

FC6A 形

MICROSmart

通信 マニュアル



安全上のご注意

- ・本製品の取り付け、配線作業、運転および保守・点検を行う前に、本書をよくお読みいただき、正しくご使用ください。
- ・本製品は弊社の厳しい品質管理体制のもとで製造されておりますが、万一本製品の故障により重大な事故や損害の発生のおそれがある用途へご使用の際は、バックアップやフェールセーフ機能をシステムに追加してください。
- ・本製品への外部機器からの不正アクセス等に対しては、ネットワークシステム側で対策を講じてください。不正アクセス等により直接または間接的に生じた損失、損害その他の費用について当社は、一切の責任を負いかねますので、ご了承ください。
- ・本書では、誤った取り扱いをした場合に生じることが想定される危険の度合いを「警告」「注意」として区分しています。それぞれの意味するところは、次のとおりです。



警告

取り扱いを誤った場合、人が死亡または重傷を負う可能性があります。



注意

取り扱いを誤った場合、人が傷害を負うか物的損害が発生する可能性があります。



警告

- ・FC6A 形マイクロスマートは、高度な信頼性・安全性が必要とされる用途への使用を想定しておりません。これらの用途に使用しないでください。
- ・上記以外でも、機能・精度において高い信頼性が求められる用途で使用する場合は、組み込まれるシステム機器全般として、フェールセーフ設計や冗長設計等の処置を講じたいうで使用してください。次に具体例を記載します。
 - ・非常停止回路やインタロック回路などは FC6A 形マイクロスマートの外部回路で構成してください。
 - ・出力回路のリレー、トランジスタなどの故障により、出力が ON あるいは OFF の状態を維持することがあります。重大事故の可能性のある出力信号については、外部に状態を監視する回路を設けてください。
 - ・FC6A 形マイクロスマートは自らの自己診断機能により、内部回路もしくはユーザープログラムの異常を検出し、ユーザープログラムの実行を停止させ出力を OFF させる場合があります。出力が OFF 時に組み込まれたシステムが危険に陥らないよう、回路を構成してください。
- ・取り付け、取り外し、配線作業および保守・点検は必ず電源を切って行ってください。破損、感電および火災発生のおそれがあります。
- ・本製品の設置、配線、ユーザープログラムの入力および操作を行うには専門の知識が必要です。専門の知識のない一般消費者が扱うことはできません。
- ・本書に記載の指示にしたがって取り付けてください。取り付けに不備があると落下、故障、誤動作の原因となります。



注意

- ・本製品は、装置内への組み込み設置専用品ですので、装置外には設置しないでください。
- ・カタログ、本書に記載の環境下で使用してください。高温、高湿、結露、腐食性ガス、過度の振動・衝撃のある所で使用すると感電、火災、誤動作の原因となります。
- ・本製品の使用環境の汚損度は " 汚損度 2 " です。汚損度 2 の環境下で使用してください。(IEC60664-1 規格に基づく)
- ・移動・運送時などに本製品を落下させないでください。本製品の破損や故障の原因となります。
- ・配線は印加電圧、通電電流に適した電線サイズを使用し、端子ねじは規定締付トルクで締め付けてください。
- ・設置・配線作業時に配線くずやドリルの切り粉などが本製品内部に入らないように注意してください。配線くずなどが本製品内部に入ると火災、故障、誤動作の原因になります。
- ・定格にあった電源を接続してください。定格と異なる電源を接続すると火災の原因になるおそれがあります。
- ・電源ラインの外側には、IEC60127 準拠品のヒューズをご使用ください。(FC6A 形マイクロスマートを組み込んだ機器を欧州に出荷する場合に適用)
- ・出力回路には、IEC60127 準拠品のヒューズをご使用ください。(FC6A 形マイクロスマートを組み込んだ機器を欧州に出荷する場合に適用)
- ・サーキットブレーカは、EU 承認品をご使用ください。(FC6A 形マイクロスマートを組み込んだ機器を欧州に出荷する場合に適用)
- ・I/O フォース機能の使用、運転、停止などの操作は、十分に安全を確認してから行ってください。操作ミスにより機械の破損や事故の原因になることがあります。
- ・本製品から直接保護接地に接続しないでください。保護接地は装置側で M4 以上のねじを使用して接地してください。(FC6A 形マイクロスマートを組み込んだ機器を欧州に出荷する場合に適用)
- ・分解、修理、改造等を行わないでください。
- ・本製品は電子部品や電池を含んだ製品です。廃棄する場合は、廃棄される国・自治体の法規制にしたがい廃棄してください。

はじめに

本書は、FC6A形マイクロスマートの通信に関わる仕様、設定方法および各通信機能について記載しています。本書をよくお読みいただき、製品の機能および性能をご理解のうえ、正しくご使用くださいますようお願い致します。

関連マニュアル

ご利用目的に応じて以下をご覧ください。

マニュアル名称	内容
FC6A形マイクロスマート ユーザーズマニュアル	FC6A形マイクロスマートの製品仕様、設置と配線の方法、プログラミングのための基本的な操作やファンクション設定の設定方法、デバイスや命令の一覧およびトラブル対策について説明しています。
FC6A形マイクロスマート 通信 マニュアル (本書)	FC6A形マイクロスマートの通信に関する仕様や機能の説明、設定方法および使用例を記載しています。
FC6A形マイクロスマート 温調モジュール ユーザーズ マニュアル	温調モジュールの仕様、機能について説明しています。
ラダープログラミング マニュアル	プログラミングのための基本的な操作およびデバイスや命令の一覧、各種命令の動作について説明しています。

弊社 Web サイト上では随時、最新の製品マニュアル PDF を無償公開しています。最新の製品マニュアル PDF は弊社 Web サイトからダウンロードいただけますようお願いいたします。

改定履歴

2015年12月	初版発行
2016年2月	第2版発行
2016年4月	第3版発行
2017年3月	第4版発行
2017年8月	第5版発行
2018年3月	第6版発行
2018年6月	第7版発行
2018年11月	第8版発行
2019年5月	第9版発行
2019年12月	第10版発行
2020年4月	第11版発行
2020年7月	第12版発行
2020年12月	第13版発行
2021年3月	第14版発行
2021年7月	第15版発行
2022年2月	第16版発行
2022年9月	第17版発行
2023年11月	第18版発行

ご注意

- ・本書に関するすべての権利は、IDEC 株式会社に帰属しています。弊社に無断で複製、転載、販売、譲渡、賃貸することはできません。
- ・本書の内容については、将来予告なく変更することがあります。
- ・製品の内容につきましては万全を期しておりますが、ご不審の点や誤りなど、お気付きの点がございましたら、お買い求めの販売店またはお問い合わせ窓口までご連絡ください。

商標について

- ・WindLDR および MICROSmart は、IDEC 株式会社の日本国での登録商標です。
- ・MELSEC は三菱電機株式会社の登録商標です。
- ・記載されているその他の会社名、製品名は、各社の商標または登録商標です。

法規および適合規格に関して

本製品が対応している各国の法規および適合規格について以下に記載します。

欧州法規・規格

本製品は以下の欧州指令に適合しています。

- 低電圧指令
- EMC 指令
- RoHS 指令
- RE 指令 (FC6A-PC4 のみ)

これらの指令に対応するため、本製品は以下に示す国際規格および欧州規格にもとづき、設計・評価されています。

- IEC/EN 61131-2: 2007
- EN50581:2012
- EN301 489-1 V2.1.1 & EN301 489-17 V2.1.1 (FC6A-PC4 のみ)

北米法規・規格

本製品は UL から以下の認証を取得しています。

- UL508*1
- UL61010-1*1
- UL61010-2-201*1
- CSA C22.2 No.142*1
- CSA C22.2 No.61010-1*1
- CSA C22.2 No.61010-2-201*1
- ANSI/ISA 12.12.01
- CAN/CSA C22.2 No.213

*1 FC6A 形マイクロスマートの内、一部機種は対応していません。適用規格についての詳細は弊社宛にお問い合わせください。

船舶規格

本製品は以下の船級協会から認証を取得しています。

(FC6A-C16R1DE、FC6A-C16P1DE および FC6A-C16K1DE は認証を取得していません。)

- ABS (アメリカ船級協会)
- DNV GL (DNV GL 船級協会)
- LR (ロイド船級協会)
- NK (日本海事協会)
- ブリッジ (船橋) 及びデッキ (甲板) での使用は認証を取得していません。

適用規格や EU 指令の詳細はお買い求めの販売店にお問い合わせいただくか、弊社 Web サイトにてご確認ください。

本書で使う略称、総称、用語

項目	内容
FC6A形	CPUモジュール、増設モジュール、増設拡張モジュール、HMIモジュール、カートリッジベースモジュール、カートリッジの総称です。
CPUモジュール	All-in-One CPUモジュール、CAN J1939 All-in-One CPUモジュール、Plus CPUモジュールの総称です。
All-in-One CPUモジュール	FC6A-C****EのCPUモジュールの総称です。
16点タイプ	入出力の合計点数が16点のAll-in-One CPUモジュールの総称です。(FC6A-C16****)
24点タイプ	入出力の合計点数が24点のAll-in-One CPUモジュールの総称です。(FC6A-C24****)
CAN J1939 All-in-One CPUモジュール	FC6A-C40***EJのCPUモジュールの総称です。
Plus CPUモジュール	FC6A-D****CEEのCPUモジュールの総称です。
Plus16点タイプ	入出力の合計点数が16点のPlus CPUモジュールの総称です。(FC6A-D16****)
Plus32点タイプ	入出力の合計点数が32点のPlus CPUモジュールの総称です。(FC6A-D32****)
40点タイプ	入出力の合計点数が40点のCPUモジュールの総称です。(FC6A-C40****)
AC電源タイプ	電源仕様がAC電源のCPUモジュールの総称です。(FC6A-C****AE、FC6A-C****AEJ)
DC電源タイプ	DC24V電源タイプ、DC12V電源タイプのCPUモジュールの総称です。
DC24V電源タイプ	電源仕様がDC24VのCPUモジュールの総称です。(FC6A-C****CE、FC6A-C****CEJ、FC6A-D****CEE)
DC12V電源タイプ	電源仕様がDC12VのCPUモジュールの総称です。(FC6A-C****DE、FC6A-C****DEJ)
リレー出力タイプ	出力がリレー出力のCPUモジュールの総称です。(FC6A-C**R**E、FC6A-C**R**E*)
トランジスタ出力タイプ	トランジスタシンク出力タイプ、トランジスタプロテクトソース出力タイプのCPUモジュールの総称です。
トランジスタシンク出力タイプ	出力がトランジスタシンク出力のCPUモジュールの総称です。(FC6A-C**K**E、FC6A-C**K**E*、FC6A-D**K*CEE)
トランジスタプロテクトソース出力タイプ	出力がトランジスタプロテクトソース出力のCPUモジュールの総称です。(FC6A-C**P**E、FC6A-C**P**E*、FC6A-D**P*CEE)
増設モジュール	I/Oモジュール、通信モジュール、温調モジュールの総称です。
I/Oモジュール	デジタルI/Oモジュール、アナログI/Oモジュールの総称です。
デジタルI/Oモジュール	デジタル入力モジュール、デジタル出力モジュール、デジタル入出力混合モジュールの総称です。
デジタル入力モジュール	入力端子を装備したデジタルI/Oモジュールの総称です。(FC6A-N****)
デジタル出力モジュール	出力端子を装備したデジタルI/Oモジュールの総称です。(FC6A-R***、FC6A-T****)
デジタル入出力混合モジュール	入力端子、出力端子を装備したデジタルI/Oモジュールの総称です。(FC6A-M****)
アナログI/Oモジュール	アナログ入力モジュール、アナログ出力モジュール、アナログ入出力混合モジュールの総称です。
アナログ入力モジュール	入力端子を装備したアナログI/Oモジュールの総称です。(FC6A-J***、FC6A-J4CN*、FC6A-J4CH**、FC6A-J8CU*)
アナログ出力モジュール	出力端子を装備したアナログI/Oモジュールの総称です。(FC6A-K****)
アナログ入出力混合モジュール	入力端子、出力端子を装備したアナログI/Oモジュールの総称です。(FC6A-L03CN*、FC6A-L06A*)
通信モジュール	シリアル通信モジュールの略称です。
シリアル通信モジュール	FC6A-SIF52、FC6A-SIF524の総称です。
温調モジュール	FC6A-F2M*、FC6A-F2MR*の総称です。
増設拡張モジュール	増設拡張モジュール一体型、増設拡張モジュール分離型マスター、増設拡張モジュール分離型スレーブの総称です。
増設拡張モジュール一体型	FC6A-EXM2、FC6A-EXM24の総称です。





項目	内容
増設拡張モジュール分離型マスター	FC6A-EXM1Mのことです。
増設拡張モジュール分離型スレーブ	FC6A-EXM1S、FC6A-EXM1S4の総称です。
HMIモジュール	FC6A-PH1のことです。
カートリッジベースモジュール	FC6A-HPH1のことです。
カートリッジ	I/Oカートリッジ、通信カートリッジの総称です。
I/Oカートリッジ	デジタルI/Oカートリッジ、アナログI/Oカートリッジの総称です。
デジタルI/Oカートリッジ	デジタル入力またはデジタル出力を拡張するI/Oカートリッジの総称です。 (FC6A-PN4、FC6A-PT*4)
アナログI/Oカートリッジ	アナログ入力またはアナログ出力を拡張するI/Oカートリッジの総称です。 (FC6A-PJ2A、FC6A-P*2**)
通信カートリッジ	RS232C通信カートリッジ、RS485通信カートリッジ、Bluetooth通信カートリッジの総称です。
RS232C通信カートリッジ	FC6A-PC1のことです。
RS485通信カートリッジ	FC6A-PC3のことです。
Bluetooth通信カートリッジ	FC6A-PC4のことです。
WindLDR	FC6A形のラダープログラム作成用のソフトウェアです。
USBケーブル	USBメンテナンスケーブル (HG9Z-XCM42)、USB-miniBポート用延長ケーブル (HG9Z-XCE21) の総称です。
ユーザープログラム	ラダープログラミングソフトウェアWindLDRで設定されるファンクション設定およびラダープログラムを一つにまとめたデータのことです。
ファンクション設定	コントローラの各種設定内容です。 [設定] タブ、モジュール構成エディタで設定する内容です。
ラダープログラム	メインプログラム、サブルーチンプログラム、ユーザー定義マクロの総称です。
メインプログラム	先頭行がラダープログラムのエントリーポイントであるプログラムです。エントリーポイントとは、ラダープログラムが実行される開始点のことです。ラダープログラムエディタの [メインプログラム] タブで作成します。
サブルーチンプログラム	次のいずれかのプログラムです。 ・メインプログラム内の LABEL 命令から LRET 命令までのプログラム ・WindLDR のサブルーチン機能で作成したプログラム (ラダープログラムエディタのタブでは、[# サブルーチン] (# : サブルーチンの番号) で表示されます)
ユーザー定義マクロ	WindLDRのユーザー定義マクロ機能で作成したプログラムです。 (ラダープログラムエディタのタブでは、[# ユーザー定義マクロ] (# : ユーザー定義マクロの番号) で表示されます)
ソースデバイス	演算の対象となるデバイス (演算命令を実行するためのデータの格納場所) のことです。
デスティネーションデバイス	演算結果を格納するデバイス (演算命令の実行結果のデータの格納場所) のことです。
デバイスの範囲外	デバイスタイプでサポートされているデバイスアドレスの範囲外の意味で使用しています。

WindLDR の名称

本文中の使用名称	WindLDR 操作手順
ファンクション設定	[設定] タブの [ファンクション設定] グループ
モニタ	[オンライン] タブの [モニタ] で [モニタ] から [モニタ開始] をクリック
PLC ステータス	[オンライン] タブの [PLC 本体] で [ステータス] をクリック
通信設定	[オンライン] タブの [通信] で [設定] をクリック
Modbus マスターリクエストテーブル	[設定] タブの [ファンクション設定] で [通信ポート] をクリックし、表示される [ファンクション設定] ダイアログボックス内で、[通信ポート] の [通信モード] から "Modbus RTUマスター" または "Modbus TCPクライアント" を選択
アプリケーションボタン	メニューバーの左側に表示されるボタン。クリックすると [新規]、[保存]、[名前を付けて保存] などのメニューや最近使ったプロジェクト、ならびに [WindLDR オプション] や [WindLDR の終了] ボタンを表示

本書で使う絵記号

本書では、説明を簡潔にするために次の絵記号を使用しています。

絵記号	意味
 警告	取り扱いを誤った場合、人が死亡または重傷を負う可能性がある項目について記載していることを示します。
 注意	取り扱いを誤った場合、人が傷害を負うか物的損害が発生する可能性がある項目について記載していることを示します。
	本製品を使用するにあたり守っていただきたいことや、操作するうえで誤りやすい事項について記載していることを示します。
	その項目に関する補足情報や覚えておくに役立つ情報を記載していることを示します。

目次

	安全上のご注意	序-1
	はじめに	序-2
	本書で使う略称、総称、用語	序-4
	本書で使う絵記号	序-6
第1章	概要	1-1
	概要	1-1
	通信機能の概要	1-9
第2章	デバイス	2-1
	デバイス一覧	2-1
	特殊内部リレー	2-4
	特殊データレジスタ	2-18
第3章	通信設定	3-1
	設定一覧	3-1
	通信ポートの設定	3-2
	ネットワークの設定	3-3
	ネットワーク管理	3-11
	コネクション設定	3-14
	リモートホストリスト	3-21
第4章	メンテナンス通信	4-1
	概要	4-1
	USBポートでのメンテナンス通信	4-3
	ポート1でのメンテナンス通信	4-4
	ポート1 (Bluetooth) でのメンテナンス通信	4-6
	Ethernetポート1および2でのメンテナンス通信	4-7
	ポート2～33でのメンテナンス通信	4-12
	ポート1～3 (Bluetooth) でのメンテナンス通信	4-15
	HMI-Ethernetポートでのメンテナンス通信	4-16
第5章	ユーザー通信	5-1
	概要	5-1
	TXD (ユーザー通信送信)	5-2
	RXD (ユーザー通信受信)	5-9
	ETXD (イーサネットユーザー通信送信)	5-22
	ERXD (イーサネットユーザー通信受信)	5-22
	シリアル通信でのユーザー通信	5-23
	シリアル通信 (Bluetooth) でのユーザー通信	5-38
	イーサネット通信でのユーザー通信	5-39
	ユーザー通信送信命令・受信命令のエラー	5-53
	ユーザー通信命令のキャラクタコード	5-54
	ユーザー通信を用いたプログラム例	5-55
第6章	Modbus通信	6-1
	概要	6-1
	シリアル通信による Modbus RTU 通信	6-2
	イーサネット通信による Modbus TCP 通信	6-19
	Modbus RTU パススルー機能	6-30
第7章	データリンク通信	7-1
	データリンク通信の概要	7-1
	ユニットの接続例	7-10
	弊社製品とのリンク	7-10
第8章	J1939通信	8-1
	CANによるJ1939通信の概要	8-1
	J1939通信の設定	8-7
第9章	Bluetooth通信	9-1
	概要	9-1
	Bluetooth通信カートリッジの設定	9-3
	Bluetooth機器との設定と通信	9-8

第10章	FTPサーバー/クライアント機能	10-1
	FTPサーバー機能.....	10-1
	FTPクライアント機能.....	10-8
第11章	PING送信機能	11-1
	PING命令 (Ping送信).....	11-1
第12章	EMAIL送信機能	12-1
	概要.....	12-1
	EMAIL命令 (Eメール送信).....	12-1
	Eメールアドレス帳.....	12-11
	Eメールエディタ.....	12-13
	添付ファイルエディタ.....	12-16
	履歴ファイル添付設定.....	12-19
第13章	Webサーバー	13-1
	概要.....	13-1
	対応機種および通信ポート.....	13-1
	動作環境.....	13-2
	基本仕様.....	13-3
	Plus CPU モジュールの Web サーバー.....	13-5
	[証明書および秘密鍵の一括設定] ウィザード.....	13-11
	認証局のルート証明書のインポート.....	13-17
	HMI モジュールの Web サーバー.....	13-20
	ユーザーアカウント設定.....	13-24
	システム Web ページ.....	13-27
	ユーザー Web ページ.....	13-31
	カスタム Web ページ.....	13-43
	CGI.....	13-55
	SD メモリカード読み出し (ファイルのダウンロード).....	13-57
	制限事項.....	13-57
第14章	通信モニタ	14-1
	概要.....	14-1
	通信モニタの開始と停止.....	14-1
第15章	BACnet/IP	15-1
	概要.....	15-1
	BACnet/IP で使用する通信ポート.....	15-1
	BACnet 仕様.....	15-2
	BACnet について.....	15-3
	機能.....	15-5
	BACnet/IP の動作.....	15-11
	BACnet/IP の設定.....	15-13
	オブジェクト.....	15-33
	主なプロパティ.....	15-41
第16章	EtherNet/IP通信	16-1
	概要.....	16-1
	EtherNet/IP で使用する通信ポート.....	16-2
	EtherNet/IP 仕様.....	16-3
	EtherNet/IP について.....	16-4
	機能.....	16-5
	EtherNet/IP 通信の動作.....	16-13
	EtherNet/IP 通信の設定.....	16-18
	[EtherNet/IP 設定] ダイアログボックス.....	16-19
	EtherNet/IP 通信の設定の流れ.....	16-32
	EtherNet/IP の設定例.....	16-35
	オブジェクト.....	16-41
第17章	MQTT通信	17-1
	概要.....	17-1
	対応機種および通信ポート.....	17-2
	基本仕様.....	17-3
	対応 MQTT ブローカーおよびクラウドサービス.....	17-3
	MQTT 通信の設定.....	17-5
	[MQTT 設定] ダイアログボックス.....	17-6
	MQTT 設定.....	17-6

汎用ブローカーへ接続する.....	17-7
AWS IoT Core へ接続する.....	17-12
Azure IoT Hub へ SAS を使って接続する.....	17-17
Azure IoT Hub へ X.509 証明書を使って接続する.....	17-22
Azure IoT Hub へ DPS 経由で接続する.....	17-26
パブリッシュ.....	17-33
サブスクライブ.....	17-38
トピックにパブリッシュする.....	17-51
トピックをサブスクライブする.....	17-59
第18章 MCプロトコル通信.....	18-1
概要.....	18-1
イーサネット通信による MC プロトコル通信.....	18-2
索引.....	索-1

第1章 概要

この章では、通信インターフェイスを備えた FC6A 形の概要について説明します。

概要

FC6A 形は、各通信インターフェイスを使用して、様々な通信ができます。

All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュール

- : 通信するためのポートとして使用できます
- × : 通信するためのポートとして使用できません
- : CPU モジュールに装備されていません

通信 / 機能		ポート					カートリッジスロット			増設モジュール	
		シリアルポート 1	USBポート	Ethernetポート 1	Ethernetポート 2	HMI-Ethernetポート	CANポート	1	2	3	通信モジュール
RS232C通信/ RS485通信	メンテナンス通信	○*1	○	○	—	○*2	—	○*4	○*4	—	○
	ユーザー通信										
	Modbus RTU通信 (マスター/スレーブ)		×	×							
	データリンク通信 (親局/子局)										
Bluetooth 通信	ユーザー通信										
イーサネット 通信	メンテナンス通信 (サーバー)	×		○	—	○*2	×	×	×	—	×
	ユーザー通信 (サーバー/クライアント)										
	ユーザー通信 (UDP)		×	×							
	Modbus TCP通信 (サーバー/クライアント)		×	○		○*2					
	MCプロトコル通信 (クライアント)										
	BACnet/IP通信										
	EtherNet/IP通信			×		×					
	MQTT通信										
	FTPサーバー/ クライアント機能										
	Webサーバー機能					○*2					
	PING送信			○		×					
EMAIL送信		×	○*2								
CAN通信	J1939通信		×		×	○*3					

*1 All-in-One CPU モジュールのみ、シリアルポート 1 を使用して各通信ができます。
 *2 CPU モジュールに HMI モジュールを接続し、HMI-Ethernet ポートを増設して通信できます。
 *3 CAN J1939 All-in-One CPU モジュールのみ、CAN ポートを使用して J1939 通信ができます。
 *4 カートリッジスロットに通信カートリッジを接続して通信できます。

Plus CPU モジュール

- : 通信するためのポートとして使用できます
- × : 通信するためのポートとして使用できません
- : CPU モジュールに装備されていません

通信機能	ポート						カートリッジ スロット			増設 モジュール	
	シリアル ポート 1	USB ポート	Ethernet ポート 1	Ethernet ポート 2	HMI- Ethernet ポート	CAN ポート	1	2	3	通信 モジュール	
メンテナンス通信		○	○	○	○*1						
RS232C通信/ RS485通信	ユーザー通信										
	Modbus RTU通信 (マスター/スレーブ)		×	×	×		○*2	○*2	○*3	○	
	データリンク通信 (親局/子局)										
Bluetooth 通信											
イーサネット 通信	メンテナンス通信 (サーバー)	—	×	○	○	—	×	×	×	×	
	ユーザー通信 (サーバー/クライアント)										○*1
	ユーザー通信 (UDP)										
	Modbus TCP通信 (サーバー/クライアント)										
	MCプロトコル通信 (クライアント)										
	BACnet/IP通信										○
	EtherNet/IP通信										×
	MQTT通信										
	FTPサーバー/ クライアント機能										○
	Webサーバー機能										○*1
	PING送信										○
	EMAIL送信										○*1
CAN通信			×	×	×						
J1939通信			×	×	×						

*1 CPU モジュールに HMI モジュールを接続し、HMI-Ethernet ポートを増設して通信できます。

*2 CPU モジュールにカートリッジベースモジュールを接続し、カートリッジスロットに通信カートリッジを接続して通信できます。

*3 CPU モジュールに HMI モジュールを接続し、カートリッジスロットに通信カートリッジを接続して通信できます。

通信インターフェイス

各通信インターフェイスの概要および仕様は、次のとおりです。

USB ポート

本ポートを使用してパソコンと接続し、メンテナンス通信ができます。

通信タイプ	USB2.0 Full speed, CDC class
通信機能	パソコンとのメンテナンス通信、USB電源でプログラムダウンロードが可能
コネクタ	USB mini-B
内部回路との絶縁	非絶縁

シリアルポート 1

本ポートを使用してパソコンやプログラマブル表示器、プリンタなどの RS232C/RS485 通信に対応した外部機器と通信できます。メンテナンス通信、ユーザー通信、Modbus RTU 通信（マスター/スレーブ）、データリンク通信（親局/子局）ができます。シリアルポート 1 は All-in-One CPU モジュールのみ装備しています。

通信タイプ	EIA RS232CまたはRS485をソフト選択
最大通信速度	115200bps
通信機能	メンテナンス通信、ユーザー通信、Modbus RTU通信（マスター/スレーブ）、データリンク通信（親局/子局）
コネクタ	RJ45
ケーブル	RS232C：シールド付き多芯ケーブル RS485：シールド付きツイストペアケーブル
最大ケーブル長	RS232C：5m RS485：200m
内部回路との絶縁	非絶縁

シリアル通信モジュール

シリアル通信モジュールは、CPU モジュールの基本増設側または拡張増設側に接続して使用します。

本ポートを使用してパソコンやプログラマブル表示器、プリンタなど RS232C/RS485 通信に対応した外部機器と通信できます。メンテナンス通信、ユーザー通信、Modbus RTU 通信（マスター/スレーブ）、データリンク通信（親局/子局）ができます。

形番	FC6A-SIF52	
点数	2	
電気的特性	EIA RS232C/EIA RS485*1	
最大通信速度	115200bps	
通信機能	メンテナンス通信、ユーザー通信、Modbus RTU通信（マスター/スレーブ）、データリンク通信（親局/子局）	
最大ケーブル長	15m (EIA RS232C) / 1200m (EIA RS485)	
内部回路との絶縁	通信ポート-内部回路間	フォトカプラ
	ポート間	トランス
ケーブル	推奨ケーブル	RS232C：シールド付き多芯ケーブル 24AWG RS485：シールド付きツイストペアケーブル 24AWG

*1 WindLDR の [ファンクション設定] で指定します。「データビット：7ビット」かつ「パリティ：なし」の設定はできません。

RS232C 通信カートリッジ、RS485 通信カートリッジ

RS232C 通信カートリッジおよび RS485 通信カートリッジは、次のカートリッジスロットに接続して使用します。

- All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールのカートリッジスロット 1 および 2
- Plus CPU モジュールに増設したカートリッジベースモジュールのカートリッジスロット 1 および 2
- Plus CPU モジュールに増設した HMI モジュールのカートリッジスロット 3

本ポートを使用してパソコンやプログラマブル表示器、プリンタなど RS232C/RS485 通信に対応した外部機器と通信できます。メンテナンス通信、ユーザー通信、Modbus RTU 通信（マスター/スレーブ）、データリンク通信（親局/子局）ができます。

形番	FC6A-PC1		FC6A-PC3
電気的特性	EIA RS232C		EIA RS485
最大通信速度	115200bps		115200bps
通信機能	メンテナンス通信、ユーザー通信、Modbus RTU通信（マスター/スレーブ）、データリンク通信（親局/子局）		
最大ケーブル長	5m	200m	
内部回路との絶縁	非絶縁		非絶縁
ケーブル	推奨ケーブル	シールド付き多芯ケーブル：24AWG	シールド付きツイストペアケーブル：24AWG

Bluetooth 通信カートリッジ

Bluetooth 通信カートリッジは、次のカートリッジスロットに接続した場合のみ使用できます。

- All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールのカートリッジスロット 1 および 2
- Plus CPU モジュールに増設したカートリッジベースモジュールのカートリッジスロット 1 および 2
- Plus CPU モジュールに増設した HMI モジュールのカートリッジスロット 3

Bluetooth を使用してパソコンやスマートフォン、バーコードリーダーなど Bluetooth 通信に対応した外部機器と通信できます。メンテナンス通信およびユーザー通信ができます。

形番	FC6A-PC4	
電気的特性	Bluetooth ver.2.1 + EDR	
プロファイル	SPP(Serial Port Profile) iAP(iPod Accessory Protocol)	
通信機能	メンテナンス通信、ユーザー通信	
伝送距離	10m (Class 2)	

Ethernet ポート 1 および 2

本ポートは次の通信および機能に対応しています。本ポートは複数のコネクションを持ち、各コネクションで異なる通信プロトコルを同時に使用できます。詳細は「第3章 コネクション設定」を参照してください。

- ：通信するためのポートとして使用できます
- ×：通信するためのポートとして使用できません
- －：CPU モジュールに装備されていません

通信 / 機能		All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュール		Plus CPU モジュール																	
		Ethernet ポート 1	Ethernet ポート 2	Ethernet ポート 1	Ethernet ポート 2																
イーサネット 通信	メンテナンス通信 (サーバー)	○	－	○	○																
	ユーザー通信 (サーバー/クライアント)																				
	ユーザー通信 (UDP)																				
	Modbus TCP通信 (サーバー/クライアント)	×				○	○														
	MCプロトコル通信 (クライアント)	×						×	×												
	BACnet/IP通信									○	○										
	EtherNet/IP通信											×	○								
	MQTT通信													○	×						
	FTPサーバー/ クライアント機能															○	×				
	Webサーバー機能																	○	○		
	PING送信																			×	×
	EMAIL送信																				

通信タイプ	IEEE802.3 規格準拠
通信速度	10BASE-T、100BASE-TX
コネクション数	All-in-One CPUモジュール/CAN J1939 All-in-One CPUモジュール：8個 Plus CPUモジュール：16個
通信機能	メンテナンス通信 (サーバー)、ユーザー通信 (サーバー/クライアント)、 ユーザー通信 (UDP)、Modbus TCP通信 (サーバー/クライアント)、MCプロトコル通信 (クライアント)
Web サーバー機能	あり
Web データ保存先	FROM
Web データ最大サイズ*1	5.0MB
EMAIL 送信機能	あり
コネクタ	RJ45
ケーブル	CAT5 以上のSTP
最大ケーブル長	100m
内部回路との絶縁	パルストランス絶縁

*1 Web データ最大サイズの内、ユーザー使用可能領域はシステム Web ページと Web ページエディタを使用する条件によって、以下のように異なります。

		システム Web ページ	
		使用する	使用しない
Webページエディタ	使用する	2.5MB	3.0MB
	使用しない	4.5MB	5.0MB

CAN ポート

CAN J1939 All-in-One CPU モジュールは、本ポートを使用して J1939 通信ができます。

通信タイプ	CAN
通信速度	250kbps
通信機能	J1939通信
コネクタ	FC6A-PMTE05PN02
ケーブル	SAE J1939-11 : シールド付きツイストペアケーブル SAE J1939-15 : シールド無しツイストペアケーブル
最大ケーブル長	SAE J1939-11 : 40m、スタブ最大1m SAE J1939-15 : 40m、スタブ最大3m
終端抵抗	120Ω (0.5W以上)
内部回路との絶縁	電源部：トランス絶縁 信号部：ガルバニック絶縁, フォトカプラ絶縁

HMI-Ethernet ポート

HMI-Ethernet ポートは、CPU モジュールと HMI モジュールを接続した場合のみ使用できます。

本ポートを使用してパソコンやプログラマブル表示器などのイーサネット通信に対応した外部機器と通信できます。

本ポートは、イーサネット通信で使用できる 8 つのコネクションを持っています。各コネクションには、メンテナンス通信（サーバー）を設定することができます。

また、本ポートは Web サーバー機能、EMAIL 送信機能に対応しています。

通信タイプ	IEEE802.3 規格準拠
通信速度	10BASE-T、100BASE-TX
コネクション数	8個
通信機能	メンテナンス通信
Web サーバー機能	あり
Web データ保存先	FROM
Web データ最大サイズ*1	5.0MB
EMAIL 送信機能	あり
コネクタ	RJ45
ケーブル	CAT 5. STP以上
最大ケーブル長	100m
内部回路との絶縁	パルストランス絶縁

*1 Web データ最大サイズの内、ユーザー使用可能領域はシステム Web ページと Web ページエディタを使用する条件によって、以下のようになります。

		システム Web ページ	
		使用する	使用しない
Webページエディタ	使用する	2.5MB	3.0MB
	使用しない	4.5MB	5.0MB



注意

- インターネットに FC6A 形を接続する際は、不正アクセス防止等のネットワークに対する十分なセキュリティ対策が必要となります。必ずネットワーク管理者、インターネットサービスプロバイダなどにご相談ください。インターネットでの通信で生じるセキュリティ上の損害や問題について、当社は一切の責任を負いません。
- セキュリティ対策として、必ずファイアウォールなどを使用して、接続可能な IP アドレスやポートを制限してください。

CPU モジュールと通信インターフェイス一覧

各 CPU モジュールに装備されている、または増設できる通信インターフェイスは、次のとおりです。

各モジュールにおける通信インターフェイスの位置は、FC6A 形マイクロスマート ユーザーズ マニュアル「第2章 製品仕様」を参照してください。

形番	USB ポート	シリアル ポート 1	通信モジュール ポート 4～33 ^{*1}	通信 カートリッジ	Ethernet ポート 1	Ethernet ポート 2	CAN ポート	HMI-Ethernet ポート ^{*4}
FC6A-C16 ^{***E} FC6A-C24 ^{***E}	1	1	最大 6	最大 1 ^{*2}	1	—	—	最大 1
FC6A-C40 ^{***E}				最大 2 ^{*2}				
FC6A-C40 ^{***EJ}	1	—	最大 6	最大 2 ^{*2}	1	—	1	最大 1
FC6A-D ^{***CEE}			最大 30	最大 3 ^{*3}		1	—	

*1 シリアル通信モジュールは、CPU モジュールの基本増設側または拡張増設側に接続した場合のみ使用できます。

CPU モジュールに近い方から通信ポート 4、5・・・と順に割り付きます。

*2 All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールの場合、カートリッジスロット 1 および 2 に接続すると使用できません。

*3 Plus CPU モジュールの場合、カートリッジスロット 1、2 および 3 に接続すると使用できます。

*4 HMI-Ethernet ポートは、HMI モジュールで使用できます。

通信ポート番号の割り付け

All-in-One CPU モジュール、CAN J1939 All-in-One CPU モジュールおよび Plus CPU モジュールは相手機器とシリアル通信できます。

各 CPU モジュールは、標準装備されたインターフェイス、またはインターフェイスを増設してシリアル通信できます。シリアル通信できるインターフェイスは次のとおりです。

- All-in-One CPU モジュール : シリアルポート 1、カートリッジスロット 1 および 2、シリアル通信モジュールのポート
- CAN J1939 All-in-One CPU モジュール : カートリッジスロット 1 および 2、シリアル通信モジュールのポート
- Plus CPU モジュール : カートリッジスロット 1～3、シリアル通信モジュールのポート

各インターフェイスを通信ポートとして使用するには、通信相手機器の通信仕様に合わせて通信フォーマットを設定する必要があります。

[ファンクション設定] の [通信ポート] で、通信フォーマットを設定してください。

各インターフェイスと通信ポート番号の対応は、次のとおりです。

設定の詳細は、「第 3 章 通信ポートの設定」(3-2 頁) を参照してください。

■CPU モジュールおよび HMI モジュールのポート、カートリッジスロット 1～3

- : CPU モジュールに装備されていません
- × : シリアル通信するためのポートとして使用できません

CPU モジュール		ポート						カートリッジスロット		
		シリアルポート 1	USBポート	Ethernetポート 1	Ethernetポート 2	HMI-Ethernetポート	CANポート	1	2	3
All-in-One CPUモジュール	16点タイプ	通信ポート 1	—	—	—	—	—	通信ポート 2 *2*5	—	—
	24点タイプ								通信ポート 3 *2*5	
	40点タイプ									
CAN J1939 All-in-One CPUモジュール	40点タイプ	×	×	—	×	×	通信ポート 1 *3*5	通信ポート 2 *3*5	通信ポート 3 *4*5	
Plus CPUモジュール	Plus16点タイプ	—	—	×	—	—	—	通信ポート 1 *3*5	通信ポート 2 *3*5	通信ポート 3 *4*5
	Plus32点タイプ									

*1 CPU モジュールに HMI モジュールを接続し、HMI-Ethernet ポートを増設した場合

*2 カートリッジスロットに通信カートリッジを接続した場合

*3 CPU モジュールにカートリッジベースモジュールを接続し、カートリッジスロットに通信カートリッジを接続した場合

*4 CPU モジュールに HMI モジュールを接続し、カートリッジスロットに通信カートリッジを接続した場合

*5 「データビット：7ビット」かつ「パリティ：なし」の設定はできません。



シリアルポート 1、カートリッジスロット 1 および 2 の位置、シリアルポート 1、通信モジュール、通信カートリッジの配線は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 2 章 製品仕様」を参照してください。

■シリアル通信モジュールのポート

CPU モジュール		シリアル通信モジュール					
		1 台目	2 台目	3 台目	4 台目	...	15 台目
All-in-One CPUモジュール	16点タイプ	通信ポート 4, 5	通信ポート 6, 7	通信ポート 8, 9	*1	*1	*1
	24点タイプ						
	40点タイプ						
CAN J1939 All-in-One CPUモジュール	40点タイプ	通信ポート 4, 5	通信ポート 6, 7	通信ポート 8, 9	*1	...	通信ポート 32, 33
Plus CPUモジュール	Plus16点タイプ	通信ポート 4, 5	通信ポート 6, 7	通信ポート 8, 9	*1	...	通信ポート 32, 33
	Plus32点タイプ						

*1 All-in-One CPU モジュール、CAN J1939 All-in-One CPU モジュールにはシリアル通信モジュールを 4 台以上接続できません。

通信機能の概要

FC6A 形は、メンテナンス通信、ユーザー通信、Modbus 通信、データリンク通信、J1939 通信、BACnet 通信、MQTT 通信の通信機能に対応しています。

各通信機能の概要と接続例について説明します。

メンテナンス通信

メンテナンス通信により、パソコンにインストールされたプログラミングソフトウェア WindLDR を使用して FC6A 形の運転状態、入出力状態の確認、デバイス値のモニタや変更、ユーザープログラムのダウンロード/アップロードができます。メンテナンス通信の詳細は、「第4章 メンテナンス通信」(4-1 頁)を参照してください。

使用可能なインターフェイス *1

USB ポート	シリアルポート 1	Ethernet ポート 1、2 および HMI-Ethernet ポート *2	通信カートリッジ および 通信モジュール	CAN ポート
○	○	○	○	×

*1 通信インターフェイスによって、使用できるメンテナンス通信の機能に違いがあります。詳細は、「第4章 メンテナンス通信」(4-1 頁)を参照してください。

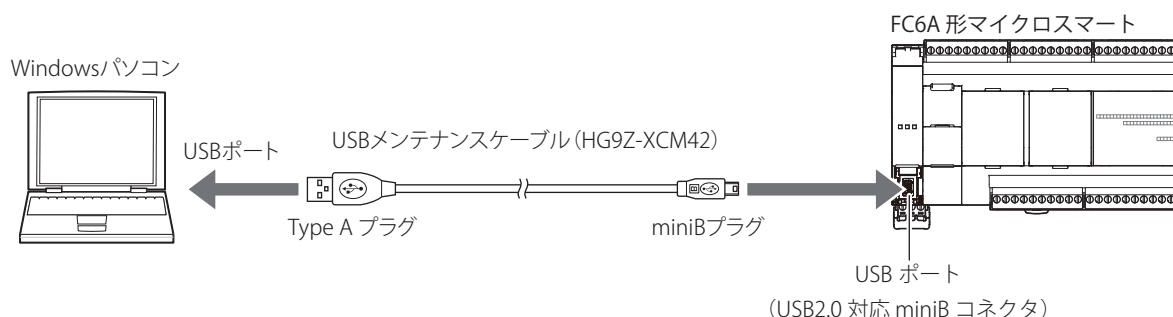
*2 HMI-Ethernet ポートはメンテナンス通信のみ使用できます。



HMI モジュールを接続している場合は、HMI-Ethernet ポートを使用して、メンテナンス通信を行うことができます。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第7章 HMI 機能」を参照してください。

■USB ポートを使用した 1:1 メンテナンス通信の例

パソコンと FC6A 形を USB ケーブルで接続する例です。



■シリアルポート 1 を使用した IDEC 製プログラマブル表示器との 1:1 メンテナンス通信例

IDEC 製プログラマブル表示器と FC6A 形を接続する例です。FC6A 形のシリアルポート 1 とプログラマブル表示器を表示器接続用ケーブル (FC6A-KC2C) を使用して接続します。

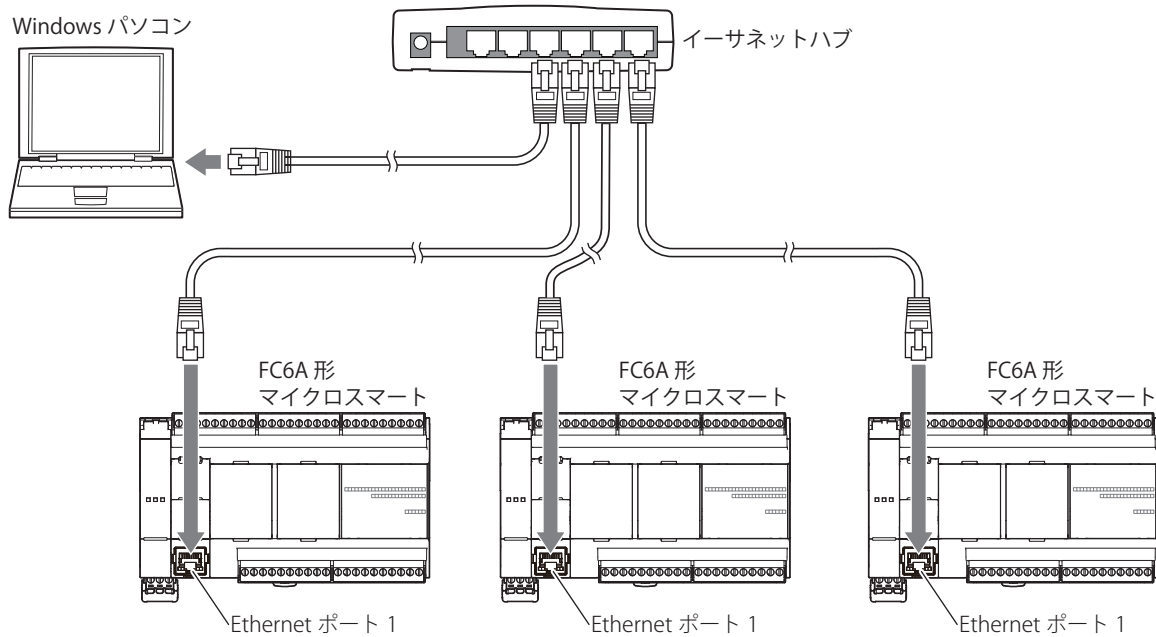
通信設定などの詳細は、WindO/I-NV2 ユーザーズ マニュアル、WindO/I-NV3 ユーザーズ マニュアルおよび WindO/I-NV4 ユーザーズ マニュアルを参照してください。



*1 表示器接続用ケーブルの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「付録 各種ケーブル」を参照してください。

■Ethernet ポート 1 を使用した 1 : N メンテナンス通信の例

パソコンと FC6A 形をイーサネット接続する例です。FC6A 形の Ethernet ポート 1 にイーサネットケーブルを接続して、イーサネットハブを使用してパソコンと接続します。



ユーザー通信

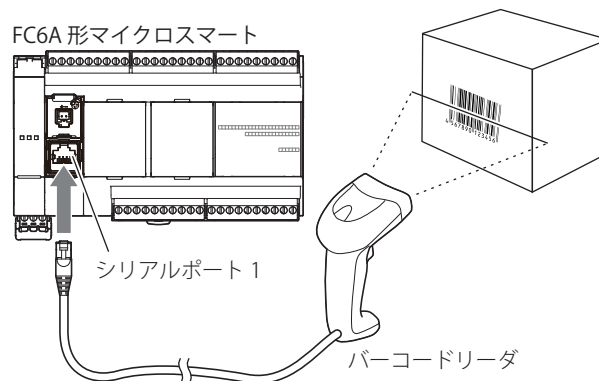
ユーザー通信により、パソコン、プリンタ、バーコードリーダなどの外部機器を制御できます。ユーザー通信の詳細は、「第 5 章 ユーザー通信」(5-1 頁) を参照してください。

使用可能なインターフェイス

USB ポート	シリアルポート 1	Ethernet ポート 1、2	通信カートリッジ および 通信モジュール	CAN ポート
×	○	○	○	×

■シリアルポート 1 を使用したユーザー通信の例

バーコードリーダで読み取ったデータを FC6A 形で受信する例です。FC6A 形のシリアルポート 1 とバーコードリーダを接続します。



Modbus 通信

FC6A 形のシリアル通信は、Modbus RTU 通信プロトコルに対応しており、Modbus RTU 通信のマスターやスレーブとして使用できます。Modbus RTU 通信により、インバータや温度調節器のデータのモニタや変更ができます。

また、Ethernet ポート 1 および 2 では Modbus TCP 通信プロトコルに対応しています。

Modbus 通信の詳細は、「第 6 章 Modbus 通信」(6-1 頁)を参照してください。

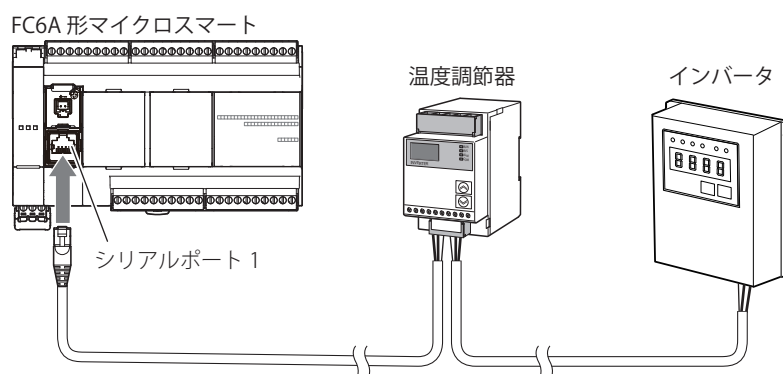
使用可能なインターフェイス

USB ポート	シリアルポート 1	Ethernet ポート 1、2	通信カートリッジ*1 および 通信モジュール	CAN ポート
×	○	○	○	×

*1 RS232C 通信カートリッジおよび RS485 通信カートリッジのみ対応しています。

■シリアルポート 1 を使用した Modbus RTU 通信の例

Modbus RTU 通信対応の温度調節器やインバータと通信する例です。FC6A 形のシリアルポート 1 と温度調節器を接続します。



データリンク通信

FC6A 形は、データリンク通信に対応しており、シリアルポート 1 や通信カートリッジを使用して、CPU モジュール間でデータの共有が行えます。また、FC6A 形は FC5A 形や FC4A 形の CPU モジュールともデータの共有が行えます。WindLDR で設定を行い、最大 31 台の子局とデータの共有が行えます。

データリンク通信の詳細は、「第 7 章 データリンク通信」(7-1 頁)を参照してください。

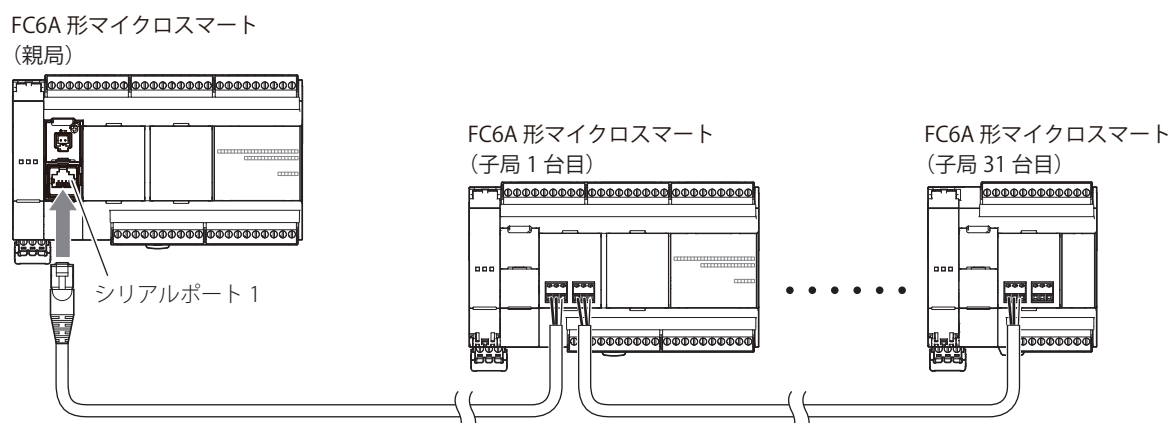
使用可能なインターフェイス

USB ポート	シリアルポート 1	Ethernet ポート 1、2	通信カートリッジ*1 および 通信モジュール	CAN ポート
×	○	×	○	×

*1 RS232C 通信カートリッジおよび RS485 通信カートリッジのみ対応しています。

■シリアルポート 1 を使用したデータリンク通信の例

FC6A 形を親局として、複数の CPU モジュールと通信する例です。FC6A 形のシリアルポート 1 と子局となる CPU モジュールを接続します。



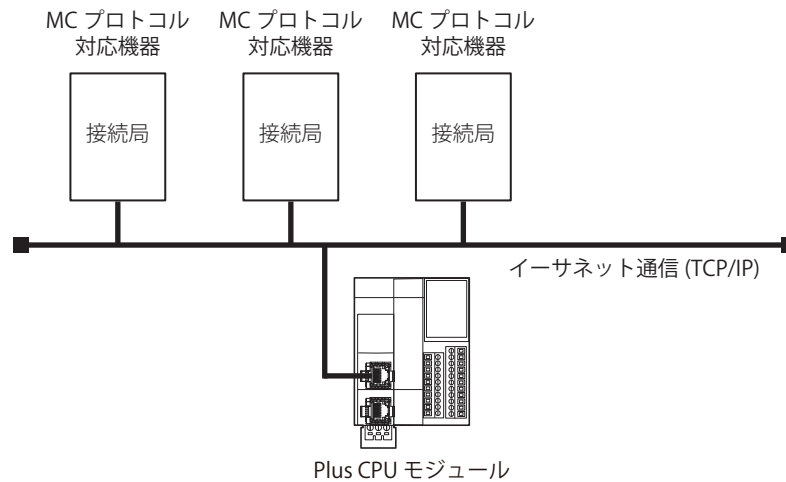
MC プロトコル通信

Plus CPU モジュールは、Ethernet ポート 1 または Ethernet ポート 2 を使用して、MC プロトコル通信におけるクライアントとして MC プロトコル対応機器のデバイス値を読み出しおよび書き込みできます。

MC プロトコル通信の詳細は、「第 18 章 MC プロトコル通信」(18-1 頁)を参照してください。

USB ポート	シリアルポート 1	Ethernet ポート 1	Ethernet ポート 2	通信カートリッジ および 通信モジュール	CAN ポート
×	×	○	○	×	×

■Ethernet ポート 1 を使用した BACnet 通信の例



BACnet 通信

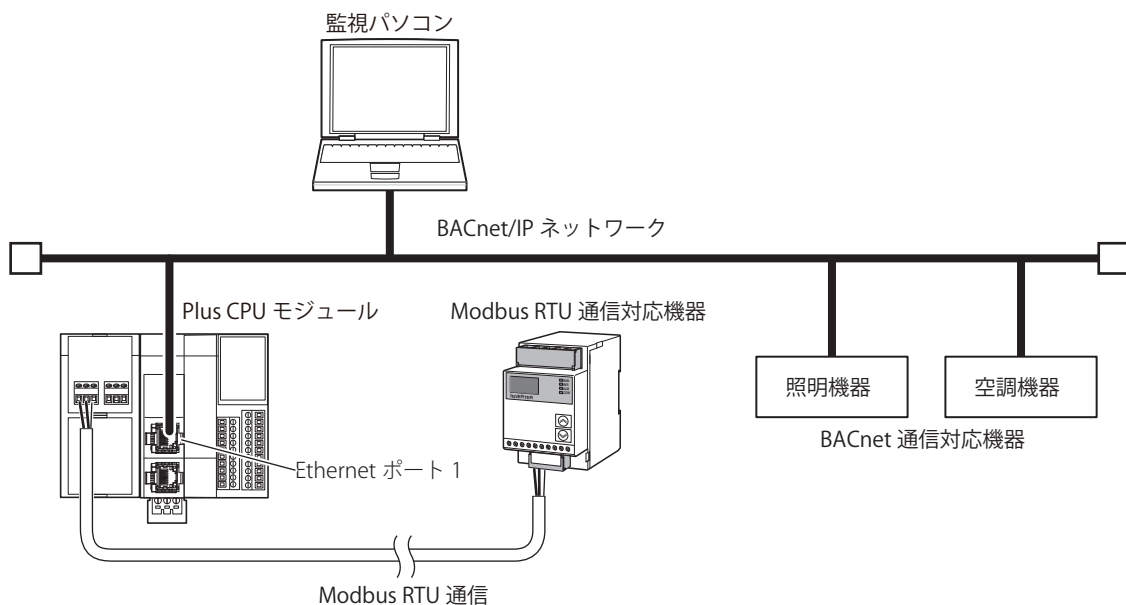
Plus CPU モジュールは、Ethernet ポート 1 を使用して BACnet/IP ネットワークへ接続し、他の BACnet 通信対応機器と通信できます。BACnet 通信の詳細は、「第 15 章 BACnet/IP」(15-1 頁)を参照してください。

使用可能なインターフェイス

USB ポート	シリアルポート 1	Ethernet ポート 1	Ethernet ポート 2	通信カートリッジ および 通信モジュール	CAN ポート
×	×	○	×	×	×

■Ethernet ポート 1 を使用した BACnet 通信の例

Plus CPU モジュールが Modbus RTU 通信対応機器の情報を集約し、BACnet/IP ネットワークに接続された BACnet 通信対応機器と通信して、その情報を公開する例です。



EtherNet/IP 通信

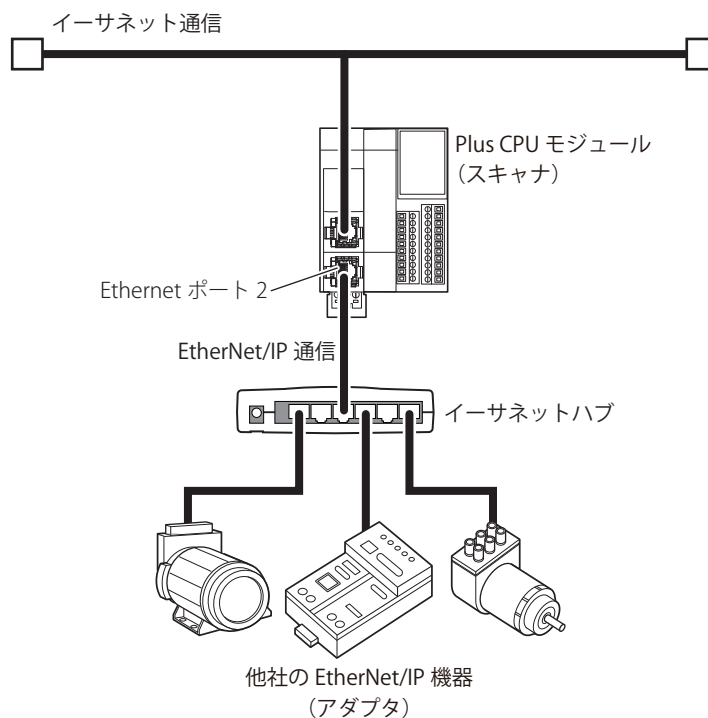
Plus CPU モジュールは、Ethernet ポート 2 を使用して既存のイーサネットネットワークを利用し、他の EtherNet/IP 通信対応機器と通信できます。また、EtherNet/IP 通信は標準のイーサネット技術を使用しているため、イーサネットに対応した様々な機器を混在させてネットワークを構築できます。EtherNet/IP 通信の詳細は、「第 16 章 EtherNet/IP 通信」(16-1 頁)を参照してください。

使用可能なインターフェイス

USB ポート	シリアルポート 1	Ethernet ポート 1	Ethernet ポート 2	通信カートリッジ および 通信モジュール	CAN ポート
×	×	×	○	×	×

■Ethernet ポート 2 を使用した EtherNet/IP 通信の例

Plus CPU モジュールが EtherNet/IP 通信対応機器と通信して、各機器を制御する例です。



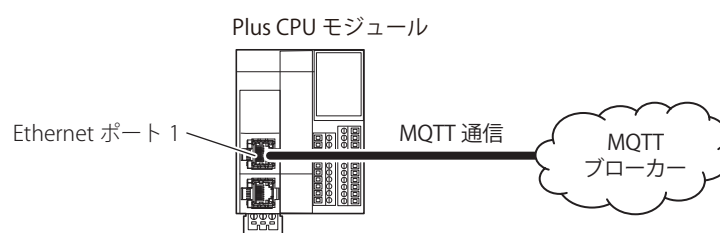
MQTT 通信

Plus CPU モジュールは、Ethernet ポート 1 を使用して、MQTT 通信におけるクライアント (パブリッシャーおよびサブスクリバ) としてブローカーと通信できます。MQTT 通信の詳細は、「第 17 章 MQTT 通信」(17-1 頁)を参照してください。

使用可能なインターフェイス

USB ポート	シリアルポート 1	Ethernet ポート 1	Ethernet ポート 2	通信カートリッジ および 通信モジュール	CAN ポート
×	×	○	×	×	×

■Ethernet ポート 1 を使用した MQTT 通信の例



J1939 通信

CAN J1939 All-in-One CPU モジュールは、CAN ポートを使用して J1939 通信ネットワークへ接続し、他の J1939 通信対応機器と通信できます。SAE J1939 規格に則したメッセージの送受信ができます。

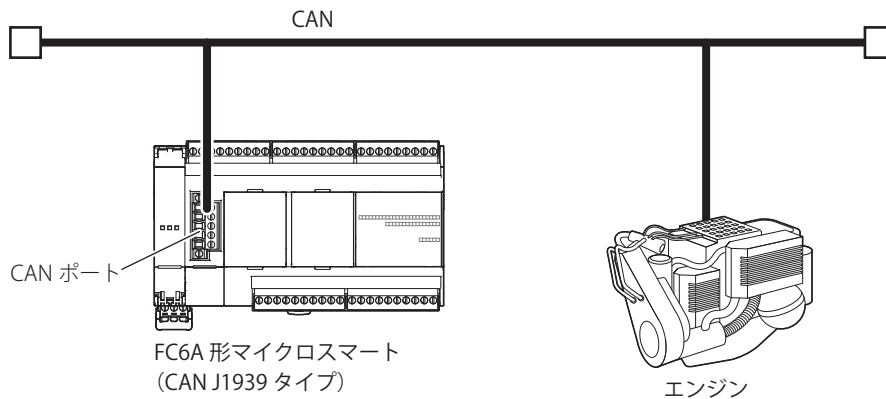
J1939 通信の詳細は、「第 8 章 J1939 通信」(8-1 頁)を参照してください。

使用可能なインターフェイス

USB ポート	シリアルポート 1	Ethernet ポート 1、2	通信カートリッジ	CAN ポート
×	×	×	×	○

■CAN ポートを使用した J1939 通信の例

FC6A 形を J1939 通信対応のエンジンと通信する例です。CAN J1939 All-in-One CPU モジュールの CAN ポートとエンジンを接続します。



第2章 デバイス

この章では、FC6A形の各種デバイスの割り付け、特殊内部リレーおよび特殊データレジスタの割り付けについて説明します。通信機能を使用したユーザープログラムの入力および編集を行う際の参考に使用してください。



FC6A形のユーザープログラムの入力および操作には、専門の知識が必要です。本書の内容やプログラムについて十分理解したうえで、FC6A形をご使用ください。

デバイス一覧

All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュール

デバイス名	記号	単位	範囲 (点数)		
			16点タイプ	24点タイプ	40点タイプ
入力*1	I	ビット	I0 ~ I10 (9点)	I0 ~ I15 (14点)	I0 ~ I27 (24点)
増設入力*1	I	ビット	I30 ~ I187 (128点) I190 ~ I507*2 (256点) I630 ~ I633*4 (4点)	I30 ~ I307 (224点) I310 ~ I627*3 (256点) I630 ~ I633*4 (4点)	I30 ~ I307 (224点) I310 ~ I627*3 (256点) I630 ~ I637*4 (8点)
出力*1	Q	ビット	Q0 ~ Q6 (7点)	Q0 ~ Q11 (10点)	Q0 ~ Q17 (16点)
増設出力*1	Q	ビット	Q30 ~ Q187 (128点) Q190 ~ Q507*2 (256点) Q630 ~ Q633*4 (4点)	Q30 ~ Q307 (224点) Q310 ~ Q627*3 (256点) Q630 ~ Q633*4 (4点)	Q30 ~ Q307 (224点) Q310 ~ Q627*3 (256点) Q630 ~ Q637*4 (8点)
内部リレー*1	M	ビット	M0 ~ M7997 (6400点) M10000 ~ M17497 (6000点)		
特殊内部リレー*1	M	ビット	M8000 ~ M8317 (256点)		
シフトレジスタ	R	ビット	R0 ~ R255 (256点)		
タイマ	T	ビット/ワード	T0 ~ T1023 (1024点)		
カウンタ	C	ビット/ワード	C0 ~ C511 (512点)		
データレジスタ	D	ビット/ワード	D0000 ~ D7999 (8000点) D10000 ~ D55999 (46000点)		
特殊データレジスタ	D	ビット/ワード	D8000 ~ D8499 (500点)		
間接指定レジスタ*5	P	2ワード	P0 ~ P15 (16点)		

*1 デバイスアドレスの下1桁は、0～7の8進数です。

*2 I190～I507およびQ190～Q507は、増設拡張モジュール一体型を使用して増設モジュール（増設拡張側）を接続した場合に使用可能なデバイスです。

*3 I310～I627およびQ310～Q627は、増設拡張モジュール一体型を使用して増設モジュール（増設拡張側）を接続した場合に使用可能なデバイスです。

*4 I630～I637およびQ630～Q637は、I/Oカートリッジを接続した場合に使用可能なデバイスです。

*5 使用可能なデータタイプはL（ロング）のみです。

Plus CPU モジュール

デバイス名	記号	単位	範囲 (点数)	
			Plus16 点タイプ	Plus32 点タイプ
入力*1	I	ビット	I0 ~ I7 (8 点)	I0 ~ I17 (16 点)
増設入力*1	I	ビット	I30 ~ I307 (224 点) I310 ~ I627*2 (256 点) I630 ~ I643*3 (12 点) I1000 ~ I10597*4 (2016 点)	
出力*1	Q	ビット	Q0 ~ Q7 (8 点)	Q0 ~ Q17 (16 点)
増設出力*1	Q	ビット	Q30 ~ Q307 (224 点) Q310 ~ Q627*2 (256 点) Q630 ~ Q643*3 (12 点) Q1000 ~ Q10597*4 (2016 点)	
内部リレー *1	M	ビット	M0 ~ M7997 (6400 点) M10000 ~ M21247 (9000 点)	
特殊内部リレー *1	M	ビット	M8000 ~ M9997 (1600 点)	
シフトレジスタ	R	ビット	R0 ~ R255 (256 点)	
タイマ	T	ビット/ワード	T0 ~ T1999 (2000 点)	
カウンタ	C	ビット/ワード	C0 ~ C511 (512 点)	
データレジスタ	D	ビット/ワード	D0000 ~ D7999 (8000 点) D10000 ~ D61999 (52000 点)	
非保持データレジスタ	D	ビット*/ワード	D70000 ~ D269999*5 (200000 点)	
特殊データレジスタ	D	ビット/ワード	D8000 ~ D8899 (900 点)	
間接指定レジスタ*6	P	2ワード	P0 ~ P15 (16 点)	

*1 デバイスアドレスの下1桁は、0～7の8進数です。

*2 I310～I627およびQ310～Q627は、増設拡張モジュール一体型を使用して増設モジュール（増設拡張側）を接続した場合に使用可能なデバイスです。（ノード0）

*3 I630～I643およびQ630～Q643は、I/Oカートリッジを接続した場合に使用可能なデバイスです。

*4 I1000～I10597およびQ1000～Q10597は、増設拡張モジュール分離型マスター/分離型スレーブおよび増設拡張モジュール一体型を使用して増設モジュールを接続した場合に使用可能なデバイスです。（ノード1～10）

*5 D70000～D269999はキープ指定できません。STOP→RUNでは保持しますが、電源投入時は0にリセットします。

*6 使用可能なデータタイプはL（ロング）のみです。

*7 SCRPT命令で実行するスクリプト内とUMACRO命令で使用する引数デバイスのみ使用可能です。

デバイス名と記号について

- **入力 (I)、増設入力 (I)**
外部機器からの ON/OFF 情報を FC6A 形に入力するデバイスです。
- **出力 (Q)、増設出力 (Q)**
FC6A 形からの ON/OFF 情報を外部機器へ出力するデバイスです。
- **内部リレー (M)**
FC6A 形内部で使用するビット単位のデバイスです。
- **特殊内部リレー (M)**
FC6A 形内部で使用するビット単位のデバイスです。それぞれのビットに特殊な機能が割り当てられています。
- **シフトレジスタ (R)**
SFR 命令および SFRN 命令で使用するビット単位のデバイスです。パルス入力にしたがってデータのビット列をシフトします。
- **タイマ (T)**
FC6A 形内部で使用するタイマです。タイマビット (記号：T、単位：ビット)、タイマ設定値 (記号：TP、単位：ワード)、タイマ現在値 (記号：TC、単位：ワード) の3つのデバイスがあります。
オンディレータイマ、オフディレータイマとして使用できます。タイマ (T) の詳細は、ラダープログラミングマニュアル「第3章 ソースデバイスまたはデスティネーションデバイスにタイマ / カウンタを指定した場合の表記」を参照してください。
- **カウンタ (C)**
FC6A 形内部で使用するカウンタです。カウンタビット (記号：C、単位：ビット)、カウンタ設定値 (記号：CP、単位：ワード)、カウンタ現在値 (記号：CC、単位：ワード) の3つのデバイスがあります。加算式カウンタ、可逆カウンタとして使用できます。
カウンタ (C) の詳細は、ラダープログラミングマニュアル「第3章 ソースデバイスまたはデスティネーションデバイスにタイマ / カウンタを指定した場合の表記」を参照してください。
- **データレジスタ (D)**
FC6A 形内部で数値データを書き込むために使用するワード単位のデバイスです。ビット単位のデバイスとしても使用できます。
- **特殊データレジスタ (D)**
FC6A 形内部で数値データを書き込むために使用するワード単位のデバイスです。それぞれのデータレジスタに特殊な機能が割り当てられています。ビット単位のデバイスとしても使用できます。



- 内部リレー (M0000 ~ M7997、M10000 ~ M21247) と特殊内部リレー (M8000 ~ M9997) のデバイス記号は同じ "M" ですが、デバイスの特性が異なります。特殊内部リレーのそれぞれのビットには特殊な機能が割り当てられています。
- データレジスタ (D0000 ~ D7999、D10000 ~ D61999、D70000 ~ D269999) および特殊データレジスタ (D8000 ~ D8899) のデバイス記号は同じ "D" ですが、デバイスの特性が異なります。それぞれの特殊データレジスタには特殊な機能が割り当てられています。

特殊内部リレー

特殊内部リレー一覧



特殊内部リレー一覧でリザーブと書かれたエリアのデータは書き換えしないでください。システムが正常に動作しなくなる恐れがあります。



R/W は、Read（読み出し）/Write（書き込み）の略です。

R/W 欄の表記は、次のとおりです。

R/W：読み出し・書き込み両方可能

R：読み出しのみ可能

W：書き込みのみ可能

アドレス	内容	STOP 時	停電時	R/W	
M8000	スタートコントロール	保持	保持	R/W	
M8001	1秒クロックリセット	クリア	クリア	R/W	
M8002	全出力OFF	クリア	クリア	R/W	
M8003	キャリア/ボロー	クリア	クリア	R/W	
M8004	ユーザープログラム実行エラー	クリア	クリア	R/W	
M8005	通信エラー	保持	クリア	R/W	
M8006	通信禁止フラグ（データリンク親局時）	保持	保持	R/W	
M8007	初期化フラグ（データリンク親局時）/通信停止フラグ（データリンク子局時）	クリア	クリア	R/W	
M8010	ステータスLED 動作	動作	クリア	R/W	
M8011 M8012	リザーブ	—	—	—	
M8013	カレンダー・時計書き込み・アジャストエラー	動作	クリア	R/W	
M8014	カレンダー・時計読み出しエラー	動作	クリア	R/W	
M8015	リザーブ	—	—	—	
M8016	カレンダー書き込み	動作	クリア	R/W	
M8017	時計書き込み	動作	クリア	R/W	
M8020	カレンダー・時計書き込み	動作	クリア	R/W	
M8021	時計アジャスト	動作	クリア	R/W	
M8022	ユーザー通信受信命令キャンセルフラグ（ポート1）	クリア	クリア	R/W	
M8023	ユーザー通信受信命令キャンセルフラグ（ポート2）	クリア	クリア	R/W	
M8024	WSFT・BMOV命令実行中	保持	保持	R/W	
M8025	STOP中出力保持	保持	クリア	R/W	
M8026	ユーザー通信受信命令キャンセルフラグ（ポート3）	クリア	クリア	R/W	
M8027	高速カウンタ（グループ1/10）	カウント方向フラグ	保持	クリア	R/W
M8030		外部出力クリア	クリア	クリア	R/W
M8031		ゲート入力	保持	クリア	R/W
M8032		ソフトリセット	保持	クリア	R/W
M8033	ユーザー通信受信命令キャンセルフラグ（ポート4）	クリア	クリア	R/W	
M8034	高速カウンタ（グループ3/13）	外部出力クリア	クリア	クリア	R/W
M8035		ゲート入力	保持	クリア	R/W
M8036		ソフトリセット	保持	クリア	R/W
M8037	リザーブ	—	—	—	
M8040	高速カウンタ（グループ4/14）	外部出力クリア	クリア	クリア	R/W
M8041		ゲート入力	保持	クリア	R/W
M8042		ソフトリセット	保持	クリア	R/W
M8043	高速カウンタ（グループ5/16）	カウント方向フラグ	保持	クリア	R/W
M8044		外部出力クリア	クリア	クリア	R/W
M8045		ゲート入力	保持	クリア	R/W
M8046		ソフトリセット	保持	クリア	R/W

アドレス	内容	STOP時	停電時	R/W	
M8047 M8050	リザーブ	—	—	—	
M8051	高速カウンタ (グループ2/11)	外部出力クリア	クリア	クリア	R/W
M8052		ゲート入力	保持	クリア	R/W
M8053		ソフトリセット	保持	クリア	R/W
M8054		比較一致	保持	クリア	R
M8055		オーバーフロー	保持	クリア	R
M8056	リザーブ	—	—	—	
M8057	高速カウンタ (グループ6/17)	外部出力クリア	クリア	クリア	R/W
M8060		ゲート入力	保持	クリア	R/W
M8061		ソフトリセット	保持	クリア	R/W
M8062		比較一致	保持	クリア	R
M8063		オーバーフロー	保持	クリア	R
M8064 } M8067	リザーブ	—	—	—	
M8070	SDメモ리카ードマウント状態	保持	クリア	R	
M8071	SDメモ리카ードアクセス中	保持	クリア	R	
M8072	SDメモ리카ードマウント解除	動作	クリア	R/W	
M8073	機能スイッチ状態	動作	クリア	R	
M8074	電池電圧計測フラグ	動作	クリア	R/W	
M8075 } M8077	リザーブ	—	—	—	
M8080	データリンク子局1通信完了リレー (データリンク親局時)	動作	クリア	R	
M8081	データリンク子局2 通信完了リレー	動作	クリア	R	
M8082	データリンク子局3 通信完了リレー	動作	クリア	R	
M8083	データリンク子局4 通信完了リレー	動作	クリア	R	
M8084	データリンク子局5 通信完了リレー	動作	クリア	R	
M8085	データリンク子局6 通信完了リレー	動作	クリア	R	
M8086	データリンク子局7 通信完了リレー	動作	クリア	R	
M8087	データリンク子局8 通信完了リレー	動作	クリア	R	
M8090	データリンク子局9 通信完了リレー	動作	クリア	R	
M8091	データリンク子局10 通信完了リレー	動作	クリア	R	
M8092	データリンク子局11 通信完了リレー	動作	クリア	R	
M8093	データリンク子局12 通信完了リレー	動作	クリア	R	
M8094	データリンク子局13 通信完了リレー	動作	クリア	R	
M8095	データリンク子局14 通信完了リレー	動作	クリア	R	
M8096	データリンク子局15 通信完了リレー	動作	クリア	R	
M8097	データリンク子局16 通信完了リレー	動作	クリア	R	
M8100	データリンク子局17 通信完了リレー	動作	クリア	R	
M8101	データリンク子局18 通信完了リレー	動作	クリア	R	
M8102	データリンク子局19 通信完了リレー	動作	クリア	R	
M8103	データリンク子局20 通信完了リレー	動作	クリア	R	
M8104	データリンク子局21 通信完了リレー	動作	クリア	R	
M8105	データリンク子局22 通信完了リレー	動作	クリア	R	
M8106	データリンク子局23 通信完了リレー	動作	クリア	R	
M8107	データリンク子局24 通信完了リレー	動作	クリア	R	
M8110	データリンク子局25 通信完了リレー	動作	クリア	R	
M8111	データリンク子局26 通信完了リレー	動作	クリア	R	
M8112	データリンク子局27 通信完了リレー	動作	クリア	R	
M8113	データリンク子局28 通信完了リレー	動作	クリア	R	

アドレス	内容		STOP時	停電時	R/W
M8114	データリンク子局29 通信完了リレー		動作	クリア	R
M8115	データリンク子局30 通信完了リレー		動作	クリア	R
M8116	データリンク子局31 通信完了リレー		動作	クリア	R
M8117	データリンク全子局通信完了リレー		動作	クリア	R
M8120	イニシャライズパルス		クリア	クリア	R
M8121	1秒クロック		動作	クリア	R
M8122	100ミリ秒クロック		動作	クリア	R
M8123	10ミリ秒クロック		動作	クリア	R
M8124	タイマ・カウンタ設定値変更ステータス		保持	クリア	R
M8125	運転中出力		クリア	クリア	R
M8126	RUN 中ダウンロード完了後 1 スキャンON		クリア	クリア	R
M8127	リザーブ		—	—	—
M8130	高速カウンタ (グループ1/10)	リセットステータス	保持	クリア	R
M8131		比較一致	保持	クリア	R
M8132	リザーブ		—	—	—
M8133	高速カウンタ (グループ3/13)	比較一致	保持	クリア	R
M8134	高速カウンタ (グループ4/14)	比較一致	保持	クリア	R
M8135	高速カウンタ (グループ5/16)	リセットステータス	保持	クリア	R
M8136		比較一致	保持	クリア	R
M8137	割込入力10 ステータス (グループ1/10)	(ON : 許可、OFF : 禁止)	クリア	クリア	R
M8140	割込入力11 ステータス (グループ2/11)		クリア	クリア	R
M8141	割込入力13 ステータス (グループ3/13)		クリア	クリア	R
M8142	割込入力14 ステータス (グループ4/14)		クリア	クリア	R
M8143	割込入力16 ステータス (グループ5/16)		クリア	クリア	R
M8144	タイマ割込ステータス		クリア	クリア	R
M8145	ユーザー通信受信命令キャンセルフラグ (ポート5)		クリア	クリア	R/W
M8146	ユーザー通信受信命令キャンセルフラグ (ポート6)		クリア	クリア	R/W
M8147	ユーザー通信受信命令キャンセルフラグ (ポート7)		クリア	クリア	R/W
M8150	比較結果1		保持	クリア	R
M8151	比較結果2		保持	クリア	R
M8152	比較結果3		保持	クリア	R
M8153	キャッチ入力時のON/OFF状態	グループ1/10	保持	クリア	R
M8154		グループ2/11	保持	クリア	R
M8155		グループ3/13	保持	クリア	R
M8156		グループ4/14	保持	クリア	R
M8157		グループ5/16	保持	クリア	R
M8160		グループ6/17	保持	クリア	R
M8161	高速カウンタ (グループ1/10)	オーバーフロー	保持	クリア	R
M8162		アンダーフロー	保持	クリア	R
M8163	高速カウンタ (グループ5/16)	オーバーフロー	保持	クリア	R
M8164		アンダーフロー	保持	クリア	R
M8165	高速カウンタ (グループ3/13)	オーバーフロー	保持	クリア	R
M8166	高速カウンタ (グループ4/14)	オーバーフロー	保持	クリア	R
M8167	割込入力17 ステータス (グループ6/17)	(ON : 許可、OFF : 禁止)	保持	クリア	R
M8170	ユーザー通信受信命令キャンセルフラグ (ポート8)		クリア	クリア	R/W
M8171	リザーブ		—	—	—
M8172	トランジスタソース出力過電流検出	グループ1	動作	クリア	R
M8173		グループ2	動作	クリア	R
M8174		グループ3	動作	クリア	R
M8175		グループ4	動作	クリア	R
M8176	ユーザー通信受信命令キャンセルフラグ (ポート9)		クリア	クリア	R/W

アドレス	内容	STOP時	停電時	R/W	
M8177 └ M8183	リザーブ	—	—	—	
M8184	HMIモジュール ネットワーク設定変更トリガ	動作	クリア	R/W	
M8185	サマータイム期間中	動作	クリア	R	
M8186	Ethernetポート1 自動Ping実行中	動作	クリア	R	
M8187	Ethernetポート1 自動Ping停止フラグ	動作	クリア	R/W	
M8190	CPUモジュール Ethernetポート1 ネットワーク設定変更トリガ	動作	クリア	R/W	
M8191	SNTP取得フラグ	動作	クリア	R/W	
M8192	割込入力I0エッジ	ON：立ち上がりエッジ OFF：立ち下がりエッジ	クリア	クリア	R
M8193	割込入力I3エッジ		クリア	クリア	R
M8194	割込入力I4エッジ		クリア	クリア	R
M8195	割込入力I6エッジ		クリア	クリア	R
M8196	割込入力I7エッジ		クリア	クリア	R
M8197	割込入力I1エッジ		クリア	クリア	R
M8200	ユーザー通信受信命令キャンセルフラグ	コネクション1	クリア	クリア	R/W
M8201		コネクション2	クリア	クリア	R/W
M8202		コネクション3	クリア	クリア	R/W
M8203		コネクション4	クリア	クリア	R/W
M8204		コネクション5	クリア	クリア	R/W
M8205		コネクション6	クリア	クリア	R/W
M8206		コネクション7	クリア	クリア	R/W
M8207		コネクション8	クリア	クリア	R/W
M8210	リザーブ	—	—	—	
M8211	HMIモジュール 送信メールサーバー設定の初期化	動作	クリア	R/W	
M8212	コネクションステータス (ON：接続あり、OFF：接続なし)	コネクション1	動作	クリア	R
M8213		コネクション2	動作	クリア	R
M8214		コネクション3	動作	クリア	R
M8215		コネクション4	動作	クリア	R
M8216		コネクション5	動作	クリア	R
M8217		コネクション6	動作	クリア	R
M8220		コネクション7	動作	クリア	R
M8221		コネクション8	動作	クリア	R
M8222	ユーザー通信コネクション切断	コネクション1	動作	クリア	R/W
M8223		コネクション2	動作	クリア	R/W
M8224		コネクション3	動作	クリア	R/W
M8225		コネクション4	動作	クリア	R/W
M8226		コネクション5	動作	クリア	R/W
M8227		コネクション6	動作	クリア	R/W
M8230		コネクション7	動作	クリア	R/W
M8231		コネクション8	動作	クリア	R/W
M8232	HMIモジュールコネクション情報参照 コネクションステータス	動作	クリア	R	
M8233 └ M8247	リザーブ	—	—	—	
M8250	SDメモ리카ードからのダウンロード実行フラグ	動作	クリア	R/W	
M8251	SDメモ리카ードへのアップロード実行フラグ	動作	クリア	R/W	
M8252	SDメモ리카ードダウンロード実行中	動作	クリア	R	
M8253	SDメモ리카ードアップロード実行中	動作	クリア	R	
M8254	SDメモ리카ードダウンロード/アップロード実行完了出力	動作	クリア	R	
M8255	SDメモ리카ードダウンロード/アップロード実行エラー出力	動作	クリア	R	

アドレス	内容		STOP時	停電時	R/W
M8256 M8257	リザーブ		—	—	—
M8260	レシビ書き込み実行フラグ		動作	クリア	R/W
M8261	レシビ読み出し実行フラグ		動作	クリア	R/W
M8262	レシビ書き込み実行中		動作	クリア	R/W
M8263	レシビ読み出し実行中		動作	クリア	R/W
M8264	レシビチャンネル実行完了出力		動作	クリア	R/W
M8265	レシビ実行エラー出力		動作	クリア	R/W
M8266	レシビブロック実行完了出力		動作	クリア	R/W
M8267	レシビ内蔵メモリ (ROM-領域) 読み出し制限		動作	クリア	R
M8270	リザーブ		—	—	—
M8271	SDメモリカードからのMQTT基本設定ダウンロード実行フラグ		動作	クリア	R/W
M8272	SDメモリカードからのMQTT基本設定ダウンロード実行完了出力		動作	クリア	R
M8273	SDメモリカードからのMQTT基本設定ダウンロードエラー出力		動作	クリア	R
M8274 } M8297	リザーブ		—	—	—
M8300	J1939通信許可フラグ		クリア	クリア	R/W
M8301	J1939オンラインステータス		クリア	クリア	R
M8302	J1939自局アドレス確定ステータス		クリア	クリア	R
M8303	J1939通信エラー発生出力		クリア	クリア	R
M8304	J1939通信バスオフ発生出力		クリア	クリア	R
M8305 } M8310	リザーブ		—	—	—
M8311	ESC+キー入力 (上)	ESC+キー入力 (◀)	クリア	クリア	R
M8312	ESC+キー入力 (下)	ESC+キー入力 (◀)	クリア	クリア	R
M8313	ESC+キー入力 (左)	ESC+キー入力 (◀)	クリア	クリア	R
M8314	ESC+キー入力 (右)	ESC+キー入力 (▶)	クリア	クリア	R
M8315 } M8319	リザーブ		—	—	—
M8320	増設拡張モジュール分離型マスター /スレーブ初期化		動作	クリア	R/W
M8321 } M8330	リザーブ		—	—	—
M8331	Ethernetポート2 自動Ping実行中		動作	クリア	R
M8332	Ethernetポート2 自動Ping停止フラグ		動作	クリア	R/W
M8333	CPUモジュール Ethernetポート2 ネットワーク設定変更トリガ		動作	クリア	R/W
M8334	ユーザー通信受信命令キャンセルフラグ	コネクション9	クリア	クリア	R/W
M8335		コネクション10	クリア	クリア	R/W
M8336		コネクション11	クリア	クリア	R/W
M8337		コネクション12	クリア	クリア	R/W
M8340		コネクション13	クリア	クリア	R/W
M8341		コネクション14	クリア	クリア	R/W
M8342		コネクション15	クリア	クリア	R/W
M8343		コネクション16	クリア	クリア	R/W
M8344	Ethernetポート1 送信メールサーバー設定の初期化		動作	クリア	R/W

アドレス	内容	STOP時	停電時	R/W	
M8345	コネクションステータス (ON:接続あり、OFF:接続なし)	コネクション9	動作	クリア	R
M8346		コネクション10	動作	クリア	R
M8347		コネクション11	動作	クリア	R
M8350		コネクション12	動作	クリア	R
M8351		コネクション13	動作	クリア	R
M8352		コネクション14	動作	クリア	R
M8353		コネクション15	動作	クリア	R
M8354		コネクション16	動作	クリア	R
M8355	ユーザー通信コネクション切断	コネクション9	動作	クリア	R/W
M8356		コネクション10	動作	クリア	R/W
M8357		コネクション11	動作	クリア	R/W
M8360		コネクション12	動作	クリア	R/W
M8361		コネクション13	動作	クリア	R/W
M8362		コネクション14	動作	クリア	R/W
M8363		コネクション15	動作	クリア	R/W
M8364		コネクション16	動作	クリア	R/W
M8365	ユーザー通信受信命令キャンセルフラグ (ポート10)	クリア	クリア	R/W	
M8366	ユーザー通信受信命令キャンセルフラグ (ポート11)	クリア	クリア	R/W	
M8367	ユーザー通信受信命令キャンセルフラグ (ポート12)	クリア	クリア	R/W	
M8370	ユーザー通信受信命令キャンセルフラグ (ポート13)	クリア	クリア	R/W	
M8371	ユーザー通信受信命令キャンセルフラグ (ポート14)	クリア	クリア	R/W	
M8372	ユーザー通信受信命令キャンセルフラグ (ポート15)	クリア	クリア	R/W	
M8373	ユーザー通信受信命令キャンセルフラグ (ポート16)	クリア	クリア	R/W	
M8374	ユーザー通信受信命令キャンセルフラグ (ポート17)	クリア	クリア	R/W	
M8375	ユーザー通信受信命令キャンセルフラグ (ポート18)	クリア	クリア	R/W	
M8376	ユーザー通信受信命令キャンセルフラグ (ポート19)	クリア	クリア	R/W	
M8377	ユーザー通信受信命令キャンセルフラグ (ポート20)	クリア	クリア	R/W	
M8380	ユーザー通信受信命令キャンセルフラグ (ポート21)	クリア	クリア	R/W	
M8381	ユーザー通信受信命令キャンセルフラグ (ポート22)	クリア	クリア	R/W	
M8382	ユーザー通信受信命令キャンセルフラグ (ポート23)	クリア	クリア	R/W	
M8383	ユーザー通信受信命令キャンセルフラグ (ポート24)	クリア	クリア	R/W	
M8384	ユーザー通信受信命令キャンセルフラグ (ポート25)	クリア	クリア	R/W	
M8385	ユーザー通信受信命令キャンセルフラグ (ポート26)	クリア	クリア	R/W	
M8386	ユーザー通信受信命令キャンセルフラグ (ポート27)	クリア	クリア	R/W	
M8387	ユーザー通信受信命令キャンセルフラグ (ポート28)	クリア	クリア	R/W	
M8390	ユーザー通信受信命令キャンセルフラグ (ポート29)	クリア	クリア	R/W	
M8391	ユーザー通信受信命令キャンセルフラグ (ポート30)	クリア	クリア	R/W	
M8392	ユーザー通信受信命令キャンセルフラグ (ポート31)	クリア	クリア	R/W	
M8393	ユーザー通信受信命令キャンセルフラグ (ポート32)	クリア	クリア	R/W	
M8394	ユーザー通信受信命令キャンセルフラグ (ポート33)	クリア	クリア	R/W	
M8395 } M8400	リザーブ	—	—	—	
M8401	SDメモ리카ードからのサーバー機能用ファイルダウンロード実行フラグ	動作	クリア	R/W	
M8402	SDメモ리카ードからのサーバー機能用ファイルダウンロード実行完了出力	動作	クリア	R	
M8403	SDメモ리카ードからのサーバー機能用ファイルダウンロードエラー出力	動作	クリア	R	
M8404	Plus CPUモジュールのWebサーバーの起動状態	動作	クリア	R	
M8405 } M8447	リザーブ	—	—	—	
M8450	BACnet通信許可	動作	クリア	R/W	

アドレス	内容	STOP時	停電時	R/W	
M8451 ┆ M8457	リザーブ	—	—	—	
M8460	EtherNet/IP通信許可	クリア	クリア	R/W	
M8461 ┆ M8597	リザーブ	—	—	—	
M8600		リセットステータス	保持	クリア	R
M8601	高速カウンタ (グループ3/13)	アンダーフロー	保持	クリア	R
M8602		カウント方向フラグ	保持	クリア	R
M8603 ┆ M8997	リザーブ	—	—	—	

特殊内部リレー補足説明

■ M8000：スタートコントロール

FC6A 形の RUN/STOP 状態をコントロールします。M8000 を ON にすると RUN 状態になり、OFF にすると STOP 状態になります。ラダープログラミングマニュアル「第 1 章 ユーザープログラムを RUN/STOP する」を参照してください。ただし、機能スイッチ、ストップ入力およびリセット入力は、スタートコントロールよりも優先されます。M8000 は停電時に状態を保持しますが、バックアップ時間を超えて保持データが消えた場合は、[ファンクション設定]の[キープデータエラー発生時の RUN/STOP 指定]で設定した内容 (RUN 指定 /STOP 指定) にしたかった動作となります。詳細は、FC6A 形マイクロスマート ユーザーズマニュアル「第 5 章 キープデータエラー発生時の RUN/STOP 指定」を参照してください。

■ M8001：1 秒クロックリセット

M8001 が ON の間、M8121 (1 秒クロック) は常に OFF になります。

■ M8002：全出力 OFF

M8002 が ON の間、全出力が OFF になります。ラダープログラムで作成した自己保持回路も OFF になります。

■ M8003：キャリア / ボロー

演算命令を実行中にキャリア (Cy) またはボロー (Bw) が発生すると ON になります。詳細は、ラダープログラミングマニュアル「第 3 章 ●キャリア / ボロー」を参照してください。

■ M8004：ユーザープログラム実行エラー

ユーザープログラムを実行中にエラーが発生すると ON になります。ユーザープログラム実行エラーの詳細は、FC6A 形マイクロスマート ユーザーズマニュアル「第 13 章 ユーザープログラム実行エラー一覧」を参照してください。

■ M8005：通信エラー

データリンク通信時にエラーが発生すると ON になります。エラーが解除されても保持します。

■ M8006：通信禁止フラグ (データリンク親局時)

データリンク通信時、M8006 が ON の間、通信を停止します。

■ M8007：初期化フラグ (データリンク親局時) / 通信停止フラグ (データリンク子局時)

データリンク親局時 : RUN 状態でこのフラグを OFF から ON にすると、接続状態を確認するためにデータリンクが 1 回のみ初期化されます。データリンクを構成している子局の電源立ち上げタイミングが、親局よりも遅いときに使用します。

データリンク子局時 : 親局からの通信が 10s 以上途絶えると、このフラグが ON になります。正常な受信ができれば OFF になります。

■ M8010：ステータス LED 動作

M8010 が ON の間、ステータス LED [STAT] が点灯します。OFF の間、ステータス LED [STAT] が消灯します。

■ M8013：カレンダー・時計書き込み・アジャストエラー

時計書き込みまたは時計アジャスト処理が正常に実行できなかったときに ON になります。処理が正常に完了すると OFF になります。

■ M8014：カレンダー・時計読み出しエラー

内蔵時計から特殊データレジスタ (D8008 ~ D8021) のカレンダーデータおよび時計データへの読み出しに失敗すると ON になります。読み出しが正常に完了すると OFF になります。

■ M8016：カレンダー書き込み

カレンダーデータ (書き込み専用) の特殊データレジスタ (D8015 ~ D8018) にデータを書き込んだあとで M8016 を OFF から ON にすると、内蔵時計にカレンダーデータ (年、月、日、曜日) を書き込みます。

■ M8017：時計書き込み

時計データ (書き込み専用) の特殊データレジスタ (D8019 ~ D8021) にデータを書き込んだあとで M8017 を OFF から ON にすると、内蔵時計に時計データ (時、分、秒) を書き込みます。

■ M8020：カレンダー・時計書き込み

カレンダー・時計データ (書き込み専用) の特殊データレジスタ (D8015 ~ D8021) にデータを書き込んだあとで M8020 を OFF から ON にすると、内蔵時計にカレンダーデータ (年、月、日、曜日) および時計データ (時、分、秒) を書き込みます。

■ M8021：時計アジャスト

M8021 を OFF から ON にすると、内蔵時計の秒データを補正します。

・秒データが 0 ~ 29 の間に M8021 を OFF から ON にすると、秒データを 0 にします。

・秒データが 30 ~ 59 の間に M8021 を OFF から ON にすると、分データを +1 して、秒データを 0 にします。

- **M8022：ユーザー通信受信命令キャンセルフラグ（ポート 1）**
M8022 が ON の間、ポート 1 で実行中のユーザー通信（受信命令）をキャンセルします。
- **M8023：ユーザー通信受信命令キャンセルフラグ（ポート 2）**
M8023 が ON の間、ポート 2 で実行中のユーザー通信（受信命令）をキャンセルします。
- **M8024：WSFT・BMOV 命令実行中**
WSFT（ワードシフト）命令、BMOV（ブロックムーブ）命令の実行中に ON になり、命令動作完了（正常終了）後に OFF になります。
- **M8025：STOP 中出力保持**
FC6A 形の RUN 中に M8025 を ON にして STOP すると、出力は RUN 中の状態を保持します。再び RUN すると自動的に M8025 は OFF になります。
- **M8026：ユーザー通信受信命令キャンセルフラグ（ポート 3）**
M8026 が ON の間、ポート 3 で実行中のユーザー通信（受信命令）をキャンセルします。
- **M8033、M8145～M8147、M8170、M8176、M8365～M8394：ユーザー通信受信命令キャンセルフラグ（ポート 4～33）**
これらのフラグが ON の間、対応するポートで実行中のユーザー通信（受信命令）をキャンセルします。
M8033 = ユーザー通信受信命令キャンセルフラグ（ポート 4）
M8145 = ユーザー通信受信命令キャンセルフラグ（ポート 5）
M8146 = ユーザー通信受信命令キャンセルフラグ（ポート 6）
M8147 = ユーザー通信受信命令キャンセルフラグ（ポート 7）
M8170 = ユーザー通信受信命令キャンセルフラグ（ポート 8）
M8176 = ユーザー通信受信命令キャンセルフラグ（ポート 9）
M8365 = ユーザー通信受信命令キャンセルフラグ（ポート 10）
M8366 = ユーザー通信受信命令キャンセルフラグ（ポート 11）
M8367 = ユーザー通信受信命令キャンセルフラグ（ポート 12）
M8370 = ユーザー通信受信命令キャンセルフラグ（ポート 13）
M8371 = ユーザー通信受信命令キャンセルフラグ（ポート 14）
M8372 = ユーザー通信受信命令キャンセルフラグ（ポート 15）
M8373 = ユーザー通信受信命令キャンセルフラグ（ポート 16）
M8374 = ユーザー通信受信命令キャンセルフラグ（ポート 17）
M8375 = ユーザー通信受信命令キャンセルフラグ（ポート 18）
M8376 = ユーザー通信受信命令キャンセルフラグ（ポート 19）
M8377 = ユーザー通信受信命令キャンセルフラグ（ポート 20）
M8380 = ユーザー通信受信命令キャンセルフラグ（ポート 21）
M8381 = ユーザー通信受信命令キャンセルフラグ（ポート 22）
M8382 = ユーザー通信受信命令キャンセルフラグ（ポート 23）
M8383 = ユーザー通信受信命令キャンセルフラグ（ポート 24）
M8384 = ユーザー通信受信命令キャンセルフラグ（ポート 25）
M8385 = ユーザー通信受信命令キャンセルフラグ（ポート 26）
M8386 = ユーザー通信受信命令キャンセルフラグ（ポート 27）
M8387 = ユーザー通信受信命令キャンセルフラグ（ポート 28）
M8390 = ユーザー通信受信命令キャンセルフラグ（ポート 29）
M8391 = ユーザー通信受信命令キャンセルフラグ（ポート 30）
M8392 = ユーザー通信受信命令キャンセルフラグ（ポート 31）
M8393 = ユーザー通信受信命令キャンセルフラグ（ポート 32）
M8394 = ユーザー通信受信命令キャンセルフラグ（ポート 33）
- **M8027～M8032、M8034～M8036、M8040～M8046、M8051～M8055、M8057～M8063、M8130、M8131、M8133～M8136、M8161～M8167、M8600～M8602：高速カウンタ用特殊内部リレー**
高速カウンタに使用する特殊内部リレーです。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 高速カウンタ」を参照してください。
M8027～M8032、M8130、M8131、M8161、M8162 = 高速カウンタ（グループ 1/10）
M8034～M8036、M8133、M8165、M8600～M8602 = 高速カウンタ（グループ 3/13）
M8040～M8042、M8134、M8166 = 高速カウンタ（グループ 4/14）
M8043～M8046、M8135、M8136、M8163、M8164 = 高速カウンタ（グループ 5/16）
M8051～M8055 = 高速カウンタ（グループ 2/11）
M8057～M8063 = 高速カウンタ（グループ 6/17）

■ M8070：SD メモリカードマウント状態

FC6A 形に SD メモリカードを挿入し、SD メモリカードが認識されて使用できる状態になると ON になります。SD メモリカードが挿入されていない、または認識されないときは OFF になります。

■ M8071：SD メモリカードアクセス中

SD メモリカードへのアクセス中は ON になります。アクセスが完了すると OFF になります。

■ M8072：SD メモリカードマウント解除

M8072 を OFF から ON にすると、SD メモリカードへのアクセスを停止します。アクセス停止した SD メモリカードをアクセス可能にするためには、SD メモリカードを再度挿入してください。

■ M8073：機能スイッチ状態

CPU モジュールの前面にある機能スイッチの状態を示します。

機能スイッチが 1 のときに ON になります。機能スイッチが 0 のときに OFF になります。

■ M8074：電池電圧計測フラグ

バックアップ用電池の電池電圧の計測状態を示します。

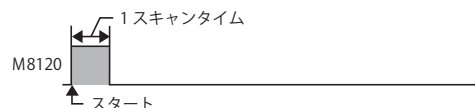
M8074 を OFF から ON にすると電池電圧の計測を開始し、計測が完了すると OFF になります。

■ M8080～M8117：データリンク通信完了リレー

データリンク通信に使用する特殊内部リレーです。詳細は、FC6A 形マイクロスマート 通信 マニュアル「第 7 章 親局側データリンク通信完了リレー」を参照してください。

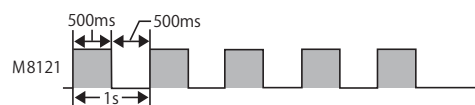
■ M8120：イニシャライズパルス

運転 (RUN) 開始時の 1 スキャンのみ ON になります。



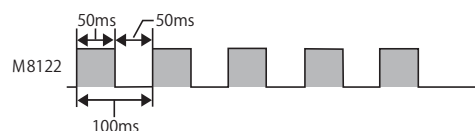
■ M8121：1 秒クロック

M8001 が OFF の間、M8121 は 1s 周期で ON と OFF (デューティ比 1:1) を繰り返します。



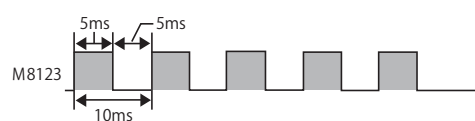
■ M8122：100 ミリ秒クロック

M8122 は 100ms 周期で ON と OFF (デューティ比 1:1) を繰り返します。



■ M8123：10 ミリ秒クロック

M8123 は 10ms 周期で ON と OFF (デューティ比 1:1) を繰り返します。



■ M8124：タイマ・カウンタ設定値変更ステータス

タイマ・カウンタの設定を変更すると ON します。ユーザープログラム転送時または変更データクリア時に OFF になります。

■ M8125：運転中出力

RUN 状態の間、ON します。

■ M8126：RUN 中ダウンロード完了後 1 スキャン ON

RUN 状態でユーザープログラムを変更すると (RUN 中ダウンロード)、ダウンロード完了後、変更されたユーザープログラムの運転開始時に 1 スキャンのみ ON します。

■ M8137～M8143、M8167：割込入力ステータス

対応するユーザー割込が許可されているときに ON になります。ユーザー割込が禁止のときは OFF になります。

M8137 = 割込入力 I0 ステータス

M8140 = 割込入力 I1 ステータス

M8141 = 割込入力 I3 ステータス

M8142 = 割込入力 I4 ステータス

M8143 = 割込入力 I6 ステータス

M8167 = 割込入力 I7 ステータス

■ M8144：タイマ割込ステータス

タイマ割込が許可されているときに ON します。タイマ割込が禁止のときは OFF します。

■ M8150～M8152：比較結果

CMP= (コンペア (=)) 命令、ICMP>= (区間比較) 命令の比較結果に応じて M8150～M8152 が ON します。

詳細は、ラダープログラミング マニュアル「第 6 章 特殊内部リレーの動作 (M8150、M8151、M8152)」を参照してください。

■ M8153～M8160：キャッチ入力時の ON/OFF 状態

1 スキャン中に、キャッチ入力に指定した入力接点の立ち上がり / 立ち下がり入力を検出すると、入力接点の状態を取り込みます。検出可能なエッジは 1 スキャンに 1 回です。

M8153 = グループ 1/10 の状態

M8154 = グループ 2/11 の状態

M8155 = グループ 3/13 の状態

M8156 = グループ 4/14 の状態

M8157 = グループ 5/16 の状態

M8160 = グループ 6/17 の状態

■ M8172～M8175：トランジスタソース出力過電流検出

CPU モジュールのトランジスタプロテクトソース出力で過電流出力が発生すると、特殊内部リレー (M8172～M8175) を ON します。出力 4 点を 1 グループとして、次の特殊内部リレーが割り当てられています。いずれかの特殊内部リレーで過電流出力が発生すると ON します。

過電流出力が解消されても、これらの特殊内部リレーは OFF に戻りません。OFF に戻す場合は、ラダープログラムで OFF となるようプログラミングしてください。

M8172 = グループ 1 (Q0～Q3) の状態

M8173 = グループ 2 (Q4～Q7) の状態

M8174 = グループ 3 (Q10～Q13) の状態

M8175 = グループ 4 (Q14～Q17) の状態

■ M8184：HMI モジュール ネットワーク設定変更トリガ

M8184 を OFF から ON にすると、D8437～D8456 に格納されている値を HMI モジュールの IP アドレスとして設定します。

D8437～D8456 の値を変更しただけでは、設定されません。HMI モジュール IP アドレス変更の詳細は、「第 3 章 HMI モジュールの特殊データレジスタによるネットワーク設定」(3-6 頁) を参照してください。

■ M8185：サマータイム期間中

サマータイム機能が有効のとき、サマータイム期間中のときに ON になります。サマータイム期間外のときは OFF になります。

サマータイム機能が無効のとき、OFF になります。

■ M8186：Ethernet ポート 1 自動 Ping 実行中

Ethernet ポート 1 の自動 Ping が動作中のときに ON になります。自動 Ping が停止中のときに OFF になります。自動 Ping の詳細は、「第 3 章 自動 Ping 機能」(3-23 頁) を参照してください。

■ M8187：Ethernet ポート 1 自動 Ping 停止フラグ

M8187 が ON の間、Ethernet ポート 1 の自動 Ping を停止します。M8187 が OFF の間、自動 Ping を実行します。このとき、自動 Ping は前回の終了状態によらず、リモートホストリストで指定したリモートホスト番号の最小番号から実行します。

■ M8190：CPU モジュール Ethernet ポート 1 ネットワーク設定変更トリガ

M8190 を OFF から ON にすると、D8303～D8323 に格納されている値を CPU モジュールの Ethernet ポート 1 の IP 設定 / DNS 設定として設定します。

D8303～D8323 は値を変更しただけでは、設定されません。CPU モジュールの Ethernet ポート 1 の IP 設定 / DNS 設定変更の詳細は、「第 3 章 特殊データレジスタによるネットワーク設定」(3-4 頁) を参照してください。

■ M8191：SNTP 取得フラグ

M8191 を OFF から ON にすると、SNTP サーバーからの時刻情報取得を実行します。

■ M8192～M8197：割込入力エッジ

割込入力の立上りエッジで割り込みが発生したときに ON になります。割込入力の立ち下がりエッジで割り込みが発生したときは OFF します。

M8192 = 割込入力 I0 エッジ

M8193 = 割込入力 I3 エッジ

M8194 = 割込入力 I4 エッジ

M8195 = 割込入力 I6 エッジ

M8196 = 割込入力 I7 エッジ

M8197 = 割込入力 I1 エッジ

■ M8200 ～ M8207、M8334 ～ M8343：ユーザー通信受信命令キャンセルフラグ

M8200 ～ M8207、M8334 ～ M8343 を OFF から ON にすると、実行中のユーザー通信受信命令を中断します。

M8200 = クライアント接続 1 で実行中のユーザー通信受信命令
M8201 = クライアント接続 2 で実行中のユーザー通信受信命令
M8202 = クライアント接続 3 で実行中のユーザー通信受信命令
M8203 = クライアント接続 4 で実行中のユーザー通信受信命令
M8204 = クライアント接続 5 で実行中のユーザー通信受信命令
M8205 = クライアント接続 6 で実行中のユーザー通信受信命令
M8206 = クライアント接続 7 で実行中のユーザー通信受信命令
M8207 = クライアント接続 8 で実行中のユーザー通信受信命令
M8334 = クライアント接続 9 で実行中のユーザー通信受信命令
M8335 = クライアント接続 10 で実行中のユーザー通信受信命令
M8336 = クライアント接続 11 で実行中のユーザー通信受信命令
M8337 = クライアント接続 12 で実行中のユーザー通信受信命令
M8340 = クライアント接続 13 で実行中のユーザー通信受信命令
M8341 = クライアント接続 14 で実行中のユーザー通信受信命令
M8342 = クライアント接続 15 で実行中のユーザー通信受信命令
M8343 = クライアント接続 16 で実行中のユーザー通信受信命令

■ M8211：HMI モジュール送信メールサーバー設定の初期化

M8211 を OFF から ON にすると、HMI-Ethernet ポートの送信メールサーバーの設定を初期化します。

■ M8212 ～ M8221、M8345 ～ M8354：コネクションステータス

メンテナンス通信サーバー、ユーザー通信サーバー / クライアント、Modbus TCP サーバー / クライアントによりネットワーク機器と接続している間、コネクションステータスが ON になります。接続されていない間、OFF になります。

M8212 = コネクション 1
M8213 = コネクション 2
M8214 = コネクション 3
M8215 = コネクション 4
M8216 = コネクション 5
M8217 = コネクション 6
M8220 = コネクション 7
M8221 = コネクション 8
M8345 = コネクション 9
M8346 = コネクション 10
M8347 = コネクション 11
M8350 = コネクション 12
M8351 = コネクション 13
M8352 = コネクション 14
M8353 = コネクション 15
M8354 = コネクション 16

■ M8222 ～ M8231、M8355 ～ M8364：ユーザー通信コネクション切断

ユーザー通信で接続しているとき、M8222 ～ M8231、M8355 ～ M8364 を OFF から ON にすると、対応するコネクションを切断します。これらの特殊内部リレーが ON のときは、コネクションを確立できません。

M8222 = コネクション 1
M8223 = コネクション 2
M8224 = コネクション 3
M8225 = コネクション 4
M8226 = コネクション 5
M8227 = コネクション 6
M8230 = コネクション 7
M8231 = コネクション 8
M8355 = コネクション 9
M8356 = コネクション 10
M8357 = コネクション 11
M8360 = コネクション 12
M8361 = コネクション 13
M8362 = コネクション 14
M8363 = コネクション 15
M8364 = コネクション 16

ユーザー通信クライアントを使用しているときのみ有効です。

- **M8232：HMI モジュールコネクション情報参照 コネクションステータス**
D8429 で指定したコネクション番号がコネクションありのときに ON になります。コネクションなしのときは OFF になります。
- **M8250：SD メモリカードからのダウンロード実行フラグ**
M8250 を OFF から ON にすると、SD メモリカードから ZLD ファイルをダウンロードします。ダウンロードするファイルは、autoexec.ini ファイルで指定した ZLD ファイルです。
- **M8251：SD メモリカードへのアップロード実行フラグ**
M8251 を OFF から ON にすると、SD メモリカードへ ZLD ファイルをアップロードします。autoexec.ini ファイルで指定したファイル名で ZLD ファイルを作成します。
- **M8252：SD メモリカードダウンロード実行中**
SD メモリカードからのダウンロードを実行開始時に ON し、完了時に OFF します。
- **M8253：SD メモリカードアップロード実行中**
SD メモリカードへのアップロードを実行開始時に ON し、完了時に OFF します。
- **M8254：SD メモリカードダウンロード/アップロード実行完了出力**
SD メモリカードからのダウンロード、SD メモリカードへのアップロードの実行開始時に OFF し、ダウンロード/アップロード完了時に ON します。
- **M8255：SD メモリカードダウンロード/アップロード実行エラー出力**
SD メモリカードからのダウンロード、SD メモリカードへのアップロードの実行完了時に更新します。D8255（ダウンロード/アップロード実行ステータス）が 0 以外のときに ON します。
- **M8260：レシピ書き込み実行フラグ**
OFF から ON にすると、D8260（レシピブロック番号）で指定したブロック番号のすべてのチャンネルに対してレシピ書き込みを実行します。
- **M8261：レシピ読み出し実行フラグ**
OFF から ON にすると、D8260（レシピブロック番号）で指定したブロック番号のすべてのチャンネルに対してレシピ読み出しを実行します。
- **M8262：レシピ書き込み実行中**
レシピ書き込み処理開始時に ON し、完了時に OFF します。レシピ読み出し処理開始時にも OFF します。
- **M8263：レシピ読み出し実行中**
レシピ読み出し処理開始時に ON し、完了時に OFF します。レシピ書き込み処理開始時にも OFF します。
- **M8264：レシピチャンネル実行完了出力**
レシピ読み書き開始時に OFF し、レシピ読み書き完了時に ON します。
- **M8265：レシピ実行エラー出力**
レシピ実行完了時、D8264（レシピ実行ステータス）が 0 以外のときに ON します。レシピの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 レシピ機能」を参照してください。
- **M8266：レシピブロック実行完了出力**
レシピブロック読み書き開始時に OFF し、レシピブロック読み書き完了時に ON します。
- **M8267：レシピ内蔵メモリ（ROM-領域）読み出し制限**
内蔵メモリ（ROM-領域）へのレシピブロック読み出し完了時に ON します。M8267 が ON のとき、内蔵メモリ（ROM-領域 1）および内蔵メモリ（ROM-領域 2）へのレシピ読み出しができません。レシピ読み出しするときは、M8267 を OFF してください。
- **M8271：SD メモリカードからの MQTT 基本設定ダウンロード実行フラグ**
M8271 を ON すると MQTT 基本設定用ファイルをダウンロードします。MQTT 基本設定用ファイルの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。
- **M8272：SD メモリカードからの MQTT 基本設定ダウンロード実行完了出力**
MQTT 基本設定用ファイルをダウンロード開始時に OFF し、終了後に ON します。
- **M8273：SD メモリカードからの MQTT 基本設定ダウンロードエラー出力**
MQTT 基本設定用ファイルをダウンロードする処理を開始したときに OFF します。処理中にエラーが発生した場合、ON します。
- **M8300～M8304：J1939 通信**
J1939 通信で使用する特殊内部リレーです。詳細は、「第 8 章 特殊内部リレー割付」（8-4 頁）を参照してください。
- **M8311～M8314：キー入力状態**
HMI モジュールの ESC キーと方向キーを同時に押している間、ON になります。キーを押していないときは OFF になります。
M8311 = ESC キー+上 (⬆) キー
M8312 = ESC キー+下 (⬇) キー

M8313 = ESC キー+左 (◀) キー

M8314 = ESC キー+右 (▶) キー

■ **M8320：増設拡張モジュール分離型マスター/スレーブ初期化**

M8320 を ON にすると増設拡張モジュール分離型マスター/スレーブおよび増設拡張モジュール分離型スレーブに接続している増設モジュールの初期化します。初期化終了後は、自動的に OFF に戻り、増設拡張モジュール分離型スレーブは I/O リフレッシュを再開します。

■ **M8331：Ethernet ポート 2 自動 Ping 実行中**

Ethernet ポート 2 の自動 Ping が動作中のときに ON になります。自動 Ping が停止中のときに OFF になります。自動 Ping の詳細は、「第 3 章 自動 Ping 機能」(3-23 頁)を参照してください。

■ **M8332：Ethernet ポート 2 自動 Ping 停止フラグ**

M8332 が ON の間、Ethernet ポート 2 の自動 Ping を停止します。M8332 が OFF の間、自動 Ping を実行します。このとき、自動 Ping は前回の終了状態によらず、リモートホストリストで指定したリモートホスト番号の最小番号から実行します。

■ **M8333：CPU モジュール Ethernet ポート 2 ネットワーク設定変更トリガ**

M8333 を OFF から ON にすると、D8630 ～ D8650 に格納されている値を CPU モジュールの Ethernet ポート 2 の IP 設定 / DNS 設定として設定します。

D8630 ～ D8650 は値を変更しただけでは、設定されません。CPU モジュールの Ethernet ポート 2 の IP 設定 / DNS 設定変更の詳細は、「第 3 章 特殊データレジスタによるネットワーク設定」(3-4 頁)を参照してください。

■ **M8344：Ethernet ポート 1 送信メールサーバー設定の初期化**

M8344 を OFF から ON にすると、Ethernet ポート 1 の送信メールサーバーの設定を初期化します。

■ **M8401：SD メモリカードからのサーバー機能用ファイルダウンロード実行フラグ**

M8401 を ON すると、サーバー機能用ファイルをダウンロードします。サーバー機能用ファイルの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

■ **M8402：SD メモリカードからのサーバー機能用ファイルダウンロード実行完了出力**

サーバー機能用ファイルのダウンロード開始時に OFF し、終了後に ON します。

■ **M8403：SD メモリカードからのサーバー機能用ファイルダウンロードエラー出力**

サーバー機能用ファイルのダウンロード開始時に OFF し、処理中にエラーが発生した場合、ON します。

■ **M8404：Plus CPU モジュールの Web サーバーの起動状態**

Plus CPU モジュールの Web サーバーの起動完了時に ON します。下記の場合は OFF します。

- ・ Plus CPU モジュールの電源投入時
- ・ ユーザープログラムのダウンロード開始時
- ・ システムソフトウェアのダウンロード開始時
- ・ サーバー機能用ファイルのダウンロード開始時



D8303 (CPU モジュール Ethernet ポート 1 IP 設定 / DNS 設定切り替え) を変更しても、M8404 は OFF しません。

■ **M8450：BACnet 通信許可**

M8450 が ON の間、BACnet 通信を行います。

■ **M8460：EtherNet/IP 通信許可**

EtherNet/IP 通信を許可 / 禁止します。

OFF：EtherNet/IP 通信禁止

ON：EtherNet/IP 通信許可

特殊データレジスタ

特殊データレジスタ一覧



特殊データレジスタ一覧でリザーブと書かれたエリアのデータは書き換えしないでください。システムが正常に動作しなくなる恐れがあります。



R/W は、Read (読み出し) /Write (書き込み) の略です。

R/W 欄の表記は、次のとおりです。

R/W : 読み出し・書き込み両方可能

R : 読み出しのみ可能

W : 書き込みのみ可能

アドレス	内容		更新のタイミング	R/W
D8000	入力点数		I/O 初期化時	R
D8001	出力点数		I/O 初期化時	R
D8002	CPUモジュール機種情報		電源投入時	R
D8003 D8004	リザーブ		—	—
D8005	一般エラーコード		エラー発生時	R/W
D8006	ユーザープログラム実行エラーコード		エラー発生時	R
D8007	リザーブ		—	—
D8008	カレンダー・時計 現在値 (読み出し専用)	年	500ms ごと	R
D8009		月	500ms ごと	R
D8010		日	500ms ごと	R
D8011		曜日	500ms ごと	R
D8012		時	500ms ごと	R
D8013		分	500ms ごと	R
D8014		秒	500ms ごと	R
D8015	カレンダー・時計 設定値 (書き込み専用)	年	—	W
D8016		月	—	W
D8017		日	—	W
D8018		曜日	—	W
D8019		時	—	W
D8020		分	—	W
D8021		秒	—	W
D8022	スキャンタイムデータ	コンスタントスキャン設定値	—	W
D8023		スキャンタイム (現在値)	毎スキャン	R
D8024		スキャンタイム (最大値)	更新時	R
D8025		スキャンタイム (最小値)	更新時	R
D8026	通信モード情報 (ポート1~3)		毎スキャン	R
D8027 D8028	リザーブ		—	—
D8029	システムバージョン番号		電源投入時	R
D8030	通信カートリッジ情報		電源投入時	R
D8031	オプション接続情報		電源投入時	R
D8032	割込入力ジャンプ先ラベル番号 (I1)		—	R/W
D8033	割込入力ジャンプ先ラベル番号 (I3)		—	R/W
D8034	割込入力ジャンプ先ラベル番号 (I4)		—	R/W
D8035	割込入力ジャンプ先ラベル番号 (I6)		—	R/W
D8036	タイマ割込ジャンプ先ラベル番号		—	R/W
D8037	入出力モジュール接続台数		I/O 初期化時	R

アドレス	内容	更新のタイミング	R/W
D8038 D8039	リザーブ	—	—
D8040	スレーブ番号 (ポート4)	—	R/W
D8041	スレーブ番号 (ポート5)	—	R/W
D8042	スレーブ番号 (ポート6)	—	R/W
D8043	スレーブ番号 (ポート7)	—	R/W
D8044	スレーブ番号 (ポート8)	—	R/W
D8045	スレーブ番号 (ポート9)	—	R/W
D8046 } D8051	リザーブ	—	—
D8052	J1939通信エラーコード	毎スキャン	R/W
D8053 } D8055	リザーブ	—	—
D8056	電池電圧	—	R
D8057	アナログボリューム (AIO)	毎スキャン	R
D8058	内蔵アナログ入力 (AI1)	毎スキャン	R
D8059	アナログ入力ステータス AIO	毎スキャン	R
D8060	アナログ入力ステータス AI1	毎スキャン	R
D8061 } D8066	リザーブ	—	—
D8067	バックライト点灯時間	—	R/W
D8068	リザーブ	—	—
D8069	子局1通信ステータス/エラー (データリンク親局モード時) 子局通信ステータス/エラー (データリンク子局モード時)	エラー発生時	R
D8070	子局2通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8071	子局3通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8072	子局4通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8073	子局5通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8074	子局6通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8075	子局7通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8076	子局8通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8077	子局9通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8078	子局10通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8079	子局11通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8080	子局12通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8081	子局13通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8082	子局14通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8083	子局15通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8084	子局16通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8085	子局17通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8086	子局18通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8087	子局19通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8088	子局20通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8089	子局21通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8090	子局22通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8091	子局23通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8092	子局24通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8093	子局25通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8094	子局26通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R
D8095	子局27通信ステータス/エラー (データリンク親局モード時)	エラー発生時	R

アドレス	内容		更新のタイミング	R/W	
D8096	子局28通信ステータス/エラー (データリンク親局モード時)		エラー発生時	R	
D8097	子局29通信ステータス/エラー (データリンク親局モード時)		エラー発生時	R	
D8098	子局30通信ステータス/エラー (データリンク親局モード時)		エラー発生時	R	
D8099	子局31通信ステータス/エラー (データリンク親局モード時)		エラー発生時	R	
D8100	スレーブ番号 (ポート1)		—	R/W	
D8101	リザーブ		—	—	
D8102	スレーブ番号 (ポート2)		—	R/W	
D8103	スレーブ番号 (ポート3)		—	R/W	
D8104	制御ライン状態 (ポート1~5)		毎スキャン	R	
D8105	DR 制御ライン状態 (ポート1~5)		毎スキャン	R/W	
D8106	ER 制御ラインコントロール (ポート1~5)		毎スキャン	R/W	
D8107 } D8119	リザーブ		—	—	
D8120	HMIモジュール情報	機種ID/ステータス	—	R	
D8121		システムソフトウェアバージョン	—	R	
D8122	カートリッジスロット1情報	機種ID/ステータス	—	R	
D8123		システムソフトウェアバージョン	—	R	
D8124	カートリッジスロット2情報	機種ID/ステータス	—	R	
D8125		システムソフトウェアバージョン	—	R	
D8126	カートリッジスロット3情報	機種ID/ステータス	—	R	
D8127		システムソフトウェアバージョン	—	R	
D8128 } D8169	リザーブ		—	—	
D8170	アナログI/Oカートリッジ入出力 AI2/AQ2		毎スキャン	R	
D8171	アナログI/Oカートリッジ入出力 AI3/AQ3		毎スキャン	R	
D8172	アナログI/Oカートリッジステータス AI2/AQ2		毎スキャン	R	
D8173	アナログI/Oカートリッジステータス AI3/AQ3		毎スキャン	R	
D8174	アナログI/Oカートリッジ入出力 AI4/AQ4		毎スキャン	R	
D8175	アナログI/Oカートリッジ入出力 AI5/AQ5		毎スキャン	R	
D8176	アナログI/Oカートリッジステータス AI4/AQ4		毎スキャン	R	
D8177	アナログI/Oカートリッジステータス AI5/AQ5		毎スキャン	R	
D8178	アナログI/Oカートリッジ入出力 AI6/AQ6		毎スキャン	R	
D8179	アナログI/Oカートリッジ入出力 AI7/AQ7		毎スキャン	R	
D8180	アナログI/Oカートリッジステータス AI6/AQ6		毎スキャン	R	
D8181	アナログI/Oカートリッジステータス AI7/AQ7		毎スキャン	R	
D8182 } D8191	リザーブ		—	—	
D8192	高速カウンタ (グループ2/11)	上位ワード	現在値/周波数測定 (I1) 現在値	毎スキャン	R
D8193		下位ワード		毎スキャン	R
D8194		上位ワード	設定値	—	R/W
D8195		下位ワード		—	R/W
D8196	上位ワード	プリセット値	—	R/W	
D8197	下位ワード		—	R/W	
D8198	高速カウンタ (グループ6/17)	上位ワード	現在値/周波数測定 (I7) 現在値	毎スキャン	R
D8199		下位ワード		毎スキャン	R
D8200		上位ワード	設定値	—	R/W
D8201		下位ワード		—	R/W
D8202		上位ワード	プリセット値	—	R/W
D8203		下位ワード		—	R/W
D8204	制御ライン状態 (ポート6~9)		毎スキャン	R	

アドレス	内容		更新のタイミング	R/W
D8205	DR 制御ライン状態 (ポート6~9)		毎スキャン	RW
D8206	ER 制御ラインコントロール (ポート6~9)		毎スキャン	RW
D8207 } D8209	リザーブ		—	—
D8210	高速カウンタ (グループ1/10)	上位ワード	現在値/周波数測定 (10) 現在値	毎スキャン
D8211		下位ワード		毎スキャン
D8212		上位ワード	設定値	—
D8213		下位ワード		—
D8214	割込入力ジャンプ先ラベル番号 (17)		—	R/W
D8215	割込入力ジャンプ先ラベル番号 (10)		—	R/W
D8216	高速カウンタ (グループ1/10)	上位ワード	プリセット値	—
D8217		下位ワード		—
D8218	高速カウンタ (グループ3/13)	上位ワード	現在値/周波数測定 (13) 現在値	毎スキャン
D8219		下位ワード		毎スキャン
D8220		上位ワード	設定値	—
D8221		下位ワード		—
D8222	高速カウンタ (グループ4/14)	上位ワード	現在値/周波数測定 (14) 現在値	毎スキャン
D8223		下位ワード		毎スキャン
D8224		上位ワード	設定値	—
D8225		下位ワード		—
D8226	高速カウンタ (グループ5/16)	上位ワード	現在値/周波数測定 (16) 現在値	毎スキャン
D8227		下位ワード		毎スキャン
D8228		上位ワード	設定値	—
D8229		下位ワード		—
D8230 D8231	リザーブ		—	—
D8232	高速カウンタ (グループ5/16)	上位ワード	プリセット値	—
D8233		下位ワード		—
D8234	高速カウンタ (グループ3/13)	上位ワード	プリセット値	—
D8235		下位ワード		—
D8236	高速カウンタ (グループ4/14)	上位ワード	プリセット値	—
D8237		下位ワード		—
D8238	リザーブ		—	—
D8239	絶対位置管理ステータス		毎スキャン	R
D8240	絶対位置カウンタ1	上位ワード	絶対位置	毎スキャン
D8241		下位ワード		毎スキャン
D8242	絶対位置カウンタ2	上位ワード	絶対位置	毎スキャン
D8243		下位ワード		毎スキャン
D8244	絶対位置カウンタ3	上位ワード	絶対位置	毎スキャン
D8245		下位ワード		毎スキャン
D8246	絶対位置カウンタ4	上位ワード	絶対位置	毎スキャン
D8247		下位ワード		毎スキャン
D8248 D8249	リザーブ		—	—
D8250	SDメモ리카ード容量表示		毎スキャン	R
D8251	SDメモ리카ード空き容量表示		毎スキャン	R
D8252 D8253	リザーブ		—	—
D8254	SDメモ리카ード ダウンロード/アップロード実行情報		処理完了時	R
D8255	SDメモ리카ード ダウンロード/アップロード実行ステータス		処理完了時	R
D8256	SDメモ리카ードからのMQTT基本設定ダウンロード実行情報		処理完了時	R

アドレス	内容		更新のタイミング	R/W
D8257	SDメモ리카ードからのMQTT基本設定ダウンロード実行エラー情報		処理完了時	R
D8258 └ D8359	リザーブ		—	—
D8260	レシピブロック番号		—	R/W
D8261	レシピ実行ブロック番号		レシピ実行完了時	R
D8262	レシピ実行チャンネル番号		レシピ実行完了時	R
D8263	レシピ実行動作		レシピ実行完了時	R
D8264	レシピ実行ステータス		レシピ実行完了時	R
D8265	レシピ実行エラー情報		レシピ実行完了時	R
D8266	レシピ内蔵メモリ (ROM-領域1) 読み出し回数		レシピ実行完了時	R
D8267	レシピ内蔵メモリ (ROM-領域2) 読み出し回数		レシピ実行完了時	R
D8268	コネクション1のリモートホスト番号 (1~255)		—	R/W
D8269	コネクション2のリモートホスト番号 (1~255)		—	R/W
D8270	コネクション3のリモートホスト番号 (1~255)		—	R/W
D8271	コネクション4のリモートホスト番号 (1~255)		—	R/W
D8272	コネクション5のリモートホスト番号 (1~255)		—	R/W
D8273	コネクション6のリモートホスト番号 (1~255)		—	R/W
D8274	コネクション7のリモートホスト番号 (1~255)		—	R/W
D8275	コネクション8のリモートホスト番号 (1~255)		—	R/W
D8276 D8277	リザーブ		—	—
D8278	通信モード情報	コネクション1~4	—	R
D8279		コネクション5~8	—	R
D8280 └ D8283	リザーブ		—	—
D8284	通信モード情報 (HMIコネクション)	HMIコネクション1~4	—	R
D8285		HMIコネクション5~8	—	R
D8286 └ D8302	リザーブ		—	—
D8303	CPUモジュール Ethernetポート1 IP設定 / DNS設定切り替え		—	R/W
D8304	CPUモジュール Ethernetポート1 IPアドレス (書き込み専用)		—	W
D8305			—	W
D8306			—	W
D8307			—	W
D8308			—	W
D8309	CPUモジュール Ethernetポート1 サブネットマスク (書き込み専用)		—	W
D8310			—	W
D8311			—	W
D8312			—	W
D8313	CPUモジュール Ethernetポート1 デフォルトゲートウェイ (書き込み専用)		—	W
D8314			—	W
D8315			—	W
D8316			—	W
D8317			—	W
D8318	CPUモジュール Ethernetポート1 優先DNSサーバー (書き込み専用)		—	W
D8319			—	W
D8320			—	W
D8321	CPUモジュール Ethernetポート1 代替DNSサーバー (書き込み専用)		—	W
D8322			—	W
D8323			—	W
D8323			—	W

アドレス	内容	更新のタイミング	R/W
D8324	CPUモジュール Ethernetポート1 MACアドレス (現在値読み出し専用)	1sごと	R
D8325		1sごと	R
D8326		1sごと	R
D8327		1sごと	R
D8328		1sごと	R
D8329		1sごと	R
D8330	CPUモジュール Ethernetポート1 IPアドレス (現在値読み出し専用)	1sごと	R
D8331		1sごと	R
D8332		1sごと	R
D8333		1sごと	R
D8334	CPUモジュール Ethernetポート1 サブネットマスク (現在値読み出し専用)	1sごと	R
D8335		1sごと	R
D8336		1sごと	R
D8337		1sごと	R
D8338	CPUモジュール Ethernetポート1 デフォルトゲートウェイ (現在値読み出し専用)	1sごと	R
D8339		1sごと	R
D8340		1sごと	R
D8341		1sごと	R
D8342	CPUモジュール Ethernetポート1 優先DNSサーバー (現在値読み出し専用)	1sごと	R
D8343		1sごと	R
D8344		1sごと	R
D8345		1sごと	R
D8346	CPUモジュール Ethernetポート1 代替DNSサーバー (現在値読み出し専用)	1sごと	R
D8347		1sごと	R
D8348		1sごと	R
D8349		1sごと	R
D8350	コネクション1接続IPアドレス	1sごと	R
D8351		1sごと	R
D8352		1sごと	R
D8353	コネクション2接続IPアドレス	1sごと	R
D8354		1sごと	R
D8355		1sごと	R
D8356		1sごと	R
D8357	コネクション3接続IPアドレス	1sごと	R
D8358		1sごと	R
D8359		1sごと	R
D8360		1sごと	R
D8361	コネクション4接続IPアドレス	1sごと	R
D8362		1sごと	R
D8363		1sごと	R
D8364		1sごと	R
D8365	コネクション5接続IPアドレス	1sごと	R
D8366		1sごと	R
D8367		1sごと	R
D8368		1sごと	R
D8369	コネクション6接続IPアドレス	1sごと	R
D8370		1sごと	R
D8371		1sごと	R
D8372		1sごと	R
D8373		1sごと	R

アドレス	内容	更新のタイミング	R/W	
D8374	コネクション7接続IPアドレス	1sごと	R	
D8375		1sごと	R	
D8376		1sごと	R	
D8377		1sごと	R	
D8378	コネクション8接続IPアドレス	1sごと	R	
D8379		1sごと	R	
D8380		1sごと	R	
D8381		1sごと	R	
D8382	HMIモジュール MACアドレス (現在値読み出し専用)	1sごと	R	
D8383		1sごと	R	
D8384		1sごと	R	
D8385		1sごと	R	
D8386		1sごと	R	
D8387		1sごと	R	
D8388	HMIモジュール IP アドレス (現在値読み出し専用)	1sごと	R	
D8389		1sごと	R	
D8390		1sごと	R	
D8391		1sごと	R	
D8392	HMIモジュール サブネットマスク (現在値読み出し専用)	1sごと	R	
D8393		1sごと	R	
D8394		1sごと	R	
D8395		1sごと	R	
D8396	HMIモジュール デフォルトゲートウェイ (現在値読み出し専用)	1sごと	R	
D8397		1sごと	R	
D8398		1sごと	R	
D8399		1sごと	R	
D8400	HMIモジュール 優先DNSサーバー (現在値読み出し専用)	1sごと	R	
D8401		1sごと	R	
D8402		1sごと	R	
D8403		1sごと	R	
D8404	HMIモジュール 代替DNSサーバー (現在値読み出し専用)	1sごと	R	
D8405		1sごと	R	
D8406		1sごと	R	
D8407		1sごと	R	
D8408 └ D8412	リザーブ	—	—	
D8413	タイムゾーンオフセット	—	R/W	
D8414	SNTP 動作ステータス	—	R	
D8415	SNTP アクセス経過時間	—	R	
D8416 └ D8428	リザーブ	—	—	
D8429	HMIモジュール コネクション情報参照	コネクション番号	—	R/W
D8430		接続IPアドレス	1sごと	R
D8431			1sごと	R
D8432			1sごと	R
D8433			1sごと	R
D8434	接続ポート番号	1sごと	R	
D8435 D8436	リザーブ	—	—	

アドレス	内容	更新のタイミング	R/W
D8437	HMIモジュール IPアドレス (書き込み専用)	—	W
D8438		—	W
D8439		—	W
D8440		—	W
D8441	HMIモジュール サブネットマスク (書き込み専用)	—	W
D8442		—	W
D8443		—	W
D8444		—	W
D8445	HMIモジュール デフォルトゲートウェイ (書き込み専用)	—	W
D8446		—	W
D8447		—	W
D8448		—	W
D8449	HMIモジュール 優先DNSサーバー (書き込み専用)	—	W
D8450		—	W
D8451		—	W
D8452		—	W
D8453	HMIモジュール 代替DNSサーバー (書き込み専用)	—	W
D8454		—	W
D8455		—	W
D8456		—	W
D8457	EMAIL命令詳細エラー情報 (HMI-Ethernetポート)	—	R
D8458 ↳ D8469	リザーブ	—	—
D8470	増設モジュール スロット1情報	機種ID/ステータス	R
D8471		システムソフトウェアバージョン/位置情報	R
D8472	増設モジュール スロット2情報	機種ID/ステータス	R
D8473		システムソフトウェアバージョン/位置情報	R
D8474	増設モジュール スロット3情報	機種ID/ステータス	R
D8475		システムソフトウェアバージョン/位置情報	R
D8476	増設モジュール スロット4情報	機種ID/ステータス	R
D8477		システムソフトウェアバージョン/位置情報	R
D8478	増設モジュール スロット5情報	機種ID/ステータス	R
D8479		システムソフトウェアバージョン/位置情報	R
D8480	増設モジュール スロット6情報	機種ID/ステータス	R
D8481		システムソフトウェアバージョン/位置情報	R
D8482	増設モジュール スロット7情報	機種ID/ステータス	R
D8483		システムソフトウェアバージョン/位置情報	R
D8484	増設モジュール スロット8情報	機種ID/ステータス	R
D8485		システムソフトウェアバージョン/位置情報	R
D8486	増設モジュール スロット9情報	機種ID/ステータス	R
D8487		システムソフトウェアバージョン/位置情報	R
D8488	増設モジュール スロット10情報	機種ID/ステータス	R
D8489		システムソフトウェアバージョン/位置情報	R
D8490	増設モジュール スロット11情報	機種ID/ステータス	R
D8491		システムソフトウェアバージョン/位置情報	R
D8492	増設モジュール スロット12情報	機種ID/ステータス	R
D8493		システムソフトウェアバージョン/位置情報	R
D8494	増設モジュール スロット13情報	機種ID/ステータス	R
D8495		システムソフトウェアバージョン/位置情報	R
D8496	増設モジュール スロット14情報	機種ID/ステータス	R
D8497		システムソフトウェアバージョン/位置情報	R

アドレス	内容		更新のタイミング	R/W
D8498	増設モジュール スロット15情報	機種ID/ステータス	—	R
D8499		システムソフトウェアバージョン/位置情報	—	R
D8500	増設モジュール スロット16情報	機種ID/ステータス	—	R
D8501		システムソフトウェアバージョン/位置情報	—	R
D8502	増設モジュール スロット17情報	機種ID/ステータス	—	R
D8503		システムソフトウェアバージョン/位置情報	—	R
D8504	増設モジュール スロット18情報	機種ID/ステータス	—	R
D8505		システムソフトウェアバージョン/位置情報	—	R
D8506	増設モジュール スロット19情報	機種ID/ステータス	—	R
D8507		システムソフトウェアバージョン/位置情報	—	R
D8508	増設モジュール スロット20情報	機種ID/ステータス	—	R
D8509		システムソフトウェアバージョン/位置情報	—	R
D8510	増設モジュール スロット21情報	機種ID/ステータス	—	R
D8511		システムソフトウェアバージョン/位置情報	—	R
D8512	増設モジュール スロット22情報	機種ID/ステータス	—	R
D8513		システムソフトウェアバージョン/位置情報	—	R
D8514	増設モジュール スロット23情報	機種ID/ステータス	—	R
D8515		システムソフトウェアバージョン/位置情報	—	R
D8516	増設モジュール スロット24情報	機種ID/ステータス	—	R
D8517		システムソフトウェアバージョン/位置情報	—	R
D8518	増設モジュール スロット25情報	機種ID/ステータス	—	R
D8519		システムソフトウェアバージョン/位置情報	—	R
D8520	増設モジュール スロット26情報	機種ID/ステータス	—	R
D8521		システムソフトウェアバージョン/位置情報	—	R
D8522	増設モジュール スロット27情報	機種ID/ステータス	—	R
D8523		システムソフトウェアバージョン/位置情報	—	R
D8524	増設モジュール スロット28情報	機種ID/ステータス	—	R
D8525		システムソフトウェアバージョン/位置情報	—	R
D8526	増設モジュール スロット29情報	機種ID/ステータス	—	R
D8527		システムソフトウェアバージョン/位置情報	—	R
D8528	増設モジュール スロット30情報	機種ID/ステータス	—	R
D8529		システムソフトウェアバージョン/位置情報	—	R
D8530	増設モジュール スロット31情報	機種ID/ステータス	—	R
D8531		システムソフトウェアバージョン/位置情報	—	R
D8532	増設モジュール スロット32情報	機種ID/ステータス	—	R
D8533		システムソフトウェアバージョン/位置情報	—	R
D8534	増設モジュール スロット33情報	機種ID/ステータス	—	R
D8535		システムソフトウェアバージョン/位置情報	—	R
D8536	増設モジュール スロット34情報	機種ID/ステータス	—	R
D8537		システムソフトウェアバージョン/位置情報	—	R
D8538	増設モジュール スロット35情報	機種ID/ステータス	—	R
D8539		システムソフトウェアバージョン/位置情報	—	R
D8530	増設モジュール スロット36情報	機種ID/ステータス	—	R
D8541		システムソフトウェアバージョン/位置情報	—	R
D8542	増設モジュール スロット37情報	機種ID/ステータス	—	R
D8543		システムソフトウェアバージョン/位置情報	—	R
D8544	増設モジュール スロット38情報	機種ID/ステータス	—	R
D8545		システムソフトウェアバージョン/位置情報	—	R
D8546	増設モジュール スロット39情報	機種ID/ステータス	—	R
D8547		システムソフトウェアバージョン/位置情報	—	R

アドレス	内容		更新のタイミング	R/W
D8548	増設モジュール スロット40情報	機種ID/ステータス	—	R
D8549		システムソフトウェアバージョン/位置情報	—	R
D8550	増設モジュール スロット41情報	機種ID/ステータス	—	R
D8551		システムソフトウェアバージョン/位置情報	—	R
D8552	増設モジュール スロット42情報	機種ID/ステータス	—	R
D8553		システムソフトウェアバージョン/位置情報	—	R
D8554	増設モジュール スロット43情報	機種ID/ステータス	—	R
D8555		システムソフトウェアバージョン/位置情報	—	R
D8556	増設モジュール スロット44情報	機種ID/ステータス	—	R
D8557		システムソフトウェアバージョン/位置情報	—	R
D8558	増設モジュール スロット45情報	機種ID/ステータス	—	R
D8559		システムソフトウェアバージョン/位置情報	—	R
D8560	増設モジュール スロット46情報	機種ID/ステータス	—	R
D8561		システムソフトウェアバージョン/位置情報	—	R
D8562	増設モジュール スロット47情報	機種ID/ステータス	—	R
D8563		システムソフトウェアバージョン/位置情報	—	R
D8564	増設モジュール スロット48情報	機種ID/ステータス	—	R
D8565		システムソフトウェアバージョン/位置情報	—	R
D8566	増設モジュール スロット49情報	機種ID/ステータス	—	R
D8567		システムソフトウェアバージョン/位置情報	—	R
D8568	増設モジュール スロット50情報	機種ID/ステータス	—	R
D8569		システムソフトウェアバージョン/位置情報	—	R
D8570	増設モジュール スロット51情報	機種ID/ステータス	—	R
D8571		システムソフトウェアバージョン/位置情報	—	R
D8572	増設モジュール スロット52情報	機種ID/ステータス	—	R
D8573		システムソフトウェアバージョン/位置情報	—	R
D8574	増設モジュール スロット53情報	機種ID/ステータス	—	R
D8575		システムソフトウェアバージョン/位置情報	—	R
D8576	増設モジュール スロット54情報	機種ID/ステータス	—	R
D8577		システムソフトウェアバージョン/位置情報	—	R
D8578	増設モジュール スロット55情報	機種ID/ステータス	—	R
D8579		システムソフトウェアバージョン/位置情報	—	R
D8580	増設モジュール スロット56情報	機種ID/ステータス	—	R
D8581		システムソフトウェアバージョン/位置情報	—	R
D8582	増設モジュール スロット57情報	機種ID/ステータス	—	R
D8583		システムソフトウェアバージョン/位置情報	—	R
D8584	増設モジュール スロット58情報	機種ID/ステータス	—	R
D8585		システムソフトウェアバージョン/位置情報	—	R
D8586	増設モジュール スロット59情報	機種ID/ステータス	—	R
D8587		システムソフトウェアバージョン/位置情報	—	R
D8588	増設モジュール スロット60情報	機種ID/ステータス	—	R
D8589		システムソフトウェアバージョン/位置情報	—	R
D8590	増設モジュール スロット61情報	機種ID/ステータス	—	R
D8591		システムソフトウェアバージョン/位置情報	—	R
D8592	増設モジュール スロット62情報	機種ID/ステータス	—	R
D8593		システムソフトウェアバージョン/位置情報	—	R
D8594	増設モジュール スロット63情報	機種ID/ステータス	—	R
D8595		システムソフトウェアバージョン/位置情報	—	R
D8596	増設拡張モジュール分離型マスター スロット情報	機種ID/ステータス	—	R
D8597		システムソフトウェアバージョン/ 増設拡張モジュール分離型スレーブの接続台数	—	R

アドレス	内容		更新のタイミング	R/W
D8598	増設拡張モジュール分離型スレーブ (ノード1) スロット情報	機種ID/ステータス	—	R
D8599		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8600	増設拡張モジュール分離型スレーブ (ノード2) スロット情報	機種ID/ステータス	—	R
D8601		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8602	増設拡張モジュール分離型スレーブ (ノード3) スロット情報	機種ID/ステータス	—	R
D8603		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8604	増設拡張モジュール分離型スレーブ (ノード4) スロット情報	機種ID/ステータス	—	R
D8605		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8606	増設拡張モジュール分離型スレーブ (ノード5) スロット情報	機種ID/ステータス	—	R
D8607		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8608	増設拡張モジュール分離型スレーブ (ノード6) スロット情報	機種ID/ステータス	—	R
D8609		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8610	増設拡張モジュール分離型スレーブ (ノード7) スロット情報	機種ID/ステータス	—	R
D8611		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8612	増設拡張モジュール分離型スレーブ (ノード8) スロット情報	機種ID/ステータス	—	R
D8613		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8614	増設拡張モジュール分離型スレーブ (ノード9) スロット情報	機種ID/ステータス	—	R
D8615		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8616	増設拡張モジュール分離型スレーブ (ノード10) スロット情報	機種ID/ステータス	—	R
D8617		システムソフトウェアバージョン/ 増設モジュールの接続情報	—	R
D8618	増設拡張モジュール分離型I/Oリフレッシュ時間 現在値		毎スキャン	R
D8619	増設拡張モジュール分離型I/Oリフレッシュ時間 最大値		毎スキャン	R
D8620 └ D8629	リザーブ		—	—
D8630	CPUモジュール Ethernetポート2 IP設定 / DNS設定切り替え		—	R/W
D8631	CPUモジュール Ethernetポート2 IPアドレス (書き込み専用)		—	W
D8632			—	W
D8633			—	W
D8634			—	W
D8635	CPUモジュール Ethernetポート2 サブネットマスク (書き込み専用)		—	W
D8636			—	W
D8637			—	W
D8638			—	W
D8639	CPUモジュール Ethernetポート2 デフォルトゲートウェイ (書き込み専用)		—	W
D8640			—	W
D8641			—	W
D8642			—	W
D8643	CPUモジュール Ethernetポート2 優先DNSサーバー (書き込み専用)		—	W
D8644			—	W
D8645			—	W
D8646			—	W

アドレス	内容	更新のタイミング	R/W
D8647	CPUモジュール Ethernetポート2 代替DNSサーバー (書き込み専用)	—	W
D8648		—	W
D8649		—	W
D8650		—	W
D8651	CPUモジュール Ethernetポート2 MACアドレス (現在値読み出し専用)	1s ごと	R
D8652		1s ごと	R
D8653		1s ごと	R
D8654		1s ごと	R
D8655		1s ごと	R
D8656		1s ごと	R
D8657	CPUモジュール Ethernetポート2 IP アドレス (現在値読み出し専用)	1s ごと	R
D8658		1s ごと	R
D8659		1s ごと	R
D8660		1s ごと	R
D8661	CPUモジュール Ethernetポート2 サブネットマスク (現在値読み出し専用)	1s ごと	R
D8662		1s ごと	R
D8663		1s ごと	R
D8664		1s ごと	R
D8665	CPUモジュール Ethernetポート2 デフォルトゲートウェイ (現在値読み出し専用)	1s ごと	R
D8666		1s ごと	R
D8667		1s ごと	R
D8668		1s ごと	R
D8669	CPUモジュール Ethernetポート2 優先DNSサーバー (現在値読み出し専用)	1s ごと	R
D8670		1s ごと	R
D8671		1s ごと	R
D8672		1s ごと	R
D8673	CPUモジュール Ethernetポート2 代替DNSサーバー (現在値読み出し専用)	1s ごと	R
D8674		1s ごと	R
D8675		1s ごと	R
D8676		1s ごと	R
D8677	コネクション9接続IPアドレス	1s ごと	R
D8678		1s ごと	R
D8679		1s ごと	R
D8680		1s ごと	R
D8681	コネクション10接続IPアドレス	1s ごと	R
D8682		1s ごと	R
D8683		1s ごと	R
D8684		1s ごと	R
D8685	コネクション11接続IPアドレス	1s ごと	R
D8686		1s ごと	R
D8687		1s ごと	R
D8688		1s ごと	R
D8689	コネクション12接続IPアドレス	1s ごと	R
D8690		1s ごと	R
D8691		1s ごと	R
D8692		1s ごと	R
D8693	コネクション13接続IPアドレス	1s ごと	R
D8694		1s ごと	R
D8695		1s ごと	R
D8696		1s ごと	R

アドレス	内容	更新のタイミング	R/W
D8697	コネクション14接続IPアドレス	1sごと	R
D8698		1sごと	R
D8699		1sごと	R
D8700		1sごと	R
D8701	コネクション15接続IPアドレス	1sごと	R
D8702		1sごと	R
D8703		1sごと	R
D8704		1sごと	R
D8705	コネクション16接続IPアドレス	1sごと	R
D8706		1sごと	R
D8707		1sごと	R
D8708		1sごと	R
D8709 ┆ D8716	リザーブ	—	—
D8717	制御ライン状態 (ポート10~13)	毎スキャン	R
D8718	DR 制御ライン状態 (ポート10~13)	毎スキャン	RW
D8719	ER 制御ラインコントロール (ポート10~13)	毎スキャン	RW
D8720	制御ライン状態 (ポート14~17)	毎スキャン	R
D8721	DR 制御ライン状態 (ポート14~17)	毎スキャン	RW
D8722	ER 制御ラインコントロール (ポート14~17)	毎スキャン	RW
D8723	制御ライン状態 (ポート18~21)	毎スキャン	R
D8724	DR 制御ライン状態 (ポート18~21)	毎スキャン	RW
D8725	ER 制御ラインコントロール (ポート18~21)	毎スキャン	RW
D8726	制御ライン状態 (ポート22~25)	毎スキャン	R
D8727	DR 制御ライン状態 (ポート22~25)	毎スキャン	RW
D8728	ER 制御ラインコントロール (ポート22~25)	毎スキャン	RW
D8729	制御ライン状態 (ポート26~29)	毎スキャン	R
D8730	DR 制御ライン状態 (ポート26~29)	毎スキャン	RW
D8731	ER 制御ラインコントロール (ポート26~29)	毎スキャン	RW
D8732	制御ライン状態 (ポート30~33)	毎スキャン	R
D8733	DR 制御ライン状態 (ポート30~33)	毎スキャン	RW
D8734	ER 制御ラインコントロール (ポート30~33)	毎スキャン	RW
D8735	スレーブ番号 (ポート10)	—	R/W
D8736	スレーブ番号 (ポート11)	—	R/W
D8737	スレーブ番号 (ポート12)	—	R/W
D8738	スレーブ番号 (ポート13)	—	R/W
D8739	スレーブ番号 (ポート14)	—	R/W
D8740	スレーブ番号 (ポート15)	—	R/W
D8741	スレーブ番号 (ポート16)	—	R/W
D8742	スレーブ番号 (ポート17)	—	R/W
D8743	スレーブ番号 (ポート18)	—	R/W
D8744	スレーブ番号 (ポート19)	—	R/W
D8745	スレーブ番号 (ポート20)	—	R/W
D8746	スレーブ番号 (ポート21)	—	R/W
D8747	スレーブ番号 (ポート22)	—	R/W
D8748	スレーブ番号 (ポート23)	—	R/W
D8749	スレーブ番号 (ポート24)	—	R/W
D8750	スレーブ番号 (ポート25)	—	R/W
D8751	スレーブ番号 (ポート26)	—	R/W
D8752	スレーブ番号 (ポート27)	—	R/W
D8753	スレーブ番号 (ポート28)	—	R/W

アドレス	内容	更新のタイミング	R/W
D8754	スレーブ番号 (ポート29)	—	R/W
D8755	スレーブ番号 (ポート30)	—	R/W
D8756	スレーブ番号 (ポート31)	—	R/W
D8757	スレーブ番号 (ポート32)	—	R/W
D8758	スレーブ番号 (ポート33)	—	R/W
D8759	EMAIL命令詳細エラー情報 (Ethernetポート1)	—	R
D8760	通信モード情報	コネクション9~12	R
D8761		コネクション13~16	R
D8762 } D8773	リザーブ	—	—
D8774	コネクション9のリモートホスト番号 (1~255)	—	R/W
D8775	コネクション10のリモートホスト番号 (1~255)	—	R/W
D8776	コネクション11のリモートホスト番号 (1~255)	—	R/W
D8777	コネクション12のリモートホスト番号 (1~255)	—	R/W
D8778	コネクション13のリモートホスト番号 (1~255)	—	R/W
D8779	コネクション14のリモートホスト番号 (1~255)	—	R/W
D8780	コネクション15のリモートホスト番号 (1~255)	—	R/W
D8781	コネクション16のリモートホスト番号 (1~255)	—	R/W
D8782	BACnet運転状態	—	R
D8783	BACnetエラー情報	—	R
D8784 } D8789	リザーブ	—	—
D8790	EtherNet/IP運転状態	—	R
D8791	EtherNet/IPエラー情報	—	R
D8792 } D8819	リザーブ	—	—
D8820	SDメモリカードからのサーバー機能用ファイルダウンロード実行情報	処理完了時	R
D8821	SDメモリカードからのサーバー機能用ファイルダウンロード実行エラー情報	処理完了時	R
D8822	Webサーバーの現在のコネクション数	約 1s 周期	R
D8823 } D8899	リザーブ	—	—

特殊データレジスタ補足説明

■ D8000：入力点数

FC6A 形の入力点数を書き込みます。
CPU モジュールと増設モジュールの入力点数の合計値を書き込みます。

■ D8001：出力点数

FC6A 形の出力点数を書き込みます。
CPU モジュールと増設モジュールの出力点数の合計値を書き込みます。

■ D8002：CPU モジュール機種情報

CPU モジュールの機種情報を書き込みます。
0 (00h)：16 点タイプ
1 (01h)：24 点タイプ
2 (02h)：40 点タイプ (All-in-One CPU モジュール)
18 (12h)：40 点タイプ (CAN J1939 All-in-One CPU モジュール)
32 (20h)：Plus16 点タイプ
33 (21h)：Plus32 点タイプ

■ D8005：一般エラーコード

FC6A 形の一般エラー情報を書き込みます。一般エラーが発生すると、発生したエラーに対応するビットを ON します。また、ユーザープログラムを使用して、最上位ビットに“1”を書き込むことで、一般エラーおよびユーザープログラム実行エラーをクリアできます。一般エラーコードの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 13 章 一般エラー」を参照してください。

■ D8006：ユーザープログラム実行エラーコード

FC6A 形のユーザープログラム実行エラー情報を書き込みます。ユーザープログラム実行エラーが発生すると、エラー内容に対応するエラーコードが書き込まれます。ユーザープログラム実行エラーの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 13 章 ユーザープログラム実行エラー一覧」を参照してください。

■ D8008～D8021：カレンダー・時計データ

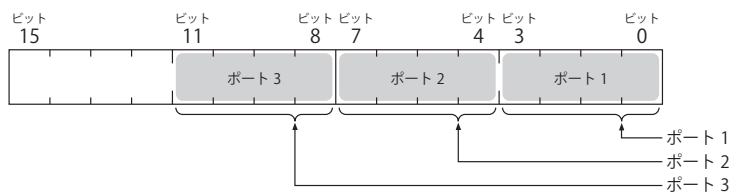
カレンダー・時計データの内蔵時計からの読み出しや、内蔵時計への書き込みに使用します。カレンダー・時計データの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 時計機能」を参照してください。

■ D8022～D8025：スキャンタイムデータ

スキャンタイムの確認や、スキャンタイムのコンスタント設定を行う特殊データレジスタです。スキャンタイムの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 コンスタントスキャン」を参照してください。

■ D8026：通信モード情報（ポート 1～3）

ポート 1～3 の通信モードを示します。
デバイス内の各通信ポートの割り当て（ビットアサイン）は、次のとおりです。



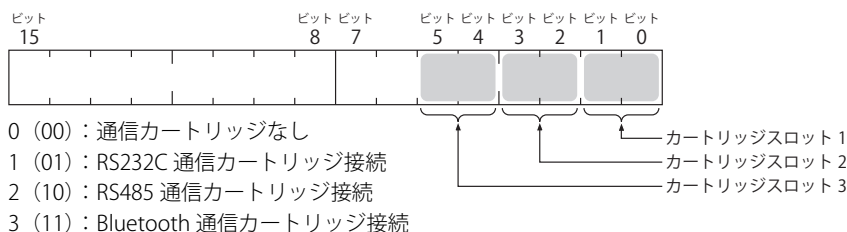
- 0 (0000)：メンテナンス通信
- 1 (0001)：ユーザー通信
- 2 (0010)：Modbus RTU マスター
- 3 (0011)：Modbus RTU スレーブ
- 4 (0100)：データリンク通信

■ D8029：システムバージョン情報

CPU モジュールのシステムソフトウェアのバージョン番号を書き込みます。

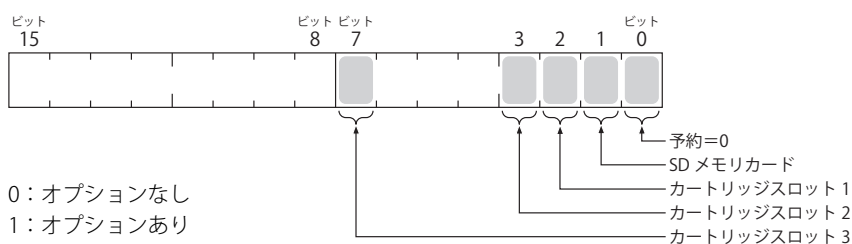
■ D8030：通信カートリッジ情報

カートリッジスロットへの通信カートリッジの接続状況を示します。
デバイス内の各カートリッジスロットの割り当て（ビットアサイン）は、次のとおりです。



■ D8031：オプション接続情報

オプション接続情報を書き込みます。
デバイス内の割り当て（ビットアサイン）は、次のとおりです。



■ D8032～D8035、D8214、D8215：割込入力ジャンプ先ラベル番号

割込入力のジャンプ先ラベル番号を書き込みます。割込入力を使用するとき、割込入力に割り当てられた特殊データレジスタに対応するラベル番号を格納してください。割込入力の詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 割込入力」を参照してください。

D8032 = I1
D8033 = I3
D8034 = I4
D8035 = I6
D8214 = I7
D8215 = I0

■ D8036：タイマ割込ジャンプ先ラベル番号

タイマ割込発生時のジャンプ先ラベル番号を書き込みます。タイマ割込を使用するとき、対応するラベル番号を格納してください。

タイマ割込の詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 タイマ割込」を参照してください。

■ D8037：増設モジュール接続台数

CPU モジュールに接続されている増設モジュール（I/O モジュール、温調モジュールおよび通信モジュール）の台数を書き込まれます。

■ D8052：J1939 通信エラーコード

J1939 通信でエラーが発生すると、エラーコードが書き込まれます。J1939 通信エラーコードの詳細は、「第 8 章 J1939 通信エラーコード (D8052)」(8-5 頁) を参照してください。

■ D8056：電池電圧

バックアップ用電池の電池電圧の計測結果を 1mV 単位で書き込みます。

0：計測エラー

■ D8057：アナログボリューム (AI0)

アナログボリュームの値をデジタル値に変換して書き込みます。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 アナログボリューム」を参照してください。

■ D8058：内蔵アナログ入力 (AI1)

アナログ入力のアナログ入力値 (DC0～10V) をデジタル値に変換して書き込みます。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 内蔵アナログ入力」を参照してください。

■ D8059：アナログ入力ステータス AI0

アナログボリュームのアナログ入力ステータスを書き込みます。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 アナログボリューム」を参照してください。

■ **D8060：アナログ入力ステータス AI1**

アナログ入力のアナログ入力ステータスを書き込みます。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第5章 内蔵アナログ入力」(を参照してください)。

■ **D8067：バックライト点灯時間**

HMI モジュールのバックライトの点灯時間を書き込みます。バックライト点灯時間は、D8067 の値を変更することで 1 ～ 65535s の範囲で設定できます。D8067 の値を 0s とした場合、バックライトは常時点灯します。バックライト点灯時間は HMI モジュールの環境設定で変更できます。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第7章 ●LCD のバックライト点灯時間の設定」を参照してください。

■ **D8069 ～ D8099：子局 (1 ～ 31) 通信ステータス / エラー**

データリンク通信に使用する特殊データレジスタです。詳細は、「第7章 親局側データリンク通信ステータス」(7-5 頁) を参照してください。

■ **D8040 ～ D8045、D8100、D8102、D8103、D8735 ～ D8758：スレーブ番号 (ポート 1 ～ 33)**

ポート 1 ～ 33 の通信モードがメンテナンス通信、Modbus RTU スレーブまたはデータリンク通信のとき、スレーブ番号を書き込みます。[ファンクション設定] で対応するデバイスの値を変更することでスレーブ番号を変更できます。

D8100 =ポート 1 スレーブ番号
 D8102 =ポート 2 スレーブ番号
 D8103 =ポート 3 スレーブ番号

D8742 =ポート 17 スレーブ番号
 D8743 =ポート 18 スレーブ番号
 D8744 =ポート 19 スレーブ番号
 D8745 =ポート 20 スレーブ番号
 D8746 =ポート 21 スレーブ番号
 D8747 =ポート 22 スレーブ番号
 D8748 =ポート 23 スレーブ番号
 D8749 =ポート 24 スレーブ番号
 D8750 =ポート 25 スレーブ番号
 D8751 =ポート 26 スレーブ番号
 D8752 =ポート 27 スレーブ番号
 D8753 =ポート 28 スレーブ番号
 D8754 =ポート 29 スレーブ番号
 D8755 =ポート 30 スレーブ番号
 D8756 =ポート 31 スレーブ番号
 D8757 =ポート 32 スレーブ番号
 D8758 =ポート 33 スレーブ番号

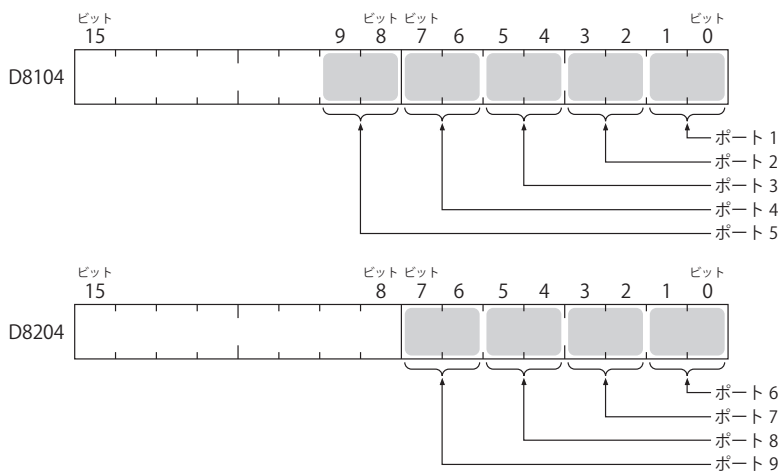
D8040 =ポート 4 スレーブ番号
 D8041 =ポート 5 スレーブ番号
 D8042 =ポート 6 スレーブ番号
 D8043 =ポート 7 スレーブ番号
 D8044 =ポート 8 スレーブ番号
 D8045 =ポート 9 スレーブ番号
 D8735 =ポート 10 スレーブ番号
 D8736 =ポート 11 スレーブ番号
 D8737 =ポート 12 スレーブ番号
 D8738 =ポート 13 スレーブ番号
 D8739 =ポート 14 スレーブ番号
 D8740 =ポート 15 スレーブ番号
 D8741 =ポート 16 スレーブ番号

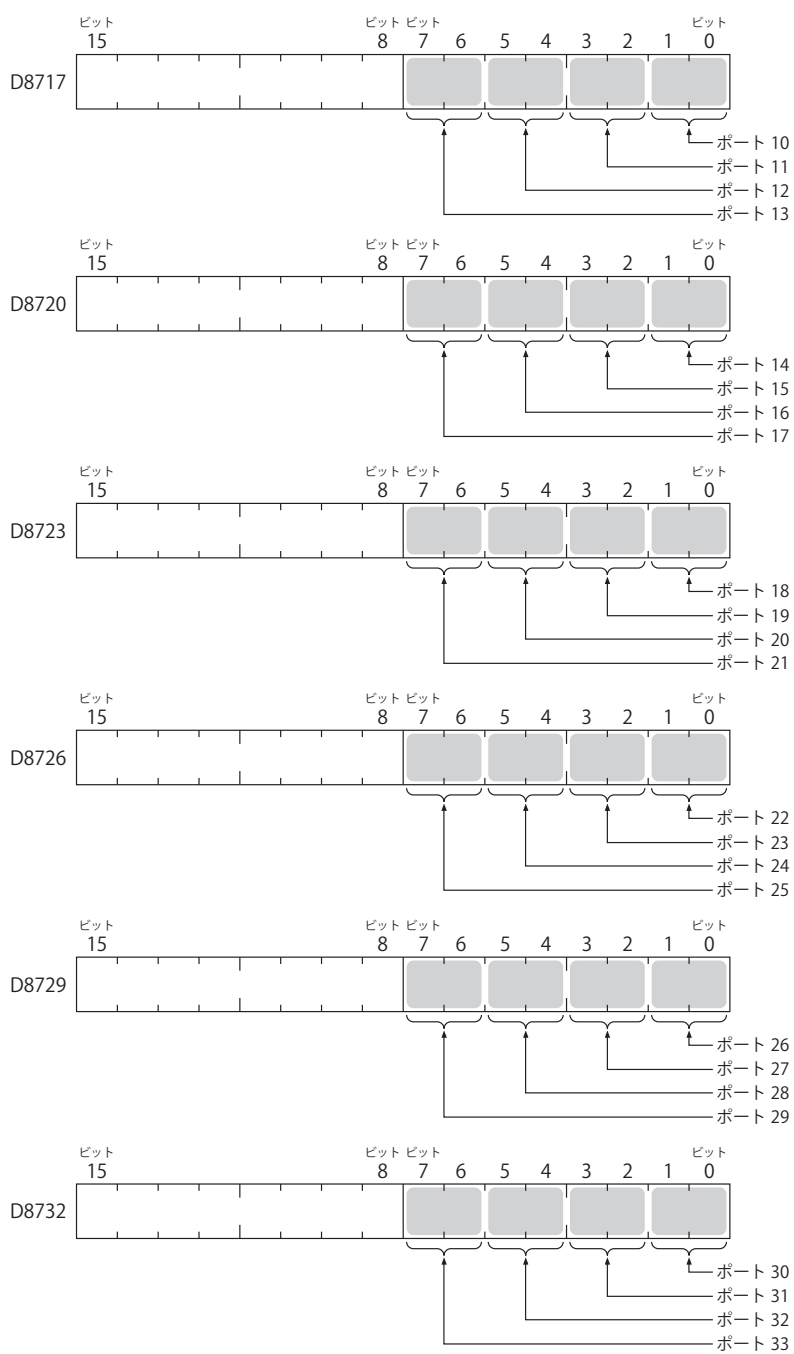
各通信モードの詳細は、以下を参照してください。

- ・メンテナンス通信：「第4章 メンテナンス通信」(4-1 頁)
- ・Modbus RTU スレーブ：「第6章 Modbus 通信」(6-1 頁)
- ・データリンク通信：「第7章 データリンク通信」(7-1 頁)

■ **D8104、D8204、D8717、D8720、D8723、D8726、D8729、D8732：制御ライン状態 (ポート 1 ～ 33)**

DR、ER の各制御ラインの信号状態が書き込まれます。STOP 中および RUN 中の END 処理で更新します。デバイス内の各通信ポートの割り当て (ビットアサイン) は、次のとおりです。

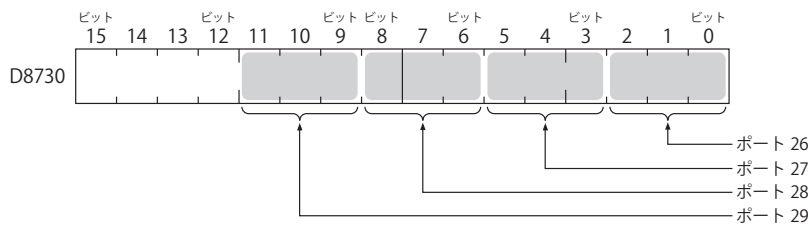
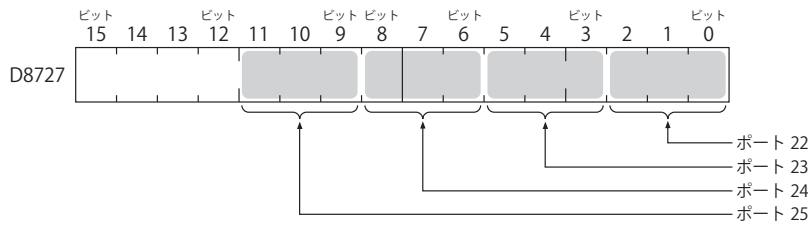
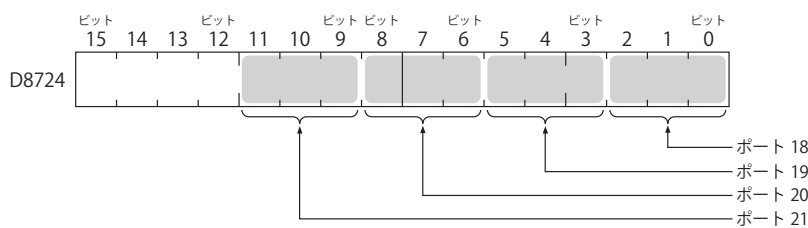
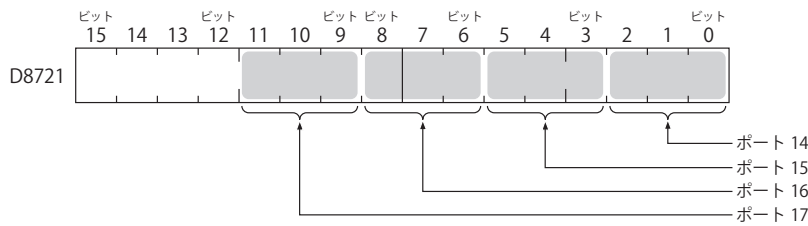
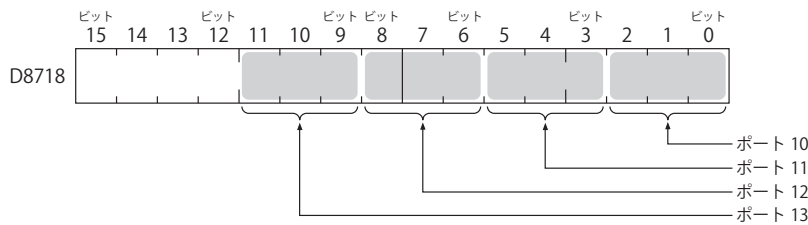
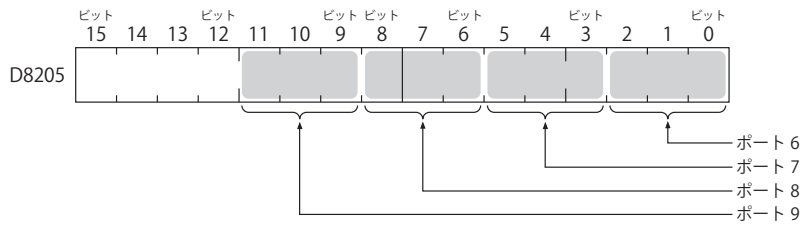
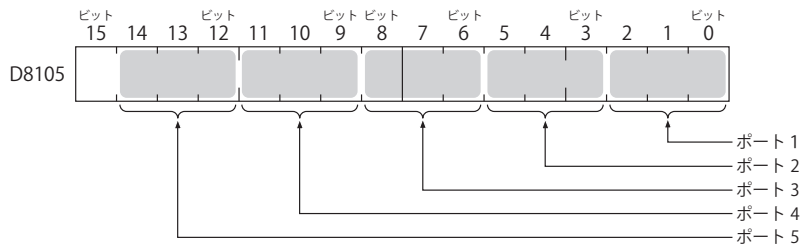


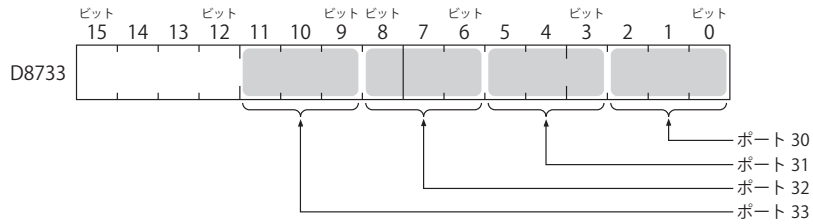


- 0 (00) : ER と DR がともに OFF です。
 1 (01) : ER が OFF、DR が ON です。
 2 (10) : ER が ON、DR が OFF です。
 3 (11) : ER と DR がともに ON です。

■ D8105、D8205、D8718、D8721、D8724、D8727、D8730、D8733：DR 制御ライン状態（ポート 1～33）

DR、ER の各制御ラインの信号状態を書き込みます。STOP 中および RUN 中の END 処理で更新します。デバイス内の各通信ポートの割り当て（ビットアサイン）は、次のとおりです。





- 0 (000) : FC6A 形の送受信制御に、DR 信号の状態を使用しません。
DR 信号制御を行う必要がないときは、この状態で使用してください。
- 1 (001) : DR 信号が ON のときに、FC6A 形が送受信可能になります。



- 2 (010) : DR 信号が OFF のときに、FC6A 形が送受信可能になります。



- 3 (011) : DR 信号が ON のときに、送信可能になります (受信は常に可能です)。



これは通常「Busy 制御」と呼び、処理速度が遅い機器 (プリンタなど) の送信制御に使います。
(接続機器から見れば、入力データの制限となります。)

- 4 (100) : DR 信号が OFF のときに、送信可能になります。

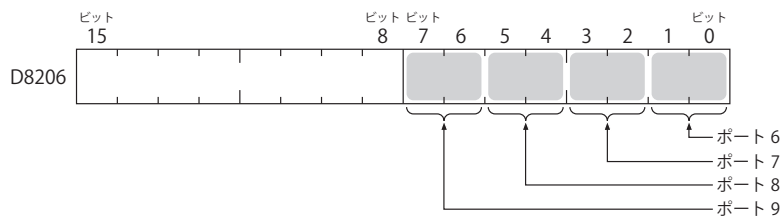
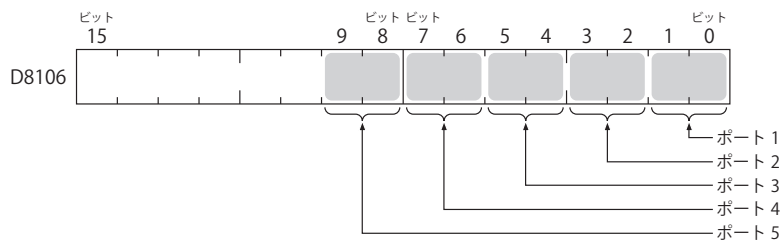


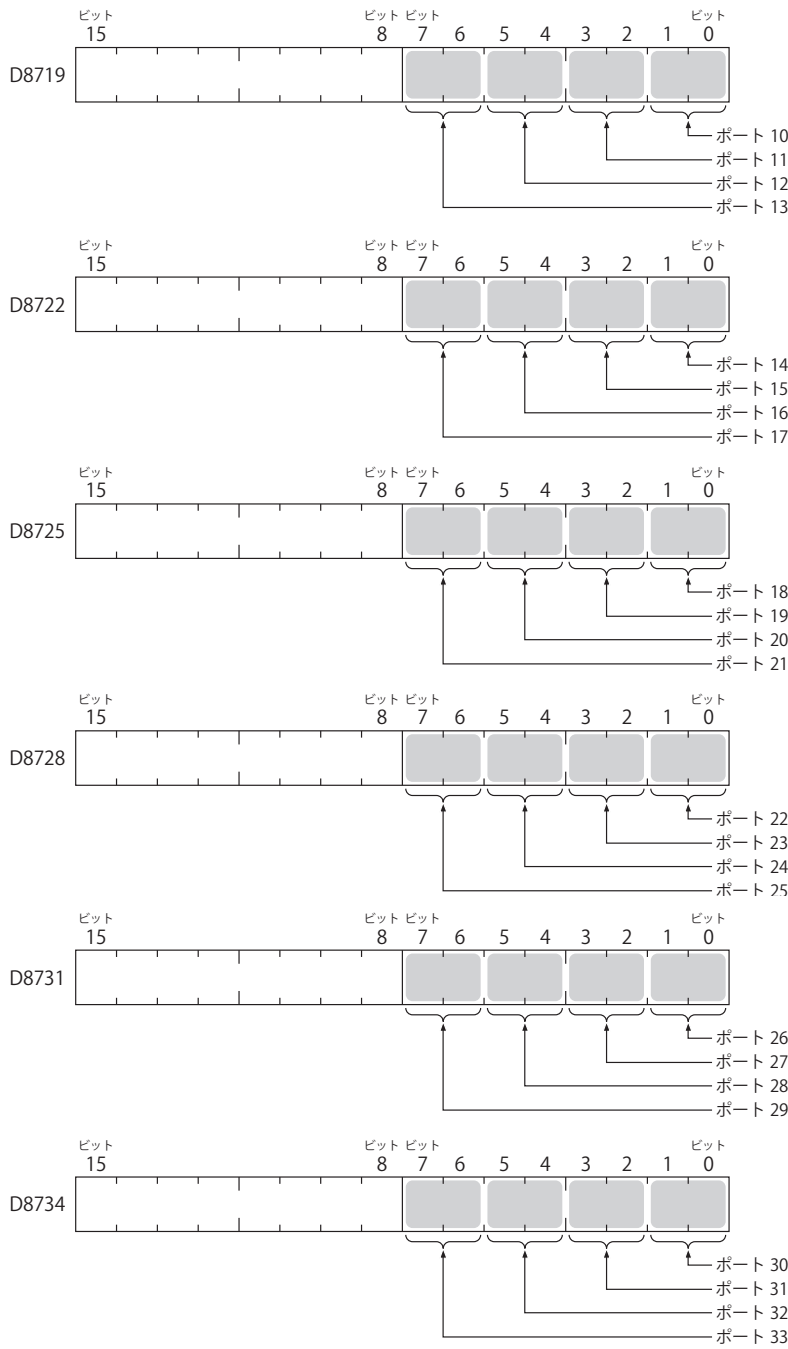
- 5 以上 : 設定値 "000" と同じ動作をします。

■ D8106、D8206、D8719、D8722、D8725、D8728、D8731、D8734 : ER 出力制御ラインコントロール (ポート 1 ~ 33)

FC6A 形のコントロール状態や、送受信状態を相手機器に示すときに使用します。この制御ラインは、FC6A 形から相手機器への出力信号です。ユーザー通信時のみ有効です。

デバイス内の各通信ポートの割り当て (ビットアサイン) は、次のとおりです。





0 (00) : FC6A 形が RUN 状態のときに ON、STOP 状態のときに OFF になります。
 RUN 中はデータの送受信に関わらず常に ON です。RUN 状態の表示が必要なときに設定します。



1 (01) : 常時 OFF になります。
 2 (10) : 受信データをフロー制御するときに設定します。相手機器からデータを受信できるときに、ON になります。
 受信できないときは OFF になります。



3 (11) : 設定値 "0" と同一の動作をします。

■ D8120、D8121：HMI モジュール情報

HMI モジュールの機種情報を書き込みます。

情報の割り当て（ビットアサイン）は、次のとおりです。



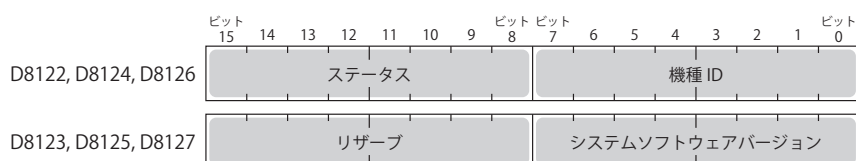
システムソフトウェアバージョンには機器内に書き込まれたソフトウェアバージョンが表示されます。

機種 ID とステータスの詳細は、「機種 ID、ステータス一覧」（2-50 頁）を参照してください。

■ D8122～D8127：カートリッジスロット情報

カートリッジスロット 1～3 の情報を書き込みます。

情報の割り当て（ビットアサイン）は、次のとおりです。



D8122、D8123＝カートリッジスロット1情報

D8124、D8125＝カートリッジスロット2情報

D8126、D8127＝カートリッジスロット3情報

システムソフトウェアバージョンには機器内に書き込まれたソフトウェアバージョンが表示されます。

機種 ID とステータスの詳細は、「機種 ID、ステータス一覧」（2-50 頁）を参照してください。

■ D8170、D8171、D8174、D8175、D8178、D8179：アナログ I/O カートリッジ入出力

アナログ I/O カートリッジのアナログ入出力値を書き込みます。

アナログ入力タイプの場合：アナログ I/O カートリッジに入力されたアナログ値をデジタル値に変換して書き込みます。

アナログ出力タイプの場合：デジタル値として格納したデジタル値をアナログ値に変換してアナログ I/O カートリッジより出力します。

詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 10 章 アナログ I/O カートリッジ」を参照してください。

D8170 = AI2/AQ2

D8171 = AI3/AQ3

D8174 = AI4/AQ4

D8175 = AI5/AQ5

D8178 = AI6/AQ6

D8179 = AI7/AQ7

■ D8172、D8173、D8176、D8177、D8180、D8181、：アナログ I/O カートリッジステータス

アナログ I/O カートリッジのアナログステータスを書き込みます。

詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 10 章 アナログ I/O カートリッジ」を参照してください。

D8172 = AI2/AQ2

D8173 = AI3/AQ3

D8176 = AI4/AQ4

D8177 = AI5/AQ5

D8180 = AI6/AQ6

D8181 = AI7/AQ7

■ D8192～D8203、D8210～D8213、D8216～D8229、D8232～D8237：高速カウンタ

高速カウンタ機能および周波数測定機能で使用する特殊データレジスタです。

高速カウンタの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 高速カウンタ」を参照してください。

D8210～D8213、D8216、D8217＝高速カウンタ（グループ 1/10）

D8218～D8221、D8234、D8235＝高速カウンタ（グループ 3/13）

D8222～D8225、D8236、D8237＝高速カウンタ（グループ 4/14）

D8226～D8229、D8232、D8233＝高速カウンタ（グループ 5/16）

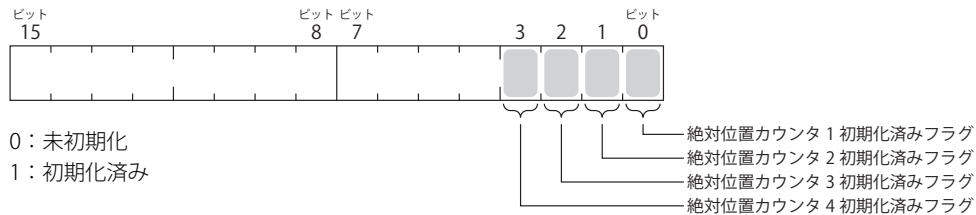
D8192～D8197＝高速カウンタ（グループ 2/11）

D8198～D8203＝高速カウンタ（グループ 6/17）

■ D8239：絶対位置管理ステータス

絶対位置管理の状態を示します。

デバイス内の絶対位置カウンタ初期化済みフラグの割り当て（ビットアサイン）は、次のとおりです。絶対位置管理ステータスの詳細は、ラダープログラミング マニュアル「第 18 章 ABS（絶対位置セット）」を参照してください。



■ D8240～D8247：絶対位置カウンタ 1～4

パルス出力に応じて、絶対位置を書き込みます。絶対位置は ABS 命令で初期化できます。

パルス出力命令（RAMP/ARAMP/ZRN/JOG）で方向制御ありを設定した場合、出力結果に応じて絶対位置が加減算されます。絶対位置カウンタの詳細は、ラダープログラミング マニュアル「第 18 章 ABS（絶対位置セット）」を参照してください。

D8240、D8241＝絶対位置カウンタ 1

D8242、D8243＝絶対位置カウンタ 2

D8244、D8245＝絶対位置カウンタ 3

D8246、D8247＝絶対位置カウンタ 4

■ D8250：SD メモリカード容量表示

認識した SD、SDHC（最大 32 ギガバイト）対応の SD メモリカードの容量をメガバイト単位で表示します。

SD メモリカードが挿入されていない場合、または認識していない場合は 0 になります。

■ D8251：SD メモリカード空き容量表示

認識した SD メモリカードの空き容量をメガバイト単位で表示します。

SD メモリカードが挿入されていない場合、または認識していない場合は 0 になります。

■ D8254：SD メモリカードダウンロード/アップロード実行情報

実行した SD メモリカードダウンロード/アップロードの情報です。

詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

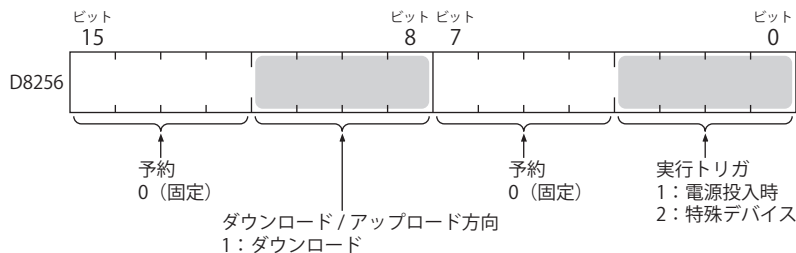
■ D8255：SD メモリカードダウンロード/アップロード実行ステータス

実行した SD メモリカードダウンロード/アップロードのステータスです。

詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

■ D8256：SD メモリカードからの MQTT 基本設定ダウンロード実行情報

実行した MQTT 基本設定用ファイルのダウンロードの情報です。デバイス内の割り当て（ビットアサイン）は、次のとおりです。



■ D8257：SD メモリカードからの MQTT 基本設定ダウンロード実行エラー情報

実行した MQTT 基本設定用ファイルのダウンロードのエラー情報です。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「MQTT 基本設定のダウンロード」を参照してください。

■ D8260：レシピブロック番号

特殊内部リレーを使用して読み書きするレシピブロック番号です。

指定したブロック番号のすべてのチャンネルが対象となります。

■ D8261：レシピ実行ブロック番号

実行したレシピのレシピブロック番号です。

レシピブロック開始時に更新し、完了時は状態を維持します。

■ D8262：レシピ実行チャンネル番号

実行したレシピのチャンネル番号です。

チャンネル処理開始時に更新し、完了時は状態を維持します。

■ **D8263：レシピ実行動作**

実行したレシピの動作に関する情報です。

レシピの詳細は、FC6A形マイクロスマート ユーザーズ マニュアル「第 11 章 レシピ機能」を参照ください。

■ **D8264：レシピ実行ステータス**

実行したレシピのステータスです。

レシピの詳細は、FC6A形マイクロスマート ユーザーズ マニュアル「第 11 章 レシピ機能」を参照ください。

■ **D8265：レシピ実行エラー情報**

実行したレシピのエラー情報です。

レシピの詳細は、FC6A形マイクロスマート ユーザーズ マニュアル「第 11 章 レシピ機能」を参照ください。

■ **D8266：レシピ内蔵メモリ（ROM-領域 1）読み出し回数**

デバイス値をレシピデータとして内蔵メモリ（ROM-領域 1）へ読み出した回数が 100 単位で格納されます。

■ **D8267：レシピ内蔵メモリ（ROM-領域 2）読み出し回数**

デバイス値をレシピデータとして内蔵メモリ（ROM-領域 2）へ読み出した回数が 100 単位で格納されます。

■ **D8268～D8275、D8774～D8781：リモートホスト番号 1～255（コネクション 1～16）**

ユーザー通信クライアントを設定しているときに通信相手を変更する特殊データレジスタです。詳細は、「第 5 章 リモートホスト番号の切り替え」（5-42 頁）を参照してください。

■ **D8278、D8279、D8760、D8761：通信モード情報（コネクション 1～16）**

D8278 = コネクション 1～4 の通信モードを示します。

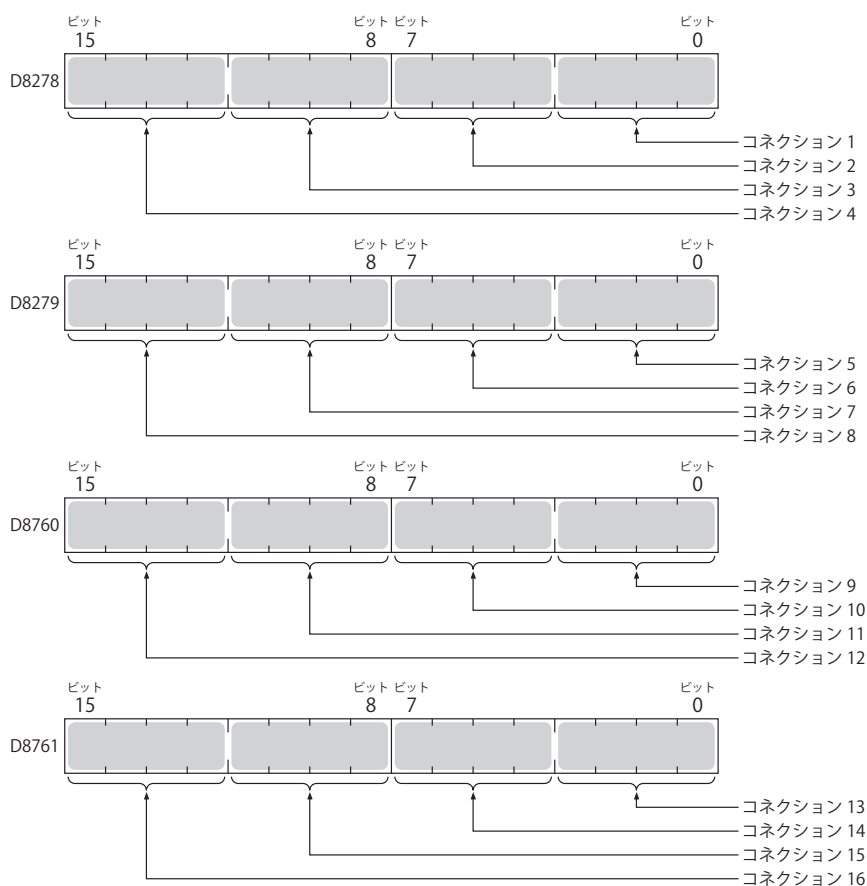
D8279 = コネクション 5～8 の通信モードを示します。

D8760 = コネクション 9～12 の通信モードを示します。

D8761 = コネクション 13～16 の通信モードを示します。

デバイス内の割り当て（ビットアサイン）は、次のとおりです。

コネクションは、コネクション番号の小さいものから順に、下位の桁から 4 ビット単位で割り付きます。



コネクション (4bit) の詳細は次のとおりです。

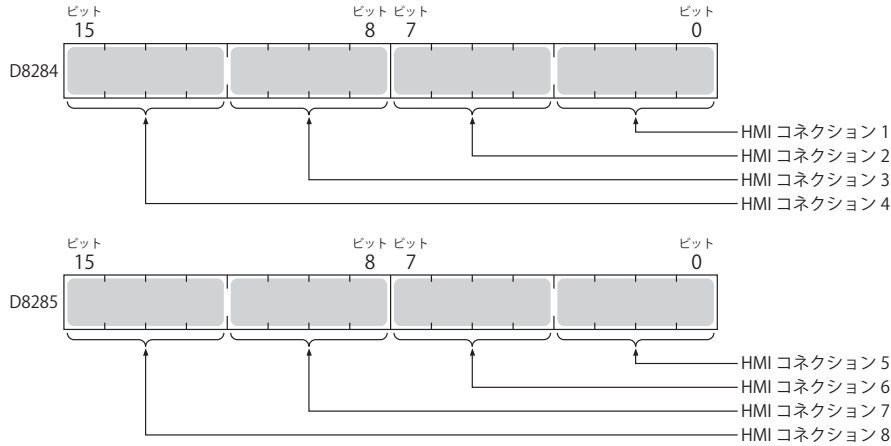
- ・ クライアントコネクション (最上位 bit = 0)
 - 0000 : 未使用
 - 0001 : ユーザー通信クライアント
 - 0010 : Modbus TCP クライアント
 - 0011 : MC プロトコル クライアント
 - 0100 : ユーザー通信 UDP
- ・ サーバー コネクション (最上位 bit = 1)
 - 1000 : メンテナンス通信サーバー
 - 1001 : ユーザー通信サーバー
 - 1010 : Modbus TCP サーバー

■ D8284、D8285 : 通信モード情報 (HMI コネクション) (HMI コネクション 1 ~ 8)

D8284 : HMI コネクション 1 ~ 4 の通信モードを示します。

D8285 : HMI コネクション 5 ~ 8 の通信モードを示します。

デバイス内の各コネクションの割り当て (ビットアサイン) は、次のとおりです。



- ・ クライアントコネクション (最上位 bit = 0)
 - 0000 : 未使用
- ・ サーバー コネクション (最上位 bit = 1)
 - 1000 : メンテナンス通信

■ D8303 : CPU モジュール Ethernet ポート 1 IP 設定 / DNS 設定切り替え

D8303 に下表にある設定値を書き込み、M8190 を ON することで Ethernet ポート 1 の IP 設定 / DNS 設定を変更できます。本機能を使用するには、WindLDR の [ファンクション設定] 以下の項目を有効にしてください。

- ・ All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュール
[ファンクション設定] の [ネットワーク設定] にある [D8303 (IP 設定 / DNS 設定切り替え) を有効にする]
- ・ Plus CPU モジュール
[ファンクション設定] の [イーサネットポート 1] にある [D8303 (IP 設定 / DNS 設定切り替え) を有効にする]

設定値の意味は、次のとおりです。

設定値	IP 設定 / DNS 設定
0	ファンクション設定に従います。
1	DHCPを有効にします。
2	特殊データレジスタ (D8304~D8323) の設定に従います。

■ D8304 ~ D8307 : CPU モジュール Ethernet ポート 1 IP アドレス (書き込み専用)

CPU モジュールの IP アドレスの書き込みに使用します。

IP アドレス : aaa.bbb.ccc.ddd に設定する場合、次のように書き込みます。

D8304 = aaa、D8305 = bbb、D8306 = ccc、D8307 = ddd

■ D8308 ~ D8311 : CPU モジュール Ethernet ポート 1 サブネットマスク (書き込み専用)

CPU モジュールのサブネットマスクの書き込みに使用します。

サブネットマスク : aaa.bbb.ccc.ddd の場合、次のように書き込みます。

D8308 = aaa、D8309 = bbb、D8310 = ccc、D8311 = ddd

■ D8312 ~ D8315 : CPU モジュール Ethernet ポート 1 デフォルトゲートウェイ (書き込み専用)

CPU モジュールのデフォルトゲートウェイの書き込みに使用します。

デフォルトゲートウェイ : aaa.bbb.ccc.ddd の場合、次のように書き込みます。

D8312 = aaa、D8313 = bbb、D8314 = ccc、D8315 = ddd

- **D8316 ～ D8319：CPU モジュール Ethernet ポート 1 優先 DNS サーバー（書き込み専用）**
CPU モジュールの優先 DNS サーバーの書き込みに使用します。
優先 DNS サーバー：aaa.bbb.ccc.ddd の場合、次のように書き込みます。
D8316 = aaa、D8317 = bbb、D8318 = ccc、D8319 = ddd
- **D8320 ～ D8323：CPU モジュール Ethernet ポート 1 代替 DNS サーバー（書き込み専用）**
CPU モジュールの代替 DNS サーバーの書き込みに使用します。
代替 DNS サーバー：aaa.bbb.ccc.ddd の場合、次のように書き込みます。
D8320 = aaa、D8321 = bbb、D8322 = ccc、D8323 = ddd
- **D8324 ～ D8329：CPU モジュール Ethernet ポート 1 MAC アドレス（現在値読み出し専用）**
CPU モジュールの MAC アドレスが各特殊データレジスタに次のように書き込まれます。
例) MAC アドレス：AA-BB-CC-DD-EE-FF
D8324=AA、D8325=BB、D8326=CC、D8327=DD、D8328=EE、D8329=FF
- **D8330 ～ D8333：CPU モジュール Ethernet ポート 1 IP アドレス（現在値読み出し専用）**
CPU モジュールの IP アドレスが各特殊データレジスタに次のように書き込まれます。
例) 自機 IP アドレス：aaa.bbb.ccc.ddd
D8330=aaa、D8331=bbb、D8332=ccc、D8333=ddd
- **D8334 ～ D8337：CPU モジュール Ethernet ポート 1 サブネットマスク（現在値読み出し専用）**
CPU モジュールのサブネットマスクの値が各特殊データレジスタに次のように書き込まれます。
例) サブネットマスク：aaa.bbb.ccc.ddd
D8334=aaa、D8335=bbb、D8336=ccc、D8337=ddd
- **D8338 ～ D8341：CPU モジュール Ethernet ポート 1 デフォルトゲートウェイ（現在値読み出し専用）**
CPU モジュールのデフォルトゲートウェイのアドレスが各特殊データレジスタに次のように書き込まれます。
例) デフォルトゲートウェイ：aaa.bbb.ccc.ddd
D8338=aaa、D8339=bbb、D8340=ccc、D8341=ddd
- **D8342 ～ D8345：CPU モジュール Ethernet ポート 1 優先 DNS サーバー（現在値読み出し専用）**
CPU モジュールの優先 DNS サーバーのアドレスが各特殊データレジスタに次のように書き込まれます。
例) 優先 DNS サーバー：aaa.bbb.ccc.ddd
D8342=aaa、D8343=bbb、D8344=ccc、D8345=ddd
- **D8342 ～ D8345：CPU モジュール Ethernet ポート 1 代替 DNS サーバー（現在値読み出し専用）**
CPU モジュールの代替 DNS サーバーのアドレスが各特殊データレジスタに次のように書き込まれます。
例) 代替 DNS サーバー：aaa.bbb.ccc.ddd
D8346=aaa、D8347=bbb、D8348=ccc、D8349=ddd
- **D8350 ～ D8381、D8677 ～ D8708：コネクション接続 IP アドレス**
コネクションにアクセス中の相手機器の IP アドレスを次のように書き込みます。
コネクション 1 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8350=aaa、D8351=bbb、D8352=ccc、D8353=ddd
コネクション 2 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8354=aaa、D8355=bbb、D8356=ccc、D8357=ddd
コネクション 3 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8358=aaa、D8359=bbb、D8360=ccc、D8361=ddd
コネクション 4 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8362=aaa、D8363=bbb、D8364=ccc、D8365=ddd
コネクション 5 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8366=aaa、D8367=bbb、D8368=ccc、D8369=ddd
コネクション 6 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8370=aaa、D8371=bbb、D8372=ccc、D8373=ddd
コネクション 7 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8374=aaa、D8375=bbb、D8376=ccc、D8377=ddd
コネクション 8 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8378=aaa、D8379=bbb、D8380=ccc、D8381=ddd
コネクション 9 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8677=aaa、D8678=bbb、D8679=ccc、D8680=ddd
コネクション 10 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8681=aaa、D8682=bbb、D8683=ccc、D8684=ddd

コネクション 11 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8685=aaa、D8686=bbb、D8687=ccc、D8688=ddd

コネクション 12 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8689=aaa、D8690=bbb、D8691=ccc、D8692=ddd

コネクション 13 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8693=aaa、D8694=bbb、D8695=ccc、D8696=ddd

コネクション 14 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8697=aaa、D8698=bbb、D8699=ccc、D8700=ddd

コネクション 15 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8701=aaa、D8702=bbb、D8703=ccc、D8704=ddd

コネクション 16 接続 IP アドレス：aaa.bbb.ccc.ddd の場合
D8705=aaa、D8706=bbb、D8707=ccc、D8708=ddd

■ **D8382 ～ D8387：HMI モジュール MAC アドレス（現在値読み出し専用）**

MAC アドレスが各特殊データレジスタに次のように書き込まれます。

例) MAC アドレス：AA-BB-CC-DD-EE-FF

D8382=AA、D8383=BB、D8384=CC、D8385=DD、D8386=EE、D8387=FF

■ **D8388 ～ D8391：HMI モジュール IP アドレス（現在値読み出し専用）**

HMI モジュールの IP アドレスが各特殊データレジスタに次のように書き込まれます。

例) HMI モジュール IP アドレス：aaa.bbb.ccc.ddd

D8388=aaa、D8389=bbb、D8390=ccc、D8391=ddd

■ **D8392 ～ D8395：HMI モジュールサブネットマスク（現在値読み出し専用）**

HMI モジュールのサブネットマスクの値が各特殊データレジスタに次のように書き込まれます。

例) HMI モジュールサブネットマスク：aaa.bbb.ccc.ddd

D8392=aaa、D8393=bbb、D8394=ccc、D8395=ddd

■ **D8396 ～ D8399：HMI モジュールデフォルトゲートウェイ（現在値読み出し専用）**

HMI モジュールのデフォルトゲートウェイのアドレスが各特殊データレジスタに次のように書き込まれます。

例) HMI モジュールデフォルトゲートウェイ：aaa.bbb.ccc.ddd

D8396=aaa、D8397=bbb、D8398=ccc、D8399=ddd

■ **D8400 ～ D8403：HMI モジュール優先 DNS サーバー（現在値読み出し専用）**

HMI モジュールの優先 DNS サーバーのアドレスが各特殊データレジスタに次のように書き込まれます。

例) HMI モジュール優先 DNS サーバー：aaa.bbb.ccc.ddd

D8400=aaa、D8401=bbb、D8402=ccc、D8403=ddd

■ **D8404 ～ D8407：HMI モジュール代替 DNS サーバー（現在値読み出し専用）**

HMI モジュールの代替 DNS サーバーのアドレスが各特殊データレジスタに次のように書き込まれます。

例) HMI モジュール代替 DNS サーバー：aaa.bbb.ccc.ddd

D8404=aaa、D8405=bbb、D8406=ccc、D8407=ddd

■ **D8413：タイムゾーンオフセット**

ファンクション設定で設定したタイムゾーンを 15 分単位で微調整できます。

詳細は、「第 3 章 SNTP 設定」(3-11 頁) を参照してください。

■ **D8414：SNTP 動作ステータス**

M8191 (SNTP 時刻取得フラグ) の操作による時刻情報取得、または自動取得で時刻情報取得を実行した場合の動作ステータスが書き込まれます。

詳細は、「第 3 章 動作ステータスの確認 (D8414)」(3-12 頁) を参照してください。

■ **D8415：SNTP アクセス経過時間**

最後に SNTP サーバーから時刻情報を取得したときからの経過時間が分単位で書き込まれます。

詳細は、「第 3 章 現在時刻を取得してからの経過時間 (D8415)」(3-12 頁) を参照してください。

■ **D8429：HMI モジュール コネクション情報参照 コネクション番号**

指定したコネクション番号のコネクション情報を、D8430 ～ D8434 に反映します。0 を書き込むと、D8430 ～ D8434 に 0 を書き込みます。存在しないコネクション番号を指定した場合も D8430 ～ D8434 に 0 を書き込みます。

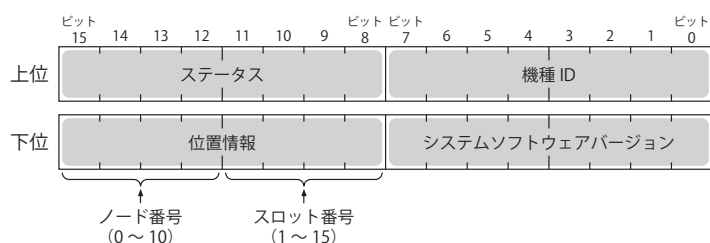
■ **D8430 ～ D8433：HMI モジュール コネクション情報参照 接続 IP アドレス**

コネクションにアクセス中の端末の IP アドレスが各特殊データレジスタに次のように書き込まれます。

例) 読み出し対象の IP アドレス：aaa.bbb.ccc.ddd

D8430=aaa、D8431=bbb、D8432=ccc、D8433=ddd

- **D8434：HMI モジュール コネクション情報参照 接続ポート番号**
コネクションにアクセス中の端末のポート番号が書き込まれます。
- **D8437～D8440：HMI モジュール IP アドレス（書き込み専用）**
HMI モジュールの IP アドレスの書き込みに使用します。
HMI モジュール IP アドレス：aaa.bbb.ccc.ddd に設定する場合、次のように書き込みます。
D8437 = aaa、D8438 = bbb、D8439 = ccc、D8440 = ddd
- **D8441～D8444：HMI モジュールサブネットマスク（書き込み専用）**
HMI モジュールのサブネットマスクの書き込みに使用します。
HMI モジュールサブネットマスク：aaa.bbb.ccc.ddd の場合、次のように書き込みます。
D8441 = aaa、D8442 = bbb、D8443 = ccc、D8444 = ddd
- **D8445～D8448：HMI モジュールデフォルトゲートウェイ（書き込み専用）**
HMI モジュールのデフォルトゲートウェイの書き込みに使用します。
HMI モジュールデフォルトゲートウェイ：aaa.bbb.ccc.ddd の場合、次のように書き込みます。
D8445 = aaa、D8446 = bbb、D8447 = ccc、D8448 = ddd
- **D8449～D8452：HMI モジュール優先 DNS サーバー（書き込み専用）**
HMI モジュールの優先 DNS サーバーの書き込みに使用します。
HMI モジュール優先 DNS サーバー：aaa.bbb.ccc.ddd の場合、下記のように書き込みます。
D8449 = aaa、D8450 = bbb、D8451 = ccc、D8452 = ddd
- **D8453～D8456：HMI モジュール代替 DNS サーバー（書き込み専用）**
HMI モジュールの代替 DNS サーバーの書き込みに使用します。
HMI モジュール代替 DNS サーバー：aaa.bbb.ccc.ddd の場合、下記のように書き込みます。
D8453 = aaa、D8454 = bbb、D8455 = ccc、D8456 = ddd
- **D8457：EMAIL 命令詳細エラー情報（HMI-Ethernet ポート）**
EMAIL 命令（HMI-Ethernet ポート）の詳細なエラー情報が書き込まれます。
詳細は、「第 12 章 特殊データレジスタで、Eメールの詳細エラーを確認する」（12-3 頁）を参照してください。
- **D8470～D8595：増設モジュール スロット情報**
増設モジュールの機種情報を書き込みます。
情報の割り当て（ビットアサイン）は、次のとおりです。



上位 下位

- D8470、D8471 = 増設モジュールスロット 1 情報
- D8472、D8473 = 増設モジュールスロット 2 情報
- D8474、D8475 = 増設モジュールスロット 3 情報
- D8476、D8477 = 増設モジュールスロット 4 情報
- D8478、D8479 = 増設モジュールスロット 5 情報
- D8480、D8481 = 増設モジュールスロット 6 情報
- D8482、D8483 = 増設モジュールスロット 7 情報
- D8484、D8485 = 増設モジュールスロット 8 情報
- D8486、D8487 = 増設モジュールスロット 9 情報
- D8488、D8489 = 増設モジュールスロット 10 情報
- D8490、D8491 = 増設モジュールスロット 11 情報
- D8492、D8493 = 増設モジュールスロット 12 情報
- D8494、D8495 = 増設モジュールスロット 13 情報
- D8496、D8497 = 増設モジュールスロット 14 情報
- D8498、D8499 = 増設モジュールスロット 15 情報
- D8500、D8501 = 増設モジュールスロット 16 情報

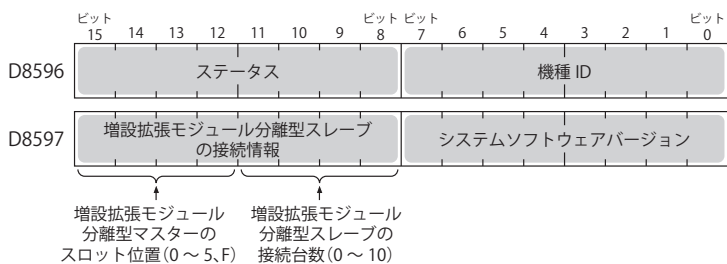
上位 下位

D8502、D8503 = 増設モジュール スロット 17 情報
D8504、D8505 = 増設モジュール スロット 18 情報
D8506、D8507 = 増設モジュール スロット 19 情報
D8508、D8509 = 増設モジュール スロット 20 情報
D8510、D8511 = 増設モジュール スロット 21 情報
D8512、D8513 = 増設モジュール スロット 22 情報
D8514、D8515 = 増設モジュール スロット 23 情報
D8516、D8517 = 増設モジュール スロット 24 情報
D8518、D8519 = 増設モジュール スロット 25 情報
D8520、D8521 = 増設モジュール スロット 26 情報
D8522、D8523 = 増設モジュール スロット 27 情報
D8524、D8525 = 増設モジュール スロット 28 情報
D8526、D8527 = 増設モジュール スロット 29 情報
D8528、D8529 = 増設モジュール スロット 30 情報
D8530、D8531 = 増設モジュール スロット 31 情報
D8532、D8533 = 増設モジュール スロット 32 情報
D8534、D8535 = 増設モジュール スロット 33 情報
D8536、D8537 = 増設モジュール スロット 34 情報
D8538、D8539 = 増設モジュール スロット 35 情報
D8540、D8541 = 増設モジュール スロット 36 情報
D8542、D8543 = 増設モジュール スロット 37 情報
D8544、D8545 = 増設モジュール スロット 38 情報
D8546、D8547 = 増設モジュール スロット 39 情報
D8548、D8549 = 増設モジュール スロット 40 情報
D8550、D8551 = 増設モジュール スロット 41 情報
D8552、D8553 = 増設モジュール スロット 42 情報
D8554、D8555 = 増設モジュール スロット 43 情報
D8556、D8557 = 増設モジュール スロット 44 情報
D8558、D8559 = 増設モジュール スロット 45 情報
D8560、D8561 = 増設モジュール スロット 46 情報
D8562、D8563 = 増設モジュール スロット 47 情報
D8564、D8565 = 増設モジュール スロット 48 情報
D8566、D8567 = 増設モジュール スロット 49 情報
D8568、D8569 = 増設モジュール スロット 50 情報
D8570、D8571 = 増設モジュール スロット 51 情報
D8572、D8573 = 増設モジュール スロット 52 情報
D8574、D8575 = 増設モジュール スロット 53 情報
D8576、D8577 = 増設モジュール スロット 54 情報
D8578、D8579 = 増設モジュール スロット 55 情報
D8580、D8581 = 増設モジュール スロット 56 情報
D8582、D8583 = 増設モジュール スロット 57 情報
D8584、D8585 = 増設モジュール スロット 58 情報
D8586、D8587 = 増設モジュール スロット 59 情報
D8588、D8589 = 増設モジュール スロット 60 情報
D8590、D8591 = 増設モジュール スロット 61 情報
D8592、D8593 = 増設モジュール スロット 62 情報
D8594、D8595 = 増設モジュール スロット 63 情報

システムソフトウェアバージョンには機器内には書き込まれたソフトウェアバージョンが表示されます。
機種 ID とステータスの詳細は、「機種 ID、ステータス一覧」(2-50 頁) を参照してください。

■ D8596、D8597：増設拡張モジュール分離型マスタースロット情報

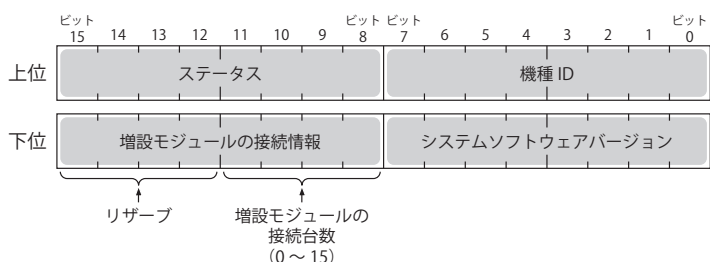
増設拡張モジュール分離型マスターの機種情報を書き込みます。
情報の割り当て（ビットアサイン）は、次のとおりです。



システムソフトウェアバージョンには機器内に書き込まれたソフトウェアバージョンが表示されます。
機種IDとステータスの詳細は、「機種ID、ステータス一覧」（2-50 頁）を参照してください。

■ D8598～D8617：増設拡張モジュール分離型スレーブスロット情報

増設拡張モジュール分離型スレーブの機種情報を書き込みます。
情報の割り当て（ビットアサイン）は、次のとおりです。



上位 下位

- D8598、D8599 = 増設拡張モジュール分離型スレーブ（1 台目）スロット情報
- D8600、D8601 = 増設拡張モジュール分離型スレーブ（2 台目）スロット情報
- D8602、D8603 = 増設拡張モジュール分離型スレーブ（3 台目）スロット情報
- D8604、D8605 = 増設拡張モジュール分離型スレーブ（4 台目）スロット情報
- D8606、D8607 = 増設拡張モジュール分離型スレーブ（5 台目）スロット情報
- D8608、D8609 = 増設拡張モジュール分離型スレーブ（6 台目）スロット情報
- D8610、D8611 = 増設拡張モジュール分離型スレーブ（7 台目）スロット情報
- D8612、D8613 = 増設拡張モジュール分離型スレーブ（8 台目）スロット情報
- D8614、D8615 = 増設拡張モジュール分離型スレーブ（9 台目）スロット情報
- D8616、D8617 = 増設拡張モジュール分離型スレーブ（10 台目）スロット情報

システムソフトウェアバージョンには機器内に書き込まれたソフトウェアバージョンが表示されます。
機種IDとステータスの詳細は、「機種ID、ステータス一覧」（2-50 頁）を参照してください。

■ D8618：増設拡張モジュール分離型 I/O リフレッシュ時間 現在値

増設拡張モジュール分離型以降に接続した増設モジュールの I/O リフレッシュ時間の現在値が 1ms 単位で格納されます。

■ D8619：増設拡張モジュール分離型 I/O リフレッシュ時間 最大値

増設拡張モジュール分離型以降に接続した増設モジュールの I/O フレッシュ時間の最大値が 1ms 単位で格納されます。

■ D8630：CPU モジュール Ethernet ポート 2 IP 設定 / DNS 設定切り替え

D8630 に下表にある設定値を書き込み、M8333 を ON することで、Ethernet ポート 2 の IP 設定 / DNS 設定を変更できます。
本機能を使用するには、WindLDR の [ファンクション設定] で [イーサネットポート 2] の [D8630 (IP 設定 / DNS 設定切り替え)] を有効にする] を有効にしてください。
設定値の意味は、次のとおりです。

設定値	IP 設定 / DNS 設定
0	ファンクション設定に従います。
1	DHCPを有効にします。
2	特殊データレジスタ（D8631～D8650）の設定に従います。

- **D8631 ～ D8634：CPU モジュール Ethernet ポート 2 IP アドレス（書き込み専用）**
Plus CPU モジュールの Ethernet ポート 2 の IP アドレスの書き込みに使用します。
IP アドレス：aaa.bbb.ccc.ddd に設定する場合、次のように書き込みます。
D8631 = aaa、D8632 = bbb、D8633 = ccc、D8634 = ddd
- **D8635 ～ D8638：CPU モジュール Ethernet ポート 2 サブネットマスク（書き込み専用）**
Plus CPU モジュールの Ethernet ポート 2 のサブネットマスクの書き込みに使用します。
サブネットマスク：aaa.bbb.ccc.ddd の場合、次のように書き込みます。
D8635 = aaa、D8636 = bbb、D8637 = ccc、D8638 = ddd
- **D8639 ～ D8642：CPU モジュール Ethernet ポート 2 デフォルトゲートウェイ（書き込み専用）**
Plus CPU モジュールの Ethernet ポート 2 のデフォルトゲートウェイの書き込みに使用します。
デフォルトゲートウェイ：aaa.bbb.ccc.ddd の場合、次のように書き込みます。
D8639 = aaa、D8640 = bbb、D8641 = ccc、D8642 = ddd
- **D8643 ～ D8646：CPU モジュール Ethernet ポート 2 優先 DNS サーバー（書き込み専用）**
Plus CPU モジュールの Ethernet ポート 2 の優先 DNS サーバーの書き込みに使用します。
優先 DNS サーバー：aaa.bbb.ccc.ddd の場合、次のように書き込みます。
D8643 = aaa、D8644 = bbb、D8645 = ccc、D8646 = ddd
- **D8647 ～ D8650：CPU モジュール Ethernet ポート 2 代替 DNS サーバー（書き込み専用）**
Plus CPU モジュールの Ethernet ポート 2 の代替 DNS サーバーの書き込みに使用します。
代替 DNS サーバー：aaa.bbb.ccc.ddd の場合、次のように書き込みます。
D8647 = aaa、D8648 = bbb、D8649 = ccc、D8650 = ddd
- **D8651 ～ D8656：CPU モジュール Ethernet ポート 2 MAC アドレス（現在値読み出し専用）**
Plus CPU モジュールの Ethernet ポート 2 の MAC アドレスが各特殊データレジスタに次のように書き込まれます。
例) MAC アドレス：AA-BB-CC-DD-EE-FF
D8651=AA、D8652=BB、D8653=CC、D8654=DD、D8655=EE、D8656=FF
- **D8657 ～ D8660：CPU モジュール Ethernet ポート 2 IP アドレス（現在値読み出し専用）**
Plus CPU モジュールの Ethernet ポート 2 の IP アドレスが各特殊データレジスタに次のように書き込まれます。
例) 自機 IP アドレス：aaa.bbb.ccc.ddd
D8657=aaa、D8658=bbb、D8659=ccc、D8660=ddd
- **D8661 ～ D8664：CPU モジュール Ethernet ポート 2 サブネットマスク（現在値読み出し専用）**
Plus CPU モジュールの Ethernet ポート 2 のサブネットマスクの値が各特殊データレジスタに次のように書き込まれます。
例) サブネットマスク：aaa.bbb.ccc.ddd
D8661=aaa、D8662=bbb、D8663=ccc、D8664=ddd
- **D8665 ～ D8668：CPU モジュール Ethernet ポート 2 デフォルトゲートウェイ（現在値読み出し専用）**
Plus CPU モジュールの Ethernet ポート 2 のデフォルトゲートウェイのアドレスが各特殊データレジスタに次のように書き込まれます。
例) デフォルトゲートウェイ：aaa.bbb.ccc.ddd
D8665=aaa、D8666=bbb、D8667=ccc、D8668=ddd
- **D8669 ～ D8672：CPU モジュール Ethernet ポート 2 優先 DNS サーバー（現在値読み出し専用）**
Plus CPU モジュールの Ethernet ポート 2 の優先 DNS サーバーのアドレスが各特殊データレジスタに次のように書き込まれます。
例) 優先 DNS サーバー：aaa.bbb.ccc.ddd
D8669=aaa、D8670=bbb、D8671=ccc、D8672=ddd
- **D8673 ～ D8676：CPU モジュール Ethernet ポート 2 代替 DNS サーバー（現在値読み出し専用）**
Plus CPU モジュールの Ethernet ポート 2 の代替 DNS サーバーのアドレスが各特殊データレジスタに次のように書き込まれます。
例) 代替 DNS サーバー：aaa.bbb.ccc.ddd
D8673=aaa、D8674=bbb、D8675=ccc、D8676=ddd
- **D8759：EMAIL 命令詳細エラー情報（Ethernet ポート 1）**
EMAIL 命令（Ethernet ポート 1）の詳細なエラー情報が書き込まれます。
詳細は、「第 12 章 特殊データレジスタで、E メールの詳細エラーを確認する」（12-3 頁）を参照してください。
- **D8782：BACnet 運転状態**
BACnet 通信の運転状態が書き込まれます。
詳細は、「第 15 章 BACnet/IP で使用する特殊デバイス」（15-11 頁）を参照してください。
- **D8783：BACnet エラー情報**
BACnet 通信で最後に発生したエラー情報が書き込まれます。
詳細は、「第 15 章 BACnet/IP で使用する特殊デバイス」（15-11 頁）を参照してください。

- **D8790：EtherNet/IP 運転状態**

EtherNet/IP 通信の動作状態が書き込まれます。

詳細は、「第 16 章 EtherNet/IP 通信で使用する特殊デバイス」(16-13 頁)を参照してください。

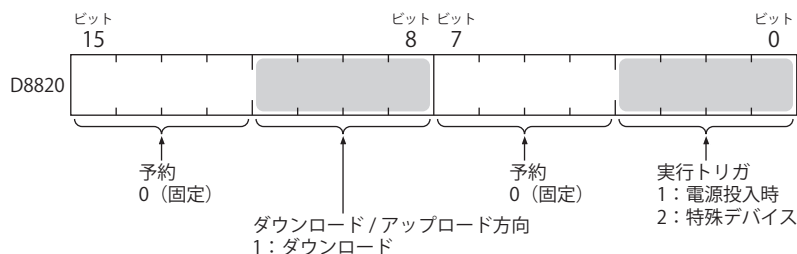
- **D8791：EtherNet/IP エラー情報**

EtherNet/IP 通信で最後に発生したエラー情報が書き込まれます。

詳細は、「第 16 章 EtherNet/IP 通信で使用する特殊デバイス」(16-13 頁)を参照してください。

- **D8820：SD メモリカードからのサーバー機能用ファイルダウンロード実行情報**

実行したサーバー機能用ファイルのダウンロードの情報です。デバイス内の割り当て（ビットアサイン）は、次のとおりです。



- **D8821：SD メモリカードからのサーバー機能用ファイルダウンロード実行エラー情報**

実行したサーバー機能用ファイルのダウンロードのエラー情報です。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

- **D8822：Web サーバーの現在の接続数**

Web サーバー機能の現在の接続数です。Plus CPU モジュールのみ対応しています。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

機種 ID、ステータス一覧

■機種 ID

増設モジュール、HMI モジュール

機種 ID		形番
16 進	2 進	
0x00	0000 0000	FC6A-N16B1, FC6A-N16B4, FC6A-N16B3
0x01	0000 0001	FC6A-R161, FC6A-R164, FC6A-T16K1, FC6A-T16K4, F6A-T16P1, FC6A-T16P4, F6A-T16K3, FCA-T16P3
0x02	0000 0010	FC6A-N32B3
0x03	0000 0011	FC6A-T32K3, FC6A-T32P3
0x04	0000 0100	FC6A-N08B1, FC6A-N08B4, FC6A-N08A11, FC6A-N08A14
0x05	0000 0101	FC6A-R081, FC6A-R084, FC6A-T08K1, FC6A-T08K4, FC6A-T08P1, FC6A-T08P4
0x06	0000 0110	FC6A-M08BR1, FC6A-M08BR4
0x07	0000 0111	FC6A-M24BR1, FC6A-M24BR4
0x18	0001 1000	FC6A-PH1
0x19	0001 1001	FC6A-EXM2, FC6A-EXM24
0x1A	0001 1010	FC6A-EXM1S, FC6A-EXM1S4
0x20	0010 0000	FC6A-J2C1, FC6A-J2C4
0x21	0010 0001	FC6A-J4A1, FC6A-J4A4
0x22	0010 0010	FC6A-J8A1, FC6A-J8A4
0x23	0010 0011	FC6A-K2A1, FC6A-K2A4
0x24	0010 0100	FC6A-K4A1, FC6A-K4A4
0x25	0010 0101	FC6A-L06A1, FC6A-L06A4
0x26	0010 0110	FC6A-L03CN1, FC6A-L03CN4
0x27	0010 0111	FC6A-J4CN1, FC6A-J4CN4
0x28	0010 1000	FC6A-J8CU1, FC6A-J8CU4
0x29	0010 1001	FC6A-F2M1, FC6A-F2M4
0x2A	0010 1010	FC6A-F2MR1, FC6A-F2MR4
0x2B	0010 1011	FC6A-J4CH1Y, FC6A-J4CH4Y
0x2C	0010 1100	FC6A-EXM1M
0x2E	0010 1110	FC6A-SIF52, FC6A-SIF524
0xFF	1111 1111	未接続

カートリッジ

機種 ID		形番
16 進	2 進	
0x00	0000 0000	FC6A-PJ2A
0x01	0000 0001	FC6A-PK2AV
0x02	0000 0010	FC6A-PK2AW
0x03	0000 0011	FC6A-PJ2CP
0x06	0000 0110	FC6A-PC1
0x07	0000 0111	FC6A-PC3
0x09	0000 1001	FC6A-PTS4, FC6A-PTK4
0x0A	0000 1010	FC6A-PN4
0x0C	0000 1100	FC6A-PC4
0xFF	1111 1111	未接続

■ステータス

ステータス		内容
16進	2進	
0x00	0000 0000	正常
0x81	1000 0001	通信エラー（増設モジュール、HMIモジュール、カートリッジとCPUモジュール間の通信で、異常が発生しています。）
0x82	1000 0010	不明な機器の検出（FC6A形以外の機器が接続されてます。）
0x83	1000 0011	機器設定エラー（機器が接続されていないか、ユーザープログラムで設定した機器と異なる機器で接続されています。）
0x84	1000 0100	機器書込みエラー（機器の動作設定に失敗しました。）
0x85	1000 0101	システムアップデートエラー（システムアップデートに失敗しました。）
0x86	1000 0110	増設拡張モジュール分離型マスター通信エラー（増設拡張モジュール分離型マスターと増設拡張モジュール分離型スレーブ間の通信で、異常が発生しています。）

第3章 通信設定

この章では、FC6A形が持つ通信機能を使用するための設定方法について説明します。
 通信ポートやネットワーク設定など、あらかじめ設定が必要な項目については、WindLDRの「[ファンクション設定] ダイアログボックスなどで設定を行ってから、ユーザープログラムをFC6A形にダウンロードします。
 CAN J1939通信に関する設定方法は、「第8章 J1939通信」(8-1頁)を参照してください。

設定一覧

設定項目	概要	参照頁	設定場所
通信ポート	通信相手機器に合った通信機能およびポートを設定します。	3-2	ファンクション設定
ネットワーク設定	FC6A形をネットワークに接続するための情報を設定します。	3-3	
ネットワーク管理	SNTP設定やPING命令のタイムアウト時間を設定します。	3-11	
コネクション設定	FC6A形のイーサネット通信で使用する通信モードを設定します。	3-14	
リモートホストリスト	ネットワーク上の通信相手（リモートホスト）を登録・管理します。	3-21	リモートホストリスト
自動Ping機能	リモートホストのネットワーク接続状態を監視する自動Ping機能を設定します。	3-23	

通信ポートの設定

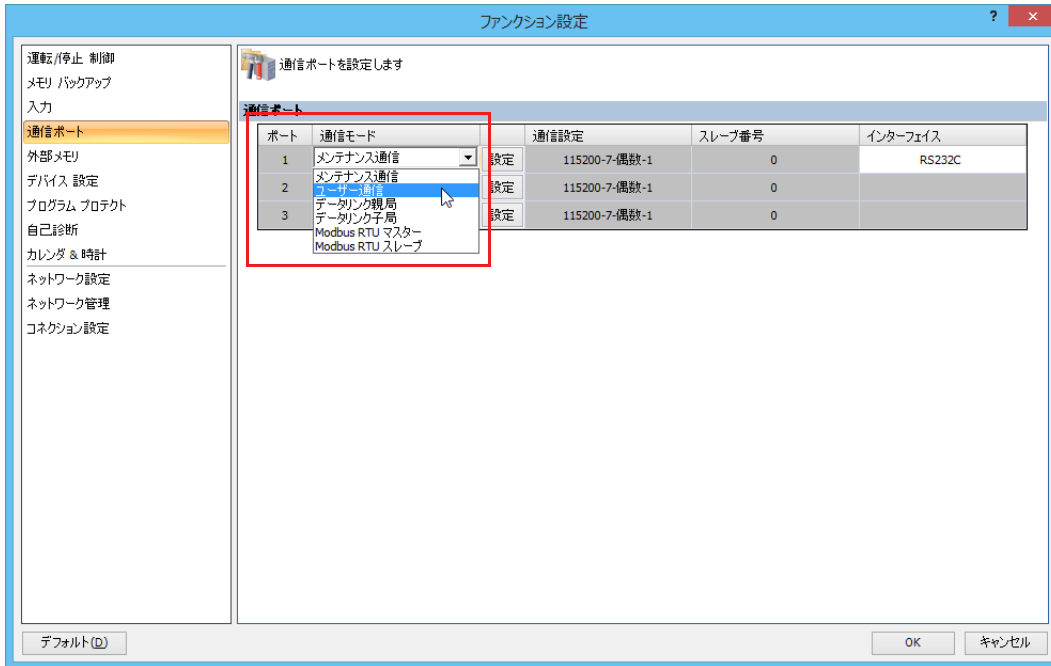
通信ポート 1～33 を使用して通信相手機器と通信する場合の通信ポートの設定について説明します。

WindLDR の設定

周辺機器の通信仕様に合わせて通信フォーマットを設定します。

●操作手順

1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するポートの [通信モード] をクリックし、使用する通信モードを選択します。
各通信モードに対応する設定ダイアログボックスが表示されます。



3. 設定ダイアログボックスで通信相手機器の通信フォーマットに合わせて、設定を変更します。
以下は、ユーザー通信の場合の例です。



4. [OK] ボタンをクリックします。

以上で、設定が完了します。

ネットワークの設定

FC6A 形の Ethernet ポート 1、Ethernet ポート 2、HMI-Ethernet ポートを使用して、ネットワークに接続するための設定について説明します。

概要

FC6A 形をイーサネットのネットワークに接続するために、FC6A 形のネットワーク設定で設定できる Ethernet ポートと設定項目は次のとおりです。

FC6A 形	Ethernet ポート	設定項目								
		IP 設定	DNS 設定	SNTp 設定	Ping 設定	E メール 設定	Web サーバー 設定	BACnet 設定	FTP 設定	EtherNet/IP 設定
All-in-One CPUモジュール /CAN J1939 All-in-One CPUモジュール	Ethernetポート1	○	○	○	○	—	—	—	—	—
Plus CPUモジュール	Ethernetポート1	○	○	○	○	○	○	○	○	—
	Ethernetポート2	○	○	—	○	—	—	—	—	○
HMIモジュール	HMI-Ethernetポート	○	○	—	—	○	○	—	—	—

ネットワーク設定

IP 設定

FC6A 形の IP アドレス、サブネットマスク、デフォルトゲートウェイは、次の 3 種類から設定できます。

設定方法	内容
IP アドレスを自動的に取得する (DHCP)	FC6A形は、ユーザープログラムダウンロード完了時およびFC6A形の電源起動時に、DHCPサーバーからIPアドレス、サブネットマスク、デフォルトゲートウェイを自動的に取得します。ただし、FC6A形を設置するネットワーク上にDHCPサーバーが存在する必要があります。
特殊データレジスタを使って IP アドレスを設定する	特殊データレジスタを使ってEthernetポート1、Ethernetポート2、HMI-EthernetポートのIPアドレス、サブネットマスク、デフォルトゲートウェイを設定します。それらの値をFC6A形の電源起動時およびM8190、M8333、M8184のOFFからONへの立ち上がり時に、FC6A形に設定します。
次の IP アドレスを使う	固定のIPアドレス、サブネットマスク、デフォルトゲートウェイを設定します。ここで設定した値をユーザープログラムダウンロード完了時に、FC6A形に設定します。



- FC6A 形のデフォルトの IP アドレス、サブネットマスク、デフォルトゲートウェイは、「192.168.1.5」、「255.255.255.0」、「0.0.0.0」です。
- FC6A 形の IP アドレス、サブネットマスク、デフォルトゲートウェイは、HMI モジュールを使用して変更できます。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 7 章 HMI 機能」を参照してください。

DNS 設定

DNS サーバーのアドレスのは、次の 3 種類から設定できます。

設定方法	内容
DNS サーバーのアドレスを自動的に取得する (DHCP)	FC6A形は、ユーザープログラムダウンロード完了時およびFC6A形の電源起動時に、DNSサーバーのアドレスを自動的に取得します。ただし、FC6A形を設置するネットワーク上にDHCPサーバーが存在する必要があります。
特殊データレジスタを使って DNS サーバーのアドレスを設定する	特殊データレジスタを使ってEthernetポート1、Ethernetポート2、HMI-Ethernetポートの優先DNSサーバーのアドレスおよび代替DNSサーバーのアドレスを設定します。それらの値をFC6A形の電源起動時およびM8190、M8333、M8184のOFFからONへの立ち上がり時に、FC6A形に設定します。
次の DNS サーバーのアドレスを使う	固定の優先DNSサーバーのアドレスおよび代替DNSサーバーのアドレスを設定します。ここで設定した値をユーザープログラムダウンロード完了時に、FC6A形に設定します。



- 優先 DNS サーバーから IP アドレスを取得できない場合、代替 DNS サーバーにアクセスします。
- DNS サーバーのアドレスは、HMI モジュールを使用して変更できます。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 7 章 HMI 機能」を参照してください。

特殊データレジスタによるネットワーク設定

IP アドレス、DNS サーバーのアドレスを特殊データレジスタを使って設定する場合、FC6A 形の電源起動時および M8190、M8333 の OFF から ON への立ち上がり時に、特殊データレジスタに格納された値を設定します。

例)

- M8190 (CPU モジュール Ethernet ポート 1 ネットワーク設定変更トリガ) を OFF から ON にすると、D8304 ~ D8323 に格納された値を Ethernet ポート 1 の IP アドレス、DNS サーバーのアドレスとして設定します。
- M8333 (CPU モジュール Ethernet ポート 2 ネットワーク設定変更トリガ) を OFF から ON にすると、D8631 ~ D8650 に格納された値を Ethernet ポート 2 の IP アドレス、DNS サーバーのアドレスとして設定します。

特殊データレジスタ	内容	読み出し / 書き込み
D8304 ~ D8307	Ethernetポート1に設定するIPアドレスの設定値。 例) IPアドレスが 192.168.0.1の場合 D8304=192、D8305=168、D8306=0、D8307=1	R/W
D8308 ~ D8311	Ethernetポート1に設定するサブネットマスクの設定値。 例) サブネットマスクが 255.255.255.0の場合 D8308=255、D8309=255、D8310=255、D8311=0	R/W
D8312 ~ D8315	Ethernetポート1に設定するデフォルトゲートウェイの設定値。 例) デフォルトゲートウェイが 192.168.0.24の場合 D8312=192、D8313=168、D8314=0、D8315=24	R/W
D8316 ~ D8319	Ethernetポート1が使用する優先DNSサーバーのアドレスの設定値。 例) 優先DNSサーバーのアドレスが 192.168.0.100の場合 D8316=192、D8317=168、D8318=0、D8319=100	R/W
D8320 ~ D8323	Ethernetポート1が使用する代替DNSサーバーのアドレスの設定値。 例) 代替DNSサーバーのアドレスが 192.168.0.101の場合 D8320=192、D8321=168、D8322=0、D8323=101	R/W
D8631 ~ D8634	Ethernetポート2に設定するIPアドレスの設定値。 例) IPアドレスが 192.168.1.1の場合 D8631=192、D8632=168、D8633=1、D8634=1	R/W
D8635 ~ D8638	Ethernetポート2に設定するサブネットマスクの設定値。 例) サブネットマスクが 255.255.255.0の場合 D8635=255、D8636=255、D8637=255、D8638=0	R/W
D8639 ~ D8642	Ethernetポート2に設定するデフォルトゲートウェイの設定値。 例) デフォルトゲートウェイが 192.168.1.24の場合 D8639=192、D8640=168、D8641=1、D8642=24	R/W
D8643 ~ D8646	Ethernetポート2が使用する優先DNSサーバーのアドレスの設定値。 例) 優先DNSサーバーのアドレスが 192.168.1.100の場合 D8643=192、D8644=168、D8645=1、D8646=100	R/W
D8647 ~ D8650	Ethernetポート2が使用する代替DNSサーバーのアドレスの設定値。 例) 代替DNSサーバーのアドレスが 192.168.1.101の場合 D8647=192、D8648=168、D8649=1、D8650=101	R/W

ネットワーク設定の確認

FC6A形のEthernetポート1およびEthernetポート2で使用する現在のIPアドレス、サブネットマスク、デフォルトゲートウェイおよびDNSサーバーのアドレスを特殊データレジスタで確認できます。

特殊データレジスタ	内容	読み出し / 書き込み
D8330 ~ D8333	Ethernetポート1で使用するIPアドレスの現在値。 例) IPアドレスが 192.168.0.1の場合 D8330=192、D8331=168、D8332=0、D8333=1	R
D8334 ~ D8337	Ethernetポート1で使用するサブネットマスクの現在値。 例) サブネットマスクが 255.255.255.0の場合 D8334=255、D8335=255、D8336=255、D8337=0	R
D8338 ~ D8341	Ethernetポート1で使用するデフォルトゲートウェイの現在値。 例) デフォルトゲートウェイが 192.168.0.24の場合 D8338=192、D8339=168、D8340=0、D8341=24	R
D8342 ~ D8345	Ethernetポート1が使用する優先DNSサーバーのアドレスの現在値。 例) 優先DNSサーバーのアドレスが 192.168.0.100の場合 D8342=192、D8343=168、D8344=0、D8345=100	R
D8346 ~ D8349	Ethernetポート1が使用する代替DNSサーバーのアドレスの現在値。 例) 代替DNSサーバーのアドレスが 192.168.0.101の場合 D8346=192、D8347=168、D8348=0、D8349=101	R
D8657 ~ D8660	Ethernetポート2で使用するIPアドレスの現在値。 例) IPアドレスが 192.168.1.1の場合 D8657=192、D8658=168、D8659=1、D8660=1	R
D8661 ~ D8664	Ethernetポート2で使用するサブネットマスクの現在値。 例) サブネットマスクが 255.255.255.0の場合 D8661=255、D8662=255、D8663=255、D8664=0	R
D8665 ~ D8668	Ethernetポート2で使用するデフォルトゲートウェイの現在値。 例) デフォルトゲートウェイが 192.168.1.24の場合 D8665=192、D8666=168、D8667=1、D8668=24	R
D8669 ~ D8672	Ethernetポート2が使用する優先DNSサーバーのアドレスの現在値。 例) 優先DNSサーバーのアドレスが 192.168.1.100の場合 D8669=192、D8670=168、D8671=1、D8672=100	R
D8673 ~ D8676	Ethernetポート2が使用する代替DNSサーバーのアドレスの現在値。 例) 代替DNSサーバーのアドレスが 192.168.1.101の場合 D8673=192、D8674=168、D8675=1、D8676=101	R

HMI モジュールの特殊データレジスタによるネットワーク設定

IP アドレス、DNS サーバーのアドレスを特殊データレジスタを使って設定する場合、FC6A 形の電源起動時および M8184 の OFF から ON への立ち上がり時に、特殊データレジスタに格納された値を設定します。

例)

M8184 (HMI モジュール ネットワーク設定変更トリガ) を OFF から ON にすると、D8437 ~ D8456 に格納された値を HMI-Ethernet ポートの IP アドレス、DNS サーバーのアドレスとして設定します。

特殊データレジスタ	内容	読み出し / 書き込み
D8437 ~ D8440	HMIモジュールのIPアドレスの設定値。 例) IPアドレスが 192.168.0.1の場合 D8437=192、D8438=168、D8439=0、D8440=1	R/W
D8441 ~ D8444	HMIモジュールのサブネットマスクの設定値。 例) サブネットマスクが 255.255.255.0の場合 D8441=255、D8442=255、D8443=255、D8444=0	R/W
D8445 ~ D8448	HMIモジュールのデフォルトゲートウェイの設定値。 例) デフォルトゲートウェイが 192.168.0.24の場合 D8445=192、D8446=168、D8447=0、D8448=24	R/W
D8449 ~ D8452	HMIモジュールが使用する優先DNSサーバーのアドレスの設定値。 例) 優先DNSサーバーのアドレスが 192.168.0.100の場合 D8449=192、D8450=168、D8451=0、D8452=100	R/W
D8453 ~ D8456	HMIモジュールが使用する代替DNSサーバーのアドレスの設定値。 例) 代替DNSサーバーのアドレスが 192.168.0.101の場合 D8453=192、D8454=168、D8455=0、D8456=101	R/W

HMI モジュールのネットワーク設定の確認

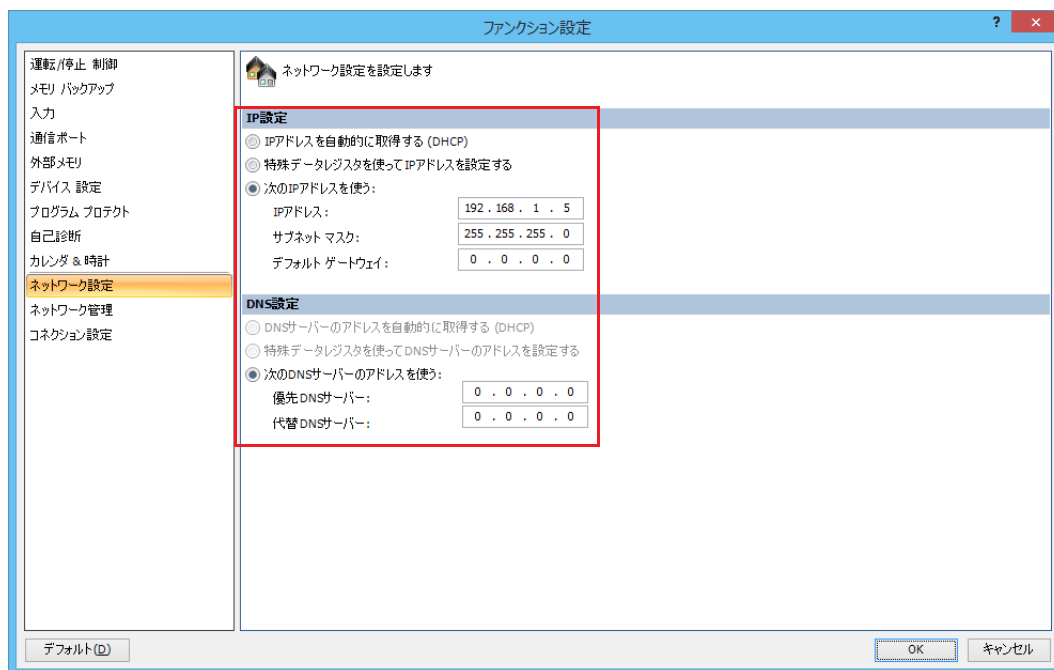
HMI モジュールの現在の IP アドレス、サブネットマスク、デフォルトゲートウェイおよび DNS サーバーのアドレスを特殊データレジスタで確認できます。

特殊データレジスタ	内容	読み出し / 書き込み
D8388 ~ D8391	HMIモジュールのIPアドレスの現在値。 例) IPアドレスが 192.168.0.1の場合 D8388=192、D8389=168、D8390=0、D8391=1	R
D8392 ~ D8395	HMIモジュールのサブネットマスクの現在値。 例) サブネットマスクが 255.255.255.0の場合 D8392=255、D8393=255、D8394=255、D8395=0	R
D8396 ~ D8399	HMIモジュールのデフォルトゲートウェイの現在値。 例) デフォルトゲートウェイが 192.168.0.24の場合 D8396=192、D8397=168、D8398=0、D8399=24	R
D8400 ~ D8403	HMIモジュールが使用する優先DNSサーバーのアドレスの現在値。 例) 優先DNSサーバーのアドレスが 192.168.0.100の場合 D8400=192、D8401=168、D8402=0、D8403=100	R
D8404 ~ D8407	HMIモジュールが使用する代替DNSサーバーのアドレスの現在値。 例) 代替DNSサーバーのアドレスが 192.168.0.101の場合 D8404=192、D8405=168、D8406=0、D8407=101	R

WindLDR の設定

●操作手順

1. [設定] タブの [ファンクション設定] で次の項目をクリックします。
 - ・ All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュール [ネットワーク設定]
 - ・ Plus CPU モジュール
Ethernet ポート 1 を設定する場合は [イーサネットポート 1]
Ethernet ポート 2 を設定する場合は [イーサネットポート 2]
[ファンクション設定] ダイアログボックスが表示されます。
2. [IP 設定]、[DNS 設定] をそれぞれ設定します。

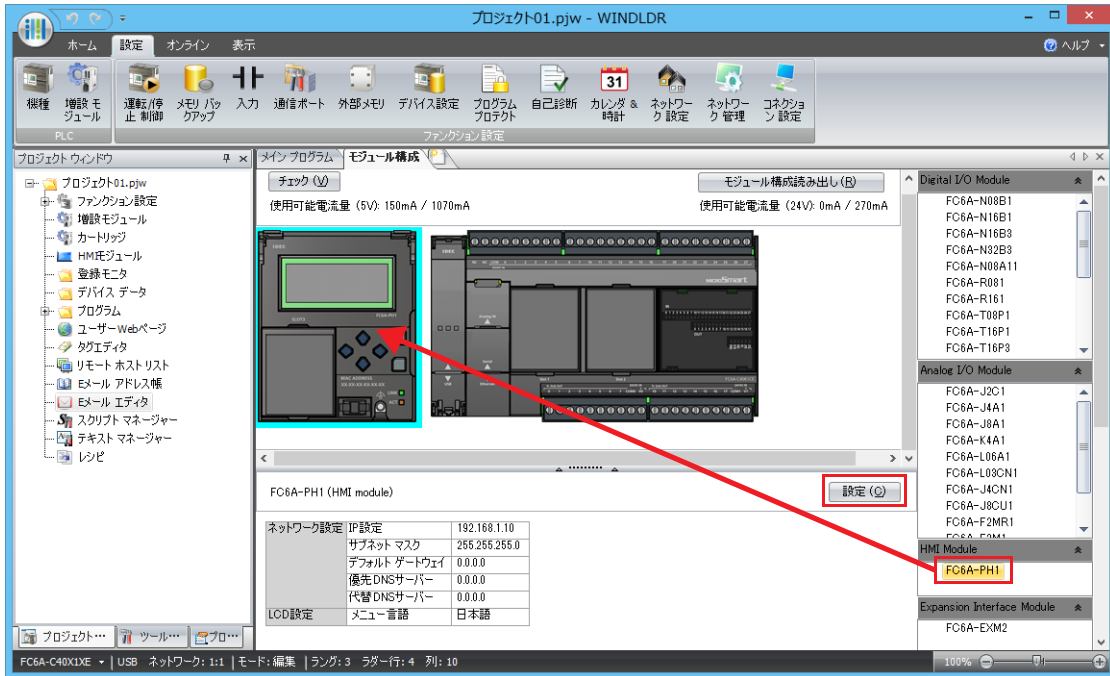


3. [OK] ボタンをクリックします。
4. HMI モジュールのネットワーク設定は、モジュール構成エディタで行います。
[設定] タブの [PLC] で [増設モジュール] をクリックします。

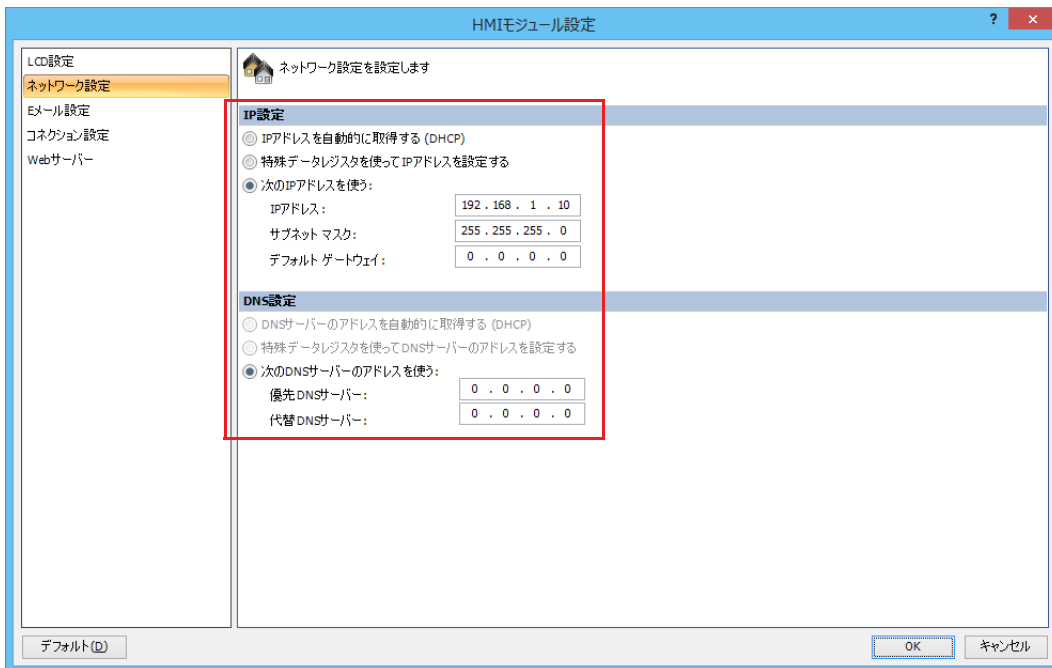
- モジュール構成エリアに挿入した HMI モジュールをクリックし、[設定] ボタンをクリックします。
[HMI モジュール設定] ダイアログボックスが表示されます。



プロジェクトウィンドウの [HMI モジュール] をダブルクリックしても [HMI モジュール設定] ダイアログボックスが表示されます。



- [ネットワーク設定] タブをクリックし、[IP 設定]、[DNS 設定] をそれぞれ設定します。



以上で、設定が完了します。

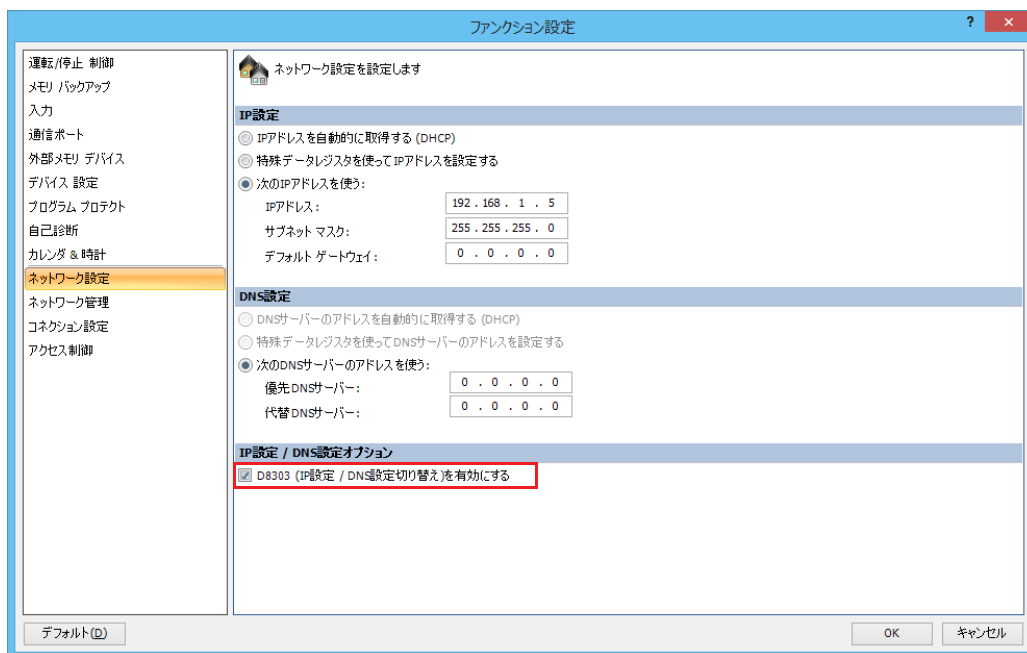
IP 設定 / DNS 設定の切り替え

特殊データレジスタ（D8303、D8630）と特殊内部リレー（M8190、M8333）を使って、CPU モジュールの Ethernet ポート 1 または Ethernet ポート 2 の IP 設定と DNS 設定を切り替えることができます。切り替え手順は次のとおりです。

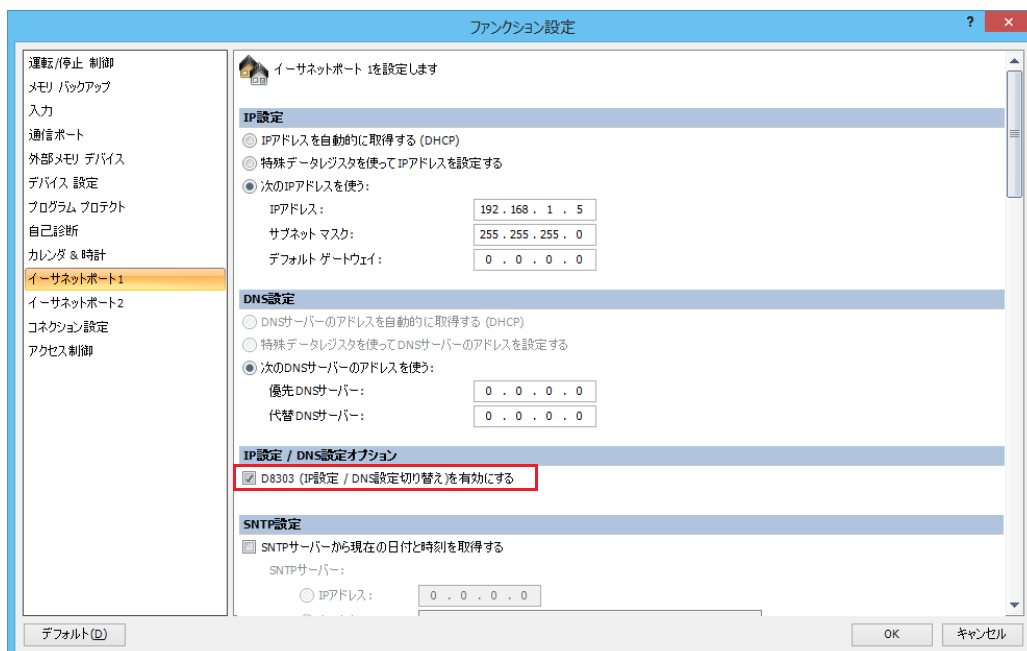
WindLDR の設定

●操作手順

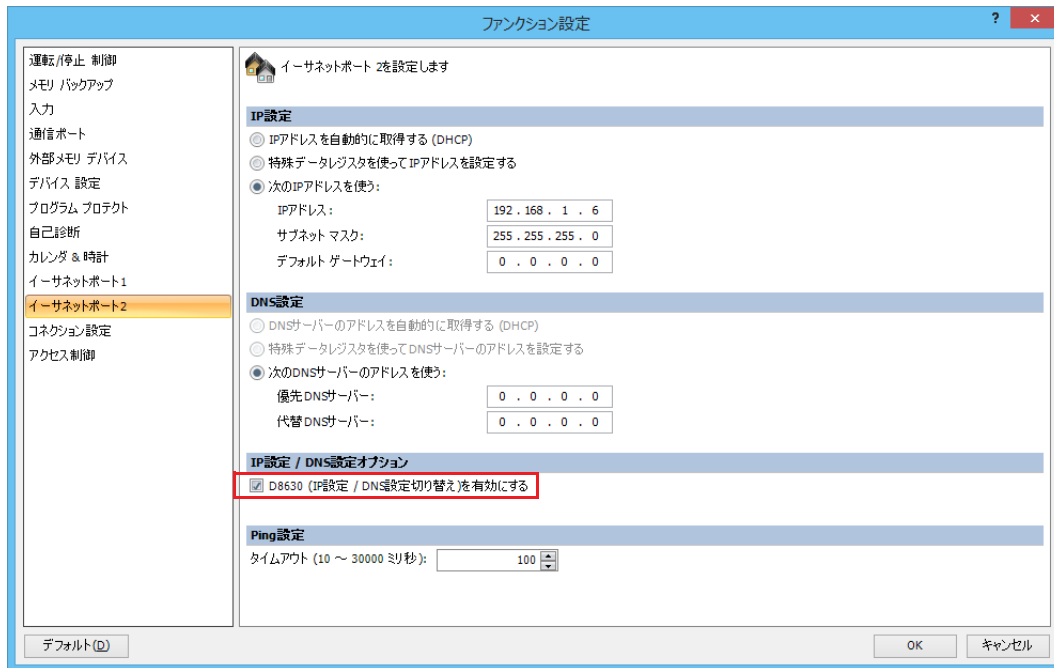
1. [設定] タブの [ファンクション設定] で次の項目をクリックします。
 - All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュール [ネットワーク設定]
 - Plus CPU モジュール
Ethernet ポート 1 を設定する場合は [イーサネットポート 1]
Ethernet ポート 2 を設定する場合は [イーサネットポート 2]
[ファンクション設定] ダイアログボックスが表示されます。
2. [IP 設定]、[DNS 設定] をそれぞれ設定します。
3. Ethernet ポート 1 を設定する場合は、[D8303 (IP 設定 / DNS 設定切り替え) を有効にする] チェックボックスをオンにします。
 - All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュール



- Plus CPU モジュール



4. Ethernet ポート 2 を設定する場合は、[D8630 (IP 設定 / DNS 設定切り替え) を有効にする] チェックボックスをオンにします。
- Plus CPU モジュール



5. [OK] ボタンをクリックします。

以上で、設定が完了します。

ユーザープログラムをダウンロードすると、D8303 または D8630 の設定値に従って、IP 設定と DNS 設定が切り替わります。D8303 と D8630 の設定値の意味は、次のとおりです。

D8303 または D8630 に格納する値の意味

設定値	IP 設定 / DNS 設定
0	ファンクション設定に従います。
1	DHCPを有効にします。
2	特殊データレジスタ (D8304~D8323またはD8631~D8650) の設定に従います。



- D8303 または D8630 の値を 2 (特殊データレジスタの設定に従う) にし、M8190 または M8333 を OFF から ON にする、または電源を再投入すると、ファンクション設定に関わらず、強制的に特殊データレジスタを使用して「IP 設定」、「DNS 設定」を設定することができます。
- D8303 または D8630 の値を 1 (DHCP を有効にする) にし、M8190 または M8333 を OFF から ON にする、または電源を再投入すると、ファンクション設定に関わらず、強制的に自動取得 (DHCP) 設定にすることができます。
- ファンクション設定を有効に戻したい場合は、D8303 または D8630 の値を 0 にし、M8190 または M8333 を OFF から ON にするか、電源を再投入してください。

ネットワーク管理

SNTP 設定

ネットワーク上の SNTP サーバーから現在時刻（UTC）を取得して、FC6A 形の内蔵時計を調整する設定について説明します。

概要

FC6A 形は、ネットワーク上の SNTP サーバーから自動または設定した間隔で現在時刻（UTC）を取得し、タイムゾーンの設定にしたがって内蔵時計を調整します。サマータイムの設定を有効にしている場合は、サマータイム期間中、SNTP サーバーから取得した現在時刻（UTC）にサマータイム補正が反映されます。

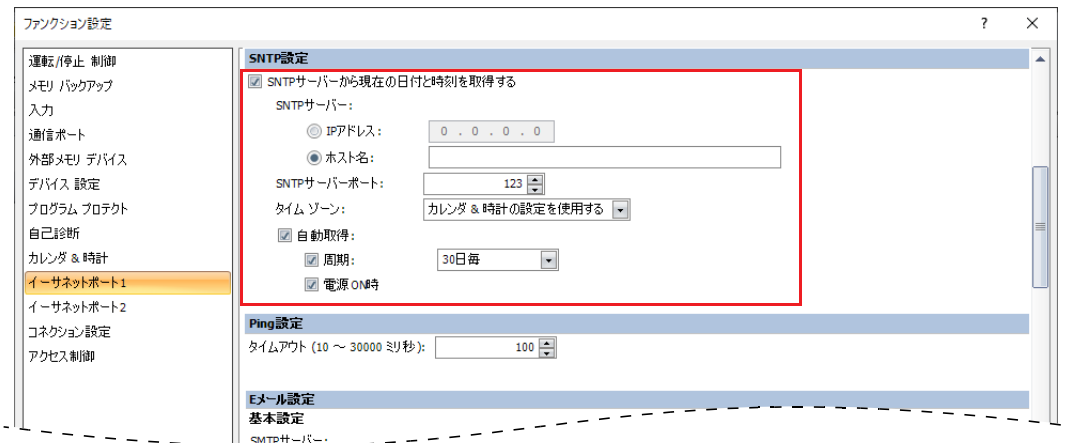
特殊データレジスタを使用して、タイムゾーンの調整や SNTP サーバーの動作ステータスの確認もできます。

WindLDR の設定

現在時刻を取得する SNTP サーバーと現在時刻の取得方法を設定します。

●操作手順

- [設定] タブの [ファンクション設定] で次の項目をクリックします。
 - All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュール
[ネットワーク管理]
 - Plus CPU モジュール
[イーサネットポート 1]
- [SNTP サーバーから現在の日付と時刻を取得する] チェックボックスをオンにします。



- SNTP サーバー、タイムゾーン、現在時刻の取得方法を設定します。

項目	設定値
SNTPサーバー (IPアドレス)	現在時刻を取得するSNTPサーバーのIPアドレスを設定します。 入力形式は“xxx.xxx.xxx.xxx”となります。“xxx”には0から255までの値を入力します。
SNTPサーバー (ホスト名)	現在時刻を取得するSNTPサーバーのホスト名を設定します。 ホスト名は半角英数字で最大40文字まで入力できます。
SNTPサーバーポート	SNTPサーバーのポート番号 (0 ~ 65535) を設定します。
タイムゾーン	SNTPサーバーで取得した世界協定時 (UTC) に対する地域の時間帯の差を設定します。 “カレンダー&時計の設定を使用する” ^{*1} 、または“UTC-12:00”~“UTC+14:00”の範囲で設定できます。

*1 [ファンクション設定] ダイアログボックスの [カレンダー&時計] タブで設定したタイムゾーンを使用します。詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 タイムゾーン」を参照してください。



タイムゾーンは、タイムゾーンオフセット (D8413) を使用して 15 分単位で調整できます。詳細は、「タイムゾーンの調整 (D8413)」(3-12 頁) を参照してください。

- 現在時刻を自動取得する場合は、[自動取得] チェックボックスをオンにします。
キープデータエラー発生時およびユーザープログラムをダウンロードして自動取得がオフからオンに変更された場合に、自動的に SNTP サーバーから現在時刻を取得します。
上記以外のタイミングで定期的に現在時刻を取得する場合は、[周期] チェックボックスをオンにし、“10 分毎”、“毎時”、“毎日”、“毎週”、“30 日毎”から選択します。
また、FC6A 形の電源を ON したときに、SNTP サーバーから現在時刻を取得する場合は、[電源 ON 時] チェックボックスをオンにします。
[自動取得] チェックボックスがオフの場合は、SNTP 取得フラグ (M8191) を OFF から ON にすると、SNTP サーバーから現

在時刻を取得します。

5. [OK] ボタンをクリックします。

以上で、設定が完了します。

タイムゾーンの調整 (D8413)

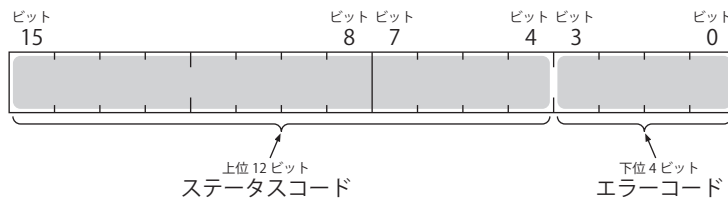
タイムゾーンオフセット (D8413) を使用して、設定したタイムゾーン (UTC-12:00 ~ UTC+14:00) を 15 分単位で調整できます。たとえば、タイムゾーンに "UTC+09:00" を設定している場合、D8413 の値が +1 なら、調整後のタイムゾーンは +15 分 (15 分 × (+1)) を加えた「UTC+09:15」になります。

D8413 の値が -2 の場合、調整後のタイムゾーンに -30 分 (15 分 × (-2)) を加えた「UTC+08:30」になります。

動作ステータスの確認 (D8414)

SNTP 動作ステータス (D8414) には、SNTP サーバーの動作ステータスが格納されます。

動作ステータスは、動作状態 (ステータスコード) およびエラー内容 (エラーコード) を示します。ステータスコードは D8414 の上位 12 ビット、エラーコードは D8414 の下位 4 ビットに格納されます。



ステータスコード	動作の状態	状態説明
000h (000000000000)	無処理	SNTPサーバーにアクセスしていない状態
020h (000000100000)	応答待ち	SNTPサーバーにアクセスし、SNTPサーバーからの返答を待っている状態
040h (000001000000)	時刻情報取得成功	SNTPサーバーからの返答を正常に受信した状態

エラーコード	エラー内容
0h (0000)	正常
2h (0010)	タイムアウトエラー
3h (0011)	設定されたSNTPサーバーのIPアドレスをDNS解決できなかった
9h (1001)	不正なデータを受信した

任意のタイミングで現在時刻を取得 (M8191)

SNTP 取得フラグ (M8191) が OFF から ON になったとき、SNTP サーバーから現在時刻を取得します。

現在時刻を取得してからの経過時間 (D8415)

SNTP アクセス経過時間 (D8415) には、SNTP サーバーから現在時刻を最後に取得してから現在までの経過時間 (0 ~ 65535) が分単位で格納されます。経過時間が 65535 分を超えた場合は、計測値を 0 にリセットして計測を続けます。格納できる値が 0 ~ 65535 のため、65535 分 ÷ 60 分 ÷ 24 時間 = 約 45 日間となり、最大 45 日間まで計測できます。

たとえば、SNTP サーバーから最後に現在時刻を取得した日時が 1 月 1 日 12 時 00 分の場合、1 月 1 日 15 時 00 分に D8415 を確認すると、180 が格納されています。

D8415 は、現在時刻取得成功時に 0 にリセットされ、経過時間のカウントが開始されます。SNTP サーバーから現在時刻の取得が一度も成功していない場合は、D8415 の値は更新されません。

PING 設定

PING 命令または自動 Ping 機能実行時の各 Ping のタイムアウト時間の設定について説明します。

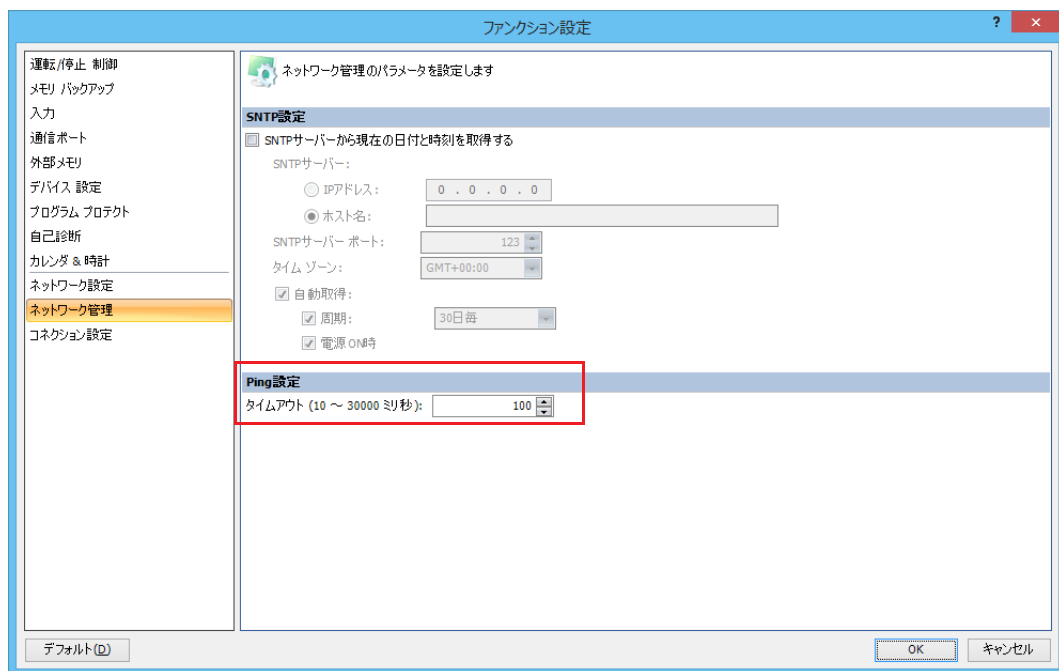
概要

PING 命令または自動 Ping 機能で Ping を実行すると、設定したリモートホストに向けて Ping パケットを送信し、IP レベルで通信できるかを確認できます。その Ping を送信してからタイムアウトになるまでの時間を設定します。

WindLDR の設定

● 操作手順

- [設定] タブの [ファンクション設定] で次の項目をクリックします。
 - All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュール
[ネットワーク管理]
 - Plus CPU モジュール
Ethernet ポート 1 を設定する場合は [イーサネットポート 1]
Ethernet ポート 2 を設定する場合は [イーサネットポート 2]
- [タイムアウト (10 ~ 30000 ミリ秒)] で PING 命令のタイムアウト時間を 10 ~ 30000ms の間で設定します。デフォルトは 100ms です。



- [OK] ボタンをクリックします。

以上で、設定が完了します。

コネクション設定

FC6A形のEthernetポート1、Ethernetポート2およびHMI-Ethernetポートを使用したTCP/IP通信におけるクライアント/サーバー通信の設定について説明します。

概要

All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールは、Ethernet ポート 1 に最大 8 個のコネクションを設定できません。Plus CPU モジュールは、Ethernet ポート 1 および Ethernet ポート 2 に最大 16 個のコネクションを設定できます。各コネクションに設定できる通信については下表を参照してください。

HMI モジュールを使用する場合は、HMI-Ethernet ポートに最大 8 個のコネクションを設定できます。ただし、コネクションにはメンテナンス通信のみ設定できます。

通信を行うために必要なコネクションの設定は、[ファンクション設定] ダイアログボックスで行います。

コネクションをサーバーとして使用する場合、コネクションに対して IP アドレスによるフィルタリングを設定できます。特定の IP アドレスを持つ外部機器のみが FC6A 形と通信でき、不特定のクライアントからのアクセスを制限できます。

Ethernet ポートで使用できる通信モード

各 Ethernet ポートで対応する通信モードは、次のとおりです。

通信モード	All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュール	Plus CPU モジュール		HMI モジュール
	Ethernet ポート 1	Ethernet ポート 1	Ethernet ポート 2	HMI-Ethernet ポート
メンテナンス通信サーバー (デフォルト)	○	○	○	○
ユーザー通信サーバー	○	○	○	×
ユーザー通信クライアント	○	○	○	×
ユーザー通信 (UDP)	×	○	○	×
Modbus TCP サーバー	○	○	○	×
Modbus TCP クライアント	○	○	○	×
MC プロトコル クライアント	×	○	○	×
未使用	○	○	○	○

コネクション数

モジュール	コネクション数
All-in-One CPUモジュール/ CAN J1939 All-in-One CPUモジュール	最大8個
Plus CPUモジュール	最大16個 ^{*1}
HMIモジュール	最大8個

*1 Modbus TCP クライアントを使用する場合、1 つコネクション設定に最大 255 個の通信相手を登録できます。同時に設定できるコネクション設定の数は最大 16 個です。各コネクション設定に登録された通信相手のうち、合計 128 個の通信相手とはコネクションを維持したまま通信相手を切り替えて通信できます。詳細は「第 6 章 Modbus TCP クライアント」を参照してください。

MCプロトコル クライアントを使用する場合は、1つコネクション設定に最大255個の通信相手を登録できます。MCプロトコル通信は、通信相手が変わるたびにコネクションを切断および接続します。

通信モードの概要

各通信モードの概要は、次のとおりです。

通信モード	概要	参照頁
メンテナンス通信サーバー (デフォルト)	WindLDRからデバイスのモニタや変更、ユーザープログラムのダウンロードやアップロードができます。	4-7
ユーザー通信サーバー	ETXD (イーサネットユーザー通信送信) 命令およびERXD (イーサネットユーザー通信受信) 命令にしたがって、クライアント機器と通信できます。	5-45
ユーザー通信クライアント	ETXD (イーサネットユーザー通信送信) 命令およびERXD (イーサネットユーザー通信受信) 命令にしたがって、サーバー機器と通信できます。	5-42
ユーザー通信 (UDP) *1	ETXD (イーサネットユーザー通信送信) 命令およびERXD (イーサネットユーザー通信受信) 命令にしたがって、UDP通信機器と通信できます。	5-47
Modbus TCP サーバー	Modbus TCPクライアント対応機器からFC6A形のデータのモニタや変更が実行できます	6-25
Modbus TCP クライアント	FC6A形からModbus TCPサーバー対応機器のデータのモニタや変更が実行できます。	6-19
MC プロトコル クライアント	Plus CPUモジュールからMCプロトコル対応機器のデータのモニタや変更が実行できます。	18-2
未使用	コネクションを使用しません。内部の制御処理を停止するため、他のコネクションの応答性が向上します。	—

*1 Plus CPU モジュールのみ選択できます。

コネクションのステータス情報および接続 IP アドレス

外部機器との接続の状態は M8212 ~ M8231 および M8345 ~ M8364 で確認できます。サーバー、またはクライアント機器と接続しているときは ON、接続していないときは OFF になります。また、D8350 ~ D8381 で接続相手の IP アドレスを確認できます。



「R/W は、Read (読み出し) /Write (書き込み) の略で、R/W の場合は読み出し・書き込み可能、R の場合は読み出しのみ可能、W の場合は書き込みのみ可能です。

ステータス情報

特殊内部リレー	内容		読み出し / 書き込み
M8212	コネクション1ステータス	サーバー、またはクライアント機器と接続有りのときON、接続無し的时候OFFします。	R
M8213	コネクション2ステータス		
M8214	コネクション3ステータス		
M8215	コネクション4ステータス		
M8216	コネクション5ステータス		
M8217	コネクション6ステータス		
M8220	コネクション7ステータス		
M8221	コネクション8ステータス		
M8345*1	コネクション9ステータス		
M8346*1	コネクション10ステータス		
M8347*1	コネクション11ステータス		
M8350*1	コネクション12ステータス		
M8351*1	コネクション13ステータス		
M8352*1	コネクション14ステータス		
M8353*1	コネクション15ステータス		
M8354*1	コネクション16ステータス		
M8222	コネクション1切断フラグ	OFFからONにするとサーバーとのコネクションを強制的に切断し、通信を停止します。ONからOFFにするとサーバーへコネクションを確立し、通信を再開します。ユーザー通信クライアントでのみ有効です。	R/W
M8223	コネクション2切断フラグ		
M8224	コネクション3切断フラグ		
M8225	コネクション4切断フラグ		
M8226	コネクション5切断フラグ		
M8227	コネクション6切断フラグ		
M8230	コネクション7切断フラグ		
M8231	コネクション8切断フラグ		
M8355*1	コネクション9切断フラグ		
M8356*1	コネクション10切断フラグ		
M8357*1	コネクション11切断フラグ		
M8360*1	コネクション12切断フラグ		
M8361*1	コネクション13切断フラグ		
M8362*1	コネクション14切断フラグ		
M8363*1	コネクション15切断フラグ		
M8364*1	コネクション16切断フラグ		

*1 Plus CPU モジュールのみ使用できます。

接続 IP アドレス

特殊データレジスタ	内容		読み出し / 書き込み
D8350～D8353	コネクション1接続IPアドレス	外部機器と各コネクションとの通信が確立している場合、接続している外部機器のIPアドレスが各特殊データレジスタに以下のように格納されます。 例) IPアドレス：aaa.bbb.ccc.ddd、 D8350=aaa、D8351=bbb、D8352=ccc、D8353=ddd	R
D8354～D8357	コネクション2接続IPアドレス		
D8358～D8361	コネクション3接続IPアドレス		
D8362～D8365	コネクション4接続IPアドレス		
D8366～D8369	コネクション5接続IPアドレス		
D8370～D8373	コネクション6接続IPアドレス		
D8374～D8377	コネクション7接続IPアドレス		
D8378～D8381	コネクション8接続IPアドレス		
D8677～D8680*1	コネクション9接続IPアドレス		
D8681～D8684*1	コネクション10接続IPアドレス		
D8685～D8688*1	コネクション11接続IPアドレス		
D8689～D8692*1	コネクション12接続IPアドレス		
D8693～D8696*1	コネクション13接続IPアドレス		
D8697～D8700*1	コネクション14接続IPアドレス		
D8701～D8704*1	コネクション15接続IPアドレス		
D8705～D8708*1	コネクション16接続IPアドレス		

*1 Plus CPU モジュールのみ使用できます。

HMI モジュールでのコネクションのステータス情報および接続 IP アドレス

D8429 で設定した HMI モジュールのコネクション番号の接続状態を、特殊内部リレー、特殊データレジスタに格納します。

HMI モジュールの外部機器との接続の状態は M8232 で確認できます。

外部機器と接続しているときは ON、接続していないときは OFF になります。

D8430 ～ D8433 で接続相手の IP アドレスを確認できます。

読み出し対象コネクション番号設定

特殊内部リレー	内容		読み出し / 書き込み
D8429	HMIモジュール コネクション情報参照 コネクション番号	設定した番号のコネクション情報を、D8430～D8434、M8232に反映します。 0を設定すると、対象のデバイスをすべて0にします。 存在しないコネクション番号を設定した場合は、0を設定したときと同じ動作をします。	R/W

ステータス情報

特殊内部リレー	内容		読み出し / 書き込み
M8232	HMIモジュール コネクション情報参照 コネクションステータス	D8429で設定したコネクションが、外部機器と接続ありのときON、接続なしのときOFFします。	R

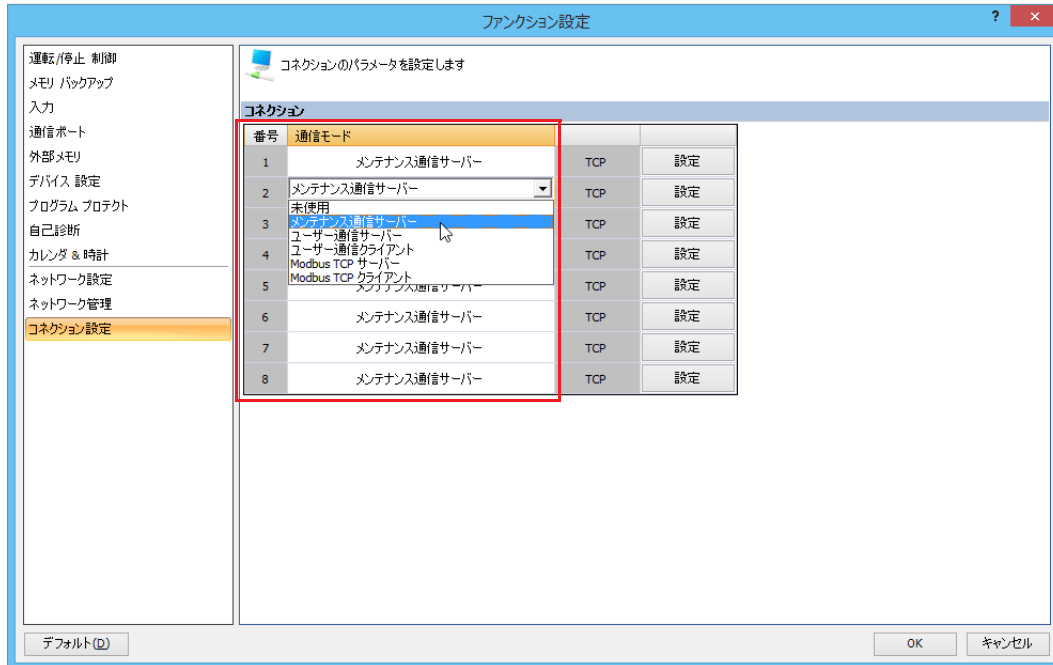
接続 IP アドレス

特殊内部リレー	内容		読み出し / 書き込み
D8430 ～ D8433	HMIモジュール コネクション情報参照 接続IPアドレス	外部機器とD8429で設定したコネクションとの通信が確立している場合、接続している外部機器のIPアドレスが各特殊データレジスタに以下のように格納されます。 例) IPアドレス：aaa.bbb.ccc.ddd、 D8430=aaa、D8431=bbb、D8432=ccc、 D8433=ddd	R

WindLDR の設定

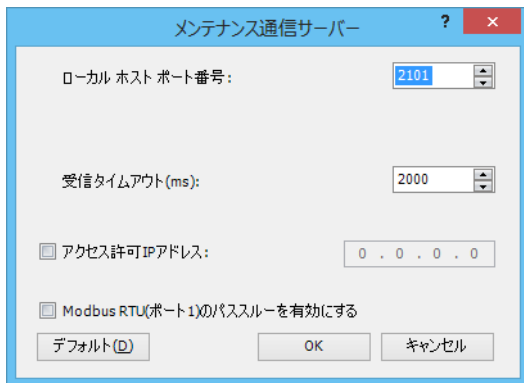
●操作手順（メンテナンス通信サーバーの場合）

1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するコネクション番号の [通信モード] をクリックし、通信モードを選択します。



クライアント、サーバーとして使用できる All-in-One CPU モジュールは最大 8 個、Plus CPU モジュールは最大 16 個のコネクションに割り当てる通信モードを設定します。各コネクションには異なるポート番号を設定できます。使用しないコネクションは、“未使用”に設定してください。通信モードに応じたダイアログボックスが表示されます。

3. 通信相手機器の通信フォーマットに合わせて、各パラメータを設定します。

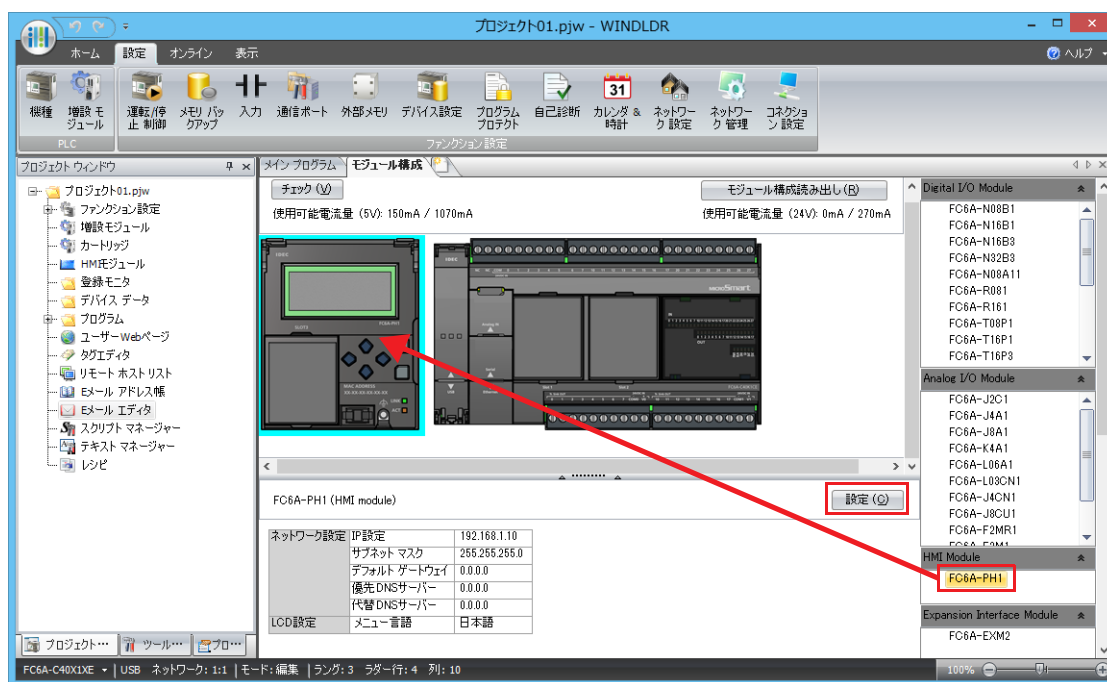


4. [OK] ボタンをクリックします。
5. HMI モジュールのコネクション設定は、モジュール構成エディタで行います。
[設定] タブの [PLC] で [増設モジュール] をクリックします。

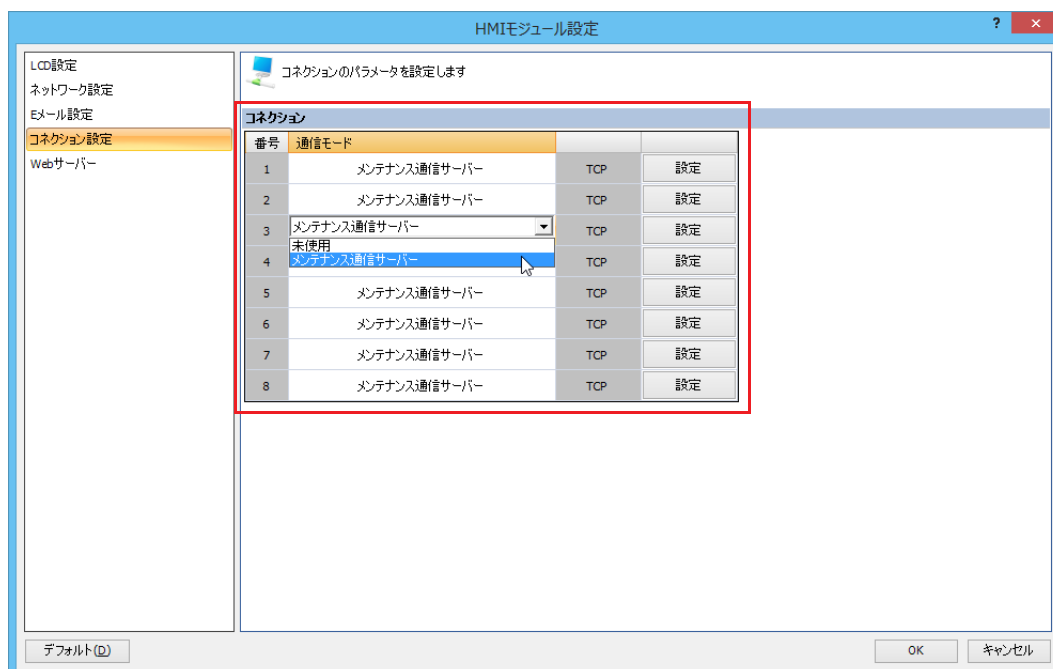
- モジュール構成エリアに挿入したHMIモジュールをクリックし、[設定] ボタンをクリックします。
[HMIモジュール設定] ダイアログボックスが表示されます。



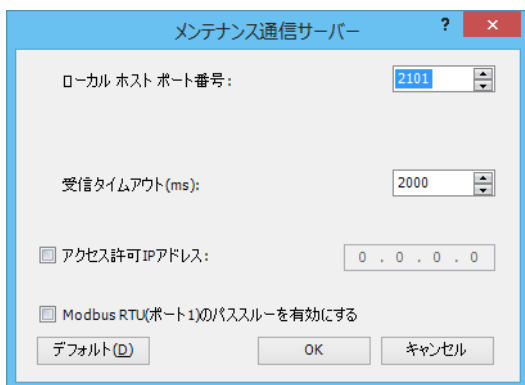
プロジェクトウィンドウの [HMIモジュール] をダブルクリックしても [HMIモジュール設定] ダイアログボックスが表示されます。



- [コネクション設定] タブをクリックします。
- 使用するコネクション番号の [通信モード] をクリックし、通信モードを選択します。



9. 通信相手機器の通信フォーマットに合わせて、各パラメータを設定します。



10. [OK] ボタンをクリックします。

以上で、設定が完了します。

リモートホストリスト

リモートホストリスト

ネットワーク上の通信相手の機器（リモートホスト）を登録/管理するリモートホストリストの設定について説明します。

概要

FC6A形は、次のイーサネット通信の機能を使用する場合は、通信相手の機器をリモートホストとしてリモートホストリストに登録する必要があります。

- PING 命令
- ETXD 命令 / ERXD 命令（ユーザー通信クライアント）
- Modbus TCP クライアント
- MC プロトコル クライアント

リモートホストを登録すると、リモートホストの番号が自動的に割り当てられます。

上記の機能を使用する際には、通信相手の機器としてリモートホスト番号を設定します。

WindLDR の設定

●操作手順

1. [表示] タブの [ワークスペース] で [プロジェクトウィンドウ] をクリックします。
画面左にプロジェクトウィンドウが表示されます。
2. [リモートホストリスト] をダブルクリックします。
[リモートホストリスト] ダイアログボックスが表示されます。
3. [新規] ボタンをクリックします。
[リモートホスト] ダイアログボックスが表示されます。



4. [リモートホスト] ダイアログボックスの各項目に入力し、[追加] ボタンをクリックします。

リモートホストは、次の項目で構成されます。

項目	設定値
IPアドレス	リモートホストをIPアドレスで設定します。FC6A形は設定されたIPアドレスとそのポート番号に対して接続を確立し、通信を行います。
ホスト名	リモートホストをホスト名で設定します。半角英数40文字まで入力できます。
データレジスタ	リモートホストのIPアドレスをデータレジスタ（4ワード）で設定します。
ポート番号	リモートホストが開いているポート番号を設定します。このポート番号はTCP通信のポート番号であり、FC6A形のUSBポート、ポート1～3の番号とは異なります。
コメント	リモートホストを説明するコメントです。入力内容および、長さは通信に影響しません。
イーサネットポート	リモートホストと通信するイーサネットポートを選択します。

リモートホストが登録され、リモートホストリストに反映されます。引き続きリモートホストを登録する場合は、同じ操作を繰り返してください。

5. 設定したリモートホストに対して周期的に Ping を実行する場合は、自動 Ping を設定します。
 詳細は、「自動 Ping 機能」（3-23 頁）を参照してください。
6. 登録がすべて完了したら、[閉じる] ボタンをクリックします。

番号	リモートホスト	ポート番号	コメント	オンライン状態	動作ステータス	イーサネットポート
1	192.168.1.6	0		<input checked="" type="checkbox"/>		1

7. 登録したリモートホストを削除する場合は、[リモートホストリスト] ダイアログボックスを開き、削除するリモートホストを選択して [削除] ボタンをクリックします。

以上で、設定が完了します。

自動 Ping 機能

設定したリモートホストに対して周期的に Ping を実行する自動 Ping 機能について説明します。

概要

設定したリモートホストに対して周期的に Ping を実行する機能を自動 Ping 機能と呼びます。この機能を使用し、設定したリモートホストに向けてパケットを送信することで、ホストと通信可能かどうかを確認できます。自動 Ping 機能の結果を Modbus TCP クライアントおよび MC プロトコル クライアントのリクエスト処理と連動させることができます。Modbus TCP クライアントとの連動の詳細は、「第 6 章 Modbus TCP クライアント」を参照してください。MC プロトコル クライアントとの連動の詳細は、「第 18 章 イーサネット通信による MC プロトコル通信」を参照してください。

リモートホストは、リモートホストリストで設定します。

自動 Ping が有効の場合、FC6A 形の電源が ON およびユーザープログラムの変更によって自動 Ping が有効になった直後に、自動 Ping の実行を開始し、設定したリモートホスト番号の最も小さい番号から順に Ping を実行します。FC6A 形の RUN/STOP 状態やラダープログラムの影響を受けません。

Ethernet ポート 1 では、Ethernet ポート 1 自動 Ping 実行中 (M8186 が ON) の場合、Ethernet ポート 1 自動 Ping 停止フラグ (M8187) を OFF から ON すると、自動 Ping が停止します。M8187 を OFF すると自動 Ping の実行が開始されます。

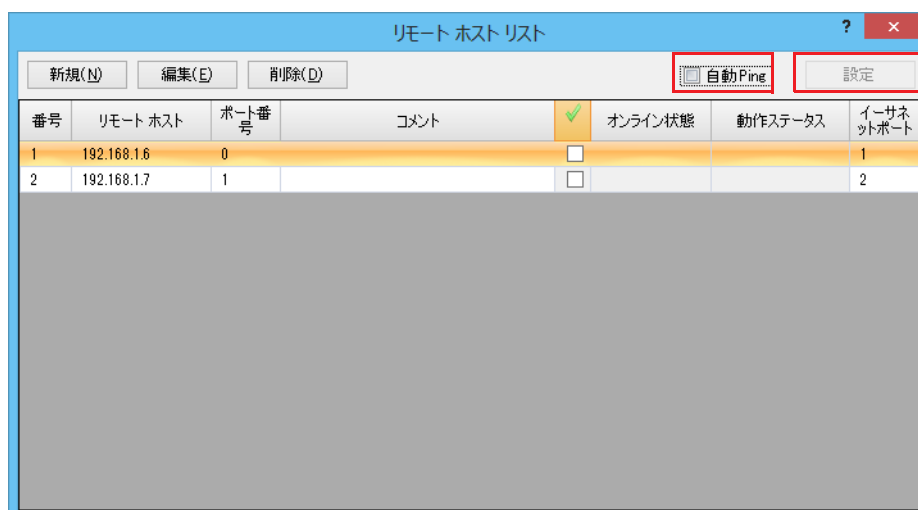
Ethernet ポート 2 では、Ethernet ポート 2 自動 Ping 実行中 (M8331 が ON) の場合、Ethernet ポート 2 自動 Ping 停止フラグ (M8332) を OFF から ON すると、自動 Ping が停止します。M8332 を OFF すると自動 Ping の実行が開始されます。Ethernet ポート 2 は Plus CPU モジュールのみ使用できます。

WindLDR の設定

●操作手順

1. [リモートホストリスト] ダイアログボックスで [自動 Ping] チェックボックスをオンにし、[設定] ボタンをクリックします。

[自動 Ping 設定] ダイアログボックスが表示されます。



2. 自動 Ping 機能の各項目を設定し、[OK] ボタンをクリックします。

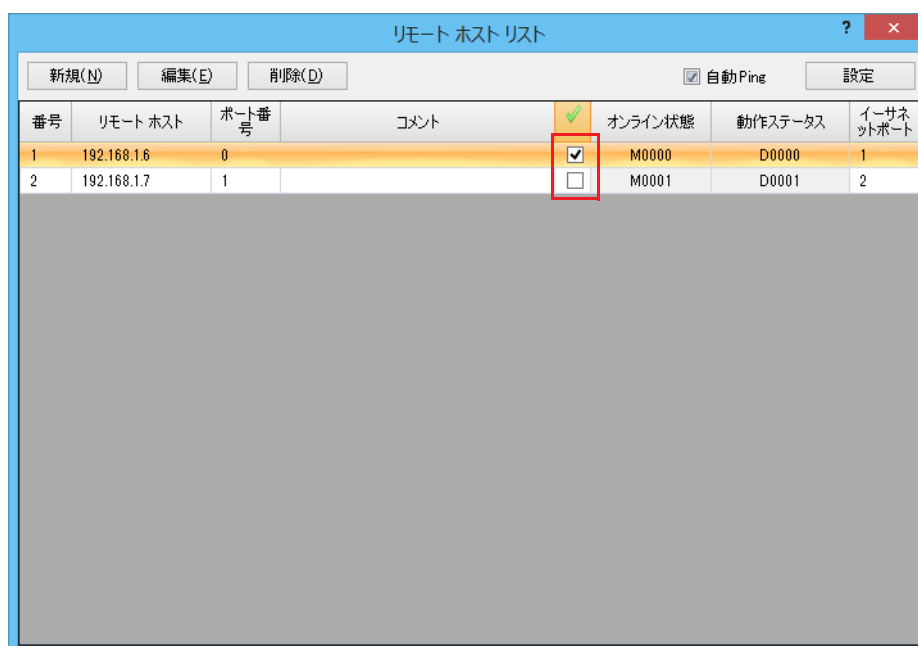
自動 Ping 機能は、次の項目で構成されます。

項目	説明
オンライン状態 ^{*1}	自動Ping機能で送信したPingの結果を格納する内部リレーを設定します。設定した内部リレーを先頭にして、内部リレーがリモートホスト番号順に自動的に割り付きます。
動作ステータス	自動Ping機能で送信したPingの動作ステータスを格納するデータレジスタを設定します。設定したデータレジスタを先頭にして、データレジスタがリモートホスト番号順に自動的に割り付きます。 動作ステータスは、動作の状態（ステータスコード）とエラー内容（エラーコード）を示します。ステータスコードは上位12ビットに格納し、エラーコードは下位4ビットに格納されます。ステータスコードおよびエラーコードの内容は「動作ステータス」(3-26頁)を参照してください。 PING命令の動作は影響しません。
リトライ	自動Ping機能で送信したPingのリトライ回数 (0~5回) を設定します。
Ping間隔	自動Ping機能で送信する各Pingの送信間隔 (0~60000ms) を10ms単位で設定します。
自動Ping間隔	自動Ping実行後、次の自動Pingを実行するまでの間隔 (100~6000000ms) を100ms単位で設定します。

*1 [オンライン状態] は自動 Ping 機能の実行開始時に自動的に OFF になります。さらに Ping 送信後、応答を受信した時点で [オンライン状態] が ON し、応答を受信できなくなった時点で自動的に OFF になります。[オンライン状態] は自動 Ping が停止するとき、直前の ON/OFF 状態を保持します。自動 Ping 機能で使用する特殊内部リレーおよび [オンライン状態] の状態は次のとおりです。

リモートホストと通信するイーサネットポート	[オンライン状態] の状態
Ethernetポート1	Ethernetポート1自動Ping実行中 (M8186がON) の場合、Ethernetポート1自動Ping停止フラグ (M8187) をOFFからONして自動Pingが停止すると、直前のON/OFF状態を保持します。 自動Pingが停止中 (M8186がOFF) の場合、Ethernetポート1自動Ping停止フラグ (M8187) をONからOFFにすると、[オンライン状態] はOFFし、OFF状態を保持します。
Ethernetポート2 (Plus CPUモジュールのみ)	Ethernetポート2自動Ping実行中 (M8331がON) の場合、Ethernetポート2自動Ping停止フラグ (M8332) をOFFからONして自動Pingが停止すると、直前のON/OFF状態を保持します。 自動Pingが停止中 (M8331がOFF) の場合、Ethernetポート2自動Ping停止フラグ (M8332) をONからOFFにすると、[オンライン状態] はOFFし、OFF状態を保持します。

3. [リモートホストリスト] ダイアログボックスで、自動 Ping 実行時に Ping を送信するリモートホストのチェックボックスをオンにします。



4. [閉じる] ボタンをクリックします。

以上で、設定が完了します。

動作ステータス

動作ステータスは、動作の状態（ステータスコード）とエラー内容（エラーコード）を示します。

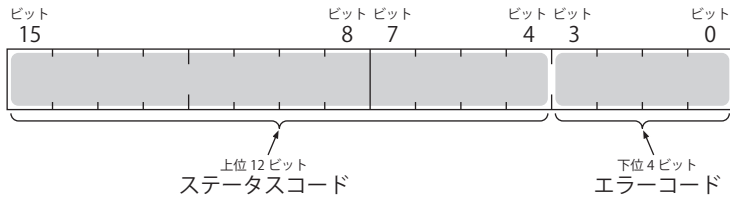
動作の状態（ステータスコード）は D8414 の上位 12 ビット、エラー内容（エラーコード）は D8414 の下位 4 ビットに格納されます。

動作ステータスのデータレジスタの値を 16 で割ると、その余りがエラーコードとなります。

[動作ステータスのデータレジスタの値が“66”の場合]

$$66 \div 16 = 4 \text{ 余り } 2$$

となり、エラーコードは“2”となります。



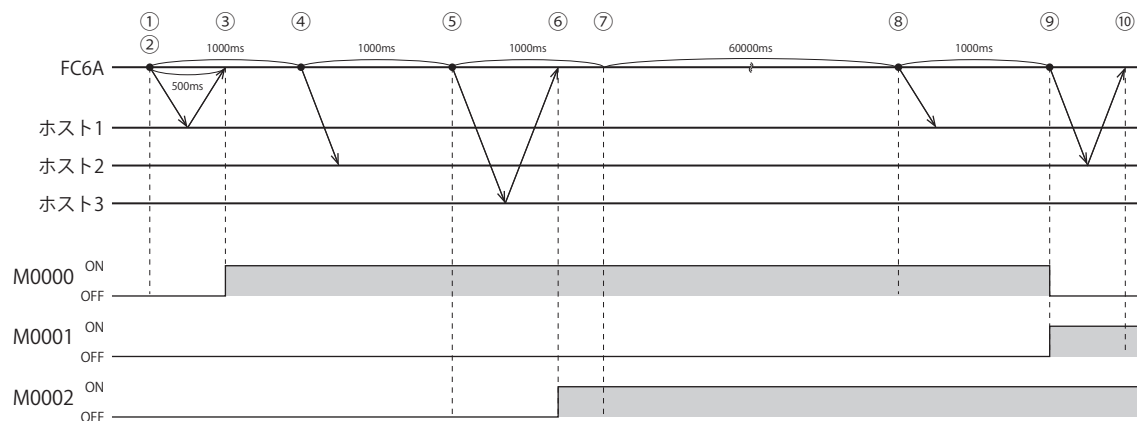
ステータスコード	動作の状態
16 (000000010000)	Ping 実行後、パケット送信前の状態
32 (000000100000)	パケットの送信処理完了後、ホストからの応答を待機している状態
64 (000001000000)	パケットに対する応答の受信が正常に完了、またはタイムアウトエラーが発生し、次の Ping が実行可能な状態

エラーコード	エラー内容
0 (0000)	正常
2 (0010)	タイムアウトエラー
3 (0011)	宛先のホスト名を DNS 解決できなかった

自動 Ping 動作例 1

次の条件で 3 つのリモートホストに対して自動 Ping を実行した場合の動作について説明します。

設定項目	設定内容
Ping設定	タイムアウト 1000ms
自動Ping設定	接続状態 M0000
	リトライ 0回
	Ping間隔 1000ms
	自動Ping実行間隔 60000ms

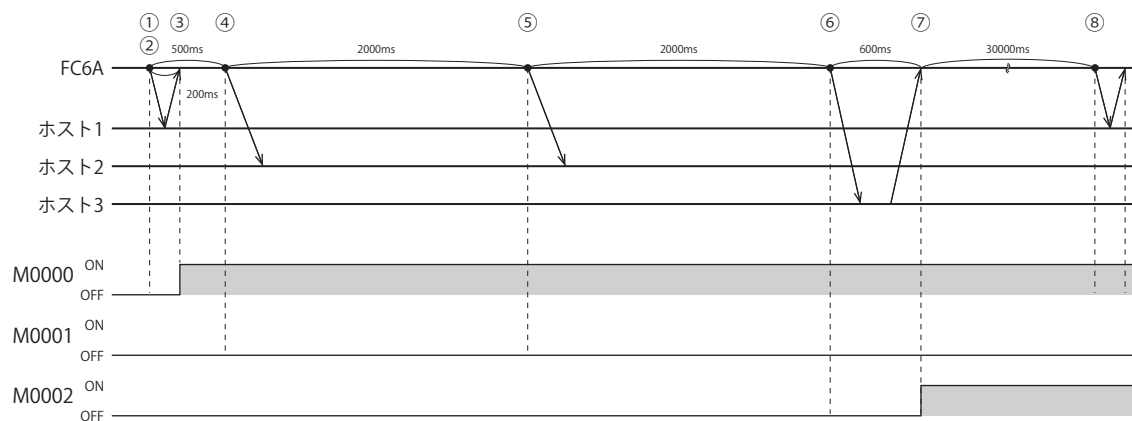


- ① : 自動 Ping 動作開始
- ②~③ : ホスト 1 からタイムアウト時間内に応答あり
- ④ : ホスト 2 からタイムアウト時間内に応答なし
- ⑤~⑥ : ホスト 3 からタイムアウト時間内に応答あり
- ⑦~⑧ : 自動 Ping 実行間隔
- ⑧ : ホスト 1 からタイムアウト時間内に応答なし
- ⑨~⑩ : ホスト 2 からタイムアウト時間内に応答あり

自動 Ping 動作例 2

次の条件で3つのリモートホストに対して自動 Ping を実行した場合の動作について説明します。

設定項目		設定内容
Ping設定	タイムアウト	2000ms
自動Ping設定	接続状態	M0000
	リトライ	1回
	Ping間隔	500ms
	自動Ping実行間隔	30000ms



- ① : 自動 Ping 動作開始
- ②~③ : ホスト 1 からタイムアウト時間内に応答あり
- ④~⑤ : ホスト 2 からタイムアウト時間内に応答なし、リトライしても応答なし
Ping 間隔よりもタイムアウト時間が長い場合は、Ping 間隔の待ち時間はありません。
- ⑥ : ホスト 3 からの応答が遅れたが、タイムアウト時間内に応答あり
- ⑦~⑧ : 自動 Ping 実行間隔

第4章 メンテナンス通信

この章では、FC6A 形のメンテナンス通信について説明します。

概要

メンテナンス通信は、IDEC 製プログラマブルコントローラ専用の通信プロトコルです。WindLDR や IDEC 製プログラマブル表示器が FC6A 形と通信する際に使用します。

FC6A 形は、それぞれの通信ポートでメンテナンス通信をサポートしており、様々なシステム構成において、最適な通信方法を選択できます。

メンテナンス通信でできること

FC6A 形とメンテナンス通信して使用できる機能は、次のとおりです。

項目	内容
ユーザープログラムのダウンロード	WindLDR で作成したユーザープログラムを、FC6A 形に書き込みます。 FC6A 形マイクロスマート ユーザーズ マニュアル「第4章 ユーザープログラムのダウンロード」を参照してください。
ユーザープログラムのアップロード	FC6A 形に保存されているユーザープログラムを読み出します。
デバイス値のモニタ・変更	デバイス値を表示したり、FC6A 形の動作を確認しながら、WindLDR でユーザープログラムのデバイス値を変更できます。 FC6A 形マイクロスマート ユーザーズ マニュアル「第4章 動作確認」を参照してください。
システムソフトウェアのダウンロード	FC6A 形にシステムソフトウェアをダウンロードします。 FC6A 形マイクロスマート ユーザーズ マニュアル「付録 ダウンロード手順」を参照してください。



メンテナンス通信を使用するには、FC6A 形マイクロスマート ユーザーズ マニュアル「第4章 WindLDR の起動と機種設定」を参照して初期設定を行ってください。

メンテナンス通信で使用する通信ポート

接続する機器に対応するポートをメンテナンス通信に設定することで、メンテナンス通信ができます。

対応する機種と通信ポートは、次のとおりです。

通信ポート	All-in-One CPU モジュール			CAN J1939 All-in-One CPU モジュール	Plus CPU モジュール	
	16 点タイプ	24 点タイプ	40 点タイプ		Plus16 点タイプ	Plus32 点タイプ
USB ポート	○	○	○	○	○	○
ポート 1	○	○	○	—	○*2	○*2
ポート 2	○*1	○*1	○*1	○*1	○*2	○*2
ポート 3	—	—	○*1	○*1	○*3	○*3
ポート 4～33	○ (ポート 4～9) *4				○ (ポート 4～33) *5	
Ethernet ポート 1	○	○	○	○	○	○
Ethernet ポート 2	—	—	—	—	○	○
HMI-Ethernet ポート *6	○	○	○	○	○	○

*1 通信カートリッジを接続した場合

*2 カートリッジベースモジュールを接続し、通信カートリッジを接続した場合

*3 HMI モジュールを接続し、通信カートリッジを接続した場合

*4 All-in-One CPU モジュールおよび CAN J1939 All-in-One CPU モジュールに通信モジュールを最大 3 台接続してポート 4～9 を増設できます。

*5 Plus CPU モジュールに通信モジュールを最大 15 台接続してポート 4～33 を増設できます。

*6 CPU モジュールに HMI モジュールを接続して HMI-Ethernet ポートを増設できます。

各通信ポートで対応しているメンテナンス通信の通信用途は、次のとおりです。

通信用途		USB ポート	ポート 1	ポート 2 および 3	ポート 4～33	Ethernet ポート 1	Ethernet ポート 2	HMI-Ethernet ポート
メンテナンス 通信	システムソフトウェアの ダウンロード	○	○*1	×	×	○	○	○
	ユーザープログラムの ダウンロード/アップロード	○	○*1	×	×	○	○	○
	RUN 中ダウンロード	○	○*1	×	×	○	○	○
	デバイスのモニタ・変更	○	○	○	○	○	○	○

*1 通信カートリッジの場合は対応できません。

各通信ポートで対応しているメンテナンス通信の設定は、次のとおりです。

All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュール

通信ポート	標準 / 増設	通信設定
USBポート	標準	なし
ポート1	標準*1	ファンクション設定「ポート1でのメンテナンス通信」(4-4頁) 参照
ポート 2	増設*2	RS232C/RS485通信カートリッジを使用する場合： ファンクション設定「ポート2～33でのメンテナンス通信」(4-12頁) 参照
ポート 3		Bluetooth通信カートリッジを使用する場合： ファンクション設定「ポート1～3 (Bluetooth) でのメンテナンス通信」(4-15頁) 参照
ポート 4～9	増設*3	ファンクション設定「ポート2～33でのメンテナンス通信」(4-12頁) 参照
Ethernetポート1	標準	ファンクション設定「Ethernetポート1および2でのメンテナンス通信」(4-7頁) 参照
Ethernetポート2	—	—
HMI-Ethernetポート	増設*4	ファンクション設定「HMI-Ethernetポートでのメンテナンス通信」(4-16頁) 参照

*1 CAN J1939 All-in-One CPU モジュールにポート 1 はありません。

*2 CPU モジュールに通信カートリッジを接続してポートを増設できます。

*3 通信モジュールを使用してポートを増設できます。

*4 HMI モジュールを使用してポートを増設できます。

Plus CPU モジュール

通信ポート	標準 / 増設	通信設定
USBポート	標準	なし
ポート1	増設*1	RS232C/RS485通信カートリッジを使用する場合： ファンクション設定「ポート1でのメンテナンス通信」(4-4頁) 参照
ポート 2		Bluetooth通信カートリッジを使用する場合： ファンクション設定「ポート1 (Bluetooth) でのメンテナンス通信」(4-6頁) 参照
ポート 3	増設*2	RS232C/RS485通信カートリッジを使用する場合： ファンクション設定「ポート2～33でのメンテナンス通信」(4-12頁) 参照
ポート 4～33	増設*3	Bluetooth通信カートリッジを使用する場合： ファンクション設定「ポート1～3 (Bluetooth) でのメンテナンス通信」(4-15頁) 参照
Ethernetポート1	標準	ファンクション設定「ポート2～33でのメンテナンス通信」(4-12頁) 参照
Ethernetポート2		ファンクション設定「Ethernetポート1および2でのメンテナンス通信」(4-7頁) 参照
HMI-Ethernetポート	増設*4	ファンクション設定「HMI-Ethernetポートでのメンテナンス通信」(4-16頁) 参照

*1 カートリッジベースモジュールに通信カートリッジを接続してポートを増設できます。

*2 HMI モジュールに通信カートリッジを接続してポートを増設できます。

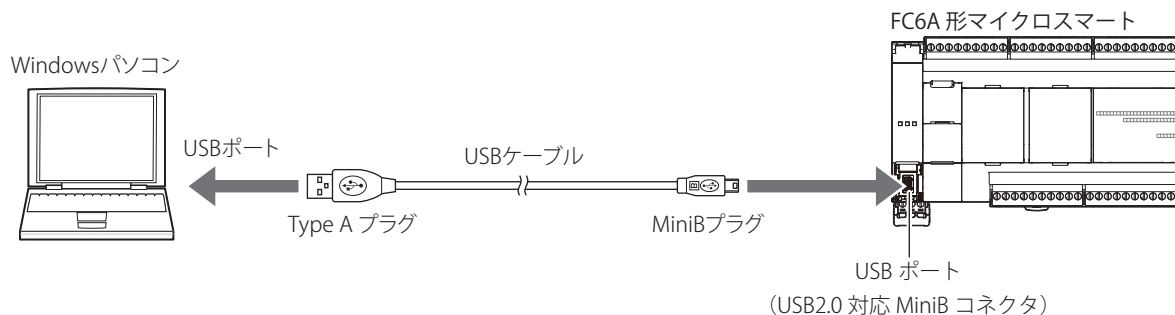
*3 通信モジュールを使用してポートを増設できます。

*4 HMI モジュールを使用してポートを増設できます。

USB ポートでのメンテナンス通信

FC6A 形の USB ポートを使用して WindLDR がインストールされたパソコンと接続し、デバイス値のモニタや変更、ユーザープログラムのダウンロードやアップロード、システムソフトウェアのダウンロードができます。パソコンと FC6A 形を USB ケーブル (推奨ケーブル：HG9Z-XCM42) で接続します。

FC6A 形の USB ポート



メンテナンス通信仕様

項目	仕様・機能
ケーブル	推奨ケーブル：USBメンテナンスケーブル (HG9Z-XCM42)
メンテナンス通信機能	デバイス値のモニタ・変更 ユーザープログラムのダウンロード/アップロード システムソフトウェアのダウンロード RUN中ダウンロード

メンテナンス通信の各機能の操作の詳細

FC6A 形マイクロスマート ユーザーズ マニュアルの以下を参照してください。

- ・デバイス値のモニタ・変更：「第4章 動作確認」
- ・ユーザープログラムのダウンロード/アップロード：「第4章 ユーザープログラムのダウンロード」
- ・システムソフトウェアのダウンロード：「付録 ダウンロード手順」
- ・RUN 中ダウンロード：「第5章 オンラインエディット」

ポート 1 でのメンテナンス通信

FC6A 形を WindLDR がインストールされたパソコンや IDEC 製プログラマブル表示器などと接続し、デバイス値のモニタや変更、ユーザープログラムのダウンロードやアップロード、システムソフトウェアのダウンロードができます。

All-in-One CPU モジュールのシリアルポート 1 を使用する、または Plus CPU モジュールに増設したカートリッジベースモジュールのカートリッジスロット 1 に接続した通信カートリッジを使用することで、ポート 1 でメンテナンス通信ができます。通信ケーブルの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「付録 各種ケーブル」を参照してください。

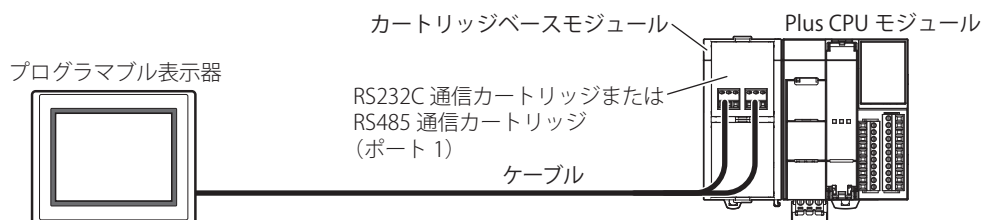
All-in-One CPU モジュールのポート 1



メンテナンス通信仕様

項目	仕様・機能
ケーブル	表示器接続用ケーブル (FC6A-KC2C)
メンテナンス通信機能	デバイス値のモニタ・変更 ユーザープログラムのダウンロード/アップロード システムソフトウェアのダウンロード RUN中ダウンロード

Plus CPU モジュールのポート 1



メンテナンス通信仕様

項目	仕様・機能
ケーブル	RS232C通信カートリッジ：シールド付き多芯ケーブル RS485通信カートリッジ：シールド付きツイストペアケーブル
メンテナンス通信機能	デバイス値のモニタ・変更



RS232C 通信カートリッジおよび RS485 通信カートリッジを使用したメンテナンス通信は、下記の機能に対応していません。

- ・ユーザープログラムのダウンロード/アップロード
- ・システムソフトウェアのダウンロード
- ・RUN 中ダウンロード

メンテナンス通信の各機能の操作の詳細

FC6A 形マイクロスマート ユーザーズ マニュアルの以下を参照してください。

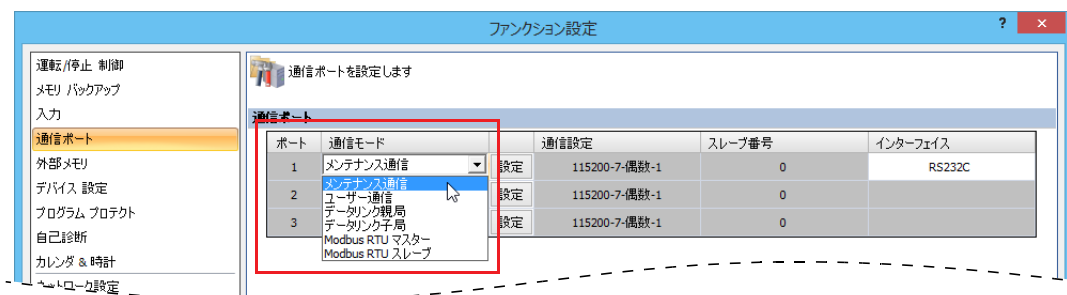
- ・デバイス値のモニタ・変更：「第 4 章 動作確認」

WindLDR の設定

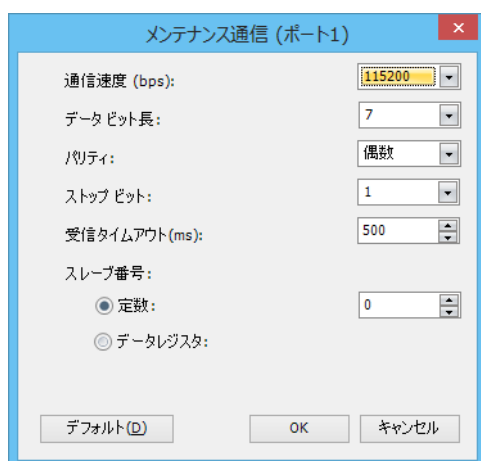
メンテナンス通信をするための各種の設定をします。

●操作手順

1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. ポート 1 の [通信モード] をクリックし、“メンテナンス通信” を選択します。
[メンテナンス通信 (ポート 1)] ダイアログボックスが表示されます。



3. パソコンやプログラマブル表示器の通信フォーマットに合わせて、各パラメータを設定します。



- 通信速度 (bps) : 115200bps
(1200bps / 2400bps / 4800bps / 9600bps / 19200bps / 38400bps / 57600bps / 115200bps)
- データビット長 : 7bit (7bit/8bit)
- パリティ : 偶数 (なし / 偶数 / 奇数)
- ストップビット : 1bit (1bit/2bit)
- 受信タイムアウト (ms) : 500ms (10ms ~ 2550ms)
- スレープ番号 : 0 (0 ~ 31)
スレープ番号の設定方法には定数とデータレジスタがあります。

設定方法	詳細
定数	0 ~ 31の範囲で設定
データレジスタ	特殊データレジスタの値を0~31の範囲で設定 ポート1: D8100 ポート2: D8102 ポート3: D8103 ポート4~9*1: D8040~D8045 ポート10~33*1*2: D8735~D8758

*1 通信モジュール使用時

*2 Plus CPU モジュールのみ

* () 外の値はデフォルト値です。

* ポート 1 でシステムソフトウェアのダウンロードをする場合、データビット長は 8bit に設定してください。

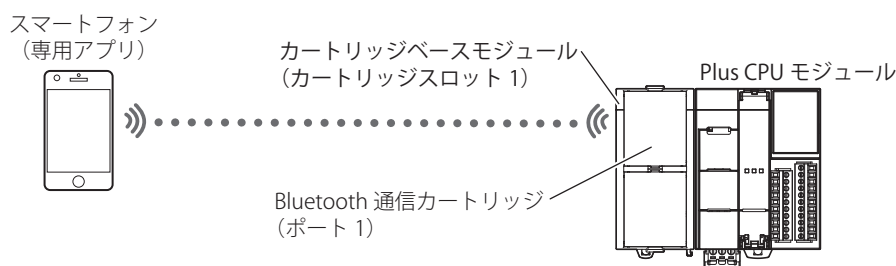
4. [OK] ボタンをクリックします。

以上で、メンテナンス通信の設定が完了します。

ポート 1 (Bluetooth) でのメンテナンス通信

Plus CPU モジュールに増設したカートリッジベースモジュールのカートリッジスロット 1 に Bluetooth 通信カートリッジを接続することで、Bluetooth を備えたパソコンやスマートフォンと接続し、デバイス値のモニタや変更ができます。

Plus CPU モジュールの Bluetooth



メンテナンス通信仕様

項目	仕様・機能
メンテナンス通信機能	デバイス値のモニタ・変更 ユーザープログラムのダウンロード/アップロード



- Bluetooth 通信カートリッジを使用したメンテナンス通信は、下記の機能に対応していません。
 - システムソフトウェアのダウンロード
 - RUN 中ダウンロード
- Bluetooth 通信カートリッジ設定を変更するようなユーザープログラムのダウンロードを行うと、ダウンロード終了時に Bluetooth 通信できなくなりダウンロードエラーとなる可能性があります。

メンテナンス通信の各機能の操作の詳細

FC6A 形マイクロスマート ユーザーズ マニュアルの以下を参照してください。

- デバイス値のモニタ・変更：「第 4 章 動作確認」

WindLDR の設定

詳細は、「第 9 章 Bluetooth 通信」(9-1 頁)を参照してください。

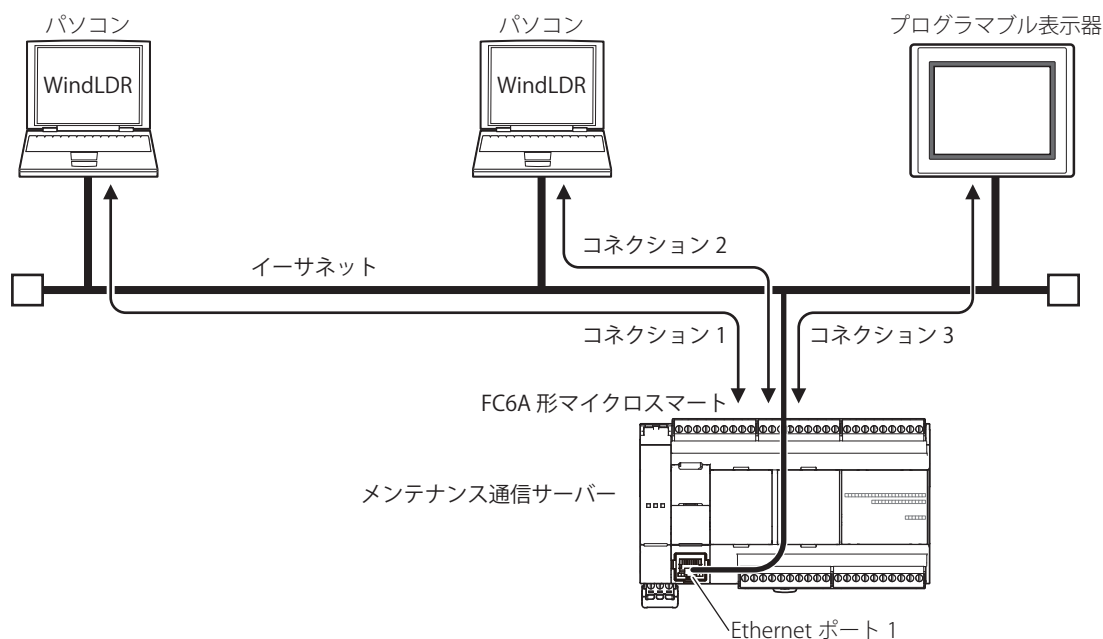
Ethernet ポート 1 および 2 でのメンテナンス通信

All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールの Ethernet ポート 1、Plus CPU モジュールの Ethernet ポート 1 および 2 を使用して、パソコンやプログラマブル表示器などのネットワーク対応機器と通信できます。

ネットワークに接続された外部機器から FC6A 形のデバイス値のモニタや変更、ユーザープログラムのダウンロードやアップロードができます。

All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールは最大 8 個、Plus CPU モジュールは最大 16 個のコネクションにメンテナンス通信サーバーや Modbus TCP 通信など、それぞれ別の通信機能を割り当てることができ、メンテナンス通信サーバー機能と他の通信機能を同時に使用できます。

FC6A 形の Ethernet ポート 1 および 2



メンテナンス通信仕様

項目	仕様・機能
ケーブル	イーサネットケーブル (Cat 5. STP)
メンテナンス通信機能	デバイス値のモニタ・変更 ユーザープログラムのダウンロード/アップロード システムソフトウェアのダウンロード RUN中ダウンロード

メンテナンス通信の各機能の操作の詳細

FC6A 形マイクロスマート ユーザーズ マニュアルの以下を参照してください。

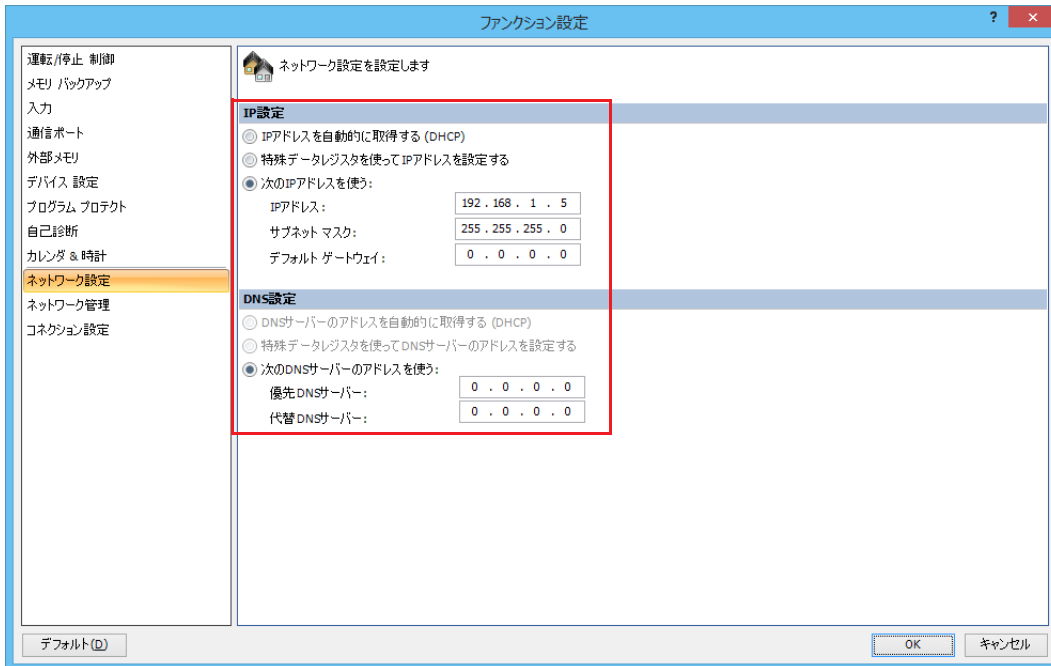
- ・デバイス値のモニタ・変更：「第 4 章 動作確認」
- ・ユーザープログラムのダウンロード/アップロード：「第 4 章 ユーザープログラムのダウンロード」
- ・システムソフトウェアのダウンロード：「付録 ダウンロード手順」
- ・RUN 中ダウンロード：「第 5 章 オンラインエディット」

WindLDR の設定

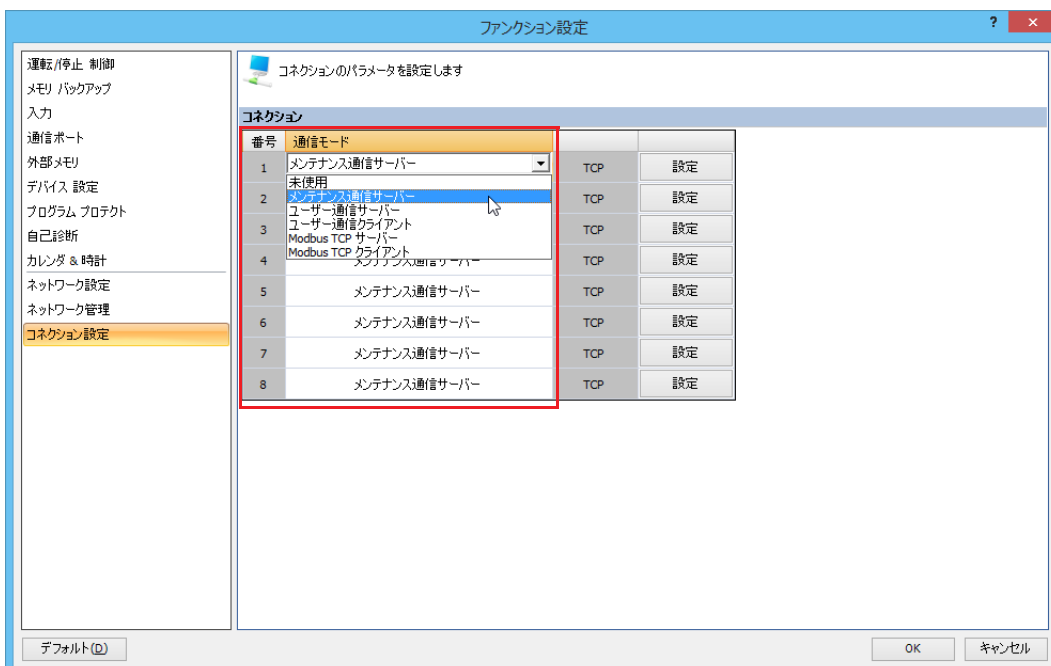
Ethernet ポート 1 および 2 でメンテナンス通信をするための各種の設定およびイーサネット上で FC6A 形と通信するための手順を説明します。

●メンテナンス通信の操作手順

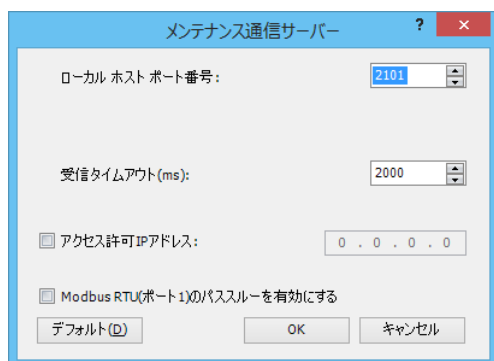
1. [設定] タブの [ファンクション設定] で次の項目をクリックします。
 - All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュール
[ネットワーク設定]
 - Plus CPU モジュール
Ethernet ポート 1 を設定する場合は [イーサネットポート 1]
Ethernet ポート 2 を設定する場合は [イーサネットポート 2]
[ファンクション設定] ダイアログボックスが表示されます。
2. IP アドレス、サブネットマスク、デフォルトゲートウェイを設定します。



3. [コネクション設定] ボタンをクリックします。
4. 使用するコネクションの [通信モード] をクリックし、“メンテナンス通信サーバー”を選択します。
[メンテナンス通信サーバー] ダイアログボックスが表示されます。



5. パソコンやプログラマブル表示器の通信フォーマットに合わせて、各パラメータを設定します。



ローカルホストポート番号：FC6A形がメンテナンス通信サーバーに使用するポートの番号です。
(初期値：2101)

受信タイムアウト (ms)：100ms～25500ms (初期値：2000ms)

アクセス許可 IP アドレス：入力した IP アドレス以外の端末からのアクセスを防ぐことができます。(初期設定：無効)

Modbus RTU (ポート 1) のパススルーを有効にする：

ポート 1 をパススルーポートとして使用できます。この場合、ポート 1 の通信モードを Modbus RTU マスターに設定してください。パススルーポートに使用するポートのローカルホストポート番号は、他のサーバーコネクションのポート番号と異なる番号を指定してください。



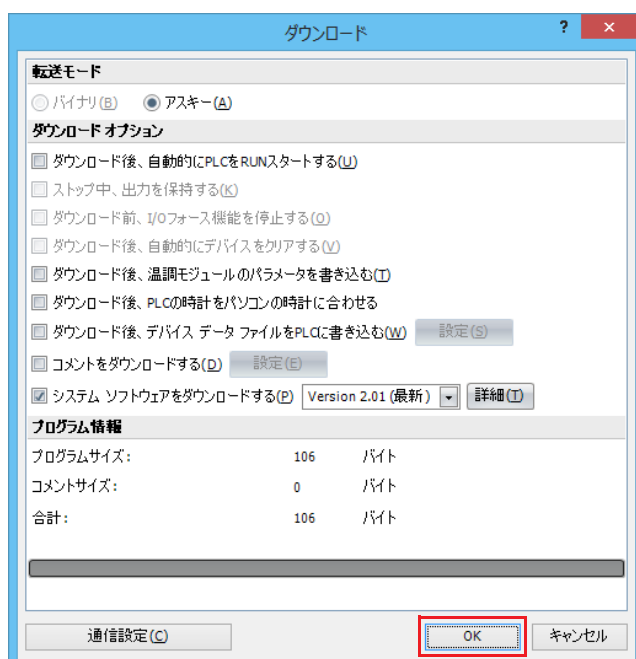
FC6A形に同時に接続可能なクライアントの数は、1 コネクションにつき 1 台です。コネクション 1～8 をすべてメンテナンス通信サーバーに設定した場合、同時に 8 台のクライアントが FC6A 形に接続できます。

6. [OK] ボタンをクリックします。
以上で、メンテナンス通信の設定が完了します。

● USB ポートによるユーザープログラムのダウンロードと IP アドレスの確認

イーサネット通信を開始する前に、ファンクション設定を行ったユーザープログラムを USB ポートを使用して FC6A 形にダウンロードします。

7. パソコンと FC6A 形を USB ケーブルで接続します。
8. [オンライン] タブの [転送] で [ダウンロード] ボタンをクリックします。
[ダウンロード] ダイアログボックスが表示されます。
9. [OK] ボタンをクリックします。
FC6A 形にユーザープログラムがダウンロードされます。



10. ダウンロード成功後、FC6A 形の状態を確認するために、[オンライン] タブの [モニタ] で [モニタ] ボタンをクリックします。
11. [オンライン] タブの [モニタ] で [一括] ボタンをクリックします。
[一括モニタ] ダイアログボックスが表示されます。
12. D8330 ~ D8333 に手順 2 で入力した IP アドレスが正しく表示されていることを確認します。



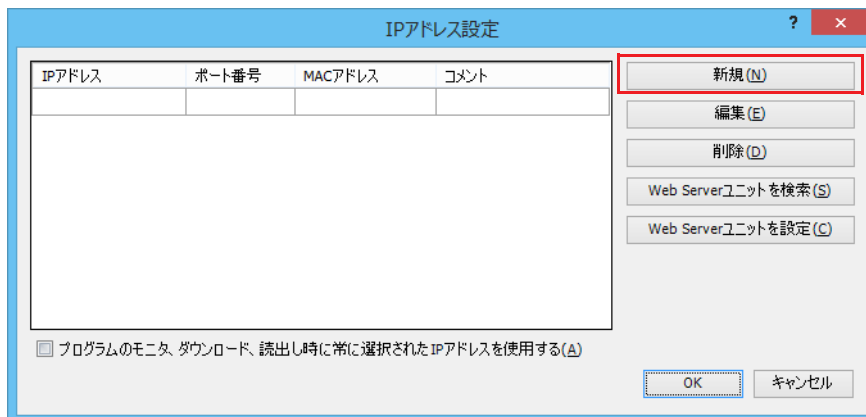
以上で、USB ポートによるユーザープログラムのダウンロードと IP アドレスの確認が完了します。

● WindLDR による Ethernet ポート 1 および 2 での FC6A 形のモニタ手順
WindLDR を使用して、FC6A 形をモニタします。

13. [オンライン] タブの [通信] で [設定] ボタンをクリックします。
[通信設定] ダイアログボックスが表示されます。
14. [イーサネット] タブの [参照] ボタンをクリックします。
[IP アドレス設定] ダイアログボックスが表示されます。



15. [新規] ボタンをクリックします。
[IP アドレス入力] ダイアログボックスが表示されます。



16. 手順2で入力したIPアドレスを入力し、[OK] ボタンをクリックします。

IPアドレス入力

IPアドレス:	192.168.1.5	OK
ポート番号:	2101	キャンセル
MACアドレス:		クリア(C)
コメント:		

17. [オンライン] タブの [モニタ] で [モニタ] ボタンをクリックします。
[IP アドレス設定] ダイアログボックスが表示されます。
18. 入力したIPアドレスが設定されていることを確認して [OK] ボタンをクリックします。

IPアドレス設定

IPアドレス	ポート番号	MACアドレス	コメント
192.168.1.5	2101		

プログラムのモニタ、ダウンロード、読み出し時に常に選択されたIPアドレスを使用する(A)

OK キャンセル

19. [オンライン] タブの [PLC本体] で [ステータス] ボタンをクリックします。
[PLC ステータス] ダイアログボックスが表示されます。
20. FC6A形の機種とシステムバージョンが正しく表示されていることを確認します。

PLCステータス

一般情報 | ネットワーク | コネクション

システム情報

機種: FC6A-C40X1XE

PLCシステムソフトバージョン: 1.00.07

動作ステータス

運転状態: 運転中

スキャンタイム: 現在値: 1 ミリ秒
最大値: 3 ミリ秒

タイマ/カウンタ設定値変更状態: 変更なし

日付: 2015/11/26 19:56:16

書き込みプロテクト: 未設定

読み出しプロテクト: 未設定

エラー状態:

電池電圧: 100 %

OK

以上で、通信の初期設定は完了し、イーサネットでユーザープログラムのダウンロード/アップロードやデバイス値のモニタや変更ができます。

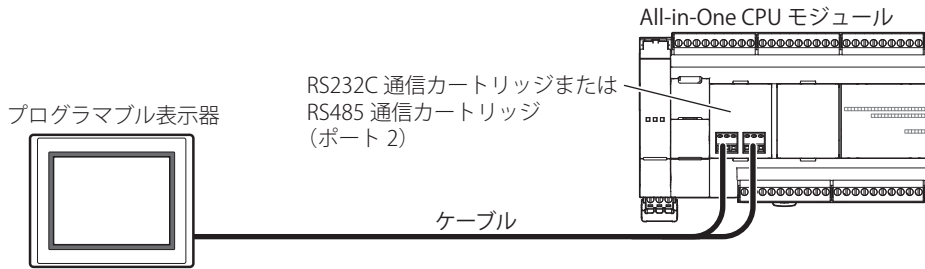
ポート 2 ～ 33 でのメンテナンス通信

FC6A 形に次の拡張をすることで、RS232C や RS485 ポートを備えたパソコンやプログラマブル表示器と接続し、デバイス値のモニタや変更ができます。

- All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールのカートリッジスロットに RS232C 通信カートリッジまたは RS485 通信カートリッジを接続する
- Plus CPU モジュールに増設したカートリッジベースモジュールのカートリッジスロット 2 に RS232C 通信カートリッジまたは RS485 通信カートリッジを接続する
- Plus CPU モジュールに増設した HMI モジュールのカートリッジスロットに RS232C 通信カートリッジまたは RS485 通信カートリッジを接続する
- FC6A 形に通信モジュールを接続する

通信ケーブルの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「付録 各種ケーブル」を参照してください。

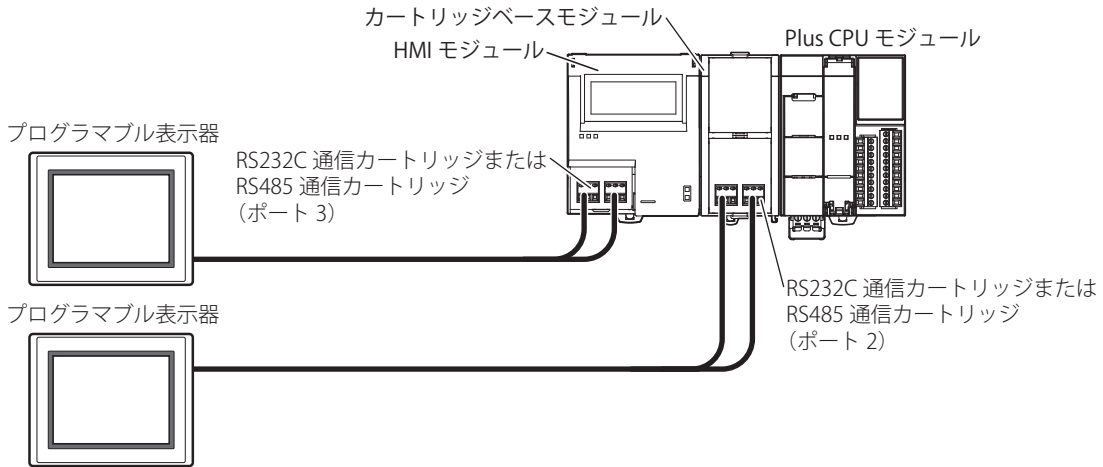
All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールのポート 2 および 3



メンテナンス通信仕様

項目	仕様・機能
ケーブル	RS232C通信カートリッジ：シールド付き多芯ケーブル RS485通信カートリッジ：シールド付きツイストペアケーブル
メンテナンス通信機能	デバイス値のモニタ・変更

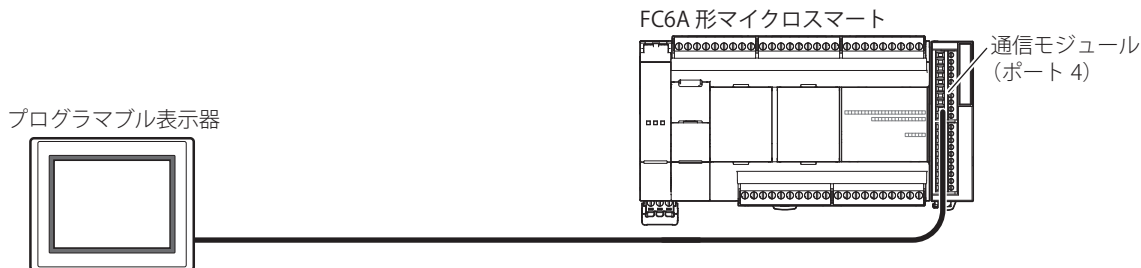
Plus CPU モジュールのポート 2 および 3



メンテナンス通信仕様

項目	仕様・機能
ケーブル	RS232C通信カートリッジ：シールド付き多芯ケーブル RS485通信カートリッジ：シールド付きツイストペアケーブル
メンテナンス通信機能	デバイス値のモニタ・変更

FC6A 形のポート 4 ~ 33



All-in-One CPU モジュールおよび CAN J1939 All-in-One CPU モジュールは通信モジュールを最大 3 台接続してポート 4 ~ 9 を増設できます。

Plus CPU モジュールは通信モジュールを最大 15 台接続してポート 4 ~ 33 を増設できます。通信モジュールの詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 2 章 通信モジュール」を参照してください。

メンテナンス通信仕様

項目	仕様・機能
ケーブル	RS232C通信カートリッジ：シールド付き多芯ケーブル RS485通信カートリッジ：シールド付きツイストペアケーブル
メンテナンス通信機能	デバイス値のモニタ・変更

メンテナンス通信の各機能の操作の詳細

FC6A 形マイクロスマート ユーザーズ マニュアルの以下を参照してください。

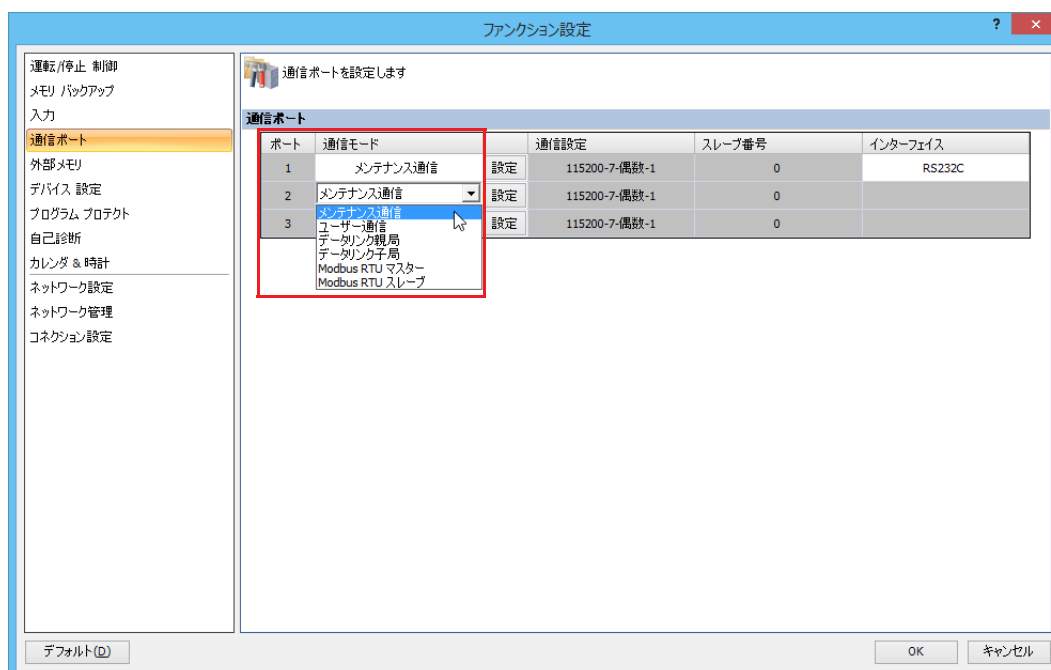
- ・ デバイス値のモニタ・変更：「第 4 章 動作確認」

WindLDR の設定

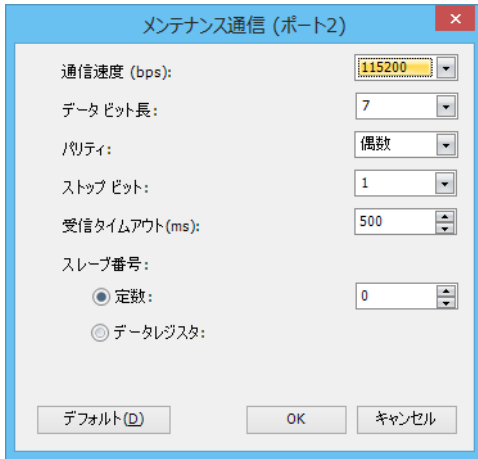
メンテナンス通信をするための各種の設定をします。

●操作手順

1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. ポート 2 ~ 33 の [通信モード] をクリックし、“メンテナンス通信” を選択します。
[メンテナンス通信 (ポート n)] ダイアログボックスが表示されます。(n: 2 ~ 33)



3. パソコンやプログラマブル表示器の通信フォーマットに合わせて、各パラメータを設定します。



- 通信速度 (bps) : 115200bps
(1200bps / 2400bps / 4800bps / 9600bps / 19200bps / 38400bps / 57600bps / 115200bps)
- データビット長 : 7bit (7bit/8bit)
- パリティ : 偶数 (なし / 偶数 / 奇数)
- ストップビット : 1bit (1bit/2bit)
- 受信タイムアウト (ms) : 500ms (10ms ~ 2550ms)
- スレーブ番号 : 0 (0 ~ 31)
スレーブ番号の設定方法には定数とデータレジスタがあります。

設定方法	詳細
定数	0 ~ 31の範囲で設定
データレジスタ	特殊データレジスタの値を0~31の範囲で設定 ポート1: D8100 ポート2: D8102 ポート3: D8103 ポート4~9*1: D8040~D8045 ポート10~33*1*2: D8735~D8758

*1 通信モジュール使用時
*2 Plus CPU モジュールのみ

* () 外の値はデフォルト値です。

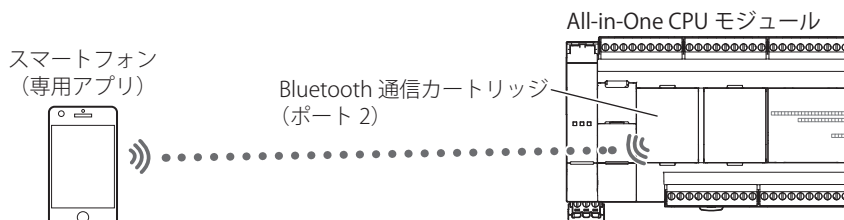
4. [OK] ボタンをクリックします。

以上で、メンテナンス通信の設定が完了します。

ポート 1 ~ 3 (Bluetooth) でのメンテナンス通信

FC6A 形は、Bluetooth を備えたパソコンやスマートフォンと接続し、デバイス値のモニタや変更が行えます。カートリッジスロット 1、2、(Plus CPU モジュールの場合、さらにカートリッジスロット 3) に Bluetooth 通信カートリッジを接続することで、Bluetooth 通信ができます。

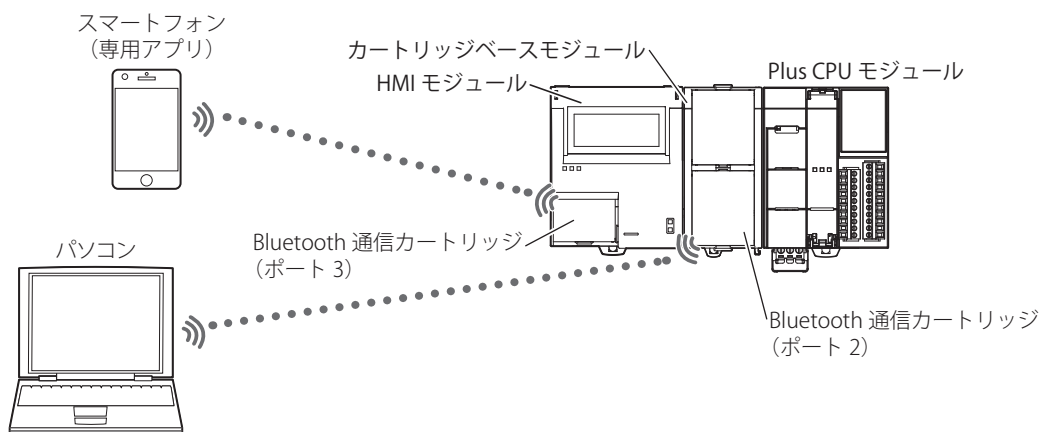
All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールの Bluetooth



メンテナンス通信仕様

項目	仕様・機能
メンテナンス通信機能	デバイス値のモニタ・変更 ユーザープログラムのダウンロード/アップロード

Plus CPU モジュールの Bluetooth



メンテナンス通信仕様

項目	仕様・機能
メンテナンス通信機能	デバイス値のモニタ・変更 ユーザープログラムのダウンロード/アップロード



- Bluetooth 通信カートリッジを使用したメンテナンス通信は、下記の機能に対応していません。
 - システムソフトウェアのダウンロード
 - RUN 中ダウンロード
- Bluetooth 通信カートリッジ設定を変更するようなユーザープログラムのダウンロードを行うと、ダウンロード終了時に Bluetooth 通信できなくなりダウンロードエラーとなる可能性があります。

メンテナンス通信の各機能の操作の詳細

FC6A 形マイクロスマート ユーザーズ マニュアルの以下を参照してください。

- デバイス値のモニタ・変更：「第 4 章 動作確認」

WindLDR の設定

詳細は、「第 9 章 Bluetooth 通信」(9-1 頁) を参照してください。

HMI-Ethernet ポートでのメンテナンス通信

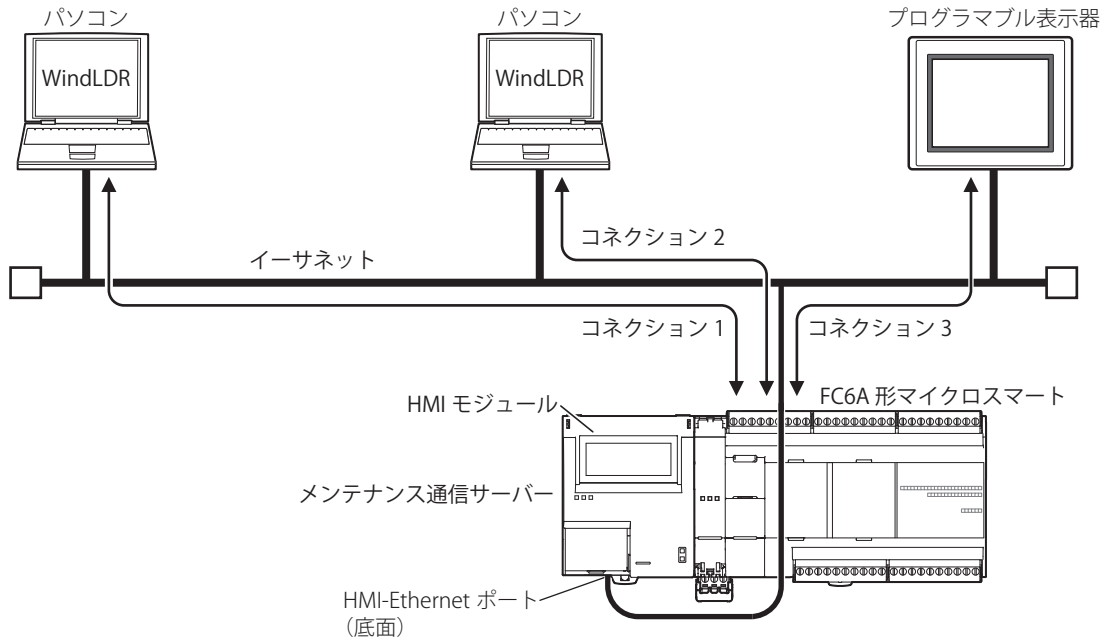
CPU モジュールに接続した HMI モジュールの HMI-Ethernet ポートを使用して、パソコンやプログラブル表示器などのネットワーク対応機器とメンテナンス通信できます。

ネットワークに接続された外部機器から FC6A 形のデバイス値のモニタや変更、ユーザープログラムのダウンロードやアップロードなどが行えます。

FC6A 形は、HMI-Ethernet ポートを使用することにより、All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールに標準装備されている Ethernet ポート 1 が持つコネクション（最大 8 個）、Plus CPU モジュールに標準装備されている Ethernet ポート 1 および 2 が持つコネクション（最大 16 個）に加えて、さらに最大 8 個のコネクションを増設することができます。

HMI-Ethernet ポートにより増設されたコネクション（最大 8 個）に、メンテナンス通信サーバーを割り当てることができます。

HMI モジュールの HMI-Ethernet ポート 1



メンテナンス通信仕様

項目	仕様・機能
ケーブル	イーサネットケーブル (Cat 5.STP)
メンテナンス通信機能	デバイス値のモニタ・変更 ユーザープログラムのダウンロード/アップロード システムソフトウェアのダウンロード RUN中ダウンロード

メンテナンス通信の各機能の操作の詳細

FC6A 形マイクロスマート ユーザーズ マニュアルの以下を参照してください。

- ・ デバイス値のモニタ・変更：「第 4 章 動作確認」
- ・ ユーザープログラムのダウンロード/アップロード：「第 4 章 ユーザープログラムのダウンロード」
- ・ システムソフトウェアのダウンロード：「付録 ダウンロード手順」
- ・ RUN 中ダウンロード：「第 5 章 オンラインエディット」

WindLDR の設定

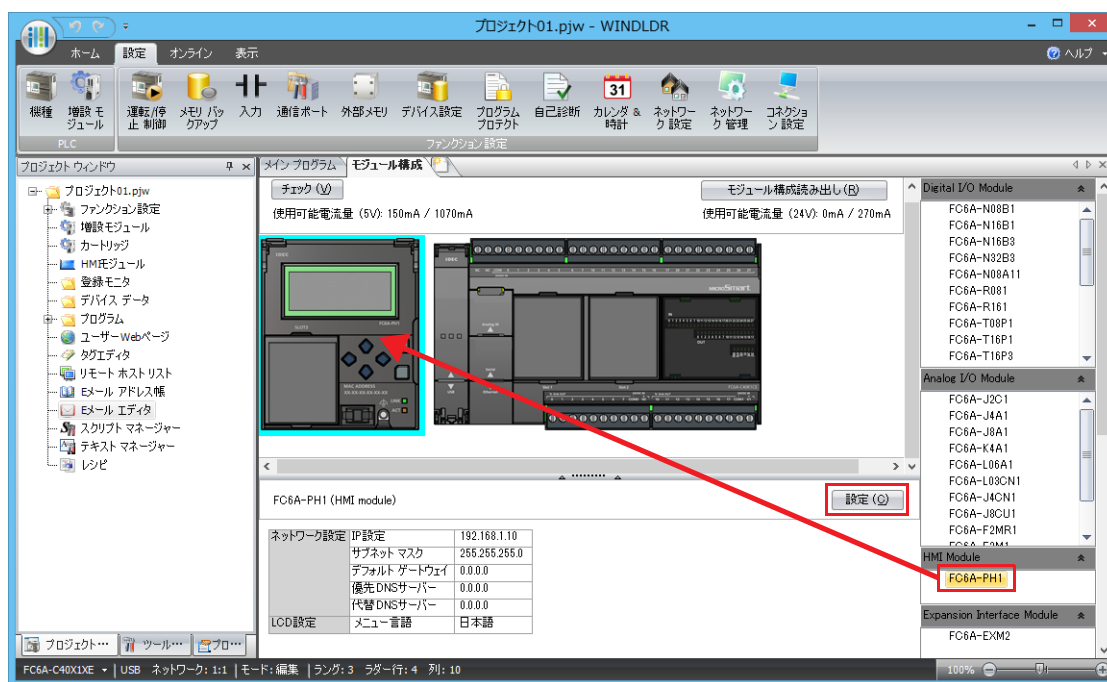
メンテナンス通信をするための各種の設定をします。

●操作手順

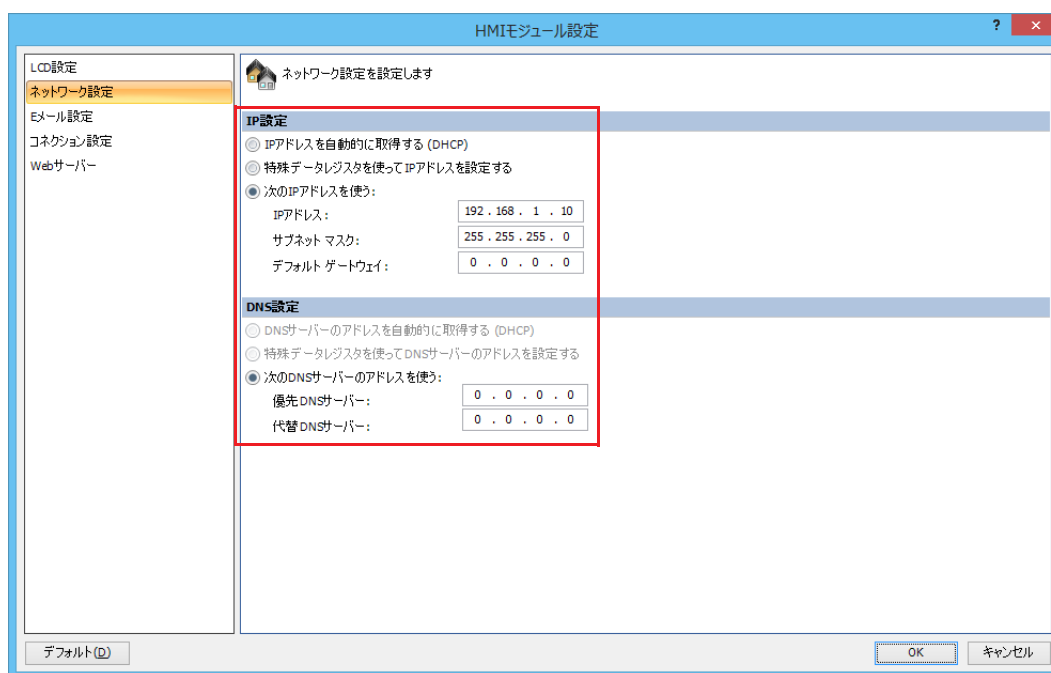
1. HMI モジュールのネットワーク設定およびコネクション設定は、モジュール構成エディタで行います。
[設定] タブの [PLC] で [増設モジュール] をクリックします。
2. モジュール構成エリアに挿入した HMI モジュールをクリックし、[設定] ボタンをクリックします。
[HMI モジュール設定] ダイアログボックスが表示されます。



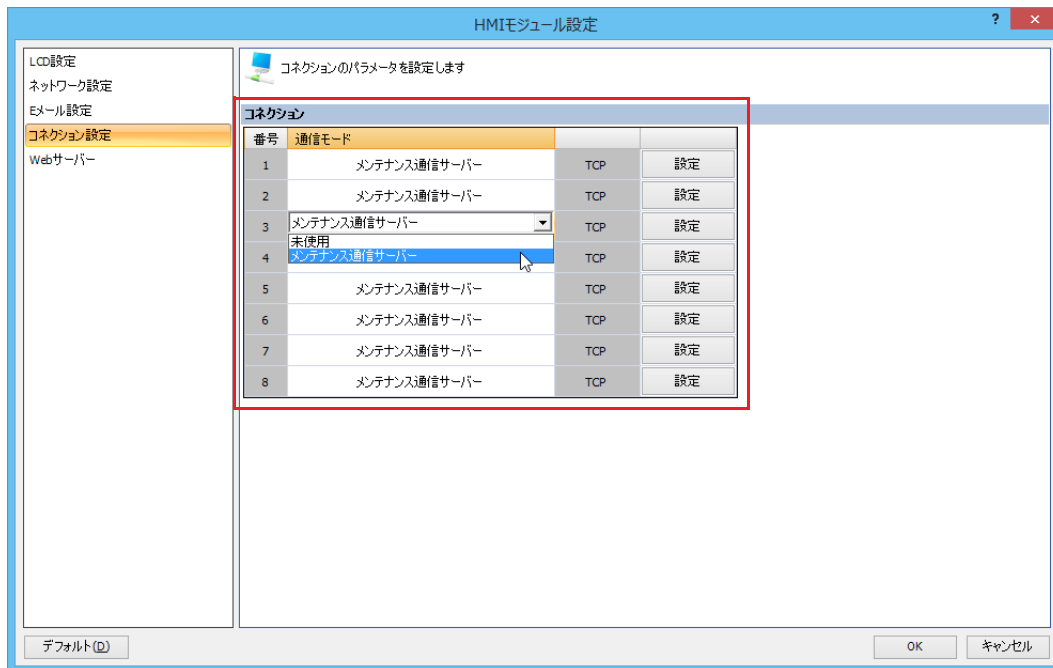
プロジェクトウィンドウの [HMI モジュール] をダブルクリックしても [HMI モジュール設定] ダイアログボックスが表示されます。



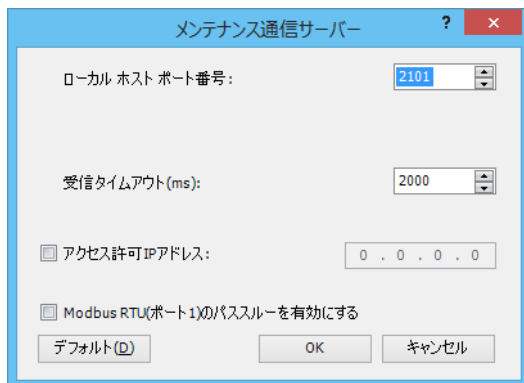
3. [ネットワーク設定] タブをクリックし、[IP 設定]、[DNS 設定] をそれぞれ設定します。



4. [コネクション設定] タブをクリックします。
5. 使用するコネクション番号の [通信モード] をクリックし、通信モードを選択します。



6. パソコンやプログラマブル表示器の通信フォーマットに合わせて、各パラメータを設定します。



ローカルホストポート番号 : FC6A 形がメンテナンス通信サーバーに使用するポートの番号です。
(初期値 : 2101)

受信タイムアウト (ms) : 100ms ~ 25500ms (初期値 : 2000ms)

アクセス許可 IP アドレス : 入力した IP アドレス以外の端末からのアクセスを防ぐことができます。(初期設定 : 無効)



FC6A 形に同時に接続可能なクライアントの数は、1 コネクションにつき 1 台です。コネクション 1 ~ 8 をすべてメンテナンス通信サーバーに設定した場合、同時に 8 台のクライアントが FC6A 形に接続できます。

7. [OK] ボタンをクリックします。

以上で、メンテナンス通信の設定が完了します。

第5章 ユーザー通信

この章では、FC6A形のユーザー通信について説明します。

概要

ユーザー通信は、設定したデータを FC6A 形に接続した外部機器に合ったデータタイプに変換して送受信します。

ユーザー通信の実行

ユーザー通信の実行には、ユーザー通信命令を使用します。

ユーザー通信命令は、使用する通信インターフェイスによって使用するユーザー通信命令が異なります。

- ・ポート 1～33 を使用して外部機器とシリアル通信 (RS232C/RS485) でユーザー通信する場合
ポート 1～3 を使用して外部機器とシリアル通信 (Bluetooth) でユーザー通信する場合
「TXD (ユーザー通信送信)」(5-2 頁)
「RXD (ユーザー通信受信)」(5-9 頁)
- ・Ethernet ポート 1 および 2 を使用して外部機器とイーサネット通信でユーザー通信する場合
「ETXD (イーサネットユーザー通信送信)」(5-22 頁)
「ERXD (イーサネットユーザー通信受信)」(5-22 頁)

ユーザー通信で使用する通信ポート

対応する機種と通信ポートは、次のとおりです。

通信ポート	All-in-One CPU モジュール			CAN J1939 All-in-One CPU モジュール	Plus CPU モジュール	
	16 点タイプ	24 点タイプ	40 点タイプ		Plus16 点タイプ	Plus32 点タイプ
ポート 1	○	○	○	—	○*2	○*2
ポート 2	○*1	○*1	○*1	○*1	○*2	○*2
ポート 3	—	—	○*1	○*1	○*3	○*3
ポート 4～33	○ (ポート 4～9) *4				○ (ポート 4～33) *5	
Ethernet ポート 1	○	○	○	○	○	○
Ethernet ポート 2	—	—	—	—	○	○
HMI-Ethernet ポート	—	—	—	—	—	—

*1 通信カートリッジを接続した場合

*2 カートリッジベースモジュールを接続し、通信カートリッジを接続した場合

*3 HMI モジュールを接続し、通信カートリッジを接続した場合

*4 All-in-One CPU モジュールおよび CAN J1939 All-in-One CPU モジュールに通信モジュールを最大 3 台接続してポート 4～9 を増設できます。

*5 Plus CPU モジュールに通信モジュールを最大 15 台接続してポート 4～33 を増設できます。

ユーザー通信の設定

各通信ポートで対応しているユーザー通信の設定は、次のとおりです。

- ・ポート 1～33 を使用して外部機器とシリアル通信 (RS232C/RS485)
ポート 1～3 を使用して外部機器とシリアル通信 (Bluetooth)
「シリアル通信でのユーザー通信」(5-23 頁)
- ・Ethernet ポート 1 および 2 を使用して外部機器とイーサネット通信
「イーサネット通信でのユーザー通信」(5-39 頁)

TXD (ユーザー通信送信)

ポート 1 ~ 33 を使用して外部機器へ、送信データを設定したデータタイプに変換して送信します。

たとえば、0123h のようなバイナリデータを、アスキーデータ "0"(30h)、"1"(31h)、"2"(32h)、"3"(33h) に変換して送信します。

ラダー図



動作説明

入力が ON すると、S1 で設定した送信データをポート 1 ~ 33 から送信します。

すべてのデータの送信が完了すると、D1 で設定したデバイスを ON し、D2 で設定したデバイスに送信動作のステータス（送信動作の遷移状態とエラーコード）を格納します。

D2+1 には送信したデータのバイト数を格納します。



- ・シリアル通信でのユーザー通信については、「シリアル通信でのユーザー通信」(5-23 頁) を参照してください。
- ・Bluetooth 通信でのユーザー通信については、「シリアル通信 (Bluetooth) でのユーザー通信」(5-38 頁) を参照してください。



- ・FC6A 形は、送信命令を実行するための準備エリアを 5 つ持っています。実際に送信命令を実行する場合には、この準備エリアの情報にしたがって、送信データを内部送信バッファに順次展開して実行します。6 つ以上の送信命令を同時に実行しようとした場合、6 つ目以降の送信命令は実行されません。実行されなかった送信命令は、対応する送信動作ステータスにエラーコード "1" を格納します。
- ・送信命令実行中に別の入力条件が成立して、新しく送信命令を実行する場合には、現在実行中の送信命令が完了してから 2 スキャン後に新しい送信命令を実行します。
- ・送信命令は、入力条件が成立している間、繰り返し送信を実行します。送信を 1 回のみ実行したい場合には、ショット命令を入力条件に追加してください。ショット命令については、ラダープログラミングマニュアル「第 4 章 SOTU (ショットアップ)」、「第 4 章 SOTD (ショットダウン)」を参照してください。
- ・TXD 命令および RXD 命令では、送信 / 受信動作ステータスと送信 / 受信データバイト数を格納するデータレジスタは重複できません。
- ・TXD 命令は割込プログラムでは使用できません。割込プログラムで TXD 命令を使用すると、ユーザープログラム実行エラーコード (D8006) にエラーコード 18 が格納されます。ユーザープログラム実行エラーについては、ラダープログラミングマニュアル「第 3 章 ●ユーザープログラム実行エラー」を参照してください。

対象デバイス

			I	Q	M	R	T	C	D	P	定数	リピート指定
S1	ソース1	送信データ	—	—	—	—	—	—	○	—	○	—
D1	デスティネーション1	送信完了出力	—	○	○*1	—	—	—	—	—	—	—
D2	デスティネーション2	送信動作ステータス	—	—	—	—	—	—	○*2	—	—	—

*1 特殊内部リレーは使用できません。

*2 特殊データレジスタは使用できません。

設定項目

■S1 (ソース 1) の設定

送信するデータを設定します。

送信するデータは、データタイプを組合せて構成します。データタイプには、定数、データレジスタおよび BCC があります。

1 つの送信命令で送信できるデータの送信バイト数の合計は最大 1500 バイトです。

データタイプ	変換タイプ	送信バイト数	リピート回数	計算方法	計算開始位置
定数 (文字)	無変換	1	—	—	—
定数 (16進数)					
間接 (データレジスタ)	バイナリ→アスキー	1~4	1~99	—	—
	バイナリ→BCD→アスキー	1~5			
	無変換	1~2			
BCC	バイナリ→アスキー 無変換	1~2	—	XOR ADD ADD-2の補数 Modbus ASCII Modbus RTU	1~15

* WindLDR のラダー図上では、TXD 命令で送信するデータの総バイト数を S1 に表示します。

定数

1 バイト (00h ~ FFh) のデータを無変換で送信します。

定数データは文字または 16 進数で設定します。

送信できるデータの範囲は、通信設定のデータビット長の設定によって異なります。データビット長が 7 ビットの場合は 00h ~ 7Fh、8 ビット設定の場合は 00h ~ FFh となります。

定数 (文字) : 送信する定数データを文字で設定します。
キーボードで入力できるアスキーデータ (半角) を入力します。
半角 1 文字あたり 1 バイトとして換算します。

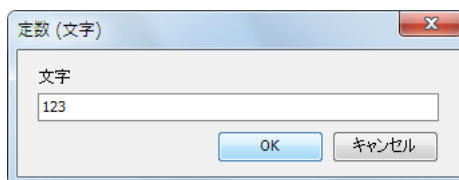
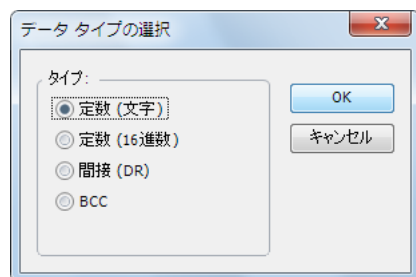
定数 (16 進数) : 送信する定数データを 16 進数で設定します。
アスキーデータなどのデータを 16 進数で入力します。
アスキーコードの制御文字 NUL (00h) ~ US (1Fh) を入力する場合は、必ずこの設定で入力します。

例) 3 バイトのアスキーデータ "1" (31h)、"2" (32h)、"3" (33h) を定数データとして設定する場合

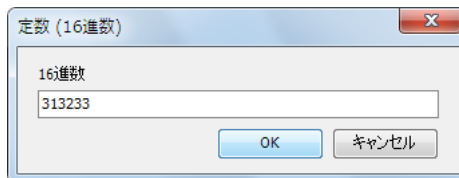
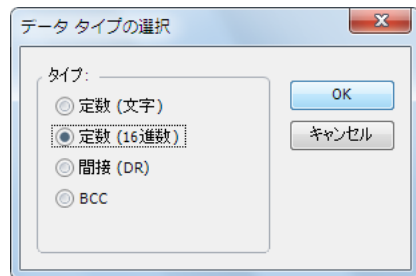
定数 (文字) → 123

定数 (16 進数) → 313233

[定数 (文字) で設定した場合]



[定数 (16 進数) で設定した場合]



間接（データレジスタ）

設定したデータレジスタの内容を変換タイプの設定にしたがってデータ変換し、設定したバイト数だけ送信します。リピート回数を設定することにより、設定したデータレジスタを先頭に設定したリピート回数分のデータレジスタのデータを連続して送信できます。

設定項目	内容
データレジスタ番号	送信するデータをデータレジスタで設定します。
変換タイプ	データレジスタの内容をデータ変換するときの変換方法を設定します。通信相手機器の通信プロトコルに合わせて設定してください。 バイナリ→アスキー変換、バイナリ→BCD→アスキー変換、無変換 の3種類から選択します。
送信バイト数	送信するデータのバイト数を設定します。変換タイプによって、設定できる送信バイト数が異なります。 バイナリ→アスキー変換： 1～4バイト バイナリ→BCD→アスキー変換： 1～5バイト 無変換： 1～2バイト
リピート回数	送信するデータが連続したデータレジスタに格納されている場合に、同一フォーマットのデータを連続して送信できます。 リピート回数は最大99回まで設定できます。

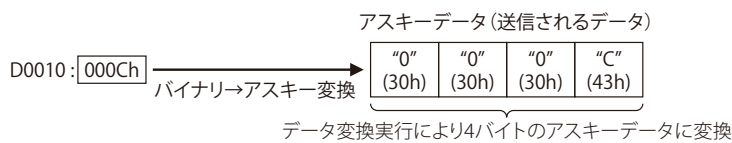
●データ変換後のデータと送信されるデータの動作例

[設定した送信バイト数がデータ変換後のデータのバイト数と同じ場合]

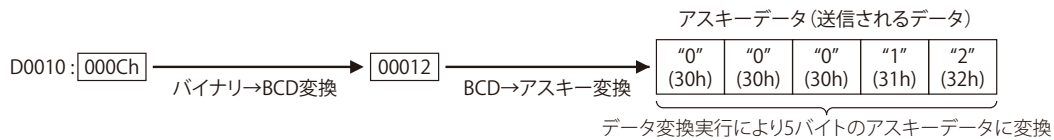
データ変換後のデータをすべて送信します。

[データレジスタ番号が D0010 のデータ "000Ch" (C (16 進数) = 12 (10 進数)) を送信する場合]

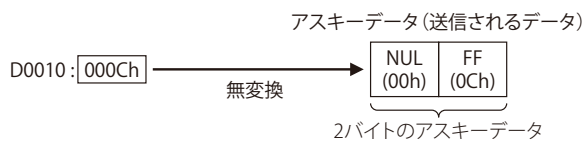
- 変換タイプでバイナリ→アスキー変換、送信バイト数で4バイトを設定した場合
"000Ch" (12) をバイナリ→アスキー変換して4バイト送信すると、アスキーデータ "000C" が送信されます。



- 変換タイプでバイナリ→BCD→アスキー変換、送信バイト数で5バイトを設定した場合
"000Ch" (12) をバイナリ→BCD→アスキー変換して5バイト送信すると、アスキーデータ "00012" が送信されます。



- 変換タイプで無変換、送信バイト数で2バイトを設定した場合
"000Ch" (12) を無変換で2バイト送信すると、バイナリデータ "000Ch" が送信されます。

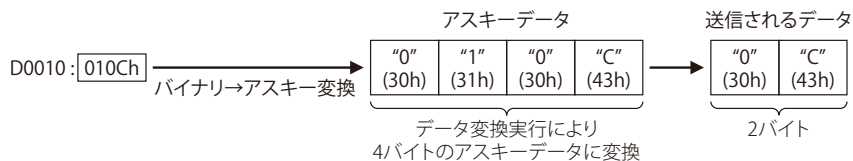


[設定した送信バイト数がデータ変換後のデータのバイト数より小さい場合]

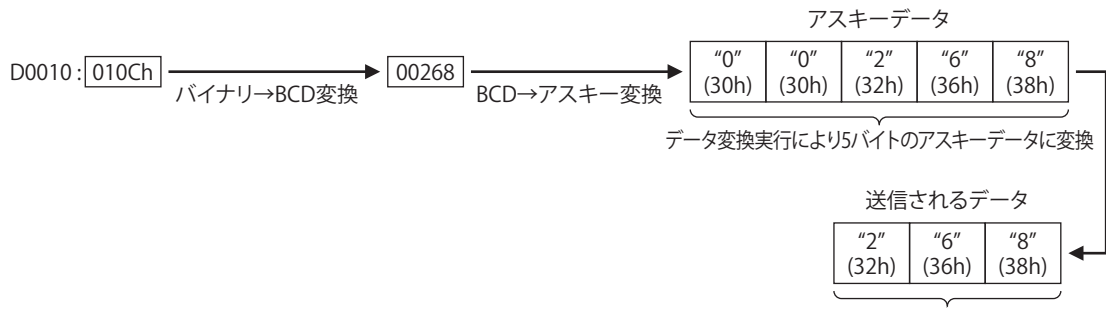
変換後のデータを設定したバイト数分だけ送信します。

[データレジスタ番号が D0010 のデータ "010Ch" (10C (16 進数) = 268 (10 進数)) を送信する場合]

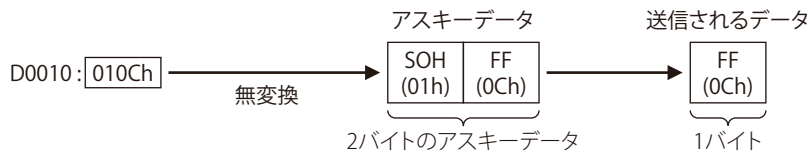
- 変換タイプでバイナリ→アスキー変換、送信バイト数で2バイトを設定した場合
"010Ch" (268) をバイナリ→アスキー変換して2バイト送信すると、アスキーデータ "0C" が送信されます。



- 変換タイプでバイナリ→BCD→アスキー変換、送信バイト数で3バイトを設定した場合
 “010Ch” (268) をバイナリ→BCD→アスキー変換して3バイト送信すると、アスキーデータ “268” が送信されます。



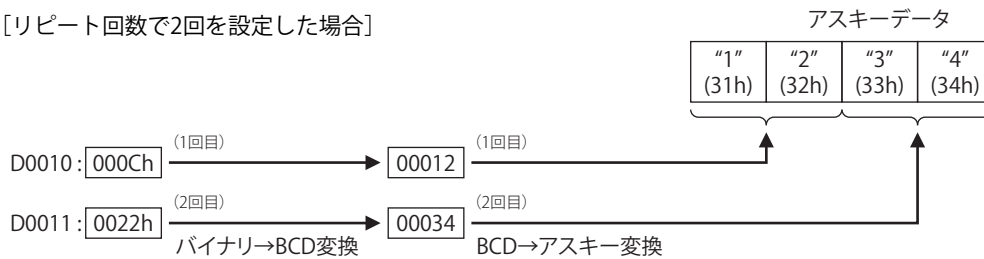
- 変換タイプで無変換、送信バイト数で1バイトを設定した場合
 “010Ch” (268) を無変換で1バイト送信すると、バイナリデータ “0Ch” が送信されます。



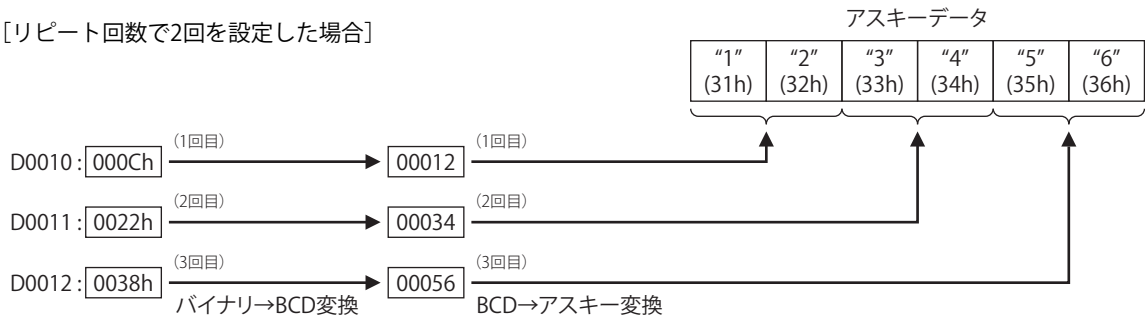
●リピート回数の設定と動作例

データレジスタ番号が D0010、変換タイプがバイナリ→BCD→アスキー変換、送信バイト数が2バイトで、D0010 のデータ “000Ch”、D0011 のデータ “0022h”、D0012 のデータ “0038h” の場合、リピート回数に応じて次のデータが送信されます。

[リピート回数で2回を設定した場合]



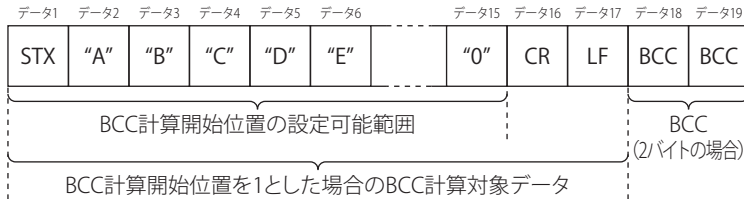
[リピート回数で2回を設定した場合]



BCC (Block Check Character)

送信データの BCC を自動的に算出し、送信データの任意の位置に BCC データを付加できます。各設定項目は通信相手機器の通信プロトコルに合わせて設定してください。

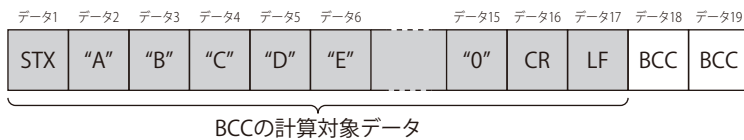
設定項目	設定内容
計算開始位置	BCCの計算を開始する位置を設定します。 計算開始位置は、送信データの先頭から15バイト以内で設定します。計算終了位置は、BCCの直前になります。
計算方法	BCCの計算方法を設定します。 排他的論理和(XOR)、加算(ADD)、ADD-2の補数、Modbus ASCII、Modbus RTUの5種類から選択します。
変換タイプ	データ変換するときの変換方法を設定します。 バイナリ→アスキー変換と無変換の2種類から選択します。
送信バイト数	送信データに付加するBCCコードのバイト数を1~2バイトで設定します。



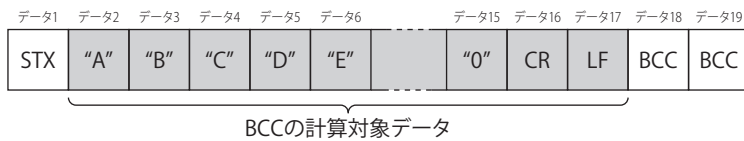
●計算開始位置の設定と動作例

設定した計算開始位置による BCC の計算対象データは次のとおりです。

[計算開始位置で1バイト目を設定した場合]



[計算開始位置で2バイト目を設定した場合]



●計算方法の設定と動作例

BCC の計算対象データが次の場合

"A"	"B"	"C"	"D"
(41h)	(42h)	(43h)	(44h)

計算結果は次のとおりです。

[計算方法で排他的論理和 (XOR) を設定した場合]

BCC 結果 = 41h ∨ 42h ∨ 43h ∨ 44h = 04h

[計算方法で加算 (ADD) を設定した場合]

BCC 結果 = 41h + 42h + 43h + 44h = 010Ah

[計算方法で ADD-2 の補数を設定した場合]

BCC 結果 = 41h + 42h + 43h + 44h = 010Ah → FEF6h

↑
2の補数に変換

[計算方法で Modbus ASCII を設定した場合]

BCC (LRC) 結果 = "8", "8" (アスキー)

[計算方法で Modbus RTU を設定した場合]

BCC (CRC-16) 結果 = 85h, 0Fh (バイナリ)

**ADD-2 の補数**

1. 計算開始位置から BCC 格納位置手前までのデータの和を計算します。
2. 1.の結果をビット反転し 1 を足します。(2 の補数)
3. 2.結果を変換タイプ (BIN → ASCII, 無変換) と桁指定 (1 桁、2 桁) によって BCC 格納位置に格納します。

Modbus ASCII

1. 計算開始位置から BCC (LRC) 格納位置手前までの ASCII 文字を 2 文字単位で 1 バイト HEX データに変換 (例: 37h, 35h → 75h) します。
2. 1.の結果得られたデータの和を計算します。
3. 2.の結果をビット反転し 1 を足します。(2 の補数)
4. 3.の結果の下位 1 バイトのデータを ASCII 文字に変換 (例: 75h → 37h, 35h) し、BCC (LRC) 格納位置に格納します。BCC の計算対象データが奇数バイト設定されている場合、BCC は不定となります。(Modbus プロトコル定義外)

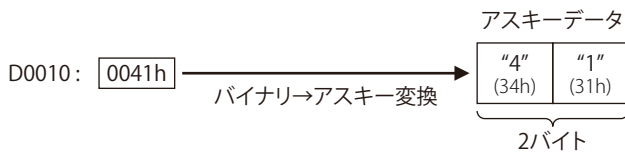
Modbus RTU

1. 計算開始位置の 1 バイトのデータと FFFFh の排他的論理和 (XOR) を取ります。
2. 1.の結果を右に 1 ビットシフトし、キャリーが出れば、固定値 (A001h) で XOR を取ります。
3. 8 回シフトするまで②を繰り返します。
4. 次の 1 バイトデータと③の結果の排他的論理和 (XOR) を取ります。
5. BCC (CRC) 格納位置手前のデータまで 2. ~ 4. を繰り返します。
6. 5.の結果を CRC-16 として BCC (CRC) 格納位置に下位上位 (例: 1234h → 34h, 12h) の順で格納します。

●変換タイプの設定と動作例

BCC の計算結果が "0041h" の場合、データ変換後のデータは次のとおりです。

[変換タイプでバイナリ→アスキー変換を設定した場合]



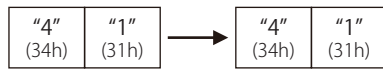
[変換タイプで無変換を設定した場合]



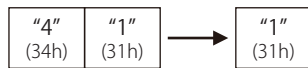
●バイト数設定の設定と動作例

BCC の計算結果が "0041h" の場合、送信データに付加するデータは次のとおりです。

[送信バイト数で2バイトを設定した場合]



[送信バイト数で1バイトを設定した場合]



■D1 (デスティネーション 1) の設定

送信完了出力として内部リレーまたは出力を設定します。

TXD 命令の起動入力 ON して、送信前処理→送信処理の一連の処理を終えた時点で、送信完了出力は ON します。

■D2 (デスティネーション 2) の設定

送信動作ステータスおよび送信バイト数を格納するデータレジスタを設定します。

送信動作ステータスは D2+0 に、送信バイト数は D2+1 に格納します。



送信動作ステータスとして D0100 を設定した場合、D0100 を先頭に連続した 2 個のデータレジスタを使用します。

D0100: ← 送信動作ステータス

D0101: ← 送信バイト数

D2+0 (送信動作ステータス)

D2 で設定したデータレジスタには送信動作ステータスが格納されます。

ステータスコード*1	送信状態	状態説明
16	送信前処理中	TXD命令の起動入力ONしてから、設定された送信データを展開し内部バッファに格納するまでの間
32	送信中	送信命令が実行可能になり、すべてのデータが送信されるまでの間
48	送信データ完了	データを送信したあと、送信命令で終了処理が実行されるまでの間
64	送信命令完了	一連の通信処理がすべて終了した状態。この状態になると送信命令の実行が可能になり、送信命令が実行されると送信前処理中 (ステータスコード16) の状態に移ります。

*1 ステータスコードが上記以外の場合には、送信命令の実行でエラーが発生しています。詳細は、「ユーザー通信送信命令・受信命令のエラー」(5-53 頁) を参照してください。

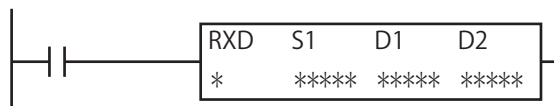
D2+1 (送信バイト数)

D2 で設定したデータレジスタの次のデータレジスタ (D2+1) には送信バイト数が格納されます。送信バイト数には BCC データも含まれます。

RXD (ユーザー通信受信)

ポート 1 ~ 33 を使用して外部機器からデータを受信し、適切なデータタイプに変換してデータレジスタに格納します。
たとえば、'0'(30h)、'1'(31h)、'2'(32h)、'3'(33h) のようなアスキーデータを、バイナリデータ 0123h に変換し、データレジスタに格納します。

ラダー図



動作説明

入力が ON すると、S1 で設定したフォーマットの受信データをポート 1 ~ 33 から受信します。
すべてのデータの受信が完了すると、D1 で設定したデバイスを ON し、D2 で設定したデバイスに受信動作のステータス（受信動作の遷移状態とエラーコード）を格納します。
D2+1 には実際に受信したデータのバイト数を格納します。
受信データ待ちの状態ユーザー通信受信命令キャンセルフラグ（M8022、M8023、M8026、M8033、M8145 ~ M8147、M8170、M8176、M8365 ~ M8394）を ON すると、対応する通信ポートに対して実行中のすべての受信命令の実行をキャンセルします。



- シリアル通信でのユーザー通信については、「シリアル通信でのユーザー通信」(5-23 頁) を参照してください。
- Bluetooth 通信でのユーザー通信については、「シリアル通信 (Bluetooth) でのユーザー通信」(5-38 頁) を参照してください。



- FC6A 形の受信命令は、スタートデリミタを設定することで、最大 5 つの受信命令を同時に実行できます。スタートデリミタを設定しない場合は 1 命令のみ実行できます。
- 受信命令は、入力条件が成立している間、繰り返し受信を実行します。受信を 1 回のみ実行したい場合には、ショット命令を入力条件に追加してください。ショット命令については、ラダープログラミングマニュアル「第 4 章 SOTU (ショットアップ)」、「第 4 章 SOTD (ショットダウン)」を参照してください。
- TXD 命令および RXD 命令では、送信 / 受信動作ステータスと送信 / 受信データバイト数を格納するデータレジスタは重複できません。
- RXD 命令は割込プログラムでは使用できません。割込プログラムで RXD 命令を使用すると、ユーザープログラム実行エラーコード (D8006) にエラーコード 18 が格納されます。ユーザープログラム実行エラーについては、ラダープログラミングマニュアル「第 3 章 ●ユーザープログラム実行エラー」を参照してください。

対象デバイス

			I	Q	M	R	T	C	D	P	定数	リピート指定
S1	ソース1	受信データ	—	—	—	—	—	—	○	—	○	—
D1	デスティネーション1	受信完了出力	—	○	○*1	—	—	—	—	—	—	—
D2	デスティネーション2	受信動作ステータス	—	—	—	—	—	—	○*2	—	—	—

*1 特殊内部リレーは使用できません。

*2 特殊データレジスタは使用できません。

設定項目

■S1 (ソース 1) の設定

受信するデータを設定します。

受信データは、データタイプを組合せて構成します。データタイプには、定数、データレジスタ、BCC およびスキップがあります。1つの受信命令で受信できるデータの受信バイト数の合計は最大 1500 バイトです。

データタイプ	変換タイプ	受信バイト数	リピート回数	計算方法	計算開始位置	スキップ
定数 (文字)	無変換	1	—	—	—	—
定数 (16進数)						
間接 (データレジスタ)	アスキー→バイナリ	1~4	1~99	—	—	—
	アスキー→BCD→バイナリ	1~5				
	無変換	1~2				
BCC	バイナリ→アスキー	1~2	—	XOR ADD ADD-2の補数 Modbus ASCII Modbus RTU	1~15	—
	無変換					
スキップ	—	—	—	—	—	1~99

* WindLDR のラダー図上では、RXD 命令で受信するデータの総バイト数を S1 に表示します。

定数

1 バイト (00h ~ FFh) のデータを無変換で受信します。

定数データは文字または 16 進数で設定します。

受信できるデータの範囲は、通信設定のデータビット長の設定によって異なります。データビット長が 7 ビットの場合は 00h ~ 7Fh、8 ビット設定の場合は 00h ~ FFh となります。

受信データの先頭に定数を設定した場合は、定数データをスタートデリミタとして扱います。詳細は、「●スタートデリミタの設定」(5-17 頁) を参照してください。

受信データの最終または BCC の直前に定数を設定した場合は、定数データをエンドデリミタとして扱います。詳細は、「●エンドデリミタの設定」(5-19 頁) を参照してください。

スタートデリミタとエンドデリミタ以外の用途で定数を設定した場合、受信データの照合用定数として扱います。詳細は、「●定数設定によるデータ照合」(5-20 頁) を参照してください。

定数 (文字) : 受信する定数データを文字で設定します。
キーボードで入力できるアスキーデータ (半角) を入力します。
半角 1 文字あたり 1 バイトとして換算します。

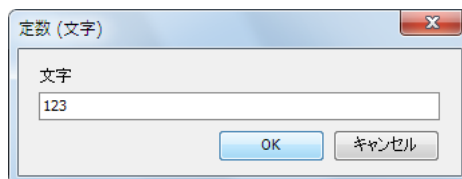
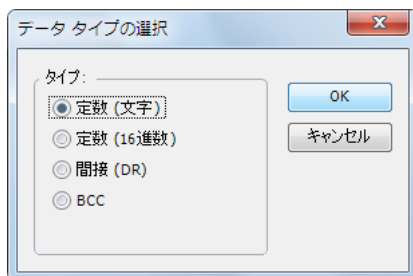
定数 (16 進数) : 受信する定数データを 16 進数で設定します。
アスキーデータなどのデータを 16 進数で入力します。
アスキーコードの制御文字 NUL (00h) ~ US (1Fh) を入力する場合は、必ずこの設定で入力します。

例) 3 バイトのアスキーデータ "1" (31h)、"2" (32h)、"3" (33h) を定数データとして設定する場合

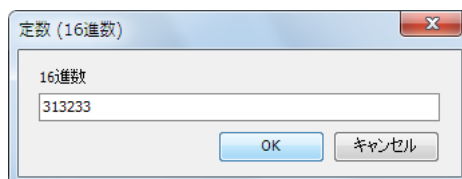
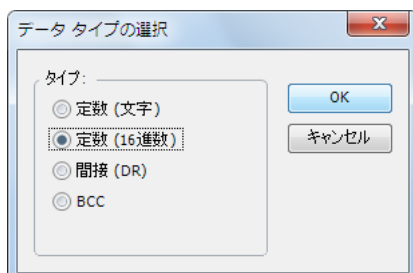
定数 (文字) → 123

定数 (16 進数) → 313233

[定数 (文字) で設定した場合]



[定数 (16 進数) で設定した場合]



間接（データレジスタ）

受信したデータを変換タイプの設定にしたがってデータ変換し、設定したバイト数分だけ設定したデータレジスタに格納します。リピート回数を設定することにより、設定したデータレジスタを先頭に設定したリピート回数分のデータを連続して格納できます。

設定項目	内容
データレジスタ番号	受信データを格納するデータレジスタを設定します。
変換タイプ	受信データをデータ変換するときの変換方法を設定します。通信相手機器の通信プロトコルに合わせて設定してください。 アスキー→バイナリ、アスキー→BCD→バイナリ、無変換の3種類から選択します。
受信バイト数	受信するデータのバイト数を設定します。変換タイプによって、設定できる受信バイト数が異なります。通信相手機器の通信プロトコルに合わせて設定してください。 アスキー→バイナリ変換： 1～4バイト アスキー→BCD→バイナリ変換： 1～5バイト 無変換： 1～2バイト
リピート回数	受信データを連続したデータレジスタに格納する場合に、同一フォーマットのデータを連続して格納できます。リピート回数は最大99回まで設定できます。
可変デリミタ	区切りとなる定数（デリミタ）を設定すると、可変長のデータを受信できます。デリミタまたは設定したデータ数（バイト数 × リピート回数）のデータを受信するまで、受信データを変換タイプの設定にしたがってデータ変換し、順次設定したデータレジスタに格納します。区切りとなる定数（デリミタ）は、16進数またはアスキーで設定します。

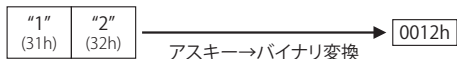
●変換タイプの設定と動作例

次の受信データを変換タイプに応じてデータ変換した場合



次のようにデータがデータレジスタに格納されます。

[変換タイプでアスキー→バイナリ変換を設定した場合]



[変換タイプでアスキー→BCD→バイナリ変換を設定した場合]



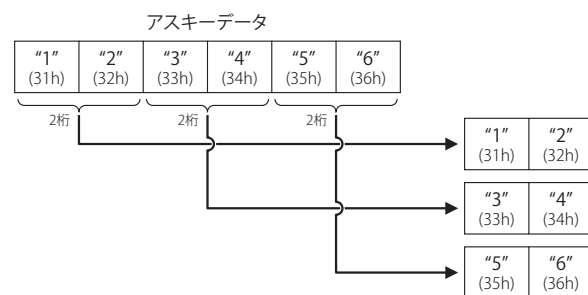
[変換タイプで無変換を設定した場合]



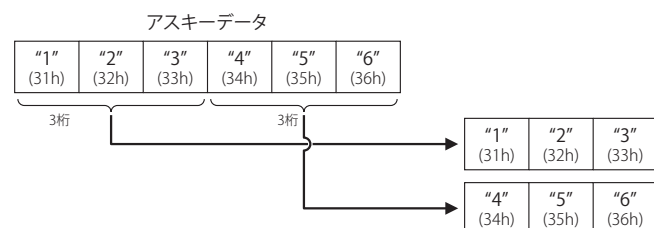
●受信バイト数の設定と動作例

アスキーデータ 6 バイトの受信データは、受信バイト数に応じて次のようにデータが分割されます。

[受信バイト数で 2 バイトを設定した場合]



[受信バイト数で 3 バイトを設定した場合]



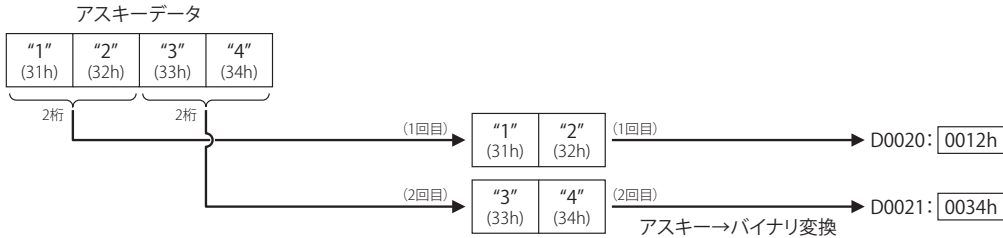
●リポート回数の設定と動作例

次の設定で受信データをデータ変換した場合

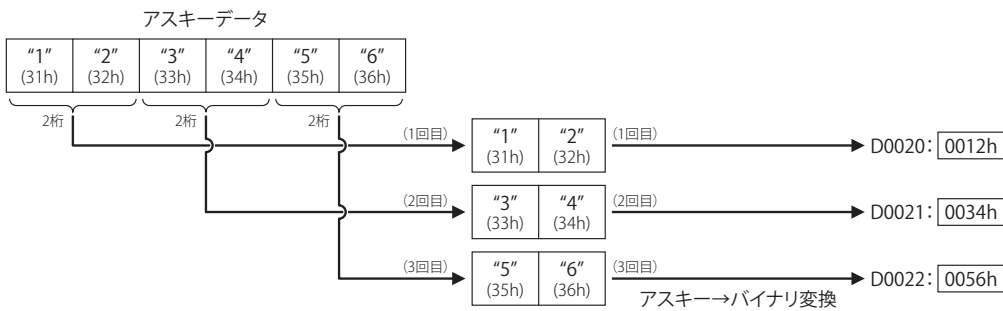
アスキーデータ						設定項目	データレジスタ番号	: 「D0020」
"1" (31h)	"2" (32h)	"3" (33h)	"4" (34h)	"5" (35h)	"6" (36h)	受信バイト数	: 「2 バイト」	
						変換タイプ	: 「アスキー→バイナリ変換」	

リポート回数に応じて次のデータがデータレジスタに格納されます。

[リポート回数で 2 回を設定した場合]



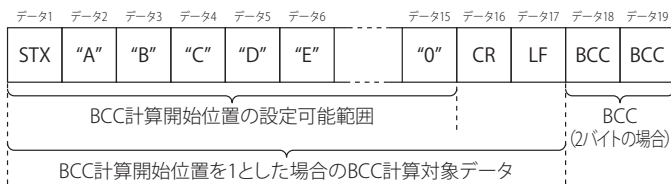
[リポート回数で 3 回を設定した場合]



BCC (Block Check Character)

外乱による受信時のデータ誤りを検出するために、BCC の計算および比較機能があります。任意の開始位置から終了位置までの BCC を計算し、受信した BCC コードと比較します。設定項目は通信相手機器の通信プロトコルに合わせて設定してください。

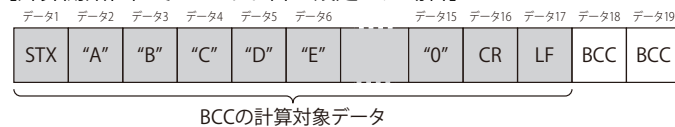
設定項目	設定内容
計算開始位置	BCCの計算を開始する位置を設定します。計算開始位置は、受信データの先頭から15バイト以内で設定します。計算終了位置は、BCCの直前になります。
計算方法	BCCの計算方法を設定します。排他的論理和(XOR)、加算(ADD)、ADD-2の補数、Modbus ASCII、Modbus RTUの5種類から選択します。
変換タイプ	データ変換するときのタイプを設定します。バイナリ→アスキー変換と無変換の2種類から選択します。
受信バイト数	受信データに付加するBCCコードのバイト数を1~2バイトで設定します。



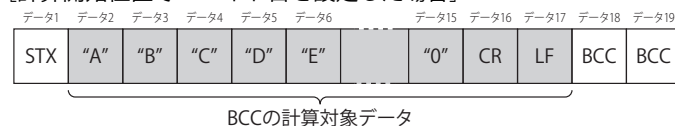
●計算開始位置の設定と動作例

設定した計算開始位置による BCC の計算対象データは次のとおりです。

[計算開始位置で 1 バイト目を設定した場合]



[計算開始位置で 2 バイト目を設定した場合]



●計算方法の設定と動作例

BCC の計算対象データが次の場合、

"A" (41h)	"B" (42h)	"C" (43h)	"D" (44h)
--------------	--------------	--------------	--------------

計算結果は次のとおりです。

[計算方法が排他的論理和 (XOR) の場合]

BCC 結果 = $41h \vee 42h \vee 43h \vee 44h = 04h$

[計算方法が加算 (ADD) の場合]

BCC 結果 = $41h + 42h + 43h + 44h = 010Ah$

[計算方法が ADD-2 の補数の場合]

BCC 結果 = $41h + 42h + 43h + 44h = 010Ah \longrightarrow$ FEF6h

↑
2 の補数に変換

[計算方法で Modbus ASCII を設定した場合]

BCC (LRC) 結果 = "8", "8" (アスキー)

[計算方法で Modbus RTU を設定した場合]

BCC (CRC-16) 結果 = 85h, 0Fh (バイナリ)



ADD-2 の補数

1. 計算開始位置から BCC 格納位置手前までのデータの和を計算します。
2. 1. の結果をビット反転し 1 を足します。(2 の補数)
3. 2. の結果を変換タイプ (BIN → ASCII, 無変換) と桁指定 (1 桁、2 桁) によって BCC 格納位置に格納します。

Modbus ASCII

1. 計算開始位置から BCC(LRC) 格納位置手前までの ASCII 文字を 2 文字単位で 1 バイト HEX データに変換 (例: 37h, 35h → 75h) します。
2. 1. の結果得られたデータの和を計算します。
3. 2. の結果をビット反転し 1 を足します。(2 の補数)
4. 3. の結果の下位 1 バイトのデータを ASCII 文字に変換 (例: 75h → 37h, 35h) し、BCC(LRC) 格納位置に格納します。
BCC の計算対象データが奇数バイト設定されている場合、BCC は不定となります。(Modbus プロトコル定義外)

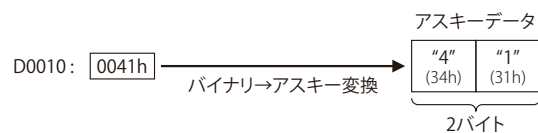
Modbus RTU

1. 計算開始位置の 1 バイトのデータと FFFFh の排他的論理和 (XOR) を取ります。
2. 1. の結果を右に 1 ビットシフトし、キャリーが出れば、固定値 (A001h) で XOR を取ります。
3. 8 回シフトするまで②を繰り返します。
4. 次の 1 バイトデータと③の結果の排他的論理和 (XOR) を取ります。
5. BCC (CRC) 格納位置手前のデータまで 2. ~ 4. を繰り返します。
6. 5. の結果を CRC-16 として BCC (CRC) 格納位置に下位上位 (例: 1234h → 34h, 12h) の順で格納します。

●変換タイプの設定と動作例

BCC の計算結果が "0041h" の場合、データ変換後のデータは次のとおりです。

[変換タイプでバイナリ→アスキー変換を設定した場合]



[変換タイプで無変換を設定した場合]



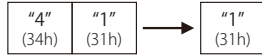
●バイト数設定の設定と動作例

BCCの計算結果が"0041h"の場合、受信データに付加するデータは次のとおりです。

[バイト数設定で2バイトを設定した場合]



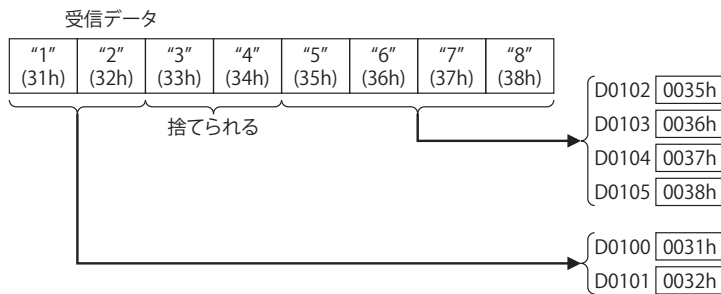
[バイト数設定で1バイトを設定した場合]



スキップ

スキップを設定すると、スキップを設定した位置から設定したバイト数の受信データは読み捨てられて、データレジスタには格納されません。たとえば、受信データの途中の決まった位置にデータレジスタに格納する必要がない定数が含まれており、その定数を定数照合に使用しない場合などに使用できます。連続してスキップできる数は最大99バイトです。

[受信データの3バイト目からスキップ（2バイト）の設定があり、格納するデータレジスタの先頭がD0100の場合]



■D1 (デスティネーション1) の設定

内部リレーまたは出力を受信完了出力として設定します。

RXD 命令の起動入力 ON して、受信前処理→データ受信→受信データ展開の一連の処理を終えた時点で、受信完了出力は ON します。

データ受信完了条件

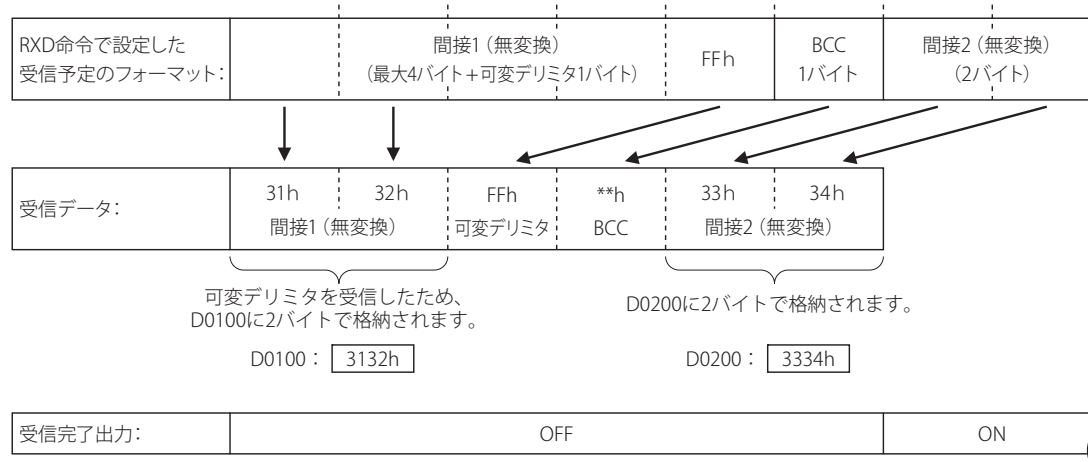
間接設定の可変デリミタ設定有無および、エンドデリミタの設定有無により、データ受信完了条件が異なります。

エンドデリミタ	可変デリミタ	データ受信完了条件
有	有/無	RXD命令で設定されたバイト数のデータを受信するか、エンドデリミタを受信して受信を完了します。ただし、エンドデリミタの直後にBCC設定がある場合は、BCCまで受信して受信を完了します。
無	有	RXD命令で設定されたデータ数のデータを受信して受信を完了します。
無	無	RXD命令で設定されたデータ数のデータを受信して受信を完了します。



受信タイムアウトが発生した場合は、無条件で受信を終了します。

[RXD 命令でエンドデリミタの設定なしで、間接設定の可変デリミタを設定した場合]



■D2 (デスティネーション2) の設定

受信動作ステータスおよび受信バイト数を格納するデータレジスタを設定します。

受信動作ステータスは D2+0 に、受信バイト数は D2+1 に格納します。



受信動作ステータスとして D0100 を設定した場合、D0100 を先頭に連続した 2 個のデータレジスタを使用します。

D0100: ← 受信動作ステータス

D0101: ← 受信バイト数

D2+0 (受信動作ステータス)

D2 で設定したデータレジスタには受信動作ステータスを格納します。受信動作ステータスは、受信動作状態とエラー情報を含みます。

ステータスコード*3	受信状態	状態説明
16	受信前処理中	RXD命令の起動入力ONしてから、受信フォーマットを解釈し、受信命令がデータ受信可能となるまでの前処理中の状態
32	受信中	受信命令がデータ受信可能となってから、すべてのデータを受信するまでの間
48	受信データ展開中	データ受信が終了した後、受信フォーマットにしたがって受信データを処理し、データレジスタに格納するまでの間
64	受信完了	データ受信が完了し、もう一度このRXD命令の起動が可能となる状態。この状態になると受信命令の実行が可能になり、受信命令が実行されると受信前処理中 (ステータスコード16) の状態に移ります。
128	受信キャンセル	ユーザー通信受信命令キャンセルフラグ (M8022、M8023、M8026、M8033、M8145~M8147、M8170、M8176、M8365~M8394) によって、受信命令の実行をキャンセルし、終了した状態

*3 ステータスコードが上記以外の場合には、受信命令の実行でエラーが発生しています。詳細は、「ユーザー通信送信命令・受信命令のエラー」(5-53 頁) を参照してください。

ユーザー通信受信命令キャンセルフラグ

ユーザー通信受信命令の受信前処理が既に完了し、受信中（ステータスコード32）となっている状態でユーザー通信受信命令キャンセルを ON すると、該当ポートに対するすべての受信命令の実行をキャンセルします。受信データ待ちの状態が長く、受信命令の実行をキャンセルしたい場合に有効です。

キャンセルした受信命令をアクティブにする場合は、ユーザー通信受信命令キャンセルフラグを OFF したあと、受信命令の入力条件を再度 ON にしてください。

ユーザー通信受信命令キャンセルフラグは、各通信ポートに特殊内部リレーとして次のように割り付けられます。

特殊内部リレー	内容	ストップ時	停電時	読み出し / 書き込み
M8022	ユーザー通信受信命令キャンセル（ポート1）	クリア	クリア	R/W
M8023	ユーザー通信受信命令キャンセル（ポート2）	クリア	クリア	R/W
M8026	ユーザー通信受信命令キャンセル（ポート3）	クリア	クリア	R/W
M8033	ユーザー通信受信命令キャンセル（ポート4）*1	クリア	クリア	R/W
M8145	ユーザー通信受信命令キャンセル（ポート5）*1	クリア	クリア	R/W
M8146	ユーザー通信受信命令キャンセル（ポート6）*1	クリア	クリア	R/W
M8147	ユーザー通信受信命令キャンセル（ポート7）*1	クリア	クリア	R/W
M8170	ユーザー通信受信命令キャンセル（ポート8）*1	クリア	クリア	R/W
M8176	ユーザー通信受信命令キャンセル（ポート9）*1	クリア	クリア	R/W
M8365	ユーザー通信受信命令キャンセル（ポート10）*1*2	クリア	クリア	R/W
M8366	ユーザー通信受信命令キャンセル（ポート11）*1*2	クリア	クリア	R/W
M8367	ユーザー通信受信命令キャンセル（ポート12）*1*2	クリア	クリア	R/W
M8370	ユーザー通信受信命令キャンセル（ポート13）*1*2	クリア	クリア	R/W
M8371	ユーザー通信受信命令キャンセル（ポート14）*1*2	クリア	クリア	R/W
M8372	ユーザー通信受信命令キャンセル（ポート15）*1*2	クリア	クリア	R/W
M8373	ユーザー通信受信命令キャンセル（ポート16）*1*2	クリア	クリア	R/W
M8374	ユーザー通信受信命令キャンセル（ポート17）*1*2	クリア	クリア	R/W
M8375	ユーザー通信受信命令キャンセル（ポート18）*1*2	クリア	クリア	R/W
M8376	ユーザー通信受信命令キャンセル（ポート19）*1*2	クリア	クリア	R/W
M8377	ユーザー通信受信命令キャンセル（ポート20）*1*2	クリア	クリア	R/W
M8380	ユーザー通信受信命令キャンセル（ポート21）*1*2	クリア	クリア	R/W
M8381	ユーザー通信受信命令キャンセル（ポート22）*1*2	クリア	クリア	R/W
M8382	ユーザー通信受信命令キャンセル（ポート23）*1*2	クリア	クリア	R/W
M8383	ユーザー通信受信命令キャンセル（ポート24）*1*2	クリア	クリア	R/W
M8384	ユーザー通信受信命令キャンセル（ポート25）*1*2	クリア	クリア	R/W
M8385	ユーザー通信受信命令キャンセル（ポート26）*1*2	クリア	クリア	R/W
M8386	ユーザー通信受信命令キャンセル（ポート27）*1*2	クリア	クリア	R/W
M8387	ユーザー通信受信命令キャンセル（ポート28）*1*2	クリア	クリア	R/W
M8390	ユーザー通信受信命令キャンセル（ポート29）*1*2	クリア	クリア	R/W
M8391	ユーザー通信受信命令キャンセル（ポート30）*1*2	クリア	クリア	R/W
M8392	ユーザー通信受信命令キャンセル（ポート31）*1*2	クリア	クリア	R/W
M8393	ユーザー通信受信命令キャンセル（ポート32）*1*2	クリア	クリア	R/W
M8394	ユーザー通信受信命令キャンセル（ポート33）*1*2	クリア	クリア	R/W

*1 通信モジュール使用時

*2 Plus CPU モジュールのみ



「R/W は、Read（読み出し）/Write（書き込み）の略で、R/W の場合は読み出し・書き込み可能、R の場合は読み出しのみ可能、W の場合は書き込みのみ可能です。

D2+1（受信バイト数）

D2 で設定したデータレジスタの次のデータレジスタ（D2+1）には受信バイト数を格納します。実際に受信したバイト数を格納します。スタートデリミタ、エンドデリミタを含むすべての受信データのバイト数を計数します。

●スタートデリミタの設定

RXD 命令には、スタートデリミタを設定できます。スタートデリミタを設定すると、RXD 命令はスタートデリミタが一致した受信データのみ受信し、処理します。

スタートデリミタは、RXD 命令の先頭（受信データの1バイト目）から設定する必要があります。1つのRXD 命令に対して連続した最大5バイトの定数をスタートデリミタとして設定できます。

スタートデリミタは、WindLDR で [タイプ] を “定数（文字）” または “定数（16進数）” に設定します。設定方法は、「定数」（5-10頁）を参照してください。

■スタートデリミタを設定する場合

RXD 命令は、同時に5つまで起動できます。

同時に起動するRXD 命令には、それぞれ異なるスタートデリミタを設定する必要があります。

受信データは、スタートデリミタが一致したRXD 命令の受信フォーマットにしたがって、変換処理されます。

受信したデータとスタートデリミタが一致しなかったRXD 命令は、その受信データを破棄して、次のデータ（スタートデリミタ）の受信待ちとなります。

スタートデリミタが同じRXD 命令を2つ以上同時に実行した場合は、2つ目以降に起動したRXD 命令の受信動作ステータスにエラーコード“5”を格納します。この状態で、スタートデリミタが一致するデータを受信した場合、先に起動したRXD 命令のみ受信動作を行います。

スタートデリミタが同じRXD 命令を2つ以上起動した場合の受信動作ステータスの動作例


1. RXD 命令①を起動します。RXD 命令①の受信動作ステータスに“32”を格納します。

RXD命令①:	定数 05h	間接	定数 0Dh	受信動作ステータス :32 受信完了出力 :0
RXD命令②:	定数 05h	間接	定数 0Ah	受信動作ステータス :0 受信完了出力 :0

2. RXD 命令②を起動します。

RXD命令①:	定数 05h	間接	定数 0Dh	受信動作ステータス :32 受信完了出力 :0
RXD命令②:	定数 05h	間接	定数 0Ah	受信動作ステータス :5 受信完了出力 :0

同一スタートデリミタ（05h）の命令が起動中のため、受信動作ステータスにエラーコード“5”を格納します。

 スタートデリミタのデータ長が異なるRXD 命令を2つ以上同時に起動した場合で、下図のようにスタートデリミタが途中まで同じ場合も、同じスタートデリミタと見なして、2つ目以降に起動したRXD 命令の受信動作ステータスにエラーコード“5”を格納します。

:スタートデリミタ :スタートデリミタ以外のデータ

	バイト	1	2	3	4	5	6	7
RXD命令①:	定数 01h	(間接)	(定数)					
RXD命令②:	定数 01h	定数 02h	(SKIP)	(定数)				
RXD命令③:	定数 01h	定数 02h	定数 03h	定数 04h	定数 05h	(間接)		

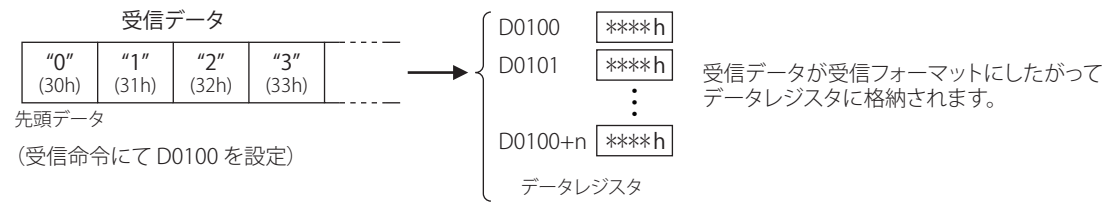
※RXD命令①、②、③のいずれの組み合わせも、同一スタートデリミタとみなします。

■スタートデリミタを設定しない場合

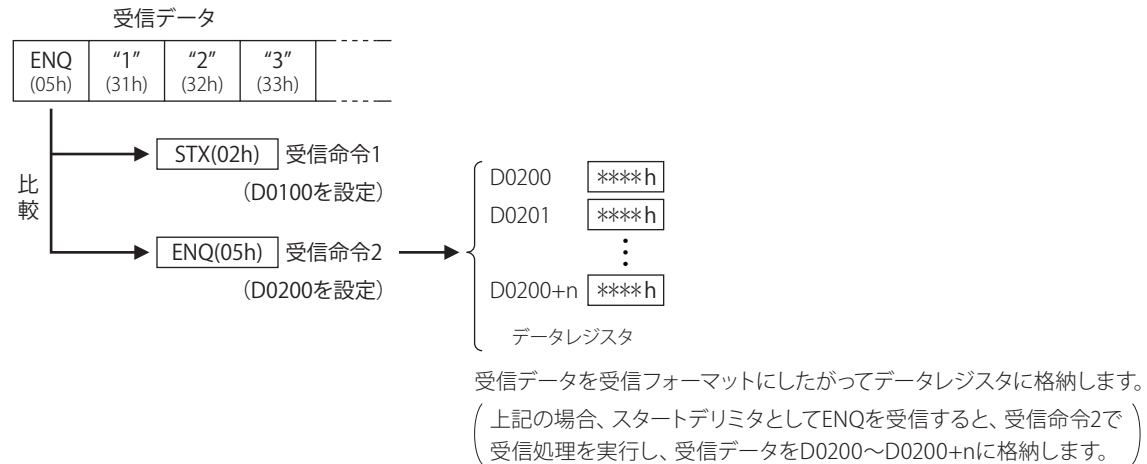
同時に起動できるRXD 命令は、1つのみです。同時に2つ以上のRXD 命令を実行できません。RXD 命令を順次実行し、データを受信します。

■ スタートデリミタの設定と動作例

[スタートデリミタを設定しない場合]
 順次データを受信します。



[スタートデリミタの設定が STX (02h) と ENQ (05h) の場合]
 同時に 2 つの RXD 命令を起動できます。



- スタートデリミタを2バイト以上設定している状態で、受信したデータがスタートデリミタと一致しなかった場合は、それまでに受信したデータ（スタートデリミタ）を破棄して、再度スタートデリミタの1バイト目の受信待ちとなります。
- スタートデリミタの1バイト目が検出された時点で受信キャラクタ間タイマが起動されます。スタートデリミタを2バイト以上設定している状態で、受信タイムアウトを経過してもスタートデリミタの2バイト目以降を受信しない場合は、スタートデリミタの受信途中でも受信タイムアウトとしてデータ受信を終了します。受信タイムアウトは、WindLDRの「通信設定」で設定できます。
- スタートデリミタ設定の有無にかかわらず、データを1バイト受信すると、受信キャラクタ間タイマが起動します。データを1バイト受信するまでは、受信待ちでも受信タイムアウトにはなりません。

●エンドデリミタの設定

RXD 命令には、エンドデリミタを設定できます。受信の終了を判定するコードとして使用し、1つのRXD 命令に対して1バイトの定数を設定します。

■エンドデリミタを設定する場合

WindLDR で [タイプ] を「定数 (文字)」または「定数 (16 進数)」に設定します。設定方法は、「定数」(5-10 頁) を参照してください。

1つのRXD 命令で、エンドデリミタとエンドデリミタ以外の定数が同じコードにならないようにしてください。間接設定の受信データに含まれるコードは、エンドデリミタに使用しないでください。間接設定の受信データ中にエンドデリミタと同じコードが含まれていた場合、受信したコードをエンドデリミタとみなしてデータの受信を終了します。

受信データが S1 で設定した受信データの総バイト数に満たない場合でも、エンドデリミタを検出するとデータ受信を終了します。エンドデリミタの後に BCC コードがある場合は、BCC コードを受信してからデータ受信を終了します。エンドデリミタと BCC コードの設定の間に他の受信データの設定 [変数 (データレジスタ)] があると、定数はエンドデリミタではなく、照合用の定数となります。

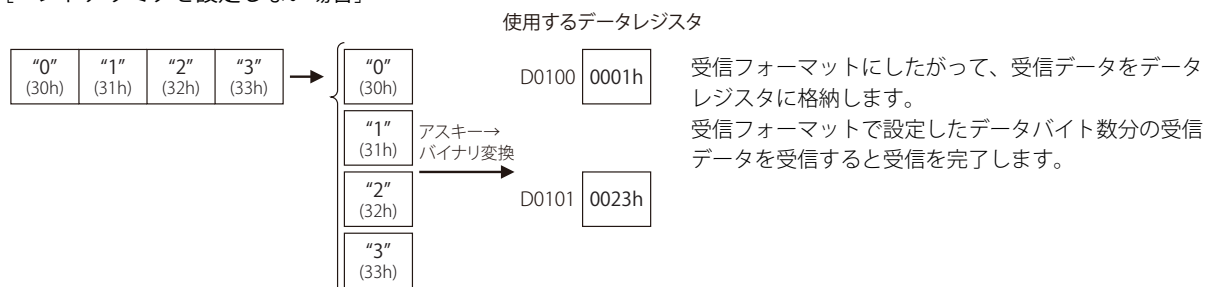
■エンドデリミタを設定しない場合

変数・スキップなどの設定分のデータをすべて受信して処理を終了します。ただし、1バイトのデータを受信した時点から次の1バイトのデータを受信するまでの時間を監視するための受信キャラクタ間タイムアウトが起動します。受信キャラクタ間タイムアウトはデータを1バイト受信するごとにリセットし、「ファンクション設定」の受信タイムアウトとして設定した時間を経過しても受信データが来ない場合には、受信タイムアウトとしてデータ受信を終了します。

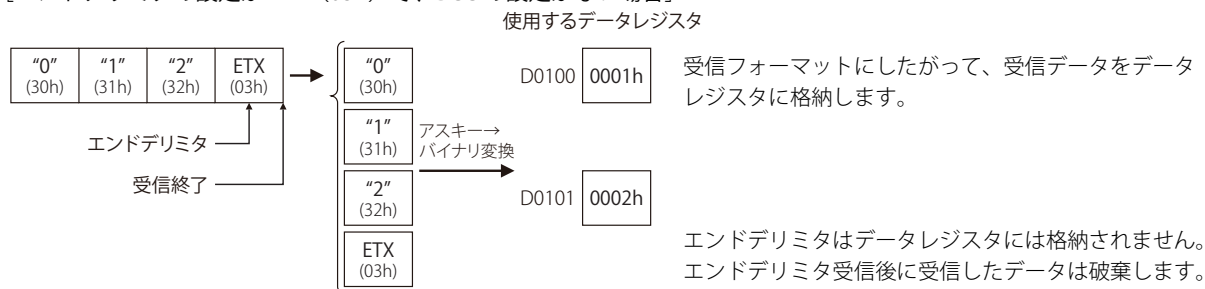
■エンドデリミタの設定と動作例

受信命令にエンドデリミタが設定されている場合の動作は次のとおりです。

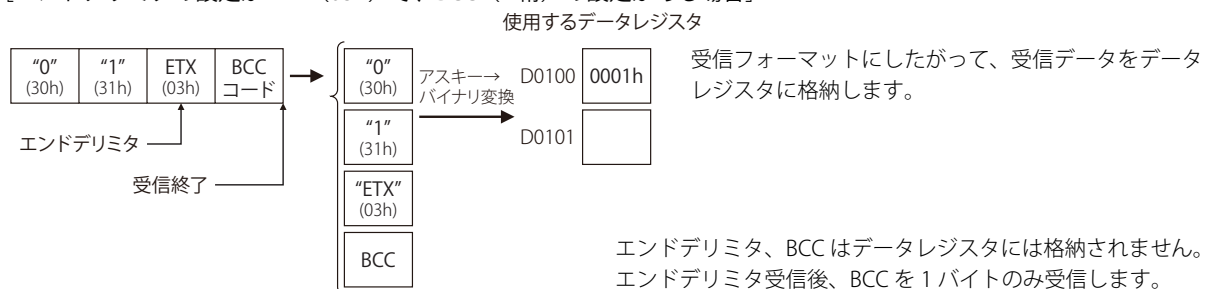
[エンドデリミタを設定しない場合]



[エンドデリミタの設定が ETX (03h) で、BCC の設定がない場合]



[エンドデリミタの設定が ETX (03h) で、BCC (1 桁) の設定がある場合]




●定数設定によるデータ照合

スタートデリミタとエンドデリミタ以外でも、定数（文字）または定数（16進数）を設定して、受信データを照合できます。受信データの照合用データとして、1つのRXD命令に対して複数の定数を設定できます。RXD命令で設定したすべての定数を照合します。照合結果は受信動作ステータスに格納します。

定数設定によるデータ照合を使用して、次のような場合にユーザープログラムを簡単にすることができます。

[受信データの途中に定数が存在する通信プロトコルを扱う場合]

中間（4バイト目）の定数を照合して、不一致の場合はRXD命令の受信動作ステータスにエラーコード“10”を格納します。							
バイト	1	2	3	4	5	6	7
受信データ1:	STX	1	2	定数	3	4	CR
ユーザー通信受信命令1:	スタートデリミタ	間接設定		定数	間接設定		エンドデリミタ

 RXD命令で、定数と受信データの照合結果が不一致の場合、その定数がスタートデリミタの場合と照合用の定数の場合で受信動作が異なります。

[スタートデリミタの照合が不一致の場合]


RXD命令:	定数 05h	間接	定数 FFh	間接	定数 0Dh	受信動作ステータス :7 受信完了出力 :0
受信データ:	50h	**h	FFh	**h	0Dh	

受信動作ステータスにエラーコード“7”を格納し、受信完了出力はONしません。
※ 受信中の状態を継続します。以後、スタートデリミタが一致する受信データを正常に受信した場合、受信を終了します。

[照合用定数の照合が不一致の場合]

RXD命令:	定数 05h	間接	定数 FFh	間接	定数 0Dh	受信動作ステータス :74 受信完了出力 :1
受信データ:	05h	**h	0Fh	**h	0Dh	

定数（FFh）と受信データ（0Fh）とが不一致のため、受信動作ステータスに“74”（ステータスコード“64”にエラーコード“10”を加算した値）を格納し、受信完了出力をONします。
※ 受信を終了します。同じ受信命令を実行する場合は、受信命令の入力条件を再度ONにしてください。

 RXD命令で、スタートデリミタとして設定できる最大バイト数（5バイト）を超えて定数を設定している場合や、定数と定数以外（間接、BCC、SKIP）を混在して設定している場合は、各定数の扱いが異なります。

[スタートデリミタとして設定可能な最大バイト数（5バイト）を超えて定数設定している場合]

バイト	1	2	3	4	5	6	...	200
RXD命令:	定数	定数	定数	定数	定数	定数	...	定数

5バイト目までの定数をスタートデリミタとして扱います。 6バイト目以降の定数は照合用の定数として扱います。

[定数と定数以外（間接、BCC、SKIP）を混在して設定している場合]

バイト	1	2	3	4	5	6	...	200
RXD命令:	定数	*1	定数	定数	定数		...	

命令の先頭から連続した最大5バイト目までの定数をスタートデリミタとして扱います。 5バイト目までの定数でも命令の先頭から連続していない場合は照合用の定数として扱います。

:スタートデリミタとして扱う定数

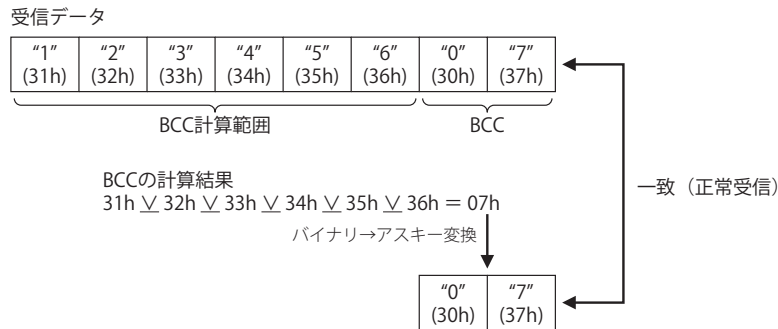
:照合用として扱う定数（エンドデリミタ含む）

*1 :間接設定、BCC、SKIP

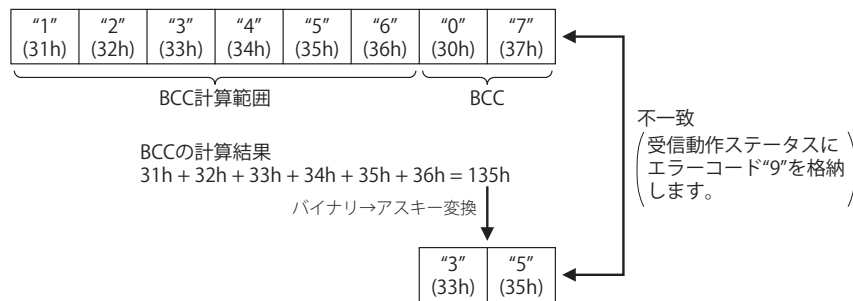
● BCCの比較機能

受信した BCC と受信したデータから計算した BCC を比較して、外乱などによる受信データの誤りを検出します。比較した結果は受信動作ステータスに格納します。

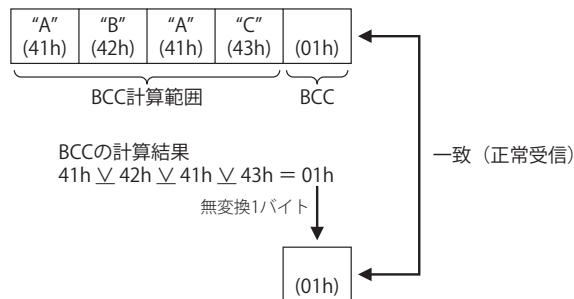
[1～6バイト目までの受信データの排他的論理和を計算した後、バイナリ→アスキー変換し、7～8バイト目に付加された BCC と比較する場合]



[1～6バイト目までの受信データを加算した後、バイナリ→アスキー変換し、7～8バイト目に付加された BCC と比較する場合]



[1～4バイト目までの受信データの排他的論理和を計算した後、無変換で、5バイト目に付加された BCC と比較する場合]



ETXD (イーサネットユーザー通信送信)

Ethernet ポート 1 および 2 を使用して外部機器へ、送信データを設定したデータタイプに変換して送信します。

ラダー図



動作説明

入力が ON すると、S1 で設定した送信データを設定した接続で接続された機器へ送信します。

すべてのデータの送信が完了すると、D1 で設定したデバイスを ON し、D2 で設定したデバイスに送信動作のステータス（送信動作の遷移状態とエラーコード）を格納します。

D2+1 には送信したデータのバイト数を格納します。

接続設定以外の設定は、TXD 命令と同じです。TXD 命令については、「TXD (ユーザー通信送信)」(5-2 頁) を参照してください。



ユーザー通信クライアント、ユーザー通信サーバーおよびユーザー通信(UDP)の仕様、イーサネット通信でのユーザー通信については、「イーサネット通信でのユーザー通信」(5-39 頁) を参照してください。

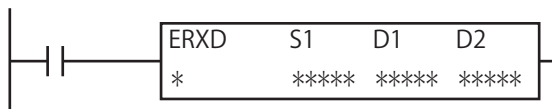


ETXD 命令は割込プログラムでは使用できません。割込プログラムで ETXD 命令を使用すると、ユーザープログラム実行エラーコード (D8006) にエラーコード 18 が格納されます。ユーザープログラム実行エラーについては、ラダープログラミングマニュアル「第 3 章 ●ユーザープログラム実行エラー」を参照してください。

ERXD (イーサネットユーザー通信受信)

Ethernet ポート 1 および 2 を使用して外部機器からデータを受信し、適切なデータタイプに変換してデータレジスタに格納します。

ラダー図



動作説明

入力が ON すると、S1 で設定した受信データを設定した接続で接続された機器から受信します。

すべてのデータの受信が完了すると、D1 で設定したデバイスを ON し、D2 で設定したデバイスに受信動作のステータス（受信動作の遷移状態とエラーコード）を格納します。

D2+1 には実際に受信したデータのバイト数を格納します。

受信データ待ちの状態ユーザー通信受信キャンセルフラグ (M8200 ~ M8207、M8334 ~ M8343) を ON すると、対応する接続に対して実行中のすべての受信命令の実行をキャンセルします。

接続設定およびユーザー通信受信キャンセルフラグの割付以外の設定は、RXD 命令と同じです。RXD 命令については、「RXD (ユーザー通信受信)」(5-9 頁) を参照してください。



ユーザー通信クライアント、ユーザー通信サーバーおよびユーザー通信(UDP)の仕様、イーサネット通信でのユーザー通信については、「イーサネット通信でのユーザー通信」(5-39 頁) を参照してください。

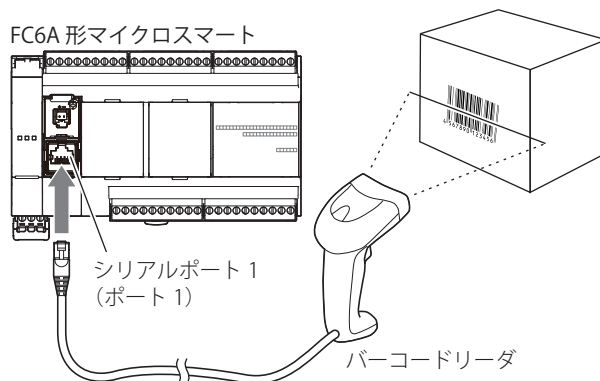


ERXD 命令は割込プログラムでは使用できません。割込プログラムで ERXD 命令を使用すると、ユーザープログラム実行エラーコード (D8006) にエラーコード 18 が格納されます。ユーザープログラム実行エラーについては、ラダープログラミングマニュアル「第 3 章 ●ユーザープログラム実行エラー」を参照してください。

シリアル通信でのユーザー通信

シリアル通信でのユーザー通信により、TXD（ユーザー通信送信）命令および RXD（ユーザー通信受信）命令を使用することで、ポート 1～33 に接続したプリンタ、バーコードリーダなどの外部機器とデータの送受信ができます。

バーコードリーダで読み取ったデータを受信する例
FC6A 形のポート 1 にバーコードリーダを接続します。



通信仕様

項目	通信フォーマット	
	RS232C	RS485
電気的特性	EIA RS232C規格準拠	EIA RS485規格準拠
使用可能ポート	ポート1～33	ポート1～33
1ポートあたりの最大接続台数	1台	最大31台
通信速度 (bps)	115200 ^{*1} (1200/2400/4800/9600/19200/38400/57600/115200)	
ストップビット	1bit ^{*1} (1bit/2bit)	
受信タイムアウト ^{*2}	500ms ^{*1} (10ms～2540ms)	
通信モード	調歩同期方式	
最大ケーブル長	FC6A-PC1 : 5m FC6A-SIF52 : 15m	FC6A-PC3 : 200m FC6A-SIF52 : 1200m
最大送信データ数	1500バイト	
最大受信データ数	1500バイト	
BCCの設定	XOR/ADD/ADD-2の補数/Modbus ASCII/Modbus RTU	

*1 () 外の数値はデフォルト値です。

*2 受信タイムアウトは、RXD 命令を使うときに有効です。受信タイムアウトをなしに設定する場合は、2550ms に設定してください。

■ 設定の流れ

- ① 外部機器との接続に使用する通信ポートの通信フォーマットを設定する
「通信ポートと通信フォーマットの設定」(5-24 頁)
- ② ユーザー通信命令を使用したユーザープログラムを作成して、FC6A 形にダウンロードする
「ユーザー通信命令の設定」(5-25 頁)

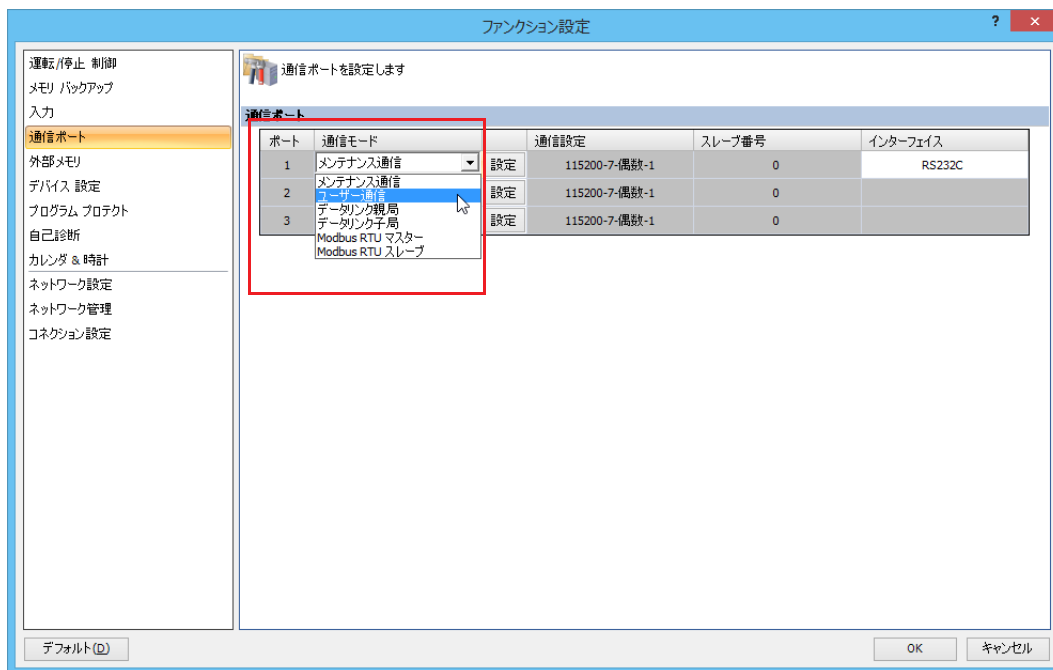
通信ポートと通信フォーマットの設定

外部機器との接続に使用する通信ポートの通信フォーマットを設定します。
次の例では、WindLDR を使用してポート 1 にユーザー通信を設定する手順について説明します。

WindLDR の設定

● 操作手順

1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. ポート 2 の [通信モード] をクリックし、“ユーザー通信” を選択します。
[ユーザー通信 (ポート 1)] ダイアログボックスが表示されます。



3. 外部機器の通信フォーマットに合わせて、各設定項目を設定します。



4. [OK] ボタンをクリックします。

以上で、通信設定は完了です。

次にユーザー通信命令を使用したユーザープログラムを作成して、FC6A 形にダウンロードします。詳細は、「ユーザー通信命令の設定」(5-25 頁) を参照してください。

ユーザー通信命令の設定

シリアル通信でのユーザー通信は、TXD（ユーザー通信送信）命令および RXD（ユーザー通信受信）命令を使用します。

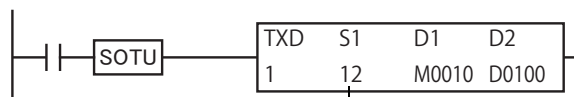
■TXD（ユーザー通信送信）命令の場合

送信データが、スタートデリミタ、送信するデータを格納するデータレジスタ、BCC、エンドデリミタを含む場合を例として、WindLDR で TXD 命令を設定する手順について説明します。



TXD 命令については、「TXD（ユーザー通信送信）」(5-2 頁) を参照してください。

ラダー図



ラダー図上では、TXD 命令の S1 に設定した「送信データ」の総バイト数が表示されます。

TXD（ユーザー通信送信）命令の設定内容

項目	設定内容			
通信ポート	ポート1			
送信データ	BCC計算範囲 			
	定数 (16進数)	スタートデリミタ (STX)	02h	
	間接 (データレジスタ)	データレジスタ番号	D0010	
		変換タイプ	バイナリ→アスキー	
		送信バイト数設定	4 (バイト)	
	BCC	リピート回数	2 (回)	
		計算開始位置	1 (バイト目)	
計算方式		ADD		
定数 (16進数)	変換タイプ	バイナリ→アスキー		
定数 (16進数)	バイト数設定	2 (バイト)		
定数 (16進数)	エンドデリミタ (ETX)	03h		
送信完了出力	M0010			
送信動作ステータス	D0100			
送信バイト数	D0101			

データレジスタの設定内容

デバイスアドレス	設定内容
D0010	1234 (04D2h)
D0011	5678 (162Eh)

WindLDR の設定

●操作手順

1. TXD 命令を入力します。

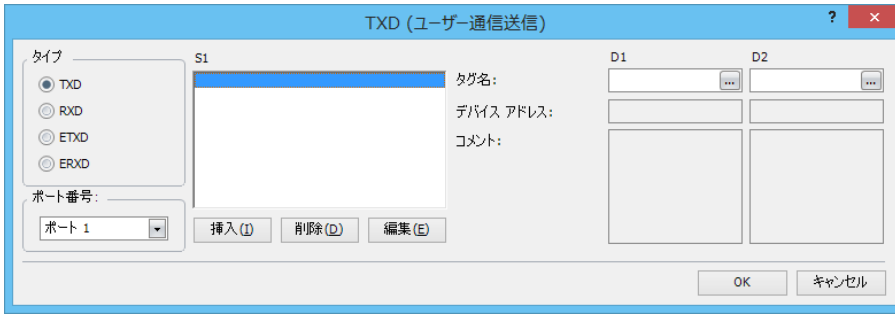
ユーザープログラムの TXD 命令を挿入する場所にカーソルを移動し、「TXD」と入力します。[コイル選択] ダイアログボックスが表示されます。[TXD] が選択されていることを確認し、[OK] ボタンをクリックします。

または、[ホーム] タブの [命令] で [演算] をクリックし、[ユーザー通信]、[TXD (ユーザー通信送信)] の順にクリックします。ユーザープログラムの TXD 命令を挿入する場所をクリックします。

[TXD (ユーザー通信送信)] ダイアログボックスが表示されます。

2. [タイプ] で“TXD” が選択されていることを確認し、[ポート番号] で“ポート 1”を選択します。次に、[挿入] ボタンをクリックします。

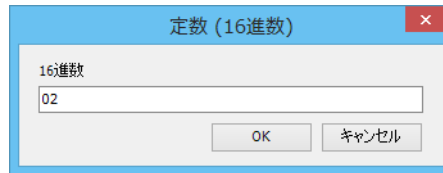
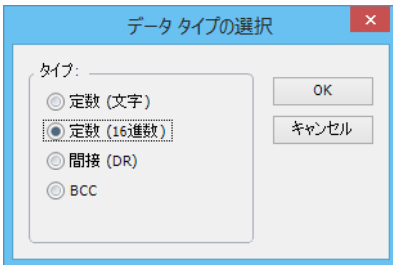
[データタイプ選択] ダイアログボックスが表示されます。



3. S1 (ソース 1) のデバイスを設定します。

[タイプ] で“定数 (16 進数)”を選択し、[OK] ボタンをクリックします。[定数 (16 進数)] ダイアログボックスが表示されます。スタートデリミタの STX (02h) を設定します。「02」を入力し、[OK] ボタンをクリックします。

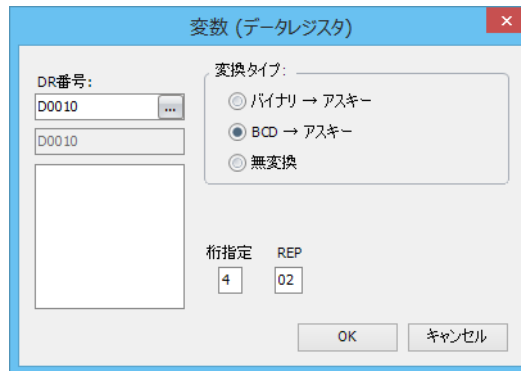
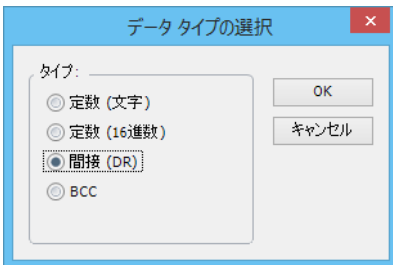
[TXD (ユーザー通信送信)] ダイアログボックスに戻ります。



4. [挿入] ボタンをクリックします。データタイプ選択のダイアログボックスが表示されます。

[タイプ] で“間接 (DR)” (間接 (データレジスタ)) を選択し、[OK] ボタンをクリックします。[変数 (データレジスタ)] ダイアログボックスが表示されます。送信するデータを格納するデータレジスタを設定します。[DR 番号] (データレジスタ番号) で「D0010」を入力し、[変換タイプ] で“バイナリ→アスキー”を選択します。次に [桁指定] (送信バイト数設定) で「4」(4 バイト)、[REP] (リピート回数) で「2」(2 回) を入力し、[OK] ボタンをクリックします。

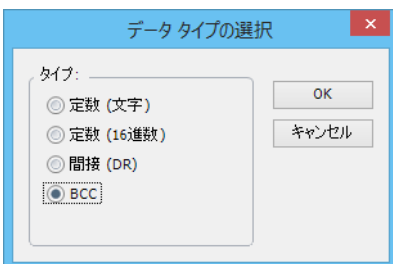
[TXD (ユーザー通信送信)] ダイアログボックスに戻ります。



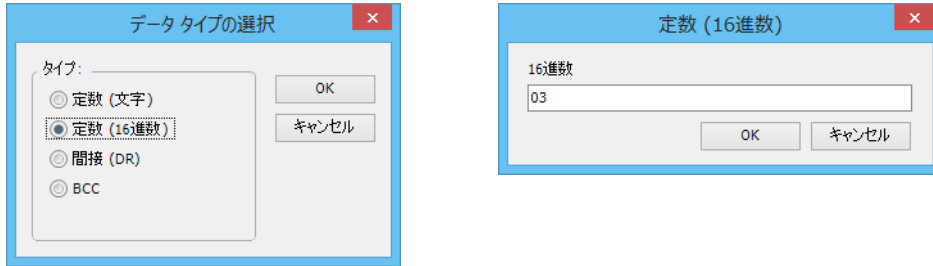
5. [挿入] ボタンをクリックします。[データタイプ選択] ダイアログボックスが表示されます。

[タイプ] で“BCC”を選択し、[OK] ボタンをクリックします。[BCC] ダイアログボックスが表示されます。BCC データについて設定します。[開始位置] (計算開始位置) で「1」(1 バイト目)を入力し、[計算方法] で“ADD”を選択します。[変換タイプ] で“バイナリ→アスキー”、[桁指定] (バイト数設定) で“2” (バイト) を選択し、[OK] ボタンをクリックします。

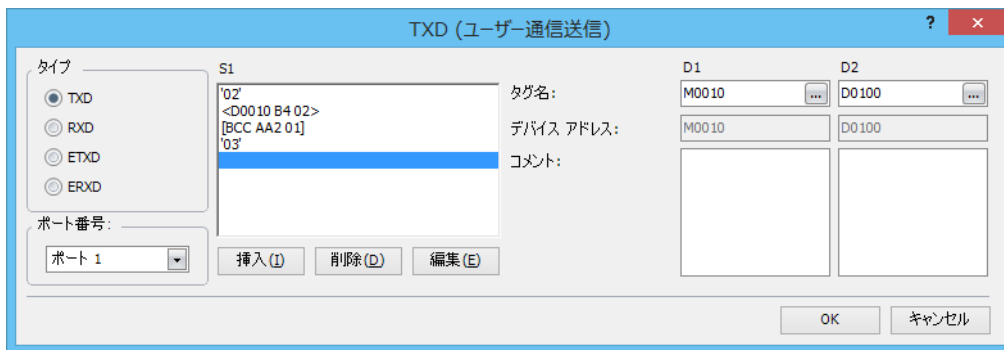
[TXD (ユーザー通信送信)] ダイアログボックスに戻ります。



6. [挿入] ボタンをクリックします。[データタイプ選択] ダイアログボックスが表示されます。[タイプ] で“定数 (16進数)” を選択し、[OK] ボタンをクリックします。[定数 (16進数)] ダイアログボックスが表示されます。エンドデリミタのETX (03h) を設定します。[16進数] で「03」を入力し、[OK] ボタンをクリックします。[TXD (ユーザー通信送信)] ダイアログボックスに戻ります。



7. [D1] (デスティネーション 1) で送信完了出力先「M0010」、[D2] (デスティネーション 2) で送信動作ステータス「D0100」を入力します。[OK] ボタンをクリックします。

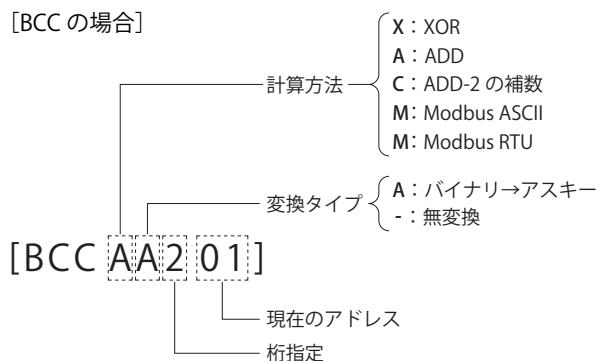
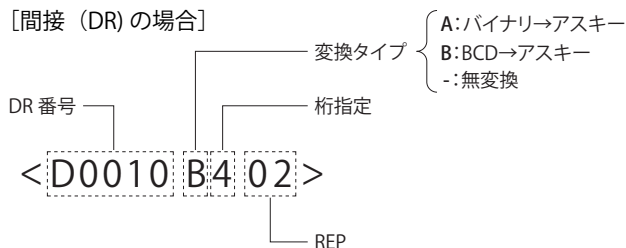


以上で、TXD 命令の設定が完了し、次のように送信データが設定されます。

STX (02h)	"1" (31h)	"2" (32h)	"3" (33h)	"4" (34h)	"5" (35h)	"6" (36h)	"7" (37h)	"8" (38h)	BCC (H) (41h)	BCC (L) (36h)	ETX (03h)
-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------	---------------	---------------	-----------

WindLDR の S1 の表記について

データタイプの選択で“間接 (DR)” または“BCC” を選択した場合、ダイアログボックスの S1 には次のように表示されます。



8. ユーザープログラムを FC6A 形にダウンロードします。

以上で、設定は完了です。

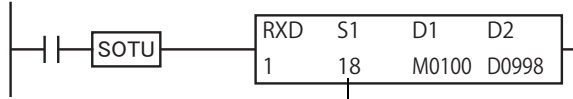
■RXD（ユーザー通信受信）命令の場合

受信データが、スタートデリミタ、受信データ照合用の定数、受信したデータをデータ変換して格納するデータレジスタ、BCC、エンドデリミタを含む場合を例として、WindLDR で RXD 命令を設定する手順について説明します。



RXD 命令については、「RXD（ユーザー通信受信）」（5-9 頁）を参照してください。

ラダー図



ラダー図上では、RXD 命令の S1 に設定した「受信データ」の総バイト数が表示されます。

RXD（ユーザー通信受信）命令の設定内容

項目	設定内容																																													
通信ポート	ポート1																																													
受信データ	<table border="1" style="width:100%; text-align:center; border-collapse: collapse;"> <tr> <td colspan="15">BCC計算範囲</td> </tr> <tr> <td>STX (02h)</td> <td>局番 (H) (00h)</td> <td>局番 (L) (10h)</td> <td>"0" (30h)</td> <td>"0" (30h)</td> <td>"." (2Ch)</td> <td>"1" (31h)</td> <td>"2" (32h)</td> <td>"3" (33h)</td> <td>"4" (34h)</td> <td>"5" (35h)</td> <td>CR (0Dh)</td> <td>BCC (H) (41h)</td> <td>BCC (L) (41h)</td> <td>ETX (03h)</td> </tr> <tr> <td colspan="3">スタートデリミタ</td> <td colspan="2">スキップ 範囲</td> <td colspan="1">照合用 定数</td> <td colspan="5">D0010~D0011</td> <td colspan="2">BCC</td> <td colspan="1">エンド デリミタ</td> </tr> </table>		BCC計算範囲															STX (02h)	局番 (H) (00h)	局番 (L) (10h)	"0" (30h)	"0" (30h)	"." (2Ch)	"1" (31h)	"2" (32h)	"3" (33h)	"4" (34h)	"5" (35h)	CR (0Dh)	BCC (H) (41h)	BCC (L) (41h)	ETX (03h)	スタートデリミタ			スキップ 範囲		照合用 定数	D0010~D0011					BCC		エンド デリミタ
	BCC計算範囲																																													
	STX (02h)	局番 (H) (00h)	局番 (L) (10h)	"0" (30h)	"0" (30h)	"." (2Ch)	"1" (31h)	"2" (32h)	"3" (33h)	"4" (34h)	"5" (35h)	CR (0Dh)	BCC (H) (41h)	BCC (L) (41h)	ETX (03h)																															
	スタートデリミタ			スキップ 範囲		照合用 定数	D0010~D0011					BCC		エンド デリミタ																																
	定数（16進数）	スタートデリミタ（STX）	02h																																											
		スレーブ番号（H）	00h																																											
		スレーブ番号（L）	10h																																											
	スキップ	バイト数設定	2（バイト）																																											
	定数（文字）	受信データ照合用の定数	","																																											
	間接（データレジスタ）	データレジスタ番号	D0010																																											
変換タイプ		アスキー→バイナリ																																												
受信バイト数設定		4（バイト）																																												
リピート回数		2（回）																																												
可変デリミタ		16進数、0Dh（CR）																																												
BCC	計算開始位置	1（バイト目）																																												
	計算方式	ADD																																												
	変換タイプ	バイナリ→アスキー																																												
	バイト数設定	2（バイト）																																												
定数（16進数）	03h（ETX）																																													
受信完了出力	M0100																																													
受信動作ステータス	D0998																																													
受信バイト数	D0999																																													

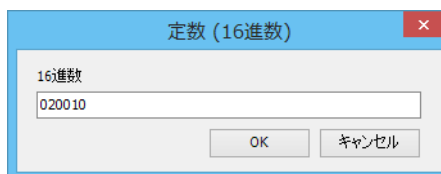
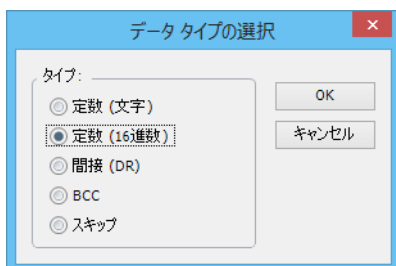
WindLDR の設定

●操作手順

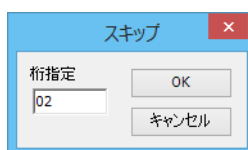
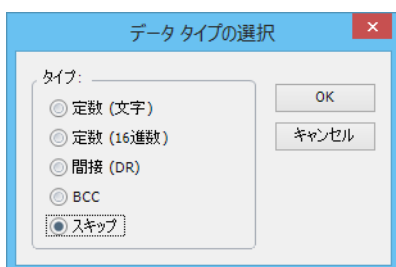
1. RXD 命令を入力します。
ユーザープログラムの RXD 命令を挿入する場所にカーソルを移動し、「RXD」と入力します。[コイル選択] ダイアログボックスが表示されます。「RXD」が選択されていることを確認し、[OK] ボタンをクリックします。
または、[ホーム] タブの [命令] で [演算] をクリックし、[ユーザー通信]、[RXD (ユーザー通信受信)] の順にクリックします。ユーザープログラムの RXD 命令を挿入する場所をクリックします。
[RXD (ユーザー通信受信)] ダイアログボックスが表示されます。
2. [タイプ] で「RXD」が選択されていることを確認し、[ポート番号] で「ポート 1」を選択します。次に、[挿入] ボタンをクリックします。
[データタイプ選択] ダイアログボックスが表示されます。



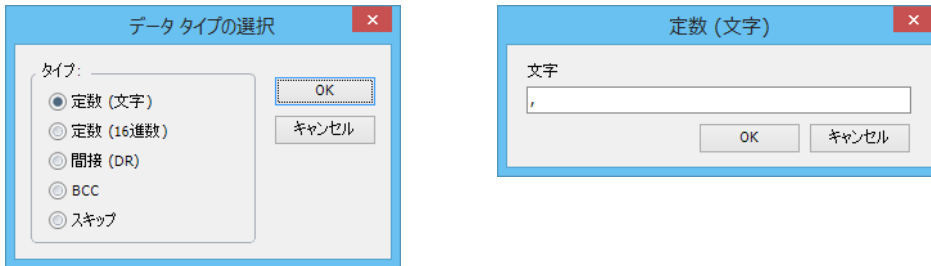
3. [S1] (ソース 1) のデバイスを設定します。
[タイプ] で「定数 (16 進数)」を選択し、[OK] ボタンをクリックします。[定数 (16 進数)] ダイアログボックスが表示されます。スタートデリミタの STX (02h)、スレープ番号 (H) (00h)、スレープ番号 (L) (10h) を設定します。「020010」を入力し、[OK] ボタンをクリックします。
[RXD (ユーザー通信受信)] ダイアログボックスに戻ります。



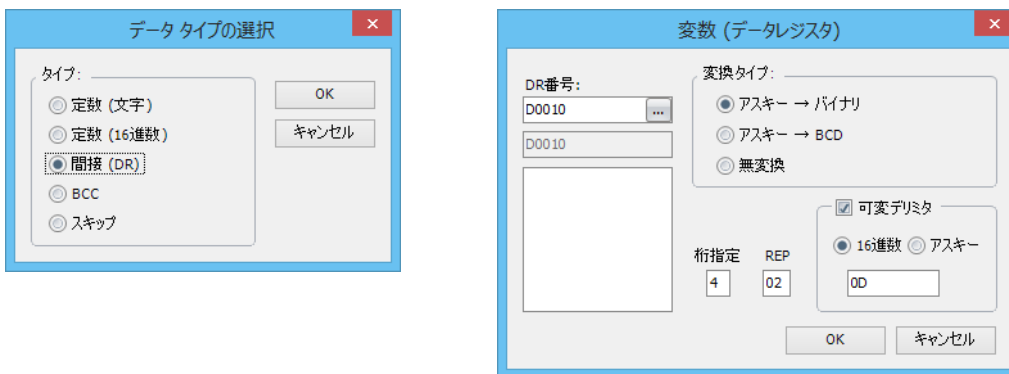
4. [挿入] ボタンをクリックします。[データタイプ選択] ダイアログボックスが表示されます。
[タイプ] で「スキップ」を選択し、[OK] ボタンをクリックします。[スキップ] ダイアログボックスが表示されます。読み捨てるデータのバイト数を設定します。[桁指定] (バイト数設定) で「02」(2 バイト) を入力し、[OK] ボタンをクリックします。
[RXD (ユーザー通信受信)] ダイアログボックスに戻ります。



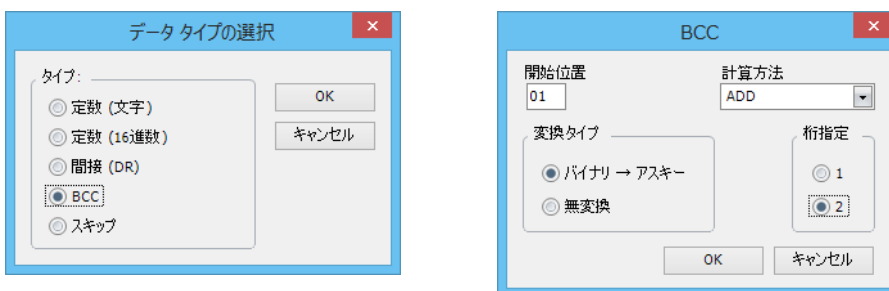
5. [挿入] ボタンをクリックします。[データタイプ選択] ダイアログボックスが表示されます。
 [タイプ] で“定数 (文字)” を選択し、[OK] ボタンをクリックします。[定数 (文字)] ダイアログボックスが表示されます。
 受信データ照合用の定数を設定します。[文字] で「,」 (2Ch) を入力し、[OK] ボタンをクリックします。
 [RXD (ユーザー通信受信)] ダイアログボックスに戻ります。



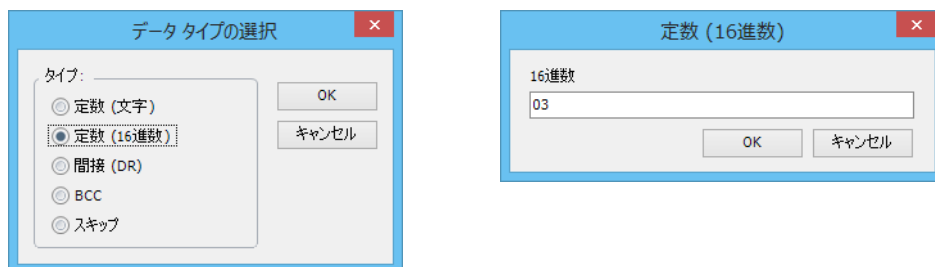
6. [挿入] をクリックします。[データタイプ選択] ダイアログボックスが表示されます。
 [タイプ] で“間接 (DR)” (間接 (データレジスタ)) を選択し、[OK] ボタンをクリックします。[変数 (データレジスタ)]
 ダイアログボックスが表示されます。受信したデータを格納するデータレジスタについて設定します。[DR 番号] (データレジ
 スタ番号) で「D0010」を入力し、[変換タイプ] で“アスキー→バイナリ”を選択します。次に [桁指定] (受信バイト数設
 定) で「4」 (4 バイト)、[REP] (リピート回数) で「02」 (2 回) を入力します。
 可変長のデータを受信するために [可変デリミタ] チェックボックスをオンにし、“16 進数” を選択します。区切りとなる定数
 (デリミタ) として「0D」を入力し、[OK] ボタンをクリックします。
 [RXD (ユーザー通信受信)] ダイアログボックスに戻ります。



7. [挿入] をクリックします。[データタイプ選択] ダイアログボックスが表示されます。
 [タイプ] で“BCC” を選択し、[OK] ボタンをクリックします。[BCC] ダイアログボックスが表示されます。BCC データにつ
 いて設定します。[現在のアドレス] (計算開始位置) で「01」 (1 バイト目) を入力し、[計算方法] で“ADD” を選択します。
 [変換タイプ] で“バイナリ→アスキー”、[桁指定] (バイト数設定) で“2” (バイト) を選択し、[OK] ボタンをクリックしま
 す。
 [RXD (ユーザー通信受信)] ダイアログボックスに戻ります。



8. [挿入] ボタンをクリックします。[データタイプ選択] ダイアログボックスを表示します。
[タイプ] で“定数 (16進数)” を選択し、[OK] ボタンをクリックします。[定数 (16進数)] ダイアログボックスが表示されます。エンドデリミタのETX (03h) を設定します。[16進数] で「03」を入力し、[OK] ボタンをクリックします。
[RXD (ユーザー通信受信)] ダイアログボックスに戻ります。



9. [D1] (デスティネーション 1) で受信完了出力先「M0010」、[D2] (デスティネーション 2) で受信動作ステータス「D0100」を入力します。
[OK] ボタンをクリックします。



以上で、RXD 命令の設定が完了し、次の受信データがデータレジスタに格納されます。

D0010 : 1234h = 4660

D0011 : 0005h = 5

10. ユーザープログラムを FC6A 形にダウンロードします。

以上で、設定は完了です。

制御ラインコントロール

ユーザー通信命令で制御ラインコントロールが必要な場合に、特殊データレジスタに値を設定することでポート 1～33 の DR 信号および ER 信号をコントロールできます。DR 信号は外部機器から FC6A 形への入力信号、ER 信号は FC6A 形から外部機器への出力信号です。



- RS485 通信では制御ラインコントロールを使用できません。
- RS232C 通信カートリッジの RS 信号は常時 ON の出力信号です。

特殊データレジスタ割付

特殊データレジスタ	機能	設定のタイミング	読み出し / 書き込み
D8104	制御ライン状態 (ポート1~5)	毎スキャン	R
D8105	DR制御ラインコントロール (ポート1~5)	毎スキャン	R/W
D8106	ER制御ラインコントロール (ポート1~5)	毎スキャン	R/W
D8204	制御ライン状態 (ポート6~9)	毎スキャン	R
D8205	DR制御ラインコントロール (ポート6~9)	毎スキャン	R/W
D8206	ER制御ラインコントロール (ポート6~9)	毎スキャン	R/W
D8717	制御ライン状態 (ポート10~13)	毎スキャン	R
D8718	DR制御ラインコントロール (ポート10~13)	毎スキャン	R/W
D8719	ER制御ラインコントロール (ポート10~13)	毎スキャン	R/W
D8720	制御ライン状態 (ポート14~17)	毎スキャン	R
D8721	DR制御ラインコントロール (ポート14~17)	毎スキャン	R/W
D8722	ER制御ラインコントロール (ポート14~17)	毎スキャン	R/W
D8723	制御ライン状態 (ポート18~21)	毎スキャン	R
D8724	DR制御ラインコントロール (ポート18~21)	毎スキャン	R/W
D8725	ER制御ラインコントロール (ポート18~21)	毎スキャン	R/W
D8726	制御ライン状態 (ポート22~25)	毎スキャン	R
D8727	DR制御ラインコントロール (ポート22~25)	毎スキャン	R/W
D8728	ER制御ラインコントロール (ポート22~25)	毎スキャン	R/W
D8729	制御ライン状態 (ポート26~29)	毎スキャン	R
D8730	DR制御ラインコントロール (ポート26~29)	毎スキャン	R/W
D8731	ER制御ラインコントロール (ポート26~29)	毎スキャン	R/W
D8732	制御ライン状態 (ポート30~33)	毎スキャン	R
D8733	DR制御ラインコントロール (ポート30~33)	毎スキャン	R/W
D8734	ER制御ラインコントロール (ポート30~33)	毎スキャン	R/W



R/W は、Read (読み出し) / Write (書き込み) の略で、R/W の場合は読み出し・書き込み可能、R の場合は読み出しのみ可能、W の場合は書き込みのみ可能です。

制御ラインコントロール一覧 (RUN 中)

DR 制御ラインコントロール (入力)		ER 制御ラインコントロール (出力)	
設定値	内容	設定値	内容
0	DR信号を処理しない	0	ON
1	ON : 送受信可 OFF : 送受信不可	1	OFF
2	ON : 送受信不可 OFF : 送受信可	2	ON : 受信可 OFF : 受信不可
3	ON : 送信可 OFF : 送信不可	3	ON
4	ON : 送信不可 OFF : 送信可	4	ON
5 以上	DR信号を処理しない	5 以上	ON

* ユーザー通信時のみ有効です。メンテナンス通信時は、DR 信号は処理せず、ER 信号は常に ON となります。

制御ラインコントロール一覧 (STOP 中)

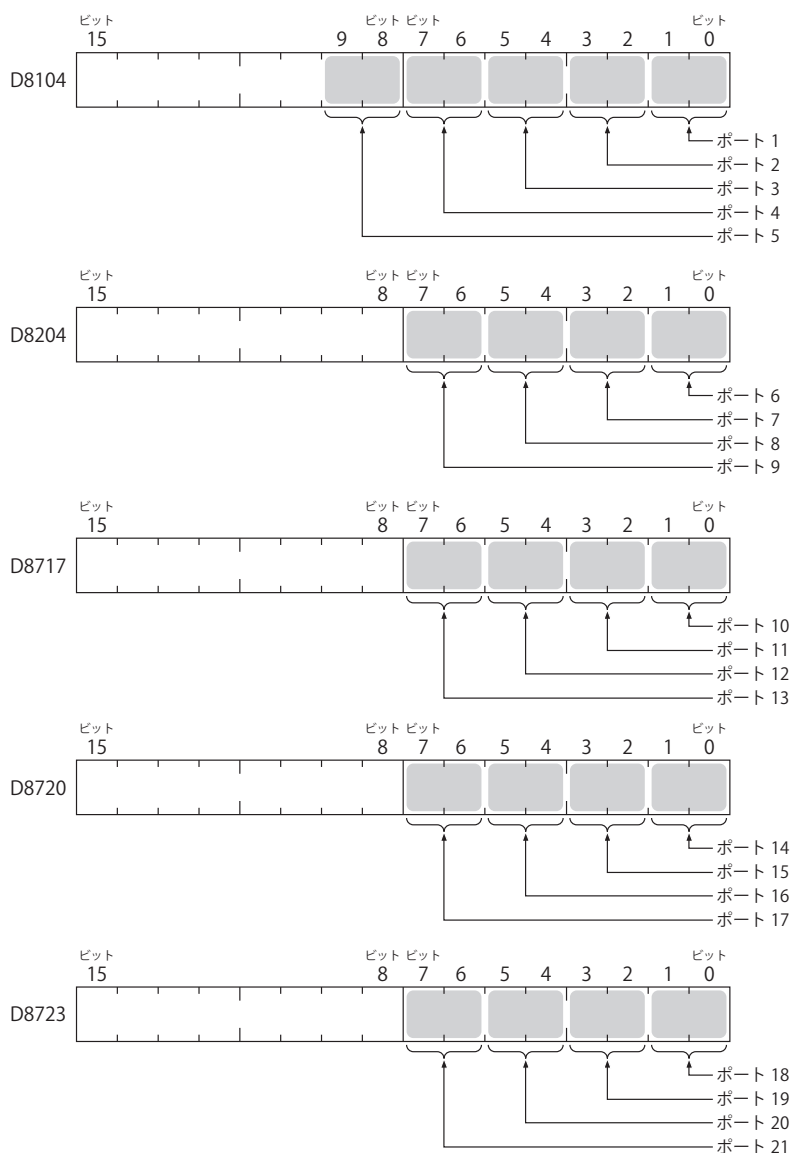
DR 制御ラインコントロール (入力)		ER 制御ラインコントロール (出力)	
設定値	内容	設定値	内容
0	DR信号を処理しない (送受信不可)	0	OFF
1	DR信号を処理しない (送受信不可)	1	OFF
2	DR信号を処理しない (送受信不可)	2	OFF
3	DR信号を処理しない (送受信不可)	3	OFF
4	DR信号を処理しない (送受信不可)	4	OFF
5以上	DR信号を処理しない (送受信不可)	5以上	OFF

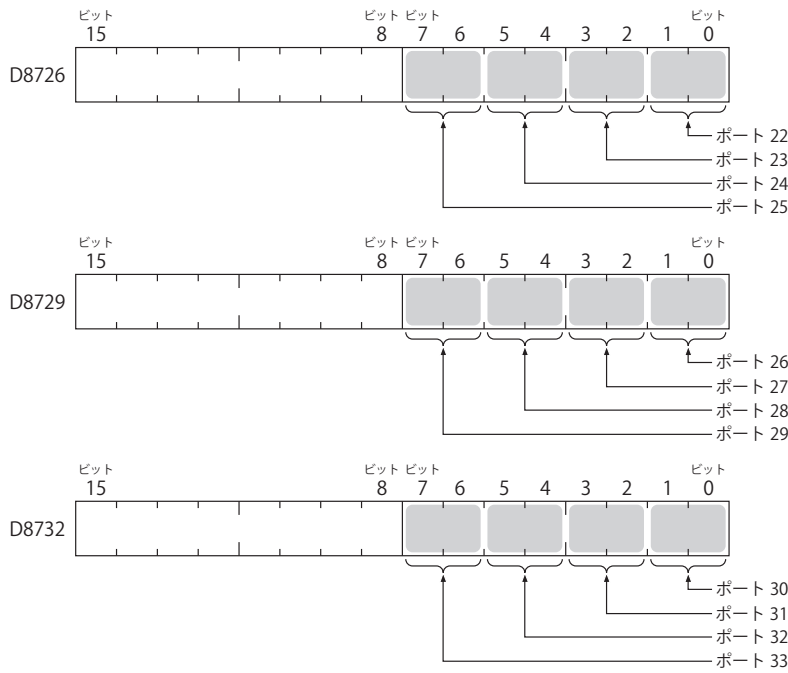
* ユーザー通信時のみ有効です。メンテナンス通信時は、DR信号は処理せず、ER信号は常にONとなります

制御ライン状態

制御ライン (DR、ER) の信号状態を D8104、D8204、D8717、D8720、D8723、D8726、D8729、D8732 に格納します。DR信号およびER信号の状態は通信ポートに対応する上記デバイスに格納されます。デバイスの値はSTOP中、RUN中のEND処理で更新します。

デバイス内の各通信ポートの割り当て (ビットアサイン) は、次のとおりです。





ポート 1 ～ 33 に割り当てられた領域に格納される値の意味は次のとおりです。

設定値*1	状態
0 (00)	DR信号とER信号がOFFです。
1 (01)	DR信号がONです。
2 (10)	ER信号がONです。
3 (11)	DR信号とER信号がONです。

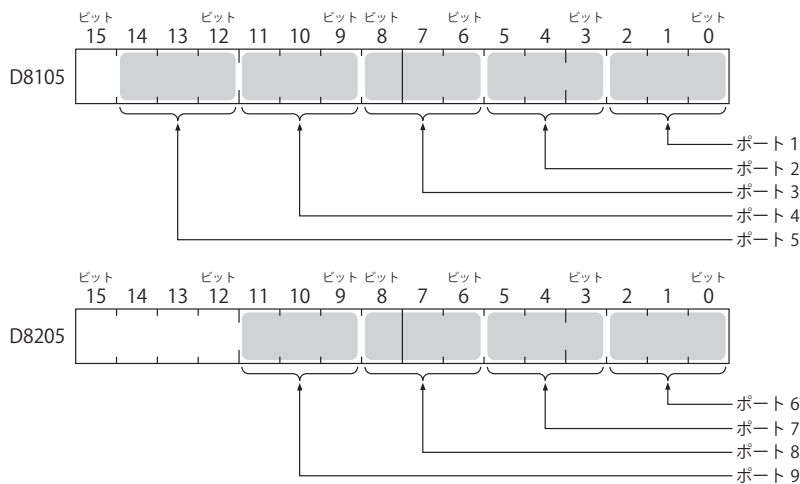
*1 () 内は2進数値

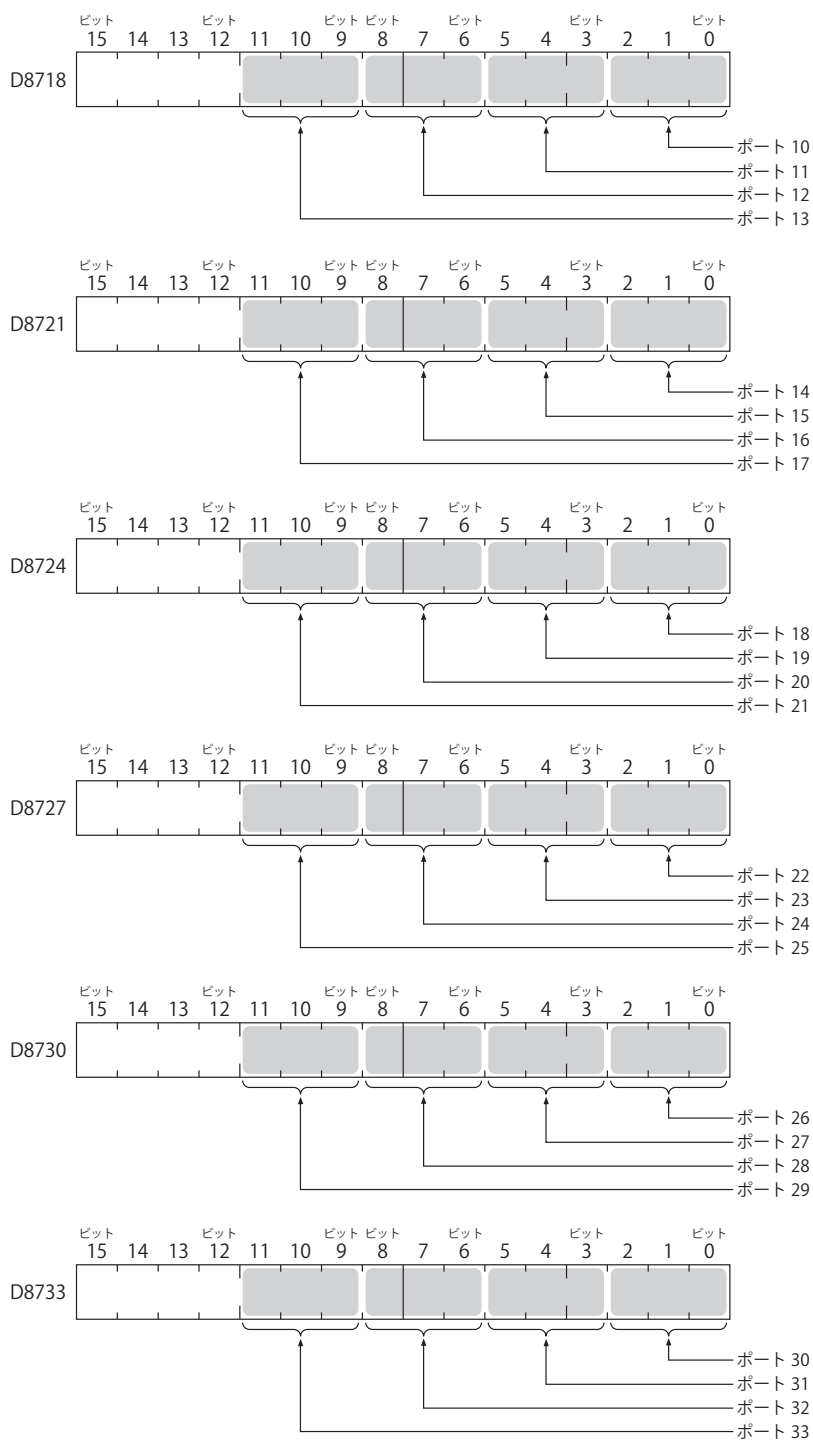
DR 制御ラインコントロール

外部機器は、DR 信号を使用して外部機器が受信可能か、または有効なデータを送信しているかなどの状態を FC6A 形に伝えます。FC6A 形は、外部機器の DR 信号の状態によって送受信を決定します。

D8105、D8205、D8718、D8721、D8724、D8727、D8730、D8733 に値を設定することで DR 信号をコントロールできます。ユーザー通信時のみ有効です。

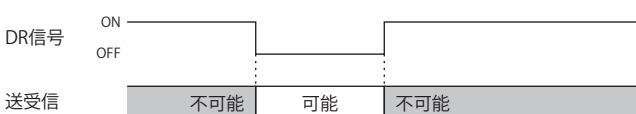
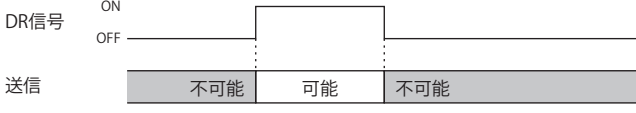
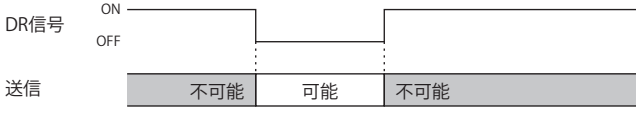
デバイス内の各通信ポートの割り当て（ビットアサイン）は次のとおりです。





ポート 1 ~ 33 に割り当てられた領域に格納される値の意味は次のとおりです。

設定値 *1	説明
0 (000)	FC6A形の送受信制御に、DR信号の状態を使用しません。DR信号制御を行う必要がなければ、この状態でご使用ください。
1 (001)	DR信号がONの場合に、FC6A形が送受信可能になります。 <div style="display: flex; align-items: center; margin-top: 5px;"> <div style="margin-right: 10px;">DR信号</div> <div style="margin-right: 10px;">ON</div> </div> <div style="display: flex; align-items: center; margin-top: 5px;"> <div style="margin-right: 10px;">送受信</div> <div style="display: flex; border: 1px solid black; background-color: #cccccc;"> <div style="width: 33%; text-align: center;">不可能</div> <div style="width: 33%; text-align: center;">可能</div> <div style="width: 33%; text-align: center;">不可能</div> </div> </div>

設定値*1	説明
2 (010)	DR信号がOFFの場合に、FC6A形が送受信可能になります。 
3 (011)	DR信号がONの場合に、FC6A形が送信可能になります。これは通常「Busy制御」と呼ばれ、プリンタなどの処理速度が遅い機器への送信制御に使用します（外部機器から見れば、入力データの制限となります）。受信は常に可能です。 
4 (100)	DR信号がOFFの場合に、FC6A形が送信可能になります。受信は常に可能です。 
5 以上	設定値“0”と同じ動作をします。

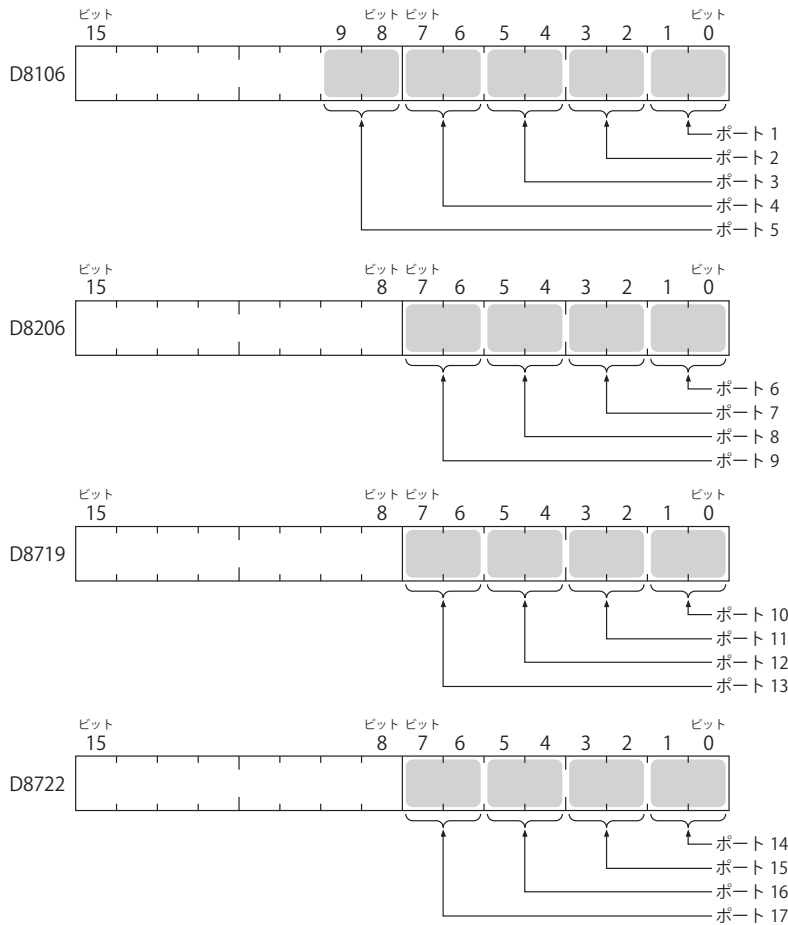
*1 () 内は2進数値

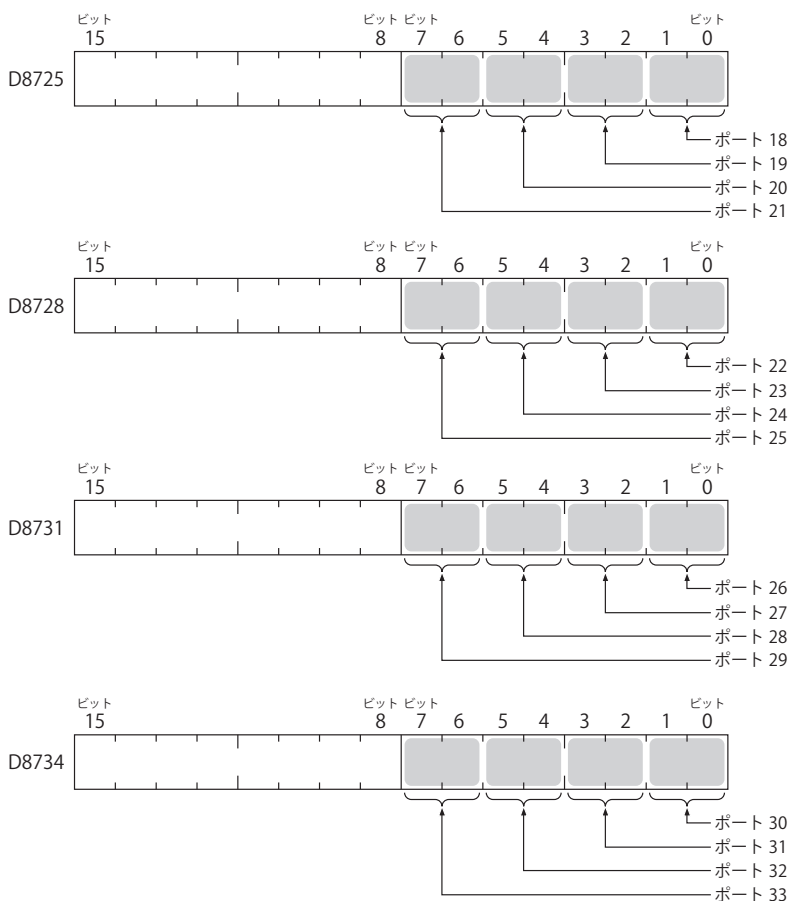
ER 制御ラインコントロール

FC6A 形は、ER 信号を使用して FC6A 形のコントロール状態や送受信状態を外部機器に伝えます。

D8106、D8206、D8719、D8722、D8725、D8728、D8731、D8734 に値を設定することでER 信号をコントロールできます。ユーザー通信時のみ有効です。

デバイス内の各通信ポートの割り当て（ビットアサイン）は次のとおりです。





ポート 1～33 に割り当てられた領域に格納される値の意味は次のとおりです。

設定値*1	説明						
0 (00)	<p>FC6A形がRUN中はON、STOP中はOFFになります。 RUN中はデータの送受信に関わらず常にONです。FC6A形のRUN/STOP状態の表示が必要な場合に設定します。</p> <p>FC6A形マイクロスマートの RUN/STOP状態</p> <table border="1"> <tr> <td>RUN中</td> <td>STOP中</td> <td>RUN中</td> </tr> </table> <p>ER信号</p> <table border="1"> <tr> <td>ON</td> <td>OFF</td> <td>ON</td> </tr> </table>	RUN中	STOP中	RUN中	ON	OFF	ON
RUN中	STOP中	RUN中					
ON	OFF	ON					
1 (01)	常時OFFになります。						
2 (10)	<p>受信データをフロー制御したい場合に設定します。FC6A形が外部機器からのデータを受信できる場合、ER信号がONになります。受信できない場合はER信号がOFFになります。</p> <p>受信</p> <table border="1"> <tr> <td>不可能</td> <td>可能</td> <td>不可能</td> </tr> </table> <p>ER信号</p> <table border="1"> <tr> <td>ON</td> <td>OFF</td> <td>ON</td> </tr> </table>	不可能	可能	不可能	ON	OFF	ON
不可能	可能	不可能					
ON	OFF	ON					
3 (11)	設定値“0”と同じ動作をします。						

*1 () 内は2進数値

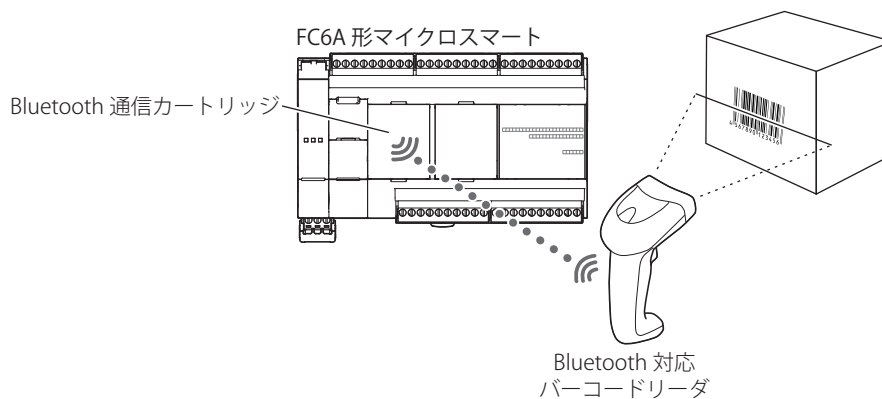
シリアル通信 (Bluetooth) でのユーザー通信

All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールのカートリッジスロット、Plus CPU モジュールに接続したカートリッジベースモジュールまたは HMI モジュールのカートリッジスロットに Bluetooth 通信カートリッジを接続することで、Bluetooth 通信でのユーザー通信を使用できます。

Bluetooth 通信でのユーザー通信により、TXD (ユーザー通信送信) 命令および RXD (ユーザー通信受信) 命令を使用することで、Bluetooth 接続したバーコードリーダーなどの外部機器とデータの送受信ができます。

バーコードリーダーで読み取ったデータを受信する例

FC6A 形のカートリッジスロット 1 に Bluetooth 通信カートリッジを接続し、バーコードリーダーと通信します。

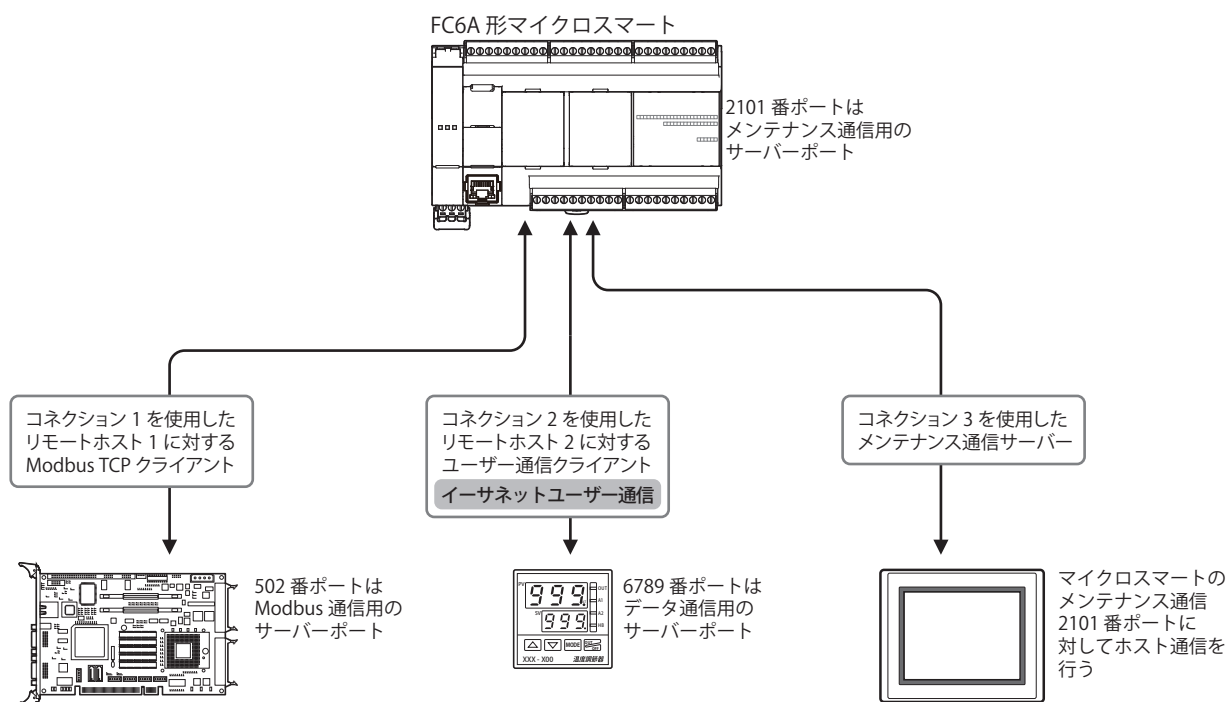


通信仕様および WindLDR の設定については、「第 9 章 Bluetooth 通信」(9-1 頁) を参照してください。

イーサネット通信でのユーザー通信

FC6A形は、TCP プロトコルおよび UDP プロトコルに対応しています。
 ETXD（イーサネットユーザー通信送信）命令および ERXD（イーサネットユーザー通信受信）命令を使用することで、ネットワーク上の機器とのデータの送受信ができます。
 FC6A形は、イーサネットユーザー通信（TCP）のクライアントおよびサーバーとして使用できます。Plus CPU モジュールはユーザー通信（UDP）としても使用できます。
 All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールが持つ 8 個、Plus CPU モジュールが持つ 16 個のコネクションには、それぞれ異なる通信を割り当てることができます。

3つのコネクションを使用したイーサネット通信の例



FC6A形のファンクション設定のコネクション設定

コネクション	通信プロトコル	その他設定
1	Modbus TCPクライアント	通信相手：リモートホスト1
2	ユーザー通信クライアント	通信相手：リモートホスト2
3	メンテナンス通信サーバー	ポート番号：2101

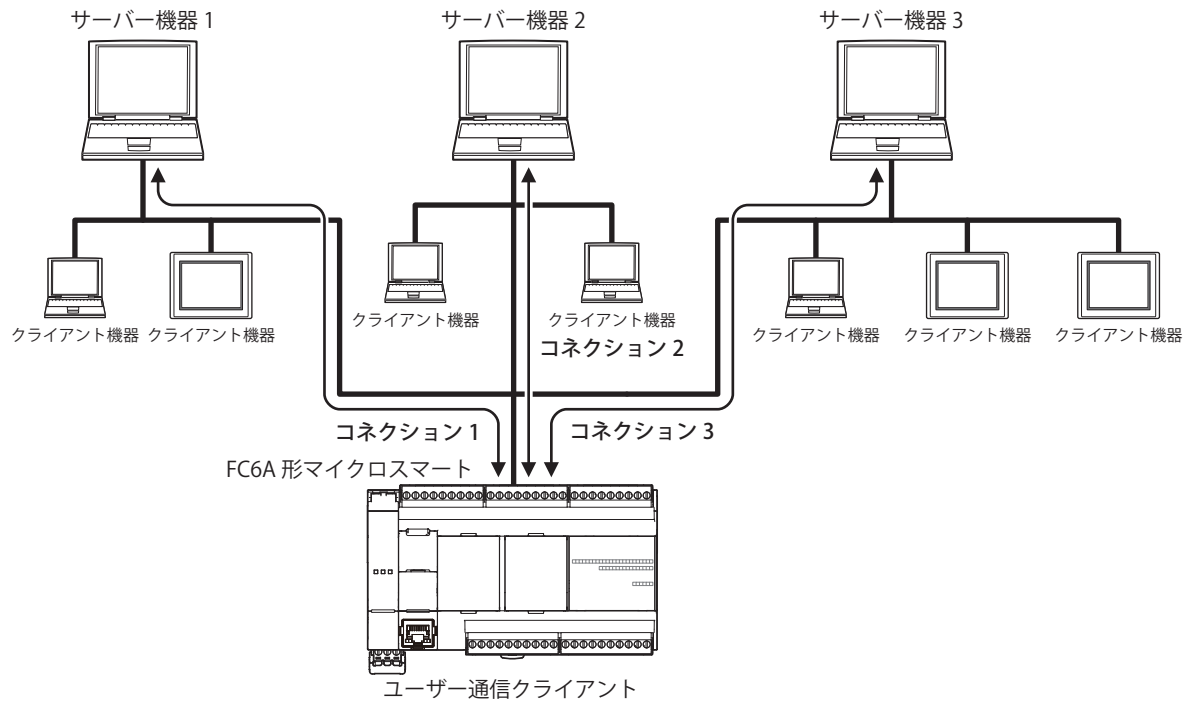
リモートホストテーブル

リモートホスト番号	IPアドレス	ポート番号
1	192.168.0.12	502
2	192.168.0.13	6789

FC6A 形をユーザー通信クライアントとして使用する場合

サーバー機器にネットワーク経由で FC6A 形を接続し、イーサネットユーザー通信命令を使用してサーバー機器と通信します。All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールは最大 8 個、Plus CPU モジュールは最大 16 個のコネクションをユーザー通信クライアントに割り当てることで、異なるサーバー機器と同時に接続して通信できます。

3 つのコネクションをユーザー通信クライアントに割り当てた場合

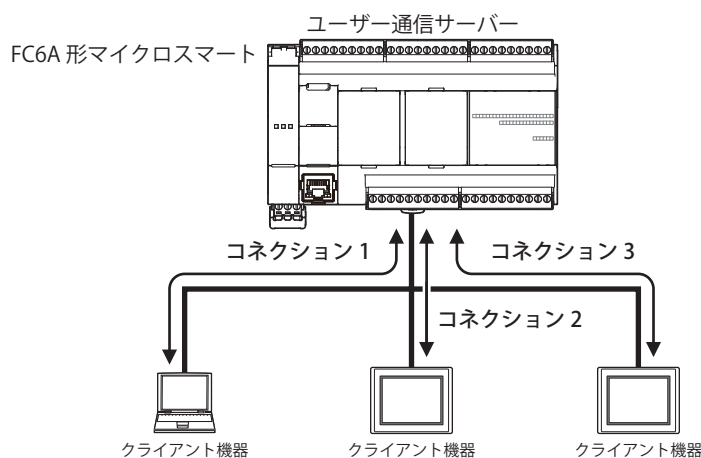


■ 設定の流れ

- ① ユーザー通信クライアントを設定する
「WindLDR の設定」(5-44 頁)
- ② イーサネットユーザー通信命令を使用したユーザープログラムを作成して、FC6A 形にダウンロードする
「イーサネットユーザー通信命令の設定手順」(5-49 頁)

FC6A 形をユーザー通信サーバーとして使用する場合

FC6A 形にクライアント機器を接続し、イーサネットユーザー通信命令を使用してクライアント機器と通信します。All-in-One CPU モジュール /CAN J1939 All-in-One CPU モジュールは最大 8 個、Plus CPU モジュールは最大 16 個のコネクションをユーザー通信サーバーに割り当てること、それぞれ最大 8 台、最大 16 台のクライアント機器を同時に接続して通信できます。3 つのコネクションをユーザー通信サーバーに割り当てた場合

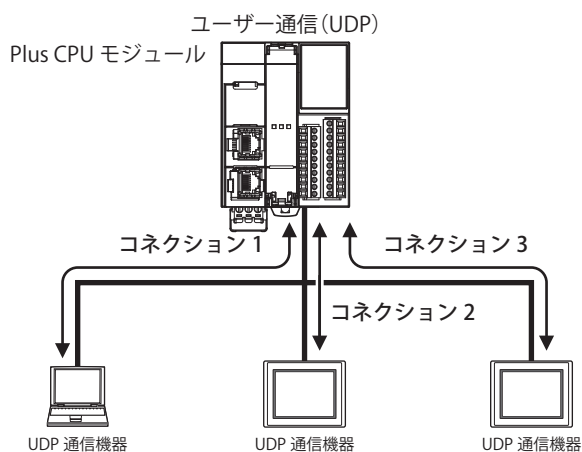


■ 設定の流れ

- ① ユーザー通信サーバーを設定する
「WindLDR の設定」(5-46 頁)
- ② イーサネットユーザー通信命令を使用したユーザープログラムを作成して、FC6A 形にダウンロードする
「イーサネットユーザー通信命令の設定手順」(5-49 頁)

FC6A 形をユーザー通信 (UDP) として使用する場合

Plus CPU モジュールに UDP 通信機器を接続し、イーサネットユーザー通信命令を使用して UDP 通信機器と通信します。Plus CPU モジュールは、最大 16 個のコネクションをユーザー通信 (UDP) に割り当てること、Plus CPU モジュールに最大 16 台の UDP 通信機器を同時に接続して通信できます。3 つのコネクションをユーザー通信 (UDP) に割り当てた場合



■ 設定の流れ

- ① ユーザー通信サーバーを設定する
「WindLDR の設定」(5-48 頁)
- ② イーサネットユーザー通信命令を使用したユーザープログラムを作成して、Plus CPU モジュールにダウンロードする
「イーサネットユーザー通信命令の設定手順」(5-49 頁)

ユーザー通信クライアント

FC6A形は、ユーザー通信クライアントにより、該当する接続番号が設定された ETXD（イーサネットユーザー通信送信）命令および ERXD（イーサネットユーザー通信受信）命令にしたがって、サーバー機器と通信します。ユーザー通信クライアントのリモートホスト番号の設定や通信設定は、[設定] タブの [ファンクション設定] で [接続設定] を設定します。

仕様

項目	ユーザー通信クライアント
リモートホスト番号	1～255 特殊データレジスタ（D8268～D8275、D8774～D8781）で切り替え
接続の確立	・ETXD命令、ERXD命令実行時 ・FC6A形のSTOP→RUN時 ^{*1}
接続の切断	・FC6A形のRUN→STOP時 ・特殊内部リレー（M8222～M8231、M8355～M8364）ON時
同時接続台数	1（1接続に接続できるサーバー機器 ^{*2} ）
受信タイムアウト時間	1000ms（100～25500ms（100ms単位）） ^{*3}

*1 [ファンクション設定] の [接続設定] で有効・無効を選択できます。

*2 All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールは最大 8 個、Plus CPU モジュールは最大 16 個の接続をユーザー通信クライアントに割り当てることで、それぞれ最大 8 台、最大 16 台の異なるサーバー機器と同時に通信できます。

*3 () 外の数値はデフォルト値です。

リモートホスト番号の切り替え

ユーザー通信クライアントを設定している場合、特殊データレジスタ（D8268～D8275、D8774～D8781）を使用し、通信相手を変更できます。特殊データレジスタには、切り替えたい通信相手のリモートホスト番号を指定します。1～255 以外を指定した場合は、ファンクション設定で設定したリモートホスト番号になります。

特殊データレジスタ	内容
D8268	接続1のリモートホスト番号（1～255）
D8269	接続2のリモートホスト番号（1～255）
D8270	接続3のリモートホスト番号（1～255）
D8271	接続4のリモートホスト番号（1～255）
D8272	接続5のリモートホスト番号（1～255）
D8273	接続6のリモートホスト番号（1～255）
D8274	接続7のリモートホスト番号（1～255）
D8275	接続8のリモートホスト番号（1～255）
D8774 ^{*1}	接続9のリモートホスト番号（1～255）
D8775 ^{*1}	接続10のリモートホスト番号（1～255）
D8776 ^{*1}	接続11のリモートホスト番号（1～255）
D8777 ^{*1}	接続12のリモートホスト番号（1～255）
D8778 ^{*1}	接続13のリモートホスト番号（1～255）
D8779 ^{*1}	接続14のリモートホスト番号（1～255）
D8780 ^{*1}	接続15のリモートホスト番号（1～255）
D8781 ^{*1}	接続16のリモートホスト番号（1～255）

*1 Plus CPU モジュールのみ

コネクションの確立 / 切断

ユーザー通信クライアントは、TCP プロトコルにしたがってコネクションを確立します。ETXD 命令および ERXD 命令実行時および FC6A 形を STOP → RUN したとき（前項*1）にコネクションを確立します。コネクション確立後、FC6A 形を RUN → STOP するか、特殊内部リレー（M8222～M8231、M8355～M8364）を ON することで、コネクションを切断します。

特殊内部リレー	内容	動作
M8222	コネクション1切断	OFFからONすると、該当のクライアントコネクションを切断します。 ONのとき、コネクションを確立できません。
M8223	コネクション2切断	
M8224	コネクション3切断	
M8225	コネクション4切断	
M8226	コネクション5切断	
M8227	コネクション6切断	
M8230	コネクション7切断	
M8231	コネクション8切断	
M8355*2	コネクション9切断	
M8356*2	コネクション10切断	
M8357*2	コネクション11切断	
M8360*2	コネクション12切断	
M8361*2	コネクション13切断	
M8362*2	コネクション14切断	
M8363*2	コネクション15切断	
M8364*2	コネクション16切断	

*1 [ファンクション設定] の [コネクション設定] で有効・無効を選択できます。

*2 Plus CPU モジュールのみ

ERXD（イーサネットユーザー通信受信）命令の受信キャンセルフラグの割り付け

各コネクションの受信キャンセルフラグの割り付けを示します。

ユーザー通信受信命令の受信前処理が既に完了し、受信中（ステータスコード 32）となっている状態でユーザー通信受信命令キャンセルフラグを ON すると、該当コネクションに対するすべての受信命令の実行をキャンセルします。受信データ待ちの状態が長く、受信命令の実行をキャンセルしたい場合に有効です。

キャンセルした受信命令をアクティブにする場合は、ユーザー通信受信キャンセルフラグを OFF したあと、受信命令の入力条件を再度 ON にしてください。

ユーザー通信受信命令キャンセルフラグは、各コネクションに特殊内部リレーとして次のように割り付けられます。

特殊内部リレー	内容
M8200	ユーザー通信受信命令キャンセルフラグ（コネクション1）
M8201	ユーザー通信受信命令キャンセルフラグ（コネクション2）
M8202	ユーザー通信受信命令キャンセルフラグ（コネクション3）
M8203	ユーザー通信受信命令キャンセルフラグ（コネクション4）
M8204	ユーザー通信受信命令キャンセルフラグ（コネクション5）
M8205	ユーザー通信受信命令キャンセルフラグ（コネクション6）
M8206	ユーザー通信受信命令キャンセルフラグ（コネクション7）
M8207	ユーザー通信受信命令キャンセルフラグ（コネクション8）
M8334*1	ユーザー通信受信命令キャンセルフラグ（コネクション9）
M8335*1	ユーザー通信受信命令キャンセルフラグ（コネクション10）
M8336*1	ユーザー通信受信命令キャンセルフラグ（コネクション11）
M8337*1	ユーザー通信受信命令キャンセルフラグ（コネクション12）
M8340*1	ユーザー通信受信命令キャンセルフラグ（コネクション13）
M8341*1	ユーザー通信受信命令キャンセルフラグ（コネクション14）
M8342*1	ユーザー通信受信命令キャンセルフラグ（コネクション15）
M8343*1	ユーザー通信受信命令キャンセルフラグ（コネクション16）

*1 Plus CPU モジュールのみ

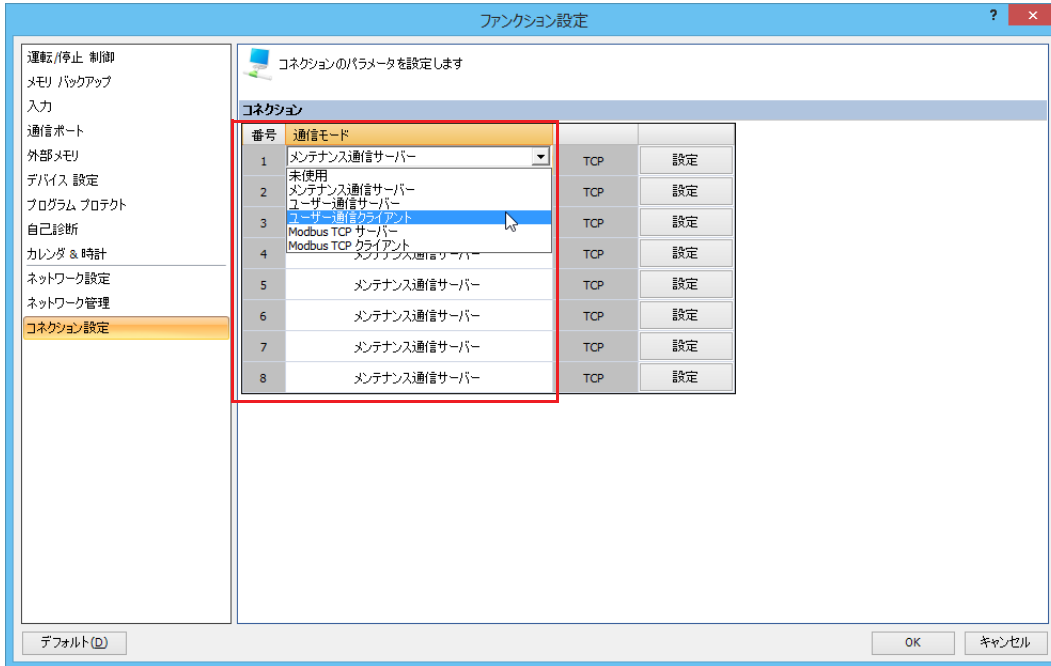
WindLDR の設定

FC6A 形をユーザー通信クライアントとして設定します。

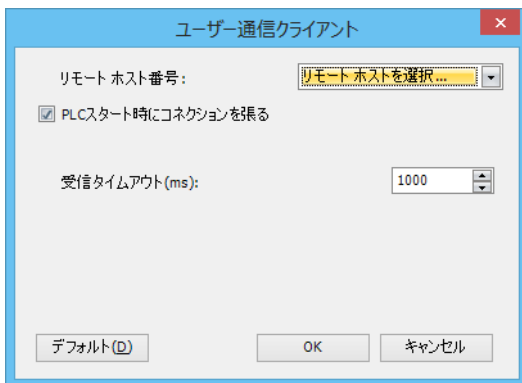
次の例では、WindLDR を使用してコネクション 1 にユーザー通信クライアントを設定する手順について説明します。

●操作手順

1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. コネクション番号 1 の [通信モード] をクリックし、“ユーザー通信クライアント” を選択します。
[ユーザー通信クライアント] ダイアログボックスが表示されます。



3. サーバー機器の仕様に合わせて、[リモートホスト番号]、[受信タイムアウト (ms)] を設定します。また、FC6A 形を STOP → RUN でコネクションを確立するは、[PLC スタート時にコネクションを張る] チェックボックスをオンにします。
リモートホスト番号については、「第 3 章 リモートホストリスト」(3-21 頁) を参照してください。



4. [OK] ボタンをクリックします。

以上で、ユーザー通信クライアントの設定は完了です。

次に、設定したコネクションに対応するユーザー通信命令を使用したユーザープログラムを作成して、FC6A 形にダウンロードします。詳細は、「イーサネットユーザー通信命令の設定手順」(5-49 頁) を参照してください。

ユーザー通信サーバー

FC6A形は、ユーザー通信サーバーにより、該当するコネクション番号が設定された ETXD（イーサネットユーザー通信送信）命令および ERXD（イーサネットユーザー通信受信）命令にしたがってクライアント機器と通信します。

ユーザー通信サーバーのポート番号の設定や通信設定は、[設定] タブの [ファンクション設定] で [コネクション設定] を設定します。

仕様

項目	ユーザー通信サーバー
ローカルホストポート番号	2102~2117 (0~65535の間で変更可能)
同時接続可能クライアント数	1 (1コネクションに接続できるクライアント機器 ^{*1})
受信タイムアウト時間	100ms~25500ms (100ms単位)

*1 All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールは最大 8 個、Plus CPU モジュールは最大 16 個のコネクションをユーザー通信サーバーに割り当てることができ、それぞれ最大 8 台、最大 16 台のクライアント機器を同時に FC6A 形に接続して通信できます。

ERXD（イーサネットユーザー通信受信）命令の受信キャンセルフラグの割り付け

各コネクションの受信キャンセルフラグの割り付けを示します。

ユーザー通信受信命令の受信前処理が既に完了し、受信中（ステータスコード 32）となっている状態でユーザー通信受信命令キャンセルフラグを ON すると、該当コネクションに対するすべての受信命令の実行をキャンセルします。受信データ待ちの状態が長く、受信命令の実行をキャンセルしたい場合に有効です。

キャンセルした受信命令をアクティブにする場合は、ユーザー通信受信キャンセルフラグを OFF したあと、受信命令の入力条件を再度 ON にしてください。

ユーザー通信受信命令キャンセルフラグは、各コネクションに特殊内部リレーとして次のように割り付けられます。

特殊内部リレー	内容
M8200	ユーザー通信受信命令キャンセルフラグ (コネクション1)
M8201	ユーザー通信受信命令キャンセルフラグ (コネクション2)
M8202	ユーザー通信受信命令キャンセルフラグ (コネクション3)
M8203	ユーザー通信受信命令キャンセルフラグ (コネクション4)
M8204	ユーザー通信受信命令キャンセルフラグ (コネクション5)
M8205	ユーザー通信受信命令キャンセルフラグ (コネクション6)
M8206	ユーザー通信受信命令キャンセルフラグ (コネクション7)
M8207	ユーザー通信受信命令キャンセルフラグ (コネクション8)
M8334 ^{*1}	ユーザー通信受信命令キャンセルフラグ (コネクション9)
M8335 ^{*1}	ユーザー通信受信命令キャンセルフラグ (コネクション10)
M8336 ^{*1}	ユーザー通信受信命令キャンセルフラグ (コネクション11)
M8337 ^{*1}	ユーザー通信受信命令キャンセルフラグ (コネクション12)
M8340 ^{*1}	ユーザー通信受信命令キャンセルフラグ (コネクション13)
M8341 ^{*1}	ユーザー通信受信命令キャンセルフラグ (コネクション14)
M8342 ^{*1}	ユーザー通信受信命令キャンセルフラグ (コネクション15)
M8343 ^{*1}	ユーザー通信受信命令キャンセルフラグ (コネクション16)

*1 Plus CPU モジュールのみ

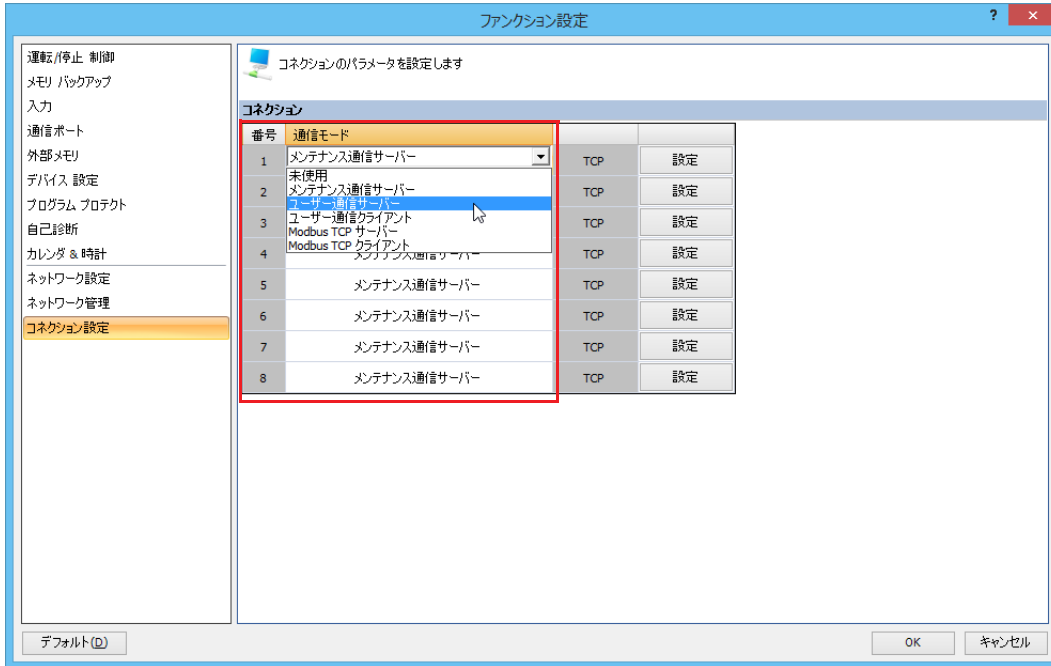
WindLDR の設定

FC6A 形をユーザー通信サーバーとして設定します。

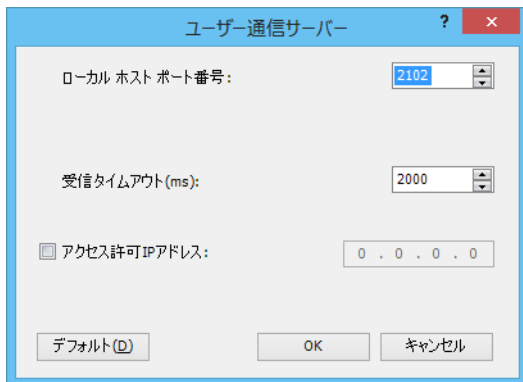
次の例では、WindLDR を使用してコネクション 1 にユーザー通信サーバーを設定する手順について説明します。

●操作手順

1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. コネクション番号 1 の [通信モード] をクリックし、“ユーザー通信サーバー”を選択します。
[ユーザー通信サーバー] ダイアログボックスが表示されます。



3. [ローカルホストポート番号]、[受信タイムアウト (ms)] を設定します。IP アドレスによるアクセス制限を有効にする場合は、[アクセス許可 IP アドレス] チェックボックスをオンにし、アクセス許可する IP アドレスを設定します。



4. [OK] ボタンをクリックします。

以上で、ユーザー通信サーバーの設定は完了です。

次に、設定したコネクションに対応するユーザー通信命令を使用したユーザープログラムを作成して、FC6A 形にダウンロードします。詳細は、「イーサネットユーザー通信命令の設定手順」(5-49 頁)を参照してください。

ユーザー通信 (UDP)

Plus CPU モジュールは、ユーザー通信サーバーにより、該当するコネクション番号が設定された ETXD (イーサネットユーザー通信送信) 命令および ERXD (イーサネットユーザー通信受信) 命令にしたがって UDP 通信機器と通信します。

ユーザー通信サーバーのポート番号の設定や通信設定は、[設定] タブの [ファンクション設定] で [コネクション設定] を設定します。

仕様

項目	ユーザー通信サーバー
リモートホスト番号	1 ~ 255
ローカルホストポート番号	0 ~ 65535
アクセス許可IPアドレス	0.0.0.0 ~ 255.255.255.255

ERXD (イーサネットユーザー通信受信) 命令の受信キャンセルフラグの割り付け

各コネクションの受信キャンセルフラグの割り付けを示します。

ユーザー通信受信命令の受信前処理が既に完了し、受信中 (ステータスコード 32) となっている状態でユーザー通信受信命令キャンセルフラグを ON すると、該当コネクションに対するすべての受信命令の実行をキャンセルします。受信データ待ちの状態が長く、受信命令の実行をキャンセルしたい場合に有効です。

キャンセルした受信命令をアクティブにする場合は、ユーザー通信受信キャンセルフラグを OFF したあと、受信命令の入力条件を再度 ON にしてください。

ユーザー通信受信命令キャンセルフラグは、各コネクションに特殊内部リレーとして次のように割り付けられます。

特殊内部リレー	内容
M8200	ユーザー通信受信命令キャンセルフラグ (コネクション1)
M8201	ユーザー通信受信命令キャンセルフラグ (コネクション2)
M8202	ユーザー通信受信命令キャンセルフラグ (コネクション3)
M8203	ユーザー通信受信命令キャンセルフラグ (コネクション4)
M8204	ユーザー通信受信命令キャンセルフラグ (コネクション5)
M8205	ユーザー通信受信命令キャンセルフラグ (コネクション6)
M8206	ユーザー通信受信命令キャンセルフラグ (コネクション7)
M8207	ユーザー通信受信命令キャンセルフラグ (コネクション8)
M8334	ユーザー通信受信命令キャンセルフラグ (コネクション9)
M8335	ユーザー通信受信命令キャンセルフラグ (コネクション10)
M8336	ユーザー通信受信命令キャンセルフラグ (コネクション11)
M8337	ユーザー通信受信命令キャンセルフラグ (コネクション12)
M8340	ユーザー通信受信命令キャンセルフラグ (コネクション13)
M8341	ユーザー通信受信命令キャンセルフラグ (コネクション14)
M8342	ユーザー通信受信命令キャンセルフラグ (コネクション15)
M8343	ユーザー通信受信命令キャンセルフラグ (コネクション16)

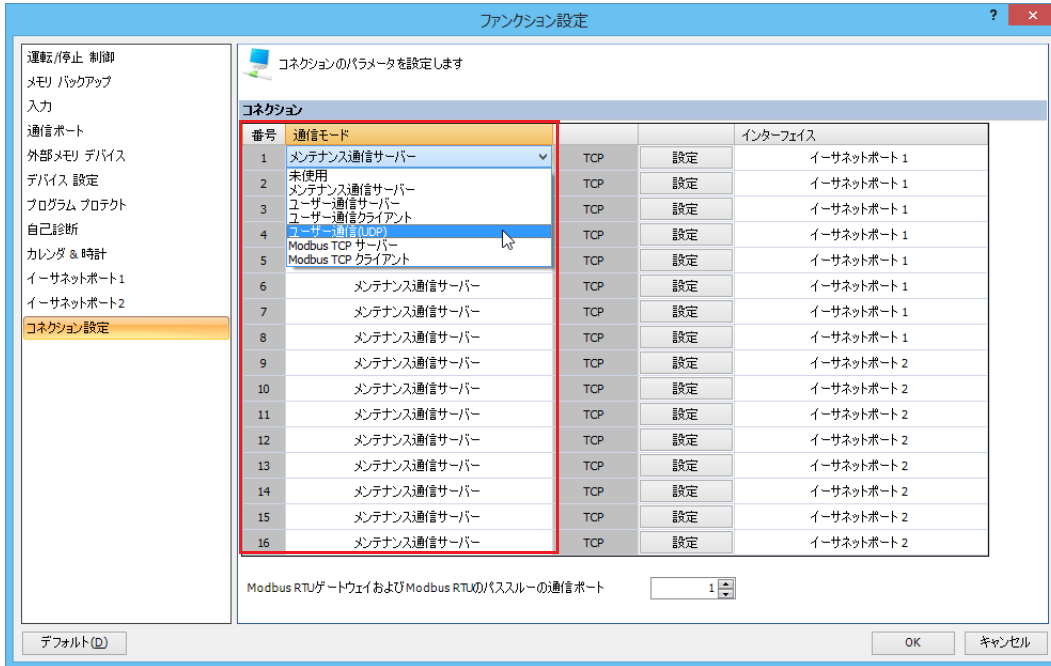
WindLDR の設定

Plus CPU モジュールをユーザー通信（UDP）として設定します。

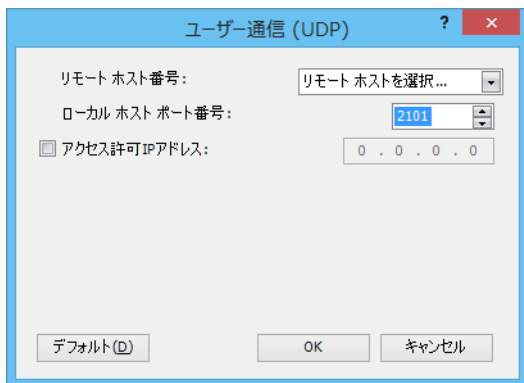
次の例では、WindLDR を使用してコネクション 1 にユーザー通信（UDP）を設定する手順について説明します。

●操作手順

1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. コネクション番号 1 の [通信モード] をクリックし、“ユーザー通信（UDP）” を選択します。
[ユーザー通信（UDP）] ダイアログボックスが表示されます。



3. [リモートホスト番号]、[ローカルホストポート番号] を設定します。IP アドレスによるアクセス制限を有効にする場合は、[アクセス許可 IP アドレス] チェックボックスをオンにし、アクセス許可する IP アドレスを設定します。
リモートホスト番号については、「第 3 章 リモートホストリスト」(3-21 頁) を参照してください。



4. [OK] ボタンをクリックします。

以上で、ユーザー通信（UDP）の設定は完了です。

次に、設定したコネクションに対応するユーザー通信命令を使用したユーザープログラムを作成して、Plus CPU モジュールにダウンロードします。詳細は、「イーサネットユーザー通信命令の設定手順」(5-49 頁) を参照してください。

イーサネットユーザー通信命令の設定手順

イーサネット通信でのユーザー通信は、ETXD（イーサネットユーザー通信送信）命令および ERXD（イーサネットユーザー通信受信）命令を使用します。

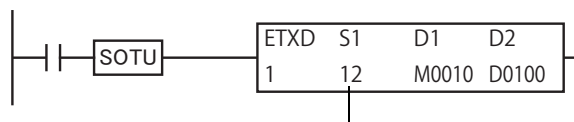
ETXD（イーサネットユーザー通信送信）命令の場合

送信データが、スタートデリミタ、送信するデータを格納するデータレジスタ、BCC、エンドデリミタを含む場合を例として、WindLDR で ETXD 命令を設定する手順について説明します。



ETXD 命令については、「ETXD（イーサネットユーザー通信送信）」(5-22 頁) を参照してください。

ラダー図



ラダー図上では、ETXD 命令の S1 に設定した「送信データ」の総バイト数が表示されます。

ETXD（イーサネットユーザー通信送信）命令の設定内容

項目	設定内容		
ポート番号	コネクション1		
送信データ	BCC計算範囲		
	STX (02h)	"1" (31h)	"2" (32h)
	"3" (33h)	"4" (34h)	"5" (35h)
	"6" (36h)	"7" (37h)	"8" (38h)
	BCC (H) (41h)	BCC (L) (36h)	ETX (03h)
	定数 (16進数)	D0010	
	D0011		BCC
	定数 (16進数)		定数 (16進数)
	定数 (16進数)	スタートデリミタ (STX)	02h
	間接 (データレジスタ)	データレジスタ番号	D0010
変換タイプ		バイナリ→アスキー	
送信バイト数設定		4 (バイト)	
リピート回数		2 (回)	
BCC	計算開始位置	1 (バイト目)	
	計算方式	ADD	
	変換タイプ	バイナリ→アスキー	
	バイト数設定	2 (バイト)	
定数 (16進数)	エンドデリミタ (ETX)	03h	
送信完了出力	M0010		
送信動作ステータス	D0100		
送信バイト数	D0101		

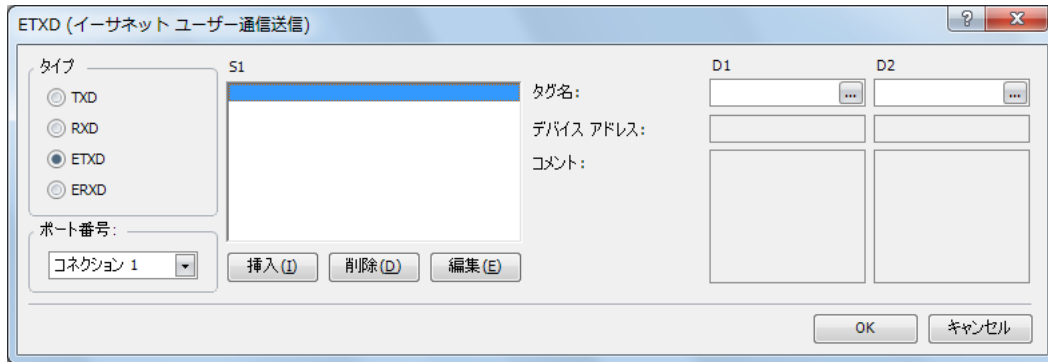
データレジスタの設定内容

デバイスアドレス	設定内容
D0010	1234 (04D2h)
D0011	5678 (162Eh)

WindLDR の設定

●操作手順

1. ETXD 命令を入力します。
ユーザープログラムの ETXD 命令を挿入する場所にカーソルを移動し、「ETXD」と入力します。[コイル選択] ダイアログボックスが表示されます。“ETXD”が選択されていることを確認し、[OK] ボタンをクリックします。
または、[ホーム] タブの [命令] で [演算] をクリックし、[イーサネット]、[ETXD (イーサネットユーザー通信送信)] の順にクリックします。ユーザープログラムの ETXD 命令を挿入する場所をクリックします。
[ETXD (イーサネットユーザー通信送信)] ダイアログボックスが表示されます。
2. [タイプ] で “ETXD” が選択されていることを確認し、[ポート番号] で “コネクション 1” を選択します。



3. [S1] (ソース 1)、[D1] (デスティネーション 1)、[D2] (デスティネーション 2) を設定します。
設定方法は、TXD 命令と同様です。詳細は、「ユーザー通信命令の設定」－「WindLDR の設定」(5-25 頁) を参照してください。

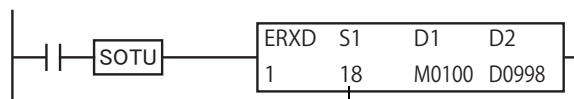
ERXD (イーサネットユーザー通信受信) 命令の場合

受信データが、スタートデリミタ、受信データ照合用の定数、受信したデータをデータ変換して格納するデータレジスタ、BCC、エンドデリミタを含む場合を例として、WindLDR で ERXD 命令を設定する手順について説明します。



ERXD 命令については、「ERXD (イーサネットユーザー通信受信)」(5-22 頁) を参照してください。

ラダー図



ラダー図上では、ERXD 命令の S1 に設定した「受信データ」の総バイト数が表示されます。

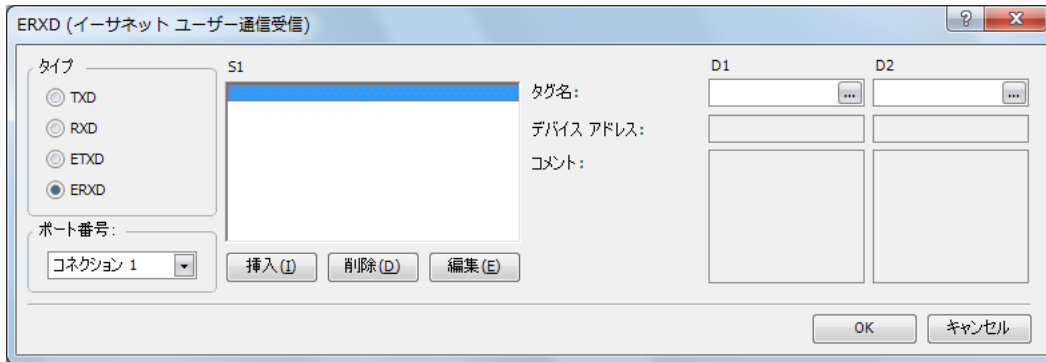
ERXD (イーサネットユーザー通信受信) 命令の設定内容

項目	設定内容																																														
ポート番号	コネクション1																																														
受信データ	<div style="text-align: center;"> <table border="1" style="margin: auto;"> <tr> <td colspan="15">BCC計算範囲</td> </tr> <tr> <td>STX (02h)</td> <td>局番 (H) (00h)</td> <td>局番 (L) (10h)</td> <td>"0" (30h)</td> <td>"0" (30h)</td> <td>" (2Ch)</td> <td>"1" (31h)</td> <td>"2" (32h)</td> <td>"3" (33h)</td> <td>"4" (34h)</td> <td>"5" (35h)</td> <td>CR (0Dh)</td> <td>BCC (H) (41h)</td> <td>BCC (L) (41h)</td> <td>ETX (03h)</td> </tr> <tr> <td colspan="3">スタートデリミタ</td> <td colspan="2">スキップ 範囲</td> <td colspan="2">照合用 定数</td> <td colspan="5">D0010~D0011</td> <td colspan="2">BCC</td> <td>エンド デリミタ</td> </tr> </table> </div>		BCC計算範囲															STX (02h)	局番 (H) (00h)	局番 (L) (10h)	"0" (30h)	"0" (30h)	" (2Ch)	"1" (31h)	"2" (32h)	"3" (33h)	"4" (34h)	"5" (35h)	CR (0Dh)	BCC (H) (41h)	BCC (L) (41h)	ETX (03h)	スタートデリミタ			スキップ 範囲		照合用 定数		D0010~D0011					BCC		エンド デリミタ
	BCC計算範囲																																														
	STX (02h)	局番 (H) (00h)	局番 (L) (10h)	"0" (30h)	"0" (30h)	" (2Ch)	"1" (31h)	"2" (32h)	"3" (33h)	"4" (34h)	"5" (35h)	CR (0Dh)	BCC (H) (41h)	BCC (L) (41h)	ETX (03h)																																
	スタートデリミタ			スキップ 範囲		照合用 定数		D0010~D0011					BCC		エンド デリミタ																																
	定数 (16進数)	スタートデリミタ (STX)	02h																																												
		スレーブ番号 (H)	00h																																												
		スレーブ番号 (L)	10h																																												
	スキップ	バイト数設定	2 (バイト)																																												
	定数 (文字)	受信データ照合用の定数	"																																												
	間接 (データレジスタ)	データレジスタ番号	D0010																																												
変換タイプ		アスキー→バイナリ																																													
受信バイト数設定		4 (バイト)																																													
リピート回数		2 (回)																																													
BCC	可変デリミタ	16進数、0Dh (CR)																																													
	計算開始位置	1 (バイト目)																																													
	計算方式	ADD																																													
	変換タイプ	バイナリ→アスキー																																													
定数 (16進数)	バイト数設定	2 (バイト)																																													
		03h (ETX)																																													
受信完了出力	M0100																																														
受信動作ステータス	D0998																																														
受信バイト数	D0999																																														

WindLDR の設定

●操作手順

1. ERXD 命令を入力します。
ユーザープログラムの ERXD 命令を挿入する場所にカーソルを移動し、「ERXD」と入力します。[コイル選択] ダイアログボックスが表示されます。“ERXD”が選択されていることを確認し、[OK] ボタンをクリックします。
または、[ホーム] タブの [命令] で [演算] をクリックし、[イーサネット] から [ERXD (イーサネットユーザー通信受信)] をクリックします。ユーザープログラムの ERXD 命令を挿入する場所をクリックします。
[ERXD (イーサネットユーザー通信受信)] ダイアログボックスが表示されます。
2. [タイプ] で “ERXD” が選択されていることを確認し、[ポート番号] で “コネクション 1” を選択します。



3. [S1] (ソース 1)、[D1] (デスティネーション 1)、[D2] (デスティネーション 2) のデバイスを設定します。
設定方法は、RXD 命令と同様です。詳細は、「ユーザー通信命令の設定」－「WindLDR の設定」(5-29 頁) を参照してください。

ユーザー通信送信命令・受信命令のエラー

ユーザー通信命令でエラーが発生すると、ユーザー通信の送信・受信動作ステータスとして設定したデータレジスタにエラーコードを格納します。複数のエラーが発生した場合は順次上書きするため、最後に発生したエラーコードが結果として格納されます。送信動作ステータス、または受信動作ステータスのデータレジスタには、ステータスコードとエラーコードの和を格納します。送信動作ステータス、または受信動作ステータスのデータレジスタの値を16で割ると、その余りがエラーコードとなります。

[受信動作ステータスのデータレジスタの値が“74”の場合]

$74 \div 16 = 4$ 余り 10

となり、エラーコードは“10”となります。

送信・受信動作ステータスコード

送信動作ステータスについては、「D2+0（送信動作ステータス）」(5-8頁)を参照してください。

受信動作ステータスについては、「D2+0（受信動作ステータス）」(5-15頁)を参照してください。

送信動作ステータスコード

コネクション	送信状態
16	送信前処理中
32	送信中
48	データ送信完了
64	送信命令完了

受信動作ステータスコード

ステータスコード	受信状態
16	受信前処理中
32	受信中
48	受信データ展開中
64	受信完了
128	受信命令キャンセル

エラーコード

発生したエラーによって通信（送信・受信）完了出力の動作が異なります。ユーザー通信命令のエラーコードに対応するエラー内容と通信（送信）完了出力の動作を参考にして、ユーザープログラムを修正してください。

エラーコード	エラー内容	通信（送信・受信）完了出力の動作
1	起動入力ONしている送信命令が5つを超えた。	アドレスの小さい側から5つ以内の送信完了出力はONする
2	送信先機器のビジー状態が一定時間を超えた。（ビジータイムオーバーエラー）	送信完了出力がONする
3	起動入力ONしているスタートデリミタを設定した受信命令が5つを超えた。	アドレスの小さい側から5つ以内で、スタートデリミタが受信データと一致した受信命令の受信完了出力はONする
4	スタートデリミタを設定した受信命令と設定していない受信命令を混在して同時に起動した。または、スタートデリミタを設定していない受信命令を2つ以上同時に起動した。	アドレスの小さい側の受信命令の受信完了出力がON
5	同じスタートデリミタの受信命令を同時に起動した。	出力は変化しない。
7	受信データが設定したスタートデリミタと一致しなかった。	出力は変化しない。 ただし、その後スタートデリミタを含む正常なデータを受信すると受信完了出力がONする
8	受信フォーマットでアスキーコードをバイナリまたはBCDに変換する設定があった場合に、データとして“0”～“9”または“A”～“F”以外のコードを受信した。（変換時には“0”として扱う）	受信完了出力がONする
9	受信命令で計算したBCCとデータに付加されて送られてきたBCCとが一致しなかった。	受信完了出力がONする
10	受信命令で設定したエンドデリミタまたは定数と、受信したエンドデリミタまたは定数が一致しなかった。	受信完了出力がONする
11	受信命令で1キャラクタ（1バイト）受信した後、通信フォーマットで設定されている受信タイムアウトを待っても次のデータが来なかった。	受信完了出力がONする
12	オーバーランエラーが発生（データ受信が終了するまでに次のデータを受信）した。	受信完了出力がOFFする
13	フレーミングエラー（スタート・ストップビットの検出誤り）が発生した。	出力は変化しない
14	パリティエラー（パリティでエラーを検出）が発生した。	出力は変化しない
15	ポート設定またはコネクション設定でユーザー通信が設定されていない状態で、ユーザー通信命令を使用した。または、通信カートリッジまたは通信モジュールが未接続の状態、ユーザー通信を設定したポートでユーザー通信命令を使用した。	出力は変化しない

ユーザー通信命令のキャラクタコード

上位 4ビット 下位 4ビット	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	DEL	SP	0	@	P	`	p				-	タ	ミ		
10進数	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
1	SOH	DC1	!	1	A	Q	a	q			。	ア	チ	ム		
10進数	1	17	33	49	65	81	97	113	129	145	161	177	193	209	225	241
2	STX	DC2	"	2	B	R	b	r			「	イ	ツ	メ		
10進数	2	18	34	50	66	82	98	114	130	146	162	178	194	210	226	242
3	ETX	DC3	#	3	C	S	c	s			」	ウ	テ	モ		
10進数	3	19	35	51	67	83	99	115	131	147	163	179	195	211	227	243
4	EOT	DC4	\$	4	D	T	d	t			、	エ	ト	ヤ		
10進数	4	20	36	52	68	84	100	116	132	148	164	180	196	212	228	244
5	ENQ	NAK	%	5	E	U	e	u			・	オ	ナ	ユ		
10進数	5	21	37	53	69	85	101	117	133	149	165	181	197	213	229	245
6	ACK	SYN	&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
10進数	6	22	38	54	70	86	102	118	134	150	166	182	198	214	230	246
7	BEL	ETB	'	7	G	W	g	w			ア	キ	ヌ	ウ		
10進数	7	23	39	55	71	87	103	119	135	151	167	183	199	215	231	247
8	BS	CAN	(8	H	X	h	x			イ	ク	ネ	リ		
10進数	8	24	40	56	72	88	104	120	136	152	168	184	200	216	232	248
9	HT	EM)	9	I	Y	i	y			ウ	ケ	ノ	ル		
10進数	9	25	41	57	73	89	105	121	137	153	169	185	201	217	233	249
A	LF	SUB	*	:	J	Z	j	z			エ	コ	ハ	レ		
10進数	10	26	42	58	74	90	106	122	138	154	170	186	202	218	234	250
B	VT	ESC	+	;	K	[k	{			オ	サ	ヒ	ロ		
10進数	11	27	43	59	75	91	107	123	139	155	171	187	203	219	235	251
C	FF	FS	,	<	L	¥	l				ヤ	シ	フ	ワ		
10進数	12	28	44	60	76	92	108	124	140	156	172	188	204	220	236	252
D	CR	GS	-	=	M]	m	}			ユ	ス	ハ	ン		
10進数	13	29	45	61	77	93	109	125	141	157	173	189	205	221	237	253
E	SO	RS	.	>	N	^	n	~			ヨ	セ	ホ	。		
10進数	14	30	46	62	78	94	110	126	142	158	174	190	206	222	238	254
F	SI	US	/	?	O	_	o				ッ	ソ	マ	°		
10進数	15	31	47	63	79	95	111	127	143	159	175	191	207	223	240	255

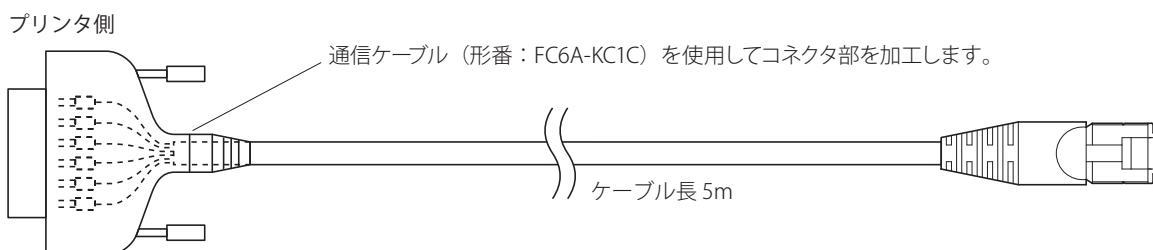
ユーザー通信を用いたプログラム例

シリアルインターフェイス (RS232C) を装備したプリンタに対して、FC6A 形を用いて印字する例について説明します。

システム構成図



ケーブル結線図



D-sub 9 ピンコネクタ

名称	ピン番号
シールド	カバー
NC	1
NC	2
DATA	3
NC	4
SG	5
NC	6
NC	7
BUSY	8
NC	9

RJ45 コネクタ

ピン番号	名称	色
カバー	シールド	—
1	RD	白/橙
2	SD	橙
3	ER	白/緑
4	A	青
5	B	白/青
6	DR	緑
7	NC	白/茶
8	SG	茶

“BUSY” はプリンタの状態 (データ印字の不可) を外部に知らせるための信号です。“BUSY” は、使用するプリンタにより名称が異なる場合があります (例: DTR 等)。



“BUSY” は、プリンタによって動作仕様が異なりますのでプリンタの仕様を確認のうえ、結線を行ってください。



NC は接続しないでください。誤動作や故障の原因となります。

第6章 Modbus通信

この章では、FC6A形のModbus通信機能について説明します。

概要

Modbus通信は、FC6A形をModbus RTU通信のマスターやスレーブ、Modbus TCP通信のクライアントやサーバーとして使用し、外部機器と通信します。

Modbus通信で使用する通信ポート

対応する機種と通信ポートは、次のとおりです。

通信ポート	All-in-One CPU モジュール			CAN J1939 All-in-One CPU モジュール	Plus CPU モジュール	
	16点タイプ	24点タイプ	40点タイプ		Plus16点タイプ	Plus32点タイプ
ポート 1	○	○	○	—	○ ^{*2}	○ ^{*2}
ポート 2	○ ^{*1}	○ ^{*1}	○ ^{*1}	○ ^{*1}	○ ^{*2}	○ ^{*2}
ポート 3	—	—	○ ^{*1}	○ ^{*1}	○ ^{*3}	○ ^{*3}
ポート 4～33	○ (ポート 4～9) ^{*4}				○ (ポート 4～33) ^{*5}	
Ethernet ポート 1	○	○	○	○	○	○
Ethernet ポート 2	—	—	—	—	○	○
HMI-Ethernet ポート	—	—	—	—	—	—

*1 RS232C通信カートリッジまたはRS485通信カートリッジを接続した場合

*2 カートリッジベースモジュールを接続し、RS232C通信カートリッジまたはRS485通信カートリッジを接続した場合

*3 HMIモジュールを接続し、RS232C通信カートリッジまたはRS485通信カートリッジを接続した場合

*4 All-in-One CPUモジュールおよびCAN J1939 All-in-One CPUモジュールに通信モジュールを最大3台接続してポート4～9を増設できます。

*5 Plus CPUモジュールに通信モジュールを最大15台接続してポート4～33を増設できます。

Modbus通信の設定

各通信ポートで対応しているModbus通信の設定は、次のとおりです。

- ・ポート1～33を使用して外部機器とシリアル通信 (RS232C/RS485)
「シリアル通信によるModbus RTU通信」(6-2頁)
- ・Ethernetポート1および2を使用して外部機器とイーサネット通信
「イーサネット通信によるModbus TCP通信」(6-19頁)

シリアル通信による Modbus RTU 通信

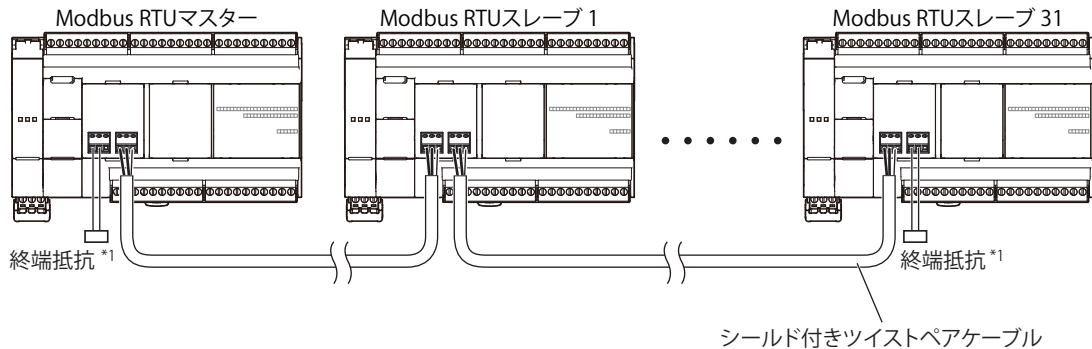
FC6A 形は、Modbus RTU プロトコルをサポートしており、Modbus RTU マスターと Modbus RTU スレーブの両機能に対応しています。

FC6A 形を Modbus RTU マスターに設定した場合、Modbus RTU スレーブ機器に対して、データの変更やモニタが行えます。

FC6A 形を Modbus RTU スレーブに設定した場合、Modbus RTU マスター機器から FC6A 形のデバイス内容の変更やモニタが行えます。

Modbus RTU マスターの機能と設定方法は、「Modbus RTU マスター」(6-2 頁)に、Modbus RTU スレーブの機能と設定方法は、「Modbus RTU スレーブ」(6-8 頁)にそれぞれ記載しています。

例) RS485 による Modbus RTU 通信



*1 通信品位が不安定な場合、特性インピーダンスに合わせた終端抵抗を両端に挿入ください。定格 1/2W 以上の抵抗を使用してください。

Modbus RTU マスター

FC6A 形を Modbus RTU マスターとして設定し、Modbus RTU スレーブ機器にリクエストを送信することにより、データの書き込み / 読み出しができます。各リクエストは、あらかじめ設定されたリクエストテーブルにしたがって、Modbus RTU スレーブ機器へ送信されます。

Modbus RTU マスターの通信設定やリクエストテーブルの作成は、WindLDR の [ファンクション設定] で行います。また、リクエストはリクエストテーブルに登録されている順 (番号 1、番号 2、...) に END 処理で実行され、通信はユーザープログラムの実行とは非同期で処理されます。

仕様

項目	設定値
通信速度 (bps)	9600 / 19200 / 38400 / 57600 / 115200
データビット長	8ビット固定
ストップビット	1 / 2ビット
パリティ	なし / 奇数 / 偶数
スレーブ番号*1	0、または1 ~ 247
最大スレーブ接続台数	RS232CによるModbus RTU通信：1台 RS485によるModbus RTU通信：31台
ケーブル総延長	RS232CによるModbus RTU通信 FC6A-PC1：5m FC6A-SIF52：15m RS485によるModbus RTU通信 FC6A-PC3：200m FC6A-SIF52：1200m
受信タイムアウト*2	10ms ~ 2550ms (10ms単位)
キャラクタ間タイムアウト時間	10ms固定
送信待ち時間	0ms ~ 5000ms (1ms単位)
リトライ回数	0 ~ 10回

*1 スレーブ番号 0 を設定するとブロードキャスト通信になります。ブロードキャストを使う事で、すべての Modbus RTU スレーブを対象にしたリクエストを送信できます。Modbus RTU スレーブは、ブロードキャスト通信に対してはリプライを返しません。同じデータをすべての Modbus RTU スレーブに書き込む場合に使用します。

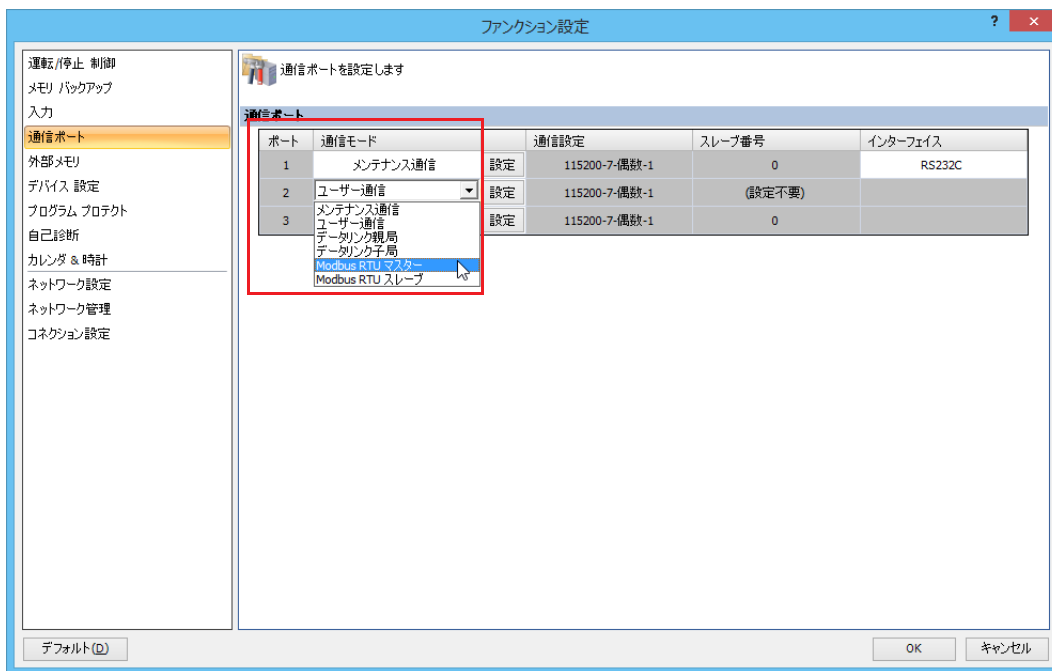
*2 スレーブの応答フレームの先頭を受信するまでのタイムアウト時間です。

WindLDR の設定

Modbus RTU マスターとして使用するには、[ファンクション設定] で Modbus RTU マスターの設定を行ったユーザープログラムを FC6A 形にダウンロードします。

●操作手順

1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するポートの [通信モード] をクリックし、“Modbus RTU マスター”を選択します。
[Modbus RTU マスターリクエストテーブル] ダイアログボックスが表示されます。



3. [通信設定] ボタンをクリックします。
[通信設定] ダイアログボックスが表示されます。



4. 通信速度、パリティ、ストップビット、リトライ回数、受信タイムアウト、送信待ち時間を設定します。

5. [OK] ボタンをクリックします。
[Modbus RTU マスターリクエストテーブル] ダイアログボックスに戻ります。
6. リクエスト内容を設定します。
255 個までリクエストの作成が可能です。通信実行デバイスとエラーステータスも必要に応じて設定します。

リクエスト番号	機能コード	マスターデバイスアドレス	データサイズ	ワードビット	スレーブ番号 (0 ~ 247)	Modbus スレーブアドレス	通信実行デバイス	エラーステータス
1	01 コイルの状態読み出し	D0000	08	ビット	012	001622		D0051
2	02 入力リレーの状態読み出し	D0012	04	ビット	014	101624		D0052
3								
4								
5								
6								
7								
8								
9								

7. [OK] ボタンをクリックします。
8. ユーザープログラムを FC6A 形にダウンロードします。
- 以上で、Modbus RTU マスターの設定が完了します。

設定内容

Modbus RTU マスターリクエストテーブル

通信実行デバイス

FC6A形がリクエストを実行するかどうかを、通信実行デバイスの ON/OFF により制御できます。通信実行デバイスには、内部リレーまたはデータレジスタを設定できます。データレジスタを設定した場合、データレジスタの最下位ビットから順番に、各リクエストに割り付けられます。

通信実行デバイス	開始・停止
使用	通信実行デバイスとして設定したデバイスを先頭に、リクエスト登録個数分のデバイスが通信実行デバイスとして割り当てられます。 たとえば、M0000を通信実行デバイスに設定した場合、番号順に、リクエスト番号1がM0000、番号2がM0001、…となります。各リクエストを実行するかどうかを、割り付けられたデバイスのON/OFFによって制御できます。Modbus RTUスレーブに対してリクエストを実行したい場合は、対応する通信実行デバイスをONします。FC6A形はリクエストを実行し、通信が完了すると対応する通信実行デバイスを自動的にOFFします。リクエストを常時実行したい場合は、該当する通信実行デバイスを、ユーザープログラムのOUT命令で常時ONして下さい。
未使用	登録されているすべてのリクエストを番号順に繰り返し実行します。

エラーステータス

Modbus RTU スレーブに対する書き込み / 読み出しが正常終了すると、リクエストが完了します。書き込み / 読み出し処理が失敗した場合、設定回数分リトライします。すべてのリトライで通信処理が失敗した場合は通信エラーとなり、エラーステータスにエラー情報を格納します。通信実行デバイスを使用している場合、通信エラー発生時も通信実行デバイスは OFF します。通信エラーが発生すると、エラーとなったリクエストの実行をキャンセルし、次のリクエストを実行します。

エラーステータス	内容
使用	設定したデバイスを先頭としてリクエスト番号順にエラーコードを格納します。 たとえば、エラーステータスにD0000を設定した場合、リクエスト番号1がD0000、番号2がD0001、…となります。 エラーステータスに格納されるエラー情報には、上位バイトにスレーブ番号、下位バイトにエラーコードが格納されます。
未使用	エラーが発生してもエラー情報は格納しません。

[同一 DR をすべての通信リクエストに使用する] にチェックが入っている場合、エラーステータスはリクエストごとに割り付けられず、設定したデータレジスタ（たとえば D0000）だけに、すべてのリクエストのエラー情報を格納します。

同一 DR をすべての通信リクエストに使用する	エラーステータス
チェックなし	リクエストごとにエラー情報を確認できます。設定したデータレジスタを先頭に、リクエスト登録数分のデータレジスタが各リクエストのエラーステータスとして割り当てられます。エラーが発生した場合、該当するリクエストのエラーステータスにエラー情報を格納し、保持します。
チェックあり	エラーが発生すると、設定したデータレジスタにエラー情報を格納します。エラーが複数回発生した場合、新しいエラー情報を上書きします。

エラーコード一覧

エラー情報の下位 8 ビットには、エラーコードとして下記の値を格納します。

エラーコード	内容	詳細
00h	正常終了	
01h	機能コードエラー（未対応の機能コード）	未対応の機能コードが送信された。
02h	アクセス先エラー	範囲外のデバイスアドレスに書き込み、または読み出しが実行された。
03h	デバイス数エラー 1ビット書き込みデータエラー	設定したデータサイズ、1ビット書き込みに対応していない。
12h	フレーム長エラー	範囲外のフレーム長のフレームを送信した。
13h	BCCエラー	BCC (Block Check Character) が一致しない。
14h	スレーブ番号異常	送信したスレーブ番号とリプライのスレーブ番号が一致しない。
16h	タイムアウトエラー	通信タイムアウトが発生した。

リクエスト番号

リクエストテーブルには、最大で 255 個のリクエストを登録できます。

1つのリクエストにつき 8 バイトのユーザープログラム領域を使用します。ユーザープログラムのサイズは、WindLDR のダウンロードダイアログボックスで確認できます。



リクエストテーブル作成時の注意事項

通信実行デバイスやエラーステータスに設定したデバイスは、リクエスト番号順に割り付けられます。リクエストテーブル途中のリクエストを削除したり、リクエストの順番を入れ替えたりした場合、通信実行デバイスやエラーステータスの割り付けが更新され、ユーザープログラムで使用しているデバイスと不整合が発生する可能性があります。リクエストテーブルを変更した場合、ユーザープログラムの確認を十分に行ってください。

機能コード

Modbus RTU マスターからスレーブへ送信するリクエストの内容を機能コードで設定します。FC6A 形の Modbus RTU マスターは次の表の機能コードをサポートします。通信相手となる Modbus RTU スレーブ機器の種類により、サポートしている機能コードや、アクセス可能なデータ範囲は異なります。通信する相手機器の仕様にしたがって、機能コードを設定してください。

機能コード	データサイズ範囲	スレーブアドレス範囲	FC6A 形を Modbus RTU スレーブとして使用する場合
01 コイルの状態読み出し	1～2000ビット	000001～065535	デバイスQ, R, Mのビット情報を読み出します。
02 入力リレーの状態読み出し	1～2000ビット	100001～165535	デバイスI, T (接点), C (接点) のビット情報を読み出します。
03 保持レジスタの内容読み出し	1～125ワード	400001～465535	デバイスD, T (設定値), C (設定値) のワード情報を読み出します。
04 入力レジスタの内容読み出し	1～125ワード	300001～365535	デバイスT (現在値), C (現在値) のワード情報を読み出します。
05 1点コイルの状態変更	1ビット	000001～065535	デバイスQ, R, Mのビット状態を変更します。
06 1点保持レジスタ書き込み	1ワード	400001～465535	デバイスDの内容を変更します。
15 N点コイルの状態変更	1～1968ビット	000001～065535	デバイスQ, R, Mのビット状態をN点連続で変更します。
16 N点保持レジスタへの書き込み	1～123ワード	400001～465535	デバイスDの内容をN点連続で変更します。

マスターデバイスアドレス

Modbus RTU スレーブのデータを読み出す（機能コードに 01, 02, 03, 04 のいずれかを選択した）場合は、Modbus RTU スレーブから読み出したデータを格納する先頭デバイスをマスターデバイスアドレスに設定します。また、Modbus RTU スレーブへデータを書き込む（機能コードに 05, 06, 15, 16 のいずれかを選択した）場合は、Modbus RTU スレーブへ書き込むデータが格納された先頭デバイスをマスターデバイスアドレスに設定します。マスターデバイスアドレスには、データレジスタと内部リレーが設定できません。

データサイズ

読み出しサイズ / 書き込みサイズを設定します。機能コードに 01, 02, 05, 15 のいずれかを選択した場合、ビット単位の設定となります。機能コードに 03, 04, 06, 16 のいずれかを選択した場合、ワード単位の設定となります。設定可能なデータサイズの範囲は、機能コードにより異なります。機能コード表のデータサイズ範囲を参照してください。

ワード / ビット

選択された機能コードに応じて、データ処理単位が「ワード」、または「ビット」として表示されます。

スレーブ番号

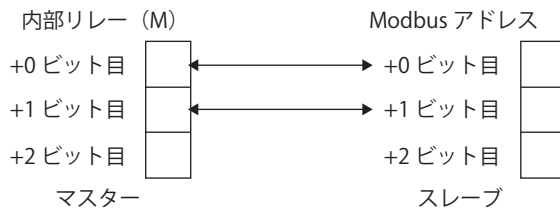
スレーブ番号を 0、または 1～247 の範囲で設定します。リクエスト No.1～255 の間に、同一のスレーブ番号を繰り返し設定できます。なお、Modbus 通信では、スレーブ番号 0 はブロードキャスト番号として扱います。ブロードキャストは、同一のデータをすべてのスレーブに書き込む場合に使用します。

スレーブアドレス

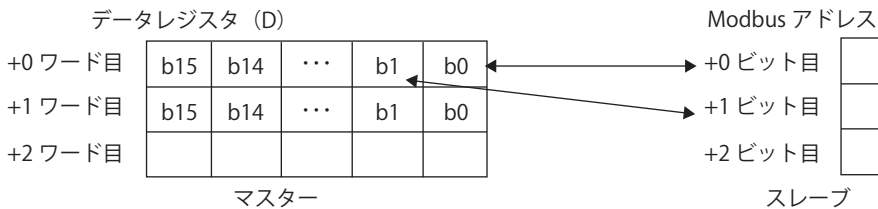
Modbus RTU スレーブのデータメモリアドレスを設定します。設定可能なスレーブアドレスの範囲は機能コードにより異なります。機能コード表のスレーブアドレス範囲を参照してください。また、スレーブアドレスの割り当ては、各 Modbus RTU スレーブ機器によって異なります。詳細は、各機器のマニュアルを参照してください。

スレーブアドレス（ビット単位）のリクエスト処理（機能コード：01, 02, 05, 15）

- マスターデバイスアドレスに内部リレー（M）を設定した場合

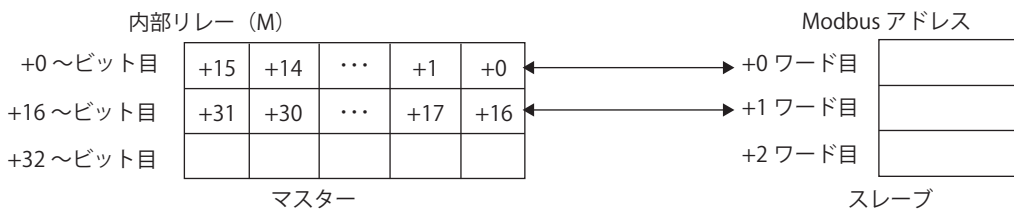


- マスターデバイスアドレスにデータレジスタ（D）を設定した場合

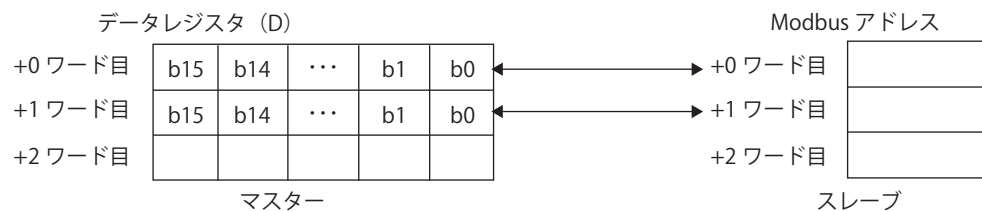


スレーブアドレス（ワード単位）のリクエスト処理（機能コード：03, 04, 06, 16）

- マスターデバイスアドレスに内部リレー（M）を設定した場合



- マスターデバイスアドレスにデータレジスタ（D）を設定した場合



通信設定

FC6A形の通信ポートを設定します。通信速度、パリティ、ストップビットについては、Modbus RTU スレーブ機器側の設定と合わせてください。

通信速度

Modbus RTU スレーブ機器との通信速度（bps）を設定します。9600/19200/38400/57600/115200 から選択します。

データビット長

データビット長は8ビット固定です。

パリティ

パリティを設定します。なし / 奇数 / 偶数から選択します。

ストップビット

ストップビットを設定します。1/2 から選択します。

リトライ回数

0～10回の間で設定します。

受信タイムアウト

10～2550msの間で設定してください。10ms単位で変更できます。通信遅れによるタイムアウトエラーが頻繁に発生する場合には、受信タイムアウトを大きい値にしてください。

送信待ち時間

Modbus RTU マスターが送信するリクエスト間の待ち時間を0～5000msの間で設定してください。1ms単位で変更できます。Modbus RTU マスターのリクエスト送信が早すぎてModbus RTU スレーブ機器が正しく受信しない場合は、待ち時間を大きくしてください。

Modbus RTU スレーブ

Modbus RTU スレーブは、Modbus RTU マスターから送られた自局宛のリクエスト（データメモリの読み出し要求や書き込み要求）を受け取り、ユーザープログラムの END 処理で、Modbus RTU マスターへ結果を返します。

なお、Modbus RTU スレーブは、ブロードキャスト番号（スレーブ番号 0）のリクエストに対しては、リクエスト処理のみを行い Modbus RTU マスターへは応答を返しません。

仕様

項目		設定値
通信速度 (bps)		9600 / 19200 / 38400 / 57600 / 115200
データビット長		8ビット固定
ストップビット		1 / 2ビット
パリティ		なし / 奇数 / 偶数
スレーブ番号	定数	1 ~ 247
	データレジスタ	特殊データレジスタの値を1~247の範囲で設定 ポート1 : D8100 ポート2 : D8102 ポート3 : D8103 ポート4~9 ^{*2} : D8040~D8045 ポート10~33 ^{*2*3} : D8735~D8758
キャラクタ間タイムアウト時間 ^{*1}		1.5キャラクタ以上 ^{*4}
フレーム間タイムアウト ^{*1}		3.5キャラクタ以上 ^{*5}

*1 タイムアウト発生時、FC6A 形は受信中のデータを破棄し、先頭フレーム受信待ちに移行します。

*2 通信モジュール使用時

*3 Plus CPU モジュールのみ

*4 19200bps を超える場合は 750 μ s 以上

*5 19200bps を超える場合は 1.75ms 以上

Modbus RTU スレーブのスレーブアドレスマップ

Modbus RTU マスターは、Modbus RTU スレーブの Modbus 用デバイス（コイルや入力リレー、入力レジスタ、保持レジスタ）にアクセスし、FC6A 形のデバイス（I, Q, M, R, T, C, D）の読み出しや書き込みを行うことができます。

下表を参照し、Modbus RTU マスター機器に、FC6A 形の Modbus 用デバイスのアドレス（FC6A 形ではスレーブアドレス）を設定してください。

Modbus 用デバイス名称	スレーブアドレス ^{*1}	通信上のスレーブアドレス ^{*2}	FC6A 形デバイス ^{*3}	対応機能コード
コイル (0xxxx番台)	000001~000516	0000~0203	Q0~Q643	1,5,15
	000701~000956	02BC~03BB	R000~R255	
	001001~003048	03E8~0BE7	M0000~M2557	
	003049~007400	0BE8~1CE7	M2560~M7997	
	009001~009256	2328~2427	M8000~M8317	
	009257~010600	2428~2967	M8320~M9997	
	011001~017000	2AF8~4267	M10000~M17497	
	017001~020000	4268~4E1F	M17500~M21247	
	030001~030480	7530~770F	Q1000~Q1597	
	030481~030960	7710~78EF	Q2000~Q2597	
	030961~031440	78F0~7ACF	Q3000~Q3597	
	031441~031920	7AD0~7CAF	Q4000~Q4597	
	031921~032400	7CB0~7E8F	Q5000~Q5597	
	032401~032880	7E90~806F	Q6000~Q6597	
	032881~033360	8070~824F	Q7000~Q7597	
	033361~033840	8250~842F	Q8000~Q8597	
033841~034320	8430~860F	Q9000~Q9597		
034321~034800	8610~87EF	Q10000~Q10597		

Modbus 用デバイス名称	スレーブアドレス *1	通信上の スレーブアドレス *2	FC6A 形デバイス *3	対応機能コード
入力リレー (1xxxx番台)	10001~100516	0000~0203	I0~I643	2
	101001~101256	03E8~04E7	T000~T255 (接点)	
	101501~101756	05DC~06DB	C000~C255 (接点)	
	102001~102768	07D0~0ACF	T256~T1023 (接点)	
	102769~103744	0AD0~0E9F	T1024~T1999 (接点)	
	104001~104256	0FA0~109F	C256~C511 (接点)	
	130001~130480	7530~770F	I1000~I1597	
	130481~130960	7710~78EF	I2000~I2597	
	130961~131440	78F0~7ACF	I3000~I3597	
	131441~131920	7AD0~7CAF	I4000~I4597	
	131921~132400	7CB0~7E8F	I5000~I5597	
	132401~132880	7E90~806F	I6000~I6597	
	132881~133360	8070~824F	I7000~I7597	
	133361~133840	8250~842F	I8000~I8597	
133841~134320	8430~860F	I9000~I9597		
134321~134800	8610~87EF	I10000~I10597		
入力レジスタ (3xxxx番台)	30001~300256	0000~00FF	T000~T255 (現在値)	4
	300501~300756	01F4~02F3	C000~C255 (現在値)	
	302001~302768	07D0~0ACF	T256~T1023 (現在値)	
	302769~303744	0AD0~0E9F	T1024~T1999 (現在値)	
	304001~304256	0FA0~109F	C256~C511 (現在値)	
保持レジスタ (4xxxx番台)	40001~408000	0000~1F3F	D0000~D7999	3,6,16
	408001~408500	1F40~2133	D8000~D8499	
	408501~408900	2134~22C3	D8500~D8899	
	409001~409256	2328~2427	T000~T255 (設定値)	3
	409501~409756	251C~261B	C000~C255 (設定値)	
	410001~456000	2710~DABF	D10000~D55999	3,6,16
	456001~462000	DAC0~F22F	D56000~D61999	
	462001~462768	F230~F52F	T256~T1023 (設定値)	3
	462769~463744	F530~F8FF	T1024~T1999 (設定値)	
464001~464256	FA00~FAFF	C256~C511 (設定値)		

*1 スレーブアドレスは、Modbus 通信で使用するアドレスです。「スレーブアドレスの計算方法」(6-10 頁)に FC6A 形のデバイスからスレーブアドレスを計算する方法を示します。

*2 通信上のスレーブアドレスは、通信フレームのアドレス部に入る 4 桁の数値です。スレーブアドレスの下 5 桁から 1 引いた値を 16 進数で格納します。

*3 ご使用の機種種のデバイス範囲内でアクセスしてください。

スレーブアドレスの計算方法

スレーブアドレスは使用するデバイスの種類によって、以下のように計算してください。

Modbus 用デバイス名称	FC6A 形のデバイス	最小アドレス①	オフセット②
コイル	Q0～Q643	0	1
	R000～R255	0	701
	M0000～M2557	0	1001
	M2560～M7997	256	3049
	M8000～M8317	800	9001
	M8320～M9997	832	9257
	M10000～M17497	1000	11001
	M17500～M21247	1750	17001
	Q1000～Q1597	100	30001
	Q2000～Q2597	200	30481
	Q3000～Q3597	300	30961
	Q4000～Q4597	400	31441
	Q5000～Q5597	500	31921
	Q6000～Q6597	600	32401
	Q7000～Q7597	700	32881
	Q8000～Q8597	800	33361
	Q9000～Q9597	900	33841
	Q10000～Q10597	1000	34321
入力リレー	I0～I643	0	100001
	T000～T255 (接点)	0	101001
	C000～C255 (接点)	0	101501
	T256～T1023 (接点)	256	102001
	T1024～T1999 (接点)	1024	102769
	C256～C511 (接点)	256	104001
	I1000～I1597	100	130001
	I2000～I2597	200	130481
	I3000～I3597	300	130961
	I4000～I4597	400	131441
	I5000～I5597	500	131921
	I6000～I6597	600	132401
	I7000～I7597	700	132881
	I8000～I8597	800	133361
	I9000～I9597	900	133841
	I10000～I10597	1000	134321
入力レジスタ	T000～T255 (現在値)	0	300001
	C000～C255 (現在値)	0	300501
	T256～T1023 (現在値)	256	302001
	T1024～T1999 (現在値)	1024	302769
	C256～C511 (現在値)	256	304001

Modbus 用デバイス名称	FC6A 形のデバイス	最小アドレス①	オフセット②
保持レジスタ	D0000~D7999	0	400001
	D8000~D8499	8000	408001
	D8500~D8899	8500	408501
	T000~T255 (設定値)	0	409001
	C000~C255 (設定値)	0	409501
	D10000~D55999	10000	410001
	D56000~D61999	56000	456001
	T256~T1023 (設定値)	256	462001
	T1024~T1999 (設定値)	256	462769
C256~C511 (設定値)	256	464001	

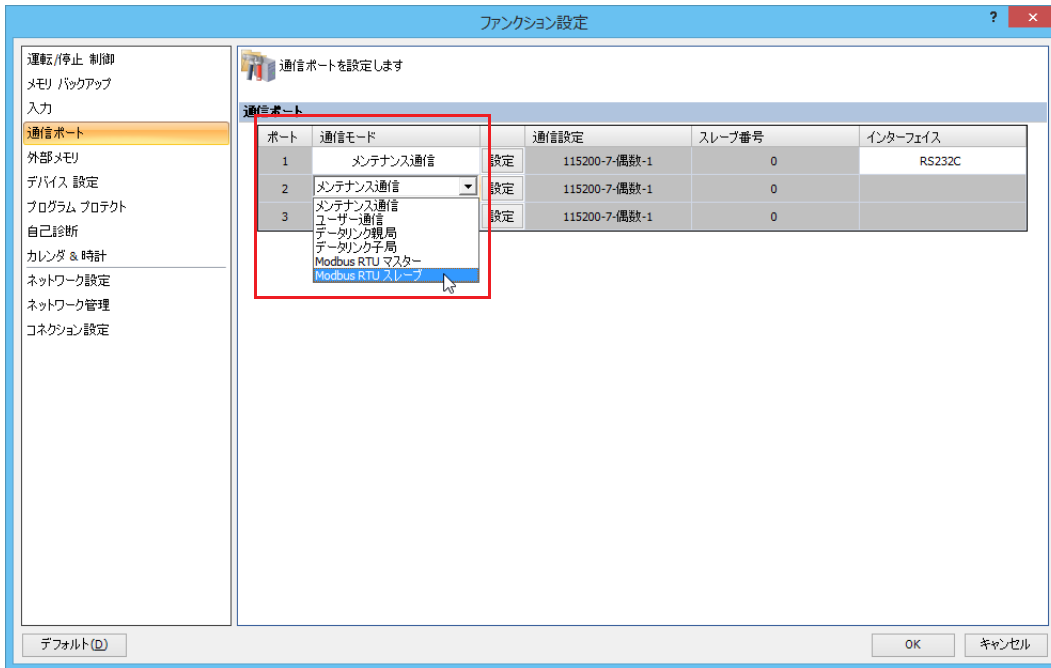
	FC6A 形のデバイス	スレーブアドレス算出	例
I, Q, M	M <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> 	$(II - ①) \times 8 + I + ②$ ①: 最小アドレス ②: オフセット	M325の場合 $(32-0) \times 8 + 5 + 1001 = 1262$ スレーブアドレス: 001262 通信上のスレーブアドレス: 04ED (16進)
R, T, C, D	D <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> 	$(III - ①) + ②$ ①: 最小アドレス ②: オフセット	D756の場合 $(756-0) + 400001 = 400757$ スレーブアドレス: 400757 通信上のスレーブアドレス: 02F4 (16進)

WindLDR の設定

FC6A 形を Modbus RTU スレーブとして使用するには、[ファンクション設定] で設定を行ったユーザープログラムを FC6A 形にダウンロードします。

●操作手順

1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するポートの [通信モード] をクリックし、“Modbus RTU スレーブ” を選択します。
[Modbus RTU スレーブ] ダイアログボックスが表示されます。



3. 通信速度、データビット長、パリティ、ストップビット、スレーブ番号を設定します。



4. [OK] ボタンをクリックします。
5. ユーザープログラムを FC6A 形にダウンロードします。

以上で、Modbus RTU スレーブの設定が完了します。

設定内容

通信設定

FC6A形の通信ポートを設定します。通信速度、パリティ、ストップビットについては、Modbus RTU マスター機器側の設定と合わせてください。

通信速度

Modbus RTU マスター機器との通信速度 (bps) を設定します。9600/19200/38400/57600/115200 から選択します。

データビット長

データビット長は8ビット固定です。

パリティ

パリティを設定します。なし / 奇数 / 偶数から選択します。

ストップビット

ストップビットを設定します。1 / 2 から選択します。

スレーブ番号

スレーブ番号は定数か、データレジスタかで設定方法が異なります。以下の範囲で設定してください。

定数 : 1 ~ 247

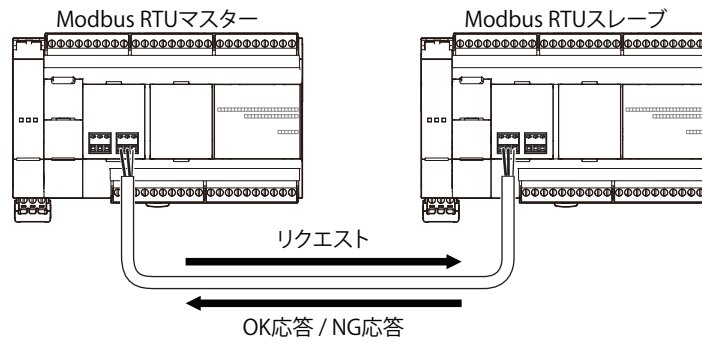
データレジスタ : D8100 (ポート 1)、D8102 (ポート 2)、D8103 (ポート 3)、D8040 ~ D8045 (ポート 4 ~ 9^{*1})、D8735 ~ D8758 (ポート 10 ~ 33^{*1*2}) の値を使用します。対応するデータレジスタにスレーブ番号を設定してください。

*1 通信モジュール使用時

*2 Plus CPU モジュールのみ

Modbus RTU 通信のフレームフォーマット

Modbus RTU マスター/スレーブの通信フレームフォーマットは以下の内容でやりとりされます。



RTU モードの通信フォーマット

Modbus RTU マスターからのリクエスト

"Idle" 3.5 文字	スレーブ番号	機能コード	各機能コードのデータ	CRC	"Idle" 3.5 文字
	1 バイト	1 バイト		2 バイト	

Modbus RTU スレーブからの OK 応答

"Idle" 3.5 文字	スレーブ番号	機能コード	各機能コードのデータ	CRC	"Idle" 3.5 文字
	1 バイト	1 バイト		2 バイト	

Modbus RTU スレーブからの NG 応答

"Idle" 3.5 文字	スレーブ番号	機能コード +80h	エラーコード	CRC	"Idle" 3.5 文字
	1 バイト	1 バイト	1 バイト	2 バイト	

* "Idle" とは、通信回線上にデータが流れていない状態のことです。

* Modbus RTU では、フレーム先頭を認識するために、フレーム間に 3.5 文字以上の "Idle" が必要です。

NG 応答エラーコード一覧

エラーコードに下記の値が格納されます。

エラーコード	内容	詳細
01h	機能コードエラー	未対応の機能コードが送信された。
02h	アクセス先エラー	範囲外のデバイスアドレスに書き込みまたは読み出しが実行された。
03h	デバイス数エラー 1ビット書き込みデータエラー	設定したデータサイズ、1ビット書き込みに対応していない。

チェックコードの計算方法

Modbus RTU で使用されるチェックコードは、以下の方法で計算します。

Modbus RTU プロトコル CRC の計算

スレーブ番号から CRC 格納位置の手前までの CRC-16 (周期冗長検査) を計算し、求めた 16 ビットデータを下位-上位の順に格納します。

CRC の計算方法 (生成多項式: $x^{16}+x^{15}+x^2+1$)

- ① 1 つ目のデータと FFFFh との排他的論理和 (XOR) を計算します。
- ② 結果を右へ 1 ビットシフトします。
- ③ シフト結果でキャリーが出たら、②の結果と固定値 (A001h) で XOR 演算をします。
- ④ 8 回シフトするまで②, ③を繰り返します。
- ⑤ 次のデータと上記結果の XOR 演算をします。
- ⑥ 最後のデータまで②から⑤を繰り返します。
- ⑦ 結果を CRC 格納位置に下位-上位の順で格納します。

機能ごとの通信フレームフォーマット

ここでは、Modbus RTU 通信で使用される機能コードとマスター / スレーブ間の通信フレームについて説明しています。マスターからスレーブに対して読み出し、書き込みリクエストが発行される場合、マスターおよびスレーブ間の通信データはこのフォーマットにしたがいます。

機能コード 1、2

Modbus RTU マスターは、Modbus RTU スレーブのコイルおよび入力リレーの状態を読み出します。読み出し連続ビット数は、1 ～ 2000 ビットです。FC6A 形の Modbus RTU スレーブの場合、デバイス Q, R, M および、I, T (接点), C (接点) のビット情報が読み出せません。

通信フレーム

Modbus RTU マスターからのリクエスト

スレーブ番号	機能コード	通信上のスレーブアドレス	ビット数
xxh	01h / 02h	xxxxh	xxxxh

Modbus RTU スレーブの OK 応答

スレーブ番号	機能コード	データ数	最初の 8 ビット	次の 8 ビット	...	最終の 8 ビット
xxh	01h / 02h	xxh	xxh	xxh	...	xxh

Modbus RTU スレーブの NG 応答

スレーブ番号	機能コード	エラーコード
xxh	81h / 82h	xxh

通信例

Q10 から 16 ビットを読み出します。(スレーブ番号を 8、Q10 ～ Q27 を 1234h とします。) 通信上のスレーブアドレスは Q10 = 0008h となり、ビット数は 16 ビット = 0010h となります。

リクエスト	<08 01 0008 0010 (CRC)>
OK 応答	<08 01 02 34 12 (CRC)>
NG 応答	<08 81 xx (CRC)>

機能コード 3, 4

デバイス D, T (設定値), C (設定値) および T (現在値), C (現在値) のワード情報を読み出します。読み出し連続ワード数は、1 ～ 125 ワードです。

通信フレーム

Modbus RTU マスターからのリクエスト

スレーブ番号	機能コード	通信上のスレーブアドレス	ワード数
xxh	03h / 04h	xxxxh	xxxxh

Modbus RTU スレーブの OK 応答

スレーブ番号	機能コード	データ数	最初の上位バイト	最初の下位バイト	...	最終の下位バイト
xxh	03h / 04h	xxh	xxh	xxh	...	xxh

Modbus RTU スレーブの NG 応答

スレーブ番号	機能コード	エラーコード
xxh	83h / 84h	xxh

通信例

D1710 から 2 ワード読み出します。(スレーブ番号を 8、D1710=1234h、D1711=5678h とします。) 通信上のスレーブアドレスは D1710 = 06AEh となり、ワード数は 2 ワード = 0002h となります。

リクエスト	<08 03 06AE 0002 (CRC)>
OK 応答	<08 03 04 12 34 56 78 (CRC)>
NG 応答	<08 83 xx (CRC)>

機能コード 5

デバイス Q, R, M のビット状態を変更します。

■ 通信フレーム

Modbus RTU マスターからのリクエスト

スレーブ番号	機能コード	通信上のスレーブアドレス	OFF : 0000h ON : FF00h
xxh	05h	xxxxh	xxxxh

Modbus RTU スレーブの OK 応答

スレーブ番号	機能コード	通信上のスレーブアドレス	OFF : 0000h ON : FF00h
xxh	05h	xxxxh	xxxxh

Modbus RTU スレーブの NG 応答

スレーブ番号	機能コード	エラーコード
xxh	85h	xxh

■ 通信例

M1320 を ON します。(スレーブ番号を 8 とします。)

通信上のスレーブアドレスは M1320 = 0808h となり、書込データは ON = FF00h となります。

リクエスト	<08 05 0808 FF00 (CRC)>
OK 応答	<08 05 0808 FF00 (CRC)>
NG 応答	<08 85 xx (CRC)>

機能コード 6

デバイス D の内容を変更します。

■ 通信フレーム

Modbus RTU マスターからのリクエスト

スレーブ番号	機能コード	通信上のスレーブアドレス	書込データ
xxh	06h	xxxxh	xxxxh

Modbus RTU スレーブの OK 応答

スレーブ番号	機能コード	通信上のスレーブアドレス	確認情報
xxh	06h	xxxxh	xxxxh

Modbus RTU スレーブの NG 応答

スレーブ番号	機能コード	エラーコード
xxh	86h	xxh

■ 通信例

D1708 に 8,000 を書き込みます。(スレーブ番号を 8 とします。)

通信上のスレーブアドレスは D1708 = 06ACh となり、書込データは 8,000 = 1F40h となります。

リクエスト	<08 06 06AC 1F40 (CRC)>
OK 応答	<08 06 06AC 1F40 (CRC)>
NG 応答	<08 86 xx (CRC)>

機能コード 15

デバイス Q, R, M のビット状態を N 点連続で変更します。連続書き込みビット数は 1 ~ 1968 です。

■通信フレーム

Modbus RTU マスターからのリクエスト

スレーブ番号	機能コード	通信上のスレーブアドレス	ビット数	データ数	最初の 8 ビット	次の 8 ビット	...	最終の 8 ビット
xxh	0Fh	xxxxh	xxxxh	xxh	xxh	xxh	...	xxh

Modbus RTU スレーブの OK 応答

スレーブ番号	機能コード	通信上のスレーブアドレス	ビット数
xxh	0Fh	xxxxh	xxxxh

Modbus RTU スレーブの NG 応答

スレーブ番号	機能コード	エラーコード
xxh	8Fh	xxh

■通信例

M0605 ~ M0624 の 16 点に次のビット情報を書き込みます。(スレーブ番号を 8 とします。)
通信上のスレーブアドレスは M0605 = 05CDh となり、データ数は 2 = 02h となります。

ービット情報ー

"6B" = M0605 (下位) ~ M0614 (上位)、"02" = M0615 (下位) ~ M0624 (上位)

M0607	M0606	M0605	M0604	M0603	M0602	M0601	M0600
(OFF)	(ON)	(ON)				
M0617	M0616	M0615	M0614	M0613	M0612	M0611	M0610
(OFF)	(ON)	(OFF)	(OFF)	(ON)	(ON)	(OFF)	(ON)
M0627	M0626	M0625	M0624	M0623	M0622	M0621	M0620
		(OFF)	(OFF)	(OFF)	(OFF)	(OFF)

リクエスト	<08 0F 05CD 0010 02 6B 02(CRC)>
OK 応答	<08 0F 05CD 0010 (CRC)>
NG 応答	<08 8F xx (CRC)>

機能コード 16

デバイス D の内容を連続で変更します。連続書き込みワード数は 1 ～ 123 です。

■ 通信フレーム

Modbus RTU マスターからのリクエスト

スレーブ番号	機能コード	通信上のスレーブアドレス	ワード数	データ数	最初の上 位バイト	最初の下 位バイト	...	最終の下 位バイト
xxh	10h	xxxxh	xxxxh	xxh	xxh	xxh	...	xxh

Modbus RTU スレーブの OK 応答

スレーブ番号	機能コード	通信上のスレーブアドレス	ワード数
xxh	10h	xxxxh	xxxxh

Modbus RTU スレーブの NG 応答

スレーブ番号	機能コード	エラーコード
xxh	90h	xxh

■ 通信例

D1708 から 4 ワードに次の内容を書き込みます。(スレーブ番号を 8 とします。)

通信上のスレーブアドレスは D1708 = 06ACh となり、ワード数は 4 ワード = 0004h となります。

ーワード情報ー

D1708	D1709	D1710	D1711
(1234h)	(5678h)	(ABCDh)	(EF01h)

リクエスト	<08 10 06AC 0004 08 12 34 56 78 AB CD EF 01 (CRC)>
OK 応答	<08 10 06AC 0004 (CRC)>
NG 応答	<08 90 xx (CRC)>

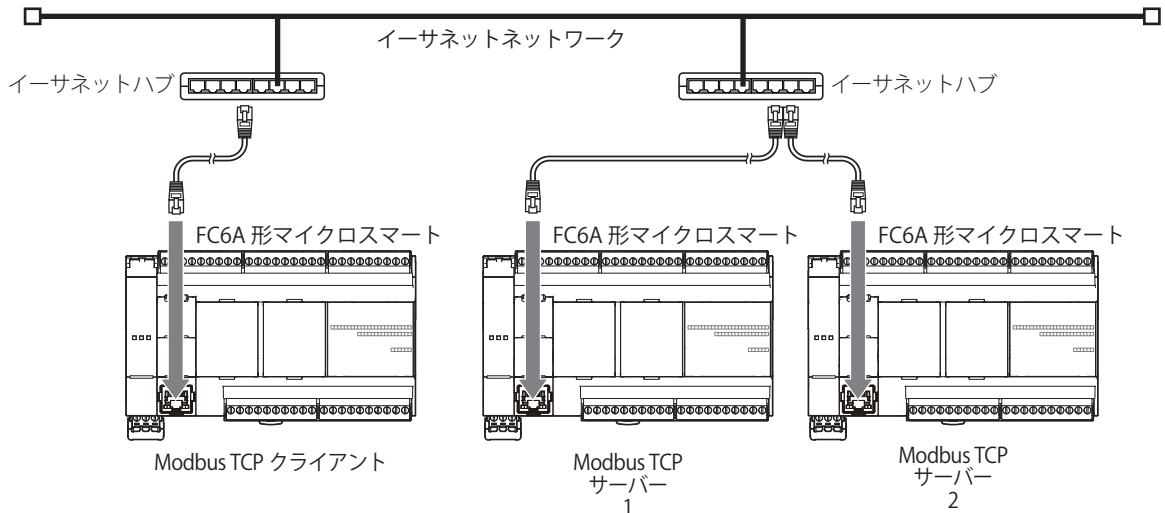
イーサネット通信による Modbus TCP 通信

FC6A 形は、Modbus TCP クライアントと Modbus TCP サーバーに対応しています。Ethernet ポート 1 と Ethernet ポート 2 (Plus CPU モジュールのみ) を使用して、Modbus TCP 対応機器と通信できます。

FC6A 形を Modbus TCP クライアントに設定した場合、Modbus TCP サーバー対応機器のデータメモリの変更やモニタが行えます。All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールは最大 8 個、Plus CPU モジュールは最大 16 個のコネクションを Modbus TCP クライアントに割り当てることができます。各コネクションは、それぞれ複数台 (1 ~ 255 台) の Modbus TCP サーバー機器と通信できます。

FC6A 形を Modbus TCP サーバーに設定した場合、Modbus TCP クライアント対応機器から FC6A 形のデバイス内容の変更やモニタが行えます。All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールは最大 8 個、Plus CPU モジュールは最大 16 個のコネクションを Modbus TCP サーバーに割り当てることができます。

Modbus TCP クライアントの機能と設定方法は、「Modbus TCP クライアント」(6-19 頁) に、Modbus TCP サーバーの機能と設定方法は、「Modbus TCP サーバー」(6-25 頁) にそれぞれ記載しています。



Modbus TCP クライアント

Modbus TCP 通信では、Modbus TCP クライアントから Modbus TCP サーバーへリクエストを送信することにより、データの書き込み / 読み出しを行うことができます。各リクエストは、あらかじめ設定されたリクエストテーブルにしたがって、Modbus TCP サーバーへ送信されます。

Modbus TCP クライアントの通信設定やリクエストテーブルの作成は、WindLDR の [ファンクション設定] で行います。また、リクエストはリクエストテーブルに登録されている順 (番号 1、番号 2、...) に END 処理で実行され、通信はユーザープログラムの実行とは非同期で処理されます。Plus CPU モジュールの Modbus TCP クライアント通信では、Modbus TCP サーバーと常時接続して (コネクションを切断せずに) Modbus TCP 通信を行うかをリクエストテーブルごとに設定できます。常時接続可能なリモートホストの数は最大で 128 個です。常時接続設定しないリモートホストの場合は、Modbus TCP 通信のたびに Modbus TCP サーバーとのコネクションの接続を切断します。

通信設定

項目	内容
受信タイムアウト*1	100ms~25500ms (100ms単位)
送信待ち時間	0ms~5000ms (0ms)

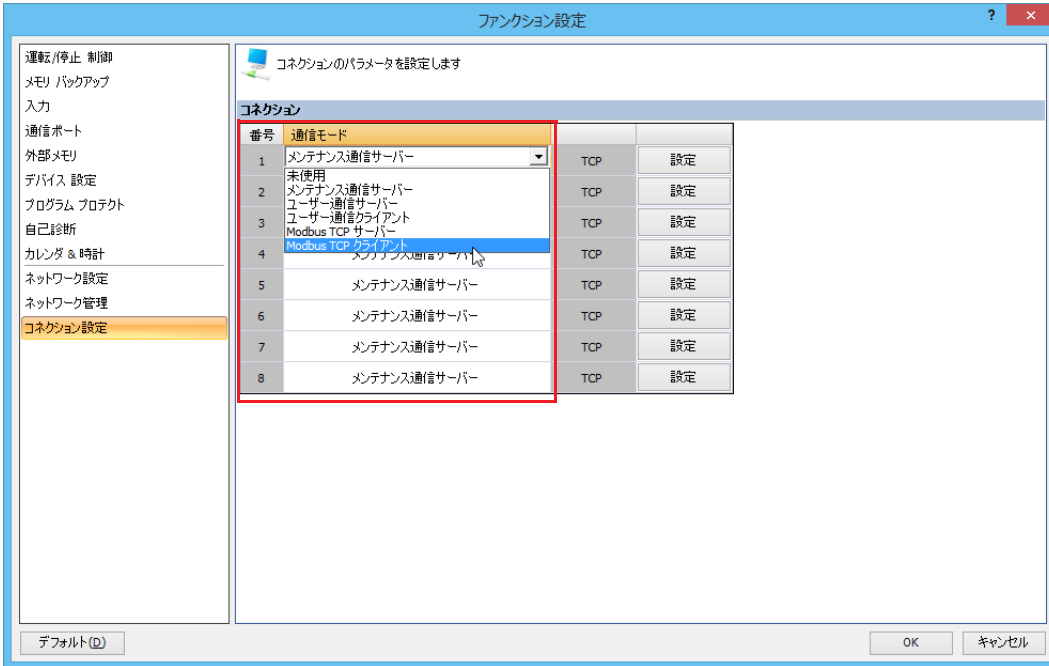
*1 Modbus TCP サーバーからの応答フレームの先頭を受けるまでのタイムアウト時間です。

WindLDR の設定

FC6A 形を Modbus TCP クライアントとして使用するには、[ファンクション設定] で設定を行ったユーザープログラムを FC6A 形にダウンロードします。

●操作手順

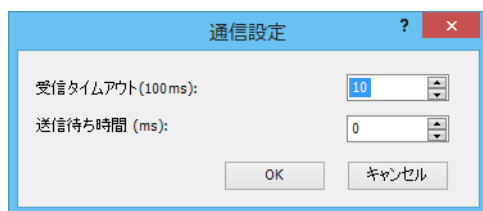
1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するコネクション番号の [通信モード] をクリックし、“Modbus TCP クライアント” を選択します。
[Modbus TCP クライアントのリクエストテーブル設定] ダイアログボックスが表示されます。



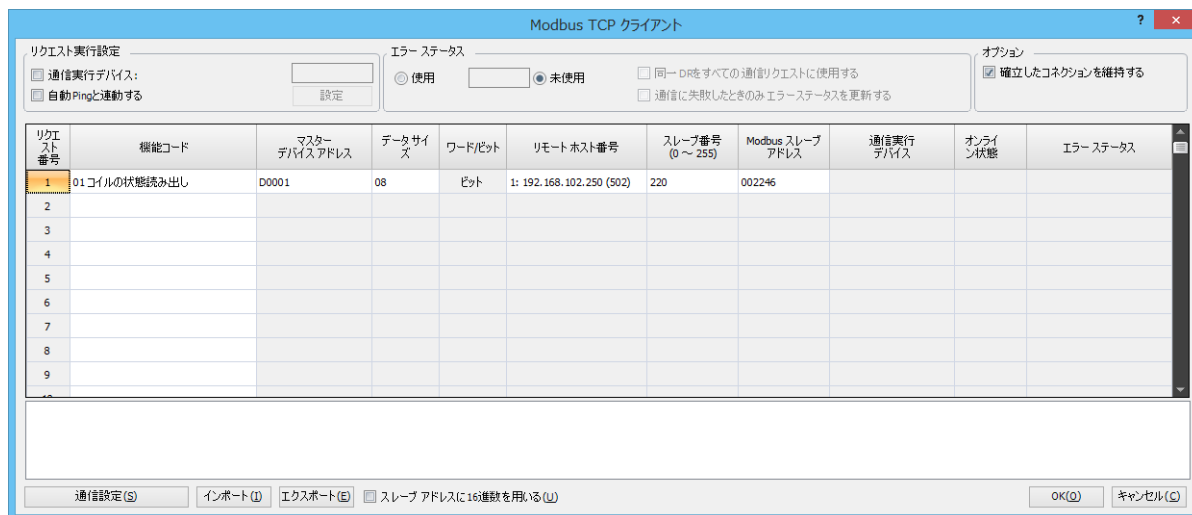
3. [通信設定] ボタンをクリックします。
[通信設定] ダイアログボックスが表示されます。



- 受信タイムアウト時間と送信待ち時間を設定します



- [OK] ボタンをクリックします。
[Modbus TCP クライアントリクエストテーブル] ダイアログボックスに戻ります。
- リクエスト内容を設定します。
255 個までリクエストの作成が可能です。通信実行デバイスとエラーステータスも必要に応じて設定します。



- [OK] ボタンをクリックします。
- ユーザープログラムを FC6A 形にダウンロードします。

以上で、Modbus TCP クライアントの設定が完了します。

設定内容

Modbus TCP クライアントリクエストテーブル

通信実行デバイス

Modbus TCP クライアントからのリクエストを送信するかどうか、通信実行デバイスの ON/OFF により制御できます。通信実行デバイスには、内部リレーまたはデータレジスタを設定できます。データレジスタを設定した場合、データレジスタの最下位ビットから順番に、各リクエストに割り付けられます。

通信実行デバイス	開始・停止
使用	リクエストテーブルに設定したリクエスト登録個数分のデバイスを、通信実行デバイスとして使用します。たとえば、M0000を通信実行デバイスに設定した場合、リクエスト番号1がM0000、番号2がM0001、…となります。各リクエストを実行するかどうかを、割り付けられたデバイスのON/OFFによって制御できます。Modbus TCPサーバーに対してリクエストを実行したい場合は、対応する通信実行デバイスをONします。FC6A形はリクエストを実行し、通信が完了すると、対応する通信実行デバイスを自動的にOFFにします。リクエストを常時実行したい場合は、該当する通信実行デバイスを、ユーザープログラムのOUT命令で常時ONしてください。
未使用	登録されているすべてのリクエストを番号順に繰り返し実行します。
自動Ping連動	Modbus TCPクライアントからのリクエストを送信するかどうか、自動Ping連動によるオンライン状態のON/OFFにより制御できます。ネットワークに参加してないリモートホストへのリクエストを送信対象外とすることで、不要なタイムアウト時間などが発生しなくなります。自動Ping連動については、「第3章 自動Ping機能」(3-23頁)を参照してください。

リクエスト実行設定の組み合わせによる本体の動作は、次のとおりです。

リクエスト実行設定		自動 Ping 連動	
		使用しない	使用する
通信実行デバイス	使用しない	常時、全リクエストを送信します。	自動Pingの送信対象になっているリモートホスト： オンライン状態がONの場合は該当リクエストを送信し、OFFの場合は該当リクエストを送信しません。 自動Pingの送信対象になっていないリモートホスト： オンライン状態の内部リレーのON/OFFに関わらず、該当リクエストを送信します。
	使用する	通信実行デバイスがONの場合は該当リクエストを送信し、OFFの場合は該当リクエストを送信しません。	自動Pingの送信対象になっているリモートホスト： オンライン状態と通信実行デバイスの両方がONの場合は該当リクエストを送信し、それ以外は該当リクエストを送信しません。 自動Pingの送信対象になっていないリモートホスト： オンライン状態の内部リレーのON/OFFに関わらず、通信実行デバイスがONの場合は該当リクエストを送信し、OFFの場合は該当リクエストを送信しません。

エラーステータス

Modbus TCP サーバーに対する書き込み/読み出しが正常終了すると、リクエスト実行が完了します。書き込み/読み出し処理が失敗した場合、3回リトライします。すべてのリトライで通信処理が失敗した場合は通信エラーとなり、エラーステータスにエラー情報を格納します。通信実行デバイスを使用している場合は通信エラー発生時も通信実行デバイスはOFFします。通信エラーとなったリクエストの実行はキャンセルされ、続けて次のリクエストを実行します。リクエストのエラーステータスをスキャンごとに確認することで、通信状況を確認することができます。

エラーステータス	内容
使用	設定したデバイスを先頭としてリクエスト番号順にエラーコードを格納します。たとえば、エラーステータスにD0000を設定した場合、リクエスト番号1がD0000、番号2がD0001、…となります。エラーステータスに格納されるエラー情報には、上位バイトにリモートホスト番号、下位バイトにエラーコードが格納されます。
未使用	エラーが発生してもエラー情報を格納しません。

[同一 DR をすべての通信リクエストに使用する] チェックボックスをオンにしている場合、エラーステータスはリクエストごとに割り付けられず、設定したデータレジスタ (たとえば D0000) だけに、すべてのリクエストのエラーステータスを格納します。

同一 DR をすべての通信リクエストに使用する	エラーステータス
チェックなし	リクエストごとにエラー情報を確認できます。設定したデータレジスタを先頭に、リクエスト登録数分のデータレジスタが各リクエストのエラーステータスとして割り当てられます。エラーが発生した場合、該当するリクエストのエラーステータスにエラー情報を格納し、保持します。
チェックあり	エラーが発生すると、設定したデータレジスタにエラー情報を格納します。エラーが複数回発生した場合、新しいエラー情報を上書きします。


エラーコード一覧

エラー情報の下位 8 ビットには、エラーコードとして下記の値を格納します。

エラーコード	内容	詳細
00h	正常終了	
01h	機能コードエラー (未対応の機能コード)	未対応の機能コードが送信された。
02h	アクセス先エラー	範囲外のデバイスアドレスに書き込み、または読み出しが実行された。
03h	デバイス数エラー 1ビット書き込みデータエラー	設定したデータサイズ、1ビット書き込みに対応していない。
12h	フレーム長エラー	範囲外のフレーム長のフレームを送信した。
13h	BCCエラー	BCC (Block Check Character) が一致しない。
14h	スレーブ番号異常	送信したスレーブ番号とリブライのスレーブ番号が一致しない。
16h	タイムアウトエラー	通信タイムアウトが発生した。

リクエスト番号

リクエストテーブルには、最大で 255 個のリクエストを登録できます。1 つのリクエストにつき 10 バイトのユーザープログラム領域を使用します。ユーザープログラムのサイズは、WindLDR の [ダウンロード] ダイアログボックスで確認できます。

 リクエストテーブル作成時の注意事項

通信実行デバイスやエラーステータスに設定したデバイスは、リクエスト番号順に割り付けられます。リクエストテーブル途中のリクエストを削除したり、リクエストの順番を入れ替えたりした場合、通信実行デバイスやエラーステータスの割り付けが更新され、ユーザープログラムで使用しているデバイスと不整合が発生する可能性があります。リクエストテーブルを変更した場合、ユーザープログラムの確認を十分に行ってください。

機能コード

Modbus TCP クライアントからサーバーへ送信するリクエストの内容を機能コードで設定します。FC6A 形の Modbus TCP クライアントは次の表の機能コードをサポートします。通信相手となる Modbus TCP サーバー機器の種類により、サポートしている機能コードや、アクセス可能なデータ範囲は異なります。通信する相手機器の仕様にしたがって、機能コードを設定してください。

機能コード	データサイズ範囲	スレーブ アドレス範囲	FC6A 形を Modbus TCP サーバーとして使用する場合
01 コイルの状態読み出し	1 ~ 2000ビット	000001 ~ 065535	デバイスQ, R, Mのビット情報を読み出します。
02 入力リレーの状態読み出し	1 ~ 2000ビット	100001 ~ 165535	デバイスI, T (接点), C (接点) のビット情報を読み出します。
03 保持レジスタの内容読み出し	1 ~ 125ワード	400001 ~ 465535	デバイスD, T (設定値), C (設定値) のワード情報を読み出します。
04 入力レジスタの内容読み出し	1 ~ 125ワード	300001 ~ 365535	デバイスT (現在値), C (現在値) のワード情報を読み出します。
05 1点コイルの状態変更	1ビット	000001 ~ 065535	デバイスQ, R, Mのビット状態を変更します。
06 1点保持レジスタ書き込み	1ワード	400001 ~ 465535	デバイスDの内容を変更します。
15 N点コイルの状態変更	1 ~ 1968ビット	000001 ~ 065535	デバイスQ, R, Mのビット状態をN点連続で変更します。
16 N点保持レジスタへの書き込み	1 ~ 123ワード	400001 ~ 465535	デバイスDの内容を連続で変更します。

マスターデバイスアドレス

Modbus TCP サーバーのデータを読み出す (機能コードに 01, 02, 03, 04 のいずれかを選択した) 場合は、Modbus TCP サーバーから読み出したデータを格納する先頭デバイスをマスターデバイスアドレスに設定します。また、Modbus TCP サーバーへデータを書き込む (機能コードに 05, 06, 15, 16 のいずれかを選択した) 場合は、Modbus TCP サーバーへ書き込むデータが格納された先頭デバイスをマスターデバイスアドレスに設定します。マスターデバイスアドレスには、データレジスタと内部リレーが設定できます。

データサイズ

読み出しサイズ / 書き込みサイズを設定します。機能コードに 01, 02, 05, 15 のいずれかを選択した場合、ビット単位の設定となります。機能コードに 03, 04, 06, 16 のいずれかを選択した場合、ワード単位の設定となります。設定可能なデータサイズの範囲は、機能コードにより異なります。機能コード表のデータサイズ範囲を参照してください。

ワード / ビット

選択された機能コードに応じて、データ処理単位が「ワード」、または「ビット」として表示されます。

リモートホスト番号 (1 ~ 255)

リモートホスト番号を 1 ~ 255 の範囲で設定します。Modbus TCP クライアントは 1 : N 通信をサポートしていますので、複数の Modbus TCP サーバーと通信することができます。

Plus CPU モジュールの場合、リモートホスト番号は、Modbus TCP クライアントのコネクション番号ごとに設定されている Ethernet ポート番号と同じ Ethernet ポート番号のリモートホスト番号のみ設定できます。

リモートホストリストについては、「リモートホストリスト」(3-21 頁) を参照してください。

スレーブ番号

スレーブ番号を 0 ~ 255 の範囲で設定します。リクエスト No.1 ~ 255 の間に、同一のスレーブ番号を繰り返し設定できます。

Modbus TCP サーバーがスレーブ番号を参照しない場合は、任意のスレーブ番号を設定してください。

スレーブアドレス

Modbus TCP サーバーのデータメモリアドレスを設定します。設定可能なスレーブアドレスの範囲は機能コードにより異なります。

機能コード表のスレーブアドレス範囲を参照してください。また、スレーブアドレスの割り当ては、各 Modbus TCP サーバー機器によって異なります。詳細は、各機器のマニュアルを参照してください。

確立したコネクションを維持する

Plus CPU モジュールの Modbus TCP クライアント通信では、Modbus TCP サーバーと常時接続して (コネクションを切断せずに)

Modbus TCP 通信を行うかをリクエストテーブルごとに設定できます。常時接続可能なリモートホストの数は最大で 128 個です。

常時接続設定しないリクエストの場合は、Modbus TCP 通信のたびに Modbus TCP サーバーとのコネクションの接続を切断します。

Modbus TCP サーバー

FC6A 形を Modbus TCP サーバーとして設定すると、Modbus TCP クライアント機器は FC6A 形に対し、Modbus TCP 通信でデータの書き込み / 読み出しが出来ます。Modbus TCP サーバーでは、Modbus TCP クライアント機器側から送信されたリクエストを受信すると、そのリクエストにしたがい、デバイスの読み出し / 書き込みを行います。FC6A 形は Modbus TCP クライアントから受信したリクエストを END 処理で処理します。

仕様

項目	Modbus TCP サーバー
応答時間	1.5ms
同時接続可能クライアント数*1	All-in-One CPUモジュール/CAN J1939 All-in-One CPUモジュール：8（1コネクションにつき1個） Plus CPUモジュール：16（1コネクションにつき1個）
ポート番号	502（0～65535の間で変更可能）
対応機能コード	1：コイルの状態読み出し 2：入力リレーの状態読み出し 3：保持レジスタの状態読み出し 4：入力レジスタの状態読み出し 5：1点コイルの状態変更 6：1点保持レジスタ書き込み 15：N点コイルの状態変更 16：N点保持レジスタの書き込み

*1 All-in-One CPU モジュール / CAN J1939 All-in-One CPU モジュールは 8 個、Plus CPU モジュールは 16 個のコネクションすべてが Modbus TCP サーバーに設定された場合の個数です。

Modbus TCP サーバーのスレーブアドレスマップ

Modbus TCP クライアントは、Modbus TCP サーバーの Modbus 用デバイス（コイルや入力リレー、入力レジスタ、保持レジスタ）にアクセスし、FC6A 形のデバイス（I, Q, M, R, T, C, D）の読み出しや書き込みを行うことができます。下表を参照し、Modbus TCP クライアント機器に、FC6A 形の Modbus 用デバイスのアドレス（FC6A 形ではスレーブアドレス）を設定してください。

Modbus 用デバイス名称	スレーブアドレス	通信上のスレーブアドレス*1	FC6A 形デバイス*2	対応機能コード
コイル (0xxxx番台)	000001～000516	0000～0203	Q0～Q643	1,5,15
	000701～000956	02BC～03BB	R000～R255	
	001001～003048	03E8～0BE7	M0000～M2557	
	003049～007400	0BE8～1CE7	M2560～M7997	
	009001～009256	2328～2427	M8000～M8317	
	009257～010600	2428～2967	M8320～M9997	
	011001～017000	2AF8～4267	M10000～M17497	
	017001～020000	4268～4E1F	M17500～M21247	
	030001～030480	7530～770F	Q1000～Q1597	
	030481～030960	7710～78EF	Q2000～Q2597	
	030961～031440	78F0～7ACF	Q3000～Q3597	
	031441～031920	7AD0～7CAF	Q4000～Q4597	
	031921～032400	7CB0～7E8F	Q5000～Q5597	
	032401～032880	7E90～806F	Q6000～Q6597	
	032881～033360	8070～824F	Q7000～Q7597	
	033361～033840	8250～842F	Q8000～Q8597	
033841～034320	8430～860F	Q9000～Q9597		
034321～034800	8610～87EF	Q10000～Q10597		

Modbus 用デバイス名称	スレーブアドレス	通信上の スレーブアドレス*1	FC6A 形デバイス*2	対応機能コード
入力リレー (1xxxx番台)	10001~100516	0000~0203	I0~I643	2
	101001~101256	03E8~04E7	T000~T255 (接点)	
	101501~101756	05DC~06DB	C000~C255 (接点)	
	102001~102768	07D0~0ACF	T256~T1023 (接点)	
	102769~103744	0AD0~0E9F	T1024~T1999 (接点)	
	104001~104256	0FA0~109F	C256~C511 (接点)	
	130001~130480	7530~770F	I1000~I1597	
	130481~130960	7710~78EF	I2000~I2597	
	130961~131440	78F0~7ACF	I3000~I3597	
	131441~131920	7AD0~7CAF	I4000~I4597	
	131921~132400	7CB0~7E8F	I5000~I5597	
	132401~132880	7E90~806F	I6000~I6597	
	132881~133360	8070~824F	I7000~I7597	
	133361~133840	8250~842F	I8000~I8597	
	133841~134320	8430~860F	I9000~I9597	
134321~134800	8610~87EF	I10000~I10597		
入力レジスタ (3xxxx番台)	300001~300256	0000~00FF	T000~T255 (現在値)	4
	300501~300756	01F4~02F3	C000~C255 (現在値)	
	302001~302768	07D0~0ACF	T256~T1023 (現在値)	
	302769~303744	0AD0~0E9F	T1024~T1999 (現在値)	
	304001~304256	0FA0~109F	C256~C511 (現在値)	
保持レジスタ (4xxxx番台)	400001~408000	0000~1F3F	D0000~D7999	3,6,16
	408001~408500	1F40~2133	D8000~D8499	
	408501~408900	2134~22C3	D8500~D8899	
	409001~409256	2328~2427	T000~T255 (設定値)	3
	409501~409756	251C~261B	C000~C255 (設定値)	
	410001~456000	2710~DABF	D10000~D55999	3,6,16
	456001~462000	DAC0~F22F	D56000~D61999	
	462001~462768	F230~F52F	T256~T1023 (設定値)	3
	462769~463744	F530~F8FF	T1024~T1999 (設定値)	
464001~464256	FA00~FAFF	C256~C511 (設定値)		

*1 通信上のスレーブアドレスは、通信フレームのアドレス部に入る 4 桁の数値です。スレーブアドレスの下 5 桁から 1 引いた値を 16 進数で格納します。アドレスの計算方法については、「Modbus RTU スレーブ」(6-8 頁)の項を参照してください。

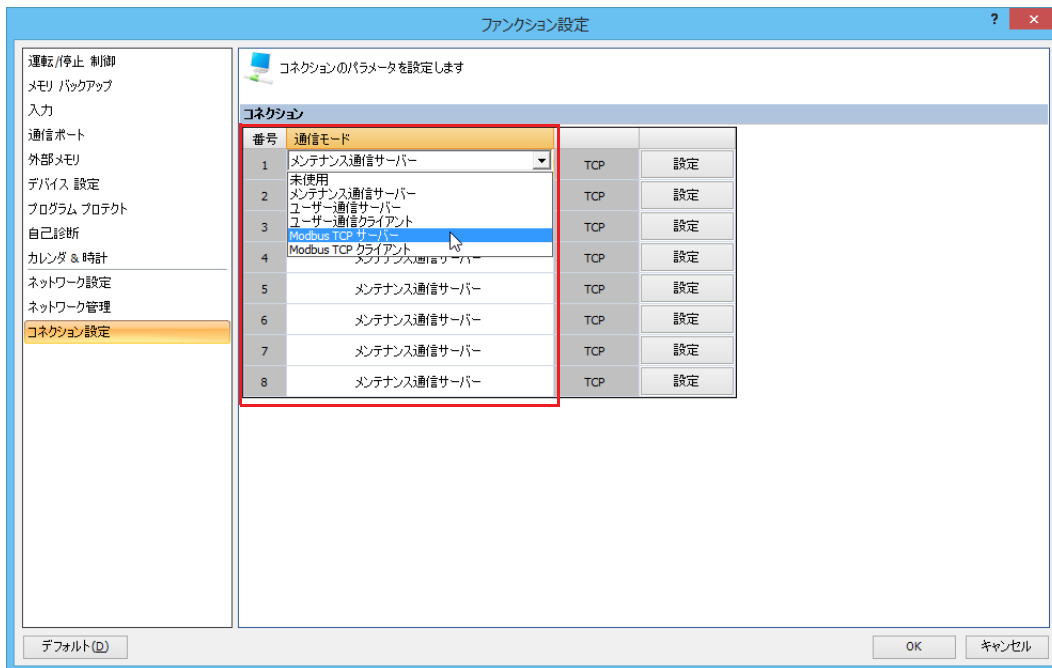
*2 ご使用の機種種のデバイス範囲内でアクセスしてください。

WindLDR の設定

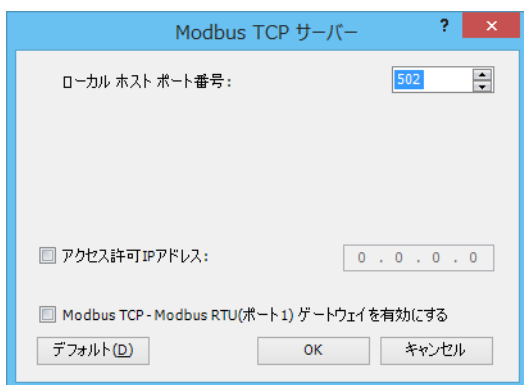
FC6A 形を Modbus TCP サーバーとして使用するには、[ファンクション設定] で設定を行ったユーザープログラムを FC6A 形にダウンロードします。

●操作手順

1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するコネクション番号の [通信モード] をクリックし、“Modbus TCP 通信サーバー” を選択します。
[Modbus TCP サーバー] ダイアログボックスが表示されます。



3. ローカルホストポート番号、アクセス許可 IP アドレスを設定します。



4. [OK] ボタンをクリックします。
5. ユーザープログラムを FC6A 形にダウンロードします。

以上で、Modbus TCP サーバーの設定が完了します。

設定内容

Modbus TCP サーバー通信設定

ローカルホストポート番号

0～65535の間で設定してください。複数のコネクション番号で、同一のローカルホストポート番号を設定することが可能です。同一のポート番号を設定したコネクションの数だけ、Modbus TCP クライアントを受け入れることが可能になります。

アクセス IP 許可アドレス

アクセスを許可する機器の IP アドレスを設定することができます。アクセス許可 IP アドレスを設定すると、設定した IP アドレスを持つ機器だけが、FC6A 形とコネクションを確立して Modbus TCP サーバーと通信できます。複数のコネクション番号で、同一のローカルホストポート番号を設定している場合は、すべてのアクセス許可 IP アドレスの設定が反映されます。アクセス許可 IP アドレスの設定を行わないコネクションを含む場合は、任意のアクセスを許可します。

- 例 1) 同一のローカルホストポート番号を設定したコネクション 1 とコネクション 2 において、どちらもアクセス許可 IP アドレスの設定を行わない場合は、任意の IP アドレスのクライアントからのアクセスを合計 2 つ受け入れます。
- 例 2) 同一のローカルホストポート番号を設定したコネクション 1 とコネクション 2 において、アクセス許可 IP アドレスの設定をそれぞれ 192.168.1.101 と 192.168.1.102 に設定した場合は、192.168.1.101 と 192.168.1.102 からのアクセスを合計 2 つ受け入れます。
- 例 3) 同一のローカルホストポート番号を設定したコネクション 1 とコネクション 2 において、コネクション 1 ではアクセス許可 IP アドレスを 192.168.1.101 に設定し、コネクション 2 ではアクセス許可 IP アドレスの設定を行わない場合は、任意の IP アドレスのクライアントからのアクセスを合計 2 つ受け入れます。

Modbus TCP 通信フォーマット

Modbus TCP 通信フォーマットは、Modbus TCP ヘッダに Modbus RTU 通信プロトコルフォーマットの CRC を削除したものを付加したフォーマットになります。Modbus RTU 通信プロトコルのフォーマットについては、「Modbus RTU 通信のフレームフォーマット」(6-14 頁)を参照してください。

Modbus TCP 通信フォーマット

トランザクション ID	プロトコル ID	メッセージ長(バイト)	ユニット ID	機能コード	内容
2 バイト	2 バイト	2 バイト	1 バイト	1 バイト	N バイト

Modbus TCP ヘッダ

RTU 通信フォーマット

"Idle"	局番	機能コード	内容	CRC	"Idle"
3.5 文字	1 バイト	1 バイト	N バイト	2 バイト	3.5 文字

トランザクション ID

Modbus TCP サーバーはクライアントからのトランザクション ID をそのまま返します。クライアント機器は、どのトランザクションに対する応答なのか確認することができます。特に確認を行わない場合は 0 を入れます。

プロトコル ID

Modbus TCP プロトコルを示す番号で、0 となります。

メッセージ長

以降に続くメッセージの長さをバイト単位で表します。

ユニット ID

機器を識別するための ID です。Modbus TCP サーバーのスレーブ番号を格納します。FC6A 形の Modbus TCP サーバーでは、受信したリクエストのユニット ID が 0 以外の場合は、リクエストを受け入れて処理します。ユニット ID が 0 の場合は、ブロードキャスト通信として処理し、Modbus TCP クライアントへは応答を返しません。

機能コード

読み出し、書き込みなどの機能の番号です。

内容

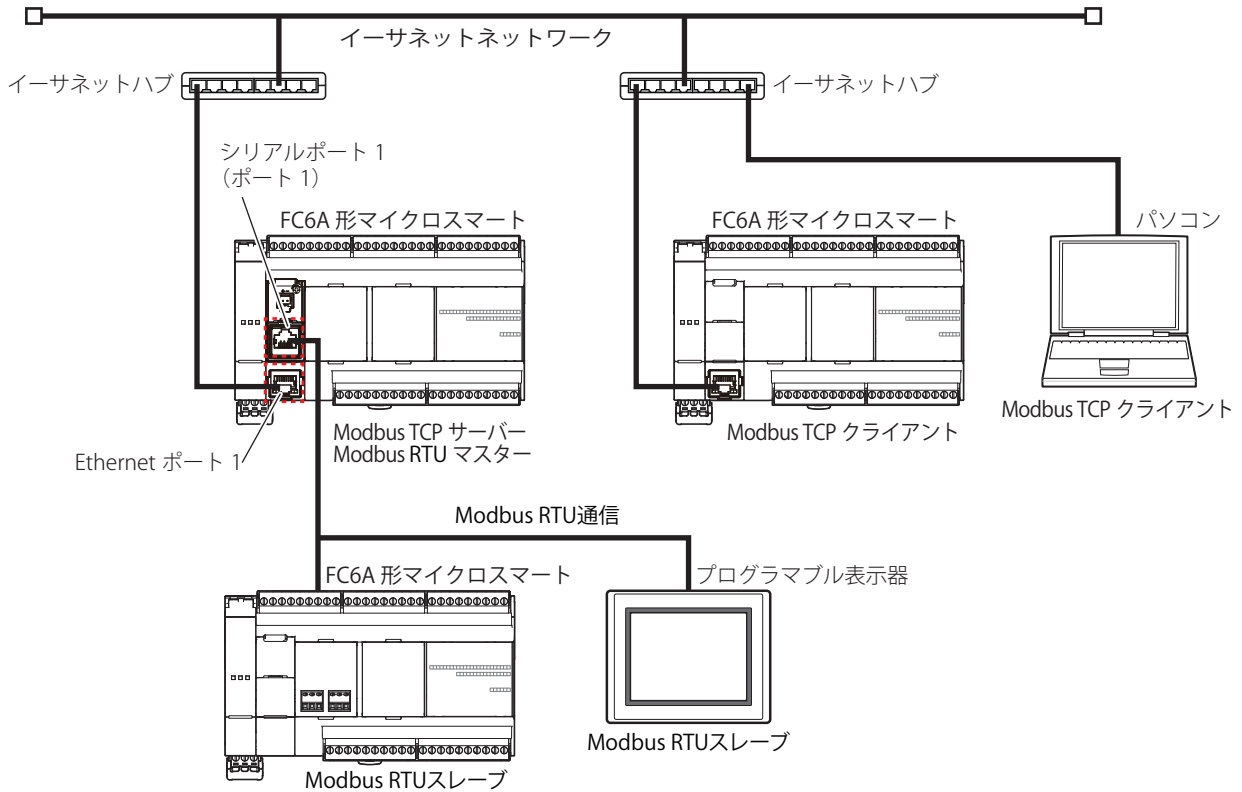
各処理に必要なデータです。

Modbus RTU パススルー機能

Modbus RTU パススルー機能は、Modbus TCP ネットワークに接続した Modbus TCP クライアントが、Modbus RTU ネットワークに接続した Modbus RTU スレーブへアクセスするための機能です。Modbus RTU パススルー機能は、All-in-One CPU モジュールおよび Plus CPU モジュールで使用できます。CAN J1939 All-in-One CPU モジュールでは使用できません。

All-in-One CPU モジュールの場合は、Ethernet ポート 1 に接続した Modbus TCP クライアント機器から、FC6A 形のシリアルポート 1 に接続した Modbus RTU スレーブ機器のデバイスの情報の読み出しや書き込みが行えます。Modbus RTU パススルー機能は、All-in-One CPU モジュールの Ethernet ポート 1、ポート 1 間でのみ対応しています。

Plus CPU モジュールの場合は、Ethernet ポート 1 および 2 に接続した Modbus TCP クライアント機器から、Plus CPU モジュールのポート 1 ～ 33 に接続した Modbus RTU スレーブ機器を選択して、デバイスの情報の読み出しや書き込みが行えます。Modbus RTU パススルー機能は、Plus CPU モジュールの Ethernet ポート 1 および 2 とポート 1 ～ 33 の間で対応しています。



Modbus RTU パススルー機能での Modbus TCP 通信フォーマット

Modbus RTU パススルー機能では、Modbus TCP サーバーで受けた Modbus TCP 通信フレームから Modbus TCP ヘッダーを削除し、「ユニット ID」「機能コード」「内容」のフレームに CRC を付加したフレームをユニット ID で設定された Modbus RTU スレーブに送信します。

Modbus RTU パススルー機能での Modbus TCP 通信フォーマットは、次のとおりです。

Modbus TCP 通信フォーマット

トランザクション ID	プロトコル ID	メッセージ長(バイト)	ユニット ID	機能コード	内容
2 バイト	2 バイト	2 バイト	1 バイト	1 バイト	N バイト

Modbus TCP ヘッダ

RTU 通信フォーマット

"Idle"	局番	機能コード	内容	CRC	"Idle"
3.5 文字	1 バイト	1 バイト	N バイト	2 バイト	3.5 文字

トランザクション ID

Modbus TCP サーバーはクライアントからのトランザクション ID をそのまま返します。クライアント機器は、どのトランザクションに対する応答なのか確認することができます。特に確認を行わない場合は 0 を入れます。

プロトコル ID

Modbus TCP プロトコルを示す番号で、0 となります。

メッセージ長

以降に続くメッセージの長さをバイト単位で表します。

ユニット ID

機器を識別するための ID です。Modbus RTU スレーブのスレーブ番号を格納します。ユニット ID 以降を Modbus RTU ネットワークへパススルーします。

ユニット ID に 255 を設定した場合は、Modbus RTU スレーブにはパススルーせず、Modbus TCP サーバー（自局）で処理します

機能コード

読み出し、書き込みなどの機能の番号です。

内容

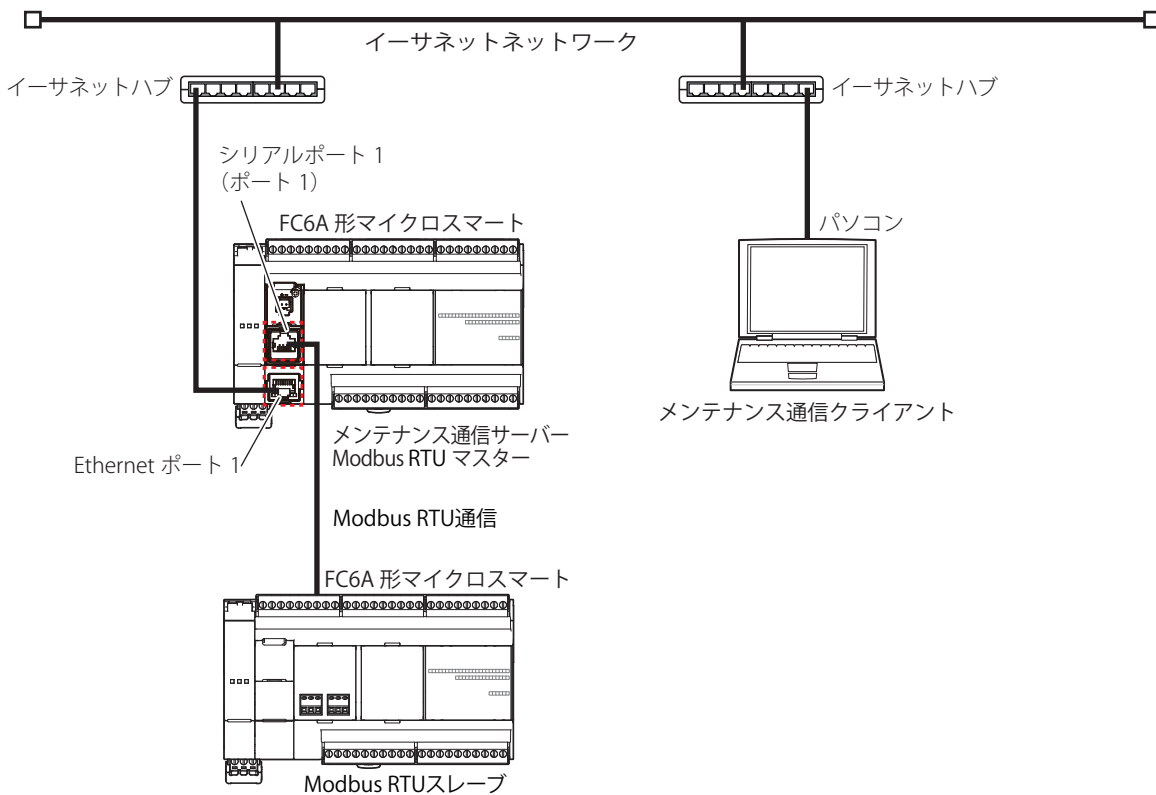
各処理に必要なデータです。

Modbus RTU パススルー機能の仕様

対応通信ポート	イーサネット通信	All-in-One CPUモジュール：Ethernetポート1 Plus CPUモジュール：Ethernetポート1および2
	シリアル通信	All-in-One CPUモジュール：ポート1 Plus CPUモジュール：ポート1～33（1つのポートを選択できます。）
通信モード	Modbus TCP サーバー（スレーブ）とModbus RTU マスター	
有効 / 無効設定	[ファンクション設定] で設定します。 ※有効にできるサーバーコネクション数に制限はありません。 ※メンテナンス通信サーバー経由のパススルーと併用できます。	
パススルー先の設定	<p>パススルーするModbus RTU スレーブは、Modbus TCP通信フレームのユニットIDで設定します。</p> <ul style="list-style-type: none"> Modbus RTUパススルー機能が有効な場合 ユニットID（0～254）： Modbus RTUパススルーを行います。 ユニットID（255）： Modbus TCPサーバーで処理します Modbus RTUパススルー機能が無効な場合 ユニットID（0～255）： Modbus TCPサーバーで処理します。 	
通信できるバイト数	512バイト ※512バイトを超えるデータ（フレーム）をパススルーすることはできません。512バイト以上のデータを受信した場合、それまでの受信データを含め破棄します。	
パススルー動作	<p>ユニットID（スレーブ局番）が255でないとき、通信ポートにパススルー要求を発行しModbus RTUスレーブからの応答を待ちます。ユニットIDが255のときは、自局への通信として解析しModbus TCPクライアントへ応答を返します。</p> <p>Modbus TCPサーバーからのパススルー要求があれば、Modbus RTUマスタのリクエスト処理要求より優先して処理を行います。</p>	

メンテナンス通信サーバーを使用した Modbus RTU パススルー

FC6A 形をメンテナンス通信サーバーにして、Modbus RTU ネットワークに接続した Modbus RTU スレーブへアクセスすることもできます。遠隔のパソコンにインストールされた WindLDR やデータ ファイル マネージャーなどのメンテナンス通信クライアントから、イーサネットに接続された FC6A 形 (Modbus RTU マスター) を経由して、Modbus RTU スレーブに設定された FC6A 形 (Modbus RTU スレーブ) のデバイスの読出しや書き込み、プロジェクトデータのダウンロード/アップロードができます。



Modbus RTU パススルー機能の設定

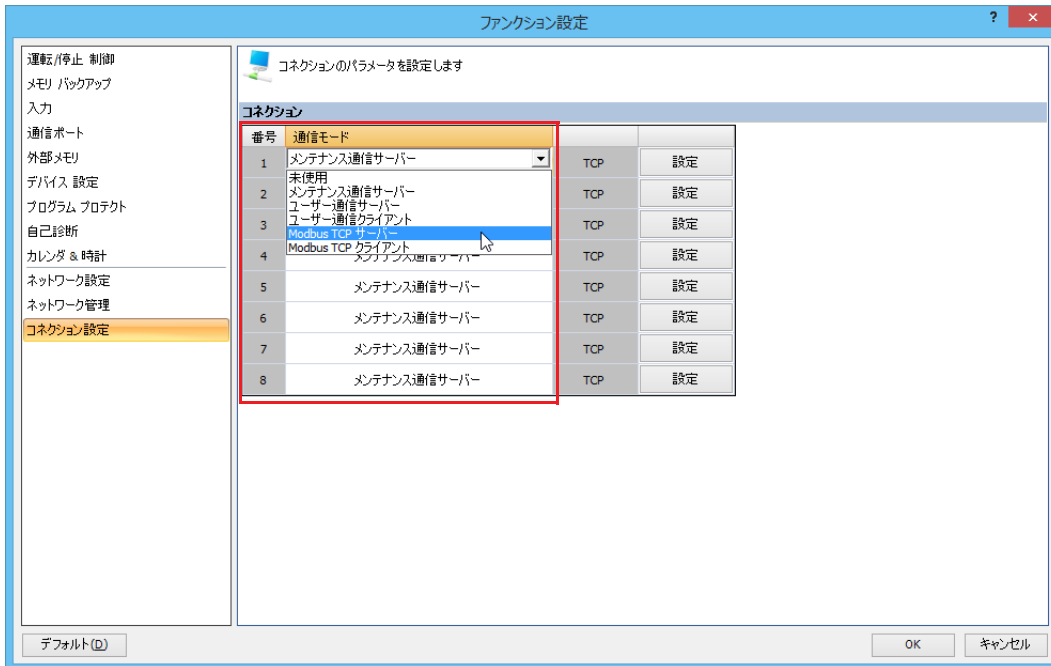
WindLDR の設定

Modbus RTU パススルー機能を使用するには、[ファンクション設定] で設定を行ったユーザープログラムを FC6A 形にダウンロードします。

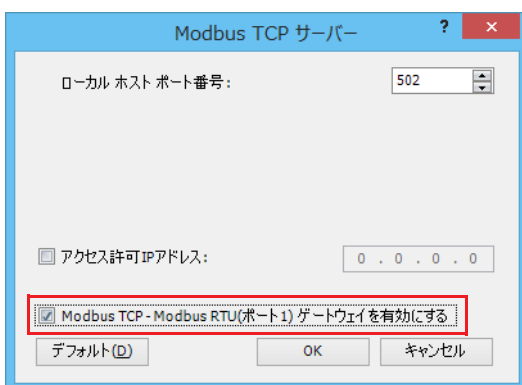
●操作手順

All-in-One CPU モジュールの場合

1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するコネクション番号の [通信モード] をクリックし、“Modbus TCP サーバー” を選択します。
[Modbus TCP サーバー] ダイアログボックスが表示されます。

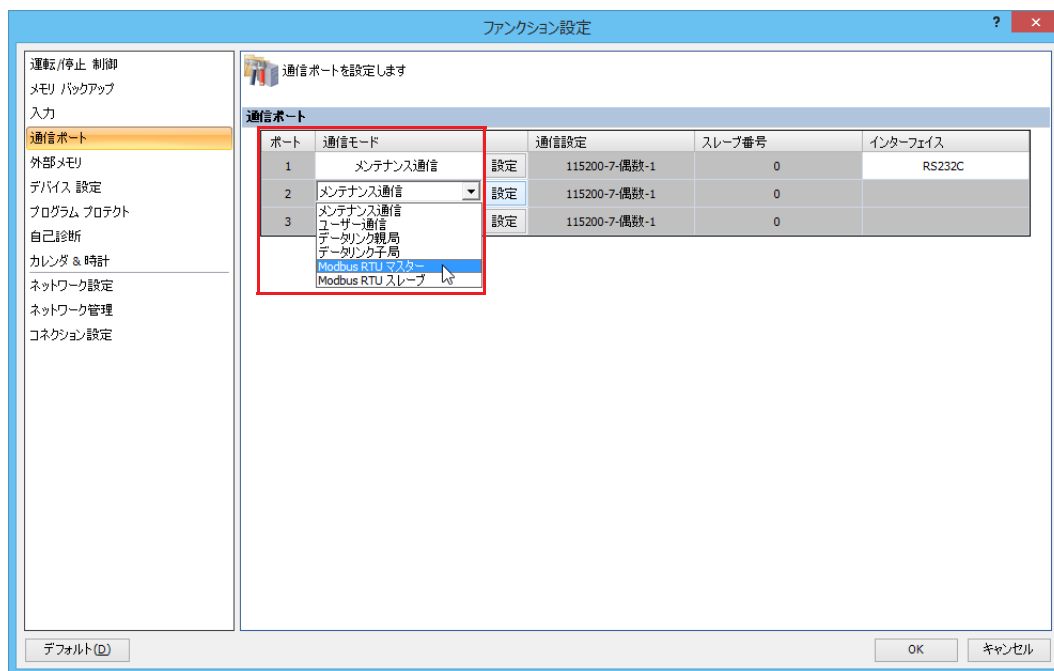


3. [Modbus TCP - Modbus RTU (ポート 1) ゲートウェイを有効にする] チェックボックスをオンにし、ローカルホストポート番号、アクセス許可 IP アドレスを設定します。
パススルー機能を使用するポートのローカルホストポート番号は、他のサーバーコネクションと異なる番号を設定してください。



4. [OK] ボタンをクリックします。

5. [通信ポート] タブをクリックします。
6. ポート 1 の [通信モード] をクリックし、“Modbus RTU マスター” を選択します。
[Modbus RTU マスターリクエストテーブル] ダイアログボックスが表示されます。

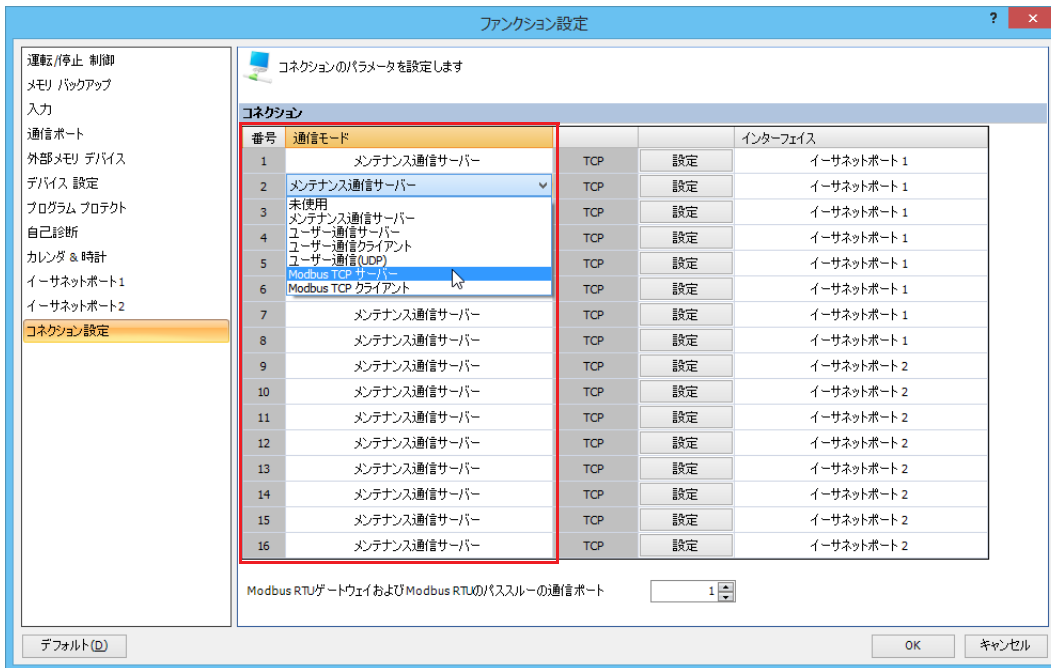


7. [Modbus RTU マスターリクエストテーブル] ダイアログボックスの各項目を設定します。
詳細は、「Modbus RTU マスター」(6-2 頁) を参照してください。
8. ユーザープログラムを FC6A 形にダウンロードします。

以上で、Modbus RTU パススルー機能を使用するための設定が完了します。

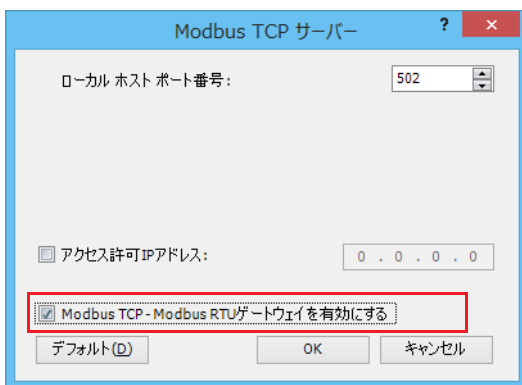
Plus CPU モジュールの場合

1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するコネクション番号の [通信モード] をクリックし、“Modbus TCP サーバー” を選択します。
[Modbus TCP サーバー] ダイアログボックスが表示されます。



メンテナンス通信サーバーを使用した Modbus RTU パススルーの場合は、“メンテナンス通信サーバー” を選択します。
[メンテナンス通信サーバー] ダイアログボックスが表示されます。

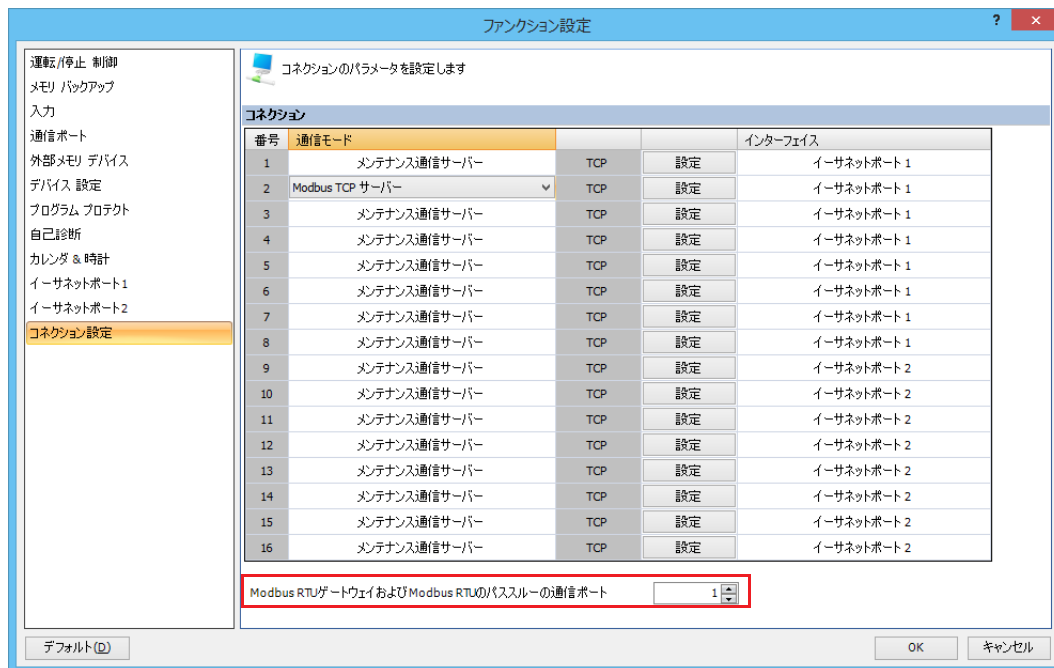
3. [Modbus TCP - Modbus RTU ゲートウェイを有効にする] チェックボックスをオンにし、ローカルホストポート番号、アクセス許可 IP アドレスを設定します。
パススルー機能を使用するポートのローカルホストポート番号は、他のサーバーコネクションと異なる番号を設定してください。



[メンテナンス通信サーバー] ダイアログボックスの場合は、[Modbus RTU のパススルーを有効にする] チェックボックスをオンにし、ローカルホストポート番号、アクセス許可 IP アドレスを設定します。

4. [OK] ボタンをクリックします。

5. [Modbus RTU ゲートウェイおよび Modbus RTU のパススルーの通信ポート] で、使用するポート番号 (1 ~ 33) を設定します。



6. ユーザープログラムを FC6A 形にダウンロードします。

以上で、Modbus RTU パススルー機能を使用するための設定が完了します。

第7章 データリンク通信

この章では、分散制御システムに有効なデータリンク機能について説明します。

データリンク通信の概要

FC6A形のポートをRS485通信に設定することで、データリンク通信が使用できます。

データリンク通信では、1台のFC6A形（親局）と最大31台のFC6A形（子局）を接続し、親局と子局の間でデータの交換ができます。親局は子局ごとに子局への送信データ6ワード、子局からの受信データ6ワード、通信ステータス/エラー1ワードの計13ワードのデータレジスタを持ちます。子局は親局への送信データ6ワード、親局からの受信データ6ワード、通信ステータス/エラー1ワードの計13ワードのデータレジスタを持ちます。親局-子局間の通信処理は、ユーザープログラムの処理とは非同期で行われ、END処理で、データレジスタ内容の更新（データ更新）を行います。

FC6A形のデータリンク通信は、FC4A形やFC5A形とのデータリンク通信と互換性があります。

1台のCPUモジュールで同時に使用できるデータリンク通信は、親局または子局いずれか1つのみです。

データリンク通信で使用する通信ポート

対応する機種と通信ポートは、次のとおりです。

通信ポート	All-in-One CPU モジュール			CAN J1939 All-in-One CPU モジュール	Plus CPU モジュール	
	16点タイプ	24点タイプ	40点タイプ		Plus16点タイプ	Plus32点タイプ
ポート1	○	○	○	—	○*2	○*2
ポート2	○*1	○*1	○*1	○*1	○*2	○*2
ポート3	—	—	○*1	○*1	○*3	○*3
ポート4～33	○（最大3台）*4			○（最大15台）*5		

*1 RS232C通信カートリッジまたはRS485通信カートリッジを接続した場合

*2 カートリッジベースモジュールを接続し、RS232C通信カートリッジまたはRS485通信カートリッジを接続した場合

*3 HMIモジュールを接続し、RS232C通信カートリッジまたはRS485通信カートリッジを接続した場合

*4 All-in-One CPUモジュールおよびCAN J1939 All-in-One CPUモジュールに通信モジュールを最大3台接続してポート4～9を増設できます。

*5 Plus CPUモジュールに通信モジュールを最大15台接続してポート4～33を増設できます。

仕様

項目	内容
電気特性	EIA RS485規格準拠
通信速度	19200bps/ 38400bps/ 57600bps
調歩同期方式	スタートビット：1 データビット長：7 パリティ：偶数 ストップビット：1
最大ケーブル長	総延長200m
最大子局台数	31台
最大データ数（ワード）	送信：186/受信：186（1局当たり送信：6/受信：6）

データ更新について

項目	説明
スキャンタイムへの影響	親局-子局間の通信処理は、ユーザープログラムとは非同期に行われますので、スキャンタイムに影響を与えることはありません。
データ更新タイミング	親局、子局ともにデータ更新は、END処理で行われます。更新タイミングは通信完了フラグで確認できます。
親局対応機種	FC6A形（FC4A形/FC5A形）
子局対応機種	FC6A形（FC4A形/FC5A形）

親局は送信ワード数分のデータを子局へ送信し、それを受信した子局は受信ワード数分のデータを親局へ送信し、データの交換を行います。データ更新は親局、子局ともにEND処理で行います。親局は、1回の通信処理で1子局のみデータの交換ができます。子局を31台接続している場合、すべての子局とデータを交換するためには31回の通信処理が必要となります。親局は通信処理が完了した子局について、通信完了直後の1スキャンのみ通信完了リレーをONします。

デバイス割付

親局側データリンク

子局1	D0900~D0905	子局1への送信データ
	D0906~D0911	子局1からの受信データ
子局2	D0912~D0917	子局2への送信データ
	D0918~D0923	子局2からの受信データ
子局3	D0924~D0929	子局3への送信データ
	D0930~D0935	子局3からの受信データ
子局4	D0936~D0941	子局4への送信データ
	D0942~D0947	子局4からの受信データ
子局5	D0948~D0953	子局5への送信データ
	D0954~D0959	子局5からの受信データ
子局6	D0960~D0965	子局6への送信データ
	D0966~D0971	子局6からの受信データ
子局7	D0972~D0977	子局7への送信データ
	D0978~D0983	子局7からの受信データ
子局8	D0984~D0989	子局8への送信データ
	D0990~D0995	子局8からの受信データ
子局9	D0996~D1001	子局9への送信データ
	D1002~D1007	子局9からの受信データ
子局10	D1008~D1013	子局10への送信データ
	D1014~D1019	子局10からの受信データ
子局11	D1020~D1025	子局11への送信データ
	D1026~D1031	子局11からの受信データ
子局12	D1032~D1037	子局12への送信データ
	D1038~D1043	子局12からの受信データ
子局13	D1044~D1049	子局13への送信データ
	D1050~D1055	子局13からの受信データ
子局14	D1056~D1061	子局14への送信データ
	D1062~D1067	子局14からの受信データ
子局15	D1068~D1073	子局15への送信データ
	D1074~D1079	子局15からの受信データ
子局16	D1080~D1085	子局16への送信データ
	D1086~D1091	子局16からの受信データ
子局17	D1092~D1097	子局17への送信データ
	D1098~D1103	子局17からの受信データ
子局18	D1104~D1109	子局18への送信データ
	D1110~D1115	子局18からの受信データ
子局19	D1116~D1121	子局19への送信データ
	D1122~D1127	子局19からの受信データ
子局20	D1128~D1133	子局20への送信データ
	D1134~D1139	子局20からの受信データ
子局21	D1140~D1145	子局21への送信データ
	D1146~D1151	子局21からの受信データ
子局22	D1152~D1157	子局22への送信データ
	D1158~D1163	子局22からの受信データ
子局23	D1164~D1169	子局23への送信データ
	D1170~D1175	子局23からの受信データ

子局24	D1176~D1181	子局24への送信データ
	D1182~D1187	子局24からの受信データ
子局25	D1188~D1193	子局25への送信データ
	D1194~D1199	子局25からの受信データ
子局26	D1200~D1205	子局26への送信データ
	D1206~D1211	子局26からの受信データ
子局27	D1212~D1217	子局27への送信データ
	D1218~D1223	子局27からの受信データ
子局28	D1224~D1229	子局28への送信データ
	D1230~D1235	子局28からの受信データ
子局29	D1236~D1241	子局29への送信データ
	D1242~D1247	子局29からの受信データ
子局30	D1248~D1253	子局30への送信データ
	D1254~D1259	子局30からの受信データ
子局31	D1260~D1265	子局31への送信データ
	D1266~D1271	子局31からの受信データ

親局のデータレジスタは、子局が未接続の場合には通常のデータレジスタとして使用できます。

子局側データリンク

子局データ	D0900~D0905	親局への送信データ
	D0906~D0911	親局からの受信データ

子局のデータレジスタ D0912 ~ D1271 は、通常のデータレジスタとして使用できます。

親局側データリンク通信ステータス

親局側データリンク通信ステータスは、[ファンクション設定] で設定したデータレジスタを先頭として、子局数だけ格納されます。

子局側データリンク通信ステータスは、[ファンクション設定] で設定したデータレジスタに格納されます。

以下は、D8069 を設定したときのデバイス割付です。この設定を行うと、FC4A 形や FC5A 形のポート 2 と同じ割付となります。

子局1	D8069	子局1通信ステータス/エラー
子局2	D8070	子局2通信ステータス/エラー
子局3	D8071	子局3通信ステータス/エラー
子局4	D8072	子局4通信ステータス/エラー
子局5	D8073	子局5通信ステータス/エラー
子局6	D8074	子局6通信ステータス/エラー
子局7	D8075	子局7通信ステータス/エラー
子局8	D8076	子局8通信ステータス/エラー
子局9	D8077	子局9通信ステータス/エラー
子局10	D8078	子局10通信ステータス/エラー
子局11	D8079	子局11通信ステータス/エラー
子局12	D8080	子局12通信ステータス/エラー
子局13	D8081	子局13通信ステータス/エラー
子局14	D8082	子局14通信ステータス/エラー
子局15	D8083	子局15通信ステータス/エラー
子局16	D8084	子局16通信ステータス/エラー
子局17	D8085	子局17通信ステータス/エラー
子局18	D8086	子局18通信ステータス/エラー
子局19	D8087	子局19通信ステータス/エラー
子局20	D8088	子局20通信ステータス/エラー
子局21	D8089	子局21通信ステータス/エラー
子局22	D8090	子局22通信ステータス/エラー

子局23	D8091	子局23通信ステータス/エラー
子局24	D8092	子局24通信ステータス/エラー
子局25	D8093	子局25通信ステータス/エラー
子局26	D8094	子局26通信ステータス/エラー
子局27	D8095	子局27通信ステータス/エラー
子局28	D8096	子局28通信ステータス/エラー
子局29	D8097	子局29通信ステータス/エラー
子局30	D8098	子局30通信ステータス/エラー
子局31	D8099	子局31通信ステータス/エラー

子局側データリンク通信ステータス

子局側データリンク通信ステータスは、[ファンクション設定] で設定したデータレジスタに格納されます。

以下は、D8069 を設定したときのデバイス割付です。この設定を行うと、FC4A 形や FC5A 形のポート 2 と同じ割付となります。

子局n	D8069	通信ステータス/エラー
-----	-------	-------------

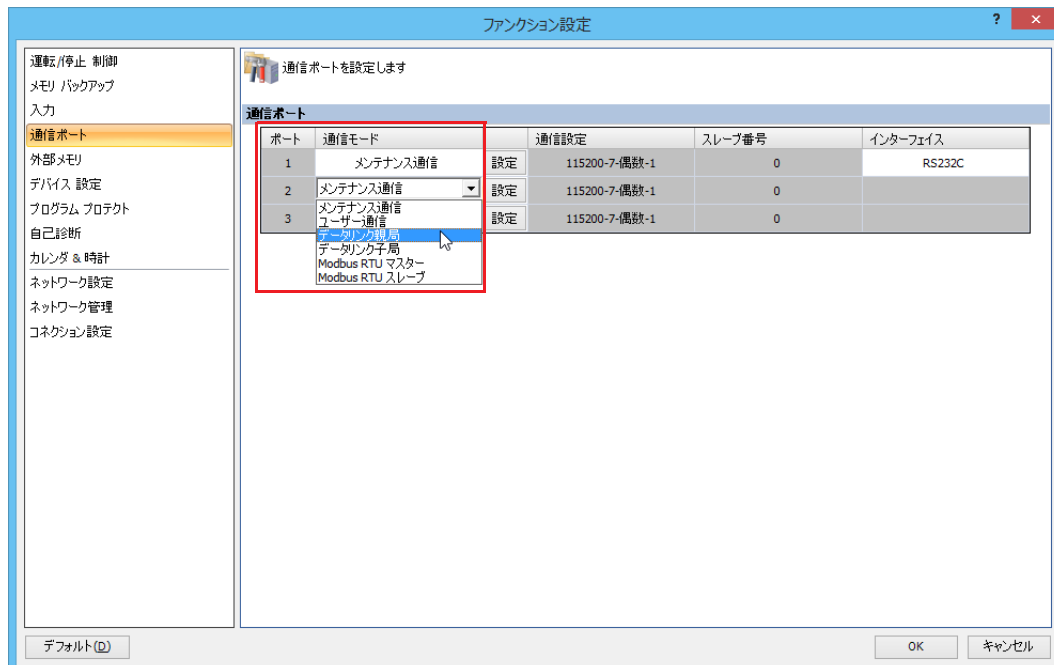
親局側データリンク通信完了リレー

子局1	M8080	子局1通信完了リレー
子局2	M8081	子局2通信完了リレー
子局3	M8082	子局3通信完了リレー
子局4	M8083	子局4通信完了リレー
子局5	M8084	子局5通信完了リレー
子局6	M8085	子局6通信完了リレー
子局7	M8086	子局7通信完了リレー
子局8	M8087	子局8通信完了リレー
子局9	M8090	子局9通信完了リレー
子局10	M8091	子局10通信完了リレー
子局11	M8092	子局11通信完了リレー
子局12	M8093	子局12通信完了リレー
子局13	M8094	子局13通信完了リレー
子局14	M8095	子局14通信完了リレー
子局15	M8096	子局15通信完了リレー
子局16	M8097	子局16通信完了リレー
子局17	M8100	子局17通信完了リレー
子局18	M8101	子局18通信完了リレー
子局19	M8102	子局19通信完了リレー
子局20	M8103	子局20通信完了リレー
子局21	M8104	子局21通信完了リレー
子局22	M8105	子局22通信完了リレー
子局23	M8106	子局23通信完了リレー
子局24	M8107	子局24通信完了リレー
子局25	M8110	子局25通信完了リレー
子局26	M8111	子局26通信完了リレー
子局27	M8112	子局27通信完了リレー
子局28	M8113	子局28通信完了リレー
子局29	M8114	子局29通信完了リレー
子局30	M8115	子局30通信完了リレー
子局31	M8116	子局31通信完了リレー
全子局	M8117	全子局通信完了リレー

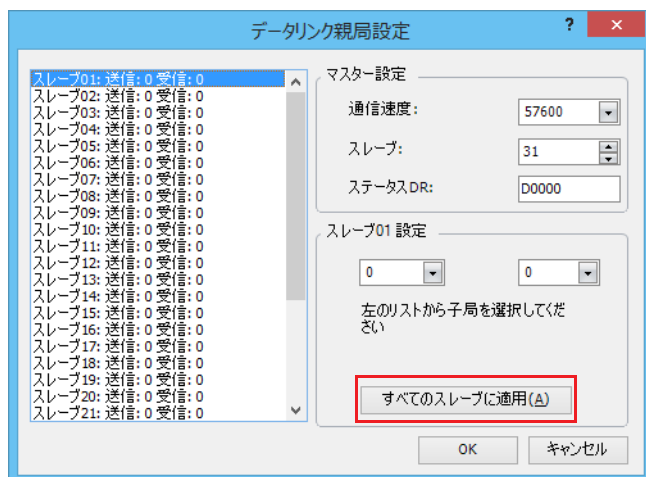
WindLDR の設定

●親局の操作手順

1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するポートの [通信モード] をクリックし、“データリンク親局” を選択します。
[データリンク親局設定] ダイアログボックスが表示されます。



3. マスタ設定の通信速度、スレーブ、ステータス DR を設定します。
4. リスト内のスレーブ番号をクリックして選択し、送信ワード数と受信ワード数を設定します。
リストの子局すべてに同じ設定をする場合は、[すべてのスレーブに適用] ボタンをクリックします。



送信ワード数： 親局が子局へ送信するデータのワード数

受信ワード数： 親局が子局から受信するデータのワード数

5. [OK] ボタンをクリックします。
6. ユーザープログラムを作成します。
7. ユーザープログラムを転送します。

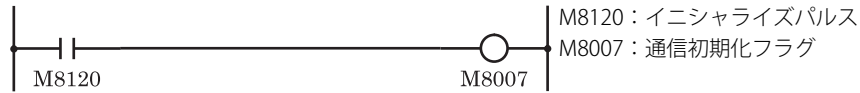
以上で、設定が完了します。



データリンク通信を使用する場合、子局の電源を先に ON したあと、親局の電源を ON する必要があります。親局の電源を ON したあとに子局の電源を ON した場合、親局は子局を認識できません。親局が子局を認識できない場合、データリンクの初期化を行う必要があります。データリンクの初期化を行うには、WindLDR で [データリンク初期化] ボタンをクリックする、またはユーザープログラムで特殊内部リレー M8007 を ON します。

1. [オンライン] タブの [モニタ] で [モニタ] をクリックし、[モニタ開始] をクリックします。
モニタモードになります。
2. [オンライン] タブの [PLC 本体] で [初期化] をクリックし、[データリンク初期化] をクリックします。
子局が親局に認識されます。

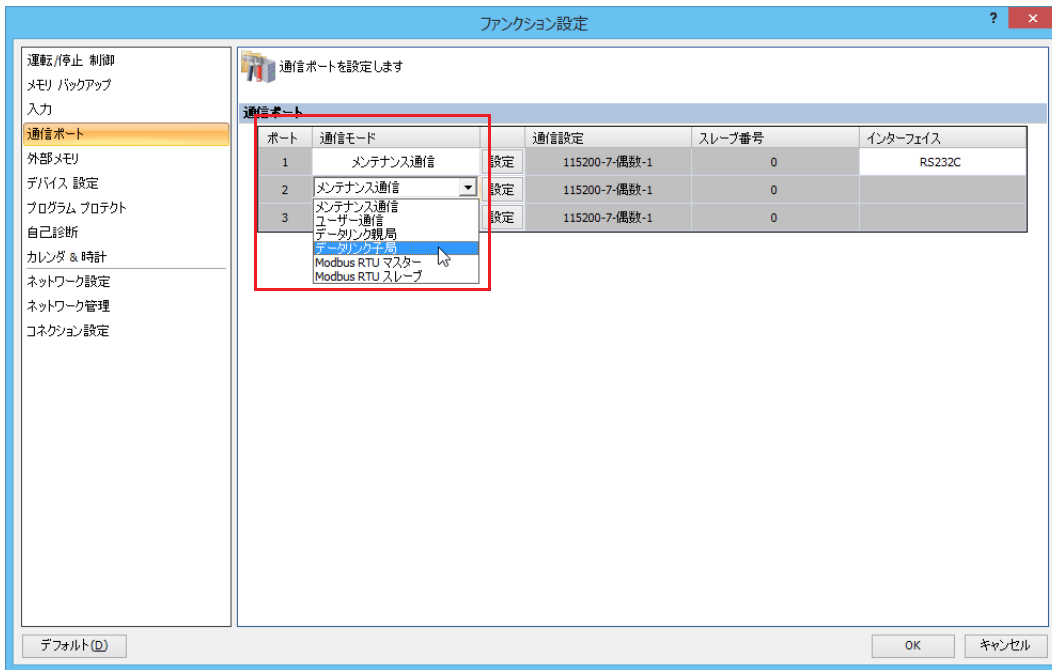
WindLDR のメンテナンス通信が行えない場合、親局のユーザープログラムに次のプログラムを入れてください。親局を一度 STOP 後、RUN させることで子局を認識できます。



RUN 開始時の 1 スキャンのみ M8007 が ON となり、データリンクの初期化が行われます。

●子局の操作手順

1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. 使用するポート番号の [通信モード] をクリックし、“データリンク子局”を選択します。
[データリンクスレーブ設定] ダイアログボックスが表示されます。



3. データリンク局番、通信速度を設定します。
 [データリンク局番] で“定数”または“データレジスタ”を選択します。“定数”を選択した場合はデータリンク局番を設定します。“データレジスタ”を選択した場合は、特殊データレジスタにデータリンク局番を書き込みます。

4. [OK] ボタンをクリックします。
 5. ユーザープログラムを作成します。
 6. ユーザープログラムを転送します。

以上で、設定が完了します。



[データリンクスレーブ設定] ダイアログボックスの [データリンク局番] で“データレジスタ”を選択した場合、以下の特殊データレジスタにデータリンク局番を書き込むことで、ユーザープログラムのダウンロードを行わずに子局のデータリンク局番を変更することができます。

通信ポート	特殊データレジスタ
ポート1	D8100
ポート2	D8102
ポート3	D8103
ポート4～9 ^{*1}	D8040～D8045
ポート10～33 ^{*1*2}	D8735～D8758

*1 通信モジュール使用時

*2 Plus CPU モジュールのみ

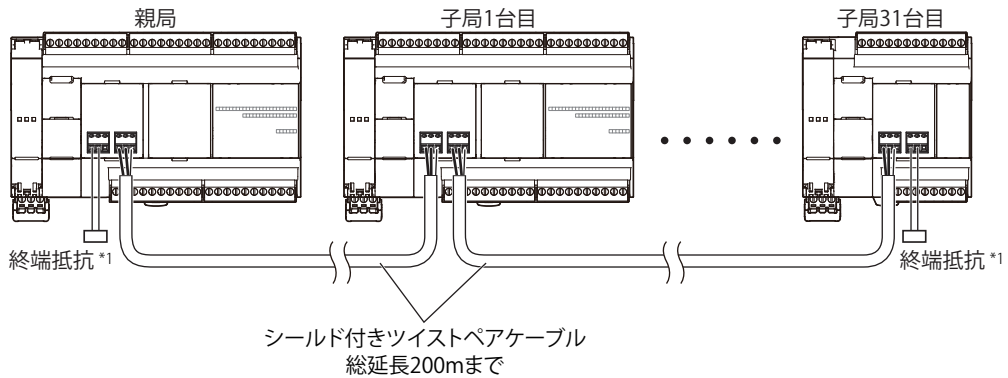
変更後は、親局の初期化および子局設定を正しく行ってください。
 親局の初期化は以下のいずれかの方法で行います。

- ① 親局の電源を切ったあと再投入する。
- ② 親局 M8007 をセットする。
- ③ [オンライン] タブの [PLC 本体] で [初期化] をクリックし、[データリンク初期化] をクリックする。

なお、データリンク局番の値は、1～31 が有効です。それ以外の値の場合、[ファンクション設定] で設定した局番が有効になります。

ユニットの接続例

各ユニット（親局および子局）のRS485ボード間をシールド付き3芯ケーブルで接続してください。



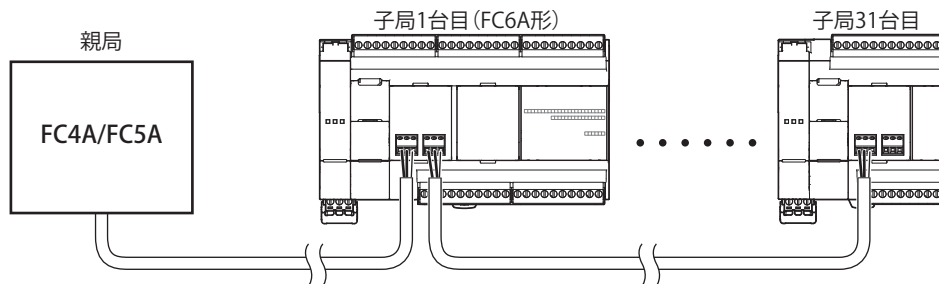
*1 通信品位が不安定な場合、特性インピーダンスに合わせた終端抵抗を両端に挿入してください。定格 1/2W 以上の抵抗を使用してください。

弊社製品とのリンク

FC6A形は、FC4A形やFC5A形と通信によりデータの交換（データリンク）ができます。

FC4A/FC5A とリンクする

FC4A/FC5A 側設定	FC6A 形側設定
送信データ6ワード 受信データ6ワード 19200bps/38400bps	データリンク子局を1～31に設定



第8章 J1939通信

この章では、CAN J1939 All-in-One CPU モジュールの J1939 通信について説明します。

本文中では、説明を簡潔にするために次の略語を使用しています。

本文中の略語	正式名称
CAN	Controller Area Network
ECU	Electronic Controller Unit
NIECU	Network Interconnection ECU
CA	Controller Application
PDU	Protocol Data Unit
PGN	Parameter Group Number
SA	Source Address
DA	Destination Address
SAE	Society of Automotive Engineers
SPN	Suspect Parameter Number

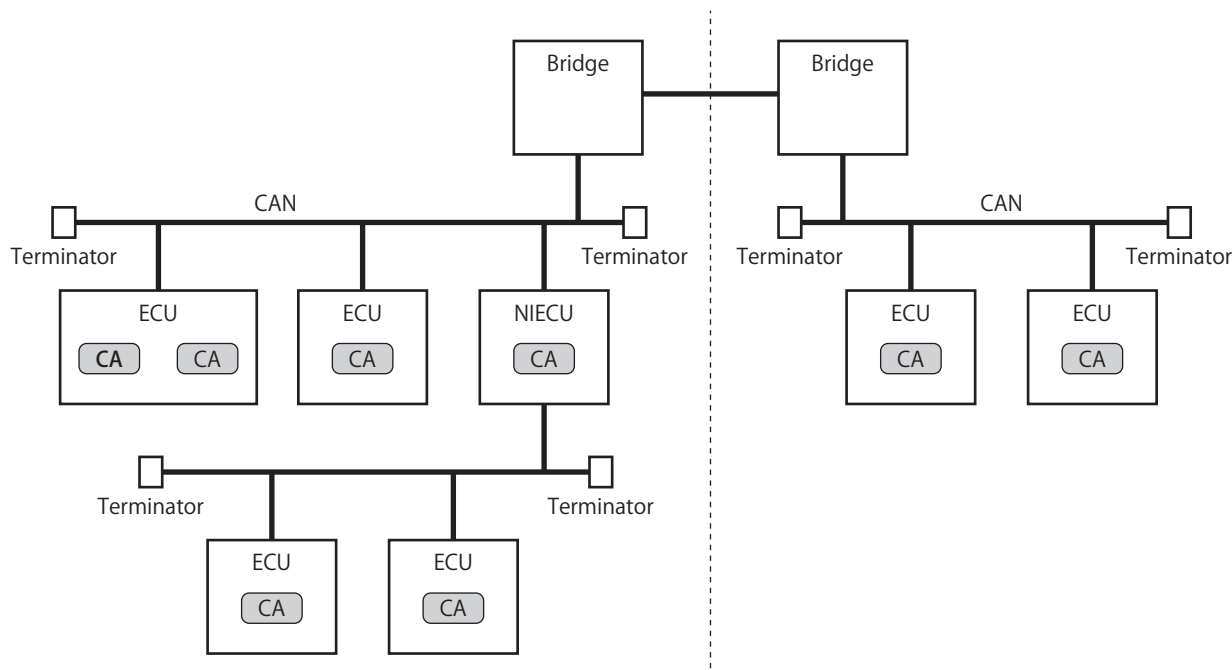
CAN による J1939 通信の概要

CAN J1939 All-in-One CPU モジュールは、SAE J1939 に基づいた CAN 通信に対応しています。CAN ポートを J1939 通信ネットワークへ接続することで、他の J1939 通信対応機器と通信できます。J1939 通信に関する詳細は、SAE J1939 の規格書を参照してください。

SAE J1939 の概要

J1939 通信ネットワーク

J1939 通信ネットワークは、ECU（エンジン、ブレーキなど）と NIECU（リピーター、ルーターなど）で構成され、各 ECU は 1 つ以上の CA を持ちます。CA には、0 ~ 253 までのアドレスと、64 ビットの固有の ID (NAME) が割り当てられており、アドレスを使用して CA 同士がメッセージの送受信を行います。また、CA 間でアドレスが競合した場合は、NAME で示された優先度に基づいて、競合の解決を行います。



アドレス

アドレスは 0 ~ 255 の範囲で定義されています。

ただし、254 (ヌルアドレス) はアドレス競合の結果、アドレスが取得できなかったときだけ使用されます。また、255 (グローバルアドレス) は特定の CA を指定せずに、すべての CA に対してブロードキャストでメッセージの送信をするときに使用されます。

アドレス	内容	詳細
0~253	CAに割り当てられるアドレス範囲	接続機器の種類に応じて、一部のアドレスは用途が決められています。詳細は、SAE J1939の規格書を参照してください。
254	ヌルアドレス	アドレスが競合した結果、0~253の範囲のアドレスが取得できなかったときに設定されます。
255	グローバルアドレス	ブロードキャストでメッセージを送信するときに使用します。

NAME (合計 64 ビット)

64 ビットの NAME は複数のフィールドに分けられます。一部のフィールドの内容は、規格で事前に定義されています。NAME は、同一の J1939 通信ネットワーク内では重複しない値に設定する必要があります。各ビットの定義の詳細は、SAE J1939 の規格書を参照してください。

フィールド名	ビット長	SAE 定義済み*1	項目
Arbitrary Address Capable	1 (上位)	—	動的アドレス対応
Industry Group	3	○	産業グループ
Vehicle System Instance	4	—	車両システムインスタンス
Vehicle System	7	○	車両システム
Reserved	1	○	予約済み (0 固定)
Function	8	○	機能
Function Instance	5	—	機能インスタンス
ECU Instance	3	—	ECUインスタンス
Manufacturer Code	11	○	メーカーコード
Identify Number	21 (下位)	—	IDナンバー

*1 “○”の項目は規格で定義されています。ご使用のネットワークと目的に合わせて設定してください。



ご使用のネットワークが不明な場合、各フィールドの最大値を NAME として設定してください。

CAN J1939 All-in-One CPU モジュールは、Arbitrary Address Capable ビットの値に応じて、CA とのアドレス競合を解決します。

Arbitrary Address Capable ビットが 0 の場合

CAN J1939 All-in-One CPU モジュールは固定アドレス割付の CA として動作します。自局アドレスと重複するアドレスを持つ CA が存在する場合、CAN J1939 All-in-One CPU モジュールの優先度が高ければ設定した自局アドレスでメッセージの送受信ができます。

CAN J1939 All-in-One CPU モジュールの優先度が低くければ自局アドレスを 254 (ヌルアドレス) とします。

Arbitrary Address Capable ビットが 1 の場合

CAN J1939 All-in-One CPU モジュールは動的アドレス割付の CA として動作します。自局アドレスと重複するアドレスを持つ CA があった場合、該当する CA と比べて CAN J1939 All-in-One CPU モジュールの優先度が低ければ、他のアドレスの取得を繰り返し試みます。自局アドレスが決定すれば、メッセージの送受信ができます。

最後まで自局アドレスが決定しなかった場合は、自局アドレスを 254 (ヌルアドレス) とします。

PGN

CA は、エンジン回転数やスイッチの ON/OFF 状態などのパラメータの情報をやり取りします。各パラメータは事前に定義され、固有の ID 番号 (SPN) が割り当てられています。また、パラメータは内容に応じて複数まとめられ、パラメータのまとまりごとに固有の ID 番号 (PGN) が割り当てられています。

CA は、PGN ごとにメッセージの送受信を行います。

例) PGN 1792 General Purpose Valve Pressure を構成する SPN*1

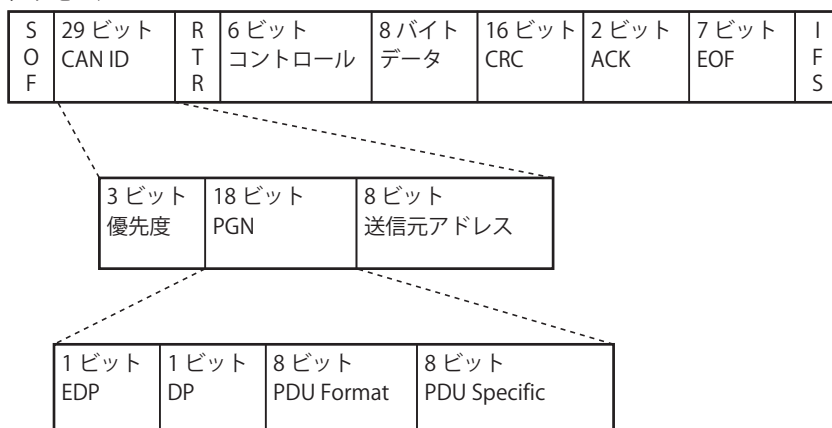
SPN	パラメータ名	データサイズ
4086	Valve Load Sense Pressure	2バイト
4087	Valve Pilot Pressure	1バイト
4088	Valve Assembly Load sense Pressure	2バイト
4089	Valve Assembly Supply Pressure	2バイト

*1 SAE J1939-71 より

メッセージ

CAは、29ビットのCAN IDを持つCANの拡張フォーマットのデータフレームを使用して、メッセージの送受信を行います。CAN IDは優先度、PGN、送信元アドレスで構成されます。さらにPGNは、EDP、DP、PDU Format (PDUF)、PDU Specific (PDUS)で構成されます。

メッセージ



PDU 1 フォーマット

PDUFが00h～EFhのとき、PDU1フォーマットと呼ばれる形式でPGNが定義されます。PGNは、EDP、DP、PDUFで定義され、1対1通信のためのPGNとして取り扱われます。このとき、PDUSには送信先アドレスが格納されます。ただし、PDUSに255（グローバルアドレス）を指定したときは、1対1通信ではなくブロードキャストのPGNとして取り扱われます。

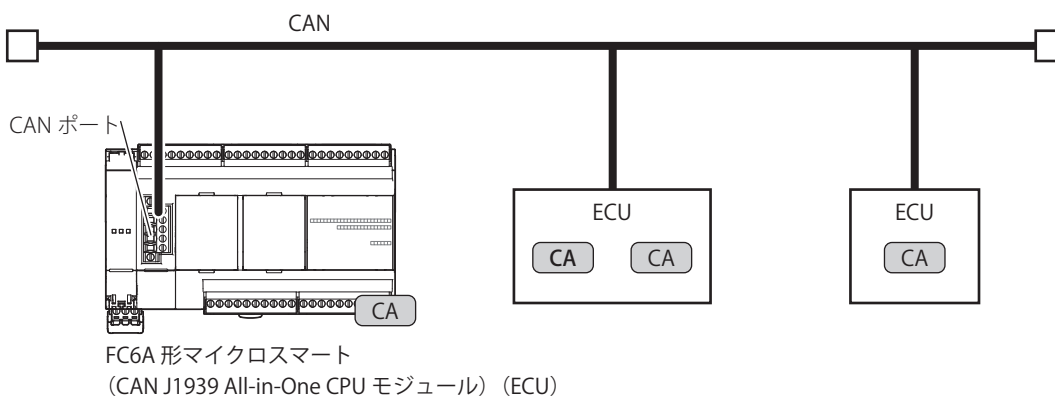
PDU 2 フォーマット

PDUFがF0h～FFhのとき、PDU2フォーマットと呼ばれる形式でPGNが定義されます。PGNは、EDP、DP、PDUF、PDUSで定義され、ブロードキャスト通信のためのPGNとして取り扱われます。

CANのデータフレームに格納できるデータは8バイトまでのため、PGNのデータが9バイト以上のときは、8バイトごとのメッセージに分割して送受信を行います。

CAN J1939 All-in-One CPU モジュールが対応する J1939 通信の概要

CAN J1939 All-in-One CPU モジュールは、1つのCAを持つECUとして、J1939通信ネットワークに接続できます。また、CA間でアドレスを管理するためのネットワークマネジメント機能に対応していますので、アドレスの競合の解決や動的アドレス割付が行えます。



仕様

項目	内容	備考
対応するSAE J1939	SAE J1939-11 : Physical Layer, 250K bits/s, Twisted Shielded Pair SAE J1939-15 : Reduced Physical Layer, 250K bits/s, Un-Shielded Twisted Pair SAE J1939-21 : Data Link Layer SAE J1939-71 : Vehicle Application Layer SAE J1939-73 : Application Layer - Diagnostics SAE J1939-75 : Application Layer - Generator Sets and Industrial SAE J1939-81 : Network Management	
通信速度	250k [bps]	
送信方法	イベント送信、周期送信	
受信方法	ラダーのENDで受信、周期監視あり	
最大ノード数	128ノード	アドレス管理ができるCAの数
送信メッセージの最大数	100 個	
受信メッセージの最大数	200 個	
送信メッセージの最大長	252 bytes	
受信メッセージの最大長	252 bytes	
ネットワークマネジメント	有効	アドレス競合の解決、周辺CAのアドレス情報の監視など

CAN J1939 All-in-One CPU モジュールで取り扱う特殊な PGN

一部の PGN は、CPU モジュール内部で通信制御のために使用されています。これらの PGN は、受信メッセージとして設定しても受信できません。

受信メッセージとして設定できない PGN

PGN	パラメータ名	内容
59392 (E800h)	Acknowledgment	CA間の応答に使用されます。
60160 (EB00h)	Data Transfer Message	9 バイト以上のメッセージを分割して送受信するために使用されます。
60416 (EC00h)	Connection Management Message	
60928 (EE00h)	Address Claim	ネットワーク内のアドレス管理のために使用されます。

通信制御

J1939 通信の許可 / 禁止と通信状態の確認は、特殊内部リレーと特殊データレジスタで行えます。



「R/W は、Read (読み出し) / Write (書き込み) の略で、R/W の場合は読み出し・書き込み可能、R の場合は読み出しのみ可能、W の場合は書き込みのみ可能です。

特殊内部リレー割付

特殊内部リレー	説明	設定のタイミング	読み出し / 書き込み
M8300	J1939通信許可フラグ		R/W
M8301	J1939オンラインステータス	毎スキャン	R
M8302	J1939自局アドレス確定ステータス	毎スキャン	R
M8303	J1939通信エラー発生出力	毎スキャン	R
M8304	J1939通信バスオフ発生出力	毎スキャン	R

J1939 通信許可フラグ (M8300)

J1939 通信の許可 / 禁止を制御します。M8300 を ON すると、初期化処理を行ったあとに J1939 通信を開始します。

OFF : 通信禁止 (デフォルト)

ON : 通信許可



初期化処理で送信するメッセージに対して、どの CA からレスポンスがない場合、送信リトライを繰り返し、エラーパツシブ (エラーコード 2002) が発生します。ネットワーク上の CA が送受信できる状態となつてから、M8300 を ON してください。

J1939 オンラインステータス (M8301)

J1939 通信のオンライン状態を示します。オフライン中は、J1939 通信ネットワークへ接続されていないため、メッセージの送受信はできません。

OFF：オフライン

ON：オンライン

J1939 自局アドレス確定ステータス (M8302)

J1939 通信中の自局アドレス確定状態を示します。自局アドレスが 0～253 のとき ON します。オンライン中は、通信開始直後またはアドレスの競合が発生して自局アドレスが 254 (ヌルアドレス) になったとき OFF します。M8302 が OFF のとき、ラダープログラムを使用したメッセージの送受信はできません。

OFF：自局アドレス未確定

ON：自局アドレス確定

J1939 通信エラー発生出力 (M8303)

J1939 通信エラーの発生状態を示します。D8052 (J1939 通信エラーコード) に "0" 以外の値が格納されているとき、M8303 は ON します。

OFF：通信エラー未発生

ON：通信エラー発生中

J1939 通信バスオフ発生出力 (M8304)

J1939 通信中のバスオフ状態を示します。バスオフは、バス上の通信に参加できない状態です。送受信すべての動作が禁止されます。CAN J1939 All-in-One CPU モジュールは、内部に送信エラーカウンタと受信エラーカウンタを持ち、このカウンタが一定数に達するとバスオフ状態になります。

バスオフが発生すると、CAN J1939 All-in-One CPU モジュールは通信を停止してオフラインになります。再度通信を行うには、M8300 を OFF から ON して、オンラインに戻す必要があります。M8300 が OFF から ON したとき、M8304 は OFF します。

OFF：バスオフ未発生

ON：バスオフ発生中

特殊データレジスタ割付

特殊データレジスタ	機能	設定のタイミング	読み出し / 書き込み
D8052	J1939通信エラーコード	エラー発生時	R/W

J1939 通信エラーコード (D8052)

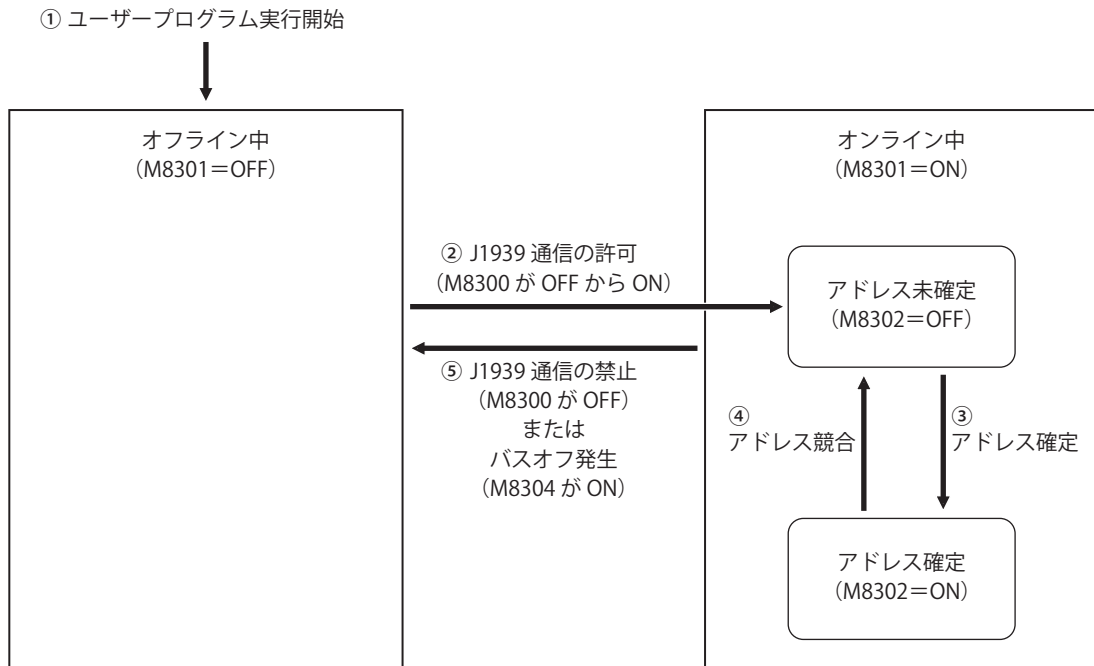
J1939 通信エラーコードを格納します。初期化する場合は "0" を書き込んでください。

J1939 通信エラーコードの詳細は、次のとおりです。

ステータスコード	内容	詳細
0	正常	
100	分割送受信エラー	予期しないBAMフレームを受信した。
101	分割送受信エラー	予期しないRTSフレームを受信した。
102	分割送受信エラー	予期しないCTSフレームを受信した。
103	分割送受信エラー	予期しないEOMフレームを受信した。
104	分割送受信エラー	予期しないConnection Abortフレームを受信した。
105	分割送受信エラー	予期しないDTフレームを受信した。
106	分割送受信エラー	分割受信メッセージのデータ長が範囲外。
107	分割送受信エラー	Connection Abortメッセージ送信に失敗した。
110	NACK送信失敗	PGNの送信要求に対して、NACKの送信ができなかった。
200	内部送信キューオーバーフロー	一度に送信できるメッセージ数が上限を超えた。
201	内部受信キューオーバーフロー	一度に受信できるメッセージ数が上限を超えた。
1000	自局アドレス未確定	自局アドレスが254になった。
2000	CAN通信初期化エラー	CANコントローラの初期化ができなかった。
2001	CAN通信初期化エラー	CANコントローラのリセットができなかった。
2002	エラーパッシブ	エラー / ワーニング割込みが発生し、CANコントローラがエラーパッシブ状態になった。
2003	エラーアクティブ	CANコントローラがエラーアクティブ状態になった。
2004	データオーバーランエラー	データオーバーラン割込みが発生した。
3000	バスオフエラー	CANでバスオフが発生した。

通信制御の状態遷移

J1939 通信の通信状態と初期化処理の遷移は、次のとおりです。



- ① ユーザープログラムの実行開始直後は、オフライン状態です。
- ② M8300 が OFF から ON になると、オンライン状態となり、初期化処理を行います。
初期化処理では、初期化用メッセージを送信してアドレスの重複を確認し、重複した場合は競合の解決を行います。
初期化中で自局アドレスが確定する前は、アドレス未確定状態 (M8302 = OFF) です。
- ③ 自局アドレスが 0 ～ 253 に確定すると、アドレス確定状態 (M8302 = ON) となります。
FC6A 形と CA の間でメッセージの送受信ができます。自局アドレスが 254 の場合は、アドレス未確定状態のままです。
- ④ アドレスの競合が発生し、自局のアドレスが 254 (マルチアドレス) になったとき、アドレス未確定状態になります。
- ⑤ M8300 が OFF になると、通信処理を停止してオフライン状態になります。
また、通信中にバスオフが発生 (M8304 が ON) してもオフライン状態になります。このとき、M8300 は OFF のままです。
再度オンライン状態に戻すには、M8300 をいったん OFF してから、再度 ON してください。

J1939 通信の設定

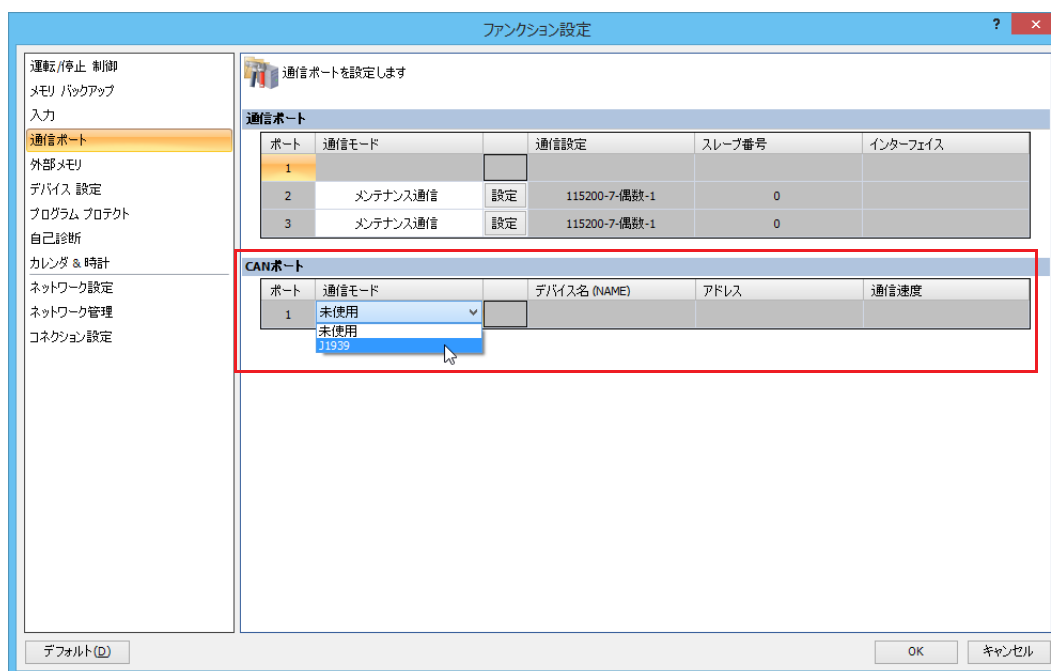
J1939 通信に関する操作手順と各項目の詳細について説明します。

WindLDR の設定

J1939 通信を使用環境に合わせて設定をします。

●操作手順

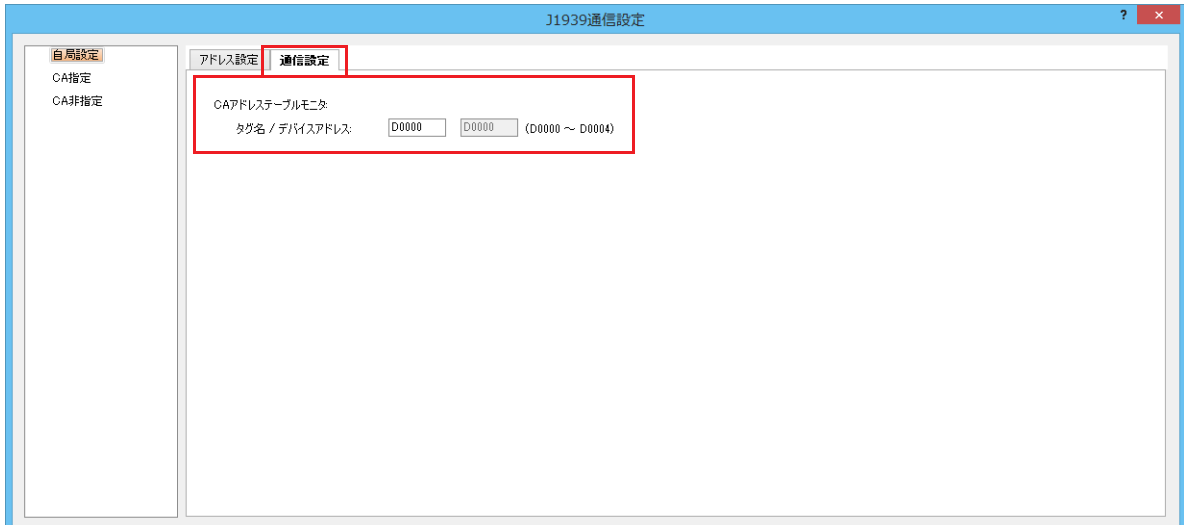
1. [設定] タブの [ファンクション設定] で [通信ポート] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。
2. CAN ポートグループの通信モードから "J1939" を選択します。
[J1939 通信設定] ダイアログボックスが表示されます。



3. [自局設定] の [アドレス設定] タブで自局のアドレスやデバイス名 (NAME)、CA アドレステーブルを設定します。
自局のアドレスやデバイス名 (NAME) を設定します。



4. [通信設定] タブをクリックし、CA アドレステーブルを格納するデバイスアドレスを設定します。



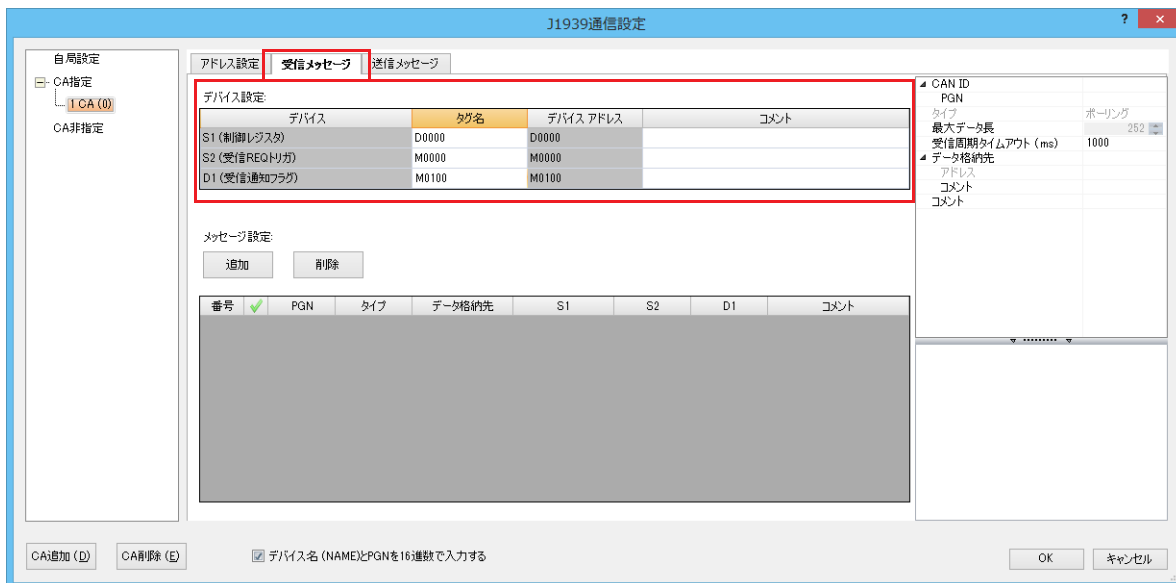
5. [CA 追加] ボタンをクリックし、各 CA を設定するノードを追加します。



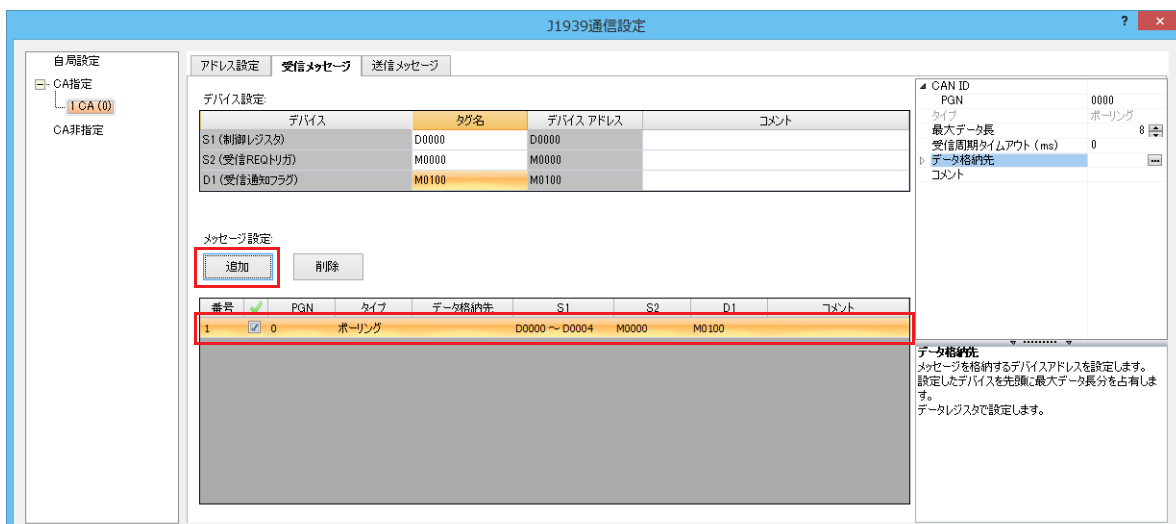
6. 設定する CA に対応したノードをクリックし、[アドレス設定] タブをクリックします。各 CA のアドレス、デバイス名を設定します。



7. [受信メッセージ] タブをクリックし、はじめに受信メッセージで使用するデバイスを設定します。



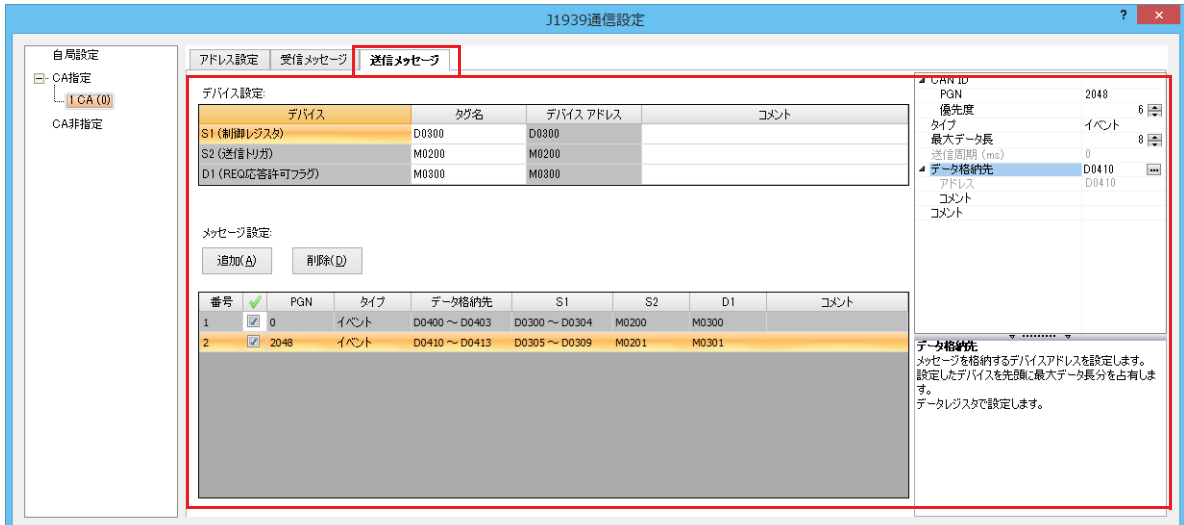
8. [追加] ボタンをクリックし、受信メッセージを追加します。



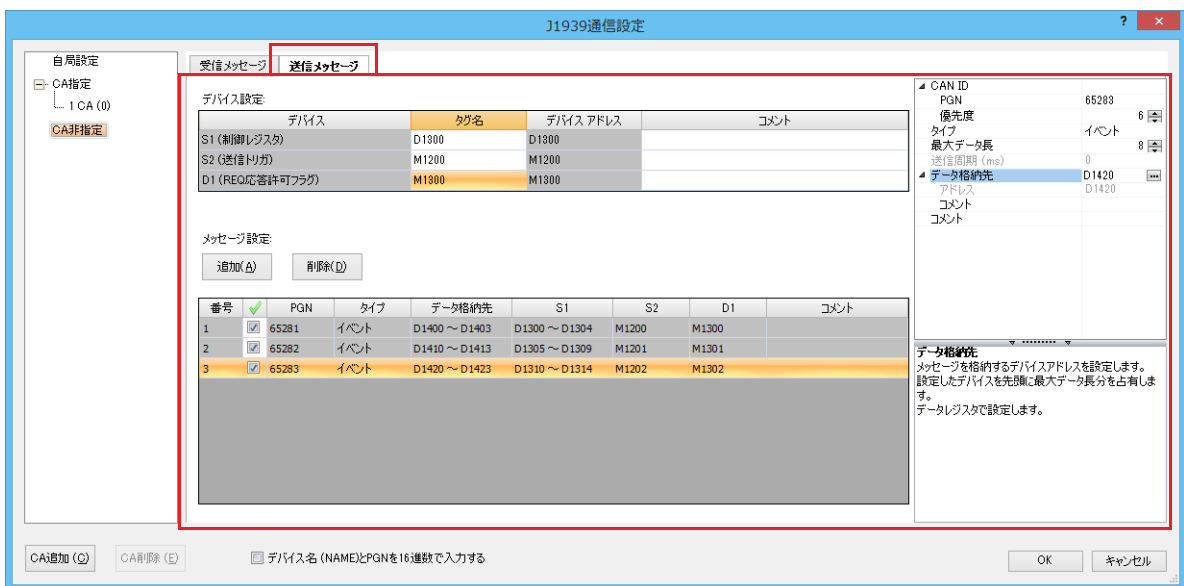
9. メッセージ設定の任意のセルを選択すると、各メッセージのパラメータが右側に表示されますので、各項目を設定します。



10. [送信メッセージ] タブをクリックし、[受信メッセージ] タブと同様に設定します。



11. [CA 非指定] をクリックします。[CA 指定] の [受信メッセージ] タブ、[送信メッセージ] タブと同様に設定します。



12. [OK] ボタンをクリックします。

以上で、設定が完了します。

J1939 通信のパラメータ設定

J1939 通信を使用するために必要な [J1939 設定] ダイアログボックスの各パラメータ設定について説明します。

自局設定

CAN J1939 All-in-One CPU モジュールの自局アドレス、自局 NAME および CA アドレステーブルを設定します。

■ [アドレス設定] タブ

CAN J1939 All-in-One CPU モジュールの自局アドレスと自局 NAME を設定します。



① アドレス

CAN J1939 All-in-One CPU モジュールの自局アドレスの初期値を 0 ~ 253 の範囲で設定します。

[デバイス名 (NAME) と PGN を 16 進数で入力する] チェックボックスをオンにした場合は、16 進数で入力します。

例) 16 進数の場合: 01、F0 など

同じネットワークに接続する CA とのアドレスの競合が発生した場合は、自局 NAME の Arbitrary Address Capable ビットに応じて、自局アドレスを再取得します。

② デバイス名 (NAME)

CAN J1939 All-in-One CPU モジュールの自局 NAME を 64 ビットで設定します。

[デバイス名 (NAME) と PGN を 16 進数で入力する] チェックボックスをオンにした場合は、16 進数で入力します。

例) 16 進数の場合: 123456789ABCDEF1、0000FFFFFFFFFOA など

同一ネットワークの他の CA と重複しない値を設定してください。詳細は、「NAME (合計 64 ビット)」(8-2 頁) を参照してください。

デバイス名 (NAME) は、デバイス名フィールド一覧 (③) の設定値で構成されます。デバイス名 (NAME) を変更すると、デバイス名フィールド一覧の [設定値] に反映されます。

③ デバイス名フィールド一覧

デバイス名 (NAME) (②) がフィールドごとに表示されます。[設定値] を変更すると、デバイス名 (NAME) に反映されます。

■ [通信設定] タブ

CA アドレステーブルモニタを設定します。CA アドレステーブルモニタは、CAN J1939 All-in-One CPU モジュール（自局）および [CA 指定] で登録した CA のアドレスを管理するためのテーブルです。



① CA アドレステーブルモニタ

CA アドレステーブルを格納するデータレジスタの先頭アドレスを設定します。

WindLDR で設定した値が初期値として CA アドレステーブルへ格納され、アドレスの競合によって CAN J1939 All-in-One CPU モジュールのアドレスが変わった場合、それに合わせて CA アドレステーブルが更新されます。また、CA からアドレス情報を受信した場合は、NAME が一致する CA のアドレスが更新されます。CAN J1939 All-in-One CPU モジュールは、CA アドレステーブルのアドレスをもとに、メッセージの送受信を行います。

CA アドレステーブルモニタの定義

CA	格納先	機能	設定内容
自局	先頭番号+0	自局 NAME	CAN J1939 All-in-One CPUモジュールのNAMEを格納します。
	先頭番号+1		
	先頭番号+2		
	先頭番号+3		
	先頭番号+4	自局 アドレス*1	CAN J1939 All-in-One CPUモジュールのアドレスを格納します。オンライン中にアドレスが変わった場合、以降のメッセージの送受信は新しいアドレスに基づいて処理されます。
CA#1 (5ワード)	先頭番号+5	CA#1 NAME*2	CA#1 のNAMEを格納します。
	先頭番号+6		
	先頭番号+7		
	先頭番号+8		
	先頭番号+9	CA#1 アドレス*2*3	CA#1 のアドレスを格納します。NAMEを設定した場合は、オンライン状態のとき、途中でアドレスが変わることがあります。途中でアドレスが変わった場合、以降のメッセージ送受信は新しいアドレスに基づいて処理されます。
⋮	⋮	⋮	
CA#N*3 (5ワード)	先頭番号+5×N	CA#N NAME*2	CA#NのNAMEを格納します。
	先頭番号+5×N+1		
	先頭番号+5×N+2		
	先頭番号+5×N+3		
		先頭番号+5×N+4	CA#N アドレス*2*3

*1 ネットワーク上の CA とのアドレスの競合によりアドレスが割り当てられない場合は、254(ヌルアドレス)となります。この場合、メッセージの送受信はできません。

*2 アドレスが 0 ~ 253 の範囲でない場合、該当する CA とメッセージの送受信はできません。

*3 N は 1 ~ 253 です。

例 1) 自局アドレスが競合によって更新される場合

CA アドレステーブル

自局 NAME	+0ワード	9234...5678h	※Arbitrary Address Capable=1 →128に更新される
自局 アドレス	+4ワード	30	
CA#1 NAME	+5ワード	1345...6789h	
CA#1 アドレス	+9ワード	150	
		⋮	
		⋮	
		⋮	

例 2) CA アドレス情報を受信して更新する場合

CA アドレステーブル

自局 NAME	+0ワード	2233...5678h	CAアドレス情報 A844...1111h NAME 130 アドレス
自局 アドレス	+4ワード	128	
CA#1 NAME	+5ワード	2444...6789h	
CA#1 アドレス	+9ワード	150	
		⋮	
		⋮	
CA#4 NAME	+20ワード	A844...1111h	NAMEが一致するCAのアドレスが更新される →130に更新される
CA#4 アドレス	+24ワード	50	
		⋮	
		⋮	

例 3) 例 2 で CA #4 の NAME を 0 に設定した場合

CA アドレステーブル

自局 NAME	+0ワード	2233...5678h	CAアドレス情報 A844...1111h NAME 130 アドレス
自局 アドレス	+4ワード	128	
CA#1 NAME	+5ワード	2444...6789h	
CA#1 アドレス	+9ワード	150	
		⋮	
		⋮	
CA#4 NAME	+20ワード	0h	NAMEがCAアドレステーブルに存在しないため、 CAアドレステーブルは更新されない 更新されない
CA#4 アドレス	+24ワード	50	
		⋮	
		⋮	

CA 指定

追加した CA ごとに、メッセージを送受信する CA のアドレス、NAME および送受信メッセージを設定します。

■ [アドレス設定] タブ

CA 一覧で選択した CA のアドレスと NAME を設定します。

The screenshot shows the 'J1939通信設定' dialog box with the 'アドレス設定' tab selected. On the left, there is a tree view with 'CA指定' and 'CA (0)'. The main area contains the following fields and table:

- ① アドレス: 0
- ② デバイス名 (NAME): 0000000000000000
- ③ パラメータ一覧表:

パラメータ名	設定値	入力範囲	ビット領域	備考
動的アドレス対応 (Arbitrary Address Capable)	0	0 ~ 1	1	SAE定義済み(最上位ビット)
産業グループ (Industry Group)	0	0 ~ 7	3	SAE定義済み
車両システムインスタンス (Vehicle System Instance)	0	0 ~ 15	4	
車両システム (Vehicle System)	0	0 ~ 127	7	SAE定義済み
予約済み (Reserved)	0	0	1	SAE定義済み(固定値)
機能 (Function)	0	0 ~ 255	8	SAE定義済み
機能インスタンス (Function Instance)	0	0 ~ 31	5	
ECUインスタンス (ECU Instance)	0	0 ~ 7	3	
メーカーコード (Manufacturer Code)	0	0 ~ 2047	11	SAE定義済み
IDナンバー (Identity Number)	0	0 ~ 2097151	21	最下位ビット

- ④ コメント: (empty field)

At the bottom, there are buttons for 'CA追加 (D)', 'CA削除 (E)', a checked checkbox 'デバイス名 (NAME)とPGNを16進数で入力する', and 'OK' and 'キャンセル' buttons.

① アドレス

メッセージを送受信する CA のアドレスを 0 ~ 254 の範囲で設定します。

[デバイス名 (NAME) と PGN を 16 進数で入力する] チェックボックスをオンにした場合は、16 進数で入力します。

例) 16 進数の場合: 01、F0 など

② デバイス名 (NAME)

CA の NAME を 64 ビットで設定します。CA の NAME が定義されていない場合は、0 を設定してください。

[デバイス名 (NAME) と PGN を 16 進数で入力する] チェックボックスをオンにした場合は、16 進数で入力します。

例) 16 進数の場合: 123456789ABCDEF1、0000FFFFFFFFF0A など

デバイス名 (NAME) は、デバイス名フィールド一覧 (③) の設定値で構成されます。デバイス名 (NAME) を変更すると、デバイス名フィールド一覧の [設定値] に反映されます。

③ デバイス名フィールド一覧

デバイス名 (NAME) (②) がフィールドごとに表示されます。[設定値] を変更すると、デバイス名 (NAME) に反映されます。

④ コメント

CA のコメントを最大 127 文字で設定します。[CA 指定] の CA 一覧に反映されます。

■ [受信メッセージ] タブ

CAN J1939 All-in-One CPU モジュールが CA 一覧で選択した CA から受信するメッセージを設定します。



① デバイス設定

受信メッセージで使用するデータレジスタを設定します。

設定項目	内容
タグ名	デバイスのタグ名、またはデバイスアドレスを指定します。
デバイスアドレス	タグ名に対応するデバイスアドレスを表示します。
コメント	デバイスアドレスのコメントを表示します。編集可能です。

S1 (制御レジスタ)

受信結果を格納するデータレジスタの先頭アドレスを設定します。

格納先	項目	内容	
先頭番号+0	受信データ長	受信したメッセージのデータ長を格納します。 8バイトを超えるメッセージを受信する場合、8バイトごとに分割されたメッセージを受信します。受信完了後、分割されたデータが結合され、データ格納先へ格納されます。 受信したメッセージのデータ長が最大データ長よりも大きい場合、最大データ長を格納し、受信結果ステータスに100が格納されます。	
先頭番号+1	リザーブ		
先頭番号+2	受信結果ステータス	受信結果を格納します。初期化する場合に0を書き込んでください。受信結果ステータスの詳細は、「受信結果ステータス」(8-16頁)を参照してください。	
先頭番号+3	受信動作フラグ		
	ビット0	受信バッファ上書きフラグ	内部の受信バッファに格納したメッセージのデータをデータ格納先に書き込む前に、次のメッセージによって受信バッファが上書きされたとき、1スキャンだけONします。
	ビット1~15	リザーブ	
先頭番号+4	リザーブ		

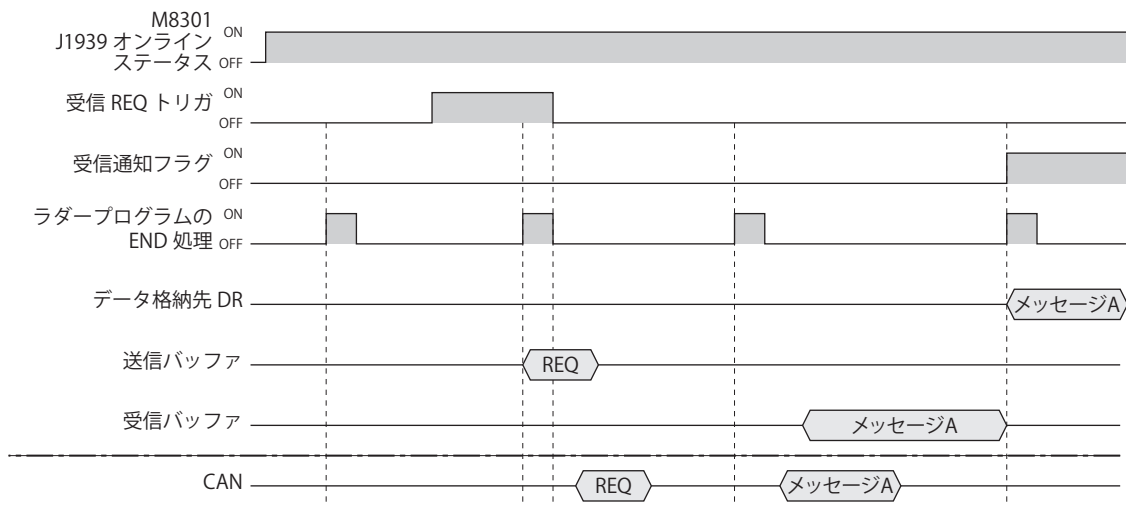
受信結果ステータス

ステータスコード	内容	詳細
0	正常	
10	受信周期タイムアウト	前回受信してから、受信周期タイムアウトの設定時間を経過した。
100	受信データ長エラー	受信したメッセージのデータ長が、最大データ長よりも長い。
101	受信データ長エラー	受信したメッセージが252バイト以上だった。
102	分割受信エラー	データ長が9バイト以上のメッセージの受信ができなかった。

S2 (受信 REQ トリガ)

対応する CA にメッセージの送信を要求するための機能です。受信 REQ トリガを ON すると、ラダープログラムの END 処理で CA に PGN 59904 (Request) を送信します。

受信 REQ トリガの使用例



D1 (受信通知フラグ)

メッセージの受信を通知するためのフラグを設定します。メッセージを受信したときに、1 スキャンだけ ON します。

② [追加] ボタン / [削除] ボタン

[追加] ボタンをクリックすると、メッセージ一覧 (③) に受信メッセージを追加します。
削除する場合は、メッセージ一覧で削除したいメッセージを選択し、[削除] ボタンをクリックします。

③メッセージ一覧

メッセージパラメータ設定 (④) で設定した内容が表示されます。また、受信メッセージの有効 / 無効を設定します。チェックボックスがオンの場合は有効、オフの場合は無効になります。



- チェックボックス列のヘッダをクリックすると、次の動作になります。
- 各受信メッセージのチェックボックスがすべてオンの場合は、すべてオフになります。
 - いずれかの受信メッセージのチェックボックスがオフの場合は、すべてオンになります。

④メッセージパラメータ設定

メッセージ一覧で選択した受信メッセージの詳細を設定します。

CAN ID

PGN : 受信するメッセージの PGN を設定します。設定できる PGN の範囲は、0 ~ 61184 (0000h ~ EF00h) (PDU1 フォーマット) です。PDU1 フォーマットで送信先アドレスを示す下位 1 バイト (PDU Specific) は 00h としてください。
[...] をクリックして表示される PGN マネージャーで PGN を選択することもできます。詳細は、「PGN マネージャー」(8-29 頁) を参照してください。

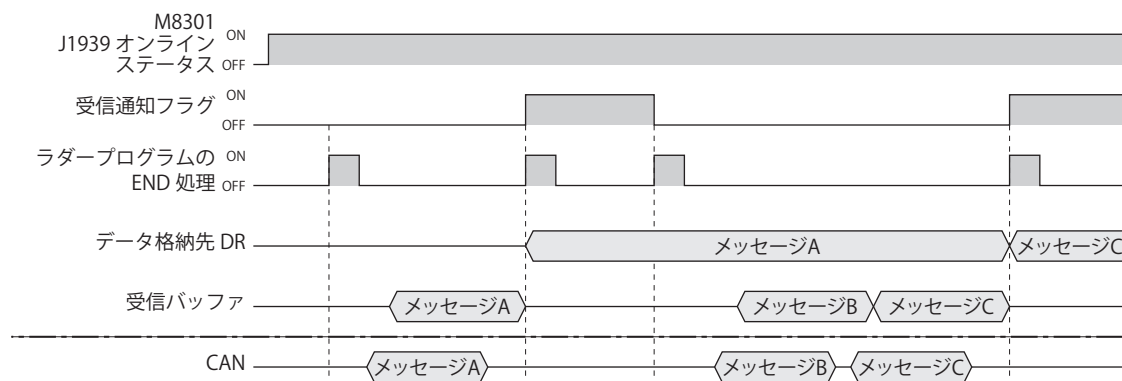
タイプ

メッセージの受信方法は、“ポーリング”固定です。

受信したメッセージを内部の受信バッファで一時保存し、ラダープログラムの END 処理で受信バッファの内容をデータ格納先に設定したデータレジスタに書き込みます。このとき、受信通知フラグを 1 スキャンだけ ON します。

ラダープログラムの END 処理から次の END 処理までの間に、同じ CA から同じ PGN のメッセージを複数回受信した場合、最後に受信したメッセージの内容をデータ格納先に書き込みます。

ポーリング受信例



最大データ長

受信する PGN の最大データ長を設定します。設定できる最大値は 252 バイトです。

受信周期タイムアウト (ms)

周期的に受信するメッセージの受信周期を監視するための時間を 10ms 単位で設定します。設定できる範囲は 0 ~ 655350 です。0 を設定した場合は、受信周期の監視を行いません。

データ格納先

受信したメッセージを格納するデータレジスタを設定します。設定したデバイスを先頭に最大データ長分を占有します。

アドレス： デバイスのタグ名、またはデバイスアドレスに対応するデバイスアドレスを表示します。

コメント： デバイスのタグ名、またはデバイスアドレスに対応するコメントを表示します。編集可能です。

コメント

受信メッセージのコメントを設定します。設定できるコメントは最大 127 バイトです。

■ [送信メッセージ] タブ

CAN J1939 All-in-One CPU モジュールが CA 一覧で選択した CA に送信するメッセージを設定します。



① デバイス設定

送信メッセージで使用するデータレジスタを設定します。

設定項目	内容
タグ名	デバイスのタグ名、またはデバイスアドレスを指定します。
デバイスアドレス	タグ名に対応するデバイスアドレスを表示します。
コメント	デバイスアドレスのコメントを表示します。編集可能です。

S1 (制御レジスタ)

送信設定および送信結果を格納するデータレジスタの先頭アドレスを設定します。

格納先	項目	内容
先頭番号+0	送信データ長	送信するメッセージのデータ長を格納します。データ格納先を先頭に、送信データ長分のデータをCAへ送信します。送信データ長に0を格納した場合、最大データ長のデータをCAへ送信します。最大データ長より長くは設定できません。8バイトを超えるメッセージを送信する場合、8バイトごとに分割されたメッセージを送信します。
先頭番号+1	リザーブ	
先頭番号+2	送信結果ステータス	送信結果を格納します。初期化する場合は0を書き込んでください。送信結果ステータスの詳細は、「送信結果ステータス」(8-19頁)を参照してください。
先頭番号+3	送信動作フラグ	送信に関する動作フラグなどを格納します。
	ビット0	REQ受信ありフラグ
先頭番号+4	ビット1~15	リザーブ
	リザーブ	

送信結果ステータス

ステータスコード	内容	詳細
0	正常	
1	分割送信完了	データ長が9バイト以上のメッセージの送信が完了した。
10	送信データ長エラー	送信したメッセージのデータ長が最大データ長よりも長い。
20	内部送信キューオーバーフロー	一度に送信予約できるメッセージの数が上限に達した。
100	分割送信エラー	データ長が9バイト以上のメッセージの送信ができなかった。
200	自局アドレス未確定	自局アドレスが未確定のため、送信できなかった。
201	送信先アドレス未確定	送信先CAのアドレスが未確定のため、送信できなかった。

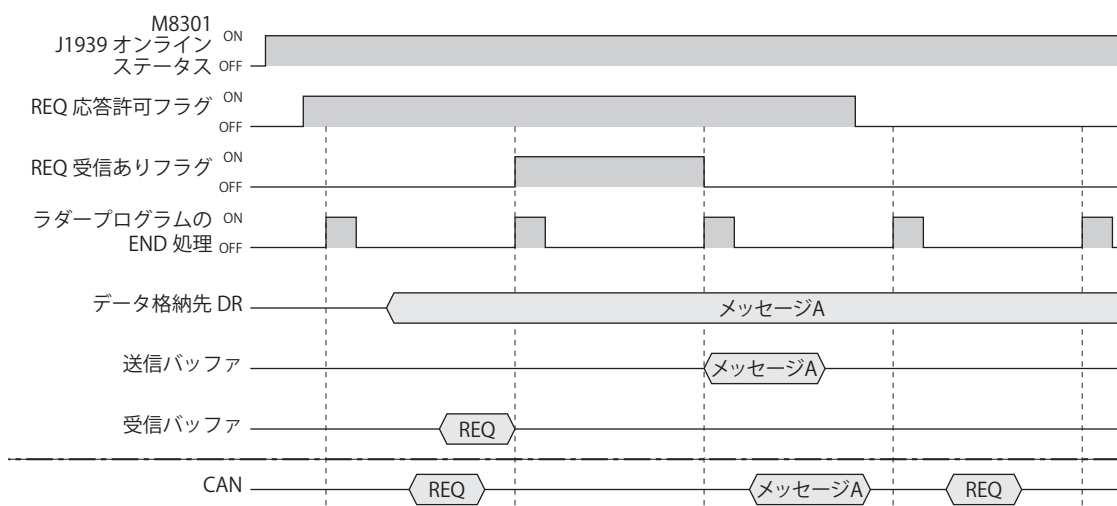
S2 (送信トリガ)

[タイプ] で“イベント送信”を選択した場合、送信トリガを設定します。送信トリガを ON すると、ラダープログラムの END 処理でメッセージを送信します。送信後、送信トリガは OFF します。

D1 (REQ 応答許可フラグ)

REQ 応答許可フラグが ON の間、対応する CA から受信した PGN の送信要求に応じて、PGN を送信する機能です。CA から PGN59904 (Request) を受信すると、送信動作フラグの REQ 受信ありフラグが ON し、次の END 処理で対応する PGN のメッセージを送信します。

リクエスト応答例



② [追加] ボタン / [削除] ボタン

[追加] ボタンをクリックすると、メッセージ一覧 (③) に送信メッセージの行を追加します。削除する場合は、メッセージ一覧で削除したいメッセージを選択し、[削除] ボタンをクリックします。

③メッセージ一覧

メッセージパラメータ設定 (④) で設定した内容が表示されます。また、送信メッセージの有効 / 無効を設定します。チェックボックスがオンの場合は有効、オフの場合は無効になります。



チェックボックス列のヘッダをクリックすると、次の動作になります。

- ・各送信メッセージのチェックボックスがすべてオンの場合は、すべてオフになります。
- ・いずれかの送信メッセージのチェックボックスがオフの場合は、すべてオンになります。

④メッセージパラメータ設定

メッセージ一覧で選択した送信メッセージの詳細を設定します。

CAN ID

PGN： 送信するメッセージのPGNを設定します。設定できるPGNの範囲は、0～61184 (0000h～EF00h) (PDU1フォーマット) です。PDU1フォーマットで送信先アドレスを示す下位1バイト (PDU Specific) は00hとしてください。

[...] をクリックして表示されるPGNマネージャーでPGNを選択することもできます。詳細は、「PGNマネージャー」(8-29頁)を参照してください。

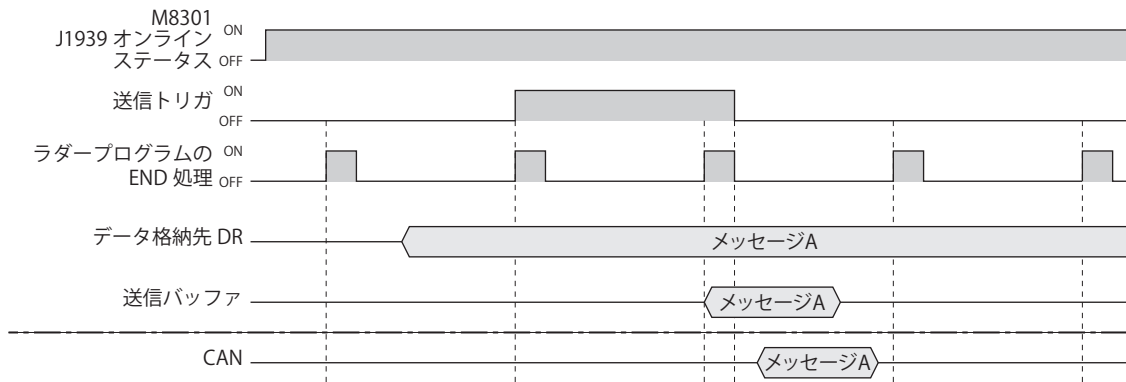
優先度： 送信するメッセージの優先度を0～7の範囲で設定します。

タイプ

メッセージの送信方法を設定します。“イベント”または“周期”から選択します。

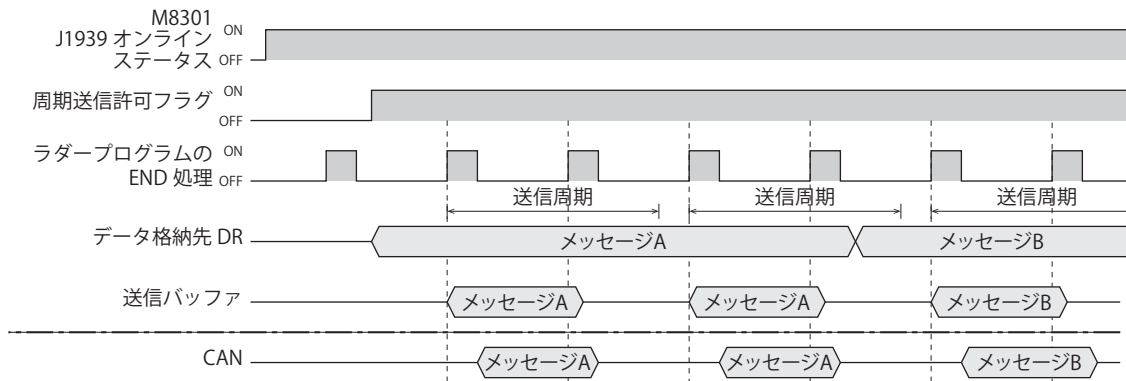
イベント： 送信トリガをONすると、データ格納先に格納した送信データがラダープログラムのEND処理で送信されます。

CA 指定時のイベント送信例



周期： 周期送信許可フラグをONしている間、送信周期になると、データ格納先に格納した送信データが、ラダープログラムのEND処理で送信されます。実際に送信する周期は、ラダープログラムのスキャン時間に影響されます。

CA 指定の周期送信例



最大データ長

送信するPGNの最大データ長を設定します。設定できる最大値は252バイトです。

送信周期 (ms)

[タイプ] で“周期”を選択した場合に、送信周期を10ミリ秒単位で設定します。設定できる値の範囲は10～655350です。

データ格納先

送信メッセージを格納先するデータレジスタを設定します。設定したデバイスを先頭に最大データ長分を占有します。メッセージを送信するときは、データ格納先からデータを内部の送信バッファにコピーしてから送信します。

アドレス： デバイスのタグ名、またはデバイスアドレスに対応するデバイスアドレスを表示します。

コメント： デバイスのタグ名、またはデバイスアドレスに対応するコメントを表示します。編集可能です。

コメント

送信メッセージのコメントを設定します。設定できるコメントは最大 127 バイトです。

CA 非指定

特定の CA を指定せずにブロードキャストで送受信するメッセージを設定します。

■ [受信メッセージ] タブ

他の CA から J1939 通信ネットワークにブロードキャストで送信されるメッセージを設定します。



① デバイス設定

受信メッセージで使用するデータレジスタを設定します。

設定項目	内容
タグ名	デバイスのタグ名、またはデバイスアドレスを指定します。
デバイスアドレス	タグ名に対応するデバイスアドレスを表示します。
コメント	デバイスアドレスのコメントを表示します。編集可能です。

S1 (制御レジスタ)

受信結果を格納するデータレジスタの先頭アドレスを設定します。

格納先	項目	内容	
先頭番号+0	受信データ長	受信したメッセージのデータ長を格納します。 8バイトを超えるメッセージを受信する場合、8バイトごとに分割されたメッセージを受信します。受信完了後、分割されたデータが結合され、データ格納先へ格納されます。 受信したメッセージのデータ長が最大データ長よりも大きい場合、最大データ長を格納し、受信結果ステータスに100が格納されます。	
先頭番号+1	送信元アドレス	受信したメッセージの送信元アドレスを格納します。	
先頭番号+2	受信結果ステータス	受信結果を格納します。初期化する場合は0を書き込んでください。 受信結果ステータスの詳細は、「受信結果ステータス」(8-16頁)を参照してください。	
先頭番号+3	受信動作フラグ		
	ビット0	受信バッファ上書きフラグ	内部の受信バッファに格納したメッセージのデータをデータ格納先に書き込む前に、次のメッセージによって受信バッファが上書きされたとき、1スキャンだけONします。
	ビット1~15	リザーブ	
先頭番号+4	リザーブ		

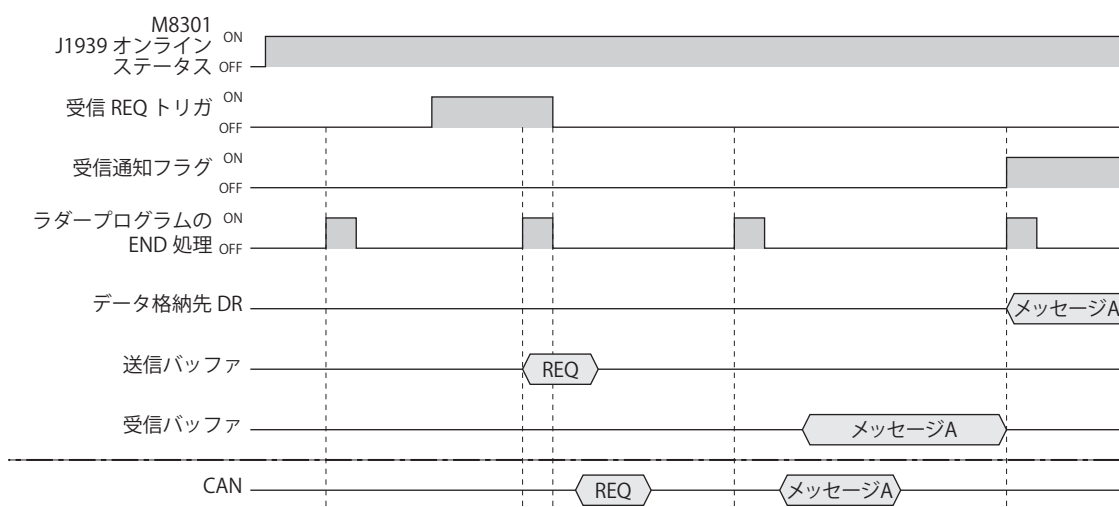
受信結果ステータス

ステータスコード	内容	詳細
0	正常	
10	受信周期タイムアウト	前回受信してから、受信周期タイムアウトの設定時間を経過した。
100	受信データ長エラー	受信したメッセージのデータ長が、最大データ長よりも長い。
101	受信データ長エラー	受信したメッセージが252バイト以上だった。
102	分割受信エラー	データ長が9バイト以上のメッセージの受信ができなかった。

S2 (受信 REQ トリガ)

J1939 通信ネットワークにブロードキャストで PGN の送信要求を行うための機能です。受信 REQ トリガを ON すると、ラダープログラムの END 処理で PGN 599034 (Request) をブロードキャストで送信します。

受信 REQ トリガの使用例



D1 (受信通知フラグ)

メッセージの受信を通知するためのフラグを設定します。メッセージを受信したときに、1 スキャンだけ ON します。

② [追加] ボタン / [削除] ボタン

[追加] ボタンをクリックすると、メッセージ一覧 (③) に受信メッセージの行を追加します。

削除する場合は、メッセージ一覧で削除したいメッセージを選択し、[削除] ボタンをクリックします。

③メッセージ一覧

メッセージパラメータ設定 (④) で設定した内容が表示されます。また、受信メッセージの有効 / 無効を設定します。チェックボックスがオンの場合は有効、オフの場合は無効になります。



チェックボックス列のヘッダをクリックすると、次の動作になります。

- 各受信メッセージのチェックボックスがすべてオンの場合は、すべてオフになります。
- いずれかの受信メッセージのチェックボックスがオフの場合は、すべてオンになります。

④メッセージパラメータ設定

メッセージ一覧で選択した受信メッセージの詳細を設定します。

CAN ID

PGN : 受信するメッセージの PGN を 2 バイトで設定します。設定できる PGN の範囲は、0 ~ 61184 (0000h ~ EF00h) (PDU1 フォーマット)、61440 ~ 65535 (F000h ~ FFFFh) (PDU2 フォーマット) です。PDU1 フォーマットで送信先アドレスを示す下位 1 バイト (PDU Specific) は 00h としてください。

PDU1 フォーマットの範囲で設定した場合、送信先アドレスが 255 (グローバルアドレス) のメッセージのみ受信できます。

[...] をクリックして表示される PGN マネージャーで PGN を選択することもできます。詳細は、「PGN マネージャー」(8-29 頁) を参照してください。

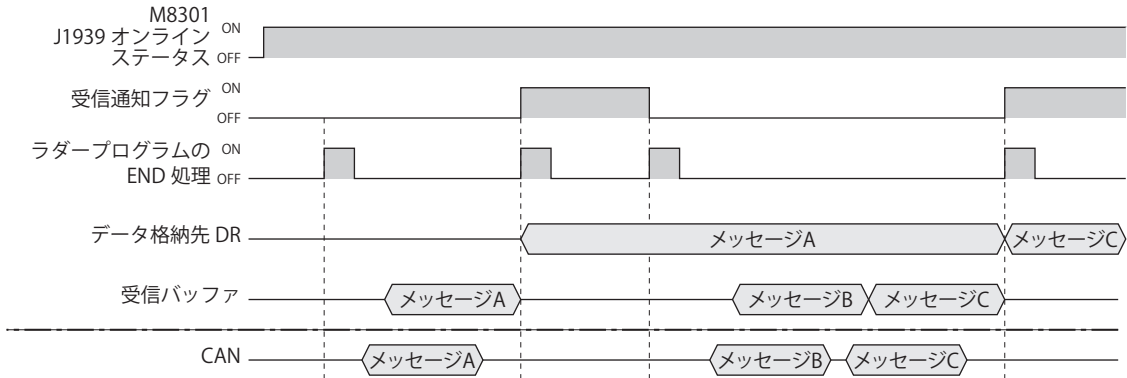
タイプ

メッセージの受信方法は、“ポーリング”固定です。

受信したメッセージを内部の受信バッファで一時保存し、ラダープログラムのEND処理で受信バッファの内容をデータ格納先に設定したデータレジスタに書き込みます。このとき、受信通知フラグを1スキャンだけONします。

ラダープログラムのEND処理から次のEND処理までの間に、同じCAから同じPGNのメッセージを複数回受信した場合、最後に受信したメッセージの内容をデータ格納先に書き込みます。

ポーリング受信例



最大データ長

受信するPGNの最大データ長を設定します。設定できる最大値は252バイトです。

受信周期タイムアウト (ms)

周期的に受信するメッセージの受信周期を監視するための時間を10ms単位で設定します。設定できる範囲は0～655350です。0を設定した場合は、受信周期の監視を行いません。

データ格納先

受信したメッセージを格納先するデータレジスタを設定します。設定したデバイスを先頭に最大データ長分を占有します。

アドレス： デバイスのタグ名、またはデバイスアドレスに対応するデバイスアドレスを表示します。

コメント： デバイスのタグ名、またはデバイスアドレスに対応するコメントを表示します。編集可能です。

コメント

受信メッセージのコメントを設定します。設定できるコメントは最大127バイトです。

■ [送信メッセージ] タブ

CAN J1939 All-in-One CPU モジュールが J1939 通信ネットワークにブロードキャストで送信するメッセージを設定します。



① デバイス設定

送信メッセージで使用するデータレジスタを設定します。

設定項目	内容
タグ名	デバイスのタグ名、またはデバイスアドレスを指定します。
デバイスアドレス	タグ名に対応するデバイスアドレスを表示します。
コメント	デバイスアドレスのコメントを表示します。編集可能です。

S1 (制御レジスタ)

送信設定および送信結果を格納するデータレジスタの先頭アドレスを設定します。

格納先	項目	内容	
先頭番号+0	送信データ長	送信するメッセージのデータ長を格納します。 データ格納先を先頭に、送信データ長分のデータをCAへ送信します。 送信データ長に0を格納した場合、最大データ長のデータをCAへ送信します。最大データ長より長くは設定できません。 8バイトを超えるメッセージを送信する場合、8バイトごとに分割されたメッセージを送信します。	
先頭番号+1	リクエスト応答先アドレス	受信したリクエストPGNの送信元アドレスをPGNの応答先アドレスとして格納します。	
先頭番号+2	送信結果ステータス	送信結果を格納します。初期化する場合に0を書き込んでください。 送信結果ステータスの詳細は、「送信結果ステータス」(8-19頁)を参照してください。	
先頭番号+3	送信動作フラグ		
	ビット0	REQ受信ありフラグ	CAからリクエストPGNを受信した場合にONします。対応するPGNを送信するとOFFします。
	ビット1~15	リザーブ	
先頭番号+4	リザーブ		

送信結果ステータス

ステータスコード	内容	詳細
0	正常	
1	分割送信完了	データ長が9バイト以上のメッセージの送信が完了した。
10	送信データ長エラー	送信したメッセージのデータ長が最大データ長よりも長い。
20	内部送信キューオーバーフロー	一度に送信予約できるメッセージの数が上限に達した。
100	分割送信エラー	データ長が9バイト以上のメッセージの送信ができなかった。
200	自局アドレス未確定	自局アドレスが未確定のため、送信できなかった。
201	送信先アドレス未確定	送信先CAのアドレスが未確定のため、送信できなかった。

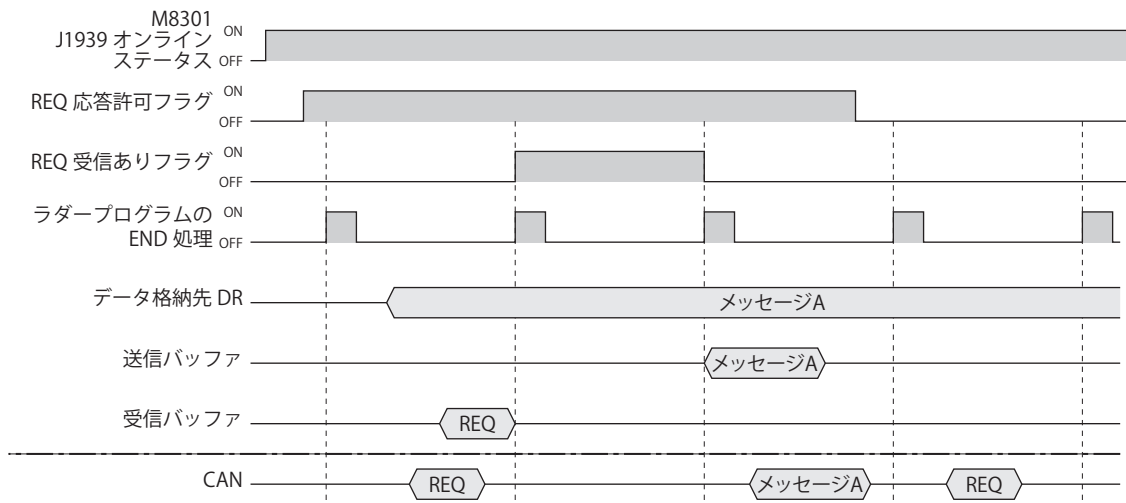
S2 (送信トリガ)

[タイプ] で“イベント送信”を選択した場合、送信トリガを設定します。送信トリガを ON すると、ラダープログラムの END 処理でメッセージを送信します。送信後、送信トリガは OFF します。

D1 (REQ 応答許可フラグ)

ブロードキャストで送信されたメッセージの送信要求へ応答する機能です。ブロードキャストで送信された PGN 59904 (Request) を受信したとき、対応する PGN の REQ 応答許可フラグが ON なら、PGN 59904 (Request) の送信元アドレスがリクエスト応答先アドレスに格納され、送信動作フラグの REQ 受信ありフラグが ON します。次の END 処理で、リクエスト応答先アドレスに対して、要求された PGN のメッセージが送信されます。

リクエスト応答例



② [追加] ボタン / [削除] ボタン

[追加] ボタンをクリックすると、メッセージ一覧 (③) に送信メッセージの行を追加します。

削除する場合は、メッセージ一覧で削除したいメッセージを選択し、[削除] ボタンをクリックします。

③メッセージ一覧

メッセージパラメータ設定 (④) で設定した内容が表示されます。また、送信メッセージの有効 / 無効を設定します。チェックボックスがオンの場合は有効、オフの場合は無効になります。



チェックボックス列のヘッダをクリックすると、次の動作になります。

- 各送信メッセージのチェックボックスがすべてオンの場合は、すべてオフになります。
- いずれかの送信メッセージのチェックボックスがオフの場合は、すべてオンになります。

④メッセージパラメータ設定

メッセージ一覧で選択した送信メッセージの詳細を設定します。

CAN ID

PGN： 送信するメッセージのPGNを設定します。設定できるPGNの範囲は、0～61184 (0000h～EF00h) (PDU1フォーマット)、61440～65535 (F000h～FFFFh) (PDU2フォーマット)です。PDU1フォーマットで送信先アドレスを示す下位1バイト (PDU Specific) は00hとしてください。PDU1フォーマットの範囲で設定した場合、送信先アドレスが255 (グローバルアドレス) のメッセージとして送信します。
[...] をクリックして表示されるPGNマネージャーでPGNを選択することもできます。詳細は、「PGNマネージャー」(8-29頁)を参照してください。

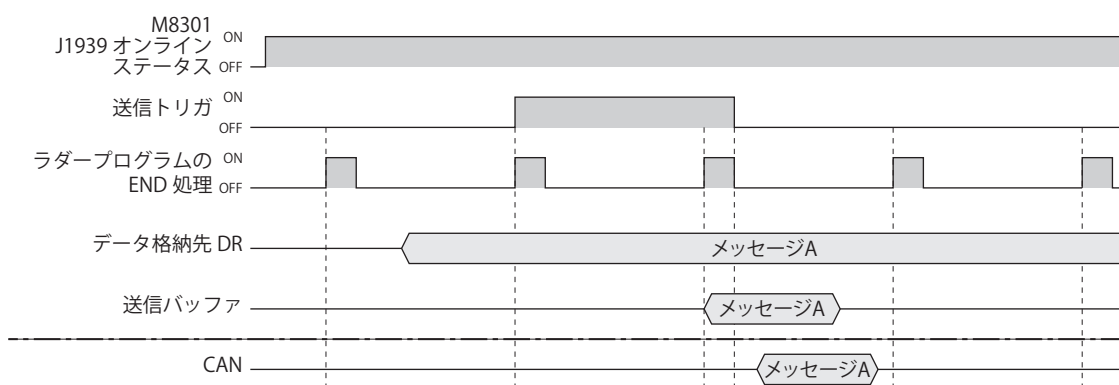
優先度： 送信するメッセージの優先度を0～7の範囲で設定します。

タイプ

メッセージの送信方法を設定します。“イベント”または“周期”から選択します。

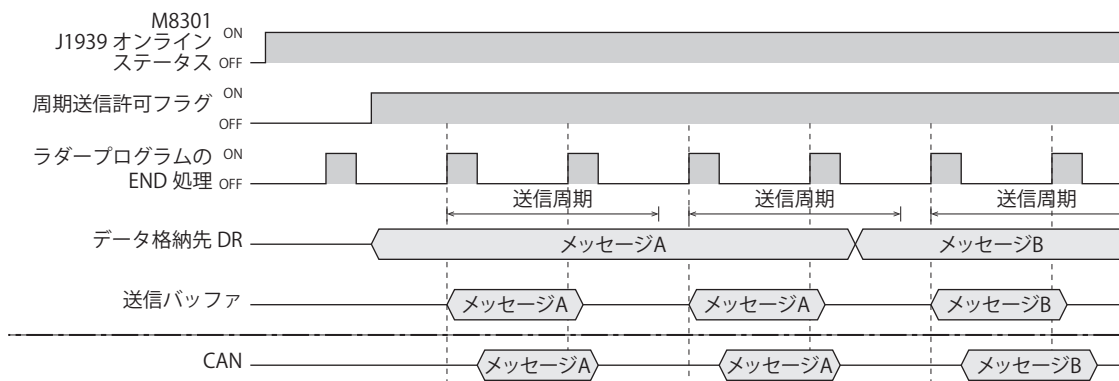
イベント： 送信トリガをONすると、データ格納先に格納した送信データがラダープログラムのEND処理で送信されます。

CA非指定時のイベント送信例



周期： 周期送信許可フラグをONしている間、送信周期になると、データ格納先に格納した送信データが、ラダープログラムのEND処理で送信されます。実際に送信する周期は、ラダープログラムのスキャン時間に影響されます。

CA非指定の周期送信例



最大データ長

送信するPGNの最大データ長を設定します。設定できる最大値は252バイトです。

送信周期 (ms)

[タイプ]で“周期”を選択した場合に、送信周期を10ミリ秒単位で設定します。設定できる値の範囲は10～655350です。

データ格納先

送信メッセージを格納先するデータレジスタを設定します。設定したデバイスを先頭に最大データ長分を占有します。メッセージを送信するときは、データ格納先からデータを内部の送信バッファにコピーしてから送信します。

アドレス： デバイスのタグ名、またはデバイスアドレスに対応するデバイスアドレスを表示します。

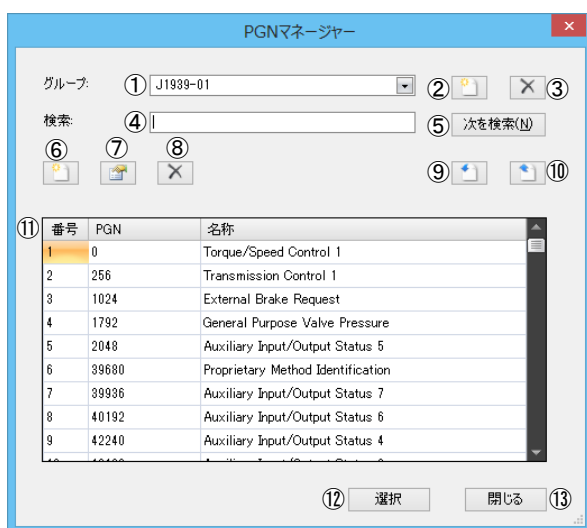
コメント： デバイスのタグ名、またはデバイスアドレスに対応するコメントを表示します。編集可能です。

コメント

送信メッセージのコメントを設定します。設定できるコメントは最大 127 バイトです。

PGN マネージャー

送受信メッセージの PGN は、PGN マネージャーで一括管理できます。



① グループリスト

登録されている PGN のグループが表示されます。

グループを変更すると、選択したグループに登録されている PGN が PGN 一覧 (⑪) に表示されます。

あらかじめ "J1939" が登録されています。"J1939" 含む最大 20 個のグループを登録できます。

② グループ追加ボタン

グループを追加します。

このボタンをクリックすると、PGN グループエディタが表示されます。PGN グループエディタで新しいグループを追加します。詳細は、「PGN グループエディタ」(8-31 頁) を参照してください。

追加したグループは、グループリストに追加されます。

③ グループ削除ボタン

グループを削除します。

グループリストからグループを選択して、このボタンをクリックすると、グループが削除されます。

デフォルトのグループ "J1939" は削除できません。

④ 検索

PGN 一覧内で検索する文字列を最大 128 バイトで入力します。

⑤ [次を検索] ボタン

[検索] で入力した文字列を PGN 一覧内で検索します。[検索] で文字列を入力し、このボタンをクリックします。

⑥ PGN 追加ボタン

グループに PGN を追加します。

グループリストで PGN を追加するグループを選択し、このボタンをクリックすると、PGN エディタが表示されます。

PGN エディタで新しい PGN を追加します。詳細は、「PGN エディタ」(8-31 頁) を参照してください。

追加した PGN は、PGN 一覧に追加されます。

⑦ PGN 編集ボタン

PGN を編集します。

グループリストで編集する PGN が登録されているグループを選択し、PGN 一覧で編集する PGN を選択します。このボタンをクリックすると、PGN エディタが表示されます。

PGN エディタで PGN を編集します。詳細は、「PGN エディタ」(8-31 頁) を参照してください。

PGN 一覧で選択した PGN の内容が更新されます。

⑧ PGN 削除ボタン

PGN を削除します。

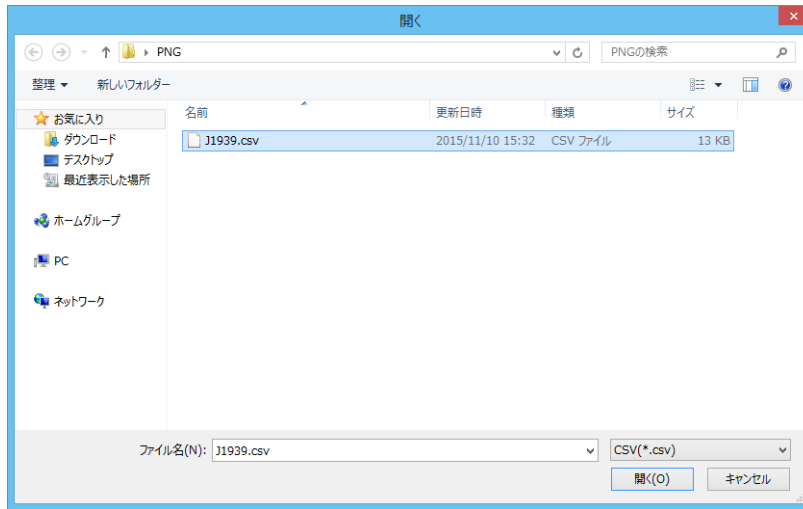
グループリストで削除する PGN が登録されているグループを選択し、PGN 一覧で削除する PGN を選択します。このボタンをクリックすると、PGN が削除されます。

⑨ PGN インポートボタン

CSV 形式のファイルとして保存した PGN を取り込みます。

グループリストで PGN を取り込むグループを選択します。選択したグループに既に登録されている PGN は、確認メッセージが表示されますので、[OK] ボタンをクリックすると、インポート実行前にすべて削除されます。

このボタンをクリックすると、[開く] ダイアログボックスが表示されます。

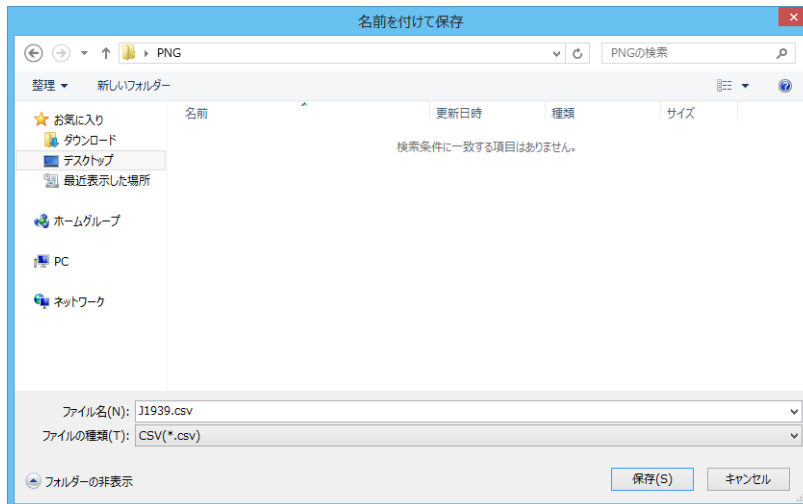


PGN を保存した CSV 形式のファイルを選択し、[開く] ボタンをクリックすると、ファイル内の PGN をグループリストで選択しているグループの PGN 一覧に取り込みます。

⑩ PGN エクスポートボタン

グループリストで選択しているグループに登録されている PGN を CSV 形式のファイルとして保存します。

このボタンをクリックすると、[名前を付けて保存] ダイアログボックスが表示されます。



保存する場所を選択し、ファイル名を入力して [保存] ボタンをクリックすると、グループに登録されている PGN を CSV 形式で保存します。

⑪ PGN 一覧

グループリストで選択したグループに登録されている PGN が表示されます。

No. : 登録されている PGN の管理番号 (1 ~ 65535) が表示されます。

PGN : PGN が表示されます。

名称 : PGN の名称が表示されます。

⑫ [選択] ボタン

PGN マネージャーを閉じ、呼び出し元のメッセージパラメータ設定の [CAN ID] に PGN 一覧で選択した PGN を設定します。

⑬ [閉じる] ボタン

PGN マネージャーを閉じます。

■ PGN グループエディタ

新しいグループを追加します。

① グループ名

PGN を登録するグループのグループ名を最大 64 バイトで入力します。

② [追加] ボタン

PGN グループエディタを閉じ、PGN マネージャーのグループリストに追加します。

③ [閉じる] ボタン

PGN グループエディタを閉じます。

■ PGN エディタ

新しい PGN を追加または既存の PGN を編集します。

① PGN

登録する PGN を最大 5 文字で入力します。

[16 進数で入力する] チェックボックスがオフの場合は 10 進数、オンの場合は 16 進数で入力してください。

例) 16 進数の場合：0001、FF00 など

② 名称

登録する PGN の名称を最大 128 バイトで入力します。

③ [PGN を 16 進数で入力する] チェックボックス

PGN の入力形式を設定します。

10 進数で入力する場合はオフ、16 進数で入力する場合はオンにします。

④ [追加] ボタン

PGN エディタを閉じ、PGN マネージャーのグループリストに追加します。

⑤ [閉じる] ボタン

PGN エディタを閉じます。

無線通信仕様

形番	FC6A-PC4
Bluetooth 規格	Bluetooth ver.2.1 + EDR
プロファイル	SPP (Serial Port Profile) iAP (iPod Accessory Protocol)
周波数帯域	2402MHz~2480MHz
伝送距離 *1	10m (Class 2)
マルチペアリング台数	8台
通信プロトコル	メンテナンス通信、ユーザー通信
Bluetooth 無線許可取得済国 *2	日本 アメリカ カナダ 欧州 中国 オーストラリア ニュージーランド

*1 Bluetooth 通信は、障害物（人体、金属、壁など）や電波状態によって、接続有効範囲が変動します。ご採用の際には、事前にお使いになる環境下でご試用の上、ご採用ください。

*2 国によっては、別途装置の評価が必要な場合があります。



メンテナンス通信の場合の通信パフォーマンス（所要時間）は、次のとおりです。

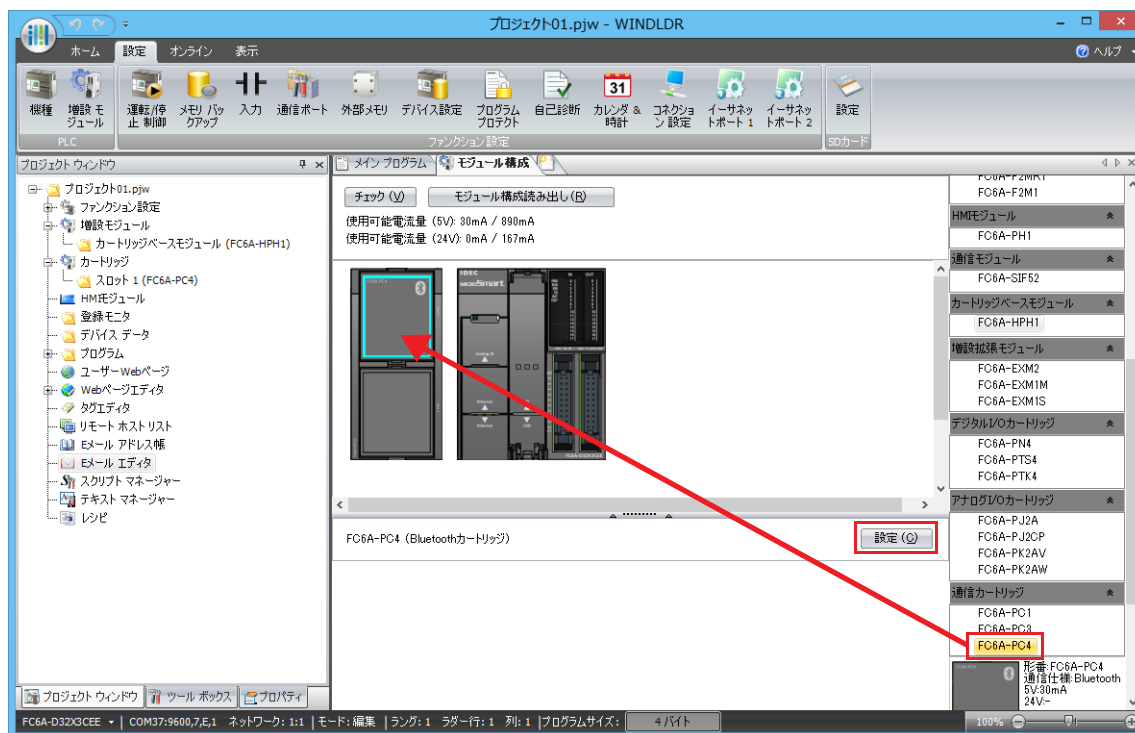
- 10000 ステップ相当のユーザープログラムアップロード：およそ 2分 40 秒
- 10000 ステップ相当のユーザープログラムダウンロード：およそ 2分 30 秒
- 20000 ステップ相当のユーザープログラムアップロード：およそ 5分 20 秒
- 20000 ステップ相当のユーザープログラムダウンロード：およそ 5分
- 100KB の CSV ファイルの読み出し：およそ 1分 30 秒
- 200KB の CSV ファイルの読み出し：およそ 3分

Bluetooth 通信カートリッジの設定

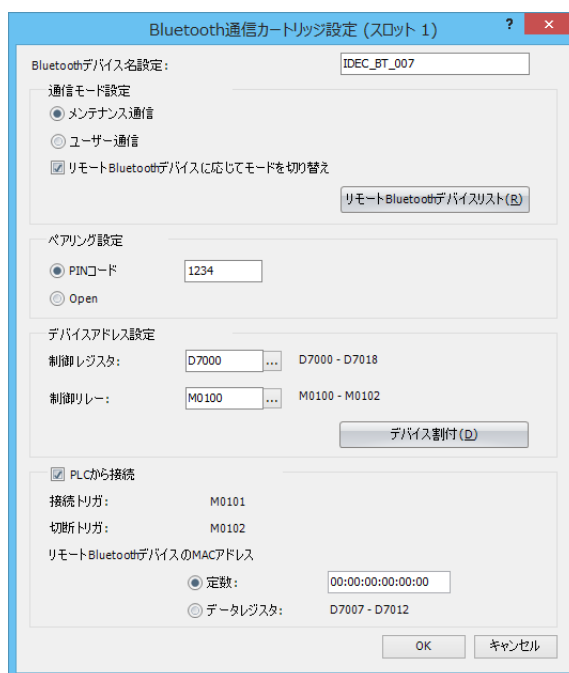
WindLDR の設定

●操作手順

- Bluetooth 通信カートリッジの設定は、モジュール構成エディタで行います。
[設定] タブの [PLC] で [増設モジュール] をクリックします。
- モジュール構成エリアに挿入した Bluetooth 通信カートリッジをクリックし、[設定] ボタンをクリックします。
[Bluetooth 通信カートリッジ設定] ダイアログボックスが表示されます。



- Bluetooth 通信カートリッジの各設定項目を設定します。



- ユーザープログラムを FC6A 形にダウンロードします。

以上で、Bluetooth 通信カートリッジの設定が完了します。

■ Bluetooth 通信カートリッジ設定

① Bluetooth デバイス名設定

パソコンやスマートフォンから周囲の Bluetooth 機器を検索をしたときに、パソコンやスマートフォンに表示される名前を設定します。

② 通信モード設定

Bluetooth 通信時の FC6A 形の通信モードを “メンテナンス通信” または “ユーザー通信” から選択します。

リモート Bluetooth デバイスに応じてモードを切り替え (③) を設定している場合は、リモート Bluetooth デバイスに応じてモードを切り替え (③) の設定が優先されます。

③ リモート Bluetooth デバイスに応じてモードを切り替え

接続する Bluetooth 機器によって FC6A 形の通信モードを切り替える場合は、このチェックボックスをオンにします。

例えば、FC6A 形をスマートフォンと接続するとメンテナンス通信になり、スマートフォンと切断後 FC6A 形をバーコードリーダーと接続するとユーザー通信になるように設定したい場合に使用します。

このチェックボックスがオフの場合、またはオンの場合でも [リモート Bluetooth デバイスリスト] ダイアログボックスで接続先の Bluetooth 機器を登録していない場合は、FC6A 形は通信モード設定 (②) で設定した通信モードで動作します。

リモート Bluetooth デバイスに応じてモードを切り替え (③) を設定している場合、FC6A 形が Bluetooth 機器との接続開始時に通信モードを判断します。そのため、FC6A 形は Bluetooth 接続開始後数秒間 SPP データを受信できません。通信モードを切り替えたときに、制御レジスタ (⑥) の Bluetooth 通信カートリッジ (カートリッジスロット) のステータスが変更されると SPP 通信ができます。

④ [リモート Bluetooth デバイスリスト] ボタン

リモート Bluetooth デバイスに応じてモードを切り替え (③) で使用する接続先の Bluetooth 機器の MAC アドレスと切り替える通信モードを設定します。

このボタンをクリックすると [リモート Bluetooth デバイスリスト] ダイアログボックスが表示されます。

ダイアログボックスには、各 Bluetooth 機器の MAC アドレス、通信モード、コメントが表示されます。詳細は、「リモート Bluetooth デバイスリスト」(9-6 頁) を参照してください。

⑤ペアリング設定

Bluetooth 接続時のセキュリティを次の中から選択します。

PIN コード : Bluetooth 機器との接続時に、設定した PIN コードの入力を要求します。通信内容は暗号化されます。
一般的に Bluetooth v2.0 までの機器で採用されているペアリング方式です。

Open : 通信内容は暗号化されません。

Bluetooth v2.0 までの機器でセキュリティなしの場合に使用する設定です。

⑥制御レジスタ

FC6A 形に接続した Bluetooth 通信カートリッジや接続先の Bluetooth 機器の MAC アドレスを格納するデータレジスタの先頭アドレスを設定します。特殊データレジスタは設定できません。

デバイスアドレス	項目	内容	R/W
先頭番号+0	Bluetooth通信カートリッジ (カートリッジスロット) のステータス	Bluetooth通信のステータスと通信モードを格納します。 0: 無処理 255: 処理中 (Bluetooth接続待ち/切断待ちなど) 1: メンテナンス通信中 2: ユーザー通信中 デフォルト値は0 (無処理) です。	R
先頭番号+1	Bluetooth通信カートリッジ (カートリッジスロット) の MACアドレス	カートリッジスロットに接続したBluetooth通信カートリッジのMACアドレスを格納します。 例) MACアドレス: AA-BB-CC-DD-EE-FF 先頭番号+1=AA、先頭番号+2=BB、先頭番号+3=CC、 先頭番号+4=DD、先頭番号+5=EE、先頭番号+6=FF	R
先頭番号+2			
先頭番号+3			
先頭番号+4			
先頭番号+5			
先頭番号+6			
先頭番号+7	Bluetooth通信カートリッジ (カートリッジスロット) の接続トリガ用の接続先の Bluetooth機器のMACアドレス	PLCから接続 (⑨) の [接続先のBluetooth MACアドレス] で"データレジスタ"を選択した場合に、接続対象となるBluetooth機器のMACアドレスを格納します。 例) MACアドレス: AA-BB-CC-DD-EE-FF 先頭番号+1=AA、先頭番号+2=BB、先頭番号+3=CC、 先頭番号+4=DD、先頭番号+5=EE、先頭番号+6=FF	R/W
先頭番号+8			
先頭番号+9			
先頭番号+10			
先頭番号+11			
先頭番号+12	リザーブ		-
先頭番号+13			
先頭番号+14			
先頭番号+15			
先頭番号+16			
先頭番号+17			
先頭番号+18			

⑦制御リレー

Bluetooth 通信カートリッジ (カートリッジスロット) の通信状態、PLC から接続 (⑨) の接続トリガと切断トリガを格納する内部リレーの先頭アドレスを設定します。特殊内部リレーは設定できません。

デバイスアドレス	項目	内容	R/W
先頭番号+0	Bluetooth通信カートリッジ (カートリッジスロット) のステータス	0: 接続先のBluetooth機器 と通信できない状態 1: 接続先のBluetooth機器 と通信できる状態	R
先頭番号+1	Bluetooth通信カートリッジ (カートリッジスロット) の接続トリガ	0: 処理しない 1: 接続対象となるBluetooth機器を接続する	R/W
先頭番号+2	Bluetooth通信カートリッジ (カートリッジスロット) の切断トリガ	0: 処理しない 1: 接続対象となるBluetooth機器を切断する	R/W

⑧ [デバイス割付] ボタン

このボタンをクリックすると、[デバイス割付] ダイアログボックスが表示されます。

ダイアログボックスには、Bluetooth 通信カートリッジの各設定の内容とデータレジスタ、内部リレーの対応表が表示されます。詳細は、「デバイス割付」(9-7 頁) を参照してください。

⑨ PLC から接続

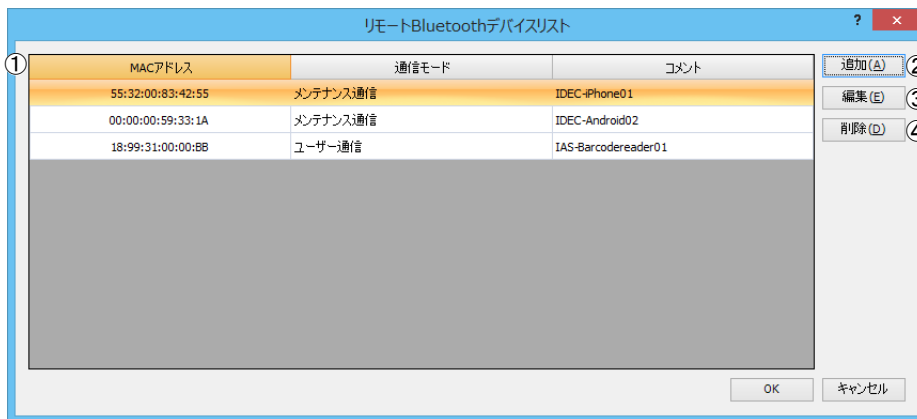
FC6A 形がマスターのときに、接続対象となる Bluetooth 機器を接続および切断する場合は、このチェックボックスをオンにします。チェックボックスがオンの場合は、iOS 機器と接続できません。

接続トリガ : 制御リレー (⑦) で設定したデバイスアドレス+1の接続トリガが表示されます。OFF から ON になると、[接続先の Bluetooth MAC アドレス] で設定した接続対象となる Bluetooth 機器を接続します。

切断トリガ : 制御リレー (⑦) で設定したデバイスアドレス+2の切断トリガが表示されます。OFF から ON になると、[接続先の Bluetooth MAC アドレス] で設定した接続対象となる Bluetooth 機器を切断します。

リモート Bluetooth デバイスの MAC アドレス : 接続対象を“定数”または“データレジスタ”から選択します。
 “定数”を選択した場合は、[Bluetooth 通信カートリッジ設定] ダイアログで設定した MAC アドレスが接続対象となります。
 “データレジスタ”を選択した場合は、制御デバイス (⑥) の先頭番号+7 ~+12 で設定した Bluetooth 機器の MAC アドレスが接続対象となります。

■ リモート Bluetooth デバイスリスト



① リモート Bluetooth デバイス一覧

各 Bluetooth デバイスの MAC アドレス、通信モード、コメントが表示されます。

② 追加

このボタンをクリックすると、[リモート Bluetooth デバイス設定] ダイアログボックスが表示されます。接続先の Bluetooth デバイスの MAC アドレス、接続開始時に切り替える通信モード、コメントを入力します。

最大 20 台まで登録できます。詳細は、「リモート Bluetooth デバイス設定」(9-7 頁) を参照してください。

③ 編集

リモート Bluetooth デバイス一覧 (①) で Bluetooth デバイスを選択し、このボタンをクリックすると、[リモート Bluetooth デバイス設定] ダイアログボックスが表示されます。接続先の Bluetooth デバイスの MAC アドレス、接続開始時に切り替える通信モード、コメントを編集します。詳細は、「リモート Bluetooth デバイス設定」(9-7 頁) を参照してください。

④ 削除

リモート Bluetooth デバイス一覧 (①) で Bluetooth デバイスを選択し、このボタンをクリックすると、Bluetooth デバイスが削除されます。

■ リモート Bluetooth デバイス設定

- ① MAC アドレス
リモート Bluetooth デバイスの MAC アドレスを設定します。
- ② 通信モード
接続開始時に切り替える通信モードを “メンテナンス通信” または “ユーザー通信” から選択します。
- ③ コメント
リモート Bluetooth デバイスのコメントを入力します。

■ デバイス割付

デバイスアドレス	内容
D7000	Bluetooth通信カードリッジ (スロット 1) ステータス
D7001	Bluetooth通信カードリッジ (スロット 1) MACアドレ...
D7002	Bluetooth通信カードリッジ (スロット 1) MACアドレ...
D7003	Bluetooth通信カードリッジ (スロット 1) MACアドレ...
D7004	Bluetooth通信カードリッジ (スロット 1) MACアドレ...
D7005	Bluetooth通信カードリッジ (スロット 1) MACアドレ...
D7006	Bluetooth通信カードリッジ (スロット 1) MACアドレ...
D7007	Bluetooth通信カードリッジ (スロット 1) 相手デバイ...
D7008	Bluetooth通信カードリッジ (スロット 1) 相手デバイ...
D7009	Bluetooth通信カードリッジ (スロット 1) 相手デバイ...
D7010	Bluetooth通信カードリッジ (スロット 1) 相手デバイ...
D7011	Bluetooth通信カードリッジ (スロット 1) 相手デバイ...
D7012	Bluetooth通信カードリッジ (スロット 1) 相手デバイ...

- ① デバイス割付一覧
Bluetooth 通信カートリッジの各設定の内容とデータレジスタ、内部リレーの対応表が表示されます。
- ② [コメント割付] ボタン
このボタンをクリックすると、対応表の内容が、各デバイスのコメントとして設定されます。

Bluetooth 機器との設定と通信

パソコンとのメンテナンス通信

Bluetooth 通信機能を使用して、パソコンとメンテナンス通信する方法について説明します。

WindLDR がインストールされたパソコンと接続し、デバイス値のモニタや変更、ユーザープログラムのダウンロードやアップロードができます。

下図のように、パソコンをマスター、FC6A 形をスレーブとして接続します。



FC6A 形の設定

1. FC6A 形に Bluetooth 通信カートリッジを接続します。
2. WindLDR で Bluetooth 通信カートリッジを設定します。
詳細は、「WindLDR の設定」(9-3 頁) を参照してください。
例) [Bluetooth 通信カートリッジ設定] ダイアログボックスの設定内容

設定項目		内容
Bluetoothデバイス名設定		IDEC_BT_007
通信モード設定		メンテナンス通信
ペアリング設定		PINコード"1234"
デバイスアドレス設定	制御レジスタ	D7000
	制御リレー	M0100

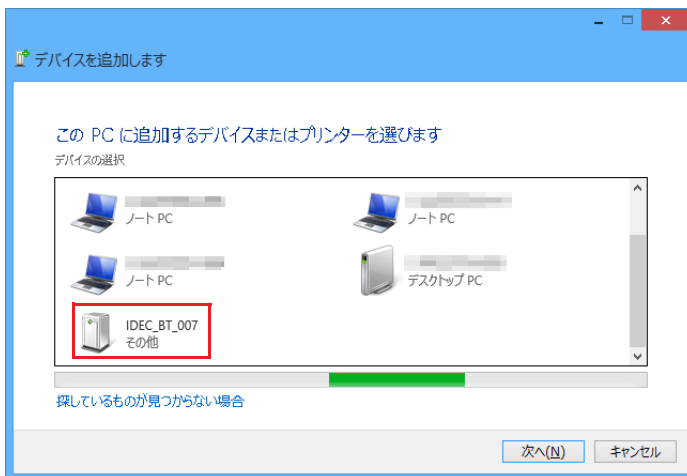
3. ユーザープログラムを FC6A 形にダウンロードします。

パソコンの設定

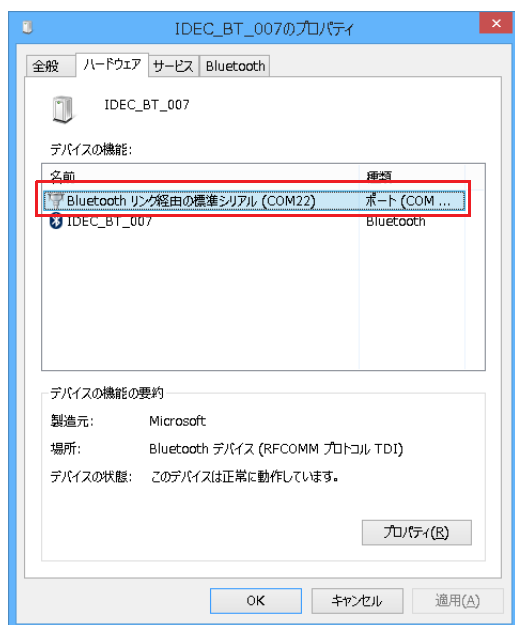
1. パソコンで Bluetooth 通信カートリッジの名前を検索します。

WindLDR の [Bluetooth 通信カートリッジ設定] ダイアログボックスの [Bluetooth デバイス名設定] で設定した名前になります。

(IDEC_BT_007)

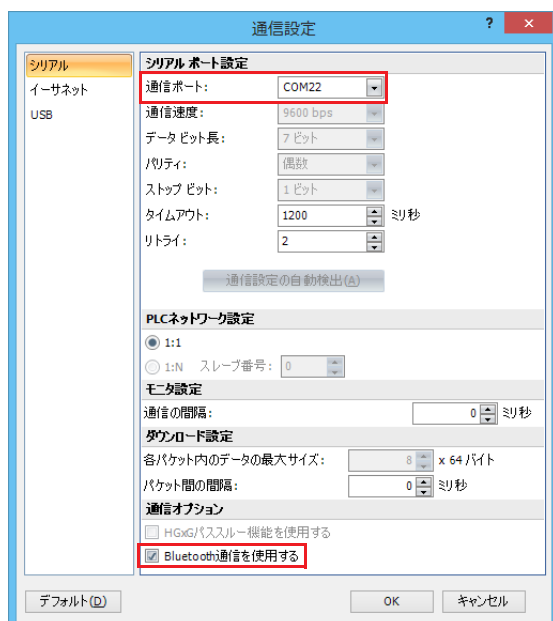


- Bluetooth 通信カートリッジに接続し、仮想 COM ポートを生成します。初回接続時はペアリングが必要となり PIN コードの入力を要求されますので、“1234”を入力します。



Bluetooth 通信の確認

- 仮想 COM ポートを使用し、シリアル通信を行います。
[Bluetooth 通信を使用する] チェックボックスをオンにします。



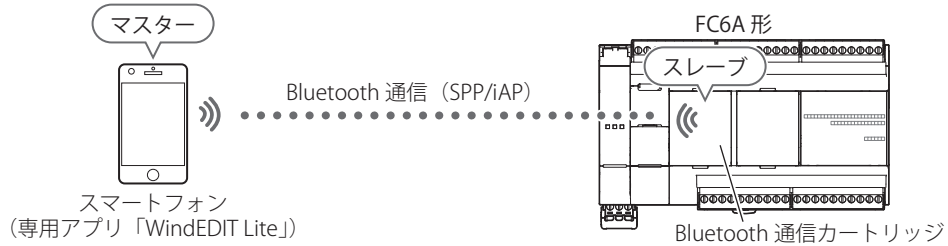
- WindLDR でモニタを実行します。
- Bluetooth 通信状態を示すデバイスの値を確認します。
[Bluetooth 通信カートリッジ設定] ダイアログボックスの [デバイス設定] で設定した制御リレーの先頭番号 + 0 : Bluetooth 通信カートリッジ (カートリッジスロット) のステータス (M0100) が 1 になっていることを確認します。

スマートフォンとのメンテナンス通信

Bluetooth 通信を使用して、スマートフォンとメンテナンス通信する方法について説明します。

専用アプリがインストールされたスマートフォンと接続し、デバイス値のモニタや変更、ユーザープログラムのダウンロードやアップロードができます。

下図のようにスマートフォンをマスター、FC6A 形をスレーブとして接続します。



また、パスワード認証を用いることでアクセス制限をかけることができますので、不特定多数のアクセスを遮断できます。

FC6A 形の設定

1. FC6A 形に Bluetooth 通信カートリッジを接続します。
2. WindLDR で Bluetooth 通信カートリッジを設定します。
詳細は、「WindLDR の設定」(9-3 頁)を参照してください。
例) [Bluetooth 通信カートリッジ設定] ダイアログボックスの設定内容

設定項目		内容
Bluetoothデバイス名設定		IDEC_BT_007
通信モード設定		メンテナンス通信
ペアリング設定		PINコード"1234"
デバイスアドレス設定	制御レジスタ	D7000
	制御リレー	M0100

3. ユーザープログラムを FC6A 形にダウンロードします。

スマートフォンの設定

1. スマートフォンで Bluetooth 通信カートリッジの名前を検索します。
WindLDR の [Bluetooth 通信カートリッジ設定] ダイアログボックスの [Bluetooth デバイス名設定] で設定した名前になります。
(IDEC_BT_007)



お使いのスマートフォンの機種によっては Bluetooth 通信デバイスの検索画面は異なります。

2. 専用アプリ「WindEDIT Lite」から Bluetooth 通信カートリッジに接続します。



Bluetooth 通信の確認

1. 専用アプリ「WindEDIT Lite」で Bluetooth 通信状態を示すデバイスの値を確認します。
[Bluetooth 通信カートリッジ設定] ダイアログボックスの [デバイスアドレス設定] で設定した制御リレーの先頭番号+0 : Bluetooth 通信カートリッジ (カートリッジスロット) のステータス (M0100) が1になっていることを確認します。



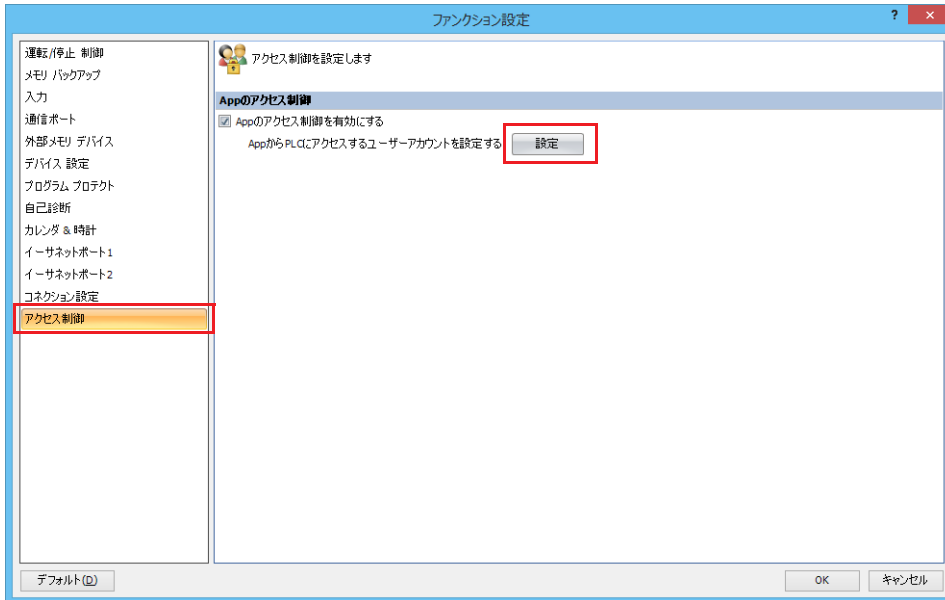
専用アプリ「WindEDIT Lite」の操作方法については、WindEDIT Lite ユーザーズマニュアルを参照してください。

ユーザーアカウント設定

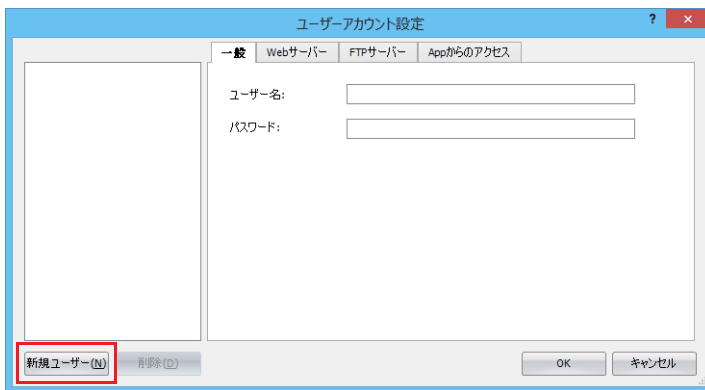
■ユーザーアカウントを作成する

●操作手順

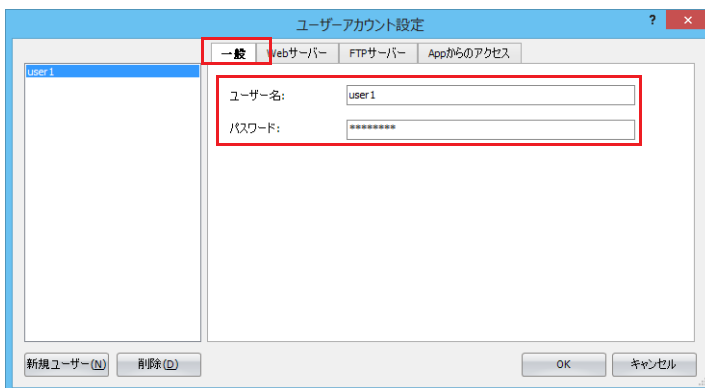
1. [ファンクション設定] ダイアログボックスの [アクセス制御] で [Appのアクセス制御を有効にする] チェックボックスをオンにします。
2. [設定] ボタンをクリックします。
[ユーザーアカウント設定] ダイアログボックスが表示されます。



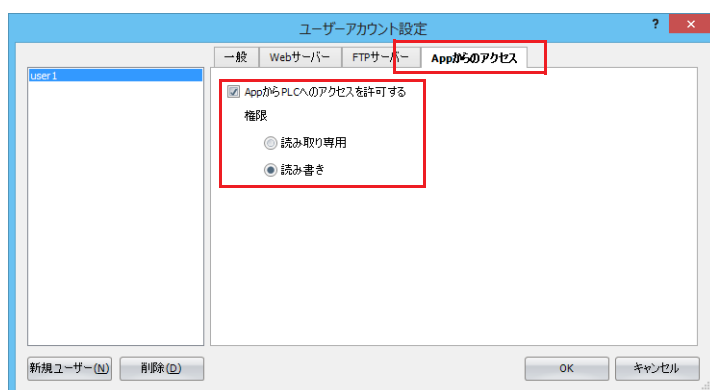
3. [新規ユーザー] ボタンをクリックします。
リストにユーザーアカウントが作成されます。



4. [一般] タブで [ユーザー名] と [パスワード] をそれぞれ設定します。



5. [Appからのアクセス] タブをクリックし、[AppからPLCへのアクセスを許可する] チェックボックスをオンにします。
6. [権限] でアカウントの権限を“読み取り専用”または“読み書き”から選択します。

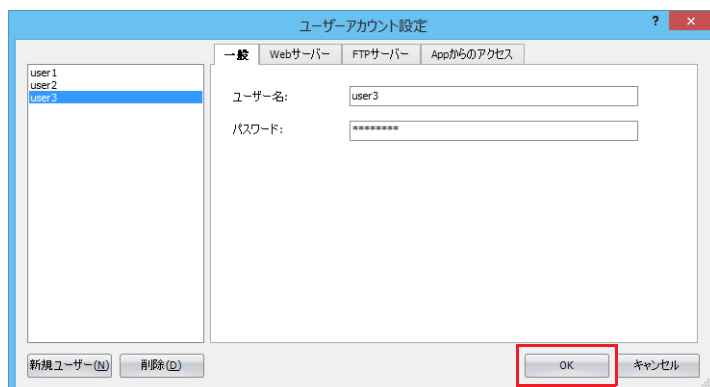


7. 手順2～6の操作を繰り返し、必要な数のユーザーアカウントを作成します。



リストでユーザーアカウントを選択すると、選択したユーザーアカウントの [一般] タブ、[Appからのアクセス] タブの内容を編集できます。

8. [OK] ボタンをクリックします。

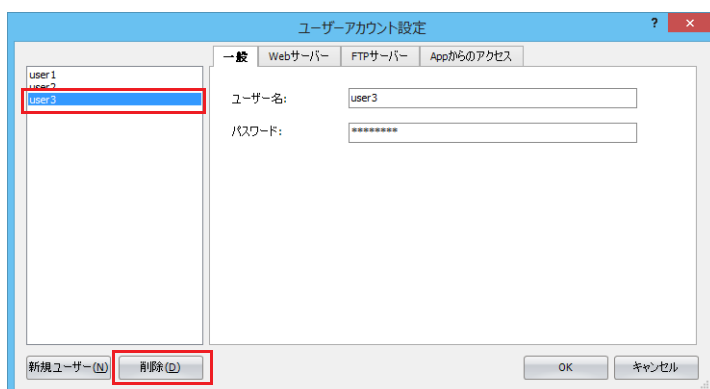


以上で、ユーザーアカウントの作成が完了します。

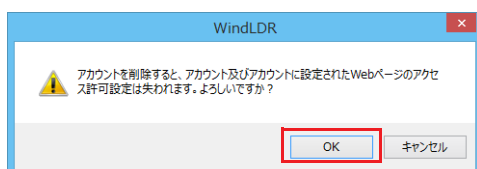
■ 作成したユーザーアカウントを削除する

● 操作手順

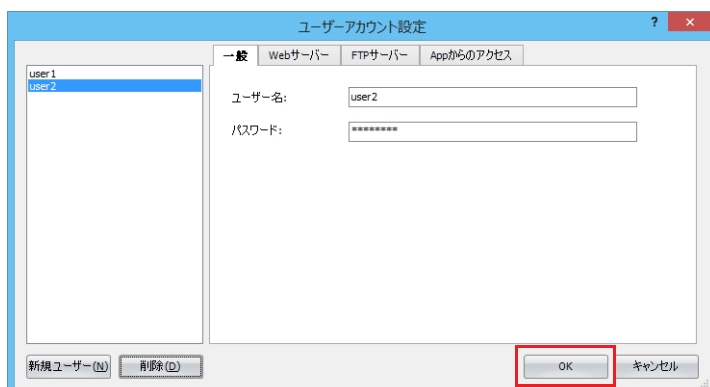
1. リストでユーザーアカウントを選択し、[削除] ボタンをクリックします。
確認メッセージが表示されます。



2. [OK] ボタンをクリックします。



3. [OK] ボタンをクリックします。



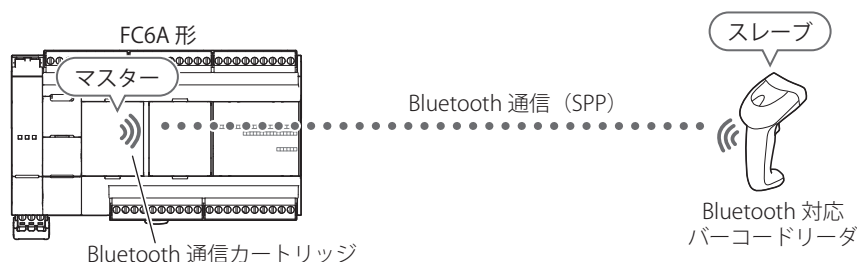
以上で、ユーザーアカウントの削除が完了します。

バーコードリーダーとのユーザー通信

Bluetooth 通信機能を使用して、バーコードリーダーとユーザー通信する方法について説明します。

FC6A 形が対応しているバーコードリーダーと接続し、ユーザー通信ができます。

使用するバーコードリーダーの仕様によって、設定方法は異なります。次の説明では、下図のように FC6A 形をマスター、バーコードリーダーをスレーブとして接続します。



推奨バーコードリーダーは GBT4400 (IDEC 製) です。

FC6A 形の設定

1. FC6A 形に Bluetooth 通信カートリッジを接続します。
2. WindLDR で Bluetooth 通信カートリッジを設定します。
詳細は、「WindLDR の設定」(9-3 頁) を参照してください。
例) [Bluetooth 通信カートリッジ設定] ダイアログボックスの設定内容

設定項目		内容
Bluetoothデバイス名設定		IDEC_BT_007
通信モード設定		ユーザー通信
ペアリング設定		PINコード"1234"
デバイスアドレス設定	制御レジスタ	D7000
	制御リレー	M0100
PLCから接続		オン ^{*1}

*1 接続には、バーコードリーダーの MAC アドレスが必要です。

3. ユーザープログラムを FC6A 形にダウンロードします。

バーコードリーダーの設定

1. バーコードリーダーの PIN コードを設定します。
バーコードリーダーで PIN コード設定用のバーコードを読み取り設定します。バーコードリーダーの設定方法は仕様によって異なります。
WindLDR の [Bluetooth 通信カートリッジ設定] ダイアログボックスの [ペアリング設定] で PIN コードになります。
(1234)
2. バーコードリーダーを接続待ち受け状態に設定します。
バーコードリーダーで接続待ち受け設定用のバーコードを読み取り設定します。バーコードリーダーの設定方法は仕様によって異なります。

Bluetooth 通信の確認

1. FC6A 形で接続トリガを ON にします。
[Bluetooth 通信カートリッジ設定] ダイアログボックスの [デバイス設定] で設定した制御リレーの先頭番号 + 1 : Bluetooth 通信カートリッジ (カートリッジスロット) の接続トリガ (M0101) を 1 にします。制御レジスタの先頭番号 + 0 : Bluetooth 通信カートリッジ (カートリッジスロット) のステータス (D7000) が 255 になります。
2. WindLDR でモニタを実行します。
3. Bluetooth 通信状態を示すデバイスの値を確認します。
[Bluetooth 通信カートリッジ設定] ダイアログボックスの [デバイス設定] で設定した制御リレーの先頭番号 + 0 : Bluetooth 通信カートリッジ (カートリッジスロット) のステータス (M0100) が 1 になっていることを確認します。

第10章 FTPサーバー / クライアント機能

この章では、Plus CPU モジュールの FTP サーバー機能および FTP クライアント機能について説明します。

FTP サーバー機能

FTP サーバー機能は、Plus CPU モジュールのみ使用できます。

FTP サーバー機能の概要

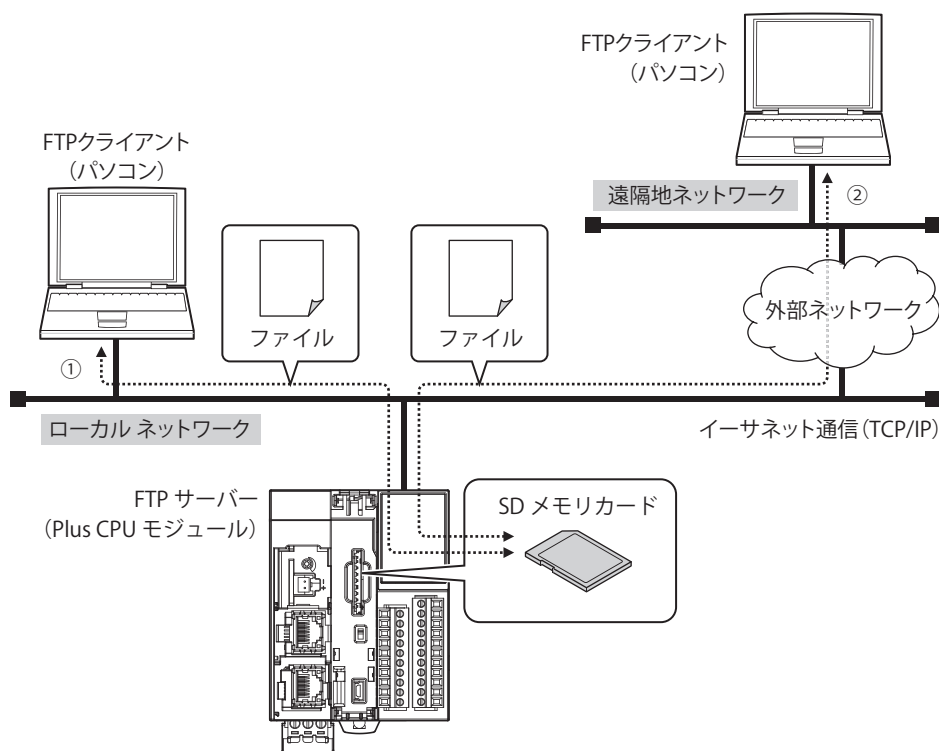
FTP サーバー機能でできること

FTP サーバー機能は、Plus CPU モジュールを FTP サーバーとして、パソコンなどの FTP クライアントと通信する機能です。FTP クライアントから次のことができます。

- Plus CPU モジュールに挿入した SD メモリカードからファイルを読み出す
- Plus CPU モジュールに挿入した SD メモリカードへファイルを書き込む

システム構成例

FTP サーバー機能を使用するためのシステム構成例を次に示します。



Plus CPU モジュールの Ethernet ポート 1 を ネットワークへ接続します。

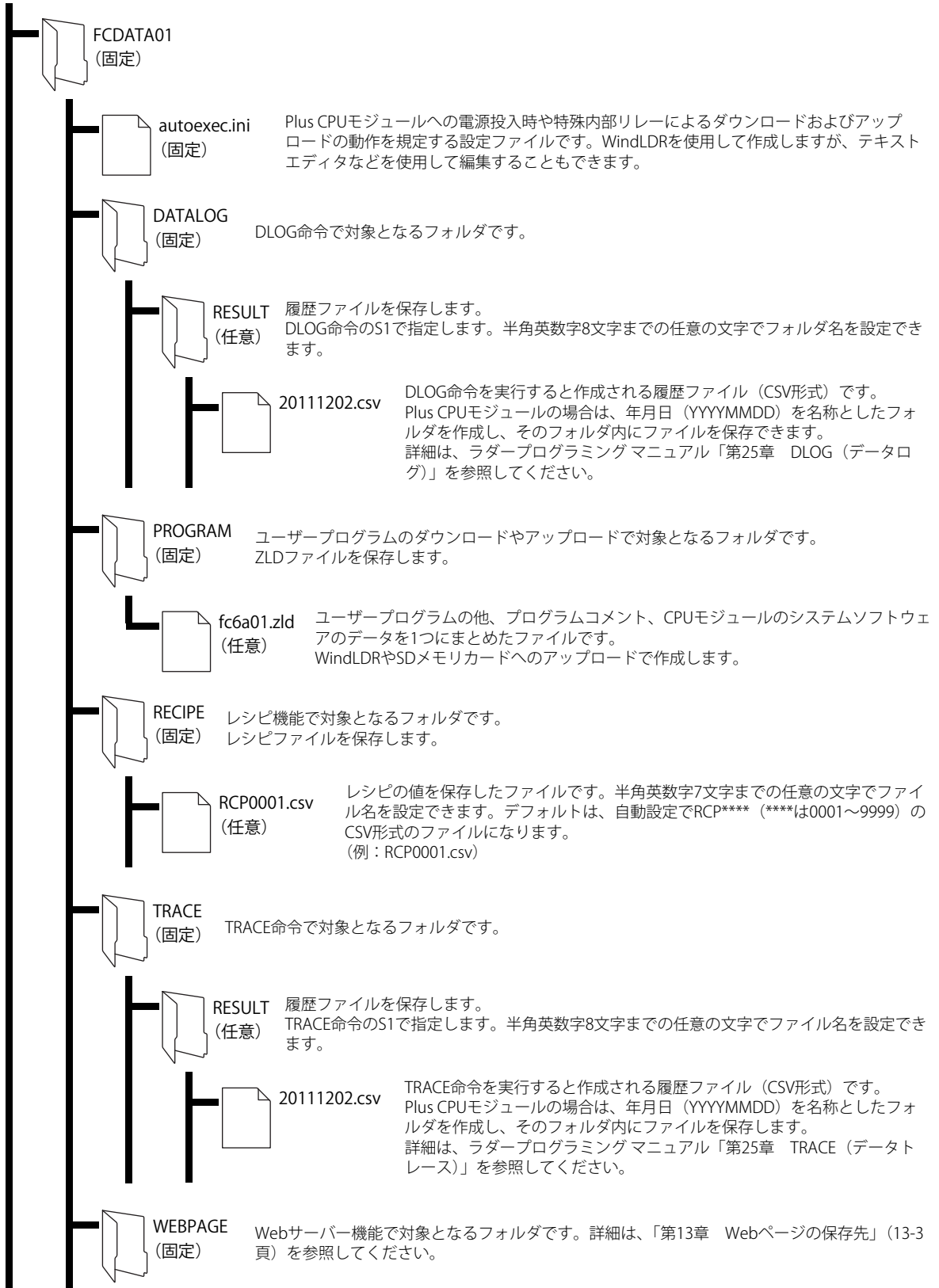
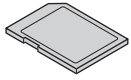
- ① ネットワークに接続している FTP クライアントから Plus CPU モジュールへアクセスし、Plus CPU モジュールに挿入した SD メモリカードのファイルを読み出したりは書き込みます。
- ② ネットワークを外部ネットワークに接続している場合は、遠隔地ネットワークに接続している FTP クライアントにローカルネットワークのゲートウェイ、ルータなどを設定します。遠隔地の FTP クライアントから Plus CPU モジュールへアクセスし、Plus CPU モジュールに挿入した SD メモリカードのファイルを読み出したりは書き込みます。ゲートウェイ、ルータなどの設定については、Plus CPU モジュールを接続したネットワークの管理者へお問い合わせください。

FTP サーバーで扱えるファイル

FTP サーバーでは、Plus CPU モジュールに挿入されている SD メモリカード内のすべてのファイルにアクセスできます。SD メモリカードのフォルダとファイル構成の一例を示します。

SD メモリカード

(任意) ファンクション設定や、命令のパラメータで指定可能なファイル名、フォルダ名です。
 (固定) システムで固定となっているファイル名、フォルダ名です。



対応コマンドおよび接続方式

- RFC959 で定義されたコマンドに対応しています。
- アクティブ モードおよびパッシブ モードに対応しています。
- FTPS (FTP over SSL/TLS) に対応しています。Explicit モードに対応しています。
- 同時に接続できる FTP クライアントは 1 台です。

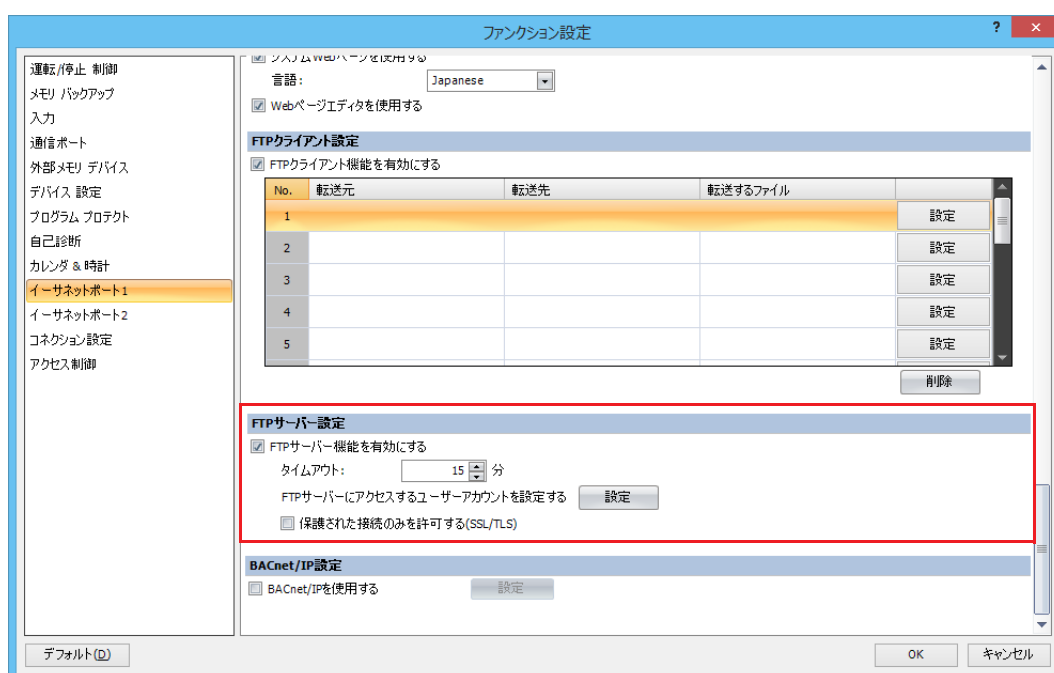
推奨 FTP クライアント ソフトウェアは、次のとおりです。

- FFFTP
- FileZilla
- Core FTP Lite

WindLDR の設定

●操作手順

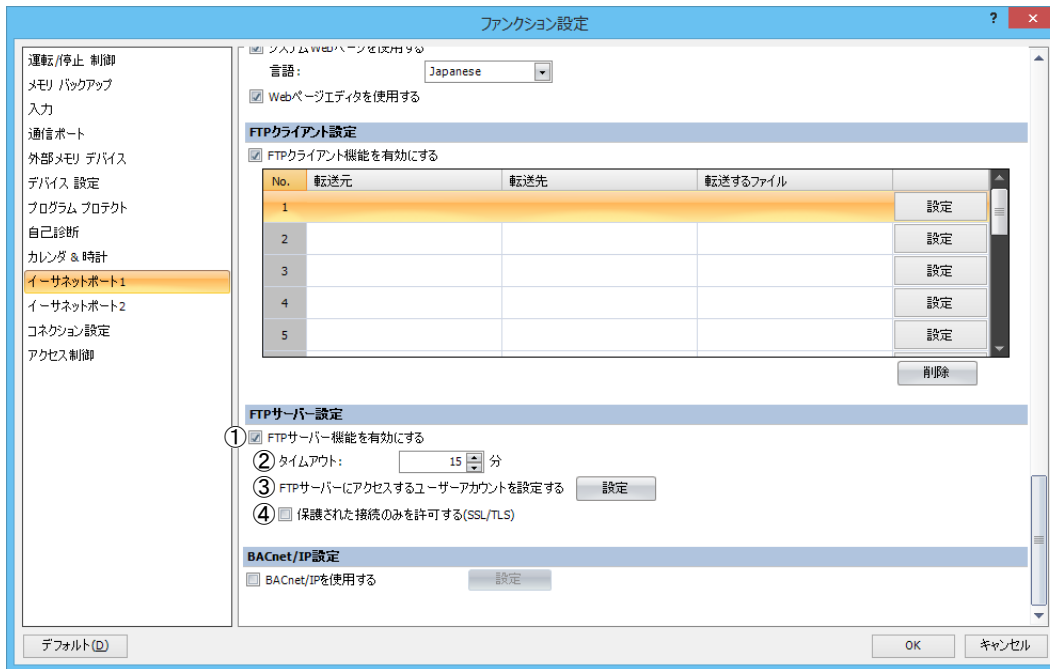
1. [設定] タブの [ファンクション設定] で [イーサネットポート 1] をクリックします。
2. [FTP サーバー設定] で [FTP サーバー機能を有効にする] チェックボックスをオンにします。
3. FTP サーバー設定の各設定項目を設定します。



4. ユーザープログラムを Plus CPU モジュールにダウンロードします。

以上で、FTP サーバーの設定が完了します。

FTP サーバー設定



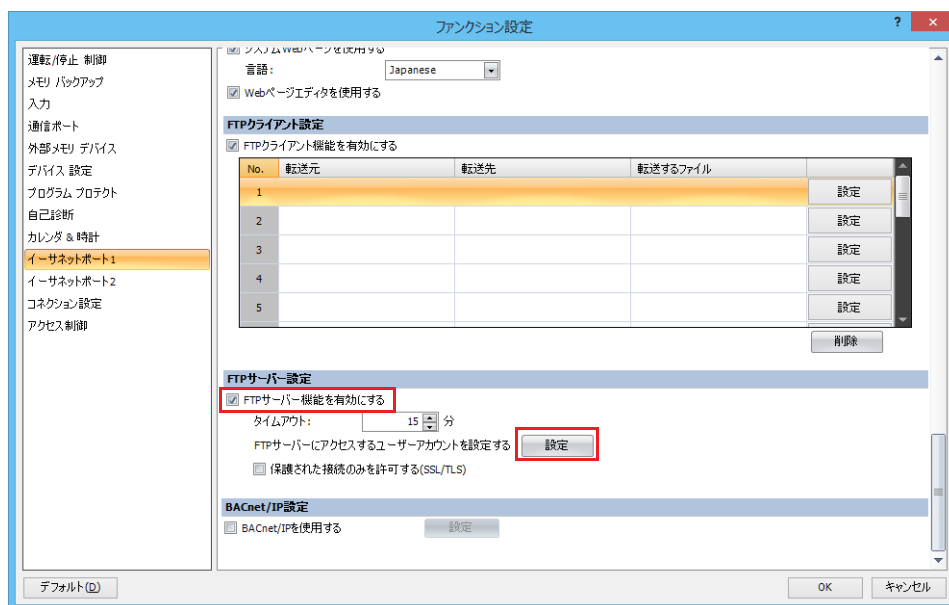
- ① **FTP サーバー機能を有効にする**
FTP サーバー機能を使用する場合は、このチェックボックスをオンにして、ユーザーアカウントを設定します。FTP クライアントから Plus CPU モジュールに挿入した SD メモリカードにあるファイルの読み出しまたは書き込みができます。FTP サーバー機能に使用するポート番号は 21 です。
- ② **タイムアウト**
Plus CPU モジュールにログインしたあと、Plus CPU モジュールと FTP クライアントの間で何も通信がない場合に自動で接続を切断するまでの時間を指定します。(デフォルト：15 分)
- ③ **FTP サーバーにアクセスするユーザーアカウントを設定**
[設定] ボタンをクリックし、表示される [ユーザーアカウント設定] ダイアログボックスでユーザーアカウントを作成してください。詳細は、「ユーザーアカウント設定」(10-5 頁) を参照してください。
- ④ **保護された接続のみを許可する (SSL/TLS)**
SSL 通信または TLS 通信でのみ FTP サーバーへのログインを許可する場合は、このチェックボックスをオンにします。

ユーザーアカウント設定

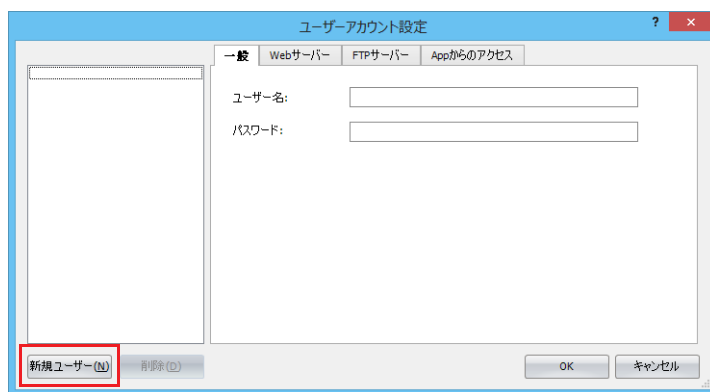
■ユーザーアカウントを作成する

●操作手順

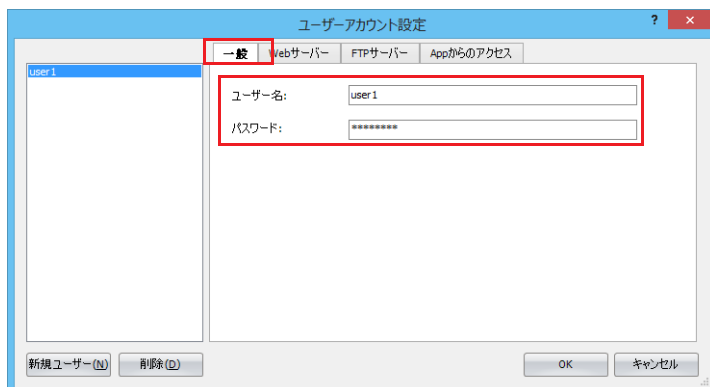
1. [ファンクション設定] ダイアログボックスの [FTP サーバー設定] で [FTP サーバー機能を有効にする] チェックボックスをオンにします。
2. [設定] ボタンをクリックします。
[ユーザーアカウント設定] ダイアログボックスが表示されます。



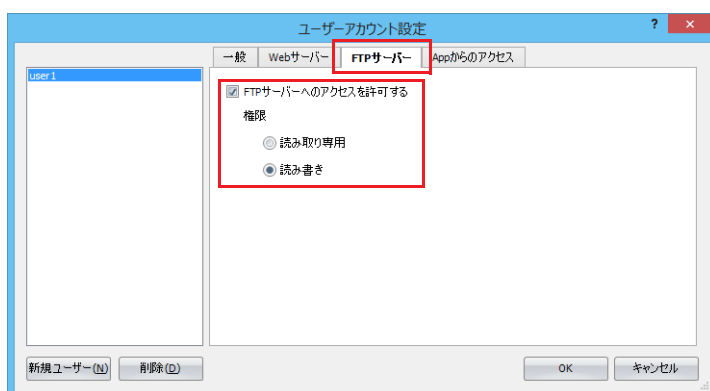
3. [新規ユーザー] ボタンをクリックします。
リストにユーザーアカウントが作成されます。



4. [一般] タブで [ユーザー名] と [パスワード] をそれぞれ設定します。



5. [FTP サーバー] タブをクリックし、[FTP サーバーへのアクセスを許可する] チェックボックスをオンにします。
 6. [権限] でアカウントの権限を“読み取り専用”または“読み書き”から選択します。

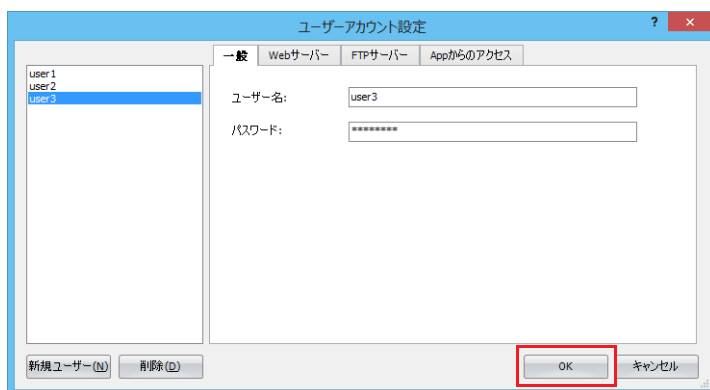


7. 手順 2～6 の操作を繰り返し、必要な数のユーザーアカウントを作成します。



リストでユーザーアカウントを選択すると、選択したユーザーアカウントの [一般] タブ、[FTP サーバー] タブの内容を編集できます。

8. [OK] ボタンをクリックします。

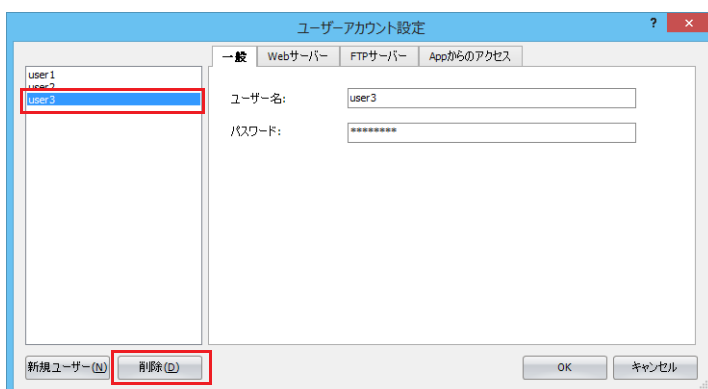


以上で、ユーザーアカウントの作成が完了します。

■ 作成したユーザーアカウントを削除する

● 操作手順

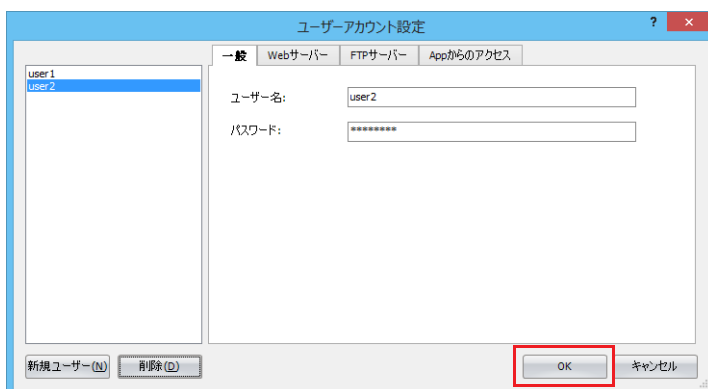
1. リストでユーザーアカウントを選択し、[削除] ボタンをクリックします。
確認メッセージが表示されます。



2. [OK] ボタンをクリックします。



3. [OK] ボタンをクリックします。



以上で、ユーザーアカウントの削除が完了します。

FTP クライアント機能

FTP クライアント機能は、Plus CPU モジュールのみ使用できます。

FTP クライアント機能の概要

FTP クライアント機能でできること

FTP クライアント機能は、Plus CPU モジュールを FTP クライアントとして、Plus CPU モジュールに挿入した SD メモリカードと FTP サーバーの間でファイルをコピーまたは移動する機能です。

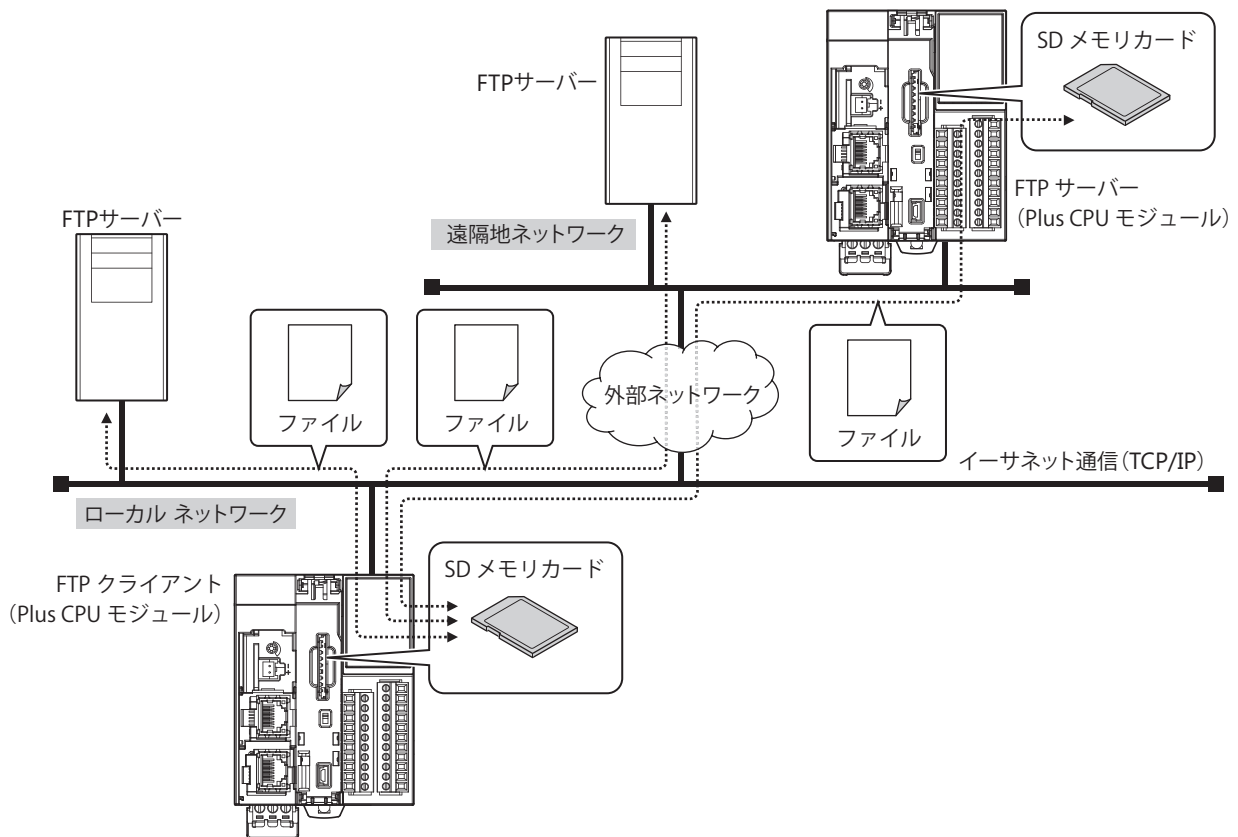
FTP クライアントとして動作することで、次のことができます。

- Plus CPU モジュールに挿入した SD メモリカードから FTP サーバーへファイルをコピーまたは移動する
- FTP サーバーから Plus CPU モジュールに挿入した SD メモリカードへファイルをコピーする

システム構成例

FTP クライアント機能を使用するためのシステム構成例を次に示します。

Plus CPU モジュールの Ethernet ポート 1 を ネットワークへ接続します。



Plus CPU モジュールのイーサネット設定 (IP アドレス、サブネット マスク、デフォルト ゲートウェイ) を行い、ローカル ネットワークへ接続します。

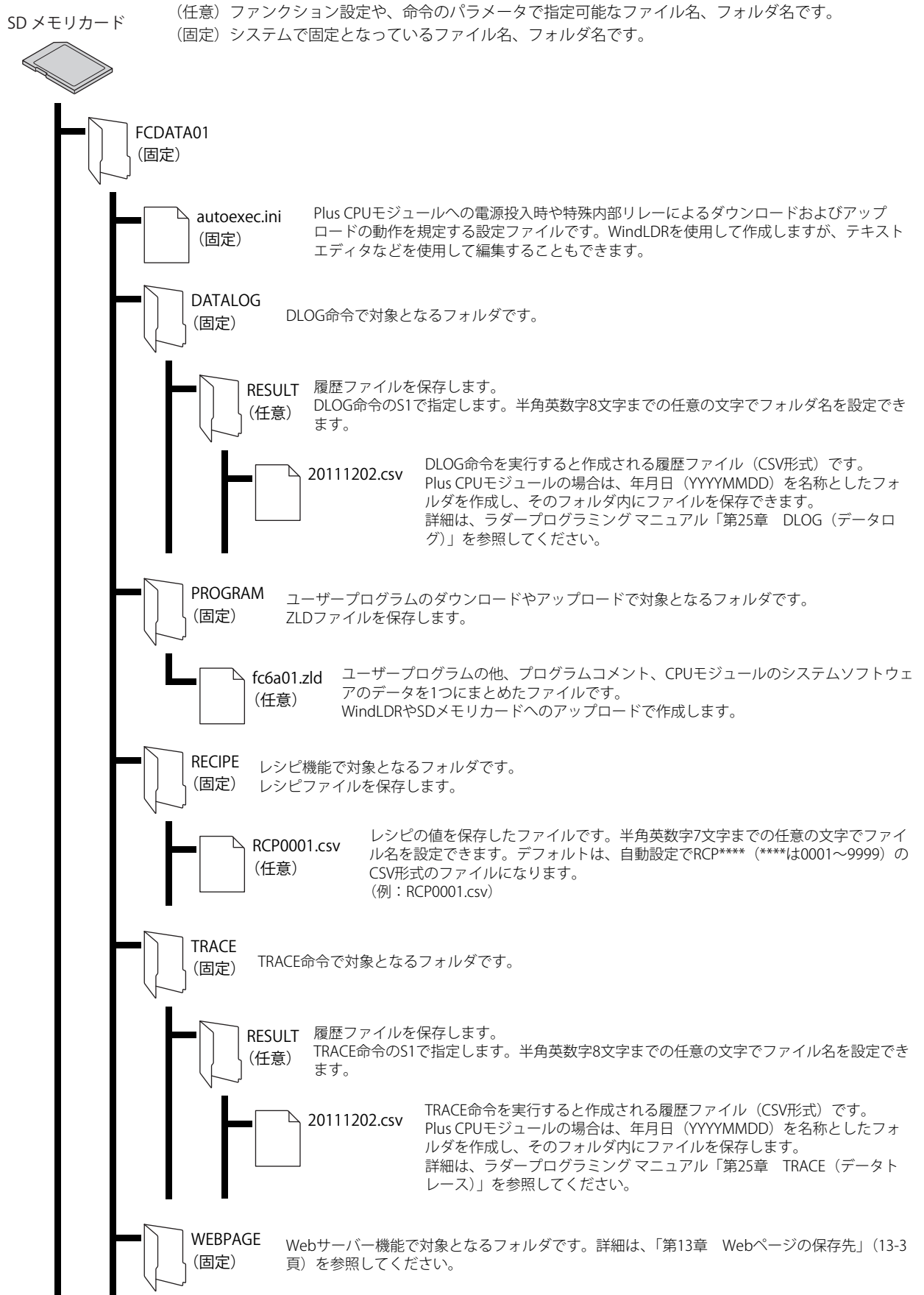
Plus CPU モジュールに接続先の FTP サーバーを登録します。

FTP クライアント機能 (ファイル転送設定) の動作条件が成立すると、次の内容を実行します。

- Plus CPU モジュールに挿入した SD メモリカードから FTP サーバーへファイルをコピーまたは移動する
- FTP サーバーから Plus CPU モジュールに挿入した SD メモリカードへファイルをコピーする

FTP クライアントで扱えるファイル

FTP クライアントでは、Plus CPU モジュールに挿入されている SD メモリカード内のすべてのファイルにアクセスできます。SD メモリカードのフォルダとファイル構成の一例を示します。



対応コマンドおよび接続方式

- RFC959 で定義されたコマンドに対応しています。
- アクティブ モードおよびパッシブ モードに対応しています。
- FTPS (FTP over SSL/TLS) に対応しています。

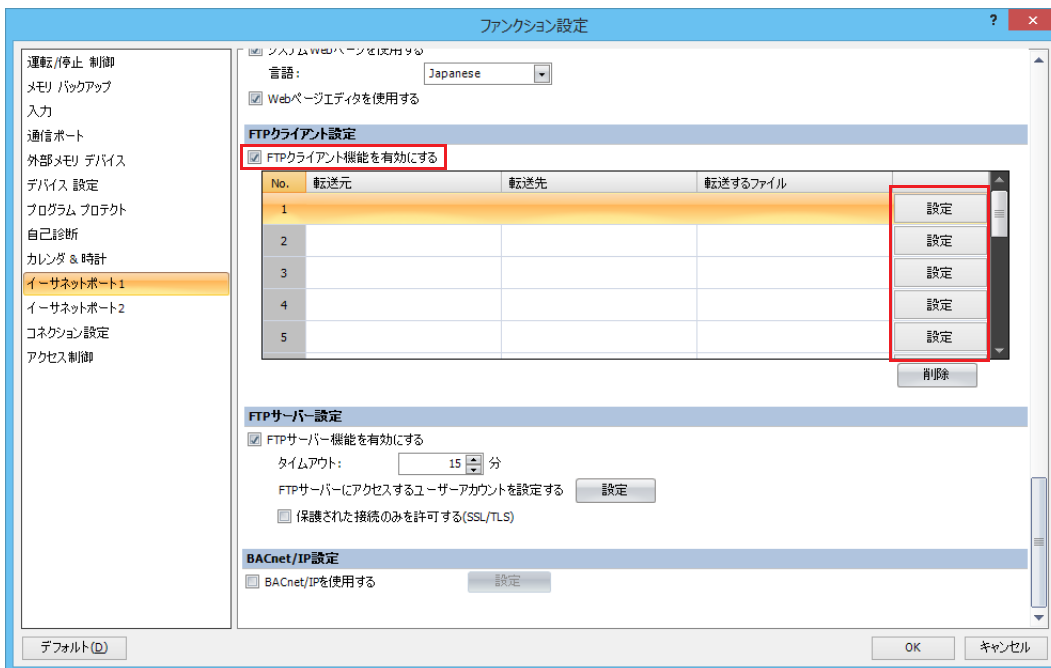


推奨 FTP サーバー ソフトウェアは、Microsoft Internet Information Services です。

WindLDR の設定

●操作手順

1. [設定] タブの [ファンクション設定] で [イーサネットポート 1] をクリックします。
2. [FTP クライアントを有効にする] チェックボックスをオンにします。
3. FTP クライアント登録一覧で設定する FTP クライアントの [設定] ボタンをクリックします。
[ファイル転送設定] ダイアログボックスが表示されます。



4. [一般] タブ、[動作条件] タブの各設定項目を設定します。



5. [OK] ボタンをクリックします。

6. 手順3～5の操作を繰り返し、必要な数のFTPクライアント設定を作成します。
7. ユーザープログラムをPlus CPUモジュールにダウンロードします。

以上で、FTPクライアントの設定が完了します。

ファイル転送設定

■ [一般] タブ

① 転送元

コピーや移動するファイルを保存している“SDメモリーカード”または“(FTPサーバー)”を選択します。
 (FTPサーバー)には、FTPサーバーマネージャーで設定したFTPサーバーの[番号] + [.(ピリオド)] + [サーバー名]が表示されます。

例) [番号]が1、[サーバー名]がTestServer1の場合
 “1.TestServer1”

[FTPサーバーマネージャー]ボタンをクリックすると、FTPサーバーマネージャーが表示されます。FTPサーバーの追加や変更ができます。詳細は、「FTPサーバーマネージャー」(10-16頁)を参照してください。

② 場所

コピーや移動するファイルの保存場所の指定方法を選択します。

フォルダーパス/ファイルパス : コピーや移動するファイルの保存場所のフォルダーパスまたはファイルパスを文字列で指定します。最大文字数は半角英数 247 文字です。

例) SDメモリーカードの「FCDATA01」フォルダーの「DATALOG」フォルダーのファイルをコピーや移動する場合
 FCDATA01¥DATALOG

データレジスタ : コピーや移動するファイルの保存場所のフォルダーパスまたはファイルパスとして使用するデータの読み出し元のワードデバイスを指定します。指定したデータレジスタを先頭として順に値を読み出し、端末文字 NULL (0x00) の前までを文字データとして扱い、フォルダーパスまたはファイルパスとします。

③ 転送するファイル

コピーや移動するファイルの指定方法を選択します。

条件を指定する : コピーや移動するファイルを条件で指定します。条件を次の中から選択します。
 “全てのファイル”、“最新ファイルのみ”、“24時間以内に更新されたファイル”、“7日以内に更新されたファイル”

ファイル名を指定する : コピーや移動するファイルを文字列で指定します。最大文字数は半角英数 247 文字です。

④転送先

コピーや移動したファイルを保存する“SDメモリーカード”または“(FTPサーバー)”を選択します。

(FTPサーバー)には、FTPサーバーマネージャーで設定したFTPサーバーの[番号] + [, (ピリオド)] + [サーバー名]が表示されます。

例) [番号]が1、[サーバー名]がTestServer1の場合

“1.TestServer1”

[FTPサーバーマネージャー] ボタンをクリックすると、FTPサーバーマネージャーが表示されます。FTPサーバーの追加や変更ができます。詳細は、「FTPサーバーマネージャー」(10-16頁)を参照してください。

⑤場所

コピーや移動したファイルの保存先の指定方法を選択します。

フォルダーパス : コピーや移動したファイルの保存先のフォルダーパスを文字列で指定します。最大文字数は半角英数247文字です。

例) コピーや移動したファイルをFTPサーバーの「20170423」フォルダーの「DATALOG」フォルダーに保存する場合
20170423\DATALOG

データレジスタ : コピーや移動したファイルの保存先のフォルダーパスとして使用するデータの読み出し元のワードデバイスを指定します。指定したデータレジスタを先頭として順に値を読み出し、終端文字 NULL (0x00) の前までを文字データとして扱い、フォルダーパスとします。

⑥操作

ファイルの転送方法を“コピー”または“移動”から選択します。[転送元]で“(FTPサーバー)”を選択した場合、“コピー”のみ設定できます。

⑦転送先に同じ名前のファイルが存在する場合

転送先に同じ名前のファイルが存在するときの処理方法を“転送しない”または“上書きする”から選択します。



転送するファイルには必ず拡張子を付けてください。拡張子がない場合は、転送されないことがあります。

⑧転送時のエラー情報をデバイスアドレスに格納する

ファイルのコピーや移動時にエラーが発生したときにエラー情報をデバイスアドレスに格納する場合は、このチェックボックスをオンにし、書き込み先のデバイスアドレスを指定します。

ファイルのコピーや移動開始時にデバイスアドレスに0を書き込み、エラーが発生すると対応するビットに1を書き込みます。指定したデバイスアドレスの各ビットには、以下のエラー情報が設定されます。

ビット位置	エラー情報	内容	対策
0	SDメモリーカードアクセスエラー	<ul style="list-style-type: none"> 転送元や転送先として設定したSDメモリーカードが挿入されていない。 転送元や転送先として設定したSDメモリーカードへアクセスできない。 	<ul style="list-style-type: none"> アクセス可能なSDメモリーカードを挿入する。 外部メモリーSDメモリーカードをマウントする。
1	SDメモリーカード読み書きエラー	<ul style="list-style-type: none"> 転送元や転送先として設定したSDメモリーカードからフォルダーやファイルの内容を読み出せない。 転送元や転送先として設定したSDメモリーカードへ、フォルダーの作成、ファイルの書き込み、ファイルの消去ができない。 	<ul style="list-style-type: none"> 読み書き可能なSDメモリーカードを挿入する。 空き容量が十分なSDメモリーカードを挿入する。
2	FTPサーバー接続エラー	接続タイムアウトで指定した時間が経過しても、転送元や転送先として設定したFTPサーバーへアクセスできない。	<ul style="list-style-type: none"> LANケーブルが正しく接続されていることを確認する。 Plus CPUモジュールのネットワーク設定が正しいかを確認する。 FTPサーバーのIPアドレスやポート番号が正しいかを確認する。
3	FTPサーバー認証エラー	ユーザー名やパスワードが間違っており、FTPサーバーへアクセスできない。	ユーザー名やパスワードが正しいかを確認する。
4	FTPサーバーコマンドエラー	FTPサーバーに送信したコマンドに対し、エラーが返ってきた。	FTPサーバーの管理者へ問い合わせる。
5	FTPサーバー転送エラー	転送タイムアウトで設定した時間が経過しても、FTPサーバーからの応答がない。	FTPサーバーの管理者へ問い合わせる。
6	リザーブ	—	—
7	その他エラー	その他の異常	FTPサーバーの管理者へ問い合わせる。
8～15	リザーブ	—	—

⑨転送が終了した時に通知する

ファイルのコピーや移動が終了したときに通知する場合は、このチェックボックスをオンにし、書き込み先のデバイスアドレスを指定します。



- FTP サーバーから FTP サーバーへのファイルのコピーや移動はできません。
- Plus CPU モジュールに挿入した SD メモリカードの間でファイルのコピーや移動はできません。
- サブフォルダーはコピーや移動できません。
- フォルダーパスまたはファイルパスに、次の半角文字は使用できません。
; : * ? " < > |
- [データレジスタ] で制限を超えた場合や使用できない文字を設定した場合のフォルダーパスは、次のとおりです。
 - フォルダーパスまたはファイルパスの文字列が最大デバイスアドレス点数を超える（NULL がない）場合は、先頭から最大デバイスアドレス点数までのデバイスアドレスに格納された文字になります。
 - 使用できない文字が設定されている場合は、その前の文字までになります。

■ [動作条件] タブ

①デバイス

ファイルをコピーまたは移動する条件をデバイスアドレスで指定する場合は、このチェックボックスをオンにします。“立ち上がりエッジ”または“立ち下がりエッジ”から選択し、条件となるデバイスを指定します。内部リレーのみ使用できません。

②周期

ファイルをコピーまたは移動する条件を曜日および時刻で指定する場合は、このチェックボックスをオンにし、実行する曜日のチェックボックスをオンにします。

③動作時刻

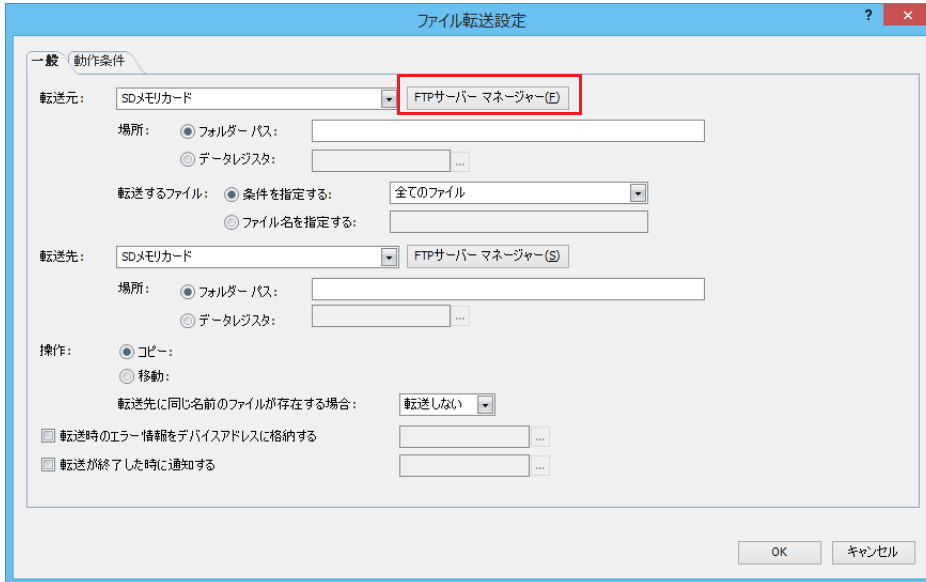
ファイルをコピーまたは移動する条件を時刻で指定する場合は、このチェックボックスをオンにし、時刻を指定します。0時00分から24時00分までの範囲で2つまで設定できます。

FTP サーバー設定

■ 接続先の FTP サーバを設定する

● 操作手順

1. [ファイル転送設定] ダイアログボックスの [一般] タブで [FTP サーバー マネージャー] ボタンをクリックします。FTP サーバー マネージャーが表示されます。



2. FTP サーバー一覧で FTP サーバーを登録する番号を選択し、[編集] ボタンをクリックします。選択した番号の [FTP サーバー No. n 設定] ダイアログボックス (n : FTP サーバー マネージャーの番号) が表示されます。



登録した FTP サーバーを削除する場合は、FTP サーバー一覧で FTP サーバーを選択し、[削除] ボタンをクリックします。

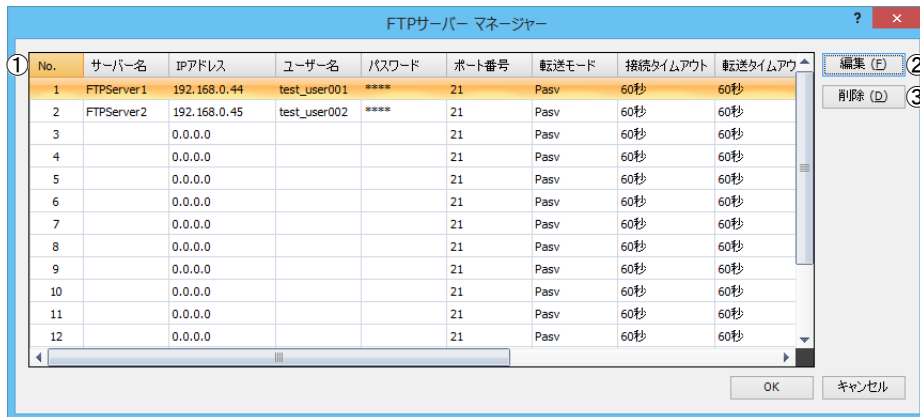
3. FTP サーバーの各項目を設定し、[OK] ボタンをクリックします。

4. 手順 2 と 3 の操作を繰り返し、必要な FTP サーバーを登録します。
5. [OK] ボタンをクリックします。

No.	サーバー名	IPアドレス	ユーザー名	パスワード	ポート番号	転送モード	接続タイムアウト	転送タイムアウト
1	FTPServer1	192.168.0.44	test_user001	****	21	Pasv	60秒	60秒
2	FTPServer2	192.168.0.45	test_user002	****	21	Pasv	60秒	60秒
3		0.0.0.0			21	Pasv	60秒	60秒
4		0.0.0.0			21	Pasv	60秒	60秒
5		0.0.0.0			21	Pasv	60秒	60秒
6		0.0.0.0			21	Pasv	60秒	60秒
7		0.0.0.0			21	Pasv	60秒	60秒
8		0.0.0.0			21	Pasv	60秒	60秒
9		0.0.0.0			21	Pasv	60秒	60秒
10		0.0.0.0			21	Pasv	60秒	60秒
11		0.0.0.0			21	Pasv	60秒	60秒
12		0.0.0.0			21	Pasv	60秒	60秒

以上で、FTP サーバー の設定が完了します。

■FTP サーバー マネージャー



① FTP サーバー一覧

各番号のFTPサーバーの設定を一覧表示します。セルをダブルクリックすると、[FTPサーバー No. n 設定] ダイアログボックス (n: FTPサーバー マネージャーの番号) が表示され、編集できます。詳細は、「FTPサーバーの個別設定」(10-17 頁) を参照してください。

- No. : FTPサーバーの設定を管理する番号 (1～16) が表示されます。
- サーバー名 : FTPサーバーの名前が表示されます。
- IPアドレス : FTPサーバーのIPアドレスが表示されます。
- ユーザー名 : FTPサーバーのユーザー名が表示されます。
- パスワード : FTPサーバーのパスワードが* (アスタリスク) で表示されます。
- ポート番号 : FTPサーバーのポート番号が表示されます。
- 転送モード : FTPサーバーの転送モードが表示されます。
- 接続タイムアウト : [ファイル転送設定] で設定した [動作条件] が成立してから、接続を試みたFTPサーバーからの応答がない場合、自動で接続処理を中止するまでの時間が表示されます。
- 転送タイムアウト : [ファイル転送設定] で設定した [動作条件] が成立してから、データコネクションを確立したあと、FTPサーバーからの応答がない場合、自動でファイルの転送処理を中止するまでの時間が表示されます。
- 先頭デバイスアドレス : FTPサーバーの設定をデバイスアドレスの値で指定するときの先頭デバイスアドレスが表示されます。
- 設定内容を初期値としてデバイスアドレスにコピーする : FTPサーバーの設定の設定内容を初期値としてコピーする条件となるビットデバイスが表示されます。

② [編集] ボタン

選択した番号の設定を登録または変更します。

FTPサーバー一覧 (①) から番号を選択して、このボタンをクリックすると、[FTPサーバー No. n 設定] ダイアログボックスが表示されます。[FTPサーバー No. n 設定] ダイアログボックスの設定を選択した番号に反映します。

詳細は、「FTPサーバーの個別設定」(10-17 頁) を参照してください。

③ [削除] ボタン

FTPサーバー一覧 (①) で選択した番号の設定を削除します。

■ FTP サーバーの個別設定

- ① **サーバー名**
FTP サーバーの名前を入力します。最大文字数は 40 文字です。
デフォルトは「FTPServern」です。(n：FTP サーバー マネージャーの番号)
- ② **IP アドレス**
FTP サーバーの識別方法選択します。

IP アドレス：FTP サーバーの IP アドレスを入力します。
入力形式は“xxx.xxx.xxx.xxx”となります。“xxx”には 0 から 255 までの値を入力します。
ホスト名：FTP サーバーのホスト名を入力します。
ホスト名の最大文字数は半角で 40 文字です。英数字および記号のみ使用できます。
- ③ **ユーザー名**
FTP サーバーのユーザー アカウントの名前を入力します。
ユーザー名の最大文字数は半角で 40 文字です。英数字および記号のみ使用できます。
- ④ **パスワード**
FTP サーバーのパスワードを入力します。
パスワードの最大文字数は半角で 40 文字です。英数字および記号のみ使用できます。
- ⑤ **ポート番号**
FTP サーバーのポート番号 (0 ~ 65535) を指定します。
- ⑥ **転送モード**
FTP サーバーの転送モードを選択します。

Pasv：パッシブモードを使用してデータ コネクションを確立します。
Active：アクティブモードを使用してデータ コネクションを確立します。
- ⑦ **接続タイムアウト**
[ファイル転送設定] で設定した [動作条件] が成立してから、接続を試みた FTP サーバーからの応答がない場合、自動で接続処理を中止するまでの時間 (10 ~ 300 秒) を指定します。
- ⑧ **転送タイムアウト**
[ファイル転送設定] で設定した [動作条件] が成立してから、データ コネクションを確立したあと、FTP サーバーからの応答がない場合、自動でファイルの転送処理を中止するまでの時間 (10 ~ 300 秒) を指定します。
- ⑨ **保護された接続を使用する (SSL/TLS)**
使用する FTP サーバーで SSL 通信または TLS 通信が必要な場合は、このチェックボックスをオンにします。
- ⑩ **FTP サーバーの設定をデバイス アドレスの値で指定する**
[FTP サーバーの設定] をデバイスアドレスで指定する場合は、このチェックボックスをオンにします。
[先頭デバイス アドレス] で使用するデバイスアドレスを指定します。指定したデバイスアドレスのアドレス番号を先頭に、[FTP サーバーの設定] の設定項目を割り当てます。

⑩設定内容を初期値としてデバイスアドレスにコピーする

[FTPサーバーの設定] の設定内容を初期値としてデバイスアドレスにコピーする場合は、このチェックボックスをオンにします。

設定内容をコピーする条件となるデバイスアドレスを指定します。

このデバイスアドレスに 1 を書き込むと、[先頭デバイスアドレス] で設定したデバイスアドレスのアドレス番号を先頭に、割り当てられたデバイスアドレスに [FTPサーバーの設定] で設定した値が書き込まれます。

FTPサーバーをデバイスアドレスの値で指定する場合のアドレスの割り付け

[FTPサーバーの設定をデバイスアドレスの値で指定する] チェックボックスをオンにした場合、[先頭デバイスアドレス] で設定したデバイスアドレスのアドレス番号を先頭に、[FTPサーバーの設定] の設定項目を割り当てます。割り当てについては、次のとおりです。

設定項目	アドレス番号	ワード数	データ形式
IPアドレスまたはホスト名	+0 ~ +20	21 ^{*1} *2	数値
ユーザー名	+21 ~ +41	21 ^{*2} *3	文字列
パスワード	+42 ~ +62	21 ^{*2} *3	文字列
ポート番号	+63	1	数値
転送モード (0 : Pasv, 1 : Active)	+64	1	数値
接続タイムアウト (秒単位)	+65	1	数値
転送タイムアウト (秒単位)	+66	1	数値
保護された接続を使用する (SSL/TLS) (0 : SSL/TLSを使用する、1 : SSL/TLSを使用しない)	+67	1	数値

*1 IPアドレスの場合は、先頭から4ワードのみ使用し、残り17ワードは予約領域となります。

*2 21ワード目はデータレジスタの値に関わらず、終端文字 NULL (0x00) として扱われます。

*3 設定する文字列が20ワードより短い場合は、文字列の終わりとして終端文字 NULL (0x00) を追加してください。

第11章 PING送信機能

この章では、設定したリモートホストに向けて PING パケットを送信し、IP レベルで通信できるかを確認する PING 送信機能について説明します。

PING 送信機能の実行には、PING 命令を使用します。

PING 命令 (Ping 送信)

Ping を送信します。Plus CPU モジュールでは、Ethernet ポート 1 または 2 を選択して使用します。

ラダー図



動作説明

入力が ON すると、S1 で設定したリモートホストへ PING パケットを送信します。応答を完了した時点で、D1 で設定したデバイスに、完了出力がセットされます。D2 で設定したデバイスに、動作のステータス（動作の遷移状態とエラー）がセットされます。D2+1 は、システムで使用します。

PING 命令は入力が ON の場合に実行され、PING 要求が送信されます。その入力が ON を保持し続けた場合、PING 送信後に同じ PING 命令が実行されます。これを回避するために、PING 命令の入力が 1 スキャンだけ ON となるよう、ショット命令をと組み合わせ使用してください。ショット命令については、ラダープログラミングマニュアル「第 4 章 SOTU (ショットアップ)」、「第 4 章 SOTD (ショットダウン)」を参照してください。

PING 命令を使用するために、FC6A 形のリモートホストリストを作成しておく必要があります。リモートホストリストの作成の詳細は、「第 3 章 リモートホストリスト」(3-21 頁)を参照してください。については

対象デバイス

			I	Q	M	R	T	C	D	P	定数	リポート指定
S1	ソース1	宛先リモートホスト番号	—	—	—	—	—	—	○	—	○*1	—
D1	ディスティネーション1	完了出力先	—	○	○*2	—	—	—	—	—	—	—
D2	ディスティネーション2	動作ステータス	—	—	—	—	—	—	○	—	—	—

*1 1 ~ 255 の範囲内です。

*2. 特殊内部リレーは使用できません。

設定項目

S1 (ソース 1) の設定

リモートホストリストの番号を設定します。定数またはデータレジスタが設定できます。

D1 (ディスティネーション 1) の設定

内部リレーまたは出力を設定します。PING 命令の実行後、応答が完了した時点、応答がない場合は、[ファンクション設定] で設定したタイムアウト時間待って ON します。タイムアウトの設定の詳細は、「第 3 章 PING 設定」(3-13 頁)を参照してください。

D2 (ディスティネーション 2) の設定

データレジスタを動作ステータスとして使用します。設定したデータレジスタを先頭に 2 ワード使用します。D2 に動作ステータスが格納されます。D2+1 は、システムで使用します。

デバイスの範囲を超えないように先頭番号を指定してください。



- S1 をデータレジスタで設定する場合、完了出力 (D1) が ON するまで S1 の内容を保持してください。
- FC6A 形は、自身から発行された PING 送信に対する PING 応答に対応していません。

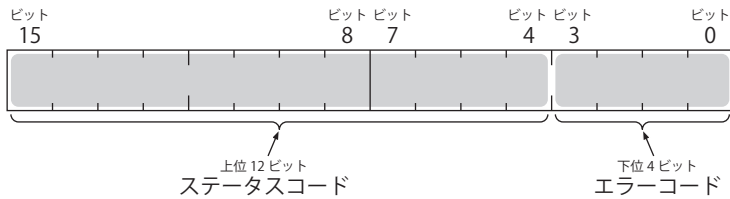
動作ステータス (D2) は、動作の遷移状態 (ステータスコード) とエラー内容 (エラーコード) を示しています。ステータスコードは、動作ステータスの下位 4 ビットをマスクした値になります。エラーコードは、動作ステータスの下位 4 ビットに格納されま

す。
動作ステータスのデータレジスタの値を 16 で割ると、その余りがエラーコードとなります。

[動作ステータスのデータレジスタの値が "66" の場合]

$$66 \div 16 = 4 \text{ 余り } 2$$

となり、エラーコードは "2" となります。



ステータスコード	動作の遷移状態	状態説明
16 (000000010000)	送信前処理中	PING命令の入力がONしてから、設定されたリモートホストへ向けてPING/パケットを送信するまでの間
32 (000000100000)	応答待ち	PING/パケットの送信処理が終わったあと、応答が返ってくるまでの間
64 (000001000000)	応答正常受信	PING/パケットに対する応答の受信が正常に完了し、次の送信が可能な状態

ステータスコードが上記以外の場合には、PING 命令のエラーと考えられます。

エラーコード	エラー内容	完了出力 (D1)
0 (0000)	正常	ステータスコードが64のとき、ONします。
1 (0001)	—	—
2 (0010)	タイムアウトエラー	ON
3 (0011)	宛先のリモートホスト名をDNS解決できなかった	ON
4 (0100)	存在しないリモートホスト番号を設定した	ON
5 (0101)	複数のPING命令を同時に実行した	ON
6 (0110)	パラメータ異常	ON

第12章 EMAIL送信機能

この章では、あらかじめ登録しておいた E メールを送信する EMAIL 送信機能について説明します。

概要

Plus CPU モジュールの Ethernet ポート 1 および HMI モジュールの HMI-Ethernet ポートは、EMAIL 送信機能に対応しています。EMAIL 命令の実行により、最大 255 種類の E メールを送信できます。

EMAIL 命令 (E メール送信)

E メールを送信します。

ラダー図



動作説明

入力が ON すると、Plus CPU モジュールの Ethernet ポート 1 および HMI モジュールの HMI-Ethernet ポートから、S1 で設定した E メール番号の E メールを送信します。応答を完了した時点で、D1 で設定したデバイスに、完了出力がセットされます。D2 で設定したデバイスに動作のステータス（動作の遷移状態とエラー）がセットされます。D2+1 は、システムで使用します。

EMAIL 命令は起動入力が ON の場合に実行され、E メールが送信されます。その起動入力が ON を保持し続けた場合、E メール送信後に同じ EMAIL 命令が実行されます。これを回避するために、EMAIL 命令の起動入力が 1 スキャンだけ ON となるよう、ショット命令と組み合わせて使用してください。ショット命令については、ラダープログラミングマニュアル「第 4 章 SOTU (ショットアップ)」、「第 4 章 SOTD (ショットダウン)」を参照してください。

EMAIL 命令を使用するには、HMI モジュールの HMI-Ethernet ポートを使用する場合は、モジュール構成エリアに挿入した HMI モジュールをクリックし、[設定] ボタンをクリックして表示される [HMI モジュール設定] ダイアログボックスで、Plus CPU モジュールの Ethernet ポート 1 を使用する場合は、[ファンクション設定] の [イーサネットポート 1] でメールサーバーなどの設定を行い、FC6A 形の E メールを作成しておく必要があります。

対象デバイス

			I	Q	M	R	T	C	D	P	定数	リピート指定
S1	ソース1	Eメール番号	—	—	—	—	—	—	○*1	—	○*2	—
D1	デスティネーション1	完了出力先	—	○	○*3	—	—	—	—	—	—	—
D2	デスティネーション2	動作ステータス	—	—	—	—	—	—	○*1	—	—	—

*1 特殊データレジスタは使用できません。

*2 1～255 の範囲内です。

*3 特殊内部リレーは使用できません。

設定項目

Ethernet ポート番号の設定

Eメールを送信する Ethernet ポートを設定します。Plus CPU モジュールの Ethernet ポート 1 または HMI モジュールの HMI-Ethernet ポートを設定します。

S1 (ソース 1) の設定

Eメールの番号を設定します。定数またはデータレジスタが設定できます。

D1 (ディスティネーション 1) の設定

内部リレーまたは出力を設定します。EMAIL 命令の実行後、Eメール送信が完了した時点または応答がない場合は一定のタイムアウト時間を待って ON します。

D2 (ディスティネーション 2) の設定

データレジスタを動作ステータスとして使用します。設定したデータレジスタを先頭に 2 ワード使用します。D2 に動作ステータスが格納されます。D2+1 は、システムで使用します。

デバイスの範囲を超えないように先頭番号を指定してください。



S1 をデータレジスタで設定する場合、完了出力 (D1) が ON するまで S1 の内容を保持してください。

動作ステータス (D2) は、動作の遷移状態 (ステータスコード) とエラー内容 (エラーコード) を示しています。ステータスコードは、動作ステータスの下位 4 ビットをマスクした値になります。エラーコードは、動作ステータスの下位 4 ビットに格納されません。

動作ステータスのデータレジスタの値を 16 で割ると、その余りがエラーコードとなります。

[動作ステータスのデータレジスタの値が "66" の場合]

$$66 \div 16 = 4 \text{ 余り } 2$$

となり、エラーコードは "2" となります。

ステータスコード	動作の遷移状態	状態説明
16 (000000010000)	送信前処理中	EMAIL命令の起動入力ONしてから、Eメールを送信するまでの間
32 (000000100000)	応答待ち	Eメールの送信処理が終わったあと、応答が返ってくるまでの間
64 (000001000000)	応答正常受信	Eメールサーバーに対する応答の受信が正常に完了し、次の送信が可能な状態

ステータスコードが上記以外の場合には、EMAIL 命令のエラーと考えられます。

エラーコード	エラー内容	完了出力 (D1)
0 (0000)	正常	ステータスコードが64のとき、ONします。
1 (0001)	HMIモジュール接続エラー ・HMIモジュールが接続されていない ・HMIモジュールが正常に動作していない	ON
2 (0010)	タイムアウトエラー ・ポート番号が間違っている	ON
3 (0011)	・宛先のリモートホスト名をDNS解決できなかった ・送信メールサーバーが見つからなかった ・送信メールサーバーが認証機能に対応していなかった ・イーサネットケーブルが抜けていたり断線したりして、FC6A形がネットワークに正しく接続できていなかった	ON
4 (0100)	存在しないEメール番号を設定した	ON
5 (0101)	複数のEMAIL命令を同時に実行した	ON
6 (0110)	パラメータ異常 ・[Eメール設定をデータレジスタで行う] が有効の場合、データレジスタに設定が格納されていない	ON
7 (0111)	その他の異常 ^{*1}	ON
8 (1000)	データレジスタの変換に失敗した ^{*2}	ON
9 (1001)	添付する履歴ファイルの1ファイルのサイズがDLOG/TRACE命令で設定した上限サイズを超えた ^{*3}	ON
10 (1010)	指定された添付する履歴ファイルが存在しない ^{*3}	ON
11 (1011)	添付する履歴ファイルの数が7を超えた ^{*4}	ON

- *1 その他の異常については特殊データレジスタ D8457 (HMI-Ethernet ポート)、または D8759 (Ethernet ポート 1) に格納されます。
- *2 Eメールの送信は実行されますが、変換に失敗したデータレジスタの値の代わりに ---- が挿入されます。
- *3 エラーコード 9 または 10 に該当する履歴ファイルは添付されず、添付ファイルなしの Eメールが送信されます。
履歴データのファイルサイズの上限設定については、FC6A 形マイクロスマート ユーザーズマニュアル「第 5 章 機能と設定」を参照してください。
- *4 7 ファイルまで添付された Eメールが送信されます。

Eメールの送信は、実際には複数の SMTP サーバーによるメールの中継によって実現されています。このため、EMAIL 命令の動作ステータス (D2) の値が 64 (応答正常受信) であっても、以下のような場合は設定した宛先でメールが受信できません。また、FC6A 形は受信できていないことを判断できません。

- ・中継を行っている SMTP サーバーが正常に機能していない場合
- ・メールの受信者が「ドメイン指定受信」などの機能を使用するなど、設定したメールアドレスやドメインからの受信しか受け付けていない場合

特殊データレジスタ

特殊データレジスタで、Eメールの詳細エラーを確認する

FC6A 形のエラーコードが 7 の場合、特殊データレジスタでメールサーバーが通知するレスポンス内容を確認できます。特殊データレジスタの割り当ては、次のとおりです。

Eメールを送信する Ethernet ポート	Eメールの詳細エラーコードが格納される特殊データレジスタ
HMIモジュールのHMI-Ethernetポート	D8457
Plus CPUモジュールのEthernetポート1	D8759

エラー内容はメールサーバーにより異なりますので、詳細は使用しているメールサーバーの管理者にご確認ください。エラー内容の例を次に示します。

エラーコード	内容	考えられる要因
451	要求されたアクションが中止された	メールサーバー側が要因となっている
452	システム容量不足	メールサーバーの容量不足
500	文法エラー、認識できないコマンド	メールサーバーが解釈できないコマンドを送信した
501	パラメータまたは引数の文法エラー	コマンドのパラメータが不正
502	コマンドが実装されていない	メールサーバーが未実装のコマンドを送信した
504	コマンドのパラメータが実装されていない	メールサーバーが未実装のコマンドパラメータを送信した
521	メールを受け取らない	スパムメールとしてみなされている
530	アクセス拒否	[Eメールを送信するために認証が必要] チェックボックスがオフになっている
535	認証エラー	アカウント名やパスワードを間違えた
550	メールボックス利用不可	ポリシー上の理由でコマンドが拒否された
552	割り当て容量を超えた	メッセージが長すぎる
554	トランザクション失敗 (または接続開始時の応答の場合はSMTPサービスがない)	メールサーバー側が要因となっている

WindLDR の設定

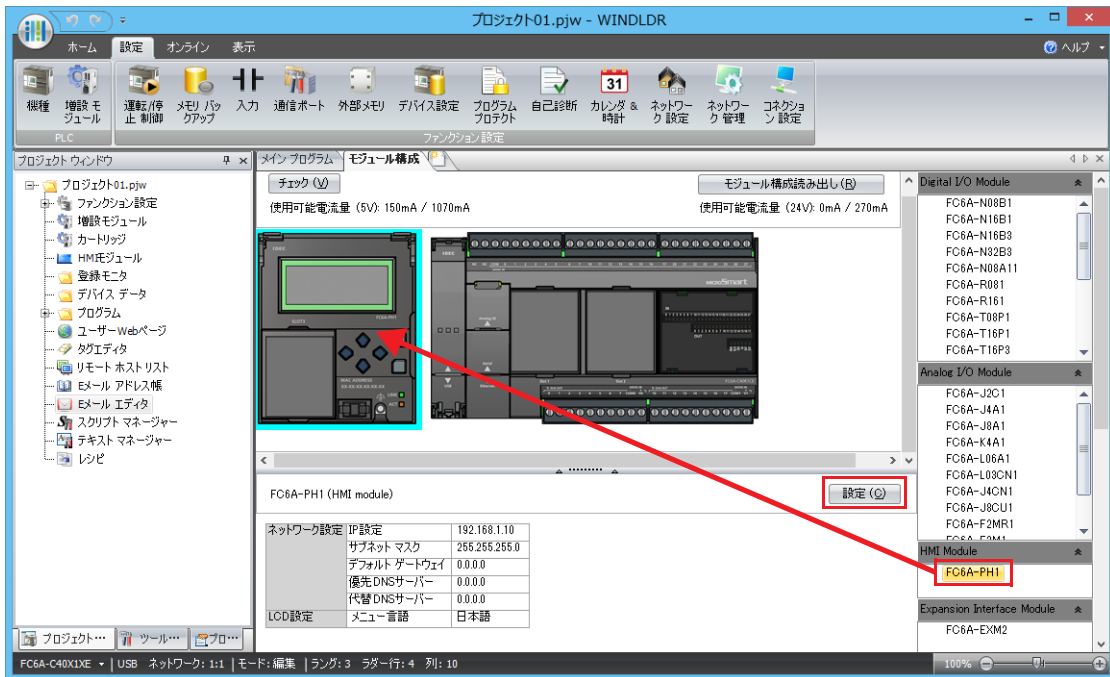
●操作手順

HMI モジュールの HMI-Ethernet ポート

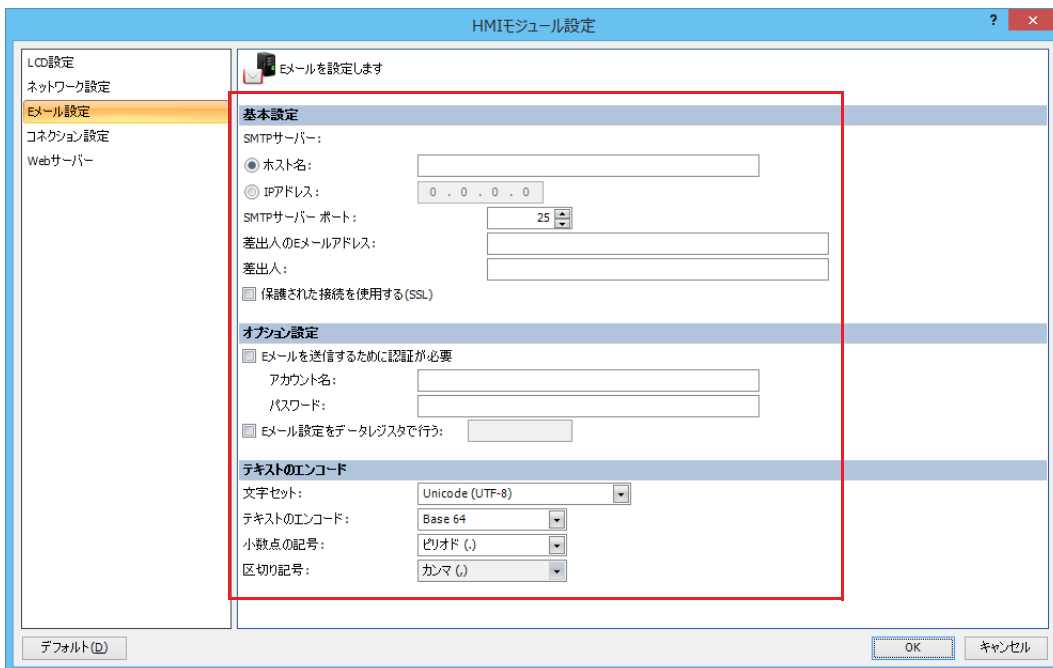
1. HMI モジュールの EMAIL 設定は、モジュール構成エディタで行います。
[設定] タブの [PLC] で [増設モジュール] をクリックします。
2. モジュール構成エリアに挿入した HMI モジュールをクリックし、[設定] ボタンをクリックします。
[HMI モジュール設定] ダイアログボックスが表示されます。



プロジェクトウィンドウの [HMI モジュール] をダブルクリックしても [HMI モジュール設定] ダイアログボックスが表示されます。



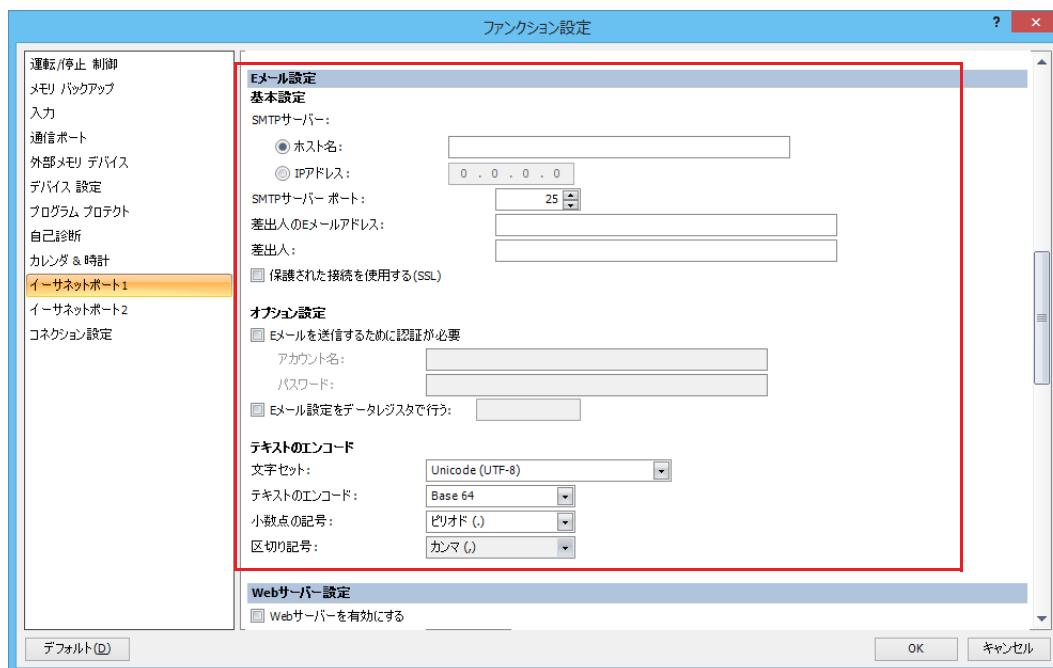
3. [Eメール設定] タブをクリックします。
4. 基本設定、オプション設定、テキストのエンコードの各設定項目を設定します。



5. ユーザープログラムを FC6A 形にダウンロードします。
以上で、Eメール設定が完了します。

Plus CPU モジュールの Ethernet ポート 1

1. [設定] タブの [ファンクション設定] で [イーサネットポート 1] をクリックします。
2. [E メール設定] で基本設定、オプション設定、テキストのエンコードの各設定項目を設定します。



3. ユーザープログラムを Plus CPU モジュールにダウンロードします。
以上で、E メール設定が完了します。

■ 基本設定

画面は [HMI モジュール設定] ダイアログボックスの例です。

The screenshot shows the 'HMIモジュール設定' dialog box with the 'Eメール設定' (Email Settings) tab selected. The settings are organized into three main sections:

- 基本設定 (Basic Settings):**
 - ① SMTPサーバー: Radio buttons for 'ホスト名' (Host Name) and 'IPアドレス' (IP Address). The IP address field shows '0.0.0.0'.
 - ② SMTPサーバーポート: A dropdown menu showing '587'.
 - ③ 差出人のEメールアドレス: A text input field.
 - ④ 差出人: A text input field.
 - ⑤ 保護された接続を使用する(SSL/TLS)
- オプション設定 (Option Settings):**
 - ⑥ Eメールを送信するために認証が必要
 - ⑦ アカウント名: A text input field.
 - ⑧ パスワード: A text input field.
 - ⑨ 認証方式: A dropdown menu set to '自動' (Automatic).
 - ⑩ Eメール設定をデータベースで行う: A checkbox with a small icon.
- テキストのエンコード (Text Encoding):**
 - ⑪ 文字セット: A dropdown menu set to 'Unicode (UTF-8)'.
 - ⑫ テキストのエンコード: A dropdown menu set to 'Base 64'.
 - ⑬ 小数点の記号: A dropdown menu set to 'ピリオド (.)'.
 - ⑭ 区切り記号: A dropdown menu set to 'カンマ (,)'.

At the bottom, there are buttons for 'デフォルト(D)' (Default), 'OK', and 'キャンセル' (Cancel).

① SMTP サーバー

Eメールの送信に使用する送信メールサーバーのIPアドレスまたは送信メールサーバー名を設定します。半角英数で40文字まで入力できます。

② SMTP サーバーポート

Eメールの送信に使用する送信メールサーバーのポート番号を設定します。通常、SMTPでは25番、SMTP-AUTHでは587番、SMTPsでは465番が使用されます。

③ 差出人のEメールアドレス

送信メールの「差出人」に含めたいメールアドレスを設定します。半角英数40文字まで入力できます。

④ 差出人

送信メールの「差出人」に含めたい名前を設定します。半角英数40文字まで入力できます。

⑤ 保護された接続を使用する (SSL/TLS)

使用する送信メールサーバーでSSL通信またはTLS通信が必要な場合は、このチェックボックスをオンにします。



- SMTPサーバーをIPアドレスで設定する場合、数値の上位桁の0は無視されます。たとえば、IPアドレス"192.168.1.234"と"192.168.001.234"は、どちらも192.168.1.234を設定したものとなります。
- SMTPサーバーをSMTPサーバー名で設定する場合は、DNSによるSMTPサーバーの名前解決が必要です。DNSおよびDNSサーバーの設定については、「第3章 ネットワークの設定」(3-3頁)を参照してください。

設定例

ここで設定した「差出人」の情報は、以下のように組み合わせられます。差出人のEメールアドレスが[test@sample.com]、差出人が[Test Mail]だった場合、Test Mail <test@sample.com> となります。

■ オプション設定：SMTP 認証

送信メールサーバーによっては、Eメールの送信を行う場合に SMTP-AUTH や SMTPs による認証が必要な場合があります。FC6A 形は、SMTP-AUTH や SMTPs による認証に対応していますので、認証が必要な送信メールサーバーであれば、この機能を使用することでEメールを送信できます。

使用する送信メールサーバーが認証を必要とするかどうかについては、該当の送信メールサーバーの管理者にお問い合わせください。

⑥ Eメールを送信するために認証が必要

使用する送信メールサーバーで認証が必要な場合は、このチェックボックスをオンにします。

⑦ アカウント名

認証で使用するアカウント名を設定します。

半角英数 40 文字まで入力できます。

⑧ パスワード

認証で使用するパスワードを設定します。

半角英数 40 文字まで入力できます。

⑨ 認証方式

認証で使用する認証方式を設定します。認証方式は、次の 5 種類から設定できます。

認証方式	内容
自動	CRAM-MD5、DIGEST-MD、LOGIN、PLAINの認証方式を順に使用して、接続に成功した認証方式で送信メールサーバーと通信します。
CRAM-MD5	CRAM-MD5を使用して、送信メールサーバーと接続します。
DIGEST-MD5	DIGEST-MD5を使用して、送信メールサーバーと接続します。
LOGIN	LOGINを使用して、送信メールサーバーと接続します。
PLAIN	PLAINを使用して、送信メールサーバーと接続します。



認証方式に“自動”を設定した場合、送信メールサーバーによっては接続できない場合があります。この場合、送信メールサーバーで使用する認証方式を設定してください。

⑩ Eメール設定をデータレジスタで行う

この項目のチェックボックスがオンの場合、基本設定および認証の設定をデータレジスタの文字列および数値を用いて行います。設定したデータレジスタを先頭のデータレジスタとする 107 ワードのデータレジスタに格納された設定を用いてメールの送信を行います。

各設定項目の開始データレジスタと終了データレジスタおよび設定値については、次のとおりです。

設定項目	データの形式	使用するデータレジスタ数	設定項目の開始データレジスタ	設定項目の終了データレジスタ	設定値		
SMTP サーバー	文字列 (40文字)	21*1	先頭のデータレジスタ	先頭のデータレジスタ+20	Eメールの送信に使用する送信メールサーバーのIPアドレスもしくは送信メールサーバー名を、文字列で設定します。*2		
SMTP サーバーポート	10進数数値	1	先頭のデータレジスタ+21		Eメールの送信に使用する送信メールサーバーのポート番号を、10進数数値で設定します。		
差出人のEメールアドレス	文字列 (40文字)	21*1	先頭のデータレジスタ+22	先頭のデータレジスタ+42	送信メールの「差出人」に含めたいメールアドレスを、文字列で設定します。*2		
差出人	文字列 (40文字)	21*1	先頭のデータレジスタ+43	先頭のデータレジスタ+63	送信メールの「差出人」に含めたい名前を、文字列で設定します。*2		
Eメールを送信するために認証が必要 / 認証方式	10進数数値	1	先頭のデータレジスタ+64		使用する送信メールサーバーにおける認証の要否および認証方式を設定します。		
					値	認証の要否	認証方式
					0	不要	—
					1	必要	自動
					3		CRAM-MD5
					5		DIGEST-MD5
7	LOGIN						
9	PLAIN						
アカウント名	文字列 (40文字)	21*1	先頭のデータレジスタ+65	先頭のデータレジスタ+85	認証で使用するアカウント名を、文字列で設定します。*2		
パスワード	文字列 (40文字)	21*1	先頭のデータレジスタ+86	先頭のデータレジスタ+106	認証で使用するパスワードを、文字列で設定します。*2		
保護された接続を使用する(SSL/TLS)	10進数数値	1	先頭のデータレジスタ+107		使用する送信メールサーバーでSSL通信またはTLS通信が必要な場合は1を、SSL通信またはTLS通信が不要な場合は0を設定します。		

*1 「使用するデータレジスタ数-1」ワード目までに設定した文字列が設定値として有効になります。文字列の終端を表すため、最終ワードは実際のデータレジスタの値に関わらず 0000h とみなします。

*2 設定する文字列が「使用するデータレジスタ数-1」よりも短い場合は、それ以降のデータは 00h で埋めてください。

設定例

以下に示す E メール設定を、データレジスタにより設定します。

送信メールサーバー : smtp.example.com
 ポート番号 : 587
 差出人の E メールアドレス : test@example.com
 差出人の名前 : Test
 E メールを送信するために認証が必要 : チェックあり
 アカウント名 : test_account
 パスワード : test_password
 認証方式 : LOGIN
 保護された接続を使用する (SSL/TLS) : チェックあり

先頭のデータレジスタに D100 を設定した場合、以下のようにデータレジスタ D100～D207 の値を設定します。

設定項目	設定項目の開始アドレス	設定項目の終了アドレス	設定値										
SMTP サーバー	D100	D120	データレジスタ	D100	D101	D102	D103	D104	D105	D106	D107	D108	D109 ～ D120
			ASCII	's' 'm'	't' 'p'	'!' 'e'	'x' 'a'	'm' 'p'	'l' 'e'	'!' 'c'	'o' 'm'	'\0' '\0'	'\0' '\0'
			値 (16進数)	736Dh	7470h	2E65h	7861h	6D70h	6C65h	2E63h	6F6Dh	0000h	0000h
SMTP サーバー ポート	D121	データレジスタ	D121										
		値 (10進数)	587										
差出人の Eメール アドレス	D122	D142	データレジスタ	D122	D123	D124	D125	D126	D127	D128	D129	D130	D131 ～ D142
			ASCII	't' 'e'	's' 't'	'@' 'e'	'x' 'a'	'm' 'p'	'l' 'e'	'!' 'c'	'o' 'm'	'\0' '\0'	'\0' '\0'
			値 (16進数)	7465h	7374h	4065h	7861h	6D70h	6C65h	2E63h	6F6Dh	0000h	0000h
差出人	D143	D163	データレジスタ	D143	D144	D145	D146～D163						
			ASCII	'T' 'e'	's' 't'	'\0' '\0'	'\0' '\0'						
			値 (16進数)	5465h	7374h	0000h	0000h						
Eメールを 送信する ために認証が 必要 / 認証 方式	D164	データレジスタ	D164										
		値 (10進数)	7										
アカウント 名	D165	D185	データレジスタ	D165	D166	D167	D168	D169	D170	D171	D172～D185		
			ASCII	't' 'e'	's' 't'	'_' 'a'	'c' 'c'	'o' 'u'	'n' 't'	'\0' '\0'	'\0' '\0'		
			値 (16進数)	7465h	7374h	5F61h	6363h	6F75h	6E74h	0000h	0000h		
パスワード	D186	D206	データレジスタ	D186	D187	D188	D189	D190	D191	D192	D193～D206		
			ASCII	't' 'e'	's' 't'	'_' 'p'	'a' 's'	's' 'w'	'o' 'r'	'd' '\0'	'\0' '\0'		
			値 (16進数)	7465h	7374h	5F70h	6173h	7377h	6F72h	6400h	0000h		
保護された 接続を使用 する (SSL/ TLS)	D207	データレジスタ	D207										
		値 (10進数)	1										



文字列は1ワード以上の連続したデータレジスタで設定します。2バイトのデータを1ワードとし、上位バイト→下位バイトの順で設定します。文字列の終わりには00hを設定してください。

特殊内部リレーを用いて送信メールサーバー設定を初期化する

[Eメール設定をデータレジスタで行う] チェックボックスがオンの場合、特殊内部リレー M8211 を用いて、[ファンクション設定] の設定値を設定したデータレジスタを先頭とするデータレジスタに初期値として設定できます。

使用する特殊内部リレー

番号	内容	説明
M8211	HMIモジュール送信メールサーバー設定の初期化	OFFからONにすると、ファンクション設定の設定値が対象のデータレジスタに初期値として設定されます。
M8344	Ethernetポート1送信メールサーバー設定の初期化	OFFからONにすると、ファンクション設定の設定値が対象のデータレジスタに初期値として設定されます。



文字列を設定する項目において、項目のワード数に対して [ファンクション設定] の設定文字列が短い場合は、それ以降のデータは 00h で埋められます。

■ テキストのエンコード

⑪ 文字セット

メール本文、添付ファイルの文字セットを設定します。

- ASCII : 本文が半角英数字のみで構成されたメールを作成する場合に設定します。
- 日本語 (ISO-2022-JP) : 本文が半角英数字と日本語文字で構成されたメールを作成する場合に設定します。添付ファイルの文字エンコード方式は Shift_JIS になります。
- 中国語 (GB2312) : 本文が半角英数字と中国語文字で構成されたメールを作成する場合に設定します。
- 西欧 (ISO-8859-1) : 本文が半角英数字と西欧文字で構成されたメールを作成する場合に設定します。
- Unicode (UTF-8) : あらゆる文字を含んだメールを作成する場合に設定します。

基本的に Unicode であれば、どのような文字を含んだメールも作成することが可能ですが、転送するメールサーバーや受信するメールソフトやフォントによっては Unicode でエンコードされたメールを転送および閲覧できない場合があります。また ASCII であれば、マルチバイト文字は使用できませんが、どのようなメールソフトで閲覧しても同じように表示されます。使用可能な文字セットについては使用するメールサーバーの管理者にお問い合わせのうえ、Eメールの宛先にあわせた文字セットを使用してください。

⑫ テキストのエンコード

本文のエンコード形式を設定します。送信メールサーバーによっては、8ビットデータを含むメールの転送に対応していない場合があります。その場合は Base64 でエンコードし7ビットのデータに変換することで転送が可能になります。

- 無し : 本文をエンコードしません。
- Base64 : 本文を Base64 でエンコードします。

⑬ 小数点の記号

Eメール本文に埋め込むデータレジスタの変換タイプを「フロート」にした場合に埋め込まれる小数文字列の小数点表記方法を設定します。

- ピリオド : ' (2Eh) を使用します。
- カンマ : ' (2Ch) を使用します。

⑭ 区切り記号

添付の CSV ファイルの区切り記号は、選択した「小数点の記号」によって決定されます。

小数点の記号	区切り記号
ピリオド' (2Eh)	カンマ' (2Ch)
カンマ' (2Ch)	セミコロン' (3Eh)



テキストのエンコードは HMI-Ethernet ポートと Ethernet ポート 1 で同じ設定になります。

例えば、[HMI モジュール設定] ダイアログボックスの [Eメール設定] でテキストのエンコードに設定した内容は、[ファンクション設定] の [イーサネットポート 1] のテキストのエンコードに自動的に反映されます。

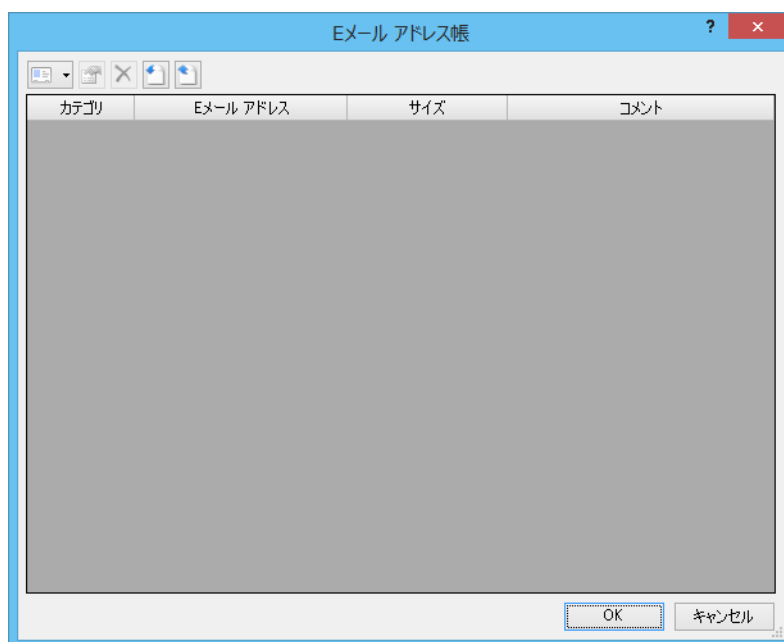
E メールアドレス帳

作成する E メールに、宛先として設定する E メールアドレスや E メールグループを作成します。
E メールグループとは、複数の E メールアドレスをグループ化したものです。E メールを作成するときに、宛先に E メールグループを設定することで、複数の E メールアドレスへ簡単にメールを送信できます。
E メールアドレスと E メールグループをあわせて最大 255 件登録できます。

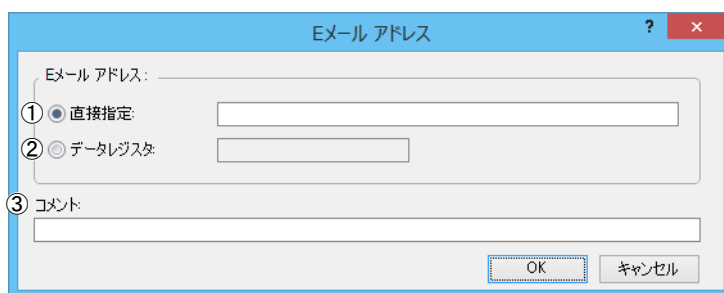
WindLDR の設定

●操作手順

1. [プロジェクトウィンドウ] の [E メールアドレス帳] をダブルクリックします。
[E メールアドレス帳作成] ダイアログボックスが表示されます。



2. [新規 E メールアドレス] の [▼] をクリックし、[新規 E メールアドレス] をクリックします。または既存の E メールアドレスを選択して [編集] をクリックします。
[E メール アドレス] ダイアログボックスが表示されます。
3. 各設定項目を設定します。



- ①直接指定
登録する E メールアドレスを設定します。半角英数 40 文字まで入力できます。
- ②データレジスタ
E メールアドレスを格納するデータレジスタを設定します。
E メール送信時に、設定したデータレジスタを先頭とする最大 30 ワードのデータレジスタに格納されている文字列を E メールアドレスとして宛先や CC で使用します。データレジスタを用いて E メールアドレスを文字列で設定する場合、E メールアドレスの長さは 60 バイト固定として計算されます。
すべての E メールアドレスをデータレジスタで設定する場合、宛先と CC、それぞれに最大 8 個の E メールアドレスを設定できます。
- ③コメント
E メールアドレスを説明するコメントです。この内容および長さは命令には影響しません。

設定例

データレジスタ D100 を設定して test@example.com に E メールを送信する場合、以下のようにデータレジスタの値を設定します。

データレジスタ	D100	D101	D102	D103	D104	D105	D106	D107	D108
ASCII	't' 'e'	's' 't'	'@' 'e'	'x' 'a'	'm' 'p'	'l' 'e'	'!' 'c'	'o' 'm'	'\0' '\0'
値 (16 進数)	7465h	7374h	4065h	7861h	6D70h	6C65h	2E63h	6F6Dh	0000h

データレジスタ D123 を設定して you@example.com に E メールを送信する場合、以下のようにデータレジスタの値を設定します。

データレジスタ	D123	D124	D125	D126	D127	D128	D129	D130
ASCII	'y' 'o'	'u' '@'	'e' 'x'	'a' 'm'	'p' 'l'	'e' '!''	'c' 'o'	'm' '\0'
値 (16 進数)	796Fh	7540h	6578h	616Dh	706Ch	652Eh	636Fh	6D00h



- 文字列は 1 ワード以上の連続したデータレジスタで設定します。2 バイトのデータを 1 ワードとし、上位バイト→下位バイトの順で設定します。文字列の終わりには 00h を設定してください。
- 文字列が 30 ワードよりも短い場合は、それ以降のデータは 00h で埋めてください。
- E メールアドレスとして設定したデータレジスタから 30 ワード (60 バイト) の中に 00h が含まれていない場合は、60 バイトすべてを E メールアドレスとして使用します。

- [OK] ボタンをクリックします。
- [新規 E メールアドレス] の [▼] をクリックし、[新規 E メールグループ] をクリックします。または既存の E メールグループを選択して [編集] をクリックします。
[E メールグループ] ダイアログボックスが表示されます。
- 各設定項目を設定します。

①グループ名

登録しておく E メールグループの名称を設定します。

②コメント

E メールグループを説明するコメントです。この内容および長さは命令には影響しません。

左側のボックスには、E メールグループに追加されていない E メールアドレスが、E メールアドレス帳に登録された順に表示されます。

左側のボックスにある E メールアドレスのうち、E メールグループに追加したい E メールアドレスを選択して [>>] をクリックすると、該当の E メールアドレスが右側のボックスの一番下の項目に移動し、E メールグループに追加されます。右側のボックスには、E メールグループに含まれている E メールアドレスが、E メールグループに追加された順に表示されます。

右側のボックスにある E メールアドレスのうち、E メールグループから削除したい E メールアドレスを選択して [<<] をクリックすると、該当の E メールアドレスが左側のボックスの一番下の項目に移動し、E メールグループから削除されます。

使用しない E メールアドレスや E メールグループを E メールアドレス帳から削除する場合は、E メールアドレス帳ダイアログを開き、削除したい E メールアドレスや E メールグループを選択した状態で [削除] をクリックします。

このとき、削除した項目以降の内容はひとつずつ上に移動します。また、E メールアドレスを削除する際に、削除した E メールアドレスが E メールグループに追加されていた場合は E メールグループからも自動的に削除されます。

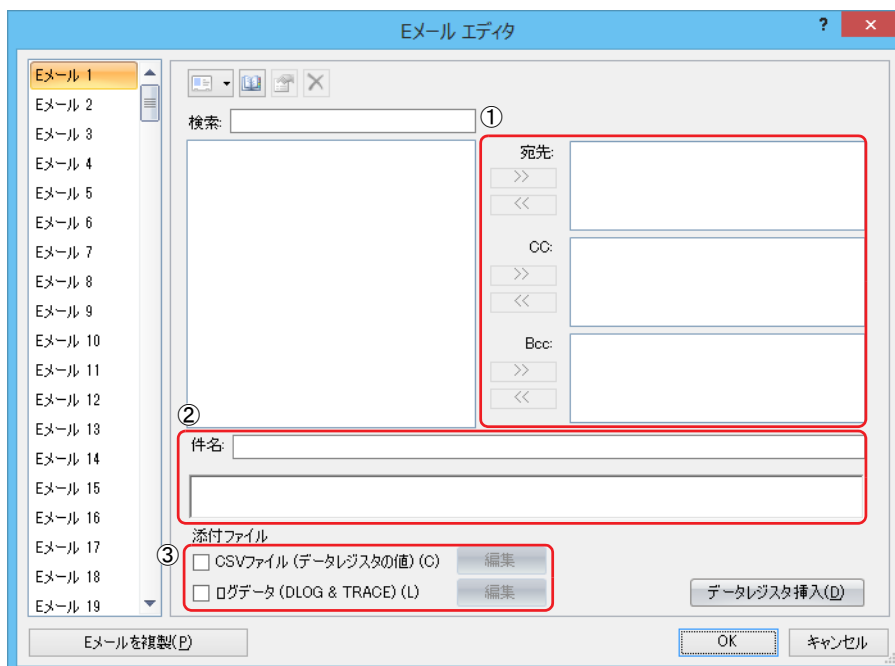
- [OK] ボタンをクリックします。

以上で、設定が完了します。

Eメールエディタ

EMAIL 命令で送信する E メールを作成します。

最大 255 種類の E メールを設定することができます。Eメールの作成は 5 つの部分に対して行います。



設定項目

①あて先 [宛先] [CC] [BCC]

[宛先]、[CC] および [BCC] には、それぞれ複数の E メールアドレスおよび E メールグループを設定することができます。それぞれの最大サイズは 512 バイトです。なお、E メールアドレスの区切りとして自動的に 1 バイト分のカンマ (",") を補います。

例："aa@example.com" (カンマ含め 15 バイト) と "bbb@example.com" (カンマ含め 16 バイト) とが含まれる E メールグループ "cccc" の合計サイズは 31 バイトになります。

目安として、30 バイトの E メールアドレスであれば、16 件まで登録可能ですので、[宛先]、[CC] および [BCC] を合計した場合は 32 件まで同時に送信することが可能です。

② [件名] [本文]

[件名] の最大サイズは 256 バイト、[本文] の最大サイズは 4096 バイトです。[件名] と [本文] は以下のものを含みます

- 文字列 ([ファンクション設定] の [Eメール設定] における [文字セット] を用いた文字列)
- 数値文字列 (データレジスタ挿入により本文に埋め込まれたデータレジスタの値の文字列)
- 空白、改行

[件名] と [本文] のサイズは、上記を含むテキストを対象とします。

テキストに対し、[ファンクション設定] の [Eメール設定] における [テキストのエンコード] にしたがってエンコード処理を行った結果を、Eメールとして送信します。

目安として、ASCII (シングルバイト文字) のみで構成する場合の上限は、件名 200 文字、本文 3500 文字です。マルチバイト文字で構成する場合の上限は、件名 100 文字、本文 2000 文字です。

③ [添付ファイル]

Eメールには、1 個の CSV ファイルを添付できます。

添付ファイルには、テキストやデータレジスタを含めることができます。CSV ファイルのファイル名は必ず「data.csv」になります。添付ファイルの最大サイズは 4096 バイトです。

[添付ファイル] は以下のものを含みます。

- [ファンクション設定] の [Eメール設定] における [文字セット] を用いた文字列
- データレジスタ挿入により本文に埋め込まれたデータレジスタの値の数値文字列
- 空白、区切り記号、改行

[添付ファイル] のサイズは、上記を含むテキストのサイズです。

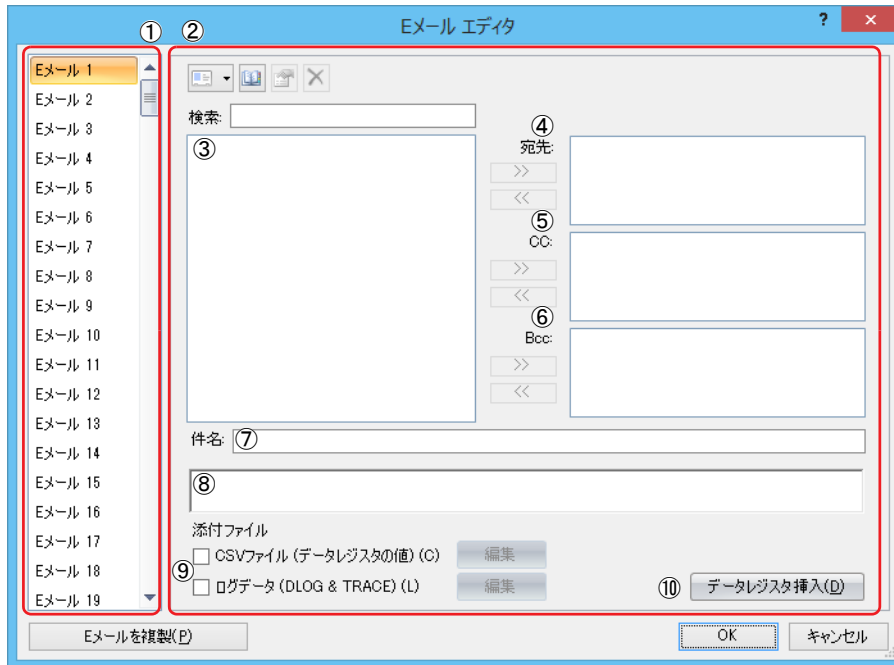
構成されたテキストに対し、[ファンクション設定] の [Eメール設定] における [テキストのエンコード] にしたがってエンコード処理を行った結果を Eメールに添付します。

また、Plus CPU モジュールの Ethernet ポート 1 から Eメールを送信する場合は、DLOG 命令や TRACE 命令で作成された履歴ファイルを添付できます。

WindLDR の設定

●操作手順

1. [プロジェクトウィンドウ] の [Eメールエディタ] をダブルクリックします。
[Eメールエディタ] ダイアログボックスが表示されます。
2. Eメールを作成します。

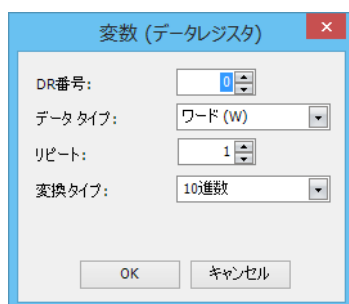


- ① Eメール
登録されている Eメールが表示されます。
- ② Eメールの内容
選択した Eメールの内容が表示されます。
[宛先] [CC] [BCC] [件名] [本文] [添付ファイル] の 6 つから構成されます。
- ③ 項目
Eメールアドレス帳に登録されている項目が表示されます。
[宛先]、[CC] または [BCC] に追加したい項目を選択して追加したい側の [>>] をクリックすると、該当の項目が [宛先]、[CC] または [BCC] に追加されます。
[宛先]、[CC] または [BCC] のボックスにある項目のうち、削除したい項目を選択して削除したい側の [<<] をクリックすると、該当の項目が [宛先]、[CC] または [BCC] から削除されます。
- ④ [宛先] / ⑤ [CC] / ⑥ [BCC]
Eメールを送信する宛先を設定します。
- ⑦ [件名]
送信する Eメールの件名を設定します。
- ⑧ [本文]
メール本文を設定します。
メール本文は、複数行で構成することが可能です。
- ⑨ 添付ファイル
Eメールにファイルを添付する場合は、チェックボックスをオンにします。

CSV ファイル (データレジスタの値) : テキストやデータレジスタの値を含めた添付ファイル (CSV ファイル) は、添付ファイルエディタで設定します。[編集] ボタンをクリックすると、添付ファイルエディタが表示されます。
ファイルサイズの上限に収まる範囲で自由にレイアウトや項目を設定することが可能です。
詳細は、「添付ファイルエディタ」(12-16 頁) を参照ください。
ログデータ (DLOG & TRACE) : Plus CPU モジュールの場合は、DLOG 命令や TRACE 命令で作成された履歴ファイルを添付できます。[履歴ファイル添付] ダイアログボックスで設定します。[編集] ボタンをクリックすると、[履歴ファイル添付] ダイアログボックスが表示されます。
詳細は、「履歴ファイル添付設定」(12-19 頁) を参照してください。

⑩データレジスタ挿入

クリックすると、[変数 (データレジスタ)] ダイアログが表示されます。



メールを発行するときに、データレジスタの値を本文中に含めることが可能です。

項目		内容
DR 番号		埋め込むデータレジスタを設定します。
データタイプ	ワード (W)	設定したデータレジスタ1ワードの値を符号なし正数に変換します。
	インテジャ (I)	設定したデータレジスタ1ワードの値を符号つき10進数整数に変換します。
	ダブルワード (D)	設定したデータレジスタ2ワードの値を符号なし正数に変換します。
	ロング (L)	設定したデータレジスタ2ワードの値を符号つき10進数整数に変換します。
	フロート (F)	設定したデータレジスタ2ワードの値を、IEEE754にしたがった浮動小数に変換します。有効桁は1桁から7桁まで設定できます。
	文字列	設定したデータレジスタの値をテキストのエンコードの文字セットで設定された文字列に変換します。文字数はワード単位で1~512ワード (終端NULL含む) まで設定できます。

このうち、ワードとダブルワードは、変換タイプを設定できます。

例として、データレジスタの値が 1234h (4660) のときの違いを以下に記載します。

10進数の場合、“4660”に置き換わります。

16進数の場合、“1234”に置き換わります。

リピート指定をすると、連続したアドレスのデータを埋め込むことが可能です。

このとき、各データレジスタの間には半角空白が一つ含まれます。

たとえば、連続した3つのデータレジスタが (D100) = 1234h, (D101) = 5678h, (D102) = ABCDh だった場合、DR 番号：100、データタイプ：ワード、リピート：3、変換タイプ：16進数で設定した場合、以下のような数値が埋め込まれます。

1234 5678 ABCD

このEメールエディタで作成された本文は、[ファンクション設定] の [Eメール設定] で設定した [エンコード方式] の設定にしたがい、エンコーディングされて送信されます。

3. [OK] ボタンをクリックします。

以上で、設定が完了します。

添付ファイルエディタ

Eメールに添付する CSV ファイルの内容を編集します。

CSV ファイル中には任意のテキスト、データレジスタの値を含むことができます。CSV ファイルのファイル名は必ず「data.csv」になります。CSV ファイルのファイルサイズは最大 4096 バイトです。

添付ファイル

行: 10 列: 5 ①

テキスト(T) データレジスタ(D) ②

	A	B	C	D	E
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					

データサイズ: 63 バイト (最大: 4096 バイト)
残りバイト数: 4033 バイト

OK キャンセル

設定項目

① [行]、[列]

CSV ファイルの行数および列数を変更することができます。行数および列数は 1～64 の範囲で設定可能です。行数および列数を小さくした場合、行数および列数の範囲外にある設定値は削除されます。

② [テキスト]、[データレジスタ]

CSV ファイルの各セルに [テキスト] か [データレジスタ] を設定することができます。セルを選択した状態で [テキスト] ボタンか [データレジスタ] ボタンをクリックすると、それぞれ編集ダイアログが表示されます。編集ダイアログで入力した設定値がセルの値として反映されます。

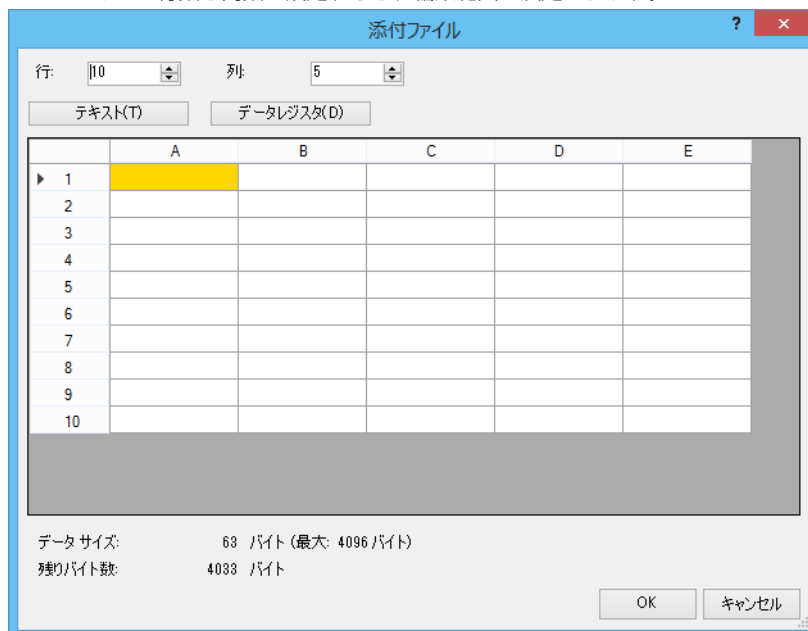


文字セット、区切り記号、小数点の記号の設定は、ファンクション設定における設定内容がすべての添付ファイルに適用されます。設定内容については、「WindLDR の設定」(12-4 頁)を参照してください。

WindLDR の設定

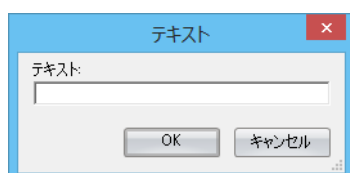
●操作手順

1. [E メールエディタ] の [添付ファイル] チェックボックスをオンにし、[データレジスタ値] の [編集] ボタンをクリックします。
[添付ファイル] ダイアログボックスが表示されます。
2. 「行」と「列」を入力します。
CSV ファイルの行数と列数を設定すると、編集範囲が決定されます。



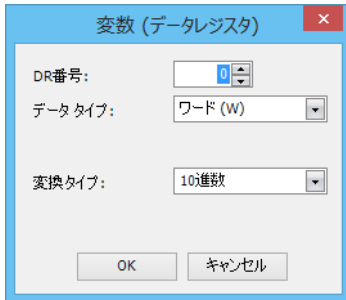
ダイアログボックスの上部で、CSV ファイルの内容の編集を行います。ダイアログボックスの下部には、現在のファイルのサイズと、編集可能なサイズが表示されます。ダイアログボックスの大きさを変化させることで、セル編集の領域を広げることが可能です。ファイルサイズには区切り記号と改行文字を含みます。

3. 任意のセルを選択し、[テキスト] をクリックします。
[テキスト] ダイアログが表示されます。
4. 選択したセルに任意のテキストを入力します。
設定できる文字列の長さは最大 63 バイトです。



5. [OK] ボタンをクリックします。
[添付ファイル] ダイアログボックスに戻ります。
6. 任意のセルを選択し、[データレジスタ] をクリックします。
[データレジスタ] ダイアログが表示されます。

7. 各設定項目を設定します。



メールを発行するときに、データレジスタの値を選択したセルに含めることができます。

項目	内容	
DR 番号	埋め込むデータレジスタを設定します。	
データタイプ	ワード (W)	設定したデータレジスタ1ワードの値を符号なし正数に変換します。
	インデジャ (I)	設定したデータレジスタ1ワードの値を符号つき10進数整数に変換します。
	ダブルワード (D)	設定したデータレジスタ2ワードの値を符号なし正数に変換します。
	ロング (L)	設定したデータレジスタ2ワードの値を符号つき10進数整数に変換します。
	フロート (F)	設定したデータレジスタ2ワードの値を、IEEE754にしたがった浮動小数に変換します。有効桁は1桁から7桁まで設定できます。

このうち、ワードとダブルワードは、変換タイプを設定できます。

例として、データレジスタの値が 1234h (4660) のときの違いを以下に記載します。

10進数の場合、“4660”に置き換わります。

16進数の場合、“1234”に置き換わります。

8. [OK] ボタンをクリックします。
 [添付ファイル] ダイアログボックスに戻ります。

9. [OK] ボタンをクリックします。

以上で、設定が完了します。

履歴ファイル添付設定

Eメールに添付する履歴ファイルを設定します。

ログデータ添付

① DLOG
フォルダ名: ② RESULT
転送するファイル: ③ 最新ファイルのみ

① TRACE
フォルダ名: ②
転送するファイル: ③ 最新ファイルのみ

OK キャンセル

設定項目

- ①履歴ファイル種別
添付する履歴ファイルの種別を選択します。
“DLOG”、“TRACE”
- ②フォルダ名
DLOG 命令または TRACE 命令で指定した履歴ファイルの保存先フォルダ名を指定します。最大 8 文字まで入力できます。
- ③転送するファイル
履歴ファイルの指定方法を次の中から選択します。
 - ・ 24 時間以内に更新されたファイル
 - ・ 7 日以内に更新されたファイル
 - ・ 最新のファイルのみ

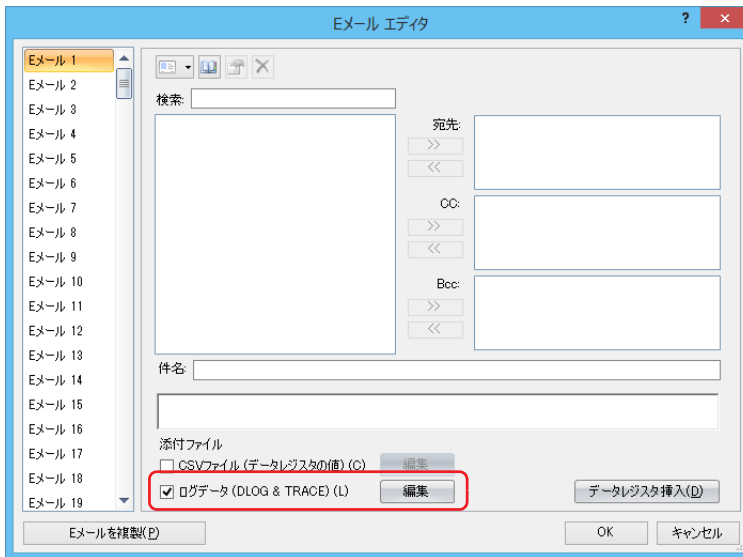


添付できる履歴ファイルの最大数は 7 ファイルです。8 ファイル目以降は、履歴ファイルは添付されず、EMAIL 命令のディスプレイステーション 2 にエラーコード 11 が格納されます。

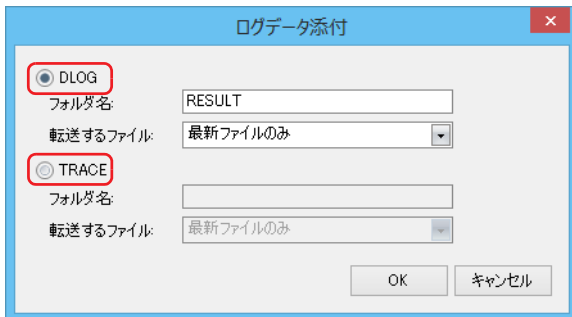
WindLDR の設定

●操作手順

1. [Eメールエディタ] の [ログデータ (DLOG & TRACE)] チェックボックスをオンにし、[編集] ボタンをクリックします。
[履歴ファイル添付] ダイアログボックスが表示されます。



2. Eメールに添付する履歴ファイルの種別を "DLOG" または "TRACE" から選択します。



3. [フォルダ名] で履歴ファイルの保存先フォルダ名を指定します。最大8文字まで入力できます。
4. [転送するファイル] で履歴ファイルの指定方法を次の中から選択します。
 - ・24時間以内に更新されたファイル
 - ・7日以内に更新されたファイル
 - ・最新ファイルのみ
5. [OK] ボタンをクリックします。

以上で、設定が完了します。

第13章 Webサーバー

この章では、FC6A形のWebサーバー機能について説明します。

概要

Plus CPU モジュールおよび HMI モジュールは、Web サーバー機能に対応しています。
Web サーバー機能で表示する Web ページは次の 3 種類です。

Web ページ	説明
システム Web ページ	Web ブラウザ上で FC6A 形の動作状態をモニタしたり、デバイス値を変更したりすることができます。
ユーザー Web ページ	ユーザーが HTML や JavaScript を使って自由に作成することができます。
カスタム Web ページ	WindLDR の Web ページエディタを使って作成することができます。

これらの Web ページを使って、プロジェクトごとに使いやすい Web サイトを構築することができます。
Web ページは、Plus CPU モジュールの内蔵メモリ、SD メモリカードまたは HMI モジュールの内蔵メモリに保存できます。詳細は、「Web ページの保存先」(13-3 頁)を参照してください。
Plus CPU モジュールの Web サーバー機能は、SSL/TLS に対応しており、第三者によるデータの盗聴や改ざん、なりすましを防ぐことができます。

対応機種および通信ポート

各 CPU モジュールの Web サーバー機能は、次のとおりです。

- : Web サーバー機能に対応しています
- × : Web サーバー機能に対応していません
- : CPU モジュールに装備されていません

CPU モジュール	ポート		
	Ethernet ポート 1	Ethernet ポート 2	HMI-Ethernet ポート
All-in-One CPU モジュール	×	—	○ *2*3
CAN J1939 All-in-One CPU モジュール			
Plus CPU モジュール	○ *1	×	

*1 SSL/TLS 通信に対応しています。

*2 CPU モジュールに HMI モジュールを接続し、HMI-Ethernet ポートを増設した場合

*3 SSL/TLS 通信に対応していません。

動作環境

FC6A 形にダウンロードした Web ページを閲覧する際に使用する推奨 OS と推奨 Web ブラウザは次のとおりです。

SSL/TLS 通信が有効の場合

OS	Web ブラウザ
Windows10	Google Chrome 90以降
	Mozilla Firefox 88以降
Android	Google Chrome 90以降
iOS	Safari 14以降

SSL/TLS 通信が無効の場合

OS	Web ブラウザ
Windows10/8/7	Google Chrome 47以降
	Mozilla Firefox 42以降
	Microsoft Internet Explorer 11
Android	Google Chrome 47以降
iOS	Safari 8以降



推奨 OS および推奨 Web ブラウザ以外でも Web サーバー機能を使用できますが、Web ページの表示や動作に問題が発生する場合があります。

基本仕様

Web ページの保存先

Plus CPU モジュールは Web ページの保存先を設定できます。All-in-One CPU モジュールは、Web ページを HMI モジュールの内蔵メモリにのみ保存できます。

- ：保存できます
×：保存できません

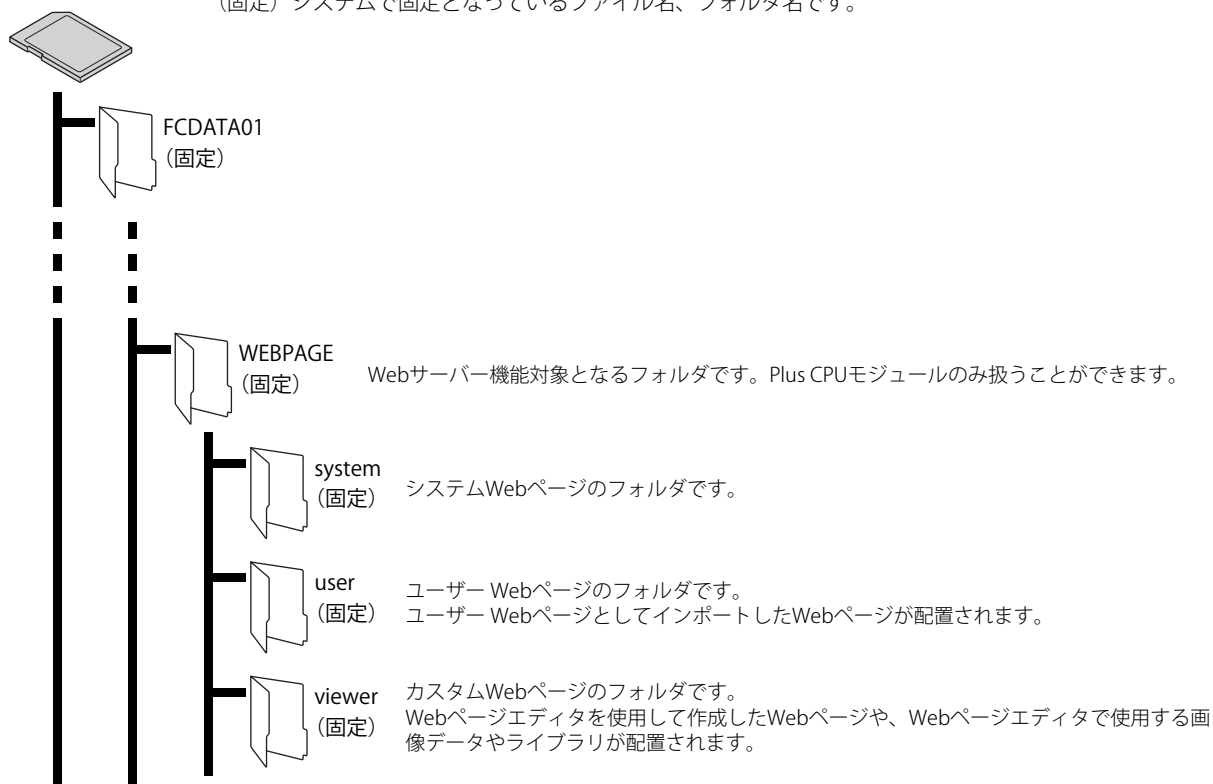
CPU モジュール	CPU モジュールの内蔵メモリ	CPU モジュールに挿入した SD メモリカード	HMI モジュールの内蔵メモリ
All-in-One CPUモジュール	×	×	○ (最大 5 メガバイト)
CAN J1939 All-in-One CPU モジュール			
Plus CPU モジュール	○ (最大 5 メガバイト)	○ (最大 32 ギガバイト *1)	

*1 1 ファイルのサイズは最大 4 ギガバイトです。

SD メモリカードへの Web ページの保存先

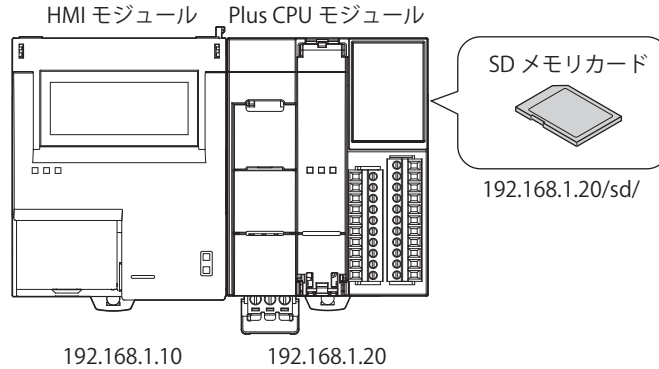
Plus CPU モジュールで Web サーバー機能を使用する場合、Web ページを SD メモリカードに保存することができます。SD メモリカードに Web ページを保存する場合の保存場所は次のとおりです。

SD メモリカード (任意) ファンクション設定や、命令のパラメータで指定可能なファイル名、フォルダ名です。
(固定) システムで固定となっているファイル名、フォルダ名です。



アクセス先と表示する Web ページ

ダウンロードした Web ページにアクセスする場合、Web ページの保存先によってアクセスする URL が異なります。たとえば、Plus CPU モジュールの Ethernet ポート 1 の IP アドレスが 192.168.1.20、Plus CPU モジュールに接続した HMI モジュールの HMI-Ethernet ポートの IP アドレスが 192.168.1.10 の場合、ダウンロードした Web ページにアクセスするときの URL は、次のとおりです。



Web ページの保存先	Web ページアクセス時の URL
Plus CPUモジュールの内蔵メモリ	http://192.168.1.20/
Plus CPUモジュールに挿入したSDメモリカード	http://192.168.1.20/sd/
HMIモジュールの内蔵メモリ	http://192.168.1.10/

また、Web ページを Plus CPU モジュールの内蔵メモリに保存し、Web ブラウザを使って Web サーバーにアクセスする場合に表示する Web ページは、次のとおりです。

アクセス先	表示する Web ページ
http://192.168.1.20/	Webサーバー設定の [リダイレクトターゲット] で設定したWebページを表示します。リダイレクトが無効な場合はエラーとなります。
http://192.168.1.20/system/	システムWebページのPLCステータス表示ページ*1を表示します。
http://192.168.1.20/user/	userフォルダにindex.html というファイルがあれば、そのindex.html を表示します。

*1 PLC ステータス表示ページについては、「PLC ステータス表示ページ」(13-27 頁) を参照してください。



- http://192.168.1.20/viewer/ にアクセスしてもカスタム Web ページにリダイレクトしません。カスタム Web ページを表示したい場合は、Web サーバー設定の [リダイレクトターゲット] でカスタム Web ページを設定し、http://192.168.1.20/ にアクセスしてください。
- Web ページにアクセスする際、ユーザー名およびパスワードの入力が要求されます。詳細は「(10) ユーザー設定」(13-9 頁) を参照してください。

コネクション数

Plus CPU モジュールの Web サーバーにおいて、SSL/TLS 通信を有効にする場合、最大 8 コネクション接続できます。



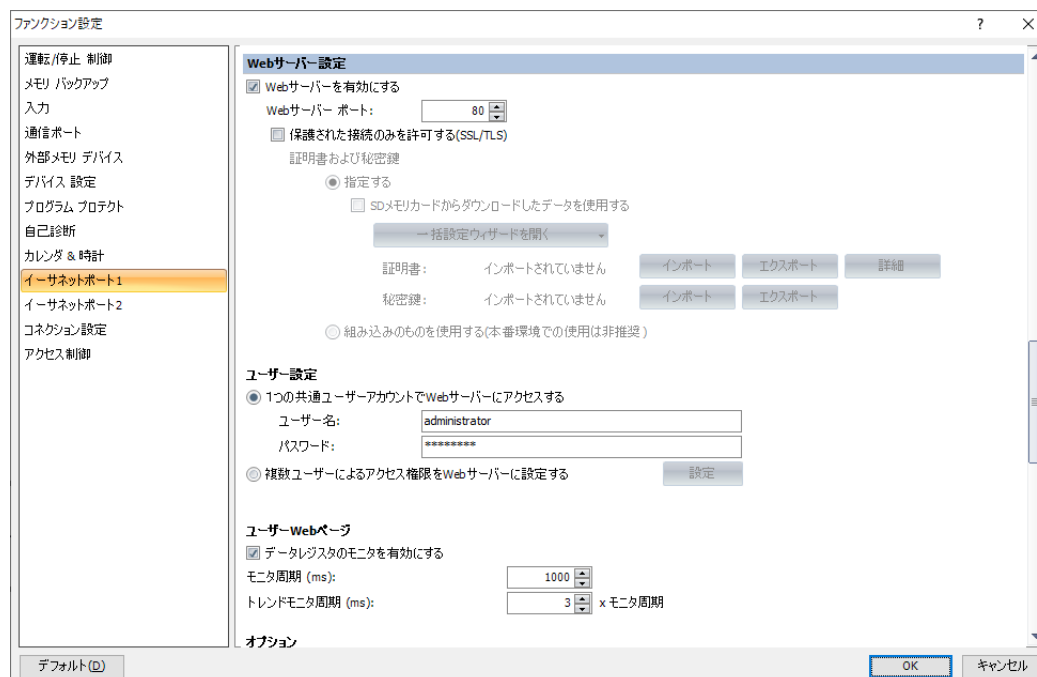
Plus CPU モジュールの Web サーバーに接続するクライアントの設定によって、クライアントが接続中および Web ページの閲覧中に使用するコネクション数は異なります。現在のコネクション数は D8822 (Web サーバーの現在のコネクション数) で確認できます。詳細は「特殊データレジスタ」(2-18 頁) を参照してください。

Plus CPU モジュールの Web サーバー

Plus CPU モジュールで Web サーバー機能を使用する場合の Web サーバーの設定について説明します。

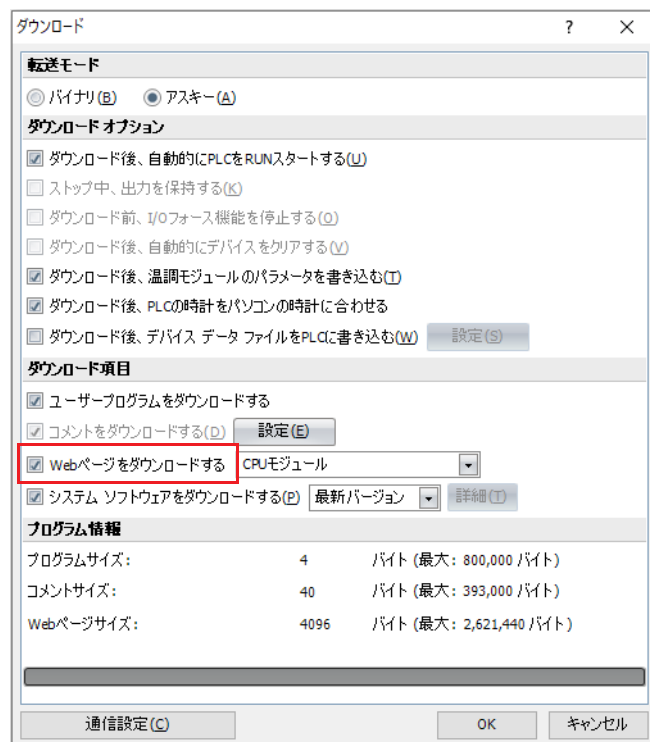
●操作手順

1. [設定] タブの [ファンクション設定] で [イーサネットポート 1] をクリックします。
2. [Web サーバー設定] で Web サーバー機能を使用するための設定を行います。



各設定項目については、「Web サーバー設定」(13-7 頁)を参照してください。

3. ユーザープログラムを Plus CPU モジュールにダウンロードします。
[オンライン] タブの [転送] で [ダウンロード] ボタンをクリックします。[ダウンロード] ダイアログボックスが表示されます。
4. [Web ページをダウンロードする] チェックボックスをオンにします。



5. Web ページの保存先を次の中から選択します。

項目	説明
ダウンロードなし	Web ページをダウンロードしません。
CPUモジュール	ダウンロードしたWeb ページをPlus CPUモジュールの内蔵メモリに保存します。
CPUモジュール (高速・非保持)	ダウンロードしたWeb ページをPlus CPUモジュールの内蔵メモリに保存します。Plus CPUモジュールの電源をOFFすると、ダウンロードしたWebページが消去されます。
CPUモジュール (SDメモリカード)	ダウンロードしたWebページをPlus CPUモジュールに挿入されたSDメモリカードに保存します。



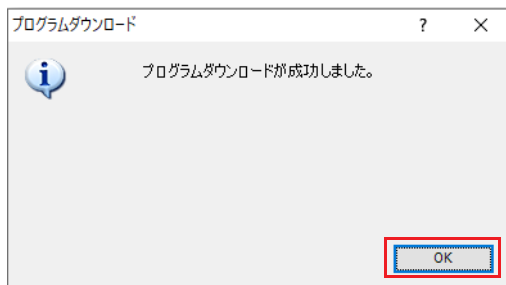
[CPU モジュール (高速・非保持)] は、[CPU モジュール] 選択時よりも Web ページを高速にダウンロードできますが、Plus CPU モジュールの電源を OFF すると、ダウンロードした Web ページが消去されます。作成した Web ページの動作確認など、Web ページの修正とダウンロードを繰り返す場合などに選択してください。



[CPU モジュール (SD メモリカード)] を選択した場合は、Plus CPU モジュールに書き込み禁止を解除した SD メモリカードを挿入した状態でダウンロードしてください。

6. [OK] ボタンをクリックします。

手順 5. で選択した保存先に、作成した Web ページがダウンロードされます。
次のメッセージが表示されたらダウンロードは成功です。[OK] ボタンをクリックします。

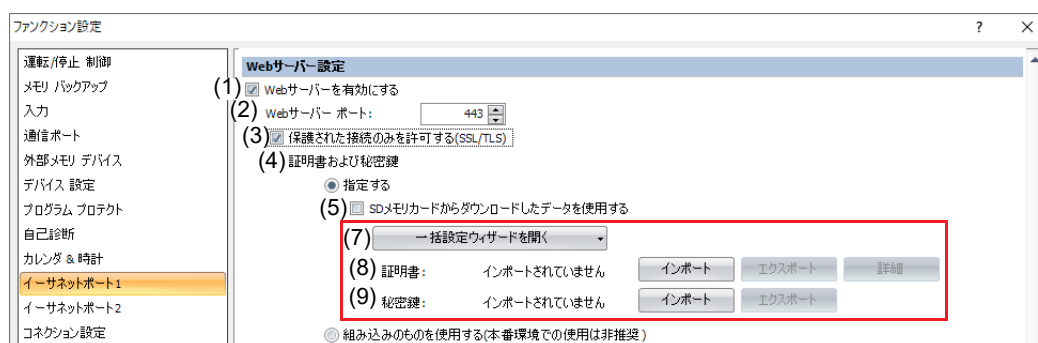


以上で、Web サーバーの設定が完了します。

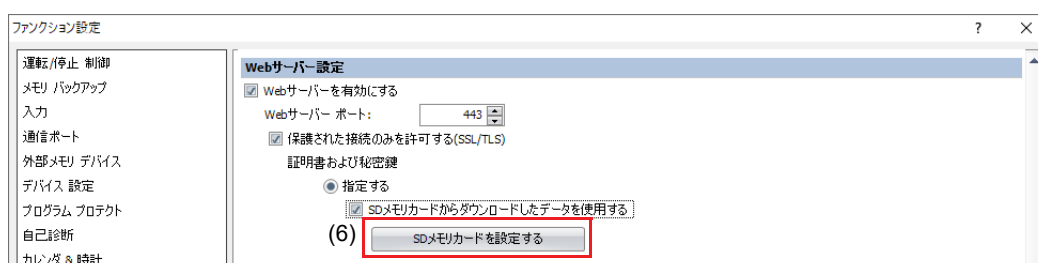
Web サーバー設定

[Web サーバー設定] で Web サーバー機能を使用するための設定を行います。
[設定] タブの [ファンクション設定] で [イーサネットポート 1] をクリックします。

[SD メモリカードからダウンロードしたデータを使用する] チェックボックスがオフの場合



[SD メモリカードからダウンロードしたデータを使用する] チェックボックスがオンの場合



- (1) Web サーバーを有効にする
Web サーバー機能を使用する場合は、このチェックボックスをオンにします。デフォルト値はオンです。
- (2) Web サーバーポート
Plus CPU モジュールの Web サーバーが開けておくポート番号を設定します。通常、HTTP では 80、HTTP over TLS では 443 が使用されます。ポート番号は 0 ～ 65535 の範囲で設定します。デフォルト値は 80 です。
- (3) 保護された接続のみを許可する (SSL/TLS)
SSL/TLS 通信を有効にする場合は、このチェックボックスをオンにします。デフォルト値はオフです。SSL/TLS 通信を有効にした場合の制限事項については「制限事項」(13-57 頁) を参照してください。
- (4) 証明書および秘密鍵
SSL/TLS 通信で使用する証明書および秘密鍵を次の中から選択します。デフォルト値は [指定する] です。

設定	内容
指定する	SSL/TLS通信で使用したい証明書と秘密鍵を自由に指定できます。
組み込みのものを使用する	Plus CPUモジュールに標準で内蔵された自己署名証明書および秘密鍵を使用します。



[組み込みのものを使用する] を選択した場合、Web ブラウザのアドレスバーに「https://～」と入力して Plus CPU モジュールの Web サーバーにアクセスしようとすると、Web ブラウザ上で警告が表示されます。この警告を無視すれば、SSL/TLS 通信を使用して Plus CPU モジュールの Web サーバーにアクセスできます。

- (5) SD メモリカードからダウンロードしたデータを使用する
SD メモリカードを使用して、証明書および秘密鍵を Plus CPU モジュールに書き込むかどうかを指定します。このチェックボックスをオンにすると、(6) が表示され、(7)～(9) が非表示になります。この場合、[SD メモリカード] ダイアログボックスで (7)～(9) を設定します。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。
- (6) SD メモリカードを設定する
[SD メモリカード] ダイアログボックスを開きます。[SD メモリカードからダウンロードしたデータを使用する] チェックボックスがオンの場合のみ表示されます。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

(7) 一括設定ウィザードを開く

[証明書および秘密鍵の一括設定] ウィザードを開き、証明書および秘密鍵をウィザード形式で設定できます。表示される手順にしたがって設定してください。



本ウィザードの中で、認証局のルート証明書を設定し、Windows の証明書ストア（論理ストア：信頼されたルート証明機関）にインポートします。さらに Plus CPU モジュールの Web サーバー機能で使用する証明書および秘密鍵を設定し、ファンクションエリア設定にインポートします。詳細は、「[証明書および秘密鍵の一括設定] ウィザード」（13-11 頁）を参照してください。

(8) 証明書

[インポート] ボタンをクリックして、SSL/TLS 通信で使用する証明書をインポートします。インポートできるのは、pem または der ファイルです。WindLDR は、pem ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルは、インポートできません。[保護された接続のみを許可する (SSL/TLS)] チェックボックスがオンで、[SD メモリカードからダウンロードしたデータを使用する] チェックボックスがオフの場合のみ設定できます。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートした証明書の情報が表示されます。

[エクスポート] ボタンをクリックすると、インポートした証明書（der ファイル）がエクスポートされます。

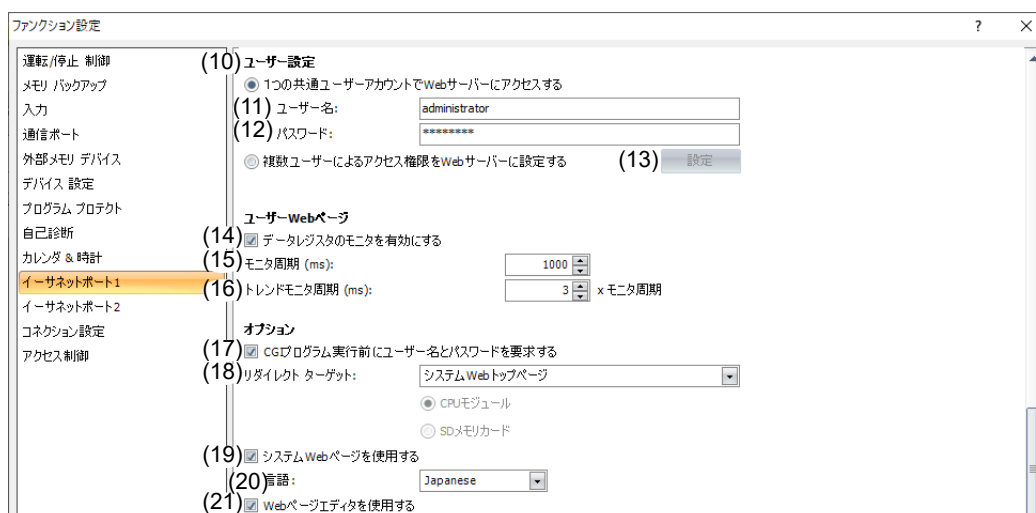
(9) 秘密鍵

[インポート] ボタンをクリックして、SSL/TLS 通信で使用する秘密鍵（暗号化方式 RSA、鍵長は 2048bit まで）をインポートします。WindLDR は次の形式のファイルに対応しています。

- PKCS#1 形式の pem または der ファイル
- PKCS#8 形式（暗号化なし）の pem または der ファイル

[保護された接続のみを許可する (SSL/TLS)] チェックボックスがオンで、[SD メモリカードからダウンロードしたデータを使用する] チェックボックスがオフの場合のみ設定できます。

[エクスポート] ボタンをクリックすると、インポートした秘密鍵（der ファイル）がエクスポートされます。



(10) ユーザー設定

作成するアカウントの数（1つまたは複数個）を次の中から選択します。デフォルト値は「1つの共通ユーザーアカウントでWebサーバーにアクセスする」です。

ユーザー設定	内容
1つの共通ユーザーアカウントでWebサーバーにアクセスする	1つの共通ユーザーアカウントでWebサーバーにアクセスする場合に選択します。
複数ユーザーによるアクセス権限をWebサーバーに設定する	複数のユーザーアカウントでWebサーバーへのアクセスする場合に選択します。



複数のユーザーアカウントは、「ユーザーアカウント設定」ダイアログボックスで設定します。詳細は、「ユーザーアカウント設定」(13-24 頁)を参照してください。

(11) ユーザー名

(12) パスワード

ユーザー名およびパスワードを設定します。ユーザー名の最大文字数は 40 文字、パスワードの最大文字数は 16 文字です。英数字および記号のみ使用できます。[1つの共通ユーザーアカウントでWebサーバーにアクセスする] ラジオボタンがオンの場合のみ設定できます。

項目	出荷時の設定
ユーザー名	administrator
パスワード	password

(13) [設定] ボタン

[ユーザーアカウント設定] ダイアログボックスを開きます。[ユーザーアカウント設定] ダイアログボックスでユーザーアカウントを作成してください。詳細は、「ユーザーアカウント設定」(13-24 頁)を参照してください。

(14) データレジスタのモニタを有効にする

次の場合に、このチェックボックスをオンにします。

- HTML ページにメタ文字を埋め込みデータレジスタのモニタを使用する場合
- グラフライブラリを使用する場合

拡張子が html および htm のファイルのみ有効です。

(15) モニタ周期 (ms)

モニタを行う場合、Web ブラウザから FC6A 形ヘリクエストを送る周期 (ms) を設定します。

(16) トレンドモニタ周期 (ms)

トレンドグラフを使用したモニタを行う場合、Web ブラウザから FC6A 形ヘリクエストを送る周期 (ms) をモニタ周期の整数倍で設定します。

(17) CGI プログラム実行前にユーザー名とパスワードを要求する

CGI プログラム実行前にユーザー名とパスワードを要求する場合は、このチェックボックスをオンにします。デフォルト値はオンです。CGI プログラムについては「CGI」(13-55 頁)を参照してください。

(18) リダイレクトターゲット

[リダイレクトターゲット] で設定した Web ページにリダイレクトさせることができます。

設定	説明	リダイレクト先の設定
リダイレクトを無効にする	リダイレクトしません。	不要
システムWebページ	システムWebページのトップページにリダイレクトします。	
ユーザー WebページおよびカスタムWebページ*1	設定したユーザー WebページまたはカスタムWebページにリダイレクトします。	必要

*1 インポートした、または保存したフォルダ名およびファイル名を選択します。

リダイレクト先に指定した Web ページの保存先を、CPU モジュールまたは SD メモリーカードから選択できます。

リダイレクト先	説明
CPUモジュール	CPUモジュールの内蔵メモリに保存されたリダイレクトターゲットにリダイレクトします。
SDメモリーカード	SDメモリーカードに保存されたリダイレクトターゲットにリダイレクトします。

たとえば、Plus CPU モジュールの Ethernet ポート 1 の IP アドレスが 192.168.1.20 で、リダイレクト先に SD メモリーカードを設定して Web ページを SD メモリーカードにダウンロードした場合、次の URL でリダイレクトターゲットを表示することができます。

• <http://192.168.1.20/sd/>



リダイレクト先と Web ページのダウンロード先は同じになるように設定してください。

(19) システム Web ページを使用する

システム Web ページを使用する場合は、このチェックボックスをオンにします。

(20) 言語

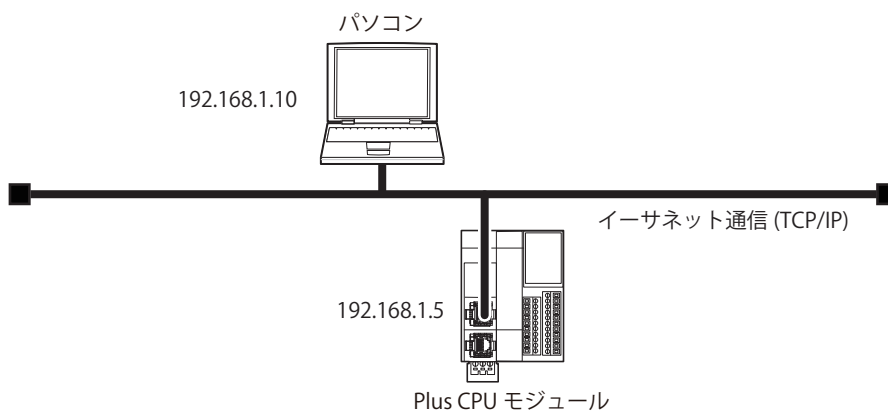
システム Web ページの言語を日本語、英語、中国語、ドイツ語またはスペイン語から設定します。

(21) Web ページエディタを使用する

Web ページエディタを使用する場合は、このチェックボックスをオンにします。詳細は、「Web ページエディタ」(13-43 頁)を参照してください。

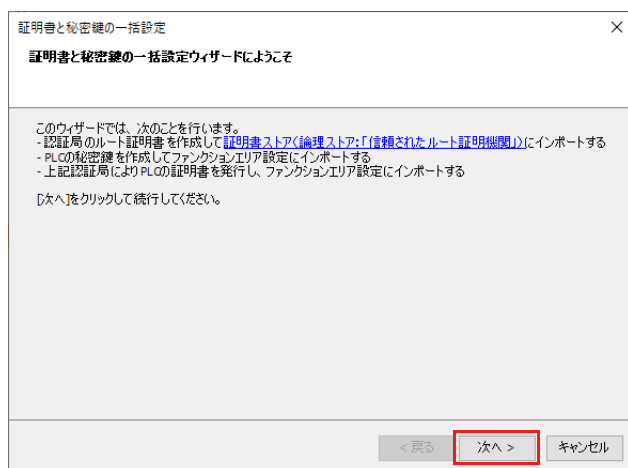
[証明書および秘密鍵の一括設定] ウィザード

Plus CPU モジュールでは、SSL/TLS 通信で使用する証明書および秘密鍵を [証明書および秘密鍵の一括設定] ウィザードで設定できます。下図のようなシステム構成で Web サーバー機能を使用する場合に、証明書および秘密鍵を設定する手順について説明します。



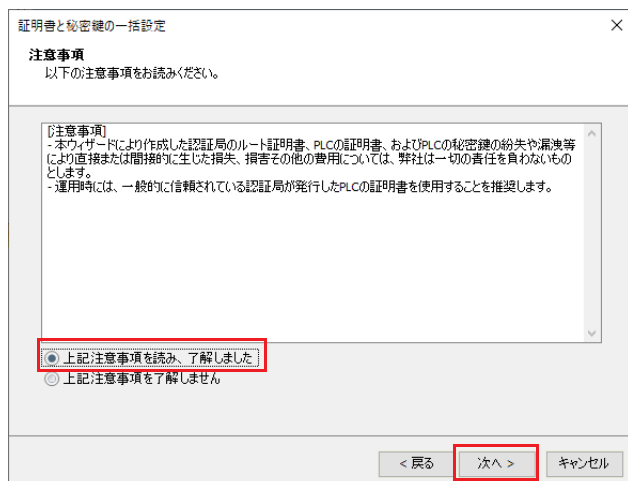
● 操作手順

1. [一括設定ウィザードを開く] をクリックします。
[証明書および秘密鍵の一括設定] ウィザードが表示されます。
2. [次へ] をクリックします。



[証明書ストア (論理ストア: 「信頼されたルート証明機関」)] リンクをクリックすると、Windows に現在ログオンしているユーザーの証明書ストア (certmgr.msc) が開きます。

3. [上記注意事項を読み、了解しました] を選択して [次へ] をクリックします。



4. 設定したい項目を設定し、[次へ] をクリックします。特に設定したい項目がない場合は、各設定項目を設定せずに、[次へ] をクリックします。

各設定項目の説明は次のとおりです。

(1) 以下の設定で認証局のルート証明書を作成します

認証局のルート証明書を新規作成する場合オンします。デフォルト値はオンです。

(2) コモンネーム

認証局のルート証明書のコモンネームを設定します。最大文字数は 128 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?'

[以下の設定で認証局のルート証明書を作成します] ラジオボタンがオンの場合のみ設定できます。デフォルト値は MyRootCA です。

(3) 国

認証局の所在地（国）を設定します。最大文字数は 2 文字です。半角英字（大文字）のみ使用できます。[以下の設定で認証局のルート証明書を作成します] ラジオボタンがオンで、[国] チェックボックスがオンの場合のみ設定できます。

(4) 都道府県

認証局の所在地（都道府県）を設定します。最大文字数は 128 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?'

[以下の設定で認証局のルート証明書を作成します] ラジオボタンがオンで、[都道府県] チェックボックスがオンの場合のみ設定できます。

(5) 市町村

認証局の所在地（市町村）を設定します。最大文字数は 128 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?'

[以下の設定で認証局のルート証明書を作成します] ラジオボタンがオンで、[市町村] チェックボックスがオンの場合のみ設定できます。

(6) 組織

認証局の組織名を設定します。最大文字数は 64 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?'

[以下の設定で認証局のルート証明書を作成します] ラジオボタンがオンで、[組織] チェックボックスがオンの場合のみ設定できます。

(7) 部門

認証局の部門名を設定します。最大文字数は 64 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?'

[以下の設定で認証局のルート証明書を作成します] ラジオボタンがオンで、[部門] チェックボックスがオンの場合のみ設定できます。

(8) 有効期間の開始

認証局のルート証明書の有効期間（開始日）を設定します。デフォルト値はパソコンの日付です。[以下の設定で認証局のルート証明書を作成します] ラジオボタンがオンの場合のみ設定できます。

(9) 有効期間の終了

認証局のルート証明書の有効期間（終了日）を設定します。デフォルト値はパソコンの日付から 365 日後です。[以下の設定で認証局のルート証明書を作成します] ラジオボタンがオンの場合のみ設定できます。

(10) 前回作成した認証局のルート証明書を再利用します

前回作成した認証局のルート証明書を再利用する場合、オンします。前回作成した認証局のルート証明書が存在する場合のみ設定できます。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、前回作成した認証局のルート証明書の情報が表示されます。

5. [2048 ビット] または [1024 ビット] を選択して [次へ] をクリックします。

6. [コモンネーム] および [サブジェクト代替名] に Plus CPU モジュールの Ethernet ポート 1 の IP アドレスを設定して [次へ] をクリックします。その他設定したい項目がある場合は、設定したい項目を設定し、[次へ] をクリックします。

各設定項目の説明は次のとおりです。

(1) コモンネーム

PLC のドメイン名または IP アドレスを設定します。ドメイン名の場合、最大文字数は 128 文字です。半角英数字および、- (ハイフン) および . (ピリオド) のみ使用できます。IP アドレスの場合、入力形式は "xxx.xxx.xxx.xxx" です。"xxx" には 0 から 255 までの値を入力します。



「*」から始まるドメイン名を指定して、ワイルドカード証明書を作成することができます。

(2) 国

認証局が認証する対象の所在地（国）を設定します。最大文字数は 2 文字です。半角英字（大文字）のみ使用できます。[国] チェックボックスがオンの場合のみ設定できます。

(3) 都道府県

認証局が認証する対象の所在地（都道府県）を設定します。最大文字数は 128 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?

[都道府県] チェックボックスがオンの場合のみ設定できます。

(4) 市町村

認証局が認証する対象の所在地（市町村）を設定します。最大文字数は 128 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?

[市町村] チェックボックスがオンの場合のみ設定できます。

(5) 組織

認証局が認証する対象の組織名を設定します。最大文字数は 64 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?

[組織] チェックボックスがオンの場合のみ設定できます。

(6) 部門

認証局が認証する対象の部門名を設定します。最大文字数は 64 文字です。半角英数字、次の特殊文字、およびスペース文字のみ使用できます。

'()-./:?

[組織] チェックボックスがオンの場合のみ設定できます。

(7) サブジェクト代替名

PLC のドメイン名または IP アドレスを設定します。ドメイン名の場合、最大文字数は 128 文字です。半角英数字および、-（ハイフン）および .（ピリオド）のみ使用できます。IP アドレスの場合、入力形式は "xxx.xxx.xxx.xxx" です。"xxx" には 0 から 255 までの値を入力します。設定できるドメイン名または IP アドレスの数は最大 10 個です。



「*」から始まるドメイン名を指定して、ワイルドカード証明書を作成することができます。

(8) 有効期間の開始

PLC の証明書の有効期間（開始日）を設定します。デフォルト値はパソコンの日付です。認証局のルート証明書の有効期間（開始日）より前の日付は設定できません。

(9) 有効期間の終了

PLC の証明書の有効期間（終了日）を設定します。デフォルト値はパソコンの日付から 60 日後です。認証局のルート証明書の有効期間（終了日）より後の日付は設定できません。

7. 設定内容を確認して [次へ] をクリックします。

証明書と秘密鍵の一括設定 ×

設定内容の確認

以下の内容でよろしければ「次へ」をクリックしてください。
セキュリティ警告が表示された場合は「はい」をクリックしてください。

[認証局のルート証明書]

発行者/サブジェクト: CN=MyRootCA

有効期間の開始: 2021/05/18

有効期間の終了: 2022/05/18

[PLCの秘密鍵]

鍵長: 2048ビット

[PLCの証明書]

発行者: CN=MyRootCA

サブジェクト: CN=192.168.1.5

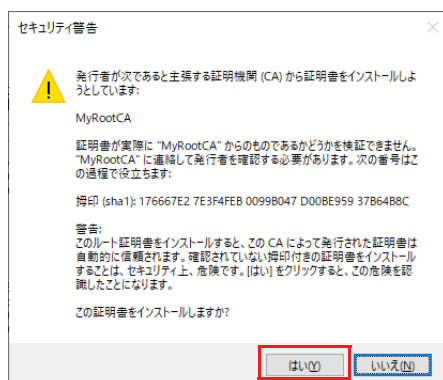
サブジェクト代替名: IP Address=192.168.1.5

有効期間の開始: 2021/05/18

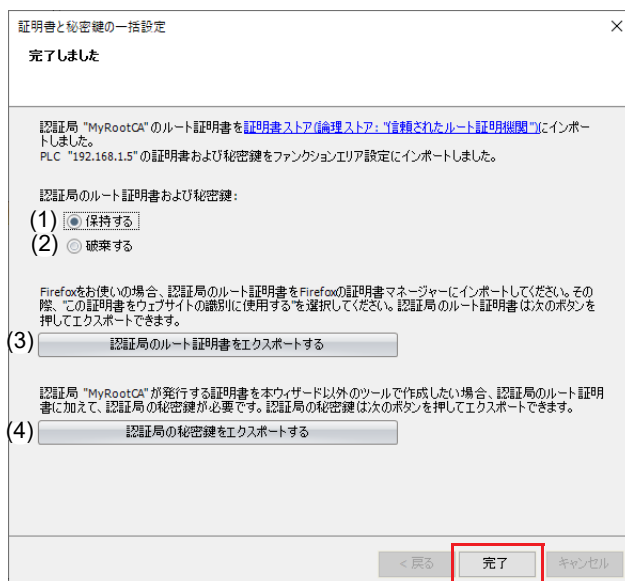
有効期間の終了: 2021/07/17

< 戻る
次へ >
キャンセル

8. [はい] をクリックします。



9. [完了] をクリックします。

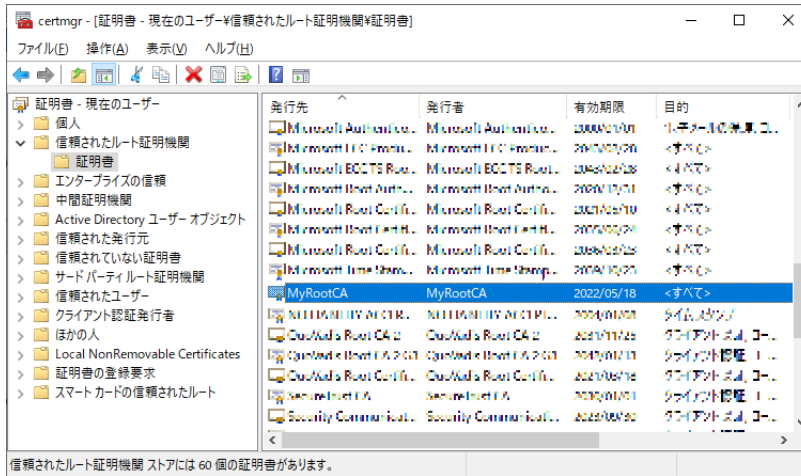


設定項目およびボタンの説明は次のとおりです。

- (1) 保持する
認証局のルート証明書および秘密鍵をプロジェクトに保存する場合、オンします。デフォルト値はオンです。
- (2) 破棄する
認証局のルート証明書および秘密鍵をプロジェクトから削除する場合、オンします。
- (3) [認証局のルート証明書をエクスポートする] ボタン
認証局のルート証明書 (der ファイル) をエクスポートします。
- (4) [認証局の秘密鍵をエクスポートする] ボタン
認証局の秘密鍵 (der ファイル) をエクスポートします。



- [証明書ストア (論理ストア:「信頼されたルート認証機関」)] リンクをクリックすると、Windows に現在ログオンしているユーザーの証明書ストア (certmgr.msc) が開きます。左側のウィンドウで「信頼されたルート認証機関」を展開し「証明書」をクリックすると、MyRootCA (認証局のルート証明書) が追加されたことが確認できます。

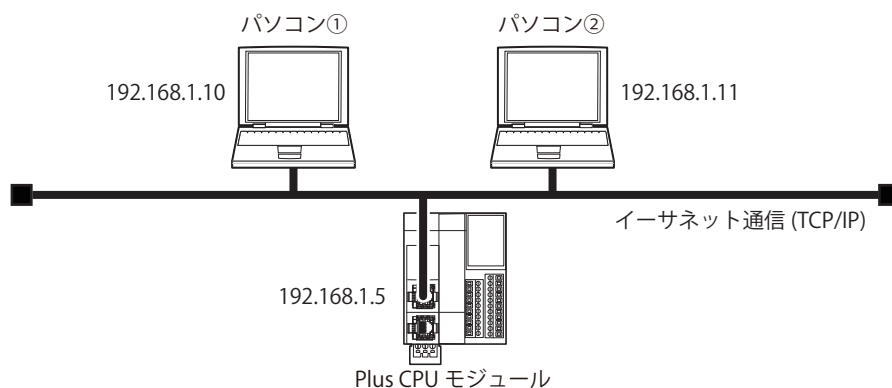


- 認証局のルート証明書が証明書ストアにインポートされていないパソコンから Web ブラウザを使って Web サーバーにアクセスする場合、Web ブラウザ上で警告が表示されます。[認証局のルート証明書をエクスポートする] ボタンをクリックして、認証局のルート証明書をエクスポートし、パソコンにインポートすることで、この警告を消すことができます。詳細は、「認証局のルート証明書のインポート」(13-17 頁) を参照してください。

以上で、証明書および秘密鍵の設定が完了します。

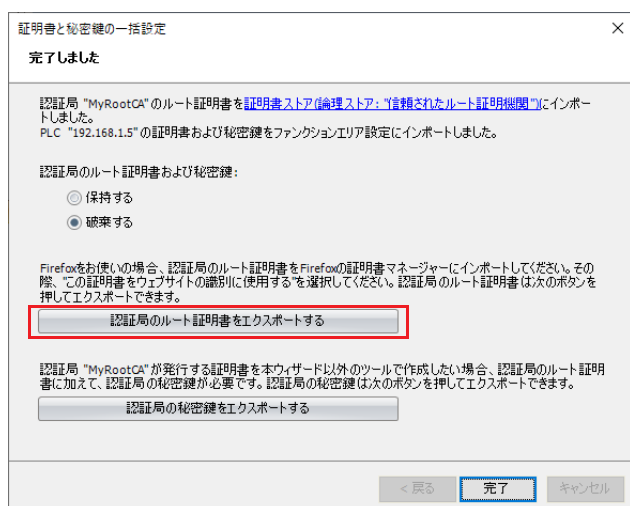
認証局のルート証明書のインポート

認証局のルート証明書が証明書ストアにインポートされていないパソコンから Web ブラウザを使って Web サーバーにアクセスする場合、Web ブラウザ上で警告が表示されます。認証局のルート証明書を Windows の証明書ストアにインポートすればこの警告を消すことができます。下図のパソコン①で認証局のルート証明書を設定し、その証明書をパソコン②の証明書ストアにインポートする場合の手順は次のとおりです。



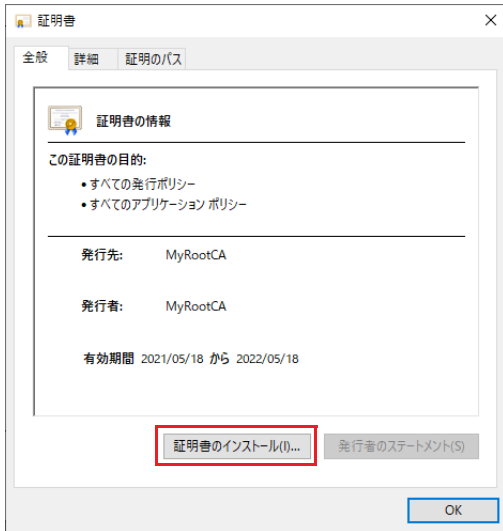
●操作手順

1. パソコン①で「[証明書および秘密鍵の一括設定] ウィザード」(13-11 頁) の手順 1.～手順 8. までを実施します。
2. [認証局のルート証明書をエクスポートする] ボタンをクリックして、認証局のルート証明書をパソコン①に保存します。



3. 手順 1. で保存した認証局のルート証明書をパソコン②にコピーします。

4. パソコン②のエクスプローラー上で、認証局のルート証明書をダブルクリックしてプロパティを開き、[証明書のインストール] をクリックします。



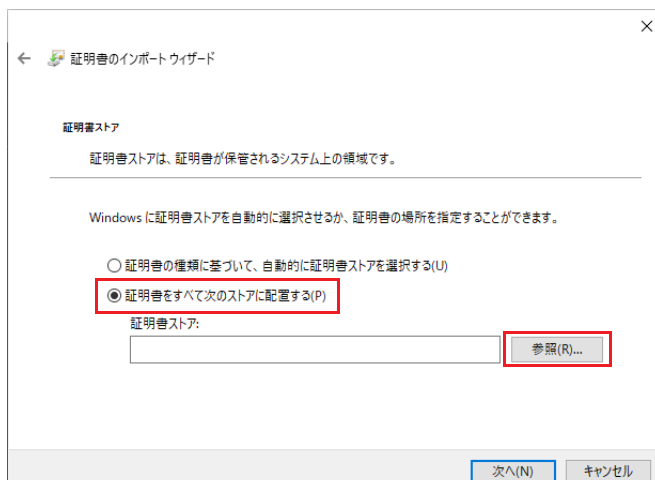
証明書のインポートウィザードが開きます。

5. [次へ] をクリックします。

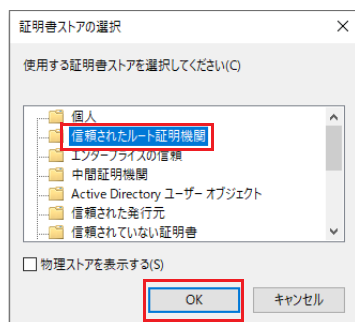


パソコン②のすべてのユーザーで認証局のルート証明書を参照できるようにするには、[ローカルコンピューター] をオンします。ローカルコンピューターの証明書ストア (certlm.msc) に認証局のルート証明書がインポートされます。(管理者権限が必要です)

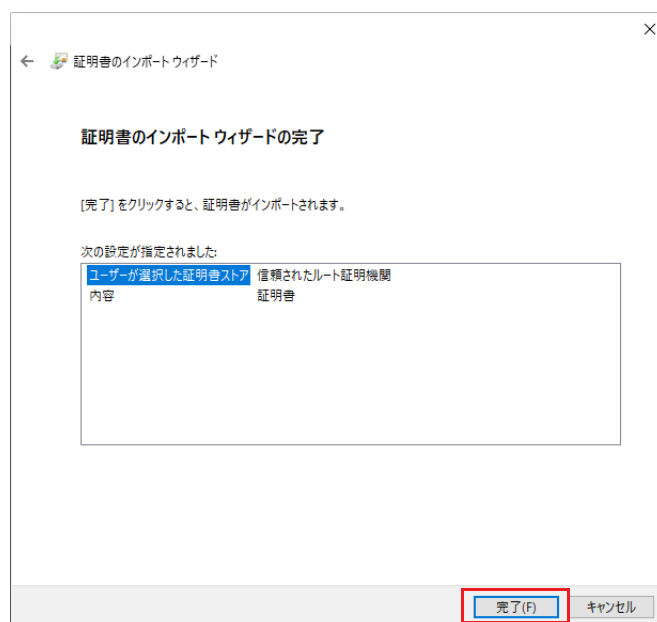
6. [証明書をすべて次のストアに配置する] を選択し、[参照] をクリックします。



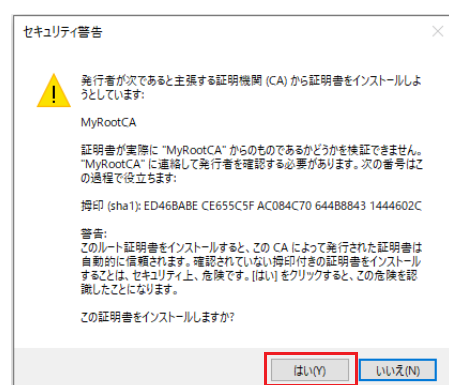
7. [信頼されたルート認証機関] を選択し、[OK] をクリックします。



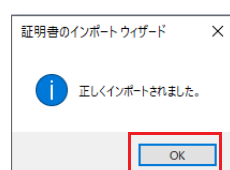
8. [完了] をクリックします。



9. [はい] をクリックします。



10. [OK] をクリックします。



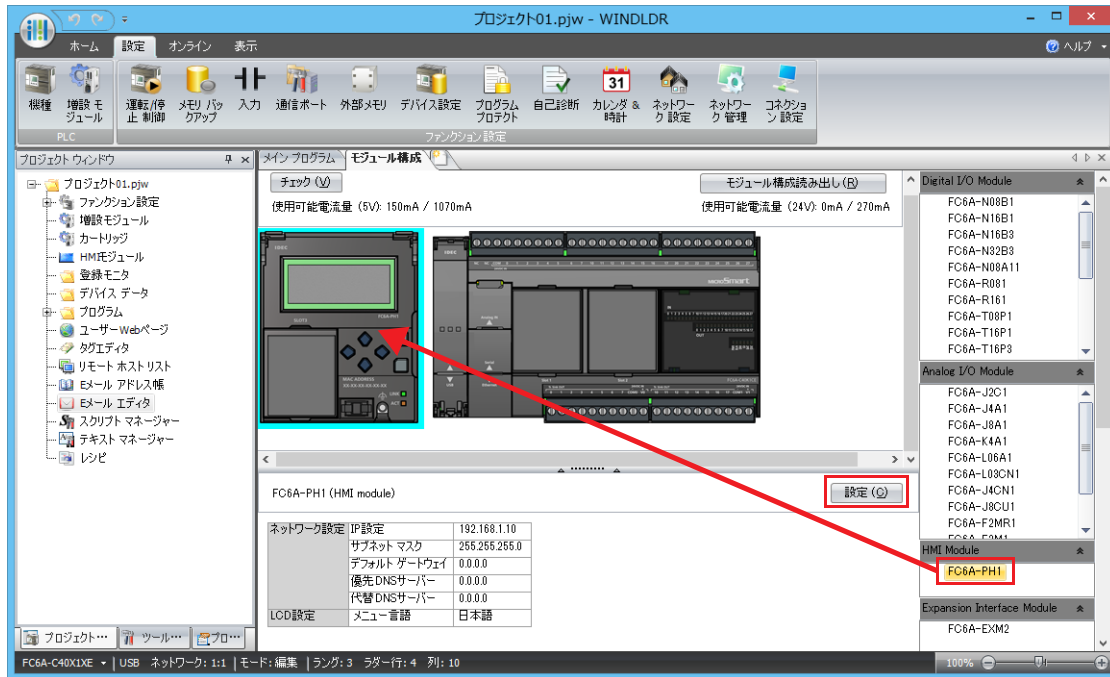
以上で、パソコン②に認証局のルート証明書がインポートされました。

HMI モジュールの Web サーバー

HMI モジュールの HMI-Ethernet ポートで Web サーバー機能を使用する場合の Web サーバーの設定について説明します。

●操作手順

1. HMI モジュールの Web サーバー設定は、モジュール構成エディタで行います。
[設定] タブの [PLC] で [増設モジュール] をクリックします。
2. モジュール構成エリアに挿入した HMI モジュールをクリックし、[設定] ボタンをクリックします。
[HMI モジュール設定] ダイアログボックスが表示されます。

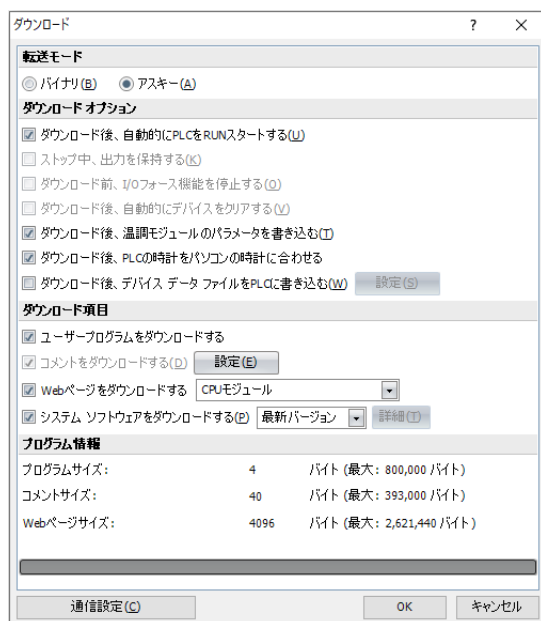


次の方法でも [HMI モジュール設定] ダイアログボックスを表示できます。

- ・プロジェクトウィンドウの [HMI モジュール] をダブルクリックする
- ・モジュール構成エリアの HMI モジュールをダブルクリックする

3. [Web サーバー] タブをクリックし、各設定項目を設定します。各設定項目については、「Web サーバー設定」(13-7 頁) を参照してください。
4. ユーザープログラムを CPU モジュールにダウンロードします。
[オンライン] タブの [転送] で [ダウンロード] ボタンをクリックします。
[ダウンロード] ダイアログボックスが表示されます。

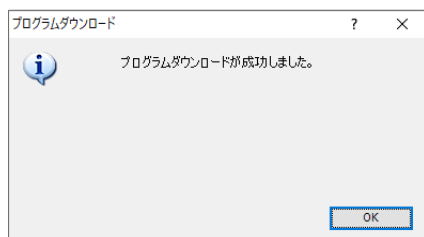
5. [Web ページをダウンロードする] チェックボックスをオンにします。



Web ページの保存先は HMI モジュールの内蔵メモリのみです。

6. [OK] ボタンをクリックします。

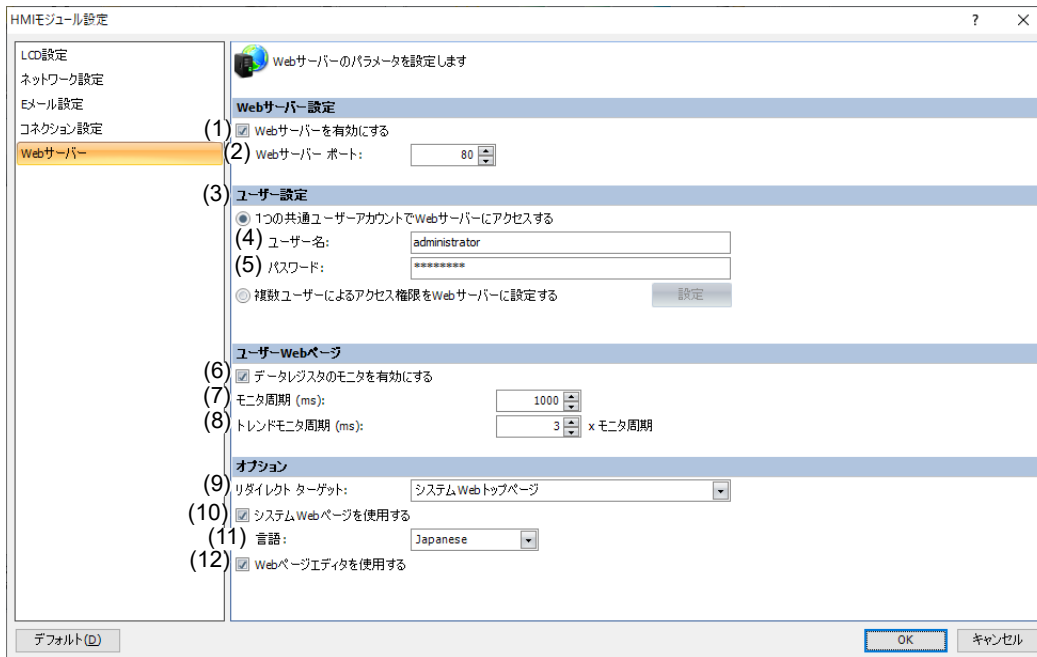
作成した Web ページが HMI モジュールの内蔵メモリにダウンロードされます。
次のメッセージが表示されたらダウンロードは成功です。[OK] ボタンをクリックします。



以上で、Web サーバーの設定が完了します。

Web サーバー設定

[HMI モジュール設定] ダイアログボックスの [Web サーバー] タブで Web サーバー機能を使用するための設定を行います。



(1) Web サーバーを有効にする

Web サーバー機能を使用する場合は、このチェックボックスをオンにします。デフォルト値はオンです。

(2) Web サーバーポート

HMI モジュールの Web サーバーが開けておくポート番号を設定します。ポート番号は 0 ～ 65535 の範囲で設定します。デフォルト値は 80 です。

(3) ユーザー設定

作成するアカウントの数（1 つまたは複数個）を設定します。次の中から設定できます。デフォルト値は [1 つの共通ユーザーアカウントで Web サーバーにアクセスする] です。

ユーザー設定	内容
1つの共通ユーザーアカウントで Webサーバーにアクセスする	1つの共通ユーザーアカウントでWebサーバーにアクセスする場合に選択します。
複数ユーザーによるアクセス権限を Webサーバーに設定する	複数のユーザーアカウントでWebサーバーへのアクセスを制限する場合に選択します。



複数のユーザーアカウントは、[ユーザーアカウント設定] ダイアログボックスで設定します。詳細は、「ユーザーアカウント設定」を参照してください。

(4) ユーザー名

(5) パスワード

ユーザー名およびパスワードを設定します。ユーザー名の最大文字数は 40 文字、パスワードの最大文字数は 16 文字です。英数字および記号のみ使用できます。[1 つの共通ユーザーアカウントで Web サーバーにアクセスする] ラジオボタンがオンの場合のみ設定できます。

項目	出荷時の設定
ユーザー名	administrator
パスワード	password

(6) データレジスタのモニタを有効にする

次の場合に、このチェックボックスをオンにします。

- HTML ページにメタ文字を埋め込みデータレジスタのモニタを使用する場合
- グラフィブラリを使用する場合

拡張子が html および htm のファイルのみ有効です。

(7) モニタ周期 (ms)

モニタを行う場合、Web ブラウザから FC6A 形ヘリクエストを送る周期 (ms) を設定します。

(8) トレンドモニタ周期 (ms)

トレンドグラフを使用したモニタを行う場合、Web ブラウザから FC6A 形へリクエストを送る周期 (ms) をモニタ周期の整数倍で設定します。

(9) リダイレクトターゲット

[リダイレクトターゲット] で設定した Web ページにリダイレクトさせることができます。

設定	説明
リダイレクトを無効にする	リダイレクトしません。
システムWebページ	システムWebページのトップページにリダイレクトします。
ユーザー Webページおよび カスタムWebページ ^{*1}	設定したユーザー WebページまたはカスタムWebページにリダイレクトします。

*1 インポートした、または保存したフォルダ名およびファイル名を選択します。

たとえば、HMI モジュールの HMI-Ethernet ポートの IP アドレスが 192.168.1.10 である場合、次の URL でリダイレクトターゲットを表示することができます。

- <http://192.168.1.10/>

(10) システム Web ページを使用する

システム Web ページを使用する場合は、このチェックボックスをオンにします。

(11) 言語

システム Web ページの言語を日本語、英語、中国語、ドイツ語またはスペイン語から設定します。

(12) Web ページエディタを使用する

Web ページエディタを使用する場合は、このチェックボックスをオンにします。詳細は、「Web ページエディタ」(13-43 頁)を参照してください。デフォルト値はオンです。

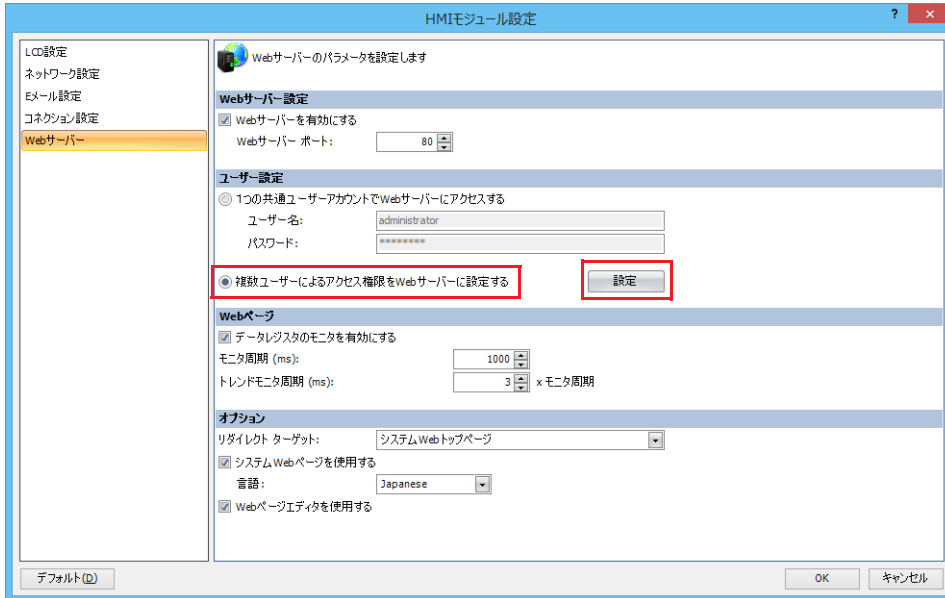
ユーザーアカウント設定

■ 複数のユーザーアカウントを作成する

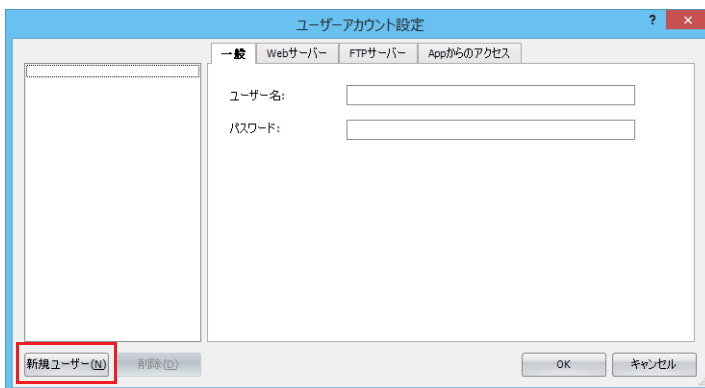
● 操作手順

1. [ファンクション設定] ダイアログボックスの [イーサネットポート 1] の [Web サーバー設定]、または [HMI モジュール設定] ダイアログボックスの [Web サーバー] タブで [複数ユーザーによるアクセス制限を Web サーバーに設定する] を選択し、[設定] ボタンをクリックします。

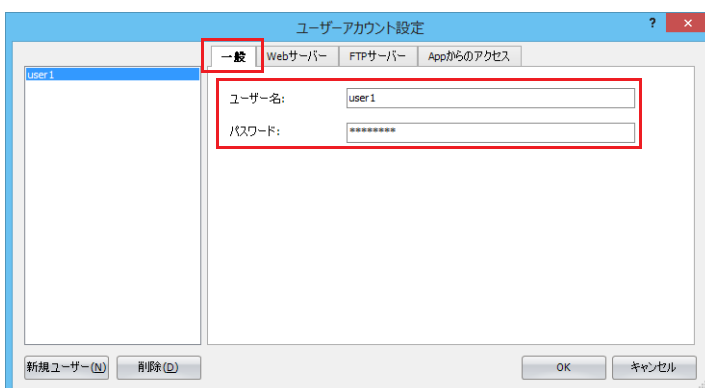
[ユーザーアカウント設定] ダイアログボックスが表示されます。



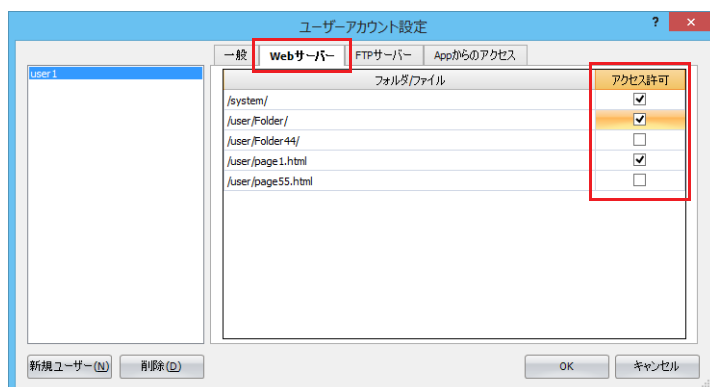
2. [新規ユーザー] ボタンをクリックします。
リストにユーザーアカウントが作成されます。



3. [一般] タブで [ユーザー名] と [パスワード] をそれぞれ設定します。



4. [Web サーバー] タブをクリックし、アクセスを許可するフォルダおよびファイルのチェックボックスをオンにします。

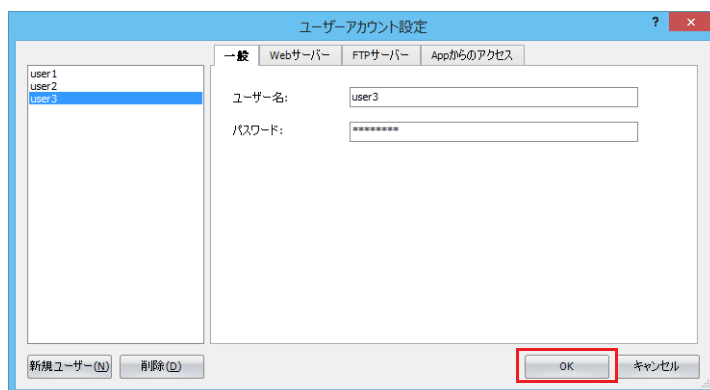


5. 手順2～4の操作を繰り返し、必要な数のユーザーアカウントを作成します。



リストでユーザーアカウントを選択すると、選択したユーザーアカウントの [一般] タブ、[Web サーバー] タブの内容を編集できます。

6. [OK] ボタンをクリックします。

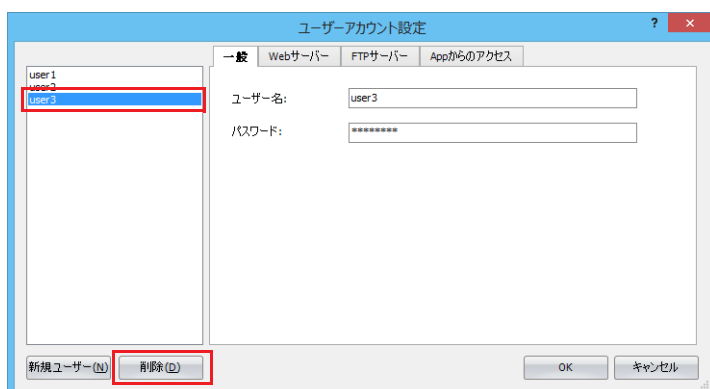


以上で、複数のユーザーアカウントの作成が完了します。

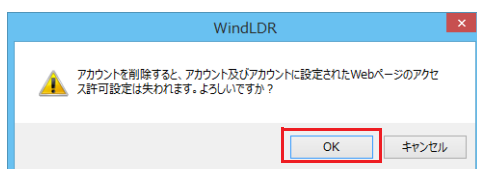
■ 作成したユーザーアカウントを削除する

● 操作手順

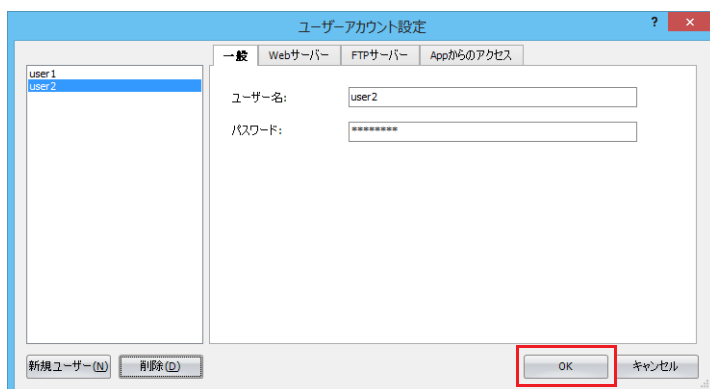
1. リストでユーザーアカウントを選択し、[削除] ボタンをクリックします。
確認メッセージが表示されます。



2. [OK] ボタンをクリックします。



3. [OK] ボタンをクリックします。



以上で、ユーザーアカウントの削除が完了します。

システム Web ページ

システム Web ページは、4 つの Web ページ（PLC ステータス表示ページ、一括モニタページ、登録モニタページおよび LCD モニタページ）で構成されています。

PLC ステータス表示ページ

PLC ステータス表示ページでは、FC6A 形のバージョンやスキャンタイムなど、動作している PLC のステータスを表示することができます。また、配置されているボタンをクリックすることで、RUN/STOP を切り替えることができます。

MICROSmartシステム ステータス

(1) システム情報

項目	値
PLC機種	FC6A-C40
システムバージョン	1.00

(2) 動作ステータス

項目	値
RUN/STOPステータス	<input type="button" value="RUN"/>
スキャンタイム(現在値)	1ミリ秒
スキャンタイム(最大値)	5ミリ秒
電池電圧	3182mV
一般エラーコード	0000

(3) 時計情報

項目	値
時計(年/月/日/曜日/時/分/秒)	2015/12/02 (水) 15:28:01

(4) ネットワーク設定

項目	値
MACアドレス	00-03-7b-f0-16-5f
IPアドレス	192.168.1.5
サブネットマスク	255.255.255.0
デフォルトゲートウェイ	0.0.0.0
優先DNSサーバー	0.0.0.0
代替DNSサーバー	0.0.0.0

(5) コネクション設定

項目	値
コネクション1-メンテナンス通信サーバー	0.0.0.0
コネクション2-メンテナンス通信サーバー	0.0.0.0
コネクション3-メンテナンス通信サーバー	0.0.0.0
コネクション4-メンテナンス通信サーバー	0.0.0.0
コネクション5-メンテナンス通信サーバー	0.0.0.0
コネクション6-メンテナンス通信サーバー	0.0.0.0
コネクション7-メンテナンス通信サーバー	0.0.0.0
コネクション8-メンテナンス通信サーバー	0.0.0.0

Webサーバー機能の詳細については、FC6Aユーザーズマニュアル 通信編を参照してください。

PLC ステータス表示ページでは、以下の項目を確認することができます。

- (1) システム情報
FC6A 形の形番とファームウェアのバージョン番号を確認できます。
- (2) 動作ステータス
RUN/STOP 状態、スキャンタイム、エラー番号を確認できます。RUN/STOP ステータスの項目のボタンをクリックすると、RUN/STOP 状態をトグルで切り替えることができます。
- (3) 時計情報
SNTP により取得した時刻情報が確認できます。
- (4) ネットワーク設定
FC6A 形に設定されているネットワーク設定を確認できます。Plus CPU モジュールの場合は、Ethernet ポート 1 と Ethernet ポート 2 のネットワーク設定を確認できます。
- (5) コネクション設定
FC6A 形が通信している相手の IP アドレスを確認できます。[メンテナンス通信サーバー]、[サーバーコネクション] については、それぞれのサーバー用ポートが通信しているクライアントの IP アドレスが、[クライアントコネクション] については、クライアント用ポートが通信しているサーバーの IP アドレスが表示されます。Plus CPU モジュールの場合は、コネクション 1 ～ 16 の通信している相手の IP アドレスが表示されます。

一括モニタページ

一括モニタページでは、連続した 200 ワードにデータレジスタに対し、データタイプをそれぞれ設定してモニタすることができます。データタイプについては「データタイプ」(13-30 頁)を参照してください。

IDEC Think Automation and Beyond ...

- PLCステータス
- 一括モニタ
- 登録モニタ
- LCD モニタ

一括モニタ(データレジスタ)

データレジスタ: 0 モニタタイプ: DEC-W ▼

	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9
D0000	43	0	0	0	99	0	0	0	0	0
D0010	0	0	776	0	0	0	0	0	0	0
D0020	0	0	0	837	0	0	0	0	0	0
D0030	0	0	0	0	0	0	0	0	0	0
D0040	0	0	0	0	0	0	0	0	0	0
D0050	0	0	0	0	0	0	0	0	0	0
D0060	0	0	0	0	0	0	0	0	0	0
D0070	0	0	0	0	0	0	0	0	0	0
D0080	0	0	0	0	0	0	0	0	0	0
D0090	0	0	0	0	0	0	0	0	0	0
D0100	0	0	0	0	0	0	0	0	0	0
D0110	0	0	0	0	0	0	0	0	0	0
D0120	0	0	0	0	0	0	0	0	0	0
D0130	0	0	0	0	0	0	0	0	0	0
D0140	0	0	0	0	0	0	0	0	0	0
D0150	0	0	0	0	0	53422	0	0	0	0
D0160	0	0	0	0	0	0	0	0	0	0
D0170	0	0	0	0	0	0	0	0	0	0
D0180	0	0	0	0	0	0	0	0	0	0
D0190	0	0	0	0	0	0	0	0	0	0

●操作手順

モニタの手順は、次のとおりです。

1. [デバイス番号] にモニタするデバイス番号を設定します。

ここで設定したデータレジスタの番号から、連続した 200 ワードのデータがモニタ対象になります。設定できるデバイス番号は下表のとおりです。範囲外の値を設定した場合、あるいは連続した 200 ワードの最後のデータが範囲境界を超える場合は、範囲内の値に自動的に変換されて入力されます。

CPU モジュールの種類	デバイス番号の範囲
All-in-One CPU モジュール	0000～7800 8000～8300 10000～55800
Plus CPU モジュール	0000～7800 8000～8700 10000～61800 70000～269800

2. [モニタタイプ] を変更し、表示する形式を設定します。

設定したデータタイプで、データのモニタが開始されます。

3. モニタ中のデータをクリックします。

[デバイスデータ書き込み] ダイアログボックスが表示されます。ここで入力した数値が FC6A 形に書き込まれます。

登録モニタページ

登録モニタページでは、最大 30 個のデバイスに対し、データタイプをそれぞれ設定してモニタすることができます。データタイプについては「データタイプ」(13-30 頁)を参照してください。

登録モニタ

データレジスタ/内部リレーを入力し、モニタタイプを選択してください。

番号	デバイス アドレス	タイプ	値
1	D0000	DEC-W ▼	43
2	D0023	DEC-I ▼	837
3		DEC-W ▼	
4		DEC-W ▼	
5		DEC-W ▼	
6		DEC-W ▼	
7		DEC-W ▼	
8		DEC-W ▼	
9		DEC-W ▼	
10		DEC-W ▼	
11		DEC-W ▼	
12		DEC-W ▼	
13		DEC-W ▼	
14		DEC-W ▼	
15		DEC-W ▼	
16		DEC-W ▼	
17		DEC-W ▼	
18		DEC-W ▼	
19		DEC-W ▼	
20		DEC-W ▼	
21		DEC-W ▼	
22		DEC-W ▼	
23		DEC-W ▼	
24		DEC-W ▼	
25		DEC-W ▼	
26		DEC-W ▼	
27		DEC-W ▼	
28		DEC-W ▼	
29		DEC-W ▼	
30		DEC-W ▼	

●操作手順

モニタの手順は、次のとおりです。

1. [デバイスアドレス] にデバイス名とデバイス番号を設定します。
設定できるデバイスは、“D” (データレジスタ) と “m” (内部リレー) です。デバイス番号は 10 進数で設定します。
(例：D2058、m0112)
2. [タイプ] を変更し、表示する形式を設定します。
設定したデータタイプで、データのモニタが開始されます。
3. [値] に表示されているモニタ中のデータをクリックします。
[デバイスデータ書き込み] ダイアログボックスが表示されます。ここで入力した数値が FC6A 形に書き込まれます。

LCD モニタページ

LCD モニタページでは、HMI モジュールの画面をモニタすることができます。また表示されているキーやショートカットをクリックすることで、HMI モジュールを操作することができます。LCD モニタページは、HMI モジュールにのみ保存されます。

システム情報

項目	値
PLC機種	FC6A-C40
システムバージョン	1.00

動作ステータス

項目	値
RUN/STOPステータス	STOP
スキャンタイム(現在値)	1ミリ秒
スキャンタイム(最大値)	2ミリ秒
電池電圧	3182mV
一般エラーコード	0000

●操作手順

モニタの手順は、次のとおりです。

1. HMI モジュールの LCD に表示されているメニューやメッセージが表示されます。
2. キーをクリックすることで、HMI モジュールのキーを押した場合と同じ動作を HMI モジュールに行います。

データタイプ

一括モニタページ上や登録モニタページ上で、データレジスタの値や内部リレーの状態を表示させたり、入力する値のタイプを設定することができます。

設定できるデータタイプは次のとおりです。

データタイプ	説明
DEC-W	ワード (10進数)
DEC-I	インテジャ (10進数)
DEC-D	ダブルワード (10進数)
DEC-L	ロング (10進数)
DEC-F	フロート (10進数)
HEX-W	ワード (16進数)
HEX-D	ダブルワード (16進数)
BIN-B	2進数

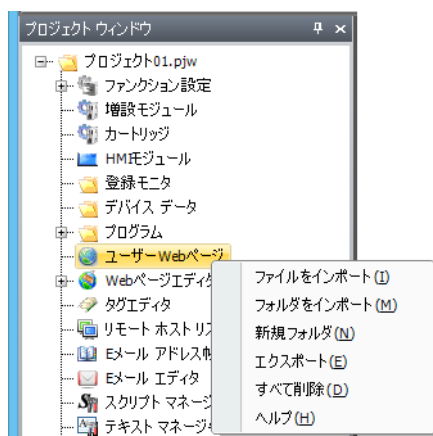
ワード、インテジャ、ダブルワード、ロングおよびフロートについては、ラダープログラミングマニュアル「第3章 データタイプ」を参照してください。

ユーザー Web ページ

HTML や JavaScript を使って自由に作成した Web ページ（ユーザー Web ページ）をインポートすることで、自由度の高い Web ページを作成できます。

ユーザー Web ページツリーの操作

プロジェクトウィンドウの「ユーザー Web ページ」の項目および「Web ページ」として登録してあるファイルやフォルダに対して、次の操作を行い、ユーザー Web ページツリーを構築できます。



Web ページツリーの作成

プロジェクトウィンドウで Web ページツリーを作成します。

次の手順では、下記の構成の Web ページツリーを作成する方法を例にして説明します。

Web ページツリーの構成

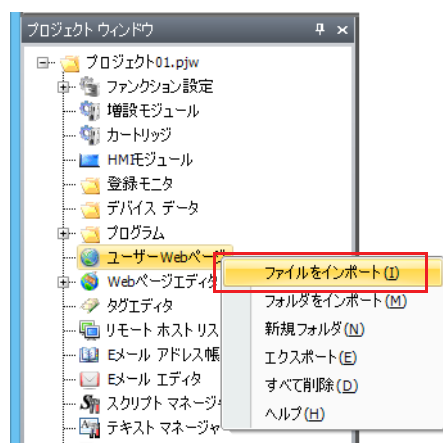
page1.html

Folder/page2.html

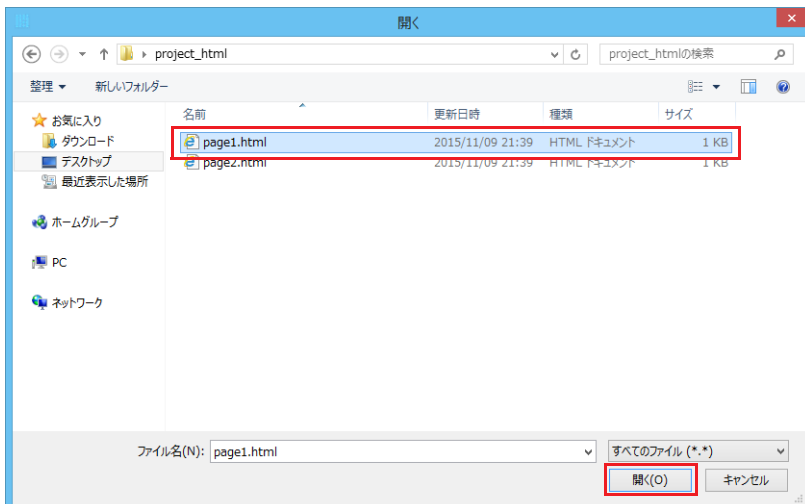
FC6A 形上で表示する page1.html および page2.html を作成しておきます。

●操作手順

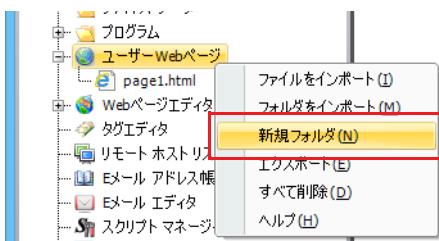
1. [ユーザー Web ページ] を右クリックし、[ファイルをインポート] をクリックします。
[ファイルを開く] ダイアログボックスが表示されます。



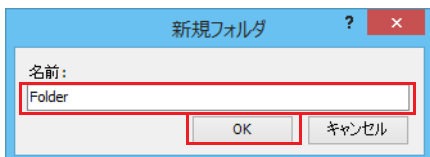
- 「page1.html」を選択し、「開く」ボタンをクリックします。



- 「ユーザー Web ページ」を右クリックし、「新規フォルダ」をクリックします。
「新規フォルダ」ダイアログボックスが表示されます。

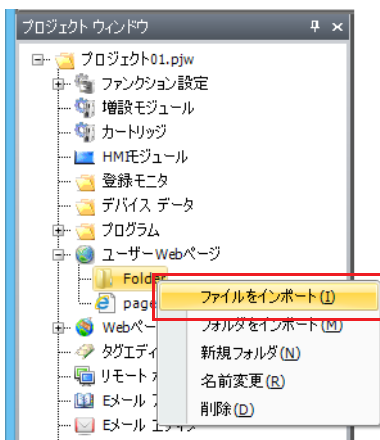


- 新規作成するフォルダの名称として「Folder」を入力し、「OK」ボタンをクリックします。
「ユーザー Web ページ」に「Folder」フォルダが作成されます。

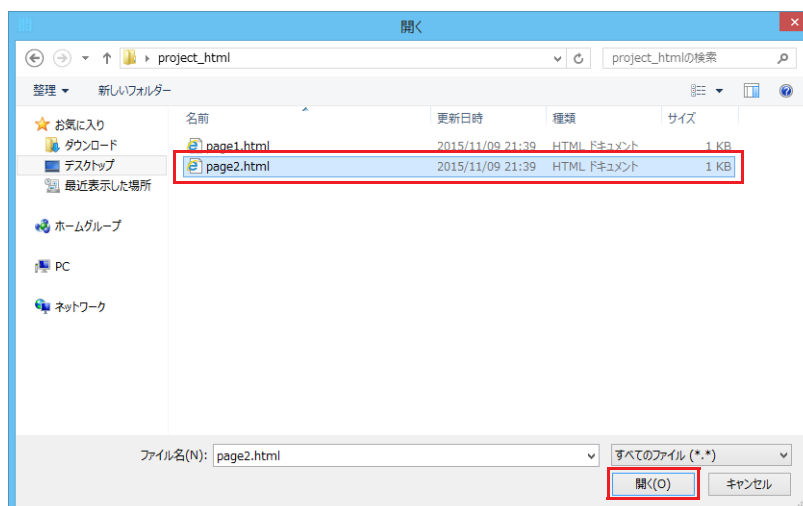


間違ったフォルダ名を入力した場合は、該当のフォルダを右クリックし、「名前を変更」をクリックすると、フォルダ名を変更できます。

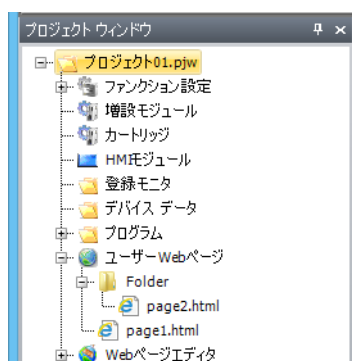
- 手順4で作成した「Folder」フォルダを右クリックし、「ファイルをインポート」をクリックします。
「ファイルを開く」ダイアログボックスが表示されます。



6. 「page2.html」を選択し、[開く] ボタンをクリックします。



プロジェクトウィンドウには、次のように表示されます。



Web ページツリーのインポート

あらかじめツリー構成まで考慮した HTML ファイルツリーがある場合は、そのツリー全体をインポートしてユーザー Web ページに追加できます。

プロジェクトウィンドウ内の [ユーザー Web ページ] または [ユーザー Web ページ] 以下にあるフォルダを右クリックし、[フォルダをインポート] をクリックするとダイアログボックスが表示されます。インポートする Web ページツリーのあるフォルダを選択すると、設定したフォルダ以下すべてのファイルがインポートされます。

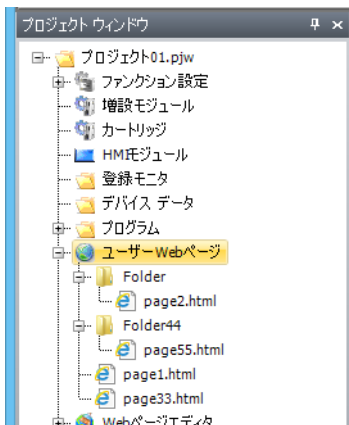
Web ページツリーのエクスポート

ユーザー Web ページデータをエクスポートすることで、Web ページツリーの再構築やファイルの変更などできます。プロジェクトウィンドウ内の [ユーザー Web ページ] を右クリックし、[フォルダをエクスポート] をクリックするとダイアログボックスが表示されます。エクスポート先のフォルダを設定すると、ユーザー Web ページのデータがエクスポートされます。

ユーザー Web ページの削除

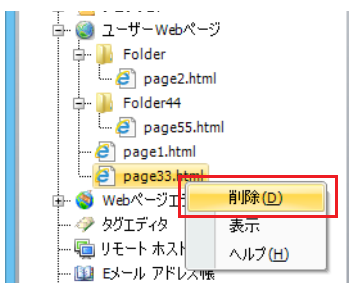
プロジェクトウィンドウで Web ページツリーを削除します。

次の手順では、Web ページツリーから page33.html と Folder44 を削除する方法を例にして説明します。

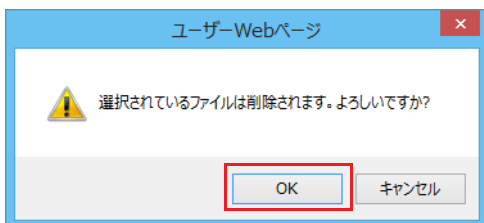


●操作手順

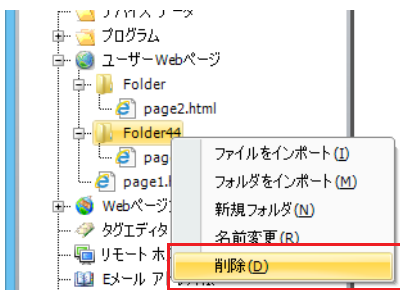
1. 「page33.html」を右クリックし、[削除] をクリックします。
削除の確認メッセージが表示されます。



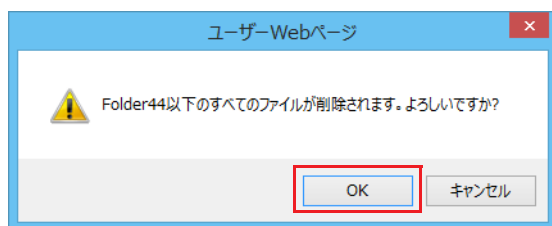
2. [OK] をクリックします。
「page33.html」が削除されます。



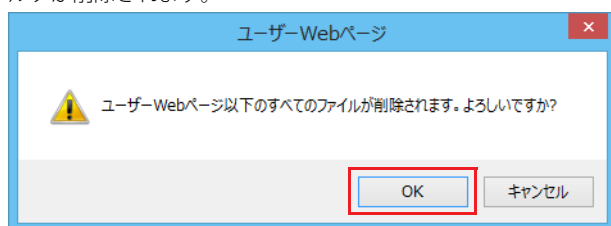
3. [Folder44] フォルダを右クリックし、[削除] をクリックします。
削除の確認メッセージが表示されます。



4. [OK] ボタンをクリックします。
[Folder44] フォルダ 以下すべてのファイルおよびフォルダが削除されます。



- すべてのユーザー Web ページを削除する場合は、プロジェクトウィンドウの [ユーザー Web ページ] を右クリックし、[すべて削除] をクリックします。削除の確認メッセージが表示されますので [OK] ボタンをクリックすると、登録していたすべてのファイルおよびフォルダが削除されます。



- ユーザー Web ページとしてインポートする htm/html ファイルでは、エンコーディングは UTF8 を使用してください。また、WindLDR にインポートするファイルは、別途 PC 上にバックアップをしてください。

モニタ機能

ユーザー Web ページに特定のフォーマットの文字列（メタ文字）が含まれる場合、その文字列をデータレジスタの値または内部リレーの状態に置き換えて表示します。この機能を使用するには、次のダイアログボックスで「データレジスタのモニタを有効にする」チェックボックスをオンにしてください。

- ・ [ファンクション設定] ダイアログボックスの [イーサネットポート 1] の [Web サーバー設定]
- ・ [HMI モジュール設定] ダイアログボックスの [Web サーバー] タブ



WindLDR でプログラムを変換するとき、インポートされた htm/html ファイルの head 要素内に次の script 要素を追加します。

```
<script type="text/javascript" src="/system/lib/jquery.js"></script>
<script type="text/javascript" src="/system/lib/idec.js"></script>
<script type="text/javascript" src="/system/lib/idec_glib.js"></script>
<script type="text/javascript" src="/system/lib/idec_tag_start.js"></script>
```

メタ文字のフォーマット

■ データレジスタの値

{{ 読み書きタイプ, デバイス, デバイス番号, ドライバ番号, ネットワーク番号, データタイプ, 最大サイズ }}

パラメータ	説明
読み書きタイプ	RまたはWを設定します。Rは読み出しのみ、Wは書き込みおよび読み出しすることができます。
デバイス	モニタしたいデバイスの記号を設定します。D（データレジスタ）固定です。
デバイス番号	モニタしたいデバイス番号を10進数で設定します。
ドライバ番号	ドライバ番号です。0 固定です。
ネットワーク番号	ネットワーク番号です。0 固定です。
データタイプ	データタイプを設定します。*1
最大サイズ	データタイプが数値型の場合、設定不要です。 データタイプが文字列型の場合、読み出しまたは書き込みを行う最大サイズをバイト単位で設定します。 最小値は1、最大値は63です。

*1 次の中から設定します。

種別	データタイプ	説明
数値型	DEC-W	ワード（10進数）
	DEC-I	インテジャ（10進数）
	DEC-D	ダブルワード（10進数）
	DEC-L	ロング（10進数）
	DEC-F	フロート（10進数）
	HEX-W	ワード（16進数）
	HEX-D	ダブルワード（16進数）
文字列型	STR_ASCII	ASCII文字列
	STR_HOSTNAME	ASCII文字列
	STR_EMAILADR	ASCII文字列
	STR_PASSWORD	ASCII文字列

ワード、インテジャ、ダブルワード、ロングおよびフロートについては、ラダープログラミングマニュアル「第3章 データタイプ」を参照してください。



- ・ [読み書きタイプ] に W（書き込み）を設定した場合、Web ブラウザ上でその数値をクリックすると、書き込みダイアログボックスが表示され、デバイスに値を書き込むことができます。
- ・ Plus CPU モジュールまたは HMI モジュールとの通信が正常に行われなかった場合や存在しないデバイス番号のデータを取得しようとした場合、メタ文字は -（ハイフン）で置き換えられます。

[データタイプ] に文字列型のデータタイプを設定した場合、読み出しおよび書き込みはワード単位で、上位バイト、下位バイトの順で処理します。読み出しは、設定したデバイスの上位バイトから、端末文字 NULL (0x00) まで、または [最大サイズ] で指定したサイズ分のデータまでを読み出します。書き込みは、設定したデバイスの上位バイトから、[最大サイズ] ÷ 2 ワード分（小数点切り上げ）のデータを書き込みます。入力した文字列の末尾以降は、端末文字 NULL (0x00) で埋められます。



- ・ [読み書きタイプ] に W (書き込み) を設定し、かつ [データタイプ] に STR_HOSTNAME を設定した場合、書き込みダイアログボックスに入力した文字列がホストとして正しい書式である場合のみ書き込みを行います。書式のチェックは書き込み処理開始時に行います。
- ・ [読み書きタイプ] に W (書き込み) を設定し、かつ [データタイプ] に STR_EMAILADR を設定した場合、書き込みダイアログボックスに入力した文字列が E メールアドレスとして正しい書式である場合のみ書き込みを行います。書式のチェックは書き込み処理開始時に行います。
- ・ [データタイプ] に STR_PASSWORD を設定した場合、データレジスタの値を伏せ字で表示します。伏せ字で使用される文字は Web ブラウザごとに異なります。

■ 内部リレーの状態

{{ 読み書きタイプ, デバイス, デバイス番号, ドライバ番号, ネットワーク番号, データタイプ, OFF データ, ON データ }}

パラメータ	説明
読み書きタイプ	RまたはWを設定します。Rは読み出しのみ、Wは書き込みおよび読み出しすることができます。
デバイス	モニタしたいデバイスの記号を設定します。m (内部リレー) 固定です。
デバイス番号	モニタしたいデバイス番号を10進数で設定します。
ドライバ番号	ドライバ番号です。0 固定です。
ネットワーク番号	ネットワーク番号です。0 固定です。
データタイプ	読み出しまたは書き込みするデータのタイプを設定します。*1
OFFデータ	データタイプがBIT_FORMの場合、設定した内部リレーがOFFのときに表示される文字列を設定します。 データタイプがBITの場合、設定した内部リレーがOFFのときに表示される画像を設定します。画像はファイル名で設定し、このユーザー webページが配置されているフォルダからの相対パスで設定します。
ONデータ	データタイプがBIT_FORMの場合、設定した内部リレーがONのときに表示される文字列を設定します。 データタイプがBITの場合、設定した内部リレーがONのときに表示される画像を設定します。画像はファイル名で設定し、このユーザー webページが配置されているフォルダからの相対パスで設定します。

*1次の中から設定します。

データタイプ	説明
BIT_FORM	ビットデバイスの状態を文字列で表示します。
BIT	ビットデバイスの状態を画像で表示します。



- ・ [読み書きタイプ] に W (書き込み) を設定し、かつ [データタイプ] に BIT_FORM を設定した場合、フォームのボタンが表示されます。ボタン上に内部リレーの ON/OFF 状態が表示されます。ボタンをクリックして ON と OFF を切り替えることができます。
- ・ [読み書きタイプ] に W (書き込み) を設定し、かつ [データタイプ] に BIT を設定した場合、画像をクリックして ON と OFF を切り替えることができます。
- ・ Plus CPU モジュールまたは HMI モジュールとの通信が正常に行われなかった場合や存在しないデバイス番号のデータを取得しようとした場合、メタ文字は - (ハイフン) で置き換えられます

例

記述例	内容
{{R,m,123,0,0,BIT_FORM,BITOFF,BITON}}	M0123がOFFのとき"BITOFF"を表示します。 M0123がONのとき"BITON"を表示します。
{{R,m,123,0,0,BIT,img/BIT_IMG_OFF.jpg,img/BIT_IMG_ON.jpg}}	M0123がOFFのときimgフォルダ内のBIT_IMG_OFF.jpgを表示します。 M0123がONのときimgフォルダ内のBIT_IMG_ON.jpgを表示します。



ユーザー Web ページを SD メモリカードに保存する場合、[OFF データ] および [ON データ] に WEBPAGE フォルダより上のフォルダを設定しても WEBPAGE フォルダが設定されたものとして扱われます。



{{ と }} の間に余分なスペース文字や改行文字等を記述すると、動作しません。

グラフのフォーマット

棒グラフ（縦）

モニタ機能を有効にした場合、HTML 形式のファイルにおいて div タグと、その div タグ内にパラメータを記述することで、自動的に JavaScript のリクエストに変換し、CGI を通じて取得したデータレジスタの値を棒グラフ（縦）の形式で表示します。

棒グラフ（縦）を描画するためには、div タグに対し、グラフごとに固有の ID を示す属性 id となる文字列を設定したうえで、データの種類を示す属性 data-graph として "vbar" を設定し、div タグ内のパラメータを記述します。

設定できるパラメータは、以下のものです。パラメータがデフォルト値のままであればよい場合は、設定する必要はありません。

device	: デバイスです。D（データレジスタ）固定です。
address	: デバイス番号です。数値で設定してください。
driver	: ドライバ番号です。0 固定です。
net_no	: ネットワーク番号です。0 固定です。
format	: Web データタイプの文字列です。
width	: div 枠の幅（ピクセル数）です。デフォルト値は 300 です。
height	: div 枠の高さ（ピクセル数）です。デフォルト値は 300 です。
line_col	: div 枠のライン色です。デフォルト値は "#000000" です。
barvgutter	: div 枠からグラフに対する垂直方向の間隔（ピクセル数）です。デフォルト値は 30 です。
barwidth	: グラフの幅（ピクセル数）です。デフォルト値は 20 です。
gutter	: 複数のグラフを表示する場合の、各グラフ間の間隔（ピクセル数）です。デフォルト値は 20 です。
type	: グラフの先端の形状です。"square"（角形）、"round"（円形）、"sharp"（鋭頭）、"soft"（鈍頭）から選択できます。デフォルト値は "square" です。
scalewidth	: 目盛りの幅（ピクセル数）です。デフォルト値は 5 です。
labelvgutter	: ラベルを表示する、グラフからの垂直方向の間隔（ピクセル数）です。デフォルト値は 20 です。
labelhgutter	: ラベルを表示する、グラフからの水平方向の間隔（ピクセル数）です。デフォルト値は 30 です。
bars[]	: 各グラフごとの設定を行う、配列のパラメータです。各配列のパラメータは {} 内に記述し、配列と配列は " " で区切ります。

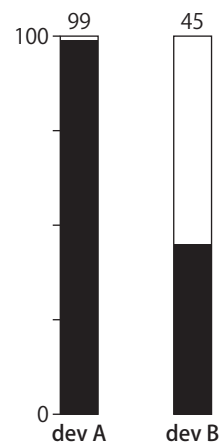
bars[] の要素として設定できるパラメータは、以下のものです。パラメータがデフォルト値のままであればよい場合は、設定する必要はありません。

min_val:	: グラフの最小値です。必須のパラメータです。
max_val	: グラフの最大値です。必須のパラメータです。
label	: ラベル名です。デフォルト値はデバイスとデバイス番号とを組み合わせた文字列です。
back_col	: グラフの背景色です。デフォルト値は "#FFFFFF" です。
front_col	: グラフの前景色です。デフォルト値は要素ごとに異なった色になります。
scale_on	: グラフへの目盛りの有無です。デフォルト値は false（目盛りなし）です。
scale_lbl_on	: グラフへの目盛りに対するラベルの有無です。デフォルト値は false（ラベルなし）です。

以下のように記述すると、D2040 と D2042 のダブルワード（10 進数）のデータを対象とした 2 本の棒グラフ（縦）を表示します。

```
<div id="div11" data-graph="vbar">
```

```
device:"D", address:2040, driver:0, net_no:0, format:"DEC-D",
width:300, height:300, line_col:"#000000",
barvgutter:30, barwidth:20,
gutter:20, type:"square", scalewidth:5,
bars:[
{
min_val:0, max_val:100, label:"dev A",
back_col:"#FFFFFF", front_col:"#0000FF",
scale_on:true, scale_lbl_on:true
},
{
min_val:0, max_val:100, label:"dev B",
back_col:"#FFFFFF", front_col:"#FF0000",
scale_on:false, scale_lbl_on:false
}
]
</div>
```



棒グラフ（横）

モニタ機能を有効にした場合、HTML 形式のファイルにおいて div タグと、その div タグ内にパラメータを記述することで、自動的に JavaScript のリクエストに変換し、CGI を通じて取得したデータレジスタの値を棒グラフ（横）の形式で表示します。棒グラフ（横）を描画するためには、div タグに対し、グラフごとに固有の ID を示す属性 id となる文字列を設定したうえで、データの種類を示す属性 data-graph として "hbar" を設定し、div タグ内のパラメータを記述します。設定できるパラメータは、以下のものです。パラメータがデフォルト値のままであれば、設定する必要はありません。

device : デバイスです。D（データレジスタ）固定です。
 address : デバイス番号です。数値で設定してください。
 driver : ドライバ番号です。0 固定です。
 net_no : ネットワーク番号です。0 固定です。
 format : Web データタイプの文字列です。
 width : div 枠の幅（ピクセル数）です。デフォルト値は 300 です。
 height : div 枠の高さ（ピクセル数）です。デフォルト値は 300 です。
 line_col : div 枠のライン色です。デフォルト値は "#000000" です。
 barhgutter : div 枠からグラフに対する水平方向の間隔（ピクセル数）です。デフォルト値は 30 です。
 barwidth : グラフの幅（ピクセル数）です。デフォルト値は 20 です。
 gutter : 複数のグラフを表示する場合の、各グラフ間の間隔（ピクセル数）です。デフォルト値は 20 です。
 type : グラフの先端の形状です。"square"（角形）、"round"（円形）、"sharp"（鋭頭）、"soft"（鈍頭）から選択できます。デフォルト値は "square" です。
 scalewidth : 目盛りの幅（ピクセル数）です。デフォルト値は 5 です。
 labelvgutter : ラベルを表示する、グラフからの垂直方向の間隔（ピクセル数）です。デフォルト値は 20 です。
 labelhgutter : ラベルを表示する、グラフからの水平方向の間隔（ピクセル数）です。デフォルト値は 30 です。
 bars[] : 各グラフごとの設定を行う、配列のパラメータです。各配列のパラメータは {} 内に記述し、配列と配列は "," で区切ります。

bars[] の要素として設定できるパラメータは、以下のものです。パラメータがデフォルト値のままであれば、設定する必要はありません。

min_val : グラフの最小値です。必須のパラメータです。
 max_val : グラフの最大値です。必須のパラメータです。
 label : ラベル名です。デフォルト値はデバイスとデバイス番号とを組み合わせさせた文字列です。
 back_col : グラフの背景色です。デフォルト値は "#FFFFFF" です。
 front_col : グラフの前景色です。デフォルト値は要素ごとに異なった色になります。
 scale_on : グラフへの目盛りの有無です。デフォルト値は false（目盛りなし）です。
 scale_lbl_on : グラフへの目盛りに対するラベルの有無です。デフォルト値は false（ラベルなし）です。

以下のように記述すると、D2040 と D2042 のダブルワード（10 進数）のデータを対象とした 2 本の棒グラフ（横）を表示します。

```
<div id="div21" data-graph="hbar">
  device:"D", address:2040, driver:0, net_no:0, format:"DEC-D",
  width:300, height:300, line_col:"#000000",
  barhgutter:30, barwidth:20,
  gutter:20, type:"square", scalewidth:5,
  bars:[
    {
      min_val:0, max_val:100, label:"dev A",
      back_col:"#FFFFFF", front_col:"#0000FF",
      scale_on:true, scale_lbl_on:true
    },
    {
      min_val:0, max_val:100, label:"dev B",
      back_col:"#FFFFFF", front_col:"#FF0000",
      scale_on:false, scale_lbl_on:false
    }
  ]
</div>
```



トレンドグラフ

モニタ機能を有効にした場合、HTML 形式のファイルにおいて div タグと、その div タグ内にパラメータを記述することで、自動的に JavaScript のリクエストに変換し、CGI を通じて取得したデータレジスタの値をトレンドグラフの形式で表示します。トレンドグラフを描画するためには、div タグに対し、グラフごとに固有の ID を示す属性 id となる文字列を設定したうえで、データの種類を示す属性 data-graph として "trend" を設定し、div タグ内のパラメータを記述します。設定できるパラメータは、以下のものです。パラメータがデフォルト値のままであればよい場合は、設定する必要はありません。

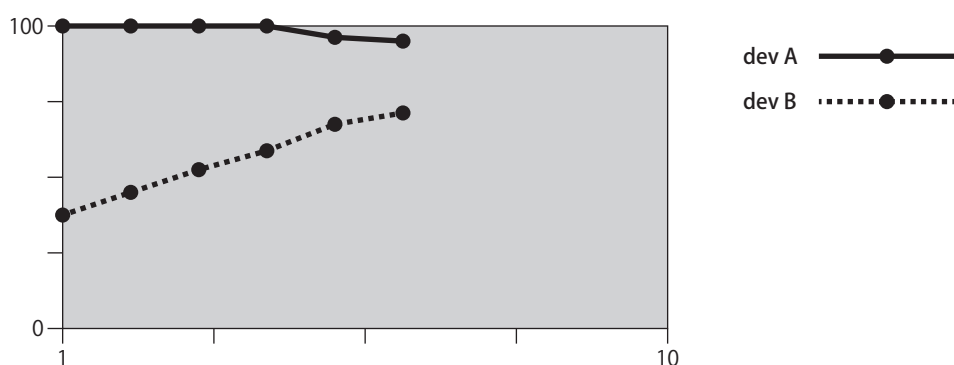
device	: デバイスです。D (データレジスタ) 固定です。
address	: デバイス番号です。数値で設定してください。
driver	: ドライバ番号です。0 固定です。
net_no	: ネットワーク番号です。0 固定です。
format	: Web データタイプの文字列です。
width	: div 枠の幅 (ピクセル数) です。デフォルト値は 300 です。
height	: div 枠の高さ (ピクセル数) です。デフォルト値は 300 です。
line_col	: div 枠のライン色です。デフォルト値は "#000000" です。
min_val	: グラフの最小値です。必須のパラメータです。
max_val	: グラフの最大値です。必須のパラメータです。
plot_num	: データをプロットする数を設定します。必須のパラメータです。
scale_col	: 目盛りの色です。デフォルト値は "#000000" です。
x_val_col	: X 軸のラベルの色です。デフォルト値は "#000000" です。
y_val_col	: Y 軸のラベルの色です。デフォルト値は "#000000" です。
g_x	: グラフ枠の水平方向の描画開始位置 (ピクセル数) です。デフォルト値は 50 です。
g_y	: グラフ枠の垂直方向の描画開始位置 (ピクセル数) です。デフォルト値は 30 です。
g_width	: グラフ枠の水平幅 (ピクセル数) です。デフォルト値は 200 です。
g_height	: グラフ枠の垂直幅 (ピクセル数) です。デフォルト値は 100 です。
g_line_col	: グラフ枠の色です。デフォルト値は line_col と同じ値です。
g_line_width	: グラフ枠の太さ (ピクセル数) です。デフォルト値は 1 です。
g_back_col	: グラフ枠の背景です。デフォルト値は "#C0C0C0" です。
mode	: データがプロット領域を超える場合に、切り捨てるデータを設定します。"one" (ひとつだけ切り捨てる)、"half" (全体の半分を切り捨てる)、"all" (すべて切り捨てる) から選択できます。デフォルト値は "all" です。
legend_gutter	: グラフ枠から凡例までの幅 (ピクセル数) です。デフォルト値は 30 です。
legend_margin_x	: 凡例内の水平方向の余白 (ピクセル数) です。デフォルト値は 10 です。
legend_margin_y	: 凡例内の垂直方向の余白 (ピクセル数) です。デフォルト値は 10 です。
legend_line_width	: 凡例内の線の長さ (ピクセル数) です。デフォルト値は 20 です。
legend_line_gutter	: 凡例内の複数の線を表示する場合の垂直方向の幅 (ピクセル数) です。デフォルト値は 20 です。
legend_line_col	: 凡例内の線の色です。デフォルト値は line_col と同じ値です。
legend_width	: 凡例内の水平方向の幅 (ピクセル数) です。デフォルト値は 100 です。
line_width	: グラフ上のラインの太さの初期値 (ピクセル数) です。デフォルト値は 1 です。
marker_on	: グラフ上のマーカースの有無の初期値です。デフォルト値は false (マーカースなし) です。
marker_width	: グラフ上のマーカースの大きさの初期値 (ピクセル数) です。デフォルト値は 3 です。
lines[]	: 各線ごとの設定を行う、配列のパラメータです。各配列のパラメータは {} 内に記述し、配列と配列は "/" で区切ります。

lines[] の要素として設定できるパラメータは、以下のものです。パラメータがデフォルト値のままであればよい場合は、設定する必要はありません。

label	: ラベル名です。デフォルト値はデバイスとデバイス番号とを組み合わせた文字列です。
front_col	: ライン毎に設定できる、ラインの前景色です。 デフォルト値は要素ごとに異なった色になります。
marker_col	: ラインのマーカースの色です。デフォルト値は front_col と同じ値です。
line_width	: ライン毎に設定できる、ラインの太さ (ピクセル数) です。 初期値の line_width の値より、こちらのほうが優先されます。
marker_width	: ライン毎に設定できる、マーカースの大きさ (ピクセル数) です。 初期値の marker_width の値より、こちらのほうが優先されます。

以下のように記述すると、D2040 と D2042 のダブルワード（10 進数）のデータを対象としたプロット数 10 点、マーカー付きのトレンドグラフを表示します。

```
<div id="div31" data-graph="trend">
  device:"D", address:2040, driver:0, net_no:0, format:"DEC-D",
  width:400, height:300, line_col:"#000000",
  min_val:0, max_val:100, plot_num:10,
  line_width:3, marker_on:true, marker_width:5,
  lines:[
    {
      label:"dev A",
      front_col:"#0000FF"
    },
    {
      label:"dev B",
      front_col:"#FF0000"
    }
  ]
</div>
```



グラフ描画に関する注意



- グラフを記述する `<div>` ~ `</div>` の間の文字列は、JavaScript プログラムのパラメータとして渡されます。よって、任意の位置へ空白や改行を含めることは可能ですが、コメントを含めることはできませんので、記述する際はご注意ください。
- パラメータとして数値を設定しないものは "" で囲んでください。
- すべてのパラメータは "," で区切ってください。また、最終パラメータのあとに "," を置かないようにしてください。パラメータのフォーマットが不正規である場合は、グラフとして描画されません。
- Web ブラウザの種類によって、表示のされかたが異なる場合があります。
- 更新頻度やデータのプロット数などによっては、Web ブラウザの表示の更新が遅くなる可能性があります。

JavaScript 関数

モニタ機能を有効にした場合、デバイスデータの読み書きを JavaScript の関数を通じて行うことができます。



JavaScript 関数は、内部的には CGI を使用します。CGI については「CGI」(13-55 頁) を参照してください。

■ デバイスデータ読み出し関数

`idec.device_read` (device, address, length, driver, net_no)

- `device` : 読み出し対象のデバイスを文字列で設定します。
現在は D (データレジスタ) のみ使用できます。
- `address` : 読み出したいデバイス番号を設定します。
- `length` : `address` を先頭として読み出したいデータのサイズを設定します。
10 進数で 1 ~ 64 の範囲で設定できます。
- `driver` : ドライバ番号です。0 固定です。
- `net_no` : ネットワーク番号です。0 固定です。

戻り値は 16 進数数値の文字列で "XXXX" の形式となります。length が 2 以上の場合は、データとデータの間 "_" が入ります。サーバーから応答がない場合やパラメータが正しくない場合は、戻り値が " 空文字列 " になります。



D2058 が 49910 (C2F6h)、D2059 が 59768 (E978h) の場合は、以下のようになります。

```
var raw_data = idec.device_read ("D", 2058, 2, 0, 0) ;
```

正常終了時の `raw_data` は "C2F6_E978" になります。

■ デバイスデータ書き込み関数

`idec.device_write` (device, address, length, driver, net_no, data)

- `device` : 書き込み対象のデバイスを文字列で設定します。現在は D (データレジスタ) のみ使用できます。
- `address` : 書き込みたいデバイス番号を設定します。
- `length` : `address` を先頭として書き込みたいデータのサイズを設定します。10 進数で 1 ~ 64 の範囲で設定できます。
- `driver` : ドライバ番号です。0 固定です。
- `net_no` : ネットワーク番号です。0 固定です。
- `data` : 書き込みたいデータを 16 進数数値の文字列で設定します。length が 2 以上の場合は、データとデータの間 "_" を含めます。

戻り値は、正常終了時には true、それ以外は false となります。



D2058 に 49910 (C2F6h)、D2059 に 59768 (E978h) を書き込む場合は、以下のようになります。

```
var status_write = idec.device_write ("D", 2058, 2, 0, 0, "C2F6_E978") ;
```

正常終了時の `status_write` は true になります。

カスタム Web ページ

WindLDR で用意されたツールの Web ページエディタを使用して、目的や用途に応じた Web ページ（カスタム Web ページ）を簡単に作成できます。カスタム Web ページの作成については、「Web ページエディタ」（13-43 頁）を参照してください。

Web ページエディタ

Web ページエディタでは、FC6A 形にダウンロードするユーザー Web ページを作成できます。
FC6A 形にダウンロードしたユーザー Web ページは、パソコンなどの端末から Web ブラウザでアクセスでき、FC6A 形の各デバイスの値に応じたユーザー Web ページを表示できます。
Web ページエディタを使用するには、モジュール構成エリアに挿入した HMI モジュールをクリックし、[設定] ボタンをクリックして表示される [HMI モジュール設定] ダイアログボックスで [Web ページエディタを使用する] チェックボックスをオンにしてください。

作成したユーザー Web ページと WindLDR のシミュレーション機能を連動することで、ユーザー Web ページを FC6A 形にダウンロードすることなく、FC6A 形の各デバイスの値に応じたユーザー Web ページの動作を確認できます。

カスタム Web ページの作成

Web ページエディタでカスタム Web ページを作成する操作手順を説明します。
入力 M0000 が ON で入力 M0001 が OFF の場合、出力 Q1 が ON する Web ページを作成します。

Web ページで配置する表示部品



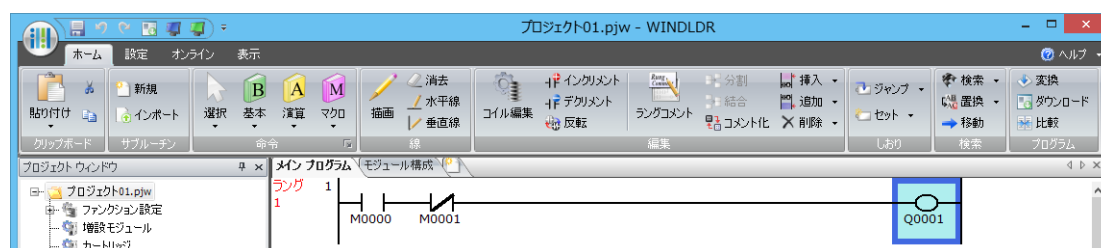
WindLDR で作成するプログラム



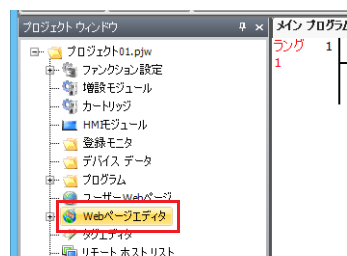
デバイスについては、「第 2 章 デバイス」（2-1 頁）を参照してください。
命令語については、ラダープログラミング マニュアルを参照してください。

●操作手順

1. 入力 M0000 が ON で入力 M0001 が OFF の場合、出力 Q1 が ON するプログラムを作成します。
詳細は、FC6A 形マイクロスマート ユーザーズ マニュアル「第 4 章 ラダープログラムの作成」を参照してください。

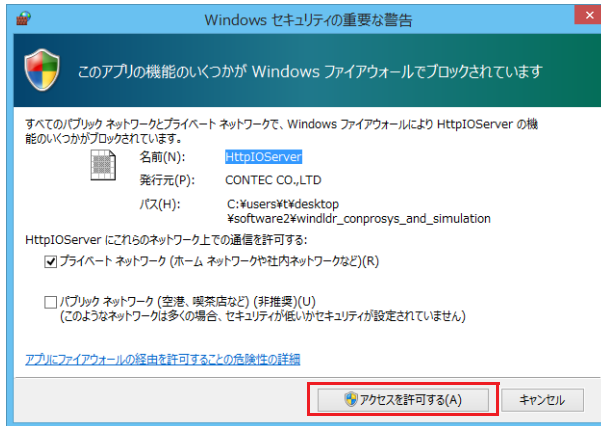


2. プロジェクトウィンドウの [Web ページエディタ] をダブルクリックします。
Web ページエディタが起動します。

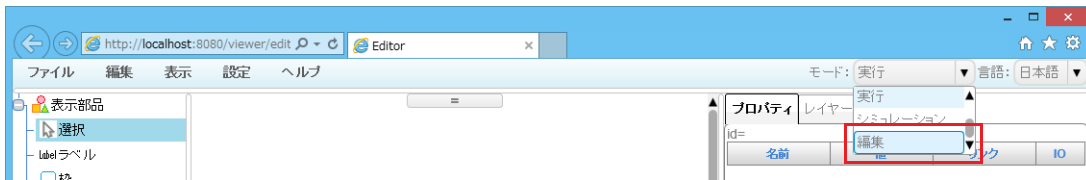




[Windows セキュリティの重要な警告] ダイアログボックスが表示された場合は、[アクセスを許可する] ボタンをクリックします。

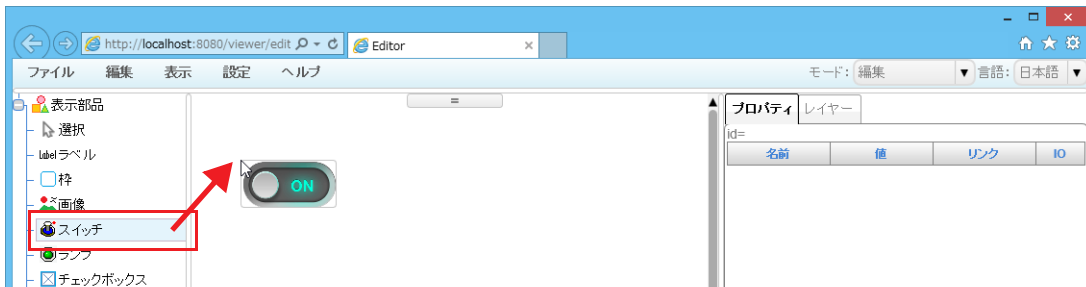


3. Web ページエディタの [モード] で “編集” をクリックします。




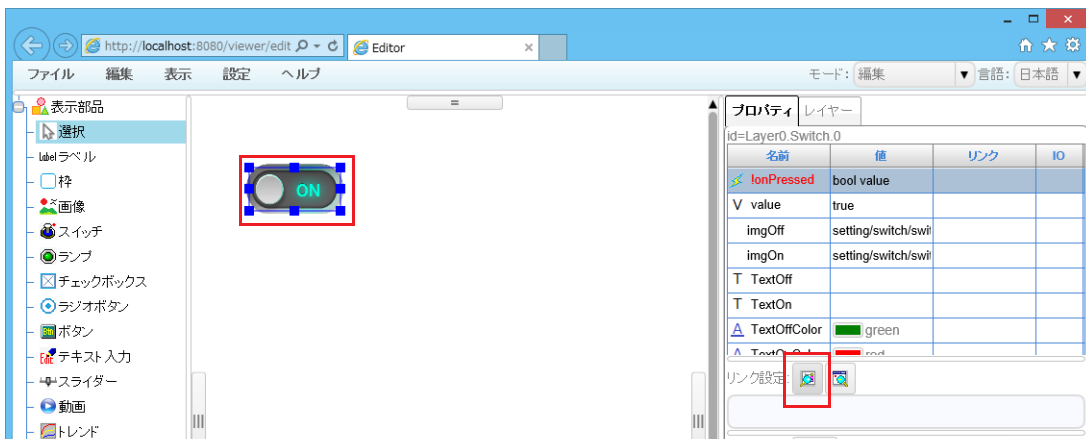
4. スイッチ 1 を配置し、A 接点 (M0000) を設定します。

[表示部品] リストから [スイッチ] を選択し、作画エリアにドラッグ & ドロップします。スイッチが規定の大きさに配置されます。



任意の大きさにスイッチを配置する場合は、[表示部品] リストから [スイッチ] をクリックし、作画エリアで十字カーソルをドラッグします。

5. 配置したスイッチをクリックし、[プロパティ] タブの [リンク設定] で  ボタンをクリックします。[デバイス ツリー] ダイアログボックスが表示されます。

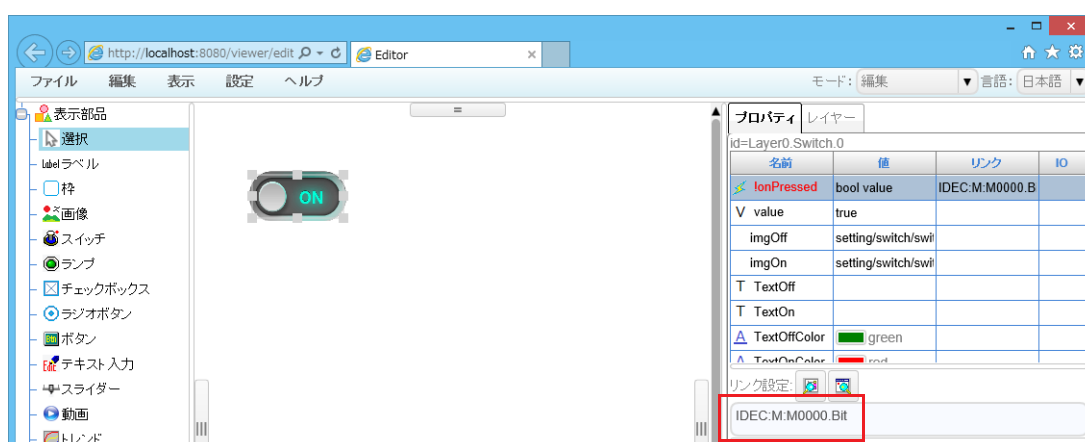


6. スイッチ 1 に設定するデバイスアドレスをクリックし、[OK] ボタンをクリックします。

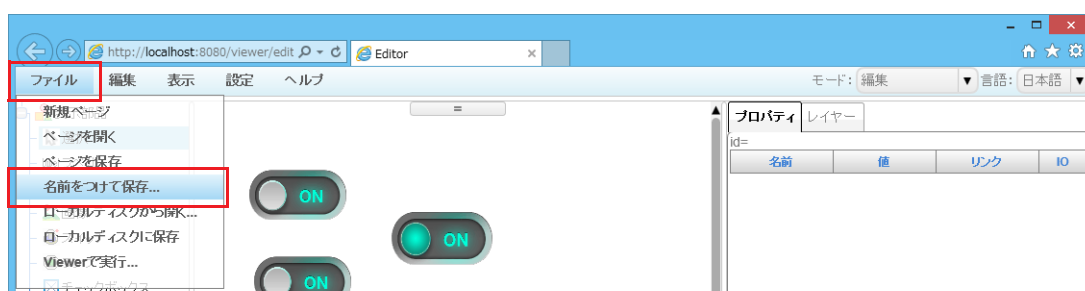


- [デバイスツリー] ダイアログボックスには、WindLDR で使用しているデバイスが表示されます。
- デバイスツリーに設定したいデバイスアドレスがない場合は、[キャンセル] ボタンをクリックし、[プロパティ] タブの [リンク設定] で、テキストボックスにデバイスアドレスを直接入力してください。

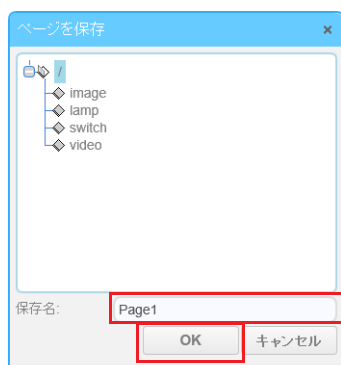
以上で、スイッチ 1 に A 接点 (M0000) が設定されます。



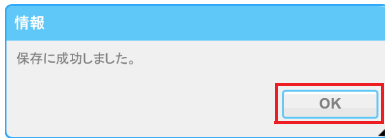
- 手順 4 ~ 6 の操作を繰り返し、スイッチ 2、ランプを設定します。
- メニューバーの [ファイル] メニューで [名前をつけて保存] をクリックします。
[ページを保存] ダイアログボックスが表示されます。



- [保存名] にファイル名を入力し、保存先のフォルダを指定して [OK] ボタンをクリックします。
確認メッセージが表示されます。



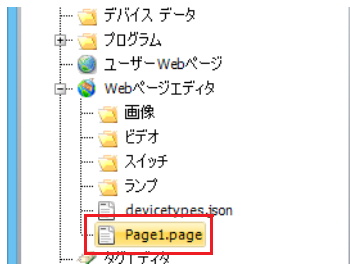
10. [OK] ボタンをクリックします。



以上で、カスタム Web ページの作成が完了します。



- Web ページエディタについては、メニューバーの [ヘルプ] メニューで [ヘルプ] をクリックして表示され Web ページエディタのヘルプを参照してください。
- 作成したカスタム Web ページは、プロジェクトウィンドウの Web ページエディタに作成されます。また、カスタム Web ページでインポートした画像ファイル、ビデオファイル、スイッチ部品の画像ファイル、ランプ部品の画像ファイルがそれぞれのフォルダ内に一覧表示されます。
作成したカスタム Web ページをダブルクリックすると、Web ページエディタが起動し、作成したカスタム Web ページが開きます。作成したカスタム Web ページを編集できます。

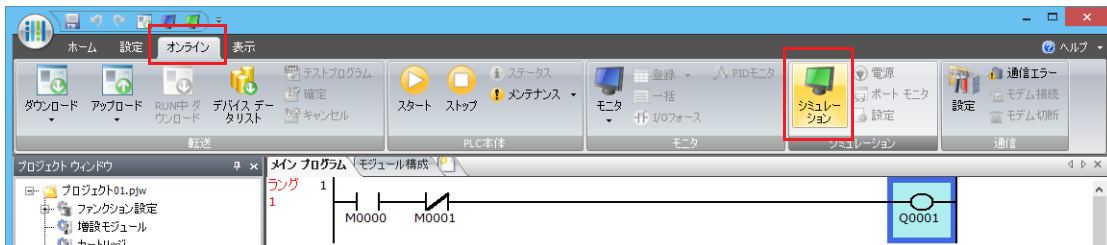


カスタム Web ページの動作確認

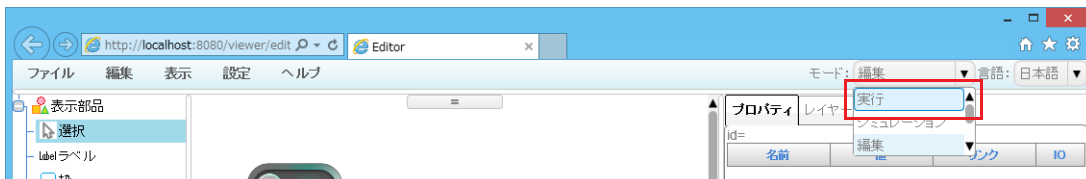
作成したカスタム Web ページの動作を Web ページエディタで実行し、動作を確認する操作手順を説明します。

●操作手順

1. WindLDR の [オンライン] タブで [シミュレーション] をクリックします。

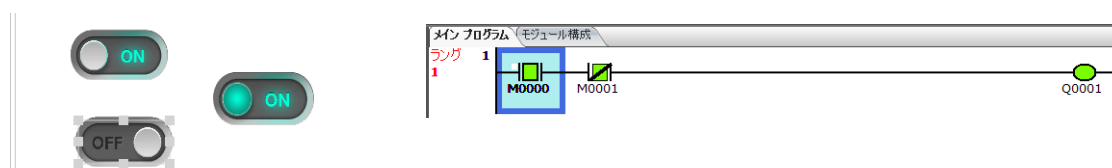


2. Web ページエディタの [モード] で“実行”をクリックします。

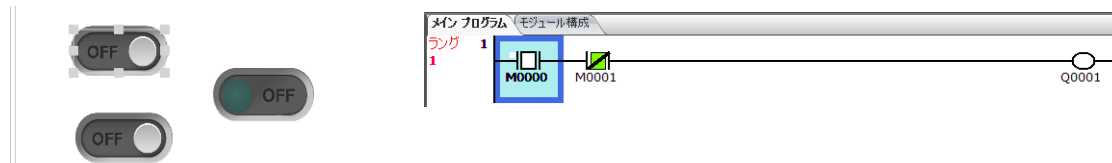


3. スイッチ1とスイッチ2をON/OFFさせながら、下記の動作を確認します。

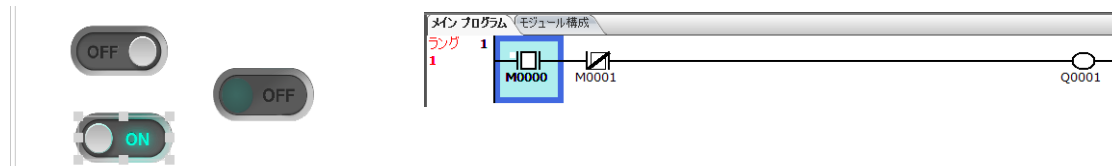
- ・スイッチ1がONかつスイッチ2がOFFの場合、ランプがONすること



- ・スイッチ1がOFFの場合、ランプがOFFすること



- ・スイッチ2がONの場合、ランプがOFFすること



Web ページエディタの状態にあわせて、WindLDR のプログラムも ON/OFF します。

以上で、動作確認が完了します。

作成したカスタム Web ページを保存したあと、WindLDR でユーザープログラムを FC6A 形にダウンロードすると、カスタム Web ページを Web ブラウザから見ることができます。

設定方法については、「Plus CPU モジュールの Web サーバー」(13-5 頁)を参照してください。

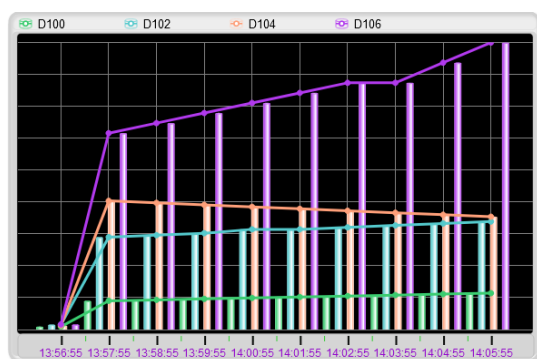
カスタム Web ページの履歴データ連携

Web ページエディタで作成したカスタム Web ページは、履歴データと連携できます。連携可能な履歴データは次のとおりです。

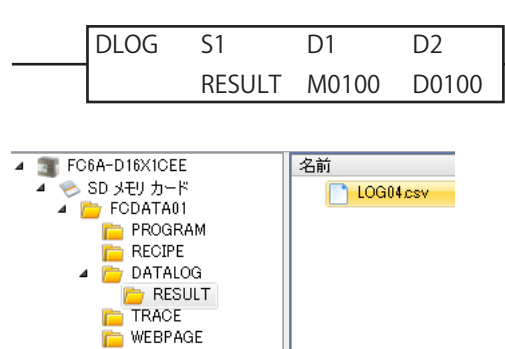
- ・DLOG (データログ) 命令で SD メモリカードに保存した履歴データ

Web ページエディタで作成したカスタム Web ページから DLOG (データログ) 命令で SD メモリカードに保存した履歴データを参照する操作手順について説明します。

Web ページで配置する部品



WindLDR で作成するプログラム





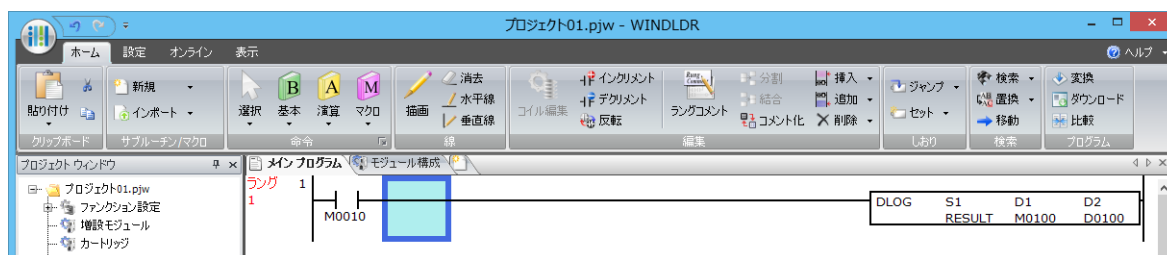
- DLOG 命令については、ラダープログラミングマニュアル「第 25 章 DLOG (データログ)」を参照してください。
- カスタム Web ページの履歴データ連携は、Plus CPU モジュールの内蔵メモリおよび Plus CPU モジュールに挿入した SD メモリカードでのみ使用できます。HMI モジュールの内蔵メモリへ保存したカスタム Web ページとの履歴データ連携はできません。
- 外部メモリの設定で履歴データ CSV ファイルフォーマットの区切り文字をセミicolon (;) に設定した場合は、カスタム Web ページとの履歴データ連携はできません。外部メモリの設定については、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 機能と設定」を参照してください。
- カスタム Web ページの履歴データ連携の部品として、Web ページエディタには、稼働状況グラフ部品、表部品、リスト部品、ドロップダウンリスト部品が用意されており、これらの部品を使うことでも履歴データ連携を設定できます。



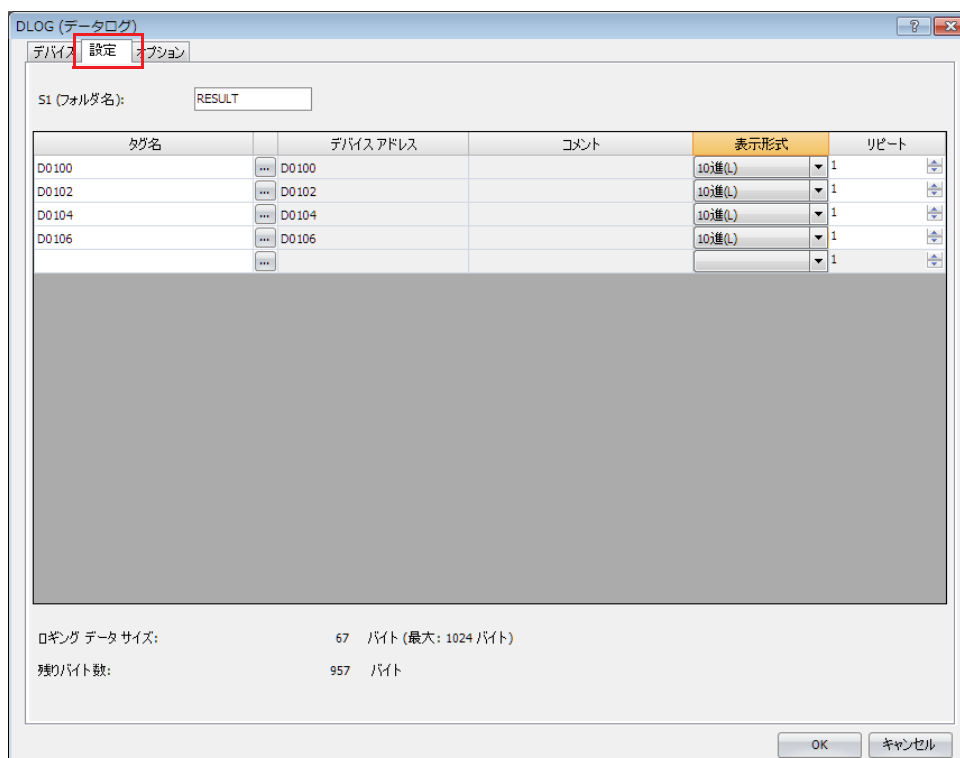
- WindLDR のシミュレーション機能では、カスタム Web ページとの履歴データ連携はできません。

●操作手順

1. DLOG 命令を挿入します。

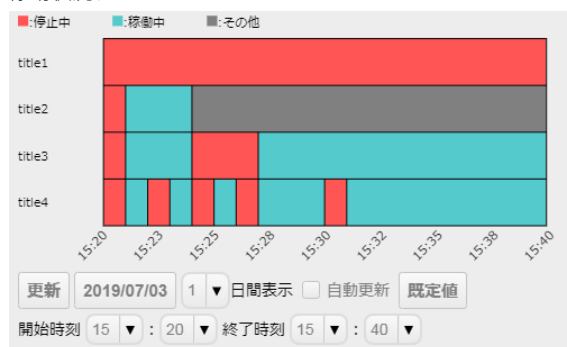


2. [設定] タブで CSV ファイルに出力するデータのデバイスアドレスを設定します。
設定したデバイスアドレスの値は、ユーザー Web ページの履歴データ連携する部品に表示されます。



ユーザー Web ページの履歴データ連携する部品で稼働状況グラフ部品を使用する場合は、DLOG 命令で CSV ファイルに出力するデータの値と、稼働状況グラフの入力値が合うようにラダープログラムを作成してください。
この例では、CSV ファイルに出力したデータの値を、0：停止中、1：稼働中という稼働状況として稼働状況グラフ部品に出力しています。

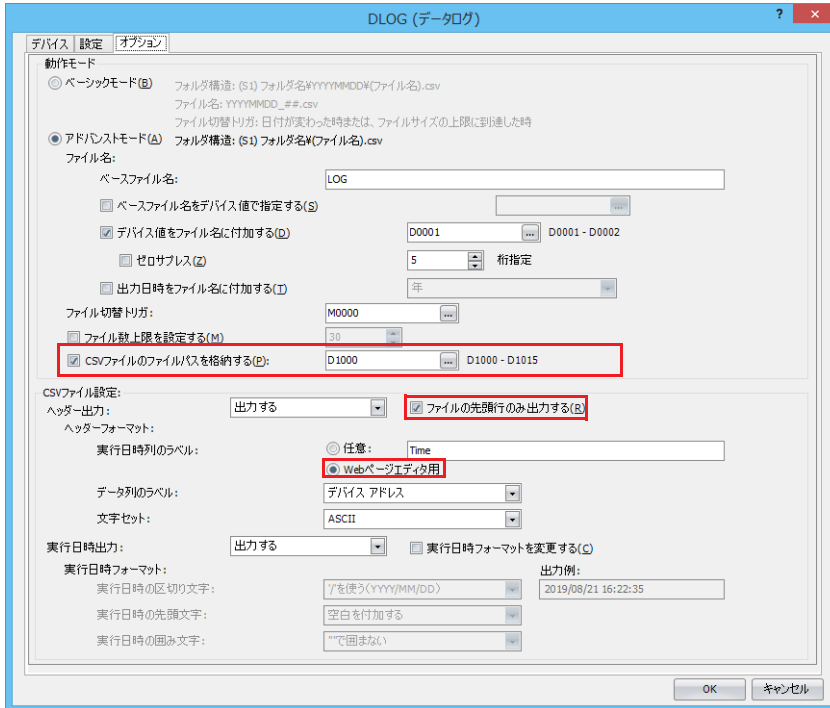
稼働状況グラフ



CSV ファイル

#_date_time_	D0100	D0102	D0104	D0106
2019/7/3 15:18:01	0	0	0	0
2019/7/3 15:19:01	0	0	0	1
2019/7/3 15:20:01	0	0	0	0
2019/7/3 15:21:01	0	1	1	1
2019/7/3 15:22:01	0	1	1	0
2019/7/3 15:23:01	0	1	1	1
2019/7/3 15:24:01	0	2	0	0
2019/7/3 15:25:01	0	3	0	1
2019/7/3 15:26:01	0	4	0	0
2019/7/3 15:27:01	0	5	1	1
2019/7/3 15:30:15	0	6	1	0
2019/7/3 15:31:15	0	7	1	1

3. [オプション] タブの [共通設定] にある [ヘッダー出力] の [ファイルの先頭行のみ出力する] チェックボックスをオンにします。
 [ヘッダーフォーマット] の [実行日時のラベル] で、“Web ページエディタ用” を選択します。
 [CSV ファイルパスを格納する] チェックボックスをオンにし、格納する先頭のデータレジスタを指定します。



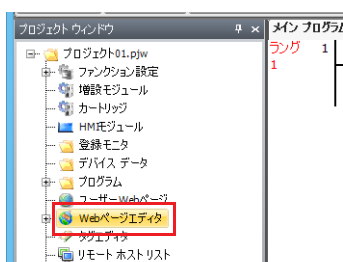
- CSV ファイルのファイルパスを格納する： チェックボックスをオンにした場合、トレンドバー部品の [リンク設定] にデバイスアドレスを設定するだけで、グラフ表示されます。
 チェックボックスをオフにした場合、トレンドバー部品のプロパティに CSV ファイルのファイルパスを直接指定しないとグラフ表示されません。
- ヘッダー出力： “出力する” に設定した場合、トレンドバー部品の X 軸にヘッダーの内容として反映されます。
 “出力しない” に設定した場合、トレンドバー部品にはグラフ表示されません。
- ファイルの先頭行のみ出力する： チェックボックスをオンにした場合、FC6A 形を STOP 状態から RUN すると、トレンドバー部品の X 軸にヘッダーの内容として反映されます。
 チェックボックスをオフにした場合、FC6A 形を STOP 状態から RUN してもトレンドバー部品にはグラフ表示されません。
- ヘッダーファイルフォーマット：
 実行日時のラベル “Web ページエディタ用” を選択した場合、トレンドバー部品の X 軸にヘッダーの内容として反映されます。
 “任意” を選択した場合、トレンドバー部品にはグラフ表示されません。

上記以外の [オプション] タブの設定に関してはトレンドバー部品に必要な設定はありません。[オプション] タブの設定については、ラダープログラミング マニュアル「第 25 章 DLOG (データログ)」を参照してください。

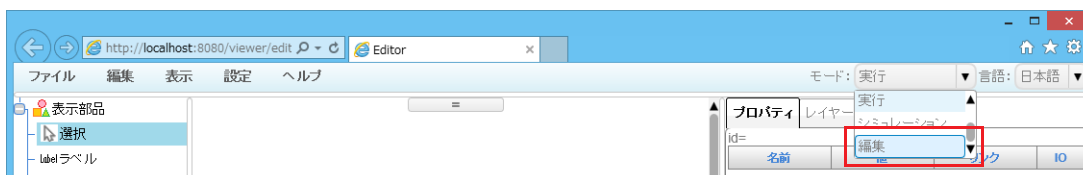


- Web ページエディタで履歴データ連携し ASCII 文字以外の文字列を表示させる場合には、[文字セット] で “Unicode(UTF-8)BOM 付き” または “Unicode(UTF-8)BOM なし” のいずれかを選択してください。
 “Unicode(UTF-8)BOM 付き” は開始 3Byte は BOM (0xEF 0xBB 0xBF) が出力されます。
 “Unicode(UTF-8)BOM なし” は特定のアプリケーションで開くと、“UTF-8” で開かれずに文字化けする可能性があります。
- 表部品、リスト部品、ドロップダウンリスト部品を使う場合は、各部品で使用可能なプロパティにより [オプション] タブの [共通設定] で設定する内容は異なります。各部品で使用可能なプロパティは、Web ページエディタメニューバーの [ヘルプ] メニューで [ヘルプ] をクリックして表示される Web ページエディタのヘルプを参照してください。

- プロジェクトウィンドウの [Web ページエディタ] をダブルクリックします。
Web ページエディタが起動します。



- Web ページエディタの [モード] で “編集” をクリックします。

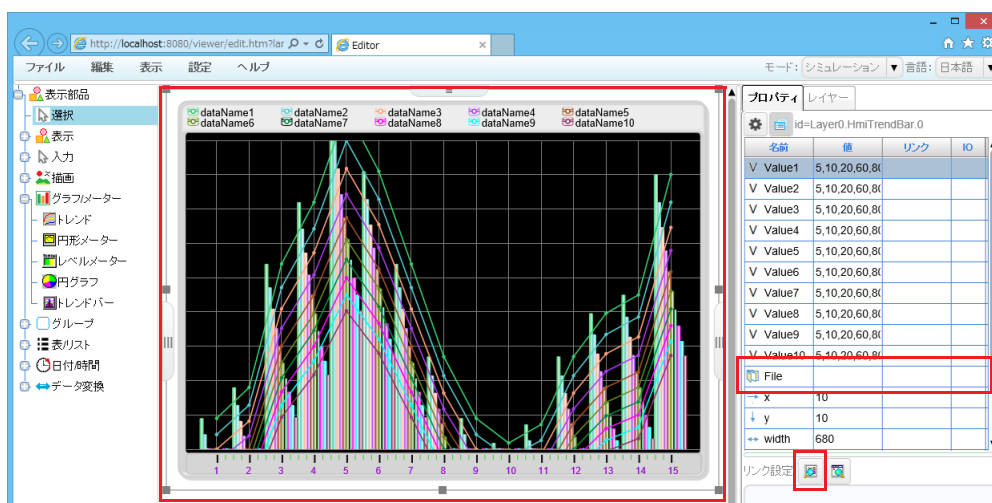


- トレンドバーを配置し、データレジスタ (D0100) を設定します。
[表示部品] リストから [トレンドバー] を選択し、作画エリアにドラッグ & ドロップします。
トレンドバーが規定の大きさで配置されます。

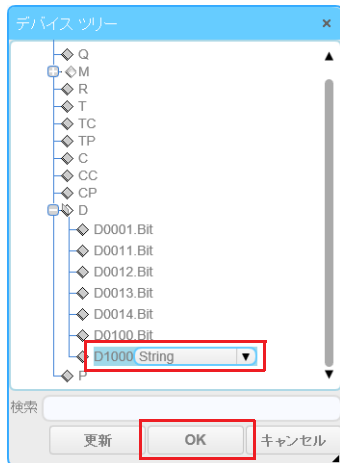


- 任意の大きさでトレンドバーを配置する場合は、[表示部品] リストから [トレンドバー] をクリックし、作画エリアで十字カーソルをドラッグします。
- 表部品、リスト部品、ドロップダウンリスト部品を使う場合は、[表]、[リスト]、[ドロップダウンリスト] を選択し、作画エリアにドラッグ & ドロップしてください。表部品、リスト部品、ドロップダウンリスト部品については、メニューバーの [ヘルプ] メニューで [ヘルプ] をクリックして表示される Web ページエディタのヘルプを参照してください。

- 配置したトレンドバーをクリックし、[プロパティ] タブの [File] をクリックしたあと、[リンク設定] で  ボタンをクリックします。
[デバイス ツリー] ダイアログボックスが表示されます。



- トレンドバーに設定するデバイスアドレスをクリックし、[OK] ボタンをクリックします。なお、[データタイプ] は“String”を選択してください。



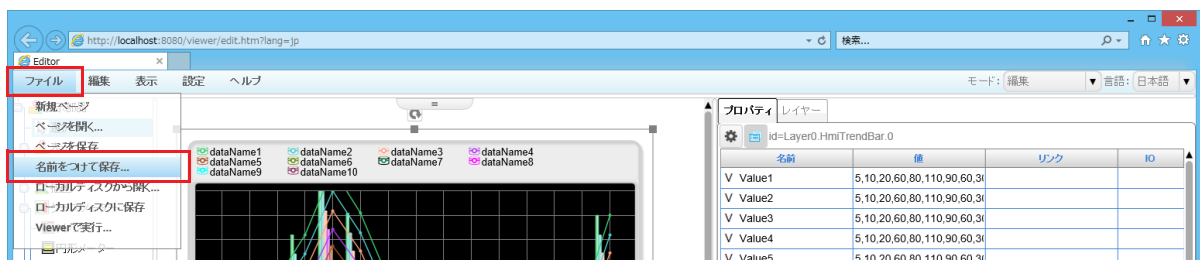
- ・ [デバイス ツリー] ダイアログボックスには、WindLDR で使用しているデバイスが表示されます。
- ・ デバイスツリーに設定したいデバイスアドレスがない場合は、[キャンセル] ボタンをクリックし、[プロパティ] タブの [リンク設定] で、変数リンク入力領域にデバイスアドレスを直接入力してください。
- ・ ファイルパスを File プロパティの値として直接指定する場合、フォルダ名の前に「<D>¥」を付加します。
ファイルパスのフォーマット：
<D>¥ フォルダ名 ¥CSV ファイル名
なお、DLOG 命令では、更新中のファイルのファイルパスが、[CSV ファイルのファイルパスを格納する] で指定したデータレジスタに格納されます。

- 配置したトレンドバーをダブルクリックします。
トレンドバーの [オプション] ダイアログボックスが表示されます。
- [X 軸] タブの [スタイル] で“CSV ファイル”を選択します。

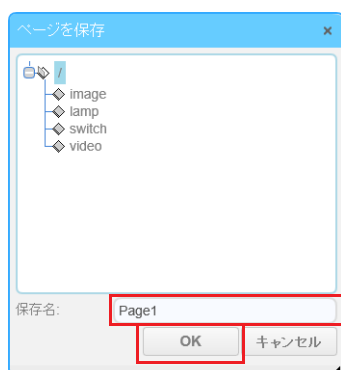


- ・ [オプション] ダイアログボックスでは、[全般] タブ、[データ] タブ、[メインパネル] タブ、[凡例] タブで、部品内の色、スケール、データ項目、単位、ツールチップ、凡例などを設定できます。

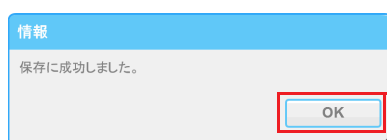
- メニューバーの [ファイル] メニューで [名前をつけて保存] をクリックします。
[ページを保存] ダイアログボックスが表示されます。



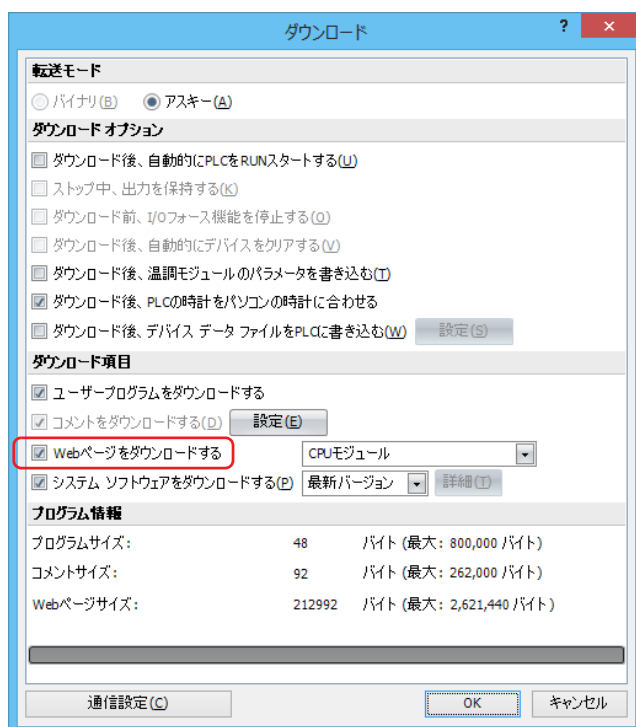
12. [保存名] にファイル名を入力し、保存先のフォルダを指定して [OK] ボタンをクリックします。
確認メッセージが表示されます。



13. [OK] ボタンをクリックします。



14. ユーザープログラムを Plus CPU モジュールにダウンロードします。
[オンライン] タブの [転送] で [ダウンロード] ボタンをクリックします。
[ダウンロード] ダイアログボックスが表示されます。
15. [Web ページをダウンロードする] チェックボックスをオンにします。



16. Web ページの保存先で "CPU モジュール" または "CPU モジュール (SD メモリカード)" を選択します。

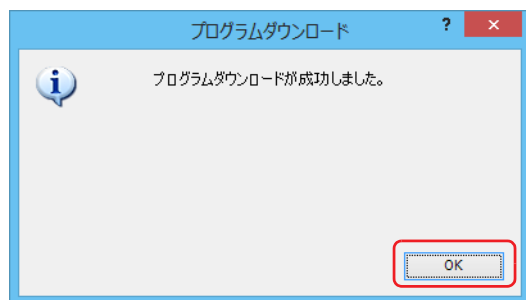


Plus CPU モジュールに SD メモリカードを挿入していない状態で "CPU モジュール (SD メモリカード)" を選択した場合は、Web ページはダウンロードされません。

17. [OK] ボタンをクリックします。

Web ページが選択した保存先にダウンロードされます。

次のメッセージが表示されたらダウンロードは成功です。[OK] ボタンをクリックします。



18. DLOG (データログ) 命令を動作させると、CSV ファイルが SD メモリカードに保存されます。保存された CSV ファイルをカスタム Web ページからグラフ表示で閲覧できます。



CSV ファイルのファイルサイズが大きい場合には、Web ページのグラフ表示をするまでに時間がかかることがあります。通信回線の状況により CSV ファイルのファイルサイズの上限を調整してください。履歴データのファイルサイズの上限設定については、FC6A 形マイクロスマート ユーザーズ マニュアル「第 5 章 機能と設定」を参照してください。

CGI

Plus CPU モジュールおよび HMI モジュールは、デバイスデータの読み出しおよび書き込みの CGI プログラムに対応しています。ここでは CGI フォーマットを説明します。

リクエストフォーマット（デバイスデータ読み出し）

デバイスデータの取得は GET メソッドを用いて以下のフォーマットで行います。

```
/system/device_read.cgi?device=DEV&address=XXXX&length=XXXX&driver=XX&net_no=XX
```

CGI のパラメータは以下のとおりです。

/system/device_read.cgi	: CGI ファイルの名称です。
device	: 読み出し対象のデバイスを文字列で指定します。"D" (データレジスタ) と "m" (内部リレー) のみ使用可能です。
address	: 読み出したいデバイス番号を、16 進数数値で指定します。
length	: address を先頭として読み出したいデータのサイズを指定します。16 進数で 0001 ~ 0040 の範囲で指定可能です。
driver	: 将来拡張用に予約してある引数です。00 を指定してください。
net_no	: 将来拡張用に予約してある引数です。00 を指定してください。

例

D2058 から 2 ワード読み出したい場合は、以下のようになります。

```
/system/device_read.cgi?device=D&address=080A&length=0002&driver=00&net_no=00
```

リプライフォーマット（デバイスデータ読み出し）

GET リクエストを受け付けた Web サーバーは、以下の形式でデータを返します。

```
Content-type: text/javascript+json; charset=utf-8
{
  "STATUS": "0",
  "DEVICE": "D",
  "ADDRESS": "080A",
  "LENGTH": "0002",
  "DRIVER": "00",
  "NET_NO": "00",
  "DATA": "C2F6_E978"
}
```

STATUS : 0 が正常応答、それ以外は異常となります。

DEVICE、ADDRESS、LENGTH、DRIVER、NET_NO : リクエストの折り返しになります。

DATA : 読み出したデータが返ります。length が 2 以上の場合は、データとデータの間には "_" が入ります。

リクエストフォーマット（デバイスデータ書き込み）

デバイスデータの書き込みは POST メソッドを用いて以下のフォーマットで行います。

```
/system/device_write.cgi?device=DEV&address=XXXX&length=XXXX&driver=XX&net_no=XX&data=XXXX_XXXX...
```

CGI のパラメータは以下のとおりです。

/system/device_write.cgi	: CGI ファイルの名称です。
device	: 書き込み対象のデバイスを文字列で指定します。"D" (データレジスタ) と "m" (内部リレー) を使用可能です。
address	: 書き込みたいデバイス番号を、16 進数数値で指定します。
length	: address を先頭として書き込みたいデータのサイズを指定します。16 進数で 0001 ~ 0040 の範囲で指定可能です。
driver	: 将来拡張用に予約してある引数です。00 を指定してください。
net_no	: 将来拡張用に予約してある引数です。00 を指定してください。
data	: データレジスタに書き込みたい値を 16 進数数値で指定します。

length が 2 以上の場合は、データとデータの間には "_" を含めます。

例

D2058 に 49910(C2F6h)、D2059 に 59768(E978h) を書き込む場合は、以下のようになります。

```
/system/device_write.cgi?device=D&address=080A&length=0002&driver=00&net_no=00&data=C2F6_E978
```

リプライフォーマット（デバイスデータ書き込み）

POST リクエストを受け付けた Web サーバーは、以下の形式でデータを返します。

```
Content-type: text/javascript+json; charset=utf-8
```

```
{
  "STATUS": "0",
  "DEVICE": "D",
  "ADDRESS": "080A",
  "LENGTH": "0002",
  "DRIVER": "00",
  "NET_NO": "00"
}
```

STATUS : 0 が正常応答、それ以外は異常となります。

DEVICE、ADDRESS、LENGTH、DRIVER、NET_NO : リクエストの折り返しになります。

SD メモリカード読み出し（ファイルのダウンロード）

Plus CPU モジュールで Web サーバー機能を使用する場合、SD メモリカード内の WEBPAGE フォルダ、または DATALOG フォルダに保存したファイルをパソコンにダウンロードすることができます。

SD メモリカード内の WEBPAGE フォルダに保存したファイルをダウンロードする場合、SD メモリカード内の Web ページにアクセスするときの URL（「アクセス先と表示する Web ページ」（13-4 頁）参照）に続けて、SD メモリカード内の WEBPAGE フォルダからの相対パスを指定します。

SD メモリカード内の DATALOG フォルダに保存したファイルをダウンロードする場合、内蔵メモリ内の Web ページにアクセスするときの URL（「アクセス先と表示する Web ページ」（13-4 頁）参照）に /viewer/<D>/ を付加し、SD メモリカード内の DATALOG フォルダからの相対パスを指定します。

例

http://192.168.1.5/sd/viewer/img/picture.png

SD メモリカード内の ¥FCDATA01¥WEBPAGE¥viewer¥img¥picture.png ファイルがパソコンにダウンロードされます。

http://192.168.1.5/viewer/%3CD%3E/RESULT/LOG04.csv

SD メモリカード内の ¥FCDATA01¥DATALOG¥RESULT¥LOG04.csv ファイルがパソコンにダウンロードされます。



ファイルのダウンロード中は、Web ページの更新が数秒間停止します。

制限事項

SSL/TLS 通信を有効にした場合、次の制限事項があります。

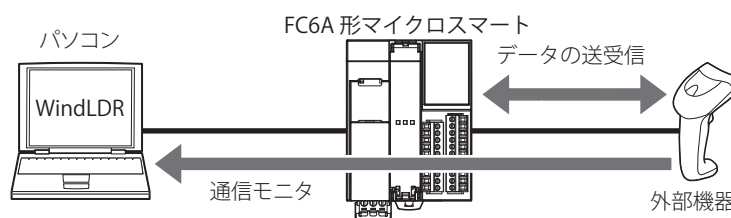
- Web ページの表示を完了するまでに時間がかかります。
- Web ページにアクセスしているときに、別の Web ブラウザなどから同じ Web サーバーへのアクセスがあった場合、Web ページの更新が数秒間停止します。
- 低速な回線やプロキシを介したネットワーク環境で Web サーバーにアクセスすると、Web サーバーの最大同時接続数の制限を超える接続が必要となり、Web ページが正常に表示されない場合があります。カスタム Web ページの場合は複数の Web ページに分割するなど、1 つの Web ページ上に配置する部品の種類を減らしてください。

第14章 通信モニタ

この章では、通信モニタ機能について説明します。

概要

通信モニタは、FC6A 形が外部機器と送受信しているデータを WindLDR の [通信モニタ] ダイアログボックスでモニタする機能です。モニタできる通信プロトコルはユーザー通信および Modbus 通信です。通信モニタは、Plus CPU モジュールのみ使用できません。



通信モニタの開始と停止

通信モニタを起動し、外部機器とのモニタを開始および停止する方法について説明します。

●操作手順

1. [オンライン] タブの [モニタ] で [通信モニタ] をクリックします。



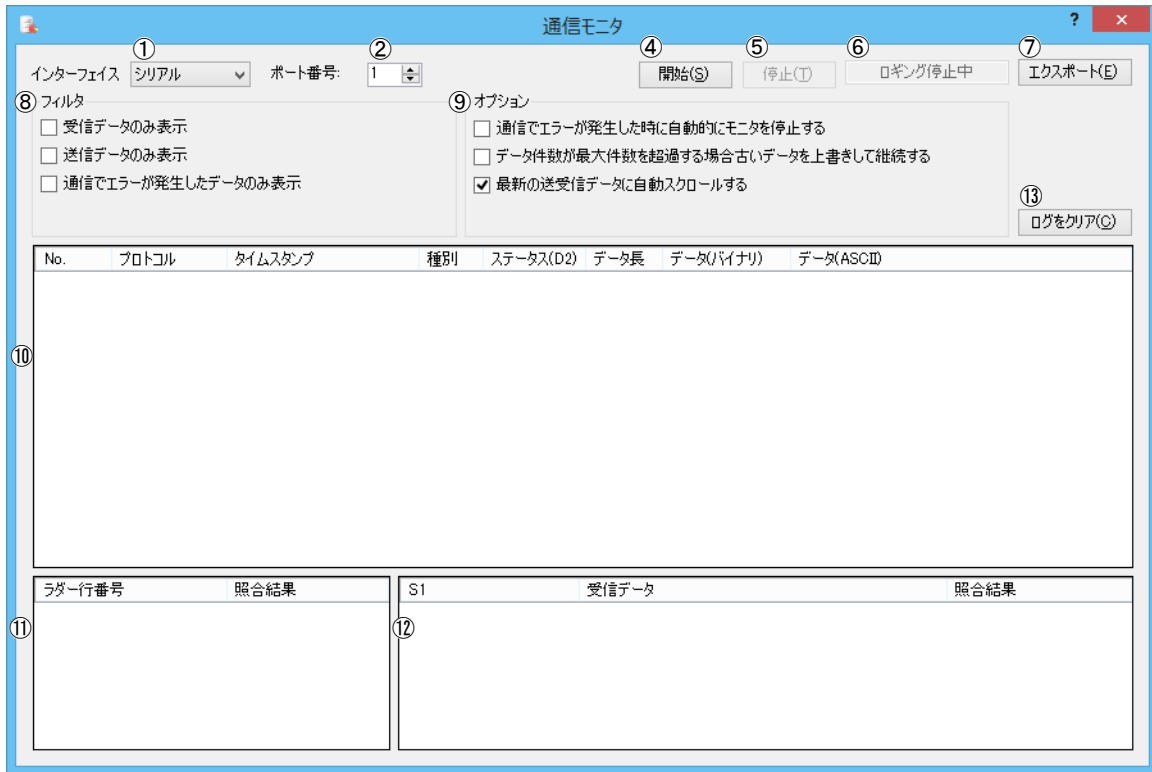
モニタする外部機器は、通信インターフェイス、ポート番号およびコネクション番号で指定します。モニタするデータは、[開始] ボタンをクリックした時点から FC6A 形が内蔵メモリに一時保存しているデータとなります。[停止] ボタンをクリックおよび [通信モニタ] ダイアログボックスを閉じることで、内蔵メモリへの保存が停止します。このため、通信モニタでモニタしていない間の送受信データは見ることができません。

モニタできるデータは最大 65535 件です。最大データ件数に到達すると、自動的にモニタを停止します。最大データ件数に到達してもモニタを継続したい場合は、[データ件数が最大件数を超過する場合古いデータを上書きして継続する] チェックボックスをオンにしてください。古いデータから順に消去してモニタを続けます。

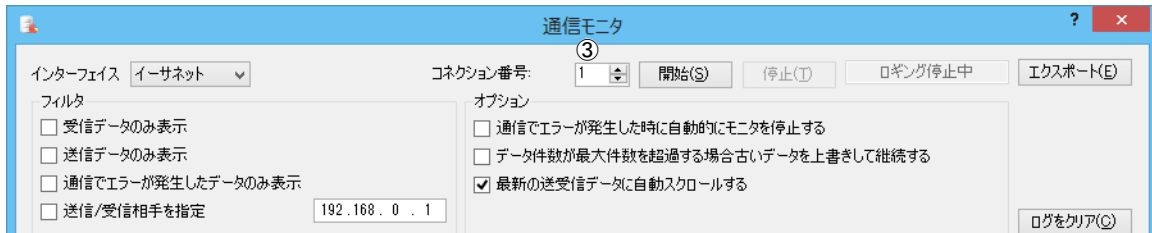
WindLDR 上の通信モニタで送受信データを表示更新する間隔よりもモニタする外部機器との通信間隔のほうが速い場合、FC6A 形が一時保存する内蔵メモリが不足する場合があります。その場合、内蔵メモリへの保存が停止します。内蔵メモリへの保存が停止したあとの通信モニタの表示は、内蔵メモリに一時保存されている送受信データをすべて表示したあとにモニタ停止状態に切り替わります。

通信モニタ

シリアル通信でのモニタ



イーサネット通信でのモニタ



① インターフェイス

モニタする通信インターフェイスを“シリアル”または“イーサネット”から選択します。

② ポート番号

モニタする通信インターフェイスのポートを設定します。インターフェイス (①) が“シリアル”の場合のみ設定できます。
ポート：1～33

③ コネクション番号

モニタするコネクションの番号を設定します。
インターフェイス (①) が“イーサネット”の場合のみ設定できます。
コネクション：1～16

④ [開始] ボタン

このボタンをクリックすると、通信モニタを開始します。WindLDRは一定周期で通信データをPLCから取得します。

⑤ [停止] ボタン

このボタンをクリックすると、通信モニタを停止します。
停止中はPLCからの通信データの蓄積を停止します。

⑥ 状態表示

通信モニタのモニタ状態が表示されます。

⑦ [エクスポート] ボタン

送受信データ表示 (⑩) で表示している通信データをエクスポートできます。
このボタンをクリックし、表示された[名前を付けて保存]ダイアログボックスでファイル名を入力し、[保存]ボタンをクリックします。

⑧フィルター

送受信データ表示 (⑩) で表示する項目を絞り込みます。表示する内容のチェックボックスをオンにします。複数のフィルターを設定できます。
[送信 / 受信相手を指定] チェックボックスをオンにした場合は、IP アドレスを入力します。

⑨オプション

次の処理でモニタを動作させる場合は、チェックボックスをオンにします。

通信でエラーが発生した時に自動的にモニタを停止する：

エラーが発生した場合にモニタを停止します。

データ件数が最大件数を超過する場合古いデータを上書きして継続する：

通信データが最大データ件数 (65535 件) に到達した場合にモニタを停止せずに、古いデータから順に消去してモニタを続けます。

最新の送受信データに自動スクロールする：

送受信データが追加された際に、自動的に最新の送受信データを表示します。

⑩送受信データ表示

FC6A 形と外部機器との通信データが表示されます。表示内容は、次のとおりです。

- ・ プロトコル
- ・ タイムスタンプ (Plus CPU モジュールのシステムバージョンが 1.01 以上)
- ・ 送受信種別
- ・ 送信 / 受信相手の IP アドレス (イーサネットの場合のみ)
- ・ ステータス
- ・ データ長
- ・ データ (バイナリおよび ASCII)



タイムスタンプ表示は、モニタ開始時点の FC6A 形のカレンダー・時計データを基点として送受信データに付加するモニタ開始からの差分時間を足すことで実現しています。通信データの通信間隔目安を表示することを目的としており、パソコンの現在時刻と一致しなかったり、サマータイムの調整などで日時がずれる可能性があります。また、タイムスタンプ表示形式はパソコンのカレンダー・時刻表示形式に合わせて表示されます。

⑪ユーザー通信命令選択

送受信データ表示 (⑩) で選択した送信および受信したデータのユーザー通信の一覧が表示されます。プロトコルが Modbus の場合には表示されません。

行番号 : WindLDR 上のユーザー通信命令 (RXD/ERXD/TXD/ETXD) の行番号が表示されます。

該当するユーザー通信命令は、PLC から取得する D2 のデバイスアドレスをもとに WindLDR 上の命令を検索して決定されます。

照合結果 : 各ユーザー通信受信命令の受信結果が表示されます。(送信の場合は表示されません。)

⑫照合結果の詳細

ユーザー通信命令選択 (⑪) で選択したユーザー通信命令のフォーマット (S1) と通信データの照合結果の詳細が表示されます。これにより、ユーザー通信受信命令のフォーマットと、実際に受信したデータとの差異を確認することができます。ただし、プロトコルが Modbus の場合には表示されません。

S1 : ユーザー通信命令選択 (⑪) で選択したユーザー通信受信命令のフォーマットが表示されます。

- ・ 定数 (文字)
- ・ 定数 (16 進数)
- ・ データレジスタ
- ・ BCC
- ・ スキップ (受信データのみ)

受信データ / 送信データ : 通信データの内容が表示されます。

照合結果 : 照合結果が表示されます。

一致 : OK

不一致 : NG



WindLDR 上のユーザー通信命令で指定したフォーマットに対して、FC6A 形が受信したデータを照合し、フォーマットとの照合結果を紐づけて表示されます。このため、WindLDR 上のユーザープログラムと FC6A 形内のユーザープログラムが一致している必要があります。ユーザープログラムが不一致の場合は、[通信モニタ] ダイアログボックスを表示する際に正確な解析結果を表示できない旨の警告が表示され、送受信データ一覧表示 (⑩) のみ表示されます。

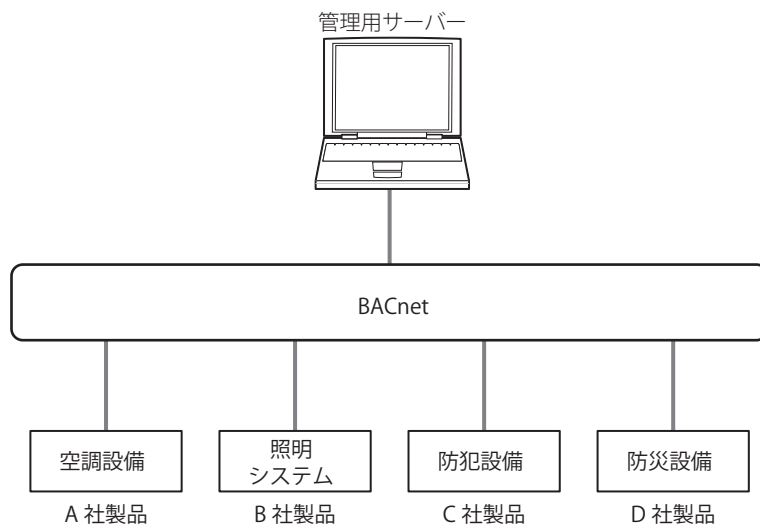
- ⑬ [ログをクリア] ボタン
表示している通信データを消去します。

第15章 BACnet/IP

この章では、BACnet 通信機能について説明します。

概要

Plus CPU モジュールは、Internet Protocol (IP) を使用した BACnet 通信 (BACnet/IP) に対応しています。BACnet とは、「Building Automation and Control Networking Protocol」の略で、ビル設備において異なるメーカーで構築されたシステムを相互接続するための標準化されたオープンプロトコルで、ビルディングネットワークのための通信規格です。従来、空調・照明・防犯・防災システムなどはビル設備・システムメーカー独自の方式で接続していましたが、BACnet に準拠することで、共通の方式で接続し、監視などを行うことができるようになります。



BACnet/IP で使用する通信ポート

対応する機種と通信ポートは、次のとおりです。

通信ポート	All-in-One CPU モジュール			CAN J1939 All-in-One CPU モジュール	Plus CPU モジュール	
	16点タイプ	24点タイプ	40点タイプ		Plus16点タイプ	Plus32点タイプ
Ethernet ポート 1	—	—	—	—	○	○
Ethernet ポート 2	—	—	—	—	—	—
HMI-Ethernet ポート	—	—	—	—	—	—

BACnet/IP はポート 1 ~ 33 では使用できません。

BACnet 仕様

形番	FC6A-D16R1CEE, FC6A-D16R4CEE FC6A-D16K1CEE, FC6A-D16K4CEE FC6A-D16P1CEE, FC6A-D16P4CEE	FC6A-D32K3CEE, FC6A-D32K4CEE FC6A-D32P3CEE, FC6A-D32P4CEE
通信ポート	Ethernetポート1	
準拠規格	ANSI/ASHRAE135-2012	
基本仕様	プロトコル	BACnet/IP
	プロファイル	B-ASC
	オブジェクトタイプ	Device Object, Analog Input Object, Analog Output Object, Analog Value Object, Binary Input Object, Binary Output Object, Binary Value Object
	オブジェクト数	最大256個*1
	BIBBs	DS-RP-B, DS-WP-B, DS-RPM-B, DS-WPM-B, DS-COV-B, DS-COVU-B, DM-DDB-B, DM-DOB-B, DM-DCC-B
	BBMD	None-BBMD Device
	Virtual Device	No
	Foreign Device	Yes
Subscribed COV 機能	受付可能リクエスト数	最大256リクエスト
Unsubscribed COV 機能	送信単位	オブジェクト毎
	送信周期	1~65535 [s]*2
Foreign Device 機能	登録方法	登録トリガデバイスによる都度登録
	Lifetime	0~65535 [s]
デバイス連携機能	<ul style="list-style-type: none"> • プロパティ、デバイス間の同期*3 • Present_Valueのデータ型変換*4 • Present_Valueの係数変換*4 	

*1 Device Object は含まれません。

*2 送信周期は全オブジェクトで一括して設定します。

*3 内部メモリに生成するオブジェクトのプロパティと、指定したデバイスの同期を行います。

*4 対応するオブジェクトは、Analog Input Object、Analog Output Object、Analog Value Object です。

BACnet について

準拠規格

Plus CPU モジュールの BACnet/IP は、次の BACnet 規格に準拠しています。

- ANSI/ASHRAE135-2012

プロファイル

Plus CPU モジュールの BACnet/IP は、次のプロファイルに対応しています。

- B-ASC

オブジェクト

BACnet/IP に対応した機器が扱う入出力値などの情報は、オブジェクトという単位で管理されます。オブジェクトは、その内容によっていくつかの種類に分けられ、これをオブジェクトタイプと呼びます。Plus CPU モジュールの BACnet/IP が対応しているオブジェクトタイプは、次のとおりです。

オブジェクトタイプ				規格
名称	略称	識別子	ANSI/ASHRAE135-2012	
基本入出力	Analog Input Object Type	AI	0	○
	Analog Output Object Type	AO	1	○
	Analog Value Object Type	AV	2	○
	Binary Input Object Type	BI	3	○
	Binary Output Object Type	BO	4	○
	Binary Value Object Type	BV	5	○
BACnetデバイスの特性	Device Object Type	DV	8	○

Plus CPU モジュールは、オブジェクトタイプごとに任意の数のオブジェクトを設定でき、その情報は、同じ BACnet/IP ネットワークの BACnet デバイスから読み書きすることができます。

各オブジェクトの詳細については、「オブジェクト」(15-33 頁)を参照してください。

プロパティ

プロパティとは、各オブジェクトの詳細な情報および属性です。オブジェクトが持つプロパティの一部は、データレジスタなどの Plus CPU モジュールのデバイスに割り付けることができ、ラダープログラムから読み出し、書き込みできます。

オブジェクトタイプごとのプロパティの一覧は「オブジェクト」(15-33 頁)を参照してください。

サービス

サービスとは、BACnet デバイス間で情報を交換するためのインターフェースです。サービスを発行するクライアント側と、サービスを実行するサーバー側とに分類されます。

Plus CPU モジュールがサポートするサービスを示します。

サービス	発行*1	実行*2
ReadProperty	—	○
ReadPropertyMultiple	—	○
WriteProperty	—	○
WritePropertyMultiple	—	○
SubscribeCOV	—	○
ConfirmedCOVNotification	○	—
UnconfirmedCOVNotification	○	—
Who-Is	—	○
I-Am	○	—
Who-Has	—	○
I-Have	○	—
DeviceCommunicationControl	—	○

*1 Plus CPU モジュールが他の BACnet デバイスへサービスを発行します。

*2 Plus CPU モジュールが他の BACnet デバイスから発行されたサービスを実行します。



サービスの詳細については、規格書 ANSI/ASHRAE 135-2012 (ISSN 1041-2336)、または一般社団法人 電気設備学会より発行の書籍 "BACnet ビルディングオートメーション用データ通信プロトコル" をご確認ください。

BIBB

BACnet 相互運用性ビルディングブロック (BIBB) は、実現する機能ごとに複数のサービスをグループ化したものです。BIBB は、機能を利用するクライアント側と、機能を提供するサーバー側とに分類されます。クライアント側の BIBB には末尾に '-A' が、サーバー側の BIBB には末尾に '-B' が付加されます。BACnet デバイスは、BIBB を用いて、自身が対応する機能を定義します。

Plus CPU モジュールの BACnet/IP は、次の BIBB に対応しています。

BIBB 区分	BIBB		サービス
Data Sharing	DS-RP-B	Data Sharing Read Property B	ReadProperty
	DS-WP-B	Data Sharing Write Property B	ReadPropertyMultiple
	DS-RPM-B	Data Sharing Read Property Multiple B	WriteProperty
	DS-WPM-B	Data Sharing Write Property Multiple B	WritePropertyMultiple
	DS-COV-B	Data Sharing COV B	SubscribeCOV ConfirmedCOVNotification UnconfirmedCOVNotification
	DS-COVU-B	Data Sharing COV Unsubscribed B	UnconfirmedCOVNotification
Device & Network Management	DM-DDB-B	Device Management Dynamic Device Binding B (Who-Is, I-Am)	Who-Is I-Am
	DM-DOB-B	Device Management Dynamic Object Binding B (Who-Has, I-Have)	Who-Has I-Have
	DM-DCC-B	Device Management Device Communication Control B	DeviceCommunicationControl



BIBB の詳細については、書籍 "BACnet ビルディングオートメーション用データ通信プロトコル" をご確認ください。

機能

Plus CPU モジュールは、1 つの BACnet デバイスとして、次の機能を提供します。

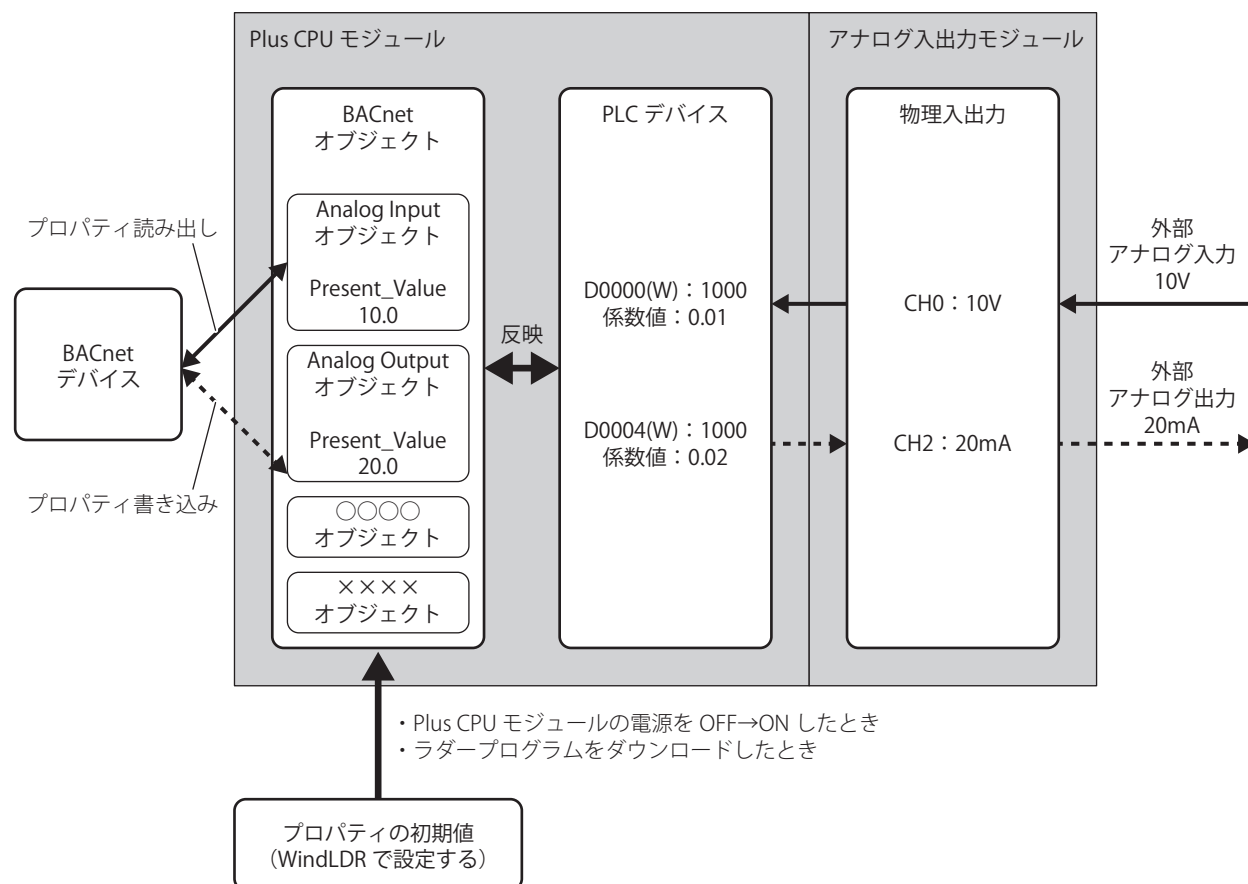
- ・オブジェクトとデバイスの連動機能
- ・プロパティの読み出し機能
- ・プロパティの書き込み機能
- ・Subscribed COV 機能
- ・Unsubscribed COV 機能
- ・Foreign Device 機能

オブジェクトとデバイスの連動機能

Plus CPU モジュール内部に生成したオブジェクトが持つプロパティの一部は、データレジスタなどのデバイスに割り付けることができ、ラダープログラムから読み出し、書き込みできます。

例えば、Plus CPU モジュールのアナログ入力値を、Analog Input オブジェクトの Present_Value として割り付けたデータレジスタ (D0000) に格納することで、BACnet デバイスがそのアナログ入力値を読み出すことができます。また、Plus CPU モジュールのアナログ出力値を格納しているデータレジスタ (D0004) を、Analog Output オブジェクトの Present_Value として割り付けることで、BACnet デバイスからそのアナログ出力値を変更することができます。

オブジェクトの種類および個数は任意に設定できます。またプロパティの初期値は WindLDR で設定します。

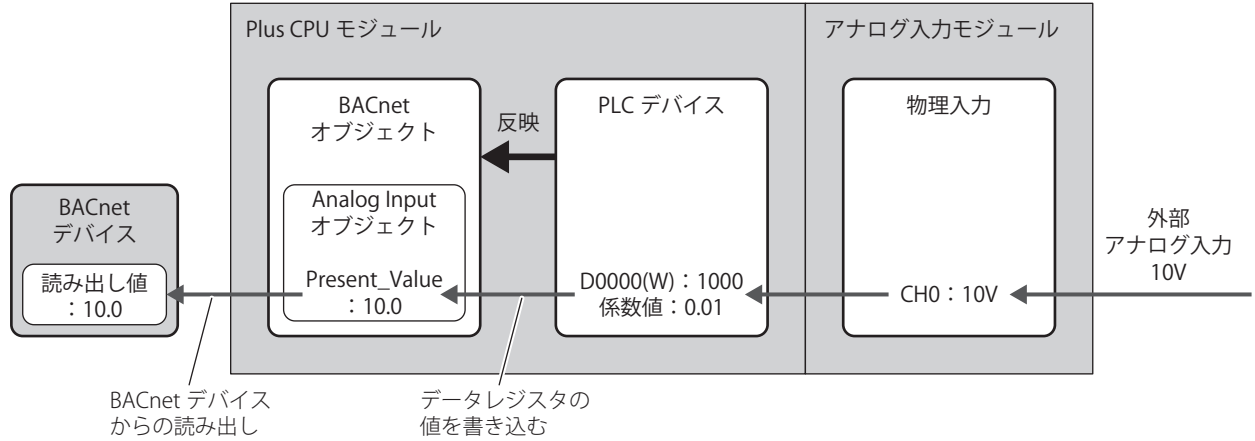


- ・プロパティとデバイスを相互に反映する処理は、ラダープログラムの実行周期とは無関係に行われます。ラダープログラムの実行中にオブジェクトに割り付けたデバイスの読み出しや書き込みが行われますので、デバイスの参照や更新を行うときに問題にならないようラダープログラムを作成してください。
- ・デバイスがオブジェクトのプロパティに反映される周期よりも短い間隔で変化した場合、その変化がプロパティに反映されることがあります。この変化をプロパティに反映したい場合は、デバイスを 1 秒間保持してください。例えば、Binary input オブジェクトの Present_Value に割り付けた M0000 が 10ms だけ ON したとき、プロパティに反映されることがあります。この変化をプロパティに反映したい場合は、M0000 が 1 秒間、ON となるようにしてください。

プロパティの読み出し機能

プロパティの読み出し機能とは、BACnet デバイスからプロパティの読み出し要求があった際に、Plus CPU モジュールがプロパティの値を返す機能です。本機能は、DS-RP-B (Data Sharing Read Property B)、DS-RPM-B (Data Sharing Read Property Multiple B) を使用することで実現します。

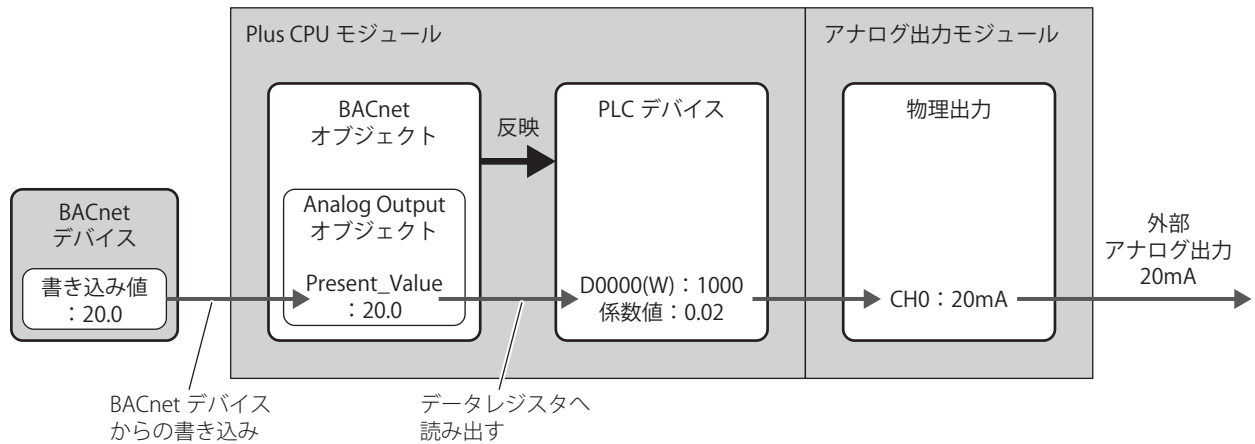
下記に、Analog Input オブジェクトに対応付けた Plus CPU モジュールのアナログ入力値を、BACnet デバイスが読み出すイメージを示します。Plus CPU モジュールのアナログ入力値を読み出すイメージを示します。



プロパティの書き込み機能

プロパティの書き込み機能とは、BACnet デバイスからプロパティの書き込み要求があった際に、Plus CPU モジュールがオブジェクトのプロパティへ値を書き込む機能です。本機能は、DS-WP-B (Data Shring Write Property B)、DS-WPM-B (Data Sharing Write Property Multiple B) を使用することで実現します。

下記に、Analog Output オブジェクトに対応付けた Plus CPU モジュールのアナログ出力値を、BACnet デバイスが変更するイメージを示します。



Subscribed COV (COV) 機能

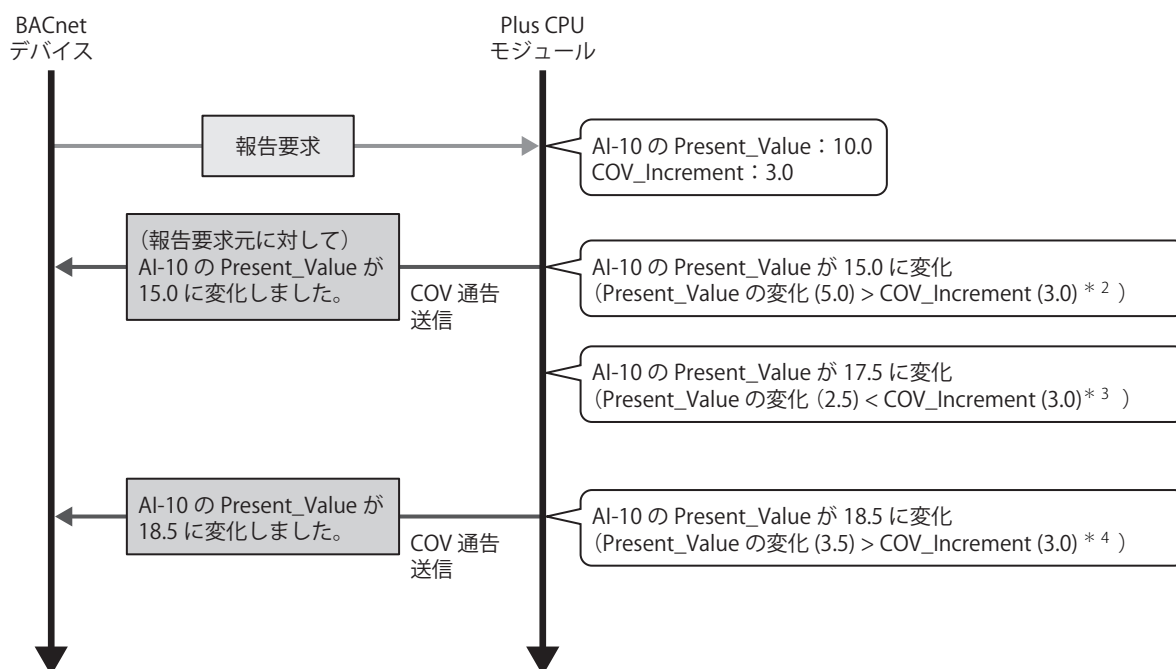
COV (Change Of Value) 機能とは、Plus CPU モジュールが BACnet デバイスから報告要求を受けたオブジェクトを監視し、Present_Value または Status_Flags の値が変化したタイミングで、BACnet デバイスへ変化を通知する機能です。

COV 機能に対応したオブジェクトのプロパティと変化を通知するタイミングは、次のとおりです。

オブジェクト	プロパティ	変化を通知するタイミング
Analog Input Analog Output Analog Value	Present_Value Status_Flags	次の①または②のとき変化を通知します。 ① Present_ValueがCOV_Incrementに設定された値以上に変化した場合（起点は前回COV通告を送信したタイミングから）*1 ② Status_Flagsのいずれかのビットが変化した場合
Binary Input Binary Output Binary Value	Present_Value Status_Flags	次の①または②のとき変化を通知します。 ① Present_Valueが変化した場合 ② Status_Flagsのいずれかのビットが変化した場合

*1 例えば、AI-10 (Analog Input オブジェクト、インスタンス番号 10) が下表の状態では BACnet デバイスから報告要求を受信した場合のイメージを示します。

AI-10 のプロパティ	値
Present_Value	10.0
COV_Increment	3.0



*2 この変化は、変化を通知するタイミング①に相当します。

Present_Value の変化 = 現在値 - 報告要求を受信したときの値 = 15.0 - 10.0 = 5.0
COV_Increment = 3.0 であるため、Present_Value の変化 \geq COV_Increment を満たします。

*3 Present_Value の変化 = 現在値 - 前回 COV 通告を送信したときの値 = 17.5 - 15.0 = 2.5
Present_Value の変化 < COV_Increment となり、変化を通知するタイミング①の条件を満たしません。
そのため、COV 通告を送信しません。

*4 この変化は、変化を通知するタイミング①に相当します。

Present_Value の変化 = 現在値 - 前回 COV 通告を送信したときの値 = 18.5 - 15.0 = 3.5
COV_Increment = 3.0 であるため、Present_Value の変化 \geq COV_Increment を満たします。



- オブジェクトとデバイスの同期周期よりも速い値の変化は、通知できない場合があります。
- BACnet サービス発行のための送信キューは 30 個です。送信キューの上限を超える送信要求が同時にあった場合、送信されないことがあります。

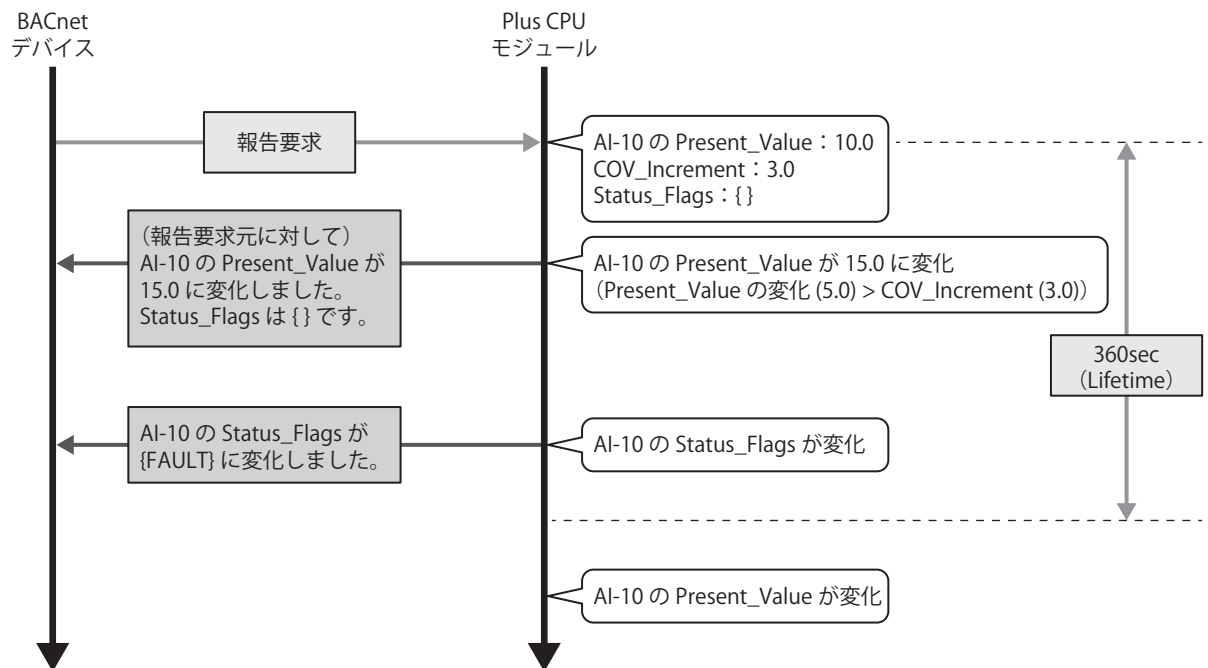
Plus CPU モジュールは、BACnet デバイスが送信した報告要求 (Subscribe COV サービス) を受信すると、報告要求に含まれるパラメータにしたがって、報告要求元の BACnet デバイスに Confirmed COV Notification サービス / Unconfirmed COV Notification サービスを送信することで COV 機能を実現します。

報告要求に含まれる主なパラメータは、次のとおりです。

主なパラメータ	説明
Monitored Object Identifier	COV機能を有効にするオブジェクトの種類とIDです。
Issue Confirmed Notifications	Plus CPUモジュールからBACnetデバイスに送信するメッセージの確認あり/なしを選択します。 ・確認あり (ConfirmedCOVNotification) ・確認なし (UnconfirmedCOVNotification)
Lifetime	COV機能を有効にする時間 (1sec単位) です。 0または省略した場合、無期限でCOV機能が有効になります。

BACnet デバイスが送信した次のような報告要求を、Plus CPU モジュールが受信した場合のイメージを示します。

報告要求の主なパラメータ	内容
Monitored Object Identifier	Analog Inputオブジェクト (インスタンス番号: 10)
Issue Confirmed Notifications	確認あり (ConfirmedCOVNotification)
Lifetime	360sec



- 登録できる COV の最大数は 256 個です。
- 報告要求の各パラメータの詳細については、書籍 "BACnet ビルディングオートメーション用データ通信プロトコル" をご確認ください。

Unsubscribed COV (COVU) 機能

COVU (Change Of Value Unsubscribed) 機能とは、Plus CPU モジュールが、自発的に、特定のオブジェクトの Present_Value や Status_Flags が変化したことを同じネットワークに接続されているすべての BACnet デバイスに通知する機能です。

COVU 機能に対応したオブジェクトのプロパティと変化を通知するタイミングは、次のとおりです。

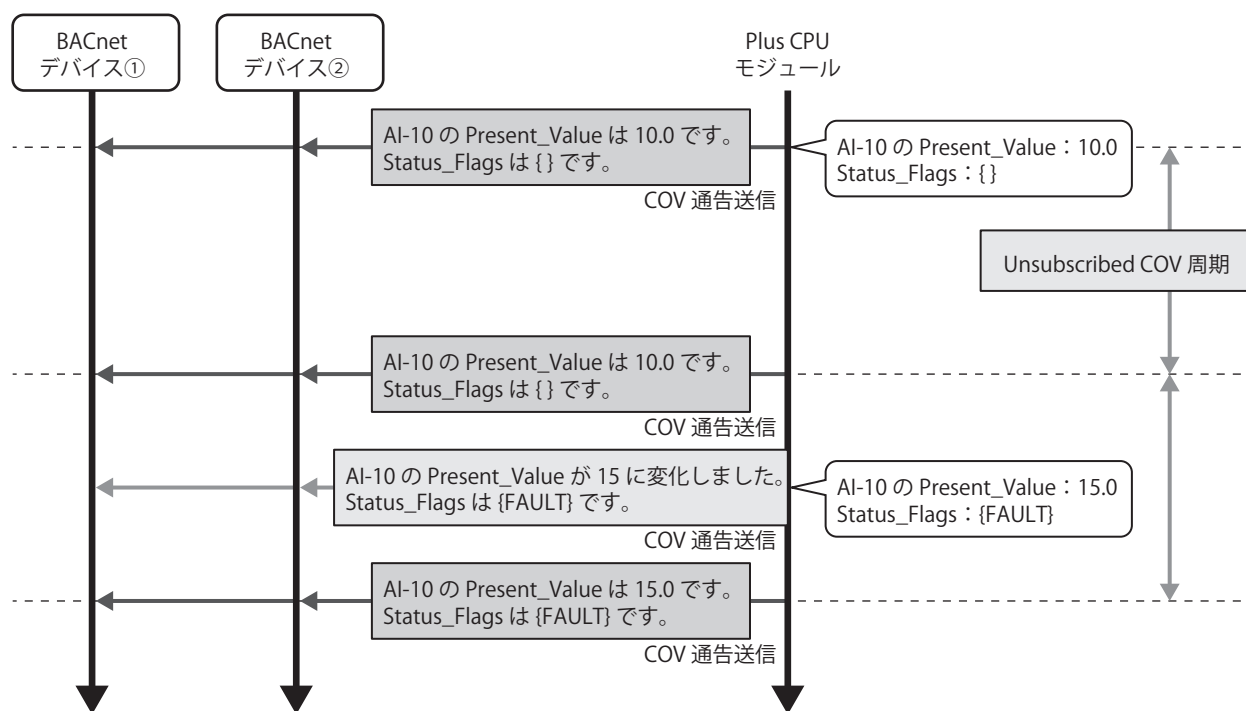
オブジェクト	プロパティ	変化を通知するタイミング
Analog Input Analog Output Analog Value	Present_Value Status_Flags	Unsubscribed COV周期*1の間隔で状態を通知します。 また、次の①または②のとき変化を通知します。 ① Present_ValueがCOV_Incrementに設定された値以上に変化した場合（起点は前回COV通告を送信したタイミングから） ② Status_Flagsのいずれかのビットが変化した場合
Binary Input Binary Output Binary Value	Present_Value Status_Flags	次の①または②のとき変化を通知します。 ① Present_Valueが変化した場合 ② Status_Flagsのいずれかのビットが変化した場合

*1 「基本設定」(15-14 頁) を参照してください。

Plus CPU モジュールが、BACnet デバイスに Unconfirmed COV Notification サービスを送信することで COVU 機能を実現します。

例えば、AI-10 (Analog Input オブジェクト、インスタンス番号 10) が下表の状態では BACnet/IP を開始した場合、設定した周期 (Unsubscribed COV 周期) でプロパティの状態を通知します。

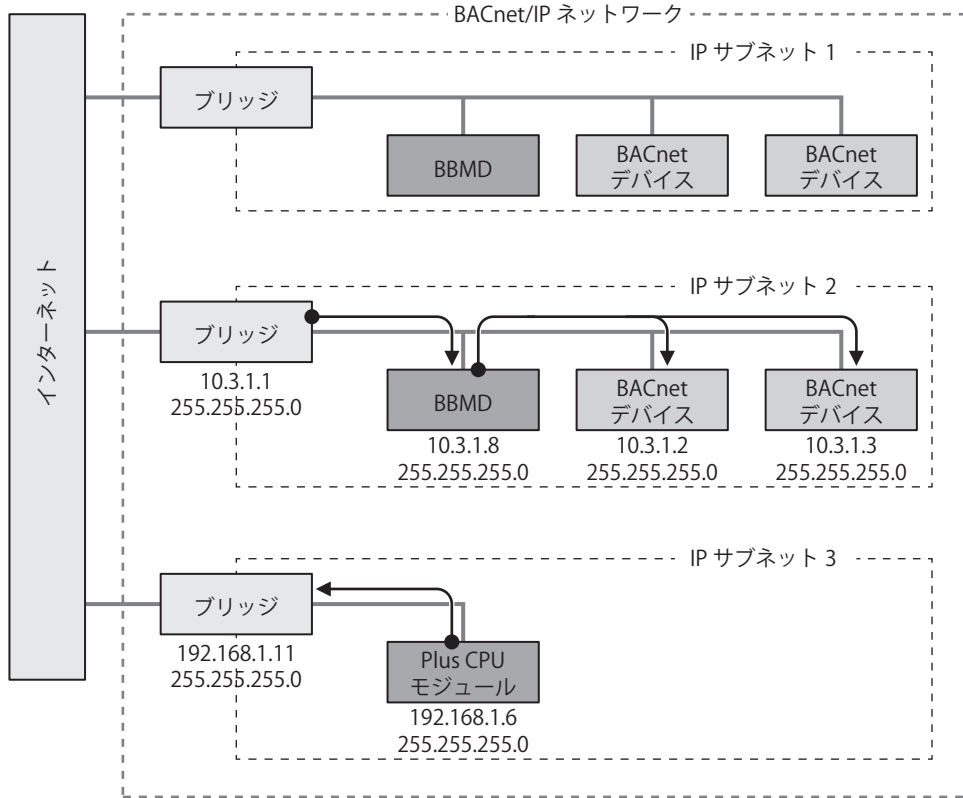
AI-10 のプロパティ	値
Present_Value	10.0
COV_Increment	3.0
Status_Flags	{}



- COVU 機能を使用すると、プロパティの値の変化に関係なく、定期的に BACnet デバイスにプロパティの値を送信することができます。
- COVU 機能はオブジェクトごとに設定できます。
- Unsubscribed COV 周期は、共通で 1 つ設定できます。オブジェクトごとに異なる周期を設定できません。
- Unsubscribed COV 周期に 0 を設定した場合、定期的に通知しません。対象のプロパティに変化があったときだけ通知します。
- Unsubscribed COV 周期をデバイスで変更した場合、次回発行後に適応されます。
- オブジェクトとデバイスの同期周期よりも速い値の変化は、通知できない場合があります。
- BACnet サービス発行のための送信キューは 30 個です。送信キューの上限を超える送信要求が同時にあった場合、送信されないことがあります。

Foreign Device 機能

BACnet/IP ネットワークを複数の IP サブネット構成する時、各 IP サブネットには 1 台の BBMD (BACnet Broadcast Management Device) を設置します。BBMD とは、BACnet デバイスのブロードキャスト通信を異なる IP サブネットへ転送するための装置です。BACnet デバイスは、BBMD を介して、異なる IP サブネットの BACnet デバイスとブロードキャスト通信を行います。Foreign Device 機能とは、Plus CPU モジュールの IP サブネットに BBMD が無い場合でも、異なる IP サブネットの BACnet デバイスとブロードキャスト通信するための機能です。Plus CPU モジュールを、指定した BBMD へ Foreign Device として登録することで、BACnet/IP ネットワーク内の BACnet デバイスとブロードキャスト通信ができます。




BACnet/IP の動作

BACnet/IP で使用する特殊デバイス

BACnet/IP で使用する特殊データレジスタおよび特殊内部リレーについて示します。

特殊データレジスタ	内容	
D8782	BACnet運転状態	BACnet通信の運転状態を格納します。 0：停止中 1：運転準備中 2：運転中 3：エラー停止中
D8783	BACnetエラー情報	BACnet通信で発生したエラー情報を格納します。 最後に発生したエラー情報が格納されます。 0：正常 1：デバイスIDが不正 2：IPアドレスが不正 3：BBMDのIPアドレスが不正 4：BBMD登録失敗

特殊内部リレー	内容	
M8450	BACnet通信許可	BACnet通信を許可/禁止します。 OFF：BACnet通信禁止 ON：BACnet通信許可

 BACnet 運転状態（D8782）がエラー停止中のときに BACnet 通信を再開する場合、一旦 M8450 を OFF したあと、BACnet 運転状態（D8782）が停止中になるのを待ってから、M8450 を ON してください。


RUN および STOP 中の動作

BACnet/IP の有効 / 無効は、RUN/STOP の状態に関係なく、BACnet 通信許可（M8450）にしたがって切り替わります。

デバイスを割り付けたプロパティは、そのデバイスを参照して動作します。

BACnet/IP が有効である場合、STOP 中もプロパティまたはプロパティと関連付けたデバイスの値が変更されると、変更後の値で動作します。

RUN/STOP 状態	BACnet/IP 有効 / 無効	デバイスを割り付けたプロパティとそのデバイスの値
STOP	有効	連動する
	無効	連動しない

 STOP 中の出力は、BACnet/IP の有効 / 無効に関係なく、M8025（STOP 中出力保持）の状態にしたがって出力します。
M8025（STOP 中出力保持）については、「第 2 章 M8025：STOP 中出力保持」（2-12 頁）を参照してください。

STOP → RUN 時の動作

デバイスを割り付けたプロパティは、そのデバイスを参照して動作します。

Plus CPU モジュールの電源 OFF/ON や STOP/RUN の切り替え、ラダープログラムによるデバイス値の変更などでプロパティと関連付けたデバイスの値が変更されると、変更後の値で動作します。そのため、予期せぬ動作をすることがありますのでご注意ください。

例えば、Out_Of_Service に M0000 を割り付けている場合、デバイスのメモリバックアップの設定によって次のように Out_Of_Service の値が異なります。

① STOP → RUN で内部リレーを "すべてクリア" する場合

RUN/STOP 状態	BACnet/IP 有効 / 無効	Out_Of_Service	M000
STOP	有効	TRUE	ON
↓			
RUN	有効	FALSE	OFF

RUN 開始時に内部リレーの状態がクリアされるため、Out_Of_Service= FALSE となります。

② STOP → RUN で内部リレーを "すべてキープ" する場合

RUN/STOP 状態	BACnet/IP 有効 / 無効	Out_Of_Service	M000
STOP	有効	TRUE	ON
↓			
RUN	有効	TRUE	ON

内部リレーの状態が保持されるため、Out_Of_Service= TRUE のままです



- ・プロパティに割り付けるデバイスは、STOP → RUN でその状態を保持するよう設定してください。
- ・内部リレーおよびデータレジスタのメモリバックアップのデフォルト設定は次のとおりです。

デバイス	デフォルト設定
内部リレー	すべてクリア
データレジスタ	すべてキープ

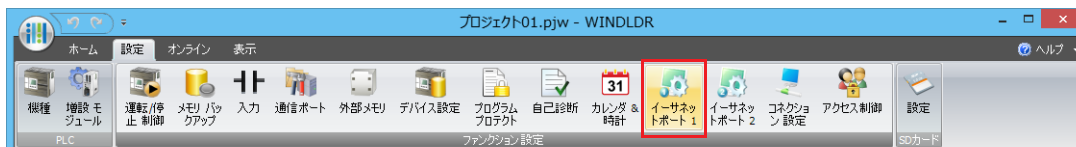
デバイスのメモリバックアップの設定については、FC6A 形マイクロスマート ユーザーズマニュアル「第 5 章 機能と設定」を参照してください。

BACnet/IP の設定

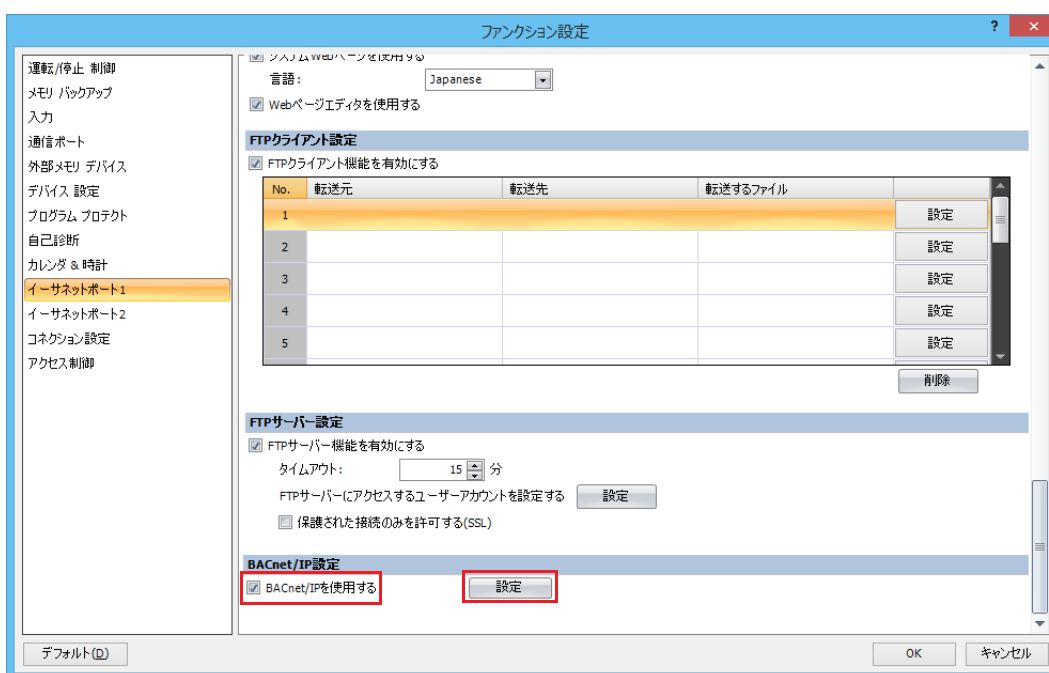
BACnet/IP を使った通信を行うための操作手順と各項目の詳細について説明します。

●操作手順

1. [設定] タブの [ファンクション設定] で [イーサネットポート 1] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。



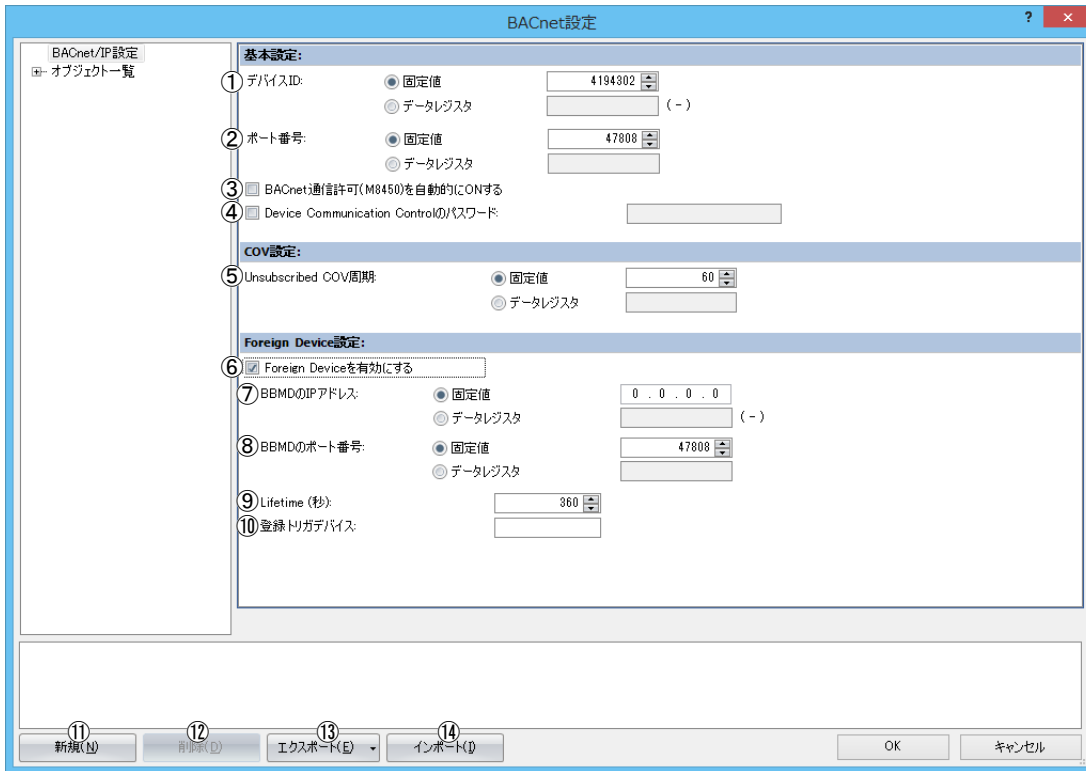
2. BACnet/IP 設定で [BACnet/IP を使用する] チェックボックスをオンにし、[設定] ボタンをクリックします。
[BACnet 設定] ダイアログボックスが表示されます。



基本設定

BACnet/IP を行うための基本的な設定について説明します。

画面は [BACnet 設定] ダイアログボックスで、[BACnet/IP 設定] を選択したときの例です。



基本設定

①デバイス ID

BACnet/IP ネットワークで各 BACnet デバイスを識別するために割り付けられた ID です。
Plus CPU モジュールのデバイス ID は、次の 2 種類から設定できます。

設定方法	内容
固定値	デバイスIDに固定値を設定します。 固定値を1~4194302の範囲で設定します。デフォルト値は4194302です。
データレジスタ	デバイスIDをデータレジスタで設定します。 指定したデータレジスタを先頭に連続して2ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。 デバイスIDは1~4194302の範囲で設定します。

②ポート番号

BACnet/IP を行うポート番号を設定します。
ポート番号は次の 2 種類から設定できます。

設定方法	内容
固定値	ポート番号に固定値を設定します。 固定値を0~65535の範囲で設定します。デフォルト値は47808 (BAC0h) です。
データレジスタ	ポート番号をデータレジスタで設定します。 ポート番号は0~65535の範囲で設定します。

③ BACnet 通信許可 (M8450) を自動的に ON する

運転 (RUN) 開始時、1 スキャン目の END 処理で、BACnet 通信許可 (M8450) を ON するかどうかを設定します。オンラインエディット中は、ユーザープログラムのダウンロードが完了した最初のスキャンの END 処理で BACnet 通信許可 (M8450) が ON します。

下表は各状態について示します。

M8450 の状態	RUN/STOP 状態	BACnet/IP	Presnet_Value *1 とデバイスの値	Status_Flags *2 の Overridden フラグ
ON	STOP	有効	連動する	TRUE
	RUN			FALSE
OFF	STOP	無効	連動しない	—
	RUN			—

*1 Present_Value については、「オブジェクト」(15-33 頁)、「Present_Value」(15-41 頁)を参照してください。

*2 Status_Flags については、「オブジェクト」(15-33 頁)、「Status_Flags」(15-54 頁)を参照してください。



BACnet 通信許可 (M8450) は次の場合に OFF します。

- STOP 中、RUN 中に問わずユーザープログラムをダウンロードしたあと
- Plus CPU モジュールの電源を OFF したとき

④ Device Communication Control のパスワード

BACnet デバイスから Device Communication Control サービスを受信したときに要求するパスワードを設定します。半角英数字で最大 20 文字まで設定できます。Plus CPU モジュールは、Device Communication Control サービスを受け付けると、指定された期間はサービスの発行と応答を行いません。

COV 設定

⑤ Unsubscribed COV 周期

COVU 機能で定期的にプロパティの値を通知する周期を 1 sec 単位で設定します。

設定できる範囲は 0 ~ 65535 です。0 を設定した場合は、COVU 機能が停止し、対象のプロパティに変化があったときだけ通知します。

設定方法	内容
固定値	周期に固定値を設定します。デフォルト値は60secです。
データレジスタ	周期をデータレジスタで設定します。RUN中に周期を変更できます。



- Unsubscribed COV 周期は Plus CPU モジュールに 1 つ設定できますが、オブジェクトごとに異なる周期は設定できません。
- COVU 機能を使用するかどうかはオブジェクトごとに選択できます。また、データレジスタを指定し、その値を変更することで、COVU 機能の有効 / 無効を切り替えることができます。
- Unsubscribed COV 周期にデータレジスタを設定し、データレジスタの値を変更して Unsubscribed COV 周期を変更する場合、変更後の Unsubscribed COV 周期は次回発行後に適応されます。

Foreign Device 設定

⑥ Foreign Device を有効にする

Foreign Device 機能を使用するかどうかを設定します。

⑦ BBMD の IP アドレス

⑧ BBMD のポート番号

Plus CPU モジュールの IP サブネットに BBMD が無い状態で BACnet/IP ネットワークへ参加する場合、他の IP サブネットにある BBMD の IP アドレスとポート番号を設定します。Plus CPU モジュールは、自身を Foreign Device として設定した BBMD へ登録します。

BBMD の IP アドレスは、次の 2 種類から設定できます。

設定方法	内容
固定値	BBMDのIPアドレスに固定値を設定します。
データレジスタ	BBMDのIPアドレスをデータレジスタで設定します。 指定したデータレジスタを先頭に連続して4ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。 例えば、BBMDのIPアドレスに192.168.2.5を設定する場合、次のように書き込みます。 先頭データレジスタ+0 = 192 先頭データレジスタ+1 = 168 先頭データレジスタ+2 = 2 先頭データレジスタ+3 = 5

BBMD のポート番号は、次の 2 種類から設定できます。

設定方法	内容
固定値	BBMDのポート番号に固定値を設定します。 固定値を0~65535の範囲で設定します。デフォルト値は47808 (BAC0h) です。
データレジスタ	BBMDのポート番号をデータレジスタで設定します。 BBMDのポート番号は0~65535の範囲で設定します。

⑨ Lifetime (秒)

Plus CPU モジュールを BBMD に Foreign Device として登録する期間を 1sec 単位で設定します。

登録してから (Lifetime + 30) sec を超えると、登録した内容が BBMD から削除されます。

設定できる範囲は 0 ~ 65535 です。デフォルト値は 360sec です。

設定方法	内容
固定値	Lifetimeに固定値を設定します。デフォルト値は360secです。

⑩登録トリガデバイス

Plus CPU モジュールを、上記⑦で設定した BBMD へ Foreign Device として登録するためのデバイスです。登録トリガデバイスを OFF から ON にすると、Plus CPU モジュールを BBMD へ登録します。継続的に BBMD へ登録する場合は、(Lifetime + 30) sec が経過する前に登録トリガデバイスを用いて再登録してください。

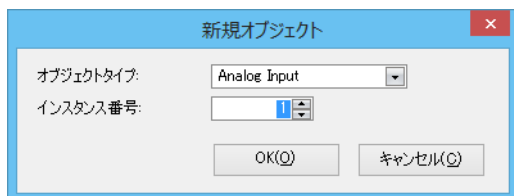
登録トリガデバイスには内部リレーを設定します。

オブジェクトの追加と削除

⑪ [新規] ボタン

新しいオブジェクトを追加します。

新規ボタンをクリックすると、[新規オブジェクト] ダイアログボックスが表示されます。オブジェクトタイプとインスタンス番号を設定します。インスタンス番号は、0～4194302の範囲で同じオブジェクトタイプの他のオブジェクトと重複しない番号に設定してください。



⑫ [削除] ボタン

オブジェクト一覧で選択しているノード以下のオブジェクトを削除します。

⑬ [エクスポート] ボタン

次のオブジェクトの設定内容を CSV 形式でエクスポートします。

Analog Input Object、Analog Output Object、Analog Value Object、Binary Input Object、Binary Output Object、Binary Value Object

⑭ [インポート] ボタン

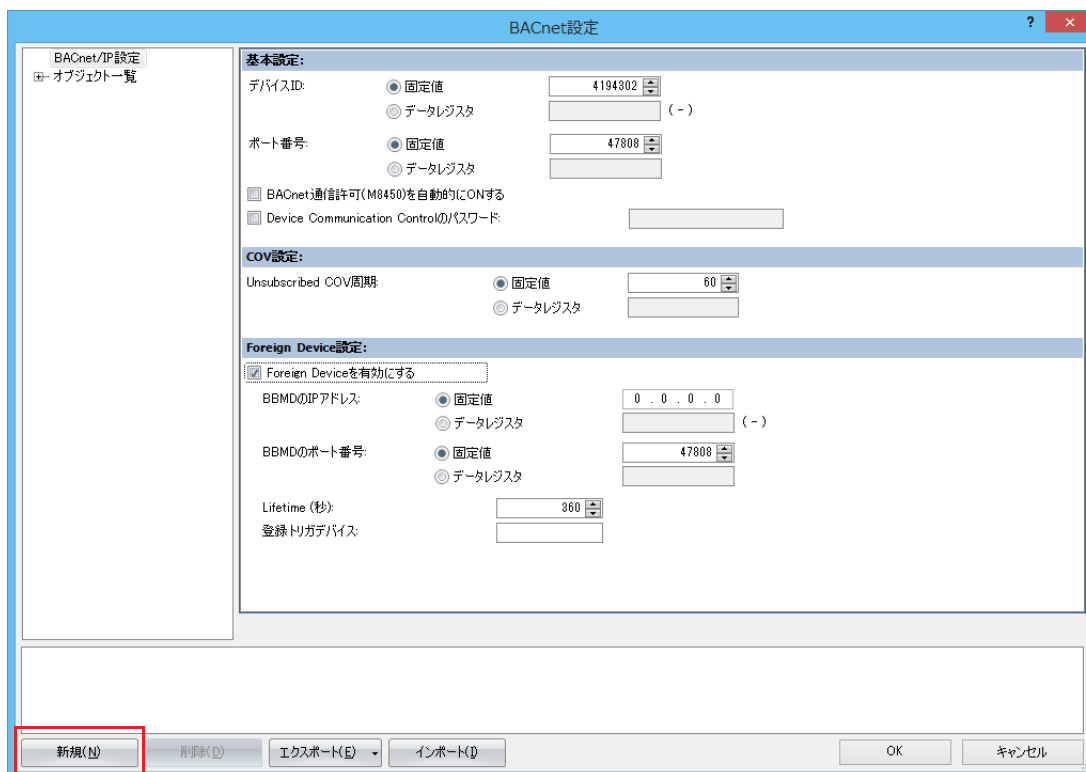
エクスポートした CSV ファイルをインポートし、オブジェクトを自動生成します。

CSV ファイルのフォーマットが不正である場合、およびオブジェクトが最大数に達している場合は、インポートできません。

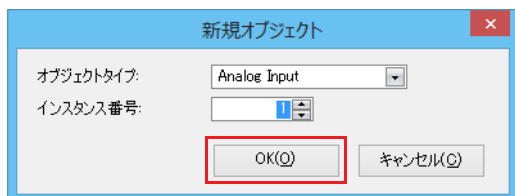
新規オブジェクトの追加

新しいオブジェクトを追加するための手順について説明します。

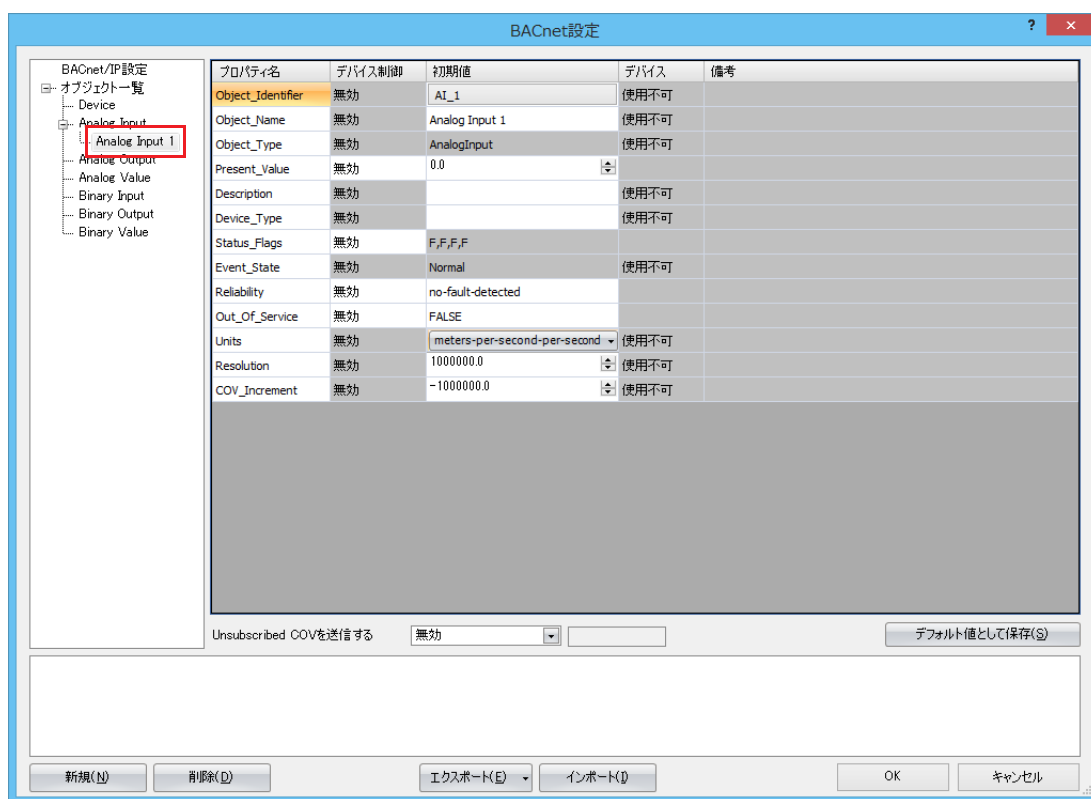
1. [新規] ボタンをクリックします。
[新規オブジェクト] ダイアログボックスが表示されます。



2. 登録するオブジェクトのオブジェクトタイプを選択し、インスタンス番号を割り付けます。
[OK] ボタンをクリックします。



- オブジェクト一覧で登録したオブジェクト ID をクリックします。
プロパティが表示されます。



- 各プロパティを設定し、[OK] ボタンをクリックします。
詳細は「オブジェクト設定」(15-20 頁) を参照してください。

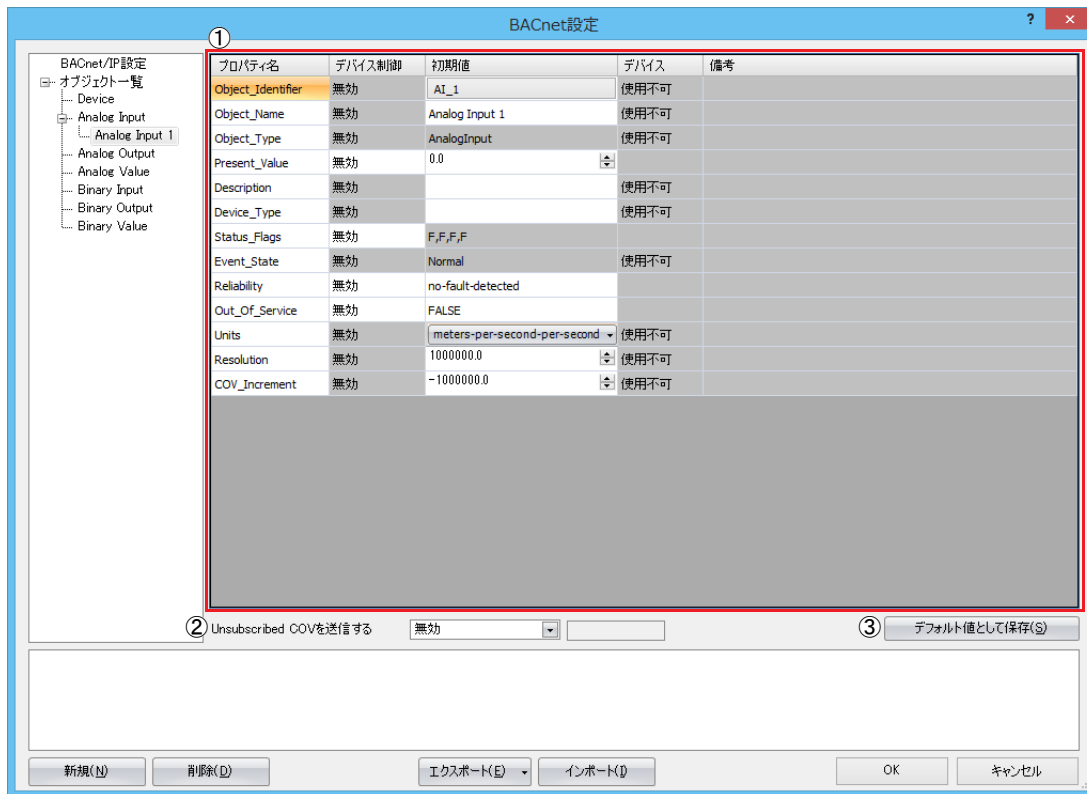


登録できるオブジェクトの最大数は 256 個です。

オブジェクト設定

オブジェクトが持つプロパティの設定について説明します。

画面は [BACnet 設定] ダイアログボックスで、登録済みのオブジェクト ID を選択したときの例です。



① プロパティ設定

オブジェクト一覧で選択しているオブジェクト ID のプロパティを表示します。
一部のプロパティは編集できます。

② Unsubscribed COV を送信する

Unsubscribed COV を送信する / しないを設定します。

次の 3 種類から設定できます。

- 有効
- 無効
- デバイスで制御する

設定	内容
有効	BACnet通信許可 (M8450) がONのとき、COV機能が常に有効となり、Unconfirmed COV Notificationサービスを送信します。
無効	COV機能が常に無効となり、Unconfirmed COV Notificationサービスを送信しません。
デバイスで制御する	内部リレーを設定し、ON/OFFすることで有効/無効を切り替えることができます。 特殊内部リレーは設定できません。

③ [デフォルト値として保存] ボタン

各プロパティの設定値をオブジェクトのデフォルト値として保存します。

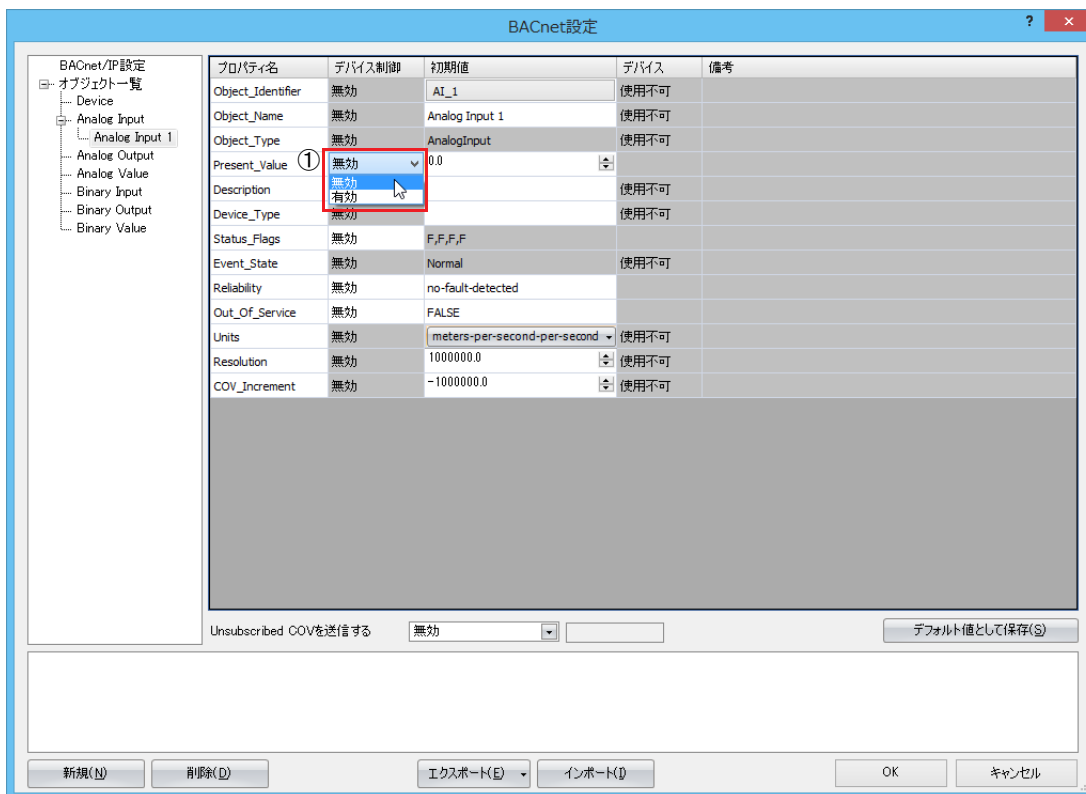
Present_Value 設定

Present_Value の設定方法について説明します。

Analog Input オブジェクト

Analog Input オブジェクトの Present_Value へ固定値やデバイスを設定します。

画面はオブジェクト一覧で Analog Input オブジェクト ID を選択したときの例です。



① デバイス制御

Present_Value に固定値を設定するか、またはデータレジスタを割り付けるかを選択します。

デバイス制御	内容
無効	Present_Valueに固定値を設定する
有効	Present_Valueにデータレジスタを割り付ける

■ Present_Value に固定値を設定する場合

Present_Value のデバイス制御 (①) に "無効" を設定し、初期値に固定値を入力します。



■ Present_Value にデバイスを割り付ける場合

Present_Value のデバイス制御 (①) に "有効" を設定し、[Present_Value 設定] ダイアログボックスで各パラメータを設定します。



① Present_Value デバイス

Present_Value を格納するデバイスを設定します。
Present_Value に割り付けることができるデバイスはデータレジスタのみです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	—	—	—	—	—	—	○*1	—

*1 特殊データレジスタは使用できません。

"変換タイプ" (②) にしたがって、指定したデータレジスタを先頭に連続して1ワード分または2ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。

Present_Value デバイス	格納先	
	ワード インテジャ	ダブルワード ロング フロート (無変換)
書き込み用Present_Value	先頭番号+0	先頭番号+0、先頭番号+1

②変換タイプ

Present_Value (フロート型) を割り付けるデバイスのデータタイプを設定します。詳細については、「Analog Input オブジェクト」(15-45 頁) の Present_Value を参照してください。

③係数値

データレジスタに格納された値に係数値を乗算した値を Present_Value とします。
Present_Value = データレジスタに格納された値 × 係数値

例えば、下表のように Present_Value は計算されます。

Plus CPU モジュール			係数値	BACnet 設定デバイス
データレジスタ	データタイプ	値		Present_Value
D0000	ワード	1000	0.01	10.0
D0000、D0001	フロート	2.5	0.5	1.25



係数値の乗算はフロート型で行います。変換順は、次のとおりです。

- ① データレジスタの値をフロート型に変換する。
- ② ①の変換結果に係数値を乗算する。

Analog Output オブジェクト

Analog Output オブジェクトの Present_Value へデバイスを設定します。

画面はオブジェクト一覧で Analog Output オブジェクト ID を選択したときの例です。

BACnet設定

プロパティ名	デバイス制御	初期値	デバイス	備考
Object_Identifier	無効	AO_1		使用不可
Object_Name	無効	Analog Output 1		使用不可
Object_Type	無効	AnalogOutput		使用不可
Present_Value ①	有効	0.0		
Description	無効			使用不可
Device_Type	無効			使用不可
Status_Flags	無効	F,F,F,F		使用不可
Event_State	無効	Normal		使用不可
Reliability	無効	no-fault-detected		
Out_Of_Service	無効	FALSE		
Units	無効	meters-per-second-per-second		使用不可
Resolution	無効	1000000.0		使用不可
Relinquish_Default	無効	0.0		使用不可
COV_Increment	無効	-1000000.0		使用不可
Priority_Array	無効			使用不可

Unsubscribed COVを送信する: 無効

デフォルト値として保存(S)

新規(N) 削除(D) エクスポート(E) インポート(I) OK キャンセル

① デバイス制御

Present_Value のデバイス制御は "有効" 固定です。



Analog Output オブジェクトの Present_Value に固定値を設定することはできません。

■ Present_Value にデバイスを割り付ける場合

[Present_Value 設定] ダイアログボックスで各パラメータを設定します。



① Present_Value デバイス

Present_Value を格納するデバイスを設定します。
Present_Value に割り付けることができるデバイスはデータレジスタのみです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	—	—	—	—	—	—	○*1	—

*1 特殊データレジスタは使用できません。

"変換タイプ" (②) にしたがって、指定したデータレジスタを先頭に連続して1ワード分または2ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。

Present_Value デバイス	格納先	
	ワード インテジャ	ダブルワード ロング フロート (無変換)
読み出し用Present_Value	先頭番号+0	先頭番号+0、先頭番号+1

②変換タイプ

Present_Value (フロート型) を割り付けるデバイスのデータタイプを設定します。詳細については、「Analog Output オブジェクト」(15-46 頁) の Present_Value を参照してください。

③係数値

Present_Value に 1/ 係数値を乗算した値がデータレジスタに格納されます。

$$\text{データレジスタの値} = \text{Present_Value} \times (1/\text{係数値})$$

例えば、次のように係数値およびデータタイプを設定した場合、Present_Value は次のようにデータレジスタに格納されます。

BACnet デバイス	係数値	Plus CPU モジュール		
		データレジスタ	データタイプ	値
Present_Value				
10.0	0.01	D0000	ワード	1000
1.25	0.5	D0000、D0001	フロート	2.5



係数値の乗算はフロート型で行います。変換順は、次のとおりです。

- ① Present_Value に (1/ 係数値) を乗算する。
- ② ①の結果 (フロート型) をデータタイプ変換する。

Analog Value オブジェクト

Analog Value オブジェクトの Present_Value へデバイスを設定します。

画面はオブジェクト一覧で Analog Value オブジェクト ID を選択したときの例です。

プロパティ名	デバイス制御	初期値	デバイス	備考
Object_Identifier	無効	AV_1	使用不可	
Object_Name	無効	Analog Value 1	使用不可	
Object_Type	無効	AnalogValue	使用不可	
Present_Value ①	有効	0.0		
Description	無効		使用不可	
Status_Flags	無効	F,F,F,F		
Event_State	無効	Normal	使用不可	
Out_Of_Service	無効	FALSE		
Units	無効	meters-per-second-per-second	使用不可	
Relinquish_Default	無効	0.0	使用不可	
COV_Increment	無効	-1000000.0	使用不可	
Resolution	無効	1000000.0	使用不可	
Priority_Array	無効		使用不可	

① デバイス制御

Present_Value のデバイス制御は "有効" 固定です。



Analog Value オブジェクトの Present_Value に固定値を設定することはできません。

■ Present_Value にデバイスを割り付ける場合

[Present_Value 設定] ダイアログボックスで各パラメータを設定します。

Present_Value 設定

- Present_Value デバイス:
- 読み出し用 Present_Value:
- 書き込み用 Present_Value:
- 優先度およびトリガの先頭デバイス:
- 優先度用デバイス:
- 書き込みトリガ用デバイス:
- 変換タイプ: フロート(無変換)
- 係数値(C): 0.00

OK キャンセル

① Present_Value デバイス

Present_Value の読み出し用デバイスと、Present_Value への書き込み用デバイスを設定します。
Present_Value に割り付けることができるデバイスはデータレジスタのみです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	—	—	—	—	—	—	○*1	—

*1 特殊データレジスタは使用できません。

設定したデバイスと "変換タイプ" (③) にしたがって、読み出し用 Present_Value と書き込み用 Present_Value のデバイスが自動的に割り付きます。指定したデータレジスタを先頭に連続して 2 ワード分または 4 ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。

Present_Value デバイス	格納先	
	ワード インテジャ	ダブルワード ロング フロート (無変換)
読み出し用Present_Value	先頭番号+0	先頭番号+0、先頭番号+1
書き込み用Present_Value	先頭番号+1	先頭番号+2、先頭番号+3

②優先度およびトリガの先頭デバイス

Present_Value へデバイスの値を書き込む場合に使用します。詳細については、「Analog Value オブジェクト」(15-47 頁) の Present_Value を参照してください。

データレジスタを設定すると、"優先度用デバイス" および "書き込みトリガ用デバイス" が自動的に割り付きます。指定したデータレジスタを先頭に連続して 2 ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。

格納先	内容	
先頭番号+0	優先度用デバイス	
	ビット15	NULL書き込みフラグ*1 OFF：書き込み用Present_Valueデバイスの値を書き込む ON：NULLを書き込む
	ビット14~5	無効
	ビット4~0	優先度*2
先頭番号+1	書き込みトリガ用デバイス	

*1 NULL 書き込みフラグが ON のとき、書き込みトリガ用デバイスを OFF から ON にすると、優先度が示すインデックス番号の Priority_Array へ NULL を書き込みます。

*2 優先度は 1 ~ 16 を指定してください。範囲外を指定した場合は、書き込みトリガ用デバイスを操作しても書き込まれません。

③変換タイプ

Present_Value (フロート型) を割り付けるデバイスのデータタイプを設定します。詳細については、「Analog Value オブジェクト」(15-47 頁) の Present_Value を参照してください。

④係数値

Present_Value に 1/ 係数値を乗算した値が読み出し用 Present_Value として割り付けたデータレジスタに格納されます。

$$\text{読み出し用 Present_Value} = \text{Present_Value} \times (1/\text{係数値})$$

また、書き込みトリガ用デバイスを OFF から ON にすると、書き込み用 Present_Value として割り付けたデータレジスタの値に係数値を乗算した値が Present_Value として設定されます。

$$\text{Present_Value} = \text{書き込み用 Present_Value} \times \text{係数値}$$

Present_Value	係数値	変換タイプ	読み出し用 / 書き込み用 Present_Value として割り付けたデータレジスタ	
			アドレス	値
10.0	0.01	ワード	D0000	1000
1.25	0.5	フロート	D0002、D0003	2.5

係数値の乗算はフロート型で行います。変換順は、次のとおりです。

データレジスタ → Present_Value

- ① データレジスタの値をフロート型に変換する。
- ② ①の変換結果に係数値を乗算する。

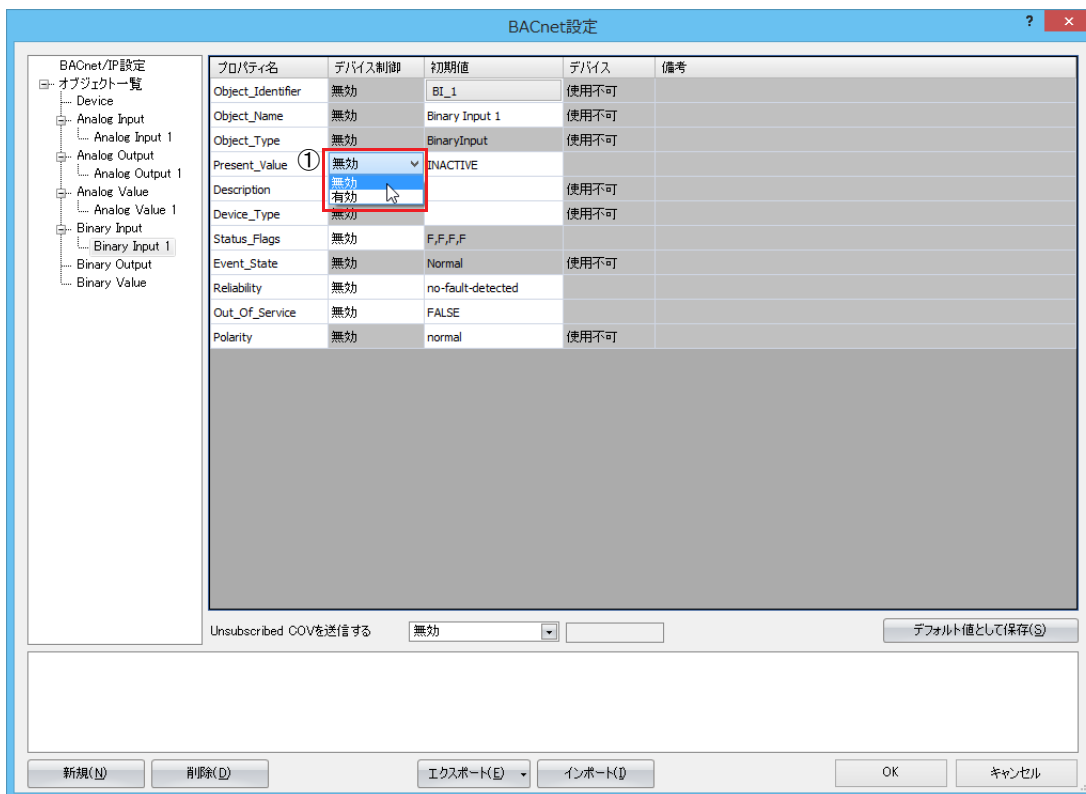
Present_Value → データレジスタ

- ① Present_Value に (1/ 係数値) を乗算する。
- ② ①の結果 (フロート型) をデータタイプ変換する。

Binary Input オブジェクト

Binary Input オブジェクトの Present_Value へ固定値やデバイスを設定します。

画面はオブジェクト一覧で Binary Input オブジェクト ID を選択したときの例です。



① デバイス制御

Present_Value に固定値を設定するか、またはデータレジスタを割り付けるかを選択します。

デバイス制御	内容
無効	Present_Valueに固定値を設定する
有効	Present_Valueにデータレジスタを割り付ける

■ Present_Value に固定値を設定する場合

Present_Value のデバイス制御 (①) に "無効" を設定し、初期値に固定値を設定します。



■ Present_Value にデバイスを割り付ける場合

Present_Value のデバイス制御 (①) に "有効" を設定し、[Present_Value 設定] ダイアログボックスで各パラメータを設定します。



① Present_Value デバイス

Present_Value を格納するデバイスを設定します。

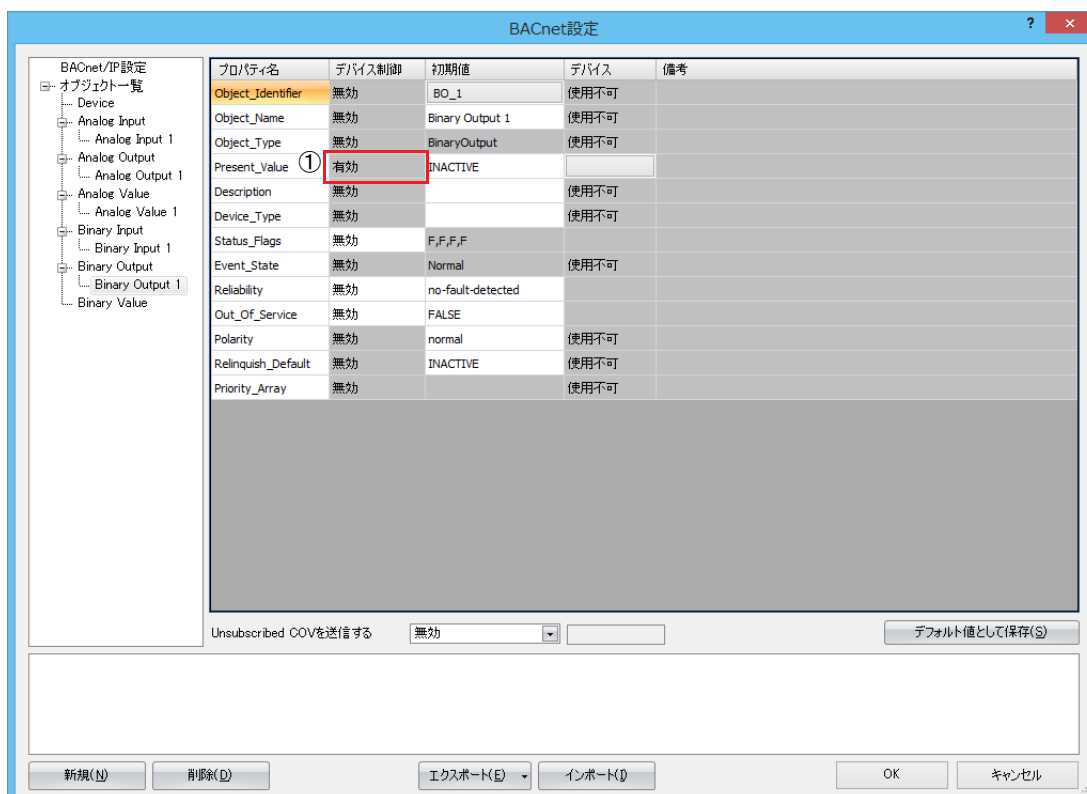
プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	○	—	○*1	—	—	—	—	—

*1 特殊内部リレーは使用できません。

Binary Output オブジェクト

Binary Output オブジェクトの Present_Value へデバイスを設定します。

画面はオブジェクト一覧で Binary Output オブジェクト ID を選択したときの例です。



① デバイス制御

Present_Value のデバイス制御は "有効" 固定です。



Binary Output オブジェクトの Present_Value に固定値を設定することはできません。

■ Present_Value にデバイスを割り付ける場合

[Present_Value 設定] ダイアログボックスで各パラメータを設定します。



① Present_Value デバイス

Present_Value を格納するデバイスを設定します。

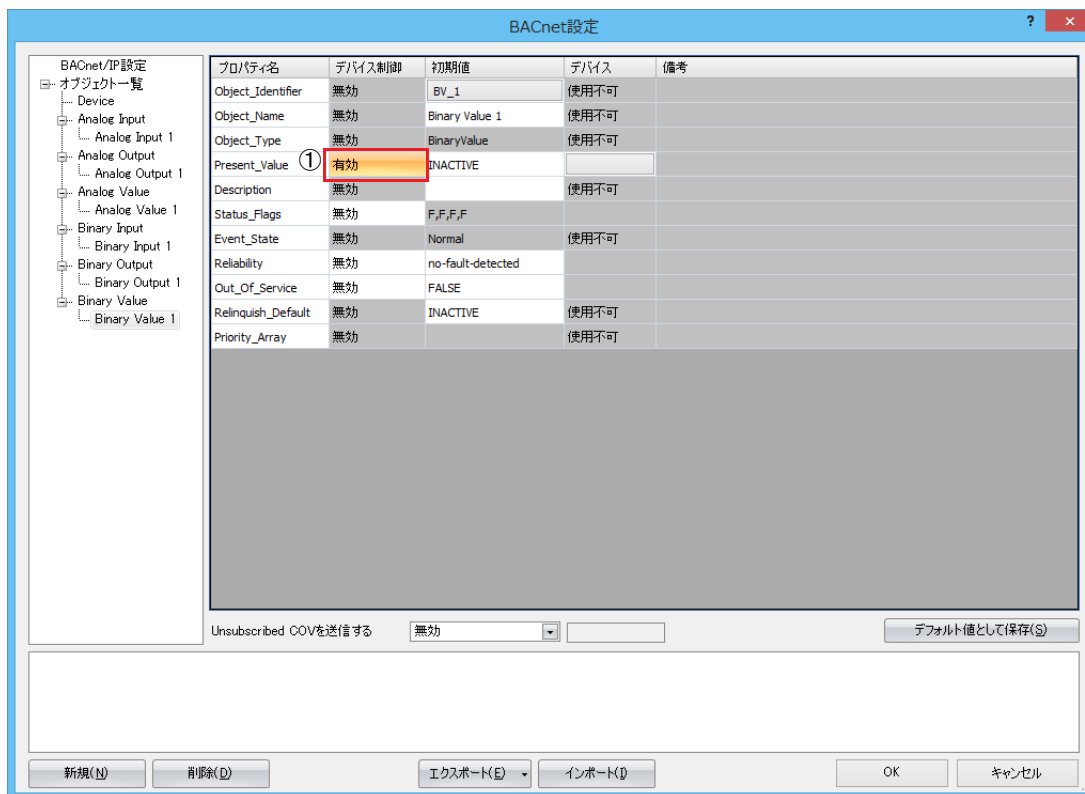
プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	—	○	○*1	—	—	—	—	—

*1 特殊内部リレーは使用できません。

Binary Value オブジェクト

Binary Value オブジェクトの Present_Value へデバイスを設定します。

画面はオブジェクト一覧で Binary Value オブジェクト ID を選択したときの例です。



① デバイス制御

Present_Value のデバイス制御は "有効" 固定です。



Binary Value オブジェクトの Present_Value に固定値を設定することはできません。

■ Present_Value にデバイスを割り付ける場合

[Present_Value 設定] ダイアログボックスで各パラメータを設定します。



① Present_Value デバイス

Present_Value の読み出し用デバイスと、Present_Value への書き込み用デバイスを設定します。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	—	—	○*1	—	—	—	—	—

*1 特殊内部リレーは使用できません。

設定したデバイスにしたがって、読み出し用 Present_Value と書き込み用 Present_Value のデバイスが自動的に割り付きます。指定したデータレジスタを先頭に連続して 2 ビット分のビットデバイスを使用します。デバイスの範囲を超えないように先頭のビットデバイスを指定してください。

Present_Value デバイス	格納先
読み出し用Present_Value	先頭番号+0
書き込み用Present_Value	先頭番号+1

② 優先度およびトリガの先頭デバイス

Present_Value へデバイスの値を書き込む場合に使用します。詳細については、「Binary Value オブジェクト」(15-52 頁)の Present_Value を参照してください。

データレジスタを設定すると、「優先度用デバイス」および「書き込みトリガ用デバイス」が自動的に割り付きます。指定したデータレジスタを先頭に連続して 2 ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。

格納先	内容
先頭番号+0	優先度用デバイス
	ビット15 NULL書き込みフラグ*1 OFF：書き込み用Present_Valueデバイスの値を書き込む ON：NULLを書き込む
	ビット14~5 無効
	ビット4~0 優先度*2
先頭番号+1	書き込みトリガ用デバイス

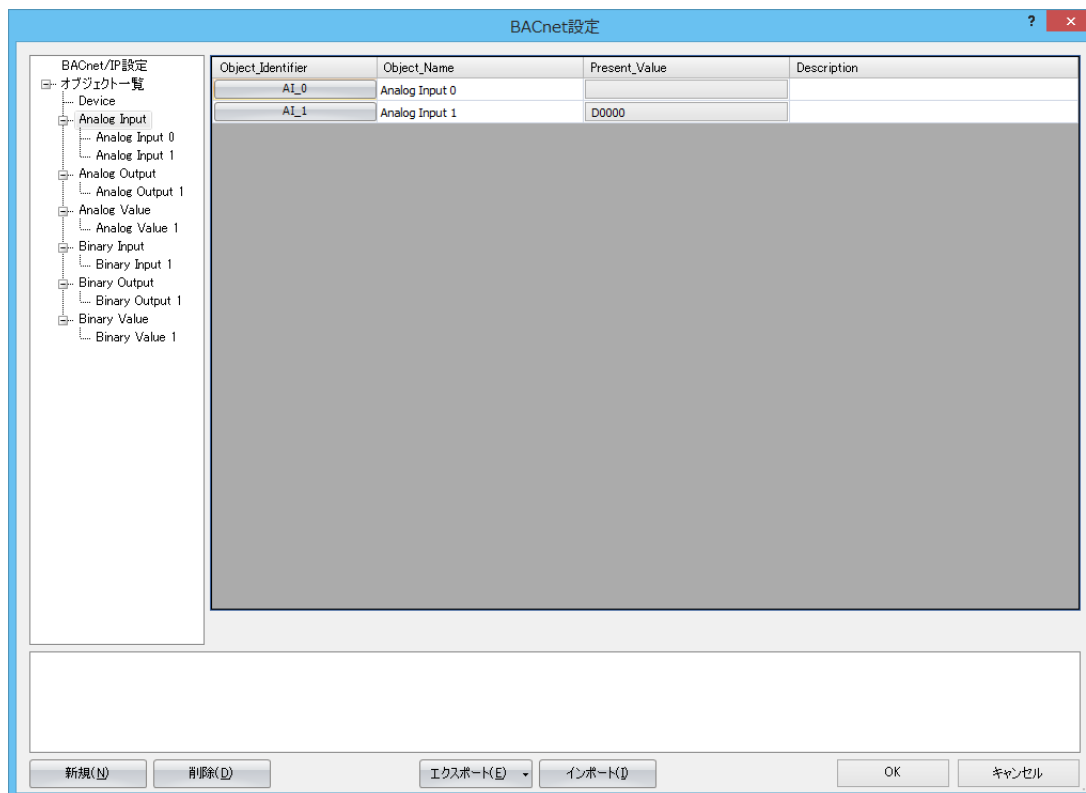
*1 NULL 書き込みフラグが ON のとき、書き込みトリガ用デバイスを OFF から ON にすると、優先度が示すインデックス番号の Priority_Array へ NULL を書き込みます。

*2 優先度は 1 ~ 16 を指定してください。範囲外を指定した場合は、書き込みトリガ用デバイスを操作しても書き込まれません。

オブジェクト一覧

各オブジェクトのノードを選択すると、登録済みのオブジェクトの一覧を表示します。

例えば、Analog Input を選択すると、Analog Input オブジェクトの一覧を表示、オブジェクト一覧ノードを選択すると、すべてのオブジェクトの一覧を表示します。なお、一覧表示時も各プロパティを変更することができます。



オブジェクト

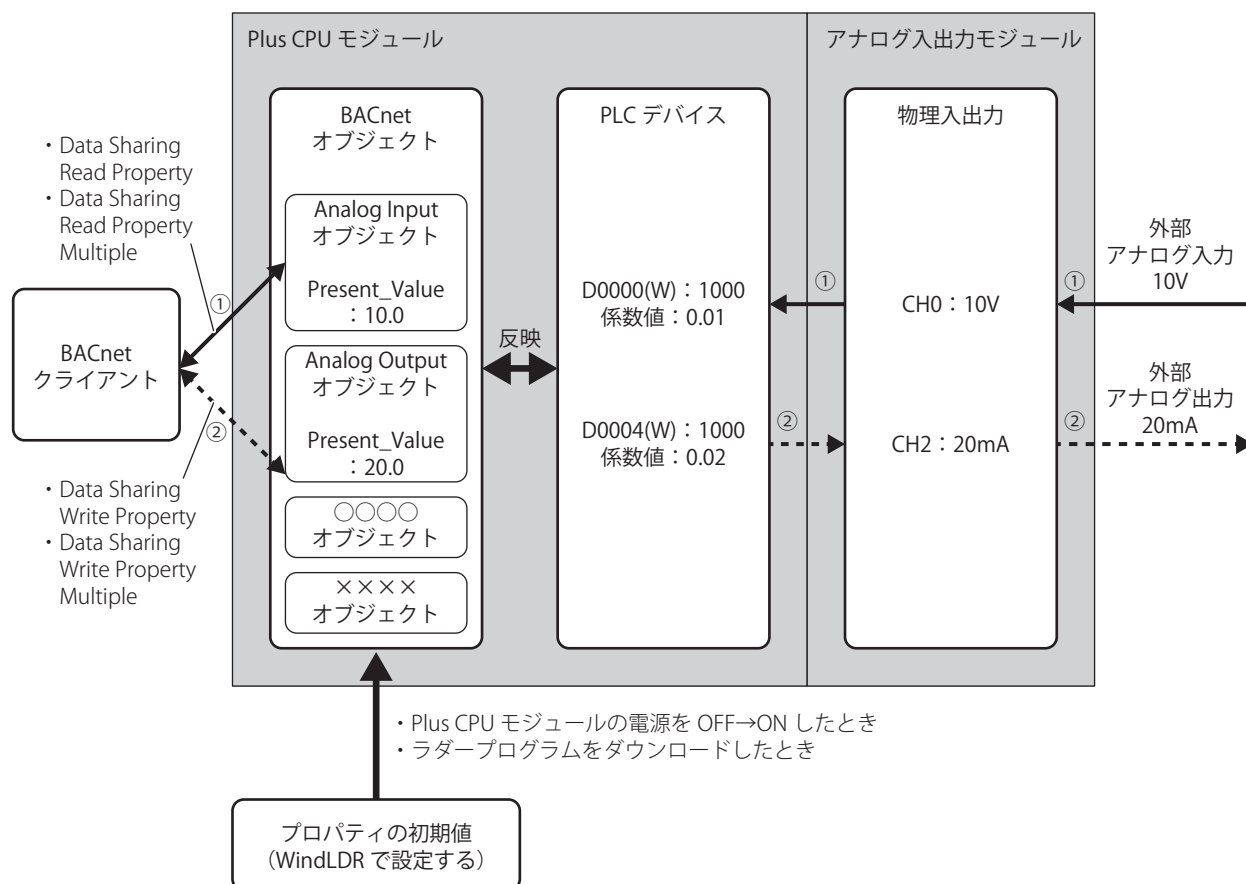
Plus CPU モジュールは、WindLDR で登録したオブジェクトを、内部メモリ上に持ちます。オブジェクトの持つプロパティの一部は、データレジスタに割り付けることができ、ラダープログラムから読み出し、書き込みできます。BACnet デバイスは、サービスを使用して Plus CPU モジュールのオブジェクトの各プロパティを読み出し、書き込みできます。

Plus CPU モジュールのオブジェクトのプロパティと、プロパティへ割り付けたデバイスの値は随時同期します。

WindLDR を使ったオブジェクトの登録方法は、「新規オブジェクトの追加」(15-18 頁) を参照してください。なお、登録できるオブジェクトの最大数は 256 個です。

下記に 2 つのイメージを示します。

- ① BACnet デバイスが Analog Input オブジェクトのアナログ入力値を読み出す
- ② BACnet デバイスが Analog Output オブジェクトのアナログ出力値を書き込む



オブジェクトが持つ各プロパティの詳細については、書籍 "BACnet ビルディングオートメーション用データ通信プロトコル" を参照してください。

Analog Input オブジェクト

フロート型の数値を管理するオブジェクトです。Plus CPU モジュールが任意の数値を BACnet デバイスへ公開できます。例えば、アナログ入力モジュールで取り込んだアナログ値や、計測した室内温度などを扱う際に使用します。

プロパティ一覧

Analog Input オブジェクトでサポートするプロパティを示します。

- ①：BACnet デバイスからの読み出し / 書き込み
- ②：プロパティをデバイスに割り付けた際のラダープログラムからの読み出し / 書き込み

R：読み出しのみ可能、W：書き込みのみ可能、R/W：読み出し / 書き込み可能、-：読み出し / 書き込み不可

プロパティの識別子	プロパティのデータ型	①	②	備考
Object_Identifier	BACnetObjectIdentifier	R	-	WindLDRで固定値を設定します。
Object_Name	文字列*1	R	-	WindLDRで固定値を設定します。
Object_Type	BACnetObjectType	R	-	WindLDRで固定値を設定します。
Present_Value	実数	R	W	「Present_Value」(15-41頁) 参照
Description	文字列*1	R	-	WindLDRで固定値を設定します。
Device_Type	文字列*1	R	-	WindLDRで固定値を設定します。
Status_Flags	BACnetStatus_Flags	R	R	「Status_Flags」(15-54頁) 参照
Event_State	BACnetEventState	R	-	Normal固定です。
Reliability	BACnetReliability	R	R/W	Present_Valueの値が信頼できるかどうかを示します。
Out_Of_Service	論理値	R/W	R/W	「Out_Of_Service」(15-55頁) 参照
Units	BACnetEngineeringUnits	R/W	-	WindLDRで初期値を設定します。
Resolution	実数	R	-	WindLDRで固定値を設定します。
COV_Increment	実数	R/W	-	「COV_Increment」(15-54頁) 参照
Property_List	BACnetPropertyIdentifier型のBACnetARRAY[N]	Hidden	-	WindLDRでは非表示です。

*1 エンコードは ISO 10646(UTF-8) で、サイズは最大 64 バイトです。

Analog Output オブジェクト

フロー型の数値を管理するオブジェクトです。Plus CPU モジュールが BACnet デバイスから任意の数値を受け取ることができません。例えば、アナログ出力モジュールで出力するアナログ値や、動作パラメータとしての設定温度などを、BACnet デバイスから受け取る際に使用します。

プロパティ一覧

Analog Output オブジェクトでサポートするプロパティを示します。

①：BACnet デバイスからの読み出し / 書き込み

②：プロパティをデバイスに割り付けた際のラダープログラムからの読み出し / 書き込み

R：読み出しのみ可能、W：書き込みのみ可能、R/W：読み出し / 書き込み可能、-：読み出し / 書き込み不可

プロパティの識別子	プロパティのデータ型	①	②	備考
Object_Identifier	BACnetObjectIdentifier	R	-	WindLDRで固定値を設定します。
Object_Name	文字列*1	R	-	WindLDRで固定値を設定します。
Object_Type	BACnetObjectType	R	-	WindLDRで固定値を設定します。
Present_Value	実数	R	R	「Present_Value」(15-41頁) 参照
Description	文字列*1	R	-	WindLDRで固定値を設定します。
Device_Type	文字列*1	R	-	WindLDRで固定値を設定します。
Status_Flags	BACnetStatus_Flags	R	R	「Status_Flags」(15-54頁) 参照
Event_State	BACnetEventState	R	-	Normal固定です。
Reliability	BACnetReliability	R	R/W	Present_Valueの値が信頼できるかどうかを示します。
Out_Of_Service	論理値	R/W	R/W	「Out_Of_Service」(15-55頁) 参照
Units	BACnetEngineeringUnits	R/W	-	WindLDRで初期値を設定します。
Resolution	実数	R	-	WindLDRで固定値を設定します。
Priority_Array	BACnetPriority_Array	R/W	-	「Priority_Array」(15-54頁) 参照
Relinquish_Default	実数	R/W	-	「Relinquish_Default」(15-54頁) 参照
COV_Increment	実数	R/W	-	「COV_Increment」(15-54頁) 参照
Property_List	BACnetPropertyIdentifier型のBACnetARRAY[N]	Hidden	-	WindLDRでは非表示です。

*1 エンコードはISO 10646(UTF-8)で、サイズは最大64バイトです。

Analog Value オブジェクト

Analog Input オブジェクトまたは Analog Output オブジェクトと同じように使用できます。

プロパティ一覧

Analog Value オブジェクトでサポートするプロパティを示します。

- ①：BACnet デバイスからの読み出し / 書き込み
- ②：プロパティをデバイスに割り付けた際のラダープログラムからの読み出し / 書き込み

R：読み出しのみ可能、W：書き込みのみ可能、R/W：読み出し / 書き込み可能、-：読み出し / 書き込み不可

プロパティの識別子	プロパティのデータ型	①	②	備考
Object_Identifier	BACnetObjectIdentifier	R	-	WindLDRで固定値を設定します。
Object_Name	文字列*1	R	-	WindLDRで固定値を設定します。
Object_Type	BACnetObjectType	R	-	WindLDRで固定値を設定します。
Present_Value	実数	R	R	「Present_Value」(15-41頁) 参照
Description	文字列*1	R	-	WindLDRで固定値を設定します。
Status_Flags	BACnetStatus_Flags	R	R	「Status_Flags」(15-54頁) 参照
Event_State	BACnetEventState	R	-	Normal固定です。
Reliability	BACnetReliability	R	R/W	Present_Valueが信頼できる値かどうかを示します。
Out_Of_Service	論理値	R/W	R/W	「Out_Of_Service」(15-55頁) 参照
Units	BACnetEngineeringUnits	R/W	-	WindLDRで初期値を設定します。
Priority_Array	BACnetPriority_Array	R/W	R/W	「Priority_Array」(15-54頁) 参照
Relinquish_Default	実数	R/W	-	「Relinquish_Default」(15-54頁) 参照
COV_Increment	実数	R/W	-	「COV_Increment」(15-54頁) 参照
Resolution	実数	R	-	WindLDRで固定値を設定します。
Property_List	BACnetPropertyIdentifier型の BACnetARRAY[N]	Hidden	-	WindLDRでは非表示です。

*1 エンコードは ISO 10646(UTF-8) で、サイズは最大 64 バイトです。

Binary Input オブジェクト

バイナリ値 (ON/OFF) を管理するオブジェクトです。Plus CPU モジュールがバイナリ値を BACnet デバイスへ公開する際に使用します。

プロパティ一覧

Binary Input オブジェクトでサポートするプロパティを示します。

- ①：BACnet デバイスからの読み出し / 書き込み
 ②：プロパティをデバイスに割り付けた際のラダープログラムからの読み出し / 書き込み

R：読み出しのみ可能、W：書き込みのみ可能、R/W：読み出し / 書き込み可能、-：読み出し / 書き込み不可

プロパティの識別子	プロパティのデータ型	①	②	備考
Object_Identifier	BACnetObjectIdentifier	R	-	WindLDRで固定値を設定します。
Object_Name	文字列*1	R	-	WindLDRで固定値を設定します。
Object_Type	BACnetObjectType	R	-	WindLDRで固定値を設定します。
Present_Value	BACnetBinaryPV	R	W	「Present_Value」(15-41頁) 参照
Description	文字列*1	R	-	WindLDRで固定値を設定します。
Device_Type	文字列*1	R	-	WindLDRで固定値を設定します。
Status_Flags	BACnetStatus_Flags	R	R	「Status_Flags」(15-54頁) 参照
Event_State	BACnetEventState	R	-	Normal固定です。
Reliability	BACnetReliability	R	R/W	Present_Valueの値が信頼できるかどうかを示します。
Out_Of_Service	論理値	R/W	R/W	「Out_Of_Service」(15-55頁) 参照
Polarity	BACnetPolarity	R/W	-	「Polarity」(15-54頁) 参照
Property_List	BACnetPropertyIdentifier型のBACnetARRAY[N]	Hidden	-	WindLDRでは非表示です。

*1 エンコードは ISO 10646(UFT-8) で、サイズは最大 64 バイトです。

Binary Output オブジェクト

バイナリ値 (ON/OFF) を管理するオブジェクトです。Plus CPU モジュールが BACnet デバイスからバイナリ値を受け取る際に使用します。

プロパティ一覧

Binary Output オブジェクトでサポートするプロパティを示します。

- ① : BACnet デバイスからの読み出し / 書き込み
- ② : プロパティをデバイスに割り付けた際のラダープログラムからの読み出し / 書き込み

R : 読み出しのみ可能、W : 書き込みのみ可能、R/W : 読み出し / 書き込み可能、- : 読み出し / 書き込み不可

プロパティの識別子	プロパティのデータ型	①	②	備考
Object_Identifier	BACnetObjectIdentifier	R	-	WindLDRで固定値を設定します。
Object_Name	文字列*1	R	-	WindLDRで固定値を設定します。
Object_Type	BACnetObjectType	R	-	WindLDRで固定値を設定します。
Present_Value	BACnetBinaryPV	R	R	「Present_Value」(15-41頁) 参照
Description	文字列*1	R	-	WindLDRで固定値を設定します。
Device_Type	文字列*1	R	-	WindLDRで固定値を設定します。
Status_Flags	BACnetStatus_Flags	R	R	「Status_Flags」(15-54頁) 参照
Event_State	BACnetEventState	R	-	Normal固定です。
Reliability	BACnetReliability	R	R/W	Present_Valueの値が信頼できるかどうかを示します。
Out_Of_Service	論理値	R/W	R/W	「Out_Of_Service」(15-55頁) 参照
Polarity	BACnetPolarity	R/W	-	「Polarity」(15-54頁) 参照
Priority_Array	BACnetPriority_Array	R/W	-	「Priority_Array」(15-54頁) 参照
Relinquish_Default	BACnetBinaryPV	R/W	-	「Relinquish_Default」(15-54頁) 参照
Property_List	BACnetPropertyIdentifier型のBACnetARRAY[N]	Hidden	-	WindLDRでは非表示です。

*1 エンコードは ISO 10646(UTF-8) で、サイズは最大 64 バイトです。

Binary Value オブジェクト

Binary Input オブジェクトまたは Binary Output オブジェクトと同じように使用できます。

プロパティ一覧

Binary Value オブジェクトでサポートするプロパティを示します。

①：BACnet デバイスからの読み出し / 書き込み

②：プロパティをデバイスに割り付けた際のラダープログラムからの読み出し / 書き込み

R：読み出しのみ可能、W：書き込みのみ可能、R/W：読み出し / 書き込み可能、-：読み出し / 書き込み不可

プロパティの識別子	プロパティのデータ型	①	②	備考
Object_Identifier	BACnetObjectIdentifier	R	-	WindLDRで固定値を設定します。
Object_Name	文字列*1	R	-	WindLDRで固定値を設定します。
Object_Type	BACnetObjectType	R	-	WindLDRで固定値を設定します。
Present_Value	BACnetBinaryPV	R	R	「Present_Value」(15-41頁) 参照
Description	文字列*1	R	-	WindLDRで固定値を設定します。
Status_Flags	BACnetStatus_Flags	R	R	「Status_Flags」(15-54頁) 参照
Event_State	BACnetEventState	R	-	Normal固定です。
Reliability	BACnetReliability	R	R/W	Present_Valueの値が信頼できるかどうかを示します。
Out_Of_Service	論理値	R/W	R/W	「Out_Of_Service」(15-55頁) 参照
Priority_Array	BACnetPriority_Array	R/W	R/W	「Priority_Array」(15-54頁) 参照
Relinquish_Default	BACnetBinaryPV	R/W	-	「Relinquish_Default」(15-54頁) 参照
Property_List	BACnetPropertyIdentifier型の BACnetARRAY[N]	Hidden	-	WindLDRでは非表示です。

*1 エンコードは ISO 10646(UTF-8) で、サイズは最大 64 バイトです。

Device オブジェクト

Plus CPU モジュールの基本的な情報を BACnet デバイスへ公開する際に使用します。

プロパティ一覧

Device オブジェクトでサポートするプロパティを示します。

- ①：BACnet デバイスからの読み出し / 書き込み
- ②：プロパティをデバイスに割り付けた際のラダープログラムからの読み出し / 書き込み

R：読み出しのみ可能、W：書き込みのみ可能、R/W：読み出し / 書き込み可能、-：読み出し / 書き込み不可

プロパティの識別子	プロパティのデータ型	①	②	備考
Object_Identifier	BACnetObjectIdentifier	R	-	WindLDRで編集できません。
Object_Name	文字列*1	R	-	WindLDRで固定値を設定します。
Object_Type	BACnetObjectType	R	-	WindLDRで編集できません。
System_Status	BACnetDeviceStatus	R	-	WindLDRで編集できません。
Vendor_Name	文字列*1	R	-	WindLDRで編集できません。
Vendor_Identifier	16ビット符号無し整数	R	-	WindLDRで編集できません。
Model_Name	文字列*1	R	-	WindLDRで固定値を設定します。
Firmware_Revision	文字列*1	R	-	WindLDRで編集できません。
Application_Software_Version	文字列*1	R	-	WindLDRで固定値を設定します。
Location	文字列*1	R/W	-	WindLDRで初期値を設定します。
Description	文字列*1	R/W	-	WindLDRで初期値を設定します。
Protocol_Version	符号無し整数	R	-	WindLDRで編集できません。
Protocol_Revision	符号無し整数	R	-	WindLDRで編集できません。
Protocol_Services_Supported	BACnetServicesSupported	R	-	WindLDRで編集できません。
Protocol_Object_Types_Supported	BACnetObjectType Supported	R	-	WindLDRで編集できません。
Object_List	BACnetObjectIdentifier型の BACnetARRAY[N]	R	-	WindLDRで編集できません。
Max_APDU_Length_Accepted	符号無し整数	R	-	WindLDRで編集できません。
Segmentation_Supported	BACnetSegmentation	R	-	WindLDRで編集できません。
Local_Time	時刻	R	-	WindLDRで編集できません。
Local_Date	日付	R	-	WindLDRで編集できません。
APDU_Timeout	符号無し整数	R	-	WindLDRで編集できません。
Number_of_APDU_Retries	符号無し整数	R	-	WindLDRで編集できません。
Device_Address_Binding	BACnetAddressBiding型の BACnetLIST	R	-	WindLDRで編集できません。
Database_Revision	符号無し整数	R	-	WindLDRで編集できません。
Property_List	BACnetPropertyIdentifier型の BACnetARRAY[N]	Hidden	-	WindLDRでは非表示です。
Profile_Name	文字列*1	R	-	WindLDRで固定値を設定します。

*1 エンコードは ISO 10646(UTF-8) で、サイズは最大 64 バイトです。

主なプロパティ

Present_Value

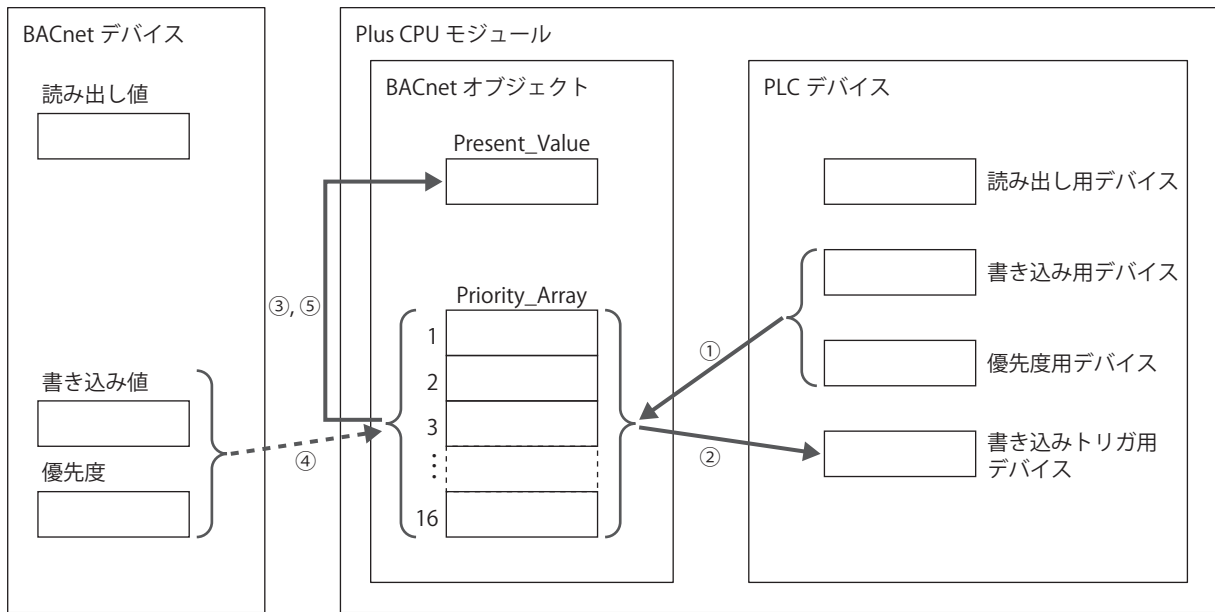
現在値を表すプロパティです。オブジェクトに関連付けられた入出力値などを取り扱うプロパティです。

優先順位機構

BACnet デバイスから書き込み可能な Present_Value を持つオブジェクト*1 は、Priority_Array を利用した優先順位機構を用いて、書き込み指示に優劣を付け、Present_Value の値を決定します。この場合、Present_Value には直接、値を書き込むことができません。Present_Value に値を書き込む場合、値は一旦 " 優先度 " (④「Present_Value 設定」(15-21 頁)) の示すインデックス番号の Priority_Array (「Priority_Array」(15-54 頁)) に格納されます。そして NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array に格納された値が Present_Value の値として採用されます。

*1 Analog Output/ Analog Value/ Binary Output/ Binary Value オブジェクト

■ラダープログラムからデータレジスタの値を Present_Value として書き込む、または BACnet デバイスから Present_Value を書き込む場合



上図に使用されている各項目の説明は下表を参照してください。

項目	内容
読み出し用デバイス	BACnetデバイスから読み出したPresent_Valueを格納するデバイスです。
書き込み用デバイス	Present_Valueとして、BACnetデバイスへ書き込むための値を格納するデバイスです。
優先度用デバイス	優先順位値が格納された配列 (Priority_Array) のインデックス番号を格納するデバイスです。
書き込みトリガ用デバイス	1のとき書き込み用デバイスに格納された値を、優先度用デバイスに格納されたインデックス番号の Priority_Array に格納します。
Present_Value	Plus CPUモジュールのBACnetオブジェクトが保持しているPresent_Valueです。
Priority_Array	「Priority_Array」(15-54頁) 参照
読み出し値	現在のPresent_Valueです。
書き込み値	BACnetデバイスから書き込むPresent_Valueです。

ラダープログラムからの書き込み

- ① "優先度用デバイス" のビット 15 が 0 のとき、"書き込みトリガ用デバイス" が 1 のとき、"優先度用デバイス" に格納されたインデックス番号の Priority_Array に、"書き込み用デバイス" の値を書き込みます。
- ② "書き込みトリガ用デバイス" を自動的に 0 に戻します。
- ③ NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array に格納された値が Present_Value の値として採用されます。(Present_Value として採用されているインデックス番号の Priority_Array の値を NULL に戻すまで、そのインデックス番号の Priority_Array の値が Present_Value の値として採用されます。)



"優先度用デバイス" のビット 15 が 1 のとき、"書き込みトリガ用デバイス" が 1 のとき、"優先度用デバイス" に格納されたインデックス番号の Priority_Array に NULL を書き込みます。
優先度用デバイスのビットアサインの詳細については、「Analog Value オブジェクト」(15-25 頁) の「②優先度およびトリガの先頭デバイス」(15-26 頁) を参照してください。

BACnet デバイスからの書き込み

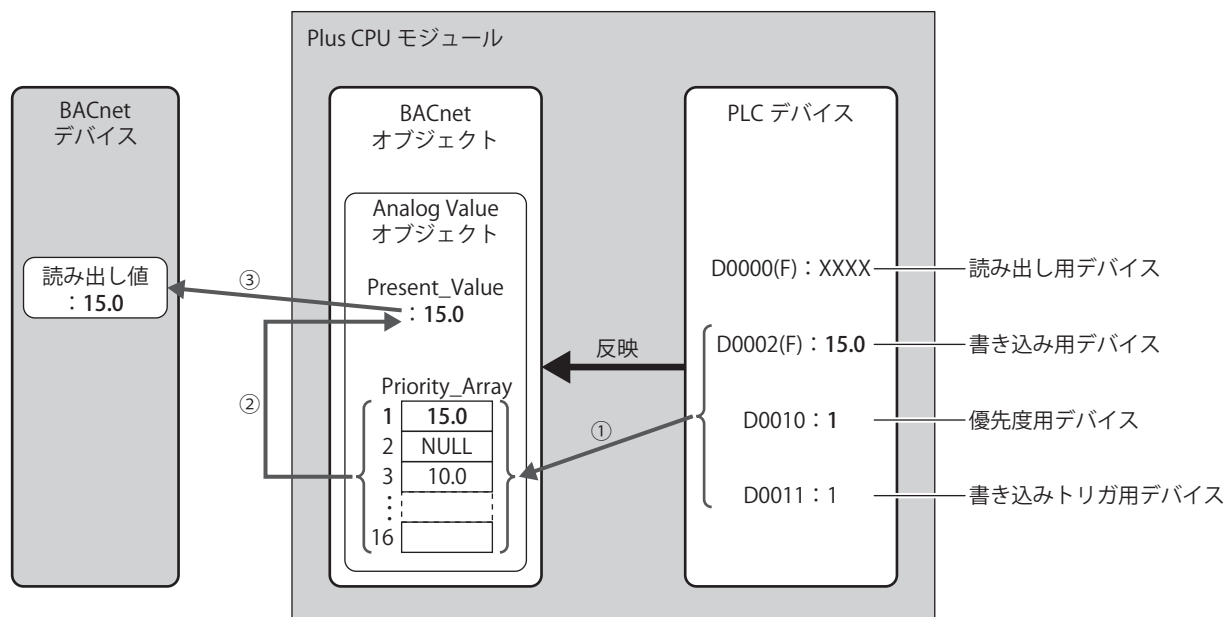
- ④ BACnet デバイスが指定した優先度が示すインデックス番号の Priority_Array に、Present_Value の値を書き込みます。
- ⑤ NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array プロパティに格納された値が Present_Value の値として採用されます。(Present_Value として採用されているインデックス番号の Priority_Array の値を NULL に戻すまで、そのインデックス番号の Priority_Array の値が Present_Value の値として採用されます。)



Priority_Array に格納された値がすべて NULL である場合は、「Relinquish_Default」(15-54 頁) が Present_Value として採用されます。

ラダープログラムからデータレジスタの値を Present_Value に書き込む場合

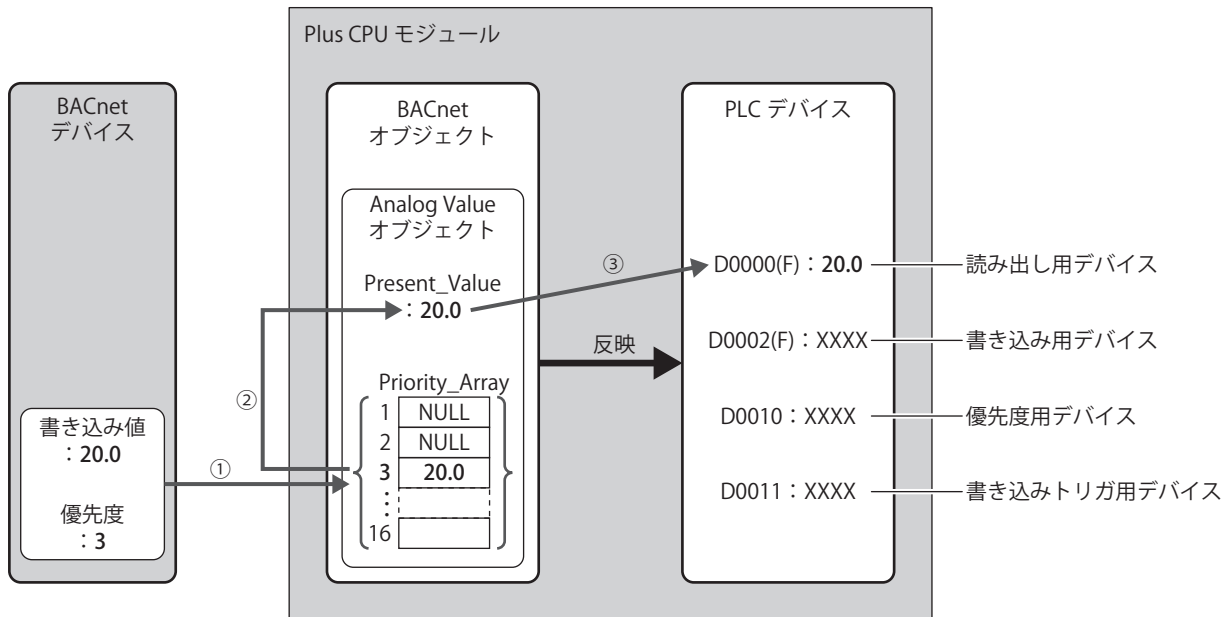
例えば、Present Value デバイスに D0000、優先度およびトリガの先頭デバイスに D0010、変換タイプにフロート（無変換）を割り付けた場合、読み出し用デバイス：D0000、書き込み用デバイス：D0002、優先度用デバイス：D0010、書き込みトリガ用デバイス：D0011 が割り付きます。



- ① 書き込みトリガ用デバイス（D0011）が1のとき、優先度用デバイス（D0010）が1であるので、Priority_Arrayの第1要素に書き込み用デバイス（D0002）の値（15.0）を書き込みます。書き込み完了後、書き込みトリガ用デバイス（D0011）を0に戻します。
- ② NULL以外の値が格納されている最もインデックス番号の小さいPriority_Arrayは、第1要素（15.0）であるため、15.0がPresent_Valueの値として採用されます。
- ③ BACnetデバイスからはPresent_Value（15.0）が読み出されます。

BACnet デバイスから Present_Value に値を書き込む場合

例えば、Present Value デバイスに D0000、優先度およびトリガの先頭デバイスに D0010、変換タイプにフロート（無変換）を割り付けた場合、読み出し用デバイス：D0000、書き込み用デバイス：D0002、優先度用デバイス：D0010、書き込みトリガ用デバイス：D0011 が割り付きます。

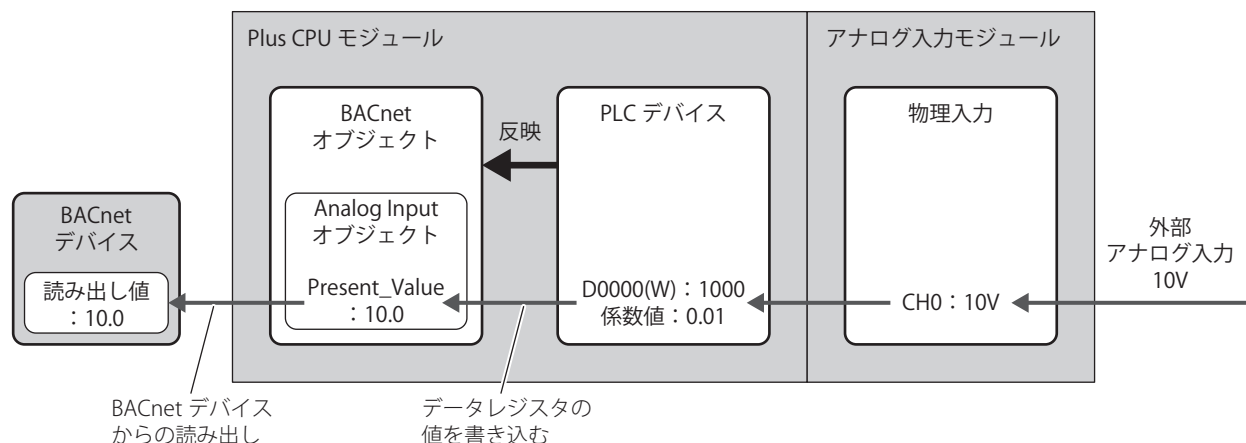


- ① BACnet デバイスが指定した優先度（3）が示すインデックス番号の Priority_Array に、書き込み値（20.0）を書き込みます。
- ② NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array は、第 3 要素（20.0）であるため、20.0 が Present_Value の値として採用されます。
- ③ Present_Value（20.0）を読み出し用デバイス（D0000）に書き込みます。

Analog Input オブジェクト

Analog Input オブジェクトの Present_Value には、固定値を設定するか、データレジスタを割り付けて、そのデータレジスタの値を設定できます。Present_Value はフロート型の数値です。Present_Value にデータレジスタを割り付けた場合、そのデータレジスタの値に係数値を乗算して Present_Value に設定します。

下図に、Analog Input オブジェクトの Present_Value にデータレジスタを割り付けて、BACnet デバイスがアナログ入力値を読み出す場合のイメージを示します。



Analog Input オブジェクトの Present_Value に割り付けることができるデバイスはデータレジスタのみです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	-	-	-	-	-	-	○*1	-

*1 特殊データレジスタは使用できません。

データレジスタの値を Present_Value として書き込む場合は、次のように Present_Value が変化します。

データタイプ	データレジスタの値	BACnet デバイスの Present_Value
ワード	各データタイプの範囲内の値	データレジスタの値
インデジャ		
ダブルワード		
ロング		
フロート (無変換)	±0	±0.0
	非正規化数	データレジスタの値
	正規化数	
	±∞ (±無限大)	Present_Valueは変化しない
	非数	

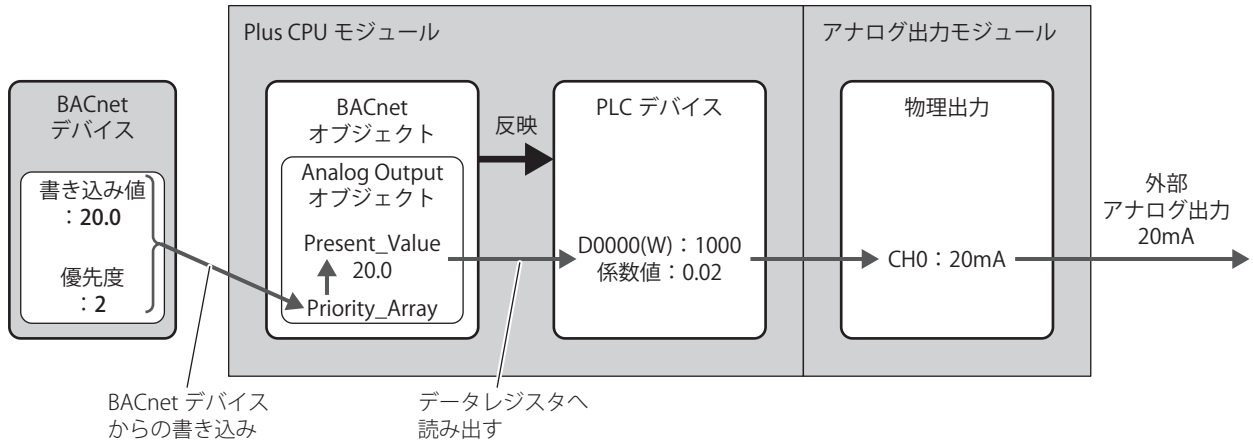
Plus CPU モジュールで扱う Analog Input オブジェクトの Present_Value は、そのデータタイプによって、書き込みのために必要なデータレジスタの個数が異なります。

Plus CPU モジュールで扱う Present_Value のデータタイプ	データレジスタの個数
ワード	1
インデジャ	
ダブルワード	
ロング	2
フロート (無変換)	

Analog Output オブジェクト

Analog Output オブジェクトの Present_Value には、データレジスタを割り付けて、そのデータレジスタの値を設定できます。Present_Value はフロート型の数値です。Present_Value にデータレジスタを割り付けた場合、Present_Value に 1/ 係数値を乗算した値をデータレジスタに格納します。

下図に、Analog Output オブジェクトの Present_Value にデータレジスタを割り付けて、BACnet デバイスがアナログ出力値を書き込む場合のイメージを示します。



Analog Output オブジェクトの Present_Value はラダープログラムから変更できません。

Analog Output オブジェクトの Present_Value に割り付けることができるデバイスはデータレジスタのみです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	-	-	-	-	-	-	○*1	-

*1 特殊データレジスタは使用できません。

Present_Value をデータレジスタに読み出す場合は、データタイプを意識する必要があります。データレジスタのデータタイプにしたがって、次のようにデータレジスタに格納されます。Present_Value の値に応じてデータタイプを設定してください。Present_Value のデータタイプは、[Present_Value 設定] ダイアログボックス (15-24 頁) の「②変換タイプ」(15-24 頁) で設定します。

データタイプ	BACnet デバイスの Present_Value	データレジスタの値
ワード	0~65535の範囲内の値	Present_Value
	0~65535の範囲外の値	0
インテジャ	-32768~32767の範囲内の値	Present_Value
	-32768~32767の範囲外の値	0
ダブルワード	0~4294967295の範囲内の値	Present_Value
	0~4294967295の範囲外の値	0
ロング	-2147483648~2147483647の範囲内の値	Present_Value
	-2147483648~2147483647の範囲外の値	0
フロート (無変換)	-	Present_Value

Plus CPU モジュールで扱う Analog Output オブジェクトの Present_Value は、そのデータタイプによって、読み出しのために必要なデータレジスタの個数が異なります。

Plus CPU モジュールで扱う Present_Value のデータタイプ	データレジスタの個数
ワード	1
インテジャ	
ダブルワード	2
ロング	
フロート (無変換)	

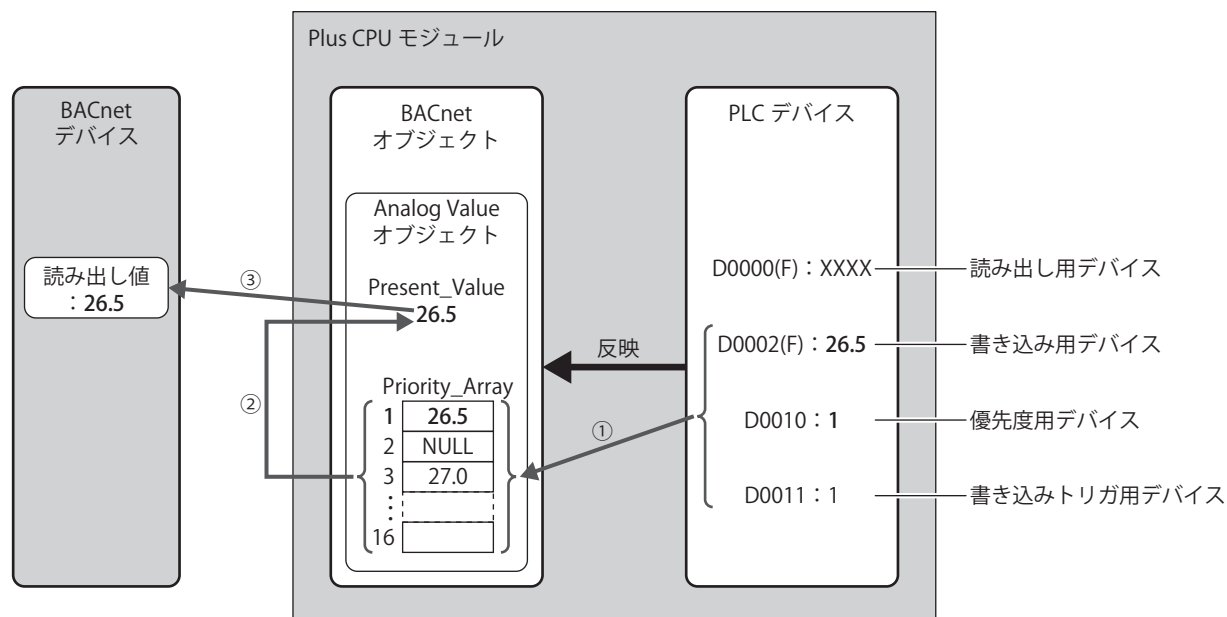
Analog Value オブジェクト

Analog Value オブジェクトの Present_Value は、Analog Input オブジェクトと Analog Output オブジェクトの Present_Value の両方の使い方でできます。Present_Value はフロート型の数値です。

下図に、Analog Value オブジェクトの Present_Value にデータレジスタを割り付けて、オフィスのエアコンの温度を基準温度 (27.0 °C) から一時的に 26.5 °C に下げる場合のイメージを示します。

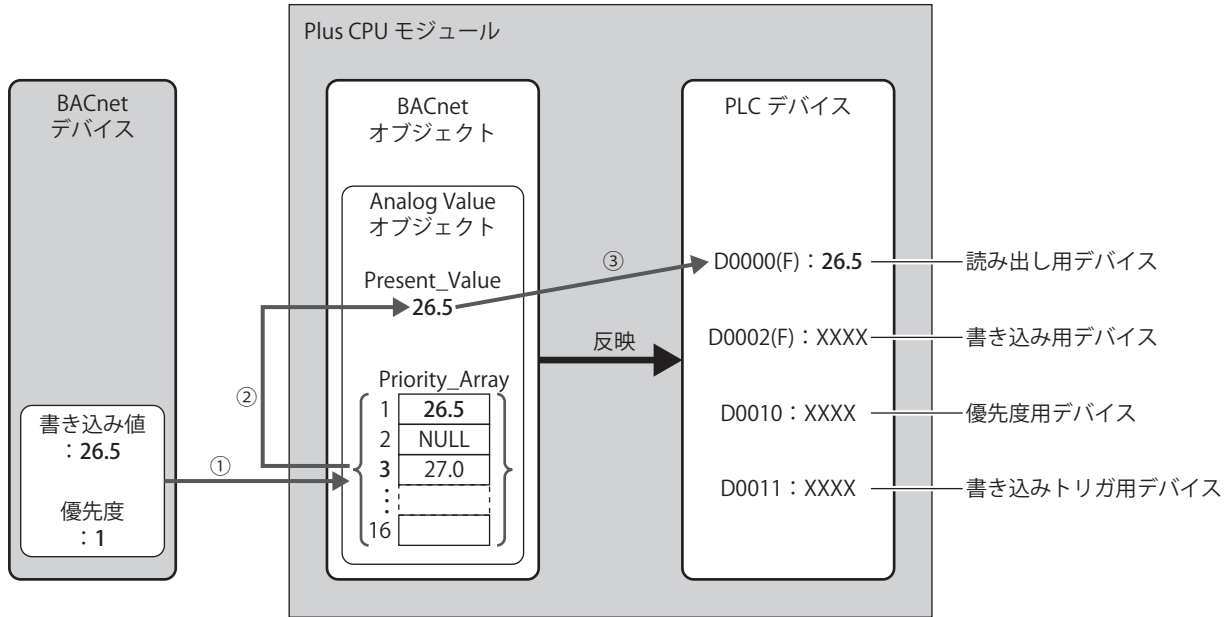
例えば、Present_Value デバイスに D0000、優先度およびトリガの先頭デバイスに D0010、変換タイプにフロート（無変換）を割り付けた場合、読み出し用デバイス：D0000、書き込み用デバイス：D0002、優先度用デバイス：D0010、書き込みトリガ用デバイス：D0011 が割り付きます。

ラダープログラムからデータレジスタの値を Present_Value に書き込む場合



- ① 書き込みトリガ用デバイス (D0011) が 1 のとき、優先度用デバイス (D0010) に格納されたインデックス番号の Priority_Array に、書き込み用デバイス (D0002) の値を書き込みます。
- ② 書き込みトリガ用デバイス (D0011) を自動的に 0 に戻します。
- ③ NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array は、第 1 要素 (26.5) であるため、26.5 が Present_Value の値として採用されます。

BACnet デバイスから Present_Value に値を書き込む場合



- ① BACnet デバイスが指定した優先度 (1) が示すインデックス番号の Priority_Array に、書き込み値 (26.5) を書き込みます。
- ② NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array は、第 1 要素 (26.5) であるため、26.5 が Present_Value の値として採用されます。
- ③ Present_Value (26.5) を読み出し用デバイス (D0000) に書き込みます。



Present_Value として採用されている Priority_Array の第 1 要素 (26.5) を NULL に戻すまで、第 1 要素が Present_Value の値として採用されます。第 1 要素を NULL に戻すと、第 3 要素 (27.0) が Present_Value の値として採用されます。NULL を書き込む方法については、「Analog Value オブジェクト」(15-25 頁) の「②優先度およびトリガの先頭デバイス」(15-26 頁) を参照してください。

Analog Value オブジェクトの Present_Value に割り付けることができるデバイスはデータレジスタのみです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	-	-	-	-	-	-	○*1	-

*1 特殊データレジスタは使用できません。

Plus CPU モジュールで扱う Analog Value オブジェクトの Present_Value のデータタイプによって、読み出し / 書き込みに必要なデータレジスタの個数が異なります。指定したデータレジスタを先頭に 2 個または 4 個のデータレジスタを占有します。

Plus CPU モジュールで扱う Present_Value のデータタイプ	データレジスタの個数		
	読み出し	書き込み	合計
ワード	1	1	2
インテジャ			
ダブルワード	2	2	4
ロング			
フロート (無変換)			

Present_Value は、割り付けたデータレジスタのデータタイプにしたがって、次のようにデータレジスタに格納されます。Present_Value の値に応じてデータタイプを設定してください。Present_Value のデータタイプは、[Present_Value 設定] ダイアログボックス (15-25 頁) の「③変換タイプ」(15-26 頁) で設定します。

データタイプ	BACnet デバイスの Present_Value × (1/ 係数値)	データレジスタの値
ワード	0~65535の範囲内の値	Present_Value
	0~65535の範囲外の値	0
インテジャ	-32768~32767の範囲内の値	Present_Value
	-32768~32767の範囲外の値	0
ダブルワード	0~4294967295の範囲内の値	Present_Value
	0~4294967295の範囲外の値	0
ロング	-2147483648~2147483647の範囲内の値	Present_Value
	-2147483648~2147483647の範囲外の値	0
フロート (無変換)	—	Present_Value

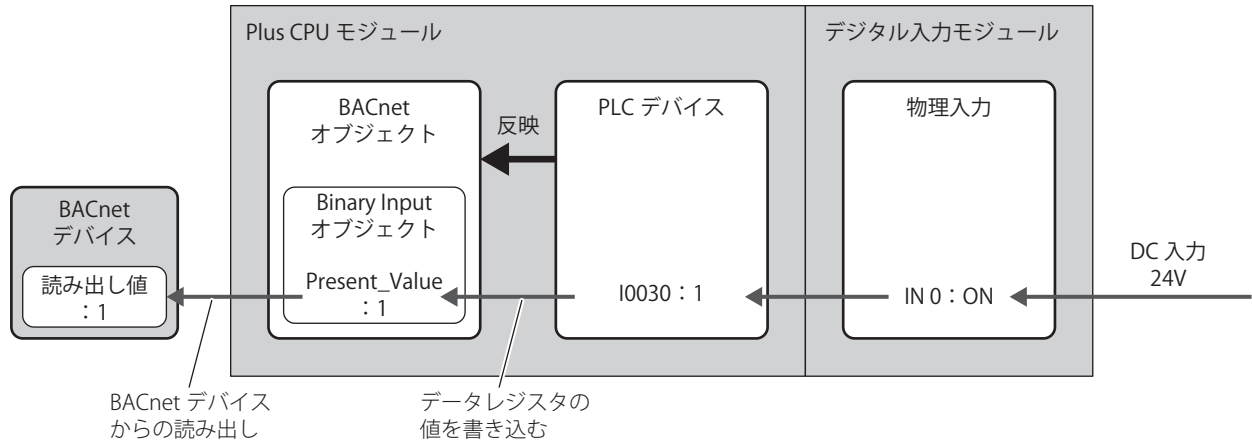
また、データレジスタの値を Present_Value に書き込む場合は、次のように Present_Value が変化します。

データタイプ	データレジスタの値	BACnet デバイスの Present_Value
ワード	各データタイプの範囲内の値	データレジスタの値 × 係数値
インテジャ		
ダブルワード		
ロング		
フロート (無変換)	±0	±0.0
	非正規化数	データレジスタの値 × 係数値
	正規化数	
	±∞ (±無限大)	Present_Valueは変化しない
	非数	

Binary Input オブジェクト

Binary Input オブジェクトの Present_Value には、固定値を設定するか、ビットデバイスを割り付けて、そのビットデバイスの値を設定できます。Present_Value はバイナリ値です。

下図に、Binary Input オブジェクトの Present_Value に外部入力を割り付けて、BACnet デバイスが外部入力の状態を読み出す場合のイメージを示します。



Binary Input オブジェクトの Present_Value に割り付けることができるデバイスは次のとおりです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	○	—	○*1	—	—	—	—	—

*1 特殊内部リレーは使用できません。

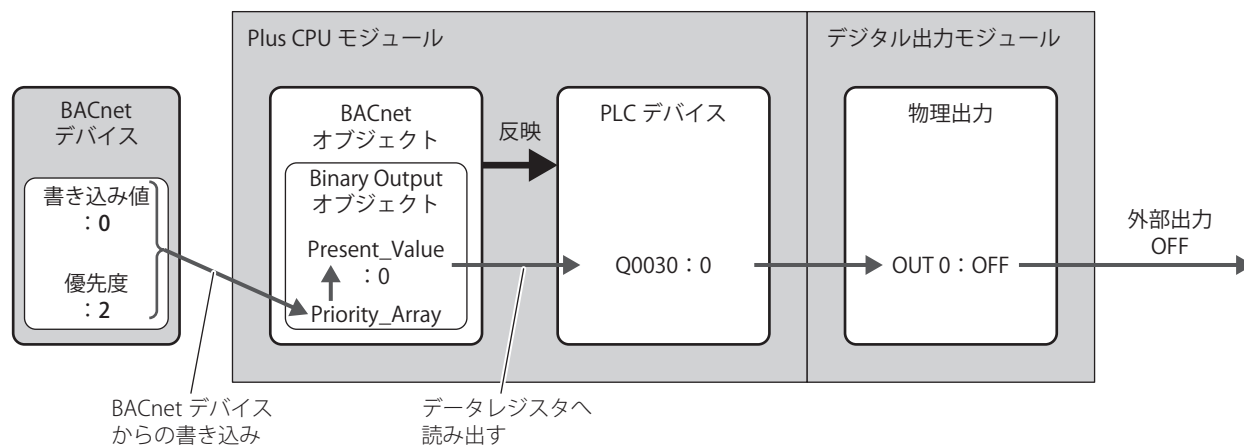
下表は Present_Value、Polarity および入力の物理状態について示します。

Present_Value	Polarity	入力の物理状態
INACTIVE	NORMAL	OFFまたはINACTIVE
ACTIVE	NORMAL	ONまたはACTIVE
INACTIVE	REVERSE	ONまたはACTIVE
ACTIVE	REVERSE	OFFまたはINACTIVE

Binary Output オブジェクト

Binary Output オブジェクトの Present_Value には、ビットデバイスを割り付けて、そのビットデバイスの値を設定できます。Present_Value はバイナリ値です。

下図に、Binary Output オブジェクトの Present_Value にデータレジスタを割り付けて、BACnet デバイスが外部出力の状態を書き込む場合のイメージを示します。



Binary Output オブジェクトの Present_Value に割り付けることができるデバイスは次のとおりです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	—	○	○*1	—	—	—	—	—

*1 特殊内部リレーは使用できません。

下表は Present_Value、Polarity および出力の物理状態について示します。

Present_Value	Polarity	出力の物理状態
INACTIVE	NORMAL	OFFまたはINACTIVE
ACTIVE	NORMAL	ONまたはACTIVE
INACTIVE	REVERSE	ONまたはACTIVE
ACTIVE	REVERSE	OFFまたはINACTIVE

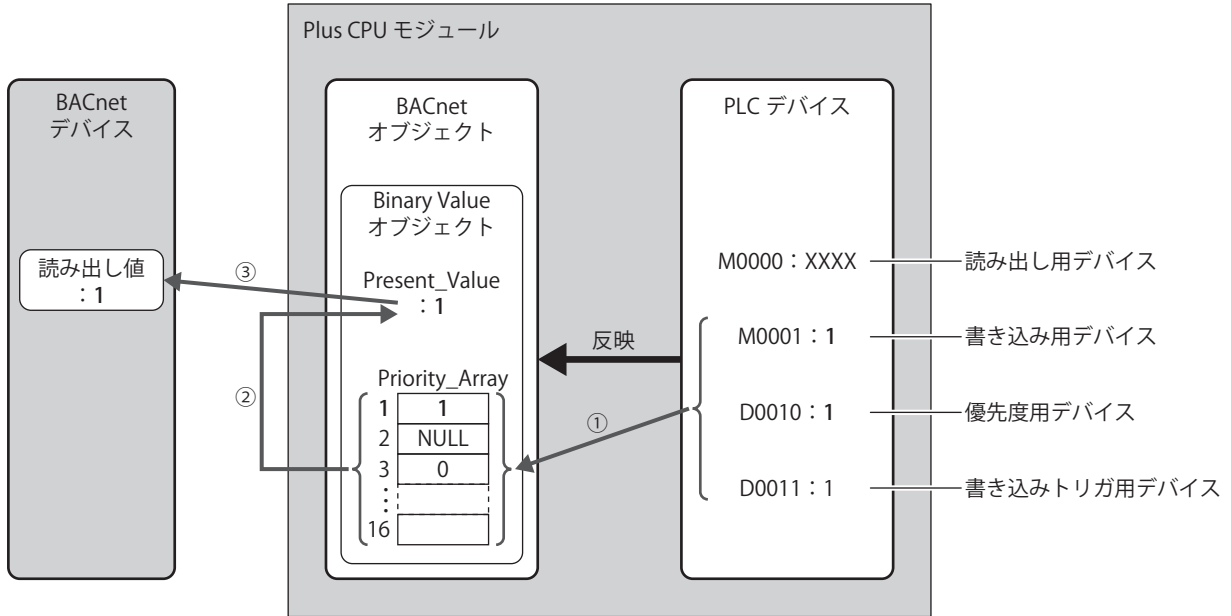
Binary Value オブジェクト

Binary Value オブジェクトの Present_Value は、Binary Input オブジェクトと Binary Output オブジェクトの Present_Value の両方の使い方できます。Present_Value はバイナリ値です。

下図に、Binary Value オブジェクトの Present_Value に内部リレーを割り付けて、オフィスの照明を OFF から一時的に ON する場合のイメージを示します。

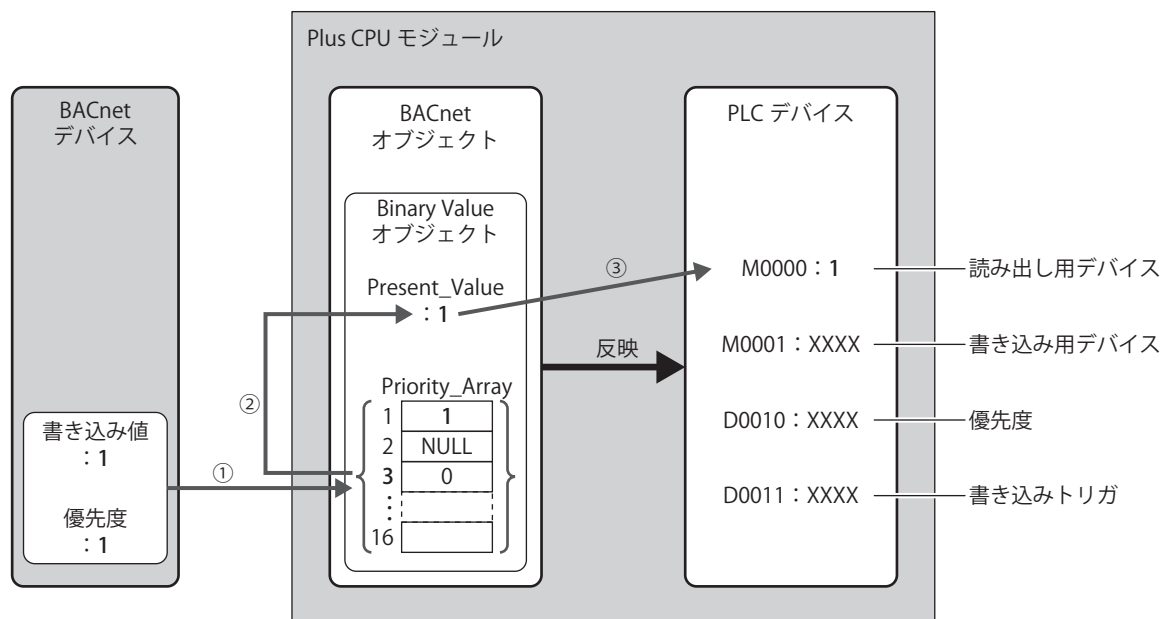
例えば、Present_Value デバイスに M0000、優先度およびトリガの先頭デバイスに D0010 を割り付けた場合、読み出し用デバイス：M0000、書き込み用デバイス：M0001、優先度用デバイス：D0010、書き込みトリガ用デバイス：D0011 が割り付きます。

ラダープログラムから内部リレーの値を Present_Value に書き込む場合




- ① 書き込みトリガ用デバイス (D0011) が 1 のとき、優先度用デバイス (D0010) に格納されたインデックス番号の Priority_Array に、書き込み用デバイス (M0001) の値を書き込みます。
- ② 書き込みトリガ用デバイス (D0011) を自動的に 0 に戻します。
- ③ NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array は、第 1 要素 (1) であるため、1 が Present_Value の値として採用されます。

BACnet デバイスから Present_Value に値を書き込む場合



- ① BACnet デバイスが指定した優先度 (1) が示すインデックス番号の Priority_Array に、書き込み値 (1) を書き込みます。
- ② NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array は、第 1 要素 (1) であるため、1 が Present_Value の値として採用されます。
- ③ Present_Value (1) を読み出し用デバイス (M0000) に書き込みます。

 Present_Value として採用されている Priority_Array の第 1 要素 (1) を NULL に戻すまで、第 1 要素が Present_Value の値として採用されます。第 1 要素を NULL に戻すと、第 3 要素 (0) が Present_Value の値として採用されます。NULL を書き込む方法については、「Binary Value オブジェクト」(15-30 頁) の「②優先度およびトリガの先頭デバイス」(15-31 頁) を参照してください。

Binary Value オブジェクトの Present_Value に割り付けることができるデバイスは次のとおりです。

プロパティ	I	Q	M	R	T	C	D	定数
Present_Value	-	-	○*1	-	-	-	-	-

*1 特殊内部リレーは使用できません。

Plus CPU モジュールで扱う Binary Value オブジェクトの Present_Value には、読み出し用および書き込み用としてそれぞれ内部リレーが必要です。指定した内部リレーを先頭に 2 個の内部リレーを占有します。

デバイス	読み出し	書き込み	合計
内部リレー	1	1	2

Status_Flags

オブジェクトの現在の状態（警報中 / 故障中 / メンテナンス中など）を表すプロパティです。

Status_Flags	値	論理値	条件
IN_ALARM*1	0	FALSE	Event StateがNormalという値を取る場合
	1	TRUE	上記以外
FAULT	0	FALSE	下記以外
	1	TRUE	Reliabilityが存在し、no-fault-detectedという値を取らない場合
OVERRIDDEN*1	0	FALSE	下記以外
	1	TRUE	Present_ValueおよびReliabilityが物理入力の変化に追従していない
OUT_OF_SERVICE	0	FALSE	Out_Of_ServiceがFALSEの場合
	1	TRUE	Out_Of_ServiceがTRUEの場合

*1 Plus CPU モジュールでは常に FALSE です。

Status_Flags はデータレジスタに読み出すことができます。

プロパティ	I	Q	M	R	T	C	D	定数
Status_Flags	—	—	—	—	—	—	○*1	—

*1 特殊データレジスタは使用できません。

データレジスタ内の各フラグの割り当て（ビットアサイン）は、次のとおりです。

ビット	フラグ	値
ビット15~8	リザーブ	不定
ビット7	IN_ALARM	0固定
ビット6	FAULT	0 / 1
ビット5	OVERRIDDEN	0固定
ビット4	OUT_OF_SERVICE	0 / 1
ビット3~0	リザーブ	不定

COV_Increment

Present_Value における最小変化分を表すプロパティです。

最後に送信した COV 通知の Present_Value の値が、COV_Increment で設定した値以上に変わると、COV 通知が送信されます。

COV_Increment はデバイスに読み出すことができません。WindLDR で初期値を設定します。

Priority_Array

Priority_Array は、読み出し専用で、優先順位値が格納された配列を表すプロパティです。

16 個の要素（第 1 要素～第 16 要素）のうち、NULL 以外の値が格納されている最もインデックス番号の小さい Priority_Array に格納された値が Present_Value の値として採用されます。Priority_Array に格納された値がすべて NULL である場合は、Relinquish_Default が Present_Value として採用されます。

Relinquish_Default

Relinquish_Default は、Priority_Array に格納された値がすべて NULL である場合に、Present_Value として採用される値のデフォルト値です。

Polarity

Binary Input および Binary Output オブジェクトにおいて、入出力の物理状態と Present_Value が示す論理状態との関係を表すプロパティです。

Polarity	入出力の物理状態	Present_Value	デバイスの物理状態
NORMAL	OFF/ INACTIVE	INACTIVE	非運転中
NORMAL	ON/ ACTIVE	ACTIVE	運転中
REVERSE	ON/ ACTIVE	INACTIVE	非運転中
REVERSE	OFF/ INACTIVE	ACTIVE	運転中

Out_Of_Service

Out_Of_Service は、Present_Value と物理入出力が切り離されているかどうかを表すプロパティです。

Out_Of_Service が TRUE である場合、メンテナンス中となります。

プロパティ	値	論理値	条件
Out_Of_Service	0	FALSE	サービス中 (Present_Valueと物理入出力が連動している)
	1	TRUE	メンテナンス中 (Present_Valueと物理入出力が切り離されている)

Out_Of_Service を内部リレーに読み出す、または内部リレーの状態を Out_Of_Service として書き込むことができます。

プロパティ	I	Q	M	R	T	C	D	定数
Out_Of_Service	—	—	○*1	—	—	—	—	—

*1 特殊内部リレーは使用できません。



Out_Of_Service = TRUE はシミュレーションのために使用されます。

Reliability

オブジェクトのプロパティの信頼性を表すプロパティです。

下表は各オブジェクトタイプの Reliability の定義について示します。

○：有効、×：無効

定義	値	Analog Input	Analog Output	Analog Value	Binary Input	Binary Output	Binary Value
no-fault-detected	0	○	○	○	○	○	○
no-sensor	1	○	×	×	○	×	×
over-range	2	○	×	○	×	×	×
under-range	3	○	×	○	×	×	×
open-loop	4	○	○	×	○	○	×
shorted-loop	5	○	○	×	○	○	×
no-output	6	×	○	×	×	○	×
unreliable-other	7	○	○	○	○	○	○
process-error	8	×	×	×	×	×	×
multi-state-fault	9	×	×	×	×	×	×
configuration-error	10	×	×	×	×	×	×
-- enumeration value 11 is reserved for a future addendum	11	○	○	×	○	○	○
communication-failure	12	○	○	○	○	○	○
member-fault	13	×	×	×	×	×	×
monitored-object-fault	14	×	×	×	×	×	×
tripped	15	×	×	×	×	×	×

Reliability をデータレジスタに読み出す、またはデータレジスタの値を Reliability として書き込むことができます。

プロパティ	I	Q	M	R	T	C	D	定数
Reliability	—	—	—	—	—	—	○*1	—

*1 特殊データレジスタは使用できません。

System_Status

Plus CPU モジュールの物理状態および論理状態を示します。

パラメータ	値
OPERATIONAL	0
OPERATIONAL_READ_ONLY	1
DOWNLOAD_REQUIRED	2
DOWNLOAD_IN_PROGRESS	3
NON_OPERATIONAL	4
BACKUP_IN_PROGRESS	5

Plus CPU モジュールの System_Status は OPERATIONAL 固定です。

Firmware_Revision

Plus CPU モジュールのシステムファームウェアバージョンがセットされます。

Application_Software_Version

作成したラダープログラムの更新日付など、アプリケーションに応じた情報を、WindLDR で固定の文字列として設定します。

Protocol_Services_Supported

Plus CPU モジュールがサポートしているサービスの種類を示します。

Protocol_Object_Types_Supported

Plus CPU モジュールがサポートしているオブジェクトの種類を示します。

Object_List

作成済みのオブジェクトのリストを示します。

第16章 EtherNet/IP通信

この章では、EtherNet/IP 通信機能について説明します。

概要

Plus CPU モジュールは、EtherNet/IP (スキャナ*) 通信に対応しています。

EtherNet/IP とは、「Ethernet Industrial Protocol」の略で、イーサネットを使用した産業用のマルチベンダネットワークです。

Plus CPU モジュールは EtherNet/IP 通信のスキャナとなり、EtherNet/IP 通信に対応した機器と通信することができます。

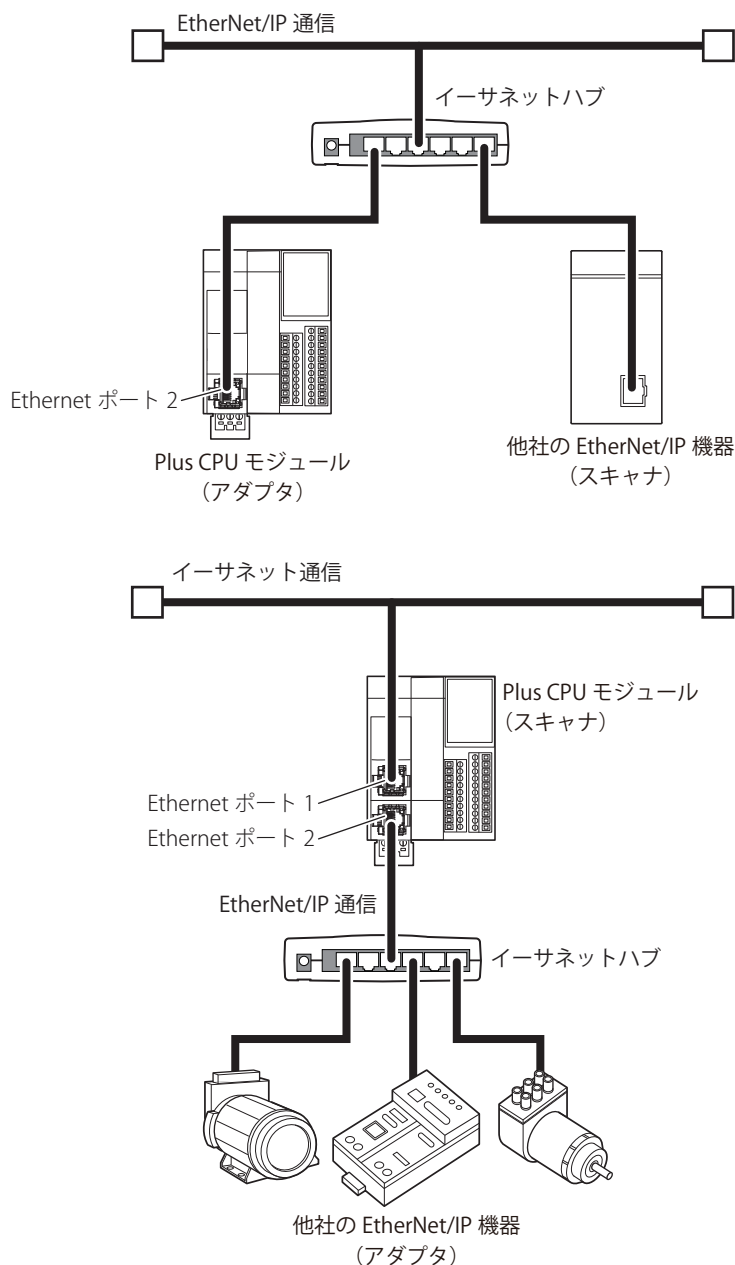
EtherNet/IP 通信は標準のイーサネット技術を使用しているため、イーサネットに対応した様々な機器を混在させてネットワークを構築できます。Plus CPU モジュールはスキャナの機能だけでなく、アダプタ*1の機能も持っており、EtherNet/IP 通信のアダプタとなることもできます。

*1 EtherNet/IP 通信ではマスター側を「スキャナ」、スレーブ側を「アダプタ」と呼びます。一般的に CIP コネクションの開設を要求される機能を持つ EtherNet/IP 機器をアダプタ、さらに要求する機能も併せて持つ EtherNet/IP 機器をスキャナと呼びます。

*2 以降、本書では EtherNet/IP 通信に対応した機器を EtherNet/IP 機器と呼びます。

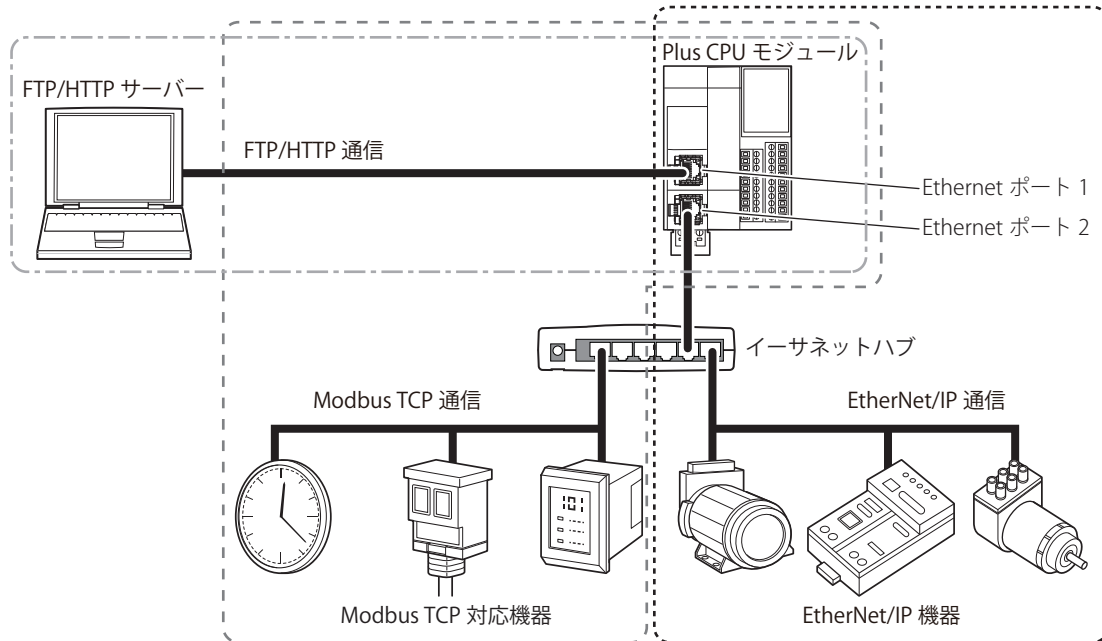
EtherNet/IP 機器間のマルチベンダネットワーク

EtherNet/IP 機器とのマルチベンダネットワークを構築できます。



汎用イーサネット機器と混在したネットワーク

EtherNet/IP 機器と汎用イーサネット機器を混在したネットワークを構築できます。



同じネットワーク内で EtherNet/IP 通信と他の Ethernet 通信を同時に行う場合、負荷状況によって意図したタイミングでデータの送受信ができなくなることがあります。この場合は、QoS 機能に対応したイーサネットスイッチを使用してネットワークを構築し、ネットワーク内に流れるデータの優先度を調整してください。

EtherNet/IP で使用する通信ポート

対応する機種と通信ポートは、次のとおりです。

通信ポート	All-in-One CPU モジュール			CAN J1939 All-in-One CPU モジュール	Plus CPU モジュール	
	16点タイプ	24点タイプ	40点タイプ		Plus16点タイプ	Plus32点タイプ
Ethernet ポート 1	—	—	—	—	—	—
Ethernet ポート 2	—	—	—	—	○	○
HMI-Ethernet ポート	—	—	—	—	—	—

EtherNet/IP 仕様

モジュール		Plus CPU モジュール	
対応ポート		Ethernetポート2	
I/O メッセージ 通信機能	CIP コネクション数		32本*1
	CIP コネクション ポイント	設定数	8個
		事前定義済み	インスタンスID 198 (Input Only出力用) インスタンスID 199 (Listen Only出力用)
	RPI (通信周期)		10~10000ms (1ms単位)
	CIP コネクションタイプ		Input Only/Exclusive Owner/Listen Only
	送信トリガ		サイクリック/COS (Change Of State) *2
	1CIP コネクションあたりの最大データサイズ		504バイトまたは1444バイト*3,*4
	I/O メッセージ 通信許容通信帯域	504 バイト / パケット	600pps
		1444 バイト / パケット	200pps
マルチキャストフィルタ機能*5		あり (IGMPクライアント機能)	
Explicit メッセージ 通信機能	Class 3 (コネクション型)	サーバー	CIPコネクション数：32本*6
		クライアント	非対応
	UCMM	サーバー	同時実行数：32
		クライアント	非対応

*1 Explicit メッセージ通信機能の Class3 (コネクション型) で使用する CIP コネクション数と合計で最大 32 本です。

*2 Change Of State (状態に変化があったときにデータを送信する) 方式でデータを送信する EtherNet/IP 機器と通信できます。

*3 1444 バイトの場合、EtherNet/IP 機器が Large Forward Open (CIP オプション仕様) に対応している必要があります。

*4 I/O メッセージ通信機能で使用する CIP コネクションのデータサイズの合計は 10240 バイト以内にしてください。

*5 IGMP クライアント機能 (IGMPv2) を持つため、IGMP スヌーピングに対応したイーサネットスイッチを使い、不要なマルチキャストパケットをフィルタリングできます。

*6 I/O メッセージ通信機能で使用する CIP コネクション数と合計で最大 32 本となります。

EtherNet/IP について

対応規格

Plus CPU モジュールの EtherNet/IP 通信は、次の規格に対応しています。

- EtherNet/IP

ベンダー ID

ベンダー ID は 159 です。

デバイスプロファイル

Plus CPU モジュールの EtherNet/IP 通信は、スキャナに対応しています。

オブジェクト

Plus CPU モジュールの EtherNet/IP 通信は、次のオブジェクトに対応しています。

オブジェクト名	クラスコード
Identity	01H
Message Router	02H
Assembly	04H
Connection Manage	06H
TCP/IP Interface	F5H
Ethernet Link	F6H

各オブジェクトの詳細については、「オブジェクト」(16-41 頁) を参照してください。

機能

Plus CPU モジュールは、EtherNet/IP 機器のスキャナとして、次の機能を提供します。

- I/O メッセージ通信（オリジネータ / ターゲット）機能（16-5 頁）
- Explicit メッセージ通信（サーバー）機能（16-10 頁）
- データとデバイスの連動機能（16-11 頁）

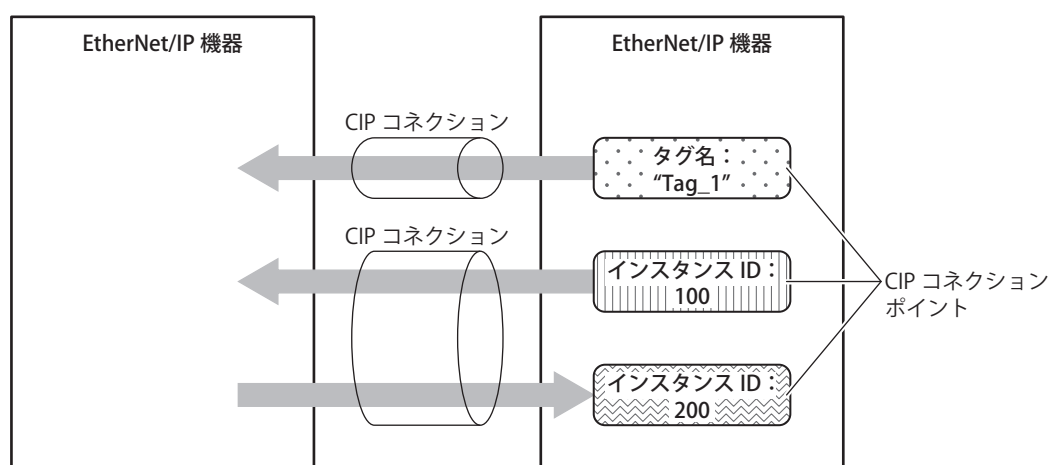
I/O メッセージ通信機能

ここでは、I/O メッセージ通信機能と使用方法について説明します。

■ I/O メッセージ通信とは

EtherNet/IP 機器同士で周期的にデータを読み書きする機能です。データを読み書きする対象の単位を CIP コネクションポイントといいます。CIP コネクションポイントには CIP タグ名またはインスタンス ID が割り付いています。I/O メッセージ通信では、CIP コネクションポイントの CIP タグ名またはインスタンス ID を指定してデータを読み書きします。

I/O メッセージ通信の概念図



■ CIP コネクションとは

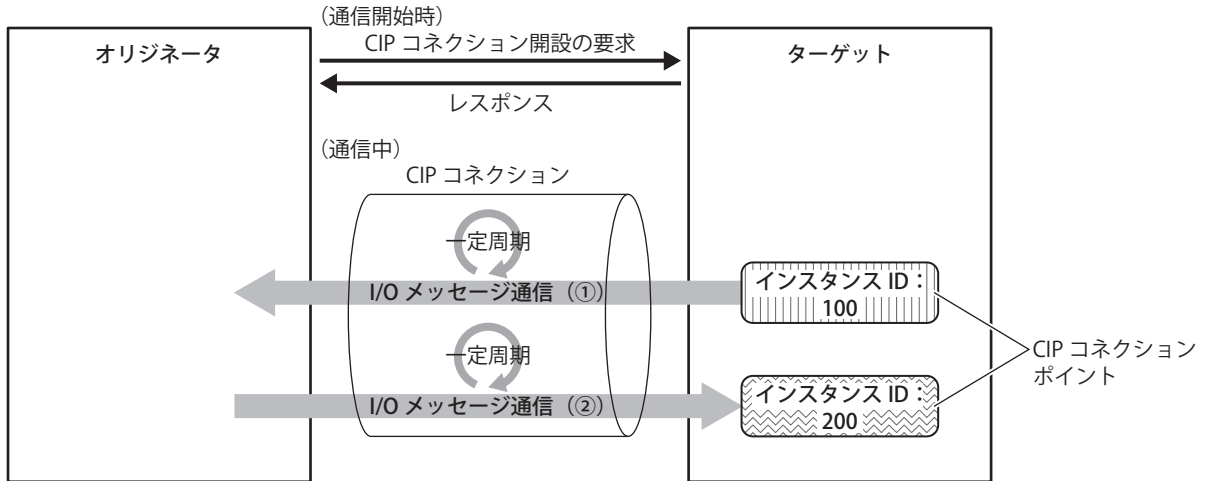
CIP コネクションとは、EtherNet/IP 機器の間の仮想的な通信回線です。I/O メッセージ通信では、一方の機器が相手機器の CIP コネクションポイントに対して CIP コネクションの開設を要求し、成功するとデータを読み書きします。CIP コネクションの開設を要求する側をオリジネータ、要求される側をターゲットと呼びます。Plus CPU モジュールはオリジネータにもターゲットにもなることができます。Plus CPU モジュールは、同時に 32 本の CIP コネクションを開設できます。

■ CIP コネクションの種類

CIP コネクションは次の3種類あります。送受信可能なもの（Exclusive Owner）や受信のみ可能なもの（Input Only、Listen Only）があります。

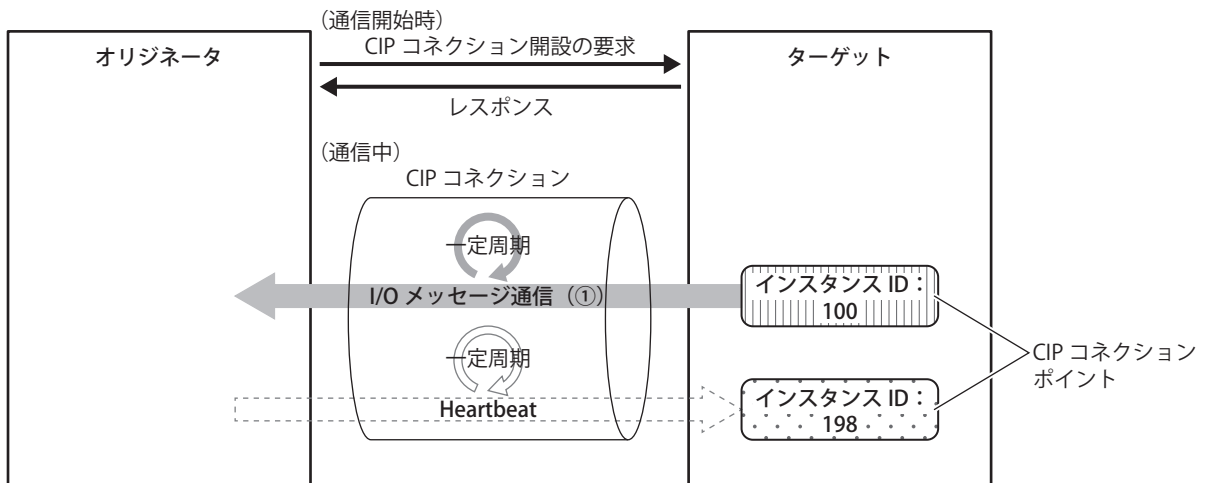
(1) Exclusive Owner

ターゲットからオリジネータへのデータの送信（①）、オリジネータからターゲットへのデータの送信（②）を周期的に行う CIP コネクションです。CIP コネクションポイントは、インスタンス ID または CIP タグ名で指定します。CIP コネクションの開設時に指定した周期（RPI）で通信します。オリジネータがコネクションを開設するとデータを送受信できます。



(2) Input Only

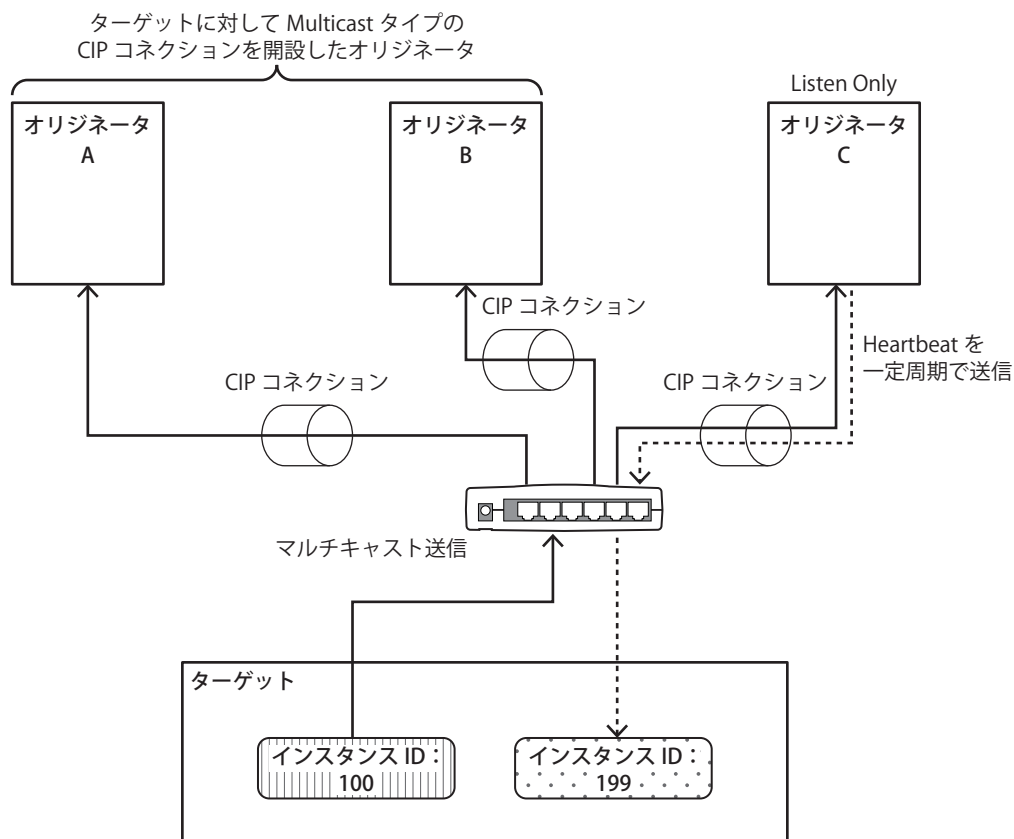
ターゲットからオリジネータへのデータの送信（①）を周期的に行う CIP コネクションです。CIP コネクションポイントは、インスタンス ID または CIP タグ名で指定します。CIP コネクションの開設時に指定した周期（RPI）で通信します。ターゲットがオリジネータの生存監視を行うため、オリジネータからターゲットへデータを含まない Heartbeat を周期的に送信します。オリジネータは Heartbeat の送信先として、ターゲットが持つ Input Only 用のインスタンス ID（インスタンス ID：198）を指定します。Heartbeat は指定した周期（RPI）で行います。



- Input Only では、ターゲットから受信する入力データの CIP コネクションポイントをインスタンス ID で設定する場合、ターゲットへ送信する Heartbeat の CIP コネクションポイントもインスタンス ID で設定してください。
- Input Only では、ターゲットから受信する入力データの CIP コネクションポイントを CIP タグ名で設定する場合、ターゲットへ送信する出力データの CIP コネクションポイントの CIP タグ名は存在しないため設定しません。
- Input Only では、必ず Heartbeat が行われます。

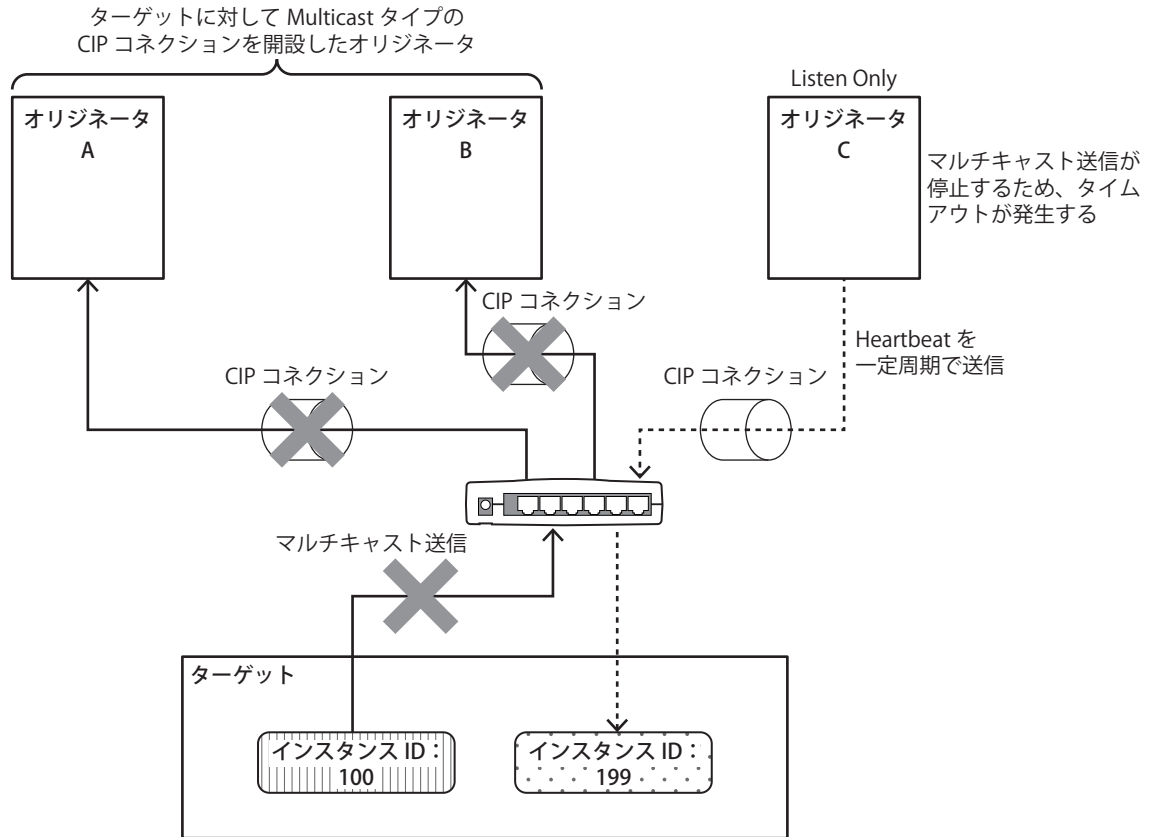
(3) Listen Only

Exclusive Owner や Input Only を使用した Multicast タイプの CIP コネクションが既に開設されているターゲットからオリジネータへのデータ受信を周期的に行う CIP コネクションです。ターゲットが他のオリジネータに対してデータをマルチキャスト送信しているときに、そのデータを同時に受信する場合に設定します。CIP コネクションポイントは、インスタンス ID で指定します。CIP コネクションの開設時に指定した周期 (RPI) で通信します。ターゲットがオリジネータの生存監視を行うため、オリジネータからターゲットへデータを含まない Heartbeat を周期的に送信します。オリジネータは Heartbeat の送信先として、ターゲットが持つ Listen Only 用のインスタンス ID (インスタンス ID : 199) を指定します。Heartbeat は指定した周期 (RPI) で行います。



- Listen Only では、ターゲットから受信する入力データの CIP コネクションポイントをインスタンス ID で設定する場合、ターゲットへ送信する出力データの CIP コネクションポイントもインスタンス ID で設定してください。
- Listen Only では、必ず Heartbeat が行われます。
- IGMP スヌーピングに対応したイーサネットスイッチを使い、不要なマルチキャストパケットをフィルタリングできます。この場合、IGMP クライアント機能に対応したオリジネータと、IGMP クエリを周期送信する IGMP クエリア (ルータ等) を設置してください。IGMP クエリアがないと、一定時間経過後にオリジネータがマルチキャストパケットを受信できなくなります。

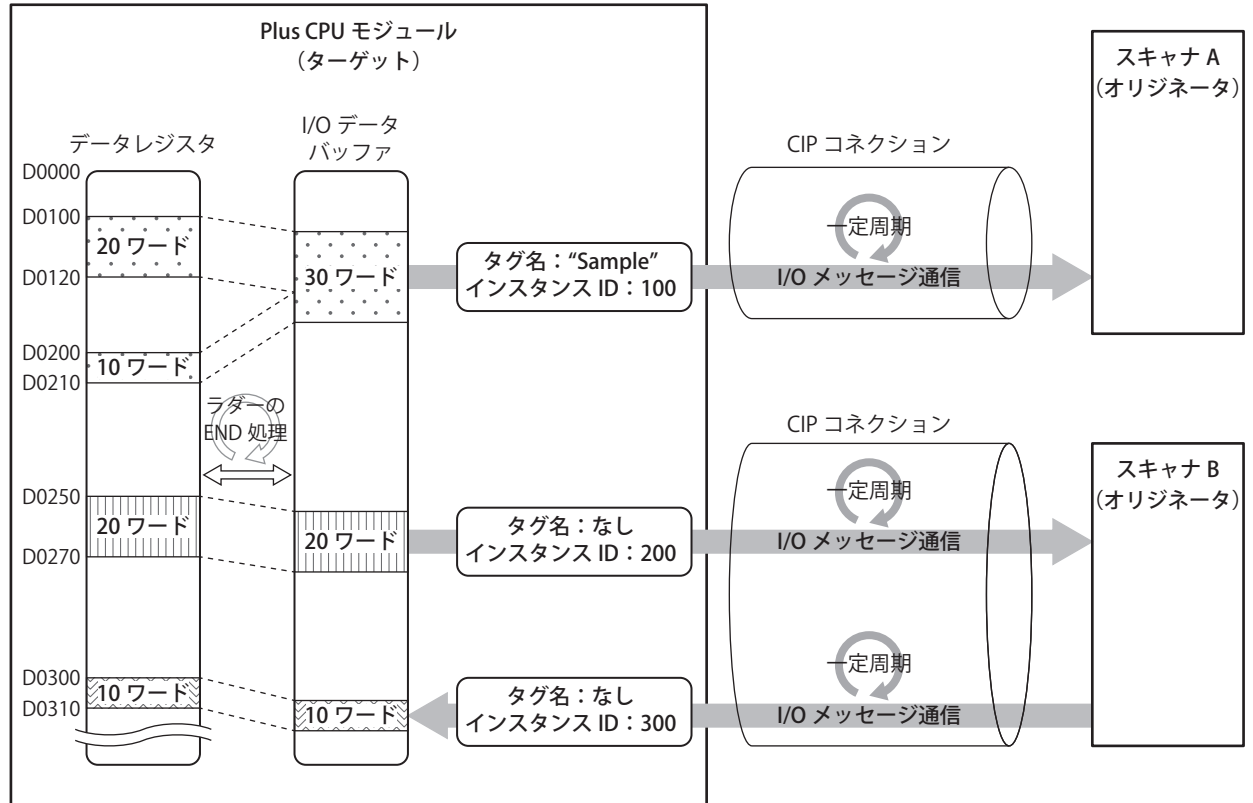
Multicast タイプの CIP コネクションが開設されていない状態では Listen Only の CIP コネクションは開設できません。また Exclusive Owner や Input Only を使用した Multicast タイプの CIP コネクションを開設しているオリジネータとターゲットの CIP コネクションが切断されると、ターゲットから Listen Only の CIP コネクションを開設しているオリジネータへのマルチキャスト送信が停止します。



I/O メッセージ通信（ターゲット）機能

Plus CPU モジュールがターゲットとして CIP コネクションポイントを持ち、オリジネータが周期的にデータを読み書きする機能です。定義できる CIP コネクションポイントは最大 8 個です。

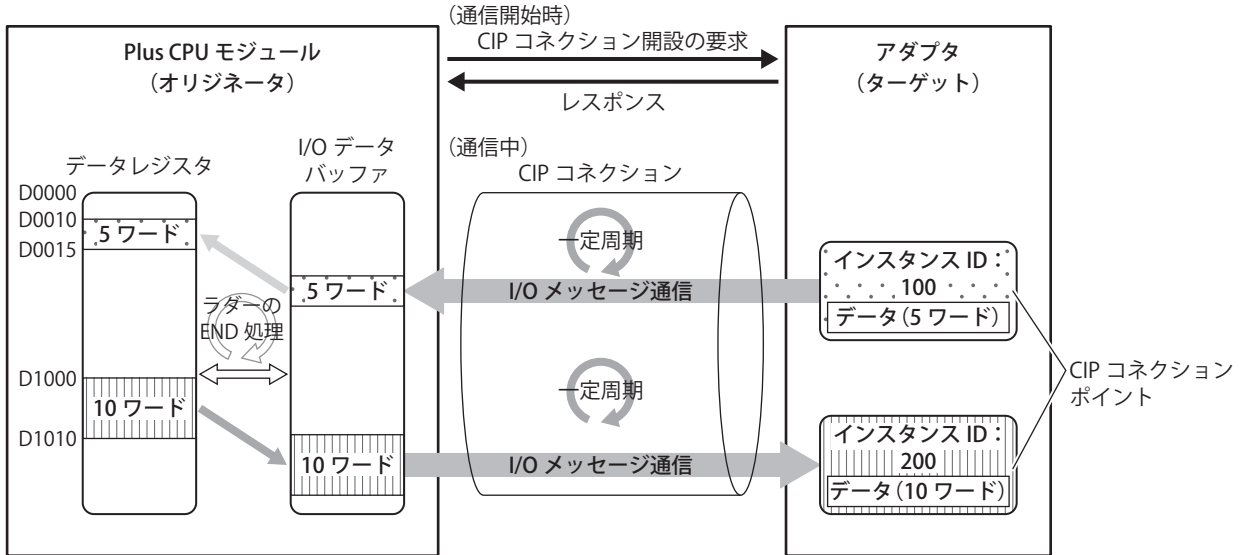
オリジネータが Plus CPU モジュールの CIP コネクションポイントに対して、CIP コネクションの開設を要求し、成功すると CIP コネクションポイントのデータを読み書きします。Plus CPU モジュールは、データレジスタの値を I/O データバッファ*¹ に書き込んでからオリジネータへ送信します。またオリジネータから受信したデータを I/O データバッファに書き込んでからデータレジスタに格納します。



*1 I/O データバッファとは、Plus CPU モジュールが内部に持つ、I/O メッセージ通信で送受信するデータを格納するバッファです。I/O データバッファとデータレジスタの連動についての詳細は、「データとデバイスの連動機能」(16-11 頁)を参照してください。

I/O メッセージ通信（オリジネータ）機能

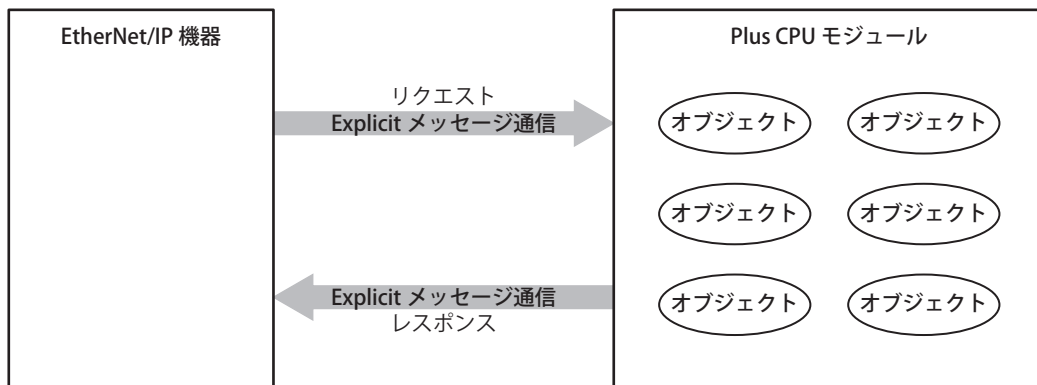
Plus CPU モジュールが、ターゲットの CIP コネクションポイントのデータを周期的に読み書きする機能です。EtherNet/IP 通信許可 (M8460) を ON すると、Plus CPU モジュールはスキャンリスト*1 に登録されたターゲットの CIP コネクションポイントに対して、CIP コネクションの開設を要求し、成功すると*2 CIP コネクションポイントのデータを読み書きします。Plus CPU モジュールは、ターゲットから受信したデータを、Plus CPU モジュール内部の I/O データバッファ*3 に書き込んでからデータレジスタに格納します。またデータレジスタの値を I/O データバッファに書き込んでからターゲットへ送信します。



- *1 スキャンリストとは、オリジネータである Plus CPU モジュールが開設する CIP コネクションの一覧です。[EtherNet/IP 設定] ダイアログボックスの EtherNet/IP ツリーエリアに、設定済みのターゲットまたは CIP コネクションの一覧が表示されます。詳細は、「EtherNet/IP ツリーエリア」(16-20 頁) を参照してください。
- *2 CIP コネクションの開設に失敗した場合、成功するまでリトライを続けます。I/O メッセージ通信中、指定したタイムアウト時間が経過してもターゲットから I/O メッセージを受信できなかった場合、自動的に新しい CIP コネクションを開設し直します。新しい CIP コネクションの開設に失敗した場合、成功するまでリトライを続けます。
- *3 I/O データバッファとは、Plus CPU モジュールが内部に持つ、I/O メッセージ通信で送受信するデータを格納するバッファです。I/O データバッファとデータレジスタの連動については、「データとデバイスの連動機能」(16-11 頁) を参照してください。

Explicit メッセージ通信（サーバー）機能

他の EtherNet/IP 機器から、Plus CPU モジュールのオブジェクト*1 が対応するサービスを実行するための機能です。

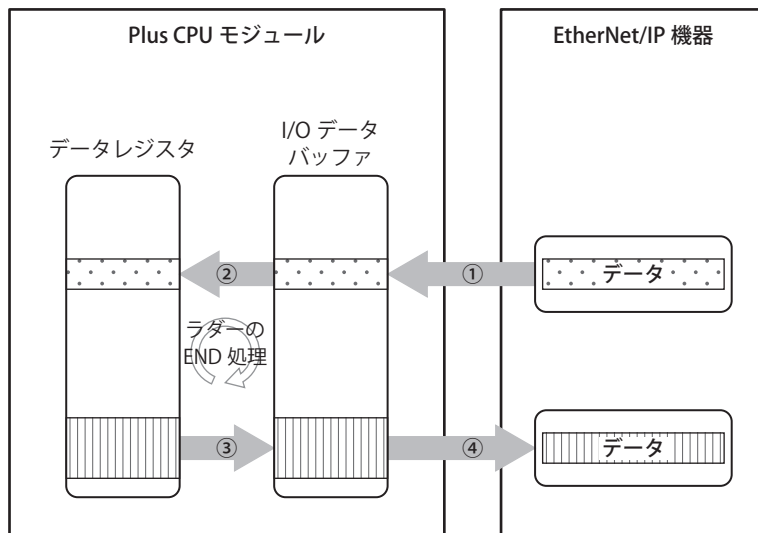


- *1 オブジェクトとは、EtherNet/IP 機器の機能やデータを抽象的に表現したものです。

データとデバイスの連動機能

Plus CPU モジュールは、スキャンリストおよび CIP コネクションポイントで設定された IN データ*¹ および OUT データ*² を、Plus CPU モジュール内部の I/O データバッファに割り付けます。

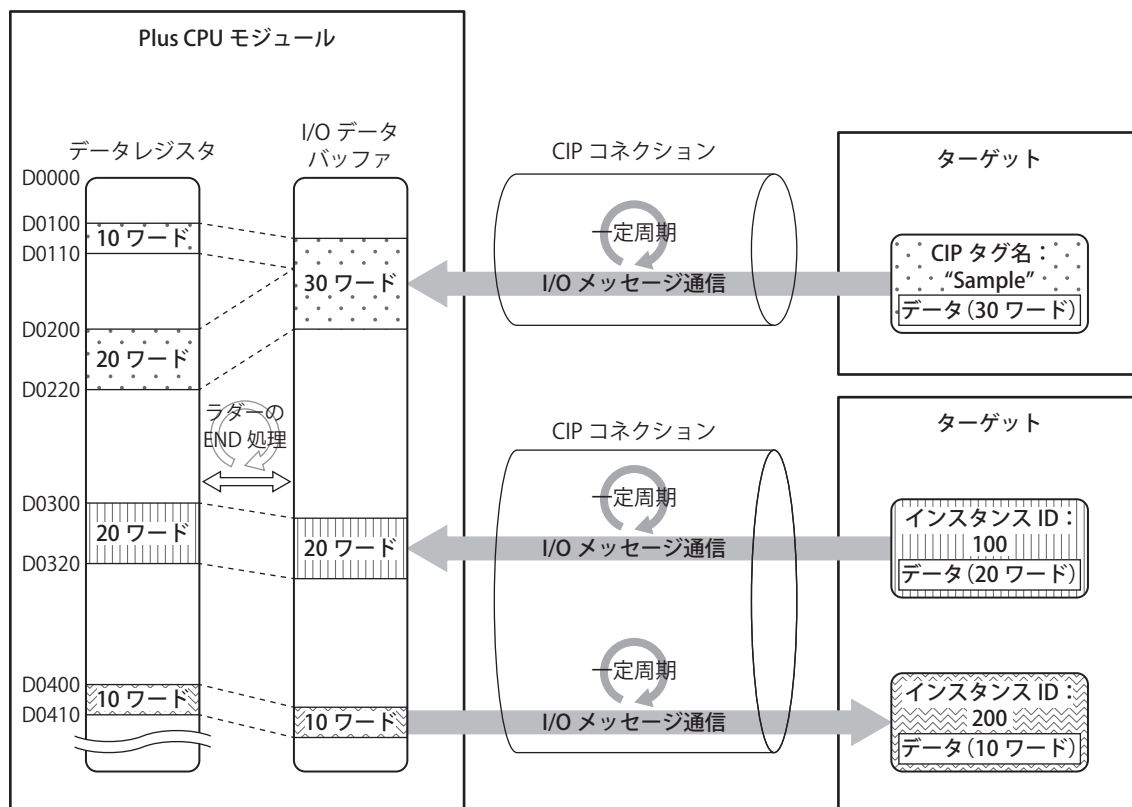
Plus CPU モジュールは、I/O メッセージ通信で相手機器から受信したデータを、いったん I/O データバッファに書き込んでから (①)、データレジスタに格納します (②)。またデータレジスタに格納した値をいったん I/O データバッファに書き込んでから (③)、相手機器へ送信します (④)。I/O データバッファとデータレジスタを相互に反映する処理は、ラダープログラムの END 処理で行われます。



*1 IN データとは、オリジネータが、I/O メッセージ通信でターゲットから受信したデータです。

*2 OUT データとは、オリジネータが、I/O メッセージ通信でターゲットへ送信するデータです。

例えば、Plus CPU モジュール (オリジネータ) が、ターゲットの CIP コネクションポイントのデータを周期的に読み書きする場合、ターゲットから受信したデータはいったん I/O データバッファに書き込まれてからデータレジスタに格納されます。またデータレジスタの値は I/O データバッファに書き込まれてからターゲットへ送信されます。



■ リフレッシュ上限数

1回のEND処理で、I/Oデータバッファとデータレジスタを相互に反映できるサイズは最大400ワードです。このサイズは1～400ワードの範囲でリフレッシュ上限数として設定できます。例えばリフレッシュ上限数を400ワードに設定した場合、1回のEND処理でI/Oデータバッファからデータレジスタへ最大256ワード、データレジスタからI/Oデータバッファへ最大256ワードの反映を行います。リフレッシュ上限数を超えるINデータおよびOUTデータを設定した場合、複数のEND処理に分割して処理されます。リフレッシュ上限数を大きくすると、END処理で反映するデータ量が多くなるため、ラダープログラムのスキャンタイムが長くなります。対象のシステムに合わせて調整してください。



I/Oメッセージ通信は高頻度で行われます。EtherNet/IP機器から受信した値がスキャンタイムよりも短い間隔で変化した場合、その変化がデータレジスタに反映されないことがあります。


EtherNet/IP 通信の動作


EtherNet/IP 通信で使用する特殊デバイス

EtherNet/IP 通信で使用する特殊データレジスタおよび特殊内部リレーについて示します。

特殊データレジスタ	内容		R/W
D8790	EtherNet/IP運転状態	EtherNet/IP通信の動作状態を格納します。 0000h：停止中 0100h：オンライン準備中 0200h：オンライン中 0300h：再起動中 0F00h：エラー停止中	R
D8791	EtherNet/IPエラー情報	EtherNet/IP通信で発生したエラーを格納します。 0000h：正常 0800h：リンクアップ待ちタイムアウト 8000h：通信開始待ちタイムアウト 8100h：通信停止待ちタイムアウト EtherNet/IP通信がオンライン準備中 ((D8790)=0100h) のとき、0にリセットされます。EtherNet/IP通信で複数のエラーが発生した場合、最後に発生したエラー情報が格納されます。	R

特殊内部リレー	内容		R/W
M8460	EtherNet/IP通信許可	EtherNet/IP通信を許可 / 禁止します。 OFF：EtherNet/IP通信禁止 ON：EtherNet/IP通信許可 RUN 状態でM8460をOFFからONにすると、Plus CPUモジュールは相手機器とCIP接続を確立し、EtherNet/IP通信を開始します。M8460をOFFにすると、CIP接続を切断し、EtherNet/IP通信を停止します。RUN→STOPのとき、M8460は自動的にOFFになります。	R/W

 EtherNet/IP 運転状態 (D8790) がエラー停止中のときに EtherNet/IP 通信を再開する場合、いったん M8460 を OFF したあと、EtherNet/IP 運転状態 (D8790) が停止中になるのを待ってから、M8460 を ON してください。

 Plus CPU モジュールの電源 ON 時および RUN 開始時、EtherNet/IP 通信許可 (M8460) は OFF しています。EtherNet/IP 通信を開始するためには、ラダープログラムで M8460 を ON するか、または [EtherNet/IP 設定] ダイアログボックスの [EtherNet/IP 通信許可を自動的に ON する] チェックボックスをオンにしたユーザープログラムをダウンロードして RUN してください。

基本動作

ここでは、EtherNet/IP 通信の基本的な動作について説明します。

■ EtherNet/IP 通信の開始と停止

RUN 状態で EtherNet/IP 通信許可 (M8460) を OFF から ON にすると、Plus CPU モジュールは、スキャンリストに登録されたターゲットまたは CIP コネクションポイントと CIP コネクションを確立し、EtherNet/IP 通信を開始します (I/O メッセージ通信による周期的なデータ交換を開始します)。

EtherNet/IP 通信許可 (M8460) を OFF にすると、EtherNet/IP 通信を停止します (I/O メッセージ通信を停止し、すべての CIP コネクションを切断します)。RUN → STOP すると、EtherNet/IP 通信許可 (M8460) は自動的に OFF になります。

■ EtherNet/IP 通信中の動作

Plus CPU モジュールが持つ I/O データバッファは、データレジスタと対応付けられており、ラダープログラムの END 処理で相互にデータを反映します。

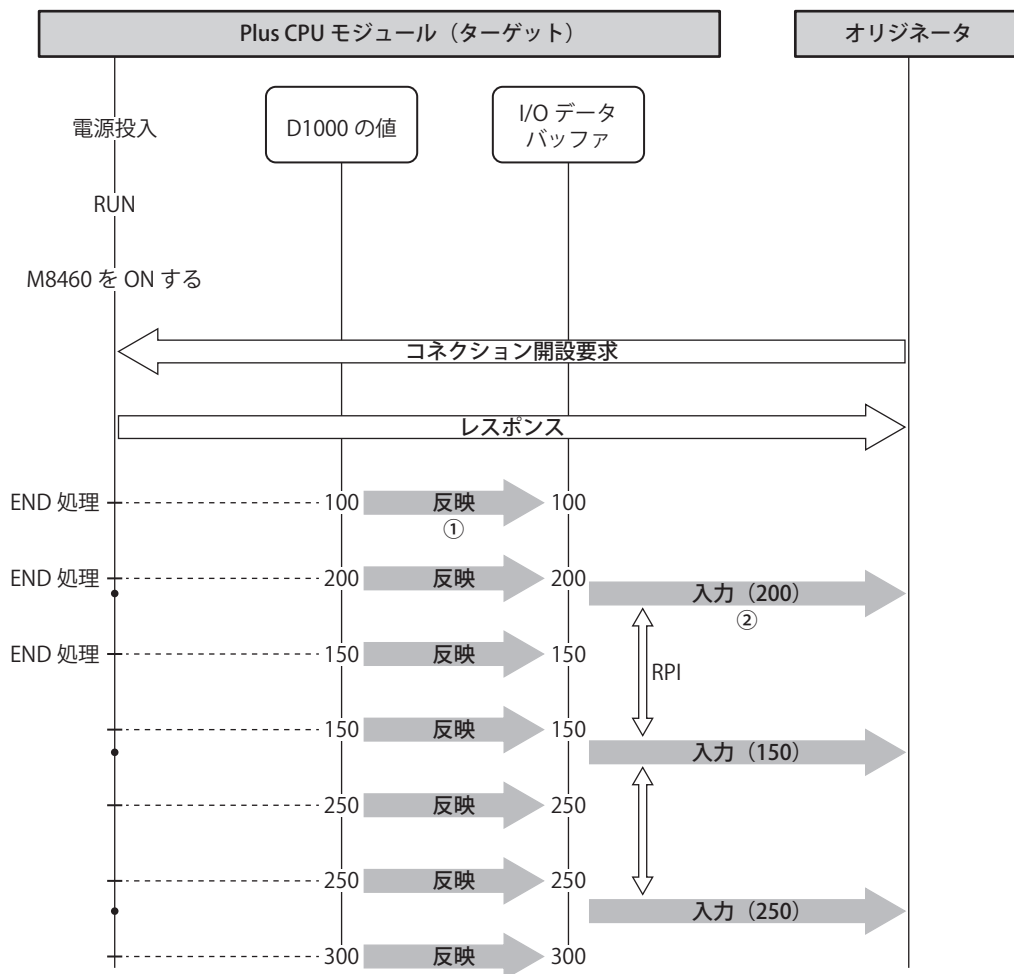
I/O データバッファとデータレジスタを相互に反映する処理と、Plus CPU モジュールと EtherNet/IP 機器間の I/O メッセージ通信は、非同期で行われます。



I/O メッセージ通信は高頻度で行われるため、EtherNet/IP 機器から受信した値がスキャンタイムよりも短い間隔で変化した場合、その変化がデータレジスタに反映されないことがあります。

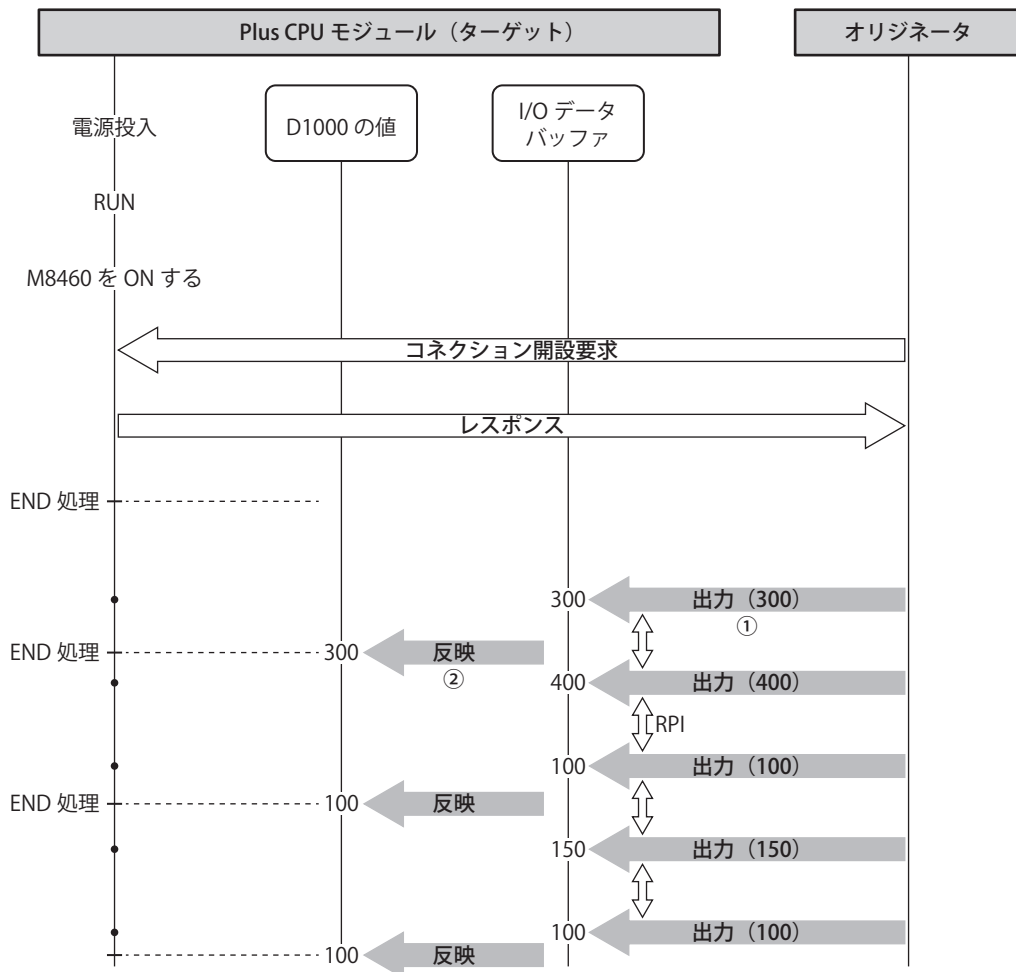
Plus CPU モジュール (ターゲット) がオリジネータへデータを送信 (入力) する場合

例えば、D1000 に格納された値をオリジネータに送信する場合、データを送信する直前のラダープログラムの END 処理で、D1000 に格納された値を I/O データバッファに反映します (①)。そして、その直後の I/O メッセージ通信で I/O データバッファのデータをターゲットへ送信します (②)。



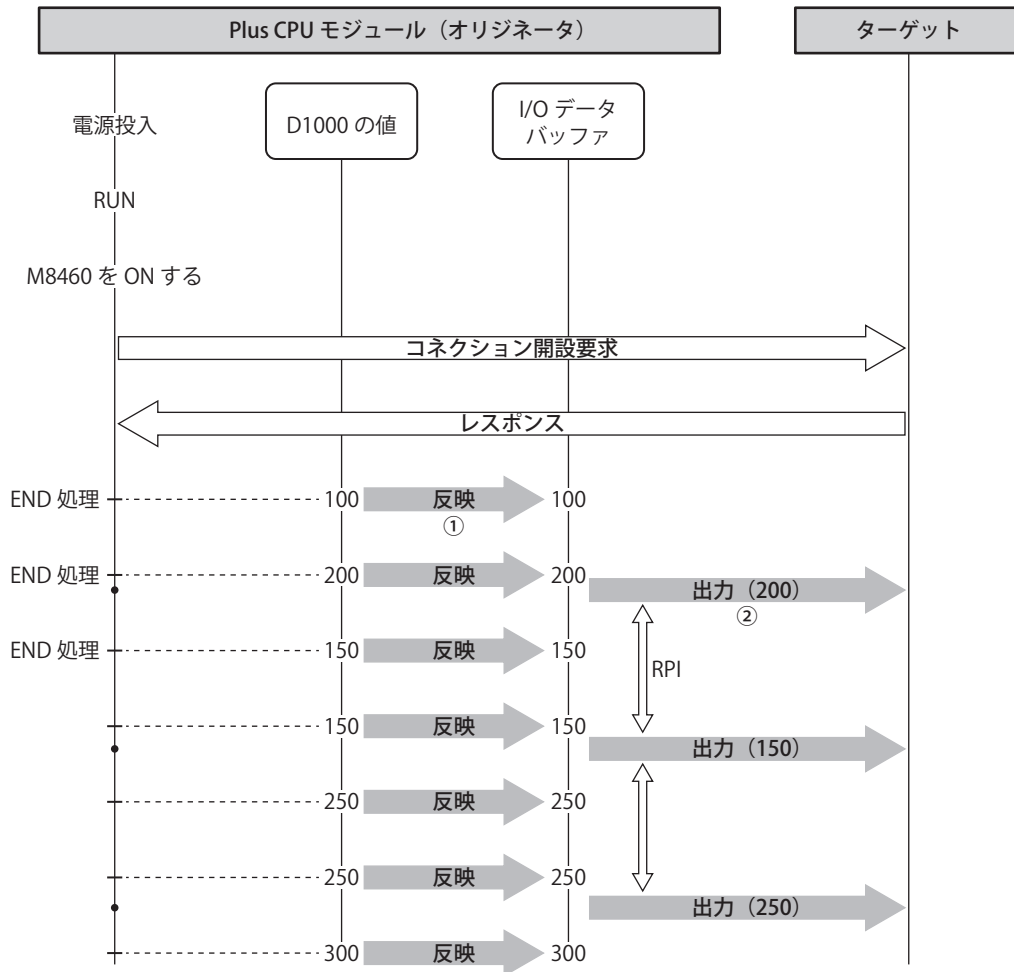
Plus CPU モジュール（ターゲット）がオリジネータからデータを受信（出力）する場合

例えば、オリジネータから受信したデータを D1000 に格納する場合、Plus CPU モジュールは I/O メッセージ通信でオリジネータから受信したデータを、いったん I/O データバッファに書き込みます (①)。そして、データを受信した直後のラダープログラムの END 処理で、I/O データバッファのデータをデータレジスタに反映します (②)。



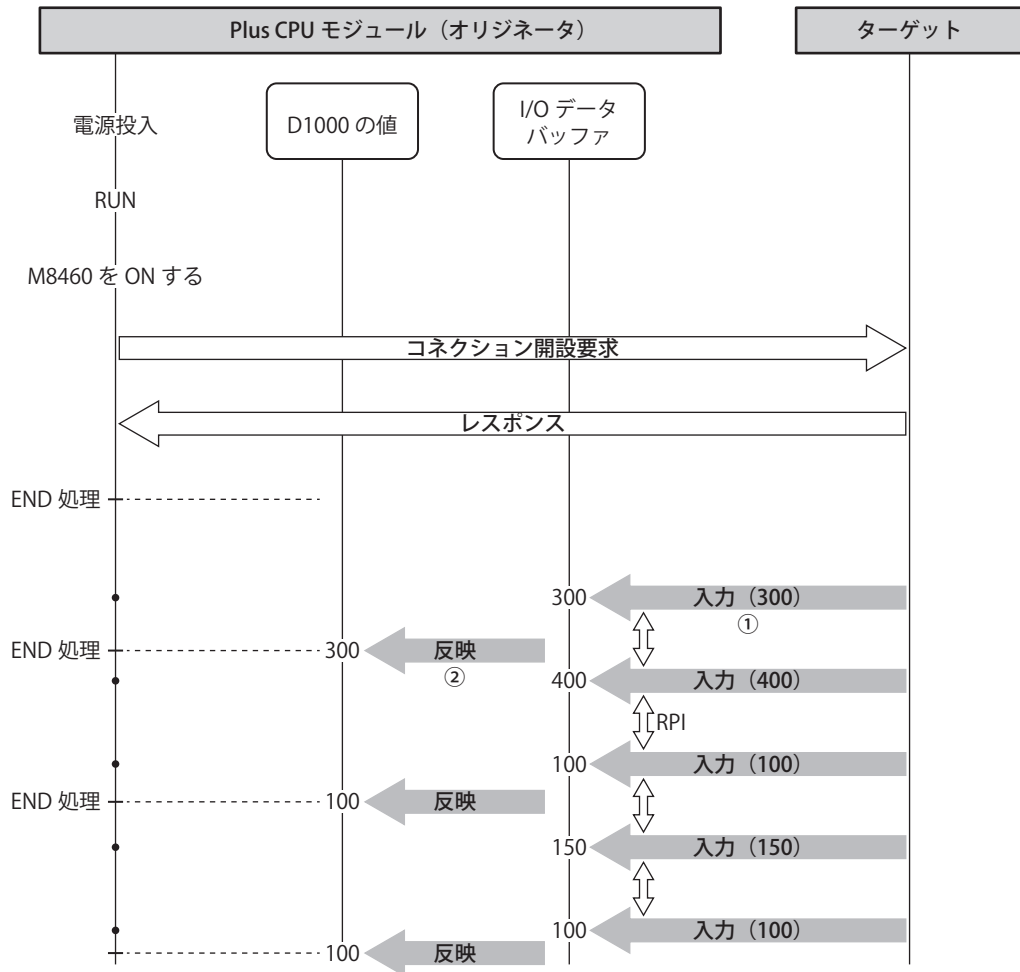
Plus CPU モジュール（オリジネータ）がターゲットへデータを送信（出力）する場合

例えば、D1000 に格納された値をターゲットに送信する場合、データを送信する直前のラダープログラムの END 処理で、D1000 に格納された値を I/O データバッファに反映します (①)。そして、その直後の I/O メッセージ通信で I/O データバッファのデータをターゲットへ送信します (②)。



Plus CPU モジュール（オリジネータ）がターゲットからデータを受信（入力）する場合

例えば、ターゲットから受信したデータを D1000 に格納する場合、Plus CPU モジュールは I/O メッセージ通信でターゲットから受信したデータを、いったん I/O データバッファに書き込みます (①)。そして、データを受信した直後のラダープログラムの END 処理で、I/O データバッファのデータをデータレジスタに反映します (②)。



RUN および STOP 中の動作

RUN および STOP 中の EtherNet/IP 通信の状態を示します。

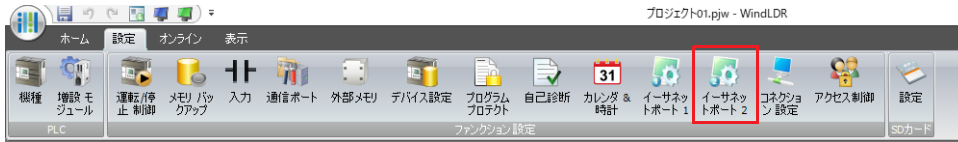
RUN/STOP 状態	M8460 の状態	EtherNet/IP 通信
STOP	OFF	停止
	ON	
RUN	OFF	開始
	OFF → ON	通信中
	ON → OFF	終了

EtherNet/IP 通信の設定

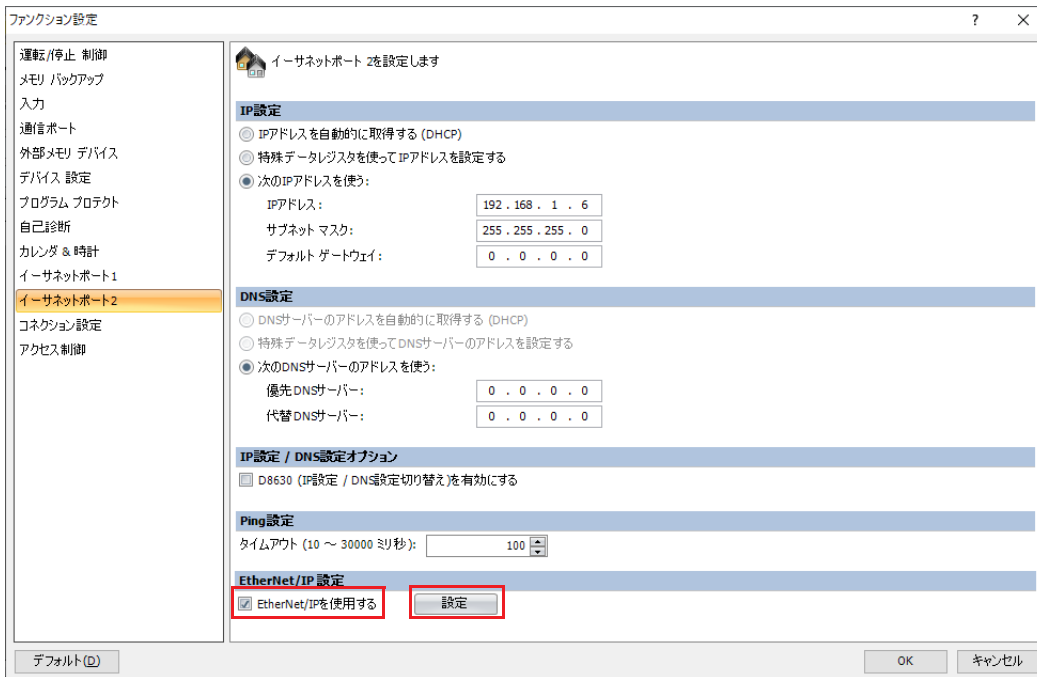
EtherNet/IP 通信の設定は [EtherNet/IP 設定] ダイアログボックスで行います。ここでは [EtherNet/IP 設定] タブと使用方法に応じた設定について説明します。

● [EtherNet/IP 設定] ダイアログボックスの表示手順

1. [設定] タブの [ファンクション設定] で [イーサネットポート 2] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。



2. EtherNet/IP 設定で [EtherNet/IP を使用する] チェックボックスをオンにし、[設定] ボタンをクリックします。
[EtherNet/IP 設定] ダイアログボックスが表示されます。



EtherNet/IP 設定の概要

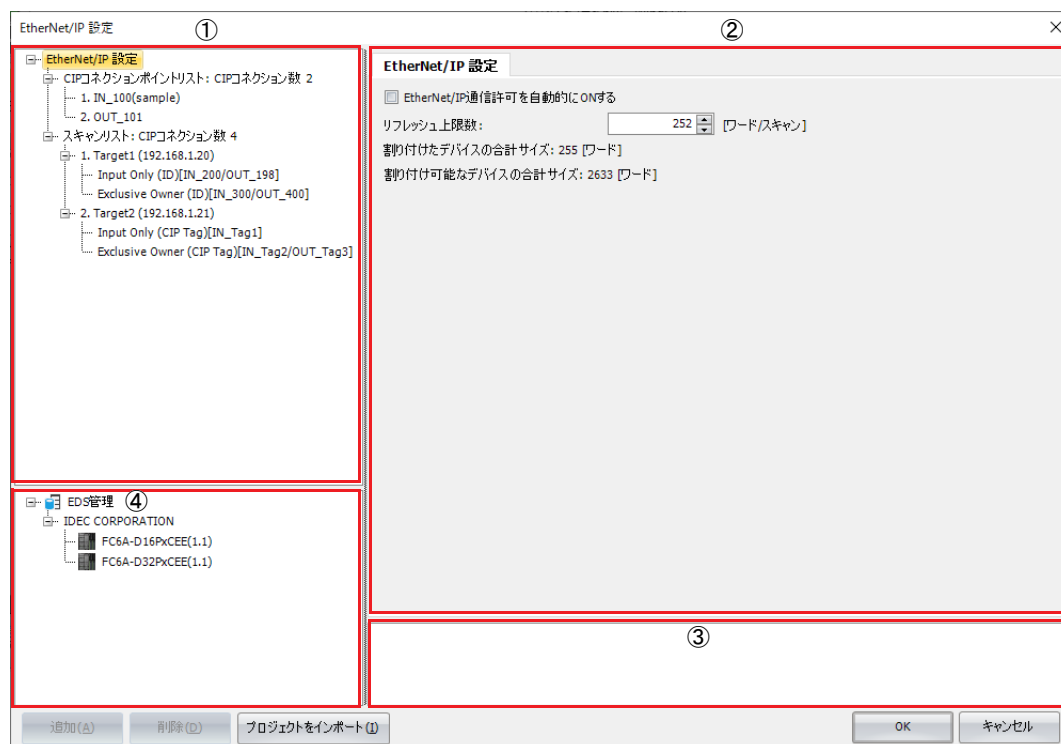
EtherNet/IP 通信を行うために必要な設定と関連する WindLDR の設定ダイアログボックスは、次のとおりです。

内容	必要な設定	WindLDR の設定ダイアログボックス	使用できる機能
EtherNet/IP通信を行う	EtherNet/IP通信の基本設定	EtherNet/IP設定	EtherNet/IP通信共通
Plus CPUモジュールをターゲットとして使用する	CIPコネクションポイントの設定 ・ CIP タグ名またはインスタンス ID の設定 ・ デバイスの割り付け	CIPコネクションポイント設定	I/Oメッセージ通信 (ターゲット) 機能
Plus CPUモジュールをオリジネータとして使用する	スキャンリストの作成 ・ ターゲットの設定 ・ CIP コネクションの設定	ターゲット設定 CIPコネクション設定	I/Oメッセージ通信 (オリジネータ) 機能

[EtherNet/IP 設定] ダイアログボックス

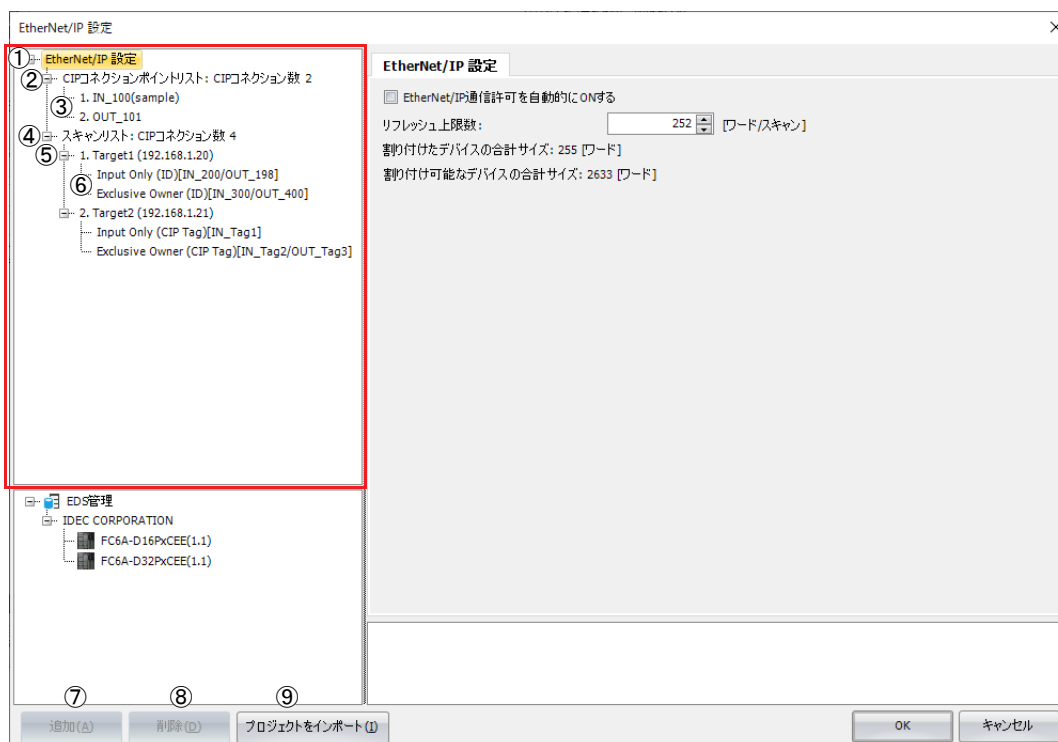
[EtherNet/IP 設定] ダイアログボックスは次の3つのエリアで構成されます。

- ① EtherNet/IP ツリーエリア (16-20 頁)
設定済みの CIP コネクションポイント、ターゲットおよび CIP コネクションの一覧が表示されます。
- ② パラメータ設定エリア (16-22 頁)
EtherNet/IP ツリーエリアで選択しているノードの詳細が表示されます。
- ③ 情報表示エリア (16-30 頁)
[EtherNet/IP 設定] ダイアログボックスで設定した内容にエラーがある場合、エラーの内容が表示されます。
- ④ EDS 管理エリア (16-31 頁)
パソコンにインポートされた EDS ファイルの一覧が表示されます。



EtherNet/IP ツリーエリア

設定済みの CIP コネクションポイント、ターゲットおよび CIP コネクションの一覧が表示されます。



- ① [EtherNet/IP 設定] ノード
[EtherNet/IP 設定] ノードを選択して展開すると、設定済みの CIP コネクションポイントとスキャンリストの一覧を表示します。
- ② [CIP コネクションポイントリスト] ノード
CIP コネクションポイント数が表示されます。[CIP コネクションポイントリスト] ノードを展開すると、設定済みの CIP コネクションポイントの一覧が表示されます。
- ③ [CIP コネクションポイント] ノード
設定済みの CIP コネクションポイントの CIP タグ名およびインスタンス ID が表示されます。
- ④ [スキャンリスト] ノード
CIP コネクション数が表示されます。[スキャンリスト] ノードを展開すると、設定済みのターゲットの一覧が表示されます。
- ⑤ [ターゲット] ノード
設定済みのターゲットのノード名および IP アドレスが表示されます。各 [ターゲット] ノードを選択して展開すると、設定済みの CIP コネクションの一覧が表示されます。
- ⑥ [CIP コネクション] ノード
設定済みの CIP コネクションの CIP コネクション名および CIP コネクションポイントが表示されます。
- ⑦ [追加] ボタン
EtherNet/IP ツリーエリアで選択しているノードに応じて、ノードを新規に追加します。ノードを右クリックしたときに表示されるメニューからも追加できます。
- ⑧ [削除] ボタン
EtherNet/IP ツリーエリアで選択しているノードを削除します。ノードを右クリックしたときに表示されるメニューからも削除できます。
- ⑨ [プロジェクトをインポート] ボタン
WindLDR のプロジェクトファイルから、EtherNet/IP 設定のみをインポートします。

EtherNet/IP ツリーエリアの各ノードを右クリックして表示されるコンテキストメニューの動作を説明します。

ノード	コンテキストメニュー	動作
EtherNet/IP 設定	—	—
CIP コネクションポイントリスト	CIP コネクションポイント設定の追加	CIP コネクションポイントを1つ追加します。
	CIP コネクションポイント設定の削除	すべてのCIP コネクションポイントを削除します。
CIP コネクションポイント	CIP コネクションポイント設定の追加	CIP コネクションポイントを1つ追加します。
	選択しているCIP コネクションポイント設定の削除	選択しているCIP コネクションポイントを削除します。
スキャンリスト	ターゲットの追加	ターゲットを1つ追加します。
	ターゲットの削除	すべてのターゲットを削除します。
ターゲット	CIP コネクションの追加	CIP コネクションを1つ追加します。
	選択しているターゲットの削除	選択しているターゲットを削除します。
	EDS ファイルを 'EDS 管理' にインポート	選択しているターゲットの EDS ファイルを EDS 管理エリアにインポートします。EDS ファイルはインポートすれば、他のプロジェクトでも使用できるようになります。
CIP コネクション	CIP コネクションの追加	CIP コネクションを1つ追加します。
	選択しているCIP コネクションの削除	選択しているCIP コネクションを削除します。

パラメータ設定エリア

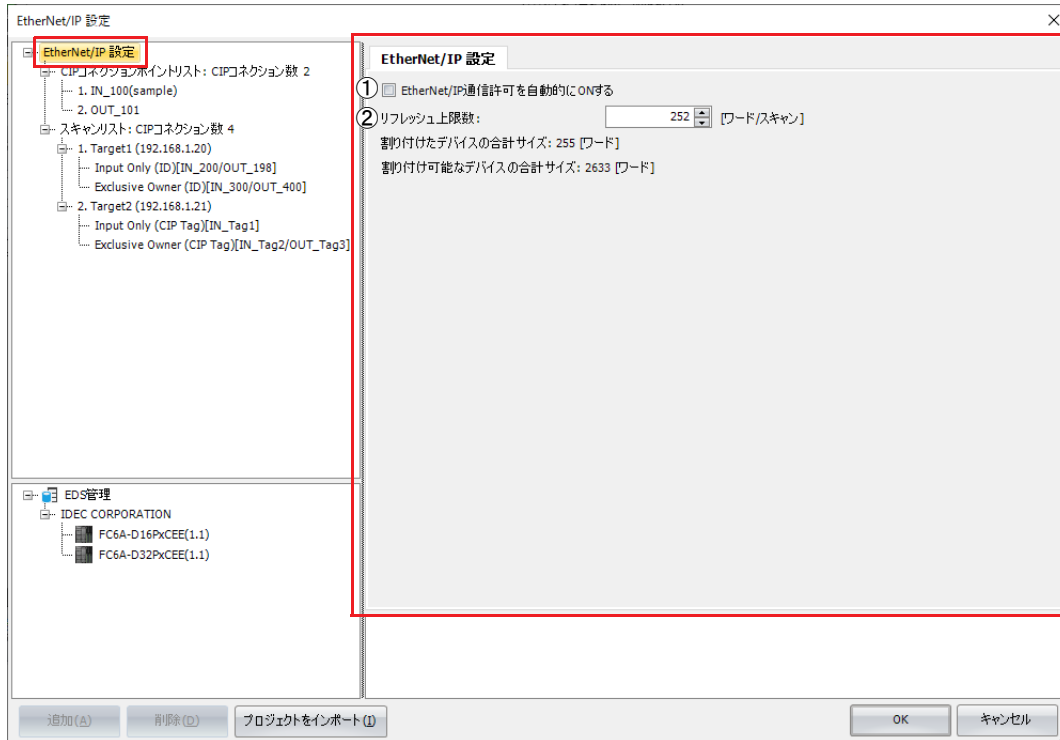
EtherNet/IP ツリーエリアで選択しているノードの詳細が表示されます。

[EtherNet/IP 設定] ノードを選択している場合

EtherNet/IP ツリーエリアの [EtherNet/IP 設定] ノードを選択すると、パラメータ設定エリアに [EtherNet/IP 設定] タブが表示されます。

■ [EtherNet/IP 設定] タブ

EtherNet/IP 通信を行うための基本設定を行います。



① EtherNet/IP 通信許可を自動的に ON する

STOP → RUN で EtherNet/IP 通信許可 (M8460) を自動的に ON するかどうかを設定します。デフォルト値はオフです。オフの場合は、EtherNet/IP 通信許可を自動的に ON しません。

設定方法	内容
オン	STOP→RUNでEtherNet/IP通信許可 (M8460) を自動的にONします。 EtherNet/IP通信許可 (M8460) をONすると、Plus CPUモジュールはスキャンリストに登録された内容にしたがって、相手機器とEtherNet/IP通信を行います。
オフ	EtherNet/IP通信許可 (M8460) を自動的にONしません。

② リフレッシュ上限数

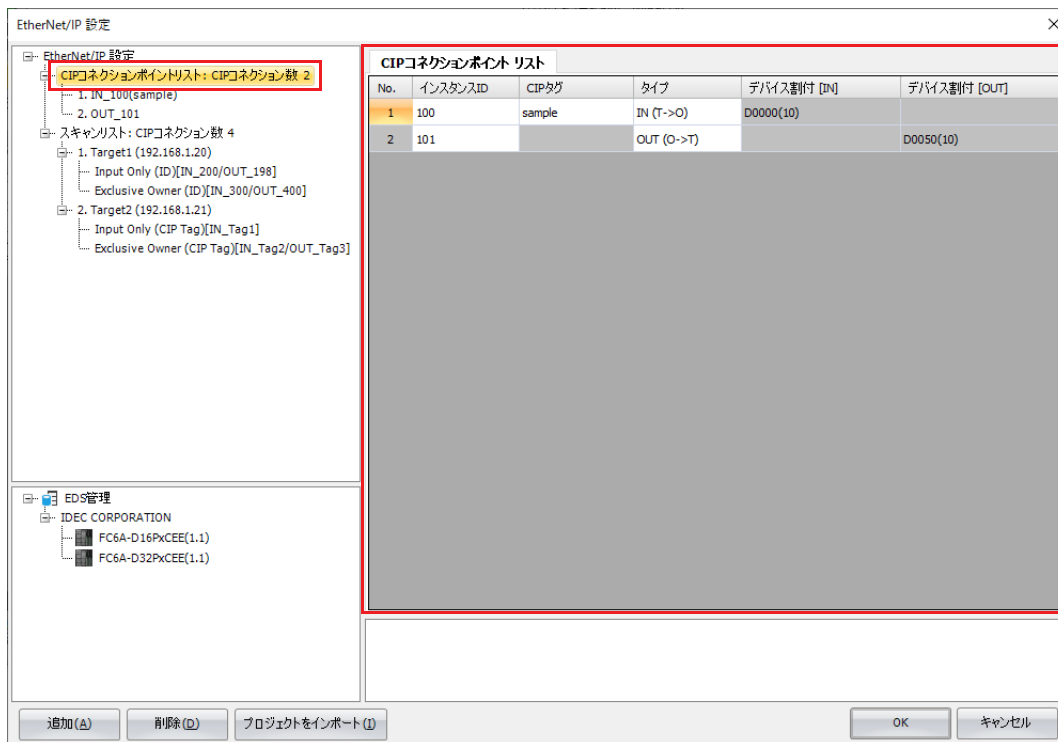
1 回の END 処理で、I/O データバッファとデータレジスタを相互に反映できる最大サイズを設定します。リフレッシュ上限は 1 ~ 400 ワード / スキャンの範囲で設定します。デフォルト値は 252 ワード / スキャンです。1 回の END 処理で I/O データバッファからデータレジスタへ書き込む最大サイズ、およびデータレジスタから I/O データバッファへ書き込む最大サイズはリフレッシュ上限の半分です。

【CIP コネクションポイントリスト】 ノードを選択している場合

EtherNet/IP ツリーエリアの【CIP コネクションポイントリスト】 ノードを選択すると、パラメータ設定エリアに【CIP コネクションポイントリスト】 タブが表示されます。

■【CIP コネクションポイントリスト】 タブ

設定済みの CIP コネクションポイントの一覧が表示されます。CIP タグ、インスタンス ID およびタイプは変更できます。デバイス割付のセルをダブルクリックすると、【CIP コネクションポイントリスト】 タブが表示され、デバイス割付の該当セルに移動できます。

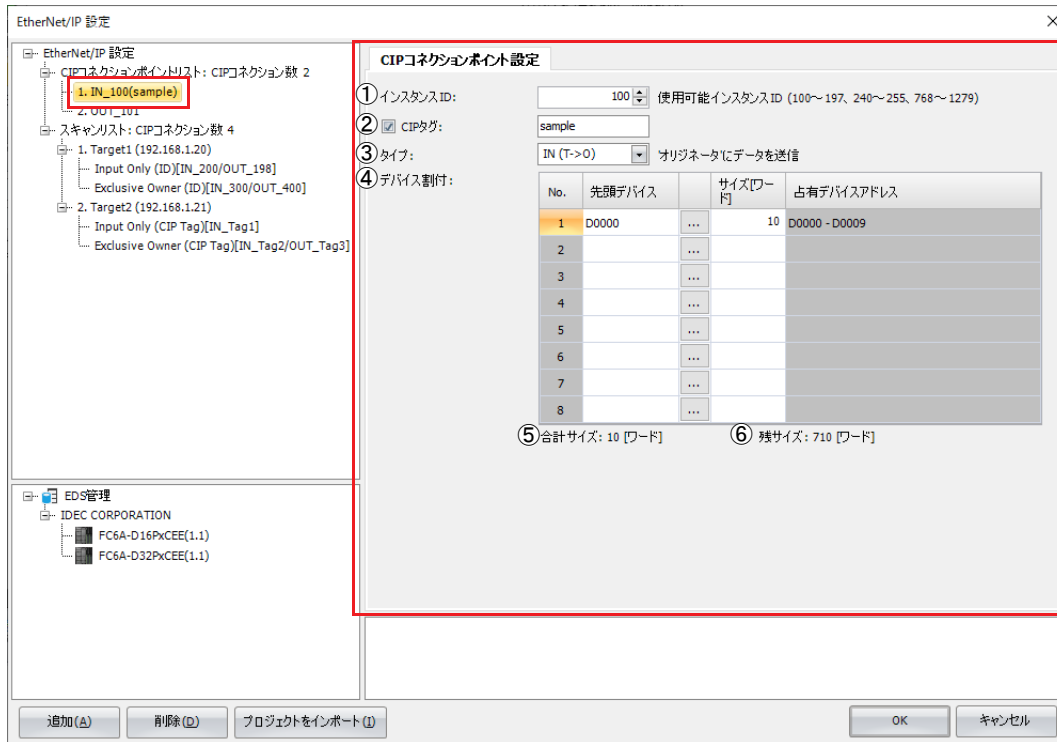


[CIP コネクションポイント] ノードを選択している場合

EtherNet/IP ツリーエリアの [CIP コネクションポイント] ノードを選択すると、パラメータ設定エリアに [CIP コネクションポイント設定] タブが表示されます。

■ [CIP コネクションポイント設定] タブ

CIP コネクションポイントを設定します。



① インスタンス ID

インスタンス ID を設定します。Plus CPU モジュール（ターゲット）が CIP タグ名を扱えない他のスキャナ機器（オリジネータ）と I/O メッセージ通信を行う場合、オリジネータはこのインスタンス ID に対して CIP コネクションを設定します。インスタンス ID は 100 ～ 1279 の範囲で設定します。デフォルト値は 100 です。他の CIP コネクションと重複しない値を設定してください。



100 ～ 1279 の範囲には使用できないインスタンス ID があります。WindLDR で表示している使用可能インスタンス ID 内の値を設定してください。

② CIP タグ

CIP タグ名を設定します。最大 64 バイトまで設定できます。他のスキャナ機器（オリジネータ）はこのタグ名に対して CIP コネクションを設定し、Plus CPU モジュール（ターゲット）と I/O メッセージ通信を行います。チェックボックスをオンにすると、CIP タグを設定できます。

③ タイプ

I/O メッセージ通信で扱うデータの方向を設定します。次の 2 種類から設定できます。デフォルト値は "IN(T → O)" です。

設定	内容
IN(T→O)	ターゲット（Plus CPUモジュール）がオリジネータへデータを送信します。
OUT(O→T)	ターゲット（Plus CPUモジュール）がオリジネータからデータを受信します。

④ デバイス割付

I/O メッセージ通信で受信または送信するデータを格納するためのデータレジスタを設定します。設定したデータレジスタを先頭に、データサイズで設定した分だけデータレジスタを使用します。データサイズは 1 ～ 720 の範囲で設定します。デフォルト値は 1 です。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

⑤ 合計サイズ

I/O メッセージ通信で使用するデータレジスタの合計値が表示されます。

⑥ 残サイズ

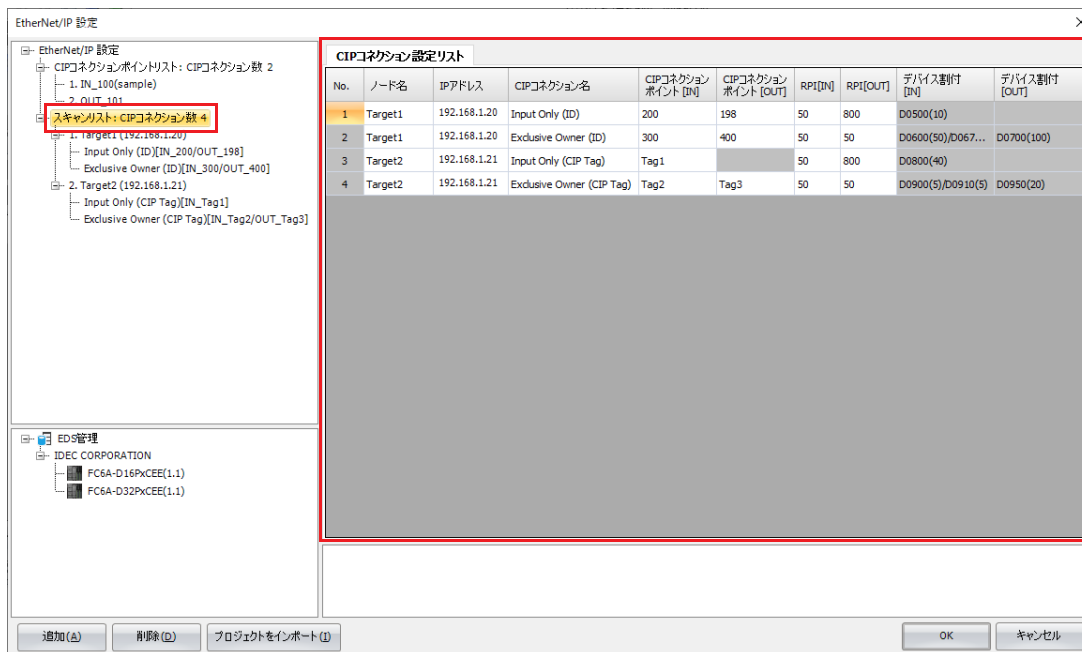
使用できる残りのデータサイズ（720 バイトから使用するデータレジスタの合計値を引いた値）が表示されます。

[スキャンリスト] ノードを選択している場合

EtherNet/IP ツリーエリアの [スキャンリスト] ノードを選択すると、パラメータ設定エリアに [CIP コネクション設定リスト] タブが表示されます。

■ [CIP コネクション設定リスト] タブ

設定済みの CIP コネクションの一覧が表示されます。ノード名、IP アドレス、CIP コネクション名、CIP コネクションポイントおよび RPI は変更できます。デバイス割付のセルをダブルクリックすると、[CIP コネクション設定] タブが表示され、デバイス割付の該当セルに移動できます。

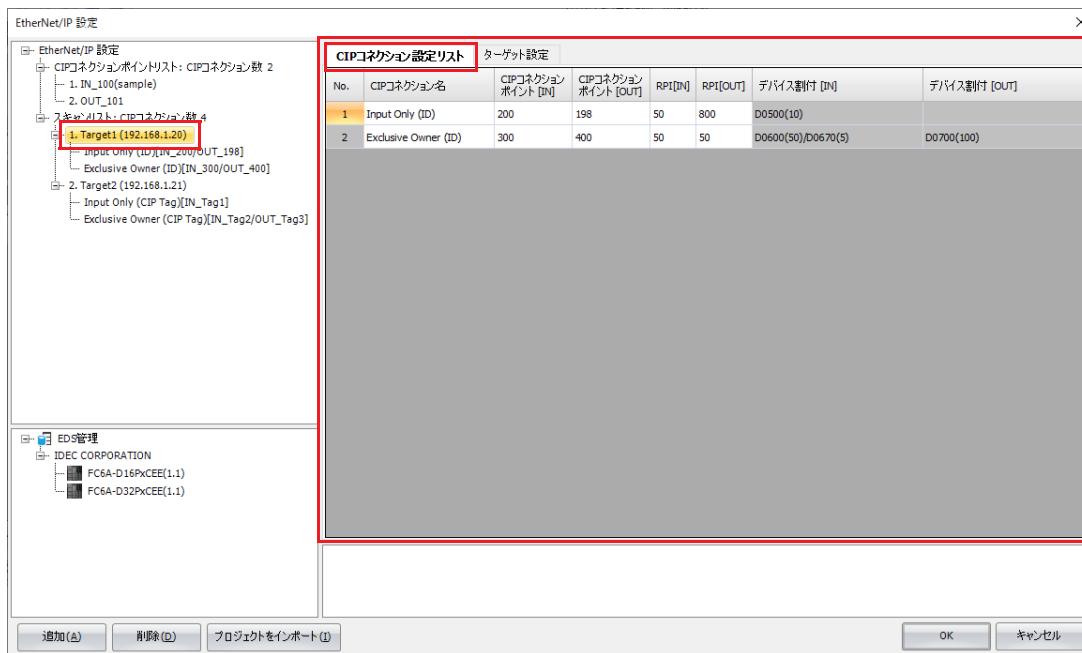


[ターゲット] ノードを選択している場合

EtherNet/IP ツリーエリアの [ターゲット] ノードを選択すると、パラメータ設定エリアに [CIP コネクション設定リスト] タブおよび [ターゲット設定] タブが表示されます。

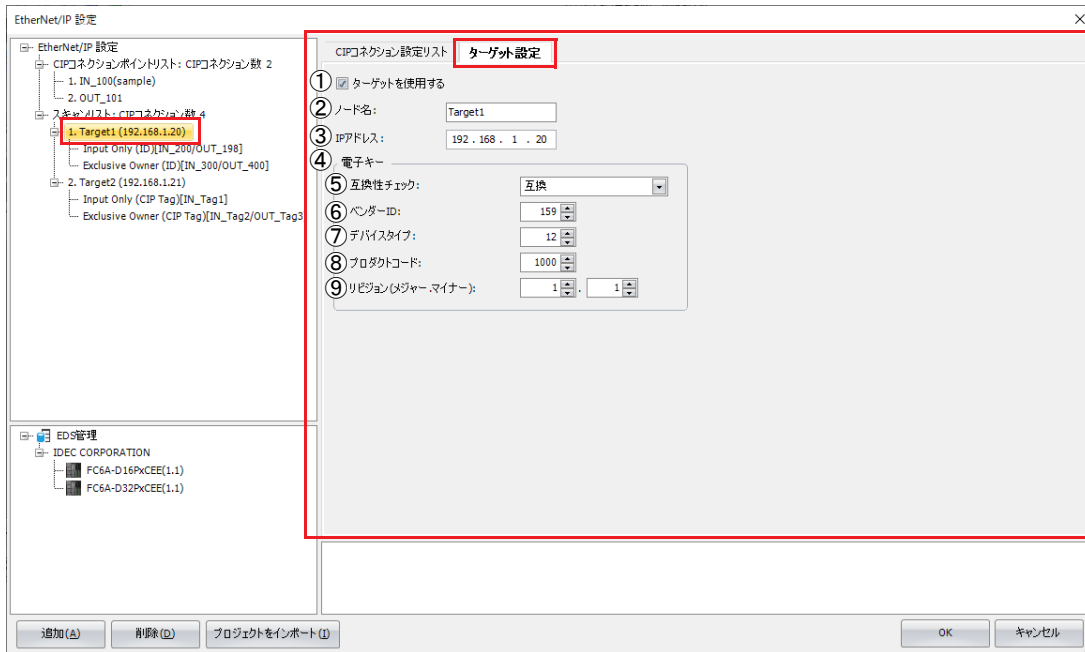
■ [CIP コネクション設定リスト] タブ

設定済みのターゲットの CIP コネクションの一覧が表示されます。このタブで CIP コネクション名、CIP コネクションポイントおよび RPI (通信周期) を変更できます。



■ [ターゲット設定] タブ

ターゲット機器の設定を行います。



①ターゲットを使用する

スキャンリストの CIP コネクションを、ターゲット単位で有効/無効にします。CIP コネクションを有効にすると、EtherNet/IP 通信許可 (M8460) が ON のとき、Plus CPU モジュールはターゲットと I/O メッセージ通信を行います。CIP コネクションを無効にすると、Plus CPU モジュールは I/O メッセージ通信を行いません。CIP コネクションを有効にする場合はチェックボックスをオンにします。

②ノード名

ターゲットのノード名を設定します。設定したノード名はスキャンリストで表示されます。最大 30 バイトまで設定できます。

③ IP アドレス

ターゲットの IP アドレスを設定します。

④電子キー

CIP コネクションの開設時に、スキャンリストに登録されたターゲットと実際に接続されている EtherNet/IP 機器が一致するかを互換性チェックの設定にしたがってチェックします。照合して一致した EtherNet/IP 機器とのみ I/O メッセージ通信を許可します。一致しない場合、Plus CPU モジュールは EtherNet/IP 機器と I/O メッセージ通信を行いません。

⑤互換性チェック

互換性チェックの方法を次の 3 種類から設定します。

互換性チェック	内容
無効	電子キーの照合を行いません。
互換	次の条件をすべて満たすターゲットとのみ I/O メッセージ通信を許可します。 ・ 設定と実機のベンダー ID、デバイスタイプおよびプロダクトコードが一致する ・ 設定よりも実機のメジャーリビジョンおよびマイナーリビジョンが大きい
完全一致	設定した電子キーがすべて一致するターゲットとのみ I/O メッセージ通信を行います。

⑥ベンダー ID

EtherNet/IP 機器のベンダー ID を設定します。0 ~ 65535 の範囲で設定します。

⑦デバイスタイプ

EtherNet/IP 機器のデバイスタイプを設定します。0 ~ 65535 の範囲で設定します。

⑧プロダクトコード

EtherNet/IP 機器のプロダクトコードを設定します。0 ~ 65535 の範囲で設定します。

⑨リビジョン (メジャー、マイナー)

EtherNet/IP 機器のメジャーリビジョンおよびマイナーリビジョンを設定します。

リビジョン	設定範囲
メジャーリビジョン	0~127
マイナーリビジョン	0~255

〔CIP コネクション〕 ノードを選択している場合

EtherNet/IP ツリーエリアの〔CIP コネクション〕ノードを選択すると、パラメータ設定エリアに〔CIP コネクション設定タブ〕が表示されます。

■〔CIP コネクション設定〕タブ

ターゲットのCIPコネクションを設定します。

①ノード名

〔ターゲット設定〕タブで設定したノード名が表示されます。このノード名を持つターゲットのCIPコネクションを設定します。

② CIP コネクション名

ノード名 (①) を持つターゲットのCIPコネクション名を設定します。次の5種類から設定できます。

設定	内容
Exclusive Owner (ID)	Plus CPUモジュールがターゲットとデータを送受信する場合に設定します。 Exclusive Owner (ID)は、ターゲットのCIPコネクションポイントをインスタンスIDで設定します。
Exclusive Owner (CIP Tag)	Exclusive Owner (CIP Tag)は、ターゲットのCIPコネクションポイントをCIPタグ名で設定します。 Plus CPUモジュールは、ターゲットのCIPコネクションポイントをCIPタグ名またはインスタンスIDで指定して、ターゲットとI/Oメッセージ通信を行います。
Input Only (ID)	Plus CPUモジュールがターゲットからデータを受信するだけの場合に設定します。 Input Only (ID)は、ターゲットのCIPコネクションポイントをインスタンスIDで設定します。
Input Only (CIP Tag)	Input Only (CIP Tag)は、ターゲットのCIPコネクションポイントをCIPタグ名で設定します。 Plus CPUモジュールは、ターゲットのCIPコネクションポイントをCIPタグ名またはインスタンスIDで指定して、ターゲットとI/Oメッセージ通信を行います。
Listen Only (ID)	他のオリジネータとターゲットがExclusive OwnerやInput OnlyでCIPコネクションを開けた状態で、そのターゲットがマルチキャストでデータを送信している場合に、Plus CPUモジュールがそのデータを受信する場合に設定します。 Listen Only (ID)は、ターゲットのCIPコネクションポイントをインスタンスIDで指定します。



Plus CPU モジュールは、ListenOnly の CIP コネクションを CIP タグ名で指定できません。

③タイムアウト

I/Oメッセージ通信のタイムアウト時間を設定します。Plus CPUモジュールとターゲットは、お互いにデータを受信する時間間隔を監視します。設定できる範囲はRPI (⑨、⑮) の整数倍 (4、8、16、32、64、128、256、512倍) です。デフォルト値はRPIの16倍です。

④インスタンスID

ターゲットのConfigurationのインスタンスIDを設定します。チェックボックスをオンにすると、インスタンスIDを設定できます。インスタンスIDは1～65535の範囲で設定します。デフォルト値は1です。

⑤データ

ターゲットの Configuration 用データを設定します。Plus CPU モジュールが CIP コネクションを開設する際、Configuration のインスタンス ID と一緒にターゲットへ送信します。ターゲットの仕様に応じて、0 ～ 400 バイトのパラメータを 16 進数で設定できます。



スキャンリストに追加したEtherNet/IP機器のEDSファイルにCIPコネクションの開設に関するパラメータの情報が定義されている場合、[データ] ボタンをクリックするとそのパラメータの一覧が表示されます。このパラメータは変更することができます。

⑥送信トリガ

Plus CPU モジュールがターゲットにデータを送信する方法を設定します。次の 2 種類から設定できます。デフォルト値は“サイクリック”です。

設定	内容
サイクリック	Plus CPUモジュールとターゲットは設定されたRPI（通信周期）（⑨）でデータを送信します。
Change Of State (COS)	Plus CPUモジュールとターゲットは設定されたRPI（通信周期）（⑨）または値が変化したときにデータを送信します。

⑦ COS 禁止時間

送信トリガ（⑥）で Change Of State (COS) を選択した場合、値が変化してから送信されるまでの時間を設定します。Plus CPU モジュールは、値の変化を検知してから設定した時間が経過するまでの間、データを送信しません。



値の変化とは、Plus CPUモジュールが内部に持つI/Oデータバッファのデータの変化を意味します。データレジスタからI/Oデータバッファへの反映はEND処理で行われます。

⑧制御レジスタ

CIP コネクションで使用するデータレジスタを設定します。

デバイスアドレス	項目		内容
先頭番号+0	ビット0	コネクションステータス	ON：接続して通信できている状態 OFF：接続していない状態（リトライ中など）
	ビット1～15	リザーブ	
先頭番号+1	リザーブ		
先頭番号+2			
先頭番号+3			
先頭番号+4			

⑨～⑭で I/O メッセージ通信の入力方向（ターゲット→Plus CPU モジュール（オリジネータ））に関するパラメータを設定します。

⑨ RPI

Plus CPU モジュール（オリジネータ）がターゲットと I/O メッセージ通信するときの入力方向（ターゲット→オリジネータ）の通信周期を設定します。設定できる範囲は 10 ～ 10000ms です。デフォルト値は 50ms です。

⑩ CIP コネクションタイプ

Plus CPU モジュール（オリジネータ）が、ターゲットの送信したデータを受信する方法を設定します。次の 2 種類から設定できます。デフォルト値は“Point to point”です。

設定	内容
Point to point	Plus CPUモジュールがターゲットからのデータを1対1で受信する場合に設定します。
Multicast	Plus CPUモジュールを含む複数台のオリジネータが1台のターゲットからデータを受信する場合に設定します。この場合、IN (T->O)のタイムアウト時間が250ms以上になるよう、タイムアウト（③）とRPI（⑨）を設定してください。

⑪ インスタンス ID/CIP タグ

ターゲットの CIP コネクションポイントを設定します。CIP コネクション名に応じて、ターゲットのインスタンス ID または CIP タグ名を設定します。次の 5 種類から設定できます。

CIP コネクション名	CIP コネクションポイント	設定範囲
Exclusive Owner (ID)	インスタンスID	1~65535
Exclusive Owner (CIP Tag)	CIPタグ名	UTF-8、最大64バイト（文字列終端含む）
Input Only (ID)	インスタンスID	1~65535
Input Only (CIP Tag)	CIPタグ名	UTF-8、最大64バイト（文字列終端含む）
Listen Only (ID)	インスタンスID	1~65535

⑫ デバイス割付 (IN (T->O))

Plus CPU モジュール（オリジネータ）がターゲットから受信したデータを格納するためのデータレジスタを設定します。データは複数のデータレジスタにワード単位で割り付けできます。最大 8 つの領域を設定できます。

割り付ける領域の合計サイズは最大 720 ワードです。設定したデータレジスタを先頭に、データサイズで設定した分だけデータレジスタを使用します。データサイズは 1 ~ 720 の範囲で設定します。デフォルト値は 1 です。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

⑬ 合計サイズ (IN (T->O))

デバイス割付 (IN (T->O)) (⑫) で使用するデータレジスタの合計値が表示されます。

⑭ 設定可能サイズ (IN (T->O))

使用できる残りのデータサイズ (720 バイトから使用するデータレジスタの合計値を引いた値) が表示されます。

⑮~⑳で I/O メッセージ通信の出力方向 (Plus CPU モジュール (オリジネータ) →ターゲット) に関するパラメータを設定します。

⑮ RPI

Plus CPU モジュール（オリジネータ）がターゲットと I/O メッセージ通信するときの出力方向 (オリジネータ→ターゲット) の通信周期を設定します。設定できる範囲は 10 ~ 10000ms です。デフォルト値は 50ms です。

CIP コネクション名に Input Only および Listen Only を設定している場合、この RPI は Heartbeat の周期となります。IN (T->O) データの RPI を 16 倍した値が自動的にセットされますが、任意に変更できます。

⑯ CIP コネクションタイプ

Plus CPU モジュール（オリジネータ）がターゲットにデータを送信する方法を設定します。“Point to point”のみ設定できます。

設定	内容
Point to point	Plus CPUモジュールがターゲットからのデータを1対1で送信する場合に設定します。

⑰ インスタンス ID/CIP タグ

ターゲットの CIP コネクションポイントを設定します。CIP コネクション名に応じて、ターゲットのインスタンス ID または CIP タグ名を設定します。次の 5 種類から設定できます。

CIP コネクション名	CIP コネクションポイント	設定範囲
Exclusive Owner (ID)	インスタンスID	1~65535
Exclusive Owner (CIP Tag)	CIPタグ名	UTF-8、最大64バイト（文字列終端含む）
Input Only (ID)	インスタンスID	1~65535*1
Input Only (CIP Tag)	CIPタグ名	設定不要
Listen Only (ID)	インスタンスID	1~65535*2

*1 ターゲットの Input Only 用のインスタンス ID を設定してください。

*2 ターゲットの Listen Only 用のインスタンス ID を設定してください。

⑱ デバイス割付 (OUT (O->T))

Plus CPU モジュール（オリジネータ）がターゲットに送信するデータを格納するデータレジスタを設定します。データは複数のデータレジスタにワード単位で割り付けできます。最大 8 つの領域を設定できます。

割り付ける領域の合計サイズは最大 720 ワードです。設定したデータレジスタを先頭に、データサイズで設定した分だけデータレジスタを使用します。データサイズは 1 ~ 720 の範囲で設定します。デフォルト値は 1 です。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

⑲ 合計サイズ (OUT (O->T))

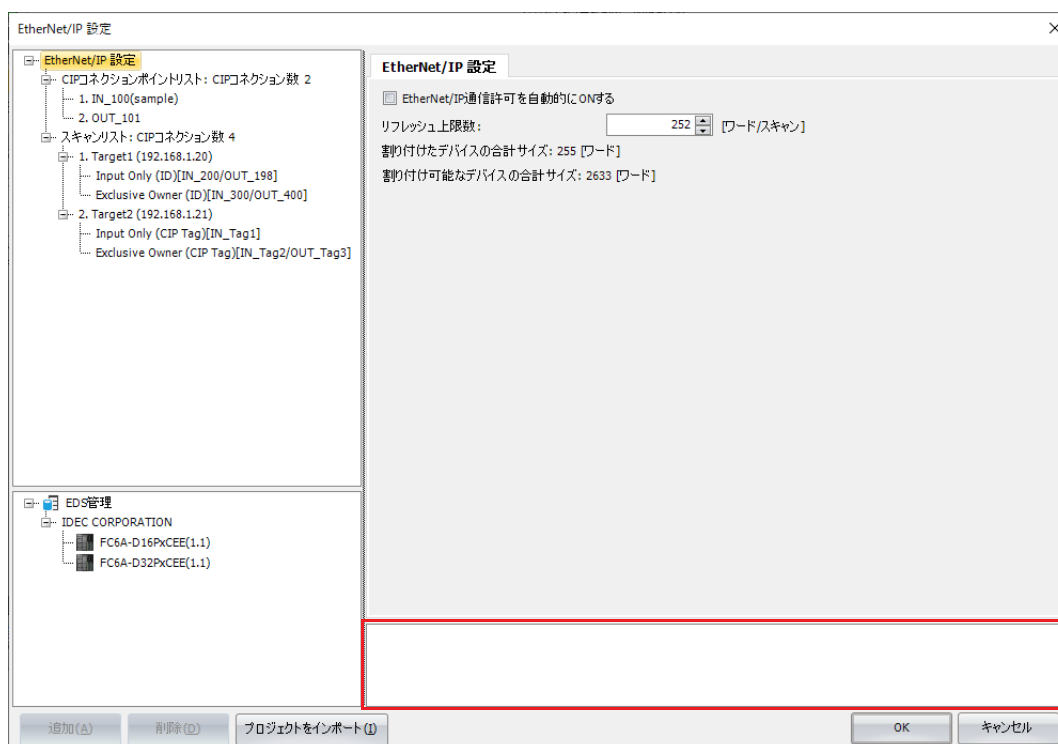
デバイス割付 (OUT (O->T)) (⑱) で使用するデータレジスタの合計値が表示されます。

⑳ 設定可能サイズ (OUT (O->T))

使用できる残りのデータサイズ (720 バイトから使用するデータレジスタの合計値を引いた値) が表示されます。

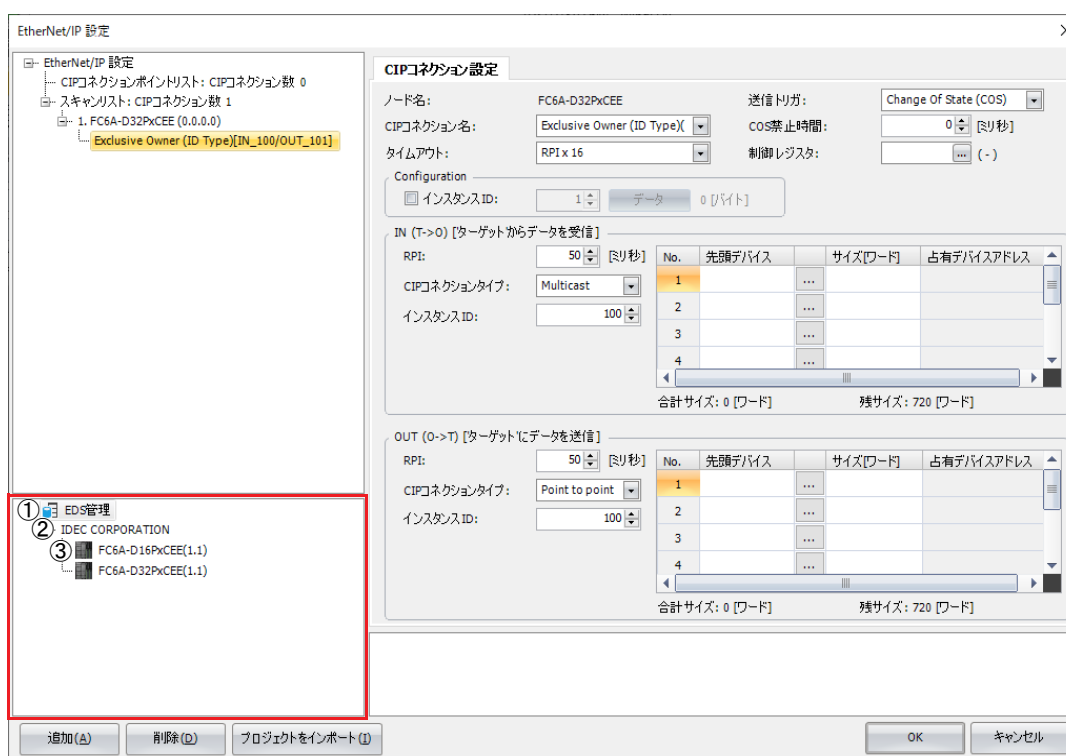
情報表示エリア

[EtherNet/IP 設定] ダイアログボックスで設定した内容にエラーがある場合、エラーの内容が表示されます。エラーの内容をクリックすると、エラーの箇所が表示されます。



EDS 管理エリア

インポートした EDS ファイルに設定された EtherNet/IP 機器の一覧が表示されます。



- ① [EDS 管理] ノード
[EDS 管理] ノードを選択して展開すると、インポートした EDS ファイルに設定された EtherNet/IP 機器の情報が一覧表示されます。
- ② [ベンダー] ノード
インポートした EDS ファイルに設定された EtherNet/IP 機器のベンダー名が表示されます。
- ③ [機器] ノード
インポートした EDS ファイルに設定された EtherNet/IP 機器の名前とそのリビジョンが表示されます。

[EDS 管理] エリアの各ノードを右クリックして表示されるコンテキストメニューの動作を説明します。

ノード	コンテキストメニュー	動作
EDS 管理	インポート	EDS ファイルをインポートします。
ベンダー	選択しているベンダーの EDS ファイルを削除	選択しているベンダーの EDS ファイルを削除します。
機器	エクスポート	選択している EtherNet/IP 機器の EDS ファイルをテキスト形式で出力します。
	開く	選択している EtherNet/IP 機器の EDS ファイルを開きます。
	削除	選択している EtherNet/IP 機器の EDS ファイルを削除します。
	スキャンリストに追加	選択している EtherNet/IP 機器をスキャンリストに追加します。



- ・ [機器] ノードを選択した状態で、EtherNet/IP ツリーエリアのスキャンリストにドラッグ & ドロップすると、選択した EtherNet/IP 機器をスキャンリストに追加することができます。
- ・ インポートした EDS ファイルは、パソコン内部に保存されるため、次回 WindLDR を起動したときも [EDS 管理] エリアにその EDS ファイルに設定された EtherNet/IP 機器の情報が一覧表示されます。
- ・ 弊社製品の EtherNet/IP 機器の EDS ファイルは WindLDR バージョン 8.18.0 以降で最初からインポートされています。
- ・ EDS ファイルをインポートした機器を使用した場合、CIP コネクション名は EDS ファイルに記載されている名称が表示されます。

EtherNet/IP 通信の設定の流れ

EtherNet/IP 通信の基本設定

EtherNet/IP ツリーエリアの [EtherNet/IP 設定] ノード*1 を選択すると、パラメータ設定エリアに [EtherNet/IP 設定] タブが表示されます。このタブで EtherNet/IP 通信を行うための基本設定を行います。詳細は、「[EtherNet/IP 設定] ノードを選択している場合」(16-22 頁) を参照してください。

*1 [EtherNet/IP 設定] ノードの詳細は、「EtherNet/IP ツリーエリア」(16-20 頁) を参照してください。

Plus CPU モジュールをターゲットとして使用する

Plus CPU モジュールをターゲットとして使用する場合に必要な設定について説明します。

Plus CPU モジュール (ターゲット) が他のスキャナ機器 (オリジネータ) とデータを送受信する場合、CIP コネクションポイントを設定する必要があります。

CIP コネクションポイントの設定

Plus CPU モジュールがオリジネータとデータを送受信するための設定で、通信するデータを格納するデータレジスタやデータサイズ、データの入力・出力方向などを設定します。

EtherNet/IP ツリーエリアの [CIP コネクションポイント] ノード*1 を選択すると、パラメータ設定エリアに [CIP コネクションポイント設定] タブが表示されます。このタブで CIP コネクションポイントを設定します。詳細は、「[CIP コネクションポイント] ノードを選択している場合」(16-24 頁) を参照してください。

*1 [CIP コネクションポイント] ノードの詳細は、「EtherNet/IP ツリーエリア」(16-20 頁) を参照してください。

Plus CPU モジュールをオリジネータとして使用する

Plus CPU モジュールをオリジネータとして使用する場合に必要な設定について説明します。

Plus CPU モジュールと I/O メッセージ通信するターゲット機器の設定および CIP コネクションの設定を行う必要があります。これらの設定は、一般に各社が用意している EDS ファイルを使用して簡単に設定することができます。詳細は、「EDS ファイルを使用したスキャンリストの作成」(16-32 頁) を参照してください。

ターゲットの設定

Plus CPU モジュールと I/O メッセージ通信を行うターゲットの IP アドレスなど、ターゲット機器の設定を行います。

EtherNet/IP ツリーエリアの [ターゲット] ノード*1 を選択すると、パラメータ設定エリアに [ターゲット設定] タブが表示されます。このタブでターゲット機器の設定を行います。詳細は、「[ターゲット] ノードを選択している場合」(16-25 頁) を参照してください。

*1 [ターゲット] ノードの詳細は、「EtherNet/IP ツリーエリア」(16-20 頁) を参照してください。

CIP コネクションの設定

Plus CPU モジュールがターゲットとデータを送受信するための設定で、CIP コネクション名、CIP コネクションタイプ、通信するデータを格納するデータレジスタやデータサイズ、I/O メッセージ通信の RPI (通信周期) などの設定をターゲットごとに行います。

EtherNet/IP ツリーエリアの [CIP コネクション] ノード*1 を選択すると、パラメータ設定エリアに [CIP コネクション設定] タブが表示されます。このタブで CIP コネクションの設定を行います。詳細は、「[CIP コネクション] ノードを選択している場合」(16-27 頁) を参照してください。

*1 [CIP コネクション] ノードの詳細は、「EtherNet/IP ツリーエリア」(16-20 頁) を参照してください。

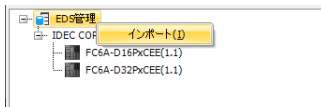
EDS ファイルを使用したスキャンリストの作成

Plus CPU モジュールがターゲットとデータを送受信するために必要なターゲット機器の設定、CIP コネクションの設定を、EDS (Electric Data Sheets) ファイルを使用して簡単に設定することができます。EDS ファイルは、ベンダー名など EtherNet/IP 機器固有の情報や、データの送受信の設定、パラメータの仕様などが定義されています。EDS ファイルはそれぞれの EtherNet/IP 機器のベンダーが作成し、提供しています。EDS ファイルはベンダーのサイトなどから入手してください。

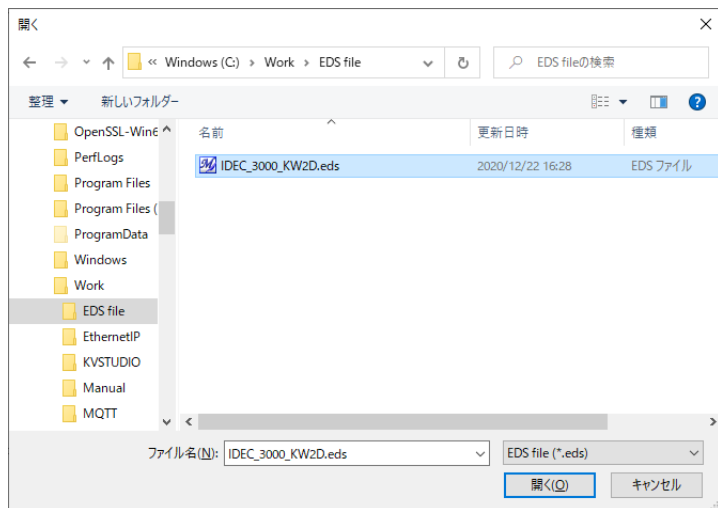
EDS ファイルをインポートして、スキャンリストを作成する手順を説明します。

●操作手順

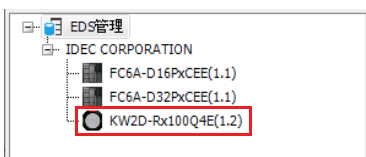
1. [EDS 管理] エリアで [EDS 管理] ノードを右クリックし、[インポート] をクリックします。



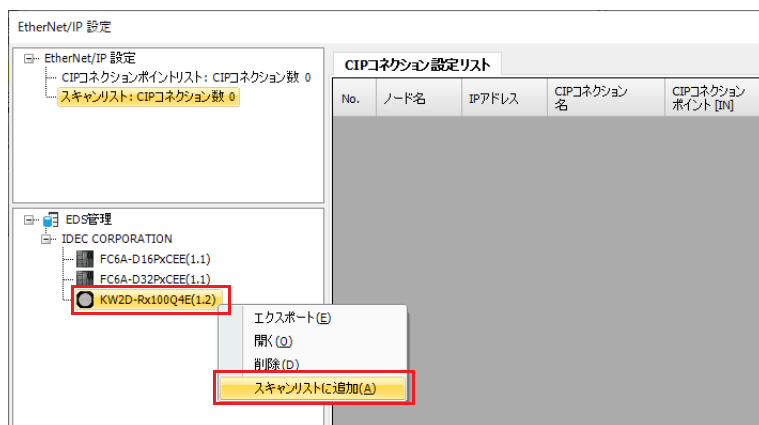
[開く] ダイアログボックスが表示されます。



2. EDS ファイルを選択し、[開く] ボタンをクリックします。
[EDS 管理] エリアに選択した EDS ファイルがインポートされます。



3. EDS 管理エリアで [EDS 管理] ノードを右クリックし、[スキャンリストに追加] をクリックします。

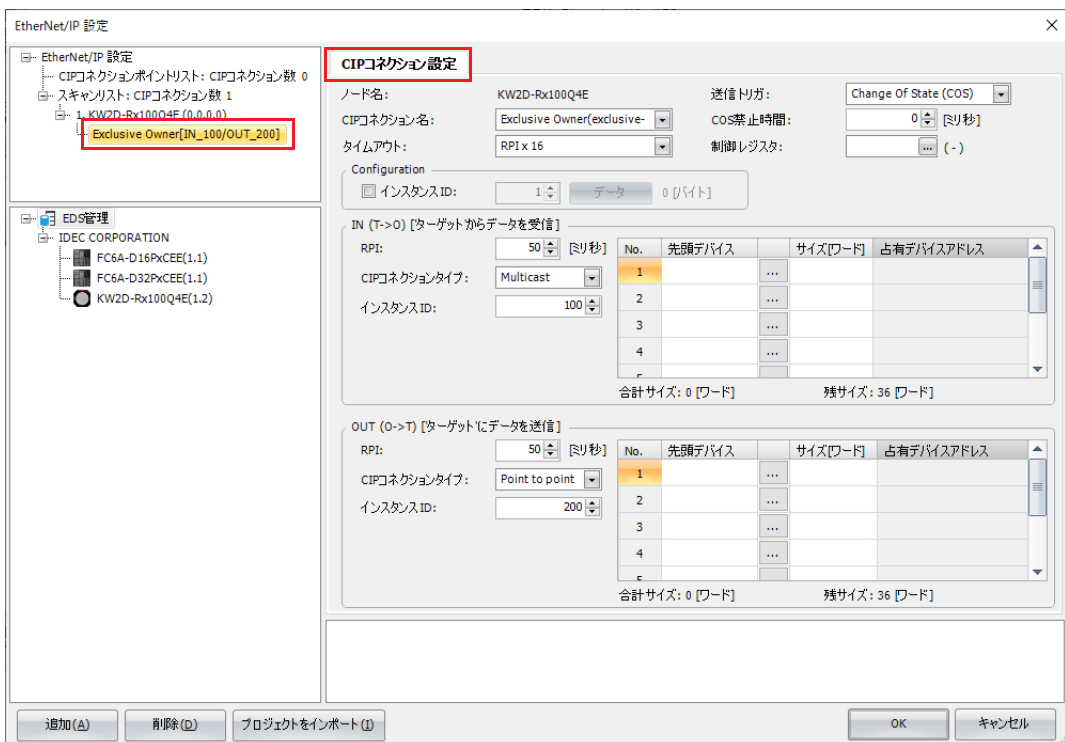


選択した EDS ファイルの設定がスキャンリストに追加されます。

4. EtherNet/IP ツリーエリアの [ターゲット] ノードを選択し、[ターゲット設定] タブで IP アドレスを設定します。また、[ターゲット設定] タブの設定を確認し、必要であれば設定を変更します。



5. EtherNet/IP ツリーエリアの [CIP コネクション] ノードを選択し、[CIP コネクション設定] タブでタイムアウト、制御レジスタおよびデバイス割付を設定します。また、[CIP コネクション設定] タブの設定を確認し、必要であれば設定を変更します。



6. [OK] ボタンをクリックします。

以上で、スキャンリストの作成が完了します。



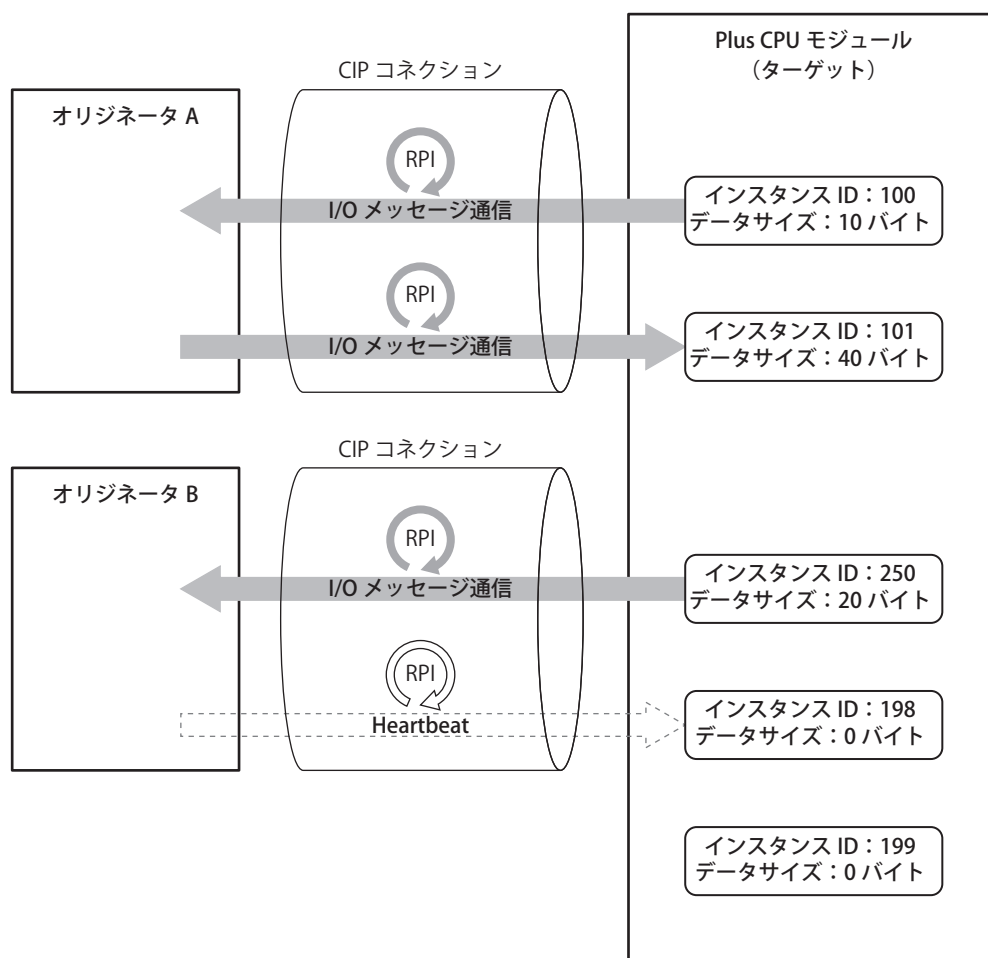
EDS ファイルは各 EtherNet/IP 機器メーカーが用意しています。EDS ファイルの入手方法については、各 EtherNet/IP 機器メーカーにお問い合わせください。

EtherNet/IP の設定例

Plus CPU モジュールをターゲットとして使用する

Plus CPU モジュールをターゲットとして、オリジネータと接続する場合の設定例について説明します。

システム構成図



Plus CPU モジュールへ設定する仕様

Plus CPU モジュールをターゲットとし、オリジネータ A および B から読み書きする CIP コネクションポイントを設定します。

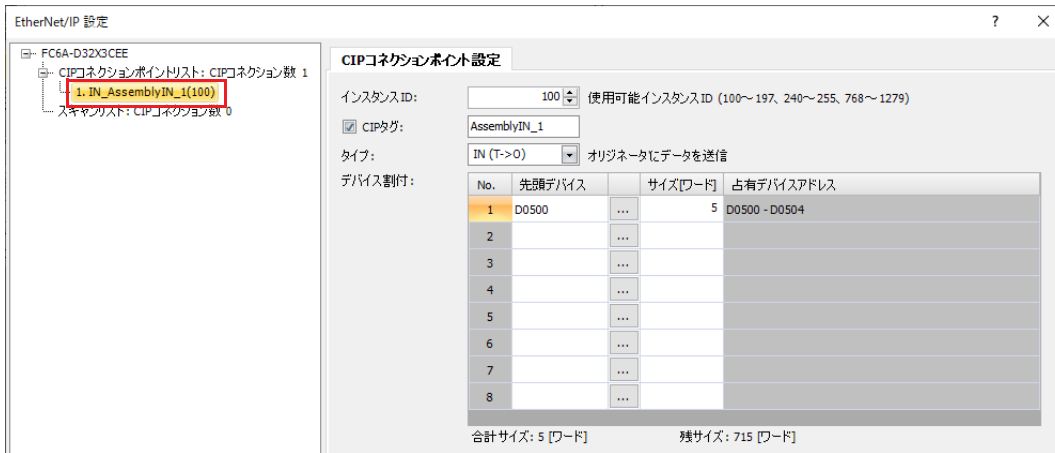
入出力タイプ	CIP コネクションポイント		データ格納用のデータレジスタ
	インスタンス ID	データサイズ	
入力	100	10 バイト (5 ワード)	D0500
出力	101	40 バイト (20 ワード)	D0600
入力	250	20 バイト (10 ワード)	D0700
出力 (Input Only 用)	198 (システム固定)	0 バイト	—
出力 (Listen Only 用)	199 (システム固定)	0 バイト	—

設定

Plus CPU モジュールの CIP コネクションポイントの設定を行います。

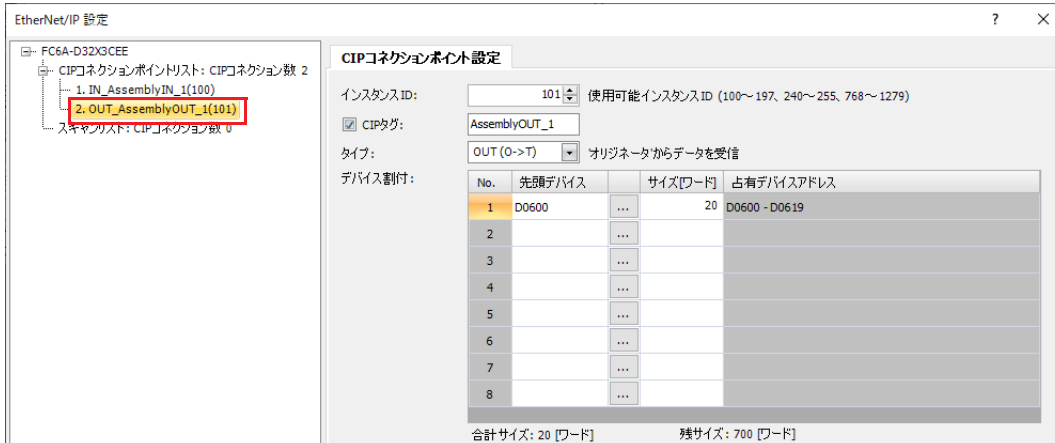
■ Plus CPU モジュールの CIP コネクションポイントリスト設定

Plus CPU モジュール（ターゲット）は、データレジスタに格納された CIP コネクションポイント（インスタンス ID：100）のデータをオリジネータに送信します。



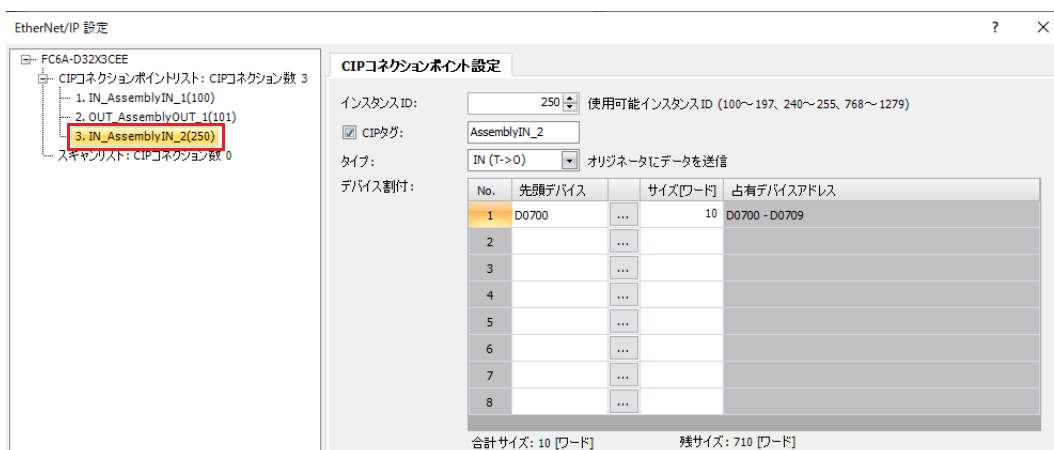
設定項目	内容	
1.	CIPタグ	"AssemblyIN_1"
	インスタンスID	100
	タイプ	IN (T->O)
	デバイス割付	D0500～D0504 (5ワード)

Plus CPU モジュール（ターゲット）は、CIP コネクションポイント（インスタンス ID：101）のデータをオリジネータから受信してデータレジスタに格納します。



設定項目	内容	
2.	CIPタグ	"AssemblyOUT_1"
	インスタンスID	101
	タイプ	OUT (O->T)
	デバイス割付	D0600～D0619 (20ワード)

Plus CPU モジュール（ターゲット）は、データレジスタに格納された CIP コネクションポイント（インスタンス ID：200）のデータをオリジネータに送信します。



設定項目		内容
3.	CIPタグ	"AssemblyIN_2"
	インスタンスID	250
	タイプ	IN (T->O)
	デバイス割付	D0700～D0709 (10ワード)



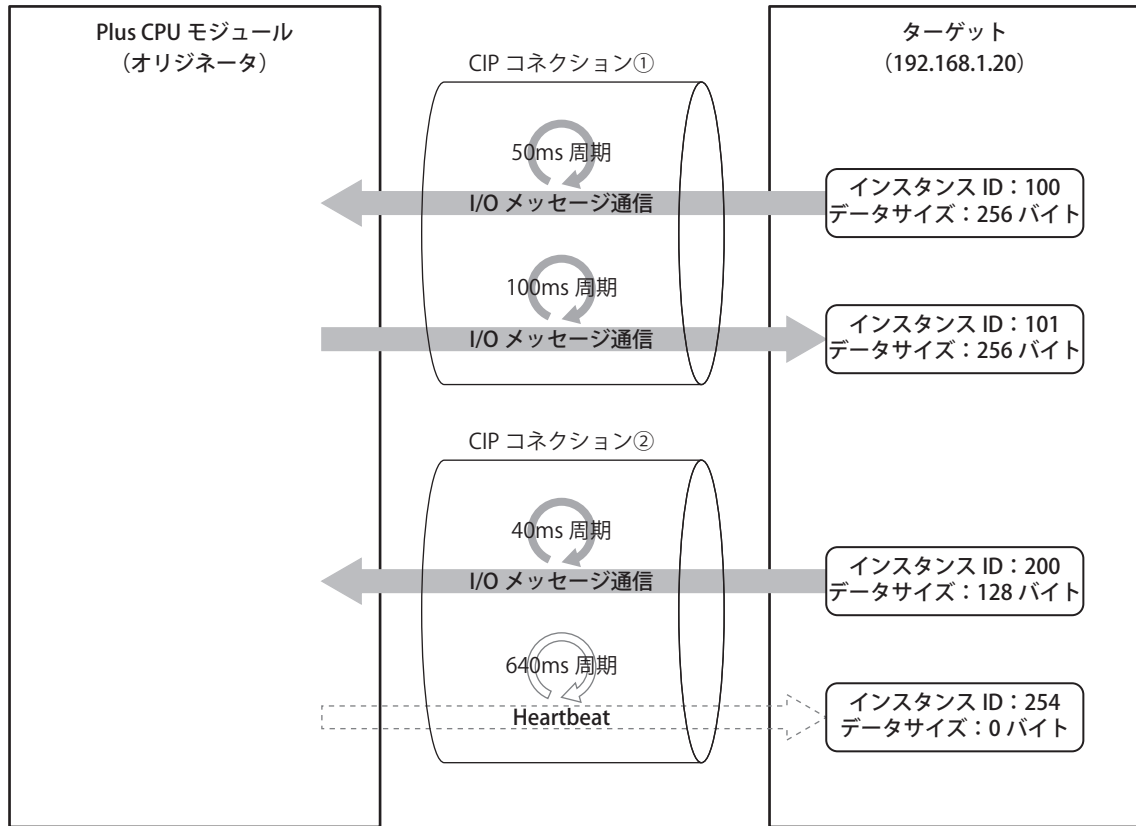
RPI や Exclusive Owner 等の CIP コネクションの種類は、オリジネータが CIP コネクション開設するときに指定します。

以上で、設定が完了します。

Plus CPU モジュールをオリジネータとして使用する

Plus CPU モジュールをオリジネータとして、ターゲットと接続する場合の設定例について説明します。

システム構成図



接続先ターゲットの仕様

IP アドレス	192.168.1.20			
CIP コネクションポイント	次の表は、Plus CPUモジュールがオリジネータとして接続するターゲットのCIPコネクションポイントを示します。			
	CIP コネクション	タイプ	インスタンス ID	データサイズ
	CIPコネクション①	Configuration	1	0/バイト
		入力	100	256バイト (128ワード)
	CIPコネクション②	出力	101	256バイト (128ワード)
		Configuration	1	0/バイト
入力		200	128バイト (64ワード)	
	出力 (Heartbeat)	254	0/バイト	

設定

ターゲット機器、CIP コネクションの設定を行います。

■ Plus CPU モジュールのスキャンリスト設定（ターゲット機器の設定）

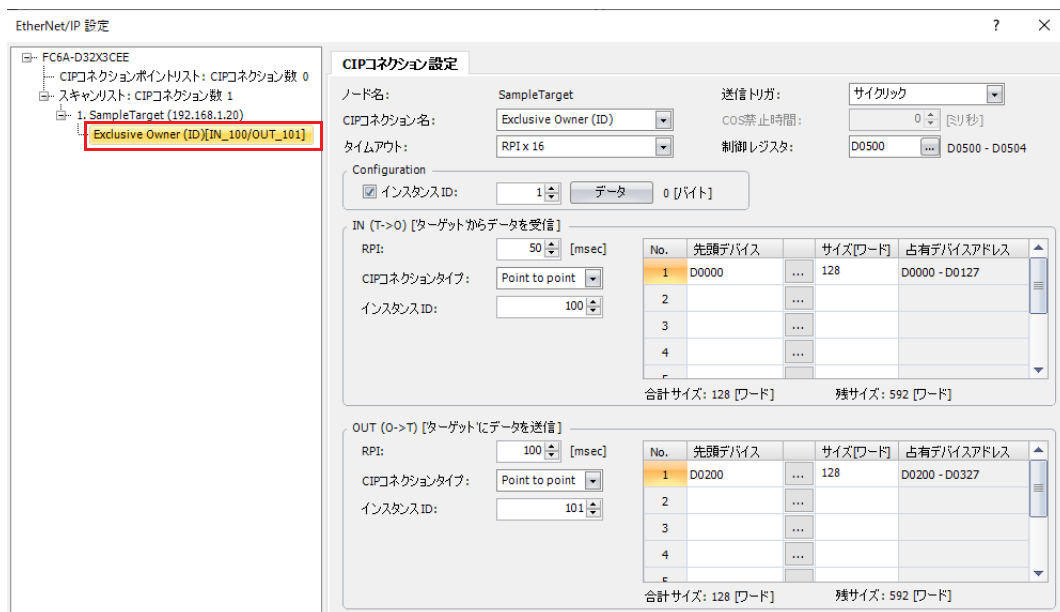


設定項目	内容
ターゲットを使用する	有効
ノード名	SampleTarget
IPアドレス	192.168.1.20
電子キー	互換性チェック
	無効

■ Plus CPU モジュールのスキャンリスト設定（CIP コネクションの設定）

CIP コネクション①

Plus CPU モジュール（オリジネータ）は、CIP コネクションポイント（インスタンス ID：100 および 101）に対して CIP コネクションの開設を要求し、成功するとターゲットから 50 ms ごとにデータを受信し、ターゲットへ 100 ms ごとにデータを送信します。

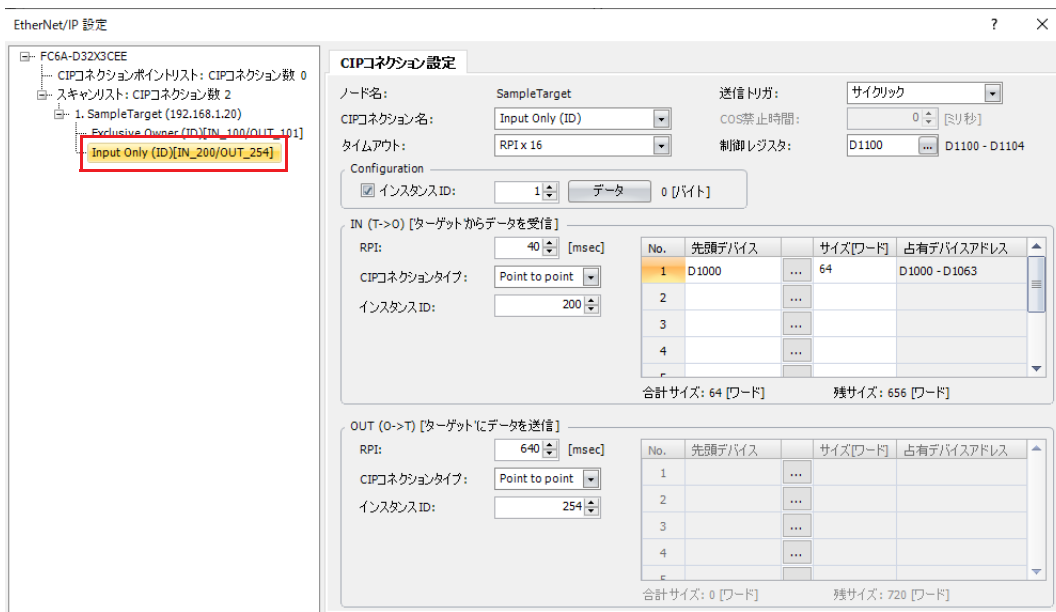


設定項目	内容	
CIPコネクション名	Exclusive Owner (ID)	
送信トリガ	サイクリック	
タイムアウト	RPI x 16	
制御レジスタ	D0500	
Configuration	インスタンスID	1
	データ	0/バイト

設定項目		内容
IN (T->O)	RPI	50 msec
	CIPコネクションタイプ	Point to point
	インスタンスID	100
	デバイス割付	D0000~D0127 (128ワード)
OUT (O->T)	RPI	100 msec
	CIPコネクションタイプ	Point to point
	インスタンスID	101
	デバイス割付	D0200~D0327 (128ワード)

CIP コネクション②

ターゲットへデータを送信しない場合、CIP コネクション名に Input Only を設定してください。Plus CPU モジュール（オリジネータ）は、CIP コネクションポイント（インスタンス ID：200 および 254）に対して CIP コネクションの開設を要求し、成功するとターゲットから 40ms ごとにデータを受信します。ターゲットへはデータではなく Heartbeat を 640ms ごとに送信します。



設定項目		内容
CIPコネクション名		Input Only (ID)
送信トリガ		サイクリック
タイムアウト		RPI x 16
制御レジスタ		D1100
Configuration	インスタンスID	1
	データ	0バイト
IN (T->O)	RPI	40 msec
	CIPコネクションタイプ	Point to point
	インスタンスID	200
	デバイス割付	D1000~D1063 (64ワード)
OUT (O->T)	RPI	640msec
	CIPコネクションタイプ	Point to point
	インスタンスID	254

以上で、設定が完了します。

オブジェクト

Identity オブジェクト

クラス

■クラスサービス

Get_Attribute_Single (0EH)、Get_Attributes_All (01H) に対応しています。

ID	サービス
0EH	Get_Attribute_Single
01H	Get_Attributes_All

■クラスアトリビュート (インスタンス ID : 0)

ID	アクセス	名称	データタイプ
1	Get	Revision	UINT
2	Get	Max Instance	UINT
3	Get	Number of Instances	UINT
6	Get	Maximum ID Number Class Attributes	UINT
7	Get	Maximum ID Number Instance Attributes	UINT

インスタンス

■インスタンスサービス

Get_Attribute_Single (0EH)、Get_Attributes_All (01H)、Reset サービス (05H) に対応しています。

ID	サービス
0EH	Get_Attribute_Single
01H	Get_Attributes_All
05H	Reset

■インスタンスアトリビュート (インスタンス ID : 1)

ID	アクセス	名称	データタイプ	説明	属性値
1	Get	Vendor ID	UINT	ベンダー識別番号	159
2	Get	Device Type	UINT	一般的なデバイスタイプ	14 (Programmable Logic Controller)
3	Get	Product Code	UINT	製品の識別コード	*1
4	Get	Revision	STRUCT	Identityオブジェクトのレビジョン	
		Major Revision	USINT	メジャーレビジョン	
		Minor Revision	USINT	マイナーレビジョン	
5	Get	Status	WORD	デバイスの現在のステータス	
6	Get	Serial Number	UDINT	シリアル番号	
7	Get	Product Name	SHORT-STRING	製品名	*1

*1 製品毎に属性値が異なります。

	FC6A-D16R1CEE, FC6A-D16R4CEE	FC6A-D16P1CEE, FC6A-D16P4CEE	FC6A-D16K1CEE, FC6A-D16K4CEE	FC6A-D32P1CEE, FC6A-D32P4CEE	FC6A-D32K1CEE, FC6A-D32K4CEE
Product Code	2000	2010	2020	2030	2040
Product Name	"FC6A-D16RxCEE"	"FC6A-D16PxCEE"	"FC6A-D16KxCEE"	"FC6A-D32PxCEE"	"FC6A-D32KxCEE"

Message Router オブジェクト

クラス

■ クラスサービス

ID	サービス
0EH	Get_Attribute_Single

■ クラスアトリビュート (インスタンス ID : 0)

ID	アクセス	名称	データタイプ
1	Get	Revision	UINT
2	Get	Max Instance	UINT
3	Get	Number of Instances	UINT
6	Get	Maximum ID Number Class Attributes	UINT
7	Get	Maximum ID Number Instance Attributes	UINT

インスタンス

■ インスタンスサービス

ID	サービス
0EH	Get_Attribute_Single

■ インスタンスアトリビュート (インスタンス ID : 1)

ID	アクセス	名称	データタイプ
1	Get	Object_list	STRUCT
	Get	Number	UINT
	Get	Classes	UINTの配列
2	Get	Number Available	UINT

Assembly オブジェクト

クラス

■クラスサービス

ID	サービス
0EH	Get_Attribute_Single

■クラスアトリビュート（インスタンス ID：0）

ID	アクセス	名称	データタイプ
1	Get	Revision	UINT

インスタンス

■インスタンスサービス

ID	サービス
0EH	Get_Attribute_Single
10H	Set_Attribute_Single

■インスタンスアトリビュート

①オリジネータ→ターゲット

ID	アクセス	名称	データタイプ
3	Get/Set	Data	BYTEの配列
4	Get	Size	UINT

②ターゲット→オリジネータ

ID	アクセス	名称	データタイプ
3	Get	Data	BYTEの配列
4	Get	Size	UINT

Connection Manager オブジェクト

クラス

■クラスサービス

クラスサービスはありません。

■クラスアトリビュート

クラスアトリビュートはありません。

インスタンス

■インスタンスサービス

ID	サービス
54H	Forward_Open
5BH	Large_Forward_Open
4EH	Forward_Close

■インスタンスアトリビュート

インスタンスアトリビュートはありません。

TCP/IP Interface オブジェクト

クラス

■ クラスサービス

ID	サービス
0EH	Get_Attribute_Single

■ クラスアトリビュート

ID	アクセス	名称	データタイプ
1	Get	Revision	UINT
2	Get	Max Instance	UINT
3	Get	Num Instances	UINT
6	Get	Maximum ID Number Class Attributes	UINT
7	Get	Maximum ID Number Instance Attributes	UINT

インスタンス

■ インスタンスサービス

ID	サービス
0EH	Get_Attribute_Single
10H	Set_Attribute_Single

■ インスタンスアトリビュート

ID	アクセス	名称	データタイプ
1	Get	Status	DWORD
2	Get	Configuration Capability	DWORD
3	Get	Configuration Control	DWORD
4	Get	Physical Link Object	STRUCT
		Path size	UINT
		Path	EPATH
5	Get	Interface Configuration	STRUCT
		IP Address	UDINT
		Network Mask	UDINT
		Gateway Address	UDINT
		Name Server	UDINT
		Name Server 2	UDINT
Domain Name	STRING		
6	Get	Host Name	STRING
13	Get/Set	Encapsulation Inactivity Timeout	UINT

Ethernet Link オブジェクト

イーサネット通信の設定や読み出しを行います。またイーサネット通信のステータスを読み出します。

クラス

■ クラスサービス

ID	サービス
0EH	Get_Attribute_Single

■ クラスアトリビュート

ID	アクセス	名称	データタイプ
1	Get	Revision	UINT
2	Get	Max Instance	UINT
3	Get	Num Instances	UINT

インスタンス

■ インスタンスサービス

Get_Attribute_Single (0EH)、Get_Attributes_All (01H) に対応しています。

ID	サービス
0EH	Get_Attribute_Single
01H	Get_Attributes_All

■ インスタンスアトリビュート

ID	アクセス	名称	データタイプ
1	Get	Interface Speed	UDINT
2	Get	Interface Flags	DWORD
3	Get	Physical Address	ARRAY of 6 USINTs
11	Get	Interface Capability	STRUCT
		Capability Bits	DWORD
		Speed/Duplex Options	STRUCT
		Speed/Duplex Array Count	USINT
		Speed/Duplex Array	ARRAY of STRUCT
		Interface Speed	UINT
		Interface Duplex Mode	USINT

第17章 MQTT通信

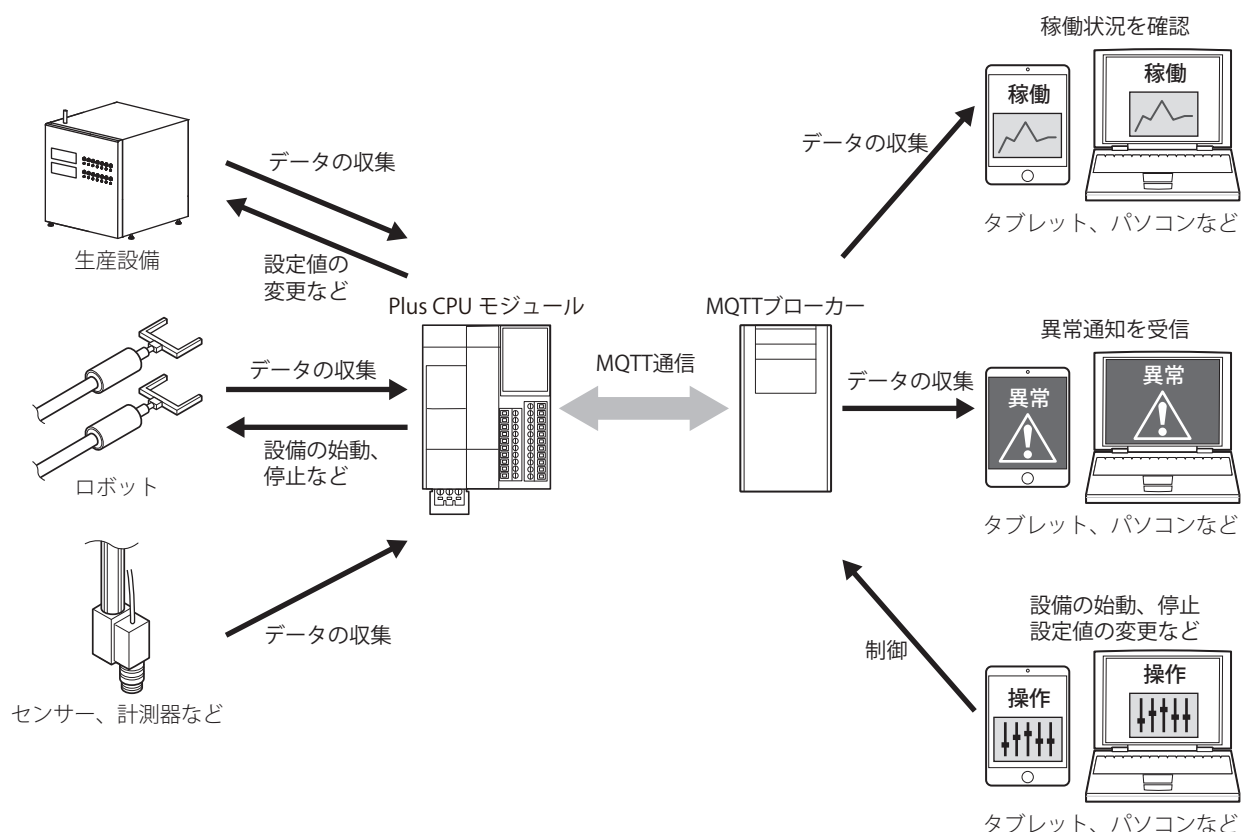
この章では、Plus CPU モジュールの MQTT 通信について説明します。

概要

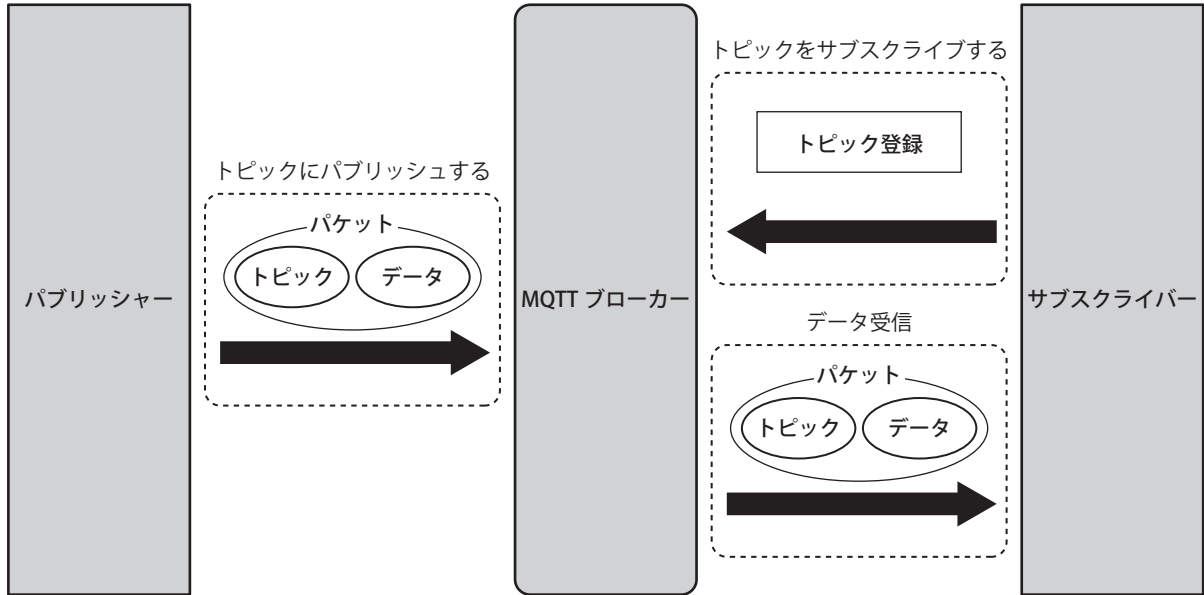
Plus CPU モジュールは、MQTT 通信に対応しています。

MQTT とは、送受信するデータの軽量さやスケーリングの容易さが特徴のプロトコルです。MQTT 通信を使用して、次のようなことができます。

- ・遠隔地にある設備のデータを収集して稼働状況を確認する
- ・遠隔地にある設備に異常があった場合に通知を受け取る
- ・遠隔地にある設備の始動・停止、または設定値を変更する



Plus CPU モジュールは、MQTT 通信におけるクライアント（パブリッシャーおよびサブスクライバー）としてブローカーと通信できます。MQTT 通信に関する詳細、および MQTT 通信で一般的に使用される用語の説明については、OASIS（Organization for the Advancement of Structured Information Standards）が公開している MQTT の仕様書を参照してください。MQTT のネットワークは、ブローカーおよびパブリッシャー・サブスクライバーで構成されます。サブスクライバーはブローカーに接続し、サブスクライブするトピックを登録します。パブリッシャーはブローカーに接続し、ブローカーにトピックおよびデータを含むパケットを送信します。ブローカーは、パケット内のトピックを参照し、そのトピックをサブスクライブしているサブスクライバーに、本パケットを送信し、サブスクライバーは本パケット内のデータを受信します。



- サブスクライバーがサブスクライブするトピックをブローカーに登録することを「トピックをサブスクライブする」と呼びます。
- パブリッシャーがブローカーにトピックおよびデータを含むパケットを送信することを「トピックにパブリッシュする」と呼びます。
- MQTT 通信で送受信されるパケットを MQTT パケットと呼びます。
- トピックとはデータの種類を区別するための情報です。パブリッシャーは、送信するデータの種類に応じたトピックを付加してデータを送信します。サブスクライバーは、受信したいデータの種類に応じたトピックをサブスクライブすることで、必要なデータのみ受信できます。

対応機種および通信ポート

対応する機種と通信ポートは、次のとおりです。

通信ポート	All-in-One CPU モジュール			CAN J1939 All-in-One CPU モジュール	Plus CPU モジュール	
	16点タイプ	24点タイプ	40点タイプ		Plus16点タイプ	Plus32点タイプ
Ethernet ポート 1	—	—	—	—	○	○
Ethernet ポート 2	—	—	—	—	—	—
HMI-Ethernet ポート	—	—	—	—	—	—

MQTT 通信はシリアル通信するための通信ポート 1～33 では使用できません。

基本仕様

基本仕様は、次のとおりです。

モジュール	Plus CPU モジュール	
通信ポート	Ethernetポート1	
準拠規格	MQTT Version 3.1.1	
基本仕様	ホスト名	最大128バイト
	認証	対応
	アカウント名	最大128バイト
	パスワード	最大496バイト
	SSL/TLS	対応
	クライアント ID	最大128バイト
	キープアライブ	5~65535秒
	QoS	0, 1, 2
	トピック	最大256バイト
	データの形式	JSON形式、最大32KB（パブリッシュ）、最大8KB（サブスクライブ）
パブリッシュ仕様	Retain	対応
	Will	未対応
サブスクライブ仕様	ワイルドカード (#, +)	対応
	Persistent Session	未対応

対応 MQTT ブローカーおよびクラウドサービス

次の MQTT ブローカーおよびクラウドサービスを利用して、接続・パブリッシュ・サブスクライブの基本的な動作の確認を行っています。(2022年7月現在)

- Mosquitto
- AWS IoT Core (Amazon.com) (ポート番号 8883) *1
- Azure IoT Hub (azure.microsoft.com) (ポート番号 8883) *1

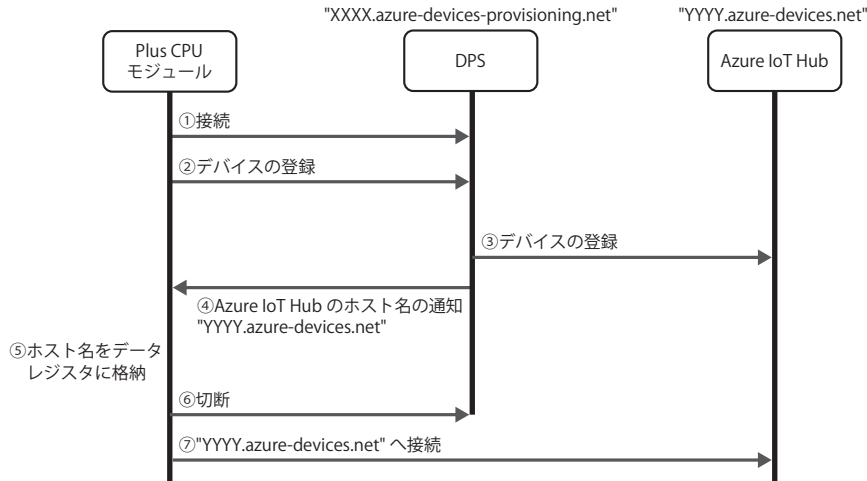
Plus CPU モジュールと上記の MQTT ブローカーおよびクラウドサービスの両方でサポートしているサービスが使用できます。ただし、AWS IoT Core および Azure IoT Hub は、クラウドサービスの仕様変更に伴い、接続できなくなる場合があります。

また、Plus CPU モジュールは MQTT ブローカーおよびクラウドサービスと次の接続方法をサポートします。

- 汎用の MQTT ブローカーへ接続する**
 Plus CPU モジュールを、汎用の MQTT ブローカーへ接続できます。
- AWS IoT Core へ接続する**
 AWS IoT Core へ事前に登録した Plus CPU モジュールを、AWS IoT Core へ接続できます。
- Azure IoT Hub へ SAS を使って接続する**
 Azure IoT Hub へ事前に登録した Plus CPU モジュールを、Azure IoT Hub へ Shared Access Signature (SAS) を使用して接続できます。Azure IoT Hub との接続には SAS トークンが使用されます。
- Azure IoT Hub へ X.509 証明書を使って接続する**
 Azure IoT Hub へ事前に登録した Plus CPU モジュールを、Azure IoT Hub へ X.509 証明書を使用して接続できます。X.509 証明書はデバイス証明書（クライアント証明書）として指定します。
- Azure IoT Hub へ DPS 経由で接続する**
 Device Provisioning Service (DPS) を経由して Plus CPU モジュールを Azure IoT Hub へ登録し、Azure IoT Hub へ接続できます。DPS および Azure IoT Hub との接続には SAS トークンが使用されます。

*1 AWS IoT Core / Azure IoT Hub のポート番号 443 に接続できません。

DPS へ接続してから Azure IoT Hub へ接続するまでの流れを下図に示します。



- ① Plus CPU モジュールが DPS へ接続します。
- ② DPS へ Plus CPU モジュールを登録します。
- ③ DPS が Plus CPU モジュールを Azure IoT Hub へ登録します。
- ④ DPS が Plus CPU モジュールの接続先となる Azure IoT Hub のホスト名を通知します。
- ⑤ Plus CPU モジュールはホスト名をデータレジスタに格納します。
- ⑥ Plus CPU モジュールが DPS と切断します。
- ⑦ Plus CPU モジュールが⑤のデータレジスタに格納されたホスト名を持つ Azure IoT Hub へ接続します。



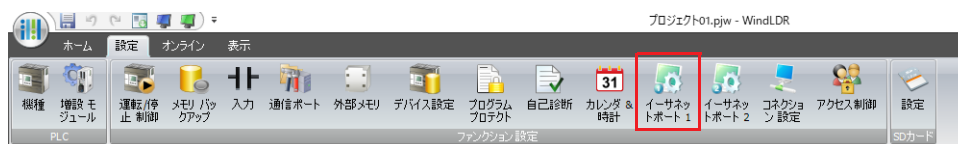
既に登録済みの DPS へ再度接続した場合、DPS の登録グループの再プロビジョニング設定に応じて別の Azure IoT Hub にデバイスが登録される場合があります。詳細は、Azure IoT Hub Device Provisioning Service のドキュメントを参照してください。

MQTT 通信の設定

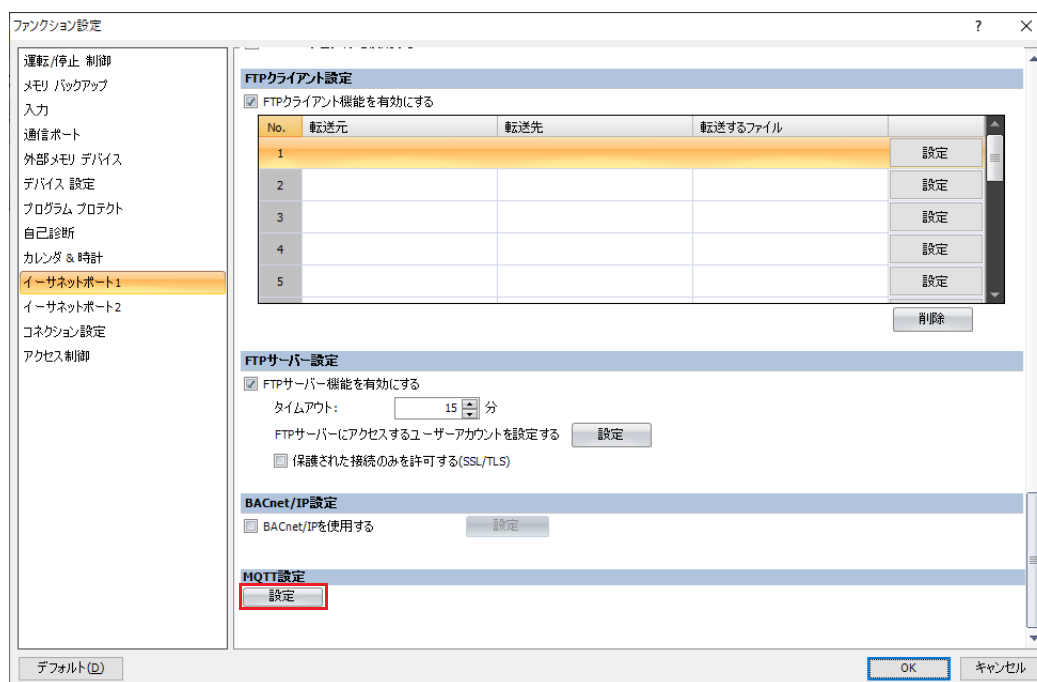
MQTT 通信の設定は [MQTT 設定] ダイアログボックスで行います。ここでは [MQTT 設定] ダイアログボックスと使用方法に応じた設定について説明します。

● [MQTT 設定] ダイアログボックスの表示手順

1. [設定] タブの [ファンクション設定] で [イーサネットポート 1] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。



2. MQTT 設定で [設定] ボタンをクリックします。
[MQTT 設定] ダイアログボックスが表示されます。



[MQTT 設定] ダイアログボックス

接続する MQTT ブローカーおよびクラウドサービスとその接続方法を設定します。

- ① MQTT を有効にする
MQTT を有効にするかどうかを設定します。
- ② クラウドサービス名
接続するクラウドサービス名を設定します。

クラウドサービス名	内容
汎用	汎用のMQTTブローカーへ接続します。 詳細は、「汎用ブローカーへ接続する」(17-7頁)を参照してください。
AWS IoT Core	AWS IoT Coreへ接続します。 詳細は、「AWS IoT Core へ接続する」(17-12頁)を参照してください。
Azure IoT Hub	Azure IoT Hubへ接続します。 Azure IoT Hubとの接続方法は3種類あります。詳細は、「MQTT接続方法」(17-6頁)を参照してください。

MQTT 設定

[MQTT 設定] ダイアログボックスで MQTT を有効にすると、[MQTT 設定] タブが表示されます。
このタブで MQTT 通信を行うための基本設定を行います。

■ MQTT 接続方法

クラウドサービス名に "Azure IoT Hub" を設定した場合のみ、[MQTT 接続方法] グループが表示され、Azure IoT Hub との接続方法を設定します。接続方法は、次の3種類から設定します。

MQTT 接続方法	内容
直接Azure IoT Hubに接続	Shared Access Signagure (SAS) を使用する Azure IoT HubへSASを使って接続します。 詳細は、「Azure IoT HubへSASを使って接続する」(17-17頁)を参照してください。
	X.509証明書を使用する Azure IoT HubへX.509証明書を使って接続します。 詳細は、「Azure IoT HubへX.509証明書を使って接続する」(17-22頁)を参照してください。
DPS経由でAzure IoT Hubに接続	Azure IoT HubへDPS経由で接続します。 詳細は、「Azure IoT HubへDPS経由で接続する」(17-26頁)を参照してください。

汎用ブローカーへ接続する

■ MQTT 基本設定

[SD メモリカードで指定する] チェックボックスがオフの場合

The screenshot shows the 'MQTT設定' (MQTT Settings) dialog box. The 'MQTTを有効にする' (Enable MQTT) checkbox is checked. The 'クラウドサービス名' (Cloud Service Name) is set to '汎用' (General). Under 'MQTT基本設定' (MQTT Basic Settings), the 'SDメモリカードで指定する' (Specify using SD Memory Card) checkbox is unchecked. The 'ブローカー' (Broker) section includes: 'ホスト名' (Host Name) with a text field, 'IPアドレス' (IP Address) with a field containing '0.0.0.0', 'ポート番号' (Port Number) with a dropdown set to '1883', and 'キープアライブ' (Keep Alive) with a dropdown set to '60' seconds. The 'クライアントID' (Client ID) section has '固定値' (Fixed Value) selected with the value '2012448d-7e54-419d-8f55-2a119187319a' and a 'ランダムIDを生成' (Generate Random ID) button. The '認証' (Authentication) section has 'プロローカーと接続するために認証が必要' (Authentication required for connection to broker) checked, with fields for 'アカウント名' (Account Name) and 'パスワード' (Password). The 'SSL/TLS' section has '保護された接続を使用する(SSL/TLS)' (Use protected connection (SSL/TLS)) checked, with fields for 'ルート証明書' (Root Certificate), 'クライアント証明書' (Client Certificate), and 'クライアント秘密鍵' (Client Private Key), each with an 'インポート' (Import) button. The 'デバイス' (Device) section has '接続コントロール' (Connection Control) and '接続ステータス' (Connection Status) fields. At the bottom, there are buttons for 'プロジェクトからインポート' (Import from project), 'OK', and 'キャンセル' (Cancel).

[SD メモリカードで指定する] チェックボックスがオンの場合

The screenshot shows the 'MQTT設定' (MQTT Settings) dialog box. The 'MQTTを有効にする' (Enable MQTT) checkbox is checked. The 'クラウドサービス名' (Cloud Service Name) is set to '汎用' (General). Under 'MQTT基本設定' (MQTT Basic Settings), the 'SDメモリカードで指定する' (Specify using SD Memory Card) checkbox is checked. The 'SDメモリカードを設定する' (Configure SD Memory Card) checkbox is also checked. The 'デバイス' (Device) section is visible at the bottom.

① SD メモリカードで指定する

MQTT 通信を行うための基本設定を、SD メモリカードを使用して Plus CPU モジュールに書き込むかどうかを指定します。
[SD メモリカードで指定する] チェックボックスをオンにすると、②が表示され、③～⑭が非表示になります。この場合、
[SD メモリカード] ダイアログボックスで詳細を設定します。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズマニュアル「第 11 章 SD メモリカード」を参照してください。

② SD メモリカードを設定する

[SD メモリカード] ダイアログボックスを開きます。[SD メモリカードで指定する] チェックボックスがオンの場合のみ表示されます。
[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズマニュアル「第 11 章 SD メモリカード」を参照してください。

③ ホスト名 / IP アドレス

ブローカーのホスト名または IP アドレスを設定します。ホスト名の最大文字数は 128 文字です。英数字および記号 (-) のみ使用できます。

④ ポート番号

ブローカーのポート番号を設定します。通常、MQTT では 1883、MQTT over TLS では 8883 が使用されます。ポート番号は 0 ~ 65535 の範囲で設定します。デフォルト値は 1883 です。ポート番号に 0 を設定した場合、保護された接続を使用する (SSL/TLS) (⑪) のチェックボックスがオフの場合は 1883、オンの場合は 8883 のポート番号を使用します。

⑤キープアライブ

ブローカーと接続中、ブローカーに対して接続確認を実行する時間間隔を設定します。Plus CPU モジュールが送受信を行わない時間が設定した時間間隔以上続いた場合、ブローカーに接続確認を行います。設定できる範囲は 5 ～ 65535 秒です。デフォルト値は 60 秒です。

⑥クライアント ID

クライアント ID を設定します。クライアント ID は、次の 3 種類から設定できます。

クライアント ID	内容
MACアドレス	Plus CPUモジュールのEthernetポート1のMACアドレスを設定します。 ^{*1}
固定値	任意の文字列を設定します。 最大文字数は128文字です。英数字および記号のみ使用できます。
データレジスタ	クライアントIDを格納するためのデータレジスタを設定します。 設定したデータレジスタの上位バイトから順に値を読み出して、文字データとして扱い、クライアントIDとします。 ^{*2} 設定したデータレジスタを先頭に連続して64ワード分のデータレジスタを使用します。 ^{*3} デバイスの範囲を超えないように先頭のデータレジスタを設定してください。クライアントIDには英数字および記号のみ使用できます。

*1 Plus CPU モジュールの Ethernet ポート 1 の MAC アドレスが 12-34-56-78-9A-BC (D8324=0012h、D8325=0034h、D8326=0056h、D8327=0078h、D8328=009Ah、D8329=00BCh) の場合、クライアント ID は "123456789ABC" です。

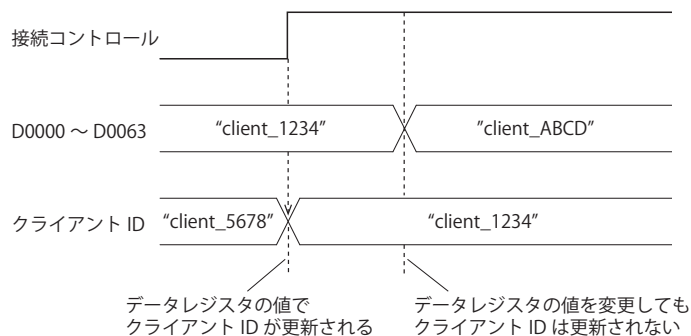
*2 例えば、D0000 を指定し、各データレジスタに次の値が格納されている場合、クライアント ID は "client_1234" です。

データレジスタ	格納値	
	上位バイト	下位バイト
D0000	"c"=63h	"l"=6Ch
D0001	"i"=69h	"e"=65h
D0002	"n"=6Eh	"t"=74h
D0003	"_"=5Fh	"1"=31h
D0004	"2"=32h	"3"=33h
D0005	"4"=34h	00h

*3 設定する文字列が 128 バイト (64 ワード) より短い場合は、文字列の終わりとして終端文字 NULL (00h) を追加してください。指定したデータレジスタの上位バイトから終端文字 NULL (00h) の前までを文字データとして扱い、クライアント ID とします。



- クライアント ID をデータレジスタで設定する場合、MOVC (ムーブキャラクタ) 命令を使用して、データレジスタに文字列を格納することができます。MOVC 命令については、ラダープログラミングマニュアル「第 5 章 MOVC (ムーブキャラクタ)」を参照してください。
- 接続コントロール (⑤) を OFF から ON したとき、Plus CPU モジュール内部で記憶するクライアント ID が更新されます。例えば、クライアント ID をデータレジスタ (D0000) で設定する場合、クライアント ID は次のタイミングで更新されません。



⑦ランダム ID を生成

クライアント ID (⑥) を固定値で設定する場合、このボタンをクリックすると無作為にクライアント ID を生成します。

⑧ブローカーと接続するために認証が必要

ブローカーに接続する際にアカウント名とパスワードによる認証を行うかどうかを設定します。チェックボックスがオンの場合、ブローカーに接続する際にアカウント名とパスワードによる認証を行います。デフォルト値はオフです。

⑨アカウント名 / ⑩パスワード

[ブローカーと接続するために認証が必要] チェックボックスがオンの場合、アカウント名およびパスワードを設定します。アカウント名の最大文字数は 128 文字、パスワードの最大文字数は 496 文字です。英数字および記号のみ使用できます。

⑪保護された接続を使用する (SSL/TLS)

接続先のブローカーで SSL/TLS 通信が必要な場合は、このチェックボックスをオンにします。デフォルト値はオフです。

⑫ルート証明書

[インポート] ボタンをクリックして、ブローカーと SSL/TLS 通信を行う場合に使用するサーバー証明書のルート証明書をインポートします。インポートできるのは、pem または crt ファイルです。WindLDR は、pem および crt ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルは、インポートできません。pem ファイルに複数のルート証明書が含まれている場合、先頭から 2 個のルート証明書を der ファイルに変換します。[保護された接続を使用する (SSL/TLS)]

チェックボックスがオンの場合のみ設定できます。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートしたサーバー証明書のルート証明書の情報が表示されます。

[削除] ボタンをクリックすると、インポートした証明書が削除されます。

⑬クライアント証明書

[インポート] ボタンをクリックして、ブローカーと SSL/TLS 通信を行う場合に使用するクライアント証明書をインポートします。インポートできるのは、pem または crt ファイルです。WindLDR は、pem ファイルおよび crt ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルは、インポートできません。[保護された接続を使用する (SSL/TLS)]

チェックボックスがオンの場合のみ設定できます。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートしたクライアント証明書の情報が表示されます。

[削除] ボタンをクリックすると、インポートした証明書が削除されます。

⑭クライアント秘密鍵

[インポート] ボタンをクリックして、ブローカーと SSL/TLS 通信を行う場合に使用するクライアント秘密鍵をインポートします (暗号化方式 RSA、鍵長は 3072bit まで)。WindLDR は次の形式のファイルに対応しています。

- PKCS#1 形式の pem または der ファイル
- PKCS#8 形式 (暗号化なし) の pem または der ファイル

[保護された接続を使用する (SSL/TLS)] チェックボックスがオンの場合のみ設定できます。

[削除] ボタンをクリックすると、インポートした秘密鍵が削除されます。

■ デバイス

⑮ 接続コントロール

ブローカーとの接続および切断を行うための内部リレーを設定します。設定した内部リレーを ON すると、ブローカーへ接続します。OFF すると、ブローカーとの接続を切断します。



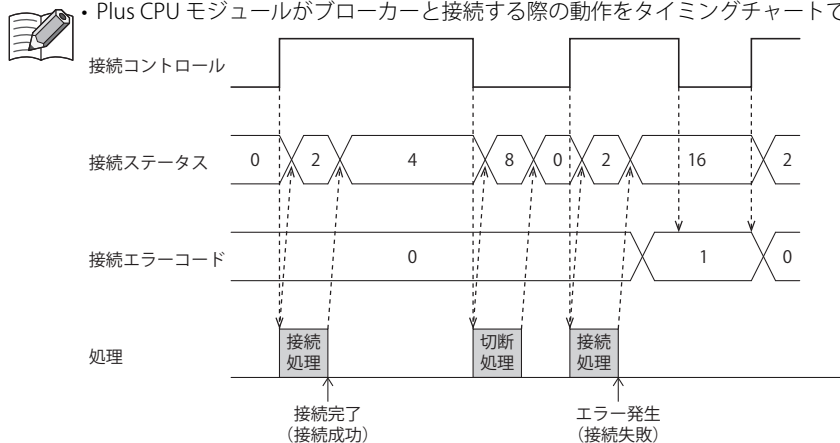
クライアント ID (⑥) をデータレジスタに設定した場合、接続コントロールを OFF から ON したときに、Plus CPU モジュール内部で記憶するクライアント ID が更新されます。

⑯ 接続ステータス

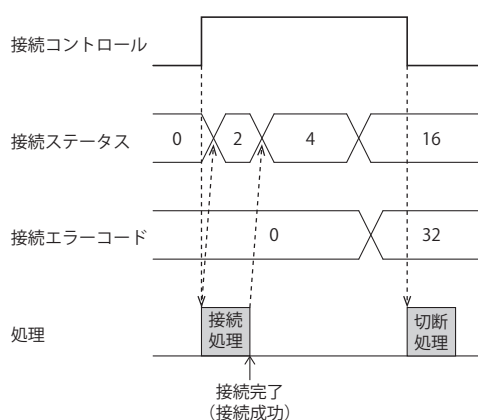
ブローカーに接続する際のステータスおよびエラー情報を格納するデータレジスタを設定します。設定したデータレジスタを先頭に 2 ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

格納先	項目	内容																																
先頭番号+0	接続ステータス	ブローカーに接続する際のステータスを格納します。																																
		<table border="1"> <thead> <tr> <th>ステータスコード</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>0 (0000h)</td> <td>初期状態 (切断状態)</td> </tr> <tr> <td>2 (0002h)</td> <td>接続処理中</td> </tr> <tr> <td>4 (0004h)</td> <td>接続状態</td> </tr> <tr> <td>8 (0008h)</td> <td>切断処理中</td> </tr> <tr> <td>16 (0010h)</td> <td>接続処理エラー</td> </tr> <tr> <td>32 (0020h)</td> <td>切断処理エラー</td> </tr> </tbody> </table>	ステータスコード	状態	0 (0000h)	初期状態 (切断状態)	2 (0002h)	接続処理中	4 (0004h)	接続状態	8 (0008h)	切断処理中	16 (0010h)	接続処理エラー	32 (0020h)	切断処理エラー																		
		ステータスコード	状態																															
		0 (0000h)	初期状態 (切断状態)																															
		2 (0002h)	接続処理中																															
		4 (0004h)	接続状態																															
		8 (0008h)	切断処理中																															
16 (0010h)	接続処理エラー																																	
32 (0020h)	切断処理エラー																																	
先頭番号+1	接続エラーコード	ブローカーに接続する際に発生したエラー情報を格納します。																																
		<table border="1"> <thead> <tr> <th>エラーコード</th> <th>エラー内容</th> </tr> </thead> <tbody> <tr> <td>1 (0001h)</td> <td>イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった</td> </tr> <tr> <td>2 (0002h)</td> <td>SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した</td> </tr> <tr> <td>4 (0004h)</td> <td>クライアントIDのフォーマットが不正である</td> </tr> <tr> <td>16 (0010h)</td> <td>未知のパケットを受信した</td> </tr> <tr> <td>32 (0020h)</td> <td>不正なMQTTパケットを受信した</td> </tr> <tr> <td>64 (0040h)</td> <td>キープアライブタイムアウトエラー</td> </tr> <tr> <td>80 (0050h)</td> <td>宛先ホストに到達できなかった</td> </tr> <tr> <td>96 (0060h)</td> <td>MQTTパケット受信タイムアウトエラー</td> </tr> <tr> <td>112 (0070h)</td> <td>TLSエラー</td> </tr> <tr> <td>256 (0100h)</td> <td>ブローカー接続拒否 (MQTTプロトコルバージョン許容不可)</td> </tr> <tr> <td>512 (0200h)</td> <td>ブローカー接続拒否 (クライアントIDが不正)</td> </tr> <tr> <td>768 (0300h)</td> <td>ブローカー接続拒否 (ブローカー使用不可)</td> </tr> <tr> <td>1024 (0400h)</td> <td>ブローカー接続拒否 (アカウント名またはパスワードが不正)</td> </tr> <tr> <td>1280 (0500h)</td> <td>ブローカー接続拒否 (権限なし)</td> </tr> <tr> <td>32768 (8000h)</td> <td>ブローカー応答異常</td> </tr> </tbody> </table>	エラーコード	エラー内容	1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった	2 (0002h)	SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した	4 (0004h)	クライアントIDのフォーマットが不正である	16 (0010h)	未知のパケットを受信した	32 (0020h)	不正なMQTTパケットを受信した	64 (0040h)	キープアライブタイムアウトエラー	80 (0050h)	宛先ホストに到達できなかった	96 (0060h)	MQTTパケット受信タイムアウトエラー	112 (0070h)	TLSエラー	256 (0100h)	ブローカー接続拒否 (MQTTプロトコルバージョン許容不可)	512 (0200h)	ブローカー接続拒否 (クライアントIDが不正)	768 (0300h)	ブローカー接続拒否 (ブローカー使用不可)	1024 (0400h)	ブローカー接続拒否 (アカウント名またはパスワードが不正)	1280 (0500h)	ブローカー接続拒否 (権限なし)	32768 (8000h)	ブローカー応答異常
		エラーコード	エラー内容																															
		1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった																															
		2 (0002h)	SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した																															
		4 (0004h)	クライアントIDのフォーマットが不正である																															
		16 (0010h)	未知のパケットを受信した																															
		32 (0020h)	不正なMQTTパケットを受信した																															
		64 (0040h)	キープアライブタイムアウトエラー																															
		80 (0050h)	宛先ホストに到達できなかった																															
		96 (0060h)	MQTTパケット受信タイムアウトエラー																															
		112 (0070h)	TLSエラー																															
		256 (0100h)	ブローカー接続拒否 (MQTTプロトコルバージョン許容不可)																															
		512 (0200h)	ブローカー接続拒否 (クライアントIDが不正)																															
		768 (0300h)	ブローカー接続拒否 (ブローカー使用不可)																															
		1024 (0400h)	ブローカー接続拒否 (アカウント名またはパスワードが不正)																															
1280 (0500h)	ブローカー接続拒否 (権限なし)																																	
32768 (8000h)	ブローカー応答異常																																	

• Plus CPU モジュールがブローカーと接続する際の動作をタイミングチャートで示します。



• 接続エラーコード 16 (0010h) ~ 112 (0070h) はブローカーと接続した状態 (接続ステータス 4 (0004h)) でも発生する場合があります。



• STOP 中は MQTT 通信を行いません。ブローカーとの接続中に RUN から STOP した場合は接続コントロールを OFF して接続ステータスに切断状態 0 (0000h) を格納します。

- ⑰ [プロジェクトからインポート] ボタン
WindLDR のプロジェクトファイルから、MQTT 設定のみをインポートします。
- ⑱ [OK] ボタン
設定を保存し、[MQTT 設定] ダイアログボックスを閉じます。



クライアント ID (⑥) をデータレジスタに設定している場合、[OK] ボタンを押すと、クライアント ID を 1 バイトとしてトピックのサイズを計算します。設定したトピックが 256 バイトを超える場合、エラーとなります。

- ⑲ [キャンセル] ボタン
設定を保存せずに、[MQTT 設定] ダイアログボックスを閉じます。

AWS IoT Core へ接続する

■ MQTT 基本設定

[SD メモリカードで指定する] チェックボックスがオフの場合

MQTT設定

MQTTを有効にする
クラウドサービス名: AWS IoT Core

MQTT基本設定

① SDメモリカードで指定する

AWS IoT Core

③ エンドポイント:

④ ポート番号:

⑤ キープアライブ: 秒

⑥ クライアントID:

MACアドレス:

固定値:

⑦

データレジスタ:

保護された接続 (SSL/TLS)

⑧ ルート証明書: インポート済

⑨ クライアント証明書: インポートされていません

⑩ クライアント秘密鍵: インポートされていません

デバイス

⑪ 接続コントロール:

⑫ 接続ステータス:

⑬ プロジェクトからインポート⑬

⑭ OK

⑮ キャンセル

[SD メモリカードで指定する] チェックボックスがオンの場合

MQTT設定

MQTTを有効にする
クラウドサービス名: AWS IoT Core

MQTT基本設定

SDメモリカードで指定する ②

デバイス

① SD メモリカードで指定する

MQTT 通信を行うための基本設定を、SD メモリカードを使用して Plus CPU モジュールに書き込むかどうかを指定します。
[SD メモリカードで指定する] チェックボックスをオンにすると、②が表示され、③～⑩が非表示になります。この場合、
[SD メモリカード] ダイアログボックスで詳細を設定します。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

② SD メモリカードを設定する

[SD メモリカード] ダイアログボックスを開きます。[SD メモリカードで指定する] チェックボックスがオンの場合のみ表示されます。
[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

③ エンドポイント

AWS IoT Core のエンドポイントを設定します。エンドポイントの最大文字数は 128 文字です。英数字および記号 (-) のみ使用できます。

④ ポート番号

AWS IoT Core のポート番号を設定します。ポート番号は 0 ～ 65535 の範囲で設定します。デフォルト値は AWS IoT Core で使われる 8883 です。ポート番号に 0 を設定した場合、8883 のポート番号を使用します。

⑤ キープアライブ

AWS IoT Core と接続中、AWS IoT Core に対して接続確認を実行する時間間隔を設定します。Plus CPU モジュールが送受信を行わない時間が設定した時間間隔以上続いた場合、AWS IoT Core に接続確認を行います。設定できる範囲は 5 ～ 65535 秒です。デフォルト値は 60 秒です。

⑥クライアント ID

クライアント ID を設定します。クライアント ID は、次の 3 種類から設定できます。

クライアント ID	内容
MACアドレス	Plus CPUモジュールのEthernetポート1のMACアドレスを設定します。 ^{*1}
固定値	任意の文字列を設定します。 最大文字数は128文字です。英数字および記号のみ使用できます。
データレジスタ	クライアントIDを格納するためのデータレジスタを設定します。 設定したデータレジスタの上位バイトから順に値を読み出して、文字データとして扱い、クライアントIDとします。 ^{*2} 設定したデータレジスタを先頭に連続して64ワード分のデータレジスタを使用します。 ^{*3} デバイスの範囲を超えないように先頭のデータレジスタを設定してください。クライアントIDには英数字および記号のみ使用できます。

*1 Plus CPU モジュールの Ethernet ポート 1 の MAC アドレスが 12-34-56-78-9A-BC (D8324=0012h、D8325=0034h、D8326=0056h、D8327=0078h、D8328=009Ah、D8329=00BCh) の場合、クライアント ID は "123456789ABC" です。

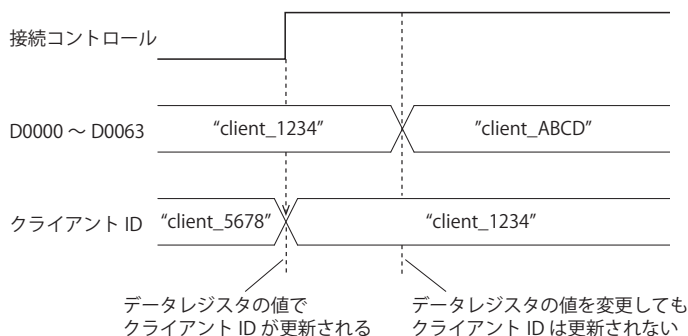
*2 例えば、D0000 を指定し、各データレジスタに次の値が格納されている場合、クライアント ID は "client_1234" です。

データレジスタ	格納値	
	上位バイト	下位バイト
D0000	"c"=63h	"l"=6Ch
D0001	"i"=69h	"e"=65h
D0002	"n"=6Eh	"t"=74h
D0003	"_"=5Fh	"1"=31h
D0004	"2"=32h	"3"=33h
D0005	"4"=34h	00h

*3 設定する文字列が 128 バイト (64 ワード) より短い場合は、文字列の終わりとして終端文字 NULL (00h) を追加してください。指定したデータレジスタの上位バイトから終端文字 NULL (00h) の前までを文字データとして扱い、クライアント ID とします。



- クライアント ID をデータレジスタで設定する場合、MOVC (ムーブキャラクタ) 命令を使用して、データレジスタに文字列を格納することができます。MOVC 命令については、ラダープログラミング マニュアル「第 5 章 MOVC (ムーブキャラクタ)」を参照してください。
- 接続コントロール (⑩) を OFF から ON したとき、Plus CPU モジュール内部で記憶するクライアント ID が更新されます。例えば、クライアント ID をデータレジスタ (D0000) で設定する場合、クライアント ID は次のタイミングで更新されます。



⑦ランダム ID を生成

クライアント ID (⑥) を固定値で設定する場合、このボタンをクリックすると無作為にクライアント ID を生成します。

⑧ルート証明書

[インポート] ボタンをクリックして、AWS IoT Core と SSL/TLS 通信を行う場合に使用するルート証明書をインポートできます。インポートできるのは pem または crt ファイルです。WindLDR は pem および crt ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルはインポートできません。pem ファイルに複数のルート証明書が含まれている場合、先頭から 2 個のルート証明書を der ファイルに変換します。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートしたルート証明書の情報が表示されます。

[削除] ボタンをクリックすると、インポートした証明書が削除されます。

⑨クライアント証明書

[インポート] ボタンをクリックして、AWS IoT Core と SSL/TLS 通信を行う場合に使用するクライアント証明書をインポートします。インポートできるのは pem または crt ファイルです。WindLDR は pem および crt ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルはインポートできません。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートしたクライアント証明書の情報が表示されます。

[削除] ボタンをクリックすると、インポートした証明書が削除されます。

⑩クライアント秘密鍵

[インポート] ボタンをクリックして、AWS IoT Core と SSL/TLS 通信を行う場合に使用するクライアント秘密鍵をインポートします（暗号化方式 RSA、鍵長は 3072bit まで）。WindLDR は次の形式のファイルに対応しています。

- PKCS#1 形式の pem または der ファイル
- PKCS#8 形式（暗号化なし）の pem または der ファイル

[削除] ボタンをクリックすると、インポートした秘密鍵が削除されます。

■ デバイス

① 接続コントロール

AWS IoT Core との接続および切断を行うための内部リレーを設定します。設定した内部リレーを ON すると、AWS IoT Core へ接続します。OFF すると、AWS IoT Core との接続を切断します。



クライアント ID (⑥) をデータレジスタで設定した場合、接続コントロールを OFF から ON したときに、Plus CPU モジュール内部で記憶するクライアント ID が更新されます。

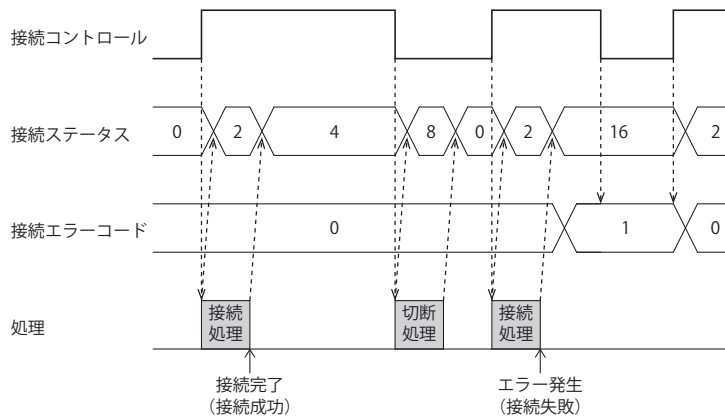
② 接続ステータス

AWS IoT Core に接続する際のステータスおよびエラー情報を格納するデータレジスタを設定します。設定したデータレジスタを先頭に 2 ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

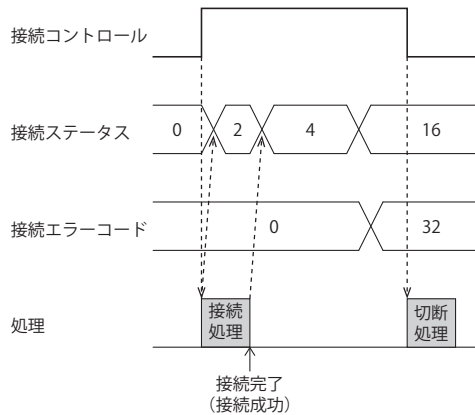
格納先	項目	内容																																
先頭番号+0	接続ステータス	AWS IoT Core に接続する際のステータスを格納します。																																
		<table border="1"> <thead> <tr> <th>ステータスコード</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>0 (0000h)</td> <td>初期状態 (切断状態)</td> </tr> <tr> <td>2 (0002h)</td> <td>接続処理中</td> </tr> <tr> <td>4 (0004h)</td> <td>接続状態</td> </tr> <tr> <td>8 (0008h)</td> <td>切断処理中</td> </tr> <tr> <td>16 (0010h)</td> <td>接続処理エラー</td> </tr> <tr> <td>32 (0020h)</td> <td>切断処理エラー</td> </tr> </tbody> </table>	ステータスコード	状態	0 (0000h)	初期状態 (切断状態)	2 (0002h)	接続処理中	4 (0004h)	接続状態	8 (0008h)	切断処理中	16 (0010h)	接続処理エラー	32 (0020h)	切断処理エラー																		
		ステータスコード	状態																															
		0 (0000h)	初期状態 (切断状態)																															
		2 (0002h)	接続処理中																															
		4 (0004h)	接続状態																															
		8 (0008h)	切断処理中																															
16 (0010h)	接続処理エラー																																	
32 (0020h)	切断処理エラー																																	
先頭番号+1	接続エラーコード	AWS IoT Core に接続する際に発生したエラー情報を格納します。																																
		<table border="1"> <thead> <tr> <th>エラーコード</th> <th>エラー内容</th> </tr> </thead> <tbody> <tr> <td>1 (0001h)</td> <td>イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった</td> </tr> <tr> <td>2 (0002h)</td> <td>SDメモ리카ードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した</td> </tr> <tr> <td>4 (0004h)</td> <td>クライアントIDのフォーマットが不正である</td> </tr> <tr> <td>16 (0010h)</td> <td>未知の packets を受信した</td> </tr> <tr> <td>32 (0020h)</td> <td>不正な MQTT パケットを受信した</td> </tr> <tr> <td>64 (0040h)</td> <td>キープアライブタイムアウトエラー</td> </tr> <tr> <td>80 (0050h)</td> <td>宛先ホストに到達できなかった</td> </tr> <tr> <td>96 (0060h)</td> <td>MQTT パケット受信タイムアウトエラー</td> </tr> <tr> <td>112 (0070h)</td> <td>TLS エラー</td> </tr> <tr> <td>256 (0100h)</td> <td>AWS IoT Core 接続拒否 (MQTT プロトコルバージョン許容不可)</td> </tr> <tr> <td>512 (0200h)</td> <td>AWS IoT Core 接続拒否 (クライアントIDが不正)</td> </tr> <tr> <td>768 (0300h)</td> <td>AWS IoT Core 接続拒否 (AWS IoT Core 使用不可)</td> </tr> <tr> <td>1024 (0400h)</td> <td>AWS IoT Core 接続拒否 (アカウント名またはパスワードが不正)</td> </tr> <tr> <td>1280 (0500h)</td> <td>AWS IoT Core 接続拒否 (権限なし)</td> </tr> <tr> <td>32768 (8000h)</td> <td>AWS IoT Core 応答異常</td> </tr> </tbody> </table>	エラーコード	エラー内容	1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった	2 (0002h)	SDメモ리카ードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した	4 (0004h)	クライアントIDのフォーマットが不正である	16 (0010h)	未知の packets を受信した	32 (0020h)	不正な MQTT パケットを受信した	64 (0040h)	キープアライブタイムアウトエラー	80 (0050h)	宛先ホストに到達できなかった	96 (0060h)	MQTT パケット受信タイムアウトエラー	112 (0070h)	TLS エラー	256 (0100h)	AWS IoT Core 接続拒否 (MQTT プロトコルバージョン許容不可)	512 (0200h)	AWS IoT Core 接続拒否 (クライアントIDが不正)	768 (0300h)	AWS IoT Core 接続拒否 (AWS IoT Core 使用不可)	1024 (0400h)	AWS IoT Core 接続拒否 (アカウント名またはパスワードが不正)	1280 (0500h)	AWS IoT Core 接続拒否 (権限なし)	32768 (8000h)	AWS IoT Core 応答異常
		エラーコード	エラー内容																															
		1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった																															
		2 (0002h)	SDメモ리카ードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した																															
		4 (0004h)	クライアントIDのフォーマットが不正である																															
		16 (0010h)	未知の packets を受信した																															
		32 (0020h)	不正な MQTT パケットを受信した																															
		64 (0040h)	キープアライブタイムアウトエラー																															
		80 (0050h)	宛先ホストに到達できなかった																															
		96 (0060h)	MQTT パケット受信タイムアウトエラー																															
		112 (0070h)	TLS エラー																															
		256 (0100h)	AWS IoT Core 接続拒否 (MQTT プロトコルバージョン許容不可)																															
		512 (0200h)	AWS IoT Core 接続拒否 (クライアントIDが不正)																															
		768 (0300h)	AWS IoT Core 接続拒否 (AWS IoT Core 使用不可)																															
1024 (0400h)	AWS IoT Core 接続拒否 (アカウント名またはパスワードが不正)																																	
1280 (0500h)	AWS IoT Core 接続拒否 (権限なし)																																	
32768 (8000h)	AWS IoT Core 応答異常																																	



• Plus CPU モジュールが AWS IoT Core と接続する際の動作をタイミングチャートで示します。



• 接続エラーコード 16 (0010h) ~ 112 (0070h) は AWS IoT Core と接続した状態 (接続ステータス 4 (0004h)) でも発生する場合があります。



• STOP 中は MQTT 通信を行いません。AWS IoT Core との接続中に RUN → STOP した場合は接続コントロールを OFF して接続ステータスに切断状態 0 (0000h) を格納します。

- ⑬ [プロジェクトからインポート] ボタン
WindLDR のプロジェクトファイルから、MQTT 設定のみをインポートします。
- ⑭ [OK] ボタン
設定を保存し、[MQTT 設定] ダイアログボックスを閉じます。



クライアント ID (⑥) をデータレジスタに設定している場合、[OK] ボタンを押すと、クライアント ID を 1 バイトとしてトピックのサイズを計算します。設定したトピックが 256 バイトを超える場合、エラーとなります。

- ⑮ [キャンセル] ボタン
設定を保存せずに、[MQTT 設定] ダイアログボックスを閉じます。

Azure IoT Hub へ SAS を使って接続する

■ MQTT 接続方法

[MQTT 接続方法] グループで “直接 Azure IoT Hub に接続” を選択して、“Shared Access Signature (SAS) を使用する” を設定します。

■ MQTT 基本設定

[SD メモリカードで指定する] チェックボックスがオフの場合

The screenshot shows the 'MQTT 設定' (MQTT Settings) dialog box. The 'MQTT 接続方法' (MQTT Connection Method) section has '直接 Azure IoT Hub に接続' (Connect directly to Azure IoT Hub) selected, with 'Shared Access Signatures (SAS) を使用する' (Use Shared Access Signatures) chosen from the dropdown. The 'MQTT 基本設定' (MQTT Basic Settings) section has the 'SDメモリカードで指定する' (Specify using SD memory card) checkbox turned off. Below it, the '接続文字列' (Connection string) field is empty, and the 'Azure IoT Hub' section shows 'ホスト名' (Host name) as N/A, 'ポート番号' (Port number) as 8883, and 'キーアライブ' (Key alive) as 60 seconds. The 'デバイス' (Device) section shows 'デバイスID' (Device ID) as N/A. The '保護された接続 (SSL/TLS)' (Secured connection) section shows 'ルート証明書' (Root certificate) as 'インポート済' (Imported), with 'インポート' (Import), '詳細' (Details), and '削除' (Delete) buttons. The 'デバイス' (Device) section at the bottom has '接続コントロール' (Connection control) and '接続ステータス' (Connection status) fields. The dialog box has a 'プロジェクトからインポート' (Import from project) button at the bottom left and 'OK' and 'キャンセル' (Cancel) buttons at the bottom right.

[SD メモリカードで指定する] チェックボックスがオンの場合

The screenshot shows the 'MQTT 設定' (MQTT Settings) dialog box. The 'MQTT 基本設定' (MQTT Basic Settings) section has the 'SDメモリカードで指定する' (Specify using SD memory card) checkbox turned on. A button labeled 'SDメモリカードを設定する' (Set SD memory card) is visible next to the checkbox. The 'デバイス' (Device) section is highlighted in blue.

① SD メモリカードで指定する

MQTT 通信を行うための基本設定を、SD メモリカードを使用して Plus CPU モジュールに書き込むかどうかを指定します。

[SD メモリカードで指定する] チェックボックスをオンにすると、②が表示され、③～⑧が非表示になります。この場合、[SD メモリカード] ダイアログボックスで詳細を設定します。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズマニュアル「第 11 章 SD メモリカード」を参照してください。

② SD メモリカードを設定する

[SD メモリカード] ダイアログボックスを開きます。[SD メモリカードで指定する] チェックボックスがオンの場合のみ表示されます。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズマニュアル「第 11 章 SD メモリカード」を参照してください。

③ 接続文字列

Azure IoT Hub でデバイスに割り当てられた接続文字列（プライマリ接続文字列またはセカンダリ接続文字列）を設定します。接続文字列には Azure IoT Hub のホスト名、デバイス ID および SharedAccessKey が含まれます。接続文字列の最大文字数は 300 文字です。英数字および記号のみ使用できます。

④ ホスト名

接続文字列に記載された HostName の値を表示します。

例えば、HostName=abcd.azure-devices.net の場合、abcd.azure-devices.net をホスト名として表示します。

⑤ポート番号

Azure IoT Hub のポート番号を設定します。ポート番号は 0 ～ 65535 の範囲で設定します。デフォルト値は Azure IoT Hub で使われる 8883 です。ポート番号に 0 を設定した場合、8883 のポート番号を使用します。

⑥キープアライブ

Azure IoT Hub と接続中、Azure IoT Hub に対して接続確認を実行する時間間隔を設定します。Plus CPU モジュールが送受信を行わない時間が設定した時間間隔以上続いた場合、Azure IoT Hub に接続確認を行います。設定できる範囲は 5 ～ 65535 秒です。デフォルト値は 60 秒です。

⑦デバイス ID

接続文字列に記載された DeviceId の値を表示します。

例えば、DeviceId=1234 の場合、1234 をデバイス ID として表示します。デバイス ID は、MQTT 通信のクライアント ID として使用します。

⑧ルート証明書

[インポート] ボタンをクリックして、Azure IoT Hub と SSL/TLS 通信を行う場合に使用するルート証明書をインポートできます。インポートできるのは pem または crt ファイルです。WindLDR は pem および crt ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルはインポートできません。pem ファイルに複数のルート証明書が含まれている場合、先頭から 2 個のルート証明書を der ファイルに変換します。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートしたルート証明書の情報が表示されます。

[削除] ボタンをクリックすると、インポートした証明書が削除されます。

■デバイス

⑨接続コントロール

Azure IoT Hub との接続および切断に関する操作を行うための内部リレーを設定します。設定した内部リレーを先頭に 5 ビット分の内部リレーを使用します。デバイスの範囲を超えないように先頭の内部リレーを設定してください。

格納先	項目	内容
先頭番号+0	接続コントロール*1*2	Azure IoT Hub との接続および切断を行います。ONすると、Azure IoT Hubへ接続します。OFFすると、Azure IoT Hubとの接続を切断します。
先頭番号+1	リザーブ	
先頭番号+2	リザーブ	
先頭番号+3	リザーブ	
先頭番号+4	リザーブ	

*1 Azure IoT Hub と接続するために SAS トークンを生成するとき、Plus CPU モジュールの時計から算出した現在時刻 (UTC) を使用します。

Plus CPU モジュールの時計が実際の現在時刻から大きくずれている場合、Azure IoT Hub と接続できない場合があります。Plus CPU モジュールの時計および時計に関する機能を設定してください。

- SNTP サーバーから現在時刻 (UTC) を取得する場合

次の設定項目を参照し、Plus CPU モジュールの時計および時計に関する機能を設定してください。

設定項目	内容	参照先
SNTP設定	SNTPサーバーから現在時刻 (UTC) を取得して、Plus CPUモジュールの時計を調整します。	「第3章 SNTP設定」(3-11頁)
タイムゾーン	標準時に対する地域の時間帯ごとの差を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 タイムゾーン」
サマータイム	サマータイムの期間を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 サマータイム」




Plus CPU モジュールの時計は時間の経過とともに、精度に応じたずれが生じます。最後に SNTP サーバーから現在時刻 (UTC) を取得してから 6 か月以上が経過すると、Azure IoT Hub と接続できなくなる場合があります。適切な頻度で SNTP サーバーから現在時刻 (UTC) を取得してください。

- 直接、Plus CPU モジュールの時計を設定する場合

次の設定項目を参照し、Plus CPU モジュールの時計および時計に関する機能を設定してください。

設定項目	内容	参照先
時計機能	WindLDRまたは特殊データレジスタおよび特殊内部リレーを使って、Plus CPUモジュールの時計を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 時計機能」
タイムゾーン	標準時に対する地域の時間帯ごとの差を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 タイムゾーン」
サマータイム	サマータイムの期間を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 サマータイム」

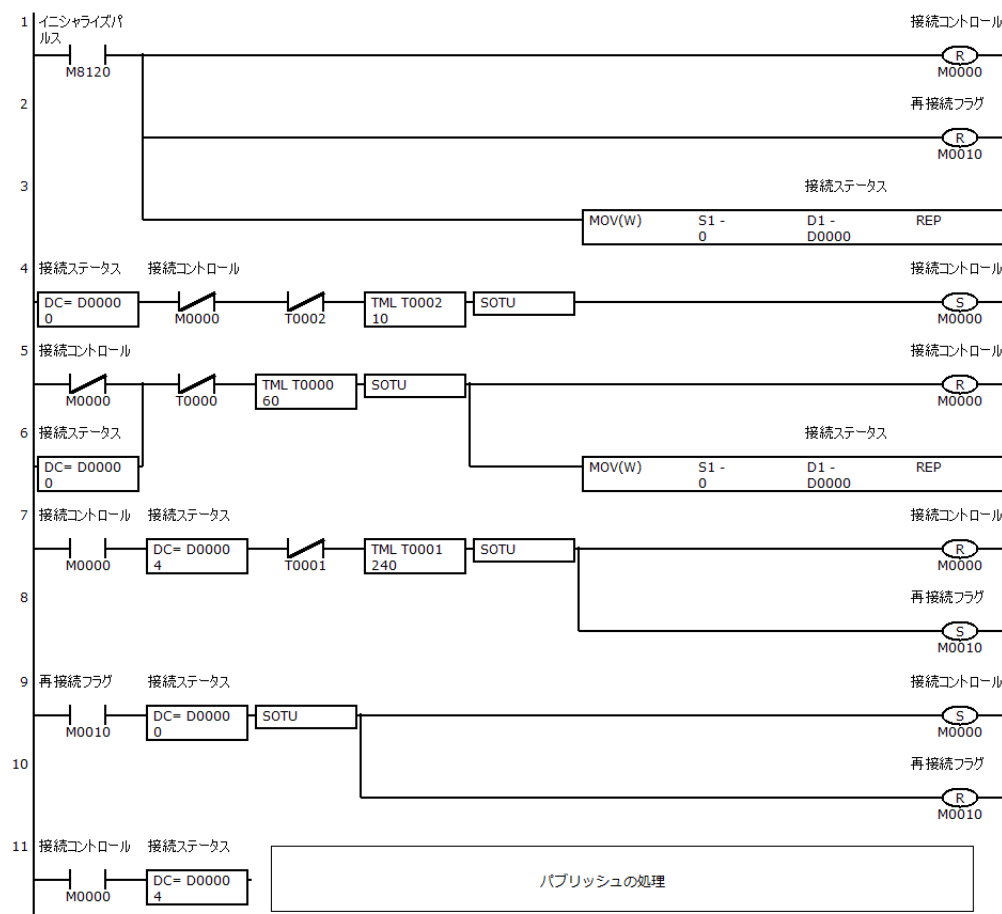
 Plus CPU モジュールの時計は時間の経過とともに、精度に応じたずれが生じます。最後に時計を設定してから6か月以上が経過すると、Azure IoT Hub と接続できなくなる場合があります。適切な頻度で時計を設定してください。

*2 Plus CPU モジュールが生成する SAS トークンは、生成してから5分後に有効期限が切れます。Azure IoT Hub に接続した状態で有効期限が切れると、Azure IoT Hub 応答異常（「@接続ステータス」(17-20 頁)の接続エラーコード参照)が発生し、Azure IoT Hub と切断します。引き続き接続する場合は、再度、接続処理を行ってください。

• SAS トークンの有効期限が切れる前に、Azure IoT Hub に再接続するラダープログラムの一例を示します。

ラダープログラムのパブリッシュは、次の条件をすべて満たすときに実行してください。

- 接続コントロールが ON である
- 接続ステータスが接続状態である（ステータスコードが4 (0004h)である）



ラダー行	概略
1～3	Plus CPUモジュールの電源投入時に、接続に関するパラメータを初期化します。
4	接続ステータスが初期状態（D0000に0 (0000h)が格納されている）で、接続コントロールがOFFのとき、10秒後に接続を開始します。
5,6	切断状態が続いた場合、接続に関するパラメータを初期化します。 次のどちらかの状態が60秒以上継続された場合、M0000をOFFし、D0000に0 (0000h)を格納します。 ・接続コントロールがOFFである（M0000がOFFである） ・接続ステータスが切断状態である（D0000が0 (0000h)である）
7～10	接続ステータスが切断状態のとき、M0000をONします。 接続してから240秒後（有効期限の60秒前）に再接続処理を行います。
11	再接続中、パブリッシュの処理を行います。

⑩接続ステータス

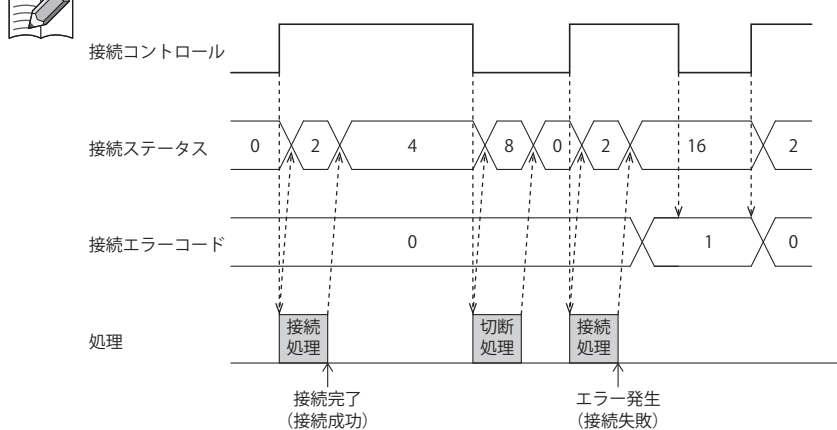
Azure IoT Hub に接続する際のステータスおよびエラー情報を格納するデータレジスタを設定します。設定したデータレジスタを先頭に5ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

格納先	項目	内容																														
先頭番号+0	接続ステータス	Azure IoT Hub に接続する際のステータスを格納します。																														
		<table border="1"> <thead> <tr> <th>ステータスコード</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>0 (0000h)</td> <td>初期状態 (切断状態)</td> </tr> <tr> <td>2 (0002h)</td> <td>接続処理中</td> </tr> <tr> <td>4 (0004h)</td> <td>接続状態</td> </tr> <tr> <td>8 (0008h)</td> <td>切断処理中</td> </tr> <tr> <td>16 (0010h)</td> <td>接続処理エラー</td> </tr> <tr> <td>32 (0020h)</td> <td>切断処理エラー</td> </tr> </tbody> </table>	ステータスコード	状態	0 (0000h)	初期状態 (切断状態)	2 (0002h)	接続処理中	4 (0004h)	接続状態	8 (0008h)	切断処理中	16 (0010h)	接続処理エラー	32 (0020h)	切断処理エラー																
		ステータスコード	状態																													
		0 (0000h)	初期状態 (切断状態)																													
		2 (0002h)	接続処理中																													
		4 (0004h)	接続状態																													
		8 (0008h)	切断処理中																													
		16 (0010h)	接続処理エラー																													
32 (0020h)	切断処理エラー																															
先頭番号+1	接続エラーコード	Azure IoT Hub に接続する際に発生したエラー情報を格納します。																														
		<table border="1"> <thead> <tr> <th>エラーコード</th> <th>エラー内容</th> </tr> </thead> <tbody> <tr> <td>1 (0001h)</td> <td>イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった</td> </tr> <tr> <td>2 (0002h)</td> <td>SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した</td> </tr> <tr> <td>16 (0010h)</td> <td>未知のパケットを受信した</td> </tr> <tr> <td>32 (0020h)</td> <td>不正なMQTTパケットを受信した</td> </tr> <tr> <td>64 (0040h)</td> <td>キープアライブタイムアウトエラー</td> </tr> <tr> <td>80 (0050h)</td> <td>宛先ホストに到達できなかった</td> </tr> <tr> <td>96 (0060h)</td> <td>MQTTパケット受信タイムアウトエラー</td> </tr> <tr> <td>112 (0070h)</td> <td>TLSエラー</td> </tr> <tr> <td>256 (0100h)</td> <td>Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)</td> </tr> <tr> <td>512 (0200h)</td> <td>Azure IoT Hub 接続拒否 (デバイスIDが不正) *1</td> </tr> <tr> <td>768 (0300h)</td> <td>Azure IoT Hub 接続拒否 (Azure IoT Hub 使用不可)</td> </tr> <tr> <td>1024 (0400h)</td> <td>Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正) *1*2</td> </tr> <tr> <td>1280 (0500h)</td> <td>Azure IoT Hub 接続拒否 (権限なし)</td> </tr> <tr> <td>32768 (8000h)</td> <td>Azure IoT Hub 応答異常</td> </tr> </tbody> </table>	エラーコード	エラー内容	1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった	2 (0002h)	SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した	16 (0010h)	未知のパケットを受信した	32 (0020h)	不正なMQTTパケットを受信した	64 (0040h)	キープアライブタイムアウトエラー	80 (0050h)	宛先ホストに到達できなかった	96 (0060h)	MQTTパケット受信タイムアウトエラー	112 (0070h)	TLSエラー	256 (0100h)	Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)	512 (0200h)	Azure IoT Hub 接続拒否 (デバイスIDが不正) *1	768 (0300h)	Azure IoT Hub 接続拒否 (Azure IoT Hub 使用不可)	1024 (0400h)	Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正) *1*2	1280 (0500h)	Azure IoT Hub 接続拒否 (権限なし)	32768 (8000h)	Azure IoT Hub 応答異常
		エラーコード	エラー内容																													
		1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった																													
		2 (0002h)	SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した																													
		16 (0010h)	未知のパケットを受信した																													
		32 (0020h)	不正なMQTTパケットを受信した																													
		64 (0040h)	キープアライブタイムアウトエラー																													
		80 (0050h)	宛先ホストに到達できなかった																													
		96 (0060h)	MQTTパケット受信タイムアウトエラー																													
		112 (0070h)	TLSエラー																													
		256 (0100h)	Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)																													
		512 (0200h)	Azure IoT Hub 接続拒否 (デバイスIDが不正) *1																													
		768 (0300h)	Azure IoT Hub 接続拒否 (Azure IoT Hub 使用不可)																													
		1024 (0400h)	Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正) *1*2																													
1280 (0500h)	Azure IoT Hub 接続拒否 (権限なし)																															
32768 (8000h)	Azure IoT Hub 応答異常																															
先頭番号+2	リザーブ																															
先頭番号+3	リザーブ																															
先頭番号+4	リザーブ																															

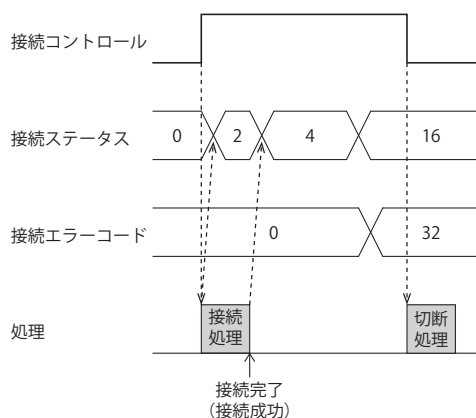
*1 接続文字列に記載された DeviceId の値を確認してください。

*2 Plus CPU モジュールの時計および時計に関する機能を設定してください。

- Plus CPU モジュールが Azure IoT Hub と接続する際の動作をタイミングチャートで示します。



- 接続エラーコード 16 (0010h) ~ 112 (0070h) は Azure IoT Hub と接続した状態 (接続ステータス 4 (0004h)) でも発生する場合があります。



- STOP 中は MQTT 通信を行いません。Azure IoT Hub との接続中に RUN から STOP した場合は接続コントロールを OFF して接続ステータスに切断状態 0 (0000h) を格納します。

- ① [プロジェクトからインポート] ボタン
WindLDR のプロジェクトファイルから、MQTT 設定のみをインポートします。
- ② [OK] ボタン
設定を保存し、[MQTT 設定] ダイアログボックスを閉じます。
- ③ [キャンセル] ボタン
設定を保存せずに、[MQTT 設定] ダイアログボックスを閉じます。

Azure IoT Hub へ X.509 証明書を使って接続する

■ MQTT 接続方法

[MQTT 接続方法] グループで “直接 Azure IoT Hub に接続” を選択して、“X.509 証明書を使用する” を設定します。

■ MQTT 基本設定

[SD メモリカードで指定する] チェックボックスがオフの場合

The screenshot shows the 'MQTT 設定' (MQTT Settings) dialog box. The 'MQTT 有効にする' (Enable MQTT) checkbox is checked. The 'クラウドサービス名' (Cloud Service Name) is set to 'Azure IoT Hub'. Under 'MQTT 接続方法' (MQTT Connection Method), '直接 Azure IoT Hub に接続' (Connect directly to Azure IoT Hub) is selected, and 'X.509 証明書を使用する' (Use X.509 certificate) is chosen. In the 'MQTT 基本設定' (MQTT Basic Settings) section, the 'SD メモリカードで指定する' (Specify using SD memory card) checkbox is unchecked. The 'Azure IoT Hub' section includes fields for 'ホスト名' (Host Name), 'ポート番号' (Port Number) set to 8883, and 'キープアライブ' (Keep Alive) set to 60 seconds. The 'デバイス' (Device) section has a 'デバイスID' (Device ID) field. Below that, a 'ブローカーと接続するために認証が必要' (Authentication required to connect to the broker) section contains an 'アカウント名' (Account Name) field and a '生成' (Generate) button. The '保護された接続 (SSL/TLS)' (Secured connection (SSL/TLS)) section has three rows: 'ルート証明書' (Root Certificate) with an 'インポート済' (Imported) status and buttons for 'インポート', '詳細', and '削除'; 'クライアント証明書' (Client Certificate) with an 'インポートされていません' (Not imported) status and buttons for 'インポート', '詳細', and '削除'; and 'クライアント秘密鍵' (Client Private Key) with an 'インポートされていません' (Not imported) status and buttons for 'インポート' and '削除'. At the bottom, the 'デバイス' (Device) section has '接続コントロール' (Connection Control) and '接続ステータス' (Connection Status) fields. The dialog has 'プロジェクトからインポート' (Import from project), 'OK', and 'キャンセル' (Cancel) buttons.

[SD メモリカードで指定する] チェックボックスがオンの場合

The screenshot shows the 'MQTT 設定' (MQTT Settings) dialog box. The 'MQTT 基本設定' (MQTT Basic Settings) section is expanded, and the 'SD メモリカードで指定する' (Specify using SD memory card) checkbox is checked. A button labeled 'SDメモリカードを設定する' (Configure SD memory card) is visible next to it. The 'デバイス' (Device) section is also visible at the bottom.

① SD メモリカードで指定する

MQTT 通信を行うための基本設定を、SD メモリカードを使用して Plus CPU モジュールに書き込むかどうかを指定します。

[SD メモリカードで指定する] チェックボックスをオンにすると、②が表示され、③～⑪が非表示になります。この場合、[SD メモリカード] ダイアログボックスで詳細を設定します。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

② SD メモリカードを設定する

[SD メモリカード] ダイアログボックスを開きます。[SD メモリカードで指定する] チェックボックスがオンの場合のみ表示されます。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

③ ホスト名

Azure IoT Hub のホスト名を設定します。ホスト名の最大文字数は 128 文字です。英数字および記号 (-) のみ使用できます。

④ポート番号

Azure IoT Hub のポート番号を設定します。ポート番号は 0 ～ 65535 の範囲で設定します。デフォルト値は Azure IoT Hub で使われる 8883 です。ポート番号に 0 を設定した場合、8883 のポート番号を使用します。

⑤キープアライブ

Azure IoT Hub と接続中、Azure IoT Hub に対して接続確認を実行する時間間隔を設定します。Plus CPU モジュールが送受信を行わない時間が設定した時間間隔以上続いた場合、Azure IoT Hub に接続確認を行います。設定できる範囲は 5 ～ 65535 秒です。デフォルト値は 60 秒です。

⑥デバイス ID

Azure IoT Hub へ登録した Plus CPU モジュールのデバイス ID を設定します。デバイス ID の最大文字数は 128 文字です。英数字および記号のみ使用できます。デバイス ID は、MQTT 通信のクライアント ID として使用します。

⑦アカウント名

Azure IoT Hub への接続に使用するアカウント名を設定します。アカウント名の最大文字数は 128 文字です。英数字および記号のみ使用できます。

⑧生成

このボタンをクリックすると、ホスト名 (③) とデバイス ID (⑥) から Azure IoT Hub への接続に使用するアカウント名を生成します。

⑨ルート証明書

[インポート] ボタンをクリックして、Azure IoT Hub と SSL/TLS 通信を行う場合に使用するルート証明書をインポートできます。インポートできるのは pem または crt ファイルです。WindLDR は pem および crt ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルはインポートできません。pem ファイルに複数のルート証明書が含まれている場合、先頭から 2 個のルート証明書を der ファイルに変換します。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートしたサーバー証明書のルート証明書の情報が表示されます。

[削除] ボタンをクリックすると、インポートした証明書が削除されます。

⑩クライアント証明書

[インポート] ボタンをクリックして、Azure IoT Hub と SSL/TLS 通信を行う場合に使用するクライアント証明書をインポートします。インポートできるのは pem または crt ファイルです。WindLDR は pem および crt ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルはインポートできません。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートしたクライアント証明書の情報が表示されます。

[削除] ボタンをクリックすると、インポートした証明書が削除されます。

⑪クライアント秘密鍵

[インポート] ボタンをクリックして、Azure IoT Hub と SSL/TLS 通信を行う場合に使用するクライアント秘密鍵をインポートします (暗号化方式 RSA、鍵長は 3072bit まで)。WindLDR は次の形式のファイルに対応しています。

- PKCS#1 形式の pem または der ファイル
- PKCS#8 形式 (暗号化なし) の pem または der ファイル

[削除] ボタンをクリックすると、インポートした秘密鍵が削除されます。

■ デバイス

⑫ 接続コントロール

Azure IoT Hub との接続および切断に関する操作を行うための内部リレーを設定します。設定した内部リレーを先頭に5ビット分の内部リレーを使用します。デバイスの範囲を超えないように先頭の内部リレーを設定してください。

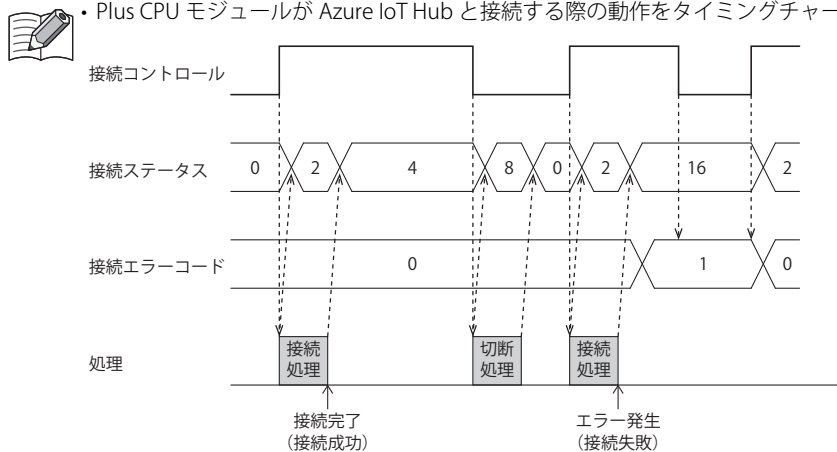
格納先	項目	内容
先頭番号+0	接続コントロール	Azure IoT Hub との接続および切断を行います。 ONすると、Azure IoT Hubへ接続します。OFFすると、Azure IoT Hubとの接続を切断します。
先頭番号+1	リザーブ	
先頭番号+2	リザーブ	
先頭番号+3	リザーブ	
先頭番号+4	リザーブ	

⑬ 接続ステータス

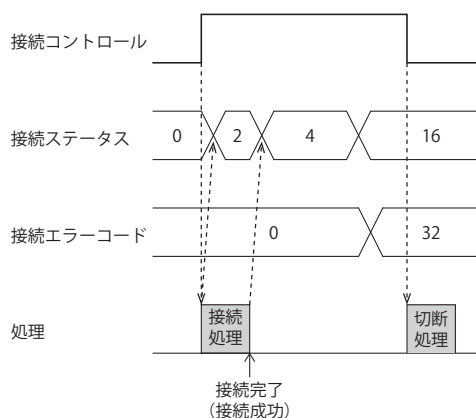
Azure IoT Hub に接続する際のステータスおよびエラー情報を格納するデータレジスタを設定します。設定したデータレジスタを先頭に5ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

格納先	項目	内容																																
先頭番号+0	接続ステータス	Azure IoT Hub に接続する際のステータスを格納します。																																
		<table border="1"> <thead> <tr> <th>ステータスコード</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>0 (0000h)</td> <td>初期状態 (切断状態)</td> </tr> <tr> <td>2 (0002h)</td> <td>接続処理中</td> </tr> <tr> <td>4 (0004h)</td> <td>接続状態</td> </tr> <tr> <td>8 (0008h)</td> <td>切断処理中</td> </tr> <tr> <td>16 (0010h)</td> <td>接続処理エラー</td> </tr> <tr> <td>32 (0020h)</td> <td>切断処理エラー</td> </tr> </tbody> </table>	ステータスコード	状態	0 (0000h)	初期状態 (切断状態)	2 (0002h)	接続処理中	4 (0004h)	接続状態	8 (0008h)	切断処理中	16 (0010h)	接続処理エラー	32 (0020h)	切断処理エラー																		
		ステータスコード	状態																															
		0 (0000h)	初期状態 (切断状態)																															
		2 (0002h)	接続処理中																															
		4 (0004h)	接続状態																															
		8 (0008h)	切断処理中																															
		16 (0010h)	接続処理エラー																															
32 (0020h)	切断処理エラー																																	
先頭番号+1	接続エラーコード	Azure IoT Hub に接続する際に発生したエラー情報を格納します。																																
		<table border="1"> <thead> <tr> <th>エラーコード</th> <th>エラー内容</th> </tr> </thead> <tbody> <tr> <td>1 (0001h)</td> <td>イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった</td> </tr> <tr> <td>2 (0002h)</td> <td>SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した</td> </tr> <tr> <td>4 (0004h)</td> <td>デバイスIDのフォーマットが不正である</td> </tr> <tr> <td>16 (0010h)</td> <td>未知のパケットを受信した</td> </tr> <tr> <td>32 (0020h)</td> <td>不正なMQTTパケットを受信した</td> </tr> <tr> <td>64 (0040h)</td> <td>キープアライブタイムアウトエラー</td> </tr> <tr> <td>80 (0050h)</td> <td>宛先ホストに到達できなかった</td> </tr> <tr> <td>96 (0060h)</td> <td>MQTTパケット受信タイムアウトエラー</td> </tr> <tr> <td>112 (0070h)</td> <td>TLSエラー</td> </tr> <tr> <td>256 (0100h)</td> <td>Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)</td> </tr> <tr> <td>512 (0200h)</td> <td>Azure IoT Hub 接続拒否 (デバイスIDが不正)</td> </tr> <tr> <td>768 (0300h)</td> <td>Azure IoT Hub 接続拒否 (Azure IoT Hub 使用不可)</td> </tr> <tr> <td>1024 (0400h)</td> <td>Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正)</td> </tr> <tr> <td>1280 (0500h)</td> <td>Azure IoT Hub 接続拒否 (権限なし)</td> </tr> <tr> <td>32768 (8000h)</td> <td>Azure IoT Hub 応答異常</td> </tr> </tbody> </table>	エラーコード	エラー内容	1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった	2 (0002h)	SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した	4 (0004h)	デバイスIDのフォーマットが不正である	16 (0010h)	未知のパケットを受信した	32 (0020h)	不正なMQTTパケットを受信した	64 (0040h)	キープアライブタイムアウトエラー	80 (0050h)	宛先ホストに到達できなかった	96 (0060h)	MQTTパケット受信タイムアウトエラー	112 (0070h)	TLSエラー	256 (0100h)	Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)	512 (0200h)	Azure IoT Hub 接続拒否 (デバイスIDが不正)	768 (0300h)	Azure IoT Hub 接続拒否 (Azure IoT Hub 使用不可)	1024 (0400h)	Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正)	1280 (0500h)	Azure IoT Hub 接続拒否 (権限なし)	32768 (8000h)	Azure IoT Hub 応答異常
		エラーコード	エラー内容																															
		1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった																															
		2 (0002h)	SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した																															
		4 (0004h)	デバイスIDのフォーマットが不正である																															
		16 (0010h)	未知のパケットを受信した																															
		32 (0020h)	不正なMQTTパケットを受信した																															
		64 (0040h)	キープアライブタイムアウトエラー																															
		80 (0050h)	宛先ホストに到達できなかった																															
		96 (0060h)	MQTTパケット受信タイムアウトエラー																															
		112 (0070h)	TLSエラー																															
		256 (0100h)	Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)																															
		512 (0200h)	Azure IoT Hub 接続拒否 (デバイスIDが不正)																															
768 (0300h)	Azure IoT Hub 接続拒否 (Azure IoT Hub 使用不可)																																	
1024 (0400h)	Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正)																																	
1280 (0500h)	Azure IoT Hub 接続拒否 (権限なし)																																	
32768 (8000h)	Azure IoT Hub 応答異常																																	
先頭番号+2	リザーブ																																	
先頭番号+3	リザーブ																																	
先頭番号+4	リザーブ																																	

- Plus CPU モジュールが Azure IoT Hub と接続する際の動作をタイミングチャートで示します。



- 接続エラーコード 16 (0010h) ~ 112 (0070h) は Azure IoT Hub と接続した状態 (接続ステータス 4 (0004h)) でも発生する場合があります。



- STOP 中は MQTT 通信を行いません。Azure IoT Hub との接続中に RUN から STOP した場合は接続コントロールを OFF して接続ステータスに切断状態 0 (0000h) を格納します。

- ⑭ [プロジェクトからインポート] ボタン
WindLDR のプロジェクトファイルから、MQTT 設定のみをインポートします。
- ⑮ [OK] ボタン
設定を保存し、[MQTT 設定] ダイアログボックスを閉じます。
- ⑯ [キャンセル] ボタン
設定を保存せずに、[MQTT 設定] ダイアログボックスを閉じます。

Azure IoT Hub へ DPS 経由で接続する

■ MQTT 接続方法

[MQTT 接続方法] グループで “DPS 経由で Azure IoT Hub に接続” を選択します。

■ MQTT 基本設定

[SD メモリカードで指定する] チェックボックスがオフの場合

The screenshot shows the 'MQTT 設定' (MQTT Settings) dialog box. The 'MQTT 有効にする' (Enable MQTT) checkbox is checked. The 'クラウドサービス名' (Cloud Service Name) is set to 'Azure IoT Hub'. Under 'MQTT 接続方法' (MQTT Connection Method), 'DPS 経由で Azure IoT Hub に接続' (Connect to Azure IoT Hub via DPS) is selected. In the 'MQTT 基本設定' (MQTT Basic Settings) section, the 'SD メモリカードで指定する' (Specify using SD Memory Card) checkbox is unchecked. The 'Azure IoT Hub' section includes fields for Host Name (3), Port Number (4, 8883), and Keep Alive (5, 60 seconds). The 'デバイス' (Device) section has radio buttons for Device ID (6), MAC Address, Fixed Value (7, 32937fc-554d-4368-aa93-f5ddd65b57a8), and Data Register. The 'デバイスプロビジョニングサービス(DPS)' (Device Provisioning Service) section includes fields for Service Endpoint (8), ID Scope (9), Symmetric Key (10), and Port Number (11, 8883). The '保護された接続(SSL/TLS)' (Secured Connection) section has a checkbox for Certificate (12) which is unchecked. The 'デバイス' (Device) section at the bottom includes Connection Control (13) and Connection Status (14). Buttons for 'プロジェクトからインポート' (15), 'OK' (16), and 'キャンセル' (17) are visible.

[SD メモリカードで指定する] チェックボックスがオンの場合

The screenshot shows the 'MQTT 設定' (MQTT Settings) dialog box. The 'MQTT 有効にする' (Enable MQTT) checkbox is checked. The 'クラウドサービス名' (Cloud Service Name) is set to 'Azure IoT Hub'. Under 'MQTT 接続方法' (MQTT Connection Method), 'DPS 経由で Azure IoT Hub に接続' (Connect to Azure IoT Hub via DPS) is selected. In the 'MQTT 基本設定' (MQTT Basic Settings) section, the 'SD メモリカードで指定する' (Specify using SD Memory Card) checkbox is checked, and a button labeled 'SD メモリカードを確認する' (2) is visible. The 'デバイス' (Device) section at the bottom is partially visible.

① SD メモリカードで指定する

MQTT 通信を行うための基本設定を、SD メモリカードを使用して Plus CPU モジュールに書き込むかどうかを指定します。

[SD メモリカードで指定する] チェックボックスをオンにすると、②が表示され、③～④が非表示になります。この場合、[SD メモリカード] ダイアログボックスで詳細を設定します。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

② SD メモリカードを設定する

[SD メモリカード] ダイアログボックスを開きます。[SD メモリカードで指定する] チェックボックスがオンの場合のみ表示されます。[SD メモリカード] ダイアログボックスについては、FC6A 形マイクロスマート ユーザーズ マニュアル「第 11 章 SD メモリカード」を参照してください。

③ ホスト名

DPS から取得した Azure IoT Hub のホスト名 (アスキーコード) を格納するデータレジスタを設定します。設定したデータレジスタを先頭に 64 ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

④ ポート番号

Azure IoT Hub のポート番号です。Device Provisioning Service (DPS) のポート番号 (①) と同じポート番号が使用されます。

⑤ キープアライブ

Azure IoT Hub と接続中、Azure IoT Hub に対して接続確認を実行する時間間隔を設定します。Plus CPU モジュールが送受信を行わない時間が設定した時間間隔以上続いた場合、Azure IoT Hub に接続確認を行います。設定できる範囲は 5 ～ 65535 秒です。デフォルト値は 60 秒です。

⑥ デバイス ID

DPS 経由で Azure IoT Hub へ登録するデバイス ID を設定します。デバイス ID は、次の 3 種類から設定でき、MQTT 通信のクライアント ID としても使用します。

デバイス ID	内容
MACアドレス	Plus CPUモジュールのEthernetポート1のMACアドレスを設定します。 ^{*1}
固定値	任意の文字列を設定します。 最大文字数は128文字です。英数字および記号のみ使用できます。
データレジスタ	デバイスIDを格納するためのデータレジスタを設定します。 設定したデータレジスタの上位バイトから順に値を読み出して、文字データとして扱い、デバイスIDとします。 ^{*2} 設定したデータレジスタを先頭に連続して64ワード分のデータレジスタを使用します。 ^{*3} デバイスの範囲を超えないように先頭のデータレジスタを設定してください。デバイスIDには英数字および記号のみ使用できます。

*1 Plus CPU モジュールの Ethernet ポート 1 の MAC アドレスが 12-34-56-78-9A-BC (D8324=0012h、D8325=0034h、D8326=0056h、D8327=0078h、D8328=009Ah、D8329=00BCh) の場合、デバイス ID は "123456789ABC" です。

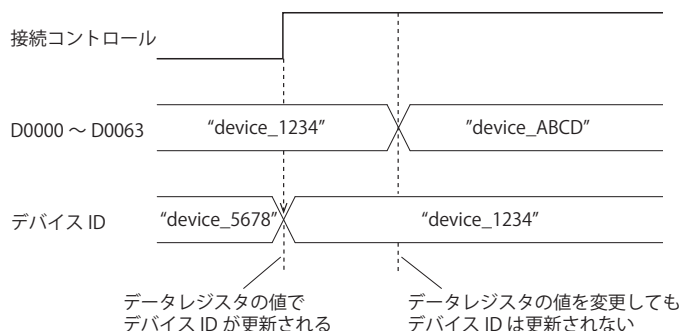
*2 例えば、D0000 を指定し、各データレジスタに次の値が格納されている場合、デバイス ID は "device_1234" です。

データレジスタ	格納値	
	上位バイト	下位バイト
D0000	"d"=64h	"e"=65h
D0001	"v"=76h	"i"=69h
D0002	"c"=63h	"e"=65h
D0003	"_"=5Fh	"1"=31h
D0004	"2"=32h	"3"=33h
D0005	"4"=34h	00h

*3 設定する文字列が 128 バイト (64 ワード) より短い場合は、文字列の終わりとして終端文字 NULL (00h) を追加してください。設定したデータレジスタの上位バイトから終端文字 NULL (00h) の前までを文字データとして扱い、デバイス ID とします。



- デバイス ID を変更した場合、再度 DPS へ接続して Plus CPU モジュールを登録してください。
- デバイス ID をデータレジスタで設定する場合、MOVC (ムーブキャラクタ) 命令を使用して、データレジスタに文字列を格納することができます。MOVC 命令については、ラダープログラミング マニュアル「第 5 章 MOVC (ムーブキャラクタ)」を参照してください。
- 接続コントロール (⑬) を OFF から ON したとき、Plus CPU モジュール内部で記憶するデバイス ID が更新されます。例えば、デバイス ID をデータレジスタ (D0000) で設定する場合、デバイス ID は次のタイミングで更新されます。



⑦ランダム ID を生成

デバイス ID (⑥) を固定値で設定する場合、このボタンをクリックすると無作為にデバイス ID を生成します。

⑧サービスエンドポイント

DPS のサービスエンドポイントを設定します。サービスエンドポイントの最大文字数は 81 文字です。英数字および記号のみ使用できます。

⑨ ID スコープ

DPS の ID スコープを設定します。ID スコープの最大文字数は 11 文字です。英数字のみ使用できます。

⑩対称キー

Plus CPU モジュールの登録先となる DPS の登録グループの対称キー（主キーまたはセカンダリキー）を設定します。対称キーの最大文字数は 88 文字です。英数字および記号のみ使用できます。

⑪ポート番号

Device Provisioning Service (DPS) のポート番号を設定します。ポート番号は 0 ～ 65535 の範囲で設定します。デフォルト値は DPS で使われる 8883 です。ポート番号に 0 を設定した場合、8883 のポート番号を使用します。

⑫ルート証明書

[インポート] ボタンをクリックして、DPS および Azure IoT Hub と SSL/TLS 通信を行う場合に使用するルート証明書をインポートできます。インポートできるのは pem または crt ファイルです。WindLDR は pem および crt ファイルを der ファイルに変換します。2K バイトより大きいサイズの der ファイルはインポートできません。pem ファイルに複数のルート証明書が含まれている場合、先頭から 2 個のルート証明書を der ファイルに変換します。

[詳細] ボタンをクリックすると、[証明書情報] ダイアログボックスが表示され、インポートしたサーバー証明書のルート証明書の情報が表示されます。

[削除] ボタンをクリックすると、インポートした証明書が削除されます。

■ デバイス

③ 接続コントロール

DPS および Azure IoT Hub との接続および切断に関する内部リレーを設定します。設定した内部リレーを先頭に 5 ビット分の内部リレーを使用します。デバイスの範囲を超えないように先頭の内部リレーを設定してください。

格納先	項目	内容
先頭番号+0	接続コントロール ^{*1*}	DPSおよびAzure IoT Hubとの接続および切断を行います。 接続コントロールをONすると、DPS接続許可（先頭番号+1）の状態に応じて、DPSおよびAzure IoT Hubへ接続します。 接続コントロールをOFFすると、DPSおよびAzure IoT Hubとの接続を切断します。
先頭番号+1	DPS接続許可	DPSへの接続を許可します。 DPS接続許可がONのとき、接続コントロール（先頭番号+0）をONすると、DPS経由でAzure IoT Hubへ接続します。 ^{*2} DPS接続許可がOFFのとき、接続コントロール（先頭番号+0）をONすると、DPSを経由せずに、ホスト名（③）に格納されたAzure IoT Hubへ直接接続します ^{*4} 。
先頭番号+2	リザーブ	
先頭番号+3	リザーブ	
先頭番号+4	リザーブ	

*1 DPS および Azure IoT Hub と接続するために SAS トークンを生成するとき、Plus CPU モジュールの時計から算出した現在時刻 (UTC) を使用します。Plus CPU モジュールの時計が実際の現在時刻から大きくずれている場合、DPS および Azure IoT Hub と接続できない場合があります。Plus CPU モジュールの時計および時計に関する機能を設定してください。

- SNTP サーバーから現在時刻 (UTC) を取得する場合

次の項目を参照し、Plus CPU モジュールの時計および時計に関する機能を設定してください。

設定項目	内容	参照先
SNTP設定	SNTPサーバーから現在時刻 (UTC) を取得して、Plus CPUモジュールの時計を調整します。	「第3章 SNTP設定」(3-11頁)
タイムゾーン	標準時に対する地域の時間帯ごとの差を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 タイムゾーン」
サマータイム	サマータイムの期間を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 サマータイム」



Plus CPU モジュールの時計は時間の経過とともに、精度に応じたずれが生じます。最後に SNTP サーバーから現在時刻 (UTC) を取得してから 6 か月以上が経過すると、DPS および Azure IoT Hub と接続できなくなる場合があります。適切な頻度で SNTP サーバーから現在時刻 (UTC) を取得してください。

- 直接、Plus CPU モジュールの時計を設定する場合

次の項目を参照し、Plus CPU モジュールの時計および時計に関する機能を設定してください。

設定項目	内容	参照先
時計機能	WindLDRまたは特殊データレジスタおよび特殊内部リレーを使って、Plus CPUモジュールの時計を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 時計機能」
タイムゾーン	標準時に対する地域の時間帯ごとの差を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 タイムゾーン」
サマータイム	サマータイムの期間を設定します。	FC6A形マイクロスマート ユーザーズ マニュアル「第5章 サマータイム」



Plus CPU モジュールの時計は時間の経過とともに、精度に応じたずれが生じます。最後に時計を設定してから 6 か月以上が経過すると、DPS および Azure IoT Hub と接続できなくなる場合があります。適切な頻度で時計を設定してください。

- *2 Plus CPU モジュールが生成する SAS トークンは、生成してから 5 分後に有効期限が切れます。Azure IoT Hub に接続した状態で有効期限が切れると、Azure IoT Hub 応答異常（「④接続ステータス」(17-30 頁)の接続エラーコード参照）が発生し、Azure IoT Hub と切断します。引き続き接続する場合は、再度、接続処理を行ってください。
- *3 このとき、DPS から取得した Azure IoT Hub のホスト名が、ホスト名（③）で設定したデータレジスタに格納されます。
- *4 デバイス ID を変更した場合、再度 DPS へ接続してデバイスの登録を行ってください。

⑭接続ステータス

DPS および Azure IoT Hub に接続する際のステータスおよびエラー情報を格納するデータレジスタを設定します。設定したデータレジスタを先頭に5ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

格納先	項目	内容																																												
先頭番号+0	接続ステータス	DPSおよびAzure IoT Hubに接続する際のステータスを格納します。																																												
		<table border="1"> <thead> <tr> <th>ステータスコード</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>0 (0000h)</td> <td>初期状態 (切断状態)</td> </tr> <tr> <td>2 (0002h)</td> <td>Azure IoT Hub 接続処理中</td> </tr> <tr> <td>4 (0004h)</td> <td>Azure IoT Hub 接続状態</td> </tr> <tr> <td>8 (0008h)</td> <td>Azure IoT Hub 切断処理中</td> </tr> <tr> <td>16 (0010h)</td> <td>Azure IoT Hub 接続処理エラー</td> </tr> <tr> <td>32 (0020h)</td> <td>Azure IoT Hub 切断処理エラー</td> </tr> <tr> <td>512 (0200h)</td> <td>DPS 接続処理中</td> </tr> <tr> <td>1024 (0400h)</td> <td>DPS 接続状態</td> </tr> <tr> <td>2048 (0800h)</td> <td>DPS 切断処理中</td> </tr> <tr> <td>4096 (1000h)</td> <td>DPS 接続処理エラー</td> </tr> <tr> <td>8192 (2000h)</td> <td>DPS 切断処理エラー</td> </tr> </tbody> </table>	ステータスコード	状態	0 (0000h)	初期状態 (切断状態)	2 (0002h)	Azure IoT Hub 接続処理中	4 (0004h)	Azure IoT Hub 接続状態	8 (0008h)	Azure IoT Hub 切断処理中	16 (0010h)	Azure IoT Hub 接続処理エラー	32 (0020h)	Azure IoT Hub 切断処理エラー	512 (0200h)	DPS 接続処理中	1024 (0400h)	DPS 接続状態	2048 (0800h)	DPS 切断処理中	4096 (1000h)	DPS 接続処理エラー	8192 (2000h)	DPS 切断処理エラー																				
		ステータスコード	状態																																											
		0 (0000h)	初期状態 (切断状態)																																											
		2 (0002h)	Azure IoT Hub 接続処理中																																											
		4 (0004h)	Azure IoT Hub 接続状態																																											
		8 (0008h)	Azure IoT Hub 切断処理中																																											
		16 (0010h)	Azure IoT Hub 接続処理エラー																																											
		32 (0020h)	Azure IoT Hub 切断処理エラー																																											
		512 (0200h)	DPS 接続処理中																																											
		1024 (0400h)	DPS 接続状態																																											
		2048 (0800h)	DPS 切断処理中																																											
		4096 (1000h)	DPS 接続処理エラー																																											
8192 (2000h)	DPS 切断処理エラー																																													
先頭番号+1	接続エラーコード	DPSおよびAzure IoT Hubに接続する際に発生したエラー情報を格納します。																																												
		<table border="1"> <thead> <tr> <th>エラーコード</th> <th>エラー内容</th> </tr> </thead> <tbody> <tr> <td>1 (0001h)</td> <td>イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった</td> </tr> <tr> <td>2 (0002h)</td> <td>SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した</td> </tr> <tr> <td>4 (0004h)</td> <td>デバイスIDのフォーマットが不正である</td> </tr> <tr> <td>8 (0008h)</td> <td>データレジスタに格納されたAzure IoT Hubのホスト名が不正である</td> </tr> <tr> <td>16 (0010h)</td> <td>未知のパケットを受信した</td> </tr> <tr> <td>32 (0020h)</td> <td>不正なMQTTパケットを受信した</td> </tr> <tr> <td>64 (0040h)</td> <td>キーブアライブタイムアウトエラー</td> </tr> <tr> <td>80 (0050h)</td> <td>宛先ホストに到達できなかった*1</td> </tr> <tr> <td>96 (0060h)</td> <td>MQTTパケット受信タイムアウトエラー</td> </tr> <tr> <td>112 (0070h)</td> <td>TLSエラー</td> </tr> <tr> <td>256 (0100h)</td> <td>Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)</td> </tr> <tr> <td>512 (0200h)</td> <td>Azure IoT Hub 接続拒否 (デバイスIDが不正)</td> </tr> <tr> <td>768 (0300h)</td> <td>Azure IoT Hub 接続拒否 (ブローカー使用不可)</td> </tr> <tr> <td>1024 (0400h)</td> <td>Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正)</td> </tr> <tr> <td>1280 (0500h)</td> <td>Azure IoT Hub 接続拒否 (権限なし) *2*3*4</td> </tr> <tr> <td>4352 (1100h)</td> <td>DPS 接続拒否 (MQTTプロトコルバージョン許容不可)</td> </tr> <tr> <td>4608 (1200h)</td> <td>DPS 接続拒否 (デバイスIDが不正)</td> </tr> <tr> <td>4864 (1300h)</td> <td>DPS 接続拒否 (DPS使用不可)</td> </tr> <tr> <td>5120 (1400h)</td> <td>DPS 接続拒否 (アカウント名またはパスワードが不正)</td> </tr> <tr> <td>5376 (1500h)</td> <td>DPS 接続拒否 (権限なし) *2*3*4</td> </tr> <tr> <td>32768 (8000h)</td> <td>Azure IoT Hub / DPS応答異常</td> </tr> </tbody> </table>	エラーコード	エラー内容	1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった	2 (0002h)	SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した	4 (0004h)	デバイスIDのフォーマットが不正である	8 (0008h)	データレジスタに格納されたAzure IoT Hubのホスト名が不正である	16 (0010h)	未知のパケットを受信した	32 (0020h)	不正なMQTTパケットを受信した	64 (0040h)	キーブアライブタイムアウトエラー	80 (0050h)	宛先ホストに到達できなかった*1	96 (0060h)	MQTTパケット受信タイムアウトエラー	112 (0070h)	TLSエラー	256 (0100h)	Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)	512 (0200h)	Azure IoT Hub 接続拒否 (デバイスIDが不正)	768 (0300h)	Azure IoT Hub 接続拒否 (ブローカー使用不可)	1024 (0400h)	Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正)	1280 (0500h)	Azure IoT Hub 接続拒否 (権限なし) *2*3*4	4352 (1100h)	DPS 接続拒否 (MQTTプロトコルバージョン許容不可)	4608 (1200h)	DPS 接続拒否 (デバイスIDが不正)	4864 (1300h)	DPS 接続拒否 (DPS使用不可)	5120 (1400h)	DPS 接続拒否 (アカウント名またはパスワードが不正)	5376 (1500h)	DPS 接続拒否 (権限なし) *2*3*4	32768 (8000h)	Azure IoT Hub / DPS応答異常
		エラーコード	エラー内容																																											
		1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった																																											
		2 (0002h)	SDメモリカードから認証情報をダウンロードしていない、またはダウンロードした認証情報の読み出しに失敗した																																											
		4 (0004h)	デバイスIDのフォーマットが不正である																																											
		8 (0008h)	データレジスタに格納されたAzure IoT Hubのホスト名が不正である																																											
		16 (0010h)	未知のパケットを受信した																																											
		32 (0020h)	不正なMQTTパケットを受信した																																											
		64 (0040h)	キーブアライブタイムアウトエラー																																											
		80 (0050h)	宛先ホストに到達できなかった*1																																											
		96 (0060h)	MQTTパケット受信タイムアウトエラー																																											
		112 (0070h)	TLSエラー																																											
		256 (0100h)	Azure IoT Hub 接続拒否 (MQTTプロトコルバージョン許容不可)																																											
		512 (0200h)	Azure IoT Hub 接続拒否 (デバイスIDが不正)																																											
		768 (0300h)	Azure IoT Hub 接続拒否 (ブローカー使用不可)																																											
		1024 (0400h)	Azure IoT Hub 接続拒否 (アカウント名またはパスワードが不正)																																											
		1280 (0500h)	Azure IoT Hub 接続拒否 (権限なし) *2*3*4																																											
		4352 (1100h)	DPS 接続拒否 (MQTTプロトコルバージョン許容不可)																																											
		4608 (1200h)	DPS 接続拒否 (デバイスIDが不正)																																											
		4864 (1300h)	DPS 接続拒否 (DPS使用不可)																																											
		5120 (1400h)	DPS 接続拒否 (アカウント名またはパスワードが不正)																																											
		5376 (1500h)	DPS 接続拒否 (権限なし) *2*3*4																																											
32768 (8000h)	Azure IoT Hub / DPS応答異常																																													
先頭番号+2	リザーブ																																													
先頭番号+3	リザーブ																																													
先頭番号+4	リザーブ																																													

*1 DPS のサービスエンドポイントまたは Azure IoT Hub のホスト名を確認してください。

*2 サービスエンドポイント (Ⓢ)、ID スコープ (Ⓣ) および対称キー (Ⓤ) を確認してください。

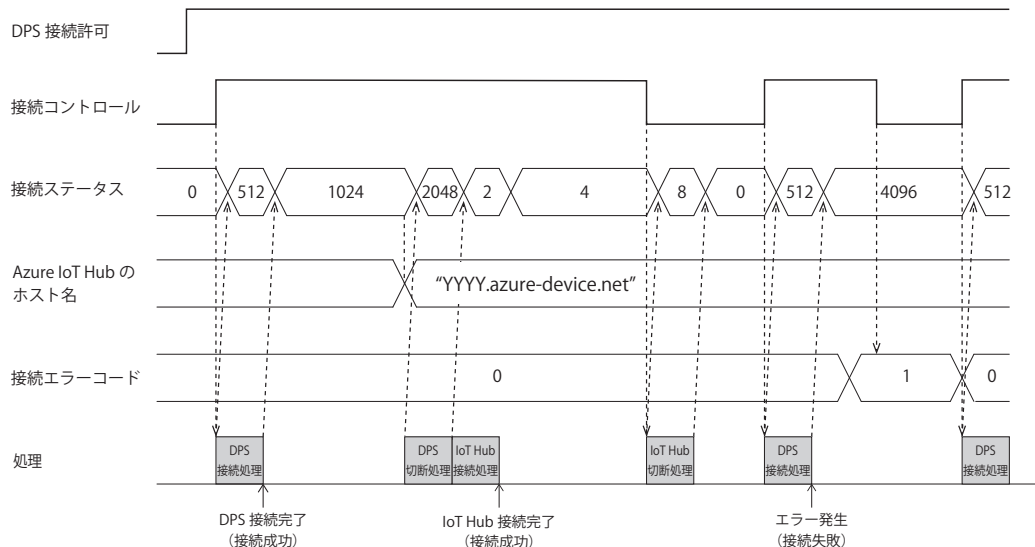
*3 Plus CPU モジュールの時計および時計に関する機能を設定してください。

*4 DPS および Azure IoT Hub に Plus CPU モジュールが登録されているか確認してください。必要に応じて、DPS への接続を再度行っ

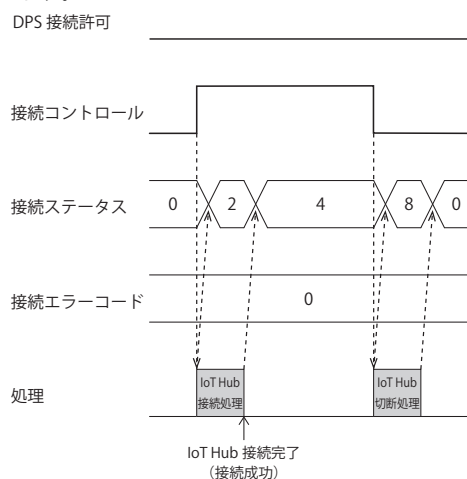
てください。



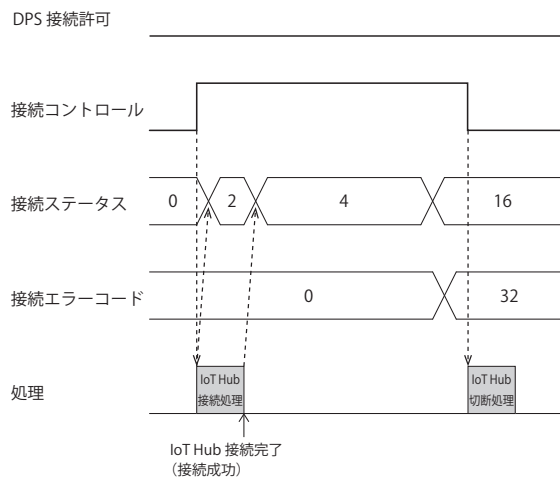
- DPS 接続許可が ON のとき、Plus CPU モジュールが DPS および Azure IoT Hub と接続する際の動作をタイミングチャートで示します。



- DPS 接続許可が OFF のとき、Plus CPU モジュールが Azure IoT Hub と直接接続する際の動作をタイミングチャートで示します。



- 接続エラーコード 16 (0010h) ~ 112 (0070h) は Azure IoT Hub と接続した状態 (接続ステータス 4 (0004h)) でも発生する場合があります。



- STOP 中は MQTT 通信を行いません。Azure IoT Hub との接続中に RUN から STOP した場合は接続コントロールを OFF して接続ステータスに切断状態 0 (0000h) を格納します。

- ⑮ [プロジェクトからインポート] ボタン
WindLDRのプロジェクトファイルから、MQTT 設定のみをインポートします。
- ⑯ [OK] ボタン
設定を保存し、[MQTT 設定] ダイアログボックスを閉じます。

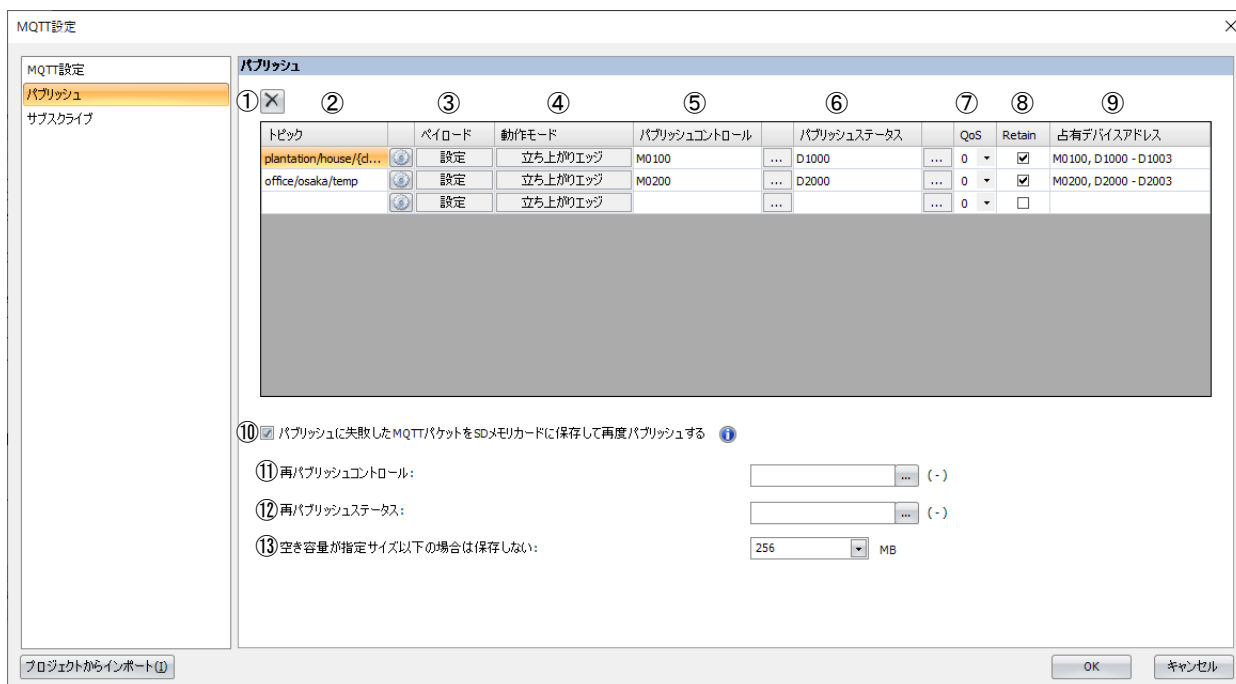


デバイス ID (⑥) をデータレジスタに設定している場合、[OK] ボタンを押すと、デバイス ID を 1 バイトとしてトピックのサイズを計算します。設定したトピックが 256 バイトを超える場合、エラーとなります。

- ⑰ [キャンセル] ボタン
設定を保存せずに、[MQTT 設定] ダイアログボックスを閉じます。

パブリッシュ

[MQTT 設定] ダイアログボックスで [パブリッシュ] を選択すると、[パブリッシュ] タブが表示されます。このタブでパブリッシュするための設定を行います。



① ボタン

選択した行の設定内容を削除します。

② トピック

トピックを設定します。トピックは UTF-8 で最大 256 バイトまで設定できます。

ボタンをクリックすると、[トピック] ダイアログボックスが表示されます。詳細は、「[トピック] ダイアログボックス」(17-40 頁) を参照してください。



- トピックは最大 10 個まで設定できます。
- パブリッシュコントロール (⑤) が OFF → ON したとき、[トピック] ダイアログボックスで設定したトピックが生成されます。生成したトピックが不正な場合、そのトピックにパブリッシュできません。エラーについては、パブリッシュステータス (⑥) を参照してください。
- +, #, / を含むクライアント ID はトピックに使用できません。

③ ペイロード

ペイロード内のデータを設定します。[設定] ボタンをクリックすると、[ペイロード] ダイアログボックスが表示されます。詳細は、「[ペイロード] ダイアログボックス」(17-42 頁) を参照してください。

④ 動作モード

動作モードを設定します。動作モードをクリックすると、[動作モード設定] ダイアログボックスが表示されます。詳細は、「[動作モード設定] ダイアログボックス」(17-37 頁) を参照してください。

⑤ パブリッシュコントロール

トピックにパブリッシュするための内部リレーを設定します。設定した内部リレーを ON すると、動作モード (④) に従ってトピックにパブリッシュします。

⑥パブリッシュステータス

パブリッシュする際のステータス等を格納するデータレジスタを設定します。設定したデータレジスタを先頭に4ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。

格納先	項目	内容																
先頭番号+0	パブリッシュステータス	パブリッシュする際のステータスを格納します。																
		<table border="1"> <thead> <tr> <th>ステータスコード</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>0 (0000h)</td> <td>初期状態</td> </tr> <tr> <td>2 (0002h)</td> <td>パブリッシュ処理中</td> </tr> <tr> <td>4 (0004h)</td> <td>パブリッシュ完了状態</td> </tr> <tr> <td>16 (0010h)</td> <td>パブリッシュエラー</td> </tr> </tbody> </table>	ステータスコード	状態	0 (0000h)	初期状態	2 (0002h)	パブリッシュ処理中	4 (0004h)	パブリッシュ完了状態	16 (0010h)	パブリッシュエラー						
		ステータスコード	状態															
		0 (0000h)	初期状態															
		2 (0002h)	パブリッシュ処理中															
4 (0004h)	パブリッシュ完了状態																	
16 (0010h)	パブリッシュエラー																	
先頭番号+1	パブリッシュエラーコード	パブリッシュした際に発生したエラー情報を格納します。																
		<table border="1"> <thead> <tr> <th>エラーコード*1</th> <th>エラー内容</th> </tr> </thead> <tbody> <tr> <td>1 (0001h)</td> <td>イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった</td> </tr> <tr> <td>2 (0002h)</td> <td>接続ステータスが0 (0000h) (切断状態) のときにパブリッシュした</td> </tr> <tr> <td>4 (0004h)</td> <td>不正なフォーマットのトピック、または256バイトより大きいサイズのトピックへパブリッシュしようとした</td> </tr> <tr> <td>5 (0005h)</td> <td>32768バイトより大きいサイズのJSON形式のデータをパブリッシュしようとした</td> </tr> <tr> <td>7 (0007h)</td> <td>データタイプがフロートの場合に±∞ (±無限大) または非数をパブリッシュした、またはデータタイプが文字列の場合にUTF-8以外の文字コードをパブリッシュした</td> </tr> <tr> <td>96 (0060h)</td> <td>MQTTパケット受信タイムアウトエラー</td> </tr> <tr> <td>32768 (8000h)</td> <td>ブローカー応答異常</td> </tr> </tbody> </table>	エラーコード*1	エラー内容	1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった	2 (0002h)	接続ステータスが0 (0000h) (切断状態) のときにパブリッシュした	4 (0004h)	不正なフォーマットのトピック、または256バイトより大きいサイズのトピックへパブリッシュしようとした	5 (0005h)	32768バイトより大きいサイズのJSON形式のデータをパブリッシュしようとした	7 (0007h)	データタイプがフロートの場合に±∞ (±無限大) または非数をパブリッシュした、またはデータタイプが文字列の場合にUTF-8以外の文字コードをパブリッシュした	96 (0060h)	MQTTパケット受信タイムアウトエラー	32768 (8000h)	ブローカー応答異常
		エラーコード*1	エラー内容															
		1 (0001h)	イーサネットケーブルが抜けていたり断線したりしていて、Plus CPUモジュールがネットワークに正しく接続できていなかった															
		2 (0002h)	接続ステータスが0 (0000h) (切断状態) のときにパブリッシュした															
		4 (0004h)	不正なフォーマットのトピック、または256バイトより大きいサイズのトピックへパブリッシュしようとした															
		5 (0005h)	32768バイトより大きいサイズのJSON形式のデータをパブリッシュしようとした															
		7 (0007h)	データタイプがフロートの場合に±∞ (±無限大) または非数をパブリッシュした、またはデータタイプが文字列の場合にUTF-8以外の文字コードをパブリッシュした															
96 (0060h)	MQTTパケット受信タイムアウトエラー																	
32768 (8000h)	ブローカー応答異常																	
先頭番号+2	送信データバイト数	パブリッシュステータスが4 (0004h)になったとき、送信したデータのサイズをバイト単位で格納します。																
先頭番号+3	パブリッシュエラーID	パブリッシュエラーコードが7 (0007h)になったとき、エラーが発生したID*2を格納します。																

*1 パブリッシュエラーコードが4 (0004h)～7 (0007h) 以外のときは接続コントロールもOFFとなります。パブリッシュエラーコードが4 (0004h)～7 (0007h) のとき、パブリッシュコントロールおよび接続コントロールの状態は保持されます。

*2 IDとは、[ペイロード] ダイアログボックスで各要素を識別するために割り付けられた番号です。詳細は、「[ペイロード] ダイアログボックス」(17-42 頁) を参照してください。

⑦ QoS

QoS を0～2の範囲で設定します。デフォルト値は0です。

⑧ Retain

Retainの有効/無効を設定します。チェックボックスがオンの場合、Retainは有効です。デフォルト値はオフです。

⑨占有デバイスアドレス

パブリッシュコントロールおよびパブリッシュステータスを設定することで占有されるデバイスアドレスが表示されます。

⑩パブリッシュに失敗したMQTTパケットをSDメモリカードに保存して再度パブリッシュする

パブリッシュに失敗したMQTTパケットをSDメモリカードに保存して、再度パブリッシュするかどうかを設定します。このチェックボックスをオンにすると、再パブリッシュコントロール (⑩) の状態に応じて、再度パブリッシュします。



- パブリッシュに失敗したMQTTパケットは、再パブリッシュデータとしてSDメモリカードに保存されます。再パブリッシュデータをパブリッシュすることを再パブリッシュと呼びます。
- 再パブリッシュする場合、QoS (⑦) を1または2に設定してください。
QoS (⑦) を0に設定した場合、MQTTブローカーからの応答がない場合でもエラーにならないため、このMQTTパケットは再パブリッシュデータとしてSDメモリカードに保存されません。
- パブリッシュ処理およびサブスクリプト処理「サブスクリプト」(17-38 頁) は、再パブリッシュ処理より優先されません。
- クライアントIDを変更した場合、古いクライアントIDで保存した再パブリッシュデータは削除されます。

①再パブリッシュコントロール

再パブリッシュするための内部リレーを設定します。設定した内部リレーを先頭に5ビット分の内部リレーを使用します。デバイスの範囲を超えないように先頭の内部リレーを設定してください。

格納先	項目	内容
先頭番号+0	再パブリッシュ許可	再パブリッシュを許可します。 再パブリッシュ許可がONのとき、再パブリッシュデータがSDメモ리카ードに保存されていれば、再パブリッシュします。 ^{*1*2*3} 再パブリッシュ許可がOFFのとき、再パブリッシュデータがSDメモ리카ードに保存されていても再パブリッシュしません。
先頭番号+1	再パブリッシュデータ保存許可	SDメモ리카ードへの再パブリッシュデータの保存を許可します。 再パブリッシュデータ保存許可がONのとき、パブリッシュに失敗した ^(*) MQTTパケットを、再パブリッシュデータとしてSDメモ리카ードに保存します。 ^{*5} 再パブリッシュデータ保存許可がOFFのとき、パブリッシュに失敗してもMQTTパケットをSDメモ리카ードに保存しません。
先頭番号+2	リザーブ	
先頭番号+3	リザーブ	
先頭番号+4	リザーブ	

*1 SDメモ리카ードに保存された再パブリッシュデータを順不同で再パブリッシュします。必要に応じてペイロードにタイムスタンプを設定してください。

*2 再パブリッシュに成功してから再パブリッシュデータを削除するまでの間に、SDメモ리카ードを取り外したり、Plus CPUモジュールの電源をOFFした場合、再パブリッシュデータがSDメモ리카ードから削除されないことがあります。この場合、次にMQTTブローカーと接続したときに、削除できなかった再パブリッシュデータを再パブリッシュします。

*3 再パブリッシュ処理が完了してから200ms以上経過した後に、次の再パブリッシュ処理を行います。再パブリッシュ処理にかかる時間、および再パブリッシュ処理が完了してから次の再パブリッシュ処理までの時間は、次のような場合に長くなります。

- SDメモ리카ードへアクセスする他の機能（レシピ機能、FTPサーバー/クライアント機能など）や命令（DLOG、TRACE命令など）を使用する
- パブリッシュ処理またはサブスクリプション処理を行う

*4 パブリッシュエラーコード（Ⓒ）が4（0004h）～7（0007h）の場合、トピックまたはペイロードが不正のためMQTTパケットを生成できません。この場合、再パブリッシュデータはSDメモ리카ードに保存されません。

*5 パブリッシュに失敗してから再パブリッシュデータが保存されるまでの間に、SDメモ리카ードを取り外したり、Plus CPUモジュールの電源をOFFした場合、再パブリッシュデータがSDメモ리카ードに保存されないことがあります。



SDメモ리카ードを取り付ける、または取り外す場合は、再パブリッシュ許可および再パブリッシュデータ保存許可をOFFしてください。

⑫再パブリッシュステータス

再パブリッシュデータを保存する際、または再パブリッシュする際のステータス等を格納するデータレジスタを設定します。設定したデータレジスタを先頭に5ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを設定してください。

格納先	項目	内容																		
先頭番号+0	再パブリッシュステータス	再パブリッシュデータを保存する際、または再パブリッシュする際のステータスを格納します。																		
		<table border="1"> <thead> <tr> <th>ステータスコード</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>0 (0000h)</td> <td>初期状態</td> </tr> <tr> <td>1 (0001h)</td> <td>再パブリッシュデータなし</td> </tr> <tr> <td>2 (0002h)</td> <td>再パブリッシュ処理中</td> </tr> <tr> <td>4 (0004h)</td> <td>再パブリッシュ完了状態</td> </tr> <tr> <td>16 (0010h)</td> <td>再パブリッシュエラー*1</td> </tr> <tr> <td>512 (0200h)</td> <td>再パブリッシュデータ保存中</td> </tr> <tr> <td>1024 (0400h)</td> <td>再パブリッシュデータ保存完了</td> </tr> <tr> <td>2048 (0800h)</td> <td>再パブリッシュデータ保存エラー*1</td> </tr> </tbody> </table>	ステータスコード	状態	0 (0000h)	初期状態	1 (0001h)	再パブリッシュデータなし	2 (0002h)	再パブリッシュ処理中	4 (0004h)	再パブリッシュ完了状態	16 (0010h)	再パブリッシュエラー*1	512 (0200h)	再パブリッシュデータ保存中	1024 (0400h)	再パブリッシュデータ保存完了	2048 (0800h)	再パブリッシュデータ保存エラー*1
		ステータスコード	状態																	
		0 (0000h)	初期状態																	
		1 (0001h)	再パブリッシュデータなし																	
		2 (0002h)	再パブリッシュ処理中																	
		4 (0004h)	再パブリッシュ完了状態																	
		16 (0010h)	再パブリッシュエラー*1																	
		512 (0200h)	再パブリッシュデータ保存中																	
		1024 (0400h)	再パブリッシュデータ保存完了																	
2048 (0800h)	再パブリッシュデータ保存エラー*1																			
先頭番号+1	再パブリッシュエラーコード	再パブリッシュデータを保存する際、または再パブリッシュする際に発生したエラー情報を格納します。*2																		
		<table border="1"> <thead> <tr> <th>エラーコード</th> <th>エラー内容</th> </tr> </thead> <tbody> <tr> <td>3 (0003h)</td> <td>クライアントIDを記憶していないため、再パブリッシュデータを保存できない*3*6</td> </tr> <tr> <td>8 (0008h)</td> <td>トピックまたはペイロードが不正であるため、再パブリッシュデータを生成できない*3</td> </tr> <tr> <td>16 (0010h)</td> <td>SDメモ리카ードが挿入されていない*3</td> </tr> <tr> <td>17 (0011h)</td> <td>空き容量が指定サイズ以上の場合に保存する(ⓐ)で設定したサイズよりも、SDメモ리카ードの空き容量が小さい*3</td> </tr> <tr> <td>18 (0012h)</td> <td>SDメモ리카ードの読み書きエラーのため、再パブリッシュデータを保存できない*3</td> </tr> <tr> <td>19 (0013h)</td> <td>SDメモ리카ードの読み書きエラーのため、再パブリッシュできない*4</td> </tr> <tr> <td>96 (0060h)</td> <td>MQTTパケット受信タイムアウトエラー*4*5</td> </tr> <tr> <td>32768 (8000h)</td> <td>ブローカー応答異常*4*5</td> </tr> </tbody> </table>	エラーコード	エラー内容	3 (0003h)	クライアントIDを記憶していないため、再パブリッシュデータを保存できない*3*6	8 (0008h)	トピックまたはペイロードが不正であるため、再パブリッシュデータを生成できない*3	16 (0010h)	SDメモ리카ードが挿入されていない*3	17 (0011h)	空き容量が指定サイズ以上の場合に保存する(ⓐ)で設定したサイズよりも、SDメモ리카ードの空き容量が小さい*3	18 (0012h)	SDメモ리카ードの読み書きエラーのため、再パブリッシュデータを保存できない*3	19 (0013h)	SDメモ리카ードの読み書きエラーのため、再パブリッシュできない*4	96 (0060h)	MQTTパケット受信タイムアウトエラー*4*5	32768 (8000h)	ブローカー応答異常*4*5
		エラーコード	エラー内容																	
		3 (0003h)	クライアントIDを記憶していないため、再パブリッシュデータを保存できない*3*6																	
		8 (0008h)	トピックまたはペイロードが不正であるため、再パブリッシュデータを生成できない*3																	
		16 (0010h)	SDメモ리카ードが挿入されていない*3																	
		17 (0011h)	空き容量が指定サイズ以上の場合に保存する(ⓐ)で設定したサイズよりも、SDメモ리카ードの空き容量が小さい*3																	
		18 (0012h)	SDメモ리카ードの読み書きエラーのため、再パブリッシュデータを保存できない*3																	
		19 (0013h)	SDメモ리카ードの読み書きエラーのため、再パブリッシュできない*4																	
96 (0060h)	MQTTパケット受信タイムアウトエラー*4*5																			
32768 (8000h)	ブローカー応答異常*4*5																			
先頭番号+2	リザーブ																			
先頭番号+3	リザーブ																			
先頭番号+4	リザーブ																			

- *1 次に再パブリッシュする、または再パブリッシュデータの保存を行うときに、再パブリッシュステータスが更新されます。
- *2 次に再パブリッシュに失敗した、または再パブリッシュデータの保存に失敗したときに、再パブリッシュエラーコードが更新されます。
- *3 再パブリッシュデータをSDメモ리카ードに保存する際に発生するエラーです。この場合、再パブリッシュステータスは2048(0800h)です。
- *4 再パブリッシュする際に発生するエラーです。この場合、再パブリッシュステータスは16(0010h)です。
- *5 再パブリッシュエラーコードが96(0060h)～32768(8000h)のとき、接続コントロールはOFFします。
- *6 接続コントロール(先頭番号+0)をOFFからONして、クライアントIDを記憶してください。



再パブリッシュに失敗した再パブリッシュデータはSDメモ리카ードから削除されません。

⑬空き容量が指定サイズ以上の場合に保存する

再パブリッシュデータを保存する際に必要なSDメモ리카ードの空き容量(64MB/128MB/256MB/512MB/1024MB)を設定します。デフォルト値は256MBです。

SDメモ리카ードの空き容量が指定サイズ以上のとき、パブリッシュに失敗したMQTTパケットを再パブリッシュデータとして保存します。

SDメモ리카ードの空き容量が指定サイズ未満のとき、パブリッシュに失敗してもMQTTパケットを再パブリッシュデータとして保存しません。

〔動作モード設定〕 ダイアログボックス

〔動作モード設定〕 ダイアログボックスで、パブリッシュする際の動作モードを設定します。



①動作モード

パブリッシュする際の動作モードを設定します。

動作モード	内容
立ち上がりエッジ	パブリッシュコントロールをOFF→ONすると、指定したトピックにパブリッシュします。パブリッシュが完了すると、パブリッシュコントロールが自動的にOFFします。
定周期	パブリッシュコントロールがONのとき、間隔 (②) で設定した周期で、指定したトピックにパブリッシュします。パブリッシュコントロールをOFFするとパブリッシュを終了します。

②間隔

パブリッシュ処理を開始した直後から次のパブリッシュ処理を開始するまでの時間間隔を 1～3600 秒の範囲で設定します。デフォルト値は 60 秒です。次のパブリッシュ処理を開始するタイミングで、前回のパブリッシュ処理が完了していない場合、次のパブリッシュ処理の開始は待たされます。

③ [OK] ボタン

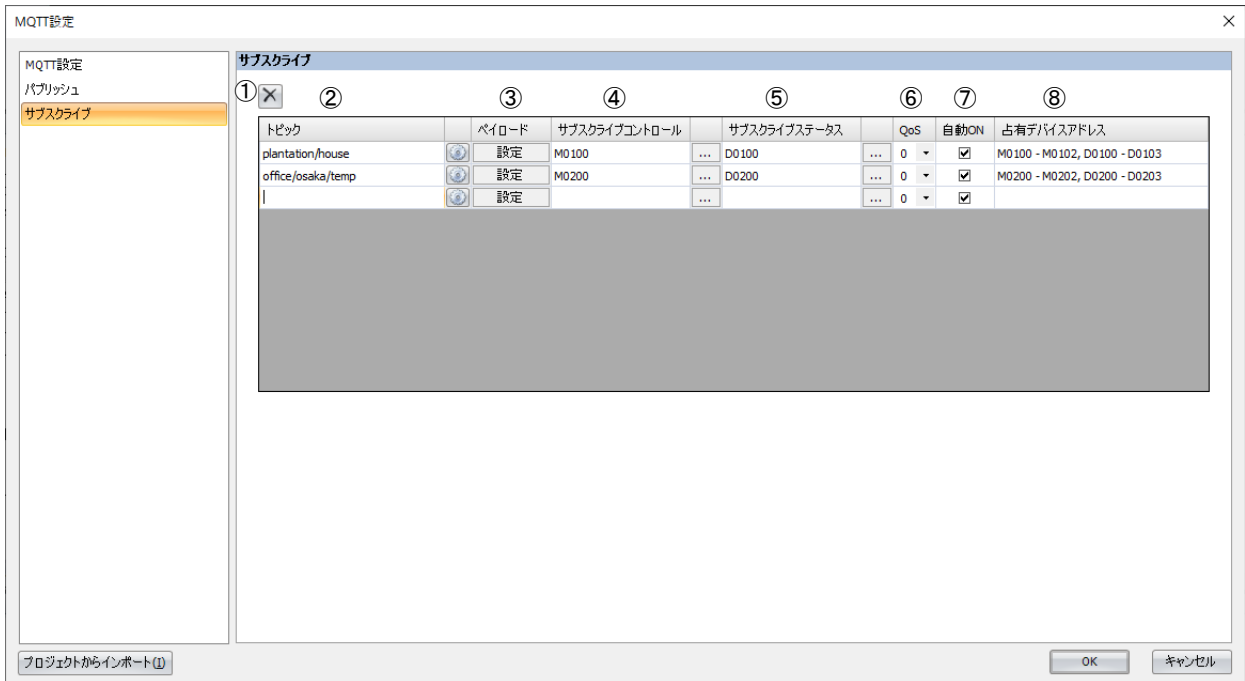
設定を保存し、〔動作モード設定〕 ダイアログボックスを閉じます。

④ [キャンセル] ボタン

設定を保存せずに、〔動作モード設定〕 ダイアログボックスを閉じます。

サブスクリブ

[MQTT 設定] ダイアログボックスで [サブスクリブ] を選択すると、[サブスクリブ] タブが表示されます。このタブでサブスクリブするための設定を行います。




- ① ボタン
選択した行の設定内容を削除します。
- ② トピック
トピックを設定します。トピックは UTF-8 で最大 256 バイトまで設定できます。なお、ワイルドカード (# および +) を設定できます。
 ボタンをクリックすると、[トピック] ダイアログボックスが表示されます。詳細は、「[トピック] ダイアログボックス」(17-40 頁) を参照してください。

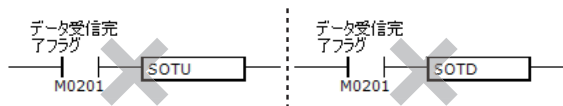


- ・トピックは最大 10 個まで設定できます。
- ・サブスクリブコントロール (④) が OFF → ON したとき、[トピック] ダイアログボックスで設定したトピックが生成されます。生成したトピックが不正な場合、そのトピックをサブスクリブできません。エラーについては、サブスクリブステータス (⑤) を参照してください。
- ・+, #, / を含むクライアント ID はトピックに使用できません。

- ③ ペイロード
ペイロード内のデータを設定します。[設定] ボタンをクリックすると、[ペイロード] ダイアログボックスが表示されます。詳細は、「[ペイロード] ダイアログボックス」(17-42 頁) を参照してください。
- ④ サブスクリブコントロール
トピックをサブスクリブするための内部リレーを設定します。指定した内部リレーを先頭に 3 ビット分の内部リレーを使用します。デバイスの範囲を超えないように先頭の内部リレーを指定してください。

格納先	項目	内容
先頭番号 + 0	サブスクリブコントロール	ON のとき、指定したトピックをサブスクリブします。OFF するとサブスクリブを解除します。
先頭番号 + 1	データ受信完了フラグ	データを正常に受信した場合 (サブスクリブステータスが 4 (0004h) のまま) に ON します。次のデータの受信を検知したい場合は、ラダープログラムで OFF してください。
先頭番号 + 2	全データ格納フラグ	データを正常に受信した場合 (サブスクリブステータスが 4 (0004h) のまま)、かつ、[ペイロード] ダイアログボックスで設定されたすべてのデバイスに値を格納した場合に ON します

 SOTU 命令または SOTD 命令をデータ受信完了フラグの入力と組み合わせて使用しないでください。これらの命令をデータ受信完了フラグの入力と組み合わせて使用すると、データの受信を認識しないことがあります。



⑤ サブスクライブステータス

トピックをサブスクライブする際のステータス等を格納するデータレジスタを設定します。設定したデータレジスタを先頭に4ワード分のデータレジスタを使用します。デバイスの範囲を超えないように先頭のデータレジスタを指定してください。

格納先	項目	内容																				
先頭番号+0	サブスクライブステータス	サブスクライブのステータスを格納します。																				
		<table border="1"> <thead> <tr> <th>ステータスコード</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>0 (0000h)</td> <td>初期状態</td> </tr> <tr> <td>2 (0002h)</td> <td>サブスクライブ処理中</td> </tr> <tr> <td>4 (0004h)</td> <td>サブスクライブ中 (サブスクライブエラーコードが5 (0005h)～8 (0008h)となる場合、ステータスは20 (0014h)となります)</td> </tr> <tr> <td>8 (0008h)</td> <td>サブスクライブ解除処理中</td> </tr> <tr> <td>16 (0010h)</td> <td>サブスクライブエラー</td> </tr> <tr> <td>32 (0020h)</td> <td>サブスクライブの解除エラー</td> </tr> </tbody> </table>	ステータスコード	状態	0 (0000h)	初期状態	2 (0002h)	サブスクライブ処理中	4 (0004h)	サブスクライブ中 (サブスクライブエラーコードが5 (0005h)～8 (0008h)となる場合、ステータスは20 (0014h)となります)	8 (0008h)	サブスクライブ解除処理中	16 (0010h)	サブスクライブエラー	32 (0020h)	サブスクライブの解除エラー						
		ステータスコード	状態																			
		0 (0000h)	初期状態																			
		2 (0002h)	サブスクライブ処理中																			
		4 (0004h)	サブスクライブ中 (サブスクライブエラーコードが5 (0005h)～8 (0008h)となる場合、ステータスは20 (0014h)となります)																			
		8 (0008h)	サブスクライブ解除処理中																			
16 (0010h)	サブスクライブエラー																					
32 (0020h)	サブスクライブの解除エラー																					
先頭番号+1	サブスクライブエラーコード	サブスクライブで発生したエラー情報を格納します。																				
		<table border="1"> <thead> <tr> <th>エラーコード*1</th> <th>エラー内容</th> </tr> </thead> <tbody> <tr> <td>1 (0001h)</td> <td>イーサネットケーブルが抜けていたり断線したりして、Plus CPUモジュールがネットワークに正しく接続できていなかった</td> </tr> <tr> <td>2 (0002h)</td> <td>接続ステータスが0 (0000h) (切断状態) のときにサブスクライブまたはサブスクライブを解除した</td> </tr> <tr> <td>4 (0004h)</td> <td>不正なフォーマットのトピック、または256バイトより大きいサイズのトピックをサブスクライブしようとした</td> </tr> <tr> <td>5 (0005h)</td> <td>受信したデータのサイズまたは要素数が範囲外</td> </tr> <tr> <td>6 (0006h)</td> <td>受信したデータがJSON形式ではない</td> </tr> <tr> <td>7 (0007h)</td> <td>データタイプ範囲外の値を受信した</td> </tr> <tr> <td>8 (0008h)</td> <td>[ペイロード] ダイアログボックスで設定されたすべてのデバイスに値を格納しなかった</td> </tr> <tr> <td>96 (0060h)</td> <td>MQTTパケット受信タイムアウトエラー</td> </tr> <tr> <td>32768 (8000h)</td> <td>ブローカー応答異常</td> </tr> </tbody> </table>	エラーコード*1	エラー内容	1 (0001h)	イーサネットケーブルが抜けていたり断線したりして、Plus CPUモジュールがネットワークに正しく接続できていなかった	2 (0002h)	接続ステータスが0 (0000h) (切断状態) のときにサブスクライブまたはサブスクライブを解除した	4 (0004h)	不正なフォーマットのトピック、または256バイトより大きいサイズのトピックをサブスクライブしようとした	5 (0005h)	受信したデータのサイズまたは要素数が範囲外	6 (0006h)	受信したデータがJSON形式ではない	7 (0007h)	データタイプ範囲外の値を受信した	8 (0008h)	[ペイロード] ダイアログボックスで設定されたすべてのデバイスに値を格納しなかった	96 (0060h)	MQTTパケット受信タイムアウトエラー	32768 (8000h)	ブローカー応答異常
		エラーコード*1	エラー内容																			
		1 (0001h)	イーサネットケーブルが抜けていたり断線したりして、Plus CPUモジュールがネットワークに正しく接続できていなかった																			
		2 (0002h)	接続ステータスが0 (0000h) (切断状態) のときにサブスクライブまたはサブスクライブを解除した																			
		4 (0004h)	不正なフォーマットのトピック、または256バイトより大きいサイズのトピックをサブスクライブしようとした																			
		5 (0005h)	受信したデータのサイズまたは要素数が範囲外																			
		6 (0006h)	受信したデータがJSON形式ではない																			
		7 (0007h)	データタイプ範囲外の値を受信した																			
8 (0008h)	[ペイロード] ダイアログボックスで設定されたすべてのデバイスに値を格納しなかった																					
96 (0060h)	MQTTパケット受信タイムアウトエラー																					
32768 (8000h)	ブローカー応答異常																					
先頭番号+2	受信データバイト数	受信したデータのサイズをバイト単位で格納します。																				
先頭番号+3	サブスクライブエラー ID	以下のいずれかのエラーが発生した場合、最初にエラーが発生したID*2を格納します。 <ul style="list-style-type: none"> 受信した JSON 形式のデータ内に、[ペイロード] ダイアログボックスで設定した ID の深さレベル、名前およびフォーマットと一致する要素が存在しない データタイプ範囲外の値を受信した 																				

*1 サブスクライブエラーコードが4 (0004h)～8 (0008h)以外のときは接続コントロールもOFFとなります。サブスクライブエラーコードが4 (0004h)～8 (0008h)のとき、サブスクライブコントロールおよび接続コントロールの状態は保持されます。

*2 IDとは、[ペイロード] ダイアログボックスで各要素を識別するために割り付けられた番号です。詳細は、「[ペイロード] ダイアログボックス」(17-42頁)を参照してください。

⑥ QoS

QoSを0～2の範囲で設定します。デフォルト値は0です。

⑦ 自動 ON

Plus CPU モジュールをブローカーに接続した直後に、サブスクライブコントロールを自動的にONするかどうかを設定します。チェックボックスがオンの場合、ブローカーに接続直後、サブスクライブコントロールを自動的にONします。デフォルト値はオンです。

⑧ 占有デバイスアドレス

サブスクライブコントロールおよびサブスクライブステータスを設定することで占有されるデバイスアドレスが表示されます。

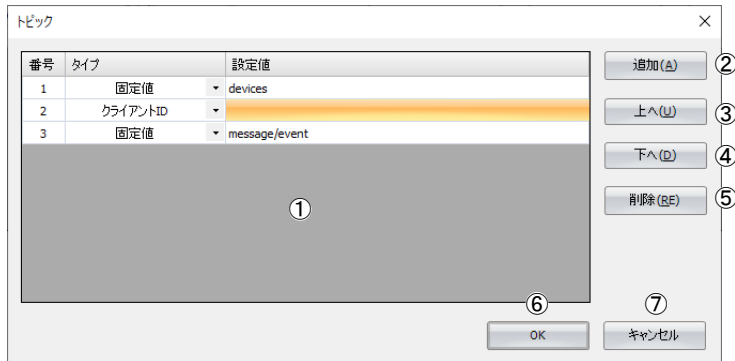
[トピック] ダイアログボックス

[トピック] ダイアログボックスでトピックを設定します。

[パブリッシュ] タブから [トピック] ダイアログボックスを開いた場合、パブリッシュするトピックを設定します。

[サブスクライブ] タブから [トピック] ダイアログボックスを開いた場合、サブスクライブするトピックを設定します。

各行にトピックを形成する文字列を設定します。複数行を設定した場合、各行の設定値を番号順に "/" で連結した文字列がトピックになります。



① (設定一覧)

設定が一覧表示されます。

番号 : 各行を識別する番号 (1 ~ 10) が表示されます。

タイプ : トピックを形成する文字列のタイプが表示されます。

タイプ	内容
固定値	任意の文字列を設定します。
クライアント ID/ デバイス ID	[MQTT 設定] タブの [MQTT 基本設定] グループで設定したクライアント ID または デバイス ID が設定されます。

設定値 : [タイプ] に応じた設定値が表示されます。

② [追加] ボタン

最下に 1 行を追加します。

③ [上へ] ボタン

(設定一覧) で選択している行と、1 つ上の行を交換します。

④ [下へ] ボタン

(設定一覧) で選択している行と、1 つ下の行を交換します。

⑤ [削除] ボタン

(設定一覧) で選択している行を削除します。

⑥ [OK] ボタン

設定を保存し、[トピック] ダイアログボックスを閉じます。

⑦ [キャンセル] ボタン

設定を保存せずに、[トピック] ダイアログボックスを閉じます。



- 1 トピックは最大 10 個（番号 1 ～番号 10）の文字列を使って設定できます。
- Plus CPU モジュールのシステムソフトウェアバージョンによって、トピックの設定方法が異なります。

システムソフトウェア	トピックの設定方法
バージョン 2.10 未満	1 個の固定値のみで設定する
バージョン 2.10 以上	上記以外

トピックを 1 個の固定値のみで設定する例：

トピック		
番号	タイプ	設定値
1	固定値	devices/message/event

トピックを複数個の固定値で設定する例：

トピック		
番号	タイプ	設定値
1	固定値	devices
2	固定値	message/event

[ペイロード] ダイアログボックス

JavaScript Object Notation (JSON) 形式のペイロード内のデータを設定します。

[パブリッシュ] タブから [ペイロード] ダイアログボックスを開いた場合、送信する JSON 形式のデータの内容を設定します。

[サブスクライブ] タブから [ペイロード] ダイアログボックスを開いた場合、受信した JSON 形式のデータについての値をどのデバイスに格納するかを設定します。

ID	名前	フォーマット	データタイプ	データ
1	(root)	オブジェクト (3)		
2	Plastic greenhouse (North)	オブジェクト (5)		
3	Temperature (Lower) (°C)	値	フロート(F)	D0510-D0511
4	Temperature (Upper) (°C)	値	フロート(F)	D0512-D0513
5	Humidity (Lower) (%)	値	フロート(F)	D0514-D0515
6	Humidity (Upper) (%)	値	フロート(F)	D0516-D0517
7	CO2 Level (Lower) (ppm)	値	フロート(F)	D0518-D0519
8	Plastic greenhouse (South)	オブジェクト (5)		
9	Temperature (Lower) (°C)	値	フロート(F)	D0610-D0611
10	Temperature (Upper) (°C)	値	フロート(F)	D0612-D0613
11	Humidity (Lower) (%)	値	フロート(F)	D0614-D0615
12	Humidity (Upper) (%)	値	フロート(F)	D0616-D0617
13	CO2 Level (Lower) (ppm)	値	フロート(F)	D0618-D0619
14	Timestamp	値	タイムスタンプ (T)	1604674555

②サイズ: 482 バイト (最大 32768 バイト)
 ③ID数: 14 (最大 800)
 ④深さ: 3 (最大 10)

① (設定一覧)

設定が一覧表示されます。

- ID : 各行を識別する番号が表示されます。
- 名前 : 各 ID の名前が表示されます。
- フォーマット : 各 ID のフォーマット (オブジェクト、配列または値) が表示されます。オブジェクトおよび配列は子要素の数も表示されます。
- データタイプ : 各値のデータタイプが表示されます。データタイプが文字列 (S) の場合、そのサイズも表示されます。
- データ : 固定値またはデバイスアドレスが表示されます。デバイスアドレスの場合、占有されるデバイスアドレスの範囲が表示されます。



ID1 の "名前" は (root) になります。配列の子要素の "名前" は、0 から始まる通し番号になります。

②サイズ

(設定一覧) で設定した JSON 形式のデータの最大サイズが表示されます。最大サイズは [パブリッシュ] タブから [ペイロード] ダイアログボックスを開いた場合 32768 バイト、[サブスクライブ] タブから [ペイロード] ダイアログボックスを開いた場合 8192 バイトです。

③ ID 数

ID の数が表示されます。最大値は [パブリッシュ] タブから [ペイロード] ダイアログボックスを開いた場合 800、[サブスクライブ] タブから [ペイロード] ダイアログボックスを開いた場合 200 です。

④深さ

(設定一覧) で設定した JSON 形式のデータの現在のネストの深さが表示されます。最大値は 10 です。



深さレベル 1 をルートと呼びます。

⑤ [新規オブジェクト] ボタン

(設定一覧) で選択しているオブジェクトまたは配列の末尾にオブジェクトを追加します。(設定一覧) が空の場合、ルートにオブジェクトを追加します。

- ⑥ **[新規配列] ボタン**
 (設定一覧) で選択しているオブジェクトまたは配列の末尾に配列を追加します。(設定一覧) が空の場合、ルートに配列を追加します。
- ⑦ **[一括] メニュー**
 連続したデバイスの子要素とする配列を追加します。このボタンをクリックすると、[一括] ダイアログボックスが表示されます。詳細は、「[一括] ダイアログボックス」(17-46 頁) を参照してください。
- ⑧ **[新規値] ボタン**
 (設定一覧) で選択しているオブジェクトまたは配列の末尾に値を追加します。(設定一覧) が空の場合、ルートに値を追加します。このボタンをクリックすると、[新規値] ダイアログボックスが表示されます。詳細は、「[新規値] または [編集] ダイアログボックス」(17-44 頁) を参照してください。
- ⑨ **[編集] ボタン**
 このボタンをクリックすると、[編集] ダイアログボックスが表示されます。詳細は、「[新規値] または [編集] ダイアログボックス」(17-44 頁) を参照してください。
- ⑩ **[削除] ボタン**
 (設定一覧) で選択している ID を、フォーマットに応じて次のとおり削除します。

選択している ID のフォーマット	動作
オブジェクト	選択しているオブジェクトおよびその子要素を削除します。
配列	選択している配列およびその子要素を削除します。
値	選択している値を削除します。

- ⑪ **[上へ] ボタン**
 (設定一覧) で選択している ID と、同じ深さに存在する 1 つ上の ID を交換します。
- ⑫ **[下へ] ボタン**
 (設定一覧) で選択している ID と、同じ深さに存在する 1 つ下の ID を交換します。
- ⑬ **[JSON テキストをインポート] ボタン**
 [JSON テキストをインポート] ダイアログボックスが表示されます。詳細は、「[JSON テキストをインポート] ダイアログボックス」(17-47 頁) を参照してください。
- ⑭ **[JSON テキストをエクスポート] ボタン**
 [JSON テキストをエクスポート] ダイアログボックスが表示されます。詳細は、「[JSON テキストをエクスポート] ダイアログボックス」(17-48 頁) を参照してください。
- ⑮ **[OK] ボタン**
 設定を保存し、[ペイロード] ダイアログボックスを閉じます。
- ⑯ **[キャンセル] ボタン**
 設定を保存せずに、[ペイロード] ダイアログボックスを閉じます。

〔新規値〕 または 〔編集〕 ダイアログボックス

オブジェクトまたは配列を編集します。または、値を追加または編集します。

①名前

値の名前を表示または設定します。名前は UTF-8 で最大 255 バイトまで設定できます。

②データタイプ

値のデータタイプを設定します。値を追加または編集するときのみ表示されます。各データタイプで扱えるデータの範囲は次のとおりです。

データタイプ	範囲
ワード (W)	0~65535
インテジャ (I)	-32768~32767
ダブルワード (D)	0~4294967295
ロング (L)	-2147483648~2147483647
フロート (F)	32ビット単精度浮動小数点で扱える範囲*1 ただし±∞ (±無限大) および非数を除く。
ブーリアン (B)	true (1) または false (0)
文字列 (S)	文字コード: UTF-8 最大サイズ: 1023バイト
タイムスタンプ (T) *2	タイムゾーン: カレンダ&時計の設定を使用する*3、UTC+14~UTC-12 フォーマット*4: ローカル時間、UTC、UNIX時間

*1 ラダープログラミングマニュアル「第3章 ●データタイプについて」を参照してください。

*2 [パブリッシュ] タブから [ペイロード] ダイアログボックスを開いた場合のみ表示されます。

*3 [ファンクション設定] ダイアログボックスの [カレンダ&時計] タブで設定したタイムゾーンを使用します。

*4 フォーマットの説明は下表を参照してください。

フォーマット		説明
ローカル時間 (s)	YYYYMMDDThhmmss+/-hhmm	ISO8601基本形式/拡張形式のローカル時間
	YYYY-MM-DDThh:mm:ss+/-hh:mm	
ローカル時間 (ms) *5	YYYYMMDDThhmmssSSS+/-hhmm	
	YYYY-MM-DDThh:mm:ss.SSS+/-hh:mm	
UTC (s)	YYYYMMDDThhmmssZ	ISO8601基本形式/拡張形式のUTC
	YYYY-MM-DDThh:mm:ssZ	
UTC (ms) *5	YYYYMMDDThhmmssSSSZ	
	YYYY-MM-DDThh:mm:ss.SSSZ	
UNIX時間 (s)		1970年1月1日 (UTC+0) からの、うるう秒を除く経過時間 (単位: s)
UNIX時間 (ms) *5		1970年1月1日 (UTC+0) からの、うるう秒を除く経過時間 (単位: ms)

*5 ms 単位を指定した場合、ms の桁には常に 000 が入ります。

たとえば、フォーマットで“ローカル時間 (ms)”の“YYYY-MM-DDThh:mm:ss.SSS+/-hh:mm”が選択されている場合、2022-05-11T08:30:05.000+09:00 が表示されます。



タイムゾーンは、タイムゾーンオフセット (D8413) を使用して 15 分単位で調整できます。詳細は、「第 3 章 タイムゾーンの調整 (D8413)」(3-12 頁) を参照してください。

③データ

値のデータを設定します。値を追加または編集するときのみ表示されます。

データタイプ	設定方法	内容
ワード (W) / インテジャ (I) / ダブルワード (D) / ロング (L) フロート (F)	固定値	数値を設定します。 設定できる数値の範囲は、データタイプによって異なります。
	デバイス	データレジスタを設定します。*1
ブーリアン (B)	固定値	trueまたはfalseを設定します。
	デバイス	内部リレーを設定します。
文字列 (S)	固定値	文字列を設定します。
	デバイス	データレジスタおよびサイズ (バイト) を1~1023の範囲で設定します。*2
タイムスタンプ (T) *3	—	タイムゾーンおよびフォーマットを設定します。

*1 データタイプがD (ダブルワード)、L (ロング) またはF (フロート) の場合、設定したデータレジスタを先頭に2ワード分のデータレジスタを使用します。

*2 設定したデータレジスタを先頭アドレスとして、設定されたサイズ分のアドレスを使用します。

*3 [パブリッシュ] タブから [ペイロード] ダイアログボックスを開いた場合のみ表示されます。

④ [OK] ボタン

[OK] ボタンをクリックすると、(設定一覧) に値を追加、または編集内容を保存します。

⑤ [キャンセル] ボタン

[キャンセル] ボタンをクリックすると、値を追加または編集を保存しません。

[一括] ダイアログボックス

配列およびその配列の子要素（値）を追加します。

①名前

追加する配列の名前を表示または設定します。名前は UTF-8 で最大 255 バイトまで設定できます。

②データタイプ

配列の各子要素（値）のデータタイプを設定します。

データタイプ	範囲
ワード (W)	0~65535
インデジャ (I)	-32768~32767
ダブルワード (D)	0~4294967295
ロング (L)	-2147483648~2147483647
フロート (F)	32ビット単精度浮動小数点で扱える範囲*1 ただし±∞（±無限大）および非数を除く。
ブーリアン (B)	true (1) または false (0)
文字列 (S)	文字コード：UTF-8 最大サイズ：1023バイト

*1 ラダープログラミングマニュアル「第3章 ●データタイプについて」を参照してください。

③データ

配列の子要素（値）のデータを設定します。

データタイプ	設定方法	内容
ワード (W) / インデジャ (I) / ダブルワード (D) / ロング (L) / フロート (F)	デバイス	データレジスタを設定します。*1
ブーリアン (B)		内部リレーを設定します。
文字列 (S)		データレジスタおよびサイズ（バイト）を1~1023の範囲で設定します。*2

*1 データタイプが D（ダブルワード）、L（ロング）または F（フロート）の場合、設定したデータレジスタを先頭に 2ワード分のデータレジスタを使用します。

*2 設定したデータレジスタを先頭アドレスとして、設定されたサイズ分のアドレスを使用します。

④長さ

配列の子要素（値）の数を設定します。長さは、使用できる残りの ID の数まで設定できます。デフォルト値は 1 です。

⑤ [OK] ボタン

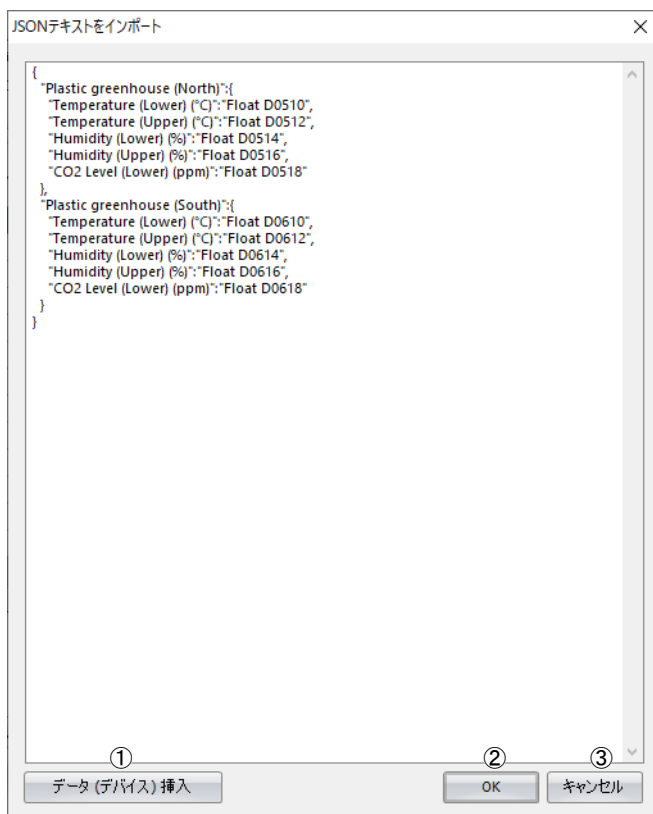
[OK] ボタンをクリックすると、配列を追加します。

⑥ [キャンセル] ボタン

[キャンセル] ボタンをクリックすると、配列を追加しません。

[JSON テキストをインポート] ダイアログボックス

インポートしたい JSON 形式のテキストを設定します。



① データ (デバイス) 挿入

[データ (デバイス) 挿入] ダイアログボックスを開きます。

② [OK] ボタン

テキストの内容を [ペイロード] ダイアログボックスの (設定一覧) に反映します。



JSON テキスト内の各キーと値のペアの値の部分で、データタイプとデバイスが指定されている場合、デバイスに変換してインポートします。

例) { "key": "Word D100" }

ID2 のデータタイプはワード (W) になり、デバイスは D0100 になります。

{ "key": "String D200 10" }

ID2 のデータタイプは文字列 (S) になり、デバイスは D0200 になります。サイズは 10 バイトになります。

{ "Timestamp": "Timestamp UTC+0 UnixTime" }

ID2 のデータタイプはタイムスタンプ (T) になり、タイムゾーンは UTC+0、フォーマットは UNIX 時間 (s) になります。

③ [キャンセル] ボタン

テキストの内容を [ペイロード] ダイアログボックスの (設定一覧) に反映せずに、[ペイロード] ダイアログボックスに戻ります。

[JSON テキストをエクスポート] ダイアログボックス

[ペイロード] ダイアログボックスで設定したペイロードを JSON 形式のテキストで出力します。



- ① [クリップボードにコピー] ボタン
テキストの内容をクリップボードにコピーします。
- ② [ファイルに保存] ボタン
テキストの内容をテキスト形式 (.txt) で保存します。
- ③ [キャンセル] ボタン
[JSON テキストをエクスポート] ダイアログボックスを閉じます。

[データ (デバイス) 挿入] ダイアログボックス

デバイスとデータタイプを指定する文字列を作成します。

①データタイプ

挿入するデータ (デバイス) のデータタイプを設定します。

データタイプ	範囲
ワード (W)	0~65535
インテジャ (I)	-32768~32767
ダブルワード (D)	0~4294967295
ロング (L)	-2147483648~2147483647
フロート (F)	32ビット単精度浮動小数点で扱える範囲*1 ただし±∞ (±無限大) および非数を除く。
ブーリアン (B)	true (1) またはfalse (0)
文字列 (S)	文字コード：UTF-8 最大サイズ：1023バイト
タイムスタンプ (T) *2	タイムゾーン：カレンダー&時計の設定を使用する*3、UTC+14~UTC-12 フォーマット*4：ローカル時間、UTC、UNIX時間

*1 ラダープログラミング マニュアル「第3章 ●データタイプについて」を参照してください。

*2 [パブリッシュ] タブから [ペイロード] ダイアログボックスを開いた場合のみ表示されます。

*3 [ファンクション設定] ダイアログボックスの [カレンダー&時計] タブで設定したタイムゾーンを使用します。

*4 フォーマットの説明は下表を参照してください。

	フォーマット	説明
ローカル時間 (s)	YYYYMMDDThhmmss+/-hhmm	ISO8601基本形式/拡張形式のローカル時間
	YYYY-MM-DDThh:mm:ss+/-hh:mm	
ローカル時間 (ms) *5	YYYYMMDDThhmmssSSS+/-hhmm	
	YYYY-MM-DDThh:mm:ss.SSS+/-hh:mm	
UTC (s)	YYYYMMDDThhmmssZ	ISO8601基本形式/拡張形式のUTC
	YYYY-MM-DDThh:mm:ssZ	
UTC (ms) *5	YYYYMMDDThhmmssSSSZ	
	YYYY-MM-DDThh:mm:ss.SSSZ	
UNIX時間 (s)		1970年1月1日 (UTC+0) からの、うるう秒を除く経過時間 (単位：s)
UNIX時間 (ms) *5		1970年1月1日 (UTC+0) からの、うるう秒を除く経過時間 (単位：ms)

*5 ms 単位を指定した場合、ms の桁には常に 000 が表示されます。

たとえば、フォーマットで“ローカル時間 (ms)”の“YYYY-MM-DDThh:mm:ss.SSS+/-hh:mm”が選択されている場合、2022-05-11T08:30:05.000+09:00 が表示されます。



タイムゾーンは、タイムゾーンオフセット (D8413) を使用して 15 分単位で調整できます。詳細は、「第3章 タイムゾーンの調整 (D8413)」(3-12 頁) を参照してください。

②データ

挿入するデータ（デバイス）のデータを設定します。

データタイプ	設定方法	内容
ワード (W) / インデジャ (I) / ダブルワード (D) / ロング (L) / フロート (F)	デバイス	データレジスタを設定します。*1
ブーリアン (B)		内部リレーを設定します。
文字列 (S)		データレジスタおよびサイズ (バイト) を1~1023の範囲で設定します。*2
タイムスタンプ (T) *3	—	タイムゾーンおよびフォーマットを設定します。

*1 データタイプが D (ダブルワード)、L (ロング) または F (フロート) の場合、設定したデータレジスタを先頭に 2 ワード分のデータレジスタを使用します。

*2 設定したデータレジスタを先頭アドレスとして、設定されたサイズ分のアドレスを使用します。

*3 [パブリッシュ] タブから [ペイロード] ダイアログボックスを開いた場合のみ表示されます。

③ [OK] ボタン

設定されたデータタイプおよびデータから文字列を作成し、それを [JSON テキストをインポート] ダイアログボックスのカーソル位置に挿入します。

④ [キャンセル] ボタン

データ（デバイス）の挿入を中止し、[JSON テキストをインポート] ダイアログボックスに戻ります。

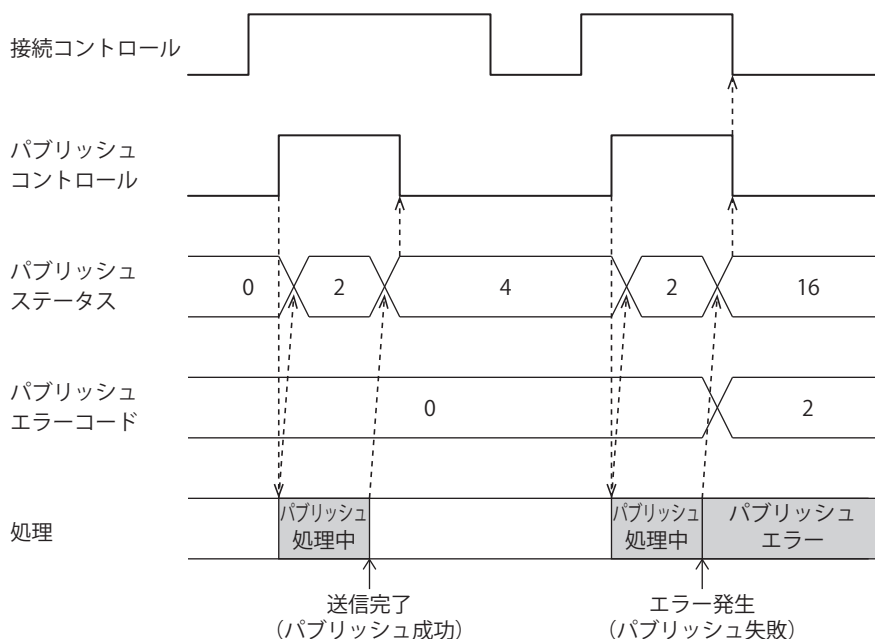
トピックにパブリッシュする

Plus CPU モジュールは、[動作モード設定] ダイアログボックスの "動作モード" の設定にしたがってパブリッシュします。詳細は、「[動作モード設定] ダイアログボックス」(17-37 頁) を参照してください。

動作モードで立ち上がりエッジを設定した場合

Plus CPU モジュールをブローカーに接続した後、パブリッシュコントロールを OFF → ON すると、指定したトピックにパブリッシュ、パブリッシュステータスにステータスが格納されます。

パブリッシュが成功すると、ステータス 4 (0004h) が格納され、パブリッシュコントロールは OFF になります。パブリッシュが失敗すると、ステータス 16 (0010h)、パブリッシュエラーコードにエラーコードが格納され、パブリッシュコントロールは OFF となります。エラーコード 7 (0007h) 以外の場合は接続コントロールも OFF となります。エラーコード 7 (0007h) のとき、パブリッシュコントロールおよび接続コントロールの状態は保持されます。



パブリッシュコントロールはラダープログラムで常時 ON しないでください。パブリッシュが成功すると、ステータス 4 (0004h) が格納され、パブリッシュコントロールは OFF になります。パブリッシュコントロールを常時 ON するようなラダープログラムの場合、繰り返しパブリッシュします。多額の packets 通信料が発生する恐れがありますので注意してください。



パブリッシュコントロールをラダープログラムで OFF する場合、ステータス 2 (0002h) 以外の場合にパブリッシュコントロールを OFF すると、パブリッシュが実行されないことがあります。ステータス 2 (0002h) のときに、[ペイロード] ダイアログボックスで設定したデバイスの値を変更しないでください。ステータス 2 (0002h) 以外の場合に変更してください。

動作例

動作説明

- いちごを栽培しているビニールハウス内の2か所（北側、南側）の温度、湿度およびCO₂濃度を計測します。
- 計測した温度、湿度およびCO₂濃度が次の範囲外である場合、トピック "plantation/house" にパブリッシュします。

項目	範囲
温度	15.0~25.0 °C
湿度	60.0~80.0 %
CO ₂ 濃度	400.0 ppm~

設定

[MQTT設定] ダイアログボックスの [MQTT設定] タブで基本設定を行った後、各タブで次のように項目を設定します。

タブ	項目	内容
[MQTT設定] タブ	接続コントロール	M0000
	接続ステータス	D0000
[パブリッシュ] タブ	トピック	"plantation/house"
	[パイロード] ダイアログボックスの設定	下図参照
	動作モード	立ち上がりエッジ
	パブリッシュコントロール	M0100
	パブリッシュステータス	D1000
	QoS	0
	Retain	無効

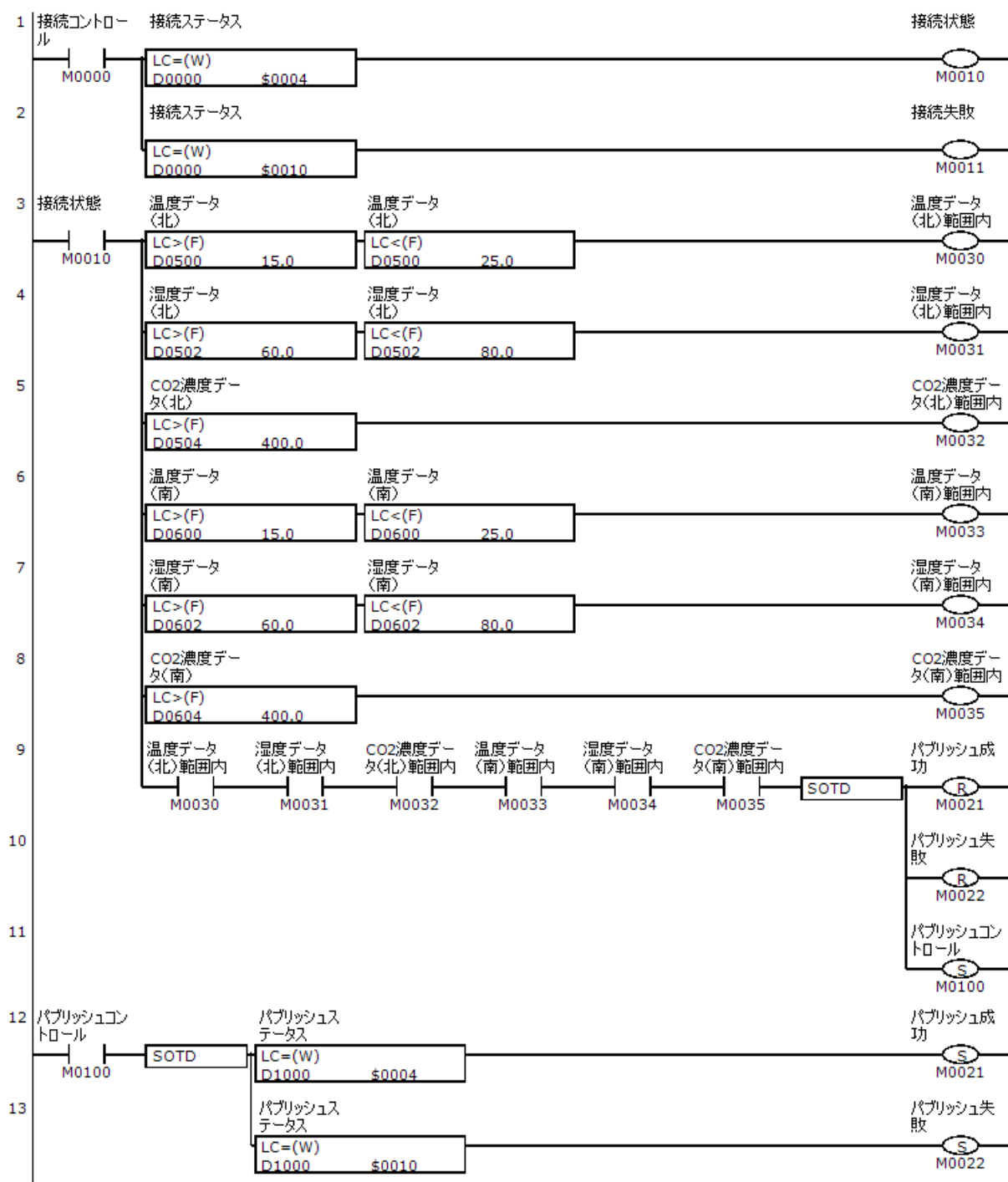
[パイロード] ダイアログボックスの設定

パイロード ×

ID	名前	フォーマット	データタイプ	データ
1	[-] (root)	オブジェクト (3)		
2	[-] Plastic greenhouse (North)	オブジェクト (3)		
3	Temperature (°C)	値	フロート(F)	D0500-D0501
4	Humidity (%)	値	フロート(F)	D0502-D0503
5	CO2 Level (ppm)	値	フロート(F)	D0504-D0505
6	[-] Plastic greenhouse (South)	オブジェクト (3)		
7	Temperature (°C)	値	フロート(F)	D0600-D0601
8	Humidity (%)	値	フロート(F)	D0602-D0603
9	CO2 Level (ppm)	値	フロート(F)	D0604-D0605
10	Timestamp	値	タイムスタンプ(T)	1604554494

サイズ: 276 バイト (最大32768バイト)
ID数: 10 (最大800)
深さ: 3 (最大10)

ラダープログラム



デバイスアドレス	コメント
M0000	接続コントロール
M0010	接続状態
M0011	接続失敗
M0020	パブリッシュ要求
M0021	パブリッシュ成功
M0022	パブリッシュ失敗
M0030	温度データ（北） 範囲内
M0031	湿度データ（北） 範囲内
M0032	CO2濃度データ（北） 範囲内
M0033	温度データ（南） 範囲内
M0034	湿度データ（南） 範囲内
M0035	CO2濃度データ（南） 範囲内
M0100	パブリッシュコントロール

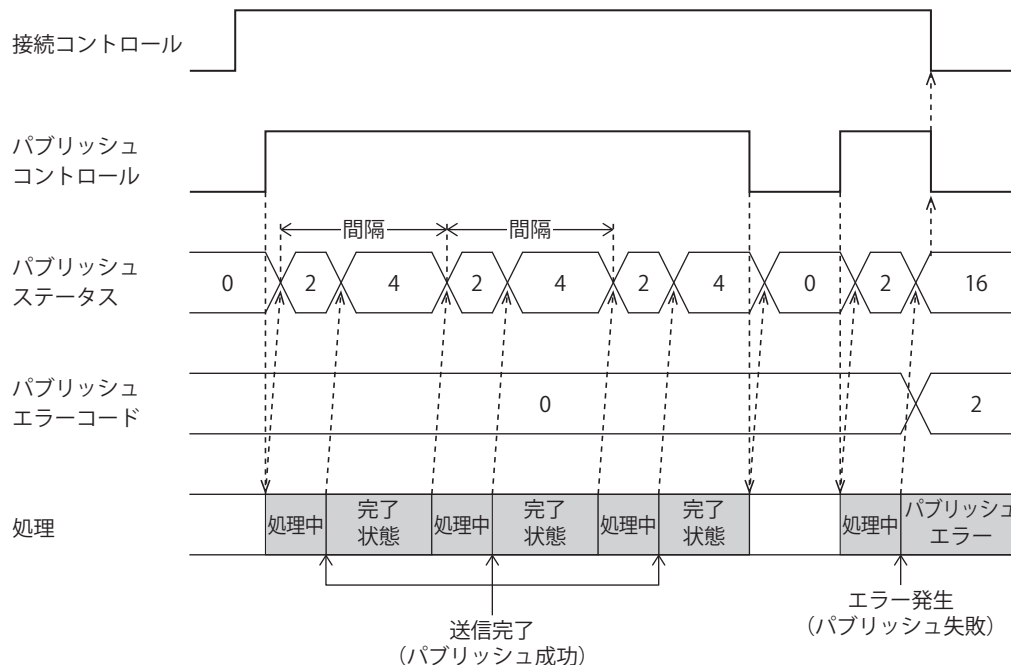
デバイスアドレス	コメント
D0000	接続ステータス
D0500	温度データ（北）
D0502	湿度データ（北）
D0504	CO2濃度データ（北）
D0600	温度データ（南）
D0602	湿度データ（南）
D0604	CO2濃度データ（南）
D1000	パブリッシュステータス

ラダー行	説明												
1	M0000をONすると、ブローカーへの接続を開始します。M0010がONのとき、ブローカーに接続した状態です。												
2	ブローカーへ接続できなかったとき、M0011がONします。												
3～8	M0010がONのとき（ブローカーへ接続した状態のとき）、ビニールハウス内の2か所（北側、南側）で計測した温度、湿度およびCO ₂ 濃度が設定範囲内であれば、M0030～M0035がONします。												
9, 10	M0010がONのとき（ブローカーへ接続した状態のとき）、温度、湿度およびCO ₂ 濃度のいずれか1つでも設定範囲外になると、該当する内部リレー（M0030～M0035）がONからOFFに変化します。このとき、D1000に0 (0000h)を格納して、パブリッシュステータスをクリアします。さらにM0100がONして、トピックにパブリッシュします。（パブリッシュが完了すると、M0100は自動的にOFFします。）												
11	<p>パブリッシュが成功したとき、M0021がONします。</p> <p>下表のとき、(D0500,D0501)=18.5、(D0502,D0503)=55.5、(D0504,D0505)=410.1、(D0600,D0601)=26.1、(D0602,D0603)=64.5、(D0604,D0605)=420.2となります。</p> <table border="1" data-bbox="414 1131 1212 1288"> <thead> <tr> <th>項目</th> <th>ビニールハウス内（北側）</th> <th>ビニールハウス内（南側）</th> </tr> </thead> <tbody> <tr> <td>温度</td> <td>18.5 °C</td> <td>26.1 °C</td> </tr> <tr> <td>湿度</td> <td>55.5 %</td> <td>64.5 %</td> </tr> <tr> <td>CO₂ 濃度</td> <td>410.1 ppm</td> <td>420.2 ppm</td> </tr> </tbody> </table> <p>さらに内蔵時計の現在時刻のデータ（D8008～D8014）が2020年12月9日9時5分46秒およびD8413（タイムゾーンオフセット）の値が0である場合、送信されるJSON形式のデータは次のとおりです。</p> <pre> { "Plastic greenhouse (North)":{ "Temperature (°C)": 18.5, "Humidity (%)": 55.5, "CO2 Level (ppm)": 410.1 }, "Plastic greenhouse (South)":{ "Temperature (°C)": 26.1, "Humidity (%)": 64.5, "CO2 Level (ppm)": 420.2 }, "Timestamp": 1607504746 } </pre> <p>このときD1002（送信データバイト数）には224が格納されます。</p>	項目	ビニールハウス内（北側）	ビニールハウス内（南側）	温度	18.5 °C	26.1 °C	湿度	55.5 %	64.5 %	CO ₂ 濃度	410.1 ppm	420.2 ppm
項目	ビニールハウス内（北側）	ビニールハウス内（南側）											
温度	18.5 °C	26.1 °C											
湿度	55.5 %	64.5 %											
CO ₂ 濃度	410.1 ppm	420.2 ppm											
12	パブリッシュが失敗したとき、M0022がONします。												

動作モードで定周期を設定した場合

Plus CPU モジュールをブローカーに接続した後、パブリッシュコントロールが ON のとき、間隔 (②) で設定した周期で、指定したトピックにパブリッシュし、パブリッシュステータスにステータスを格納します。パブリッシュコントロールを OFF すると、パブリッシュを終了します。このときステータス 0 (0000h) が格納されます。

パブリッシュが失敗すると、ステータス 16 (0010h)、パブリッシュエラーコードにエラーコードが格納され、パブリッシュコントロールは OFF となります。エラーコード 7 (0007h) 以外のときは接続コントロールも OFF となります。エラーコード 7 (0007h) のとき、パブリッシュコントロールおよび接続コントロールの状態は保持されます。



ステータス 2 (0002h) のときに、[ペイロード] ダイアログボックスで設定したデバイスの値を変更しないでください。ステータス 2 (0002h) 以外の変更にしてください。

動作例

動作説明

- いちごを栽培しているビニールハウス内の2か所の温度、湿度およびCO₂濃度を30分毎に計測し、トピック "plantation/house" にパブリッシュします。
- 遠隔地からビニールハウス内の状態を確認します。

設定

[MQTT設定] ダイアログボックスの [MQTT設定] タブで基本設定を行った後、各タブで次のように項目を設定します。

タブ	項目	内容	
[MQTT設定] タブ	接続コントロール	M0000	
	接続ステータス	D0000	
[パブリッシュ] タブ	トピック	"plantation/house"	
	[ペイロード] ダイアログボックスの設定	下図参照	
	動作モード	動作モード：定周期	
		間隔：1800秒	
	パブリッシュコントロール	M0100	
	パブリッシュステータス	D1000	
	QoS	1	
Retain	無効		

[ペイロード] ダイアログボックスの設定

ペイロード

ID	名前	フォーマット	データタイプ	データ
1	(root)	オブジェクト (3)		
2	Plastic greenhouse (North)	オブジェクト (3)		
3	Temperature (°C)	値	フロート(F)	D0500-D0501
4	Humidity (%)	値	フロート(F)	D0502-D0503
5	CO2 Level (ppm)	値	フロート(F)	D0504-D0505
6	Plastic greenhouse (South)	オブジェクト (3)		
7	Temperature (°C)	値	フロート(F)	D0600-D0601
8	Humidity (%)	値	フロート(F)	D0602-D0603
9	CO2 Level (ppm)	値	フロート(F)	D0604-D0605
10	Timestamp	値	タイムスタンプ(T)	2020-11-05T14:39:24Z

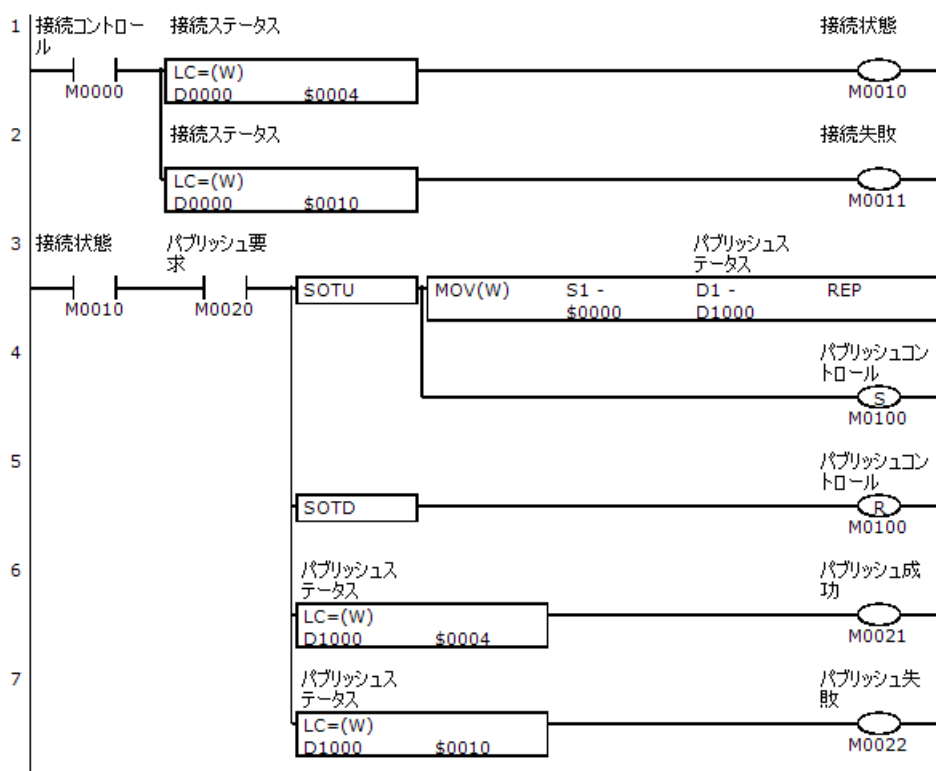
サイズ: 293 バイト (最大32768/バイト)
 ID数: 10 (最大800)
 深さ: 3 (最大10)

新規オブジェクト 新規配列 新規値 編集 削除

上へ 下へ

JSONテキストをインポート JSONテキストをエクスポート OK キャンセル

ラダープログラム



デバイスアドレス	コメント
M0000	接続コントロール
M0010	接続状態
M0011	接続失敗
M0020	パブリッシュ要求
M0021	パブリッシュ成功
M0022	パブリッシュ失敗
M0100	パブリッシュコントロール

デバイスアドレス	コメント
D0000	接続ステータス
D0500	温度データ (北)
D0502	湿度データ (北)
D0504	CO2濃度データ (北)
D0600	温度データ (南)
D0602	湿度データ (北)
D0604	CO2濃度データ (北)
D1000	パブリッシュステータス

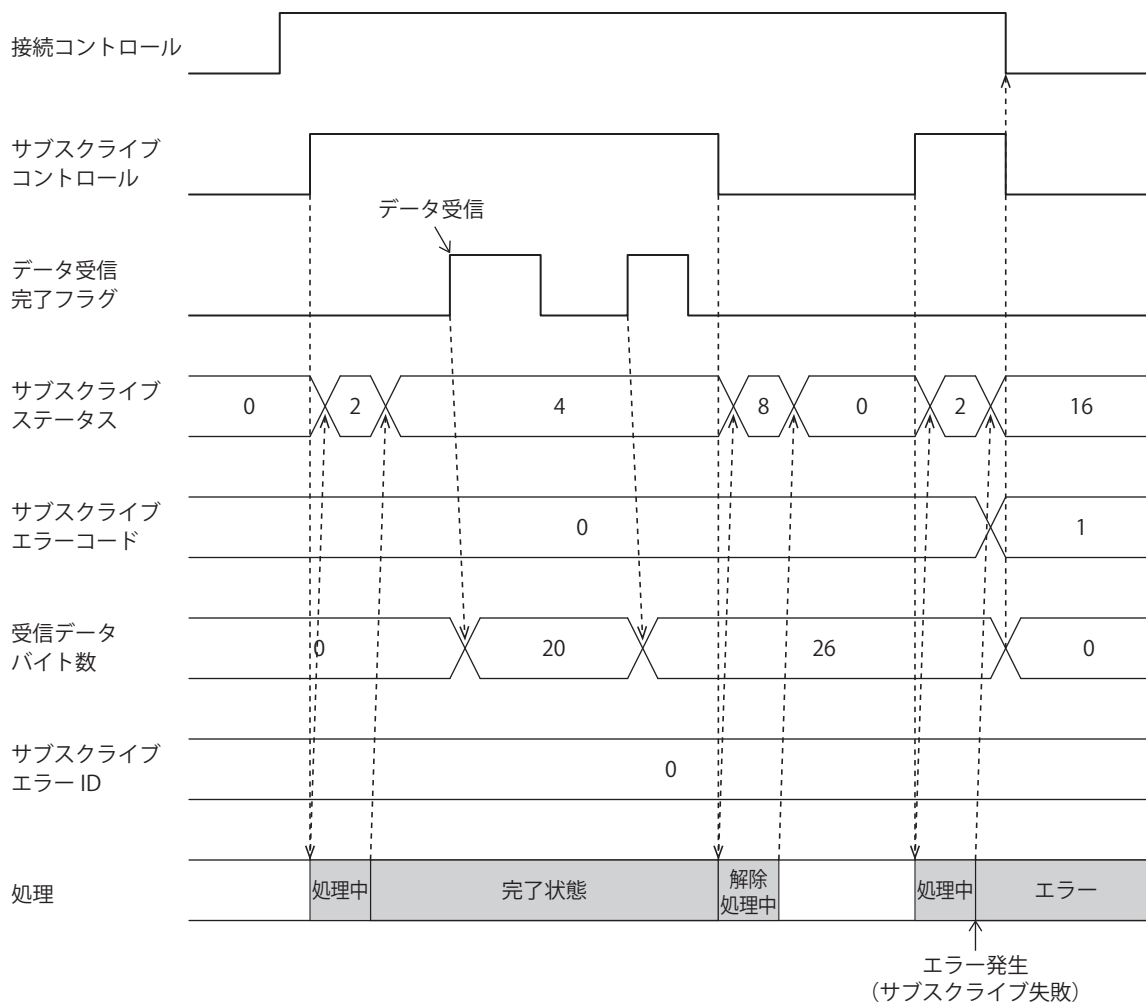
ラダー行	説明												
1	M0000をONすると、ブローカーへの接続を開始します。M0010がONのとき、ブローカーに接続した状態です。												
2	ブローカーへ接続できなかったとき、M0011がONします。												
3	M0010がONのとき（ブローカーへ接続した状態のとき）、M0020をOFF→ONすると、D1000に0 (0000h)を格納して、パブリッシュステータスをクリアします。												
4	M0100がONして、1800秒（30分）毎にトピックにパブリッシュします。												
5	M0020をOFFすると、パブリッシュを終了します。												
5	<p>パブリッシュが成功したとき、M0021がONします。 下表のとき、(D0500,D0501)=20.5、(D0502,D0503)=60.5、(D0504,D0505)=410.1、(D0600,D0601)=26.1、(D0602,D0603)=64.5、(D0604,D0605)=420.2となります。</p> <table border="1" data-bbox="416 544 1214 698"> <thead> <tr> <th>項目</th> <th>ビニールハウス内（北側）</th> <th>ビニールハウス内（南側）</th> </tr> </thead> <tbody> <tr> <td>温度</td> <td>20.5 °C</td> <td>26.1 °C</td> </tr> <tr> <td>湿度</td> <td>60.5 %</td> <td>64.5 %</td> </tr> <tr> <td>CO₂ 濃度</td> <td>410.1 ppm</td> <td>420.2 ppm</td> </tr> </tbody> </table> <p>さらに内蔵時計の現在時刻のデータ（D8008～D8014）が2020年11月9日10時42分8秒およびD8413（タイムゾーンオフセット）の値が0である場合、送信されるJSON形式のデータは次のとおりです。</p> <pre>{ "Plastic greenhouse (North)":{ "Temperature (°C)": 18.5, "Humidity (%)": 55.5, "CO2 Level (ppm)": 410.1 }, "Plastic greenhouse (South)":{ "Temperature (°C)": 26.1, "Humidity (%)": 64.5, "CO2 Level (ppm)": 420.2 }, "Timestamp": "2020-11-09T10:42:08Z" }</pre> <p>このときD1002（送信データバイト数）には236が格納されます。</p>	項目	ビニールハウス内（北側）	ビニールハウス内（南側）	温度	20.5 °C	26.1 °C	湿度	60.5 %	64.5 %	CO ₂ 濃度	410.1 ppm	420.2 ppm
項目	ビニールハウス内（北側）	ビニールハウス内（南側）											
温度	20.5 °C	26.1 °C											
湿度	60.5 %	64.5 %											
CO ₂ 濃度	410.1 ppm	420.2 ppm											
12	パブリッシュが失敗したとき、N0022がONします。												

トピックをサブスクライブする

サブスクライブおよび解除

Plus CPU モジュールをブローカーに接続した後、サブスクライブコントロールを OFF → ON すると、指定したトピックをサブスクライブし、サブスクライブステータスにステータスを格納します。

サブスクライブが成功すると、ステータス 4 (0004h) が格納されます。この状態で、ブローカーから指定したトピックがパブリッシュされると、データ受信完了フラグが ON し、受信データバイト数に受信したデータの長さがバイト単位で格納されます。受信したデータがどのように処理されるかについて、「受信した JSON 形式のデータの内容をデバイスに格納する」(17-62 頁) を参照してください。



動作例

動作説明

- いちごを栽培しているビニールハウス内の温度、湿度および CO₂ 濃度を管理します。
- 管理者が遠隔からビニールハウス内の温度、湿度の上下制限値、または CO₂ 濃度の下限値をトピック "plantation/house" にパブリッシュします。
- Plus CPU モジュールはトピック "plantation/house" をサブスクライブして、管理者がパブリッシュした設定で動作します。

設定

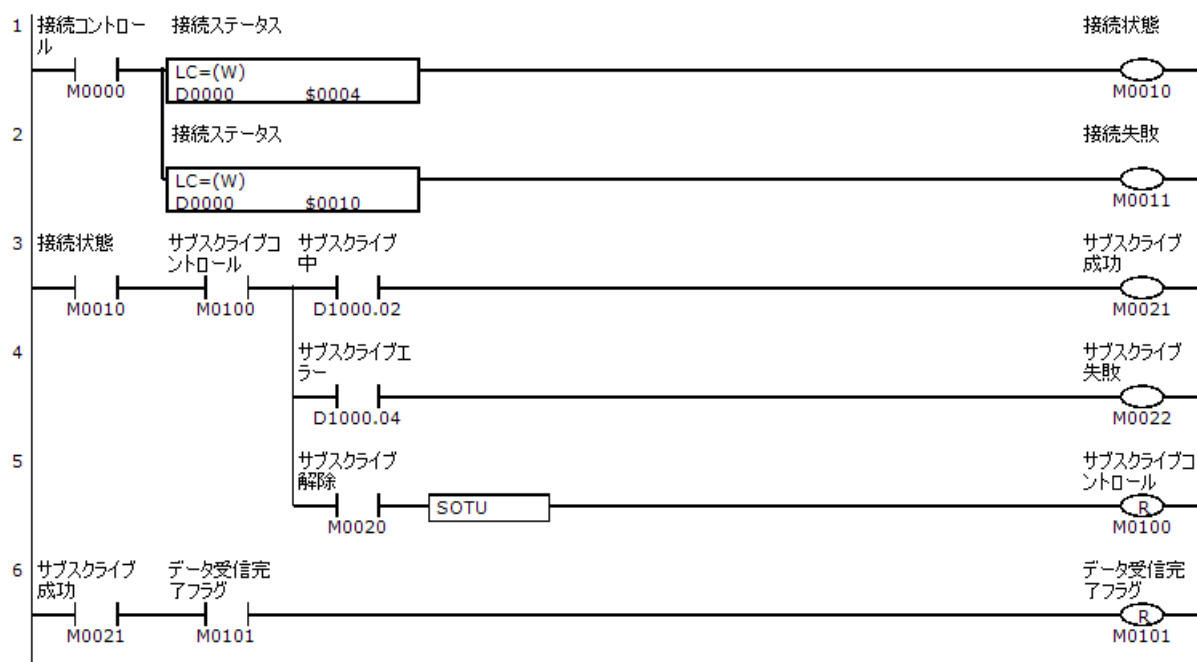
[MQTT 設定] ダイアログボックスの [MQTT 設定] タブで基本設定を行った後、各タブで次のように項目を設定します。

タブ	項目	内容
[MQTT設定] タブ	接続コントロール	M0000
	接続ステータス	D0000
[サブスクライブ] タブ	トピック	"plantation/house"
	[パイロード] ダイアログボックスの設定	下図参照
	サブスクライブコントロール	M0100
	サブスクライブステータス	D1000
	QoS	1
	自動ON	有効

[パイロード] ダイアログボックスの設定



ラダープログラム



デバイスアドレス	コメント
M0000	接続コントロール
M0010	接続状態
M0011	接続失敗
M0020	サブスクライブ解除
M0021	サブスクライブ成功
M0022	サブスクライブ失敗
M0100	サブスクライブコントロール
M0101	データ受信完了フラグ

デバイスアドレス	コメント
D0000	接続ステータス
D0510	温度データ (北) 下限値
D0512	温度データ (北) 上限値
D0514	湿度データ (北) 下限値
D0516	湿度データ (北) 上限値
D0518	CO2濃度データ (北) 下限値
D0610	温度データ (南) 下限値
D0612	温度データ (南) 上限値
D0614	湿度データ (南) 下限値
D0616	湿度データ (南) 上限値
D0618	CO2濃度データ (南) 下限値
D1000	サブスクライブステータス

ラダー行	説明
1	M0000をONすると、ブローカーへの接続を開始します。M0010がONのとき、ブローカーに接続した状態です。
2	ブローカーへ接続できなかったとき、M0011がONします。
3	M0010がONすると (ブローカーへ接続すると)、M0100が自動的にONして、トピックをサブスクライブします。サブスクライブが成功したとき、M0021がONします。
4	サブスクライブが失敗したとき、M0022がONします。
5	M0020をONすると、サブスクライブを解除します。
6	データを受信すると、M0101をOFFします。

受信した JSON 形式のデータの内容をデバイスに格納する

受信した JSON 形式のデータと [ペイロード] ダイアログボックスで設定した ID を先頭から順に検索します。

受信した JSON 形式のデータ内に、[ペイロード] ダイアログボックスで設定した ID の深さレベル、名前およびフォーマットと一致する要素が存在し、その要素の値がその ID のデータタイプの範囲内である場合、その値をデバイスに格納します。



- 受信した JSON 形式のデータ内に、[ペイロード] ダイアログボックスで設定した ID の深さレベル、名前およびフォーマットと一致する要素が存在しない場合、サブスクリプションエラー ID に該当の ID が格納されます。
- 受信した JSON 形式のデータ内に、[ペイロード] ダイアログボックスで設定した ID の深さレベル、名前およびフォーマットと一致する要素が存在し、その要素の値がその ID のデータタイプの範囲外である場合、サブスクリプションエラーコードにエラーコード 7 (0007h) が格納され、サブスクリプションエラー ID にエラーが発生した ID が格納されます。

例

- Plus CPU モジュールは工場のライン 2 本を制御しています。
- 設定値を受信したら、それに応じて処理を行います。設定値は複数ありますが、実際に受信するのは変更がある設定値のみです。

[ペイロード] ダイアログボックスで下図のように設定した場合について説明します。

[ペイロード] ダイアログボックス

ID	名前	フォーマット	データタイプ	データ
① 1	■ (root)	オブジェクト (2)		
② 2	▢ production line1	オブジェクト (1)		
3	▢ settings	オブジェクト (3)		
4	production count	値	ワード (W)	D0100
5	conveyor1 speed	値	ワード (W)	D0101
6	conveyor2 speed	値	ワード (W)	D0102
③ 7	▢ production line2	オブジェクト (1)		
④ 8	▢ settings	オブジェクト (3)		
⑤ 9	production count	値	ワード (W)	D0200
⑥ 10	conveyor1 speed	値	ワード (W)	D0201
⑦ 11	conveyor2 speed	値	ワード (W)	D0202

受信した JSON 形式のデータ

行番号	内容
1	{
2	"production line2" :{
3	"settings":{
4	"conveyor1 speed":50
5	}
6	}
7	}

処理の流れ

- 受信した JSON 形式のデータの 1 行目から 7 行目が中括弧で囲われています。これは [ペイロード] ダイアログボックスの ID1 の設定と一致しますので、次の ID2 を検索します。
- ID2 は受信した JSON 形式のデータに存在しないので、ID2 およびその子要素の検索をスキップし、サブスクリプションエラー ID に 2 (0002h) を格納します。
- 受信した JSON 形式のデータの 2 行目から 6 行目が中括弧で囲われており、"production line2" という名前が設定されています。これは ID7 の設定と一致しますので、次の ID8 を検索します。
- 受信した JSON 形式のデータの 3 行目から 5 行目が中括弧で囲われており、"settings" という名前が設定されています。これは ID8 の設定と一致しますので、次の ID9 を検索します。
- ID9 は受信した JSON 形式のデータに存在しないのでスキップし、次の ID10 を検索します。
- 受信した JSON 形式のデータの 4 行目の名前 "conveyor1 speed" は ID10 の名前と一致し、かつ、この要素の値 (50) はワード (W) の範囲内のため、D0201 に 50 が格納されます。
- ID11 は受信した JSON 形式のデータに存在しないのでスキップし、サブスクリプションエラーコードに 0 (0000h) を格納します。

第18章 MCプロトコル通信

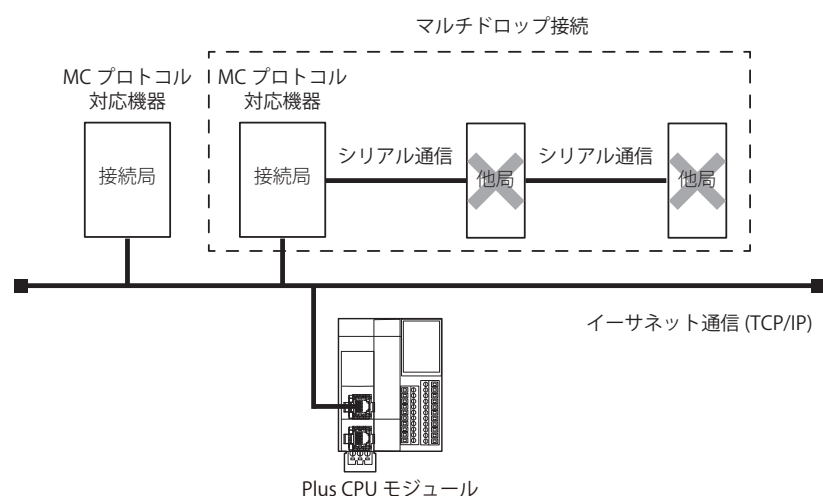
この章では、FC6A形のMCプロトコル通信機能について説明します。

概要

MCプロトコルは、MELSECコミュニケーションプロトコルの略称で、三菱電機株式会社製PLCをはじめとしたMCプロトコル対応機器（以降、MCプロトコル対応機器）と通信するためのプロトコルです。Plus CPU モジュールは、MCプロトコル通信のクライアントとして、MCプロトコル対応機器のデバイス値の読み出しおよび書き込みをすることができます。MCプロトコルについては、MELSECコミュニケーションプロトコルリファレンスマニュアルを参照してください。

アクセス範囲

MCプロトコル通信のネットワーク構成を以下に示します。Plus CPU モジュールはQnA 互換 3E フレームに対応したMCプロトコル対応機器（接続局）と通信できます。Plus CPU モジュールは、接続局に接続（マルチドロップ接続）された他のMCプロトコル対応機器（他局）と通信できません。



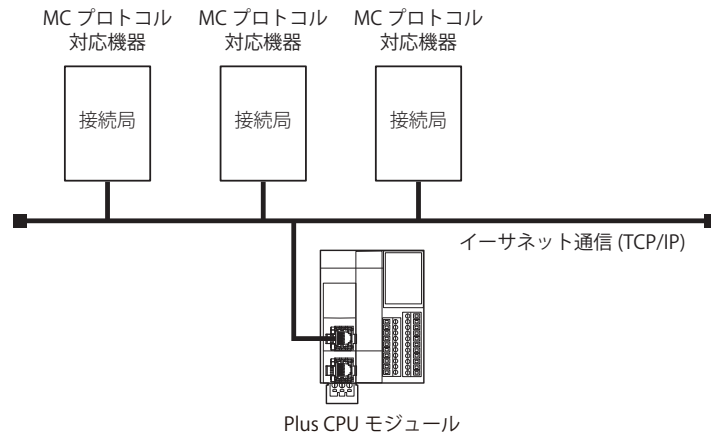
基本仕様

モジュール	Plus CPU モジュール
通信ポート	Ethernet ポート1、Ethernet ポート2
対応フレーム	QnA互換3Eフレーム
対応コマンドおよびサブコマンド	ワード単位の読み出し(コマンド: 0401、サブコマンド: 0000) ビット単位の読み出し(コマンド: 0401、サブコマンド: 0001) ワード単位の書き込み(コマンド: 1401、サブコマンド: 0000) ビット単位の書き込み(コマンド: 1401、サブコマンド: 0001)
対応コード	バイナリコード
接続数	最大16個 ^{*1}

*1 コネクションに最大 255 個のサーバーを接続できます。

イーサネット通信による MC プロトコル通信

Plus CPU モジュールは、Ethernet ポート 1 および Ethernet ポート 2 で、MC プロトコル通信のクライアントに対応しており、最大 16 個のコネクションを MC プロトコル通信のクライアントに割り当てることができます。各コネクションは、それぞれ複数台 (1 ~ 255 台) の MC プロトコル対応機器と通信できます。



Plus CPU モジュールは QnA 互換 3E フレーム (バイナリコード) に対応した MC プロトコル対応機器と通信できます。

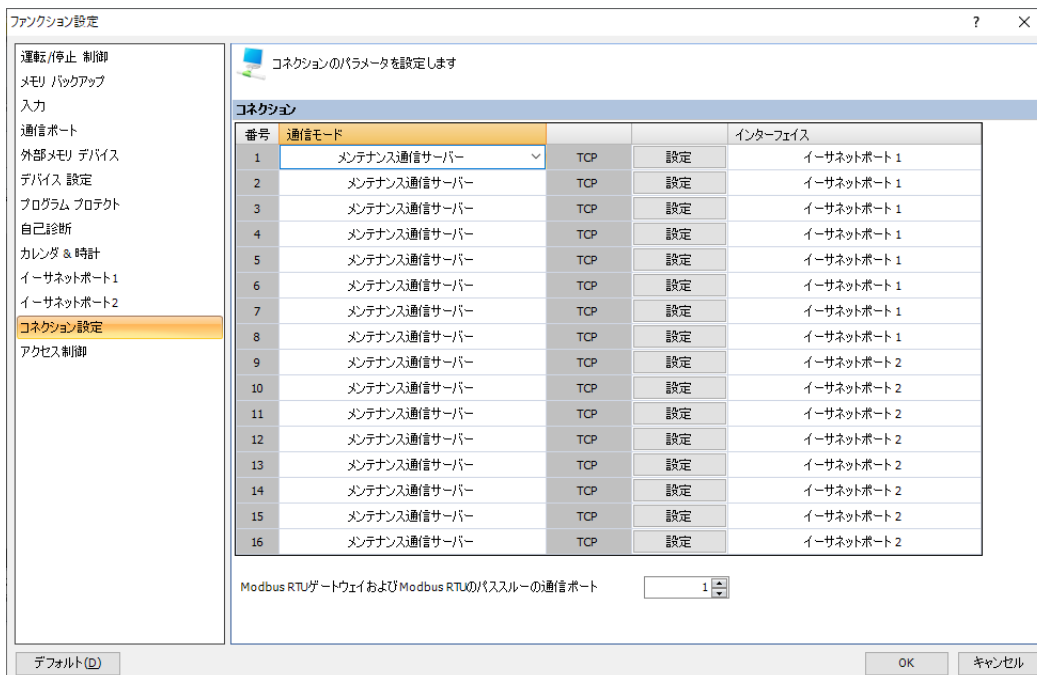
MC プロトコル通信の設定

MC プロトコル通信の設定は [MC プロトコル クライアント] ダイアログボックスで行います。ここでは [MC プロトコル クライアント] ダイアログボックスとその設定について説明します。

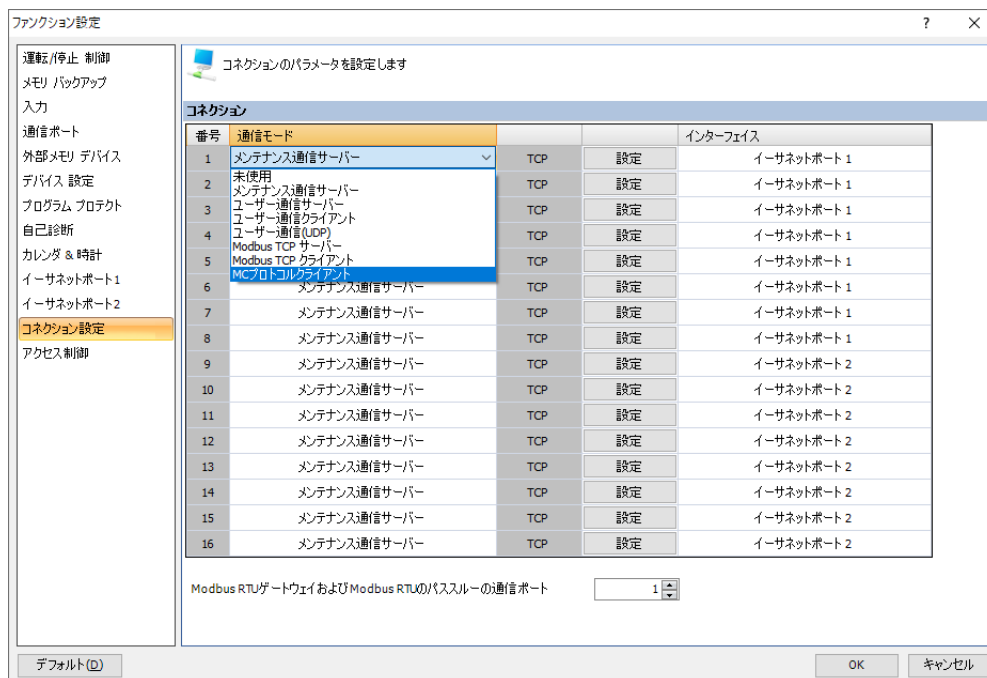
WindLDR の設定

●操作手順

1. [設定] タブの [ファンクション設定] で [コネクション設定] をクリックします。
[ファンクション設定] ダイアログボックスが表示されます。



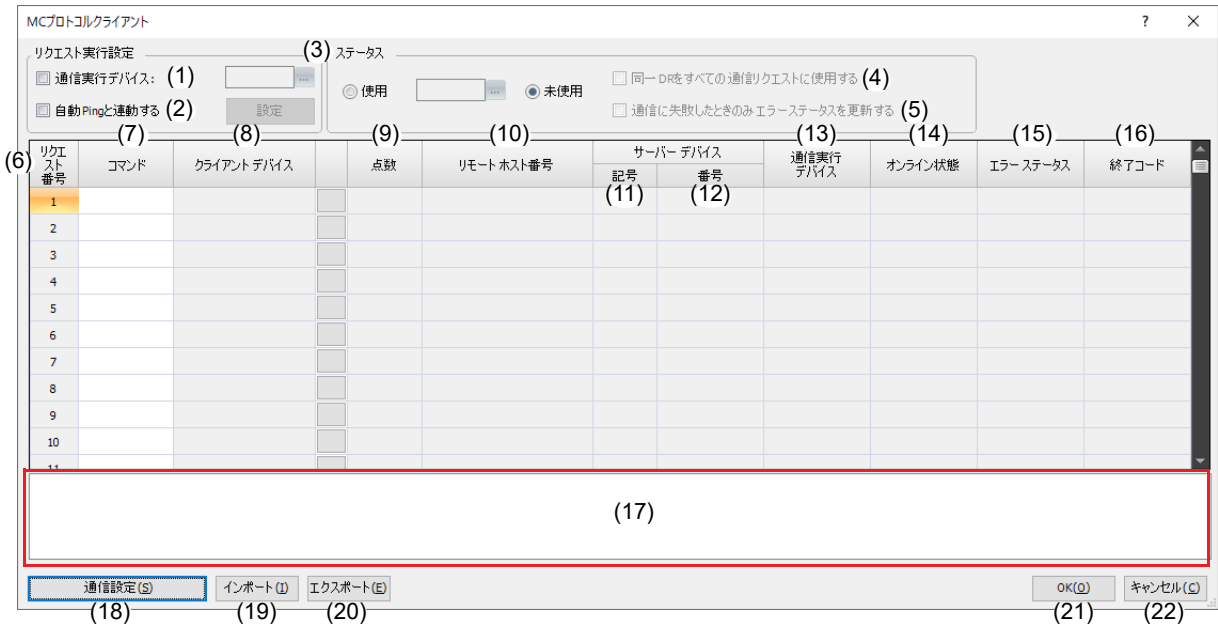
2. 使用する接続番号の [通信モード] をクリックし、"MC プロトコル クライアント" を選択します。




[MC プロトコル クライアント] ダイアログボックスが表示されます。

[MCプロトコルクライアント] ダイアログボックス

[MCプロトコルクライアント] ダイアログボックスでMCプロトコル通信を行うための設定を行います。



 [MCプロトコルクライアント] ダイアログボックス中央の表の部分をリクエストテーブルと呼びます。リクエストテーブル内の各行をリクエストと呼びます。

(1) 通信実行デバイス

Plus CPU モジュールのリクエストの実行を通信実行デバイスの ON/OFF により制御します。


通信実行デバイスが OFF から ON になったとき、「(2) 送信待ち時間」(18-10 頁) で設定した時間が経過した後、Plus CPU モジュールはリクエストを送信します。リクエストを送信した後、応答を受信すると、対応する通信実行デバイスは自動的に OFF になります。


応答を受信できない場合は、「(1) 受信タイムアウト」(18-10 頁) で設定した時間が経過した後、リトライを実施します。リトライは 2 回実施します。その後、対応する通信実行デバイスは自動的に OFF になります。この一連の処理をリクエストの実行と言います。また、正常応答を受信した場合をリクエストの実行が成功したと言います。異常応答を受信した場合、またはリトライしても応答を受信できなかった場合をリクエストの実行が失敗したと言います。

通信実行デバイスが OFF のとき、Plus CPU モジュールはリクエストを送信しません。

通信実行デバイスには、内部リレーまたはデータレジスタを設定できます。設定されたデバイスを先頭に、リクエストテーブルに登録されているリクエストに対して、リクエスト番号順にデバイスが割り付けられます。通信実行デバイスにデータレジスタを設定した場合、通信実行デバイスはデータレジスタの最下位ビットを先頭にして、リクエスト番号順に割り付けられます。

通信実行デバイス	説明
未使用 (チェックボックスがオフの場合)	リクエストテーブルに登録されているリクエストをリクエスト番号順に繰り返し実行します。
使用 (チェックボックスがオンの場合)	リクエストテーブルに登録されているリクエストに対して、通信実行デバイスに設定したデバイスを先頭に、リクエスト番号順に通信実行デバイスが割り付きます。実行したいリクエストの通信実行デバイスを ON すると、リクエストを実行します。

 通信実行デバイスが自動的に OFF になるまで、読み出すデバイス値および書き込むデバイス値を変更しないでください。変更前のデバイス値を読み書きできない可能性があります。

 複数の通信実行デバイスを同時に OFF から ON にした場合、直前に実行が完了したリクエスト番号以降で一番近いリクエスト番号のリクエストから順に 1 個ずつ処理されます。実行が完了 (成功または失敗) したときに次のリクエストの処理を行います。

(2) 自動 Ping と連動する

リクエストを実行するかどうかを、自動 Ping の結果により制御します。本機能は、あるリモートホストが故障などにより応答しない状況でも他のリモートホストとの MC プロトコル通信を最小限の遅延で実行したい場合に便利です。

自動 Ping と連動する	説明
自動Pingと連動しない (チェックボックスがオフの場合)	リクエストテーブルに登録されているリクエストをリクエスト番号順に繰り返し実行します。
自動Pingと連動する (チェックボックスがオンの場合)	<ul style="list-style-type: none"> 自動 Ping の送信対象であるリモートホスト： オンライン状態（送信したPingの結果を格納する内部リレー）がONの場合、該当リクエストを実行します。オンライン状態がOFFの場合は該当リクエストを実行しません。 自動 Ping の送信対象でないリモートホスト： オンライン状態の内部リレーの状態に関わらず、該当リクエストを実行します。

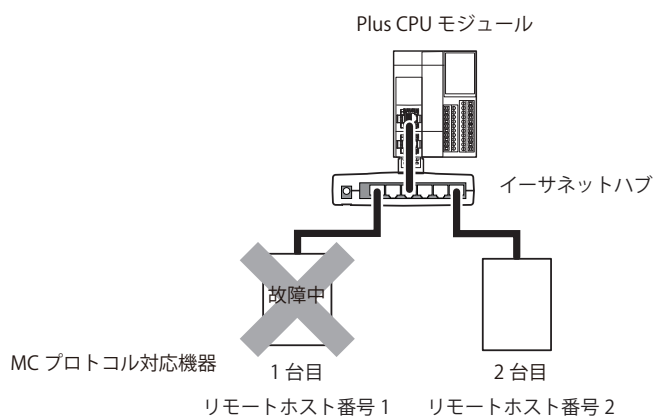
MC プロトコル通信を行う場合、通信相手の MC プロトコル対応機器をリモートホストとしてリモートホストリストに登録する必要があります。自動 Ping と連動するのチェックボックスがオンの場合、[設定] ボタンが表示されます。[設定] ボタンをクリックすると、[リモートホストリスト] ダイアログボックスが表示されますので、リモートホストを登録してください。リモートホストリストについては、「リモートホストリスト」(3-21 頁)を参照してください。

自動 Ping と連動しない場合、および自動 Ping と連動する場合のリクエスト処理について説明します。

システム構成例

設定内容

通信実行デバイス：未使用

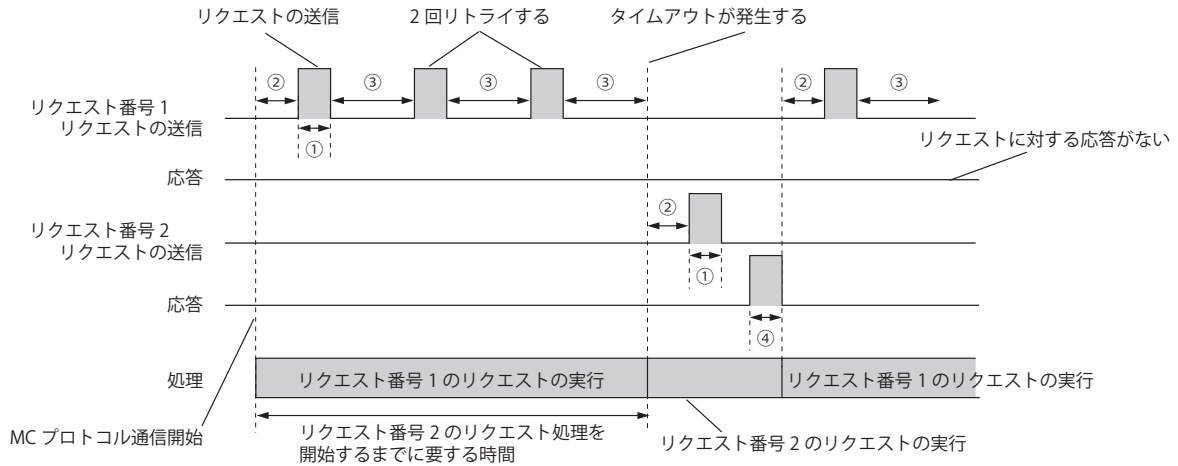


■自動 Ping と連動しない（[自動 Ping と連動する] チェックボックスがオフの場合）

リクエスト番号順にリクエストを実行します。

リクエスト番号	リモートホスト番号	リクエストの実行
1	1	実行する
2	2	実行する

リクエスト処理のタイミングチャート



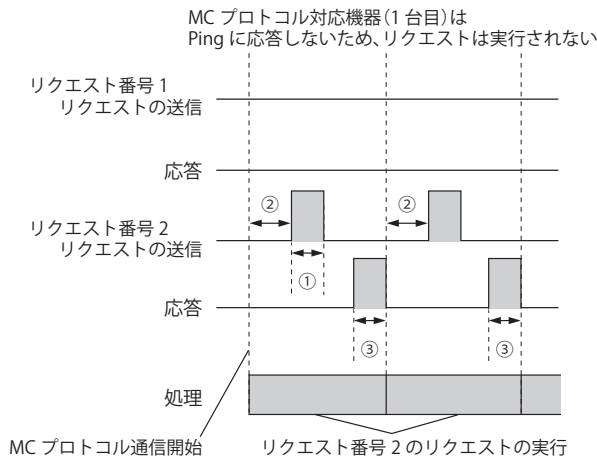
- ①： リクエストの送信
- ②： リクエストの実行開始からリクエストを送信するまでの送信待ち時間
- ③： [受信タイムアウト] で設定した時間が経過した後、リトライする
- ④： 応答

■自動 Ping と連動する ([自動 Ping と連動する] チェックボックスがオンの場合)
 オンライン状態のリモートホストへのみリクエストを実行します。

リクエスト番号	リモートホスト番号	オンライン状態*1	リクエストの実行
1	1	OFF	実行する
2	2	ON	実行する

*1 オンライン状態とは、送信した Ping の結果を格納する内部リレーのことです。オンライン状態の詳細は、「本章 (14) オンライン状態」を参照してください。

リクエスト処理のタイミングチャート



- ①： リクエストの送信
- ②： リクエストの実行開始からリクエストを送信するまでの送信待ち時間
- ③： 応答



本機能は、あるリモートホストが故障などにより応答しない状況でも他のリモートホストとの MC プロトコル通信を最小限の遅延で実行したい場合に有効です。応答しないリモートホストが設定されたリクエストの実行を完了するのに要する時間は、[通信設定] ダイアログボックスの [受信タイムアウト] で設定した時間の 3 倍です。自動 Ping 連動については、「自動 Ping 機能」(3-23 頁) を参照してください。

(3) ステータス

リクエストを実行した際のエラーステータスおよび終了コードをデバイスに格納するかどうかを設定します。エラーステータスおよび終了コードをデバイスに格納する場合、[使用] を選択し、データレジスタを設定します。1つのリクエストあたり2ワードのデータレジスタを使用します。



リクエストを最大数（255個）作成した場合、下表のようにデータレジスタが割り付きます。

デバイスアドレス	内容		R/W
先頭番号+0	リクエスト番号1のステータス	エラーステータス	R
先頭番号+1		終了コード	R
先頭番号+2	リクエスト番号2のステータス	エラーステータス	R
先頭番号+3		終了コード	R
...
先頭番号+508	リクエスト番号255のステータス	エラーステータス	R
先頭番号+509		終了コード	R

エラーステータス

上位バイトにリモートホスト番号、下位バイトにエラーコードが格納されます。

エラーステータス（16進数）	
上位バイト	下位バイト
リモートホスト番号	エラーコード

エラーコードの詳細は、次のとおりです。

エラーコード（16進数）	内容	詳細
0x04	異常終了	MCプロトコル対応機器から受信したデータの終了コードが0x0000（正常終了）でない
0x16	タイムアウトエラー	MCプロトコル対応機器からの応答がなかった

終了コード

MCプロトコル通信の終了コードを格納します。終了コードについては、通信相手のMCプロトコル対応機器のマニュアルを参照してください。



通信中に、イーサネットケーブルが抜けたり、MCプロトコル対応機器の電源がOFFになるなど、異常が発生した場合、タイムアウトエラーが発生し、エラーコードは0x16となります。このとき、「(8) クライアントデバイス」(18-8頁) 設定したデバイスには異常が発生する直前にMCプロトコル対応機器から読み出したデータが保存されています。このデバイスのデータを0クリアしたい場合は、0を書き込んでください。

(4) 同一DRをすべての通信リクエストに使用する

[ステータス] が "使用" の場合、エラーステータスおよび終了コードを格納するためのデータレジスタを設定します。詳細は次のとおりです。

同一DRをすべての通信リクエストに使用する	説明
使用しない (チェックボックスがオフの場合)	各リクエストのエラーステータスおよび終了コードを格納します。 [ステータス] で設定したデータレジスタを先頭に、1つのリクエストあたり2ワード（最大510ワード）のデータレジスタを使用します。
使用する (チェックボックスがオンの場合)	最後に実行したリクエストのエラーステータスおよび終了コードを格納します。 [ステータス] で設定したデータレジスタを先頭に、2ワードのデータレジスタを使用します。

(5) 通信に失敗したときのみステータスを更新する

エラーステータスおよび終了コードを更新するタイミングを設定します。

[通信に失敗したときのみエラーステータスを更新する] チェックボックスがオンの場合、リクエストの実行に失敗したときのみ、該当のリクエストのステータス（エラーステータスおよび終了コード）を更新します。このチェックボックスがオフの場合、すべてのリクエストのステータスを更新します。

- (6) リクエスト番号
各リクエストの番号 (1 ~ 255) です。



通信実行デバイスやステータスを使用する場合、設定したデバイスを先頭にして、デバイスがリクエスト番号順に自動的に割り付きます。リクエストテーブル途中のリクエストを削除したり、リクエストの順番を入れ替えたりした場合、通信実行デバイスやエラーステータスの割り付けが更新され、ユーザープログラムで使用しているデバイスと不整合が発生する可能性があります。リクエストテーブルを変更した場合、ユーザープログラムを十分に確認してください。

- (7) コマンド
Plus CPU モジュールが MC プロトコル対応機器に送信するコマンドを次の中から選択します。

コマンド	内容
04010000 ワード単位の読出し	[記号] および [番号] で指定したデバイスから [点数] で指定した点数分のワードデバイスの値を読み出し、[クライアント デバイス] から点数分に格納します。
04010001 ビット単位の読出し	[記号] および [番号] で指定したデバイスから [点数] で指定した点数分のビットデバイスの値を読み出し、[クライアント デバイス] から点数分に格納します。
14010000 ワード単位の書込み	[クライアント デバイス] から [点数] で指定した点数分のワードデバイスの値を、[記号] および [番号] で指定したデバイスから [点数] で指定した点数分のワードデバイスに書き込みます。
14010001 ビット単位の書込み	[クライアント デバイス] から [点数] で指定した点数分のビットデバイスの値を、[記号] および [番号] で指定したデバイスから [点数] で指定した点数分のビットデバイスに書き込みます。

- (8) クライアント デバイス
Plus CPU モジュールのデバイスを設定します。[コマンド] の設定によって、設定可能なデバイスタイプが異なります。詳細は次のとおりです。

コマンド	入力 (I)	出力 (Q)	内部リレー (M)	特殊内部リレー (M)	データレジスタ (D)	特殊データレジスタ (D)
04010000 ワード単位の読出し	×	×	×	×	○	×
04010001 ビット単位の読出し	×	×	○	×	×	×
14010000 ワード単位の書込み	○	○	○	×	○	×
14010001 ビット単位の書込み	○	○	○	×	×	×



クライアント デバイスはデバイスの範囲を超えないように設定してください。

- (9) 点数
読出しまたは書込みを行うデータの点数を 1 ~ 512 の範囲で設定します。

(10) リモートホスト番号

Plus CPU モジュールと通信する MC プロトコル対応機器の IP アドレスおよびポート番号を設定します。

ネットワーク上の通信相手の機器（リモートホスト）を登録して管理するリストをリモートホストリストと呼びます。リモートホストリストについては、「リモートホストリスト」（3-21 頁）を参照してください。



・ リクエストの実行順と接続の断続時間

リクエストの実行順によって接続の切断および接続に要する時間が変わることがあります。

リクエストを実行すると、リモートホストと接続を接続してリクエストを送信します。接続は次のリクエストを実行するまで維持されます。次に実行するリクエストのリモートホストと、接続中のリモートホストが同じ場合、接続を維持した状態でリクエストを送信します。リモートホストが異なる場合、接続を切断し、次に実行するリクエストのリモートホストと接続を接続してリクエストを送信します。同じリモートホストが設定されたリクエストを順に実行することで、接続の切断および接続に要する時間を省くことができます。

たとえば、通信実行デバイスを使用せずに、下表のようなリモートホストテーブル①、②の場合、リモートホストテーブル①は接続の切断および接続に要する時間を省くことができます。

リモートホストテーブル①	
リクエスト番号	リモートホスト番号
1	1
2	1
3	2
4	2



リクエスト番号 1 および 3 を実行したときのみ、既存の接続を切断し、新しい接続を接続します。



接続の切断および接続に要する時間を省くことができます。

リモートホストテーブル②	
リクエスト番号	リモートホスト番号
1	1
2	2
3	1
4	2



各リクエスト番号のリクエストを実行するたびに、既存の接続を切断し、新しい接続を接続します。



接続の切断および接続に要する時間がかかります。

(11) サーバー デバイスの記号

(12) サーバー デバイスの番号

MC プロトコル対応機器のデバイス記号および番号を設定します。

■ 記号

[コマンド] の設定によって、設定可能なデバイスが異なります。詳細は次のとおりです

コマンド	入力 (X)	出力 (Y)	内部リレー (M)	特殊内部リレー (SM)	データレジスタ (D)	特殊データレジスタ (SD)
04010000 ワード単位の読出し	○	○	○	○	○	○
04010001 ビット単位の読出し	○	○	○	○	×	×
14010000 ワード単位の書込み	×	×	×	×	○	×
14010001 ビット単位の書込み	○	○	○	×	×	×

■ 番号

[記号] の設定によって、サーバー デバイス番号の設定可能な範囲が異なります。詳細は次のとおりです。

記号	番号
入力 (X)	0000 ~ FFFFFFF*1
出力 (Y)	0000 ~ FFFFFFF*1
内部リレー (M)	0 ~ 16777215
特殊リレー (SM)	0 ~ 16777215
内部レジスタ (D)	0 ~ 16777215
特殊レジスタ (SD)	0 ~ 16777215

*1 16 進数で設定します。



設定可能な番号は、MCプロトコル対応機器によって異なります。MCプロトコル対応機器のマニュアルをご確認の上、デバイスの範囲を超えないように [番号] を設定してください。

(13) 通信実行デバイス

各リクエスト番号に割り当てられた通信実行デバイスが表示されます。これは [通信実行デバイス] のチェックボックスがオンの場合のみ行われます。



リクエストテーブル内のリクエストを削除したり、リクエストの順番を入れ替えたりして、既存のリクエストのリクエスト番号が変更されると、そのリクエストの通信実行デバイスも変更されます。このような場合、ユーザープログラムで使用するデバイスと不整合が発生する可能性がありますので、ユーザープログラムを十分に確認してください。

(14) オンライン状態

[自動 Ping と連動する] のチェックボックスがオンの場合、[自動 Ping 設定] ダイアログボックスで各リモートホストの [オンライン状態] に設定された内部リレーを表示します。[自動 Ping 設定] ダイアログボックスについては、「自動 Ping 機能」(3-23 頁) を参照してください。

(15) エラーステータス**(16) 終了コード**

各リクエスト番号に割り当てられたエラーステータスおよび終了コードを格納するデータレジスタが表示されます。これは [ステータス] で [使用] を選択している場合のみ行われます。



[同一 DR をすべての通信リクエストに使用する] のチェックボックスがオフの場合、リクエストテーブル内のリクエストを削除したり、リクエストの順番を入れ替えたりして、既存のリクエストのリクエスト番号が変更されると、そのリクエストのエラーステータスおよび終了コードの格納先も変更されます。このような場合、ユーザープログラムで使用するデバイスと不整合が発生する可能性がありますので、ユーザープログラムを十分に確認してください。

(17) エラー情報表示エリア

[MCプロトコルクライアント] ダイアログボックスで設定した内容にエラーがある場合、エラーの内容が表示されます。

(18) [通信設定] ボタン

[通信設定] ボタンをクリックすると [通信設定] ダイアログボックスが開きます。[通信設定] ダイアログボックスについては、「[通信設定] ダイアログボックス」(18-10 頁) を参照してください。

(19) [インポート] ボタン

リクエストテーブルの内容をインポートします。

(20) [エクスポート] ボタン

リクエストテーブルの内容をエクスポートします。
リクエストテーブルの内容を一括で編集したり、他のプロジェクトにインポートしたりできます。

(21) [OK] ボタン

設定を保存し、[MCプロトコルクライアント] ダイアログボックスを閉じます。

(22) [キャンセル] ボタン

設定を保存せずに [MCプロトコルクライアント] ダイアログボックスを閉じます。

[通信設定] ダイアログボックス

このダイアログボックスで MC プロトコル通信の詳細設定を行います。

(1) 受信タイムアウト

Plus CPU モジュールが MC プロトコル対応機器にデータを送信した後、MC プロトコル対応機器から応答が返るまでの待ち時間を設定します。300ms ~ 25500ms (100ms 間隔) で設定します。

(2) 送信待ち時間

リクエストを実行した後、次のリクエストを実行するまでの待ち時間を 0ms ~ 5000ms の範囲で設定します。

MC プロトコル クライアントの動作

リクエストの処理と通信は独立しており、リクエストテーブルの各リクエストは、リクエスト番号順に END 処理で繰り返し実行されます。通信は、ユーザープログラムの実行とは非同期で行われます。リクエストが実行されるかどうかは、以下の設定の組み合わせで決まります。詳細は次のとおりです。

設定		[自動 Ping と連動する]	
		未使用	使用
			自動 Ping の送信対象
[通信実行デバイス]	未使用	リクエストを実行します。	リクエストの [オンライン状態] が、(1)ON のとき、そのリクエストを実行します。 (2)OFF のとき、そのリクエストを実行しません。
	使用	リクエストの [通信実行デバイス] が、(1)ON のとき、そのリクエストを実行します。 (2)OFF のとき、そのリクエストを実行しません。	リクエストの [オンライン状態] および [通信実行デバイス] が ON のとき、リクエストを実行します。それ以外のときはリクエストを実行しません。

リクエストの実行が完了すれば、次のリクエスト番号のリクエストを実行します。リクエストは次のいずれかの場合に実行完了となります。

- ・ [通信実行デバイス] を使用している場合、リクエストを実行するために ON した通信実行デバイスが自動的に OFF になったとき
- ・ [ステータス] を使用している場合、エラーステータスを格納するデータレジスタの値が更新されたとき

リクエストの実行が成功した（正常応答を受信した）場合

リクエストのエラーコード (0x00) および終了コード (0x0000) が書き込まれ、[送信待ち時間] で設定した時間が経過した後、次のリクエストが実行されます。エラーコードと終了コードの書き込みは、[ステータス] の [使用] を選択し、かつ [通信に失敗したときのみステータスを更新する] のチェックボックスがオフの場合のみ行われます。

リクエストの実行が失敗した場合

リクエストのエラーコードおよび終了コードに 0x0000 以外の値が書き込まれます。エラーコードと終了コードの書き込みは、[ステータス] の [使用] を選択している場合のみ行われます。次の場合に、[送信待ち時間] で設定した時間が経過した後、次のリクエストを実行します。

- ・ 異常応答を受信した場合
- ・ 応答（正常応答または異常応答）を受信できず、2 回リトライするもすべてのリトライに失敗した場合



- ・ 通信実行デバイスを使用する場合、リクエストを実行した後、対応する通信実行デバイスは自動的に OFF になります。リクエストを常時送信したい場合は、該当する通信実行デバイスを、ユーザープログラムの OUT 命令で常時 ON してください。
- ・ 通信実行デバイスはエラー発生時、2 回のリトライに失敗した後に自動的に OFF になります。
- ・ リクエストに設定されたリモートホスト番号（以下、ホスト番号①）が、1 つ前に実行したリクエストに設定されたリモートホスト番号（以下、ホスト番号②）と異なる場合、ホスト番号②のリモートホストとの接続を切断して、次のリクエストを実行します。リクエストに設定されたホスト番号①が、1 つ前に実行したリクエストに設定されたホスト番号②と同じ場合、その接続を維持して、次のリクエストを実行します。

使用する特殊デバイス

MC プロトコル通信で使用する特殊デバイスについて示します。

特殊データレジスタ

アドレス	機能	説明	R/W
D8278	通信モード情報（コネクション 1～4）	コネクション番号 1～4 の通信モードを示します。	R
D8279	通信モード情報（コネクション 5～8）	コネクション番号 5～8 の通信モードを示します。	R
D8760	通信モード情報（コネクション 9～12）	コネクション番号 9～12 の通信モードを示します。	R
D8761	通信モード情報（コネクション 13～16）	コネクション番号 13～16 の通信モードを示します。	R

デバイス内の割り当て（ビットアサイン）については、「特殊データレジスタ」（2-18 頁）を参照ください。

索引

A

Analog Inputオブジェクト	
Present_Value 設定	15-21
オブジェクト	15-34
プロパティ	15-45
Analog Outputオブジェクト	
Present_Value 設定	15-23
オブジェクト	15-35
プロパティ	15-46
Analog Valueオブジェクト	
Present_Value 設定	15-25
オブジェクト	15-36
プロパティ	15-47
Application_Software_Version	15-56
Assemblyオブジェクト	16-43

B

BACnet/IP	15-1
BACnet/IPの設定	15-13
BACnet/IPの動作	15-11
BACnet通信	1-12
BCC (ブロック・チェック・キャラクタ)	
受信命令	5-12
送信命令	5-6
BCCの比較	5-21
BIBB	15-4
Binary Inputオブジェクト	
Present_Value 設定	15-27
オブジェクト	15-37
プロパティ	15-50
Binary Outputオブジェクト	
Present_Value 設定	15-29
オブジェクト	15-38
プロパティ	15-51
Binary Valueオブジェクト	
Present_Value 設定	15-30
オブジェクト	15-39
プロパティ	15-52
Bluetooth通信カートリッジ	1-4
Bluetooth通信カートリッジ設定	9-4
Bluetoothでのメンテナンス通信	4-6, 4-15
Bluetoothデバイス設定	9-7
Bluetoothデバイスリスト	9-6

C

CANポート	1-6
CA指定	8-14
CA非指定	8-22
Connection Managerオブジェクト	16-43
COV_Increment	15-54
CRC	6-14

D

Deviceオブジェクト	
オブジェクト	15-40
DR制御ラインコントロール	5-34
DR制御ライン状態	2-36

E

EDSファイル	16-32
ERXD (イーサネットユーザー通信受信) 命令	5-51
ERXD (イーサネットユーザー通信受信) 命令の受信キャンセルフラグの割り付け	5-43, 5-45, 5-47
ER出力制御ラインコントロール	2-37
ER制御ラインコントロール	5-36
Ethernet Linkオブジェクト	16-45
EtherNet/IP通信	1-13, 16-1
EtherNet/IP通信の設定	16-18
EtherNet/IPの設定例	16-35
Ethernetポート1および2	1-5
Ethernetポートでのメンテナンス通信	4-7
ETXD (イーサネットユーザー通信送信) 命令	5-49
Explicitメッセージ通信 (サーバー) 機能	16-10
Eメールアドレス帳	12-11
Eメールエディタ	12-13

F

Firmware_Revision	15-56
Foreign Device機能	15-10
FTPサーバー設定	10-4

H

HMI-Ethernetポート	1-6
HMI-Ethernetポートでのメンテナンス通信	4-16

I

I/Oメッセージ通信機能	16-5
Identityオブジェクト	16-41
IPアドレス	3-15, 3-22, 4-9

J

J1939通信	1-14, 8-1
J1939通信の設定	8-7
JSONテキストをインポートダイアログボックス	17-47, 17-48

L

LCDモニタページ	13-30
-----------	-------

M

MCプロトコル	18-1
---------	------

MCプロトコル通信	
終了コード	18-7
通信実行デバイス	18-4
通信設定ダイアログボックス	18-10
エラーステータス	18-7
コマンド	18-8
サーバーデバイスの記号	18-9
サーバーデバイスの番号	18-9
リクエスト番号	18-8
リモートホスト番号	18-9
MCプロトコルクライアントダイアログボックス	18-4
Message Routerオブジェクト	16-42
Modbus RTUスレーブ	6-8
Modbus RTUスレーブアドレス	6-8
Modbus RTU通信のフレームフォーマット	6-14
Modbus RTUバススルー機能	6-30
Modbus RTUマスター	
エラーコード一覧	6-5
機能コード	6-6
通信実行デバイス	6-5
リクエスト番号	6-6
Modbus RTUマスターリクエストテーブル	6-5
Modbus TCPクライアントリクエストテーブル	6-22
Modbus TCPサーバー	6-25
Modbus TCPサーバーアドレス	6-25
Modbus TCP通信フォーマット	6-29
Modbus RTUマスター	6-2
Modbus TCPクライアント	6-19
Modbus TCPヘッダ	6-29
Modbus通信	1-11, 6-1, 18-1
MQTT設定	17-6
MQTT設定ダイアログボックス	17-6
MQTT通信	1-13, 17-1
MQTT通信の設定	17-5
N	
NAME	8-2
O	
Object_List	15-56
Out_Of_Service	15-55
P	
PGN	8-2
PGNエディタ	8-31
PGNグループエディタ	8-31
PGNマネージャー	8-29
PING設定	3-13
PING命令 (Ping送信)	11-1
PLCステータス表示ページ	13-27
Polarity	15-54
Present_Value	15-41
Present_Value設定	15-21
Priority_Array	15-54
Protocol_Object_Types_Supported	15-56
Protocol_Services_Supported	15-56

R

Reliability	15-55
Relinquish_Default	15-54
RS232C通信カートリッジ	1-4
RS485通信カートリッジ	1-4
RXD (ユーザー通信受信) 命令	5-28

S

SAE J1939	8-1
SMTP認証 (login)	12-7
SNTP設定	3-11
Status_Flags	15-54
Subscribed COV (COV) 機能	15-7
System_Status	15-56

T

TCP/IP Interfaceオブジェクト	16-44
TXD (ユーザー通信送信) 命令	5-25

U

Unsubscribed COV (COVU) 機能	15-9
USBポート	1-3
USBポートでのメンテナンス通信	4-3

W

WebサーバーのWebページツリー構成	13-1
Webページエディタ	13-43

あ

アクセスIP許可アドレス	6-28
アクセス許可IPアドレス	4-9, 4-18

い

イーサネット通信でのユーザー通信	5-39
イーサネット通信によるModbus TCP通信	6-19
イーサネットユーザー通信命令	5-49
ERXD (イーサネットユーザー通信受信) 命令	5-51
ETXD (イーサネットユーザー通信送信) 命令	5-49
イーサネットユーザー通信受信命令 (ERXD)	5-22
イーサネットユーザー通信送信命令 (ETXD)	5-22
一括ダイアログボックス	17-46
一括モニタページ	13-28

え

エンドデリミタの設定	5-19
------------	------

お

オブジェクト	15-33, 16-41
オブジェクト一覧	15-32
オブジェクト設定	15-20
オブジェクトとデバイスの連動機能	15-5
オプション	13-24
オプション接続情報	2-33

き

機種ID	2-50
機能コード	6-29, 6-31
機能コード1、2	6-15

- 機能コード15 6-17
機能コード16 6-18
機能コード3,4 6-15
機能コード5 6-16
機能コード6 6-16
キャラクタ間タイムアウト時間 6-2, 6-8
キャリア 2-11
キャリア/ボロー 2-11
- ## け
- ケーブル総延長 6-2
- ## こ
- コネクション設定 3-14
コネクションのステータス情報 3-15
- ## さ
- サービス 15-3
最大スレーブ接続台数 6-2
サブスクライブ 17-38
- ## し
- 自局設定 8-11
システムWebページ 13-27
システムソフトウェアのダウンロード 4-1
自動Ping機能 3-23
受信タイムアウト 4-5, 4-9, 4-14, 4-18, 6-2, 6-7, 6-19
シリアル通信 (Bluetooth) でのユーザー通信 5-38
シリアル通信でのユーザー通信 5-23
シリアル通信によるModbus RTU通信 6-2, 18-2
シリアル通信モジュール 1-3
シリアルポート1 1-3
新規値または編集ダイアログボックス 17-44
- ## す
- スキップ 5-14
スタートデリミタの設定 5-17
ステータス 2-51
ストップビット 4-5, 4-14, 6-2, 6-7, 6-8, 6-13
スレーブアドレス 6-6, 6-24
スレーブ番号 4-5, 4-14, 6-2, 6-6, 6-8, 6-13, 6-24
- ## せ
- 制御ラインコントロール 5-32
DR 制御ラインコントロール 5-34
ER 制御ラインコントロール 5-36
制御ライン状態 5-33
制御ライン状態 2-34
設定一覧 3-1
- ## そ
- 送信待ち時間 6-2, 6-7, 6-19
- ## ち
- チェックコード 6-14
- ## つ
- 通信インターフェイス 1-3
- 通信カートリッジ情報 2-33
通信機能 1-1
通信設定 3-1, 6-7
通信速度 4-5, 4-14, 6-2, 6-7, 6-8, 6-13
通信フレームフォーマット 6-15
通信ポートの設定 3-2
通信ポート番号の割り付け 1-8
- ## て
- データ (デバイス) 挿入ダイアログボックス 17-49
データビット長 4-5, 4-14, 6-2, 6-7, 6-8, 6-13
データリンク
仕様 7-1, 8-1
割付 7-2
データリンク通信 1-11, 7-1
定数設定によるデータ照合 5-20
定数データの設定 (受信命令) 5-10
定数データの設定 (送信命令) 5-3
テキストのエンコード 12-10
デバイス 2-1
デバイス一覧 2-1
デバイス値のモニタ・変更 4-1
デバイスデータ書き込み関数 13-42
デバイスデータ読み出し関数 13-42
添付ファイルエディタ 12-16
- ## と
- 同一DRをすべての通信リクエストに使用する 6-5, 6-22
動作モード設定ダイアログボックス 17-37
同時接続可能クライアント数 6-25
登録モニタページ 13-29
特殊データレジスタ一覧 2-18
特殊内部リレー一覧 2-4
トピックにパブリッシュする 17-51
トピックをサブスクライブする 17-59
トランザクションID 6-29, 6-31
トレンドグラフ 13-40
- ## ね
- ネットワーク管理 3-11
ネットワークの設定 3-3
- ## は
- パブリッシュ 17-33
パリティ 4-5, 4-14, 6-2, 6-7, 6-8, 6-13
- ## ふ
- フレーム間タイムアウト 6-8
ブロードキャスト通信 6-2
プロトコルID 6-29, 6-31
プロパティの書き込み機能 15-6
プロパティの読み出し機能 15-6
- ## へ
- ペイロードダイアログボックス 17-42

ほ

- ポート1でのメンテナンス通信 4-4
- ポート2~33でのメンテナンス通信 4-12
- ポート番号 3-22
- 棒グラフ
 - 縦 13-38
 - 横 13-39
- ボロー 2-11

ま

- マスターデバイスアドレス 6-6, 6-23

め

- メタ文字のフォーマット
 - ワードデバイス 13-36
- メッセージ長 6-29, 6-31
- メンテナンス通信 1-9, 4-1

も

- モニタ機能 13-36

ゆ

- ユーザーWebページツリーの操作 13-31
- ユーザーWebページの作成 13-43
- ユーザーWebページの動作確認 13-46
- ユーザーアカウント設定 9-12, 10-5, 13-24
- ユーザー設定 13-22
- ユーザー通信 1-10, 5-1
- ユーザー通信 (UDP) 5-47
- ユーザー通信クライアント 5-42
- ユーザー通信サーバー 5-45
- ユーザー通信受信命令 (RXD) 5-9
- ユーザー通信送信命令 (TXD) 5-2
- ユーザー通信送信命令・受信命令のエラー 5-53
- ユーザー通信命令 5-25
 - RXD (ユーザー通信受信) 命令 5-28
 - TXD (ユーザー通信送信) 命令 5-25
- ユーザー通信命令のキャラクタコード 5-54
- ユーザープログラムのアップロード 4-1
- ユーザープログラムのダウンロード 4-1
- ユニットID 6-29, 6-31

り

- リトライ回数 6-2, 6-7
- リモートホスト 3-22
- リモートホストリスト 3-21
- リモートホスト番号 6-24

ろ

- ローカルホストポート番号 4-9, 4-18, 6-28

製品の保証について

1 保証期間

弊社製品の保証期間は、ご購入後またはご指定の場所に納入後3年間といたします。ただし、カタログ類に別途の記載がある場合やお客様と弊社との間で別途の合意がある場合は、この限りではありません。

2 保証範囲

上記保証期間中に弊社側の責により弊社製品に故障が生じた場合は、その製品の交換または修理を、その製品のご購入場所・納入場所、または弊社サービス拠点において無償で実施いたします。ただし、故障の原因が次に該当する場合は、この保証の対象範囲から除外いたします。

- 1) カタログ類に記載されている条件・環境の範囲を逸脱した取り扱いまたは使用による場合
- 2) 弊社製品以外の原因の場合
- 3) 弊社以外による改造または修理による場合
- 4) 弊社以外の者によるソフトウェアプログラムによる場合
- 5) 弊社製品本来の使い方以外の使用による場合
- 6) 取扱説明書、カタログ類の記載に従って、保守部品の交換、アクセサリ類の取り付けなどが正しくされていなかったことによる場合
- 7) 弊社からの出荷当時の科学・技術の水準では予見できなかった場合
- 8) その他弊社側の責ではない原因による場合（天災、災害など不可抗力による場合を含む）

なお、ここでの保証は、弊社製品単体の保証を意味するもので、弊社製品の故障により誘発される損害は保証の対象から除かれるものとします。

※ お客様がプログラム可能な製品については、お客様ご自身の責任の下で動作確認いただくことといたします。お客様にてプログラミングされたプログラムの動作およびそれにより発生した損害については、当社はいかなる場合も責任を負いかねます。

3 サービス範囲

弊社製品の価格には、技術者派遣等のサービス費用は含んでおりませんので、次の場合は別途費用が必要となります。

- 1) 取付調整指導および試験運転立ち合い（アプリケーション用ソフトの作成、動作試験等を含む）
- 2) 保守点検、調整および修理
- 3) 技術指導および技術教育
- 4) お客様のご指定による製品試験または検査

IDEC株式会社

〒532-0004 大阪市淀川区西宮原2-6-64

 jp.idec.com



お問い合わせはこちらから

- 本マニュアル中に記載されている社名及び商品名はそれぞれ各社が商標または登録商標として使用している場合があります。
- 仕様、その他記載内容は予告なしに変更する場合があります。
- 本マニュアルにご不明な点がございましたら、製品問合せ窓口にお問い合わせください。

B-1729(17) 本マニュアル記載の情報は、2023年11月現在のものです。

■ IDEC