼び出すことができ、HMIで簡単な計算を実行し、Enterで目標の数値オブジェクトに出力できます。



本機能には[数値を出力した後、電卓を開いた状態に維持する]を使用することができます。下図をご参考ください。



キーボード入力

キーボードの入力信号に使われます。主は[数値]や[ワード]オブジェクトにキーボードで数値或いは文字を入力する必要がある場合に使われます。

Enter:キーボードの入力(Enter)と同じ。

Backspace:キーボードの後退して削除(Backspace)と同じ。

Clear:レジスタに入力されたデータを消去する。

Esc:[ウインドウを閉じる]機能と同じで、ポップアップされたウインドウを閉じることができる。

Delete:キーボードの削除(Delete)と同じで、カーソル右側の一個のワードを削除することができる。

Left:キーボードの←と同じで、カーソルを左側の一個のワードへ移動できる。

Right:キーボードの→と同じで、カーソルを右側の一個のワードへ移動できる。

Line feed:改行する。

Inc: 数値を+1 にする。

Dec: 数値を-1 にする。

ASCII/Unicode:キーボードの入力キャラクタを設定する。

マクロをトリガ ーする

本機能を選択すると、指定したマクロコマンドを実行します。本機能を選択する前には予めマクロコマンドを作成する必要があります。

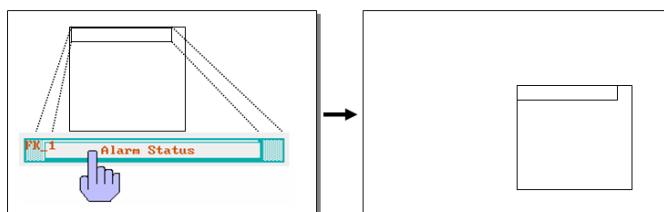
す。

マクロコマンドを作成するには、《18 マクロコマンド》をご参照ください。

<input checked="" type="radio"/> マクロをトリガーする	マクロ : [ID:000] macro_0
---	------------------------

ウインドウタイトルバー

ポップアップのウインドウにはウインドウタイトルバーがない場合、ウインドウを移動したかったら、当オブジェクトをワンクリックし、目的地に再度ワンクリックしたら、ウインドウは指定した場所に移動されます。



まずはタイトルバーを起動します。

そして目的地に再度ワンクリックしたら、ウインドウは指定した場所に移動されます。

USBディスク、SDカードまたはプリンターヘスクリーンのハードコピーを取る

本機能は現在の画面をプリントする機能です。本機能を選択するには、事前に[システムパラメータ設定] » [HMI 属性]で使用しているプリンタータイプを選ぶ必要があります。単色プリンターをご利用の場合、[グレースケール]を選択したらプリントの品質が向上できるが、文字の表示効果にも影響します。よりまして、文字のプリント効果を重視する場合、グレースケール機能を選択しないでください。

USBディスク、SDカードまたはプリンターヘスクリーンのハードコピーを取る

スクリーンのハードコピーを取る

出力 : HP PCLシリーズ(USB)

イメージを90度回転する

モード : カラー

*カスタマイズファイル名を使用したい場合、[PLCコントロール]オブジェクトを使用してください。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

全てのイベント(アラーム)を承認する(cMT シリーズ)

本ファンクションキーを押せば、一回で全てのイベントを承認することができます。

ユーザー情報をインポート/[USBセキュリ

高度安全のユーザー アカウント或いは E メールの連絡先をインポートします。USB セキュリティキーでログインすることもできます。

ティキー]を使 用



データ位置 : USB ディスク、SD カードから読み取るとの 2 箇所を選択できます。

アカウントインポートモード : [上書き]を選択したら、HMI には今回インポートされたアカウント資料のみ保存します。[付加]を選択したら、HMI 内のアカウント資料は保存され、それに今回インポートした新アカウント資料を加えます。

ユーザー アカウントをインポートした後、ファイルを削除 : USB ディスク内のユーザー アカウントをインポートしたらソースデータを削除し、情報漏れを防ぎます。

通知

本設定を使用すると、動作完了後には連帶に本項が指定したレジスタの状態を設定することができ、[ON]と[OFF]で設定する状態を設定します。

Note

- E メールの連絡先をインポートする場合、[上書き]の方式のみインポートされます。よりまして、既存の連絡先を全部削除したから、新規連絡先をインポートすることになります。

詳細情報は《6 ウィンドウ》、《12 キーボードのデザインと使用》、《36 管理者ツール》をご参照ください。

13.6. トグルスイッチ

13.6.1. 概要

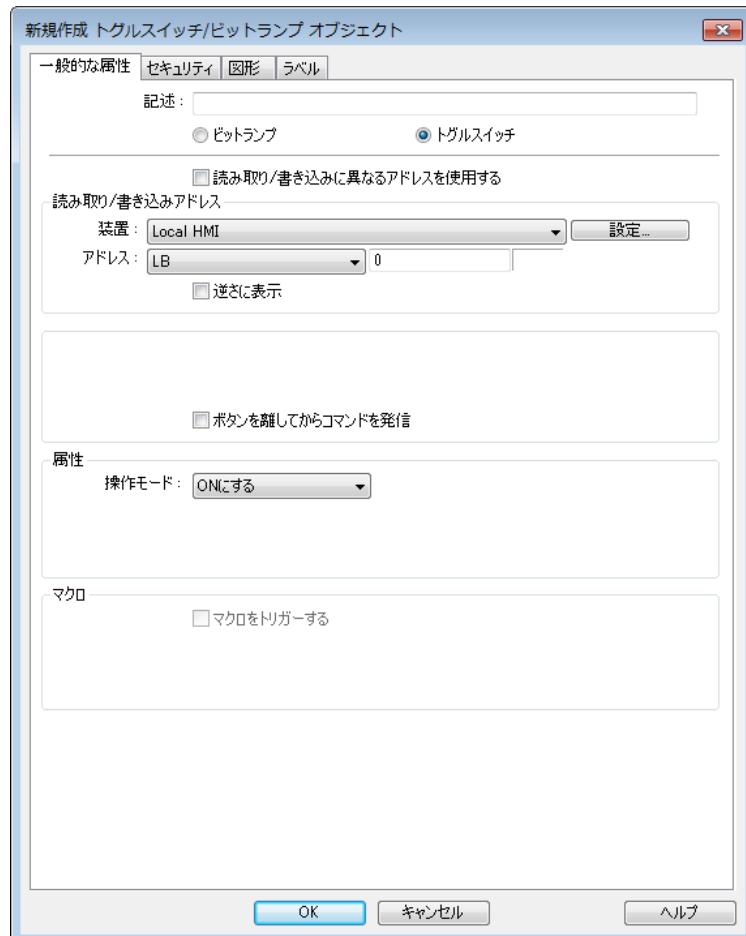
[トグルスイッチ]は[ビットランプ]オブジェクトと[ビット状態設定]オブジェクトを組み合わせたものです。本オブジェクトを押すと、指定したレジスタの状態を ON または OFF に設定できます。

13.6.2. 設定



ツールバー上の[オブジェクト]»[トグルスイッチ]ボタンを押せば、[トグルスイッチ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[トグルスイッチ]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述	
記述	<p>ユーザーはこのオブジェクトに関連情報を記述できます。</p> <p>ビットランプ/トグルスイッチ</p> <p>[ビットランプ]機能と相互に切り替えることができます。</p>	
読み取り/書き込みに異なるアドレスを使用する	<p>ユーザーはデータの読み取りアドレス及び書き込みアドレスを別々に設定することができます。</p>	
読み取りアドレス	<p>[設定]をクリックした後、ビットレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、トグルスイッチオブジェクトをコントロールします。[一般的な属性]タブでアドレスを設定することもできます。</p> <p>逆さに表示</p> <p>読み取りの状態を逆さに表示できます。例えば、ビットの実際状態は OFF だったが、[逆さに表示]にチェックマークを入れたら ON に表示します。</p> <p>[読み取り/書き込みに異なるアドレスを使用する]にチェックマークを入れていなかったら、読み取りアドレスフィールドの名前は[読み取り/書き込みアドレス]になります。</p>	
書き込みアドレス	<p>[設定]をクリックした後、ビットレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、トグルスイッチオブジェクトをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。</p> <p>ボタンを離してからコマンドを発信</p> <p>本設定を使用すると、オブジェクトを押した後、完全にボタンから離してから、オブジェクトが定義した操作モードが実行されます。もし本設定を使用しなかったら、一旦ボタンを押すと、オブジェクトはすぐに実行されます。復帰モードを選択したら、本機能をサポートしていません。</p>	
属性	スイッチタイプ	記述
	ON にする	このオブジェクトを押すと、指定したレジスタの状態が ON に設定されます。
	OFF にする	このオブジェクトを押すと、指定したレジスタ状態が OFF に設定されます。
	トグル	このオブジェクトを押すと、指定したレジスタの状態が逆さに表示されます。即ち ON→OFF, OFF→ON になります。

	モーメンタリモード	このオブジェクトを押すと、指定したレジスタの状態はまず ON に設定されたが、手を放せば、状態は OFF に設定されます。
マクロコマンド	[トグルスイッチ]オブジェクトはマクロコマンドを合わせて使用することができます。本機能を使用する前、マクロコマンドを作成する必要があります。	 マクロコマンドを作成するには、《18 マクロコマンド》をご参照ください。

13.7. マルチ状態切替スイッチ

13.7.1. 概要

[マルチ状態切替スイッチ]オブジェクトは[ワードランプ(マルチ状態ランプ)オブジェクトと[ワード設定(マルチ状態設定)]オブジェクトを組み合わせたものです。本オブジェクトはレジスタ内の数値で異なる状態を表示できる以外、ウインドウの中で一個のタッチャエリアを定義することができます。本オブジェクトを押すと、指定したレジスタ内の数値を設定できます。

13.7.2. 設定

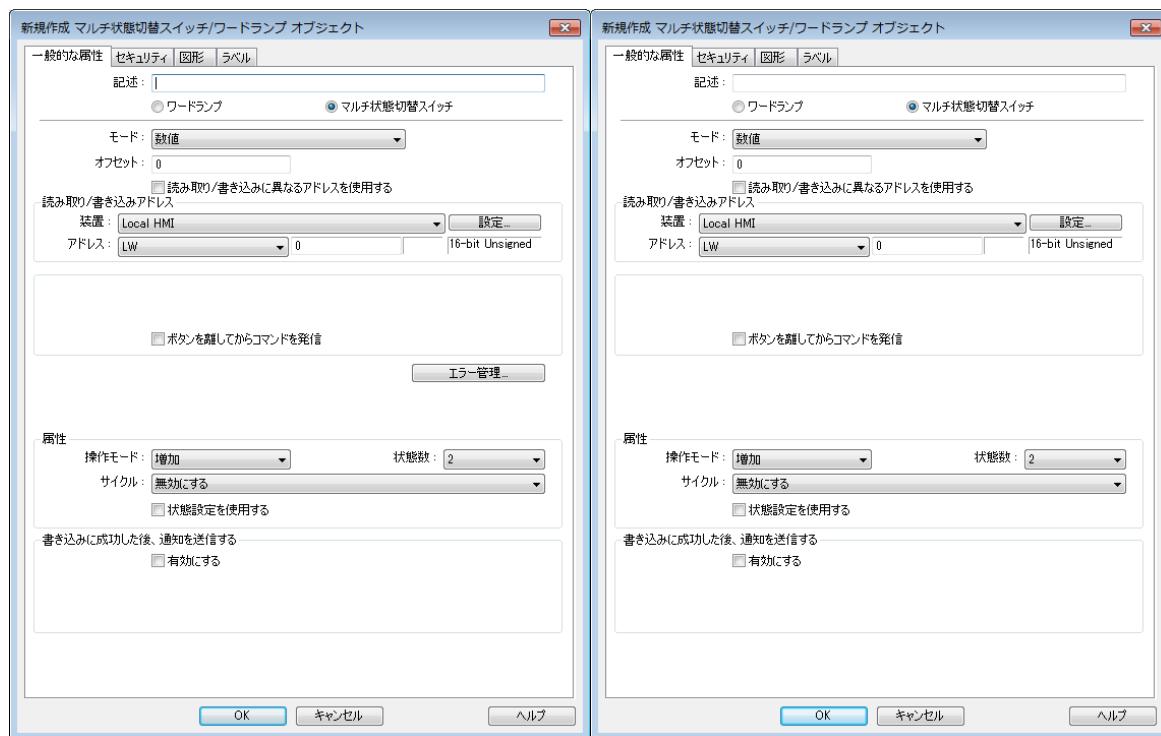


ツールバー上の[オブジェクト] » [マルチ状態切替スイッチ]ボタンを押せば、[マルチ状態切替スイッチ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[マルチ状態切替スイッチ]オブジェクトが新規作成されます。

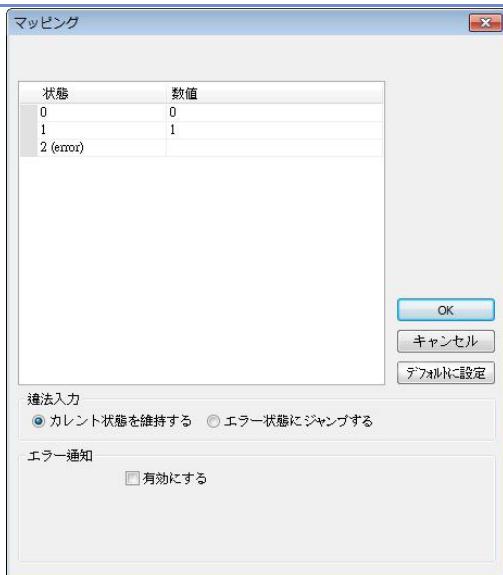
一般的な属性の設定

cMT シリーズ

eMT、iE、XE、mTV シリーズ



設定	記述
記述	<p>ユーザーはこのオブジェクトに関連情報を記述できます。</p> <p>ワードランプ/マルチ状態切替スイッチ</p> <p>[ワードランプ]機能と相互に切り替えることができます。</p>
モード/オフセット量	<p>異なる数値表示モードを提供します：数値、LSB。</p> <p> 詳細は《13.2 ワードランプ》をご参照ください。</p>
読み取り/書き込みに異なるアドレスを使用する	<p>ユーザーはデータの読み取りアドレス及び書き込みアドレスを別々に設定することができます。</p>
読み取りアドレス	<p>[設定]をクリックした後、ワードレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、マルチ状態切替スイッチをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。</p> <p>[読み取り/書き込みに異なるアドレスを使用する]にチェックマークを入れていなかったら、読み取りアドレスフィールドの名前は[読み取り/書き込みアドレス]になります。</p>
書き込みアドレス	<p>[設定]をクリックした後、ビットレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、マルチ状態切替スイッチオブジェクトをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。</p> <p>ボタンを離してからコマンドを発信</p> <p>本設定を使用すると、オブジェクトを押した後、完全にボタンから離してから、オブジェクトが定義した操作モードが実行されます。もし本設定を使用しなかったら、一旦ボタンを押すと、オブジェクトはすぐに実行されます。復帰モードを選択したら、本機能をサポートしていません。</p>
属性	<p>操作モード</p> <p>オブジェクトの動作モードを選択します。選択できるモードを下記例1をご参照ください。</p> <p>状態設定を使用する</p> <p>ユーザーは状態を対応する数値を変更できます。無効な数値が入力された際の対処動作を指定、及び指定したビットに状態を切り替えるように通知することもできます。</p>



カレント状態を維持する

範囲を超えた数値が入力された場合、マルチ状態切替スイッチは現在の状態を維持します。

エラー状態にジャンプする

範囲を超えた数値が入力された場合、マルチ状態切替スイッチはエラー状態に切り替えします。

エラー通知

無効な数値が入力された際に、自動的に指定したアドレスの状態を設定することができます。

書き込みに成功した後、通知を送信する

装置に書き込む動作に成功した後、指定したビットレジスタの状態をオン/オフに設定します。

エラー管理 (cMT シリーズ)

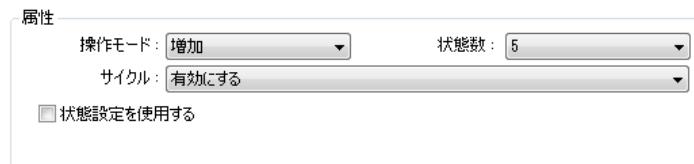
違法な数値が入力された場合、対処動作や指定したビットアドレスの切替状態を設定します。用途は[状態設定を使用する]と似ているが、各状態を対応する数値を設定する必要がありません。

例 1

選択できるモードは以下のとおりです：

- 増加(JOG+)

数値を増加する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値を+1 します。但し、増加した結果は[上限値]の設定値を超えるかもしれません。もし“サイクル”を[使用]すれば、最大状態に達すると、また最小状態 0 に復帰します。下図を例として、操作モードを[増加]を選択し、状態数は 5 でサイクルを[使用]すると、このオブジェクトを一回押すごとに、状態は状態 0 から+1 し続けて、状態 4([状態数]-1)までに達した後、また状態 0 に復帰し、改めて増加します。



● 減少(JOG-)

数値を減少する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値が 0 になるまで -1 にし続けます。もし“サイクル”を[使用]すれば、最小状態に達すると、また最大状態に復帰します。下図を例として、操作モードを[減少]を選択し、状態数は 5 でサイクルを[使用]すると、このオブジェクトを一回押すごとに、状態は状態 0 になるまで -1 にし続け、そしてまた状態 4([状態数]-1) に復帰し、改めて減少します。



13.8. スライドスイッチ

13.8.1. 概要

[スライドスイッチ]は一箇所のスライドエリアを作成し、その中に数値を表示、またはスライド軌跡で指定したレジスタ内の数値を変更する機能を提供するオブジェクトです。

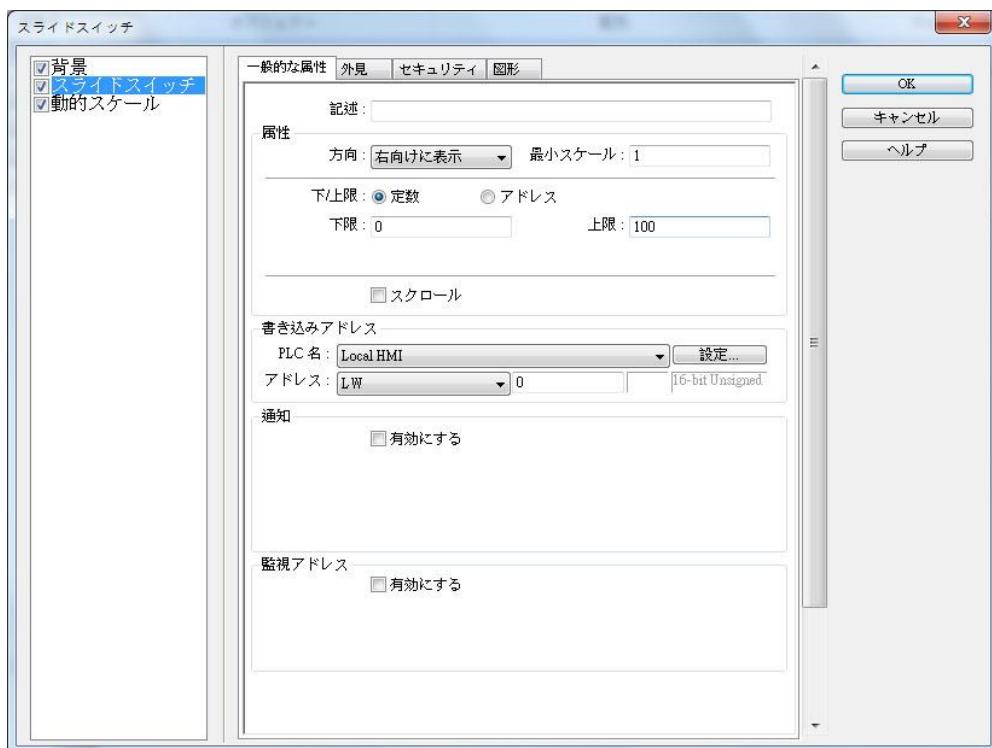
13.8.2. 設定



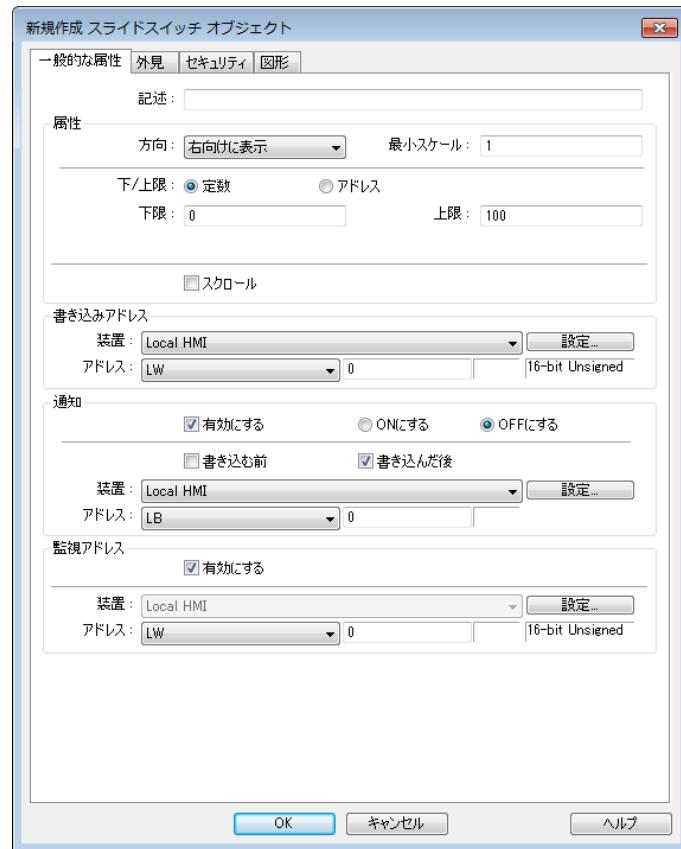
ツールバー上の[オブジェクト]»[スライドスイッチ]ボタンを押せば、[スライドスイッチ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[スライドスイッチ]オブジェクトが新規作成されます。

一般的な属性の設定

cMT シリーズ



eMT、iE、XE、mTV シリーズ



設定	記述
属性	方向 スライドスイッチオブジェクトは四方向向けで表示できます(右向 けで表示、上向けに表示、左向けに表示、下向けに表示)。
	最小スケール 記入した最小スケール値で表示します。例えば:[最小スケール]を 10 にした場合、数値は毎回 10 の倍数で表示します。
定数	直接にワードレジスタの上下限定数値を設定できます。例えば:[下 限]を 5、[上限]を 100 に設定した場合、設定できる数値範囲は 5~100 になります。
アドレス	

上下限は指定したレジスタで設定でき、下記例 1 をご参考ください。

スクロール

[最小スケール]がスライドスイッチを引きすることで数値を変更する異なり、ただ[スライドスイッチ]オブジェクトをタッチすれば、[スクロール値]の設定によって数値を増加/減少します。

書き込みアドレス

[設定]をクリックした後、ワードレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、スライドスイッチオブジェクトをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。

通知

本設定を使用すると、手動操作モードを使用する際に、動作が完了されたら、本項が指定したレジスタの状態を設定できます。[ON]と[OFF]で設定する状態を選択します。

[設定]をクリックした後、ビットレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、通知の設定をコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。

[書き込む前]/[書き込んだ後]

書き込みの前/後で指定したレジスタの状態を設定します。

監視アドレス

スライドロックをドラッグ＆ドロップされた際に、即時に現在の書き込みアドレスの設定値を表示します。

例 1

上下限は指定したレジスタで設定します。書き込みアドレスは LW-n の場合、上/下限は以下の規則で自動的に設定されます：

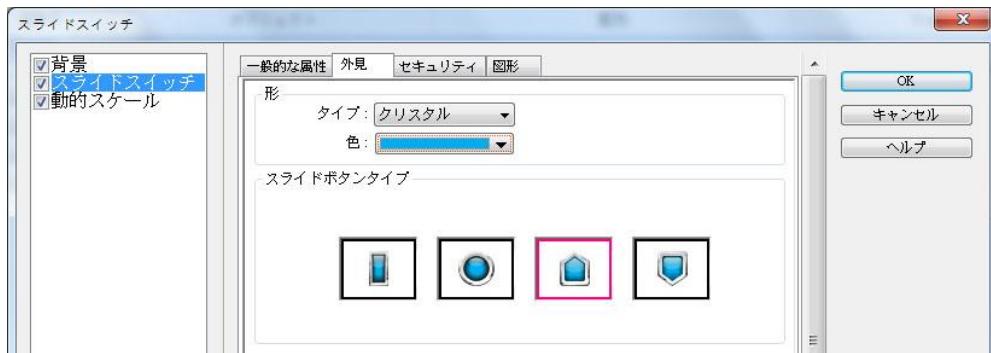
アドレス フォーマット	16-bit	32-bit	64-bit (cMT のみ)
アドレス	LW-n	LW-n	LW-n
下限	LW-n	LW-n	LW-n
上限	LW-n+1	LW-n+2	LW-n+4

下表を例として、[レジスタアドレス]は LW-100 の場合、上/下限のアドレスは自動的に以下のとおりに設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMT のみ)
アドレス	LW-100	LW-100	LW-100
下限	LW-100	LW-100	LW-100
上限	LW-101	LW-102	LW-104

外見設定

cMT シリーズ



eMT、iE、XE、mTV シリーズ



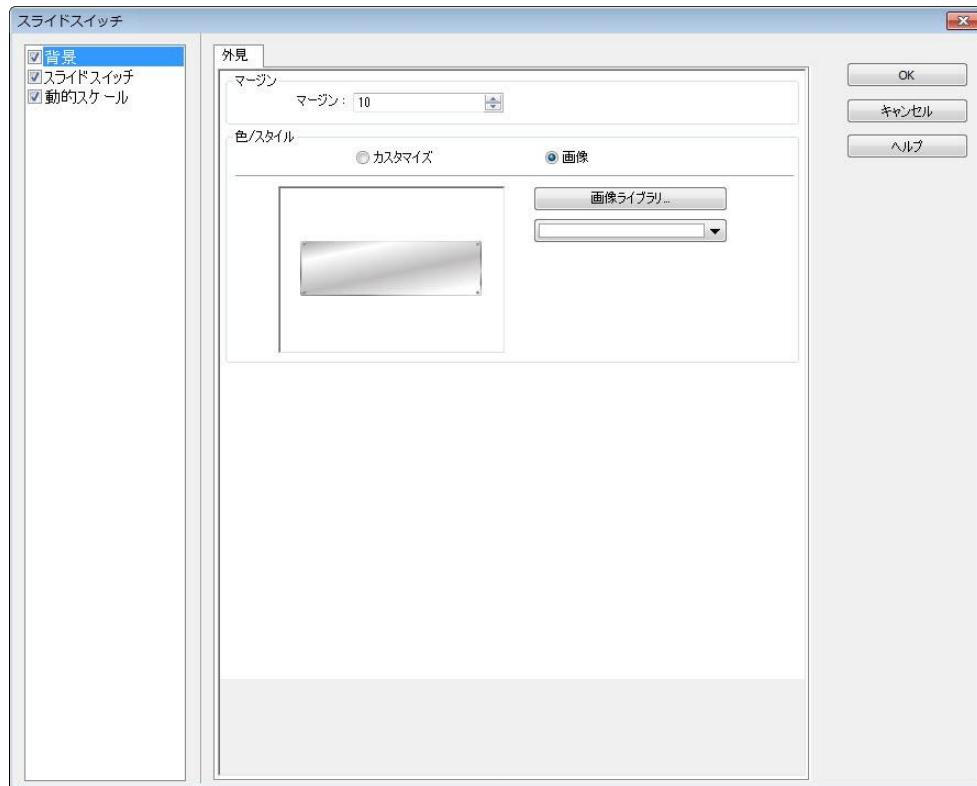
設定	記述
スタイル	スライドボタンのスタイルを選択します。 デフォルトを使用すると、画像ライブラリの画像をスライドの表示図にすることができます。
スライドボタン	システムには合計 4 種類のデフォルトスライドボタンスタイルを提供しています。スライドボタンの幅、枠、背景とスライド軌道の色も調整できます。

☞ 動的スケールは《Ch13.45 動的スケール》をご参照ください。

13.8.3. 複合型設定

cMT シリーズにはスライドスイッチの複合型設定を提供し、一回で関連する複数のオブジェクトを設定することができます。スライドスイッチには[背景]、[スライドスイッチ]、[動的スケール]を同時に設定することができます。

背景の設定



設定

記述

マージン

背景の縁とオブジェクトの間の距離を設定します。

色/スタイル

カスタマイズ



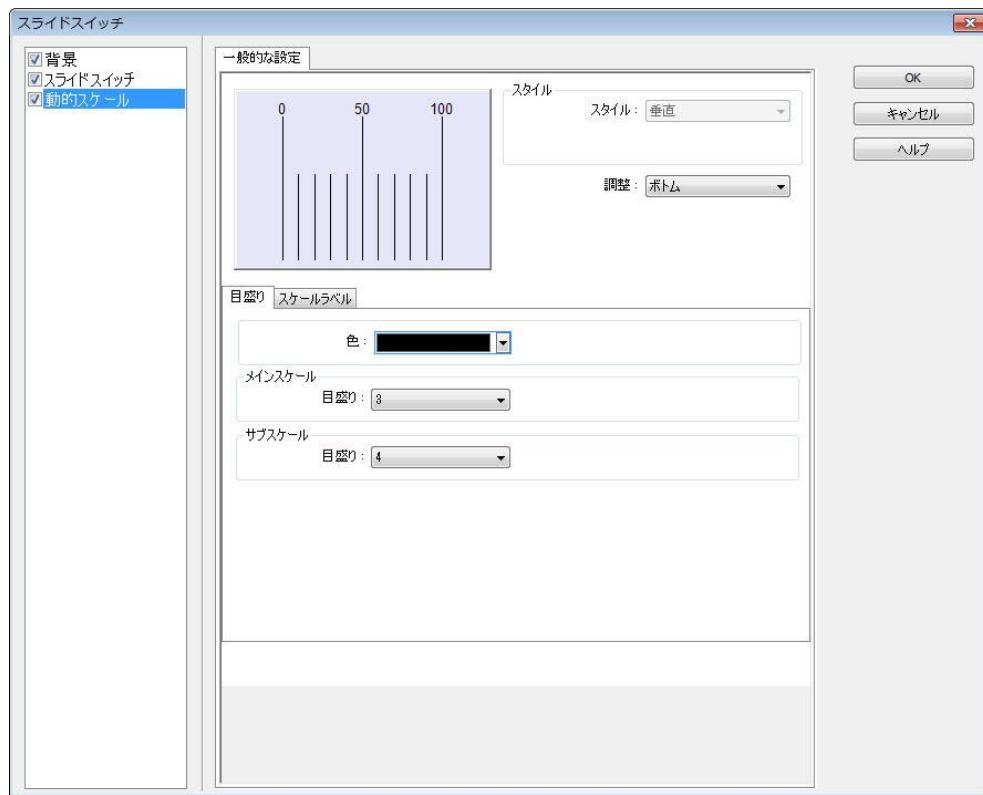
模様スタイルと色を選択して、背景をカスタマイズします。

画像



内蔵の背景画像を使用するか、画像ライブラリ内の画像を使用してもいいです。

動的スケールの設定



設定	記述
スタイル	スライドスイッチの設定に従います。
揃え	スケールがスライドスイッチに表示する位置を設定します。
目盛り	[メインスケール]と[サブスケール]の分割数及び色を設定します。
スケールラベル	スケールラベルのフォント、色、サイズと他の属性を設定します。 [スライドスイッチ]で上下限を[アドレス]に設定すると、[動的スケール]の[上下限をレジスタから取得]は自動的に設定されます。[スライドスイッチ]の[方向]を[左向けに表示]に設定すると、[動的スケール]の[変換(左揃え/右揃え)]は自動的に設定されます。



13.9. 数値

13.9.1. 概要

[数値]は指定したワードレジスタ内の数値を入力または表示するオブジェクトです。

13.9.2. 設定

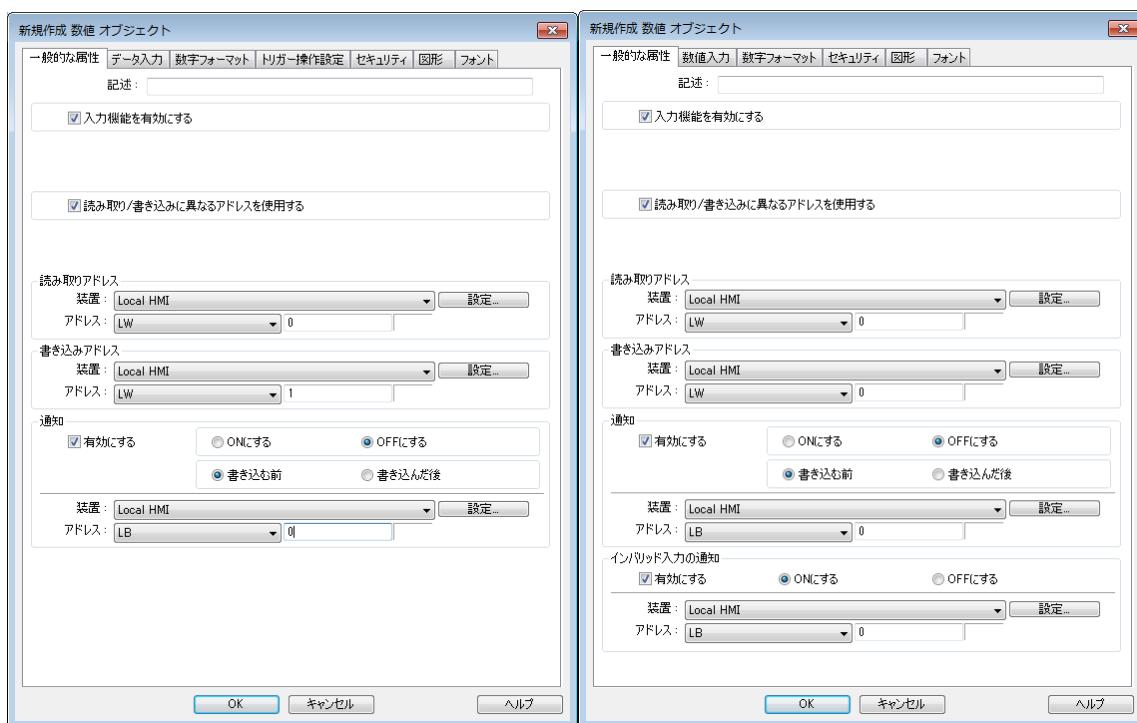


ツールバー上の[オブジェクト]»[数値]ボタンを押せば、[数値]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[数値]オブジェクトが新規作成されます。

一般的な属性の設定

cMT シリーズ

eMT、iE、XE、mTV シリーズ

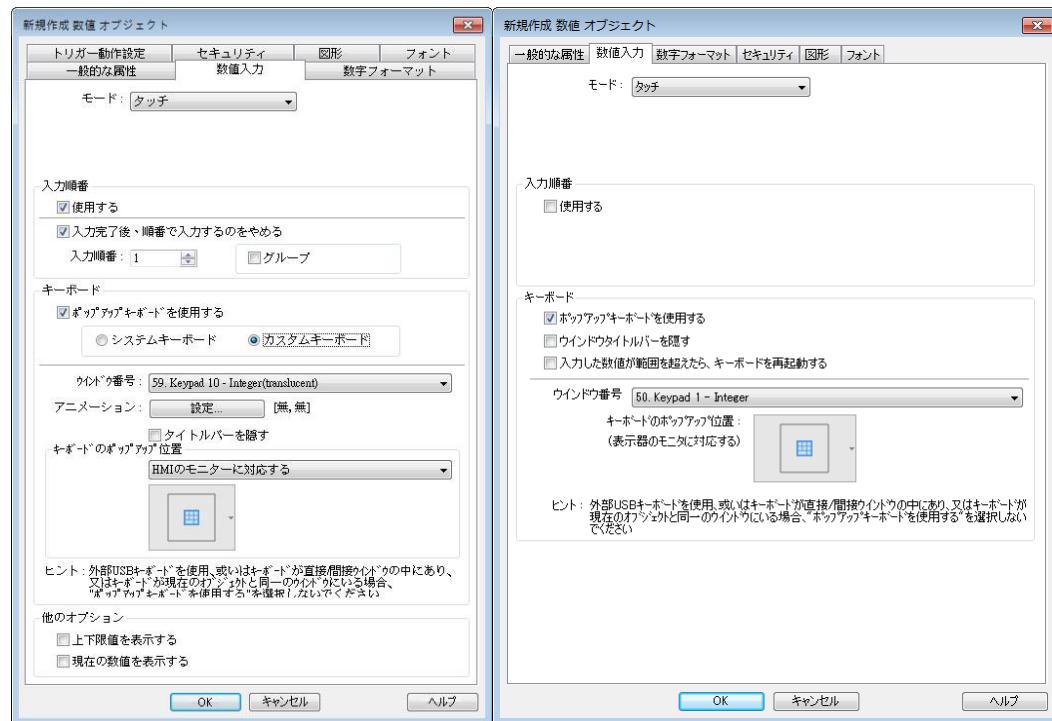


設定	記述
入力機能を有効にする	本項にチェックマークを入れたら、入力機能と関連する属性設定ウインドウがオープンされます。
読み取り/書き込みに異なるアドレスを使用する	ユーザーは数値の読み取りアドレスと、書き込みアドレスを別々で設定することができます。
読み取りアドレス	[設定]をクリックし、ワードレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択して数値を表示します。[アドレスタグライブラリ]の定義されたアドレスタグを使用してもいいです。 [読み取り/書き込みに異なるアドレスを使用する]にチェックマークを入れていなかったら、読み取りアドレスフィールドの名前は[読み取り/書き込みアドレス]になります。
書き込みアドレス	ワード関連の[装置]、[装置タイプ]、[アドレス]を選択し、数値の書き込み先とします。
通知	本設定を使用すると、動作完了前/後では連帶的に本項が指定したレジスタ状態を設定することができます。[ON]と[OFF]で設定する状態を選択します。 書き込む前/書き込んだ後 書き込む前/後で指定したレジスタの状態を設定します。
インバリッド入力の通知	無効な数値を入力した場合、自動的に指定したレジスタの状態を[ON]または[OFF]に設定します。

数値入力設定

cMT シリーズ

eMT、iE、XE、mTV シリーズ



設定

記述

モード

タッチ

オブジェクトをタッチすることで入力プロセスを起動します。

ビットコントロール

指定したビットレジスタの ON/OFF で入力プロセスを起動/終了します。

ビットアドレスの入力を許可

起動及び終了のビットレジスタアドレスを指定します。入力順番は[入力順番]の設定に従わなければなりません。入力するには、外部取付 USB キーボードを使用する必要があり、スクリーンタッチキーボードを使用してはいけません。

cMT-SVR は cMT Viewer の内蔵キーボードを使用します。

入力順番

入力順番及び入力順番グループを設定し、複数の入力オブジェクトを連続に入力することができるようになります。

使用基準：

- 入力順番の範囲:1~511。グループ範囲:1~15。
- [グループ]にチェックマークを入れない場合、

入力順番グループは 0 です。

- システムは同じ入力順番グループの入力オブジェクトだけを探します。
- より小さい入力順番に属する数値は、入力する順番がより先で、逆の場合はより後ろになります。
- 複数のオブジェクトに、同じ入力順番グループと入力順番が同時に有する場合、より下のレイヤーにあるオブジェクトは優先に入力されます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

キーボード

ポップアップキーボード

チェックマークを入れる：ウインドウ及びポップアップの位置を指定します。入力を起動する時に、システムは指定した位置でキーボードウインドウをポップアップし、入力終了の際に閉じます。

チェックマークを入れない：入力する時、システムはキーボードウインドウをポップアップしないので、下記方法で入力する必要があります：

- 自分でウインドウにキーボードをデザインする。
- 外部取付キーボードを使用する。

システムキーボード

cMT シリーズを使用する時、cMT のデフォルトキーボードを使用できます。自らキーボードウインドウをデザインする必要がありません。

アニメーション設定

cMT シリーズを使用してそれに[カスタムキーボード]を使用する場合、キーボードウインドウをポップアップする時のアニメーション効果を設定できます。



《Ch13.5 ファンクションキー》をご参照ください。

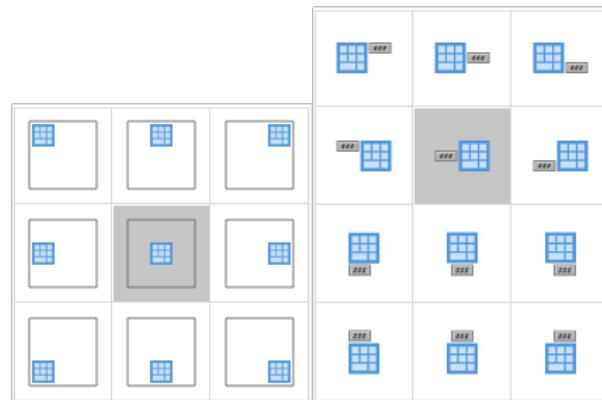
ウインドウタイトルバーを隠す

数値/ワードキーボードはウインドウタイトルバーを使用しないと選択できます。

キーボードポップアップ位置

cMT シリーズを使用し、それに[カスタムキーボード]を選択する場合、キーボードをポップアップする位置を HMI モニターに対応するか、またはオブジェクトに対応するかを指定できます。

HMI モニターに対応する オブジェクトに対応する



入力した数値が範囲を超えたたら、キーボードを再起動する

入力オブジェクトを使用した際に、入力した数値が設定した範囲を超えた場合、システムが自動的にキーボードを再起動します。

他のオプション

(cMT シリーズに適しています)

上下限値を表示する

本項にチェックマークを入れると、数値を入力する時、オブジェクトの傍に当オブジェクトアドレスの上下限値が表示されます。



前の数値を表示する

本項にチェックマークを入れると、数値を入力する時、オブジェクトの傍に当オブジェクトアドレスが変更された前の数値が表示されます。



現在のウインドウにキーボードを組み込むには、《12 キーボードのデザインと使用》をご参考ください。



このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

例 1

グループ式数値オブジェクトをデザインします。

入力順番及び入力順番グループを設定することで、複数の入力オブジェクトへ連続に入力することができます。現在のオブジェクトへの入力を完了後、自動的に同一グループでの次ぎの入力オブジェクトに移します。

- 三つの数値オブジェクトを作成し、全部[入力順番]を使用します。順番は順番 1、順番 2、順番 3 に分かれ、[グループ 1]に設定します。入力順番は下図の通りです：

LW-0

入力順番

有効にする

入力完了後、順番で入力するのをやめる

入力順番 : 1 グループ 1

LW-1

入力順番

有効にする

入力完了後、順番で入力するのをやめる

入力順番 : 2 グループ 1

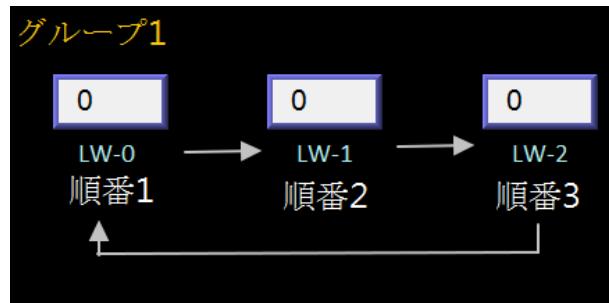
LW-2

入力順番

有効にする

入力完了後、順番で入力するのをやめる

入力順番 : 3 グループ 1



- 最後の数値オブジェクトを入力完了後、キー ボードのポップアップを停止し、全ての入力動作を終了したい場合、[入力完了後、順番で入力するのをやめる]にチェックマークを入れればいいです。

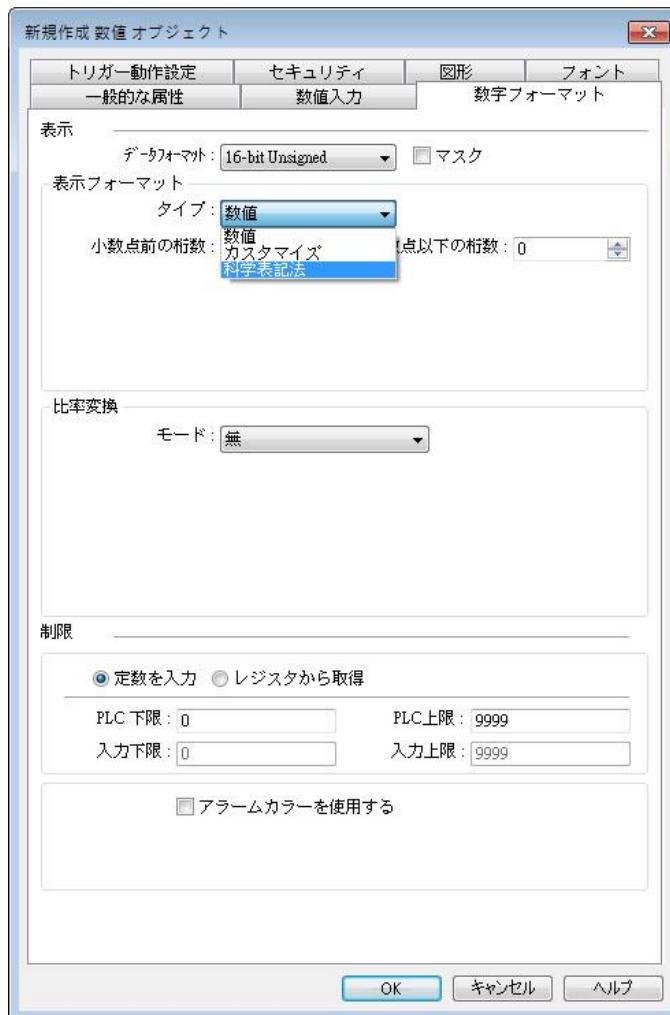
入力順番

有効にする

入力完了後、順番で入力するのをやめる

入力順番 : 3 グループ 1

フォーマットの設定



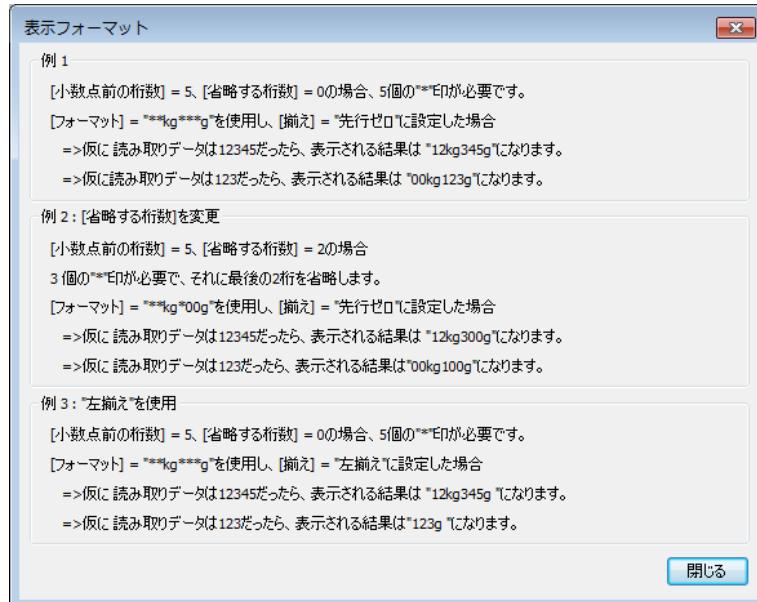
設定	記述
データフォーマット	二進化十進表現(BCD)、二進数(Binary)、符号つき二進数/符号なし二進数(Signed/Unsigned)、十六進数(Hex)、浮動小数点数(Float)の読み取りと書き込みをサポートします。16-bit フォーマットを使用する場合、レジスタ内のワードを 1 個使います。32-bit フォーマットを使用する場合、ワードを 2 個使います。
表示フォーマット	「*」で全ての数字を代わって表示します。 [フォーマット]の「*」符号とは、数値オブジェクトでの位を示しています。数値以外、[フォーマット]にも付加の文字を入力することができ、例えば“kg”などです。文字揃えには[左揃え]、[中央揃え]、[右揃え]と[先行ゼロ]が選択できます。数値オブジェクトで表示する桁は、「*」の数量によって上位 → 下位の順次で表示します。

省略する桁数

表示しない桁を設定することができます。下位→上位の順次で省略します。

必要する“*”印の数量=[小数点前の桁数]-[省略する桁数]

下記例をご参照ください：



[先行ゼロ]を使用しない場合、"*"に囲まれた文字が省略されます。

例えば：

[小数点前の桁数] = 5、[省略する桁数] = 0

表示フォーマットは"Total=**kg***g"です。

読み取ったデータが 255 の場合、"Total=255g"に表示されます。

読み取ったデータが 1000 の場合、"Total=1kg000g"に表示されます。

科学表記法

科学表記法で数値を表示します。例えば：100 を 1e+2 で表示します。BCD、HEX、Binary は本フォーマットにサポートされていません。

数字ビット

小数点前の桁数

数

小数点の前に表示する桁数です。

小数点以下の桁数

小数点の後に表示する桁数です。

比率変換

内挿法

表示する数値はレジスタ内の原始数値を利用し、換算した後得たものです。

本機能を選択するには[比率最小値]、[比率最大値]と[制限]項目の[入力下限]、[入力上限]を設定する必要があります。下記例 2 をご参照ください。

テスト：

設定した変換比率が正確かどうかをテストします。下記例 2 をご参照ください。

動的比率：

変換の上下限は指定したレジスタで設定します。下記例 4 をご参照ください。

マクロサブルーチン

オブジェクトの読み取り/書き込みデータはマクロ関数ライブラリでのサブルーチンの運算結果で得たものです。

読み取り変換：オブジェクトが読み取った数値はマクロでの運算を経てから表示します。

書き込み変換：オブジェクトに書き込んだ数値はマクロでの運算を経てから表示します。

本機能を使用するには、《13.9.2.1 数値オブジェクトにマクロサブルーチンを使用する規則》をご参照ください。

制限

入力数値の上下限のソースを設定します。アラームカラーとアラーム効果を設定することができます。

定数を入力

数値を入力する上下限の[装置下限]と[装置上限]の限界を設定します。入力した数値が上下限の範囲にいない場合、レジスタ内の数値を変更できません。

レジスタで設定

上下限はレジスタで設定することができます。下記例 5 を参考してください。

アラームカラーを使用する

下限

レジスタ内の数値が下限値より小さい場合、オブジェクトはこの色で数値を表示します。

上限

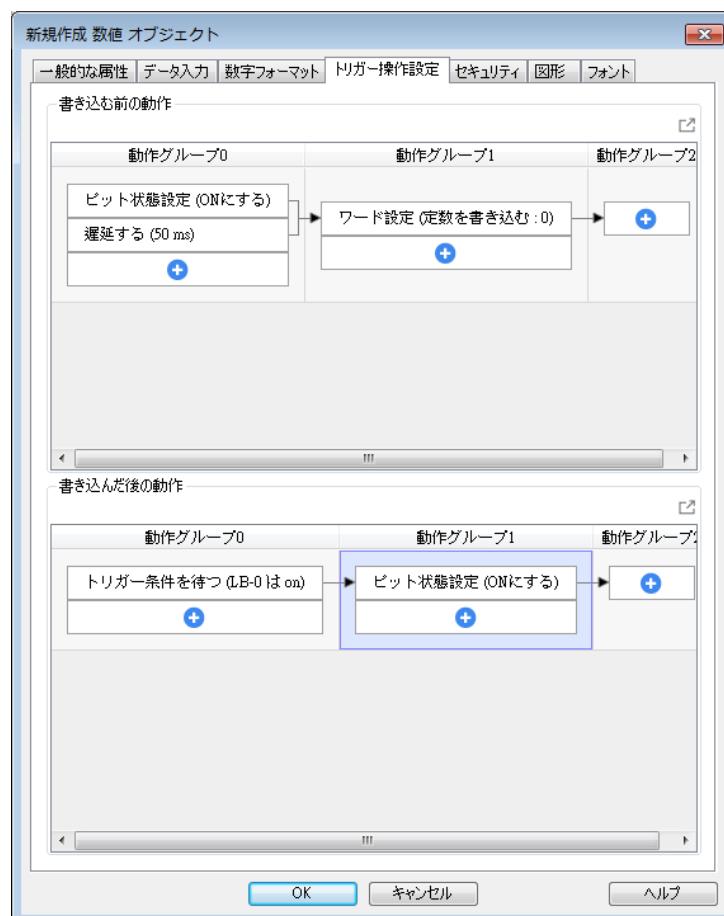
レジスタ内の数値が上限値より大きい場合、オブジェクトはこの色で数値を表示します。

リンク

レジスタ内の数値が下限より小さい、または上限より大きい場合、オブジェクトは点滅してユーザーを注意します。

トリガー動作設定 (cMT にのみサポートされる)

本設定を使用すると、動作が完了前・後でグループごとに一連の動作を実行させることができます。動作を設定する時に、事前に動作をグループ分けにすれば、同一グループ内の動作は一緒に実行されます。前のグループ内の動作が全てトリガーされた場合のみ、次のグループの動作が実行されます。[一般的な属性]設定タブの通知機能に比べ、[トリガー動作設定]はより多様で柔軟なデザイン方法を提供します。本設定に関する詳細は、Ch13.53をご参照ください。



設定	記述
書き込む前の動作	一旦書き込みが始まったら(書き込みがまだ完了していない)、一連の動作が実行されます。
書き込んだ後の動作	書き込みが完了した後、一連の動作が実行されます。

13.9.2.1. 数値オブジェクトにマクロサブルーチンを使用する規則

- 戻り値が必要で、それで一個のパラメータのみ許可されます。

例えば：

sub char test(short a)// (正確)

sub test(char a)// (エラー、戻り値がない)

```
sub char test(char a, char b)//(エラー、二個のパラメータがある)
```

- 数値オブジェクトのデータフォーマットは特定したパラメータタイプに対応します。下表のように示します：

マクロパラメータ タイプ	数値オブジェクトのデータフォーマット
short	16-bit Signed
Int	32-bit Signed
Unsigned short	16-bit BCD, 16-bit HEX, 16-bit Binary, 16-bit Unsigned
unsigned int	32-bit BCD, 32-bit HEX, 32-bit Binary, 32-bit Unsigned
float	32-bit Float
long	64-bit Signed(cMT にのみサポートされる)
unsigned long	64-bit Unsigned(cMT にのみサポートされる)
double	64-bit Double(cMT にのみサポートされる)

仮にある数値オブジェクトのデータフォーマットは 16-bit Unsigned の場合、パラメータタイプが unsigned short であるマクロサブルーチンのみ選択できます。例えば：

```
sub char test(unsigned short a)//(正確)
```

```
sub char test(char a)//(エラー)
```

- ローカル HMI のアドレスのみアクセスできます。

例えば：

```
Get Data(var, "Local HMI", LB, 0, 1)//(正確)
```

```
Get Data(var, "MODBUS RTU", 0x, 0, 1)//(エラー)
```

- 下記関数を呼び出すことができません：

ASYNC_TRIG_MACRO, SYNC_TRIG_MACRO, DELAY, FindDataSamplingDate, FindDataSamplingIndex, FindEventLogDate, FindEventLogIndex, IMPORT, IMPORT2, OUTPORT, PURGE, TRACE

- 下記ステートメントを使用することができません：

For-Next, While-Wend

例 2

比率変換に[内挿入法]公式を使用する場合は下記になります：

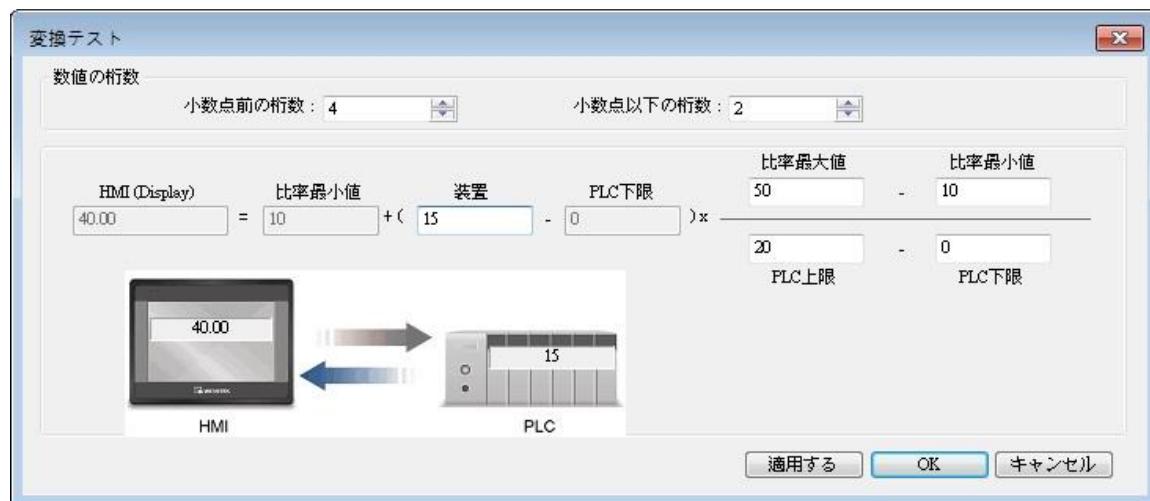
$$\text{新数値} = \frac{\text{比率最小値} + (\text{原数値} - PLC\text{下限}) \times \frac{\text{比率最大値} - \text{比率最小値}}{PLC\text{上限} - PLC\text{下限}}}{PLC\text{上限} - PLC\text{下限}}$$

下図を例として、元の数値が 15 の場合、換算した結果は 40 です。



[テスト]

テスト機能で設定した変換比率が正確なのかをテストします。テストしたい数値を[PLC]欄に入力します。下図を例として、仮に 15 を入力した場合、比率変換後には計算結果 40 が得られました。



例 3

数値オブジェクトが Float 以外の変数で小数点を使用した場合、内挿法で計算した結果は小数点の桁数を自動的に調整しません。下記説明をご参考ください：

- 二個の数値オブジェクトを作成します、その中の一個の数値オブジェクトに内挿法を使用し、それに小数点以下 1 桁を表示するように設定します。



2. 数値“123”を内挿法を使用していないオブジェクトに入力したら、もう一つのオブジェクトで表示されるのは“246.0”で、“24.6”ではありません。

原始数値 123

小数点以下の桁数+変換(Conversion) 246.0

3. 数値を左に一位を移したら、下図のように内挿法の比率最大値を調整する必要があります。



原始数値 123

小数点以下の桁数+変換(Conversion) 24.6

例 4

比率変換に[内挿法]を使用した時、比率最小値/比率最大値は指定したレジスタで設定します。書き込みアドレスは LW-n の場合、比率最小値/比率最大値は以下のとおり自動的に設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMTのみ)
動的比率アド レス	LW-n	LW-n	LW-n
比率最小値	LW-n	LW-n	LW-n
比率最大値	LW-n+1	LW-n+2	LW-n+4

下表を例として、[動的比率アドレス]は LW-100 の場合、比率最小値/比率最大値アドレスは自動的に下記のように設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMTのみ)
動的比率アド レス	LW-100	LW-100	LW-100
比率最小値	LW-100	LW-100	LW-100
比率最大値	LW-101	LW-102	LW-104

例 5

上下限は指定したレジスタで設定できます。書き込みアドレスは LW-n の場合、上/下限は下記の規則によって自動的に設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMTのみ)
レジスタアド レス	LW-n	LW-n	LW-n
下限	LW-n	LW-n	LW-n
上限	LW-n+1	LW-n+2	LW-n+4

下表を例として、[レジスタアドレス]は LW-100 の場合、上/下限のアドレスは自動的に下記のように設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMTのみ)
レジスタアド レス	LW-100	LW-100	LW-100
下限	LW-100	LW-100	LW-100
上限	LW-101	LW-102	LW-104

例 6

本例は[数値]オブジェクトがどのように[マクロサブルーチン]を使用し、比率の変換を実行するのかを説明します。

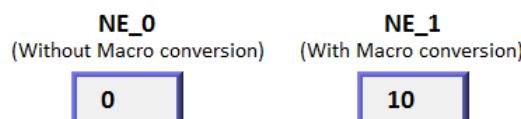
仮に下記のように、二個のマクロがある場合：一つは[読み取り変換]を実行し、もう一つは[書き込み変換]を実行します。

Read Conversion	Write Conversion
<pre>sub short read(short a) short b b = a + 10 return b end sub</pre>	<pre>sub short write(short a) short b b = a - 10 return b end sub</pre>

- 二個のコントロールアドレスが同じの[数値]オブジェクト NE_0 及び NE_1 を作成し、NE_1 に[マクロサブルーチン]を用いて読み取り/書き込みの比率変換を実行します。

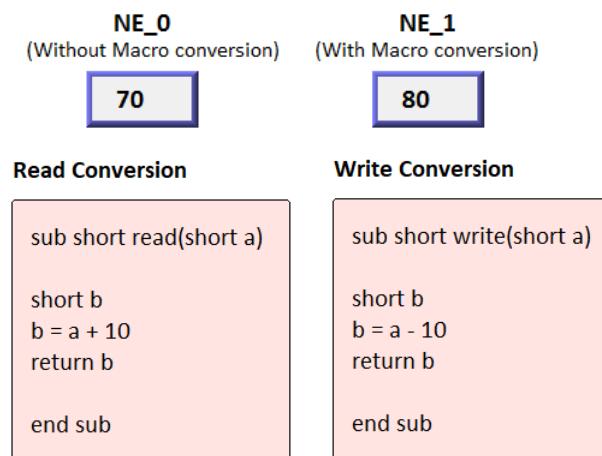


- NE_0 に 0 を記入した場合、NE_1 は[読み取り変換]を実行し、10 に換算されます。



Read Conversion	Write Conversion
<pre>sub short read(short a) short b b = a + 10 return b end sub</pre>	<pre>sub short write(short a) short b b = a - 10 return b end sub</pre>

3. NE_1 に 80 を記入した場合、[書き込み変換]が実行され、換算した結果が 70 で、NE_0 は 70 を表示します。



Note

- 一個の数値オブジェクトが同時に[読み取り変換]及び[書き込み変換]を実行する場合、このオブジェクトに 1 個の数値を入力したら、オブジェクトで表示する数値はまず[書き込み変換]を実行し、そして換算後の数値で[読み取り変換]を実行します。本例で、[書き込み変換]を $b=a-20$ に設定した場合、NE_1 に 80 を記入した後、まず[書き込み変換]が実行され、戻り値 60 を送信してから、[読み取り変換]を実行し、数値 70 を表示します。

セキュリティ



設定

インターロック機能を使用する

記述

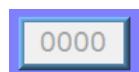
本項にチェックマークを入れ、そして[ワード]を選択すると、このオブジェクトを操作できるかどうかのは、一個の指定したワードアドレスの[状態]によります。上図に示されたように、LW-1 の数値が 1 より大きい場合のみ、本オブジェクトを操作できます。

無効の場合、オブジェクトを隠す

指定したワードアドレス内の数値が[状態]に合致したい場合、オブジェクトは隠されます。

無効の場合、グレースケールで文字を表示する

オブジェクトの数値は指定したワードアドレス内の数値が[状態]に合致していない場合に、グレースケールで表示されます。



状態

指定したワードアドレスをトリガーする状態を設定します。トリガー条件は、>、<、==、<>、>= 或いは <= が設定できます。その中、== と <> は[許容誤差]の設定に用いられます。

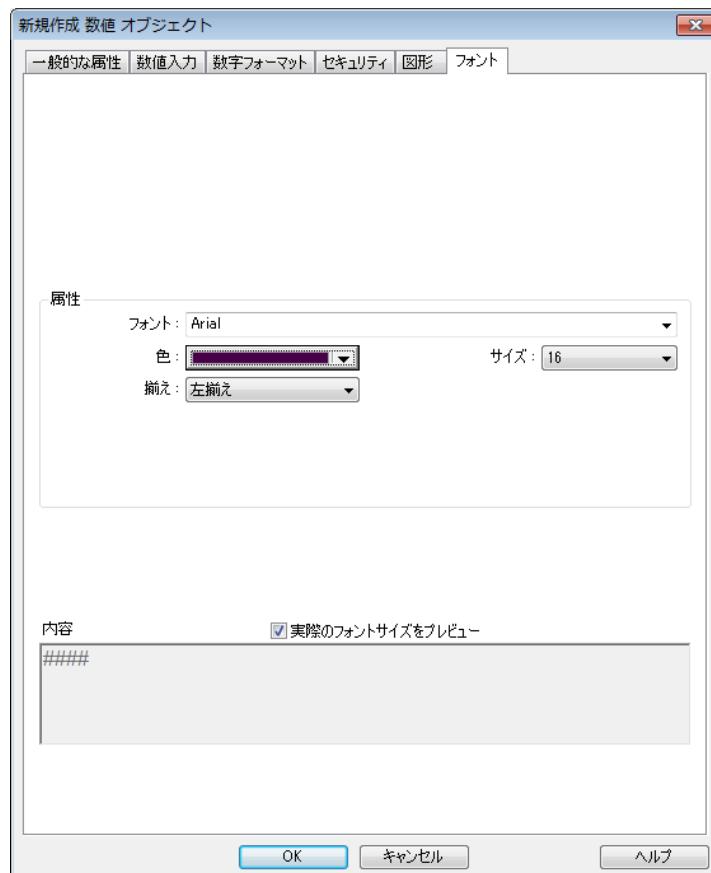
例えば：

装置:	Local HMI	設定...
アドレス:	LW	0
状態:	==	10
許容誤差:	1	

動作:

指定したワードアドレスの数値が大きいイコール 11、或いは小さいイコール 9 の場合、オブジェクトが無効されて隠されます。

フォント設定



設定

色

揃え

記述

数値が上下限の範囲内にある時、この色で表示します。

左揃え：数値を左揃えにします。

中央揃え：数値を中央揃えにします。

右揃え：数値を右揃えにします。

先行ゼロ：数値が設定した位に未満の場合、数値の前にはゼロが先行します。

左揃え

中央揃え

右揃え

先行ゼロ

サイズ

文字のサイズを決めます。

13.10. アスキーオブジェクト

13.10.1. 概要

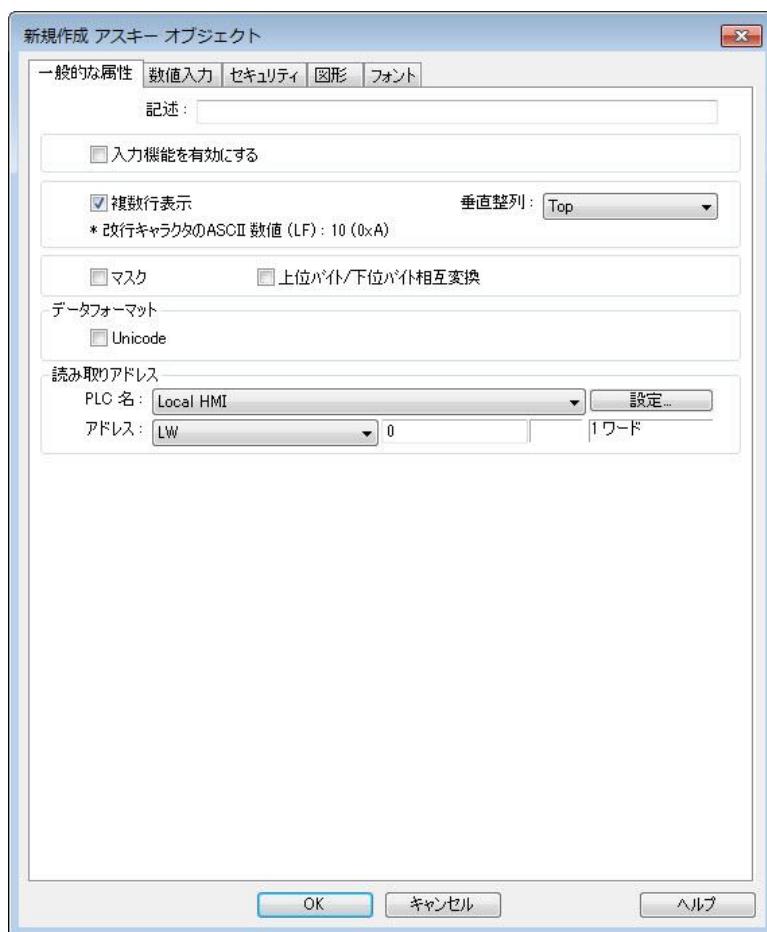
[アスキーオブジェクト]は ASCII コードで指定したレジスタ内のデータを表示するオブジェクトです。

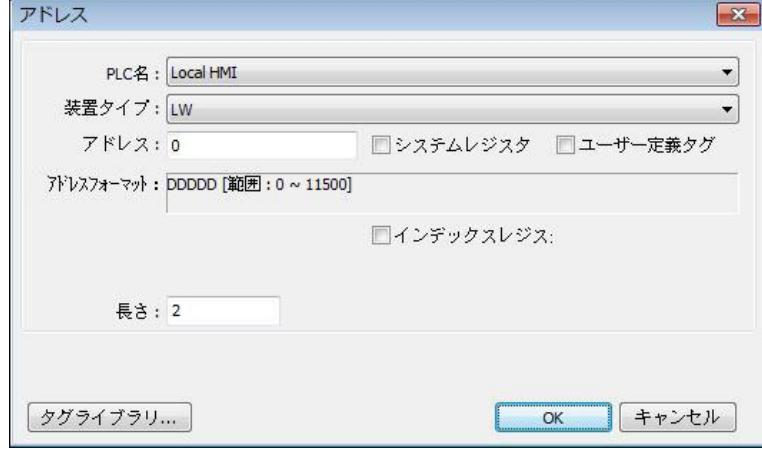
13.10.2. 設定



ツールバー上の[オブジェクト]»[ASCII]ボタンを押せば、[アスキーオブジェクト]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[アスキーオブジェクト]オブジェクトが新規作成されます。

一般的な属性の設定

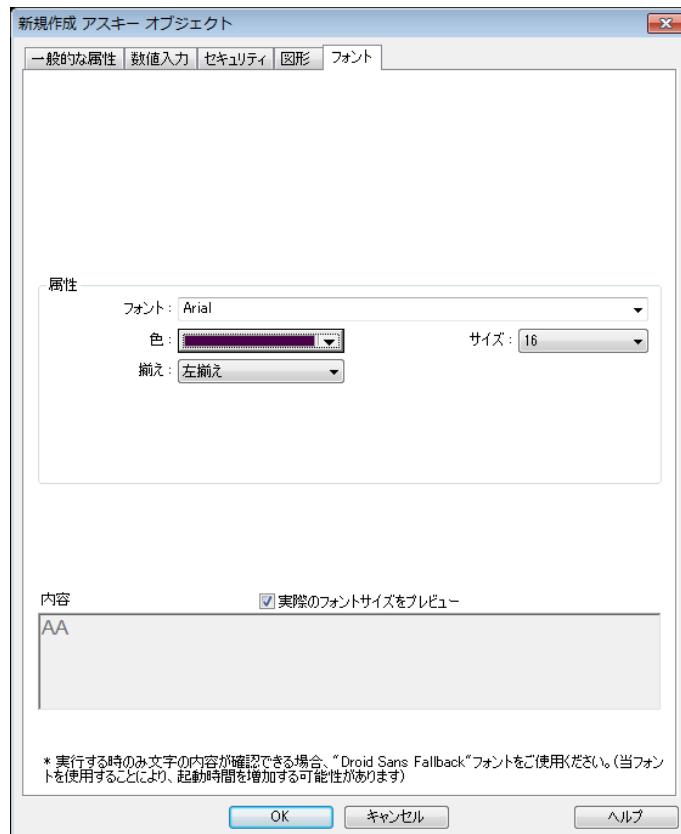


設定	記述
入力機能を有効にする	本項目にチェックマークを入れると、入力機能に関連する属性の設定をオープンすることができます。
複数行表示	複数行の文字を表示します。ストリング内で改行キャラクタの ASCII 数値 : LF (0xA)を使用すると、改行されます。
垂直整列	[複数行表示]を有効にする場合、複数行の文字の整列方法を選択できます。
パスワード	"*"で全ての数字を代わって表示します。
Unicode を使用する	Unicode フォーマットの資料を表示することができます。使用しないと、システムはワードを ASCII フォーマットに表示します。本機能はファンクションキーの[ASCII/Unicode]に合わせて使用することができます。
上位バイト/下位バイト相互変換	一般的な状況下、ASCII code の表示順番は[上位バイト]+[下位バイト]です。本機能にチェックマークを入れたら、表示順番が[下位バイト]+[上位バイト]になります。
	<p style="text-align: center;">   上位バイト/下位バイトの相互変換をしない 上位バイト/下位バイト相互変換 </p>
読み取りアドレス	[設定]をクリックし、ワードレジスタ装置タイプの[PLC名]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択してワードを表示します。[アドレスラベルライブラリ]タブで直接に PLC 名、レジスタ名及びアドレスを設定することもできます。
	
ワード数	表示する最大の文字数を設定します。単位は word です。 cMT シリーズは最大 160 words をサポートし、eMT/iE/XE/mTV/iP シリーズは最大 64 words をサポートします。

 Note

- Unicode を使用する場合、一個の Unicode 文字は一個のワードに当たります(word) ; ASCII を使用する場合、一個の ASCII 文字は一個のバイト(byte)に当たり、だから一個のワード(word)には二個の ASCII 文字を許可します。(1 個のワード(word)は 2 個のバイト(byte)に相当する)

フォント設定



設定	記述
属性	文字表示に使用するフォント、サイズ及び色、揃え方を決めます。
揃え	左揃え：数値を左揃えにします。 中央揃え：数値を中央揃えにします。 右揃え：数値を右揃えにします。

13.11. 間接ウインドウ

13.11.1. 概要

[間接ウインドウ]はワードレジスタで特定番号のウインドウのオープン及び閉じるのをコントロールするオブジェクトです。ポップアップウインドウの範囲表示は二種類があり、一種類はまずウインドウで一箇所の表示エリアを表示し、この表示エリアでポップアップウインドウを表示します。表示されたポップアップウインドウの幅と高さはこの表示エリアに超えません。もう一種類は[自動的にウインドウ寸法を調整する]機能を使用する方法で、本機能を使用したら、事先にポップアップウインドウの表示エリアを設定する必要がなく、システムは自動的に対応するポップアップウインドウの寸法に基づいてその表示エリアを調整します。ポップアップウインドウを閉じたい場合、コントロールしたワードレジスタの内容を0に設定すればいいです。[間接ウインドウ]と[直接ウインドウ]の違いは、間接ウインドウはワードレジスタにコントロールされ、直接ウインドウはビットレジスタにコントロールされます。

13.11.2. 設定



ツールバー上の[オブジェクト]»[ウインドウを組み込む]»[間接ウインドウ]ボタンを押せば、[間接ウインドウ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[間接ウインドウ]オブジェクトが新規作成されます。

一般的な属性

cMT シリーズ



eMT、iE、XE、mTV シリーズ

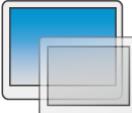
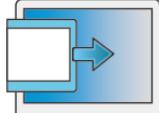
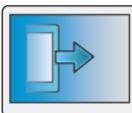
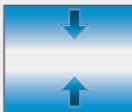


設定	記述
タイプ	<p>cMT シリーズにのみサポートされます。本機能の設定は、ベースウインドウに下敷きウインドウを使用した場合、下敷きウインドウ内の間接ウインドウの表示に影響を与えます。本節の例 2 を参考してください。</p> <p>親ウインドウに組み込まれる</p> <p>間接ウインドウは親ウインドウの上に表示されます。間接ウインドウが下敷きウインドウの上に置かれた場合、ベースウインドウでこの間接ウインドウを呼び出すと、オブジェクトは下敷きウインドウの上に表示されます。</p> <p>ポップアップウインドウ</p> <p>間接ウインドウはベースウインドウの上に表示されます。間接ウインドウが下敷きウインドウの上に置かれた場合、ベースウインドウでこの間接ウインドウを呼び出すと、オブジェクトはベースウインドウの上に表示されます。</p>
読み取りアドレス	[設定]をクリックした後、ワードレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、ウインドウのポップアップをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。
属性	<p>スタイル</p> <p>ポップアップウインドウのスタイルを選択します。二種類のスタイルをサポートします：</p> <ul style="list-style-type: none"> ● ウインドウタイトルバーを隠す <p>ポップアップの子ウインドウにはウインドウのタイトルバーを含めていなく、ウインドウをドラッグ＆ドロップや移動することができません。</p>  <ul style="list-style-type: none"> ● ウインドウタイトルバーを表示する <p>ポップアップの子ウインドウにはウインドウタイトルバーを含めていて、ウインドウの位置はタイトルバーを通じて任意にドラッグ＆ドロップできます。</p> 

アニメーション(cMT シリーズ)

効果

ウインドウの[開始]及び[終了]動作を表すアニメーション効果を選択できます。

効果	スタイル
フーイン	
飛んでくる	
浮かび上がる	
ワイプ	
スプリット	
サイクル	
クロック	
ズーム	
ターン	

表示時間

ウインドウの開始/終了アニメーションが完了するまでの時間を設定できます。

表示方向

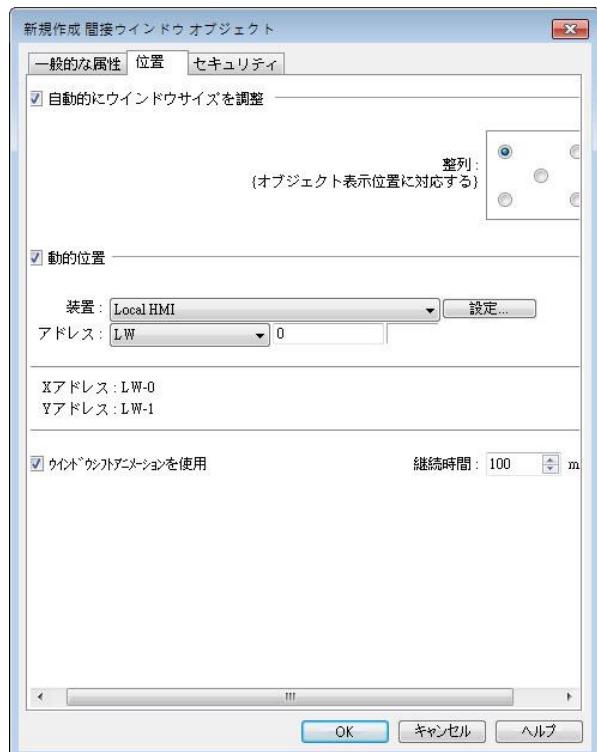
ウインドウの開始/終了アニメーションの移動方向を設定できます。

**ウインドウ番
号オフセット
を使用**

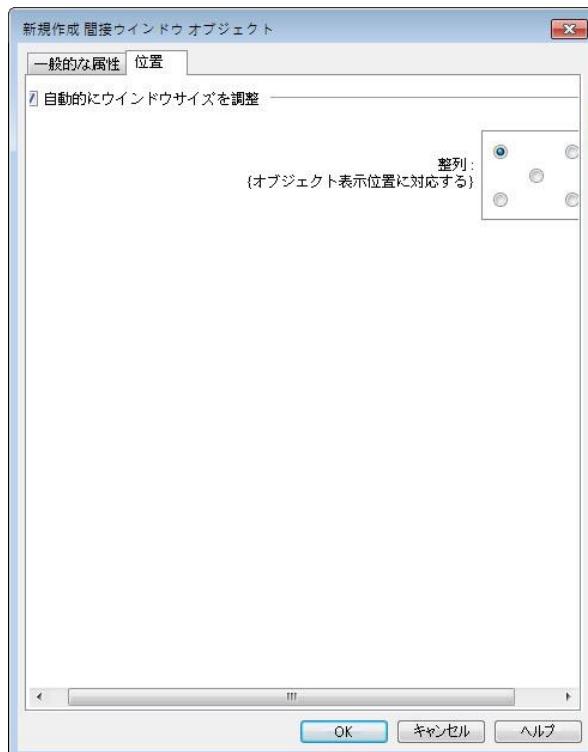
ポップアップのウインドウ番号はレジスタ内の数値にオフセットを足したものです。例えば：対象ワードレジス内の数値は 20、オフセットは 5 の場合、ポップアップウインドウの番号は 25 になります。

位置設定

cMT シリーズ



eMT、iE、XE、mTV シリーズ

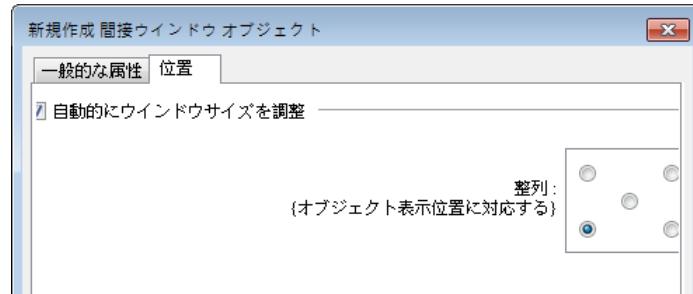


設定	記述
自動的にウイ ンドウサイズ を調整する	<p>システムはポップアップされたウインドウの寸法に応じて表示範囲と相対位置を自動的に調整します。</p> <p>(オブジェクト表示位置に対応する)</p> <p>ポップアップするウインドウの位置基準点を設定します(間接ウインドウオブジェクトに対応する基準点)。例えば：右下隅を基準点にする場合、任一ウインドウがポップアップされた際に、当ウインドウの右下隅をこの基準点に揃えます。左上隅を基準点にする場合、任一ウインドウがポップアップされた際に、当ウインドウの左上隅をこの基準点に揃え、このように類推します。下記例 1 をご参照ください。</p>
動的位置	<p>本機能を有効にすると、HMI で指定したアドレスでウインドウの位置を調整できます。</p> <p>ウインドウシフトアニメーションを有効にする</p> <p>ウインドウ位置が移動される際に、アニメーションを使用するかを設定します。</p>

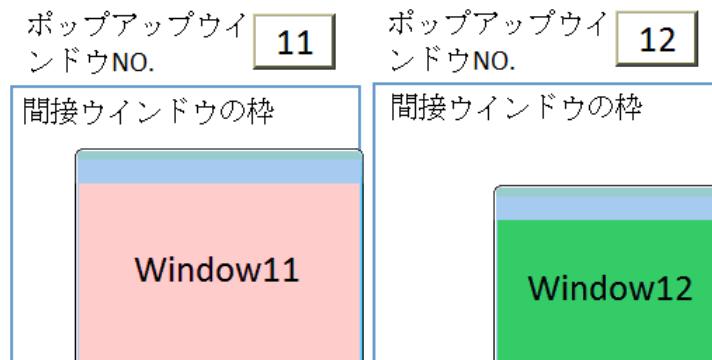
例 1

仮に 11 番と 12 番と 2 個のポップアップウインドウがあり、ワードレジスタアドレス LW-0 でコントロールし、[自動的にウインドウサイズを調整する]にチェックマークを入れ、また右下隅をポップアップ基準点にする場合：

1. [間接ウインドウ]オブジェクトを作成し、読み取りアドレスを LW-0 にして、[自動的にウインドウサイズを調整する]にチェックマークを入れます。
2. [間接ウインドウ]オブジェクトでポップアップウインドウの表示エリアを調整します。



3. 数値 11 を LW-0 に書き込んだら、ポップアップウインドウ番号は 11 になります。
4. 数値 12 を LW-0 に書き込んだら、ポップアップウインドウ番号は 12 になります。
5. 数値 0 を LW-0 に書き込んだら、ポップアップウインドウを閉じます。

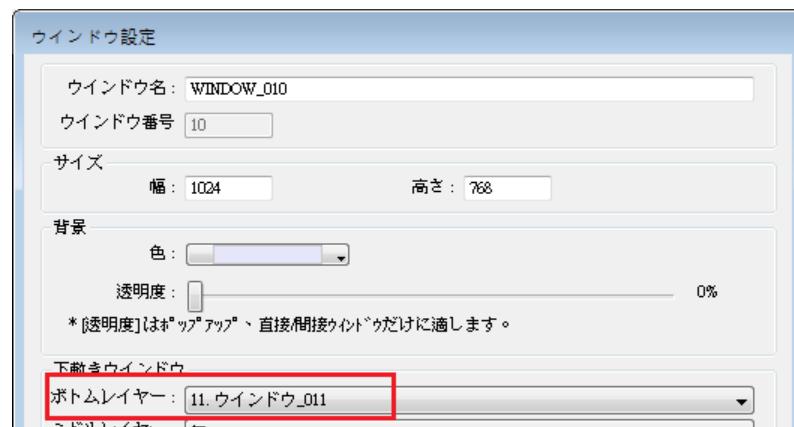


ポップアップウインドウを閉じるには、コントロールワードレジスタに数値 0 を書き込む以外、ポップアップウインドウで一個の[ファンクションキー]オブジェクトを作成し、[ウインドウを閉じる]モードを選択し、このオブジェクトを押したら、ポップアップウインドウを閉じることができます。

例 2

本例では、間接ウンドウオブジェクトが下敷きウンドウの上に置かれた場合、ベースウンドウでこの間接ウンドウオブジェクトを呼び出すと、親ウンドウに組み込まれるとポップアップウンドウを選択した場合のそれぞれの表示方式について説明いたします。

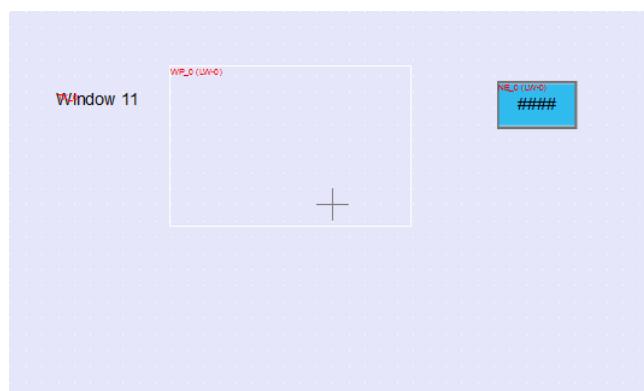
1. 2 個のベースウンドウ：ウンドウ 10 とウンドウ 11 を作成し、ウンドウ 10 で下敷きウンドウを有効にし、それをウンドウ 11 に指定します。



2. ウィンドウ 10 で数個のオブジェクトを作成します。

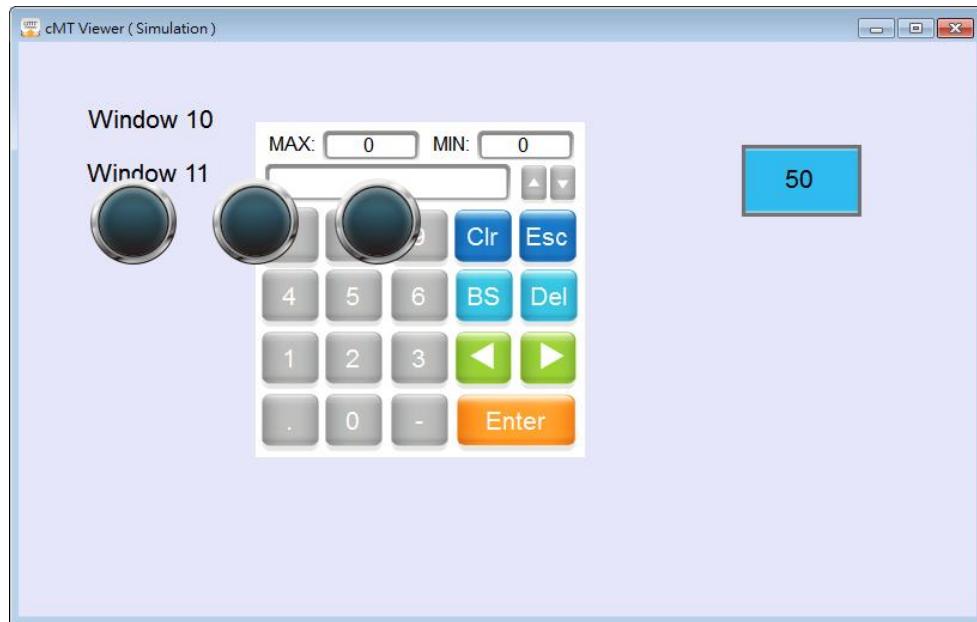


3. ウィンドウ 11 で一個の[間接ウィンドウ]オブジェクトと一個の[数値入力]オブジェクトを作成し、読み取りアドレスを全部 LW-0 に設定します。



4. [数値入力]オブジェクトでポップアップウィンドウの番号を入力すると、タイプを[親ウィンドウに組み込まれる]と[ポップアップウィンドウ]に選択した場合のそれぞれの表示方式(下図のとおり)が見られます。

親ウィンドウに組み込まれる



ポップアップアップウィンドウ



Note

- eMT, iE, XE, mTV シリーズは最大同時に 24 個のシステムメッセージウィンドウ、直接ウインドウ、間接ウインドウを含むポップアップアップウィンドウを表示できます。cMT シリーズはポップアップアップウィンドウの表示上限がありません。
- システムは一個のベースウインドウの上で、2 個の直接(または間接)ウインドウを使用し同一のウインドウをポップアップアップすることを禁じます。
- ポップアップされたウインドウが[独占]属性を有する場合、ウインドウがポップアップされた後、背景ウインドウの操作は完全に一時停止され、独占ウインドウが閉じられてから他のウインドウを操作することが可能になります。

▶このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

13.12. 直接ウインドウ

13.12.1. 概要

[直接ウインドウ]はビットレジスタでポップアップウインドウのオープンと閉じることをコントロールするオブジェクトです。

まず、ウインドウで一箇所の表示エリアを定義し、指定したビットレジスタの状態が変わった場合、この表示エリアで本ウインドウの内容を表示します。表示されるウインドウの幅と高さはこの表示エリアを超ません。本ポップアップウインドウをコントロールするビットレジスタの状態を復帰した後、本ポップアップウインドウを閉じることができます。

[直接ウインドウ]と[間接ウインドウ]の違いは、直接ウインドウはビットレジスタにコントロールされ、間接ウインドウはワードレジスタにコントロールされます。

13.12.2. 設定

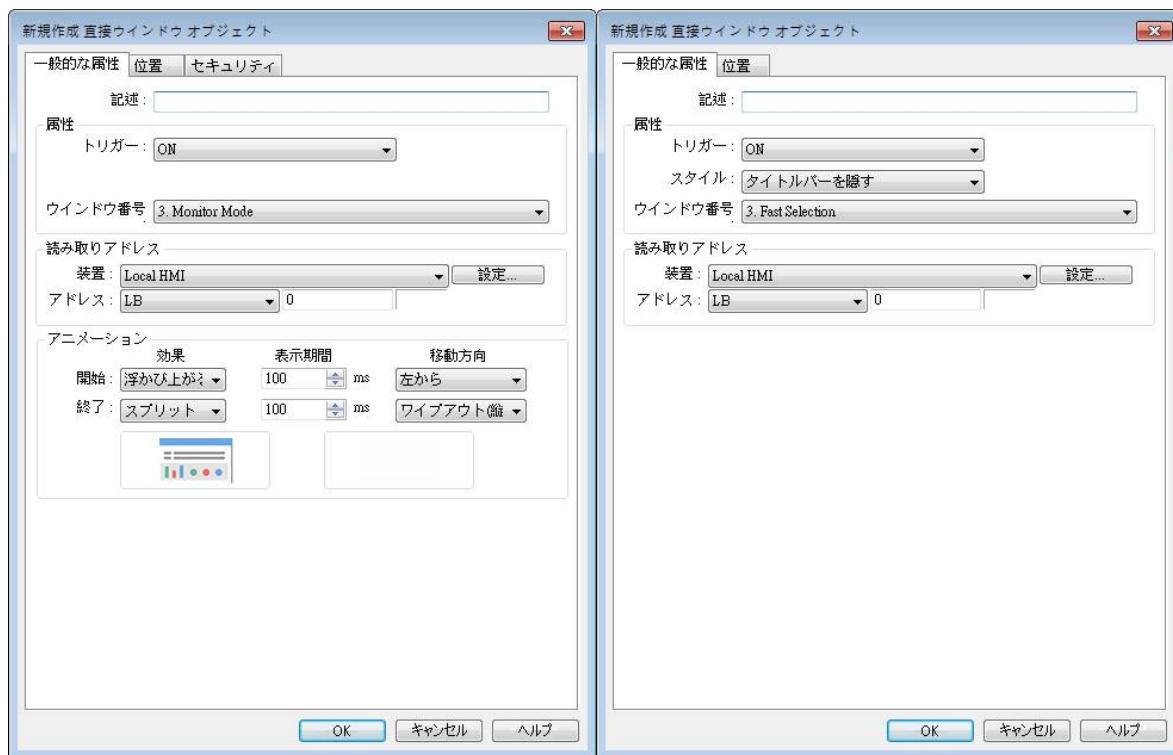


ツールバーの[オブジェクト]»[ウインドウを組み込む]»[直接ウインドウ]ボタンを押せば、[直接ウインドウ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[直接ウインドウ]オブジェクトが新規作成されます。

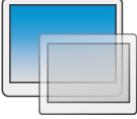
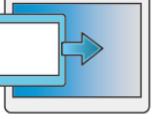
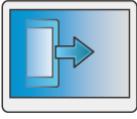
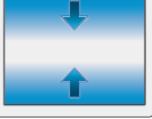
一般的な属性の設定

cMT シリーズ

eMT、iE、XE、mTV シリーズ



設定	記述
タイプ	<p>cMT シリーズにのみサポートされます。本機能の設定は、ベースウインドウに下敷きウインドウを使用した場合、下敷きウインドウ内の直接ウインドウの表示に影響を与えます。ch 13.11 間接ウインドウの例 2 を参考してください。</p> <p>親ウインドウに組み込まれる</p> <p>直接ウインドウは親ウインドウの上に表示されます。直接ウインドウが下敷きウインドウの上に置かれた場合、ベースウインドウでこの直接ウインドウを呼び出すと、オブジェクトは下敷きウインドウの上に表示されます。</p> <p>ポップアップウインドウ</p> <p>直接ウインドウはベースウインドウの上に表示されます。直接ウインドウが下敷きウインドウの上に置かれた場合、ベースウインドウでこの直接ウインドウを呼び出すと、オブジェクトはベースウインドウの上に表示されます。</p>
読み取りアドレス	[設定]をクリックした後、ビットレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、ウインドウのポップアップをコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。
属性	<p>スタイル</p> <p>ポップアップウインドウのスタイルを選択します。二種類のスタイルをサポートします：</p> <ul style="list-style-type: none"> ● タイトルバーを隠す <p>ポップアップウインドウにタイトルバーがなく、ドラッグされません。</p>  <ul style="list-style-type: none"> ● タイトルバーを表示する <p>ポップアップウインドウにはタイトルバーがあり、ドラッグするとウインドウを移動できます。</p> <p>ウインドウ番号</p> <p>ポップアップするウインドウの番号を設定します。</p>
アニメーション(cMT シリーズ)	<p>効果</p> <p>ウインドウの[開始]及び[終了]動作を表すアニメーション効果を選択できます。</p>

効果	スタイル
フェードイン	
飛んでくる	
浮かび上がる	
ワイプ	
スプリット	
サイクル	
クロック	
ズーム	
ターン	
プッシュ	

表示時間

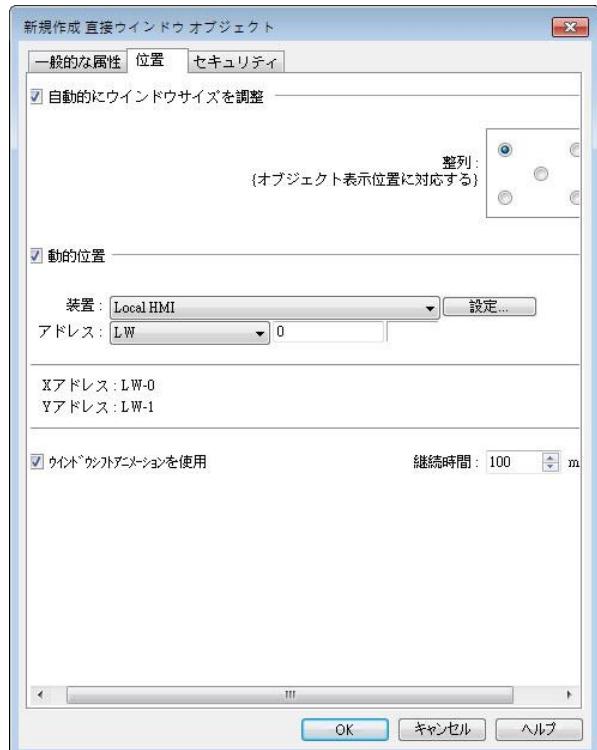
ウインドウの開始/終了アニメーションが完了するまでの時間を設定できます。

表示方向

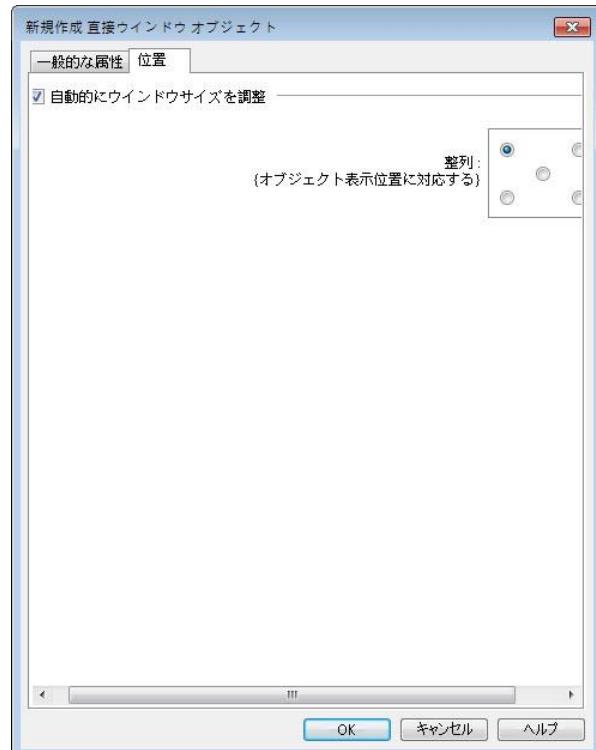
ウインドウの開始/終了アニメーションの移動方向を設定できます。

位置設定

cMT シリーズ



eMT、iE、XE、mTV シリーズ

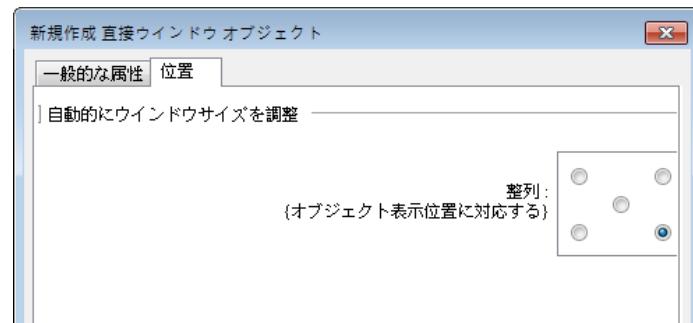


設定	記述
自動的にウ ィンドウサイズ を調整する	<p>システムはポップアップされたウインドウの寸法に応じて表示範囲と相対位置を自動的に調整します。</p> <p>(オブジェクト表示位置に対応する)</p> <p>ポップアップするウインドウの位置基準点を設定します(直接ウインドウオブジェクトに対応する基準点)。例えば：右下隅を基準点にする場合、任一ウインドウがポップアップされた際に、当ウインドウの右下隅をこの基準点に揃えます。左上隅を基準点にする場合、任一ウインドウがポップアップされた際に、当ウインドウの左上隅をこの基準点に揃え、このように類推します。下記例 1 をご参照ください。</p>
動的位置	<p>本機能を有効にすると、HMI で指定したアドレスでウインドウの位置を調整できます。</p> <p>ウインドウシフトアニメーションを有効にする</p> <p>ウインドウ位置が移動される際に、アニメーションを使用するかを設定します。</p>

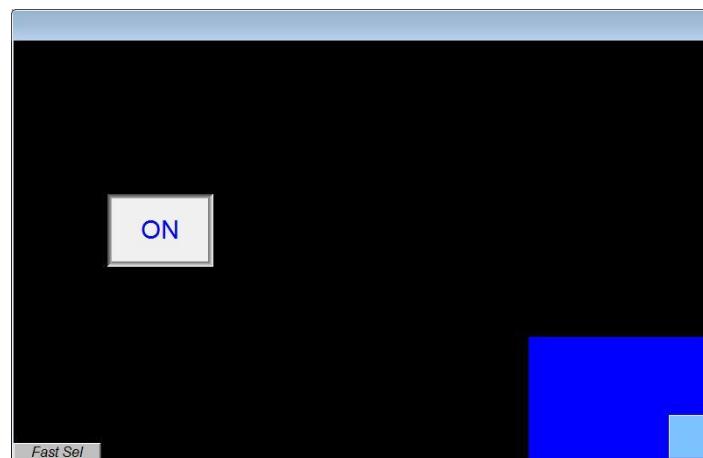
例 1

番号 11 というポップアップウインドウを作成し、[トグルスイッチ]でコントロールします。アドレスを LB-0 に設定します。

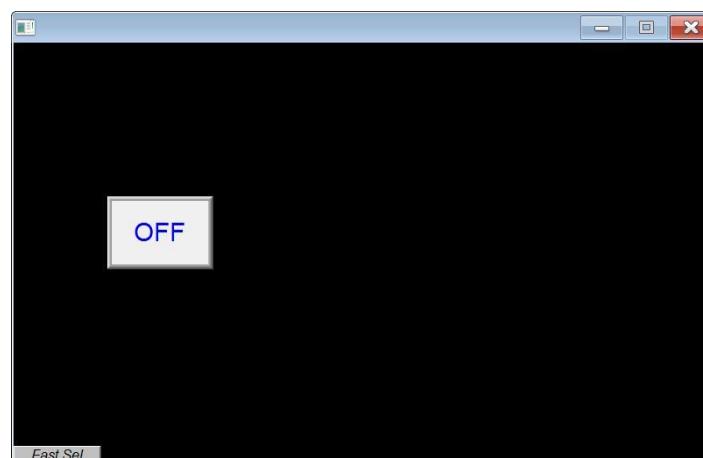
1. [直接ウインドウ]オブジェクトを作成し、読み取りアドレスを LB-0 に設定します。[自動的にウインドウサイズを調整する]にチェックマークを入れます。
2. {オブジェクト表示位置に対応する}で基準点を設定します。本例では右下隅に設定します。



3. LB-0 の状態が ON になると、ウインドウ 11 がポップアップされます。



4. LB-0 の状態が OFF になると、ウインドウ 11 が閉じられます。



 Note

- eMT、iE、XE、mTV シリーズは最大同時に 24 個のシステムメッセージウインドウ、直接ウインドウ、間接ウインドウを含むポップアップウインドウを表示できます。cMT シリーズはポップアップウインドウの表示上限がありません。
 - システムは一個のベースウインドウの上で、2 個の直接(または間接)ウインドウを使用し同一のウインドウをポップアップすることを禁じます。
 - ポップアップされたウインドウが[独占]属性を有する場合、ウインドウがポップアップされた後、背景ウインドウの操作は完全に一時停止され、独占ウインドウが閉じられてから他のウインドウを操作することが可能になります。
-  このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

13.13.移動/回転図形

13.13.1. 概要

[移動/回転図形]はオブジェクトの状態、移動距離と回転角度を定義するオブジェクトです。オブジェクトは読み取りアドレス及び連続したレジスタ内のデータにより、オブジェクトの状態、移動距離と回転角度を変更します。

13.13.2. 設定



ツールバー上の[オブジェクト]»[アニメーション]»[移動/回転図形]ボタンを押せば、[移動/回転図形]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[移動/回転図形]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述
読み取りアドレス	[設定]をクリックした後、ワードレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択し、移動/回転図形をコントロールします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。
属性	オブジェクトの移動方式及び移動範囲を選択します。下の《13.13.2.1 属性モード》をご参照ください。
表示比	オブジェクト各状態の図形を表示する時、個別で拡大/縮小の比率を設定することができます。下図をご参考ください。
	<p>比率 : 1 比率 : 1.2 比率 : 1.4 比率 : 1.6</p> <p>State 0 State 1 State 2 State 3</p>
回転	オブジェクトの回転方向を選択します。時計回りまたは反時計回りを選択できます。単位は度で、設定範囲は0度から360度までで、最大角度は最小角度より大きくなければなりません。下の《13.13.2.2 回転モードの説明》をご参照ください。
制限値アドレス	[制限値アドレス]のデータでオブジェクトの表示エリアを調整します。例1をご参照ください。

例 1

[制限値アドレス]のデータでオブジェクトの表示エリアを調整します。制限値アドレスが LW-n の場合、X 軸、Y 軸と回転角度の上下限は以下の規則で自動的に設定されます：

データフォーマット	16-bit	32-bit	64-bit (cMTのみ)
位置-X 軸座標下限	LW-n	LW-n	LW-n
位置-X 軸座標上限	LW-n+1	LW-n+2	LW-n+4
位置-Y 軸座標下限	LW-n+2	LW-n+4	LW-n+8
位置-Y 軸座標上限	LW-n+3	LW-n+6	LW-n+12
回転-入力下限	LW-n+4	LW-n+8	LW-n+16
回転-入力上限	LW-n+5	LW-n+10	LW-n+20
回転-比率下限	LW-n+6	LW-n+12	LW-n+24
回転-比率上限	LW-n+7	LW-n+14	LW-n+28

13.13.2.1. 属性モード

(以下は仮に読み取りアドレスを LW-n にする場合)

- X 軸に沿って水平方向に移動する

オブジェクトが X 軸に沿って水平方向で移動することだけが許可されます。移動範囲は[X 軸座標下限]と[X 軸座標上限]で決定します。

属性

モード:	X軸に沿って水平方向に移動する		
状態数:	8	X 座標下限:	0
	X 座標上限:	600	

データフォーマット	16-bit	32-bit	64-bit (cMT のみ)
オブジェクト状態アドレス	LW-n	LW-n	LW-n
X 軸での移動距離アドレス	LW-n+1	LW-n+2	LW-n+4

13.13.2.2. 回転モードの説明

オブジェクトの回転を制御するアドレスは読み取りアドレス(LW-n)からの連続したアドレスであり、これらのアドレスは選択したモードにより変わる可能性があります。対応するアドレスは、読み取りアドレスの下側にある[使用方法...]をご参照ください。



● 回転

オブジェクトを時計回りまたは反時計回りの方向で回転させます。回転角度の範囲は[最小角度]と[最大角度]で決められます。

アニメーションは[デフォルト]か[最短]に設定することができます。デフォルトは、両角度間の線形補間でオブジェクトを回転させます。最短は、最短のアニメーションルートを生成する方向で回転させます。例えば、時計回りで回転させる場合、デフォルトに設定すれば、回転角度が 10 度から 350 度になった際に、オブジェクトは時計回りで 340 度回転させられます。最短に設定すれば、回転角度が 10 度から 350 度になった際に、オブジェクトは反時計回りで 20 度回転させられます。

回転

モード:	回転		
角度測定:	<input checked="" type="radio"/> 時計回り	<input type="radio"/> 反時計回り	
アニメーション:	<input checked="" type="radio"/> デフォルト	<input type="radio"/> 最短	
最小角度:	0	最大角度:	359

- 比率で回転する

オブジェクトを時計回りまたは反時計回りの方向で回転させます。下の数式で回転角度を決めます。

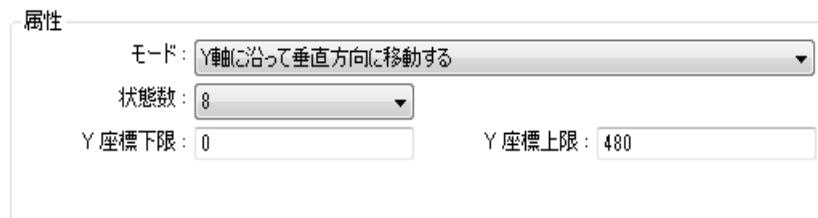
$$\text{回転角度} = (\text{読み取りアドレス数値} - [\text{入力下限}]) \times \frac{[\text{比率上限}] - [\text{比率下限}]}{[\text{入力上限}] - [\text{入力下限}]}$$

- 逆比率で回転する

本機能は[比率で回転する]と同じですが、回転方向が逆です。

- Y 軸に沿って垂直方向に移動する

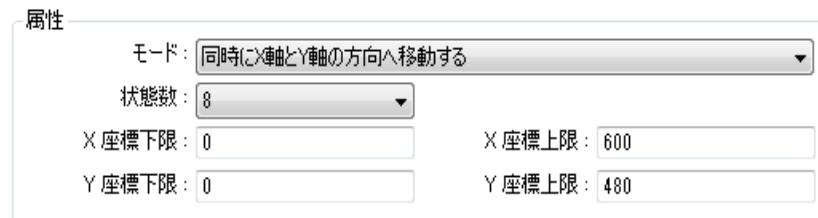
オブジェクトが Y 軸に沿って垂直方向で移動することだけが許可されます。移動範囲は[Y 軸座標下限]と[Y 軸座標上限]で決定します。



データフォーマット	16-bit	32-bit	64-bit
(cMT のみ)			
オブジェクト状態アドレス	LW-n	LW-n	LW-n
Y 軸での移動距離アドレス	LW-n+1	LW-n+2	LW-n+4

- 同時に X 軸と Y 軸の方向へ移動する

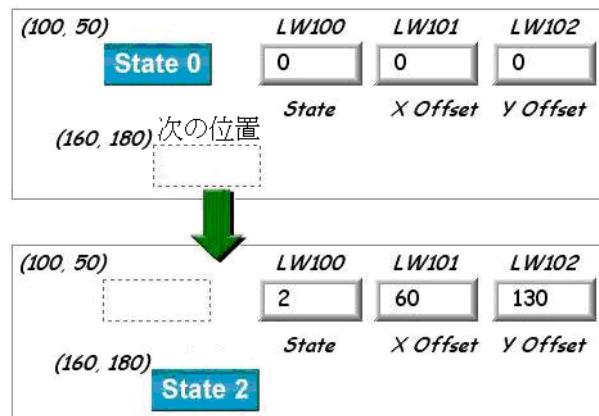
オブジェクトが X 軸と Y 軸に沿って移動することが許可されます。移動範囲は[X 軸座標下限]、[X 軸座標上限]や[Y 軸座標下限]、[Y 軸座標上限]で決定します。



データフォーマット	16-bit	32-bit	64-bit
(cMT のみ)			
オブジェクト状態アドレス	LW-n	LW-n	LW-n
X 軸での移動距離アドレス	LW-n+1	LW-n+2	LW-n+4
Y 軸での移動距離アドレス	LW-n+2	LW-n+4	LW-n+8

例として、もしオブジェクトの読み取りレジスタは LW-100 で、しかも変数が[16-bit Unsigned]の場合、LW-100 でオブジェクトの状態をコントロールし、LW-101 で X 軸方向の移動距離をコントロールし、LW-102 で Y 軸方向の移動距離をコントロールします。

下図を例として、オブジェクトの読み取りアドレスは LW-100 で、開始位置は(100、 50)の場合、仮にプロジェクトを(160、 180)に移動し、それに状態 2 の図形を表示したかったら、LW-100 は 2 に設定する必要があり、LW-101=160-100=60、LW-102=180-50=130 になります。



- X 軸に沿って、比率を基づいて水平方向に移動する

オブジェクトが X 軸に沿って、比率を基づいて水平方向に移動することだけが許可されます。

$$\text{公式: 移動距離} = (\text{読み取りアドレス} - \text{入力下限}) \times \frac{\text{比率上限} - \text{比率下限}}{\text{入力上限} - \text{入力下限}}$$

データフォーマット	16-bit	32-bit	64-bit (cMT のみ)
オブジェクト状態アドレス	LW-n	LW-n	LW-n
X 軸での移動距離アドレス	LW-n+1	LW-n+2	LW-n+4

- Y 軸に沿って、比率を基づいて垂直方向に移動する

オブジェクトが Y 軸に沿って、比率を基づいて垂直方向に移動することだけが許可されます。

公式は[X 軸に沿って、比率を基づいて水平方向に移動する]と同じです。

データフォーマット	16-bit	32-bit	64-bit (cMT のみ)
オブジェクト状態アドレス	LW-n	LW-n	LW-n
Y 軸での移動距離アドレス	LW-n+1	LW-n+2	LW-n+4

- X 軸に沿って、逆比を基づいて水平方向に移動する

本機能は[X 軸に沿って、比率を基づいて水平方向に移動する]と同じで、ただ移動する方向は逆です。

- Y 軸に沿って、逆比を基づいて垂直方向に移動する

本機能は[Y 軸に沿って、比率を基づいて垂直方向に移動する]と同じで、ただ移動する方向は逆です。

13.14. アニメーション

13.14.1. 概要

ユーザーは予め[アニメーション]オブジェクトの移動軌跡を定義し、それにレジスタ内のデータを変更することにより、オブジェクトの状態と移動軌跡上の位置をコントロールすることができます。システムは二個連続のレジスタ内のデータでアニメーションオブジェクトをコントロールします。一番目のレジスタはオブジェクトの状態をコントロールし、二番目はオブジェクトの位置をコントロールします。

13.14.2. 設定

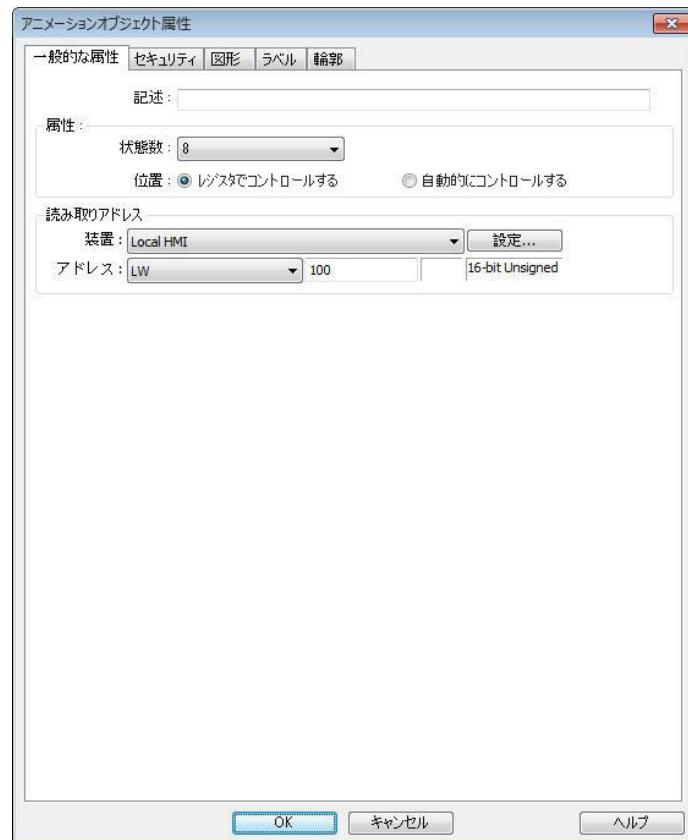


ツールバー上の[オブジェクト]»[アニメーション]»[アニメーション]ボタンを押せば、[アニメーション]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[アニメーション]オブジェクトが新規作成されます。



オブジェクトの属性を変更するには、オブジェクトの所在位置にマウスの左ボタンをダブルクリックし、現れる[アニメーション]オブジェクト属性ダイアログウインドウでオブジェクトの各属性を変更することができます。

一般的な属性の設定



設定	記述						
属性	<p>状態数 オブジェクトの状態数を設定します。</p> <p>レジスタでコントロールする レジスタのデータでオブジェクトの状態と位置を決めます。この時は正確にオブジェクト状態と位置の読み取りアドレスを設定する必要があります。下記例 1 をご参照ください。</p> <p>時間間隔でコントロールする オブジェクトが自動的に状態と表示位置を変更する機能で、[自動的にコントロールする]項目で状態と表示位置の変更方式を設定します。</p> <p>自動的にコントロールする</p> <table border="1"> <tr> <td>速度 : 10</td> <td>* 0.1 秒</td> </tr> <tr> <td>状態変換 : 時間に基づく</td> <td><input checked="" type="checkbox"/> 戻る</td> </tr> <tr> <td>変換周期 : 5</td> <td>* 0.1 秒</td> </tr> </table> <p>速度 : 位置を変更する速度です。単位は 0.1 秒です。例えば 10 に設定すると、オブジェクトは 1 秒ごとに位置を変更します。</p> <p>状態変更 : 状態を変更する方式です。[位置に基づく]と[時間をに基づく]を選択することができます。[位置に基づく]を選択すると、位置が変わったとともに、状態も変わります。[時間に基づく]を選択する</p>	速度 : 10	* 0.1 秒	状態変換 : 時間に基づく	<input checked="" type="checkbox"/> 戻る	変換周期 : 5	* 0.1 秒
速度 : 10	* 0.1 秒						
状態変換 : 時間に基づく	<input checked="" type="checkbox"/> 戻る						
変換周期 : 5	* 0.1 秒						

と、位置は固定した頻度で自動的に変わります。変更する頻度は[変換周期]で設定します。

逆さに表示：仮にオブジェクトは 4箇所の位置があり、それぞれ position 0、position 1、position 2、position 3 と分けています。本項を選択しない場合、最後の位置(position 3)に移ったたびに、オブジェクトはまた開始位置 position 0 に戻り、本来のように位置の変更を重複します。位置の移動順次は以下のとおりです：

position 0→position 1→position 2→position 3→position 0→position 1→position 2...

本設定を選択すると、最後の位置(position 3)に移ったたびに、オブジェクトはまず逆方向の移動方式で、開始位置 position 0 に戻ってから、本来のように位置変更を重複します。位置の移動順次は以下のとおりです：

position 0→position 1→position 2→position 3→position 2→position 1→position 0...

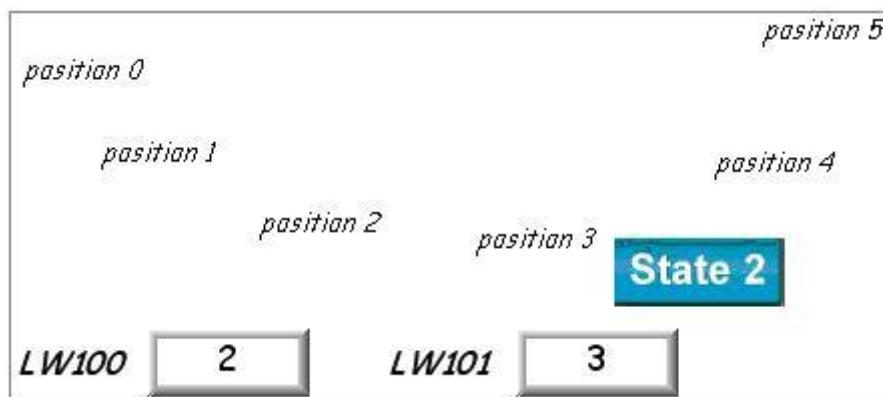
例 1

オブジェクトの状態と位置はレジスタ内のデータで決められる場合、下表のように正確にオブジェクト状態と位置の読み取りアドレスを設定する必要があります。

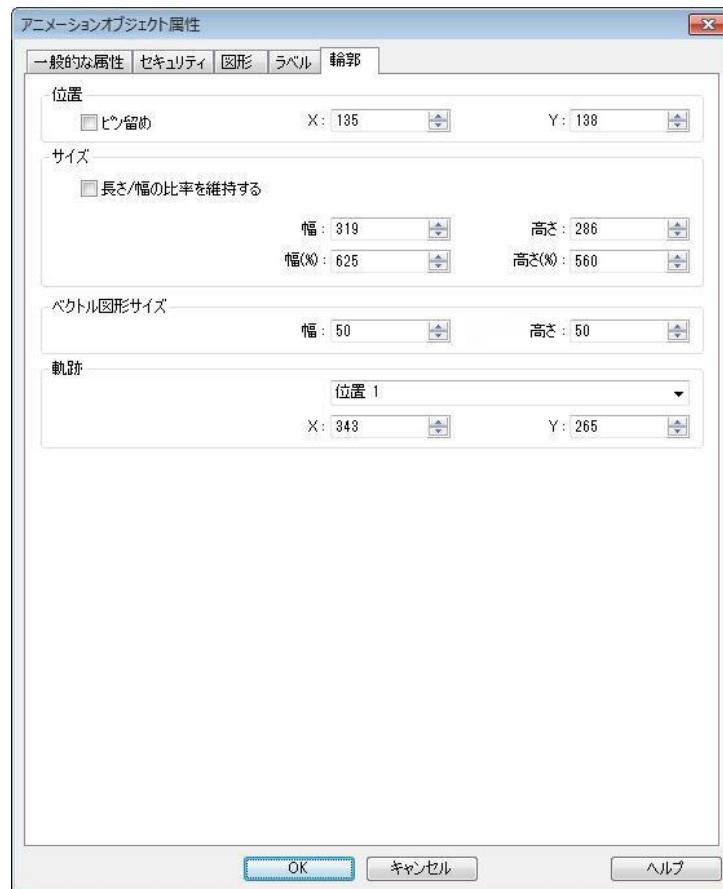
データフォーマット	16-bit	32-bit	64-bit (cMTのみ)
オブジェクト状態アドレス	LW-n	LW-n	LW-n
オブジェクト位置アドレス	LW-n+1	LW-n+2	LW-n+4

例として、もしオブジェクトの読み取りレジスタは LW-100 で、フォーマットは[16-bit Unsigned]の場合、LW-100 でオブジェクトの状態をコントロールし、LW-101 はオブジェクトの表示位置をコントロールします。

下図を例として、LW-100=2、LW-101=3、だからオブジェクトは状態 2 を表示し、位置 3 に現れます。



輪郭



設定	記述
図形サイズ	オブジェクトが表示する図形の大きさを設定します。
軌跡	移動軌跡上の各点の位置を設定します。

Note

- 一個のアニメーションオブジェクトには複数の異なる図面が設定されるので、[原寸サイズを使用する]機能を使用することができません。

13.15.棒グラフ

13.15.1. 概要

[棒グラフ]は百分率と棒グラフの形で、レジスタ内のデータを表示するオブジェクトです。

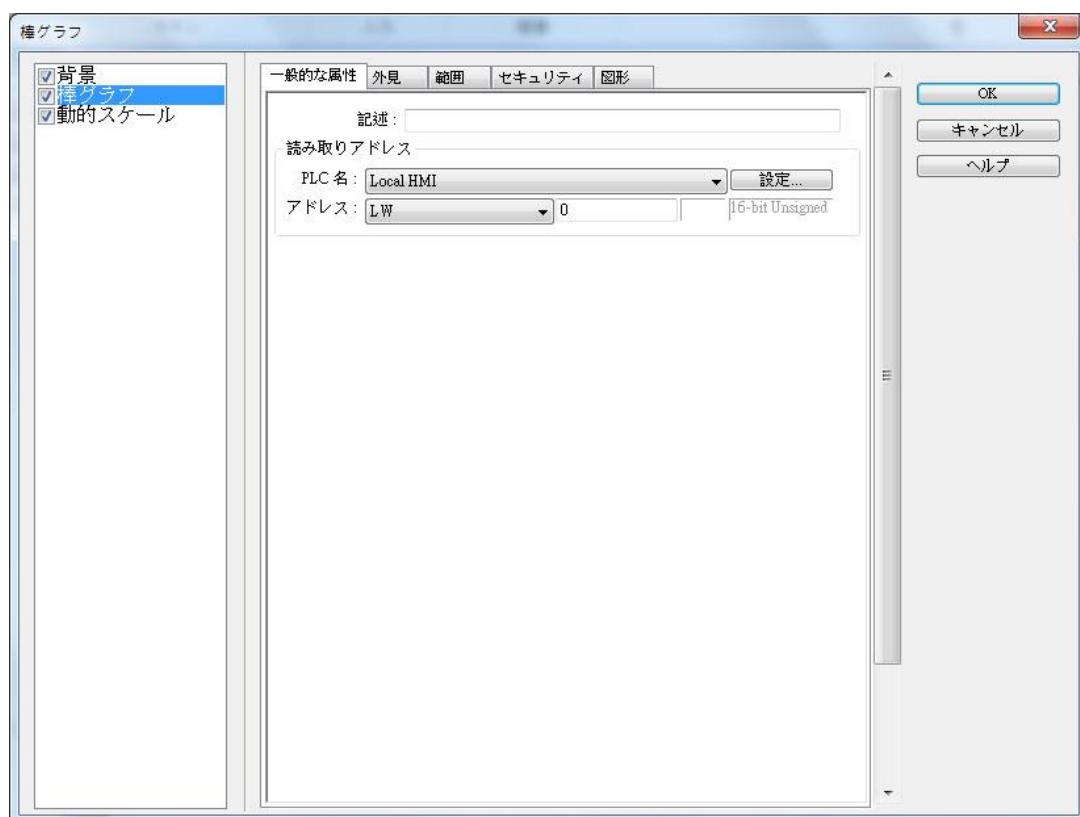
13.15.2. 設定



ツールバー上の[オブジェクト]»[オブジェクト]»[チャート]»[棒グラフ]ボタンを押せば、[棒グラフ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[棒グラフ]オブジェクトが新規作成されます。

一般的な属性の設定

cMT シリーズ



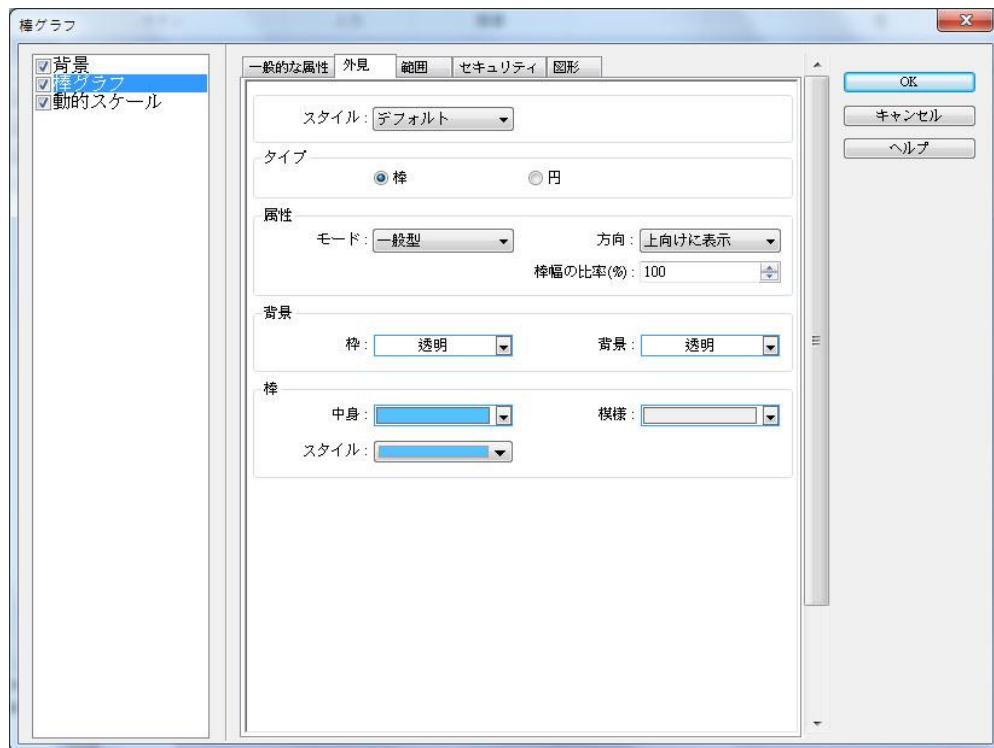
eMT、iE、XE、mTV シリーズ



設定	記述
読み取りアドレス	[設定]をクリックした後、ワードレジスタ装置タイプの[装置]、[アドレス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択することで、棒グラフの表示ソースとします。

外見設定

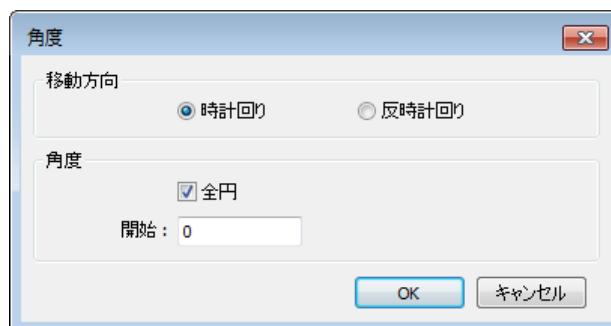
cMT シリーズ



eMT、iE、XE、mTV シリーズ



設定	記述
タイプ	[棒]または[円]タイプの棒グラフが選択可能です。
属性	<p>モード</p> <p>[一般型]と[オフセット型]が選択可能です。[オフセット型]を選択する場合、原点の位置を設定する必要があります。</p> <p>表示方向/角度</p> <p>[棒]タイプの棒グラフは、表示方向を選択することができます。[上向けに表示]、[下向けに表示]、[右向けに表示]、[左向けに表示]があります。</p> <p>[円]タイプの棒グラフは、グラフの[移動方向]と[角度]を選択することができます。</p>

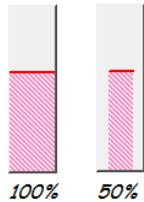


[移動方向]では[時計回り]または[反時計回り]が選択可能です。
 [角度]で[全円]を選択したら、[開始角度]を設定する必要があります。
 [全円]を選択していない場合、[開始角度]と[終了角度]を設定する必

要があります。

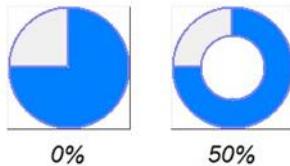
棒幅の比率(%)

[棒]タイプの棒グラフには、棒の幅とオブジェクトの幅との間の百分率を表示します。設定することができます。下図では異なる比率を表示しています：100%と50%。



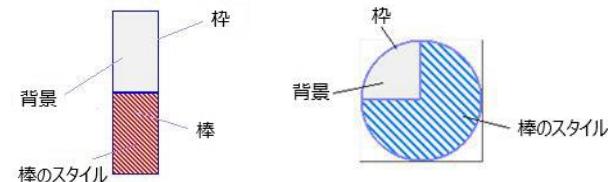
円心半径(%)

[円]タイプの棒グラフには円心の比率を設定することができます。下図では異なる円心の比率を示しています：0%及び50%です。



色/スタイル

棒の枠、背景色と充填部分の模様と色を設定します。下図を参考してください。



範囲の設定



設定	記述
最小値/最大値	棒グラフに充填する百分率は公式で換算して得られます。下記例 1 をご参考ください。
目標値	レジスタ内のデータが条件に満たせれば、充填部分の色は本項が定義した色に変更可能です。下記例 2 をご参照ください。
アラーム表示	数値が[上限値]より大きい場合、充填部分の色は[上限色]が定義した色に変更可能です。数値が[下限値]より小さい場合、充填部分の色を[下限色]が定義した色に変更可能です。
範囲上下限	[上下限値をレジスタから取得]を選択した場合、[アラーム範囲]で使用する[下限値]、[上限値]と[目標値]項目内の[目標値]は全部指定したレジスタから取得するものになります。指定したレジスタは入力の欄に表示されます。下記例 3 をご参照ください。

例 1

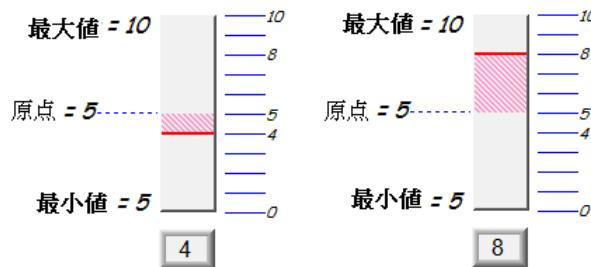
棒グラフに充填する百分率は下記公式で換算して得られます：

$$\text{表示する百分率} = \frac{\text{レジスタ内の数値}-\text{最小値}}{\text{最大値}-\text{最小値}} \times 100\%$$

もし(レジスタ数値-[原点位置])が 0 より大きい場合、棒グラフは[原点位置]から上へ充填します。
もし(レジスタ数値-[原点位置])が 0 より小さい場合、棒グラフは[原点位置]から下へ充填します。

下図は[原点位置]を 5、[最大値]を 10、[最小値]を 0 に設定し、それに異なる数値を使用する場合、棒グラフの充填状況を示しています。

読み取り数値は 4 の場合 読み取り数値は 8 の場合

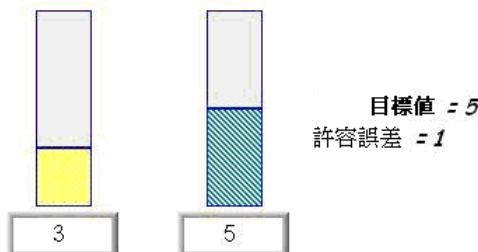


例 2

レジスタ内のデータが下記条件に満たせば、充填部分の色は本項が定義した色に変更することができます。

$$[\text{目標値}]-[\text{許容誤差}] \leq \text{レジスタ内の数値} \leq [\text{目標値}]+[\text{許容誤差}]$$

下図をご参考ください。この場合、[目標値]=5、[誤差値]=1 で、レジスタの値は $\geq 5-1=4$ 、それに $\leq 5+1=6$ になったら、充填部分の色は“目標値色”に変更します。



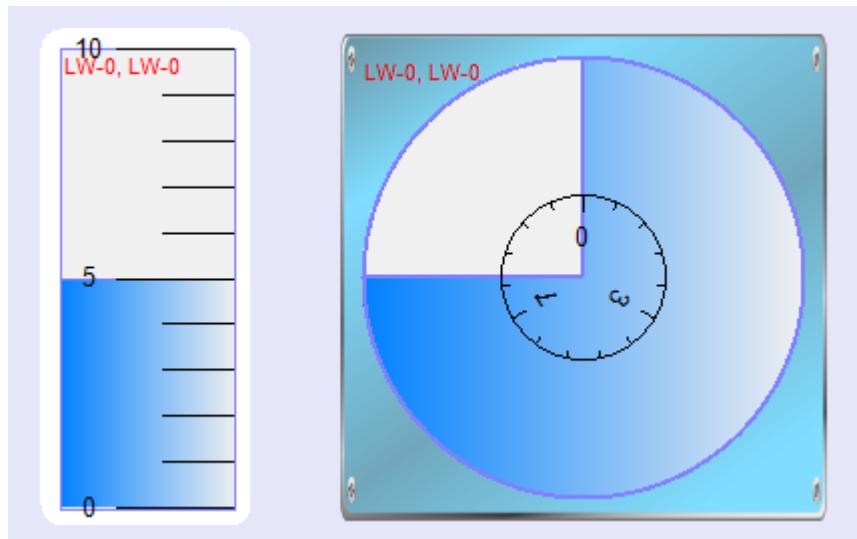
例 3

アラーム範囲の上下限は指定したレジスタで設定することができます。書き込みアドレスは $LW-n$ の場合、上/下限及び関連数値は以下の規則で自動的に設定されます：(仮に[上下限値をレジスタから取得]にチェックマークを入れた場合)

データフォーマット	16-bit	32-bit	64-bit (cMT のみ)
アラーム下限値	$LW-n$	$LW-n$	$LW-n$
アラーム上限値	$LW-n+1$	$LW-n+2$	$LW-n+4$
目標値	$LW-n+2$	$LW-n+4$	$LW-n+8$
最小値	$LW-n+3$	$LW-n+6$	$LW-n+12$
最大値	$LW-n+4$	$LW-n+8$	$LW-n+16$
原点位置	$LW-n+5$	$LW-n+10$	$LW-n+20$

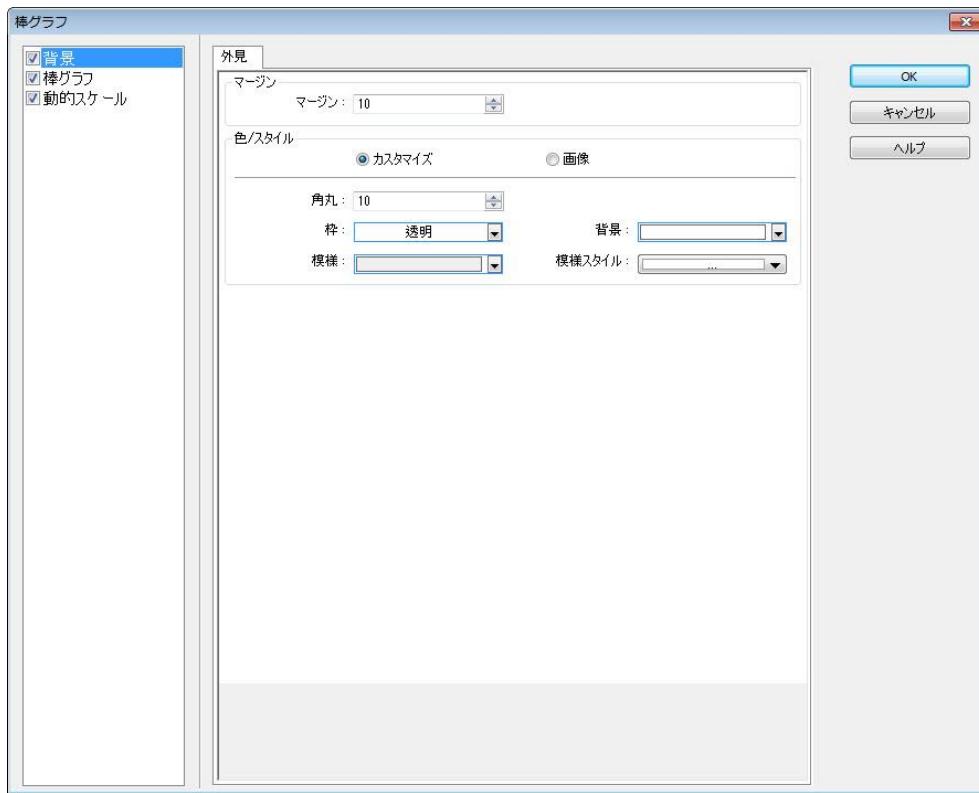
13.15.3. 複合型設定

cMT シリーズには棒グラフの複合型設定を提供し、一回で関連する複数のオブジェクトを設定することができます。棒グラフには[背景]、[棒グラフ]、[動的スケール]を同時に設定することができます。



👉 動的スケールは《Ch13.45 動的スケール》をご参照ください。

背景の設定



設定

マージン

記述

背景の縁とオブジェクトの間の距離を設定します。

色/スタイル

カスタマイズ



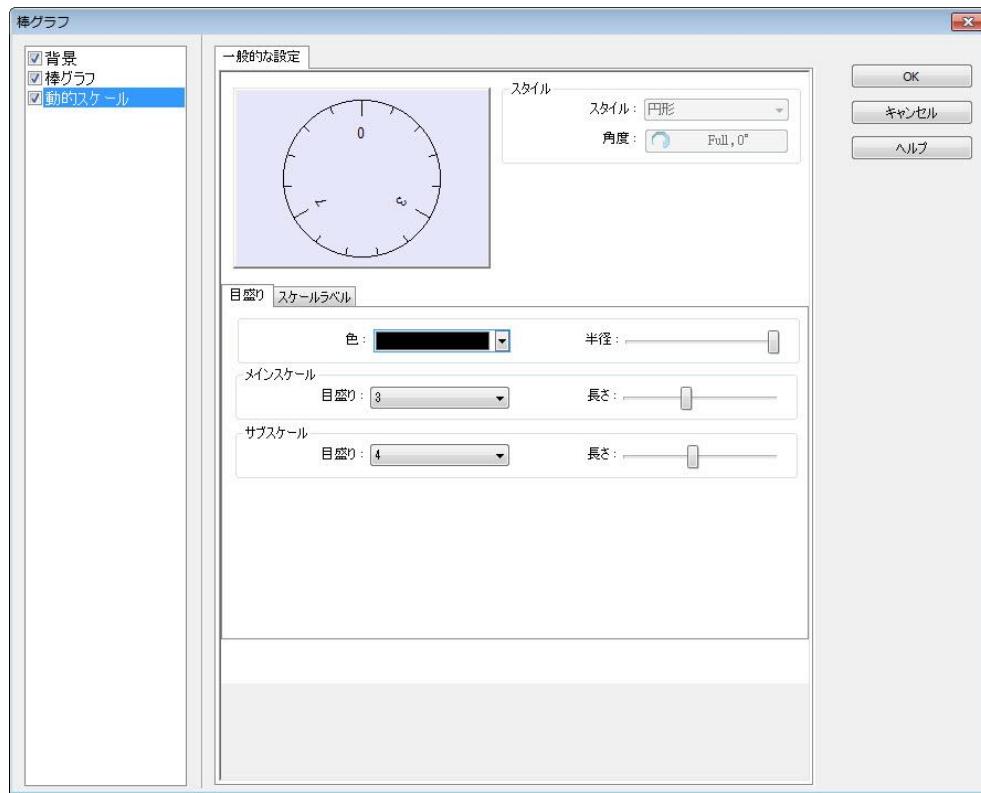
模様スタイルと色を選択して、背景をカスタマイズします。

画像



内蔵の背景画像を使用するか、画像ライブラリ内の画像を使用してもいいです。

動的スケール



設定	記述
スタイル	棒グラフの設定によります。
目盛り	[メインスケール]と[サブスケール]の分割数を設定します。スタイルは円形の場合、スケールの半径と長さを設定することができます。
スケールラベル	スケールラベルのフォント、色、サイズと他の属性を設定します。

13.16. メーター

13.16.1. 概要

[メーター]はメーター図形の形でレジスタ内のデータを表示するオブジェクトです。

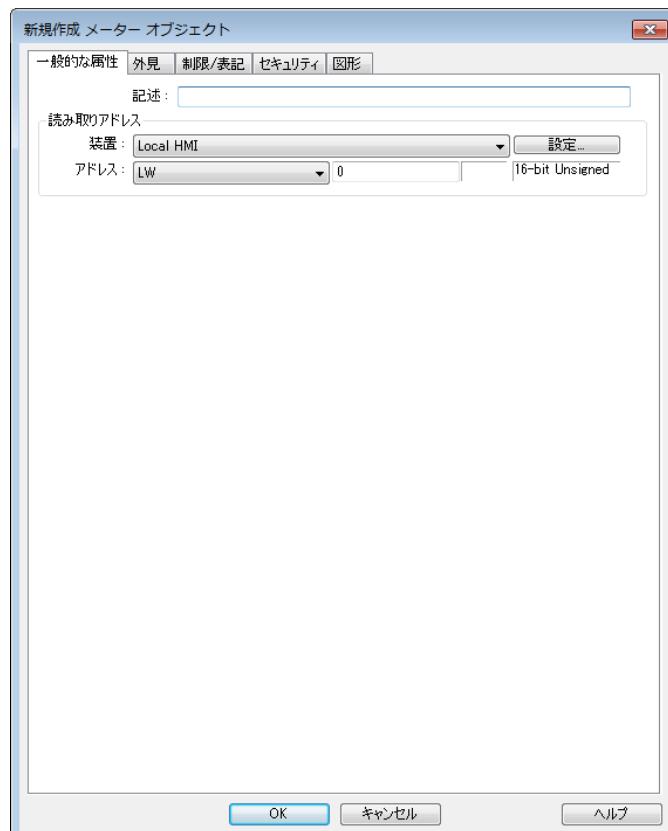
13.16.2. 設定



ツールバー上の[オブジェクト]»[オブジェクト]»[チャート]»[メーター]ボタンを押せば、[メーター]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[メーター]オブジェクトが新規作成されます。

13.16.2.1. eMT、iE、XE、mTV シリーズ

一般的な属性の設定

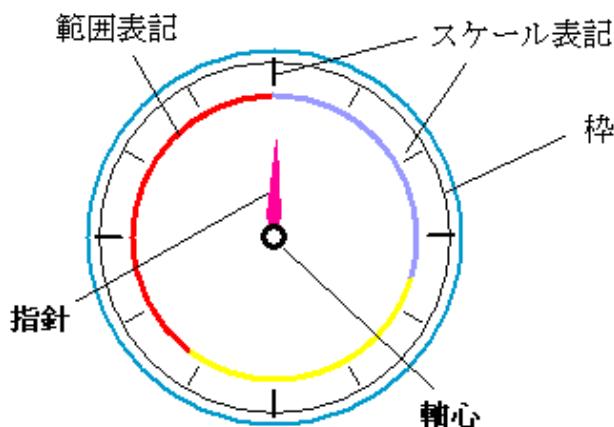
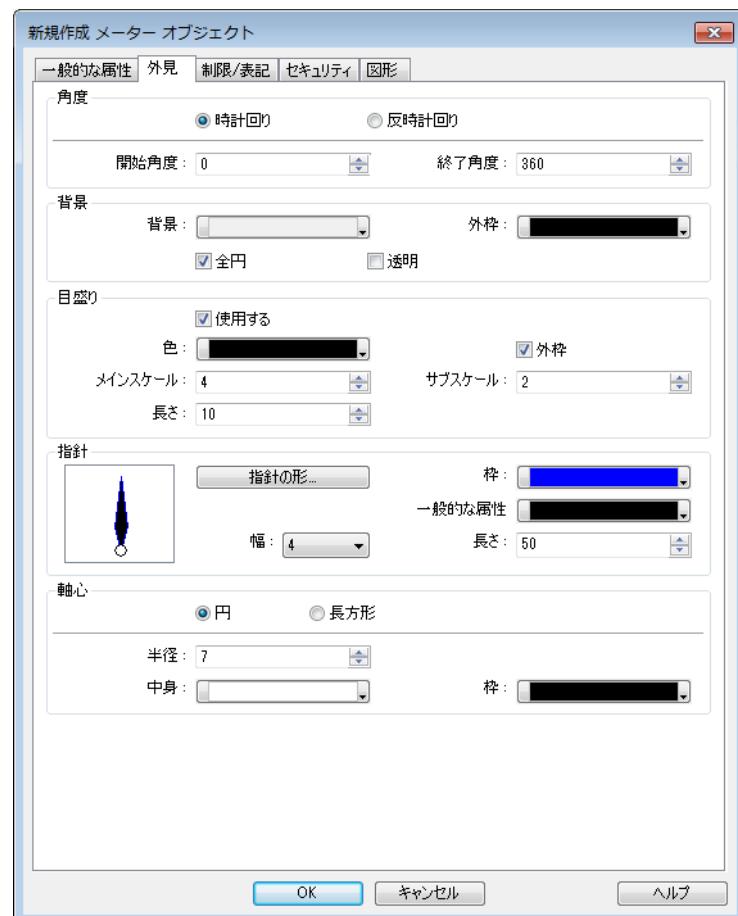


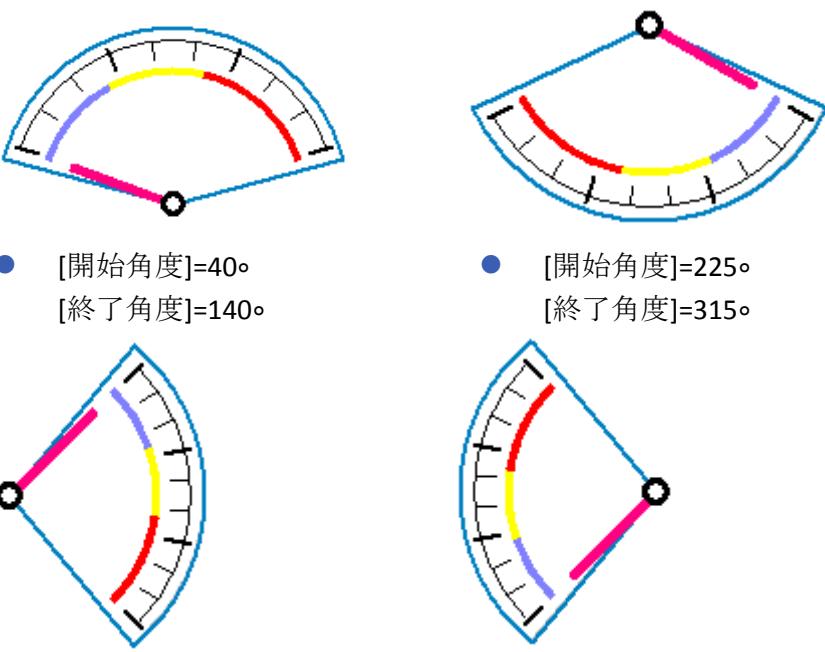
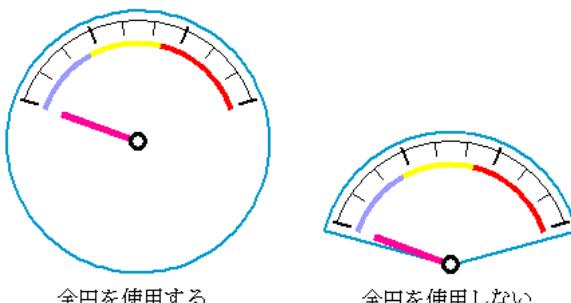
設定	記述
読み取りアド	[設定]をクリックした後、ワードレジスタ装置タイプの[装置]、[アド

レス

[レス]、[装置タイプ]、[システムレジスタ]、[インデックスレジスタ]を選択することで、メーターの表示ソースとします。ユーザーも[一般的な属性]タブでアドレスを設定することができます。

外見設定



設定	記述
角度	<p>指針の回る方向を[時計回り]または[反時計回り]に設定します。</p> <p>メーターオブジェクトは図形中心点上方を開始点とし、0度或いは360度と表示します。左へは反時計回り、右へは時計回りです。角度の設定範囲は0~360度です。異なる設定値が表示する結果は、下記の例をご参考ください。</p> <ul style="list-style-type: none"> [開始角度]=290° [終了角度]=70° [開始角度]=120° [終了角度]=240° [開始角度]=40° [終了角度]=140° [開始角度]=225° [終了角度]=315° 
背景	<p>オブジェクトの背景や円周の色を設定します。</p> <p>全円</p> <p>[全円]を選択する場合、メータオブジェクトは丸ごとの円形を表示します。選択していない場合、定義された角度範囲のみ表示します。</p>  <p>全円を使用する 全円を使用しない</p>
透明	<p>[透明]を選択する場合、メータオブジェクトは背景と枠の色を選択します。</p>
スケール表記	スケールの数量と色を設定します。

指針

指針の形、長さ、幅と色を設定します。

軸心

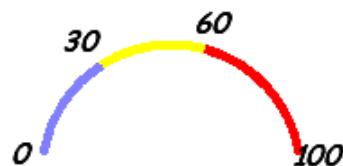
軸心のスタイルと色を設定します。

制限/表記設定**設定****記述****数値**

オブジェクトが表示する数値範囲を表示します。下記例 1 をご参照ください。

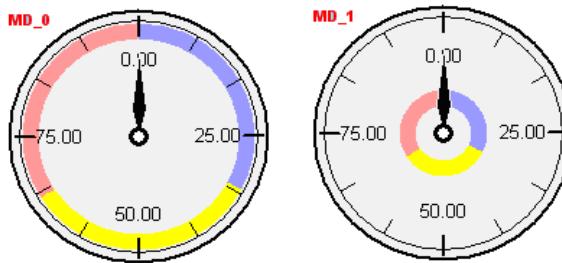
範囲

上、下限値及び表示する色と幅を設定します。

**ユーザ一定半径**

数値範囲は色と円心との距離を表示します。

例えば 80 に設定した場合 : 30 に設定した場合 :



上下限値をレジスタから取得

上下限は指定したレジスタで設定されます。下記例 2 をご参照ください。

スケールラベ

ル

スケールラベルを使用するかを選択します。



例 1

オブジェクトが表示する数値範囲を設定します。指針の的指示角度の計算方法は以下のとおりです：

$$\text{角度(度)} = \frac{\text{読み取った数値} - \text{最小値}}{\text{最大値} - \text{最小値}} \times (\text{終了角度} - \text{開始角度})$$

仮に読み取った数値は 30、開始角度 0°、終了角度 360°、最小値 0、最大値 100 の場合、指針の指示角度は：

$$\text{角度(度)} = \frac{30 - 0}{100 - 0} \times (360 - 0) = 108(\text{度})$$

例 2

上下限は指定したレジスタで設定可能です。書き込みアドレスは LW-n の場合、上/下限は下記の規則で自動的に設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMT のみ)
下限	LW-n	LW-n	LW-n
上限	LW-n+1	LW-n+2	LW-n+4

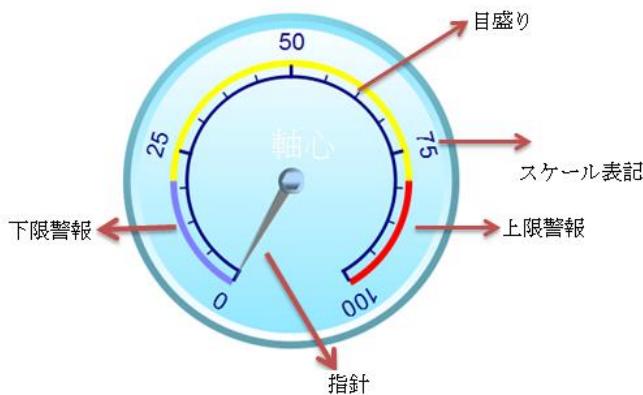
下表を例として、[レジスタアドレス]は LW-100 の場合、上/下限のアドレスは自動的に設定されます：

アドレス フォーマット	16-bit	32-bit	64-bit (cMT のみ)
下限	LW-100	LW-100	LW-100

上限	LW-101	LW-102	LW-104
----	--------	--------	--------

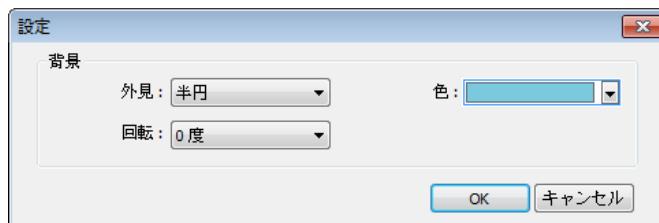
13.16.2.2. cMT シリーズ

一般的な属性の設定



設定	記述
スタイル	[カスタマイズ]、[スタイル 1]、[スタイル 2]、[クラシカル]に分けられます。文字をクリックすると、スタイルの属性を設定することができます。[カスタマイズ]を選択する場合、自分でメーターの各属性(指針の長さと幅、軸心半径、背景画像など)を設定する必要があります。
設定	[スタイル]という文字をクリックすると、指針を設定することができます

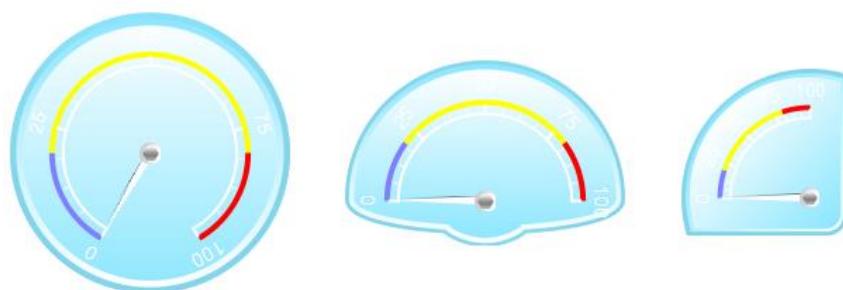
ます。



外見

メーターの表示範囲です。下図はスタイル 1 の全円、半円及び四分の一円の表示スタイル

です。



下図はスタイル 2 の全円、半円及び四分の一円の表示スタイルです。



回転

メーターの背景画像は指定した角度により時計回りします。

色

メーター図形の背景色です。

角度

メーターオブジェクトは図形中心点上方を開始点とし、0 度或いは 360 度を示しています。左へは反時計回り、右へは時計回りです。

全円

本項にチェックマークを入れると、指針は[開始角度]の角度を原点とし、[時計回り/反時計回り]の方向で一個の全円を画きます。数値の最小値及び最大値は[数値]欄の[最小値]及び[最大値]で設定します。

アニメーションを有効にする

指針が動く時に、スライド式で移動するかを選択します。本項にチェックマークを入れないと、数値が変更されたら、指針は直接に指定位置にスキップします。

数値	指針の上下限値を設定します。
読み取りアドレス	メーターが表示するデータのソースアドレス。
指針	指針及び軸心のスタイルを設定します。[カスタマイズ]モードを選択した場合、指針の的指す方向は上向けの場合のみ正確に表示されます。
制限範囲	アラーム上下限の色を設定します。 上下限値をレジスタから取得 アラーム上下限の数値は指定したレジスタで設定します。上記例 2 をご参照ください。
スケール	スケールの表記間隔及び色を設定します。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.17. トレンド図

13.17.1. 概要

[トレンド図]は[データサンプリング]に設定したデータを連続した線でトレンド傾向グラフを描くオブジェクトです。

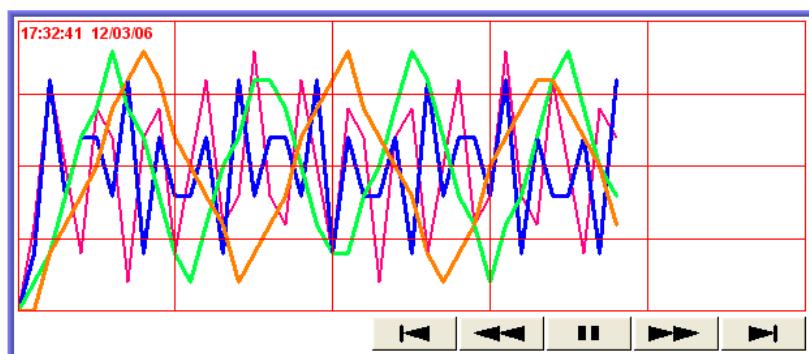
13.17.2. 設定



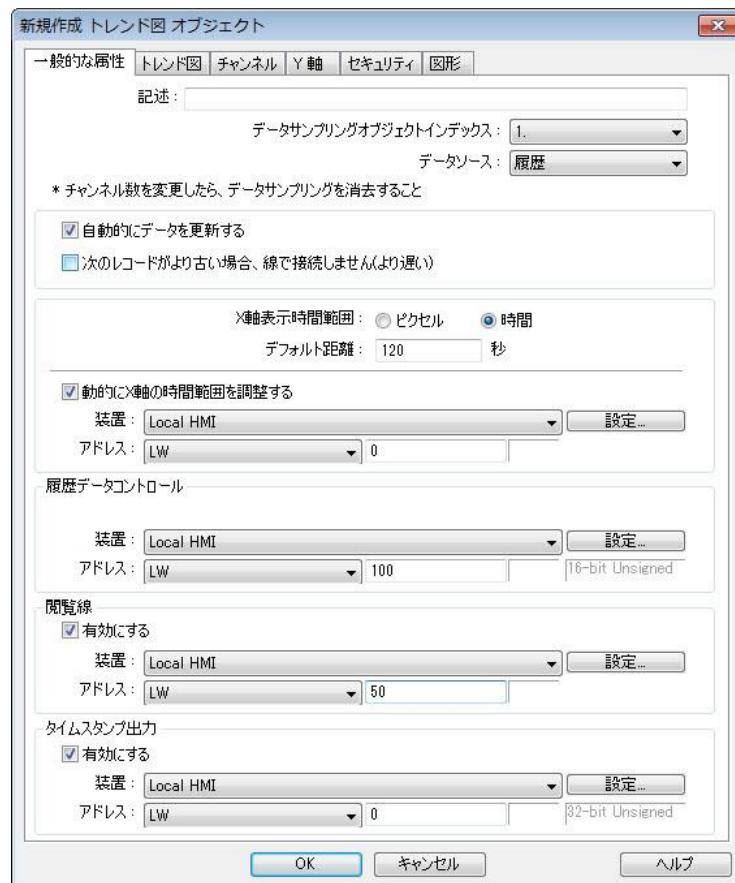
ツールバー上の[オブジェクト]»[データ/履歴]タブをクリックし、[トレンド図]ボタンを押せば、[トレンド図]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[トレンド図]オブジェクトが新規作成されます。

13.17.2.1. eMT、iE、XE、mTV シリーズ

一般的な属性の設定



ボタン	記述
	最初のサンプリングしたデータを表示します。
	一つ前のサンプリングデータを表示します。
	画面の自動的にスクロールする機能を一時停止します。新しいサンプリングデータが発生しても、画面はスクロールしなく、画面範囲に超えた最新のサンプリングデータも表示しません。
	画面の自動的にスクロールする機能を起動します。新しいサンプリングデータが発生した時、画面はスクロールし最新のサンプリングデータを表示します。
	一つ後のサンプリングしたデータを表示します。
	最新のサンプリングデータを表示します。



設定	記述
データサンプリングオブジェクトインデックス	[データサンプリング]オブジェクトを選択し、グラフを描くのに必要なデータのソースとします。
自動的にデータを更新する	本項目にチェックマークを入れると、[履歴]モードでシステムは 10 秒ごとに自動的にトレンド図をリフレッシュします。チェックマークを入れないと、ウインドウが切り替えられた場合のみ、トレンド図がリフレッシュされます。
次のレコードがより古い場合、線で接続しません(より遅い)	本項目にチェックマークを入れると、HMI の時間が前に調整され、それにデータサンプリングが継続されている場合、システムは 2 レコードの間に線を書いて接続しません。(表示の速度が遅延されます)
データソース	データソースのモードを選択します。[リアルタイム]または[履歴]を選択することができます。
リアルタイム	[データサンプリング]オブジェクトは HMI が起動された後、固定したレコード数のサンプリングデータを表示することができます。サ

サンプリングデータの表示数は[データサンプリング]オブジェクトの[最大データ数(リアルタイムモード)]で設定されます。ここで設定した数量を超えた場合、より古いデータは画面から削除されます。もし他の日付のデータやより古いデータを閲覧する必要がある場合、[履歴]モードを選択してください。

[一時停止]機能で画面の更新を中止することができるが、画面の更新だけが中止され、[データサンプリング]オブジェクトのサンプリング動作は中止されません。

履歴

履歴データは[データサンプリング]オブジェクトが日付で分類、保存したサンプリングデータです。

[履歴]モードを使用すれば、[データサンプリングオブジェクトインデックス]で表示する履歴データを選択し、また[履歴データコントロール]で異なる日付の履歴データを確認することができます。

HMI はサンプリングデータの履歴データを時間順で配列し、最新日付けのデータをレコード 0(一般的には今日セーブしたサンプリングデータのことを指す)、二番目新しいデータをレコード 1 のように順位付けます。

[履歴コントロール]で指定したレジスタ内の数値は 0 の場合、[トレンド図]オブジェクトはレコード 0 のデータを表示し、レジスタ内の数値は 1 の場合、レコード 1 のデータを表示します。つまり、レジスタ内の数値は n の場合、レコード n のデータが表示されます。

例：履歴コントロールレジスタは LW-0 の場合、仮に現在、[データサンプリング]オブジェクトが保存したサンプリングデータは時間順で pressure_20061120.dtl、pressure_20061123.dtl、

pressure_20061127.dtl、pressure_20061203.dtl との 4 レコードのデータがあり、それに本日の日付は 2006/12/3 の場合、LW-0 内の数値内容に基づき、[トレンド図]に表示されるサンプリングデータは下記のようになります：

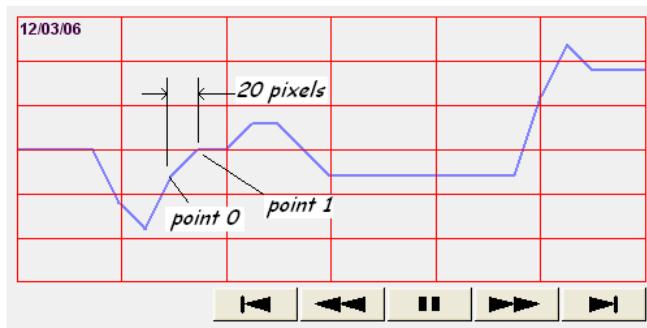
LW-0 内の数値	表示されるサンプリングデータ
0	pressure_20061203.dtl
1	pressure_20061127.dtl
2	pressure_20061123.dtl
3	pressure_20061120.dtl

[オプションリスト]オブジェクトに合わせて、データソースは[履歴データ日付]を選択すれば、全ての履歴データが日付で分類され、オプションリストオブジェクトに表示されます。詳細は《13.29 オプシ

ヨンリスト》をご参考ください。

ピクセル

下記のように、2個サンプリング点の間の距離を設定します。



時間

下記のように、X軸が表示する時間範囲を表示します。



[トレンド図]タブの[罫線]項目で[時間スケール]機能を使用することができます。

動的に2個サンプリング点の間隔距離を調整する/動的にX軸の時間範囲を調整する

32-bit フォーマットのワードレジスタを指定し、オンラインで[ピクセル]または[時間]の距離を設定します。本機能を使用すると、レジスタの中に数値が記入されていない場合、デフォルト数値で距離を決めます。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

データを自動更新する

本項を使用すると、毎回[履歴モード]の[トレンド図]オブジェクトの所在するウインドウをオープンしたら、オブジェクトの画面は毎秒ごとに更新されます。下記をご留意ください：

- 自動更新機能の状態は画面スクロールボタンで見られます：
図示は■の場合、トレンド図オブジェクトのデータは自動的に更新されると示しています。
図示は▶の場合、トレンド図オブジェクトのデータの更新が中止されたと示しています。
- 前へスクロールし、より古いデータを見ると、[データを自動更

新する]機能が取り消されます。この場合、コントロールボタンの図示は▶です。

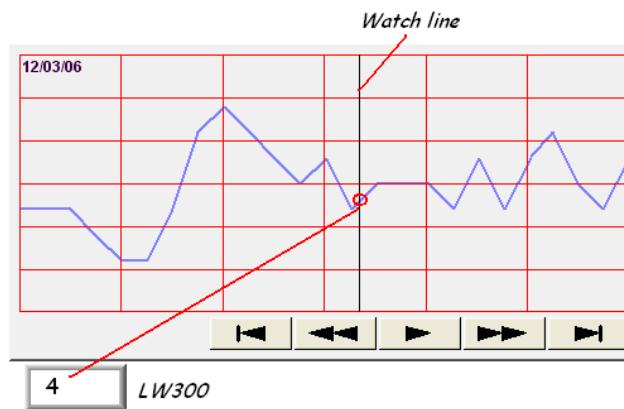
- [データを自動更新する]にチェックマークを入れると、この前では画面スクロールボタンで自動更新を起動/中止したことがあるかどうかを問わず、このウインドウに切り替えれば、必ず画面が自動的に更新されます。
例：仮にオブジェクトは[データを自動的に更新する]機能を使用している場合、前へスクロールして旧データを見たら、自動更新機能は中止されます。この際に、一回他のウインドウに切替してまた本ウインドウに戻れば、オブジェクト画面は依然に自動的に更新されます。
- もしプロジェクトを作成した際に、[データを自動的に更新する]機能を使用していない、その後にHMIに本機能を使用したい場合、▶ボタンを押せばいいです。注意すべきなのは、この際の自動更新機能はウインドウが切り替えられるたびに使用中止になります。つまり、一旦他のウインドウに切替したら、また本来のウインドウに戻っても、オブジェクトの画面は更新が中止されたままです。

コントロール を一時停止

一時停止ビットレジスタをONにする場合、トレンド図画面の更新が中止されるが、データサンプリングのサンプリング動作は中止されません。本項は[データソース]が[リアルタイム]モードを採用する場合のみ表示されます。

閲覧線

本項を使用すると、[トレンド図]オブジェクトは垂直な閲覧線が生じ、閲覧線上のサンプリングデータを指定したアドレスに出力し、下記のように[数値]オブジェクトで表示することができます。



[閲覧線]も複数のチャンネル内のサンプリングデータを出力することができます。システムは[データサンプリング]オブジェクトで定義したサンプリングデータフォーマットを基づき、指定したアドレス内のサンプリングデータを[閲覧線]が定義した開始アドレスへ順

次に書き込みます。例えば、仮に[データサンプリング]のサンプリングデータが複数の異なるフォーマットのデータを含め、それで LW-300 は[閲覧]機能が定義したレジスタである場合、閲覧線が表記したサンプリングデータのエクスポート位置は下記のようになります：

レジスタ	チャンネル	データフォマット
LW-300	0	16-bit Unsigned (1 word)
L-301	1	32-bit Unsigned (words)
LW-303	2	32-bit float (words)
LW305	3	16-bit Signed (1 word)

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

タイムスタンプ

出力

本項を使用すると、システムは一番目のサンプリング点でのサンプリング時刻を時間原点としてカウントし始め、最新のサンプリング点の積算秒数を[タイムスタンプ出力アドレス+2]に出力します。

オブジェクト上のカーブをクリックする際に、一番近づいているサンプリング点での積算秒数を[タイムスタンプ出力アドレス]に出力することができます。

データサンプリングオブジェクトの[リアルタイムデータアドレスをクリアする]機能をトリガーしたら、現在のサンプリングデータを消去する以外、サンプリング時間原点をリセットすることもできます。

注意:[タイムスタンプ出力アドレス]と[タイムスタンプ出力アドレス+2]は全部 32-bit フォーマットでなければなりません。[タイムスタンプ出力アドレス+2]はリアルタイムモードのみに適用し、[タイムスタンプ出力アドレス]はリアルタイム及び履歴モード両方でも適用します。

トレンド図タブの[相対時間モード]が選択された場合のみ、本機能を使用することができます。

トレンド図の設定



設定

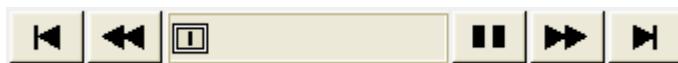
記述

透明/枠/背景

オブジェクトの枠と背景の色を設定します。

画面スクロールボタン

下記のような画面コントロールボタンを有効/無効するのを選択します。



野線

スケール線の数量及び色を設定します。[一般的な属性]タブの[両サンプリング点の間の距離]または[X軸表示時間範囲]の設定違いによって異なります。システムはこれらの設定に基づいて、自動的に垂直スケール線の数量を計算します。

X 軸間隔

垂直スケール線の数量を設定します。

- [両サンプリング点の間の距離]を基づく：
2本ごとの垂直スケール線で含まれるサンプリング点の数を選択します。
- [X軸表示時間範囲]を基づく：
2本ごとの垂直スケール線の間で表示する時間間隔を設定します。

X 軸-中身等分

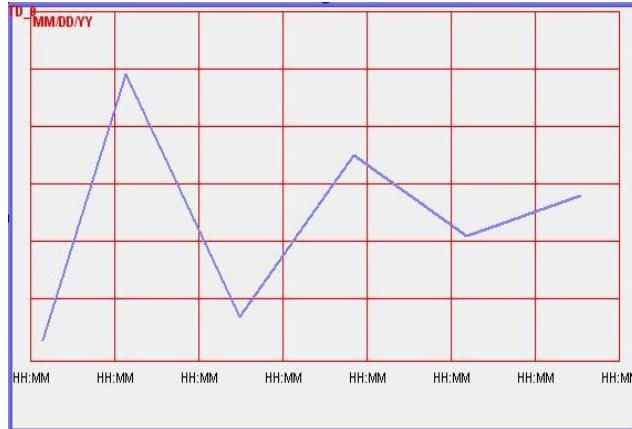
垂直スケール線の数量を設定します。

Y 軸-中身等分

水平スケール線の数量を設定します。

タイムスケール

[表示]を選択し、下記のように X 軸にタイムスケールを表示します。



フォーマット

タイムスケールフォーマットを HH:MM もしくは HH:MM:SS を選択します。

フォント/色/サイズ

タイムスケールが表示する文字のフォント、色、サイズを選択します。

文字サイズのデフォルトは 8 です。

時間/日付

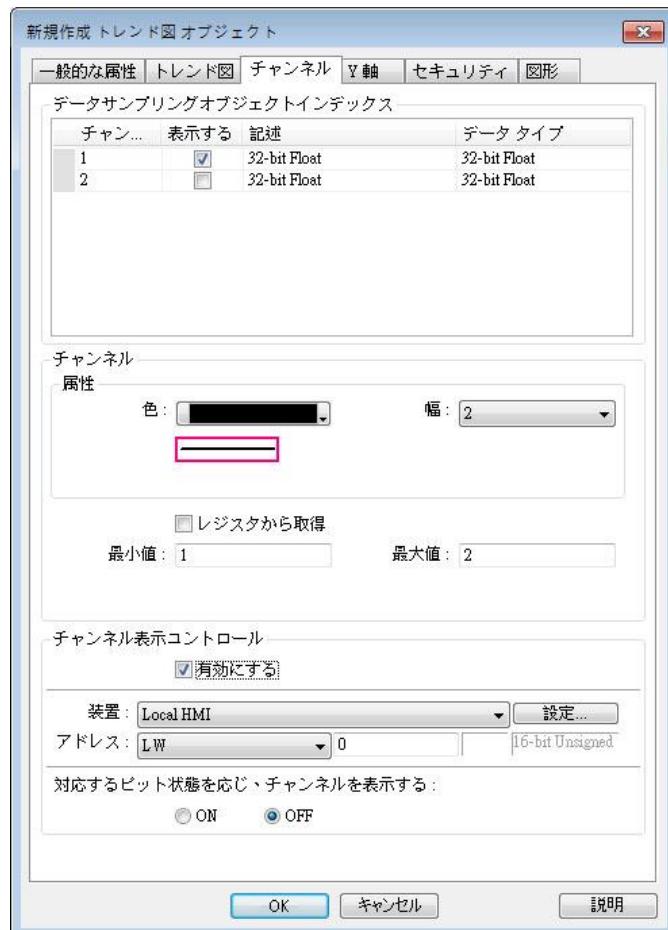
● 相対時間モード

本項にチェックマークを入れると、HMI は一番目のデータの時間を記録し、タイムスケールの SSSS に合わせて使用すると、相対の時間を表示することができます。この場合、タイムスケールの[日付]は使用できません。

時間/日付	
<input checked="" type="checkbox"/> 相対時間モード	
<input type="checkbox"/> 時間	
<input checked="" type="checkbox"/> 日付	MM/DD/YY
色:	<input type="color" value="#FF0000"/>

最新のサンプリングデータの時間情報はオブジェクトの左上隅に表示され、本項で時間/日付の表示フォント及び色を設定します。

チャンネルの設定



設定

描述

Y 軸比率

Y 軸が主軸或いは補助軸に設定されたのを表示します。《Y 軸の設定》の説明をご参照ください。

チャン...	表示する	記述	データ タイプ	Y軸比率
1	True	channel A	16-bit Unsigned	補助軸
2	True	channel B	16-bit Unsigned	主軸
3	True	channel C	16-bit Unsigned	補助軸

チャンネル

各サンプリング線分の色、幅、スタイルを決めます。同時に最大 64 チャンネルをサポートすることができます。

レジスタから取得

- [レジスタから取得]にチェックマークを入れない場合

[最小値]と[最大値]は各カーブが描いたサンプリングデータの最小値と最大値を示しています。つまり、もあるサンプリング線分のサンプリング最小値は 50、最大値は 100 の場合、[最小値]と[最大値]は 50 と 100 に設定する必要があり、それでこそ全てのサンプリングデータが完全にオブジェクトに描かれます。

- [レジスタから取得]にチェックマークを入れる場合

上下限を指定したレジスタで設定することができます。書き込みアドレ

スは LW-n の場合、上/下限は自動的に以下のとおりに設定されます：

アドレスフォーマット	16-bit	32-bit	64-bit (cMTのみ)
下限	LW-n	LW-n	LW-n
上限	LW-n+1	LW-n+2	LW-n+4

下表を例として、[レジスタアドレス]は LW-100 の場合、上/下限のアドレスは自動的に以下のとおりに設定されます：

アドレスフォーマット	16-bit	32-bit	64-bit (cMTのみ)
下限	LW-100	W-00	LW-100
上限	LW-101	LW102	LW-104

本設定は常にトレンド図の拡大や縮小に用いられます(cMT シリーズに適用しない)。下記例 1 をご参照ください。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

チャンネル表示

[使用]にチェックマークを入れると、このアドレス内の各ビットは各チャンネルの表示をコントロールします。Bit-1 はチャンネル 1 をコントロールし、Bit-2 はチャンネル 2 をコントロールし、このように類推します。例を挙げれば、5 つのチャンネルを作成し、またチャンネルを設定するアドレスを LW-0 にする場合、各チャンネルは下記アドレスにコントロールされます：

チャンネル	コントロールアドレス	ビット状態	表示するか
1	LW_bit-000	OFF	YES
2	LW_bit-001	ON	NO
3	LW_bit-002	ON	NO
4	LW_bit-003	OFF	YES
5	LW_bit-004	OFF	ES

[チャンネルを表示しない]を選択した場合、そのチャンネルはチャンネルコントロールアドレスを使用しません。上表を例にすると：合計 5 つのチャンネルがあり、3 番目チャンネルはトレンド図に表示しないと選択されたので、最大 4 つのチャンネルが同時にトレンド図に表示します。それに、コントロールアドレスは LW_bit-000~003 のみ使用されます。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

Y 軸の設定



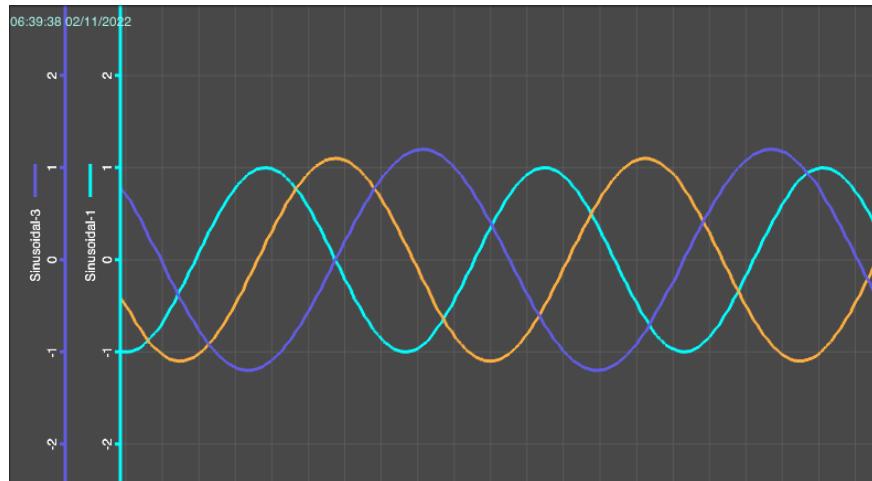
設定	記述
Y 軸比率	Y 軸を主軸或いは補助軸に設定することができます。または[無]を選択し、Y 軸を表示しないようにします。最大 32 個の Y 軸が表示できます。
スケールフォント	Y 軸スケールのフォント、色、サイズを設定します。
動的に Y 軸を調整する	Y 軸を表示するかをコントロールします。コントロールアドレスは LW-50 の場合、チャンネル 1 をコントロールするアドレスは LW_Bit 5000 で、チャンネル 2 をコントロールするアドレスは LW_Bit 5001 で、このように類推します。
動的に主軸を調整する	主軸を変更することができます。LW-80 に 1 を記入すれば、主軸はチャンネル 1 になり、2 を記入すれば、主軸はチャンネル 2 になります。このように類推します。

▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

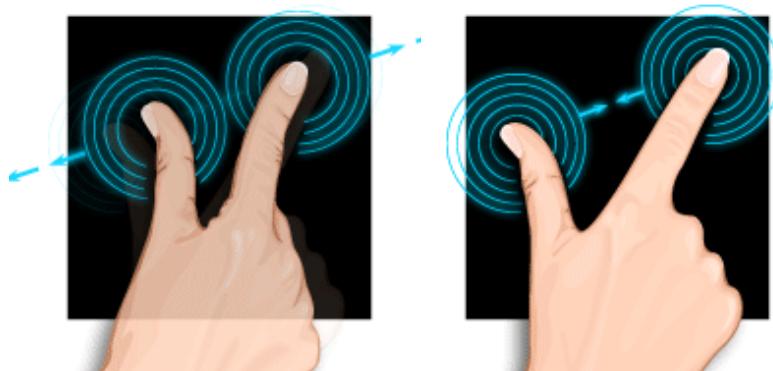
⬇ このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.17.2.2. cMT シリーズ

一般的な属性の設定



cMT シリーズのトレンド図は[リアルタイム]モードと[履歴]モードを融合し、オブジェクトで左側へドラッグ&ドロップしたら履歴データが見られ、右側へドラッグ&ドロップしたら最新のサンプリングデータが見られます。また、両指でトレンド図を拡大/縮小することができます。



トレンド図を拡大する トレンド図を縮小する

☞ cMT シリーズのデータサンプリングの保存方式については、《8 データサンプリング》をご参考ください。



設定	記述
データサンプル	
リングオブジェクトインデックス	[データサンプリング]オブジェクトをデータソースとして選択します。
ミリメーター	eMT、iE、XE、mTV シリーズと同じです。
時間	eMT、iE、XE、mTV シリーズと同じです。
閲覧線	eMT、iE、XE、mTV シリーズと同じです。
Y 軸のスクローリングを無効にする	有効にすると、指でトレンド図をドラッグすれば、X 軸範囲だけが変更されます。Y 軸をズームをすることができるが、その位置がずっとオブジェクトの真ん中に維持されます。
ズームを無効にする	有効にすると、トレンド図をズームすることができなくなり、右上隅の アイコンを押してもズームできません。ただし、オブジェクトを少し上下に移動させることができます。
カスタムファイル名を表示する	データサンプリングに[カスタムファイル管理]を使用した場合、[カスタムファイル名を表示する]を有効にすると、トレンド図の左上隅にカスタムのファイル名が表示されます。

トレンド図の設定

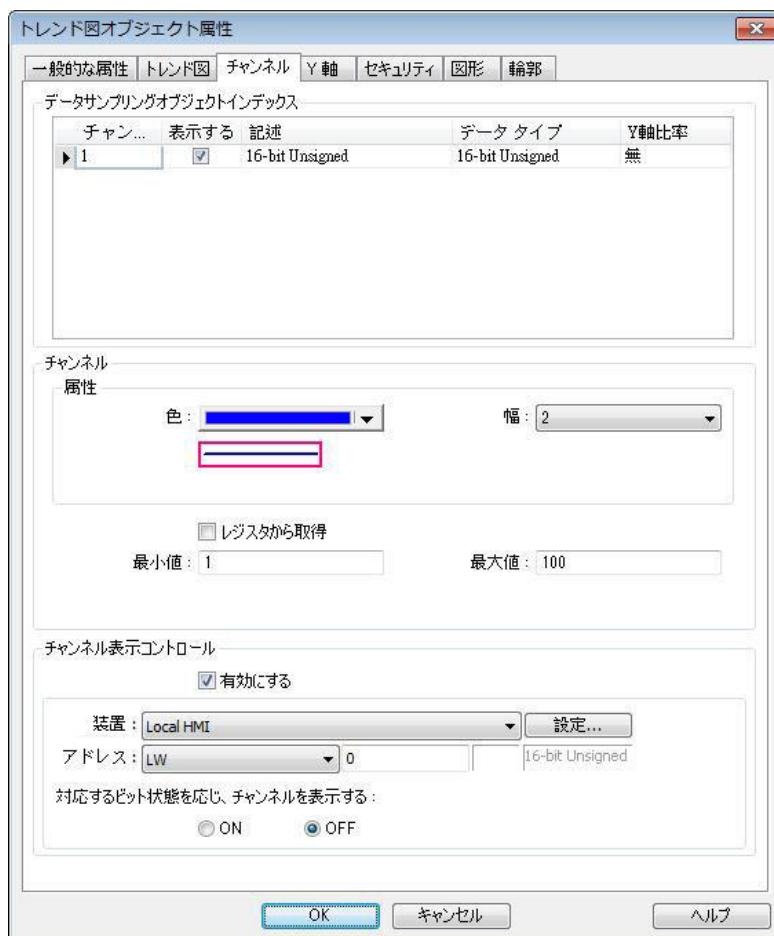


設定	記述
透明/枠/背景	オブジェクトの枠と背景の色を設定します。
野線	スケール線の数量及び色を設定します。[一般的な属性]タブの[両サンプリング点の間の距離]または[X軸表示時間範囲]の設定違いによって異なります。システムはこれらの設定に基づいて、自動的に垂直スケール線の数量を計算します。
X 軸間隔	垂直スケール線の数量を設定します。 <ul style="list-style-type: none">● [両サンプリング点の間の距離]を基づく： 2本ごとの垂直スケール線で含まれるサンプリング点の数を選択します。● [X軸表示時間範囲]を基づく： 2本ごとの垂直スケール線の間で表示する時間間隔を設定します。
X 軸-等分	垂直スケール線の数量を設定します。
Y 軸等分	水平スケール線の数量を設定します。
タイムスケール	[時刻]或いは[日付]を選択し、X軸にタイムスケールを表示します。 異なるフォーマットが選択できます。 フォントサイズのデフォルトは8です。

時間/日付**相対時間モード**

本項目にチェックマークを入れると、HMIは一番目のデータの時間を記録し、タイムスケールの SSSS に合わせて使用すれば、相対時間を表示できます。この場合、タイムスケールの日付が使用できません。

最新のサンプリングデータの時間情報はオブジェクトの左上隅に表示され、本項で時間/日付の表示フォント及び色を設定します。

チャンネルの設定**設定****記述****Y 軸比率**

Y 軸が主軸或いは補助軸に設定されたのを表示します。《Y 軸の設定》の説明をご参照ください。最大 32 個の Y 軸を表示できます。

チャン...	表示する	記述	データ タイプ	Y 軸比率
1	True	channel A	16-bit Unsigned	補助軸
2	True	channel B	16-bit Unsigned	主軸
3	True	channel C	16-bit Unsigned	補助軸

チャンネル

各サンプリング線分の色、幅、スタイルを決めます。同時に最大 64 チャンネルをサポートすることができます。

レジスタから取得

● [レジスタから取得]にチェックマークを入れない場合

[最小値]と[最大値]は各カーブが描いたサンプリングデータの最小値と最大値を示しています。つまり、もあるサンプリング線分のサンプリング最小値は 50、最大値は 100 の場合、[最小値]と[最大値]は 50 と 100 に設定する必要があり、それでこそ全てのサンプリングデータが完全にオブジェクトに描かれます。

● [レジスタから取得]にチェックマークを入れる場合

上下限を指定したレジスタで設定することができます。書き込みアドレスは LW-n の場合、上/下限は自動的に以下のとおりに設定されます：

アドレスフォーマット	16-bit	32-bit	64-bit (cMT のみ)
下限	LW-n	LW-n	LW-n
上限	LW-n+1	LW-n+2	LW-n+4

下表を例として、[レジスタアドレス]は LW-100 の場合、上/下限のアドレスは自動的に以下のとおりに設定されます：

アドレスフォーマット	16-bit	32-bit	64-bit (cMT のみ)
下限	L-100	LW-100	LW-100
上限	LW-101	LW-102	LW-104

本設定は常にトレンド図の拡大や縮小に用いられます(cMT シリーズに適用しない)。下記例 1 をご参照ください。

チャンネル表示

[使用]にチェックマークを入れると、このアドレス内の各ビットは各チャンネルの表示をコントロールします。Bit-1 はチャンネル 1 をコントロールし、Bit-2 はチャンネル 2 をコントロールし、このように類推します。例を挙げれば、5 つのチャンネルを作成し、またチャンネルを設定するアドレスを LW-0 にする場合、各チャンネルは下記アドレスにコントロールされます：

チャンネル	コントロールアドレス	ビット状態	表示するか
1	LW_bit-000	OFF	YES
2	LW_bit-001	ON	NO
3	LW_bit-002	N	NO
4	LW_bit-003	OFF	YES
5	LW_bit-004	OFF	ES

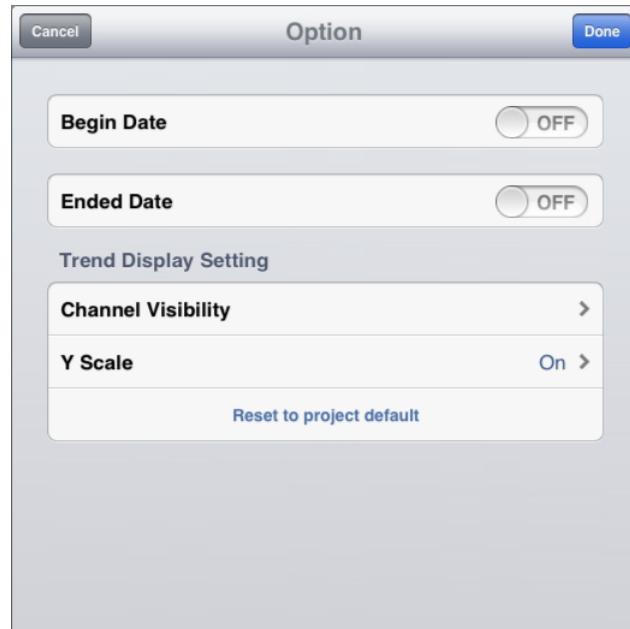
[チャンネルを表示しない]を選択した場合、そのチャンネルはチャンネルコントロールアドレスを使用しません。上表を例にすると：合計 5 つのチャンネルがあり、3 番目チャンネルはトレンド図に表示しないと選択されたので、最大 4 つのチャンネルが同時にトレンド図に表示します。それに、コントロールアドレスは LW_bit-000~003 のみ使用されます。

Y 軸の設定



オブジェクト上の Y 軸は当チャンネルのスケールを表示します。[トレンド図]設定タブの罫線に [表示]を選択した場合のみ、本機能を使用することができます。Y 軸比率も cMT Viewer で調整することができます。下記手順をご参考ください。

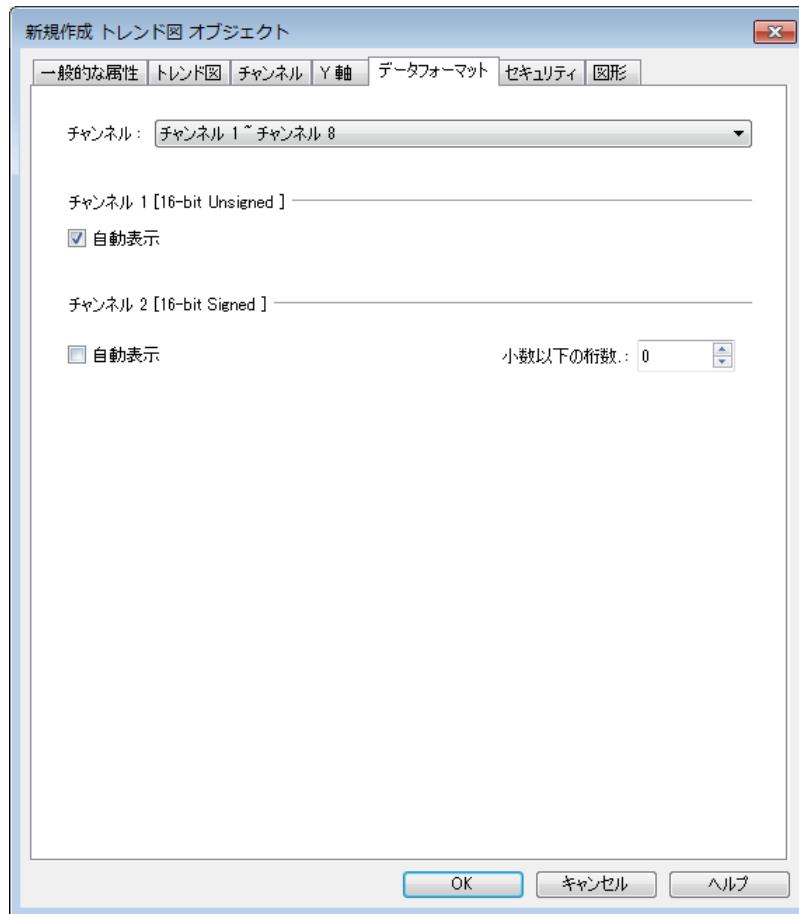
1. [トレンド図]オブジェクト右上隅の ボタンをクリックします。
2. [Trend Display Setting]で Y Scale を選択します。



3. Y Scale を表示するチャンネルを選択します。



データフォーマットの設定



設定	記述
自動表示	自動表示にチェックマークを入れると、当該チャンネルの数値は調整されません。
小数点以下の 桁数	本項を調整すると、当該チャンネルではまず数値を変換してから、変換後の新たな数値でトレンド図に表示します。 例: 小数点以下の桁数を 1 にした場合、サンプリングしたソース数値が 45 だったら、トレンド図では 4.5 に描かれます。

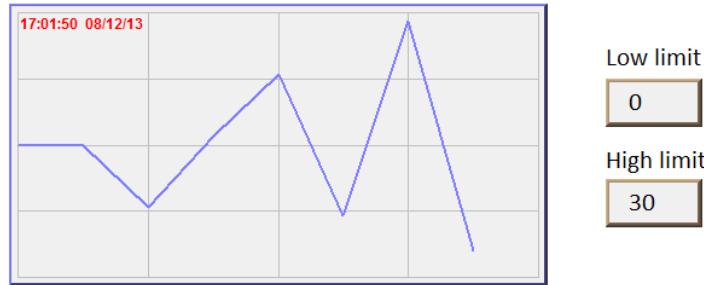
例 1

本例はトレンド図の拡大/縮小する方法を説明します。ただし、cMT シリーズに適用しません。本機能を使用するには、ユーザーはチャンネルの上限/下限設定欄で**最大/最小値をレジスタから取得**にチェックマークを入れる必要があります。それでアドレスを **LW-n** に設定した場合、**LW-n** は最小値をコントロールし、**LW-n+1** は最大値をコントロールします。

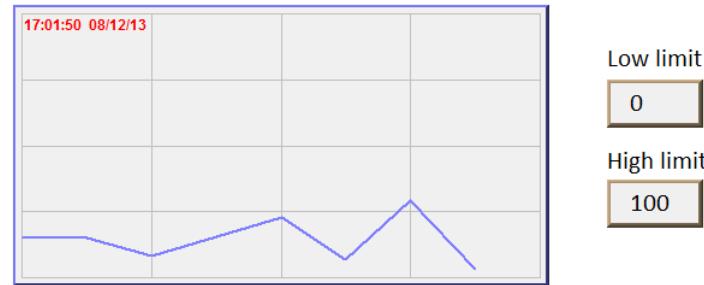


最大/最小値をレジスタ LW-0 から取得と設定し、二つの[数値]オブジェクトを最大値及び最小値のコントロール用に作成し、アドレスを LW-0 と LW-1 に設定します。

仮にデータの大きさが 0~30 の間にある場合、最小値のコントロールアドレスに 0 を入力し、最大値のコントロールアドレスに 30 を入力すれば、トレンド図は下記のようになります。



トレンド図を縮小したい場合、最大値に大きめの数値を入力すればいいです。例えば：最小値のコントロールアドレスに 0 を入力し、最大値のコントロールアドレスに 100 を入力すれば、トレンド図は下記のようになります。



トレンド図を拡大したい場合、最大値に小さめの数値を入力すればいいです。例えば：最小値のコントロールアドレスに 0 を入力し、最大値のコントロールアドレスに 20 を入力すれば、トレンド図は下記のようになります。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.18. 履歴データ表示

13.18.1. 概要

[履歴データ表示]は既に保存されたサンプリングデータを表示するオブジェクトです。トレンド図との相違点は、[履歴データ表示]オブジェクトはリスト式で直接にこれらのデータ内容を表示します。下図を履歴データリストの例にします。

番号	時間	日付	ch.1
14	15:54	02/03/16	
13	15:53	02/03/16	
12	15:53	02/03/16	
11	15:53	02/03/16	
10	15:53	02/03/16	
9	15:53	02/03/16	
8	15:53	02/03/16	
7	15:53	02/03/16	
6	15:53	02/03/16	

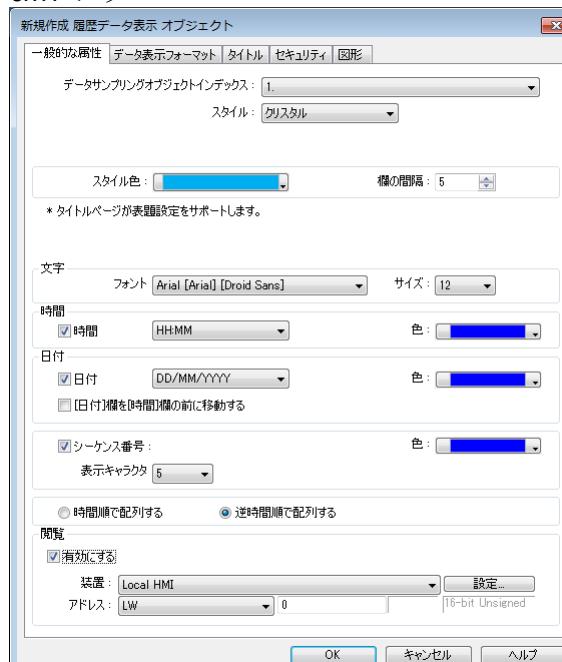
13.18.2. 設定



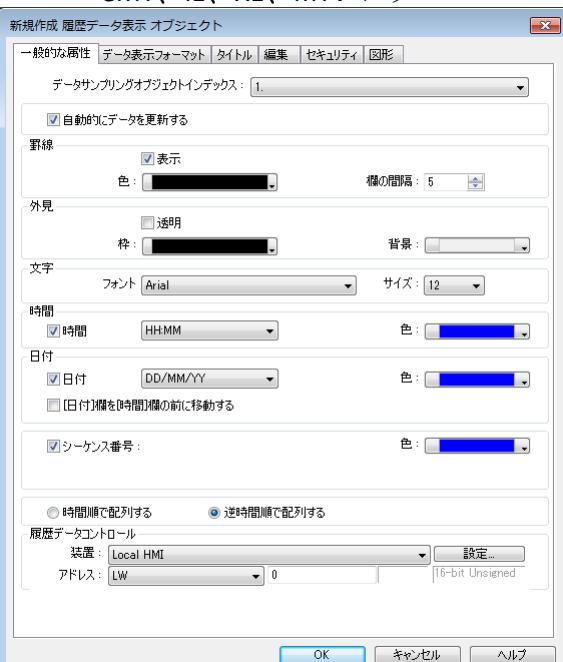
ツールバー上の[オブジェクト] » [データ/履歴]タブをクリックし、[履歴データ表示]ボタンを押せば、[履歴データ表示]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[履歴データ表示]オブジェクトが新規作成されます。

一般的な属性の設定

cMT シリーズ

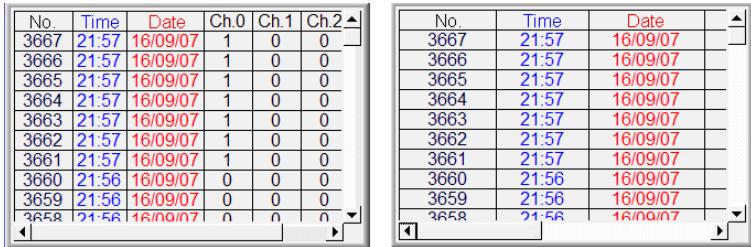


eMT、iE、XE、mTV シリーズ



設定

記述

データサンプ	[データサンプリング]オブジェクトを選択し、所要するデータのソースとします。																																																																																																																																				
自動的にデータを更新する	本項目にチェックマークを入れると、システムは十秒ごとに自動的にデータを更新します。そうしなければ、ウインドウが切り替えされてから、データが更新されます。																																																																																																																																				
スタイル	履歴データ表示オブジェクトの表示スタイルです。																																																																																																																																				
罫線	リストに枠線を使用するかを選択します。																																																																																																																																				
色	枠線の色を設定します。																																																																																																																																				
欄の間隔	本設定値で各欄の間隔を設定します。下図は異なる[欄の間隔]で設定した場合の表示状況です。																																																																																																																																				
 <table border="1"> <thead> <tr> <th>No.</th> <th>Time</th> <th>Date</th> <th>Ch.0</th> <th>Ch.1</th> <th>Ch.2</th> </tr> </thead> <tbody> <tr><td>3667</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3666</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3665</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3664</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3663</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3662</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3661</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3660</td><td>21:56</td><td>16/09/07</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>3659</td><td>21:56</td><td>16/09/07</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>3658</td><td>21:56</td><td>16/09/07</td><td>0</td><td>0</td><td>0</td></tr> </tbody> </table> <table border="1"> <thead> <tr> <th>No.</th> <th>Time</th> <th>Date</th> <th>Ch.0</th> <th>Ch.1</th> <th>Ch.2</th> </tr> </thead> <tbody> <tr><td>3667</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3666</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3665</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3664</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3663</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3662</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3661</td><td>21:57</td><td>16/09/07</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>3660</td><td>21:56</td><td>16/09/07</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>3659</td><td>21:56</td><td>16/09/07</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>3658</td><td>21:56</td><td>16/09/07</td><td>0</td><td>0</td><td>0</td></tr> </tbody> </table>	No.	Time	Date	Ch.0	Ch.1	Ch.2	3667	21:57	16/09/07	1	0	0	3666	21:57	16/09/07	1	0	0	3665	21:57	16/09/07	1	0	0	3664	21:57	16/09/07	1	0	0	3663	21:57	16/09/07	1	0	0	3662	21:57	16/09/07	1	0	0	3661	21:57	16/09/07	1	0	0	3660	21:56	16/09/07	0	0	0	3659	21:56	16/09/07	0	0	0	3658	21:56	16/09/07	0	0	0	No.	Time	Date	Ch.0	Ch.1	Ch.2	3667	21:57	16/09/07	1	0	0	3666	21:57	16/09/07	1	0	0	3665	21:57	16/09/07	1	0	0	3664	21:57	16/09/07	1	0	0	3663	21:57	16/09/07	1	0	0	3662	21:57	16/09/07	1	0	0	3661	21:57	16/09/07	1	0	0	3660	21:56	16/09/07	0	0	0	3659	21:56	16/09/07	0	0	0	3658	21:56	16/09/07	0	0	0	
No.	Time	Date	Ch.0	Ch.1	Ch.2																																																																																																																																
3667	21:57	16/09/07	1	0	0																																																																																																																																
3666	21:57	16/09/07	1	0	0																																																																																																																																
3665	21:57	16/09/07	1	0	0																																																																																																																																
3664	21:57	16/09/07	1	0	0																																																																																																																																
3663	21:57	16/09/07	1	0	0																																																																																																																																
3662	21:57	16/09/07	1	0	0																																																																																																																																
3661	21:57	16/09/07	1	0	0																																																																																																																																
3660	21:56	16/09/07	0	0	0																																																																																																																																
3659	21:56	16/09/07	0	0	0																																																																																																																																
3658	21:56	16/09/07	0	0	0																																																																																																																																
No.	Time	Date	Ch.0	Ch.1	Ch.2																																																																																																																																
3667	21:57	16/09/07	1	0	0																																																																																																																																
3666	21:57	16/09/07	1	0	0																																																																																																																																
3665	21:57	16/09/07	1	0	0																																																																																																																																
3664	21:57	16/09/07	1	0	0																																																																																																																																
3663	21:57	16/09/07	1	0	0																																																																																																																																
3662	21:57	16/09/07	1	0	0																																																																																																																																
3661	21:57	16/09/07	1	0	0																																																																																																																																
3660	21:56	16/09/07	0	0	0																																																																																																																																
3659	21:56	16/09/07	0	0	0																																																																																																																																
3658	21:56	16/09/07	0	0	0																																																																																																																																
外見	オブジェクトの外枠と背景色を選択します。[透明]にチェックマークを入れた場合、外枠と背景色を使用しないとなります。																																																																																																																																				
文字	表示するフォントとサイズを設定します。																																																																																																																																				
時間/日付	データのサンプリング時間や日付を表示するかを選択し、それに時間や日付の表示色とフォーマットを決めます。																																																																																																																																				
[日付]欄を[時間]欄の前に移動する	本項目にチェックマークを入れると、この 2 欄の表示順番が変わります。シーケンス番号																																																																																																																																				
データの番号	本項目にチェックマークを入れると、データの番号欄が表示されます。																																																																																																																																				
時間順で配列する	データは古い順で表示されます。最新のデータはボトム部に表示されます。																																																																																																																																				
逆時間順で配列する	データは新しい順で表示されます。最新のデータはトップ部に表示されます。																																																																																																																																				
履歴データコントロール (eMT、iE、XE、	HMI はサンプリングデータの履歴データを時間順で配列し、最新日付けのデータをレコード 0(一般的には今日セーブしたサンプリングデータのことを指す)、二番目新しいデータをレコード 1 のように順																																																																																																																																				

mTV シリーズ) 位付けます。[履歴コントロール]を用いて表示するデータを指定します。

**閲覧
(cMT シリーズ)** 履歴データオブジェクトで任意のデータでクリックすると、そのデータの内容は指定したアドレスに出力されます。

Note

- cMT シリーズを使用する場合、直接に各装置の cMT-Viewer 画面で履歴データ表示オブジェクトの右上隅にある歯車アイコンをクリックして、日付を指定しデータを表示させます。



- ▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

データ表示フォーマット



設定

記述

チャンネル

同時に最大 64 個チャンネルを表示することができます。本設定タブでは、現在選択している[資料サンプリング]オブジェクトが一回サンプリングしたデータの長さ、各チャンネルのデータフォーマット、本オブジェクトで表示するチャンネル番号を知ることができます。上図のように:[データサンプリング]オブジェクトが一回実行すると、4 データを読み取り(チャンネル 1~チャンネル 4)、各チャンネルのデータフォーマットは全部 16-bit Unsigned と示します。チャンネル 1 及びチャンネル 4 だけを表示するように選択したので、履歴データ表示オブジェクトが表示するデータリストは下図の通りです。

番号	時間	日付	ch.1	ch.3
19	15:48	02/03/16	0	0
18	15:48	02/03/16	4	5
17	15:48	02/03/16	4	5
16	15:48	02/03/16	4	5
15	15:48	02/03/16	4	5
14	15:48	02/03/16	4	3
13	15:48	02/03/16	4	3
12	15:48	02/03/16	1	3
11	15:48	02/03/16	1	3

履歴データ表示オブジェクトで[ストリング]フォーマットを表示する場合、下記が選択可能です：



- [Unicode]モードで表示します。
- データの上位バイトと下位バイトデータを逆さに変更してから表示します。

タイトルの設定

cMT シリーズ

eMT、iE、XE、mTV シリーズ



設定

記述

タイトルを使用する

タイトルを使用するかを選択します。

番号	時間	日付	ch.1	ch.4
1	19:56	18/04/17	#####	#####

カスタムファイル名を表示する

データサンプリングに[カスタムファイル管理]を使用した場合、[カスタムファイル名を表示する]を有効にすると、履歴表示オブジェクトの上側にカスタムのファイル名が表示されます。

背景

透明

[透明]にチェックマークを入れると、タイトル文字の背景に色つけません。

色

タイトル文字の背景に色付けます。

設定

タイトルの文字を設定します。

タイトルの文字ソースはラベルライブラリから取得してもいいです。ラベルライブラリを使用するには、前もってラベルライブラリを定義した後、履歴データ表示のラベルライブラリ欄でチェックマークを入れ、そしてラベルタグを選択すればいいです。

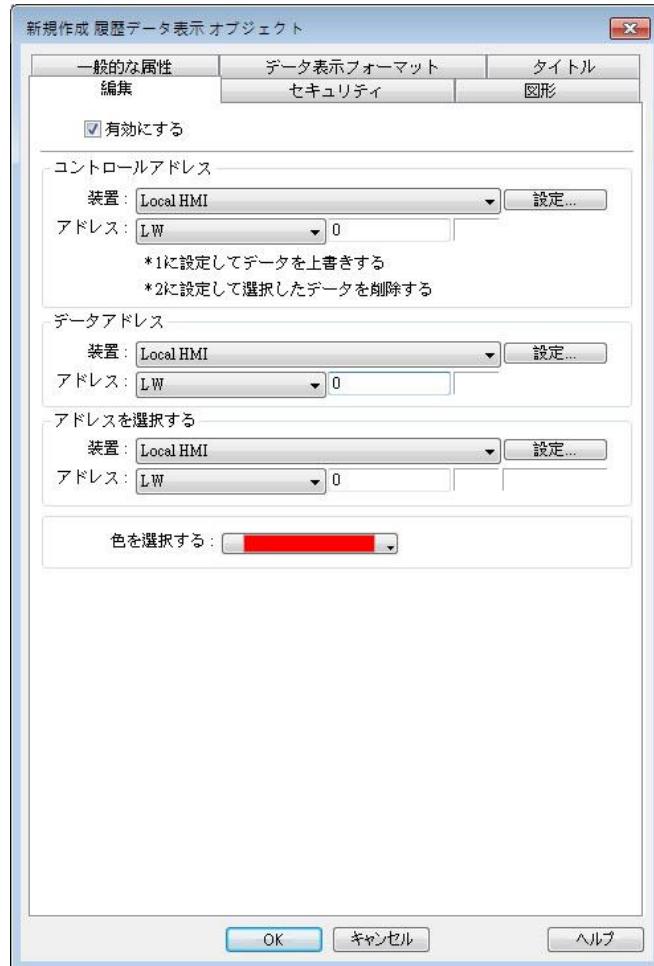
タイトル名	ラベルライブラリ	ラベルタグ	タイトル
▶ シーケンス番号	<input checked="" type="checkbox"/>	Label_1	No.
時間	<input type="checkbox"/>		時間
日付	<input type="checkbox"/>		日付
チャンネル1	<input type="checkbox"/>		ch.1
チャンネル2	<input type="checkbox"/>		ch.2
チャンネル3	<input type="checkbox"/>		ch.3

**Note**

- データサンプリングした後でデータフォーマットを変更したことがあつたら、オフラインモードを実行する際に、再度にデータフォーマットを変更してはいけません。変更する必要がある場合、C:\EasyBuilder\HMI_memory\datalog で古い履歴データ表示を記録したデータを削除してから、オフラインシミュレーションを行ってください。
PC でシミュレーションを実行した後、同一のプロジェクトにデータサンプリング内のデータフォーマットを変更して再度シミュレーションを実行したかつたら、システムが誤って旧データを読み取ることを避けるため、先に EasyBuilder インストールフォルダ内の HMI_memory / SD_card / usb1 フォルダ内の旧データサンプリング記録を削除してください。

編集設定

eMT、iE、XE、mTV、iP にサポートされている



設定	記述
制御アドレス	<p>数値を 1 に設定した場合、[データアドレス]内のデータをデータサンプリングに上書きすることができます。</p> <p>数値を 2 に設定した場合、選択したデータを削除することができます。</p>
データアドレス	<p>履歴データ表示オブジェクト内の特定の行を選択した場合、システムはその行から順序にデータを読み取り、この時にデータサンプリング内のデータを修正できます。</p> <p>注意: 本アドレスのデータフォーマットはデータサンプリングのデータフォーマットに一致する必要があります。</p>
選択アドレス	<p>現在履歴データ表示オブジェクトで選択された行数を指示します。本アドレス内の数値を変更すれば、選択する行数を変更できます。</p> <p>注意:</p> <ol style="list-style-type: none"> 選択アドレス内の数値は 0 の場合、行数は選択されません。そ

れに、[データアドレス]内の数値は前回の選択された行に維持します。

2. 選択アドレス内の数値が実際の行数より大きい場合、現在最大の行数を選択します。
-

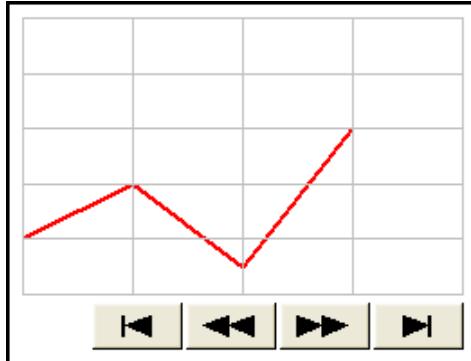
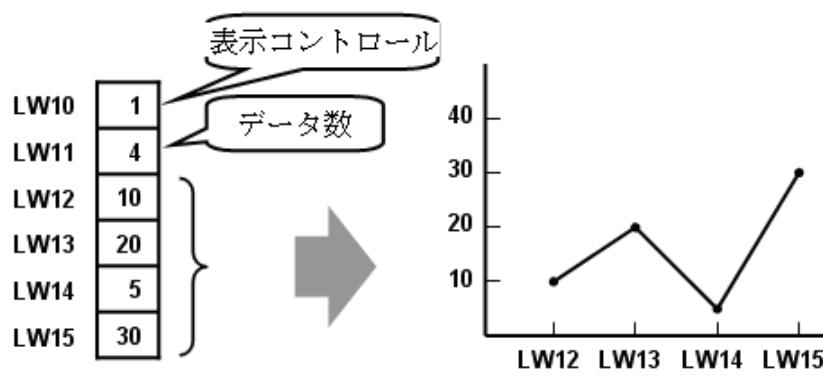
 **Note**

- データサンプリングの保存先を外部装置に指定した場合、外部装置を外すと、本機能が使用できません。外部装置を再度挿入してからシステムが履歴データを読み取った後、動作が継続されます。
- ▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

13.19. データグループ表示

13.19.1. 概要

一個のデータグループというのは、1セットの連続アドレス内のデータのことを指しています。X軸をアドレス、Y軸をデータとし、下図のようにデータグループ表示オブジェクトを使用し、データグループ LW-12~LW-15 内のデータを表示します。データグループ表示オブジェクトは同時に複数のデータグループの内容を表示することができます。例えば、同時に LW-12~LW-15 と RW-12~RW-15 との二つのデータグループを表示でき、ユーザーはこれを通じて各レジスタ内のデータを観察、比較することができます。



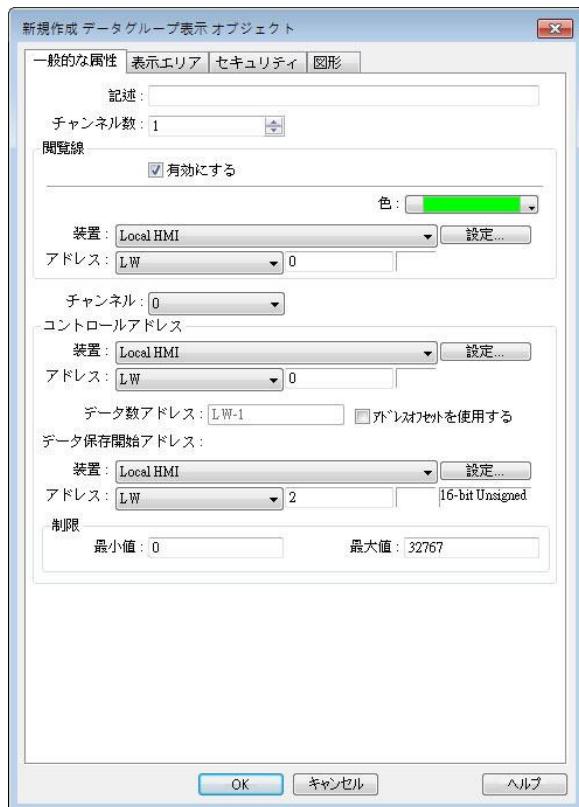
実際の実行結果

13.19.2. 設定



ツールバー上の[オブジェクト]»[オブジェクト]»[チャート]» [データグループ表示]ボタンを押せば、[データグループ表示]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[データグループ表示]オブジェクトが新規作成されます。

一般的な属性の設定



設定

記述

記述

ユーザーはこのオブジェクトに関連情報を記述できます。

チャンネル数

オブジェクトのチャンネル数量を設定します。一つのチャンネルは 1 セットのデータグループを代表しています。同時に最大 12 セットをサポートできます。

閲覧線

本項を使用すると、垂直閲覧線上のデータインデックスと数値を表示することができます。下記例 1 をご参照ください。

入力機能を有効にする(cMT シリーズ)

[閲覧アドレス]で数値を入力して垂直閲覧線を移動させることができます。

チャンネル

一個のチャンネルを選択し、コントロールアドレスの関連属性を設定します。

コントロール アドレス

データグループのコントロールアドレス及びデータソースを選択します。コントロールアドレスを LW-n に設定した場合、特定の数値を LW-n に入力することで図形の表示と削除をコントロールします。コマンドの実行を完了後、システムは改めてコントロールアドレスを 0 に設定します。

“0”を入力すると：動作しない(デフォルト)

“1”を入力すると：作画する

“2”を入力すると：消去する

“3”を入力すると：改めて作画する

データ数アドレス

コントロールアドレスを $LW-n$ に設定した場合、 $LW-n+1$ は各データグループの中に、どれぐらいのデータを保存できるのかを設定するアドレスです。最大 1024 個です。

データ保存開始アドレス

[アドレスオフセットを使用する]を使用する場合、[オフセット値保存アドレス]は $LW-n+2$ と定義します。

16-bit フォーマットを選択する場合、各開始アドレスの間隔は 1 となります。

例えば：開始アドレス+1、開始アドレス+2、などです。

32-bit フォーマットを選択する場合、各開始アドレスの間隔は 2 となります。

例えば：開始アドレス+2、開始アドレス+4、などです。

コントロールアドレスの各設定に関しては、下記例 2~5 をご参照ください。

限界

表示する図形の最大値及び最小値を設定します。

Note

- 重複作画の上限回数は 32÷チャンネル数です。

例 1

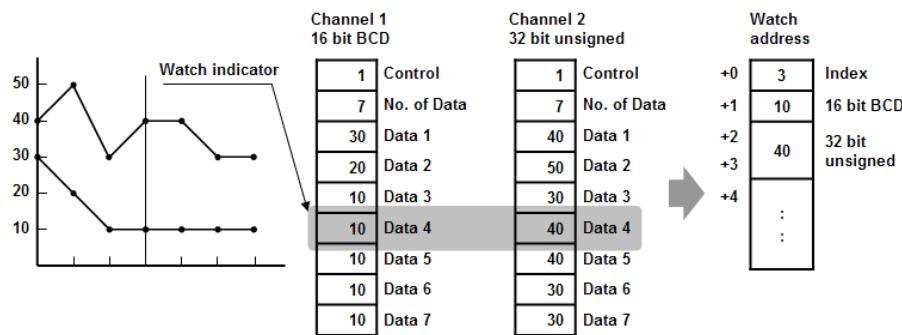
数値閲覧機能

[閲覧線]機能を使用すると、本オブジェクトに任意の点をクリックすると、一本の垂直な閲覧線が表示され、カーブと閲覧線が交差し合って生じた点が対応する数値が指定したレジスタに書き込まれます。

データフォーマット	データインデックス	チャンネル 1 数値	チャンネル 2 数値
16-bit	アドレス	アドレス+1	アドレス+2
32-bit	アドレス	アドレス+2	アドレス+4
64-bit (cMT のみ)	アドレス	アドレス+4	アドレス+8

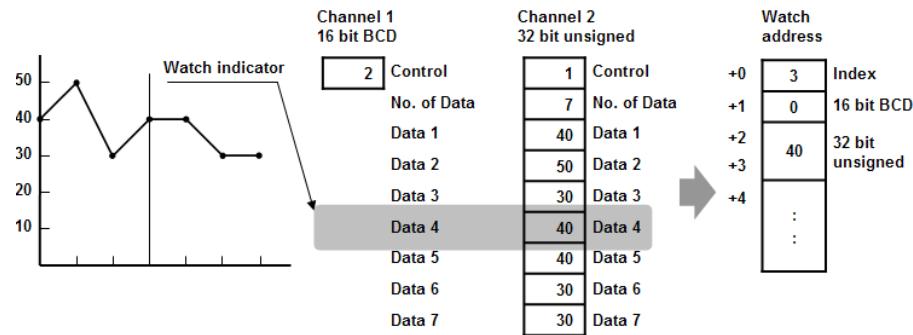
閲覧アドレスを $LW-n$ に設定した場合、 $LW-n$ に数値を書き込むと、その数値は呼び出したいインデックス番号のことを代表しています(0 から計算する)。下図をご参照ください：

オブジェクト

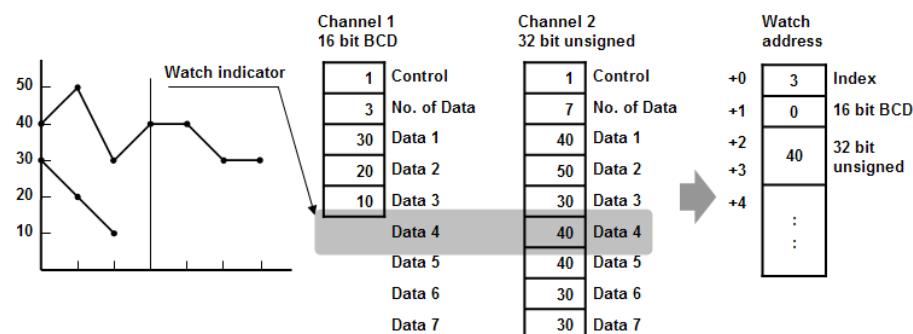


Note

- インデックス番号は16ビットの符号なし整数です。指定するレジスタは32ビットの場合、下位ビットの16ビットだけが作用し、上位ビットの16ビットは省略されます。
- 閲覧するチャンネルにデータがない場合、0で代替します。
EX:チャンネル2のみデータがあり、チャンネル1にデータがない場合、LW-n+1は0に表示します。



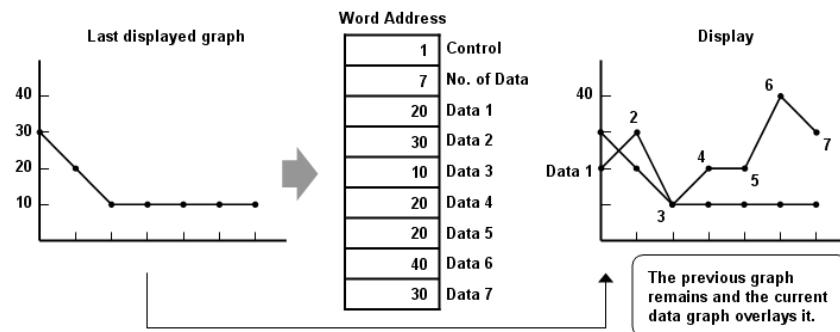
EX:チャンネル2のみデータがあり、チャンネル1にデータがない場合、LW-n+1は0に表示します。



例 2

データグループの内容を表示する方法

1. [データ数アドレス]に表示させたいデータ数を入力します。即ち“コントロールアドレス+1”です。
2. [データ保存開始アドレス]に順次にデータ内容を記入します。
3. [コントロールアドレス]に”1”を記入します。この時に、HMIは折れ線グラフで現在レジスタの内容を描きます(前の図形は保存されます)。
4. HMIが前項の動作を完了後、[コントロールアドレス]に”0”を書き込みます。



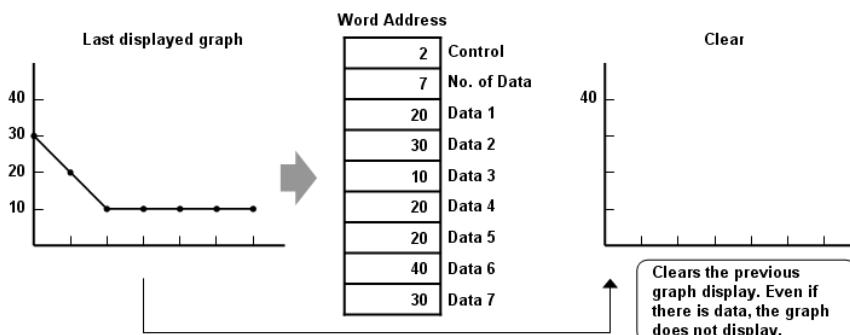
Note

- 上記動作 3 と 4 の間で、[コントロールアドレス]、[データ数アドレス]及び[データ保存開始アドレス]の内容を変更しないでください。さもないと、予期できない結果が生じる可能性があります。

例 3

すでに表示された図形を消去する方法

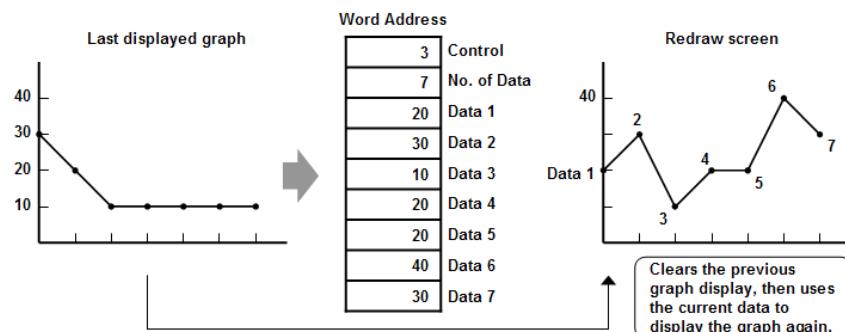
- [コントロールアドレス]に"2"を記入することで、前のグラフをクリアします。
- HMI が前項の動作を完了した後、[コントロールアドレス]に"0"を書き込みます。



例 4

すでに表示された図形を消去し、新しいデータの図形を表示する

- [データ数アドレス]で表示させたいデータ数量を入力し、即ち“コントロールアドレス+1”。
- [データ保存開始アドレス]に順次にデータ内容を記入します。
- [コントロールアドレス]に"3"を入力します。この時、HMI はまず前の折れグラフをクリアしてから、現在アドレス内の内容を描きます。
- HMI が前項の動作を完成したら、[コントロールアドレス]に"0"を書き込みます。



例 5

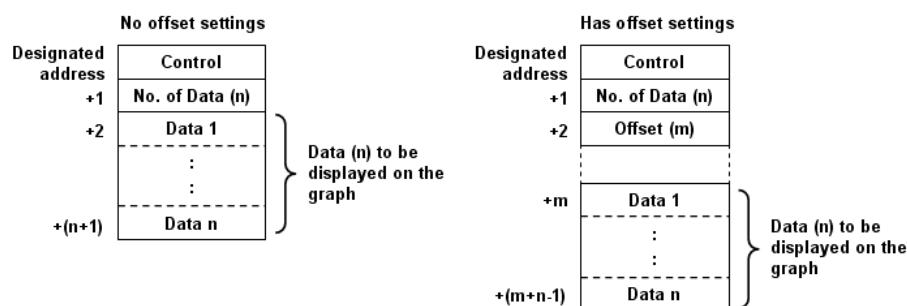
アドレスオフセットを使用する

本項を選択したら、各チャンネルの[コントロールアドレス]、[データ数アドレス]、[オフセット値保存アドレス]は連続したアドレスを使用します。

仮に3個のチャンネルがあり(チャンネル0からチャンネル2)、また[コントロールアドレス]はLW-0、LW-100とLW-200の場合、各チャンネルの[コントロールアドレス]、[データ数アドレス]、[オフセット値保存アドレス]は下記のようになります:(下表では3チャンネルを使用し、フォーマットは全部16-bit Unsignedでオフセット値保存アドレス内の数値をmにする場合)

項目	チャンネル0	チャンネル1	チャンネル2
コントロールアドレス	LW-0	LW-100	LW-200
データ数アドレス	LW-1	LW-101	LW-201
オフセット値保存アドレス	LW-2 (=m)	LW-102 (=m)	LW-202 (=m)
データ1	LW-0+m	LW-100+m	LW-200+m
データ2	LW-1+m	LW-101+m	LW-201+m
...

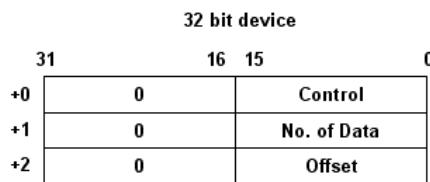
下図左側は[オフセットモード]を使用していない読み取り方式を示しています。右側はアドレスオフセットモードを使用する読み取り方式を示しています。



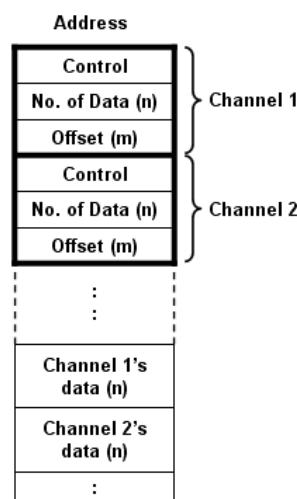
- [コントロールアドレス]を LW-n に設定する場合、[数値アドレス]及び[データ保存オフセット]は下記規則に基づいて設定します：

データフォーマット	16-bit	32-bit	64-bit (cMT のみ)
コントロールアドレス	LW-n	LW-n	LW-n
データ数アドレス	LW-n+1	LW-n+2	LW-n+4
オフセット値保存アドレス	LW-n+2	LW-n+4	LW-n+8

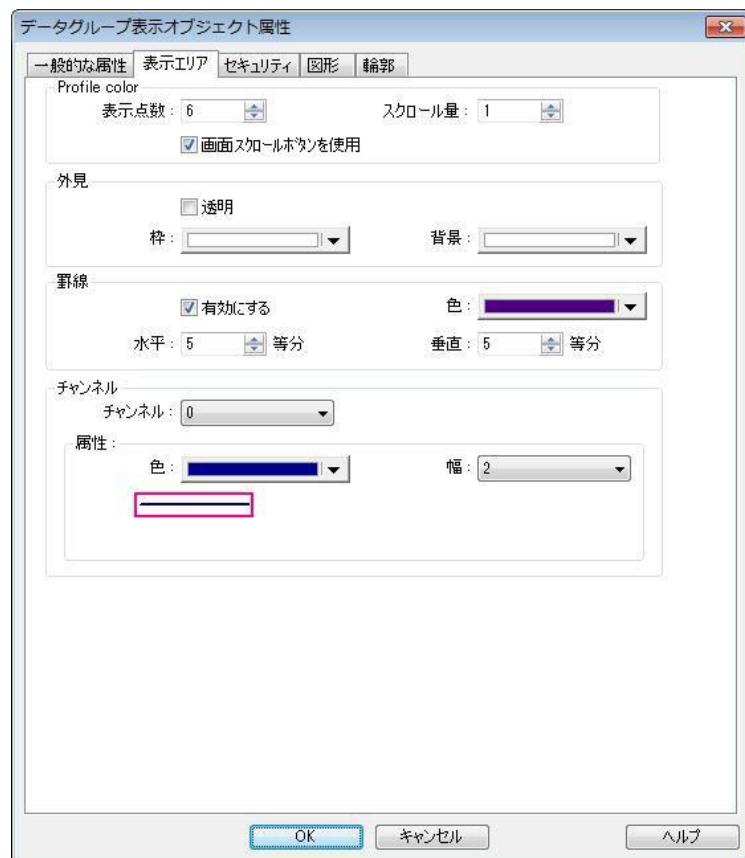
- [コントロールアドレス]は 32 ビットの場合、より下位の 16 ビットだけが作用します。より上位の 16 ビット内容を 0 に設定してください。



- システムは[コントロールアドレス]の数値が 0 ではない時に[数値番号アドレス]及び[オフセット値保存アドレス]の内容を読み取ります。
- 2 個以上の同じタイプのレジスタのチャンネルを使用した場合、[アドレスオフセットを使用する]を使用し、また連続したアドレスをコントロールアドレスにすることで読み取り時間を短縮できます。下図をご参照ください。16 ビットフォーマットを使用する場合、チャンネル 1 のコントロールアドレスを LW-n、チャンネル 2 のコントロールアドレスを LW-n+3 に設定し、このように類推します。



表示エリア設定



設定	記述
記述	<p>表示点数 1 ページで表示する最大データ数(点)を設定します。</p> <p>スクロール量 左右へスクロールする際に表示するデータ数を設定します。</p> <p>画面スクロールボタン [◀◀] [▶▶] を押すと、画面は前/後ろへ 1 点のデータを表示します。 [◀] [▶] を押すと、画面は最初/最後のデータを表示します。</p>
外見	オブジェクトの外枠の色及び背景の色を設定します。
透明	[透明]にチェックマークを入れると、オブジェクトは背景色を使用しなく、[色]という選択肢も現れません。
野線	[有効にする]を選択し、水平及び垂直の野線の数量を設定します。
チャンネル	各カーブの色、幅とスタイルを設定します。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.20.XY プロット

13.20.1. 概要

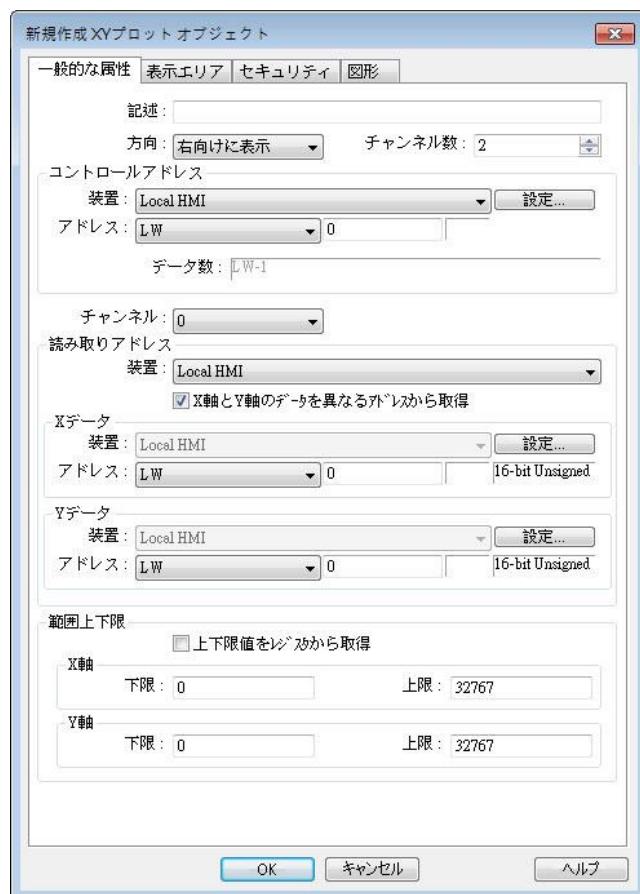
[XY プロット]はレジスタから読み取られた 2 次元座標の XY 変数を表示するオブジェクトです。同時に最大 16 セットの曲線を表示することができます。本機能に通じ、ユーザーは各レジスタ内のデータを観察、分析することができます。負数でも使用できます。

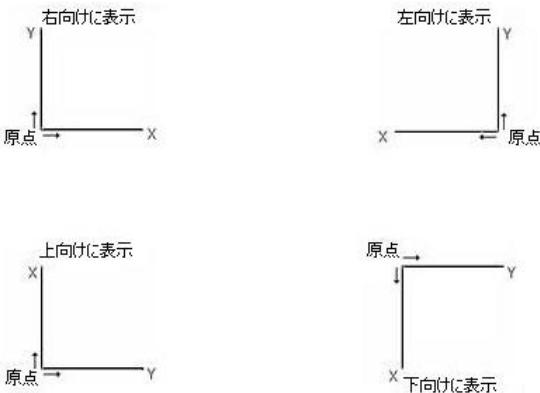
13.20.2. 設定



ツールバー上の[オブジェクト]»[チャート]»[XY プロット]ボタンを押すと、オブジェクト属性ダイアログボックスが現れます。

一般的な属性の設定



設定	記述	
方向	XY 軸の方向を[右向け]、[左向け]、[上向け]或いは[下向け]に表示するように選択できます。下図をご参照ください：	
		
チャンネル数	観察したいチャンネルの数量を設定します。	
コントロールアドレス	XY プロットの表示または消去をコントロールします。コントロールアドレスを LW-n に設定する場合、LW-n に異なる数値を書き込むことは異なるコマンドを代表しています。同時に、LW-n+1 で表示するデータの数量をコントロールします。HMI が指定した動作を完了した後、[コントロールアドレス]を 0 に設定します。	
コントロールアレ	数値	結果
LW-n	1	現在の図形を確保(作画された図形を確保)
	2	図形を消去
	3	全ての図形を消去し、改めて作画
LW-n	任意数	表示するデータ数量
データ数量アドレス		
	データ(点)を保存するためのアドレスです。一個チャンネルは最大 1023 点を保存できます。	
チャンネル	一個のチャンネルを指定し、読み取りの関連属性を設定します。	
読み取りアドレス	装置 データ読み取りのソースである装置を選択します。 レジスタのデータをアクセスする際に、[X 軸と Y 軸のデータを異なるアドレスから取得]や[上下限値をレジスタから取得]を使用するかを考える必要があります。下記例 1 をご参照ください。	

上下限

- チェックマークを入れていない場合：
上限/下限は定数です。上下限は X、Y 軸のスケール百分比を計算することに用いられます。下記例 2 をご参照ください。
 - チェックマークを入れた場合：
ユーザーは上下限を変更することによって拡大/縮小効果が得られます。下記例 3 をご参照ください。
-

例 1

レジスタのデータをアクセスする際に、[X 軸と Y 軸のデータを異なるアドレスから取得][上下限値をレジスタから取得]を使用するかを考える必要があります。以下は実例で各状況を説明します。(全部 16-bit レジスタを使用する場合) :

- 仮に[X 軸と Y 軸のデータを異なるアドレスから取得]を選択しなく、[アドレス]を LW-n に設定する場合 :

	[上下限値をレジスタから取得] を使用する	[上下限値をレジスタから取得] を使用しない	
X データ	Y データ	X データ	Y データ
下限	LW-n	LW-n+2	定数
上限	LW-n+1	LW-n+3	定数
一番目のデータ	LW-n+4	LW-n+5	LW-n+0
			LW-n+1
二番目のデータ	LW-n+6	LW-n+7	LW-n+2
			LW-n+3
三番目のデータ	LW-n+8	LW-n+9	LW-n+4
			LW-n+5
四番目のデータ	LW-n+10	LW-n+11	LW-n+6
			LW-n+7

- 仮に[X 軸と Y 軸のデータを異なるアドレスから取得]を選択し、[X データ]は LW-m、[Y データ]は LW-n の場合 :

	[上下限値をレジスタから取得] を使用する	[上下限値をレジスタから取得] を使用しない	
X データ	Y データ	X データ	Y データ
下限	LW-m+0	LW-n+0	定数
上限	LW-m+1	LW-n+1	定数
一番目のデータ	LW-m+2	LW-n+2	LW-m+0
			LW-n+0
二番目のデータ	LW-m+3	LW-n+3	LW-m+1
			LW-n+1
三番目のデータ	LW-m+4	LW-n+4	LW-m+2
			LW-n+2
四番目のデータ	LW-m+5	LW-n+5	LW-m+3
			LW-n+3

例 2

[上下限値をレジスタから取得]を選択していない場合、上限/下限は定数です。上下限は X、Y 軸のスケール百分比を計算することに用いられます：

$$\text{スケール百分比(%)} = \frac{\text{レジスタ内データ-下限}}{\text{上限 - 下限}}$$

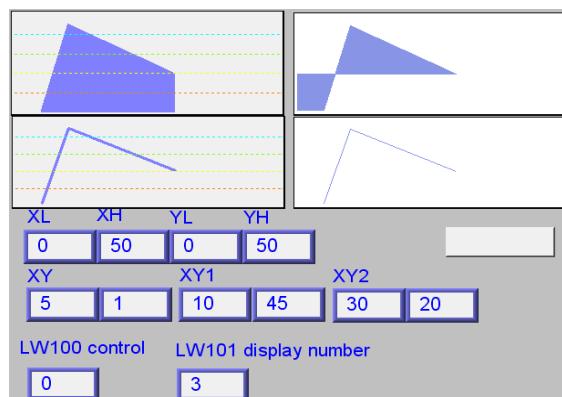
仮にレジスタは LW-n で、[X 軸と Y 軸のデータを異なるアドレスから取得]を選択していない場合、上限/下限のデータソースは下記方式で設定されます：

データフォーム マット	16-bit	32-bit	64-bit (cMT のみ)
X 軸下限	LW-n	LW-n	LW-n
X 軸上限	LW-n+1	LW-n+2	LW-n+4
Y 軸下限	LW-n+2	LW-n+4	LW-n+8
Y 軸上限	LW-n+3	LW-n+6	LW-n+12

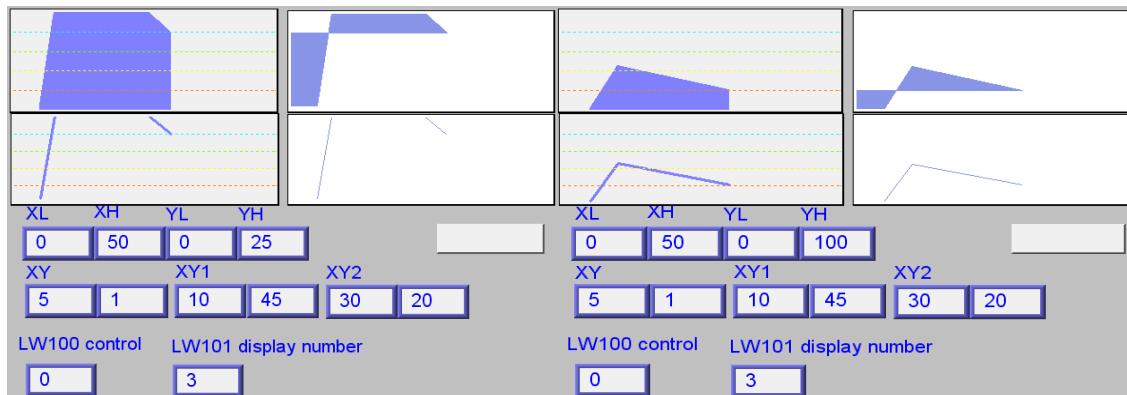
例 3

[上下限値をレジスタから取得]を選択する場合、ユーザーは上下限を変更することによって拡大/縮小効果が得られます。

下記例のように、XL=X 下限、XH=X 上限、YL=Y 下限、YH=Y 上限、XY、XY1、XY2 は三個の XY データです。この時、Y 軸の上限を変更すれば、拡大/縮小効果が見られます。下記をご参照ください：



原図



Y 軸上限を 25 に変更(拡大効果)

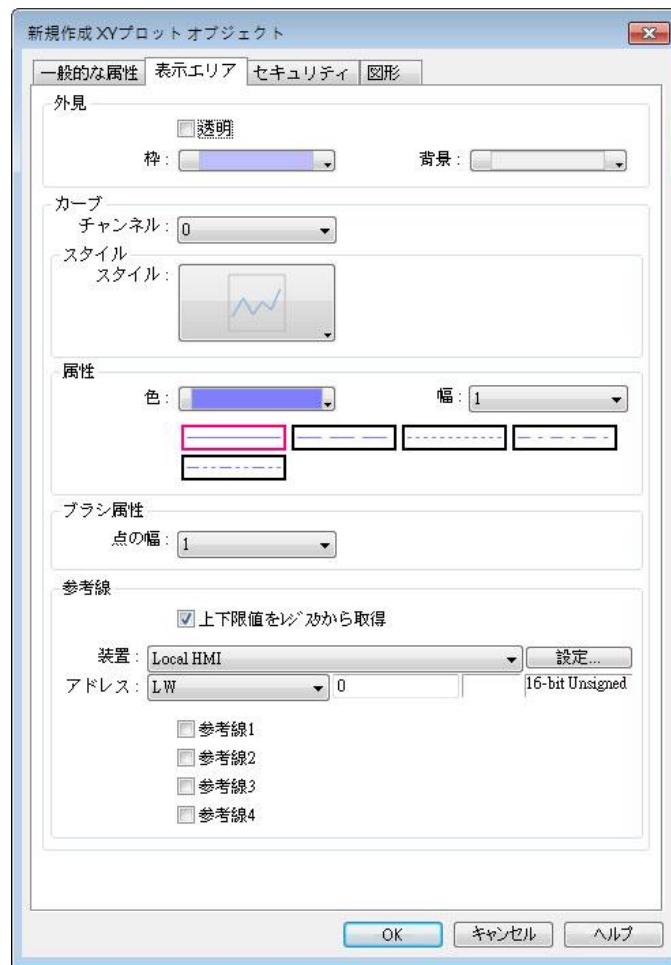
Y 軸上限を 100 に変更(縮小効果)

👉 詳細は、《13.17 トレンド図》をご参照ください。

Note

- X と Y データは異なるフォーマットを使用することができます。例えば X データは 16-bit unsigned を使用し、Y データは 32-bit signed を使用してもいいです。この場合、特別にアドレスの設定にご留意ください。
- 装置は Tag PLC の場合、例えば AB tag PLC の場合、X と Y は必ず同じアドレスフォーマットを使用すること。異なるフォーマットを選択したら、ウォーニングが出る可能性があります。

表示エリア設定



設定

記述

外見

[透明]を選択すると、背景は透明になり、選択していない場合は選んだ色で外枠及び背景を表現します。

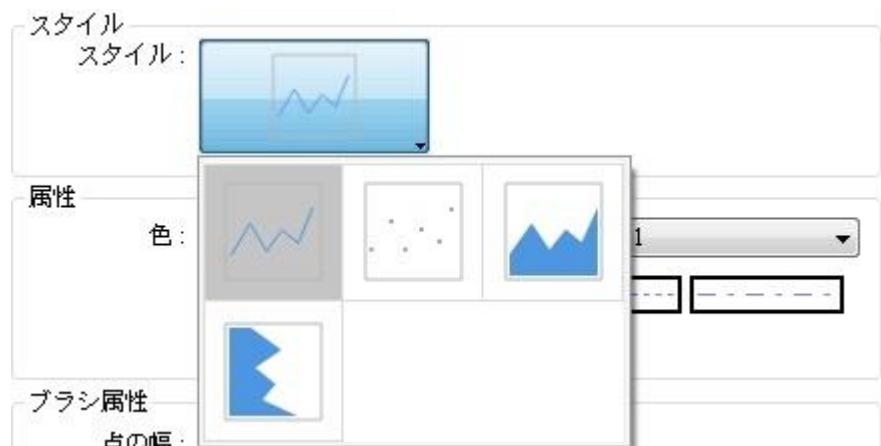
カーブ

ここでチャンネルが表示する属性を設定します。

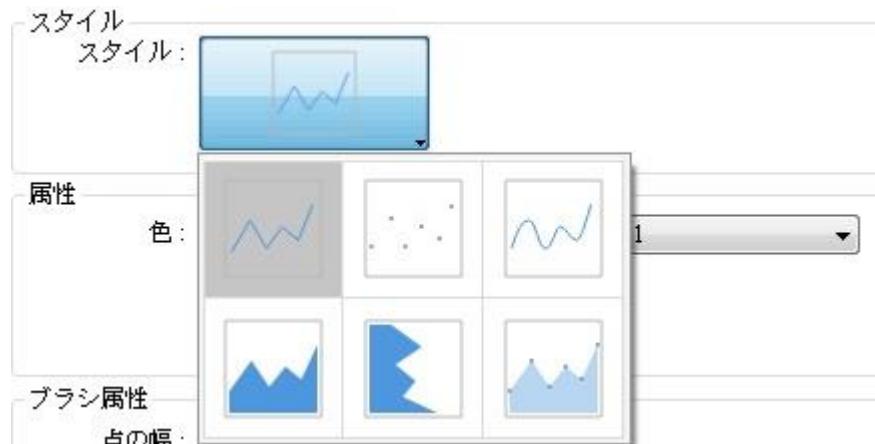
スタイル

モニターでの表示方式を設定します。

非 cMT シリーズ機種は 4 種類のスタイルをサポートします：



cMT シリーズは 6 種類のスタイルをサポートします：



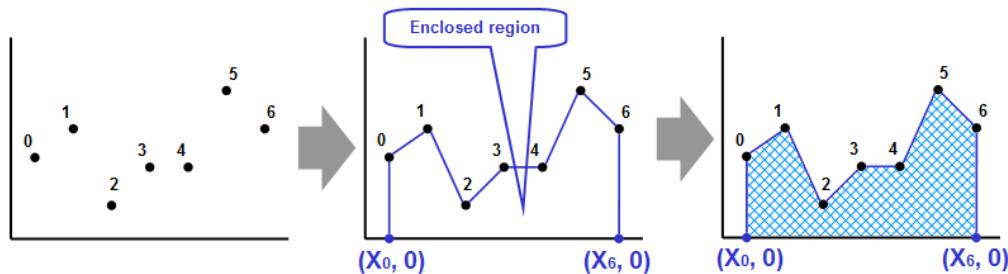
下記例 4 をご参照ください。

属性	線の[色]と[幅]を設定します。
ブラシ属性	点の幅を設定します。
参考線	最大四本の参考線を曲線図に描くことができます。ユーザーは自分で線の色彩及び参考する数値を選択し、設定した数値に基づいて表示します。[上下限値をレジスタから取得]を選択した場合、参考線の読み取りアドレスを設定する必要があります。

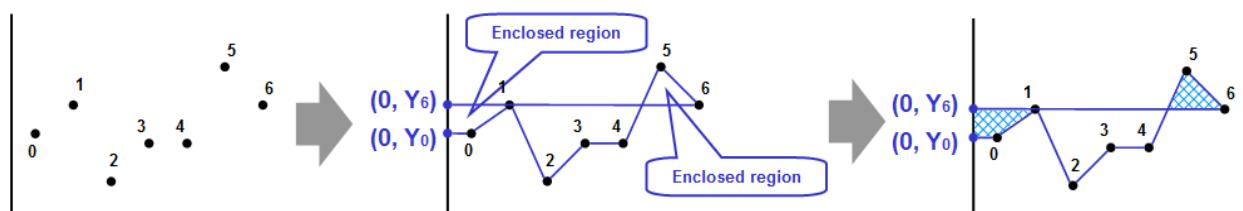
例 4

下図の曲線は P0~P6 との 7 つの点で構成されます。システムが[X 軸に投影]に作画する仕方は下記手順の通りになります：

1. 自動的に二つ投影する点を算出します：X 軸-(X₀, 0) と(X₆, 0)
2. 点が現れる順によって、全ての点をリンクします：(X₀, 0)、P0...P6、(X₆, 0) 最後に一番目の点にリンクします(X₀, 0)。
3. 囲まれた範囲を満たします。結果は以下のとおりです：



同じように、[Y 軸に投影]は以下のとおりです：



 Note

- XY Plot を最大 32 回重複で作画することができます。計算方式は以下のとおりです：
1 個のチャンネルは 32 回重複で作画でき、2 個のチャンネルがある場合、16 回のみ作画で
きます。 $32 \div \text{チャンネル数}$ で、重複で作画できる最大回数が得られます。

13.21. アラームバーとアラーム表示

13.21.1. 概要

[アラームバー]と[アラーム表示]オブジェクトは[イベントログ]で定義され、それにシステム現在の状態でトリガーできるイベントを表示することができます。これらのイベントはアラームとも言われます。[アラームバー]と[アラーム表示]オブジェクトはイベントがトリガーされた時間順で順次にこれらのアラームを表示します。その中、[アラームバー]オブジェクトは1行のスクロール文字で警告内容を表示します。[アラーム表示]オブジェクトは同時に複数行の警告内容を表示することができます。下図ではオブジェクトが警告を表示する形式を示しています。

👉 イベントログの関する詳細情報については、《7 イベントログ》をご参照ください。

I (When LW 1 >= 10) 13:21:06 Event 0 (when LW0)

[アラームバー]オブジェクトの表示形式です。1行で複数のイベントを表示します。

13/12/06	13:21:38	Event 2 (when LB10 = ON)
13/12/06	13:21:38	Event 3 (when LB11 = ON)
13/12/06	13:21:38	Event 0 (when LW0 == 100)
13/12/06	13:21:38	Event 1 (When LW 1 >= 10)

[アラーム表示]オブジェクトの表示形式です。複数行を表示することができます。

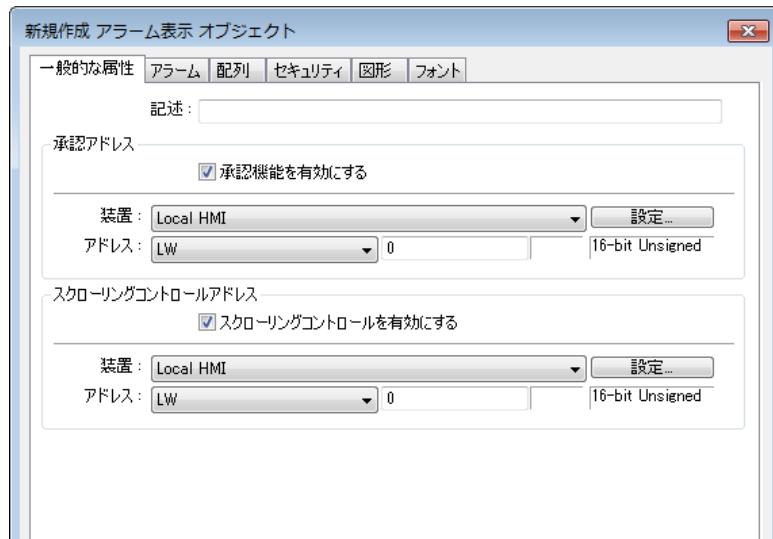
13.21.2. 設定



ツールバー上の[オブジェクト]»[データ/履歴]タブをクリックし、[アラームバー]ボタンを押せば、オブジェクトの属性ダイアログボックスがオープンされます。同じようにツールバー上の[オブジェクト]»[データ/履歴]タブをクリックし、[アラーム表示]ボタンを押せば、オブジェクトの属性ダイアログボックスがオープンされます。正確に各属性を設定完了してOKボタンを押すと、新規オブジェクトが新規作成されます。

一般的な属性の設定

[アラームバー]と[アラーム表示]が異なるところは、[アラーム表示]では[承認アドレス]及び[スクローリングコントロールアドレス]を設定することができます。下図をご参照ください。

**設定****記述****承認アドレス**

イベントが承認された時に、[イベントログ]»[メッセージ]タブで設定した[イベントが承認された時にイベント/アラーム表示オブジェクトに書き込む]の数値は[イベント表示オブジェクト]の[承認アドレス]に書き込まれます。詳細情報は《7 イベントログ》をご参照ください。

イベントが承認された時、イベント/アラーム表示オブジェクトに書き込む
承認値 : 11

スクローリングコントロールアドレス

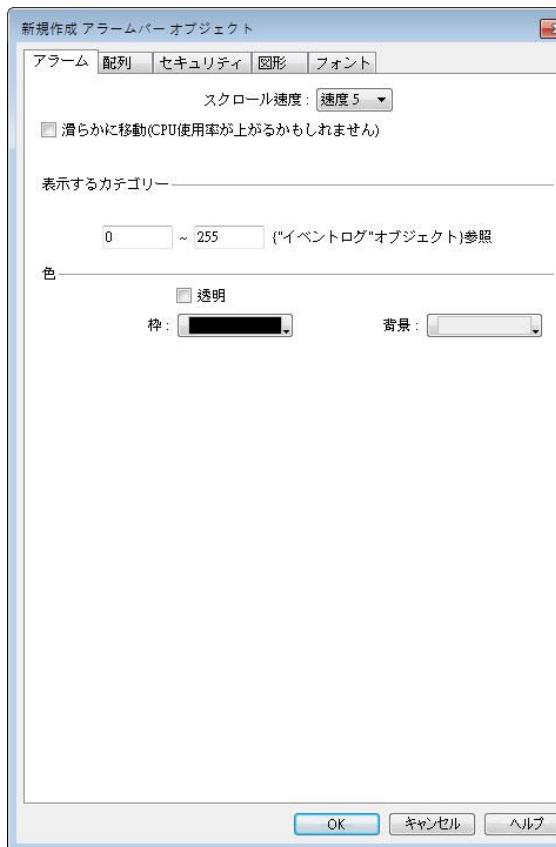
スクローリングコントロールアドレスを使用すると、オブジェクトはコントロールアドレスの数値を基づいて、下向けに指定した行数をスクロールします。数値は 0 から起算します。注意：本機能を有効にした場合、オブジェクトの上下及び左右のスクローリングバーが無効にされ、完全に内容を表示できるようにオブジェクトの幅を確認してください。

 **Note**

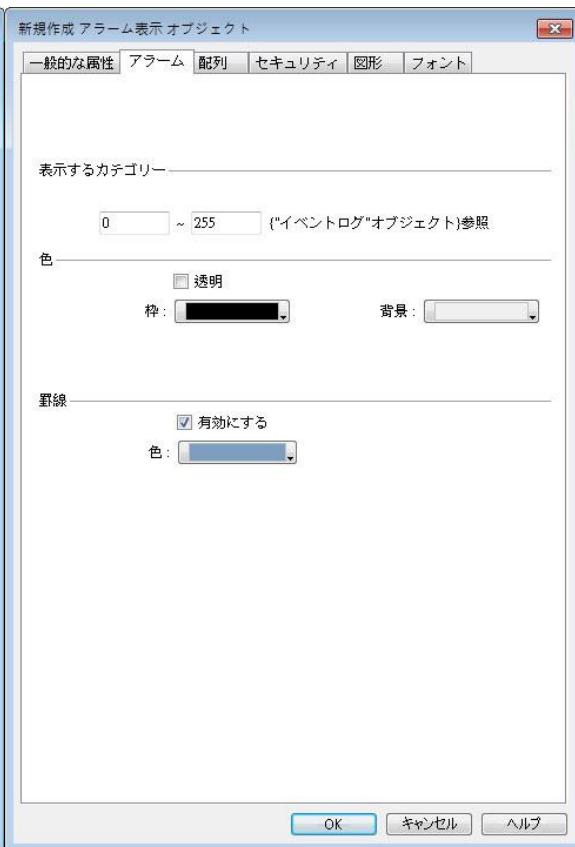
- cMT シリーズを使用する場合、指でモニタをタッチしたまま場所を移動しないと、当イベントに肯定応答が返ってきたと視されます。指でタッチしてスライドすると、スクロールしていると視されます。

アラームの設定

アラームバー



アラーム表示



設定

記述

表示するカテゴリー範囲

トリガーされたイベントの[カテゴリー]はここで設定された範囲に含まれた場合のみ表示されます(イベントのカテゴリーは[イベントログ]で設定する)。例えば、[アラームバー]オブジェクトの[表示するカテゴリー範囲]が 2~4 に設定された場合、[カテゴリー]は 2、3、4 のイベントのみ、[アラームバー]オブジェクトに表示されます。詳細な説明は《7 イベントログ》をご参照ください。

cMT シリーズ

動的カテゴリー範囲を選択できます。オブジェクトはレジスタ内の数値に基づいて動的に表示するイベントカテゴリーをフィルターします。

表示するカテゴリー

事前定義済み 動的カテゴリー範囲

装置: Local HMI
アドレス: LW 0 16-bit Unsigned

範囲開始: LW-0
範囲終了: LW-1

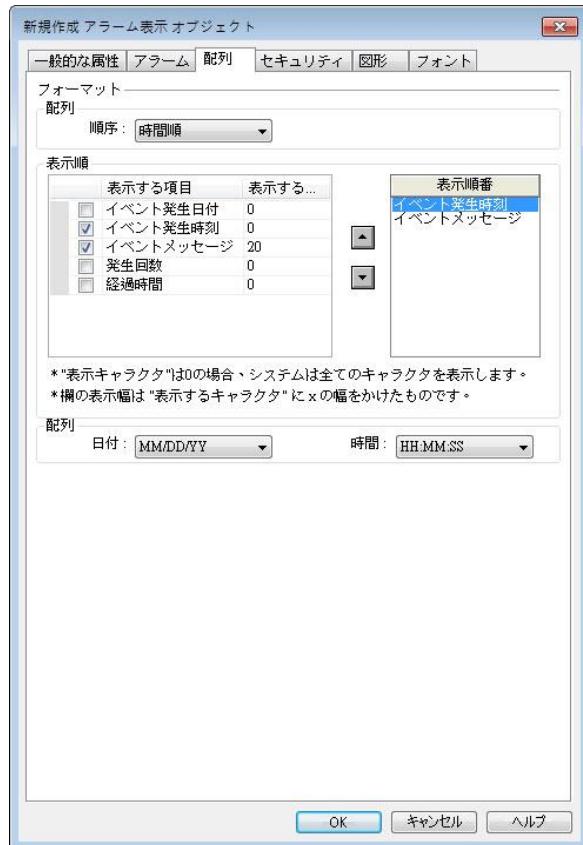
移動速度	[アラームバー]だけが設定できます。ここで[アラームバー]オブジェクトで表示する文字の移動速度を設定します。
滑らかに移動	本機能を有効にすると、アラームメッセージの文字がより平滑に移動するようになりますが、CPUの負荷を上げる可能性があります。

配列の設定

アラームバー



アラーム表示



[アラーム表示]の表示は[アラームバー]に比べると、[発生回数]及び[経過時間]があり、その他は同じです。下記をご参照ください：

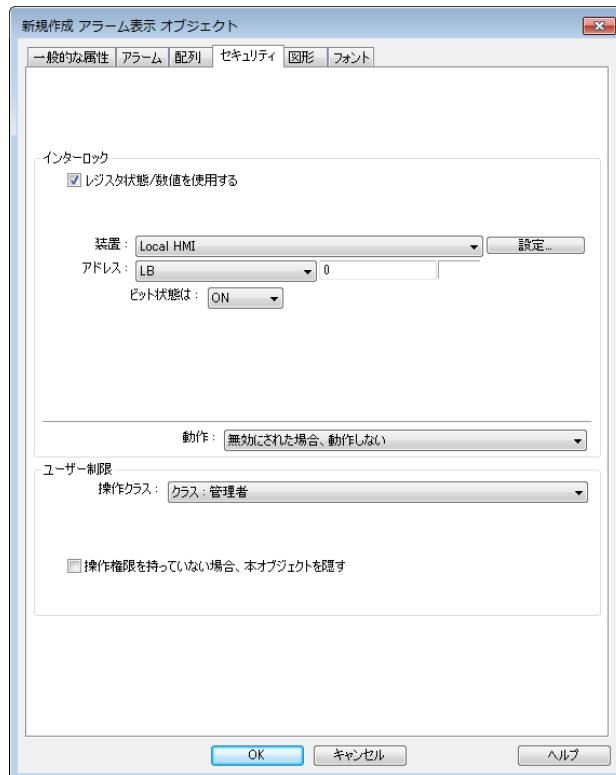
設定	記述
フォーマット	時間順で配列する より遅く発生したアラームは後ろ(または下)に配列されます。 逆時間順で配列する より遅く発生したアラームは前(または上)に配列されます。
	順番を表示 表示してほしい情報及び表示の順番を設定します。
日付	イベント発生日付の表示フォーマットを選択します。下記4モードがあります。
	MM/DD/YY、DD/MM/YY、DD.MM.YY、YY/MM/DD

時間

イベント発生時間の表示フォーマットを選択します。下記 4 モード
があります。

HH:MM:SS、HH:MM、DD:HH:MM、HH

セキュリティ設定



設定	記述
インターロック機能を使用する	<p>本項にチェックマークを入れると、このオブジェクトを操作できるかどうかのは、一個の指定したビットアドレスの[状態]によります。上図に示されたように、LB-0 が ON の場合のみ、本オブジェクトを操作できます。</p> <p>無効の場合、オブジェクトを隠す</p> <p>指定したビットアドレスが OFF の場合、オブジェクトは隠されます。</p>
ユーザー制限	<p>操作クラス</p> <p>“無”はどのユーザーでも操作できることを示しています。“管理者”は admin アカウントのみ操作できることを示しています。</p> <p>操作権限を持っていない場合、本オブジェクトを隠す</p> <p>ユーザーの操作権限が本オブジェクトのセキュリティクラスに満たしていない場合、オブジェクトが隠されます。</p> <p>アラーム表示オブジェクトを使用する場合、本項にチェックマークを入れなかったら、ユーザーは依然にアラーム表示オブジェクトが見られます。但し、本オブジェクトを変更、またはトリガーすることができません。</p> <p>アラームバーオブジェクトを使用する場合、本項のチェックマークを取り消すことができません。</p>

フォント設定

オブジェクト文字のサイズと斜体効果を選択します。



[アラームバー]と[アラーム表示]のイベントが表示する情報内容、フォントと色は[イベントログ]での設定を基づいて設定されます。下図をご参考ください：



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

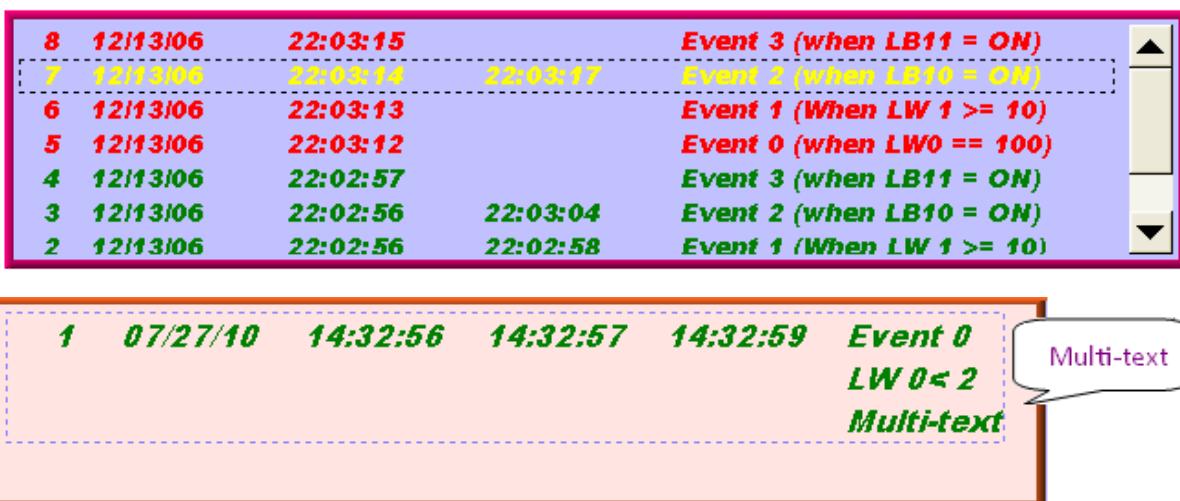
13.22. イベント表示

13.22.1. 概要

[イベント表示]は[イベントログ]に定義され、またトリガー条件を満たしたことがあるイベントを表示するオブジェクトです。

[イベント表示]オブジェクトはトリガーされた時間順で、順次にイベントを表示します。

[イベント表示]オブジェクトはイベントの発生日付、発生時刻、承認時刻、正常に戻った時刻、イベントメッセージ内容、発生回数、及び経過時間を表示することができます。メッセージ内容は複数行の形で表示することが可能です。



The screenshot shows two windows. The top window displays a table of events with columns for ID, Date, Start Time, End Time, and Description. The bottom window shows a multi-line text message with a speech bubble icon.

	8 12/13/06	22:03:15	Event 3 (when LB11 = ON)		
7	12/13/06	22:03:14	22:03:17	Event 2 (when LB10 = ON)	
6	12/13/06	22:03:13		Event 1 (When LW 1 >= 10)	
5	12/13/06	22:03:12		Event 0 (when LW0 == 100)	
4	12/13/06	22:02:57		Event 3 (when LB11 = ON)	
3	12/13/06	22:02:56	22:03:04	Event 2 (when LB10 = ON)	
2	12/13/06	22:02:56	22:02:58	Event 1 (When LW 1 >= 10)	

1 07/27/10 14:32:56 14:32:57 14:32:59 Event 0 LW 0< 2 Multi-text
--

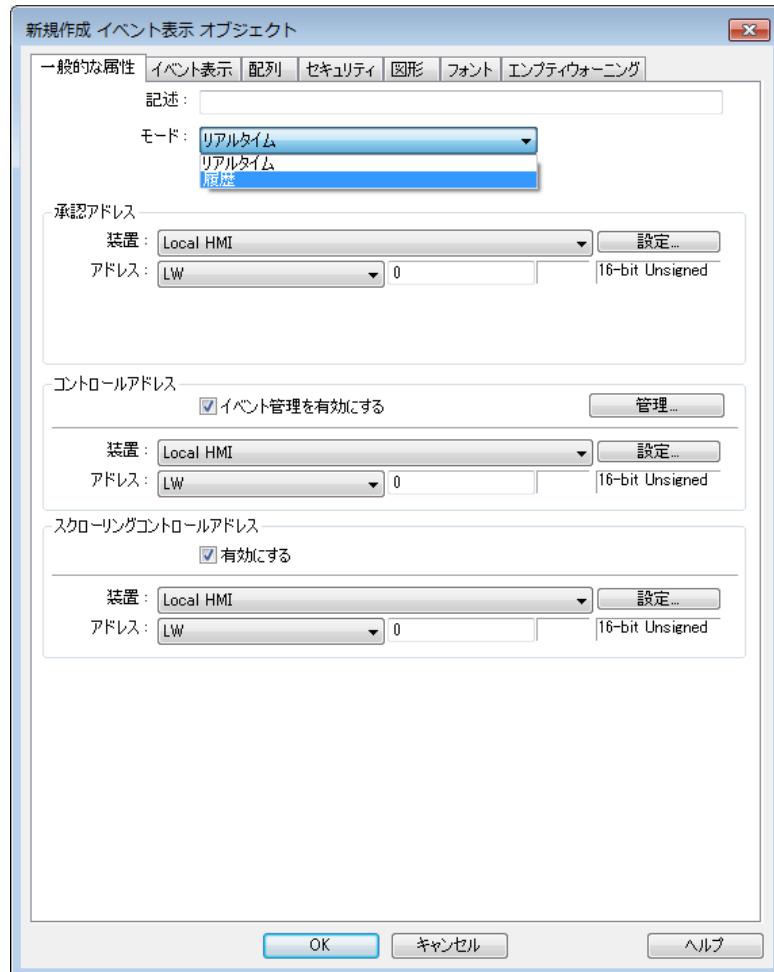
13.22.2. 設定



ツールバー上の[オブジェクト] » [データ/履歴]タブをクリックし、[イベント表示]ボタンを押せば、[イベント表示]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[イベント表示]オブジェクトが新規作成されます。

一般的な属性の設定

13.22.2.1. eMT、iE、XE、mTV シリーズ機種

**設定****記述****モード**

イベントソースのモードを選択します。[リアルタイム]或いは[履歴]モードを選択することができます。

- リアルタイム

HMI を起動して以来、全てのトリガーされたイベントを表示します。

- 履歴

システムはメモリー内のイベントデータを読み取って、オブジェクトに表示します。閲覧する内容は当日の履歴データの場合、毎秒ごとに閲覧する内容を自動的に更新します。

承認アドレス

[リアルタイム]モードを選択した場合、本アドレスを設定することができます。

イベントが承認された時に、[イベントログ] » [メッセージ]タブで設定した[イベントが承認された時にアラーム表示/イベント表示オブジェクトに書き込む]での数値は[イベント表示オブジェクト]の[承認アドレス]に書き込まれます。詳細情報は《7 イベントログ》をご参照ください。

イベントが承認された時、イベント/アラーム表示オブジェクトに書き込む

承認値 :

履歴データコントロール

[履歴]モードを選択した際に、本アドレスを設定することができます。

- [複数の履歴ファイルの読み取りを有効にする]を選択していない場合

履歴モードでは一日の履歴データだけが表示できます。履歴データコントロールアドレスは LW-n の場合、LW-n に特定した数値を入力すれば、対応する履歴データが表示されます。

システムはインデックスを通して履歴データ.evt ファイルを選択します：

0 を入力：最新の履歴データを表示する

1 を入力：二番目新しい履歴データを表示する

2 を入力：三番目新しい履歴データを表示する

このように類推します。

仮に履歴データコントロールアドレスは LW-100 で、現在は下記 4 レコードの履歴データがある場合：EL_20100720.evt、

EL_20100723.evt、EL_20100727.evt、EL_20100803.evt。

数値を LW-100 に入力した際に、表示される履歴データは下表の通りです：

LW-100 数値	対応する履歴データ
0	EL_20100803.evt
1	EL_20100727.evt
2	EL_20100723.evt
3	EL_20100720.evt

- [複数の履歴データを読み取る]を選択する場合

本項にチェックマークを入れると、一つのイベント表示オブジェクトで数日間の履歴データを表示することができます。本機能を使用すると、2 個の連続アドレスが使われます。仮に[履歴データコントロール]アドレスが LW-n の場合、LW-n と LW-n+1 はそれぞれ、表示させたいデータのインデックスと、データ範囲をコントロールします。[日数]と[最新の履歴データインデックス]を選択することによって、コントロールアドレスに対する定義の違いをご留意ください。

[日数]を選択する：

履歴データの表示範囲は[LW-n]内の数値が代表する日付から起算し、前へ LW-n+1 日を推算します。仮に現在の日付は 2010/6/10、LW-n に入力された数値は“1”、LW-n+1 に入力された数値は“3”的場合、表示する履歴データの範囲は前日 (LW-n の値)20100609 から、前へ三

日間を推算するので(20100609 を含めて三日間)、20100607~20100609範囲内のデータを表示するはずです。しかし、履歴データベースでは 20100607 のデータが存在しないため、表示する履歴データは 20100609 と 20100608 になります。

 EL_20100604	No.4	1 KB	EasyBuilder Proevt
 EL_20100605	No.3	1 KB	EasyBuilder Proevt
 EL_20100608	No.2	1 KB	EasyBuilder Proevt
 EL_20100609	No.1	1 KB	EasyBuilder Proevt
 EL_20100610	No.0	1 KB	EasyBuilder Proevt

[最新の履歴データインデックス]を選択する：

[LW-n] 内の数値は記録のインデックス開始値を示し、 [LW-n+1] 内の数値は記録のインデックス終了値を示しています。仮に LW-n に入力された数値は“1”、 LW-n+1 に入力された数値は“3”的場合、インデックス No.1~No.3 の履歴データのことを指し、即ち、表示される履歴データは下図の No.1、 No.2、 No.3 になります。

 EL_20100604	No.4	1 KB	EasyBuilder Proevt
 EL_20100605	No.3	1 KB	EasyBuilder Proevt
 EL_20100608	No.2	1 KB	EasyBuilder Proevt
 EL_20100609	No.1	1 KB	EasyBuilder Proevt
 EL_20100610	No.0	1 KB	EasyBuilder Proevt

注意: [LW-n]内の数値は[LW-n + 1]内の数値より小さなければなりません。それでこそ、合法な検索範囲になります。

システムは最大 4MB の履歴データを表示でき、超えた部分は省略されます。

以下はデータが大きすぎた場合の例です。

5 個の履歴データがあり、一個 0.5MB の場合→最大:8x0.5MB を表示できる

5 個の履歴データがあり、一個 1MB の場合→最大:4x1MB を表示できる

5 個の履歴データがあり、一個 1.5MB の場合→最大:2x1.5MB+1x1MB(局部)を表示できる

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

コントロール

イベント管理を使用する

アドレス

使用したら、特定した数値をレジスタ LW-n 及び LW-n+1 に書き込むと、「イベント表示」オブジェクトに異なるコマンドを下ることがで

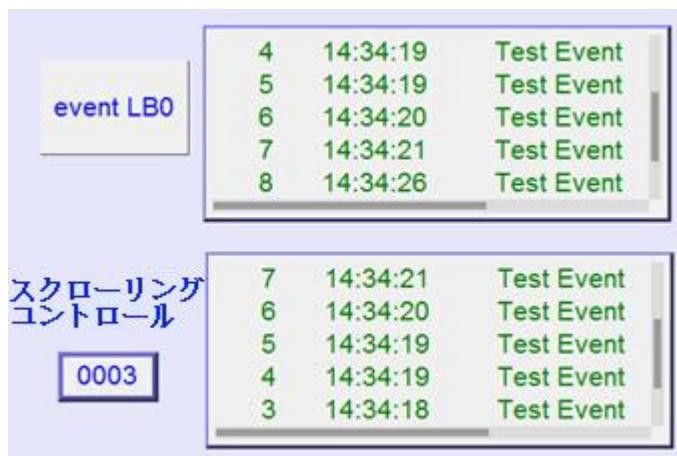
きます。コマンド条件及び内容は下表の通りです：

アドレ ス	数値	コマンド内容
LW-n	0	全てのイベントを表示する
	1	「承認された」イベントを隠す
	2	「復元された」イベントを隠す
	3	「承認された」或いは「復元された」イベントを隠す
	4	「承認された」と「復元された」イベントを隠す
	LW-n+1	選択したイベントを削除する

スクローリン
グコントロー
ルアドレス

オブジェクトはコントロールアドレスの数値を基づいて、下へ指定した行数をスクロールします。数値は0から起算します。

例：下図のように、オブジェクトでは合計10イベントがあります。スクローリングコントロールアドレスを3に設定した場合、上図では時間順で配列したので、一番上のイベントは番号4のイベントです。下図では逆時間で配列したので、一番上のイベントは番号7のイベントです。



スクローリングコントロール機能を使用した際に、スクロールバーはコントロール機能を失い、現在閲覧している内容の相対位置を表示することだけができます。コントロールアドレス内の数値がオブジェクトでの最大行数を超えた場合、一番後ろの内容を表示します。注意：本機能を有効にした場合、オブジェクトの上下及び左右のスクローリングバーが無効にされ、完全に内容を表示できるようにオブジェクトの幅を確認してください。

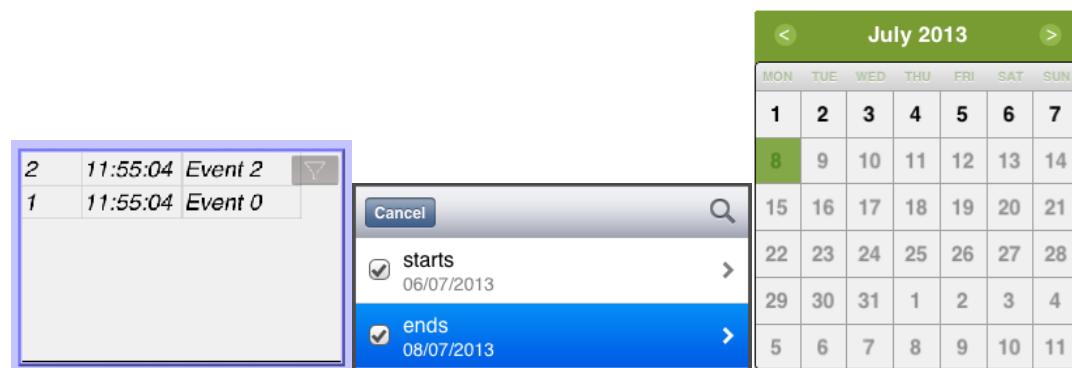
▶このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

13.22.2.2. cMT シリーズ機種



cMT シリーズ機種のイベント表示オブジェクトは全ての発生したイベントを表示し、それに即時に更新します。

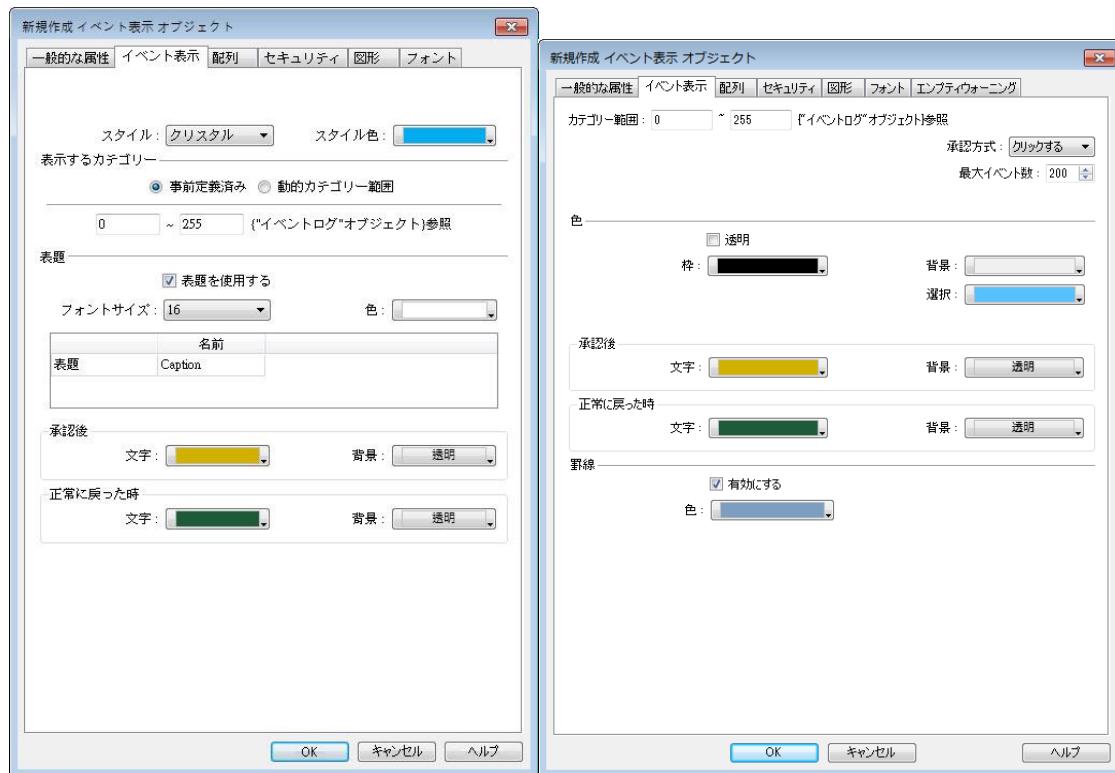
オブジェクトの右上隅にあるじょうごアイコンをクリックすると、開始と終了の日付を設定することができます。もし設定していない場合、全てのイベントが表示されます。



イベント表示設定

cMT シリーズ

eMT、iE、XE、mTV シリーズ



設定

記述

表示するカテゴリ一範囲

イベントの[カテゴリ]はここで設定された範囲に含まれた場合のみ表示されます(イベントのカテゴリは[イベントログ]で設定する)。例えば、[表示するカテゴリ一範囲]で 2~4 に設定された場合、[カテゴリ]は 2、3、4 のイベントのみ、[イベント表示]オブジェクトに表示されます。詳細な情報は《7 イベントログ》に[カテゴリ]に関する説明をご参照ください。

cMT シリーズ

動的カテゴリ一範囲を選択できます。オブジェクトはレジスタ内の数値に基づいて動的に表示するイベントカテゴリをフィルターします。

表示するカテゴリー

事前定義済み 動的カテゴリー範囲

装置 : Local HMI
アドレス : LW 16-bit Unsigned

範囲開始 : LW-0
範囲終了 : LW-1

承認方式 (eMT、iE、XE、mTV シリーズ)	[クリックする]或いは[ダブルクリック]との承認方式を選択することができます。イベントが発生した際に、ユーザーは設定した方式で承認できます。ここでの“承認”という動作とは、既に発生し、それに[イベント表示]オブジェクトで表示されたイベントに対し、当イベントは表示色を[承認]の指定色に変更する以外、当イベントのデフォルトエクスポート値を[承認アドレス]に設定されたレジスターに書き込みます。 仮にエクスポートアドレスは LW-100、またイベントが承認された時に書き込まれた数値は 31 の場合、ユーザーが[承認]を使用すると、LW-100 内の数値は 31 に設定され、本機能を利用して[間接ウインドウ]オブジェクトに合わせることで、異なるイベントが発生した際に異なるウインドウがポップアップされ、イベントの内容を説明します。
最大イベント数(eMT、iE、XE、mTV シリーズ)	オブジェクトが表示できるイベントの最大数量です。オブジェクトが表示するイベントが設定された最大数量に達した場合、新しく発生したイベントは一番先頭にあるデータを入れ替えます。
色	イベントが承認された後、正常に戻った後、クリックされているなどの状態の表示色を設定します。システムは点線でクリックされているイベントを表記します。



履歴記録背景 (eMT, iE, XE, mTV)

イベント表示オブジェクトを使用し、それに履歴モードを選択した場合、履歴記録の背景色を設定することができます。履歴記録が発生した時、背景は選択した色で表示されます。



**罫線**

オブジェクトの上で罫線を引き、また罫線の色を選択することができます。

欄の幅を自動調整する(cMT のデフォルトスタイルにのみ適用する)
内容に基づいて自動的に欄の幅を調整します。

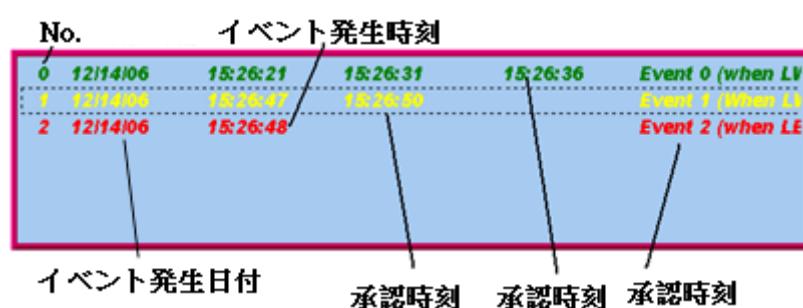
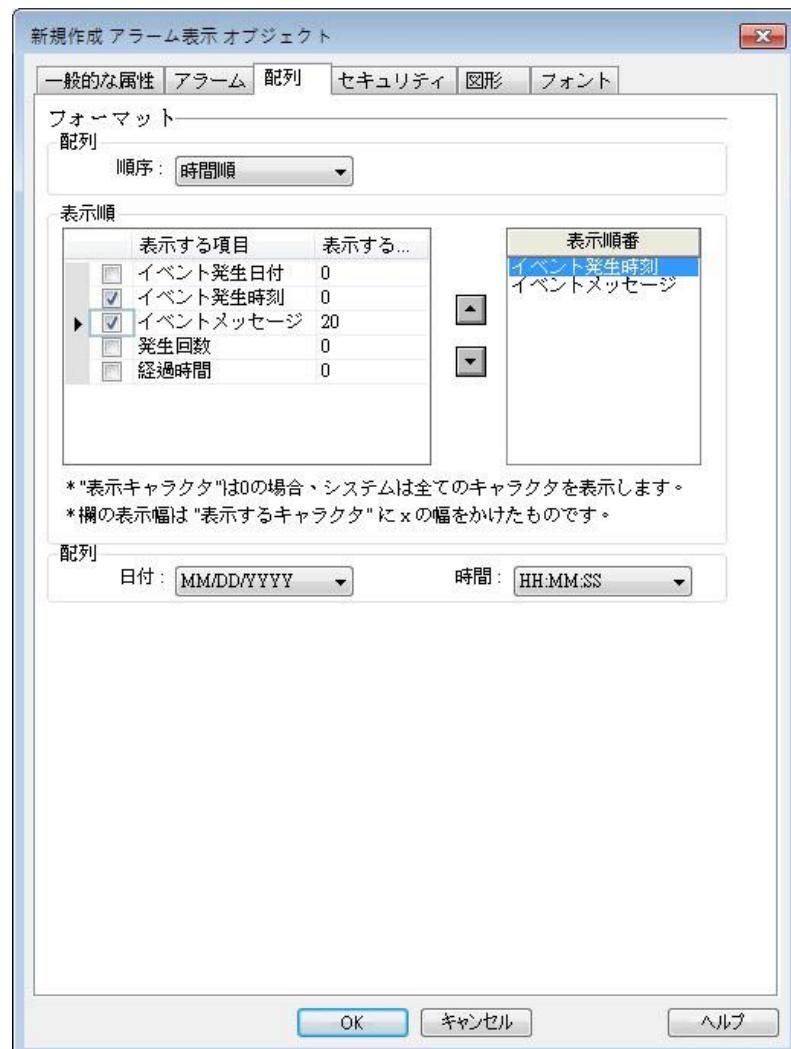
**スタイル
(cMT に適用する)**

[デフォルト]、[クリスタル]、[フラット]スタイルが選べます。

表題を使用する

本項目にチェックマークを入れると、[レシピ閲覧]オブジェクトで表題を表示させることができます。[フォントサイズ]、[色]、[名前]を設定できます。

配列

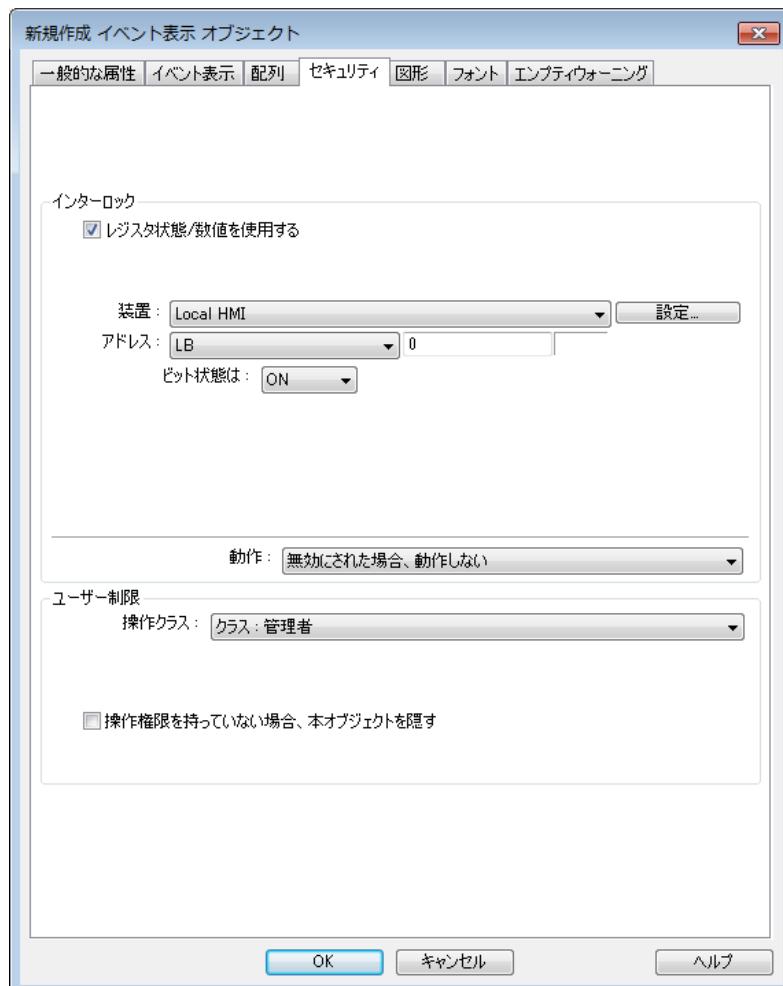


設定	記述
配列	時間順で配列する より遅く発生したイベントは後ろ(または下)に配列されます。 逆時間順で配列する より遅く発生したイベントは前(または上)に配列されます。
表示順	表示してほしい情報及び表示する順を設定することができます。
日付	イベント発生日付の表示フォーマットを選択します。下記 4 モードがあります。

MM/DD/YY、DD/MM/YY、DD.MM.YY、YY/MM/DD

時刻	イベント発生時刻の表示フォーマットを選択します。下記 4 モード があります。 HH:MM:SS、HH:MM、DD:HH:MM、HH
----	--

セキュリティ



設定

記述

インターロック機能を使用する

本項にチェックマークを入れると、このオブジェクトを操作できるかどうかのは、一個の指定したビットアドレスの[状態]によります。上図に示されたように、LB-0 が ON の場合のみ、本オブジェクトを操作できます。

無効の場合、オブジェクトを隠す

指定したビットアドレスが OFF の場合、オブジェクトは隠されます。

ユーザー制限

操作クラス

“無”はどのユーザーでも操作できることを示しています。“管理者”は admin アカウントのみ操作できることを示しています。

操作権限を持っていない場合、本オブジェクトを隠す

ユーザーの操作権限が本オブジェクトのセキュリティクラスに満たしていない場合、オブジェクトが隠されます。

イベント表示オブジェクトを使用する場合、本項にチェックマークを入れなかつたら、ユーザーは依然にイベント表示オブジェクトが見られます。但し、本オブジェクトを変更、またはトリガーすることができます。

フォント

リアルタイムモード：斜体及びサイズを設定することができます。フォントは[イベントログ]での設定で表示されます。

履歴モード：斜体及びサイズ、フォント及び色を設定可能で、または[ラベルライブラリと同じのフォント使用]にチェックマークを入れて使用することができます。



エンプティウォーニング



エンプティウォーニングを使用すると、まだイベントが発生していない時に、オブジェクトに表示する文字を設定できます。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.23. データ転送

13.23.1. データ転送(ウインドウ)

13.23.1.1. 概要

[データ転送(ウインドウ)]オブジェクトは、オブジェクトの所在しているウインドウがオープンされた時に、指定したアドレス内のデータを他のアドレスに転送することができます。[データ転送(ウインドウ)]は手動でボタンを押す方式でデータ転送を起動でき、或いは特定したアドレスの状態変化により、データ転送をトリガーできます。

Note

- [データ転送(ウインドウ)]オブジェクトのビットトリガー機能を使用する時、オブジェクトを共有ウインドウに置けば、ベースウインドウがどれでも関わらず、一旦トリガー条件が満たされると、データ転送が起動されます。

13.23.1.2. 設定



ツールバー上の[オブジェクト]»[オブジェクト]» [データ転送(ウインドウ)]» ボタンを押せば、[データ転送(ウインドウ)]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[データ転送(ウインドウ)]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述
ソースアドレス	データのソースアドレスを設定します。
宛先アドレス	データの宛先アドレスを設定します。
属性	ワード数 データの送信数量です。単位はワードです。 モード 手動モード 手動でデータ転送オブジェクトをクリックしてから、データ転送が実行されます。 トリガーモード 指定したアドレスの状態変更により、データ転送をトリガーします。 [トリガーモード]でトリガーの方式を選択します。下記のような方 式があります： 状態が OFF→ON、或いは ON→OFF になる際に、データ転送の動作を 実行します。または両方とも選択できます(OFF<->ON)。
トリガーアドレス	[トリガーアドレス]でトリガーモードが使用するレジスタアドレスを設定します。
通知	本項を有効にすると、システムはデータ転送をし始める前、指定してレジスタの状態を[ON]、または[OFF]に設定します。 自動的にフォローする システムがデータを転送し始めた時、[通知]アドレスの状態を復帰します。

13.23.2. データ転送(背景)時間ベース型

13.23.2.1. 概要

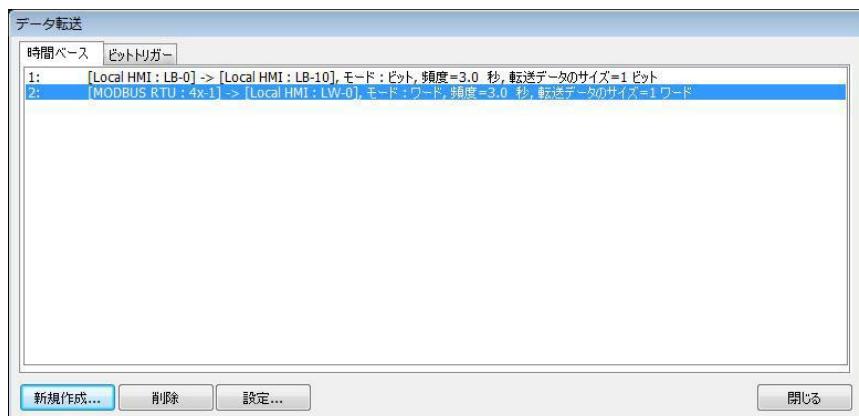
[データ転送(背景)]オブジェクトの時間ベース型は固定した頻度で自動的にデータを他のアドレスに送信することができます。

13.23.2.2. 設定

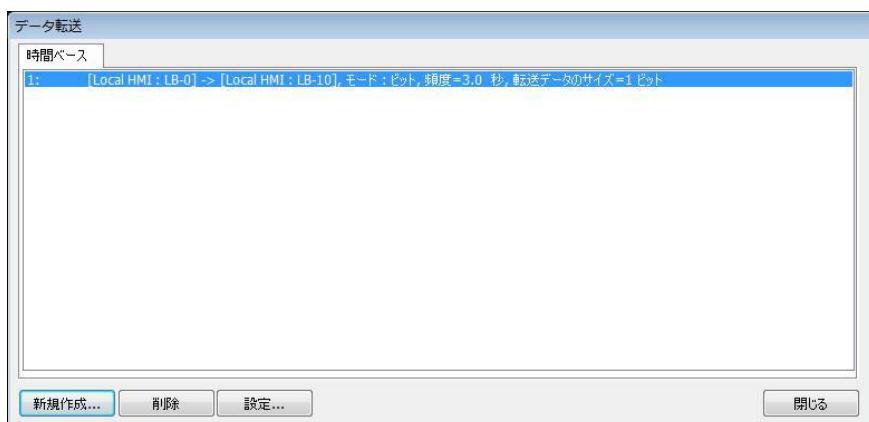


ツールバーの[オブジェクト] » [データ転送(背景)] » [時間ベース]タブ » [新規作成]ボタンを押せば、[時間ベース型データ転送]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[時間ベース型データ転送]オブジェクトが新規作成されます。

cMT シリーズ



eMT、iE、XE、mTV シリーズ



一般的な属性



設定

記述

属性

アドレスタイプ

転送するデータのタイプを選択します。[Bit]或いは[Word]のデータが選択可能です。

ビット数/ワード数

[アドレスタイプ]で[Bit]を選択する場合、データ転送単位は bit で、[ビット数]を使用し送信数量を設定します。

[アドレスタイプ]で[Word]タイプを選択する場合、データ転送単位は word で、[ワード数]で転送する数量を設定します。

間隔

データ転送の頻度を設定します。例えば 3 秒を選択すると、3 秒ごとにデータを指定したアドレスに送信します。

小さい間隔にまたは大量なデータを送信する場合、システムの実行速度が遅くなる可能性があります。こういう状況を避けるには、送信する間隔を長くし、または一回で少量のデータを送信することがお勧めです。

短時間での送信が必要の場合、間隔時間はデータ転送時間より長く

設定する必要があります。例えば：一回でのデータ転送時間は2秒の場合、間隔時間は2秒を超えるように設定しなければなりません。

指定したウインドウがオープンされた場合のみ実行

本項目にチェックマークを入れると、データ転送は指定されたウインドウがオープンされた場合のみ実行されます。

ソースアドレス

データのソースアドレスを設定します。

宛先アドレス

データの宛先アドレスを設定します。

通知

自動重置

當系統開始資料傳輸後、恢復 [通知] 位址至原狀態。

本項目を有効にすると、システムはデータ転送をし始める前、指定してレジスタの状態を[ON]、または[OFF]に設定します。

自動的にフォローする

システムがデータを転送し始めた時、[通知]アドレスの状態を復帰します。

13.23.3. データ転送(背景)ビットトリガー型

13.23.3.1. 概要

[データ転送(背景)]オブジェクトのビットトリガー型は、バックグラウンドで継続に指定したビットアドレスの状態を監視し、変化が検出するとデータ転送をトリガーして他のアドレスにデータを転送します。

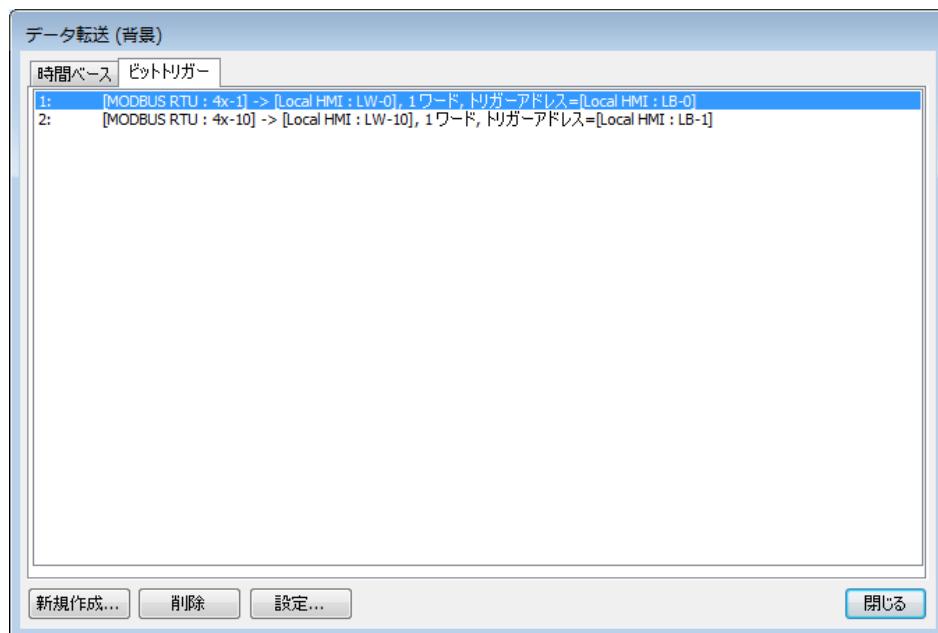
eMT、iE、XE、mTV シリーズは本機能をサポートしません。

13.23.3.2. 設定



ツールバー上の[オブジェクト] » [オブジェクト] » [データ転送(背景)] » [ビットトリガー]タブ » [新規作成]ボタンを押せば、[データ転送(トリガー型)]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[データ転送(トリガー型)]オブジェクトが新規作成されます。

cMT シリーズ



一般的な属性



設定	記述
ソースアドレス	データのソースアドレスを設定します。
宛先アドレス	データの宛先アドレスを設定します。
属性	ワード数 データの送信数量です。単位：ワード。
トリガーアドレス	トリガーを実行するレジスタアドレス及びトリガーモードを設定します。 トリガーモード 指定した状態が変更されたたびに、データ転送を実行します。 状態が OFF→ON、或いは ON→OFF になる際に、データ転送の動作を行います。または両方とも選択できます(OFF<->ON)。
通知	本項を有効にすると、システムはデータ転送をし始める前、指定してレジスタの状態を[ON]、または[OFF]に設定します。 自動的にフォローする システムがデータを転送し始めた時、[通知]アドレスの状態を復帰します。

13.24. バックアップ

13.24.1. 概要

[バックアップ(ウインドウ)]または[バックアップ(背景)]オブジェクトを利用すると、レシピデータ(RW、RW_A)、イベントログ、レシピデータベース、サンプリングデータ及び操作記録を外部記憶装置(SDカードまたはUSBディスク)に転送し、またはバックアップの時間範囲とフォーマットを指定することができます。例えばイベントログは元々SDカードに保存したが、この場合はHMIを起動しているままでUSBディスクを接続し、[バックアップ]オブジェクトでデータをコピーしUSBディスクに伝送することができます。それにシャットダウンしていないまま直接にUSBディスクを取り外し、PCでデータを分析することができます。

バックアップ動作を実行している際に、[LB-9039]の状態はONに維持します。また、[メール]機能を合わせて、データを添付ファイルの形で指定した宛先へメールを送ることができます。

[バックアップ(ウインドウ)]と[バックアップ(背景)]オブジェクトの違いは、[バックアップ(ウインドウ)]オブジェクトはオブジェクトの所在ウインドウのみで実行され、[バックアップ(背景)]オブジェクトのほうは各ウインドウでも実行されます。[バックアップ(背景)]オブジェクトはcMTシリーズのみサポートします。

13.24.2. 設定



ツールバー上の[オブジェクト]»[バックアップ]を押せば、[バックアップ(ウインドウ)]または[バックアップ(背景)]オブジェクトを選択できます。

[バックアップ(ウインドウ)]を選択すれば、オブジェクトの属性ダイアログボックスがオープンされます。正確に各属性を設定完了してOKボタンを押すと、[バックアップ(ウインドウ)]オブジェクトが新規作成されます。

cMTシリーズを使用する場合、[バックアップ(背景)]オブジェクトを選択できます。バックアップ(背景)オブジェクト管理ダイアログボックスで[新規作成]ボタンを押し、正確にオブジェクトの各属性を設定してOKボタンを押すと、[バックアップ(背景)]オブジェクトが新規作成されます。

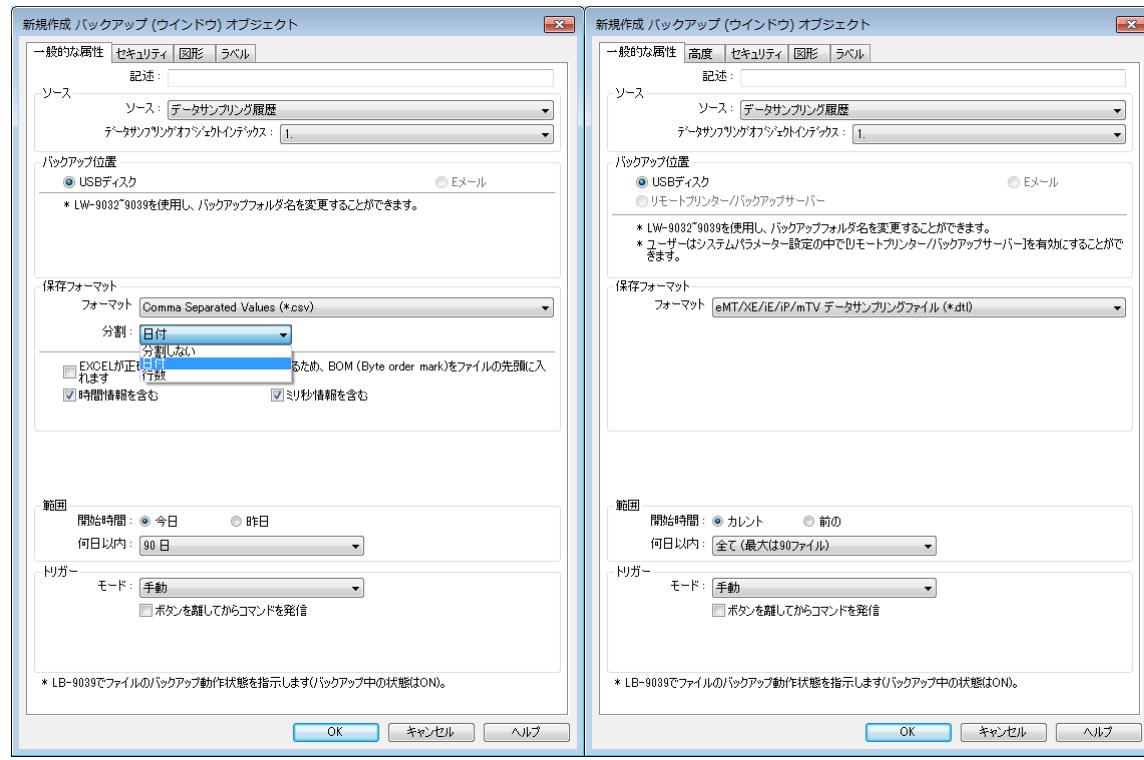
オブジェクト

一般的な属性の設定

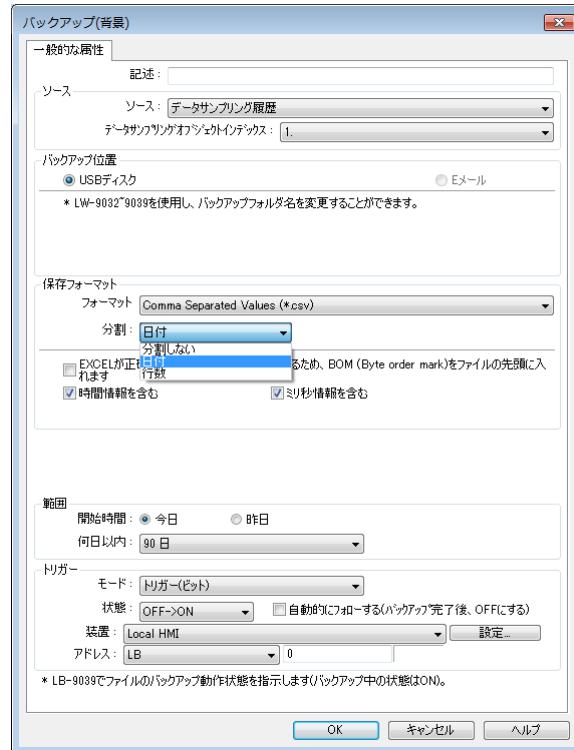
バックアップ(ウインドウ)オブジェクト

cMT シリーズ

eMT、iE、XE、mTV シリーズ



バックアップ(背景)オブジェクト : cMT シリーズ



設定

記述

ソース [RW]、[RW_A]、[レシピデータベース]、[イベントログ履歴]、[データサンプリング履歴]、[操作ログ]、[SQL クエリ]

これらの選択肢でバックアップしたいデータのデータソースを指定します。例えば、データソースを[データサンプリング履歴]を指定する場合、[データサンプリングオブジェクトインデックス]でコピーしたいデータサンプリング履歴を選択する必要があります。

[RW]、[RW_A]以外、他のオプションは EasyBuilder Pro のプロジェクトで使用された場合のみ、表示されます。

バックアップ位置 ソースファイルのバックアップ位置を設定します。

SD カード/USB ディスク

SD カード或いは USB ディスクにバックアップします。外部装置は予め HMI に接続する必要があります。

リモートプリンター/バックアップサーバー(eMT, iE, XE, mTV シリーズのみ適用)

バックアップサーバーにバックアップします。本項を使用すると、ユーザーは事前に[システムパラメータ設定] » [リモートプリンター/バックアップサーバー]で設定する必要があります。

注意すべきなのは、操作ログはリモートプリンター/バックアップサーバーのみにバックアップできます。SD カード/USB ディスクにバックアップしたい場合、操作ログオブジェクトのコントロールアドレスをご使用ください。

 詳細情報は、《26 EasyPrinter》をご参照ください。

メール

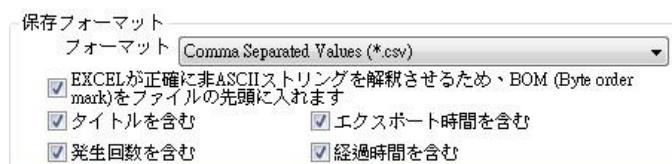
バックアップされたデータを添付ファイルでメールします。

本項を使用すると、ユーザーは事前に[システムパラメータ設定] » [メール]で設定する必要があり、それから[バックアップ]オブジェクト属性 » [E メール]で宛先、件名、メッセージなどのメール設定をします。

保存フォーマット バックアップするデータのフォーマットを選択します。

eMT, iE, mTV シリーズ :

- HMI イベントログファイル(.evt)/HMI データログファイル(.dtl)
- Comma Separated Values(.csv)



また、イベントログのバックアップデータには[イベント]の欄が含ま

れています。そこでイベントのタイプを表示します。

A	B	C	D	E
Event	Category	Date	Time	Message
2	0	1	2013/7/4	16:12:11 Event A
3	2	1	2013/7/4	16:12:12 Event A
4	0	0	2013/7/4	16:12:33 Event B
5	2	0	2013/7/4	16:12:36 Event B
6	0	0	2013/7/4	16:12:37 Event B
7	1	0	2013/7/4	16:12:37 Event B
8	2	0	2013/7/4	16:12:39 Event B
9	0	0	2013/7/4	16:12:40 Event B

0:イベントがトリガーされた

1:イベントが承認された

2:イベントが正常に戻った

HMI イベントログファイル(.evt)と HMI データログファイル(.dtl)は EasyConverter を使用することで、.xls または.csv フォーマットに変換できます。

- SQLite データベースファイル(.db) (E メールにバックアップする場合に限る)

cMT シリーズ：

- SQLite データベースファイル(.db)
- Comma Separated Values(.csv)

分割方式

分割しない	履歴データのバックアップを全部一つのファイルに保存します。
日付	履歴データを日付を単位にしてバックアップします。データサンプリングに[カスタムファイル管理]に設定している場合、ファイル個数を単位にしてバックアップします。
行数	履歴データを指定した行数を単位にしてバックアップします。

EXCEL が正確に非 ASCII ストリングを判読させるため、BOM をファイルの先頭に入れます

バックアップするイベントファイルのフォーマットは CSV の場合、ファイルの先頭で BOM(Byte Order Mark)を加え、Excel で直接に非 ASCII 内容の.csv ファイルをオープンすることができます。それにタイトル、エクスポート時間、発生回数及び経過時間の情報を含むのかを選択できます。



イベントカテゴリ一範囲(イベントログ履歴に合わせて使用する必要がある) イベントログをバックアップする時、[保存フォーマット]は CSV ファイルの場合のみ、本オプションを使用することができます。カテゴリ一範囲は[全て]と[部分]に分けられます。例に挙げると、[部分]の欄に 3、5、8 を入力すると、バックアップするファイルにはカテゴリー3、5、8 のものが含まれます。3-8 を入力すると、バックアップするファイルはカテゴリー3~8 のものが含まれます。

範囲 **データサンプリング履歴**
バックアップするファイル数を設定します。
例えば、開始時間を[カレント]を設定し、5 ファイルを選択したら、メモリー内の最新の 5 ファイルをバックアップすることを示しています。

イベントログ履歴
バックアップする日数を選択します。例えば、開始時間を[昨日]に設定して 2 日を選択する場合、昨日及び一昨日のデータをバックアップします。[全て]を選択する場合、最大 90 日のデータをバックアップすることができます。

トリガー **モード**
バックアップオブジェクトの実行方式を選択します。
手動

ユーザーはオブジェクトを押すだけで、バックアップ動作を実行することができます。

トリガー(ビット)
指定したレジスタの状態がトリガー条件に満たされたようになった際に、オブジェクトはバックアップ動作を実行します。
また、状態が OFF→ON、或いは ON→OFF になる際に、バックアップ動作を行います。または両方とも選択できます(OFF<->ON)。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

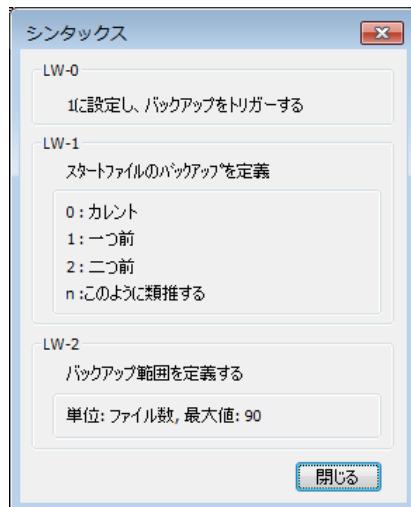
トリガー(ワード)
[トリガーアドレス]を通じてバックアップしたいデータの時間範囲を設定することができます

[トリガーアドレス]の使用方法は以下のとおりです(仮に現在のトリガーアドレスは LW-0 に設定された場合)：

LW-0:本アドレスが指定したレジスタ内の数値が 0→1 になる場合、バックアップ動作がトリガーされます。

LW-1:本アドレスが指定したレジスタ内の数値でバックアップの開始時間を指定します。

LW-2:本アドレスが指定したレジスタ内の数値でバックアップの日数を指定します(最大 90 日)。



cMT-SVR での[バックアップ(ウインドウ)]オブジェクトは[手動]のみサポートし、[トリガー(ビット)]と[トリガー(ワード)]をサポートしません。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

トリガーアドレス

指定したレジスタの状態が ON に設定した際に、オブジェクトはバックアップ動作を実行します。バックアップが完了後、当レジスタは OFF に設定されます。

Note

- 注意すべきなのは、全てのデータは任意のメモリーにセーブされた必要があります(例:HMI memory/USB ディスク/SD カード)。そうでないと、バックアップ機能が使われません。
- 一回でのバックアップできる日数は最大 90 日です。
- ファイルを USB ディスク、または SD カードに保存する時、一個の FAT32 フォルダが保存されるファイル数はファイル名の長さによります。ファイル名が長ければ長いほど、フォルダ内に保存されるファイル数が少なくなります。

 cMT シリーズがデータサンプリング履歴及びイベントログ履歴を外部装置に同期する方式については、《7 イベントログ》、《8 データサンプリング》をご参照ください。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

高度設定(eMT、iE、XE、mTV シリーズ)

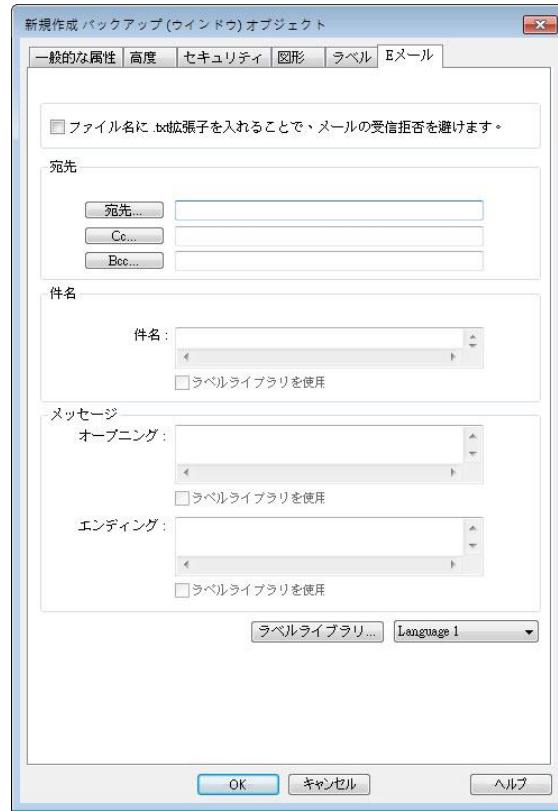


設定	記述
シリアル番号	<p>有効にすると、履歴データをバックアップする際に、ファイル名の末にユーザー定義のシリアル番号を付けることができます。</p> <p>それで LW アドレスで設定した数値をファイル名の末に 5 数字の番号をつけます。</p> <p>バックアップが完成した後、システムは自動的に LW アドレス+1 になります。</p> <p>シリアル番号の範囲は 0~65535 です。</p> <p>例えば:シリアル番号は 123 の場合、00123 がファイル名の末につけられます</p> <p>データサンプリングファイル-20140407.dtl は 2014040700123.dtl になります</p> <p>イベントログファイル-20140407.evt は 2014040700123.evt になります。</p>
オプション	<p>バックアップした後、旧ファイルを削除する</p> <p>本機能を使用すると、バックアップした後、古い履歴ファイルが自動的に削除されます。</p>

Note

- cMT シリーズは高度設定をサポートしていません。

Eメール設定



設定	記述
ファイル名に.txt拡張子を入れることで、メールの受信拒否を避けます。	本項目を有効にすると、バックアップデータをEメールの添付資料で送信する場合、ファイル名の末尾で.txt拡張子を入れることでメールサーバー、またはアンチウィルスソフトにメール受信拒否をされることを避けます。
宛先、件名、メッセージ	受信者のメールアドレス、Eメールの件名、メッセージの内容をバックアップします。

13.25. メディアプレーヤー

13.25.1. 概要

初めてメディアプレーヤーを使用する時、イーサネットでプロジェクトを HMI にダウンロードする必要があります。EasyBuilder Pro は自動的にメディアプレーヤーのドライバをインストールします。メディアプレーヤーの機能は映像を再生するだけでなく、他の操作機能も提供しています。例えば：検索、拡大/縮小、音量調整などです。

映像を通じて作業及びメンテナンスの手順を指導することで、現場の作業員により分かりやすく、容易にメンテナンスを実行できる環境を建ちます。

cMT シリーズ HMI を使用する場合、cMT Viewer では本オブジェクトが見られません。

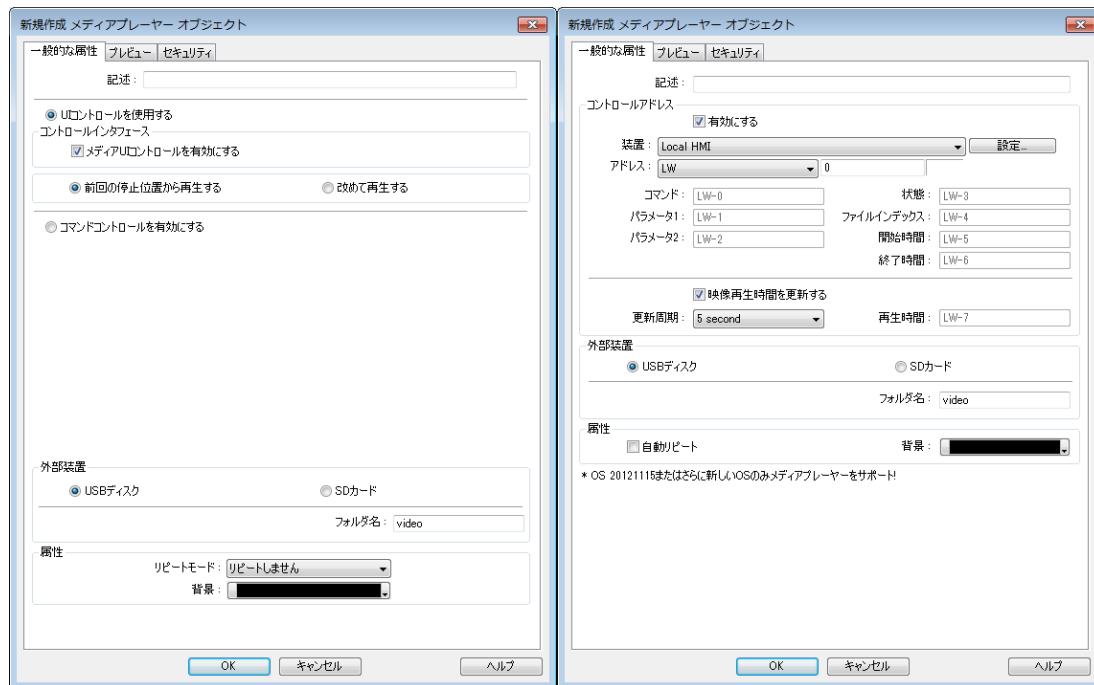
13.25.2. 設定



ツールバー上の[オブジェクト] » [オブジェクト] » [メディアプレーヤー]ボタンを押せば、[メディアプレーヤー]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[メディアプレーヤー]オブジェクトが新規作成されます。

一般的な属性の設定

cMT シリーズ(cMT-SVR、cMT-HDM、cMT-FHD 以外) eMT、XE シリーズ



設定	記述
UI コントロールを使用する	<p>ユーザーインターフェイスで提供されるメディアコントロール機能を使用します。</p> <p>前回の停止位置から再生すると改めて再生する：他のウィンドウからメディアプレーヤーがあるウィンドウに戻るとき、映像を再生したい時点を設定します。</p>
コントロールアドレス	<ul style="list-style-type: none"> ● 有効にする ユーザーは[メディアプレーヤー]をコントロールし、及び再生情報を取得することができます。一個のアドレスをオブジェクトのコントロール用に指定します。 ● 有効にしない 手動で映像の再生状態をコントロールすることができません。ウィンドウがオープンされた際に、システムは自動的に映像を再生します。 <p>コマンド(コントロールアドレス+0) コマンドレジスタに数値を入力し、[メディアプレーヤー]の動作モードをコントロールします。</p> <p>パラメータ 1(アドレス+1) パラメータ 1 の数値を入力します。このパラメータ 1 は、各コマンドの動作に関連しています。</p> <p>パラメータ 2(コントロールアドレス+2) パラメータ 2 の数値を入力します。このパラメータ 2 は、各コマンドの動作に関連しています。</p> <p>状態(コントロールアドレス+3) ファイルの状態、再生状況及びエラーを記録します。</p> <p>ファイルインデックス(コントロールアドレス+4) 再生するファイルが指定したディレクトリの中でのインデックスです。(ファイル名で順位付けます。数字でファイル名を命名するのがお勧めです)。</p> <p>開始時間(コントロールアドレス+5) 映像の開始時間です(秒)。(一般は 0)</p> <p>終了時間(コントロールアドレス+6) 映像の終了時間です(秒)。(映像の時間長さ)</p> <p>映像再生時間を更新する 有効にすると、システムは[更新周期](秒)ごとに映像の再生時間を[再生時間]レジスタに書き込みます。</p> <p>更新周期</p>

	[再生時間]を更新する周期です。範囲は1~60秒です。
	再生時間(コントロールアドレス+7)
	映像が再生された時間(秒)([開始時間]と[結束時間]の間にある)です。
外部装置	SD/USB内のファイルを再生します。
	フォルダ名
	映像ファイルを保存するフォルダの名前です。ファイルはルートディレクトリに保存する必要があり、サブディレクトリは受け入れられません。(例:[フォルダ名]を"example\ex"に設定したら、エラーが出ます)。
	[フォルダ名]は空白してはいけません。英数字で、全部ASCIIキャラクタで組み合わせする必要があります。
属性	自動リピート
	全ての映像が再生完了したら、自動的に一番目の映像に戻り、改めて再生します。
	例: Video1>Video2>Video1>Video2
	背景 オブジェクトの背景色を指定します。

Note

- デフォルトレジスタフォーマットは16ビットで正負符号なしです。指定したレジスタは32ビットの場合、より下位の16ビットだけが作用し、より上位の16ビットを0に設定してください。

コントロールコマンド

下記では各コントロールコマンドの設定を説明します。

- インデックスファイルを再生する

[コマンド]=1
[パラメータ1]=ファイルインデックス
[パラメータ2]=省略(0に設定)

Note

- ファイル名順で配列します。
- ファイルが見つからない場合、[状態]のビット8をONに設定します。
- 途中で映像を切り替える必要があったら、再生中の映像を中止してください。

- 1本前の映像を再生する

[コマンド]=2
[パラメータ1]=省略(0に設定)
[パラメータ2]=省略(0に設定)

Note

- [ファイルインデックス]は 0 の場合、コマンド 2 を入力すると、元のファイルが改めて再生されます。
- ファイルが見つかれない場合、[状態]のビット 8 は ON に設定され、コマンドが誤ったと示しています。

- 次の映像を再生する

[コマンド]=3
[パラメータ 1]=省略(0 に設定)
[パラメータ 2]=省略(0 に設定)

 **Note**

- ファイルが見つかれない場合、インデックス値 0 のファイルが再生されます。
- ファイルが見つかれない場合、[状態]のビット 8 は ON に設定され、コマンドが誤ったと示しています。

- 一時停止/再生する 切替

[コマンド]=4
[パラメータ 1]=省略(0 に設定)
[パラメータ 2]=省略(0 に設定)

- 再生を停止し、ファイルを閉じる

[コマンド]=5
[パラメータ 1]=省略(0 に設定)
[パラメータ 2]=省略(0 に設定)

- 指定時刻から再生する

[コマンド]=6
[パラメータ 1]=目標時刻(単位:秒)
[パラメータ 2]=省略(0 に設定)

 **Note**

- パラメータ 1(目標時刻)は終了時刻より小さく設定する必要があります。終了時刻に越えた場合、終了時刻の一秒前から再生し始めます。

- 早送り(秒)

[コマンド]=7
[パラメータ 1]=目標時刻(単位:秒)
[パラメータ 2]=省略(0 に設定)

 **Note**

- 現在の時刻から先へ[パラメータ 1]で指定した秒数に飛んで再生します。再生が一時停止された場合、再生し始めてから飛びます。

- 再生時刻が終了時刻に越えた場合、終了時刻の一秒前から再生し始めます。

- 早戻し(秒)

[コマンド]=8

[パラメータ1]=目標時刻(単位:秒)

[パラメータ2]=省略(0に設定)

 Note

- 現在の時刻から元へ[パラメータ1]で指定した秒数に飛んで再生します。再生が一時停止された場合、再生し始めてから飛びます。
- 再生時間が開始時間より早い場合、映像を最初から再生します。

- 音量

[コマンド]=9

[パラメータ1]=音量(0~128)

[パラメータ2]=省略(0に設定)

 Note

- 最大音量のデフォルトは(128)です。

- 映像の表示サイズ

[コマンド]=10

[パラメータ1]=表示サイズ(0~16)

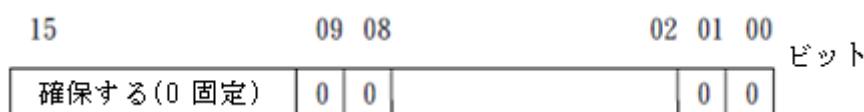
[パラメータ2]=省略(0に設定)

 Note

- [パラメータ1=0]: オブジェクトの大きさに合わせます。
- [パラメータ1=1~16]: 拡大倍率 25%~400%を示しています。1は25%に拡大、2は50%に拡大、3は75%に拡大...このように類推します。

- 状態(コントロールアドレス+3)

再生している間に、[ファイルオープンビット]のビット00及び[ファイル再生ビット]のビット01は同時にONに設定されます(0→1)。一方、ファイルが見つかりなかった、または入力したコマンドが誤った場合、[コマンドエラービット]のビット08はON(0→1)に設定されます。再生中でファイルフォーマットまたはI/Oに錯誤(例:USBディスクが取り外された)が見つかった場合、[ファイルエラービット]のビット09はON(0→1)に設定されます。

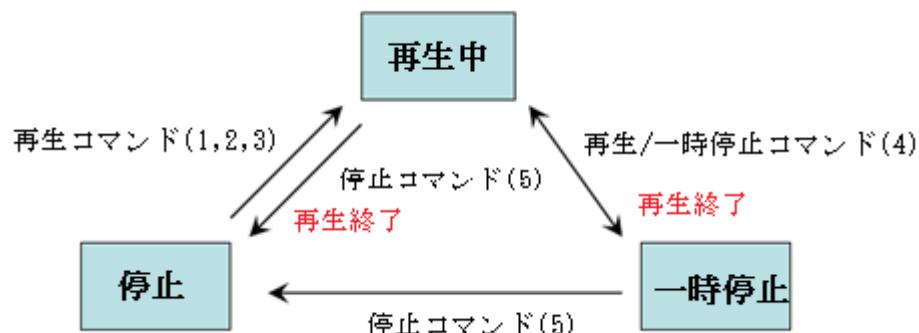


ビット00: ファイルオープンビット

(0: ファイルをオープンしていない、1: ファイルをオープンしている)
 ビット 01: ファイル再生ビット
 (0: ファイルを再生していない、1: ファイルを再生している)
 ビット 08: コマンドエラービット
 (0: 正確なコマンド、1: 不正確なコマンドまたはパラメータ)
 ビット 09: ファイルエラービット
 (0: ファイルのフォーマット及び読み取りが正確、1: ファイルのフォーマット及び読み取りが不正確)

Note

- 下記[メディアプレーヤー]の状態変換図をご参照ください。
 “停止”の際の状態=0, “一時停止”の際の状態=1, “再生”の際の状態=3



- [コマンド]、[パラメータ 1]及び[パラメータ 2]を設定することでオブジェクトをコントロールし、他のアドレスを読み取り専用に視します。

プレビュー

ユーザーはプレビュー機能を使用し、HMI が再生したい映像をサポートしているかを確認することができます。



設定	記述
早送り<</早戻し>>	早送り、または早戻しします(単位：分)。
再生/一時停止	映像を再生または一時停止します。
停止	再生を停止し、ファイルを閉じます。別の映像をテストしたい場合、現在の映像を一時停止する必要があります。
ロード	プレビューしたい映像を選択します。

 Note

- HMI では同一時間で一本の映像だけが開けます。
- [コントロールアドレス]を使用しなく、また[自動リピート]も使用しない場合、指定したフォルダ内の一一番目のファイルが再生し終えたら、システムは自動的に映像を閉じます。
- [コントロールアドレス]を使用していない場合、システムは自動的に指定したフォルダに一番目のファイルを探し出して(ファイル名で順位付け)、再生します。
- 映像がメディアプレーヤーのプレビュー機能を使用できる場合、HMI はこの映像のフォーマットをサポートしていて、再生可能ということを示しています。再生品質が良くない場合、映像の解像度を調整してください。
- サポートするファイルフォーマットは:mpeg4、xvid、flv...などです。

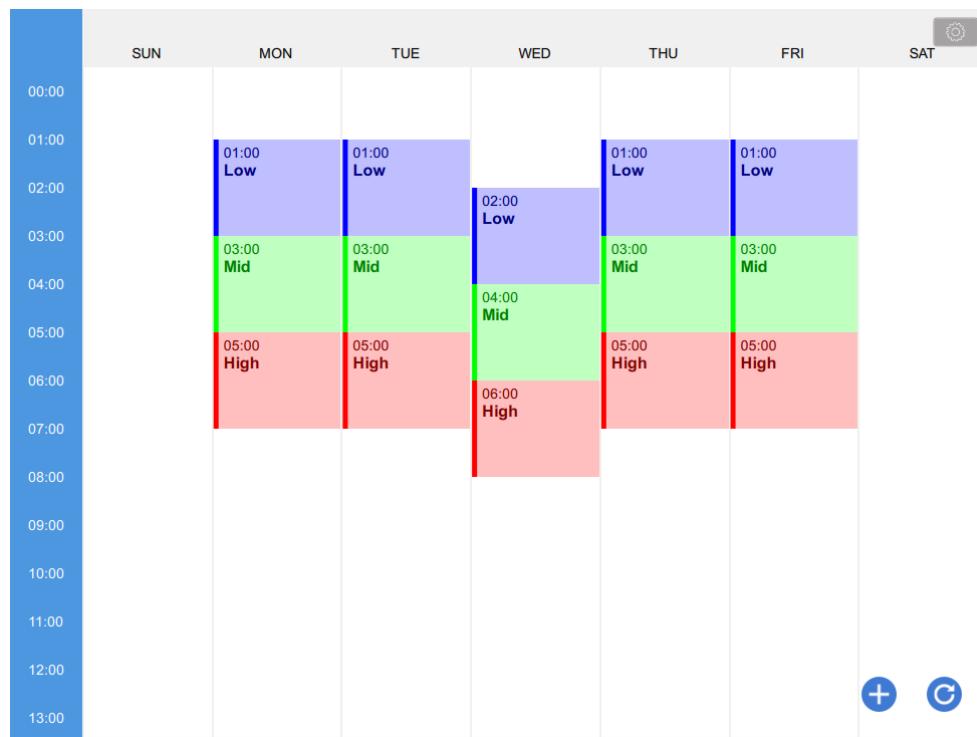


このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.26.BACnet Schedule

13.26.1. 概要

[BACnet Schedule]オブジェクトは BACnet IP 装置内のスケジュール(Schedule)情報をアクセスさせる画像化インターフェースを提供します。本機能は cMT シリーズにのみサポートされます。



13.26.2. 設定



[システムパラメータ]の[装置]で BACnet/IP ドライバを新規追加し、スキャンする BACnet/IP 装置もしくはインポートするタグファイルに Schedule が含まれていることを確認してください。ツールバーの[オブジェクト] » [時間関連] » [BACnet Schedule]ボタンを押せば、[BACnet Schedule]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[BACnet Schedule]オブジェクトが新規作成されます。

一般的な属性

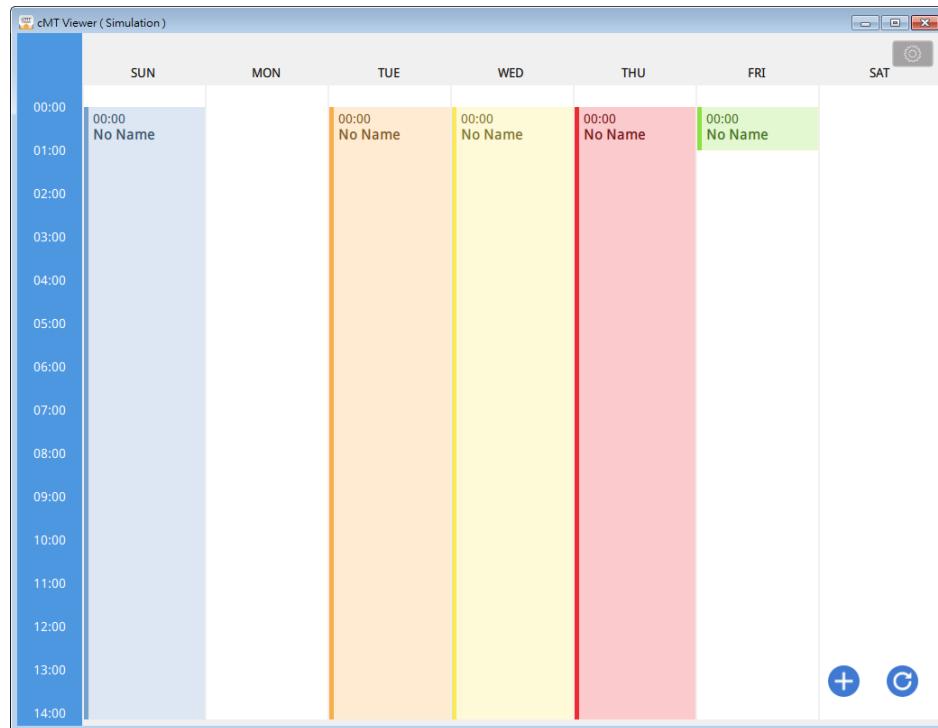


設定	記述
装置	読み取りたい BACnet/IP 装置を選択します。
アドレス	読み取りたい Schedule タグのアドレスを選択します。

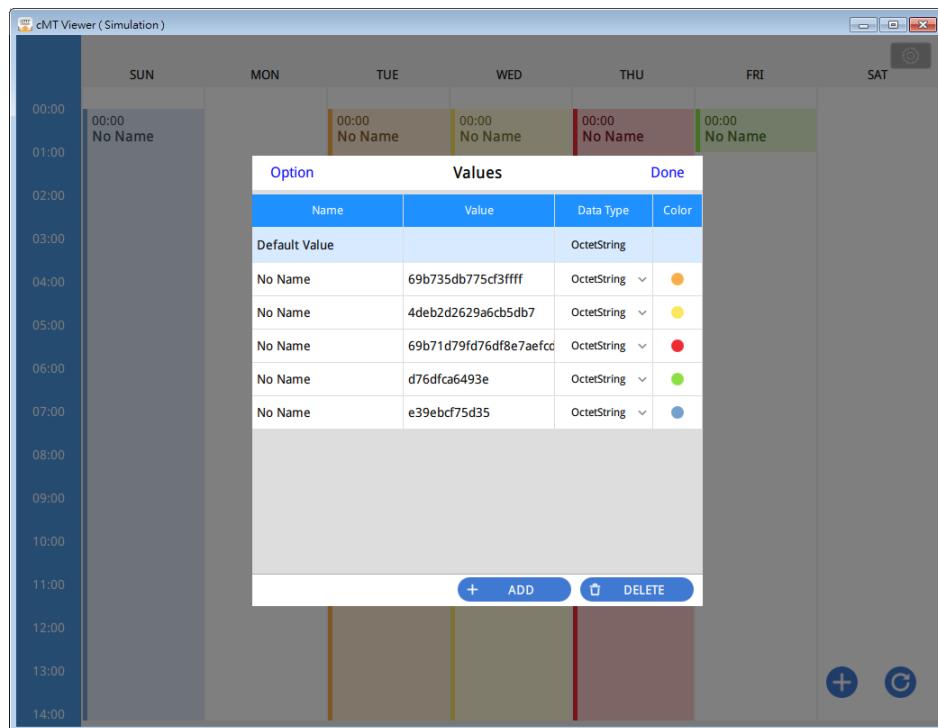
例 1

プロジェクトが HMI にダウンロードされた後、自動的に BACnet IP 装置内のスケジュール設定を読み取ります。下図を例にします：

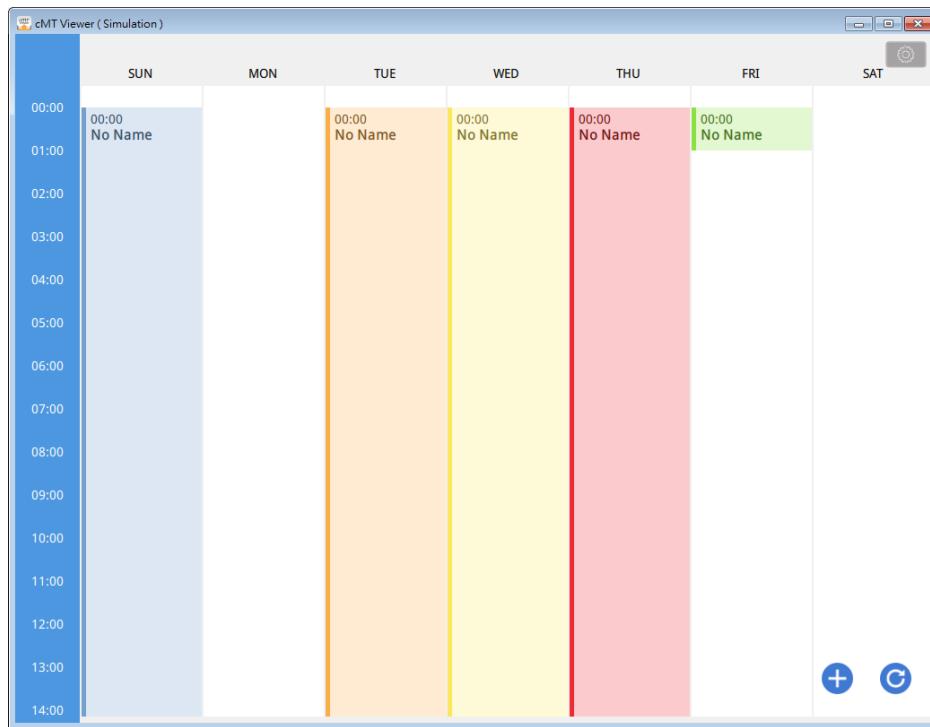
オブジェクト



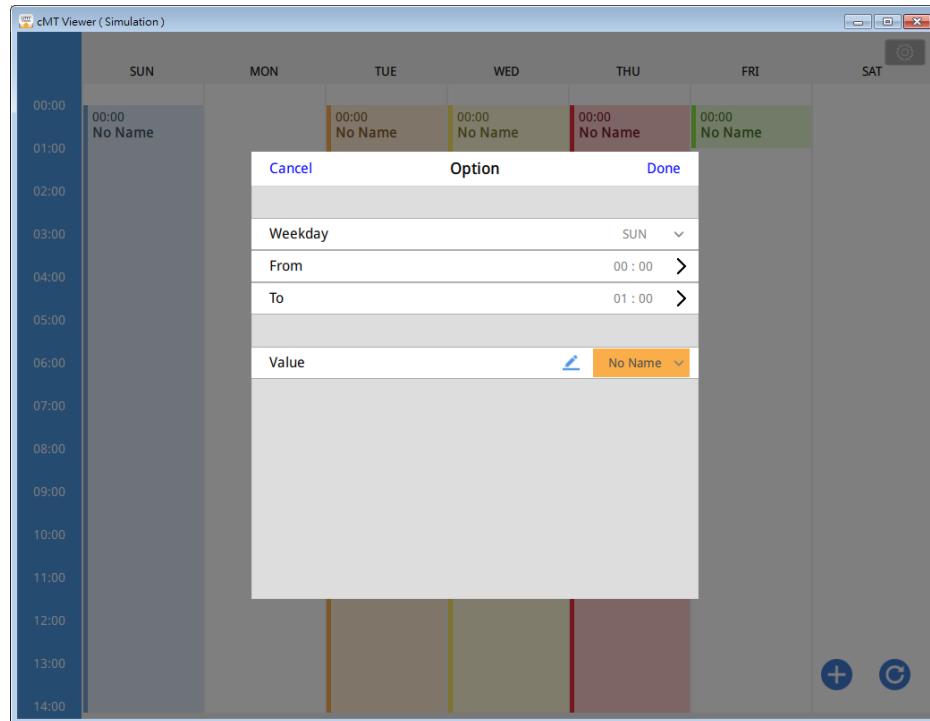
をクリックしてから、Value Editor をクリックすれば、数値テーブルを作成でき、名前(Name)、値(Value)、データ型(Data Type)、色(Color)が含まれます。下図をご参照ください：



右下の をクリックすると、1セットのスケジュール時刻を追加できます。 をクリックすると、改めて BACnet 装置からスケジュールデータを読み取ることができます。



下図に示されたとおり、1セットの設定では、曜日(Weekday)、開始時刻(From)、終了時刻(To)と値(Value)を設定する必要があります。Value欄は数値テーブルから選択でき、もしくはをクリックすれば、Value Editorに戻って数値テーブルを編集することも可能です。



追加した時刻の設定が完了したら、数値は BACnet 装置内の対応のスケジュールタグに書き込まれます。

13.27.PLC コントロール

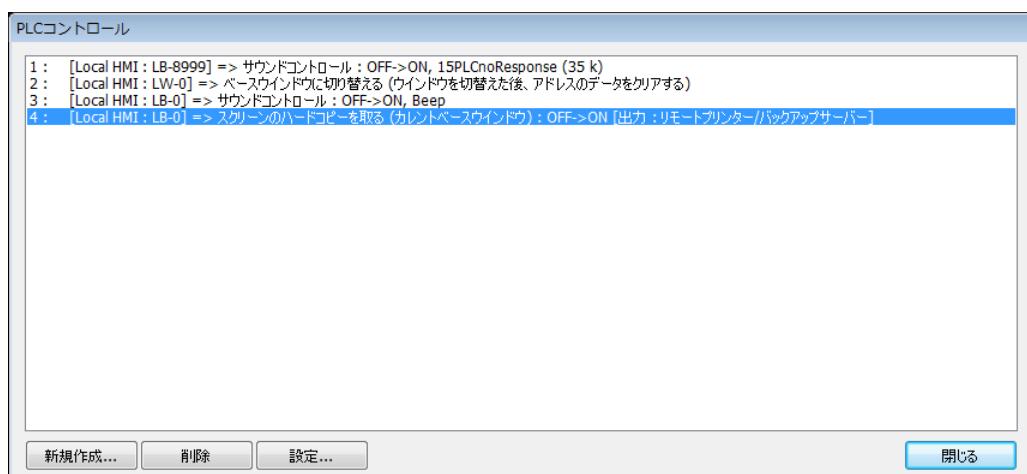
13.27.1. 概要

対応するコントロールコマンドがトリガーされた場合、[PLC コントロール]オブジェクトはある特定な動作を実行する機能です。

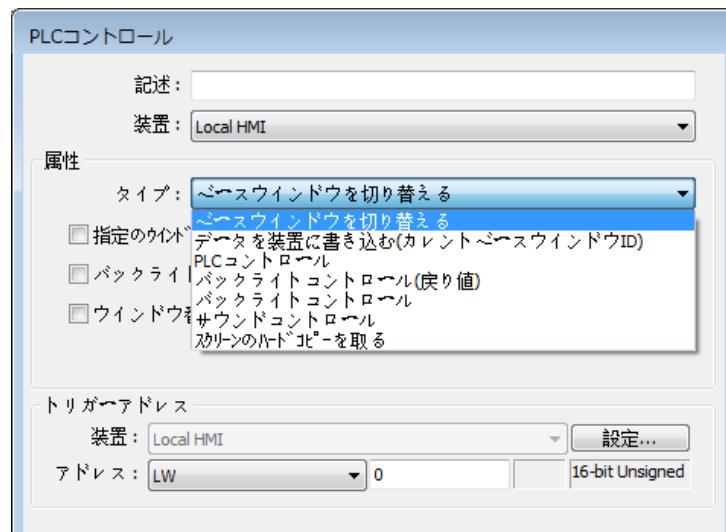
13.27.2. 設定



ツールバー上の[オブジェクト]»[オブジェクト]»[PLC コントロール]ボタンを押せば、[PLC コントロール]オブジェクトの管理ダイアログボックスがオープンされ、そこで[新規作成]ボタンを押し、現れた[PLC コントロール]管理ダイアログボックスで正確に各属性を設定完了して OK ボタンを押すと、[PLC コントロール]オブジェクトが新規作成されます。



下図は[新規作成]ボタンを押した後で現われた設定ダイアログボックスです。次の《13.27.2.1 コントロールタイプ》をご参照ください。



 Note

- cMT シリーズは[PLC コントロール]機能をサポートしていません。
- cMT シリーズ HMI を使用し、cMT Viewer で[スクリーンのハードコピーを取る]機能をトリガーする場合、ファイルは HMI に接続されている USB/SD カードに保存されます。[バックライトコントロール]機能をトリガーする場合、HMI のバックライトの開閉を制御することになります。

13.27.2.1. コントロールタイプ

- ベースウインドウを切り替える



設定	記述
指定のウイン ドウがオープ ンされた場合 のみ実行	ウィンドウを切り替える機能は指定したウインドウのみで作用できます。
バックライト をつける	本項を有効にすると、バックライトが OFF 状態の場合、ウインドウを切り替えたら、バックライトが自動的につけられます。
ウィンドウ切 替後、アドレス 内の数値をク リア	本項を使用すると、バックライトが閉じた状態にいる際に、ウインドウの切替に成功したら、自動的にバックライトをつけます。cMT シリーズは適用しません。
ウインドウを 切替えた後、ア ドレス内の数 値をクリア	本項を使用すると、ウインドウの切替に成功したら、アドレス内のデータがクリアされます。[ウィンドウ番号オフセットを使用]にチェックマークを入れた場合、オフセットがマイナスに設定された時のみ、本項が現れます。
ウインドウ番 号オフセット を使用	本項を選択すると、[トリガーアドレス]が指定したレジスタ内のデータに[ウィンドウ番号オフセット]を加えた結果が、実際の切替先ウインドウ番号になります。例えば：トリガーアドレスは LW-0、オフセットは-10 に設定したら、LW0 の数値は 25 の場合、15 番のウインドウ(数値 25+オフセット-10)に切り替えます。オフセット範囲は

-1024 から 1024 までです。

Note

- [LB-9017]の状態が ON に設定された場合、切り替えたウインドウ番号は特定のアドレスに書き込みません。
- ベースウインドウを切り替える機能を有しています。[トリガーアドレス]内の数値が変更され、また変更後のデータは一個の有效なウインドウ番号である場合、現在のウインドウは閉じられ、[トリガーアドレス]内の数値で指定したウインドウに切り替えます。また、切り替えたウインドウ番号を[トリガーアドレス+1(16bit)]もしくは[トリガーアドレス+2(32bit)]に書き込みます。

例えば：現在のウインドウ番号は 10、トリガーアドレスは LW-0：

LW-0 内の数値が 11 になったら、EasyBuilder Pro はベースウインドウをウインドウ 11 に切り替える以外、LW-1 内の数値も 11 に変更します。

このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

ウインドウの切替に成功した場合、切り替えたウインドウ番号の書き込みアドレスは[トリガーアドレス]で設定した読み取りアドレス、変数とは関係しています。下表をご参照ください：

データフォーマット	行先ウインドウ番号読み取りアドレス (トリガーアドレス)	切り替えたウインドウ番号の書き込みアドレス
16-bit BCD	アドレス	アドレス+1
32-bit BCD	アドレス	アドレス+2
16-bit Unsigned	アドレス	アドレス+1
16-bit Signed	アドレス	アドレス+1
32-bit Unsigned	アドレス	アドレス+2
32-bit Signed	アドレス	アドレス+2
64-bit Unsigned	アドレス	アドレス 4
64-bit Signed	アドレス	アドレス+4



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。

サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

● データを PLC に書き込む(ウインドウ)

ベースウインドウを切り替えた際に、ベースウインドウの番号を[トリガーアドレス]に書き込みます。[ウインドウ番号オフセットを使用]にチェックマークを入れると、ベースウインドウの番号+ウインドウ番号のオフセットを[トリガーアドレス]に書き込みます。

● PLC コントロール(eMT, iE, XE, mTV)

本機能はトリガーレジスタ内の数値を利用し、PLC と HMI の間のデータ転送をコントロールします。データ転送の方向は 4 種類があります。下表をご参考ください。

トリガーレジスタ内の 数値	データ転送方向
1	PLC レジスタ内のデータ → HMI の RW レジスタ
2	PLC レジスタ内のデータ → HMI の LW レジスタ
3	HMI の RW レジスタ内のデータ → PLC のレジスタ
4	HMI の LW レジスタ内のデータ → PLC のレジスタ

本機能を使用すると、[トリガーアドレス]で設定された 4 個連続したレジスタ内の数値で、送信するデータ型、送信するデータ数量、データソースアドレス及び宛先アドレスなどを決定することになります。下表では各レジスタ内の数値が示している意味を説明します：

アドレス	用途	説明
[トリガーアドレス]	データ転送方向を決めます。	データ転送方向を決めます。上表に記したように、四種類あります。新しい数値が書き込まれた場合、HMI は対応する送信動作を実行し、送信し終えたら、レジスタ内の数値を 0 に設定します。
[トリガーアドレス]+1	送信したいデータの大きさ	単位：ワード(word)。
[トリガーアドレス]+2	送信過程に、データソースのオフセット	送信するデータソースの開始アドレスは: [トリガーアドレス]+4+アドレスオフセット OMRON PLC を例として、仮にこの時に設定した[トリガーアドレス]は DM-100、[トリガーアドレス+2]で即ち DM-102 内の数値は“5”的場合、データソースの開始アドレスは DM-109 になり、 $109=(100+4)+5$ です。
[トリガーアドレス]+3	送信過程に、レシピデータレジスタ(RW)或いはローカルデータレジスタ(LW)の開始アドレス	OMRON PLC を例として、仮にこの時に設定した[トリガーアドレス]は DM-100、[トリガーアドレス+3]で即ち DM-103 内の数値は“100”的場合、送信過程中操作している RW 或いは LW の開始アド

レスは RW-100、もしくは LW-100
になります。

例 1

仮に、現在は[PLC コントロール]機能を使用し、OMRON PLC 内の DM-100 から起算する 16 words のデータを HMI レジスタ RW-200 から開始するアドレスに送信したい場合、設定方法は以下のとおりです：

1. 仮に DM-10 から開始する 4 レジスタでコントロールする場合とします。まずは 1 個[PLC コントロール]オブジェクトを新規作成し、タイプを[PLC コントロール]に、読み取りアドレスを DM-10 に設定しておきます。

2. 送信するデータとアドレスのオフセットを設定します。

DM-11 を 16 に設定したら、送信するデータの大きさは 16 words と示しています。DM-12

を 86 に設定したら、データのソースアドレスは DM-100(100=10+4+86) と示しています。

DM-13 を 200 に設定したら、宛先アドレスは RW-200 になります。

3. 最後では、データ転送の方向に応じ、送信タイプを設定します。

DM-10 を 1 に設定したら、PLC レジスタ内の数値を HMI の RW レジスタに送信すると示しています。

DM-10 を 3 に設定したら、送信方向は逆になります。

● バックライトのコントロール(ライトバック)

[トリガーアドレス]の状態が OFF へ ON に変更された際に、HMI はバックライトを付け/消します。この時も[トリガーアドレス]の状態を OFF に設定します。バックライトが消された場合、ユーザーはスクリーンをタッチしたら、バックライトは再度に付けられます。

● バックライトのコントロール

[トリガーアドレス]の状態が OFF へ ON に変更した際に、HMI はバックライトを付け/消します。ただし、"ライトバック"(write back)機能を持っていないので、[トリガーアドレス]の状態を OFF に設定することはありません。

● 音声コントロール

[トリガーアドレス]の状態がトリガー条件に満たされたようになった時、予め指定した音声データを放送することができます。

状態が OFF→ON、或いは ON→OFF になる際に音声データを放送するように設定できます。または状態が変更された際に(OFF<->ON)音声データを放送するように設定できます。

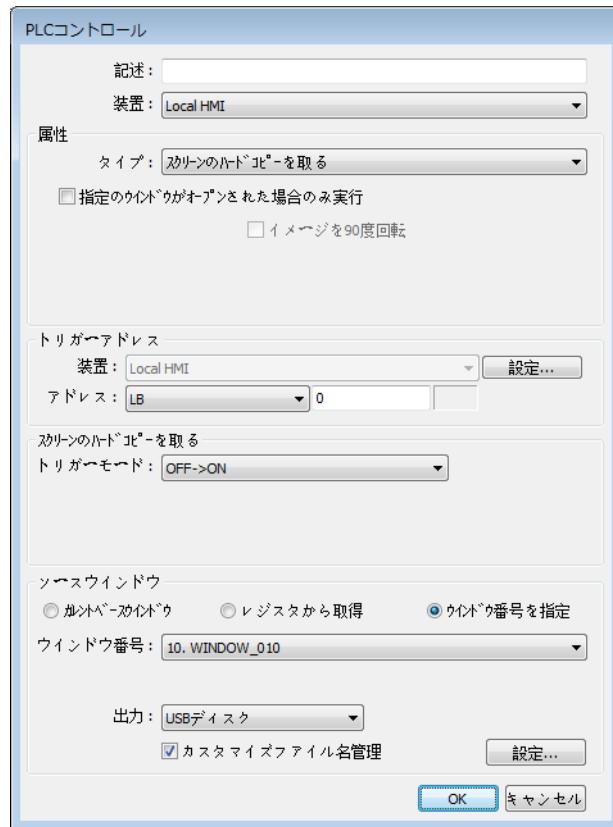
● マクロを実行する

編集完了のコマンドコマンドが既存している場合、本項が現れます。

[トリガーアドレス]の状態がトリガー条件に満たされたようになった時、指定したマクロコマンドが実行されます。

状態が OFF→ON、或いは ON→OFF になる際にマクロコマンドを実行するように設定できます。または状態が変更された際に(OFF<->ON)マクロコマンドを実行するように設定できます。或いは状態が ON にいる場合で、状態を ON に維持する間に、指定したマクロコマンドを継続に実行するように設定できます(最速 0.5 秒ごとに 1 回実行します)。

● スクリーンのハードコピーを取る



[トリガーアドレス]の状態がトリガーされた際に、指定したウィンドウ画面のハードコピーを取ります。

装置タイプは[USB ディスク]或いは[SD カード]の場合、本機能をトリガーしたら、システムは外部記憶装置に **hardcopy** フォルダを作成し、画像を保存します。画像フォーマットは **JPG** で、画像名は **yyymmdd_0000** から番号つけされます。

選択したい装置はプリンターの場合、[システムパラメータ設定/HMI 属性]タブで接続するプリンターのタイプを選択します。

選択したい装置はリモートプリンターの場合、事前に[システムパラメータ設定/プリンター/バックアップサーバー]タブで関連パラメータを設定してください。

下記三種類の方式でハードコピーを取りたいウィンドウを指定します：

カレントベースウインドウ

現在にオープンされたベースウインドウの画面のハードコピーを取ります。

ウィンドウ番号をレジスタから取得

ウィンドウ番号をワードアドレス内の数値から取ります。

ウィンドウ番号を指定

直接にハードコピーを取りたいウィンドウ番号を指定します。

カスタマイズファイル名管理

本項にチェックマークを入れると、ハードコピーを取る時の画面保存フォルダ名、またはファイル名を変更できます。

設定	記述
フォルダ名	<p>デフォルトコントロールコードや、英数字及び部分の半角文字が使用できます： !@#\$%^&()_+{}`-=;,. 動的フォーマット 本項にチェックマークを入れると、1セットのワードアドレスでフォルダ名を設定でき、それに時間ボタンのコントロールコードを入力することで、システム時刻を表示することができます。ワードの長さは1~25です。 注意:フォルダの階層は、最大10階層までです。これ以上の階層は省略されます。</p>
ファイル名	<p>デフォルトコントロールコードや、英数字及び部分の半角文字が使用できます： !@#\$%^&()_+{}`-=;,. 動的フォーマット 本項にチェックマークを入れると、1セットのワードアドレスでファイル名を設定でき、それに時間ボタンのコントロールコードを入力することで、システム時刻を表示することができます。ワードの長さは1~25です。 注意:同じ名前のファイルが既存している場合、システムは“_0001”から順番に“_9999”までに名づけします。その後にトリガーされたスクリーンのハードコピーは省略されます。 仮に“A.jpg”, “A_0001.jpg”, “A_0003.jpg”というファイル名が既存している場合、再度に同じの名前でスクリーンのハードコピーをトリガーしたら、優先に“A_0002.jpg”を生成し、そして“A_0004.jpg”, “A_0005.jpg”...のようにファイル名を生成します。</p>

Note

- 指定したウインドウはカレントウインドウではない場合、バックグラウンド印刷が提供されます。
- 背景ウインドウを指定した場合、当該ウインドウの[直接ウインドウ]或いは[間接ウインドウ]が印刷されません。
- 動的フォーマットでフォルダ或いはファイル名にサポートしていない符号を入力した場合、システムはそれの代わりにアンダーライン符号“_”を表示します。
- 動的フォーマットフォーマットを使用し、何のストリングも入力していない際にスクリーンのハードコピーをトリガーするように設定された場合、システムはデフォルトの保存先：hardcopy/yyyymmdd_0000.JPGで画像を保存します。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.28.スケジューラー

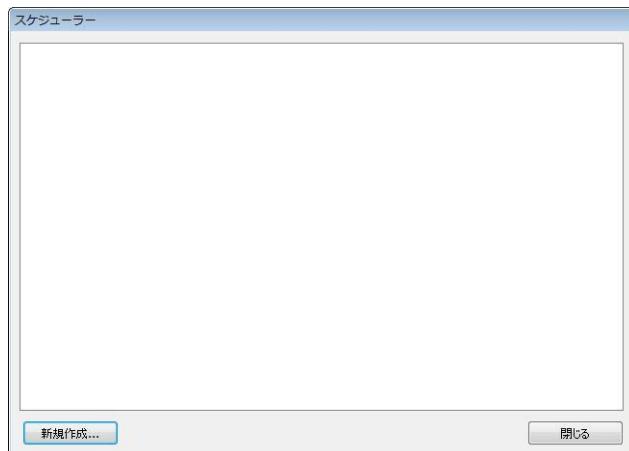
13.28.1. 概要

[スケジューラー]はビットを ON/OFF にし、またはワードアドレスに数値を書き込むことでスケジュールを設定するオブジェクトです。週間行事の企画用に適しています。

13.28.2. 設定



ツールバー上の[オブジェクト]»[オブジェクト]»[時間関連]»[スケジューラー]ボタンを押せば、ダイアログボックスがオープンされ、[新規作成]ボタンを押すと、[スケジューラー]の設定タブに入ります。

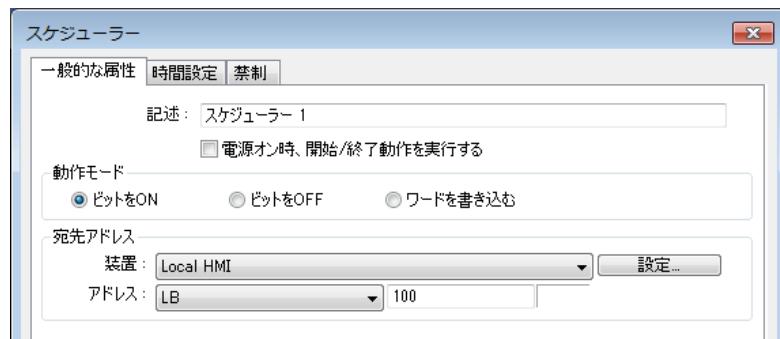


まずは 2 つの例を挙げてから、詳細に各機能を説明します：

例 1

モータ(アドレス : LB-100)は月曜日から金曜日まで運転し続け、時間は毎日午前 9 時から午後 6 時に設定したい場合、設定の手順は開始時間(朝 9 時)にてアドレス LB-100 を ON にし、終了時間(午後 6 時)にてアドレス LB-100 を OFF にします。

1. ツールバー上の[オブジェクト]»[スケジューラー]ボタンを押せば、ダイアログボックスがオープンされ、[新規作成]ボタンを押すと、[スケジューラー]の設定タブに入ることができます。
2. [一般的な属性]タブを選択し、[動作モード]で[ビットを ON にする]を選んで、そして[動作アドレス]を LB-100 に設定します。



3. [時間設定]タブを選択し、そして[定数]を選びます。

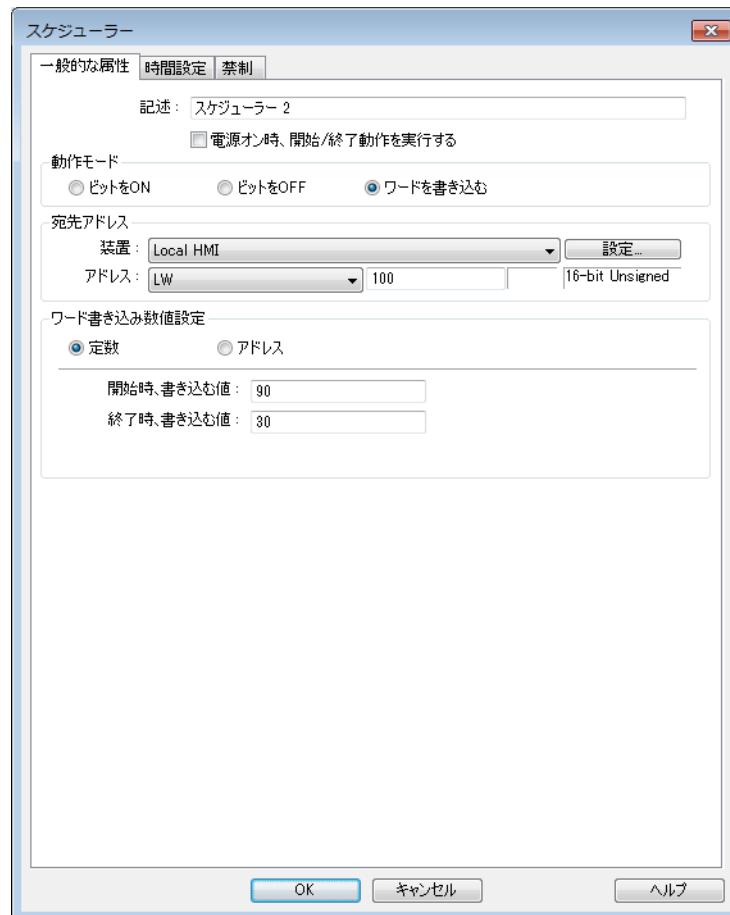


4. [開始]時間を設定します。時間を9時0分0秒に設定し、そして月曜日から金曜日までチェックマークを入れ、[單一日付に設定する]にチェックマークを入れません。
 5. [終了]時間を設定します。[終了動作を使用する]にチェックマークを入れ、終了時間を18時0分0秒に設定します。
 6. [OK]ボタンを押すと、完成したスケジュールが見られます。

例 2

月曜日から金曜日まで、開始時間8時に温度設定値90度をワードアドレスLW-100に書き込み、この時、システムは運転モードになります。終了17時に温度設定値30度をワードアドレスLW-100に書き込み、この時、システムは待機モードになります。

- ツールバー上の[オブジェクト]»[スケジューラー]ボタンを押せば、ダイアログボックスがオープンされ、[新規作成]ボタンを押すと、[スケジューラー]の設定タブに入ります。
- [一般的な属性]タブを選択し、[動作モード]で[ワードを書き込む]を選択します。[動作アドレス]でLW-100を設定します。
- [ワード書き込み数値設定]で[定数]を選択し、[開始時、書く込む値]を90に設定します。



4. [時間設定]タブを選択し、そして[定数]を選択します。
5. [開始]時間を選択します。時間を8時0分0秒に設定し、そして月曜日から金曜日までチェックマークを入れ、[單一日付に設定する]にチェックマークを入れません。
6. [終了]時間を設定します。[終了動作を使用する]にチェックマークを入れ、終了時間を17時0分0秒に設定します。
7. [一般的な属性]タブを選択し、[終了時、書き込む値]で30に設定します。
8. [OK]ボタンを押すと、完成したスケジュールが見られます。

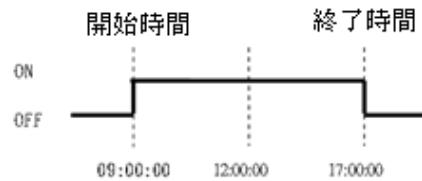
一般的な属性の設定



設定	記述
電源オン時、開始/終了動作を実行する	<p>電源オン時、設定された動作を実行します。</p> <ul style="list-style-type: none"> ● 使用する場合 <p>HMI の電源がスケジュール内でオンにされたら、開始動作が実行されます。</p> <p>HMI の電源がスケジュール外でオンにされたら、終了動作が実行されます。</p> <p>スケジュール範囲内:</p> <p>開始時間 電源ON 終了時間 開始動作 終了動作</p>
動作モード	<p>指定した時間での動作のタイプを選択します。</p> <p>ビットを ON にする</p>

スケジュール開始時に、指定したビットアドレスの状態を ON に設定し、終了時に、指定したビットアドレスの状態を OFF に設定します。

例：開始時間：09:00:00 終了時間：17:00:00



ビットを OFF にする

スケジュール開始時に、指定したビットアドレスの状態を OFF に設定し、終了時に、指定したビットアドレスの状態を ON に設定します。例：開始時間：09:00:00 終了時間：17:00:00



ワードを書き込む

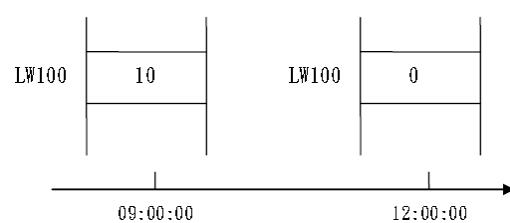
スケジュール開始時に、[開始数値]を指定したワードアドレスに書き込みます。スケジュール終了時、[終了数値]を指定したワードアドレスに書き込みます。ユーザーは直接に定数を入力し、または[アドレス]モードで数値を設定することができます。[アドレス]モードを使用すると、[コントロールアドレス]内の数値は[開始数値]、[コントロールアドレス+1]内の数値は[終了数値]になります。

例：装置アドレス：LW-100

開始時間：09:00:00 終了時間：17:00:00

定数を使用する：開始時、書き込む数値：10 終了時、書き込む数値：0

アドレスを使用する：もしコントロールアドレスを LW-n に設定した場合、LW-n で 10 を入力し、LW-(n+1) で 0 を入力します。

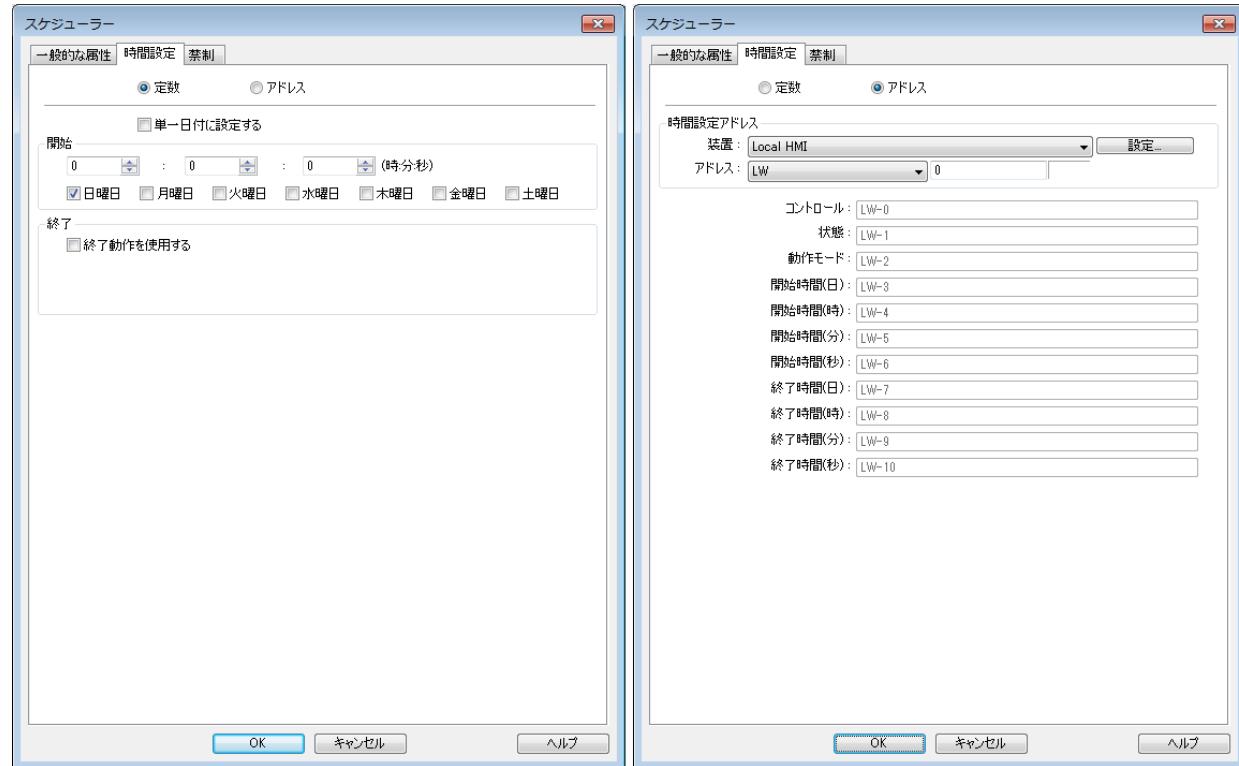


Note

- [時間設定]タブで[終了動作を使用する]にチェックマークを入れてから、[終了時、書き込む数値]を使用することができます。

時間設定

開始時間と終了時間を設定する方法を選択します。[定数]は固定した時間や日付を指定でき、[アドレス]は指定したアドレスの情報を用いて時間と日付を設定します。



- 定数

[單一日付に設定する]

使用する場合

動作は一週間での指定日付及び時間で実行されます。使用したら、全ての日付が同じの動作開始及び終了時間を設定する必要があります。

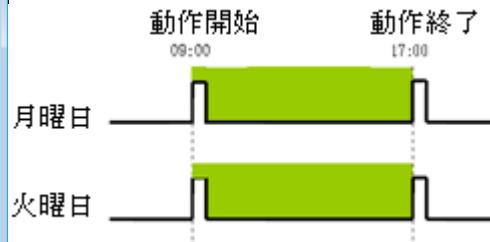


Note

- 開始時間と終了時間を入力する必要があります。
- 開始時間と終了時間の欄で全く同じの時間と日付を入力してはいけません。

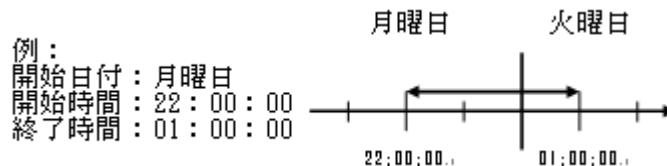
使用しない場合

スケジュール時間は一日中に限られます(開始時間から終了時間までは、24時間以内に設定しなければなりません)。



Note

- 開始時間と終了時間の欄で全く同じの時間と日付を入力してはいけません。
- こういうスケジュールは一日以内の場合のみ適しています。だから、もし入力した終了時間が開始時間より早い場合、終了動作は翌日になってから実行されます。



● アドレス

開始時間、終了時間、コマンドの実行、実行結果は全部コントロールアドレスに制御されます。ユーザーはただ[時間設定アドレス]だけを定義する必要があり、残り 11 個のワードレジスタは自動的に生成し、リストされます。図面のデータ長さが全部 16 ビットの場合を例にすれば、指定したレジスタは 32 ビットの場合、より下位の 16 ビットだけが作用するので、より上位の 16 ビットを 0 に設定してください。

以下は各アドレスの使用説明です：

コントロール(時間設定アドレス+0)

[時間更新ビット](下図参照)が ON(0→1)と検知された場合、[動作モード]、[開始時間]と[終了時間]が読み取れます。



ビット 00 : 時間更新ビット (0 : 動作無し、1 : スケジュール時間データを読み取る)

Note

- HMI は時間設定アドレスの[動作モード](アドレス+2)から[終了時間(秒)](アドレス+10)までのデータを定期的に読み取ることをしません。だから、スケジュール時間データが変わっ

た場合、是非[コントロール]の[時間更新ビット]を ON(0→1)に設定してください。

状態(時間設定アドレス+1)

[コントロール]での時間データを読み取り終えたら、HMIは[時間読み取り完了ビット]を ON(0→1)に設定します。

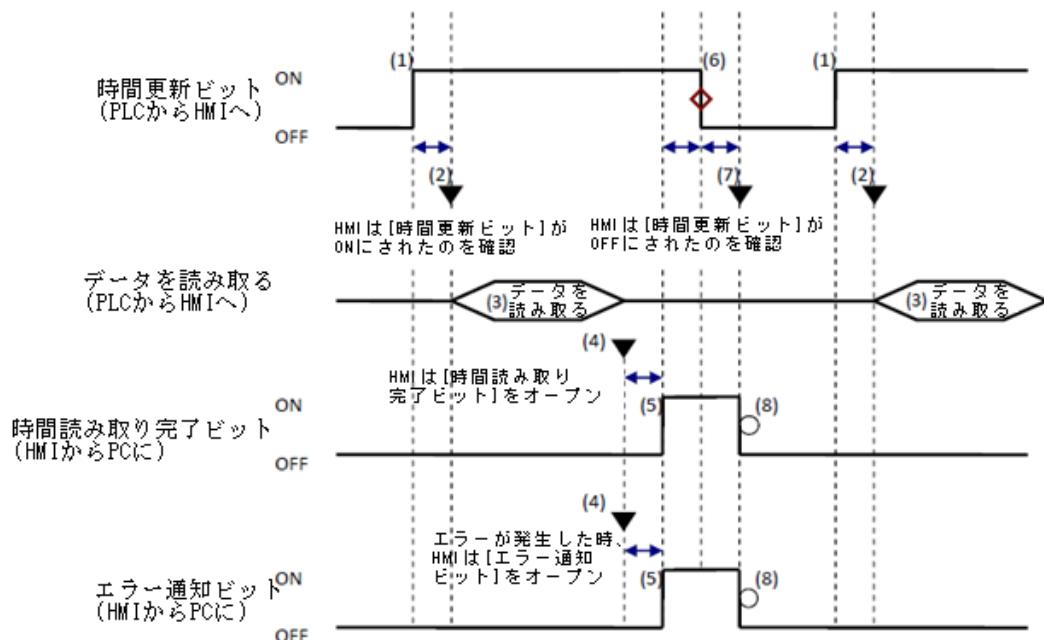
同様に、入力した時間データが不正確の場合、[エラー通知ビット]は同時に ON(0→1)に設定されます。



ビット 00 : 時間更新ビット (0 : まだ読み取っていない/読み取り中、1 : 時間の読み取りが完了)
ビット 01 : エラー通知ビット (0 : 時間データが正確に更新された、1 : 時間データにエラーがある)

Note

- 一旦[時間読み取り完了ビット]がトリガーされたと検知されたら、必ず[コントロール]の[時間ビット]を OFF(1→0)に設定してください。このビットが一旦 OFF(1→0)に設定されたら、[状態]の[時間読み取り完了ビット]及び[エラー通知ビット]も同時に OFF(1→0)に設定されます。



動作モード(時間設定アドレス+2)

[終了動作を使用する]と[单一日付に設定する]を有効/無効にする場合、[終了動作を使用する]がどのような状態にいても、全ての時間データ([時間設定アドレス]内の 11 個ワードアドレス)が読

み取られます。

15	02 01 00	ビット
確保する(0 固定)	0 0	

ビット 00：終了時間の動作設定（0：無効にする、1：有効にする）

ビット 01：單一日付指定モード（0：無効にする、1：有効にする）

Note

- もし[終了動作を使用する]に0を入力したら(無効にする)、依然に時間データを読み取るが、しかし、その内容が省略されます。
- もし[單一日付に設定する]に1を入力したら(使用する)、開始及び終了時間が入力されたかを確認してください。仮に2個以上の開始/終了日付ビットが同時にONに設定された場合、エラーが出ます。

開始/終了日付(開始日付：時間設定アドレス+3；終了日付：時間設定アドレス+7)

開始/終了動作をトリガーする日付を指定します。

15	07 06 05 04 03 02 01 00	ビット
確保する(0 固定)	Sat Fri Thu Wen Tue Mon Sun	

ビット 00：日曜日（0：無、1：指定）

ビット 01：月曜日（0：無、1：指定）

ビット 02：火曜日（0：無、1：指定）

ビット 03：水曜日（0：無、1：指定）

ビット 04：木曜日（0：無、1：指定）

ビット 05：金曜日（0：無、1：指定）

ビット 06：土曜日（0：無、1：指定）

開始/終了時間： 開始時間：時間設定アドレス+4 から+6 まで；終了時間：時間設定アドレス+8 から+10 まで)

時：0-23 分：0-59 秒：0-59

もし指定した数値が上記範囲を超えたなら、エラーが出ます。

Note

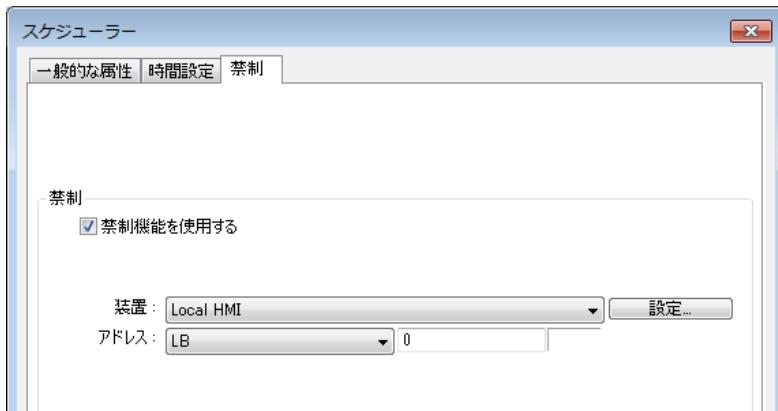
- ユーザーが入力する時間データは16ビット符号なし(unsigned)フォーマットでなければなりません。システムはBCDフォーマットの時間データを受け入れません。
- 時間設定に[アドレスモード]を使用する時、HMIを再起動した後、依然に[時間更新ビット]でスケジューラーの時間をロードする必要があります。
- RWアドレスを使用しても、HMIを再起動した後、依然に[時間更新ビット]でスケジューラーの時間をロードする必要があります。HMIを再起動した後でも前回の設定を維持するように、共有ウインドウでビット設定オブジェクトを置き、スイッチタイプを“ウインドウが

オーブンされた時、ONにする”で設定を自動的にロードさせることができます。

- 終了時間は[モード](アドレス+2)の設定に決められます。同様に、[終了時間動作設定](ビット00)の有効さは[单一日付指定モード](ビット01)の使用に決められます。

单一日付指定モード	有効にする	無効にする	
終了時間の動作設定	有効にする	有効にする	無効にする

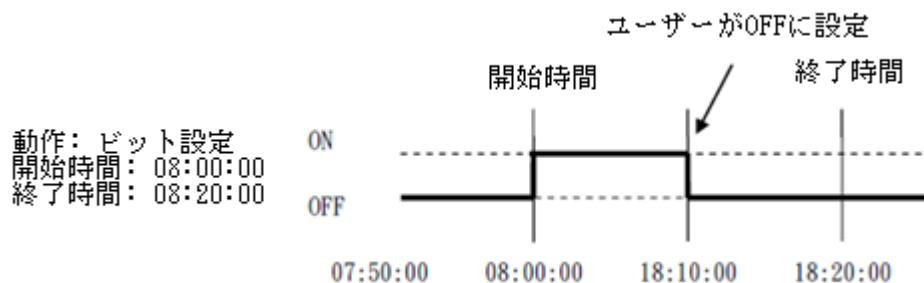
禁制タブ



使用する場合、開始動作を実行する前にHMIはこのビットの状態を読み取ります。ビット状態はONの場合、今回の開始及び終了動作を省略します。OFFの場合、設定した動作を実行します。

Note

- 最大64個の[スケジューラー]オブジェクトを登録することができます。
- 時間スケジューラーは一回だけ動作します。開始時間になると、特定の装置アドレスは一回だけ書き込まれ、この書き込む動作は重複しません。



- [開始/終了時、書き込む数値]と[禁制ビット]は開始動作実行前で一回だけ読み取ります。だから、開始動作が実行された後、[禁制ビット]の状態、或いは[終了時、書き込む値]を変更しても、終了動作の実行及び数値の書き込みを変更することができません。また、[開始/終了時、書き込む値]と[禁制ビット]データを読み取ることに応じ、開始動作はデータ通信により少し遅延されるかもしれません。
- ユーザーがHMIのRTC時刻を変更したら、システムは改めてスケジュールの開始及び終了

時間の範囲を確認します。仮に編集しているオブジェクトが新しい範囲の中に居れば、開始動作が実行されます。もし終了動作が設定されていない場合、システムは新しい範囲を確認できないので、この動作は実行されません。

- 同じの開始と終了時間が複数のスケジューラーオブジェクトに現れる場合、番号順で実行されます。
- [時間設定]を[アドレス]に指定した場合、システムは定期的に[コントロール]アドレス内のデータを読み取り、時間の長さはシステムの忙しさによります。
- [時間設定]を[アドレス]に指定し、それに指定した開始時間と終了時間が有効範囲に超えた場合、設定した時間が正確に運転しない可能性があります。注意：BCD で入力してはいけません。
- [時間設定]を[アドレス]に指定した場合、スケジューラーオブジェクトは初めてのデータ更新が成功してから、実行されます。



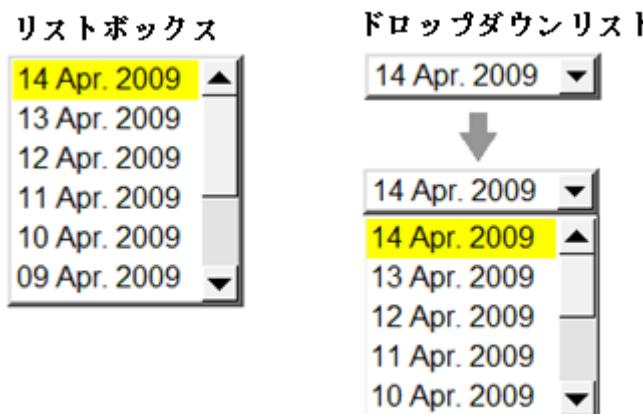
このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.29.オプションリスト

13.29.1. 概要

[オプションリスト]オブジェクトは複数の項目を表示しリスト化できるので、ユーザーはこれによって項目を閲覧、選択することができます。一旦ユーザーがある項目を選択すると、対応するデータはワードレジスタに書き込まれます。

[オプションリスト]には2種類の表示モード：[リストボックス]と[ドロップダウンリスト]があります。リストボックスは全ての項目を完全に表示し、そのうえ現在選択している項目を表示することができます。一方、一般状況下のドロップダウンリストは現在選択しているオプションのみ表示します。しかし、ユーザーがドロップダウンリストをクリックすると、システムは下記のように全ての項目(リストボックスの表示に似ている)を表示します。



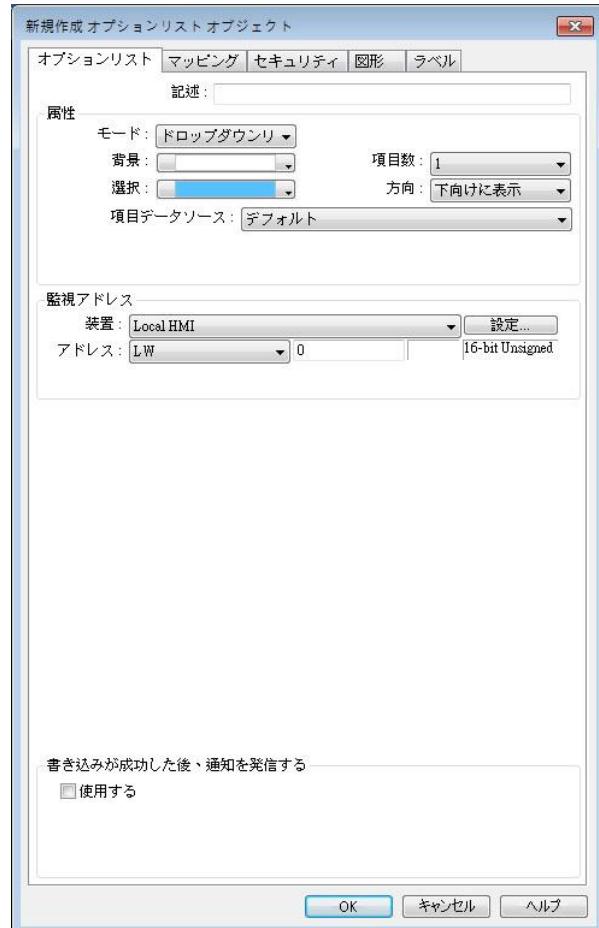
13.29.2. 設定



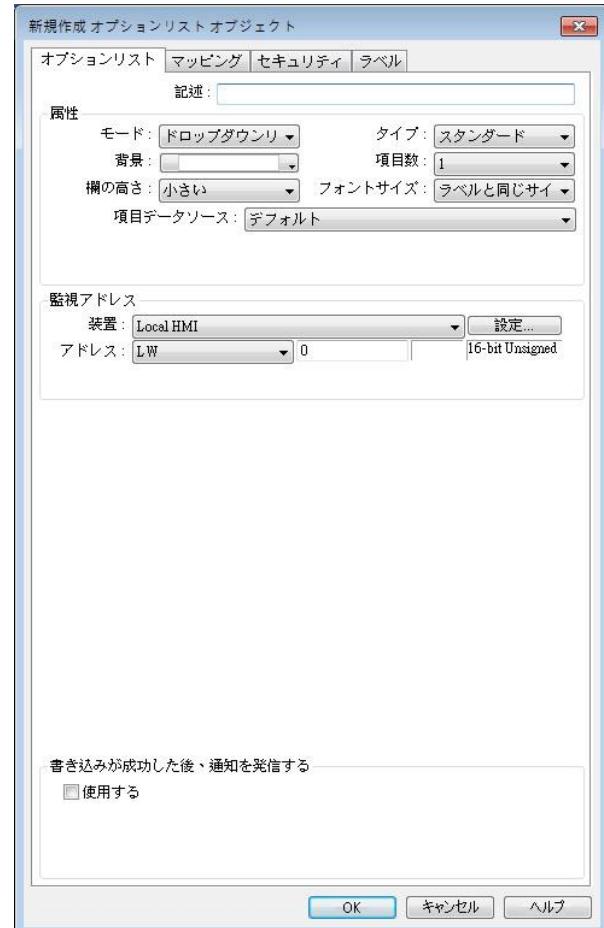
ツールバー上の[オブジェクト]»[オブジェクト]»[オプションリスト]ボタンを押せば、[オプションリスト]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[オプションリスト]オブジェクトが新規作成されます。

オプションリスト設定

eMT、iE、XE、mTV シリーズ



cMT シリーズ



設定

記述

属性

モード: [リストボックス]または[ドロップダウンリスト]が選択できます。

スタイル: 本オプションは[ドロップダウンリスト]専用のオプションです。[スタンダード]を選択すれば、オブジェクトの外見はcMTのスタイルで、[クラシック]を選択すれば、オブジェクトの外見はiE/eMT/XEのスタイルです。iE/eMT/XEのオブジェクトをcMTに変更する場合、オブジェクトのデフォルトスタイルは[クラシック]です。

項目数: 必要な項目数を設定します。1個の項目は1個の状態を示していて、ある項目が選択される場合、対応するデータを[監視アドレス]に書き込みます。

背景: 背景色を設定します。

選択: 選択した項目の背景色を設定します。

欄の高さ：ドロップダウンリスト内の項目欄の高さを設定します。

フォントサイズ：[ラベルと同じサイズ]を選択した場合、ドロップダウンリスト内のフォントサイズは[ラベル]タブ内のフォントサイズと一致にされます。[固定サイズ]を選択した場合、ドロップダウンリスト内の文字はデフォルトサイズで表示されます。

項目データソース：オプションリストが表示する内容では、四種類のモード：デフォルト、履歴データ日付、項目アドレス、ユーザー アカウントがあります。《[13.29.2.1 項目データソースの説明](#)》をご参照ください。

監視アドレス

システムは選択した項目に対応するデータを[監視アドレス]に書き込みます。

ボタンを離してからコマンドを発信

本設定を使用すると、オブジェクトを押した後、完全にボタンから離してから、指定したデータを[監視アドレス]に書き込みます。

書き込みに成功した後、通知を送信する

装置に書き込む動作に成功したら、ビットレジスタの状態をON/OFFに設定します。



Note

- cMT シリーズには[履歴データ日付]と[ボタンを離してからコマンドを発信]機能がありません。

13.29.2.1. 項目データソースの説明

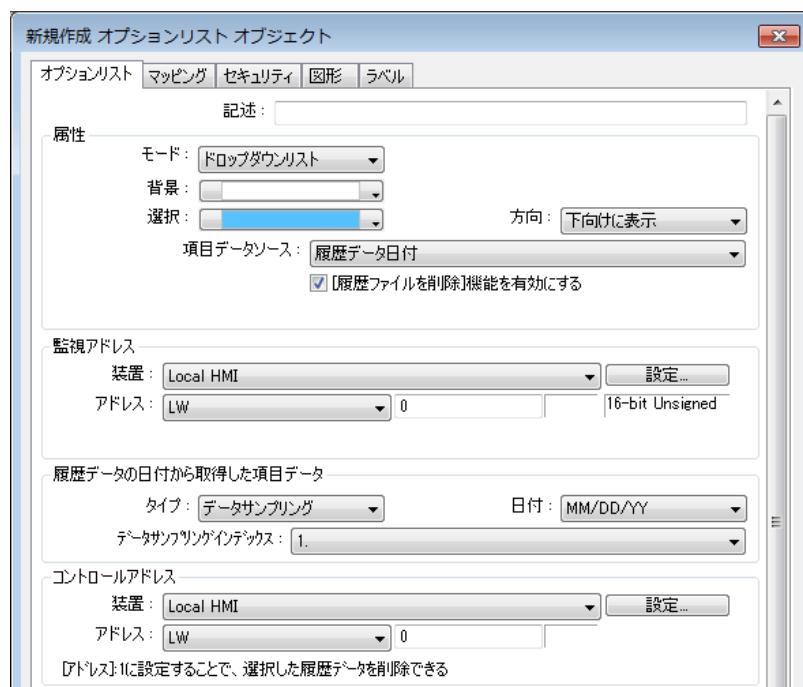
● デフォルト

ユーザーから[状態設定]タブで手動で表示させたい項目を入力します。

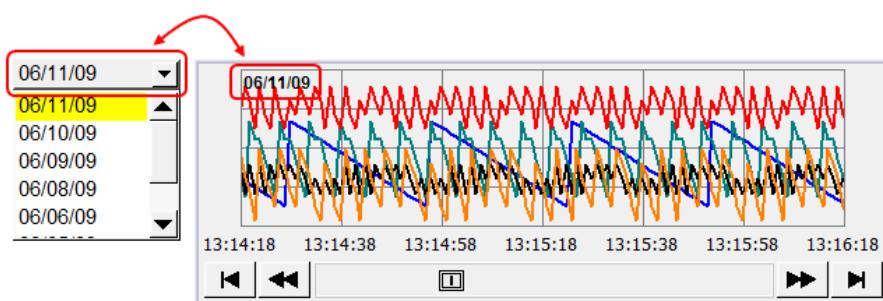
使用したい[項目数]を調整できます。1個の項目は1個の状態を示していて、リストに表示され、項目に対応するデータが[監視アドレス]に書き込まれます。

● 履歴データ日付

cMT シリーズは本機能をサポートしていません。



履歴データ表示オブジェクトを合わせて使用することができ、例えば[トレンド図]、[履歴データ表示]、[イベント表示]オブジェクトなど。オプションリストにある日付を選択したら、履歴データ表示オブジェクトはその日付を対応するデータを表示します。表示する形式は下図の通りです：



設定	記述
タイプ	[イベントログ]または[データサンプリング]が選択できます。
日付	合計 8 種類の日付モードが選べます。YYYY は四桁年(例:2012)、YY は二桁年(例 : 12)、MM は月、DD は日と示しています。
データサンプリングインデックス	[タイプ]で[データサンプリング]を選択する場合、[データサンプリングインデックス]で表示させたいデータサンプリングオブジェクトを設定する必要があります。一般的には、[トレンド図]または[履歴データ表示]での設定と一致にすればいいです。
[履歴データを削除する]機能を使用する	本機能にチェックマークを入れると、一個のコントロールアドレスを設定することができます。このコントロールアドレスを 1 に設定すれば、その日の履歴データを削除することができます。

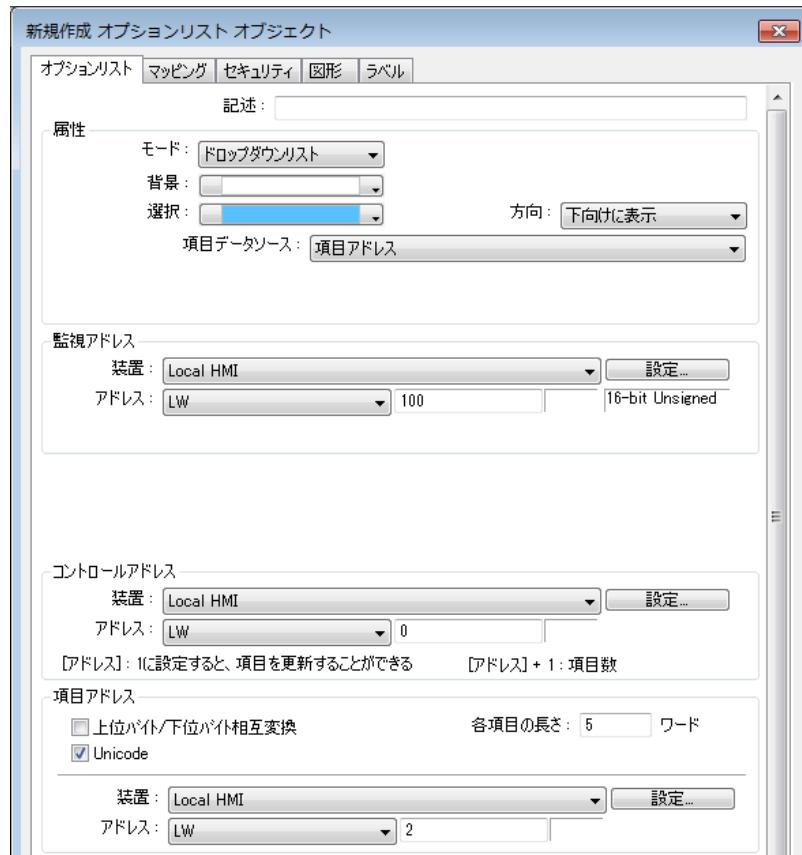
Note

- 履歴データソース(履歴データ日付)を選択する場合、システムは自動的に履歴データを読み取って、日付情報を生成するので、状態設定で再度に記入する必要がありません。
- 状態設定タブで、エラー状態に入る時に、オプションリストが表示するデータを設定することができます。

項目	数値	項目データ
0 (error)		Error !!!

● 項目アドレスモード

[項目アドレス]モードは[項目アドレス]の文字をロードし、その文字をオプションリストに表示することができます。[項目アドレス]モードを選択した後、下方では下図のように[コントロールアドレス]と[項目アドレス]が現れます。



設定	記述
コントロール アドレス	<p>[アドレス]: このアドレスが指定したレジスタ内の数値を 1 に設定したら、オブジェクトが表示する項目は[項目アドレス]の内容になります。更新完了後、レジスタ内の数値は 0 に復元します。</p> <p>[アドレス]+1: このアドレス内の数値は項目の数量を決めるために用いられます。</p>
項目アドレス	ここで項目内容を保存するための開始アドレスを設定します。
Unicode	項目内容は Unicode 文字(例：中国語)。
各項目の文字の長さ	項目の文字の長さを設定します、単位：ワード

Note

- 使用する Unicode 文字は事前に[文字]オブジェクトで入力しておいてから、EasyBuilder Pro は予め所要するフォントをコンパイルし、ダウンロードした際にフォントと一緒に HMI に

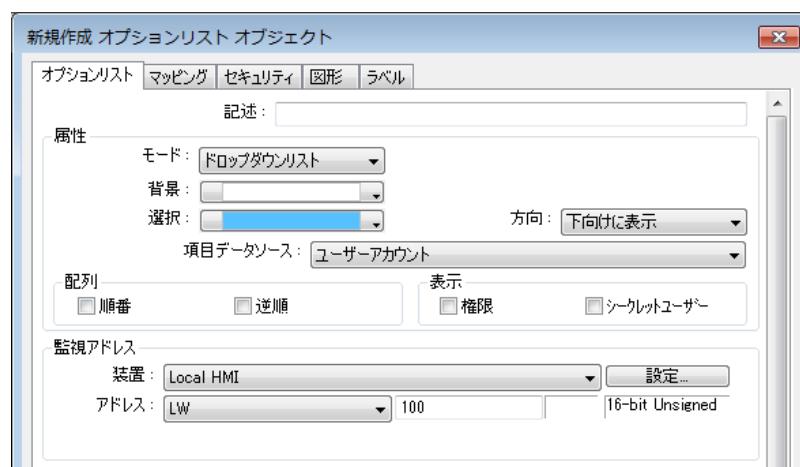
保存します。そうした場合のみ、Unicode 文字が正確に表示されます。

- (項目数)×[各項目の文字の長さ]は 4096 を超えてはいけません。
- [項目アドレス]モードを選択したら、[状態設定]タブが自動的に取り消されます。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

● ユーザーアカウント

[高度セキュリティモード]を使用すれば、[項目データソース]には[ユーザーアカウント]が現れます。この時のオプションリストにはユーザー名が表示されます。



設定	記述
配列	ユーザーアカウントの並び方を決めます。
表示	[権限]にチェックマークを入れれば、各ユーザーの使用権限が表示されます。[ユーザーを隠す]にチェックマークを入れると、隠しているユーザー情報が表示されます。

Note

- ユーザーインデックスのアドレスは[高度セキュリティモード] » [ユーザーパスワード] » [高度セキュリティモード]の中の[コントロールアドレス+2(LW-n+2)]です。

状態設定

このタブは全ての状態の項目、文字や数値を表示します。項目数を変更したかったら、[オプションリスト]»[属性]»[項目数]で選択してください。



設定

記述

項目

現在使用中の項目を全部表示します。1個の項目は1個の状態と表示し、リストに表示します。本欄は読み取り専用です。

数値

ユーザーは各項目に数値を設定することができます。下記をご注意ください：

監視アドレスを読み取る：[監視アドレス]の数値が変更されたと検出した時に、オブジェクトは一番目のマッチした項目を選択します。マッチしている項目がなかったら、エラー状態になります。エラー通知ビットが設定された場合、ビットがトリガーされます。

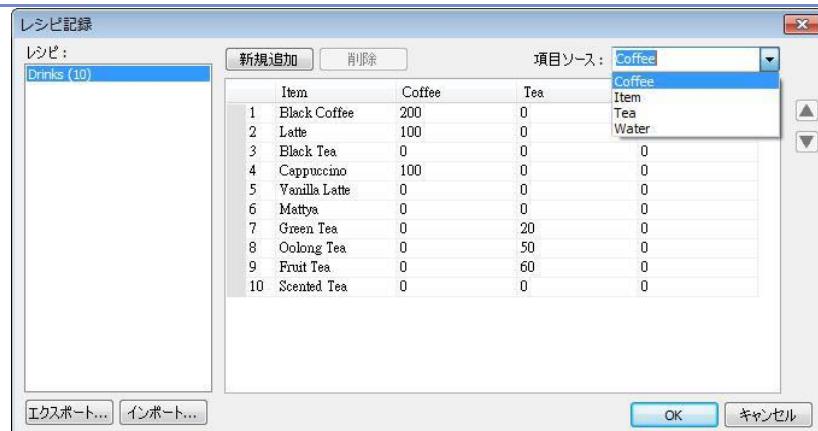
監視アドレスに書き込む：ユーザーが項目を選択したら、システムは数値を[監視アドレス]に書く込みます。

項目データ

ユーザーは各項目に表示させたい文字を設定することができます。オプションリストオブジェクトはユーザーが閲覧及び選択するため、全ての項目の文字をリストに表示します。

レシピ記録から項目データをインポート

オプションリストの[監視アドレス]に Recipe-Selection を使用する場合のみ、本機能が使用できます。[レシピ記録から項目データをインポート]をクリックすると、[レシピ記録]設定タブが開き、[項目データソース]にデータソースを選択したら、その欄にある全部のデータは自動的に[オプションリスト]にインポートします。



インポートする時、レシピデータの数によって自動的に[オプションリスト]オブジェクトの[項目数]を調整します。

インポートした後、もし[レシピ記録]オブジェクトの内容が変更されてもオプションリストには影響がありません。例えば、[レシピ記録]でレシピデータを変更する場合、インポートされた[オプションリスト]の項目データはそれに応じて変更することはありません。

エラー状態

エラー状態の文字はただ[ドロップダウン]モードのみに適用し、[リストボックス]モードでは使用できません。

エラー状態発生時、[リストボックス]モードの場合、何の項目も選択しない形でエラーを表現します。[ドロップダウンリスト]モードの場合、エラー状態を代表する文字を表示します。

例えば、[項目数]が8と設定している場合、項目番号8はエラー状態を示しています(一番目の項目は番号0なので)。

デフォルトに設定

全ての項目データをデフォルトに復元します。例：項目0の数値を0に復元し、項目1の数値を1に復元...などです。

エラー通知

オプションリストは不正な数値が書き込まれたと検知した場合、警告として指定レジスタの状態を[ON]または[OFF]にトリガーします。また、ユーザーも他のオブジェクト、例えば[イベントログ]、[アラームバー]、[ポップアップウィンドウ]を用いてエラー警告を提示することができます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.30. タイマー

13.30.1. 概要

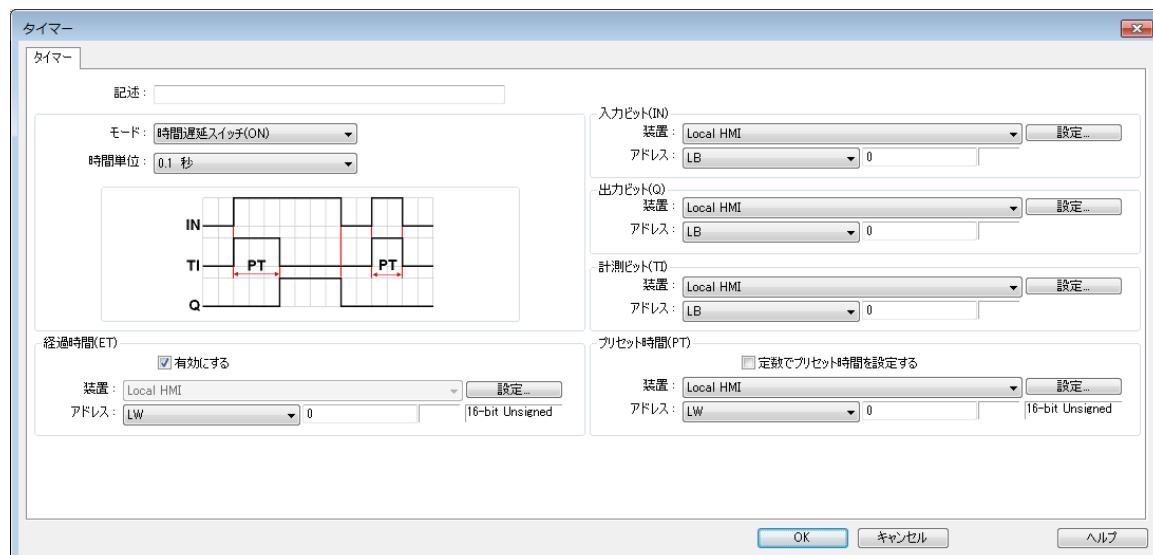
タイマーは1個の時間を計測するスイッチに相当しています。時間遅延スイッチ、パルススイッチ、及び累加式時間遅延スイッチとして使われます。その中、下記6点の変数を含めています：

タイマー変数	変数型	記述
入力ビット(IN)	ビット	タイマーのメインスイッチ
計量ビット(TI)	ビット	時間を計測し始めた際にONに設定する
出力ビット(Q)	ビット	時間を計測し終えた際に関連設定を起動する
プリセット時間(PT)	ワード	タイマーの時間数値を設定する
経過時間(ET)	ワード	経過時間を表示する
リセットビット(R)	ビット	経過時間(ET)をクリアする

13.30.2. 設定



[オブジェクト] » [時間関連] » [タイマー]アイコンをクリックし、下記に示す通り[タイマー]オブジェクト属性ダイアログボックスがオープンされます。

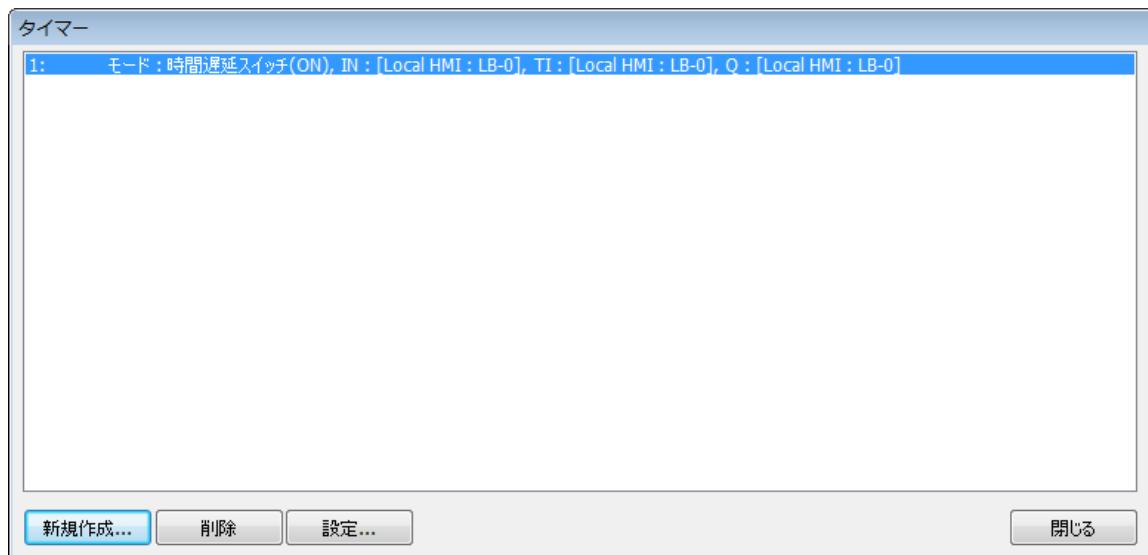


Note

- [定数でプリセット時間を設定する]はcMTシリーズのみに適用しています。
cMTシリーズを使用する場合、[タイマー]アイコンをクリックすると、先にタイマー管理ウイ

ンドウがオープンされます。

[新規作成]をクリックし、[タイマー]オブジェクトを新規作成します。



● 時間遅延スイッチ(ON)

電位図	レジスタ
	入力ビット(IN) : タイマーのメインスイッチ。 計量ビット(TI) : 時間を計測し始めた際に ON に設定する。 出力ビット(Q) : 時間を計測し終えた際に ON に設定する。 プリセット時間(PT) : タイマーの時間数値を設定する。 経過時間(ET) : 経過時間を表示する。
説明(上図参照)	
<p>期間 1：入力ビット IN を ON に設定した際に、計量ビット TI が ON になり、経過時間 ET が時間を計測し始め、出力ビット Q が OFF に維持します。</p> <p>期間 2：経過時間 ET がプリセット時間 PT に等しくなった時、計量ビット TI が OFF になり、同時に出力ビット Q が ON になります。</p> <p>期間 3：入力ビット IN を OFF に設定した際に、出力ビット Q が OFF になり、同時に経過時間 ET がクリアされます。</p> <p>期間 4：入力ビット IN を ON に設定した際に、計量ビット TI が ON になり、経過時間 ET が時間を計測し始め、出力ビット Q が OFF に維持します。</p> <p>期間 5：経過時間 ET がプリセット時間 PT に等しくなる前、入力ビット IN を OFF に設定し、計量ビット TI が OFF になり、同時に出力ビット Q が ON になります。 ET が依然に PT より小さいので、出力ビット Q が OFF に維持します。</p>	

● 時間遅延スイッチ(OFF)

電位図	レジスタ
<p>IN TI Q</p> <p>1 2 3 4 5 6</p> <p>PT</p>	入力ビット(IN) : タイマーのメインスイッチ。 計量ビット(TI) : 時間を計測し始めた際に ON に設定する。 出力ビット(Q) : 時間を計測し終えた際に ON に設定する。 プリセット時間(PT) : タイマーの時間数値を設定する。 経過時間(ET) : 経過時間を表示する。
説明(上図参照)	
期間 1 : 入力ビット IN が ON に設定した際に、計量ビット TI が OFF に維持し、出力ビット Q が ON になり、経過時間 ET がクリアされます。	
期間 2 : 入力ビット IN が OFF に設定した際に、計量ビット TI が ON になり、出力 Q が ON に維持し、経過時間 ET が時間を計測し始めます。	
期間 3 : 経過時間 ET がプリセット時間 PT に等しくなった時、出力ビット Q と計量ビット TI が OFF されます。	
期間 4 : 入力ビット IN が ON に設定した際に、計量ビット TI が OFF に維持し、出力ビット Q が ON になり、経過時間がクリアされます。	
期間 5 : 入力ビット IN が OFF に設定した際に、計量ビット TI が ON になり、出力ビット Q が ON に維持し、経過時間 ET が時間を計測し始めます。	
期間 6 : 経過時間 ET がプリセット時間 PT の数値に達する前に、入力ビット IN を ON に設定し、計量ビット TI が OFF にされ、同時に出力ビット Q が ON に維持し、経過時間 ET がクリアされます。	

● パルススイッチ

電位図	レジスタ
<p>IN TI Q</p> <p>1 2 3 4</p> <p>PT</p>	入力ビット(IN) : タイマーのメインスイッチ。 計量ビット(TI) : 時間を計測し始めた際に ON に設定する。 出力ビット(Q) : 時間を計測し終えた際に ON に設定する。 プリセット時間(PT) : タイマーの時間数値を設定する。 経過時間(ET) : 経過時間を表示する。
説明(上図参照)	

期間 1 : 入力ビット IN を ON に設定した際に、計量ビット TI と出力ビット Q が同時に ON され、経過時間 ET が時間を計測し始めます。

期間 2 : 経過時間 ET がプリセット時間 PT に等しくなった時、出力ビット Q と計量ビット TI が同時に OFF されます。(時間計測の同時に入力ビット IN を OFF に設定したので、経過時間 ET が自動的にクリアされます。)

期間 3 : 入力ビット IN が ON に設定した際に、計量ビット TI と出力ビット Q が同時に ON され、経過時間 ET が時間を計測し始めます。

期間 4 : 経過時間 ET がプリセット時間 PT に等しくなった時、出力ビット Q と計量ビット TI が同時に OFF されます。

● 累加式時間遅延スイッチ(ON)

電位図	レジスタ
	<p>入力ビット(IN) : タイマーのメインスイッチ。</p> <p>計量ビット(TI) : 時間を計測し始めた際に ON に設定する。</p> <p>出力ビット(Q) : 時間を計測し終えた際に ON に設定する。</p> <p>プリセット時間(PT) : タイマーの時間数値を設定する。</p> <p>経過時間(ET) : 経過時間を表示する。</p> <p>リセットビット(R) : 現在の経過時間(ET)をクリアする。</p>

説明(上図参照)

期間 1 : 入力ビット IN を ON に設定した際に、計量ビット TI が ON になり、経過時間 ET が時間を計測し始め、出力ビット Q が OFF に維持します。

期間 2 : 入力ビット IN を OFF に設定した際に、経過時間 ET がまだプリセット時間 PT に達していない場合、計量ビット TI が OFF され、同時に出力ビット Q が OFF に維持します。経過時間 ET が現在の状態数値を維持します。

期間 3 : 入力ビット IN を再び ON に設定した際に、計量ビット TI が ON になり、同時に経過時間 ET が再び先の状態数値から時間計測し始め、同時に出力ビット Q が OFF に維持します。

期間 4 : 経過時間 ET がプリセット時間 PT に等しくなった時、計量ビット TI が OFF になり、同時に出力ビット Q が ON されます。

期間 5 : 入力ビット IN を OFF に設定し、同時に出力ビット Q も OFF されます。(この場合、リセットビットを ON にし、経過時間 ET をクリアした後に再び OFF に設定します。)

● 累加式時間遅延スイッチ(OFF)

電位図	レジスタ
<p>IN TI R Q 1 2 3 4 5 6 7 8 9 10</p>	入力ビット(IN) : タイマーのメインスイッチ。 計量ビット(TI) : 時間を計測し始めた際に ON に設定する。 出力ビット(Q) : 時間を計測し終えた際に ON に設定する。 プリセット時間(PT) : タイマーの時間数値を設定する。 経過時間(ET) : 経過時間を表示する。 リセットビット(R) : 現在の経過時間(ET)をクリアする。
説明(上図参照)	
<p>期間 1: 入力ビット IN を ON に設定した際に、計量ビット TI が OFF に維持し、同時に出力ビット Q が ON になります。</p> <p>期間 2: 入力ビット IN を OFF に設定した際に、計量ビット TI が ON になり、同時に出力 Q が ON に維持します。経過時間 ET が時間を計測し始めます。</p> <p>期間 3: 入力ビット IN が再度に ON に設定された時、計量ビット TI と出力ビット Q が ON に維持し、同時に経過 ET が時間計測を一時停止します。</p> <p>期間 4: 入力ビット IN が再び OFF に設定された時、経過時間 ET が再び先の状態数値から時間を計測し始めます。</p> <p>期間 5: 経過時間 ET がプリセット時間 PT に等しくなった時、計量ビット TI と出力ビット Q は同時に OFF されます。(この場合、リセットビットを ON にし、経過時間 ET をクリアした後に再び OFF に設定します。)</p>	



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.31.映像入力

13.31.1. 概要

HMI は映像入力機能を提供しています。ユーザーが監視カメラを設置したら、監視カメラを通し、即時に現場の状況を監視することができます。それに、画面を記憶装置に保存し、PC で分析することができます。

本機能は各分野でも応用でき、現場の状況を監視する以外、車両や建物にも応用できます。

各機種が対応する映像入力及び画像の取得方式は下表に示したとおりです。

機種		iP	iE	eMT	mTV	XE	cMT-SVR	cMT-FHD cMT-HDM	cMT All-in-one			
映像入力	ネットワークカ メラ	N/A	N/A	Y	N/A	Y	N/A	Y	Y			
	USB カメラ	N/A	N/A	Y	Y	Y	N/A	Y	Y			
	アナログ映像入 力	N/A	N/A	eMT3121A eMT3151A	N/A	N/A	N/A	N/A	cMT3151			
機種		非 cMT 機種 – 画像取得					cMT シリーズ – 録画					
画像取得	ネットワークカ メラ	アドレスがトリガーされた後、当時の画像を撮る					アドレスがトリガーされた後、指 定時間で録画してビデオを作成 し、指定した外部記憶装置に保存 する					
	USB カメラ											
	アナログ映像入 力	アドレスがトリガーされた後、設定に応じてトリガー 前後の複数の画像を撮る										

cMT シリーズ HMI を使用する場合、cMT Viewer では本オブジェクトが見られません。

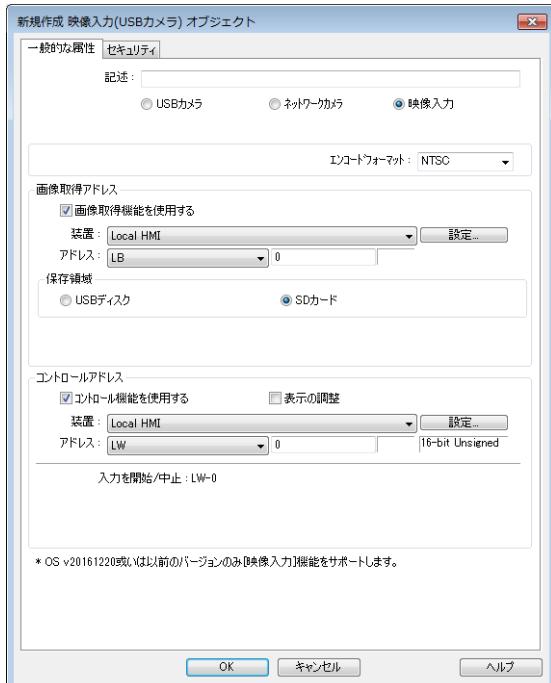
13.31.2. 設定



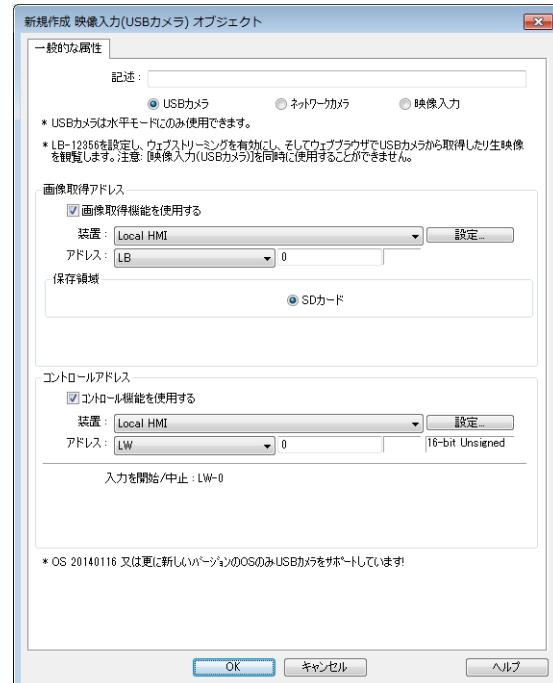
ツールバー上の[オブジェクト] » [映像入力]ボタンを押せば、[映像入力]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[映像入力]オブジェクトが新規作成されます。

一般的な属性の設定

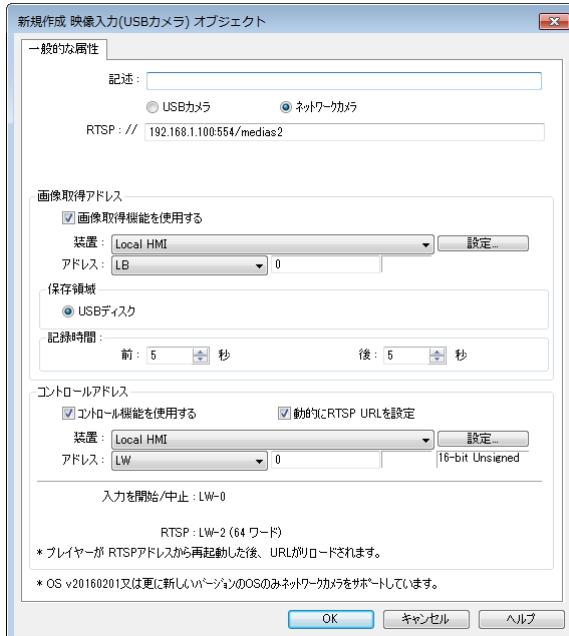
アナログ映像入力



USB カメラ映像入力



ネットワークカメラ映像入力



設定

入力チャンネル
ル

エンコードフ
オーマット

記述

映像入力チャンネル1、または映像入力チャンネル2を選択します。(ア
ナログ入力限定)

NTSC または PAL を選択します。(アナログ入力限定)

RTSP

ネットワークカメラの RTSP アドレスです。

ネットワークカメラにアクセス権限のアカウントとパスワードが設置されていたら、アカウントとパスワードを RTSP アドレスに組み込むことができます。

例えば:

`admin:admin@192.168.1.119:554/cam/realmonitor?channel=1&subtype=0`

[TCP を通してストリーミング]

本項にチェックマークを入れると、TCP を通してのストリーミングをサポートします。

画像取得アドレス

[画像取得機能]にチェックマークを入れ、入力映像の画面を取得する機能を設定します。cMT 以外のシリーズは画像を取得(キャプチャ)することになりますが、cMT シリーズはトリガー前後指定時間の avi ビデオを録画します。

画像取得アドレス

画像を取得するようにトリガーするコントロールアドレスを設定します。

保存場所

取得した画像の保存先を設定します。

フォーマット	保存装置
アナログフォーマット	画像を SD カードまたは USB ディスクに保存するのを選択します。入力チャンネル 1 の画像は外部装置の VIP1 フォルダに保存され、チャンネル 2 は VIP2 フォルダに保存されます。
USB カメラ	SD カードのみに保存されます。
ネットワークカメラ	XE シリーズは USB ディスクのみに保存されます。 eMT シリーズは SD カードのみに保存されます。

記録時間(cMT シリーズ)

アドレスがトリガーされた後、トリガー前・後の指定した時間でのビデオを録画し、指定した保存先に保存します。注意すべきなのは、録画が完了された後、HMI は少し時間をかけてビデオを圧縮します。圧縮にかかる時間の長さ HMI の CPU パフォーマンスによって異なります。録画及び圧縮している間、オブジェクトの右下隅には赤い点が表示され、それは録画中または圧縮中と示しています。圧縮過程中、次のビデオの録画ができません。

記録時間(eMT、iE、XE、mTV シリーズ)

画面を取得する時間の範囲を設定します。

フォーマット	保存方式
アナログ	● 最大の時間範囲は[画像取得アドレス]がト

	<p>リガーされた時の前後 10 秒です。</p> <ul style="list-style-type: none"> ● 画面を毎秒ごとに一回取得します。 ● 画面ファイルの命名規則は： <p>[画像取得アドレス]がトリガーされた前後： YYYYMMDDhhmmss.jpg</p> <p>[画像取得アドレス]がトリガーされた当時： YYYYMMDDhhmmss@.jpg</p> <p>例えば、時間を前後 5 秒を設定した場合、[画像取得アドレス]の状態が OFF から ON に変わった途端、システムはトリガー時間点から起算し、毎秒 1 枚で、前後 5 秒ですと合計 11 枚の画像が取得されます。</p>
USB カメラ	コントロールアドレスがトリガーされた当時の画像のみ取得します。画像ファイルの命名フォーマットは : YYYYMMDDhhmmss.png。
ネットワーク カメラ	コントロールアドレスがトリガーされた当時の画像のみ取得します。画像ファイルの命名フォーマットは : YYYYMMDDhhmmss.png。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

コントロール アドレス

使用したら、特定の数値を[コントロールアドレス]及び連続したレジスタに書き込めば、映像出力をコントロールすることができます。仮にコントロールアドレスを LW-n(n は任意のアドレス)に設定した場合、特定の数値を指定したアドレスに書き込むと、下表に示しているコマンドが実行されます。

アナログフォーマット映像入力制御機能

アドレス	数値	コマンド内容
LW-n		映像の再生を中止する
	1	入力チャンネル 1 の映像を再生し、スクリーンに表示する
	2	入力チャンネル 2 の映像を再生し、スクリーンに表示する
	3	入力チャンネル 1 の映像を再生するが、スクリーンに表示しない (画像取得を実行可能)

	4	入力チャンネル2の映像を再生するが、スクリーンに表示しない (画像取得を実行可能)
LW-n+1	1	映像を一時停止/再生する(cMTは本アドレスをサポートしません)
LW-n+2	1~100	コントラストを調整する(アナログ入力限定)
LW-n+3	10	明るさを調整する(アナログ入力限定)

USB カメラ/ネットワークカメラ制御機能

アドレス	数値	コマンド内容
LW-n	0	映像を一時停止する
	1	映像を再生する
LW-n+1	1	映像を一時停止/再生する(cMTは本アドレスをサポートしません)
LW-n+2	ストリング	ネットワークカメラのRTSP アドレス

- [コントロールアドレス(LW-n)]の数値を変更したら、システムは変更後の数値を確保します。
- [コントロールアドレス+1(LW-n+1)]の数値を変更したら、システムは対応コマンドを実行完了後、数値を0にクリアします。
- [コントロール機能を使用する]を使用しなかったら、システムは自動的に[入力チャンネル]が指定した映像を放送します。
- [表示を調整する]にチェックマークを入れてから、コントラスト及び明るさを調整することができます(アナログ入力限定)。

**アナログ映像入力に関して :**

- システムでのどの時間点でも、1個の映像入力チャンネルだけを使用できます。
- 画像取得機能は再生の一時停止に影響されなく、映像入力のリアルタイム画像を取得します。
- アナログ入力に推奨するフォーマット及び解像度は以下のとおりです :

	1:1	50%
NTSC	720x480	360x240
PAL	720x576	360x288



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

cMT シリーズのビデオ録画(画像取得機能)：

- 録画が完了された後、HMI は時間をかけてビデオを圧縮します。圧縮にかかる時間の長さは、録画された時間の長さによります。
- HMI がビデオを圧縮している間に、新しいイベントがトリガーされたら、録画が実行されません。

USB カメラの使用に関して：

- USB カメラを使用する場合、映像再生の途中で USB カメラを外したら、再度接続しても、映像が改めてロードすることはありません。コントロールアドレスにチェックマークを入れた場合、まずはコントロールアドレスで映像のロードを中止し、改めてロードしてください。コントロールアドレスを使用していない場合、ウインドウを切り替えるか、または HMI を再起動する必要があります。
- eMT3070A モデルの映像入力オブジェクトのサイズは最大 340*240 で、eMT3105P、eMT3120A、eMT3150A、XE モデル及び mTV シリーズは最大 640*480 です。
- USB カメラを使用する場合、システムは実際の表示サイズは USB カメラがサポートしている解像度の中に、最もオブジェクトのサイズに近い解像度を選択して使用します。つまり、映像の表示サイズがちょうどオブジェクトに設計されたサイズに一致することは保証できません。オブジェクトのサイズを USB カメラがサポートする解像度に調整することがお勧めです。
- USB カメラを使用する場合、カメラの実際表示画面がウインドウを越えることを避けるため、システムは右側及び底部に 50 ピクセルを確保します。即ち、映像入力オブジェクトの右側及び底部の縁とウインドウの縁には 50 ピクセルの距離があります。
- USB カメラを使用する場合、映像入力オブジェクトの背景色は黒です。入力映像の解像度がオブジェクトが設計されたサイズより小さいなら、黒い枠が表示されます。オブジェクトのサイズを USB カメラがサポートする解像度に調整するのがお勧めです。
- 現在、映像を表示できると実証された USB カメラは Logitech C170, Logitech C310, Logitech C910, LifeCam VX-2000 です。



このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

ネットワークカメラの使用に関して：

- ネットワークカメラを使用するには、RTSP アドレスを設定する必要があります。ネットワークカメラの設定タブに RTSP アドレスが見つかり、或いはインターネット上の関連データベースで見つけることができます。
- ネットワークカメラを使用する場合、映像再生の途中で接続が中断されたら、その後で接続が繋がっても、映像は継続に再生しません。だから、プロジェクトをデザインする時に、もしコントロールアドレスを使用する場合、先にコントロールアドレスで映像の停止及び再生の再開を設定することができます。もしコントロールアドレスを使用していない場合、映像の再生を再開するには、ウインドウを切り替えるか、HMI を再起動することが必要です。
- ネットワークカメラを使用する時、画面に表示されるサイズは実際のオブジェクトのサイズです。もし映像の解像度がオブジェクトのサイズと違っていたら、映像は自動的にオブジェクトのサイズに合わせて同じになるように調整します。但し、映像が調整される際に、歪みが出る可能性があるので、オブジェクトのサイズをネットワークカメラの解像度に合わせて調整するのがお勧めです。
- ONVIF 規格対応のネットワークカメラを使用し、RTSP でストリーミングしてください。
- ウェブストリーミングの品質及び流暢さを確保するため、映像の速度が遅くなったり、CPU 負荷が高すぎて HMI の応答が遅くなったりする状況があれば、HMI が正常に稼動できるようにするため、映像出力の設定を調整してください。また、各モデルのハードウェアの仕様が異なり、及び各プロジェクトも夫々違っているので、ベストなパラメータはユーザーに任せなければなりません。
- 以下は現行、お勧めの映像規格です：

解像度	最大 960x544
ビデオ圧縮フォーマット	H.264, MJPEG
画面のリフレッシュレート	15 fps
ビット転送速度	最大 800kbps



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。



このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

13.32. システムメッセージ

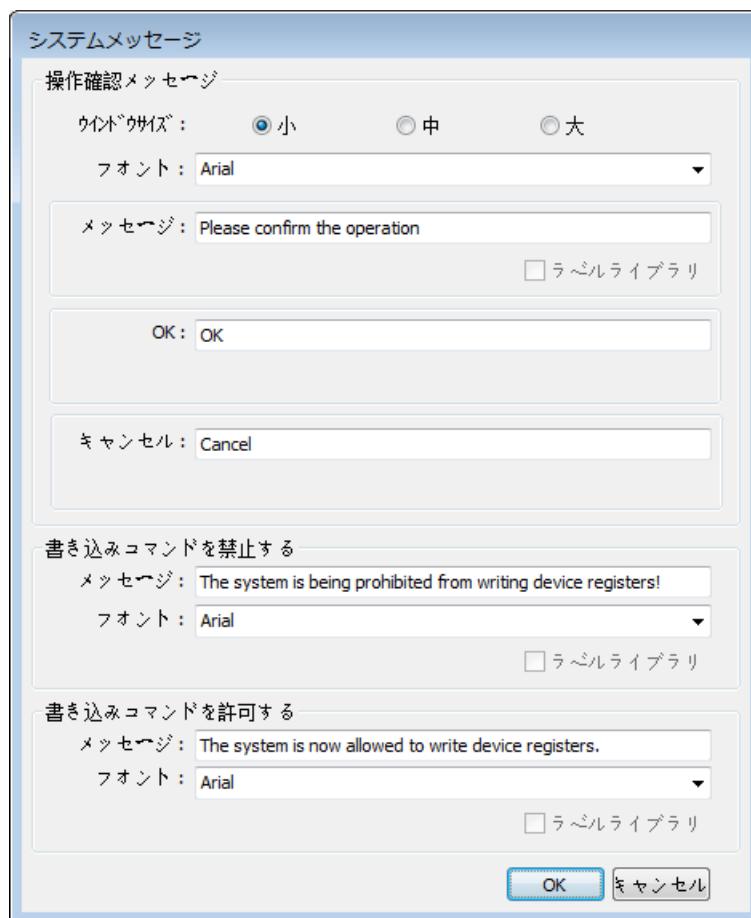
13.32.1. 概要

オブジェクトに[使用する前に確認ウインドウをポップアップする]及び[リモートログインの使用可否]を選択した場合、システムは次のメッセージウインドウをポップアップします：[操作確認メッセージ]、[書き込みコマンドを禁止する]、[書き込みコマンドを許可する]です。この3つのメッセージの内容は[システムメッセージ]の中で編集できます。

13.32.2. 設定

ツールバー上の[オブジェクト]»[システムメッセージ]ボタンを押せば、[システムメッセージ]ダイアログボックスがオープンされ、そこでシステムメッセージを設定することができます。

システムメッセージ設定



設定	記述
ウインドウサイズ	ポップアップするウインドウの大きさとフォントのサイズを決めます。

操作確認メッセージ	保護されるオブジェクトを操作する場合、メッセージを表示し、ユーザーに操作するのかを確認します。[操作確認メッセージ]の中の内容、及び[OK]、[キャンセル]の 2 ボタン上の文字を設定することができます。[OK]と[キャンセル]ボタン上の文字は同じのフォントで設定する必要があります。また、[メッセージ]でラベルライブラリを選択する場合のみ、[OK]と[キャンセル]でラベルライブラリを使用するのが許可されます。
書き込みコマンドを禁止する	システムレジスタ LB-9196(ローカル HMI では、閲覧機能のみサポートしている)が ON に設定した場合、本メッセージを表示します。
書き込みコマンドを許可する	システムレジスタ LB-9196(ローカル HMI では閲覧機能のみサポートしている)が OFF に設定した場合、本メッセージを表示します。

 Note

- cMT シリーズはダイアログウインドウサイズの設定及び LB-9196 でリモートコントロール機能を設定することをサポートしていません。

13.33. レシピ閲覧

13.33.1. 概要

[レシピ閲覧]は特定のレシピデータを閲覧することができ、ユーザーは[レシピ閲覧]でそのレシピの全ての項目や数値を閲覧することができます。

13.33.2. 設定



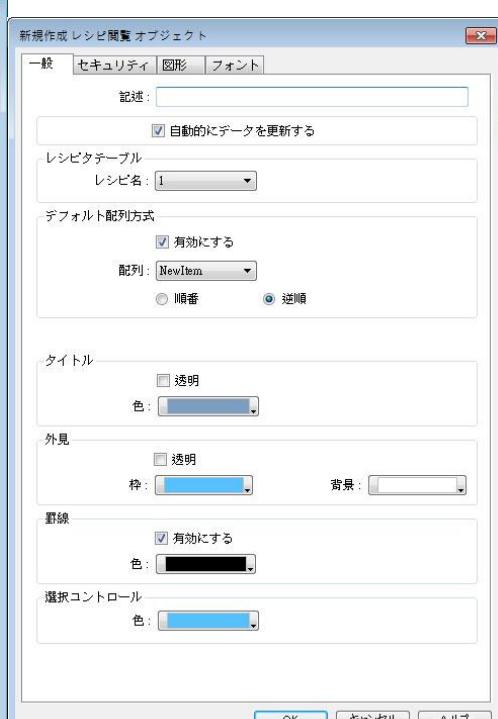
ツールバー上の[オブジェクト]»[レシピ閲覧]ボタンを押せば、[レシピ閲覧]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[レシピ閲覧]オブジェクトが新規作成されます。

一般的な属性の設定

cMT シリーズ



eMT、iE、XE、mTV シリーズ



[レシピ閲覧]各部分の名前を下図を参照してください：



設定	記述
自動的にデータを更新する	本項目にチェックマークを入れると、レシピが変更された後、システムは自動的にレシピの内容をリフレッシュします。チェックマークを入れないと、ウインドウが切り替えられた場合のみ、レシピの内容がリフレッシュされます。
レシピテーブル	<p>レシピ名 閲覧したいレシピの名前を選択します。ドロップダウンリストで他のレシピを選択することもできます。</p> <p>表示 表示したいレシピ項目欄を設定します。</p>
デフォルト配列方式	[レシピ閲覧]の項目配列方式を設定します。[順番]と[逆順]が選択できます。
スタイル (cMTにのみ適用する)	[デフォルト]、[クリスタル]、[フラット]スタイルが選べます。
表題を使用する(cMTにのみ適用する)	本項目にチェックマークを入れると、[レシピ閲覧]オブジェクトに表題が表示されます。[文字]、[フォントサイズ]、[色]、[名前]を設定できます。(選択したスタイルはクリスタルまたはフラットの場合のみ、オブジェクトの表題を設定できます。)
タイトル	<p>各項目のタイトルを設定します。[データ/履歴] » [レシピデータベース]で設定します。</p> <p>透明 本項を選択したら、タイトルには背景色がなく、[色]の項目も現れません。</p>
外見	<p>オブジェクトの枠線色及び背景色を設定します。</p> <p>透明 本項を選択したら、オブジェクトには背景色がなく、[色]の項目も現れません。</p>
罫線	<p>各行と列の罫線の色を設定します。</p> <p>透明 本項を選択したら、罫線には色がなく、[色]の項目も現れません。</p> <p>欄の幅を自動調整する 内容に基づいて欄の幅を自動的に調整します。</p>

選択コントロール**ル**

(cMT に適用しない)

選択している行に表示される色を設定します。

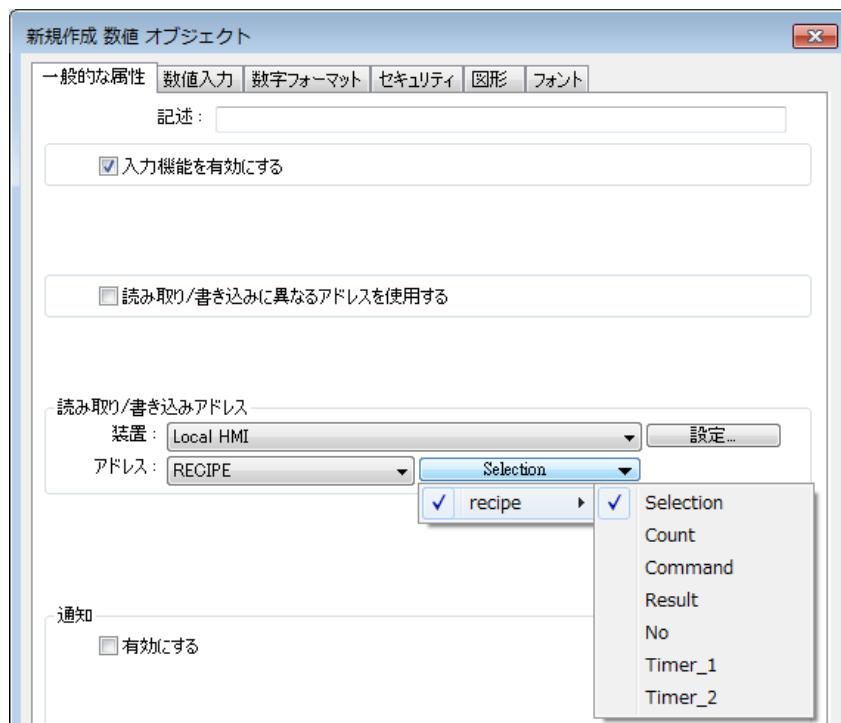
**Note**

- 四個のシステムレジスタでレシピデータベースの内容を閲覧/更新/追加/削除できます：

Selection

現在選択しているレシピの番号です。番号は 0 から計算するので、一番目のデータを選択すると、Selection の数値は 0 を表示し、このように類推しています。

[Selection] の数値が変わったら、対応のレジスタもそれによって変更されます。例えば下図の No., Timer_1, Timer_2。

**Count**

現在、レシピ内のデータ数です。

Command

特定の数値を入力したら、選択しているレシピデータに実行のコマンドを送信します。(数字はコマンドを実行する数値です)

“1”を入力：新しいレシピデータを新規追加する

“2”を入力：現在選択しているレシピデータを更新する

“3”を入力：現在選択しているレシピデータを削除する

“4”を入力：全てのレシピデータを削除する

Result

コマンドの実行結果を閲覧します。(数字はコマンドを実行した後の結果数値です)

数値“1”：コマンドの実行に成功したと示している

数値“2”：そのレシピが存在しないと示している

数値“4”：未知のコマンドであると示している

数値“8”：レシピ数が上限になり(10000 レコード)、新規追加できないと示している

 本機能を使用するには[データ/履歴] » [レシピデータベース]でレシピデータを作成する必要があります、《24 RecipeEditor》をご参照ください。

例 1

下記にて[レシピ閲覧]とレシピデータベースの簡易使用方法を見せます。

本例ではレシピデータを作成し、[レシピ閲覧]でレシピデータを閲覧、選択します。[レシピ閲覧]内の任意のデータを選択する場合、[Selection]及び対応のレジスタ数値は変わります。最後、[Command]でレシピデータベースを編集、変更することができます。

Simple Recipe View & Recipe Database

No	Name	Timer_1	Timer_2	Timer_3	Timer_4	Speed
0	Mercury	10	1	11	12	26.500
1	Venus	20	1	21	22	33.500
2	Mars	30	2	32	35	41.500
3	Jupiter	50	3	53	56	50.500
4	Saturn	80	5	85	90	60.500

System Registers:

Selection: Count: Command: Result:

Selected records: (modify here)

No.	Name:			
<input type="text" value="2"/>	<input type="text" value="Mars"/>			
Timer_1:	Timer_2:	Timer_3:	Timer_4:	Speed:
<input type="text" value="30"/>	<input type="text" value="2"/>	<input type="text" value="32"/>	<input type="text" value="35"/>	<input type="text" value="41.500"/>

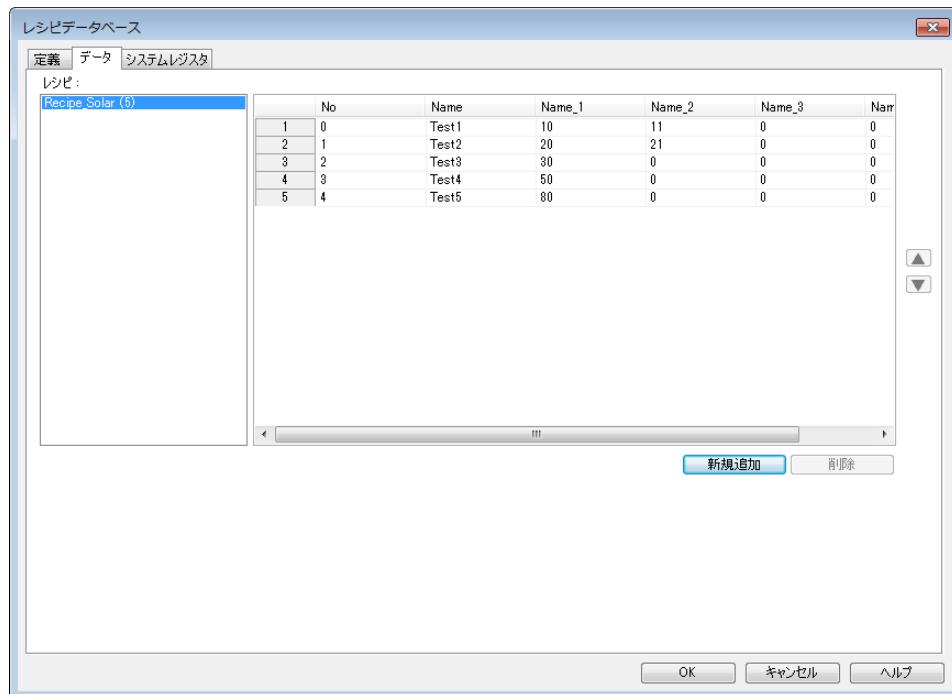
Edited by Mac

1. 下図のようにレシピを作成します：

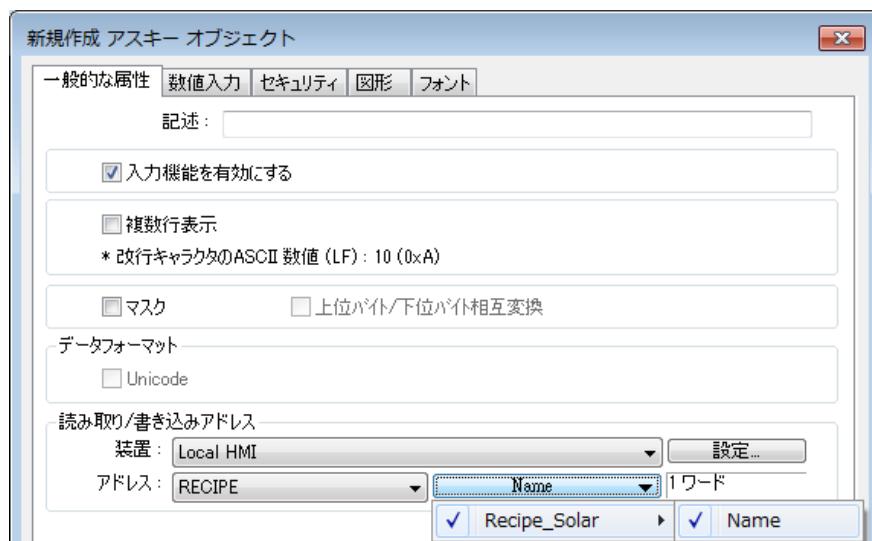


オブジェクト

2. 下図のように[データ]でデータを増加します：



3. [レシピ閲覧]オブジェクトを新規作成し、先ほど作成したレシピデータベースを使用します。
4. [Selection]、[Count]、[Command]、[Result]を使用する[数値]オブジェクトを四個作成します。
5. No., Name, Timer_1, Timer_2, ..., Timer_4, Speed 項目の対応オブジェクトを作成します。例えば、下図のように[アスキーオブジェクトを作成し、そのアドレスは“Recipe”的“Name”に当たります：



6. それで編集完了しました。
7. 画面では Mars が選択され、それに対応する Timer_1, Timer_2, ..., Speed も対応のオブジェクトに更新しました。レシピ記録には 5 レコードのデータがあるので、Count は 5 に表示します。[レシピデータ]内の他のデータを選択してみれば、対応の Name, Timer_1, ...などの情報も同時に更新されます。
8. 下記の操作をしてみてください：

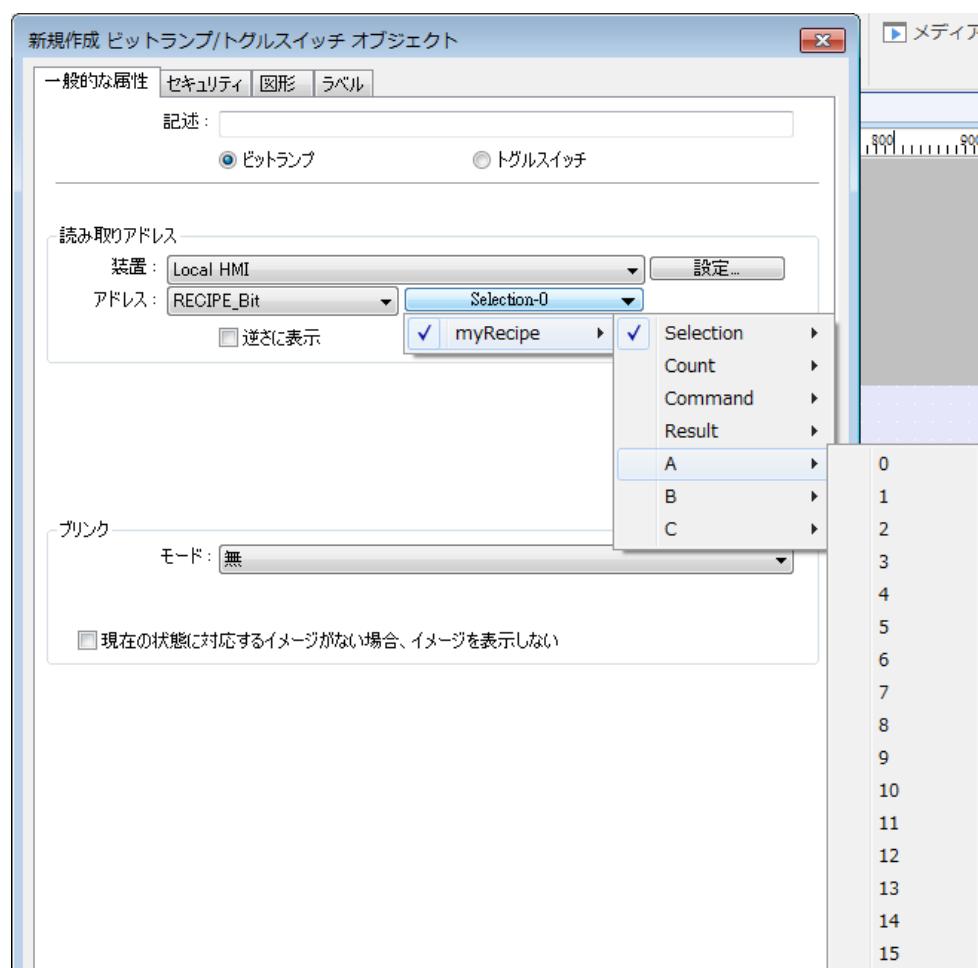
- 追加：
Command に 1 を入力し、現在入力したデータをレシピデータベースに追加します。
- 更新：
Command に 2 を入力し、[数値]及び[アスキイ]のデータをレシピデータベースに更新します。
- 削除：
Command に 3 を入力したら、現在選択しているレシピを削除することができます
- 異なる項目を配列する
[タイトル]をクリックしたら、異なる項目を配列することができます。

例 2

下記にてどうのように RECIPE_Bit を使用すれば、レシピデータのビットアドレスを読み取れるのかを実演します。

データベース自身は BOOL タイプの項目を新規追加することができませんが、ユーザーは依然に直接に 16bit, 32bit データの個別のビットデータを書き読むことができます。

下図のように、ビットオブジェクトの読み取りアドレスで RECIPE_Bit を選択し、そして書き読みたい項目に指向したら、ビットの選択リストが開かれます。そうしたら、レシピデータベースを利用し、ビットデータの記録及び書き読みができます。



13.34. フローブロック

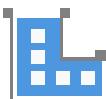
13.34.1. 概要

[フローブロック]オブジェクトはパイプ内のブロック流れ状態、または輸送ラインの状況を示すことができます。今まで[移動/回転図形]オブジェクトで流動型図形を作成する際に、ユーザー自分で正確に2点での間の位置を測定する必要があり、各線分は精確な水平または垂直的な線分で、それに流動する間隔が固定でなければなりません。cMT シリーズの場合は多角度で作画でき、水平または垂直に限られません。

下記にて[フローブロック]オブジェクトの特長を説明します：

- 各線分は垂直または水平的な直線で、それに流動する間隔が固定しています。cMT シリーズの場合は、多角度で作画でき、水平または垂直に限られません。
- 動的に流速及び流れ方向を調整することができます(流速及び方向を指定したレジスタで調整可能)。
- セキュリティ機構使用可能。指定したビットの状態でフローブロックを表示するかを決めます。

13.34.2. 設定

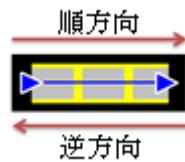


直接に[フローブロック]アイコンをクリックし、本オブジェクトを作成してください。またはツールバー上の[オブジェクト] » [画像] » [アニメーション] » [フローブロック]をクリックすることでオブジェクトを新規作成します。

一般的な属性の設定



設定	記述
逆方向	普通、フローブロックの流れ方向は作成した際の順方向で移動します(青矢印の方向)。逆方向にチェックマークを入れたら、流れ方向は逆方向になります。

**動的速度****読み取りアドレス**

フローブロックの速度及び流れ方向は指定したレジスタで調整します。使用範囲は-25~25 です。負数を入力する場合、流れ方向が逆方向になると示しています。

設定

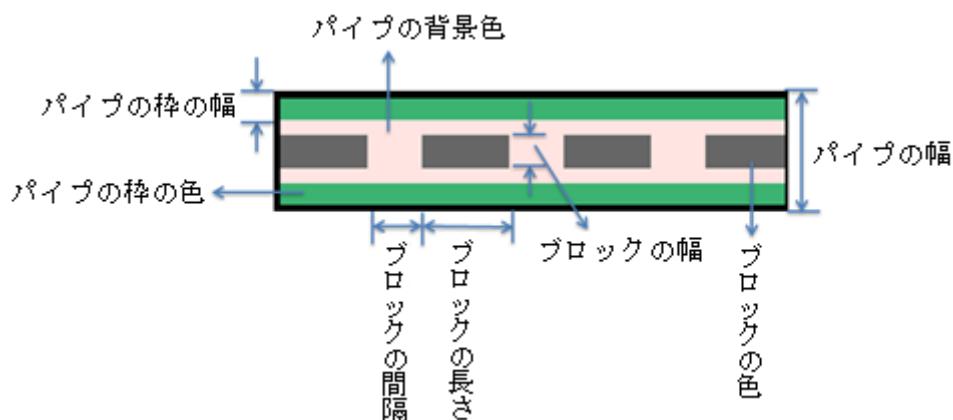
現在選択しているアドレス及び使用しているフォーマットが表示されます。それにここで[システムレジスタ]、[インデックスレジスタ]、[アドレスラベルライブラリ]を使ってアドレスを設定することができます。

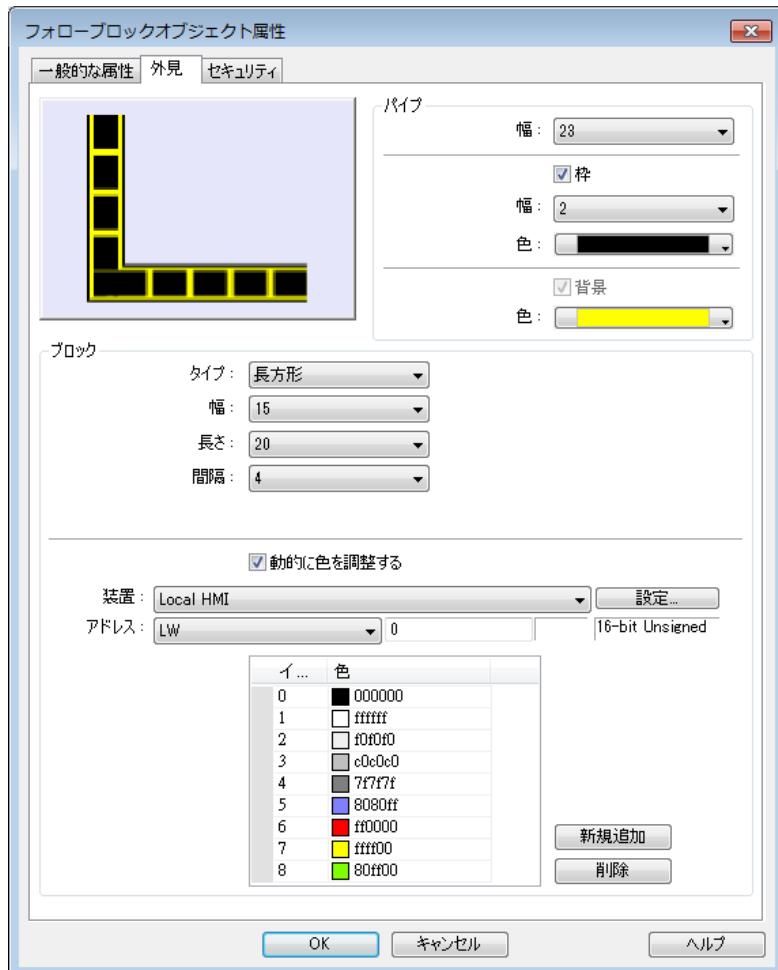
流速

25 等級の流速に分けられ、数値が大きいほど流速が速いです。[動的速度]を使用していない場合、流速の設定範囲は順方向の 0~25 のみです。

外見設定

フローブロックオブジェクトの外見を設定します。各部分の名前は、下図をご参照ください：



**設定****記述****パイプ**

パイプの属性を設定します。パイプの背景色、外枠の色及び幅を設定します。[外枠]にチェックマークを入れたら、必ず背景を使用しなければなりません。

ブロック

ブロックの中身の属性を設定します。タイプ、長さ、幅、間隔及び色を設定します。

タイプは長方形と矢印に分けられています。矢印の指向はフローブロックの流動方向になります。

長方形 : 矢印 :

動的に色を調整する

フローブロックの色を動的に調整できます。番号 0~255 と 256 個のカスタマイズ色があります。指定したアドレスを通じて、色の切り替えができます。指定したアドレスで入力した数値がカスタマイズ色の範囲上限を超えたたら、最大番号の色が表示されます。

Note

- [一般的な属性]設定タブで同時に[逆方向]及び[動的速度]にチェックマークを入れる場合、[動的速度]のコントロールアドレスに負数を入力すると、フローブロックの流れ方向は順方

向になります。

- フローブロックのタイプに[矢印]を選択し、それに[動的速度]機能を合わせて使用する場合、指定したアドレスに初期速度を設定していなかったら、矢印が表示されません。
- フローブロックオブジェクトを作画する際に、曲がり角の重合する面積が大きすぎないように、曲がり角を描く場合には最小幅が設定されています。その最小幅は図 34.1 の十字の下の図形の通りになります。図 34.2 での各曲がり角は全部最小幅を使用し作画しました。



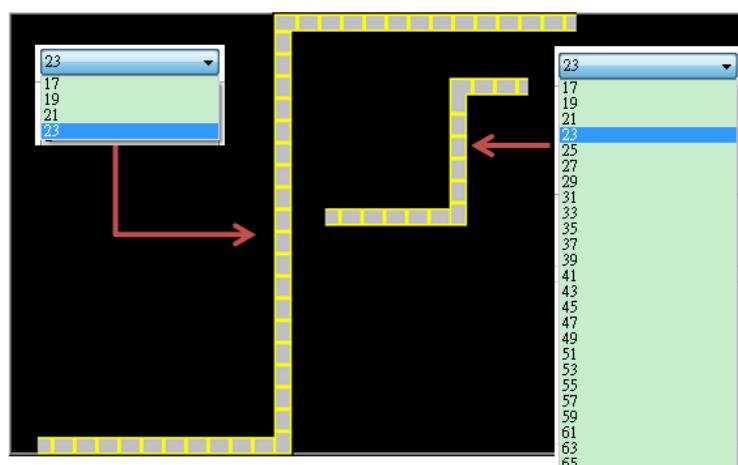
(図 34.1)



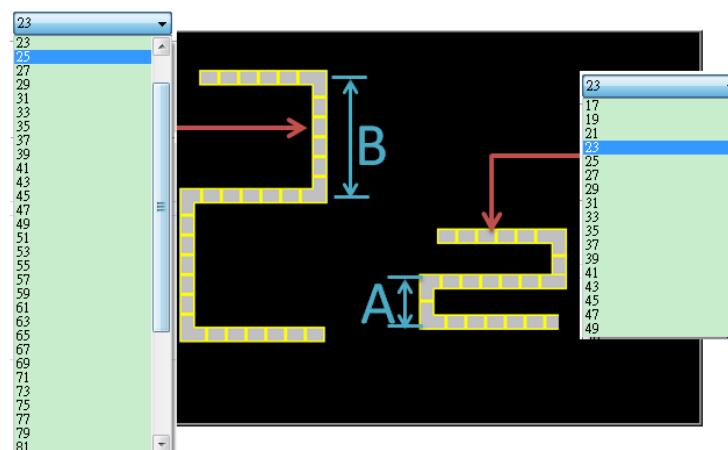
(図 34.2)

フローブロックの長さ、幅、高さの設定範囲は作画されたオブジェクトとウインドウの間の間隔とウインドウサイズによって変わります。

下図に示す通り、ウインドウのサイズを越えることが避けられるように、フローブロックの全体サイズが大きいほど、設定できる範囲が小さくなります。逆に、全体サイズが小さいほど、設定できる範囲は大きくなります。



下図に示す通り、図 34.4A 部分の上方と下方がぶつかる为了避免するため、フローブロックの曲がり角の間隔がより小さいほど、設定できる範囲がより小さくなります。逆に、図 34.4B の通りフローブロックの曲がり角が大きいほど、設定できる範囲が大きくなります。



例 1

以下では[動的速度]を使ってフローブロックの流速及び流れ方向を調整する方法を説明します。

1. フローブロックオブジェクトを作成し、[動的速度]機能を選択します。アドレスを LW-0、フォーマットを 16-bit Signed に設定します。



2. 一個の数値オブジェクトを作成します。アドレスを LW-0 で、上限を 25、下限を-25、フォーマットを 16-bit Signed に設定します。



3. シミュレーションを実行、または HMI にダウンロードすることで操作を検証します。LW-0 で整数を入力したら、流れ方向が順方向になり、それに数値が大きいほど、流速が速くなります。負数を入力したら、流れ方向が逆方向になり、それに数値が小さいほど、流速が速くなります。0 を入力したら、流動を中止します。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。

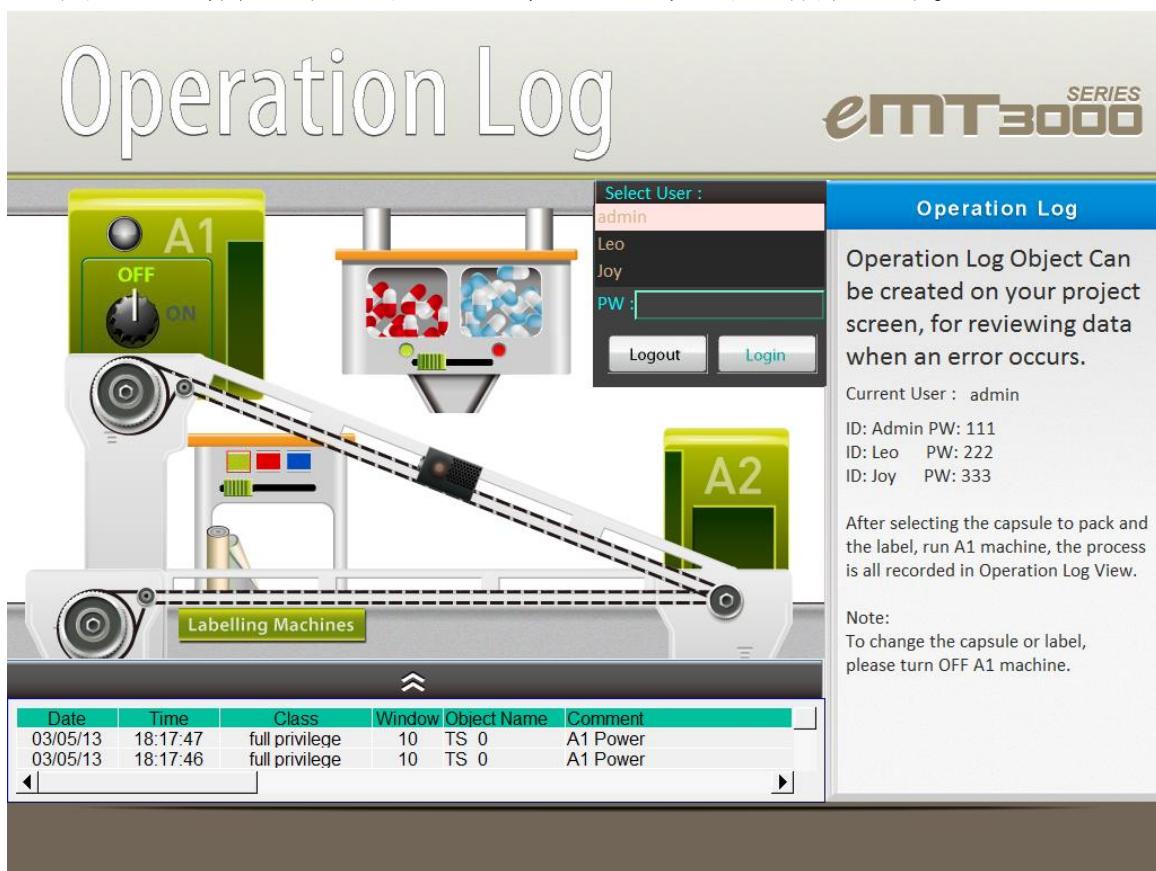
サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.35.操作ログ

13.35.1. 操作ログ設定

13.35.1.1. 概要

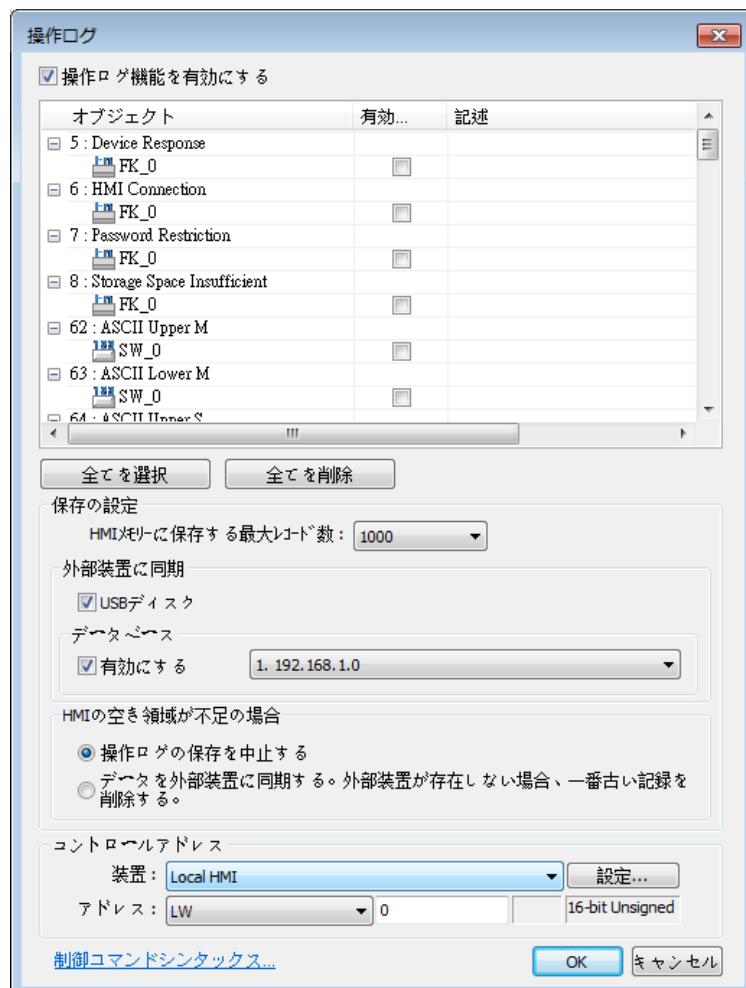
操作ログの機能を通じ、ユーザーが操作しているステップが即時に表示されます。異常な状況が出た場合、操作ログを分析することによって、操作の過程はプロセスの規定に合っているかを検証し、問題がある部分に検討、修正をすることができます。操作ログを適切に設定すれば、審査追跡及び電子署名の要求にも応えられ、データの完全性を保障します。



13.35.1.2. 設定



操作ログを記録するオブジェクトを設定します。[データ/履歴]をクリックし、[操作ログ]に指向し、[操作ログ設定]をクリックし、また[操作ログ機能を有効にする]を選択します。

**設定****記述****オブジェクト**

操作ログ機能を使用したら、システムはウインドウの番号によって自動的に全ての作成済みで記録されるオブジェクトをリストにします。

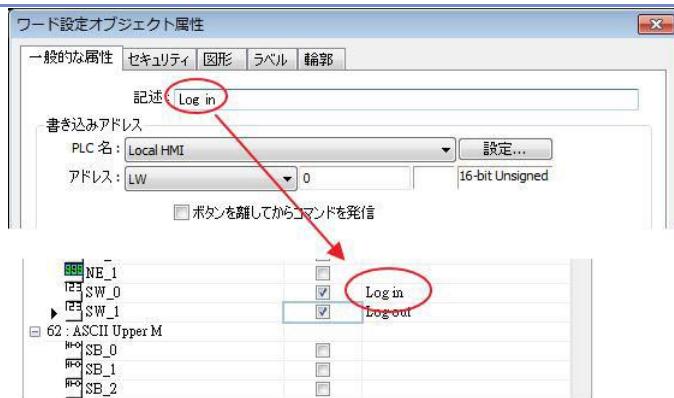
フィルタ アイコンをクリックすると、全ての記録されるオブジェクトの種類がリストに表示されます。ユーザーはフィルタ機能を通して、迅速に記録したいオブジェクトを探し出すことができます。

有効にする

オブジェクトにログ機能を使用するのかを選択します。

記述

オブジェクトの記述です。下図のように示されます。

**全てを選択**

全てのオブジェクトにログ機能を選択します。

[フィルタ]機能を使用した場合、[全てを選択]はリストに表示されるオブジェクトだけを選択します。

全てを削除

選択中のオブジェクトを全部削除します。

[フィルタ]機能を使用した場合、[全てを削除]はリストに表示されるオブジェクトだけを削除します。

新しいオブジェクトの操作ログを自動的に有効にします

本項を有効にすると、新規作成されたオブジェクトは自動的に操作ログを有効にします。

保存の設定

操作ログデータの保存形式を設定します。

HMIメモリーに保存する最大レコード数

HMI内蔵メモリーに保存できるデータの最大レコード数を設定します。

外部装置に同期/バックアップ

データをSDカードまたはUSBディスクにバックアップします。データベースに同期することもできます(cMTシリーズ)。

HMIの空き領域が不足の場合

HMI内の空き領域が不足の場合、前の操作ログを確保するため、[操作ログを一時停止]を選択するか、または[データを外部装置に同期する]を選択します。外部装置に保存するのを選択すれば、該当する装置が存在しない場合、HMIは自動的にHMI内蔵メモリー内の一一番古い記録を削除します。

コントロールアドレス

特定の数値を入力すれば、選択している操作ログにコマンドを実行し、戻り値を返すことができます。

仮にコントロールアドレスをLW-n(nは任意のアドレス)に設定すれば、戻り値を表示するアドレスはLW-n+1になります。

コントロールアドレス(LW-n) :

-
- (1) : 全ての記録を削除する
 - (2) : データをUSBディスクにバックアップする
 - (3) : データをSDカードにバックアップする
 - (4) : データをUSBディスクにバックアップしてから、HMI内の記録を削除する
 - (5) : データをSDカードにバックアップしてから、HMI内の記録を削除する
 - (6) : HMIで操作ログ機能を有効にする
 - (7) : HMIで操作ログ機能を無効にする
 - (8) : HMIを変更した後、本来USBディスクに保存された履歴データを使用する
 - (9) : HMIを変更した後、本来SDカードに保存された履歴データを使用する
 - (10) : データをデータベースサーバーにコピーします。(cMTシリーズ)
 - (11) : データをデータベースサーバーにコピーしてから、HMIメモリ内のデータを消去する(cMTシリーズ)
 - (12) : HMIを交換した後、本来データベースに保存した履歴データを流用する(cMTシリーズ)
- コマンドの戻り値(LW-n+1) :
- (0) : 処理中
 - (1) : 実行成功
 - (2) : 装置が存在しない
 - (3) : データが存在しない
 - (4) : 判断できないエラー
-

Note

- 操作ログは手動でトリガーできるオブジェクトだけを記録します。被動的なオブジェクトは記録されません。例：時間ベースデータ転送オブジェクト。
- オフライン/オンラインシミュレーションを使用する場合、操作ログ記録は EasyBuilder インストールフォルダの `HMI_memory\operationlog\operationlog.db` に保存されます。
- ビット状態設定オブジェクトを使用し、マクロをトリガーする場合、操作ログには 2 レコードのデータ：ビットのトリガー、及びマクロのトリガーを記録します。

13.35.2. 操作ログ閲覧

13.35.2.1. 概要

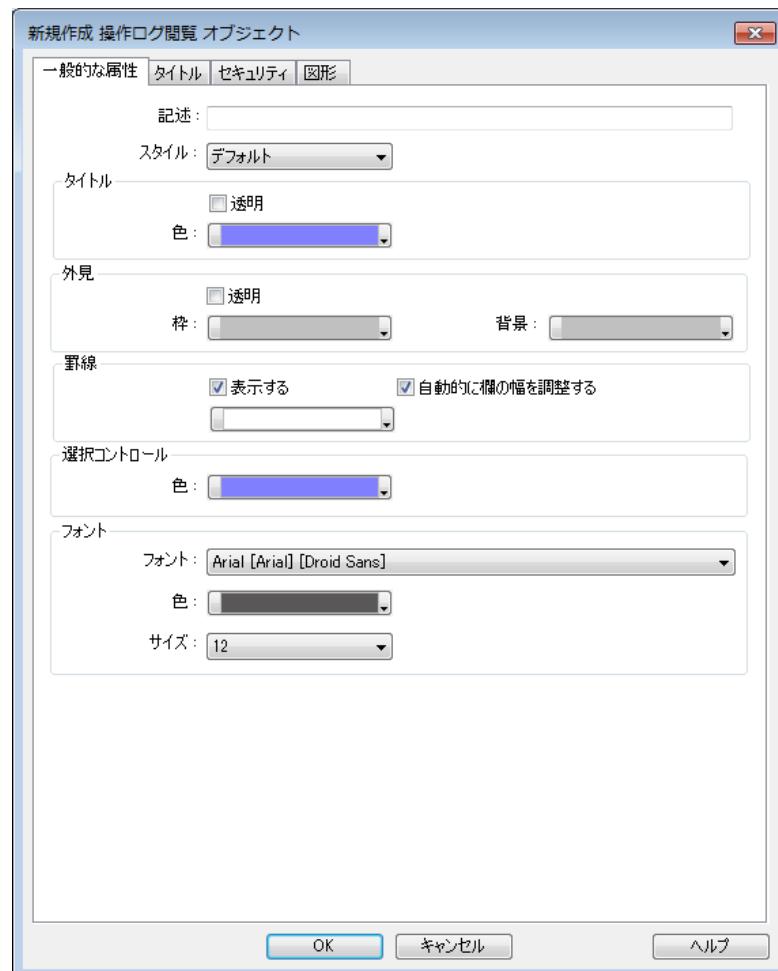
[操作ログ閲覧]オブジェクトを使用することで、操作ログを閲覧することができます。

13.35.2.2. 設定



このオブジェクトを使用する前、まずは[操作ログ設定]で閲覧対象のオブジェクトを設定する必要があります。» [データ/履歴] » [操作ログ閲覧]の順でクリックしてください。

一般的な属性の設定



[タイトル]				
Date	Time	Class	Window	Level
03/04/13	15:15:07	full privilege	10	Info
03/04/13	15:15:07	full privilege	10	Info
03/04/13	15:15:06	full privilege	10	Info
03/04/13	15:15:00	full privilege	10	Info
03/04/13	15:14:55	full privilege	10	Info
03/04/13	15:14:55	full privilege	10	Info
02/26/13	18:50:21	full privilege	10	Info
02/26/13	18:50:18	full privilege	10	Info
02/26/13	18:50:15	full privilege	10	Info
02/26/13	18:50:09	full privilege	10	Info
02/26/13	18:50:07	full privilege	10	Info

設定	記述
スタイル	操作ログテーブルのスタイルを選択できます：デフォルト、クリスタルスタイル、フラットスタイル
タイトル/外見 /罫線/選択コントロール	スタイルに[デフォルト]を選択する時、このパラメータを設定できます。
フォント	[操作ログ閲覧]オブジェクト上の文字のフォント、サイズ、色を設定します。

ラベル設定



設定	記述
タイトル	[操作ログ]オブジェクトで表示するタイトルを設定します。
配列	操作ログの配列方式を設定します。
表示順番	ログが作成された時、項目情報の表示順番を設定します。[表示キャラクタ]を 0 に設定した場合、システムは全てのキャラクタを表示します。
日付/時間	[操作ログ閲覧]オブジェクト上の日付及び時間のフォーマットを設定します。

13.35.3. 操作ログ印刷

13.35.3.1. 概要

[操作ログ印刷]は操作ログの内容をリストにし、外部記憶装置或いはプリンターにインポートする機能です。外部記憶装置を使用する場合、リストは JPEG ファイルに転じます。本機能を使用する前、まず[操作ログ設定]を使用する必要があります。

cMT シリーズ HMI を使用する場合、cMT Viewer で印刷をトリガーする時、操作ログは HMI に接続された USB/SD カード装置に保存されます。

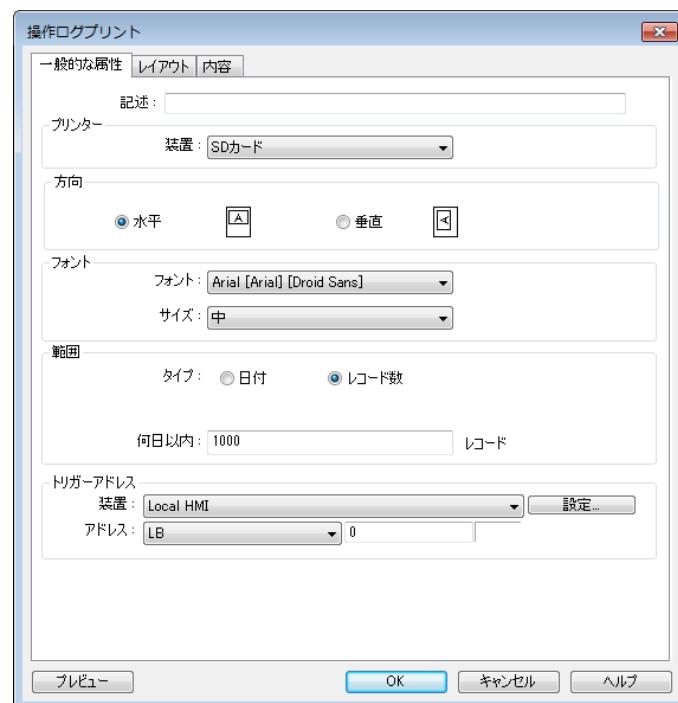
13.35.3.2. 設定



[操作ログ]の[プリント機能を有効にする]にチェックマークを入れ、[設定]をクリックしプリントの設定に入ります。

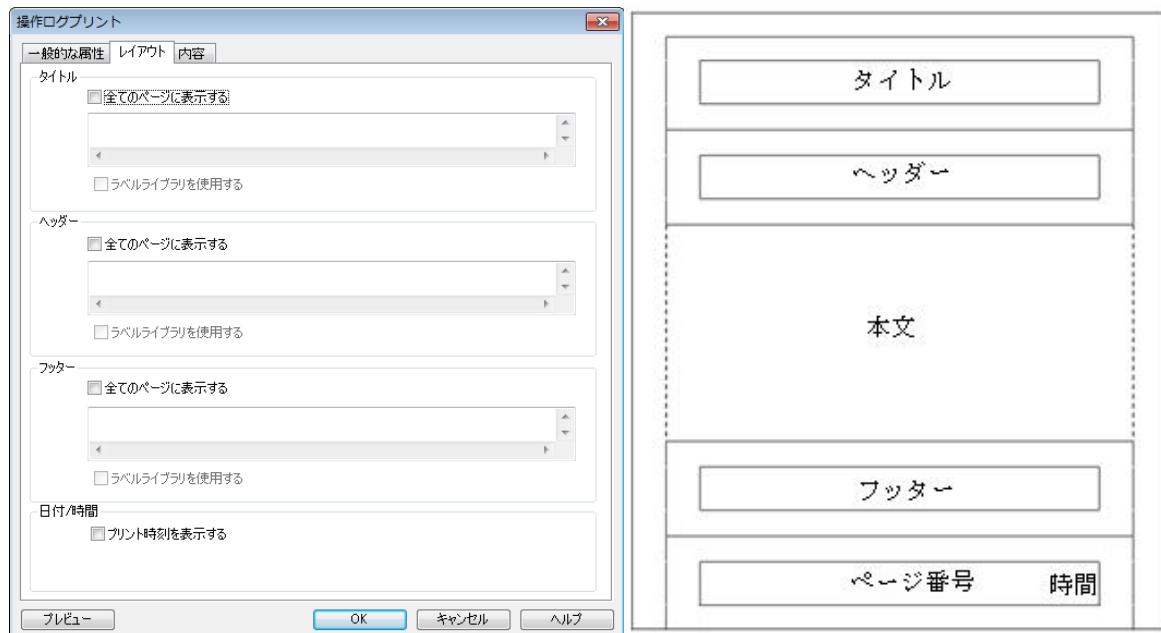


一般的な属性の設定



設定	記述												
プリンター	[操作ログ]データを保存する外部記憶装置を選択します。プリンターを選択する場合、フォーマットは A4 である必要があります。外部記憶装置に保存するのを選択した場合、システムは operationlogsheet フォルダーを作成し、操作ログのリストを JPEG に転じ、また"プリント日付_プリント番号"の形でファイルを命名し、このフォルダに保存します。 例えば: 2013/05/08 でプリントした一枚目の図面の JPEG ファイル名は 130508_0000、このように類推します。												
方向	[操作ログ]データを出力する時の方向(水平/垂直)を設定します。												
フォント	[操作ログ]データを出力する時のフォント及びサイズを設定します。文字サイズを大中小に設定する場合、対応サイズは下表の通りです:												
	<table border="1"> <thead> <tr> <th>サズ</th><th>タイトルサイズ</th><th>本文サイズ</th></tr> </thead> <tbody> <tr> <td>大</td><td>20pt</td><td>16pt</td></tr> <tr> <td>中</td><td>16pt</td><td>12pt</td></tr> <tr> <td>小</td><td>12pt</td><td>8pt</td></tr> </tbody> </table>	サズ	タイトルサイズ	本文サイズ	大	20pt	16pt	中	16pt	12pt	小	12pt	8pt
サズ	タイトルサイズ	本文サイズ											
大	20pt	16pt											
中	16pt	12pt											
小	12pt	8pt											
範囲	[操作ログ]データ出力時のデータ範囲を設定します。												
日付	日付でデータの出力範囲を決定する場合、[開始時間]を前への[何日内]で範囲を決めます。最大は 30 日間です。												
レコード数	レコード数で出力範囲を決定する場合、最大は 10000 レコードです。												
トリガーアドレス	[操作ログ印刷]をコントロールするレジスタアドレスソースを設定します。レジスタが ON に設定された場合、出力をトリガーします。出力完了後、レジスタは自動的に OFF 状態に復帰します。												
プレビュー	出力する画面をプレビューします。												

レイアウト設定



レイアウト設定タブの各部分の配置は右図の通りです。

設定	記述
タイトル	<p>表示するタイトルの内容を設定します。タイトルは一行であること。</p> <p>全てのページに表示する</p> <p>チェックマークを入れると、タイトル内容は各ページに表示されます。チェックマークを入れなかつたら1ページ目にだけ表示されます。</p>
ヘッダー	<p>表示するヘッダーの内容を設定します。ヘッダーには最大5行の文字をプリントすることができます。</p> <p>全てのページに表示する</p> <p>チェックマークを入れると、ヘッダー内容は各ページに表示されます。チェックマークを入れなかつたら1ページ目にだけ表示されます。</p>
フッター	<p>表示するフッターの内容を設定します。フッターには最大5行の文字をプリントすることができます。</p> <p>全てのページに表示する</p> <p>チェックマークを入れると、フッター内容は各ページに表示されます。チェックマークを入れなかつたら、1ページ目にだけ表示されます。</p>
日付/時間	<p>チェックマークを入れると、プリントされた時の日付/時間は各ページの右下隅に表示されます。チェックマークを入れなかつたら、日付/時間を表示しません。</p>
ページ番号	どのページにも表示されます。

内容設定



設定	記述
タイトル	タイトルを設定します。
配列	時間順で配列する 操作ログは古いから新しいへのように配列され、最新の操作ログはボトムに表示されます。 逆時間順で配列する 操作ログは新しいから古いへのように配列され、最新の操作ログはトップに表示されます。
日付/時間	時間情報の表示フォーマットを設定します。

13.35.3.3. 使用例

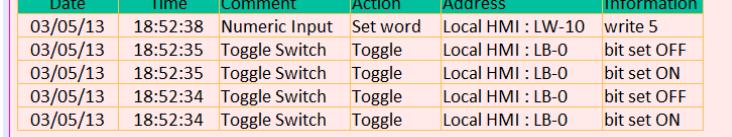
例 1

本例では、操作ログを新規作成する仕方を説明します。

- [トグルスイッチ]オブジェクトと[数値]オブジェクトをウインドウ 10 で新規作成します。
- [操作ログ設定]オブジェクトをオープンし、ウインドウ 10 内の[トグルスイッチ]オブジェクトと[数値]オブジェクトを使用します。

オブジェクト	有効に...	記述
5 : Device Response	<input type="checkbox"/>	
FK_0		
6 : HMI Connection	<input type="checkbox"/>	
FK_0		
7 : Password Restriction	<input type="checkbox"/>	
FK_0		
8 : Storage Space Insufficient	<input type="checkbox"/>	
FK_0		
10 : WINDOW_010	<input checked="" type="checkbox"/>	Toggle Switch
TS_0	<input checked="" type="checkbox"/>	Numeric Input
NE_0		
62 : ASCII Upper M	<input type="checkbox"/>	
SYM_0		

3. [操作ログ閲覧]オブジェクトを新規作成します。各項の属性を設定完了後、ダイアログボックスを閉じます。
4. オフラインシミュレーションを実行し、[トグルスイッチ]オブジェクトと[数値]オブジェクトをトリガーします。操作ログは[操作ログ閲覧]オブジェクトに表示されます。

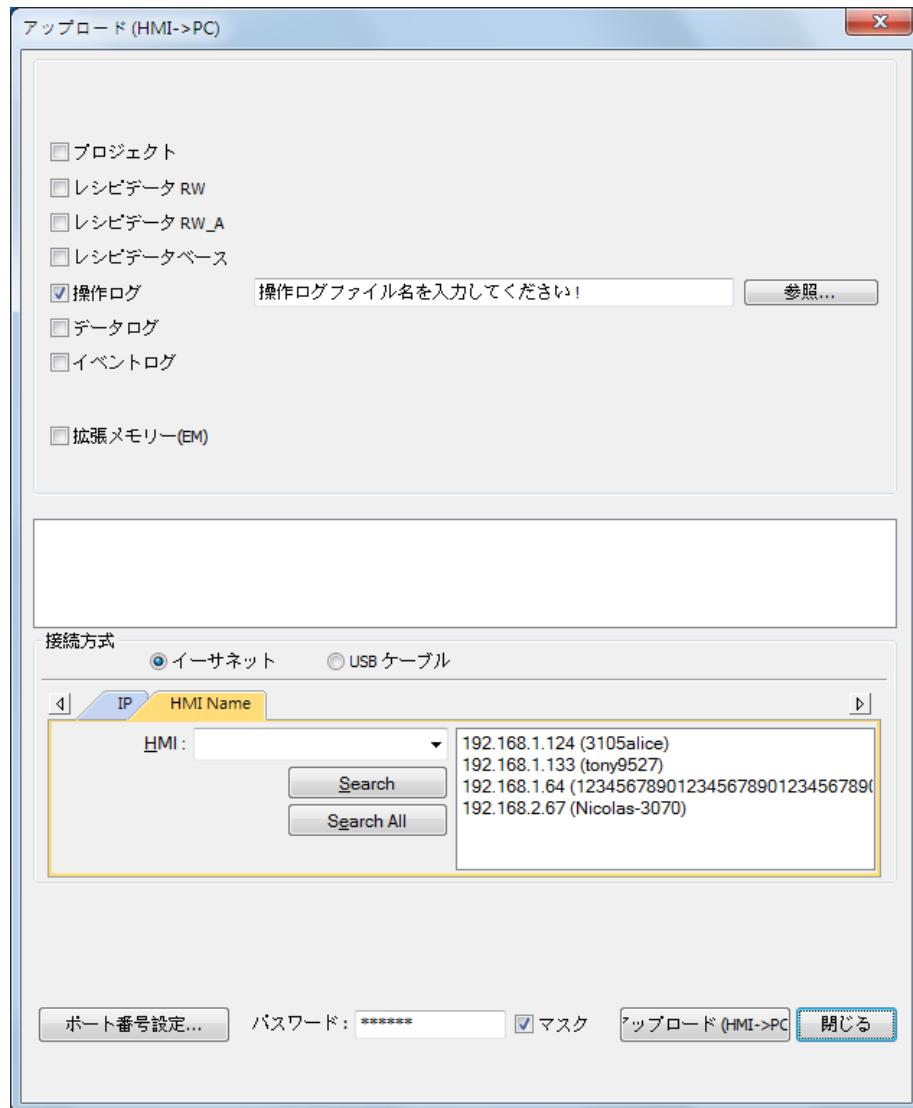
 Toggle	5	Numeric Input
		
Date	Time	Comment
03/05/13	18:52:38	Numeric Input
03/05/13	18:52:35	Toggle Switch
03/05/13	18:52:35	Toggle Switch
03/05/13	18:52:34	Toggle Switch
03/05/13	18:52:34	Toggle Switch
		Action
		Set word
		Toggle
		Address
		Local HMI : LW-10
		Local HMI : LB-0
		Information
		write 5
		bit set OFF
		bit set ON
		bit set OFF
		bit set ON

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

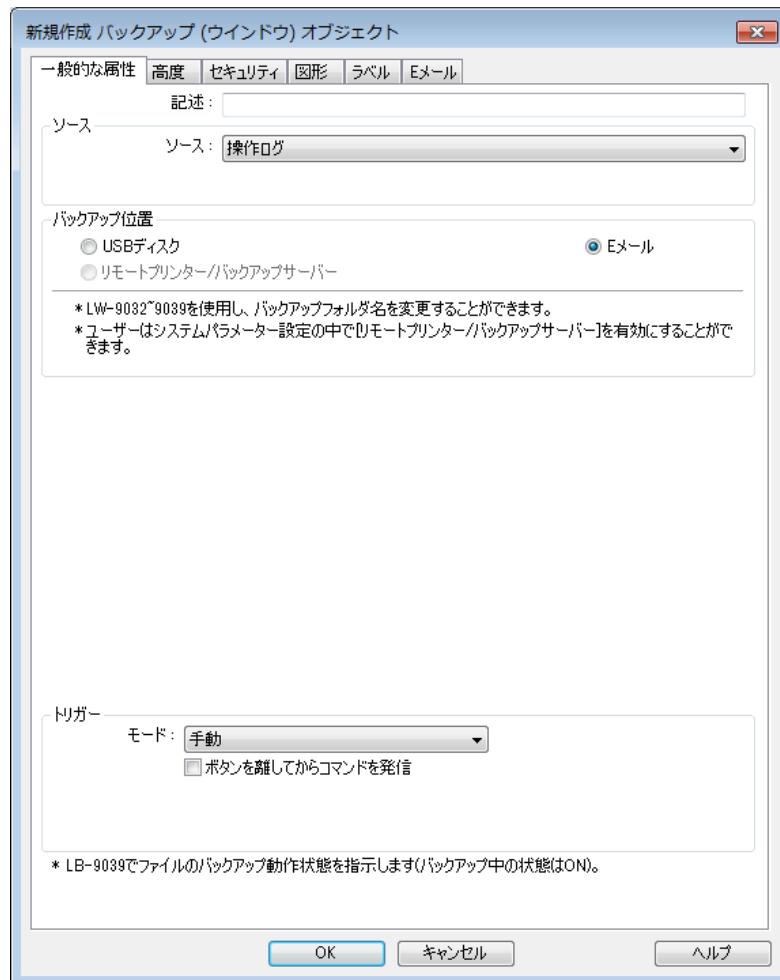
例 2

操作ログデータは Utility Manager を通じて、PC にアップロードまたは[バックアップ]オブジェクトを使って、メールの形でメールアドレスに発送することができます。

- Utility Manager を使用し、PC にアップロードする場合
1. Utility Manager をオープンし、[アップロード]をクリックします。
 2. [操作ログ]にチェックマークを入れ、ファイル名及び HMI の IP を入力し、[アップロード]をクリックします。



- メールの形で、メールアドレスに発送する場合
- 1. [システムパラメータ設定] » [メール]をクリックし、メールのサーバー及び宛先/送信元のメールアドレスを設定します。
- 2. [バックアップ]オブジェクトを作成し、ソースを[操作ログ]に、保存先を[メール]に設定します。



👉 メールサーバーの設定方法は《5 システムパラメータ設定》をご参照ください。

⬇ このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.36.複合ボタン

13.36.1. 概要

複合ボタンオブジェクトは複数のコマンドを実行することができます。今まで同一位置で複数のコマンドを実行したい場合、オブジェクトを同一位置に重ねる必要があり、それに実行する時には重ねる順でコマンドを実行します。ユーザーは実行したい順でオブジェクトを重ね、プロジェクトをデザイン時にはより多くの時間をかけて実行順をテストしなければなりません。複合ボタンはユーザーに直接に複数のコマンドを設定することを可能にし、実行順も調整できます。

下記には複合ボタンオブジェクトの特長を挙げます：

- 複数のコマンドを実行可能。
- ユーザー自分で複数コマンドの実行順を調整することができます。
- ビットまたはワードを使用し、オブジェクトの状態として表示可能。

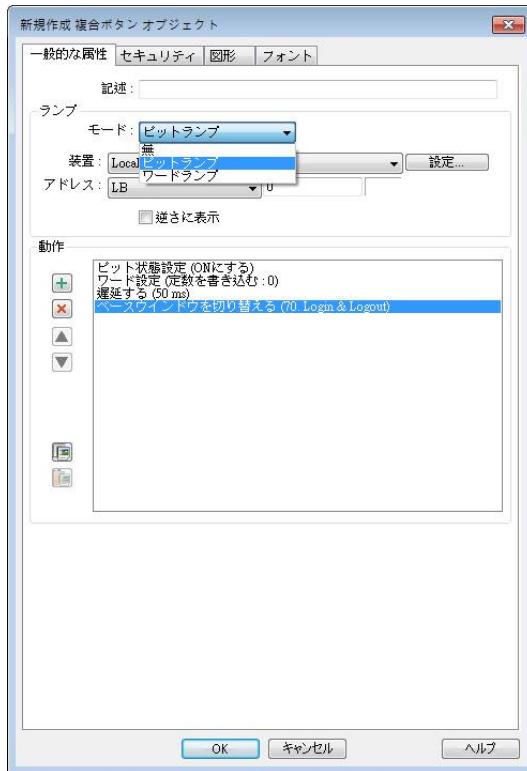
13.36.2. 設定



ツールバー上の[オブジェクト]»[複合ボタン]ボタンを押せば、[複合ボタン]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[複合ボタン]オブジェクトが新規作成されます。

13.36.2.1. eMT、iE、XE、mTV シリーズ

一般的な属性の設定



設定	記述
ランプ	オブジェクトの状態を表示するモードを設定します。マルチ状態表示を使用するかを選択可能。
無	マルチ状態表示を使用しません。
ビットランプ	ビットアドレスのデータを読み取り、状態を表示します。[逆さに表示]は読み取りの状態を逆さまに表示できます。例えば、ビットの実際状態は OFF だったが、[逆さに表示]にチェックマークを入れたら ON に表示されます。
ワードランプ	ワードアドレスのデータを読み取り、状態を表示します。 [状態数]はオブジェクトが表示する状態の数です。状態は 0 から番号を付け、表示できる最大状態番号は設定した[状態数]-1 で、設定した状態数を超えたものを求める場合、システムはラストの状態を表示します。例えば、[状態数]を 8 に設定した場合、順番で 0、1、2、...、7 の状態を表示します。レジスタに状態 8 以上の状態を表示するように要求した際に、状態 7 が表示されます。

ファンクションキーの説明	一つの複合ボタンは最大 20 動作を実行できます。												
	<table border="1"> <thead> <tr> <th>ファンクションキー</th><th>記述</th></tr> </thead> <tbody> <tr> <td></td><td>動作を新規追加します。</td></tr> <tr> <td></td><td>選した動を削除します。</td></tr> <tr> <td></td><td>選択した動作の行を変更します。</td></tr> <tr> <td></td><td>選択した動作をコピします。</td></tr> <tr> <td></td><td>選択した動作を貼り付けます。</td></tr> </tbody> </table>	ファンクションキー	記述		動作を新規追加します。		選した動を削除します。		選択した動作の行を変更します。		選択した動作をコピします。		選択した動作を貼り付けます。
ファンクションキー	記述												
	動作を新規追加します。												
	選した動を削除します。												
	選択した動作の行を変更します。												
	選択した動作をコピします。												
	選択した動作を貼り付けます。												
動作	<p>遅延 n ミリ秒に遅延してから、コマンドを実行します。一個の複合ボタンには一個の[遅延]コマンドのみ設定できます。</p> <p>ビット設定 レジスタの状態を ON または OFF に設定します。</p>												
	<table border="1"> <thead> <tr> <th>スイッチタイプ</th><th>記述</th></tr> </thead> <tbody> <tr> <td>ON にする</td><td>このオブジェクトを押すと、指定したレジスタの状態が ON に設定されます。</td></tr> <tr> <td>OFF にする</td><td>このオブジェクトを押すと、指定したレジスタの状態が OFF に設定れます。</td></tr> <tr> <td>トグル</td><td>このオブジェクトを押すと、指定したレジスタの状態が逆さまに表示されます。即ち ON→OFF, OFF→ON になります。</td></tr> </tbody> </table>	スイッチタイプ	記述	ON にする	このオブジェクトを押すと、指定したレジスタの状態が ON に設定されます。	OFF にする	このオブジェクトを押すと、指定したレジスタの状態が OFF に設定れます。	トグル	このオブジェクトを押すと、指定したレジスタの状態が逆さまに表示されます。即ち ON→OFF, OFF→ON になります。				
スイッチタイプ	記述												
ON にする	このオブジェクトを押すと、指定したレジスタの状態が ON に設定されます。												
OFF にする	このオブジェクトを押すと、指定したレジスタの状態が OFF に設定れます。												
トグル	このオブジェクトを押すと、指定したレジスタの状態が逆さまに表示されます。即ち ON→OFF, OFF→ON になります。												
	<p>マルチ状態設定 指定したレジスタ内のデータを変更できます。</p> <table border="1"> <thead> <tr> <th>スイッチタイプ</th><th>記述</th></tr> </thead> <tbody> <tr> <td>定数を設定する</td><td>定数を指定したレジスタに書き込みます。</td></tr> <tr> <td>増加(JOG+)</td><td>数値を増加する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値に[増加値]で設定した数を足します。但し、増加した結果は[上限値]の設定を超えられません。</td></tr> <tr> <td>減少(JOG-)</td><td>オブジェクトを一回押すとに、指定したジスタ内の数値に[減少値]で設定した数値を引きます。但し、減少した結果[下限値]の設定値より低くなることはありません。</td></tr> <tr> <td>動的制限</td><td>上限/下限は指定したレジスタより設定されます。</td></tr> </tbody> </table>	スイッチタイプ	記述	定数を設定する	定数を指定したレジスタに書き込みます。	増加(JOG+)	数値を増加する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値に[増加値]で設定した数を足します。但し、増加した結果は[上限値]の設定を超えられません。	減少(JOG-)	オブジェクトを一回押すとに、指定したジスタ内の数値に[減少値]で設定した数値を引きます。但し、減少した結果[下限値]の設定値より低くなることはありません。	動的制限	上限/下限は指定したレジスタより設定されます。		
スイッチタイプ	記述												
定数を設定する	定数を指定したレジスタに書き込みます。												
増加(JOG+)	数値を増加する機能です。オブジェクトを一回押すごとに、指定したレジスタ内の数値に[増加値]で設定した数を足します。但し、増加した結果は[上限値]の設定を超えられません。												
減少(JOG-)	オブジェクトを一回押すとに、指定したジスタ内の数値に[減少値]で設定した数値を引きます。但し、減少した結果[下限値]の設定値より低くなることはありません。												
動的制限	上限/下限は指定したレジスタより設定されます。												

		仮に動的制限アドレスは LW-n で、[増加(JOG+)]機能を使用する場合、上限を設定する必要があります；逆に、[減少(JOG-)]機能を使用する場合、下限を設定する必要があります。
--	--	--

13.36.2.2. cMT シリーズ

一般的な属性の設定



[ビット/ワードランプ]の動作は概ね eMT、iE、XE、mTV シリーズと同じなので、Ch13.36.2.1 の説明をご参考ください。動作を設定する時に、事前に動作をグループ分けにすれば、同一グループ内の動作は一緒に実行されます。現在のグループ内の全ての動作がトリガーされてから、次のグループの動作が実行されます。動作グループの注意点については本マニュアル Ch13 の説明をご参照ください。

設定	記述
ダウンアクション	手がボタンを押したたびに、即時に実行する動作です。
アップアクション	手がボタンを離した後、実行する動作です。

ヨン**[遅延する]動作**

n ミリ秒を遅延してから、コマンドを実行します。

[ビット設定]動作

レジスタの状態を ON または OFF に設定します。

ON に設定する

指定したレジスタの状態が ON に設定されます。

OFF に設定する

指定したレジスタの状態が OFF に設定されます。

トグル

指定したレジスタの状態が逆さに表示されます。即ち ON→OFF, OFF→ON になります。

モーメンタリ

モーメンタリを選択した場合、ボタンを押すと(ダウンアクション)ビット状態が ON に設定され、手を放せば(アップアクション)、状態は OFF に設定されます。

[ワード設定]動作

指定したレジスタの数値を変更します。

定数を書き込む

定数を指定したレジスタに書き込みます。

増加(JOG+)

数値を増加する機能です。指定したレジスタ内の数値に[増加値]で設定した数を足します。但し、増加した結果は[上限値]の設定を超えられません。

減少(JOG-)

数値を減少する機能です。指定したレジスタ内の数値に[減少値]で設定した数値を引きます。但し、減少した結果は[下限値]の設定値より低くなることはありません。

動的制限

上限/下限は指定したレジスタより設定されます。

定数ストリングを書き込む

特定のストリングを指定したレジスタに書き込みます。

[ベースウインドウを切り替える]動作**ベースウインドウを切り替える**

指定のウインドウに切り替えます。

共有ウインドウを切り替える

共有ウインドウを切り替えます。

前のベースウインドウに戻る

前のベースウインドウに戻ります。例えば、"ウインドウ 10"から "ウインドウ 20"に切り替える場合、本機能を使用すれば、再度に"ウインドウ 10"に戻ることができます。本機能はベースウインドウで

のみ使用できます。

アニメーション設定

cMT シリーズにはウインドウのアニメーション(トランジション)がサポートされます。[アニメーション設定]をクリックした後、選択できるアニメーション効果は：フェードイン、飛んでくる、浮かび上がる、ワイプ、スプリット、サイクル、クロック、ズーム、ターン、プッシュ等があり、アニメーションの表示期間と表示方向も設定できます。

[マクロをトリガする]動作

指定したマクロをトリガーします。グループ内にマクロがある場合、マクロがトリガーされた途端、動作の実行に成功したと視されます。マクロの実行時間がより長い場合でも、システムはマクロの実行が全部完了するのを待つことなく、直接に次のグループの動作をトリガーすることになります。

[ウインドウをポップアップする]動作

指定したウインドウをポップアップします。

[ウインドウを閉じる]動作

現在オープンされているウインドウを閉じます。

[キーボード入力]動作

キーボードの入力信号として用いられ、主に[数値]と[アスキイ]など、キーボードで数値や文字を入力することが必要な場合に使用されます。

Enter: キーボードの入力(Enter)動作と同じです。

Backspace: キーボードのバックスペース(Backspace)動作と同じです。

Clear: レジスタ内に入力したデータを消去します。

Esc: [ウインドウを閉じる]機能と同じで、ポップアップされたキーボードウインドウを閉じることに用いられます。

Delete: キーボードの削除(Delete)動作と同じで、カーソルの右側の数値またはキャラクタを一個削除します。

Left: キーボードの←動作と同じで、カーソルを左へキャラクター一個を移動します。

Right: キーボードの→動作と同じで、カーソルを左へキャラクター一個を移動します。

Inc: 数値を+1 します。

Dec: 数値を-1 します。

ASCII/UNICODE: キーボードで入力するキャラクタを設定します。

[スクリーンのハードコピー]

現在のスクリーンのハードコピーを取ることができ、それにハードコピーを USB ディスクまたは SD カードに保存できます。

[取る]動作**[全てのイベン****ト(アラーム)****を承認する]動****作**

実行すると、一回で全てのイベント(アラーム)を承認することが出来ます。

[データをイン**ポート]動作**

高度セキュリティのユーザーアカウントまたは e-mail の連絡先をインポートすることに用いられます。USB ハードウェアキーでログインすることに設定できます。

データ位置 : USB ディスク、SD カードから読み取るが選べます。

アカウントインポートモード : [上書き]を選択すれば、HMI 内では今回インポートしたアカウントしか保存されません。[付加]を選択すると、HMI 内のアカウントデータは保留され、今回新しくインポートしたアカウントデータが付加されます。

アカウントをインポートした後、ファイルを削除する : USB ディスク内のユーザーアカウントをインポートした後、ソースファイルを削除します。そうしたら、データ漏れの恐れが避けられます。

[トリガー条件**を待つ]動作**

ビット/ワードを使用して異なるトリガー条件を設定します。トリガー条件が満たされた場合、[複合ボタン]が次の[動作グループ]の動作を実行します。

[データ転送**(背景)]動作**

指定したアドレス内のデータを他のアドレスに転送します。

ファイル転送

FTP でファイルを転送します。HMI はパッシブモードで FTP サーバーに接続します。

[一般的な属性]タブ

転送方向を設定します。[ダウンロード]は FTP サーバーからファイルをローカル HMI に転送し、[アップロード]はファイルをローカル HMI から FTP サーバーに転送します。

[ファイル設定]タブ

ファイル位置、FTP サーバーファイルパス、ローカル HMI ファイルパスを設定できます。パスがフォルダである場合、そのフォルダ内にある全部のファイルを転送し、但しサブフォルダ内のファイルを含みません。ファイル名が同じの場合、ファイルを上書きします。

[状態]タブ

ファイル転送の実行結果及びリターンコードを表示するためのアドレスを設定します。

FTP サーバーのリターンコードの一覧表は次のリンクを参考してください : https://en.wikipedia.org/wiki/List_of_FTP_server_return_codes。

 Note

- 一个の複合ボタンでのウインドウ関連の機能は一つだけ実行でき、それに一回だけ使用できます。ウインドウ関連の機能は[ベースウインドウを切り替える]、[ウインドウをポップアップする]、[ウインドウを閉じる]があります。
- 一个の複合ボタンには最大 20 グループを作成でき、各グループに最大 20 個の動作を含むことが可能です。
- ファイル転送機能のエラーコードリスト：

エラーコード	説明
0	実行に成功した
1	ダウンロードする時、HMI のパスワードが存在しません
3	USB ディスクまたは SD カードが検出されていない
4	HMI または FTP のパスが入力されていない
5	アップロードファイルが存在しない
8	操作が FTP に拒否された
9	USB ディスクまたは SD カードの容量が一杯になった
10	未知のエラー

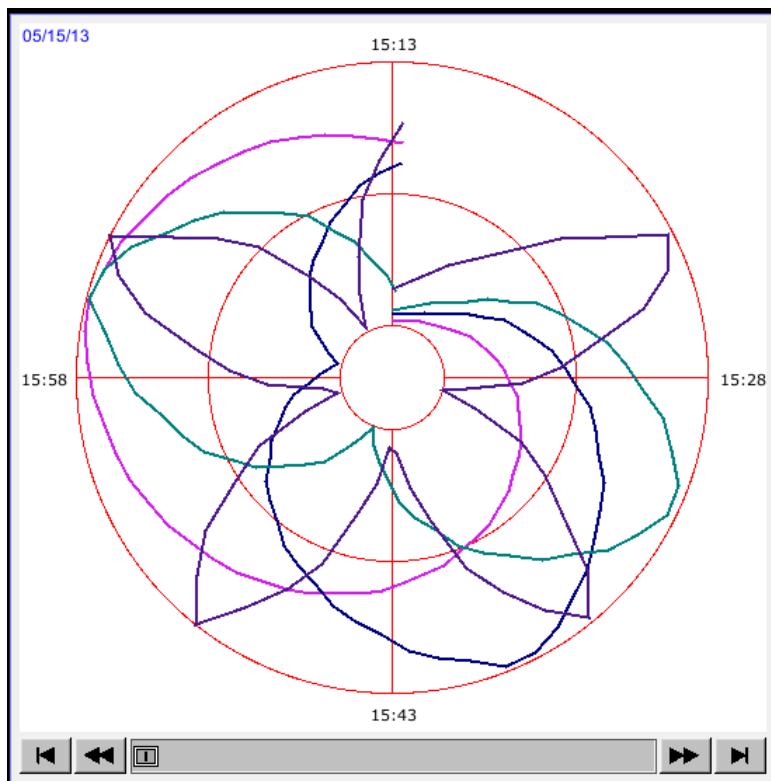


このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.37.近似曲線円グラフ

13.37.1. 概要

[近似曲線円グラフ]は[データサンプリング]のサンプリングデータ及び座標システムを近似曲線円グラフに作成し、半径はy分量、夾角はx分量を示しています。使用上では[トレンド図]に似ています。

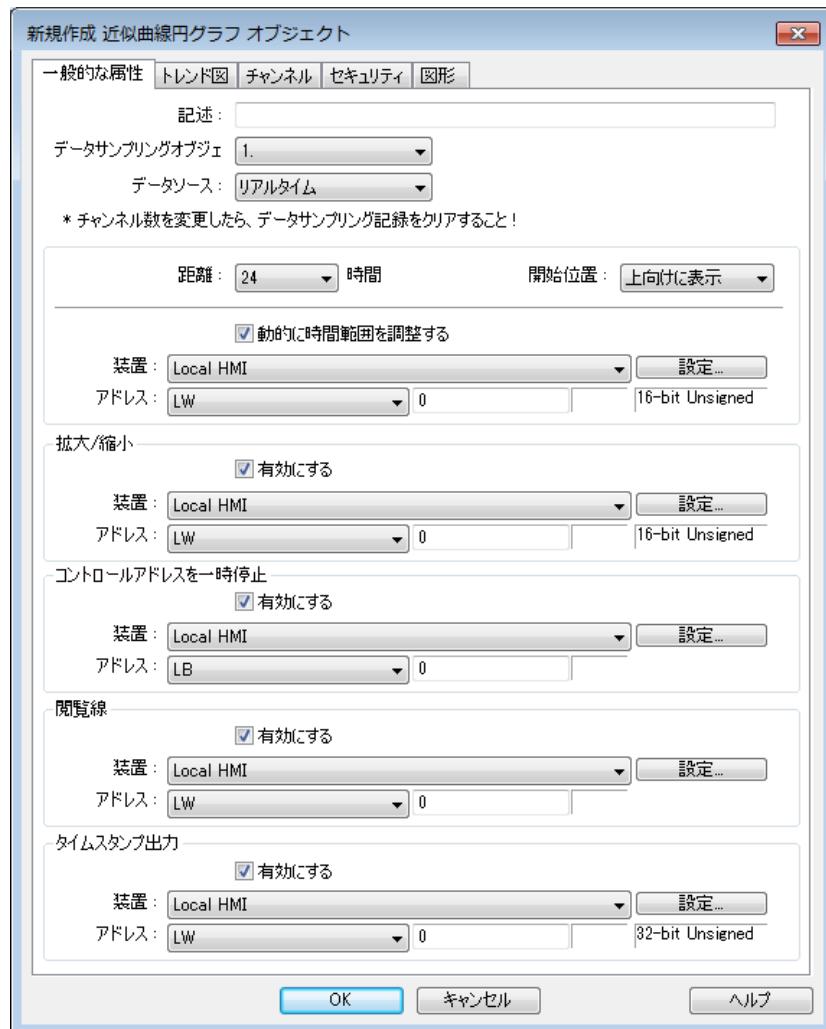


13.37.2. 設定



ツールバー上の[オブジェクト]»[近似曲線円グラフ]ボタンを押せば、[近似曲線円グラフ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[近似曲線円グラフ]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述
データサンプリング	作画のデータソースを選択します。
データソース	<p>データソースのモードを選択します。[リアルタイム]と[履歴]が選択できます。</p> <p>リアルタイム</p> <p>固定したレコード数の[データサンプリング]オブジェクトのサンプリングデータを表示することができます。[データサンプリング]オブジェクトの[最大データ数(リアルタイムモード)]でサンプリングデータの表示数量を設定します。この数量を超えると、より古いデータは画面から削除されます。他の日付、またはより古いデータを表示させたい場合、[履歴]モードを使用する必要があります。</p> <p>[コントロールを一時停止]機能で画面更新を一時停止することができますが、画面だけが中止され、[データサンプリング]オブジェクト</p>

トのサンプリング動作は中止されません。

履歴

履歴データは[データサンプリング]オブジェクトが日付で分類、保存したサンプリングデータです。[履歴]モードを使用したら、[データサンプリング]を通じて表示する履歴データを選定することができます、また[履歴データコントロール]アドレスで異なる日付の履歴データを閲覧することができます。

注意

近似曲線円グラフ設定タブの[画面スクロールボタン]機能を使用していない場合、表示させたいサンプリングデータの時間が[距離]で設定した時間を超えると、この前のリアルタイム/履歴データを閲覧することができません。例：[距離]が1時間に設定されたら、この近似曲線円グラフは1時間前のサンプリングデータを表示することができません。

自動的にデータを更新する

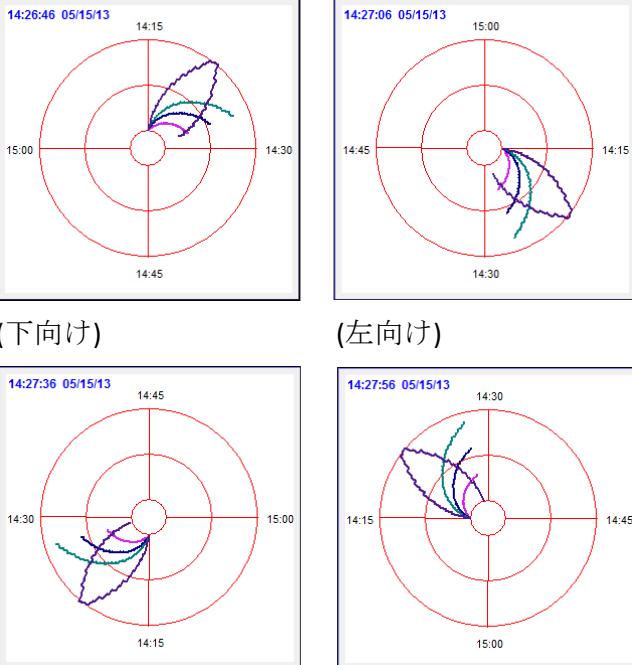
本項を使用すると、毎回[履歴モード]の[近似曲線円グラフ]オブジェクトが所在するウインドウをオープンしたら、オブジェクトの画面は毎秒ごとに更新されます。下記をご留意ください：

- 自動更新機能の状態は画面スクロールボタンで見られます：
アイコンは■の場合、近似曲線円グラフオブジェクトのデータは自動的に更新されると示しています。
アイコンは▶の場合、近似曲線円グラフオブジェクトのデータの更新が中止されたと示しています。
- 前へスクロールし、より古いデータを見る場合、[自動的にデータを更新する]機能は取り消されます。この場合、コントロールボタンのアイコンは▶です。
- [自動的にデータを更新する]にチェックマークを入れると、この前では画面スクロールボタンで自動更新を起動/中止したことがあるかどうかを問わず、このウインドウに切り替えれば、必ず画面が自動更新されます。

例：仮にオブジェクトは[自動的にデータを更新する]機能を使用している場合、前へスクロールして旧データを見たら、自動更新機能は中止されます。この際に、一回他のウインドウに切替してまた本ウインドウに戻れば、オブジェクト画面は依然に自動的に更新されます。

もしプロジェクトを作成した際に、[自動的にデータを更新する]機能を使用していないなく、その後にHMIに本機能を使用したい場合、▶ボタンを押せばいいです。注意すべきなのは、この際の自動更新機能はウインドウが切り替えられたびに使用中止になります。

す。つまり、一旦他のウインドウに切替したら、また本来のウインドウに戻っても、オブジェクトの画面は更新を中止したままです。

距離	円周一周の時間長さです。時間を単位にし、長さ範囲は 1~24(時間)です。
開始位置	作画時の開始位置です。 (上向け) (右向け) (下向け) (左向け)
	

動的に X 軸の時間範囲を調整する	本項を使用すると、一個のワードレジスタを指定し、オンラインで動的に[近似曲線円グラフ]の時間範囲を調整することができます。調整数値は時間を単位にします。レジスタの中に数値が記入されていない場合、デフォルト数値で距離を決めます。
拡大/縮小	オブジェクト表示画面を拡大/縮小します。最大 10 倍までに拡大できます。レジスタ内の数値は 0 の場合、効果は 1 に相当し、原寸サイズに表示されます。
コントロールを一時停止	一時停止ビットレジスタを ON にする場合、近似曲線円グラフ画面の更新が中止されますが、データサンプリングのサンプリング動作は中止されません。本項は[データソース]に[リアルタイム]モードを採用する場合のみ表示されます。
履歴データコントロール	システムが履歴データを時間順で配列したので、履歴データコントロールは表示する履歴データを指定する機能です。履歴データコントロールアドレス内の数値は 0 の場合、最新のデータを表示し、数値は 1 の場合、二番目新しいデータを表示し、このように類推します。

[オプションリスト]オブジェクトを合わせて、データソースで[履歴データ日付]を選択すれば、全ての履歴データは日付で分類され、オプションリストオブジェクトに表示されます。詳細は《13.29 オプションリスト》をご参考ください。

例

履歴データコントロールアドレスは LW-0、また[データサンプリング]オブジェクトが保存したサンプリングデータは 4 レコード：20061120.dtl、20061123.dtl、20061127.dtl、20061203.dtl がある場合、履歴データコントロールアドレス内の数値は下表の通りです：

LW-0 内数値	表示するサンプリングデータ
0	20061203.dtl
1	20061117.dtl
2	20061123.dtl
3	20061120.dtl

閲覧線

本機能を使用すると、近似曲線円グラブオブジェクトにタッチしている時に、一本の閲覧線が生じ、それに閲覧線でのデータを指定したアドレスにインポートします。複数のチャンネルのデータを閲覧したい場合、次のチャンネルのデータは自動的に連続したワードレジスタにロードします。もし各チャンネルのデータフォーマットが異なった場合、チャンネルが対応するレジスタのフォーマットに基づいて配列されます。

例

閲覧アドレスは LW-0、またデータサンプリングは下記 4 フォーマット：16-bit Unsigned、32-bit Unsigned、32bit Signed、16-bit Signed があるの場合、閲覧アドレスは下表の通りです：

チャンネル	データフォーマット	データ長さ	閲覧アドレス
0	1-it Unsigned	1 Word	LW-0
1	32-bit Unsigned	2 Words	LW-1
2	32-bit Signed	2 Words	LW-3
3	16-bit Signed	1 Word	LW-5

タイムスタンプ出力

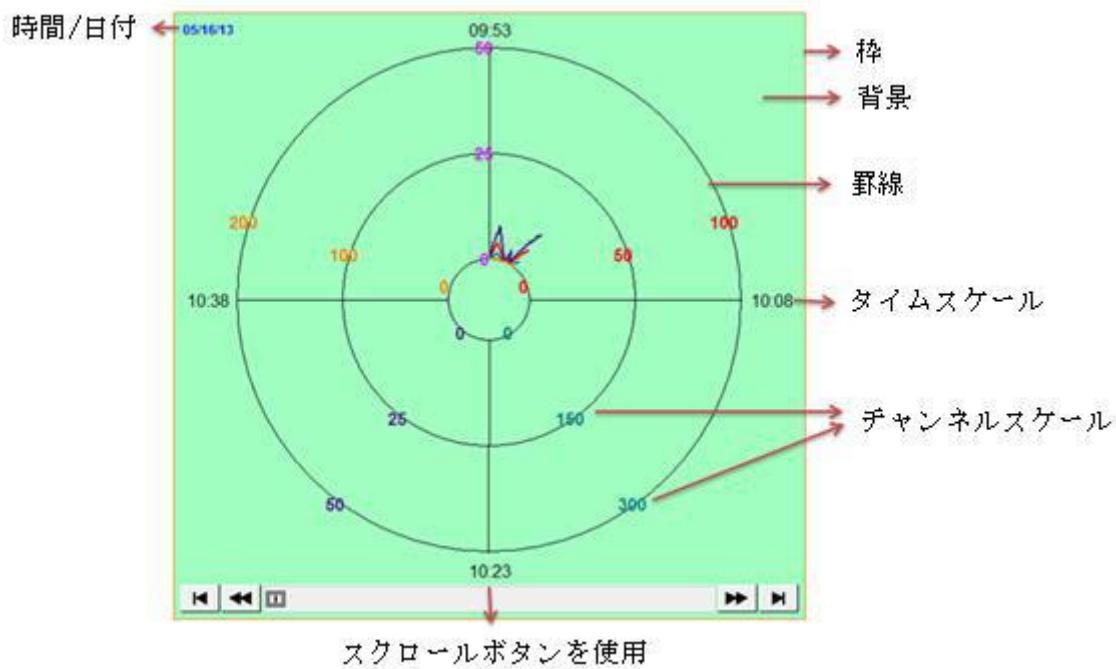
本項を使用すると、システムは一番目のサンプリング点でのサンプリング時刻を時間原点としてカウントし始め、最新のサンプリング点の積算秒数を[タイムスタンプ出力アドレス+2]に出力します。近似曲線円グラフオブジェクト上のカーブをクリックする際に、一番近づいているサンプリング点での積算秒数を[タイムスタンプ出力アドレス]に出力することができます。

注意

[タイムスタンプ出力アドレス]と[タイムスタンプ出力アドレス+2]は全部 32-bit フォーマットであること。[タイムスタンプ出力アドレス+2]はリアルタイムモードのみに適用し、[タイムスタンプ出力アドレス]はリアルタイム及び履歴モード両方にも適用します。

近似曲線円グラフ設定





設定	記述
外見	<p>外枠 オブジェクトの外枠色を設定します。</p> <p>背景 オブジェクトの背景色を設定します。</p> <p>画面スクロールボタン 他の時間範囲でのサンプリングデータを閲覧する場合に用いられます。スクロールの最小単位は、一般的な属性設定タブ内の[距離]で決定されます。[画面スクロールボタン]機能を使用していない場合、表示するサンプリングデータの時間が[距離]で設定した時間を超えたら、この前のリアルタイム/履歴データを閲覧することができません。例：[距離]が1時間に設定されたら、この近似曲線円グラフは1時間前のサンプリングデータを表示することができません。</p>
界線	<p>枠線のスケールと色を設定します。</p> <p>Y 軸 Y 軸の間隔を設定します。</p> <p>フォントサイズ タイムスケール及びチャンネルスケールのフォントサイズを設定します。</p> <p>タイムスケール タイムスケールを表示するかを設定します。時間範囲が1時間より大きい場合、タイムスケールの間隔は1時間を単位とし、時間範囲</p>

が一時間より小さい場合、タイムスケールの間隔は 15 分を単位とします。

チャンネルスケール

チャンネルスケールを表示するかを設定します。各チャンネルスケールの文字の色はチャンネルと同じ色で表示されます。

時刻/日付

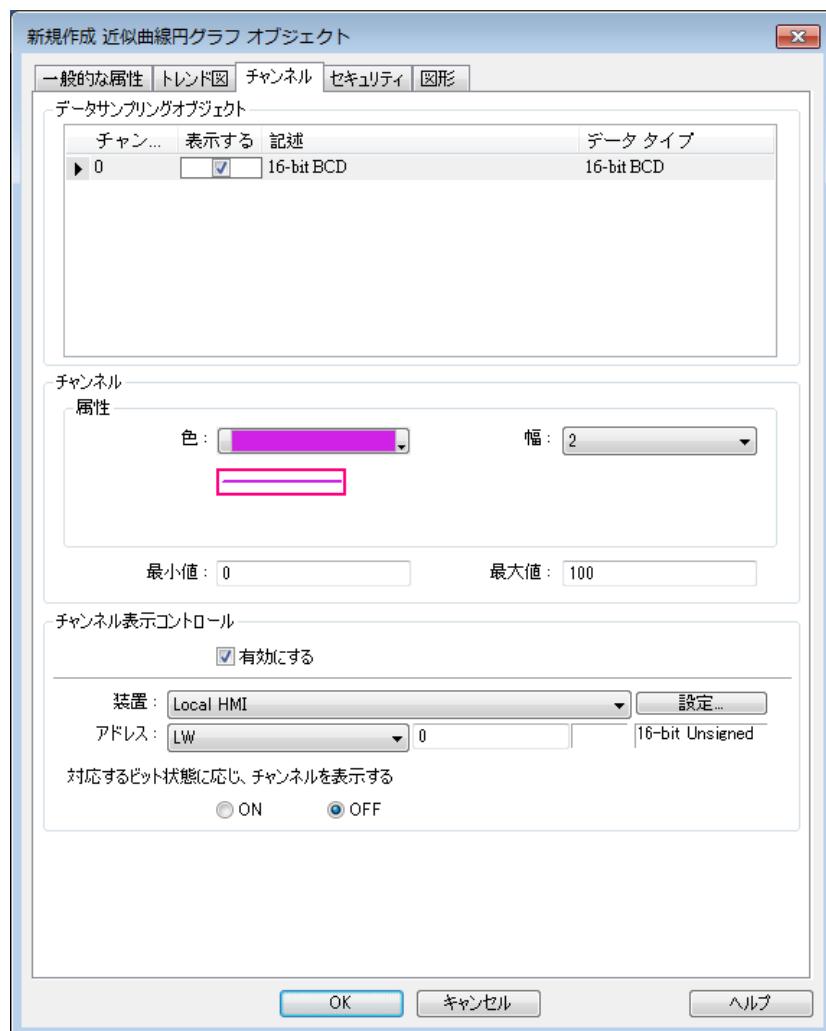
時刻

表示する時間の書式を設定します。

日付

表示する日付の書式を設定します。

チャンネル設定



設定

記述

チャンネル

各カーブのスタイルと色、及びカーブが画けるデータの上下限値を設定します。同時に最大 8 チャンネルをサポートすることができます。

[レジスタから取得]にチェックマークを入れない

データの上限と下限は定数で設定されます。

[レジスタから取得]にチェックマークを入れる

データの上限と下限は指定したレジスタで設定されます。書き込みアドレスは **LW-n** の場合、上/下限は下記規則に基づいて自動的に設定されます：

データアドレス フォーマット	16i	32-bit
下限	L-n	LW-n
上限	LW-n+1	LW-n+2

チャンネル表示コントロール

[使用]にチェックマークを入れると、このアドレス内の各ビットは各チャンネルの表示をコントロールします。Bit-1 はチャンネル 1 をコントロールし、Bit-2 はチャンネル 2 をコントロールし、このように類推します。例を挙げれば、5 つのチャンネルを作成し、またチャンネルを設定するアドレスを **LW-0** にする場合、各チャンネルは下記アドレスにコントロールされます

[チャンネル表示コントロール]を選択したら、このアドレス内の各ビットは各チャンネルの表示をコントロール用いられます。Bit-0 はチャンネル 0 をコントロールし、Bit-1 はチャンネル 1 をコントロール、このように類推します。

対応するビット状態を応じ、チャンネルを表示する

本項を **ON** に設定すると、対応するビットの状態が **OFF** なった場合、チャンネルを隠します。**OFF** に設定すると、対応するビットの状態が **ON** になった場合、チャンネルを隠します。

例

チャンネル表示コントロールアドレスを **LW-0**、また対応するビットの状態が **OFF** になった場合、チャンネルをオープンすると設定したら、仮に 5 チャンネルがある場合、以下のとおりに、表示するかどうかをコントロールします：

チャンネル番号	コントロールアドレス	ビットの状態	表示するかどうか
0	LW_bit-000	OFF	ES
1	LW_bit-001	ON	NO
2	LW_bit002	ON	NO
3	LW_i-003	OFF	YE
4	LW_bit-004	OFF	YES

13.38. 画像閲覧

13.38.1. 概要

[画像閲覧]オブジェクトは外部装置、例えばUSBディスクまたはSDカード内の画像ファイルを閲覧することができます。

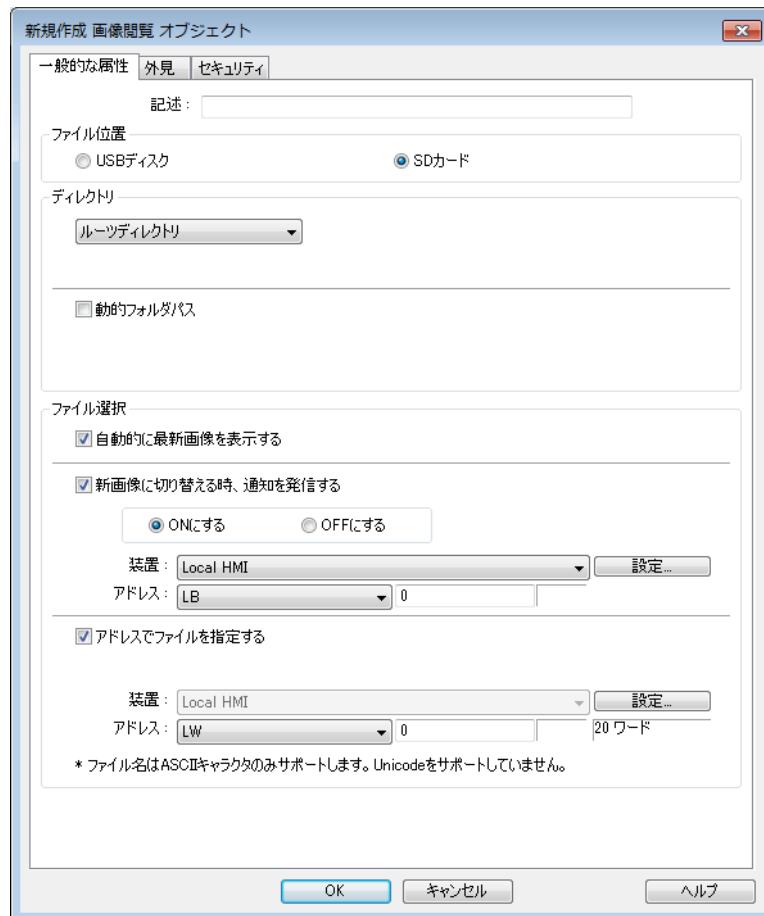
cMTシリーズHMIを使用する場合、cMT Viewerでは本オブジェクトが見られません。

13.38.2. 設定



ツールバー上の[オブジェクト]»[画像閲覧]アイコンを押せば、[画像閲覧]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[画像閲覧]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述
ファイル位置	ファイルのソース位置は、USB ディスクまたは SD カードを選択します。
ディレクトリ	画像の保存先を設定します。 動的フォルダパス ローカルアドレスでフォルダパスを指定することができます。
ファイル選択	自動的に最新画像を表示する フォルダパスで新規画像が生成された時、画像閲覧オブジェクトは自動的に最新画像を表示します。 新画像に切り替える時、通知を発信する 本項を選択すると、HMI が最新の画像に切り替えるたびに、指定したレジスタが ON/OFF に設定されます。 アドレスからファイルを指定する(ツールバーを隠す) 本項を有効にすると、ローカルアドレスのファイル名で表示する画像を指定します。ツールバーが隠されます。

Note

- 画像のファイル名は ASCII キャラクタのみ適用し、Unicode をサポートしていません。
- .jpg, .bmp, .gif, .png フォーマットの画像ファイルをサポートします。

外見設定



設定	記述
外見	ツールバーの位置、オブジェクトの背景色、文字の表示フォントを調整します。 削除ボタンを隠す 有効にすると、削除ボタンがツールバーに表示されません。削除ボタンは画像を削除することに用いられます。 画像サイズがオブジェクトより小さい場合、原寸サイズで表示する 有効にすると、画像サイズがオブジェクトのサイズより小さい場合、画像を原寸サイズで表示させます。これにより、画像の拡大に起因する歪みを避けることができます。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.39. ファイルブラウザ

13.39.1. 概要

[ファイルブラウザ]オブジェクトは、SD カードまたは USB ディスク内のファイル及びフォルダを表示することに用いられます。装置内のファイルを閲覧する以外、[ファイルブラウザ]でクリックされたファイルのファイル名及びパスは、特定のアドレスに書き込まれます。

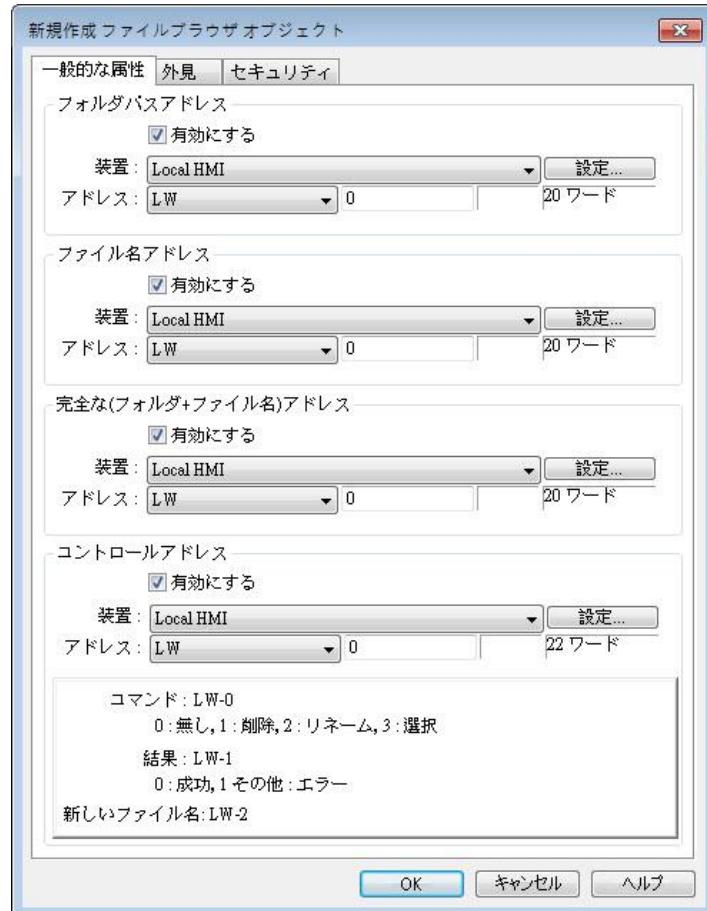
cMT シリーズ HMI を使用する場合、cMT Viewer では本オブジェクトが見られません。

13.39.2. 設定



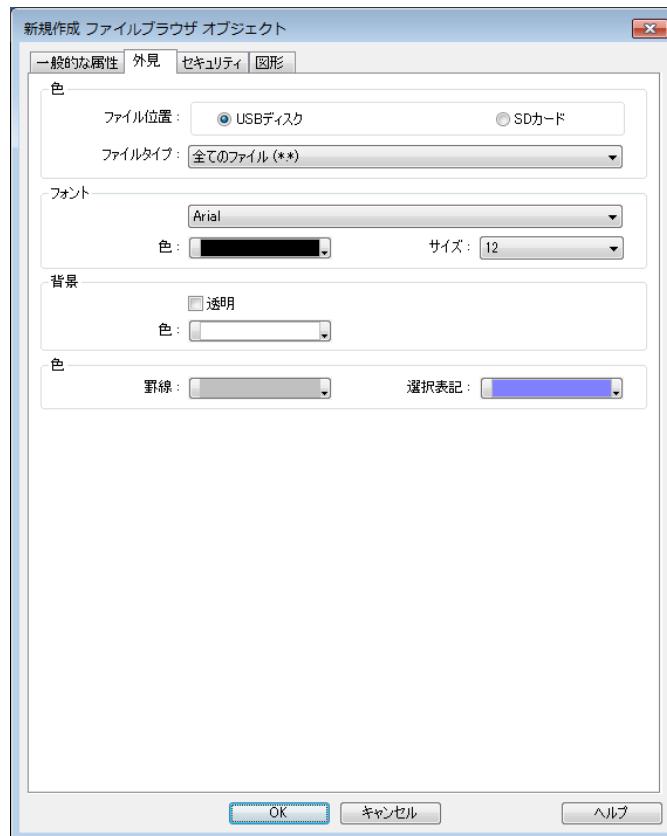
直接に[ファイルブラウザ]アイコンをクリックし、オブジェクトを新規作成してください。またはツールバー上の[オブジェクト]»[オブジェクト]»[ファイル操作]»[ファイルブラウザ]をクリックし、オブジェクトを新規作成します。

一般的な属性の設定



設定	記述
フォルダパスアドレス	現在のパスです。
ファイル名アドレス	現在、ファイルブラウザが選択しているファイル名です。
完全な(フォルダ+ファイル名)アドレス	現在、選択しているファイルの完全なパス+ファイル名です。
コントロールアドレス	関連のコントロールアドレスを設定し、ファイルブラウザ内のファイルを削除、またはリネームします。 コマンド：コントロールアドレス 0: 無 1: 削除 2: リネーム 3: 選択(cMT シリーズにのみサポートされます) 結果：コントロールアドレス +1 0: 成功 1 その他: エラー 新しいファイル名：コントロールアドレス +2

外見の属性の設定



設定	記述
ファイル位置	SDカードまたはUSBディスクからファイルを取得するのを選択します。
ファイルタイプ	表示されるファイルを選択します。全てのファイル、CSVファイルまたは画像ファイルを選択できます。
フォント、背景、色	オブジェクト及び文字の属性を設定します。

Note

- クリックされたファイルのファイル名及びパスは設定したアドレスに書き込みます。設定のアドレスを変更することによってファイルブラウザ内のファイル選択を変更させたい場合、ファイルの所在位置を入力してから、選択コマンド(=3)を入力すれば、ファイルブラウザをそのファイルに指向させることができます。
- HMIを再起動、または記憶装置に接続したばかりの際に、システムはフォルダパスアドレス及びファイル名アドレスを読み取り、そしてディレクトリを切り替えて、設定したアドレスが指向しているファイルを選択します。[フォルダパスアドレス]を使用しない場合、完全なアドレスを読み取ります。

▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

13.40.インポート/エクスポート

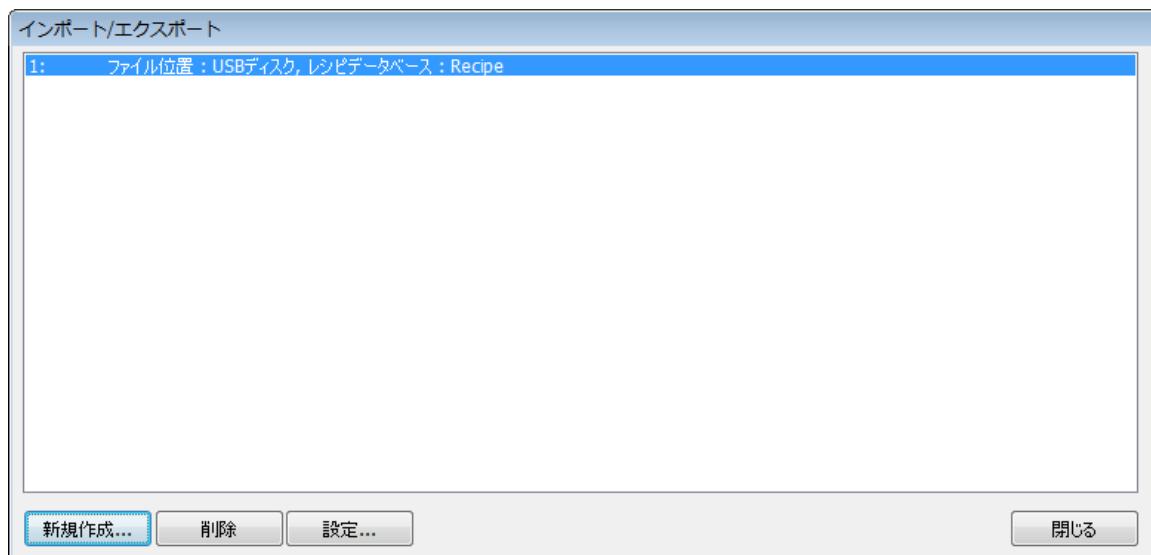
13.40.1. 概要

[インポート/エクスポート]オブジェクトは、レシピデータベース、或いはストリングテーブル内のレシピデータをインポート、またはエクスポートすることができます。

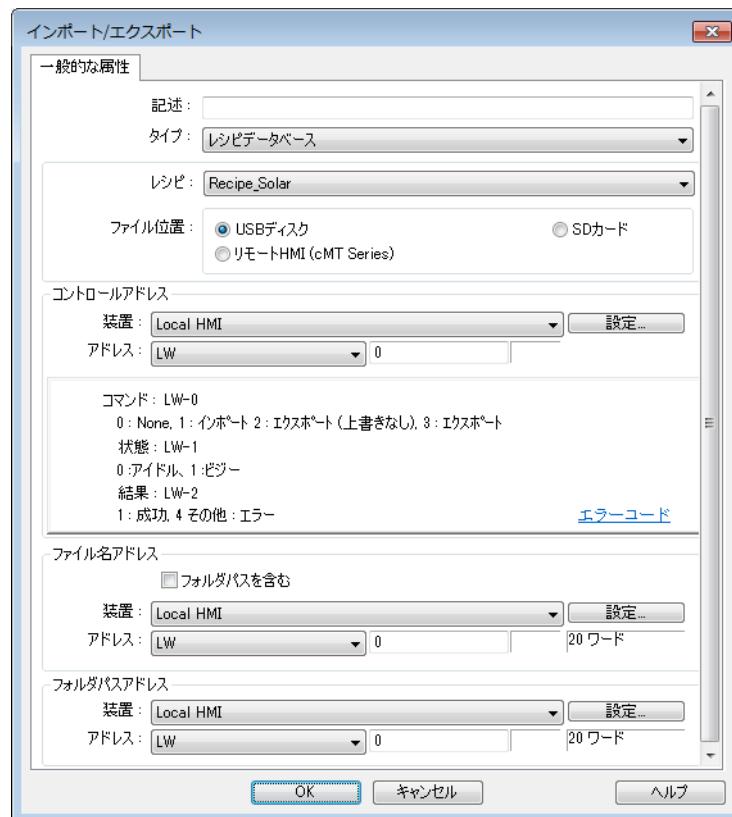
13.40.2. 設定



ツールバー上の[オブジェクト]»[インポート/エクスポート]アイコンを押せば、[インポート/エクスポート]オブジェクトの属性ダイアログボックスがオープンされ、続いて[新規作成]ボタンを押し、現れた[インポート/エクスポート]オブジェクトで正確に各属性を設定完了してOKボタンを押すと、[インポート/エクスポート]オブジェクトが新規作成されます。



一般的な属性の設定



設定	記述
タイプ	データソースを選択します。データベースまたはストリングテーブルが選択できます。
ファイル位置	SDカード、USBディスクまたはリモートHMI(cMTシリーズ)からインポート/エクスポートするのを選択します。ファイル位置をリモートHMIに設定した場合、cMTシリーズのファイルのインポートのみサポートします。
レシピ	使用するレシピを選択します。ストリングテーブルを選択した場合、本項は出てきません。
コントロールアドレス	関連のコントロールアドレスを設定し、インポート/エクスポート、及び実行結果の表示を実行します。 コントロール: コントロールアドレス レシピデータベース: 0: 動作しない None 1: インポート 2: エクスポート(上書きなし) 3: エクスポート

ストリングテーブル：

0: 準備完了
 1: 削除
 2: インポート
 4: エクスポート(上書き無し)
 5: エクスポート

状態: コントロールアドレス+1

0: アイドル(Idle)
 1: ビジー

結果: コントロールアドレス+2

1: 成功
 4: ファイルが既存している；上書き未実行
 その他: エラー

ファイル名アドレス	インポート/エクスポートに使用するファイル名を設定します。[フォルダパスを含む]にチェックマークを入れたら、このアドレスには完全なパス及びファイル名を含めます。
フォルダパスアドレス	インポート/エクスポートに使用するファイルが所在しているフォルダパスです。
リモート HMI アドレス	ファイル位置をリモート HMI (cMT シリーズ) に設定した場合、リモート HMI の IP を本欄に入力してください。

例 1

下記の例でレシピインポート/エクスポート機能を設定する方法を説明します。下表のように設定を入力します：

欄	設定値
ファイル位置	USB ディスク
レシピ	Recipe_A(または他のレシピ)
コントロールアドレス	LW-100
ファイル名アドレス	LW-200
フォルダパスアドレス	LW-250

- 2 個の[アスキーオブジェクト]を加入し、アドレスをそれぞれ LW-200 と LW-250 に設定します。

2. LW-200 でファイル名 : 2015_recipe.csv を入力します。
3. LW-250 でフォルダパス : Setting を入力します。
4. [マルチ状態設定]オブジェクトを使用し、数値 3 を LW-100 に書き込めば、レシピ Recipe_A を USB ディスクの Setting/2015_recipe.csv にインポートすることができます。

**Note**

- エクスポート(上書きなし)を実行する場合、もし指定のファイルが既存している場合、今回のエクスポートが取り消され、結果数値は 4 に設定されます。

下表では各エラーコード及びそれが代表する状態を記します：

エラーコード(HEX)	状態
0x1	実行に成功した
0x4	エクスポートしたところ、指定のファイルが既存している
0x10	無効なコマンド
0x100	欄には非数値データが含まれている
0x101	パスには“..”が含まれている
0x102	Recipe DB データを更新する時に、通信エラーが発生した
0x103	Project ファイルから Recipe DB データを読み取る時にエラーが発生した
0x200	未知の実行時エラー
0x201	未知の実行時内部エラー
0x202	未知のファイルフォーマット
0x203	Recipe DB の定義テーブル検証エラー
0x204	Recipe DB のデータテーブル検証エラー
0x205	Recipe DB の定義テーブル書き込みエラー
0x206	Recipe DB のデータテーブル書き込みエラー
0x300	ファイルエラー: 未知のエラー
0x301	ファイルエラー: ファイル名は空白
0x302	ファイルエラー: 空き領域が不足、または外部装置が存在しない
0x303	ファイルエラー: 不正なファイル名、或いはファイルの名前が既存のフォルダの名前と同じ
0x304	ファイルエラー: 指定したファイルが削除できない
0x305	ファイルエラー: ファイルオープンエラー
0x306	ファイルエラー: 判別できない BOM

0x307	CSV ファイル読み取りエラー(不正確なファイルフォーマット)
0x308	ファイルエラー: 外部装置の空き領域が不足
0x309	ファイルエラー: ファイルが見つからない
0x30A	ファイルエラー: CSV ファイル内のデータが 10000 行を超えた
0x400	未知のデータベースエラー
0x401	データベースエラー: テーブルが開けない
0x402	データベースエラー: データ列が開けない
0x403	DB とインポートした CSV ファイルの欄数が一致していなかった
0x501	リモート HMI への接続に失敗した
0x503	リモート HMI データベースはインポートを禁止する
0x504	リモート HMI データベースはインポートをサポートしない

 このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.41. 円グラフ

13.41.1. 概要

指定したワードアドレスにデータを入力することで、円グラフを作成し、各チャンネルが占めた比率を表示します。

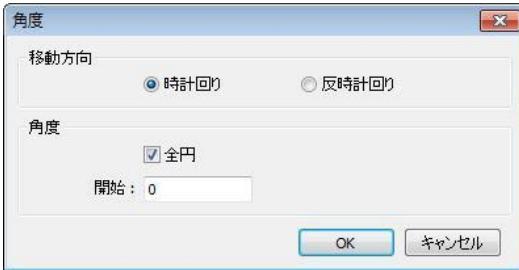
13.41.2. 設定



ツールバー上の[オブジェクト]»[円グラフ]ボタンを押せば、[円グラフ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[円グラフ]オブジェクトが新規作成されます。

一般的な属性の設定

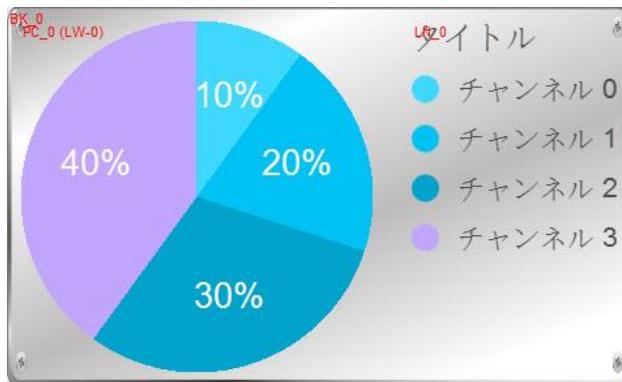


設定	記述
角度	円グラフの[開始角度]を設定し、作画方向を[時計回り]または[反時計回り]を決定します。 [全円]を選択していなかったら、[終了角度]を設定する必要があります。
	
円心	円心のサイズを設定します。
チャンネル数	閲覧したいチャンネルの数量を設定します。範囲: 2~16。
外枠色	円グラフ外枠の色を設定します。
データ表示	データが円グラフに表示するスタイルを設定します。選択肢には:[無]、[数値]、[百分比]があります。 円グラフで文字情報を表示する際に、表示させたい情報のフォント及びフォントサイズを設定することができます。 円グラフで[数値]を表示する際に、表示させたい小数点以下の桁数を設定することができます。
読み取りアドレス	ここでチャンネル1のアドレスを設定します。一つのチャンネルは一個のアドレスを表示し、それにアドレスは連続したものであること。もし読み取りアドレスを LW-0 に設定したら、チャンネル2の読み取りアドレスは LW-1 になり、チャンネル3の読み取りアドレスは LW-2 になる形で、このように類推します。
チャンネル	チャンネル番号を選択し、各チャンネルの表示スタイルを設定します。[フォント色]、[背景色]、[模様色]、[模様スタイル]が設定できます。ここでの[背景色]とは、図形スタイルの背景色のことを指し、選択したのは背景色がない図形であつたら、[背景色]を設定する必要がありません。

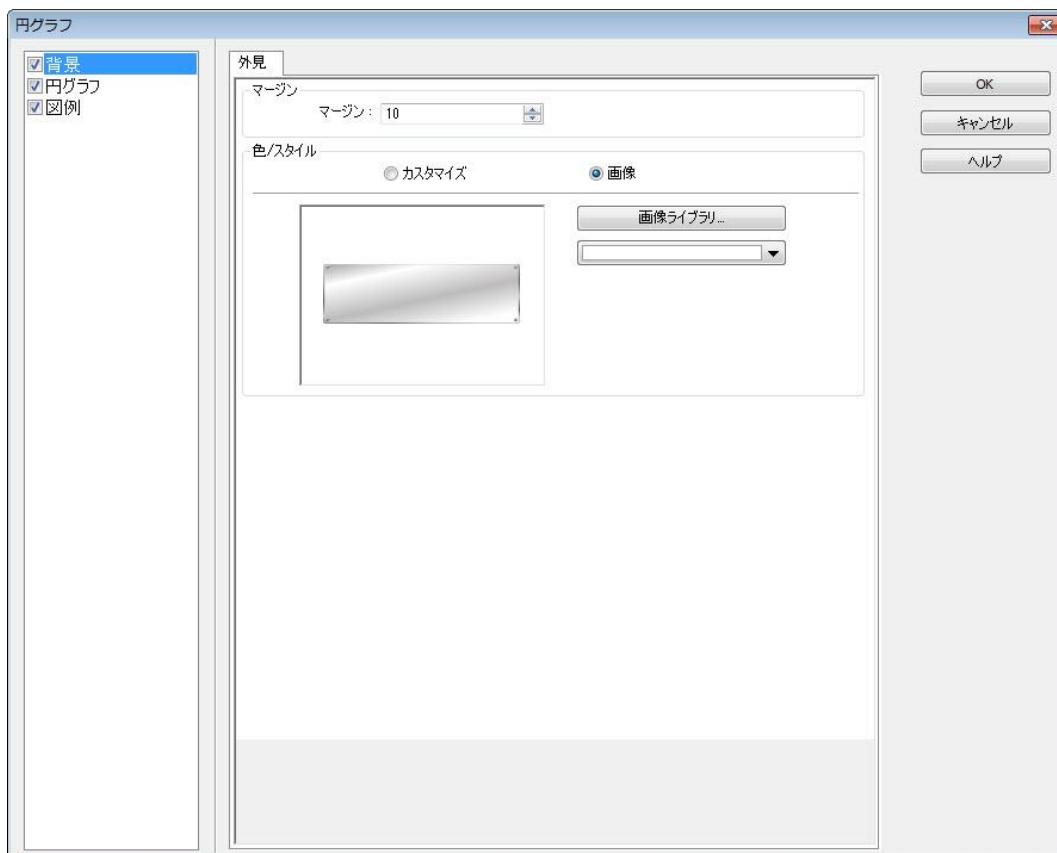
13.41.3. 複合型設定

cMT シリーズには、EasyBuilder Pro V5.07.01 からオブジェクトの複合型設定が追加されました。本機能を通じ、一回で関連する機能を設定することができます。円グラフに背景と図例を設定可能で、円グラフのデザインに活用して美化させることができます。

オブジェクト



背景の設定



設定

記述

マージン

背景の縁とオブジェクトの間の距離を設定します。

色/スタイル

カスタマイズ



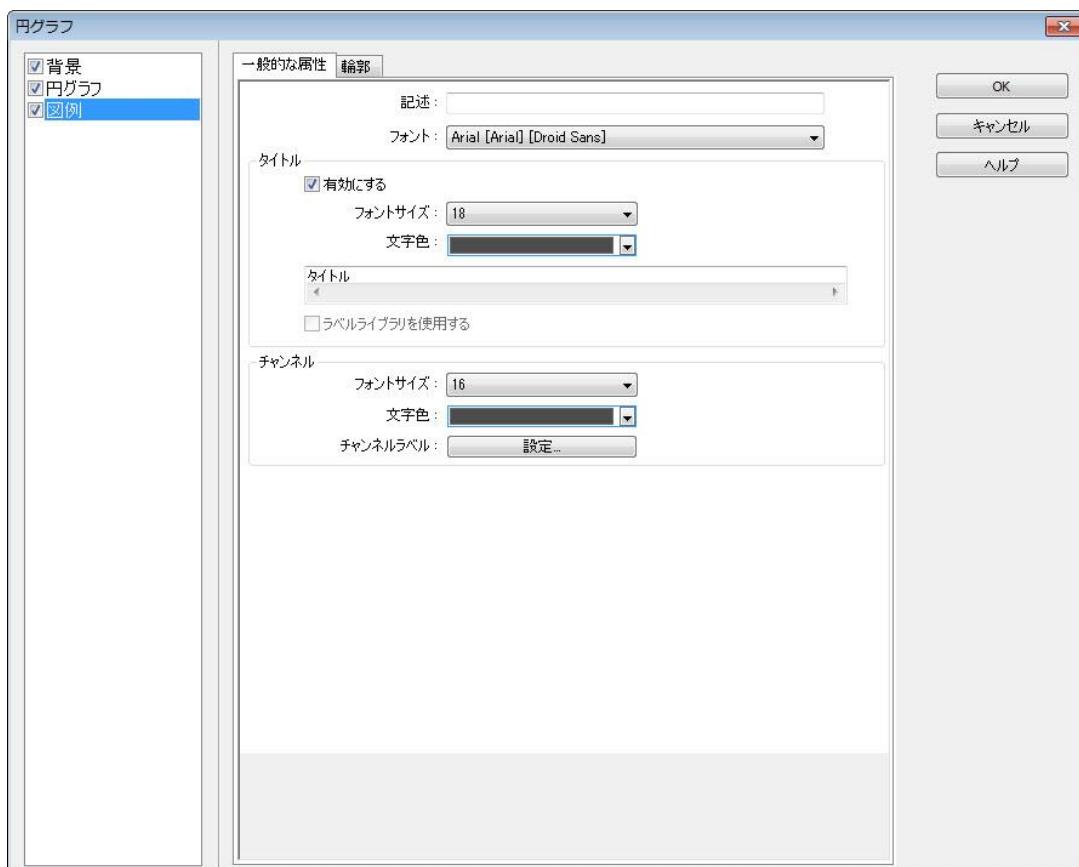
背景の色と模様スタイルを選択します。

画像



内蔵の背景画像または画像ライブラリ内のファイルを使用します。

図例の設定



設定

記述

タイトル

タイトルを表示するかを選択できます。ラベルライブラリを使用可能です。

チャンネル

チャンネルが表示する名前を設定します。ラベルライブラリを使用する時、チャンネル数はちょうど状態数を対応します。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.42. バーコード表示

13.42.1. 二次元バーコード表示

13.42.1.1. 概要

情報を指定したワードアドレスに入力したら、情報に対応する二次元バーコードがウインドウで生じ、ユーザーに読み取らせます。

13.42.1.2. 設定



ツールバー上の[オブジェクト]»[二次元バーコード表示]ボタンを押せば、[二次元バーコード表示]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[二次元バーコード表示]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述
モード	サポートするバーコードタイプ: QR コード、Aztec コード

訂正レベル

二次元バーコードには誤り訂正能力を有し、バーコード図形に破損があっても、依然に読み取られます。

QR コード：

訂正レベルは 4 種類があり：L、M、Q、H です。下記の説明をご参照ください：

訂正レベル	
L	7%のコード文字が訂正される
M	15%のコード文字が訂正される
Q	25%のコード文字が訂正される
H	30%のコード文字が訂正される

Aztec コード

エラー訂正レベルは比率によって計算されます。調整可能の範囲は 5%~95%です。正確さを確保するため、訂正レベルをより高く設定すれば、バーコードのサイズもより大きいである必要があります。

色

二次元バーコードを表示する色を選択します。

Unicode

デフォルト設定では、二次元バーコードの内容は ASCII エンコードで生成されます。本項を選択すると、二次元バーコードの内容は Unicode のエンコードで生成されます。

デフォルトの ASCII エンコードは一般的なアルファベットと数字形式の内容に適しています。Unicode を選択すると、Unicode に対応する他の文字に適用します。例えば中国語、韓国語などです。

読み取りアドレス

情報を指定したワードアドレスに入力したら、情報に対応する二次元バーコードがウインドウで生じ、ユーザーに読み取らせます。情報の長さは 1~1024 ワードです。

13.42.2. WeChat バーコード表示

13.42.2.1. 概要

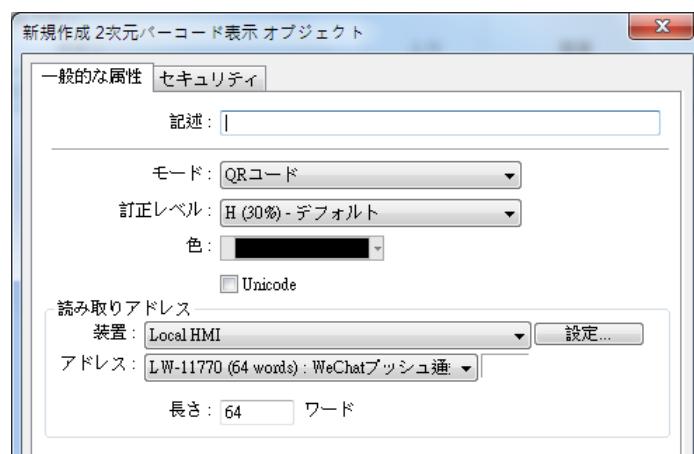
[WeChat 二次元バーコード表示]オブジェクトは二次元バーコードを通じて WeChat アカウントに接続した後、当該 HMI での全てのトリガーされたイベントをプッシュ通知で携帯電話に発送します。

13.42.2.2. 設定



ツールバー上の[オブジェクト] » [バーコード] » [WeChat 二次元バーコード表示]ボタンを押せば、[WeChat 二次元バーコード表示]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[WeChat 二次元バーコード表示]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述										
モード	サポートするバーコードタイプ : QR コード、Aztec コード										
訂正レベル	二次元バーコードには誤り訂正能力を有し、バーコード図形に破損があっても、依然に読み取られます。 QR コード 訂正レベルは 4 種類があり : L、M、Q、H です。下記の説明をご参照ください :										
	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">訂正レベル</th> </tr> </thead> <tbody> <tr> <td>L</td><td>7% のコード文字が訂正される</td></tr> <tr> <td>M</td><td>15% のコード文字が訂正される</td></tr> <tr> <td>Q</td><td>25% のコード文字が訂正される</td></tr> <tr> <td>H</td><td>30% のコード文字が訂正される</td></tr> </tbody> </table>	訂正レベル		L	7% のコード文字が訂正される	M	15% のコード文字が訂正される	Q	25% のコード文字が訂正される	H	30% のコード文字が訂正される
訂正レベル											
L	7% のコード文字が訂正される										
M	15% のコード文字が訂正される										
Q	25% のコード文字が訂正される										
H	30% のコード文字が訂正される										
	Aztec コード エラー訂正レベルは比率によって計算されます。調整可能の範囲は 5%~95% です。正確さを確保するため、訂正レベルをより高く設定すれば、バーコードのサイズもより大きいである必要があります。										
色	二次元バーコードを表示する色を選択します。										
UNICODE	デフォルト設定では、二次元バーコードの内容は ASCII エンコードで生成されます。本項を選択すると、二次元バーコード										

の内容は Unicode のエンコードで生成されます。

デフォルトの ASCII エンコードは一般的なアルファベットと数字形式の内容に適しています。Unicode を選択すると、Unicode に対応する他の文字に適用します。例えば中国語、韓国語などです。

読み取りアドレス	情報を指定したワードアドレスに入力したら、情報に対応する二次元バーコードがウインドウで生じ、ユーザーに読み取らせます。情報の長さは 1~1024 ワードです。
----------	---

システムパラメータ設定



設定

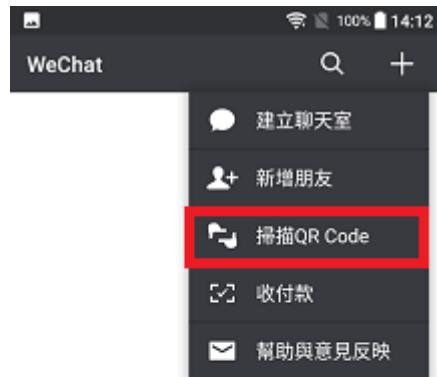
記述

EasyAccess サーバー	現在は中国のサーバーのみ WeChat のプッシュ通知機能をサポートします。
-----------------	--

WeChat 設定

ファイルを HMI にダウンロードした後、二次元コードが表示され、そして HMI のプッシュ通知を WeChat アカウントに登録します。以下は登録手順です。

1. WeChat のインターフェースで HMI の QR Code をスキャンします。



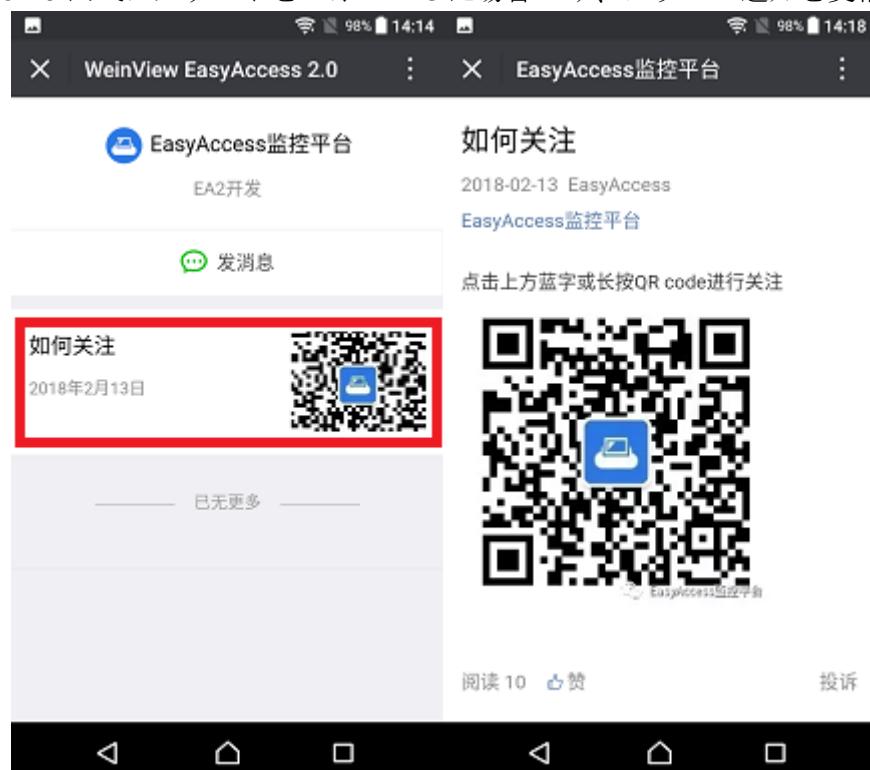
2. スキャンした後、設定を開始します(初めて使用する場合、WeChat 権限がリクエストされます)。



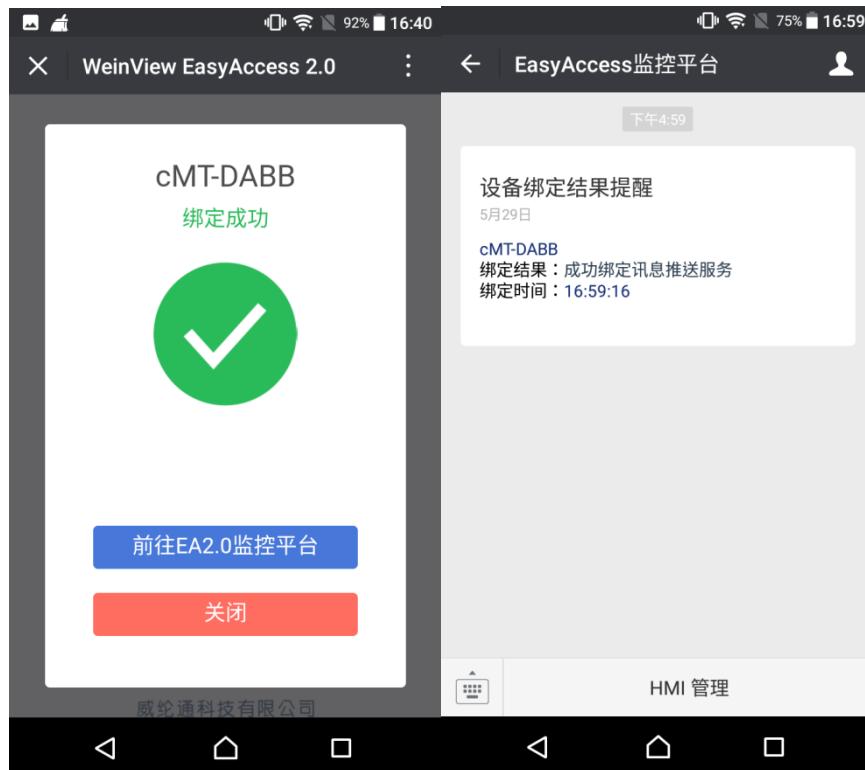
3. EasyAccess 2.0 公式アカウントをフォローしていない場合、フォローのリクエストが出て、HMI を固定するため、フォローしてください。



4. EasyAccess 2.0 公式アカウントをフォローした場合のみ、プッシュ通知を受信できます。



5. フォローに成功したら、ステップ 1 及びステップ 2 を繰り返すれば、HMI への固定に成功します。



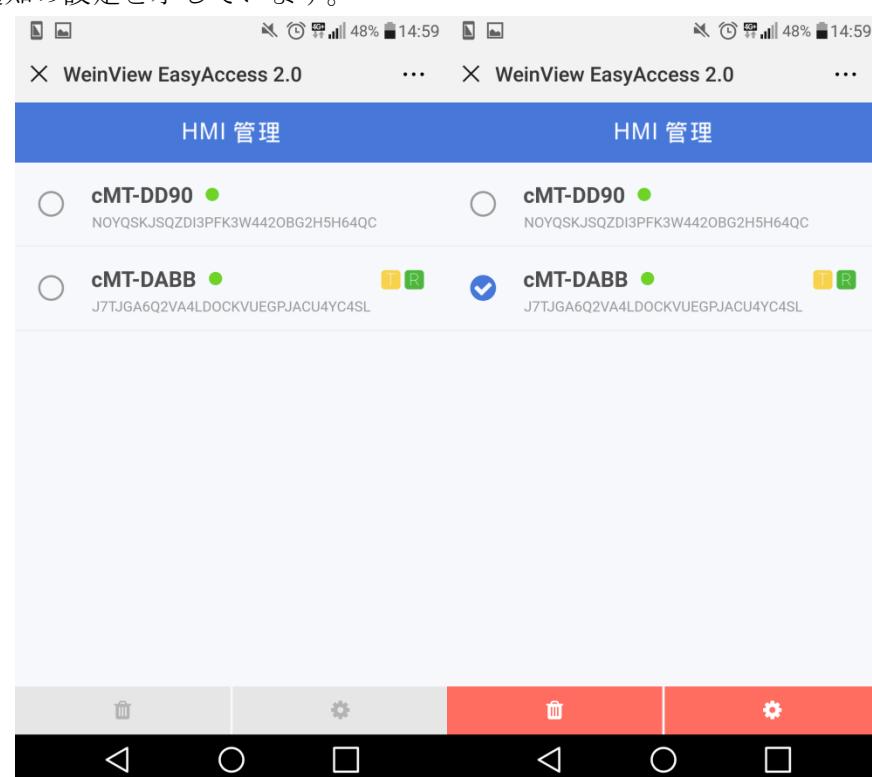
6. EA2.0 監視プラットフォームに行き、アカウントをログインします。



7. これで WeChat で当該 HMI からのプッシュ通知を受信できます。



8. 下の HMI 管理をクリックすれば、管理ページに入ることができます。HMI 名前の後にある緑の点は HMI がオンライン中と示し、灰色の点は HMI がオフライン中と示しています。一台の HMI を選択した後、左下のアイコンは固定を解除することと示し、右下のアイコンはプッシュ通知の設定と示しています。



9. HMI への固定を解除したい場合、[HMI 管理]で HMI を削除できます。



10. プッシュ通知設定で自由に受信したいプッシュ通知を選択できます。



Note

- 本機能を使用する前、HMI の EasyAccess 2.0 サービスを中国サーバーにアクティベートしてオンラインした場合のみ、プッシュ通知の送信に成功します。QR Code をスキャンする時

に HMI がまだアクティベートされていなかったら、以下のような画面が出ます。

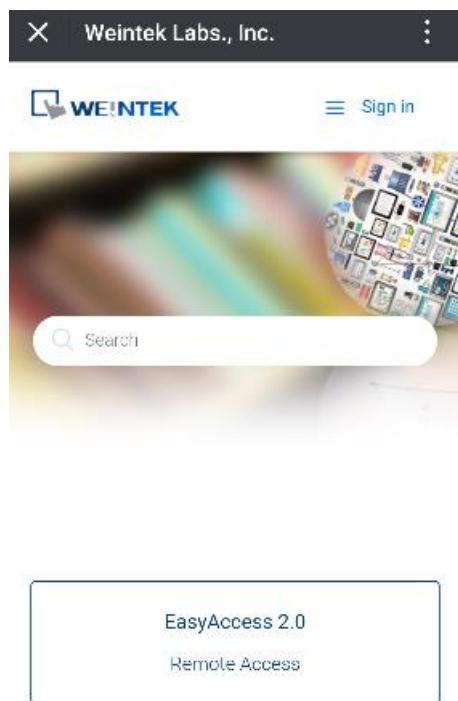


- HMI が既に WeChat アカウントに固定された場合、QR コードをスキャンすれば、以下のよ
- うな画面が出ます。



- 現在は中国サーバーのみ WeChat のプッシュ通知機能をサポートします。グローバルサー
- バーアクティベーションされた HMI が QR コードをスキャンすると、ユーザーマニュア

ルにリンクされます [https://weintek.zendesk.com/。](https://weintek.zendesk.com/)



13.42.3. バーコードスキャナー(Android カメラ)

13.42.3.1. 概要

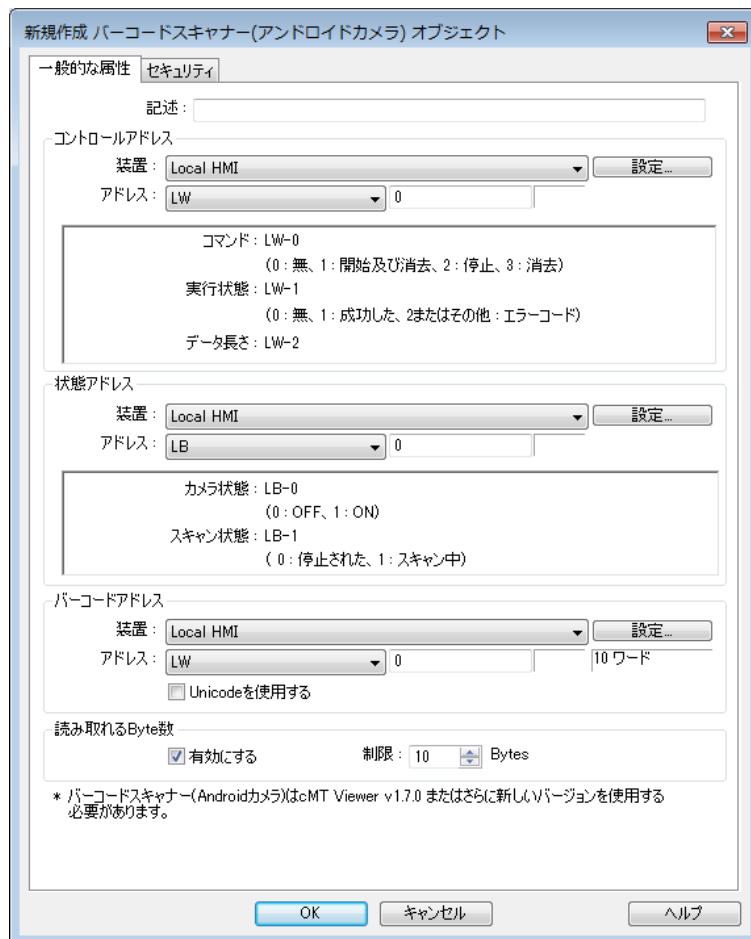
一部のモデルの HMI は、Android 装置上の cMT Viewer を cMT 装置に接続した前提で、Android 装置(スマートフォン/タブレット)のカメラを通じて、一次元及び二次元バーコードをスキャンすることができます。

13.42.3.2. 設定



ツールバー上の[オブジェクト]»[バーコード]ボタンを押し、または[オブジェクト]»[バーコード]をクリックして[バーコードスキャナー]オブジェクト設定ダイアログボックスをがオープンし、正確に各属性を設定完了して OK ボタンを押すと、[バーコードスキャナー]オブジェクトが新規作成されます。

一般的な属性の設定

設定記述

コントロールアドレス : コマンドを出すアドレス

0: 無

1: 開始及び消去

2: 停止

3: 消去

コントロールアドレス +1 : 実行状態を表示する

0: 無

1: 成功した

2 或いはその他: エラーコード

コントロールアドレス +2 : データの長さを表示する

状態アドレス : カメラが On か Off のを表示する

0: Off

1: On

状態アドレス +1 : スキャンの準備ができたかを表示する

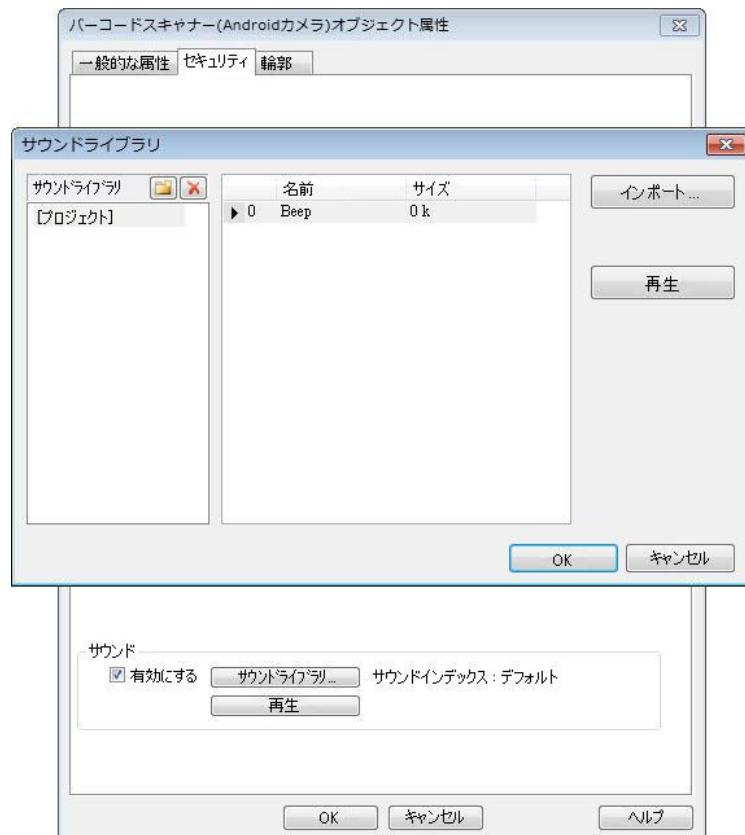
0: 停止された

1: スキャン中

バーコードアドレス : バーコードデータを保存するアドレスです。Unicode が使

ドレス	用可能です。
読み取れる最大 Byte 数	読み取ったデータの大きさが本設定を超えると、実行状態の値は 2(エラーコード)に変わります。

セキュリティ設定



設定

記述

サウンド	結果を読み込んだ時に音声を出すのか設定します。.wav の音声ファイルをサポートします。
-------------	--

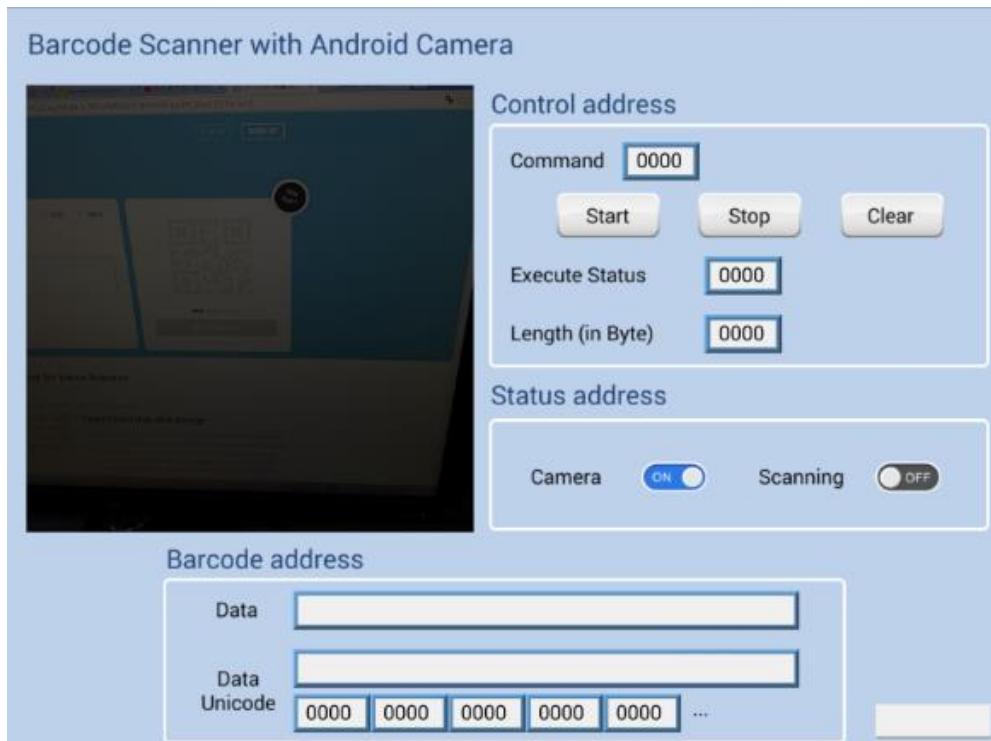
Note

- バーコードスキャナーはシミュレーション或いは cMT Viewer でオープンすることができます。
- EAN/UPC, Code 128, Code 39, Interleaved 2 of 5 and QR Code をサポートします。
- モバイルデバイスの上で、他のアプリケーションがカメラを使用、またはロックする場合、cMT Viewer が正しく動作しないかもしれません。
- プロジェクトをデザインする時、複数の cMT Viewer に接続した場合、アドレスは共有されたので、各 cMT Viewer が同じのバーコードスキャナーのあるウインドウを表示する際に、同時にスキャンを行います。

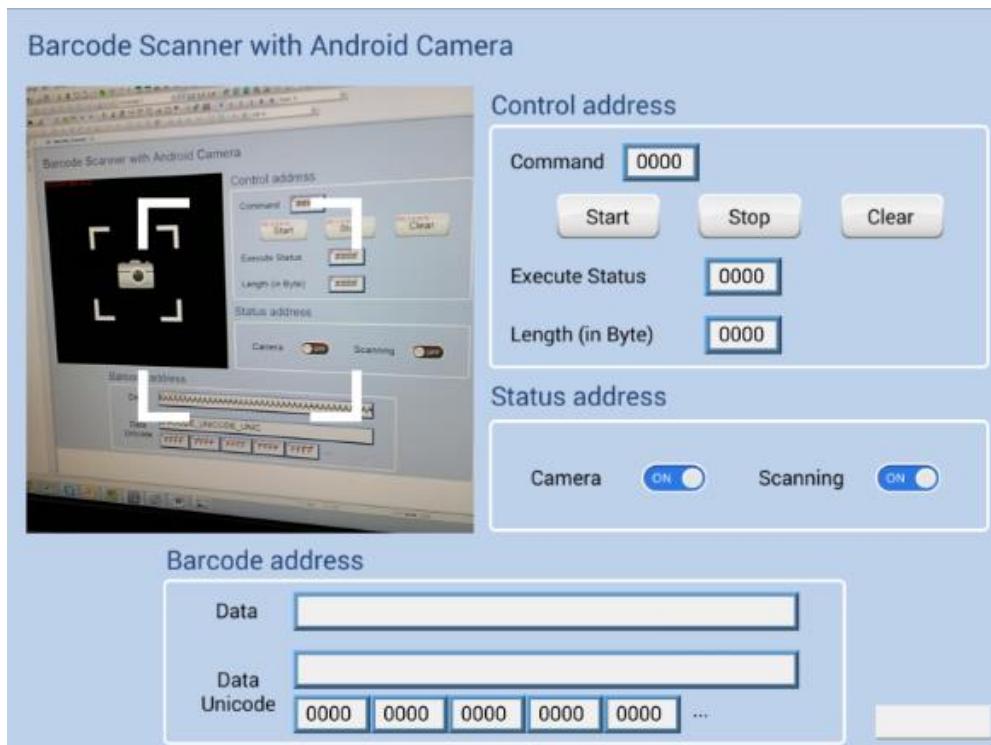
例

下記の例で実際にタブレットで QR コードをスキャンする過程を見せます。

- 最初の頃、画面は暗めの状態にあります。

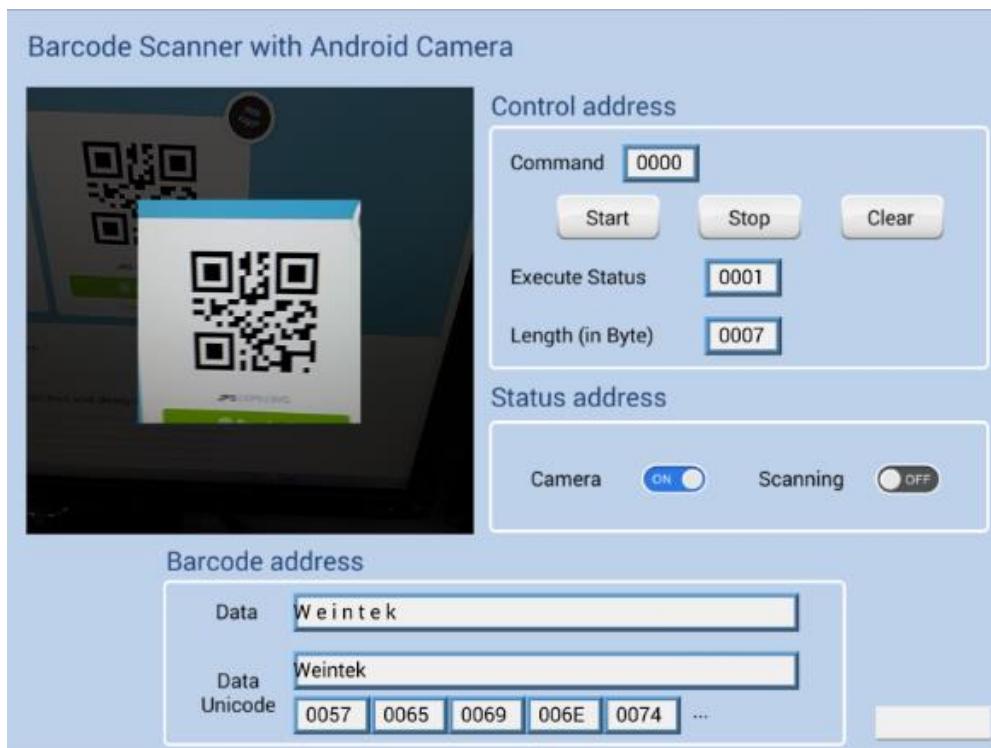


- Start ボタンを押すと、Scanning 状態がオープンされ、画面が明るくなったらスキャンが始まられます。



- データの読み取りに成功した後(実行状態は 1)、QR コードがキャプチャされ、内容は Barcode

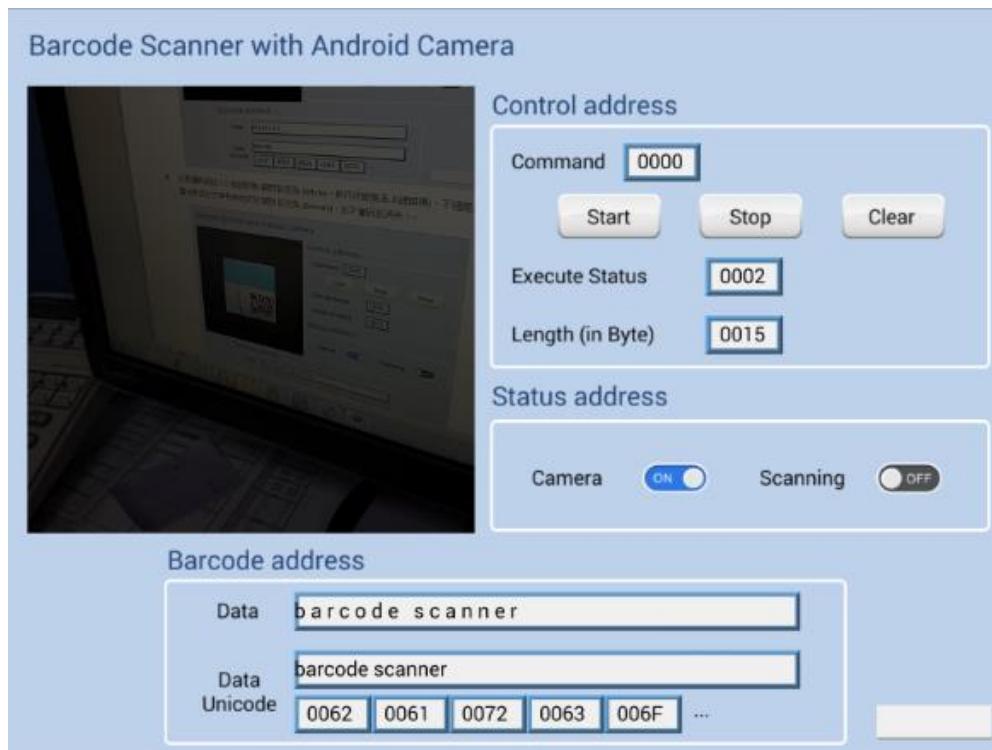
address(バーコードアドレス)に表示されます。また、Unicodeも使用可能です。



- 読み取った情報のサイズが制限(例 10 bytes)を超えた場合、実行状態は 2(エラーコード)になります、しかし、制限を超えた部分の表示はアスキーオブジェクトの設定によります(例では 20 words)。制限を超えたので表示が消されることはありません。



- ページが切り替えられたら、Scanning は Off になり、パラメータは次回で開始(Start)または消去(Clear)ボタンが押されたまで、前回のパラメータを保存し続けます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.43. ストリングテーブル

13.43.1. 概要

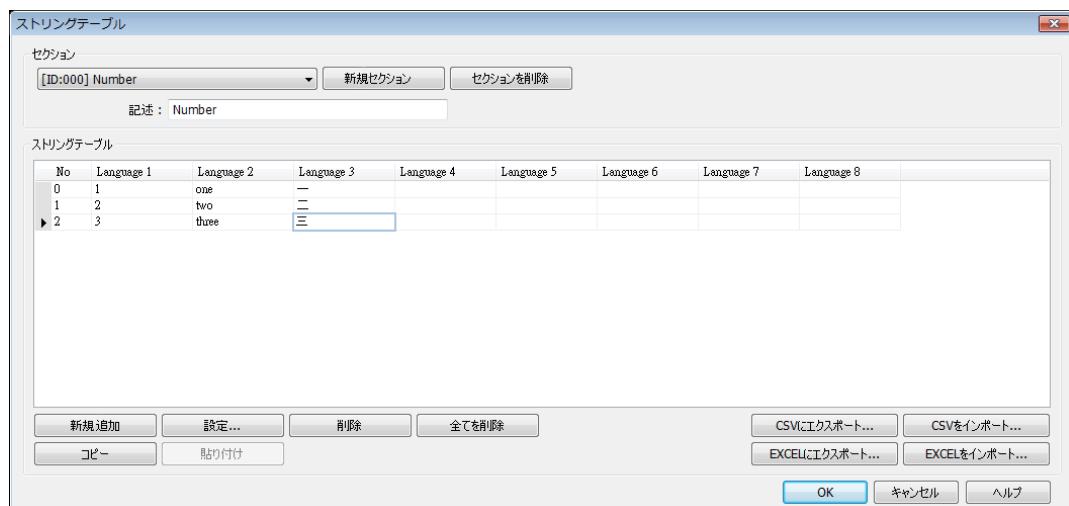
ストリングテーブルを通じれば、HMI の運転中で、デフォルト番号により動的に文字を調整することができます。多言語環境にも対応します。

13.43.2. 設定



[プロジェクト] » [ストリングテーブル] をクリックし、正確に各属性を設定完了して OK ボタンを押すと、[ストリングテーブル] オブジェクトが新規作成されます。

一般的な属性の設定



設定	説明
セクション	ストリングテーブルのリストです。 [新規セクション] : 新しいストリングテーブルを作成します。 [セクションを削除] : 選択したストリングテーブルを削除します。
新規追加	ストリングをストリングテーブルに追加します。
設定	選択したストリングの内容を選択します。
CSV ファイルにエクスポート	全てのストリングテーブルを.csv フォーマットのファイルに保存します。
CSV ファイルをインポート	既存の.csv ファイルをストリングテーブルにインポートします。

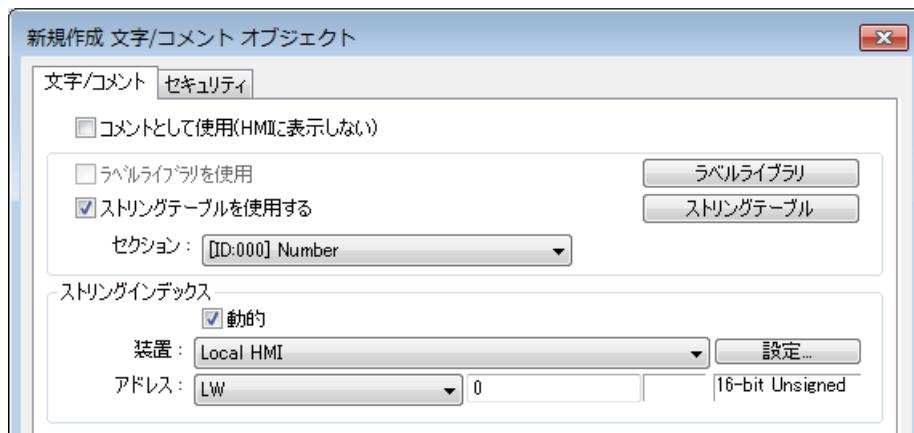
ポート	一トします。
EXCEL ファイルにエクスポート	全てのストリングテーブルを.xls フォーマットのファイルに保存します。
EXCEL をインポート	既存の.xls ファイルをストリングテーブルにインポートします。

Note

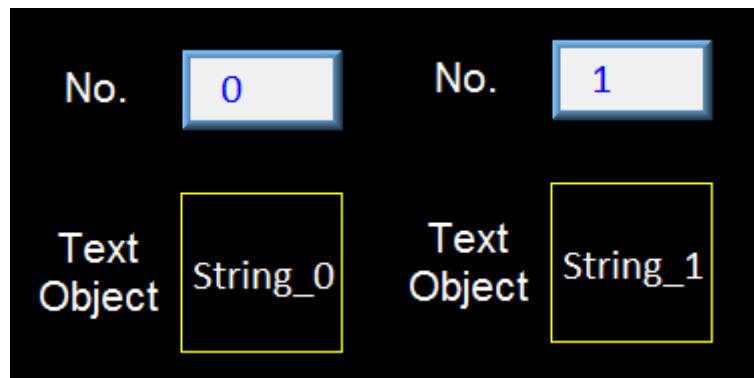
- ストリングテーブル内の多言語フォントは、[ラベルライブラリ]に設定する必要があります。
- 全てのテーブル内のストリング総数上限は 10000 です。

例 1

1. 上図に示す通り、新規ストリングテーブルを作成します。
2. [文字]オブジェクトを新規作成し、[ストリングテーブルを使用する]を選択します。ストリング ID を[動的]に、読み取りアドレスを LW-0 に設定します。



3. [数値]オブジェクトを新規作成し、アドレスを LW-0 に設定します。
4. LW-0=0 になる場合、番号 0 のストリングが表示されます。LW-0=1 の場合、番号 1 のストリングが表示されます。



13.44. データベース

13.44.1. データベースサーバー

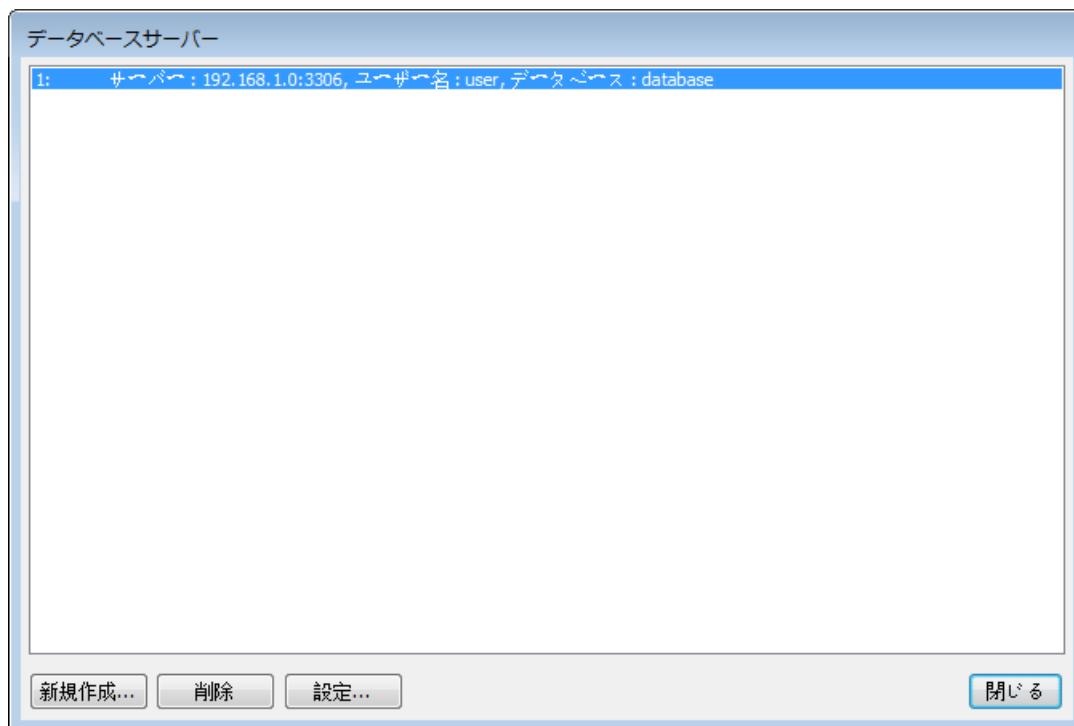
13.44.1.1. 概要

[データベースサーバー]オブジェクトは MySQL または MySQL データベースに接続することができます。ユーザーは[データベースサーバー]を通じて、データサンプリング/イベントログをデータベースに転送し、或いは SQL クエリ機能を合わせてデータベース内のデータをアクセスすることができます。

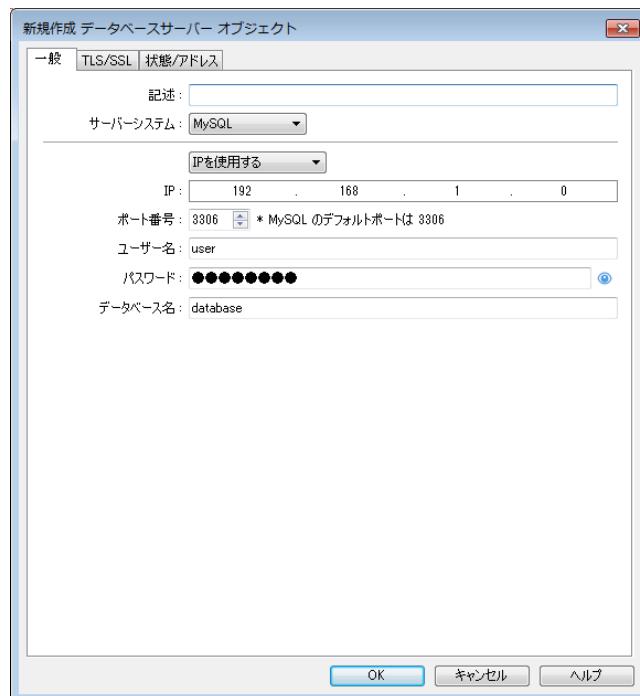
13.44.1.2. 設定



直接に[データベースサーバー]のアイコンをクリックし、本オブジェクトを作成してください。またはツールバーの[データ/履歴] » [データベースサーバー]をクリックし、オブジェクトを新規作成します。

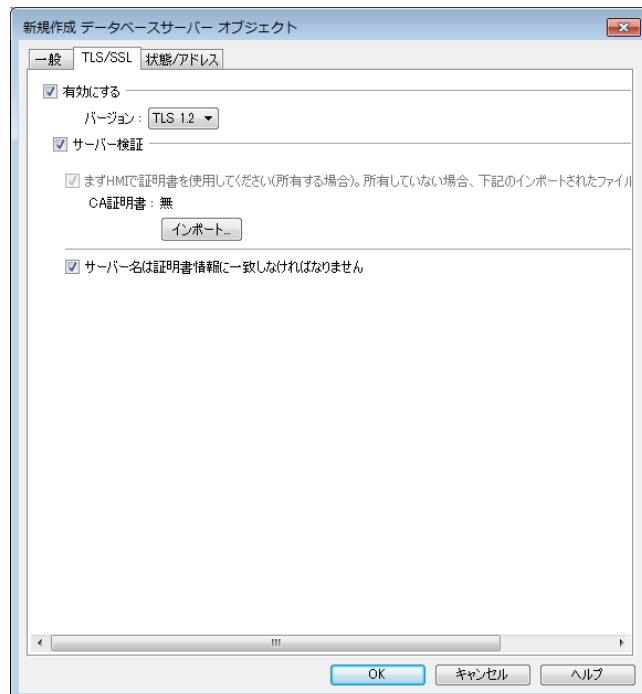


一般的な属性の設定



設定	記述
サーバーシステム	MySQL と MS SQL をサポートします。
IP	データベースの IP アドレスを入力します。
ドメイン名を使用する	<p>ドメイン名を使用する ▾</p> <p>ドメイン名: 192.168.1.0</p> <p>ドメイン名でサーバーを指定するのをサポートします。</p>
サーバー名を使用する	<p>サーバー名を使用する ▾ e.g. HOST\SQL EXPRESS</p> <p>サーバー名: PC_NAME\INSTANCE_NAME</p> <p>本機能はサーバーシステムに MS SQL サーバーを選択した場合のみ使用できます。Instance は通信ポート番号のことを示し、これを用いて一台の PC の上で複数のデータベースサーバーを判別できます。サーバー名は以下のフォーマットを使用できます：</p> <ul style="list-style-type: none"> ● <Computer name>\<Instance name> ● <Computer name> (デフォルト Instance : MSSQLSERVER に接続する) ● <IP address>\<Instance name> ● <IP address> (デフォルト Instance : MSSQLSERVER に接続する)
ポート番号	データベースのポート番号を入力します。
ユーザー名	データベースに接続用のユーザー名を入力します。

	長さは最大 32 ワードです。
パスワード	データベースに接続用のパスワードを入力します。 長さは最大 32 ワードです。
データベース名	履歴データを収集するデータベースの名前を入力します。

TLS/SSL

設定	記述
バージョン	TLS/SSL セキュリティ転送機構を使用します。手動で暗号化 TLS 1.0、TLS 1.1 または TLS 1.2 を選択できます。
サーバー検証	本項にチェックマークを入れると、当該サーバーが認証局(CA)に認証済みであるかのを検証します。サーバーの証明書は接続する時にサーバーから送信されます。 サーバー名は証明書情報に一致しなければなりません 当該ドメイン名または IP が証明書内の記載に一致であるかを検証します。ドメイン名と IP は証明書内の Subject Alternative Name に記載されています。

状態/コントロール設定

設定	記述																						
状態アドレス	<p>LW-n: [データベースサーバー]の接続状態を表示する</p> <table border="1"> <thead> <tr> <th>数値</th><th>記述</th></tr> </thead> <tbody> <tr> <td>0</td><td>データベースとの接続が停止された</td></tr> <tr> <td>1</td><td>データベースとの接続ができない</td></tr> <tr> <td>2</td><td>データベースとの接続に成功した</td></tr> </tbody> </table> <p>LW-n+1: エラー提示</p> <table border="1"> <thead> <tr> <th>数値</th><th>記述</th></tr> </thead> <tbody> <tr> <td>0</td><td>エラー無し</td></tr> <tr> <td>1</td><td>未知のエラー</td></tr> <tr> <td>2</td><td>データベースへの接続に失敗した</td></tr> <tr> <td>3</td><td>権限が不足でデータベースへの接続が拒否された</td></tr> <tr> <td>4</td><td>データベース名が間違った</td></tr> <tr> <td>5</td><td>無効なドメイン名</td></tr> </tbody> </table>	数値	記述	0	データベースとの接続が停止された	1	データベースとの接続ができない	2	データベースとの接続に成功した	数値	記述	0	エラー無し	1	未知のエラー	2	データベースへの接続に失敗した	3	権限が不足でデータベースへの接続が拒否された	4	データベース名が間違った	5	無効なドメイン名
数値	記述																						
0	データベースとの接続が停止された																						
1	データベースとの接続ができない																						
2	データベースとの接続に成功した																						
数値	記述																						
0	エラー無し																						
1	未知のエラー																						
2	データベースへの接続に失敗した																						
3	権限が不足でデータベースへの接続が拒否された																						
4	データベース名が間違った																						
5	無効なドメイン名																						
コントロールアドレス	<p>LW-n: [データベースサーバー]の実行または停止をコントロールする</p> <table border="1"> <thead> <tr> <th>数値</th><th>記述</th></tr> </thead> <tbody> <tr> <td>0</td><td>準備完了</td></tr> <tr> <td>1</td><td>開始</td></tr> <tr> <td>2</td><td>停止</td></tr> <tr> <td>3</td><td>更新</td></tr> </tbody> </table> <p>LW-n+1: データベースの IP アドレス</p> <p>LW-n+5: データベースの接続ポート番号を設定する</p> <p>LW-n+6: データベース接続用のユーザー名を設定する</p> <p>LW-n+22: データベース接続用のパスワードを設定する</p> <p>LW-n+38: 履歴データを収集するデータベースの名前を設定する</p>	数値	記述	0	準備完了	1	開始	2	停止	3	更新												
数値	記述																						
0	準備完了																						
1	開始																						
2	停止																						
3	更新																						

備考 :

- データサンプリングが成功に SQL データベースに同期した場合、データベースは 3 つのテーブルを生成します。サンプリングデータは*_data に保存します。

生成するテーブル	記述
<HMI NAME>_<DATALOG NAME>_data	サンプリングデータを保存する
<HMI NAME>_<DATALOG NAME>_data_format	システムに使用される

<i><HMI NAME> <DATALOG NAME> _data_section</i>	システムに使用される
--	------------

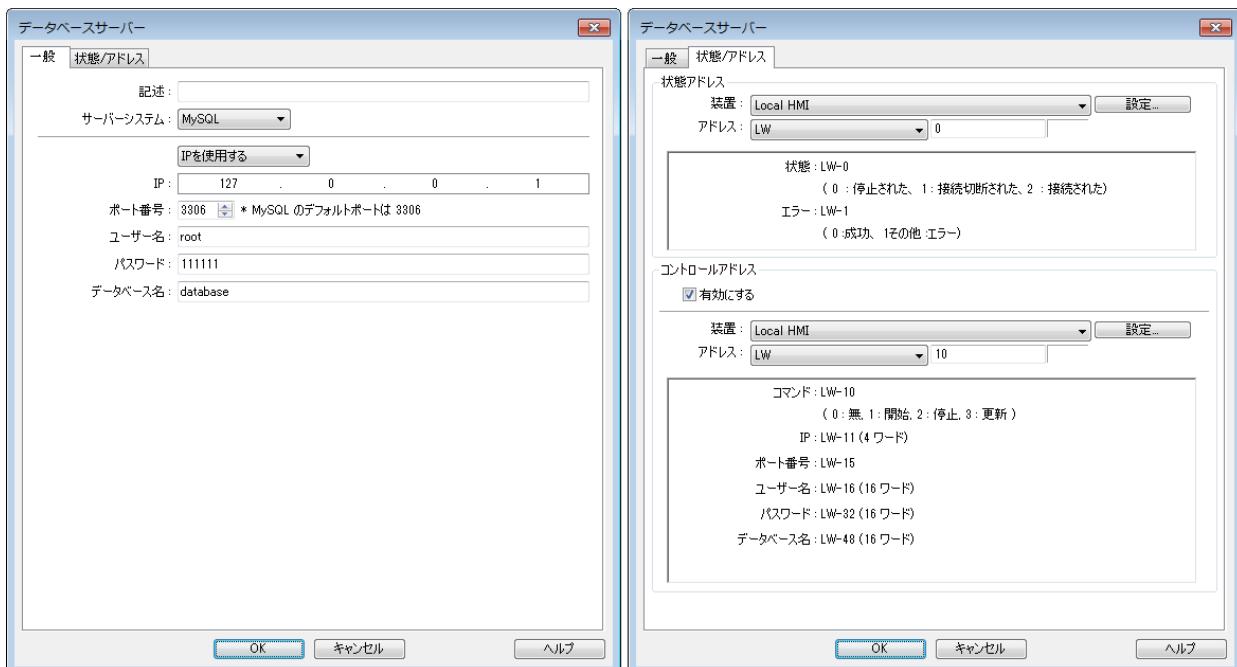
- イベントログを使用すると、データベースは下記 3 つのテーブルを生成します。イベントログは*_event に保存します。

生成するテーブル	記述
<i><HMI NAME>_event</i>	イベントログデータを保存する
<i><HMI NAME>_event_log</i>	システムに使用される
<i><HMI NAME>_event_update_time</i>	システムに使用される

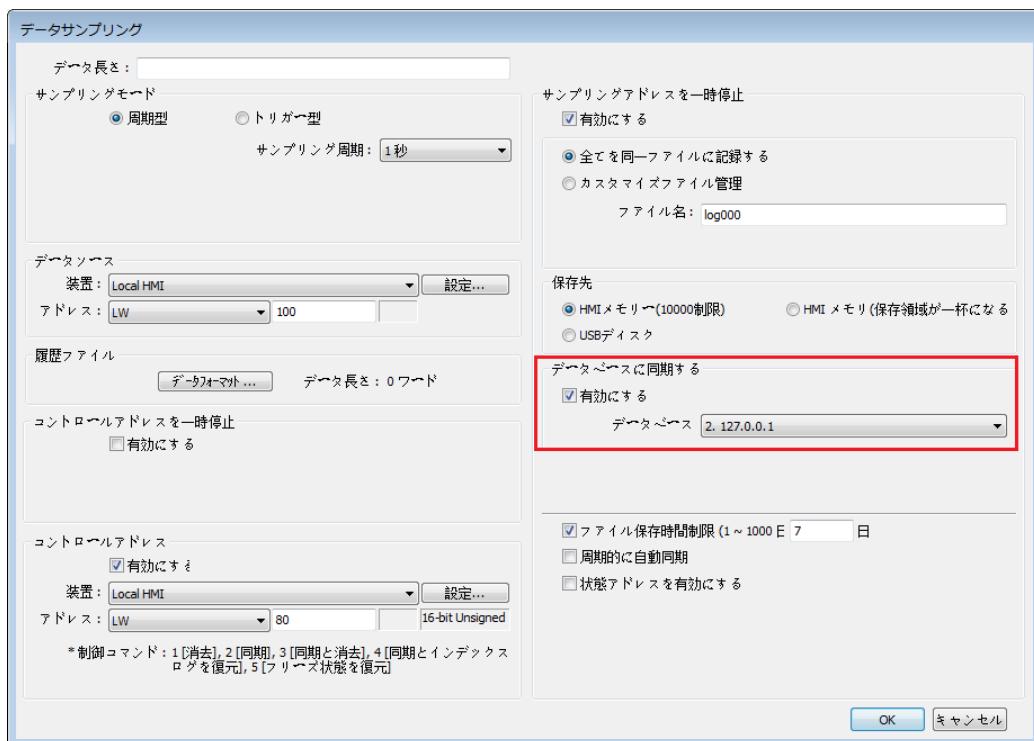
- 再度にデータサンプリング/イベントログの内容を変更する場合、例えばデータフォーマットまたはイベントログの内容を変更し、HMI にダウンロードする場合、先に SQL データベースで作成された 3 つの同期テーブルを削除してから、新しいデータサンプリング/イベントログが有効になります。

例 1

- [データベースサーバー]オブジェクトを新規作成し、状態アドレスを LW-0 に設定し、コントロールアドレスを LW-10 に設定します。



- データサンプリングオブジェクトを作成し、履歴データの欄に[データベースに同期する]を有効にし、そして[コントロールアドレス]を LW-80 に設定し、HMI 履歴データの同期と消去をコントロールします。



3. データベースが正確に接続してあれば、状態アドレス LW-0 は 2 を表示し(接続に成功した)、エラー アドレス LW-1 は 0 を表示します(エラー無し)。
4. LW-80 に 2 を書き込みます(データを同期する)。そして SQL データベースをオープンすれば、<HMI NAME>_<DATALOG NAME>_data テーブルにはデータが見られます。

Table	Action	Records	Type	Collation	Size	Overhead
hostname_datalog_data		6	MyISAM	utf8_unicode_ci	2.1 Kib	-
hostname_datalog_data_format		1	MyISAM	utf8_unicode_ci	2.0 Kib	-
hostname_datalog_data_section		0	MyISAM	utf8_unicode_ci	1.0 Kib	-
3 table(s)		Sum	7	MyISAM	utf8_unicode_ci	5.2 Kib
						0 B

このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.44.2. SQL クエリ

13.44.2.1. 概要

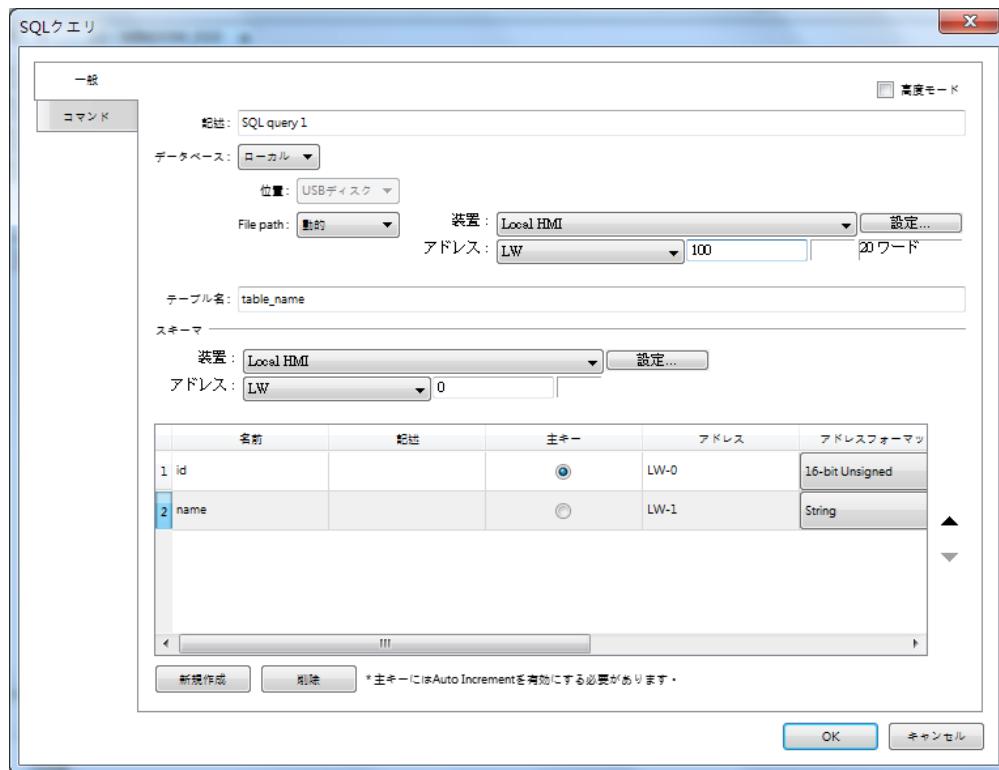
[SQL クエリ]機能は SQL のデータベースとデータを交換することができます。[SQL クエリ]機能を使用するには、まず[データベースサーバー]を設定する必要があります。

13.44.2.2. 設定



ツールバー上の[オブジェクト]»[データ/履歴]タブをクリックし、[SQL クエリ]ボタンを押せば、[SQL クエリ]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[SQL クエリ]オブジェクトが新規作成されます。

一般的な属性の設定



設定

記述

高度モード

[高度モード]を選択しない

[新規追加]をクリックすれば1行を新規追加することができ、または[サーバーからインポートする]を選択し、データベースサーバーから直接にすべての行をインポートできます。

[高度モード]を選択する

高度モードを選択すると、[コマンド]タブで手動でシンタックスを入力して MySQL/MS SQL データベースを制御する必要があります。一度高度モードを選択すると、一般モードに戻ることができないのでご留意ください。

記述

ユーザーは本クエリに関連情報を記入することができます。

データベース

データベースのソースを選択します。

ローカル: ソースは HMI に接続された USB メモリーまたは SD カード内の SQLite データベースです。

このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

リモート: ソースは[データベースサーバー]オブジェクト内に設定されたデータベースです。

ファイルパス

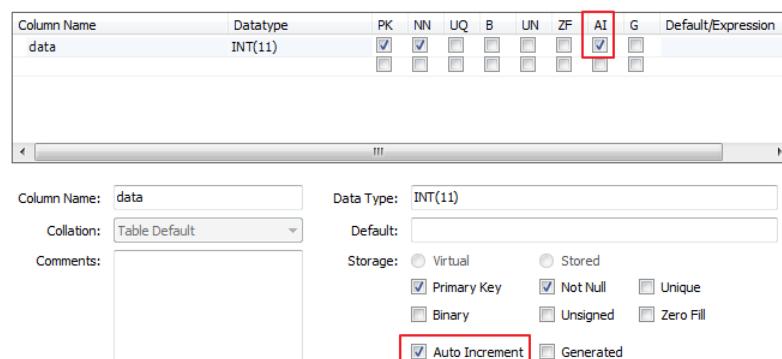
ローカルのデータベースの使用を選択する場合、[静的]パス、また

は[動的]パスを使用するのが選べます。静的パスを使用すれば、事前に本ダイアログボックスに入力する必要があります。動的パスを使用すれば、ファイルのパスは指定のアドレスを参照することになります。

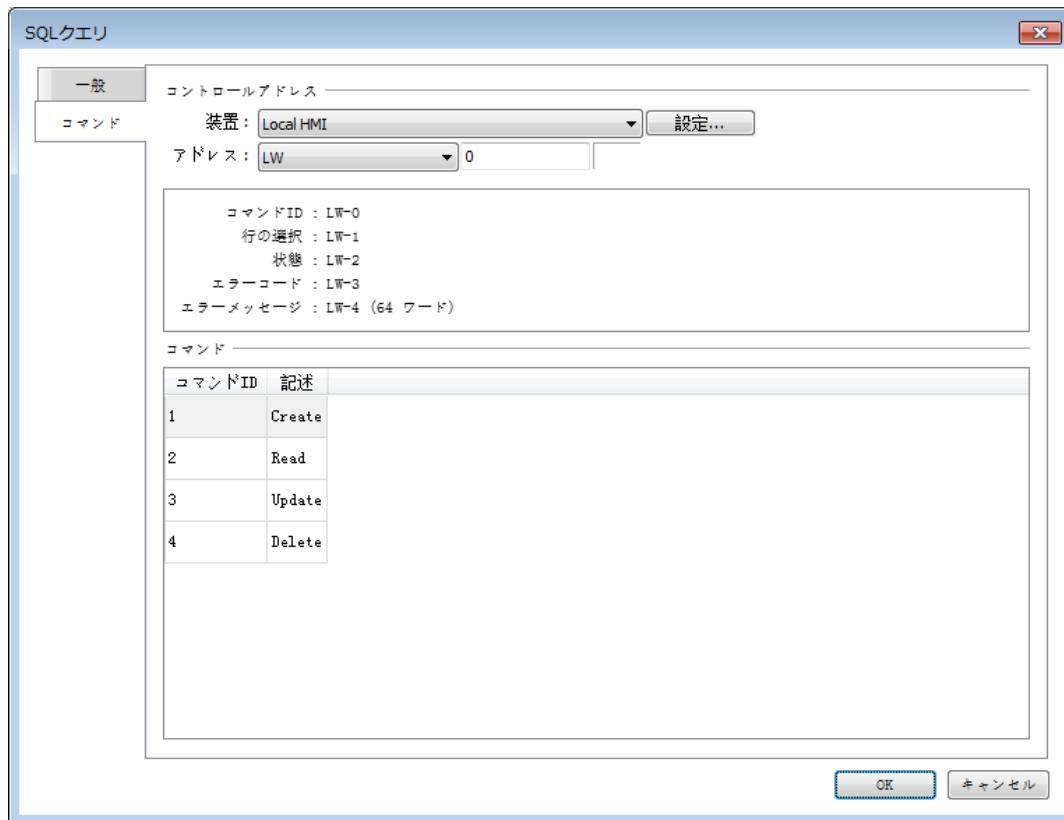
テーブル名	本クエリテーブルの名前を入力します。
スキーマ	データベースからデータを読み取った後、データは順次に[スキーマ]で設定されたアドレスに記入されます。データを読み取った後、[アドレスフォーマット]を手動で調整する必要があります。

Note

- SQL クエリオブジェクトの主キーのデータ型は数値である必要があり、文字をサポートしません。
- MySQL の中で、主キーには Auto Increment を有効にする必要があります。



コマンド設定



設定	記述
コントロール	5 個の連続したレジスタは MySQL のコマンドの実行、及び実行結果
アドレス	の表示に用いられます。[データベースからインポート]機能でテーブルをインポートする場合、コマンドウインドウには自動的に 4 個のコマンドが生成されます(Create, Read, Update, Delete)。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

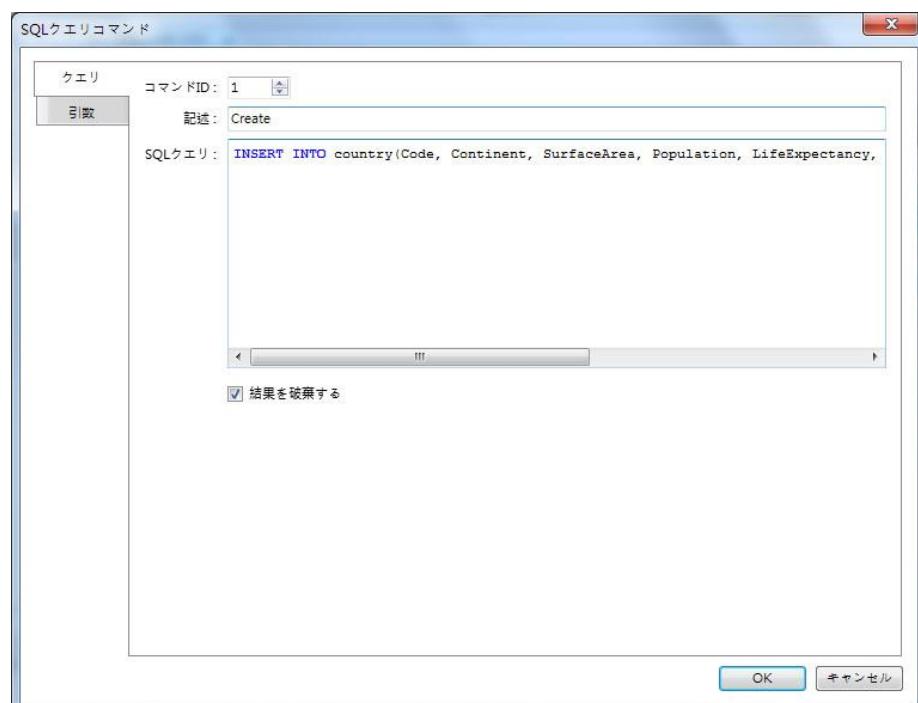
13.44.2.3. 高度モード

高度モードを選択すると、[コマンド]欄では[引数数]、[出力数]、[アクション]との三つの欄が増えます。



[新規作成]または[設定]をクリックすれば、[SQL クエリコマンド]設定ウインドウがオープンされます。

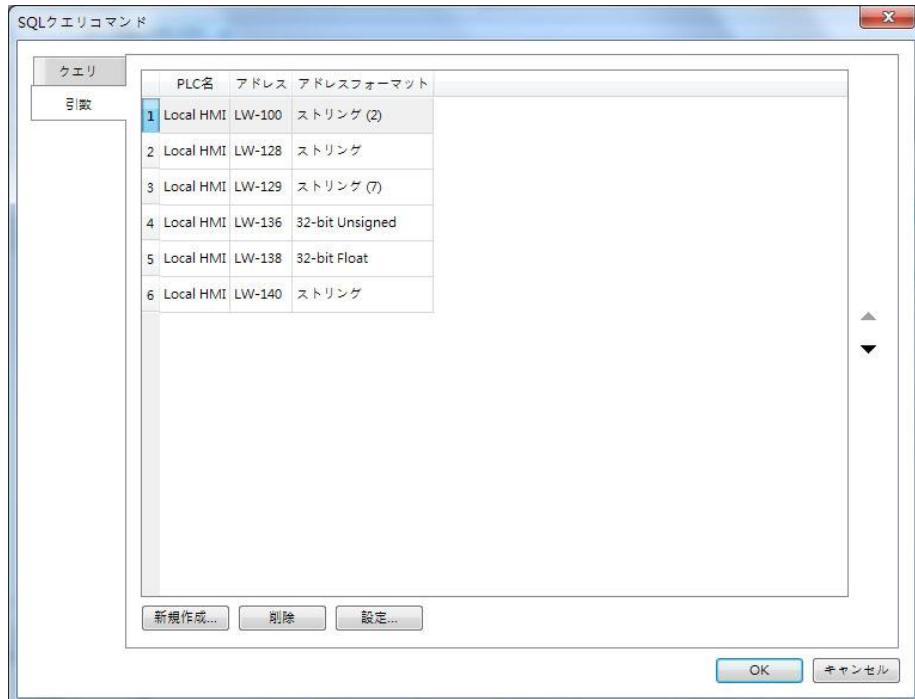
クエリタブ



設定	記述
コマンド ID	本 SQL シンタックスをコントロールするコマンドです。

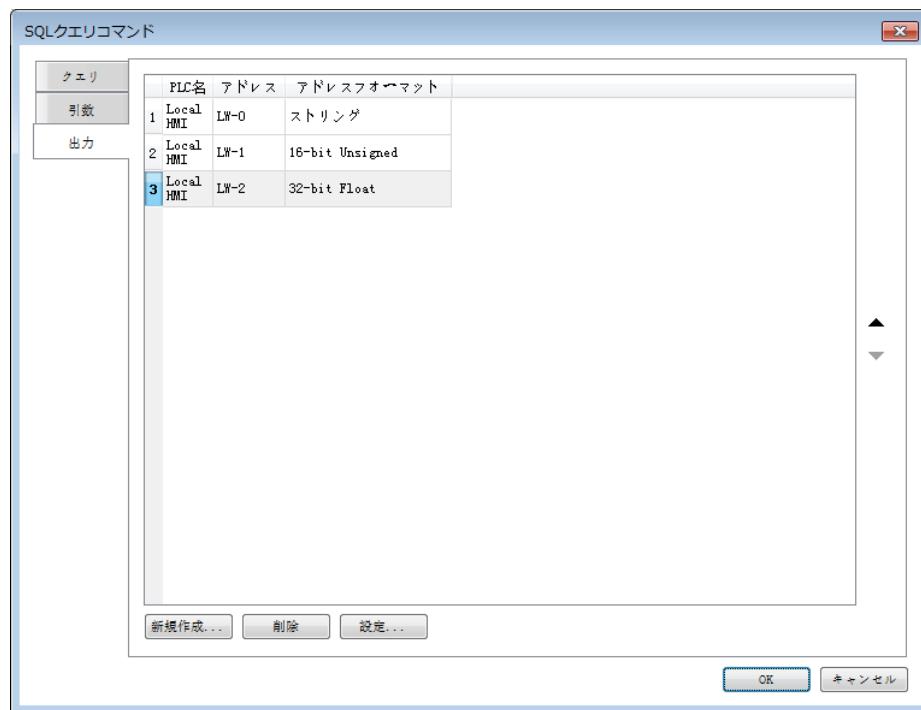
記述	本コマンドの記述を入力します。
SQL クエリ	実行したい SQL シンタックスです。引数を検索する方法は %{引数番号}。
結果を破棄する	チェックマークを入れると、実行完了の SQL コマンドは[SQL クエリ結果ビューア]オブジェクトに表示されません。

引数タブ



[クエリ]タブに引数が使用される場合、システムは %{引数番号} の設置に基づいて本設定タブで指定されたアドレスを参照します。

出力タブ



データベースを読み取る時、読み取った結果を順次に[出力]設定ページ内のアドレスに保存します。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.44.2.4. 状態

状態数値	意味
0	ノーマル
1	クエリ結果が 1000 レコードを超えた。SQL クエリステートメント LIMIT を使用すると、1 ページでの行数を制限できます。

13.44.2.5. エラーコード

エラーコード	意味
0	エラー無し
1	未知のエラー
2	不合法的なコマンド
3	データベースサーバーへの接続がまだできていない
4	引数が読み取れない
5	入力と書き込みができない
6	引数の数が正しくない

7	MySQL が正しくない。エラーメッセージの内容をご参照。
8	サポートされていないデータ型
9	欄の数が制限を超えた
10	選択した行が許容範囲を超えた
11	ローカルデータベースのパスを読み取れない
12	ローカルデータベースの名前が存在しない
13	内部エラー

13.44.2.6. データ型の変換

MySQL データベースからデータを読み取った後、下記テーブルに記載された内容を基づいてデータ型を変換します。変換されない場合、エラーコード 5 がポップアップされます。

例えば、MySQL の INT を 16-bit Unsigned に変換する時、数値の内容が 16-bit Unsigned で表示される範囲を超えた場合、エラーコード 5 がポップアップされます。

MySQL でのデータ型	EasyBuilder Pro でのデータ型
TINYINT	16/32-bit BCD
SMALLINT	16/32-bit HEX
MEDIUMINT	16/32-bit Binary
INT	16/32-bit Signed
BIGINT	16/32-bit Unsigned
BIT	
FLOAT	32-bit Float
DOUBLE	
DECIMAL	
DATETIME	String
CHAR, BINARY	
VARCHAR, VARBINARY	
TINYBLOB, TINYTEXT	
BLOB, TEXT	
MEDIUMBLOB, MEDIUMTEXT	
LONGBLOB, LONGTEXT	

13.44.3. SQL クエリ結果ビューア

13.44.3.1. 概要

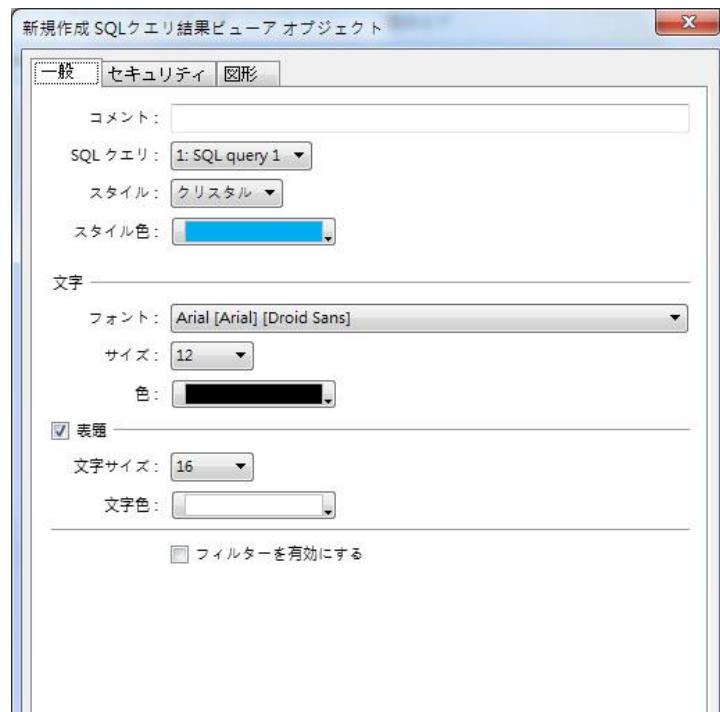
[SQL クエリ結果ビューア]オブジェクトは主に[SQL クエリ]機能と合わせて使用します。[SQL クエリ結果ビューア]機能の SQL 読み取り設定を設定した場合、[SQL クエリ結果ビューア]オブジェクトで SQL データを表示させることができます。

13.44.3.2. 設定



ツールバー上の[オブジェクト]»[データ/履歴]»[SQL クエリ結果ビューア]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[SQL クエリ結果ビューア]オブジェクトが新規作成されます。

一般的な属性



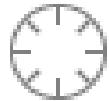
設定	記述
コメント	ユーザーは本オブジェクトに関連情報を記入することができます。
SQL クエリ	表示させたい[SQL クエリ]の番号を選択します。
スタイル / スタイル色	オブジェクトのスタイルと色を選択します。
文字	[SQL 結果ビューア]オブジェクトで表示する文字のサイズ、フォント、色を設定します。
表題	[SQL 結果ビューア]オブジェクトの表題を設定します。
テーブル	スタイルを[デフォルト]に設定した場合のみ、本欄が現れます。ここでテーブルの表示関連の設定をします。
フィルターを 有効にする	本機能を有効にすると、[SQL 結果ビューア]でキーワードを入力して指定の文字を検索することができます。

13.45. 動的スケール

13.45.1. 概要

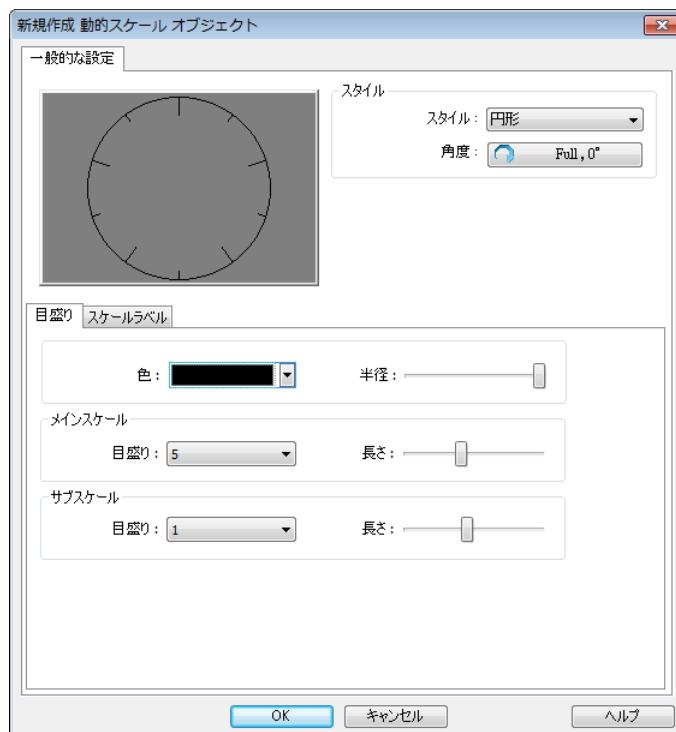
[動的スケール]オブジェクトは異なるスタイルの目盛りとスケールラベルを表現し、それにその内容を調整することができます。[トレンド図]、[棒グラフ]など、他のオブジェクトに目盛りとスケールラベルを提供します。

13.45.2. 設定



ツールバー上の[オブジェクト]»[チャート]»[動的スケール]ボタンを押せば、[動的スケール]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了してOKボタンを押すと、[動的スケール]オブジェクトが新規作成されます。

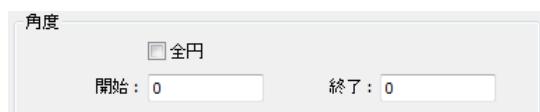
一般的な属性の設定



設定	記述
スタイル	[円形]、[水平]、[垂直]との3スタイルが選択できます。その中、[円形]では[移動方向]及び[角度]を設定することができます。



[移動方向]では[時計回り]と[反時計回り]が選択できます。
 [角度]では[全円]が選択でき、それに[開始角度]を設定することができます。
 [全円]を選択しない場合、[開始角度]及び[終了角度]を設定する必要があります。

**目盛り**

[動的スケール]オブジェクトの[色]、[メインスケール]及び[サブスケール]のスケール数を設定することができます。
 [動的スケール]のスタイルは[円形]だったら、[円形]の半径と[メイン、サブスケール]の長さを設定することができます。

**スケールラベル
ル**

[メインスケール]の上に[スケールラベル]を設置することができます。[円形]スケールラベルと[垂直]、[水平]スケールラベルがあり、詳細は下記の説明をご覧ください。

円形

動的スケールオブジェクトの中心点を原点とし、スケールラベルが表示する半径範囲を設定することができます。

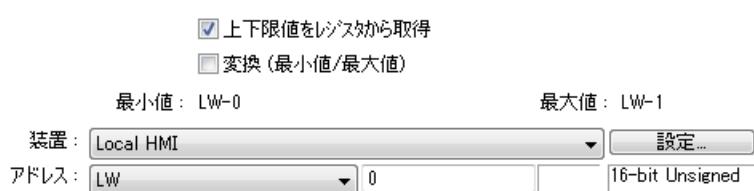
垂直、水平



スケールラベルが表示するフォント、文字色、フォントサイズ、小数点以下の桁数を設定することができます。

スケールラベルが表示する位置を設定することができます。

スケールラベルには[最大値]、[最小値]を設定することができます。
[上下限をレジスタから取得]を選択すると、レジスタを使用して上下限を設定することができます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.46. 動的作画

13.46.1. 概要

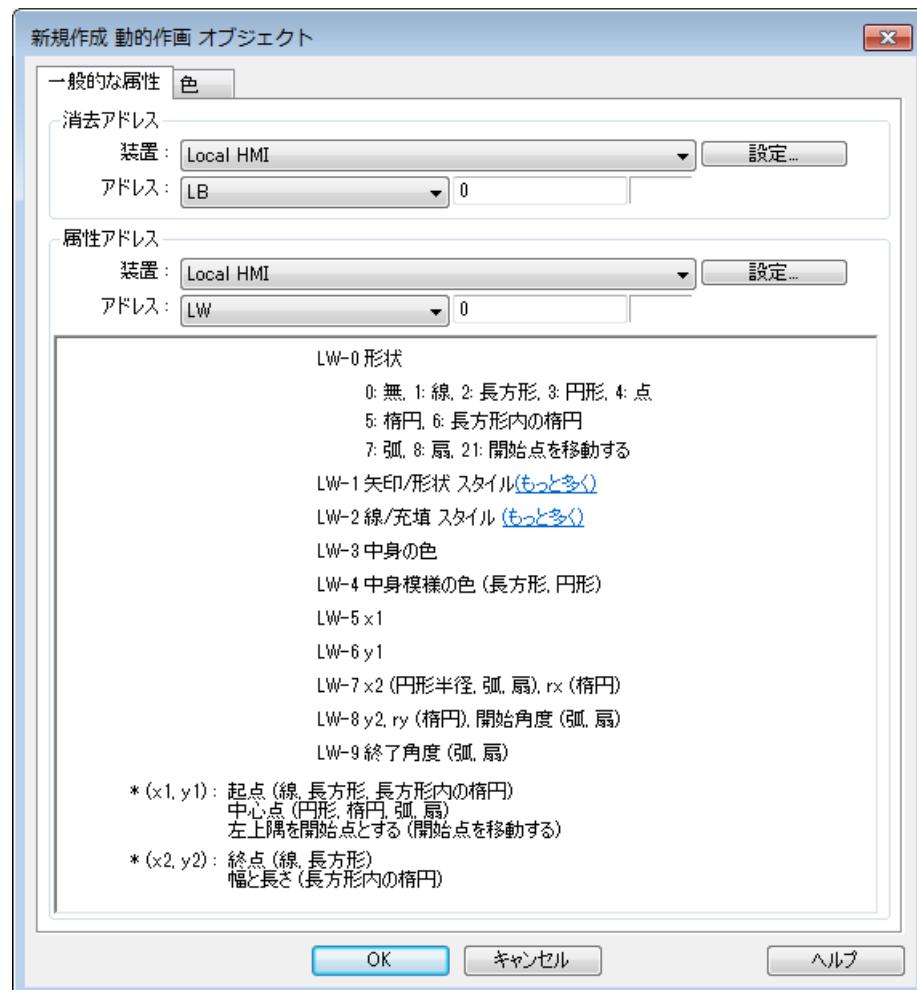
[動的作画] オブジェクトは、HMI が運転している時、特定のエリアで線分、長方形、円形、点などの図形を表示することができます。アドレス設定により、スタイル及び色のカスタマイズが可能です。

13.46.2. 設定



ツールバー上の[オブジェクト]»[チャート]»[動的作画]ボタンを押せば、[動的作画]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[動的作画]オブジェクトが新規作成されます。

一般的な属性の設定



設定	記述
消去アドレス	消去アドレスを設定し、動的作画を消去します。
属性アドレス	関連の属性アドレスを設定し、動的作画のコントロール、及び実行結果の表示をします。

各属性アドレスが対応するスタイルは以下のとおりです。

属性アドレス	属性アドレス+0	属性アドレス+1		属性アドレス+2	属性アドレス+3	属性アドレス+4
デフォルト	0	1 の位の数		10 の位の数		カスタマイズ
線	1	0:無	0:小	0:実線	線の色	
		1:片側が中空の矢印	1:大	1:破線		
		2:両側が中空の矢印		2:点線		
		3:片側が中実の矢印		3:一点鎖線		
		4:両側が中実の矢印		4:二点鎖線		
				5 またはそれ以上:実線で幅が 2 またはそれ以上		
長方形	2	0:中空		線のスタイルにより描かれる	長方形の色	中身に充填する色
		1:中実		充填のスタイルにより描かれる		
円形	3	0:中空		線のスタイルにより描かれる	円形の色	中身に充填する色
		1:中実		充填のスタイルにより描かれる		
点	4				点の色	
橢円	5	0:中空		線のスタイルにより描かれる	橢円の色	中身に充填する色
		1:中実		充填のスタイルにより描かれる		
長方形内の橢円	6	0:中空		線のスタイルにより描かれる	橢円の色	中身に充填する色
		1:中実		充填のスタイルにより描かれる		
弧	7			線のスタイルにより描かれる	弧の色	
扇	8	0:中空		線のスタイルにより描かれる	扇の色	中身に充填する色
		1:中実		充填のスタイルにより描かれる		
開始点を移動する	21					

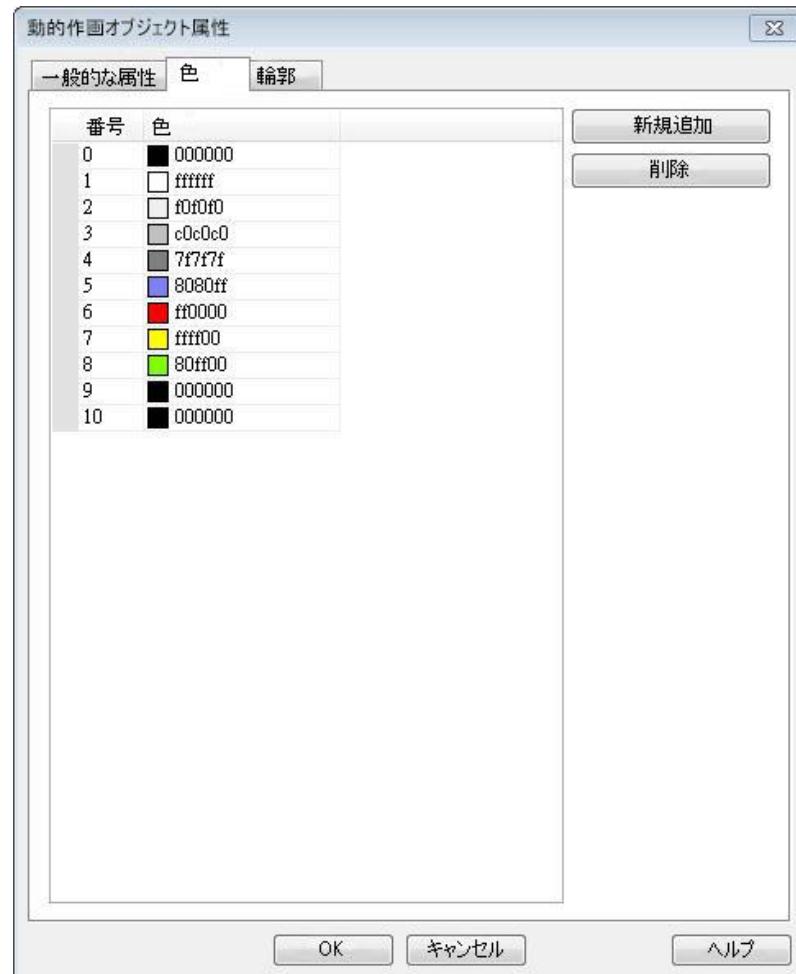
属性アドレス	属性アドレス+0	属性アドレス+5	属性アドレス+6	属性アドレス+7	属性アドレス+8	属性アドレス+9
デフォルト	0					
線	1	開始点 X	開始点 Y	終点 X	終点 Y	
長方形	2	左上隅 X	左上隅 Y	右下隅 X	右下隅 Y	

円形	3	円形の中心点 X	円形の中心点 Y	円形の半径		
点	4	点 X	点 Y			
楕円	5	楕円の中心点 X	楕円の中心点 Y	楕円の X 軸半径	楕円の Y 軸半径	
長方形内の楕円	6	左上隅 X	左上隅 Y	幅	高さ	
弧	7	弧の中心点 X	弧の中心点 Y	弧の半径	開始角度	終了角度
扇	8	扇の中心点 X	扇の中心点 Y	扇の半径	開始角度	終了角度
開始点を移動する	21	新規開始点 X	新規開始点 Y			

[属性アドレス+2]が異なるスタイルでの表示する数値は、下記に示す通りです。

線のスタイル	充填のスタイル
0	0
1	1
2	2
3	3
4	4
5	5
6	6
:	7
19	8
	9
	10
	11
	12
	13
	14
	15
	16
	17
	18
	19
	20
	21
	22
	23
	24
	25
	26

色の設定



設定

記述

新規追加

動的作画に使用する色を追加します。

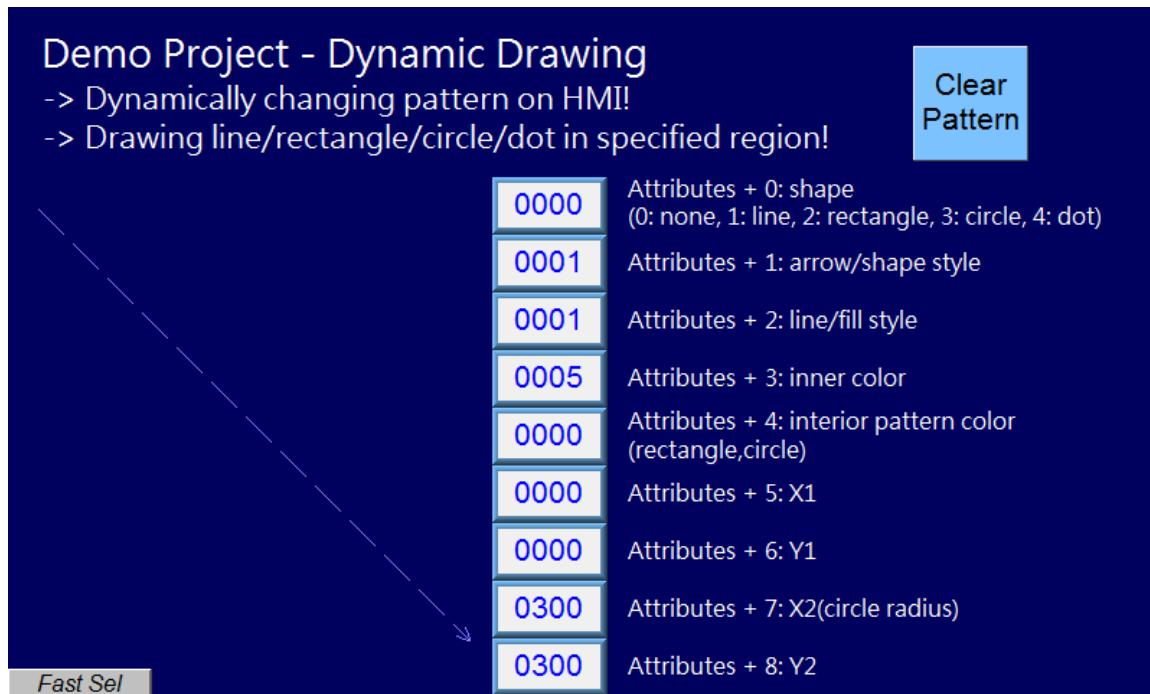
削除

選択する色を削除します。

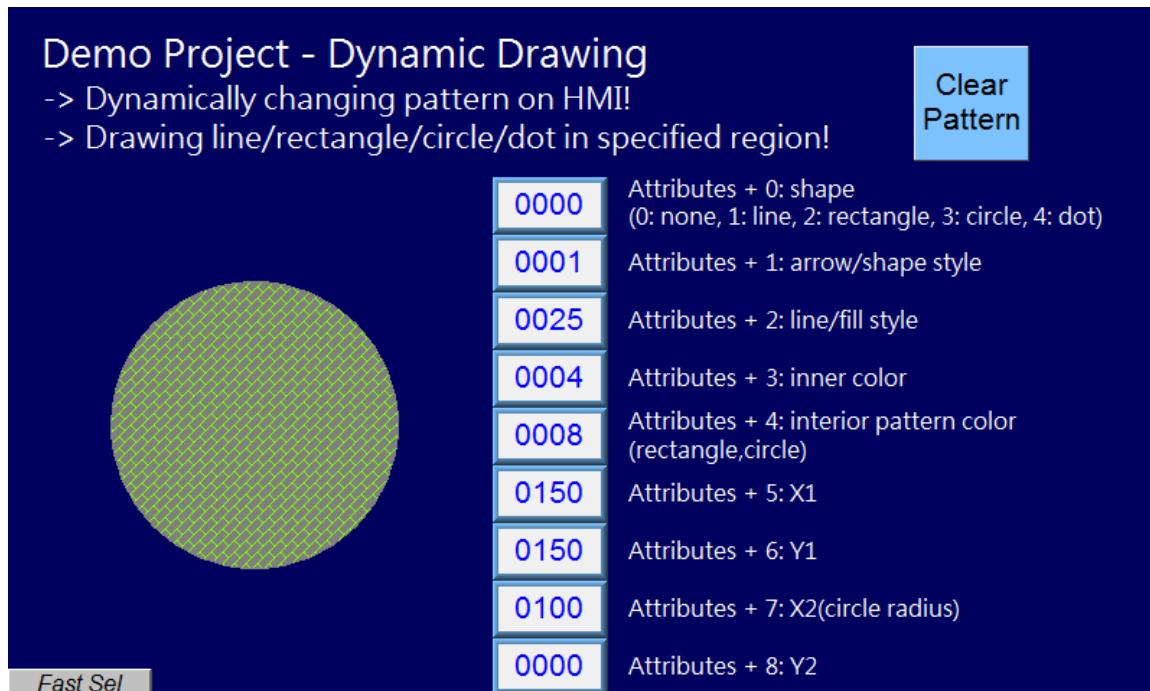
例 1

下記の例でどのように[動的作画]を利用し、線分及び円形を描く方法を見せます。

1. 動的作画オブジェクトを新規作成し、消去アドレスを LB-0 に設定し、属性アドレスを LW-0 に設定します。
2. 1 個のトグルスイッチオブジェクトを新規作成し、アドレスを LB-0 に設定し、[トグル]をスイッチスタイルに選択することで、図形の消去に用います。
3. 10 個の数値オブジェクトを作成し、アドレスを LW-0~LW-9 に設定し、作画の属性スタイルの設定に用います。
4. シミュレーションを実行、または HMI にダウンロードして操作を検証します。属性アドレスの数値設定が下図に示す通りになる場合、LW-0 アドレスに 1 を入力すれば、線分が画かれます。



- Clear Pattern ボタンを押し、属性アドレス数値設定を下図に示す通りに変更したら、LW-0 アドレスに 3 を入力すれば、円形が画かれます。



Note

- まずは[属性アドレス+1] ~ [属性アドレス+8]を設定してから、[属性]アドレスを実行します。操作完了後、システムは自動的に[属性アドレス]をリセットします。
- 消去をしなかったら、図形は重ねていきます。1個の動的作画オブジェクトの中には、最大 1000 個の図形を含むことができます。

- 線のスタイルの設定数値は、最大 19 で、即ち実線で幅が 16 のことです。数値が 19 を超えると、19 で表示します。
- 色の番号は色タブをご参照ください。
- 弧と扇の開始角度と終了角度は 0~360 に設定します。
- 本来の開始点は左上隅(0,0)ですが、開始点を移動するコマンドを出せば、開始点は(X1,Y1)に変更され、それに開始点の座標が継続に累加され、図形を消去した場合のみ、開始点が(0,0)復元されます。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.47.PDF リーダー

13.47.1. 概要

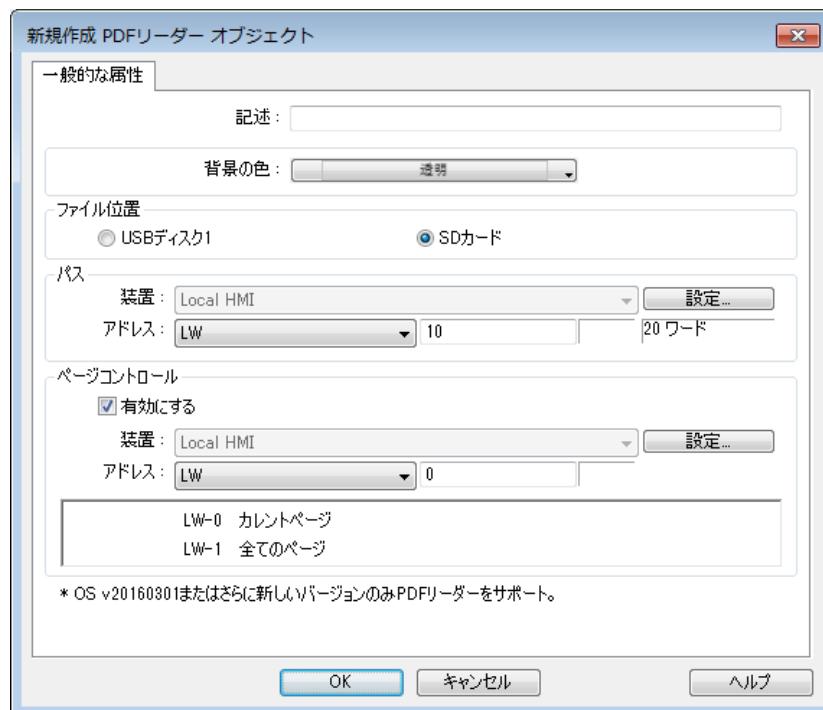
PDF リーダーオブジェクトを使用することで、HMI で PDF ファイルを閲覧することを可能にします。

13.47.2. 設定



直接に[PDF リーダー]アイコンをクリック、或いはツールバーの[オブジェクト]»[メディア]»[PDF リーダー]をクリックすることで本オブジェクトを新規作成します。

一般的な属性



設定	記述
ファイル位置	PDF ファイルを保存する位置を選択します。
パス	PDF ファイルが外部装置でのパスです。
ページコントロール	本アドレスでページ番号を入力したら、ページを切り替えることができます。

 Note

- PDF リーダーオブジェクトはシミュレーションモードまたは cMT Viewer でオープンすることができます。
- パスワード、または権限が設定された PDF ファイルは、PDF リーダーでオープンすることができます。
- 同じ時間で複数の PDF リーダーオブジェクトをオープンすると、CPU に負荷をかける可能性があります。
- マルチページ閲覧モードの下で、ページコントロールアドレスでページを切り替えたら、マルチページ閲覧モードから離れ、単一ページで選択したページを表示することになります。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

13.48. テーブル

13.48.1. 概要

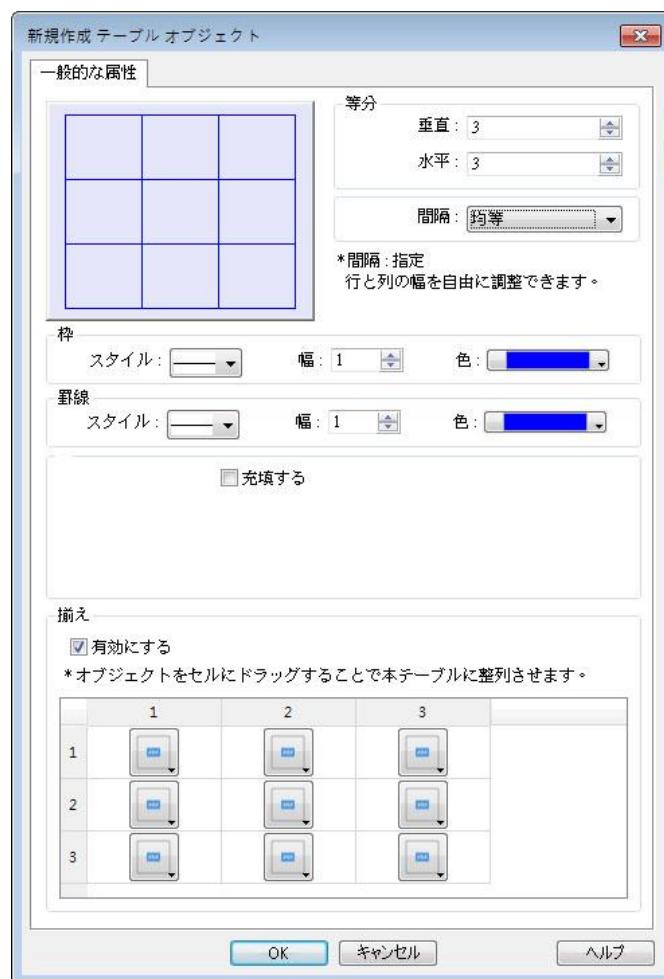
[テーブル]オブジェクトは編集ウインドウでテーブルを描くことができます。それにテーブルの枠、罫線及び充填のスタイルを設定できます。

13.48.2. 設定



直接に[テーブル]アイコンをクリック、或いはツールバーの[オブジェクト]»[テーブル]をクリックすることで本オブジェクトを新規作成します。

一般的な属性



設定	記述
プレビューウィンドウ	テーブルの設定を表示します。
垂直等分	垂直間隔の数量を設定します。範囲は 1 ~ 255 です。
水平等分	水平間隔の数量を設定します。範囲は 1 ~ 255 です。
間隔	[均等]或いは[指定]が選択できます。[指定]を選択したら、編集ウインドウで手動で垂直線と水平線の間隔を調整できます。
枠	テーブルの枠の幅及び色を設定します。線のスタイルを[実線]を選択した場合のみ、幅を設定することができます。その範囲は 0 ~ 8 です。線の幅を 0 に設定した場合、線が表示されません。
罫線	テーブルの罫線のスタイル、幅及び色を設定します。線のスタイルを[実線]を選択した場合のみ、幅を設定することができます。その範囲は 0 ~ 8 です。線の幅を 0 に設定した場合、線が表示されません。
充填	テーブルを充填する模様及び色を設定します。
揃え	他のオブジェクトをテーブルにドラッグすれば、システムは設定に従って自動的にオブジェクトの位置を調整します。テーブル内の各セルで独自の整列方向を設定でき、水平方向、垂直方向に設定できます。デフォルトは中央方向です。

13.49.VNC Viewer

13.49.1. 概要

[VNC Viewer]オブジェクトは、HMIの上でVNC機能を実行し、VNC serverをインストールしてあるPC、或いはVNC server機能をサポートしているHMIに接続できます。ユーザーはHMIで被接続側の画面を監視・コントロールできます。

13.49.2. 設定



直接に[VNC Viewer]アイコンをクリック、或いはツールバーの[オブジェクト]»[VNC Viewer]をクリックすることで本オブジェクトを新規作成します。

一般的な属性



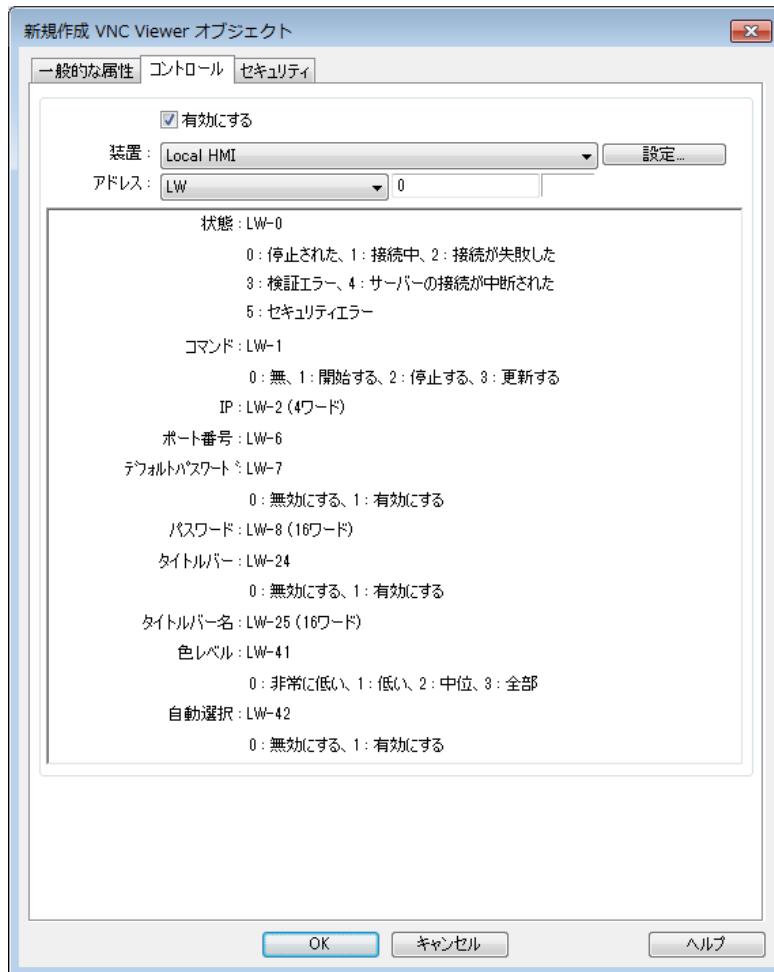
設定	記述
IP	接続先の IP アドレスです。
ポート番号	接続先のポート番号です。
デフォルトパスワード	接続先の VNC デフォルトパスワードを入力しログインすると、リモート装置に接続した時、VNC Viewer は直接に接続先に接続し、パスワードを入力する必要がありません。
色のレベル	[全部(全ての使用可能な色)]、[中位(256 色)]、[低い(64 色)]、[非常に低い(8 色)]を選択できます。 自動選択 VNC Viewer で自動に検査し、VNC サーバーがサポートする色のレベルを選択することができます。 実行時の修正を有効にする ([コントロール]機能で) コントロールアドレスを使用して色のレベルを選択、及び自動選択を有効にすることができます。本項は、[コントロール]タブで設定します。
タイトルバー	有効にする 選択した後、欄の中に VNC Viewer のタイトルバーに表示させたい文字を入力できます。ASCII 文字のみサポートし、フォントの設定ができません。[ラベルライブラリ]を合わせて使用することができます。同時に[有効にする]を選択してから、VNC Viewer ウィンドウを画面の上に任意に移動、拡大/縮小できます。



Note

- eMT、iE、XE、mTV: OS 20160418 以降、または最新バージョンの OS を使用する必要があります。
- cMT シリーズ: ローカル HMI 画面での実行のみサポートされます。OS 20180928 以降のバージョンを使用する必要があります。
- オンラインシミュレーション機能をサポートしません。
- [デフォルトパスワード]を選択した時、HMI の Virtual Keyboard(仮想キーボード)を手動で呼び出す必要があります。[デフォルトパスワード]を選択していない場合のみ、Virtual Keyboard が自動的にポップアップされます。

コントロール設定



設定	記述
コントロール	<p>関連のコントロールアドレスを指定し、[VNC Viewer] の各パラメータ及び表示状態を設定します。</p> <p>コントロールアドレス : 状態</p> <p>0 : 停止された 1 : 接続中 2 : 接続が失敗した 3 : 認証エラー 4 : サーバーの接続が中断された 5 : セキュリティエラー</p> <p>コントロールアドレス+1 : コマンド</p> <p>0 : 無 1 : 開始 2 : 停止</p> <p>コントロールアドレス+2~コントロールアドレス+5 : IP</p>

コントロールアドレス+6 : ポート番号
 コントロールアドレス+7 : デフォルトパスワード
 0 : 無効にする
 1 : 有効にする
 コントロールアドレス+8 : パスワード (16 ワード)
 コントロールアドレス+24 : タイトルバー
 0 : 無効にする
 1 : 有効にする
 コントロールアドレス+25 : タイトルバー名 (16 ワード)
 コントロールアドレス+41 : 色レベル
 0 : 非常に低い
 1 : 低い
 2 : 中位
 3 : 全部
 コントロールアドレス+42 : 自動選択
 0 : 無効にする
 1 : 有効にする

Note

- コントロールアドレス+6 のポート番号は 0-99 の場合、実際のポート番号は入力値に 5900 を足したものです。例に挙げると、コントロールアドレス+6 に 1 を入力したら、実際のポート番号は 5901 になります。しかし、コントロールアドレス+6 に 100 を入力すると、実際のポート番号は 100 になります。
 VNC Viewer をオープンした時、HMI が内蔵した Virtual Keyboard で入力したかったら、まずは Virtual Keyboard 上の [Focus] ボタンを押してから、VNC Viewer の画面をクリックしたこそ、入力先をそちらに切り替えることができます。



 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

 このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

13.50.連絡先エディター

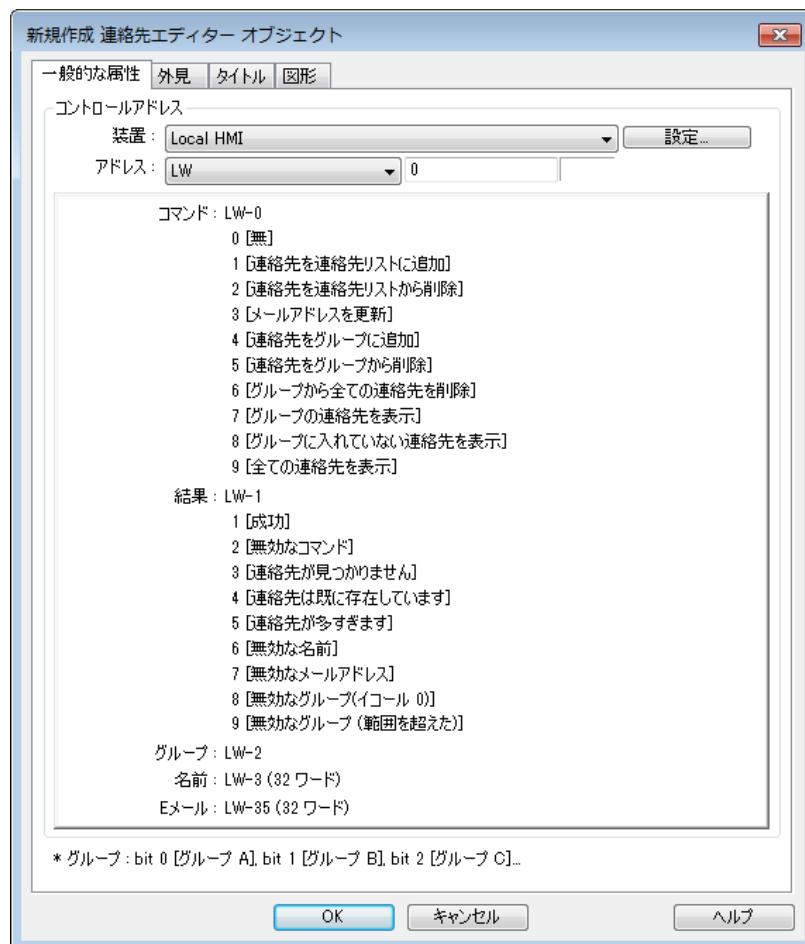
[連絡先エディター]オブジェクトは、HMIでEメール連絡先を動的に追加 / 修正 / 削除することをサポートします。

13.50.1. 設定



直接に[連絡先エディター]のアイコンをクリック、或いはツールバーの[オブジェクト]»[連絡先エディター]をクリックすることで本オブジェクトを新規作成します。

一般的な属性



設定	記述
コントロールアドレス	指定したレジスタ及びその連続したアドレスは、[連絡先エディター]のパラメータを設定することに用いられます。

コマンドを出すアドレス (LW-n) :

数値	内容
0	無
1	連絡先を連絡先リストに追加する
	連絡先リストから連絡先を削除する
3	メールアドレスを更新する
4	連絡先をグループに追加する
5	グループから連絡先を削除する
6	グループから全ての連絡先を削除する
7	グループ内の連絡先を表示する
8	グループにない連絡先を表示する
9	全ての連絡先を表示する

実行結果を表示するアドレス (LW-n+1) :

数値	内容
1	実に成功た
2	無効なコマンド
3	連絡先が見つからない
4	連絡先が既に存在する
5	連絡先の数が上限に達し、追加できない
	無効なコマン
7	無効なメールアドレス
8	無効なグループ
9	無効なグループ(範に超えた)

連絡先グループ (LW-n+2) : ビットでグループを示す

ビット	内
0	グループ A
1	グループ
2	グループ
3~1	上記のように類推する

ユーザー名 (LW-n+3) : 合計 32 ワード

メールアドレス (LW-n+35) : 合計 32 ワード



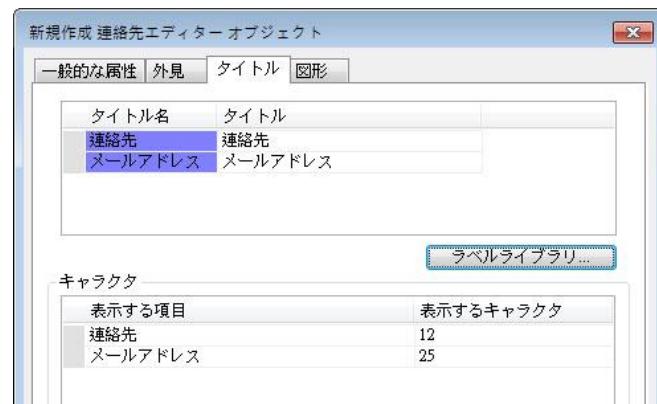
- cMT シリーズには、一般的な属性の設定タブがありません。
- 連絡先の名前は Unicode 文字をサポートしません。
- グループ数は[システムパラメータ設定] » [E メール] » [宛先]で設定されます。HMI で動的に変更することができません。

外見設定



設定	記述
フォント & 色	[連絡先エディター]オブジェクトが表示するフォント及び色を調整します。

タイトル設定



設定	記述
タイトル	タイトルで表示させる文字を設定します。
表示するキャラクタ	表示させるタイトルの文字長さを設定します。範囲 : 1~60 です。

13.51. イベントバーチャート

概要

イベントバーチャートは一種の簡単で便利な棒グラフで、常にプロジェクト管理に用いられます。

イベントバーチャートで HMI のイベント/アラームを表現すると、イベント/アラームの発生時刻、継続期間などが一目瞭然に表示されます。イベントバーチャートを使用するには、事前にイベントログオブジェクトを設定する必要があります。

本機能は cMT シリーズ HMI でのみサポートされています。(cMT Gateway 以外)。

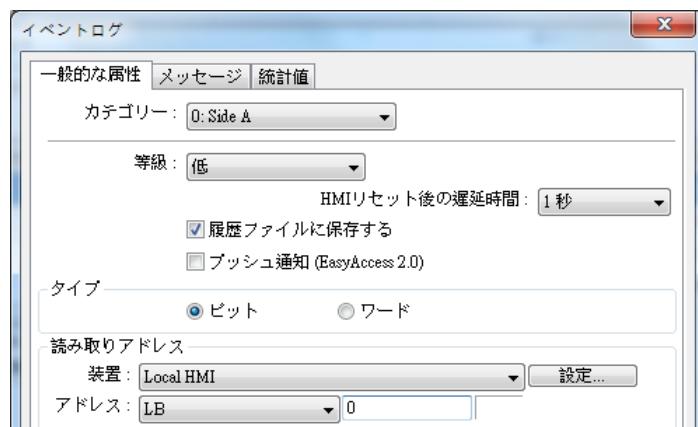
13.51.1. イベントログ

ツールバーの[データ/履歴] » [イベントログ]でイベントログオブジェクトを数個作成します。

イベントバーチャートの作成するのに、一部のパラメータは[イベントログ]で設定する必要があります。以下ではイベントバーチャートを作成するために使用されるパラメータを紹介します。

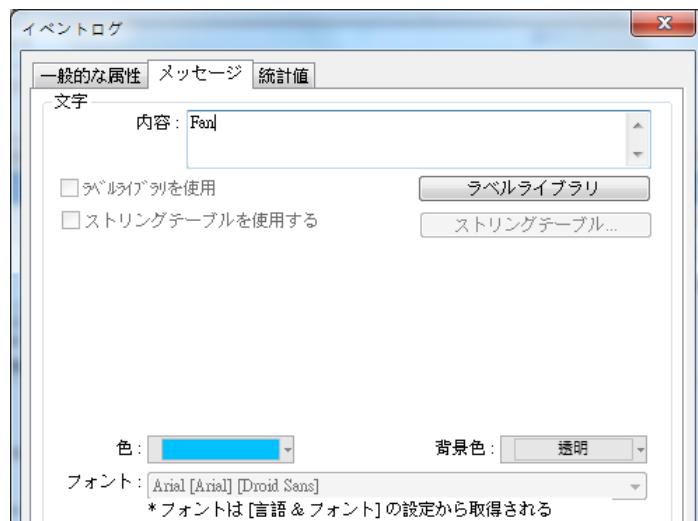
イベントログ																																																					
設定		記述																																																			
対応するカテゴリー名を編集		イベントバーチャートで表示する各カテゴリーの名前です。																																																			
<table border="1"> <thead> <tr> <th>番号</th><th>カテゴリー</th><th>イベント内容</th><th>モード</th><th>誘発条件</th><th>読み取りアドレス</th><th>通知アドレス</th><th>ブザー</th><th>Eメール</th></tr> </thead> <tbody> <tr> <td>1</td><td>0</td><td>Fan</td><td>BIT</td><td>ON</td><td>Local HMI: LB-0</td><td>無効にする</td><td>無効にする</td><td>無効にする</td></tr> <tr> <td>2</td><td>1</td><td>Fan</td><td>BIT</td><td>ON</td><td>Local HMI: LB-1</td><td>無効にする</td><td>無効にする</td><td>無効にする</td></tr> <tr> <td>3</td><td>2</td><td>Fan</td><td>BIT</td><td>ON</td><td>Local HMI: LB-2</td><td>無効にする</td><td>無効にする</td><td>無効にする</td></tr> <tr> <td>4</td><td>0</td><td>Pump</td><td>BIT</td><td>ON</td><td>Local HMI: LB-3</td><td>無効にする</td><td>無効にする</td><td>無効にする</td></tr> </tbody> </table>									番号	カテゴリー	イベント内容	モード	誘発条件	読み取りアドレス	通知アドレス	ブザー	Eメール	1	0	Fan	BIT	ON	Local HMI: LB-0	無効にする	無効にする	無効にする	2	1	Fan	BIT	ON	Local HMI: LB-1	無効にする	無効にする	無効にする	3	2	Fan	BIT	ON	Local HMI: LB-2	無効にする	無効にする	無効にする	4	0	Pump	BIT	ON	Local HMI: LB-3	無効にする	無効にする	無効にする
番号	カテゴリー	イベント内容	モード	誘発条件	読み取りアドレス	通知アドレス	ブザー	Eメール																																													
1	0	Fan	BIT	ON	Local HMI: LB-0	無効にする	無効にする	無効にする																																													
2	1	Fan	BIT	ON	Local HMI: LB-1	無効にする	無効にする	無効にする																																													
3	2	Fan	BIT	ON	Local HMI: LB-2	無効にする	無効にする	無効にする																																													
4	0	Pump	BIT	ON	Local HMI: LB-3	無効にする	無効にする	無効にする																																													

13.51.1.1. 一般的な属性



設定	記述
カテゴリー	設定していない場合、イベントバーチャートは全てのイベントを1個のカテゴリー内で作画するので、カテゴリーを正確に設定する必要があります。

13.51.1.2. メッセージ設定



設定	記述
文字内容	項目の名前です。
色	項目の色です。

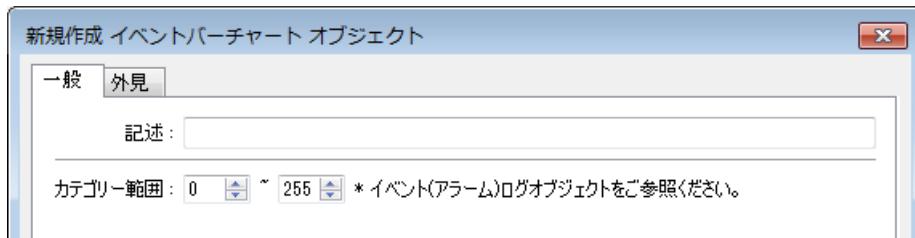
13.51.2. イベントバーチャート

13.51.2.1. 設定



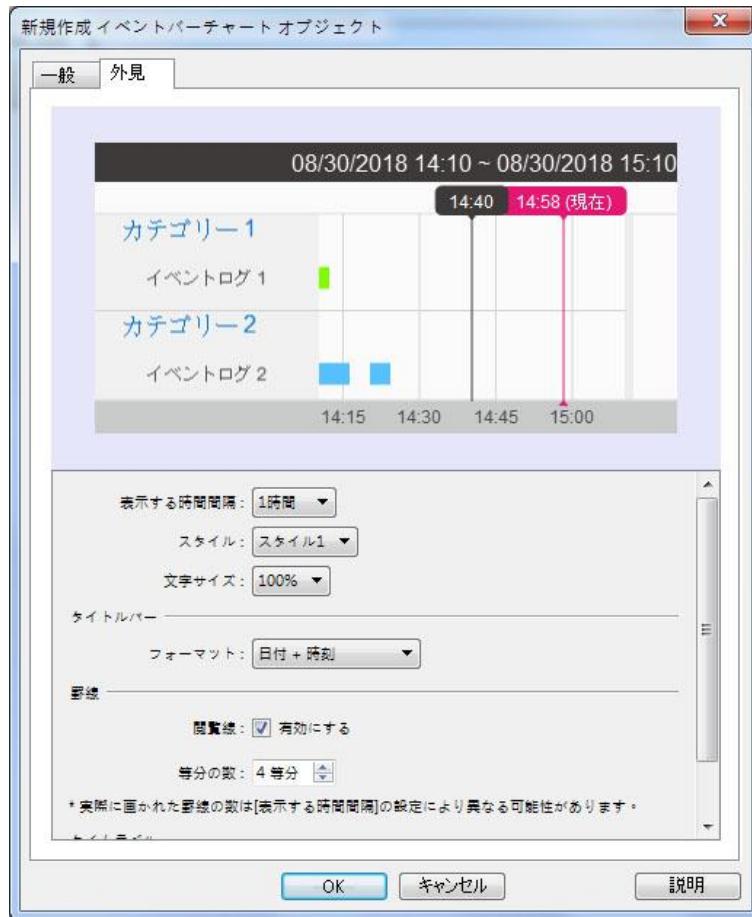
ツールバーの[データ/履歴] » [イベントバーチャート]をクリックしてイベントバーチャートオブジェクトを作成します。

一般的な属性の設定



設定	記述
カテゴリー範囲	イベントバーチャートに表示するカテゴリーを選択します。

外見設定



設定

記述

**表示する時間
間隔**



表示する時間範囲です。本項は HMI でオブジェクトの右上隅のボタンを押して調整することができます。



スタイル

イベントバーチャートのスタイルです。システムはデフォルトの 3 スタイルを提供し、カスタムスタイルも選択可能です。

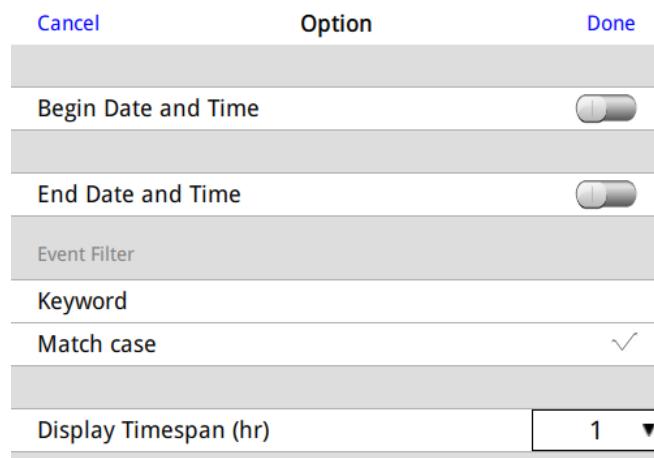
文字サイズ

イベントバーチャートでの文字のサイズです。

タイトルバーフ オーマット	日付+時刻 : 08/16/2018 13:55 ~ 08/16/2018 14:55 日付のみ表示する : 08/16/2018 時刻のみを表示する : 13:57 ~ 14:57
閲覧線	イベントバーチャートの任意の所にクリックすれば、閲覧線が現れます。閲覧線のトップ部ではこの位置の時間が表示されます。
等分の数	X 軸の罫線数です。X 軸は[等分の数]に設定された通りに等分されます。
タイムラベル	時間スケールの表示方式です。

HMI での設定

HMI で、或いは cMT Viewer ではオブジェクト右上隅の  ボタンでオブジェクトの表示を調整できます。



設定	記述
開始日付及び 時刻	表示するデータの開始日付と時刻を設定します。 デフォルトでは、本設定は無効になっています。表示するデータは既存データの開始時刻から表示されます。
終了日付及び 時刻	表示するデータの終了日付と時刻を設定します。 デフォルトでは、本設定は無効になっています。表示するデータはカレント時刻までです。
イベントフィ ルター	キーワード : キーワードで特定したイベントをろ過できます。 大文字小文字を区別 : 英語で検索する時に、大文字、小文字を異なるキャラクタに視すかを設定します。
タイムスパン を表示する (時 間)	オンラインでチャートで表示する時間範囲(1~96)を変更します。All とは既存のデータ範囲が表示されるデータ範囲です。

Note

- HMI が電源切れになった場合、各アラームの状態を取得できないため、電源が入れられた後、HMI は電源切れ期間でのバーチャートを作画しません。

13.52.動作トリガー

13.52.1. 概要

[動作トリガー]オブジェクトは指定したトリガー条件で実行したい動作を分類することができます。動作を設定する時、まずは動作をグループ分けし、同一のグループ内の動作は一緒に実行されます。前のグループ内の動作が全てトリガーされた場合のみ、次のグループの動作が実行されます。

[動作トリガー(ウインドウ)]を使用した場合、トリガー条件は[ウインドウオープン/クローズ]或いは[バックライトオン/オフ]です。

[動作トリガー(背景)]を使用した場合、トリガー条件は HMI が[アイドル時間]或いは[数値変化]を検出したことです。

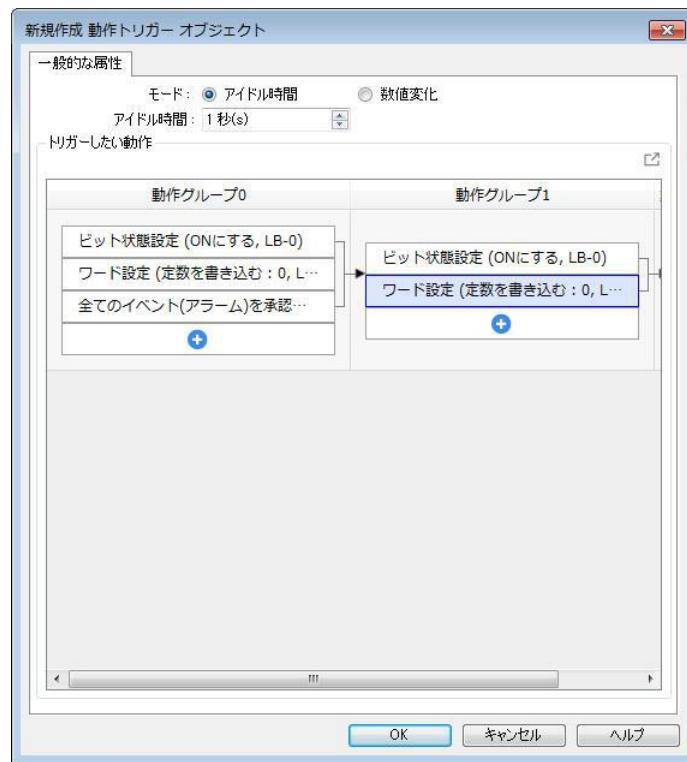
本オブジェクトは cMT シリーズ機種にのみサポートされています。

13.52.2. 設定



ツールバー上の[オブジェクト] » [動作関連] » [動作トリガー(ウインドウ)]或いは[動作トリガー(背景)]ボタンを押せば、[動作トリガー]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[動作トリガー]オブジェクトが新規作成されます。

一般的な属性



設定	記述
モード(ウインドウ)	<p>ウインドウオープン ウインドウがオープンされると、設置した動作をトリガーします。本モードを使用した場合、[セキュリティ]タブをサポートしません。</p> <p>ウインドウクローズ ウインドウが閉じられると、設置した動作をトリガーします。</p> <p>バックライトオン バックライトがつけられると、設置した動作をトリガーします。</p> <p>バックライトオフ バックライトが消されると、設置した動作をトリガーします。</p>
[アイドル時間]モード (背景)	HMI が操作されていない状況で[アイドル時間]が設定した時間を越えると、当該動作がトリガーされます。範囲：1 ~ 43200 秒です。
[数値変化]モード(背景)	<p>指定したアドレスの数値が特定条件を満たしたら、当該動作がトリガーされます。</p> <p>動作が完了した後、トリガー条件を検査する 全ての動作が実行完了した後、オブジェクトは再度にトリガーアクションが満たされたかを確認します。満たされたら、再度に動作を実行します。</p> <p>動的状態数値 [ワード]タイプを使用する場合、動的状態数値がサポートさ</p>

れます。トリガー条件に動的数値を使用し、それに[読み取り/状態表示に異なるアドレスを使用する]を有効にしていない場合、トリガー条件は[数値アドレス]の次の連続したレジスタから取得されます。

動作グループ

動作をグループ分けにします。同一のグループ内の動作は一緒に実行され、グループ内の動作が全てトリガーされた場合のみ、次のグループの動作が実行されます。

[遅延]動作

n ミリ秒に遅延してから、コマンドを実行します。

[ビット設定]動作

レジスタの状態を ON または OFF に設定します。

ON に設定する

指定したレジスタの状態が ON に設定されます。

OFF に設定する

指定したレジスタの状態が OFF に設定されます。

トグル

指定したレジスタの状態が反対に設定されます。

[ワード設定]動作

指定したレジスタの数値を変更します。

定数を書き込む

定数を指定したレジスタに書き込みます。

増加(JOG+)

数値を増加する機能です。指定したレジスタ内の数値に[増加値]で設定した数を足します。但し、増加した結果は[上限値]の設定を超られません。

減少(JOG-)

数値を減少する機能です。指定したレジスタ内の数値に[減少値]で設定した数値を引きます。但し、減少した結果は[下限値]の設定値より低くなることはありません。

動的制限

上限/下限は指定したレジスタより設定されます。

定数ストリングを書き込む

特定のストリングを指定したレジスタに書き込みます。

[ベースウインドウを切

り替える]動作

ベースウインドウを切り替える

指定のウインドウに切り替えます。

共有ウインドウを切り替える

共有ウインドウを切り替えます。

前のベースウインドウに戻る

前のベースウインドウに戻ります。例えば、"ウインドウ 10" から "ウインドウ 20" に切り替える場合、本機能を使用すれば、再度に"ウインドウ 10"に戻ることができます。本機能はベースウインドウでのみ使用できます。

アニメーション設定

cMT シリーズにはウインドウのアニメーションがサポートされます。[アニメーション設定]をクリックした後、選択でき

るアニメーション効果は：フェードイン、飛んでくる、浮かび上がる、ワイプ、スプリット、サイクル、クロック、ズーム、ターン、プッシュ等があり、アニメーションの表示期間と表示方向も設定できます。

[マクロをトリガーする]

動作

指定したマクロをトリガーします。グループ内にマクロがあり、それにマクロの実行時間が長い場合、マクロがトリガーされた途端、動作の実行に成功したと視され、マクロの実行が全部完了してから次のグループの動作をトリガーすることではありません。

[ウインドウをポップアップする]動作

指定したウインドウをポップアップします。

[キーボード入力]動作

キーボードの入力信号として用いられ、主に[数値]と[アスキイ]など、キーボードで数値や文字を入力することが必要な場合に使用されます。

Enter: キーボードの入力(Enter)動作と同じです。

Backspace: キーボードのバックスペース(Backspace)動作と同じです。

Clear: レジスタ内に入力したデータを消去します。

Esc: [ウインドウを閉じる]機能と同じで、ポップアップされたキーボードウインドウを閉じることに用いられます。

Delete: キーボードの削除(Delete)動作と同じで、カーソルの右側の数値またはキャラクタを一個削除します。

Left: キーボードの←動作と同じで、カーソルを左へキャラクター個を移動します。

Right: キーボードの→動作と同じで、カーソルを左へキャラクター個を移動します。

Inc: 数値を+1します。

Dec: 数値を-1します。

ASCII/UNICODE: キーボードで入力するキャラクタを設定します。

[スクリーンのハードコピーを取る]動作

現在のスクリーンのハードコピーを取ることができ、それにハードコピーを USB ディスクまたは SD カードに保存できます。

[全てのイベント(アラーム)を承認する]動作

実行すると、一回で全てのイベント(アラーム)を承認することができます。

[データをインポート]動作

高度セキュリティのユーザー アカウントまたは e-mail の連絡先をインポートすることに用いられます。USB ハードウェアキーでログインすることに設定できます。

データ位置: USB ディスク、SD カードから読み取るが選べます。

アカウントインポートモード: [上書き]を選択すれば、HMI 内では今回インポートしたアカウントしか保存されません。

[付加]を選択すると、HMI内のアカウントデータは保留され、今回新しくインポートしたアカウントデータが付加されます。

アカウントをインポートした後、ファイルを削除する：USBディスク内のユーザー アカウントをインポートした後、ソースファイルを削除します。そうしたら、データ漏れの恐れが避けられます。

[トリガー条件を待つ]

動作

ビット/ワードを使用して異なるトリガー条件を設定します。トリガー条件が満たされた場合のみ、次の[動作グループ]の動作が実行されます。トリガー条件が待ち時間(タイムアウト)を超えても満たされていない場合、次の[動作グループ]の動作を実行、または動作を実行しなくて、直ちに終了することを選択できます。

[データ転送(背景)]動作

ファイル転送

指定したアドレス内のデータを他のアドレスに転送します。

FTPでファイルを転送します。HMIはパッシブモードでFTPサーバーに接続します。

[一般的な属性]タブ

転送方向を設定します。[ダウンロード]はFTPサーバーからファイルをローカルHMIに転送し、[アップロード]はファイルをローカルHMIからFTPサーバーに転送します。

[ファイル設定]タブ

ファイル位置、FTPサーバーファイルパス、ローカルHMIファイルパスを設定できます。パスがフォルダである場合、そのフォルダ内にある全部のファイルを転送します。但し、サブフォルダ内のファイルを含みません。ファイル名が同じの場合、ファイルを上書きします。

[状態]タブ

ファイル転送の実行結果及びリターンコードを表示するためのアドレスを設定します。

FTPサーバーのリターンコードの一覧表は次のリンクを参考してください：

https://en.wikipedia.org/wiki/List_of_FTP_server_return_codes

Note

- プロジェクトには最大 1000 個の動作トリガーオブジェクトを作成できます。
- 一個の複合ボタンには最大 20 グループを作成でき、各グループに最大 20 個の動作を含むことが可能です。
- ファイル転送機能のエラーコードリスト：

エラーコード	説明
0	実行に成功した

1	ダウンロードする時、HMI のパスワードが存在しません
3	USB ディスクまたは SD カードが検出されていない
4	HMI または FTP のパスが入力されていない
5	アップロードファイルが存在しない
8	操作が FTP に拒否された
9	USB ディスクまたは SD カードの容量が一杯になった
10	未知のエラー

13.53. カレンダー

13.53.1. 概要

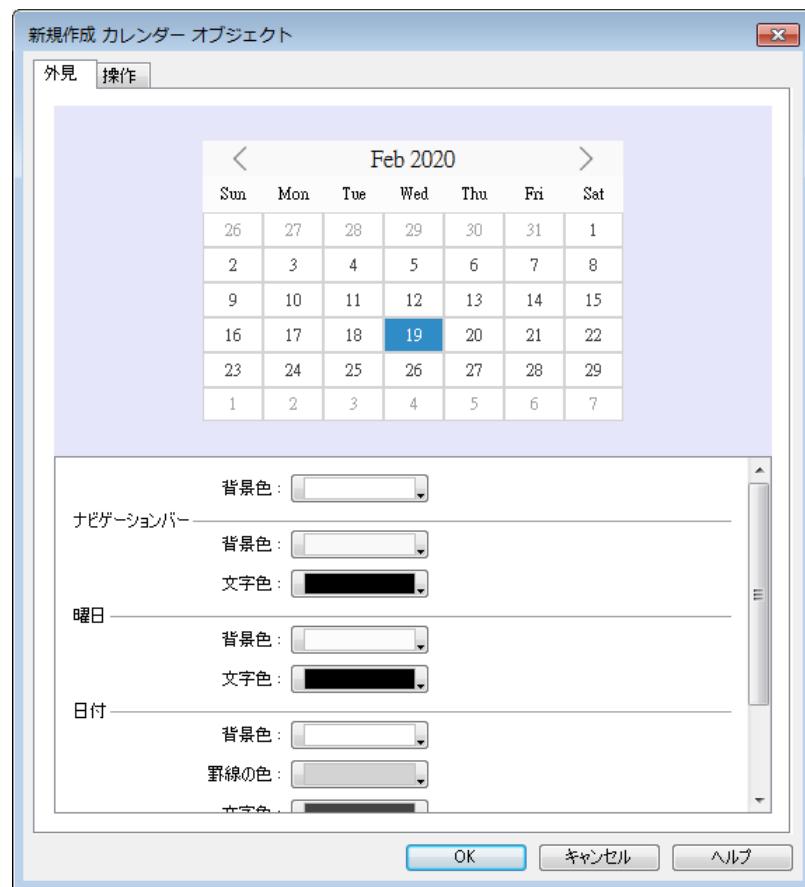
[カレンダー]オブジェクトを使用することにより、cMT シリーズの HMI でカレンダーを表示することができます。

13.53.2. 設定



[ツールバー上の[オブジェクト] » [時間関連] » [カレンダー]ボタンを押せば、[カレンダー]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を設定完了して OK ボタンを押すと、[カレンダー]オブジェクトが新規作成されます。

一般的な属性



設定	記述
外見	ナビゲーションバー：背景及び文字の色を設定できます。 曜日：背景及び文字の色を設定できます。 日付：背景、罫線及び文字の色を設定できます。 今日：背景及び文字の色を設定できます。
操作	日付情報を指定したアドレスに出力し、出力フォーマットは数値、或いはストリングに指定することができます。
	

▶このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

13.54. タッチジェスチャー

13.54.1. 概要

現在のスマートフォン、タブレットには、マルチタッチの方式、例えば同時に5本指でピンチすれば、現在のウインドウが閉じられるなどを操作できます。[タッチジェスチャー]オブジェクトを使用すれば、HMIの上でもスマートフォンのような操作感を体験できます。[タッチジェスチャー]オブジェクトは23種類のジェスチャーをサポートします。

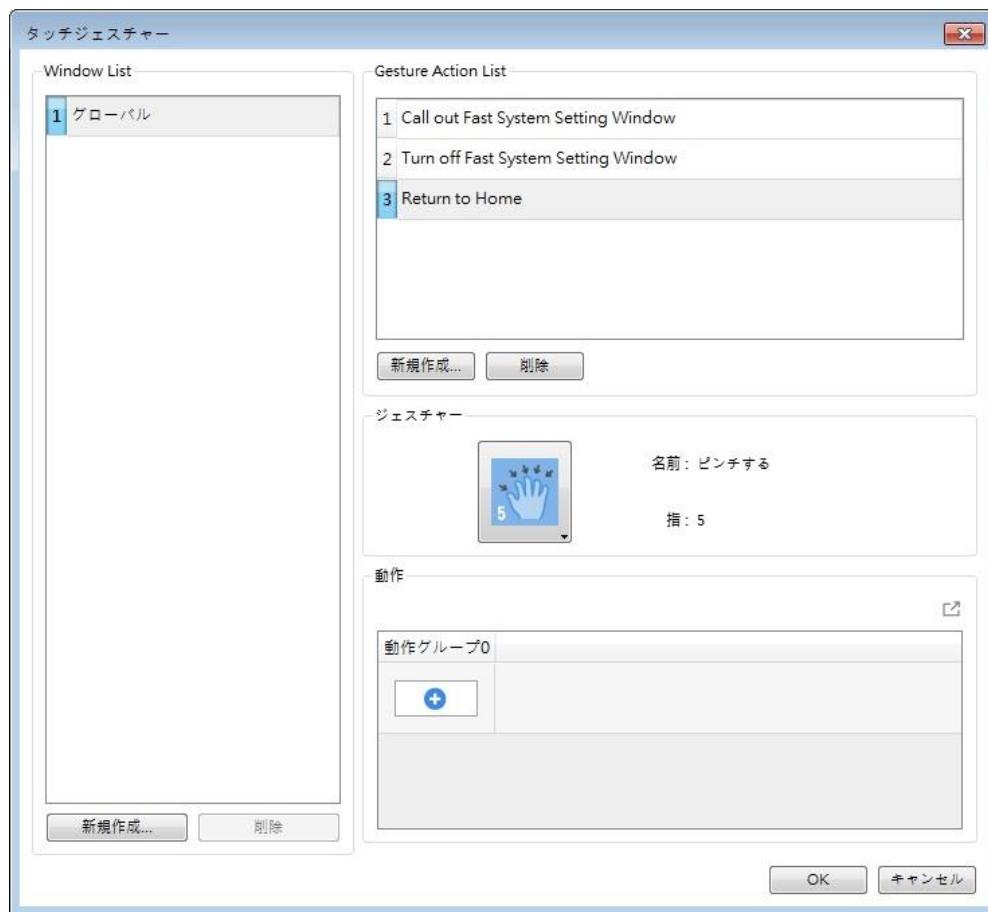
最大5本指で操作できます。

13.54.2. 設定



ツールバー上の[オブジェクト]»[動作関連]»[タッチジェスチャー]ボタンを押せば、[タッチジェスチャー]オブジェクトの属性ダイアログボックスがオープンされ、まず対象ウインドウを選択してから、[新規作成]をクリックして一個のタッチジェスチャーオブジェクトを新規作成します。

一般的な属性



設定	記述
ウインドウリスト	ジェスチャーが特定したウインドウでのみ動作するか、或いは全てのウインドウ(グローバル)で動作するかを選択します。グローバルを選択した場合、1本指のジェスチャーをサポートしません。 新規追加/削除 対象ウインドウを新規追加/削除します。
ジェスチャーリスト/ジェスチャー	ウインドウで使用するジェスチャーを表示します。 新規追加/削除 ジェスチャーを新規追加/削除します。
動作	当該ジェスチャーを使用する際に、トリガーできる動作です。動作の説明については、ユーザーマニュアル CH13.36 もしくは CH13.52 を参照してください。

 Note

- 各ウインドウ(グローバルウインドウ/個別ウインドウ)では最大 16 個のジェスチャーを使用することができます。
- グローバルウインドウと個別ウインドウで重複したジェスチャーが使用される場合、個別ウインドウでのみ実行されます。
- ジェスチャーを実行している際にオブジェクトをタッチした場合、オブジェクトが依然にトリガーされます。
- 個別ウインドウでのタッチジェスチャーは、ウインドウ属性がベースウインドウである場合のみ実行されます。当該ウインドウがポップアップウインドウ、または下敷きウインドウである場合、タッチジェスチャーが無効になり、関連動作も実行されません。
- トリガーされたジェスチャーに設定された動作が全部実行完了してから、次回のジェスチャーを検知します。

14. 図形ライブラリ及び画像 ライブラリの作成

本章では、図形ライブラリ及び画像ライブラリの作成方法について説明します。

14.1. 概要	14-2
14.2. 図形ライブラリの作成	14-2
14.3. 画像ライブラリの作成	14-10
14.4. 図形ライブラリ/画像ライブラリの即時調整及び管理.....	14-17

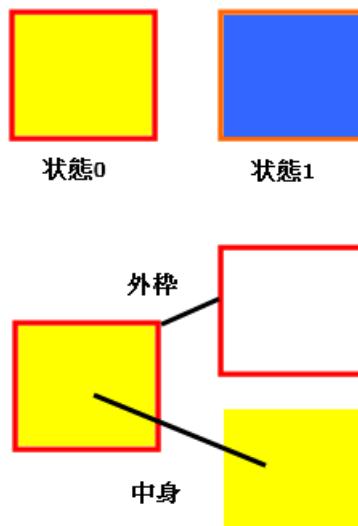
14.1. 概要

EasyBuilder Pro は図形ライブラリ及び画像ライブラリを提供します。画像管理では、[プロジェクト]及び[ライブラリ]二種類のモードを提供しています。[プロジェクト]の画像は.emtp というプロジェクトの中に保存され、[ライブラリ]の画像は EasyBuilder Pro ライブラリ、またはユーザ一定義のディレクトリに保存されます。画像はオブジェクトの視覚効果を豊富にし、一個の図形または画像には最大 256 個状態が含まれます。下記では図形ライブラリや画像ライブラリを作成する方法を説明します。

👉 図形ライブラリ及び画像ライブラリの使用について、《9 オブジェクト一般的な属性》をご参考ください。

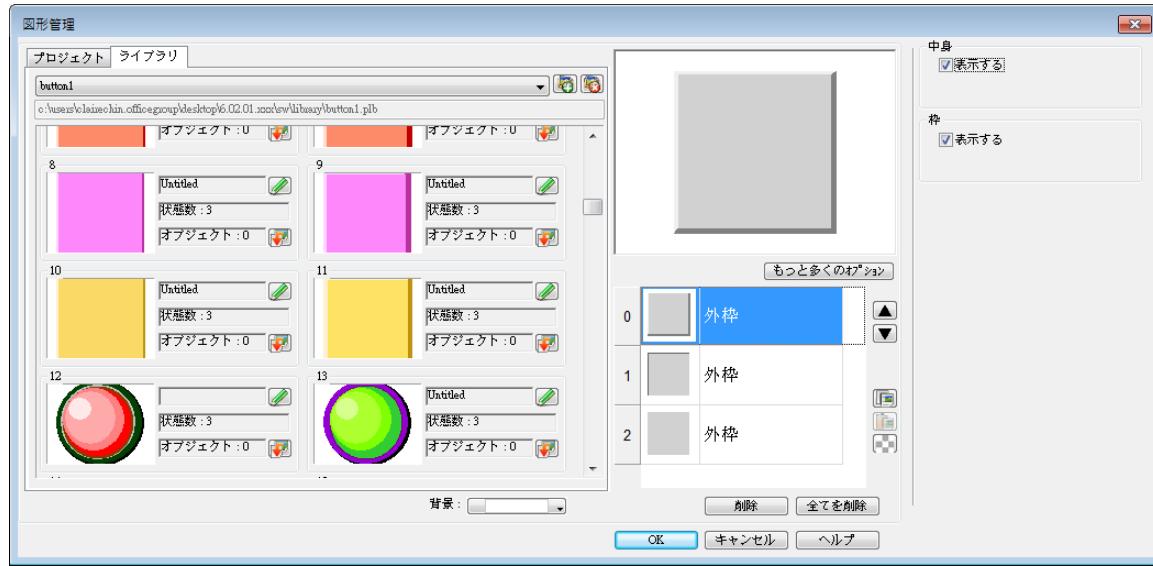
14.2. 図形ライブラリの作成

図形というのは、直線、曲線、多辺形などで構成されたベクトル図形です。一個の完全な図形には、一個以上の状態が含まれる可能性があります。各状態にも下記のように 2 部分が含まれます：外枠と中身。



14.2.1. 図形管理

オブジェクトには単独に外枠、または中身を使用することができ、或いは同時に使用してもいいです。ツールバー上の[プロジェクト]»[図形]ボタンを押せば、下図のように[図形管理]ダイアログボックスがオープンされます。



設定

記述

プロジェクト

このタブで編集する図形は.emtp プロジェクトファイルに保存され、最大 1000 個の図形を新規作成することができます。

ライブラリ

このタブで編集する図形はライブラリディレクトリに保存されるが、.emtp プロジェクトに保存されません。

ライブラリを新規作成

既存の.plb 図形ライブラリから図形を使用するか、または新規図形ライブラリを作成します。新規図形ライブラリを作成したかったら、既存していないファイル名を入力して、[オープン]を押すと、新しい空き図形ライブラリが作成されます。最大 40 個の図形ライブラリを新規追加することができます。

ライブラリを削除

現在選択しているのライブラリを削除します。

プロジェクトにコピーする

この図形を[プロジェクト]にコピーします。システムライブラリにない図形だけがコピーできます。下記 4 種類の図形をコピーすることができません：System Frame/System Button/System Lamp/System Pipe。

背景

図形の背景色を選択します。この色は[図形管理]の中にだけ表示し、実際の画面では背景色がありません。

図形ライブラリ及び画像ライブラリの作成

もっと多くのオプション [中身]、[外枠]及び[模様]の色やスタイルを設定することができます。

図形を一個の状態の前へ/後へ移動します。

コピー 図形をコピーします。

貼り付け コピーされた図形を貼り付けます。

透明状態を挿入する 図形の後に透明な状態を挿入します。

削除 この図形のある状態を削除します。

全てを削除 選択した図形の全ての状態を削除します。

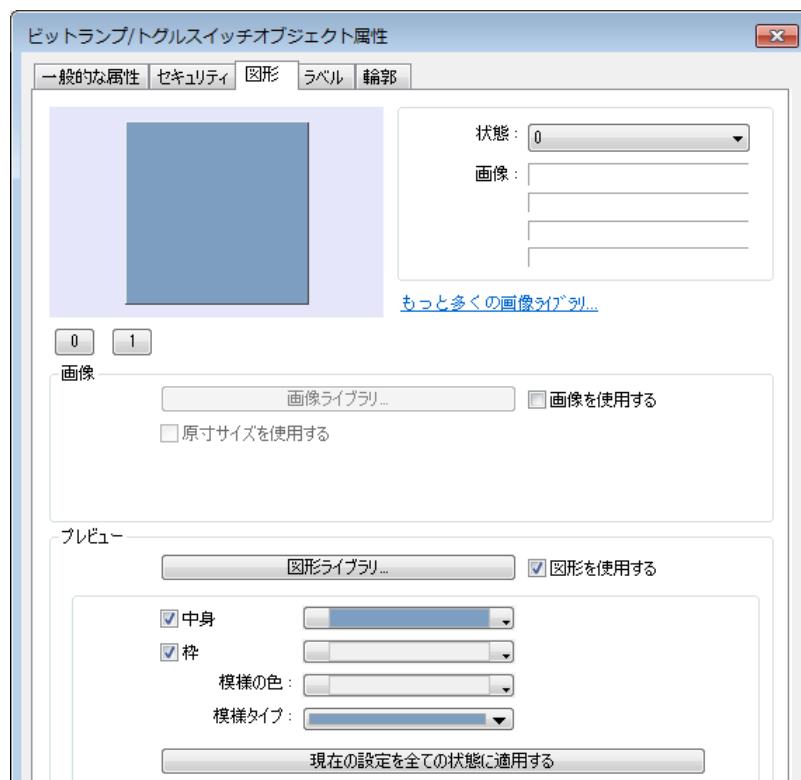
OK 今回の編集結果をセーブします。

キャンセル 今回の編集結果をキャンセルします。

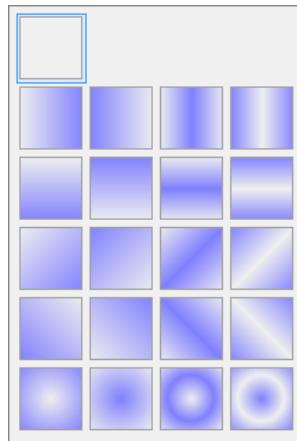
ヘルプ ヘルプをオープンします。

Note

- 図形ライブラリでは[中身]、[外枠]の色、及び[模様スタイル]を選択する機能は System Frame / System Button ライブラリのみ使用できます。



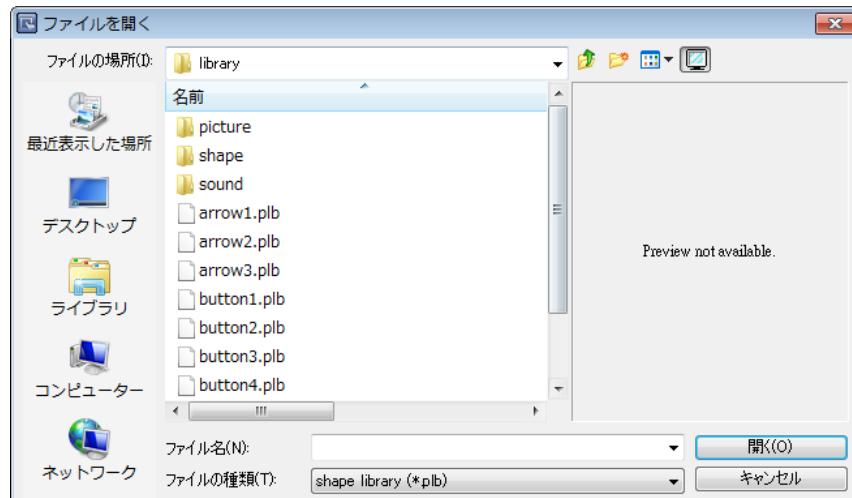
- cMT シリーズの[模様スタイル]機能はグラデーション模様をサポートします。下図のように示しています：



14.2.2. 図形ライブラリを作成する手順

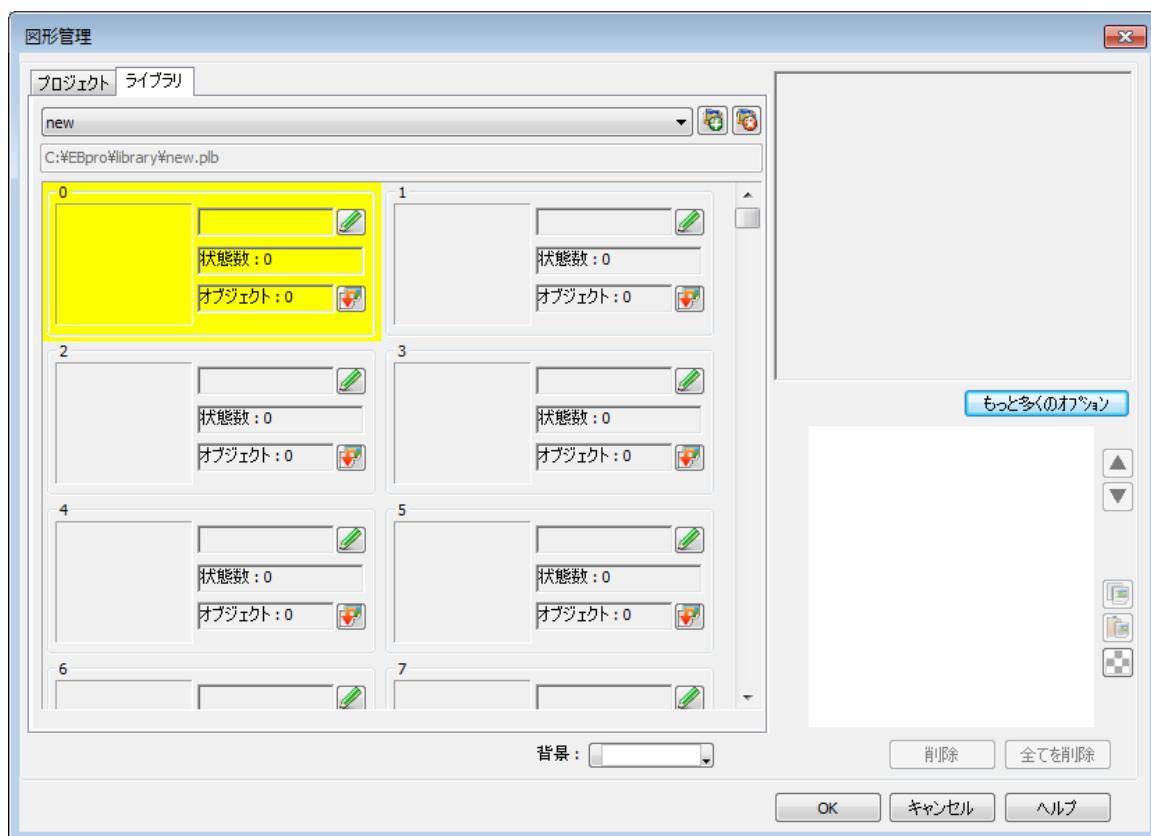
以下は図形ライブラリを新規作成し、またその中に 2 個の状態を有するベクトル図形を追加する方法を説明します。

1. [ライブラリを新規作成する]を押し、ダイアログボックスに新しい図形ライブラリ名 “new_lib”を入力します。

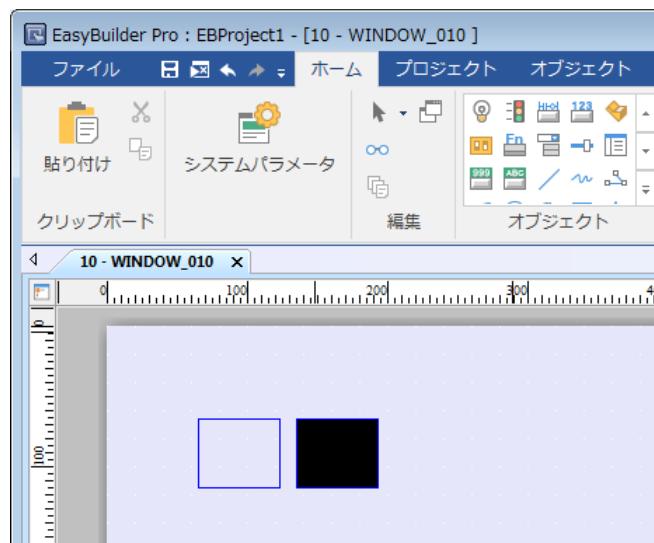


2. [開く]をクリックしたら、確認メッセージウィンドウがポップアップされます。[はい]を押して次へ進みます。
3. それで[図形管理]で新しい図形ライブラリ [new_lib]が新規作成され、ライブラリ名の下に本ライブラリのファイルパスが表示されます。下図のように、この時、新規ライブラリにはまだ図形が入っていません。

図形ライブラリ及び画像ライブラリの作成



4. 特定した図形に、一個の状態を挿入します。まずはウインドウに作画ツールで所要する外枠や中身を書き、そしてその外枠を選択し、図形ライブラリに追加します。



5. 図形オブジェクトを選択した後、ツールバー上の[プロジェクト] » [図形] » [図形ライブラリに保存する]ボタンを押し、既に作成した[new_lib]ライブラリを選択し、そして保存する図形の番号を選択します。選択された図形には黄色の背景色が表示されます。
6. 続いてはこの図形を本状態の[外枠]に設定し、オプションで[插入]を選択して[保存する]をクリックします。



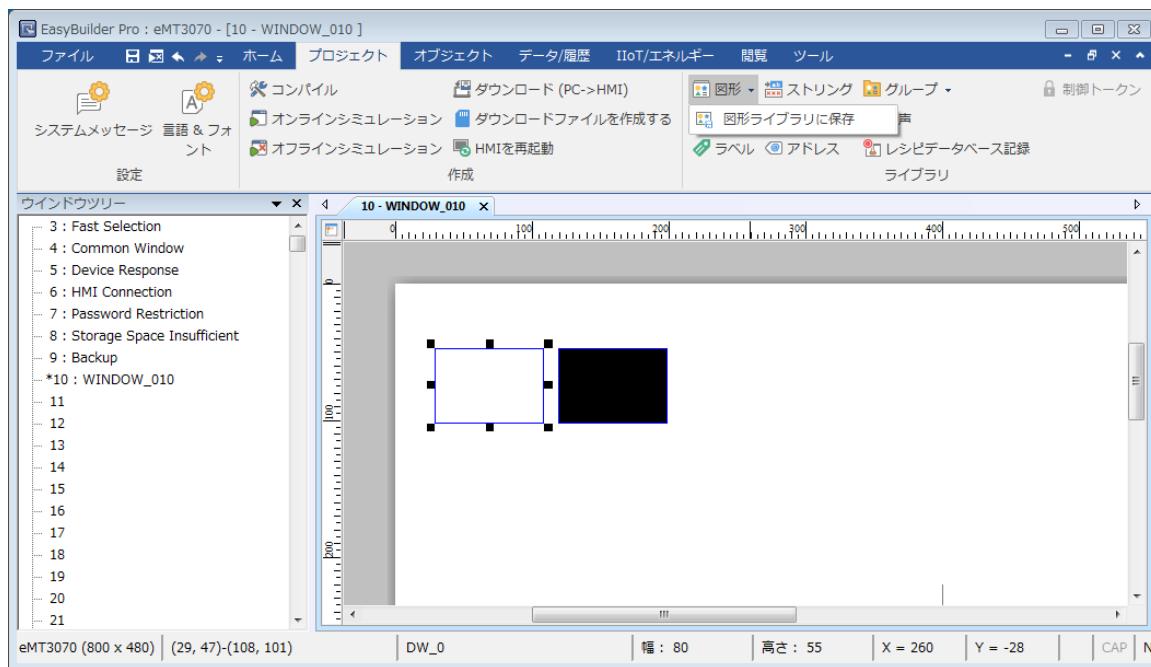
設定	記述
中身	図形の中身を表示するかを設定します。
外枠	図形の外枠を表示するかを設定します。
ライブラリに保存	外枠に保存 図形外枠に保存します。 中身に保存 図形中身に保存します。 挿入 一個の新しい状態を挿入します。 置換 図形の状態を入れ替えます。
保存する	上記設定をセーブします。

7. 下図では図形オブジェクトに一個の状態が新規追加されたのが見られました。その外枠が定義されました。

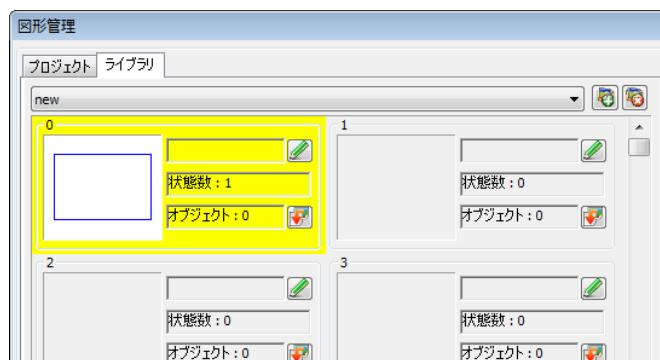


8. 図形の中身を設定します。画いた図形中身を選択します。

図形ライブラリ及び画像ライブラリの作成

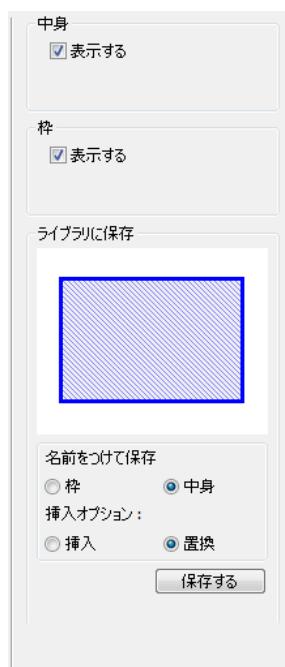


9. ツールバー上の[図形ライブラリに保存する]ボタンを押し、既に作成した[new_lib]ライブラリを選択し、そして外枠と同じの図形番号を選択して保存します。

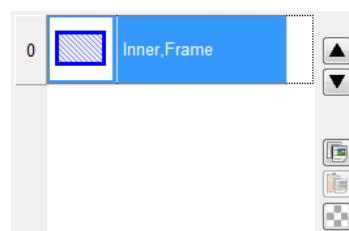


10. このベクトル図形を本状態の[中身]に設定し、[插入]で[置換]を選択した後、[保存する]をクリックします。

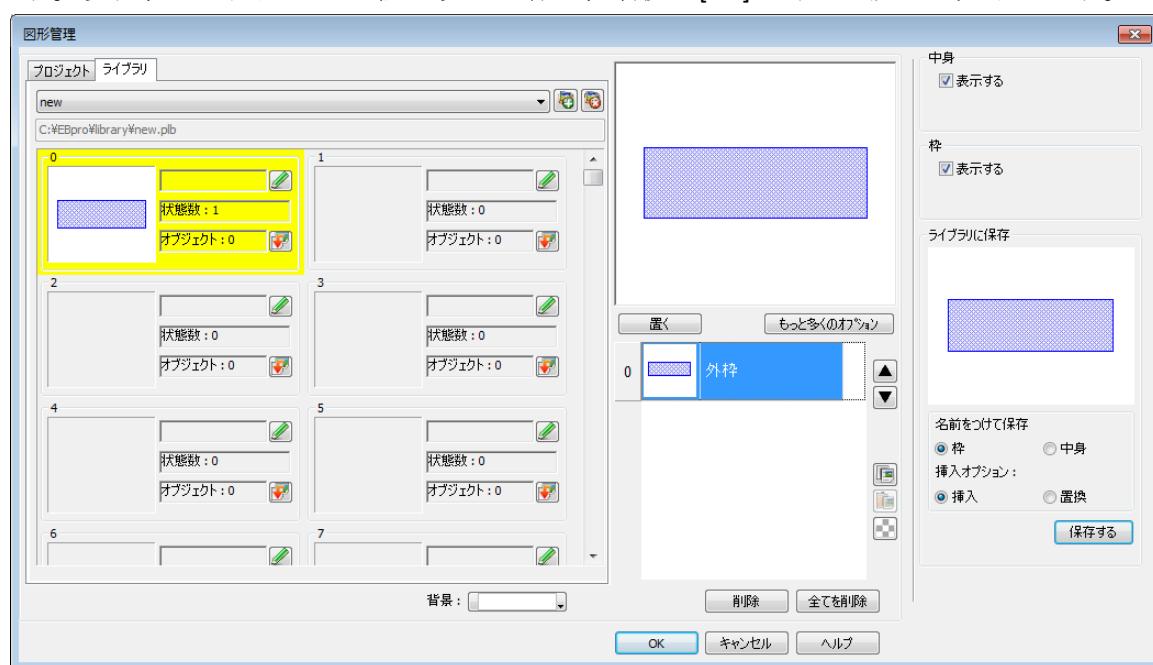
図形ライブラリ及び画像ライブラリの作成



11. 一個の状態が有するベクトル図形は[中身]のみ、または[外枠]のみあってもいいし、同時に存在するのもいいです。下図では状態 0 の図形は[中身]と[外枠]の設定を有していると見られます。[OK]を押せば、状態 0 図形の作成が完成しました。



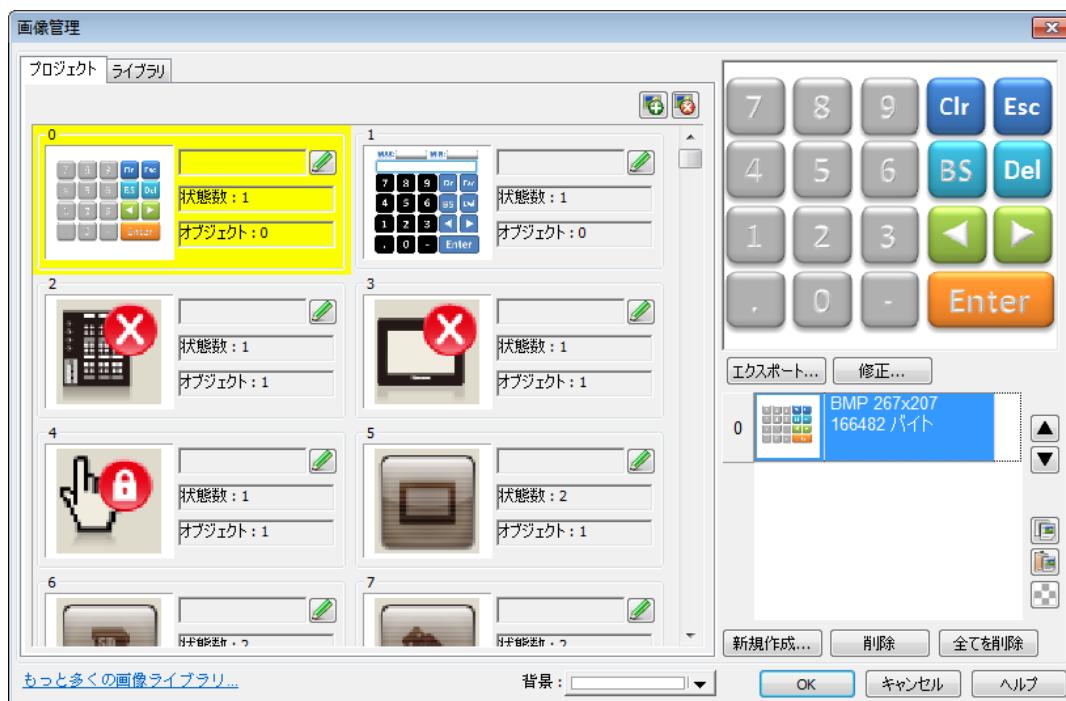
12. 状態 0 と同じの方式を使用し、下図のように 1 個のベクトル図形を状態 1 として挿入します。現在、この図形では 2 個の状態を有し、最後に[OK]を押せば設定を完了します。



14.3. 画像ライブラリの作成

14.3.1. 画像管理

ツールバー上の[プロジェクト] » [画像]ボタンを押せば、下図のように[画像管理]ダイアログボックスが現れます。

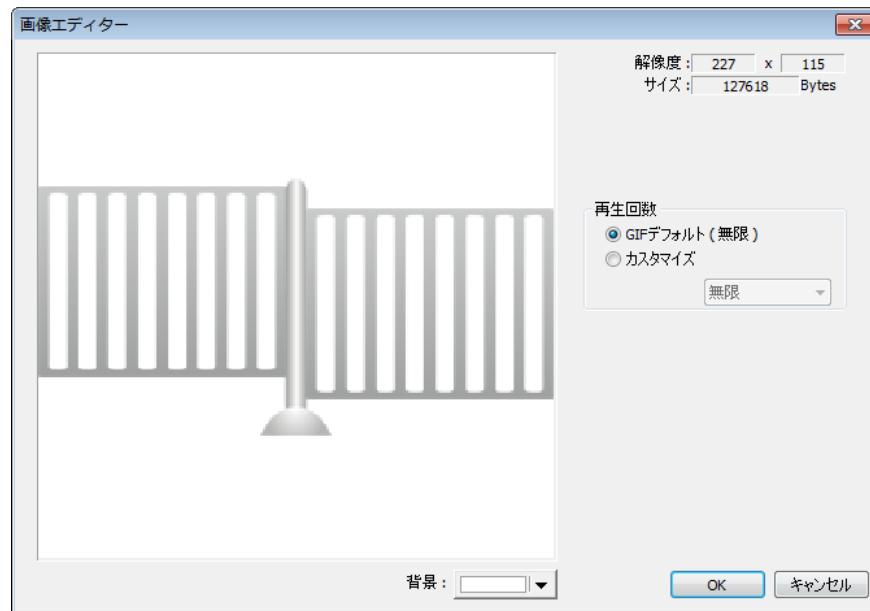


設定	記述
プロジェクト	このタブで編集する図形は.emtp プロジェクトファイルに保存され、最大 1000 個の画像を新規作成することができます。
ライブラリ	このタブで編集する画像はのライブラリディレクトリに保存されるが、.emtp プロジェクトに保存されません。
ライブラリを新規作成	既存の .flbx & .flb 画像ライブラリから画像を使用するか、または新規画像ライブラリを作成します。新規画像ライブラリを作成したかったら、既存していないファイル名を入力して、[開く]を押すと、新しい空き画像ライブラリが作成されます。最大 40 個の画像ライブラリを新規追加することができます。
ライブラリを削除	現在の画像ライブラリを削除します。

 ライブラリを参照	複数のパスを設定してから、ライブラリの内容をプレビューすることができます。
 プロジェクトにコピーする	この画像を[プロジェクト]にコピーします。
背景	画像の背景色を選択します。この色は[画像管理]の中にだけ表示し、実際の画面では背景色がありません。
もっと多くの画像ライブラリ	Weintek オフィシャルサイトの画像ライブラリダウンロードアドレスにリンクし、アカウントをログインしたら更なるの画像ライブラリをダウンロードすることができます。
エクスポート	画像をエクスポートします。
変更	画像の設定を変更します。
	画像を一個の状態の前へ/後へ移動します。
 コピー	画像をコピーします。
 貼り付け	コピーされた画像を貼り付けます。クリップボード上の画像に[貼り付け]機能を使って、画像ライブラリにインポートします。
 透明状態を挿入する	画像の後に透明な状態を挿入します。
新規作成	画像を新規作成します。
削除	この画像を削除します
全てを削除	全ての画像を削除します。
OK	今回の編集結果をセーブします。
キャンセル	今回の編集結果をキャンセルします。
ヘルプ	ヘルプをオープンします。

 **Note**

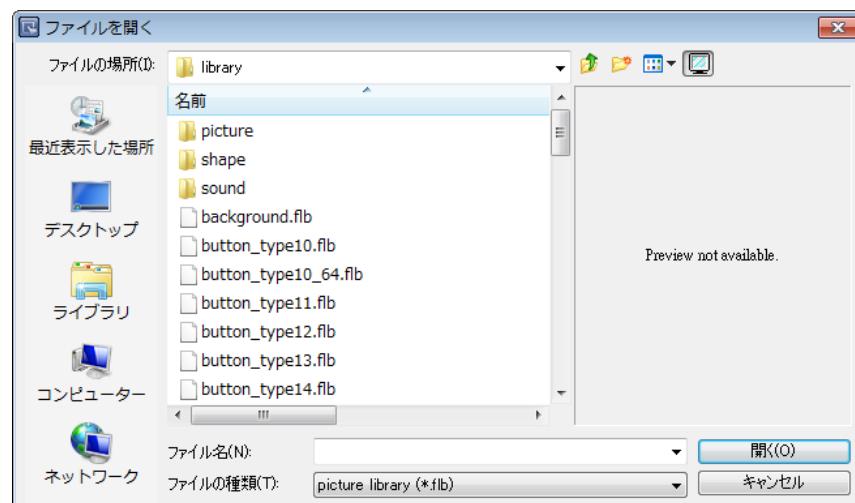
- 画像ライブラリがサポートする画像フォーマットは.bmp、.jpg、.gif、.dpd、.svg と.png です。画像ライブラリに.gif フォーマットの画像ファイルを新規作成する際に、画像がアニメーションに属した場合、下図に示したようにユーザーはアニメーションの再生回数を設定することができます。



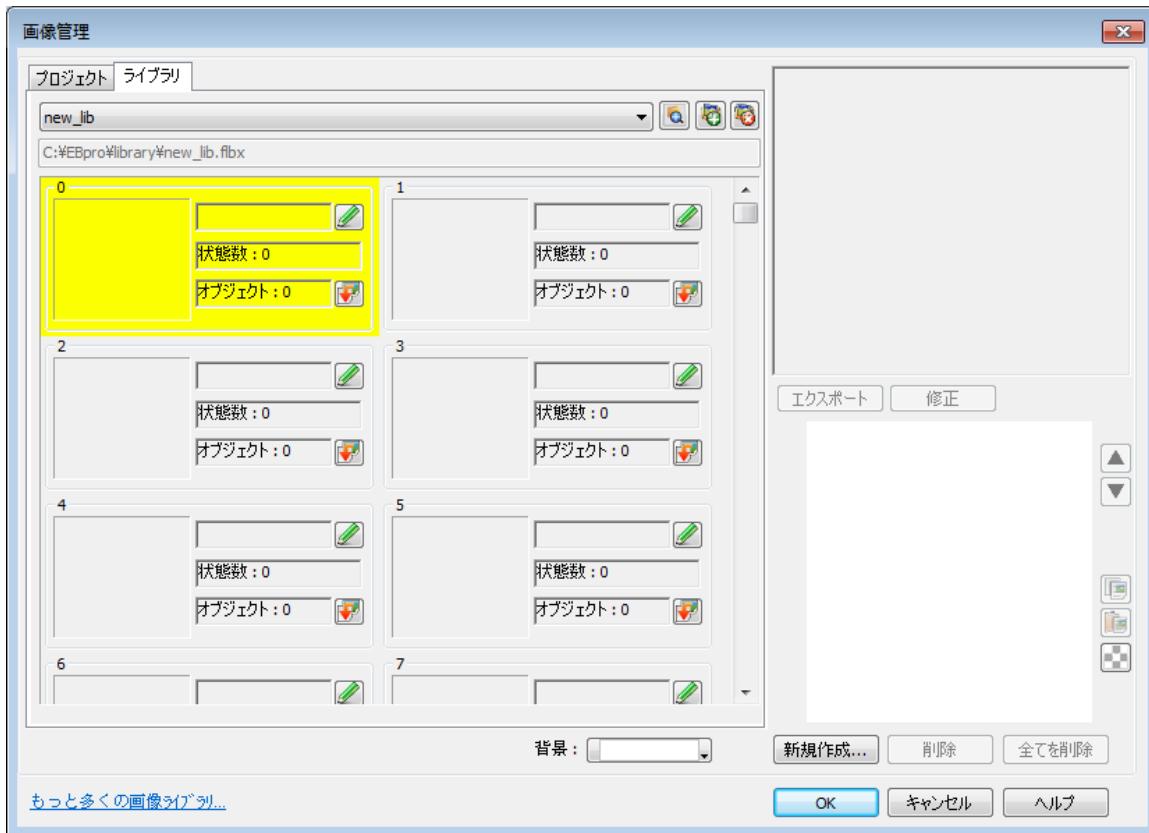
14.3.2. 画像ライブラリを新規作成する手順

以下は画像ライブラリを新規作成し、またその中に2個の状態を有する画像を追加する方法を説明します。

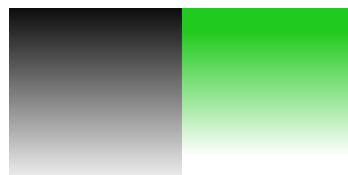
1. [ライブラリを新規作成]を押し、ダイアログボックスに新しい画像ライブラリ名を入力します。



2. [開く]をクリックしたら、確認メッセージウィンドウがポップアップされます。[はい]を押して次へ進みます。
3. それで[画像管理]で新しい画像ライブラリ[new_lib]が新規作成され、下図のようにこの新規ライブラリにはまだ画像が入っていません。



4. 仮に下記 2 枚の画像を状態 0 と状態 1 にする場合 :

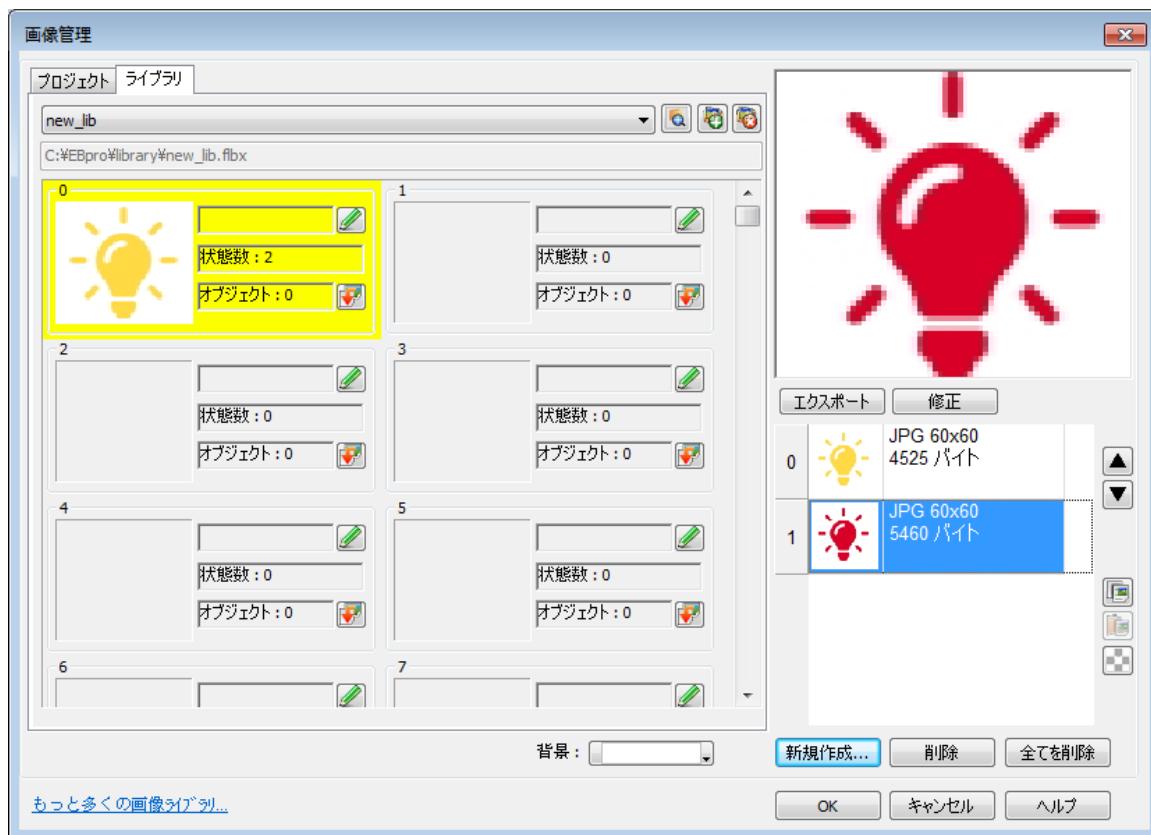


5. まずは作成された[new_lib]画像ライブラリを選択し、そして保存する画像の番号を選択します。選択された画像番号には黄色の背景色が表示されます。
6. [新規作成]を押した後、状態 0 画像のデータソースを選択します。
7. ここで、下図のような透明色設定ダイアログボックスが現れた際に、[使用する]にチェックマークを入れ、そして RGB(255,255,255)を設定した場合、このエリアが透明に設定されます。或いはマウスで透明にしたい位置をクリックすれば、システムは自動的に RGB 数値を判断します。

図形ライブラリ及び画像ライブラリの作成



8. これで状態 0 の画像設定が完了しました。続いて状態 1 の画像を状態 0 と同じように[新規作成]します。下図に示す通り、[画像管理]ダイアログボックスで新規作成された画像は番号 0 画像と見られ、画像情報の中でもこの画像は bitmap フォーマットで 2 個の状態が含まれていると見られます。



9. 上記の各動作を完了後、[OK]を押せば、一個の完全な画像が作成されます。

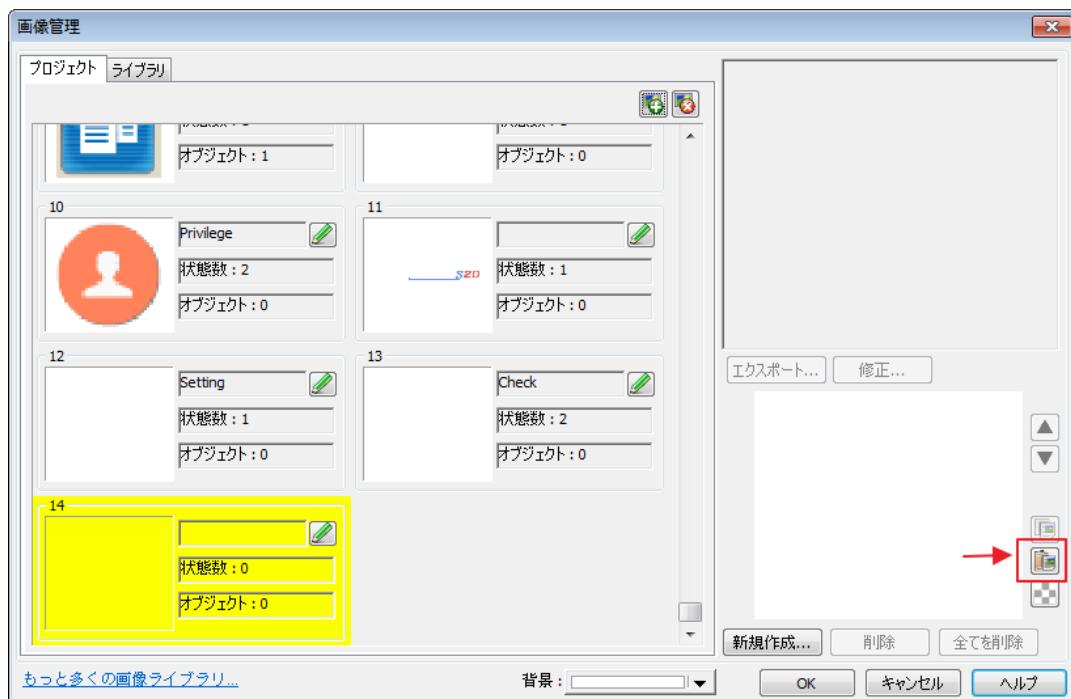
14.3.3. クリップボードから画像をインポートする

以下ではクリップボードから画像を画像ライブラリにインポートする方法を説明します。

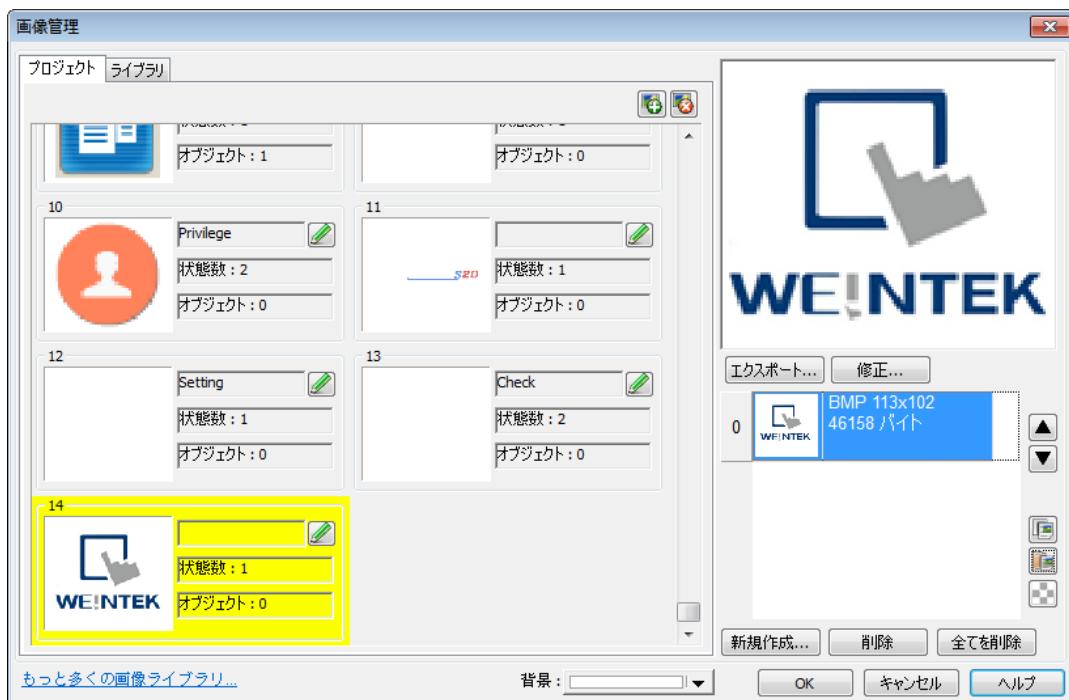
1. 仮にクリップボードには下記画像がある場合：



2. 画像ライブラリ右側の[貼り付ける]ボタンをクリックします。



3. これで簡単に画像をインポートすることができます。

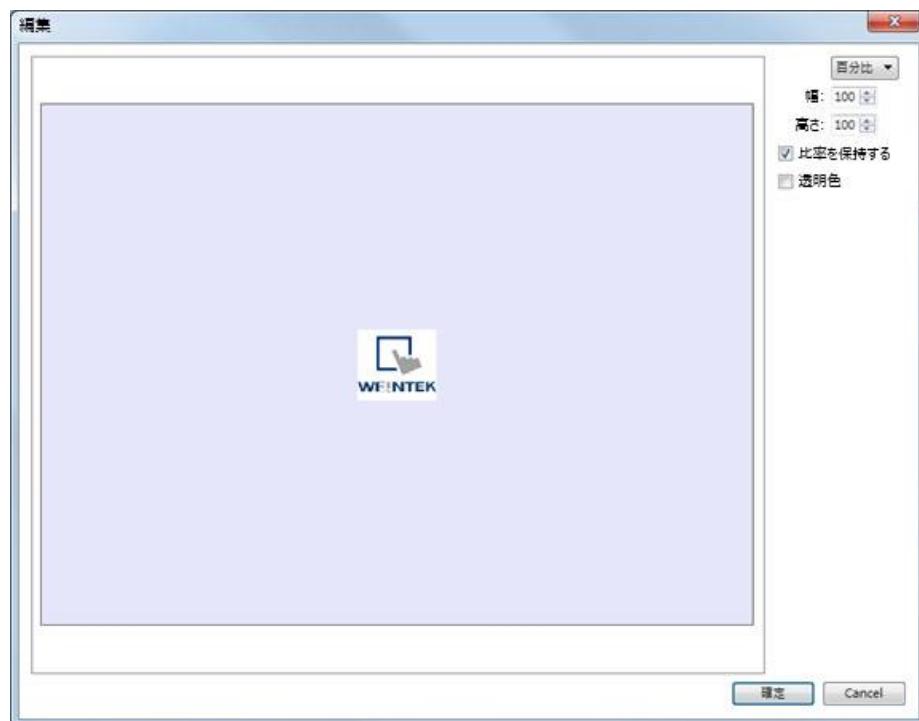


また、直接にプロジェクトの編集ウインドウにクリップボードに一時保存された画像を貼り付けることができます。

- 仮に、クリップボードには以下の画像が一時保存されています。



- 編集ウインドウで **Ctrl + V** を押すと、以下のウインドウがポップアップされます。



- [OK]をクリックした後、画像がウインドウに置かれます。
- 画像をクリックして属性を確認すれば、当該画像が[プロジェクト]に保存されたと見られます。

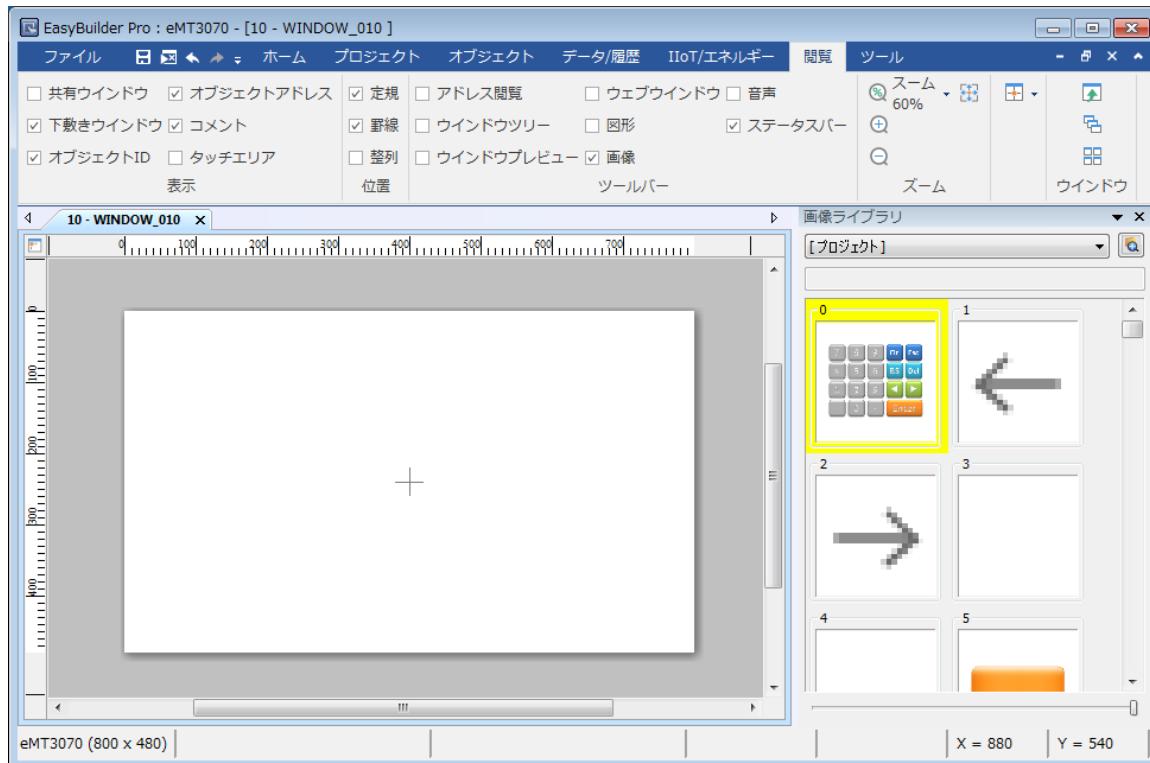
Note

- **.bmp**、**.dpd** 及び **.jpg** ファイルだけが透明色を使用できます。

14.4. 図形ライブラリ/画像ライブラリの即時調整及び管理

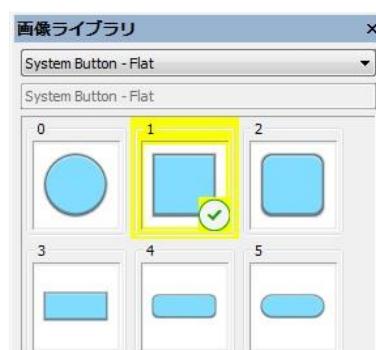
図形ライブラリ、画像ライブラリの設定ウインドウを使用すれば、即時に一枚、または複数の画像を一斉に修正することができます。

一部のライブラリは迅速に図形の色を変更することをサポートします。



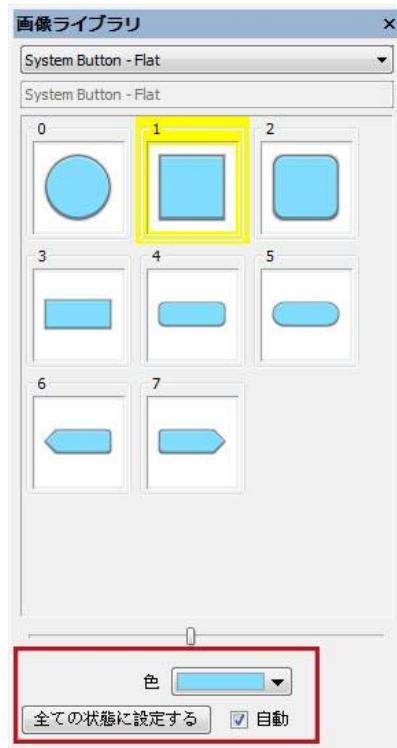
14.4.1. 画像の置換

1. 画像を置換するオブジェクトを選択します。一個、または複数のオブジェクトを選択することができます。
2. 画像ライブラリの設定ウインドウで使用したい新画像を選択し、右下隅をクリックして適用します .

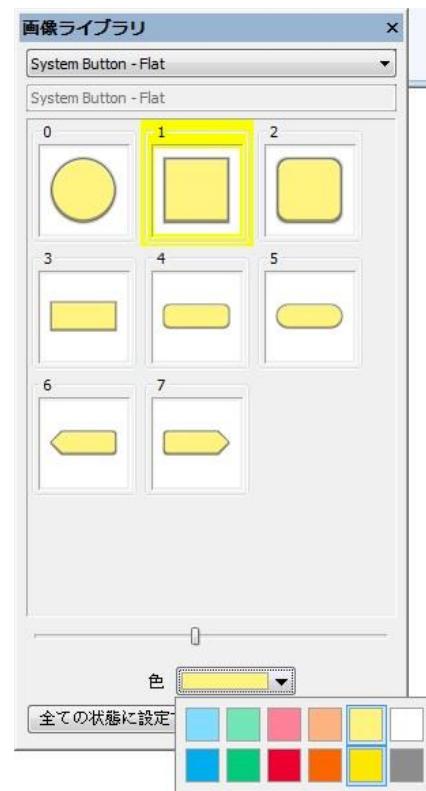


14.4.2. 色の置換

一部のライブラリには迅速に色を置換する機能をサポートします。本機能をサポートするライブラリの下方には、[色]の設定欄があります。



1. 色を置換するオブジェクトを選択します。
2. ライブラリ設定ウインドウの[色]ドロップダウンリストで色を選択します。

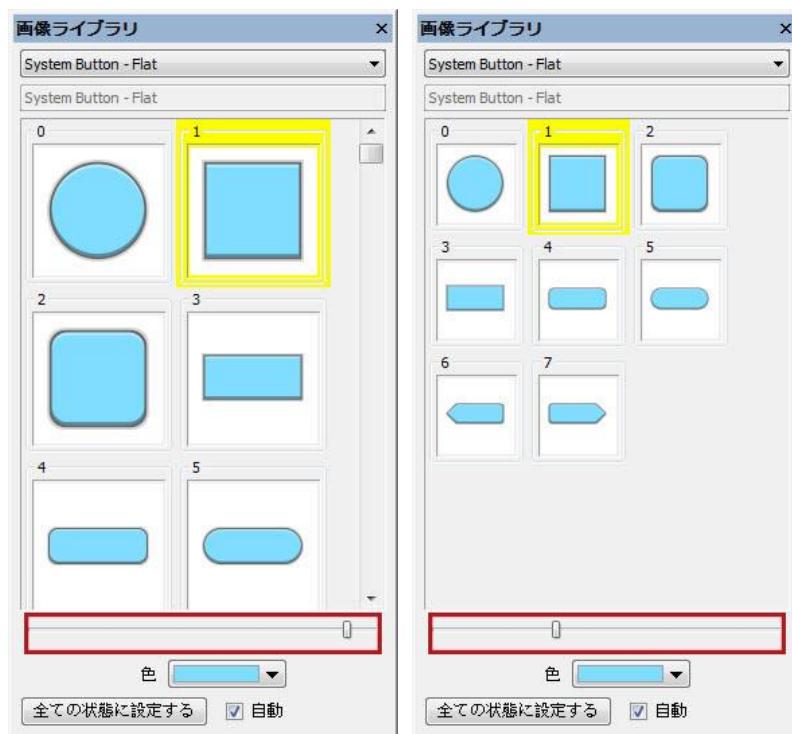


3. 使用したい新画像の右下角をクリックして適用します✓。

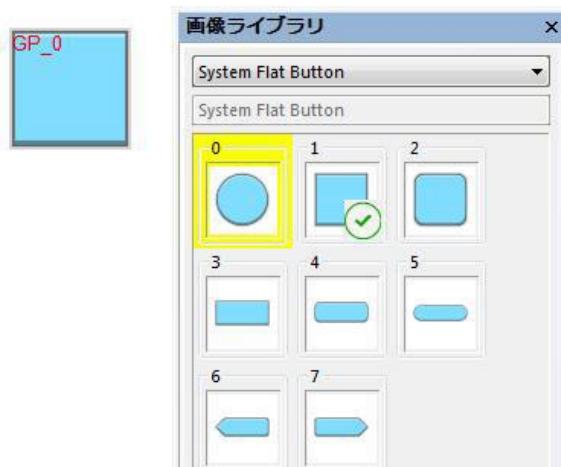
設定	記述
現在の設定を全ての状態に適用する	選択した色を本オブジェクトの全ての状態に適用します(同一画像を選択する必要があります)。
自動	自動的に[設定を全ての状態に適用する]機能を実行します。

14.4.3. 付加機能

- プレビューウィンドウにはスライドバーをスライドすることで、画像のサムネイルの大きさを調整できます。

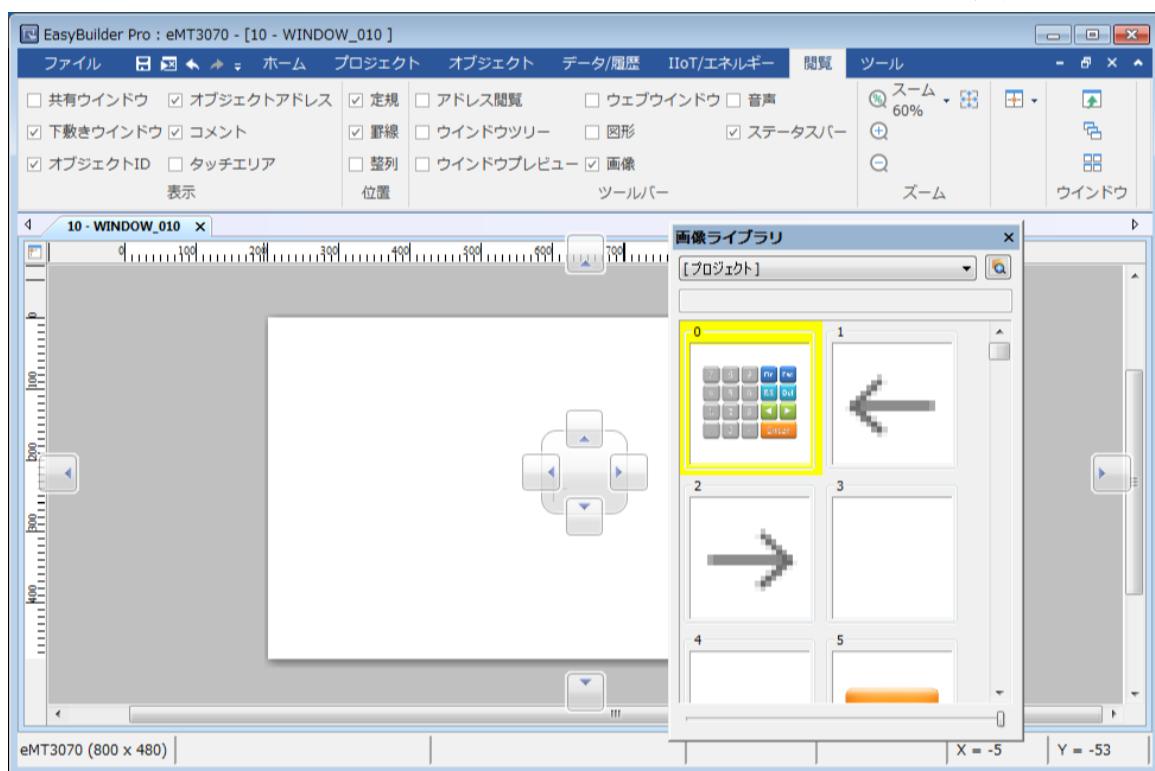


- 即時に画像がオブジェクトに適用された結果をプレビューできます。ステップは以下のとおりです：
 - オブジェクトを選択します。
 - キーボードの[Shift]キーを押し続けます。
 - 画像ライブラリ管理ウィンドウを参照して使いたい画像を選択します。画像は一時にオブジェクトに適用されます。



14.4.4. ウィンドウ位置の調整

ユーザーはドラッグ&ドロップ機能で管理ウィンドウを好きな場所に移動させることができます。ウィンドウをドラッグすると、画面に半透明なエリアが現れてドッキング位置を示します。



15. ラベルタグライブラリ 及び多言語の使用

本章では、ラベルタグライブラリの作成及び使用する方法について説明します。

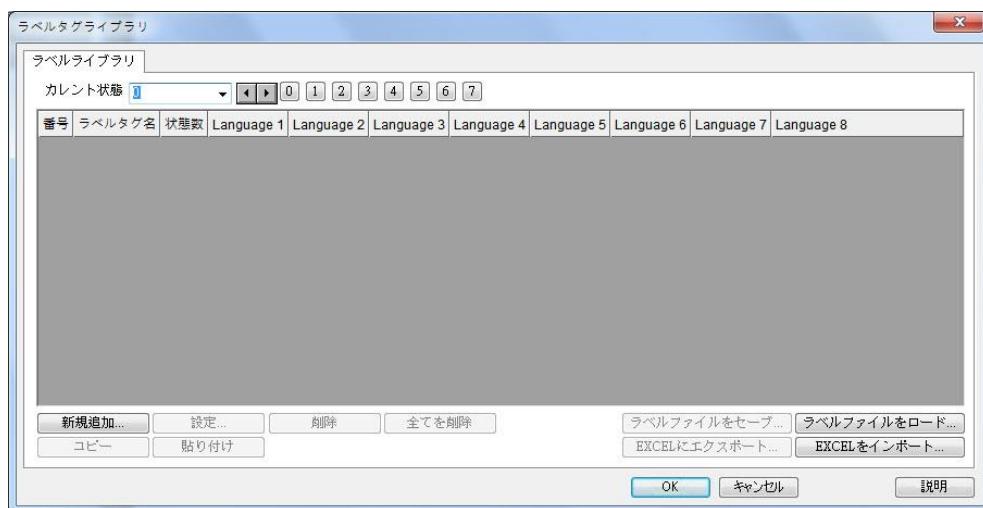
15.1.	概要	15-2
15.2.	ラベルタグライブラリの管理	15-2
15.3.	ラベルタグライブラリの作成	15-3
15.4.	ラベルタグライブラリの使用	15-4
15.5.	言語とフォント	15-5
15.6.	多言語の使用	15-9

15.1. 概要

プロジェクトファイルに多言語を使用する必要がある場合、事前にラベルタグライブラリを作成できれば、後でその中から所要するラベルを選択し使用することができます。HMIが運転する時には、言語モードの設定により、プロジェクトで選択した言語モードに対応する文字を画面に表示します。EasyBuilder Proは同時に8種類異なる言語の文字表示をサポートします。本章では、ラベルタグライブラリの作成する方法や、多言語の使い方を説明します。

15.2. ラベルタグライブラリの管理

ツールバー上の[プロジェクト]»[ラベル]をクリックしたら、下図に示す通り[ラベルタグライブラリ]ダイアログボックスに入ります。



設定	記述
カレント状態	一個のラベルタグには最大256個の状態を持つことが可能です(0~255)。 状態数は[言語数]に限られます。1~3種類の言語を使用すれば、一種の言語では最大256個状態を持つことができ、4種類以上の言語を使用する場合、768で言語数を割った結果は、一種の言語の最大状態数になります。 例:言語数は24の場合、 $768/24=32$ (状態数)。
新規作成	ラベルタグを新規作成します。
設定	選択しているラベルタグの内容を設定します。
ラベルファイルをセーブ	全てのラベルタグを.lblフォーマットのファイルにセーブします。
ラベルファイルをロード	既存のラベルタグ.lblファイルをラベルタグにロードします。

EXCEL にエクスポート	全てのラベルタグを.csv, .xls または.xlsx フォーマットのファイルにセーブします。
EXCEL をインポート	既存のラベルタグの.csv, .xls または.xlsx ファイルをラベルタグライブラリにインポートします。

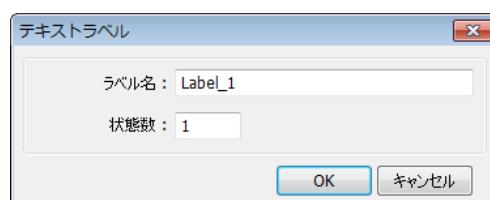
**Note**

- Excel ファイルのインポート/エクスポートは Unicode をサポートしていません。

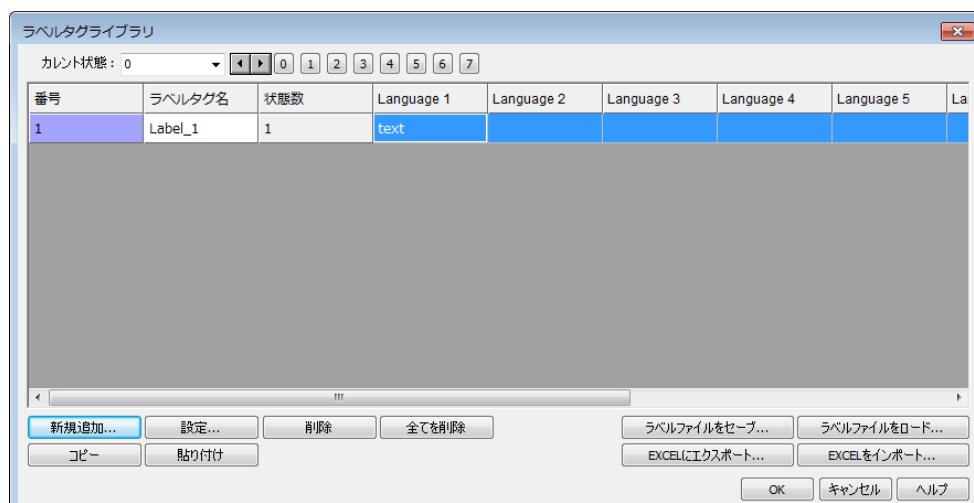
15.3. ラベルタグライブラリの作成

下記手順でラベルタグライブラリを作成します。

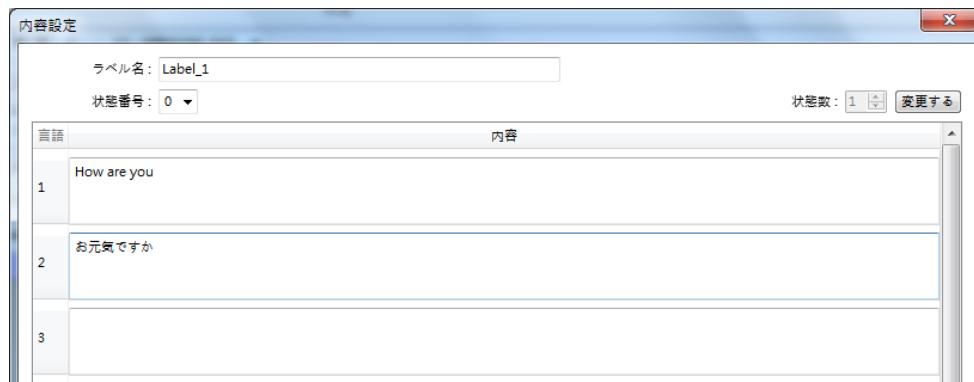
1. [ライブラリ] » [ラベルライブラリ] » [新規作成]をクリックします。ラベルタグの名前を定義し、表現する状態数を設定します。



2. [OK]を押すと、新しい空白ラベルが現れます。そのラベルを選択し、[設定]を押せば、ラベルの内容を編集することができます。

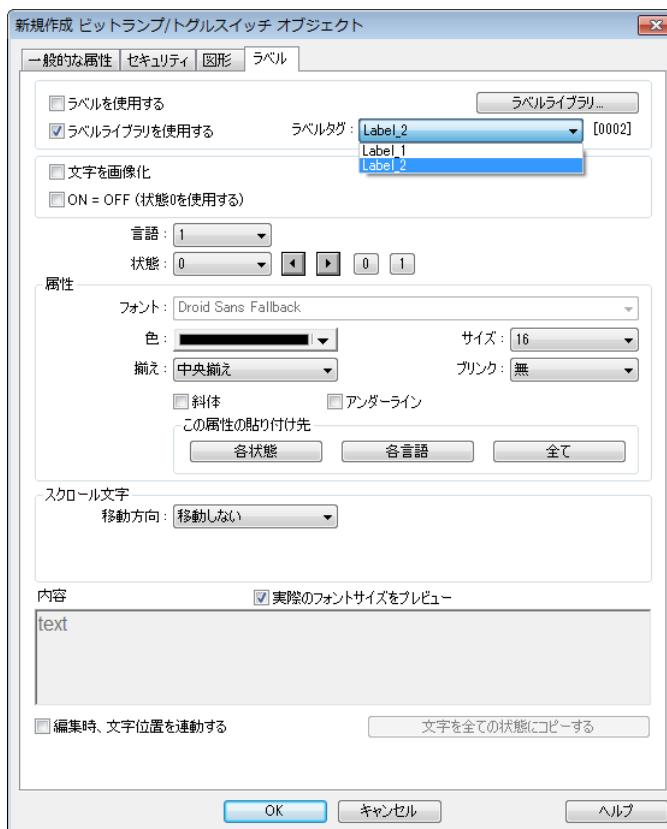


3. 関連言語の内容を設定します。



15.4. ラベルタグライブラリの使用

ラベルタグライブラリに定義されたラベルタグが存在している場合、オブジェクトの[ラベル]タブで[ラベルタグライブラリを使用する]にチェックマークを入れたら、[ラベル]のドロップダウンメニューには定義されたラベルタグが表示されます。

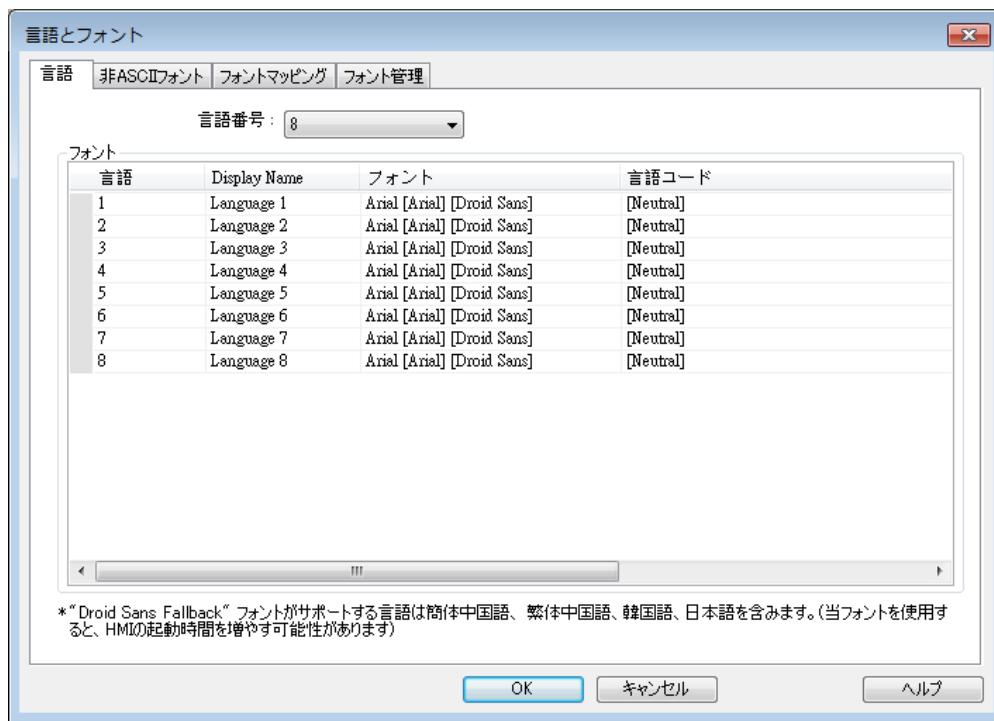


ラベルを選択した後、[内容]には選択したラベルタグの内容が表示され、使用しているフォントはラベルタグライブラリで設定したフォントであることが見られます。注意：言語 2~24 の文字属性設定には文字サイズだけが単独に設定でき、他の属性について、例えば文字の色、揃えとリンクなどは、全部言語 1と同じです。

15.5. 言語とフォント

15.5.1. 言語

ツールバーの[プロジェクト]»[言語&フォント]を押せば、下図の[言語とフォント]設定ダイアログボックスに入ることができます。

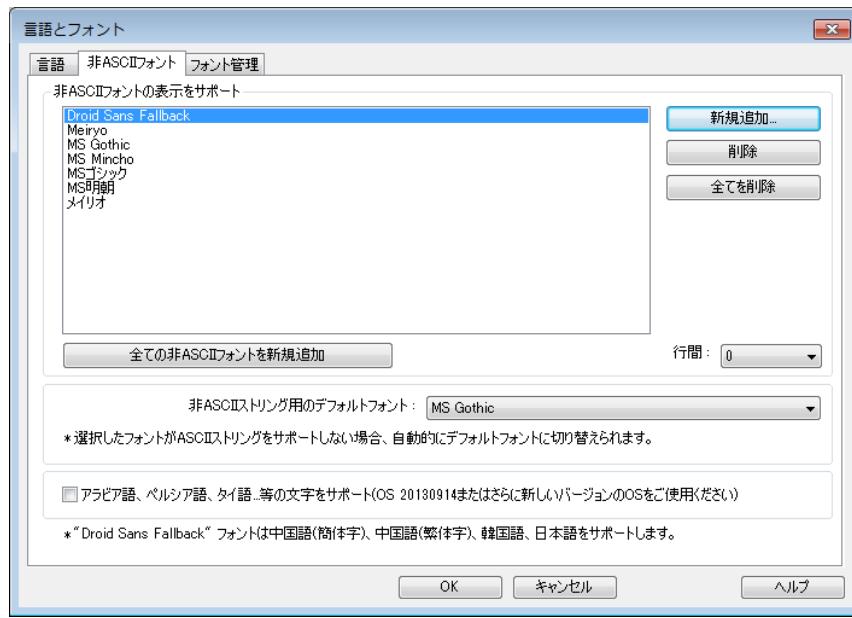


設定	記述
言語番号	ここでプロジェクトでの言語数を設定します。
フォント	<p>現在、既存のラベルが含めている言語のフォントを設定することができます。異なる言語には異なるフォントを選択でき、またもし必要であれば、各フォントに注釈を入れることもできます。[言語コード]はイベントログに[プッシュ通知(EasyAccess 2.0)]を有効にする場合、EasyAccess 2.0 アプリでイベントのプッシュ通知が使用する言語を変更することに用いられます。</p> <p>このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。</p>

15.5.2. 非 ASCII フォント

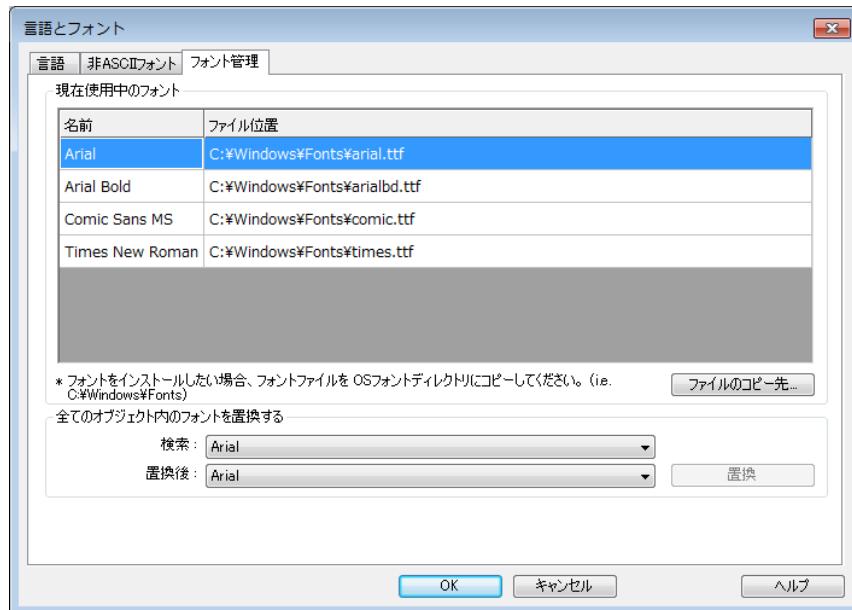
[非 ASCII フォント]設定タブで非 ASCII 文字が使用するフォントを設定します。この項目でリストされたフォントは非 ASCII 文字が使用するフォントです。ユーザーが非 ASCII の文字、或いは二

バイト文字(例：中国語(簡体字)、中国語(繁体字)、日本語、韓国語など)でそれにリストにないフォントを使用する場合、EasyBuilder Pro は自動的にそのフォントをリスト内のフォントに適用します。



設定	記述
全ての非 ASCII フォントを新規追加	WINDOWS の非 ASCII 文字フォントを本リストに追加します。
[行間]	多行の文字を使用する場合、行の間の間隔を調整します。
非 ASCII 文字用のデフォルトフォント	非 ASCII 文字を使用する際に、EasyBuilder Pro は自動的に[非 ASCII 文字用のデフォルトフォント]に適用します。
アラビア語、ペルシア語、ヘブライ語、タイ語、...等の文字をサポート	チェックマークを入れると、これらの文字が正確に表示されます。

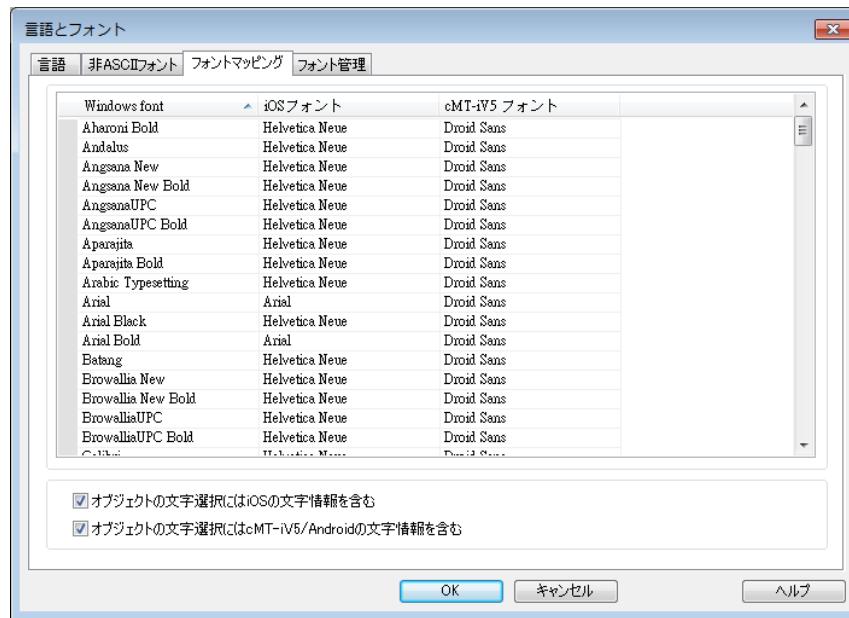
15.5.3. フォント管理



設定	記述
現在使用中の フォント	現在のプロジェクトに使用されている全てのフォント及び フォントファイルの位置が表示されます。
ファイルのコ ピー先	フォントファイルを指定の場所にコピーします。
全てのオブジ エクト内 のフォントを置換 する	本機能を使用すれば、全てのオブジェクト内のフォントを一 回で変更することができます。

15.5.4. フォントマッピング

[フォントマッピング]設定タブには、WINDOWS / cMT3000 で使用するフォントと、iOS / cMT-iV5 / iV6 / Android 装置で表示される対応フォントをリストにまとめます。

**設定****記述**

オブジェクトの文字選択には
iOS/cMT-iV5/iV6/Android
の文字情報を含む

有効にすると、フォントを選択する時に、対応している
iOS/cMT-iV5/iV6/Android フォントが[フォント]欄に表示されます。



15.6. 多言語の使用

オブジェクトの文字内容に多言語を表示する必要がある場合、ラベルタグを使用する以外、システムレジスタ[LW-9134：現在使用中の言語]を合わせて使用する必要があります。

[LW-9134]の有効設定値範囲は0~23(cMTシリーズは0~7)で、異なる数値は異なる表示言語に対応しています。

ダウンロードしたファイルをコンパイルする際に、[全ての言語]にチェックマークを入れなかつたら、[LW-9134]の使い方は変わります。

例えば、ユーザーがラベルライブラリで下記5種類の言語を作成しました：

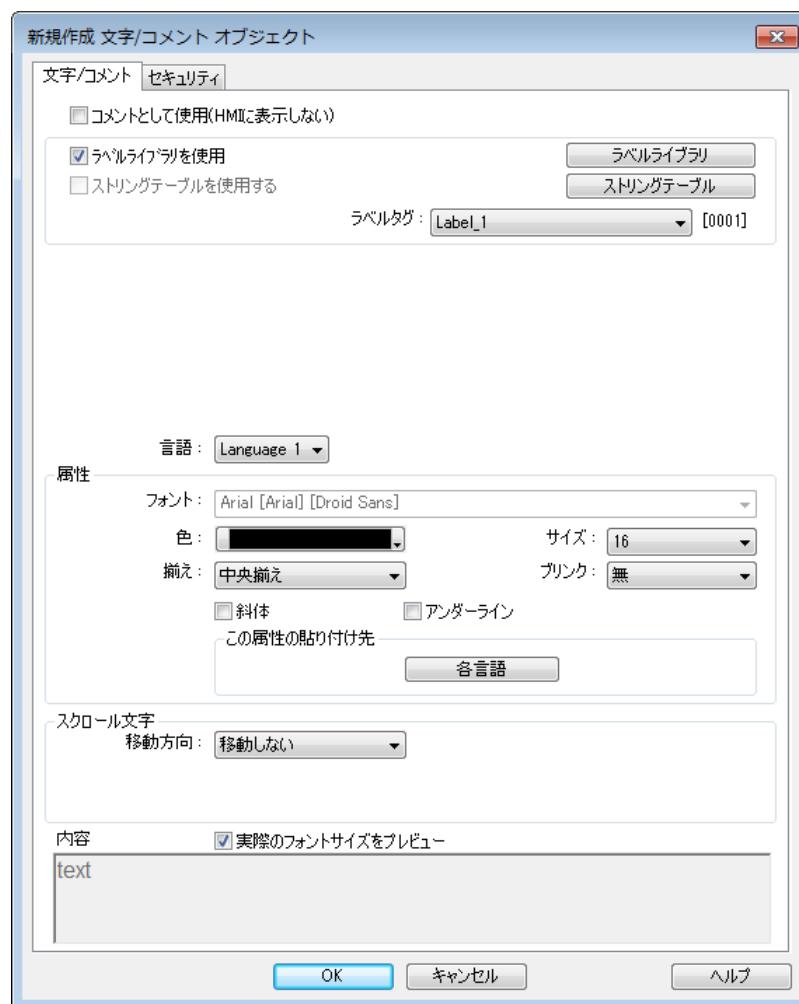
1:英語 2:中国語(繁体字)3:中国語(簡体字)4:フランス語 5:韓国語

コンパイルする際に1, 3, 5だけにチェックマークを入れたら、[LW-9134]の対応数値は：

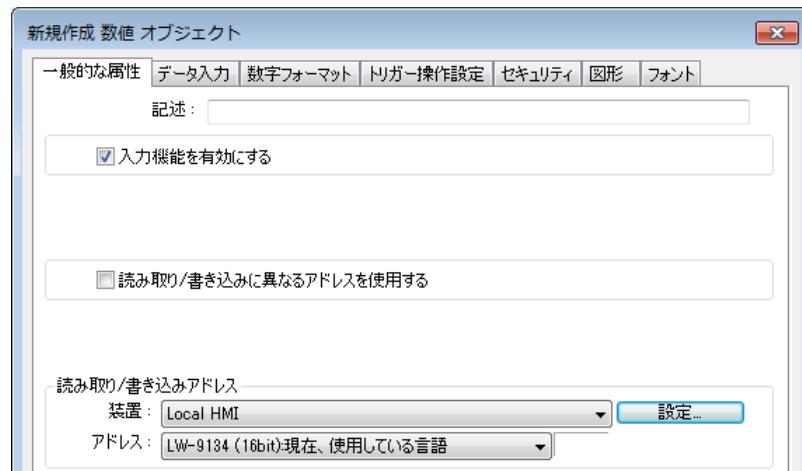
0:英語 1:中国語(簡体字)2:韓国語

下記手順に従って、多言語を使用します。

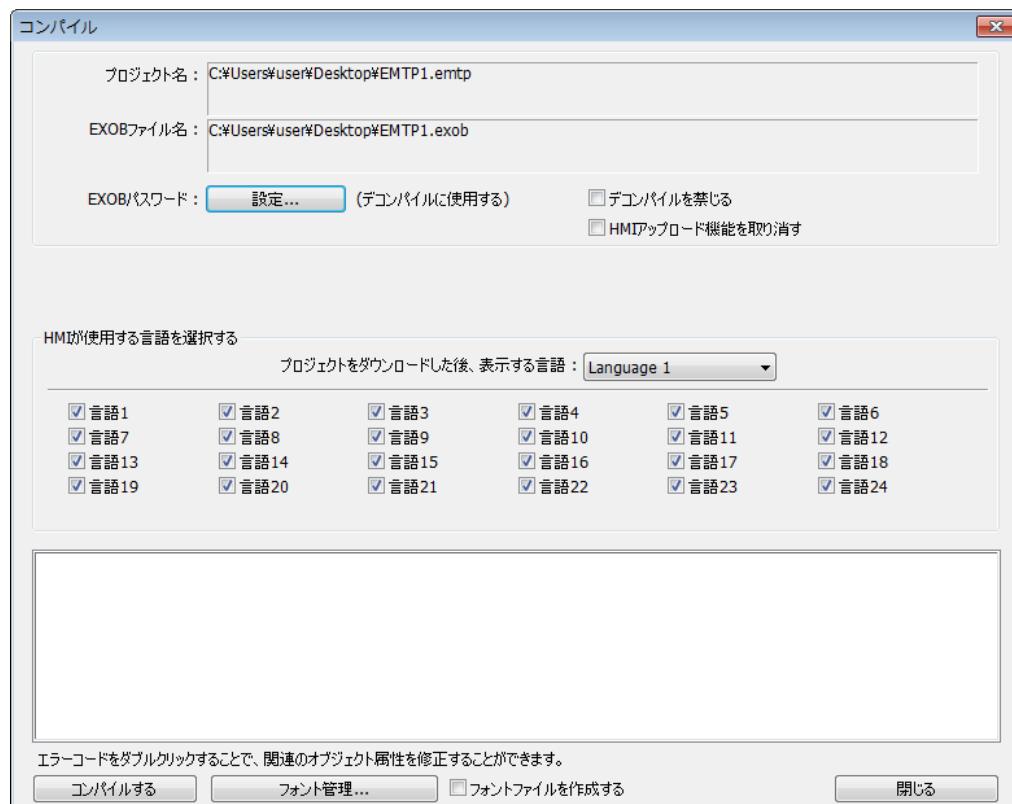
- まずは[文字/コメント]オブジェクトを作成し、[ラベルタグライブラリ]にチェックマークを入れます。



2. そして[数値オブジェクト]を作成し、アドレスをシステムレジスタ[LW-9134]に設定します。



3. コンパイルする際に、所要する定義された言語にチェックマークを入れます。



4. 下記のようにシミュレーションしました。[LW-9134]が変更されたたびに、文字オブジェクトが表示する内容も変更されます。

English

LW9134: 現在使用中の言語

简体中文(SIMPLE)

LW9134: 現在使用中の言語

한국어 웹(KOREAN)

LW9134: 現在使用中の言語

Note

- HMI タイプを cMT-SVR を選択する場合、[LW-9134]はサーバーの言語モードのことを指し、cMT-SVR の言語モードを切り替えることに用いられ、[PLW-9134]はタブレットでの言語を切り替えることに用いられます。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。本サンプルプロジェクトはオプションリストを利用し、多言語に切り替える方法を説明します。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

 このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

16. アドレスタグライブラリ の作成及び使用

本章では、アドレスタグライブラリの作成及び使用する方法について説明します。

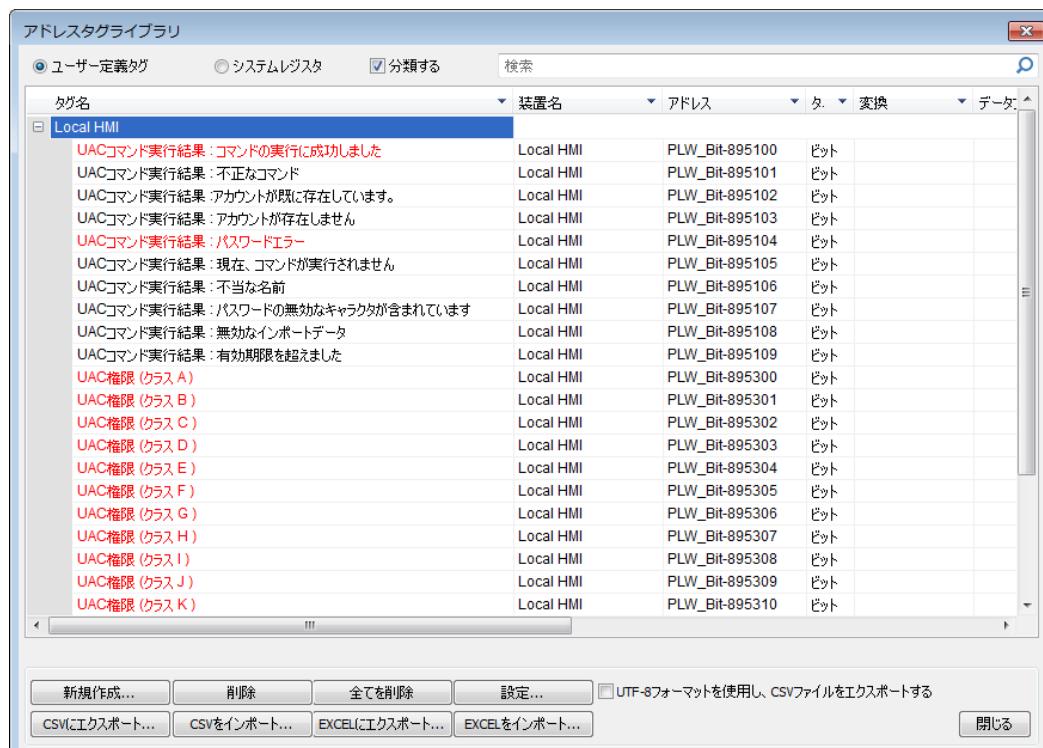
16.1. 概要	16-2
16.2. アドレスタグライブラリの作成	16-2
16.3. アドレスタグライブラリの使用	16-5

16.1. 概要

一般的には、ユーザーがプロジェクトをデザインし始めた際に、前もって常に使用するアドレスをアドレスタグライブラリに定義するのがお勧めです。そうしたら、煩雑なアドレスを入力する手間が省ける以外、オブジェクトアドレス情報の可読性も向上します。

16.2. アドレスタグライブラリの作成

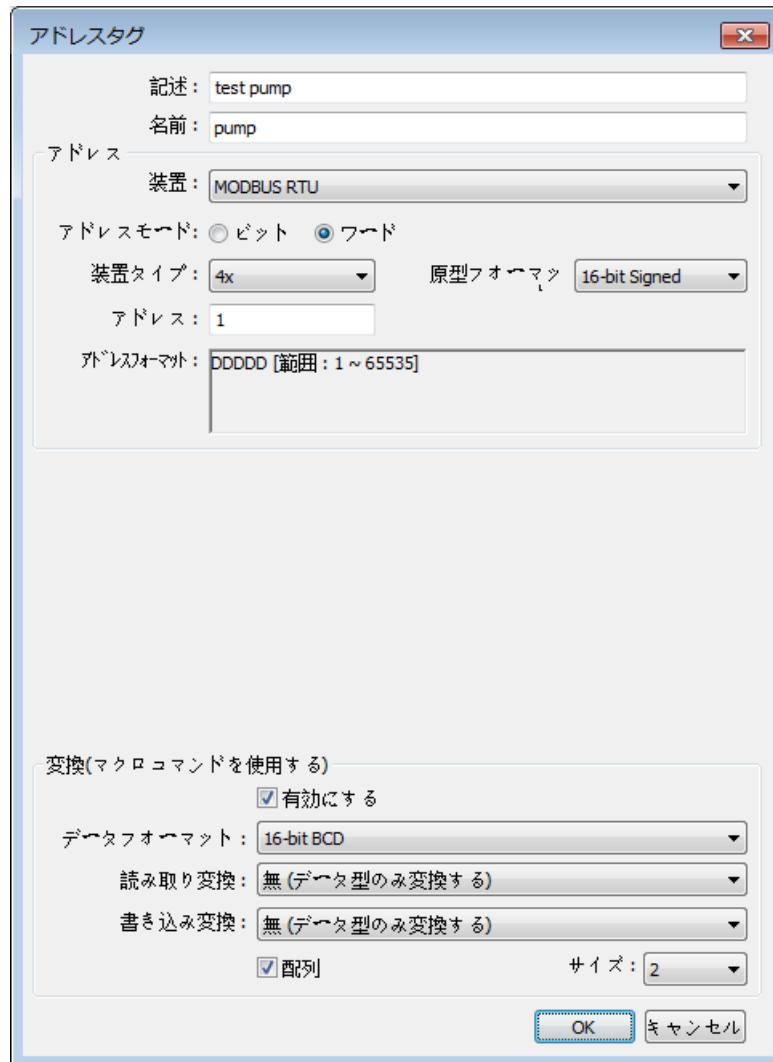
ツールバー上の[ライブラリ]»[アドレスタグライブラリ]をクリックすると、下図のように[アドレスラベルライブラリ]ダイアログボックスに入ります。



設定	記述
ユーザー定義タグ	ユーザー定義のタグを表示します。
システムレジスタ	システムレジスタを表示します。システムレジスタは削除/変更されません。
分類する	本項にチェックマークを入れると、システムレジスタを機能ごとに分類でき、また system_tag.xml ファイルを編集して分類する方法を編集することもできます。 system_tag.xml ファイルは EasyBuilder Pro インストールインデックスにあります。
検索	キーワードを入力して特定したタグを検索します。
新規作成	アドレスタグを新規作成する方法です。次の説明をご

	覧ください。
設定	選択しているアドレスタグの内容を変更します。
全てを削除	全てのアドレスタグを削除します。
削除	選択したアドレスタグを削除します。
CSV にエクスポート	アドレスタグライブラリ内の全ての内容を.csv フォーマットにエクスポートしセーブします。
CSV をインポート	既存していて、それに.csv フォーマットであるアドレスタグを現在のプロジェクトにインポートします。
EXCEL にエクスポート	.xls フォーマットでアドレスタグライブラリでの全ての内容をエクスポートしてセーブします。
EXCEL をインポート	既存していて、それに.xls フォーマットであるアドレスタグを現在のプロジェクトにインポートします。
UTF-8 フォーマットを使用し、CSV ファイルをエクスポートする	本項目にチェックマークを入れると、エクスポートする CSV ファイルは UTF-8 フォーマットになり、そうしないと ANSI フォーマットになります。

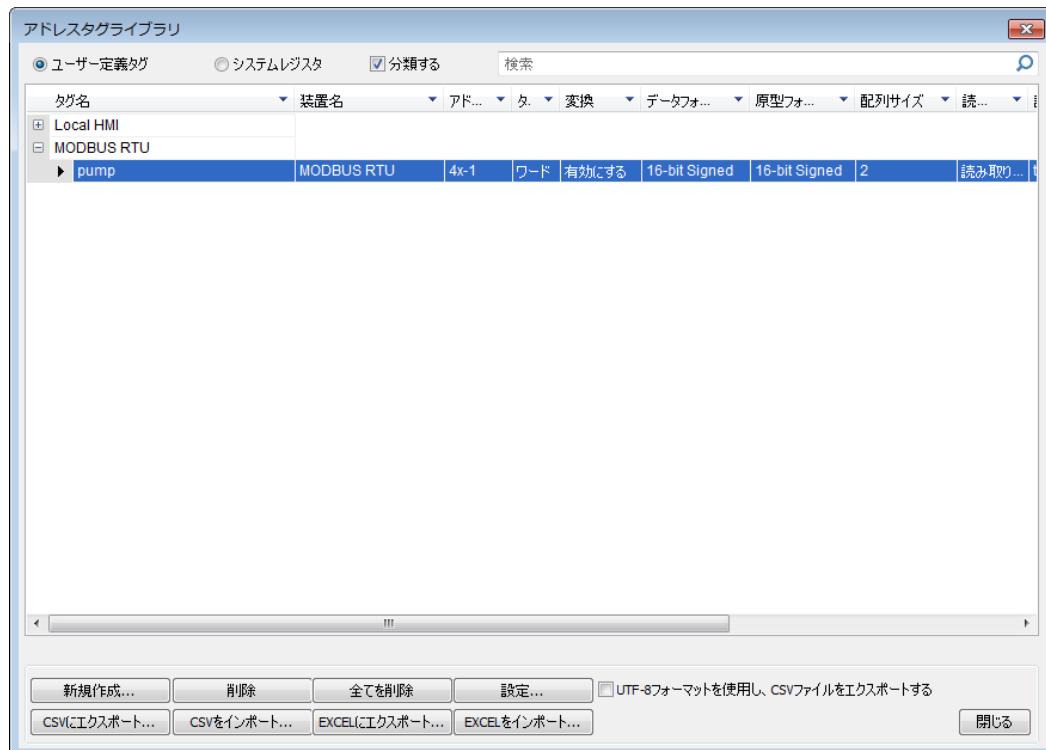
1. [新規作成]を押せば、関連属性を設定することができます。



設定	記述
記述	各タグに可読情報を入力します。
名前	タグの名前を設定します。
装置	設定できる内容は[システムパラメータ設定] » [装置リスト]タブの設定から出てきます。
アドレスモード	[ビット]または[ワード]タイプを選択します。
装置タイプ	選択できるタイプは[装置]、[アドレスモード]の設定によります。
アドレス	タグのアドレスを設定します。
アドレスフォーマット	アドレスタイプに[ワード]を選択した際に、予めデータフォーマットを定義することができます。
変換 (マクロサブルーチンを使用する)	本項を使用すると、変換したいデータフォーマットを選択することができ、またはマクロサブルーチンでデータの読み取り/書き込み変換機能を実行することができます。

読み取り/書き込み 変換	読み取り/書き込みする時に使用するマクロコマンドサブルーチンを選択します。データフォーマットがマクロサブルーチンのタイプと一致する場合のみ、対応するマクロサブルーチンを選択することができます。
-----------------	--

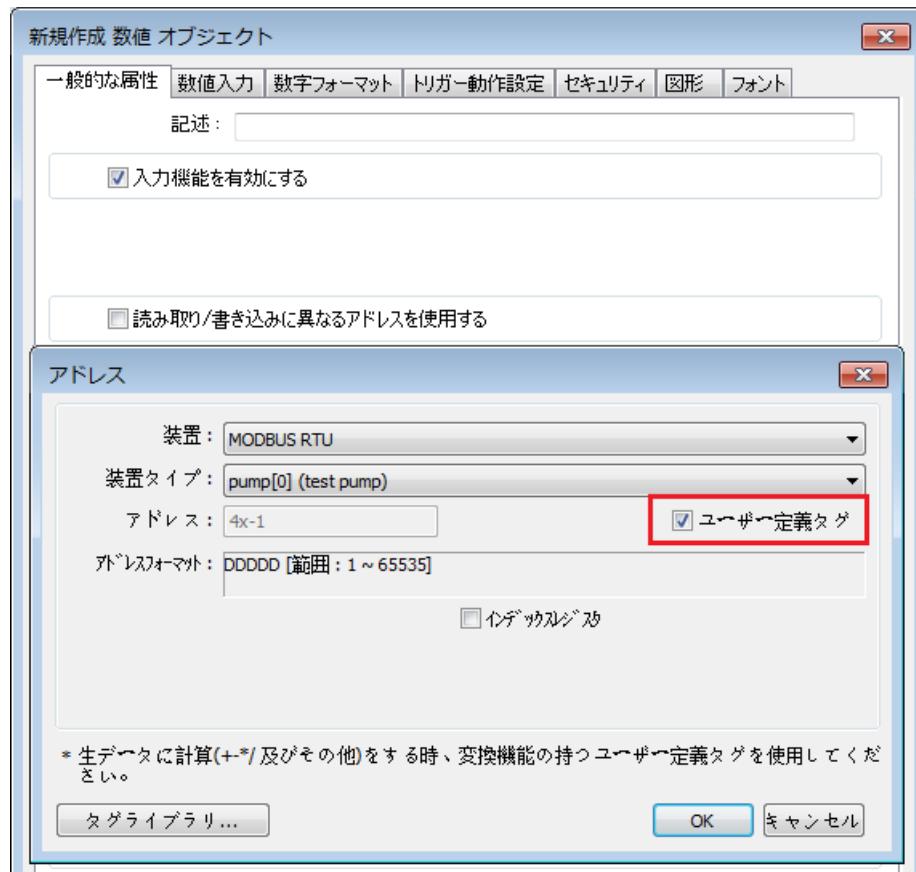
2. [OK]を押した後、[ユーザー定義タグ]で一個の新規追加されたアドレスタグが見られます。



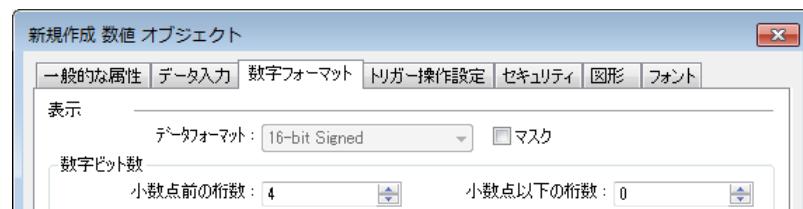
16.3. アドレスタグライブラリの使用

1. アドレスタグライブラリの関連定義を設定します。
2. オブジェクトを新規作成し、アドレス属性に関連する[装置]を選択します。
3. [設定]をクリックすると、アドレス設定ウインドウがポップアップアップされます。
4. [ユーザー定義タグ]にチェックマークを入れます。

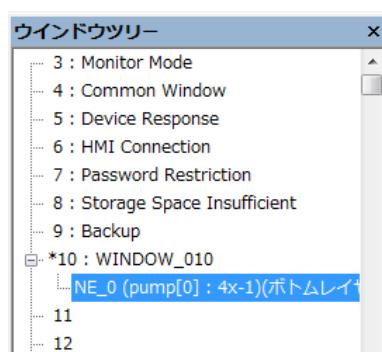
アドレスタグライブラリの作成及び使用



5. [アドレスタイプ]でユーザー定義タグを選択します。
6. 定義した数値フォーマットには予めデータフォーマットが指定された場合、システムは自動的に指定したフォーマットのみ使用するように制限します。



7. 設定完了後、オブジェクト情報ツリーに使用するアドレスタグが見られます。

**Note**

- 使用されたアドレスタグのタグ名は、アドレスタグライブラリでは赤に表記されます。

アドレスタグライブラリの作成及び使用



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

17. レシピデータの転送

本章では、レシピデータを転送する方法について説明します。

17.1. 概要	17-2
17.2. イーサネットまたは USB ケーブルでレシピデータを更新する	17-2
17.3. SD カードまたは USB ディスクでレシピデータを更新する	17-3
17.4. レシピデータの転送	17-5
17.5. レシピデータの保存	17-5

17.1. 概要

レシピデータとは、**RW** と **RW_A** アドレスに存在しているデータのことを指します。これらのアドレスを書き込む・読み取る方法は一般的のワードアドレスレジスタでのデータを書き込む・読み取る方法と変わりません。レシピデータの特性は、シャットダウンされた後、これらのデータは HMI のフラッシュメモリーに保存され、HMI をリセットしたら、**RW** と **RW_A** アドレス内のデータは前回の内容を保つことです。

RW アドレスが保存できるレシピの大きさは **512K words** で、**RW_A** アドレスは **64K words** です。ユーザーは SD カード、USB ディスク、USB ケーブルまたはイーサネットで装置内のレシピデータを更新し、それにレシピデータを PC にアップロードすることもできます。また、装置内のデータをレシピデータに保存することができます。以下はレシピデータの各種操作について説明します。

このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

17.2. イーサネットまたは USB ケーブルでレシピデータを更新する

1. Utility ManagerEx を通じて [転送] -> [ダウンロード]機能を選択します。
2. [RW] と [RW_A] にチェックマークを入れた後、ダウンロードするデータソースを選択します。
3. ダウンロードに成功した後、HMI を再起動したら、**RW** と **RW_A** の内容を更新することができます。

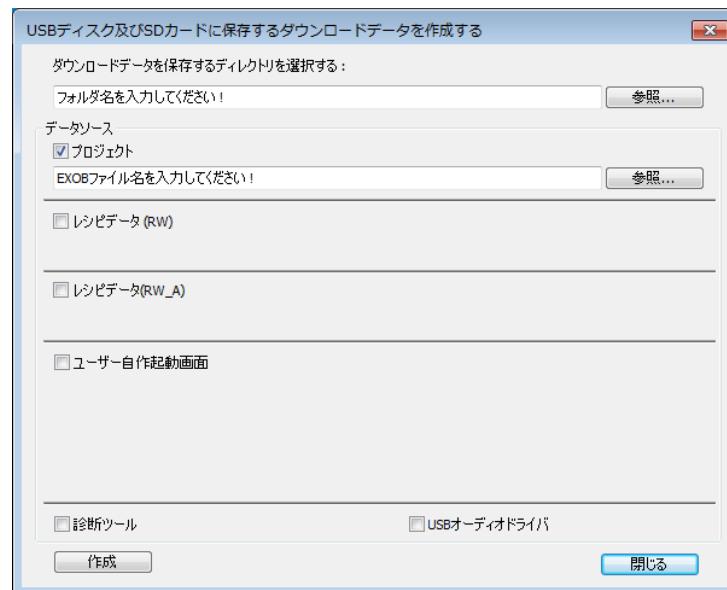
[ダウンロード完了後、プログラム画面を起動]にチェックマークを入れたら、手動で HMI を再起動する手間を省くことができます。

[レシピデータを削除]にチェックマークを入れたら、一切のダウンロード動作を実行する前、システムは先に [RW] と [RW_A] 内のデータ内容を全部削除します。



17.3. SD カードまたは USB ディスクでレシピデータを更新する

1. UtilityManagerEx で[SD カード/USB ディスクに使用するダウンロードデータを作成する]をクリックします。
2. SD カードまたは USB ディスクを PC に接続します。
3. [参照]をクリックし、データの保存先を指定します。
4. [作成]をクリックし、EasyBuilder Pro は自動的にデータを SD カードまたは USB ディスクにセーブします。

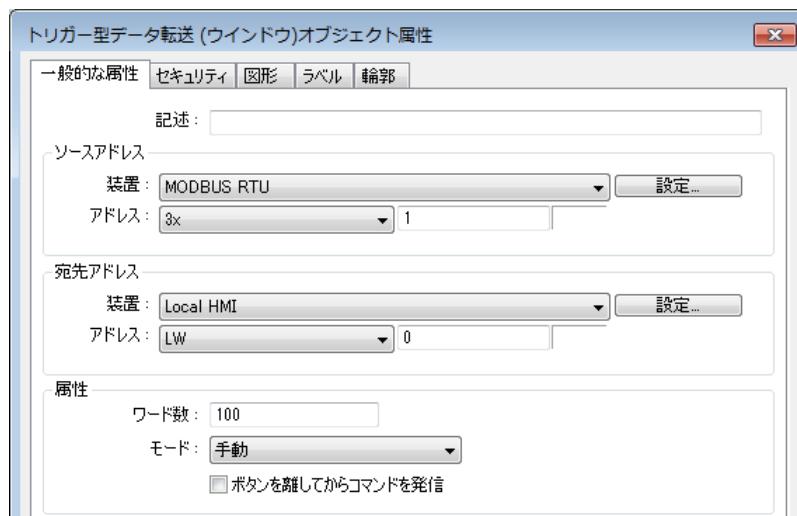


Note

- 作成完了後、2個の新しいフォルダ：history と emt3000 が見られます。emt3000 はプロジェクトを保存するフォルダで、history はレシピデータ及びデータサンプリング/イベントログを保存するフォルダです。
- レシピデータを更新するには、手動で HMI を再起動してください。

17.4. レシピデータの転送

[トリガー型データ転送]オブジェクトを利用し、レシピデータを特定したアドレスに送信することができ、また特定したアドレス内のデータを[RW]と[RW_A]に保存することもできます。



設定	記述
ソースアドレス	ソースアドレスを設定します。
宛先アドレス	宛先アドレスを設定します。
属性	ソースアドレスから宛先アドレスに送信するデータのワード数を設定します。

17.5. レシピデータの保存メカニズム

HMI のフラッシュメモリーの使用寿命を延長するため、システムは 1 分間ごとの間隔でレシピデータを HMI に保存します。それにレシピデータが 2 回の保存動作の間に、シャットダウンによるデータが紛失するのを避けるため、EasyBuilder Pro はシステムレジスタ [LB-9029:強制にレシピデータを HMI に保存する] 機能を提供し、ON に設定すれば、システムはレシピデータの保存動作を一回実行します。また、もし [LB-9028:レシピデータをリセットする] を ON に設定すれば、全てのレシピデータを消去します。

18. マクロの説明

本章では、マクロのシンタックス、編集、及び使用の方法について説明します。

18.1.	概要	18-2
18.2.	マクロエディターの使用説明	18-2
18.3.	マクロの構成	18-8
18.4.	マクロのシンタックス	18-9
18.5.	ステートメント	18-15
18.6.	サブ関数	18-21
18.7.	組み込み関数機能	18-23
18.8.	マクロの作成、及び実行の方法について	18-94
18.9.	ユーザ一定義関数機能	18-100
18.10.	マクロを使用する際の注意点	18-115
18.11.	フリー通信プロトコルで装置をコントロールする	18-116
18.12.	コンパイルエラーメッセージ	18-122
18.13.	マクロコマンドのサンプルプロジェクト	18-130
18.14.	マクロ TRACE 関数	18-135
18.15.	ストリング操作関数の使用方法	18-139
18.16.	マクロパスワード保護	18-147
18.17.	CANbus アドレスの読み取り/書き込みに変数を使用	18-149

18.1. 概要

マクロはアプリケーションソフト以外の他の付加機能を提供しています。HMI が運転している時、マクロは自動的にこれらのコマンドを実行することができます。マクロは複雑な運算、ストリング処理、ユーザーとプロジェクトとの交流などの機能を実行することができます。本章は主にマクロのシンタックスの紹介、使用及び編集する方法について説明します。本章に通じ、迅速に EasyBuilder Pro ソフトウェアが提供する強大なマクロ機能を把握していただければ幸いです。

18.2. マクロエディターの使用説明

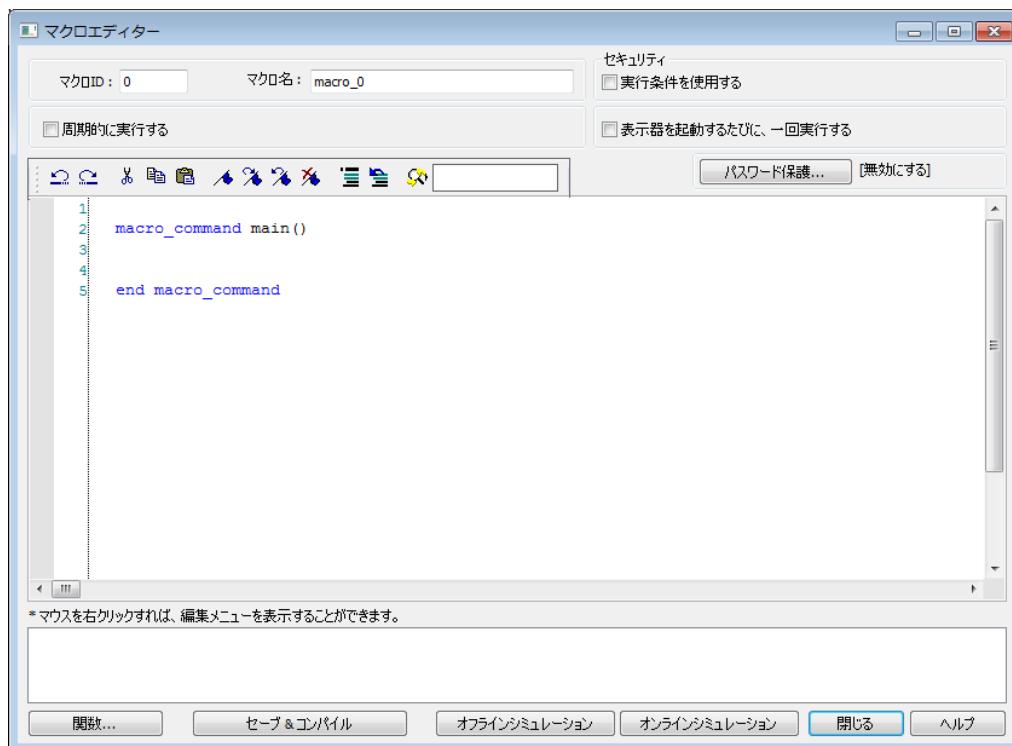
マクロエディターは下記の新機能を提供します：

- 行番号を表示する
- 元に戻す(Undo)/やり直し(Redo)
- 切り取り(Cut)/コピー(Copy)/貼り付け(Paste)
- 全てを選択(Select All)
- ブックマークを作成/キャンセル(Toggle Bookmark)/前のブックマーク(Previous Bookmark)/次のブックマーク(Next Bookmark)/全てのブックマークを削除(Clear All Bookmarks)
- アウトライン表示
- セキュリティ->実行条件を使用する
- 定期実行
- HMI を起動するたびに、一回実行する

以下は各機能を使用する方法について説明します。

1. マクロエディターをオープンしたら、編集エリアの左側に自動的に行番号が表示されているのが見えます。

マクロの説明



2. 編集エリアの中にマウス右クリックすると、下図のように右ボタンメニューが呼び出されます。カレント状態では使用できない機能が灰色に表示されます。例えば、コピー機能を使用するには編集エリアに文字を選択する必要があります、だから何も選択していない場合、コピー機能が使用できません。メニューに示す通り、キーボードショートカットも提供しています。

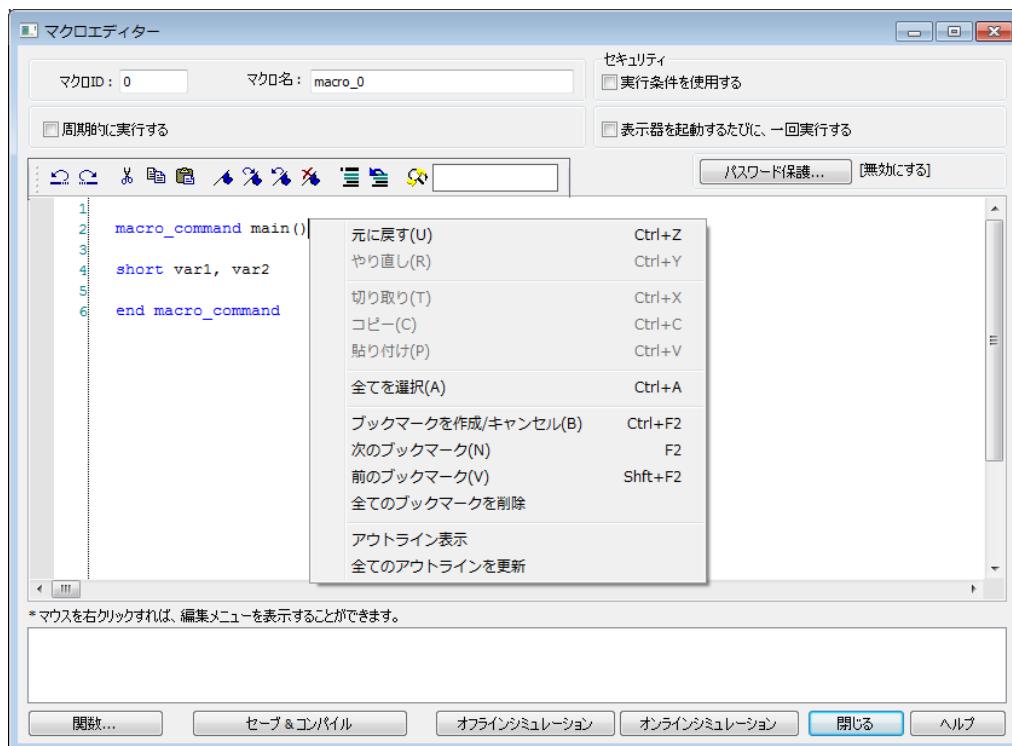


3. 編集エリアの上方にはツールバーがあり、[元に戻す]、[やり直し]、[切り取り]、[コピー]、[貼り付け]、[ブックマークを作成/キャンセル]、[次のブックマーク]、[前のブックマーク]、[全てのブックマークを削除]などのボタンを提供します。

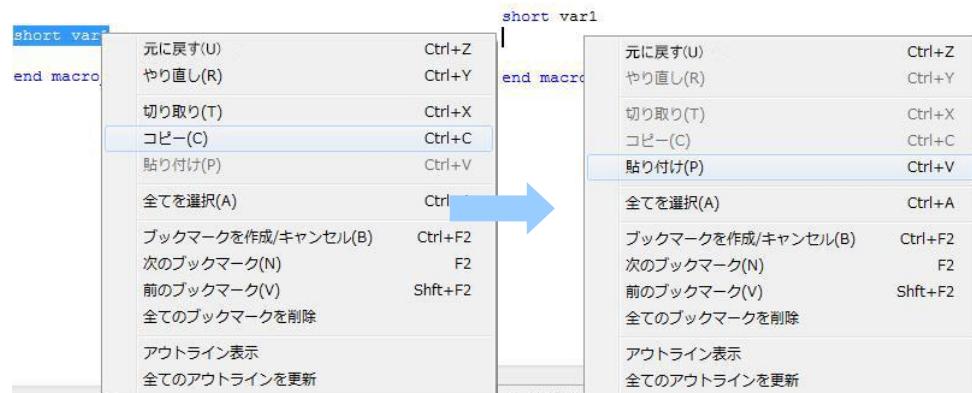


4. 編集エリアの内容を変更すると、[元に戻す]機能がオープンされ、ユーザーが元に戻すを実行後、[やり直し]機能が使用可能になります。ユーザーは右クリックまたはキーボードショートカット(Undo: Ctrl+Z、Redo: Ctrl+Y)で本機能を実行することができます。

マクロの説明



5. 編集エリアで文字を選択したら、[切り取り]、[コピー]を実行した後、[貼り付け]で選択した文字を貼り付けることができます。



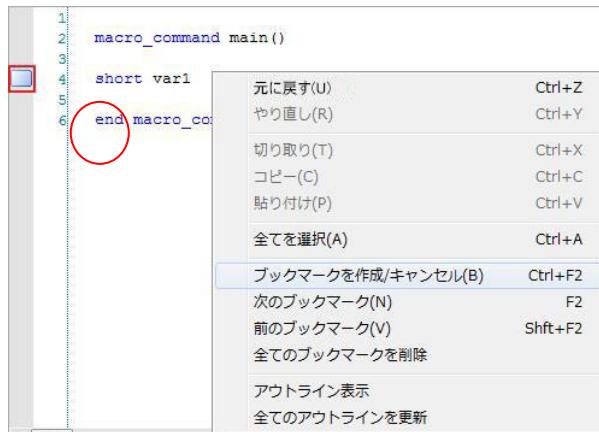
6. [全てを選択]を選択すると、編集エリアにある全部の内容を選択することができます。



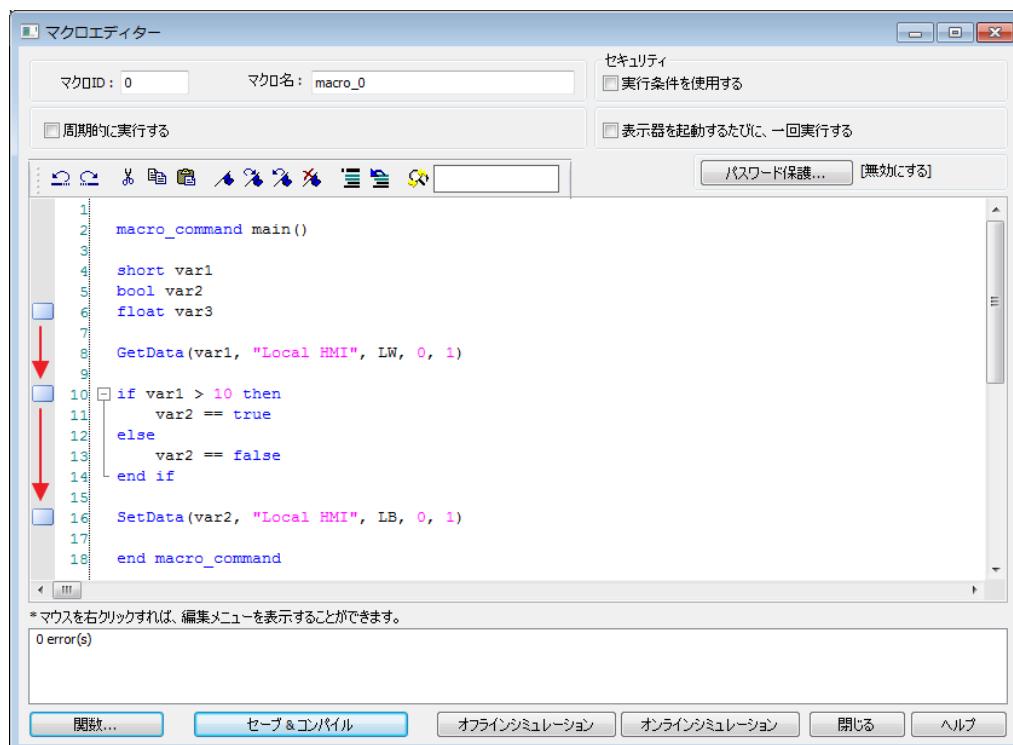
7. マクロが長い場合、ユーザーの読みやすさを向上するため、ブックマークの機能を提供し

ています。下記ではどのように本機能を使用できるのかを説明します。

- カーソルを編集エリアにブックマークを入れたい位置に移動し、右クリックして[ブックマークを作成/キャンセル]を選択します。編集エリアの左側ではブックマークを代表している青の小さいブロックが見られます。



- カーソルの所在位置には既にブックマークが存在している場合、[ブックマークを作成/キャンセル]を選択し、ブックマークを閉じます。またはブックマークをオープンします。
- 右クリックして[次のブックマーク]を選択したら、カーソルは一個次のブックマークに移動します。[前のブックマーク]を選択したら、カーソルは一個前のブックマークに移動します。

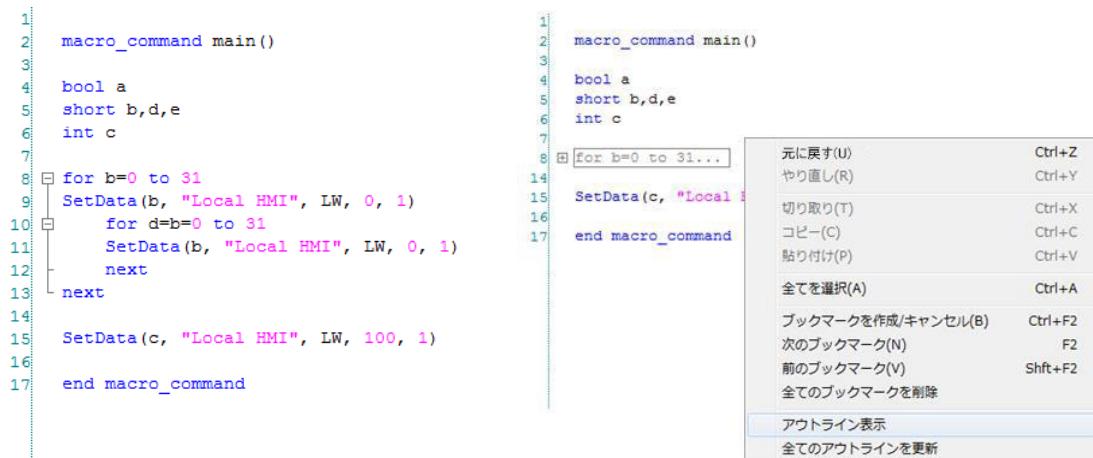


- [全てのブックマークを削除]を選択したら、全てのブックマークが閉じられます。
- マクロエディターはアウトライン表示機能を提供していて、ユーザーにより観察しやすいようにさせます。アウトライン表示というのは、エディターは同じブロックにあるコードを隠し、隠されたコードは田 [...] に表示されます。編集エリアの左側にはツリーフォルダアイコンが表示され、ユーザーは田 [...] を押してブロック程式区塊を隠し、田 [...] を押してブロックを

オープンすることができます。下図をご参考ください：



9. 右クリックで[アウトライン表示]を選択すると、隠しているマクロコードブロックをオープンすることができます。



10. 時には、コードのアウトラインが不正確になったことがあります。これはエディターが現在入力したキーワードが注釈に存在しているかどうかのが判別できないからです。例として下図をご参照ください。ユーザーは右クリックで[全てのアウトラインを更新]を選択し。この錯誤を訂正することができます。



11. 特定なキーワードに囲まれたマクロコードは“ブロック”と呼びます。既定のブロックは下記のようにあります：

- サブルーチン：sub-endsub
- 反復ステートメント：
 - i. for-next
 - ii. while-wend
- 論理ステートメント：
 - i. if-endif
- 選択ステートメント：select case-end select

12. マクロエディターは独占ウインドウではありません。マクロエディターをオープンした後、メイン画面に戻ってもいいし、直接にマクロエディターウインドウの中でオンライン/オフラインシミュレーションを実行してもいいです。



13. マクロエディターは文字を検索/置換する機能を提供します。

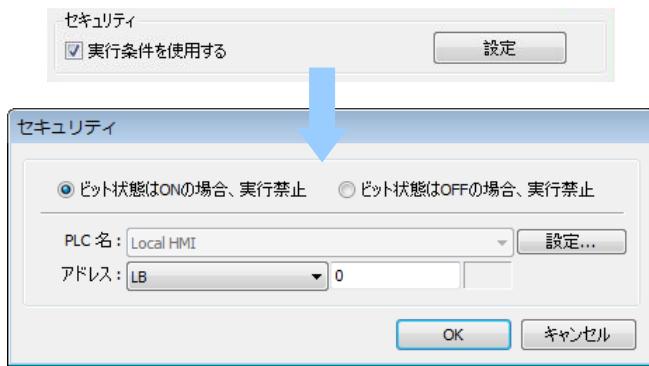


14. ユーザーが[定期実行]にチェックマークを入れると、このマクロは定期的にトリガーされます。



15. ユーザーが[セキュリティ]->[実行条件を使用する]->[設定]をクリックすると、セキュリティの設定に入ります：

- ビット状態が ON になったら、実行不可：ビット状態が ON になる時、マクロの実行を禁止します。
- ビット状態が OFF になったら、実行不可：ビット状態が OFF になる時、マクロの実行を禁止します。



16. ユーザーが[HMI を起動するたびに、一回実行する]を選択した場合、HMI を起動する時に、マクロが自動的に一回実行されます。

18.3. マクロの構成

マクロは各種のステートメントで構成されたものです。これらのステートメントには定数、変数及び各種の演算子が含まれています。これらのステートメントを特定した順・位置に置くことで要望する結果が得られます。

一般的に、コマンドは下記のようなフォーマットで構成されます：

グローバル変数宣言-----	オプション
Sub Function Block Declarations (サブルーチン宣言) -----	オプション
局所変数宣言	
End Sub (終了サブルーチン)	
macro_command main() [メイン関数] -----	必須
局所変数宣言	
[各式ステートメント]	
end macro_command [終了メイン関数] -----	必須

一個のマクロには一個で唯一のメイン関数を有する必要があり、マクロコマンドの実行を開始することに用いられます。フォーマットは：

macro_command 関数名()

endmacro_command

変数宣言はマクロステートメントの前に置く必要があり、もしステートメントが変数宣言の前に置かれたら、マクロのコンパイルが完成できなくなります。

局所変数は一般的に、マクロのメイン関数またはユーザー定義のサブルーチンに使用していま

す。その合法性は指定した関数の中にのみ有効です。

グローバル変数は一般的に、全てのマクロ関数の前に定義され、それにその合法性は全関数にも有効です。ただし、局所変数とグローバル変数が同じの名前に定義された場合、局所変数のみ有効になります。

以下は一個シンプルなマクロコマンドで、中には変数宣言と関数呼び出しが含まれています。ダブルスラッシュ"//"はプログラムの注釈を代表していて、その後ろの文字は実行されません。

```
macro_command main()
    short pressure = 10                      // 局所変数宣言
    SetData(pressure, "Allen-Bradley DF1", N7, 0, 1) // 関数呼び出し
end macro_command
```

18.4. マクロのシンタックス

18.4.1. 定数と変数

18.4.1.1. 定数

定数は各形式のステートメントに直接に使用できる固定数値です。下記フォーマットがあります：

定数の型	使用説明	例
十進数整数		345, -234, 0, 23456
十六進数	0x を頭文字にする必要があります。	0x3b, 0xffff, 0x237
ASCII	キャラクタは引用符に囲まれる必要があり、ストリングはダブル引用符に囲まれる必要があります。ダブル引用符を表示させたい場合、¥"で表示してください。	'a', "data", "関数"
ブーリアン型		true, false

以下は一個簡単な定数の使用例です。

```
macro_command main()
    short A, B // 宣言AとBは変数
    A = 1234
    B = 0x12 // 1234と0x12はショート整数型定数
end macro_command
```

18.4.1.2. 変数

変数は各種の情報を代表する名前です。マクロでは、これらの情報はマクロコマンドステート

メントの実行結果により変更することができます。

変数の命名規則

- アルファベットを頭文字にする必要があります。
- 変数名前の長さは 32 キャラクタに超えないこと。
- システム予約語は変数名に使用できません。

以下は 8 種類の異なる変数型で、前の 5 種類は符号付き数値型で、後の 3 種類は符号なし数値型です：

変数の型	記述	範囲
bool ブーリアン型	1 bit(1 ビット)	0, 1
Char キャラクタ型	8 bits(1 バイト)	+127~-128
short ショート整数型	16 bits(1 ワード)	+32767~-32768
int 整数型	32 bits(2 ワード)	+2147483647~-2147483648
float 浮動小数点数型	32 bits(2 ワード)	
Unsigned char 符号なし キャラクタ型	8 bits(1 バイト)	0~255
Unsigned short 符号な しショート整数型	16 bits(1 ワード)	0~65535
Unsigned int 符号なし 整数型	32 bits(2 ワード)	0~4, 294, 967, 295
long 長整数型	64 bits(4 ワード) (cMT にのみ対応)	+281474976710655 ~ -281474976710655
unsigned long 長整数型	64 bits(4 ワード) (cMT にのみ対応)	0 ~ 281474976710655
double ダブル精度浮動 小数点数型	64 bits(4 ワード) (cMT にのみ対応)	

変数の宣言

変数を使用する前、予め宣言する必要があります。だから、マクロでの全ての変数はステートメントを使用する前に宣言し終える必要があります。変数を宣言する時に、まずは変数の型を定義し、後ろに定数名を付きます。

例:

```
int      a
short    b, switch
float    pressure
unsigned shortc
```

配列の宣言

マクロは一次元配列をサポートします(0 からです)。配列を宣言する時に、まずは変数の型、変数名、そして当配列にある変数の数を設定し、数量は“[]”の中に入れる必要があります。配列変数の長さは 1~4096 までです。一個のマクロには最大 4096 個の変数をサポートします。

例:

```
int      a[10]
short   b[20], switch[30]
float    pressure[15]
```

配列の開始最小値は 0 で、最大値は(配列の長さ-1)

例:

```
Char data[100]           // 配列の長さは 100
だから: 最小の配列は“data[0]”、最大の配列は“data[99]”、即ち 100-1=99。
```

変数と配列の初期化

2 種類の方法で変数を初期化することができます :

- ステートメントの代入演算子を使用する

例:

```
int a
float b[3]
a = 10
b[0] = 1
```

- 変数を宣言する時に直接に数値を与える

```
char a = '5', b = 9
```

配列変数の宣言は一種の特殊の状況です。一個の完全な配列が初期化された場合、配列を宣言する時に、数値を波括弧“{}”に入れ、各数値をコンマを使って分けることができます。下記のように示されます:

```
float data[4] = {11, 22, 33, 44} // そうなったら data[0] = 11, data[1] = 22....
```

18.4.2. 演算子

演算子は通常、データがどのように操作・演算するのかを指定することに用いられます。下記のように示されます:(任意のステートメントの中においても、演算子左側の結果は全部演算子右側の条件によって得られます。)

演算子	記述	例
-----	----	---

=	代入演算子	pressure=10
数学演算子	記述	例
+	加	A=B+C
-	減	A=B-C
*	乗	A=B*C
/	除	A=B/C
%	余りを求める(剰余に戻る)	A=B%5

注意：

整数のデフォルト変数型なので、除法をする時、除数と非除数両方とも整数であり、それに計算した結果に両方とも小数点が含まれた場合、小数点以下は自動的に切り捨てられます。小数点以下の切捨てを避けたかったら、除数、或いは被除数に.0 を加えたら、数値の変数型を浮動小数点型に変換できます。

例：

A = 3 / 2 = 1 » 3 と 2 両方とも整数型だったので、演算結果も整数型になります。

B = 3 / 2.0 = 1.5 » 3 は整数型、2.0 は浮動小数点型だったので、演算結果が浮動小数点型になります。

C = 3.0 / 2 = 1.5 » 3.0 は浮動小数点型、2 は整数型だったので、演算結果が浮動小数点型になります。

比較演算子	記述	例
<	小なり	If A<10 then B=5
<=	小なりイコール	If A<=10 then B=5
>	大なり	If A>10 then B=5
>=	大なりイコール	If A>=10 then B=5
==	イコール	If A==10 then B=5
<>	ノンイコール	If A<>10 then B=5

論理演算子	記述	例
And	論理積	If A<10 and B>5 then C=10
Or	論理和	If A>=10 or B>5 then C=10
Xor	排他的論理和	If A xor 256 then B=5
Not	否定	If not A then B=5

シフト演算子及びビット演算子は通常、キャラクタ型変数、ショート整数型変数及び整数型変数のビットを操作することに用いられます。一個のステートメントの中に、これら演算子の優先順位はステートメントの左から右です。即ち、ステートメントの中に、左側にある演算子が先に実行され、左側から右側へ実行されます。

シフト演算子	記述	例
<<	各ビットを指定した数だけ左 ヘシフトする	A=B<<8
>>	各ビットを指定した数だけ右 ヘシフトする	A=B>>8

ビット演算子	記述	例
&	ビット AND	A=B&0xf
	ビット OR	A=B C
^	ビット XOR	A=B^C
~	ビット NOT	A=~B

全ての演算子の優先順位

上記全ての演算子の優先順位を上位から下位まで下記のように述べます：

1. 括弧内の演算子を最優先
2. 数学演算子
3. シフト・ビット演算子
4. 比較演算子
5. 論理演算子
6. 代入演算子

キーワード(予約語)

下記のキーワードはシステムの予約語です。これらの文字は変数名、配列名、或いは関数名に使われません。

+, -, *, /, %, >=, >, <=, <, <>, ==, and, or, xor, not, <<, >>, =, &, |, ^, ~
exit, macro_command, for, to, down, step, next, return, bool, short, int, char, float, void, if, then, else,
break, continue, set, sub, end, while, wend, true, false

SQRT, CUBERT, LOG, LOG10, SIN, COS, TAN, COT, SEC, CSC, ASIN, ACOS, ATAN, BIN2BCD, BCD2BIN,
DATE2ASCII, DATE2DEC, DEC2ASCII, FLOAT2ASCII, HEX2ASCII, ASCII2DEC, ASCII2FLOAT, ASCII2HEX, FILL,
RAND, DELAY, SWAPB, SWAPW, LOBYTE, HIBYTE, LOWORD, HIWORD, GETBIT, SETBITON, SETBITOFF,
INVBIT, ADDSUM, XORSUM, CRC, CRC8, INPORT, OUTPORT, POW, GetCnvTagArrayIndex , GetError,
GetData, GetDataEx, SetData, SetDataEx, SetRTS, GetCTS, Beep, SYNC_TRIG_MACRO,
ASYNC_TRIG_MACRO, TRACE, FindDataSamplingDate, FindDataSamplingIndex, FindEventLogDate,
FindEventLogIndex
StringGet, StringGetEx, StringSet, StringSetEx, StringCopy, StringMid, StringMD5, StringDecAsc2Bin,
StringBin2DecAsc, StringDecAsc2Float, StringFloat2DecAsc, StringHexAsc2Bin, StringBin2HexAsc,
StringLength, StringCat, StringCompare, StringCompareNoCase, StringFind, StringReverseFind,
StringFindOneOf, StringIncluding, StringExcluding, StringToUpper, StringToLower, StringToReverse,
StringTrimLeft, StringTrimRight, StringInsert, String2Unicode, Unicode2Utf8, UnicodeCat,
UnicodeCompare, UnicodeCopy, UnicodeExcluding, Utf82Unicode

18.5. ステートメント

18.5.1. 定義ステートメント

この定義ステートメントには、変数及び配列の宣言が含まれています。正式なフォーマットは以下のとおりです：

型	名
---	---

一個の変数名を"名"で、型を"型"に定義します。

例:

Int A // 変数 A を整数型に定義した

型	配列名[配列の長さ]
---	------------

一個の配列名を"名"、大きさを"配列の長さ"、それに型を"型"に定義します。

例:

Int B[10] // 一次元配列 B の長さを 10、型を整数型に定義した

18.5.2. 代入ステートメント

代入ステートメントは代入演算子を使用し、代入演算子の右側にある式の演算結果を演算子の左側にある変数の中に入れます。一個の式は変数、定数及び各種の演算子で構成され、実行後、新しいデータが生じます。

型	式
---	---

例:

A=2 // 変数 A が 2 に代入した

18.5.3. 論理演算ステートメント

論理演算ステートメントは論理(ブーリアン)数式に基づいて対応する動作を実行するのです。そのステートメントは以下のとおりです：

單一行フォーマット

If <Condition> then [Statements] else [Statements] end if

例:

```
ifa==2then
    b=1
else
    b=2
endif
```

ブロックフォーマット

```
If <Condition> then
    [Statements]
else if <Condition-n> then
    [Statements]
else
    [Statements]
end if
```

例:

```
ifa==2then
    b=1
elseif a==3then
    b=2
else
    b=3
endif
```

シンタックス記述

if	ステートメントの先頭部分に置かれる必要があります。
<Condition>	必要条件です。これはコントロールステートメントです。<Condition>は 0 の場合、“FALES”(条件は偽)になります。<Condition>は 0 でない場合、“True”(条件は真)になります。
then	<Condition>の実行は“TRUE”(真)の場合、実行するステートメントの前に置く必要があります。
[Statements]	ブロックの中には選択できるパラメータですが、單一行フォーマットの中、それに else がない場合、必要パラメータになります。 <Condition>は真の場合に実行されます。
else if	オプションです。一つまたは複数のステートメントで、対応の<Condition-n>が true の場合に実行されます。
<Condition-n>	オプションです。解釈は Condition と同じです。

else	オプションです。上記 Condition と Condition-n が全部 true でない場合に実行されます。
end if	必須です。if-then ステートメントの中で、これを使って if-then を終了します。

18.5.4. 選択ステートメント

Select-case は多重判断の記述を処理でき、その機能は if-else ステートメントに類似しています。指定した変数の数値に基づき、数値に合っている case に対応し、case の下の記述を実行します。実行して break 記述に会うまで、endselect にスキップします。シンタックスの構成は以下のとおりです：

デフォルト case がないフォーマット：

```
Select Case [variable]
Case [value]
[Statements]
break
end Select
```

例：

```
Select Case A
Case 1
    b=1
    break
end Select
```

デフォルト case があるフォーマット：

```
Select Case [variable]
Case [value]
[Statements]
break
Case else
[Statements]
break
end Select
```

例：

```
Select Case A
Case 1
    b=1
```

```

break
Case else
    b=0
    break
end Select

```

複数の異なる **case** が同一ブロックに対応する場合 :

```

Select Case [variable]
Case [value1]
    [Statements]
Case [value2]
    [Statements]
    break

end Select

```

例:

```

Select Case A
    Case 1
        break
    Case 2
        b=2
        break
    Case 3
        b=3
        break
end Select

```

シンタックス記述

Select Case	ステートメントの先頭部分に置かれる必要があります。
[variable]	必要条件です。この変数は各 case と比較します。
Case else	オプションです。デフォルト case を代表しています。 [variable] の数値がどの case にも合っていない場合、本記述の下にあるブロックが実行されます。デフォルト case がない状況で、 [variable] の数値がどの case にも合っていない場合、何の動作も実行されなく、直接に終了して select コントロール構成の結末へスキップします。
break	オプションです。ある case の下にスキップして実行する時に、 case ステートメントの下にある記述が break コマンドに会うまで逐一で実行されます。 break コマンドに会ったら、実行が終了し endselect にスキップします。 case 記述の下には break コマンドがない場合、 end

`select` 記述に会うまで引き続き実行されます。`end select` に会ってから、終了して `select` コントロール構成の結末へスキップします。

end Select `select-case` ステートメントの終了記号です。

18.5.5. 反復ステートメント

反復ステートメントは、反復条件に基づいて反復的に任務を実行します。反復ステートメントは2種類あります。

18.5.5.1. `for next` ステートメント

`For-next` ステートメントは通常、回数が固定した反復任務の実行に用いられます。一個の変数は任務の実行回数のカウンターと、反復任務を終了する条件として用いられます。この変数は固定で実行する回数です。シンタックスは以下のとおりです：

```
for [Conunter] = <StartValue> to <EndValue> [step <StepValue>]  
[Statements]  
next [Counter]
```

または

```
for [Conunter] = <StartValue> down <EndValue> [step <StepValue>]  
[Statements]  
next [Counter]
```

例:

```
for a=0 to 10 step 2  
    b=a  
    next a
```

シンタックス記述

for	ステートメントの先頭部分に置かれる必要があります。
[Counter]	必須です。反復カウンターの数値変数であります。当変数の結果で反復の回数を計算します。
<Start Value>	必須です。Counter の初期数値です。
to/down	必須です。<step>で<Counter>の増加/減少を決めます。 “to” : <StepValue>で<Counter>を増加します。 “down” : <StepValue>で<Counter>を減少します。
<End Value>	必須です。Counter の終了数値、テスト点です。<Counter>が当数値より大きい場合、マクロはこの反復任務を終了します。
step	オプションです。<Step Value>を1以外の数値に指定します。

[Step Value]	オプションです。Counter のステップを決めます。数値のみ使用できます。指定していなかったら、デフォルトを 1 にします。
[Statements]	オプションです。for と next の間のステートメントブロックです。そのブロックは指定した回数を実行します。
next	必須です。
[Counter]	オプションです。

18.5.5.2. while-wend ステートメント

While-wend ステートメントは不確定な回数の反復任務を実行します。一個の変数を設定し、反復の終了条件を判断することに用いられます。条件は“True”的場合、当ステートメントは“False”に変わったまで引き続き反復に実行します。シンタックスは以下のとおりです：

```
while <Condition>
  [Statements]
wend
```

例:

```
while a<10
  a=a+10
wend
```

シンタックス記述

while	ステートメントの先頭部分に置かれる必要があります。
continue	必要条件です。これはコントロールステートメントです。“True”になった場合、反復コマンドの実行を開始します。“False”になった場合、反復コマンドを終了します。
return [value]	“TRUE”と判断した場合、継続に反復コマンドを実行します。
wend	While-wend ステートメントの終了表記です。

18.5.5.3. 他のコントロールコマンド

break	for-next と while-wend ステートメントの中に用いられます。このステートメントに会うと、即時にステートメントの終了部分へスキップされます。
continue	for-next と while-wend ステートメントの中に用いられます。このステートメントに会うと、即時に現在の反復コマンドを終了し、次の反復コマンドを実行します。
return	ユーザー定義の function 戻り値の記述に用いられます。メイン関数の

中に書いた場合、強制的にメイン関数の結果へスキップすることに用いられます。

18.6. サブ関数

サブ関数を利用することで、有効的に反復コマンドのコードを減少することができます。サブ関数は呼び出す前に定義される必要があり、また何の変数とステートメントでも使用できます。メイン関数の中に、サブ関数のパラメータをサブ関数名の後の括弧の中に入れれば、サブ関数を呼び出すことができます。サブ関数が実行された後、実行の結果をメイン関数が所要の代入ステートメントまたは条件に返ります。サブ関数を定義する時、戻り値が必要ではなく、パラメータ部分を空白にしてもいいです。メイン関数の中にサブ関数を使用する際に、使用方式は定義に合うこと。シンタックス構成は以下のとおりです：

戻り値があるサブ関数シンタックス：

```
sub type <関数名> [(parameters)]
    Local variable declarations
    [Statements]
    [return [value]]
end sub
```

例:

```
sub int Add(int x, int y)
    int result
    result = x +y
    return result
end sub
```

```
macro_command main()
    int a = 10, b = 20, sum
    sum = Add(a, b)
end macro_command
```

または:

```
sub int Add()
    int result, x=10, y=20
    result = x +y
    return result
end sub
```

```
macro_command main()
```

```

int sum
sum = Add()
end macro_command

```

戻り値がないサブ関数のシンタックス：

```

sub <関数名> [(parameters)]
  Local variable declarations
  [Statements]
end sub

```

例：

```

sub Add(int x, int y)
  int result
  result = x +y
end sub

```

```

macro_command main()
  int a = 10, b = 20
  Add(a, b)
end macro_command

```

または：

```

sub Add()
  int result, x=10, y=20
  result = x +y
end sub

```

```

macro_command main()
  Add()
end macro_command

```

シンタックス記述

sub	ステートメントの先頭部分に置かれる必要があります。
type	オプションです。サブ関数が実行後、返ったデータの型を定義します。サブ関数は何の数値も返らなくてもいいです。
(parameters)	オプションです。これらのパラメータはメイン関数から送信された数値を確保します。送信されたパラメータはパラメータ変数宣言との型が同じである必要があります。 例 : sub int MyFunction(int x, int y). x と y はメイン関数から送信された整数型デフォーマットのデータである必要があります。このサブ

関数を呼び出すステートメントのフォーマットは大体以下のとおりです：

`ret=MyFunction(456, pressure)`、その中 `pressure` は整数型フォーマットであるこそ、サブ関数パラメータ変数宣言に合致します。

注意すべきなのは、呼び出しステートメントのパラメータ部分は定数、或いは変数のどちらでもいいです。このサブ関数を実行した後、一個の整数型データは変数“`ret`”に返却します。

Local variable declaration	送信されるパラメータ以外、サブ関数の中に使用する変数は事前に宣言する必要があります。上記の例から言えば、 <code>X</code> と <code>Y</code> はサブ関数が使用できる変数になります。グローバル変数もサブ関数の中で用いられます。
[Statements]	実行するステートメントです。
[return[value]]	オプションです。実行結果を呼び出しステートメント返却します。この結果は一個の定数、或いは変数のどちらでもいいです。返却した後、同時にサブ関数の実行を終了します。サブ関数は何の数値を返らなくてもいいが、 <code>type</code> 部分が定義された場合、この <code>return</code> 記述を加える必要があります。
End sub	必須です。サブ関数の終了に用いられます。

18.7. 組み込み関数機能

EasyBuilder Pro は、PLC からデータを取り/データを PLC に送信、情報処理、及び算術用の組み込み関数を提供しています。

18.7.1. 関数一覧表

関数名	概略
PLC 関数	
GetData	PLC のデータを読み取ります。
GetDataEx	PLC のデータを読み取り、PLC の応答を待たずに次へ実行します。
GetError	エラーコードを取得します。
SetData	データを PLC に書き込みます。
SetDataEx	データを PLC に書き込み、PLC の応答を待たずに次へ実行します。
Free Protocol 関数	
GetCTS	RS-232 の CTS 信号を検知します。
IMPORT	COM ポート/Ethernet ポートからデータを読み取ります。
IMPORT2	COM ポート/Ethernet ポートからデータを読み取った後、指定した時間を待ちます。

<u>INPUT3</u>	COM ポート/Ethernet ポートから指定した長さを読み取ります。
<u>OUTPORT</u>	データを COM ポート/Ethernet ポートから PLC またはコントローラに転送します。
<u>PURGE</u>	COM ポートの出力・入力バッファエリアをクリアします。
<u>SetRTS</u>	RS-232 の RTS 信号を上げ・下げます。
マクロ制御関数	
<u>ASYNC TRIG MACRO</u>	実行中のマクロで他の指定したマクロのをトリガーします。
<u>SYNC TRIG MACRO</u>	実行中のマクロで他の指定したマクロをトリガーし、元のマクロの実行が一時停止され、トリガーされたマクロが実行完了してから、後続の実行を続けます。
<u>DELAY</u>	マクロを指定した時間で一時停止させます。
データ操作関数	
<u>FILL</u>	データを配列に置きます。
<u>SWAPB</u>	1 個の 16 ビットのデータの上位バイトと下位バイトを逆さにします。
<u>SWAPW</u>	1 個の 32 ビットのデータの上位バイトと下位バイトを逆さにします。
<u>LOBYTE</u>	1 個の 16 bit データの下位バイトを読み取ります。
<u>HIBYTE</u>	1 個の 16 bit データの上位バイトを読み取ります。
<u>LOWORD</u>	1 個の 32 bit データの下位バイトを読み取ります。
<u>HIWORD</u>	1 個の 32 bit データの上位バイトを読み取ります。
<u>INVBIT</u>	データの指定したビットアドレスの状態を逆さにします。
<u>SETBITON</u>	データの指定したビットアドレスを 1 に設定します。
<u>SETBITOFF</u>	データの指定したビットアドレスを 0 に設定します。
<u>GETBIT</u>	データの指定したビットアドレスの状態を読み取ります。
データ変換関数	
<u>ASCII2DEC</u>	キャラクタ型 ASCII データを十進数フォーマットに変換します。
<u>ASCII2FLOAT</u>	キャラクタ型 ASCII データを浮動小数点数フォーマットに変換します。
<u>ASCII2HEX</u>	キャラクタ型 ASCII データを十六進数フォーマットに変換します。
<u>BIN2BCD</u>	BIN フォーマットのデータを BCD フォーマットに変換します。
<u>BCD2BIN</u>	BCD フォーマットのデータを BIN フォーマットに変換します。
<u>DATE2ASCII</u>	現在の日付データを ASCII フォーマットに変換します。
<u>DATE2DEC</u>	現在の日付データを十進数フォーマットに変換します。
<u>DEC2ASCII</u>	十進数フォーマットのデータを ASCII フォーマットに変換します。

FLOAT2ASCII	浮動小数点数フォーマットのデータを ASCII フォーマットに変換します。
HEX2ASCII	十六進数フォーマットのデータを ASCII フォーマットに変換します。
StringDecAsc2Bin	十進数ストリングを整数に変更します。
StringBin2DecAsc	整数を十進数ストリングに変更します。
StringDecAsc2Float	十進数ストリングを浮動式小数点数に変換します。
StringFloat2DecAsc	浮動小数点数を十進数ストリングに変換します。
StringHexAsc2Bin	十六進数ストリングを整数に変換します。
StringBin2HexAsc	整数を十六進数ストリングに変換します。
ストリング処理関数	
String2Unicode	ストリングを Unicode ストリングに変換します。
StringCat	ソースストリングを目的ストリングの後ろに接ぎます。
StringCompare	両ストリングの内容が等しいのかを比較します。大文字と小文字を同一視しません。
StringCompareNoCase	両ストリングの内容が等しいのかを比較します。大文字と小文字を同一視します。
StringCopy	ストリングをコピーします。
StringExcluding	source ストリングでのあるインデックス値 0 を頭文字(一番目のキャラクタ)にするサブストリングを取り出します。
StringFind	あるストリングの中でもう一つのストリングを探します。
StringFindOneOf	target ストリングの中で任意のキャラクタが source ストリングで始めて現れた位置を探します。
StringGet	PLC のキャラクタ配列データを読み取ります。
StringGetEx	PLC のキャラクタ配列データを読み取り、PLC の応答を待たずに次へ実行します。
StringIncluding	source ストリングでのあるインデックス値 0 を頭文字(一番目のキャラクタ)にするサブストリングを取り出します。
StringInsert	insert ストリングを目的ストリングでの特定の位置に挿入します。
StringLength	ストリングの長さを取ります。
StringMD5	本関数を利用し、MD5 message-digest 演算法を使用するストリングを生成します。
StringMid	本関数を利用し、一個のストリングでのサブストリングを取り出します。
StringReverseFind	あるストリングの中でもう一つのストリングを探します。後ろから探します。

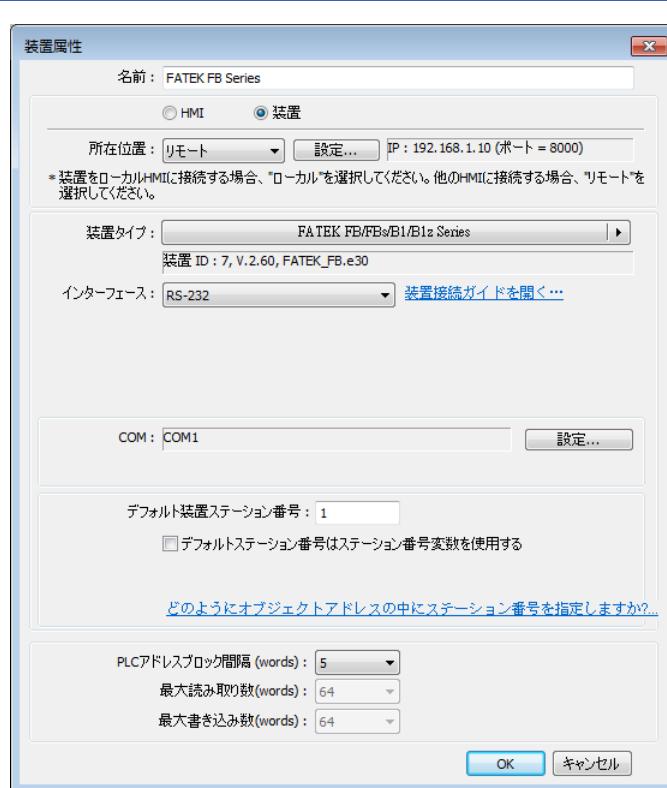
StringSet	キャラクタ配列データを PLC に書き込みます。
StringSetEx	キャラクタ配列データを PLC に書き込み、PLC の応答を待たずに次へ実行します。
StringToUpper	ストリングのキャラクタを全部大文字に変換します。
StringToLower	ストリングのキャラクタを全部小文字に変換します。
StringToReverse	ストリングを逆さにします。
StringTrimLeft	ストリング先頭での特定したキャラクタを切り取ります。
StringTrimRight	ストリング末尾での特定したキャラクタを切り取ります。
Unicode2Utf8	Unicode ストリングを UTF8 ストリングに変換します。
UnicodeCat	2 個の Unicode ストリングを連結します。
UnicodeCompare	2 個の Unicode ストリングの内容が一致するかを比較します。大文字小文字を区分します。
UnicodeCopy	Unicode ストリングをコピーします。
UnicodeExcluding	ある Unicode ストリングの頭文字(一番目のキャラクタ)から特定したキャラクタを含まないサブストリング取り出します。
Utf82Unicode	UTF8 ストリングを Unicode ストリングに変換します。
数学関数	
SQRT	平方根を計算します。
CUBERT	立方根を計算します。
POW	指数を計算します。
SIN	三角関数の正弦を計算します。
COS	三角関数の余弦を計算します。
TAN	三角関数の正接を計算します。
COT	三角関数の余接を計算します。
SEC	三角関数の正割を計算します。
 CSC	三角関数の余割を計算します。
ASIN	逆三角関数の逆正弦を計算します。
ACOS	逆三角関数の逆余弦を計算します。
ATAN	逆三角関数の逆正割を計算します。
LOG	自然対数を計算します。
LOG10	10 から起算する対数を計算します。
RAND	乱数を発生させます。
統計関数	
AVERAGE	配列から平均値を計算します。
HARMEAN	配列から調和平均値を計算します。
MAX	配列から最大値を取得します。
MEDIAN	配列から中央値を取得します。

MIN	配列から最小値を取得します。
STDEVP	配列から標準差を計算します。
STDEVS	配列からサンプル標準差を計算します。
レシピデータベース関数	
RecipeGetData	レシピ内のデータを取ります。
RecipeQuery	レシピ内のデータを調べます。
RecipeQueryGetData	RecipeQueryの検索結果を取ります。
RecipeQueryGetRecordID	RecipeQuery検索結果のレコードID番号を取ります。
RecipeSetData	データをレシピデータベースに書き込みます。
データサンプリング/イベントログ関数	
FindDataSamplingDate	データサンプリングファイルの日付を調べます。
FindDataSamplingIndex	データサンプリングファイルのデータインデックス値を調べます。
FindEventLogDate	イベントログファイルの日付を調べます。
FindEventLogIndex	イベントログファイルのファイルインデックス値を調べます。
パリティ関数	
ADD_SUM	全ての一次元のデータを積算し、checksum(チェックサム)を算出します。
XOR_SUM	全ての一次元のデータをXOR演算をし、checksum(チェックサム)を算出します。
BCC	XOR_SUMと同じです。
CRC	全ての一次元配列でのデータを16-bit CRC演算をし、checksum(チェックサム)を算出します。
CRC8	全ての一次元配列でのデータを8-bit CRC演算をし、checksum(チェックサム)を算出します。
他の関数	
Beep	システム警告音を出します。
Buzzer	ブザーを鳴らす/停止します。
TRACE	マクロ内の変数を監視・印刷します。
GetCnvTagNameIndex	ユーザー定義タグに[読み取り変換]サブルーチンを使用し、それに配列である場合、マクロサブルーチンで本関数を使用すれば、配列のインデックスを取得できます。

18.7.2. PLC

関数名	GetData
シンタックス	GetData(read_data[start], device_ 関数名, device_type, address_offset, data_count) or GetData(read_data, device_ 関数名, device_type, address_offset, 1)

記述	<p>PLCのデータを取ります。データはread_data[start]~read_data[start+data_count-1]との一次元配列変数に保存されます。</p> <p>data_countで読み取るデータの数量を設定します。一般的には、read_dataは一次元配列ですが、もしdata_countは1の場合、read_dataは一次元配列、または普通の変数のどちらでもいいです。以下はPLCから1文字を読み取るには、使用できる2種類の方法です。</p>
	<pre>macro_command main() short read_data_1[2], read_data_2 GetData(read_data_1[0], "FATEK FB Series", RT, 5, 1) GetData(read_data_2, "FATEK FB Series", RT, 5, 1) end macro_command</pre>
	<p>このdevice_関数名は、“システムパラメータ”で装置タイプを設定する時に、設定した“装置”です。ここで装置は下図に示す通り“FATEK FB Series”と設定されます。</p>  <p>device_typeは装置タイプと装置のデータの符号化方式です。例：もしdevice_typeはLW_BINだったら、読み取る装置の装置タイプはLWで、データ符号化方式はBINということを指しています。BIN符号化方式を使用すると、“_BIN”を省略してもいいです。</p> <p>もしdevice_typeはLW_BCDだったら、装置タイプはLWで、データの符号化方式はBCDということを指しています。</p> <p>address_offsetとはPLCでのアドレスオフセットです。</p> <p>例えば、GetData(read_data_1[0], "FATEK FB Series", RT, 5, 1)とは、対象アドレスのオフセットは5ということです。</p> <p>もしaddress_offsetは“N#AAAAAA”というフォーマットを使用すると、NはPLCのステーション番号で、AAAAAAはアドレスのオフセットということを指しています。これは一般的に一個のCOMポートに複数のPLC、またはコントローラを接続している状況に用いられます。</p> <p>例えば：GetData(read_data_1[0], "FATEK FB Series", RT, 2#5, 1)とは、ステーション番号が2の装置内のデータを読み取るのです。もしGetData()に“システムパラメータ/装置リスト”で設定したデフォルトステーション番号を使用したら、ここでステーション番号を記入しなくてもいいです。</p>



装置から読み取るデータ数量は、read_data変数の型とdata_countの値に基づいて決定されます。下表をご参照ください：

read_dataの型	data_countの数値	16ビット数値を読み取る数量
char(8-bit)	1	1
char(8-bit)	2	1
bool(8-bit)	1	1
bool(8-bit)	2	1
short(16-bit)	1	1
short(16-bit)	2	2
int(32-bit)	1	2
int(32-bit)	2	4
float(32-bit)	1	2
float(32-bit)	2	4

GetData()関数が32ビットのデータ(int或いはfloat型)を読み取る時、関数は自動的にこのデータを変換します。例：

```
macro_command main()
float f
GetData(f, "MODBUS ", 6x, 2, 1) // f の中には浮動式小数点数型のデータに
```

	なる end macro_command
例	<pre>macro_command main() bool a bool b bool b_array[30] char c char c_array[20] short s short s_array[50] int i int i_array[10] float f float f_array[15] // LB2の状態を読み取り、変数aに入れる GetData(a, "Local HMI", LB, 2, 1) // LB0~LB29との合計30個の状態を読み取り、変数b_array[0]~b_array[29]に入れる GetData(b_array[0], "Local HMI", LB, 0, 30) // LW-0の下位バイトを読み取り、変数cに入れる // 注意：charは1 Byteで、LWは一個のアドレスが2 Byte(1ワード)を占め、ワードレジスタの第一個のバイトを読み取る時、当該ワードの下位バイトを読み取ることになります。 // Ex: 仮にLW-0の値は0x0201の場合、cは0x01を読み取ることになります GetData(c, "Local HMI", LW, 0, 1) // LW-1~LW-10のデータを読み取り、c_array[0] ~ c_array[19]に入れる GetData(c_array[0], "Local HMI", LW, 0, 20) // LW-2のデータを読み取り、変数sに入れる GetData(s, "Local HMI", LW, 2, 1) // LW-0~LW-49との合計50ワードを読み取り、変数s_array[0]~s_array[49]に入れる GetData(s_array[0], "Local HMI", LW, 0, 50) // LW-6~LW-7との2ワードを読み取り、変数eに入れる // Ex: 仮にLW-6の値は0x0002、LW-7の値は0x0001の場合、iは0x00010002(65538)を読み取ることになります。 // 注意：一個のintデータは2個のワードを占める(32-bit) GetData(i, "Local HMI", LW, 6, 1) // LW-0~LW-19との合計20ワードを読み取り、変数i_array[0]~i_array[9]に入れ(合計10個のint型変数)、配列i_array[10]の変数型をintに定義します。 GetData(i_array[0], "Local HMI", LW, 0, 10) // LW-10~LW-11のデータ読み取り、変数fに入れる // floatこの場合、変数fの型はfloat</pre>

```

GetData(f, "Local HMI", LW, 10, 1)

// LW-0~LW-29との30ワードを読み取り、f_array[0]~f_array[14]に入る(合計15
個のfloat型変数)、配列f_array[15]の変数型をfloatに定義します。
// 注意：一個のfloat tデータは2個のワードを占める
GetData(f_array[0], "Local HMI", LW, 0, 15)

end macro_command

```

関数名	GetDataEx
シンタックス	GetDataEx(read_data[start], device_ 関数名, device_type, address_offset, data_count) or GetDataEx(read_data, device_ 関数名, device_type, address_offset, 1)
記述	PLCのデータを受け取り、PLCの返信を待たずに次へ実行します。 read_data、device_ 関数名、device_type、address_offset及びdata_countの説明はGetDataと同じです。
例	<pre> macro_command main() bool a bool b bool b_array[30] char c char c_array[20] short s short s_array[50] int i int i_array[10] float f float f_array[15] // LB-2の状態を読み取り、変数aに入れる GetDataEx(a, "LocalHMI", LB, 2, 1) // LB-0~LB-29との合計30個の状態を読み取り、変数b[0]~b[29]に入れる GetDataEx(b[0], "LocalHMI", LB, 0, 30) // LW-0下位バイトを読み取り、変数cに入れる // 注意：charは1 Byteで、LWは一個のアドレスが2 Byte(1ワード)を占め、ワードレジスタの第一個のバイトを読み取る時、当該ワードの下位バイトを読み取ることになります。 // Ex: 仮にLW-0の値は0x0201の場合、cは0x01を読み取ることになります。 GetDataEx(c, "Local HMI", LW, 0, 1) // LW-1~LW-10のデータを読み取り、c_array[0] ~ c_array[19]に入れる GetDataEx(c_array[0], "Local HMI", LW, 0, 20) // LW-2のデータを読み取り、変数sに入れる GetDataEx(s, "Local HMI", LW, 2, 1) </pre>

	<pre>// LW-0~LW-49との50ワードを読み取り、変数s_array[0]~s_array[49]に入れる GetDataEx(s_array[0], "Local HMI", LW, 0, 50) // LW-6~LW-7との2ワードを読み取り、変数eに入れる // Ex: 仮にLW-6の値は0x0002、LW-7の値は0x0001の場合、eは0x00010002(65538)を読み取ることになります。 // 注意：一個のintデータは2個のワードを占める(32-bit) GetDataEx(e, "LocalHMI", LW, 6, 1) // LW-0~LW-19との合計20ワードを変数i_array[0]~i_array[9]に入れ(合計10個のint型変数)、配列i_array[10]の変数型をintに定義します。 GetDataEx(i_array[0], "Local HMI", LW, 0, 10) // LW-10~LW-11のデータ読み取り、変数fに入れる // 注意：この場合、変数fの型はfloat GetDataEx(f, "Local HMI", LW, 10, 1) // LW-0~LW-29との30ワードを読み取り、f_array[0]~f_array[14]に入れる(合計15個のfloat型変数)、配列f_array[15]の変数型をfloatに定義します。 // 注意：一個のfloat tデータは2個のワードを占める GetDataEx(f_array[0], "Local HMI", LW, 0, 15) end macro_command</pre>
--	---

関数名	GetError
シンタック ス	GetError(err)
記述	エラーコードを受け取ります。
例	<pre>macro_command main() short err char byData[10] GetDataEx(byData[0], "MODBUS RTU", 4x, 1, 10) // 10 byteのデータを読み取る // エラーコード(err)は0の場合、GetDataExの実行に成功したと示している GetErr(err) // エラーコードを読み取り、変数errに書き込む end macro_command エラーコード 0: ノーマル 1: GetDataExエラー 2: SetDataExエラー</pre>

関数名	SetData
-----	---------

シンタックス	SetData(send_data[start], device_関数名, device_type, address_offset, data_count) or SetData(send_data, device_関数名, device_type, address_offset, 1)																		
記述	<p>データを PLC に書き込みます。データは send_data[start]~send_data[start+data_count-1] に保存されます。</p> <p>data_count で読み取るデータの数量を設定します。一般的には、send_data は一個の配列ですが、もし data_count は 1 の場合、send_data は配列、または普通の変数のどちらでもいいです。以下は 1 個のデータを PLC に書き込む方法です。</p> <pre style="background-color: #f0f0f0; padding: 10px;">macro_command main() short send_data_1[2]={5, 6}, send_data_2=5 SetData(send_data_1[0], "FATEK FB Series", RT, 5, 1) SetData(send_data_2, "FATEK FB Series", RT, 5, 1) end macro_command</pre> <p>device_関数名については上記の説明をご参照ください。</p> <p>device_type は装置タイプと PLC のデータの符号化方式です。例：もし device_type は LW_BIN だったら、対象装置の装置タイプは LW で、データ符号化方式は BIN ということを指しています。BIN 符号化方式を使用すると、“_BIN”を省略してもいいです。</p> <p>もし device_type は LW_BCD だったら、装置タイプは LW で、データの符号化方式は BCD ということを指しています。</p> <p>address_offset とは PLC でのアドレスオフセットです。</p> <p>例えば、SetData(read_data_1[0], "FATEK FB Series", RT, 5, 1) とは、対象装置アドレスのオフセットは 5 ということです。</p> <p>もし address_offset は “N#AAAAAA” というフォーマットを使用すると、N は PLC のステーション番号で、AAAAAA はアドレスのオフセットということを指しています。これは一般的に一個の COM ポートに複数の PLC、またはコントローラを接続している状況に用いられます。例えば：SetData(send_data_1[0], "FATEK FB Series", RT, 2#5, 1) とは、ステーション番号が 2 の PLC 内のデータを読み取るのでです。もし SetData() に “システムパラメータ/装置リスト” で設定したデフォルトステーション番号を使用したら、ここでステーション番号を記入しなくてもいいです。</p> <p>PLC に送信するデータ数量は、send_data 変数の型と data_count の値を基づいて決定します。下表をご参照ください：</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 5px;">send_data のタイプ</th> <th style="text-align: center; padding: 5px;">data_count の数值</th> <th style="text-align: center; padding: 5px;">送信する 16 ビットデータの数量</th> </tr> </thead> <tbody> <tr> <td style="text-align: center; padding: 5px;">char(8-bit)</td> <td style="text-align: center; padding: 5px;">1</td> <td style="text-align: center; padding: 5px;">1</td> </tr> <tr> <td style="text-align: center; padding: 5px;">char(8-bit)</td> <td style="text-align: center; padding: 5px;">2</td> <td style="text-align: center; padding: 5px;">1</td> </tr> <tr> <td style="text-align: center; padding: 5px;">bool(8-bit)</td> <td style="text-align: center; padding: 5px;">1</td> <td style="text-align: center; padding: 5px;">1</td> </tr> <tr> <td style="text-align: center; padding: 5px;">bool(8-bit)</td> <td style="text-align: center; padding: 5px;">2</td> <td style="text-align: center; padding: 5px;">1</td> </tr> <tr> <td style="text-align: center; padding: 5px;">short(16-bit)</td> <td style="text-align: center; padding: 5px;">1</td> <td style="text-align: center; padding: 5px;">1</td> </tr> </tbody> </table>	send_data のタイプ	data_count の数值	送信する 16 ビットデータの数量	char(8-bit)	1	1	char(8-bit)	2	1	bool(8-bit)	1	1	bool(8-bit)	2	1	short(16-bit)	1	1
send_data のタイプ	data_count の数值	送信する 16 ビットデータの数量																	
char(8-bit)	1	1																	
char(8-bit)	2	1																	
bool(8-bit)	1	1																	
bool(8-bit)	2	1																	
short(16-bit)	1	1																	

short(16-bit)	2	2
int(32-bit)	1	2
int(32-bit)	2	4
float(32-bit)	1	2
float(32-bit)	2	4

Setdata()関数が 32 ビットのデータ(int 或いは float 型)を PLC に書き込む時、関数は自動的にこのデータを変換します。例：

```
macro_commandmain()
float f=2.6
SetData(f, "MODBUS ", 6x, 2, 1)// ここでは一個の浮動小数点数を PLC に書き込む
end macro_command
```

例

```
macro_command main()
int i
bool a = true
bool b[30]
short c = false
short d[50]
int e = 5
int f[10]

for i = 0 to 29
b[i] = true
next i

for i = 0 to 49
d[i] = i * 2
next i

for i = 0 to 9
f [i] = i * 3
next i

// 変数aの数値をLB2に設定する
SetData(a, "LocalHMI", LB, 2, 1)

// LB0~LB29合計30個ビットの状態を設定する
SetData(b[0], "LocalHMI", LB, 0, 30)

// 変数cの数値をLW-2に設定する
SetData(c, "LocalHMI", LW, 2, 1)

// LW-0~LW-49との合計50個データを設定する
SetData(d[0], "LocalHMI", LW, 0, 50)

// 変数eに数値をLW-6~LW-7に設定する。注意：変数eの型はint。
```

	<pre> SetData(e, "LocalHMI", LW, 6, 1) // LW-0~LW-19との合計20ワードのデータを設定する // 10個の整数型データは20個ワードに相当する。一個のintデータは2個のワードを占めるから。 SetData(f[0], "LocalHMI", LW, 0, 10) end macro_command </pre>
--	--

関数名	SetDataEx
シンタックス	<pre> SetDataEx(send_data[start], device_関数名, device_type, address_offset, data_count) or SetDataEx(send_data, device_関数名, device_type, address_offset, 1) </pre>
記述	データをPLCに書き込み、PLCの返信を待たずに次へ実行します。 send_data、device_関数名、device_type、address_offsetとdata_countの説明はSetDataと同じです。
例	<pre> macro_command main() int i bool a = true bool b[30] short c = false short d[50] int e = 5 int f[10] for i = 0 to 29 b[i] = true next i for i = 0 to 49 d[i] = i * 2 next i for i = 0 to 9 f [i] = i * 3 next i // aの数値をLB2の中に設定する SetDataEx(a, "LocalHMI", LB, 2, 1) // LB0~LB29との合計30個の状態を設定する SetDataEx(b[0], "LocalHMI", LB, 0, 30) // 変数cの数値をLW-2の中に設定する SetDataEx(c, "LocalHMI", LW, 2, 1) </pre>

	<pre>// LW-0~LW-49との合計50個のデータを設定する SetDataEx(d[0], "LocalHMI", LW, 0, 50) // 変数eの数値をLW-6~LW-7に書き込む。注意：変数eの型はint。 SetDataEx(e, "LocalHMI", LW, 6, 1) // LW-0~LW-19との合計20ワードの数値を設定する // 10個の整数型データは20個のワードに相当する。一個のintデータは2個のワードを占めるから。 SetDataEx(f[0], "LocalHMI", LW, 0, 10) endmacro_command</pre>
--	--

18.7.3. Free protocol

関数名	GetCTS
シンタックス	GetCTS(com_port, result)
記述	<p>RS-232のCTS信号を検知します。</p> <p>com_portはCOMポート番号を示しています。COM1をサポートします。このパラメータは変数、または定数のどちらでもいいです。resultパラメータは変数である必要があります。</p> <p>本関数はCTS電位を検知でき、CTSが高電位になった際に、resultは1を書き込み、さもなければ0を書き込みます。</p>
例	<pre>macro_commandmain() char com_port=1 char result GetCTS(com_port, result) // COM1のCTS電位を検知する GetCTS(1, result) // COM1のCTS電位を検知する end macro_command</pre>

関数名	IMPORT
シンタックス	IMPORT(read_data[start], device_関数名, read_count, return_value)
記述	<p>COMポートまたはEthernetポートを通してデータをHMIに読み取らせます。これらのデータはread-data[start]~read-data[start+read-count-1]との一次元配列に保存されます。device_関数名は前述と同様です。</p> <p>read-countで読み取るコマンドの長さを設定します。定数または変数のどちらでもいいです。もしこの関数がPLC或いはコントローラからデータを読み取れたら、return_valueの数値は1になり、さもなければ0になります。</p>
例	以下はIMPORT関数でMODBUS の保持レジスタからデータを読み取る例です。

```

// 保持レジスタのデータを読み取る
macro_commandmain()
char command[32], response[32]
short address, checksum
short read_no, return_value, read_data[2]

FILL(command[0], 0, 32) // コマンド初期化
FILL(response[0], 0, 32)

command[0]=0x1 // ステーション番号
command[1]=0x3 // ファンクションコード：保持レジスタのデータを読み取る

address=0
HIBYTE(address, command[2])
LOBYTE(address, command[3])

read_no=2// read 2 words(4x_1and4x_2)
HIBYTE(read_no, command[4])
LOBYTE(read_no, command[5])

CRC(command[0], checksum, 6)

LOBYTE(checksum, command[6])
HIBYTE(checksum, command[7])

// OUTPORT関数を使用して、コマンドを送り出す
OUTPORT(command[0], "MODBUS RTU Device", 8)

// IMPORT関数を使用して、レスポンスを読み取る
IMPORT(response[0], "MODBUS RTU Device", 9, return_value)

if return_value > 0 then
read_data[0] = response[4] + (response[3] << 8)// data in 4x_1
read_data[1] = response[6] + (response[5] << 8)// data in 4x_2

SetData(read_data[0], "Local HMI", LW, 100, 2)
end if

end macro_command

```

関数名	IMPORT2
シンタックス	IMPORT2(response[start], device_name, receive_len, wait_time)
記述	COMポートまたはイーサネットを通してHMIにデータを読み取らせます。これらのデータはresponseとの一次元配列に保存されます。device_nameはOUTPUTと同様です。 receive_lenに送信したデータの長さを保存します。その数値は変数である必要があります。receive_lenの最大長さはresponse配列が宣言したサイズに超えません。wait_timeは待ち時間で(単位 : millisecond)、定数或いは変数のどちらでもいい

	いです。データの読み取りが完了後、指定した待ち時間で他のデータがもらつてない場合、本関数は実行を完了し、結果を返ります。
例	<pre>macro_command main() short wResponse[6], receive_len, wait_time=20 IMPORT2(wResponse[0], "Free Protocol", receive_len, wait_time) // wait_time 単位 : millisecond if receive_len > 0 then SetData(wResponse[0], "Local HMI", LW, 0, 6) // レスポンスをLW0に書き込む end if end macro_command</pre>

関数名稱	IMPORT3
シンタック ス	IMPORT3(response[start], device_name, read_count, receive_len)
記述	COMポートまたはイーサネットを通してHMIにデータを読み取らせます。これらのデータはresponseとの一次元配列に保存され、データを読み取る時には指定した数量を読み取り、読み取っていなかったデータはHMIのバッファメモリーに保存されます。device_nameの説明はOUTPORTと同じです。read_countは毎回読み取ったデータの長さです。receive_lenには送信したデータのバイト長さを保存します。その数値は変数である必要があります。receive_lenの最大長さはresponse配列が宣言したサイズに超えません。
例	<pre>macro_command main() short wResponse[6], receive_len IMPORT3(wResponse[0], "Free Protocol", 6, receive_len) // 6 wordsを読み取る if receive_len >= 6 then SetData(wResponse[0], "Local HMI", LW, 0, 6) // レスポンスをLW0に書き込む end if end macro_command</pre>

関数名	OUTPORT
シンタック ス	OUTPORT(source[start], device_ 関数名, data_count)
記述	source[start]から source[start+ data_count-1]までの全てのデータを COM ポート、またはイーサネットを通して PLC 或いはコントローラーに送信します。 device_ 関数名は“装置リスト”に定義した“装置”で、この device には“FreeProtocol”との PLC タイプを選択すること。 Data_count は送信するデータの数で、定数または変数のどちらでもいいです。
例	OUTPORT 関数を使用するには、下図で示されたように装置タイプのところに

	<p>“FreeProtocol”を選択する必要があります。</p>  <p>この device_ 関数名は“MODBUS RTU Device”となります。ポートの属性もこの“システムパラメータ”での設定を基づきます。例として、ここでは(9600、E、8、1...)と設定しました。</p> <p>下記では MODBUS RTU のフリー通信プロトコルを使用し、シングルレジスタを ON に設定する例です。</p> <pre> macro_command main() char command[32] short address, checksum FILL(command[0], 0, 32) // 初期化コマンド command[0] = 0x1 // ステーション番号 command[1] = 0x5 // ファンクションコード: シングルレジスタを書きこむ address = 0 HIBYTE(address, command[2]) LOBYTE(address, command[3]) command[4] = 0xff // 当bitをONに設定させる command[5] = 0 CRC(command[0], checksum, 6) LOBYTE(checksum, command[6]) HIBYTE(checksum, command[7]) // COMポートを通じて、コマンドを送り出す OUTPORT(command[0], "MODBUS RTU Device", 8) end macro_command </pre>
--	---

関数名	PURGE
シンタックス	PURGE(com_port)
記述	com_portはCOMポート番号を指しています。COM1~COM3をサポートします。 このパラメータは変数、または定数のどちらでもいいです。 本コマンドでCOM portのインポート/エクスポートのバッファをクリアする

	ことができます。
例	<pre>macro_command main() int com_port=3 PURGE(com_port) PURGE(1) end macro_command</pre>

関数名	SetRTS
シンタック ス	SetRTS(com_port, source)
記述	<p>RS-232のRTS状態を設定します。 com_portはCOMポート番号を指しています。COM1をサポートします。このパラメータは変数、または定数のどちらでもいいです。sourceパラメータも変数、定数のどちらでもいいです。</p> <p>送信したsourceパラメータ数値が0より大きい場合、RTS電位を上げ、sourceパラメータが0に等しい場合、RTS電位を下げます。</p>
例	<pre>macro_command main() char com_port=1 char value=1 SetRTS(com_port, value) // value>0、 COM1のRTS電位を上げる SetRTS(1, 0) // COM1のRTS電位を下げる end macro_command</pre>

18.7.4. マクロ制御

関数名	ASYNC_TRIG_MACRO
シンタック ス	ASYNC_TRIG_MACRO(macro_id or name)
記述	<p>一個の実行中のマクロで本関数を使用し、他のマクロをトリガーすることができます。他のマクロのmacro_id或いは名前をパラメータにし、トリガーされたマクロの[番号]または[名前]を指定することができます。</p> <p>マクロで本関数を使用した場合、システムはトリガーしたマクロが実行完了するのを待たずにすぐ後続のコマンドを実行します。</p> <p>macro_idを定数、または変数で表示してもいいです。</p>
例	<pre>macro_command main() char ON = 1, OFF = 0 SetData(ON, "Local HMI", LB, 0, 1) ASYNC_TRIG_MACRO(5) // call a macro (its ID is 5) ASYNC_TRIG_MACRO("macro_1") // call a macro (its name is macro_1)</pre>

	<pre>SetData(OFF, "Local HMI", LB, 0, 1) end macro_command</pre>
--	---

関数名	DELAY
シンタック ス	DELAY(time)
記述	マクロを一時停止させます。継続時間は少なくともこの指定した時間であること。時間の単位はミリ秒です。 Time は定数、または変数のどちらでもいいです。
例	<pre>macro_command main() int time == 500 DELAY(100)// delay 100 ms DELAY(time)// delay 500 ms end macro_command</pre>

関数名	SYNC_TRIG_MACRO
シンタック ス	SYNC_TRIG_MACRO(macro_id or name)
記述	一個の実行中のマクロで本関数を使用し、他のマクロの実行をトリガーすることができます。他のマクロのmacro_id或いは名前をパラメータにし、トリガーされたマクロの[番号]または[名前]を指定することができます。 マクロで本関数を使用した場合、システムは元のマクロを一時停止し、トリガーしたマクロを実行完了してから、後続のコマンドを実行します。 macro_idを定数、または変数で表示してもいいです。
例	<pre>macro_commandmain() charON=1, OFF=0 SetData(ON, "LocalHMI", LB, 0, 1) SYNC_TRIG_MACRO(5)// callamacro(itsIDis5) SYNC_TRIG_MACRO("macro_1")// callamacro(itsnameismacro_1) SetData(OFF, "LocalHMI", LB, 0, 1) endmacro_command</pre>

18.7.5. データ操作

関数名	FILL
シンタック	FILL(source[start], preset, count)

ス	
記述	順番で事前に設定した数値 (preset)を一次元配列 source[start]の先頭部分に入れ、入れるデータの数は count で決めます。 source と start は必ず変数でなければなりません。preset は定数、または変数のどちらでもいいです。
例	<pre>macro_command main() char result[4] char preset FILL(result[0], 0x30, 4) // result[0] is 0x30, result[1] is 0x30, , result[2] is 0x30, , result[3] is 0x30 preset = 0x31 FILL(result[0], preset, 2) // result[0] is 0x31, result[1] is 0x31 end macro_command</pre>

関数名	SWAPB
シンタック ス	SWAPB(source, result)
記述	16 ビットのソースデータの上位バイトと下位バイトを逆さにし、それにその結果を result 変数の中に入れます Source は定数、または変数のどちらでもいいが、奇数 result は必ず変数でなければなりません。
例	<pre>macro_command main() short source, result SWAPB(0x5678, result) // result is 0x7856 source = 0x123 SWAPB(source, result) // result is 0x2301 end macro_command</pre>

関数名	SWAPW
シンタック ス	SWAPW(source, result)
記述	32 ビットの整数型データの上位バイトと下位バイトを逆さにし、それにその結果を result 変数の中に入れます。 Source は定数、または変数のどちらでもいいが、奇数 result が必ず変数でなければなりません。
例	<pre>macro_command main() int source, result SWAPW(0x12345678, result) // result is 0x56781234</pre>

	<pre>source = 0x12345 SWAPW(source, result) // result is 0x23450001 end macro_command</pre>
--	--

関数名	LOBYTE
シンタック ス	LOBYTE(source, result)
記述	16 ビットデータの下位バイトを取り、result 変数の中に入れます。 source は定数、または変数のどちらでもいいが、result は必ず変数でなければなりません。
例	<pre>macro_command main() short source, result LOBYTE(0x1234, result) // result is 0x34 source = 0x123 LOBYTE(source, result) // result is 0x23 end macro_command</pre>

関数名	HIBYTE
シンタック ス	HIBYTE(source, result)
記述	16 ビットデータの上位バイトを取り、result 変数の中に入れます。 source は定数、または変数のどちらでもいいが、result は必ず変数でなければなりません。
例	<pre>macro_command main() short source, result HIBYTE(0x1234, result) // result is 0x12 source = 0x123 HIBYTE(source, result) // result is 0x01 end macro_command</pre>

関数名稱	LOWORD
シンタック ス	LOWORD(source, result)
記述	32 ビットデータの下キャラクタを取り、それにその結果を result 変数の中に入れます。 source は定数、または変数のどちらでもいいが、result は必ず変数でなければなりません。

例	<pre>macro_command main() int source, result LOWORD(0x12345678, result) // result is 0x5678 source = 0x12345 LOWORD(source, result) // result is 0x2345 end macro_command</pre>
---	--

関数名	HIWORD
シンタック ス	HIWORD(source, result)
記述	<p>32 ビットデータの上位キャラクタを取り、それにその結果を <code>result</code> 変数の中に入れます。</p> <p><code>source</code> は定数、または変数のどちらでもいいが、<code>result</code> は必ず変数でなければなりません。</p>
例	<pre>macro_command main() int source, result HIWORD(0x12345678, result) // result is 0x1234 source = 0x12345 HIWORD(source, result) // result is 0x0001 end macro_command</pre>

関数名	INVBIT
シンタック ス	INVBIT(source, result, bit_pos)
記述	<p>データもしくは変数(<code>source</code>)の指定したビットアドレスの状態を逆さにし、それに変更後のデータを <code>result</code> 変数の中に入れます。</p> <p><code>source</code> と <code>bit_pos</code> は定数、または変数のどちらでもいいが、<code>result</code> は必ず変数でなければなりません。</p>
例	<pre>macro_command main() int source, result short bit_pos INVBIT(4, result, 1)// result = 6 source = 6 bit_pos = 1 INVBIT(source, result, bit_pos)// result = 4 end macro_command</pre>

関数名	SETBITON
シンタックス	SETBITON(source, result, bit_pos)
記述	データもしくは変数(source)の指定したビットアドレスを 1 に設定し、それに変更後のデータを result 変数の中に入れます。 source と bit_pos は定数、または変数のどちらでもいいが、result は必ず変数でなければなりません。
例	<pre>macro_command main() int source, result short bit_pos SETBITON(1, result, 3)// result is 9 source = 0 bit_pos = 2 SETBITON (source, result, bit_pos)// result is 4 end macro_command</pre>

関数名	SETBITOFF
シンタックス	SETBITOFF(source, result, bit_pos)
記述	データもしくは変数(source)の指定したビットアドレスを 0 に設定し、それに変更後のデータを result 変数の中に入れます。 source と bit_pos は定数、または変数のどちらでもいいが、result は必ず変数でなければなりません。
例	<pre>macro_command main() int source, result short bit_pos SETBITOFF(9, result, 3)// result is 1 source = 4 bit_pos = 2 SETBITOFF(source, result, bit_pos)// result is 0 end macro_command</pre>

関数名	GETBIT
シンタックス	GETBIT(source, result, bit_pos)
記述	データもしくは変数(source)の指定したビット状態を取り、それにその結果を result 変数の中に入れます。result は 1 或いは 0 になります。 source と bit_pos は定数、または変数のどちらでもいいが、result は必ず変数でなければなりません。

例	<pre>macro_command main() int source, result short bit_pos GETBIT(9, result, 3) // result is 1 source = 4 bit_pos = 2 GETBIT(source, result, bit_pos) // result is 1 end macro_command</pre>
---	---

18.7.6. データ変換

関数名	ASCII2DEC
シンタック ス	ASCII2DEC(source[start], result, len)
記述	<p>キャラクタ型 ASCII データ(source)を十進数フォーマットに変換し、それに result 変数の中に入れます。ASCII の長さは len で、一番目のキャラクタは source[start] です。</p> <p>source と len 定数、または変数のどちらでもいいが、奇数 result は必ず変数でなければなりません。start は定数でなければなりません。</p>
例	<pre>macro_command main() char source[4] short result source[0] = '5' source[1] = '6' source[2] = '7' source[3] = '8' ASCII2DEC(source[0], result, 4) // result is 5678 end macro_command</pre>

関数名	ASCII2FLOAT
シンタック ス	ASCII2FLOAT(source[start], result, len)
記述	<p>キャラクタ型 ASCII データ(source)を浮動式小数点数フォーマットに変換し、それに result 変数の中に入れます。キャラクタの長さは len です。一番目のキャラクタは source[start] です。</p> <p>source と len 定数、または変数のどちらでもいいが、奇数 result は必ず変数でなければなりません。start は定数でなければなりません。</p>
例	macro_command main()

	<pre> char source[4] float result source[0] = '5' source[1] = '6' source[2] = '.' source[3] = '8' ASCII2FLOAT(source[0], result, 4) // result is 56.8 end macro_command </pre>
--	---

関数名	ASCII2HEX
シンタックス	ASCII2HEX(source[start], result, len)
記述	ASCII キャラクタ型データ(source)を十六進数フォーマットに変換し、それに result 変数の中に入れます。キャラクタの長さは len です。一番目のキャラクタは source[start]です。 source と len 定数、または変数のどちらでもいいが、奇数 result は必ず変数でなければなりません。start は定数でなければなりません。
例	<pre> macro_command main() char source[4] short result source[0] = '5' source[1] = '6' source[2] = '7' source[3] = '8' ASCII2HEX (source[0], result, 4) // result is 0x5678 end macro_command </pre>

関数名	BIN2BCD
シンタックス	BIN2BCD(source, result)
記述	BIN フォーマットのデータ(source)を BCD フォーマット(result)に変換します。データソース source は定数、或いは変数のどちらでもいいが、結果を示す result は必ず変数でなければなりません。
例	<pre> macro_command main() short source, result BIN2BCD(1234, result) // result is 0x1234 source = 5678 </pre>

	BIN2BCD(source, result) // result is 0x5678 end macro_command
--	--

関数名	BCD2BIN
シンタック ス	BCD2BIN(source, result)
記述	BCD フォーマットのデータ(source)を BIN フォーマット(result)に変換します。データソース source は定数、或いは変数のどちらでもいいが、結果を示す result は必ず変数でなければなりません。
例	macro_command main() short source, result BCD2BIN(0x1234, result) // result is 1234 source = 0x5678 BCD2BIN(source, result) // result is 5678 end macro_command

関数名	DATE2ASCII
シンタック ス	DATE2ASCII(day_offset, date[start], count)
記述	現在の日付データを day_offset を使用して ASCII フォーマットのデータに変換し、一個の一次元配列(date)に保存します。count は変換後のストリングの長さを示し、この長さは保存の結果によって異なります。 day_offset と count は定数、或いは変数のどちらでもいいが、start は必ず変数でなければなりません。
例	macro_command main() char result[10] DATE2ASCII(5, result[0], 10) // result[0]~[9] == "2019/02/16"// today is 2019/02/11 end macro_command

関数名	DATE2DEC
シンタック ス	DATE2DEC(day_offset, date)
記述	現在の日付データを day_offset を使用して十進数フォーマットに変換し、date 変数に保存します。 day_offset は定数、或いは変数のどちらでもいいが、date は必ず変数でなければ

	なりません。
例	<pre>macro_command main() int day_offset = 5, date DATE2DEC(0, date) // date == 20190211 (Today is 2019/02/11) DATE2DEC(day_offset, date) // date == 20190216 (20190211 + 5) end macro_command</pre>

関数名	DEC2ASCII
シンタック ス	DEC2ASCII(source, result[start], len)
記述	<p>十進数フォーマットのデータ(source)を ASCII フォーマットに変換し、それに一次元の配列(result)の中保存します。len は変換後のストリングの長さを示し、同時にこの長さも結果を保存する一次元のデータフォーマットにより決められます。例：仮に result 一次元のフォーマットは“char”(キャラクタ型、長さは 1 バイト)の場合、長さは“バイト数*len”です。もし result 一次元のフォーマットは“short”(ショート整数型、2 バイト)の場合、長さは“word*len”です。このように類推します。</p> <p>変換後一番目のワードを result[start]に入れ、二番目のワードを result[start+1]に入れ、最後のワードを result[start+(len-1)]に入れます。</p> <p>source と len は定数、または変数のどちらでもいいが、奇数 result は必ず変数でなければなりません。start は定数でなければなりません。</p>
例	<pre>macro_command main() short source char result1[4] short result2[4] char result3[6] source = 5678 DEC2ASCII(source, result1[0], 4) // result1[0] is '5', result1[1] is '6', result1[2] is '7', result1[3] is '8' // the length of the string (result1) is 4 bytes(= 1 * 4) DEC2ASCII(source, result2[0], 4) // result2[0] is '5', result2[1] is '6', result2[2] is '7', result2[3] is '8' // the length of the string (result2) is 8 bytes(= 2 * 4) source=-123 DEC2ASCII(source3, result3[0], 6) // result1[0] is '-', result1[1] is '0', result1[2] is '0', result1[3] is '1' // result1[4] is '2', result1[5] is '3' // the length of the string (result1) is 6 bytes(= 1 * 6) end macro_command</pre>

関数名	FLOAT2ASCII
シンタック ス	FLOAT2ASCII(source, result[start], len)
記述	<p>浮動小数点数フォーマットのデータ(source)を ASCII フォーマットに変換し、それに一次元の配列(result)の中保存します。len は変換後のストリングの長さを示し、同時にこの長さも結果を保存する一次元のデータフォーマットにより決められます。例：仮に result 一次元のフォーマットは“char”(キャラクタ型、長さは 1 バイト)の場合、長さは“バイト数*len”です。もし result 一次元のフォーマットは“short”(ショート整数型、2 バイト)の場合、長さは“word*len”です。このように類推します。</p> <p>source と len 定数、または変数のどちらでもいいが、奇数 result は変数でなければなりません。start は定数でなければなりません。</p>
例	<pre>macro_command main() float source char result[4] source = 56.8 FLOAT2ASCII (source, result[0], 4) // result[0] is '5', result[1] is '6', result[2] is '.', result[3] is '8' end macro_command</pre>

関数名	HEX2ASCII
シンタック ス	HEX2ASCII(source, result[start], len)
記述	<p>十六進数フォーマットのデータ(source)を ASCII フォーマットに変換し、それに一次元の配列(result)の中保存します。len は変換後のストリングの長さを示し、同時にこの長さも結果を保存する一次元のデータフォーマットにより決められます。例：仮に result 一次元のフォーマットは“char”(キャラクタ型、長さは 1 バイト)の場合、長さは“バイト数*len”です。もし result 一次元のフォーマットは“short”(ショート整数型、2 バイト)の場合、長さは“word*len”です。このように類推します。</p> <p>source と len 定数、または変数のどちらでもいいが、奇数 result は必ず変数でなければなりません。start は定数でなければなりません。</p>
例	<pre>macro_command main() short source char result[4] source = 0x5678 HEX2ASCII(source, result[0], 4) // result[0] is '5', result[1] is '6', result[2] is '7', result[3] is '8' end macro_command</pre>

関数名	StringDecAsc2Bin
シンタック ス	success=StringDecAsc2Bin(source[start], destination) 或いは success=StringDecAsc2Bin("source", destination)
記述	<p>本関数で十進数ストリングを整数に変更します。</p> <p>ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。</p> <p>destinationは必ず変数でなければなりません。ここで変換後の整数值を保存します。</p> <p>実行が完成されると、bool型戻り値をsuccess欄に返信します。変換に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。</p> <p>ソースストリングは十進数ストリングである必要があり、ストリングにプラス記号とマイナス記号、または'0'~'9'以外キャラクタが含まれていたら、関数はfalseを返信します。</p> <p>successに記入してもしなくてもいいです。</p>
例	<pre>macro_command main() char src1[5] = "12345" int result1 bool success1 success1 = StringDecAsc2Bin(src1[0], result1) // success1 = true、result1 は "12345" char src2[5] = "-6789" short result2 bool success2 success2 = StringDecAsc2Bin(src2[0], result2) // success2 = true, result2は"-6789" char result3 bool success3 success3 = StringDecAsc2Bin("32768", result2) // success3 = true、但し結果はresult3が表示できる範囲を超えている char src4[2] = "4b" char result4 bool success4 success4 = StringDecAsc2Bin (src4[0], result4) // success4 = false、src4 にはプラス記号とマイナス記号、または'0'~'9'以外の キャラクタが含まれていたから end macro_command</pre>

関数名	StringBin2DecAsc
シンタック ス	success=StringBin2DecAsc(source, destination[start])

記述	<p>本関数で整数を十進数ストリングに変更します。 ソースsourceは定数、または変数のどちらでもいいです。 destination[start]は必ず一次元キャラクタ配列変数でなければなりません。ここで変換後の十進数ストリングを保存します。 実行が完成されると、bool型戻り値をsuccess欄に返信します。変換に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。 変換後の十進数ストリングの長さが目的配列のサイズより大きい場合、関数はfalseを返信します。 successに記入してもしなくてもいいです。</p>
例	<pre>macro_command main() int src1 = 2147483647 char dest1[20] bool success1 success1 = StringBin2DecAsc(src1, dest1[0]) // success1 = true、dest1は"2147483647" short src2 = 0x3c char dest2[20] bool success2 success2 = StringBin2DecAsc(src2, dest2[0]) // success2 = true、dest2は"60" int src3 = 2147483647 char dest3[5] bool success3 success3 = StringBin2DecAsc(src3, dest3[0]) // success3 = false、dest3 内容が変わらない end macro_command</pre>

関数名	StringDecAsc2Float
シンタック ス	success=StringDecAsc2Float(source[start], destination) 或いは success=StringDecAsc2Float("source", destination)
記述	<p>本関数で十進数ストリングを浮動式小数点数に変換します。 ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 destinationは必ず変数でなければなりません。ここで変換後の浮動小数点数を保存します。 実行が完成されると、bool型戻り値をsuccess欄に返信します。変換に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。 ソースストリングは十進数ストリングである必要があり、ストリングに'0'~'9'以外のキャラクタが含まれていたら、関数はfalseを返信します。 successに記入してもしなくてもいいです。</p>
例	<pre>macro_command main() char src1[10] = "12.345"</pre>

```

float result1
bool success1
success1 = StringDecAsc2Float(src1[0], result1)
// success1 = true、result1 は “12.345”

float result2
bool success2
success2 = StringDecAsc2Float("1.234567890", result2)
// success2 = true、但し結果はresult2が表現できる範囲を超えていたから、精
確さが低減される可能性がある

char src3[2] = "4b"
float result3
bool success3
success3 = StringDecAsc2Float(src3[0], result3)
// success3 = false、src3には'0'~'9'或いは'.'以外のキャラクタが含まれていたから

end macro_command

```

関数名	StringFloat2DecAsc
シンタック ス	success=StringFloat2DecAsc(source, destination[start])
記述	<p>本関数で浮動小数点数を十進数ストリングに変換します。</p> <p>ソースsourceは定数、または変数のどちらでもいいです。</p> <p>destination[start]は必ず一次元キャラクタ配列変数でなければなりません。ここで変換後の十進数ストリングを保存します。</p> <p>実行が完成されると、bool型戻り値をsuccess欄に返信します。変換に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。</p> <p>変換後の十進数ストリングの長さが目的配列のサイズより大きい場合、関数はfalseを返信します。</p> <p>successに記入してもしなくてもいいです。</p>
例	<pre> macro_command main() float src1 = 1.2345 char dest1[20] bool success1 success1 = StringFloat2DecAsc(src1, dest1[0]) // success1 = true、dest1は “1.2345” float src2 = 1.23456789 char dest2 [20] bool success2 success2 = StringFloat2DecAsc(src2, dest2 [0]) // success2 = true、但し精度が低減される可能性がある float src3 = 1.2345 char dest3[5] bool success3 </pre>

	<pre>success3 = StringFloat2DecAsc(src3, dest3 [0]) // success3 = false、dest3内容が変わらない end macro_command</pre>
--	--

関数名	StringHexAsc2Bin
シンタック ス	<pre>success=StringHexAsc2Bin(source[start], destination) 或いは success=StringHexAsc2Bin("source", destination)</pre>
記述	<p>本関数で十六進数ストリングを整数に変換します。</p> <p>ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えはsource[start])のどちらでもいいです。</p> <p>destinationは必ず変数でなければなりません。ここで変換後の整数値を保存します。</p> <p>実行が完成されると、bool型戻り値をsuccess欄に返信します。変換に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。</p> <p>ソースストリングは十六進数ストリングである必要があり、ストリングに'0'~'9'、または'a'~'f'或いは'A'~'F'以外のキャラクタが含まれていたら、関数はfalseを返信します。</p> <p>successに記入してもしなくてもいいです。</p>
例	<pre>macro_command main() char src1[5]="0x3c" int result1 bool success1 success1 = StringHexAsc2Bin(src1[0], result1) // success1 = true、result1 は 3c short result2 bool success2 success2 = StringDecAsc2Bin("1a2b3c4d", result2) // success2 = true、但し結果はresult2が表現できる範囲を超えていた、result2 = 3c4d char src3[2] = "4g" char result3 bool success3 success3 = StringDecAsc2Bin (src3[0], result3) // success3=false、src3 には'0'~'9'または'a'~'f'或いは'A'~'F'以外のキャラクタが含まれていたから end macro_command</pre>

関数名	StringBin2HexAsc
シンタック ス	success=StringBin2HexAsc(source, destination[start])

記述	<p>本関数で整数を十六進数ストリングに変換します。 ソースsourceは定数、または変数のどちらでもいいです。 destination[start]は必ず一次元キャラクタ配列変数でなければなりません。ここで変換後の十六進数ストリングを保存します。 実行が完成されると、bool型戻り値をsuccess欄に返信します。変換に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。 変換後の十六進数ストリングの長さが目的配列のサイズより大きい場合、関数はfalseを返信します。 successに記入してもしなくてもいいです。</p>
例	<pre>macro_command main() int src1 = 20 char dest1[20] bool success1 success1 = StringBin2HexAsc(src1, dest1[0]) // success1 = true、dest1 は"14" short src2 = 0x3c char dest2[20] bool success2 success2 = StringBin2HexAsc(src2, dest2[0]) // success2 = true、dest2 は"3c" int src3 = 0x1a2b3c4d char dest3[6] bool success3 success3 = StringBin2HexAsc(src3, dest3[0]) // success3 = false、dest3 内容が変わらない end macro_command</pre>

関数名	StringFindOneOf
シンタックス	<code>position=StringFindOneOf(source[start], target[start])</code> <code>position=StringFindOneOf("source", target[start])</code> <code>position=StringFindOneOf(source[start], "target")</code> <code>position=StringFindOneOf("source", "target")</code>
記述	<p>targetストリングの中で任意のキャラクタがsourceストリングで始めて現れた位置を探します。 両ストリングとも、静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 実行が完成されると、targetストリングの任意のキャラクタがsourceストリングで始めて現れた位置を探します。もし見つかれない場合、-1を返信します。</p>
例	<pre>macro_command main() char src1[20] = "abcdeabcede" char target1[20] = "sdf" short pos1 pos1 = StringFindOneOf(src1[0], target1[0])</pre>

```
// pos1 = 3

char src2[20] = "abcdeabcde"
short pos2
pos2 = StringFindOneOf(src2[1], "agi")
// pos2 = 4

char target3 [20] = "bus"
short pos3
pos3 = StringFindOneOf("abcdeabcde", target3[1])
// pos3 = -1

end macro_command
```



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

18.7.7. ストリング処理

関数名	String2Unicode
シンタック ス	result = String2Unicode("source", destination[start])
記述	sourceストリングをUnicodeストリングに変換し、destinationに保存します。変換後のUnicodeストリングの長さはresultに保存されます。Sourceは必ず定数でなければなりません。変数ではありません。
例	<pre>macro_command main() char dest[20] int result result = String2Unicode("abcde", dest[0]) // "result" will be set to 10. result = String2Unicode("abcdefghijklmno", dest[0]) // "result" will be set to 20. // "result" will be the length of converted Unicode string end macro_command</pre>

関数名	StringCat
シンタック ス	success=StringCat(source[start], destination[start]) 或いは success=StringCat("source", destination[start])
記述	本関数でソースストリングを目的ストリングの後ろに接ぎます。本関数の実行に成功した後、目的ストリングは両ストリングが接がれた結果になります。

	<p>ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。</p> <p>destination[start]は必ず一次元キャラクタ配列変数でなければなりません。</p> <p>実行が完成されると、bool型戻り値をsuccess欄に返信します。実行に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。</p> <p>両ストリングが接がれた長さが目的配列の長さを超えた場合、目的ストリングは接ぐ前の内容を保ち、何も変わらずにいます。関数はfalseを返信します。</p> <p>successに記入してもしなくてもいいです。</p>
例	<pre>macro_command main() char src1[20] = "abcdefghijkl" char dest1[20] = "1234567890" bool success1 success1= StringCat(src1[0], dest1[0]) // success1 = true、 dest1 = "123456790abcdefghijkl" char dest2 [10] = "1234567890" bool success2 success2= StringCat("abcde", dest2 [0]) // success2 = false、 dest2 内容が変わらない char src3[20] = "abcdefghijkl" char dest3[20] bool success3 success3 = StringCat(src3[0], dest3[15]) // success3 = false、 dest3 内容が変わらない end macro_command</pre>

関数名	StringCompare
シンタック ス	<pre>ret=StringCompare(str1[start], str2[start]) ret=StringCompare("string1", str2[start]) ret=StringCompare(str1[start], "string2") ret=StringCompare("string1", "string2")</pre>
記述	<p>両ストリングの内容が等しいのかを比較します。大文字と小文字を同一視しません。</p> <p>両ストリングとも、静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。</p> <p>実行が完成されると、bool型戻り値をret欄に返信します。両ストリングが等しかったら、ret欄にtrueとの戻り値を返信し、さもなければfalseを返信します。</p>
例	<pre>macro_command main() char a1[20] = "abcde" char b1[20] = "ABCDE" bool ret1 ret1 = StringCompare(a1[0], b1[0]) // ret1 = false</pre>

```

char a2[20] = "abcde"
char b2[20] = "abcde"
bool ret2
ret2 = StringCompare(a2[0], b2[0])
// ret2 = true

char a3 [20] = "abcde"
char b3[20] = "abcdefg"
bool ret3
ret3 = StringCompare(a3[0], b3[0])
// ret3 = false

end macro_command

```

関数名	StringCompareNoCase
シンタック ス	ret=StringCompareNoCase(str1[start], str2[start]) ret=StringCompareNoCase("string1", str2[start]) ret=StringCompareNoCase(str1[start], "string2") ret=StringCompareNoCase("string1", "string2")
記述	両ストリングの内容が等しいのかを比較します。大文字と小文字を同一視します。 両ストリングとも、静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 実行が完成されると、bool型戻り値をret欄に返信します。両ストリングが等しかったら、ret欄にtrueとの戻り値を返信し、さもなければfalseを返信します。
例	macro_command main() char a1[20] = "abcde" char b1[20] = "ABCDE" bool ret1 ret1 = StringCompareNoCase(a1[0], b1[0]) // ret1 = true char a2[20] = "abcde" char b2[20] = "abcde" bool ret2 ret2 = StringCompareNoCase(a2[0], b2[0]) // ret2 = true char a3 [20] = "abcde" char b3[20] = "abcdefg" bool ret3 ret3 = StringCompareNoCase(a3[0], b3[0]) // ret3 = false end macro_command

関数名	StringCopy
-----	------------

シンタックス	success=StringCopy("source", destination[start]) 或いは success=StringCopy(source[start], destination[start])
記述	<p>本関数を利用してストリングのコピーを行います。本関数は送信された静的ストリング(二重引用符""に囲まれた文字)、或いはキャラクタ配列にいるストリングを目的bufferまでにコピーすることができます。</p> <p>ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列ストリング(例えばsource[start])のどちらでもいいです。</p> <p>destination[start]は必ず一次元キャラクタ配列変数でなければなりません。</p> <p>コピーが完成すると、bool型戻り値をsuccess欄に返信します。コピーに成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。</p> <p>ソースストリングの長さが目的bufferのサイズに等しい場合、何の処理もしなく、falseをsuccessに返信します。</p> <p>successに記入してもしなくてもいいです。</p>
例	<pre>macro_command main() char src1[5] = "abcde" char dest1[5] bool success1 success1 = StringCopy(src1[0], dest1[0]) // success1=true、dest1は "abcde" char dest2[5] bool success2 success2 = StringCopy("12345", dest2[0]) // success2 = true、dest2は"12345" char src3[10] = "abcdefghijkl" char dest3[5] bool success3 success3 = StringCopy(src3[0], dest3[0]) // success3 = false、dest3内容は変わらない char src4[10] = "abcdefghijkl" char dest4[5] bool success4 success4 = StringCopy(src4[5], dest4[0]) // success4=true、dest4は"fg hij" end macro_command</pre>

関数名	StringExcluding
シンタックス	success=StringExcluding(source[start], set[start], destination[start]) success=StringExcluding("source", set[start], destination[start]) success=StringExcluding(source[start], "set", destination[start]) success=StringExcluding("source", "set", destination[start])
記述	sourceストリングでのあるインデックス値0を頭文字(一番目のキャラクタ)にするサブストリングを取り出します。それに、このサブストリングの中にあら全てのキャラクタが必ずsetストリングの中に存在しません。本関数はsourceストリングの一番目のキャラクタから探し、setストリングに存在して

	<p>いるキャラクタを探し出したまで、探し続けます。</p> <p>sourceストリングとsetストリングとも静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。</p> <p>実行が完成されると、bool型戻り値をsuccess欄に返信します。実行に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。取り出したストリングの長さが目的配列のサイズより大きい場合、falseが返信されます。</p>
例	<pre>macro_command main() char src1[20] = "cabbageabc" char set1[20] = "ge" char dest1[20] bool success1 success1 = StringExcluding(src1[0], set1[0], dest1[0]) // success1 = true、 dest1 = "cabba" char src2[20] = "cabbage" char dest2[20] bool success2 success2 = StringExcluding(src2[0], "abc", dest2[0]) // success2 = true、 dest2 = " " char set3[20] = "ge" char dest3[4] bool success3 success3 = StringExcluding("cabbage", set3[0], dest3[0]) // success3 = false、 dest3 内容が変わらない end macro_command</pre>

関数名	StringFind
シンタック ス	position = StringFind (source[start], target[start]) position = StringFind ("source", target[start]) position = StringFind (source[start], "target") position = StringFind ("source", "target")
記述	あるストリング(target)がもう一つのストリング(source)の中で始めて現れた位置を探します。 両ストリングとも、静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 実行が完成されると、targetストリングがsourceストリングで始めて現れた位置を返信します。もしsourceストリングがにあるサブストリングに存在していて、その中に含めたキャラクタ及び並ぶ順がtargetストリングと完全に相当する場合、関数はこのサブストリングの先頭にあるインデックス値を返信し、もし見つかれない場合、-1を返信します。
例	macro_command main() char src1[20] = "abcde" char target1[20] = "cd"

```

short pos1
pos1 = StringFind(src1[0], target1[0])
// pos1 = 2

char target2[20] = "ce"
short pos2
pos2 = StringFind( "abcde", target2[0])
// pos2 = -1

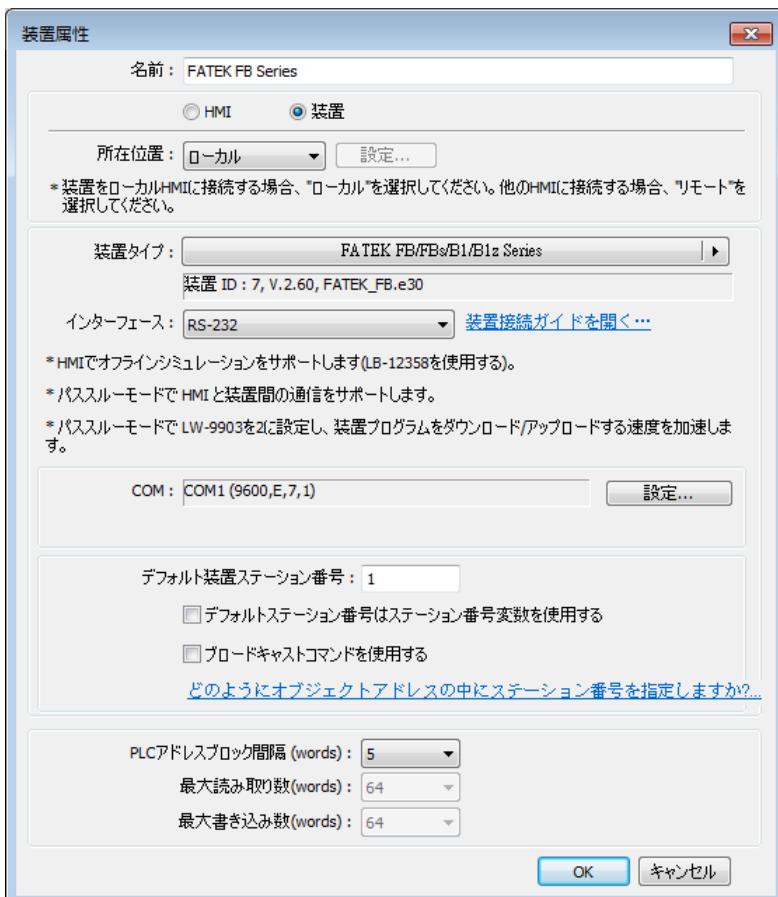
char src3[20] = "abcde"
short pos3
pos3 = StringFind(src3[3], "cd")
// pos3 = -1

end macro_command

```

関数名	StringGet
シンタック ス	StringGet(read_data[start], device_関数名, device_type, address_offset, data_count)
記述	<p>PLC のデータを受け取ります。ストリングのデータフォーマットは一次元キャラクタ配列で、read_data[start]~read_data[start+data_count-1]の一次元配列変数に保存されます。read_data は必ず一次元でなければなりません。</p> <p>data_count で読み取るキャラクタの数を設定します。それは定数、または変数のどちらでもいいです。</p> <p>ここでの device_関数名は“システムパラメータ”で PLC タイプを設定する時、設定した“装置”です。ここでの装置は下図に示す通り “FATEK FB Series”に設定されます。</p>  <p>device_type は LW_BIN だったら、読み取る装置の装置タイプは LW で、データ符号化方式は BIN ということを指しています。BIN 符号化方式を使用すると、“_BIN”を省略してもいいです。</p> <p>もし device_type は LW_BCD だったら、装置タイプは LW で、データの符号化方式は BCD ということを指しています。</p> <p>address_offset とは PLC でのアドレスオフセットです。</p> <p>例えば、StringGet(read_data_1[0], “FATEK FB Series”, RT, 5, 1) とは、対象装置アドレスのオフセットは 5 ということです。</p> <p>もし address_offset は “N#AAAAAA” というフォーマットを使用すると、N は PLC のステーション番号で、AAAAAA はアドレスのオフセットということを指しています。これは一般的に一個の COM ポートに複数の PLC またはコントローラを接続</p>

している状況に用いられます。例えば: StringGet(read_data_1[0], "FATEK FB Series", RT, 2#5, 1)とは、ステーション番号が 2 の PLC 内のデータを読み取るのです。もし StringGet()に“システムパラメータ/装置リスト”で設定したデフォルトステーション番号を使用したら、ここでステーション番号を記入しなくてもいいです。



PLC から読み取るデータ数は、`data_count` の数値を基づいて決定します。
`read_data` 変数は `char` 配列のフォーマットのみ受け入れます。下表をご参照ください：

<code>read_data</code> の型	<code>data_count</code> の数値	読み取る 16 ビットデータの数量
<code>char(8-bit)</code>	1	1
<code>char(8-bit)</code>	2	1

一個の WORD(16 ビット)は 2 個の ASCII キャラクタの長さに相当するので、装置タイプの長さは WORD の場合、上表に基づくと、2 個の ASCII キャラクタを読み取ることは実際、1 個の WORD を読み取ることになります。

例

```
macro_commandmain()
char str1[20]

// LW-0~LW-9との合計10個のWORDを読み取り、変数str1[0]~str1[19]に入れる
// 1WORDは2個のASCIIキャラクタに相当するので、
// を読み取ることは実際、10個のWORDを読み取ることになる
```

	<pre>StringGet(str1[0], "LocalHMI", LW, 0, 20) end macro_command</pre>
--	--

関数名	StringGetEx
シンタック ス	StringGetEx(read_data[start], device_ 関数名, device_type, address_offset, data_count)
記述	PLCのデータを受け取ります。PLCの返信を待たずに次へ実行します。 read_data、device_ 関数名、device_type、address_offset及びdata_countの説明はGetDataと同じです。
例	<pre>macro_commandmain() char str1[20] short test=0 // MODBUS 装置のレスポンスが来ない場合、test=1は継続に実行する StringGetEx(str1[0], "MODBUS RTU", 4x, 0, 20 test=1 // MODBUS 装置のレスポンスが来ない場合、test=2はレスポンスをもらって いない限り、実行されない StringGet(str1[0], "MODBUS RTU", 4x, 0, 20) test=2 end macro_command</pre>

関数名	StringIncluding
シンタック ス	<pre>success=StringIncluding(source[start], set[start], destination[start]) success=StringIncluding("source", set[start], destination[start]) success=StringIncluding(source[start], "set", destination[start]) success=StringIncluding("source", "set", destination[start])</pre>
記述	sourceストリングでのあるインデックス値0を頭文字(一番目のキャラクタ)にするサブストリングを取り出します。それに、このサブストリングの各キャラクタは、全部setストリングの中に見つかることができます。関数はsourceストリングの一番目のキャラクタから探し、setストリングに存在しないキャラクタを探し出したまで、探し続けます。 sourceストリングとsetストリングとも静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 実行が完成されると、bool型戻り値をsuccess欄に返信します。実行に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。 取り出したストリングの長さが目的配列のサイズより大きい場合、falseが返信されます。
例	<pre>macro_command main() char src1[20] = "cabbageabc" char set1[20] = "abc" char dest1[20]</pre>

```

bool success1
success1 = StringIncluding(src1[0], set1[0], dest1[0])
// success1 = true、 dest1 = "cabba"

char src2[20] = "gecabba"
char dest2[20]
bool success2
success2 = StringIncluding(src2[0], "abc", dest2[0])
// success2 = true、 dest2 = " "

char set3[20] = "abc"
char dest3[4]
bool success3
success3 = StringIncluding("cabbage", set3[0], dest3[0])
// success3 = false、 dest3内容が変わらない

end macro_command

```

関数名	StringInsert
シンタック ス	success=StringInsert(pos, insert[start], destination[start]) success=StringInsert(pos, "insert", destination[start]) success=StringInsert(pos, insert[start], length, destination[start]) success=StringInsert(pos, "insert", length, destination[start])
記述	insertストリングを目的ストリングでの特定の位置に挿入します。挿入位置をposで指定します。 Insertストリングは静的ストリング(例えば"insert")、または一次元キャラクタ配列変数(例えばinsert[start])のどちらでもいいです。 ユーザーもlength欄でinsertストリングの長さを指定することができます。 実行が完成されると、bool型戻り値をsuccess欄に返信します。実行に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。 挿入されたストリングの長さが目的配列のサイズより大きい場合、falseが返信されます。
例	macro_command main() char str1[20] = "but the question is" char str2[10] = ", that is" char dest[40] = "to be or not to be" bool success success = StringInsert(18, str1[3], 13, dest[0]) // success = true, dest = "to be or not to be the question" success = StringInsert(18, str2[0], dest[0]) // success=true, dest="to be or not to be, that is the question" success = StringInsert(0, "Hamlet:", dest[0]) // success = false, dest 内容が変わらない

	end macro_command
--	-------------------

関数名	StringLength
シンタック ス	length=StringLength(source[start]) 或いは length=StringLength("source")
記述	ストリングの長さを取ります。 ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 戻り値はソースストリングの長さを代表しています。
例	<pre>macro_command main() char src1[20]="abcde" int length1 length1= StringLength(src1[0]) // length1=5 char src2[20]={'a', 'b', 'c', 'd', 'e'} int length2 length2= StringLength(src2[0]) // length2=5 char src3[20]="abcdefghijkl" int length3 length3= StringLength(src3 [2]) // length3=8 end macro_command</pre>

関数名	StringMD5
シンタック ス	result = StringMD5(source[start], destination[start]) 或いは result = StringMD5("source", destination[start])
記述	本関数を利用し、MD5 message-digest演算法を使用するストリングを生成します。 ソースストリングsourceは静的ストリング(例えば"source")または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。ソースストリングは一次元キャラクタ配列変数である場合、ストリングの開始位置はインデックスの数値で決定されます。 destination[start]は一次元キャラクタ配列変数でなければなりません。ここで取り出したサブストリングを保存します。 実行を完了したら、MD5ストリングの長さがresult欄に返信されます。
例	<pre>macro_command main() char source[32] = "password", dest[32] int result</pre>

```

result = StringMD5(source[0], dest[0])
result = StringMD5("password", dest[0]) // "result" will be set to 32.
// "result" will be the length of MD5 string.
// dest[] = 5f4dcc3b5aa765d61d8327deb882cf99

end macro_command

```

関数名	StringMid
シンタックス	<pre>success = StringMid (source[start], count, destination[start]) 或いは success = StringMid ("string", start, count, destination[start])</pre>
記述	<p>本関数を利用し、一個のストリングでのサブストリングを取り出します。</p> <p>ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。</p> <p>ソースストリングは一次元キャラクタ配列変数である場合、ストリングの開始位置はインデックスの数値で決定されます。ソースストリングは静的ストリングである場合、二番目のパラメータ(start)でストリングの開始位置を決定します。</p> <p>countで取り出すサブストリングの長さを決めます。</p> <p>destination[start]は一次元キャラクタ配列変数でなければなりません。ここで取り出したサブストリングを保存します。</p> <p>実行が完成されると、bool型戻り値をsuccess欄に返信します。実行に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。取り出したサブストリングの長さが目的配列のサイズより大きい場合、関数はfalseを返信します。</p> <p>successに記入してもしなくてもいいです。</p>
例	<pre> macro_command main() char src1[20] = "abcdefghijklmnpqrst" char dest1[20] bool success1 success1 = StringMid(src1[5], 6, dest1[0]) // success1 = true、dest1 は"fghijk" char src2[20] = "abcdefghijklmnpqrst" char dest2[5] bool success2 success2 = StringMid(src2[5], 6, dest2[0]) // success2 = false、dest2 内容が変わらない char dest3[20] = "12345678901234567890" bool success3 success3 = StringMid("abcdefghijklmnpqrst", 5, 5, dest3[15]) // success3 = true、dest3 = "123456789012345fghij" end macro_command </pre>

関数名	StringReverseFind
シンタック ス	<pre>position=StringReverseFind(source[start], target[start]) position=StringReverseFind("source", target[start]) position=StringReverseFind(source[start], "target") position=StringReverseFind("source", "target")</pre>
記述	<p>あるストリング(target)がもう一つのストリング(source)の中で最後に現れた位置を探します。</p> <p>両ストリングとも、静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。</p> <p>実行が完成されると、targetストリングがsourceストリングで最後に現れた位置を返信します。もしsourceストリングにあるサブストリングに存在していて、その中に含めたキャラクタ及び並ぶ順がtargetストリングと完全に相当する場合、関数はこのサブストリングの先頭にあるインデックス値を返信し、もし見つかれない場合、-1を返信します。</p> <p>もしSourceストリングの中に複数のtargetストリングと等しいサブストリングが存在している場合、最後のサブストリングが現れた位置を返信します。</p>
例	<pre>macro_command main() char src1[20] = "abcdeabcde" char target1[20] = "cd" short pos1 pos1 = StringReverseFind(src1[0], target1[0]) // pos1 = 7 char target2[20] = "ce" short pos2 pos2 = StringReverseFind("abcdeabcde" , target2[0]) // pos2 = -1 char src3[20] = "abcdeabcde" short pos3 pos3 = StringReverseFind(src3[6], "ab") // pos3 = -1 end macro_command</pre>

関数名	StringSet
シンタック ス	<pre>StringSet(send_data[start], device_ 関数名, device_type, address_offset, data_count)</pre>
記述	<p>数値を PLC に書き込みます。ストリングデータは send_data[start]~send_data[start+data_count-1]に保存され、send_data は一次元キャラクタ配列のフォーマットであること。</p> <p>data_count は PLC に書き込むキャラクタの数であり、定数または変数のどちらでもいいです。</p> <p>device_ 関数名は上記の説明をご参照ください。</p>

`device_type`は装置タイプとPLCのデータの符号化方式です。例：もし`device_type`はLW_BINだったら、読み取る装置の装置タイプはLWで、データ符号化方式はBINということを指しています。BIN符号化方式を使用すると、"_BIN"を省略してもいいです。

もし`device_type`はLW_BCDだったら、装置タイプはLWで、データの符号化方式はBCDということを指しています。

`address_offset`とはPLCでのアドレスオフセットです。

例えば、`StringSet(read_data_1[0], "FATEK FB Series", RT, 5, 1)`とは、読み取る装置アドレスのオフセットは5ということです。

もし`address_offset`は"N#AAAAAA"というフォーマットを使用すると、NはPLCのステーション番号で、AAAAAAはアドレスのオフセットということを指しています。これは一般的に一個のCOMポートに複数のPLCまたはコントローラを接続している状況に用いられます。例えば：`StringSet(send_data_1[0], "FATEK FB Series", RT, 2#5, 1)`とは、ステーション番号が2のPLC内のデータを読み取るのです。もし`GetData()`に"システムパラメータ/装置リスト"で設定したデフォルトステーション番号を使用したら、ここでステーション番号を記入しなくてもいいです。PLCに送信するデータ数は、`data_count`の数値に基づいて決定します。`send_data`変数はchar配列のフォーマットのみ受け入れます。下表をご参照ください：

<code>seed_data</code> の型	<code>data_count</code> の数値	送信する 16 ビットデータの数量
char(8-bit)	1	1
char(8-bit)	2	1

一個のWORD(16ビット)は2個のASCIIキャラクタの長さに相当するので、装置タイプの長さはWORDの場合、上表に基づくと、2個のASCIIキャラクタを読み取ることは実際、1個のWORDを読み取ることになります。マクロはまず`lowbyte`を書き込み、そして`hightbyte`に書く込むような順番でASCIIキャラクタを書き込みます。

[アスキー表示]オブジェクトで数値を表示する場合、正確に表示するため`data_count`に2の倍数を書き込む必要があります。例えば：

```
macro_command main()
char src1[10]="abcde"
StringSet(src1[0], "LocalHMI", LW, 0, 5)
end macro_command
```

[アスキー表示]オブジェクトは下記のように表示されます：



abcd

`data_count`はストリングの長さより大なりイコールの偶数である場合、ストリングがまるごとに表示されます：

	<pre>macro_command main() char src1[10] = "abcde" StringSet(src1[0], "LocalHMI", LW, 0, 6) end macro_command</pre> <div style="text-align: center; border: 1px solid black; padding: 5px; margin-top: 10px;">abcde</div>
例	<pre>macro_command main() char str1[20] = "abcde" // str1で三個のWORDをLW-0~LW-2に書き込む // data_countは10であっても、もし3番目のWORDに書き込む際にストリング // が終わっていたら、 // その後ろのデータを書き込まない StringSet(str1[0], "LocalHMI", LW, 0, 10) end macro_command</pre>

関数名	StringSetEx
シンタックス	StringSetEx(send_data[start], device_関数名, device_type, address_offset, data_count)
記述	データをPLCに書き込みます。PLCの返信を待たずに次へ実行します。 send_data、device_関数名、device_type、address_offset及びdata_countの説明はStringSetと同じです。
例	<pre>macro_command main() char str1[20] = "abcde" short test=0 // MODBUS 装置のレスポンスが来なかったら、test=1は継続に実行する StringSetEx(str1[0], "MODBUS RTU", 4x, 0, 20) test=1 // MODBUS 装置のレスポンスが来なかったら、レスポンスがもらったまで、 // test=2は実行されない StringSet(str1[0], "MODBUS RTU", 4x, 0, 20) test=2 end macro_command</pre>

関数名	StringToUpper
シンタックス	success=StringToUpper(source[start], destination[start]) success=StringToUpper("source", destination[start])
記述	sourceストリングのキャラクタを全部大文字に変換し、destinationストリングに書き込みます。

	<p><code>source</code>ストリングと<code>set</code>ストリングとも静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えば<code>source[start]</code>)のどちらでもいいです。実行が完成されると、<code>bool</code>型戻り値を<code>success</code>欄に返信します。実行に成功したら、<code>success</code>欄に<code>true</code>との戻り値を返信し、さもなければ<code>false</code>を返信します。取り出したストリングの長さが目的配列のサイズより大きい場合、<code>false</code>が返信されます。</p>
例	<pre>macro_command main() char src1[20] = "aBcDe" char dest1[20] bool success1 success1 = StringToUpper(src1[0], dest1[0]) // success1 = true、 dest1 = "ABCDE" char dest2[4] bool success2 success2 = StringToUpper("aBcDe", dest2[0]) // success2 = false、 dest2 内容が変わらない end macro_command</pre>

関数名	StringToLower
シンタック ス	<code>success=StringToLower(source[start], destination[start])</code> <code>success=StringToLower("source", destination[start])</code>
記述	<p><code>source</code>ストリングのキャラクタを全部小文字に変換し、<code>destination</code>ストリングに書き込みます。</p> <p><code>source</code>ストリングと<code>set</code>ストリングとも静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えば<code>source[start]</code>)のどちらでもいいです。実行が完成されると、<code>bool</code>型戻り値を<code>success</code>欄に返信します。実行に成功したら、<code>success</code>欄に<code>true</code>との戻り値を返信し、さもなければ<code>false</code>を返信します。取り出したストリングの長さが目的配列のサイズより大きい場合、<code>false</code>が返信されます。</p>
例	<pre>macro_command main() char src1[20] = "aBcDe" char dest1[20] bool success1 success1 = StringToLower(src1[0], dest1[0]) // success1 = true、 dest1 = "abcde" char dest2[4] bool success2 success2 = StringToLower ("aBcDe", dest2[0]) // success2 = false、 dest2 内容が変わらない end macro_command</pre>

関数名	StringToReverse
シンタック ス	success=StringToReverse(source[start], destination[start]) success=StringToReverse("source", destination[start])
記述	sourceストリングを逆さにし、destinationストリングに書き込みます。 sourceストリングは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 実行が完成されると、bool型戻り値をsuccess欄に返信します。実行に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。 取り出したストリングの長さが目的配列のサイズより大きい場合、falseが返信されます。
例	<pre>macro_command main() char src1[20] = "abcde" char dest1[20] bool success1 success1 = StringToReverse(src1[0], dest1[0]) // success1 = true、dest1 = "edcba" char dest2[4] bool success2 success2 = StringToReverse ("abcde", dest2[0]) // success2 = false、dest2 内容が変わらない end macro_command</pre>

関数名	StringTrimLeft
シンタック ス	success=StringTrimLeft(source[start], set[start], destination[start]) success=StringTrimLeft("source", set[start], destination[start]) success=StringTrimLeft(source[start], "set", destination[start]) success=StringTrimLeft("source", "set", destination[start])
記述	sourceストリングの一番目のキャラクタから後ろへ探し、setストリングと同じキャラクタがあったら、そのキャラクタをカットし、setストリングに存在しないキャラクタを探し出したまで、探し続けます。 sourceストリングは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 実行が完成されると、bool型戻り値をsuccess欄に返信します。実行に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。 カットされたストリングの長さが目的配列のサイズより大きい場合、falseが返信されます。
例	<pre>macro_command main() char src1[20] = "# *a*#bc" char set1[20] = "# *" char dest1[20] bool success1 success1 = StringTrimLeft (src1[0], set1[0], dest1[0]) // success1 = true、dest1 = "a*#bc"</pre>

	<pre> char set2[20] = {'#', ' ', '*'} char dest2[4] bool success2 success2 = StringTrimLeft ("# *a*#bc", set2[0], dest2[0]) // success2 = false、dest2 内容が変わらない char src3[20] = "abc *#" char dest3[20] bool success3 success3 = StringTrimLeft (src3[0], "# *", dest3[0]) // success3 = true、dest3 = "abc *#" end macro_command </pre>
--	--

関数名	StringTrimRight
シンタックス	<pre> success=StringTrimRight(source[start], set[start], destination[start]) success=StringTrimRight("source", set[start], destination[start]) success=StringTrimRight(source[start], "set", destination[start]) success=StringTrimRight("source", "set", destination[start]) </pre>
記述	<p>sourceストリングでの最後のキャラクタから前へ探し、setストリングと同じキャラクタがあったら、そのキャラクタをカットし、setストリングに存在しないキャラクタを探し出したまで、探し続けます。</p> <p>sourceストリングは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。</p> <p>実行が完成されると、bool型戻り値をsuccess欄に返信します。実行に成功したら、success欄にtrueとの戻り値を返信し、さもなければfalseを返信します。</p> <p>カットされたストリングの長さが目的配列のサイズより大きい場合、falseが返信されます。</p>
例	<pre> macro_command main() char src1[20] = "# *a*#bc# * " char set1[20] = "# *" char dest1[20] bool success1 success1 = StringTrimRight(src1[0], set1[0], dest1[0]) // success1 = true、dest1= "# *a*#bc" char set2[20] = {'#', ' ', '*'} char dest2[20] bool success2 success2 = StringTrimRight("# *a*#bc", set2[0], dest2[0]) // success2 = true、dest2 = "# *a*#bc" char src3[20] = "ab**c *#" char dest3[4] bool success3 success3 = StringTrimRight(src3[0], "# *", dest3[0]) // success3 = false、dest3 内容が変わらない </pre>

	end macro_command
--	-------------------

関数名	Unicode2Utf8
シンタック ス	result = Unicode2Utf8(source[start], destination[start])
記述	sourceのUnicodeストリングをUTF8ストリングに変換し、destinationに保存します。 実行が完成されると、bool型戻り値をresult欄に送信します。実行に成功したら、result = true、さもなければ、= falseです。
例	<pre>macro_command main() char unicode_str[20] char utf8_str[20] String2Unicode("ABC", unicode_str[0]) bool result result = Unicode2Utf8(unicode_str[0], utf8_str[0]) // "result" will be set to true. "utf8_str" will equal "ABC" encoded in UTF8 StringCat("DEF", utf8_str[0]) // "utf8_str" will equal "ABCDEF" encoded in UTF8 char dst[20] bool result2 result2 = Utf82Unicode(utf8_str[0], dst[0]) // "result" will be set to true. "dst" will equal "ABCDEF" encoded in Unicode. end macro_command</pre>

関数名	UnicodeCat
シンタック ス	result = UnicodeCat(source[start], destination[start]) 或いは result = UnicodeCat("source", destination[start])
記述	本関数でソースストリングを目的ストリングの後ろに接ぎます。本関数の実行に成功した後、目的ストリングは両ストリングが接がれた結果になります。ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。 destination[start]は一次元キャラクタ配列変数でなければなりません。 実行が完成されると、bool型戻り値をresult欄に返信します。実行に成功したら、result = true、さもなければ、= falseです。 両ストリングを接いだ後の合計長さが目的配列のサイズより大きい場合、目的ストリングは接がれた前の内容のままを保ち、falseを返信します。
例	macro_command main()

	<pre> char strSrc[12] = "αθβγθδ" char strDest[28] = "ζηθλ1234" bool result result = UnicodeCat(strSrc[0], strDest[0]) // "result" will be set to true // "strDest" will be set to "ζηθλ1234αθβγθδ" result = UnicodeCat("ζηθλ", strDest[0]) // the function fails. // "result" will be set to false due to insufficient destination buffer size. // In this case, the content of "strDest" remains the same. end macro_command </pre>
--	---

関数名	UnicodeCompare
シンタックス	<pre> result = UnicodeCompare(string1[start], string2[start]) result = UnicodeCompare("string1", string2[start]) result = UnicodeCompare(string1[start], "string2") result = UnicodeCompare("string1", "string2") </pre>
記述	<p>両ストリングの内容が等しいかどうかを比較します。大文字小文字を区別します。</p> <p>両ストリングとも静的ストリング(例えば"string")或いは一次元配列変数(例えばstring[start])のどちらでもいいです。</p> <p>実行が完成されると、bool型戻り値をresult欄に返信します。両ストリングが等しかったら、result = true、さもなければ、= falseです。</p>
例	<pre> macro_command main() char str1[10] = "θαβθγ" char str2[8] = "αβγδ" bool result result = UnicodeCompare(str1[0], str2[0]) // "result" will be set to false. result = UnicodeCompare(str1[0], "θαβθγ") // "result" will be set to true. end macro_command </pre>

関数名	UnicodeCopy
シンタックス	<pre> result = UnicodeCopy("source", destination[start]) 或いは result = UnicodeCopy(source[start], destination) </pre>
記述	<p>本関数でストリングをコピーします。本関数は送信された静的ストリング(二重引用符" "に囲まれたストリング)またはキャラクタ配列に保存されたストリングを目的配列にコピーすることができます。</p> <p>ソースストリングsourceは静的ストリング(例えば"source")、または一次元キャラクタ配列変数(例えばsource[start])のどちらでもいいです。</p> <p>destination[start]は一次元キャラクタ配列変数でなければなりません。</p> <p>コピーが完成されると、bool型戻り値をresult欄に送信します。コピーに成功</p>

	<p>したら、result = true、さもなければ、 = falseです。ソースストリングの長さが目的配列のサイズより大きい場合、何の処理をしなく、falseをresult欄に送信します。</p> <p>resultに記入してもしなくてもいいです。</p>
例	<pre>macro_command main() char strSrc[14] = "αβθγδθε" //αβθγδθε char strDest[14] bool result result = UnicodeCopy(strSrc[0], strDest[0]) // "result" will be set to true. result = UnicodeCopy("αβθγδθε", strDest[0]) // "result" will be set to true, strDest = αβθγδθε" result = UnicodeCopy("αβγδεζαβγδεζ", strDest[0]) // "result" will be set to false. // The size of source string exceeds the size of destination string. end macro_command</pre>

関数名	UnicodeExcluding
シンタック ス	<pre>result = UnicodeExcluding(source[start], set[start], destination[start]) result = UnicodeExcluding("source", set[start], destination[start]) result = UnicodeExcluding(source[start], "set", destination[start]) result = UnicodeExcluding("source", "set", destination[start])</pre>
記述	<p>sourceストリング内のある頭文字(一番目のキャラクタ)が0であるサブストリングを取得します。このストリングにはsetストリング内にあるキャラクタが含まれていません。本関数はsourceストリングの一番目のキャラクタから探し始め、setストリング内にあるキャラクタを探し出す前まで、探し続けます。sourceストリングとsetストリングとも静的ストリング(例えば"source")または一次元配列変数(例えばsource[start])のどちらでもいいです。</p> <p>実行が完成されると、bool型戻り値をresult欄に返信します。実行に成功したら、result = true、さもなければ、 = falseです。取得したストリングの長さが目的配列のサイズより大きい場合、falseが返信されます。</p>
例	<pre>macro_command main() char source[14] = "γδξκθλθ, dest[8] char set[4] = "λθ" bool result result = UnicodeExcluding(source[0], set[0], dest[0]) // the function succeeds. // "result" will be set to true and "dest" will be set to "γδξκ". result = UnicodeExcluding(source[0], set[0], dest[4]) // the function fails. // "result" will be set to false due to insufficient destination buffer size. end macro_command</pre>

関数名	Utf82Unicode
-----	--------------

シンタックス	<code>result = Utf82Unicode(source[start], destination[start])</code>
記述	sourceのUTF8ストリングをUnicodeストリングに変換し、destinationに保存します。 実行が完成されると、bool型戻り値をresult欄に返信します。実行に成功したら、 <code>result = true</code> 、さもなければ、 <code>= false</code> です。
例	<pre>macro_command main() char unicode_str[20] char utf8_str[20] String2Unicode("ABC", unicode_str[0]) bool result result = Unicode2Utf8(unicode_str[0], utf8_str[0]) // "result" will be set to true. "utf8_str" will equal "ABC" encoded in UTF8 StringCat("DEF", utf8_str[0]) // "utf8_str" will equal "ABCDEF" encoded in UTF8 char dst[20] bool result2 result2 = Utf82Unicode(utf8_str[0], dst[0]) // "result" will be set to true. "dst" will equal "ABCDEF" encoded in Unicode. end macro_command</pre>



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

18.7.8. 数学

関数名	SQRT
シンタックス	<code>SQRT(source, result)</code>
記述	平方根を計算します。データソース source は定数、或いは変数のどちらでもいいが、結果を示す result は必ず変数でなければなりません。データソースは一つの正数でなければなりません。
例	<pre>macro_command main() float source, result SQRT(15, result) source = 9.0</pre>

	SQRT(source, result)// 実行後 result = 3.0 end macro_command
--	--

関数名	CUBERT
シンタック ス	CUBERT(source, result)
記述	立方根を計算します。データソース source は定数、或いは変数のどちらでもいいが、結果を示する result は必ず変数でなければなりません。データソースは一つの正数でなければなりません。
例	macro_command main() float source, result CUBERT (27, result) // 実行後 result = 3.0 source = 27.0 CUBERT (source, result) // 実行後 result = 3.0 end macro_command

関数名	POW
シンタック ス	POW(source1, source2, result)
記述	source1 の幂乗(source2)を計算します。データソース source1 と source2 は定数、或いは変数のどちらでもいいが、結果を示する result は必ず変数でなければなりません。データソースは一つの正数でなければなりません。
例	macro_command main() float y, result y = 0.5 POW (25, y, result) // 実行後 result = 5 end macro_command

関数名	SIN
シンタック ス	SIN(source, result)
記述	三角関数の正弦を計算します。データソース source は定数、或いは変数のどちらでもいいが、結果を示する result は必ず変数でなければなりません。
例	macro_command main() float source, result SIN(90, result)// result is 1 source=30 SIN(source, result)// result is 0.5

	<code>endmacro_command</code>
関数名	COS
シンタック ス	<code>COS(source, result)</code>
記述	三角関数の余弦を計算します。データソース <code>source</code> は定数、或いは変数のどちらでもいいが、結果を示す <code>result</code> は必ず変数でなければなりません。
例	<pre>macro_command main() float source, result COS(90, result) // result is 0 source = 60 COS(source, result) // result is 0.5 end macro_command</pre>

関数名	TAN
シンタック ス	<code>TAN(source, result)</code>
記述	三角関数の正接を計算します。データソース <code>source</code> は定数、或いは変数のどちらでもいいが、結果を示す <code>result</code> は必ず変数でなければなりません。
例	<pre>macro_command main() float source, result TAN(45, result) // result is 1 source = 60 TAN(source, result) // result is 1.732 end macro_command</pre>

関数名	COT
シンタック ス	<code>COT(source, result)</code>
記述	三角関数の余接を計算します。データソース <code>source</code> は定数、或いは変数のどちらでもいいが、結果を示す <code>result</code> は必ず変数でなければなりません。
例	<pre>macro_command main() float source, result COT(45, result) // result is 1 source = 60 COT(source, result) // result is 0.5774</pre>

	<code>end macro_command</code>
--	--------------------------------

関数名	SEC
シンタック ス	<code>SEC(source, result)</code>
記述	三角関数の正割を計算します。データソース <code>source</code> は定数、或いは変数のどちらでもいいが、結果を示す <code>result</code> は必ず変数でなければなりません。
例	<pre>macro_command main() float source, result SEC(45, result) // result is 1.414 source = 60 SEC(source, result) // if source is 60, result is 2 end macro_command</pre>

関数名稱	CSC
シンタック ス	<code>CSC(source, result)</code>
記述	三角関数の余割を計算します。データソース <code>source</code> は定数、或いは変数のどちらでもいいが、結果を示す <code>result</code> は必ず変数でなければなりません。
例	<pre>macro_command main() float source, result CSC(45, result) // result is 1.414 source = 30 CSC(source, result) // result is 2 end macro_command</pre>

関数名	ASIN
シンタック ス	<code>ASIN(source, result)</code>
記述	逆三角関数の逆正弦を計算します。データソース <code>source</code> は定数、或いは変数のどちらでもいいが、結果を示す <code>result</code> は必ず変数でなければなりません。
例	<pre>macro_command main() float source, result ASIN(0.8660, result) // result is 60 source = 0.5 ASIN(source, result) // result is 30 end macro_command</pre>

関数名	ACOS
シンタック ス	ACOS(source, result)
記述	逆三角関数の逆余弦を計算します。データソース source は定数、或いは変数のどちらでもいいが、結果を示す result は必ず変数でなければなりません。
例	<pre>macro_command main() float source, result ACOS(0.8660, result) // result is 30 source = 0.5 ACOS(source, result) // result is 60 end macro_command</pre>

関数名	ATAN
シンタック ス	ATAN(source, result)
記述	逆三角関数の逆正割を計算します。データソース source は定数、或いは変数のどちらでもいいが、結果を示す result は必ず変数でなければなりません。
例	<pre>macro_command main() float source, result ATAN(1, result) // result is 45 source = 1.732 ATAN(source, result) // result is 60 end macro_command</pre>

関数名	LOG
シンタック ス	LOG(source, result)
記述	自然対数を計算し、result に入れます。 データソース source は定数、或いは変数のどちらでもいいが、結果を示す result は必ず変数でなければなりません。
例	<pre>macro_command main() float source=100, result LOG (source, result) // result 約イコール 4.6052 end macro_command</pre>

関数名	LOG10
シンタック ス	LOG10(source, result)

ス	
記述	10 から起算する対数を計算し、resultに入れます。 データソース source は定数、或いは変数のどちらでもいいが、結果を示す result は必ず変数でなければなりません。
例	macro_command main() float source=100, result LOG10 (source, result) // result イコール 2 end macro_command

関数名	RAND
シンタック ス	RAND(result)
記述	乱数を発生させます。(範囲: 0 ~ 32767) 結果を示す result は必ず変数でなければなりません。
例	macro_command main() short result RAND (result) // result is not a fixed value when executes macro every time end macro_command

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

18.7.9. 統計

関数名	AVERAGE
シンタック ス	AVERAGE(source[start], result, count)
記述	配列から平均値を計算します。
例	int data[5] = {1, 2, 3, 4, 5} float result AVERAGE(data[0], result, 5) // result イコール 3 AVERAGE(data[2], result, 3) // result イコール 4

関数名	HARMEAN
シンタック ス	HARMEAN(source[start], result, count)
記述	配列から調和平均値を計算します。
例	int data[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

	float result HARMEAN(data[0], result, 10) // resultイコール3.414
--	---

関数名	MAX
シンタック ス	MAX(source[start], result, count)
記述	配列から最大値を取得します。
例	<pre>int data[5] = {1, 2, 3, 4, 5} int result</pre> <pre>MAX(data[0], result, 5) // resultイコール5 MAX(data[1], result, 3) // resultイコール4</pre>

関数名	MEDIAN
シンタック ス	MEDIAN(source[start], result, count)
記述	配列から中央値を取得します。
例	<pre>int data[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} float result</pre> <pre>MEDIAN(data[0], result, 10) // resultイコール5.5</pre>

関数名	MIN
シンタック ス	MIN(source[start], result, count)
記述	配列から最小値を取得します。
例	<pre>int data[5] = {1, 2, 3, 4, 5} int result</pre> <pre>MIN(data[0], result, 5) // resultイコール1 MIN(data[1], result, 3) // resultイコール2</pre>

関数名	STDEVP
シンタック ス	STDEVP(source[start], result, count)
記述	配列から標準差を計算します。
例	<pre>int data[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} float result</pre> <pre>STDEVP(data[0], result, 10) // resultイコール2.872</pre>

関数名	STDEVS
シンタック ス	STDEVS(source[start], result, count)
記述	配列からサンプル標準差を計算します。
例	<pre>int data[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} float result STDEVS(data[0], result, 10) // result 等於 3.027</pre>

18.7.10. レシピデータベース

関数名	RecipeGetData
シンタック ス	RecipeGetData(destination, recipe_address, record_ID)
記述	レシピ内のデータを取ります。取ったデータをdestinationに保存し、そのデータは必ず変数でなければなりません。recipe_addressはレシピ名と項目名で組み合わせたのです："recipe_name.item_name"。record_IDでレシピ内の何件目のレコードを取るのかを指定します。即ちレコードのID番号を指定します。
例	<pre>macro_command main() int data = 0 char str[20] int recordID bool result recordID = 0 result = RecipeGetData(data, "TypeA.item_weight", recordID) // レシピ"TypeA"の中で第0レコード、またitem nameは"item_weight"のデータを取る recordID = 1 result = RecipeGetData(str[0], "TypeB.item_name", recordID) // レシピ"TypeB"の中で第1レコード、またitem nameは"item_name"のデータを取る end macro_command</pre>

関数名	RecipeQuery
シンタック ス	RecipeQuery(SQL command, destination)
記述	SQLステートメントを通じ、レシピ内のデータを調べます。調べた結果のデータレコード数をdestinationに保存し、その数値は必ず変数でなければなりません。SQL commandは静的文字、またはキャラクタ配列のどちらでもいいです。例えば：

	<p>RecipeQuery("SELECT*FROMTypeA", destination) 或いは RecipeQuery(sql[0], destination) SQLステートメントは"SELECT*FROM"を先頭にし、後にレシピ名及び検索条件を付きます。</p>
例	<pre>macro_command main() int total_row = 0 char sql[100] = "SELECT * FROM TypeB" short var bool result result = RecipeQuery("SELECT * FROM TypeA", total_row) // レシピ"TypeA"を調べる。調べたデータレコード数をtotal_rowに保存する result = RecipeQuery(sql[0], total_row) // レシピ"TypeB"を調べる。調べたデータレコード数をtotal_rowに保存する result = RecipeQuery("SELECT * FROM Recipe WHERE Item >%(var)", total_row) // レシピ "Recipe"内の"Item"の数値がvarより大きいデータレコード数を調べる。調べたデータレコード数をtotal_rowに保存する end macro_command</pre>

関数名	RecipeQueryGetData
シンタックス	RecipeQueryGetData(destination, recipe_address, result_row_no)
記述	RecipeQueryの検索結果を取ります。本関数を呼び出す前には、先にRecipeQueryを呼び出すのが必須で、またrecipe_addressではRecipeQueryと同じのレシピ名を指定する必要があります。 result_row_noで検索結果での何件目のレコードを取るのかを指定します。
例	<pre>macro_command main() int data = 0 int total_row = 0 int row_number = 0 bool result_query bool result_data result_query = RecipeQuery("SELECT * FROM TypeA", total_row) // レシピ"TypeA"を調べる。調べたデータレコード数をtotal_rowに保存する if (result_query) then for row_number = 0 to total_row - 1 result_data = RecipeQueryGetData(data, "TypeA.item_weight", row_number) next row_number end if</pre>

	end macro_command
関数名	RecipeQueryGetRecordID
シンタック ス	RecipeQueryGetRecordID(destination, result_row_no)
記述	RecipeQuery検索結果のレコードID番号を取ります。本関数を呼び出す前には、先にRecipeQueryを呼び出す必要があります。 result_row_noで検索結果での何件目のレコードのID番号を取るのかを指定し、レコードID番号をdestinationに書き込みます。
例	<pre>macro_command main() int recordID = 0 int total_row = 0 int row_number = 0 bool result_query bool result_id result_query = RecipeQuery("SELECT * FROM TypeA", total_row) // レシピ"TypeA"を調べる。検索結果のレコード数をtotal_rowに保存する if(result_query) then for row_number = 0 to total_row - 1 result_id = RecipeQueryGetRecordID(recordID, row_number) next row_number end if end macro_command</pre>

関数名	RecipeSetData
シンタック ス	RecipeSetData(source, recipe_address, record_ID)
記述	データをレシピデータベースに書き込みます。実行に成功したら、trueとの戻り値を返信し、さもなければfalseを返信します。 recipe_addressはレシピ名と項目名で組み合わせたのです： "recipe_name.item_name"。 record_IDでレシピでの何件目のレコードを修正するのかを指定します。即ちレコードのID番号を指定します。
例	<pre>macro_command main() int data=99 char str[20]="abc" int recordID bool result recordID = 0</pre>

```

result = RecipeSetData(data, "TypeA.item_weight", recordID)
// set data to recipe "TypeA", where item name is "item_weight" and the record ID is
0.

recordID = 1
result = RecipeSetData(str[0], "TypeB.item_name", recordID)
// set data to recipe "TypeB", where item name is "item_name" and the record ID is 1.

end macro_command

```



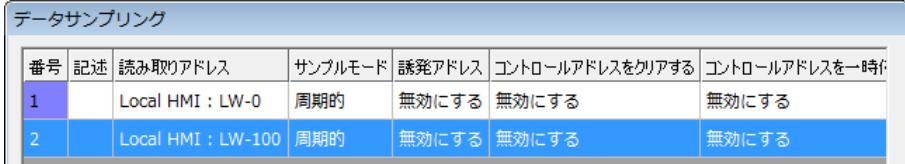
このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

18.7.11. データサンプリング/イベントログ

関数名	FindDataSamplingDate																					
シンタック ス	return_value=FindDataSamplingDate(data_log_number, index, year, month, day) or FindDataSamplingDate(data_log_number, index, year, month, day)																					
記述	<p>本関数でデータサンプリングファイルの日付を調べます。入力したサンプリング番号(data_log_number)とデータサンプリングファイルインデックス(index)により、当データのサンプリング日付を調べることができます、年、月、日の順でyear、month、dayの欄に書き込みます。</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p>データサンプリング</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>番号</th> <th>記述</th> <th>読み取りアドレス</th> <th>サンプルモード</th> <th>誘発アドレス</th> <th>コントロールアドレスをクリアする</th> <th>コントロールアドレスを一時停止する</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Local HMI : LW-0</td> <td>周期的</td> <td>無効にする</td> <td>無効にする</td> <td>無効にする</td> <td>無効にする</td> </tr> <tr> <td>2</td> <td>Local HMI : LW-100</td> <td>周期的</td> <td>無効にする</td> <td>無効にする</td> <td>無効にする</td> <td>無効にする</td> </tr> </tbody> </table> </div> <p>データサンプリングファイルの保存先は:[保存位置]¥[サンプリングフォルダ名]¥yyyyymmdd.dtlです。データサンプリングファイルインデックス(index)とは、サンプリングフォルダ内の全てのデータサンプリングファイルを、ファイル名順で配列した順位値(0から)です。Index値が小さいほど、日付が新しいです。例として、仮にあるサンプリングフォルダの中では、四個のデータサンプリングファイルがある場合：</p> <p>20101210.dtl 20101230.dtl 20110110.dtl 20110111.dtl</p> <p>Indexの順は：</p> <p>20101210.dtl ->indexは3 20101230.dtl ->indexは2 20110110.dtl ->indexは1 20110111.dtl ->indexは0</p>	番号	記述	読み取りアドレス	サンプルモード	誘発アドレス	コントロールアドレスをクリアする	コントロールアドレスを一時停止する	1	Local HMI : LW-0	周期的	無効にする	無効にする	無効にする	無効にする	2	Local HMI : LW-100	周期的	無効にする	無効にする	無効にする	無効にする
番号	記述	読み取りアドレス	サンプルモード	誘発アドレス	コントロールアドレスをクリアする	コントロールアドレスを一時停止する																
1	Local HMI : LW-0	周期的	無効にする	無効にする	無効にする	無効にする																
2	Local HMI : LW-100	周期的	無効にする	無効にする	無効にする	無効にする																

	<p>検索したいデータサンプリングファイルを探し出すことに成功した場合、1との戻り値をreturn_valueに返信し、さもなければ0を返信します。</p> <p>data_log_numberとindexは定数、または変数のどちらでもいいが、year、month、dayとreturn_valueは必ず変数でなければなりません。</p> <p>return_valueはオプションです。</p>
例	<pre>macro_command main() short data_log_number=1, index=2, year, month, day short success // 仮にデータサンプリングファイル : 20101230.dtlがある場合、そのデータサンプリング番号は1で、ファイルインデックスは2 // success==1、year==2010、month==12、day==30 success=FindDataSamplingDate(data_log_number, index, year, month, day) endmacro_command</pre>

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

関数名	FindDataSamplingIndex
シンタック ス	<pre>return_value=FindDataSamplingIndex(data_log_number, year, month, day, index) or FindDataSamplingIndex(data_log_number, year, month, day, index)</pre>
記述	<p>本関数でデータサンプリングファイルのデータインデックス値を調べます。入力したデータサンプリング番号(data_log_number)と日付により、データサンプリングファイルのデータインデックスを調べることができます。そして結果をindexに書き込みます。year、month、dayとの三つの欄は順で年、月、日を代表し、入力フォーマットはYYYY(年)、MM(月)、DD(日)です。</p> <p></p> <p>データサンプリングファイルの保存先は:[保存位置]¥[サンプリングフォルダ名]¥yyyymmdd.dtlです。データサンプリングファイルインデックス(index)とは、サンプリングフォルダ内の全てのデータサンプリングファイルを、ファイル名順で配列した順位値(0から)です。Index値が小さいほど、日付が新しいです。例として、仮にあるサンプリングフォルダの中では四個のデータサンプリングファイルがある場合：</p> <p>20101210.dtl 20101230.dtl 20110110.dtl 20110111.dtl</p>

	<p>indexの順は :</p> <p>20101210.dtl ->indexは3 20101230.dtl ->indexは2 20110110.dtl ->indexは1 20110111.dtl ->indexは0</p> <p>検索したいデータサンプリングファイルを探し出すことに成功した場合、1との戻り値をreturn_valueに返信し、さもなければ0を返信します。</p> <p>data_log_numberとindexは定数、または変数のどちらでもいいが、year、month、dayとreturn_valueは必ず変数でなければなりません。</p> <p>return_valueはオプションです。</p>
例	<pre>macro_command main() short data_log_number = 1, year = 2010, month = 12, day = 10, index short success // 仮にデータサンプリングファイル : 20101230.dtlがある場合、そのデータサンプリング番号は1で、ファイルインデックスは2 // success==1, index==2 success=FindDataSamplingIndex(data_log_number, year, month, day, index) end macro_command</pre>



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

関数名	FindEventLogDate
シンタック ス	<p>return_value=FindEventLogDate(index, year, month, day)</p> <p>or</p> <p>FindEventLogDate(index, year, month, day)</p>
記述	<p>本関数でイベントログファイルの日付を調べます。入力したイベントログファイルインデックス(index)で当イベントの日付を調べることができます、そして年、月、日の順でyear、month、dayの欄に書き込みます。</p> <p>イベントログファイルインデックス(index)とは、指定の保存位置(HMI、SDカードまたはUSBディスク)内のイベントログファイルを、ファイル名順で配列した順位値(0から)です。Index値が小さいほど、日付が新しいです。例として、仮にあるサンプリングフォルダの中では四個のデータサンプリングファイルがある場合 :</p> <p>EL_20101210.evt EL_20101230.evt EL_20110110.evt EL_20110111.evt</p> <p>Indexの順は :</p> <p>EL_20101210.evt ->indexは3</p>

	<p>EL_20101230.evt ->indexは2 EL_20110110.evt ->indexは1 EL_20110111.dtl ->indexは0 return_valueはオプションです。 検索したいイベントログファイルを探し出すことに成功した場合、1との戻り値をreturn_valueに返信し、さもなければ0を返信します。 Indexは定数または変数のどちらでもいいですが、year、month、dayとreturn_valueは必ず変数でなければなりません。 return_valueはオプションです。</p>
例	<pre>macro_commandmain() short index=1, year, month, day short success // もし一個のイベントログファイル : EL_20101230.evtがあり、インデックスは1の場合 // uccess==1, year==2010, month==12, day==30 success=FindEventLogDate(index, year, month, day) end macro_command</pre>

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

関数名	FindEventLogIndex
シンタックス	return_value=FindEventLogIndex(year, month, day, index) or FindEventLogIndex(year, month, day, index)
記述	<p>本関数でイベントログファイルのファイルインデックス値を調べます。入力した日付で当イベントログのファイルインデックスを調べることができ、そしてindexに書き込みます。year、month、dayとの三つの欄は順で年、月、日を代表し、入力フォーマットはYYYY(年)、MM(月)、DD(日)です。</p> <p>イベントログファイルインデックス(index)とは、指定の保存位置(HMI、SD カードまたは USB ディスク)内のイベントログファイルを、ファイル名順で配列した順位値(0 から)です。Index 値が小さいほど、日付が新しいです。例として、仮にある四個のイベントログファイルがある場合：</p> <p>EL_20101210.evt EL_20101230.evt EL_20110110.evt EL_20110111.evt</p> <p>indexの順は：</p> <p>EL_20101210.evt ->indexは3 EL_20101230.evt ->indexは2</p>

	<p><code>EL_20110110.evt ->indexは1</code> <code>EL_20110111.dtl ->indexは0</code></p> <p>検索したいイベントログファイルを探し出すことに成功した場合、1との戻り値を<code>return_value</code>に返信し、さもなければ0を返信します。</p> <p><code>year</code>、<code>month</code>、<code>day</code>は定数または変数のどちらでもいいですが、<code>index</code>と<code>return_value</code>は必ず変数でなければなりません。</p> <p><code>return_value</code>はオプションです。</p>
例	<pre>macro_commandmain() short year=2010, month=12, day=10, index short success // もし一個のイベントログファイル : EL_20101210.evtがあり、インデックス // は2の場合 // success==1、index==2 success=FindEventLogIndex(year, month, day, index) end macro_command</pre>

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

18.7.12. パリティ

関数名	ADDSUM
シンタックス	<code>ADDSUM(source[start], result, data_count)</code>
記述	<code>source[start]</code> から <code>source[start+data-count-1]</code> までの全ての一次元のデータを積算し、 <code>checksum(チェックサム)</code> を算出します。それにその結果を <code>result</code> 変数の中に入れます。 <code>result</code> は必ず変数でなければなりません。 <code>data_count</code> は積算するデータの数で、定数または変数のどちらでもいいです。
例	<pre>macro_command main() char data[5] short checksum data[0] = 0x1 data[1] = 0x2 data[2] = 0x3 data[3] = 0x4 data[4] = 0x5 ADDSUM(data[0], checksum, 5) // checksum is 0xf end macro_command</pre>

関数名	XORSUM
シンタック ス	XORSUM(source[start], result, data_count)
記述	source[start]から source[start+data-count-1]までの全ての一次元のデータを XOR 演算をし、checksum(チェックサム)を算出します。それにその結果を result 変数の中に入れます。 result は必ず変数でなければなりません。data_count は XOR 演算するデータの数で、定数または変数のどちらでもいいです。
例	<pre>macro_command main() char data[5] = {0x1, 0x2, 0x3, 0x4, 0x5} short checksum XORSUM(data[0], checksum, 5) // checksum is 0x1 end macro_command</pre>

関数名	BCC
シンタック ス	BCC(source[start], result, data_count)
記述	XORSUM と同じです。
例	<pre>macro_command main() char data[5] = {0x1, 0x2, 0x3, 0x4, 0x5} char checksum BCC(source[0], checksum, 5) // checksum is 0x1 end macro_command</pre>

関数名	CRC
シンタック ス	CRC(source[start], result, data_count)
記述	source[start]から source[start+data-count-1]までの全ての一次元配列でのデータを 16-bit CRC 演算をし、checksum(チェックサム)を算出します。それにその結果を result 変数の中に入れます。 Result 必ず変数でなければなりません。data_count は演算するデータの数で、定数または変数のどちらでもいいです。
例	<pre>macro_command main() char data[5] = {0x1, 0x2, 0x3, 0x4, 0x5} short checksum CRC(data[0], checksum, 5)// checksum is 0xbb2a, 16-bit CRC end macro_command</pre>

関数名	CRC8
-----	------

シンタックス	CRC8(source[start], result, data_count)
記述	source[start]から source[start+data-count-1]までの全ての一次元配列でのデータを 8-bit CRC 演算をし、checksum(checksum)を算出します。それにその結果を result 変数の中に入れます。 Result 必ず変数でなければなりません。data_count は演算するデータの数で、定数または変数のどちらでもいいです。
例	<pre>macro_command main() char source[5] = {1, 2, 3, 4, 5} short CRC8_result CRC8(source[0], CRC8_result, 5) // CRC8_result = 188 end macro_command</pre>

18.7.13. 他の関数

関数名	Beep
シンタックス	Beep()
記述	システム警告音を出します。 本関数は800ヘルツ、30ミリ秒のシステム警告音(ビーツ)を出します。
例	<pre>macro_command main() Beep() end macro_command</pre>

関数名稱	Buzzer
シンタック ス	Buzzer()
記述	ブザーを鳴らす/停止します。
例	<pre>macro_command main() char on = 1, off = 0 Buzzer(on) // turn on the buzzer DELAY(1000)// delay 1 second Buzzer(off) // turn off the buzzer DELAY(500) // delay 500ms Buzzer(1)// turn on the buzzer DELAY(1000)// delay 1 second Buzzer(0)// turn off the buzzer end macro_command</pre>

関数名	TRACE
シンタック ス	TRACE(format, argument)
記述	<p>実行中のマクロコマンドは本関数を使って、変数の変化を監視することができ、それにデバッグのため、ストリングをプリントアウトできます。ユーザーはEasyDiagnoser/cMT Diagnoserを通じて、本関数の出力結果が見られます。</p> <p>TRACE関数は頭文字が%の特殊キャラクタを取る途端、同時にargumentから一個のパラメータを取り、フォーマットしてから出力します。</p> <p>Formatとはプリントのフォーマットで、頭文字が%の特殊キャラクタをサポートします。特殊なキャラクタフォーマットは以下のとおりです。その中、角括弧内の欄はオプションで、赤文字の欄は必須です：</p> <p>%[flags][width][.precision]type</p> <p>各欄の意味は下記に述べた通りです：</p> <p>flags(オプション) :</p> <ul style="list-style-type: none"> - : 左に揃えます。幅を指定し、左側で不足の部分を空白キャラクタで補います。 + : プラス・マイナス記号を出力します。 <p>width(オプション) :</p> <p>十進数正の整数であり、キャラクタの長さを指定し、不足の部分を空白キャラクタで補います。</p> <p>precision(オプション) :</p> <p>十進数正の整数であり、精確さ、及び出力キャラクタ数を指定します。</p> <p>type :</p> <p>Cあるいはc : キャラクタで出力します。</p> <p>d : signed十進数整数で出力します。</p> <p>i : signed十進数整数で出力します。</p> <p>o : unsigned八進数整数で出力します。</p> <p>u : unsigned十進数整数で出力します。</p> <p>Xあるいはx : unsigned十六進数整数で出力します。</p> <p>f : 出力フォーマットは[-]dddd.ddddです。その中、欄dddは一個または複数の十進数数字です。</p> <p>lld : signed長整数(64bit)で出力します。(cMTにのみ対応)</p> <p>llu : unsigned長整数(64bit)で出力します。(cMTにのみ対応)</p> <p>f : 単精度浮動小数点数で出力します。</p> <p>lf : 倍精度浮動小数点数で出力します。</p> <p>Eあるいはe : 出力フォーマットは[-]d.ddde [sign]dddです。その中、欄dは十進数数字、欄dddは一個または複数の十進数数字、欄dddは三個の十進数数字である必要があります。Signは+または-です。</p> <p>Formatストリングは最大256個キャラクタをサポートし、超えたキャラクタは省略されます。</p> <p>Argument部分は書いても書かなくてもいいです。一個の特殊キャラクタは一</p>

	個の変数に対応する必要があります。
例	<pre>macro_command main() char c1 = 'a' short s1 = 32767 float f1 = 1.234567 TRACE("The results are") // 出力 : The results are TRACE("c1 = %c, s1 = %d, f1 = %f", c1, s1, f1) // 出力 : c1 = a, s1 = 32767, f1 = 1.234567 end macro_command</pre>

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

関数名	GetCnvTagArrayIndex
シンタック ス	GetCnvTagArrayIndex(array_index)
記述	<p>ユーザ一定義タグに[読み取り変換]サブルーチンを使用し、それに配列である場合、マクロサブルーチンで本関数を使用すれば、配列のインデックスを取得できます。</p> <p>データサンプリングに[読み取り変換]を使用する時、配列インデックスの数値を処理してからサンプリングすることに用いられます。</p>
例	<pre>Sub short newfun(short param) Int index GetCnvTagArrayIndex(index) //もしindexの数値は2の場合、[読み取り変換]配列内の3番目のデータを使用することを示す。 return param end sub</pre>

18.8. マクロの作成、及び実行の方法について

18.8.1. マクロの作成方法について

下記手順に従って、マクロを作成します。

1. [プロジェクト]上のマクロのアイコンをクリックし、マクロ管理ウィンドウをオープンします。

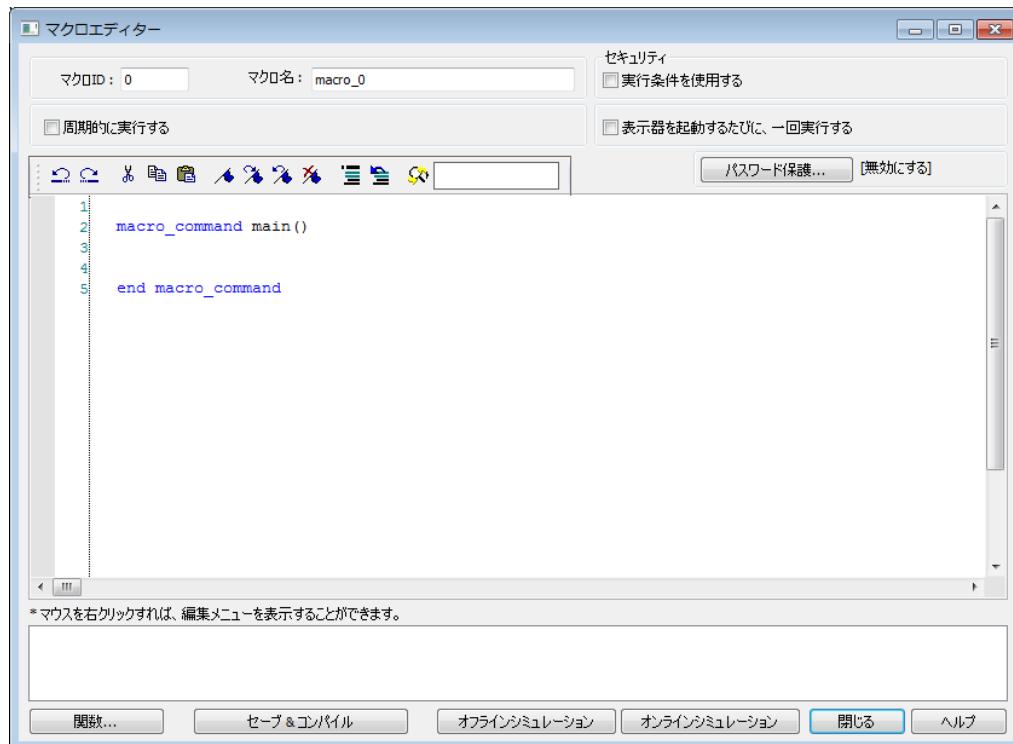


マクロ管理ダイアログボックスの中に、コンパイルに成功したマクロは[マクロリスト]に表示され、コンパイルが未完成のマクロは[未完成マクロ]リストに表示されます。以下はマクロ管理ダイアログボックスでの各ボタンの機能記述です。

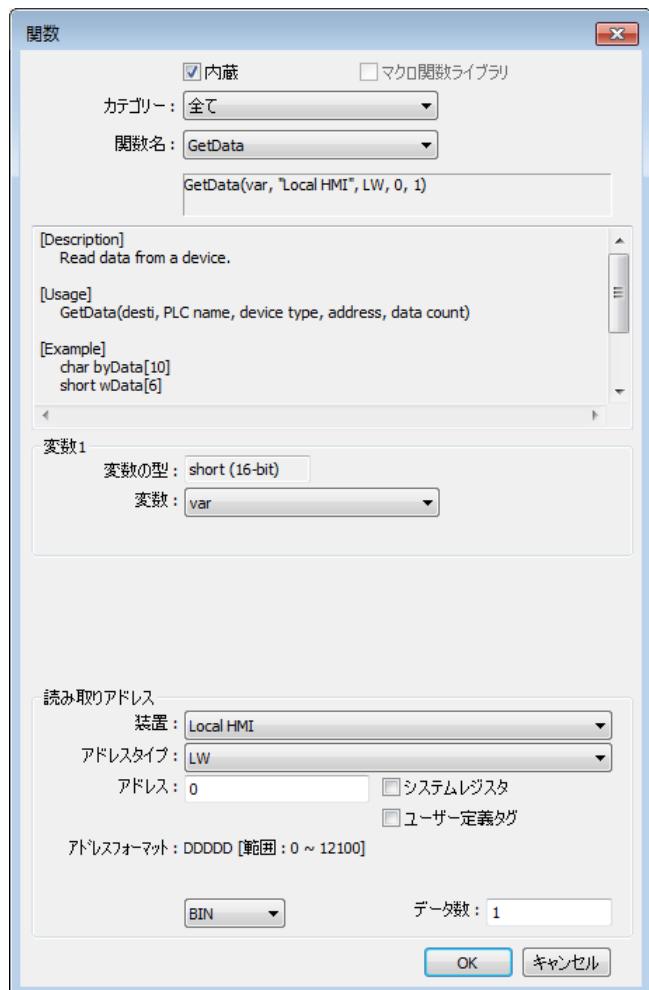
設定	記述
新規作成	マクロを新規作成し、“ワークスペース”エディターをオープンします。
削除	選択したマクロを削除します
編集	“ワークスペース”エディターをオープンし、選択したマクロをオープンします。
コピー	選択したマクロをコピーします。
貼り付け	コピーしたマクロを“マクロリスト”に貼り付け、新規マクロ名を生じます。
エクスポート	選択したマクロを*.ebm ファイルに保存します。

ト	
インポート	*.ebm ファイルをプロジェクトにインポートします。
OK	今回が編集したマクロを確認した後、マクロ管理ダイアログボックスを離れます。本ボタンを押してから、今回の編集内容が保存されます。
キャンセル	今回が編集した内容を全部キャンセルし、マクロ管理ダイアログボックスを離れます。
マクロ関数 ライブラリ	マクロ関数ライブラリ管理ダイアログをオープンします。

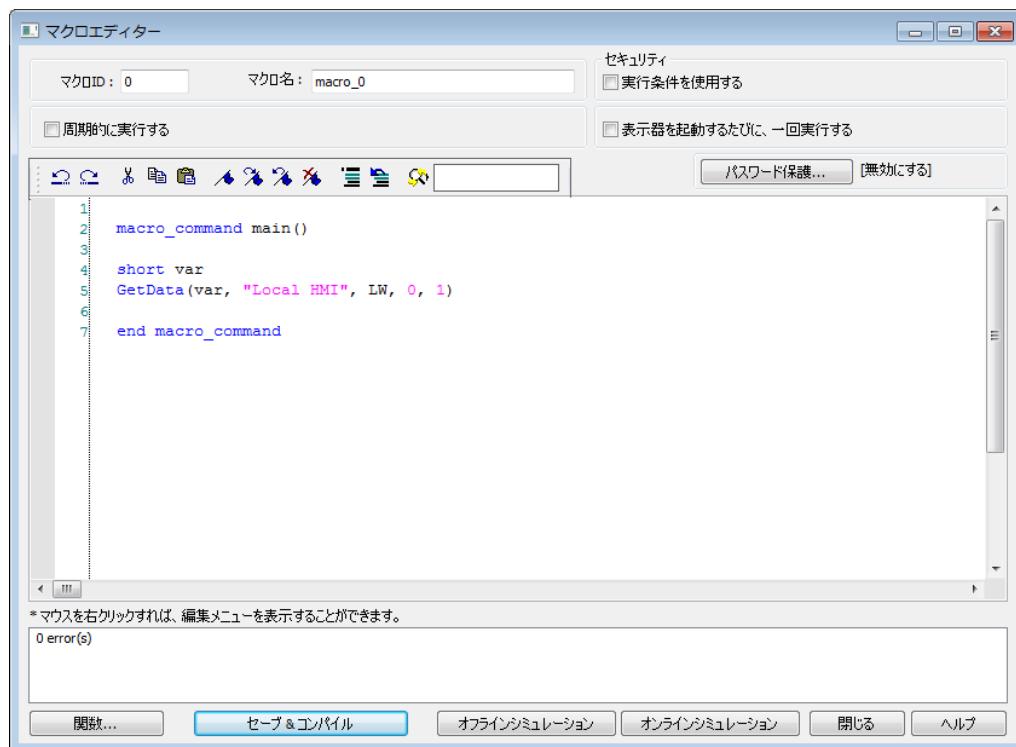
2. [新規作成]ボタンを押し、新規“ワークスペース”エディターをオープンします。各マクロでも唯一の番号があり、[マクロ ID]という位置に定義されます。[マクロ名]という欄でもマクロ名に入力する必要があります、そうでなければコンパイルは成功できません。



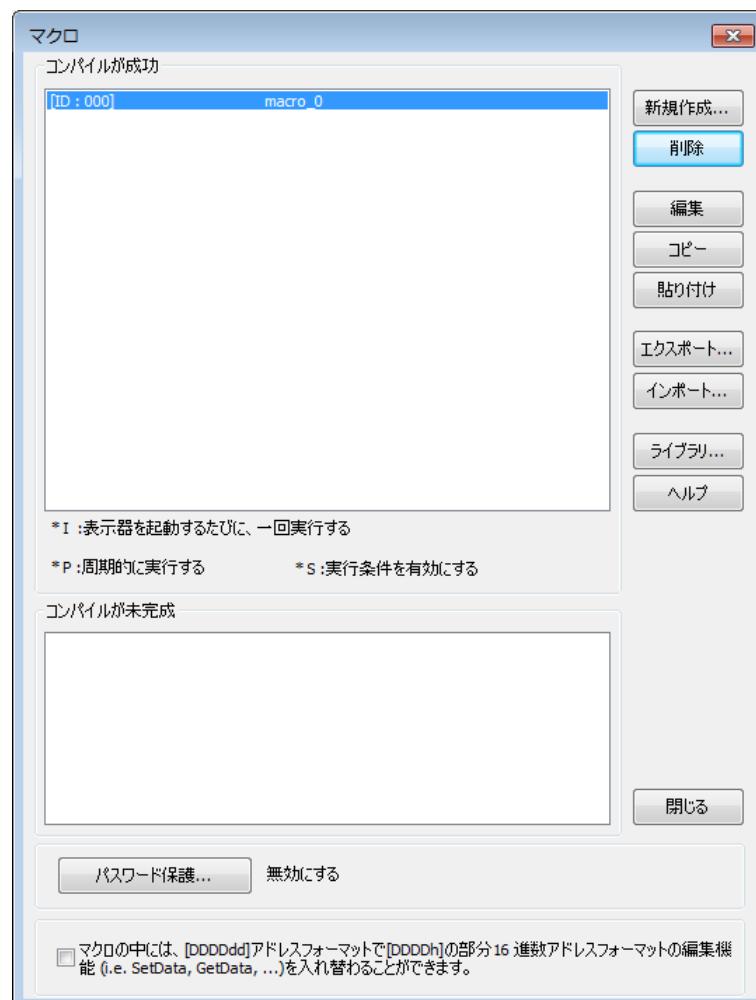
3. カスタマイズのマクロをデザインします。内蔵の関数を使用したい場合、例えば Setdata() 或いは Getdata()など、[関数]ボタンにワンクリックしてダイアログボックスをオープンし、所要する関数を選択、また必要なパラメータを設定します。



- 新規マクロを編集完了後、[コンパイル]ボタンにワンクリックし、コンパイルを行います。



5. エラーがなかったら、[閉じる]ボタンをワンクリックすると、[マクロリスト]には“macro_0”というマクロが新規作成されたと見られます。



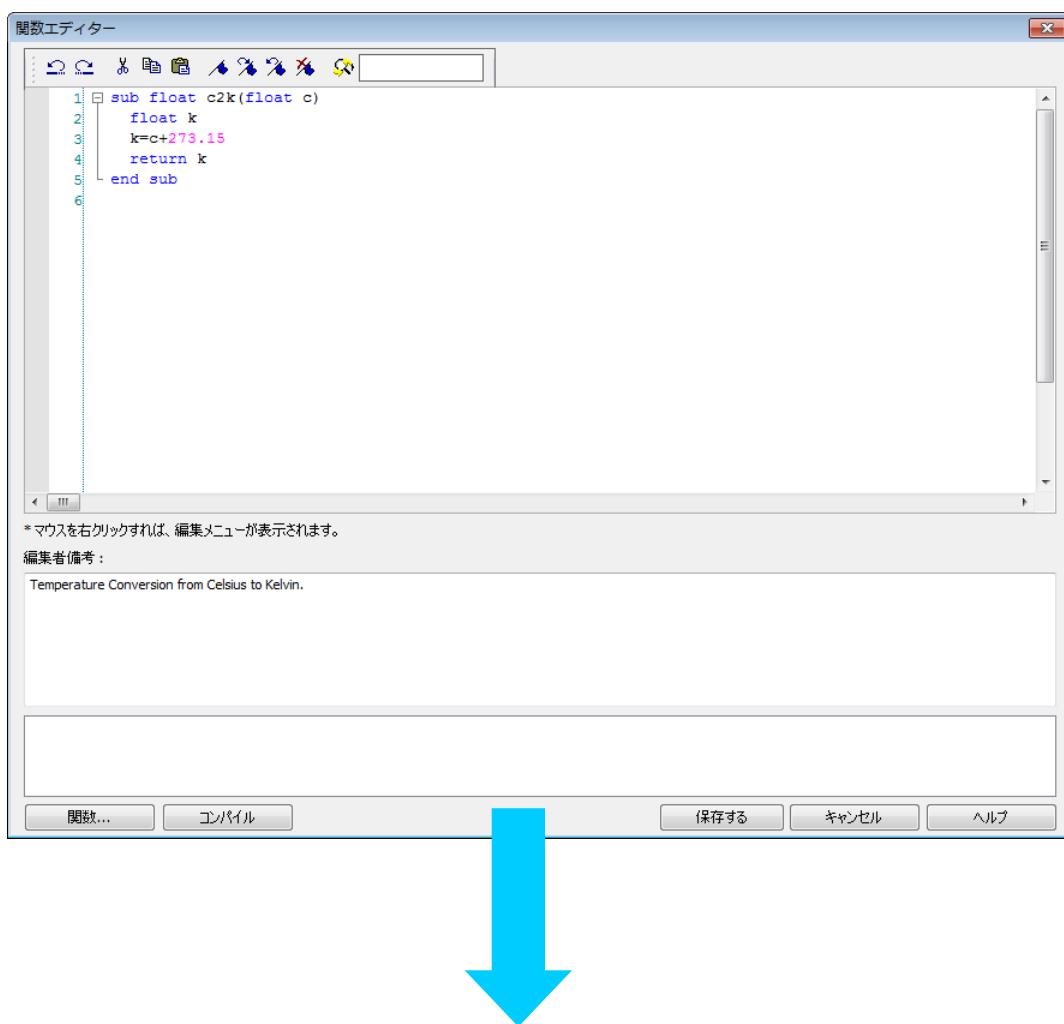
18.8.2. マクロを実行する

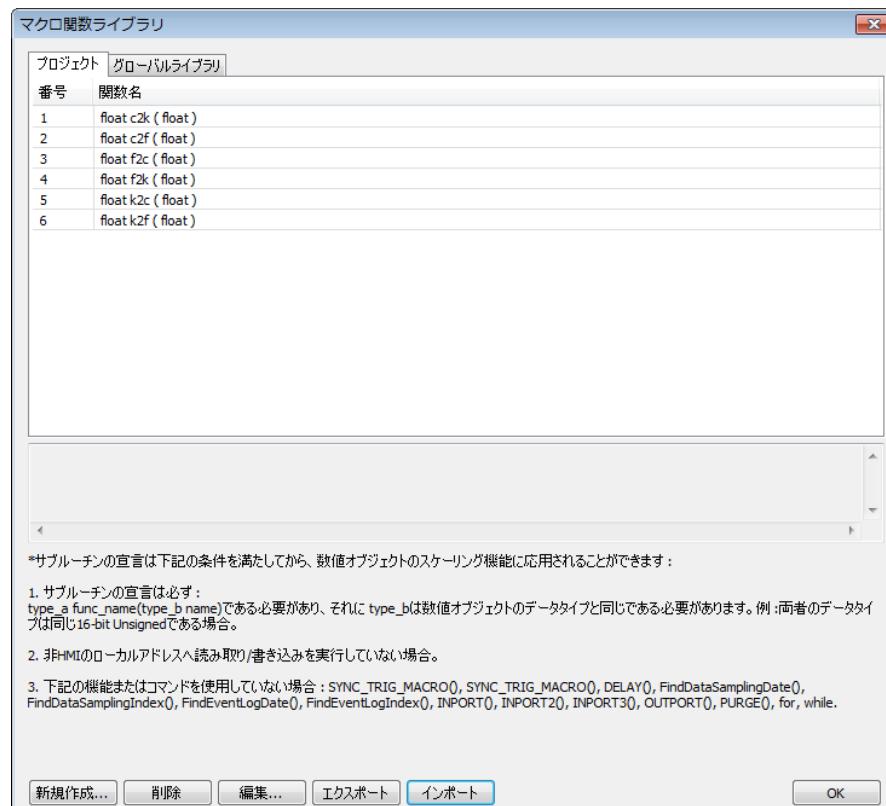
マクロを実行するには、多くの方法があり、下記にて個別に説明します。

- [PLC コントロール]オブジェクトを使用する
 - 1. [PLC コントロール]オブジェクトをオープンし、属性を[マクロを実行する]に設定します。
 - 2. 実行したいマクロに、一個のビットをマクロのトリガー用に選択し、そしてマクロのトリガー条件を選択します。条件が満足された場合、そのマクロは重複に実行されます。毎回で一回のみ実行したい場合、デザインする時には、当トリガービットが復帰するように設定する必要があります。
 - 3. [ビット状態設定]オブジェクトまたは[トグルスイッチ]オブジェクトをこのビットのコントロールスイッチにします。
-
- [ビット状態設定]オブジェクトまたは[トグルスイッチ]オブジェクトを使用する
 - 1. [ビット状態設定]オブジェクトまたは[トグルスイッチ]オブジェクトの一般的な属性タブで、[マクロを実行する]にチェックマークを入れます。
 - 2. 実行したいマクロを選択します。このオブジェクトが実行されるたびに、選択したマクロが一回実行されます。
-
- [ファンクションキー]を使用する
 - 1. [ファンクションキー]の一般的な属性タブで、[マクロをトリガーする]にチェックマークを入れます。
 - 2. 実行したいマクロを選択します。このオブジェクトが実行されるたびに、選択したマクロが一回実行されます。
-
- [マクロエディター]で条件を設定する
 - 1. [周期的に実行する] : 何秒の間隔で自動的にマクロを実行します。
 - 2. [HMI を起動するたびに、一回実行する] : HMI を再起動するたびに、マクロを一回実行します。
-
- [ウインドウ設定]でマクロを実行する時点を設定する
 - 1. [オープン時に実行] : 本ウインドウをオープンするたびに、指定したマクロを一回実行します。
 - 2. [循環的に実行] : 本ウインドウをオープンするたびに、0.5 秒ごとに循環的に指定したマクロを実行します。
 - 3. [閉じる時に実行] : このウインドウを閉じると、指定したマクロを一回実行します。
- ▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

18.9. ユーザー定義関数機能

マクロエディターを使用する時に、関数を定義する手間を減らすように、ユーザーは内蔵の[マクロ関数ライブラリ]の中で所要する関数を探すことができます。しかしながら、ユーザーがマクロを編集する際に、常に使用される関数がマクロ関数ライブラリから見つからない場合、自分で関数を定義して保存することができます。次回、同じ関数を使用したかったら、[マクロ関数ライブラリ]から保存した関数を呼び出すことができます。また、[マクロ関数ライブラリ]もユーザー定義関数の可搬性を大幅に向上しました。関数を作成する前、内蔵のマクロ関数ライブラリまたはオンライン関数ライブラリには使用できる関数があるのかを検索できます。

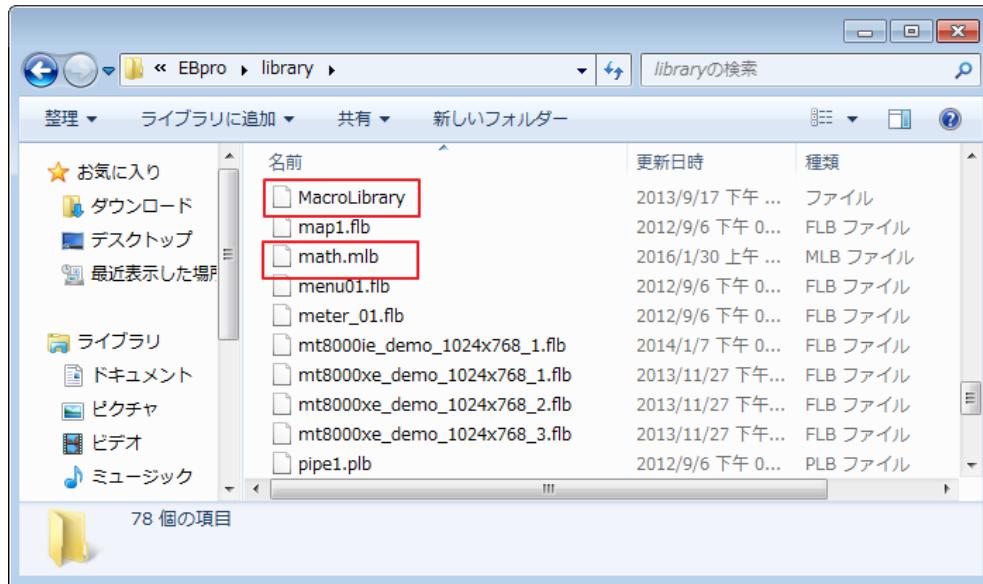




18.9.1. 関数ライブラリファイルをインポートする

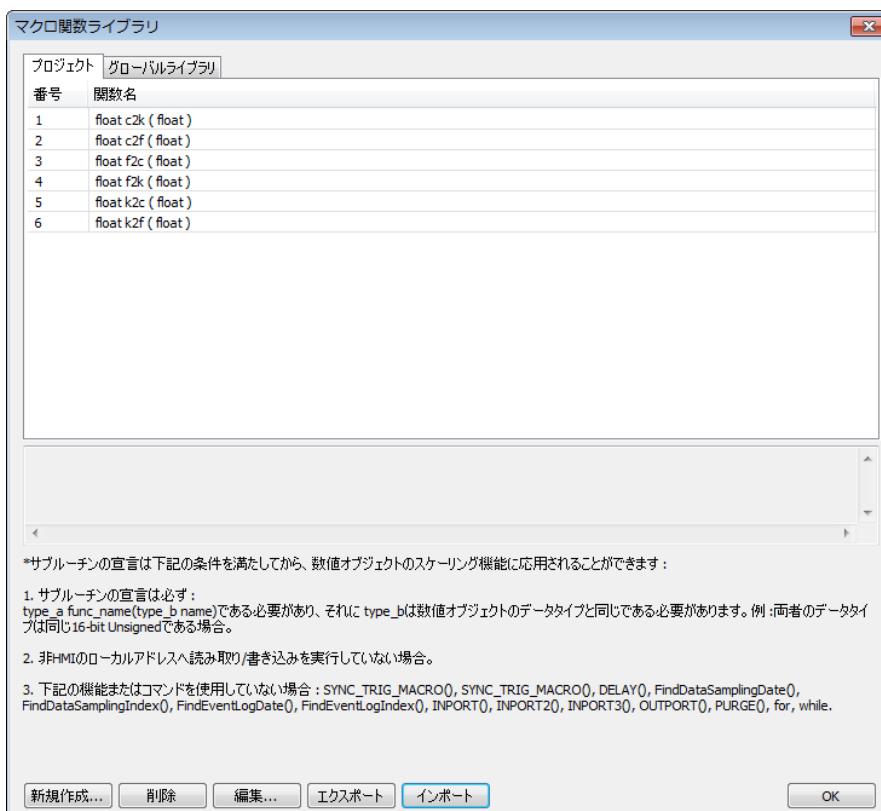
編集ソフトウェアがプロジェクトをオープンする時には、自動的にデフォルトの関数ライブラリファイルを読み取り、関数の情報をロードします。もしプロジェクトでユーザー定義関数を呼び出す場合、まず関連の.mlb ファイルをロードする必要があります。

1. デフォルト関数ライブラリファイル名 : MacroLibrary(拡張子がない)
2. 関数ライブラリパス : 編集エディターインストールディレクトリの¥library フォルダの下。
3. ¥library フォルダの下では二種類の関数ライブラリファイルが見られます :
 - 拡張子がない : MacroLibrary、デフォルト関数ライブラリであり、編集ソフトウェアがこのファイルを読み取るのを指定します。
 - 拡張子がある(.mlb) : 例えば math.mlb は、ユーザーがインポート/エクスポートする際に読み取る/書き込むファイルであり、可搬性があるので、使用したい場合になったらフォルダから呼び出せばいいです。
4. EasyBuilder Pro をオープンする時に、ただデフォルト関数ライブラリでの関数しかロードしないので、ユーザーが.mlb ファイルの関数を使用したいなら、自分でインポートする必要があります。

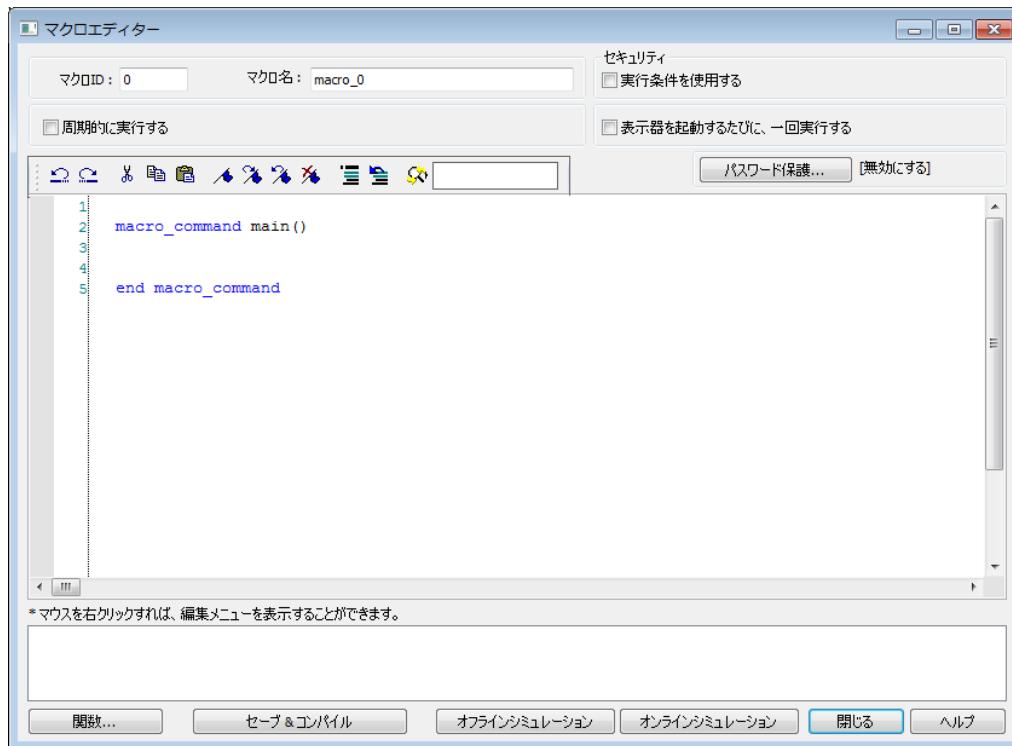


18.9.2. マクロ関数ライブラリの利用方法

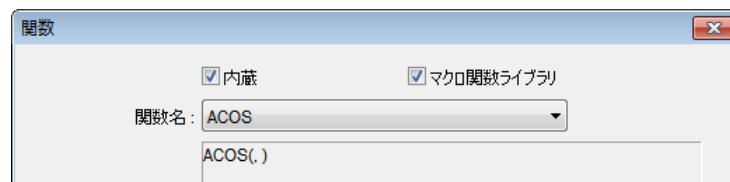
- マクロエディターの中に直接関数を呼び出します。



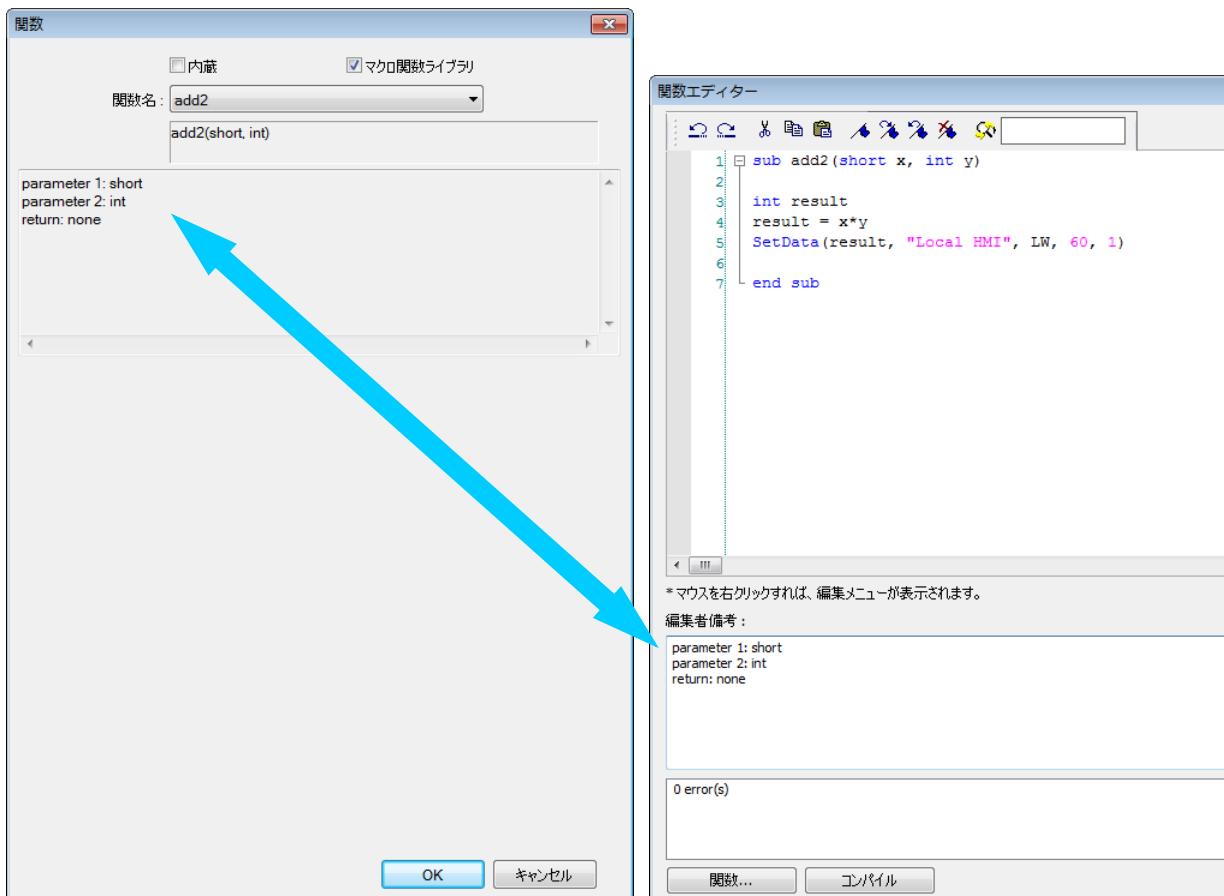
- マクロエディターの左下隅にある[関数]ボタンを押し、関数ダイアログボックスをオープンします。



3. [マクロ関数ライブラリ]或いは[内蔵]の中に少なくとも一個のチェックマークを入れ、使用したい関数を選択します。



4. 関数記述が表示されます。即ち、ユーザーが関数エディターで編集した記述です。



5. 使用したい関数を選択し、関数ライブラリ内に[変数型]を、指定した変数名に変更します。

<pre> 1 2 macro_command main() 3 4 short a 5 int b,result 6 7 add2(short, int) 8 9 end macro_command </pre>	<pre> 1 2 macro_command main() 3 4 short a 5 int b,result 6 7 result = add2(a, b) 8 9 end macro_command </pre>
---	--

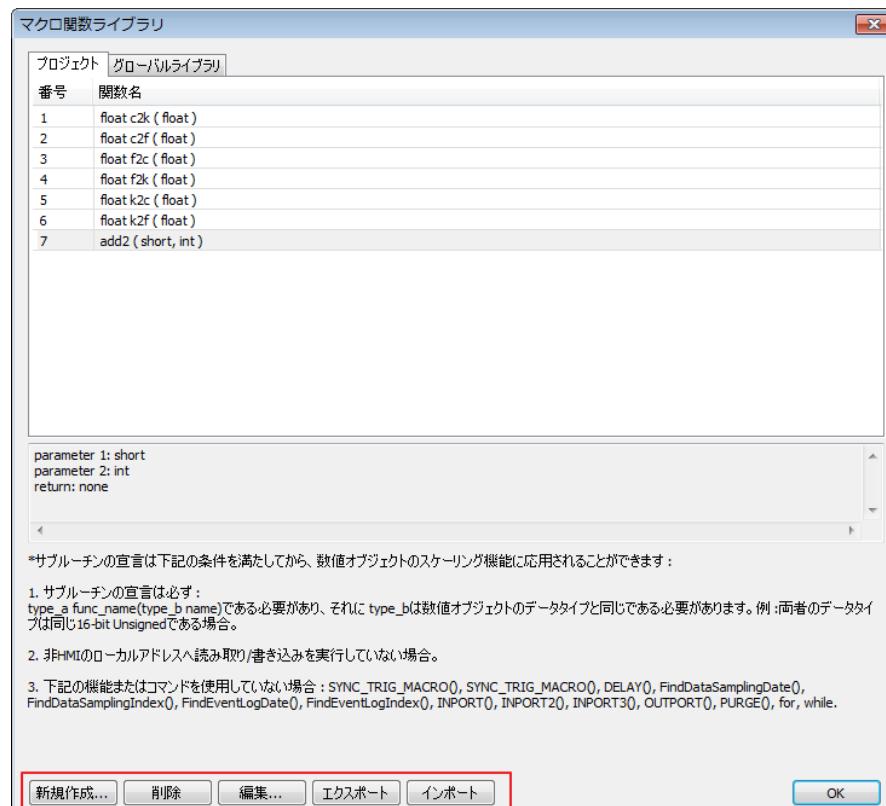
6. ユーザーが上記手順に従うと、ユーザ一定義関数を使用することができ、重複で関数を定義する手間を有効に節約できます。

18.9.3. 関数ライブラリ管理インターフェース

1. マクロ管理ダイアログボックスをオープンし、右下隅の[ライブラリ]ボタンを押し、関数ライブラリダイアログボックスインターフェースに入ります。



2. 関数ライブラリ管理ダイアログボックスの中には関数リストがあります。本リストにはプロジェクトがオープンされた時に、編集ソフトウェアがデフォルト関数ライブラリにロードする全ての関数を表示します。



3. 関数リストの各関数のフォーマットは以下のとおりです：

return_type function_name (parameter_type1, ..., parameter_typeN)

return_type は戻り値のタイプを示し、戻り値がなかったら、本欄が省略されます。

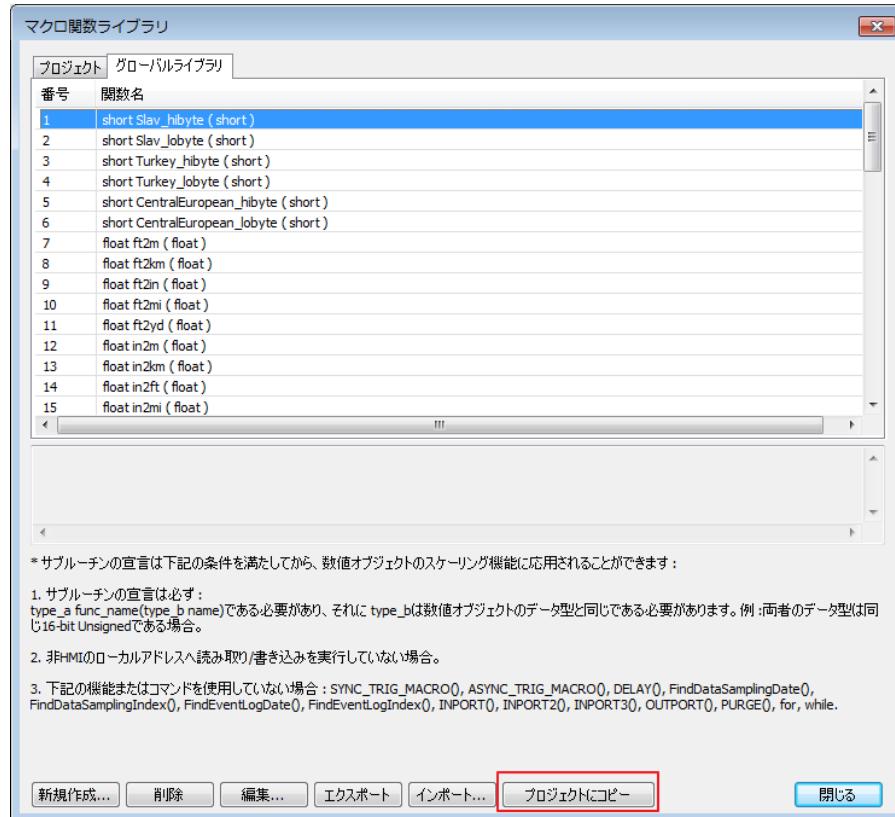
function_name は関数名を示しています。*parameter_typeN* は第 N 個のパラメータタイプを示し、もし関数にはパラメータを受け入れない場合、本欄が省略されます。

```

1 | 1 sub int ADD(int a, int b)
2 | 2   int ret
3 | 3   ret = a+b
4 | 4   return ret
5 | 5 end sub
6 |

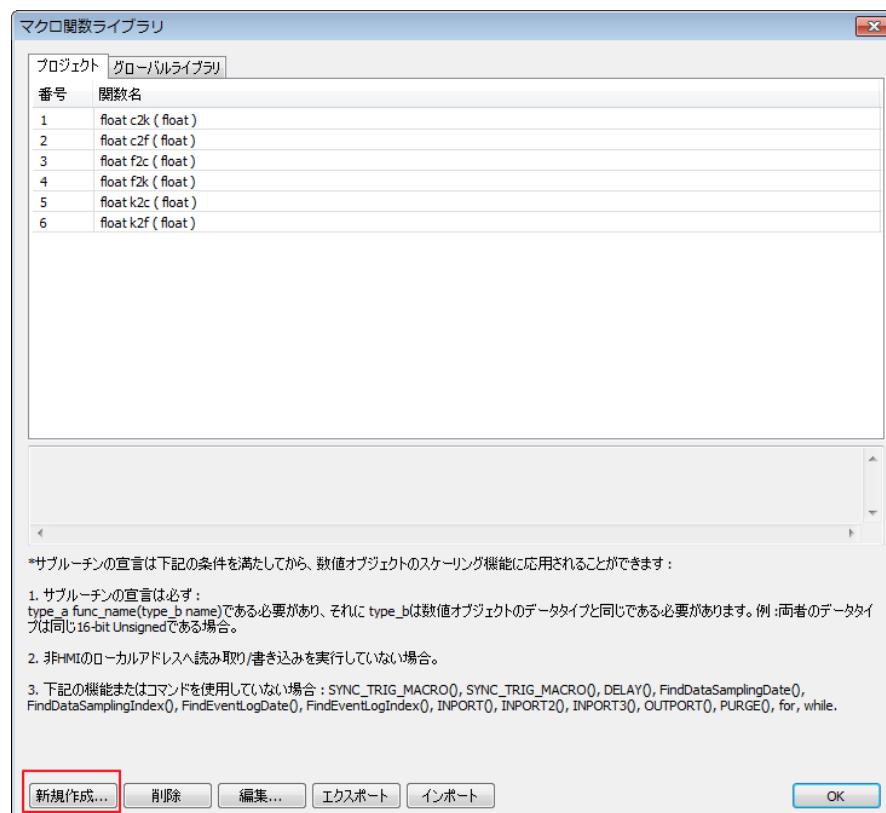
```

4. 関数はプロジェクトに埋め込みます。関数を選択した後、[プロジェクトにコピー]をクリックしたら、[プロジェクト]タブにこの関数が見られます。それで、ユーザーがプロジェクトを他の PC でオープンする場合、依然にこの関数を使用できます。プロジェクトをコンパイルする時には、使用した埋め込み関数を.exob ファイルにコンパイルします。使用していない関数の場合、デコンパイルした後、プロジェクトには当関数が見つかりません。

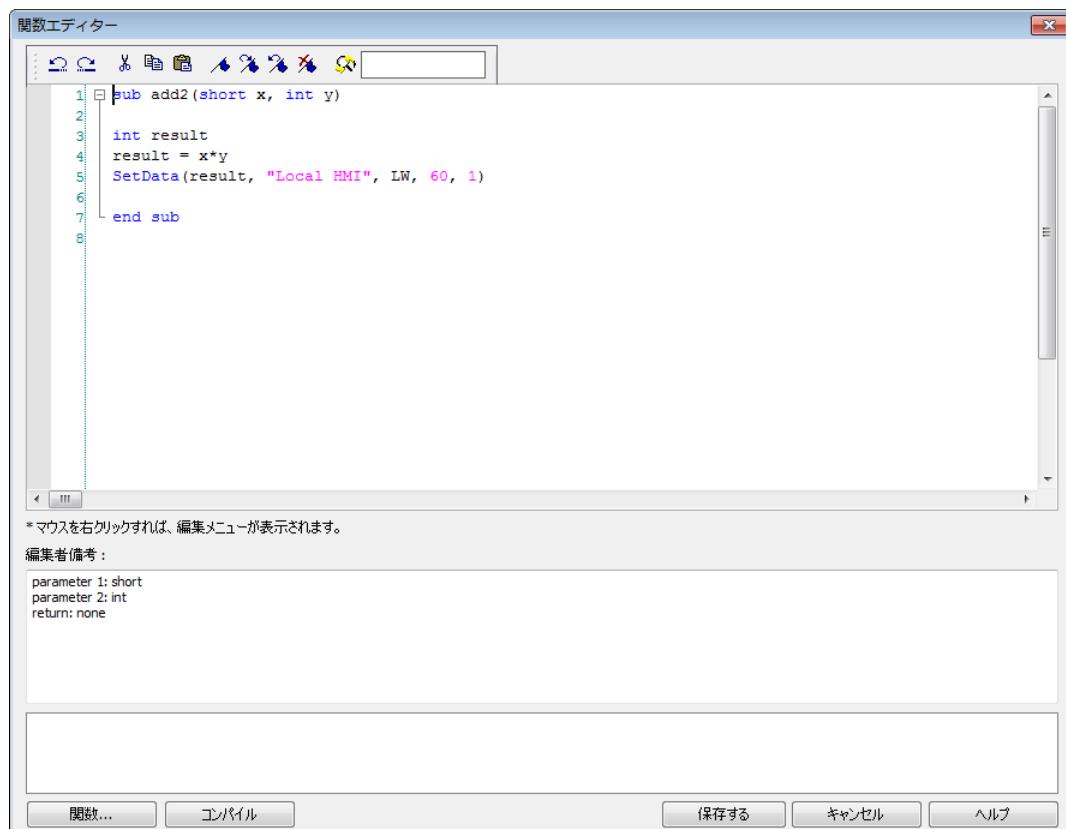


18.9.3.1. 関数を新規作成する

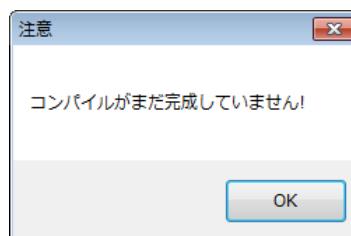
1. [新規作成]を押して関数エディターに入ります。



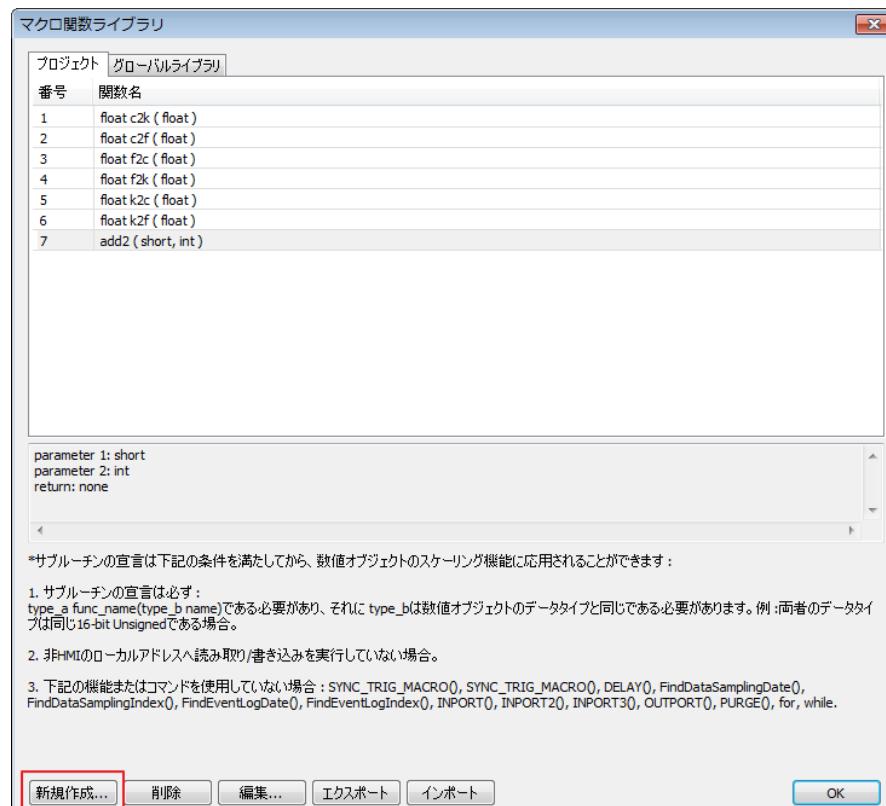
2. 関数編集フィールドで関数を編集し、そして[コンパイル]を押して[保存する]を押します。



3. ユーザーは関数記述編集フィールドで記述を編集することができ、例えば当関数の仕様、使用方法、作成者宣言...などです。
4. 関数の編集が完成後、コンパイルをしてから、[保存する]を押し関数ライブラリに保存されます。さもなければ、下図のポップアップウィンドウが出て、関数のコンパイルがまだ完成していないと警告されます。



5. 関数ライブラリに追加することに成功しました。

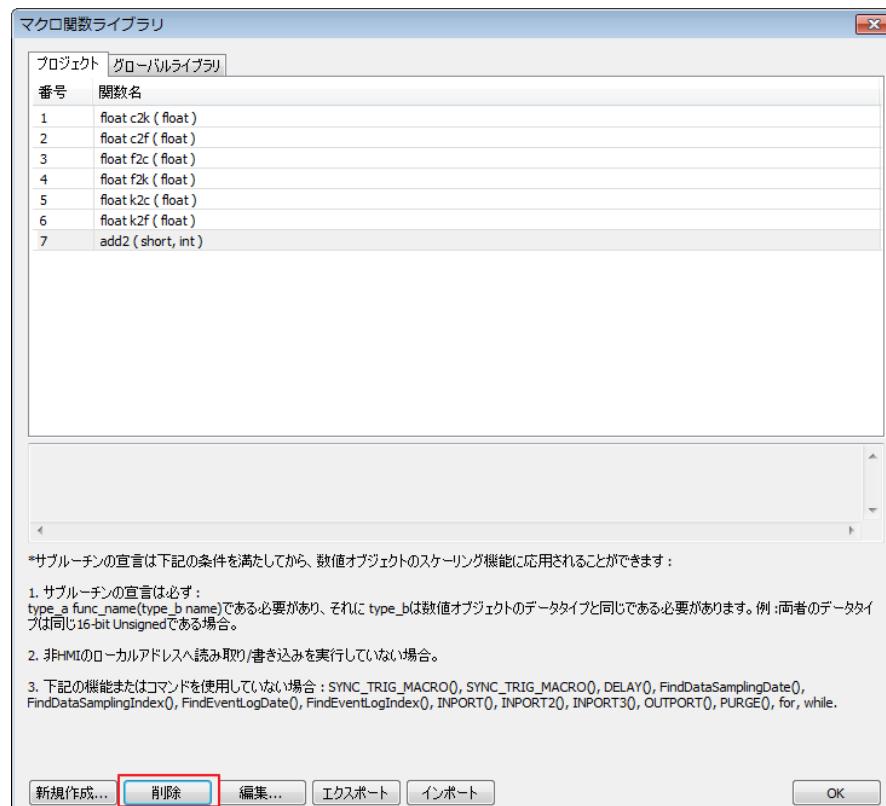


Note

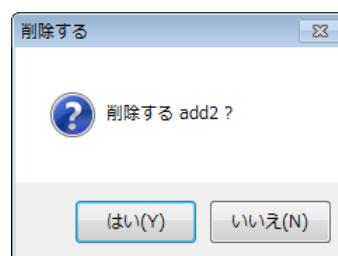
- 関数の中に宣言できるデータ型総数は合計 4096 バイトです。
- 関数名は英数字でなければなりません。それに数字を頭文字にすることを禁じます。

18.9.3.2. 関数を削除する

1. 関数リストに削除したい関数を選択し、[削除]ボタンを押せば、その関数を削除することができます。

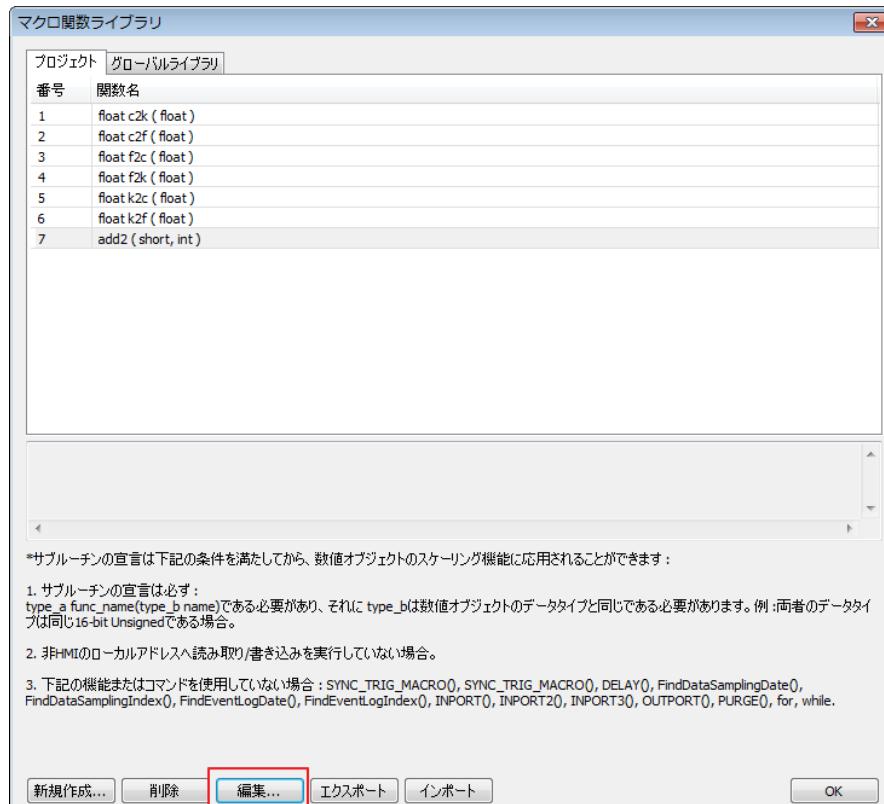


2. [Yes]を押したら、削除します。[No]を押したら、削除をキャンセルします。[Yes]を押して MAX_SHORT という関数を削除します。

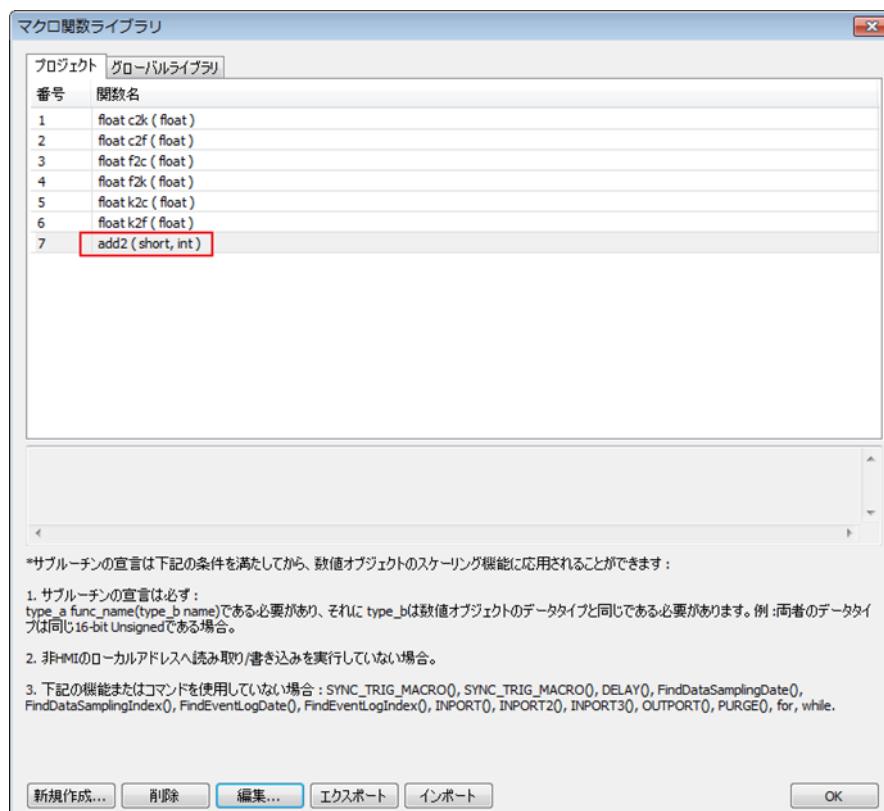


18.9.3.3. 関数を修正する

1. ユーザーは関数ライブラリにある関数を修正することができます。
2. 修正したい関数を選択し、[編集]ボタンを押し、関数エディターに入ります。



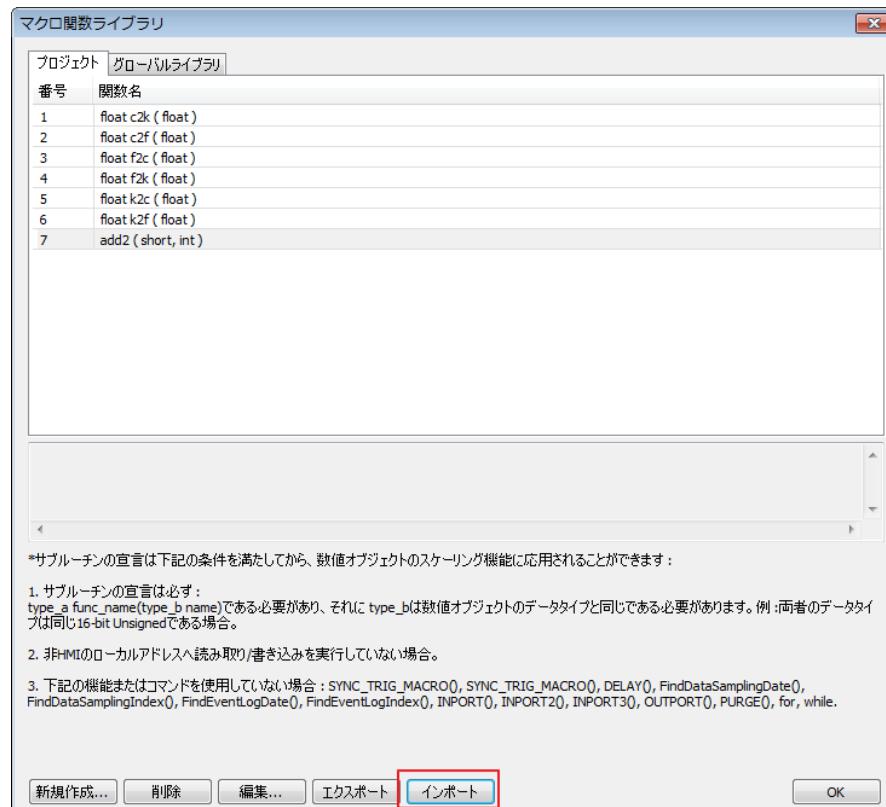
3. または直接に修正したい関数の上に、マウスの左ボタンをダブルクリックし、関数エディターに入ります。



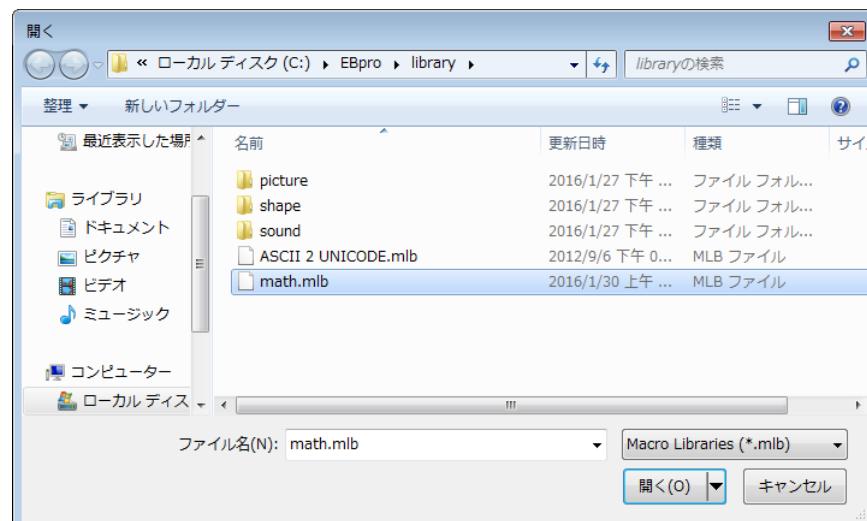
4. 修正が完成したら、同じくコンパイルを経てから保存できます。

18.9.3.4. 関数をインポートする

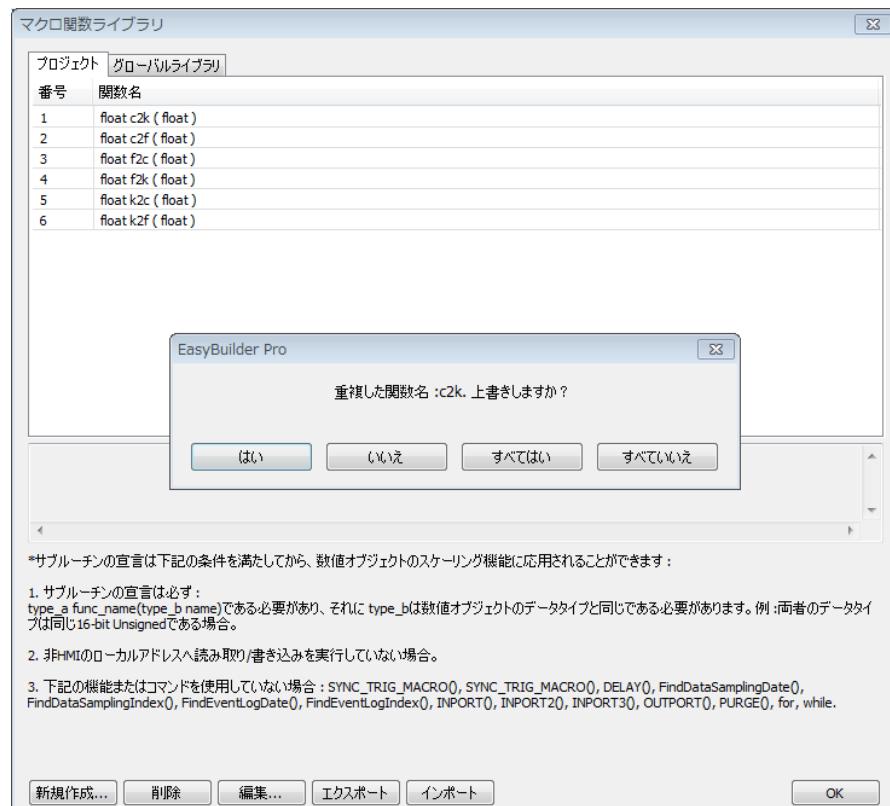
1. ユーザーは外部の.mlb ファイルを使って、関数をインポートすることができます。



2. 仮に、関数ライブラリ math.mlб 内の関数 test1 をインポートしたい場合、[開く]ボタンを押します。



3. 関数をインポートする際に、もし関数ライブラリに既に同じ名前の関数が存在している場合、下記のポップアップウインドウが出ます。



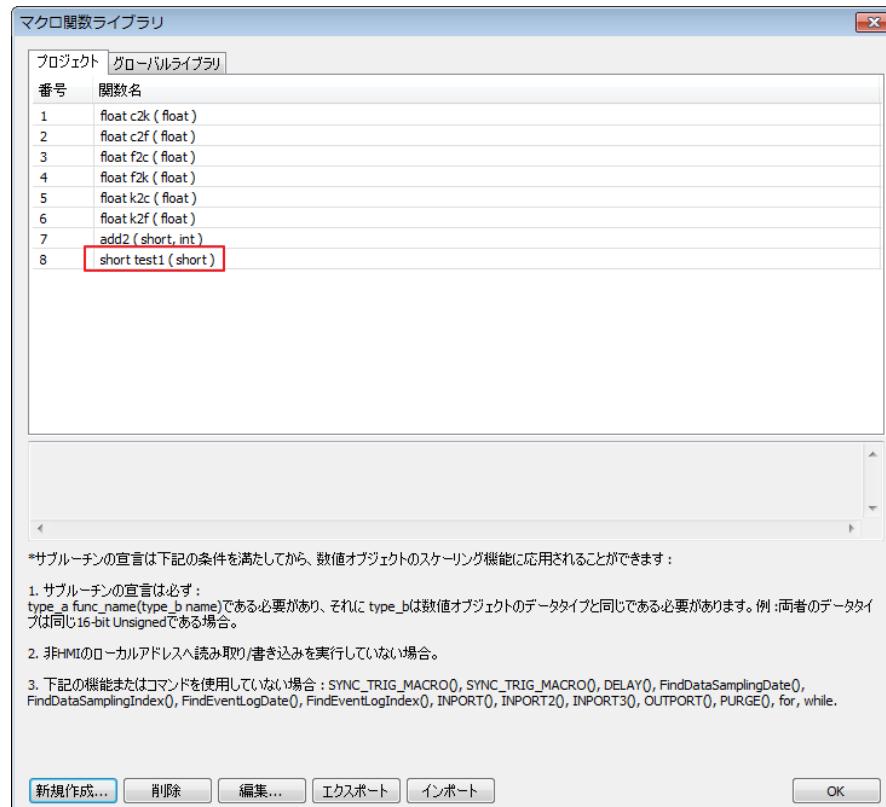
[はい]：外部からインポートする関数で、既存の同じ名前の関数を上書きします。

[いいえ]：外部からこの関数をインポートすることを諦めます。

[全てはい]：外部からインポートする関数を、全ての同じ名前の関数に上書きします。

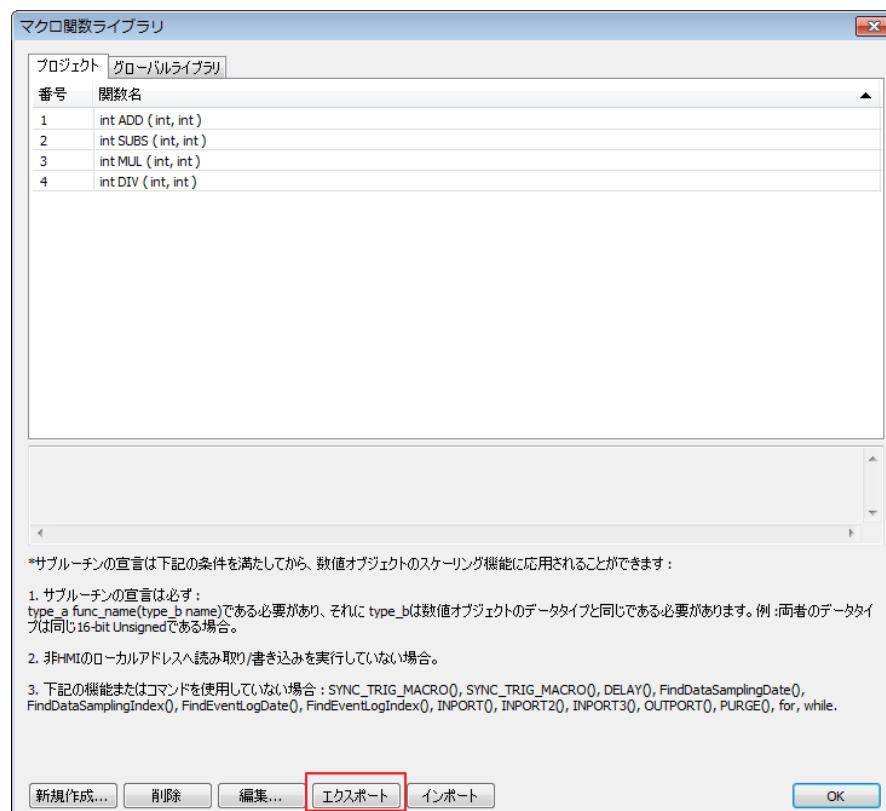
[全ていいえ]：同じ名前の関数を上書きすることを全部諦めます。

4. インポートが完成した後、インポートした関数がデフォルト関数ライブラリに書き込まれたので、ユーザーは `math.mlb` ファイルを削除してもいいです。`test1` のほうは依然に関数ライブラリに存在していて、HMI を再起動しても同様です。

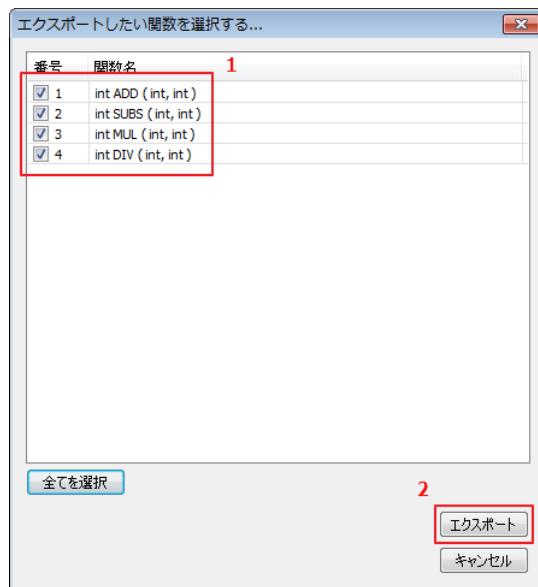


18.9.3.5. 関数をエクスポートする

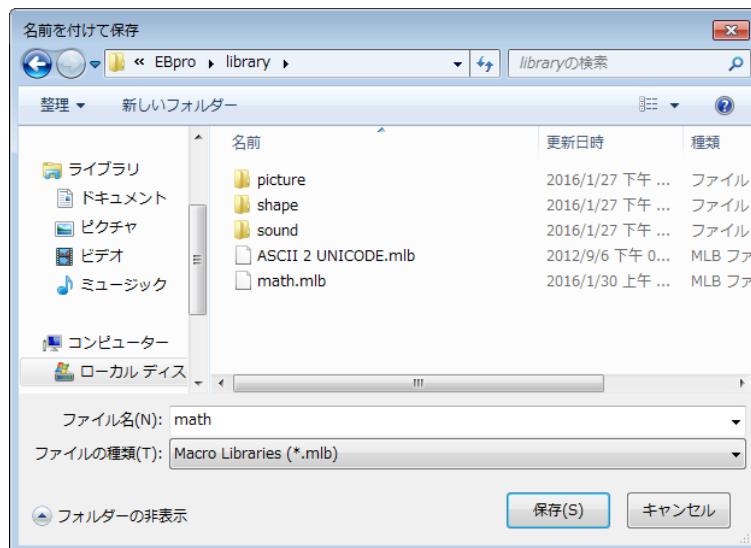
- ユーザーは関数ライブラリ内の関数を.mlbファイルをエクスポートすることができます。
[エクスポート]ボタンを押します。



2. エクスポートしたい関数を選択し、[エクスポート]ボタンを押します。



3. エクスポートしたディレクトリの下に `math.mlb` という関数ライブラリファイルが現れ、`ADD`、`SUBS`、`MUL`、`DIV` という四個の関数があります。
4. エクスポートした `.mlb` ファイルは他の PC に運ぶことができ、ユーザーが編集ソフトウェアをオープンしインポートを完成したら、このファイル内の全ての関数を使用することができます。



18.10.マクロを使用する際の注意点

1. 局所変数の容量は 4KB なので、各種の異なる変数型の最大配列のサイズは以下のとおりです：

char a[4096]

bool b[4096]

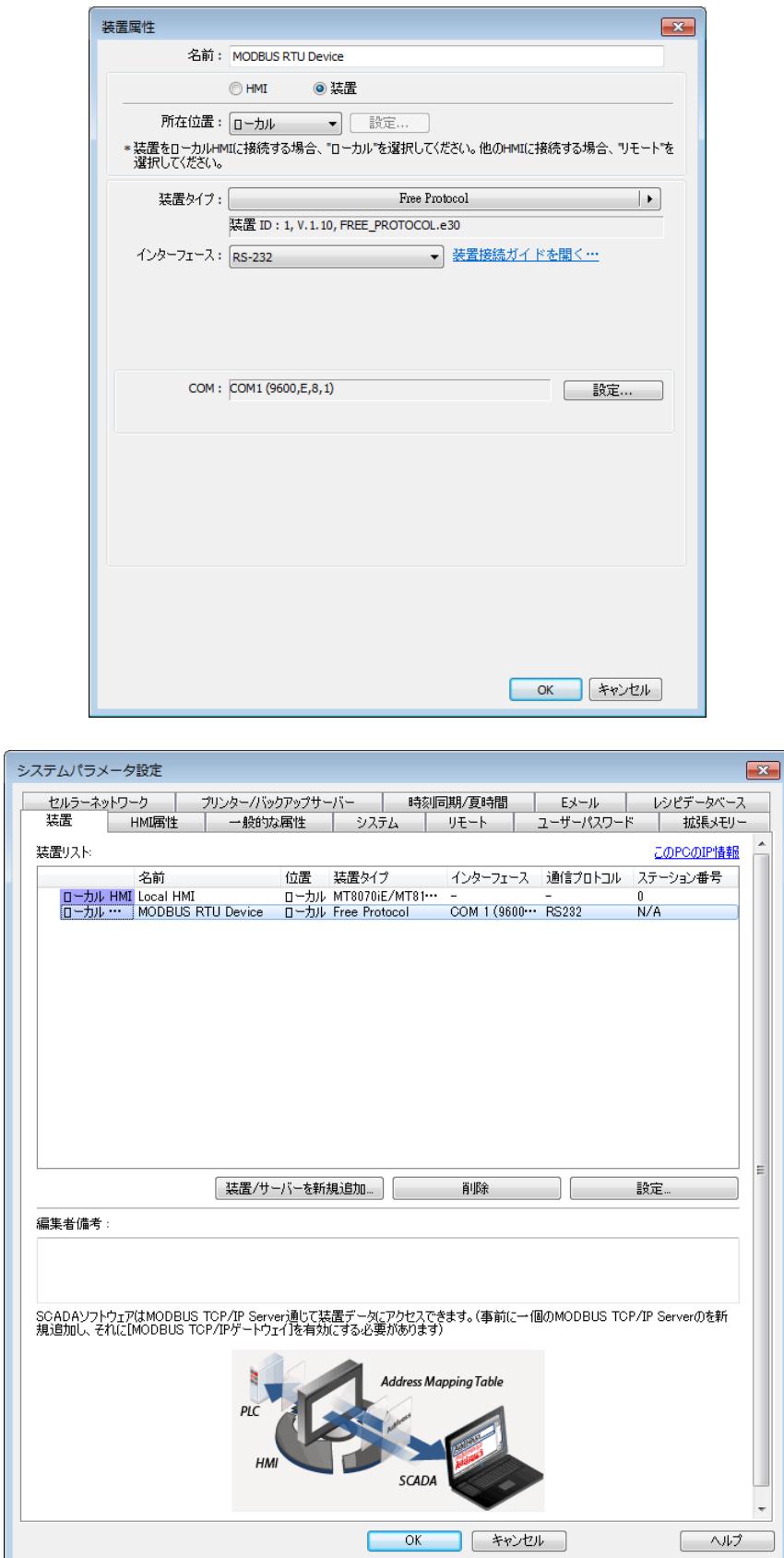
```
short  c[2048]
int    d[1024]
float  e[1024]
long   f[512]
double g[512]
```

2. 一個の EasyBuilderPro プロジェクトの中には、最大 255 個のマクロを含むことができます。
3. マクロは HMI のクラッシュを起こす可能性があります。可能な原因は：
 - マクロの中に無限ループが含まれた
 - 配列のサイズが変数の許容容量を超えた
4. PLC 通信速度はマクロコマンドの実行速度を影響する可能性があります。一方、多すぎたマクロを使用したら、PLC との通信速度が遅くなる可能性があります。

18.11. フリー通信プロトコルで装置をコントロールする

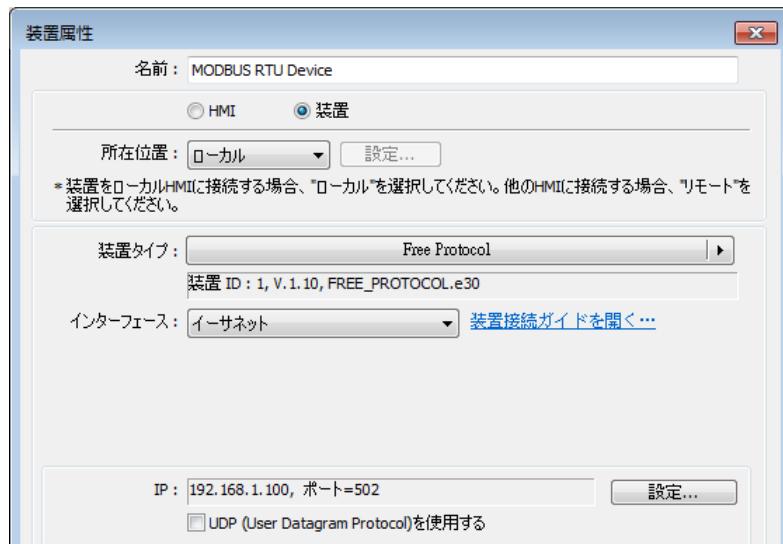
EasyBuilder Pro にある装置との通信用のドライバが内蔵していない場合、ユーザーもマクロでの OUTPORT と IMPORT 関数でその装置との通信を実現することができます。OUTPORT と IMPORT 関数で発送、及び受け入れるデータはその装置の通信プロトコルに従う必要があります。下記の例ではどのようにこの二つの関数で一個の MODBUS RTU 装置をコントロールするのかを説明します。

1. まず、システムパラメータ/装置リストで装置を新規作成します。下図のようにこの新規作成された[装置タイプ]を“Free Protocol”に設定し、[装置]を“MODBUS RTU Device”に設定します。



- この装置接続は RS232 を使用し、一個の MODBUS TCP/IP 装置を接続する場合、このインターフェースのタイプを“イーサネット”に設定する必要があり、同時に正確な IP アドレス

とポート番号を設定する必要があります。下図をご参考ください。



仮に HMI インターフェースが装置の 4x_1 と 4x_2 レジスタを読み取る場合、まず、OUTPORT 関数を使って読み取りリクエストをこの装置に発信します。

OUTPORT 関数の書き込み方は：

`OUTPORT(command[start], device_ 関数名, cmd_count)`

“MODBUS RTUDevice”は MODBUS RTU 装置なので、読み取りリクエストは MODBUS RTU プロトコルの規則に従う必要があります。だから 0x03 というコマンドで、4x_1 と 4x_2 レジスタ内のデータを読み取ることになります。下図では読み取りリクエストのフォーマットを説明します。(装置のステーション番号と最後の二個 CRC バイトは省略された)

Request		
Function code	1 Byte	0x03
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Registers	2 Bytes	1 to 125 (0x7D)

Response		
Function code	1 Byte	0x03
Byte count	1 Byte	2 x N*
Register value	N* x 2 Bytes	

*N = Quantity of Registers

Error		
Error code	1 Byte	0x83
Exception code	1 Byte	01 or 02 or 03 or 04

この MODBUS RTU プロトコルを基づくと、送信するコマンドの内容は以下のとおりです：(合計 8 バイトです)

command[0]: station number	(BYTE0)ステーション番号
command[1]: function code e	(BYTE1)機能コード
command[2]: high byte of starting address	(BYTE2)開始アドレスの上位バイト
command[3]: low byte of starting address	(BYTE3)開始アドレスの下位バイト
command[4]: high byte of quantity of registers	(BYTE4)データレジスタの上位バイト
command[5]: low byte of quantity of registers	(BYTE5)データレジスタの下位バイト
command[6]: low byte of 16-bit CRC	(BYTE6)CRC の下位バイト
command[7]: high byte of 16-bit CRC	(BYTE7)CRC の上位バイト

よりまして、データ読み取りのリクエストは、下記のようにデザインされました：

```
char command[32]
short address, checksum

FILL(command[0], 0, 32) // initialize command[0]~command[31] to 0

command[0] = 0x1 // station number
command[1] = 0x3 // read holding registers (function code is 0x3)

address = // starting address (4x_1) is 0
HIBYTE(address, command[2])
LOBYTE(address, command[3])

read_no = 2 // the total words of reading is 2 words
HIBYTE(read_no, command[4])
LOBYTE(read_no, command[5])

CRC(command[0], checksum, 6) // calculate 16-bit CRC

LOBYTE(checksum, command[6])
HIBYTE(checksum, command[7])
```

最後、OUTPORT 関数を使ってこの読み取りコマンドを MODBUS RTU 設備に送信します。

```
OUTPORT(command[0], "MODBUS RTU Device", 8) // 読み取りコマンドを送信する
```

このコマンドを送信完了後、IMPORT 関数を使ってその MODBUS RTU 装置が返信したレスポンスを読み取ります。MODBUS RTU プロトコルを基づくと、このレスポンスの内容は以下のとおりです(9 バイトがある)：

response[0]:station number	(BYTE0)ステーション番号
response[1]:function code	(BYTE1)機能コード
response[2]:byte count	(BYTE2)バイト長さ
response[3]:highbyte of 4x_1	(BYTE3)一番目のデータの上位バイト
response[4]:lowbyte of 4x_1	(BYTE4)一番目のデータの下位バイト
response[5]:highbyte of 4x_2	(BYTE5)二番目のデータの上位バイト
response[6]:highbyte of 4x_2	(BYTE6)二番目のデータの下位バイト
response[7]:lowbyte of 16-bit CRC	(BYTE7)CRC の下位バイト
response[8]:highbyte of 16-bit CRC	(BYTE8)CRC の上位バイト

この場合、IMPORT 関数のステートメントは以下のとおりです：

```
IMPORT(response[0], "MODBUS RTU Device", 9, return_value) // レスポンスを読み取る
```

関数の中で、実際に読み取ったバイトの長さを変数 `return_value`(変数の型はバイト)に保存しま

す。もし `return_value` での数値は 0 の場合、`IMPORT` でコマンドを読み取るのが失敗したことを示しています。

MODBUS RTU プロトコルを基づくと、もしレスポンスが正確だったら、`response[1]` は必ず `0x03` でなければなりません。正確なコマンドを読み取った後、`4x_1` と `4x_2` レジスタの数値が計算され、それを HMI インターフェースの `LW-100` と `LW-101` レジスタに送信します。

```
If (return_value) >0 and response[1] == 0x3) then  
    read_data[0] = response[4] + (response[3] << 8) // 4x_1 のデータを計算する  
    read_data[1] = response[6] + (response[5] << 8) // 4x_2 のデータを計算する  
  
    SetData(read_data[0], "Local HMI", LW, 100, 2) // データを HMI に保存する  
endif
```

完全なマクロは下記に示す通りです：

```
// Read Holding Registers
macro_command main()

char command[32], response[32]
short address, checksum
short read_no, return_value, read_data[2], i

FILL(command[0], 0, 32)//  initialize command[0]~command[31] to 0
FILL(response[0], 0, 32)

command[0] = 0x1//  station number
command[1] = 0x3//  read holding registers (function code is 0x3)

address = 0
address = 0//  starting address (4x_1) is 0
HIBYTE(address, command[2])
LOBYTE(address, command[3])

read_no = 2//  the total words of reading is 2 words
HIBYTE(read_no, command[4])
LOBYTE(read_no, command[5])

CRC(command[0], checksum, 6)//  calculate 16-bit CRC

LOBYTE(checksum, command[6])
HIBYTE(checksum, command[7])

OUTPORT(command[0], "MODBUS RTU Device", 8 )//  send request
INPORT(response[0], "MODBUS RTU Device", 9, return_value)//  read response

if (return_value > 0 and response[1] == 0x3) then
    read_data[0] = response[4] + (response[3] << 8)//  4x_1
    read_data[1] = response[6] + (response[5] << 8)//  4x_2

    SetData(read_data[0], "Local HMI", LW, 100, 2)
end if

end macro_command
```

下記ではどのようにフリー通信プロトコルを使って MODBUS RTU 装置での 0x_1 状態を設定するのかを説明します。これは MODBUS RTU プロトコルでの“シングルコイルを書き込む”の機能コード“0x05”で実現するものです。

Request

Function code	1 Byte	0x05
Output Address	2 Bytes	0x0000 to 0xFFFF
Output Value	2 Bytes	0x0000 or 0xFF00

Response

Function code	1 Byte	0x05
Output Address	2 Bytes	0x0000 to 0xFFFF
Output Value	2 Bytes	0x0000 or 0xFF00

Error

Error code	1 Byte	0x85
Exception code	1 Byte	01 or 02 or 03 or 04

完全なマクロは下に示す通りです：

```
// Write Single Coil (ON)
macro_command main()

char command[32], response[32]
short address, checksum
short i, return_value

FILL(command[0], 0, 32)// initialize command[0]~ command[31] to 0
FILL(response[0], 0, 32)

command[0] = 0x1// station number
command[1] = 0x5// function code : write single coil

address = 0
HIBYTE(address, command[2])
LOBYTE(address, command[3])

command[4] = 0xff// force 0x_1 on
command[5] = 0

CRC(command[0], checksum, 6)

LOBYTE(checksum, command[6])
HIBYTE(checksum, command[7])

OUTPORT(command[0], "MODBUS RTU Device", 8)// send request
IMPORT(response[0], "MODBUS RTU Device", 8, return_value)// read response

end macro_command
```



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

18.12.コンパイルエラーメッセージ

- エラーメッセージフォーマット :

errorC#:error 記述

(#はエラーメッセージ番号)

例 : error C37 : undeclared identifier : i

コンパイル後にエラーがある場合、このエラーの記述内容をエラーメッセージ番号をご参照。

- Error 記述

(C1) シンタックス error : “識別子”

この情報が出た原因としては、多くの可能性があります :

例:

```
macro_commandmain()  
chari, 123xyz// この変数型をサポートしていない  
endmacro_command
```

(C2) ‘identifier’ used without having been initialized (使用する識別子は初期化されていなかった)

マクロは宣言する配列変数のサイズを定義する必要があります。

例:

```
macro_commandmain()  
chari  
intg[i]// i 一個の数値定数である必要  
endmacro_command
```

(C3) redefinition error : ‘identifier’ (識別子が重複的に定義された)

関数、変数名が有効エリア内には唯一であるべきです。

例:

```
macro_commandmain()  
intg[10], g// 重複定義エラー  
endmacro_command
```

(C4) function 関数名 error:‘identifier’ (関数名定義エラー)

予約語は関数名に定義されません。

例：

subintif()// 関数名定義エラー

(C5) parentheses have not come in pairs (丸括弧はペアではなかった)

ステートメントには“("or")”が欠落している

例：

macro_commandmain)// “(“が欠落している

(C6) illegal expression without matching 'if'(if ステートメントには合法な数式がない)

即ち if ステートメントには数式が欠落しています。

(C7) illegal expression (no 'then') without matching 'if'(if ステートメントには then が欠落している)

即ち if と then はペアでできていませんでした。

(C8) illegal expression (no 'end if') (if ステートメントには enf if が欠落している)

“end if”が欠落しています。

(C9) illegal 'end if' without matching 'if'(end if ステートメントの前に if が欠落している)

End if ステートメントの前には if ステートメントが欠落しています。

(C10) illegal 'else' (違法な else ステートメント)

この“if”ステートメントのフォーマットは：

if[論理数式]then

[else[if[論理数式]then]]

End if

上記フォーマットと合っていないステートメントがあったら、コンパイルをする際にエラーが
出ます。

**(C17) illegal expression (no 'for') without matching 'next'(next とマッチしている for ステートメント
がない)**

“for”ステートメントエラー：“next”の前に、“for”ステートメントが欠落しています。

(C18) illegal variable type (not integer or char)(違法な変数型)

変数型の定義エラー、ここは整数またはキャラクタ型の変数であること

(C19)variable type error

代入ステートメントが欠落しています。

(C20) must be keyword 'to' or 'down'"(キーワード“to”或いは“down”が欠落している)

キーワード“to”或いは“down”が欠落しています。

(C21) illegal expression (no 'next') (違法な数式で、“next”が欠落している)

“for”ステートメントのフォーマットは：

for[変数]=[開始値]to[終了値][step]

next[変数]

上記フォーマットと合っていないステートメントがあったら、コンパイルをする際にエラーが出来ます。

(C22) ‘wend’ statement contains no ‘while’

ステートメントに“while”キーワードが欠落し、“wend”的前面には“while”があるべきです。

(C23) illegal expression without matching ‘wend’

“wend”キーワードが欠落している

“while”ステートメントのフォーマットは：

while[論理式]

wend

上記フォーマットと合っていないステートメントがあったら、コンパイルをする際にエラーが出来ます。

(C24) シンタックス error: ‘break’

不正な“break”ステートメントです。breakステートメントはforステートメント、或いはwhileステートメントのみで使用できます。

(C25) シンタックス error: ‘continue’

不正な“continue”ステートメントです。continueステートメントは“for”或いは“while”ステートメントでしか現れません。

(C26) シンタックス error

数式が正しくないです。

(C27) シンタックス error

数式には一個の演算符が欠落したら、このエラーが出る可能性があります。

例:

```
macro_command main()
inta, b
fora=0 to 2
b=4+xyz // 不正な内容 : xyz 変数は定義されていなかった
next a
end macro_command
```

(C28) must be 'macro_command'

ここは“macro_command”であるべきです

(C29) must be keyword 'sub'

サブ関数の定義フォーマットは：

```
sub [data type] function_関数名(...)
.....
endsub
```

例:

```
Sub int pow(intexp)
.....
end sub
```

上記フォーマットに合っていないステートメントがあったら、コンパイルをする際にエラーが
出ます。

(C30) number of parameters is incorrect

パラメータの数は正しくないです。

(C31) parameter type is incorrect

パラメータのデータ型が合っていません。関数を呼び出す時に、パラメータはデータ型、数に
一々に対応しているこそ、コンパイルに成功できます。さもなければ、コンパイルの際にエラ

ーが出ます。

(C32) variable is incorrect

変数型が正しくないです。変数をパラメータにし、一個の関数に送信される際に、変数のデータ型は関数が宣言するタイプと同じである必要があり、さもなければこのエラーが出ます。

(C33) function 関数名: undeclared function n

定義した関数名がありません

(C34) expected constant expression

不正な配列インデックスフォーマット

(C35) invalid array declaration

不正な配列宣言

(C36) array index error r

不正な配列インデックス

(C37) undeclared identifier : i 'identifier'

宣言していない変数を使用してしまいます。宣言された変数及び関数だけが使用でき、さもなければコンパイルの際にこのエラーメッセージが出ます。

(C38) un-supported PLC data address

通信関数 GetData(...)、SetData(...)のパラメータには PLC アドレスタイプの情報が含まれていて、PLC アドレスのタイプはこの PLC がサポートするアドレスではない場合、コンパイルの際にこのエラーメッセージが出ます。

(C39) 'idenifier' must be integer, char or constant

配列のフォーマットは：

宣言：array_関数名[constant](constantisthesizeofthearray)

使用：array_関数名[integer, characterorconstant]

上記フォーマットに合っていないステートメントがあったら、コンパイルをする際にエラーが出ます。

(C40) execution シンタックス should not exist before variable declaration or constant definition

変数宣言ステートメントの前には、実行ステートメントを禁じます。

例：

```
macro_command main()
```

```
int a, b
```

```
for a=0 To 2
```

```
b=4+a
```

int h, k // ここで変数定義ステートメントは正しくないです。関数の中に、変数定義ステートメントの前には実行ステートメントが存在してはいけません。例えば `b=4+a`。

```
next a
```

```
end macro_command
```

(C41) float variables cannot be contained in shift calculation

シフト演算の中に、被演算子は浮動小数点数であってはいけません。

(C42) function must return a value

関数は戻り値を返信すべきです。

(C43) function should not return a value

関数は戻り値を返信すべきではありません。

(C44) float variables cannot be contained in calculation

演算の中で、float 型データを含んではいけません。

(C45) PLC address error

PLC アドレスエラー。

(C46) array size overflow (max. 4k)

一次元配列のサイズが 4k を超えました。

(C47) macro command entry function is not only one

マクロコマンドの関数入り口は一つだけでなければなりません。

(C48) macro command entry function must be only one

マクロコマンドの関数入り口は唯一ではありません。マクロコマンドの関数入り口はただ一つであるべきです。フォーマットは：

```
macro_command function_関数名()
```

```
end macro_command
```

(C49) an extended addressee's station number must be between 0 and 255

拡張アドレスでのステーション番号は 0~255 の範囲にあるべきです。

例:

```
SetData(bits[0], "PLC1", LB, 300#123, 100)  
// illegal:300#123 とはステーション番号は 300 を指したが、但し最大値は 255
```

(C50)an invalid PLC 関数名

PLC 名はまだシステムパラメータの装置リストに定義されていません。

(C51) macro command do not control a remote device

マクロコマンドはローカルに接続した装置のみコントロールできます。

例:

```
SetData(bits[0], "PLC1", LB, 300#123, 100)  
"PLC1"は遠隔地にあるため、実行されません。
```

18.13.マクロコマンドのサンプルプロジェクト

- for ステートメント、各種の数式(算術、シフト、論理、関係式)

```
macro_command main()
    int a[10], b[10], i

    b[0] = (400 + 400 << 2) / 401
    b[1] = 22 * 2 - 30 % 7
    b[2] = 111 >> 2
    b[3] = 403 > 9 + 3 >= 9 + 3 < 4 + 3 <= 8 + 8 == 8
    b[4] = not 8 + 1 and 2 + 1 or 0 + 1 xor 2
    b[5] = 405 and 3 and not 0
    b[6] = 8 & 4 + 4 & 4 + 8 | 4 + 8 ^ 4
    b[7] = 6 - (~4)
    b[8] = 0x11
    b[9] = 409
```

```
    for i = 0 to 4 step 1
        if (a[0] == 400) then
            GetData(a[0],"Device 1", 4x, 0,9)
            GetData(b[0],"Device 1", 4x, 11,10)
        end If
        next i
    end macro_command
```

- while, if, break ステートメント

```
macro_command main()
    int b[10], i
    i = 5
    while i == 5 - 20 % 3
        GetData(b[1], "Device 1", 4x, 11, 1)

        if b[1] == 100 then
            break
        end if
    wend
end macro_command
```

マクロの説明

- 総体変数とサブ関数の呼び出し

```
char g
sub int fun(int j, int k)
    int y

    SetData(j, "Local HMI", LB, 14, 1)
    GetData(y, "Local HMI", LB, 15, 1)
    g = y

    return y
end Sub
```

```
macro_command main()
    int a, b, i

    a = 2
    b = 3
    i = fun(a, b)
    SetData(i, "Local HMI", LB, 16, 1)
end macro_command
```

- if ステートメント

```
macro_command main()
    int k[10], j

    for j = 0 to 10
        k[j] = j
    next j

    if k[0] == 0 then
        SetData(k[1], "Device 1", 4x, 0, 1)
    end if

    if k[0] == 0 then
        SetData(k[1], "Device 1", 4x, 0, 1)
    else
        SetData(k[2], "Device 1", 4x, 0, 1)
    end if
```

```
if k[0] == 0 then
    SetData(k[1], "Device 1", 4x, 1, 1)
else if k[2] == 1 then
    SetData(k[3], "Device 1", 4x, 2, 1)
end If
```

```
if k[0] == 0 then
    SetData(k[1], "Device 1", 4x, 3, 1)
else if k[2] == 2 then
    SetData(k[3], "Device 1", 4x, 4, 1)
else
    SetData(k[4], "Device 1", 4x, 5, 1)
end If
end macro_command
```

- while と wend ステートメント
macro_command main()

```
char i = 0
int a[13], b[14], c = 4848
```

```
b[0] = 13
```

```
while b[0]
    a[i] = 20 + i * 10

    if a[i] == 120 then
        c = 200
        break
    end if

    i = i + 1
wend

SetData(c, "Device 1", 4x, 2, 1)
end macro_command
```

マクロの説明

● break と continue ステートメント

```
macro_command main()
    char i = 0
    int a[13], b[14], c = 4848
    b[0] = 13

    while b[0]
        a[i] = 20 + i * 10

        if a[i] == 120 then
            c = 200
            i = i + 1
            continue
        end if
        i = i + 1

        if c == 200 then
            SetData(c, "Device 1", 4x, 2, 1)
            break
        end if
    wend

end macro_command
```

● 配列構成

```
macro_command main()
    int a[25], b[25], i

    b[0] = 13

    for i = 0 to b[0] step 1
        a[i] = 20 + i * 10
    next i

    SetData(a[0], "Device 1", 4x, 0, 13)
end macro_command
```

ストリング関連シンタックス内で”記号を使用する方法は、変数宣言及び関数の引数に用いられることもできます。

```
macro_command main()
char data[40]= $"Note" " "
StringCopy("This is a $"test$" for weintek", data[7])
//ストリング内容は"Note" This is a "test" for weintek
end macro_command
```

18.14. マクロ TRACE 関数

マクロの TRACE 関数に EasyDiagnoser/cMT Diagnoser を合わせて使用すると、使用している変数の現状を閲覧することができます。cMT シリーズを使用する場合、cMT Diagnoser が内蔵したマクロデバッグツールを使用することをお勧めします。TRACE 関数を使用する必要はありません。下記ではマクロで TRACE コマンドを使用し、EasyDiagnoser で監視する方法を説明します。

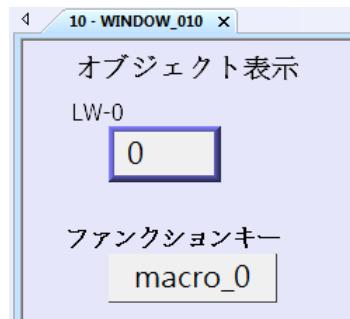
- まず、プロジェクトの中で macro_0 を新規作成し、そして macro_0 の内容に **TRACE("LW=%d", a)**を入れ、“%d”とは 10 進数で LW 現在の数値を表示することを指します。macro_0 の内容は以下のとおりです：

```

1  macro_command main()
2
3
4  short a
5  GetData(a, "Local HMI", LW, 0, 1)
6  a=a+1
7  SetData(a, "Local HMI", LW, 0, 1)
8  TRACE ("LW0 = %d" , a)
9
10 end macro_command

```

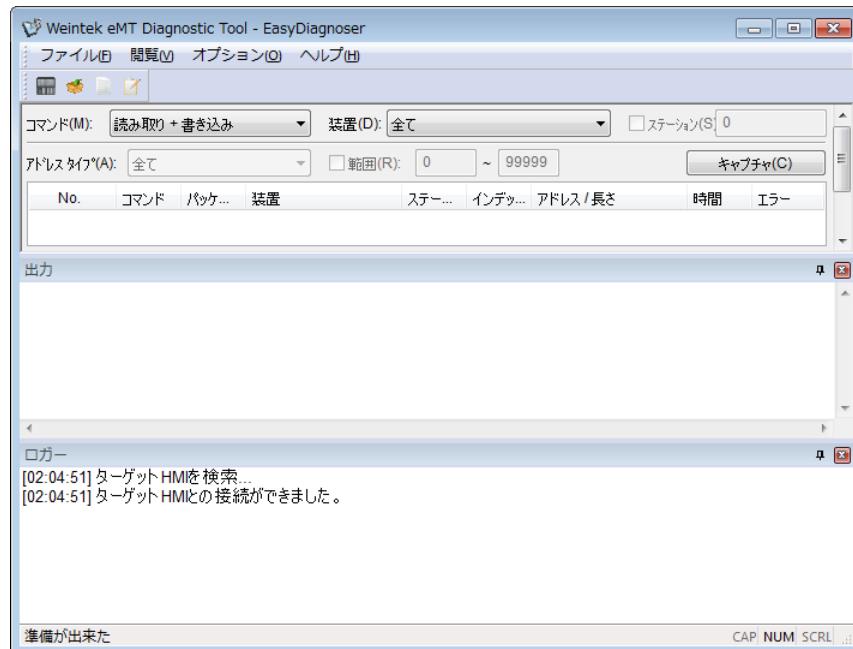
- 続いて、プロジェクトの第 10 タブに個別に[数値表示]と[ファンクションキー]オブジェクトを入れ、[ファンクションキー]オブジェクトで macro_0 を実行します。



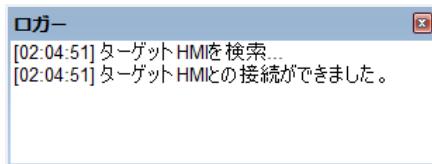
- 最後、完成したプロジェクトをコンパイルし、オフライン/オンラインシミュレーションを行います。
- PC でシミュレーション機能を行う場合、マウスを右クリックした後、メニューの"Run EasyDiagnoser"を選択します。



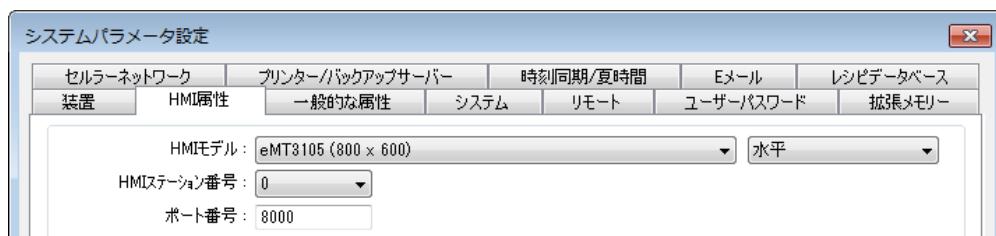
5. この時、EasyDiagnoser の画面が現れます。[ロガー]ウインドウで EasyDiagnoser が監視したい HMI に接続できるかを表示し、[出力]ウインドウで TRACE の実行結果を表示します。下図では EasyDiagnoser が HMI への接続に成功したと示しています。



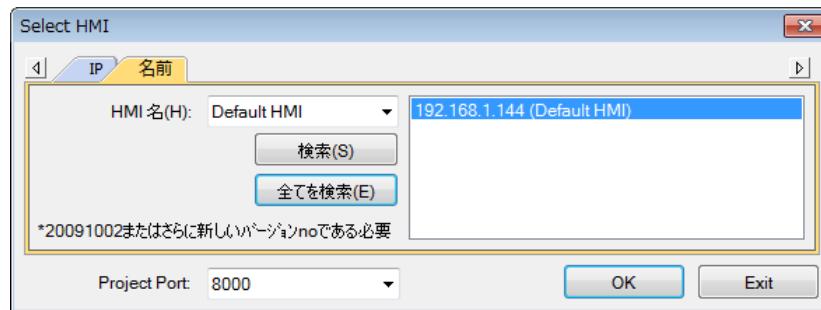
もし接続に成功していなかったら、[ロガー]のウインドウは下記の内容を表示します：



6. オンラインに成功していなかった原因としては、PC でのシミュレーションの実行に成功していなかったか、PC でシミュレーションを実行したプロジェクトが使用した[ポート番号]が不正確だった(システムに使用されたかもしれない)のが考えられ、この場合、プロジェクトの[ポート番号](下図をご参照)を変更し、再度コンパイルし改めてシミュレーションを実行すればいいです。



7. EasyDiagnoser をオープンする時、プロジェクトと同じのポート番号を設定すべきです。



設定した[ポート番号]から起算し、連続3個のポート番号はHMIの通信用で確保されます。例を挙げれば、上図では[ポート番号]を8005と設定したので、8005、8006、8007との3個のポート番号が確保されます。だから、PCでシミュレーションする時、これらの確保されたポートが他のプログラムに使用されていないことを予め確認すべきです。

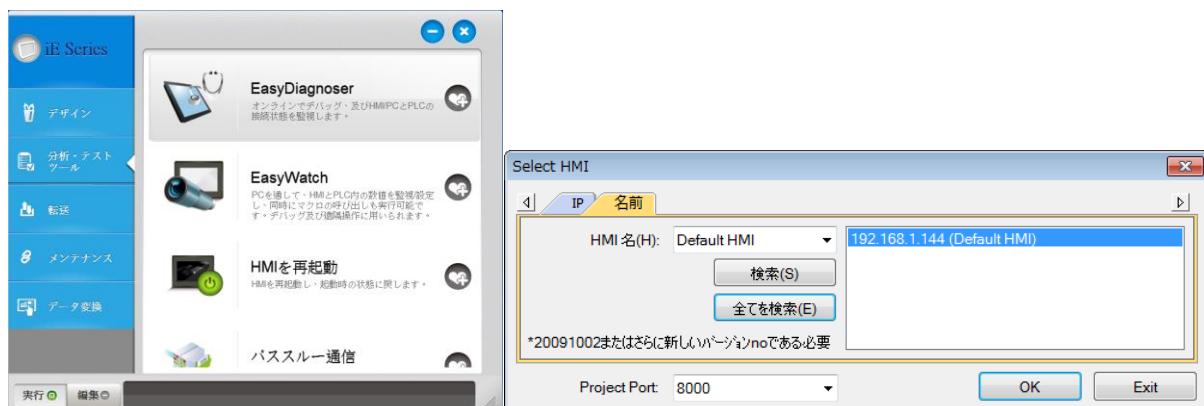
TRACEコマンドのシンタックスは下記に示す通りです：

関数名	TRACE
シンタックス	TRACE(format, argument)
記述	<p>実行中のマクロコマンドは本関数を使って、変数の変化を監視することができます。それにデバッブのため、ストリングをプリントアウトできます。ユーザーはEasyDiagnoser/cMT Diagnoserを通じて、本関数の出力結果が見られます。</p> <p>TRACE関数は頭文字が%の特殊キャラクタを取る途端、同時にargumentから一個のパラメータを取り、フォーマットしてから出力します。</p> <p>Formatとはプリントのフォーマットで、頭文字が%の特殊キャラクタをサポートします。特殊なキャラクタフォーマットは以下のとおりです。その中、角括弧内の欄はオプションで、赤文字の欄は必須です：</p> <p>%[flags][width][.precision]type</p> <p>各欄の意味は下記に述べた通りです：</p> <p>flags(オプション) :</p> <ul style="list-style-type: none"> - : 左に揃えます。幅を指定し、左側で不足の部分を空白キャラクタで補います。 + : プラス・マイナス記号を出力します。 <p>width(オプション) :</p> <p>十進数正の整数であり、キャラクタの長さを指定し、不足の部分を空白キャラクタで補います。</p> <p>precision(オプション) :</p> <p>十進数正の整数であり、精確さ、及び出力キャラクタ数を指定します。</p> <p>type :</p> <ul style="list-style-type: none"> C或いはc : キャラクタで出力します。 d : signed十進数整数で出力します。 i : signed十進数整数で出力します。 o : unsigned八進数整数で出力します。

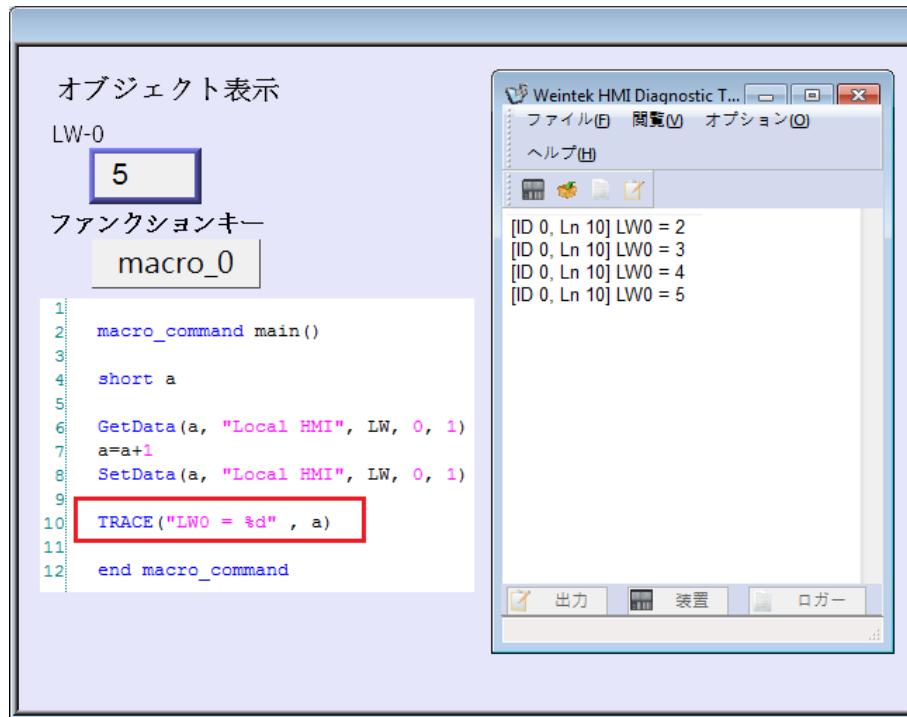
マクロの説明

	<p>u : unsigned十進数整数で出力します。</p> <p>X或いはx : unsigned十六進数整数で出力します。</p> <p>f : 出力フォーマットは[–]dddd.ddddです。その中、欄dddは一個または複数の十進数数字です。</p> <p>lld : signed長整数(64bit)で出力します。(cMTにのみ対応)</p> <p>llu : unsigned長整数(64bit)で出力します。(cMTにのみ対応)</p> <p>f : 単精度浮動小数点数で出力します。</p> <p>lf : 倍精度浮動小数点数で出力します。</p> <p>E或いはe : 出力フォーマットは[–]d.dddde [sign]dddです。その中、欄dは十進数数字、欄dddは一個または複数の十進数数字、欄dddは三個の十進数数字である必要があります。Signは+または-です。</p> <p><i>Format</i>ストリングは最大256個キャラクタをサポートし、超えたキャラクタは省略されます。</p> <p><i>Argument</i>部分は書いても書かなくてもいいです。一個の特殊キャラクタは一個の変数に対応する必要があります。</p>
例	<pre>macro_command main() char c1 = 'a' short s1 = 32767 float f1 = 1.234567 TRACE("The results are") // 出力 : The results are TRACE("c1 = %c, s1 = %d, f1 = %f", c1, s1, f1) // 出力 : c1 = a, s1 = 32767, f1 = 1.234567 end macro_command</pre>

8. LB-9059-マクロコマンド TRACE を閉じる機能(状態が ON の場合)を新規追加しました。これを ON に設定する場合、TRACE はデータを EasyDiagnoser に出力しません。
9. 直接に UtilityManager を使って EasyDiagnoser.exe を実行することもできます。UtilityManager にはネットワーク上の HMI をリストし、そこで監視したい HMI を選択すればいいです。注意すべきなのは、Project Port の部分は必ずプロジェクトが使用している[ポート番号]と同じでなければなりません。

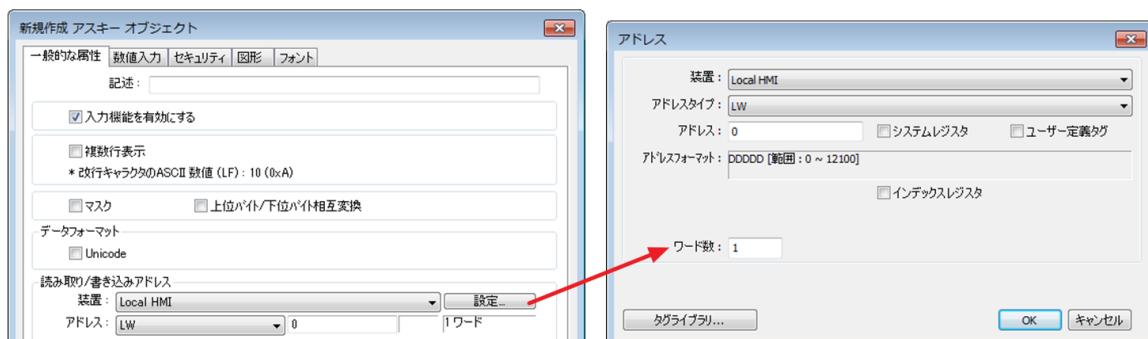


- 10.** プロジェクトを HMI にダウンロードし、実際に操作します。EasyDiagnoser が監視したい HMI に繋がれない場合、一般的な原因としては、HMI の電源が入っていなかったか、或いは[ポート番号]が不正確だったので、EasyDiagnoser は一旦 HMI に繋がったが、また繋がらないという状況になります。EasyDiagnoser がプロジェクトと同じの Port No.に設定することを確保してください。ポート番号の変更方法は、前に述べた通りです。
- 11.** EasyDiagnoser が HMI に繋がった後、macro_0 を実行すれば、[出力]ウインドウで現在 TRACE の実行結果が見られます。



18.15.ストリング操作関数の使用方法

マクロはストリング操作関数を提供しています。それにより、ユーザーは便利にストリングに対し、各種の操作を行うことができます。ストリングというのは、連続的な ASCII キャラクタで構成したもので、一個の ASCII キャラクタは 1 バイト(byte)を占め、16 ビットレジスタの中では下位バイトが優先に保存されます。例を挙げれば、仮に[アスキー]オブジェクトで LW-0~LW-2 の位置に、"abcdef" というストリングを書き込む場合、設定は下記に示す通りです：



この[アスキーオブジェクト]で"abcdef"を入力：



このストリングが LW-0~LW-2 での保存位置は下図に示す通りです(LB は下位バイトを代表し、HB が上位バイトを代表しています)：

	HB	LB
LW0	'B'	'A'
LW1	'D'	'C'
LW2	'F'	'E'
LW3		
LW4		
LW5		

[アスキーオブジェクト]が表示するデータの長さ単位は word で、各 ASCII キャラクタの長度は 1 バイトだから、毎回は最小 2 個のワードが表示されます。この部分について、[アスキーオブジェクト]の章では詳細な説明があります。よりまして、上記の例では、[アスキーオブジェクト]のキャラクタ数が 3 に設定したから、最大 6 個の ASCII キャラクタが入力/表示できると示しています。

以下は、一個のマクロを新規追加する方法から、シミュレーションを実行するまで、もれなく例を挙げてプロジェクトの作成過程を説明し、実際にストリング操作関数の強大な機能を体験できるように案内いたします。

1. 以下ではどのようにレジスタから一つのストリングを読み取る、及び書き込むかを説明します。

まず、一個のマクロを新規作成してください：



マクロエディターで下記内容を編集します：

```

1 macro_command main()
2
3
4 char str[20]
5
6 StringGet(str[0], "Local HMI", LW, 0, 20)
7 StringSet(str[0], "Local HMI", LW, 50, 20)
8
9 end macro_command

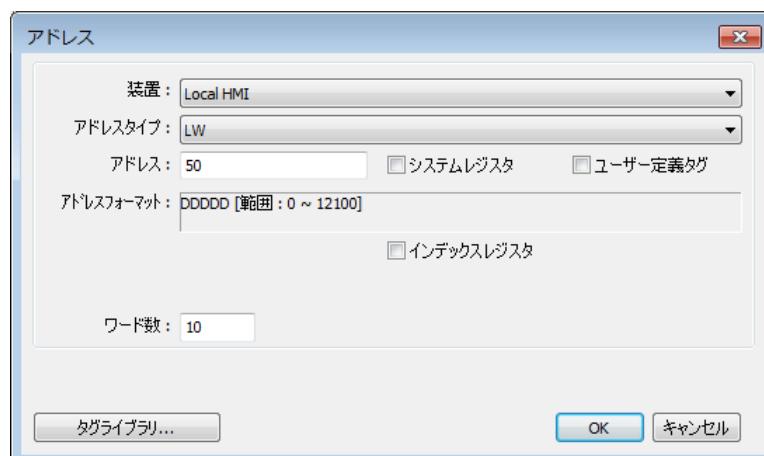
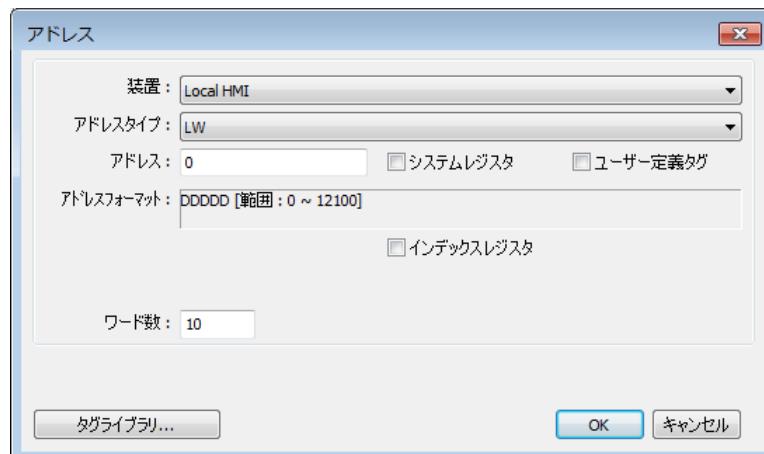
```

本マクロの目的は、StringGet 関数を利用し、レジスタの中から一つのストリングを読み取ってキャラクタ配列 str に入れ、そして StringSet で str の内容を出力します。

続いてプロジェクトの第 10 ウィンドウにそれぞれ[アスキーオブジェクト]と[ファンクションキー]

Fn オブジェクトを入れ、オブジェクトの設定内容は下図をご参照してください。[ファンクションキー]オブジェクトは macro_0 を実行することに用いられます。

[アスキイ]オブジェクト 



[ファンクション]オブジェクト 

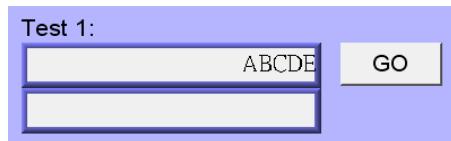
マクロをトリガーする マクロ : [ID:000] macro_0

最後に完成したプロジェクトをコンパイル  し、オフラインシミュレーション(off-line)

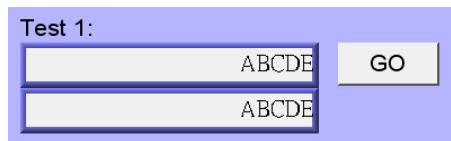
 或いはオンラインシミュレーション(on-line)  し、そしてシミュレーション画面で下記のステップに従って操作します：

- Step 1. ストリングを入力する
- Step 2. "GO"ボタンを押す

マクロの説明



Step 3. 結果を表示する



2. ストリングの初期化を行います。

マクロエディターに下記内容を編集します :

```

1  macro_command main()
2
3
4  char str1[20] = "abcde"
5  char str2[20] = {'a', 'b', 'c', 'd', 'e'}
6
7  StringSet(str1[0], "Local HMI", LW, 0, 20)
8  StringSet(str2[0], "Local HMI", LW, 50, 20)
9
10 end macro_command

```

二重引用符("")に囲まれた内容は一つのストリングを視します。str1 はストリングの形式で初期化し、str2 はキャラクタ配列の形式で初期化します。



ストリングの形式で初期化をする場合、マクロエディターはストリングの末尾に終止符'¥0'をつけ、ストリングの終了を代表しています。StringSet でストリングをレジスタに書き込んだ時、終止符に会ったら、配列の後ろにある内容を書き込むのをやめます。たとえ data count がストリングの長さより大きい数値に設定されても、終止符の前にある内容だけが表示されます。

キャラクタ配列の形式で初期化する場合、配列の内容は一つのストリングに視されないので、終止符'¥0'をつけることもないです。レジスタに書き込むキャラクタ数は data count に設定した数値で決定します。

3. これはシンプルなアカウントパスワードログインページです。

マクロエディターで下記内容を編集します(マクロコマンド[ID:001]macro_1) :

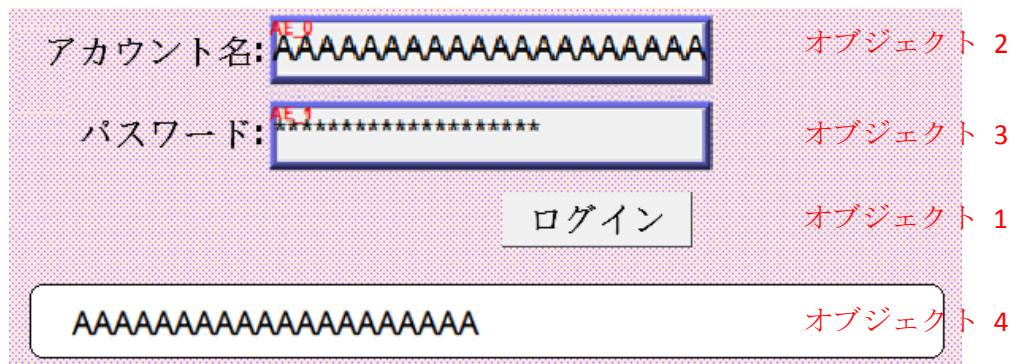
```
1 macro_command main()
2
3
4     char name[20] = "admin"
5     char password[20] = "123456"
6     char name_input[20], password_input[20]
7     char message_success[40] = "Success! Access Accepted."
8     char message_fail[40] = "Fail! Access Denied."
9     char message_clear[40]
10    bool name_match = false, password_match = false
11
12    StringGet(name_input[0], "Local HMI", LW, 0, 20)
13    StringGet(password_input[0], "Local HMI", LW, 50, 20)
14
15    name_match = StringCompare(name_input[0], name[0])
16    password_match = StringCompare(password_input[0], password[0])
17
18    FILL(message_clear[0], 0x20, 40) //FILL with white space
19    StringSet(message_clear[0], "Local HMI", LW, 100, 40)
20
21    if (name_match == true and password_match == true) then
22        StringSet(message_success[0], "Local HMI", LW, 100, 40)
23    else
24        StringSet(message_fail[0], "Local HMI", LW, 100, 40)
25    end if
26
27 end macro_command
```

本マクロは[StringGet]でユーザーが入力したアカウントパスワードのストリングを取り、それぞれ name_input と password_input との 2 個の配列に入れます。続いて[StringCompare]でアカウントパスワードを合わせ、もしアカウントが合っていたら、name_match を true に設定し、パスワードが合っていたら、password_match を true に設定します。最後に name_match と password_match の値を検査し、両者が同じ true の場合、ログインに成功したと示し、"Success! Access Accepted."ストリングがプリントアウトされます。中に一個でも合っていなかったら、"Fail! Access Denied."ストリングがプリントアウトされます。

続いてプロジェクトの第 10 ウィンドウにそれぞれ[アスキーキー]  と[ファンクションキー]

 オブジェクトを入れ、オブジェクトの設定内容は下図を参照してください。[ファンクションキー]オブジェクトは macro_1 を実行することに用いられます。

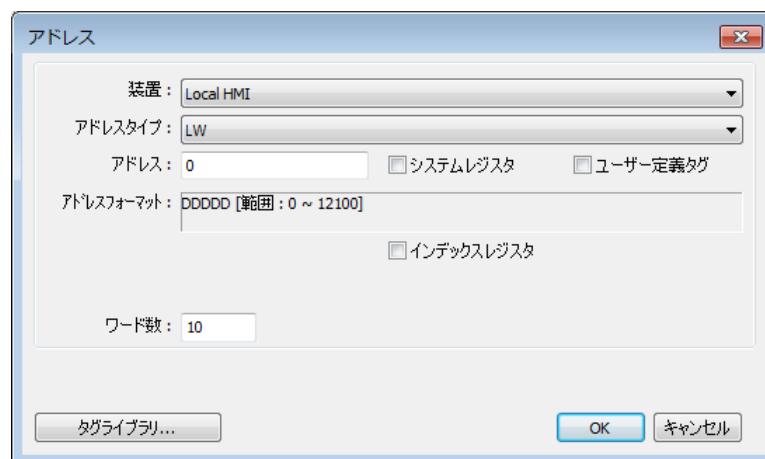
マクロの説明



オブジェクト 1 : [ファンクションキー]オブジェクト 、

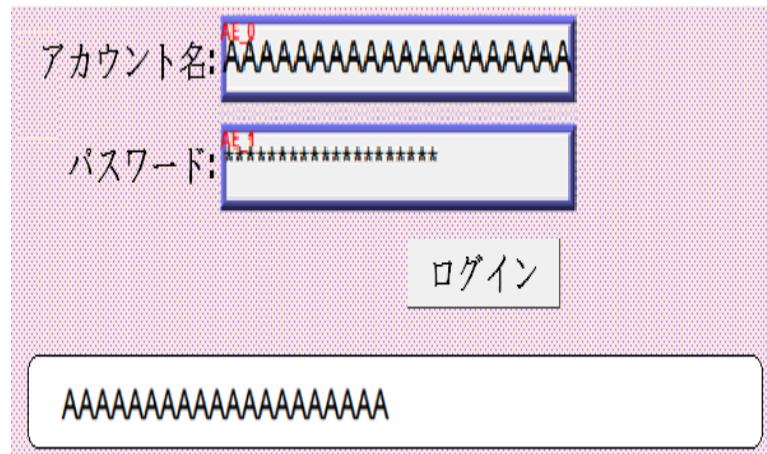
[マクロをトリガーする]をクリックし、マクロコマンド[ID:001]macro_1 を選択します。

オブジェクト 2 : [アスキーオブジェクト]オブジェクト

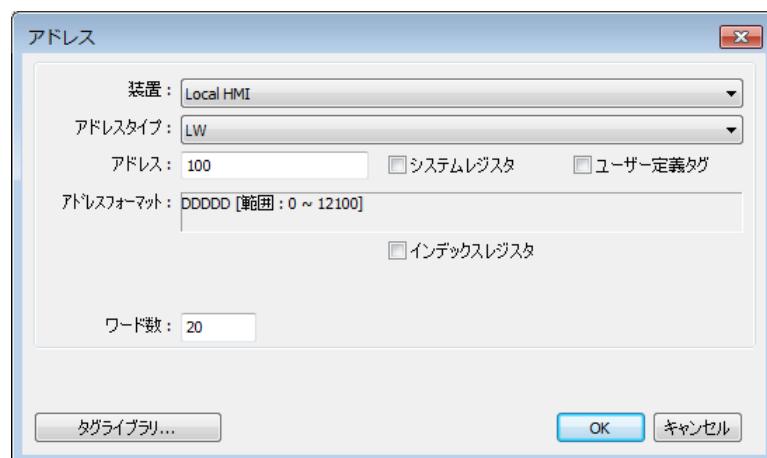


オブジェクト 3 : [アスキーオブジェクト]オブジェクト





オブジェクト 4 : [アスキーオブジェクト] ABC



最後に完成したプロジェクトをコンパイル し、オフラインシミュレーション(off-line) 或いはオンラインシミュレーション(on-line) し、そしてシミュレーション画面で下記のステップに従って操作します：

- 1.** アカウントを入力します。



2. パスワードを入力し、ログインボタンを押します。



3. ログインに成功/失敗したと表示されます。

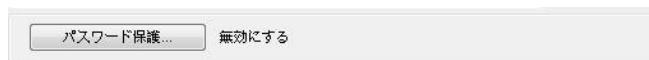
マクロの説明



18.16. マクロパスワード保護

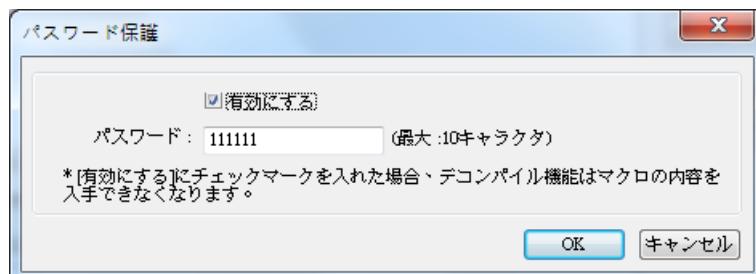
パスワード保護は2種類あります：リストされて全てのマクロを保護、或いは単一のマクロを保護します。

全てのマクロをパスワードで保護する：

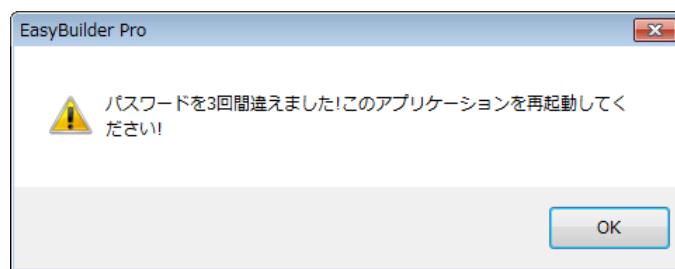


マクロリストウインドウでは[パスワード保護]オプションを提供し、[パスワード保護...]ボタンを押した後、[有効にする]にチェックマークを入れればパスワードを設定できます。パスワードは最大10個のキャラクタを超えないように(ASCIIキャラクタのみサポートし、大文字小文字を区別します。例えば、“a\$#*hFds”を入力できます)。

マクロに[パスワード保護]機能を設定した後、ユーザーがマクロエディターウィンドウをオープンしたい時、まずは正確なパスワードを入力する必要があります。



不正確のパスワードを三回入力すると、改めて EasyBuilder Pro を再起動しなくては、パスワードの入力ができません。



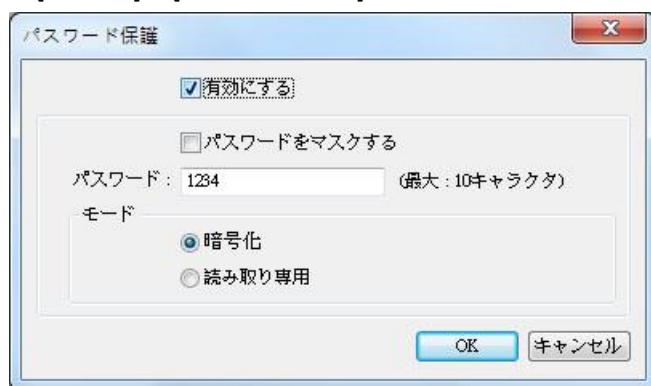
マクロパスワード保護機能を使用すると、デコンパイル EXOB ファイルはマクロの内容を復元できません。マクロには以下の内容が表示されます：

```
macro_command main()  
  
// macro is password protected  
  
end macro_command
```

单一のマクロをパスワードで保護する：

マクロエディターウィンドウでは[パスワード保護]オプションを提供し、[パスワード保護...]ボタンを押した後、[有効にする]にチェックマークを入れればパスワードを設定できます。パスワードは最大 10 個のキャラクタを超えないように(ASCII キャラクタのみサポートし、大文字小文字を区別します。例えば、“a\$#*hFds”を入力できます)。

パスワード保護の方式は[暗号化]と[読み取り専用]に分けられます。



[暗号化]

マクロの内容を暗号化します。マクロリストからマクロを選択してエディターウィンドウに入る前に、パスワードを正しく入力しなければなりません。

同一のマクロパスワードを3回間違ったら、EasyBuilder ProB Pro を再起動のプロンプトが出ます。

[読み取り専用]

ユーザーはマクロの内容を閲覧できるだけで、編集できません。

本モードを選択した場合、マクロリストから直接にマクロエディターに入ることができますが、[パスワード保護]ボタンをクリックしたら、正確なパスワードを入力する必要があります。[暗号化]と同じく、3回の入力制限があります。

マクロリストでは、各マクロが使用した保護モードが表示されます。

マクロの説明



18.17.CANbus アドレスの読み取り/書き込みに変数を使用

CAN Bus 2.0A/2.0B General and SAE J1939 ドライバを使用する場合、当該ドライバには DATA と DATA_Bit との二種類のデータ型があります。データフォーマットの記述は下図に示す通りです。

データ型	記述
DATA	HHHHHHHH: ID B: Byte 位置(1~8) HHHHHHHHBbNN b: Bit 位置(1~8) NN: Bit 数(1~64)
DATA_Bit	HHHHHHHH: ID B: Byte 位置(1~8) HHHHHHHHBb b: Bit 位置(1~8)

ID は 16 進法で表示し、位置と数量は 10 進法で表示するので、使用方法については下記の例を参考してください。

例

変数を使用していない：

```
short f  
GetData(f, "CAN Device", DATA, 4e55108, 1)  
GetData(f, "CAN Device", DATA, 4e65108, 1)
```

変数を使用している：

```
short f  
unsigned int address = 0x4e55108  
GetData(f, "CAN Device", DATA, address, 1)  
address = address + 0x10000// == 0x4e65108  
GetData(f, "CAN Device", DATA, address, 1)
```

注意事項：

1. 変数を Unsigned int に宣言し、それに 16 進法でアドレスを表示します。
2. Unsigned int の長さは 4 bytes で、それに Bb と NN がそれぞれ 1 byte の空間を要ります。変数をアドレスパラメータとして使用し、DATA_Bit タイプのアドレスを読み書きする場合、アドレスフォーマットは HHHHHBb (ID は最大 0xffffffff まで)に変更されます。DATA タイプのアドレスを読み書きする場合、アドレスフォーマットは HHHHBbNN (ID は最大 0xffff まで)に変更されます。

19. HMI を MODBUS 装置に設定する方法

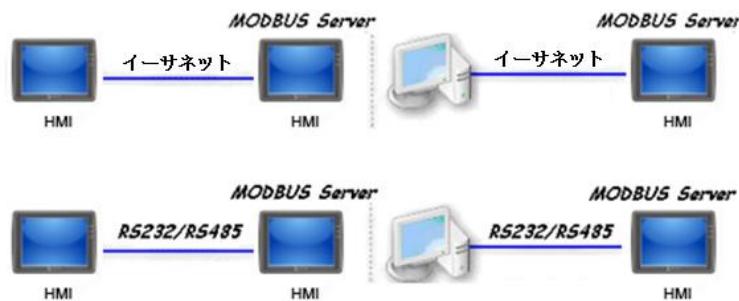
本章では、HMI を MODBUS 装置に設定する方法について説明します。

19.1. 概要	19-2
19.2. MODBUS Server 装置を作成する	19-2
19.3. 一個の MODBUS Server 装置をアクセスする	19-5
19.4. オンラインで MODBUS Server ステーション番号を変更する	19-8
19.5. 各 MODBUS アドレスの説明	19-8

19.1. 概要

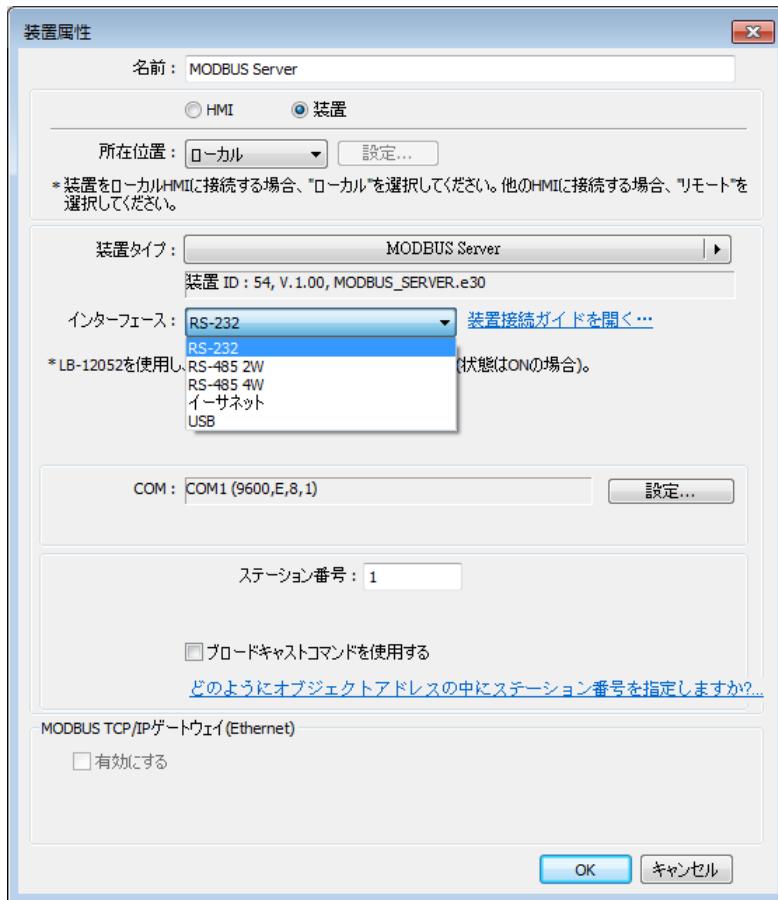
HMI を MODBUS 装置に設定した後、MODBUS プロトコルを通じ、HMI 内のデータを読み取ることができます。

下図では HMI が MODBUS 装置(または MODBUS Server と称す)に設定された状態を示しています。HMI、PC または他の装置は MODBUS プロトコルさえ使用すれば、イーサネット或いは RS-232/RS-485 インターフェースを通すと、HMI 内のデータを読み取る/書き込むことができます。

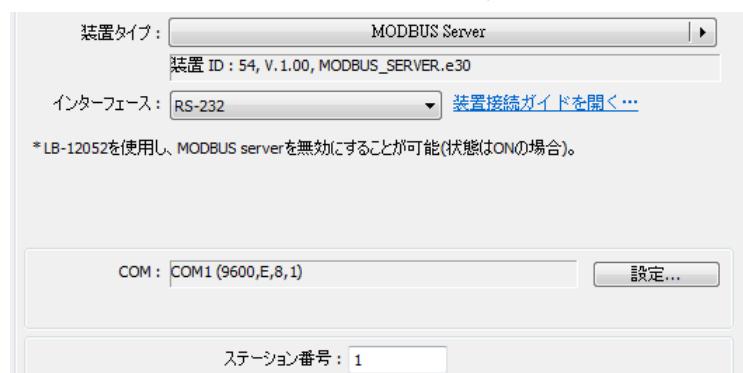


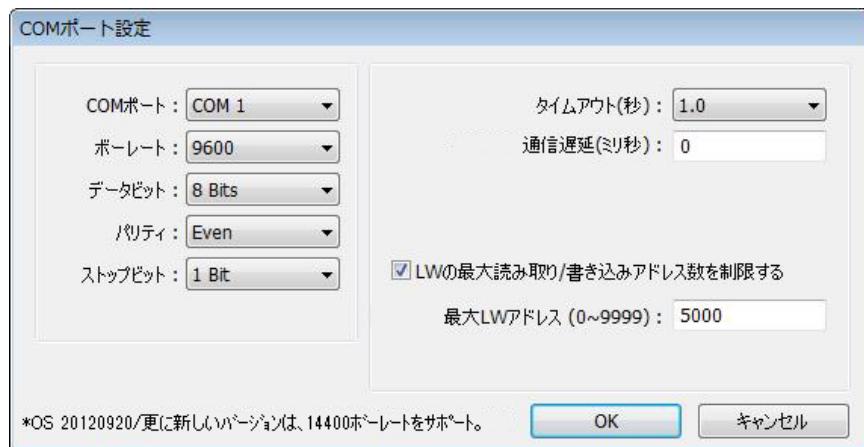
19.2. MODBUS Server 装置を作成する

1. HMI を MODBUS 装置に設定するには、まず[システムパラメータ設定] » [装置リスト]に新しい装置を追加し、装置タイプに MODBUS Server を選択します。選択できる[装置インターフェース]は下図に示す通りです。

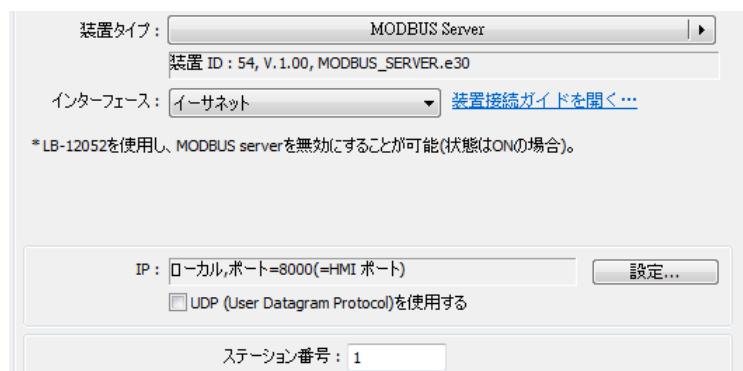


2. インターフェースに[RS-232]或いは[RS-485]を選択する時、使用する[COM](COM1~COM3)を選択し、そして正確な通信パラメータを設定する必要があります。下図に示す通り、この時、MODBUS Server の[ステーション番号]は 1 に設定されます。
[設定]をクリックしたら、[LW の最大読み取り/書き込みアドレス数を制限する]を設定できます。プロジェクトにあるオブジェクトが LW レジスタを使用する時、この範囲を超えたアドレスは MODBUS のクライアントに読み取られ/書き込まれることはありません。





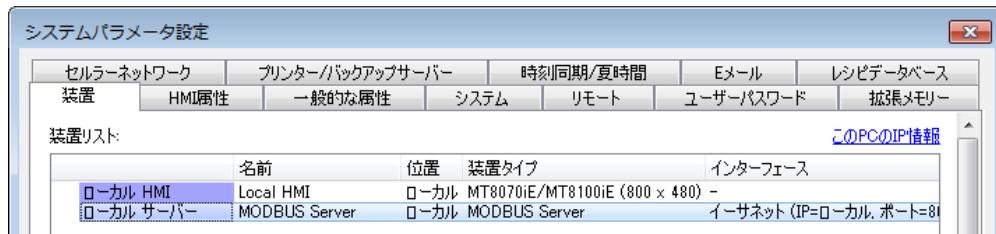
インターフェースに[イーサネット]を選択する場合、[ポート番号]を設定する必要があります。



MODBUS Server と HMI が同じの[ポート番号]を使用する必要があるので、MODBUS Server のポート番号を変更したい場合、[システムパラメータ設定] » [HMI 属性]タブで変更してください。

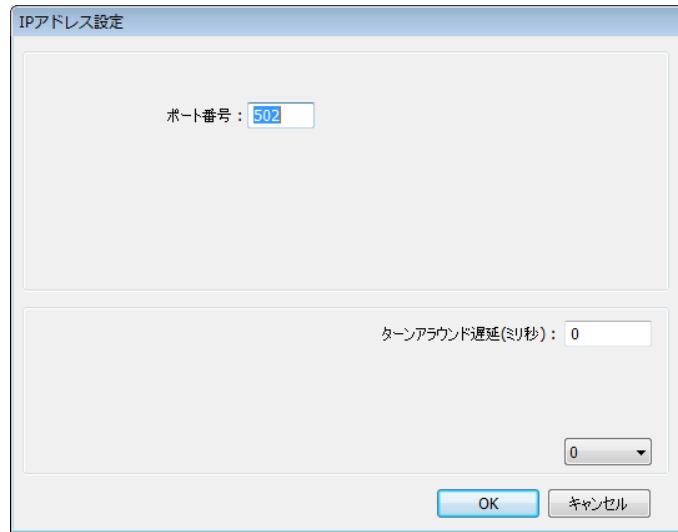


- OK ボタンを押すと、[装置リスト]に新規装置：MODBUS Server が見られます。この時、MODBUS 装置の設定が完成しました。.emtp ファイルのコンパイルを完成し、コンパイルした .exob ファイルを HMI にダウンロードした後、MODBUS プロトコルを通じて HMI 内のデータを読み取り/書き込みできます。



Note

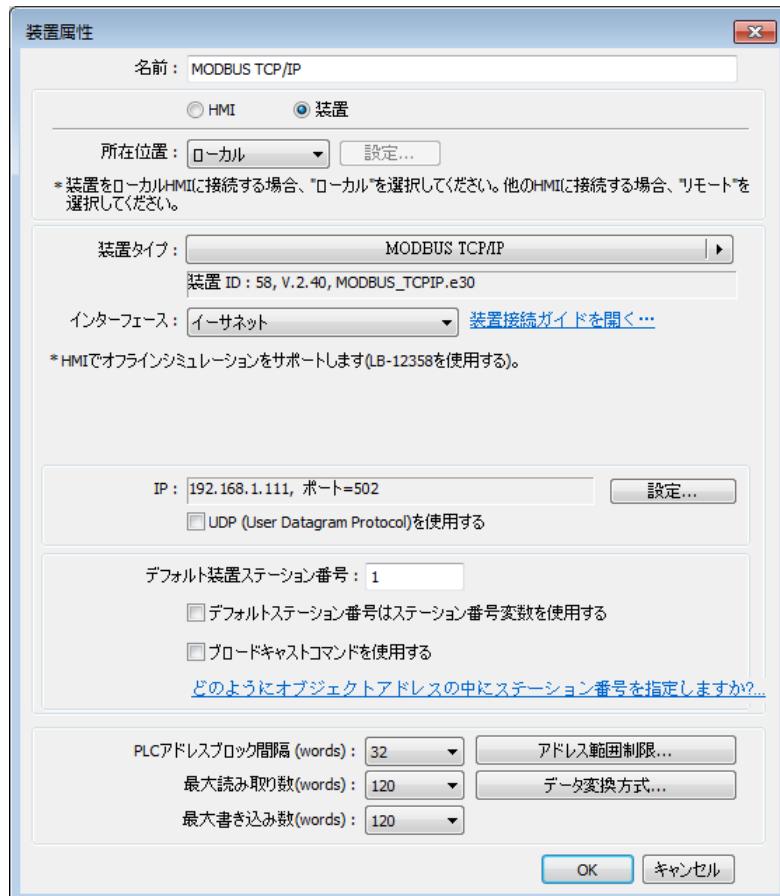
- cMT を使用する場合、インターフェースで[イーサネット]を選択すると、自分でポート番号を入力することができます。



19.3. 一個の MODBUS Server 装置をアクセスする

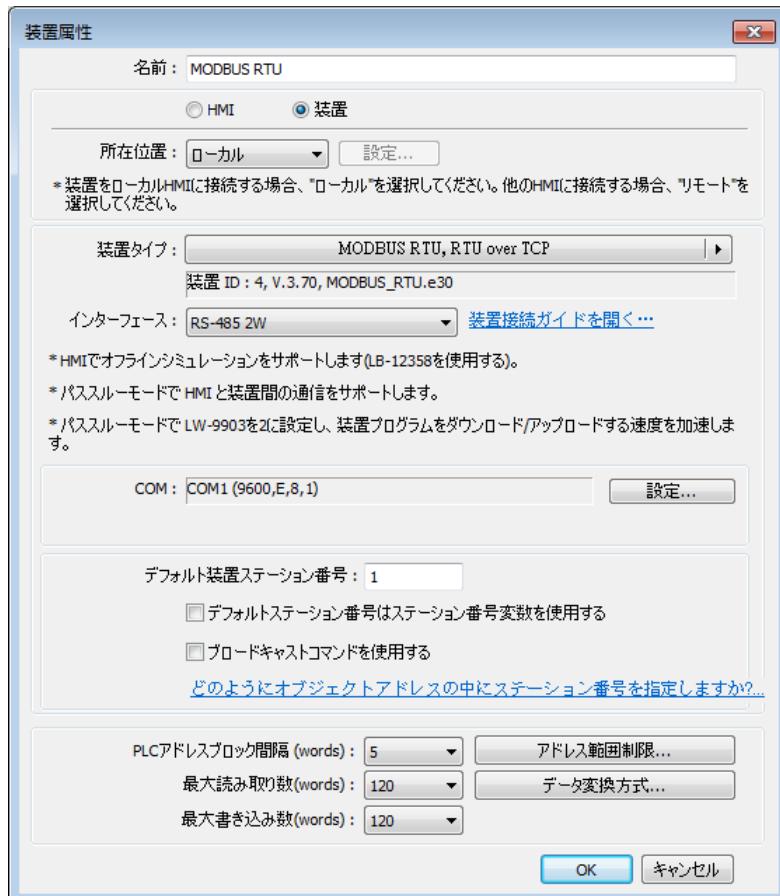
2 台の HMI は MODBUS Client(クライアント)と Server(サーバー)に設定することにより、相互に通信し合うことができます。

- Client 側の装置リストに、一個の新しい装置を追加する必要があります。Client 側で[イーサネット]インターフェースを使用する場合、[装置タイプ]で MODBUS TCP/IP を選択する必要があります、それに正確に[IP アドレス](server 側の所在位置の IP)、[ポート番号]と[ステーション番号]を設定することが必須です。

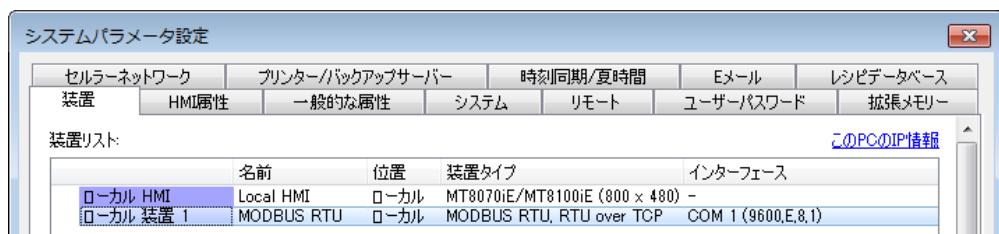


もし Client 側が[RS-232]或いは[RS-485]インターフェースを使用したい場合、[装置タイプ]で MODBUS RTU を選択し、正確に各通信パラメータを設定する必要があります。

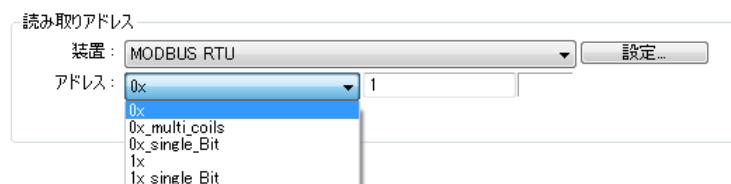
HMI を MODBUS 装置に設定する方法



2. 各設定を完了し OK ボタンを押した後、[装置リスト]に一個の新しい装置“MODBUS RTU”が見られます。



3. 各オブジェクトの設定タブをオープンし、[装置]で MODBUS RTU を選択した後、MODBUS 装置の各読み取り/書き込みアドレスを設定することができます。



この場合、読み取り/書き込みされた装置(Server 側)は HMI だったので、実際の読み取り/書き込み位置は下記のようになります：

0x/1x(1~12800)	LB(0~12799)
3x/4x/5x(1~9999)	LW(0~9998)
3x/4x/5x(10000~65535)	RW(0~55535)

19.4. オンラインで MODBUS Server ステーション番号を変更する

EasyBuilder Pro は下記システムレジスタを提供し、ユーザーにオンラインで MODBUS Server が使用しているステーション番号を変更させます。

LW-9541	MODBUS / ASCII server ステーション番号(COM1)
LW-9542	MODBUS / ASCII server ステーション番号(COM2)
LW-9543	MODBUS / ASCII server ステーション番号(COM3)
LW-9544	MODBUS / ASCII server ステーション番号(イーサネット)

19.5. 各 MODBUS アドレスの説明

EasyBuilder Pro の中に、MODBUS プロトコルのアドレスタイプは 0x、1x、3x、4x、5x、6x、及び 3x_bit、4x_bit などがあります。以下はこれらのアドレスタイプがそれぞれ MODBUS プロトコルの中でサポートしている機能コードを説明します。

0x	このアドレスタイプは読み取り、書き込み両方でもできます。装置の出力点を操作することに相当します。本アドレスタイプがビット状態を読み取る時、発信する機能コードは 01H で、ビット状態を書き込む時、発信する機能コードは 05H です。複数のビットアドレスレジスタに書き込む時、発信する機能コードは 0fH です。
1x	読み取り専用のアドレスタイプです。装置の入力点を読み取ることに相当します。ビット状態を読み取る時、発信する機能コードは 02H です。
3x	読み取り専用のアドレスタイプです。装置の読み取り専用データレジスタを読み取ることに相当します。データを読み取る時、発信する機能コードは 04H です。
4x	このアドレスタイプは読み取り、書き込み両方でもできます。装置のデータレジスタを操作することに相当します。データを読み取る時、発信する機能コードは 03H で、データを書き込む時、発信する機能コードは 10H です。
5x	そのアドレスタイプの属性は 4x と同じです。即ち、発信する読み取り/書き込みの機能コードは全く同じです。異なるところは、データはダブルワードである場合、32_bit unsigned フォーマットのデータを 5x と 4x の二種類のアドレスタイプで個別にデータを読み取る時、上

位ワードと下位ワードの位置は逆さになります。もし **4x** アドレスタイプで読み取ったデータは **0x1234** だったら、**5x** アドレスタイプで読み取ったデータは **0x3412** になります。

6x	このアドレスタイプは読み取り、書き込み両方でもできます。ビット状態を読み取る時、発信する機能コードも 03H で、 4x との違いは、データを書き込む時、発信する機能コードは 06H で、即ち単一のレジスタのデータを書き込むのです。
3x_bit	このアドレスタイプがサポートする機能コードは 3x と全く同じです。異なるところは、 3x はデータを読み取り、 3x_bit はデータ内の特定の bit の状態を読み取るのです。
4x_bit	このアドレスタイプがサポートする機能コードは 4x と全く同じです。異なるところは、 4x はデータを読み取り、 4x_bit はデータ内の特定の bit の状態を読み取るのです。
6x_bit	このアドレスタイプがサポートする機能コードは 6x と全く同じです。異なるところは、 6x はデータを読み取り、 6x_bit はデータ内の特定の bit の状態を読み取るのです。



さらなる情報は《37 MODBUS TCP / IP ゲートウェイ機能》をご参照ください。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

20. バーコードスキャナー を使用する方法

本章では、バーコードスキャナーを使用する方法及び接続する手順について説明します。

20.1. 概要	20-2
20.2. バーコードスキャナーを接続する手順	20-2

20.1. 概要

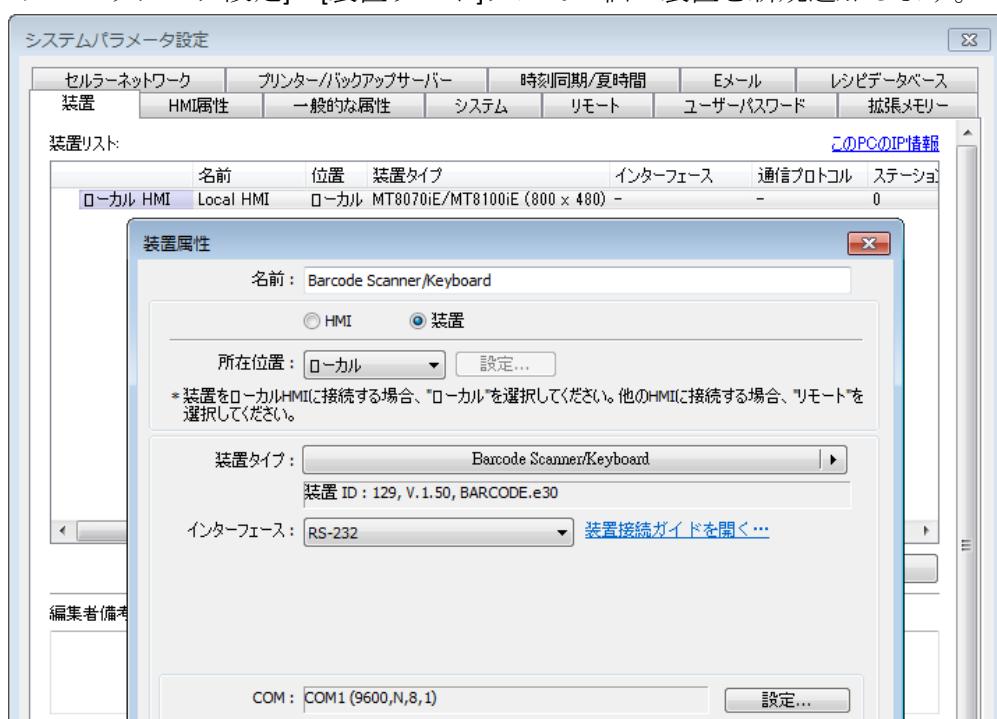
HMI は、下記通信ポートでバーコードスキャナーを接続するのをサポートしています：

- USB
- COM Port

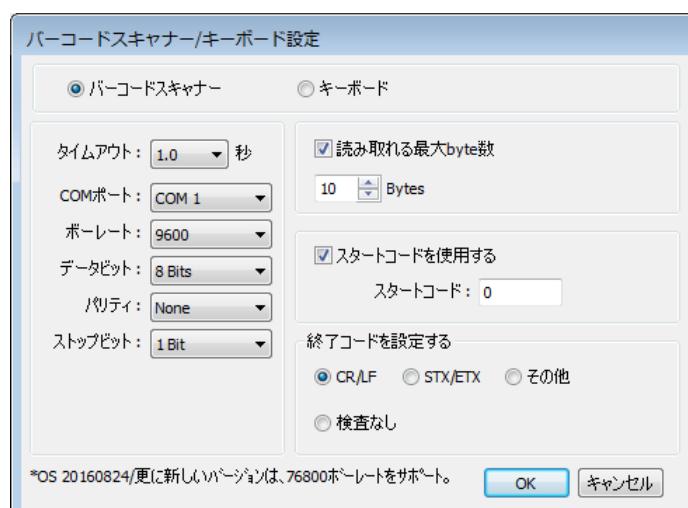
バーコードスキャナーを接続したい場合、まず、装置リストに一個の装置を新規追加してください。

20.2. バーコードスキャナーを接続する手順

1. [システムパラメータ設定] » [装置リスト]タブで一個の装置を新規追加します。



2. [設定]ボタンを押し、[バーコードスキャナー/キーボード設定]をします。



設定	記述
タイムアウト	[バーコードスキャナー]を選択した場合、もしその装置の読み取り速度がより遅くなると、タイムアウトの時間を長くし、完全なデータが読み取れるように設定することができます。 [キーボード]を選択した場合、キーボードを通じて有効な時間範囲を入力でき、システムはデータを入力し始めた時から時間を起算し始めます。
COM ポート ボーレート データビット パリティ ストップビット	COM ポートを選択すると、正確にバーコードスキャナーの通信パラメータを設定する必要があります。 USB インターフェースを選択する場合、通信パラメータを設定する必要がありません。
読み取れる最大 byte 数	本項にチェックマークを入れると、バーコードスキャナーが読み取る byte 数を制限でき、装置がデータを読み取りすぎることを避けます。本項の設定値範囲は 10~512 です。 注意：実際に読み取った byte 数が設定値を超えると、読み取れなくなります。
スタートコードを使用する	本項にチェックマークを入れると、バーコードスキャナーが読み取った一番目のデータがスタートコードと一致する場合のみ、合法的なデータと承認されます。さもなければ、読み取ったデータが無視されます。 スタートコードはバーコードスキャナーに対応するアドレスに保存されません。 例えばスタートコードは 255(0xff)、それに読み取ったデータは： 0xff 0x34 0x39 0x31 0x32 0x30 0x30 0x34 0x37 実際、バーコードスキャナーに対応するアドレスに保存されているデータは： 0x34 0x39 0x31 0x32 0x30 0x30 0x34 0x37
終了コードを設定する	終了コードでデータの末尾を示します。終了コードを読み取った場合、完全なデータが読み取れたと示しています。
CR/LF	0x0a または 0x0d、両方とも終了コードです。
STX/ETX	0x02 または 0x03、両方とも終了コードです。
その他	ユーザー自らでデータの終了コードを設定します。

検査なし

本項目を選択すると、HMI は読み取った全てのデータをバーコードスキャナーに対応するアドレスに保存します。

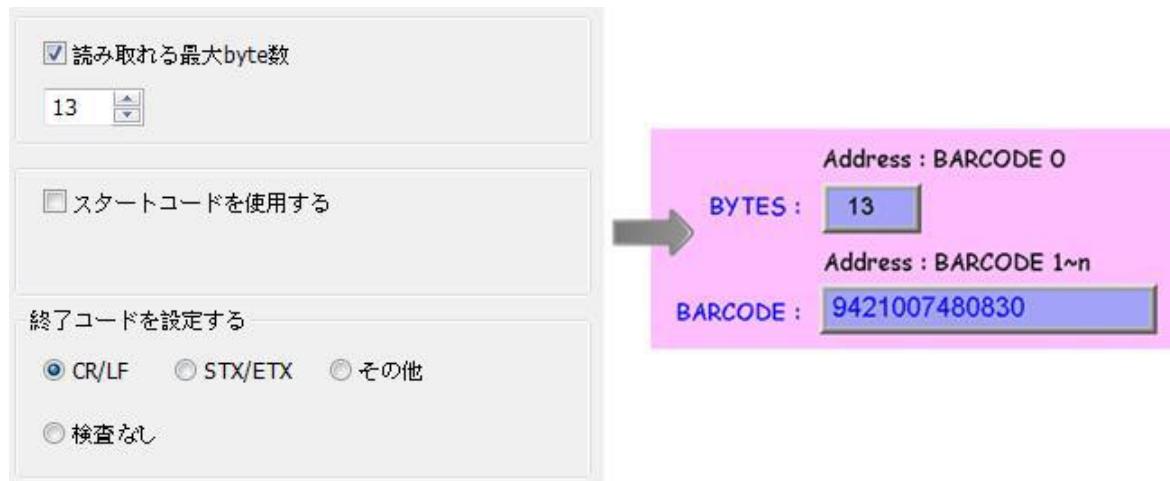
以上の各設定を完成した後、[装置リスト]の中で一個新しいバーコードスキャナー装置が見られます。

この時、オブジェクト[一般的な属性]の設定タブ上の[装置]でバーコードスキャナーを選択でき、また関連のアドレスタイプを使用できます。

アドレスタ イプ	アドレス名	記述
ビット	FLAG	FLAG 0: データの読み取りが完成したかどうかを指示します。データを読み取った時、システムは自動的に FLAG を OFF にし、読み取りに成功した後、ON に設定します。
	RESET	RESET 0:ON に設定した時、BARCODE と RESULT 内のデータをクリアできます。
	CONNECT_STA TUS	CONNECT_STATUS 0: USB インターフェースのバーコードスキャナーを接続したかどうかを指示します。状態は ON の場合、接続されたと表示しています。
ワード	BARCODE	BARCODE 0: 現在、読み取った byte 数を記録します。 BARCODE 1~n: 装置が読み取ったデータを保存します。
	RESULT	RESULT 0: BARCODE の読み取り結果を指示します。各データの意味は以下のとおりです: 0x00: BARCODE の読み取りを待つ。 0x01: BARCODE の読み取りに成功した。 0x02: 不正な BARCODE フォーマット。 0x03: [読み取れる byte 数]を使用する時、読み取ったデータの長さが設定した長さを超えた。 0x04: [スタートコードを使用する]を使用する時、読み取ったデータが設定値に合っていなかった。 0x05: [終了コードを設定する]を使用する時、読み取ったデータが設定値に合っていなかった。

例 1

仮に、現在バーコードスキャナーの設定は下図の通りで、それに読み取ったバーコードは 9421007480830 の場合、図面での数値オブジェクト(BYTES)のアドレスは BARCODE 0 のことを表示し、ASCII オブジェクト(BARCODE)のアドレスは BARCODE 1~n を表示しています。



この時、バーコードスキャナー装置に対応するアドレスが保存するデータは以下のとおりです：

**バーコードスキャナー データ
に対応するアドレス**

13 bytes(十進数)

但し、実際、アドレスに保存したデータは 14

BARCODE 0 bytes=7 words

即ち、読み取った byte 数は奇数の場合、システム
は自動的に一 byte のデータを加えます(0x00)。

BARCODE 1 3439(HEX)

BARCODE 2 3132(HEX)

BARCODE 3 3030(HEX)

BARCODE 4 3437(HEX)

BARCODE 5 3038(HEX)

BARCODE 6 3338(HEX)

BARCODE 7 0030(HEX)

 **Note**

- 一台の HMI は一台の USB バーコードスキャナーのみ接続できます。プロジェクトの装置リストに USB バーコードスキャナーを含む場合、システムレジスタ LB-9064[USB バーコードスキャナー装置を使用する(キーボード機能を閉じる)(状態は ON の場合)]は自動的に ON に設定されます。もしこの時に USB キーボードの機能を復帰し、及び USB バーコードスキャナーを一時停止したいなら、LB-9064 を OFF に設定することができます。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

21. イーサネット通信及び マルチ接続

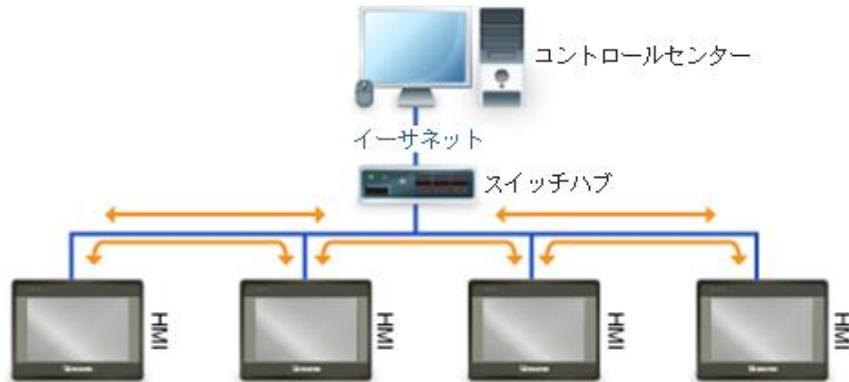
本章では、イーサネットを通じて、多種の装置を接続する方法について説明します。

21.1. 概要	21-2
21.2. HMI と HMI の間の通信	21-2
21.3. PC と HMI の間の通信	21-3
21.4. 他の HMI に接続している PLC をコントロール.....	21-4

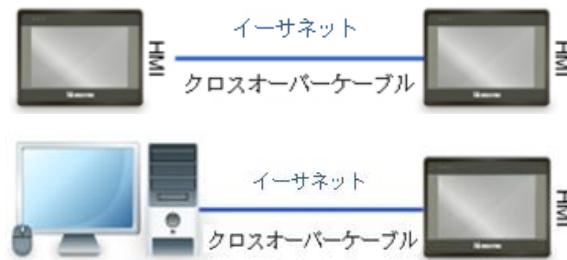
21.1. 概要

イーサネットの通信方式は二種類に分けられます：

- RJ45ストレートケーブルとハブを使用する。



- RJ45クロスケーブルを使用し、ハブを使用しません。但し、一対一の通信状況のみ使用できる(HMI対HMI、またはPC対HMI)。



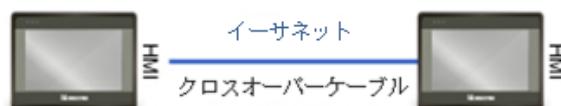
イーサネット通信を通じ、システムは下記三種類のデータ交換方式を提供します：

- HMIとHMI器の間の通信。
- PCとHMIの間の通信。
- 他のHMIに接続しているPLCをコントロール。

21.2. HMIとHMIの間の通信

HMIの間の通信は、[システムパラメータ設定]で一台の遠隔地にあるHMI(リモートHMI)を新規作成すればいいです。

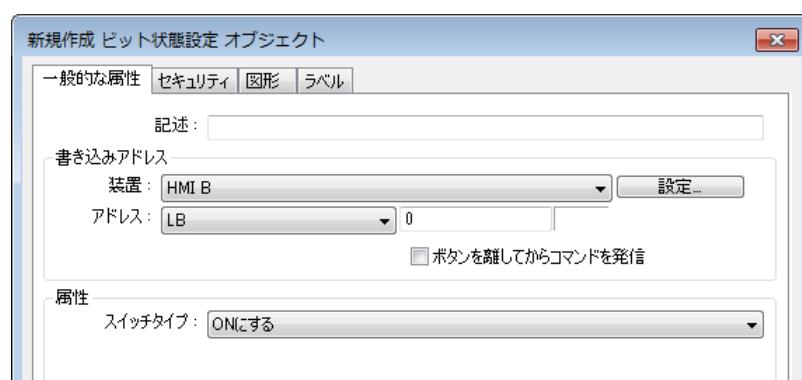
2台のHMIの間での通信を例とし(HMI AとHMI B)、仮にHMI Aはビット状態設定オブジェクトでHMI Bの[LB-0]アドレス内容をコントロールしたい場合、HMI Aに使用するプロジェクトの設定手順は以下のとおりです。



1. 各 HMI の IP アドレスを設定します。HMI A: 192.168.1.1、HMI B: 192.168.1.2 に設定したとします。
2. [システムパラメータ設定] » [装置リスト]で、一台のリモート HMI を新規追加します。即ち HMI B(IP: 192.168.1.2) あります。



3. 一個のビット状態設定オブジェクトを設定し、[装置]の中で“HMI B”を選択すると、リモート HMI のアドレスをコントロールできます。

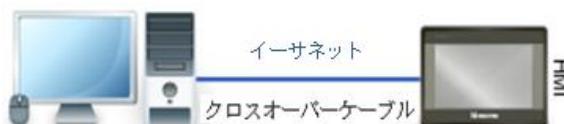


Note

- 一台の HMI は同時に最大 64 台の異なる HMI からのリクエストを処理できます。
- 一台の cMT-SVR は同時に最大 32 台の異なる HMI からのリクエストを処理できます。

21.3. PC と HMI の間の通信

オンラインシミュレーション機能を通し、PC はイーサネットを通じて HMI 内のデータを取り、それを PC に保存することができます。仮に PC が通信したい装置は 2 台の HMI(HMI A と HMI B)の場合、PC 側で使用するプロジェクトの設定手順は以下のとおりです。



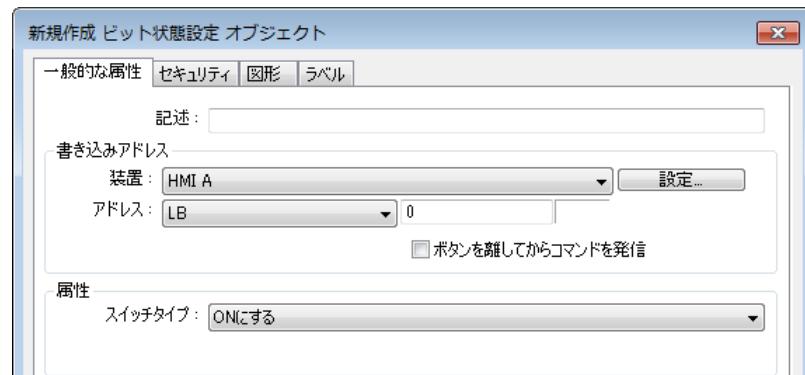
1. 各 HMI の IP アドレスを設定します。HMI A: 192.168.1.1、HMI B: 192.168.1.2 に設定したとします。

イーサネット通信及びマルチ接続

2. [システムパラメータ設定] » [装置リスト]で、2台のリモートHMIを新規追加します。即ちHMI A(IP: 192.168.1.1)と、HMI B(IP: 192.168.1.2)であります。



3. 一個のビット状態設定オブジェクトを設定し、[装置]の中で“HMI A”を選択すると、リモートHMIのアドレスをコントロールできます。それに同様な方法でHMI Bを設定します。

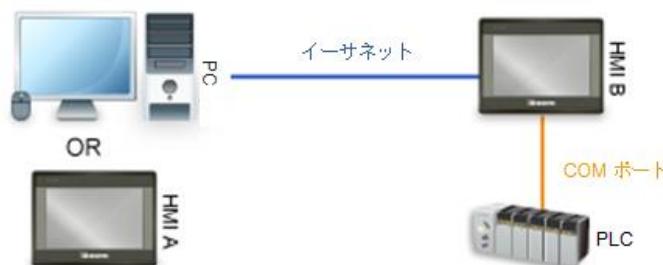


Note

- 一台のPCは同時に最大64台の遠隔地にあるHMIをコントロールできます。
- 上記の例と同じように、HMIもPCをコントロールできます。この場合、PCをもう一台のHMIに視せばいいです。つまり、HMI A / HMI Bが使用するプロジェクトの中にリモートHMIを新規追加し、そしてこのリモートHMIのIPアドレスをPCのIPアドレスと一致するよう設定します。

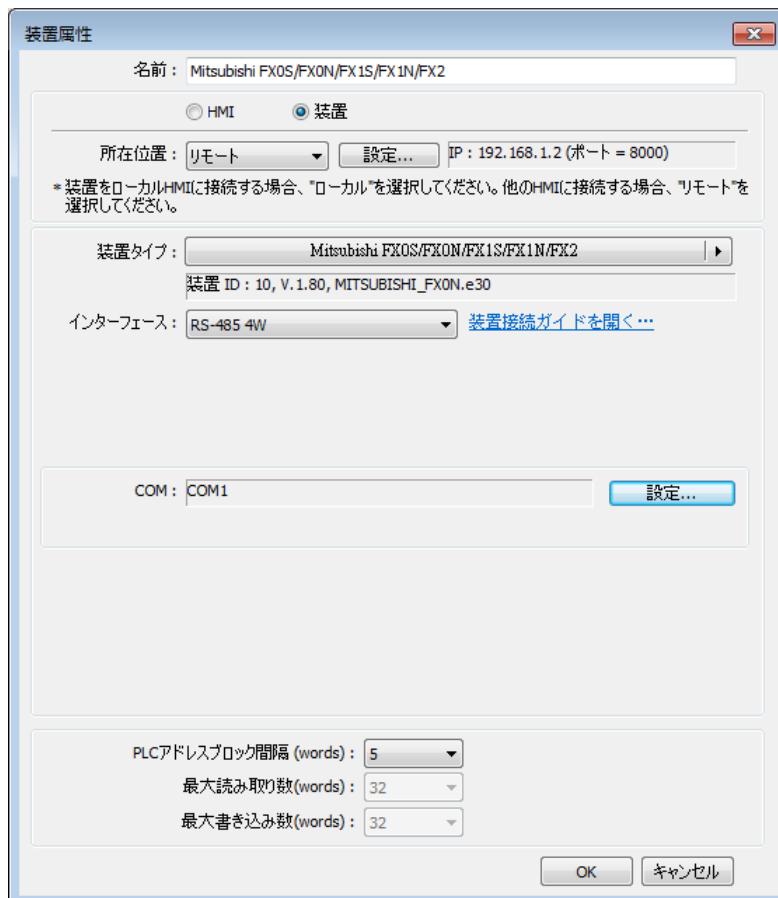
21.4. 他のHMIに接続しているPLCをコントロール

イーサネットを通じて、PCまたはHMIは他のHMIに接続しているPLCを遠隔操作できます。仮に、現在は一台のPLCがHMI BのCOM 1に接続していて、PC或いはHMI AがこのPLC内のデータを読み取りたい場合、PC側、或いはHMI A上で使用するプロジェクトの設定手順は以下のとおりです。

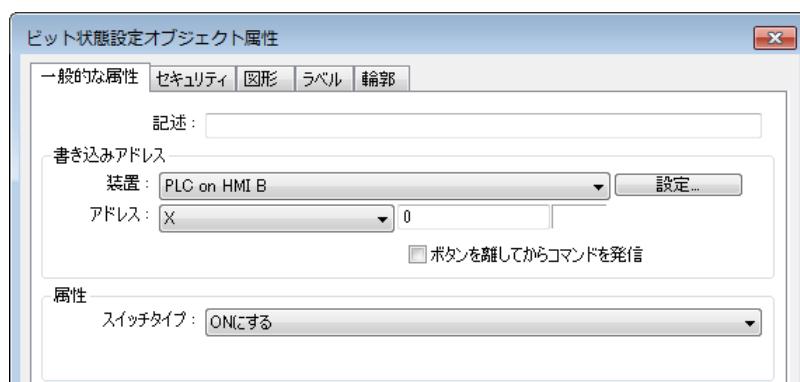


21.4.1. eMT/ iE / XE / mTV / iP シリーズの設定方法

1. HMI B の IP アドレスを設定します。HMI B: 192.168.1.2 に設定します。
2. [システムパラメータ設定] » [装置リスト]で、一台のリモート PLC を新規追加します。装置を“装置 on HMI B”にし、そして正確に PLC の関連通信パラメータを設定します。この PLC は HMI B に接続しているので、IP アドレスを HMI B(IP: 192.168.1.2)と一致するように設定します。



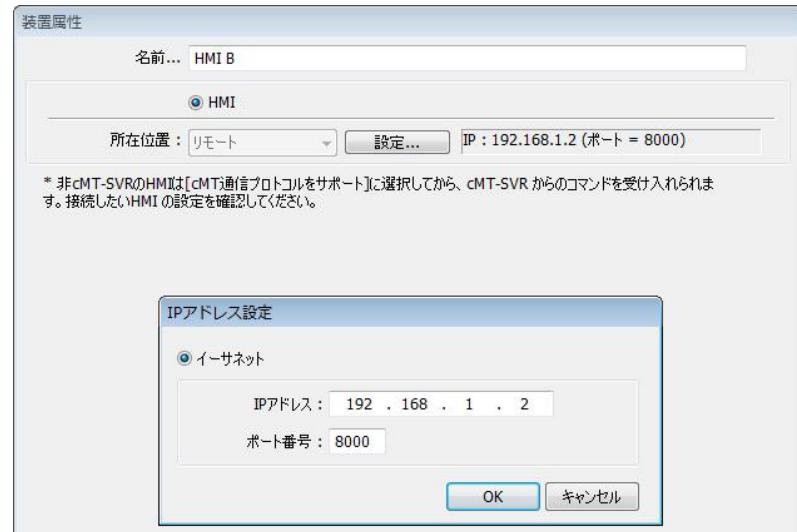
3. 一個のビット状態設定オブジェクトを設定し、[装置]の中で“装置 on HMI B”を選択すれば、HMI B 上の装置をコントロールできます。



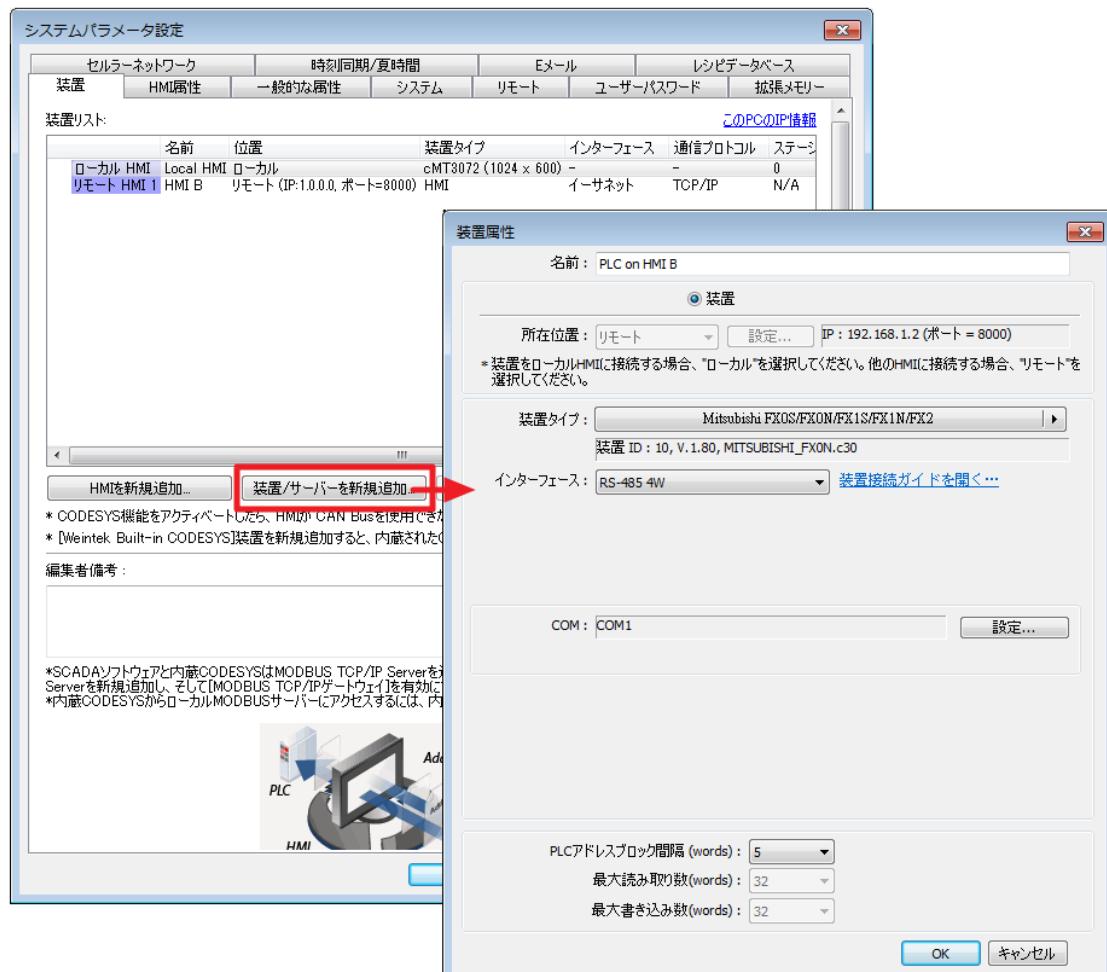
21.4.2. cMT シリーズの設定方法

1. HMI B の IP アドレスを設定します。HMI B: 192.168.1.2 に設定します。

2. [システムパラメータ設定] » [装置リスト]で[HMI を新規追加]を選択し、そして HMI B の IP アドレスを設定します。仮に HMI B: 192.168.1.2 に設定したとします。

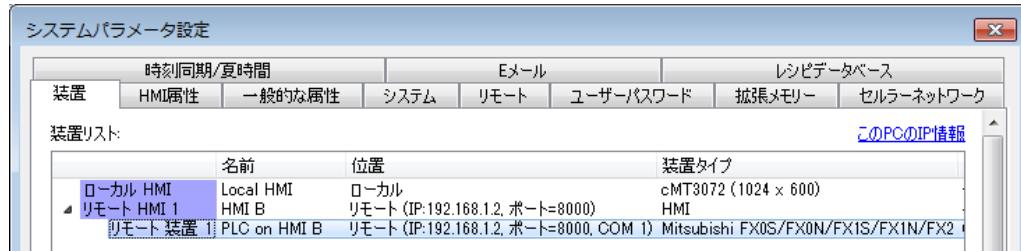


3. HMI B の下に[装置/サーバーを新規追加]をクリックし、一台のリモート装置を新規追加し、装置を“**壮途 on HMI B**”にし、そして正確に装置の関連通信パラメータを設定します。

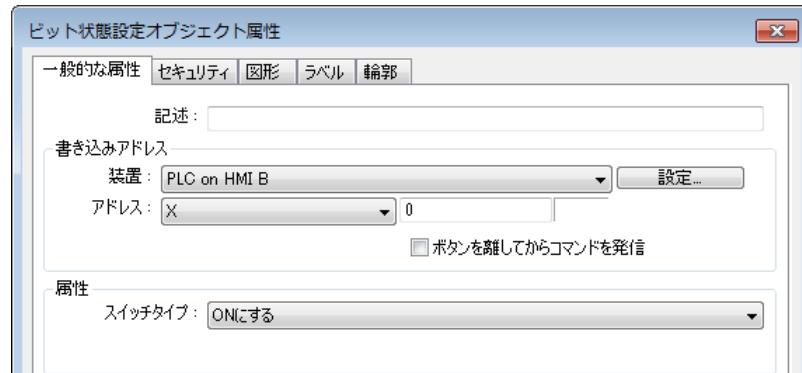


4. 作成が完了後、一台のリモート装置がリモート HMI の下に現れたのが見られ、ローカル HMI

は HMI A、リモート HMI 1 は HMI B、リモート装置 1 は HMI B に接続している PLC のことであります。



5. 一個のビット状態設定オブジェクトを設定し、[装置]の中に装置 “on HMI B”を選択すれば、リモート HMI B に接続している装置をコントロールできます。



Note

- cMT シリーズのリモート HMI は eMT/iE/XE/mTV の場合、[システムパラメータ設定] » [HMI 属性]で[iE/XE/eMT/mTV HMI 通信プロトコルと EasyWatch をサポート]にチェックマークを入れる必要があります。同様に、cMT 以外のシリーズのプロジェクトの[システムパラメータ設定] » [HMI 属性]にも[cMT 通信プロジェクトをサポート]にチェックマークを入れた場合のみ、cMT シリーズ HMI と通信できます。

22. レジスタ

本章では、各種類のレジスタについて説明します。

22.1. 概要	22-3
22.2. ローカル HMI のアドレス範囲	22-4
22.2.1. ビットアドレス	22-4
22.2.2. ワードアドレス	22-5
22.3. システムレジスタ	22-5
22.3.1. HMI 時間	22-5
22.3.2. HMI 操作	22-7
22.3.3. タッチ位置	22-9
22.3.4. ローカル HMI ネットワーク情報	22-9
22.3.5. プロジェクト情報	22-12
22.3.6. 保存領域の管理	22-13
22.3.7. レシピ及び拡張メモリー	22-14
22.3.8. データサンプリング	22-15
22.3.9. イベントログ	22-16
22.3.10. ステーション番号変数	22-19
22.3.11. インデックスレジスタ	22-21
22.3.12. MODBUS Server 通信	22-23
22.3.13. 通信パラメータの設定	22-25
22.3.14. 装置(COM)との通信状態及びコントロール	22-28
22.3.15. 装置(イーサネット)との通信状態及びコントロール	22-30
22.3.16. 装置(USB)との通信状態及びコントロール	22-39
22.3.17. 装置(CAN Bus)との通信状態及びコントロール	22-40
22.3.18. リモート HMI の通信状態及びコントロール	22-41
22.3.19. リモート装置との通信状態及びコントロール	22-46
22.3.20. ローカル/リモートコントロールの制限	22-49
22.3.21. 通信エラーコード	22-49
22.3.22. ドライバ ID	22-52
22.3.23. DLT645 コントローラ	22-52
22.3.24. [Device No Response] ウィンドウのコントロール	22-52
22.3.25. [ファーストセレクション] ウィンドウのコントロール	22-53
22.3.26. EasyAccess	22-54
22.3.27. EasyAccess2.0	22-54

22.3.28.	リモートプリンター/バックアップサーバー	22-55
22.3.29.	パススルー通信の設定	22-56
22.3.30.	VNC のコントロール	22-57
22.3.31.	HMI 識別キーとプロジェクト識別キー	22-58
22.3.32.	USB セキュリティキー	22-58
22.3.33.	ユーザーネームとパスワード	22-59
22.3.34.	マクロ	22-60
22.3.35.	入力オブジェクトの機能	22-61
22.3.36.	時刻同期/夏時間	22-62
22.3.37.	セルラーネットワーク	22-63
22.3.38.	Wi-Fi 設定	22-65
22.3.39.	OPC UA サーバー	22-67
22.3.40.	E メール	22-67
22.3.41.	cMT Viewer と診断器	22-69
22.3.42.	CODESYS 情報	22-71
22.3.43.	他の機能	22-71

22.1. 概要

EasyBuilder Pro 編集ソフトウェアは、幾らかのビットアドレスとワードアドレスのレジスタをシステムの使用に予約されました。これらの予約されたレジスタは、それぞれ異なる用途があり、ここでこれらのシステムレジスタを以下のとおり分類します。

あるシステムレジスタに、[マクロ]或いは[リモート HMI]を通じて数値を書き込んでも、そのレジスタをコントロールできるとは言えません。もしそのシステムレジスタは[マクロ]或いは[リモート HMI]に書き込み及びコントロールされることができれば、"コントロール"と表記されます。

cMT series を選択した場合、システムは PLW と PLB レジスタを提供します。LW / LB と PLW / PLB の違いは、LW / LB とはローカル HMI 上のアドレスで、PLW / PLB のほうは Client 側 (例: cMT-iV5 / iV6, iPad, Android 装置)の上でこれら機能を操作するアドレスです。各 cMT series は多数の Client 側に接続できるので、上記機能のレジスタは Client 側でそれぞれ実行します。

注意 : 各機種がサポートするシステムレジスタは、モデルを選択した後で EasyBuilder Pro 上のアドレスタグライブラリで見つかるかによります。

注意: 各機種がサポートするシステムレジスタは、機種を選択した後、EasyBuilder Pro でのアドレスタグライブラリに表示されます。

アドレスタグライブラリ							
<input checked="" type="radio"/> システムレジスタ	<input type="radio"/> ユーザー定義タグ	<input type="checkbox"/> 分類する	検索				
▼ タグ名	▼ 装置名	アド...	タ...	読み取り/書き込み	電...	記述	
1 LB-9000 : リセット時の状態はON	Local HMI	LB-9000	ビット	読み取り/書き込み	無し		
2 LB-9001 : リセット時の状態はON	Local HMI	LB-9001	ビット	読み取り/書き込み	無し		
3 LB-9002 : リセット時の状態はON	Local HMI	LB-9002	ビット	読み取り/書き込み	無し		
4 LB-9003 : リセット時の状態はON	Local HMI	LB-9003	ビット	読み取り/書き込み	無し		
5 LB-9004 : リセット時の状態はON	Local HMI	LB-9004	ビット	読み取り/書き込み	無し		
6 LB-9005 : リセット時の状態はON	Local HMI	LB-9005	ビット	読み取り/書き込み	無し		
7 LB-9006 : リセット時の状態はON	Local HMI	LB-9006	ビット	読み取り/書き込み	無し		
8 LB-9007 : リセット時の状態はON	Local HMI	LB-9007	ビット	読み取り/書き込み	無し		
9 LB-9008 : リセット時の状態はON	Local HMI	LB-9008	ビット	読み取り/書き込み	無し		
1.. LB-9009 : リセット時の状態はON	Local HMI	LB-9009	ビット	読み取り/書き込み	無し		
11 LB-9010 : データ転送書き込み指示	Local HMI	LB-9010	ビット	読み取り専用	-		
1.. LB-9011 : データ転送読み取り指示	Local HMI	LB-9011	ビット	読み取り専用	-		
1.. LB-9012 : データ転送実行指示	Local HMI	LB-9012	ビット	読み取り専用	-		
1.. LB-9016 : リモートHMIをローカルHMIに接続する(状態はONの場合)	Local HMI	LB-9016	ビット	読み取り/書き込み	無し		
1.. LB-9017 : PLCコントロールオブジェクト[ベースウインドウを切り替える]の[...]	Local HMI	LB-9017	ビット	読み取り/書き込み	無し		
1.. LB-9018 : カーソルのコントロール[表示(ON)隠す(OFF)]	Local HMI	LB-9018	ビット	読み取り/書き込み	無し		
1.. LB-9019 : 音声出力機能[無効にする(ON)有効にする(OFF)]	Local HMI	LB-9019	ビット	読み取り/書き込み	有り		
1.. LB-9020 : システム設定ツールバー[表示(ON)隠す(OFF)]	Local HMI	LB-9020	ビット	読み取り/書き込み	無し		
1.. LB-9026 : HMIメモリー内の全てのデータサンプリングファイルを削除する(ONに設定する)	Local HMI	LB-9026	ビット	読み取り/書き込み	無し		
2.. LB-9028 : レシピデータをリセットする(ONに設定する)	Local HMI	LB-9028	ビット	読み取り/書き込み	無し		
2.. LB-9029 : レシピデータをHMIに保存する(ONに設定する)	Local HMI	LB-9029	ビット	読み取り/書き込み	無し		
2.. LB-9030 : COM1通信パラメータを更新する (LW-9550~9554) (ONに...)	Local HMI	LB-9030	ビット	読み取り/書き込み	無し		

22.2. ローカル HMI のアドレス範囲

22.2.1. ビットアドレス

レジスタ	装置タイプ	範囲	フォーマット
ローカルビットアドレス (ユーザー用)	LB	0~8999	DDDDDD
ローカルビットアドレス (システムレジスタ用)	LB	9000 ~ 12799	DDDDDD
Client 側ビットアドレス	PLB	0 ~ 12399	DDDDDD
ローカルワードビットアドレス	LW_Bit	0~1230015	DDDDDDdd DDDDD:アドレス dd:ビットアドレス (00~15)
Client 側ワードビットアドレス	PLW_Bit	0 ~ 1230015	DDDDDDdd DDDDD: アドレス dd: ビットアドレス (00 ~ 15)
レシピレジスタのビットアドレスインデックスオフセット	RBI	0~65535f	DDDDDH DDDDD:アドレス h:ビットアドレス(0~f) LW-9000 をインデック スレジスタにし、RW_Bit にマッピングする
レシピレジスタ RW のビットアドレス	RW_Bit	0~524287f	DDDDDH DDDDD:アドレス h:ビットアドレス(0~f)
レシピレジスタ RW_A のビットアドレス	RW_A_Bit	0~65535f	DDDDDH DDDDD:アドレス h:ビットアドレス(0~f)

22.2.2. ワードアドレス

レジスタ 名	装置タイ プ	範囲	フォーマット
ローカルワー ドアドレス (ユーザー用)	LW	0~8999	DDDD
ローカルワー ドアドレス (システムレジ スタ用)	LW	9000 ~ 12300	DDDD
Client 側ワード アドレス	PLW	0 ~ 12300	DDDD
レシピレジス タ RW_A	RW_A	0 ~ 65535	DDDD
レシピレジス タ RW	RW	0~524287	DDDDDD
レシピレジス タのワードア ドレスインデ ックスオフセ ット	RWI	0~65535	DDDD LW-9000 をインデック スレジスタにし、RW に マッピングする
拡張メモリー レジスタ	EM0~EM9	0~1073741823	DDDDDDDDDDDD

22.3. システムレジスタ

22.3.1. HMI 時間

アド レス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-11958	時間設定エラー(状態は ON の場合)*注 3	R	R	R
LW-9010	(16bit-BCD):ローカル時間(秒)	R/W	R/C	R/C
LW-9011	(16bit-BCD):ローカル時間(分)	R/W	R/C	R/C
LW-9012	(16bit-BCD):ローカル時間(時)	R/W	R/C	R/C
LW-9013	(16bit-BCD):ローカル時間(日)	R/W	R/C	R/C

LW-9014	(16bit-BCD):ローカル時間(月)	R/W	R/C	R/C
LW-9015	(16bit-BCD):ローカル時間(年)	R	R	R
LW-9016	(16bit-BCD):ローカル時間(週)	R/W	R/C	R/C
LW-9017	(16bit):ローカル時間(秒)	R/W	R/C	R/C
LW-9018	(16bit):ローカル時間(分)	R/W	R/C	R/C
LW-9019	(16bit):ローカル時間(時)	R/W	R/C	R/C
LW-9020	(16bit):ローカル時間(日)	R/W	R/C	R/C
LW-9021	(16bit):ローカル時間(月)	R/W	R/C	R/C
LW-9022	(16bit):ローカル時間(年)*注1	R	R	R
LW-9023	(16bit):ローカル時間(週)*注2	R	R	R
LW-9030	(32bit):システム時間(単位:0.1秒)	R/W	R/C	R/C
LW-9048	(16bit):時間(0:AM, 1:PM)	R/W	R/C	R/C
LW-9049	(16bit):ローカル時間(12時間制)	R	R	R

Note

1. 数値範囲は 2000~2035 です。
2. 数値範囲は 0~6 で、即ち日曜日~土曜日です。
3. LW-9010~LW-9023 を通じて HMI の時間を更新する時、システムは RTC 内の時間を合わせ、時間の修正に成功したかどうかを確認します。成功していなかったら、システムは設定前の時間に復帰し、それに LB-11958 を ON に設定します。PC がシミュレーションしている時に、LW-9010~LW-9023 で時間を修正すれば、無効になります。.

22.3.2. HMI 操作

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカルHMI	マクロ	リモートHMI
LB-9018	カーソルを表示する(ONに設定する)/隠す(OFFに設定する)	R/W	R/C	R/C
LB-9019	音声出力機能を無効にする(ONに設定する)/有効にする(OFFに設定する)]	R/W	R/C	R/C
LB-9020	システム設定バーを表示する(ONに設定する)/隠す(OFF)に設定する	R/W	R/C	R/C
LB-9033	HMI アップロード機能を無効にする(ONに設定する)/有効にする(OFF)に設定する*注1	R/W	R/C	R
LB-9040	バックライトを明るくする(ONに設定する)*注2	W	C	C
LB-9041	バックライトを暗くする(ONに設定する)*注2	W	C	C
LB-9047	HMI を再起動する(LB-9048 状態は ON の時、ONに設定する)	W	C	C
LB-9048	保護機構を再起動する	R/W	R/C	R/C
LB-9062	ハードウェア設定ダイアログボックスをオープンする(ONに設定する)	W	C	C
LB-9063	USB ディスクを接続した時、ダウンロードウインドウのポップアップを無効にする(ONに設定する)/有効にする(OFFに設定する)	R/W	R/C	R/C
LB-9064	USB バーコードスキャナー (キーボード機能を閉じる)を使用する(状態は ON の場合)*注5	R/W	R/C	R
LB-11959	LED インジケーターコントロール*注4	R/W	R/C	R/C
LB-12042	[System information]ダイアログボックスをオープンする/閉じる (ONに設定する/OFFに設定する)	R/W	R/C	R/C
LB-12051	ブザー状態(状態は ON の場合、起動する)	R/W	R/C	R/C
LB-12360	CPU 使用率アラーム(> 95%)*注6	R	R	R
LB-12364	タッチ精度補正モードで[HMI を初期化する]を表示する(ONにする)/隠す(OFFにする)	R/W	R/C	R/C
LB-12665	USB/SD カードのアップロード/ダウンロード ウィンドウをオープンする	R/W	R/C	R/C
LW-9007	(16bit):ハードウェアインデックス番号	R/W	R/C	R/C

LW-9008	(32bit-float):バッテリー電圧*注 3	R	R	R
LW-9025	(16bit):CPU 使用率	R	R	R
LW-9026	(16bit):OS バージョン(年)	R	R	R
LW-9027	(16bit):OS バージョン(月)	R	R	R
LW-9028	(16bit):OS バージョン(日)	R	R	R
LW-9040	(16bit):バックライト明るさ*注 2	R	R	R
LW-9051	(16bit):音量調整(0~100)	R	R	R
LW-9054	(32bit) : HMI モデル ID	R	R	R
LW-9080	(16bit):バックライト省エネ時間(単位:分)	R/W	R/C	R/C
LW-9081	(16bit):スクリーンサーバー時間(単位:分)	R/W	R/C	R/C
LW-9141	(16bit) : HMI ステーション番号	R/W	R/C	R/C
LW-9199	(16bit):外部キーボード配置モード:0(QWERTY), 1(AZERTY)	R/W	R/C	R/C
LW-9350	(16bit):ローカル HMI がまだ処理していないコマンド数	R/W	R/C	R/C
LW-10884	(16words):HMI 名	R	R	R
LW-11155	(32bit):HMI メモリー総量(K bytes)	R/W	R/C	R/C
LW-11157	(32bit):HMI メモリーの空き領域(K bytes)	R	R	R
LW-11159	(16bit):メモリー使用率(x100%)	R	R	R
LW-11382	(16bit) :DIP スイッチ状態 (bit 0 : DIP 1, bit 1 : DIP 2, bit 2 : DIP 3)	R	R	R

Note

1. 設定を変更した時、設定値に更新するため、HMI を再起動する必要があります。
2. LW-9040 は LB-9040~LB-9041 を合わせ、バックライトの明るさを調整できます。明るさは 0~31 です。
3. eMT シリーズのみサポートしています。LW-9008 のバッテリー電圧が 2.80V 未満になった場合、バッテリーを交換するのがお勧めです。
4. mTV と cMT-SVR/cMT-HDM/cMT-FHD の装置を判明することに用いれます。複数の装置がある場合、このアドレスをトリガーし、LED ランプをブリンクさせることができます。
5. LB-9064 で USB バーコードスキャナー装置を有効にする(キーボード機能を閉じる)例は以下のとおりです:
 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。
6. CPU の負荷が連續に 30 秒で 95%以上になると、ON に設定されます。

22.3.3. タッチ位置

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-9041	(16bit):タッチ状態(bit 0 on=スクリーンをタッチしている)	R	R	R
LW-9042	(16bit):タッチしている時、Xの位置	R	R	R
LW-9043	(16bit):タッチしている時、Yの位置	R	R	R
LW-9044	(16bit):離した時、Xの位置	R	R	R
LW-9045	(16bit):離した時、Yの位置	R	R	R



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.4. ローカル HMI ネットワーク情報

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-12041	HMI のイーサネット情報を更新する(DHCP, ゲートウェイ, ネットマスク)(ON に設定する)	R/W	R/C	R/C
LB-12094	イーサネット 1 の設定を更新する(IP, ネットマスク、ゲートウェイ)(ON に設定する)	R/W	R/C	R/C
LB-12095	イーサネット 2 の設定を更新する(IP, ネットマスク)(ON に設定する)	R/W	R/C	R/C
LW-9125	(16bit):HMI のイーサネット 1 が使用するゲートウェイ 0(HMI でのみ有効)	R/W	R/C	R/C
LW-9126	(16bit):HMI のイーサネット 1 が使用するゲートウェイ 1(HMI でのみ有効)	R/W	R/C	R/C
LW-9127	(16bit):HMI のイーサネット 1 が使用するゲートウェイ 2(HMI でのみ有効)	R/W	R/C	R/C

LW-9128	(16bit):HMI のイーサネット 1 が使用するゲートウェイ 3(HMI でのみ有効)	R/W	R/C	R/C
LW-9129	(16bit):HMI のイーサネット 1 が使用する IP0(HMI でのみ有効)	R/W	R/C	R/C
LW-9130	(16bit):HMI のイーサネット 1 が使用する IP1(HMI でのみ有効)	R/W	R/C	R/C
LW-9131	(16bit):HMI のイーサネット 1 が使用する IP2(HMI でのみ有効)	R/W	R/C	R/C
LW-9132	(16bit):HMI のイーサネット 1 が使用する IP3(HMI でのみ有効)	R/W	R/C	R/C
LW-9133	(16bit):イーサネットが使用するポート番号(HMI でのみ有効)	R	R	R
LW-9135	(16bit): HMI のメディアアクセス制御アドレス(MAC)0	R	R	R
LW-9136	(16bit): HMI のメディアアクセス制御アドレス(MAC)1	R	R	R
LW-9137	(16bit): HMI のメディアアクセス制御アドレス(MAC)2	R	R	R
LW-9138	(16bit): HMI のメディアアクセス制御アドレス(MAC)3	R	R	R
LW-9139	(16bit): HMI のメディアアクセス制御アドレス(MAC)4	R	R	R
LW-9140	(16bit): HMI のメディアアクセス制御アドレス(MAC)5	R	R	R
LW-10750	(16bit):HMI イーサネット 1 が使用するネットマスク 0(HMI でのみ有効)	R/W	R/C	R/C
LW-10751	(16bit):HMI イーサネット 1 が使用するネットマスク 1(HMI でのみ有効)	R/W	R/C	R/C
LW-10752	(16bit):HMI イーサネット 1 が使用するネットマスク 2(HMI でのみ有効)	R/W	R/C	R/C
LW-10753	(16bit):HMI イーサネット 1 が使用するネットマスク 3(HMI でのみ有効)	R/W	R/C	R/C
LW-10786	(16bit):HMI イーサネット 2 が使用する IP0(HMI でのみ有効)	R/W	R/C	R/C
LW-10787	(16bit):HMI イーサネット 2 が使用する IP1(HMI でのみ有効)	R/W	R/C	R/C
LW-10788	(16bit):HMI イーサネット 2 が使用する IP2(HMI でのみ有効)	R/W	R/C	R/C
LW-10789	(16bit):HMI イーサネット 2 が使用する IP3(HMI でのみ有効)	R/W	R/C	R/C
LW-10790	(16bit):HMI イーサネット 2 が使用するネットマスク 0(HMI でのみ有効)	R/W	R/C	R/C

LW-10791	(16bit):HMI イーサネット 2 が使用するネットマスク 1(HMI でのみ有効)	R/W	R/C	R/C
LW-10792	(16bit):HMI イーサネット 2 が使用するネットマスク 2(HMI でのみ有効)	R/W	R/C	R/C
LW-10793	(16bit):HMI イーサネット 2 が使用するネットマスク 3(HMI でのみ有効)	R/W	R/C	R/C
LW-10794	(16bit):HMI イーサネット 2 が使用するゲートウェイ 0(HMI でのみ有効)	R/W	R/C	R/C
LW-10795	(16bit):HMI イーサネット 2 が使用するゲートウェイ 1(HMI でのみ有効)	R/W	R/C	R/C
LW-10796	(16bit):HMI イーサネット 2 が使用するゲートウェイ 2(HMI でのみ有効)	R/W	R/C	R/C
LW-10797	(16bit):HMI イーサネット 2 が使用するゲートウェイ 3(HMI でのみ有効)	R/W	R/C	R/C
LW-10798	(16bit):イーサネット 2 のメディアアクセス制御アドレス(MAC)0	R	R	R
LW-10799	(16bit):イーサネット 2 のメディアアクセス制御アドレス(MAC)1	R	R	R
LW-10800	(16bit):イーサネット 2 のメディアアクセス制御アドレス(MAC)2	R	R	R
LW-10801	(16bit):イーサネット 2 のメディアアクセス制御アドレス(MAC)3	R	R	R
LW-10802	(16bit):イーサネット 2 のメディアアクセス制御アドレス(MAC)4	R	R	R
LW-10803	(16bit):イーサネット 2 のメディアアクセス制御アドレス(MAC)5	R	R	R
LW-10804	(16bit):HMI イーサネット 1 のドメインネームシステム(DNS)サーバーIP0	R	R	R
LW-10805	(16bit):HMI イーサネット 1 のドメインネームシステム(DNS)サーバーIP1	R	R	R
LW-10806	(16bit):HMI イーサネット 1 のドメインネームシステム(DNS)サーバーIP2	R	R	R
LW-10807	(16bit):HMI イーサネット 1 のドメインネームシステム(DNS)サーバーIP3	R	R	R
LW-10808	(16bit):HMI イーサネット 2 のドメインネームシステム(DNS)サーバーIP0	R	R	R
LW-10809	(16bit):HMI イーサネット 2 のドメインネームシステム(DNS)サーバーIP1	R	R	R

LW-10810	(16bit):HMI イーサネット 2 のドメインネームシステム(DNS)サーバーIP2	R	R	R
LW-10811	(16bit):HMI イーサネット 2 のドメインネームシステム(DNS)サーバーIP3	R	R	R
LW-10812	(16bit):自動的に IP アドレスを取得する(DHCP=>0:off, 1:on)	R/W	R/C	R/C
LW-10813	(16bit):自動的にイーサネット 2 の IP アドレスを取得する(DHCP=>0:off, 1:on)	R/W	R/C	R/C
LW-10815	(16bit):イーサネット 1 のリンク速度 (0:失敗, 10 (10M), 100 (100M), 1000 (1G))	R	R	R
LW-10816	(16bit):イーサネット 2 のリンク速度 (0:失敗, 10 (10M), 100 (100M), 1000 (1G))	R	R	R

22.3.5. プロジェクト情報

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-9100	(16bit):プロジェクトの名前(16 ワード)	R	R	R
LW-9116	(32bit):プロジェクトのサイズ(単位:byte)	R	R	R
LW-9118	(32bit):プロジェクトのサイズ(単位:K bytes)	R	R	R
LW-9120	(32bit):プロジェクトが使用するコンパイラバージョン	R	R	R
LW-9122	(16bit):プロジェクトをコンパイルした日付(年)	R	R	R
LW-9123	(16bit):プロジェクトをコンパイルした日付(月)	R	R	R
LW-9124	(16bit):プロジェクトをコンパイルした日付(日)	R	R	R
LW-11440	(16bit): プロジェクトをコンパイルした時間 [時] (24 時間制)	R	R	R
LW-11441	(16bit): プロジェクトをコンパイルした時間 [分]	R	R	R
LW-11442	(16bit): プロジェクトをコンパイルした時間 [秒]	R	R	R

22.3.6. 保存領域の管理

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9035	HMI の空き領域不足メッセージ(状態は ON の場合)	R	R	R
LB-9036	SD カード空き領域不足メッセージ(状態は ON の場合)	R	R	R
LB-9037	USB ディスク 1 空き領域不足メッセージ(状態は ON の場合)	R	R	R
LB-9038	USB ディスク 2 空き領域不足メッセージ(状態は ON の場合) *注 1	R	R	R
LB-12048	USB ディスク 1 の状態(状態は ON の時は存在していると示す)	R	R	R
LB-12049	USB ディスク 2 の状態(状態は ON の時は存在していると示す)	R	R	R
LB-12050	SD カードの状態(状態は ON の時は存在していると示す) *注 1	R	R	R
LW-9070	(16bit):空き領域下限メッセージ(Mega bytes)	R	R	R
LW-9071	(16bit):システムが確保する空き領域(Mega bytes)	R	R	R
LW-9072	(32bit):現在、HMI での空き領域(K bytes)	R	R	R
LW-9074	(32bit):現在、SD カード内の空き領域(K bytes)	R	R	R
LW-9076	(32bit):現在、USB ディスク 1 内の空き領域(K bytes)	R	R	R
LW-9078	(32bit):現在、USB ディスク 2 内の空き領域(K bytes) *注 1	R	R	R
LW-11458	(32bit) :HMI 履歴データの合計容量 (K bytes)	R	R	R
LW-11460	(32bit) :現在 HMI 履歴データの残り容量 (K bytes)	R	R	R

 **Note**

1. USB ディスク 2 関連のレジスタは iP / iE シリーズにのみサポートされます。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.7. レシピ及び拡張メモリー

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9028	レシピデータをリセットする(ON に設定する)	W	C	C
LB-9029	レシピデータを HMI に保存する(ON に設定する)	W	C	C
LB-9460	EM0 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9461	EM1 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9462	EM2 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9463	EM3 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9464	EM4 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9465	EM5 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9466	EM6 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9467	EM7 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9468	EM8 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9469	EM9 の保存装置(SD カード)が存在しない(状態は ON の場合)	R	R	R
LB-9470	EM0 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-9471	EM1 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-9472	EM2 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-9473	EM3 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-9474	EM4 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R

LB-9475	EM5 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-9476	EM6 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-9477	EM7 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-9478	EM8 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-9479	EM9 の保存装置(USB ディスク)が存在しない(状態は ON の場合)	R	R	R
LB-12363	リモート HMI からレシピデータベースを更新するのを禁じる(ON に設定する)	R/W	R/C	R/C

22.3.8. データサンプリング

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9025	HMI のメモリー内の日付が一番古いデータサンプリングファイルを削除する(ON に設定する)	W	C	C
LB-9026	HMI のメモリー内の全てのデータサンプリングファイルを削除する(ON に設定する)	W	C	C
LB-9027	HMI のメモリー内のデータサンプリング情報を更新する(ON に設定する)	W	C	C
LB-9034	イベントログとサンプリングデータを HMI、USB ディスク、SD カードに保存する(ON に設定する) ^{*注 1}	W	C	C
LB-11949	SD カード内の日付が一番古いデータサンプリングファイルを削除する(ON に設定する)	W	C	C
LB-11950	SD カード内の全てのデータサンプリングファイルを削除する(ON に設定する)	W	C	C
LB-11951	SD カード内のデータサンプリング情報を更新する(ON に設定する)	W	C	C
LB-11952	USB ディスク 1 内の日付けが一番古いデータサンプリングファイルを削除する(ON に設定する)	W	C	C
LB-11953	USB ディスク 1 内の全てのデータサンプリングファイルを削除する(ON に設定する)	W	C	C

LB-11954	USB ディスク 1 内のデータサンプリング情報を更新する(ON)に設定する)	W	C	C
LB-11955	USB ディスク 2 内の日付けが一番古いデータサンプリングファイルを削除する(ON に設定する) *注 3	W	C	C
LB-11956	USB ディスク 2 内の全てのデータサンプリングファイルを削除する(ON に設定する) *注 3	W	C	C
LB-11957	USB ディスク 2 内のデータサンプリング情報を更新する(ON)に設定する) *注 3	W	C	C
LW-9063	(16bit):HMI のメモリー内に既存しているデータサンプリングファイルの数	R	R	R
LW-9064	(32bit):HMI のメモリー内に既存しているデータサンプリングファイルのサイズ(bytes)	R	R	R
LW-10489	(16bit):SD カード内に既存しているデータサンプリングファイルの数	R	R	R
LW-10490	(32bit):SD カード内に既存しているデータサンプリングファイルのサイズ(bytes)	R	R	R
LW-10492	(16bit):USB ディスク 1 に既存しているデータサンプリングファイルの数	R	R	R
LW-10493	(32bit):USB ディスク 1 に既存しているデータサンプリングファイルのサイズ(bytes)	R	R	R
LW-10495	(16bit):USB ディスク 2 に既存しているデータサンプリングファイルの数 *注 3	R	R	R
LW-10496	(32bit):USB ディスク 2 に既存しているデータサンプリングファイルのサイズ(bytes) *注 3	R	R	R

Note

1. 実行時間の間隔は最短 2 秒ごとです。
2. データサンプリングの関連レジスタを削除/更新することが、PC でシミュレーションをする時には一切作用しません。
3. USB ディスク 2 関連のレジスタは iP / iE シリーズにのみサポートされます。
4. cMT シリーズはシステムレジスタでファイルを削除、またはデータサンプリング関連のファイル情報を閲覧・更新することをサポートしません。しかし、データサンプリング用のコントロールアドレスでファイルを削除することができます。詳細は ch 8 をご参照ください。

22.3.9. イベントログ

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)

		ローカル HMI	マクロ	リモート HMI
LB-9021	現在のイベントログをリセットする(OFF->ON)	W	C	C
LB-9022	HMI のメモリー内の日付が最も古いイベントログを削除する(ON に設定する)	W	C	C
LB-9023	HMI のメモリー内の全てのイベントログファイルを削除する(ON に設定する)	W	C	C
LB-9024	HMI のメモリー内のイベントログ情報を更新する(ON に設定する)	W	C	C
LB-9034	イベントログとサンプリングデータを HMI、USB ディスク、SD カードに保存する(ON に設定する)* 注 2	W	C	C
LB-9042	全てのイベントを承認する(ON に設定する)	W	C	C
LB-9043	承認されていないイベントが存在している(状態は ON の場合)	R	R	R
LB-11940	SD カード内の日付が一番古いイベントログファイルを削除する(ON に設定する)	W	C	C
LB-11941	SD カード内の全てのイベントログファイルを削除する(ON に設定する)	W	C	C
LB-11942	SD カード内のイベントログ情報を更新する(ON に設定する)	W	C	C
LB-11943	USB ディスク 1 内の日付が一番古いイベントログファイルを削除する(ON に設定する)	W	C	C
LB-11944	USB ディスク 1 内の全てのイベントログファイルを削除する(ON に設定する)	W	C	C
LB-11945	USB ディスク 1 内のイベントログ情報を更新する(ON に設定する)	W	C	C
LB-11946	USB ディスク 2 内の日付が一番古いイベントログファイルを削除する(ON に設定する)*注 4	W	C	C
LB-11947	USB ディスク 2 内の全てのイベントログファイルを削除する(ON に設定する)*注 4	W	C	C
LB-11948	USB ディスク 2 内のイベントログ情報を更新する(ON に設定する)*注 4	W	C	C
LB-12024	警告音を無効にする(ON に設定する)	W	C	C
LB-12399	アラームがいずれかのカテゴリーに存在する場合、状態を ON に設定する	R	R	R
LB-12400	カテゴリー 0 にアラームが存在する場合、状態を ON に設定する	R	R	R

LB-12401	カテゴリー1にアラームが存在する場合、状態をONに設定する	R	R	R
LB-12402	カテゴリー2にアラームが存在する場合、状態をONに設定する	R	R	R
LB-12403	カテゴリー3にアラームが存在する場合、状態をONに設定する	R	R	R
LB-12404	カテゴリー4にアラームが存在する場合、状態をONに設定する	R	R	R
LB-12405	カテゴリー5にアラームが存在する場合、状態をONに設定する	R	R	R
LB-12406	カテゴリー6にアラームが存在する場合、状態をONに設定する	R	R	R
LB-12407	カテゴリー7にアラームが存在する場合、状態をONに設定する	R	R	R
LB-12655	カテゴリー255にアラームが存在する場合、状態をONに設定する	R	R	R
LW-9060	(16bit):HMIのメモリーに既存しているイベントログの数	R	R	R
LW-9061	(32bit):HMIのメモリーに既存しているイベントログの大きさ(bytes)	R	R	R
LW-9450	(16bit):イベントログのタイムタグ-秒*注1	R/W	R/C	R/C
LW-9451	(16bit):イベントログのタイムタグ-分*注1	R/W	R/C	R/C
LW-9452	(16bit):イベントログのタイムタグ-時*注1	R/W	R/C	R/C
LW-9453	(16bit):イベントログのタイムタグ-日*注1	R/W	R/C	R/C
LW-9454	(16bit):イベントログのタイムタグ-月*注1	R/W	R/C	R/C
LW-9455	(16bit):イベントログのタイムタグ-年*注1	R/W	R/C	R/C
LW-10480	(16bit):SDカードに既存しているイベントログファイルの数	R	R	R
LW-10481	(32bit):SDカードに既存しているイベントログファイルのサイズ(bytes)	R	R	R
LW-10483	(16bit):USBディスク1に既存しているイベントログファイルの数	R	R	R
LW-10484	(32bit):USBディスク1に既存しているイベントログファイルのサイズ(bytes)	R	R	R
LW-10486	(16bit):USBディスク2に既存しているイベントログファイルの数 *注4	R	R	R
LW-10487	(32bit):USBディスク2に既存しているイベントログファイルの数	R	R	R

	グファイルのサイズ(bytes) *注 4			
LW-11443	(16bit): プッシュ通知アラーム状態 (0: 無; 1: 緑; 2: 黄色; 3: 赤)	R	R	R
LW-11499	アラーム総数	R	R	R
LW-11500	カテゴリー0 のアラーム数	R	R	R
LW-11501	カテゴリー1 のアラーム数	R	R	R
LW-11502	カテゴリー2 のアラーム数	R	R	R
LW-11503	カテゴリー3 のアラーム数	R	R	R
LW-11504	カテゴリー4 のアラーム数	R	R	R
LW-11505	カテゴリー5 のアラーム数	R	R	R
LW-11506	カテゴリー6 のアラーム数	R	R	R
LW-11507	カテゴリー7 のアラーム数	R	R	R
LW-11755	カテゴリー255 のアラーム数	R	R	R

Note

1. LW-9450~LW-9455 を使用し、イベントログの時間を取得したいなら、前もって[システムパラメータ設定] » [一般的な属性]タブで関連した属性を設定してください。
2. 実行時間の間隔は最短 2 秒ごとです。
3. データサンプリングの関連レジスタを削除/更新することが、PC でシミュレーションをする時には一切作用しません。
4. USB ディスク 2 関連のレジスタは iP / iE シリーズにのみサポートされます。
5. cMT シリーズはシステムレジスタでファイルを削除、またはイベントログ関連のファイル情報を閲覧・更新することをサポートしません。しかし、イベントログ用のコントロールアドレスでファイルを削除することができます。詳細は ch 7 をご参照ください。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。本サンプルプロジェクトは LW-9450~LW-9455 を使用し、イベントログの時間を取得する方法を説明します。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.10. ステーション番号変数

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-10000	(16bit):var0-ステーション番号変数(シンタックス:var0#アドレス)	R/W	R/C	R/C
LW-10001	(16bit):var1-ステーション番号変数(シンタックス:var1#アドレス)	R/W	R/C	R/C

LW-10002	(16bit):var2-ステーション番号変数(シンタックス:var2#アドレス)	R/W	R/C	R/C
LW-10003	(16bit):var3-ステーション番号変数(シンタックス:var3#アドレス)	R/W	R/C	R/C
LW-10004	(16bit):var4-ステーション番号変数(シンタックス:var4#アドレス)	R/W	R/C	R/C
LW-10005	(16bit):var5-ステーション番号変数(シンタックス:var5#アドレス)	R/W	R/C	R/C
LW-10006	(16bit):var6-ステーション番号変数(シンタックス:var6#アドレス)	R/W	R/C	R/C
LW-10007	(16bit):var7-ステーション番号変数(シンタックス:var7#アドレス)	R/W	R/C	R/C
LW-10008	(16bit):var8-ステーション番号変数(シンタックス:var8#アドレス)	R/W	R/C	R/C
LW-10009	(16bit):var9-ステーション番号変数(シンタックス:var9#アドレス)	R/W	R/C	R/C
LW-10010	(16bit):var10-ステーション番号変数(シンタックス:var10#アドレス)	R/W	R/C	R/C
LW-10011	(16bit):var11-ステーション番号変数(シンタックス:var11#アドレス)	R/W	R/C	R/C
LW-10012	(16bit):var12-ステーション番号変数(シンタックス:var12#アドレス)	R/W	R/C	R/C
LW-10013	(16bit):var13-ステーション番号変数(シンタックス:var13#アドレス)	R/W	R/C	R/C
LW-10014	(16bit):var14-ステーション番号変数(シンタックス:var14#アドレス)	R/W	R/C	R/C



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.11. インデックスレジスタ

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-9000	(16bit):RWI インデックスレジスタ	R/W	R/C	R/C
LW-9200	(16bit):アドレスインデックスレジスタ 0	R/W	R/C	R/C
LW-9201	(16bit):アドレスインデックスレジスタ 1	R/W	R/C	R/C
LW-9202	(16bit):アドレスインデックスレジスタ 2	R/W	R/C	R/C
LW-9203	(16bit):アドレスインデックスレジスタ 3	R/W	R/C	R/C
LW-9204	(16bit):アドレスインデックスレジスタ 4	R/W	R/C	R/C
LW-9205	(16bit):アドレスインデックスレジスタ 5	R/W	R/C	R/C
LW-9206	(16bit):アドレスインデックスレジスタ 6	R/W	R/C	R/C
LW-9207	(16bit):アドレスインデックスレジスタ 7	R/W	R/C	R/C
LW-9208	(16bit):アドレスインデックスレジスタ 8	R/W	R/C	R/C
LW-9209	(16bit):アドレスインデックスレジスタ 9	R/W	R/C	R/C
LW-9210	(16bit):アドレスインデックスレジスタ 10	R/W	R/C	R/C
LW-9211	(16bit):アドレスインデックスレジスタ 11	R/W	R/C	R/C
LW-9212	(16bit):アドレスインデックスレジスタ 12	R/W	R/C	R/C
LW-9213	(16bit):アドレスインデックスレジスタ 13	R/W	R/C	R/C
LW-9214	(16bit):アドレスインデックスレジスタ 14	R/W	R/C	R/C
LW-9215	(16bit):アドレスインデックスレジスタ 15	R/W	R/C	R/C
LW-9230	(32bit):アドレスインデックスレジスタ 16	R/W	R/C	R/C
LW-9232	(32bit):アドレスインデックスレジスタ 17	R/W	R/C	R/C
LW-9234	(32bit):アドレスインデックスレジスタ 18	R/W	R/C	R/C
LW-9236	(32bit):アドレスインデックスレジスタ 19	R/W	R/C	R/C
LW-9238	(32bit):アドレスインデックスレジスタ 20	R/W	R/C	R/C
LW-9240	(32bit):アドレスインデックスレジスタ 21	R/W	R/C	R/C
LW-9242	(32bit):アドレスインデックスレジスタ 22	R/W	R/C	R/C
LW-9244	(32bit):アドレスインデックスレジスタ 23	R/W	R/C	R/C
LW-9246	(32bit):アドレスインデックスレジスタ 24	R/W	R/C	R/C
LW-9248	(32bit):アドレスインデックスレジスタ 25	R/W	R/C	R/C
LW-9250	(32bit):アドレスインデックスレジスタ 26	R/W	R/C	R/C
LW-9252	(32bit):アドレスインデックスレジスタ 27	R/W	R/C	R/C
LW-9254	(32bit):アドレスインデックスレジスタ 28	R/W	R/C	R/C
LW-9256	(32bit):アドレスインデックスレジスタ 29	R/W	R/C	R/C

LW-9258	(32bit):アドレスインデックスレジスタ 30	R/W	R/C	R/C
LW-9260	(32bit):アドレスインデックスレジスタ 31	R/W	R/C	R/C
PLW-9200	(16bit):プライベートアドレスインデックスレジスタ 0	R/W	無し	無し
PLW-9201	(16bit):プライベートアドレスインデックスレジスタ 1	R/W	無し	無し
PLW-9202	(16bit):プライベートアドレスインデックスレジスタ 2	R/W	無し	無し
PLW-9203	(16bit):プライベートアドレスインデックスレジスタ 3	R/W	無し	無し
PLW-9204	(16bit):プライベートアドレスインデックスレジスタ 4	R/W	無し	無し
PLW-9205	(16bit):プライベートアドレスインデックスレジスタ 5	R/W	無し	無し
PLW-9206	(16bit):プライベートアドレスインデックスレジスタ 6	R/W	無し	無し
PLW-9207	(16bit):プライベートアドレスインデックスレジスタ 7	R/W	無し	無し
PLW-9208	(16bit):プライベートアドレスインデックスレジスタ 8	R/W	無し	無し
PLW-9209	(16bit):プライベートアドレスインデックスレジスタ 9	R/W	無し	無し
PLW-9210	(16bit):プライベートアドレスインデックスレジスタ 10	R/W	無し	無し
PLW-9211	(16bit):プライベートアドレスインデックスレジスタ 11	R/W	無し	無し
PLW-9212	(16bit):プライベートアドレスインデックスレジスタ 12	R/W	無し	無し
PLW-9213	(16bit):プライベートアドレスインデックスレジスタ 13	R/W	無し	無し
PLW-9214	(16bit):プライベートアドレスインデックスレジスタ 14	R/W	無し	無し
PLW-9215	(16bit):プライベートアドレスインデックスレジスタ 15	R/W	無し	無し
PLW-9230	(32bit):プライベートアドレスインデックスレジスタ 16	R/W	無し	無し
PLW-9232	(32bit):プライベートアドレスインデックスレジスタ 17	R/W	無し	無し

PLW-9234	(32bit):プライベートアドレスインデックスレジスタ 18	R/W	無し	無し
PLW-9236	(32bit):プライベートアドレスインデックスレジスタ 19	R/W	無し	無し
PLW-9238	(32bit):プライベートアドレスインデックスレジスタ 20	R/W	無し	無し
PLW-9240	(32bit):プライベートアドレスインデックスレジスタ 21	R/W	無し	無し
PLW-9242	(32bit):プライベートアドレスインデックスレジスタ 22	R/W	無し	無し
PLW-9244	(32bit):プライベートアドレスインデックスレジスタ 23	R/W	無し	無し
PLW-9246	(32bit):プライベートアドレスインデックスレジスタ 24	R/W	無し	無し
PLW-9248	(32bit):プライベートアドレスインデックスレジスタ 25	R/W	無し	無し
PLW-9250	(32bit):プライベートアドレスインデックスレジスタ 26	R/W	無し	無し
PLW-9252	(32bit):プライベートアドレスインデックスレジスタ 27	R/W	無し	無し
PLW-9254	(32bit):プライベートアドレスインデックスレジスタ 28	R/W	無し	無し
PLW-9256	(32bit):プライベートアドレスインデックスレジスタ 29	R/W	無し	無し
PLW-9258	(32bit):プライベートアドレスインデックスレジスタ 30	R/W	無し	無し
PLW-9260	(32bit):プライベートアドレスインデックスレジスタ 31	R/W	無し	無し



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.12. MODBUS Server 通信

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル	マクロ	リモート

		HMI		HMI
LB-9055	MODBUS server (COM1)が合法なリクエストを受け取った(状態は ON の場合)	R	R	R
LB-9056	MODBUS server (COM2)が合法なリクエストを受け取った(状態は ON の場合)	R	R	R
LB-9057	MODBUS server (COM3)が合法なリクエストを受け取った(状態は ON の場合)	R	R	R
LB-9058	MODBUS server(イーサネット)が合法なリクエストを受け取った(状態は ON の場合)	R	R	R
LB-12052	MODBUS server 状態(状態は ON の場合、server 機能を閉じる)	R/W	R/C	R/C
LW-9270	(16bit):リクエストの機能コード-MODBUS server (COM1)	R	R	R
LW-9271	(16bit):リクエストの開始アドレス-MODBUS server (COM1)	R	R	R
LW-9272	(16bit):リクエストのアドレス数量-MODBUS server (COM1)	R	R	R
LW-9275	(16bit):リクエストの機能コード-MODBUS server (COM2)	R	R	R
LW-9276	(16bit):リクエストの開始アドレス-MODBUS server (COM2)	R	R	R
LW-9277	(16bit):リクエストのアドレス数量-MODBUS server (COM2)	R	R	R
LW-9280	(16bit):リクエストの機能コード-MODBUS server (COM3)	R	R	R
LW-9281	(16bit):リクエストの開始アドレス-MODBUS server (COM3)	R	R	R
LW-9282	(16bit):リクエストのアドレス数量-MODBUS server (COM3)	R	R	R
LW-9285	(16bit):リクエストの機能コード-MODBUS server(イーサネット)	R	R	R
LW-9286	(16bit):リクエストの開始アドレス-MODBUS server(イーサネット)	R	R	R
LW-9287	(16bit):リクエストのアドレス数量-MODBUS server(イーサネット)	R	R	R
LW-9288	(16bit):最後の通信エラーコード-MODBUS server(イーサネット)	R	R	R
LW-9541	(16bit):MODBUS /ASCII server ステーション番号	R/W	R/C	R/C

	(COM1)			
LW-9542	(16bit):MODBUS /ASCII server ステーション番号 (COM2)	R/W	R/C	R/C
LW-9543	(16bit):MODBUS /ASCII server ステーション番号 (COM3)	R/W	R/C	R/C
LW-9544	(16bit):MODBUS /ASCII server ステーション番号(イーサネット)	R/W	R/C	R/C
LW-9570	(32bit):受け取ったデータ(bytes)(COM1 MODBUS server)	R	R	R
LW-9572	(32bit):受け取ったデータ(bytes)(COM2 MODBUS server)	R	R	R
LW-9574	(32bit):受け取ったデータ(bytes)(COM3 MODBUS server)	R	R	R
LW-9576	(32bit):受け取ったデータ(bytes)(イーサネット MODBUS server)	R	R	R

22.3.13. 通信パラメータの設定

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9030	COM1 通信パラメータを更新する(LW-9550~9554)(ON に設定する)	R/W	R/C	R/C
LB-9031	COM2 通信パラメータを更新する(LW-9555~9559)(ON に設定する)	R/W	R/C	R/C
LB-9032	COM3 通信パラメータを更新する(LW-9560~9564)(ON に設定する)	R/W	R/C	R/C
LB-9065	COM1 ブローキャストステーション番号を無効/有効にする	R/W	R/C	R/C
LB-9066	COM2 ブローキャストステーション番号を無効/有効にする	R/W	R/C	R/C
LB-9067	COM3 ブローキャストステーション番号を無効/有効にする	R/W	R/C	R/C
LW-9550	(16bit):COM1 モード(0:RS232, 1:RS4852W, 2:RS4854W) (LB-9030 を使用して全ての通信設定を更新する)	R/W	R/C	R/C
LW-9551	(16bit):COM1 ポーレート 200, 8:2400, 0:4800, 1:9600,	R/W	R/C	R/C

	10:14400, 2:19200, 11:28800, 3:38400, 4:57600, ..) *注 1			
LW-9552	(16bit):COM1 データビット 7bits, 8:8bits)	R/W	R/C	R/C
LW-9553	(16bit):COM1 パリティ none, 1:even, 2:odd, 3:mark, 4:space)	R/W	R/C	R/C
LW-9554	(16bit):COM1 ストップビット(1:1bit, 2:2bits)	R/W	R/C	R/C
LW-9555	(16bit):COM2 モード(0:RS232, 1:RS4852W, 2:RS4854W) (LB-9031 を使用して全ての通信設定を更新する)	R/W	R/C	R/C
LW-9556	(16bit):COM2 ポーレート(7:1200, 8:2400, 0:4800, 1:9600, 10:14400, 2:19200, 11:28800, 3:38400, 4:57600, ..) *注 1	R/W	R/C	R/C
LW-9557	(16bit):COM2 データビット(7:7bits, 8:8bits)	R/W	R/C	R/C
LW-9558	(16bit):COM2 パリティ(0:none, 1:even, 2:odd, 3:mark, 4:space)	R/W	R/C	R/C
LW-9559	(16bit):COM2 ストップビット(1:1bit, 2:2bits)	R/W	R/C	R/C
LW-9560	(16bit):COM3 モード(0:RS232, 1:RS4852W) (LB-9032 を使用して全ての通信設定を更新する)	R/W	R/C	R/C
LW-9561	(16bit):COM3 ポーレート(7:1200, 8:2400, 0:4800, 1:9600, 10:14400, 2:19200, 11:28800, 3:38400, 4:57600, ..) *注 1	R/W	R/C	R/C
LW-9562	(16bit):COM3 データビット(7:7bits, 8:8bits)	R/W	R/C	R/C
LW-9563	(16bit):COM3 パリティ(0:none, 1:even, 2:odd, 3:mark, 4:space)	R/W	R/C	R/C
LW-9564	(16bit):COM3 ストップビット(1:1bit, 2:2bits)	R/W	R/C	R/C
LW-9565	(16bit):COM1 ブローキャストステーション番号	R/W	R/C	R/C
LW-9566	(16bit):COM2 ブローキャストステーション番号	R/W	R/C	R/C
LW-9567	(16bit):COM3 ブローキャストステーション番号	R/W	R/C	R/C
LW-10500	(16bit):装置 1 タイムアウト(単位:100ms, 0 : 50ms)	R/W	R/C	R/C
LW-10501	(16bit):装置 1 通信遅延(単位:ms)	R/W	R/C	R/C
LW-10502	(16bit):装置 1 ACK 送信遅延(単位:ms)	R/W	R/C	R/C
LW-10503	(16bit):装置 1 パラメータ 1	R/W	R/C	R/C
LW-10504	(16bit):装置 1 パラメータ 2	R/W	R/C	R/C
LW-10505	(16bit):装置 2 タイムアウト(単位:100ms, 0 : 50ms)	R/W	R/C	R/C
LW-10506	(16bit):装置 2 通信遅延(単位:ms)	R/W	R/C	R/C
LW-10507	(16bit):装置 2 ACK 送信遅延(単位:ms)	R/W	R/C	R/C
LW-10508	(16bit):装置 2 パラメータ 1	R/W	R/C	R/C
LW-10509	(16bit):装置 2 パラメータ 2	R/W	R/C	R/C

LW-10510	(16bit):装置 3 タイムアウト(単位:100ms, 0 : 50ms)	R/W	R/C	R/C
LW-10511	(16bit):装置 3 通信遅延(単位:ms)	R/W	R/C	R/C
LW-10512	(16bit):装置 3ACK 送信遅延(単位:ms)	R/W	R/C	R/C
LW-10513	(16bit):装置 3 パラメータ 1	R/W	R/C	R/C
LW-10514	(16bit):装置 3 パラメータ 2	R/W	R/C	R/C
LW-10515	(16bit):装置 4 タイムアウト(単位:100ms)	R/W	R/C	R/C
LW-10516	(16bit):装置 4 通信遅延(単位:ms)	R/W	R/C	R/C
LW-10517	(16bit):装置 4ACK 送信遅延(単位:ms) (SIEMENS S7/400 リンクタイプ)	R/W	R/C	R/C
LW-10518	(16bit):装置 4 パラメータ 1(SIEMENS S7/400 ラック)	R/W	R/C	R/C
LW-10519	(16bit):装置 4 パラメータ 2(SIEMENS S7/400CPU スロット)	R/W	R/C	R/C
LW-10520	(16bit):装置 5 タイムアウト(単位:100ms)	R/W	R/C	R/C
LW-10521	(16bit):装置 5 通信(単位:ms)	R/W	R/C	R/C
LW-10522	(16bit):装置 5 ACK 送信遅延(単位:ms) (SIEMENS S7/400 リンクタイプ)	R/W	R/C	R/C
LW-10523	(16bit):装置 5 パラメータ 1(SIEMENS S7/400 ラック)	R/W	R/C	R/C
LW-10524	(16bit):装置 5 パラメータ 2(SIEMENS S7/400CPU スロット)	R/W	R/C	R/C
LW-10525	(16bit):装置 6 タイムアウト(単位:100ms)	R/W	R/C	R/C
LW-10526	(16bit):装置 6 通信遅延(単位:ms)	R/W	R/C	R/C
LW-10527	(16bit):装置 6 ACK 送信遅延(単位:ms) (SIEMENS S7/400 リンクタイプ)	R/W	R/C	R/C
LW-10528	(16bit):装置 6 パラメータ 1(SIEMENS S7/400 ラック)	R/W	R/C	R/C
LW-10529	(16bit):装置 6 パラメータ 2(SIEMENS S7/400CPU スロット)	R/W	R/C	R/C
LW-10530	(16bit):装置 7 タイムアウト(単位:100ms)	R/W	R/C	R/C
LW-10531	(16bit):装置 7 通信遅延(単位:ms)	R/W	R/C	R/C
LW-10532	(16bit):装置 7 ACK 送信遅延(単位:ms) (SIEMENS S7/400 リンクタイプ)	R/W	R/C	R/C
LW-10533	(16bit):装置 7 パラメータ 1(SIEMENS S7/400 ラック)	R/W	R/C	R/C
LW-10534	(16bit):装置 7 パラメータ 2(SIEMENS S7/400CPU スロット)	R/W	R/C	R/C
LW-10535	(16bit):装置 8 タイムアウト(単位:100ms)	R/W	R/C	R/C
LW-10536	(16bit):装置 8 通信遅延(単位:ms)	R/W	R/C	R/C
LW-10537	(16bit):装置 8 ACK 送信遅延(単位:ms) (SIEMENS S7/400 リンクタイプ)	R/W	R/C	R/C

LW-10538	(16bit):装置 8 パラメータ 1(SIEMENS S7/400 ラック)	R/W	R/C	R/C
LW-10539	(16bit):装置 8 パラメータ 2(SIEMENS S7/400CPU スロット)	R/W	R/C	R/C
LW-10655	(16bit):装置 32 タイムアウト(単位:100ms)	R/W	R/C	R/C
LW-10656	(16bit):装置 32 通信遅延(単位:ms)	R/W	R/C	R/C
LW-10657	(16bit):装置 32 ACK 送信遅延(単位:ms)	R/W	R/C	R/C
LW-10658	(16bit):装置 32 パラメータ 1	R/W	R/C	R/C
LW-10659	(16bit):装置 32 パラメータ 2	R/W	R/C	R/C

 Note

1. ポーレートは： 0:4800、1:9600、2:19200、3:38400、4:57600、5:115200、6:187.5K、7:1200、8:2400、10:14400、11:28800、12:76800 です。

22.3.14. 装置(COM)との通信状態及びコントロール

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9150	自動的に装置 1(COM1)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9151	自動的に装置 2(COM2)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9152	自動的に装置 3(COM3)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9200	装置 1 との通信状態(ステーション番号 0, COM1), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9201	装置 1 との通信状態(ステーション番号 1, COM1), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9202	装置 1 との通信状態(ステーション番号 2, COM1), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9203	装置 1 との通信状態(ステーション番号 3, COM1), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9204	装置 1 との通信状態(ステーション番号 4, COM1), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9205	装置 1 との通信状態(ステーション番号 5, COM1), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9206	装置 1 との通信状態(ステーション番号 6, COM1), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9207	装置 1 との通信状態(ステーション番号 7, COM1), ON	R/W	R/C	R/C

	に設定して改めて接続する			
LB-9455	装置 1 との通信状態(ステーション番号 255, COM1), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9500	装置 2 との通信状態(ステーション番号 0, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9501	装置 2 との通信状態(ステーション番号 1, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9502	装置 2 との通信状態(ステーション番号 2, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9503	装置 2 との通信状態(ステーション番号 3, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9504	装置 2 との通信状態(ステーション番号 4, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9505	装置 2 との通信状態(ステーション番号 5, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9506	装置 2 との通信状態(ステーション番号 6, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9507	装置 2 との通信状態(ステーション番号 7, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9755	装置 2 との通信状態(ステーション番号 255, COM2), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9800	装置 3 との通信状態(ステーション番号 0, COM3), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9801	装置 3 との通信状態(ステーション番号 1, COM3), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9802	装置 3 との通信状態(ステーション番号 2, COM3), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9803	装置 3 との通信状態(ステーション番号 3, COM3), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9804	装置 3 との通信状態(ステーション番号 4, COM3), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9805	装置 3 との通信状態(ステーション番号 5, COM3), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9806	装置 3 との通信状態(ステーション番号 6, COM3), ON に設定して改めて接続する	R/W	R/C	R/C
LB-9807	装置 3 との通信状態(ステーション番号 7, COM3), ON に設定して改めて接続する	R/W	R/C	R/C
LB-10055	装置 3 との通信状態(ステーション番号 255, COM3), ON	R/W	R/C	R/C

	に設定して改めて接続する			
LB-12030	COM1 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)*注 1	R	R	R
LB-12031	COM2 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)	R	R	R
LB-12032	COM3 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)	R	R	R
LB-12033	COM4 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)	R	R	R
LB-12034	COM5 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)	R	R	R
LB-12035	COM6 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)	R	R	R
LB-12036	COM7 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)	R	R	R
LB-12037	COM8 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)	R	R	R
LB-12038	COM9 状態指示をオープンする(OFF:正常, ON:オープンに失敗した)	R	R	R
LW-9351	(16bit):装置 1(COM1)まだ処理していないコマンド数	R	R	R
LW-9352	(16bit):装置 2(COM2)まだ処理していないコマンド数	R	R	R
LW-9353	(16bit):装置 3(COM3)まだ処理していないコマンド数	R	R	R

Note

2. PC シミュレーションをする時に、COM の状態指示をオープンし、COM が他のプログラムに占められたのかを確認できます。

22.3.15. 装置(イーサネット)との通信状態及びコントロール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9153	自動的に装置 4(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9154	自動的に装置 5(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9155	自動的に装置 6(イーサネット)に接続する(状態は ON	R/W	R/C	R/C

	(場合)			
LB-9156	自動的に装置 7(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9157	自動的に装置 8(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9158	自動的に装置 9(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9189	自動的に装置 40(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-10070	オンラインで装置 4(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10071	オンラインで装置 5(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10072	オンラインで装置 6(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めてする	R/W	R/C	R/C
LB-10073	オンラインで装置 7(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10074	オンラインで装置 8(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10075	オンラインで装置 9(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10099	オンラインで装置 33(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10100	装置 4 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10101	装置 4 との通信状態(ステーション番号 0、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10102	装置 4 との通信状態(ステーション番号 1、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10103	装置 4 との通信状態(ステーション番号 2、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C

LB-10104	装置 4 との通信状態(ステーション番号 3、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10105	装置 4 との通信状態(ステーション番号 4、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10106	装置 4 との通信状態(ステーション番号 5、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10107	装置 4 との通信状態(ステーション番号 6、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10108	装置 4 との通信状態(ステーション番号 7、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10356	装置 4 との通信状態(ステーション番号 255、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10400	装置 5 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10401	装置 5 との通信状態(ステーション番号 0、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10402	装置 5 との通信状態(ステーション番号 1、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10403	装置 5 との通信状態(ステーション番号 2、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10404	装置 5 との通信状態(ステーション番号 3、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10405	装置 5 との通信状態(ステーション番号 4、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10406	装置 5 との通信状態(ステーション番号 5、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10407	装置 5 との通信状態(ステーション番号 6、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10408	装置 5 との通信状態(ステーション番号 7、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10656	装置 5 との通信状態(ステーション番号 255、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10700	装置 6 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10701	装置 6 との通信状態(ステーション番号 0、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10702	装置 6 との通信状態(ステーション番号 1、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C

LB-10703	装置 6 との通信状態(ステーション番号 2、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10704	装置 6 との通信状態(ステーション番号 3、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10705	装置 6 との通信状態(ステーション番号 4、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10706	装置 6 との通信状態(ステーション番号 5、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10707	装置 6 との通信状態(ステーション番号 6、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10708	装置 6 との通信状態(ステーション番号 7、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-10956	装置 6 との通信状態(ステーション番号 255、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11000	装置 7 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11001	装置 7 との通信状態(ステーション番号 0、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11002	装置 7 との通信状態(ステーション番号 1、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11003	装置 7 との通信状態(ステーション番号 2、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11004	装置 7 との通信状態(ステーション番号 3、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11005	装置 7 との通信状態(ステーション番号 4、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11006	装置 7 との通信状態(ステーション番号 5、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11007	装置 7 との通信状態(ステーション番号 6、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11008	装置 7 との通信状態(ステーション番号 7、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11256	装置 7 との通信状態(ステーション番号 255、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11300	装置 8 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11301	装置 8 との通信状態(ステーション番号 0、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C

LB-11302	装置 8 との通信状態(ステーション番号 1、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11303	装置 8 との通信状態(ステーション番号 2、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11304	装置 8 との通信状態(ステーション番号 3、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11305	装置 8 との通信状態(ステーション番号 4、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11306	装置 8 との通信状態(ステーション番号 5、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11307	装置 8 との通信状態(ステーション番号 6、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11308	装置 8 との通信状態(ステーション番号 7、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11556	装置 8 との通信状態(ステーション番号 255、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11600	装置 9 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11601	装置 9 との通信状態(ステーション番号 0、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11602	装置 9 との通信状態(ステーション番号 1、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11603	装置 9 との通信状態(ステーション番号 2、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11604	装置 9 との通信状態(ステーション番号 3、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11605	装置 9 との通信状態(ステーション番号 4、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11606	装置 9 との通信状態(ステーション番号 5、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11607	装置 9 との通信状態(ステーション番号 6、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11608	装置 9 との通信状態(ステーション番号 7、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11856	装置 9 との通信状態(ステーション番号 255、イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11900	装置 10 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C

LB-11901	装置 11 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11902	装置 12 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11903	装置 13 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11904	装置 14 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11905	装置 15 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11906	装置 16 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-11939	装置 49 との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12670	自動的に装置 41(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-12671	自動的に装置 42(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-12672	自動的に装置 43(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-12673	自動的に装置 44(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-12693	自動的に装置 64(イーサネット)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-12700	装置 50 的との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12701	装置 51 的との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12702	装置 52 的との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12703	装置 53 的との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12714	装置 64 的との通信状態(イーサネット)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12720	オンラインで装置 34(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12721	オンラインで装置 35(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C

	ムパラメータを変更する時、ON に設定して改めて接続する			
LB-12722	オンラインで装置 36(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12723	オンラインで装置 37(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12750	オンラインで装置 64(イーサネット)の IP 或いはシステムパラメータを変更する時、ON に設定して改めて接続する	R/W	R/C	R/C
LW-9354	(16bit):装置 4(イーサネット)まだ処理していないコマンド数	R	R	R
LW-9355	(16bit):装置 5(イーサネット)まだ処理していないコマンド数	R	R	R
LW-9356	(16bit):装置 6(イーサネット)まだ処理していないコマンド数	R	R	R
LW-9357	(16bit):装置 7(イーサネット)まだ処理していないコマンド数	R	R	R
LW-9389	(16bit):装置 39(イーサネット)まだ処理していないコマンド数	R	R	R
LW-9600	(16bit):装置 4 の IP0(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9601	(16bit):装置 4 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9602	(16bit):装置 4 の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9603	(16bit):装置 4 の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9604	(16bit):装置 4 のポート番号	R/W	R/C	R/C
LW-9605	(16bit):装置 5 の IP0(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9606	(16bit):装置 5 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9607	(16bit):装置 5 の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9608	(16bit):装置 5 の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9609	(16bit):装置 5 のポート番号	R/W	R/C	R/C
LW-9610	(16bit):装置 6 の IP0(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9611	(16bit):装置 6 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9612	(16bit):装置 6 の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9613	(16bit):装置 6 の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9614	(16bit):装置 6 のポート番号	R/W	R/C	R/C
LW-9615	(16bit):装置 7 の IP0(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C

LW-9616	(16bit):装置 7 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9617	(16bit):装置 7 の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9618	(16bit):装置 7 の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9619	(16bit):装置 7 のポート番号	R/W	R/C	R/C
LW-9620	(16bit):装置 8 の IPO(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9621	(16bit):装置 8 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9622	(16bit):装置 8 の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9623	(16bit):装置 8 の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9624	(16bit):装置 8 のポート番号	R/W	R/C	R/C
LW-9625	(16bit):装置 9 の IPO(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9626	(16bit):装置 9 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9627	(16bit):装置 9 の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9628	(16bit):装置 9 の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9629	(16bit):装置 9 のポート番号	R/W	R/C	R/C
LW-9765	(16bit):装置 37 の IPO(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9766	(16bit):装置 37 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9767	(16bit):装置 37 の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9768	(16bit):装置 37 の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9769	(16bit):装置 37 のポート番号	R/W	R/C	R/C
LW-11472	(16bit):装置 4 の ID0 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11473	(16bit):装置 4 の ID1 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11474	(16bit):装置 4 の ID2 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11475	(16bit):装置 4 の ID3 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11476	(16bit):装置 4 の ID4 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11477	(16bit):装置 4 の ID5 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11478	(16bit):装置 5 の ID0 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11479	(16bit):装置 5 の ID1 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11480	(16bit):装置 5 の ID2 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C

LW-11481	(16bit):装置 5 の ID3 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11482	(16bit):装置 5 の ID4 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11483	(16bit):装置 5 の ID5 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11484	(16bit):装置 6 の ID0 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11485	(16bit):装置 6 の ID1 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11486	(16bit):装置 6 の ID2 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11487	(16bit):装置 6 の ID3 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11488	(16bit):装置 6 の ID4 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11489	(16bit):装置 6 の ID5 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11490	(16bit):装置 7 の ID0 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11491	(16bit):装置 7 の ID1 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11492	(16bit):装置 7 の ID2 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11493	(16bit):装置 7 の ID3 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11494	(16bit):装置 7 の ID4 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-11495	(16bit):装置 7 の ID5 (Beckhoff AMS NetId = ID0:ID1:ID2:ID3:ID4:ID5)	R/W	R/C	R/C
LW-12110	(16bit):装置 38 の IP0 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12111	(16bit):装置 38 の IP1 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12112	(16bit):装置 38 の IP2 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12113	(16bit):装置 38 の IP3 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12114	(16bit):装置 38 のポート番号	R/W	R/C	R/C
LW-12115	(16bit):装置 39 の IP0 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12116	(16bit):装置 39 の IP1 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C

LW-12117	(16bit):装置 39 の IP2 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12118	(16bit):装置 39 の IP3 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12119	(16bit):装置 39 のポート番号	R/W	R/C	R/C
LW-12120	(16bit):装置 40 の IP0 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12121	(16bit):装置 40 の IP1 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12122	(16bit):装置 40 の IP2 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12123	(16bit):装置 40 の IP3 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12124	(16bit):装置 40 のポート番号	R/W	R/C	R/C
LW-12125	(16bit):装置 41 の IP0 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12126	(16bit):装置 41 の IP1 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12127	(16bit):装置 41 の IP2 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12128	(16bit):装置 41 の IP3 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12129	(16bit):装置 41 のポート番号	R/W	R/C	R/C
LW-12240	(16bit):装置 64 の IP0 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12241	(16bit):装置 64 の IP1 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12242	(16bit):装置 64 の IP2 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12243	(16bit):装置 64 の IP3 (IP アドレス = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-12244	(16bit):装置 64 のポート番号	R/W	R/C	R/C

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.16. 装置(USB)との通信状態及びコントロール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9190	自動的に装置(USB)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9191	装置との通信状態(USB), ON に設定して改めて接続する	R/W	R/C	R/C
LW-9390	(16bit):装置(USB)まだ処理していないコマンド数	R	R	R

22.3.17. 装置(CAN Bus)との通信状態及びコントロール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-12080	自動的に装置(CAN Bus)に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-12081	装置との通信状態(CAN Bus)、ON に設定して改めて接続する	R/W	R/C	R/C
LB-12100	CAN Bus 装置 1 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12101	CAN Bus 装置 2 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12102	CAN Bus 装置 3 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12103	CAN Bus 装置 4 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12104	CAN Bus 装置 5 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12105	CAN Bus 装置 6 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12106	CAN Bus 装置 7 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12107	CAN Bus 装置 8 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12108	CAN Bus 装置 9 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LB-12109	CAN Bus 装置 10 の通信を一時停止する(状態は ON の場合))	R/W	R/C	R/C
LB-12354	CAN Bus 装置 255 の通信を一時停止する(状態は ON の場合)	R/W	R/C	R/C
LW-9392	(16bit):装置(CAN Bus)まだ処理していないコマンド数	R	R	R

22.3.18. リモート HMI の通信状態及びコントロール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9068	自動的にリモート HMI1 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9069	自動的にリモート HMI2 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9070	自動的にリモート HMI3 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9071	自動的にリモート HMI4 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9072	自動的にリモート HMI5 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9073	自動的にリモート HMI6 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9074	自動的にリモート HMI7 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9075	自動的にリモート HMI8 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9099	自動的にリモート HMI32 に接続する(状態は ON の場合)	R/W	R/C	R/C
LB-9100	リモート HMI1 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9101	リモート HMI2 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9102	リモート HMI3 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9103	リモート HMI4 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9104	リモート HMI5 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9105	リモート HMI6 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9106	リモート HMI7 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9107	リモート HMI8 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9148	リモート HMI49 との通信状態、ON に設定して改めて接続する	R/W	R/C	R/C
LB-9149	オンラインでリモート HMI の IP を変更する時、ON に設定して改めてリモート HMI に接続する	R/W	R/C	R/C
LW-9800	(16bit):リモート HMI1 の IPO(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9801	(16bit):リモート HMI1 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C

LW-9802	(16bit):リモートHMI1のIP2(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9803	(16bit):リモートHMI1のIP3(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9804	(16bit):リモートHMI1のポート番号	R/W	R/C	R/C
LW-9805	(16bit):リモートHMI2のIP0(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9806	(16bit):リモートHMI2のIP1(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9807	(16bit):リモートHMI2のIP2(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9808	(16bit):リモートHMI2のIP3(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9809	(16bit):リモートHMI2のポート番号	R/W	R/C	R/C
LW-9810	(16bit):リモートHMI3のIP0(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9811	(16bit):リモートHMI3のIP1(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9812	(16bit):リモートHMI3のIP2(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9813	(16bit):リモートHMI3のIP3(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9814	(16bit):リモートHMI3のポート番号	R/W	R/C	R/C
LW-9815	(16bit):リモートHMI4のIP0(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9816	(16bit):リモートHMI4のIP1(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9817	(16bit):リモートHMI4のIP2(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9818	(16bit):リモートHMI4のIP3(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9819	(16bit):リモートHMI4のポート番号	R/W	R/C	R/C
LW-9820	(16bit):リモートHMI5のIP0(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9821	(16bit):リモートHMI5のIP1(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9822	(16bit):リモートHMI5のIP2(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9823	(16bit):リモートHMI5のIP3(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9824	(16bit):リモートHMI5のポート番号	R/W	R/C	R/C
LW-9825	(16bit):リモートHMI6のIP0(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9826	(16bit):リモートHMI6のIP1(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9827	(16bit):リモートHMI6のIP2(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9828	(16bit):リモートHMI6のIP3(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9829	(16bit):リモートHMI6のポート番号	R/W	R/C	R/C
LW-9830	(16bit):リモートHMI7のIP0(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9831	(16bit):リモートHMI7のIP1(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9832	(16bit):リモートHMI7のIP2(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9833	(16bit):リモートHMI7のIP3(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9834	(16bit):リモートHMI7のポート番号	R/W	R/C	R/C
LW-9835	(16bit):リモートHMI8のIP0(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9836	(16bit):リモートHMI8のIP1(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9837	(16bit):リモートHMI8のIP2(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9838	(16bit):リモートHMI8のIP3(IPアドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C

LW-9839	(16bit):リモート HMI8 のポート番号 =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9895	(16bit):リモート HMI20 の IP0(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9896	(16bit):リモート HMI20 の IP1(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9897	(16bit):リモート HMI20 の IP2(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9898	(16bit):リモート HMI20 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9899	(16bit):リモート HMI20 のポート番号	R/W	R/C	R/C
LW-9905	(16bit):リモート HMI21 の IP0(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9906	(16bit):リモート HMI21 の IP1(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9907	(16bit):リモート HMI21 の IP2(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9908	(16bit):リモート HMI21 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9909	(16bit):リモート HMI21 のポート番号	R/W	R/C	R/C
LW-9910	(16bit):リモート HMI22 の IP0(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9911	(16bit):リモート HMI22 の IP1(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9912	(16bit):リモート HMI22 の IP2(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9913	(16bit):リモート HMI22 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9914	(16bit):リモート HMI22 のポート番号	R/W	R/C	R/C
LW-9915	(16bit):リモート HMI23 の IP0(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9916	(16bit):リモート HMI23 の IP1(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9917	(16bit):リモート HMI23 の IP2(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9918	(16bit):リモート HMI23 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9919	(16bit):リモート HMI23 のポート番号	R/W	R/C	R/C

LW-9920	(16bit):リモート HMI24 の IPO(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9921	(16bit):リモート HMI24 の IP1(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9922	(16bit):リモート HMI24 の IP2(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9923	(16bit):リモート HMI24 の IP3(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9924	(16bit):リモート HMI24 のポート番号	R/W	R/C	R/C
LW-9925	(16bit):リモート HMI25 の IPO(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9926	(16bit):リモート HMI25 の IP1(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9927	(16bit):リモート HMI25 の IP2(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9928	(16bit):リモート HMI25 の IP3(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9929	(16bit):リモート HMI25 のポート番号	R/W	R/C	R/C
LW-9930	(16bit):リモート HMI26 の IPO(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9931	(16bit):リモート HMI26 の IP1(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9932	(16bit):リモート HMI26 の IP2(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9933	(16bit):リモート HMI26 の IP3(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9934	(16bit):リモート HMI26 のポート番号	R/W	R/C	R/C
LW-9935	(16bit):リモート HMI27 の IPO(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9936	(16bit):リモート HMI27 の IP1(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9937	(16bit):リモート HMI27 の IP2(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9938	(16bit):リモート HMI27 の IP3(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9939	(16bit):リモート HMI27 のポート番号	R/W	R/C	R/C
LW-9940	(16bit):リモート HMI28 の IPO(IP アドレス	R/W	R/C	R/C

	=IPO:IP1:IP2:IP3)			
LW-9941	(16bit):リモートHMI28のIP0(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9942	(16bit):リモートHMI28のIP1(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9943	(16bit):リモートHMI28のIP2(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9944	(16bit):リモートHMI28のポート番号	R/W	R/C	R/C
LW-9945	(16bit):リモートHMI29のIP0(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9946	(16bit):リモートHMI29のIP1(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9947	(16bit):リモートHMI29のIP2(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9948	(16bit):リモートHMI29のIP3(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9949	(16bit):リモートHMI29のポート番号	R/W	R/C	R/C
LW-9950	(16bit):リモートHMI30のIP0(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9951	(16bit):リモートHMI30のIP1(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9952	(16bit):リモートHMI30のIP2(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9953	(16bit):リモートHMI30のIP3(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9954	(16bit):リモートHMI30のポート番号	R/W	R/C	R/C
LW-9955	(16bit):リモートHMI31のIP0(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9956	(16bit):リモートHMI31のIP1(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9957	(16bit):リモートHMI31のIP2(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9958	(16bit):リモートHMI31のIP3(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9959	(16bit):リモートHMI31のポート番号	R/W	R/C	R/C
LW-9960	(16bit):リモートHMI32のIP0(IPアドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C

LW-9961	(16bit):リモート HMI32 の IP1(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9962	(16bit):リモート HMI32 の IP2(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9963	(16bit):リモート HMI32 の IP3(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9964	(16bit):リモート HMI32 のポート番号	R/W	R/C	R/C
LW-9995	(16bit):リモート HMI39 の IPO(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9996	(16bit):リモート HMI39 の IP1(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9997	(16bit):リモート HMI39 の IP2(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9998	(16bit):リモート HMI39 の IP3(IP アドレス =IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9999	(16bit):リモート HMI39 のポート番号	R/W	R/C	R/C

22.3.19. リモート装置との通信状態及びコントロール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-10050	(16bit):リモート装置 1 に接続する HMI の IPO(IP アドレス=IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10051	(16bit):リモート装置 1 に接続する HMI の IP1(IP アドレス=IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10052	(16bit):リモート装置 1 に接続する HMI の IP2(IP アドレス=IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10053	(16bit):リモート装置 1 に接続する HMI の IP3(IP アドレス=IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10054	(16bit):リモート装置 1 に接続する HMI のポート番号	R/W	R/C	R/C
LW-10055	(16bit):リモート装置 2 に接続する HMI の IPO(IP アドレス=IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10056	(16bit):リモート装置 2 に接続する HMI の IP1(IP アドレス=IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10057	(16bit):リモート装置 2 に接続する HMI の IP2(IP アドレス=IPO:IP1:IP2:IP3)	R/W	R/C	R/C

	ス=IP0:IP1:IP2:IP3)			
LW-10058	(16bit):リモート装置 2 に接続する HMI の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10059	(16bit):リモート装置 2 に接続する HMI のポート番号	R/W	R/C	R/C
LW-10060	(16bit):リモート装置 3 に接続する HMI の IP0(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10061	(16bit):リモート装置 3 に接続する HMI の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10062	(16bit):リモート装置 3 に接続する HMI の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10063	(16bit):リモート装置 3 に接続する HMI の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10064	(16bit):リモート装置 3 に接続する HMI のポート番号	R/W	R/C	R/C
LW-10065	(16bit):リモート装置 4 に接続する HMI の IP0(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10066	(16bit):リモート装置 4 に接続する HMI の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10067	(16bit):リモート装置 4 に接続する HMI の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10068	(16bit):リモート装置 4 に接続する HMI の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10069	(16bit):リモート装置 4 に接続する HMI のポート番号	R/W	R/C	R/C
LW-10205	(16bit):リモート装置 32 に接続する HMI の IP0(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10206	(16bit):リモート装置 32 に接続する HMI の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10207	(16bit):リモート装置 32 に接続する HMI の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10208	(16bit):リモート装置 32 に接続する HMI の IP3(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10209	(16bit):リモート装置 32 に接続する HMI のポート番号	R/W	R/C	R/C
LW-10300	(16bit):リモート装置 1 の IP0(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10301	(16bit):リモート装置 1 の IP1(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10302	(16bit):リモート装置 1 の IP2(IP アドレス=IP0:IP1:IP2:IP3)	R/W	R/C	R/C

LW-10303	(16bit):リモート装置 1 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10304	(16bit):リモート装置 1 のポート番号	R/W	R/C	R/C
LW-10305	(16bit):リモート装置 2 の IP0(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10306	(16bit):リモート装置 2 の IP1(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10307	(16bit):リモート装置 2 の IP2(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10308	(16bit):リモート装置 2 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10309	(16bit):リモート装置 2 のポート番号	R/W	R/C	R/C
LW-10310	(16bit):リモート装置 3 の IP0(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10311	(16bit):リモート装置 3 の IP1(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10312	(16bit):リモート装置 3 の IP2(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10313	(16bit):リモート装置 3 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10314	(16bit):リモート装置 3 のポート番号	R/W	R/C	R/C
LW-10315	(16bit):リモート装置 4 の IP0(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10316	(16bit):リモート装置 4 の IP1(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10317	(16bit):リモート装置 4 の IP2(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10318	(16bit):リモート装置 4 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10319	(16bit):リモート装置 4 のポート番号	R/W	R/C	R/C
LW-10455	(16bit):リモート装置 32 の IP0(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10456	(16bit):リモート装置 32 の IP1(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10457	(16bit):リモート装置 32 の IP2(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10458	(16bit):リモート装置 32 の IP3(IP アドレス =IP0:IP1:IP2:IP3)	R/W	R/C	R/C

	=IPO:IP1:IP2:IP3)			
LW-10459	(16bit):リモート装置 32 のポート番号	R/W	R/C	R/C

22.3.20. ローカル/リモートコントロールの制限

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9044	リモートコントロール禁止(状態は ON の場合)	R/W	R/C	R/C
LB-9053	リモートでのパスワード読み取り禁止(状態は ON の場合)	R/W	R/C	R/C
LB-9054	リモートでのパスワード書込み禁止(状態は ON の場合)	R/W	R/C	R/C
LB-9196	ローカル HMI は閲覧機能のみサポートする(状態は ON の場合)	R/W	R/C	R/C
LB-9197	リモート HMI だけが閲覧機能の使用を許される(状態は ON の場合)	R/W	R/C	R/C
LB-9198	ローカル HMI でマクロをトリガーすることを禁じる(状態は ON の場合)	R/W	R/C	R/C
LB-9199	リモート HMI でマクロをトリガーすることを禁じる(状態は ON の場合)	R/W	R/C	R/C

22.3.21. 通信エラーコード

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-9400	(16bit):装置 1 と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9401	(16bit):装置 2 と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9402	(16bit):装置 3 と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9403	(16bit):装置 4 と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9404	(16bit):装置 5 と通信時に発生した通信エラーのエラーコード	R	R	R

	コード			
LW-9405	(16bit):装置 6 と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9406	(16bit):装置 7 と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9407	(16bit):装置 8 と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9449	(16bit):装置 50 と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9490	(16bit):装置(USB)と通信時に発生した通信エラーのエラーコード	R	R	R
LW-9491	(16bit): 装置 (CAN-Bus)と通信時に発生した通信エラーのエラーコード	R	R	R

 Note

1. インポートに成功したら、出力値は **1** で、インポートに失敗したら(ファイルが存在しない)、出力値は **2** になり、ファイルフォーマットが正しくない、もしくはデコードできない場合、出力値は **4** になります。
2. 通信エラーコードの説明 :

エラーコード	通信エラーが発生した原因
0	正常
1	装置はビジー状態にあり、コマンドを受け取れない
2	通信エラー(原因不明)
3	装置が存在しない
4	指定したステーション番号の装置が存在しない
5	アドレスフォーマットが正しくない
6	サポートしないアドレスに読み取った/書き込んだ
7	装置が使用するデバイスドライバが存在しない
8	シリアルポート(COM Port)が存在しない
9	装置の IP アドレスが正しくないか、または当装置に接続できない
10	装置が返信したコマンドの内容にチェックサムエラーが発生した (checksum error)
11	判別不能なコマンド
12	無視
20	USB インターフェースを使用する装置に正しく接続していない
21	CAN Bus インターフェースを使用する装置に正しく接続していない
22	装置からの返答を受け取っていない
23	指定した時間内(timeout)で装置から十分のデータ数を読み取っていない

24	オブジェクトが使用する Conversion Tag が存在しないか、または内容が正しくない
25	HMI は Remote HMI からのコマンドを拒否している
251	MODBUS レジスタに読み取った/書き込んだワード数(word no.)が許容値を超えた
252	MODBUS 装置が返信したデータのフォーマットが正しくない
253	MODBUS 装置が返信したデータにチェックサムエラーが発生した (checksum error)

22.3.22. ドライバ ID

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-9300	(16bit): ローカル装置 1 が使用するドライバ ID	R	R	R
LW-9301	(16bit): ローカル装置 2 が使用するドライバ ID	R	R	R
LW-9302	(16bit): ローカル装置 3 が使用するドライバ ID	R	R	R
LW-9303	(16bit): ローカル装置 4 が使用するドライバ ID	R	R	R
LW-9331	(16bit): ローカル装置 32 が使用するドライバ ID	R	R	R

22.3.23. DLT645 コントローラー

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-10700	(4 words):DLT_645 ユーザー(COM1)	R/W	R/C	R/C
LW-10704	(4 words):DLT_645 パスワード(COM1)	R/W	R/C	R/C
LW-10708	(6 words):DLT_645 アドレス(COM1)	R/W	R/C	R/C
LW-10715	(4 words):DLT_645 ユーザー(COM2)	R/W	R/C	R/C
LW-10719	(4 words):DLT_645 パスワード(COM2)	R/W	R/C	R/C
LW-10723	(6 words):DLT_645 アドレス(COM2)	R/W	R/C	R/C
LW-10730	(4 words):DLT_645 ユーザー(COM3)	R/W	R/C	R/C
LW-10734	(4 words):DLT_645 パスワード(COM3)	R/W	R/C	R/C
LW-10738	(6 words):DLT_645 アドレス(COM3)	R/W	R/C	R/C

22.3.24. [Device No Response] ウィンドウのコントロール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9192	装置(USB)の"Device No Response" ウィンドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-11960	装置 1 の"Device No Response" ウィンドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C

LB-11961	装置 2 の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-11962	装置 3 の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-11963	装置 4 の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-11964	装置 5 の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-11965	装置 6 の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-11966	装置 7 の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-11967	装置 8 の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-12023	装置 64 の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C
LB-12082	CAN Bus 装置の"Device No Response"ウインドウをポップアップ禁止(状態は ON の場合)	R/W	R/C	R/C

22.3.25. [ファーストセレクション]ウインドウのコントロール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9013	ファーストセレクションウインドウを隠す(ON に設定する)/表示する(OFF に設定する)	R/W	R/C	R/C
LB-9014	ファーストセレクションボタンを隠す(ON に設定する)/表示する(OFF に設定する)	R/W	R/C	R/C
LB-9015	ファーストセレクションウインドウ/ボタンを隠す(ON に設定する)/表示する(OFF)	R/W	R/C	R/C

22.3.26. EasyAccess

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9051	EasyAccess のサーバーとの接続を切断する(OFF に設定する)/接続する(ON に設定する)	R/W	R/C	R/C
LB-9052	EasyAccess のサーバーとの接続状態(接続中の場合、状態は ON に表示される)	R	R	R

 更に多くの EasyAccess に関する情報について、<http://www.ihmi.net/>をご参考ください。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.27. EasyAccess2.0

アド レス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-10820	(16bit):無効にする(0 に設定する)/有効にする(1 に設定する)(EasyAccess2.0)	R/W	R/C	R/C
LW-10821	(5 words):sessionID(EasyAccess2.0)	R/W	R/C	R/C
LW-10826	(2 words):パスワード(EasyAccess2.0)	R/W	R/C	R/C
LW-10828	(16bit):実行状態(EasyAccess2.0)	R	R	R
LW-10829	(16bit):最後のエラーコード(EasyAccess2.0)	R	R	R
LW-11170	(16bit):プロキシサーバーを無効/有効にする(0: 無効にする, 1: 有効にする)(EasyAccess2.0)	R/W	R/C	R/C
LW-11171	(16bit):プロキシサーバーのタイプ(0:HTTP, 1:SOCKSv4, 2:SOCKSv5)(EasyAccess2.0)	R/W	R/C	R/C
LW-11172	(16bit):プロキシサーバーIP0(EasyAccess2.0)	R/W	R/C	R/C
LW-11173	(16bit):プロキシサーバーIP1(EasyAccess2.0)	R/W	R/C	R/C
LW-11174	(16bit):プロキシサーバーIP2(EasyAccess2.0)	R/W	R/C	R/C
LW-11175	(16bit):プロキシサーバーIP3(EasyAccess2.0)	R/W	R/C	R/C
LW-11176	(16bit):プロキシサーバーのポート番号(EasyAccess2.0)	R/W	R/C	R/C

LW-11177	(16bit):プロキシサーバーを認証する(0:無効にする, 1:有効にする)(EasyAccess2.0)	R/W	R/C	R/C
LW-11178	(16 words):プロキシサーバーのユーザー名(EasyAccess2.0)	R/W	R/C	R/C
LW-11194	(16 words):プロキシサーバーのパスワード(EasyAccess2.0)	R/W	R/C	R/C
LW-11210	(20 words) : ハードウェアキー(EasyAccess 2.0)	R	R	R
LW-11296	(16bit) : EasyAccess 2.0 サーバー位置(0 : グローバル, 1 : 中国)	R	R	R
LW-11770	(64 words) : WeChat プッシュ通知の QR コード(URL)(EasyAccess 2.0)	R	R	R
LW-11982	(16 words) : HMI ドメイン (EasyAccess 2.0)	R	R	R

22.3.28. リモートプリンター/バックアップサーバー

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-10069	オンラインでリモートプリンター/バックアップサーバーの IP を変更する時、ON に設定して改めてリモートプリンター/バックアップサーバーに接続する	R/W	R/C	R/C
LB-12040	リモートプリンター/バックアップサーバーの接続切断メッセージ(状態は ON の場合)	R	R	R
LW-9770	(16bit):リモートプリンター/バックアップサーバーの IP0(IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9771	(16bit):リモートプリンター/バックアップサーバーの IP1(IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9772	(16bit):リモートプリンター/バックアップサーバーの IP2(IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9773	(16bit):リモートプリンター/バックアップサーバーの IP3(IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9774	(6 words):リモートプリンター/バックアップサーバーにログインするに所要するユーザー名*注1	R/W	R/C	R/C
LW-9780	(6 words):リモートプリンター/バックアップサーバーにログインするに所要するパスワード*注1	R/W	R/C	R/C

 Note

1. LW-9774 及び LW-9780 を使用し設定を変更したい場合、変更した後には HMI を再起動する必要があります。それでこそ変更が有効になります。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.29. パススルー通信の設定

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-9901	(16bit):パススルー通信のデータソース COM ポート (1~3:COM1~COM3)	R/W	R/C	R/C
LW-9902	(16bit):パススルー通信のデータ行先 COM ポート (1~3:COM1~COM3)	R/W	R/C	R/C
LW-9903	(16bit):パススルー通信をコントロールする(0:正常ル, 1:停止, 2:パススルー機能を実行する時、HMI と装置の 通信を停止する)	R/W	R/C	R/C
LW-9904	(16bit):パススルーのサーバーCOM ポート(2000~2100)	R/W	R/C	R/C
LW-10850	(16bit):無効/有効にする(0:無効にする, 1:正常, 2:IP 制 限)(siemens パススルー 機能)	R/W	R/C	R/C
LW-10851	(16bit):行先 COM ポート(siemens パススルー機能)	R/W	R/C	R/C
LW-10852	(16bit):行先装置ステーション番号(siemens パススルー 機能)	R/W	R/C	R/C
LW-10853	(16bit):通信プロトコル(0:未定義, 1:PPI, 2:MPI)(siemens パススルー機能)	R/W	R/C	R/C
LW-10854	(16bit):client を接続する IP0 (IP address=IP0:IP1:IP2:IP3)(siemens パススルー機能)	R/W	R/C	R/C
LW-10855	(16bit):client を接続する IP1 (IP address=IP0:IP1:IP2:IP3)(siemens パススルー機能)	R/W	R/C	R/C
LW-10856	(16bit):client を接続する IP2 (IP address=IP0:IP1:IP2:IP3)(siemens パススルー機能)	R/W	R/C	R/C
LW-10857	(16bit):client を接続する IP3 (IP address=IP0:IP1:IP2:IP3)(siemens パススルー機能)	R/W	R/C	R/C

LW-10858	(16bit):指定した client の IP0 (IP address=IP0:IP1:IP2:IP3)(siemens パススルー機能)	R/W	R/C	R/C
LW-10859	(16bit):指定した client の IP1 (IP address=IP0:IP1:IP2:IP3)(siemens パススルー機能)	R/W	R/C	R/C
LW-10860	(16bit):指定した client の IP2 (IP address=IP0:IP1:IP2:IP3)(siemens パススルー機能)	R/W	R/C	R/C
LW-10861	(16bit):指定した client の IP3 (IP address=IP0:IP1:IP2:IP3)(siemens パススルー機能)	R/W	R/C	R/C
LW-10862	(16bit):接続状態(0:準備完了, 1:client 接続中)(siemens パススルー機能)	R	R	R
LW-10863	(16bit):接続状態(0:正常, 1:エラー)(siemens パススルー機能)	R	R	R
LW-10864	(16bit):最後のエラーコード(siemens パススルー機能)	R	R	R

 Siemens パススルー機能の詳細な説明について、《29 パススルー機能》をご参照ください。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.30. VNC のコントロール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-12088	VNC 監視モードを使用する(状態は ON の場合)*注 1	R/W	R/C	R/C
LB-12089	VNC パスワードがなくてもログインできる(状態は ON の場合)*注 1	R/W	R/C	R/C
LB-12090	VNC client が HMI に接続する(状態は ON の場合)(OS 20120621 以降、または最新バージョンの OS を使用してください)	R	R	R
LB-12091	VNC client が HMI に接続したら、自動的にログアウトする機能を無効にする(状態は ON の場合)(OS 20120621 またはさらに新しいバージョンの OS を使用してください)	R/W	R/C	R/C
LB-12092	VNC 機能を有効にする(ON に設定する)/無効にする	R/W	R/C	R/C

	(OFF に設定する)			
LB-12093	VNC 接続モード(OFF:シングル接続, ON:マルチ接続)*注 1	R/W	R/C	R/C
LW-9530	(8 words):VNC サーバーパスワード	R/W	R/C	R/C

 Note

1. VNC 設定モードを変更したいなら、LB-12092 で VNC 機能を閉じた後、再起動し、それでこそこの変更が有効になります。

22.3.31. HMI 識別キーとプロジェクト識別キー

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9046	HMI 識別キーとプロジェクト識別キーが異なる(状態は ON の場合)	R	R	R
LW-9046	(32bit):HMI 識別キー*注 1	R/W	R/C	R

 Note

1. LW-9046 で HMI 識別キーの設定を変更したい場合、変更した後には HMI を再起動する必要があります。それでこそ変更が有効になります。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.32. USB セキュリティキー

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-11160	(16bit):USB セキュリティキーの開始時間-年	R	R	R
LW-11161	(16bit):USB セキュリティキーの開始時間-月	R	R	R
LW-11162	(16bit):USB セキュリティキーの開始時間-日	R	R	R
LW-11163	(16bit):USB セキュリティキーの開始時間-時	R	R	R
LW-11164	(16bit):USB セキュリティキーの開始時間-分	R	R	R

LW-11165	(16bit):USB セキュリティキーの終了時間-年	R	R	R
LW-11166	(16bit):USB セキュリティキーの終了時間-月	R	R	R
LW-11167	(16bit):USB セキュリティキーの終了時間-日	R	R	R
LW-11168	(16bit):USB セキュリティキーの終了時間-時	R	R	R
LW-11169	(16bit):USB セキュリティキーの終了時間-分	R	R	R

22.3.33. ユーザー名とパスワード

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9050	ユーザログアウト	W	C	C
LB-9060	パスワード入力エラー	R	R	R
LB-9061	パスワードを更新する(ON に設定する)	W	C	C
LB-12056	ユーザーが授権されていないオブジェクトを操作した (状態は ON の場合)	R	R	R
PLB-12056	ユーザーが授権されていないオブジェクトを操作した (状態は ON の場合)(タブレットで)	R	N/A	N/A
LW-9082	(16bit):自動的にログアウト時間(単位:分, 0:本機能を無効にする)	R/W	R/C	R/C
LW-9219	(16bit):ユーザー番号(1~12)	R/W	R/C	R/C
LW-9220	(32bit):パスワード	R/W	R/C	R/C
LW-9222	(16bit):現在、ユーザーが使用できるオブジェクト種類 (bit0:A, bit1:B, bit2:C, ...)	R	N/A	N/A
PLW-9222	(16bit):現在、ユーザーが使用できるオブジェクト種類 (bit0:A, bit1:B, bit2:C, ...)	R	N/A	N/A
LW-9500	(32bit):ユーザー1 のパスワード	R/W	R/C	R/C
LW-9502	(32bit):ユーザー2 のパスワード	R/W	R/C	R/C
LW-9504	(32bit):ユーザー3 のパスワード	R/W	R/C	R/C
LW-9506	(32bit):ユーザー4 のパスワード	R/W	R/C	R/C
LW-9508	(32bit):ユーザー5 のパスワード	R/W	R/C	R/C
LW-9510	(32bit):ユーザー6 のパスワード	R/W	R/C	R/C
LW-9512	(32bit):ユーザー7 のパスワード	R/W	R/C	R/C
LW-9514	(32bit):ユーザー8 のパスワード	R/W	R/C	R/C
LW-9516	(32bit):ユーザー9 のパスワード	R/W	R/C	R/C
LW-9518	(32bit):ユーザー10 のパスワード	R/W	R/C	R/C

LW-9520	(32bit):ユーザー11のパスワード	R/W	R/C	R/C
LW-9522	(32bit):ユーザー12のパスワード	R/W	R/C	R/C
PLW-10754	(8words):現在、ログインしているユーザー名*注1	R	N/A	N/A

Note

1. [ユーザーパスワード] » [高度セキュリティモード]でのみサポートしています。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.34. マクロ

アド レス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9059	マクロ TRACE 機能を閉じる(状態は ON の場合)*注1	R/W	R/C	R/C
LW-10900	(16bit):マクロ 0 の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-10901	(16bit):マクロ 1 の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-10902	(16bit):マクロ 2 の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-10903	(16bit):マクロ 3 の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-10904	(16bit):マクロ 4 の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-10905	(16bit):マクロ 5 の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-10906	(16bit):マクロ 6 の状態(0:準備完了, 3:実行中, 5:装置の	R	R	R

	返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))			
LW-10907	(16bit):の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-10908	(16bit):の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-10909	(16bit):の状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R
LW-11154	(16bit):マの状態(0:準備完了, 3:実行中, 5:装置の返信を待つ, 9:同期されるのを待つ, 17:遅延, 32:異常的に停止される(配列サイズを超えた))	R	R	R

Note

1. LB-9059 マクロ TRACE 機能を閉じる例は下記に示す通りです :

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

22.3.35. 入力オブジェクトの機能

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-9002	(32bit-float):数値オブジェクトの入力上限値	R	R	R
LW-9004	(32bit-float):数値オブジェクトの入力下限値	R	R	R
LW-9052	(32bit-float):数値オブジェクトの前回での入力した数値	R	R	R
PLW-9052	(32bit-float):数値オブジェクトの前回での入力した数値	R	無し	無し
LW-9150	(32 words):現在、キーボードに入力したデータを表示する(ASCII)	R	R	R
LW-9540	(16bit):キーボード大文字/小文字の切替	R/W	R/C	R/C

22.3.36. 時刻同期/夏時間

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-12055	時刻同期(状態は ON の場合)	R	R	R
LB-12355	夏時間(状態は ON の場合)	R	R	R
LW-11260	(16bit):夏時間無効/有効にする(夏時間)(0:無効にする, 1:有効にする)	R/W	R/C	R/C
LW-11261	(16bit):時間調整値(時)	R/W	R/C	R/C
LW-11262	(16bit):時間調整値(分)	R/W	R/C	R/C
LW-11263	(16bit):夏時間の開始月	R/W	R/C	R/C
LW-11264	(16bit):夏時間の開始週数(1~5)	R/W	R/C	R/C
LW-11265	(16bit):夏時間び開始曜日(0~6)	R/W	R/C	R/C
LW-11266	(16bit):夏時間使用時のローカル時間(時)	R/W	R/C	R/C
LW-11267	(16bit):夏時間使用時のローカル時間(分)	R/W	R/C	R/C
LW-11268	(16bit):夏時間の終了月	R/W	R/C	R/C
LW-11269	(16bit):夏時間の終了週数(1~5)	R/W	R/C	R/C
LW-11270	(16bit):夏時間の終了曜日(0~6)	R/W	R/C	R/C
LW-11271	(16bit):夏時間終了時のローカル時間(時)	R/W	R/C	R/C
LW-11272	(16bit):夏時間終了時のローカル時間(分)	R/W	R/C	R/C
LW-11273	(16bit):NTP(Network Time Protocol)サーバーを通じて時間を同期するのを無効/有効にする(0:無効にする, 1:有効にする)	R/W	R/C	R/C
LW-11274	(16bit):HMIを起動する時、時間を同期する(0:無効にする, 1:有効にする)	R/W	R/C	R/C
LW-11275	(16bit):サーバーの返信時間は夏時間に応じて調整した(0:無効にする, 1:有効にする)	R/W	R/C	R/C
LW-11276	(16bit):HMIのタイムゾーン(単位:分)	R/W	R/C	R/C
LW-11277	(16bit):サーバーの返答時間(サーバーのタイムゾーン)(単位:分)	R/W	R/C	R/C
LW-11278	(16bit):ネットワークタイムサーバー1のIP0 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11279	(16bit):ネットワークタイムサーバー1のIP1 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11280	(16bit):ネットワークタイムサーバー1のIP2 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C

LW-11281	(16bit):ネットワークタイムサーバー1 の IP3 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11282	(16bit):ネットワークタイムサーバー2 の IP0 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11283	(16bit):ネットワークタイムサーバー2 の IP1 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11284	(16bit):ネットワークタイムサーバー2 の IP2 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11285	(16bit):ネットワークタイムサーバー2 の IP3 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11286	(16bit):ネットワークタイムサーバー3 の IP0 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11287	(16bit):ネットワークタイムサーバー3 の IP1 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11288	(16bit):ネットワークタイムサーバー3 の IP2 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11289	(16bit):ネットワークタイムサーバー3 の IP3 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11290	(16bit):ネットワークタイムサーバー4 の IP0 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11291	(16bit):ネットワークタイムサーバー4 の IP1 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11292	(16bit):ネットワークタイムサーバー4 の IP2 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11293	(16bit):ネットワークタイムサーバー4 の IP3 (IP address=IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11294	(32bit):更新周期(時間同期の間隔)(10~86400, 単位:秒)	R/W	R/C	R/C

22.3.37. セルラーネットワーク

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-11297	(16 words) : SIM カードの PIN コード (セルラーネットワーク)	R/W	R/C	R/C
LW-11313	(16 words) : アクセスポイントネーム(APN) (セルラー)	R/W	R/C	R/C

	ネットワーク)			
LW-11329	(16 words) : ユーザー名 (セルラーネットワーク)	R/W	R/C	R/C
LW-11345	(16 words) : パスワード (セルラーネットワーク)	R/W	R/C	R/C
LW-11361	(16 words) : 電話番号 (セルラーネットワーク)	R/W	R/C	R/C
LW-11377	(16bit) : 接続を停止(0に設定)/起動する(1に設定)(セルラーネットワーク)	R/W	R/C	R/C
LW-11378	(16bit) : 最後のエラーコード(0:成功, 1:間違った PIN コード, 2:SIM カードがない, 3:装置がない, 4:puk コードがロックされた, 5:その他) (セルラーネットワーク)	R	R	R
LW-11379	(16bit) : 接続エラーコード(0:装置がない, 1:接続が中断された, 2:接続中, 3:接続された) (セルラーネットワーク)	R	R	R
LW-11380	(16bit) :接続を停止する(0に設定する)/起動する (1に設定する) 連接 (USB テザリング)	R/W	R/C	R/C
LW-11381	(16bit) :接続状態(0:装置がない, 1:接続が切断された, 2:接続された, 3:失敗, 4:OS がサポートしない, 5:HMI がサポートしない) (USB テザリング)	R	R	R

22.3.38. Wi-Fi 設定

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-12365	Wi-Fi 設定を更新する(IP, サブネットマスク, デフォルトゲートウェイ, DNS) (ON に設定する)	R/W	R/C	R/C
LB-12666	Wi-Fi ホットスポットを無効にする(OFF に設定する)/有効にする(ON に設定する)	R/W	R/C	R/C
LB-12667	Wi-Fi ホットスポット設定を更新する(SSID, パスワード) (ON に設定する)	R/W	R/C	R/C
LW-11383	(16bit) : Wi-Fi 制御(1: 接続が切断された, 2: 接続された, 3: Wi-Fi 設定ダイアログボックスをポップアップする) (Wi-Fi)	R/W	R/C	R/C
LW-11384	(16bit) : エラーコード (0: 無, 1: 装置がない, 2: Wi-Fi 電波が OFF) (Wi-Fi)	R	R	R
LW-11385	(16bit) : 状態 (0: 停止された; 1: 接続中; 2: 接続された) (Wi-Fi)	R	R	R
LW-11386	(16 words) : 接続された SSID (Wi-Fi)	R	R	R
LW-11402	(16bit) : 信号強さ (0: 無, 1: 弱い, 2: 普通, 3: 良い, 4: 非常に良い) (Wi-Fi)*注 2	R	R	R
LW-11403	(16bit) : 国番号 (Wi-Fi)*注 1	R/W	R/C	R/C
LW-11404	(16bit) : Wi-Fi 電波 (0: ON, 1: OFF) (Wi-Fi)	R/W	R/C	R/C
LW-11410	(16bit) : HMI Wi-Fi IP 0 (HMI でのみ有效)	R/W	R/C	R/C
LW-11411	(16bit) : HMI Wi-Fi IP 1 (HMI でのみ有效)	R/W	R/C	R/C
LW-11412	(16bit) : HMI Wi-Fi IP 2 (HMI でのみ有效)	R/W	R/C	R/C
LW-11413	(16bit) : HMI Wi-Fi IP 3 (HMI でのみ有效)	R/W	R/C	R/C
LW-11414	(16bit) : HMI Wi-Fi ネットマスク 0 (HMI でのみ有效)	R/W	R/C	R/C
LW-11415	(16bit) : HMI Wi-Fi ネットマスク 1 (HMI でのみ有效)	R/W	R/C	R/C
LW-11416	(16bit) : HMI Wi-Fi ネットマスク 2 (HMI でのみ有效)	R/W	R/C	R/C
LW-11417	(16bit) : HMI Wi-Fi ネットマスク 3 (HMI でのみ有效)	R/W	R/C	R/C
LW-11418	(16bit) : HMI Wi-Fi ゲートウェイ 0 (HMI でのみ有效)	R/W	R/C	R/C
LW-11419	(16bit) : HMI Wi-Fi ゲートウェイ 1 (HMI でのみ有效)	R/W	R/C	R/C
LW-11420	(16bit) : HMI Wi-Fi ゲートウェイ 2 (HMI でのみ有效)	R/W	R/C	R/C
LW-11421	(16bit) : HMI Wi-Fi ゲートウェイ 3 (HMI でのみ有效)	R/W	R/C	R/C
LW-11422	(16bit) : HMI Wi-Fi メディアアクセス制御アドレス (MAC) 0	R	R	R

LW-11423	(16bit) : HMI Wi-Fi メディアアクセス制御アドレス (MAC) 1	R	R	R
LW-11424	(16bit) : HMI Wi-Fi メディアアクセス制御アドレス (MAC) 2	R	R	R
LW-11425	(16bit) : HMI Wi-Fi メディアアクセス制御アドレス (MAC) 3	R	R	R
LW-11426	(16bit) : HMI Wi-Fi メディアアクセス制御アドレス (MAC) 4	R	R	R
LW-11427	(16bit) : HMI Wi-Fi メディアアクセス制御アドレス (MAC) 5	R	R	R
LW-11428	(16bit) : HMI Wi-Fi ドメインネームシステム (DNS) サーバー IP 0	R/W	R/C	R/C
LW-11429	(16bit) : HMI Wi-Fi ドメインネームシステム (DNS) サーバー IP 1	R/W	R/C	R/C
LW-11430	(16bit) : HMI Wi-Fi ドメインネームシステム (DNS) サーバー IP 2	R/W	R/C	R/C
LW-11431	(16bit) : HMI Wi-Fi ドメインネームシステム (DNS) サーバー IP 3	R/W	R/C	R/C
LW-11432	(16bit) : 自動的に Wi-Fi IP アドレスを取得する(DHCP => 0 : off, 1 : on)	R/W	R/C	R/C
LW-12006	(16 words) : Wi-Fi ホットスポット SSID(有効にして適用する)	R/W	R/C	R/C
LW-12022	(32 words) : Wi-Fi ホットスポットパスワード(有効にして適用する)	R/W	R/C	R/C

 **Note**

1. ASCII で国番号(大文字)に入力してください。設定完了後、HMI を再起動してから有効になります。入力された国番号により、各地域の Wi-Fi 規定に従って HMI が接続する Wi-Fi チャンネルを設定します。
2. 信号強さは 1: 弱い (<-70 dBm), 2: 普通 (-60 ~ -70 dBm), 3: 良い (-50 ~ -60 dBm), 4: 非常に良い (>-50 dBm)に分けられます。

22.3.39. OPC UA サーバー

アド レス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-11435	(16bit) : OPC UA サーバー状態(0: 停止された, 1: 有効さ れた)	R	R	R
LW-11436	(16bit) : OPC UA サーバーエラーコード(0: 成功, 1 その 他: エラー)	R	R	R
LW-11437	(16bit) : OPC UA サーバーコントロールコマンド(0: 無, 1: 開始, 2: 停止)	R/W	R/C	R/C

22.3.40. E メール

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-12053	[イベントログ] E メールの送信に失敗した(状態は ON の場合)	R	R	R
LB-12054	[バックアップオブジェクト] E メールの送信に失敗し た(状態は ON の場合)	R	R	R
LW-9216	(16bit) : E メールをインポートした結果*注 1	R	R	R
LW-11444	(16bit) : 失敗したステップ(E メール)*注 2	R	R	R
LW-11445	(16bit) : エラーコード(E メール)*注 3	R	R	R

Note

1. インポートに成功した : 1、インポートが失敗した(ファイルが見つかれない) : 2、ファイ
ルフォーマットが間違ったまたはファイルがデコンパイルされなかった : 4。
2. 失敗ステップのエラーコードは以下のとおりです :

エラーコード	発生原因
0	CSMTP_NO_ERROR
100	WSA_STARTUP = Unable to initialise winsock2
101	WSA_VER = Wrong version of the winsock2
102	WSA_SEND = Function send() failed
103	WSA_RECV = Function recv() failed
104	WSA_CONNECT = Function connect failed
105	WSA_GETHOSTBY_NAME_ADDR = Unable to determine remote server

106	WSA_INVALID_SOCKET = Invalid winsock2 socket
107	WSA_HOSTNAME = Function hostname() failed
108	WSA_IOCTLSOCKET = Function ioctlsocket() failed
109	WSA_SELECT
110	BAD_IPV4_ADDR = Improper IPv4 address
200	UNDEF_MSG_HEADER = Undefined message header
201	UNDEF_MAIL_FROM = Undefined mail sender
202	UNDEF SUBJECT = Undefined message subject
203	UNDEF_RECIPIENTS = Undefined at least one recipient
204	UNDEF_RECIPIENT_MAIL = Undefined recipient mail
205	UNDEF_LOGIN = Undefined user login
206	UNDEF_PASSWORD = Undefined user password
207	BAD_LOGIN_PASSWORD = Invalid user login or password
208	BAD_DIGEST_RESPONSE = Server returned a bad digest MD5 response
209	BAD_SERVER_NAME = Unable to determine server name for digest MD5 response
300	COMMAND_MAIL_FROM = Server returned error after sending MAIL FROM
301	COMMAND_EHLO = Server returned error after sending EHLO
302	COMMAND_AUTH_PLAIN = Server returned error after sending AUTH PLAIN
303	COMMAND_AUTH_LOGIN = Server returned error after sending AUTH LOGIN
304	COMMAND_AUTH_CRAMMD5 = Server returned error after sending AUTH CRAM-MD5
305	COMMAND_AUTH_DIGESTMD5 = Server returned error after sending AUTH DIGEST-MD5
306	COMMAND_DIGESTMD5 = Server returned error after sending MD5 DIGEST
307	COMMAND_DATA = Server returned error after sending DATA
308	COMMAND_QUIT = Server returned error after sending QUIT
309	COMMAND_RCPT_TO = Server returned error after sending RCPT TO
310	MSG_BODY_ERROR = Error in message body
400	CONNECTION_CLOSED = Server has closed the connection
401	SERVER_NOT_READY = Server is not ready
402	SERVER_NOT_RESPONDING = Server not responding
403	SELECT_TIMEOUT =
404	FILE_NOT_EXIST = File not exist
405	MSG_TOO_BIG = Message is too big
406	BAD_LOGIN_PASS = Bad login or password
407	UNDEF_XYZ_RESPONSE = Undefined xyz SMTP response
408	LACK_OF_MEMORY = Lack of memory
409	TIME_ERROR = time() error
410	RECVBUF_IS_EMPTY = RecvBuf is empty
411	SENDBUF_IS_EMPTY = SendBuf is empty

412	OUT_OF_MSG_RANGE = Specified line number is out of message size
413	COMMAND_EHLO_STARTTLS = Server returned error after sending STARTTLS
414	SSL_PROBLEM = SSL problem
415	COMMAND_DATABLOCK = Failed to send data block
416	STARTTLS_NOT_SUPPORTED = The STARTTLS command is not supported by the server
417	LOGIN_NOT_SUPPORTED = AUTH LOGIN is not supported by the server

3. [システムパラメータ設定] » [E メール]タブでの[エラーメッセージ]を選択し、そして一個の開始ワードアドレスを指定(長さは自由に調整できます)して E メールサーバーが送信したエラーメッセージを表示させることができます。

22.3.41. cMT Viewer と診断器

アドレス	記述	読み取り(R)/ 書き込み(W)/ コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-12656	診断器を有効にする(ON に設定する)、無効にする(OFF に設定する)	R/W	R/C	R/C
LB-12657	診断器パスワード不要(状態は ON の場合)	R/W	R/C	R/C
LW-11756	(4 words) : 診断器パスワード	R/W	R/C	R/C
LW-11839	(16bit) : 接続した clients 数(cMT Viewer)	R	R	R
LW-11840	(8 words) : client 1 ユーザー名	R	R	R
LW-11848	(8 words) : client 2 ユーザー名	R	R	R
LW-11856	(8 words) : client 3 ユーザー名	R	R	R
LW-11864	(8 words) : client 4 ユーザー名	R	R	R
LW-11872	(8 words) : client 5 ユーザー名	R	R	R
LW-11880	(8 words) : client 6 ユーザー名	R	R	R
LW-11888	(8 words) : client 7 ユーザー名	R	R	R
LW-11896	(8 words) : client 8 ユーザー名	R	R	R
LW-11904	(8 words) : client 9 ユーザー名	R	R	R
LW-11912	(8 words) : client 10 ユーザー名	R	R	R
LW-11940	(16bit) : client 1 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11941	(16bit) : client 1 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11942	(16bit) : client 1 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11943	(16bit) : client 1 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11944	(16bit) : client 2 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11945	(16bit) : client 2 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R

LW-11946	(16bit) : client 2 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11947	(16bit) : client 2 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11948	(16bit) : client 3 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11949	(16bit) : client 3 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11950	(16bit) : client 3 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11951	(16bit) : client 3 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11952	(16bit) : client 4 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11953	(16bit) : client 4 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11954	(16bit) : client 4 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11955	(16bit) : client 4 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11956	(16bit) : client 5 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11957	(16bit) : client 5 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11958	(16bit) : client 5 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11959	(16bit) : client 5 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11960	(16bit) : client 6 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11961	(16bit) : client 6 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11962	(16bit) : client 6 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11963	(16bit) : client 6 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11964	(16bit) : client 7 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11965	(16bit) : client 7 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11966	(16bit) : client 7 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11967	(16bit) : client 7 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11968	(16bit) : client 8 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11969	(16bit) : client 8 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11970	(16bit) : client 8 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11971	(16bit) : client 8 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11972	(16bit) : client 9 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11973	(16bit) : client 9 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11974	(16bit) : client 9 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11975	(16bit) : client 9 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11976	(16bit) : client 10 IPO (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11977	(16bit) : client 10 IP1 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11978	(16bit) : client 10 IP2 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
LW-11979	(16bit) : client 10 IP3 (IP アドレス = IPO:IP1:IP2:IP3)	R	R	R
PLW-11998	(8 words) : 制御トークンリリースのカウントダウン	R	無し	無し

22.3.42. CODESYS 情報

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LW-11760	(16bit) : CODESYS フームウェア状態(0:エラー, 1:開始, 2:停止)	R	R	R
LW-11761	(16bit) : CODESYS アプリケーションソフト状態(0:エラー, 1:開始, 2:停止)	R	R	R
LW-11762	(16bit) : CODESYS ログイン状態(0:エラー, 1:ログイン, 2:ログアウト)	R	R	R
LW-12059	(16bit) : CODESYS フームウェアバージョン(年)	R	R	R
LW-12060	(16bit) : CODESYS フームウェアバージョン(月)	R	R	R
LW-12061	(16bit) : CODESYS フームウェアバージョン(日)	R	R	R

22.3.43. 他の機能

アドレス	記述	読み取り(R)/書き込み(W)/コントロール(C)		
		ローカル HMI	マクロ	リモート HMI
LB-9000~LB-9009	リセット時の状態は ON	R/W	R/C	R/C
LB-9010	データ転送書き込み指示	R	R	R
LB-9011	データ転送読み取り指示	R	R	R
LB-9012	データ転送実行指示	R	R	R
LB-9016	リモート HMI をローカル HMI に接続する(状態は ON の場合)	R	R	R
LB-9017	PLC コントロールオブジェクト[切替ウインドウ]のライトバック]機能を無効にする	R/W	R/C	R/C
LB-9039	ファイルバックアップの動作状態(バックアップ中の状態は ON)	R	R	R
LB-9045	memory-map 通信に失敗した(状態は ON の場合)	R	R	R
LB-9049	ウォッチドッグ機能[有効にする(ON)/無効にする(OFF)(LW-11456 でウォッチドッグタイムアウトを設定する)]*注 1	R/W	R/C	R/C
LB-12356	ウェブストリーミングを有効にする(ON に設定する) / 無効にする(OFF に設定する)	R/W	R/C	R/C

LB-12357	ウェブストリーミング状態(ON:有効 / OFF: 無効)	R	R	R
LB-12358	HMI 上のオフラインシミュレーションを有効にする (ON に設定する) / 無効にする(OFF に設定する) *注 5	R/W	R/C	R/C
LB-12361	操作ログ機能の状態 (OFF :無効 ON : 有効)	R	R	R
LB-12664	Cache Hit (cMT-G03)	R	R	R
LW-9006	(16bit):本機に接続しているリモート HMI の数	R	R	R
LW-9024	(16bit):memory link システムレジスタ	R/W	R/C	R/C
LW-9032	(8 words):履歴データを SD カード, USB ディスクにバックアップするフォルダ名*注 3	R/W	R/C	R/C
LW-9050	(16bit):現在、表示しているベースウインドウ番号	R	R	R
PLW-9050	(16bit):現在、表示しているベースウインドウ番号(タブレットで)	R	N/A	N/A
LW-9134	(16bit):現在、使用している言語*注 2	R/W	R/C	R/C
PLW-9134	(16bit):現在、使用している言語(タブレットで)*注 2	R/W	N/A	N/A
LW-9900	(16bit):HMI 実行モード(0:ノーマルモード, 1~3:テストモード(COM1~COM3 を使用する)	R/W	R/C	R/C
LW-10762	(8 words) : スロット 1 ユーザー名	R/W	R/C	R/C
LW-10770	(8 words) : スロット 2 ユーザー名	R/W	R/C	R/C
LW-10778	(8 words) : スロット 3 ユーザー名	R/W	R/C	R/C
LW-10814	(16bit):Weintek HMI に接続する(0:無, 1:接続中)*注 4	R	R	R
LW-11456	(16bit):ウォッチドッグタイムアウト(3 ~ 10), 単位:秒 *注 1	R/W	R/C	R/C

Note

- LB-9049 ウォッチドッグ機能を使用すると、HMI が指定された秒数内で正常に稼動し続けることができなくなる場合、ウォッチドッグが自動的にシステムを再起動します。
- オブジェクト上の文字内容を多言語で表現したい時、ラベルタグライブラリ以外、システムレジスタ LW-9134 を合わせて使用することが必要です。LW-9134 の有效的な設定値範囲は 0~23 で、このワードアドレス内の数値は HMI にダウンロードした言語の種類に関連します。ダウンロードしたファイルで全ての言語にチェックマークを入れていない場合、LW-9134 の使用方式は変わります。例えば:ユーザーがラベルタグライブラリに言語 1(中国語(繁体字))、言語 2(中国語(簡体字))、言語 3(英語)、言語 4(フランス語)、言語 5(日本語)の 5 種類の言語を作成した場合、もしユーザーは言語 1、言語 3、言語 5だけをダウンロードすれば、この場合、LW-9134 内の数値が対応する言語種類は 0→言語 1 中国語(繁体字)、1→言語 3(英語)、2→言語 5(日本語)になります。オプションリストオブジェクト LW-9134 で言語を切り替える例は下記に示す通りです:

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サン

フルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

3. システムは HMI 名をデフォルトのフォルダ名にします。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

4. USB Host を Weintek HMI に接続する時、このアドレスは 1 に設定されます。これにより、もう一台の Weintek HMI の USB クライアントは正常に運転しているのかを検証できます。

5. 本機能を通し、HMI で通信モードをオフラインシミュレーションに切り替えられます。オフラインシミュレーションモードに切り替えたら、たとえ HMI が正しく装置を接続していなくても、Device No Response メッセージがポップアップされなく、それに装置に関連するオブジェクトは依然に表示、操作されるが、読み取った/書き込んだ値は実際に PLC に保存されません。

23. HMI がサポートする プリンターの種類

本章では、HMI がサポートするプリンターの種類及び設定する手順について説明します。

23.1. サポートするプリンターの種類	23-2
23.2. 一台のプリンターを新規追加し、プリントをトリガーする方法	23-5
23.3. cMT HMI でネットワークプリンターをインストールする	23-8
23.4. cMT HMI で PPD ファイルをインストールする	23-11

23.1. サポートするプリンターの種類

HMI がサポートするプリンタードライバーは大体下記の数種類があります。

プリンターの種類	記述
● SP-M, D, E, F	<p>このドライバーは EPSON ESC シリアルポートプリンタープロトコルを使用します。シリアルポートで接続します。プリンターに合わせ、通信パラメータを調整してください。[各行のピクセル数]を正確に設定し、それにプリンターのデフォルトに超えないこと:</p> <p>100pixels: 1610 モデルのプリンター; 220pixels: 2407, 4004 モデルのプリンター。 SP-E1610SK(ページ幅 45mm), SP-E400-4S(ページ幅 57.5mm)</p> <p>中国以外のユーザーにこのタイプの SP プリンターのご使用をお勧めします。</p> <p>北京迅普:http://www.siupo.com</p>
● EPSON ESC/P2 シリーズ	<p>シリアルポートで接続します。プリンターに合わせ、通信パラメータを調整してください。このドライバは EPSON ESC/P2 プリンター通信プロトコルを使用します。</p> <p>インパクトプリンター: LQ-300, LQ-300+, LQ-300K+(RS-232), LQ-300+II(RS-232)</p> <p>インクジェットプリンター: StylusPhoto750</p> <p>レーザープリンター: EPL-5800</p>
● HP PCL シリーズ(USB)	<p>USB ポートで接続します。HP プリンターの PCL5 通信プロトコル、または PostScript3 のプリンター制御コマンドをサポートします。</p> <p>PCL プリンター制御言語のサポート特性により、PCL5 以降、または最新バージョンのプリンターは全部 PCL5 プロトコルをサポートできます。</p>

- **Axiohm A630**



Axiohm マイクロプリンターです。シリアルポートで接続します。プリンターに合わせ、通信パラメータを調整してください。

- **SPRT**



シリアルポートで接続します。プリンターに合わせ、通信パラメータを調整してください。[各行のビクセル数]を正確に設定し、それにプリンターのデフォルト値“100”を超えないこと。

サポートモデル:

SP-DN40SH: インパクトプリンター

SP-RMDIII40SH: 感熱式プリンター

- **EPSON TM-L90**



シリアルポートで接続します。プリンターに合わせ、通信パラメータを調整してください。各行のビクセル数]を正確に設定し、それにプリンターのデフォルト値“576”を超えないこと。

- **EPSON TM-T70**



シリアルポートで接続します。プリンターに合わせ、通信パラメータを調整してください。各行のビクセル数]を正確に設定し、それにプリンターのデフォルト値“576”を超えないこと。断裁モードが選べ、[裁断しない]と[半分カット]のオプションがあります。

- **BRIGHTEK WH-A19**



サポートするモデル :

A92R10-00E72A:末尾番号 72 のは 16 進数を示し、A は 5~9V び意味で、このモデルは A6 16 インパクトプリンターと同じです。

- **BRIGHTEK WH-E19**



シリアルポートで接続します。プリンターに合わせ、通信パラメータを調整してください。

- **BRIGHTEK WH-22(煌)**



サポートするモデル：

E22R10-00E725:A716 インパクトプリンターと同じで、A7 モデルは A72R90-31E72A

E221R90-00E11740GA:RS-485 インターフェースを使用し、RS-232↔485 変換器を使用する必要があります。

- **BRIGHTEK WH-C1/C2**



シリアルポートで接続します。プリンターに合わせ、通信パラメータを調整してください。断裁モードが選べ、[裁断しない]、[半分カット]、[フルカット]のオプションがあります。

- リモートプリントサーバー

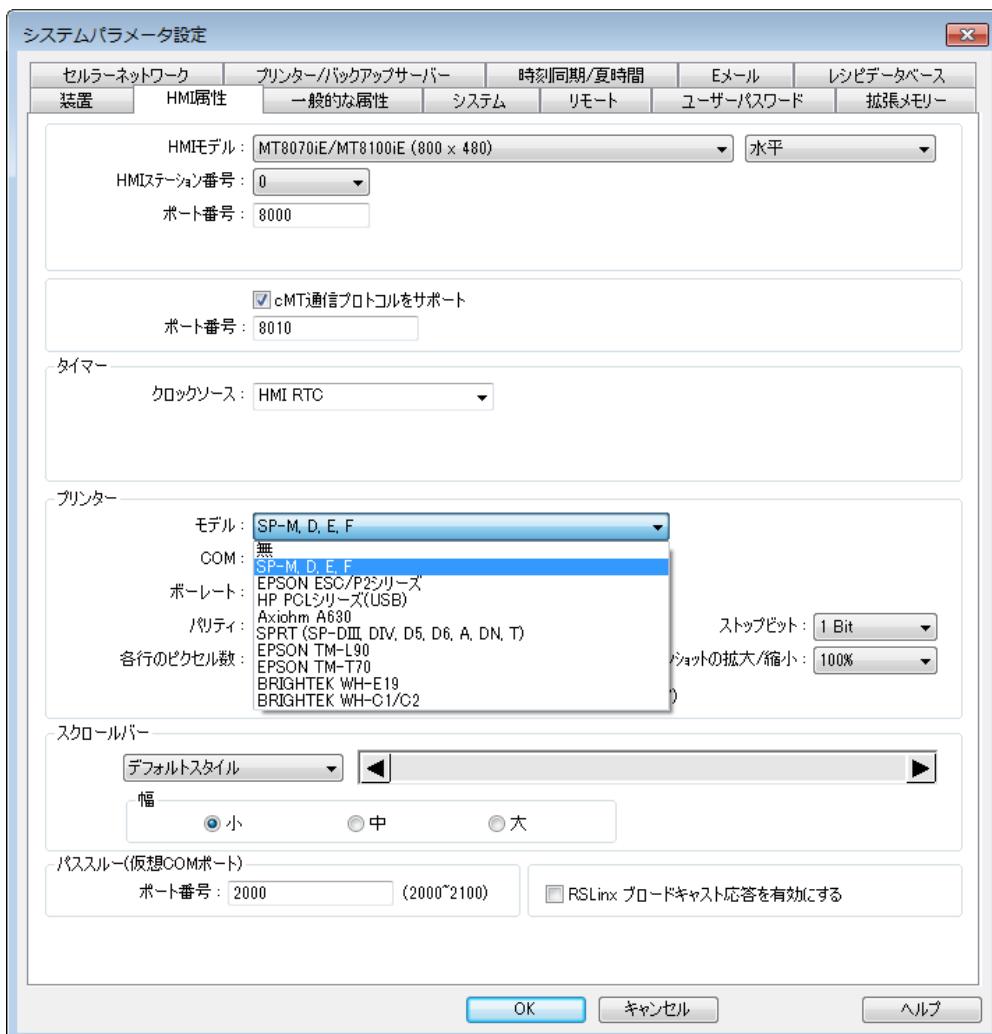


HMI はイーサネットを通し、PC でのプリンターを接続、そして EasyPrinter を使ってプリントを実行できます。EasyPrinter は MS Windows システムで動作されるので、市場での大部分のプリンターをサポートしています。

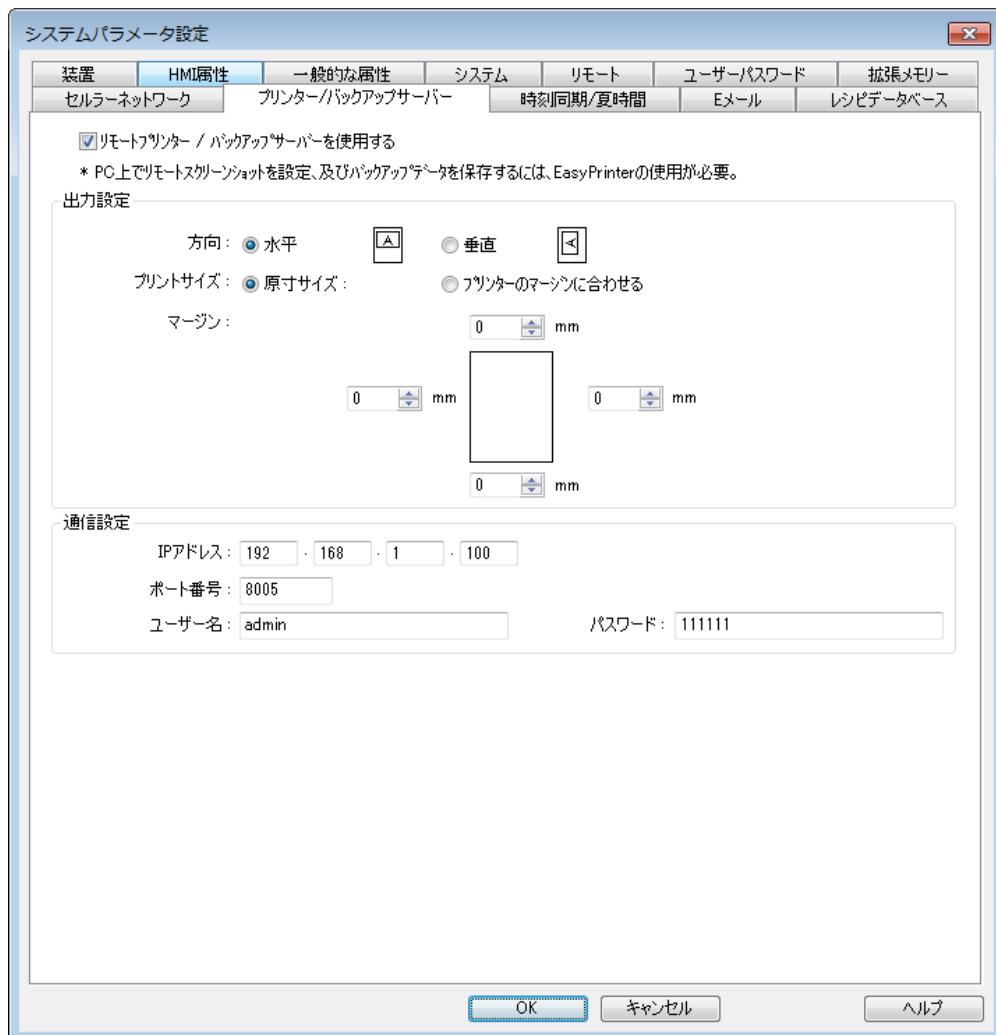
23.2. 一台のプリンターを新規追加し、プリントをトリガーする方法

1. プリンタータイプを新規追加します。

- [システムパラメータ設定] » [HMI 属性] タブで接続したいプリンターモデルを選択し、正確に関連パラメータを設定します。



- リモートプリントサーバーに接続したい場合、[システムパラメータ設定] » [プリント/バックアップサーバー] タブを選択し、正確に関連パラメータを設定します。

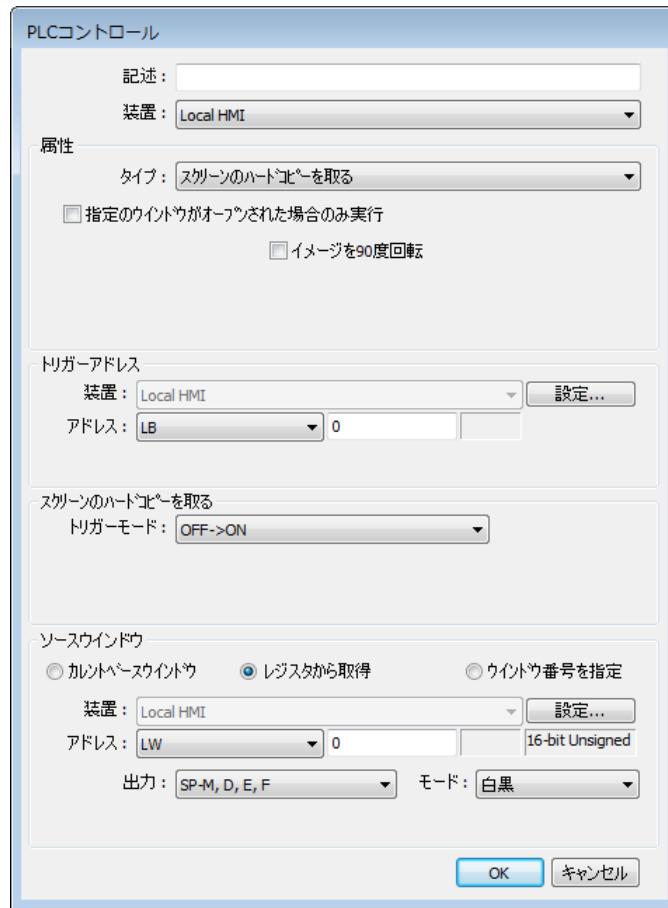


2. プリントをトリガーします。

- ユーザーは“ファンクションキー”オブジェクトで直接にプリント機能をトリガーできます。



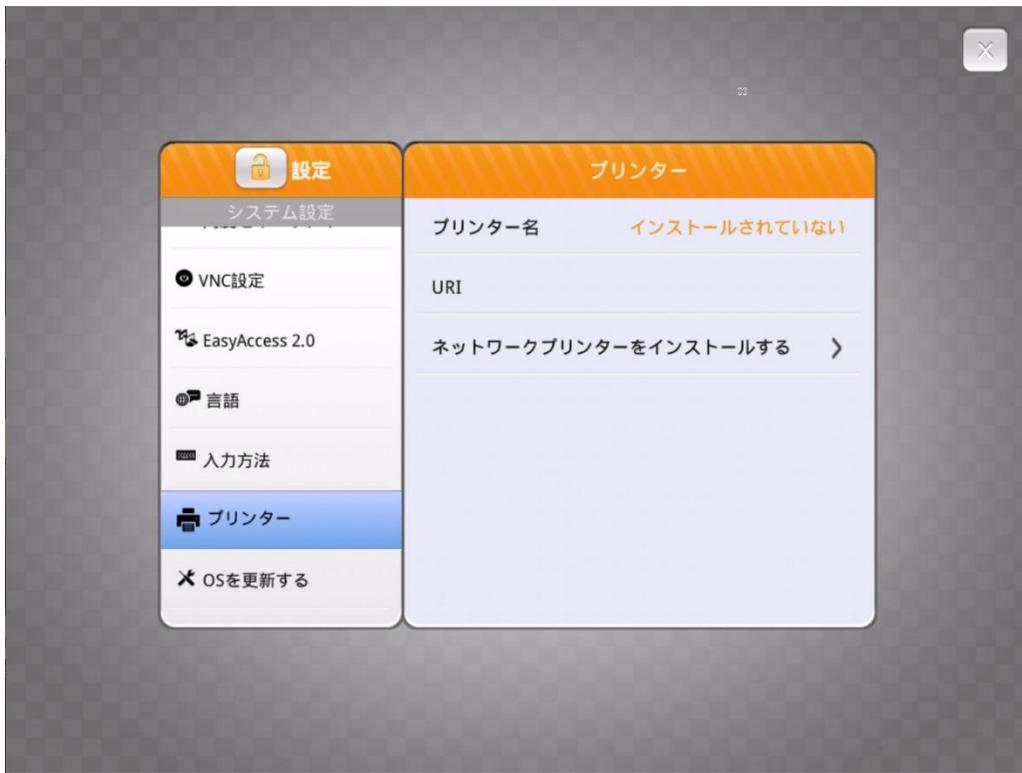
- 或いは、“PLC コントロール”オブジェクトの[スクリーンのハードコピーを取る]を使用し、予め定義されたビットアドレスでプリント機能をトリガーします。



23.3. cMT HMI でネットワークプリンターをインストールする

上記のプリンター以外、cMT もシステム設定の[ネットワークプリンターをインストールする]機能でもっと多くのプリンターを利用できます。

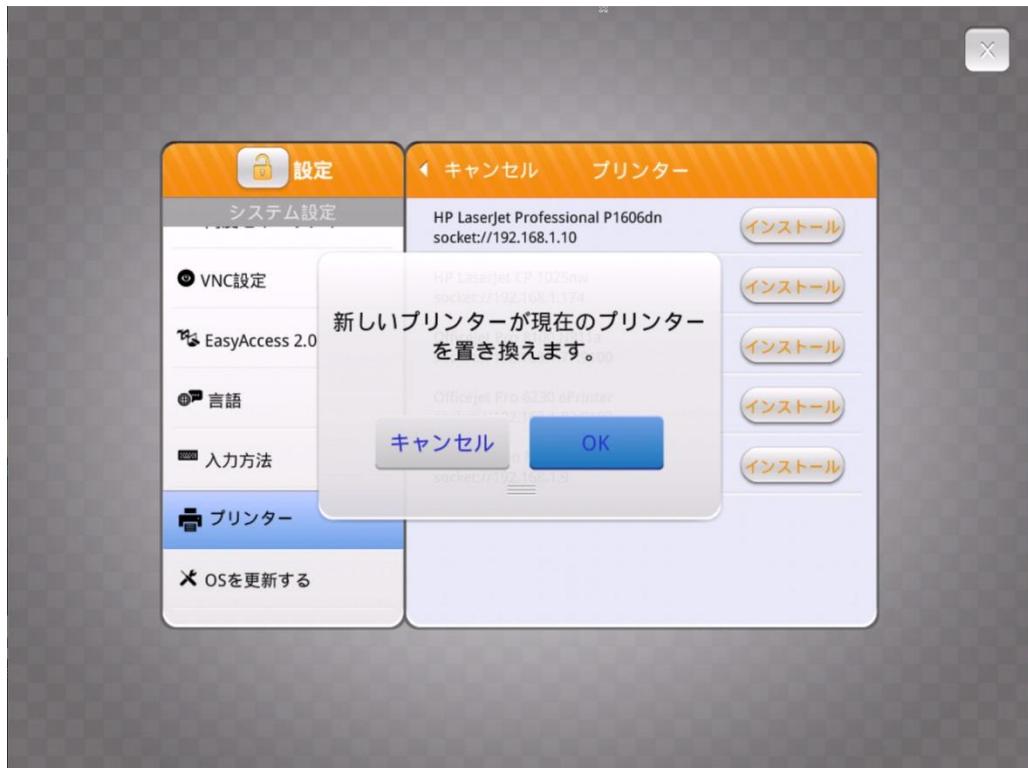
1.  をクリックしてシステム設定をオープンします。[設定]ページをログインした後、[プリンター] » [ネットワークプリンターをインストールする]を探します。



2. HMI は数秒をかけてドメイン内にあるネットワークプリンターを探し出します。



3. インストールしたいネットワークプリンターを選択します。[インストール]をクリックしてドライバをインストールします。一回では一台のネットワークプリンターだけをインストールできるので、新しいプリンタードライバは古いドライバを入れ替わります。

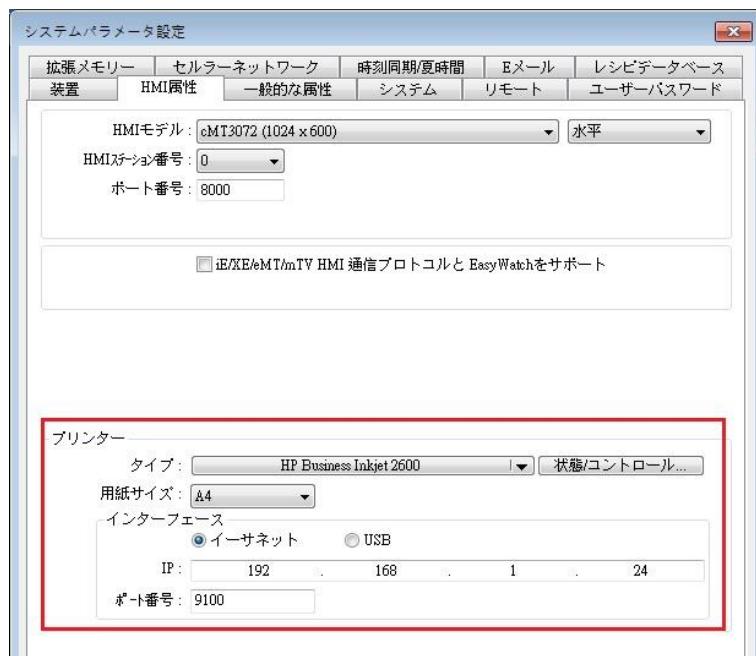


4. インストールが完成した後、そのネットワークプリンターを利用できます。

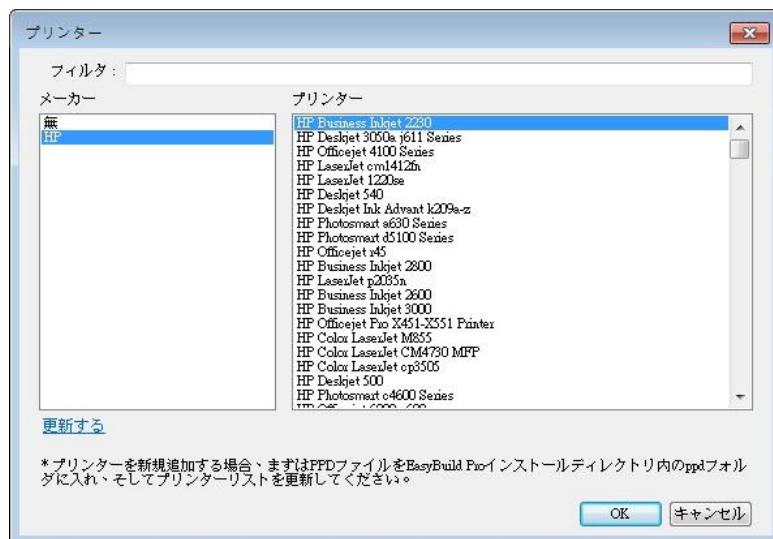
23.4. cMT HMI で PPD ファイルをインストールする

上記の方法以外、cMT HMI も PPD ファイルをインストールすることでもっと多くのプリンターを使用できます。

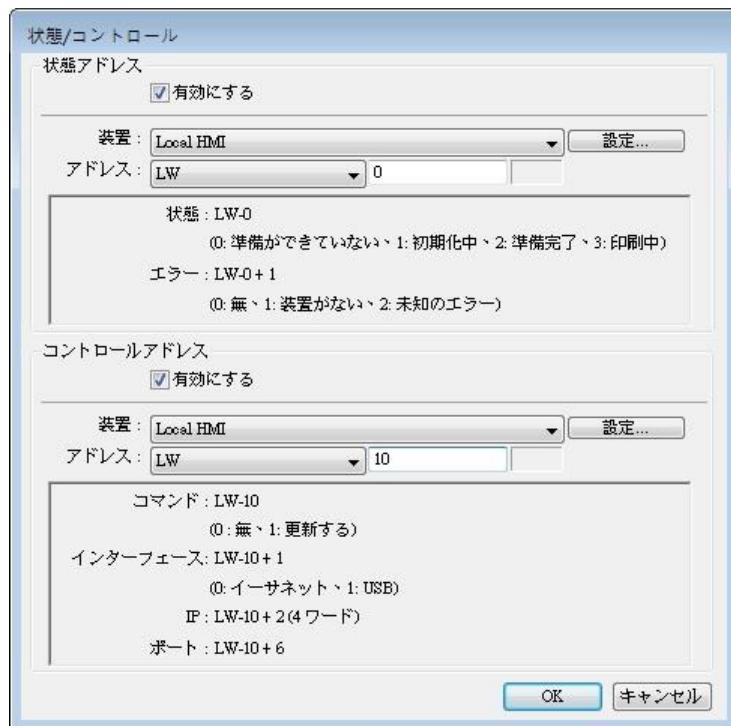
- [システムパラメータ設定] » [HMI 属性]タブをオーブンします。



- 接続したいプリンターを選択します。プリンターを追加する場合、PPD ファイルを EasyBuilder Pro インストールの `ppd` フォルダに入れ、そして[更新する]をクリックすれば良いです。



- [状態アドレス]を有効にすることでプリンターの状態を監視できます。または[コントロールアドレス]でオンラインでプリンターの接続パラメータを変更します。



Note

- 本機能は cMT-SVR 及び cMT Gateway をサポートしません。
- cMT シリーズと HP プリンターとの互換性が良好のため、HP プリンターの使用をお勧めします。ただし事前にテストする必要があります。
- cMT シリーズ HMI でプリンターをテストする際に、優先に[ネットワークプリンターをインストールする]方法でドライバをインストールしてください。これで上手く行かなかったら、PPD ファイルを追加する方法でドライバをインストールしてください。
- HMI で既にプリンターデバイスドライバがインストールしてある場合、プロジェクト内に設置しているプリンターを使用したかったら、[優先に HMI 側のプリンターを使用します]にチェックマークを入れないでください。また HMI 側でコントロールアドレスを使用し、コマンド 1: 更新を出せば、プリンタードライバソースをプロジェクト内に設置しているプリンターに指定することになります。



24. Recipe Editor

本章では、Recipe Editor を使用する方法について説明します。

24.1. 概要	24-2
24.2. レシピデータ/拡張メモリーエディタ設定.....	24-2
24.3. レシピ記録の設定	24-4

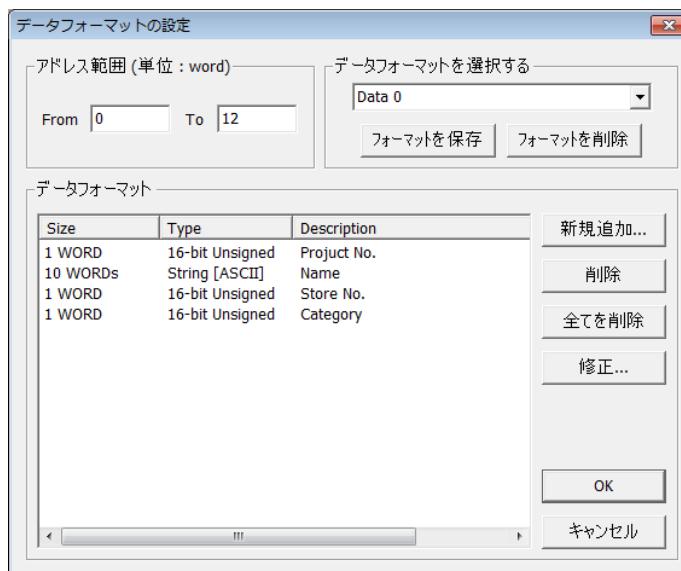
24.1. 概要

Recipe Editor は HMI が使用するレシピデータを作成でき、それに既存のレシピデータをオープン、編集することもできます。

また、EasyBuilder Pro はもう一つのレシピデータを編集するツール—レシピ記録を提供しています。本機能を使用するには、まず EasyBuilder Pro の[データ/履歴] » [レシピデータベース]でレシピを定義し、そして[レシピ閲覧オブジェクト]を使ってレシピ内容を表示します。以下はこの 2 種類のエディタを使用する方法を説明します。

24.2. レシピデータ/拡張メモリーエディタ設定

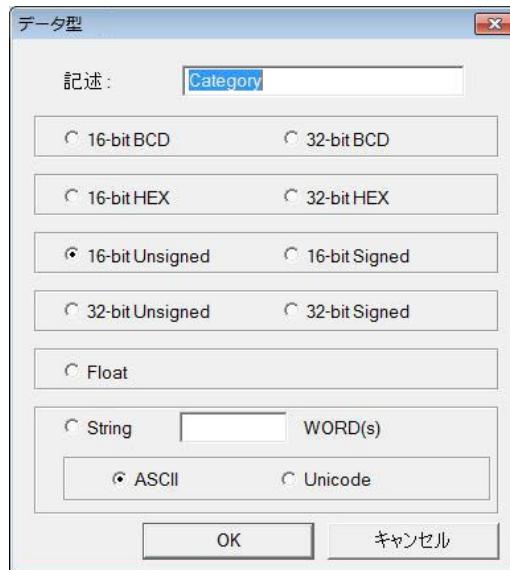
1. Utility Manager で[レシピデータ/拡張メモリーエディタ]をクリックします。
2. .rcp 或いは .emi ファイルを新規作成する必要があり、[ファイル] » [ファイルを新規作成する]をクリックしてください。
3. アドレス範囲及びデータフォーマットを設定します。



設定	記述
アドレス範囲	開始アドレス及び終了アドレスを入力します。単位：ワード。
データフォーマットを選択する	定義したデータフォーマットを保存し、次回使用したい時にロードすることができます。見本は "dataEX.fmt" ファイルにセーブされ、EasyBuilder Pro インストールディレクトリに保存されます。
データフォーマット	データフォーマットエリアで新規データフォーマットを編集できます。

4. [新規追加]をクリックしたら、データタイプ編集ウインドウが、下記に示す通りポップアップします。[記述]欄にデータタイプ名を入れ、そしてデータフォーマットを選択してください。[String]を選択すれば、ワードの長さ及びフォーマットタイプ : [ASCII]或いは[Unicode]

を選択します。



5. データフォーマットを定義完了後、[OK]をクリックすればレシピデータを編集できます。

ID	ADDRESS	Project No.	Name	Store No.	Category
0	0	0	shampoo	9	4
1	13	1	knife	1	5
2	26	2	chair	3	2
3	39	3	coffee	3	3
4	52	4	pencil	6	5
5	65	5	muffin	6	3
6	78	6	donut	5	3
7	91	7	DVD	9	6
8	104	8	postcard	4	5
9	117	9	maps	5	5
10	130	10	camera	2	1

本例中のデータフォーマット長さは 13 ワードで、それで 13 個ワードごとを一セットのレシピと視して使用できます。上記に示した通り、

第一セットの“Product no.”は address0、“Name”アドレスは address 1~10、“Store No.”は address 11、“Category”は address 12;になります。

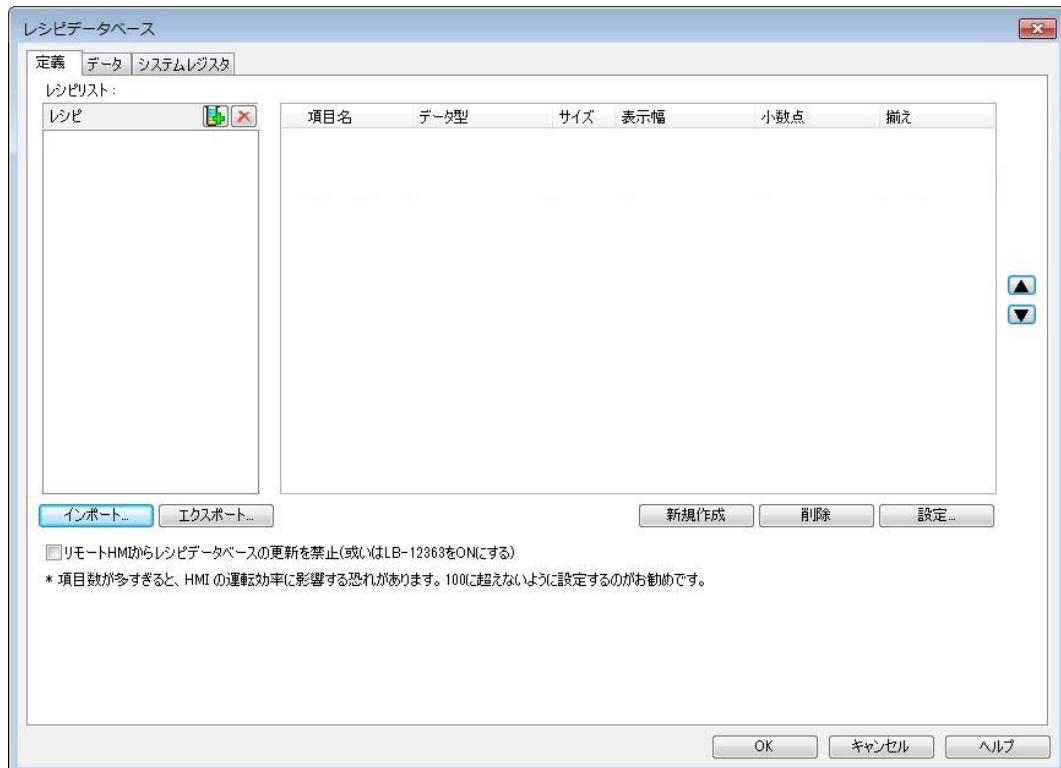
第二セットの“Product no.”は address 13、“Name”アドレスは address 14~23、“Store No.”は address 24、“Category”は address 25...このように類推します。

Note

- 編集し終えたレシピデータを .rcp、.emi または .csv ファイルにセーブできます。セーブされた .rcp ファイルは Utility Manager 或いは外部記憶装置(USB ディスク或いは SD カード)で HMI にダウンロードできます。.emi のほうは直接に外部記憶装置に保存し、HMI に挿入してロードでき、拡張アドレス EM に当たります。

24.3. レシピ記録の設定

1. EasyBuilder Pro の[データ/履歴] » [レシピデータベース]を選択します。
 [定義]タブでレシピ項目名、データ型と他の関連属性など、レシピのフォーマットを設定します。



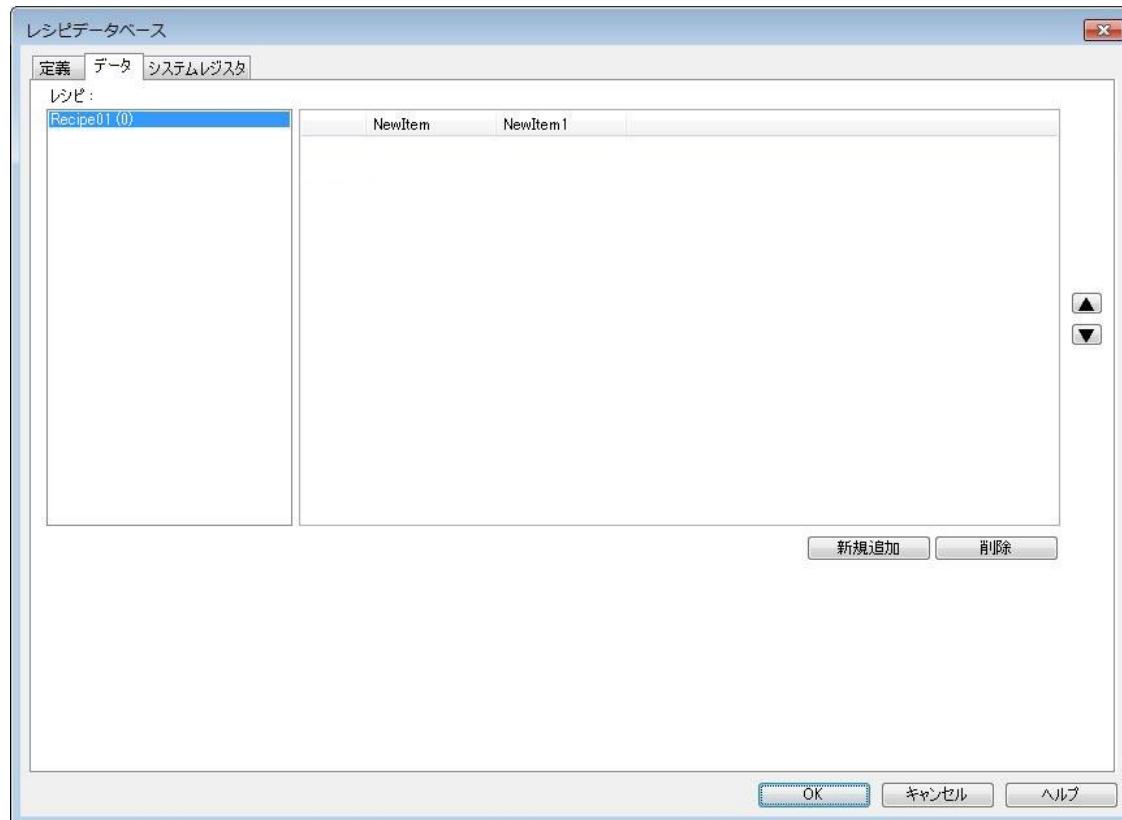
設定	記述
レシピリスト	レシピを追加/削除します。最大 100 個のレシピを作成できます。
新規作成	レシピ項目を新規作成します。最大 1000 レコードのレシピ項目を作成できます。
設定	選択したレシピ項目の関連属性を設定します。下記説明をご参照ください。
削除	削除したい項目を選択したら、削除できます。
インポート	レシピ定義ファイルをインポートします。
エクスポート	レシピ定義ファイルをエクスポートします。エクスポートされたファイルには、レシピデータベース記録内の内容が含まれません。

[設定]を押した後：



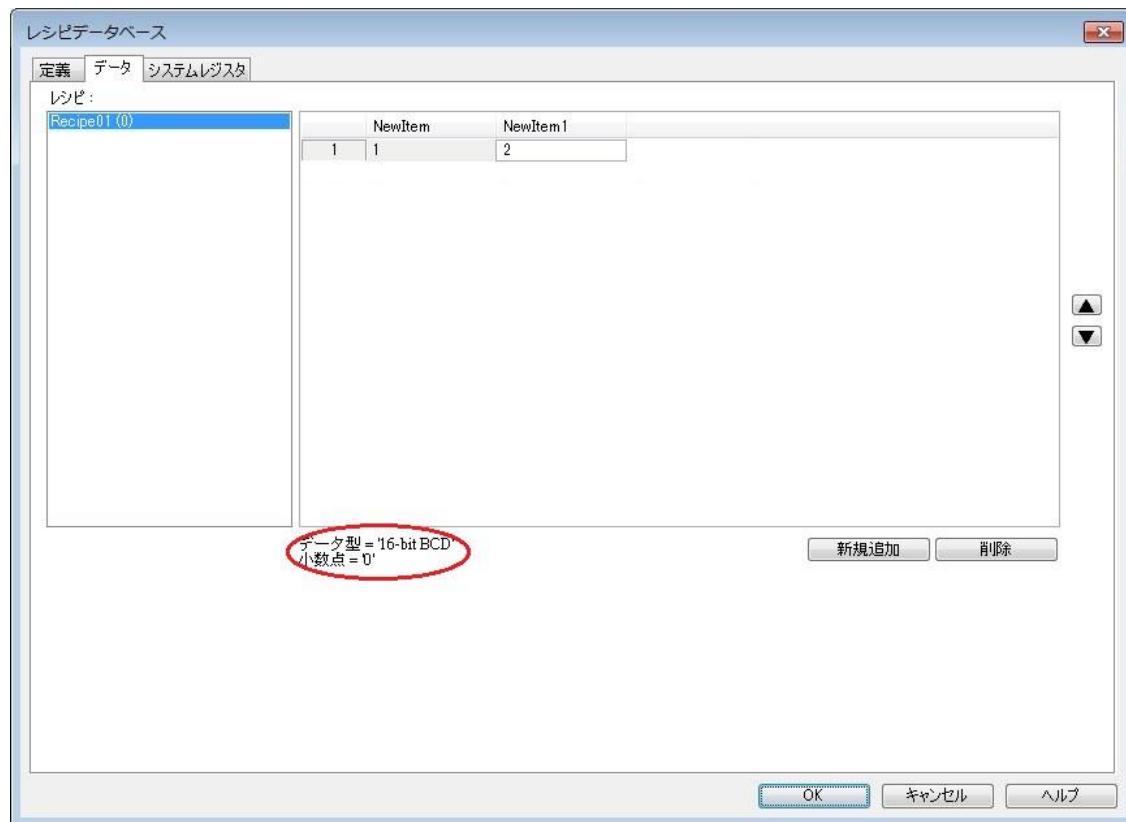
設定	記述
名前	レシピ項目の名前を入力します。
表示タイプ	項目のデータ型を設定します。
項目サイズ(ワード)	項目のサイズを設定します。単位：ワード。
表示幅(文字)	項目の表示字数を設定します。
小数点以下の桁数	データ表示時、小数点以下の桁数を設定します。
揃え	表示時の揃え方を選択します。左揃え、中央、右揃えが選択できます。

2. [データ]タブでは、左側の Recipe01、及び右上側のレシピ項目の名前は、[定義]タブで設定されたレシピから取ったものです。続いてはレシピの内容を定義します。



設定	記述
レシピ	レシピの一覧表です。ここでは[定義]タブにあるレシピが見られ、括弧内の数字はレシピデータのレコード数を指しています。
新規追加	各項目のフォーマットに従って、レシピの内容を編集し始められます。
削除	編集したレシピ内容を削除します。
矢印キー	矢印キーで選択したレシピ項目の資料を上へ/下へ移動します。

3. 各項目のフォーマットに従って定義し、[追加]ボタンを押した後、レシピ項目の内容を編集できます。各項目のフォーマットは、欄をクリックした後、下方に表示され、ユーザーは項目のフォーマットに応じ所要する内容を入力でき、そして[OK]を押してセーブします。



Note

- 一个のプロジェクトには最大 100 個のレシピデータベースを作成できます。一个のレシピデータベースには最大 1000 個のレシピを含むことができます。
- 一个のレシピには最大 10000 レコードの項目を含むことができます。
- 一个のレシピデータベースには最大 2000 ワードのデータを使用できます。これを超える

と、コンパイルができません。

- レシピデータベース名、及びレシピ項目は英数字のみサポートします。
- レシピがコンパイルされた後、.exob ファイルに保存し、HMI にダウンロードされます。このレシピデータは他のプロジェクトに共用されません。プロジェクトをダウンロード後、再度レシピデータベースを使ってレシピ内容を変更し、それに HMI にダウンロードする必要があったら、必ず[レシピデータベースを削除]にチェックマークを入れてください。チェックマークを入れていないと、HMI は元のレシピデータの内容を維持します。



- ▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

25. EasyConverter

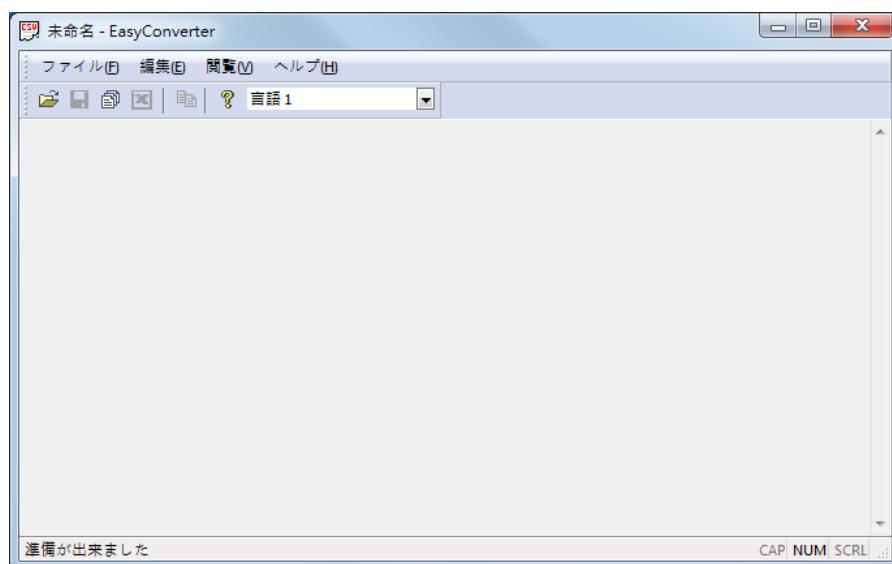
本章では、EasyConverter を使用する方法、及びその関連設定について説明します。

25.1. 概要	25-2
25.2. データサンプリングファイルを Excel にエクスポートする	25-2
25.3. イベントログファイルを Excel にエクスポートする	25-4
25.4. 操作ログファイルを Excel にエクスポートする	25-5
25.5. マルチファイル変換	25-6
25.6. スケーリング機能	25-7
25.7. バッチファイルの規則	25-8

25.1. 概要

EasyConverter は HMI から取ったデータサンプリングファイル、イベントログファイル或いは操作ログファイルを読み取り、そして Excel フォーマットに変換することができます。

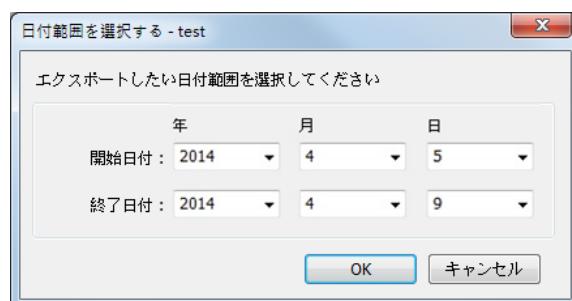
- Utility ManagerEx に[データ変換]>>[EasyConverter]をクリックする。
- EasyBuilder Pro のメニューで[イベントログ/データサンプリングファイル変換器]を選択する。



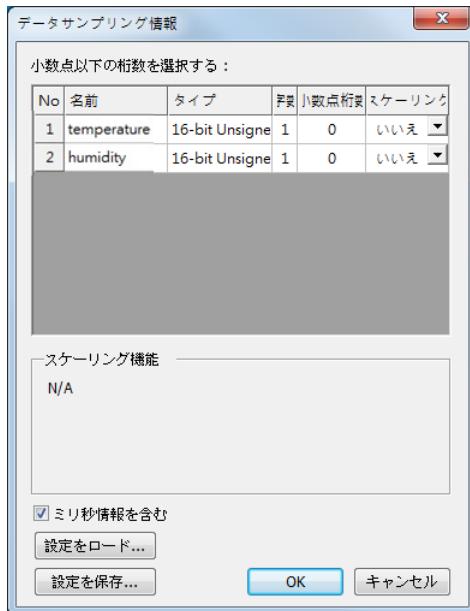
このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

25.2. データサンプリングファイルを Excel にエクスポートする

- オープンしたデータサンプリングファイルのフォーマットは DB で、それにファイル内では一日以上のデータを含めている場合、調べたい日付範囲を選択できます。(オープンしたファイルフォーマットは DTL だったら、このステップが省略されます)



- この時、下記に示すとおり、設定ウインドウがポップアップされます。ニーズに応じ関連設定をしてください。



3. [OK]を押した後、データサンプリング記録は以下のとおりに表示されます。そして[Excelにエクスポートする]を押せば、Excel フォーマットに変換することができます。

Date	Time	Millisecond	temperature	humidity
2014/4/5	04:03:30	46	0	0
2014/4/5	04:03:33	459	0	0
2014/4/5	04:03:36	456	0	0
2014/4/5	04:03:39	456	0	0
2014/4/5	04:03:42	457	0	0
2014/4/5	04:03:45	457	0	0
2014/4/5	04:03:48	457	10	0
2014/4/5	04:03:51	458	10	23
2014/4/5	04:03:54	457	10	23
2014/4/5	04:03:57	458	42	23
2014/4/5	04:04:00	456	14	67
2014/4/5	04:04:03	457	14	55
2014/4/5	04:04:06	459	14	55
2014/4/7	04:04:09	264	14	55
2014/4/7	04:04:12	209	23	55
2014/4/7	04:04:15	21	23	96
2014/4/7	04:04:18	209	23	96

4. Excel ファイルは以下のとおりに示しています。

A	B	C	D	E	F
1	Date	Time	Millisecond	temperature	humidity
2	2014/4/5	4:03:30	46	0	0
3	2014/4/5	4:03:33	459	0	0
4	2014/4/5	4:03:36	456	0	0
5	2014/4/5	4:03:39	456	0	0
6	2014/4/5	4:03:42	457	0	0
7	2014/4/5	4:03:45	457	0	0
8	2014/4/5	4:03:48	457	10	0
9	2014/4/5	4:03:51	458	10	23
10	2014/4/5	4:03:54	457	10	23
11	2014/4/5	4:03:57	458	42	23
12	2014/4/5	4:04:00	456	14	67
13	2014/4/5	4:04:03	457	14	55
14	2014/4/5	4:04:06	459	14	55
15	2014/4/7	4:04:09	264	14	55
16	2014/4/7	4:04:12	209	23	55
17	2014/4/7	4:04:15	21	23	96

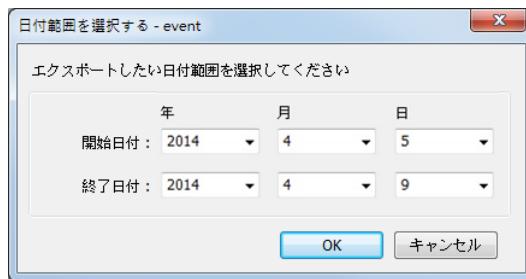


Note

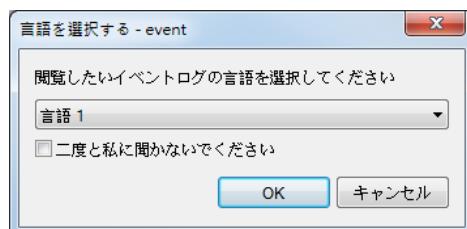
- EasyConverter でオープンする Excel ファイルの内容が、600 万セルを超えた場合、部分の内容だけを表示させます。(xls /xlsx にエクスポートした時、依然に完全なデータ内容を表示できます)
- xls /xlsx にエクスポートする時、下記の状況では異なるシートに分けられます。
 - 1 シート内の行数が 6 万行に超えた場合。
 - 1 シート内のセルが 150 万セルに超えた場合。

25.3. イベントログファイルを Excel にエクスポートする

1. オープンしたイベントログファイルのフォーマットは DB で、それにファイル内では一日以上のデータを含めている場合、調べたい日付範囲を選択できます。(オープンしたファイルフォーマットは EVT だったら、このステップが省略されます)



2. そのイベントログファイル DB に多言語を含めている場合、調べたい言語を選択できます。(オープンしたファイルフォーマットは EVT だったら、このステップが省略されます)



3. [OK]を押した後、イベントログは下記に示す通り表示されます。そして[Excel にエクスポートする]を押せば、Excel フォーマットに変換することができます。

	Event	Category	Date	Time	Message
2	0	0	2014/4/5	4:03:28	Bit OFF
3	2	0	2014/4/5	4:03:31	Bit OFF
4	0	0	2014/4/5	4:03:31	Bit ON
5	2	0	2014/4/5	4:03:32	Bit ON
6	0	0	2014/4/5	4:03:32	Bit OFF
7	1	0	2014/4/5	4:03:33	Bit OFF
8	2	0	2014/4/5	4:03:35	Bit OFF
9	0	0	2014/4/5	4:03:35	Bit ON
10	1	0	2014/4/5	4:03:36	Bit ON
11	2	0	2014/4/5	4:03:38	Bit ON
12	0	0	2014/4/5	4:03:38	Bit OFF
13	2	0	2014/4/5	4:03:40	Bit OFF
14	0	0	2014/4/5	4:03:40	Bit ON
15	2	0	2014/4/7	4:04:15	Bit ON

4. Excel ファイルは以下のとおりに表示されます。

A	B	C	D	E	F
1	Event	Category	Date	Time	Message
2	0	0	2014/4/5	4:03:28	Bit OFF
3	2	0	2014/4/5	4:03:31	Bit OFF
4	0	0	2014/4/5	4:03:31	Bit ON
5	2	0	2014/4/5	4:03:32	Bit ON
6	0	0	2014/4/5	4:03:32	Bit OFF
7	1	0	2014/4/5	4:03:33	Bit OFF
8	2	0	2014/4/5	4:03:35	Bit OFF
9	0	0	2014/4/5	4:03:35	Bit ON
10	1	0	2014/4/5	4:03:36	Bit ON
11	2	0	2014/4/5	4:03:38	Bit ON
12	0	0	2014/4/5	4:03:38	Bit OFF
13	2	0	2014/4/5	4:03:40	Bit OFF
14	0	0	2014/4/5	4:03:40	Bit ON
15	2	0	2014/4/7	4:04:15	Bit ON

Note

- フォームの第一行には"Event"欄が見られ、**0->イベントトリガー時;1->イベント確認時;2->イベントが正常に戻ったと示しています。**
- EasyConverter でオーブンする Excel ファイルの内容が、600 万セルを越えた場合、部分の内容だけを表示させます。(xls /xlsx にエクスポートした時、依然に完全なデータ内容を表示できます)
- xls /xlsx にエクスポートする時、下記の状況では異なるシートに分けられます：
- 1 シート内の行数が 6 万行に超えた場合。
- 1 シート内のセルが 150 万セルに超えた場合。

25.4. 操作ログファイルを Excel にエクスポートする

1. 操作ログファイル内では一日以上のデータを含めている場合、調べたい日付範囲を選択できます。



2. [OK]を押した後、操作ログは下記に示す通り表示されます。そして[Excel にエクスポートする]を押せば、Excel フォーマットに変換することができます。

ID	Date	Time	User_Name	Class	Window	Object_Name	Comment	Action	Address	Information
2	2014/4/28	6:47:57				10 NE_9	month	Set word	LW-9220 (32bit) : password	write 111
3	2014/4/28	6:47:59	:noname:	ADEF	10 NE_4	day		Set word	LW-9020 (16bit) : local day	write 29
4	2014/4/29	6:48:02	:noname:	ADEF	10 NE_5	hour		Set word	LW-9019 (16bit) : local hour	write 9
5	2014/4/29	9:48:10	:noname:	ADEF	10 NE_2	year		Set word	LW-9022 (16bit) : local year	write 2014
6	2014/4/29	9:48:13	:noname:	ADEF	10 NE_6	year		Set word	LW-9018 (16bit) : local year	write 2014
7	2014/4/29	9:48:16	:noname:	ADEF	10 NE_5	month		Set word	LW-9020 (16bit) : local month	write 50
8	2014/4/29	9:50:20	:noname:	ADEF	10 NE_8	month		Set word	LW-9219 (16bit) : user no. (1~12)	write 2
9	2014/4/29	9:50:22	:noname:	ADEF	10 NE_9	month		Set word	LW-9220 (32bit) : password	write 222
10	2014/4/29	9:50:26	:noname:	B	10 NE_3	month		Set word	LW-9021 (16bit) : local month	write 6

3. Excel ファイルは以下のとおりに表示されます。

A	B	C	D	E	F	G	H	I	J	K	L
ID	Date	Time	User_Name	Class	Window	Object_Name	Comment	Action	Address	Information	
2	2014/4/28	6:47:57				10 NE_9	month	Set word	LW-9220 (32bit) : password	write 111	
3	2014/4/28	6:47:59	:noname:	ADEF	10 NE_4	day		Set word	LW-9020 (16bit) : local day	write 29	
4	2014/4/29	6:48:02	:noname:	ADEF	10 NE_5	hour		Set word	LW-9019 (16bit) : local hour	write 9	
5	2014/4/29	9:48:10	:noname:	ADEF	10 NE_2	year		Set word	LW-9022 (16bit) : local year	write 2014	
6	2014/4/29	9:48:13	:noname:	ADEF	10 NE_6	year		Set word	LW-9018 (16bit) : local year	write 2014	
7	2014/4/29	9:48:16	:noname:	ADEF	10 NE_5	month		Set word	LW-9020 (16bit) : local month	write 50	
8	2014/4/29	9:50:20	:noname:	ADEF	10 NE_8	month		Set word	LW-9219 (16bit) : user no. (1~12)	write 2	
9	2014/4/29	9:50:22	:noname:	ADEF	10 NE_9	month		Set word	LW-9220 (32bit) : password	write 222	
10	2014/4/29	9:50:26	:noname:	B	10 NE_3	month		Set word	LW-9021 (16bit) : local month	write 6	

Note

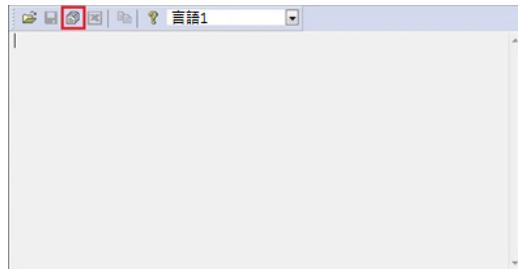
- EasyConverter でオーブンする Excel ファイルの内容が、600 万セルを越えた場合、部分の内容だけを表示させます。(xls /xlsx にエクスポートした時、依然に完全なデータ内容を表示できます)
- xls /xlsx にエクスポートする時、下記の状況では異なるシートに分けられます。

1. 1 シート内の行数が 6 万行に超えた場合。

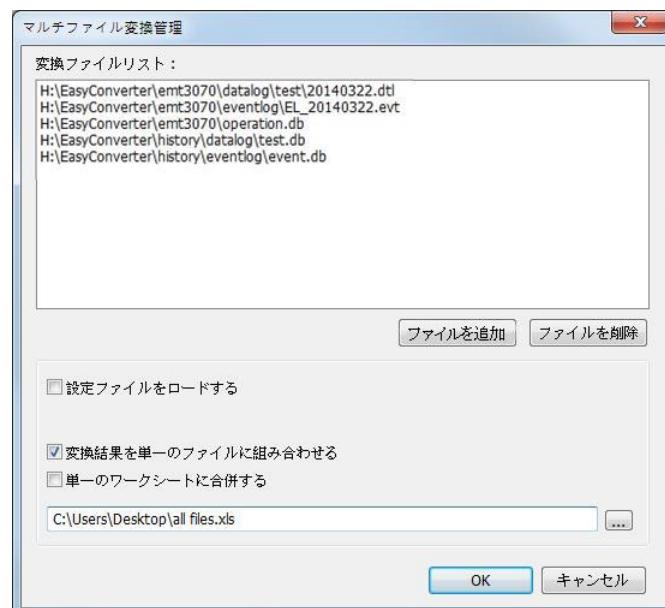
2. 1 シート内のセルが 150 万セルに超えた場合。

25.5. マルチファイル変換

1. [マルチファイル変換]のアイコンをクリックし、マルチファイル変換管理ウインドウを呼び出します。



2. [ファイルを新規追加]を選択すると、変換したいファイル名を追加できます。もし[変換結果を单一のファイルに組み合わせる]にチェックマークを入れなく、そのまま[OK]を押すと、ファイルは個別に Excel ファイルにエクスポートされます。



3. [変換結果を单一のファイルに組み合わせる]を選択すると、全てのファイルは单一の Excel ファイルにエクスポートされ、また各ファイルはそれぞれ 1 タブに分けられます。Excel ファイルは以下のとおり表示されます。

A	B	C	D	E	F	G
1	Date	Time	Millisecond	temperature	humidity	
2	2014/3/22	6:36:52	260	2	1	
3	2014/3/22	6:36:55	250	6	3	
4	2014/3/22	6:36:58	250	10	6	
5	2014/3/22	6:37:01	300	13	8	
6	2014/3/22	6:37:04	280	17	10	
7	2014/3/22	6:37:07	250	21	13	
8						
9						
10						
11						
12						
13						
14						
15						

20140322 / EL_20140322 / operation / test / event

 Note

- 合併したいファイルの合計サイズが 32 MB を超えた場合、合併を実行できません。

25.6. スケーリング機能

オープンされたファイルはデータサンプリングファイルの場合、スケーリング機能を設定できます。

スケーリング機能の使用方法は、下記に示す通りです：

新数値 = [(数値+A)xB]+C、ユーザーは A、B と C に数値を設定できます。

A->数値下限;B->[(比率最大値)-(比率例最小値)]/(数値上限)-(数値下限);C->比率最小値。

例：

ある電圧データがあり、そのフォーマットは 16-bit unsigned で、電圧数値は 0~4096 の範囲内にあり、その電圧数値をボルトに変換したい場合：(範囲は-5V~+5V の間)

新数値 = [(数値+0)xB]+(-5):



スケーリング前

スケーリング後

	スケーリング前	スケーリング後
	<pre>"Date","Time","Millisecond","sample" "2014/06/30","23:02:50","80","0" "2014/06/30","23:02:54","30","0" "2014/06/30","23:02:57","990","55" "2014/06/30","23:03:02","70","55" "2014/06/30","23:03:06","20","89" "2014/06/30","23:03:10","20","159" "2014/06/30","23:03:14","30","530" "2014/06/30","23:03:18","20","898" "2014/06/30","23:03:22","40","1024" "2014/06/30","23:03:26","0","2055" "2014/06/30","23:03:30","30","2055"</pre>	<pre>"Date","Time","Millisecond","sample" "2014/06/30","23:02:50","80","-5.000" "2014/06/30","23:02:54","30","-5.000" "2014/06/30","23:02:57","990",-4.868 "2014/06/30","23:03:02","70",-4.786 "2014/06/30","23:03:06","20",-4.786 "2014/06/30","23:03:10","20",-4.618 "2014/06/30","23:03:14","30",-3.728 "2014/06/30","23:03:18","20",-2.845 "2014/06/30","23:03:22","40",-2.542 "2014/06/30","23:03:26","0",-0.068 "2014/06/30","23:03:30","30",-0.068"</pre>

上記の資料設定を.lgs フォーマットのテンプレートファイルにセーブし、次回、必要になる時に

直接に設定をロードすることができます。テンプレートの拡張子は*.lgs です。

25.7. バッチファイルの規則

EasyConverter の command line 機能を通し、バッチファイル .bat で拡張子が.dtl 或いは .evt のソースファイルを.xls 或いは .csv ファイルに変換し、出力できます。バッチファイル内にも出力するファイルのフォーマット(例: ASCII, Unicode または UTF-8)、ミリ秒情報、設定をロードするかを定義できます。

以下はどのようにバッチファイル.bat を作成するか、及びその規則について説明します。

コマンドパラメータの説明

[/c{a,8,u}] [/t{0,1}] [/s "Format file"] ["Src file"] ["Dest file"]

例:

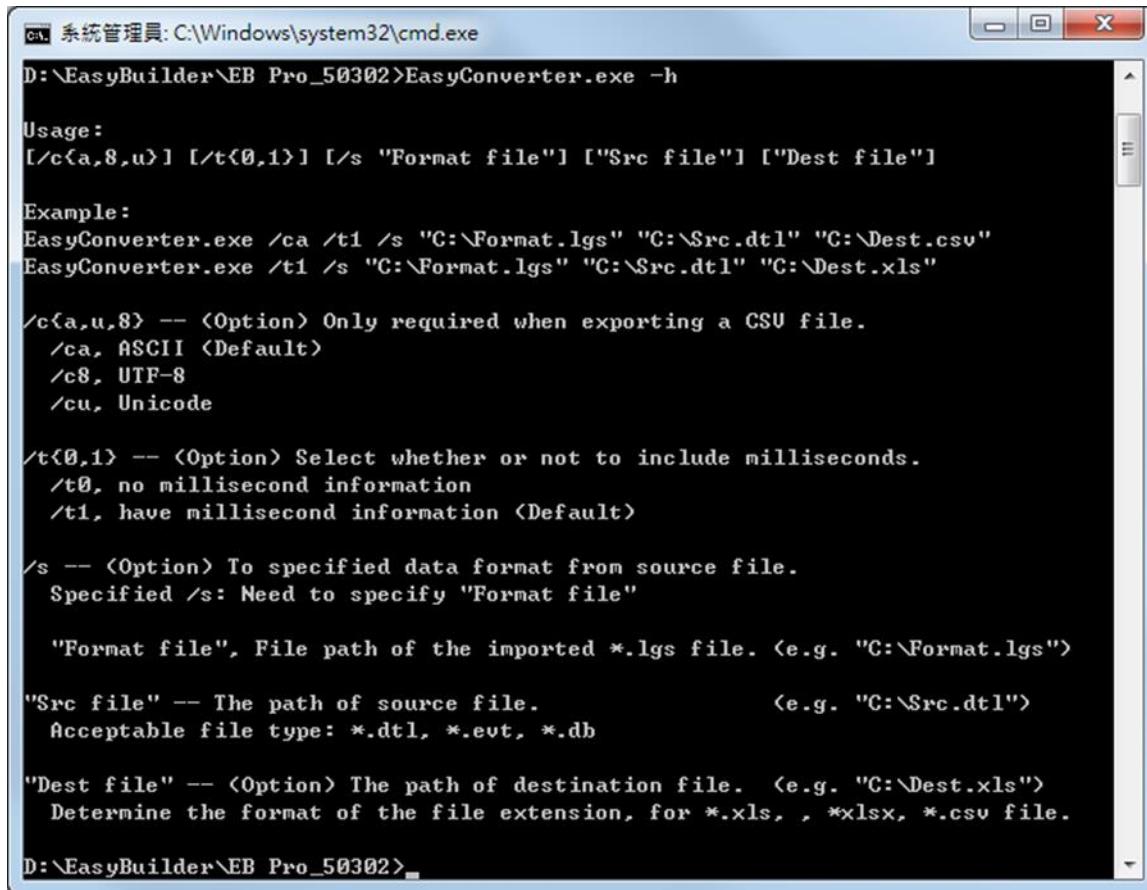
EasyConverter.exe /ca /t1 /s "C:\Format.lgs" "C:\Src.dtl" "C:\Dest.csv"

EasyConverter.exe /t1 /s "C:\Format.lgs" "C:\Src.dtl" "C:\Dest.xls"

コマンドパラメータ	説明
/c{a,8,u}	オプションです。エンコードフォーマットを設定します。.csv フォーマットにエクスポートする場合のみ必要になります。 /ca : ASCII (デフォルト) /c8 : UTF-8 /cu : Unicode
/t{0,1}	オプションです。ミリ秒情報を含むかを設定します。 /t0 : ミリ秒情報を含まない /t1 : ミリ秒情報を含む (デフォルト)
/s	オプションです。設定ファイルをロードするかを設定します。 設定ファイルをロードしたい場合、/s の後ろに.lgs のファイルパスを指定する必要があります。 例: /s "C:\Format.lgs"
"Src file"	ソースファイルのパスを指定します。ファイルフォーマットは.dtl、.evt、或いは .db でなければなりません。
"Dest file"	ファイルの出力パスを指定します。.xls または.csv のどちらでもいいです。 *注 1

注 1: command line の中に“Dest file”的ファイル名及びパスを指定していない場合、システムはファイルを“Src file”と同じディレクトリに出力します。

上記説明は Windows の cmd.exe に EasyConverter.exe のパスを入力することで確認できます。例:
D:\EasyBuilder\EB Pro>EasyConverter.exe -h



```
D:\EasyBuilder\EB_Pro_50302>EasyConverter.exe -h

Usage:
[/c{a,u,8} [/t{0,1}] [/s "Format file"] ["Src file"] ["Dest file"]

Example:
EasyConverter.exe /ca /t1 /s "C:\Format.lgs" "C:\Src.dtl" "C:\Dest.csv"
EasyConverter.exe /t1 /s "C:\Format.lgs" "C:\Src.dtl" "C:\Dest.xls"

/c{a,u,8} -- <Option> Only required when exporting a CSV file.
  /ca, ASCII <Default>
  /c8, UTF-8
  /cu, Unicode

/t{0,1} -- <Option> Select whether or not to include milliseconds.
  /t0, no millisecond information
  /t1, have millisecond information <Default>

/s -- <Option> To specified data format from source file.
  Specified /s: Need to specify "Format file"

"Format file", File path of the imported *.lgs file. <e.g. "C:\Format.lgs">

"Src file" -- The path of source file.          <e.g. "C:\Src.dtl">
  Acceptable file type: *.dtl, *.evt, *.db

"Dest file" -- <Option> The path of destination file. <e.g. "C:\Dest.xls">
  Determine the format of the file extension, for *.xls, *.xlsx, *.csv file.

D:\EasyBuilder\EB_Pro_50302>
```

説明:

D:\\$EasyBuilder\\$EB_Pro\\$HMI_memory ディレクトリの下に保存されている 20150919.dtl ファイルを 20150919.xls ファイルに変換し、それにデスクトップに保存したい場合:

バッチファイル.bat が EasyConverter と同じのディレクトリの下に保存された場合、command line は : EasyConverter.exe "D: \\$EasyBuilder\\$EB_Pro\\$HMI_memory\\$20150919.dtl"

"C: \\$Users\\$Desktop\\$20150919.xls" になります。

バッチファイル.bat が EasyConverter と異なるディレクトリの下に保存された場合、

EasyConverter.exe の保存パスを指定する必要があります。そうなると、command line は :

"D: \\$EasyBuilder\\$EB_Pro\\$EasyConverter.exe" "D: \\$EasyBuilder\\$EB_Pro\\$HMI_memory\\$20150919.dtl"

"C: \\$Users\\$Desktop\\$20150919.xls" になります。

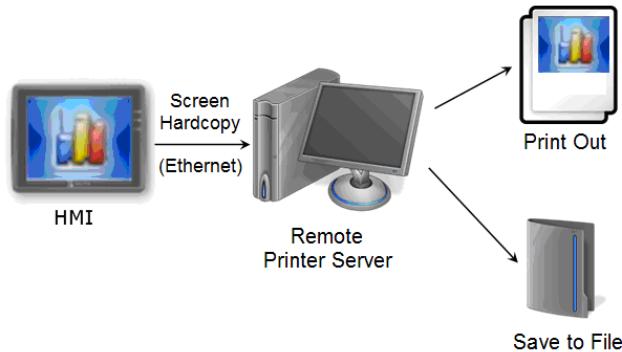
26. EasyPrinter

本章では、EasyPrinter を設定する方法について説明します。

26.1. 概要	26-2
26.2. EasyPrinter をプリンターPrinter サーバーにする	26-3
26.3. EasyPrinter をバックアップサーバーにする	26-7
26.4. EasyPrinter の操作説明	26-11
26.5. 変換バッチファイル	26-16

26.1. 概要

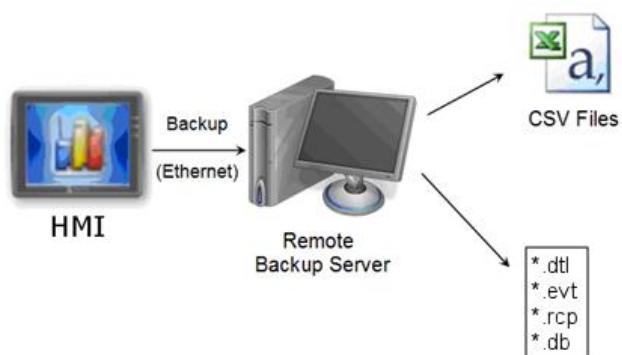
EasyPrinter は Win 32 のアプリケーションソフトで、MS Windows 7 SP1 / 8 / 10 などのシステムでのみ操作できます。本機能は HMI がイーサネットを通じ、スクリーンショットを出力し、リモート HMI でプリントすることを可能にします。



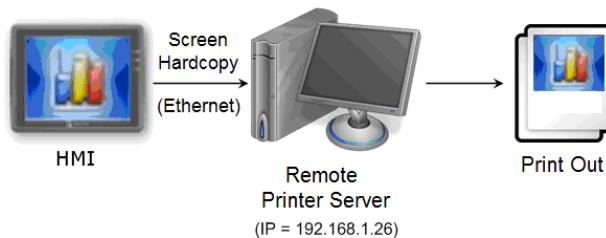
以下は EasyPrinter を使用する利点です：

- EasyPrinter は二種類のスクリーンプリントアウト出力モード：[出力する]及び[保存する]を提供しています。ユーザーは一種を使用するか、或いは二種とも使用してもいいです。
- EasyPrinter は MS Windows システムで操作するので、市場での大部分のプリンターをサポートしています。
- 本機能を使用する場合、複数の HMI が一台のプリンターを共用でき、ユーザーは各 HMI にそれぞれ一台のプリンターを用意する必要がありません。

また、EasyPrinter もバックアップサーバーとして視されます。ユーザーは HMI でのバックアップオブジェクトを使用し、イーサネットを通じ、サンプリングデータやイベントログなどの履歴データをリモート PC にバックアップすることができます。以下の説明をご覧ください：



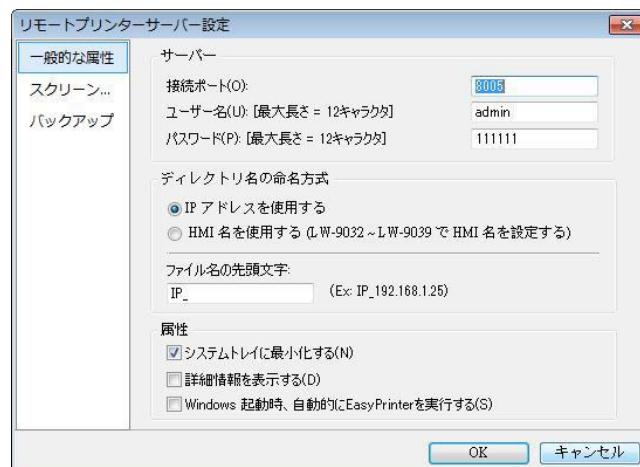
26.2. EasyPrinter をプリンターPrinter サーバーにする



ユーザーは[ファンクションキー]オブジェクトでスクリーンのハードコピーを取ることができます。これらのハードコピーはイーサネットを通し、リモート HMI サーバーに送信され、プリントアウトされます。

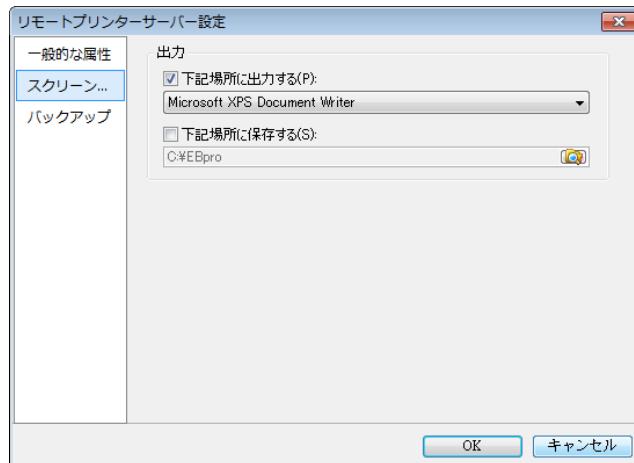
26.2.1. EasyPrinter 設定手順

EasyPrinter 設定タブで[オプション] » [設定]を選択すれば、下記のダイアログボックスが現れます：



- 左側の[一般的な属性]をクリックします。
- [サーバー]で[接続ポート]を“8005”に設定し、[ユーザー名]を“admin”、[パスワード]を“111111”に設定します。(以上は全部デフォルトであります)。
- [ディレクトリ名の命名方式]で、[IP アドレスを使用する]をクリックし、そして[ファイル名の先頭文字]に“IP_”を記入します。
- [属性]で[システムトレイに最小化する]を選択します。需要に応じ、[Windows 起動時、自動的にEasyPrinterを実行する]を選択できます。

続いて、エクスポートの行き先を設定します。

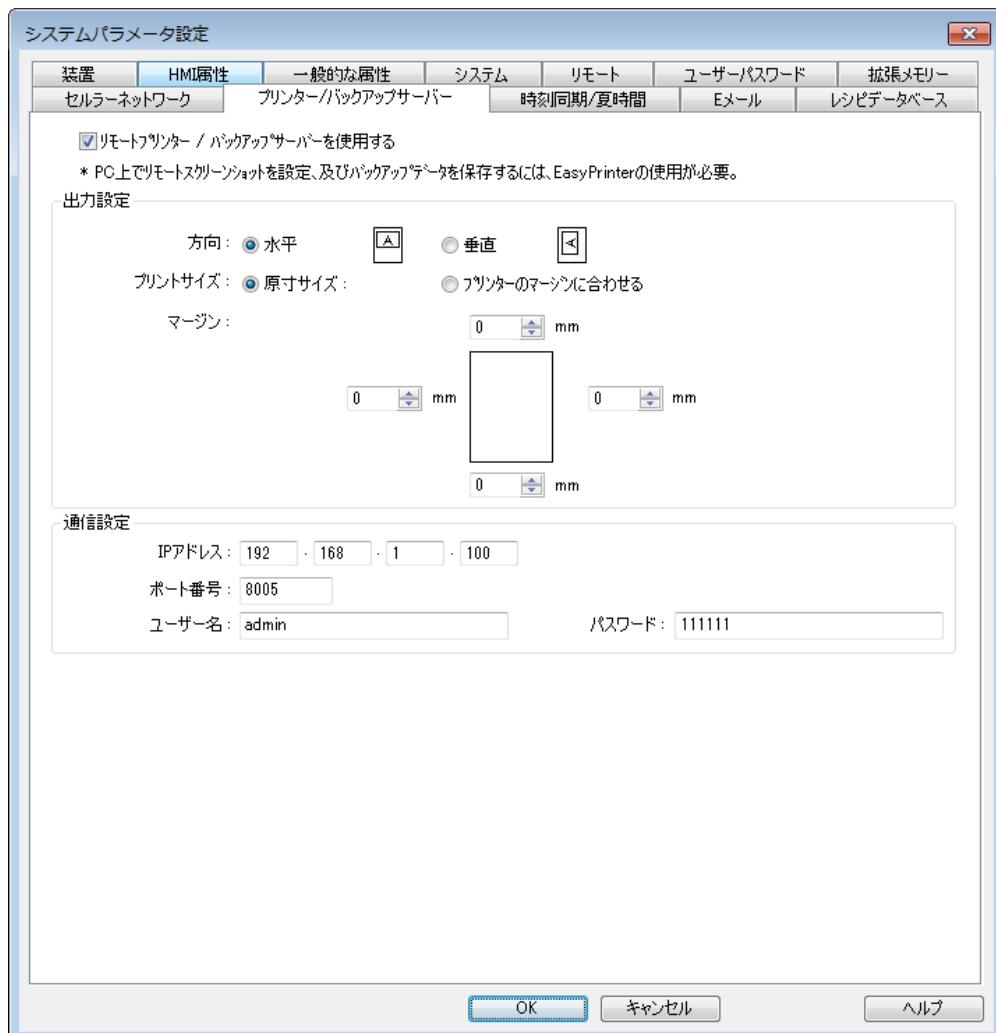


1. 左側の[スクリーンのハードコピーを取る]を選択します。
2. [出力]で[プリントアウト]を選択し、そして一台のプリンターを選択し、出力装置とします。
(注意：ユーザーは自らのシステムに実在しているプリンターのみ選択でき、上記はあくまでも参考です。)
3. [下記場所に保存する]にチェックマークを入れ、ファイルの保存先を選択します。
EasyPrinter はハードコピーファイルを指定したディレクトリに保存します。
4. [OK]を押し、以上の設定を使用するのを確定します。
5. EasyPrinter 設定タブで[ファイル] » [出力を有効にする]を選択し、EasyPrinter はこれらのプリントコマンドを出力します。

26.2.2. EasyBuilder Pro の設定手順

EasyBuilder Pro で EasyPrinter を設定する手順：

1. EasyBuilder Pro をオープンし、そして新規/既存プロジェクトをオープンします。
2. [ホーム] » [システムパラメータ設定] » [プリンター/バックアップサーバー]で、[リモートプリンター/バックアップサーバーを使用する]を選択します。



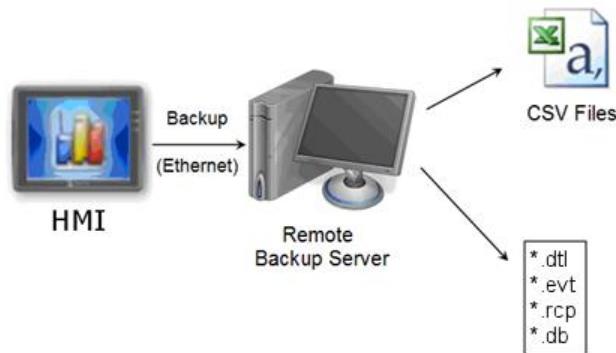
3. [出力設定]で適したマージンを設定します。(本例では、上下左右のマージンは全部 15mm に設定されました)。
4. [通信設定]でプリンターサーバーの[IP アドレス]を入力し、EasyPrinter の設定と同じです。[ポート番号]を“8005”に指定し、[ユーザー名]を“admin”、[パスワード]を“111111”に設定します。
5. [OK]を押します。
6. 続いて、メインメニューの[オブジェクト] » [ボタン]で **F** [ファンクションキー]を選択し、そしてオブジェクト設定タブで[スクリーンのハードコピーを取る]を選択し、[プリンター]を[リモートプリンター/バックアップサーバー]に設定します。

USBディスク、SDカードまたはプリンタースクリーンのハードコピーを取る
 スクリーンのハードコピーを取る 出力: リモートプリンター/バックアップサーバー
 イメージを90度回転する
* カスタマイズファイル名を使用したい場合、[PLCコントロール]オプションを選択してください。
7. [ファンクションキー]オブジェクトを[共有ウインドウ](四番ウインドウ)に置くと、ユーザーはすぐに共有ウインドウをプリントアウトできます。
8. プロジェクトを HMI に[コンパイル]及び[ダウンロード]し、設定した[ファンクションキー]オブジェクトを押し、プリントアウトします。

 Note

- ユーザーも[PLC コントロール]オブジェクトを用いてスクリーンのプリントアウトを実行できます。
- アラーム資料は EasyPrinter によってプリントアウトされません。
- EasyPrinter はイーサネットを通じる場合のみ HMI と通信できるので、使用する HMI のネットワークは正確に設定されたかを確認してください。

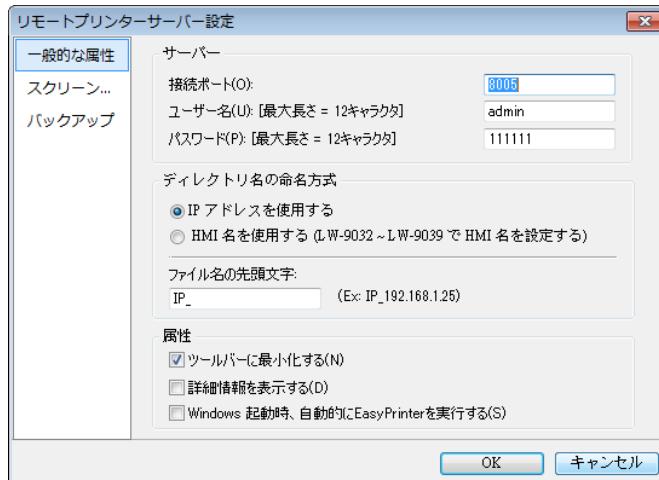
26.3. EasyPrinter をバックアップサーバーにする



ユーザーは[バックアップ]オブジェクトを利用し、履歴データ、操作ログ等をリモートバックアップサーバーにバックアップできます。

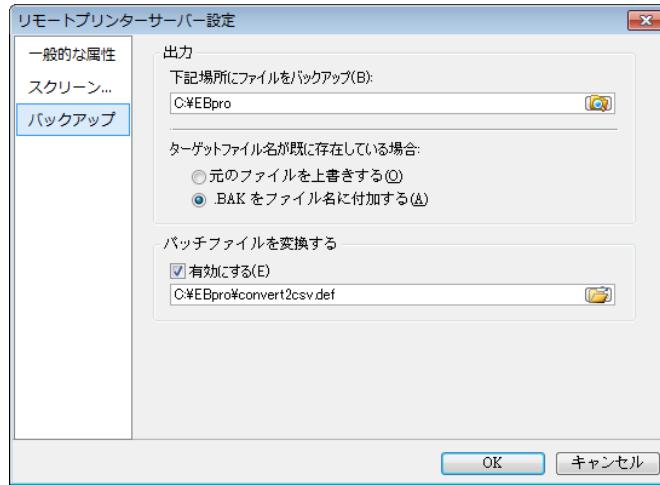
26.3.1. EasyPrinter の設定手順

EasyPrinter の設定メニューで[オプション] » [設定]を選択すると、下記のダイアログボックスが現れます：



- 左側の[一般的な属性]を選択します。
- [サーバー]で[接続ポート]を“8005”に設定し、[ユーザー名]を“admin”、[パスワード]を“111111”に設定します。(以上は全部デフォルトであります)。
- [ディレクトリ名の命名方式]で、[IP アドレスを使用]をクリックし、そして[ファイル名の先頭文字]に“IP_”を記入します。
- [属性]で[システムトレイに最小化]を選択します。需要に応じ、[Windows 起動時、自動的にEasyPrinterを実行する]を選択できます。

続いて、バックアップの位置を設定します。

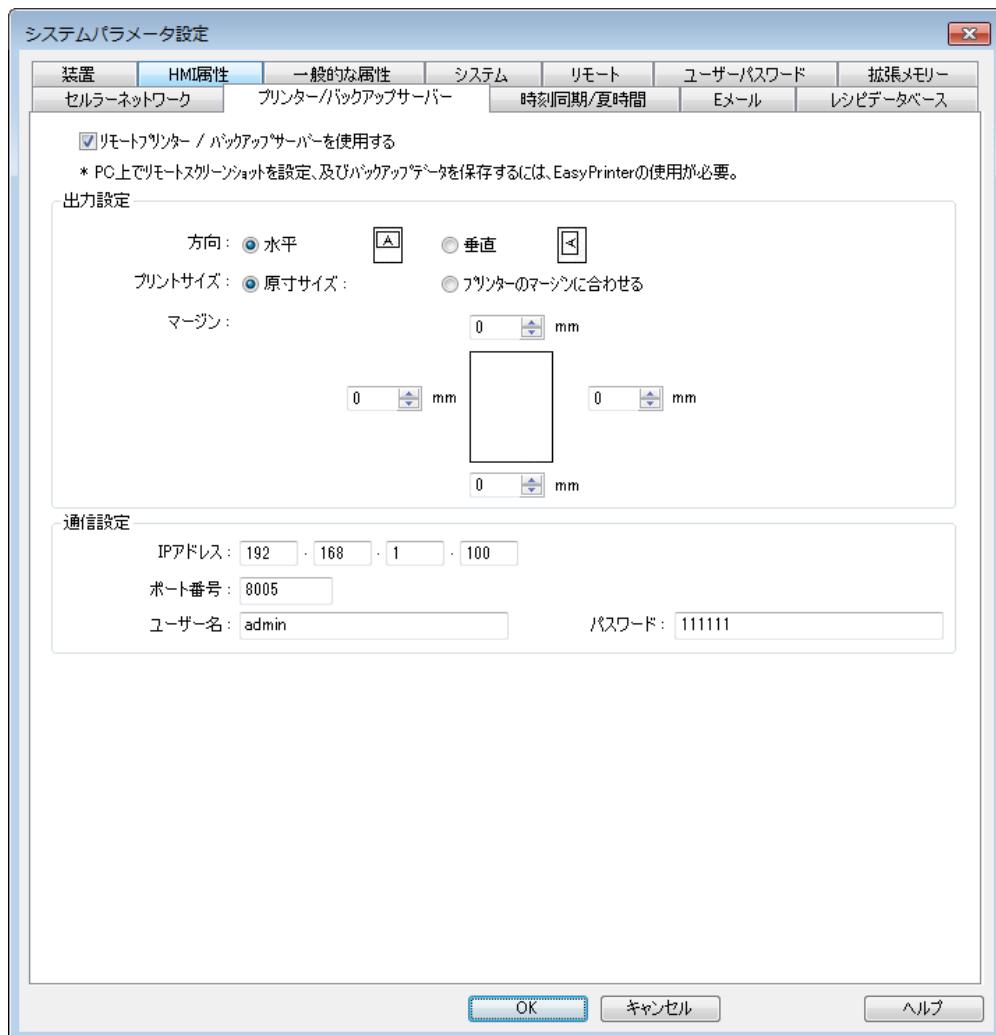


1. 左側の[バックアップ]をクリックします。
2. [出力]で^④をクリックし、履歴データの保存先を参照、選択します。
3. [OK]を押し、上記の設定を使用するのを確認します。
4. EasyPrinter の設定タブで[ファイル] » [出力を有効にする]を選択し、EasyPrinter はバックアップデータを先ほど選んだ保存先に保存します。

26.3.2. EasyBuilder Pro の設定手順

続いて、EasyBuilder Pro プロジェクトに関連設定を入れます：

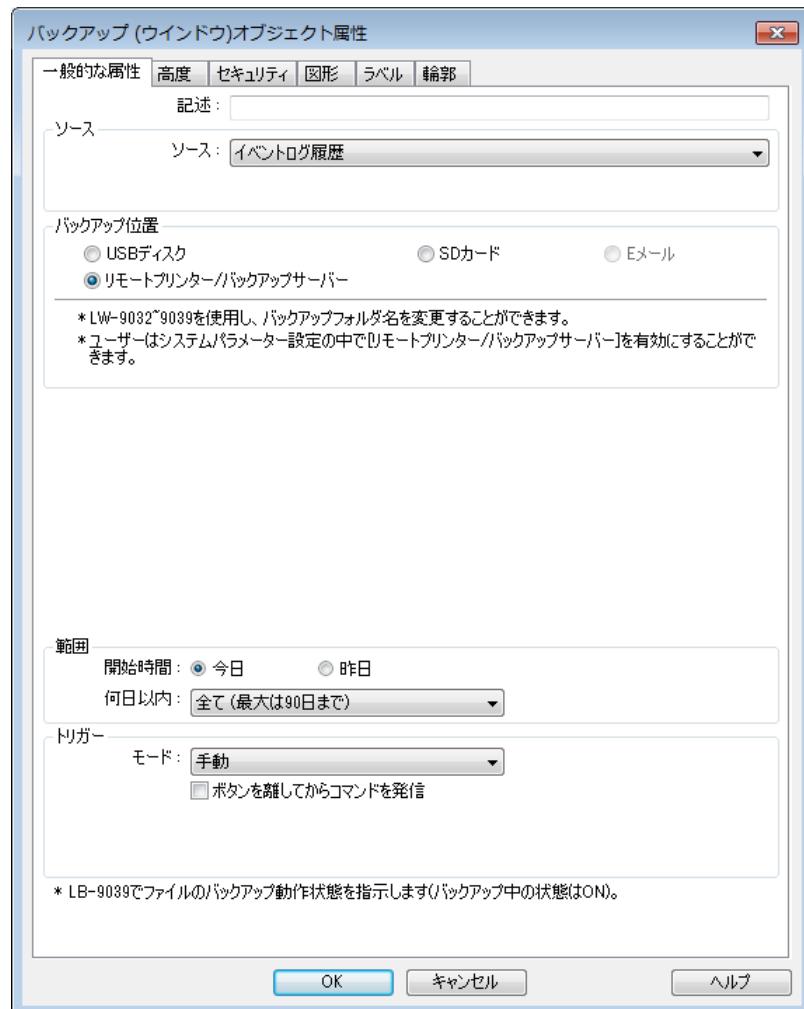
1. EasyBuilder Pro を起動し、新規或いは既存プロジェクトをオーブンします。
2. [ホーム] » [システムパラメータ設定] » [プリンター/バックアップサーバー]で、[リモートプリンター/バックアップサーバーを使用する]を選択します。



3. [通信設定]で、プリントサーバーの[IP アドレス]を EasyPrinter と同じように設定します。[ポート番号]を“8005”に設定し、[ユーザー名]を“admin”、[パスワード]を“111111”に設定します。(以上は全部デフォルトであります)。
4. [OK]を押します。

続いてはバックアップ機能をウインドウに加えます。

1. それから、メインメニューの[データ/履歴]で[バックアップ]を選択したら、下記のダイアログボックスが現れます：



2. [ソース]で[イベントログ履歴](または需要に応じ、[RW]か、[RW_A]を選択)を選択します。
3. [バックアップ位置]で、[リモートプリント/バックアップサーバー]を選択します。
4. [範囲]で[今日]と[全て]を選択します(または実際の需要に応じて変更します)。
5. [トリガー]で[手動]を選択します。
6. [OK]を押します
7. [バックアップ]オブジェクトをウインドウに入れ、例えば[4:共用ウインドウ]、そうしたら隨時にバックアップを実行できます。
8. HMI にプロジェクトを[コンパイル]及び[ダウンロード]し、先で設定した[バックアップ]オブジェクトを押し、履歴データをバックアップし始めます。

Note

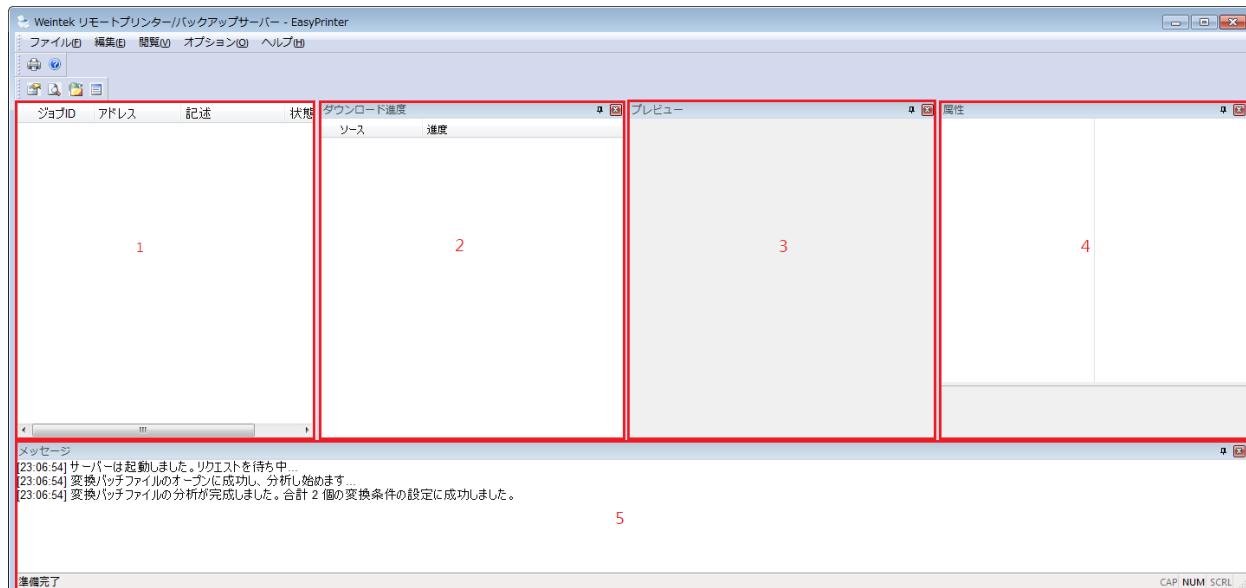
- [バックアップ]オブジェクトはビットアドレスでトリガーできます。
- ユーザーは一個の[スケジューラ]オブジェクトを置き、週の最終日に ON にすることで、[バックアップ]オブジェクトが自動的に全ての履歴データをバックアップするのをトリガーすることができます。

26.4. EasyPrinter の操作説明

以下は EasyPrinter ウィンドウのインターフェース及び操作方法を説明します。

26.4.1. ウィンドウインターフェース

EasyPrinter ウィンドウは下記に示すとおり、五つのエリアに分けられます。



エリア	名前	記述
1	ジョブリスト	本ウィンドウは全てのタスクを表示します。スクリーンハードコピーとバックアップを含みます。
2	ダウンロード進度 ウインドウ	本ウィンドウには新規タスクのダウンロード進度を表示します。
3	プレビューウィンドウ	本ウィンドウにはジョブリストで選択したスクリーンガードコピーのプレビューを表示します。
4	属性ウインドウ	本ウィンドウにはジョブリストで選択した仕タスクの情報を表示します。
5	ロガーウィンドウ	本ウィンドウにはタスク実行中の時間及びログを表示します。例えばパスワードエラーなどです。

26.4.2. 操作ガイド

以下は他の EasyPrinter 機能を説明します。

メニュー	記述
ファイル	<p>出力を有効にする 選択する : EasyPrinter で仕事を実行します。 選択しない : EasyPrinter は仕事をメモリーに保存します。</p>
編集	<p>編集 スクリーンハードコピーを編集します。ユーザーはここで自由的にプリントアウトの方向、大きさ、マージンを設定できます。</p> <p>削除 選んだ仕事を削除します。</p> <p>全てを選択 [ジョブリスト]内の全ての仕事を選択します。</p>
閲覧	<p>属性ウインドウ 属性ウインドウを表示/隠します。</p> <p>プレビューウインドウ プレビューウインドウを表示/隠します。</p> <p>ダウンロード進度ウインドウ ユーザーはダウンロード進度の表示方式を選択できます。ウインドウの中で、[進度]をクリックし、百分比或いはデータ長さで表示するのを選択します。</p>  <p>メッセージウインドウ EasyPrinter はメッセージウインドウ内のメッセージを 10,000 レコード保留できます。新しいメッセージが生じると、一番古いメッセージは削除されます。</p>
オプション	次は各設定とその意味について詳細に説明します。



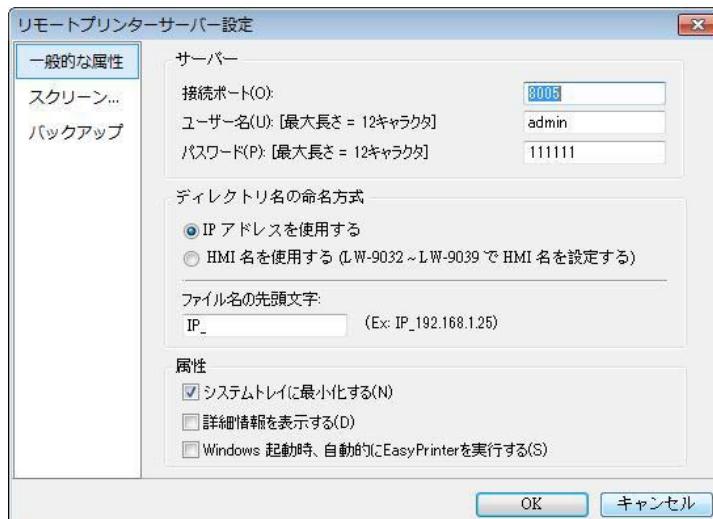
Note

- EasyPrinter はメモリーに最大 128 MB までのタスク資料しか保存できません。メモリーに空き領域がない場合、その後に生じた新規タスクを保存しません。ユーザーは[出力を有効にする]を選択し、直接に実行するか、或いは部分のタスクを削除し、メモリーの空き領域を出せるようにします。

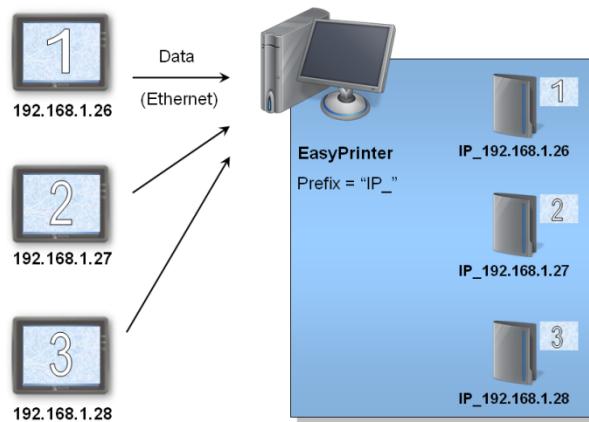
- バックアップタスクは編集できません。
- タスクを選択し終えた場合のみ、[編集]できます。
- 一個のタスクを選択してから[削除]できます。

以下は[オプション] » [設定]の各設定とその意味を詳細に説明します。

● 一般的な属性タブ



設定	記述
サーバー	<p>接続ポート Ethernet ポートを設定し、HMI に接続します。範囲は 1~65535 で、8005 はデフォルト値です。</p> <p>ユーザー名/パスワード ユーザー名及びパスワードを設定し、授権された HMI のみ EasyPrinter を使用できるようにします。</p>
ディレクトリ名の命名方式	<p>EasyPrinter は異なるフォルダで、異なる HMI からの各種ファイルを保存します(ハードコピーBMP 画像、バックアップファイル等)。これらのフォルダを命名する方式は 2 種類あります：</p> <p>IP アドレスを使用する HMI の IP アドレスを設定完了後、EasyPrinter はファイル名+IP アドレスでフォルダを命名します。詳細は下図をご参照ください。</p>



HMI名を使用する

EasyPrinter はコマンドを送った HMI の名前でフォルダに名づけます。即ち : [ファイル名]+[HMI名]になります。

属性

システムトレイに最小化する

本項にチェックマークを入れると、EasyPrinter は最小化されてシステムトレイに置かれます。ユーザーはシステムトレイ上のアイコンをダブルクリックすれば、EasyPrinter がオーブンされます。

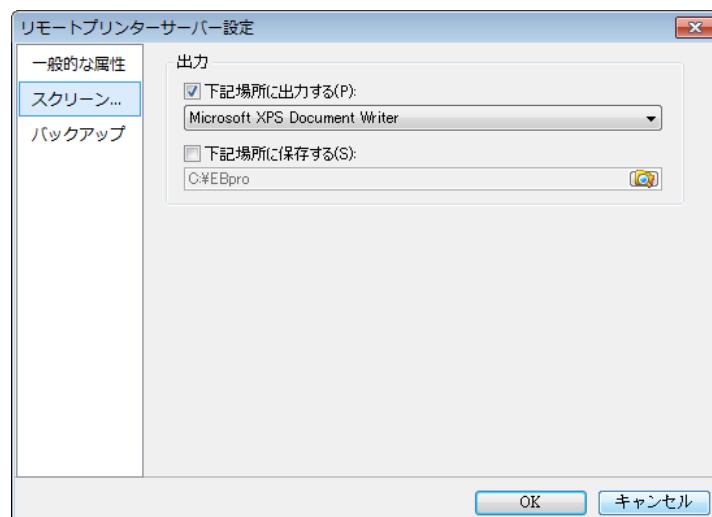
詳細情報を表示する

本項にチェックマークを入れると、メッセージウインドウにはもっと詳細な情報が表示されます。

Windows起動時、自動的に EasyPrinter を実行する

本項にチェックマークを入れると、Windows が起動された時、システムは自動的に EasyPrinter を実行します。

● スクリーンのハードコピーを取るタブ



設定	記述
出力	<p>下記場所に出力する</p> <p>本項を選択し、EasyPrinter にハードコピーを特定のプリンターでプリントアウトさせます。</p>

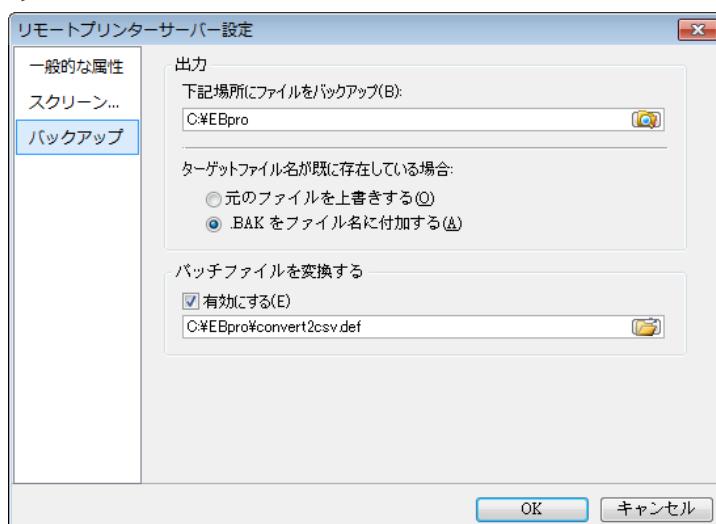
下記場所に保存する

本項目を選択し、EasyPrinter にハードコピーを BMP 画像ファイルに変換し、それに指定のパスに保存します。ユーザーは下記のパスで BMP 画像ファイルを見つけられます：

[指定したパス]¥[HMI フォルダ]¥yymmdd_hhmm.bmp

例として、あるハードコピーコマンドが 17:35:00 2009 年 1 月 12 日で発生した時、BMP 画像ファイルは "090112_1735.bmp" に命名されます。もし同一時間で、もう一個の BMP 画像ファイルが生じた場合、新しい画像ファイルは "090112_1735_01.bmp" に命名され、このように類推します。

● バックアップタブ



設定

記述

出力

EasyPrinter はバックアップファイルを特定のパスに保存します。

上層ディレクトリ : [ユーザー指定パス]¥[HMI 名] 或いは [IP アドレス]

下層ディレクトリ :

- イベントログファイル : \eventlog\EL_yyyymmdd.evt
- データサンプリングファイル : ¥datalog¥[データサンプリングオブジェクトのファイル名] : ¥yyymmdd.dtl
- レシピデータファイル : \recipe\recipe.rcp 或いは recipe_a.rcp
- レシピデータベース : \recipe\recipe.db
- 操作ログファイル : \operationlog\operationlog.db

変換バッチファイル

[オープン]にチェックマークを入れ、自動的にアップロードした履歴データを .csv 或いは .xls (Excel) ファイルに変換する変換バッチファイルを指定します。詳細は次をご参照ください。

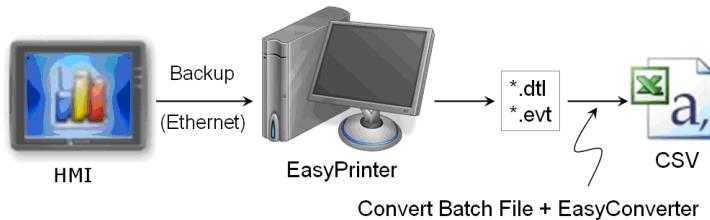
 Note

- ユーザーはレジスタ LW-9032~LW-9039 を使って HMI 名を指定することができます。

26.5. 変換バッチファイル

EasyPrinter はファイルを変換する機能を提供し、アップロードしたデータサンプリング及びイベントログなどの履歴データファイルを自動的に.csv ファイルに保存します。もしユーザーがこの機能を使用したいなら、前もって変換バッチファイルを用意し、EasyPrinter にどうやって履歴データを変換すればいいのかを指示する必要があります。

下記に示す通り、このファイルを変換する機能は実際、EasyConverter で実行され、EasyPrinter はただ変換バッチファイルの標準に従って、正確のパラメータで EasyConverter を起動し、ファイル変換の指示を完成するだけです。

 Note

- EasyConverter はもう一つの Win32 アプリケーションソフトで、履歴データファイルを.csv、或いは Excel の.xls 等のファイルを変換することができます。ユーザーは EasyBuilder Pro ダウンロードパスでこのアプリケーションソフトを見つけられます。
- ユーザーがこの機能を使用したいなら、事前に EasyPrinter 及び EasyConverter が全部同じのパスに置いているのを確定する必要があります。

26.5.1. 変換バッチファイルのデフォルト

以下はデフォルト変換バッチファイルです。

コード 1 のデフォルト変換バッチファイル(convert2csv.def)

1: "dtl", "EasyConverter /c \$(パス名)"

2: "evt", "EasyConverter /c \$(パス名)"

ファイルは 2 行の文字で表示されます。一行では 2 個パラメータを含み、カンマで分けられ、それが特定したタイプのファイル(データサンプリング及びイベントログ)を対応する処理基準になります。一番目のパラメータはそのファイルタイプの拡張子を示し、二番目のパラメータは操作モードが実行するコマンドを表示します。“\$(パス名)”はキーワードで、EasyPrinter に変換したバックアップファイル名で入れ替えることを通知します。例えば、データサンプリングファイル名は 20090112.dtl で、アップロード及び保存された場合、EasyPrinter は下記コマンドをコンソールウインドウに送信します。

1: EasyConverter /c 20090112.dtl

そうしたら、ファイル名 20090112.csv のファイルが作成されます。

よりまして、変換バッチファイルのデフォルト基準は以下のとおりです:

1. 全てのデータサンプリングファイル(.dtl)を.csv ファイルに変換します。
2. 全てのイベントログファイル(.evt)を.csv ファイルに変換します。

Note

- 実は、二番目パラメータでの“\$(パス名)”とはファイルの完全なパス名と示し、前の例では、EasyPrinter は以下の名前で入れ替えます：
[ユーザー指定パス]¥[HMI フォルダ]¥[データログ]¥[データサンプリングオブジェクトファイル名]¥20090112.dtl
- EasyPrinter は一行のファイル文字を単位にし、変換バッチファイルを解釈します。つまり、一行の文字が一個の基準になります。
- 二個のパラメータずつ、カンマで分ける必要があります。
- どのパラメータでも二重引用符で表記する必要があります。
- 同一のパラメータに、カンマを入れないこと。
- 変換バッチファイルにパラメータ\$(PathName)、\$(HmiName)と\$(IP)をサポートします。それぞれファイルパス、HMI 名と HMI IP アドレスを示しています。
- [Windows 起動時、自動的に EasyPrinter を実行する]にチェックマークを入れた場合、正確に]変換させるため、バッチファイルデフォルトの EasyConverter に PC 内のディレクトリを入れる必要があります。

1: "dtl", "C:\EBPro\EasyConverter /c \$(パス名)"

2: "evt", "C:\EBPro\EasyConverter /c \$(パス名)"

 詳細な情報は、《25 EasyConverter》をご参照ください。

26.5.2. 特定した基準

下記の特定した基準をご参照ください：

特定した HMI がアップロードしたファイルに特別な操作ができます。例えはコード 2。

- HMI 名でその HMI を判別できます。例えコード 3。
- 或いは異なるデータサンプリングファイルに異なる操作が必要場合、例えはコード 4。
([データサンプリング]ファイルで、ファイル名は"Voltage"の場合のみ適用します。)
三番目パラメータ("*)とは、任意の HMI にもかかわらず、データサンプリングの中に基準を合っているものさえあればそれを受け入れます。
ユーザーは三番目のパラメータを"192.168.1.26"、"192.168.1.*"、HMI 名等に限定でき、ターゲット HMI の範囲を局限することができます。

コード 2.が HMI IP=192.168.1.26 についての特別な定義基準

```
1: "dtl", "EasyConverter /c ${パス名}", "192.168.1.26"
```

コード 3.が HMI 名=Weintek_01 についての特別な定義基準

```
1: "dtl", "EasyConverter /c ${パス名}", "Weintek_01"
```

コード 4.がデータサンプリングオブジェクトファイル名=Voltage についての特別な定義基準

```
1: "dtl", "EasyConverter /s Voltage.lgs ${パス名}", "*", "Voltage"
```

26.5.3. 変換バッチファイルのフォーマット

以下は標準なフォーマット、及び各パラメータの説明を述べます。

ファイルタイプ	コマンド(行)	HMI IP/名	条件 1	条件 2
---------	---------	----------	------	------

- ファイルタイプ
このパラメータは本基準の対象となるアップロードファイルの拡張子を特定します。
(e.g.“.dtl”はデータサンプリングファイル、“.evt”はイベントログファイル)
- コマンド(行)
アップロードしたファイルが基準に合っている場合、EasyPrinter がコンソールウインドウ console window に発信する確定的なコマンドのことです。
- HMI IP/名
このパラメータは本基準の対象となる HMI を特定します。
- 条件 1
もしファイルタイプは“.dtl”だったら、このパラメータは本基準の対象となる[データサンプリング]オブジェクトのフォルダ名を特定します。他のファイルには適用しません。
- 条件 2
未使用(今後の使用に確保する)

26.5.4. 実行する順序

EasyPrinter は各ファイルがアップロードされた後、下から上へ各基準を検証することができます。一旦基準に合ったたら、検証が中止され、そして次のファイルを処理します。なので、ユーザーはより広範な基準を下方置き、より明確な基準を上方に置くのがお薦めです。例えば、以下はバッチファイルの内容です：

```
"evt", "EasyConverter /c $(パス名)"  
"dtl", "EasyConverter /c $(パス名)"  
"dtl", "EasyConverter /c $(パス名)", "192.168.1.26"  
"dtl", "EasyConverter /c $(パス名)", "my_HMI_01"  
"dtl", "EasyConverter /c $(パス名)", "my_HMI_02"  
"dtl", "EasyConverter /s Voltage.lgs $(パス名)", "*", "Voltage"
```

正確な順序は(最後の行から上へ実行する)：

```
"dtl", "EasyConverter /s Voltage.lgs $(パス名)", "*", "Voltage"  
"dtl", "EasyConverter /c $(パス名)", "my_HMI_02"  
"dtl", "EasyConverter /c $(パス名)", "my_HMI_01"  
"dtl", "EasyConverter /c $(パス名)", "192.168.1.26"  
"dtl", "EasyConverter /c $(パス名)"  
"evt", "EasyConverter /c $(パス名)"
```

27. EasySimulator

本章では、EasySimulator の使用方法について説明します。

27.1. 概要	27-2
27.2. EasySimulator を設定する手順.....	27-2

27.1. 概要

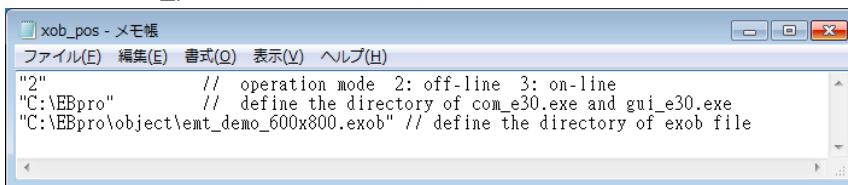
EasySimulator はユーザーに EasyBuilder Pro ソフトを丸ごとインストールする必要がなく、オンライン/オフラインシミュレーションができるようにさせる機能です。但し、事前に所要する関連ファイルを用意する必要があります。下記のステップに従って、設定してください。

27.2. EasySimulator を設定する手順

1. 関連ファイルを用意します：

- [driver]→[win32]
- com_e30.exe
- EasySimulator.exe
- gui_e30.exe
- sqlite3.dll
- xob_pos.def
- libcurl.dll
- libeay32.dll
- MFC71.dll
- mosquitto.dll
- mosquitopp.dll
- pthreadVC2.dll
- ssleay32.dll

2. 文字編集ツールで xob_pos.def をオープン(例：メモ帳)し、正確に対応内容を設定します。



行数	記述
1	"2"でオフラインシミュレーションを実行します。"3"でオンラインシミュレーションを実行します。
2	関連ファイルの保存ディレクトリです。 (例：com_e30.exe, gui_e30.exe, EasySimulator.exe...など)。
3	.exob ファイルの保存ディレクトリです。

3. EasySimulator.exe をダブルクリックすれば、シミュレーションを実行できます。

4. オンライン/オフラインシミュレーション結果はスクリーンに表示します。



Note

- 関連ファイルは全部 EasyBuilder Pro インストールディレクトリで見つけられます。だから、お先に他の PC で EasyBuilder Pro をインストールしてから、これらのファイルをターゲット PC にコピーしてください。
- EasySimulator.exe を実行した後、反応がなかったら、再度に関連ファイルのインストールディレクトリが正確であるかどうかを確認してください。

“Failed to open project file: No such file or directory”というウインドウがポップアップされたら、.exob ファイルのパスが間違っていると示しています。再度確認してください。

28. EasySystemSetting

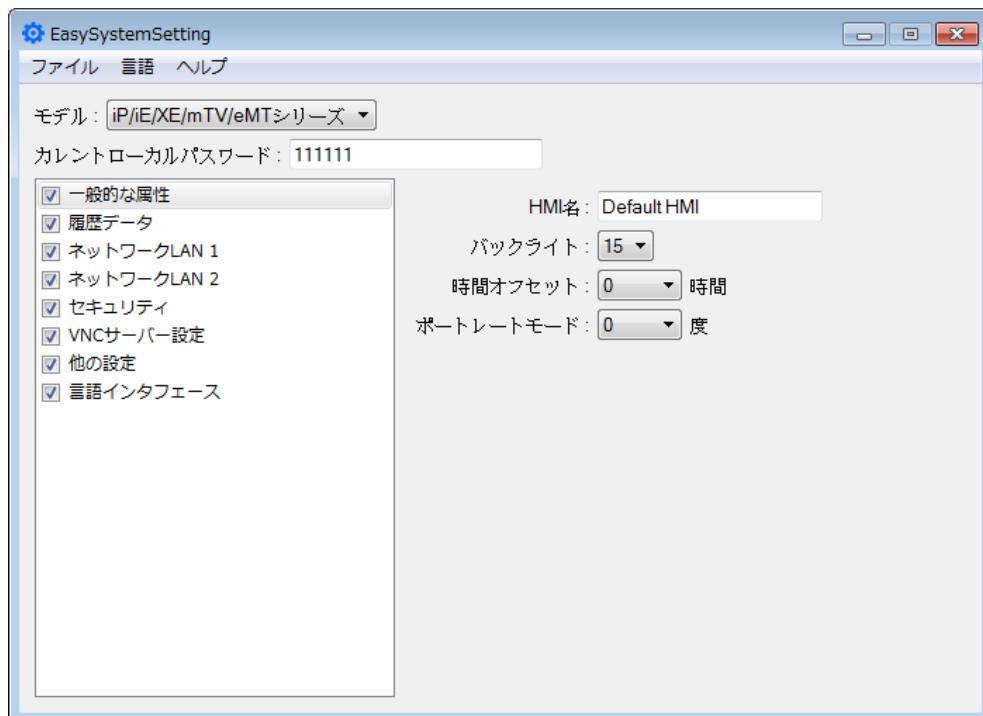
本章では、EasySystemSetting の使用方法について説明します。

28.1. EasySystemSetting	28-2
28.2. 操作手順	28-5

28.1. EasySystemSetting

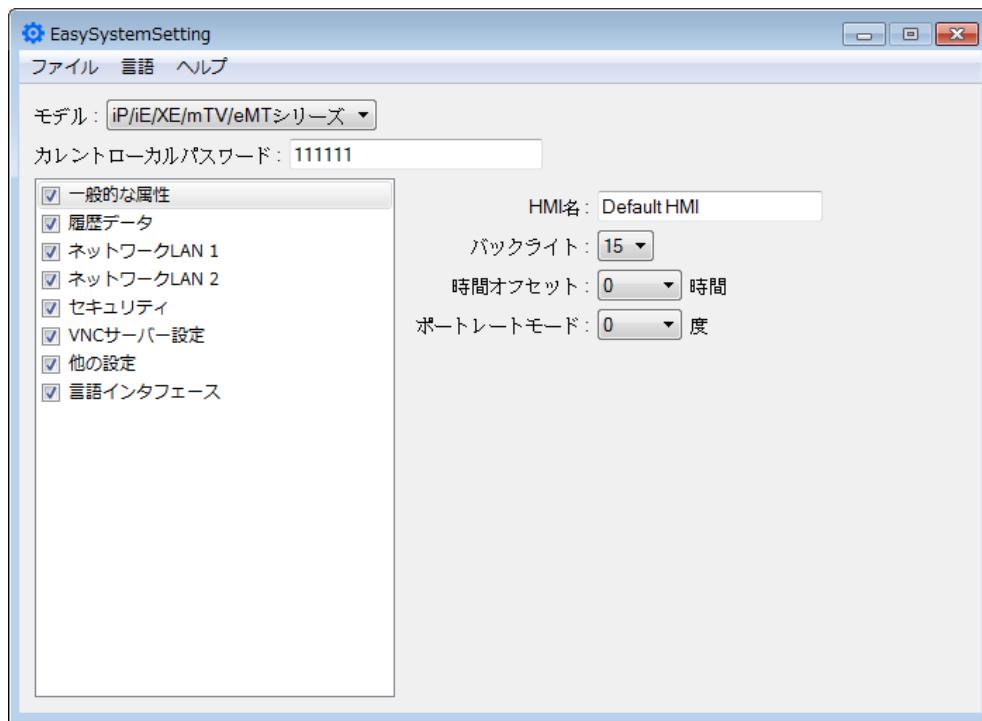


EasySystemSetting は USB ディスクまたは SD カードでハードウェア設定情報を変更させる機能を提供します。EasyBuilder Pro インストールフォルダで **EasySystemSetting.exe** をクリックしてオープンするか、EasyBuilder Pro の[ツール]タブで[システム設定エディター]をクリックしてオープンすることができます。本機能を使用するには、HMI のファームウェアが OS 20131106 以上であること。



設定	記述
ファイル	[新規作成]全ての設定をデフォルト設定に復元します。 [ロードする]保存された.conf ファイルをインポートして編集します。 [保存する]完成した設定を.conf ファイルにエクスポートします。
言語	表示言語を変更します。
ヘルプ	[アバウト]バージョン情報です。

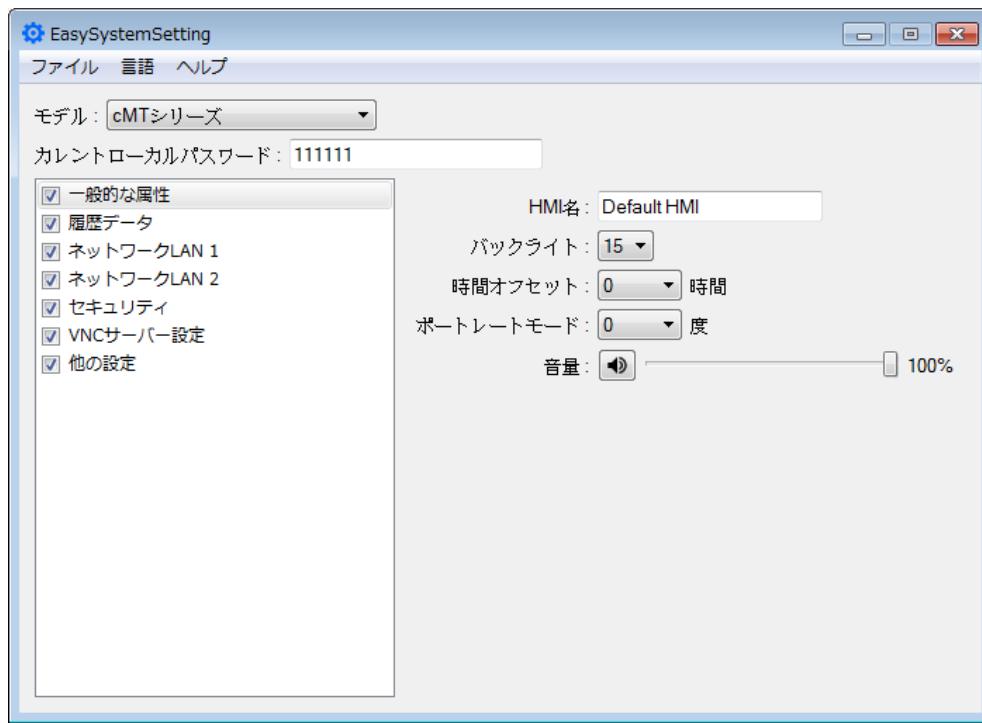
iP/iE/XE/mTV/eMT シリーズ



以上の設定は全部本マニュアル第4章に記されたシステム設定の内容と同じです。以下では、一般的な属性タブを紹介します。

設定	記述
カレントローカルパスワード	システムパラメータ設定にログインするためのパスワードを入力してください。
一般的な属性	<p>[HMI名] HMI の名前を設定します。キャラクタ : [A~Z], [a~z], [0~9], [-], [.]だけをサポートします。</p> <p>[バックライト] バックライトの明るさを調整します。</p> <p>[時間オフセット] HMI の RTC 時刻に時間オフセットを適用します。</p> <p>例: 現在 RTC 時刻が 15:00:00 で、時間オフセットを -3 に設定した場合、更新した後は 12:00:00 になります。</p> <p>[ポートレートモード] 表示モードを設定します。</p>

cMT シリーズ



以上の設定は全部本マニュアル第4章に記されたシステム設定の内容と同じです。以下では、一般的な属性タブを紹介します。

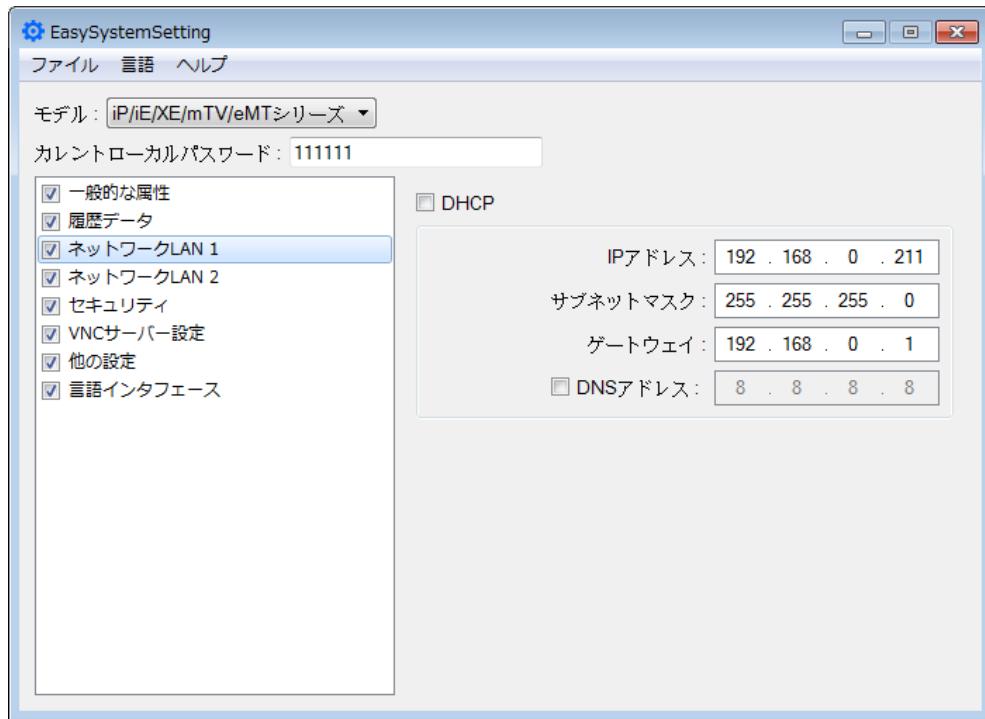
設定	記述
カレントローカルパスワード	システムパラメータ設定にログインするためのパスワードを入力してください。
一般的な属性	<p>[HMI名] HMI の名前を設定します。キャラクタ : [A~Z], [a~z], [0~9], [-], [.]だけをサポートします。</p> <p>[バックライト] バックライトの明るさを調整します。</p> <p>[時間オフセット] HMI の RTC 時刻に時間オフセットを適用します。</p> <p>例: 現在 RTC 時刻が 15:00:00 で、時刻オフセットを -3 に設定した場合、更新した後は 12:00:00 になります。</p> <p>[ポートレートモード] 表示モードを設定します。</p> <p>[音量] 音量を設定します。</p>

28.2. 操作手順

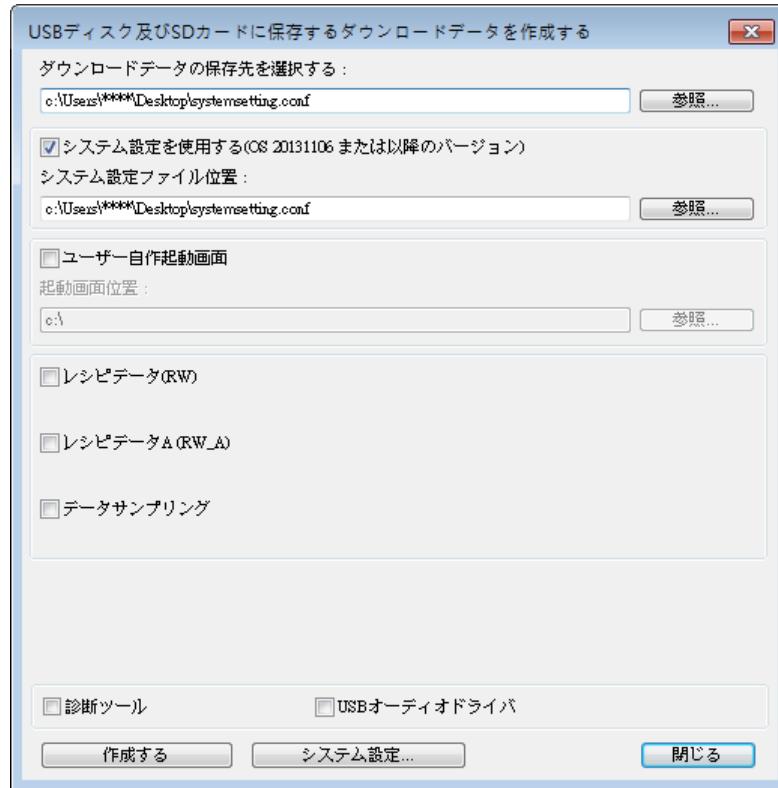
以下では、USB ディスクまたは SD カードで HMI の IP アドレスを更新する方法を説明します。

iP/iE/XE/mTV/eMT シリーズ

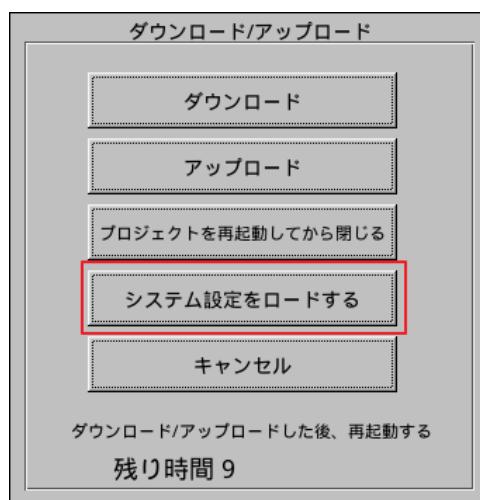
1. EasySystemSetting ダイアログボックスをオープンし、HMI の Network 情報を以下のように変更します。



2. [ファイル] » [保存する]をクリックし、一個の systemsetting.conf ファイルを生成させます。
3. EasySystemSetting を終了します。
4. [USB ディスク/SD カードに使用するダウンロードデータを作成する]をクリックし、[システム設定を使用する]にチェックマークを入れ、.conf ファイルをインポートします。



5. [USB ディスク/SD カードに使用するダウンロードデータを作成する]の[作成する]をクリックし、ダウンロードデータを生成させます。
6. データの保存先である外部記憶装置を HMI に挿入すると、ダウンロード/アップロードダイアログボックスがポップアップされます。



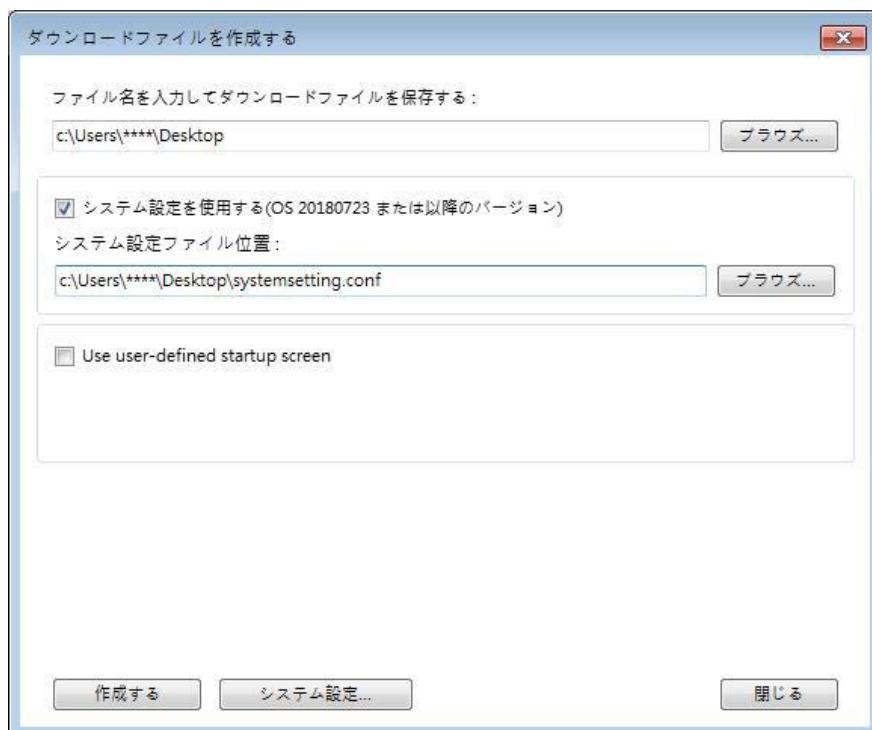
7. [Load System Setting]をクリックすると、[Download Config Settings]メッセージが表示されます。システム設定が完成した後、システムはプロジェクトファイルの更新を続けます。

cMT シリーズ

- [システム設定]エディターをクリックして EasySystemSetting ダイアログボックスをオープンし、HMI の Network 情報を下記のように変更します。



- [ファイル] » [保存]をクリックし、systemsetting.conf ファイルを生成します。
- EasySystemSetting を閉じます。
- EasyBuilder Pro で[ダウンロードファイルを作成する]をクリックし、[システム設定を使用する]にチェックマークを入れ、.conf ファイルをインポートします。



5. [作成する]を押すと、同じ名前のコンパイルファイル及びシステム設定ファイルが作成され(EBProject1.cxb & EBProject1.conf)、この2つのファイルを外部記憶装置内の同一のディレクトリに保存します。
6. その外部記憶装置をHMIに挿入すれば、ダウンロード/アップロードダイアログボックスがポップアップされ、そこに[ダウンロード]をクリックします。プロジェクトとダウンロードパスワードを正確に入力した後、ダウンロードをクリックします。



Note

- UtilityManagerのダウンロード機能またはEB Proのダウンロード機能を使用すれば、直接にシステム設定ファイルをダウンロードできます。

29. パススルー機能

本章では、パススルー機能を設定する方法について説明します。

29.1. 概要	29-2
29.2. COM ポートパススルーのイーサネットモード	29-2
29.3. COM ポートパススルーの COM ポートモード	29-7
29.4. パススルー通信のコントロール	29-10
29.5. SIEMENS S7-200 PPI と S7-300 MPI パススルー機能設定	29-10
29.6. イーサネットパススルー	29-14

29.1. 概要

パススルー機能は PC でのアプリケーションソフトが HMI を通し、直接に PLC をコントロールすることを許可します。この場合、HMI の機能はアダプタに似ています。

COM ポートパススルー機能には下記二種類のモードが含まれています：

- イーサネット
- COM ポート

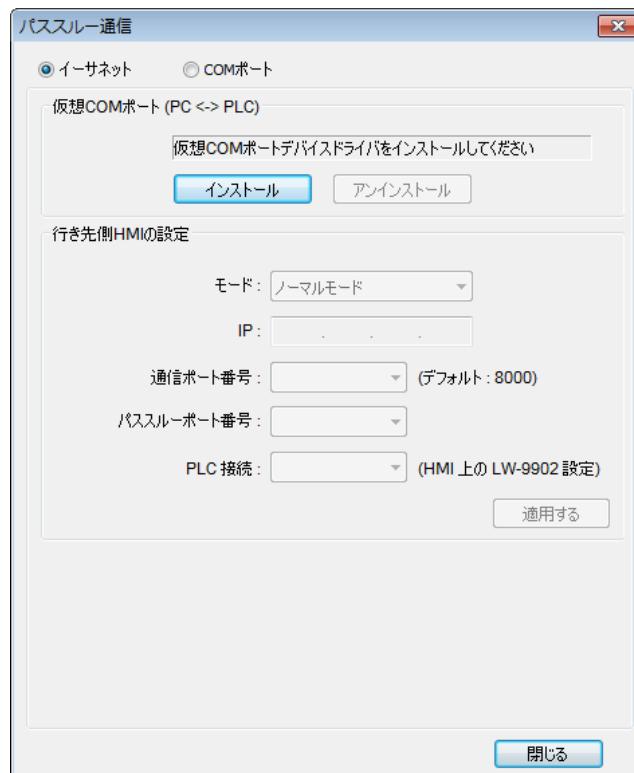
UtilityManager の[COM ポートパススルー]ボタンをクリックしたら、この二種類モードの設定内容を確認することができます。

29.2. COM ポートパススルーのイーサネットモード

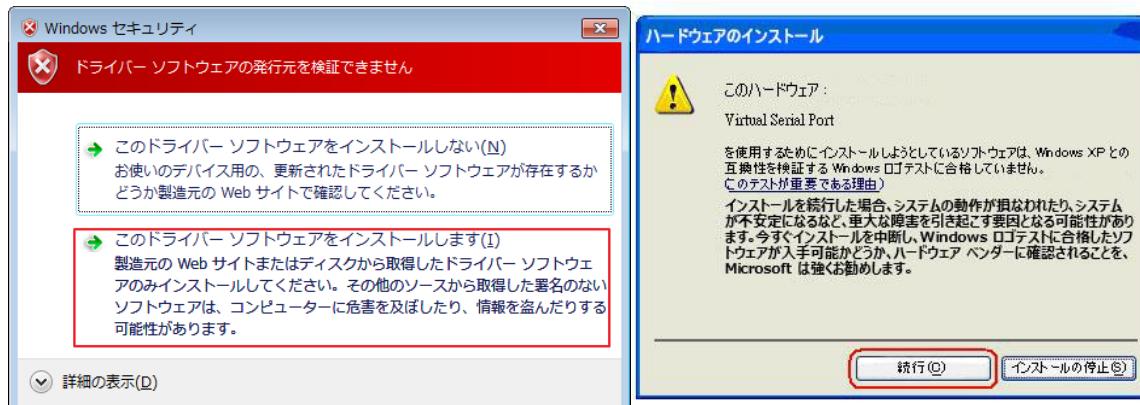
29.2.1. 仮想 COM ポートドライバをインストールする手順

[イーサネット]パススルー機能を使用する前、事前に仮想 COM ポートドライバをインストールする必要があります。

1. UtilityManager をオープンし、現時点でのドライバのインストール状態を確認します。画面では[仮想 COM ポートドライバをインストールしてください]と表示されたら、[インストール]ボタンをクリックしてください。



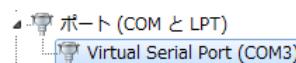
2. ドライバをインストールする過程ではインストールを確認するメッセージが出る可能性があり、[このドライバーソフトウェアをインストールします]を選択してください。



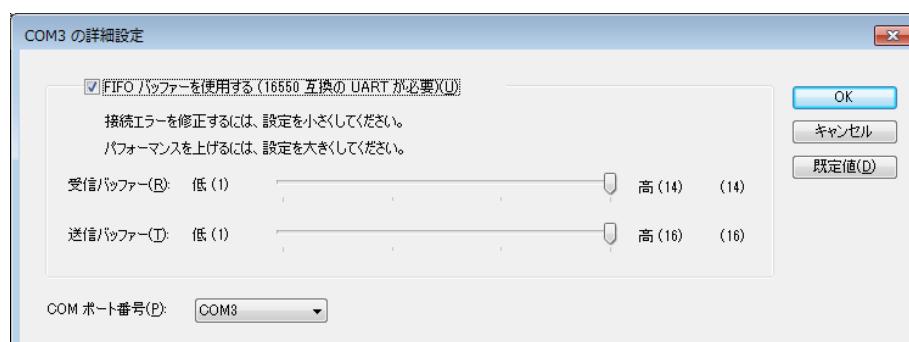
3. インストールが完成したら、もともと[仮想 COM ポートドライバをインストールしてください]が表示された位置では現在使用している仮想 COM ポートを表示します。

29.2.2. 仮想 COM ポート番号を変更する手順

1. [デバイスマネージャー]の中に、[Virtual Serial Port]のインストールが完成されたのが見られます。

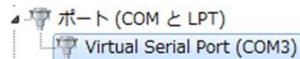


2. 仮想 COM ポート番号を変更したいなら、[Virtual Serial Port]の内容に入り、[COM ポート設定]中の[詳細]を選択したら、COM ポート番号を変更できます。

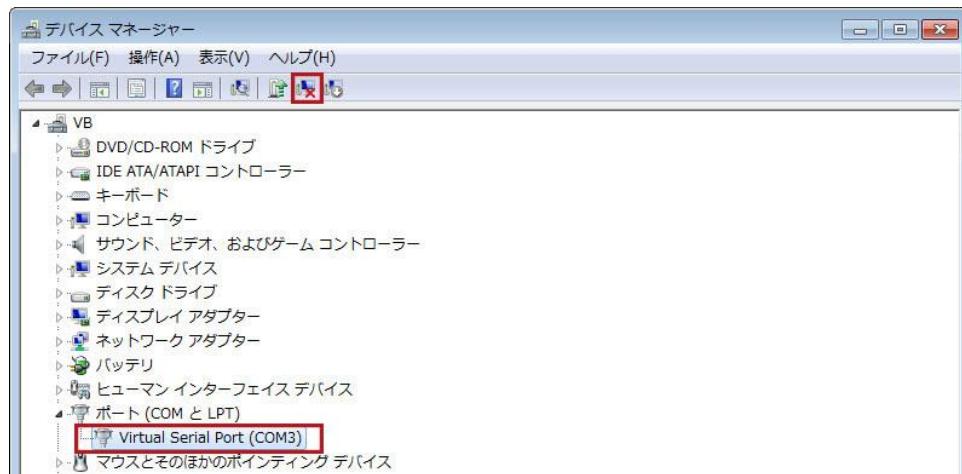


29.2.3. 仮想 COM ポートをアンインストールする手順

- [デバイスマネージャ]の中に、[Virtual Serial Port]のインストールが完成されたのが見られます。



- 仮想 COM ポートをアンインストールするには、まず、削除したい[Virtual Serial Port]を選択してから、[デバイスマネージャ]ツールバー上の[アンインストール]ボタンを選択します。



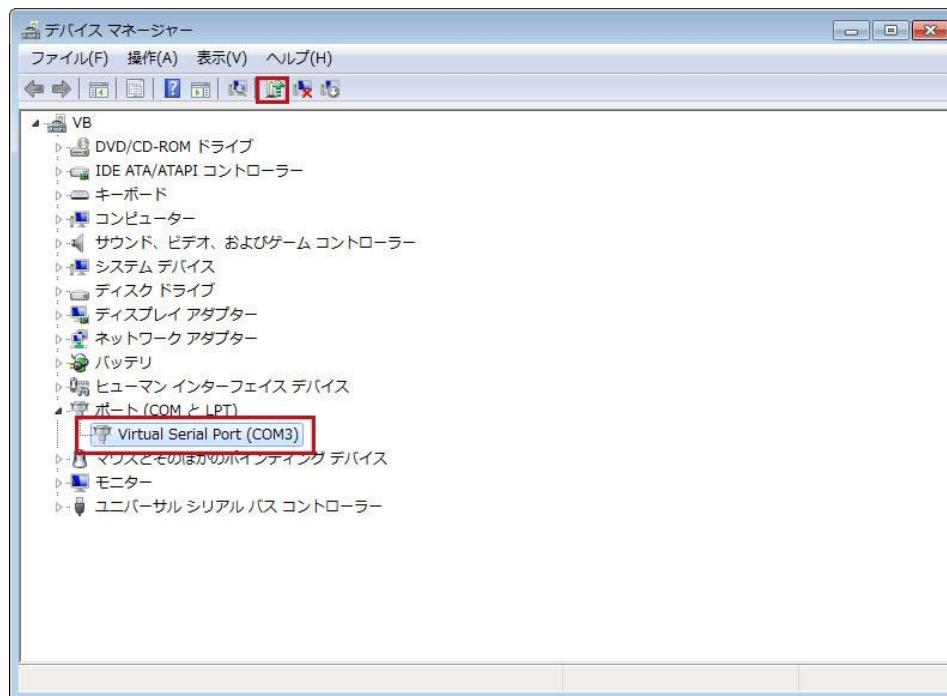
- [OK]をクリックすると、指定した仮想 COM ポートをアンインストールできます。

29.2.4. 仮想 COM ポートドライバを更新する手順

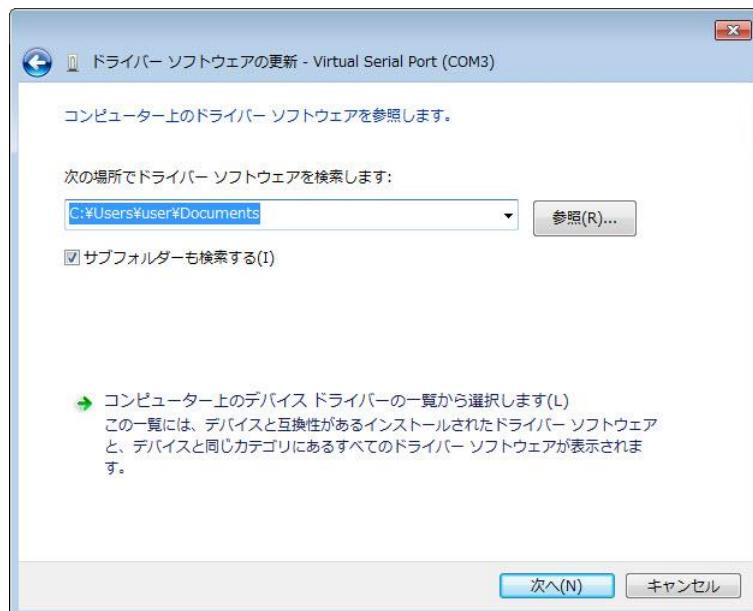
- [デバイスマネージャ]の中に、[Virtual Serial Port]のインストールが完成されたのが見られます。



- 仮想 COM ポートドライバを更新するには、まずは更新したい[Virtual Serial Port]を選択してから、[デバイスマネージャ]ウインドウの[ドライバー ソフトウェアの更新]ボタンをクリックします。



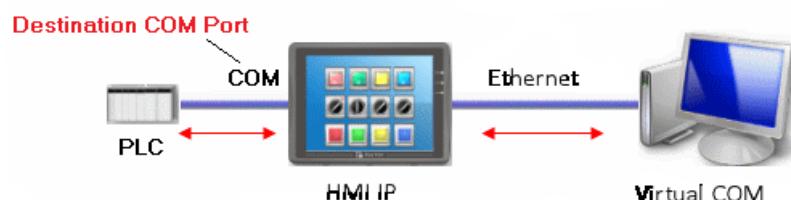
3. コンピュータを参照し、更新したいドライバを選択してから、[次へ]をクリックしてインストールを完了します。



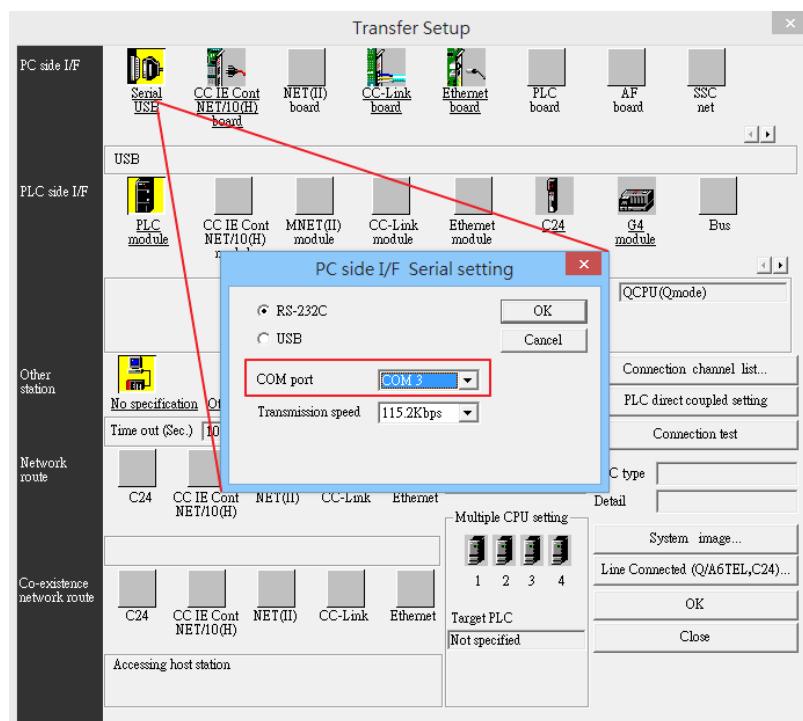
29.2.5. イーサネットモード設定

仮想 COM ポートドライバをインストールし終えた後、下記のステップに従えば、イーサネットパススルー通信機能を使用できます。

1. PLC を接続する HMI の IP アドレスを設定してください。
2. HMI の通信ポート、HMI と PLC を接続する COM ポート及びその属性を設定します。
全ての設定を完成した後、[適用する]ボタンを押してから、全ての属性が有効になります。



3. PC 上のアプリケーションソフトを使用する際に、仮想 COM ポートを使用すること。Mitsubishi のアプリを例にして、仮にこの時の仮想ポートは COM 3 だったら、[PC side I/F Serial setting] ウィンドウ内の[COM port]で COM 3 を選択する必要があります。



4. 上記の各設定を設定した後、ユーザーが PC 上の PLC アプリケーションソフトを実行する時、HMI は自動的にパススルーモードに切り替えます。この時、PC 上のアプリケーションソフトが仮想 COM ポートを通じて PLC をコントロールすることに視されます。アプリケーションソフトを閉じる時、HMI も自動的にパススルーモードを閉じます。

29.3. COM ポートパススルーの COM ポートモード



[データソース COM ポート] : HMI と PLC の接続ポートを指します。

[データ行き先 COM ポート] : HMI と PC の接続ポートを指します。

[COM ポート] パススルー機能を使用する前、正確にこの 2 つの COM ポートの属性を設定する必要があります。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

29.3.1. COM ポート設定

[COM ポート] パススルー機能を使用する方法は二種類あります。

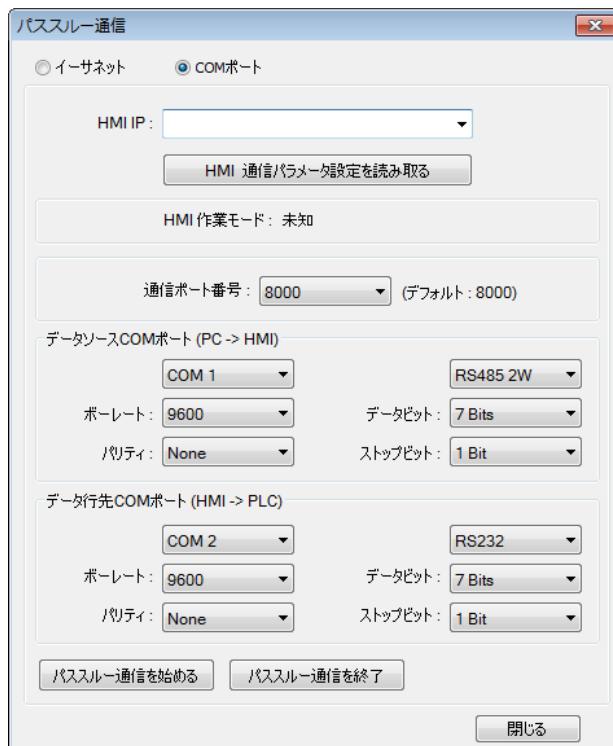
- Utility Manager を使用する
- システムレジスタを使用する

LW-9901 でデータソース COM ポートを設定する(1~3:COM1~COM3)

LW-9902 でデータ行き先 COM ポートを設定する(1~3:COM1~COM3)

29.3.2. Utility Manager を使用する

1. Utility Manager で[COM ポートパススルー]ボタンを押し、通信パラメータを設定します。下図に示す通りです。



設定	記述
HMI IP	HMI の IP アドレスを指定する必要があります。
HMI 通信パラメータ設定を読み取る	HMI のデータソース COM ポートとデータ行き先 COM ポートの各設定値を読み取ります。[HMI 通信パラメータ設定を読み取る]ボタンを押した後、全ての通信パラメータが更新されます。
データソース COM ポート(PC->HMI)/データ行き先 COM ポート(HMI->PLC)	データソース COM ポートとデータ行き先 COM ポートの通信パラメータを表示、設定します。[パススルー通信を始める]をクリックすれば、[データソース COM ポート]、[データ行き先 COM ポート]で設定した内容に基づき、パススルー通信機能を実行します。
ポートレート/データビット/パリティ/ストップビット	普通、[データソース COM ポート]と[データ行き先 COM ポート]の中で、これらの設定を同じにする必要があります。[データソース COM ポート]は PC に接続するから、一般的に、通信モードに “RS-232” を選択すればいいです。[データ行き先 COM ポート]は PLC に接続するので、通信モードは PLC の通信設定に従う必要があり、“RS-232”、“RS-4852W” 或いは “RS-4854W” が選べます。

 Note

- パススルー通信機能が不要の場合、[パススルー通信を終了する]を選択しパススルー通信機能を閉じてから、HMI が改めて PLC との通信を再開します。

三種類のモードで現在 HMI の稼動モードを表示します。

モード	記述
未知	HMI の設定値を読み取る前、表示される HMI の稼動状態です。
ノーマルモード	HMI の設定値を読み取った後、表示される状態です。HMI はノーマルな通信状態にあり、データソース COM ポートからの何のデータも受け入れません。
パススルーモード	HMI はパススルーモードにあります。この時、PC でのアプリケーションソフトはデータソース COM ポートを通じて、直接にデータ行き先 COM ポートに接続している PLC をコントロールできます。

29.3.3. システムレジスタを使用する

もう一種の HMI のパススルー機能を起動する方式は直接にシステムレジスタ LW-9901(データソース COM ポート)及び LW-9902(データ行き先 COM ポート)内のデータ内容を修正することです。LW-9901 と LW-9902 内のデータが下記条件に合致する時、HMI は自動的にパススルー機能を起動します。

- LW-9901 と LW-9902 内の数値は 1~3(1~3 はそれぞれ COM 1~COM 3 を示す)であること。
- LW-9901 と LW-9902 内の数値は同じでないこと。

各 COM ポートのパラメータを変更する必要があれば、各パラメータが対応するシステムレジスタ内の数値を変更し、それに [LB-9030 : COM 1 の通信パラメータを更新する]、[LB-9031 : COM 2 の通信パラメータを更新する]、[LB-9032 : COM 3 の通信パラメータを更新する] を ON に設定すればいいです。

Note

- HMI のパススルー通信機能を閉じたかったら、LW-9901 と LW-9902 内の数値を 0 に変更すればいいです。
- ご使用のモデルの COM1 の第 7、8 ピンがちょうど COM3 の RX、TX と共にされている場合(例えば eMT シリーズ、cMT シリーズ)、[装置リスト] に COM3 装置が見つからない状況では、第 7、8 ピンは COM1 の RTS、CTS として使用されます。COM3 の RX、TX を有効にしたい場合、[システムパラメータ設定] » [装置リスト] に COM3 を使用する Master-Slave ドライバまたは Free Protocol ドライバを追加する必要があります。

29.4. パススルーモードのコントロール

一般的には、パススルーモードを起動する時、パススルーが終わるまで、HMI は PLC との通信を閉じつつあります。しかしながら、特定的な PLC ドライバはパススルーモードの状態下でも、HMI-PLC, PC-HMI の間での同時通信をサポートします。

 上記のような HMI-PLC での通信をサポートする PLC ドライバを調べたかったら、《PLC 接続ガイド》の関連章節をご参照ください。

同時通信機能はシステムレジスタ LW-9903 でコントロールされます。

LW-9903 数値	記述
0(デフォルト)	ノーマルモードです。パススルーモードを実行する時、HMI-PLC, PC-HMI の間では同時通信できます。
2	パススルーモードを実行する時、HMI と PLC の間の通信が中止されます。

Note

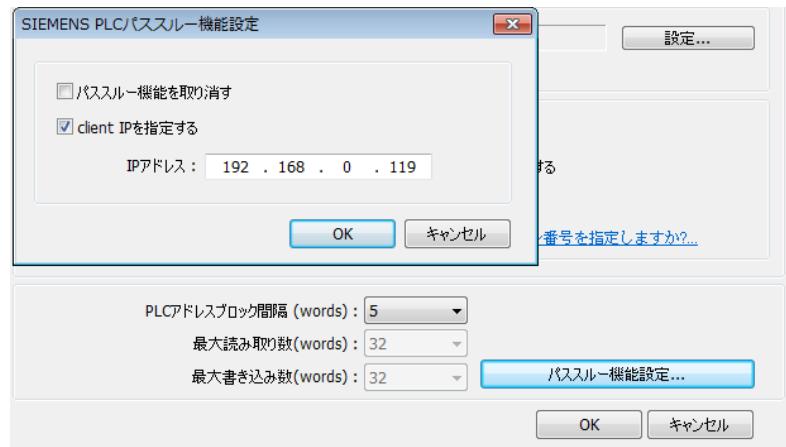
- COM ポートの転送速度が限られていたら、LW-9903 を 2 に設定し本機能を閉じることで、PLC プログラムをアップロード/ダウンロードする速度を向上できます。

29.5. SIEMENS S7-200 PPI と S7-300 MPI パススルーモード機能設定

EasyBuilder Pro は SIEMENS S7-200 PPI と S7-300 MPI パススルーモード機能をサポートします。

29.5.1. EasyBuilder Pro 設定

EasyBuilder Pro[システムパラメータ設定] » [装置リスト]で Siemens S7-200 PPI 或いは S7-300 MPI を新規追加する時、[パススルーモード機能設定]をクリックすれば、下記のダイアログボックスが現れます。

**設定****記述**

パススルー機能を取り消す

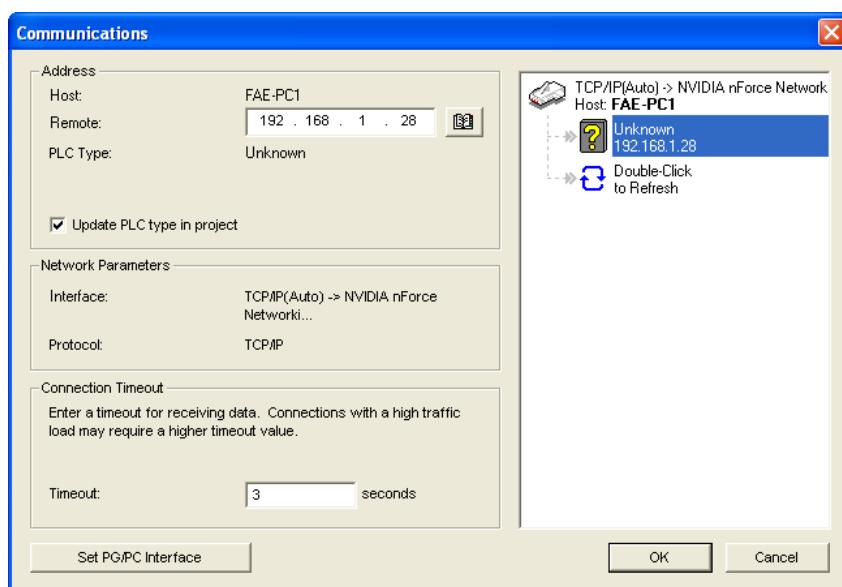
本項目にチェックマークを入れたら、パススルー機能が閉じられ、チェックマークが入っていないようになります。

Client IP を指定する

パススルー機能をオープンする時、HMI に接続するクライアント側の IP アドレスを指定します。

29.5.2. S7-200 PPI の接続方式

パススルー機能を実行している HMI を起動し、それにインターネットに接続しているのを確認取れた後、STEP 7 Micro/Win をオープンし、Communications 設定ウインドウに入り、パススルー機能を実行したい HMI の IP アドレスを探し出し、そして接続したら、パススルーが始められます。

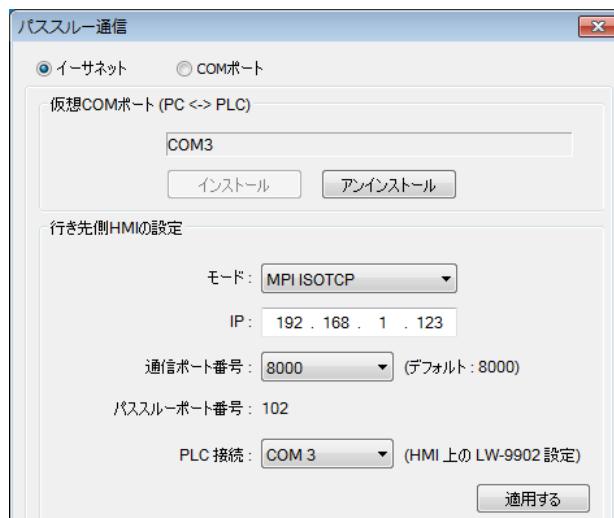


29.5.3. S7-300 MPI の接続方式

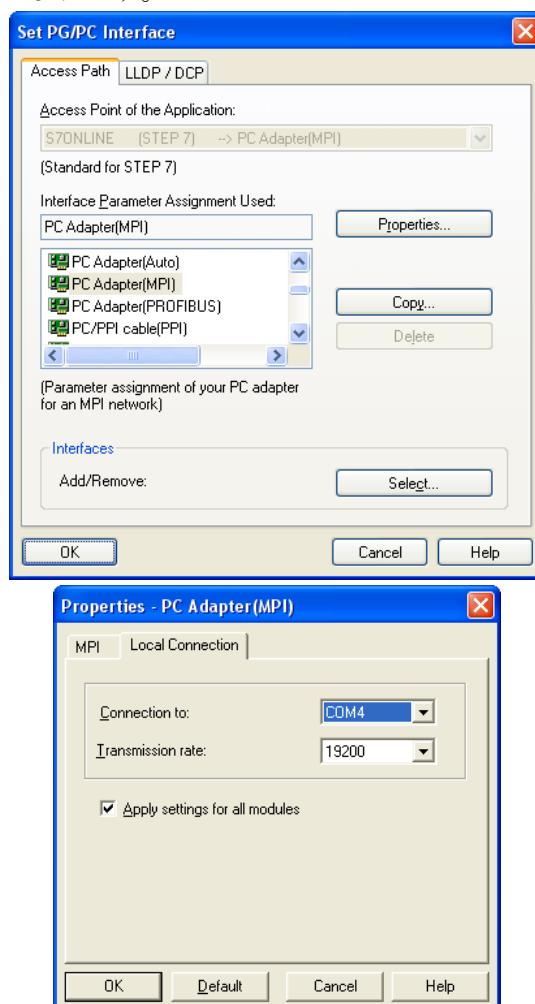
仮想 COM ポート、またはイーサネットを通じて接続します。

29.5.3.1. 仮想 COM ポートを通じる

1. Utility Manager で[パススルー通信]をオープンし、MPI ISOTCP を選択し、MPI 専用の仮想 COM ポートプログラムをインストールします。同時に HMI IP 及び HMI が PLC を接続するポートを設定し、パススルーを起動します。



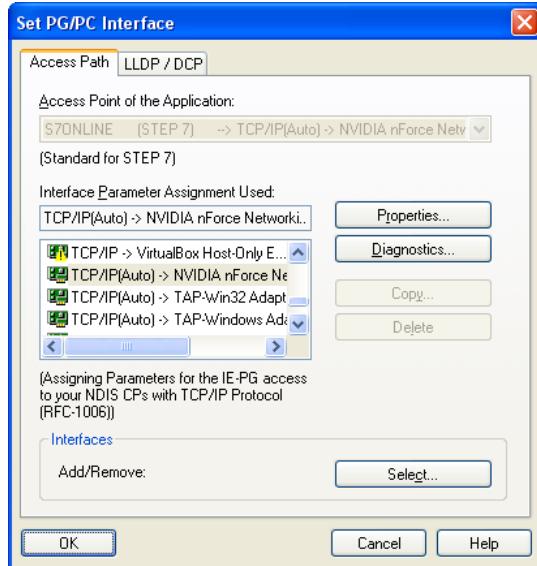
2. STEP 7 に[Option] » [SetPG/PCIInterface]を選択し、インターフェースカードは PC Adapter(MPI)であるのを確認し、[Properties]に入り、仮想 COM ポートと同じの接続ポートを設定します。下記の例では COM 4 になります。



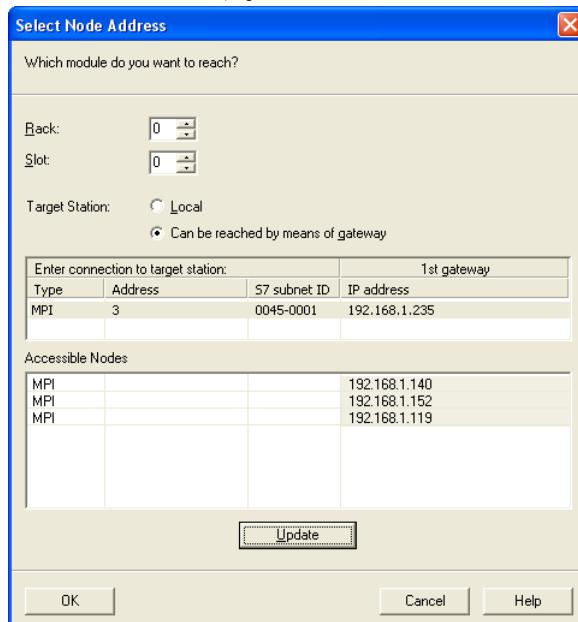
3. 上記の設定を完成したら、STEP 7 で HMI を通し、PLC プログラムをアップロード/ダウンロードできます。

29.5.3.2. イーサネットを通じる

1. STEP 7 で、[Option] » [Set PG/PC Interface]に入り、下図に示す通り、“TCP/IP(Auto)->使用中のネットワークインターフェースカード名”を選択します。



2. [PLC] » [Update station to PG] の Target Station で [Can be reached by means of gateway] を選択し、そして下方の欄に、左から右へ MPI を選択し、PLC ステーション番号、S7 Subnet ID、HMI IP アドレスを記入します。OK を押した後、S7 はパススルー機能により、HMI を通し PLC プログラムを STEP 7 にアップロードできます。



29.5.4. SIEMENS パススルーに関するレジスタ

システムレジスタ LW-10850 から LW-10864 は SIEMENS パススルー状態を設定/表示することに用いられます。

 各レジスタの機能に関し、《22 システムレジスタ》をご参照ください。

パススルーを実行する時、エラーが起こったら、LW-10863 はエラーを提示します。LW-10864 のほうはエラーコードを表示し、下表ではエラーコードが表示する内容及び可能な原因を説明します。(下記での“クライアント側”は通常、STEP 7 PLC プログラムのことを指します)

エラーコード	記述	発生原因
0	正常に実行している。	
1	クライアント側が HMI に接続するのを禁じる。	HMI はパススルー機能を実行しているので、他のクライアント側からのコマンドを受け入れません。
2	クライアント側が HMI に接続するのを禁じる。	LW-10850 が 1 に設定された時、HMI に接続したいクライアント側の IP アドレスは LW-10858~LW-10861 が定義したものと違っています。
3	不正確な通信プロトコル。	LW-10853 の設定が不正確でした。
4	不正確な PLC ステーション番号。	LW-10852 が指定したステーション番号に合致する PLC は存在しません。
5	通信遅延。	PLC は正確に接続していません。
6	通信ビジー。	PLC はパススルーコマンドを受け入れません。PLC の設定を確認してください。
7	誤ったパススルーコマンド。	パススルー機能の環境構築に失敗しました。

 このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

29.6. イーサネットパススルー

イーサネットパススルー通信機能は PLC と PC ともイーサネットで HMI に接続する場合、PC 上のアプリケーションソフトが HMI を通して直接に PLC 制御させる機能です。cMT シリーズ HMI にのみサポートされます。



- Utility Manager をオープンした後、cMT Series を選択し、[分析・テストツール]タブを選択すれば、[イーサネットパススルー]が見られます。



- [イーサネットパススルー]を選択し、正確な通信パラメータを入力して接続を作成します。



設定

記述

HMI IP	HMI の IP アドレスを指定します。
HMI ポート番号	HMI に接続するポート番号を入力します。デフォルトは 8000 です。
パススルーIP アドレス	パススルーで制御する装置の IP アドレスを指定します。
状態	設定完了後、[接続する]を押したら、ここで接続状態を確認できます。



Note

- 一台の HMI には 1 セットだけのイーサネットパススルーIP を設定できます。
- システムレジスタ LB-9044(リモートコントロール禁止)が有効にした場合、パススルーを実行できません。

30. プロジェクト保護機能

本章では、プロジェクト保護機能の関連設定について説明します。

30.1. 概要	30-2
30.2. EXOB パスワード	30-2
30.3. デコンパイル禁止	30-3
30.4. EXOB ファイルをアップロード禁止機能.....	30-3
30.5. 識別キー	30-3
30.6. EMTP パスワード	30-4

30.1. 概要

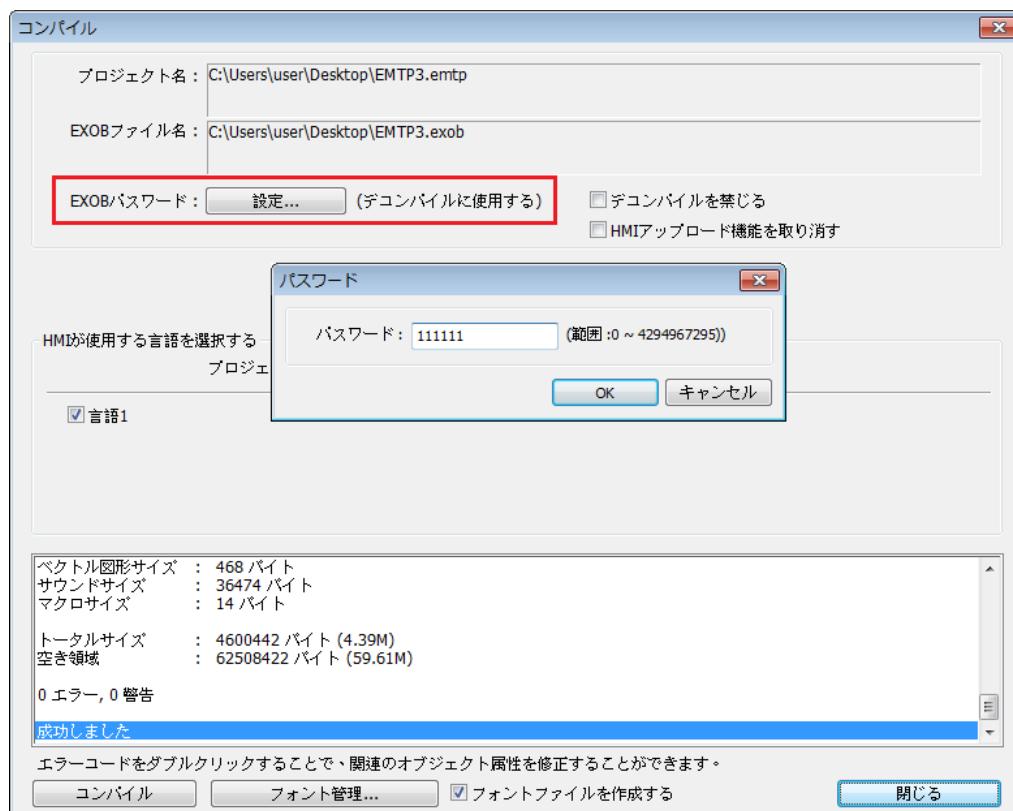
EasyBuilder Pro は数多くのプロジェクト安全暗号化機能を提供し、ファイルの安全性を向上します。

Note

- 下記で紹介した安全暗号化機能に関し、パスワードはユーザーに暗号化されたので、万が一パスワードが忘れた場合、たとえメーカーでも解読できないので、必ずパスワードを大切に保管してください。

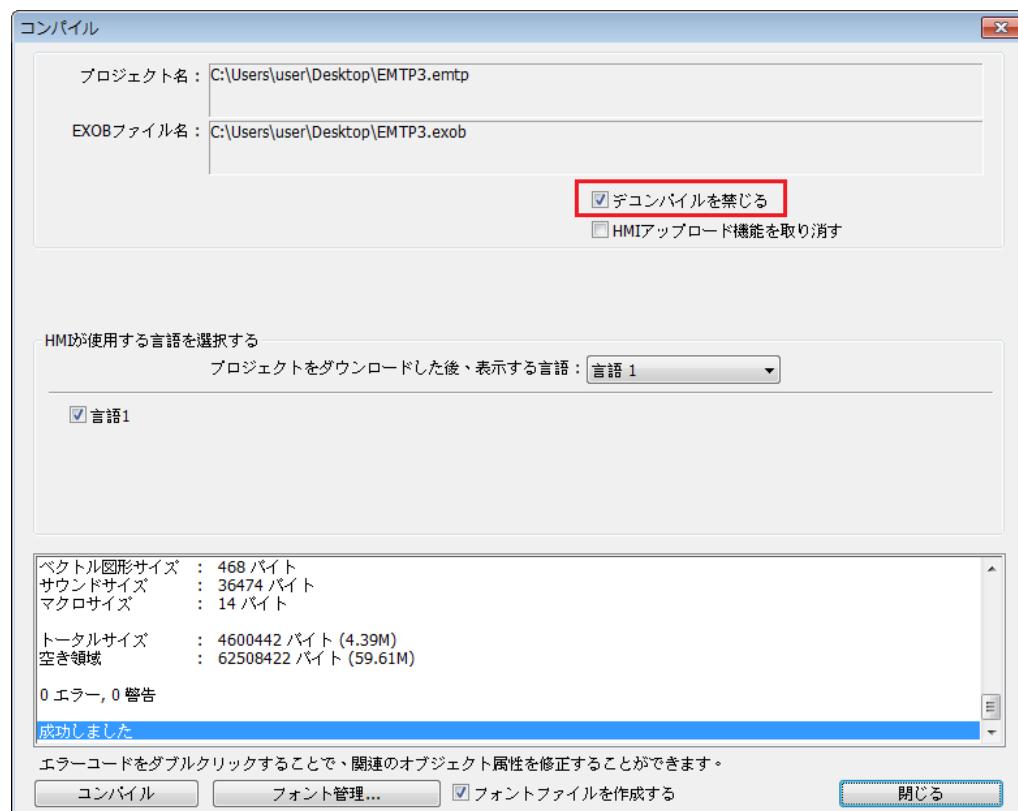
30.2. EXOB パスワード

プロジェクト(.emtp)を編集完了後、システムが提供するコンパイル機能を通じて、.emtp プロジェクトを HMI にダウンロードするのに所要する.exob ファイルにコンパイルすることができます。ユーザーはコンパイルウインドウに[EXOB パスワード](範囲:0~4294967295)を設定でき、その後でこの.exob ファイルを.emtp プロジェクトにデコンパイルしたかったら、このパスワードを入力する必要があります。デコンパイルする時、パスワードの入力を 3 回間違ったら、EasyBuilder Pro を再起動してください。



30.3. デコンパイル禁止

プロジェクト(emtp)を編集完了後、システムが提供するコンパイル機能を通じて、.emtp プロジェクトを HMI にダウンロードするのに所要する.exob ファイルにコンパイルすることができます。ユーザーはコンパイルウインドウに[デコンパイルを禁じる]を設定でき、そうしたら、システムは[EXOB パスワード]の設定を無視し、この.exob ファイルが.emtp プロジェクトにデコンパイル不可になります。

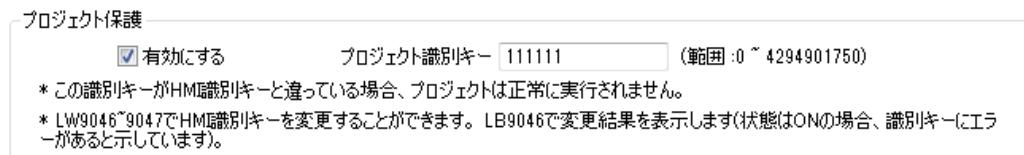


30.4. EXOB ファイルをアップロード禁止機能

EasyBuilder Pro は EXOB ファイルをアップロードするのを禁止する機能のために、システムレジスタ LB-9033 を提供します。このアドレスが ON に設定された時、即ちアップロード後に得た.exob ファイルの大きさは 0 バイトになり、デコンパイルできません。また、何の設定を変更したい場合、設定値を更新するため、HMI を再起動しなければなりません。

30.5. 識別キー

ユーザーのプロジェクトは特定の HMI でのみ実行できると制限できます。下図は[システムパラメータ設定] » [一般的な属性]タブの[プロジェクト保護]の設定画面です。



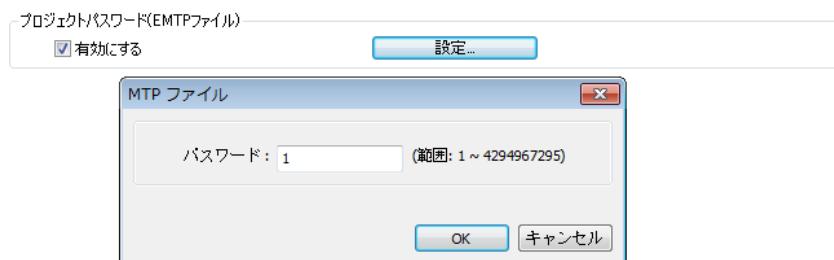
[プロジェクト保護]機能を使用する時、[プロジェクト識別キー](範囲:0~4294901750)を設定でき、それにシステムレジスタ LW-9046 及び LW-9047(32-bit)を合わせ、HMI の[HMI 識別キー]を設定する必要があります。その数値は読み取られ、或いはリモートで書き込まれません。コンパイルした後に得た.exob ファイルは[HMI 識別キー]と[プロジェクト識別キー]が同じの HMI でのみ実行されます。もし[HMI 識別キー]と[プロジェクト識別キー]が同じでない場合、LB-9046 の状態は ON に設定されます。[HMI 識別キー]の設定を変更したい場合、設定値を更新するため、HMI を再起動しなければなりません。

Note

- [HMI 識別キー]と[プロジェクト識別キー]が異なる場合、HMI と装置の間では通信できません。
-  このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

30.6. EMTP パスワード

プロジェクト(.emtp)を編集完了後、.emtp プロジェクトを保護するパスワードを設定することができます。[システムパラメータ設定] » [ユーザーパスワード]タブで本機能を設定してください(範囲:1~4294967295)。本機能を使用した後、毎回その.emtp プロジェクトをオープンするとき、まずは正しいパスワードを入力し、それでこそオープンして編集できます。



Note

- “ウインドウコピー”機能を使用する時、コピーしたいソースプロジェクトには EMTP パスワード保護が設定してある場合、まずは正しいパスワードを入力し、それでこそシステムがウインドウコピーの機能を実行できます。

31. cMT Diagnoser

本章では、cMT 診断器の関連機能について説明します。

31.1.	概要	31-2
31.2.	cMT 診断器の使用方法	31-2
31.3.	cMT 診断器のユーザーインターフェース	31-4
31.4.	補足説明	31-19

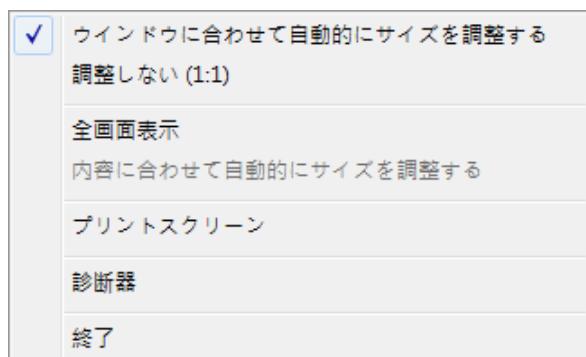
31.1. 概要

cMT 診断器はオブジェクトのアドレス状態を監視、通信パケットの状態を閲覧、MQTT 及びマクロをデバッグすることに用いられます。

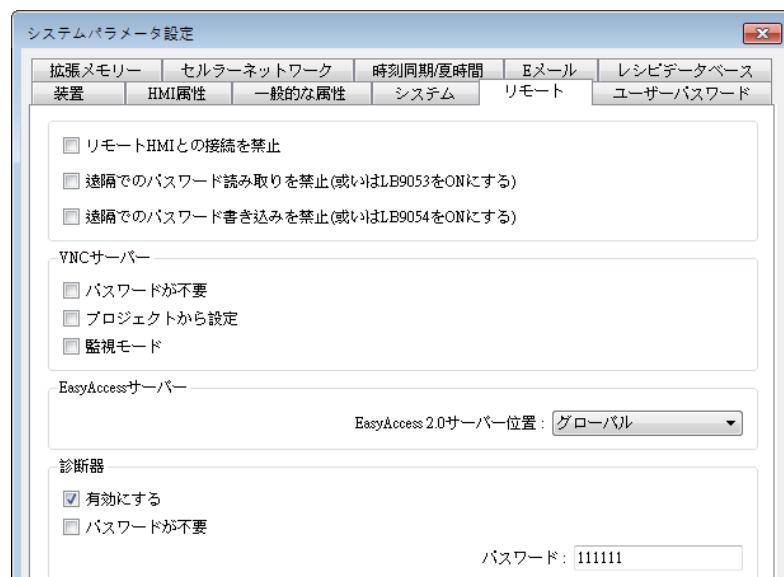
31.2. cMT 診断器の使用方法

cMT 診断器の使い方はシミュレーションでの使用と HMI での使用に分けられ、その中の HMI での使用にはまた cMT Viewer に通じて診断器を開く、及び直接に cMT 診断器で HMI に接続して使用することに分けられます。以下では順に説明致します：

- オンラインシミュレート或いはオフラインシミュレートする段階で、cMT Viewer のシミュレート画面でマウスを右クリックし、診断器を選択します。

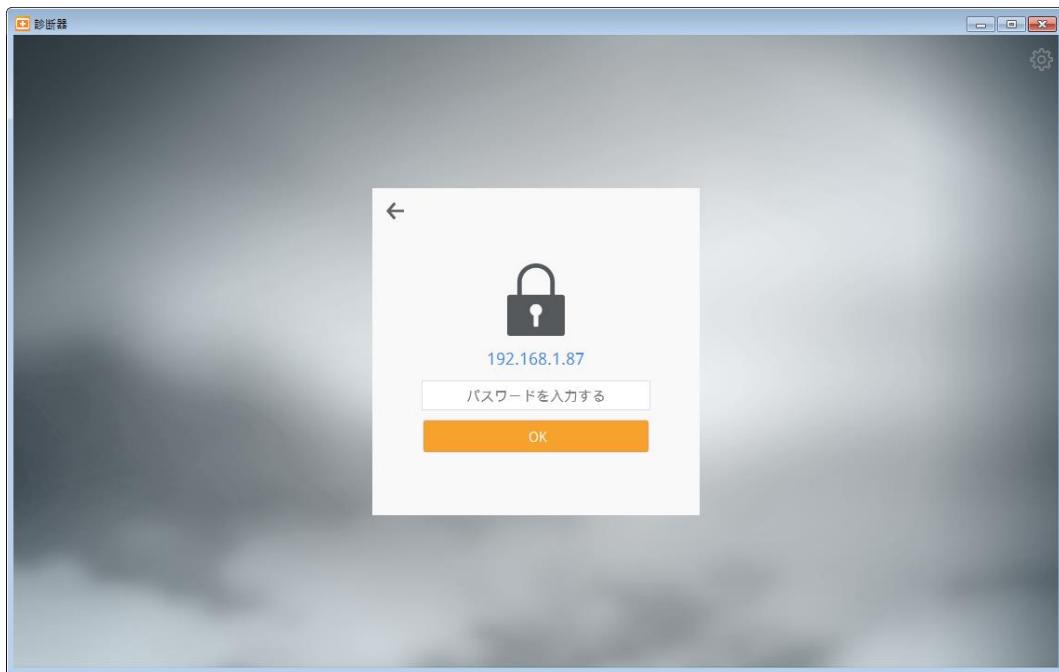


- 診断器で HMI に接続したい場合、まず、プロジェクトの[システムパラメータ設定]>>[リモート]タブで診断器を有効にしてください。

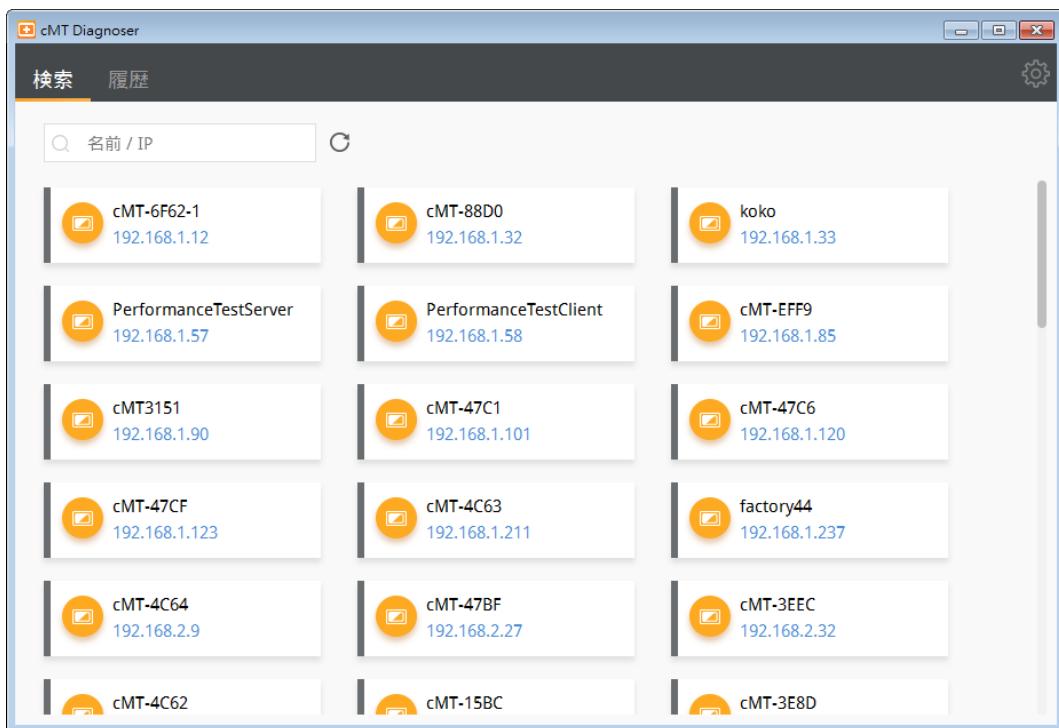


プロジェクトを HMI にダウンロードした後、cMT Viewer で診断器を起動します。[パスワード不要]にチェックマークを入れていない場合、診断器を実行するには、パスワードを入

力する必要があります。



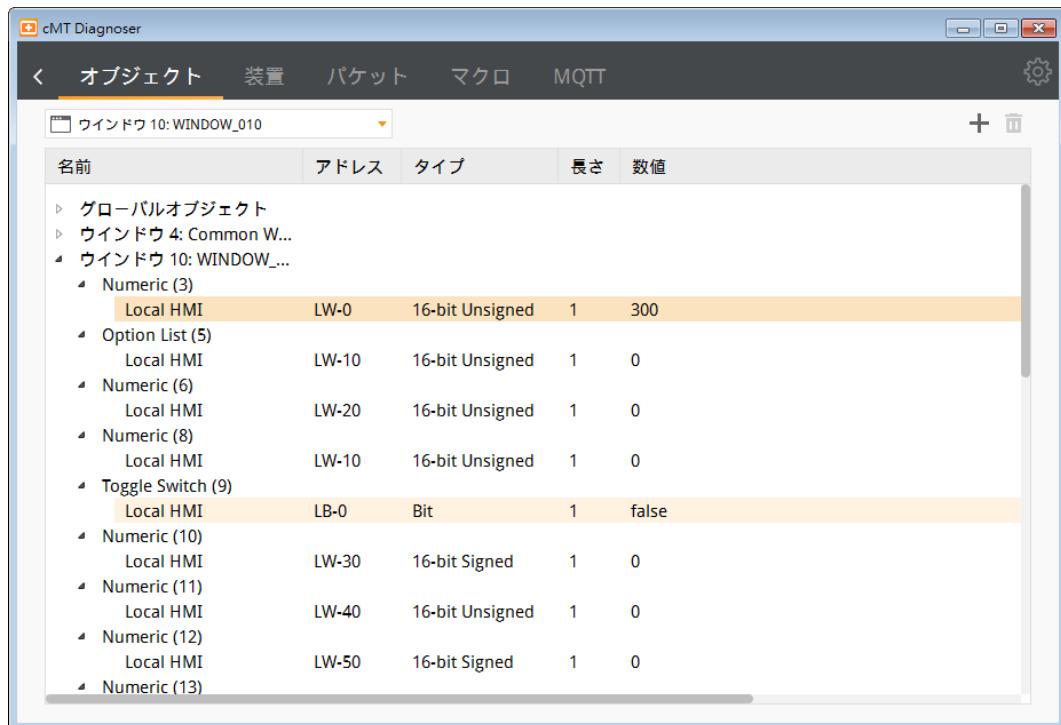
3. EB Pro インストールフォルダで cMTDiagnoser.exe(cMT 診断器の独自プログラムです)を実行し、接続する HMI を検索します。



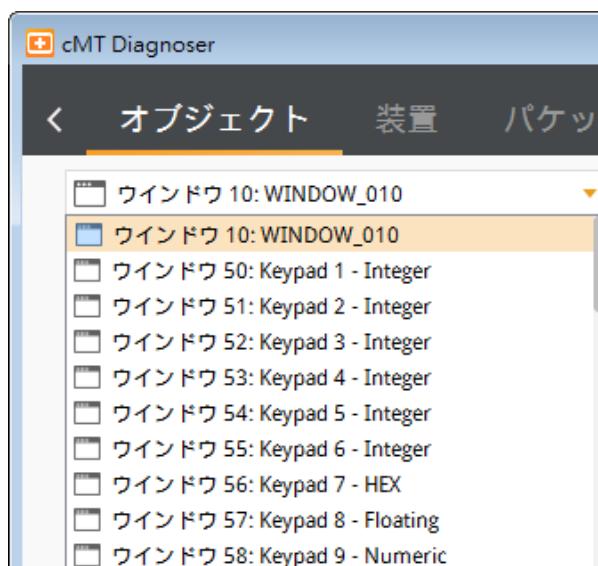
31.3. cMT 診断器のユーザーインターフェース

31.3.1. オブジェクト設定

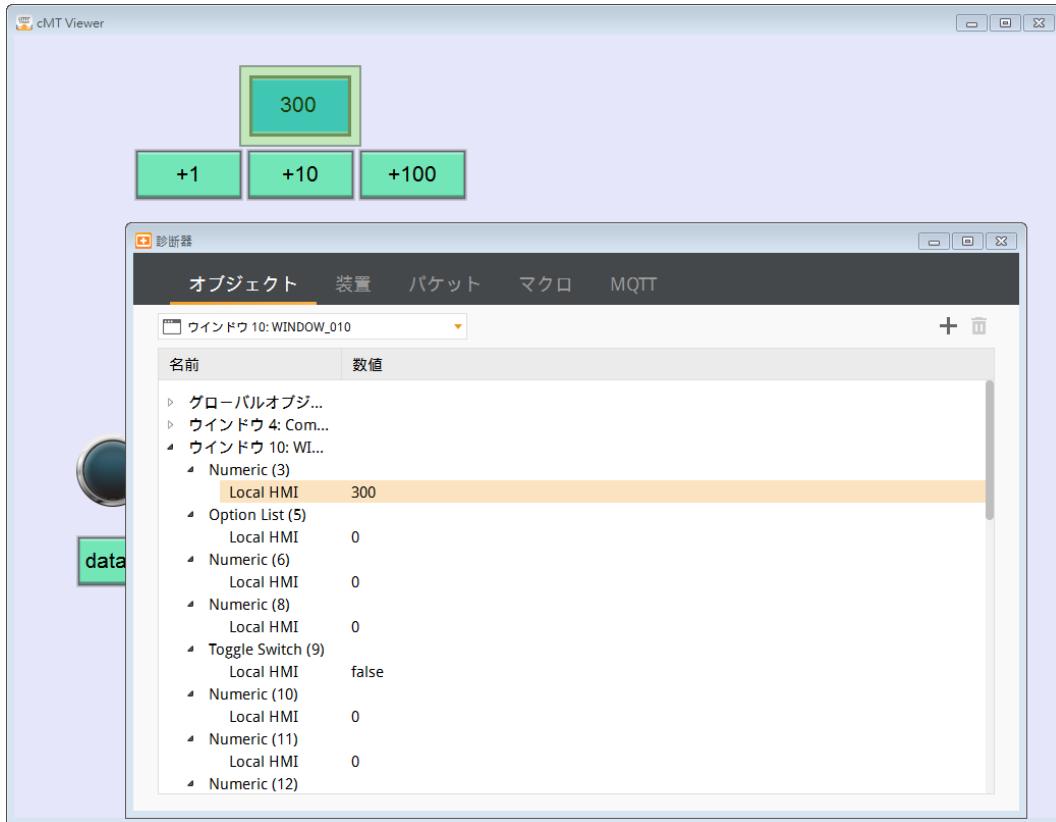
1. 本画面にはウインドウに存在する全てのオブジェクト、アドレス属性及びその状態の内容が表示されます。



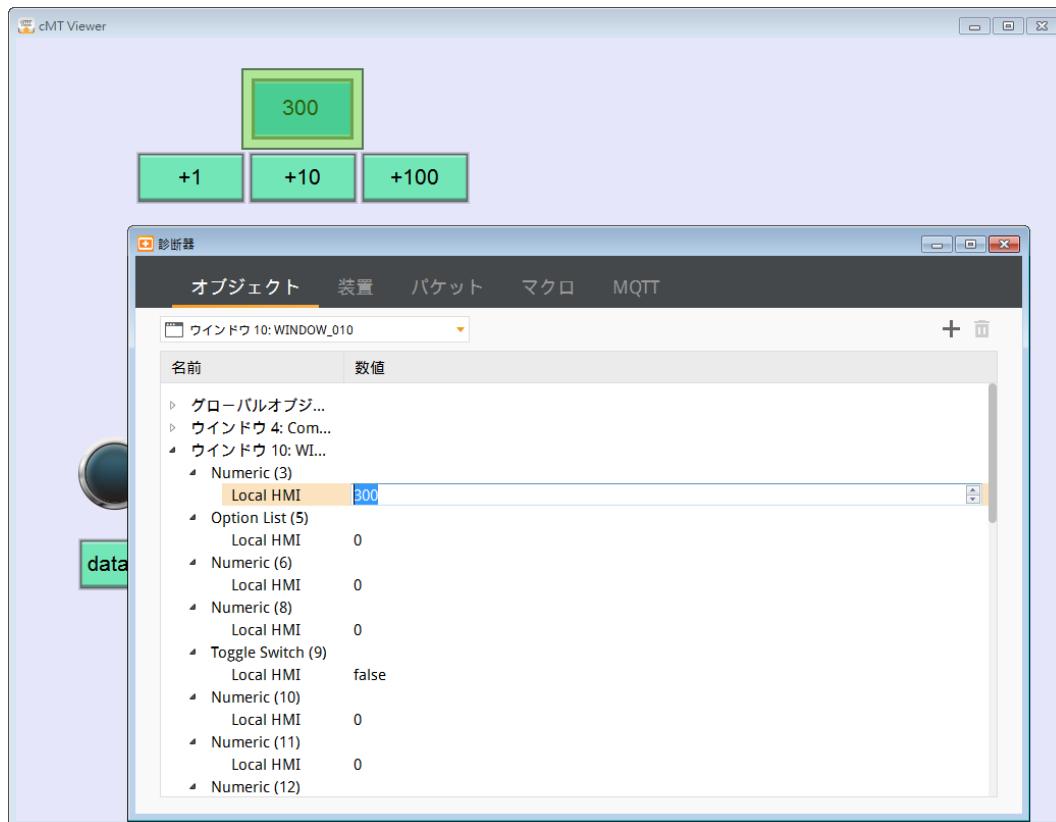
2. ドロップダウンリストで迅速にウインドウを切り替えることができます。



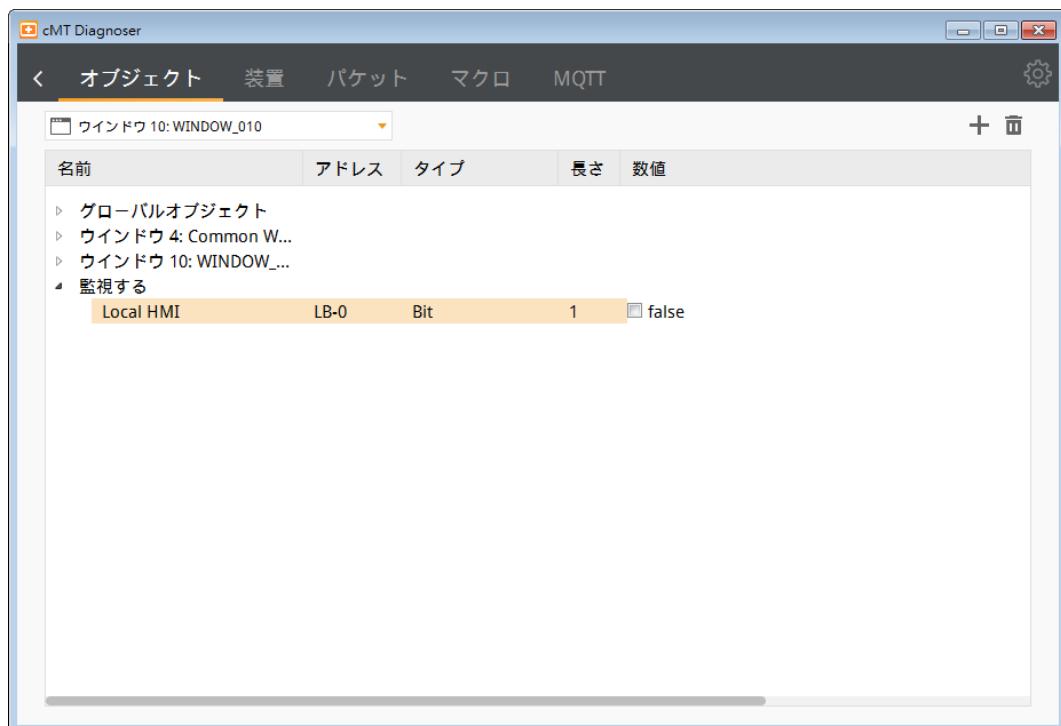
3. 診断器でオブジェクトをクリックすると、画面での対応するオブジェクトには表記が現れます。



4. 診断器でオブジェクトをダブルクリックすると、動的にそのオブジェクトの数値を変更することができます。



5. 診断器で監視したいアドレスを追加/削除できます。+ をクリックすればアドレスを追加でき、特定したアドレスを選択した後、廃止マークをクリックすれば削除できます。



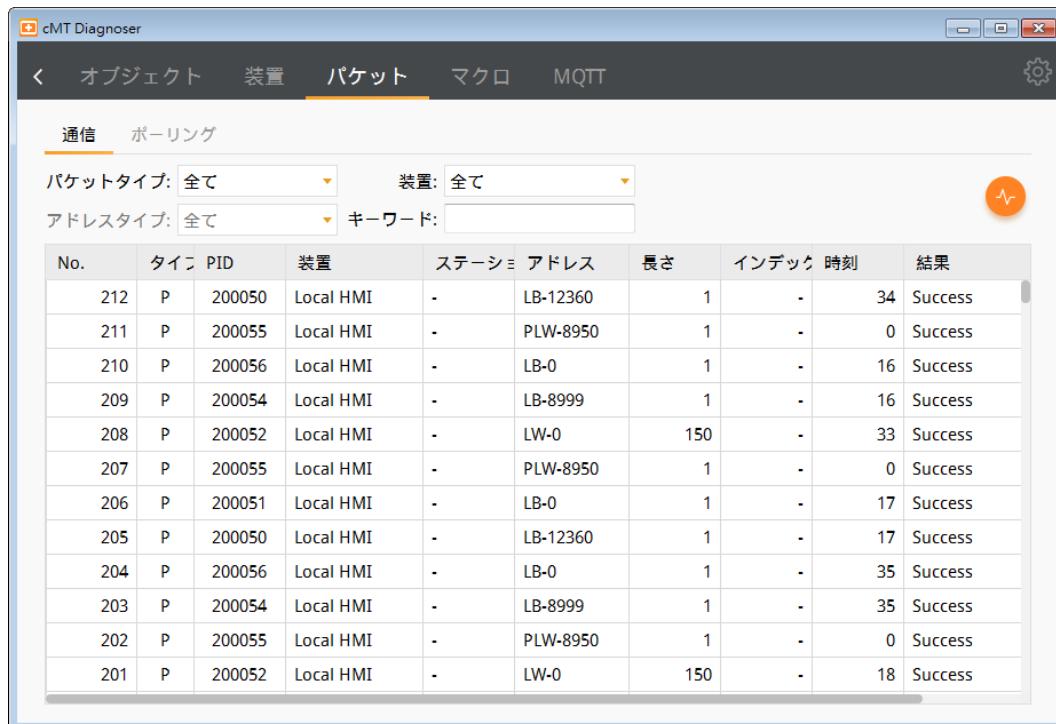
31.3.2. 装置設定

- 本画面では、この HMI と通信している装置の状態及びパラメータが表示されています。



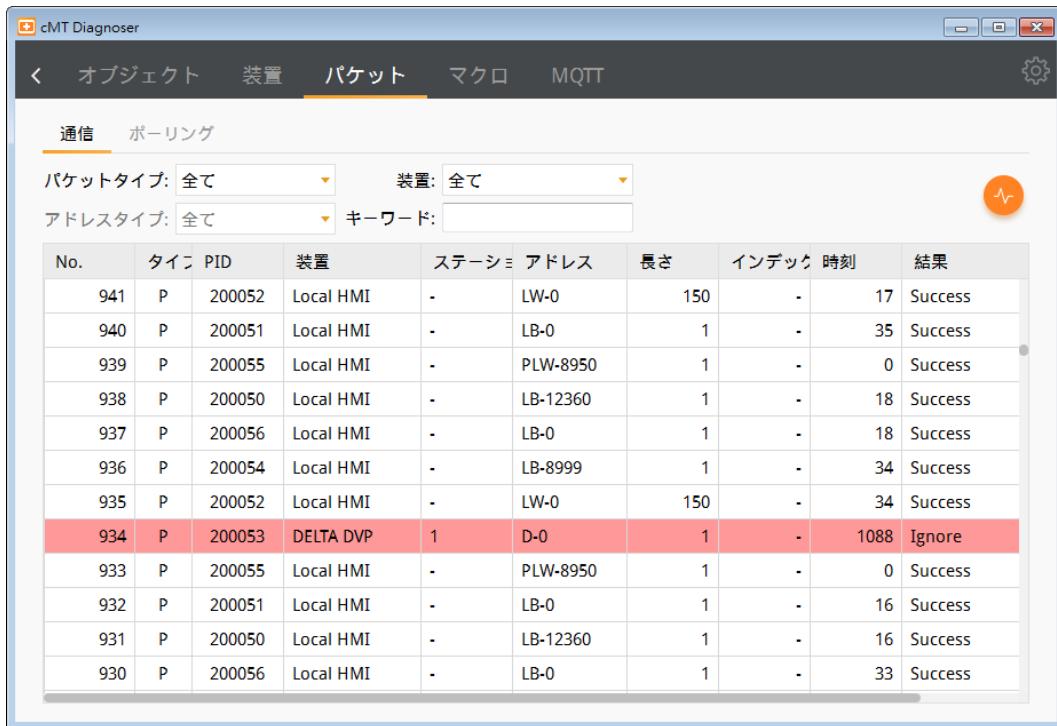
31.3.3. パケット設定

- パラメータを設定した後、を押せば HMI と PLC の間の通信を観察できます。



項目	記述
パケット タイプ	全て activity エリアで全てのパケットを表示します。
読み取り	読み取りパケットだけを表示します。
書き込み	書き込みパケットだけを表示します。
ポーリング	ポーリングパケットだけを表示します。
装置	全て ローカル HMI と PLC の情報を表示します。
	Local HMI ローカル HMI の情報だけを表示します。
	PLC PLC の情報だけを表示します。
ステーション番号	表示したい PLC のステーション番号を選択します。 ([装置]に全てを選択した場合、本機能を使用できません)
アドレス タイプ	全てまたは一部のアドレスタイプを表示させることができます。 ([装置]に全てを選択した場合、本機能を使用できません)
キーワード	入力したキーワードを含むアドレス情報を表示します。

装置との通信に異常が生じた場合、通信ページでは赤色で表示され、エラー原因は結果欄に表示されます。



No.	タイプ	PID	装置	ステーショ	アドレス	長さ	インデック	時刻	結果
941	P	200052	Local HMI	-	LW-0	150	-	17	Success
940	P	200051	Local HMI	-	LB-0	1	-	35	Success
939	P	200055	Local HMI	-	PLW-8950	1	-	0	Success
938	P	200050	Local HMI	-	LB-12360	1	-	18	Success
937	P	200056	Local HMI	-	LB-0	1	-	18	Success
936	P	200054	Local HMI	-	LB-8999	1	-	34	Success
935	P	200052	Local HMI	-	LW-0	150	-	34	Success
934	P	200053	DELTA DVP	1	D-0	1	-	1088	Ignore
933	P	200055	Local HMI	-	PLW-8950	1	-	0	Success
932	P	200051	Local HMI	-	LB-0	1	-	16	Success
931	P	200050	Local HMI	-	LB-12360	1	-	16	Success
930	P	200056	Local HMI	-	LB-0	1	-	33	Success

2. ポーリングパケットは HMI が定期的に問い合わせを行う際に送信するパケットのこととで、開始アドレス及び長さなどの関連情報を含めています。



パケットID	装置	ステーショ	アドレス	長さ	インデックス
100348	Local HMI	-	LB-0	1	-
100349	Local HMI	-	LW-0	51	-
200050	Local HMI	-	LB-12360	1	-
200051	Local HMI	-	LB-0	1	-
200052	Local HMI	-	LW-0	150	-
200053	DELTA DVP	1	D-0	1	-
200054	Local HMI	-	LB-8999	1	-
200055	Local HMI	-	PLW-8950	1	-
200056	Local HMI	-	LB-0	1	-

項目	記述
パケットID	パケット ID でエラーがあるオブジェクトを確認します。
装置	HMI と PLC 装置を表示します。

ステーション番号	PLC のステーション番号を表示します。
アドレス/長さ	パケット内の装置タイプのアドレス及びその長さを表示します。
インデックス	オブジェクトが使用するインデックスレジスタ番号を表示します。

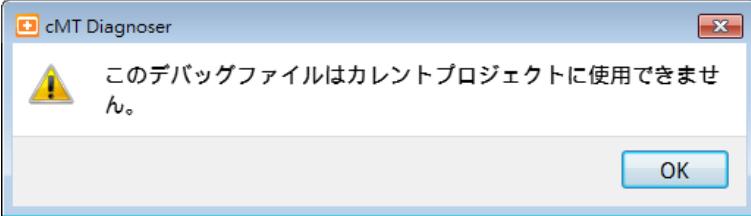
31.3.4. マクロ設定

- ここでマクロを実行・テストできます。



- ここでマクロデバッグ機能を有効にします。

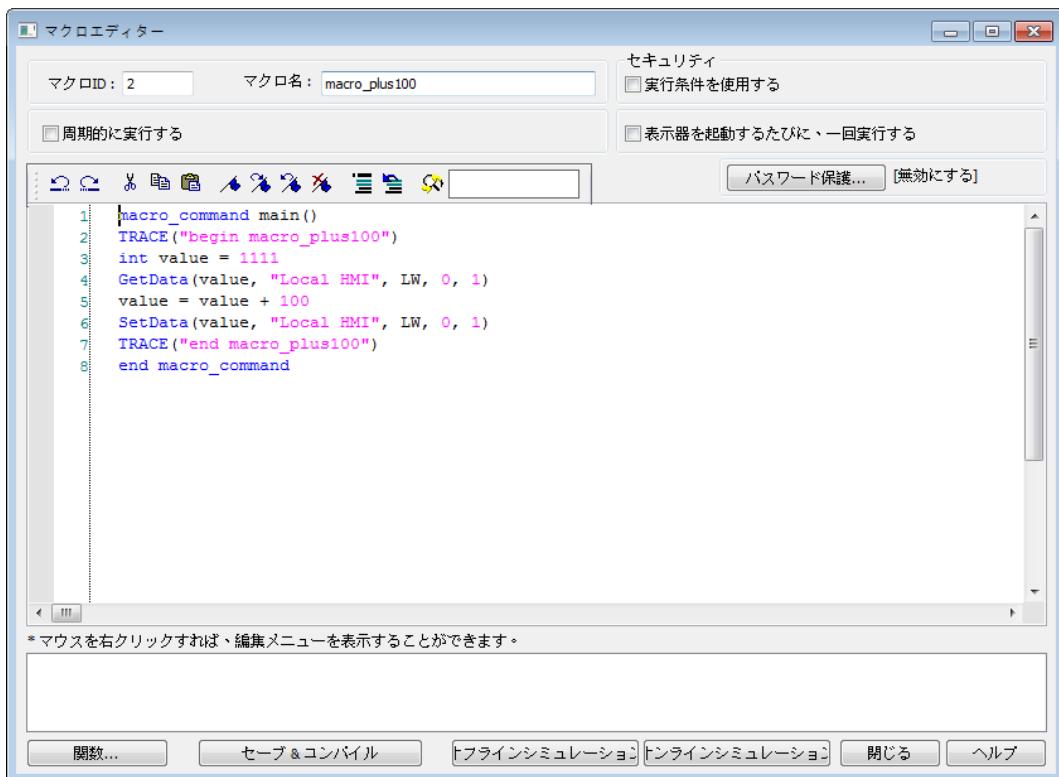


項目	記述
マクロデバッグファイルをロードする	マクロデバッグファイルをロードします。マクロデバッグファイルは EasyBuilder Pro でプロジェクトをコンパイルしている際に生成され、保存先はコンパイルファイルと同じです。拡張子は.debug です。
	マッチしていないマクロデバッグファイルをロードした場合、下記のようなウォーニングメッセージが出ます。
	
デバッグを開始	マクロデバッグを開始し、または中断箇所から実行を続けます。
デバッグを停止	マクロデバッグを停止します。
一時停止	手動で現在実行中のマクロを一時停止します。マクロの行の前でクリックすれば、中断箇所を生成させ、マクロが中断箇所まで実行し、そして一時停止して次の実行コマンドを待つようになります。
ステップイントウ	マクロを逐次に実行します。次のステップが関数である場合、関数の内容を逐次にデバッグします。

ステップ	マクロを逐次に実行します。次のステップが関数である場合、
オーバー	関数の全ての内容を送信し、そして関数の結果を送信しますが、逐次に関数の内容をデバッグしません。
検索	マクロの中のキーワードを検索します。
監視	デバッグ過程中で変数を新規追加し、数値のリアルタイムの変化を監視します。
出力	システム情報及び Trace 関数を出力します。

診断器のマクロデバッグ機能を有効にするには、三種の方法があります。各方法ではそれぞれ異なる権限及び操作方式があり、下記にて説明致します。

A. マクロエディターのオンライン/オフラインシミュレーションで実行します。



シミュレーションボタンを押すと、マクロデバッグファイルが自動的に生成され、ソフトウェアは全てのマクロエディターウィンドウを閉じ、cMT Viewer シミュレーション画面をポップアップし、同一時間に診断器も実行され、自動的にマクロデバッグファイルをインポートします。

デバッグしている時に、cMT Diagnoser のマクロタブでマクロを修正できます。マクロデバッグが実行完了後、cMT Viewer を閉じると、マクロエディターウィンドウは再度にオープンされ、診断器で行った変更をマクロエディターウィンドウに更新しますかの確認が現れます。

B. EasyBuilder Pro のオンライン/オフラインシミュレーションで実行します。

コンパイルの際に、マクロデバッグファイルが自動的に生成されます。EasyBuilder Pro でシ

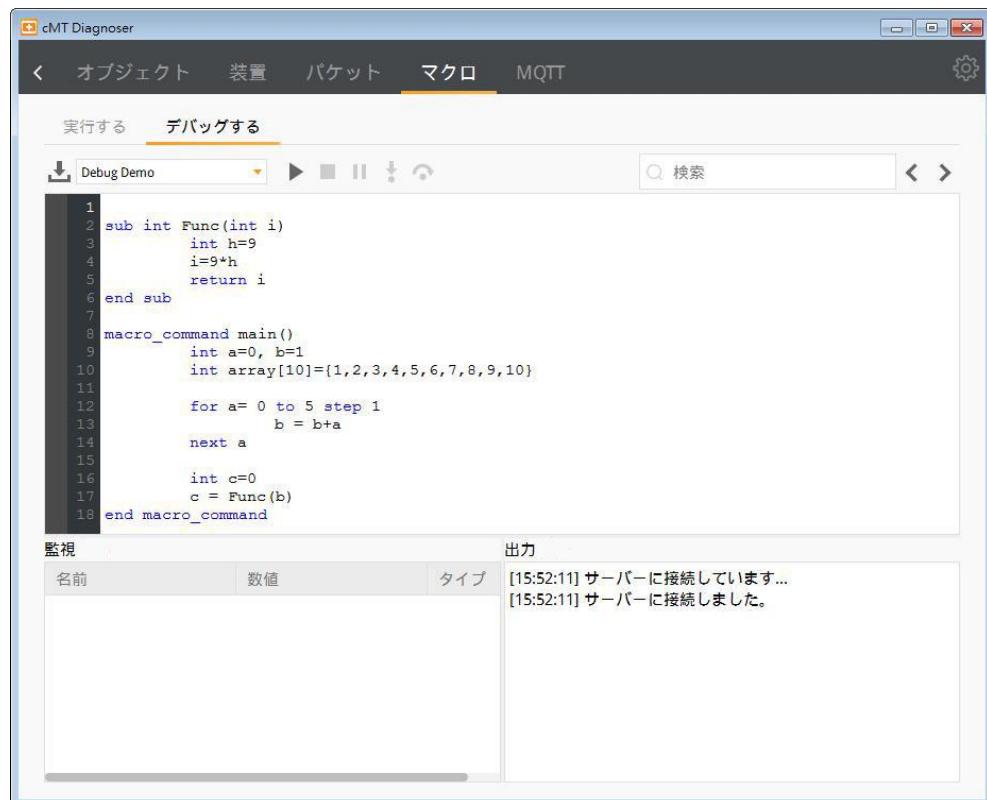
ミュレートする際に cMT Viewer シミュレーションウインドウが現れると、ウインドウで右クリックして診断器をオープンします。システムがマクロデバッグファイルを検出する場合、自動的にインポートします。このモードでは、マクロにデバッグだけを実行可能で、変更はできません。

注意：プロジェクトの中には少なくとも 1 個のマクロが存在する場合のみ、コンパイルした後マクロデバッグファイルが現れます。

- C. プロジェクトファイルを HMI にダウンロードした後、cMT Diagnoser で操作します。
手動でマクロデバッグファイルをインポートする必要があります。このモードでは、マクロにデバッグだけを実行可能で、変更はできません。

マクロデバッグの例

- 以下では、簡単な例でマクロデバッグの流れ及びその設定を説明します。[デバッグを開始する](矢印ボタン)をクリックしてデバッグを開始します。



```

1
2 sub int Func(int i)
3     int h=9
4     i=9*h
5     return i
6 end sub
7
8 macro_command main()
9     int a=0, b=1
10    int array[10]={1,2,3,4,5,6,7,8,9,10}
11
12    for a= 0 to 5 step 1
13        b = b+a
14    next a
15
16    int c=0
17    c = Func(b)
18 end macro_command

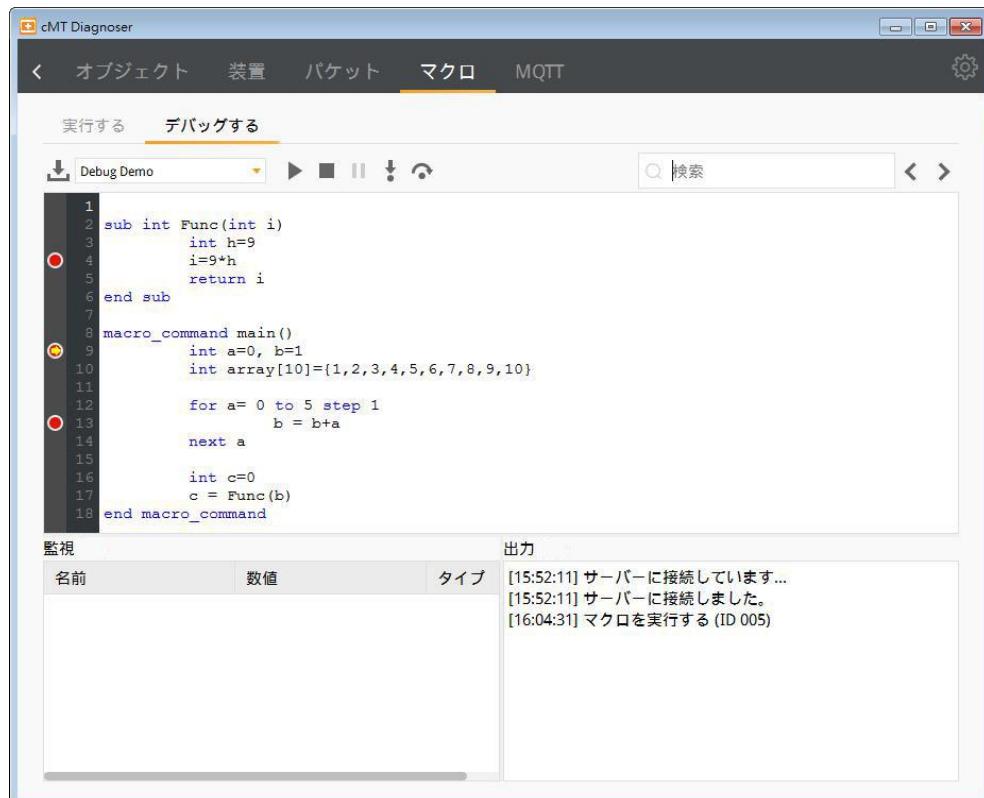
```

監視	出力	
名前	数値	タイプ
		[15:52:11] サーバーに接続しています...
		[15:52:11] サーバーに接続しました。

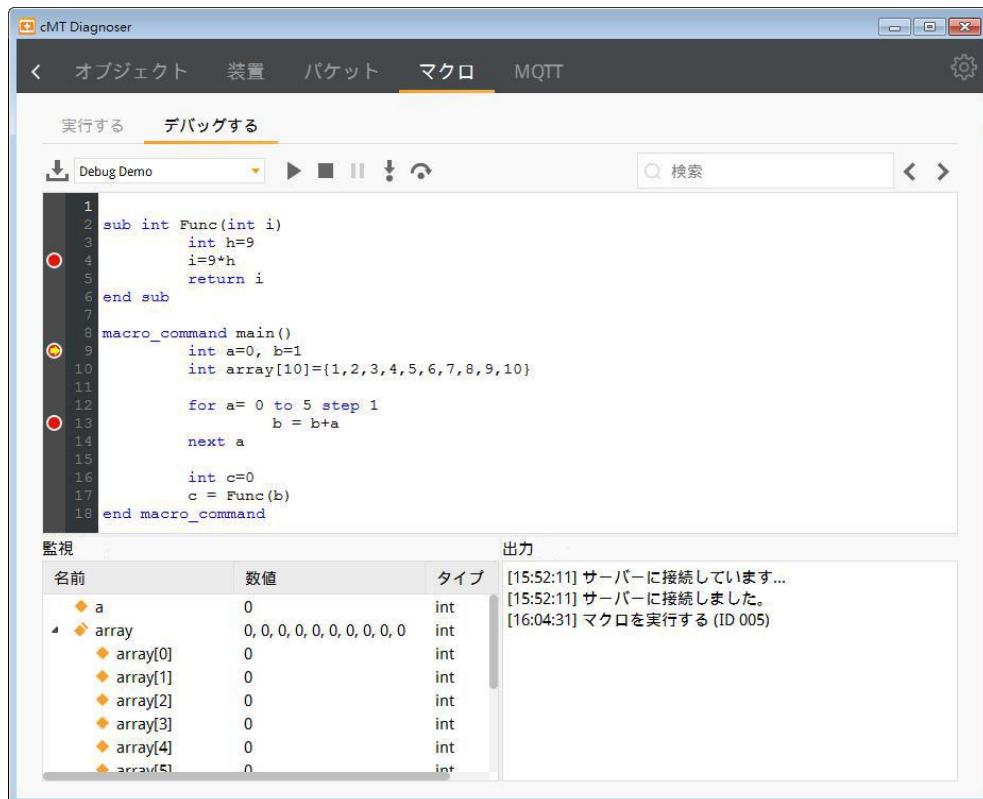
- 行数左側の灰色のゾーンで中断箇所(赤色リング)を追加または取り消します。



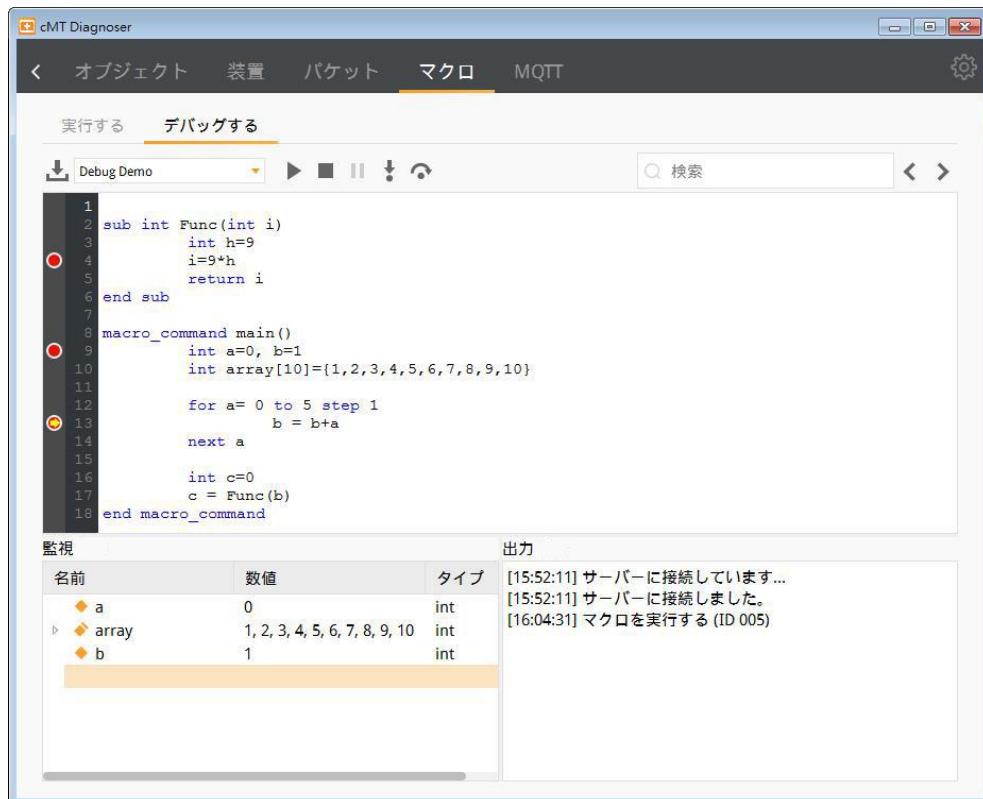
3. マクロをトリガーした後、マクロは1個目の中断箇所に止まります(黄色矢印アイコン)。



4. この時は監視ページで監視したい変数を追加でき、変数が配列である場合でも、監視ウィンドウで展開してその数値を閲覧できます。

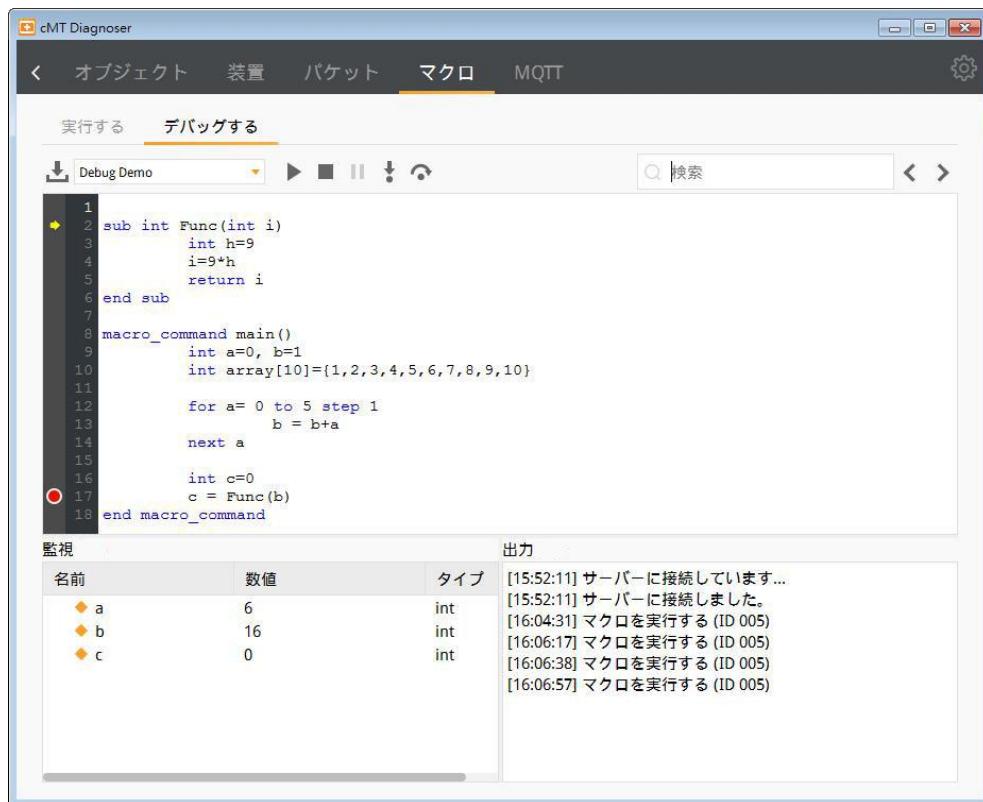


5. 矢印ボタンを押してデバッグを開始すると、マクロは次の中断箇所まで実行します。変数の数値もそれとともに変わります。

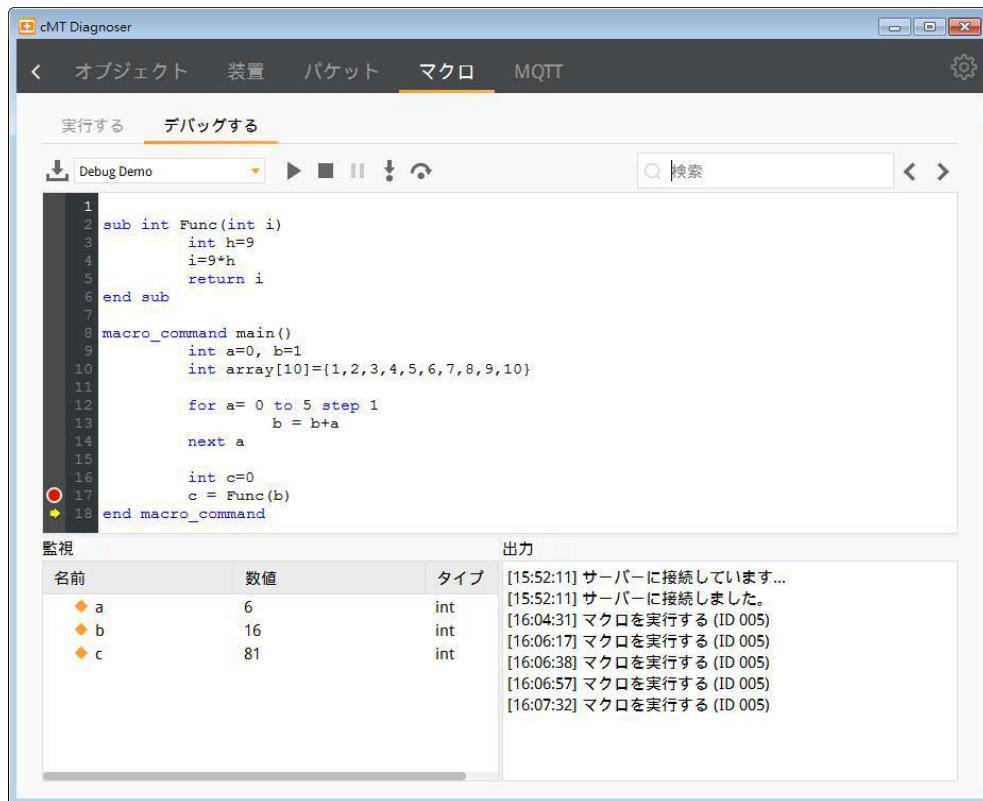


6. ステップイントウ (Step into) はマクロを逐次に実行させるコマンドです。次のステップは関数を実行することであれば、マクロを関数に入らせ、関数内容を逐次にデバッグし変数の

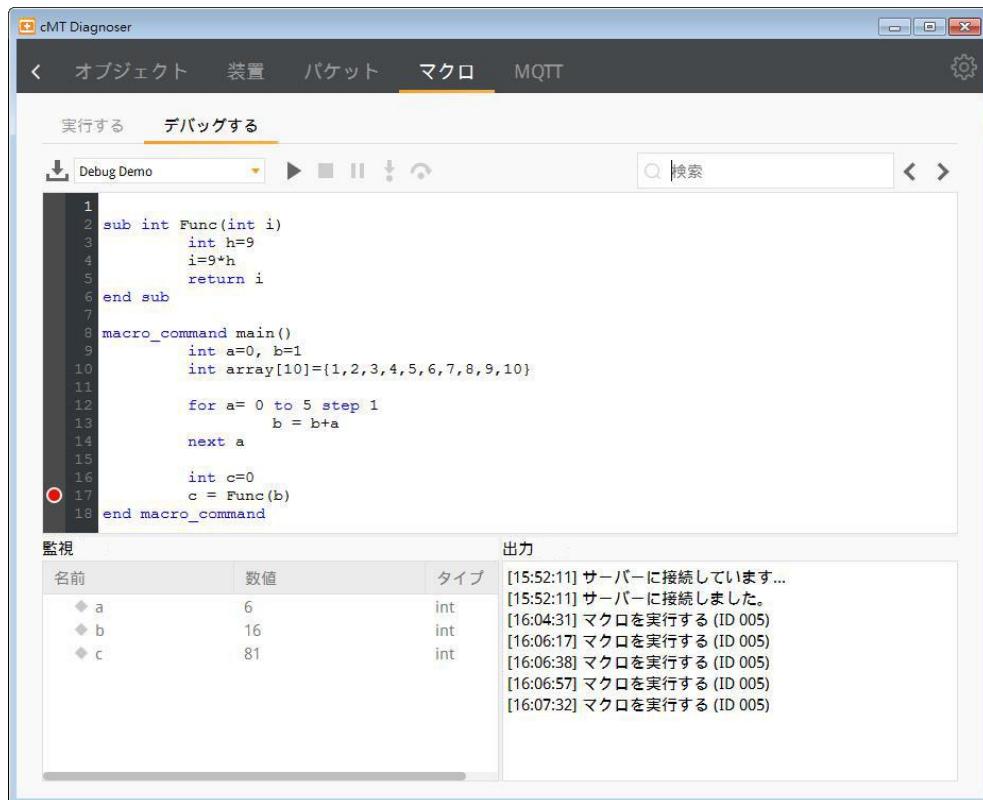
数値を更新します。



7. ステップオーバー(Step over)はマクロを逐次に実行させるコマンドです。次のステップは関数を実行することであれば、関数に入って逐次にデバッグすることをしなく、直接に関数の実行結果を送信することになります。但しもし関数に中断箇所が設定された場合、そこで停止します。



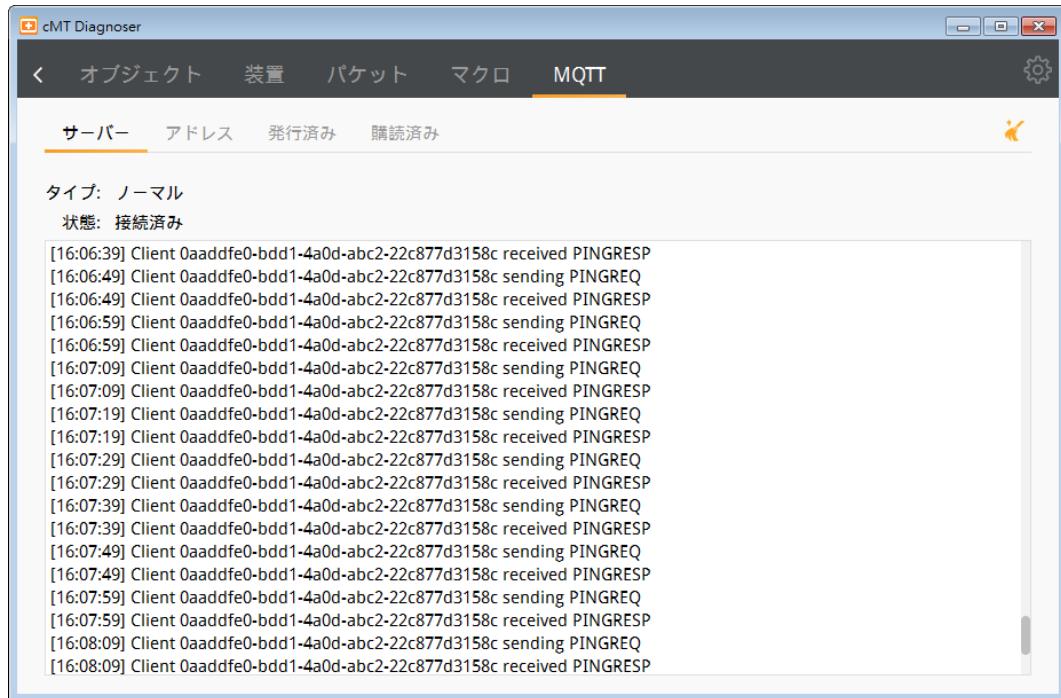
8. 停止ボタンを押し、マクロデバッグ機能を終了します。



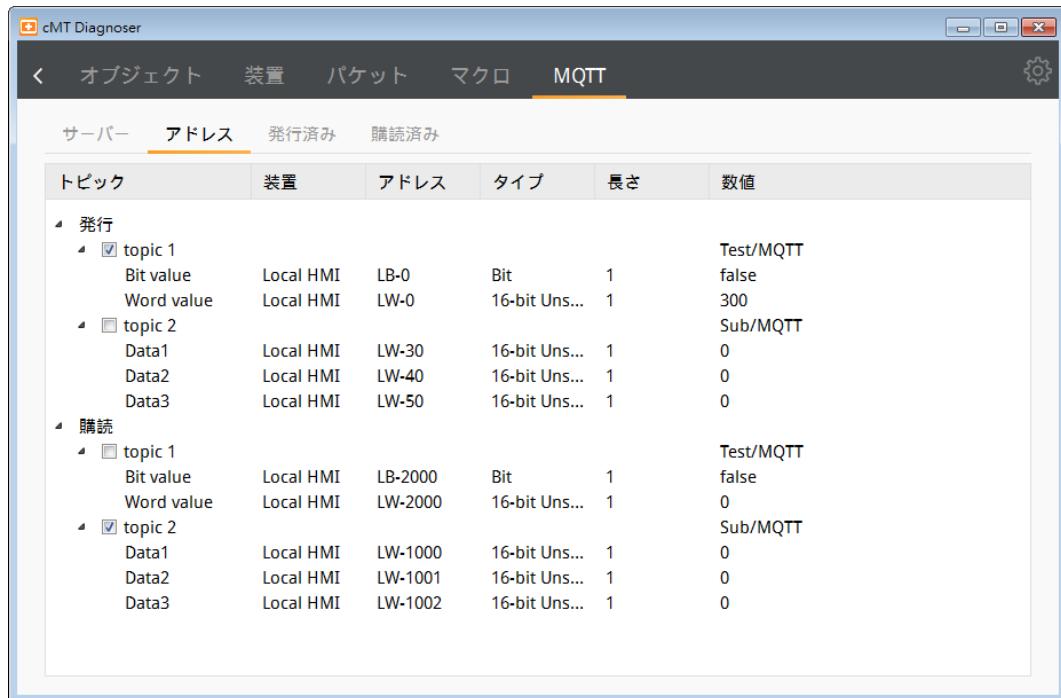
31.3.5. MQTT 設定

1. サーバータブでは MQTT サーバーのタイプと現在の状態が表示されます。コネクション・

ログはノーマル/Azure IoT Hub/Sparkplug/Google Cloud IoT Core にサポートされ、AWS IoT にサポートされていません。



- アドレスタブでは HMI が発行・購読しているトピックが表示されます。トピックの前にチェックを入れたら、診断器に全ての発行・購読された MQTT メッセージを受信させ、及び直接にアドレスのデータを変更でき、MQTT の診断をより容易にさせます。[ノーマル]のサーバーにのみ対応しています。



- 発行済みタブでは HMI が発信したメッセージ内容が表示されます。トピックに圧縮フォーマットまたは Raw Data を使用する場合、16進数で表示されます。[ノーマル]のサーバーに

のみ対応しています。



- 購読済みタブでは HMI が購読したメッセージが表示されます。トピックに圧縮フォーマットまたは Raw Data を使用する場合、16 進数で表示されます。[ノーマル]のサーバーにのみ対応しています。



31.4. 補足説明

- Tag PLC を含めた全ての HMI と通信できる PLC アドレスをサポートします。

2. 診断器は前回閉じた時の状態を記憶します。直接に cMT Viewer を閉じれば、次回シミュレーションする時に、診断器が自動的にポップアップされます。逆に、cMT Viewer を閉じる前に診断器を閉じれば、次回シミュレーションする時は、手動で診断器をオープンする必要があります。
3. マクロがパスワードで保護された場合、マクロデバッグファイルを生成できません。
4. 個別のマクロにパスワードが設定されている場合、デバッグページでパスワードを入力しロックを解除する必要があります。ロック解除されていなかったら、読み取り専用のマクロの場合、マクロの内容が見られますが、シミュレーション実行時では変更できません。暗号化されたマクロの場合、マクロの内容が見られません。
5. コンパイルに失敗したマクロはコンパイルが完成していないリストに入れられ、マクロデバッグファイルに含まれません。
6. 同一時間では、1個のマクロにのみデバッグできます。
7. マクロデバッグを実行している時、中断箇所は実行する必要がない所、例えば空白、或いは初期値を付与されていない変数宣言などである場合、下図の第7行に示されたように、中空で表示されます。

The screenshot shows the cMT Diagnoser application window. The title bar says "Diagnoser". The menu bar includes "Object", "Device", "Packet", "Macro" (which is highlighted), and "MQTT". Below the menu is a toolbar with "Execute" and "Debug" tabs, where "Debug" is selected. A dropdown menu shows "Debug Demo". To the right of the dropdown are control buttons for play, stop, and step, along with a search bar and navigation arrows. The main area is a code editor with the following content:

```
1 |  
2 sub int Func(int i)  
3     int h=9  
4     i=9*h  
5     return i  
6 end sub  
7  
8 macro_command main()  
9     int a=0, b=1  
10    int array[10]={1,2,3,4,5,6,7,8,9,10}  
11  
12    for a= 0 to 5 step 1  
13        b = b+a  
14    next a  
15  
16    int c=0  
17    c = Func(b)  
18 end macro_command
```

Below the code editor are two panes: "Watch" and "Output". The "Watch" pane is empty. The "Output" pane shows the following log entries:

```
[11:20:18] Connecting to server...  
[11:20:18] Server connected.
```

○

32. FTP サーバーの運用

本章では、FTP サーバーの運用について説明します。

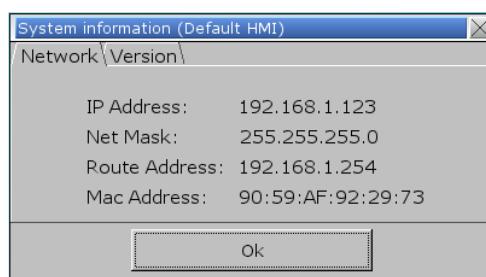
32.1. 概要	32-2
32.2. FTP サーバーにログインする手順	32-2
32.3. 履歴データのバックアップ及びレシピデータの更新	32-3

32.1. 概要

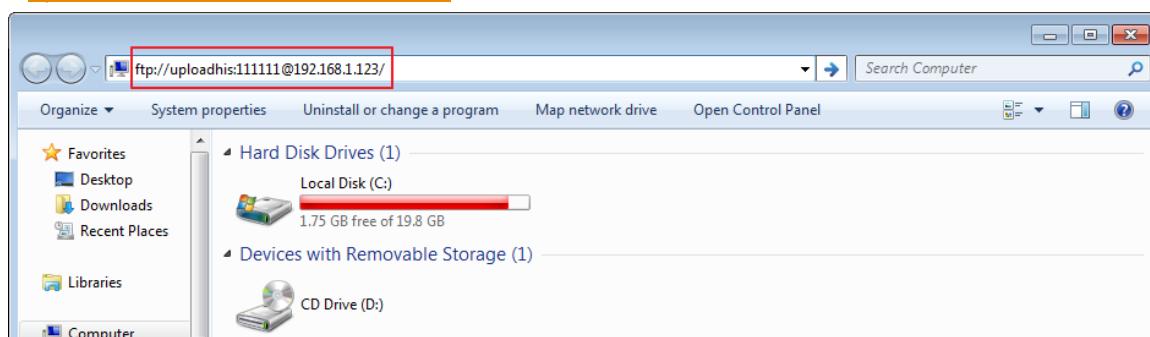
SD カード、USB ディスク或いは EasyPrinter サーバーを使って履歴データを HMI から PC にバックアップする以外、FTP サーバーも利用できます。プロジェクトを HMI にダウンロードした後、FTP サーバーを通して履歴データ及びレシピデータのバックアップ、またはレシピデータを更新することができ、しかし FTP サーバーに保存しているデータは削除されません。USB ディスクまたは SD カードを外す前、先に FTP との接続を切断してください。さもなければ、次回 USB/SD を挿入する時に、使用できない可能性があります。

32.2. FTP サーバーにログインする手順

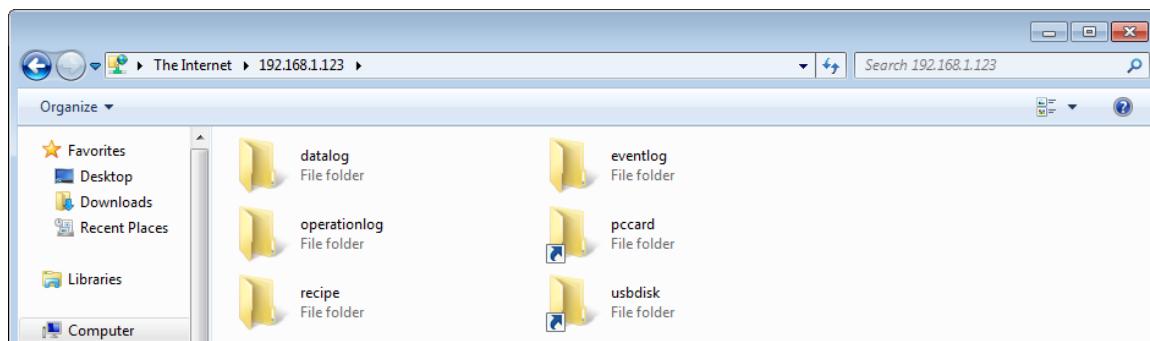
- FTP サーバーにログインする前、まずは HMI の IP アドレスを確認してください。



- [マイコンピュータ]アドレスに HMI の IP アドレス <ftp://192.168.1.123>(例)を入力し、アカウント uploadhis にログインします。パスワードは HMI の history upload password(パスワードを変更したことがない場合、デフォルトパスワードは 111111)です。或いは直接に <ftp://uploadhis:111111@192.168.1.123/>を入力します。

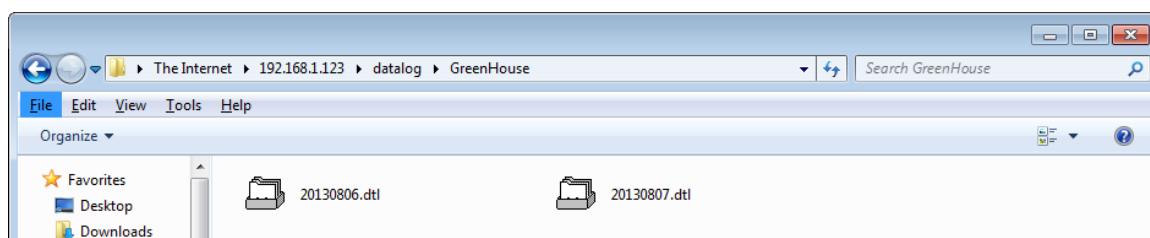


- IP アドレスを入力した後、FTP アドレスは <ftp://192.168.1.123/>と表示し、下図で示したフォルダが見られます。

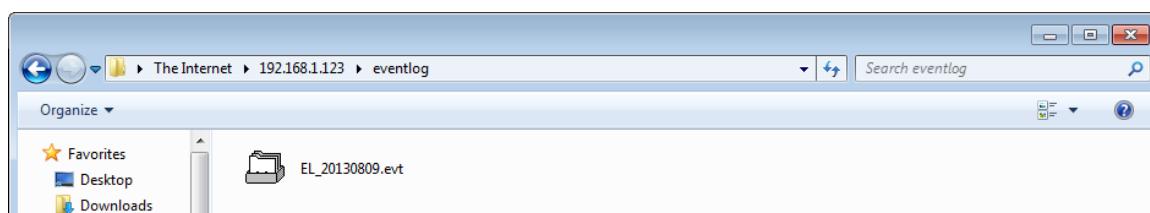


32.3. 履歴データのバックアップ及びレシピデータの更新

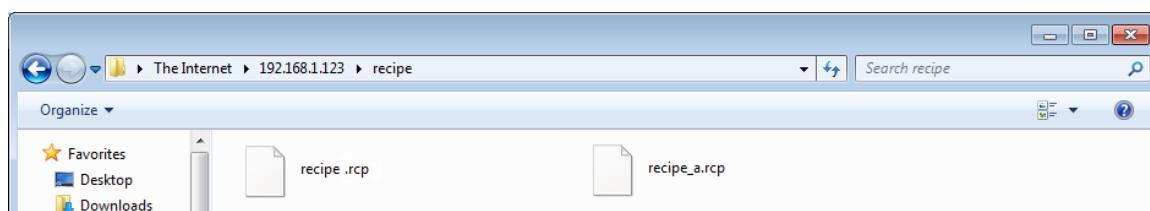
- データサンプリング記録をバックアップする
 1. Datalog フォルダをクリックします。
 2. ファイル名をクリックした後、**datalog** ファイルが見られます。
 3. コピー及び貼り付け機能で、データサンプリング記録を PC に保存します。



- アラーム及びイベントログをバックアップする
 1. eventlog フォルダをクリックした後、イベントログファイルが見られます。
 2. コピー及び貼り付け機能で、イベントログを PC に保存します。



- レシピデータをバックアップ・更新する
 1. Recipe フォルダをクリックした後、レシピデータファイルが見られます。
 2. コピー及び貼り付け機能で、レシピデータを PC に保存します。



 Note

- レシピデータベースは1分ごとに自動的に一回セーブするので、`recipe.rcp` 或いは `recipe_a.rcp` を更新する場合、1分間以内に HMI を再起動してください。そうでないと、新しいレシピデータは本来のレシピデータに重ねられます。
- ユーザーもシステムレジスタ LB-9047(HMI を再起動する)及び LB-9048(HMI 保護を再起動する)で HMI を再起動してもいいです。LB-9047 及び LB-9048 を使用すると、まず LB-9048 を ON に設定した後、また LB-9047 を ON に設定して、それでこそ HMI を再起動することができます。

33. EasyDiagnoser

本章では、EasyDiagnoser を使用する方法について説明します。

33.1.	概要	33-2
33.2.	設定手順	33-2
33.3.	EasyDiagnoser 設定	33-3
33.4.	エラーコード	33-8
33.5.	ウインドウ調整	33-10

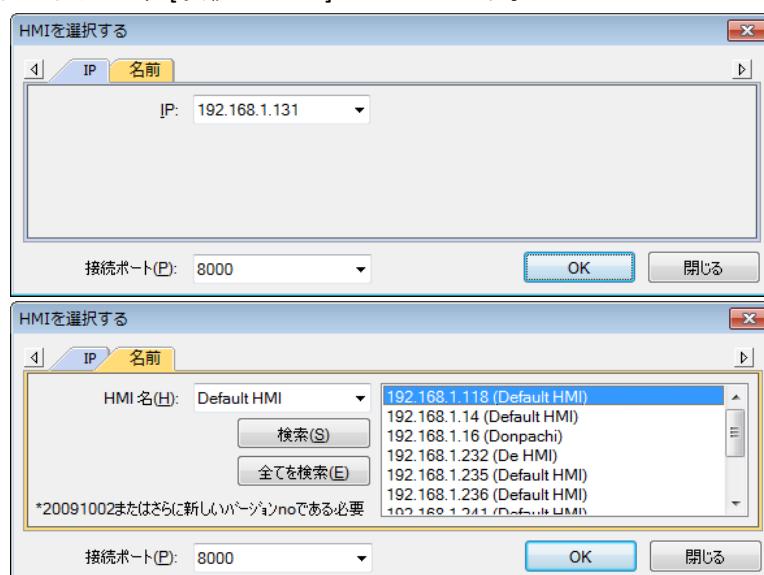
33.1. 概要

EasyDiagnoser は HMI と装置の間の通信は正常であるかを検知するツールです。

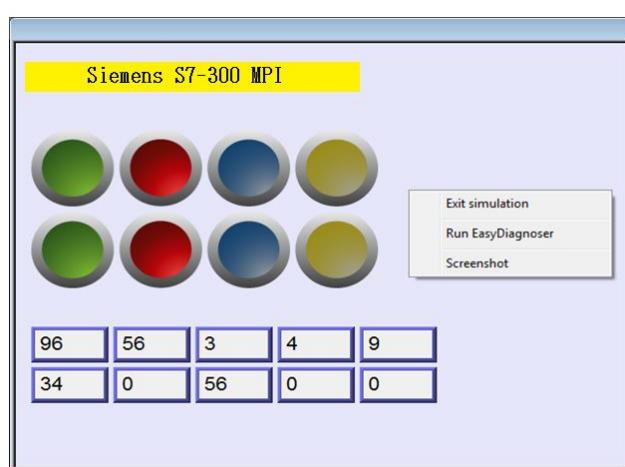
33.2. 設定手順

以下は EasyDiagnoser を設定する手順です：

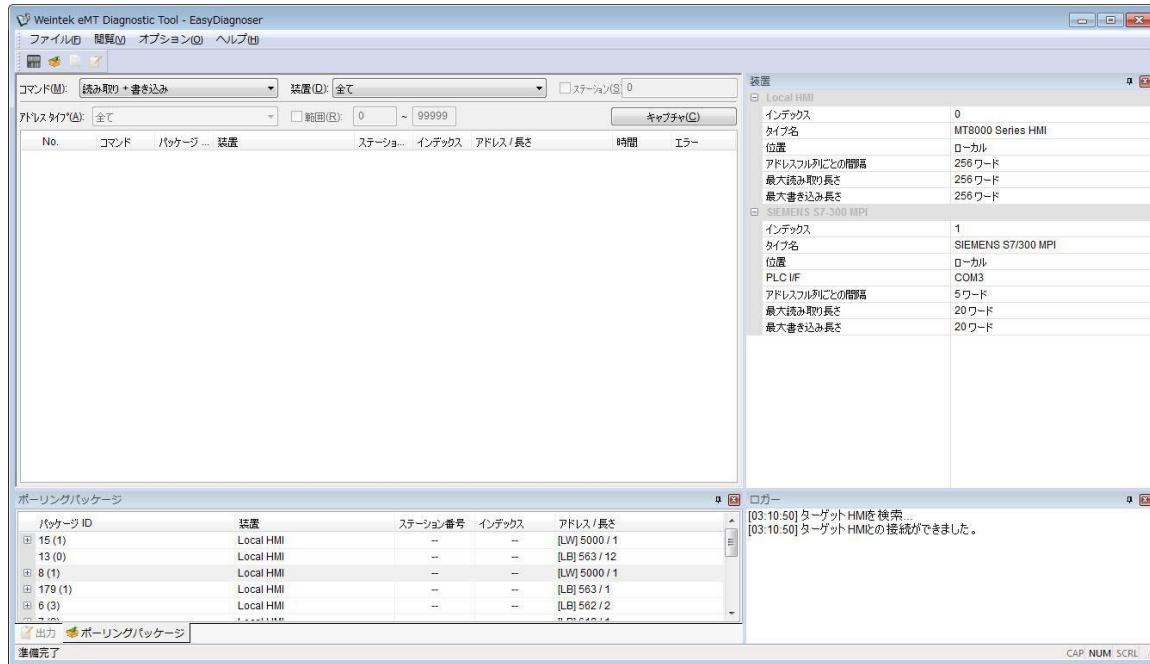
1. Utility Manager をオープンし、分析・テストツールタブで EasyDiagnoser をクリックします。
2. 通信したい HMI の IP アドレスを設定します。自分で IP アドレスを入力するか、或いは[全てを検索]機能を使用し、[接続ポート]を入力します。



EasyBuilder Pro で On Line Simulation を実行する時、右ボタンをクリックすると、[Run EasyDiagnoser]が選択でき、EasyDiagnoser に入ります。



3. 以上の設定を終えたら、[OK]を押します。EasyDiagnoser 操作画面は下図に示す通りです：



33.3. EasyDiagnoser 設定

33.3.1. メインメニュー

項目	記述
ファイル	名前をつけて保存 取り上げた通信資料を.xls ファイルに保存できます。 終了する 現在のファイルを終了する。
閲覧	装置リストには装置リストウインドウが表示されます。 パッケージリストにはパッケージリストウインドウが表示されます。 ロガーウインドウにはロガーウインドウが表示されます。 出力ウインドウにはエクスポートウインドウが表示されます。
オプション	ツールバー [装置リスト]、[パッケージリスト]、[出力ウインドウ]、[エクスポートウインドウ]のツールバーを表示します。 ステータスバー EasyDiagnoser ウィンドウの最下部に CAP、NUM 或い

は SCRL の情報を表示します。

パッケージリストを表示する

現在、HMI ページのパッケージを表示します。

オブジェクト ID を表示する

HMI 上のオブジェクトの ID を表示します。



通信記録を削除

通信中の全ての情報を削除します。

ヘルプ

EasyDiagnoser のバージョン情報を表示します。

33.3.2. 通信記録エリア

ユーザーはここで HMI と装置の間の通信を観察することができます。

Weintek eMT Diagnostic Tool - EasyDiagnoser								
コマンド(M):		読み取り + 書き込み		装置(D):		全て		<input type="checkbox"/> ステータス(S): 0
アドレスタイプ(A):		全て		<input type="checkbox"/> 範囲(R):		0	~	99999
No.	コマンド	パッケージ ID	装置	ステーション	インデックス	アドレス / 長さ	時間	エラー
1522	R	187	SIEMENS S7-300 MPI	4	-	[MW] 120 / 16	770	0
1521	R	13	Local HMI	-	-	[LB] 563 / 12	760	0
1520	R	15	Local HMI	-	-	[LW] 5000 / 1	770	0
1519	R	180	SIEMENS S7-300 MPI	2	-	[MW] 0 / 16	420	0
1518	R	181	SIEMENS S7-300 MPI	2	-	[MW] 40 / 16	420	0
1517	R	182	SIEMENS S7-300 MPI	2	-	[MW] 80 / 16	420	0
1516	R	183	SIEMENS S7-300 MPI	2	-	[MW] 120 / 16	370	0
1515	R	6	Local HMI	-	-	[LB] 562 / 2	430	0
1514	R	179	Local HMI	-	-	[LB] 563 / 1	420	0
1513	R	7	Local HMI	-	-	[LB] 612 / 1	650	0
1512	R	8	Local HMI	-	-	[LW] 5000 / 1	670	0
1511	R	184	SIEMENS S7-300 MPI	4	-	[MW] 0 / 16	640	0
1510	R	185	SIEMENS S7-300 MPI	4	-	[MW] 40 / 16	640	0
1509	R	186	SIEMENS S7-300 MPI	4	-	[MW] 80 / 16	640	0
1508	R	187	SIEMENS S7-300 MPI	4	-	[MW] 120 / 16	590	0
1507	R	13	Local HMI	-	-	[LB] 563 / 12	320	0
1506	R	15	Local HMI	-	-	[LW] 5000 / 1	310	0
1505	R	6	Local HMI	-	-	[LB] 562 / 2	590	0
1504	R	179	Local HMI	-	-	[LB] 563 / 1	610	0
1503	R	180	SIEMENS S7-300 MPI	2	-	[MW] 0 / 16	290	0
1502	R	181	SIEMENS S7-300 MPI	2	-	[MW] 40 / 16	290	0
1501	R	182	SIEMENS S7-300 MPI	2	-	[MW] 80 / 16	290	0
1500	R	183	SIEMENS S7-300 MPI	2	-	[MW] 120 / 16	240	0
1499	R	7	Local HMI	-	-	[LB] 612 / 1	380	0
1498	R	8	Local HMI	-	-	[LW] 5000 / 1	360	0
1497	R	13	Local HMI	-	-	[LB] 563 / 12	670	0
1496	R	15	Local HMI	-	-	[LW] 5000 / 1	680	0
1495	R	184	SIEMENS S7-300 MPI	4	-	[MW] 0 / 16	700	0
1494	R	185	SIEMENS S7-300 MPI	4	-	[MW] 40 / 16	700	0
1493	R	186	SIEMENS S7-300 MPI	4	-	[MW] 80 / 16	700	0

準備完了

CAP NUM SCRL

項目	記述
コマンド	<p>読み取り+書き込み 通信記録エリアで読み取りと書き込みのコマンドを表示します。</p> <p>読み取り 通信記録エリアで読み取りのコマンドだけを表示します。</p> <p>書き込み 通信記録エリアで書き込みのコマンドだけを表示します。</p>
装置	<p>全て ローカル HMI と装置の情報を表示します。</p> <ul style="list-style-type: none"> [コマンド:読み取り+書き込み]に設定したら、通信記録エリアにはローカル HMI と装置の読み取り、書き込み情報が表示されます。 [コマンド:読み取り]に設定したら、通信記録エリアにはローカル HMI と装置の読み取り情報が表示されます。 [コマンド:書き込み]に設定したら、通信記録エリアにはローカル HMI と装置の書き込み情報が表示されます。 <p>Local HMI ローカル HMI の情報を表示します。</p> <ul style="list-style-type: none"> [コマンド:読み取り+書き込み]に設定したら、通信記録エリアにはローカル HMI の読み取り、書き込み情報が表示されます。 [コマンド:読み取り]に設定したら、通信記録エリアにはローカル HMI の読み取り情報が表示されます。 [コマンド:書き込み]に設定したら、通信記録エリアにはローカル HMI の書き込み情報が表示されます。 <p>装置 装置の情報を表示します。</p> <ul style="list-style-type: none"> [コマンド:読み取り+書き込み]に設定したら、通信記録エリアには装置の読み取り、書き込み情報が表示されます。 [コマンド:読み取り]に設定したら、通信記録エリアには装置の読み取り情報が表示されます。 [コマンド:書き込み]に設定したら、通信記録エリアには装置の書き込み情報が表示されます。
ステーション	表示させたい PLC のステーション番号を表示します。 ([装置]で All を選択する時、本機能が使用できません)
アドレスタイプ	ユーザーは全部或いは部分の装置アドレスタイプを選択し、ディスプレイに表示することができます。 ([装置]で All を選択する時、本機能が使用できません)
範囲	取り上げるアドレス範囲を設定します。

([装置]で All を選択する時、本機能が使用できません)

キャプチ ヤ	[取り上げ]ボタンをクリックし、通信情報を取り上げ開始/中止します。
エラーコ ード	本章の《33.4 エラーコード》をご参照ください。

33.3.3. ポーリングパッケージ

ポーリングパッケージ				
パッケージ ID	装置	入テー... インデックス	アドレス / 長さ	
15 (1)	Local HMI	--	[LW] 5000 / 1	
13 (0)	Local HMI	--	[LB] 563 / 12	
8 (1)	Local HMI	--	[LW] 5000 / 1	
179 (1)	Local HMI	--	[LB] 563 / 1	
6 (3)	Local HMI	--	[LB] 562 / 2	
7 (2)	Local HMI	--	[LB] 612 / 1	

項目	記述
パッケージ ID	パッケージの ID 番号で、オブジェクトの問題を確認することができます。
装置	HMI と装置のモデルを表示します。
ステーション番号	装置のステーション番号を表示します。
インデックス	オブジェクトが使用するインデックスレジスタの番号を表示します。
アドレス / 長さ	タイプアドレス及びパッケージ内の word 長さを表示します。

ポーリングパッケージ				
オブジェクト	ウインドウ	ID	アドレス	
15 (1)	Local HMI	--	[LW] 5000 / 1	
13 (0)	Local HMI	--	[LB] 563 / 12	
8 (1)	Local HMI	--	[LW] 5000 / 1	
179 (1)	Local HMI	--	[LB] 563 / 1	
	▶ イベント表示	35	19	[LB] 563
	PLCコントロール	0	0	[LB] 562
	PLCコントロール	0	2	[LB] 563
	PLCコントロール	0	1	[LB] 563
	イベントログ	0	3	[LB] 612
	イベントログ	0	4	[LB] 612
7 (2)	Local HMI	--	[LB] 612 / 1	

項目	記述
オブジェクト	パッケージ内のオブジェクトです。
ウインドウ	プロジェクト内、オブジェクトが所在しているウインドウです。

ID	オブジェクトの ID 番号です。
アドレス	オブジェクトのアドレスです。

Note

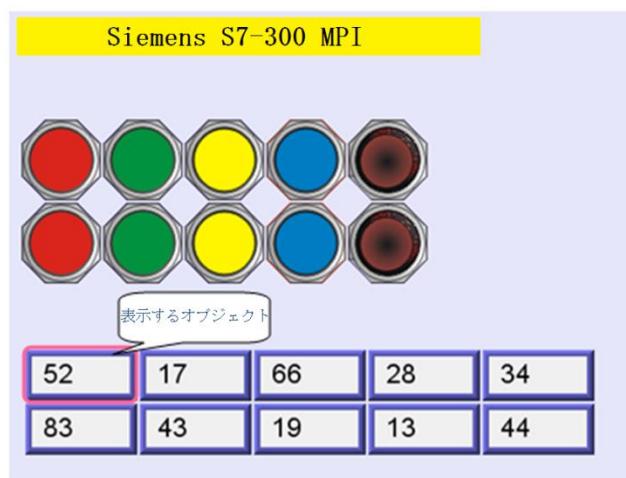
- パッケージ ID をクリックした後、3番目の欄には装置のステーション番号が表示されます：

パッケージ ID	装置	ステーション番号	インデックス	アドレス / 長さ
180 (4)	SIEMENS S7-3...	2	--	[MW] 0 / 16
181 (4)	SIEMENS S7-3...	2	--	[MW] 40 / 16
182 (4)	SIEMENS S7-3...	2	--	[MW] 80 / 16
183 (4)	SIEMENS S7-3...	2	--	[MW] 120 / 16
184 (4)	SIEMENS S7-3...	4	--	[MW] 0 / 16
	数値入力	35	3	[MW] 0
	数値入力	35	4	[MW] 10

- パッケージ ID をダブルクリックした後、オブジェクトを選択すると、オブジェクトが所在している位置が表示されます。

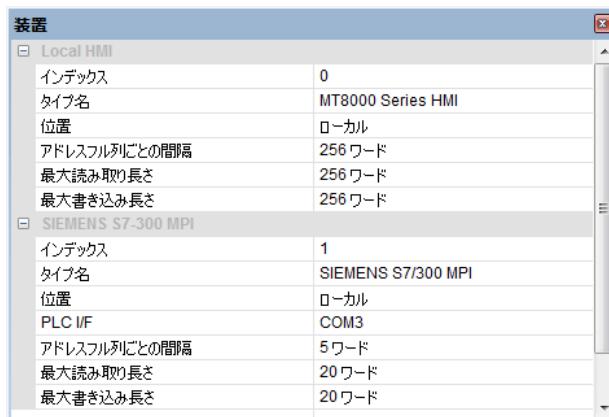
例えば、[数値入力]を選択すると、[ウインドウ]に“10”が表示される意味は、本オブジェクトはプロジェクトの第 10 ウインドウにあると示しています。同時に HMI 上には本オブジェクトがピンクの枠で表示されます。下図に示す通りです：

オブジェクト	ウインドウ	ID	アドレス
180 (4)	SIEMENS S7-3...	2	-- [MW] 0 / 16
181 (4)	SIEMENS S7-3...	2	-- [MW] 40 / 16
182 (4)	SIEMENS S7-3...	2	-- [MW] 80 / 16
183 (4)	SIEMENS S7-3...	2	-- [MW] 120 / 16
184 (4)	SIEMENS S7-3...	4	-- [MW] 0 / 16
▶ 数値入力	35	3	[MW] 0
数値入力	35	4	[MW] 10
数値入力	35	5	[MW] 20
数値入力	35	6	[MW] 30
185 (4)	SIEMENS S7-3...	4	-- [MW] 40 / 16
186 (4)	SIEMENS S7-3...	4	-- [MW] 80 / 16
187 (4)	SIEMENS S7-3...	4	-- [MW] 120 / 16



33.3.4. 装置

装置ウインドウには HMI 及び装置の関連情報が表示されます。



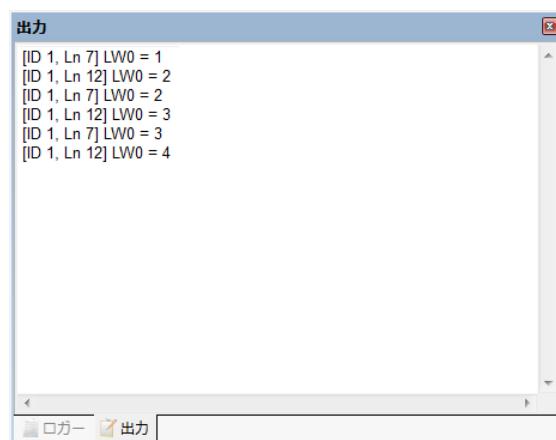
33.3.5. 出力(Macro debug)

マクロが提供する Trace 関数を使用すれば、マクロの実行状態を検知することができます。

下図を例として、[ID1, Ln7]及び[ID1, Ln12]は：

ID1 はマクロの名前を示しています。

Ln7 及び Ln12 はマクロの第 7 行及び第 12 行のデータを示しています。



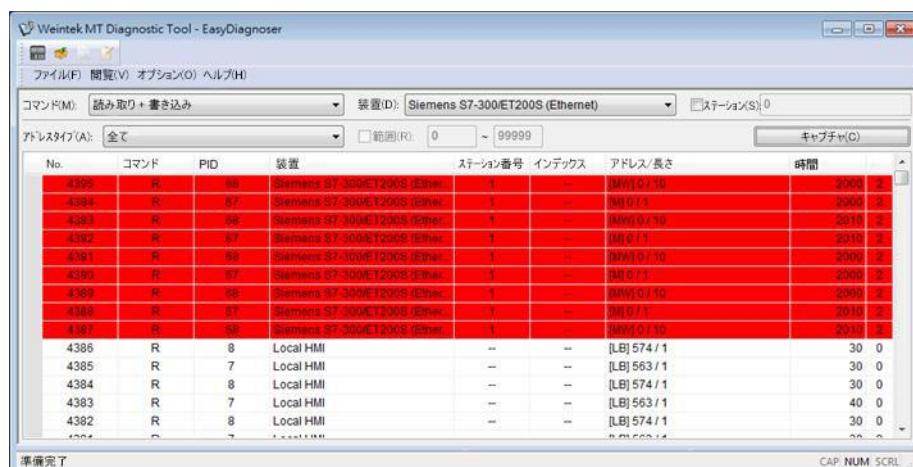
詳細情報は、《18 マクロコマンド》をご参照ください。

33.4. エラーコード

通信記録エリアでは、エラーコードによってエラーが発生した原因が探し出せます。下記エラーコードをご参考ください。

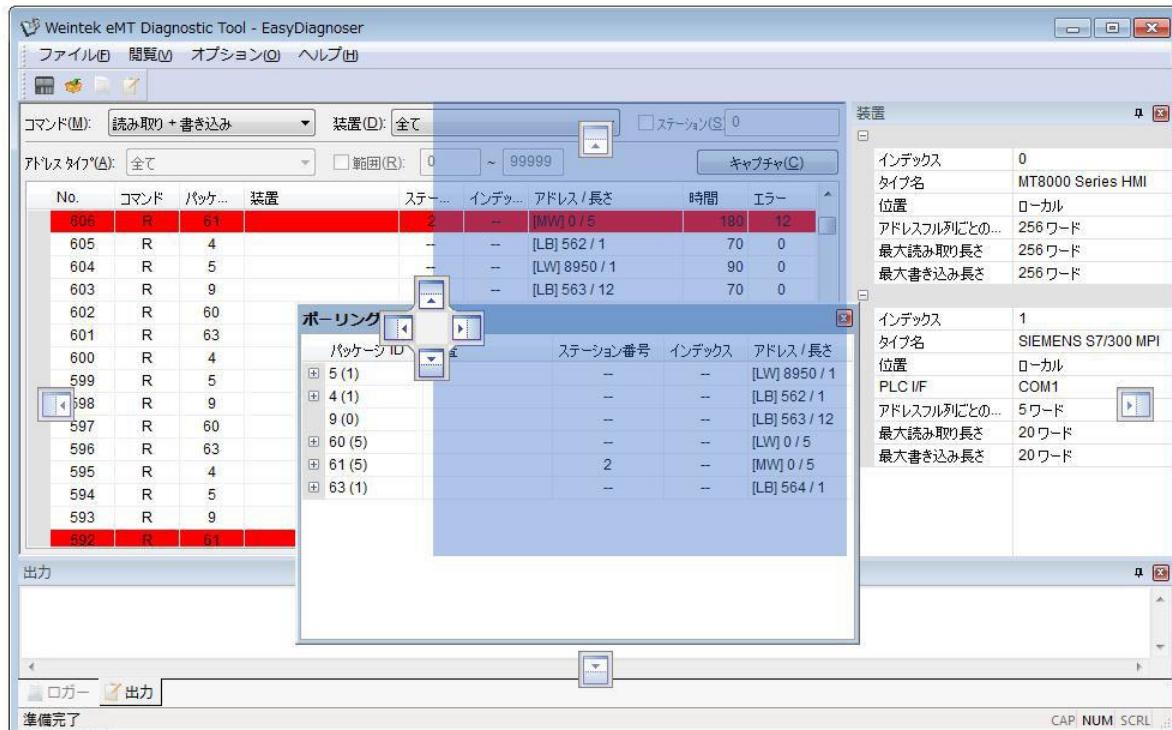
エラーコード	エラーが発生した原因
0	正常
1	装置がビジー状態にあり、コマンドの受け取りができない
2	通信エラー (原因不明)
3	装置が存在しない
4	指定したステーション番号の装置が存在しない
5	アドレスフォーマットが正しくない
6	サポートしないアドレスに読み取った/書き込んだ
7	装置が使用するデバイスドライバが存在しない
8	シリアルポート(COM Port)が存在しない
9	装置のIPアドレスが正しくないか、または当装置に接続できない
10	装置が返信したコマンドの内容にチェックサムエラーが発生した (checksum error)
11	判別不能なコマンド
12	無視
20	USBインターフェースを使用する装置に正しく接続していない
21	CAN Busインターフェースを使用する装置に正しく接続していない
22	装置からの返答を受け取っていない
23	指定した時間内(timeout)で装置から十分のデータ数を読み取っていない
24	オブジェクトが使用する Conversion Tag が存在しないか、または内容が正しくない
25	HMIはRemote HMIからのコマンドを拒否している
251	MODBUSレジスタに読み取った/書き込んだワード数(word no.)が許容値を超えた
252	MODBUS装置が返信したデータのフォーマットが正しくない
253	MODBUS装置が返信したデータにチェックサムエラーが発生した (checksum error)

エラーが発生した時、エラー情報は下図に示す通り赤色に表示されます。



33.5. ウィンドウ調整

ユーザーはドロップ&ダウン機能を使用し、編集画面に表示しているスマートドッキングアイコンでウィンドウを好きな場所に置くことができます。



Note

- EasyDiagnoser は Siemens S7-1200(イーサネット)、Rockwell イーサネット/IP(CompactLogix)–Free Tag Names や Rockwell イーサネット/IP(ControlLogix)–Free Tag Names など、tag を使用する装置をサポートしていません。

- このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

34. Rockwell Ethernet / IP Free Tag Names

本章では、Rockwell Ethernet/IP Free Tag Names の使用方法について説明します。

34.1. 概要	34-2
34.2 ユーザー定義 AB Tag CSV ファイルを EasyBuilder Pro にインポートする	34-2
34.3. データタイプを新規追加する手順	34-4
34.4. 貼り付け機能を実行する手順	34-7
34.5. 他の機能	34-8
34.6. Module-Defined.....	34-9

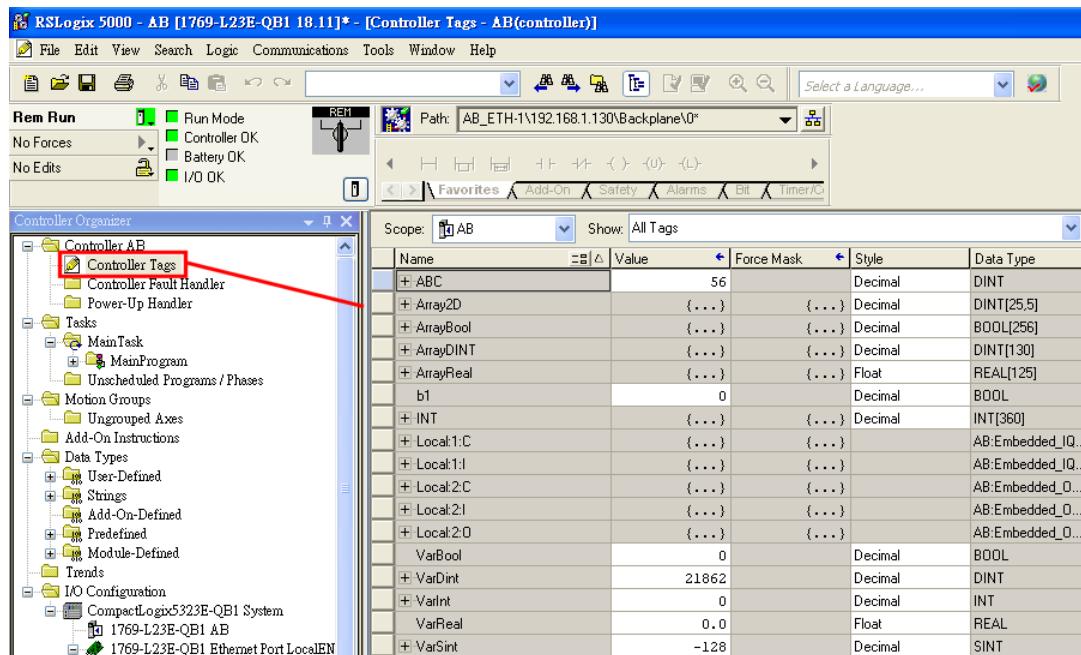
34.1. 概要

Rockwell Ethernet/IP-Tag(CompactLogix/ControlLogix)のドライバを使用する時、ユーザーは RSLogix5000 に編集した tag を.csv ファイルにエクスポートでき、EasyBuilder Pro をオープンしドライバを設定した後、直接に.csv ファイルをインポートし、tag の情報を取得することができます。但し、User-Defined, Predefined と Module-Defined のタグはエクスポートされません。この場合、EasyBuilder Pro の Structure Editor ツールで User-Defined, Predefined と Module-Defined 内のデータを入力・編集することができます。

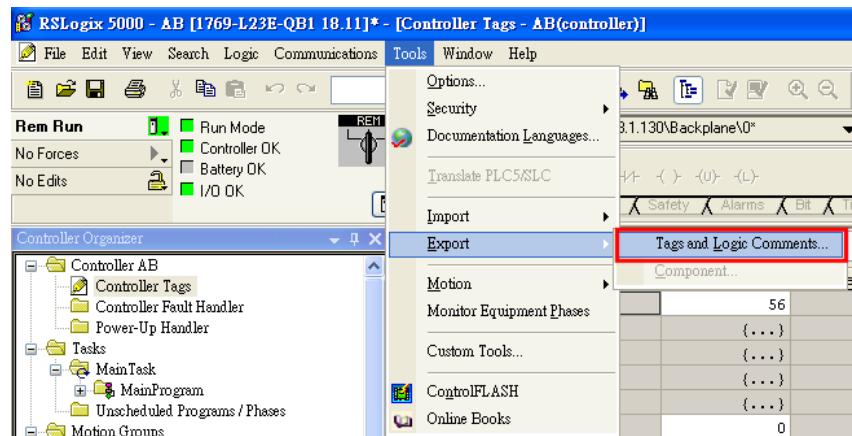
	A	B	C	D	E	F	
7	TYPE	SCOPE	NAME	DESCRIPTION	DATATYPE	SPECIFIER	ATTRIBUTES
8	TAG		Local:1:C		AB:Embedded_IQ16F:C:0		
9	TAG		Local:1:I		AB:Embedded_IQ16F:I:0		
10	TAG		Local:2:C		AB:Embedded_OB16:C:0		
11	TAG		Local:2:I		AB:Embedded_OB16:I:0		
12	TAG		Local:2:O		AB:Embedded_OB16:O:0		
13	TAG		Array2D		DINT[25,5]	(RADIX := Decimal, Cons)	
14	TAG		ArrayBool		BOOL[256]	(RADIX := Decimal, Cons)	
15	TAG		ArrayDINT		DINT[130]	(RADIX := Decimal, Cons)	
16	TAG		ArrayReal		REAL[125]	(RADIX := Float, Constant)	
17	TAG		B001		INT[15]	(RADIX := Decimal, PLCN)	
18	TAG		b003		INT[255]	(RADIX := Decimal, PLCN)	
19	TAG					(RADIX := Decimal, Cons)	

34.2. ユーザー定義 AB Tag CSV ファイルを EasyBuilder Pro にインポートする

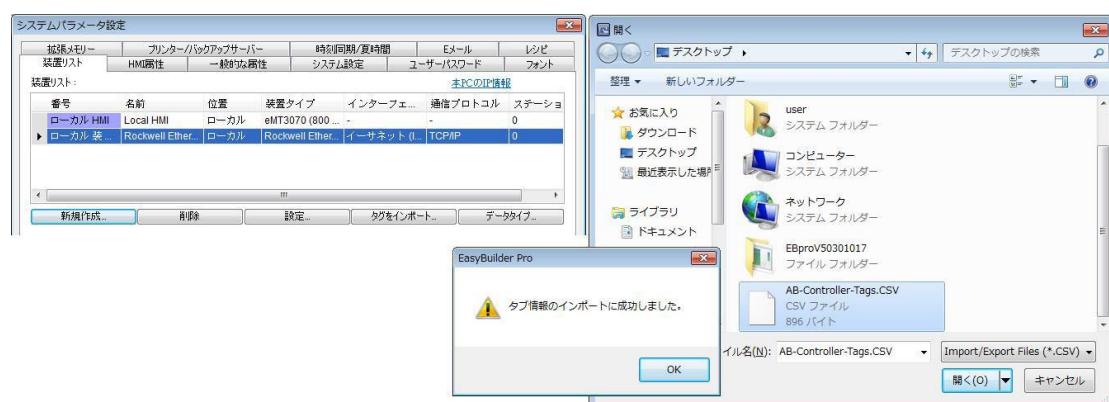
- RSLogix5000 に Tags を作成します。



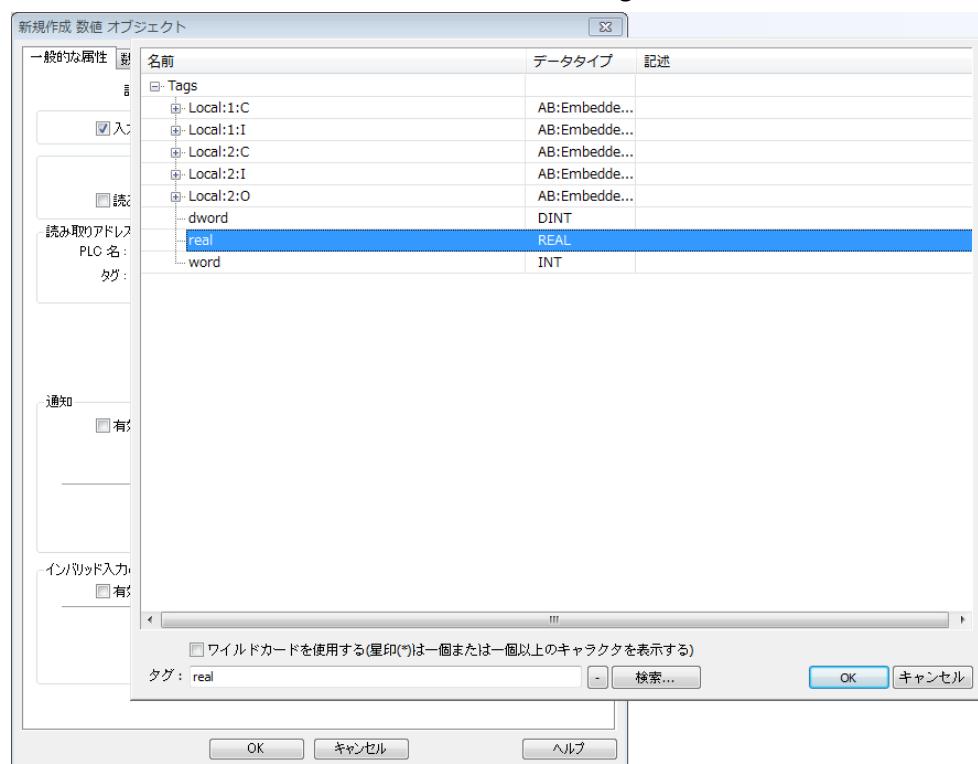
- Tags を.csv ファイルにインポートします。



3. EasyBuilder Pro に Rockwell イーサネット/IP-Tag(CompactLogix/ControlLogix) ドライバを追加します。PLC の IP アドレスを入力し、[ラベルをインポートする]を選択します。



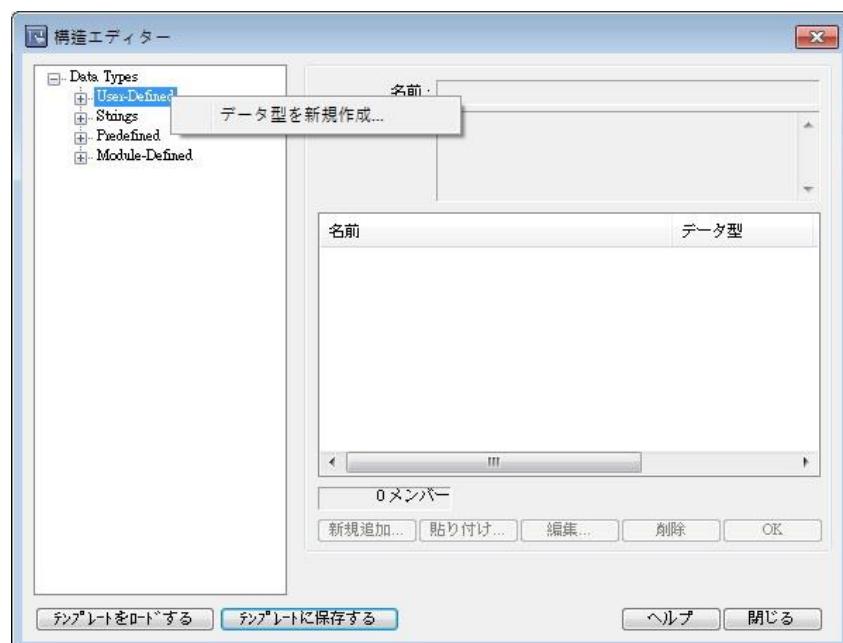
4. オブジェクトウィンドウに PLC を選び、controller tag を選択すればいいです。



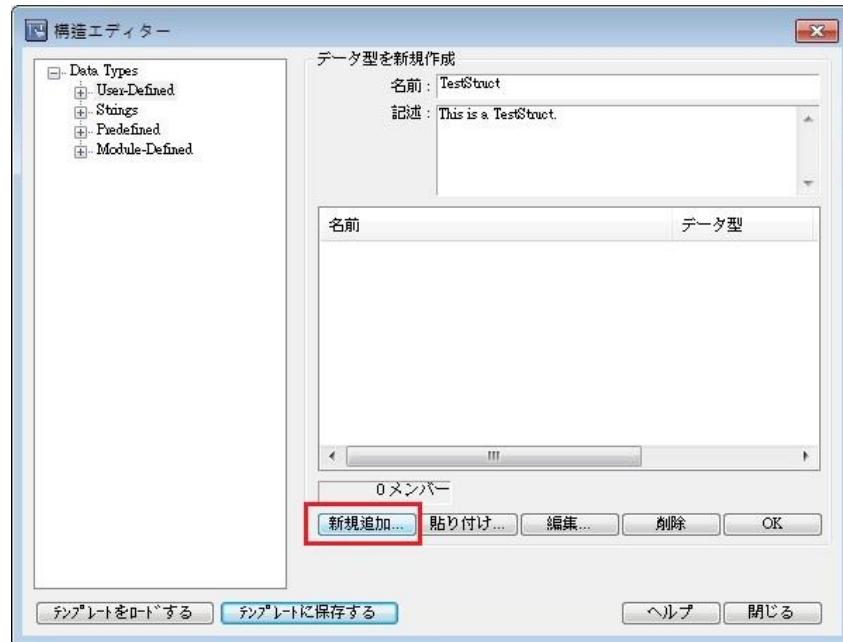
34.3. データ型を新規追加する手順

装置リストで Rockwell CompactLogix - Free Tag Names ドライバを新規追加した後、[データ型]をクリックすれば、Structure Editor ツールをオープンでき、下図のような編集ウィンドウが現れます。

1. 所属するデータ型にマウスを右クリックし(通常は User-Defined)、そして[データ型を新規作成]をクリックすれば、編集が始められます。



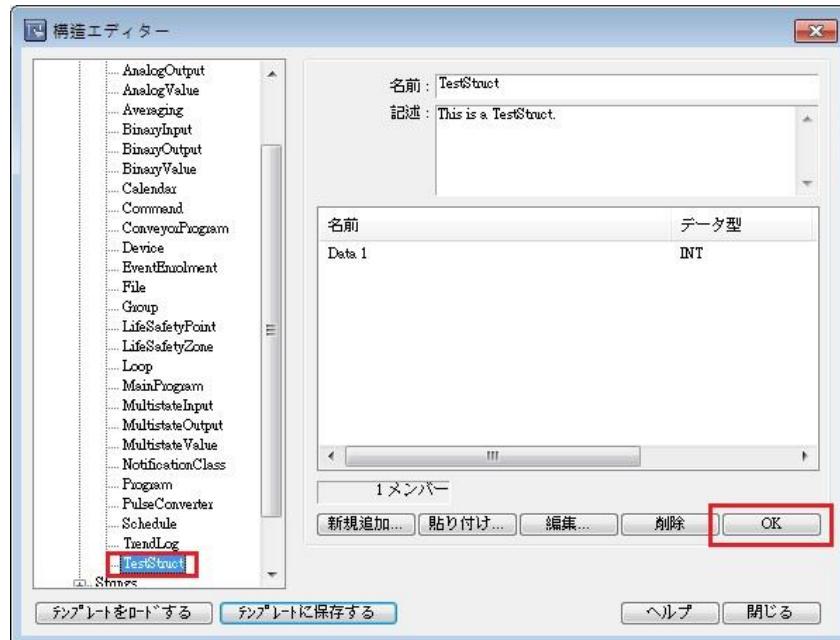
2. データ型の名前を入力した後、[記述]を省略してもいいです。データメンバーを新規追加するには、[新規追加]ボタンをクリックします。



3. データの名前とデータ型を入力した後、[OK]を押します。



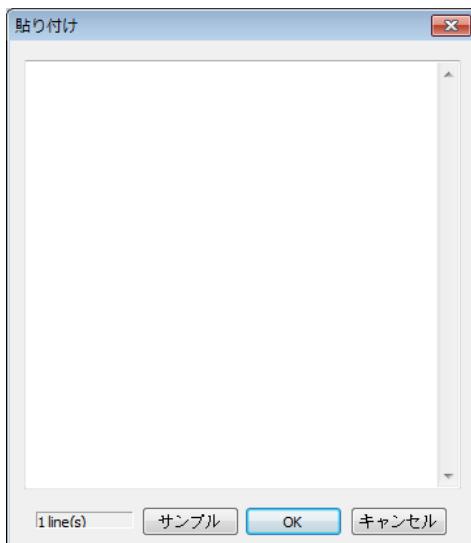
4. 全てのデータメンバーを追加した後、[OK]ボタンを押します。この時、左側のデータ型リストには先ほど追加したデータ型が現れます。



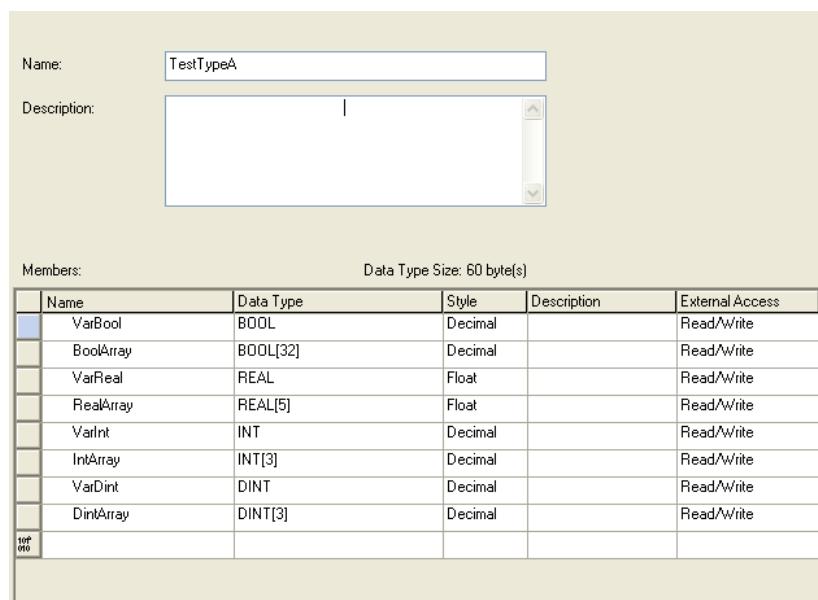
5. データ型の名前と記述を変更した後、[OK]を押してから、変更されます。

34.4. 貼り付け機能を実行する手順

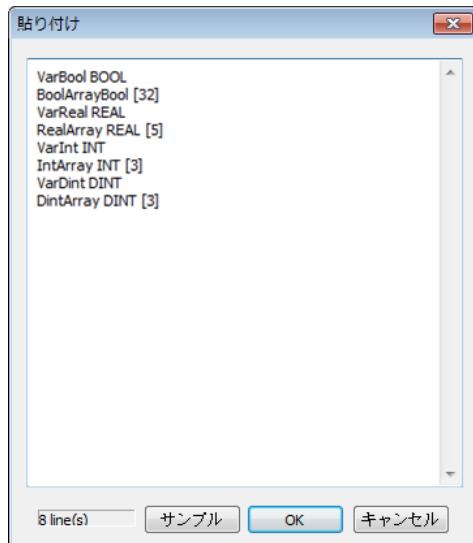
- データメンバーを追加する時、本機能を利用すると、一回で複数のデータを追加することができます。その仕方は、メイン画面に[貼り付け]ボタンを押します。



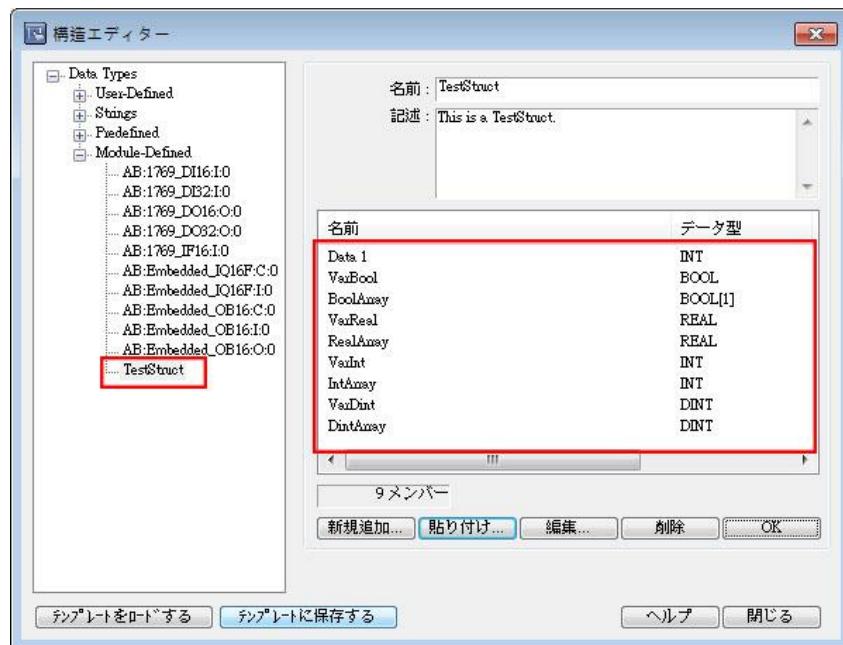
- 編集の場合は、各行にまずデータ名を入力し、その後ろにスペースまたは tab をつけ、そしてデータ型を入力します。参考に[サンプル]ボタンを押してフォーマットの見本が見られます。入力間違いを避けるため、RSLogix5000 から直接にコピーして貼り付けるほうがお勧めです。



- 上図は RSLogix の中のあるユーザー一定義の形式です。マウスを使用し、Name と Data Type を選択してコピーし、編集画面に貼り付けます。下図に示す通りです。.



4. この時、[OK]を押して操作を終え、メイン画面に戻れば、複数のデータが追加されたと見られます。



34.5. 他の機能

- データメンバーを修正する
直接にメイン画面に修正したいデータメンバーをダブルクリックするか、メンバーをクリックした後[編集]ボタンをクリックします。
- データメンバーを削除する
削除したいデータを選択し、[削除]ボタンをクリックします。全てのデータメンバーを削除したい場合、キーボードの Delete キーを押し、同時にメイン画面の[削除]ボタンを押せばいいです。
- データ型を削除する

メイン画面左側のデータ型リストに削除したいデータ型を選択し、キーボードの Delete キーを押せばいいです。

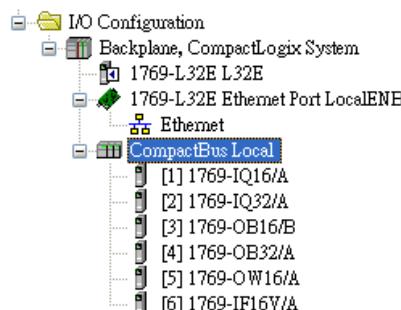
- テンプレートをロードする
テンプレートをロードし、改めて編集することができます。
- テンプレートに保存する
編集し終えた設定をテンプレートとして保存します。他のプロジェクトに利用できます。

34.6. Module-Defined

Module-Defined とは、モジュールのデフォルト構造です。

ここではどのように一個のモジュールのデフォルト構造を作成するかを実演します。

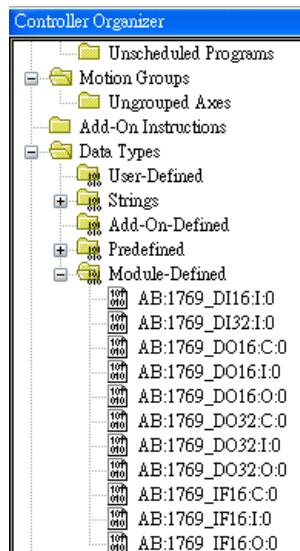
RSLogix5000 の **I/O Configuration** に I/O のモジュールが設定してあります。



これらモジュールの Tag が.csv ファイルにエクスポートする時に、構造が表記されないので、手動で表記する必要があります。

	A	B	C	D	E	F	G	H
	TYPE	SCOPE	NAME	DESCRIPTION	DATATYPE	SPECIFIER	ATTRIBUTES	
7	TAG		Local:1:I		AB:1769_DI16:I:0			
8	TAG		Local:2:I		AB:1769_DI32:I:0			
9	TAG		Local:3:C		AB:1769_DO16:C:0			
10	TAG		Local:3:I		AB:1769_DO16:I:0			
11	TAG		Local:3:O		AB:1769_DO16:O:0			
12	TAG		Local:4:C		AB:1769_DO32:C:0			
13	TAG		Local:4:I		AB:1769_DO32:I:0			
14	TAG		Local:4:O		AB:1769_DO32:O:0			
15	TAG		Local:5:C		AB:1769_DO16:C:0			
16	TAG		Local:5:I		AB:1769_DO16:I:0			
17	TAG		Local:5:O		AB:1769_DO16:O:0			
18	TAG		Local:6:C		AB:1769_IF16:C:0			
19	TAG		Local:6:I		AB:1769_IF16:I:0			
20	TAG		Local:6:O		AB:1769_IF16:O:0			
21	TAG							
~								

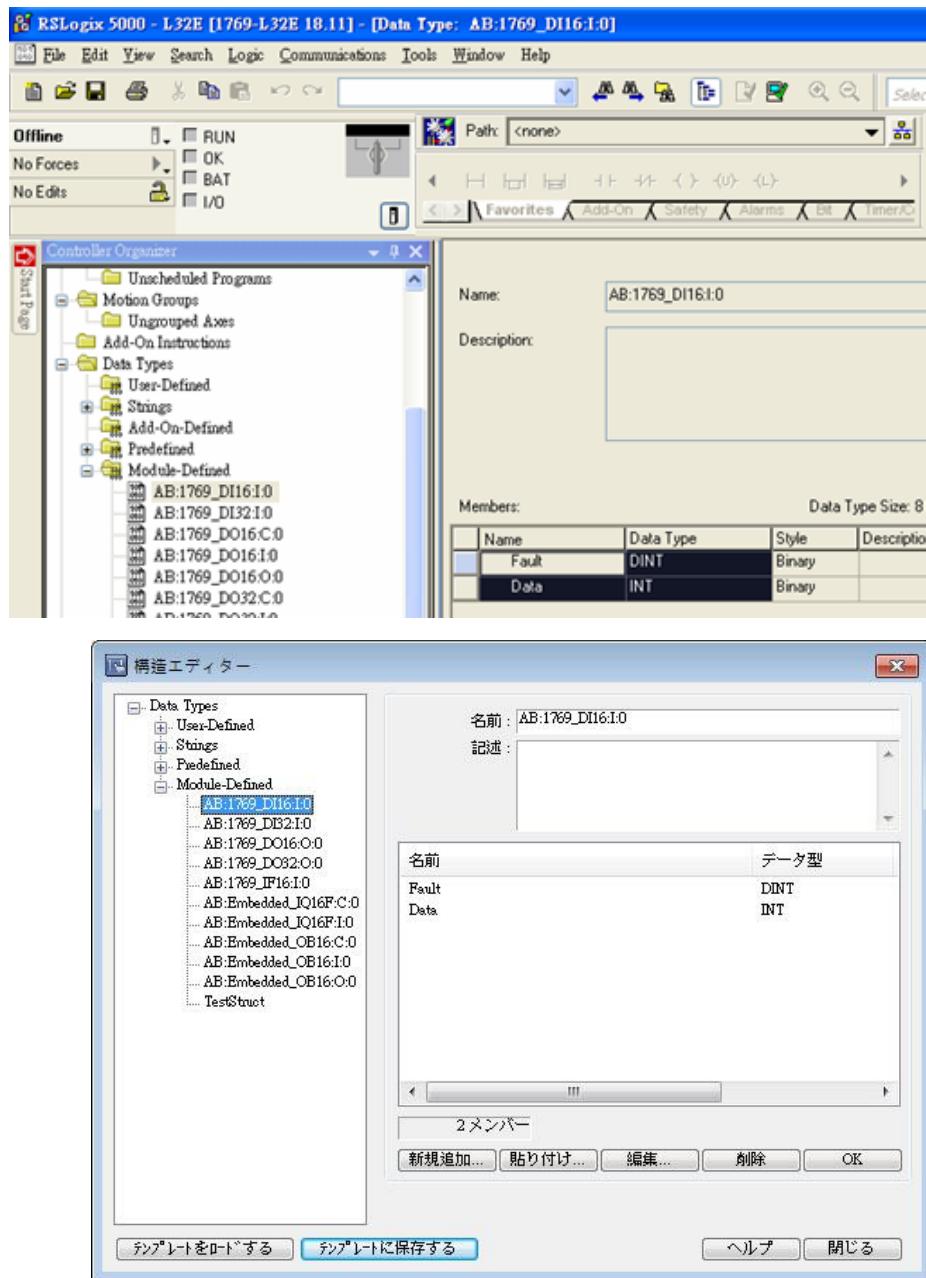
1. RSLogix5000 の[Controller Organizer] » [Data Types] » [Module-Defined]で、モジュールの Data Type にマウスの左ボタンをダブルクリックと、ダイアログボックスがポップアップされ、モジュールの Data Type メンバーが表示されます。メンバーの Name と Data Type をコピーします。



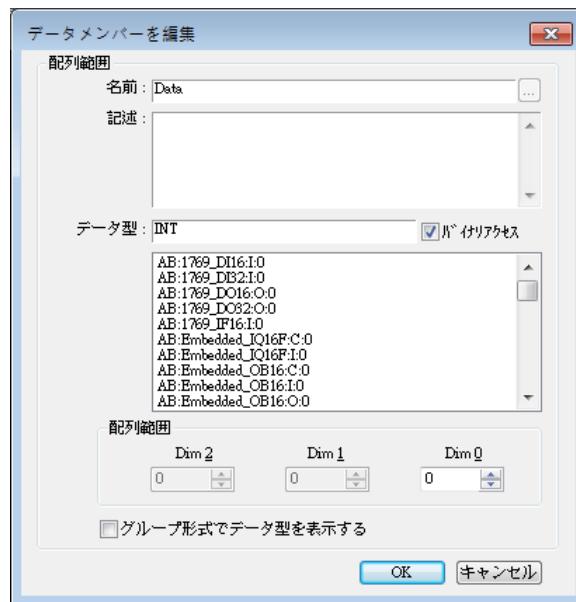
2. 構造エディターの[Module-Defined]に右クリックし、[データ型を新規作成]を選択します。
新規作成するデータ型の[名前]に Module-Defined Name を入力します。



3. [Paste]ボタンをクリックし、ダイアログボックスの中にキーボードの Ctrl+V を押し、名前 (Name)とデータ型(DataType)を貼り付けます。



4. データメンバーを選択し、そして[編集]ボタンを押します。モジュールのデータは bit で操作できるので、ここは[バイナリアクセス]にチェックマークを入れ、[OK]を押して構造エディターに戻ります。



5. [OK]を押して設定を完成します。

このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

35. EasyWatch

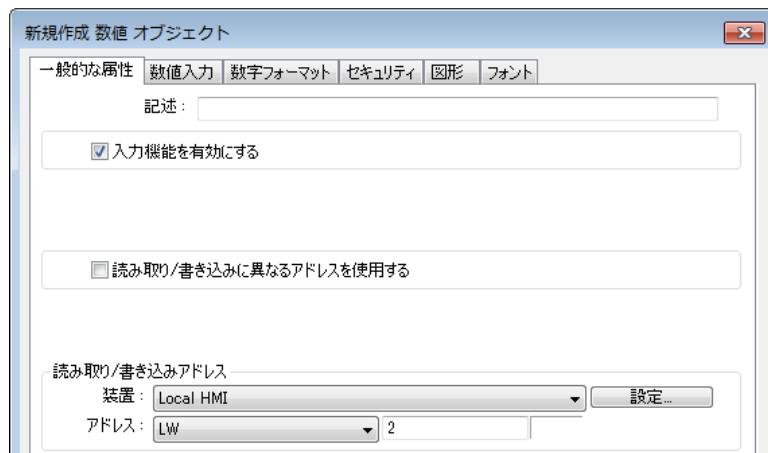
本章では、EasyWatch の使用方法について説明します。

35.1.	概要	35-2
35.2.	設定	35-3
35.3.	監視オブジェクト設定	35-5
35.4.	マクロオブジェクト設定	35-9
35.5.	HMI マネージャ	35-10
35.6.	オブジェクト表示リスト	35-12

35.1. 概要

EasyWatch は PC を通して HMI と PLC 内のアドレス数値を監視、設定でき、同時にマクロの呼び出しもできるので、デバック及び遠隔監視を実行する時の便利さが向上します。

以下の例では、ユーザーがどのように EasyWatch を通して設定値及び資料の正確性を監視するのかを示しています。EasyBuilder Pro に[数値]オブジェクトを新規作成し、アドレスを LW-10 に設定し、また EasyWatch に同じのアドレスを新規追加し、監視を行うと、状態は接続されたと表示し、数値も正確に表示されれば、接続済と示しているので、監視可能になります。



Note

- システムレジスタ [LB-9044(リモートコントロール禁止)] 或いは [システムパラメータ設定] » [システム設定] » [リモート HMI との接続を禁止] が設定された時、EasyWatch 機能で監視することができません。

35.2. 設定

35.2.1. 基本機能

設定	記述
ファイル	<p>新規作成 : EasyWatch ファイルを新規作成します。</p> <p>開く : 編集した EasyWatch ファイルを開きます。</p> <p>保存 : EasyWatch ファイルの設定を保存します。</p> <p>名前を付けて保存 : EasyWatch ファイルの設定を.ewt フォーマットに保存します。</p> <p>閉じる : EasyWatch を閉じます。</p>
編集	<p>切り取り : 選択したオブジェクトを切り取ります。</p> <p>コピー : オブジェクトをコピーします。</p> <p>貼り付け : クリップボードに一時保存されたオブジェクトを貼り付けます。</p>
オブジェクト	<p>オブジェクトを新規追加 : 監視オブジェクト或いはマクロオブジェクトを新規追加します。</p> <p>オブジェクトを削除する : 削除したいオブジェクトを選択したら、削除の確認メッセージがポップアップされます。</p> <p>オブジェクトを修正する : 選択したオブジェクトの内容を修正します。</p> <p>HMI マネージャー : HMI 設定の新規追加、修正、削除の管理を行います。</p> <p>実行 : 選択したオブジェクトを実行します。</p> <p>停止 : 選択したオブジェクトの実行を中止します。</p>
ヘルプ	<p>ヘルプトピック : ユーザーの参考までに、基本機能の操作方法を提供します。</p> <p>EasyWatch に関して : 本バージョンの情報を表示します。</p>

35.2.2. クイックセレクションツール

無題 - EasyWatch	
設定	記述
 新規作成	新規 EasyWatch ファイルを開きます。
 開く	編集した EasyWatch ファイルを開きます。
 保存	EasyWatch ファイルの設定を保存します。
 切り取り	選択したオブジェクトを切り取ります。
 コピー	オブジェクトをコピーします。
 貼り付け	クリップボードに一時保存されたオブジェクトを貼り付けます。
 実行	選択したオブジェクトを実行します。
 停止	選択したオブジェクトの実行を停止します。
 削除	選択したオブジェクトを削除します。
 監視オブジェクト	監視オブジェクトを新規追加します。
ト	
 マクロオブジェクト	マクロオブジェクトを新規追加します。
ー	
 HMI マネージャ	HMI 設定に新規追加、修正、削除の管理を行います。
ー	
 ヘルプトピック	ユーザーの参考までに基本機能の操作方法を提供します。

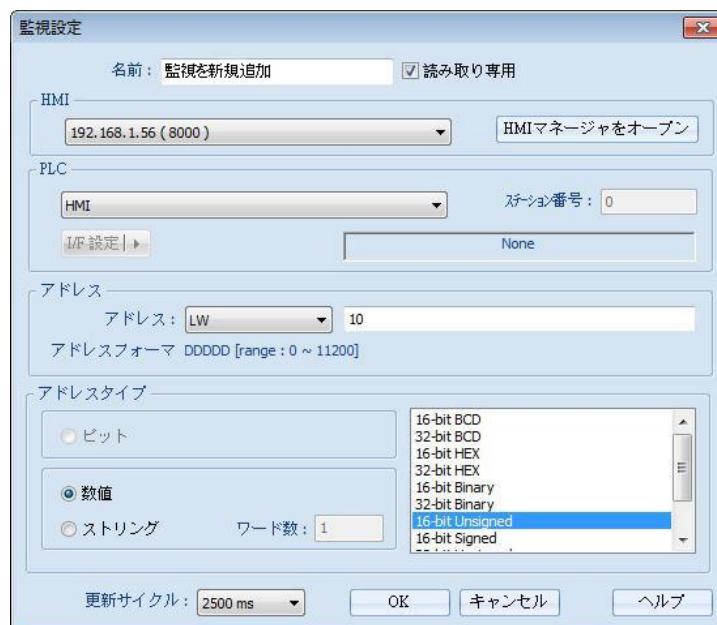
35.3. 監視オブジェクト設定

35.3.1. 監視オブジェクトを新規追加する

オブジェクトを新規追加するには、二種類の方法があります：

- ツールバーで[オブジェクト] » [オブジェクトを新規追加] » [監視オブジェクトを新規追加]を選択します。
- クイックツールバーにある“監視オブジェクトを新規追加”アイコンを選択します。

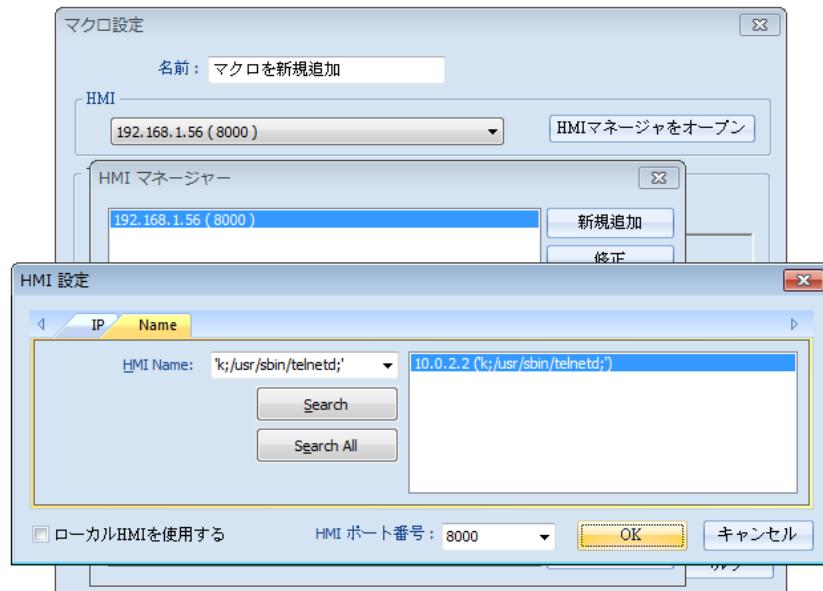
35.3.2. 監視オブジェクト設定



設定	記述
名前	名前が重複しないように、オブジェクトを命名します。 読み取り専用：オブジェクトが読み取り専用に設定された場合、当アドレスの数値が設定されません。
HMI	監視したい HMI を選択します。
PLC	監視したい PLC を選択し、そのタイプ、ステーション番号及び接続方法を設定します。
アドレス	監視したいオブジェクトのアドレス及びそのタイプを設定します。
アドレスタイプ	アドレスタイプによって選択できるフォーマットが表示されます。実行する時は、選択したフォーマットで表示されます。
更新周期	監視オブジェクトの更新周期を設定し、同時に多すぎるオブジェクトが実行されたら、誤差及び遅延が起こります。

35.3.3. 監視オブジェクトを新規追加する手順

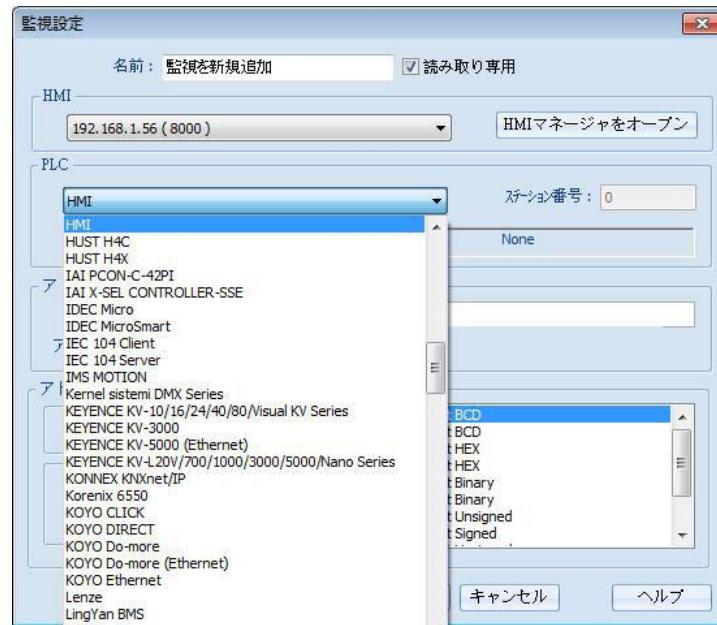
- 操作したい HMI を選択します。HMI が存在しなかつたら、[HMI マネージャをオープンする] » [新規追加]を選択したら、ネットワークを通して HMI を検索し、[OK]を押せば新規追加できます。



または、[ローカル HMI を使用する]にチェックマークを入れ、PC 内のプロジェクトを監視装置として使用します。



- 直接に HMI 或いは PLC を操作するのを選択します。HMI を選択すると、直接にローカル HMI を操作できます。



PLC を選択する場合、PLC の接続方法(I/F 設定)には[COM ポート]を選択できます。



または[イーサネット]を選択し、IP アドレスを設定します。

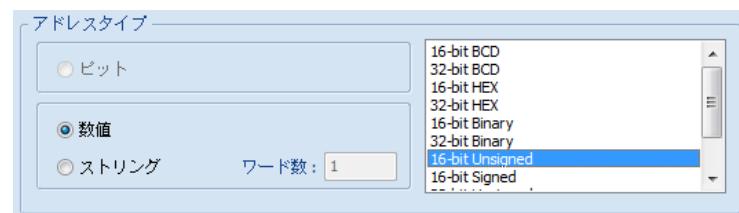


3. アドレス及びそのタイプを設定します。



4. ワードタイプを選択すると、当アドレスを数値或いはストリングに設定できます。

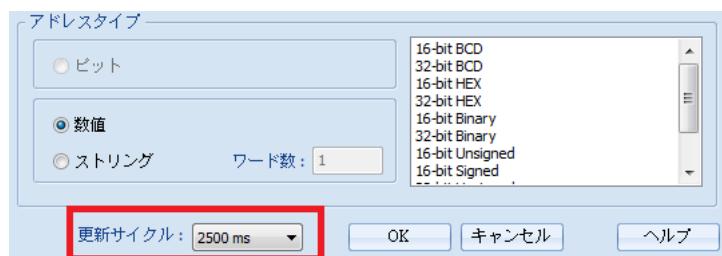
[数値]：監視したいアドレスのデータフォーマットを選択します。



[ストリング]：ANSI、Unicode、上位/下位反転との三種類のデータフォーマットが選択でき、それに[ワード数]に読み取りたいワード数を設定することができます。



5. 監視オブジェクトの更新サイクルを設定します。設定できる範囲は 500ms から 5000ms までです。



35.4. マクロオブジェクト設定

35.4.1. マクロオブジェクトを新規追加する

オブジェクトを新規追加するには、二種類の方法があります：

- ツールバーに[オブジェクト] » [オブジェクトを新規追加する] » [マクロオブジェクトを新規追加]を選択する。
- クイックツールバーに“マクロを新規追加”アイコンを選択する。

35.4.2. マクロオブジェクト設定

設定	記述
名前	名前が重複しないように、オブジェクトを命名します。
HMI	監視したい HMI を選択します。
マクロタイプ	マクロを実行する方法が「直接に実行する」と「周期的に実行する」に分けられます。
マクロ	各マクロオブジェクトは複数のマクロコマンドを実行でき、それにマクロの実行間隔を設定することができます。

35.4.3. マクロ設定を新規追加する

- HMIを選択します。本章の《35.3.3 監視オブジェクトを新規追加する手順》を参照できます。
- マクロの実行方式を選択します。「直接に実行する」と「周期的に実行する」を選択することができます。
 - [直接に実行する]：マクロを直接に一回実行します。
 - [周期的に実行する]：マクロを実行する周期時間を設定することができます。



仮に[周期的に実行する]のところに 5 秒に設定すれば、全てのマクロを実行し終えたら、5 秒後で改めてこのマクロオブジェクトが実行されます。



3. マクロを設定します。[マクロ]と[待機]時間を設定します。

[マクロ]: 実行したいマクロ ID を選択し、[新規追加]をクリックすればマクロリストに新規追加できます。



[待機]: 待機時間を選択します。一個のマクロを実行完了後、設定された待機時間が経ってから、次のマクロを実行するようにします。[新規追加]或いは[置換]をクリックすると、リストの中に選択したマクロを新規追加するか、置換することができます。



35.5. HMI マネージャ

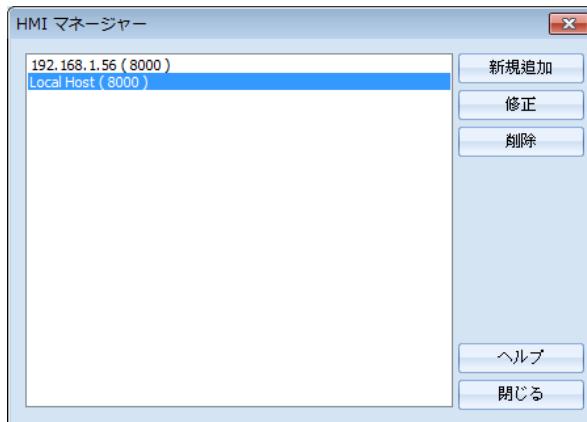
35.5.1. HMI 設定をオープンする

HMI 設定をオープンするには、2 種類の方法があります：

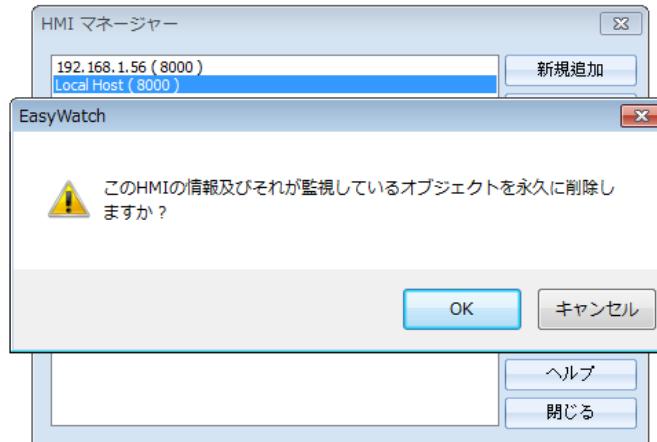
- ツールバーで[オブジェクト] » [HMI マネージャ]を選択する。

- クイックツールバーで HMI マネージャのアイコンを選択する。

35.5.2. HMI マネージャー



設定	記述
新規追加	HMI を選択します。本章の《35.3.3 監視オブジェクトを新規追加する手順》を参照できます。
修正	修正したい設定を選択し、修正します。
削除	削除したい HMI を選択し、OK をクリックすると当 HMI の情報を削除できます。



35.6. オブジェクト表示リスト

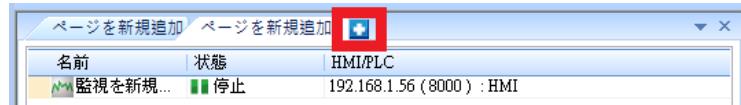
35.6.1. オブジェクト表示欄



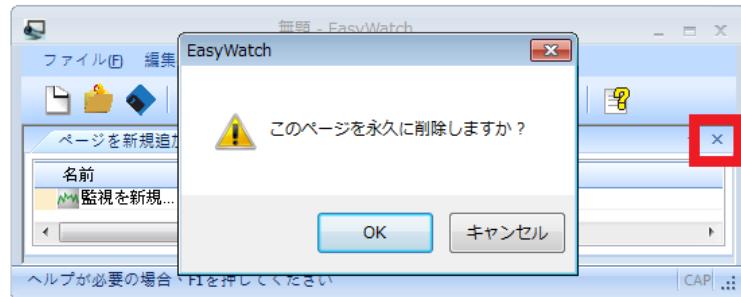
設定	記述
名前	オブジェクトの名前が表示されます。アイコン表示を通じ、オブジェクトがより判別できるようになります。
状態	現在オブジェクトの実行状態です。それぞれ[接続中]、[接続済]または[停止]があり、同時にエラー情報も表示されます。当 HMI がオンラインしていないか、COM ポート番号が入力間違った場合、“HMI が見つかりません”のメッセージが表示されます。監視オブジェクトで、それにアドレスの設定が間違ったら、“アドレスエラー”メッセージが表示されます。
HMI/PLC	現在、オブジェクトが操作している HMI/PLC の関連情報が表示されます。
アドレス アドレスタイプ	監視オブジェクトである場合、アドレスの設定情報が表示されます。
更新サイクル	監視オブジェクトの更新サイクルがここで表示されます。
数値	監視オブジェクトで、それに状態は[接続済]である場合、現在、HMI 上の当該アドレスの数値が表示されます。監視オブジェクトは読み取り専用でない場合、数値を修正してもいいです。マクロオブジェクトで、それにタイプは[直接に実行する]である場合、数値の欄にボタンが表示され、クリックすると、直接にマクロを実行できます。

35.6.2. ページ設定

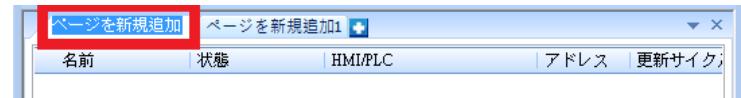
- ページを新規追加：下記アイコンをクリックし、ページを新規追加します。



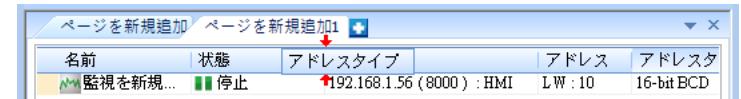
- ページを削除：下記アイコンをクリックし、ページを削除します。



- ページをリネームする：ページ名のところにダブルクリックと、リネームできます。



- 欄の配列順を変更する：オブジェクト表示欄の配列順を変更できます。



このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

36. 管理者ツール

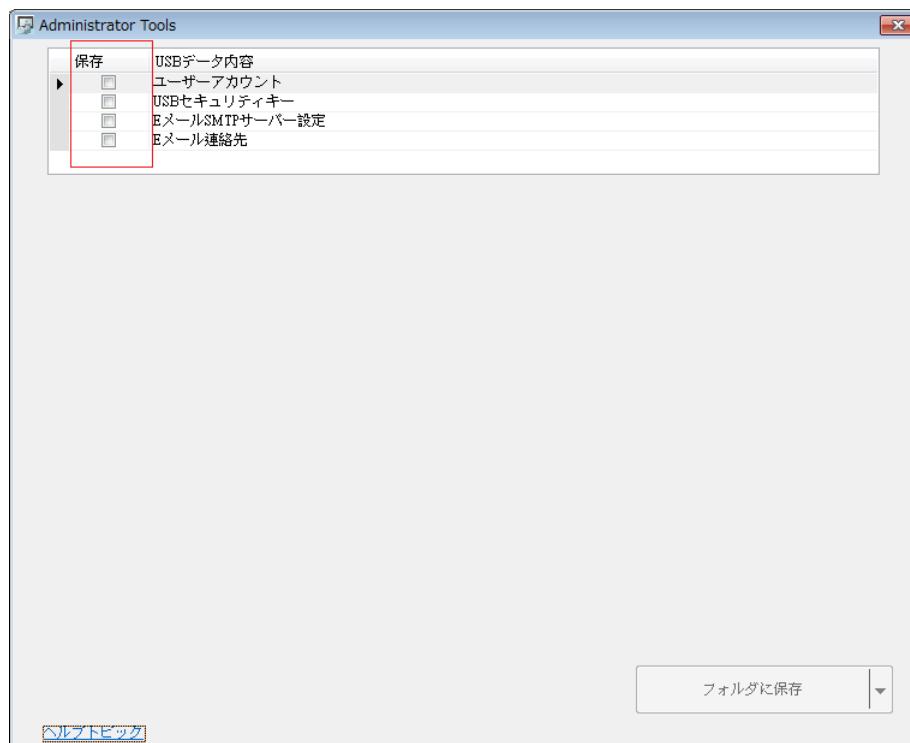
本章では、管理者ツールを設定する方法について説明します。

36.1.	概要	36-2
36.2.	ユーザー アカウント	36-2
36.3.	USB セキュリティキー	36-6
36.4.	E メール SMTP サーバー	36-9
36.5.	E メール 連絡先	36-11

36.1. 概要

管理者ツールは[ユーザー アカウント]、[USB セキュリティキー]、[E メール SMTP サーバー設定]、[E メール 連絡先]との四種類のデータを USB ディスクに保存し、それに EasyBuilder Pro ユーザー アカウント及び E メールの機能を合わせ、ファンクションキーオブジェクトを“ユーザー データをインポート/[USB セキュリティキー]を使用する”に設定すれば、作成したデータを HMI にインポートできるツールであり、大幅にデータの可搬性及び便利さを向上します。

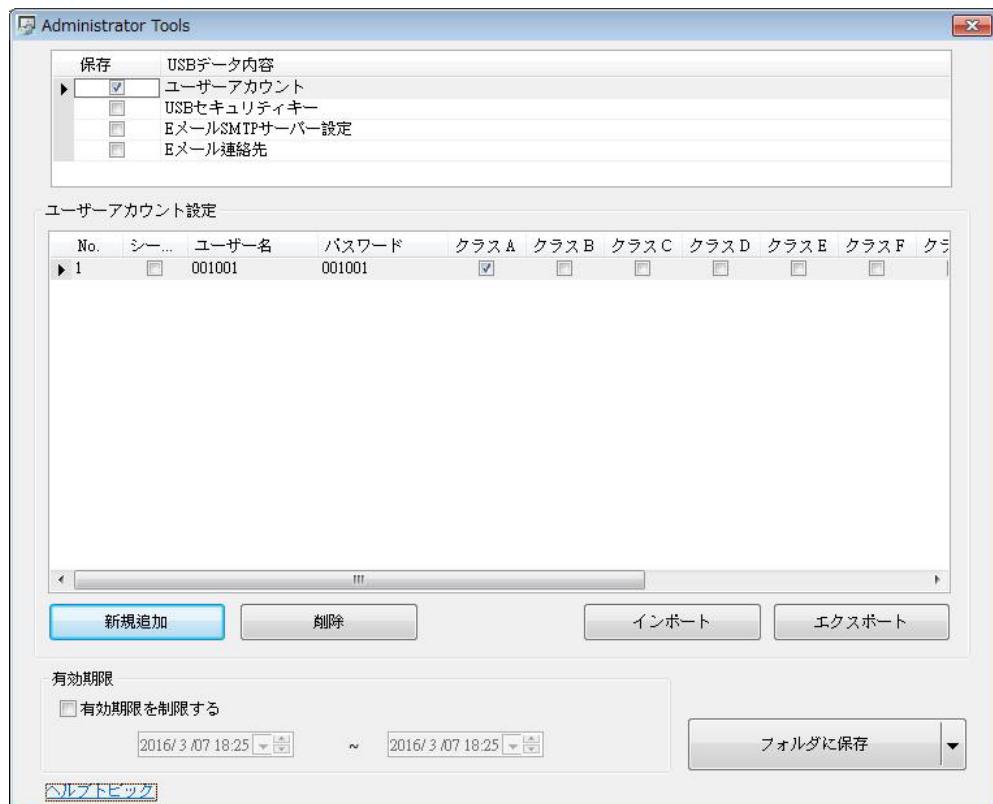
管理者ツールをオープンすると、[保存]ボックスにチェックマークを入れたら、選択した項目を使用することができます。後ろの章では、逐一各機能の設定を紹介します。



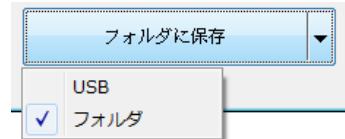
36.2. ユーザーアカウント

36.2.1. ユーザーアカウントの紹介

[ユーザー アカウント]にチェックマークを入れると、ユーザー アカウントデータを設定できます。



設定	記述
シークレットアカウント	このアカウントをシークレットアカウントに設定するのかを選択します。
ユーザー名	ユーザー名を設定します。 *Note1
パスワード	ユーザーpasswordを設定します。 *Note1
クラス A~L	ユーザー権限を設定します。
新規追加	アカウントデータを新規追加します。 *Note2
削除	アカウントデータを削除します。
インポート	ユーザーアカウントデータをインポートします。
エクスポート	ユーザーアカウントデータをエクスポートします。
有効期限	[有効期限を制限する]にチェックマークを入れなかったら、いつでもデータを HMI にインポートすることができます。 [有効期限を制限する]にチェックマークを入れ、有効期限を設定した後、データは有効期限以内に HMI にインポートする必要があります。もし有効期限が過ぎたら、データがインポートされなくなり、本ツールで改めて設定してください。
フォルダに保存	データをフォルダに保存します。 USB メモリに保存したい場合、“▼”をクリックした後、USB を選択できます。



Note

1. アルファベット、数字、“-”、“_”で組み合わせでき、大文字と小文字は異なると判定します。
2. 最大 127 レコードのアカウントデータを新規追加できます。

36.2.2. ユーザーアカウント設定

1. [新規追加]をクリックすると、1 レコードのユーザーデータを新規追加でき、[削除]をクリックすれば、そのアカウントデータを削除でき、[シークレットアカウント]をクリックすれば、当アカウントはシークレットアカウントになり、[ユーザー名]にユーザー名を入力し、[パスワード]にユーザーpasswordを入力し、そして当アカウントが使用できる権限[クラス A]~[クラス L]を選択します。

ユーザーアカウント設定

No.	シ... ー	ユーザー名	パスワード	クラス A	クラス B	クラス C	クラス D	クラス E	クラス F	クラス G	クラス H	クラス I	クラス J	クラス K	クラス L
1	<input type="checkbox"/>	Kevin	001001	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
2	<input checked="" type="checkbox"/>	Fiona	002002	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
3	<input type="checkbox"/>	Katte	003003	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
4	<input checked="" type="checkbox"/>	Barry	004004	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
5	<input type="checkbox"/>	Susan	005005	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
6	<input type="checkbox"/>	Carey	006006	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
7	<input type="checkbox"/>	Paul	007007	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
8	<input checked="" type="checkbox"/>	Ted	008008	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
9	<input type="checkbox"/>	Jim	009009	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
10	<input type="checkbox"/>	Mark	010010	<input checked="" type="checkbox"/>	<input type="checkbox"/>										

新規追加 削除 インポート エクスポート

2. アカウントデータを作成完了後、[エクスポート]をクリックすれば、データをエクスポートしバックアップできます。[passwordを使用する]にチェックマークを入れれば、データはpasswordに保護され、後日には当データをインポートしたかったら、passwordを入力する必要があります。

ユーザーアカウント設定

No.	シ... ー	ユーザー名	パスワード	クラス A	クラス B	クラス C	クラス D	クラス E	クラス F	クラス G	クラス H	クラス I	クラス J	クラス K	クラス L
1	<input type="checkbox"/>	Kevin	001001	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
2	<input checked="" type="checkbox"/>	Fiona	002002	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
3	<input type="checkbox"/>	Katte	003003	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
4	<input checked="" type="checkbox"/>	Barry	004004	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
5	<input type="checkbox"/>	Susan	005005	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
6	<input type="checkbox"/>	Carey	006006	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
7	<input type="checkbox"/>	Paul	007007	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
8	<input checked="" type="checkbox"/>	Ted	008008	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
9	<input type="checkbox"/>	Jim	009009	<input checked="" type="checkbox"/>	<input type="checkbox"/>										
10	<input type="checkbox"/>	Mark	010010	<input checked="" type="checkbox"/>	<input type="checkbox"/>										

新規追加 削除 インポート エクスポート

3. [有効期限] » [有効期限を制限する]を選択したら、この時間範囲内だけ、USB ディスクを通してユーザー アカウントデータを HMI にインポートすることができます。チェックマークを入れなかつたら、いつでもデータを HMI にインポートできます。



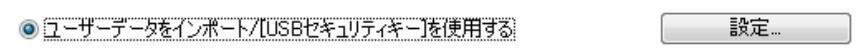
4. 全ての設定を完了後、[USB に保存する]をクリックし、保存先の USB ディスクを選択し、[作成]をクリックします。作成に成功したら、[作成に成功しました!]のメッセージが出ます。



36.2.3. EasyBuilder Pro でアカウントをインポートする

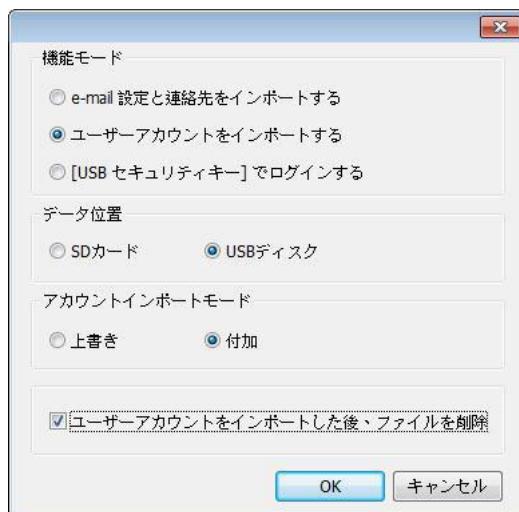
以下はオブジェクトをタッチしてから動作させるように、データをインポートするためのトリガーオブジェクトとして、EasyBuilder Pro で[ファンクションキー]オブジェクトを作成する方法を説明します。

1. EasyBuilder Pro で[ファンクションキー]オブジェクトを選択し、[ユーザーデータをインポート/[USB セキュリティキーを使用する]を選択し、[設定]をクリックします。



2. [機能モード]で[ユーザー アカウントをインポートする]を選択し、[データ位置]でデータ保存先の装置を選択します。[アカウントインポートモード]で[上書き]を選択したら、HMI は今回インポートしたアカウントデータだけを保存します。もし[付加]を選択したら、HMI

内のアカウントデータは保存され、それに今回インポートした新しいアカウントデータが付加されます。[ユーザーアカウントにインポートした後、ファイルを削除]を選択したら、アカウントをインポートした後、ソースファイルが削除されます。そして[OK]を押し、設定を完了します。



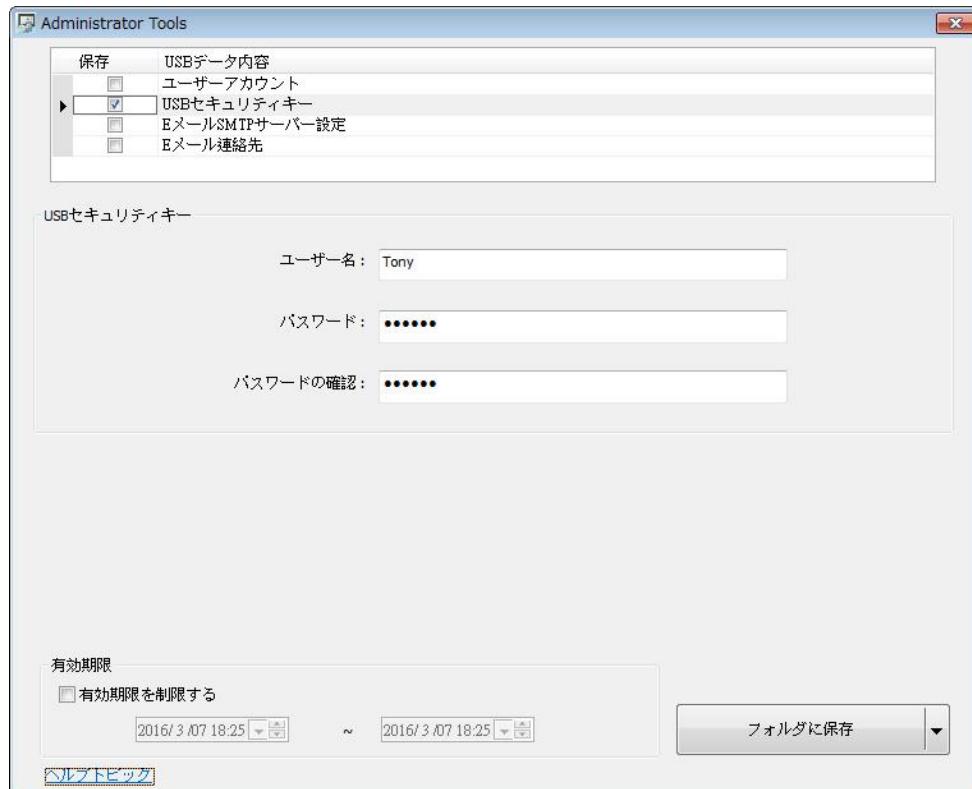
このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。本サンプルプロジェクトはファンクションキオブジェクトを用いてユーザーアカウントをインポートする方法を説明します。

サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

36.3. USB セキュリティキー

36.3.1. USB セキュリティキーの紹介

[USB セキュリティキー]にチェックマークを入れると、ユーザーセキュリティキーの設定を実行し、予めユーザーのログイン情報を設定したら、次回からユーザーセキュリティキーで直接にアカウントにログインできます。設定画面は下記に示す通りです：



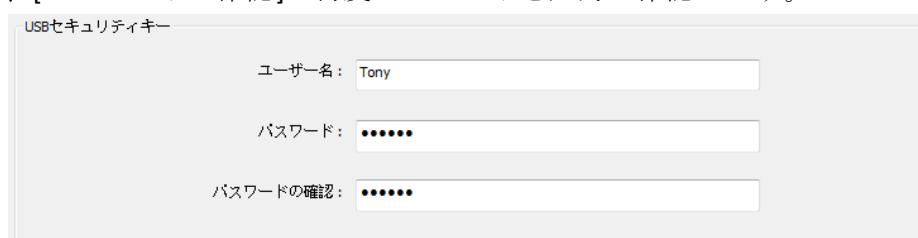
設定	記述
ユーザー名	ユーザー名を設定します。 *Note1
パスワード	ユーザーパスワードを設定します。 *Note1
パスワードの確認	ユーザーパスワードを確認します。
有効期限	有効期限とは、この期限内ではセキュリティキーでログインできます。本項にチェックマークを入れていないと、期間の制限がなくなります。
フォルダに保存	データをフォルダに保存します。

Note

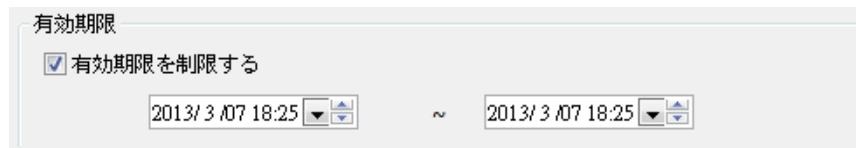
1. アルファベット、数字、“-”、“_”で組み合わせでき、大文字と小文字は異なると判定します。

36.3.2. USB セキュリティキー設定

1. [ユーザー名]にユーザーのアカウント名を入力し、[パスワード]にユーザーのパスワードを入力し、[パスワードの確認]に再度パスワードを入力し確認します。



2. [有効期限] » [有効期限を制限する]にチェックマークを入れることとは、この期限内ではセキュリティキーでログインできます。本項目にチェックマークを入れていないと、期間の制限がなくなり、いつでも本機能を実行できます。



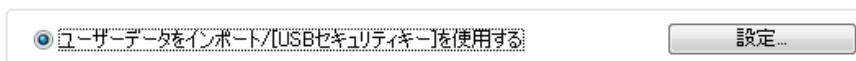
3. 全ての設定を完了後、[USB に保存]をクリックし、保存先の USB ディスクを選択し、[作成]をクリックします。作成に成功したら、[作成に成功しました!]のメッセージが出ます。



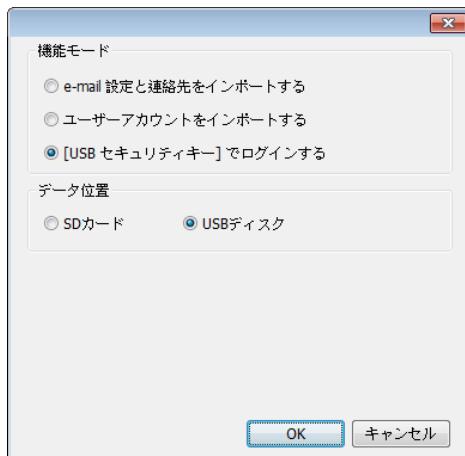
36.3.3. EasyBuilder Pro で USB セキュリティキーを設定する

EasyBuilder Pro で作成した[ファンクションキー]オブジェクトを利用し、USB セキュリティキーをトリガーしてログインすることができます。以下は上記用途で[ファンクションキー]を作成した手順を紹介します。

1. EasyBuilder Pro に[ファンクションキー]オブジェクトをクリックし、“ユーザーデータをインポート/[USB セキュリティキーを使用する]”を選択し、そして[設定]を押します。



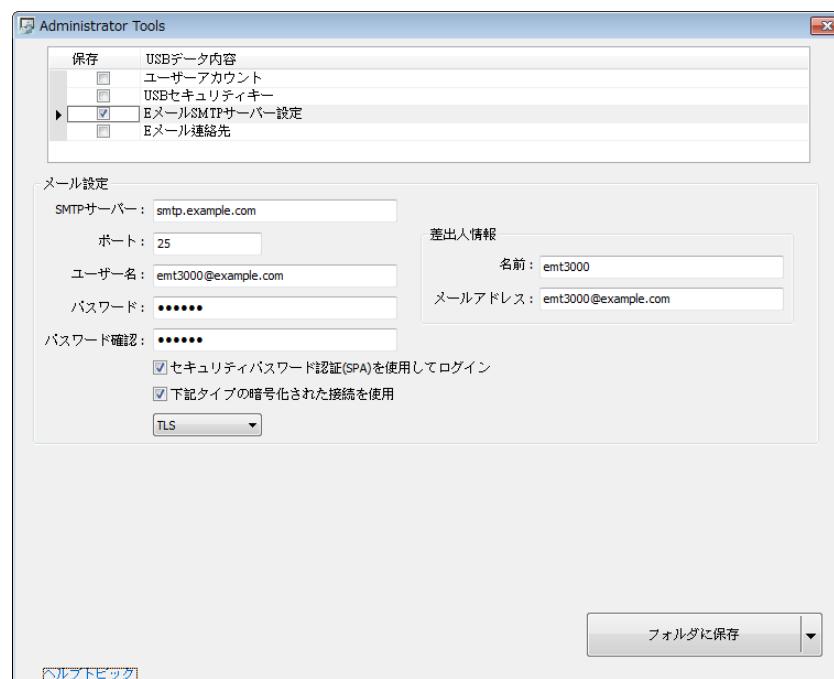
2. 機能モードで “[USB セキュリティキー]を使用する”を選択し、[データ位置]にデータの保存先を選び、そして[OK]を押して完成します。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。本サンプルプロジェクトはファンクションキオブジェクトを通して USB セキュリティキーでログインする方法を説明します。
サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

36.4. E メール SMTP サーバー

[E メール SMTP サーバー設定] にチェックマークを入れると、外部メールサーバーの設定ができます。



メール設定	記述
SMTP サーバー	SMTP サーバーを設定します。
ポート	SMTP サーバーのポート番号を設定します。
ユーザー名	メールアカウントのユーザー名を設定します。

パスワード	メールアカウントのパスワードを設定します。
パスワード確認	メールアカウントのパスワードを確認します。
差出人設定	記述
名前	メールを受け取った時、差出人の名前を表示します。
メールアドレス	メールを受け取った時、差出人のメールアドレスを表示します。
フォルダに保存	データをフォルダに保存します。

36.4.1. E メール SMTP サーバー設定

1. 下記のように E メール SMTP サーバーの設定をします。



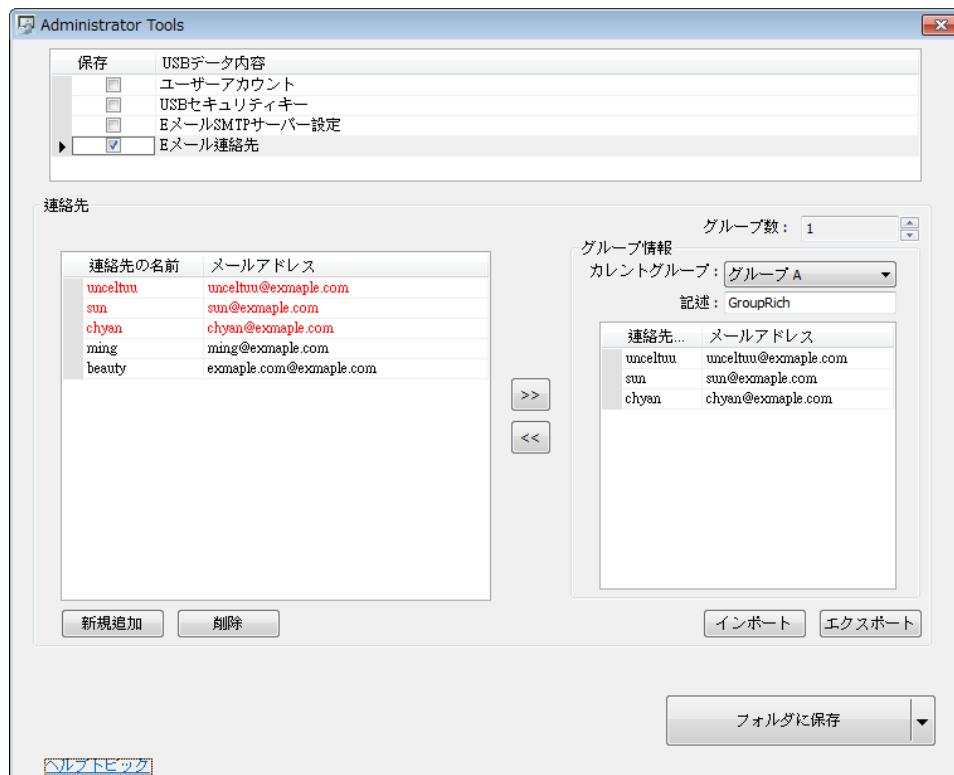
2. 全ての設定を完了後、[USB に保存]をクリックし、保存先の USB ディスクを選択し、[作成]をクリックします。作成に成功したら、[作成に成功しました!]のメッセージが出ます。



36.5. E メール連絡先

36.5.1. E メール連絡先の紹介

[E メール連絡先]にチェックマークを入れると、メールの連絡先を設定できます。



設定	記述
新規追加	連絡先を新規追加します。 *Note1
削除	連絡先を削除します。
グループ数	グループ数を設定します。 *Note2
カレントグループ	現在のグループを設定します。 *Note3
記述	グループの記述を設定します。
インポート	連絡先をインポートします。
エクスポート	連絡先をエクスポートします。
フォルダに保存	データをフォルダに保存します。

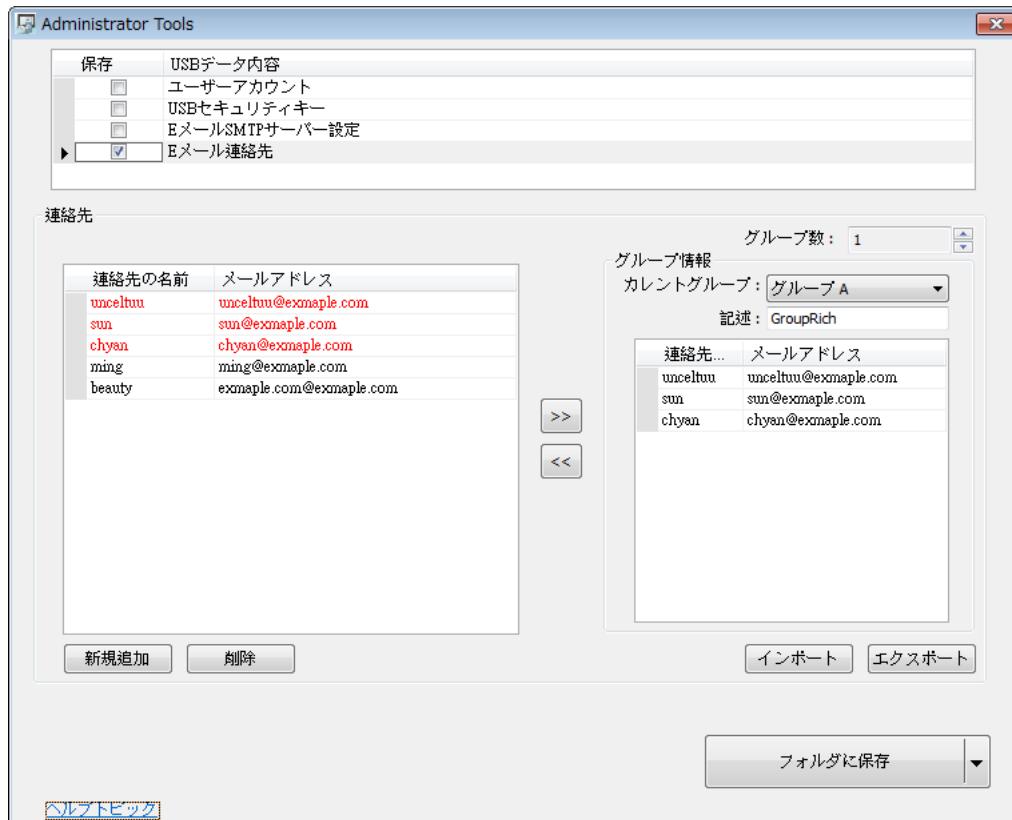


Note

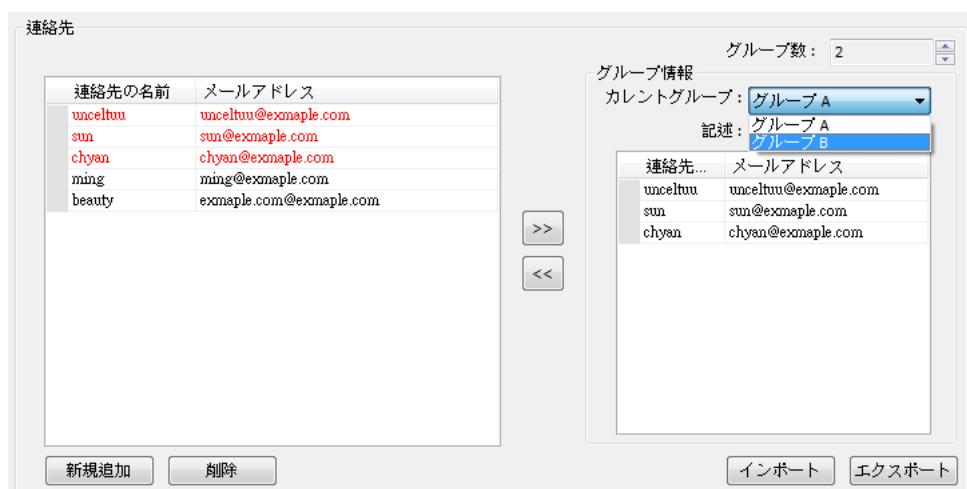
- 最大 256 レコードの連絡先を新規追加できます。
- 最大 16 個のグループを新規追加できます(グループ A~グループ P)。
- グループ A~グループ P、グループ数は 1 の時、グループ A のみ存在します。グループ数は 2 の時、同時にグループ A とグループ B が存在します。このように類推します。
- 連絡先の名前を入力するには UTF-8 フォントを使用可能です。

36.5.2. E メール連絡先設定

1. [新規追加]をクリックし、連絡先を追加します。
2. 連絡先をグループ A に追加し、このグループに追加された連絡先は赤色に表示されます。



3. グループ数を設定すれば、グループを新規追加できます。この時はグループ B が新規追加されます。ステップ a とステップ b を繰り返し、連絡先をグループに追加します。



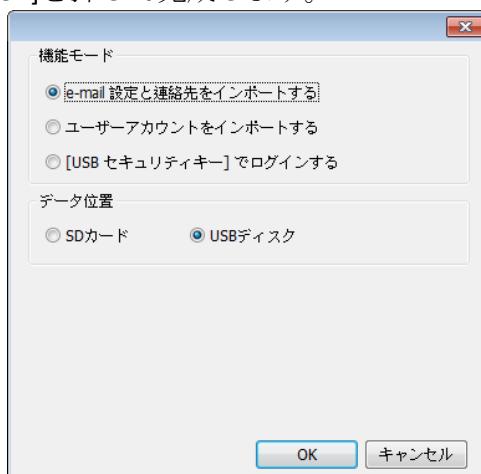
4. E メール連絡先の情報を作成できたら、[エクスポート]をクリックすれば、データをエクスポートしバックアップできます。後日に改めて作成、或いは修正したい場合があれば、[インポート]をクリックしデータをインポートできます。
5. 全ての設定を完了後、[USB に保存]をクリックし、保存先の USB ディスクを選択し、[作成]をクリックします。作成に成功したら、[作成に成功しました!]のメッセージが出ます。



36.5.3. EasyBuilder Pro を使用し、E メール設定と連絡先をインポートする

EasyBuilder Pro で作成した[ファンクションキー]オブジェクトを利用し、E メール設定と連絡先をインポートすることができます。以下は上記用途で[ファンクションキー]を作成した手順を紹介します。

1. EasyBuilder Pro に[ファンクションキー]オブジェクトをクリックし、“ユーザーデータをインポートする/[USB セキュリティキー]を使用する”を選択し、そして[設定]を押します。
2. [機能モード]で[E メール設定と連絡先をインポートする]を選択し、[データ位置]にデータの保存先を選び、そして[OK]を押して完成します。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。本サンプルプロジェクトはファンクションキーオブジェクトを通して E メール設定と連絡先をインポートする方法を説明します。

サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

37. MODBUS TCP / IP

ゲートウェイ機能

本章では、MODBUS TCP/IP ゲートウェイ機能を使用する方法及びアドレスマッピングテーブルを作成する方法について説明します。

37.1. 概要	37-2
37.2. アドレスマッピングテーブルを作成する方法	37-2
37.3. アドレスマッピング設定での注意点	37-4

37.1. 概要

従来では、SCADA (Supervisory Control and Data Acquisition) ソフトウェアで HMI と接続している PLC のデータをアクセスする時、まずは PLC のデータを HMI のローカルアドレスに送信し、そして PC で MODBUS TCP/IP プロトコルを使用して HMI のローカルアドレスに読み取ってから、PLC のデータを取得しなければなりません。

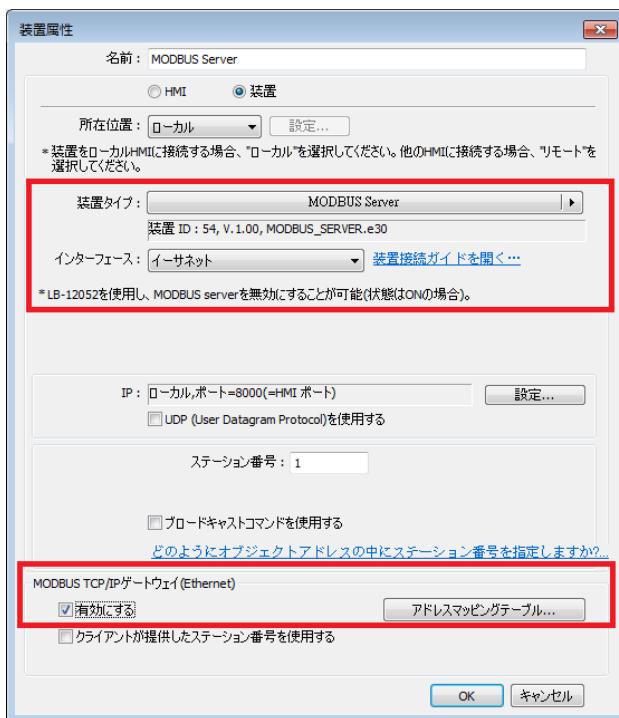
現在では、EasyBuilder Pro が提供する MODBUS TCP/IP ゲートウェイ機能を通じて、MODBUS と PLC のアドレスを前もってマッピングすれば、直接に MODBUS TCP/IP プロトコルで PLC のデータを読み取ることができます。



37.2. アドレスマッピングテーブルを作成する方法

アドレスマッピングテーブルを作成するには、下記手順に従ってください：

1. [システムパラメータ設定] » [装置リスト]に監視したいPLC装置を新規追加します。(FATEKFB Series を例として)
2. 下図に示す通り、MODBU Server(イーサネット)を新規追加し、[MODBUS TCP/IP ゲートウェイ]を使用します。



[アドレスマッピングテーブル]ボタンをクリックした後、デフォルトマッピングテーブルが表示され、ユーザーは需要に応じて修正、または他のマッピングテーブルを追加できます。

アドレスマッピングテーブル						
マッピングテーブル	記述	MODBUSアドレス	PLC名前	PLCマッピングアドレス	長さ	読み取り
1	0x <==> LB	0x-1	<==>	Local HMI	LB-0	12400 ビット 読み取り
2	1x <==> LB	1x-1	<==>	Local HMI	LB-0	12400 ビット 読み取り
3	3x <==> LW	3x-1	<==>	Local HMI	LW-0	9999 ワード 読み取り
4	4x <==> LW	4x-1	<==>	Local HMI	LW-0	9999 ワード 読み取り
5	3x <==> RW	3x-10000	<==>	Local HMI	RW-0	55536 ワード 読み取り
6	4x <==> RW	4x-10000	<==>	Local HMI	RW-0	55536 ワード 読み取り

* クロステーブルの読み取り/書き込みをサポートしていません。i.e. 同一のMODBUSコマンドで異なるテーブル内のデータをアクセスできません。

* LW-9288が最後の通信エラーコードを指示する：

0: 正常	4: 読み取り専用エラー
1: 未定義のレジスタを読み取り/書き込み	5: 書き込み専用エラー
2: 読み取り/書き込みの範囲を超えた	6: タイムアウト
3: コマンドのフォーマットが間違えた	7: インパリッド機能コード

* 下記の機能コードをサポートします：

0x: 1, 5, 15 (15はLBに設定する用のみ)
1x: 2
3x: 4
4x: 3, 6, 16

新規追加... **削除** **設定...** **OK** **キャンセル**

3. 仮に SCADA FATEK FB Series PLC の D0 レジスタからの 50 個連続したアドレスをアクセスしたい場合、設定を下記のようにします：

マッピングテーブル設定

(1) 記述: _____

(2) 装置タイプ: ビット ワード

(3) モード: 読み取り/書き込み 読み取り専用 書き込み専用

(4) MODBUSアドレス

PLC名: MODBUS Server
アドレス: 4x 0

(5) マッピングするPLCアドレス

PLC名: FATEK FB/FBs/B1/B1z Series
アドレス: D 0

(6) テーブルのサイズ
50 ワード

(7) 変換
 AB -> BA ABCD -> CDAB

OK **キャンセル**

- (1) マッピングしたいレジスタタイプを設定します。本例では[ワード]です。
- (2) マッピングしたいレジスタのアクセスモードを設定します。本例では[読み取り/書き込み]です。
- (3) マッピングしたい MODBUS 開始アドレスを設定します。本例では[4x1]です。
- (4) マッピングしたい PLC 開始アドレスを設定します。本例では[D0]です。

- (5) マッピングしたいアドレスの範囲を設定します。本例では[50]です。
 (6) 上位/下位バイト変換或いは上位/下位ワード変換を実行するかを選択します。

マッピングテーブル	記述	MODBUSアドレス	PLC名前	PLCマッピングアドレス	長さ	読み取り/書き込み
1	AC	4x1	<=>	FATEK FB/FBs/B1/B...	D-0	50ワード 読み取り/書き込み

上図の設定内容では MODBUS Server 4x1~4x50 アドレスが FATEK FB Series PLC の D0~D49 アドレスにマッピングしていると示しています。

4. 以上設定を完了後、SCADA は MODBUS TCP/IP プロトコルを通し、読み取り/書き込みコマンドを 4x1~4x50 アドレスに送信すれば、直接に FATEK FB Series PLC の D0~D49 アドレスのデータをアクセスすることができます。

37.3. アドレスマッピング設定での注意点

- [MODBUS TCP/IP ゲートウェイ]機能は、UDP をサポートしていません。
- MODBUS Server(イーサネット)インターフェースのみサポートします。
- レジスタ LW-9288 が提供され、それを用いてデータの送信が正常であるかを指示することができます。

各エラーコードは下記に示す通りです：

数値	定義
0	正常
1	アドレスマッピングテーブルに定義されていないレジスタを読み取った/書き込んだ
2	読み取った/書き込んだアドレス範囲は、アドレスマッピングテーブルが定義した範囲を超えた(或いは他のアドレスマッピングテーブル内のレジスタを読み取った/書き込んだ)
3	コマンドフォーマットは MODBUS TCP/IP プロトコルに従っていない
4	読み取り専用のレジスタを修正した
5	書き込み専用のレジスタを読み取った
6	設定した時間内、PLC からの正確な応答が来ていない
7	MODBUS Server がサポートしていないファンクションコードを使用した

- 各マッピングテーブルの間に定義したレジスタのアドレス範囲は重複してはいけません。
- Tag PLC に対するマッピングについて：
 - (1) 単一 Bool データへのマッピングに対応しません。
 - (2) Bool Array にマッピングする場合、Array の長さは 32 の倍数であること。例えば: Bool Array[32]、Bool Array[64]。

- [MODBUS TCP/IP ゲートウェイ]機能を使用した後、EasyBuilder Pro は元の MODBUS Server と HMI アドレスの間のマッピングを取り消します。下記を含みます。

- (1) 0x, 1x から LB へのマッピング
- (2) 3x, 4x から LW, RW へのマッピング

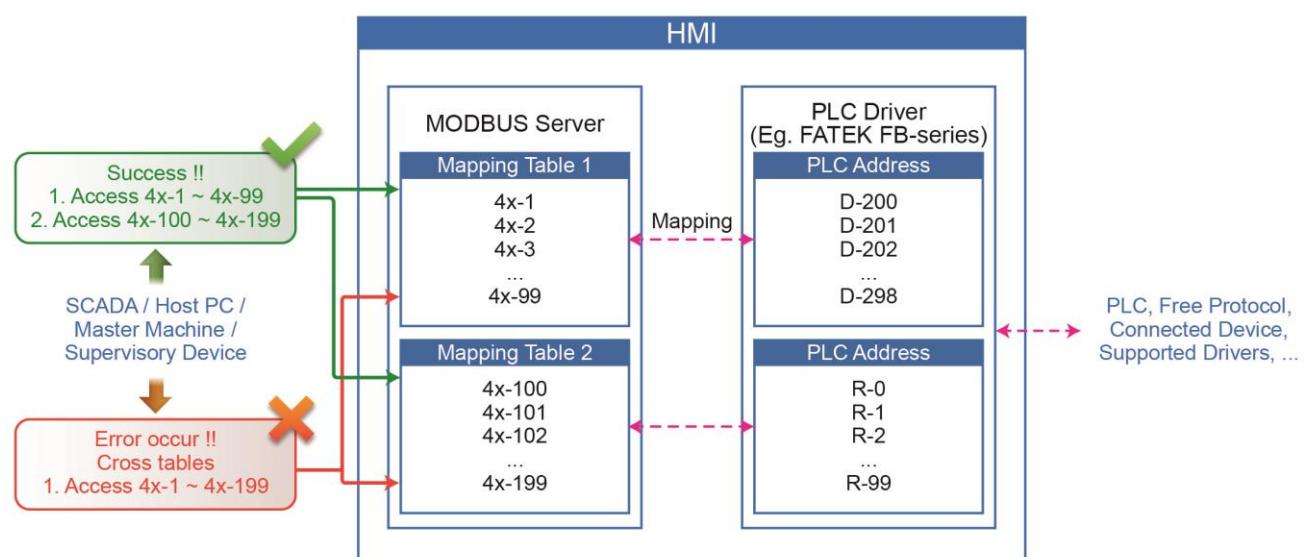
というわけで、もし 0x, 1x, 3x, 4x のコマンドを通して LB 或いは LW のデータをアクセスする必要があったら、アドレスマッピングを [アドレスマッピングテーブル]に改めて設定しなければなりません。下記の設定内容をご参考ください。

マッピングテーブル	記述	MODBUSアドレス		PLC名前	PLCマッピングアドレス	長さ	読み取り
1	0x <==> LB	0x-1	<==>	Local HMI	LB-0	12400 ビット	読み取り
2	1x <==> LB	1x-1	<==>	Local HMI	LB-0	12400 ビット	読み取り
3	3x <==> LW	3x-1	<==>	Local HMI	LW-0	9999 ワード	読み取り
4	4x <==> LW	4x-1	<==>	Local HMI	LW-0	9999 ワード	読み取り
5	3x <==> RW	3x-10000	<==>	Local HMI	RW-0	55536 ワード	読み取り
6	4x <==> RW	4x-10000	<==>	Local HMI	RW-0	55536 ワード	読み取り

- SCADA は一回で一個のマッピングテーブルのみ読み取り/書き込みできます。即ち、同一の MODBUS コマンドで異なるテーブル内のレジスタを読み取ってはいけません。

マッピングテーブル	記述	MODBUSアドレス	PLC名前	PLCマッピングアドレス	長さ	読み取り
1	AC1	4x-1	<... FATEK FB/FBs/B1/B1z Series	D-200	99 ワード	読み取り
2	AC2	4x-100	<... FATEK FB/FBs/B1/B1z Series	R-0	100 ワード	読み取り

上図を例として、[マッピングテーブル 1]に MODBUS 4x1 が D200 アドレスにマッピングし、長さを 99 に設定し、[テーブル 2]に MODBUS 4x100 が R0 アドレスにマッピングし、長さを 100 に設定した場合、SCADA が 4x1~4x199 で長さ 199 のアドレスを読み取るコマンドを発信したら、テーブルは 2 個に分けられたので、このコマンドは実行されません。コマンドを 2 個に分けて、4x1~4x99 と 4x100~4x199 を個別に読み取るようにする必要があります。下図をご参照ください：



38. EasyDownload

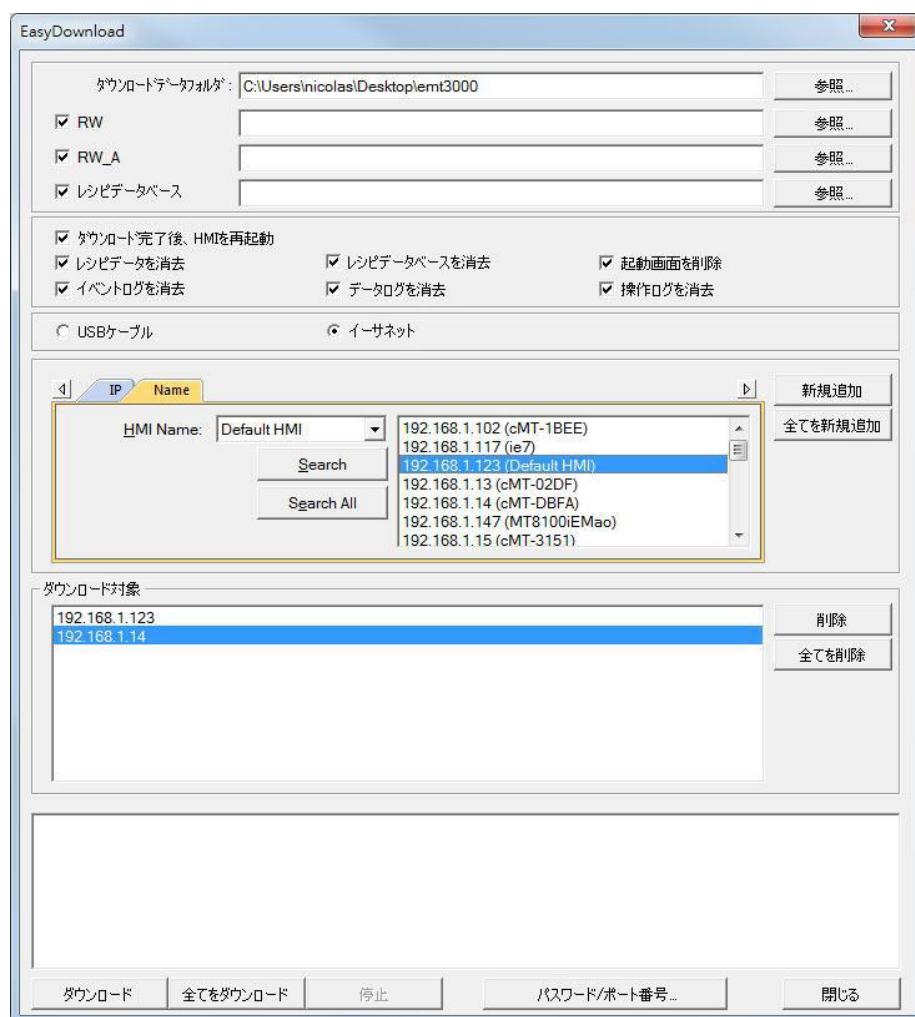
本章では、EasyDownload の使用方法について説明します。

38.1. 概要	38-2
38.2. 設定	38-2

38.1. 概要

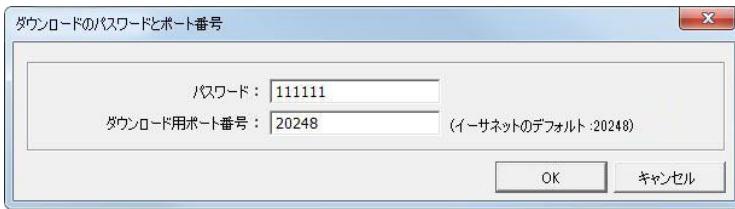
EasyDownload はイーサネットまたは USB ケーブルを通じて、EasyBuilder Pro 編集ソフトウェアで作成したプロジェクトのダウンロードデータをダウンロードできます。まずは EasyBuilder Pro 編集ソフトの[プロジェクト] » [SD カード及び USB ディスクに保存するダウンロードデータを作成]を選択し、プロジェクトのダウンロードデータを作成します。

38.2. 設定



設定	記述
ダウンロードデータフォルダ	ダウンロードしたいプロジェクトのダウンロードデータを参照します。
RW	本項目にチェックマークを入れると、ダウンロードしたいレシピデータ(.rcp)ファイルを参照できます。

RW_A	本項目にチェックマークを入れると、ダウンロードしたいレシピデータ(.rcp)ファイルを参照できます。
レシピデータベース	本項目にチェックマークを入れると、ダウンロードしたいレシピデータベース(.db)ファイルを参照できます。
ダウンロード完了後、HMI を再起動	本項目にチェックマークを入れると、ダウンロードが完了したら、自動的に HMI を再起動します。
レシピデータを消去	
レシピデータベースを消去	
起動画面を削除	チェックマークを入れると、ダウンロードを実行する前に、選択したデータが消去されます。
イベントログを消去	
データログを消去	
操作ログを消去	
USB ケーブル	USB ケーブルを通じ、ファイルを HMI にダウンロードします。まずは USB ドライバが正確に PC にインストールされたのかを確認してください。
イーサネット	イーサネットを通じ、ファイルを HMI にダウンロードします。
IP	対象 HMI の IP アドレスを入力します。
Name	対象 HMI の名前を入力します。
検索	指定した HMI 名を検索します。
全てを検索	同一のネットワーク上にある全ての HMI 名を検索します。
新規追加	検索欄で選択した HMI をダウンロード対象欄に新規追加します。
全てを新規追加	検索欄にある全ての HMI をダウンロード対象欄に新規追加します。
ダウンロード対象	ダウンロード対象になる HMI のリストです。
削除	ダウンロード対象欄で選択した HMI を削除します。
全てを削除	ダウンロード対象欄にある全ての HMI を削除します。
ダウンロード	クリックすると、ダウンロードが始まります。
全てをダウンロード	ダウンロード対象欄にある全ての HMI を順次にダウンロードします。
パスワード/ポート番号	HMI システム設定に設定されたダウンロードパスワードとポート番号を入力します。



Note

- eMT3000 シリーズを例として、作成されたフォルダのフォーマットは下記に示す通りです。ルートディレクトリの名前をダウンロードパスにしてください。

ルートディレクトリ	第二階層	第三階層
emt3000	001	
	002	
	Pub	driver
		font

- ルートディレクトリの名前はモデルによって異なります。
- イーサネットしか複数の HMI へのダウンロードをサポートしません。
- 一回でプロジェクトを複数の HMI にダウンロードする場合、全ての HMI は同じのパスワード及びポート番号を使用する必要があります。
- ダウンロードプロセスは順次に実行するので、ダウンロード中の HMI がダウンロード完了してから、次の HMI のダウンロードへ進みます。
- オフライン HMI の待ち時間が長すぎないように、待ち時間は 3 秒に設定されます。

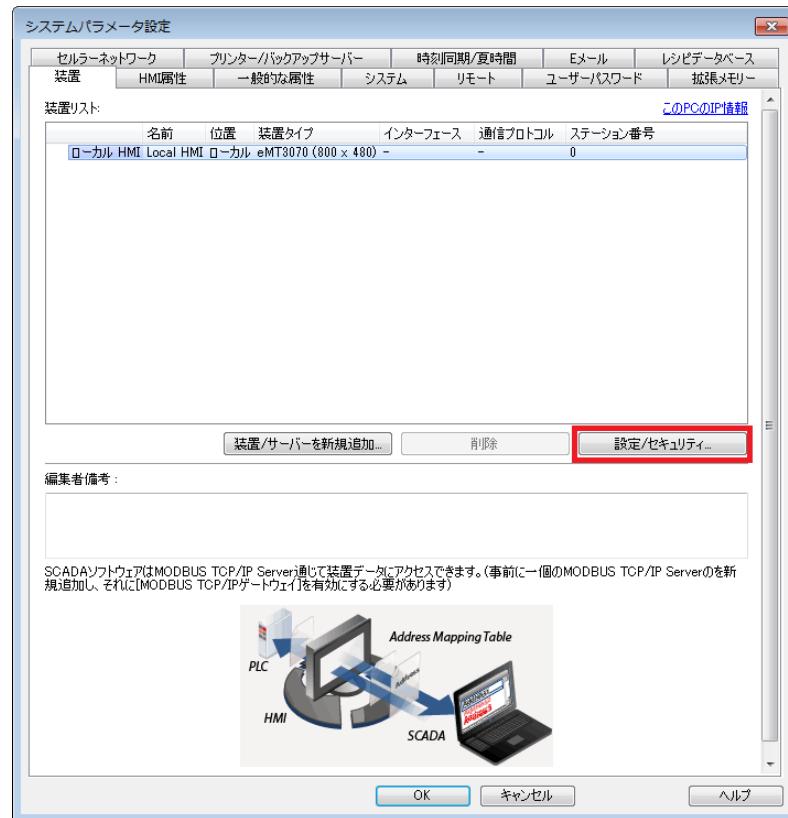
39. データセキュリティ

本章では、[データセキュリティ]機能を使用する方法について説明します。

39.1. 概要	39-2
39.2. 設定	39-2

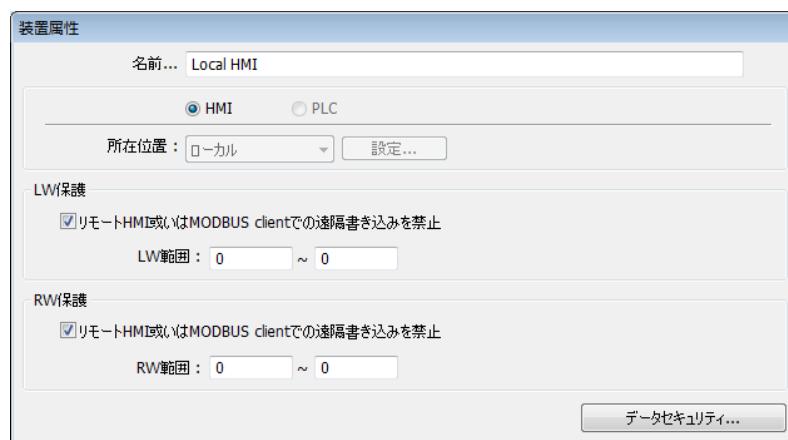
39.1. 概要

ローカル HMI のワード或いはビットアドレスレジスタに、操作時の制限を設定することができます。本機能を設定したいなら、[システムパラメータ設定] » [装置リスト]タブで[ローカル HMI]を選択し、[セキュリティ]ボタンを押します。



39.2. 設定

設定ダイアログボックスは以下のとおりです：

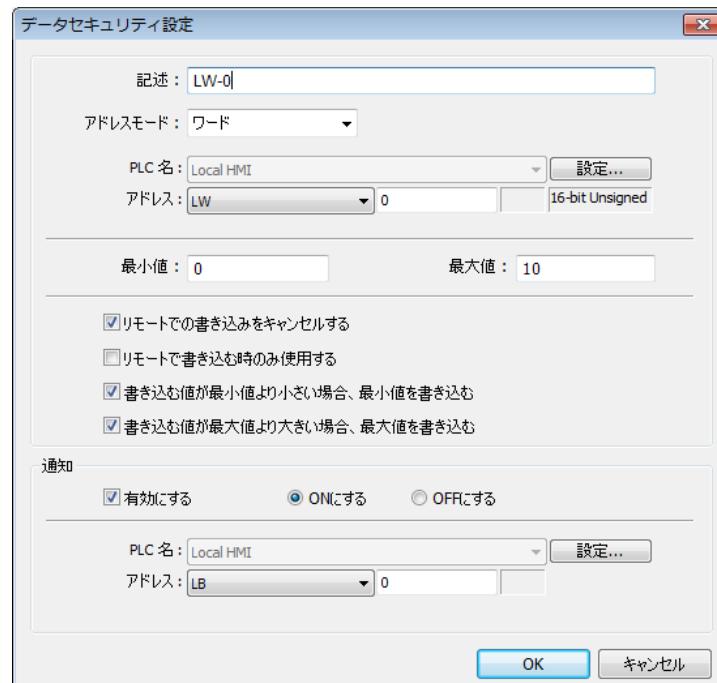


設定	記述
LW 保護	
リモート HMI 或いは MODBUS client での 遠隔書き込みを禁止	リモート HMI 或いは MODBUS client が指定された LW アドレスに書き込むのを禁じます。
RW 保護	
リモート HMI 或いは MODBUS client での 遠隔書き込みを禁止	リモート HMI 或いは MODBUS client が指定された RW アドレスに書き込むのを禁じます。

[データセキュリティ]ボタンを押した後、ローカルワードアドレス或いはビットアドレスに、書き込む時の制限を設定することができます。

39.2.1. ワードアドレス属性設定

ローカルワードアドレスに関する操作制限を設定します。



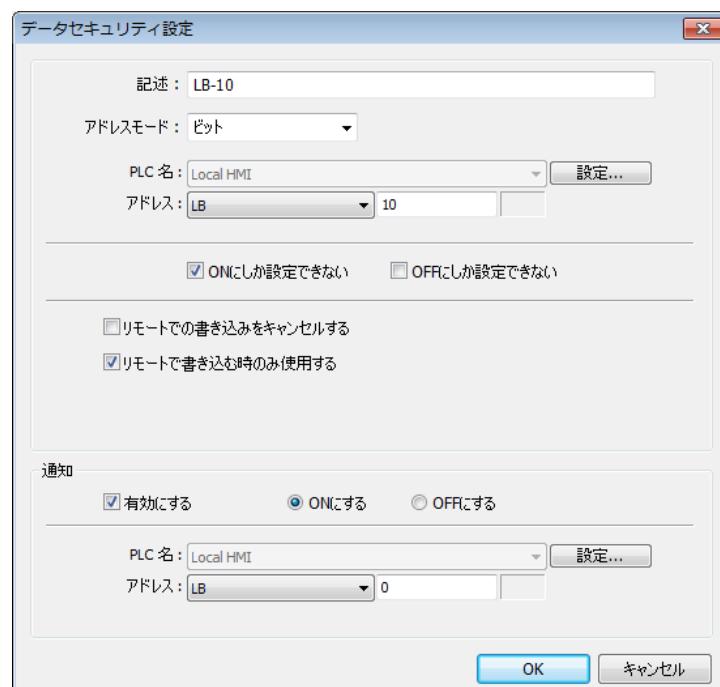
設定	記述
記述	本項目に関する説明或いは備考を入力します。
アドレスモード	[ワード]モードを選択した後、関連属性を設定できます。
最小値	このワードアドレスに入力する数値の最小値を設定します。
最大値	このワードアドレスに入力する数値の最大値を設定します。
リモートでの書き込みをキャンセルする	チェックマークを入れると、リモート HMI はこのアドレスに書き込めません。

リモートで書き込む時のみ使用する	チェックマークを入れると、リモートで書き込む場合のみ[最小値]と[最大値]の制限があります。
書き込む値が最小値より小さい場合、最小値を書き込む	チェックマークを入れると、入力された値が[最小値]より小さい場合、システムは最小値を書き込みます。チェックマークを入れないと、入力された値が[最小値]より小さい場合、システムは本来の数値を維持します。
書き込む値が最大値より大きい場合、最大値を書き込む	チェックマークを入れると、入力された値が[最大値]より大きい場合、システムは最大値を書き込みます。チェックマークを入れないと、入力された値が[最大値]より大きい場合、システムは本来の数値を維持します。
通知	入力された値が[最小値]より小さい、或いは[最大値]より大きい場合、システムは指定された通知ビットアドレスをトリガーします。

上図を例として、リモート HMI は LW-0 を書き込めません。それに、ローカルで書き込む数値が 10 より大きい場合、システムは 10 を書き込み、同時に通知ビットアドレス LB-0 を ON にトリガーします。

39.2.2. ビットアドレス属性設定

ローカルビットアドレスと関連する操作制限を設定します。



設定	記述
記述	本項目に関する説明、或いは備考を入力します。

アドレスモード	[ビット]モードを選択した後、関連属性を設定できます。
ONにしか設定できない	このビットアドレスは ON にしか設定できません。
OFFにしか設定できない	このビットアドレスは OFF にしか設定できません。
リモートでの書き込みをキャンセルする	本項にチェックマークを入れると、リモート HMI はこのセキュリティされたアドレスに書き込めません。
リモートで書き込む時のみ使用する	本項にチェックマークを入れると、リモートで書き込む時だけが[ONにしか設定できない]と[OFFにしか設定できない]の制限があります。
通知	本項を使用し、[ONにしか設定できない]にチェックマークを入れると、このセキュリティされたビットアドレスを OFF に設定したら、システムは通知ビットアドレスをトリガーします。 逆に、[OFFにしか設定できない]にチェックマークを入れると、このセキュリティされたビットアドレスを ON に設定したら、システムは通知ビットアドレスをトリガーします。

上図を例として、リモート HMI の場合、LB-10 が[ONにしか設定できない]に設定され、ローカルで書き込むのは制限されません。だから、リモート HMI が LB-10 を OFF に設定すると、システムは通知ビットアドレス LB-0 を ON にトリガーします。

40. ウェブストリーミング

本章では、ウェブストリーミングの設定方法について説明します。

40.1.	概要	40-2
40.2.	設定	40-2
40.3.	操作方式	40-2
40.4.	対応機種	40-3

40.1. 概要

ウェブストリーミング機能は、サードパーティ製装置の上に、ウェブブラウザを通し、HMIに接続されたUSBカメラが撮影した映像を閲覧・ストリーミングできる機能です。

このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

40.2. 設定

EasyBuilder Proの中では、ウェブストリーミング機能を設定するための特定のオブジェクトはありません。ウェブストリーミング機能は、システムレジスタによりコントロールされます。ウェブストリーミング機能を使用する前、ビットアドレスを設定、またはマクロを使用し、下記の関連システムレジスタをプロジェクトに入れてください。

- LB-12356: ウェブストリーミングサーバーをオープンする(ONに設定する) / 閉じる(OFFに設定する)
- LB-12357: ウェブストリーミングサーバー状態(ON:オープンする / OFF:閉じる)

LB-12356をONに設定すれば、ウェブストリーミング機能がオープンされます。LB-12356をOFFに設定すれば、ウェブストリーミング機能が閉じられます。

LB-12357はウェブストリーミング機能の状態を示します。ONの場合、ストリーミング機能が使用中と示し、OFFの場合、ストリーミング機能が停止していると示しています。USBカメラが接続していないと、ウェブストリーミング機能が使用されません。

40.3. 操作方式

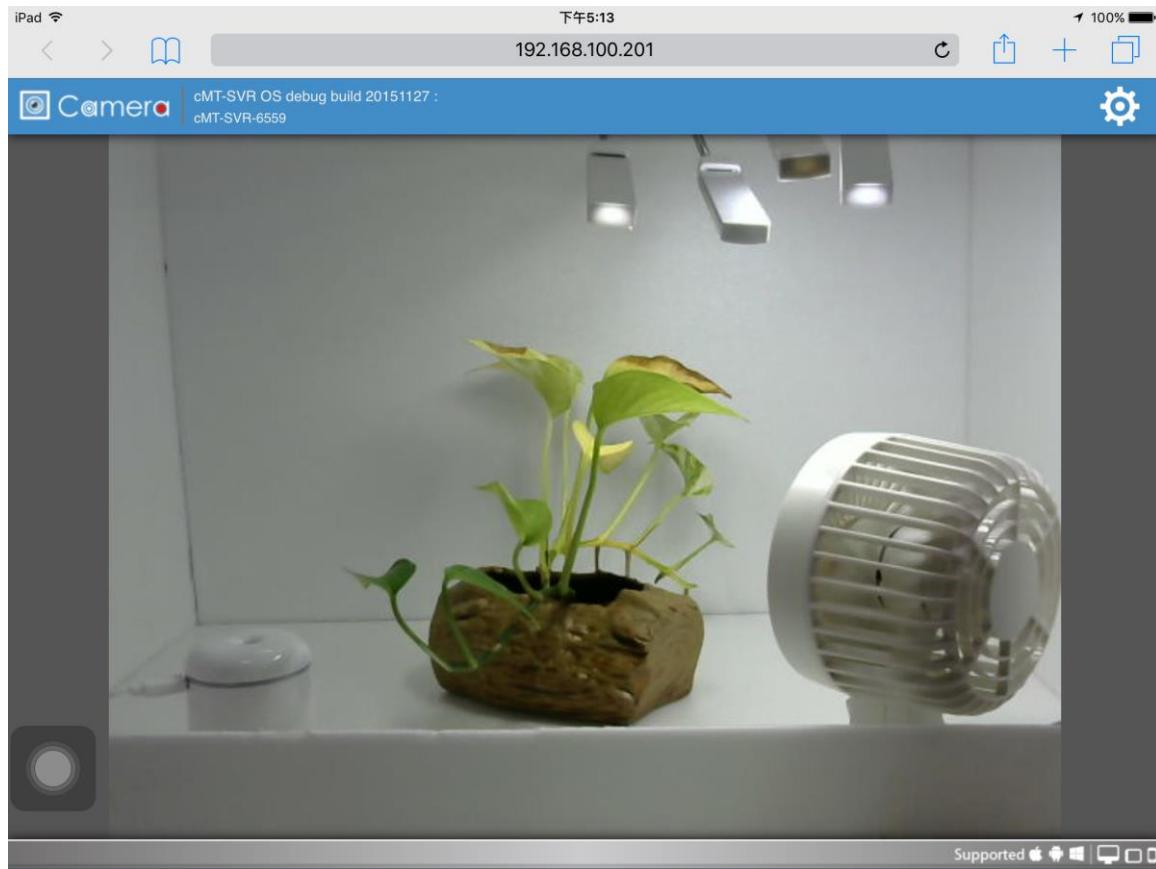
まずはウェブストリーミングサーバーの状態はオープンされたと確認してください。映像をストリーミングするには、ブラウザにHMIのIPアドレスを入力し、同時にポート番号を8080に指定してください。例えば、IPアドレスは192.168.1.100の場合、

<http://192.168.1.100:8080>

を入力してください。

例 1

以下は iPad で Safari ブラウザを通して、植物を監視する映像です。



Note

EasyAccess 2.0 を使用する場合、HMI の IP アドレスは、ソフトウェアから取得した仮想 IP アドレスです。

40.4. 対応機種

機種	OS バージョン	EBPro バージョン
eMT シリーズ	20150525	V5.04.01.013
XE シリーズ /eMT3070B	20150225	V5.04.01.013
mTV	20150508	V5.05.01
cMT-SVR	20151117	V5.03.02
cMT3090	20171024	V6.00.01
cMT3151		V6.00.01

41. エネルギー管理

本章では、どのようにエネルギー需要量設定で電力消費量を監視・記録し、それに将来のエネルギー需要量を計算する方法について説明します。

41.1. エネルギー需要量設定	41-2
41.2. エネルギー需要量表示	41-5

41.1. エネルギー需要量設定

41.1.1. 概要

エネルギー需要量設定の機能を通じて、需要量計算周期(T)と需要量更新頻度(t)の設定によって電力需要量を計算できるので、電力消費量を把握し、省エネに役に立ちます。

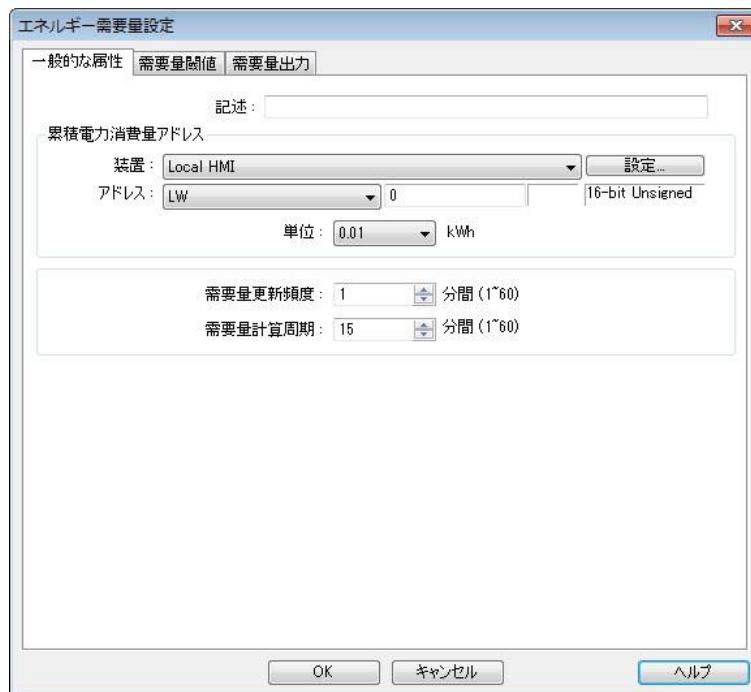
41.1.2. 設定



ツールバーの[データ/履歴]をクリックし、そして[エネルギー需要量設定]ボタンを押せば、[エネルギー需要量設定]オブジェクトの属性ダイアログボックスがオープンされ、新規作成を選択し、正確に一般的な属性及び需要量閾値を設定完了して OK ボタンを押すと、[エネルギー需要量設定]オブジェクトが新規作成されます。



一般的な属性



設定	記述
累積電力消費量	電力消費量の記録アドレスです。単位は 0.1/0.01/0.001
アドレス	kWh があります。
需要量更新頻度 (t)	一回の電力消費量を計測する頻度を設定します。範囲は 1~60 分間です。
需要量計算周期 (T)	一回の電力需要量を計算する周期を設定します。範囲は 1~60 分間です。

Note

- 需要量計算周期(T)は需要量更新頻度(t)の整数の倍数でなければなりません。

需要量閾値



設定	記述
閾値	ウォーニング欄及びアラーム欄に閾値を設定でき、閾値になると、ウォーニングとアラームはトリガーされます。動的な設定もできます。
通知	需要量がウォーニング、またはアラームの閾値を超えたら、指定したアドレス状態が変わります。
自動的にフォローする	本項を選択すると、需要量数値が復帰したら、指定したアドレス状態もデフォルトに復元します。

需要量出力



設定	記述
有効にする	チェックマークを入れ、最大需要量統計の設定を開けます。
最大需要量統計	本日・昨日、及び本月・先月の最大需要量を記録することができます。時間と日付のフォーマット、及び支払請求周期起算日を選択できます。

41.2. エネルギー需要量表示

41.2.1. 概要

[エネルギー需要量表示]オブジェクトは HMI が動作している間に、特定のエネルギー需要量の情報をグラフで表示します。自由にフォント、罫線、線分の色と大きさを調整することができ、ウォーニングとアラームの境界線も[エネルギー需要量表示]オブジェクトで表示できます。

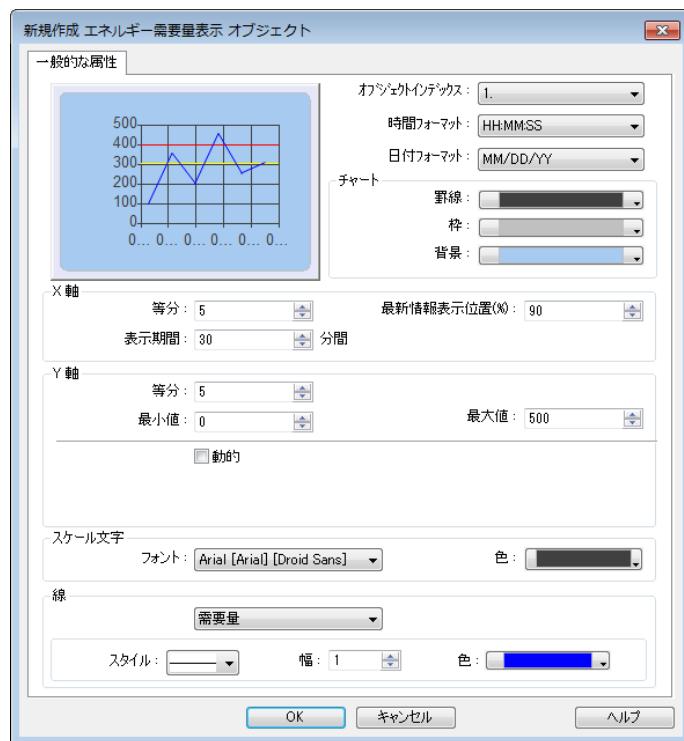
41.2.2. 設定



ツールバー上の[データ/履歴]をクリックし、そして[エネルギー需要量表示]ボタンを押せば、[エネルギー需要量表示]オブジェクトの属性ダイアログボックスがオープンされ、正確に各属性を

設定完了して OK ボタンを押すと、[エネルギー需要量表示]オブジェクトが新規作成されます。

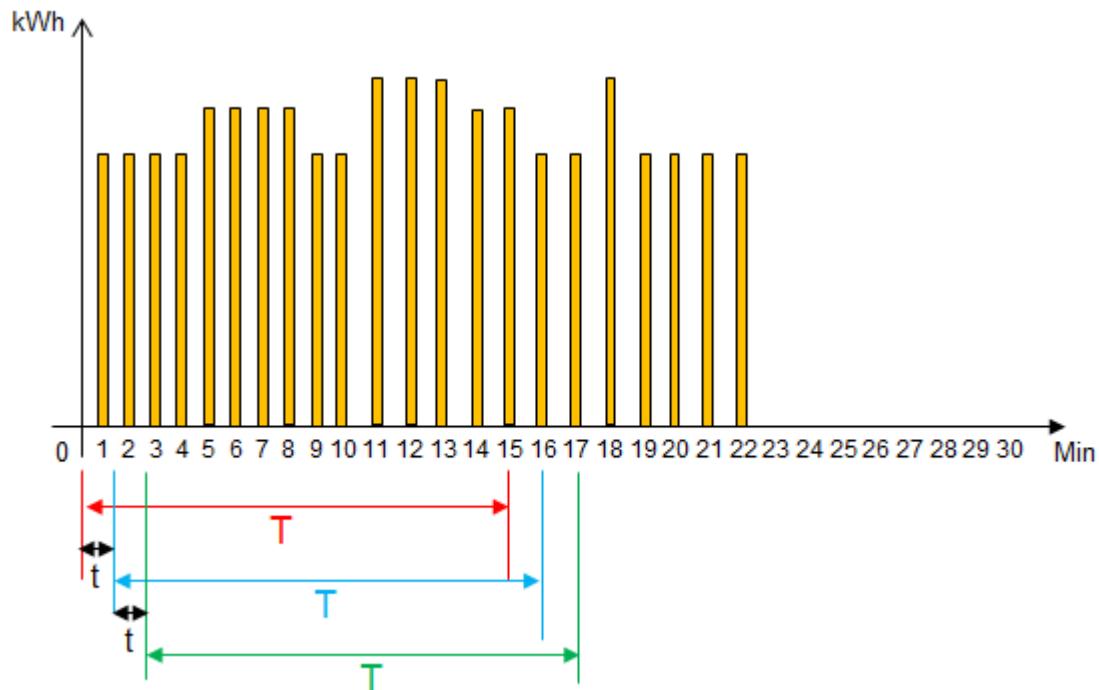
一般的な属性



設定	記述
オブジェクトインデックス	閲覧したいエネルギー需要量情報を選択します。
時間/日付フォーマット	表示する時間と日付のフォーマットを設定します。
チャート	罫線、外枠及び背景の色を設定します。
X 軸	等分、表示期間及び最新情報の表示位置(50%~100%)を設定します。
Y 軸	等分及び上下限数値を設定します。上下限値を動的に調整することができます。
スケール文字	文字のフォント及び色を選択できます。
線	需要量、ウォーニング及びアラーム線のスタイル、幅、色を設定します。

例 1 エネルギー需要量計算方法

下記の例で需要量更新頻度(t)と需要量計算周期(T)と需要量の関係を説明します。



1. 仮に上図に示す通り、需要量更新頻度(t)は1分間で、需要量計算周期(T)は15分間とします。
2. 第1~15分間は赤色の周期で、電力消費量を全部足してから、4(1時間に計算する必要がある)をかけると、一個の需要値(kwh)が算出されます。
3. 第2~16分間は青色の周期で、電力消費量を全部足してから、4(1時間に計算する必要がある)をかけると、一個の需要値(kwh)が算出されます。
4. 第3~17分間は緑色の周期で、電力消費量を全部足してから、4(1時間に計算する必要がある)をかけると、一個の需要値(kwh)が算出されます。
5. エネルギー需要量表示オブジェクトは算出された需要値を集め、グラフを作成します。
6. もし $t=3$ 、 $T=15$ だったら、最近5回の電力消費量を足してから、4(1時間に計算する必要がある)をかけることになります。
7. もし $t=5$ 、 $T=30$ だったら、最近6回の電力消費量を足してから、2(1時間に計算する必要がある)をかけることになります。このように類推します。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

42. IIoT

本章では、IIoT の通信プロトコルの使用方法について説明します。

42.1. MQTT	42-2
42.2. OPC UA サーバー	42-26

42.1. MQTT

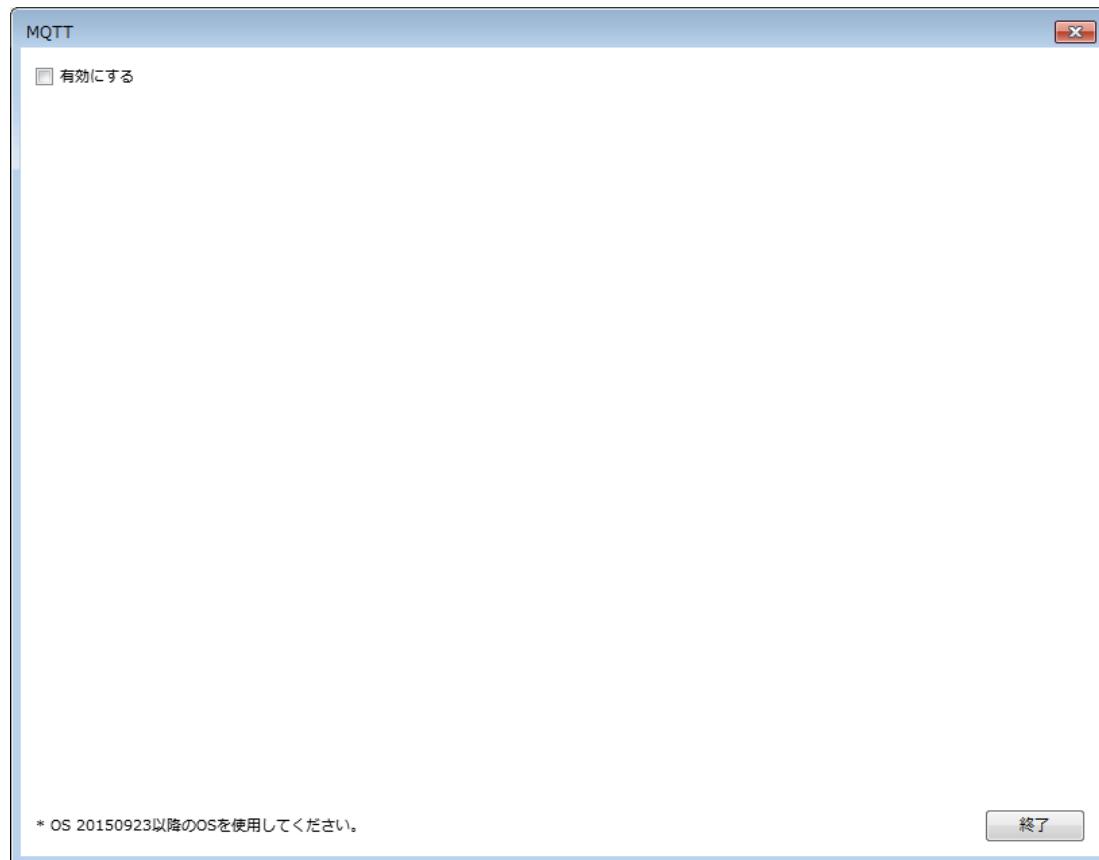
42.1.1. 概要

[MQTT]オブジェクトを使用すれば、メッセージを MQTT サーバーに送信でき、一方 MQTT サーバーからトピックを購読することもできます。HMI を MQTT サーバーとして使用することもできます。HMI を MQTT サーバーにする場合、メッセージを他のサーバーに送信しません。

42.1.2. 設定

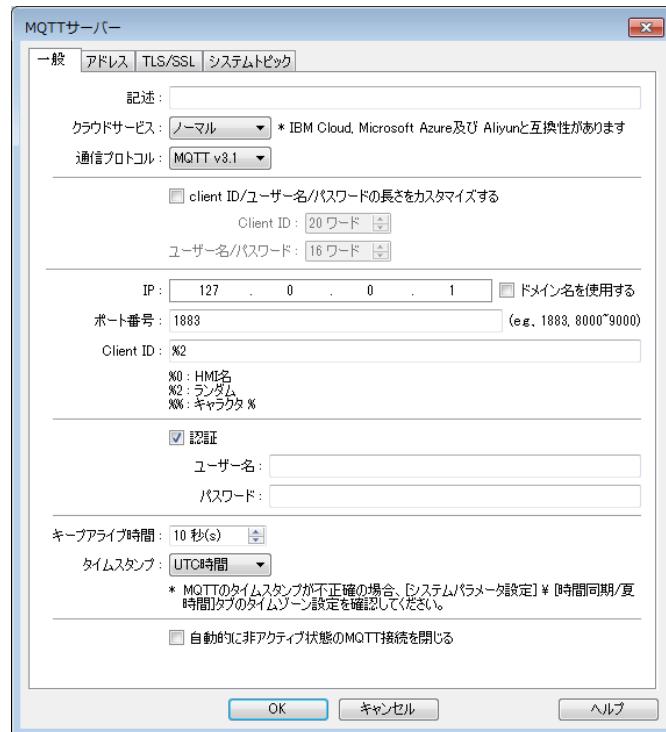


ツールバーの[オブジェクト]をクリックし、そして[IIoT] » [MQTT]をクリックして本オブジェクトを新規作成してください。有効にしてから、設定ダイアログボックスが現れます。



42.1.2.1. サーバー設定

一般的な属性の設定



設定	記述
クラウドサービス	<p>一般 一般的な MQTT トピックを発行-購読するモードです。</p> <p>AWS IoT AWS IoT をブローカーにします。Thing でデータを転送し、Shadow 機能をサポートします。詳細は“AWS IoT ユーザーマニュアル”をご参照ください。</p> <p>Sparkplug B Sparkplug B は IIoT 応用の特性に基づいてデザインされた規格です。本規格は標準 MQTT 規格が規制していないトピック及びメッセージ内容を定義するのに役立ち、それに MQTT 未対応の装置でも Edge of Network (HMI)と通じて間接に MQTT 送信ができるようにさせます。詳細は“Sparkplug B スタートアップガイド”をご参考ください。</p> <p>Azure IoT Hub Microsoft Azure IoT Hub を Broker にします。本サービ</p>

スを使用する場合、正確なストリングを記入すれば通信でき、設定手順が簡単です。接続用のストリングは Microsoft Azure > IoT devices で見つかります。



Google Cloud IoT Core

Google Cloud IoT Core を Broker にし、接続パラメータ及び関連ファイルを検証すれば通信できます。



通信プロトコル	MQTT v3.1 と v3.1.1 をサポートします。
Client ID/ユーザー名/パスワードの長さをカスタマイズする	Client ID : 長さ上限は 128 ワードです。 ユーザー名/パスワード : 長さ上限は 256 ワードです。
IP	メッセージを受信する MQTT サーバーIP を設定します。IP アドレスに 127.0.0.1 を入力すると、HMI の MQTT サーバーが起動されます。
ドメイン名を使用する	ドメイン名: <input type="text" value="127.0.0.1"/> <input checked="" type="checkbox"/> ドメイン名を使用する ドメイン名をサーバーアドレスとして使用するのをサポートします。
ポート番号	受信する MQTT サーバーの接続ポートを設定します。

Client ID	ログイン名です。変数を Client ID にすることができます。例えば: %0 を入力すれば、HMI の名前が Client ID になります。
検証	[ユーザー名]及び[パスワード]で MQTT サーバーに接続するのかを選択します。
ユーザー名	MQTT サーバーに接続用の[ユーザー名]です。
パスワード	MQTT サーバーに接続用の[パスワード]です。
キープアライブ時間	[キープアライブ時間]を超えて、まだ HMI のキープアライブが送られてこない場合、MQTT サーバーは HMI との接続が切断されたと判断します。 備考: シミュレーションを使用する場合、メッセージの通信が遅延される可能性があります。但し、遅延時間は長くとも[キープアライブ時間]を超ません。 HMI 上のメッセージは即時に送信します。
タイムスタンプ	ローカル時刻 HMI 時刻をタイムスタンプにします。 UTC 時刻 UTC+0 時刻を使用します。タイムスタンプが不正確だった場合、[システムパラメータ設定] » [時刻同期/夏時間]タブのタイムゾーンの設定を確認してください。
自動的に非アクティブ中の MQTT 接続を切断する	アイドル時間が[アイドル制限]に設定した値を超えると、自動的に接続を切断し、次回データが更新される時になってから、接続を再開します。 初めての接続の場合のみ、デフォルト数値及びトピックリストを更新するのかを選択できます。 本設定を使用する場合、コントロールアドレスの開始/停止コマンドが無効になります。

アドレス設定



設定

記述

状態アドレス

LW-n: [MQTT]の接続状態を表示する

数値	記述
0	MQTT サーバーに接続していない
1	接続が切断された。MQTT サーバーに接続できない
2	MQTT サーバーとの接続に成功した

LW-n+1: エラー提示

数値	記述
0	エラー無し
1 or more	エラー有り

バッファ使用量アドレス

配信に成功していない場合、メッセージはバッファとしてメモリーに保存します。最大 10000 レコードです。アドレスに表示される単位は%で、端数を切り上げます。

LW-n: バッファ使用量を表示する

コントロールアドレス

LW-n: [MQTT サーバー]の実行または停止をコントロールする

数値	記述

0	準備完了
1	開始
2	停止
3	更新

LW-n+1: MQTT サーバーの IP アドレスを設定する

LW-n+5: MQTT サーバーの接続ポート番号を設定する

LW-n+6: MQTT サーバーに接続用の Client ID を設定する

LW-n+26: 検証を無効/有効にする

数値	記述
0	無効にする
1	有効にする

LW-n+27: MQTT サーバーに接続用のユーザー名を設定する

LW-n+43: MQTT サーバーに接続用のパスワードを設定する

- クラウドサービスに Azure IoT Hub を選択した場合、コントロールアドレスは以下の通りです：

LW-n: [MQTT サーバー]の実行・停止を制御する

数値	記述
0	準備完了
1	開始
2	停止
3	更新

LW-n+1: 接続ストリングを設定する(128 words)

TLS/SSL 設定



設定	記述
有効にする	<p>TLS/SSL 暗号化を有効にします。手動でバージョン:TLS 1.0、TLS 1.1 或いは TLS 1.2 を選択することができます。</p> <p>TLS 1.1 及び TLS 1.2 を使用する場合、HMI の OS には 20180323 以降のバージョンを使用すること。</p>
サーバー検証	<p>有効にする</p> <p>サーバーの認証が認証局(CA)の認証を得たのかを検証します。サーバー認証は接続の時に、サーバーから送られます。</p> <p>サーバー名は認証情報に一致しなければなりません</p> <p>サーバーの IP はサーバー認証内の IP 記録に一致しているかどうかを検証します。IP 記録は認証内の Subject Alternative Name に記載されています。</p>
クライアント検証	秘密鍵及び認証はサーバーにクライアントを検証させるための必要資料です。

システムトピック

現在は四種類のシステムトピックがあり、HMI はそれらを発行するかどうかを選択できます。HMI がシステムトピックを有効にした場合、購読者はそのトピックを購読することにより、その HMI のトピックリストまたは接続状態を知ることができます。



設定	記述
Topic List	HMI 内のトピックリストです。 本メッセージは、HMI がサーバーと接続した後、サーバーに送信されます。
Birth Topic	HMI がサーバーに接続した後、送信するメッセージです。
Close Topic	HMI から自動的にサーバーとの接続を切断する前、最後に送信するメッセージです。
Last Will	クライアントがサーバーの間に異常が発生して接続が切断された場合、Last Will の購読者は本メッセージを受信します。クライアントが最初に CONNECT メッセージをサーバーに接続をリクエストした時も、同時に Last Will メッセージの内容を含めて送信します。
トピック	システムトピックの実際のトピック名です。
メッセージを保持	本項にチェックマークを入れると、MQTT サーバーは最新の 1 レコードのメッセージを確保します。
QoS	MQTT は 3 レベルの信頼性を提供します。即ち QoS(キューオーエス)のことです。メッセージ配布の信頼性はメッセージが確実に届くのかを決めます。 QoS 0: 最高一回：メッセージを一回だけ配布するが、確実に届くかは保証しません。 QoS 1: 最低一回：メッセージは最低一回届きます。 QoS 2: 正確に一回：メッセージは正確に一回届きま

内容フォーマット

す。

JSON (Default) : デフォルト値を使用します。

各システムトピックのデフォルト値 :

カレントの実際値は赤字で示します。

Topic list:

```
{  
    "d": {  
        "topics": [  
            {  
                "compression": "圧縮タイプ",  
                "nickname": "トピック名",  
                "topic": "トピック"  
            },  
            {  
                "compression": "圧縮タイプ",  
                "nickname": "トピック名",  
                "topic": "トピック"  
            }  
        ]  
    },  
    "ts": "現在時刻"  
}
```

topics 内のメッセージは実際のトピックの設定により異なります。上記はトピックが 2 個の例です。

Birth Topic:

```
{  
    "d": {  
        "connected": true  
    },  
    "ts": "現在時刻"  
}
```

Close Topic:

```
{  
    "d": {
```

```
        "connected":false
    },
    "ts":"現在時刻"
}
```

Last Will:

```
{
    "d":{
        "connected":false
    }
}
```

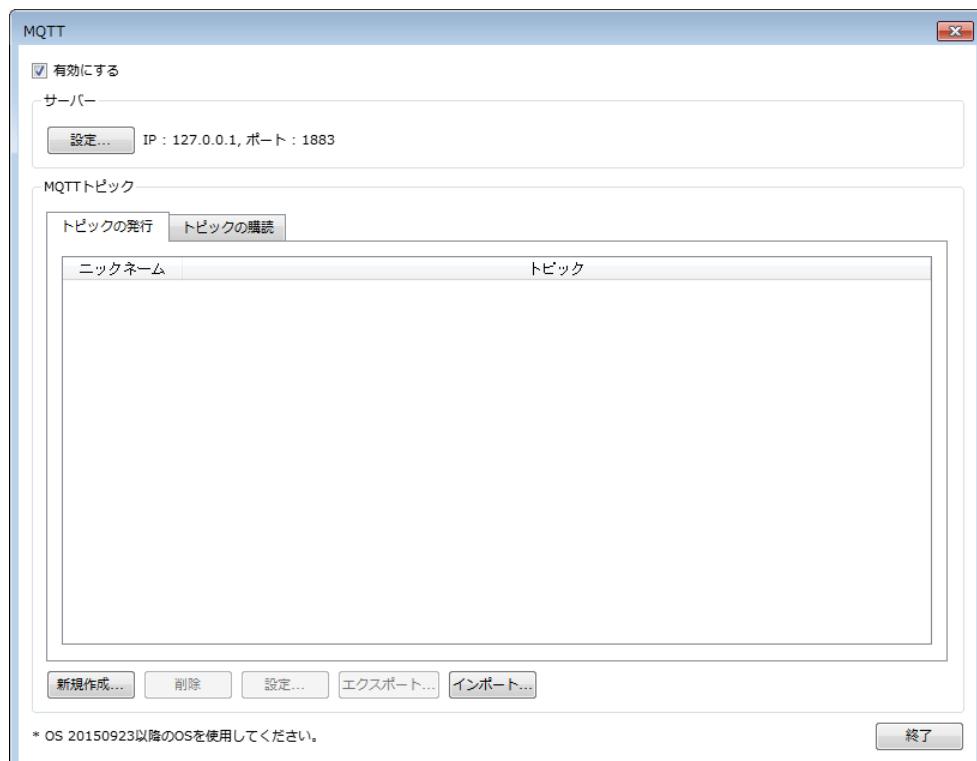
JSON (Customized) : カスタマイズした内容を使用します。

Note

クラウドサービスに Sparkplug B、Azure IoT Hub 及び Google Cloud IoT Core を選択した場合、[システムトピック]タブがサポートされません。

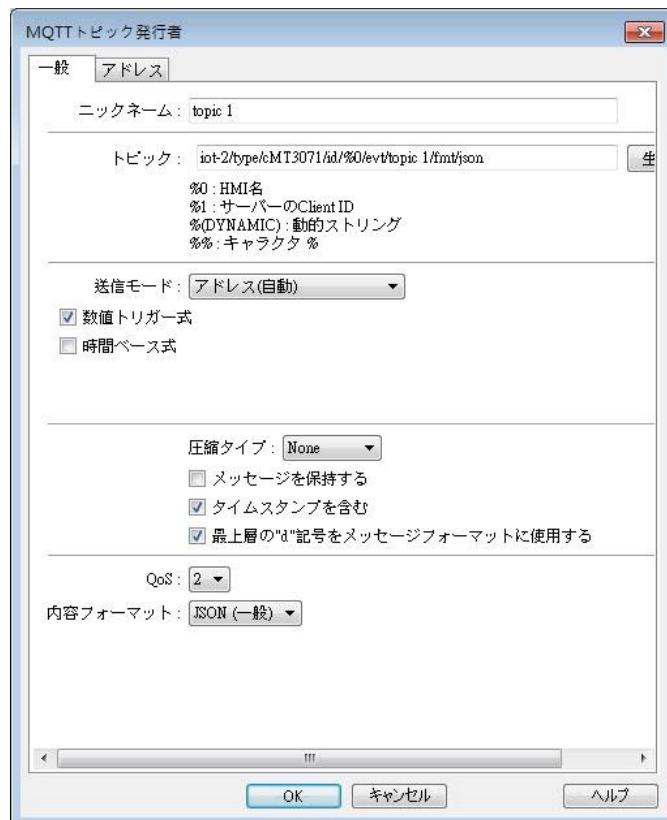
▶このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

42.1.2.2. MQTT トピックの発行



[新規作成]を選択すれば、一般的な属性及びアドレスの設定に入ることができます。または直接に CSV ファイルを[エクスポート]/[インポート]する機能で MQTT 発行トピックを作成してもいいです。MQTT 発行トピックの数量上限は 255 です。

一般的な属性の設定



設定	記述
ニックネーム	MQTT トピックの項目名を設定します。
トピック	メッセージを送信する時、MQTT サーバーが受け取れたトピックです。変数をトピックとして使用することができます。使用方法：トピック欄で%(DYNAMIC)を入力すれば、下側に動的ストリングアドレスの設定欄が現れます。%(DYNAMIC)のストリングに複数の topic level を含めることができます。例：myhome/groundfloor。



クラウドサービスに Azure IoT Hub を選択した場合、トピックは固定したフォーマットで、ユーザーは最後の level だけを指定可能です。



送信モード

アドレス(自動)

数値トリガー式 :

数値に変化があるたびに、MQTT メッセージを送信します。

周期ベース式 :

定期的に MQTT メッセージを送信します。

アドレス(ビットトリガー)

指定したビットがトリガーされたたびに、MQTT メッセージを送信します。

イベント(アラーム)ログ

イベントログをトピックのソースとして使用可能です。単一のイベント、もしくは指定したカテゴリーでの任意のイベントがトリガーされた場合 MQTT メッセージを送信するのを選択できます。

圧縮タイプ

メッセージを圧縮してから転送します。圧縮されたメッセージを読み取る前、先に解凍する必要があります。メッセージを圧縮/解凍するには、zlib、gzip または DEFLATE 算法が選べます。

メッセージを保持する

本項にチェックマークを入れると、MQTT サーバーは最新のメッセージを保持します。

タイムスタンプを含む

内容フォーマットに[JSON(一般)]を使用した場合のみ、本機能がサポートされます。手動でタイムスタンプを含むかを決定できます。

最上層の"#"記号をメ

内容フォーマットに[JSON(一般)]を使用した場合のみ、本機

メッセージフォーマットに使用する

能がサポートされます。チェックマークを入れると、メッセージフォーマットは以下の通りになります：

```
{
  "d": {
    "addressName1": ...,
    "addressName2": ...
  },
  "ts": ...
}
```

チェックマークを入れないと、メッセージフォーマットは以下の通りになります：

```
{
  "addressName1": ...,
  "addressName2": ...,
  "ts": ...
}
```

上図に示された通り、チェックマークを入れていない場合、**ts** とアドレス名は同じ階層のキー(key)ですので、アドレス名を **ts** に命名することを避けてください。

QoS

MQTT は 3 レベルの信頼性を提供します。即ち QoS(キューオーエス)のことです。メッセージ配布の信頼性はメッセージが確実に届くのかを決めます。

0: 最高一回：メッセージを一回だけ配布するが、確実に届くかは保証しません。

1: 最低一回：メッセージは最低一回届きます。

2: 正確に一回：メッセージは正確に一回届きます。

内容フォーマット

Raw data : BYTE で組み合わせたデータです。

JSON(一般)：全てのデータをメンバー "d" に置く JSON フォーマットです。

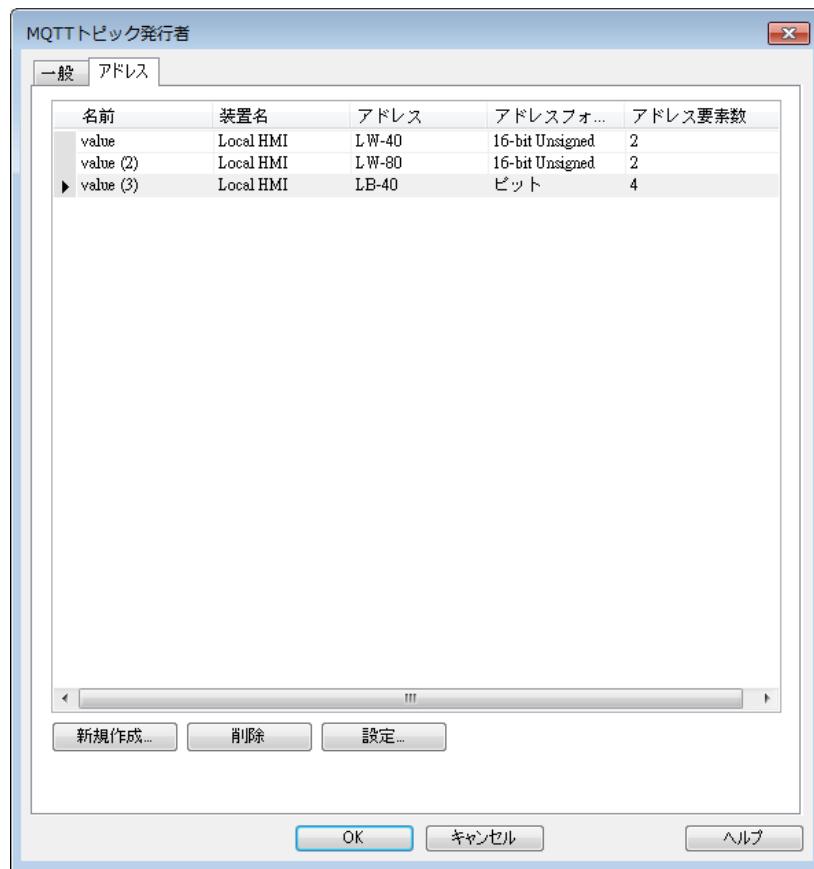
JSON(高度)：フレキシブルなネスト構造があるユーザ一定義 JSON フォーマットです。

Note

- 一个の tag には最大 255 個の word を使用できます。

アドレス設定

本節では、内容フォーマットに[Raw Data]と[JSON(一般)]を使用する場合、アドレスの設定方法について説明します。

**設定****記述****新規追加**

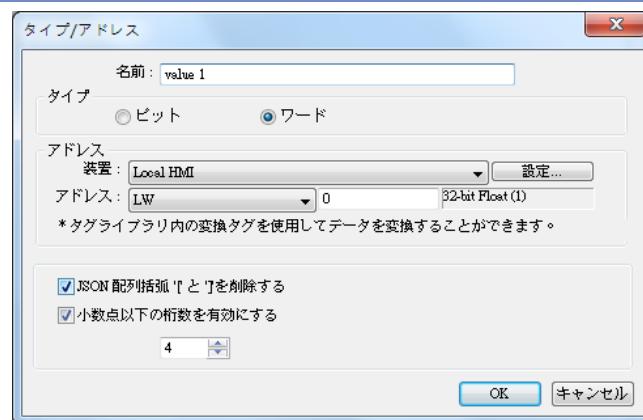
トピックのアドレスソースを作成します。各アドレスの長さを個別に設定することができます。

削除

アドレスを削除します。

設定

アドレス及び名前を変更します。

**設定****記述****JSON配列括弧 '['と']'を削除する**

本項にチェックマークを入れると、JSONフォーマットのメッセージを使用する場合、手動で括弧を削除することができます。

小数点以下の桁数を有効にする

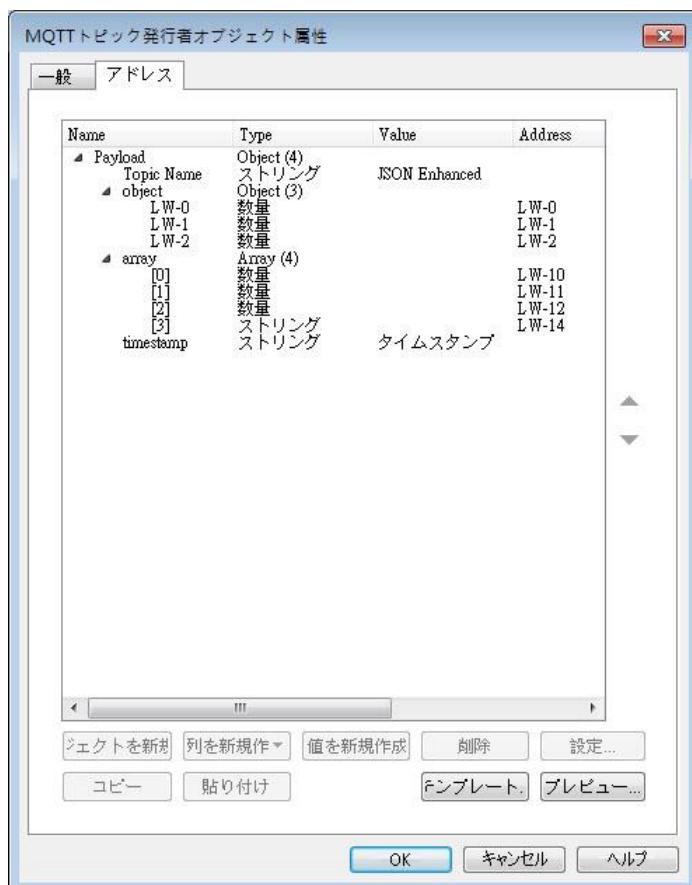
アドレスのデータ型を Float 浮動小数点数に設定する場合、小数点以下の桁数を選択することができます。

 Note

- tag の長さは最大 255 word です。

アドレス設定[JSON(高度)]

本節では、内容フォーマットに[JSON(高度)]を使用する場合、アドレスの設定方法について説明します。[JSON(高度)]はネスト構造をサポートし、オブジェクト、配列などのフォーマットを使用可能で、タイムスタンプ及びデータ名もカスタマイズでき、フレキシブルなデザインを実現できます。



上図を例に挙げると、上図のように設定すれば、購読側では以下のフォーマットの MQTT メッセージが届きます。

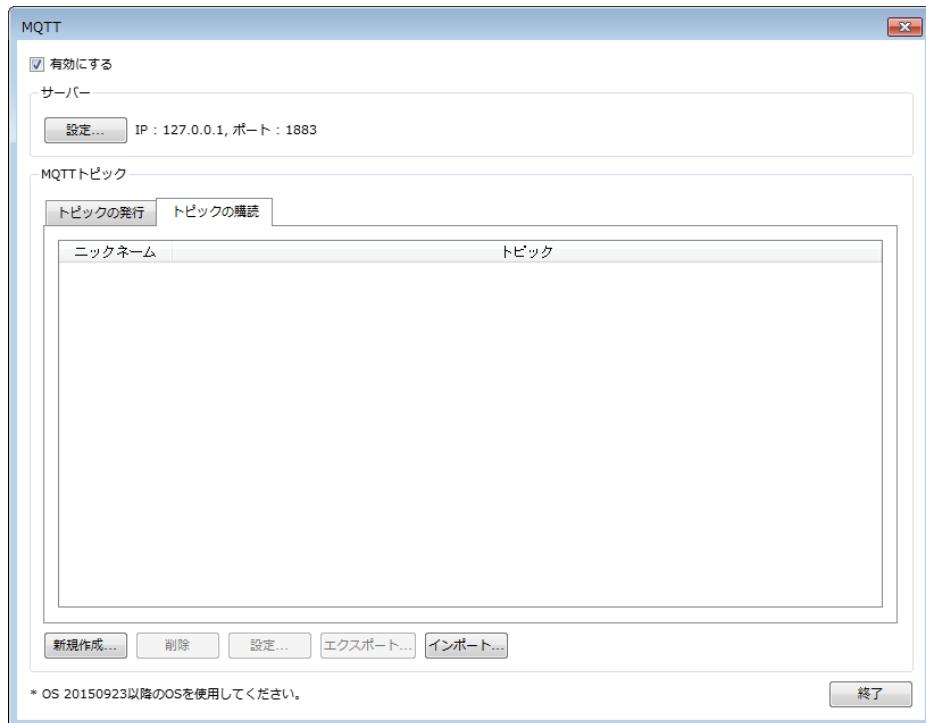
```
{
  "Topic Name" : "JSON Enhanced",
  "Object" : {
    "LW-0" : [ 1 ],
    "LW-1" : [ 2 ],
    "LW-2" : [ 3 ]
  },
  "Array" : [ [ 4 ], [ 5 ], [ 6 ], [ "AABBCCDD" ] ],
  "timestamp" : "2019-02-19T06:52:13.846038"
}
```

設定	記述
オブジェクトを新規追加	一個のデータオブジェクトを新規追加します。オブジェクトでは複数のデータフォーマットを含むことができ、各データフォーマットでは独自の名前と数値があります。オブジェクトのデータを大括弧{}で包括します。
配列を新規追加	一個のデータ配列を新規追加します。配列の中には複数のデータフォーマットを含むことができますが、名前は一個だけしかありません。オブジェクトのデータを中括弧[]で包括します。
数値を新規追加	一個の数値、ストリングまたはタイムスタンプを新規追加します。数値またはストリングである場合、固定した数値を使用、または指定のアドレスからデータを読み取ることができます。
削除	選択した欄を削除します。
設定	選択した欄を変更します。選択した欄はオブジェクト及び配列の場合、名前だけを変更できます。但し、オブジェクトと配列に包括された数値(例えば[数値を新規追加]のパラメータ)を変更できます。
コピー	選択した欄をコピーします。
貼り付け	選択した欄でコピーした内容を貼り付けます。
テンプレート	ここで JSON 文字を貼り付ければ、システムは自動的に本 JSON フォーマットに合致する構成を作成し、自ら定義する手間が省けます。
プレビュー	読みやすいフォーマットで JSON データをプレビューします。

Note

- 一个の Topic には最大 512 個のノード(payload を含む)を使用できます。一个の tag には最大 255 個の word を使用できます。

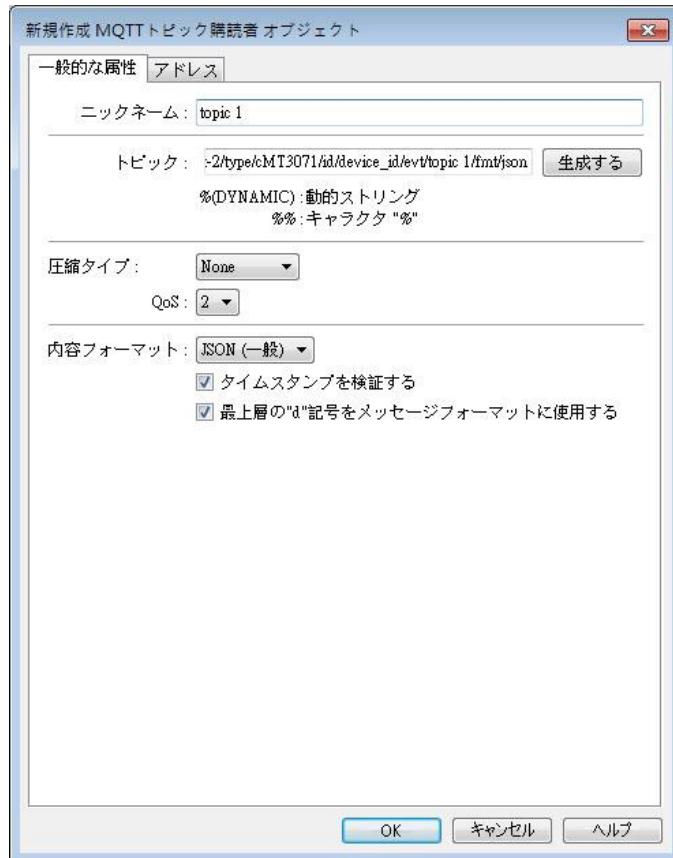
42.1.2.3. MQTT トピックの購読



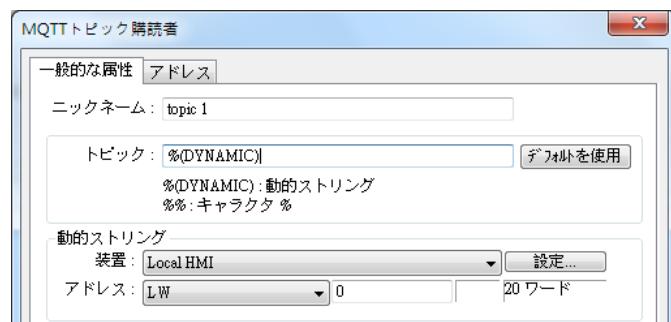
[新規作成]を選択すれば、一般的な属性及びアドレスの設定に入ることができます。または直接に CSV ファイルを[エクスポート]/[インポート]する機能で MQTT 購読トピックを作成してもいいです。MQTT 購読トピックの数量上限は 255 です。

一般的な属性の設定

本節では、内容フォーマットに[Raw Data]と[JSON(一般)]を使用する場合、アドレスの設定方法について説明します。



設定	記述
ニックネーム	MQTT トピックの項目名を設定します。
トピック	MQTT サーバーから購読するトピックです。動的ストリングで購読してもいいです。使用方法: トピック欄で%(DYNAMIC)を入力すれば、下側に動的ストリングアドレスの設定欄が現れます。%(DYNAMIC)のストリングに複数の topic level を含めることができます。例: myhome/groundfloor。



クラウドサービスに Azure IoT Hub を選択した場合、トピック

クは固定したフォーマットで、ユーザーは最後の **level** だけを指定可能です。このトピック **level** はトピック発行者の設定と同じでなければなりません。

圧縮タイプ

ここは発行者での設定と同じでなければなりません。

タイムスタンプを検証する

本項にチェックマークを入れると、メッセージのタイムスタンプを検証することになります。タイムスタンプが増加する場合のみメッセージが更新され、そうでないと、メッセージは古いものと判断され、更新されません。

最上層の"d"記号をメッセージフォーマットに使用する

チェックマークを入れると、メッセージフォーマットは以下の通りになります：

```
{
  "d": {
    "addressName1": ...,
    "addressName2": ...
  },
  "ts": ...
}
```

チェックマークを入れないと、メッセージフォーマットは以下の通りになります：

```
{
  "addressName1": ...,
  "addressName2": ...,
  "ts": ...
}
```

データソースに基づいて適する設定を選択してください。

QoS

MQTT は 3 レベルの信頼性を提供します。即ち **QoS**(キューオーエス)のことです。メッセージ配布の信頼性はメッセージが確実に届くのかを決めます。

0: 最高一回：メッセージを一回だけ配布するが、確実に届くかは保証しません。

1: 最低一回：メッセージは最低一回届きます。

2: 正確に一回：メッセージは正確に一回届きます。

内容フォーマット

Raw data : 特定したフォーマットが無い原始データです。

JSON (一般) : 単層構造の JSON フォーマットです。

JSON (高度) : フレキシブルにネスト構造を定義できる JSON フォーマットです。

アドレス設定



設定

記述

新規追加

トピックの配布先のアドレスを追加します。各アドレスの長さを個別に設定することができます。

削除

アドレスを削除します。

設定

アドレス及び名前を変更します。



設定

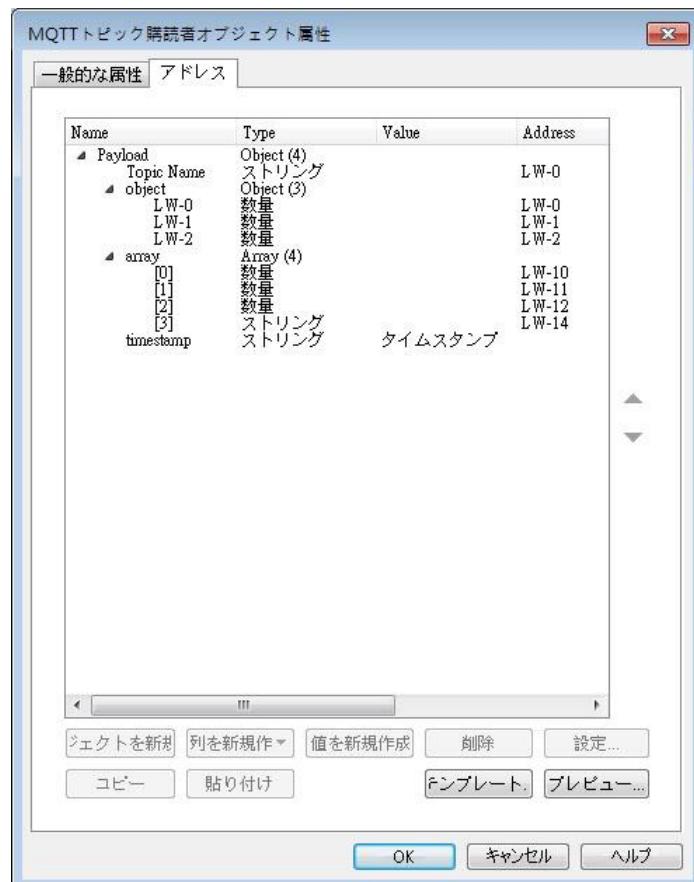
記述

JSON 配列括弧 '['と']'を削除する

本項にチェックマークを入れると、JSON フォーマットのメッセージを使用する場合、手動で括弧を削除することができます。

アドレス設定[JSON(高度)]

本節では、内容フォーマットに[JSON(高度)]を使用する場合、アドレスの設定方法について説明します。[JSON(高度)]はネスト構造をサポートし、オブジェクト、配列などの形式を使用でき、タイムスタンプ及びデータ名を定義でき、よりフレキシブルにデザインできます。



設定	記述
オブジェクトを新規追加	一個のデータオブジェクトを新規追加します。オブジェクトでは複数のデータフォーマットを含むことができ、各データフォーマットでは独自の名前と数値があります。オブジェクトのデータを大括弧{}で包括します。
配列を新規追加	一個のデータ配列を新規追加します。配列の中には複数のデータフォーマットを含むことができますが、名前は一個だけしかありません。オブジェクトのデータを中括弧[]で包括します。
数値を新規追加	一個の数値、ストリングまたはタイムスタンプを新規追加します。数値またはストリングである場合、固定した数値を使用、または指定のアドレスからデータを読み取ることができます。

削除	選択した欄を削除します。
設定	選択した欄を変更します。選択した欄はオブジェクト及び配列の場合、名前だけを変更できます。但し、オブジェクトと配列に包括された数値(例えば[数値]を新規追加)のパラメータ)を変更できます。
コピー	選択した欄をコピーします。
貼り付け	選択した欄でコピーした内容を貼り付けます。
テンプレート	ここで JSON 文字を貼り付ければ、システムは自動的に本 JSON フォーマットに合致する構成を作成し、自ら定義する手間が省けます。

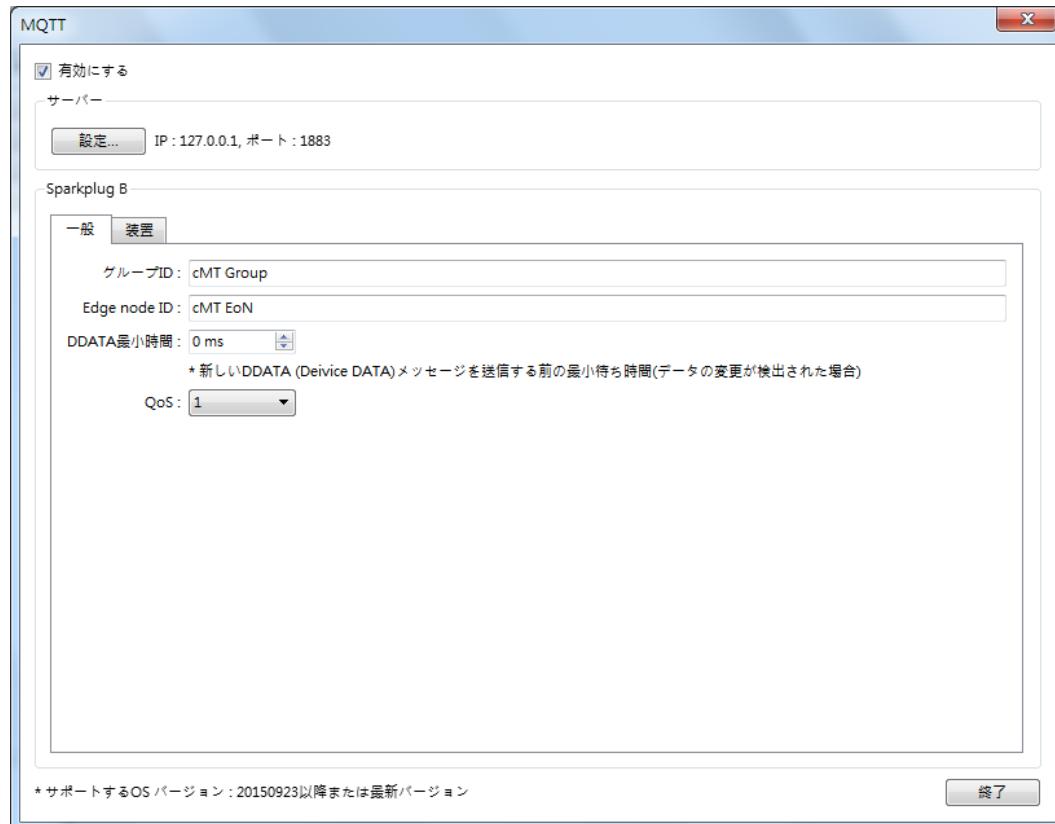
Note

- Amazon Web Service(AWS) IoT Core は、標準の MQTT プロトコルをサポートします。使用する場合、下記の事項をご注意ください：
 1. トピックは最大 8 階層まで(iot-2/type は 2 階層)です。
 2. [一般的な属性]タブでの検証をサポートしません。検証するには、[TLS/SSL]タブで行ってください。
 3. QoS 0 と QoS 1 のみをサポートします。
 4. トピック発行の[メッセージを確保する]機能をサポートしません。

42.1.2.4. Sparkplug B

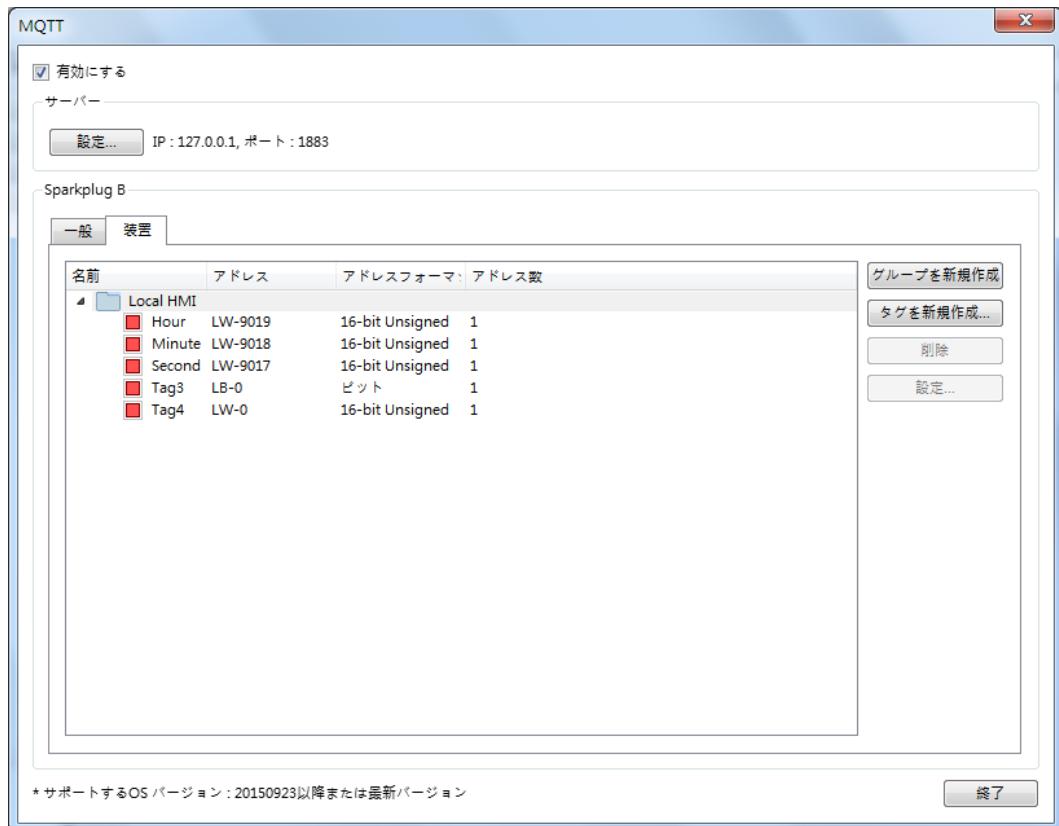
クラウドサービスに Sparkplug B を選択する場合、一般的な属性及び装置の設定方法は以下の通りです。

一般的な属性の設定



設定	記述
グループ ID	本 Edge Of Network Nodes が属するグループを識別するための ID です。
Edge node ID	本 Edge Of Network Node を識別するための ID です。
DDATA 最小時間	データに変化があったと検知すると、新しい DDATA (Device DATA) メッセージを送信する前の最小待ち時間です。
QoS	MQTT は 3 レベルの信頼性を提供します。即ち QoS(キューオーエス)のことです。メッセージ配布の信頼性はメッセージが確実に届くのかを決めます。 0: 最高一回 : メッセージを一回だけ配布するが、確実に届くかは保証しません。 1: 最低一回 : メッセージは最低一回届きます。 2: 正確に一回 : メッセージは正確に一回届きます。

装置の設定



設定	記述
グループを新規作成	タグを管理するため、グループを新規作成します。
タグを新規作成	本 EoN Node の MQTT Engine に監視されるタグを新規作成します。空白ではいけません。
削除	既存のグループまたはタグを削除します。
設定	既存のグループまたはタグを設定します。



このアイコンをクリックし、サンプルプロジェクトをダウンロードしてください。サンプルプロジェクトをダウンロードする前、インターネットケーブルが接続しているのを確認してください。

42.2. OPC UA サーバー

42.2.1. 概要

OPC UA(Unified Architecture)はファクトリーオートメーション業界での通信規格です。データの通信がプラットフォームに限られない、アクセス機構が統一、通信の標準化及びセキュリティ認証機構などの特長を持っています。cMT シリーズ HMI は OPC UA のサーバーの役目に相当し、OPC UA クライアント(Client)ソフトウェアで HMI、或いは PLC 上のアドレスタグ情報を読み取ることで、情報の垂直統合が求められます。

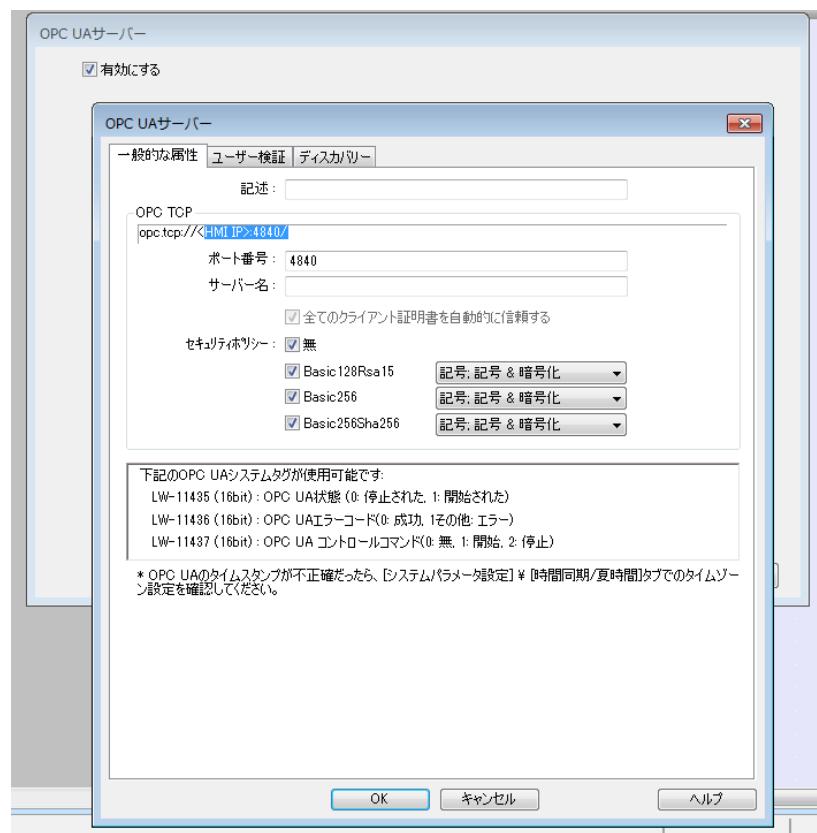
ソフトウェア・ハードウェア要件：

- 対応機種：cMT シリーズ。cMT-SVR / cMT-SVR-200 及び cMT-HDM / cMT-FHD は別途ライセンスをロードする必要があります。
- ソフトウェア：Easy Builder Pro v5.06.01 以降
- 推奨 OPC UA クライアント：Unified Automation UaExpert

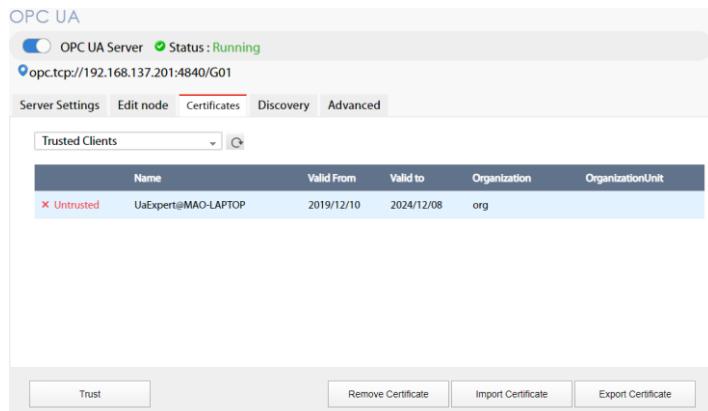
42.2.2. 設定

ツールバーの[オブジェクト]をクリックし、そして[IIoT] » [OPC UA サーバー]をクリックして本オブジェクトを新規作成してください。有効にしてから、設定ダイアログボックスが現れます。

一般的な属性



設定	記述
記述	本オブジェクトに対する記述です。
OPC TCP	サーバーの URL ウェブアドレスです。
ポート番号	クライアント側に接続させるためのポート番号を設定します。デフォルトは 4840 です。
サーバー名	サーバー名を設定します。空白にしてもいいです。 全てのクライアント証明書を自動的に信頼する 本項はデフォルトで有効にされ、cMT Gateway でのみ無効にすることができます。本項を有効にしていない場合、下図に示されたとおり、全ての OPC UA クライアントは OPC UA ウェブインターフェースで「信頼する証明書」に設定されていない限り、接続が全部拒否されます。
セキュリティポリシー	OPC UA が提供するセキュリティポリシーと、クライアントで選択可能な暗号化演算方法です。

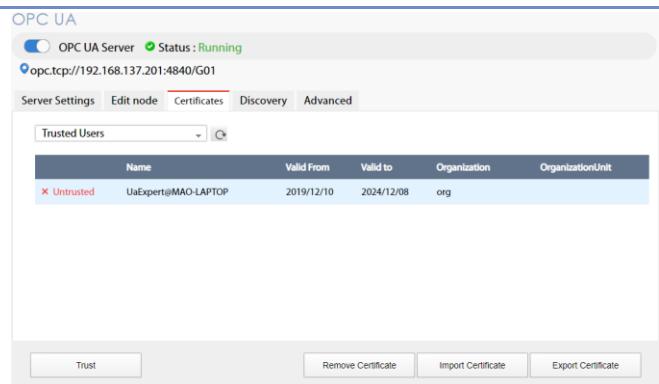


注意：OPC UA の規則により、セキュリティポリシーに [無]を選択した場合以外、OPC UA クライアントが OPC UA サーバーに接続するにはクライアント証明書を使用する必要があります。その証明書は OPC UA サーバーに正当性を検査されます。

ユーザー検証

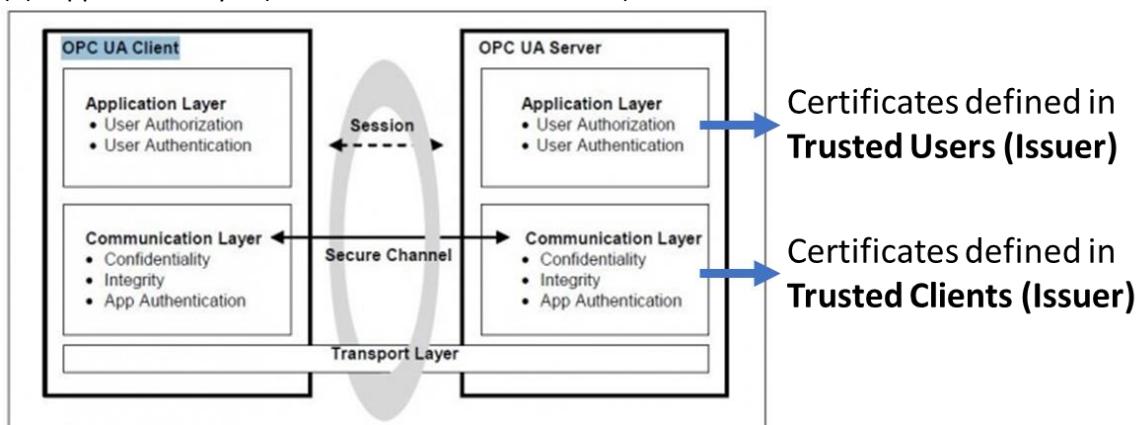


設定	記述
方式	<p>匿名 クライアントソフトウェアが匿名でログインした時、データアクセス(閲覧/読み取り/書き込み)の権限を設定します。</p> <p>ユーザー名&パスワード HMI のユーザー名&パスワードと共有しています。 クライアントソフトウェアでログインした後、データアクセスの権限はレベルで分けられます。</p> <p>証明書 本項は cMT Gateway にのみ対応しています。OPC UA クライアントはユーザー名&パスワードでなく、証明書でログインすることができます。ウェブインターフェースで信頼する/しない証明書を設定します。下図を参考してください：</p>



Note

- OPC UA のセキュリティレイヤーは以下のように分けられています：
 - (1) Communication Layer (通信レイヤー、例えば：セキュリティ)
 - (2) Application Layer (アプリケーションレイヤー)、下図をご参考ください：



セキュリティレイヤーの詳細については、次のリンクの説明をご参考ください：

(<http://wiki.opcfoundation.org/index.php/File:SecurityLayers.jpg>)

- クライアント検証は通信レイヤーに位置します。[無]以外のセキュリティポリシーを使用した場合、証明書が必須です。
- ユーザー検証はアプリケーションレイヤーに位置します。ログインの方法の 1 つとして、ユーザー証明書を使用してログインすることができます。

ディスカバリー(Discovery)設定



ネットワーク内で複数の OPC UA サーバーがある場合、OPC UA サーバーをディスカバリーサーバー(Discovery Server)に登録すれば、OPC UA クライアント側で Local Discovery Server (略称 LDS) を通じて OPC UA サーバーを探し出すことができます。

設定	記述
IP	OPC UA クライアント側の IP です。
ポート番号	OPC UA Client 側のポート番号です。
サーバー名	OPC UA Client 側のサーバー名です。
記述	備考として使用され、通信に影響を与えません。

例

以下の手順でどのようにディスカバリー機能を設定すればいいかを説明します：

- まず、Local Discovery Sever (以下は LDS と称する) をインストールします。以下のウェブページで OPC UA 協議会が提供した LDS ファイルをダウンロードすることができます。

<https://opcfoundation.org/developer-tools/developer-kits-unified-architecture/local-discovery-server-ls/>

他のサードパーティーLDS ファイルを使用することも可能ですが。

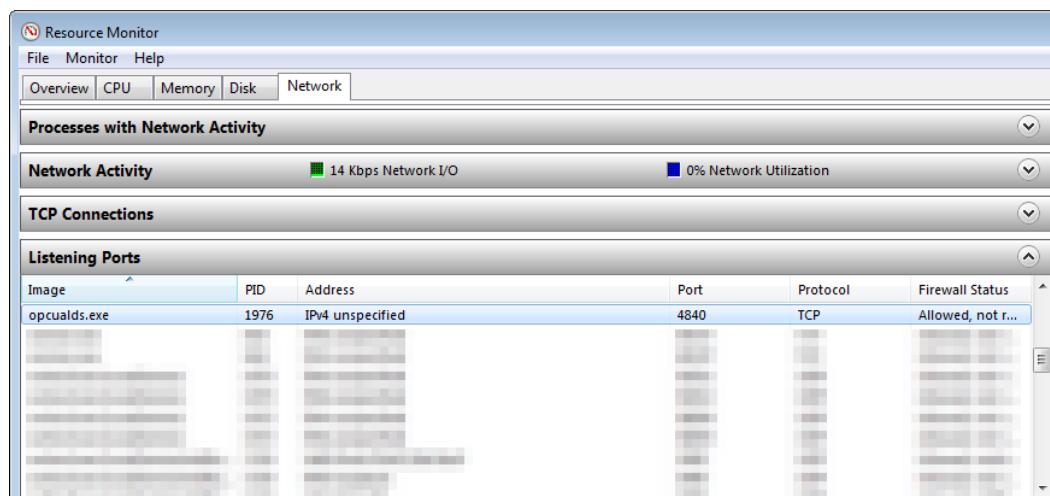
- HMI が接続したルーターが HMI Name を識別できなかったら、HMI の名前を現在 HMI の IP

アドレス(もしくは 0.0.0.0 に変更してから、own certificate を再作成)に変更してください。
例えば: HMI の IP は 192.168.1.100 の場合、HMI の名前を 192.168.1.100 或いは 0.0.0.0 にしてください。

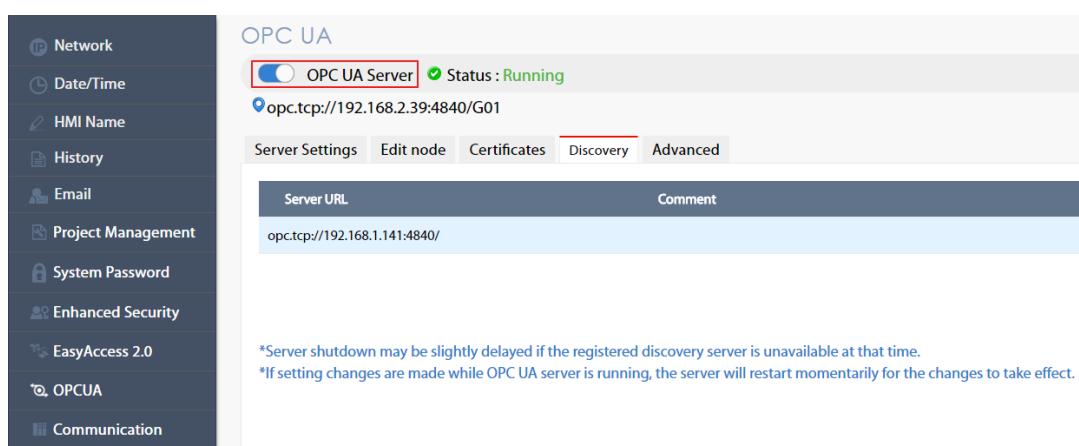
3. OPC UA サーバー実行後、C:\ProgramData\OPC Foundation\UA\pki\rejected\certs 内の Certificate を C:\ProgramData\OPC Foundation\UA\pki\trusted\certs にコピーしてください。
4. OPC UA クライアントソフトウェアを起動し、HMI の IP を入力するか、LDS Endpoint を使用することでディスカバリー機能を実行すれば、迅速に HMI の OPC UA サーバーに接続することができます。

ディスカバリー機能が正常に使用されない場合、以下の項目を確認してください：

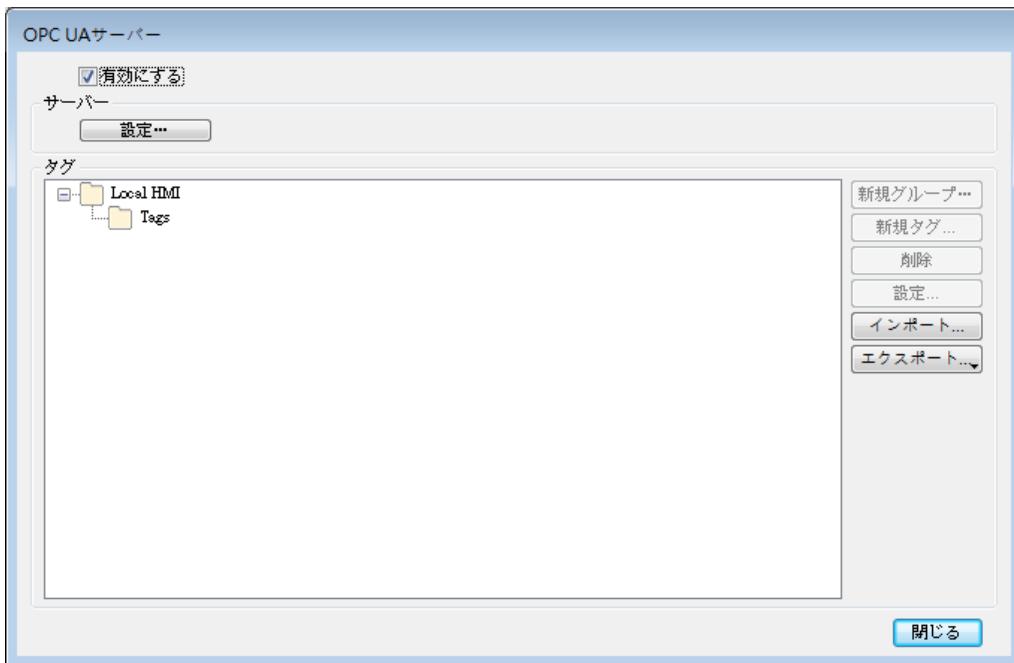
1. タスク マネージャーの起動 » [パフォーマンス] » [リソース モニター] » [ネットワーク] » [リッスンポート]で opcualds.exe が使するポート番号を確認します。下図に示されたとおり、現在この PC の opcualds.exe が使用しているポート番号は 4840 です。



2. ウェブブラウザで HMI の IP を入力し、そしてパスワードを入力してログインします。[OPC UA] 設定ページで改めて OPC UA サーバーを起動します。注: 本 OPC UA タブは、cMT Gateway にのみサポートされます。注: 本 OPC UA タブは cMT Gateway にのみサポートされます。



タグ設定



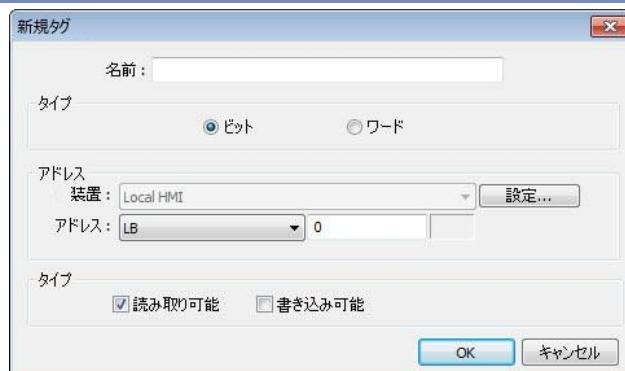
設定

新規グループ



タグを管理するために、グループを新規追加します。

新規タグ



クライアント側で監視・操作するタグを新規追加します。ここで本アドレスに書き込めるかを選択し、書き込む名前は空白にしてはいけません。

履歴(HDA)

OPC UA の HDA 機能を有効にします。

削除

既に存在しているグループ、或いはタグを削除します。

設定	既に存在しているグループ、或いはタグを設定します。
インポート	前に作成したタグをインポートします。*.xlsx、*.xls、*.csv、*.xml ファイルをインポートできます。
エクスポート	現在作成したタグをエクスポートします。Excel、または XML フォーマットにエクスポートできます。



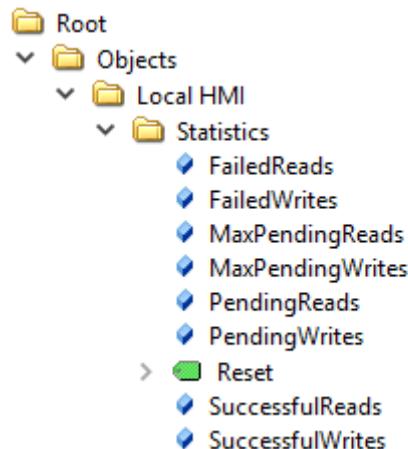
Note

- プロジェクトを HMI にダウンロードする前、まずは HMI の時刻及びタイムゾーンの設定が正確なのかを確認してください。それはクライアントソフトウェアが通信する際、検証時刻エラーで検証に失敗し、OPC UA サーバーに接続できないのを避けるためです。

▶ このアイコンをクリックし、チュートリアルビデオを閲覧してください。閲覧する前に、インターネットケーブルが接続しているのを確認してください。

42.2.3. 装置統計資料

装置の通信統計資料は、各装置 Statistics ノードで見つかります。下図をご参照ください：



各ノードの意味は以下の通りです：

ノード名	意味
FailedReads	失敗した読み取りコマンドの個数です。0 でない場合、通信に失敗した可能性があります。
FailedWrites	失敗した書き込みコマンドの個数です。0 でない場合、通信に失敗した可能性があります。
MaxPendingReads	発生した最大の読み取り待ちのコマンドの個数です。
MaxPendingWrties	発生した最大の書き込み待ちのコマンドの個数です。
PendingReads	読み取り待ちのコマンドの個数です。数値が長時間である程度の数量を維持することは、通信モジュールが全てのコマンドを処理しきれないと示しています。これで OPC UA ノードの更新を遅延させる可能性があります。極端の状況下、例えば長時間で 30 以上を維持する場合、OPC UA ノードがその期間内

	で更新されない可能性があります。
PendingWrites	書き込み待ちのコマンドの個数です。書き込みコマンドは読み取りコマンドより優先順位が高いので、本数値が長時間で高い場合、読み取りコマンドを影響します。
Reset	統計資料をリセットします。
SuccessfulReads	読み取りに成功したコマンドの個数です。
SuccessfulWrites	書き込みに成功したコマンドの個数です。

42.2.4. サポート及び制限

以下で OPC UA サーバーがサポートする機能及び制限を記します。

項目	記述
OPC UA Profile	Standard UA Server Profile には以下を含むが、これらに限定されません： * Core Server Facet * UA-TCP UA-SC UA-Binary * SecurityPolicy – None * Enhanced DataChange Subscription Server Facet * Standard DataChange Subscription Server Facet * Embedded DataChange Subscription Server Facet * User Token – X509 Certificate Server Facet * User Token – User Name Password Server Facet * Standard DataChange Subscription Server Facet * Embedded DataChange Subscription Server Facet 関連情報は Profile Reporting Visualization Tool by OPC Foundation をご参照ください。
Security policies	None Basic128Rsa15 Basic256 Basic256Sha256
最大 OPC UA ノード数	15 000
單一ノード最大配列長さ	255
読み取りキャッシュ	100ms (キャッシュは 100ms まで維持し、その後は改めて読み取る)
Client Session 数量	100

単一の Client Session が使用できる Subscription 個数	64
Publishing Interval 最小値	100ms
OPC UA HDA	<p>*最大 50 個までのノードアドレスをサポートします。</p> <p>*各ノードアドレスは 10000 レコードのデータを記録できます。</p> <p>ノードアドレスの定義： 各 HDA を有効にしているノードは、その長さと同じのノードアドレスを使用すると視されます。データ型がストリングの場合、そのワード数に当たります。</p> <p>*HMI メモリーの空き容量が 10%未満になった場合、新しいデータを保存するために、システムは空き容量が 10% 以上になるまで、最初の HDA データから削除し始めます。</p>
パフォーマンス	
最大読み取りスループット (Security: None)	<p>内蔵レジスタ(例 : LW): 27000 words/second (WPS) MODBUS RTU@9600bps: 500 WPS MODBUS RTU@115200bps: 4000 WPS MODBUS TCP/IP: 10000 WPS</p> <p>テスト環境 EBPro version: V6.02.02.242 cMT-G02 OS version: 20180917 テストする時に、配列ノードを使用して読み取りの効率を最適化します。</p>

Note

OPC UA HDA ノードアドレスの例 :

仮に 50 個のノード(node1、node2、...、node 50)があり、各ノードが長さ=1 の bit アドレスにマッピングする場合、合計 50 個のノードアドレスを使用することになります。

もし 1 個のノードが長さ=50 の 16-bit Unsigned 整数配列(長さが 50 に設定された)にマッピングする場合、その配列の構造要素は個別に 1 個のノードアドレスに視され、当該ノードは 50 個のノードアドレスを使用することになります。

1 個のノードがワード数=50 のストリングにマッピングする場合、当該ノードは 50 個のノードアドレスを使用することになります。

付録A. 各シリーズ HMI のソフトウェア の相違点

- eMT: eMT3070B, eMT3105P, eMT3120A, eMT3150A
- cMT シリーズ: cMT-SVR-100, cMT-SVR-102, , cMT-SVR-200, cMT-SVR-202, cMT-iV5/iV6, cMT3151, cMT3090, cMT3071, cMT3072, cMT3103, cMT-HDM, cMT-FHD, cMT-FHDX-220
- cMT X シリーズ: cMT3092X, cMT3152X, cMT3072X, cMT3072XH, cMT3161X, cMT3162X
- cMT シリーズゲートウェイ: cMT-G01, cMT-G02, cMT-G03, cMT-G04, cMT-CTRL01
- mTV シリーズ: mTV-100
- iE シリーズ: MT8050iE, MT8053iE, MT6070iE, MT6071iE, MT8070iE, MT8071iE, MT8073iE, MT8100iE, MT8101iE, MT8102iE, MT8103iE
- iER シリーズ: MT8070iER1
- XE シリーズ: MT8090XE, MT8091XE, MT8092XE, MT8121XE, MT8150X
- iP シリーズ: MT6051iP, MT6071iP, MT8071iP, MT6103iP, MT8102iP

以下の表は参考用であり、必ず全ての機能が含まれているわけではありません。一部の機能はソフトウェアのバージョンに依存していますので、疑問に思われる点がある場合、ご使用のソフトウェアを再確認するか、Weintek の担当者までお問い合わせください。

iP/iE/eMT/mTV/XE シリーズ

カテゴリ	機能	機種						注
		iP	iE	eMT	mTV	XE		
EasyBuilder	バックアップメモリ	16MB	22MB	64MB	64MB	64/120MB*	1	
	アプリケーションメモリ	22.5MB	22.5MB	64MB	64MB	64MB		
	USB ディスク経由でダウンロード	有り	有り	有り	有り	有り		
機能	64-bit データ型	無し	無し	無し	無し	無し		
	3G/4G ドングル	無し	無し	無し	無し	無し	2	
	オーディオ(cMT Viewer)	無し	無し	無し	無し	無し		
	制御トークン	無し	無し	無し	無し	無し		
	E メール	無し	有り	有り	有り	有り		
	高度セキュリティモード	無し	有り	有り	有り	有り		

カテゴリ	機能	機種	iP	iE	eMT	mTV	XE	注
ハードウェア	イーサネットプリンター		無し	無し	無し	無し	無し	
	レシピデータベース		無し	有り	有り	有り	有り	
	ストリングテーブル		無し	有り	有り	有り	有り	
	時刻同期		無し	有り	有り	有り	有り	
	USB テザリング		無し	有り	有り	有り	有り	3
	ユーザー定義起動画面		有り	有り	有り	有り	有り	
	VNC Server		有り	有り	有り	有り	有り	4
	ウェブストリーミング		無し	無し	有り	有り	有り	5
	ウェブインターフェース		無し	無し	無し	無し	無し	
	Web View		無し	無し	無し	無し	無し	
オブジェクト	音声出力		無し	無し	eMT3120 eMT3150	via HDMI	via USB	11
	CAN Bus		無し	無し	有り	無し	8091XE 8092XE	
	Wi-Fi		無し	8103iE	無し	無し	無し	
	バイブルレーション		無し	無し	無し	無し	無し	
機能	動作トリガー		無し	無し	無し	無し	無し	
	BACnet Schedule		無し	無し	無し	無し	無し	
	バーコードスキャナー (Android カメラ)		無し	無し	無し	無し	無し	6
	複合ボタン		無し	有り	有り	有り	有り	
	日付/時刻		有り	有り	有り	有り	有り	
	動的作画		無し	有り	有り	有り	有り	
	動的スケール		無し	有り	有り	有り	有り	
	ファイルブラウザ		無し	有り	有り	有り	有り	
	フローブロック		無し	有り	有り	有り	有り	
	メディアプレーヤー		無し	無し	eMT3105 eMT3121 eMT3150	無し	有り	7
	PDF リーダー		無し	無し	無し	無し	無し	
	画像閲覧		無し	有り	有り	有り	有り	

カテゴリー	機能	機種	iP	iE	eMT	mTV	XE	注
オブジェクト-IIoT/エネルギー	円グラフ	無し	有り	有り	有り	有り		
	テーブル	無し	有り	有り	有り	有り	有り	
	タッチジェスチャー	無し	無し	無し	無し	無し	無し	
	映像入力 - ネットワークカメラ	無し	無し	有り	無し	有り	8	
	映像入力 - USB カメラ	無し	無し	有り	有り	有り	有り	
	映像入力 - 映像入力	無し	無し	eMT3121 eMT3150	無し	無し	無し	
	VNC Viewer	無し	有り	有り	有り	有り	9	
オブジェクト-データ/履歴	エネルギー需要量表示	無し	無し	無し	無し	無し	無し	
	エネルギー需要量設定	無し	無し	無し	無し	無し	無し	
	MQTT	無し	有り	有り	有り	有り	有り	
	MQTT - AWS IoT, Sparkplug B, Azure IoT Hub, Google Cloud IoT Core	無し	無し	無し	無し	無し	無し	
	OPC UA クライアント	無し	有り	有り	有り	有り	有り	
	OPC UA サーバー	無し	無し	無し	無し	無し	無し	
他	近似曲線円グラフ	無し	有り	有り	有り	有り	有り	
	データベースサーバー	無し	無し	無し	無し	無し	無し	
	イベントバーチャート	無し	無し	無し	無し	無し	無し	
	操作ログ設定	無し	有り	有り	有り	有り	有り	
	レシピデータベース	無し	有り	有り	有り	有り	有り	
	レシピのインポート/エクスポート	無し	有り	有り	有り	有り	有り	
	レシピ閲覧	無し	有り	有り	有り	有り	有り	
	SQL クエリ	無し	無し	無し	無し	無し	無し	
ユーティリティ	CODESYS	無し	無し	無し	無し	無し	無し	
	cMT Diagnoser	無し	無し	無し	無し	無し	無し	
	EasyAccess 1.0	有り	有り	有り	有り	有り	有り	
	EasyAccess 2.0	無し	有り	有り	有り	有り	有り	
EasyAccess 2.0(ライセンス内蔵)			8053iE 8073iE 8103iE	無し	無し	8092XE		

カテゴリー	機能	機種						注
		iP	iE	eMT	mTV	XE		
EasyDiagnoser EasyLauncher EasyPrinter EasySimulator EasySystemSetting EasyWatch	EasyDiagnoser	有り	有り	有り	有り	有り		
	EasyLauncher	無し	無し	無し	無し	無し		
	EasyPrinter	有り	有り	有り	有り	有り		
	EasySimulator	有り	有り	有り	有り	有り		
	EasySystemSetting	有り	有り	有り	有り	有り	10	
	EasyWatch	有り	有り	有り	有り	有り		

cMT シリーズ

カテゴリー	機能	機種										注	
		cMT-SVR	cMT3071 cMT3072	cMT3090	cMT3103	cMT3151	cMT-HDM cMT-FHD	cMT-G01 cMT-G02 cMT-G03 cMT-G04 cMT-CTRL01	cMT3092X cMT3152X cMT3162X	cMT-FHDX-220	cMT3072X cMT3072XH	cMT3161X	
EasyBuilder	バックアップメモリ	*	*	*	*	*	*	*	*	*	*	*	1
	アプリケーションメモリ	32MB	64MB	64MB	64MB	64MB	64MB	32MB	64MB	64MB	64MB	64MB	
	USB ディスク経由でダウンロード	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
機能	64-bit データ型	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	3G/4G ドングル	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	2
	オーディオ(cMT Viewer)	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	制御トークン	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	E メール	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	高度セキュリティモード	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	イーサネットプリンター	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	レシピデータベース	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	ストリングテーブル	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	時刻同期	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	USB テザリング	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	3
	ユーザー定義起動画面	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	VNC Server	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	4

カテゴリ	機能	機種	cMT-SVR	cMT3071 cMT3072	cMT3090	cMT3103	cMT3151	cMT-HDM cMT-FHD	cMT-G01 cMT-G02 cMT-G03 cMT-G04 cMT-CTRL01	cMT3092X cMT3152X cMT3162X	cMT-FHD-X-220	cMT3072X cMT3072XH	cMT3161X	注
ハードウェア	ウェブストリーミング	有り	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	5
	ウェブインターフェース	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	WebView	無し	無し	無し	無し	無し	無し	無し	無し	有り	有り	無し	有り	
	音声出力	無し	無し	有り	無し	有り	via HDMI	無し	有り	via HDMI	無し	有り	有り	11
オブジェクト	CAN Bus	無し	有り	有り	有り	有り	無し	無し	有り	無し	有り	無し		
	Wi-Fi	cMT-SVR- 200/202	無し	無し	有り	無し	無し	cMT-G02	無し	無し	無し	無し		
	バイブルーション	無し	無し	無し	無し	無し	無し	無し	cMT3162X	無し	無し	無し		
	動作トリガー	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	無し		
BACnet Schedule	BACnet Schedule	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	無し	
	バーコードスキャナー (Android カメラ)	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	無し		6
	複合ボタン	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	日付/時刻	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	動的作画	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	動的スケール	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	ファイルブラウザ	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	フローブロック	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	メディアプレーヤー	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	有り	7
	PDF リーダー	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	画像閲覧	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	円グラフ	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	テーブル	有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り		
	タッチジェスチャー	無し	無し	無し	無し	無し	無し	無し	cMT3152X cMT3162X	無し	無し	有り		
映像入力	映像入力 - ネットワークカメラ	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	有り	8
	映像入力 - USB カメラ	無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	有り	
	映像入力 - 映像入力	無し	無し	無し	無し	有り	無し	無し	有り	無し	無し	無し		

カテゴリ	機能	機種	cMT-SVR	cMT3071 cMT3072	cMT3090	cMT3103	cMT3151	cMT-HDM cMT-FHD	cMT-G01 cMT-G02 cMT-G03 cMT-G04 cMT-CTRL01	cMT3092X cMT3152X cMT3162X	cMT-FHD-X-220	cMT3072X cMT3072XH	cMT3161X	注
オブジェクト - IIoT/エネルギー	VNC Viewer		無し	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	9
	エネルギー需要量表示		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	無し	
	エネルギー需要量設定		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	無し	
	MQTT		有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	MQTT - AWS IoT, Sparkplug B, Azure IoT Hub, Google Cloud IoT Core		有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	OPC UA クライアント		有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
オブジェクト-データ/履歴	OPC UA サーバー	License	有り	有り	有り	有り	有り	License	有り	有り	License	有り	無し	
	近似曲線円グラフ		無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	
	データベースサーバー		有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	無し	
	イベントバーチャート		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	無し	
	操作ログ設定		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	レシピのインポート/エクスポート		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	無し	
	レシピ閲覧		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	レシピ閲覧		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
他	SQL クエリ		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	CODESYS		無し	License	License	無し	License	無し	cMT-CTRL01	License	License	License	License	
ユーティリティ	cMT Diagnoser		有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	EasyAccess 1.0		無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	
	EasyAccess 2.0		有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	
	EasyAccess 2.0 Built-in License	cMT-SVR-102/202	cMT3071: 無し cMT3072: 有り	有り	有り	有り	無し	無し	cMT3092X cMT3152X	無し	無し	無し	無し	
	EasyDiagnoser		無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	
	EasyLauncher		有り	有り	有り	有り	有り	有り	無し	有り	有り	有り	有り	
	EasyPrinter		無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	

カテゴリ	機能	機種	cMT-SVR	cMT3071	cMT3090	cMT3103	cMT3151	cMT-HDM cMT-FHD	cMT-G01 cMT-G02 cMT-G03 cMT-G04 cMT-CTRL01	cMT3092X cMT3152X cMT3162X	cMT-FHDX-220	cMT3072X cMT3072XH	cMT3161X	注
			cMT3072											
	EasySimulator		無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	無し	
	EasySystemSetting		有り*	有り*	有り*	有り*	有り*	有り*	無し	有り	有り	有り	有り	10
	EasyWatch		有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	有り	

Notes

No.	機能	備考
1	History Size Limit	MT8121/8150XE: 64MB, MT8090/8091/8092XE: 120MB cMT Series: Max. data sampling entries: 40

OS Support

No	機能	OS バージョン
2	3G/4G Dongle	cMT-SVR 20151127 cMT3071/3072/3090/3103/3151: 20180723 or later cMT-HDM/FHD/FHDX: 20190130 or later
3	USB Tethering	iE Type A: 20160707 iE Type B: 20160325 eMT: 20160601 mTV: 20160527 XE: 20160503 cMT-SVR: 20160518 cMT3090/3103/3151/HDM: 20160412 or later
4	VNC Server	cMT3090/3151: 20160412 or later
5	Web Streaming	cMT-SVR: 20151127, cMT3151: 20171129 or later
6	Barcode Scanner Android	Android cMTViewer V1.7.0 or later

7	Media Player	cMT-HDM: no 1920*1080 resolution
8	Video-in IP Camera	eMT: 20160601 XE: 20160503
9	VNC Viewer	iE Type A: 20160707 iE Type B: 20160325 eMT: 20160601 mTV: 20160527 XE: 20160503 all cMT series: 20181211 or later
10	EasySystemSetting	all cMT series: 20180723 or later
11	Audio Output	USB Audio Output OS Support: XE 20190325

付録B. cMT Viewer 使用制限

ローカルの cMT シリーズ HMI を操作するには、直接にモニターで操作するか、他のクライアントで cMT Viewer 或いは VNC Viewer を実行して遠隔操作できます。しかしながら、cMT Viewer クライアントデバイス(iOS, Android, PC, cMT-iV5/iV6)で遠隔接続をする際に、ローカル cMT HMI を操作できるが、一部の機能が正常に実行されない場合があります。プロジェクトをデザインする時、以上の状況を考慮に入れてください。

以下では cMT Viewer で遠隔操作をする場合、正常に実行されない機能を記します：

- ファイルブラウザ
- メディアプレーヤー
- 映像入力：USB カメラ
- 映像入力：ネットワークカメラ
- 映像入力：アナログ入力
- 画像閲覧
- PDF リーダー
- VNC Viewer
- ファンクションキーで選択する機能：スクリーンのハードコードを取る、ユーザー情報をインポート/[USB セキュリティキー]を使用する
- PLC コントロールオブジェクトで選択する機能：
 - I. スクリーンのハードコードを取る：ファイルが HMI に接続された外部保存装置に保存される場合
 - II. バックライトコントロール：HMI モニターのバックライト制御
- 操作ログで選択する機能：
 - I. 操作ログ印刷：ファイルが HMI に接続された外部保存装置に保存される場合

付録C. PC シミュレーション 使用制限

EasyBuiler Pro プロジェクトをデザインする際に、常にオンライン/オフラインシミュレーションを使用してテストする必要があります。しかし、一部の機能は HMI にダウンロードした場合のみテスト・使用可能ですので、プロジェクトをデザインする時に、ご留意ください。

PC でシミュレートする場合、履歴データは EB Pro インストールディレクトリ内の下記フォルダに保存されます：

- HMI_memory : 保存先を HMI メモリに設定した場合。
- SD_card : 保存先を SD カードに設定した場合。
- usb1 : 保存先を USB ディスクに設定した場合。
- usb2 : 保存先を USB ディスク 2 に設定した場合。(一部の機種には 2 個の USB ディスクをサポートします)

以下は PC でのオンライン/オフラインシミュレーションで実行できない機能です：

cMT シリーズ

- メディアプレーヤー
- 映像入力：USB カメラ
- 映像入力：ネットワークカメラ
- 映像入力：アナログ入力
- PDF リーダー

eMT/iE/XE/mTV シリーズ

- 映像入力：USB カメラ
- 映像入力：ネットワークカメラ
- 映像入力：アナログ入力
- VNC Viewer

付録D. 64-bit データ型の使用 においての注意事項

cMT シリーズ機種は 64-bit のデータ型をサポートします。使用制限は以下のとおりです。

- 入力制限(数値オブジェクト、マルチ状態設定...、など)
 - I. uint64: 0 ~ 2⁶⁴
 - II. int64: -2⁶³ ~ 2⁶³
 - III. double: -2⁶⁴ ~ 2⁶⁴

● int64 / uint64

現在、論理計算は double で行われ、double が正確に整数を表示できる分は 52-bit のため、約 15 ~ 16 行が有効な数値と思われ、それを超えると、正確さが低下します。

それに EasyBuilder Pro の使用制限で、一部の入力フォーカスが 48-bit を超えたら異常になる可能性があるので、現時点では一律に 48-bit に制限します。

● double

double 型は元々、EasyBuilder Pro での入力制限は double 自身の最大値/最小値を用いて入力の上下限(+/- 1.79 * 10³⁰⁸)にしていたが、実際にその数値はほとんど上下限が無いため、一律に入力上下限を-2⁶⁴ ~ 2⁶⁴ にします。

例

ユーザーが PLC から数値(64-bit unsigned int) 144,115,188,075,855,872 を読み取ったが、数値オブジェクトで誤って 144,115,188,075,855,870 と表示される可能性があります。

他の状況

PLC 数値が外部での変更により 52-bit を超えて異常に表示される可能性がある以外、レシピ入力・SQL クエリ・データベース(データサンプリング)などの外部で数値を変更可能な機能においても、範囲を超えたら表示が異常になる可能性があります。

注意 :

double は元から正確さに欠けているので、+2⁶⁴ ~ -2⁶⁴ の範囲内なら全部正確に表示されることが保障されません。前述の規則はただ int64 / uint64 使用上の説明です。