

はじめに

このたびは、GLC用ラダー作成ソフト「Pro-Control Editor Ver.3.0」をご採用いただき、誠にありがとうございます。

この製品を正しくご使用いただくために、マニュアル類をよくお読みください。

また、マニュアル類は必ずご利用になる場所のお手元に保管し、いつでもご覧いただけるようにしておいてください。

おことわり

- (1) 「Pro-Control Editor」(以下本製品といいます)のプログラムおよびマニュアル類は、すべて(株)デジタルの著作物であり、(株)デジタルがユーザーに対し「ソフトウェア使用条件」に記載の使用権を許諾したものです。当該「ソフトウェア使用条件」に反する行為は、日本国内外の法令により禁止されています。
- (2) 本書の内容については万全を期して作成しておりますが、万一お気づきの点がありましたら、(株)デジタル「サポートダイヤル」までご連絡ください。
- (3) 本製品を使用したことによるお客様の損害、および免失利益、または第三者からのいかなる請求につきましても、当社はその責任を負いかねますので、あらかじめご了承ください。
- (4) 製品の改良のため、本書の記述と本製品のソフトウェアとの間に異なった部分が生じることがあります。最新の説明は、別冊ないし電子的な情報として提供していますので、あわせてご参照ください。
- (5) 本書は、(株)デジタルから日本国内仕様として発売された製品専用です。
- (6) 本製品が記録・表示する情報の中に、(株)デジタルおよび/または第三者が権利を有する無体財産権、知的所有権に関わる内容を含む場合がありますが、これは(株)デジタルがこれらの権利の利用について、ユーザーおよび/またはその他の第三者に、何らの保証や許諾を与えるものではありません。

© Copyright 2000 Digital Electronics Corporation. All rights reserved.

(株)デジタル 2000 November

商標・商号の権利については「商標権などについて」をご覧ください。

商標権などについて

本書に記載の社名、商品名は、各社の商号、商標(登録商標を含む)またはサービスマークです。本製品の表示・記述の中では、これら権利に関する個別の表示は省略しております。

商標等	権利者
Microsoft, MS-DOS, Windows, Windows 95, Windows 98, Windows NT, Windowsエクスプローラ	米国Microsoft社
Intel, Pentium	米国Intel社
Pro-face	(株)デジタル
Flex Network	(株)デジタル
NEC, PC-9800	日本電気(株)
IBM, PC/AT	米国IBM社
Adobe, Acrobat	アドビシステムズ社

なお、上記商号・商標類で、本書での表記と正式な表記が異なるものは以下の通りです。

本書での表記	正式な表記
Windows 98	Microsoft® Windows® 98 オペレーティングシステム
Windows 95	Microsoft® Windows® 95 オペレーティングシステム
Windows NT	Microsoft® Windows NT® オペレーティングシステム
Windows 2000	Microsoft® Windows® 2000 オペレーティングシステム
MS-DOS	Microsoft® MS-DOS® オペレーティングシステム
Word 97	Microsoft® Word 97
Acrobat Reader	Adobe® Acrobat® Reader

マニュアルの読み方

マニュアルの構成

本書は「Pro-Control Editor」(以下、本製品と呼びます)の使用方法を説明するマニュアル類の1冊、「Pro-Control Editor オペレーションマニュアル」です。

Pro-Control Editor Ver.3.0のCD-ROMには、以下のマニュアルがPDF形式で記録されています。

- ・オペレーションマニュアル
- ・ユーザズマニュアル

これらを利用するには、Adobe Acrobat Reader 4.0J(以下Acrobat Readerと言います)が必要です。PDFマニュアルは、電子メールで配布したり、しおりをクリックするだけで表示したい画面をすばやく見ることができます。

ご使用のパソコンにAcrobat Readerがインストールされていない場合は、インストールしてください。また、関連マニュアル類は以下の通りです。あわせてご覧ください。

本製品に付属	Pro-Control Editor オペレーションマニュアル (本書)		本製品を使うためのインストール方法と、機能の概要を習得するためのチュートリアルレッスン、エラーメッセージの一覧です。
	Pro-Control ユーザズマニュアル		GLCと本製品との組み合わせに関するソフトウェア的な設定について説明しています。
	オンラインヘルプ		本製品のヘルプ機能です。大きく3つの部分からなっています。 Pro-Controlヘルプ 個別の命令や機能、操作方法についての説明です。 DIOドライバヘルプ DIOドライバに関するヘルプです。 ユニワイヤI/FドライバヘルプユニワイヤI/Fドライバに関するヘルプです。 Flex Network I/Fドライバヘルプ Flex Network I/Fドライバに関するヘルプです。
関連商品のマニュアル	GLCシリーズ ユーザズマニュアル(別売)		Pro-Control Editor対応のGLCのハードウェアマニュアルです。GLC本体とは別売です。
	GP画面作成ソフトに付属 ¹	オペレーションマニュアル	GP画面作成ソフトを使うためのインストール、操作手順と機能のすべてを説明します。(PDFマニュアル)
		タグリファレンスマニュアル	GPの画面上機能を指定する「タグ」についてまとめて説明します。(PDFマニュアル)
		パーツリスト	GP画面作成ソフトにあらかじめ用意されているパーツと図記号をまとめて説明します。(PDFマニュアル)
		PLC接続マニュアル	GPと各社のPLCの接続方法について説明します。(PDFマニュアル)
2Way Communicatorソフトに付属 ²	オペレーションマニュアル	2 Way communicatorソフトを使うための操作手順と機能のすべてを説明します。(PDFマニュアル)	

これらマニュアル類のほか、データファイル readme.txtとして補足説明や機能の追加・修正情報がフロッピーディスクおよびCD-ROMに添付されていることがありますので、必ずご覧ください。

- 1 本製品に対応するGP画面作成ソフトは、GP-PRO/PB for Windows Ver.5.0以上です。GP-PRO/PB for Windows Ver.5.0のマニュアルはインストレーションガイド以外はCD-ROMにPDF形式で記録されています。
- 2 本製品に対応する2 Way Communicatorソフトは、Pro-Server with Pro-Studio for Windows Ver.3.0以上です。Pro-Server with Pro-Studio for Windows Ver.3.0のマニュアルはCD-ROMにPDF形式で記録されています。

目次

はじめに	1
商標権などについて	2
マニュアルの読み方	3
表記のルール	8
使用上の注意	10
第1章 Pro-Control Editor の基本事項	
1.1. Pro-Control Editor とは	1-1
1.2. 次のステップ	1-1
第2章 インストール	
2.1. Pro-Control Editor のインストール	2-1
2.1.1. インストール手順	2-2
第3章 プログラムの作成(入門レッスン)	
3.1. レッスン始める前に	3-1
3.1.1. ロジックプログラムを作成する前にオプション画面で設定を行う	3-2
3.2. 変数の作成と削除	3-5
3.2.1. 変数リストの作成	3-5
3.2.2. 変数タイプの指定	3-7
3.2.3. プログラムのセーブ	3-8
まとめ	3-8
3.3. ラング、命令、および分岐の挿入	3-9
3.3.1. ラングの挿入	3-9
3.3.2. ラングの削除	3-10
3.3.3. 命令の挿入	3-11
3.3.4. 命令の削除	3-14
3.3.5. 命令のコピー&貼り付け	3-15
3.3.6. 分岐の挿入	3-16
3.3.7. 初期化ロジックプログラム	3-18
まとめ	3-19

3.4. 命令への変数の割り付け	3-20
3.4.1. 命令パラメータボックス	3-20
3.4.2. 変数の入力	3-21
3.4.3. 作業の完了	3-24
まとめ	3-24
3.5. ロジックプログラムのドキュメント化	3-25
3.5.1. プログラムコメントの追加	3-25
3.5.2. ラングコメントの追加	3-26
3.5.3. 変数へのコメントの追加	3-27
3.5.4. [コメントリスト]ウィンドウ	3-28
まとめ	3-28
3.6. ラングのコピー、切り取りおよび貼り付け	3-29
3.6.1. ラングのコピー	3-29
3.6.2. ラングの貼り付け	3-29
3.6.3. [切り取り]コマンドの使用	3-30
まとめ	3-30
3.7. サブルーチンおよびラベル	3-31
3.7.1. サブルーチンの挿入	3-31
3.7.2. ラベルの挿入	3-33
まとめ	3-33
3.8. ロジックプログラム内の移動	3-34
3.8.1. [検索]コマンド	3-34
3.8.2. [リファレンス]コマンド	3-35
3.8.3. [リファレンス]ダイアログボックスと他のウィンドウの併用	3-36
3.8.4. ブックマークの使用	3-37
3.8.5. [指定ラングへ移動]コマンドの使用	3-38
3.8.6. [ラベルへ移動]コマンドの使用	3-38
まとめ	3-38
3.9. I/Oの割り付け	3-39
3.9.1. 実 I/O への変数の割り付け	3-39
3.9.2. [I/Oコンフィギュレーション]ウィンドウのアンマップ(割り付け解除)	3-46
3.9.3. I/Oにマップした変数を命令で使用方法	3-46
まとめ	3-46
3.10. プログラムエラーチェック	3-47
まとめ	3-48
3.11. ロジックプログラムの印刷	3-49
まとめ	3-50

第4章 ロジックプログラムを実行する

4.1. コントローラの設定	4-1
4.1.1. コントローラへの書き込み	4-6
4.1.2. モニタリングモードへの移行	4-7
4.1.3. イーサネット機能 (対応機種: GLC2400)	4-8
4.2. コントローラの RUN/STOP	4-10
4.3. システム変数によるプログラムのトラブルシューティング	4-12
4.4. システム変数の表示	4-13
4.5. コントローラからの読み込み	4-14
4.6. プロパティ	4-14
4.7. CF メモリロードツール (対応機種: GLC2400)	4-15
4.7.1 CF メモリロードツールの作成 / 送信	4-15
4.7.2 システム情報の表示	4-16
まとめ	4-17

第5章 モニタリングモードでの動作確認

5.1. 編集を始める前に	5-1
5.2. モニタリングモード編集にカラーを使用する	5-1
5.3. ディスクリートを ON / OFF する	5-2
5.4. ディスクリートを強制的に ON/OFF する	5-3
5.5. 変数値の変更	5-3
5.6. 変数の属性変更	5-4
5.7. データ値表示リスト	5-5
5.8. オンラインエディット (対応機種: GLC2400)	5-6
5.8.1 オンラインエディットの編集機能	5-6
5.8.2 データ保存	5-8

第6章 Pro-Control Editor と GP-PRO/PB

6.1. GP-PRO/PB での変数表示	6-1
6.1.1. GP-PRO/PB プロジェクトを開く手順	6-1
6.1.2. GP タイプと PLC タイプの選択	6-1
6.1.3. インポート	6-2
6.1.4. GP-PRO/PB による画面作成	6-4
6.2. Pro-Control Editor の変数を GP-PRO/PB のプロジェクトにリンクする	6-7
6.3. GLC へのダウンロード	6-7
6.4. ポンププロジェクトの実行	6-8
まとめ	6-8

第7章 Pro-Control Editor と Pro-Server (対応機種 : GLC2400)

- 7.1. GLC 変数のインポート 7-1
- 7.2. S100 ファイルチェック 7-2

付録 A : エラーと警告

- 200-299 : ロジックのエラーと警告 8-1
- 300-399 : 変数のエラーと警告 8-3
- 400-499 : I/O のエラーと警告 8-5
- 500-549 : 一般的な I/O ドライバのエラー 8-5
- 800-899 : 特定の I/O ドライバのエラー 8-5
- 900-1000 : 特定の I/O ドライバの警告 8-5

付録 B : 用語集

表記のルール



本書は、以下のルールで表記します。

わかりにくいところなどは「サポートダイヤル」までお問い合わせください。「サポートダイヤル」では、(株)デジタル製品についての技術的なご質問・ご相談にお答えします。

なお、パソコンやWindowsそのものに関することは、パソコンをお買い上げの販売店、メーカーにお問い合わせください。


安全に関する注意表記

本製品のご使用上、安全に関して重要な説明には、以下の表示を添えています。

表示	意味内容
 警告	この表示を無視して誤った取り扱いをすると、人が死亡または重傷を負う可能性が想定される内容を示します。
 注意	この表示を無視して誤った取り扱いをすると、人が傷害を負ったり、物的損害の発生が想定される内容を示します。
重要	この表示の説明に従わない場合、機器の異常動作やデータの消失などの不都合が起こる可能性があります。
強制	必ず実施していただきたい操作、作業などを表します。
禁止	決して行ってはならない操作、作業などを表します。

説明のための表記

本書では、説明の便宜のため、以下のように表記します。

表記	意味内容
 MEMO	参考になることがら、補足的な説明です。
<u>参照</u>	関連する説明が掲載されている項目(マニュアル名、章・節・項)を示します。
*	脚注で説明している語句についています。
Esc Ctrl	パソコンのキーを表します。 <u>参照</u> キーボード対応表
Pro-Control Editor	GLC のラダープログラムを作成/転送/モニタを行う機能をもったソフトウェアです。
コントローラ	GLC に組み込まれている制御機能を指します。
GP-PRO/PB (画面作成ソフト)	GP-PRO/PB for Windows Ver5.0 以上を指します。
GLC	(株)デジタル製グラフィックロジックコントローラの総称です。
PLC	プログラマブル ロジック コントローラ、シーケンサの総称です。

キーボード対応表

本書では、パソコンのキーを以下のように表記します。

機種によってやや異なりますが、この対応で読み替えてください。

機種 表記	PC/AT互換機		PC-9800シリーズ
	日本語 106キーボード	英語 101キーボード	
Esc	Esc	Esc	ESC
Tab	Tab ⇐⇒	Tab ⇐⇒	TAB
Ctrl	Ctrl	Ctrl	CTRL
Shift	Shift	Shift	SHIFT
Alt	Alt	Alt	GRPH
Delete	Delete	Delete	DEL
Back space	Back space	Back space	BS
日本語入力	Alt + 半角/全角	Alt + ~	CTRL + XFER

：日本語入力のオン / オフ操作は、使用する日本語FEPによって異なります。

モデル環境

本書で、操作や機能を説明する場合のモデルとなるシステム構成は以下の通りです。

これ以外のシステム構成では、表示や各部の名称が異なることがありますが、同等の機能をもつものと読み替えてください。

機材・ソフト	モデルシステムの仕様	備考
パソコン	PC/AT互換機	
メモリ	32MB	
マウス	Windows 95対応マウス	
OS	Windows 95	
GLC	GLC100シリーズ	
パソコンとGLCとの 接続方法	RS-232C	(株)デジタル製ケーブル GPW-CB02使用

使用上の注意

使用上の注意

本製品の使用について

誤動作や事故の原因となりますので、以下の点にご注意ください。



警告 本誌に記載されているアプリケーション事例は参考用ですので、ご採用に際してはシステム・機器・装置の機能や安全性をご確認の上、ご使用ください。

本誌に記載のない条件や環境での使用、および原子力用・電力用・航空宇宙用・医療用・乗用移動体用の機器あるいはシステムなど特殊用途への使用をご検討の場合は、当社営業担当までお問い合わせください。

タッチパネルスイッチは非常停止用スイッチとして使えません。産業用ロボットほか、労働大臣が指定する産業用機械設備の非常停止用スイッチとしては、必ず人間が直接操作するスイッチを設置することが関係法令で義務づけられています。また、これ以外の装置設備でも、安全確保のため、必ず同様のスイッチを設置してください。GLCの故障により重大な事故または損失の発生が予測される設備への適用に関しては、バックアップやフェイル・セーフ¹を系統的に設置してください。

禁止 GLCは一般工業等を対象とした汎用品として制作されたもので、人命にかかわるような状況下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。よって、人命や重大な物的損傷にかかわる制御には使用しないでください。

プログラム使用中に、パソコン本体の電源をOFFしないでください。

テキストエディタなどを使用して、本製品のプロジェクトファイルの中身を変更しないでください。

GLCがサポートしていない機能を使用した画面は、GLCに転送しないでください。

ディスクの取り扱いについて

ディスクの破損・故障を防ぐため、以下の点にご注意ください。

強制 ・ パソコン本体の電源のON/OFFは、ディスクを抜いてから行ってください。

禁止 ・ ディスクドライブのランプが点灯している時は、CD-ROMを取り出さないでください。
・ CD-ROMの記録面、フロッピーディスクの磁性体面（シャッターの中）に手を触れないでください。
・ 極端な高温や低温、湿気やホコリの多い場所にディスクを置かないでください。
・ フロッピーディスクを、ステレオのスピーカーやテレビ、磁気治療器などに近づけないでください。

1 オペレータの操作ミスや、センサーやコントローラの誤動作による被害を最小限にする工夫をいう。

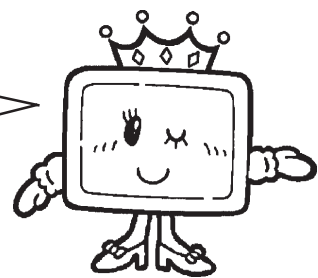
制限事項について

本製品には、以下のような制限があります。

- GLC100 および GLC300 では、Pro-Server with Pro-Studio for Windows(2Way Driver)をサポートしていません。
- GP-PRO/PB はご使用のパソコン内部の文字フォントやグラフィック機能を使用して表示します。このため、これらの表示はGLCへ転送後、GLC上での表示とパソコン上での表示に多少の相違が生じる場合があります。あらかじめご了承ください。
- GLC100 では、GP-PRO/PB 機能の中で AUX 出力 / インチングタグ / t タグ AUX 出力 / バックアップ機能などその他の GP-370 で使用できない機能は使用できません。
- Pro-Control Editor の変数にデバイスタイプがないので GP-PRO/PB 機能の E タグ、K タグの間接指定のデバイスタイプ&アドレスの指定はできません。
- GLCのロジックタイムが長い場合、折れ線グラフのサンプリングタイムが守れないときがあります。
- Pro-Control Editor で配列を使用する場合、GP-PRO/PB 側で配列の要素を削除しないでください。
- DOS転送ツールでシステムのセットアップができません。
- Pro-Control Editor より後に Word 97以降をパソコンにインストールしている場合は、拡張子 WLL が Wordに関連付けされるためエクスプローラにより Pro-Control Editor ロジックプログラム(.WLL)を開くことはできません。Pro-Control Editorのソフトを立ち上げてからロジックプログラム(.WLL)を開いてください。
- 実数はEタグ、Kタグのfloatで使用して下さい。ただし、GLCの変数とタグの精度のちがひによる誤差があります。
- メモリリンク方式を使用の場合、GLC変数は、折れ線グラフの一括表示ができません。
- GLCの変数は32bit デバイスのLow/High順で扱われます。
- GLC100 では、Qタグのサブ表示は使用できません。
- GLCのロジックタイムが長い場合、音声の再生時に音が途切れることがあります。
- 整数型変数のビット指定においてTタグ、Wタグのビット書き込み(「反転」以外)を行うと、該当する整数型変数の指定したビット以外はすべてクリア(0)します。
- ビット動作が反転のTタグに同一整数型変数の異なるビットを指定したものを複数重ねて配置した場合は、最後に配置したTタグのみ有効となります。
- GLCの保持型変数のデータを保持するSRAMはリチウム電池でバックアップされていますが、バックアップ期間は初期状態(満充電)で約60日、電池寿命時で約6日となりますので、この期間以上にバックアップが必要な場合は、ホストでバックアップしたり、Editorでバックアップするシステム構成をとってください。
- GLC2400 では、AUXとしてリセット入力以外は使用できません。
- オンラインエディットではSRAMに保存されているロジックプログラムを編集します。オンラインエディット後、電源OFF時のバッテリー切れなどによりSRAMのデータが失われた場合、FEPRMのデータが読み出されます。オンラインエディットした場合は、GLCのオフラインメニューで[FEPRMへのコピー]を実行するか、Pro-Control EditorでWLLファイルとして保存することにより、バックアップしてください。
- パソコンとGLCとの実数の精度の違いによって、モニタリングで表示する値と入力した値が一致しない場合があります。

MEMO

このページは、空白です。
ご自由にお使いください。



第1章 Pro-Control Editorの基本事項

1.1. Pro-Control Editor とは

Pro-Control Editorは、GLCのロジックプログラムを作成するソフトです。(以下、単にEditorといたします)

Editorは、以下のような機能を含んでいます。

- ・GLC専用DIOユニット用ドライバ
- ・GLC専用ユニワイヤ拡張I/Fユニット用ドライバ
- ・GLC専用Flex Network I/Fユニット用ドライバ
- ・ロジックプログラムの編集機能
- ・ロジックプログラム転送機能
- ・クロスリファレンスレポート
- ・モニター機能
- ・オンラインエディット機能¹
- ・イーサネット経由の通信¹

Editorは、ロジックプログラムをグラフィカルなWindows環境で作成することができます。

ロジックプログラムはGLCにダウンロードされ、GLC上で動作します。

Editorで作成した変数は、GLCに対応したGP作画ソフト「GP-PRO/PB for Windows Ver.5.0」に取り込むことで変数名を共有できます。

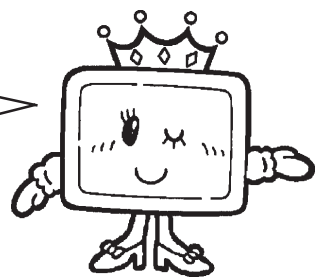
1.2. 次のステップ

Editorのインストールを開始します。次の章では、この作業の方法を説明します。画面に表示される指示を含めて、インストレーション手順を正しく実行することが重要です。

¹ GLC2400のみサポートしています。

MEMO

このページは、空白です。
ご自由にお使いください。



第2章インストール

2.1. Pro-Control Editor のインストール

動作環境

	必要な仕様		備考
パソコン	Windowsが動作する機種		Pentium133MHz以上推奨。 PC/AT互換機, PC-9801シリーズ機のどちらでも動作します。
ディスプレイ	VGA (640×480ドット) 以上で256色以上 カラー表示可能なもの		
マウス	Windows95/98/2000/NT (Ver. 4.0以上) 対応品		
ハードディスク スペース	最小	最大	インストールするために必要なスペース 単位はメガバイト (Mバイト) Acrobat Readerをインストールする場合は、 15MB以上のスペースが必要です。
	8M	10M	
メモリ	16Mバイト以上 (推奨 32Mバイト)		
プリンタ	Windows95/98/2000/NT (Ver. 4.0以上) 対応品		
転送ケーブル	(株) デジタル製GPW-CB02 (別売品)		パソコン本体のRS-232Cコネクタの形状がD- sub 25ピンまたはD-sub 9ピンでない場合は対 応する変換コネクタを別途ご用意ください。
OS	Windows95/98/2000/NT (Ver. 4.0以上)		
日本語FEP	Windows95/98/2000/NT (Ver. 4.0以上) 対応品		
GLC本体	GLCシリーズ		対象形式 : GLC100-**41-24V 対象形式 : GLC300-**41-24V 対象形式 : GLC2400-**41-24V
対応作画ソフト	GP-PRO/PB for Windows Ver. 5.0以降		

【メモ】

インストールを開始する前に、インストーラ以外のすべてのプログラムを終了させてください。通常のアプリケーションはもちろん、常駐型プログラム(ウイルス検出ソフトなど)も、忘れずに終了(常駐解除)させてください。

参考:一部の常駐型ウイルス検出ソフトを常駐させたままインストールすると、パソコンの電源が切れたり、ハングアップしたりすることが確認されています。このような場合は、パソコンを再起動し、常駐型プログラムをすべて解除してから、再度インストールしてください。また、これらの現象は一時的なものであり、以後のパソコンの使用に何ら悪影響はありません。

ディスクの構成

¥¥ — ¥ProCtrl¥Disk1
 ¥ProCtrl¥Disk2
 ⋮
 ¥ProCtrl¥Disk4
 ¥ProPBWin¥Disk1
 ¥ProPBWin¥Disk2

} インストールをする場合は、Disk1 というフォルダにある「Setup.exe」を実行してください。

} GP-PRO/PB for Windows Ver.5.0 を GLC 対応にするためのアドオンインストーラです。ここにある Disk1 というフォルダにある「Setup.exe」を実行してください。
 (通常は、Pro-Control Editor をインストールすると続いて自動的に GP-PRO/PB for Windows Ver.5.0 が実行されません。)

以下の説明では、

ハードディスクドライブ(インストール先): C

CD-ROM ドライブ : D

となっている場合を例にしています。

2.1.1.1. インストール手順

CD-ROM からのインストール

あらかじめ GP-PRO/PB for Windows Ver.5.0 をインストールしておいてください。

1. Pro-Control Editor マスター CD-ROM ディスクを CD-ROM ドライブに挿入します。
2. [スタート]をクリックし、[ファイル名を指定して実行]を選択します。
3. 名前に D:¥ProCtrl¥Disk1¥setup と入力します。
4. [OK]をクリックするか、[Enter]を押します。通知ウィンドウが表示されます。
 [次へ]をクリックすると、[セットアップタイプ]ウィンドウが表示されます。
5. インストールタイプを選択し、[次へ]をクリックします。
 画面に表示される指示に従って作業を進めます。

Pro-Control Editor のインストールが終了しますと、続いて GP-PRO/PB へのアドオンインストールがはじまります。画面に従ってインストールを行ってください。

参考: インストール中に間違った入力または選択をした場合は、[前へ]をクリックして、訂正してみてください。それでもうまく行かない場合は、はじめからインストールをやり直してください。

【禁止】

インストール完了後は、Pro-Control Editor のフォルダを変更してはいけません。Pro-Control Editor のプログラムは、自動的に必要なファイルを一定のフォルダに配置しています。このフォルダが変更されると、Pro-Control Editor は正常に動作できません。

README.TXT ファイルを見る

Windows エクスプローラを使って、Pro-Control Editor フォルダ(C:¥Pro-Control¥)に移動します。

README.TXT をダブルクリックして、Pro-Control Editor についての最新の情報を表示します。

第3章 プログラムの作成(入門レッスン)

本章では、Editorを使ってプログラミングモードでロジックプログラムを作成する方法を、順序を追って説明します。

参照：第3章で使用するチュートリアルプログラムの完成品は、¥Pro-Control¥Sampleのフォルダの中にSoda1.wllファイルとして付属しています。作成方法がわからない場合や検索などの練習の場合は、このファイルを開いて参考にしてください。

画面の各部の説明は、[参照](#) **Pro-Control Editor ヘルプ**

3.1. レッスンを始める前に

本章の各レッスンは、Editorの操作手順について、実例に沿って説明しています。

< 内容説明 >

3.2. 変数の作成と削除

Editorで作成するロジックプログラムのEditorの動作を設定する方法について詳しく説明します。また、中で使用する変数の作成と削除、初期値の設定方法を説明します。

3.3. ラング、命令、および分岐の挿入

ラングの作成、命令や分岐の挿入方法、ラングおよびラングに関連づけられている命令や分岐の削除方法について説明します。

3.4. 命令への変数の割り付け

Editorで作成するロジックプログラムの中の命令にオペランドを割り付ける方法について説明します。

3.5. ロジックプログラムのドキュメント化

Editorで作成するロジックプログラムにコメントを付ける方法について説明します。プログラム全体、特定のラング、個別の命令にコメントを付ける方法です。

3.6. ラングのコピー、切り取りおよび貼り付け

ラングのコピー、切り取りおよび貼り付け方法を説明します。

3.7. サブルーチンおよびラベル

Editorで作成するロジックプログラムにサブルーチンやラベルを挿入する方法について説明します。

3.8. 回路の検索と移動

Editorで作成するロジックプログラムの中で、目的の回路をすばやく検索し移動する方法を説明します。

3.9. I/Oの割り付け

Editorで作成するロジックプログラムの中の論理変数を実I/Oに割り付ける方法について説明します。

3.10. プログラムエラーチェック

Editorで作成するロジックプログラムのエラーをチェックする方法について説明します。

3.11. ロジックプログラムの印刷

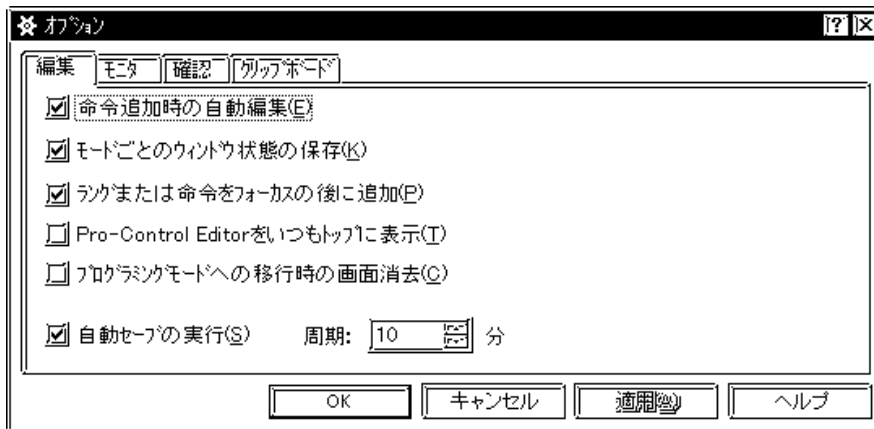
Editorで作成するロジックプログラムを印刷する方法について説明します。

3.1.1. ロジックプログラムを作成する前にオプション画面で設定を行う

Editorを使ってロジックプログラムの作成を始める前に、好みにあった動作に操作設定を合わせることができます。項目ウィンドウを使って、プログラムを作成し実行する方法をカスタマイズできます。

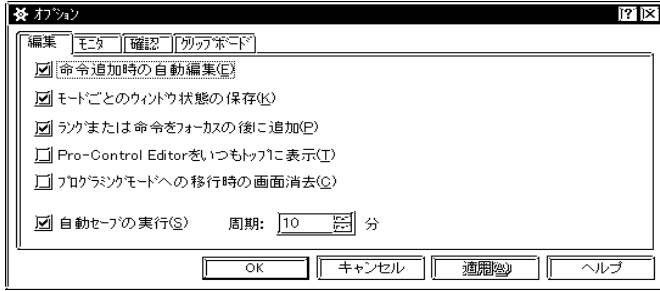
オプション画面の設定手順

1. [ファイル]メニューから、[オプション]を選択します。[オプション]ダイアログボックスが表示されます。



2. 各項目のチェックボックスをクリックすると、その項目が選択またはクリアされます。
[オプション]ダイアログボックスの中の各項目の意味は、次頁の表の通りです。

編集タブ



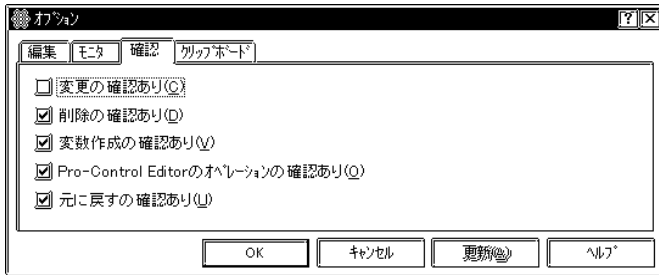
項目	説明
命令追加時の自動編集 (デフォルト=選択)	プログラムの中に挿入された新しい命令について、自動的に[命令パラメータ]ウィンドウが開きます。
モードごとのウィンドウ状態の保存 (デフォルト=選択)	Editorの再起動やモード切り替えの際、前回の作業終了時に開いていたウィンドウが、そのときの状態で開かれます(ウィンドウのサイズ、位置など)。この設定は[データ値表示リスト]ウィンドウにも適用されます([データ値表示リスト]ウィンドウは、現在のプログラムがモニタリングモードの場合、その変数の値を表示します。)
ラックまたは命令のフォーカスの後に追加 (デフォルト=選択)	この項目が選択されている場合、新しい命令は選択した命令の右側に挿入されます。また、ラベル、ラック、サブルーチンなどのオブジェクトは、選択しているラックの下に挿入されます。この項目をクリアした場合、新しい命令は、左側、新しいオブジェクトは上側に挿入されます。命令の設定をされていない横ライン(接続線)を選択している場合は、設定に関わらず、命令はライン上に挿入されます。
Pro-Control Editorをいつもトップに表示 (デフォルト=クリア)	Pro-Control Editorウィンドウが、いつも開かれているウィンドウの一番手前に表示されます。
プログラミングモードへの移行時の画面消去 (デフォルト=クリア)	モニタリングモードからプログラミングモードに移るときにラダーロジックスクリーンをクリアします。
自動セーブの実行	編集中のファイルを設定時間毎に保存します。保存するファイルの拡張子は "WL~" になります。自動セーブされたファイルを開ける場合は、拡張子を "WLL" にしてください。

モニタータブ



	項目	説明
モニターの更新	パワーフロー (デフォルト=選択)	コントローラの実行中にパワーフローが表示されます。パワーフローとは、コントローラがRUN中に実行しているラダーを強調表示します。
	状態フロー (デフォルト=未選択)	コントローラの実行中に状態フローが表示されます。状態フローとは、コントローラがRUN中に実行している命令を強調表示します。パワーフローと状態フローを同時に表示できます。
	周期(ミリセカンド) (デフォルト=500msec)	Editorがパワーフロー、状態フロー、データ値、およびステータスバーを更新するためにコントローラから新しいデータを要求する頻度を指定します。

確認タブ



項目	説明
変更の確認あり (デフォルト=クリア)	Pro-Control Editorは、[更新]をクリックしたときにだけ変更を受け入れます。この項目をクリアした場合、Pro-Control Editorは、変更の確認を要求します。
削除の確認あり (デフォルト=選択)	Pro-Control Editorは、プログラムの作成中に、すべての削除について確認を要求します。
変数の作成の確認あり (デフォルト=選択)	Pro-Control Editorは、プログラムの中での新しい変数の作成について確認を要求します。これは、オフライン環境にだけ適用されます。
Pro-Control Editorのオペレーションの確認あり (デフォルト=選択)	Pro-Control Editorは、コントローラの動作の変更(スタート/ストップ、読み出し/書き込みなど)について確認するように要求します。
元に戻すの確認あり (デフォルト=選択)	[元に戻す]を実行する前に、確認するかどうかを指定します。

クリップボードタブ



	項目	この項目が選択された場合の動作
コピーされる変数の形式	コマで区切る (デフォルト=クリア)	Pro-Control Editorウィンドウからクリップボードにコピーされたフィールドがコマで区切られます。 例、My_variable,Discrete,adescription
	タブで区切る (デフォルト=選択)	Pro-Control Editorウィンドウからクリップボードにコピーされたフィールドがタブで区切られます。 例、My_variable[TAB]Discrete [TAB] adescription
	引用符の使用 (デフォルト=クリア)	Pro-Control Editorウィンドウからクリップボードにコピーされたフィールドが区切り記号で区切られ、引用符で囲まれます。 例、"My_variable","Discrete","adescription"

デフォルト設定値

このチュートリアルでは、デフォルト設定値を使用します。

[キャンセル]をクリックし、オプションダイアログを閉じます。

キャンセルをクリックすることで、デフォルト設定値のままダイアログを閉じることができます。

3.2. 変数の作成と削除

Editorを使ってファーストフードレストランのソフトドリンクマシンの動作を制御するロジックプログラムを作成します。

このマシンには、以下の機能があります。

ボタンを1度押すことによって、自動的にL/M/Sのサイズのカップを装填する

カップがディスペンサーの下にある場合だけ、氷またはソーダを吐出する

機械に電源を投入したあとに、機械によって装填されたカップの数をカウントする

参照：第3章で使用使用するチュートリアルプログラムの完成品は、¥Pro-Control¥Sampleのフォルダの中にSoda1.wllファイルとして付属しています。

3.2.1. 変数リストの作成

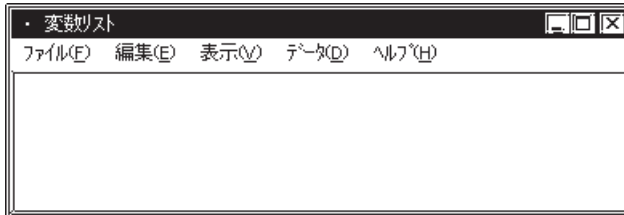
変数はロジックプログラムの作成中にいつでも追加できますが、あらかじめ設定することもできます。このチュートリアルで使用する変数のリストをここで作成しておくことで便利です。変数とは、ユーザーが作成し、ロジックプログラムの命令に割り付けられるデータを変数名で表したものです。

変数リストの作成方法

メニュー内の詳細な説明は、Pro-Control Editor オンラインヘルプをご参照ください。

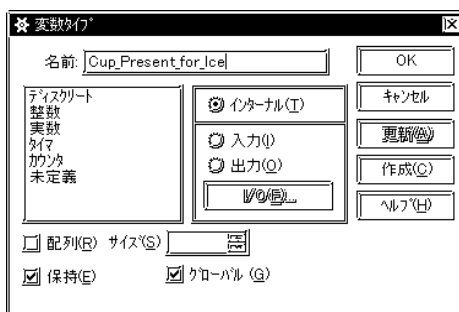
1. [データ]メニューから、[変数リスト]を選択します。

[変数リスト]ウィンドウが表示されます。



2. [編集]メニューから、[変数の追加]を選択します。

[変数タイプ]ダイアログボックスが表示されます。



3. 名称フィールドに Cup_Present_for_Ice と入力します。

【重要】

変数名の先頭に数字を使ったり、スペースを含めることはできません。たとえば、1Switchや Switch 1 という変数名は使用できません。Switch1またはSwitch_1という変数名は使用できます。変数名は、大文字 / 小文字または、全角 / 半角を区別しません。また、記号#は、メーカーの予約文字です。#ではじまる変数を作成しないでください。変数名"_"以外の記号は使用できません。変数名は、半角20文字 / 全角10文字まで設定できます。

配列、タイマ、カウンタの要素及び整数 / 実数のビット指定も文字数に含まれます。

グローバルと設定した変数のみ GP-PRO/PB for Windows Ver.5.0のインポート機能によって変数が読み込まれます。

配列

配列とは、1つの名前を割り付けて、複数の要素を持たせるタイプの変数です。要素数には、メモリサイズにより制限があります。GLCシリーズで使用可能な配列の要素数については「Pro-Control ユーザーズマニュアル」を参照してください。名前の後ろに[]をつけ、その中に番号を指定して参照します。

保持

保持型変数は静的なメモリで管理されるため、電断時もデータ値の保持が可能です。また、保持型変数は、プログラミングモードで設定した値を初期値としてもっています。電断時やGLC本体のリセット時は直前のデータを保持しますが、モニタリングモードでコントローラのリセットを行った時は、プログラミングモードで設定した値を初期値としてイニシャライズされます。また、GLCのWLLファイルを読み込むことで実行結果をEditorで保存することができます。非保持型のデータは0クリアまたはOFFになります。

グローバル

グローバルは、グローバルと非グローバルがあります。グローバルと設定すると、GP-PRO/PBで変数のインポートが可能となります。タグ等で表示するGLCの変数はグローバルに設定してください。変数リストで複数個選択する事でグローバル/非グローバルの一括変換が可能です。

3.2.2. 変数タイプの指定

現在、[変数リスト]ウィンドウに変数Cup_Present_for_Iceが表示されています。その下のリストの中の[未定義]という語が反転表示されています。これは変数 Cup_Present_for_Iceに変数タイプが指定されていないことを表しています。ここでは、ディスクリート、入力を指定します。

変数タイプの指定方法

1. [変数タイプ]ダイアログボックスから[ディスクリート]を選択します。
2. [入力]を選択します。
3. 「保持」のチェックボックスをクリアにします。これで電断、GLC本体のリセットによるデータの保持は、非保持になります。
4. [作成]をクリックします。これで変数Cup_Present_for_Iceがディスクリート、入力として指定されました。

このとき、[変数タイプ]ダイアログボックスで変数Cup_Present_for_Iceに対して行った変数タイプを設定通り[変数リスト]上に表示されています。

[OK]をクリックした場合には、[変数タイプ]ダイアログボックスは閉じられます。ラングや命令の編集操作(挿入、ドラッグ&ドロップ、クリックなど)上メリットがありますので、ここでは、[変数リスト]や[変数タイプ]ダイアログボックスは開いたままにしておいてください。

参考:[変数リスト]ウィンドウに表示する変数タイプを選択するには、[表示]メニューで表示する変数タイプを選択します。選択した変数タイプの横にチェック・マークが表示されます。

これで変数を作成し、変数タイプを割り付ける方法がわかりました。次に、下の表に示す変数を作成します。変数は[変数タイプ]ダイアログボックスで直接作成できます。

変数名	変数タイプ	I/Oタイプ	保持/非保持	グローバル
Power_On_pushbutton	ディスクリート	入力	非保持	グローバル
Cup_Present_for_soda	ディスクリート	入力	非保持	グローバル
Ice_pushbutton	ディスクリート	入力	非保持	グローバル
Large_pushbutton	ディスクリート	入力	非保持	グローバル
Medium_pushbutton	ディスクリート	入力	非保持	グローバル
Power_Off_pushbutton	ディスクリート	入力	非保持	グローバル
Small_pushbutton	ディスクリート	入力	非保持	グローバル
Ice	ディスクリート	出力	非保持	グローバル
Soda_valve	ディスクリート	出力	非保持	グローバル
Fill_Timer	タイマ	インターナル	保持	グローバル
Number_of_Larges	カウンタ	インターナル	非保持	グローバル
Number_of_Mediums	カウンタ	インターナル	非保持	グローバル
Number_of_SmallS	カウンタ	インターナル	非保持	グローバル

完了したら、[変数タイプ]ダイアログボックスを閉じます。

参考:変数名をミスタイプした場合は、[変数リスト]ウィンドウの中の編集メニューの[名前の変更]で変更できます。

[変数リスト]をウィンドウ表示中、[Insert]キーを押すと[変数タイプ]ダイアログが表示されますので、すばやく変数を作成できます。


3.2.3. プログラムのセーブ

作成データの安全性のため、プログラムを定期的にセーブするか自動セーブの設定にしておくことをお勧めします。

〔参照 3.1.1 ロジックプログラムを作成する前にオプション画面で設定を行う〕

セーブ方法

1. [ファイル]メニューから、[名前を付けて保存]を選択します。ただし、一度保存した事のあるファイルの場合は、「保存」を選択します。その場合、2./3.の手順は不要です。
2. ファイル名を入力します。ここでは、TUTORIALと入力します。
3. [保存]をクリックします。

参考：ツールバーのをクリックするか、またはCTRL+Sを押すことによってプログラムをセーブすることもできます。

まとめ

このレッスンでは、以下の操作を学習しました。

- ・[変数タイプ]ダイアログボックスでの変数の作成
- ・変数への変数タイプの割り付け
- ・プログラムのセーブ

3.3 ラング、命令、および分岐の挿入

ロジックプログラムを作成する最初のステップは、ラングの挿入です。

”新規作成”でファイルを開きます。画面には、下のような空白のプログラムが表示されます。

参照：第3章で使用するチュートリアルプログラムの完成品は、¥Pro-Control¥Sampleのフォルダの中にSoda1.wllファイルとして付属しています。

プログラム コメント



3.3.1. ラングの挿入

ロジックプログラムを新規作成します。

新しいプログラムの左側には、START、END、およびPENDというラベルが付いた3つのラングがあります。

- ・STARTラングは、メインプログラム領域の開始を表します。
- ・ENDラングは、メインプログラム領域の終わりを表します。
- ・PENDラングは、全プログラム領域の終わりを表します。PENDラングの後にはラングを挿入できません。

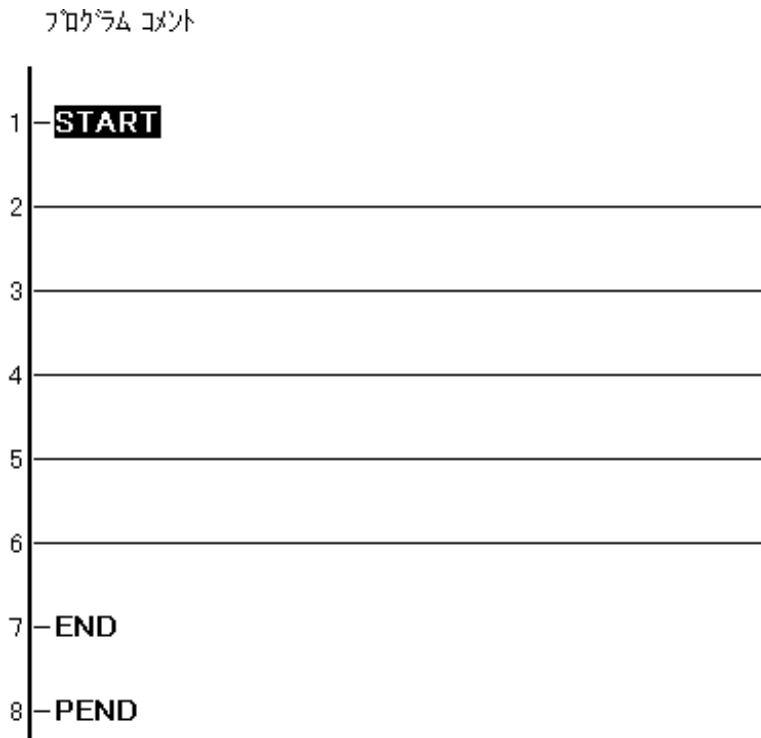
STARTとENDの間のラングは、毎スキャン実行されます。STARTの上に挿入されるラングは初期化プログラムを作成する領域です。この領域は立ち上げ時、1スキャンだけ実行されます。


ENDとPENDの間の領域は、サブルーチンプログラムのために予約されます。

START、END、およびPENDラングの詳細[参照 オンラインヘルプの[リファレンスガイド]]

ラングの挿入手順

1. STARTの左側のラング番号1をクリックします。
ラング1が選択されます。
2. 右クリックします。
ショートカットメニューが表示されます。
3. [ラングの挿入]を選択します。(または、「挿入」メニューより「ラング」を選択します。)
新しいラングが番号2 (STARTラングの下) に表示されます。
4. 上の方法を繰り返して、STARTラングの下にさらに4つのラングを挿入します。
画面は、次頁のようになります。

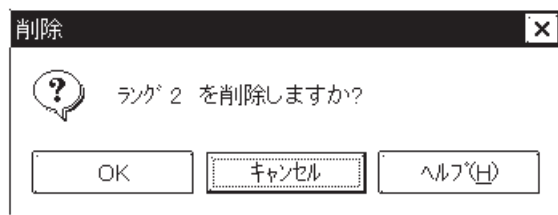


参考：挿入メニューから[ラング]を選択するか、ツールバーの  をクリックすることでも、ラングを挿入することができます。


3.3.2. ラングの削除

ラングの削除手順

1. 削除する部分を選択します。この例では、ラング2の左の番号2(ラング番号)をクリックします。
2. [Delete]キーを押します。または、右クリックしショートカットメニューより「ラングの削除」を選択します。削除ウィンドウが表示されます。



3. [キャンセル]をクリックします。

参考：Undoコマンドは、編集メニューから、[元に戻す]を選択するか、ツールバーの中の  をクリックで実行でき、ラング単位に1つ前の状態に戻すことができます。

3.3.3. 命令の挿入

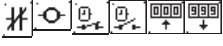
ロジックプログラムに命令を挿入し、それらに変数を割り付ける方法は数通り用意されています。このチュートリアルでロジックプログラムを作成する際に、これらの方法について説明し、実際に試してみます。

- 参考：
1. コントローラは、ロジックプログラム中の命令を左から右、上から下の順に解析し実行します。
 2. オンラインヘルプ[ロジックプログラムの実行順序]を参照してください。
 3. 命令を挿入する前に、どのラングに挿入するか指定しなければなりません。

命令を挿入するラングの選択手順


1. ここでは命令をラング2に挿入します。
ラング2のライン上のどこかをクリックします(ただし2という番号そのものをクリックしてはいけません)。選択されたラングが下のように表示されます。



2. 選択したラングにツールバーから命令を挿入できます。Editorのツールバーには、下に示すようなボタンがあります。 これらのボタンをクリックすると、選択したラングに命令が挿入されます。
ボタンの意味は、下表の通りです。


	a接点(NO)
	b接点(NC)
	アウト・コイル(OUT)
	オンディレータイマ(TON)
	オフディレータイマ(TOF)
	アップカウンタ(CTU)
	ダウンカウンタ(CTD)

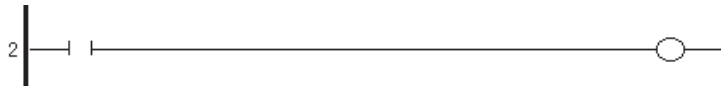
命令挿入手順1: ツールバーからの挿入

1. をクリックします。下のようラング2が表示されます。




命令の上に入力ボックスが表示され、その中でカーソルがフラッシュしています。これは、命令パラメータボックスです。通常では、このボックスは命令に関連付ける変数名を入力しますが、ここでは無視してください。

2. をクリックします。ラング2の右側にアウト・コイルが挿入されます。命令パラメータボックスがフラッシュしていますが、ここでも無視してください。変数の入力方法はあとで説明します。[参照 [3.4. : 命令への変数の割り付け]]



3. a接点(NO)とアウト・コイル(OUT)命令の間のラング2をクリックします。

4. b接点(NC)ボタンをクリックします。

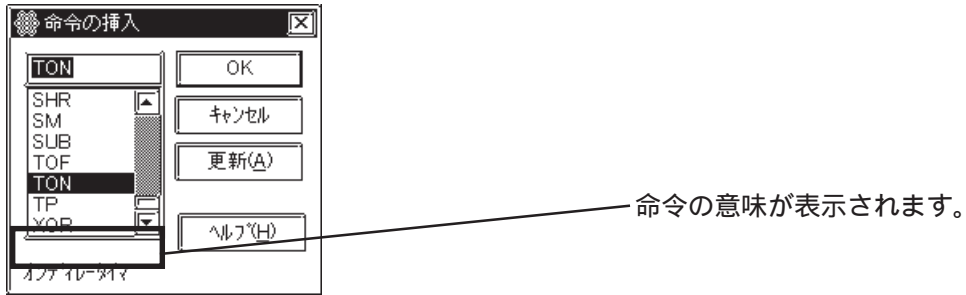
ラング2にb接点(NC)が挿入され、下のように表示されます。



参考: 各ツールボタンの機能の説明は、カーソルをそのボタンの上に置いたときにステータスバー上に表示されます。ツールバーは、頻繁に使用する命令を挿入するのに便利ですが、Editorで利用できるすべての命令を含んでいるわけではありません。以下の2つの方法を使って、[命令の挿入]ダイアログボックスから命令を挿入することもできます。

命令の挿入手順2：ダイアログボックスでのリスト選択による挿入

1. ラング3上のどこかを右クリックします。ショートカットメニューが表示されます。
2. [命令の挿入] を選択します。[命令の挿入] ダイアログボックスが表示されます。



このダイアログボックスには、Editorでロジックプログラムを作成するために使用できるすべての命令が含まれています。各命令を入力またはクリックすると、その命令の意味がダイアログボックスの下に表示されます。

参考:挿入メニューから[命令]を選択するか、ラングを選択した後で[INSERT]キーを押すことによって、[命令の挿入]ダイアログボックスを表示することもできます。すべての命令の詳しい説明は、命令を選んだオンラインヘルプの関係トピックヘルプを選択することで見ることができます。

3. ここでは、オンディレータイマを選択します。オンディレータイマ(TON)が見つかるまで、命令リストをスクロールします。
4. オンディレータイマ(TON)を選択します。
[変数タイプ]ダイアログボックスと同様に、選択を確認する際に、[OK]または[更新]のどちらかをクリックします。ここでは、ロジックプログラムにさらに命令を挿入しますから、[命令の挿入]ダイアログボックスを開いたままにしておくために[更新]をクリックします。



5. 先ほど挿入したオンディレータイマ(TON)命令の左側のライン上で接続線(コマンドとコマンドの間のポイント)をクリックします。
6. a接点(NO)が見つかるまで、命令のリストをスクロールします。
7. a接点(NO)をダブルクリックします。



命令挿入手順 3: ダイアログボックスでの入力による挿入

1. 命令リストの上のフィールドに OUT と入力します。

参考: 命令リストが自動的にスクロールして、アウト・コイル(OUT)命令がリストの最上段に表示されます。また、その名前がウィンドウの左下隅に表示されます。



2. オンディレータイマ(TON)命令の右側のラングを選択します。

3. [更新]をクリックします。
ラング3が下のように表示されます。



3.3.4. 命令の削除

ラング3に挿入したアウト・コイル(OUT)命令を削除します。

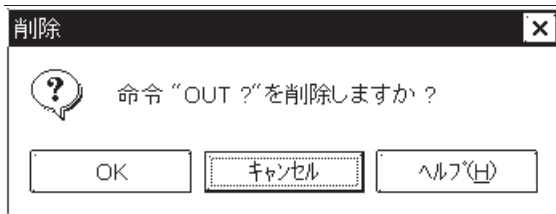
命令の削除手順


1. ラング3のアウト・コイル(OUT)命令を右クリックします。

ショートカットメニューが表示されます。

2. [削除]を選択します。

ウィンドウが表示され、この命令を削除してもよいかどうかを尋ねます。



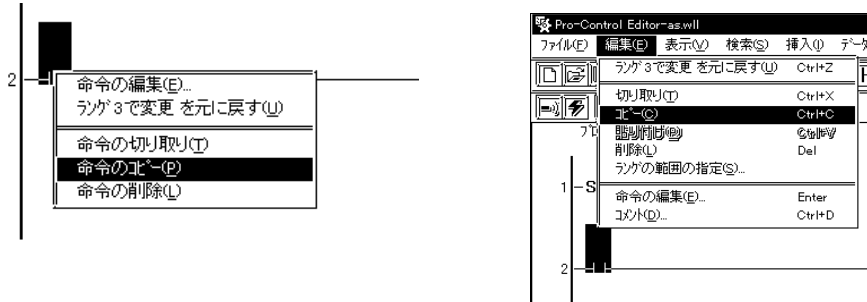
3. [OK]をクリックします。参考: 削除する命令を選択して[Delete]キーを押すか、またはツールバーの  をクリックすることでも削除できます。

3.3.5. 命令のコピー & 貼り付け

ここでは、命令のコピー & 貼り付けを練習します。

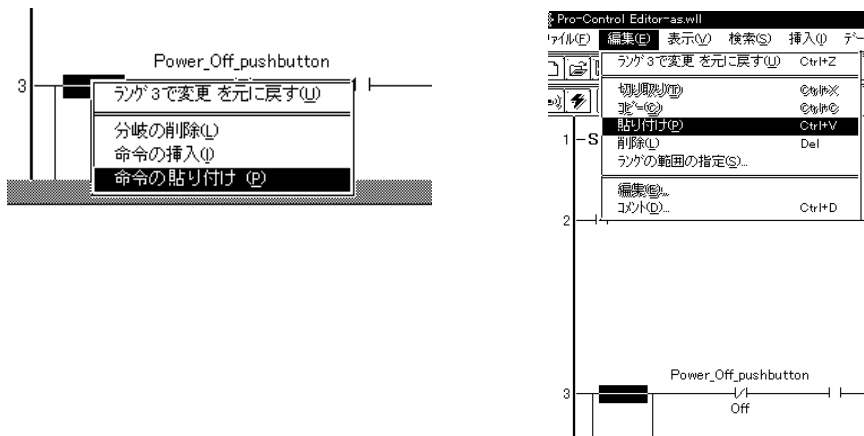
命令をコピーする手順

1. コピーしたい命令をクリックします。
2. 右クリックで[命令のコピー(P)]もしくは、メニューより[編集(E)] [コピー(C)]を選択します。

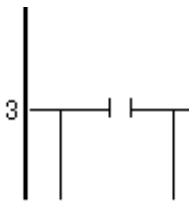


命令を貼り付けする手順

1. 命令を配置したい場所を選択します。
2. 右クリックで[命令のコピー(P)]もしくは、メニューより[編集(E)] [貼り付け(P)]を選択します。

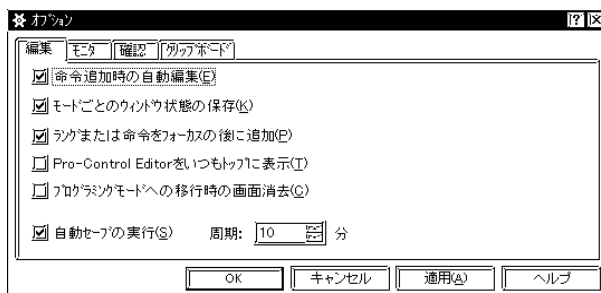


3. 以下のように命令が貼り付けられます。



命令の自動編集モード

メニューの[ファイル(F)] [オプション(O)] [命令追加時の自動編集(E)]を選択すると命令に変数が割り付けられていないとき、変数名の入力状態になります。

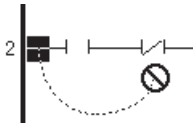


3.3.6. 分岐の挿入

この練習では、分岐をラング2のa接点(N0)とb接点(NC)の間に挿入します。この分岐はソーダポップマシンのライトをラッチするために設計されています。

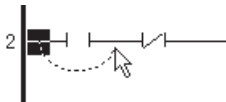
分岐を挿入する手順

1. ラング上の分岐を開始したいポイントにカーソルを置きます。この場合は、ラング2のa接点(N0)のすぐ左です。
2. マウスをクリックし右にドラッグします。カーソルが \odot に変わり、そこから破線が描かれています。

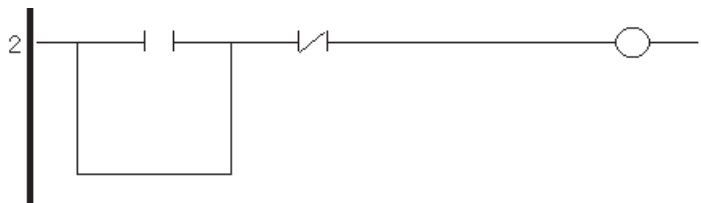


分岐の終点が無効な位置にある場合、カーソルが \odot のように表示されます。分岐の終点が有効な位置にある場合、カーソルは通常に戻ります。カーソルが通常形で表示されるポイントでマウスを離すと、開始点とマウスを離した終点の間に分岐が挿入されます。カーソルが \odot のように表示されているときにマウスを離すと、分岐は作成されません。

3. マウスをクリックして、カーソルがa接点(N0)とb接点(NC)の間に置かれ、 \odot ではなく通常形で表示されるまで右にドラッグします。

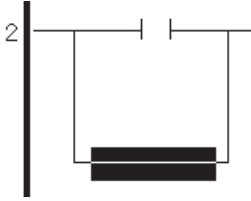


4. マウスを離します。これでa接点(N0)とb接点(NC)の間に分岐が挿入され、ラング2は下のようになります。

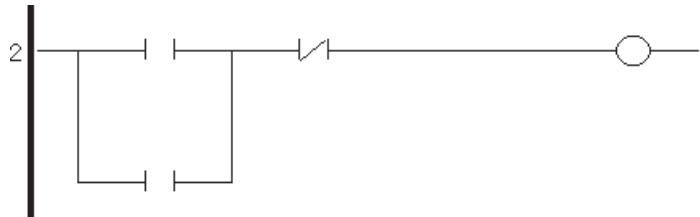


分岐に命令を挿入する手順

1. 分岐の下の部分をクリックして、分岐を選択します。



2. [命令の挿入]ダイアログボックスはまだ開かれていたはずですが、開いていない場合は、前述のいずれかの方法を使って、このウィンドウを開きます。
3. [命令の挿入]ダイアログボックスから a 接点(N0)命令を選択し、前述のいずれかの方法を使ってそれを挿入します。ラング 2 が下のように表示されます。



参考：命令が挿入されている分岐を削除するには、最初にその分岐の中のそれぞれの命令を選択して削除しなければなりません。

3.3.7. 初期化ロジックプログラム

START ラングの上に挿入されているロジックは、初期化ロジックプログラムです。このロジックプログラムは、コントローラを RUN したときに 1 回だけ実行されます。

初期化ロジック挿入手順

1. START ラングの上にある[プログラムのコメント]フィールドをクリックします。
このフィールドが表示されていない場合は、[表示]メニューから[コメント]を選択してから、[プログラム]を選択します。
2. ショートカットメニューから[ラングの挿入]を選択します。
ラングが START ラングの上に挿入されます。
参考:以降のラングは1つずつ繰り下がります(2番目にあったラングがラング3になります)。
3. 初期化ラング(ラング1)を右クリックします。
4. ショートカットメニューから[命令の挿入]を選択します。
5. [命令の挿入] ダイアログボックスから SET 命令を選択し、[OK] をクリックします。

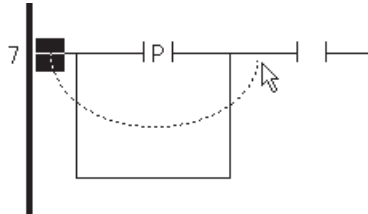


このラングは、ソーダ水製造機の製氷装置をオンにするために使用します。これは一度オンに設定しておけば、ソーダ水製造機の電源がオンになっている間、オンのままになっています。
参考:[オプション]ダイアログボックスで[ラングまたは命令をフォーカスの後に追加]を選択していない場合、ラングなどのオブジェクトは選択したライン上に挿入されます。初期化ラングを挿入するためには START ラングを選択してください。

これでロジックプログラムのラング3、4、および初期化ロジックプログラムの1つのラングが完成しました。次のページに示すようにラング5～7を作成してください。プログラムを完成させるために、下記のヒントが役に立ちます。| P | 命令とは、PT(立ち上がり接点)命令のことです。

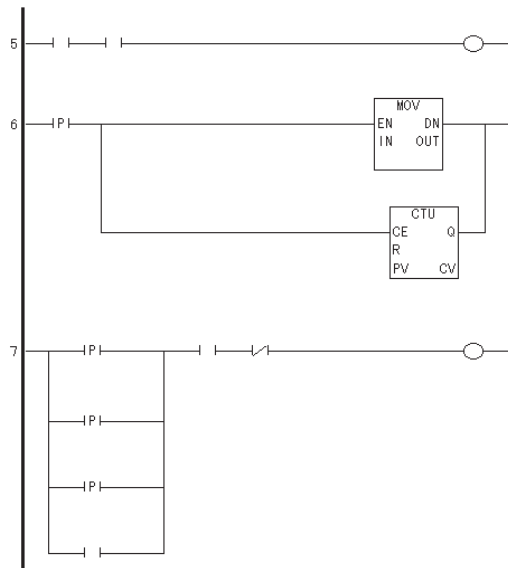
ラング7の複数の分岐を挿入する手順

1. 最初の分岐を前述の方法で挿入します。
2. 2番目の分岐を挿入するために、前の分岐と同じポイントをクリックし、そこからドラッグします。
3. カーソルを、前の分岐を越えて、新しい分岐を挿入するラングのポイントまでドラッグします。



マウスを離すと、新しい分岐が前の分岐の上に挿入されます。

下の例では、命令がラング5～7に挿入されています。



まとめ

このレッスンでは、以下の方法を学習しました。

- ・ラングの挿入と削除
- ・命令の挿入と削除
- ・分岐の挿入と削除

3.4. 命令への変数の割り付け

命令に変数を割り付ける方法を学びます。

3.2で、チュートリアルロジックプログラムで使用する変数リストを作成しました。ここでその[変数リスト]ウィンドウを開きます。

[変数リスト]ウィンドウのオープン手順

1. [データ]メニューから、[変数リスト]を選択します。
2. このウィンドウを画面の左下隅に移動します。[命令の挿入]ダイアログボックスがまだ開かれている場合は、キャンセルをクリックして、閉じます。

3.4.1. 命令パラメータボックス

前のレッスンで、最初に命令をラングに挿入したとき、カーソルがフラッシュしているフィールドが表示されたことを思い出してください。このフィールドが命令パラメータボックスです。このボックスに、命令に関連付ける変数を入力します。

基本命令パラメータボックスへのアクセス手順

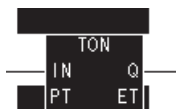
1. ラング3のアウト・コイル(OUT命令)をダブルクリックします。命令の上にテキストフィールドが表示され、その中のカーソルがフラッシュします。これがこの命令の命令パラメータボックスです。

参考：命令パラメータボックスは、命令をクリックして[Enter]を押すか、または命令を右クリックして、ショートカットメニューから[命令の編集]を選択することによってもアクセスすることができます。

応用命令には、2つ以上の命令パラメータボックスがあります。たとえば、オンディレータイマ(TON命令)には、2つの命令パラメータボックスがあります。1つの命令パラメータボックスで変数を割り付け、もう1つの命令パラメータボックスでは、設定時間(ミリ秒単位)を入力します。

応用命令パラメータボックスへのアクセス手順

1. ラング4のオンディレータイマ(TON命令)をクリックします。命令が下のように表示されます。



オンディレータイマ(TON命令)の上の領域が黒く反転表示されています。この領域に、命令に割り付ける変数を入力します。もう1つの黒く反転表示されている領域は、設定時間(PT)の専用変数です。この領域には、設定時間(ミリ秒単位)を入力します。

2. オンディレータイマ(TON命)の上の黒く反転表示されている領域をダブルクリックします。これで命令パラメータボックスが選択されました。このとき、命令にタイマ変数(変数名)を割り当てることができます。



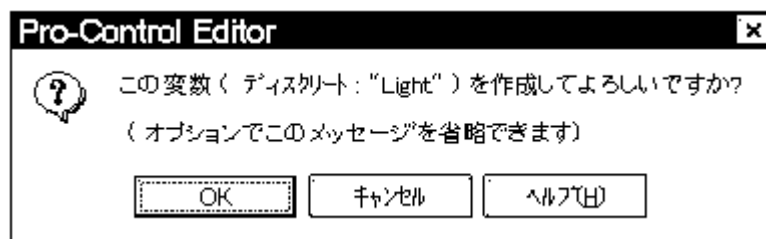
3. オンディレータイマ(TON命令)のPTのすぐ左にある領域をダブルクリックします。 [データ値変更]ダイアログボックスが表示されます。このウィンドウには、タイマ出力(Q)がオンになるまでの設定時間(ミリ秒単位)を入力します。命令に変数と他のオペランドを割り付ける方法については、次の項で説明します。
4. [データ値変更]ダイアログボックスを閉じます。

3.4.2. 変数の入力

命令パラメータボックスに変数を入力する1つの方法として、命令パラメータボックスの中に直接入力することができます。

命令パラメータボックスへのテキスト入力手順

1. 命令をダブルクリックして、ラング3のアウト・コイル(OUT命令)の命令パラメータボックスを選択します。
2. このボックスに「Light」と入力します。
3. [Enter]を押します。下のようなウィンドウが表示され、変数の作成を行うか確認します。



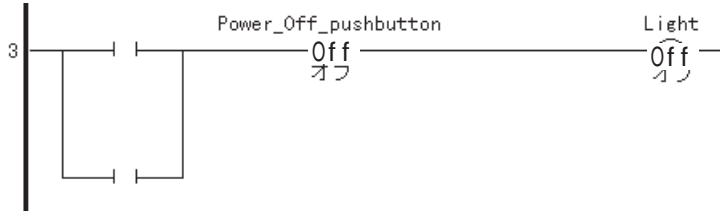
4. [OK]をクリックします。[変数リスト]ウィンドウのリストの中にLightという変数が表示されています。Editorは、命令に必要なデータタイプを自動的に変数に割り付けます。この場合、変数タイプとしてインターナルのディスクリートが割り付けられています。

参考: Editorは、命令に対して作成された新しい変数に必要なデータタイプを自動的に割り付けます。

また、変数リストにすでにある変数を命令パラメータボックスの中に直接入力することもできます。入力が完了すると、自動的に変数が割り付けられます。上記の方法を使って、オペランド Power_Off_pushbutton をラング3のb接点(NC命令)に割り付けます。

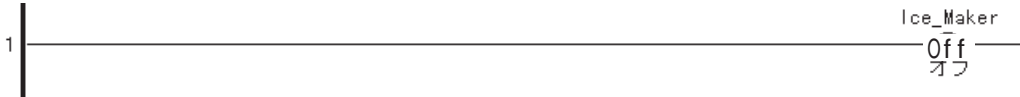
参考: コイル命令(OUT、SET、RST、NEG)に割り付けた変数を保持型に変えると自動的に保持型のコイル命令(M、SM、RM、NM)に変わります。

ラング3は、下のように表示されます。



初期化ラングのセット・コイルに変数 Ice_Maker を割り付けます。この変数は、命令パラメータボックスに直接入力することによって作成できます。

初期化ラングが下のように表示されます。

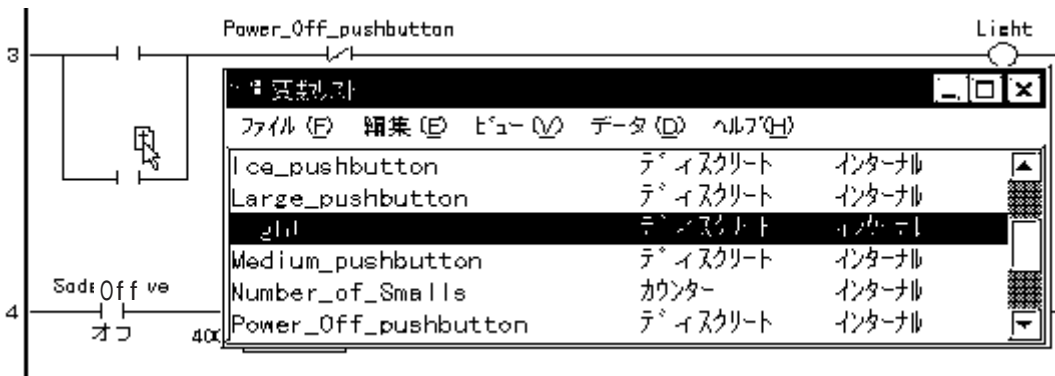


命令に変数を割り付けるもう1つの方法として、[変数リスト]ウィンドウの中の変数を命令にドラッグ&ドロップすることもできます。この方法は、多数の命令に同じ変数を割り付ける場合に便利です。この方法を使う利点は、[3.9 : I/Oの割り付け]で詳細を説明します。

[変数リスト]ウィンドウによるマッピング手順

1. [変数リスト]ウィンドウのLightをクリックし、そのままマウスボタンを押したままにします。
2. マウスボタンを押したまま、Light をラング3の分岐の上のa接点(N0命令)までドラッグします。分岐を挿入する場合と同様に、カーソルは最初は⓪になります。カーソルがこの状態のとき、命令に変数を割り付けることはできません。
3. カーソルをラング3の分岐の上のa接点(N0命令)の上にドラッグします。

カーソルは、下のように表示されます。

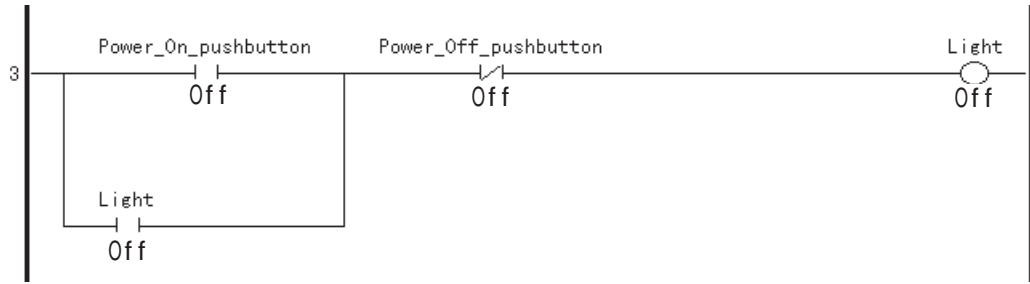




カーソルが ⓪ のときにマウスを離すと、変数が割り付けられます。

これで変数 Light がラング3の分岐のa接点(N0命令)に割り付けられました。

4. 変数 Power_On_pushbutton をクリックし、ラング3上の他のa接点(N0命令)にドラッグします。

ラング3は、下のように表示されます。



[参考]カーソルを命令のごく近くまで移動すると、カーソルは自動的に  から  に変わります。この時点でマウスを離すと、変数とその命令に割り付けられます。定数を変数として割り付けることもできます。入力方法は、通常の変数の場合と全く同じですが、ウィンドウからドラッグできないため、手操作で入力する必要があることです。

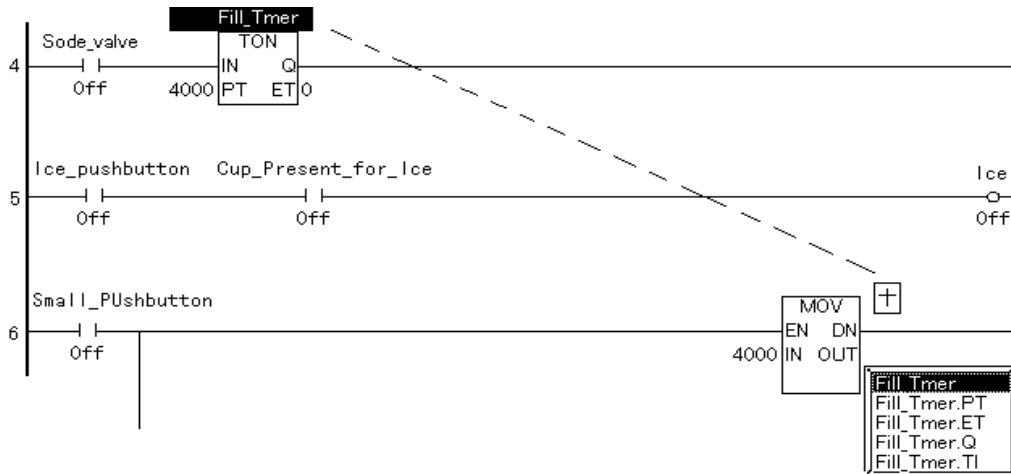
カウンタ変数・タイマ変数の要素をコピーする手順

カウンタやタイマの変数を他の命令にコピーする場合、カウンタやタイマの要素がリストボックスで表示されます。

[注意]

カウンタをカウンタ、タイマをタイマに割り付ける場合は、変数そのままコピーされます。

1. カウンタ・タイマの変数をコピーしたい命令にドラックします。
2. リストボックスより要素を選択します。



3.4.3. 作業の完了

これで、命令に変数を割り付ける方法がわかりました。プログラムの残りのラングを完成させてください。完成したラングは、次ページの図のようになります。

ラング6のMOV命令とラング7のb接点(NC命令)には、それぞれ変数 Fill_Timer.PTと Fill_Timer.Qが割り付けられています。これらの変数は、Fill_timer変数が割り付けられているタイマのPTおよびQを参照します。

これらの変数を入力する手順には、以下の2つがあります。

- ・ 命令パラメータボックスを選択し、その中に変数Fill_Timerを直接入力する。
- ・ [変数リスト]ウィンドウからFill_Timer変数をクリックしドラッグして入力する。

[重要]

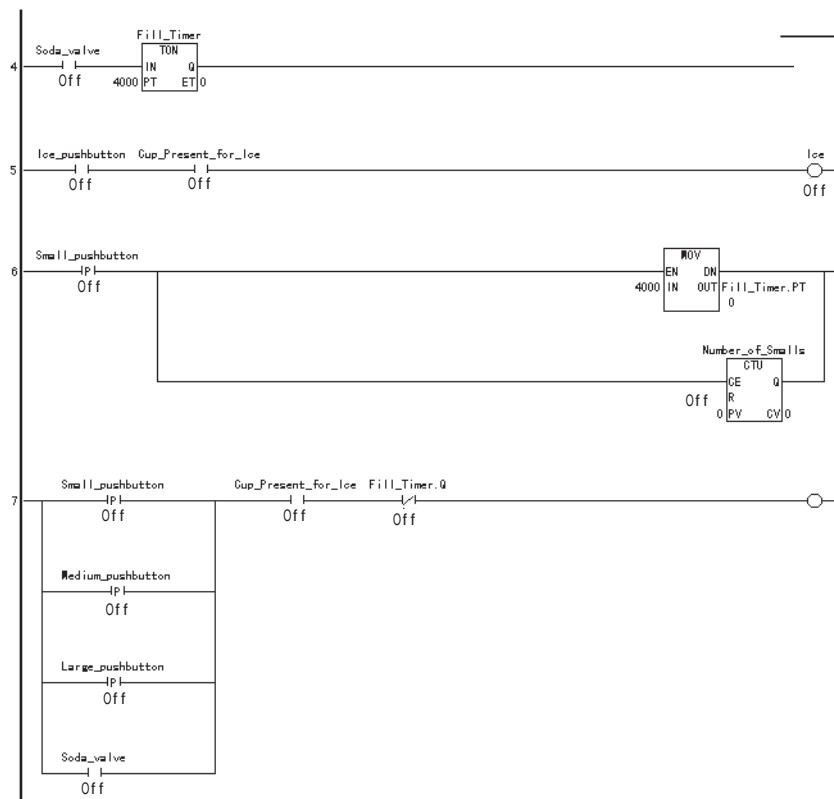
ラング6、7などで使用される。Fill_Timer.PTやFill_Timer.Qのように応用命令の専用変数は、変数名+拡張子の形で使用できます。

命令によりますが、拡張子として、

- ***.CV 現在値
- ***.PT 設定値
- ***.Q 出力ビット
- ***.R リセットビット

がプログラム上で使用できます。

<チュートリアルプログラムの完成例>



まとめ

3.4. では、命令にオペランドを割り付ける方法を学習しました。

3.5. ロジックプログラムのドキュメント化

Editor では、1. プログラム全体、2 各ラング、3. 各変数にコメントを付加することができます。ロジックプログラムにコメントを付加しておく、再読性が上がりデバックや追加修正の際に有効です。

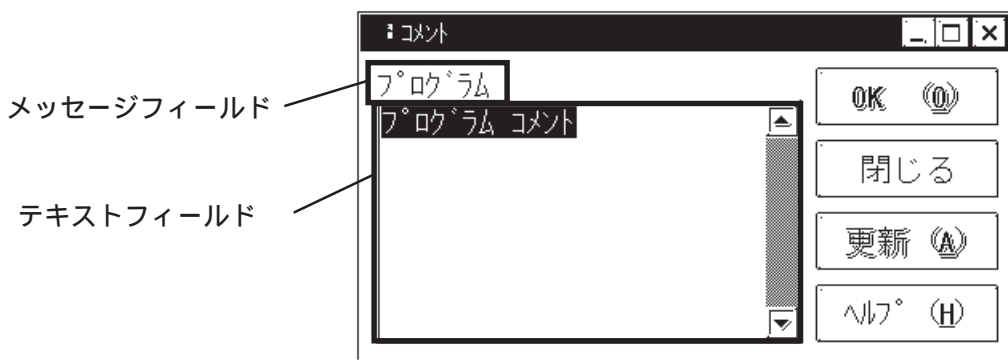
3.5.1. プログラムコメントの追加

ロジックプログラムに追加する最初のコメントは、プログラムの機能のコメントです。

プログラムコメントの追加手順

1. 画面最上段の[プログラムコメント]をダブルクリックします。

画面に[コメント]ダイアログボックスが表示されます。



このウィンドウでプログラムのコメントを入力してください。

参考:[コメント]ダイアログボックスのメッセージフィールドの上に、[プログラム]と表示されています。これは、このテキストフィールドがプログラムのコメントの入力ボックスであることを示しています。

2. [プログラムコメント]テキストを選択します。

3. 「このプログラムは、ファーストフードレストランのソフトドリンク製造機を制御します。」と入力します。

4. [OK]をクリックします。プログラムコメントがロジックプログラムの最上段に表示されます。表示されない場合は、画面を上スクロールしてください。

このプログラムは、ファーストフードレストランのソフトドリンク製造機を制御します。




参考:下側のステータスバーをダブルクリックしても、[プログラムコメント]の追加または編集ができます。

3.5.2. ラングコメントの追加

Editorでは、プログラムの各ラングにコメントを追加できます。下の例では、ラング5にコメントを追加します。

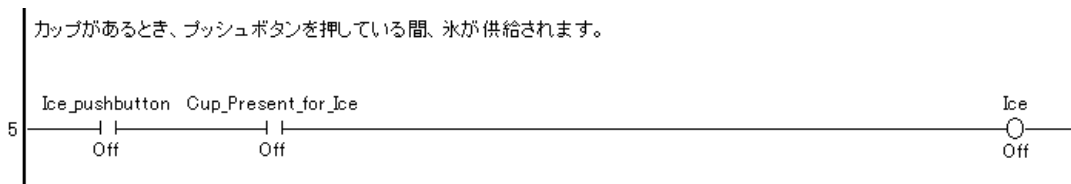
ラングコメントの追加手順

1. ラング5の左にある番号5を右クリックします。
2. ショートカットメニューから[コメント]を選択します。[コメント]ダイアログボックスが開きます。このウィンドウは、プログラムコメントを挿入したときにオープンしたウィンドウと同じです。唯一の違いは、メッセージフィールドの上のコメントが[プログラム]ではなくラング5になっていることです。

参考：編集メニューから[コメント]を選択するか、またはツールバーのをクリックすることも、[コメント]ダイアログボックスをオープンできます。

プログラムのラング5は、アイスディスペンサーの制御のために使用されます。

3. [コメント]ダイアログボックスのテキストフィールドをクリックします。
4. 「カップがあるとき、プッシュボタンを押している間、氷が供給されます。」と入力します。
5. [更新]をクリックします。



[コメント]ダイアログボックスを開いておくと、プログラムの残りのラングにコメントを追加するのが簡単になります。

ラング3へのコメントの追加手順

1. ラング3上の命令パラメータボックス以外の場所をクリックします。
[コメント]ダイアログボックスの最上段のコメントがラング3に変わります。
2. テキストフィールドをクリックします。
3. 「電源オフプッシュボタンを押すまで、ライトは点灯したままになります」と入力します。
4. [更新]をクリックします。このチュートリアルでは、ラング3および5だけを例にあげドキュメント化しました。

3.5.3. 変数へのコメントの追加

ロジックプログラムの各変数にコメントを追加できます。ラベルまたは定数にはコメントを追加できません。

変数コメントの追加手順

1. 本チュートリアルの流れでは、[変数リスト]ウィンドウが開かれているはずです。
開いていない場合は、[データ]メニューから変数リストを選択して、開きます。
2. [コメント]ダイアログボックスも開かれているはずです。
開いていない場合は、編集メニューから[コメント]を選択して、開きます。
3. 変数 Fill_Timer を表示している命令パラメータボックスをクリックします。
[コメント]ダイアログボックスのメッセージフィールド上に Fill_Timer というメッセージが表示され、[変数リスト]ウィンドウでも Fill_Timer が反転表示されます。
4. [コメント]ダイアログボックスのテキストフィールドをクリックします。
5. 「Fill_Timer は、ソーダバルブを開けておく時間を決定します。設定時間は設定値に依存します。」と入力します。
6. [更新]をクリックします。

参考: 変数にコメントを追加する方法として、変数をロジックプログラムから選択する方法のほかに、[変数リスト]ウィンドウから選択する方法もあります。

変数リストからコメントを追加する手順

この例では、Power_On_pushbutton にコメントを追加します。

1. [変数リスト]ウィンドウの中の変数 Power_On_pushbutton をクリックします。
[コメント]ダイアログボックスのメッセージフィールドに Power_On_pushbutton というメッセージが表示されます。
2. [コメント]ダイアログボックスのテキストフィールドをクリックします。
3. 「電源オンプッシュボタンを押すと、ソフトドリンク製造機が起動します。」と入力します。
4. [更新]をクリックします。

このチュートリアルでは、Fill_Timer 変数と Power_On_pushbutton 変数のコメントだけを例にあげ追加しました。他のすべての変数のコメントも、同じ方法で作成できます。

3.5.4. [コメントリスト]ウィンドウ

[コメントリスト]ウィンドウは、プログラム中のすべての変数およびラングのコメントを一覧表示します。

[コメントリスト]ウィンドウを表示する方法

[表示]メニューから、[コメントリスト]を選択します。

[コメントリスト]ウィンドウから詳細なコメントを表示する方法

コメントリストウィンドウの中のFill_Timer変数をダブルクリックします。

[コメント] ダイアログボックスに、Fill_Timer変数の詳細なコメントが表示されます。

[変数リスト]、[コメント]、[コメントリスト]の表示内容はプログラム中のラングや変数を選択するのに追従し選択表示されますが、[変数リスト]、[コメント]、[コメントリスト]表示内容を選択してもロジックプログラムの上は追従しません。

Editorでは、プログラムロジックの中から特定の変数を簡単に見つける方法があります。これは、[3.8. : ロジックプログラム内の移動]で詳しく説明します。

まとめ

プログラム、ラング、および変数にコメントを追加し、[コメントリスト]ウィンドウを表示する方法を学習しました。

3.6. ラングのコピー、切り取りおよび貼り付け

ロジックプログラムでは、いくつかのラングで同じ命令シーケンスを入力しなければならないことがあります。すでに作成したラングをコピーして貼り付けることによって、工数を削減できます。

3.6.1. ラングのコピー

以下のチュートリアルでは、2つのラングがラング5とラング7の間に追加されます。これらの追加のラングには、ラング6と同じ命令があり、異なる変数が割り付けられています。

ラングのコピー手順

1. ラングの左の番号6をクリックして、ラング6を選択します。
2. [編集]メニューから、[コピー]を選択します。

参考：切り取りまたはコピーするラングの範囲を選択する方法。

選択する範囲の中の先頭ラング番号をクリックします。次に、[Shift]キーを押したまま、選択する最後のラング番号を選択します。こうすると2つのラングの間にあるすべてのラングが選択され、切り取りまたはコピーすることができます。一度の操作でコピーできるラングの数は、25ラング程度が目安です(ラング内の命令数によって前後します)。

3.6.2. ラングの貼り付け

ラング全体が選択されている場合を除いて、Editorは、コピーまたは切り取りされたラングを、現在選択されているラングの下に貼り付けます。

[オプション]ダイアログボックスで[ラングまたは命令をフォーカスの後に追加]が選択されていない場合、コピーされたラングは、現在のラングの上に挿入されます。

【重要】

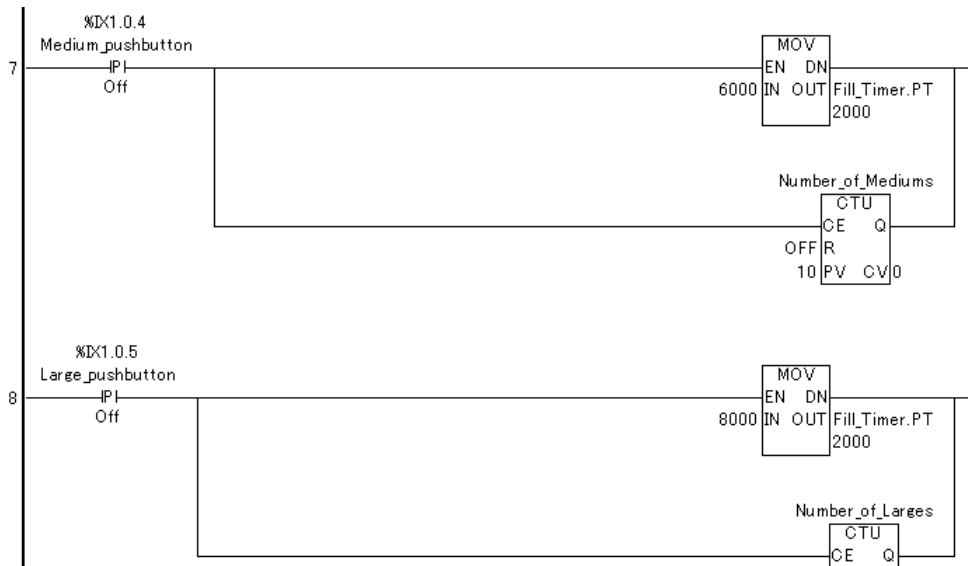
切り取り、コピーされたラングは、一旦内部のクリップボードにロードされます。

クリップボードからラングを貼り付けるとき、ラング全体を選択している場合、Editorは、全ラングをクリップボード上のラングに置換します。

ラングの貼り付け手順

1. ラング6上のどこかをクリックします。
2. 編集メニューから[貼り付け]を選択します。ラング6とラング7が同じになります。
3. ラング6上のどこかをクリックします。
4. 編集メニューから、[貼り付け]を選択します。ラング6~8がすべて同じように表示されません。

参考: ラングを貼り付けしたとき、そのラングに関連付けられているすべての変数およびコメントもそのまま貼り付けられます。目的のプログラムに合わせてラングを編集してください。このチュートリアルでは、ラング7および8の変数を、下の図のように変更してください。



5. 上の例のようにラング上のPT命令の変数名を変更します。

3.6.3. [切り取り]コマンドの使用

Editorでは、プログラムの一部からラングまたはラングの一部を取り出し、それを他の部分へ移動できます。これは、[切り取り]コマンドを使って実行します。以下の練習では、ラング4をプログラムの最後のラングへ移動します。

[切り取り]コマンドの使用手順

1. ラング4のラング番号をクリックします。ラング4が選択されます。
2. 編集メニューから、[切り取り]を選択します。
ラングが、ロジックプログラムから切り取られ、クリップボードに入れられます。
3. ラング8上のどこかをクリックします。
4. 編集メニューから、[貼り付け]を選択します。ラング4がラング8の下に付加されます。
プログラムの終わりは、下の図のようになります。



参考: ラング全体をプログラムの他の部分に移動するには、最初にラングを選択し、その中央部をドラッグして、新しい位置まで移動できます。

まとめ

ラングをコピー、切り取り、および貼り付ける方法を学習しました。

3.7. サブルーチンおよびラベル

JSR(ジャンプサブルーチン)またはJMP(ジャンプ)命令が挿入されると、コントローラは、そのサブルーチンまたはラベルまでジャンプして実行します。サブルーチンとラベルの主な違いとして、JSR命令では、ENDとPENDの間に作成された同一名サブルーチンを実行したあと、ロジックプログラムの中のJSR命令の次の位置に戻りますが、JMP命令ではJMP命令と同名のラベルにジャンプしロジックプログラムの実行を続け、そのままでは元のJMP命令の位置には戻りません。

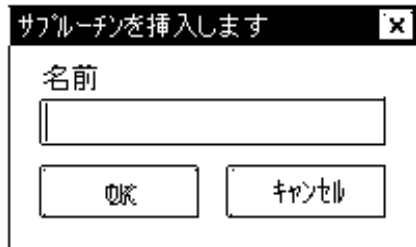
JMP命令およびJSR命令の詳細[参照 Pro-Control オンラインヘルプ]

3.7.1. サブルーチンの挿入

すべてのロジックプログラムの終わりには、ENDおよびPENDというラベルが付いています。ENDラベルは、メインプログラム領域の終わりを知らせます。Editorは、スキャンごとにSTART～ENDまでの命令を実行します。ENDラベルとPEND(プログラムの終わり)ラベルの間の領域はサブルーチンのために予約されています。以下のチュートリアルで、サブルーチンを追加します。

サブルーチンの追加手順

1. ENDラベルをクリックします。
2. 挿入メニューから、サブルーチンを選択します。
[サブルーチンを挿入します]ダイアログボックスが表示されます。



3. [サブルーチンを挿入します]ダイアログボックスの名称フィールドに、Reset_Countersと入力します。サブルーチン名/ラベルは、最大半角32文字、全角16文字以内で設定されます。サブルーチン名/ラベルを数字で始めないでください。サブルーチン名は、“_”以外の記号は使用できません。大文字と小文字、および全角と半角は区別しません。[参照 「変数リストの作成」]
4. [OK]をクリックします。ENDとPENDの間に、挿入されたサブルーチンが表示されます。

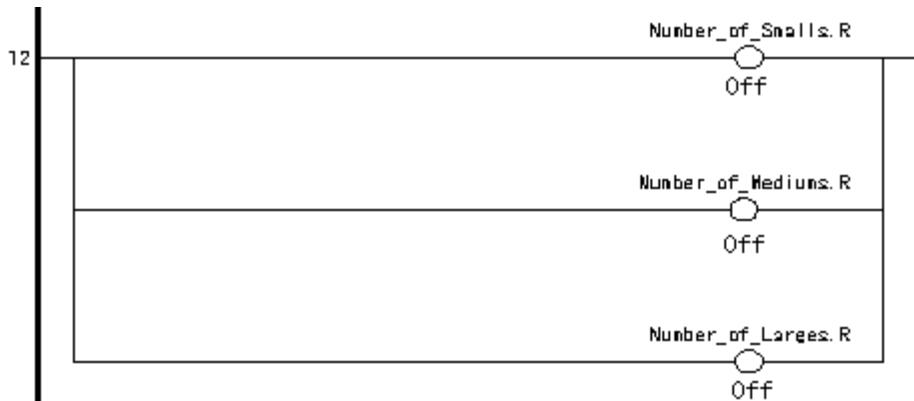
```

10—END
11—SUB STARTReset_Counters
12—SUB ENDReset_Counters
13—PEND

```

SUB STARTReset_CountersおよびSUB ENDReset_Countersというラベルが付いた2つの新しいラングが、ENDラングとPENDラングの間に挿入されました。サブルーチンを挿入する場所は、これらの2つのラングの間です。

5. SUB STARTReset_Counters ラベルを右クリックします。
6. ショートカットメニューから[ラングの挿入]を選択します。ラングがSUB STARTラングとSUB ENDラングの間に挿入されます。
7. SUB STARTラングとSUB ENDラングの間にあるラングを右クリックします。
8. そのラングにアウト・コイル(OUT 命令)を挿入します。
9. アウト・コイル(OUT 命令)の両側に2つの分岐を挿入します。
10. それぞれの分岐にアウト・コイル(OUT 命令)を挿入します。下の図は完成したサブルーチンです。
これは、GLCに電源を入れるたびに各カウンタをリセットするために使用されます。



各アウト・コイル(OUT 命令)に変数が割り付けられます。例のようにこれらの変数をここで割り付けてください。

以上でサブルーチンが完成しました。ロジックプログラムに2つ以上のサブルーチンを追加するには、SUB STARTラングまたはPENDラングを選択し、手順2)～6)を繰り返します。サブルーチンをロジックプログラムの中の特定のポイントで実行する場合、JSR命令を挿入しなければなりません。

次にこの操作を練習します。このサブルーチンは、ラング3のLight出力コイルがオンになったときに実行させます。したがって、この例では、JSR命令をラング4に置かなければなりません。

JSR 命令の挿入手順

1. ラング3を選択します。
2. 挿入メニューから、[ラング]を選択します。
3. ラング4にPT命令を挿入します。
4. 変数LightをPT命令に割り付けます。
5. [命令の挿入]ダイアログボックスを使って、JSR命令をPT命令の右に挿入します。
6. JSR命令の命令パラメータボックスの中に、サブルーチンの名前Reset_Countersを入力します。ラングは下のように表示されます。



JSR命令Reset_Countersが検出されたとき、サブルーチンReset_Countersにジャンプします。サブルーチンの実行が完了した後、ラング5から実行を再開します。

参考:サブルーチンを削除するには、はじめにそのサブルーチンの中の個々のラングを削除し、SUB STARTラングを削除してください。(SUB ENDラングは、SUB STARTラングと一緒に削除されます。)

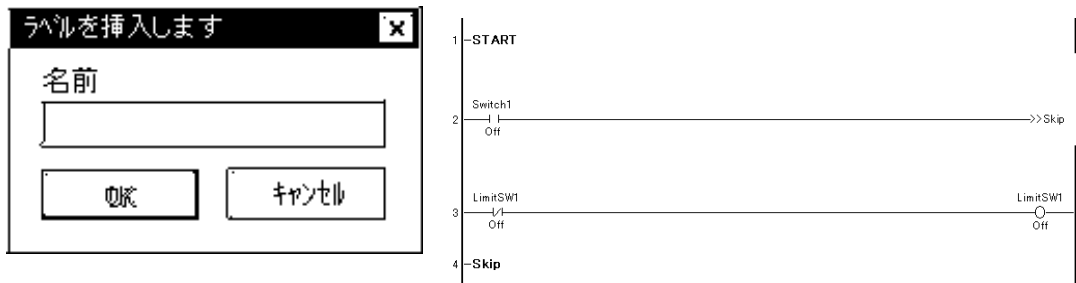
3.7.2. ラベルの挿入

JMP(ジャンプ)命令とジャンプ先のラベルをロジックプログラムの任意の部分に挿入できます。コントローラがJMP命令を実行したとき、JMP命令は、それに割り付けられている同名のラベルにジャンプし、そこからプログラムを実行します。

[オプション]ダイアログボックスの中で[ラングまたは命令をフォーカスの後に追加]が選択されているかどうかによって、ラベルは選択されているラングの上または下に挿入されます。このチュートリアルではラベルを使用しませんが、ラベルを挿入するには、以下の手順でラベルは挿入できます。

ラベルをロジックプログラムに割り付ける手順

1. ラング上のどこかをクリックします。
2. [挿入]メニューから、[ラベル]を選択します。[ラベルを挿入します]ダイアログボックスが表示され、ラベルの名前の入力を要求してきます。



これは、ラダー図の中のJMP命令に割り付けられる変数と同名にしてください。JMP命令は、同名のラベルにジャンプを行います。

JMP命令の挿入手順

1. ジャンプを開始するラングの最後の命令の右側をクリックし、[挿入]メニューから[命令]を選択します。
2. [命令]ダイアログボックスのJMP命令をダブルクリックします。JMP命令は必ずラングの最後の命令として挿入されます。Editorは、プログラムの中でこの命令を検出したとき、指定された同名のラベルにジャンプします。

まとめ

このレッスンでは、サブルーチンおよびラベルを作成する方法、およびJMP(ジャンプ)命令およびJSR(ジャンプサブルーチン)命令を挿入する方法を学びました。

3.8. ロジックプログラム内の移動

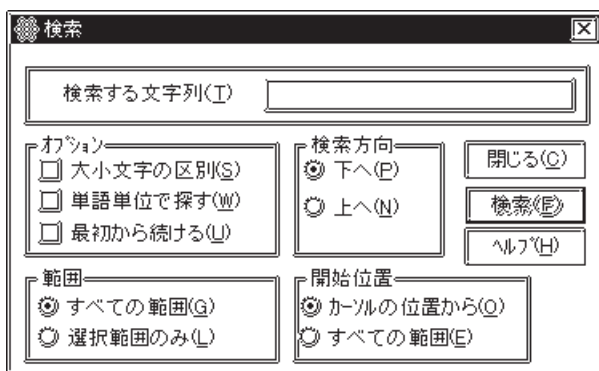
3.8. では、Editorで該当のラングや変数をすばやく検索する方法を説明します。Editorには、検索のために、[検索]、[リファレンス]、[ブックマーク]、[指定ラングへ移動]、および[指定ラベルへ移動]などのコマンドがあります。


3.8.1. [検索]コマンド

[検索]コマンドは、ロジックの中の特定のテキストを検索します。

[検索]コマンドの使用手順

1. 開かれているウィンドウがあれば、ここでそれを閉じます。
2. [検索]メニューから、[検索]を選択します。[検索]ダイアログボックスが表示されます。



参考:[検索]ダイアログボックスは、ツールバーの中のをクリックすることによってオープンすることもできます。

検索に使用する一致パターンのタイプの指定ができます。

- ・ Fillという語を探す場合、Editorは、fillという語を含むすべての語を検出します(小文字のfillや他の語の一部になっている場合、たとえばFilletも検出されます)。
- ・ [大小文字の区別]を選択した場合、Fillを検出しますが、fillを検出しません。また、[単語単位で探す]を選択した場合、Fillを検出しますが、Filletを検出しません。

検索の範囲および方向の指定

- ・ [選択範囲のみ]を選択している場合、検索の範囲は、プログラムの中の反転表示されている部分だけです。
- ・ [すべての範囲]を選択すると、検索の範囲は、プログラム全体を含みます。[すべての範囲]を選択することによって、選択範囲の最上段から検索を開始できます。また、[カーソルの位置から]を選択することによって、指定した位置から検索を開始できます。この例では、検索をプログラムの最初から開始します。

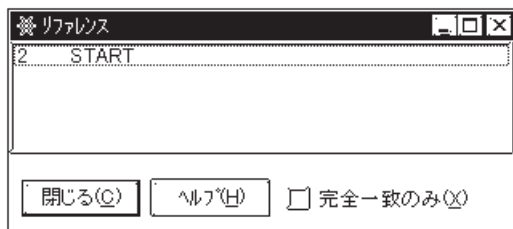
3. プログラムの中のSTARTラベルを選択します。
 4. [検索]ダイアログボックスの[検索する文字列]フィールドをクリックします。
 5. 「FILL」と入力します。
 6. [すべての範囲]、[下へ]、または[カーソルの位置から]を選択します。
 7. [検索]ボタンをクリックします。フォーカスは、検出した最初の一致、すなわちFill_Timer変数の最初の部分に移ります。
 8. [検索]ボタンをもう一度をクリックします。フォーカスは検出した2番目に一致するFill_Timerに移動します。一致するエントリがなくなった時、ピープ音が鳴ります。
- 参考：最初の[検索]コマンドの実行の後、[検索]メニューから[次を検索]を選択すれば、同じ条件で検索を続け、次に一致する文字列を検出できます。

3.8.2. [リファレンス]コマンド

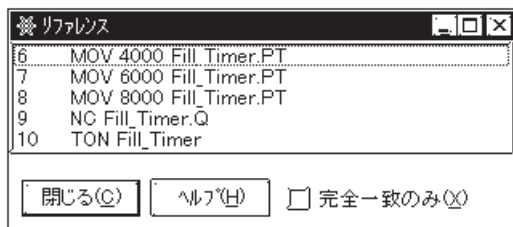
[リファレンス]コマンドは、ロジックプログラムの中の指定した変数が複数以上使用されている場合、使用されているすべての位置を検出します。このコマンドは、ラング番号とその変数が含まれる命令をリストします。このチュートリアルでは、STARTラベルを選択します。しかし、[リファレンス]コマンドは、プログラムの中のどこからでも実行できます。

[リファレンス]コマンドの使用手順

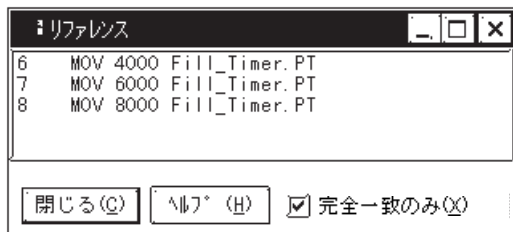
1. STARTラベルをクリックします。
2. [検索]メニューから、[リファレンス]を選択します。[リファレンス]ダイアログボックスが表示されます。



3. [リファレンス]ダイアログボックスのサイズを変更し、画面の右下隅に移動します。
4. ラング6の変数Fill_Timer.PTをクリックします。下のよう[リファレンス]ダイアログボックスが表示されます。



5. [完全一致のみ]を選択します。[リファレンス]ダイアログボックスは下のように表示されます。



[リファレンス]ダイアログボックスには、下記の情報が表示されます。

- 行の左の数値は、変数が検出されたラングの番号を示します。この画面は、Fill_Timerを含む変数がラング6、7、8、9、および10にあることを知らせています。[完全一致のみ]を選択したとき、画面は、Fill_Timer.PTがラング6、7、および8にあることを示します。
- 次の項目は、命令タイプです。ラング上で変数が割り付けられている命令です。この画面は、Fill_Timer変数が3つのMOV命令と、1つのb接点(NC命令)、および1つのオンディレータイマ(TON命令)に使用されていることを知らせます。
- 3番目の項目は、命令に割り付けられているパラメータをリストしています。(最初に参照した変数を含む)。例の6、7、8ラングでは、それぞれMOV命令のINには整数4000、6000、8000がOUTには、Fill_Timer.PTが割り付けられていることがわかります。

[リファレンス]ダイアログボックスに表示されたリストを選択するごとに、選択した変数の位置にラダー図の表示が追従します。プログラム修正や選択箇所モニタの際に一瞬で関連プログラムに移動できます。

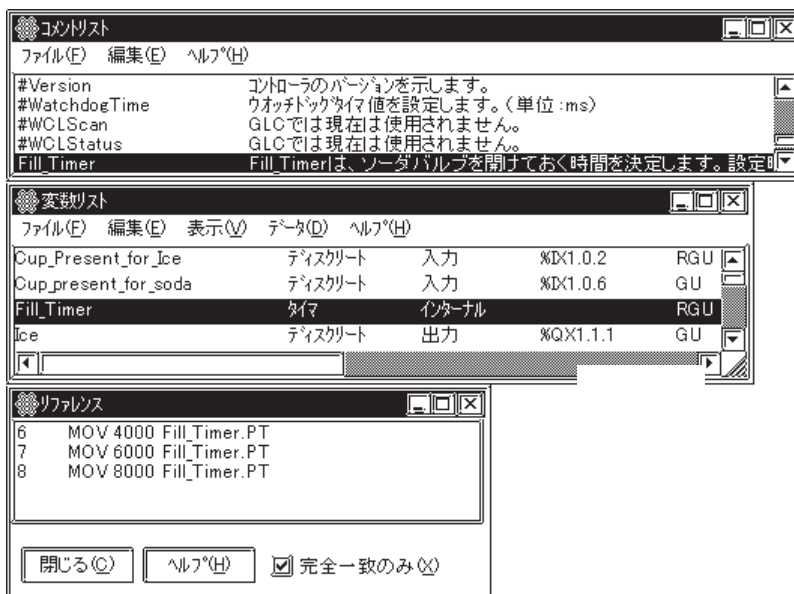
参考:[リファレンス]ダイアログボックスに対応する情報を表示するには、命令ではなくパラメータをクリックしなければなりません。

3.8.3. [リファレンス] ダイアログボックスと他のウィンドウの併用

[リファレンス]ダイアログボックスを使うときには、探している変数の少なくとも1つの所在がわかっていないと非常に面倒です。[検索]コマンドを使って見つけることもできますが、もっと早い方法があります。[リファレンス]ダイアログボックスと[変数リスト]ウィンドウおよび(または)[コメントリスト]ウィンドウを合わせて使用するという方法です。

[リファレンス]ダイアログボックスを他のウィンドウと共に使用する手順

1. [変数リスト]ウィンドウ、[コメントリスト]ウィンドウ、[リファレンス]ダイアログボックスを開きます。
2. これらのウィンドウを移動し、下の図のようにサイズを変更します。



3. [変数リスト]ウィンドウの中の変数Fill_Timerをクリックします。

参考:[コメントリスト]および[リファレンス]ダイアログボックスの表示は、選択に従って変化します。[リファレンス]ダイアログボックスには、変数Fill_Timerに含まれるすべての項目が表示されています。また、これらのウィンドウの表示が変化しても、ロジックプログラムの表示位置は変化しません。[リファレンス]ダイアログボックスで任意の変数行を選択すると、ロジックプログラムの中の変数に対応するポイントが表示されます。

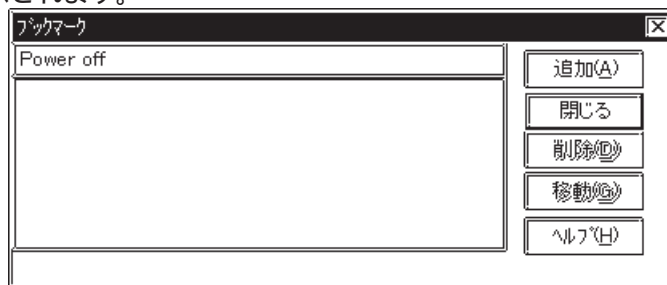
4. [リファレンス]ダイアログボックスの最初の行をクリックします。ロジックプログラムは、ラング上でユーザーが指定した命令および変数を反転表示しています。

3.8.4. ブックマークの使用

ロジックプログラムの中の特定のポイントを繰り返し参照し、そのたびにそこへスクロールするのが面倒である場合、ブックマークを使用します。ブックマークを設定するには、その設定位置を正確に指定しなければなりません。選択または反転表示できる項目はどれもブックマークとして指定できます。この例では、ラング3のNC命令をブックマークとして設定します。

ブックマークの設定手順

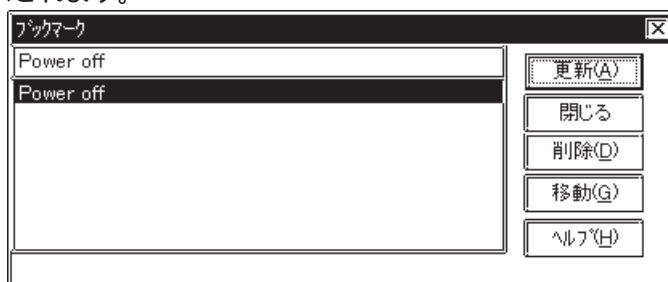
1. ラング3のNC命令をクリックします。
2. [検索]メニューから、[ブックマーク]を選択します。[ブックマーク]ダイアログボックスが表示されます。



3. [ブックマーク]フィールドにPower offと入力し、[追加]をクリックします。
これでブックマークが設定されました。Power offを選択し[移動]を入力してブックマークに戻ると、プログラムのどこからでも必ずラング3のNC命令に戻ります。新しいブックマークを設定するときは、ロジックプログラムの新しいポイントを選択し、手順1)~3)を繰り返します。Editorでは、複数のブックマークが利用できます。

ブックマークに飛ぶ手順

1. [検索]メニューから、[ブックマーク]を選択します。[ブックマーク]ダイアログボックスが表示されます。



2. [ブックマーク]ダイアログがリストから任意のブックマークを選択し、[移動]をクリックします。
ロジックプログラムが表示されているときは、自動的にブックマークの場所に戻ります。
参考: CTRL+Mを使って[ブックマーク]ダイアログボックスをオープンできます。

ブックマークの位置の変更手順

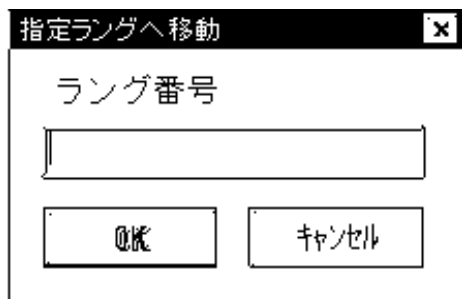
1. ロジックプログラムの中の新しい位置を選択します。
 2. 位置を選考したい任意のブックマークリストから選択します。
 3. [ブックマーク]ダイアログボックスの[更新]をクリックします。
- 以上の手順で、ブックマークの変更ができます。

3.8.5. [指定ラングへ移動]コマンドの使用

[指定ラングへ移動]コマンドを使って、選択位置をロジックプログラムの指定したラングに移動できます。

[指定ラングへ移動]コマンドの使用手順

1. [検索]メニューから、[指定ラングへ移動]を選択します。[指定ラングへ移動]ダイアログボックスが表示されます。



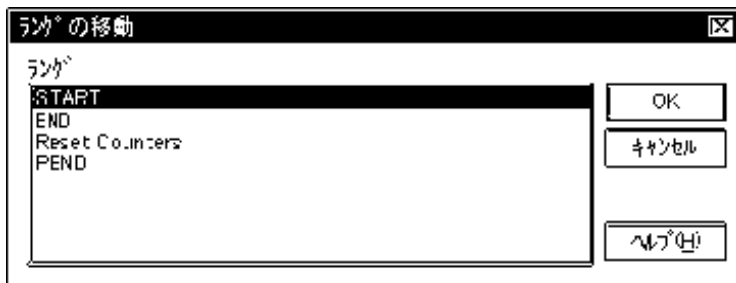
2. ラング番号を入力します。
3. [OK]をクリックすると、指定したラングに移動します。

3.8.6. [ラベルへ移動]コマンドの使用

[ラベルへ移動]コマンドを使って、ロジックプログラムの特定のラベルにジャンプできます。

[ラベルへ移動]コマンドの使用手順

1. [検索]メニューから、[ラベルへ移動]を選択します。[ラベルへ移動]ウィンドウが表示されます。



2. 移動したいラベルを選択します。
3. [OK]をクリックすると、指定したラベルに移動します。

まとめ

このレッスンでは、[検索]、[参照]、[ブックマーク]、[指定ラングへ移動]、および[指定ラベルへ移動]の各コマンドの使用方法を学びました。

3.9. I/Oの割り付け

ロジックプログラムの作成が完了した後、選択した変数に実I/Oを割り付けることで、実際のI/O制御が可能です。このチュートリアルでは、使用方法を順序良く論理的に説明するため最初に変数を作成し、ロジックプログラムが完成した後に実I/Oを割り付ける流れを取りましたが、プログラミングを開始する前に実I/Oの配置が決まっていれば最初にI/O割り付けを行い、その後でプログラム上に実I/Oと共に変数を割り付けることができます。以下のレッスンでは両方の方法を説明します。

3.9.1. 実I/Oへの変数の割り付け


ロジックプログラムの変数を作成した後、いくつかの方法でそれをI/Oに割り付けることができます。

I/Oコンフィギュレーションのオープン手順

[データ]メニューから、[I/Oコンフィギュレーション]を選択します。[I/Oコンフィギュレーション]ウィンドウが表示されます。

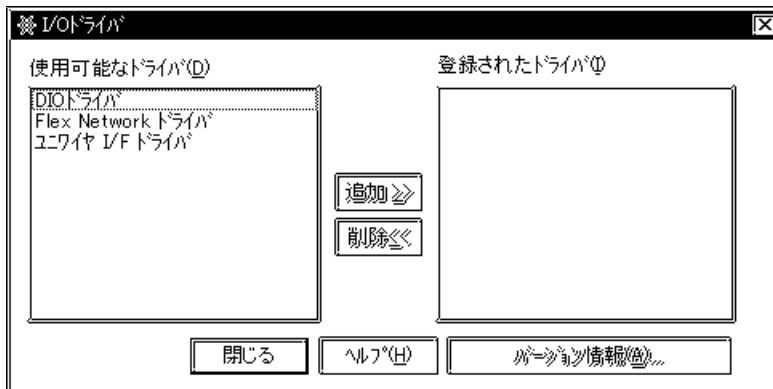


この時点では、プログラムに登録されているドライバはありません。

参考: ツールバーの  をクリックするか、[変数タイプ]ダイアログボックスの[I/O(F)...]をクリックしても、[I/Oコンフィギュレーション]ウィンドウをオープンできます。

ドライバの指定手順

1. [I/Oコンフィギュレーション]ウィンドウの[ドライバ]をクリックします。[I/Oドライバ]ウィンドウが表示されます。



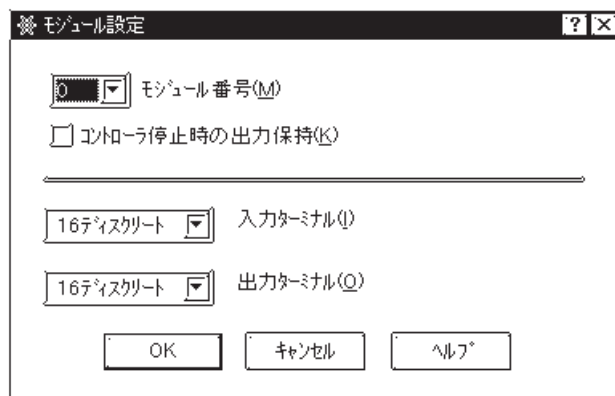
このウィンドウの左側の[使用可能なドライバ]には、すべての使用可能なドライバがリストされます。同じく右側の[登録されたドライバ]には、プログラムに登録されているドライバがリストされます。現在何も登録されていません。

2. [使用可能なドライバ]から DIOドライバを選択します。
3. 任意のI/Oドライバを選択し[追加(>>)]をクリックするか、ドライバのタイトルをダブルクリックします。
[登録されたドライバ]に選択したドライバが表示されます。
4. [閉じる]をクリックします。次頁のような[I/Oコンフィギュレーション]ウィンドウが表示されます。

この例では、DIOユニット、ユニット0を設定します。DIOユニットはそれぞれ16点ずつの入出力があります。

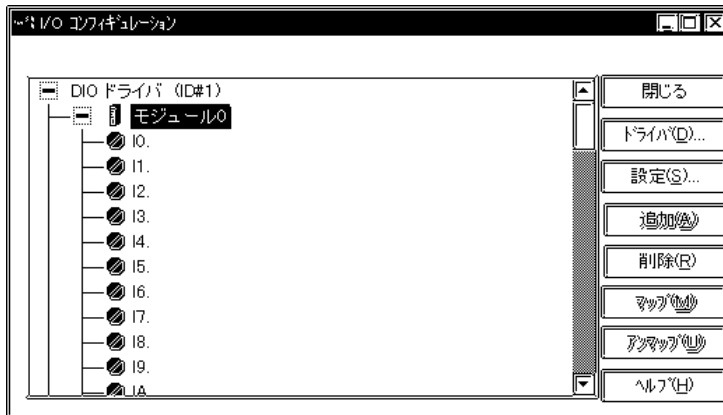
DIOドライバのセットアップ手順

1. モジュール0を選択します。
2. [設定]をクリックします。設定ウィンドウが表示されます。



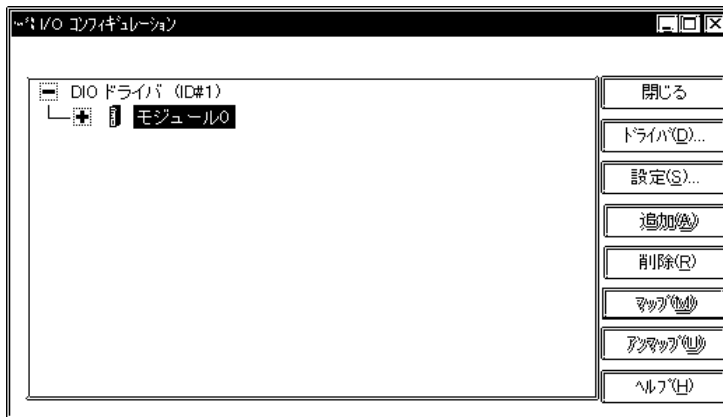
3. デフォルトで入力 / 出力は16 ディスクリート(ビット型)に設定されています。

4. [OK]をクリックします。下のように、[I/Oコンフィギュレーション]ウィンドウが表示されます。




モジュール0の下には、DIOユニットに関連付けられている16個の入力端子と16個の出力端子が割り付けられています。このレッスンの後半で、それらに変数を割り付けます。

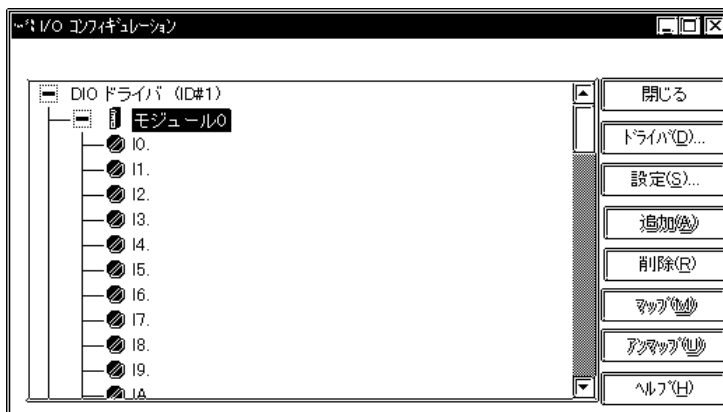
5. モジュール0の隣の  をクリックします。端子が非表示にされ、 の代わりに  が表示されます。




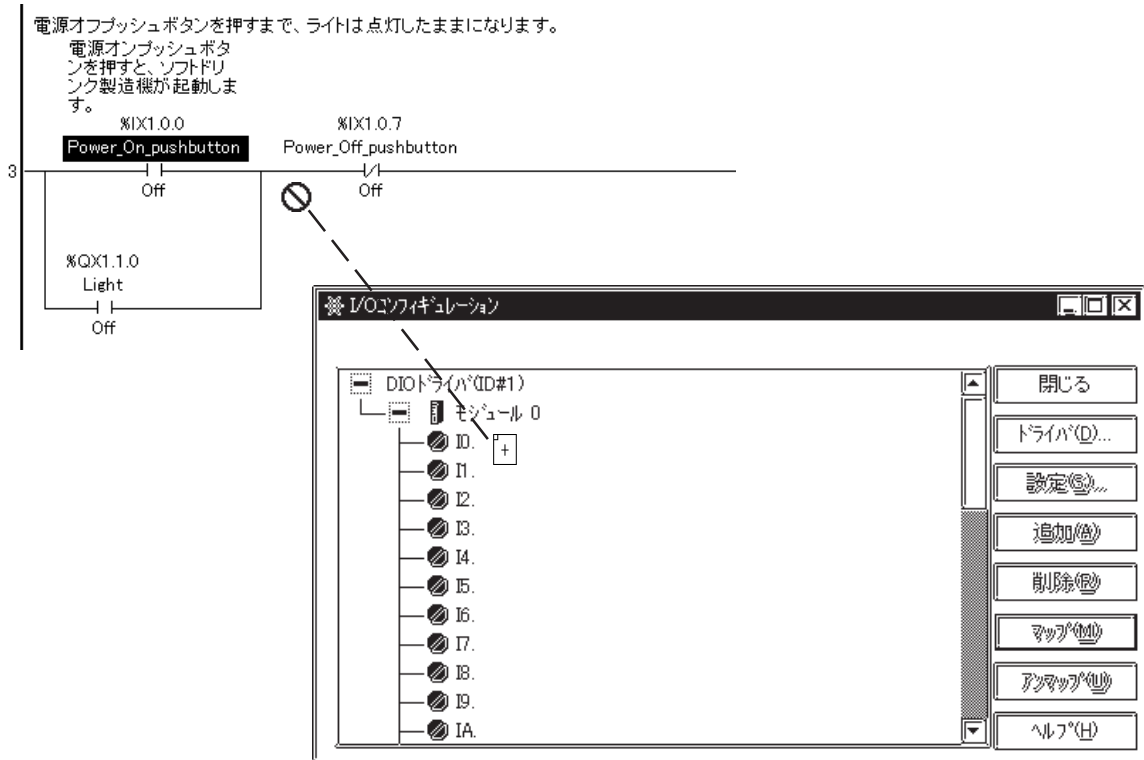
6. DIOユニットは2ユニットまで着用できます。2ユニット目の設定も、1ユニット目と同様です。

変数をクリックし、I/O端子にドラッグする手順

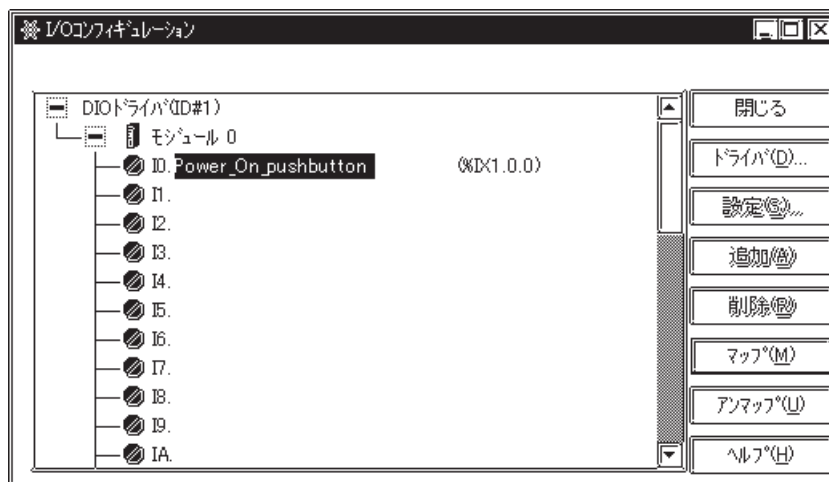
1. ユニット0の隣の  をクリックします。下のように、[I/Oコンフィギュレーション]ウィンドウが表示されます。このモジュールでは、最初の16個の端子をディスクリート(ビット型)入力のために使用します。



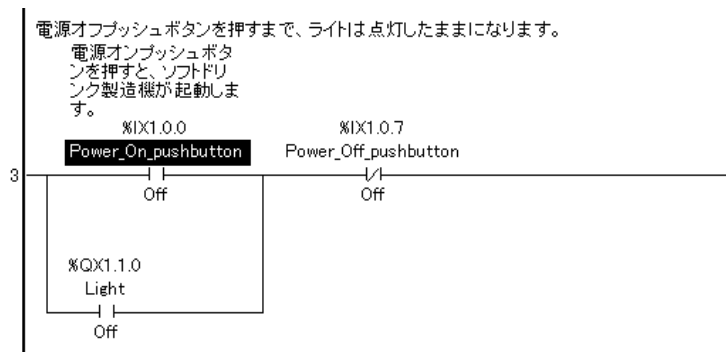
2. ラング 3 の a 接点 (NO 命令) に設定された変数 Power_On_pushbutton を選択します。
3. Power_On_pushbutton をクリックして、端子 I0. にドラッグします。分岐を挿入する場合と同様に、カーソルは最初に  になります。カーソルがこの状態のときは、I/O 端子に変数を割り付けることはできません。



4. カーソルを端子 I0. の上にドラッグし、マウスを離します。変数 Power_On_pushbutton が端子 I0. に割り付けられます。



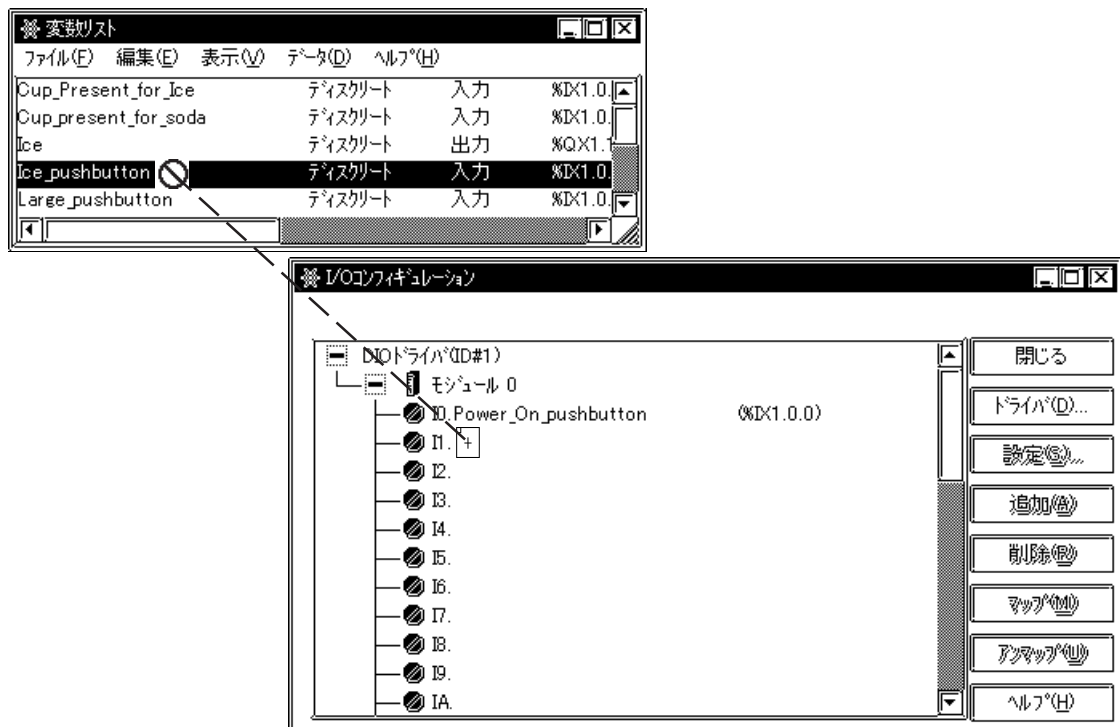
ラング3のN0命令の上の変数 Power_On_pushbutton の上に、一連の数字および文字が表示されています。これは、変数の実 I/O アドレスです。I/O アドレスは IEC 規格に合致したフォーマットで表示されています。



[変数リスト]ウィンドウから変数を I/O 端子に割り付けする手順

1. [I/Oコンフィギュレーション]ウィンドウは、オープンしたまま、[変数リスト]ウィンドウを開きます。
2. 両方のウィンドウが見えるように、2つのウィンドウを配置します。
3. [変数リスト]ウィンドウから、変数 Ice_pushbutton をクリックし、[I/Oコンフィギュレーション]ウィンドウの端末 I1 にドラッグします。
4. マウスを離します。変数 Ice_pushbutton が入力端子 1 に割り付けられました。

参考：上記の手順を使って、[コメントリスト]ウィンドウからの変数を I/O に割り付けることもできます。



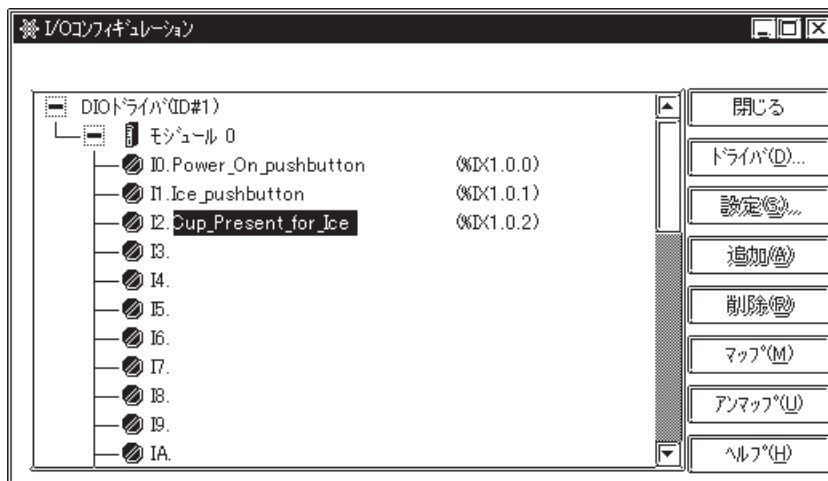
[注意] [変数リスト][コメントリスト]ウィンドウから[I/Oコンフィギュレーション]ウィンドウに変数をドラッグしてI/O割り付けを行った場合、実I/Oの属性が優先され、変数属性が入力/出力などに変更されます。

テキスト入力によるマッピングの手順

1. 端子 I2 をクリックします。



2. [Enter]を押します。端子テキストフィールドが起動します。
3. Cup_Present_for_Ice と入力します。
4. [Enter]を押します。Cup_Present_for_Ice が入力端子 I2 に割り付けられます。

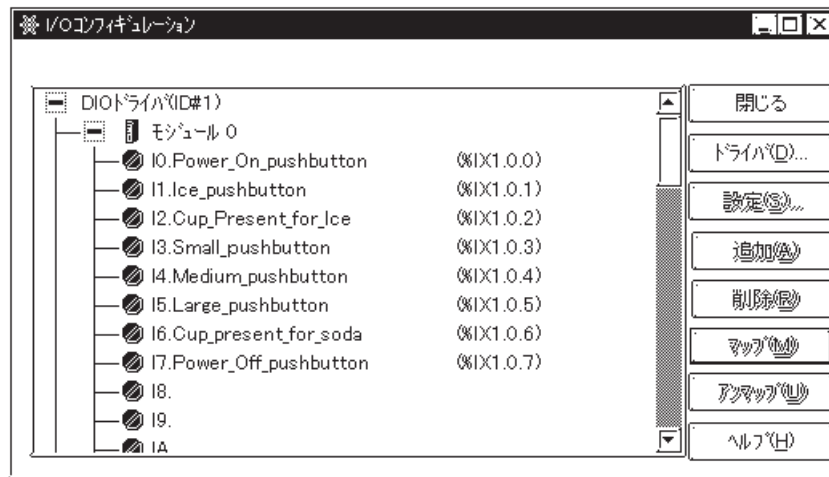


参照：テキスト入力によって I/O 割り付けを行うと自動的に変数リストにも登録されます。

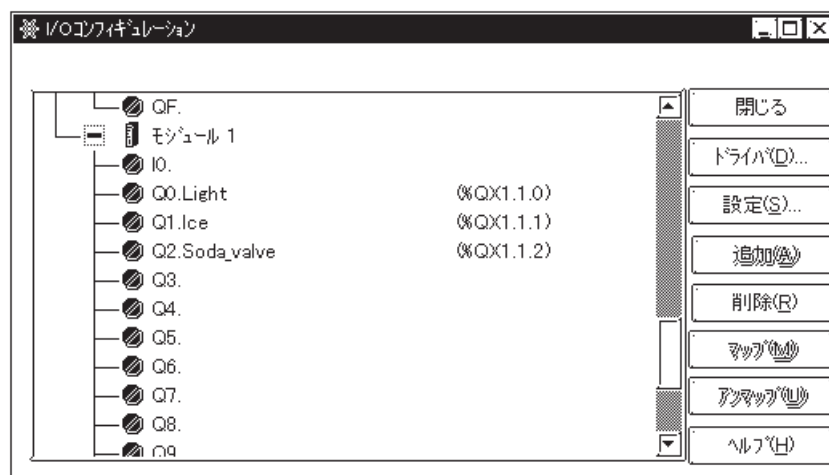
出力端子に変数を割り付ける方法は、入力端子への変数の割り付けと同じです。今までの手順を使って、以下の表の変数を入力および出力端子に割り付けてください。

変数名	端子タイプ	端子番号
Light	出力	Q0
Ice	出力	Q1
Soda valve	出力	Q2
Small_pushbutton	入力	I3
Medium_pushbutton	入力	I4
Large_pushbutton	入力	I5
Cup_Present_for_Soda	入力	I6
Power_Off_pushbutton	入力	I7

下のように入力および出力ユニットが、[I/Oコンフィギュレーション]ウィンドウに表示されます。



出力が割り付けられた I/O 設定



3.9.2. [I/O コンフィギュレーション]ウィンドウのアンマップ(割り付け解除)

[I/O コンフィギュレーション]ウィンドウのアンマップ手順

1. 入力ユニット0を選択します。
2. [I/O コンフィギュレーション]ウィンドウの端子 I0 をクリックします。
3. [アンマップ] をクリックします。Power_On_pushbutton の端子 I0. への割り付けが解除されます。その後、この変数を任意の端子にも割り付けることができます。この練習では、この変数を端子 I0 に再び割り付けます。

3.9.3. I/O にマップした変数を命令で使用方法

新しいプログラムのためにI/Oを設定する最も簡単な方法は、変数を直接に端子上に入力する方法です。それらの変数は自動的に作成され、設定され、正しいI/Oポイントにマップされます。先にI/Oを設定してから、ロジックプログラムを作成する場合、下記の方法で、I/Oポイントを作成します。

I/O にマップした変数を命令で使用する手順

1. 変数をクリックし、I/O端子にドラッグする手順で示した方法を使って、変数をドライバの入力および出力端子に割り付けます。
2. ロジックプログラムを作成します。
3. [I/O コンフィギュレーション]ウィンドウから変数をクリックし、I/Oを使用する命令にドラッグします。

まとめ

ここでは以下の方法を学びました。

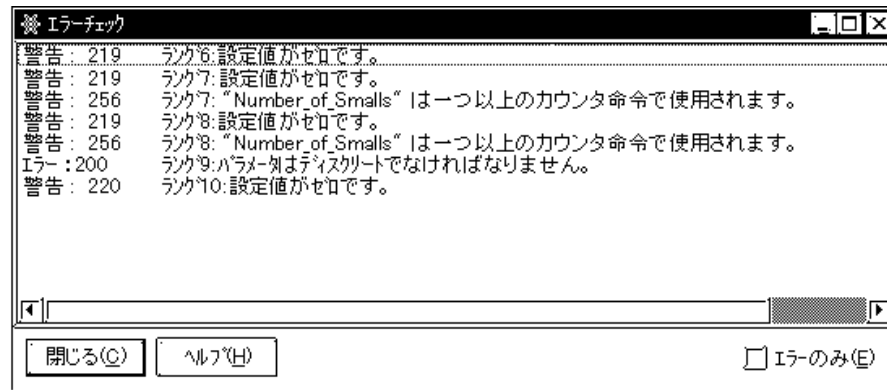
- ・ I/O ドライバの選択
- ・ DIO ドライバの設定
- ・ 変数の I/O への割り付け

3.10. プログラムエラーチェック

ロジックプログラムをGLCに転送して実行する前に、プログラムにエラーがないことを確認しておかなければなりません。そのためにエラーチェックを行います。

エラーチェック実行手順

[ファイル]メニューから、[エラーチェック]を選択します。[エラーチェック]ウィンドウが表示されます。



[エラーチェック]ウィンドウは、Editorがプログラムで検出できるすべてのエラーおよび考えられる問題箇所をリストします。問題箇所は、警告としてリストされます。ウィンドウの右下隅には、[エラーのみ]というマークが付いているチェックボックスがあります。このボックスをチェックすると、Editorがプログラムの中で検出したエラーだけが表示され、警告は表示されません。コントローラは警告を含むプログラムをGLCに書き込むことはできますが、エラーを含むプログラムはGLCに書き込むことはできません。これらのエラーを修正しなければなりません。

参考: エラーチェックは、ツールバーの をクリックすることによって実行することもできます。

[エラーチェック]ウィンドウは、エラーおよび警告を、ロジックプログラムで検出された順序に表示します。つまり、ランク1のエラーが最初に表示され、次にランク2以下のエラーが順に表示されます。[エラーチェック]ウィンドウのエラーまたは警告をダブルクリックすると、問題の行を直接表示します。

- ・ロジックの問題である場合、プログラムのその部分が表示されます。
- ・I/O割り付けに関する問題である場合、[I/Oコンフィギュレーション]ウィンドウが表示されます。

前に説明したように、[エラーチェック]ウィンドウには種々のエラータイプが表示されます。エラーチェックでは、下のようなエラーが示されています。

エラー 200 ランク 9 パラメータはディスプレイでなければなりません

エラーの修正手順

1. [エラーチェック]ウィンドウのエラー行をダブルクリックします。ラング9の命令のパラメータボックスが反転表示され、それに割り付けられる変数がないことを知らせます。
2. 変数として Soda_valve と入力します。
それぞれのエラーおよび警告に関する詳細[参照 Pro-Control オンラインヘルプまたは[付録A：エラーと警告]]
[エラーチェック]ウィンドウにリストされているエラーの修正が終わったら、もう一度エラーチェックを実行します。まだ残っているエラーが表示されます。エラーをすべて修正した後、プログラムをコントローラに書き込むことができます。

まとめ

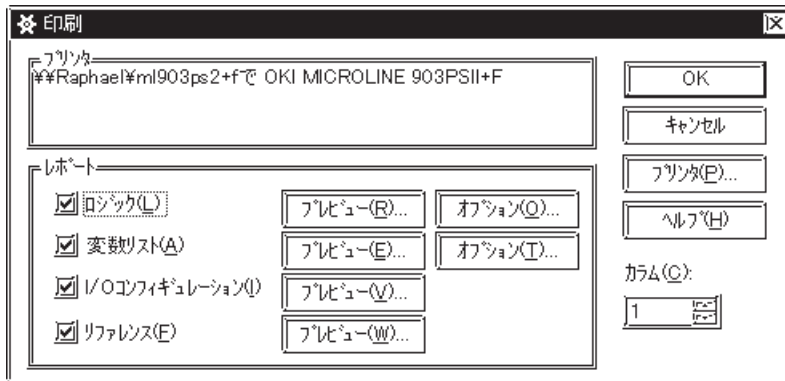
このレッスンでは、Editor ロジックプログラムのエラーチェックの方法を学びました。これでプログラムを GLC に転送して実行する準備が完了しました。このあとの手順は、「4.1 コントローラの設定」で詳しく説明します。

3.11. ロジックプログラムの印刷

Pro-Control Editor では、ロジックプログラムを印刷することができます。

ロジックプログラムの印刷手順

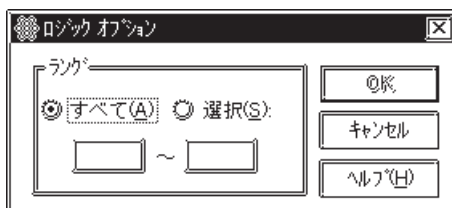
1. [ファイル]メニューから、[印刷]を選択します。[印刷オプション]ウィンドウが表示されます。プレビュー機能で、印刷する前に、差し込み文書の内容を確認することができます。



レポートを印刷する際のカラムの数(1~4)を選択できます。[レポート]セクションの下に、[ロジック]、[変数リスト]、[I/Oコンフィギュレーション]、および[リファレンス]というラベルが付いた4つのチェックボックスがあります。これらのチェックボックスは、ロジックプログラムを印刷するときに、以下のオプションを提供します。

ロジック：

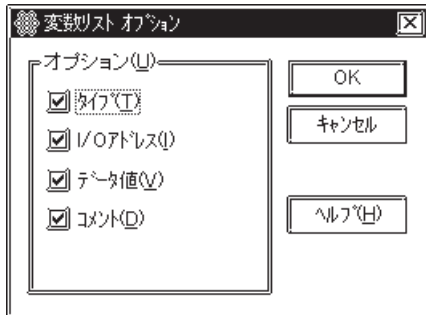
この項目では、ロジックプログラムのラングを印刷できます。その隣の[オプション]をクリックすると、下のようなウィンドウが表示されます。



プログラムのすべてのラングを印刷するには[すべて]を選択し、プログラムの特定の範囲のラングを印刷するには[選択]を選択し、その範囲を入力します。[表示]メニューの拡大、縮小で印刷するロジックの大きさを調整できます。

変数リスト :

この項目では、変数リストのオプションが設定できます。目的のチェックボックスをクリックして、変数リストに含める項目を選択します。




変数リストオプション :

このダイアログでは、変数の設定を印刷できます。

オプション	説明
タイプ	変数タイプを表示します。
I/Oアドレス	割り付けられた変数すべてのI/Oアドレスを表示します。
データ値	すべての変数のデータ値を表示します。
属性	保持、グローバルの設定を表示します。
コメント	変数に付加されたコメントを表示します。

参照 : このダイアログでは、全変数の全項目を表示するリファレンスを印刷できます。

参考 : ツールバーのをクリックすることによっても、プログラムを印刷できます。

まとめ

このレッスンでは、ロジックプログラムを印刷する方法について学びました。

第4章 ロジックプログラムを実行する

エラーのないロジックプログラムが完成したら、GLCにダウンロードしてコントローラ(ランタイム)上で実行することができます。

本章では、コントローラの設定方法、プログラムの書き込み方、モニタリングモードでの実行方法について説明します。

4.1. コントローラの設定

ロジックプログラムをGLCに書き込む前に、コントローラが正しく設定されていることを確認してください。GLCでプログラムを実行する場合、コントローラの設定にはシステムの設定、通信の設定、メモリ、GLCプラットフォームという4つの項目があります。

コントローラの設定手順

システムの設定

[コントローラ]メニューで、[システムの設定]を選択します。



[システム設定]に設定した値は、コントローラでロジックプログラムを実行する際の設定値になります。設定後にプログラムを実行すると、手動で変更しない限り、コントローラでは常にこの設定が適用されます。設定内容は個別の実行プログラム毎に設定されます。

[システム設定]タブの各項目の内容は次ページの表の通りです。

項目	内容
コンスタントスキャン	システム変数#TargetScanに、スキャンタイムをms単位で指定します。 注：コンスタントスキャンモードでも、ロジックタイムがスキャンタイムの50%を超える場合はコンスタントなスキャンを保証できません。
パーセントスキャン	システム変数#PercentAllocに、適切なスキャンタイムを、総処理時間に占める比率で指定します。
WDT設定値	システム変数#WatchdogTimeを設定します。
コントローラの自動スタートを禁止する	電断後などにコントローラが起動するとき、コントローラがロジックプログラムを自動的に実行しないことをシステム変数#DisableAutoStartで指定します。GLCのオフラインメニューで”電源ON時の動作モード”を[DEFAULT]に設定したときに有効になります。
継続異常スイッチ	コントローラにマイナー異常(マイナー異常は、無視しても安全なエラーです。マイナー異常の原因については、#FaultCodeをご覧ください。)が発生したときに、ロジックプログラムの実行を終了するかどうかをシステム変数#FaultOnMinorで指定します。
I/O使用可 (GLC2400のみ)	この項目を選択すると、物理I/Oへのデータ書き込み/読み込みが行われます。

システム変数の詳細

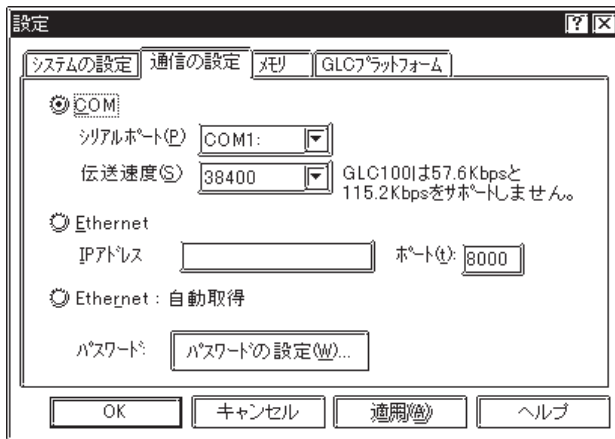
[参照 [Pro-Control オンラインヘルプのプログラマーズリファレンス]、
[4.4システム変数]、または[Pro-Control ユーザーズマニュアル]]

通信の設定

・パスワード設定方法

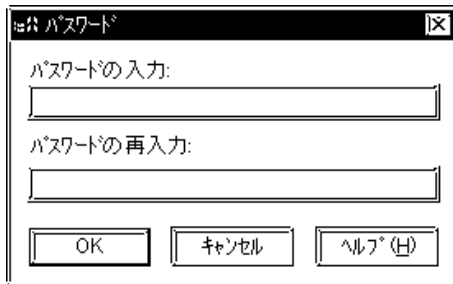
パスワードを設定した後、GLCにダウンロードすると次からファイルのダウンロード/アップロード、モニタリングモードへの移行時にパスワードの入力確認が行われます。登録したパスワードを入力しなかった場合、実行に移りません。また、パスワードがダウンロードされているGLCに対して、Ver.1.0、Ver.1.5のEditorからのダウンロード/アップロード、モニタリングモードへの移行はできません。

1. [コントローラ]をクリックし、[設定]を選択します。
2. [通信の設定]の[パスワードの設定]を選択します。



パスワード・・・半角英語字 24文字以内で設定。未入力で解除。

3. パスワードダイアログではパスワードを2度入力します。2回とも同じパスワードを入力する事で有効になります。パスワードを削除したい場合は何にも入力しないで[OK]を押すと削除されます。

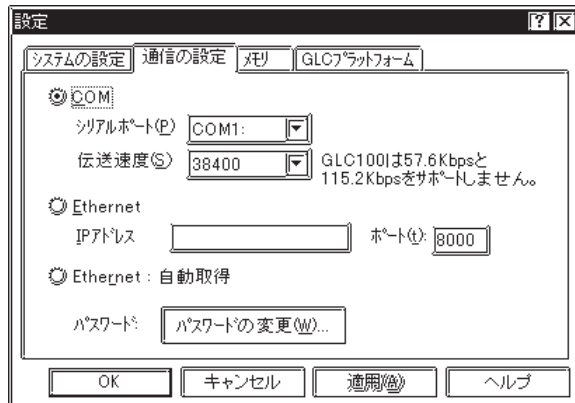


4. パスワードが設定されているときにパスワードメニューを選択するとパスワードを入力するダイアログが表示されます。



・パスワード変更方法

1. [コントローラ]をクリックし、[設定]を選択します。
2. [通信の設定]の[パスワードの変更(P)]を選択します。



パスワード・・・半角英語字 24 文字以内で設定。未入力で解除。

3. パスワードを変更したい場合は、現在設定しているパスワードを入力します。



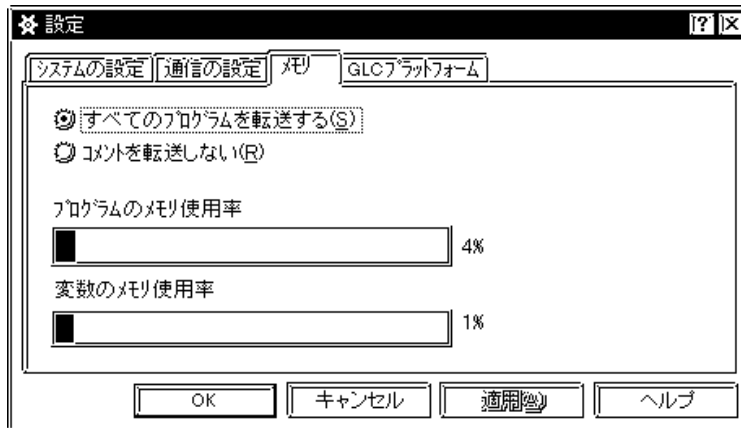
4. パスワードは解除されます。また、新しくパスワードを設定する場合は、「パスワード設定方法」の手順でパスワードの設定を再度、行ってください。

【禁止】

- ・パスワードを保存した「*.WLL」ファイルを Ver.1.0、Ver.1.5 で開いて保存をした場合、パスワードは削除されます。パスワード機能は、Ver.2.0 からの機能です。
- ・パスワードを設定していてパスワードを忘れた場合、二度とダウンロード/アップロード、モニタリングモードできません。
- ・パスワードを設定していてもGLCシステムのセットアップを行うとパスワード設定は解除されます。
- ・パスワードを設定していても、GLCデータをメモリローダによりコピーは可能ですが、メモリローダで読み込んだデータの編集は行えません。

メモリ

[メモリ]のタブをクリックします。

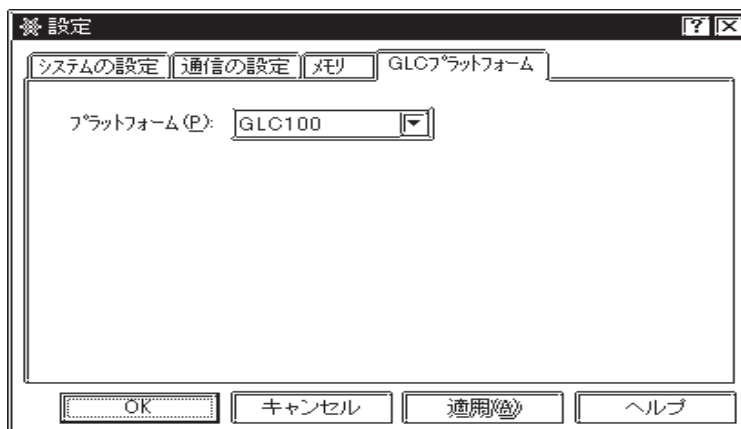


- ・ すべてのプログラムを転送する
コメントを含むすべてのロジックプログラムを転送します。
GLCより読み込むとロジックプログラムのコメントを読み込めます。
- ・ コメントを転送しない
GLCに書き込むファイルのサイズを小さくすることができます。コメントを転送しなかった場合は、GLCより読み込んでコメントはありません。
- ・ プログラムのメモリ使用率
現在編集中的ロジックプログラムの容量がGLCで使用可能なメモリのうち、何パーセントを占めているかを示します。
- ・ 変数のメモリ使用率
現在登録されている変数の容量がGLCで使用可能なメモリのうち、何パーセントを占めているかを示します。

GLC プラットフォーム

[GLC プラットフォーム]のタブをクリックします。

使用するGLC本体のタイプを選択してください。



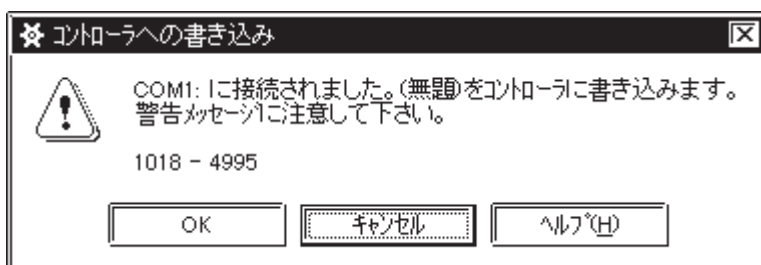
4.1.1.1. コントローラへの書き込み

Editorを使ってロジックプログラムを作成しエラーがない状態になったら、コントローラへ書き込みができ GLC での実行が可能になります。

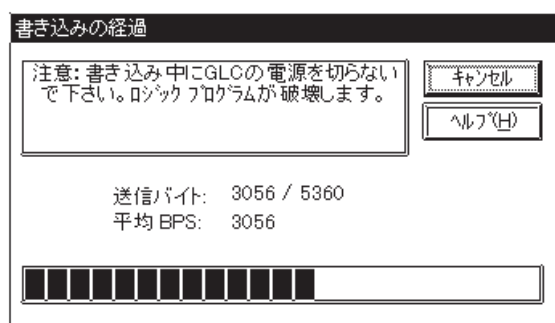
参考：GLCへロジックプログラムを書き込む前に、GLCをセットアップしてください。

コントローラへの書き込み手順

1. [コントローラ]メニューで[コントローラへの書き込み]を選択します。[コントローラへの書き込み]ウィンドウが表示されます。コントローラへ書き込むようプロンプトで要求されます。コントローラにプログラムが書き込まれる前に、Editorは自動的にエラーのチェックを実行します。プログラムにエラーがあると、コントローラへの書き込みはできません。



2. [OK]をクリックします。[ダウンロードの進行状況]ウィンドウが表示されます。GLCへのファイルの転送を開始し、進行状況が表示されます。



参考：.WLL ファイルをコントローラに書き込むと、DIO ドライバなどの必要な I/O ドライバがダウンロードされます。ドライバに変更がない場合は、ドライバのダウンロードはスキップします。

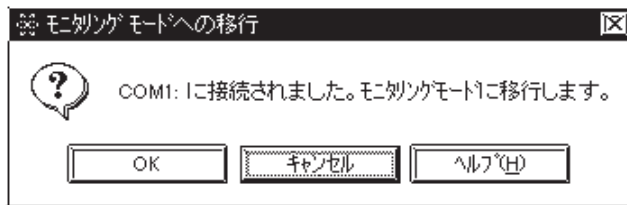
参考：転送する前にコメントを削除すれば、ダウンロードしたファイルのサイズを縮小することができます。[参照 [コントローラの設定]]

【注意】GLC300/GLC2400シリーズでは、ロジックプログラムの書き込みのあとリセットされません。

4.1.2. モニタリングモードへの移行

モニタリングモードへの移行手順

1. [コントローラ]メニューで、[モニタリングモードへの移行]を選択します。ウィンドウがモニタリングモードで移行するかどうか聞いてきます。



2. [OK]をクリックします。モニタリングモードで接続されたGLC上のプログラムのRUN/STOPができます。

4.1.3. イーサネット機能（対応機種：GLC2400）

イーサネット経由でロジックプログラムの書き込み / 読み出し、およびモニタリングモードの実行ができます。

転送設定

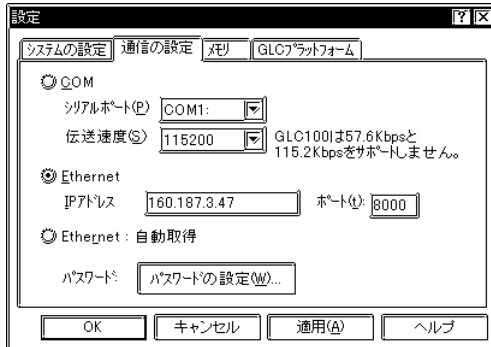
[コントローラ]メニューで、[設定]を選択します。[設定]メニューの[通信設定]タブで「イーサネット」または「イーサネット：自動取得」を選択します。

項目	内容
イーサネット	イーサネット経由で通信するGLCのIPアドレス、ポート番号を設定します。
イーサネット：自動取得	イーサネット上のGLCを検索します。検索結果が一覧表示されます。検索結果より通信したいGLCのIPアドレスを選択します。[コントローラへの書き込み]では、複数のGLCを選択できます。

1. [イーサネット]

通信したいGLCのIPアドレス、ポート番号を入力します。

[コントローラへの書き込み]、[コントローラからの読み出し]または[モニタリングモードへの移行]を実行するとイーサネット経由で通信をはじめます。



2. [イーサネット：自動取得]

[コントローラへの書き込み]、[コントローラからの読み出し]または[モニタリングモードへの移行]を実行するとイーサネット上に接続されているGLCを一覧表示します。通信したいGLCを選択して[OK]をクリックすると通信をはじめます。



IPアドレスを指定して[追加]を選択すると指定した IP アドレスの GLC と接続します。

The screenshot shows a dialog box titled "IPアドレス" (IP Address). It has a "ポート" (Port) field set to "8000". Below this is a list of IP addresses: "160.187.3.45" and "160.187.3.46". A search field labeled "検索" (Search) is present. Below the search field, the IP address "160.187.3.47" is entered in a text box, which is circled in red. To the right of this text box is an "追加" (Add) button. At the bottom of the dialog are three buttons: "OK", "キャンセル" (Cancel), and "ヘルプ(H)" (Help).

参照 イーサネット設定の詳細については、GLC2400 シリーズユーザーズマニュアル

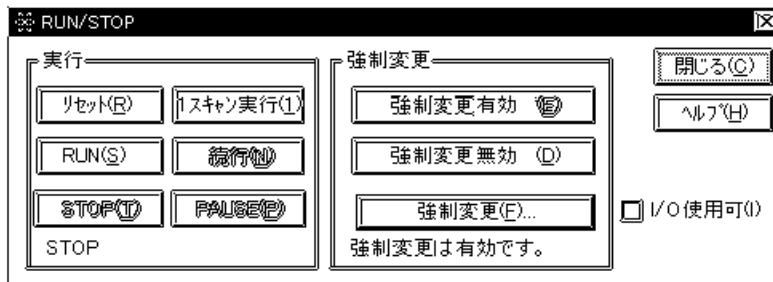
4.2. コントローラのRUN/STOP

モニタリングモードで接続すると、コントローラのRUN/STOPができます。この時点でコントローラのモニタを実行し始めます。

前述のように、コントローラへモニタリングモードで接続してから、RUN/STOPやモニタリングモードの機能を実行してください。

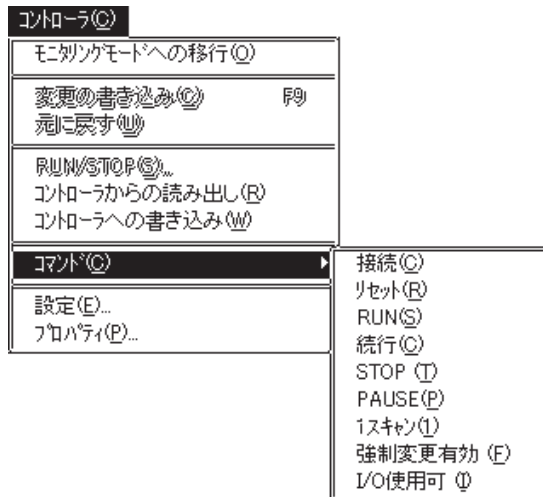
コントローラのRUN / STOP手順

1. [コントローラ]メニューで、[RUN/STOP]を選択します。プログラミングモードでは、この項目は使用できません。[RUN/STOP]ウィンドウが表示されます。



項目	内容
RUN	RUNボタンをクリックするとコントローラによるプログラム実行が開始します。最初にプログラムのスキャンが始まり、すべてのラダープログラムが順に実行されます。最初のスキャンで初期設定プログラムが実行されます。
STOP	STOPボタンをクリックするとコントローラが終了します。
リセット	リセットボタンをクリックすると、コントローラが実行ファイルを再ロードし、I/O及び変数を初期化します。
1スキャン実行	このボタンをクリックすると、プログラムのスキャンが1回行われます。この機能はアプリケーションのトラブルシューティングやデバックに有効です。
PAUSE	PAUSEボタンをクリックすると、コントローラはプログラムをスキャンするのを一時停止しますが、I/Oリフレッシュは実行されます。
続行	続行はPAUSEボタンをクリックするとコントローラは、現在のデータ値を使用してプログラムの実行を再開します。
強制変更有効	強制変更を有効にします。
強制変更無効	強制変更を無効にします。
強制変更	ラダープログラムで強制変更に登録したすべての変数をリストアップします。
I/O使用可	この項目を選択すると、物理I/Oへのデータの書き込み / 読み込みが行われます。

または、[コントラ(C)]メニュー及びツールバーより選択する事ができます。



	接続
	モニタリングモードへの移行
	コントローラへの書き込み
	コントローラからの読み出し
	変更の書き込み
	リセット
	RUN
	続行
	STOP
	PAUSE
	1スキャン
	強制変更有効
	I/O使用可

参考：リセットにより、保持型の変数はリセットされます。特別な初期設定が必要な場合は、ロジックプログラム上でMOV命令などを使って初期値を設定してください。

4.3. システム変数によるプログラムのトラブルシューティング

コントローラの起動後にアプリケーションがうまく動作しない場合、システム変数を使用してアプリケーションのトラブルシューティングを実行することができます。システム変数 #Fault、#IOFault、#IOStatus、#ScanCount は、コントローラか I/O に発生するトラブルの検出に最も有効です。

項目	
#FaultCode	最新のエラー状況を示します。立ち上がり一回目のスキャン実行時に、「0」にリセットされます。
#Fault rung	エラーが発生したラングの番号が表示されます。
#IOFault	I/Oドライバでエラーが検出されるとONになります。
#IOStatus	I/O固有のエラーを表示する配列です。これらのエラーには、ドライバごとに異なるエラーコードがつけられます。エラーの詳細については、ドライバのオンラインヘルプを参照してください。#IOFault がONになった場合にのみ、エラーが#IOStatusで表示されます。
#ScanCount	コントローラが起動してから実行したスキャン回数を示します。この変数は、モニターを開始してから常に増加しています。増加していない場合は、コントローラが動作していません。

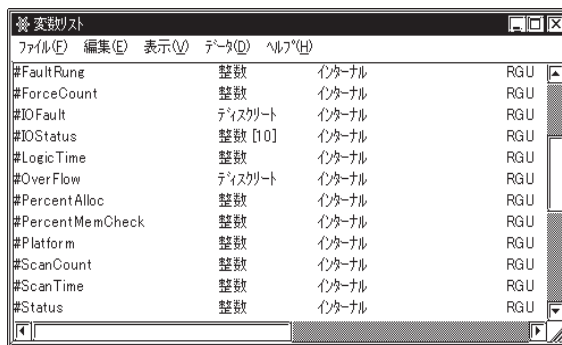
システム変数の詳細[参照 Pro-Control オンラインヘルプのプログラマーズリファレンス]

4.4. システム変数の表示

システム変数を表示させて、I/Oの状態、スキャン時間、コントローラの状態に関する情報を見ることができます。

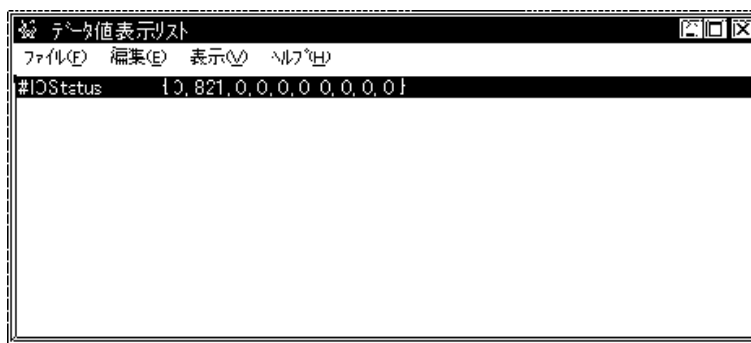
システム変数の表示手順

1. [データ]メニューで[変数リスト]を選択します。[変数リスト]ウィンドウが表示されます。このウィンドウにはPro-Control Editorのシステム変数(#で始まる変数)がすべて表示されます。表示されない場合は、[表示]メニューで[システム]を選択してください。



2. [データ]メニューで[データ値表示リスト]を選択すると、[データ値表示リスト]ウィンドウが表示されます。
3. モニタするシステム変数をクリックして、[変数リスト]ウィンドウから[データ値表示リスト]ウィンドウへドラッグします。

ロジックのスキャン中にエラーが発生した場合は、モニターした変数が当該のエラーを表示します。以下は、ドライバ1でI/Oエラー821が発生した例です。



4.5. コントローラからの読み込み

本体にあるプログラムの編集や保存を行いたい場合は、[コントローラから読み込み]をおこなわなくてはなりません。

コントローラからの読み込み手順

1. コントローラがモニタリングモードで接続されている場合は、[コントローラ]メニューで[プログラミングモードへの移行]を選択します。

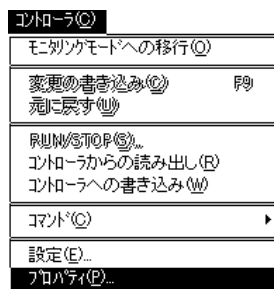
【重要】

プログラムに初期化していない値が含まれている場合、プログラムから読み込む前に、コントローラを終了してください。

2. [コントローラ]メニューで、[コントローラからの読み込み]を選択します。コントローラに書き込まれたプログラムがEditorで開かれます。これで、プログラムに変更を加えたり、*.WLLファイルというファイル名で保存することができますようになります。

4.6. プロパティ

コントローラメニューの[プロパティ]を選択するとGLCのプログラム情報がリストボックスで表示されます。



以下はプログラム情報のプロパティボックスです。



4.7.CF メモリローダツール (対応機種 : GLC2400)

4.7.1 CF メモリローダツールの作成 / 送信

CF メモリローダツールを実行するためには、まず、GP-PRO/PB でCF メモリローダツール (IPL.SYS、MLD2269.SYS、GPBACKUP.INF) を作成した後、CF カードへダウンロードします。GLCはそのCF カードからCF メモリローダツールを運転することができます。

参照 CF メモリローダ機能の詳細についてはGP-PRO/PB for Windows オペレーションマニュアル。

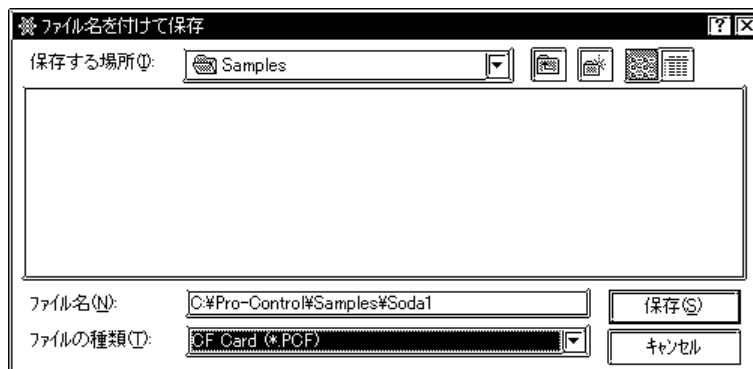
PCF ファイルの作成

GLCの起動CFカードデータまたはバックアップデータを作成するにはあらかじめPro-Control Editor で[PCF ファイル]を作成する必要があります。

1. [Pro-Control Editor の[ファイル]の[保存]でWLL ファイルを作成します。
2. [Pro-Control Editor の[ファイル]の「名前を付けて保存」でファイルの種類を* .PCFを選択します。ファイル名を入力して実行するとPCF ファイルが作成されます。

起動CF カードデータ / バックアップデータの作成 / 送信

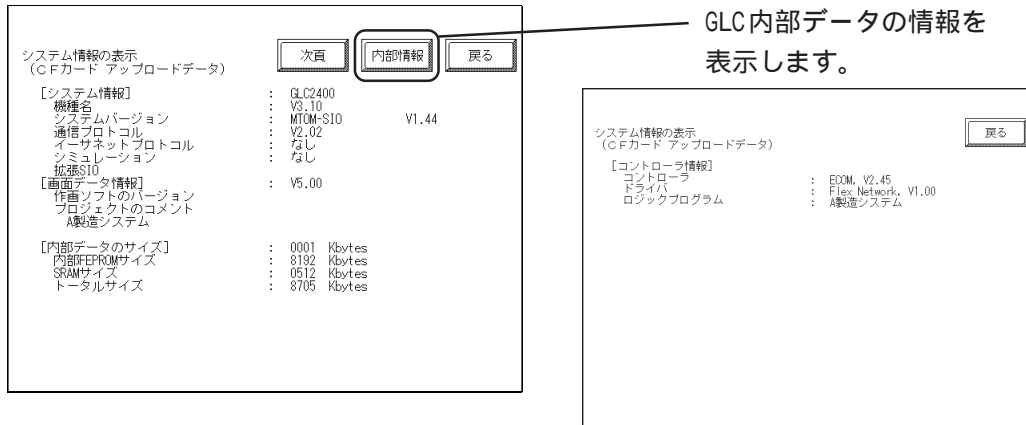
[起動CF カードデータ]、[バックアップデータ]の作成または送信を選択するとGLCのPCF ファイル選択のダイアログが表示されます。PCFファイルを選択して[OK]をクリックするとデータが作成されます。



[ロジックプログラムをバックアップする]を選択するとロジックプログラムもバックアップデータに書き込まれます。ロジックプログラムをバックアップしない場合、システムおよび画面のみバックアップされます。

4.7.2 システム情報の表示

「3. システム情報の表示」を選択すると以下の画面が表示されます。
 システム情報の表示には、CFカード内のアップロードデータ情報の表示とGLC内部のデータ情報の表示があります。GLC内部データの表示は[内部情報]をタッチすると表示されます。



以下にそれぞれのシステム情報の表示例を示します。

システム情報の表示 (CFカードアップロードデータ)

CFカード内のアップロードデータを確認することができます。

システム情報

- ・機種名 : GLC2400
- ・システムバージョン : V3.10
- ・通信プロトコル : MELSEC-ANA V1.40
- ・イーサネットプロトコル : V2.02
- ・シミュレーション : V3.20



・ システム情報のシミュレーションは、ラダーモニタプログラムが存在する場合は項目名がラダーモニタに変更されます。プログラムがCFカードのアップロードデータにない場合は、バージョンは表示されず、「なし」と表示されます。

画面データ情報

- ・作画ソフトのバージョン : V5.00
- ・プロジェクトのコメント : A製造システム (最大半角 60文字まで表示可能)



・ アップロード情報が存在しない場合は、画面データ情報は表示されず「アップロード情報がありません。」と表示されます。

アップロードデータサイズ

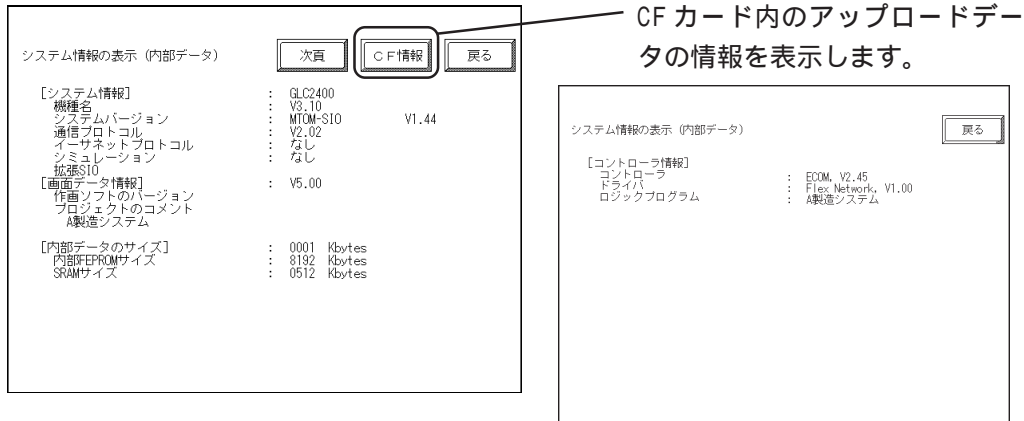
- ・情報データのサイズ : 1Kバイト
- ・内部 FEPROM サイズ : 8192Kバイト
- ・SRAM サイズ : 512Kバイト

コントローラ情報

- ・コントローラ : ECOM、V2.45
- ・ドライバ : Flex Network V1.00
- ・ロジックプログラム : A製造システム

システム情報の表示（内部データ）

[内部情報]をタッチすると以下の画面が表示されます。
GLCの内部データを確認することができます。



CF カード内のアップロードデータの情報を表示します。

システム情報

- ・機種名 : GLC2400
- ・システムバージョン : V3.00
- ・通信プロトコル : MELSEC-ANA V1.40
- ・イーサネットプロトコル : V2.00
- ・シミュレーション : V3.20



・システム情報のシミュレーションは、ラダーモニタプログラムが存在する場合は項目名がラダーモニタに変更されます。

画面データ情報

- ・作画ソフトのバージョン : V5.00
- ・プロジェクトのコメント : A製造システム（最大半角 60 文字まで表示可能）



・画面の転送時に「アップロード情報」を転送していない場合は、画面データ情報は表示されず、「アップロード情報がありません。」と表示されます。

アップロードデータサイズ

- ・内部 FEPRM サイズ : 8192K バイト
- ・SRAM サイズ : 512K バイト

コントローラ情報

- ・コントローラ : ECOM, V2.45
- ・ドライバ : Flex Network V1.00
- ・ロジックプログラム : A製造システム

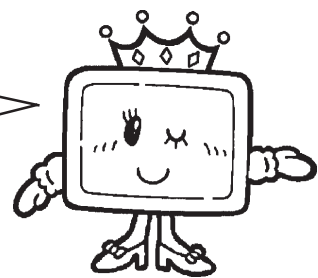
まとめ

本章では以下の方法について詳述しました。

- ・コントローラの設定方法
- ・コントローラへのプログラムの読み込み方法と書き込み方法
- ・コントローラの RUN/STOP の方法

MEMO

このページは、空白です。
ご自由にお使いください。



第5章 モニタリングモードでの動作確認

コントローラで実行しているプログラムにモニタリングモードで変更を加え、その変更内容をすぐに実施することができます。

本章での解説や例示では、C:\¥Pro-Control¥SamplesにあるSoda1.WLLを使用します。このファイルでは、ラダーのカラーやオプションをシステムのデフォルト値に設定しているものとします。

5.1. 編集を始める前に

本章では、例として Soda1.WLL を使用します。

ロジックプログラムの実行手順

1. Soda1.WLLを開きます。このファイルはサンプルプログラムとしてPro-Control Editorに添付されています。デフォルト設定のパスはC:\¥Pro-Control¥Samplesです。
2. コントローラにプログラムを書き込みます。
3. コントローラをモニタリングモード接続します。
4. コントローラをRUNします。コントローラの操作の詳細[参照] 「第4章ロジックプログラムを実行する」

モニタリングモードでGLCに加えられるプログラムの変更

コントローラにモニタリングモードで接続中、以下の変更をプログラムに加えることができます。

- ・変数の状態の変更
- ・変数の値の変更

5.2. モニタリングモード編集にカラーを使用する

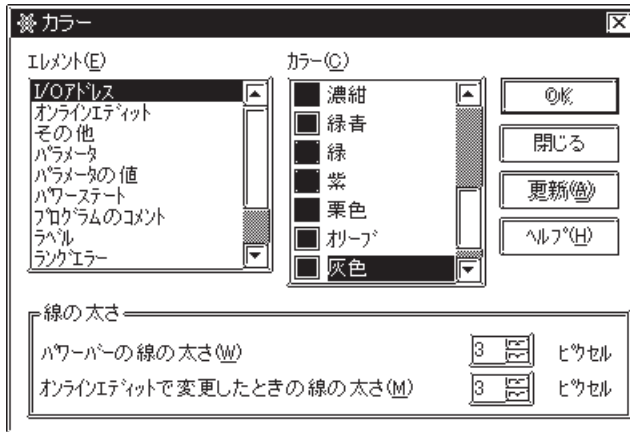
Pro-Control Editorでは、デフォルト設定のカラーを使用して、特定の状態を示したり、モ

項目	内容
緑	回路が導通しています。
赤	ラダーにエラーが発生していることを示します。
紫	オンラインエディットしていることを示します。

ニタリングモードの動作中にロジックプログラムに変更を加えることができます。
 デフォルト設定のカラーは以下のとおりです：

Pro-Control Editor のカラーのデフォルト設定の変更手順

1. [表示]メニューで[カラー]を選択します。[カラー]のウィンドウが表示されます。



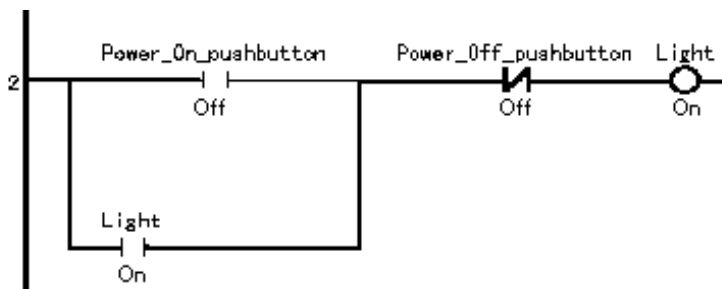
2. 変更したい[エレメント]と[カラー]を選択して、[更新]をクリックします。

5.3. ディスクリートを ON / OFF する

ロジックの実行中は、ディスクリート変数を手動で ON / OFF することができます。強制 ON にした場合は ON 状態が保持されますが、手動でディスクリートを ON にした場合スキャンが行われると、ディスクリートの状態がプログラムの影響を受けるため、プログラムに依存します。

ディスクリートの ON / OFF 方法

1. ラング2の出力コイルに割り付けた変数 Light をマウス右ボタンでクリックします。
2. ショートカットメニューで[ONにする]を選択します。Light 変数が ON になり、ラングが導通します。



3. ラング2の出力コイルに割り付けた変数 Light をマウス右ボタンでクリックします。
4. ショートカットメニューで[OFFにする]を選択します。Light 変数が OFF になり、導通状態はクリアされます。

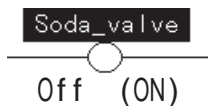
参考:[オプション]ウィンドウの[モニタリング]の[パワーフロー]チェックボックスがチェックされていないと、導通状態はロジックプログラムに表示されません。

5.4. ディスクリートを強制的に ON/OFF する

コントローラにモニタリングモードで接続している間は、ディスクリートを強制的に ON、OFF することができます。ディスクリートを強制的に ON/OFF すると、強制 ON/OFF を手動で解除するまでは状態が保持されます。整数、文字型変数のビット指定の状態は変更できません。

ディスクリートの強制的な ON / OFF 方法

1. ラング 9 の出力コイルの変数 Soda_valve を右クリックします。
2. ショートカットメニューで [強制 ON] を選択します。
3. [強制変更] ウィンドウで [OK] をクリックします。



変数が ON となり、プログラムでは OFF になりません。

参考：強制 ON/OFF を指定したのにロジックプログラムに反映されない場合は、使用禁止の設定になっています。使用できるようにするには、[RUN/STOP] ウィンドウで [強制変更有効] ボタンをクリックするか [コントローラ] メニュー及びツールバーより切り替えることができます。

5.5. 変数値の変更

コントローラにモニタリングモードで接続している間は、ロジックプログラムに含まれる変数の値を設定することができます。

変数値の変更手順

1. [データ] メニューで [データ値変更] を選択すると、[データ値変更] ウィンドウが表示されます。
2. ラダー図の変数 Number_of_SmallIs をクリックします。下のような [データ値変更] ウィンドウが表示されます。



3. [値] フィールドを選択してから、5 を入力してください。
4. [更新] をクリックします。Number_of_SmallIs の値が 5 になりました。他の値を変更するか、または [閉じる] をクリックして [データ値変更] ウィンドウを閉じます。

参考：・データ値は、10進数、16進数、8進数、2進数のいずれかのフォーマットで入力できます。

[データ表示リスト] で選択することもできます。

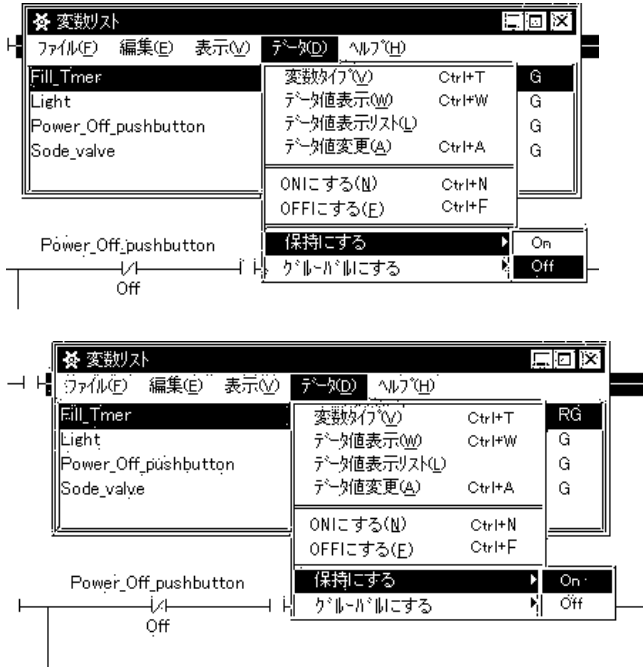
- ・[データ値変更] ウィンドウとともに、[変数リスト] か [データ値表示リスト] を使用すれば、変数をすばやく検索して設定することができます。

5.6. 変数の属性変更

変数リストの "データ" メニューに変数の保持、グローバルの属性を変更するメニューがあります。このメニューはプログラミングモードのみ有効となります。

変数(保持)の属性変更手順

1. [データ]メニューで[変数リスト]を選択すると、[変数リスト]ウィンドウが表示されます。変更したい変数を選択し、属性を変更します。システム変数の保持は変更できません。



変数(グローバル)の属性変更手順

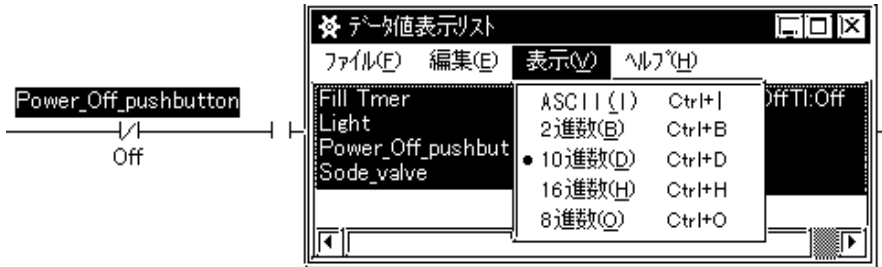
1. [データ]メニューで[変数リスト]を選択すると、[変数リスト]ウィンドウが表示されます。変更したい変数を選択し、属性を変更します。



5.7. データ値表示リスト

表示モード一括変更

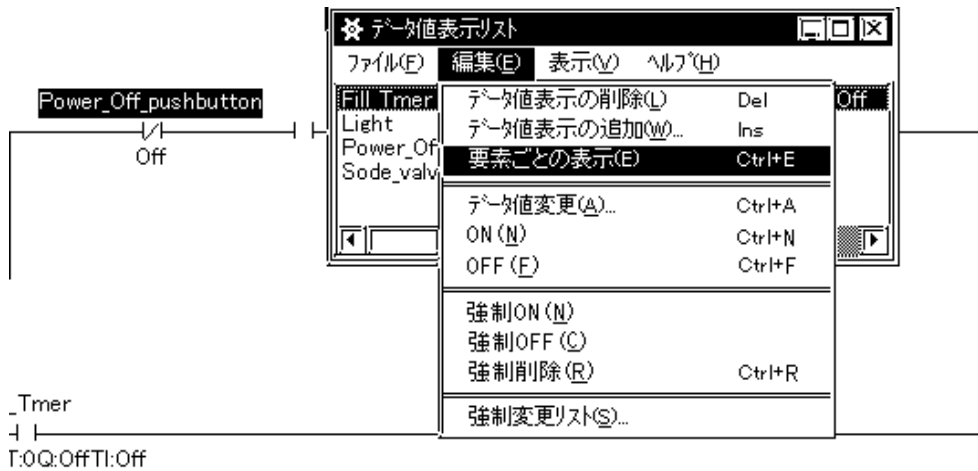
[データ]メニューから[データ値表示リスト]を選択します。選択した変数の表示モードを一括で変更が可能です。



配列の要素ごとの表示

データ値表示リストで配列の場合は、[要素ごとの表示]で配列タイマ、カウンタの値を要素ごとに表示します。

1. [データ]メニューで[変数リスト]を選択し、[データ]メニューから[データ値表示リスト]を選択します。
2. [データ値表示リスト]メニューから[要素ごとの表示]を選択します。



5.8. オンラインエディット（対応機種：GLC2400）

モニタリングモードにおいてロジックプログラムの実行中にロジックプログラムを変更することができます。

オンラインエディットでは、編集機能として6種類の編集ができます。

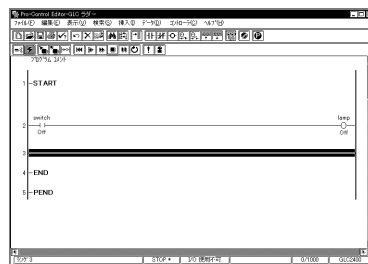
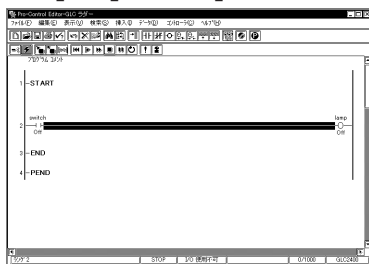
1. ラングの追加
2. ラングの置換
3. ラングの削除
4. ラベルの追加
5. サブルーチンの追加
6. 変数の追加

5.8.1 オンラインエディットの編集機能

ラングの追加

指定ラング間にラダー回路を1行追加します。

[挿入]の[ラング]を選択します。

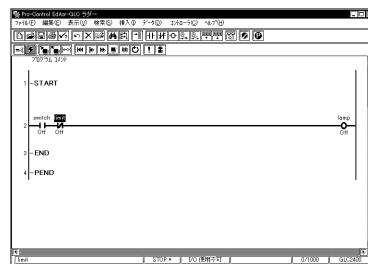
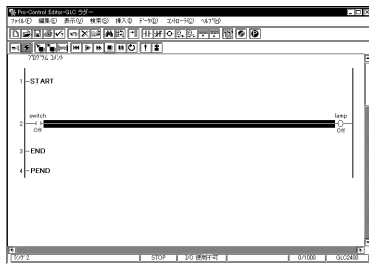


この時、変数追加をした場合は変数追加命令も同時に行われます。

ラングの置換

既存行のラダー回路を編集します。

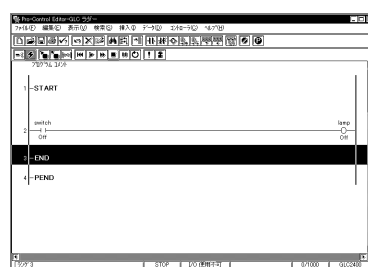
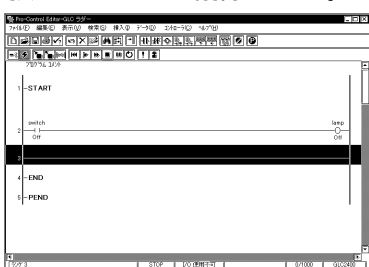
命令の挿入、置換、削除ができます。



この時、変数追加をした場合は変数追加命令も同時に行われます。

ラングの削除

選択したラングを削除します。

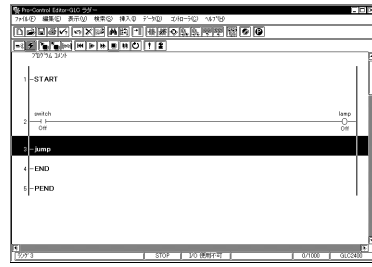
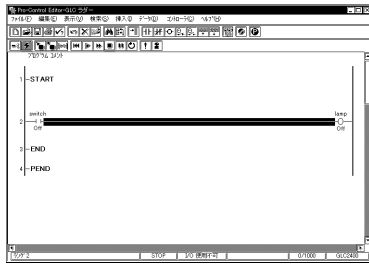


この時、変数は削除されません。

ラベルの追加

ラベルを追加します。

[挿入]の[ラベル]を選択します。

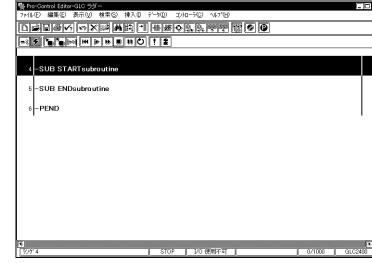
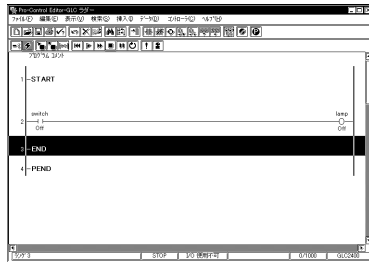


サブルーチンの追加

サブルーチンを追加します。

サブルーチンは、ENDラベルとPENDラベルの間に挿入されます。

[挿入]の[サブルーチン]を選択します。



変数の追加

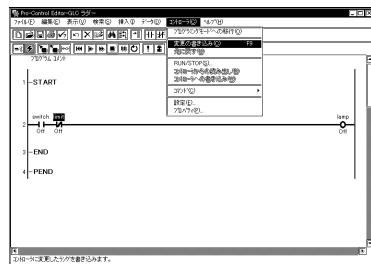
新規の変数を追加します。

[データ]の[変数タイプ]または命令の挿入時に追加することができます。

編集ロジックプログラムの書き込み

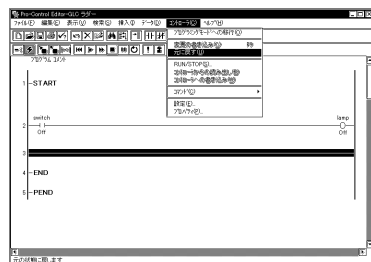
[コントローラ]の[変更の書き込み]で編集したロジックプログラムをGLCに書き込みます。

または編集後、他のラングを編集すると書き込まれます。



編集ロジックプログラムを元に戻す

直前に編集したロジックプログラムをラング単位で元に戻します。



5.8.2 データ保存

ロジックプログラムは、Pro-Control Editor で作成後、[コントローラへの書き込み] で一旦 FEPRROM に書き込まれます。書き込み後最初の GLC 起動時に、FEPRROM の内容を SRAM にコピーします。

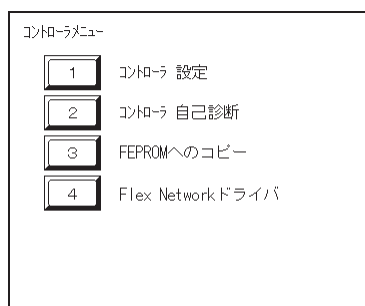
オンラインエディットでは、この SRAM に保存されているロジックプログラムを編集します。オンラインエディット後 GLC を再起動した場合、通常 GLC は SRAM に保存されているデータを読み出しますが、SRAM に保存されたデータは電源 OFF 時のバッテリー切れ¹などにより SRAM のデータが失われることがあります。その場合、次回 GLC 起動時には FEPRROM に保存されているロジックプログラムが読み出されます。

SRAM のデータが失われた場合に備え、GLC のオフラインメニューで [FEPRROM へのコピー] を実行するか、Pro-Control Editor で WLL ファイルとして保存することにより、バックアップしてください。

FEPRROM へのコピー

GLC のオフラインメニュー [FEPRROM へのコピー] を選択します。

注意：オフラインメニューへ移行すると、GLC のロジックプログラムや表示機能は停止し、その後、初期状態から起動します。



FEPRROM へ編集したロジックプログラムをコピーすることで、SRAM に保存されているロジックプログラムが失われても FEPRROM からロジックプログラムを読み出すことでシステムは継続して運転が可能となります。

【重要】FEPRROM へのコピーを実行していない状態で GLC を再起動すると“ No logic in FEPRROM ”という警告メッセージが表示されます。通常は SRAM に保存されているデータを読み出しますが、SRAM のデータが失われた場合は、FEPRROM に保存されているオンラインエディットによる編集前のデータを読み出します。SRAM のデータが失われた場合に備えて、FEPRROM へのコピーを必ず実行してください。

【重要】SRAM のデータが失われたとき、FEPRROM からロジックプログラムの読み出しは自動的に行われます。ただし、この状態のときは、マイナー異常のエラーが発生しますので自動的に FEPRROM のロジックプログラムで実行すると問題があるシステムでは [継続異常スイッチ] を選択してロジックプログラムが自動的に実行しないように設定してください。

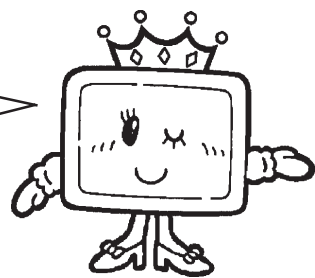
¹ リチウム電池の寿命は電池周囲温度 40 以下で 10 年以上、50 以下で 4.1 年以上、60 以下で 1.5 年となります。バックアップ期間は初期状態（満充電）で約 60 日、電池寿命時で約 6 日です。

Pro-Control Editor での保存

Pro-Control Editor でオンラインエディット終了後、プログラミングモードに移って、編集したロジックプログラムを[保存]することで、WLLファイルとして保存することができます。WLLファイルとして保存されているロジックプログラムをGLCにダウンロードすることで編集後のロジックプログラムを実行することができます。

MEMO

このページは、空白です。
ご自由にお使いください。



第6章 Pro-Control Editor と GP-PRO/PB

GP-PRO/PB を使用すれば、Editorで作成したGLC用の変数とリンクした操作画面を作成して、コントローラや稼動しているロジックプログラムの操作やモニタをすることができます。本章では、GP-PRO/PB でのGLC用画面の作成方法について説明します。本章は簡単なチュートリアル形態になっており、タンクからの水をポンプで汲み上げるために設計したロジックプログラムとリンクした画面をGP-PRO/PB で作成します。

6.1.GP-PRO/PB での変数表示

GLC用の変数をGP-PRO/PB でインポートする方法について説明します。

6.1.1.GP-PRO/PB プロジェクトを開く手順

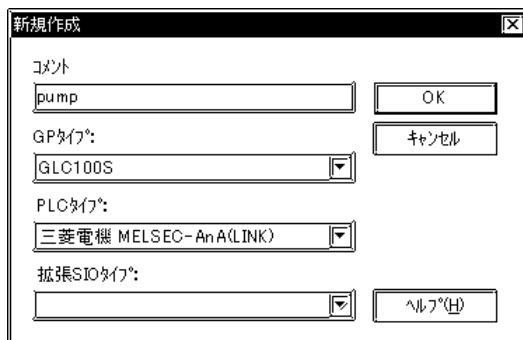
1. GP-PRO/PB を起動すると、[プロジェクトマネージャ]ウィンドウが開きます。



6.1.2.GPタイプとPLCタイプの選択

[GPタイプ]/[PLCタイプ]選択手順

1. 「プロジェクトの新規作成」により新しくGPタイプ/PLCタイプを選択します。GLCでサポートされている機種のみ有効です。



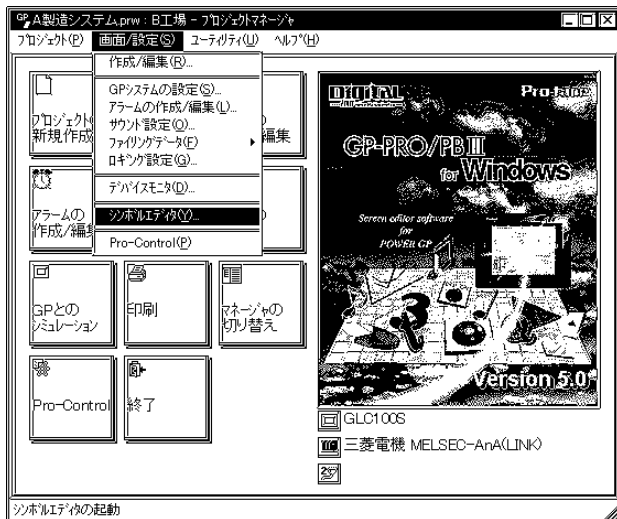
参考：GLCは内部コントローラと同時に通信機能を使って外部PLCとも同時に接続できます。ここで選択するPLCは、外部PLCです。外部PLCを接続しない場合は、「メモリリンクSIOタイプ」を選択します。

6.1.3. インポート

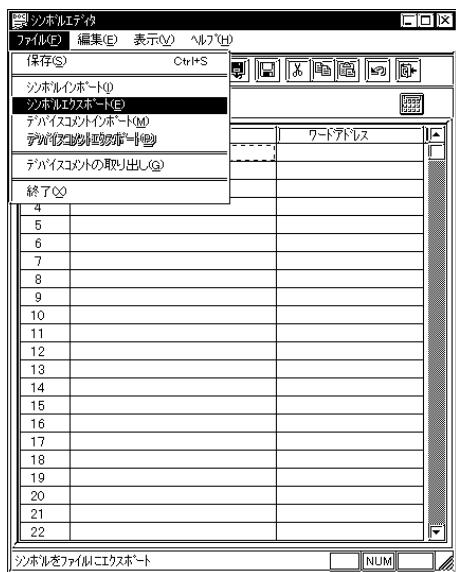
次にプロジェクト(画面ファイル)と.WLLファイル(ロジックプログラムファイル)との対応づけを行います。

[インポート]操作手順

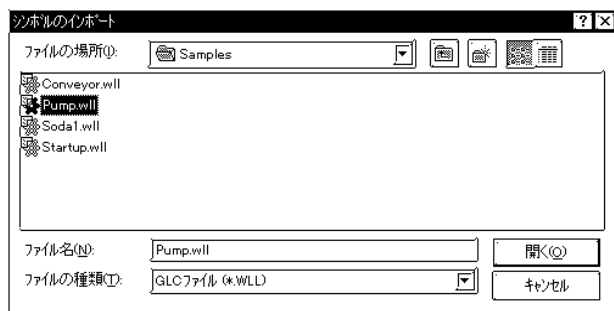
1. [画面 / 設定]メニューから[シンボルエディタ]を選択します。



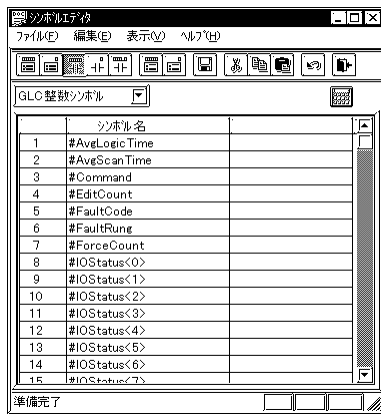
2. [シンボルエディタ]ウィンドウが表示されます。
[シンボルインポート]を選択します。



3. ファイルの種類「GLC ファイル (*.WLL)」を選択してください。



4. Pump.wll を選択し、[開く]をクリックします。Pump.wll の変数が GLC シンボルとして GP-PRO/PB にインポートされます。



次に .wll ファイルからインポートした GLC シンボルを GP-PRO/PB による作画画面のオブジェクト(部品、タグなどの要素)にリンクします。

変数の制限事項

GP-PRO/PB で GLC の変数を扱う場合、以下のような制限があります。

- 通常の変数のエクスポートではコントローラ変数は出力しません。
- 通常の変数をコピーし貼り付けした時に、コントローラ変数内に存在する変数は設定できません。
- 通常の変数の入力ではコントローラ変数内に存在する変数は設定できません。
- GP タイプの変更で GLC から GLC 以外の GP に変更された場合、コントローラ変数が設定されている時、コントローラ変数は通常変数に変更され自動的に割り当てられたアドレスは全て削除され、コントローラ変数を設定していた画面は、自動的に転送準備が必要な状態となります。
- コントローラ変数を含む画面をシュミレーションした場合、シュミレーション画面のデバイス情報にはコントローラ変数を設定したデバイスは表示されません。
- GP-PRO/PB はご使用のパソコン内部の文字フォントやグラフィック機能を使用して表示します。このため、これらの表示は GLC へ転送後、GLC 上での表示とパソコン上での表示に多少の相違が生じる場合があります。あらかじめご了承ください。
- GLC100 では、Pro-Control Editor の変数にデバイスタイプがないので GP-PRO/PB 機能の E タグ、K タグの間接指定のデバイスタイプ & アドレスの指定はできません。
- Pro-Control Editor で配列を使用する場合、GP-PRO/PB 側で配列の要素を削除しないでください。
- GLC の変数は 32bit デバイスの Low/High 順で扱われます。
- GP-PRO/PB では GLC の配列の '['] ' を '<' >' で表します。
- GP-PRO/PB で扱う GLC 変数の数は 2048 個です。配列の 1 要素を 1 つの変数として扱われます。インポートして変数が 2048 個を超える場合、GP-PRO/PB のタグ/部品で使用しない変数は削除してください。

6.1.4.GP-PRO/PB による画面作成

Pro-Control Editor ロジックプログラムの動作を見る画面を GP-PRO/PB 上で作成します。

Pro-Control Editor の変数を GP-PRO/PB へドラック&ドロップする手順

Pro-Control Editor で入力したロジックプログラムの命令をドラッグして、GP-PRO/PB にドロップすると命令に該当するパーツのダイアログが自動的に表示されます。

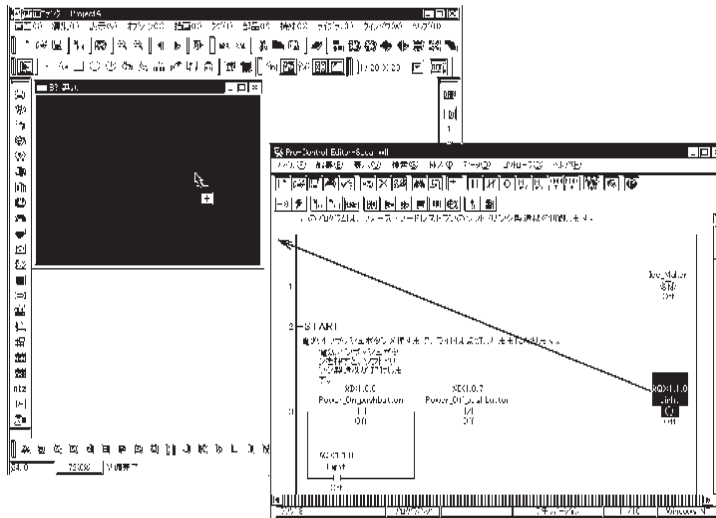
[重要]

- あらかじめGP-PRO/PB でPro-Control Editorの変数をインポートしておく必要があります。インポートしていない変数を割り付けた命令をドラッグ&ドロップした場合は警告メッセージを表示した後、変数はPro-Control Editorの変数ではなく、GP-PRO/PB のシンボルとして登録されます。(インポートしない限り動作しません。)この場合、GP-PRO/PB でPro-Control Editorの変数をインポートしなおすとGP-PRO/PB のシンボルとして登録された変数がPro-Control Editorの変数として認識されます。
- 銘板をサポートしているパーツでは銘板に変数名が入ります。
- 1種類の命令に複数種類のパーツから1種類のパーツが選択できるので、ドラック&ドロップしたときにパーツを選択するメニューが表示されます
- GP-PRO/PB のパーツをコピーする場合、パーツにはあらかじめGLCの変数が割り付けられている必要があります。
- GP-PRO/PB のパーツをドラッグ&ドロップする場合、[CTRL]キーを押しながらドラックしてください。

Pro-Control EditorからGP-PRO/PB へのコピー	
Pro-Control Editor 命令	GP-PRO/PB パーツ
NO(a接点)	ビット/トグルスイッチ
NC(b接点)	ビット/トグルスイッチ
PT(立ち上がり接点)	ビット/トグルスイッチ
NT(立ち下がり接点)	ビット/トグルスイッチ
OUT/M(アウトコイル)	ランプ
NEG/NM(反転コイル)	ランプ
SET/SM(セットコイル)	ランプ
RST/RM(リセットコイル)	ランプ
CTU(アップカウンタ)	数値表示器/グラフ/設定表示器
CTD(ダウンカウンタ)	数値表示器/グラフ/設定表示器
CTUD(アップダウンカウンタ)	数値表示器/グラフ/設定表示器
TON(オンディレータイマ)	設定表示器
TOF(オフディレータイマ)	設定表示器
TP(パルスタイマ)	設定表示器

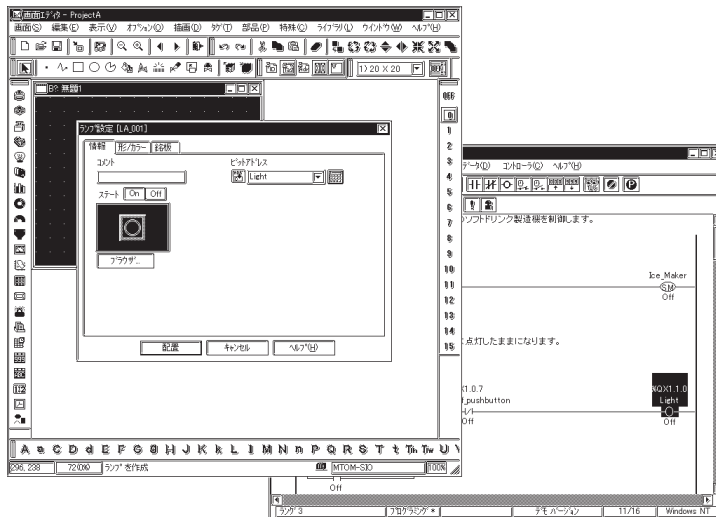
GP-PRO/PB からPro-Control Editorへのコピー	
GP-PRO/PB パーツ	Pro-Control Editor 命令
ビット/トグルスイッチ	NO(a接点), NC(b接点), PT(立ち上がり接点), NT(立ち下がり接点)
ランプ	NO(a接点), NC(b接点), PT(立ち上がり接点), NT(立ち下がり接点)OUT/M(アウトコイル), NEG/NM(反転コイル), SET/SM(セットコイル), RST/RM(リセットコイル)
数値表示器/グラフ/設定表示器	CTU(アップカウンタ), CTD(ダウンカウンタ), CTUD(アップダウンカウンタ)
設定表示器	TON(オンディレータイマ), TOF(オフディレータイマ), TP(パルスタイマ)

1. Pro-Control Editor の変数を選択し、GP-PRO/PB の画面エディタへドラックします。

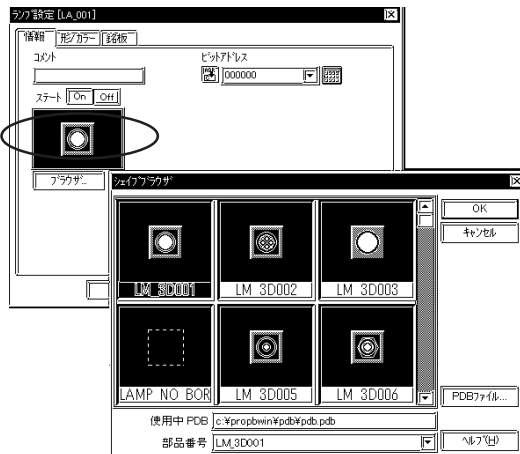


2. GP-PRO/PB の画面エディタでドロップします。

3. GP-PRO/PB の画面エディタ内で[ランプ設定]画面が自動的に開きます。

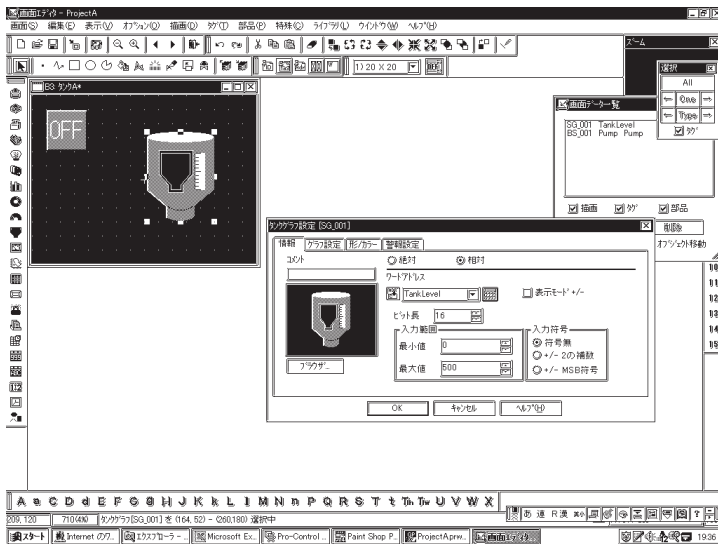
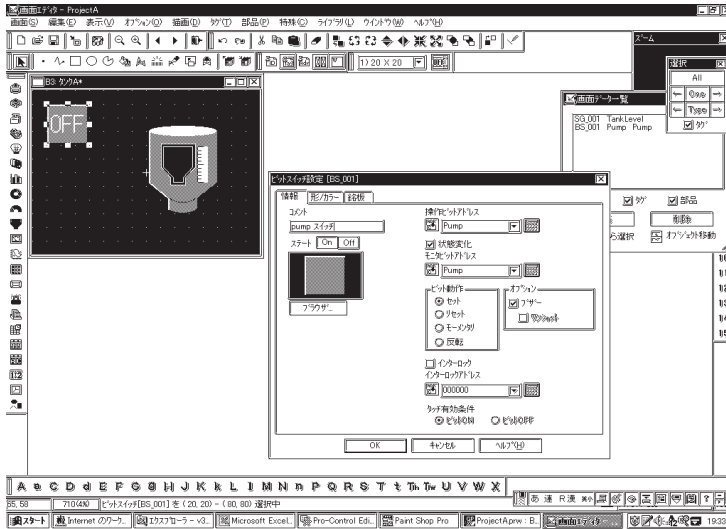


4. ブラウザからパーツを選択します。



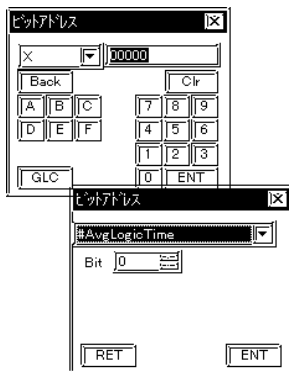
画面作成手順

1. [画面の作成 / 編集]メニューを選択します。
2. 画面エディタで画面を作成します。



タグにおいてコントロール変数の整数型のビット指定

Pro-Control Editorでは整数型変数に拡張子をつけることでビット指定ができます。同様にGP-PRO/PBでも、インポートした整数型変数に拡張子を付けることでビット指定できます。



整数型変数とは、Pro-Control Editorでビット指定を使用する / しないに関わらず、GP-PRO/PBには通常の整数型変数のみインポートされます。GP-PRO/PBで整数型変数にビット単位でアクセスしたい場合は、GP-PRO/PBでビット指定します。

6.2. Pro-Control Editor の変数を GP-PRO/PB のオブジェクトにリンク する

画面は、GP-PRO/PB で提供される組み込み済みの作画機能のオブジェクト(部品、タグなどの要素)を使って作成します。次に画面をGLCへダウンロードします。GLCはお使いのコンピュータのシリアルポートに接続してください。

GLCの接続の詳細[参照 Editor のオンラインヘルプの「GLCのセットアップ」]

6.3. GLC へのダウンロード

コントロールアプリケーションを動作させる前に、GP-PRO/PB で作成した画面をGLCへダウンロードします。

画面作成ソフト GP-PRO/PB で作成した画面をGLCにダウンロードする手順

1. [画面の転送]をクリックします。
2. [画面の転送]ウィンドウが表示されます。エラーが発生した場合にはそのエラーとともに、ダウンロードの進行状況が表示されます。



参考: ダウンロード中、他のプログラムはコンピュータの同じポートを使用することはできません。

6.4. ポンププロジェクトの実行

サンプルプログラムが動作しているのを確認するには、コントローラを実行してください。

コントローラのロジックプログラムを GLC にダウンロードする手順

1. Pro-Control Editor を起動します。
2. ¥Pro-Control Editor¥Samples¥Pump.wll を開きます。
3. [コントローラ]メニューで[コントローラへの書き込み]を選択します。
4. しばらくの間[ダウンロードの進行状況]ウィンドウが表示されます。
5. [コントローラ]メニューで[モニタリングモードへの移行]を選択します。
6. [コントローラ]メニューで[RUN/STOP]を選択します。コントローラのコントロールパネルが表示されます。
7. [RUN]をクリックします。

これで、GLCにダウンロードされたコントローラがロジックプログラムPump.wllを実行します。

参照：第4章 ロジックプログラムを実行する

動作確認

1. GLC 画面上の[ON]ボタンをタッチします。ポンプでタンクから汲み出すにつれて、液レベルが下がるのを確認してください。
2. GLC 画面上の[OFF]ボタンをタッチします。ポンプがそれ以上タンクから汲み出さず、液レベルが上がるのを確認してください。

上記内容が問題なく動作している場合は、正常です。動作しない場合は、もう一度、手順を1からやり直してください。

まとめ

本章では以下の方法について説明しました。

- ・ GP-PRO/PB プロジェクトを開く方法
- ・ コントローラとリンクして動作する GP-PRO/PB によるプロジェクトの作成方法
- ・ GLC 変数を、GP-PRO/PB プロジェクトにインポートする方法
- ・ Pro-Control Editor の変数を GP-PRO/PB の作画オブジェクト(部品、タグなどの要素)にリンクさせる方法
- ・ GP-PRO/PB と Pro-Control Editor を組み合わせた GLC での実行手順

第7章 Pro-Control EditorとPro-Server (対応機種: GLC2400)

Pro-Serverを使用すれば、イーサネット経由でGLC変数のリード/ライトおよび2Way機能(配信、アクション機能など)を実行できます。

[参照 Pro-Serverの詳細については[Pro-Server with Pro-Studio for Windows オペレーションマニュアル]]

本章では、Pro-ServerでGLC変数を使用する方法について説明します。

7.1.GLC変数のインポート

Pro-StudioでGLC変数を扱うためには、GP-PRO/PB にインポートされたGLCシンボルを、Pro-Studioを使ってインポートします。したがって、GLC変数をPro-Serverにインポートするには、まずGP-PRO/PB のシンボルエディタでGLCシンボルとしてインポートしておく必要があります。

インポートしたGLC変数はGLCローカルシンボルとして登録されます。

このGLCローカルシンボルは、

- ・GLCタイプの参加局のみ存在するシンボル
- ・編集・削除はできない
- ・デバイスの種類はビットと32ビット16進のみとなります。

GLC変数のインポート手順

Pro-StudioでGLCシンボルをインポートする方法について説明します。

あらかじめGLC変数をGP-PRO/PB でインポートしておきます。

[参照 6.1.GP-PRO/PB での変数表示]

1. Pro-Studioを起動します。
2. ネットワークプロジェクトファイルを作成または選択します。
3. [編集]の[参加局の登録]を選択します。
「PRO/PB のプロジェクトファイル」でリンクするPRWファイルを選択します。

4. [ツール]の[GLC変数のインポート]を選択します。
5. メイン画面の右にある[シンボル名(アイテム)]にGLCシンボルが表示されます。

7.2.S100 ファイルチェック

GLC変数のインポートにより各参加局が持っているS100ファイル、リンクするPRWファイルの持っているS100ファイル、およびGLC本体にダウンロードされているS100ファイルの内容が異なると、不正なデバイスをアクセスする可能性があります。そのため、各S100ファイルの日付をチェックし、異なれば警告表示します。

1. Pro-Server (Pro_API)側

ネットワークプロジェクトファイルがロードされたとき、各参加局の持っている[S100ファイルの日付]と、リンクされているPRWファイルの日付を比較し、異なれば以下の警告表示します。

2. 2WayDriver (GLC 本体)側

GLC起動時に各参加局の持っている[S100ファイル日付]と、GLC本体にダウンロードされているS100ファイルの日付を比較し、異なれば警告表示します。

付録A：エラーと警告

プログラムでエラーチェックが行われると、[エラーチェック]ウィンドウにエラーや警告が表示されます。これらのエラーや警告は、プログラムのロジック、変数、I/Oなどに起こる問題に関するものです。エラーの原因ごとにエラーコードが割り付けられています。このコードの値を参考にすると、エラーや警告の発生原因が特定できます。

200-299：ロジックのエラーと警告

ラダープログラムの命令に関する情報については、メインウィンドウで選択してから[ヘルプ]メニューで[関係トピックヘルプ]を選択して[F1]キーを押してください。

エラー 200

パラメータはディスクリートでなければなりません。

命令にディスクリートのオペランドが必要です。以下のいずれかです：

ディスクリート値

ディスクリート配列の要素

整数値のディスクリート要素

エラー 201

パラメータはカウンタでなければなりません。命令にカウンタの値が必要です。

エラー 202

パラメータはタイマでなければなりません。命令にタイマの値が必要です。

エラー 203

・・・は整数または実数でなければなりません。命令に、変数または定数として整数または実数が必要です。

エラー 204

・・・は定数でない整数または実数でなければなりません。命令に整数または実数の変数が必要です。命令を定数にすることはできません。

エラー 205

・・・は整数でなければなりません。命令にデータ値が整数である変数または整数の定数が必要です。

エラー 206

・・・は整数でなければなりません。しかし配列は許されません。命令にデータ値が整数である変数または整数の定数が必要です。命令を配列にすることはできません。

エラー 207

・・・は定数でない整数でなければなりません。命令に整数の変数が必要です。命令を定数にすることはできません。

エラー 208

パラメータがラベルでなければなりません。命令にラベル名が必要です。そのラベル名が存在しなければなりません。

エラー 209

パラメータがサブルーチンでなければなりません。命令にサブルーチン名が必要です。

エラー 210

ラベルが範囲外のところで使用されています。指定したラベルは存在しますが、有効範囲外です。

エラー 211

サブルーチンは自分自身をコールできません。サブルーチンの中でJSR命令で同じサブルーチンをコールします。

エラー 212

・・・は・・・と同じタイプでなければなりません。2個のパラメータが同じタイプ(整数、実数など)でなければなりません。

エラー 213

・・・は・・・と同じサイズでなければなりません。2個のパラメータが同じサイズでなければなりません。つまり2個のパラメータはどちらも、「同じ数の要素をもつ配列」または「非配列」でなければなりません。

エラー 214

XはYと同じサイズまたは、整数でなければなりません。2個のパラメータが同じサイズであるか、または2番目のパラメータの方が大きいサイズのように見なされる整数でなければなりません。

エラー 215

Xは整数または実数またはディスクリートの配列でなければなりません。命令に変数、定数、または配列として、整数、実数、またはディスクリートが必要です。

エラー 216

・・・は定数でない整数または実数、またはディスクリート配列でなければなりません。命令に変数または完全な配列として、整数、実数、またはディスクリートが必要です。定数にすることはできません。

警告 217

両方のパラメータは定数です。命令が2個の定数を比較しています。

警告 218

入力パラメータが出力命令に使用されました。変数は入力にマークをしているのに([変数タイプ]ウィンドウを参照) 出力命令で使用されています。I/O割り付けを確認してください。

警告 219

設定値が " 0 " です。

警告 220

設定値が " 0 " です。

警告 224

パラメータに保持型のデータを使用できません。命令のパラメータに割り付けた変数に保持型データを使用できません。

警告 225

・・・は整数配列でなければなりません。命令のパラメータに割り付ける変数は、整数配列でなければなりません。

エラー 250

重複ラベルは使用できません。同じラベルが、2回以上定義されています。プログラムの別の部分であっても重複はできません。

警告 251

空サブルーチン実行上意味がありません。サブルーチンにラングがありません。

警告 252

空ラングは実行上意味がありません。ラングに命令がありません。空のラングの修正を選択しないと、プログラムに効果が生じません。

警告 253

空分岐は実行上意味がありません。分岐に命令がありません。空の分岐の修正を選択しないと、プログラムに効果が生じません。

エラー 254

出力、演算などの制御命令がラングの最後になければなりません。命令の右側に何もありません。

警告 255

・・・が複数のタイマで使用されています。同一のタイマ変数が複数のタイマ命令で使用されています。結果は不定となります。[リファレンス]ダイアログボックスを使用して他のタイマ命令を検索し変数名と変更してください。

エラー 256

・・・が複数のカウンタで使用されています。同一のカウンタ変数が複数のカウンタ命令で使用されています。結果は不定となります。[リファレンス]ダイアログボックスを使用して他の命令を検索し変数名を変更してください。

エラー 257

ラング上の最後の命令は出力命令でなければなりません。命令が出力命令になっていません(したがってパラメータの値が変わりません)。

エラー 258

複数の出力命令を使用できません。出力命令は右側に他の命令をもつことはできません。

エラー 259

分岐の最後の命令は出力命令でなければなりません。出力命令は右側に他の命令をもつことはできません。

エラー 260

ネスティングのレベルが最大を超えました。ラングの分岐レベルが多すぎます(レベルの最大数は25です)。ラングを小さくいくつか分割してみてください。

エラー 262

プログラムが大きすぎます。GLCのフラッシュメモリの使用可能な容量を超えています。

300-399 : 変数のエラーと警告

警告 300

変数タイプは入力または出力ですがI/Oアドレスが割り付けられていません。変数は入力または出力にマークをしているのに([変数のタイプ]ウィンドウを参照) I/Oにマップされません。

エラー 301

タイプが設定されていません。変数に変数タイプが割り付けられていません。変数タイプを割り付けるには、[変数タイプ]ウィンドウを使用してください。

エラー 302

ラベルが見つかりません。存在しないラベルがジャンプサブルーチン(JSR)命令の飛び先として指示されています。

エラー 303

参照される変数はタイマまたはカウンタでなければなりません。タイマまたはカウンタの変数の要素を指定したにもかかわらず、別のタイプの変数になっています。[変数タイプ]ウィンドウを参照してください。

エラー 304

変数タイプは整数でなければなりません。変数を使用して配列の要素かまたは修飾語を指定しました。この変数は整数でなければなりません。[変数タイプ]ウィンドウを参照してください。

エラー 305

配列変数でない変数に配列参照されました。配列の要素を指定しましたが、その変数が配列として指定されていません。[変数タイプ]ウィンドウを参照してください。

エラー 306

配列の範囲を越えて参照しています。配列のサイズと等しいかそれ以上の定数を使用して、配列の要素を指定しました。(有効な要素には0からサイズ-1の番号がついています。)サイズは[変数タイプ]ウィンドウで変更できます。

エラー 308

修飾参照が範囲を超えています。範囲を超えたビット、バイト、ワードの要素を指定しています。

エラー 309

変数の参照が正しくありません。カウンタの変数にタイマの参照を指定したり、その逆を指定しています。

警告 310

・・・がすでに存在します。置き換えできません。その名前の変数はすでに存在しています。[変数のインポート状態]ウィンドウで[OK]をクリックすると、元からあった変数は新しい変数に置き換えられます。

エラー 311

クリップボードのバッファのフォーマットが認識されません。現在のクリップボードの中身は、[変数リスト]ウィンドウへの貼り付けには適していません。

エラー 312

警告が多すぎます。[変数のインポート状態]ウィンドウでは、一定の数の警告しか表示しません。このメッセージが表示されたら、他にも表示されていない警告があるということです。

警告 313

・・・の右括弧 "]" がありません。配列には、[]で囲んだサイズが必要です。たとえば、整数 [10] など。

警告 314

配列のサイズは・・・で無効です。サイズ1が使用されます。この変数は配列を意図しているようですが、サイズが認識されません。サイズは、整数[10]のように、[]で囲んだ整数にしなければなりません。

警告 315

・・・に未知の変数・・・があります。変数タイプは未定義となります。この変数はPro-Controlの変数タイプとして認識されません。以下の原因が考えられます：

- ・スペル間違い
- ・先頭か末尾にブランクがある

警告 316

・・・に未サポートの配列・・・があります。設定した配列を無視します。この変数では配列は設定できません。

エラー 317

変数名が無効です...変数名として無効な文字を入力しました。

エラー 318

エラー数が多すぎます。

エラー 320

I/O変数が多すぎます。

エラー 321

変数が多すぎます。変数の数を減らしてください。

400-499 : I/O のエラーと警告

エラー 400

変数名がすでにマップされています。この変数は複数のI/Oポイントにマップされています。
[I/Oのコンフィギュレーション]ウィンドウを参照してください。

500-549 : 一般的な I/O ドライバのエラー

エラー 501

コントローラの変数と、変数が割り当てられるI/Oターミナルの種類が一致しないときに、表示されます。

エラー 502

コントローラの変数と、変数が割り当てられるI/Oターミナルの種類が一致しないときに、表示されます。

エラー 503

コントローラの変数と、変数が割り当てられるI/Oターミナルの種類が一致しないときに、表示されます。

エラー 504

コントローラの変数と、変数が割り当てられるI/Oターミナルの種類が一致しないときに、表示されます。

エラー 505

コントローラの変数と、変数が割り当てられるI/Oターミナルの種類が一致しないときに、表示されます。

エラー 506

ドライバがコントローラの変数を確認しないときに、表示されます。

800-899 : 特定の I/O ドライバのエラー

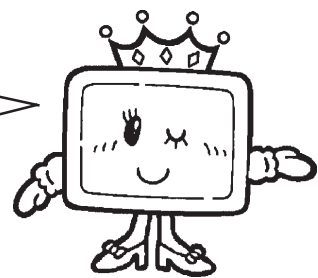
I/Oドライバに関するエラーについて詳しくは、I/Oドライバのオンラインヘルプを参照してください。

900-1000 : 特定の I/O ドライバの警告

I/Oドライバに関する警告について詳しくは、I/Oドライバのユーザーガイドを参照してください。

MEMO

このページは、空白です。
ご自由にお使いください。



付録B：用語集

16進数

16が1単位となる整数の表記法です。この値は、前に16#をつけて入力することができます。たとえば、16#FFは255になります。

GP-PRO/PB

GP-PRO/PB for Windows Ver.4.0以上を称し、(株)デジタルが提供する作画ソフトウェアで、GLCのプログラマブル表示器機能を設計し、設定するのに使用されます。

GLC

(株)デジタル社製グラフィックロジックコントローラの総称です。

I/O

Input(入力)/Output(出力)。コントローラは、デジタル製I/Oユニットやサードパーティの提供するI/Oユニットにより現実のデバイスに接続されます。

I/Oアドレス

I/Oモジュールに割り付けるときのアドレス値です。I/Oアドレスのフォーマットは、割り付けるドライバによって異なります。

IEC61131-3

国際電気標準会議(IEC)の制定した標準で、命令リスト(IL)、ラダーロジック図(LD)、機能ブロック図(FBD)、構造化テキスト(ST)、順次機能チャート(SFC)の5つの制御言語について印刷および表示方法を定義します。

Pro-Control Editor

Pro-Control Editorはコントローラ(ランタイム)へのフロントエンドとして機能します。あらゆるプログラムの設計やモニタリングを行います。

ウォッチドッグタイマ

一定時間内にENDラングまで実行できなかったとき、ウォッチドッグタイマがメジャー異常を検出します。ウォッチドッグタイマの設定は[設定]ダイアログボックスで設定できます。

エラー

エラーには、メジャー異常、マイナー異常、I/Oエラーの3つがあります。

メジャー異常が発生すると、コントローラは、すぐにロジックプログラムの実行を停止します。エディタには、「メジャー異常」と表示されます。状態をクリアするには、コントローラを[RUN/STOP]ダイアログボックスでリセットしてください。

マイナー異常は、軽微なエラーです。

I/Oエラーは、I/Oの読み込み/書き込みのエラーです。

強制変更(Forces)

ディスクリット変数は強制的にONまたはOFFにできます。これはロジックで実行する動作に優先するものです。たとえば、変数が強制的にOFFにされた場合、ロジックがONにしようとしても、OFFのままです。プログラムの強制リストは、[強制リスト]ウィンドウで見ることができます。

クリップボード(Clipboard)

Windowsの機能で、コピーや貼り付けのための一時記憶場所です。異なるアプリケーション間でも、ひとつのアプリケーション内でも使用可能な機能です。

コメント(Descriptions)

コメントは最大32767文字(半角)のテキストで、プログラムの一部についてコメントするものです。コメントの要約は、[コメントリスト]ウィンドウで見ることができます。

コントローラ

コントローラはロジックプログラムを実行し、I/Oを制御します。コントローラは表には現れず、GLCの拡張タスクとして実行します。エディタはモニタリングモードでコントローラを監視します。

サブルーチン(Subroutine)

個々に別の名前のついたロジックのグループ。サブルーチンはENDとPENDというマーカの間位置し、他のサブルーチン内には配置できません。[挿入]メニューで[サブルーチン]をクリックすると、[サブルーチン開始]と[サブルーチン終了]の両方のマーカが作成されます。2つのマーカの間ロジックを挿入することができます。サブルーチンはJump Subroutine (JSR) (ジャンプサブルーチン)という命令で呼び出されます。サブルーチンはどこからでも何度でも呼び出すことができ、コードを一回書き込むだけですむ点が便利です。サブルーチン名が必要です。

サブルーチン名

サブルーチン名には、文字・数字・下線を32文字まで使用できます。サブルーチン名を数字から始めることはできません。

システム変数(System Variables)

「システム変数」は事前に定義された特殊な変数で、コントローラの状態に関する情報を提供し、コントローラの操作に反映させます。自動で作成され削除不可能な点を除けば、通常の変数とよく似た働きをします。

実数(Real)

小数点を含む数や特定の記数法で表す数。Pro-Control Editorでは実数の範囲を $\pm 2.25e^{-308}$ から $\pm 1.79e^{-308}$ とします。有効数字は最大15桁です。

状態フロー(State Flow)

モニタリング時パラメータに基づいて個々の命令の動作を緑色で強調表示します。パラメータの状態に応じて、それぞれの接点が強調表示されます。

整数(Integer)

32ビットの情報の含まれる記憶単位。整数には、-2,147,483,648から2,147,483,647(16進数で16#0000000から16#FFFFFF)までの値が割り付けられます。整数には小数点を含むことはできません。

接続線

命令の間の垂直線です。新規の命令を挿入する場合は、挿入したい位置の接続線にフォーカスをあわせます。

ディスクリート(Discrete Point)

OFFまたはONのいずれかの状態を持つ変数です。

[データ値表示リスト]ウィンドウ(Data Watch List Window)

登録した変数のモニタリング状況がこのウィンドウに表示されます。[オプション]ダイアログボックスで更新の基準を調整することができます。

定数

42(整数)や3.14159(実数)などの数です。

デモモード(Demonstration Mode)

ロジックプログラムを編集するだけのモードです。ロジックプログラムのダウンロードやモニタリングモードの実行はできません。

ドラッグ(Drag)

マウスの左ボタンを押したままマウスを動かしてから離します。マウスポインタにより、そこがドラッグできる有効な場所かがわかります。

内部変数(Internal Variable)

I/Oに割り付けられている以外の変数です。

バイト(Byte)

8ビットの情報をもつ記憶単位。バイトには0から255の値を割り当てます。Pro-Control Editorの整数は4バイトから成ります。

配列

(1つの名前を割り付けた)同じタイプの複数の要素です。

パラメータ(Parameter)

(命令に割り付けられている、変数、変数の要素、定数、ラベル名)への入力または(命令に割り付けられている、変数、変数の要素、定数、ラベル名)からの出力。

パワーフロー(Power Flow)

ラダーロジックプログラムの実行する流れ。

ビット (Bit)

基本的な記憶単位、値は1または0。

フォーカス(Focus)

ラダーロジックでの選択事項を強調表示するための黒色のカーソルのことです。

分岐

ラングに並列接続してロジックプログラムを実行します。

ブックマーク(Bookmark)

ロジックのどこにでも配置できる非表示のマーカーで、プログラムのその箇所にすぐに戻ることができます。

ブレークポイント

ロジックプログラム中にあり、ブレークポイントでロジックプログラムの実行を一時停止します。

プログラミングモード

ロジックプログラムを含む拡張子が.WLLのファイルを編集します。

変数(Variable)

カウンタ、I/Oの値などのデータ値です。理解しやすい変数名をつけてください。

通常、変数は自動的に作成されます。パラメータボックスか[I/Oコンフィギュレーション]ウィンドウで新規の名前を入力すると、エディタは適切なタイプで新規の変数を自動的に作成します。[変数リスト]ウィンドウで不要な変数を削除できます。変数名は文字、数字、下線を含めて最大20文字で定義しますが、数字で始めることはできません。

母線

メインウィンドウの両端にある2本の垂直な線です。

モニタリングモード

エディタはコントローラに書き込んだロジックプログラムをモニタリングします。

たとえば: Power_Off_pushbutton、ResetButton、ALARM2 などです。

命令(Instruction)

ロジックプログラムの基本的な要素(ディスクリート命令、ビット演算命令、データ操作命令、演算命令、タイマとカウンタ、プログラム制御命令)です。コントローラに特定の機能を実行させる命令です。Pro-Control Editor の命令は、IEC61131-3 の仕様に基づいています。

未定義

タイプのない変数です。エディタで変数を作成するとき、タイプを確定できないと未定義になります。

要素(Element)

要素は変数全体ではなくある一部分の名前のことです。以下のような部分が該当します：

- ・ タイマの変数やカウンタの変数の要素
- ・ 配列の要素
- ・ 整数の一部；変更子を参照してください。

たとえば：FillTimer.ET、LimitSwitches<5>、LimitSwitches<Index>、Flags.X<12>、SensorArray<Position>.X<Index>

ラング

1つ以上の命令で2本の母線の間で作成する回路です。

ラベル名(Label Name)

最大32文字から成る名前で、ロジックプログラム内の位置を識別したりラベルを付けたりするための名前です。ラベル名を数字から始めることはできません。

ロジックプログラム

アプリケーションプログラム中のラングの集合。Pro-Control Editor ではラダー言語をサポートしています。

ワード(Word)

16ビットからなる記憶単位。ワードには0から65535までの値が割り付けられます。