

取扱説明書

CODESYS MODBUS I / O Driver

Version 3

(MgloDrvModbus3)

おことわり

- ・本書の内容の一部または全部を無断で複写、複製、転載することを禁じます。
- ・本書の内容は、改良のため予告なしに仕様などを変更することがあります。
- ・本書の内容は、万全を期しておりますが、万一ご不審な点や誤りなど、お気づきのことがありましたら、お手数ですが、弊社までご連絡下さい。

商標

- ・ **CODESYS** は、**CODESYS GmbH** の登録商標です。
- ・ **MODBUS** は、**Schneider Electric** の登録商標です。
- ・ その他、記載されている会社名や製品名などにつきましては、各社の登録商標または商標です。

1. ご利用いただく前に

このたびは、弊社製品をお買い上げいただき、誠にありがとうございます。

本書は、弊社製品を使用する上で、必要な情報を記載しています。

ご使用前に、本書をすべてご熟読いただき、機能、性能や注意事項などを十分にご理解いただき、システムの構築にご利用下さい。

また、お読みになった後も本書は大切に保管して、いつも手元においてお使い下さい。

2. ご注意

- ・ 本書（内容の一部または全部）を無断で転載することは禁止します。
- ・ 本書については将来予告なしに変更することがあります。
- ・ 本書については万全を期して作成しました。万一、本書に誤記、記載漏れなどがありましたらご連絡下さい。

3. 本ソフトウェアについて

本ソフトウェアは CODESYS ランタイム（64bit）で動作する MODBUS 通信ドライバーです。

このソフトウェアを使用することで CODESYS ランタイムが動作する機器で MODBUS Serial Server/Client および MODBUS Ethernet Server/Client 機能を使用することができます。

【動作確認済み環境】

動作確認は以下の環境で実施しています。

種 別	形 式	対応バージョン
PLC	BA1C-PAC-A	1.0.0.16 以降
	PLC8-NE	1.0.0.15 以降
プログラミングツール	C-CDS35-21-10	3.5.21.10
ESSENTIAL パッケージ	MG_ESSENTIAL	3.0.0.0 以降
BA1C-PAC パッケージ	MG_BA1C_PAC	3.0.0.0 以降
	MG_PLC8_NE	3.0.0.0 以降

4. 記載事項のご注意

本書の記載内容については、次の点をご理解下さい。

- ・ 設定範囲は、その全範囲において正常に動作することを保証するものではありません。また、デフォルト値 (初期値) は、お客様のシステムで適切な値であるとは限りません。
- ・ 利用事例やサンプルプログラムは、参考ですので当社は適合性等を一切保証いたしかねます。

5. ご利用に際しての注意事項

ご採用およびご使用に際しては次の点をご理解下さい。

- ・ 仕様、性能など利用に関する条件を厳守してご使用下さい。
- ・ 当社製品の「適合性」は、お客様自身でご確認いただき、当社製品のご利用の可否を判断下さい。
- ・ 「当社製品」がお客様のシステムの中で意図した用途で使用され、期待する動作や結果となっていることをお客様ご自身で事前に確認して下さい。
- ・ 当社の製品をご使用の際は、性能に十分な余裕のある利用、故障などにより生じる危険を最小にする安全設計、利用者に危険を通知する安全対策やシステムの継続的な保守を実施して下さい。
- ・ 当社は DDoS 攻撃 (分散方 DoS)、コンピュータウイルスやその他の技術的な有害プログラム、不正アクセスにより、当社製品、インストールされたソフトウェア、またはすべてのコンピュータ機器、コンピュータプログラム、ネットワーク、データが感染したとしても、そのことにより直接または間接的に生じた損失、損害その他の費用について一切責任を負わないものとします。

お客様ご自身にて、ウイルスの感染防止、データの保護、本製品への不正アクセス防止の十分な措置を講じて下さい。

- ・ 当社製品は一般工業製品向けの汎用製品として設計製造されています。従いまして、次にあげる用途での使用は想定されており一切の保証をいたしません。

- (A) 高い安全性が必要とされる用途 (例: 原子力制御設備、燃焼設備、航空・宇宙設備、鉄道設備、昇降設備、娯楽設備、医用機器、安全装置、その他生命・身体に危険が及びうる用途)
- (B) 高い信頼性が必要な用途 (例: ガス・水道・電気等の供給システム、24 時間連続運転システム、決済システムほか権利・財産を取扱う用途など)
- (C) 「カタログ等」に記載のない条件や環境での用途

目次

1. ご利用いただく前に	3
2. ご注意	3
3. 本ソフトウェアについて	3
4. 記載事項のご注意	4
5. ご利用に際しての注意事項	4
6. 使用上の注意	8
ライブラリの使用時	8
サンプルプログラムの使用時	8
当社コントローラ以外で使用する場合の制限	8
7. 概要	9
MODBUS 通信とは	9
MODBUS Protocol	9
本ソフトウェアの特徴	10
8. 用語	11
アドレス (Address)	11
ファンクション・コード	11
ASCII モード	11
RTU モード	11
MODBUS アドレス (MODBUS Address)	11
コイル (Coil)	12
入力ステータス (InputStatus)	12
入力レジスタ (Input Register)	12
保持レジスタ (Holding Register)	12
ワード	12
Server ID	12
Unit ID	12
Gateway	12
MODBUS Server Multi server ID simulation	13
9. 仕様・動作環境	14
9.1 仕様	14
[ETHERNET MODBUS CLIENT]	14
[ETHERNET MODBUS SERVER]	14
[SERIAL MODBUS CLIENT]	15
[SERIAL MODBUS SERVER]	15
9.2 動作環境	15

10. 導入手順	16
10.1 インストール	16
10.2 アンインストール	16
11. IO ドライバーの構成と設定項目	17
11.1 Modbus Ethernet	17
11.2 Modbus Ethernet Client	18
11.3 Modbus Ethernet Device.....	19
11.4 Modbus Ethernet Server	22
11.5 Modbus Ethernet Local Device	24
MODBUS Server Multi server ID simulation	24
RTU Client PATH THROUGH Gateway on MODBUS TCP / UDP Server.....	24
11.6 Modbus Serial	27
11.7 Modbus Serial Client	27
11.8 Modbus Serial Device	28
11.9 Modbus Serial Server	31
11.10 Modbus Serial Local Device	33
MODBUS Server Multi server ID simulation	33
12. 機能の解説	35
12.1 Bus Cycle Task.....	35
時間判定	35
「Bus Cycle Task」の処理時間.....	35
12.2 Client のトリガー	35
[APPLICATION]	35
[CYCLIC]	36
[RISING EDGE]	36
12.3 API で指定する DeviceIndex, ChannelIndex について.....	37
12.4 Server の Output, Holding Register の書き込み	37
12.5 割り当てた変数の初期値 (前回最終値)	37
13. 使用方法	38
13.1 MODBUS Ethernet CLIENT	38
MODBUS Ethernet CLIENT 通信の準備	38
CODESYS プロジェクトの準備	38
設定手順	39
13.2 MODBUS Ethernet SERVER.....	43
MODBUS Ethernet SERVER 通信の準備.....	43
CODESYS プロジェクトの準備	43
設定手順	43
13.3 MODBUS Serial CLIENT	48
MODBUS Serial CLIENT 通信の準備	48
CODESYS プロジェクトの準備	48
設定手順	48

13.4 MODBUS Serial SERVER	52
MODBUS Serial SERVER 通信の準備.....	52
CODESYS プロジェクトの準備	52
設定手順	52
14. 拡張機能	56
14.1 CLIENT のチャンネル状態の監視.....	56
14.2 MODBUS CLIENT 通知機能	56
14.3 MODBUS SERVER 通知機能	57
15. 添付	59
15.1 列挙型コード一覧	59
ET_CLIENT_STATUS.....	59
ET_MODBUS_CHANNEL_STATUS	59
ET_MODBUS_ERROR	59
ET_MODULE_STATUS.....	60
ET_SERVER_COMM_STATUS.....	61
15.2 セキュリティ	61
MODBUS Server を読み取り専用にする	61
MODBUS Server への要求は特定の相手に限定する	61
16. 異常時の処置とメンテナンス	62
16.1 オンラインモードでのデバイスステータス	62
16.2 デバイス ログ	62
16.3 トラブルシューティング	63
MODBUS Client に関するトラブル	63
MODBUS Sever に関するトラブル.....	63
17. 履歴	64

6. 使用上の注意

ライブラリの使用時

- ・取扱説明書やオンラインヘルプで公開されています DUT, FUNCTION, FUNCTION BLOCK 以外は使用しないで下さい。
- ・ライブラリの内部で参照しています外部ライブラリは、必要に応じて「プレースホルダ」にてお客様のプロジェクトで適切なライブラリが参照されるよう調整して下さい。
- ・「バスサイクルタスク」を必要とする I/O ドライバーを使用する際は、お客様のプロジェクトで適切なタスクを指定して下さい。これは I/O ドライバーのデータ (I/O Mapping された変数も含む) を POU で使用する際に重要な事項です (詳細は CODESYS の「バスサイクルタスク」を別途参照して下さい)。

サンプルプログラムの使用時

- ・適切なエラー処理を追加するなど、実際の装置が性能や安全面で意図した動作を行うようにユーザプログラムを作成して下さい。
- ・作成したユーザプログラムを本運転へ移行する際は、十分な動作確認の後に行うようにして下さい。

当社コントローラ以外で使用する場合の制限

- ・本ソフトウェアを当社製品以外のコントローラ (CODESYS Control Win V3-x64 など) で動作させる場合は、デモモード (動作時間に制限あり) となります。デモモードは、コントローラが RUN してから 1 時間経過すると本ソフトウェアの動作が停止します。その場合は、コントローラをコールドスタートすることで再度動作を開始できます。

7. 概要

MODBUS 通信とは

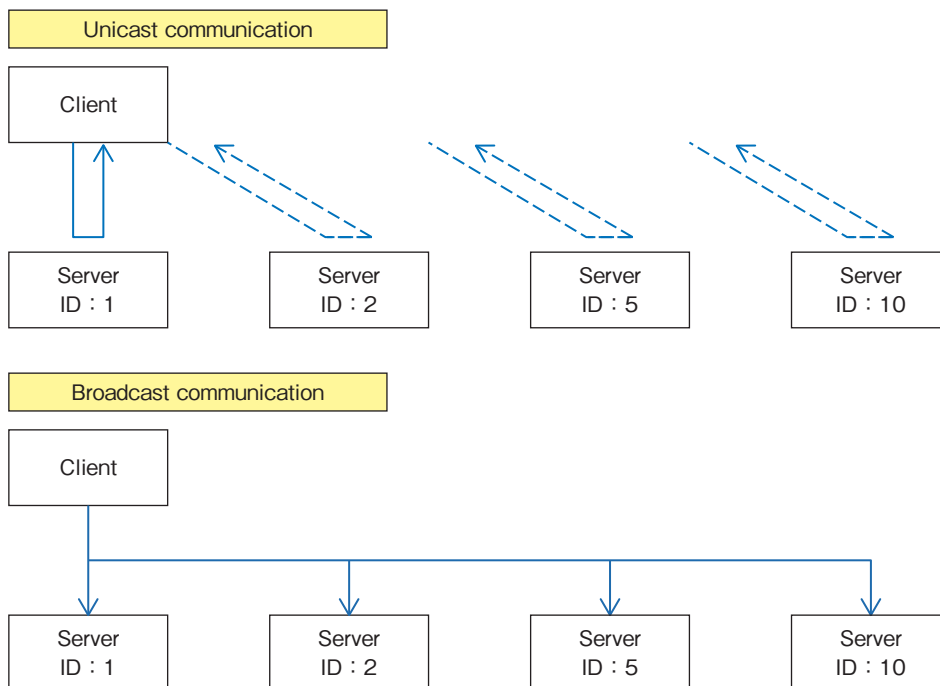
通信プロトコルとして MODBUS Protocol を実装したネットワークを、MODBUS と呼んでいます。MODBUS Protocol は、Modicon Inc. (AEG Schneider Automation International S.A.S.) が PLC 用に開発した通信プロトコルであり、その仕様は公開されています。

MODBUS Protocol

MODBUS の通信方式は、シングルクライアント/マルチサーバー方式です。クライアントだけがクエリ（通信の開始）を発行することができます。サーバーは、このクエリを見て、指定された機能を実行し、応答メッセージを返します。クライアントは、指定のサーバーに対するユニキャストクエリまたはすべてのサーバーに対するブロードキャストクエリのいずれかを発行することができます。ブロードキャストクエリの場合のサーバーは、指定の機能を実行するのみで応答メッセージは返しません。ユニキャストクエリの場合のサーバーは、自分に対するクエリの際にだけ応答メッセージを返します。

クエリの伝送フォーマットは、サーバーのアドレス（またはブロードキャスト）、要求内容を定義するファンクション・コード、データおよびエラーチェック・フィールドから構成されています。また、応答メッセージの伝送フォーマットは、要求内容の確認フィールド、応答データおよびエラーチェック・フィールドから構成されています。

シリアル伝送モードには ASCII (American Standard Code for Information Interchange) モードと RTU (Remote Terminal Unit) モードの 2 種類があり、選択することができます。ただし、1つのネットワーク上では、すべてのデバイスが同一モードでなくてはなりません。



本ソフトウェアの特徴

- MODBUS ASCII Server／Client（Broadcast 対応）
- MODBUS RTU Server／Client（Broadcast 対応）
- MODBUS TCP Server／Client
- MODBUS UDP Server／Client
- Support Function code 1,2,3,4,5,6,15,16,(20),(21),23（）はFBで対応
- RTU Client PATH THROUGH Gateway on MODBUS TCP／UDP Server
- MODBUS Server Multi server ID simulation に対応
- CODESYS で入力にマップされた変数の RETAIN 属性に対応
- CODESYS で MODBUS Server の Holding Register にマップされた変数への書き込みに対応
- CODESYS から MODBUS 通信定義 Channel 毎の状態参照に対応
- CODESYS のデバイスステータス画面に MODBUS 通信状態を表示

8. 用語

アドレス (Address)

アドレス・フィールドは、0 から 247 (10 進数) が許されます。サーバー・アドレスは 1 から 247 です。クライアントがサーバーにクエリを発する場合には、このアドレス・フィールドにサーバーのアドレスをセットします。サーバーがクライアントに応答メッセージを返す場合には、サーバーのアドレスをセットします。これによって、クライアントはどのサーバーからの応答であるかを知ることができます。アドレス 0 はブロードキャストクエリに用います。ただし、MODBUS TCP では GATEWAY 機能を使用する場合を除き 0 または 255 を指定します。この値は Server ID と呼ばれます。本通信ドライバーの設定項目では Unit ID と表示しています。

ファンクション・コード

指定可能なファンクション・コードは 1 から 255 (10 進数) です。ファンクション・コードに従って、サーバーは指定された機能を実行します。実行後、応答メッセージを返す場合、正常応答メッセージには同じファンクション・コードを設定し、例外応答メッセージにはファンクション・コードの MSB を 1 にセットします。これで、クライアントはどのファンクション・コードに関する応答メッセージであるかを知ることができます。指定したファンクション・コードが有効か否かは、サーバーデバイスに依存します。

ファンクションコード	機能	本ドライバー対応
01	Read Coils	○
02	Read Discrete Inputs	○
03	Read Holding Registers	○
04	Read Input Registers	○
05	Write Single Coil	○
06	Write Single Register	○
07	Read Exception Status	
08	Diagnostics	
11 (0x0B)	Get Comm Event Counter	
12 (0x0C)	Get Comm Event Log	
15 (0x0F)	Write Multiple Coils	○
16 (0x10)	Write Multiple Registers	○
17 (0x11)	Report Server ID	
20 (0x14)	Read File Record	△ Function Block
21 (0x15)	Write File Record	△ Function Block
22 (0x16)	Mask Write Register	
23 (0x17)	Read/Write Multiple Registers	○

ASCII モード

ASCII モードの場合には、エラーチェック・フィールドは 2 文字から構成されます。Start “:” 文字と End “CR/LF” を除く文字の和 (LRC: Longitudinal Redundancy Check) の結果 2 文字で表します。

RTU モード

RTU モードの場合には、エラーチェック・フィールドは 16 ビットのデータを 2 つの 8 ビットバイトデータで表します。エラーチェックの結果は CRC (Cyclical Redundancy Check calculation) と言われる計算方法で計算されます。

MODBUS アドレス (MODBUS Address)

クライアントの要求では Coil や Register への開始アドレス・フィールド (Offset) が指定されます。このアドレスは 5 桁あるいは 6 桁で表現されます。

領域名	5 桁アドレス範囲	6 桁アドレス範囲	アドレス・フィールド範囲
Coil	1 ~ 9999	1 ~ 65536	0 ~ 65535
Input Status	10001 ~ 19999	100001 ~ 165536	0 ~ 65535
Input Register	30001 ~ 39999	300001 ~ 365536	0 ~ 65535
Holding Register	40001 ~ 49999	400001 ~ 465536	0 ~ 65535

コイル (Coil)

コイルは、フィールドへの ON/OFF 出力である DO (Discrete Output) やサーバーデバイスの状態やモードを変更する為にスイッチとして用いられます。参照・変更が可能な 2 値のデータです。

入力ステータス (InputStatus)

入力ステータスは、フィールドからの ON/OFF 入力である DI (Discrete Input) やサーバーデバイスの状態入力として用いられます。参照のみで変更はできない 2 値のデータです。

入力レジスタ (Input Register)

入力レジスタは、フィールドからの AI (Analog Input) やサーバーデバイス内の情報として用いられます。16 ビット長のデータで、参照するのみで変更はできません。複数の連続したアドレスを割当てることにより、単精度実数、倍精度実数などのデータを扱うこともできます。

保持レジスタ (Holding Register)

保持レジスタは、フィールドからの AO (Analog Output) やサーバーデバイス内の設定情報として用いられます。16 ビット長のデータで、参照・変更ができます。複数の連続したアドレスを割当てることにより、単精度実数、倍精度実数などのデータを扱うこともできます。

ワード

16 ビット単位で構成されるデータを表す単位です。1 ワードは、符号付 10 進数表記で “-32768 ~ 32767”、符号無し表記で “0 ~ 65535”、16 進数表記で “0x0000 ~ 0xFFFF” の範囲となります。

Server ID

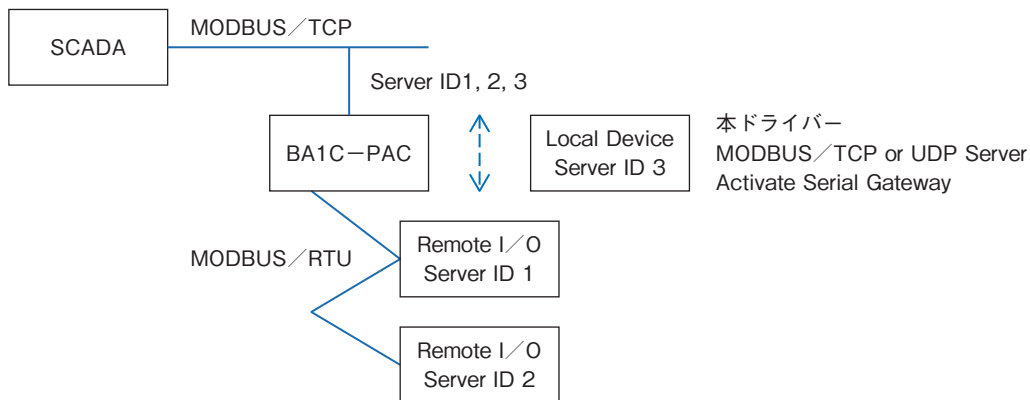
上記「アドレス」を参照下さい。

Unit ID

上記「アドレス」を参照下さい。

Gateway

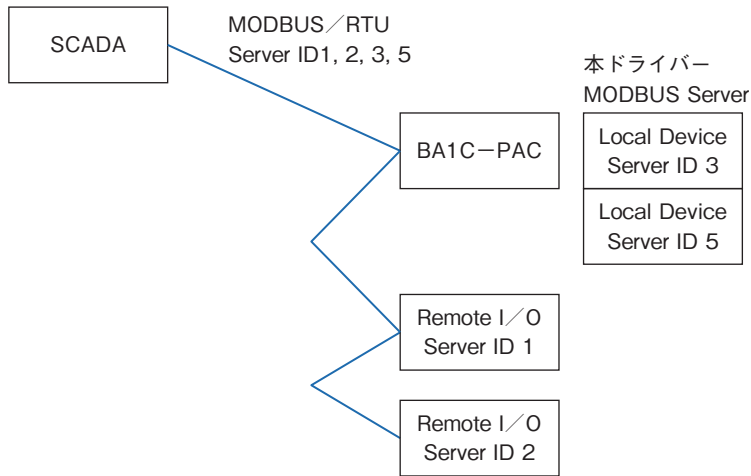
MODBUS TCP/UDP Server において LAN 上の MODBUS の要求を MODBUS RTU に変換してシリアル回線に中継する機能です。



この図では Server ID 1,2 への要求は Remote I/O がデータを返し、Server ID 3 への要求は Local Device がデータを返します。

MODBUS Server Multi server ID simulation

MODBUS Server 定義において複数のデバイスを定義することで仮想デバイス（実デバイスなし）がデータを返す機能です。



この図では Server ID 1,2 への要求は Remote I/O がデータを返し、Server ID 3,5 への要求は Local Device がデータを返します。

9. 仕様・動作環境

本製品の仕様と動作環境を次に記述します。

9.1 仕様

[ETHERNET MODBUS CLIENT]

項目	範囲	備考
通信プロトコル	MODBUS TCP/UDP	
ポート番号	1~65535	(502)
IP アドレス	IPv4 範囲 0.0.0.1~255.255.255.254	Client と Server で同じネットワークアドレスを使用
Server ID	範囲 0~255 (規定は 0,1~247,255)	1~247 は MODBUS Serial Gateway
接続 device 数 node 当たりの query	最大 32 nodes 最大 30 channels	AccessType: FC1,2,3,4,5,6,15,16,23 Trigger: Application,Cyclic,Rising Edge Cycle: 0..300,000[ms] Read error handling: Keep last value,Set to zero
Query 領域	Coil	0~65535 (MODBUS 6 桁 1~65536)
	DiscreteInput	0~65535 (MODBUS 6 桁 100001~165536)
	InputRegister	0~65535 (MODBUS 6 桁 300001~365536)
	Holding Register	0~65535 (MODBUS 6 桁 400001~465536)

FC : Function Code の略

[ETHERNET MODBUS SERVER]

項目	範囲	備考
通信プロトコル	MODBUS TCP/UDP	
ポート番号	1~65535	(502)
IP アドレス	IPv4 範囲 0.0.0.1~255.255.255.254	Client と Server で同じネットワークアドレスを使用
同時接続台数	最大 10 clients	
Server ID	範囲 0~255 (規定は 0,1~247,255)	1~247 は MODBUS Serial Gateway
device 定義数	最大 32 nodes	複数定義することで device をエミュレーション MODBUS Serial Gateway が有効な場合は、ここに定義されていない node への要求が Gateway されます。
device 領域設定	最大 10 領域	注) 一回の MODBUS 要求でアクセスできる領域は 1 つです。複数の領域で連続したメモリーを定義しても 1 回の MODBUS 要求で領域をまたぐことはできません。要求の開始位置の存在する領域の範囲で評価されます。
	Coil	0~65535 (MODBUS 6 桁 1~65536)
	Discrete Input	0~65535 (MODBUS 6 桁 100001~165536)
	Input Register	0~65535 (MODBUS 6 桁 300001~365536)
	Holding Register	0~65535 (MODBUS 6 桁 400001~465536)
MODBUS RTU GATEWAY 機能	機能は有効/無効を設定可能	MODBUS Serial Gateway 機能を有効の場合で定義された device に該当しない ServerID 要求が対象

[SERIAL MODBUS CLIENT]

項目	範囲	備考
通信プロトコル	MODBUS ASCII/RTU	
伝送速度	Baudrate: 4800, 9600, 19200, 38400, 57600, 115200 Parity: NONE, ODD, EVEN Data Bits: 7, 8 Stop Bits: 1, 2	
Server ID	範囲 0~255 (規定は 0,1~247)	0 は Broadcast
接続 device 数	最大 32 nodes	
node 当たりの query	最大 30 channels	Access Type: FC1,2,3,4,5,6,15,16,23 Trigger: Application, Cyclic, Rising Edge Cycle: 0 .. 300,000 [ms] Read error handling: Keep last value, Set to zero
Query 領域	Coil	0~65535 (MODBUS 6 桁 1~65536)
	Discrete Input	0~65535 (MODBUS 6 桁 100001~165536)
	Input Register	0~65535 (MODBUS 6 桁 300001~365536)
	Holding Register	0~65535 (MODBUS 6 桁 400001~465536)

[SERIAL MODBUS SERVER]

項目	範囲	備考
通信プロトコル	MODBUS ASCII/RTU	
伝送速度	Baudrate: 4800, 9600, 19200, 38400, 57600, 115200 Parity: NONE, ODD, EVEN Data Bits: 7, 8 Stop Bits: 1, 2	
同時接続台数	1 client	
Server ID	範囲 0~255 (規定は 0,1~247)	0 は Broadcast
device 定義数	最大 32 nodes	複数定義することで device をエミュレーション (注) Broadcast 要求では最初に定義された device だけを対象とします。その他の device は更新されません。
device 領域設定	最大 10 領域	(注) 一回の MODBUS 要求でアクセスできる領域は 1 つです。複数の領域で連続したメモリーを定義しても 1 回の MODBUS 要求で領域をまたぐことはできません。要求の開始位置の存在する領域の範囲で評価されます。
	Coil	0~65535 (MODBUS 6 桁 1~65536)
	Discrete Input	0~65535 (MODBUS 6 桁 100001~165536)
	Input Register	0~65535 (MODBUS 6 桁 300001~365536)
	Holding Register	0~65535 (MODBUS 6 桁 400001~465536)

9.2 動作環境

CODESYS V3.5 SP 21 Patch 1 以降のランタイム (SysCom, SysSocket 必須)

当社製品 : BA1C-PAC など

他社製品 : CODESYS Control Win3-x64 (64bit Windows)

10. 導入手順

本製品のインストール、およびアンインストール手順を説明します。

10.1 インストール

CODESYS の [CODESYS Installer...] Install File(s) にてインストールを行います。

10.2 アンインストール

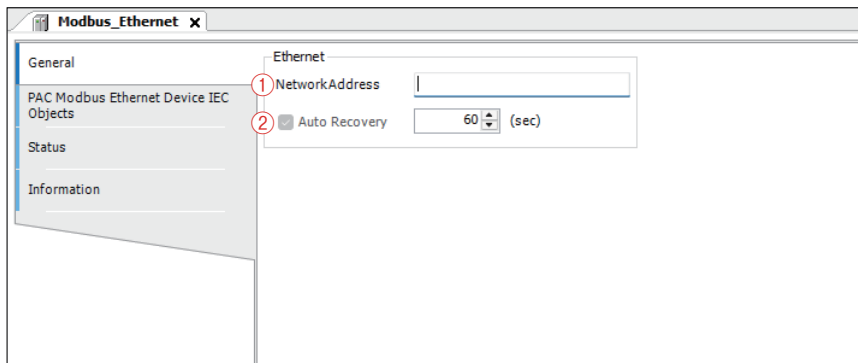
CODESYS の [CODESYS Installer...] Uninstall selected にてアンインストールを行います。

11.10 ドライバーの構成と設定項目

構成は以下の通りです。

Modbus Ethernet (Network Address)		
Client, Server 合計最大 32	Modbus Ethernet Client (IP Address)	
	最大 32	Modbus Ethernet Device (Unit ID) Channel: 最大 30
	Modbus Ethernet Server (Port)	
	最大 32	Modbus Ethernet Local Device (Unit ID) Memory Area: 最大 10
Modbus Serial (COM Port)		
Client または Server 最大 1	Modbus Serial Client	
	最大 32	Modbus Serial Device (Unit ID) Channel: 最大 30
	Modbus Serial Server	
	最大 32	Modbus Serial Local Device (Unit ID) Memory Area: 最大 10

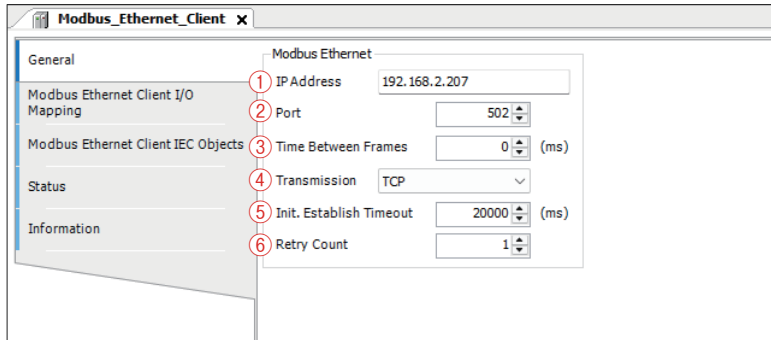
11.1 Modbus Ethernet



項目名	内容	設定範囲(): 初期値
① Network Address	ネットワークインタフェースを指定します。 Server の場合に対象とする LAN Interface を限定することができます。その場合は対象とする LAN Interface に割り当てられた IP Address を指定します。指定の無い場合は全 LAN ポートが対象となります。	使用可能な文字はシングルクォーテーションを除く半角英数字です。 最大 63 文字 (空欄: 指定しない) 設定例: 192.168.1.200
② Auto Recovery	ネットワークエラー時の復帰試行間隔を指定します。	0~300 [sec] (60)

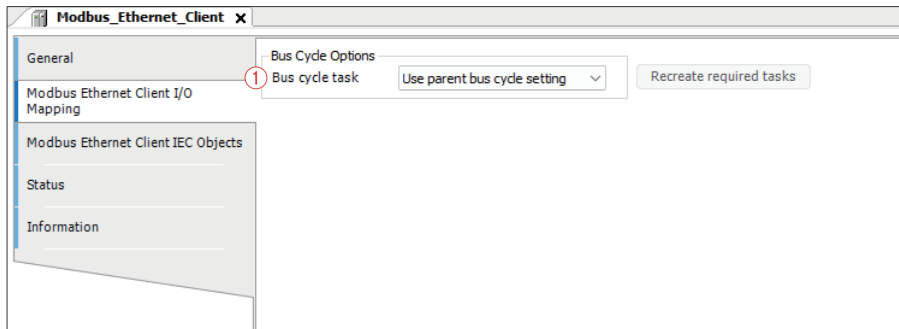
11.2 Modbus Ethernet Client

[General]



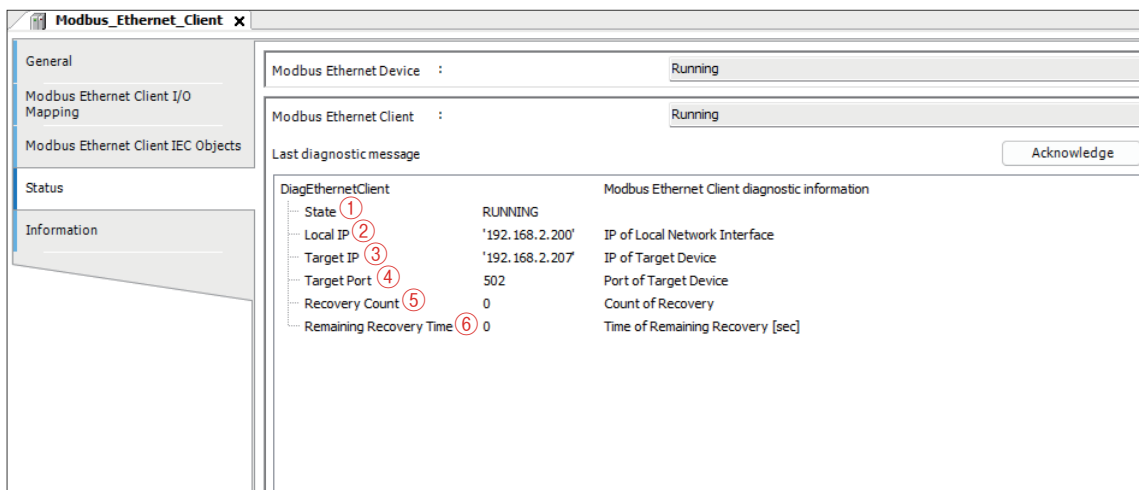
項目名	内容	設定範囲(): 初期値
① IP Address	接続先 Server デバイスの IP アドレスを指定します。	設定例: 192.168.1.207
② Port	接続先 Server デバイスの Port 番号を指定します。	0~65535 (502)
③ Time Between Frames	次の MODBUS リクエストフレームの送信に間隔時間を指定します。	0~5000 [ms] (0)
④ Transmission	通信形式を指定します。	TCP / UDP (TCP)
⑤ Init. Establish Timeout	通信ソケット生成時のタイムアウト時間を設定します。	0~300000 [ms] (20000)
⑥ Retry Count	MODBUS 通信失敗時のリトライ回数を指定します。	0~10 (1)

[Modbus Ethernet Client I/O Mapping]



項目名	内容	設定範囲(): 初期値
① Bus Cycle task	MODBUS 通信は、ここで指定したタスク周期で動作します。複数チャネルの設定が設定されていても 1 バスサイクルで実行されるのは 1 チャネルのみです。	(Use parent bus cycle setting) または表示される Task から選択

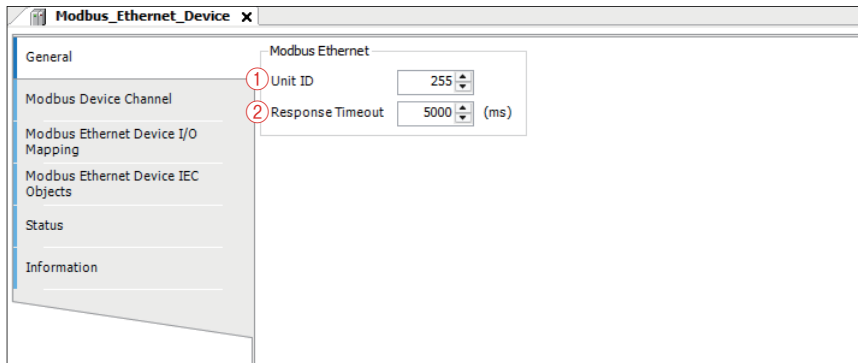
[Status]



項目名	内容	備考
① State	現在状態が表示されます。 ET_MODULE_STATE	RUNNING: 正常動作中であることを示しています。
② Local IP	自身の IP Address が表示されます。	
③ Target IP	相手の IP Address が表示されます。	
④ Target Port	相手の Port アドレスが表示されます。	
⑤ Recovery Count	試行中の回数が表示されます。	
⑥ Remaining Recovery Time	リカバリ時に開始残り時間が表示されます。	

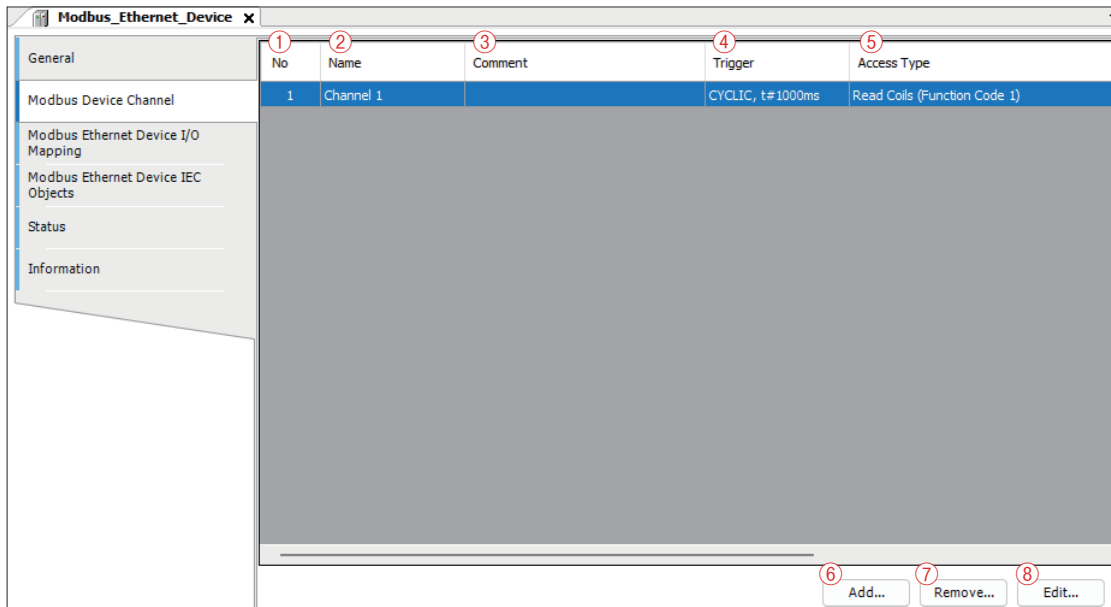
11.3 Modbus Ethernet Device

[General]



項目名	内容	設定範囲(): 初期値
① Unit ID	相手のユニット番号 (Server ID) を指定します。	0~255 (255)
② Response Timeout	レスポンスタイムアウト時間を指定します。	0~300000 [ms] (5000)

[Modbus Device Channel]



項目名	内容	備考
① No	連番が表示されます。	
② Name	名称が表示されます。	
③ Comment	注釈が表示されます。	
④ Trigger	トリガー内容が表示されます。	
⑤ Access Type	アクセス方法が表示されます。	
⑥ [Add...] ボタン	新しくチャンネルを追加します。	
⑦ [Remove...] ボタン	現在選択されているチャンネル定義を削除します。	複数行の選択も可能です。

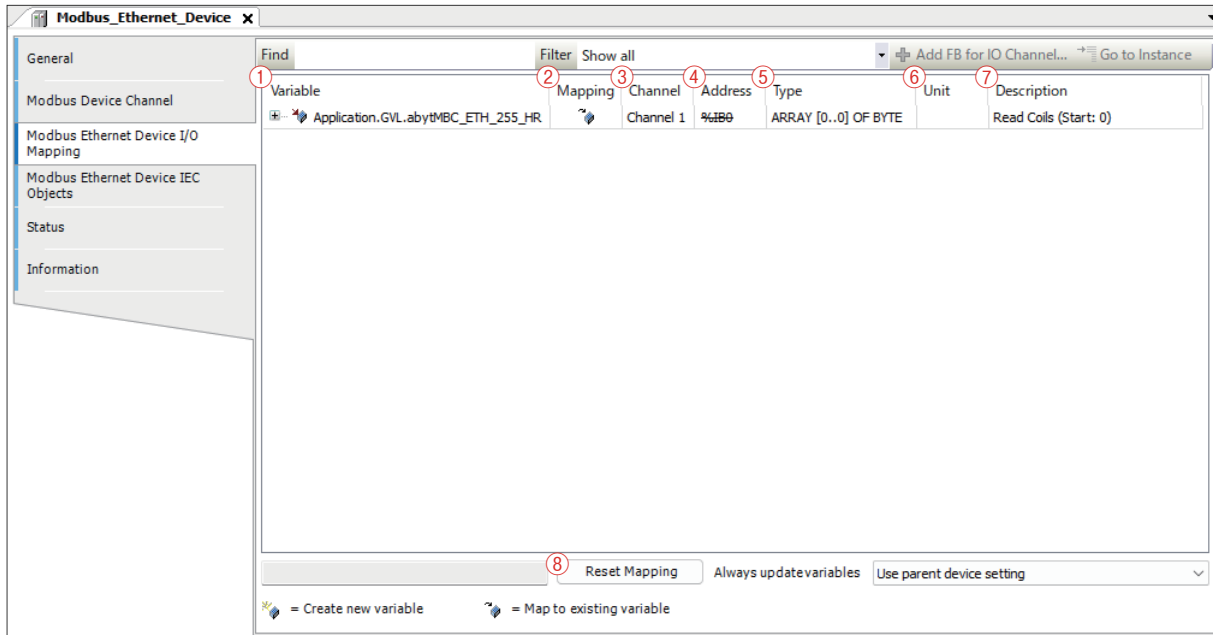
C—CDS35: MgloDrvModbus3

項目名	内容	備考
⑧ [Edit...] ボタン	現在選択されているチャンネルを編集します。	複数行の選択はできません。 ここで内容を変更すると対象チャンネルの[I/O Mapping]情報はリセットされます。必要であれば再度指定して下さい。

[Modbus Device Channel—Add/Edit]

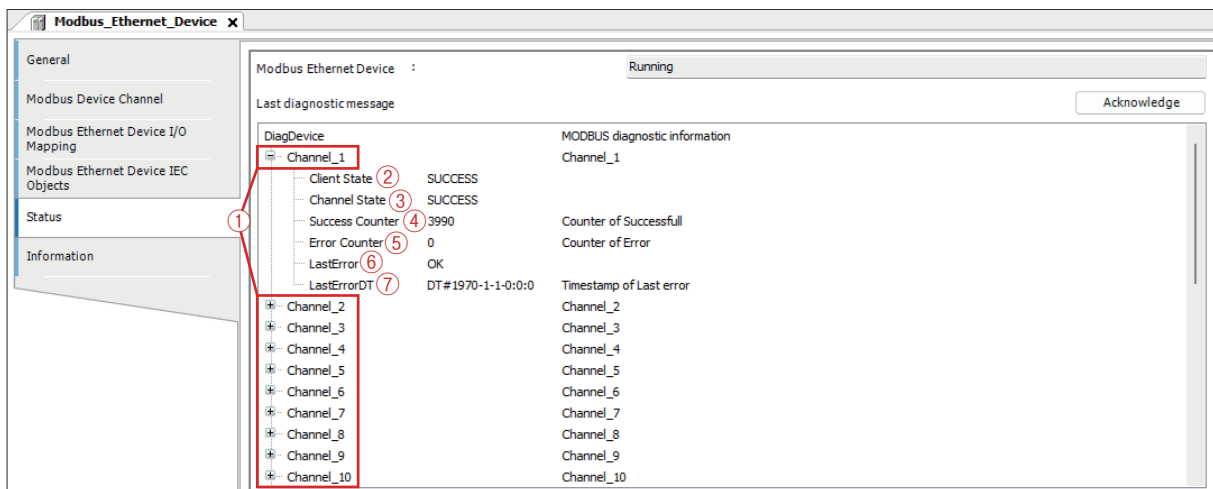
項目名	内容	設定範囲(): 初期値
[Channel]		
① Name	名称を指定します。	使用可能な文字はシングルクォーテーションを除く半角英数字です。空欄や重複する名称は不可です。 最大 80 文字
② Comment	注釈を指定します。	使用可能な文字は半角英数字です。 最大 80 文字
③ Access Type	アクセス方法を指定します。	Read Coils (Function Code 1) Read Discrete Inputs (Function Code 2) (Read Holding Registers (Function Code 3)) Read Input Registers (Function Code 4) Write Single Coil (Function Code 5) Write Single Register (Function Code 6) Write Multiple Coils (Function Code 15) Write Multiple Registers (Function Code 16) Read/Write Multiple Registers (Function Code 23)
④ Trigger	トリガーを指定します。	APPLICATION: プログラムコードで起動 (CYCLIC): 定周期で起動 RISING_EDGE: 変数値の立ち上がりで起動
⑤ Cycle Time	Trigger=CYCLIC 時に周期を指定します。	0~300000 [ms] (1000)
[READ Register]		
⑥ Offset	オフセットを指定します。 Offset+Length が 65536 を超えることはできません。	0~65535 (0)
⑦ Length	長さを指定します。	1~2000 [Function Code 1, 2] (1) 1~125 [Function Code 3, 4, 23]
⑧ Error Handling	異常時の処理を指定します。	(Keep last value) Set to zero
[WRITE Register]		
⑨ Offset	オフセットを指定します。 Offset+Length が 65536 を超えることはできません。	0~65535 (0)
⑩ Length	長さを指定します。	1 [Function Code 5, 6] (1) 1~1968 [Function Code 15] 1~123 [Function Code 16] 1~121 [Function Code 23]
⑪ [OK] ボタン	設定を確定します。	
⑫ [Cancel] ボタン	設定を破棄します。	

[Modbus Ethernet Device I/O Mapping]



項目名	内容	設定範囲(): 初期値
① Variable	割り当てる変数を指定します。	新規の名称か既存の変数を指定できます。既存の変数を指定する場合は、その変数がTypeで表示されている型と一致している必要があります。
② Mapping	新規変数か既存変数への割り当てかがアイコンで表示されます。	
③ Channel	チャンネル名称が表示されます。	
④ Address	割り当てられたアドレスが表示されます。既存の変数を割り当てた場合は取り消し線が表示されます。	
⑤ Type	必要な変数の型が表示されます。	
⑥ Uint	工業単位が表示されます。	(未使用)
⑦ Description	注釈が表示されます。	
⑧ [Reset Mapping] ボタン	割付をリセットします。	

[Status]

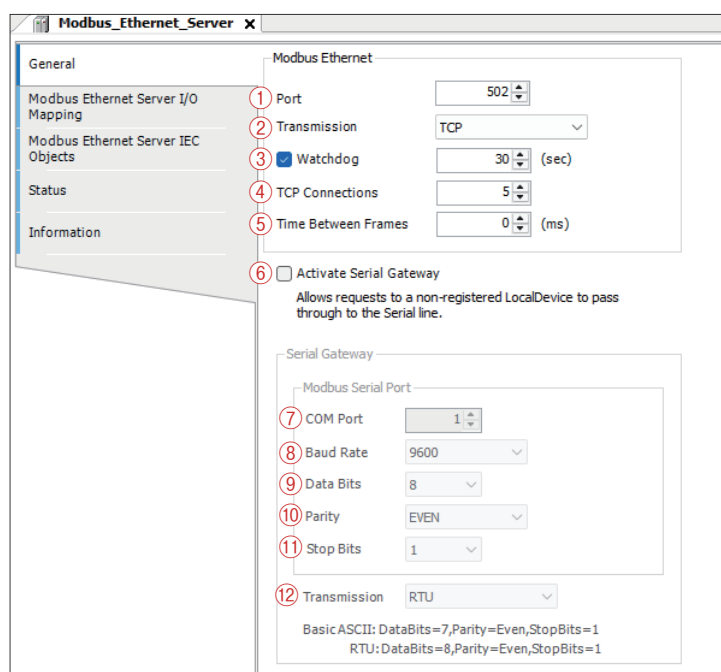


項目名	内容	備考
① Channel_N	チャンネル識別子が表示されます。	
② Client State	クライアント状態が表示されます。 ET_CLIENT_STATUS	TRIGGERD: トリガー状態を示しています。 SUCCESS: MODBUS 通信が正常に完了したことを示しています。

項目名	内容	備考
③ Channel State	チャンネル状態が表示されます。 ET_MODBUS_CHANNEL_STATUS	UNCONFIGURED: 未設定 CONFIGURED: 設定状態 SUCCESS: チャンネルが正常に処理されていることを示しています。 ERROR: 異常状態
④ Success Counter	成功カウント数が表示されます。	
⑤ Error Counter	エラーカウント数が表示されます。	
⑥ LastError	最新のエラー状態が表示されます。 ET_MODBUS_ERROR	OK: エラーなし(初期状態も OK 表示となります)
⑦ LastErrorDT	最新のエラー発生日付時刻が表示されます。	'DT#1970-0-0-0:0:0' は初期状態

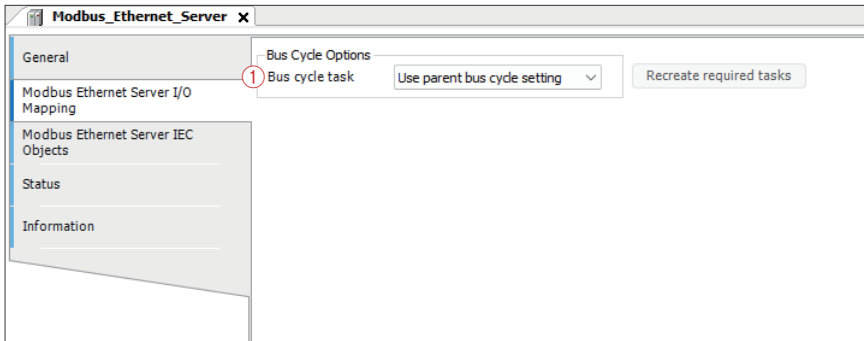
11.4 Modbus Ethernet Server

[General]



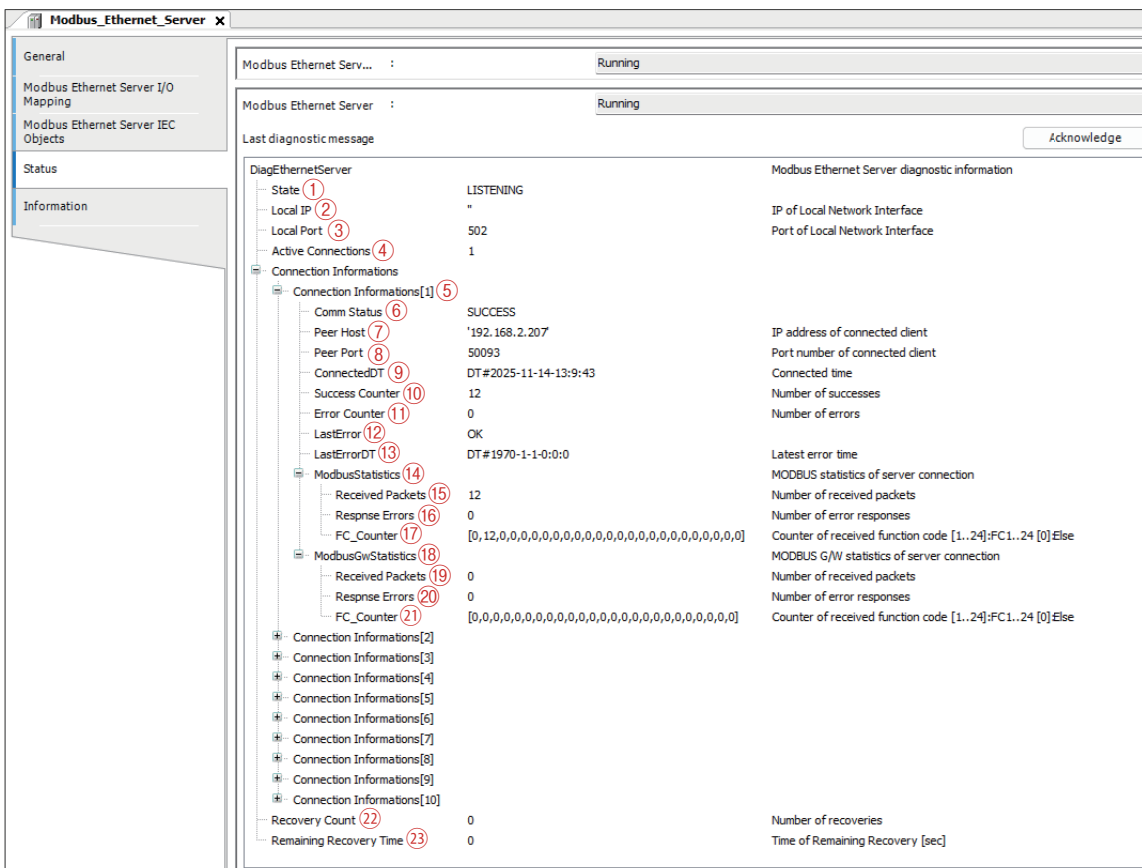
項目名	内容	設定範囲(): 初期値
① Port	自身の Port 番号を指定します。	0~65535 (502)
② Transmission	通信形式を指定します。	TCP/UDP (TCP)
③ Watchdog	クライアントの生存確認時間を設定します。 これを設定した場合は、時間内に MODBUS リクエストが到着しなければクライアントが離脱したものと判断して該当の回線を閉じます。	0~300 [sec] (30)
④ TCP Connections	TCP の場合の最大通信回線数を指定します。	1~10 (5)
⑤ Time Between Frames	次の MODBUS レスポンスフレームの送信に間隔時間を指定します。	0~5000 [ms] (0)
⑥ Activate Serial Gateway	シリアルゲートウェイ機能を指定します。	(無効)/有効
⑦ COM Port	COM ポートを指定します。 使用する機器がサポートするポート番号を指定します。	1~255 (1)
⑧ Baud Rate	通信速度を指定します。 使用する機器がサポートする通信速度を指定します。	4800, (9600), 19200, 38400, 57600, 115200
⑨ Data Bits	データビット長を指定します。	7, (8)
⑩ Parity	パリティビットを指定します。	NONE, ODD, (EVEN)
⑪ Stop Bits	ストップビットを指定します。	(1), 2
⑫ Transmission	通信形式	(RTU)

[Modbus Ethernet Server I/O Mapping]



項目名	内容	設定範囲(): 初期値
① Bus Cycle task	MODBUS 通信は、ここで指定したタスク周期で動作します。1バスサイクルで処理されるのはチャンネル毎に1リクエストです。	(Use parent bus cycle setting) または表示される Task から選択

[Status]

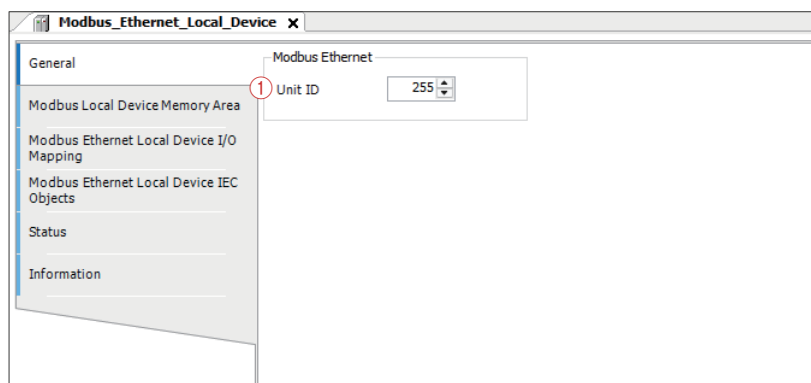


項目名	内容	備考
① State	現在状態が表示されます。 ET_MODULE_STATE	LISTENING: 正常動作しクライアントからの要求を受信待ちしている状態を示しています。
② Local IP	自身の IP Address (指定のある場合) が表示されます。	
③ Local Port	自身の Port 番号が表示されます。	
④ Active Connections	接続済みの通信回線数が表示されます。	
⑤ Connection Informations [N]	接続回線の現在情報	
⑥ Comm State	通信筐体が表示されます。 ET_SERVER_COMM_STATUS	
⑦ Peer Host	相手の IP Address が表示されます。	
⑧ Peer Port	相手の Port アドレスが表示されます。	
⑨ ConnectedDT	接続した日付時刻が表示されます。	

項目名	内容	備考
⑩ Success Counter	成功回数が表示されます。	
⑪ Error Counter	エラー回数が表示されます。	
⑫ LastError	最新のエラー状態が表示されます。 ET_MODBUS_ERROR	OK: エラーなし(初期状態も OK 表示となります)
⑬ LastErrorDT	最新のエラー発生日付時刻が表示されます。	'DT#1970-0-0-0:0:0' は初期状態
⑭ ModbusStatistics	Local Device 要求の統計情報	
⑮ Received Packets	受信パケット数	
⑯ Response Errors	エラー返信数	
⑰ FC_Counter	ファンクション・コード別受信数	配列 1-24: Function Code1-24 に対応 配列 0: それ以外の Function Code
⑱ ModbusGwStatistics	Gateway 要求の統計情報	
⑲ Received Packets	受信パケット数	
⑳ Response Errors	エラー返信数	
㉑ FC_Counter	ファンクション・コード別受信数	配列 1-24: Function Code1-24 に対応 配列 0: それ以外の Function Code
㉒ Recovery Count	試行中の回数が表示されます。	
㉓ Remaining Recovery Time	リカバリ時に開始残り時間が表示されます。	

11.5 Modbus Ethernet Local Device

[General]



項目名	内容	設定範囲(): 初期値
① Unit ID	自身のユニット番号(Server ID)を指定します。 重複しない番号を指定して下さい。	0~255 (255)

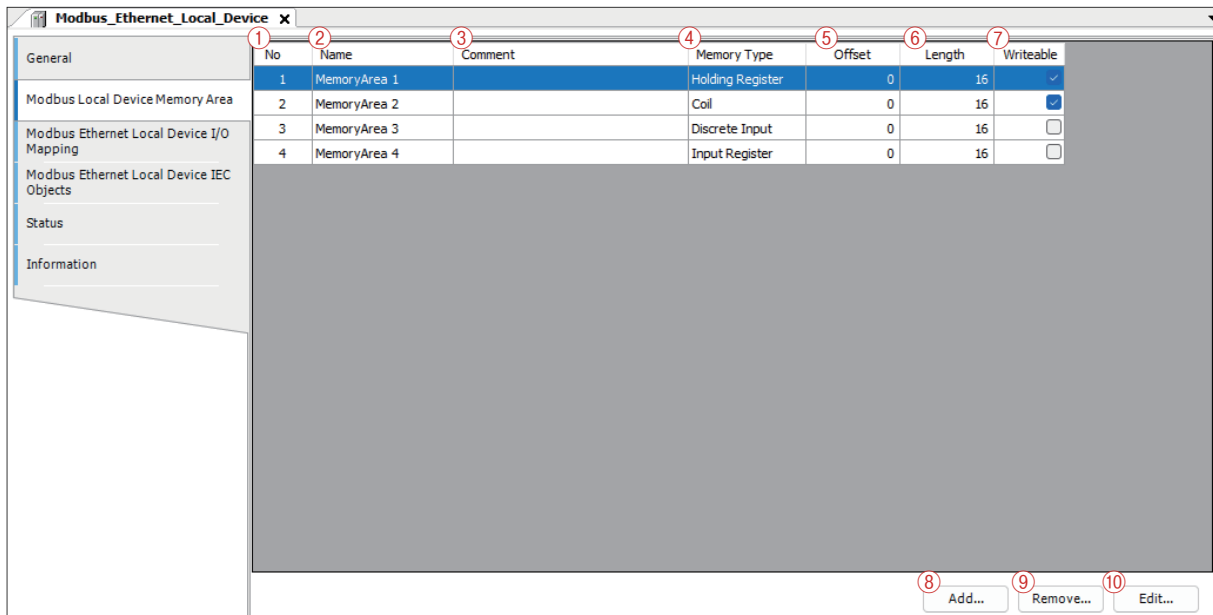
MODBUS Server Multi server ID simulation

複数の Local Device を登録することが可能です。それらはクライアントが発行する MODBUS リクエスト中の ServerID フィールド値により識別され、登録された Local Device の Unit ID (Server ID) と照合されます。一致した場合は I/O Mapping で指定された変数とデータ交換が行われます。これは複数の Server をシミュレーションできることを意味します。

RTU Client PATH THROUGH Gateway on MODBUS TCP / UDP Server

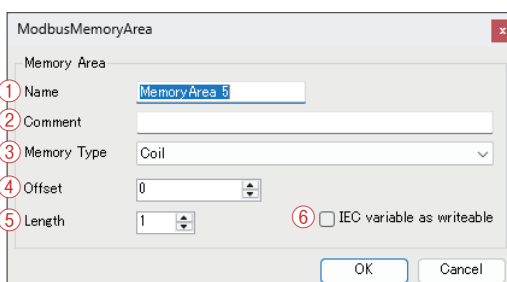
Serial Gateway 機能を有効にすると、Local Device に登録の無い Unit ID (Server ID) の MODBUS TCP / UDP リクエストを指定された Serial 回線に MODBUS RTU パケットとして通過させます。この時の MODBUS RTU Server レスポンスは MODBUS TCP / UDP のレスポンスとして返されます。タイムアウト時は “0x0A : GATEWAY PATH UNAVAILABLE” または “0x0B : GATEWAY TARGET DEVICE FAILED TO RESPOND” が返されます。

[Modbus Local Device Memory Area]



項目名	内容	備考
① No	連番が表示されます。	
② Name	名称が表示されます。	
③ Comment	注釈が表示されます。	
④ Memory Type	メモリー種別が表示されます。	
⑤ Offset	開始オフセットが表示されます。	
⑥ Length	領域の長さが表示されます。	
⑦ Writeable	I/O Mapping された変数に対する書き込み許可属性が表示されます。	
⑧ [Add...] ボタン	新しくチャンネルを追加します。	
⑨ [Remove...] ボタン	現在選択されているチャンネル定義を削除します。	複数行の選択も可能です。
⑩ [Edit...] ボタン	現在選択されているチャンネルを編集します。	複数行の選択はできません。 ここで内容を変更すると対象チャンネルの[I/O Mapping]情報はリセットされます。必要であれば再度指定して下さい。

[Modbus Local Device Memory Area – Add / Edit]



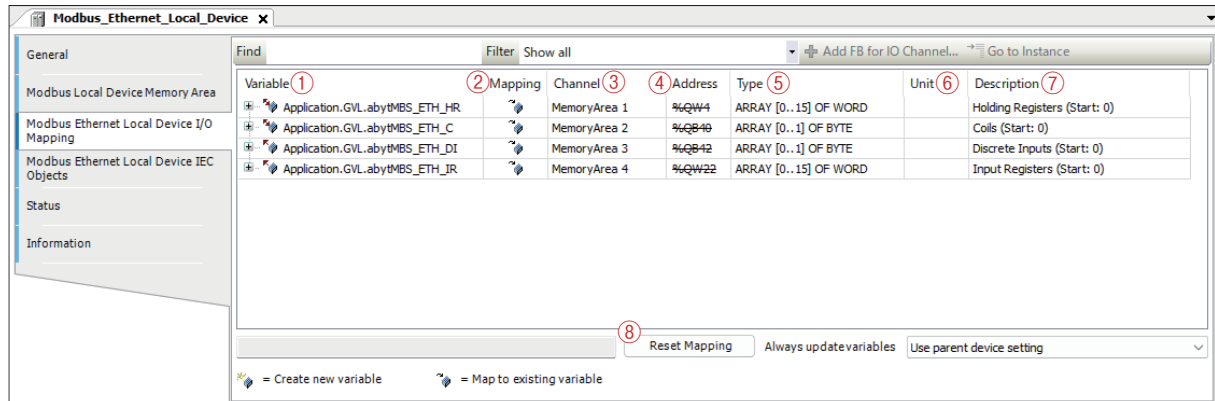
項目名	内容	設定範囲(): 初期値
① Name	名称を指定します。	使用可能な文字はシングルクォーテーションを除く半角英数字です。空欄や重複する名称は不可です。 最大 80 文字
② Comment	注釈を指定します。	使用可能な文字は半角英数字です。 最大 80 文字
③ Memory Type	アクセス方法を指定します。	(Coil) Discrete Input Input Register Holding Register
④ Offset	開始オフセットを指定します。 Offset+Length が 65536 を超えることはできません。	0~65535 (0)

C – CDS35 : MglDrvModbus3

項目名	内容	設定範囲(): 初期値
⑤ Length	領域長さを指定します。	1~65536 [Coil, Discrete Input] (1) 1~4096 [Input Register, Holding Register]
⑥ IEC variable as writeable	Coil, Holding Register に I/O Mapping された変数に対して書き込みを指定します。 許可するとアプリケーションからその変数に対して書き込みが可能となります。許可しなければ読み取りのみ可能となります。	(非許可) / 許可

- ・ 重複した領域は指定しないで下さい。指定されると正しい動作は保証されません。
- ・ Device の追加や削除を繰り返した後のプロジェクトで Length に 10000 を超える大きな領域を定義すると、定義直後の画面移動で CODESYS IDE がかなり長い時間無応答の状態になります。その場合は中断してプロジェクトを新規に作成しなおして下さい。

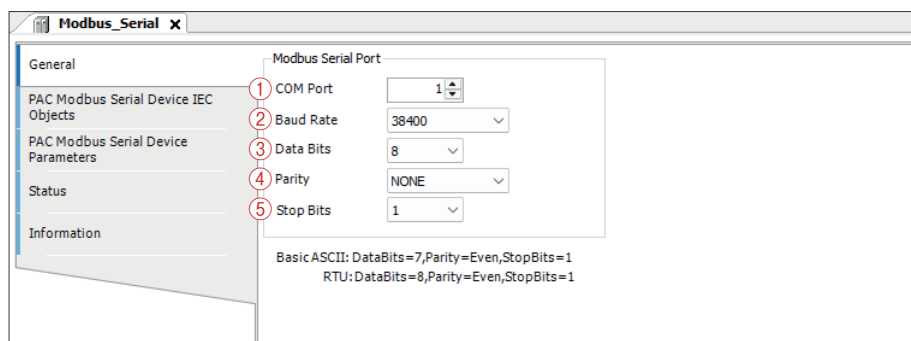
[Modbus Ethernet Local Device I/O Mapping]



項目名	内容	設定範囲(): 初期値
① Variable	割り当てる変数を指定します。	新規の名称か既存の変数を指定できます。 既存の変数を指定する場合は、その変数が Type で表示されている型と一致している必要があります。
② Mapping	新規変数か既存変数への割り当てかがアイコンで表示されます。	
③ Channel	チャンネル名称が表示されます。	
④ Address	割り当てられたアドレスが表示されます。 既存の変数を割り当てた場合は取り消し線が表示されます。	
⑤ Type	必要な変数の型が表示されます。	
⑥ Uint	工業単位が表示されます。	(未使用)
⑦ Description	注釈が表示されます。	
⑧ [Reset Mapping] ボタン	割付をリセットします。	

11.6 Modbus Serial

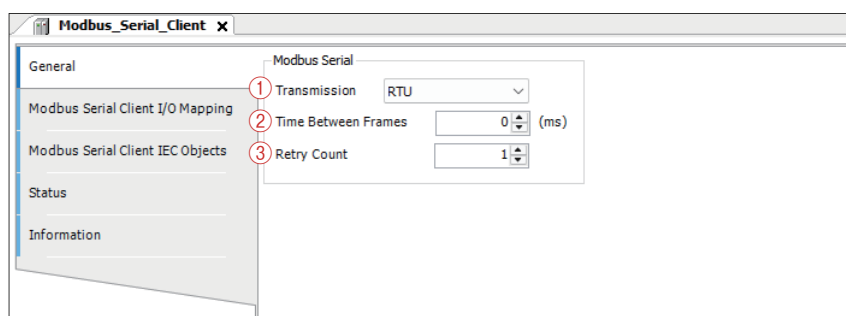
[General]



項目名	内容	設定範囲(): 初期値
① COM Port	COM ポートを指定します。 使用する機器がサポートするポート番号を指定します。	1~255 (1)
② Baud Rate	通信速度を指定します。 使用する機器がサポートする通信速度を指定します。	4800, (9600), 19200, 38400, 57600, 115200
③ Data Bits	データビット長を指定します。	7, (8)
④ Parity	パリティビットを指定します。	NONE, ODD, (EVEN)
⑤ Stop Bits	ストップビットを指定します。	(1), 2

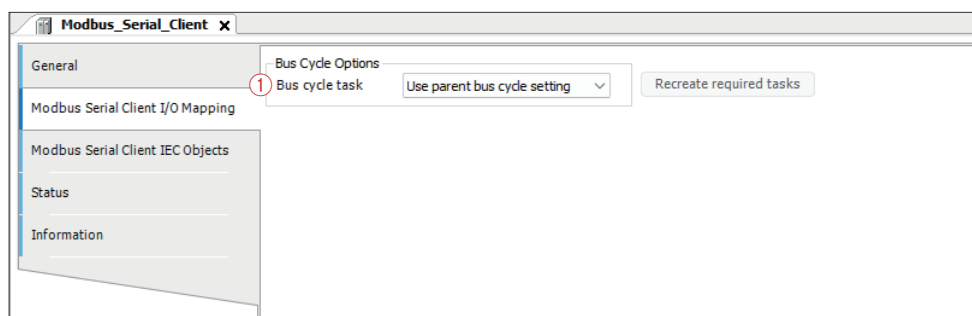
11.7 Modbus Serial Client

[General]



項目名	内容	設定範囲(): 初期値
① Transmission	通信形式を指定します。	ASCII/RTU (RTU)
② Time Between Frames	次の MODBUS リクエストフレームの送信に間隔時間を指定します。	0~5000 [ms] (0)
③ Retry Count	MODBUS 通信失敗時のリトライ回数を指定します。	0~10 (1)

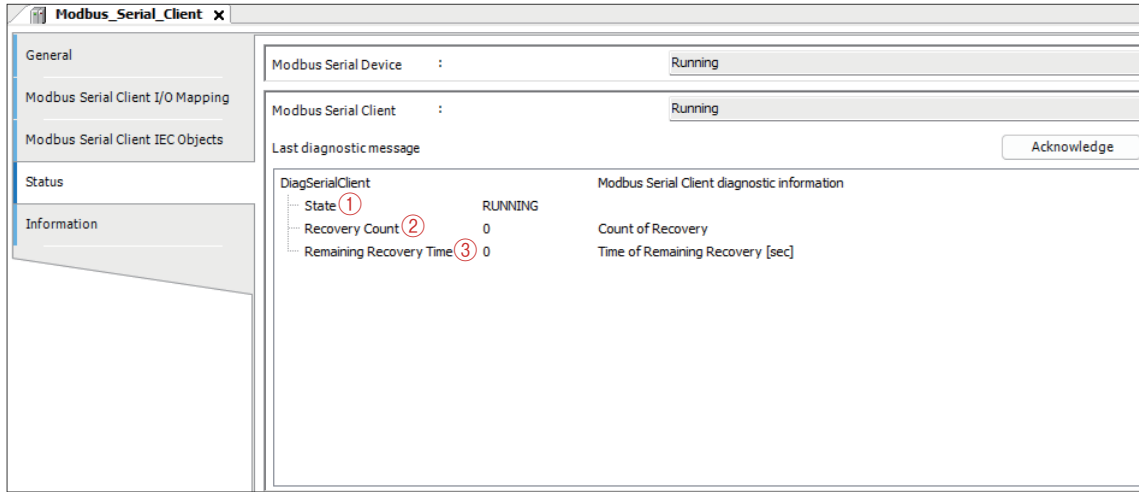
[Modbus Serial Client I/O Mapping]



C – CDS35 : MglDrvModbus3

項目名	内容	設定範囲(): 初期値
① Bus Cycle task	MODBUS 通信は、ここで指定したタスク周期で動作します。複数チャンネルの設定が設定されていても 1 バスサイクルで実行されるのは 1 チャンネルのみです。	(Use parent bus cycle setting) または表示される Task から選択

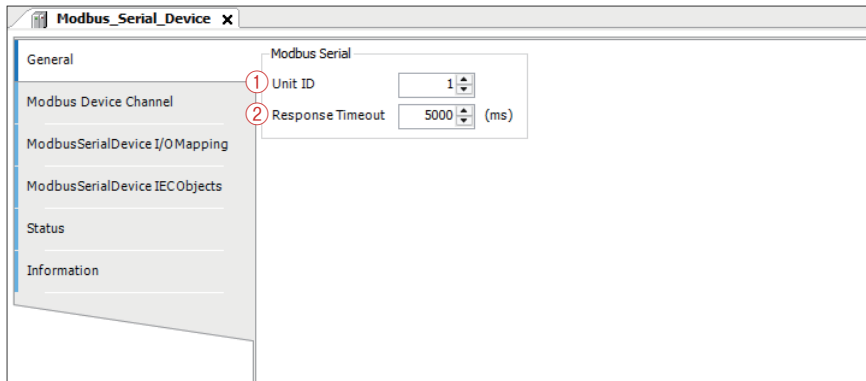
[Status]



項目名	内容	備考
① State	現在状態が表示されます。 ET_MODULE_STATE	RUNNING: 正常動作中であることを示しています。
② Recovery Count	試行中の回数が表示されます。	
③ Remaining Recovery Time	リカバリ時に開始残り時間が表示されます。	

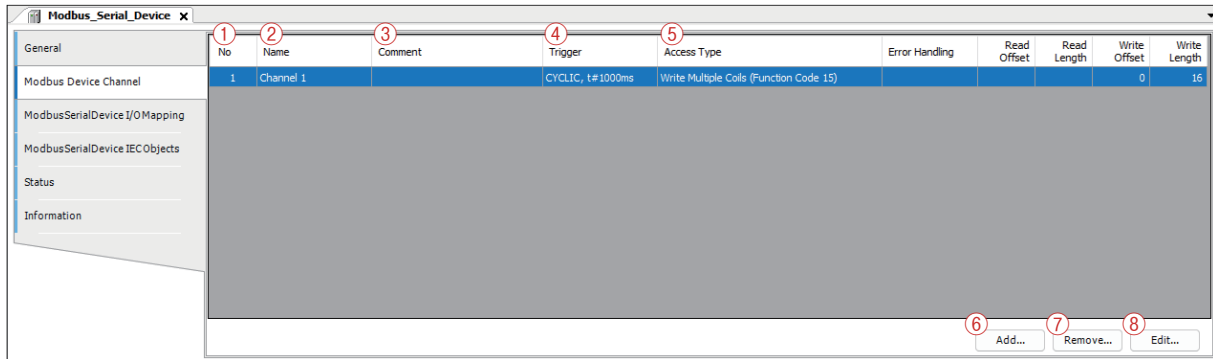
11.8 Modbus Serial Device

[General]



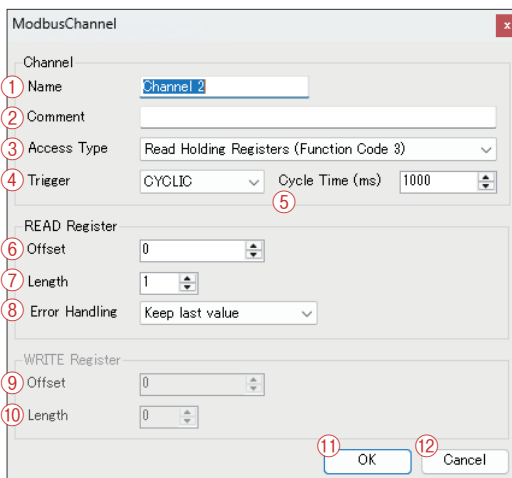
項目名	内容	設定範囲(): 初期値
① Unit ID	相手のユニット番号 (Server ID) を指定します。	0~255 (1)
② Response Timeout	レスポンスタイムアウト時間を指定します。	0~300000 [ms] (5000)

[Modbus Device Channel]



項目名	内容	備考
① No	連番が表示されます。	
② Name	名称が表示されます。	
③ Comment	注釈が表示されます。	
④ Trigger	トリガー内容が表示されます。	
⑤ Access Type	アクセス方法が表示されます。	
⑥ [Add...] ボタン	新しくチャンネルを追加します。	
⑦ [Remove...] ボタン	現在選択されているチャンネル定義を削除します。	複数行の選択も可能です。
⑧ [Edit...] ボタン	現在選択されているチャンネルを編集します。	複数行の選択はできません。 ここで内容を変更すると対象チャンネルの [I/O Mapping] 情報はリセットされます。必要であれば再度指定して下さい。

[Modbus Device Channel – Add/Edit]

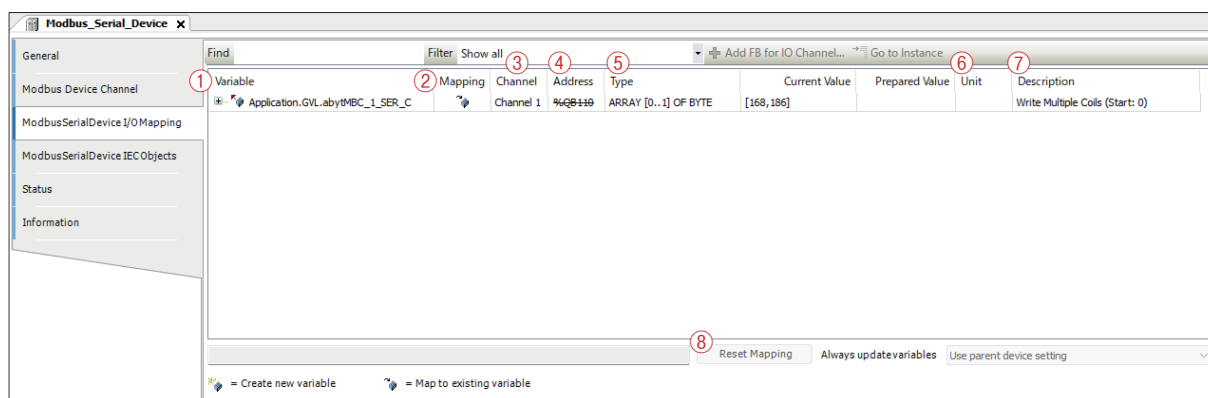


項目名	内容	設定範囲(): 初期値
[Channel]		
① Name	名称を指定します。	使用可能な文字はシングルクォーテーションを除く半角英数字です。空欄や重複する名称は不可です。 最大 80 文字
② Comment	注釈を指定します。	使用可能な文字は半角英数字です。 最大 80 文字
③ Access Type	アクセス方法を指定します。	Read Coils (Function Code 1) Read Discrete Inputs (Function Code 2) (Read Holding Registers (Function Code 3)) Read Input Registers (Function Code 4) Write Single Coil (Function Code 5) Write Single Register (Function Code 6) Write Multiple Coils (Function Code 15) Write Multiple Registers (Function Code 16) Read/Write Multiple Registers (Function Code 23)

C – CDS35 : MglDrvModbus3

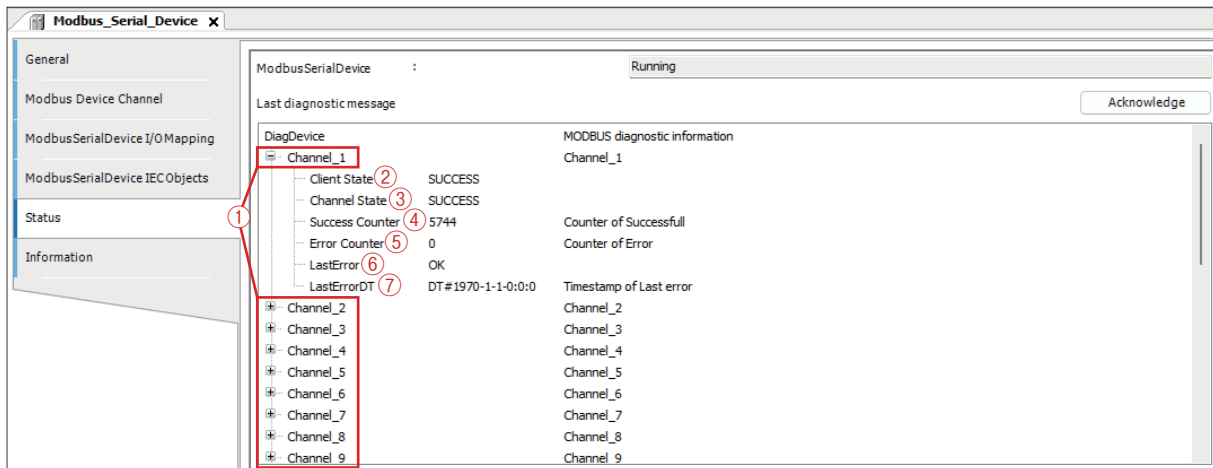
項目名	内容	設定範囲(): 初期値
④ Trigger	トリガーを指定します。	APPLICATION: プログラムコードで起動 (CYCLIC): 定周期で起動 RISING_EDGE: 変数値の立ち上がりで起動
⑤ Cycle Time	Trigger=CYCLIC 時に周期を指定します。	0~300000 [ms] (1000)
[READ Register]		
⑥ Offset	オフセットを指定します。 Offset+Length が 65536 を超えることはできません。	0~65535 (0)
⑦ Length	長さを指定します。	1~2000 [Function Code 1, 2] (1) 1~125 [Function Code 3, 4, 23]
⑧ Error Handling	異常時の処理を指定します。	(Keep last value) Set to zero
[WRITE Register]		
⑨ Offset	オフセットを指定します。 Offset+Length が 65536 を超えることはできません。	0~65535 (0)
⑩ Length	長さを指定します。	1 [Function Code 5, 6] (1) 1~1968 [Function Code 15] 1~123 [Function Code 16] 1~121 [Function Code 23]
⑪ [OK] ボタン	設定を確定します。	
⑫ [Cancel] ボタン	設定を破棄します。	

[Modbus Serial Device I/O Mapping]



項目名	内容	設定範囲(): 初期値
① Variable	割り当てる変数を指定します。	新規の名称か既存の変数を指定できます。 既存の変数を指定する場合は、その変数が Type で表示されている型と一致している必要 があります。
② Mapping	新規変数か既存変数への割り当てかがアイコンで表示 されます。	
③ Channel	チャンネル名称が表示されます。	
④ Address	割り当てられたアドレスが表示されます。 既存の変数を割り当てた場合は取り消し線が表示され ます。	
⑤ Type	必要な変数の型が表示されます。	
⑥ Uint	工業単位が表示されます。	(未使用)
⑦ Description	注釈が表示されます。	
⑧ [Reset Mapping] ボタン	割付をリセットします。	

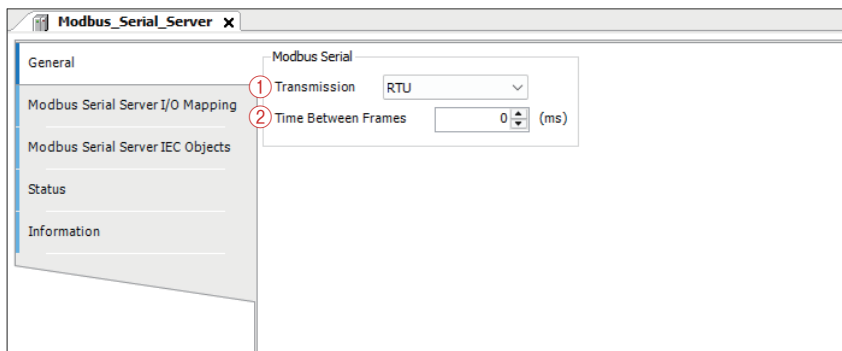
[Status]



項目名	内容	備考
① Channel_N	チャンネル識別子が表示されます。	
② Client State	クライアント状態が表示されます。 ET_CLIENT_STATUS	TRIGGERD: トリガー状態を示しています。 SUCCESS: MODBUS 通信が正常に完了したことを示しています。
③ Channel State	チャンネル状態が表示されます。 ET_MODBUS_CHANNEL_STATUS	UNCONFIGURED: 未設定 CONFIGURED: 設定状態 SUCCESS: チャンネルが正常に処理されていることを示しています。 ERROR: 異常状態
④ Success Counter	成功カウント数が表示されます。	
⑤ Error Counter	エラーカウント数が表示されます。	
⑥ LastError	最新のエラー状態が表示されます。 ET_MODBUS_ERROR	OK: エラーなし(初期状態も OK 表示となります)
⑦ LastErrorDT	最新のエラー発生日付時刻が表示されます。	'DT#1970-0-0-0:0:0' は初期状態

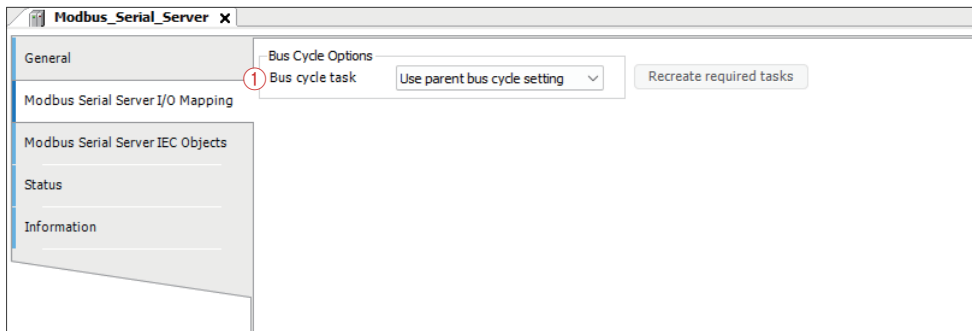
11.9 Modbus Serial Server

[General]



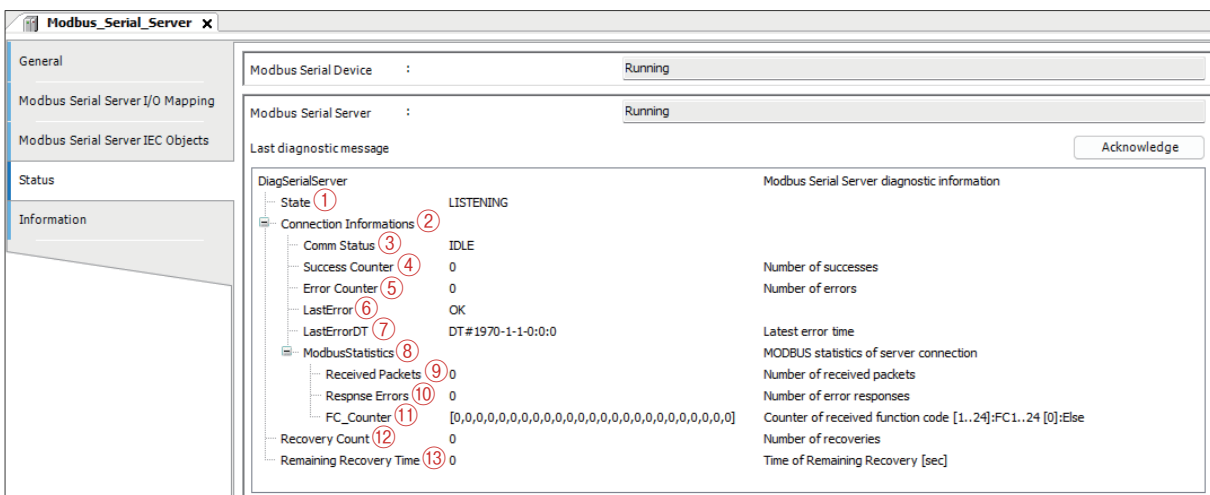
項目名	内容	設定範囲(): 初期値
① Transmission	通信形式を指定します。	ASCII/RTU (RTU)
② Time Between Frames	次の MODBUS レスポンスフレームの送信に間隔時間を指定します。	0~5000 [ms] (0)

[Modbus Serial Server I/O Mapping]



項目名	内容	設定範囲(): 初期値
① Bus Cycle task	MODBUS 通信は、ここで指定したタスク周期で動作します。1 バスサイクルで処理されるのはチャンネル毎に 1 リクエストです。	(Use parent bus cycle setting) または表示される Task から選択

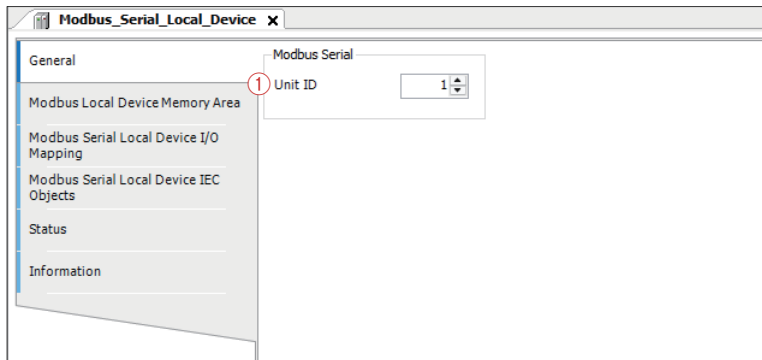
[Status]



項目名	内容	備考
① State	現在状態が表示されます。 ET_MODULE_STATE	LISTENING: 正常動作しクライアントからの要求を受信待ちしている状態を示しています。
② Connection Informations	接続回線の現在情報	
③ Comm State	通信筐体が表示されます。 ET_SERVER_COMM_STATUS	
④ Success Counter	成功回数が表示されます。	
⑤ Error Counter	エラー回数が表示されます。	
⑥ LastError	最新のエラー状態が表示されます。 ET_MODBUS_ERROR	OK: エラーなし(初期状態も OK 表示となります)
⑦ LastErrorDT	最新のエラー発生日付時刻が表示されます。	'DT#1970-0-0-0:0:0' は初期状態
⑧ ModbusStatistics	Local Device 要求の統計情報	
⑨ Received Packets	受信パケット数	
⑩ Response Errors	エラー返信数	
⑪ FC_Counter	ファンクション・コード別受信数	配列 1-24: Function Code1-24 に対応 配列 0: それ以外の Function Code
⑫ Recovery Count	試行中の回数が表示されます。	
⑬ Remaining Recovery Time	リカバリ時に開始残り時間が表示されます。	

11.10 Modbus Serial Local Device

[General]

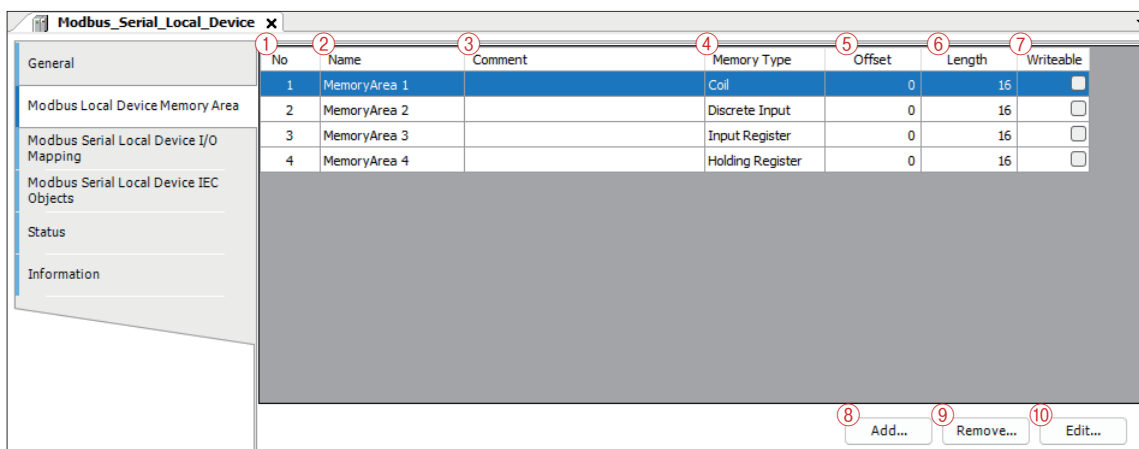


項目名	内容	設定範囲(): 初期値
① Unit ID	自身のユニット番号(Server ID)を指定します。 重複しない番号を指定して下さい。	0~255 (1)

MODBUS Server Multi server ID simulation

複数の Local Device を登録することが可能です。それらはクライアントが発行する MODBUS リクエスト中の Server ID フィールド値により識別され、登録された Local Device の Unit ID (Server ID) と照合されます。一致した場合は I/O Mapping で指定された変数とデータ交換が行われます。これは複数の Server をシミュレーションできることを意味します。

[Modbus Local Device Memory Area]



項目名	内容	設定範囲(): 初期値
① No	連番が表示されます。	
② Name	名称が表示されます。	
③ Comment	注釈が表示されます。	
④ Memory Type	メモリー種別が表示されます。	
⑤ Offset	開始オフセットが表示されます。	
⑥ Length	領域の長さが表示されます。	
⑦ Writeable	I/O Mapping された変数に対する書き込み許可属性が表示されます。	
⑧ [Add...] ボタン	新しくチャンネルを追加します。	
⑨ [Remove...] ボタン	現在選択されているチャンネル定義を削除します。	複数行の選択も可能です。
⑩ [Edit...] ボタン	現在選択されているチャンネルを編集します。	複数行の選択はできません。 ここで内容を変更すると対象チャンネルの [I/O Mapping] 情報はリセットされます。必要であれば再度指定して下さい。

[Modbus Local Device Memory Area – Add/Edit]

項目名	内容	設定範囲(): 初期値
① Name	名称を指定します。	使用可能な文字はシングルクォーテーションを除く半角英数字です。空欄や重複する名称は不可です。 最大 80 文字
② Comment	注釈を指定します。	使用可能な文字は半角英数字です。 最大 80 文字
③ Memory Type	アクセス方法を指定します。	(Coil) Discrete Input Input Register Holding Register
④ Offset	開始オフセットを指定します。 Offset+Length が 65536 を超えることはできません。	0~65535 (0)
⑤ Length	領域長さを指定します。	1~65536 [Coil, Discrete Input](1) 1~4096 [Input Register, Holding Register]
⑥ IEC variable as writeable	Coil, Holding Register に I/O Mapping された変数に対して書き込みを指定します。 許可するとアプリケーションからその変数に対して書き込みが可能となります。許可しなければ読み取りのみ可能となります。	(非許可) / 許可

重複した領域は指定しないで下さい。指定されると正しい動作は保証されません。

[Modbus Serial Local Device I/O Mapping]

Variable	Mapping	Channel	Address	Type	Unit	Description
Application.GVL.abytMBS_SER_C	[Icon]	MemoryArea 1	%I008	ARRAY [0..1] OF BYTE		Coils (Start: 0)
Application.GVL.abytMBS_SER_DI	[Icon]	MemoryArea 2	%Q0114	ARRAY [0..1] OF BYTE		Discrete Inputs (Start: 0)
Application.GVL.abytMBS_SER_IR	[Icon]	MemoryArea 3	%Q0458	ARRAY [0..15] OF WORD		Input Registers (Start: 0)
Application.GVL.abytMBS_SER_HR	[Icon]	MemoryArea 4	%I020	ARRAY [0..15] OF WORD		Holding Registers (Start: 0)

項目名	内容	設定範囲(): 初期値
① Variable	割り当てる変数を指定します。	新規の名称か既存の変数を指定できます。 既存の変数を指定する場合は、その変数が Type で表示されている型と一致している必要があります。
② Mapping	新規変数か既存変数への割り当てかがアイコンで表示されます。	
③ Channel	チャンネル名称が表示されます。	
④ Address	割り当てられたアドレスが表示されます。 既存の変数を割り当てた場合は取り消し線が表示されます。	
⑤ Type	必要な変数の型が表示されます。	
⑥ Uint	工業単位が表示されます。	(未使用)
⑦ Description	注釈が表示されます。	
⑧ [Reset Mapping] ボタン	割付をリセットします。	

12. 機能の解説

12.1 Bus Cycle Task

本ドライバーの MODBUS Client/Server の処理は指定した「Bus Cycle Task」の周期で処理されます。特に注意が必要な内容を記載します。

時間判定

時間にかかわる設定（例えば Modbus Ethernet Device の Cycle Time [ModbusTM]）の判定は「Bus Cycle Task」の周期 [TaskTM] 毎に行われます。これは [ModbusTM] < [TaskTM] である場合は [TaskTM] 経過後に判定されます。

「Bus Cycle Task」の処理時間

処理時間 = Task 本来の処理時間 + MODBUS ドライバー処理時間

このように「Bus Cycle Task」は Task 本来の処理時間より長くなることに注意して下さい。

12.2 Client のトリガー

ここではトリガーについて説明します。

トリガーには 'APPLICATION', 'CYCLIC', 'RISING EDGE' があります。

MODBUS 要求はこれらトリガー条件の成立で発行されます。

ただし、これらトリガーのタイミングで MODBUS への要求が即時に送信されるわけではありません。

MODBUS 通信処理がバスサイクル毎に行われることや処理中の要求がある場合は完了するまで他の要求は保留（遅延）されます。

それぞれの使用例を以下に示します。

[APPLICATION]

アプリケーションでトリガーとするタイミングでメソッドを呼び出します。

このメソッドは MODBUS 要求からレスポンス受信までを管理できます。

状態遷移：

	トリガー-ON→	要求受付→	MODBUS REQUEST発行→	結果 (成功/失敗) →	トリガー-OFF
xTrigger	TRUE				FALSE
xBusy		TRUE		FALSE	↓
xDone				(TRUE)	FALSE
xError				(TRUE)	FALSE

オブジェクト：Modbus Ethernet Client, Modbus Serial Client

```

METHOD PUBLIC ApplicationTrigger
VAR_INPUT
    xTrigger      : BOOL;           // Rising edge:Action start, Falling edge:Reset output status
    uiDeviceIndex : UINT;           // 1..
    uiChannelIndex : UINT;         // 1..
END_VAR
VAR_OUTPUT
    xBusy         : BOOL;           // Is TRUE as long as a request is in progress
    xDone         : BOOL;           // Is TRUE if a request finished successfully
    xError        : BOOL;           // Is TRUE if a request finished unsuccessfully
    eError        : ET_MODBUS_ERROR; // Specifies the current error
END_VAR

```

使用例：

```

VAR
    xTEST : BOOL;           // これをFALSE→TRUE にすることで開始され、完了するとFALSEになります
    _xTrigger, _xBusy, _xDone, _xError : BOOL;
    _iTrigger : INT := 0;
    _eError : ET_MODBUS_ERROR;
END_VAR
// APPLICATION TRIGGER SAMPLE
IF xTEST THEN
    CASE _iTrigger OF
        0:
            GVL.abyDATA[0] := GVL.abyDATA[0] + 1;    // この例ではIO Mappingした変数をここで更新
            _xTrigger := TRUE;
            _iTrigger := 1;
        1:
            IF _xDone OR _xError THEN
                _xTrigger := FALSE;
                _iTrigger := 2;
            END_IF
        2:
            IF NOT(_xDone OR _xError) THEN
                xTEST := FALSE;
                _iTrigger := 0;
            END_IF
    END_CASE
    Modbus_Ethernet_Client_1.ApplicationTrigger(xTrigger:=_xTrigger,
        uiDeviceIndex := 2,           // 2 番目のDevice (Modbus_Ethernet_Device_1)
        uiChannelIndex := 4,         // 4 番Channel
        xBusy=>_xBusy, xDone=>_xDone, xError=>_xError, eError=>_eError);
END_IF

```

xTrigger は xDone または xError が TRUE となるまで TRUE を維持する必要があります。xBusy=FALSE 中に xTrigger を FALSE にすると処理をリセット（中断）します。トリガー要求の完了は xTrigger を FALSE にした後に xDone と xError が FALSE であることを確認します。

[CYCLIC]

指定の周期で自動的にトリガーされるのでアプリケーションでトリガーの必要はありません。

[RISING EDGE]

トリガー変数として I/O Mapping した変数値の立ち上がりでトリガーとします。トリガーのタイミングでアプリケーションにて変数値を FALSE→TRUE にします。この変数値の評価はバスサイクルタスクで行いますので、TRUE としたサイクルでトリガーを記憶するので、次のサイクルで FALSE にすることが可能です。

使用例：

```

VAR
    xTEST : BOOL;           // これをFALSE→TRUE にすることで開始され、完了するとFALSEになります
END_VAR
// EDGE TRIGGER SAMPLE
IF xTEST THEN
    GVL.abyDATA [0] := GVL.abyDATA [0] + 1; // この例ではIO Mappingした変数をここで更新
    GVL.xTRIGGER := TRUE; // IO Mappingしたトリガー変数をここでTRUEに更新（立ち上がりエッジ）
    xTEST := FALSE;
ELSE
    GVL.xTRIGGER := FALSE; // IO Mappingしたトリガー変数をここでFALSEに更新
END_IF

```

12.3 API で指定する DeviceIndex, ChannelIndex について

関数やメソッドへ渡す uiDeviceIndex, uiChannelIndex は次の数値を指定します。

メソッドの例：

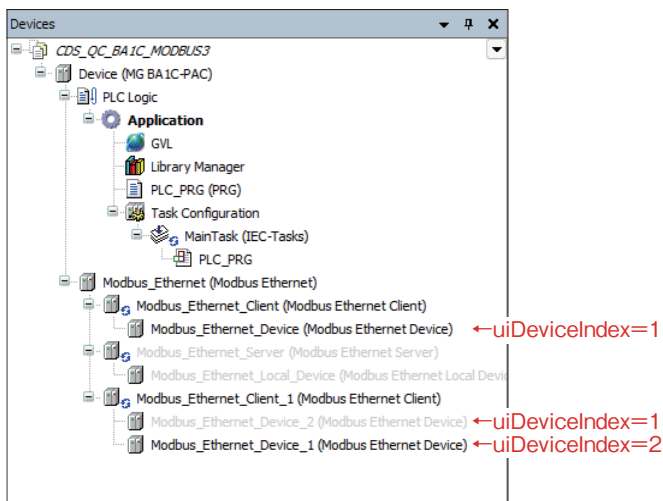
```
Modbus_Ethernet_Client_1.ApplicationTrigger (xTrigger:=_xTrigger,
      uiDeviceIndex:=2,           // 2番目のDevice (Modbus_Ethernet_Device_1)
      uiChannelIndex:=4,         // 4番Channel
      xBusy=>_xBusy, xDone=>_xDone, xError=>_xError, eError=>_eError) ;
```

uiDeviceIndex：

DeviceTree に表示される順で Client 以下に定義した無効 (Disable) 化したものも含めた 1 から始まる連番

uiChannelIndex：

Channel 設定画面に表示される No に示される 1 から始まる連番



12.4 Server の Output, Holding Register の書き込み

ここでは Server の Output, Holding Register 領域に割り当てた変数の書き込みについて説明します。

通常 Server のこれら領域はプログラムから入力 (読み取り専用) 変数として宣言されます。

[ModbusMemoryArea] 定義画面の [IEC variable as writeable] を有効にすることでプログラムから変数値を更新可能にできます。

12.5 割り当てた変数の初期値 (前回最終値)

ここでは I/O Mapping で割り当てた変数の初期値について説明します。

割り当てた変数は RETAIN 宣言することで起動時に前回の最終値から開始することができます。

ただし、MODBUS クライアントにおいて、RETAIN 宣言した変数の値は復元されますが、自動的に接続相手のデバイス (サーバー) へ初期値 (変数値) の書き込みは行われません。

特に、起動直後は IO Mapping された変数であっても変数の値と相手デバイスの現在値とは同期 (一致) しているとは限らないことに注意して下さい。

13. 使用方法

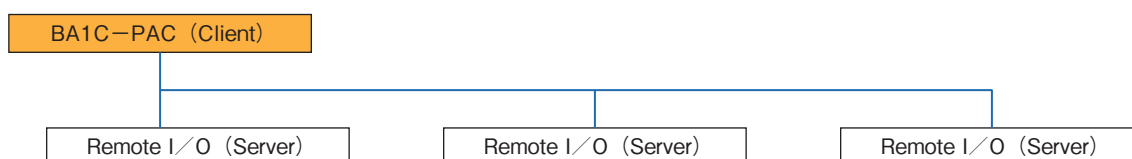
本製品の使用方法を説明します。

13.1 MODBUS Ethernet CLIENT

ここでは MODBUS Client 機能の使用方法を解説します。

MODBUS Ethernet CLIENT 通信の準備

MODBUS server device 機器と同一ネットワークに接続して下さい。



CODESYS プロジェクトの準備

MODBUS Ethernet 通信を介して device と接続する手順を説明します。

■接続する Server 機器

device-1

IP アドレス : 192.168.1.10

Port : 502

Server ID : 255

● 1 秒周期で Holding Register 40001 を読み変数 awMBC_1_40001 に値を書き込む

● 変数 bMBC_1_40002_T の立ち上がりで変数 awMBC_1_40002 の値を Holding Register 40002 に書き込む

device-2

IP アドレス : 192.168.1.11

Port : 502

Server ID : 255

device-3

IP アドレス : 192.168.1.12

Port : 502

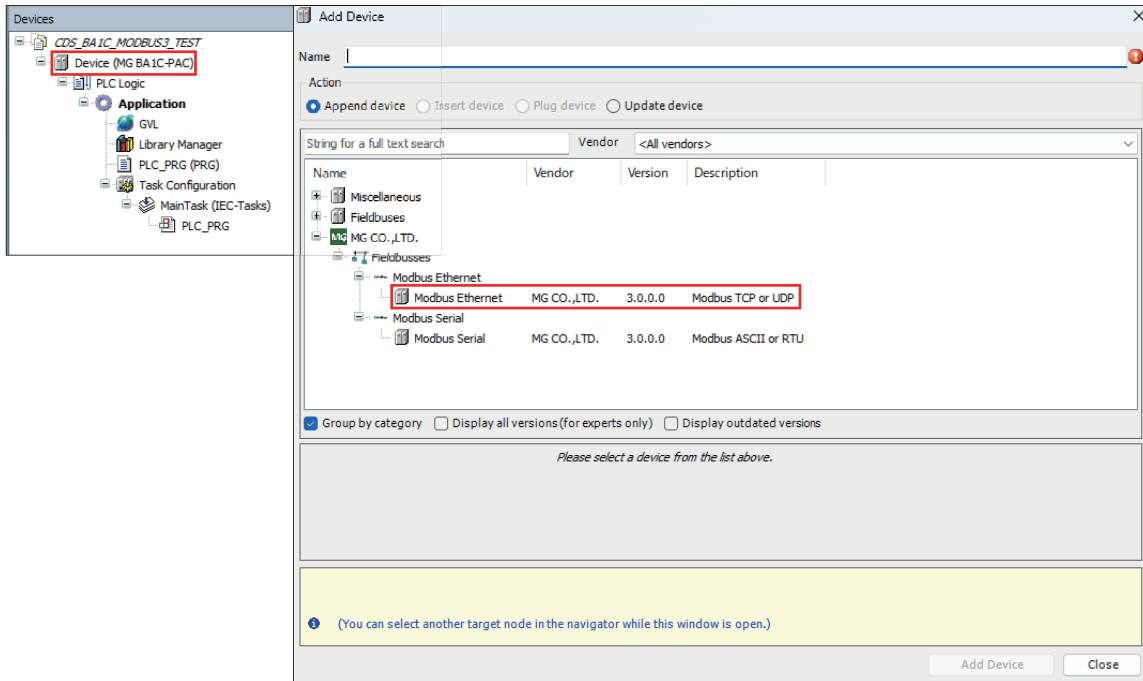
Server ID : 255

■このプロジェクトを動作させる Client 機器 (BA1C-PAC)

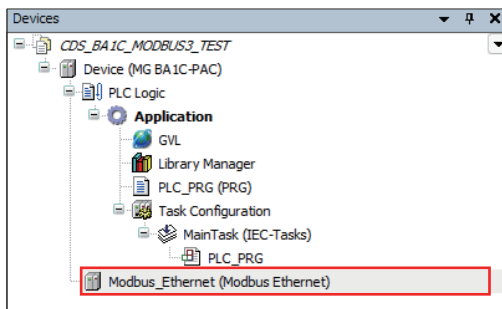
LAN1 IP アドレス : 192.168.1.200 (server 機器は LAN1 側に接続)

設定手順

- (1) MODBUS Client を必要とする CODESYS プロジェクトを開きます。
- (2) 「Modbus Ethernet」の追加
 デバイスツリーにて「導入先のデバイスを選択」－「右クリック」－「Add Device…」を選択します。
 表示された画面で「MG CO.,LTD.」(無ければ「Miscellaneous」) から「Modbus Ethernet」フォルダ内の「Modbus Ethernet」を選択します。



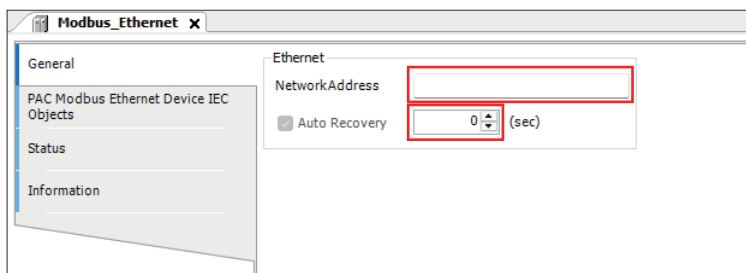
- (3) 「Modbus Ethernet」のパラメータ設定
 デバイスツリーにて「追加された「Modbus_Ethernet」をダブルクリック」します。



[General] タブ

Network Address : 空欄

Auto Recovery : 0

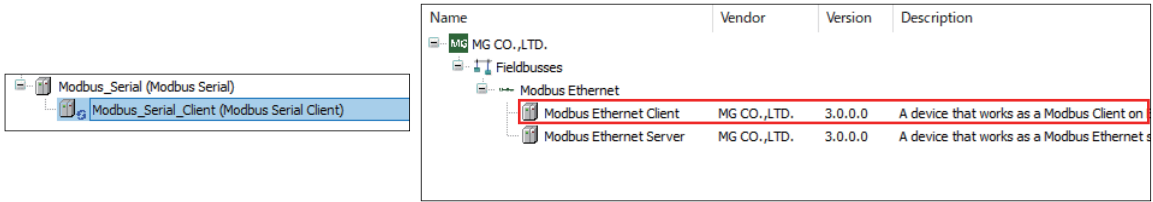


C—CDS35: MglDrvModbus3

(4) device-1 に接続する「Modbus Ethernet Client」の追加

デバイスツリーにて「追加された 'Modbus_Ethernet' インスタンスを選択」－「右クリック」－「Add Device…」を選択します。

表示された画面で「Modbus Ethernet」フォルダ内の「Modbus Ethernet Client」を選択します。



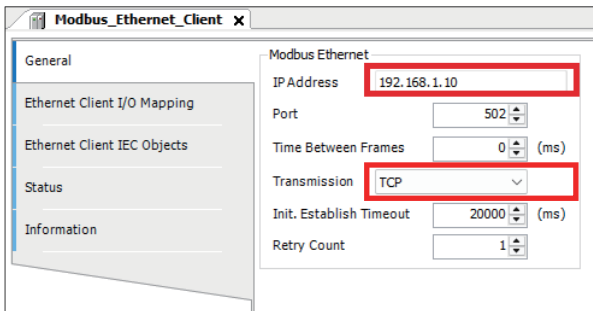
(5) 「Modbus Ethernet Client」のパラメータ設定

デバイスツリーにて「追加された 'Modbus_Ethernet_Client' をダブルクリック」します。各タブにて次の指定を行います。

「General」タブ

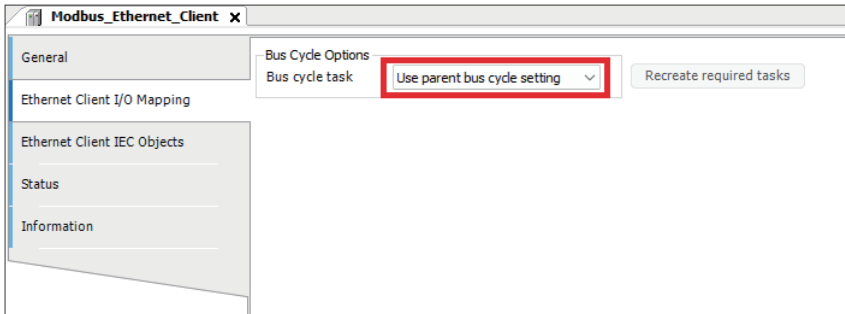
IP Address : '192.168.1.10'

Transmission : TCP (MODBUS/TCP の場合) または UDP (MODBUS/UDP の場合)



「Ethernet Client I/O Mapping」タブ

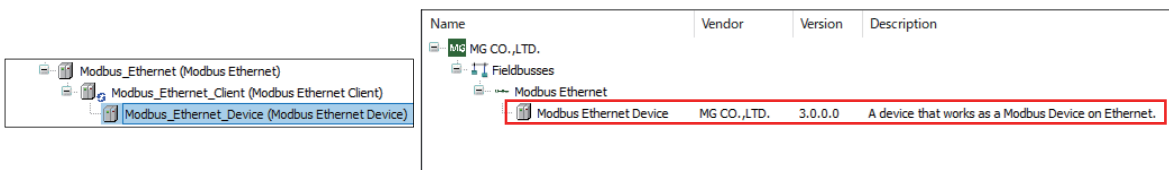
Bus cycle task : 'Use parent bus cycle setting'



(6) device への要求を定義する「Modbus Ethernet Device」の追加

デバイスツリーにて「追加された 'Modbus_Ethernet_Client' インスタンスを選択」－「右クリック」－「Add Device…」を選択します。

表示された画面で「Modbus Ethernet」フォルダ内の「Modbus Ethernet Device」を選択します。



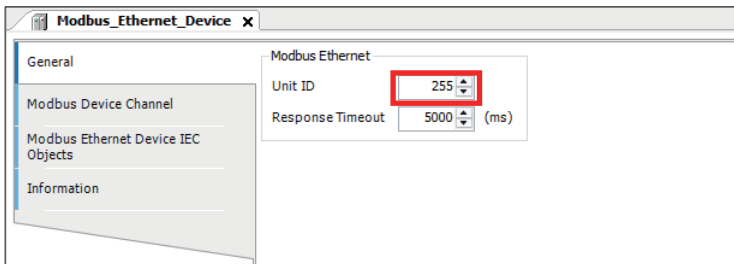
C—CDS35: MglDrvModbus3

(7) 「Modbus Ethernet Device」のパラメータ設定

デバイスツリーにて「追加された ‘Modbus_Ethernet_Device’ をダブルクリック」します。
各タブにて次の指定を行います。

[General] タブ

Unit ID : ‘255’

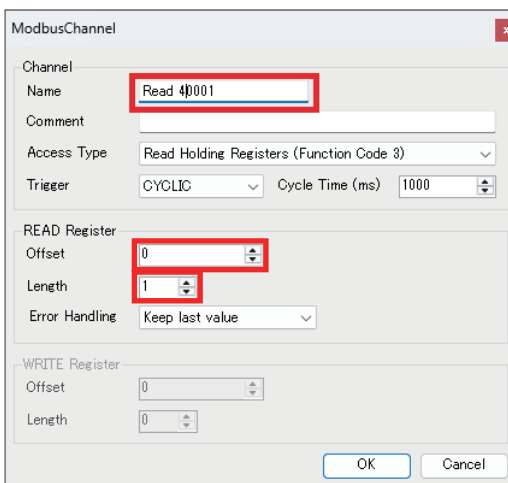


[Modbus Device Channel] タブ



[Add...] ボタンを押し 2つの新しいチャンネルを追加します。

チャンネル 1 : Cyclic, Read Holding Register 40001 (Offset : 0, Length : 1)



C—CDS35: MgloDrvModbus3

チャンネル 2 : Rising Edge, Write Holding Register 40002 (Offset : 1, Length : 1)

ModbusChannel

Channel

Name: Write 40002

Comment:

Access Type: Write Single Register (Function Code 6)

Trigger: RISING_EDGE Cycle Time (ms): 1000

READ Register

Offset: 0

Length: 0

Error Handling: Keep last value

WRITE Register

Offset: 1

Length: 1

OK Cancel

[Modbus Ethernet Device I/O Mapping] タブ

Variable 欄をダブルクリックして変数を割り当てます。

‘awMBC_1_40001’

‘awMBC_1_40002’ , ‘bMBC_1_40002_T’

Modbus Ethernet Device

General

Modbus Device Channel

Modbus Ethernet Device I/O Mapping

Modbus Ethernet Device IEC Objects

Information

Find Filter Show all Add FB for IO Channel... Go to Instance

Variable	Mapping	Channel	Address	Type	Unit	Description
awMBC_1_40001	Read 40001	Read 40001	%IW0	ARRAY [0..0] OF WORD		Read Holding Registers (Start: 0)
awMBC_1_40002	Write 40002	Write 40002	%QW0	ARRAY [0..0] OF WORD		Write Single Register (Start: 1)
bMBC_1_40002_T	Write 40002	Write 40002	%QX2.0	BIT		Trigger Variable

Reset Mapping Always update variables

Create new variable Map to existing variable

(8) 手順 (4) に戻り、同様の手順で ‘device-2’ と ‘device-3’ を追加します。

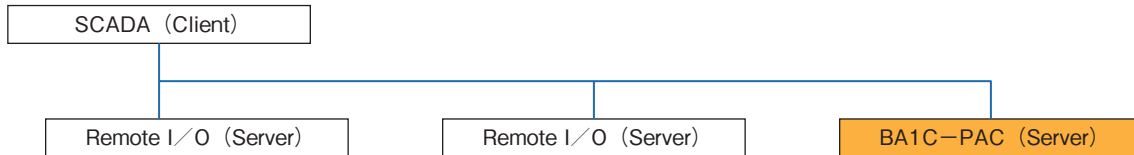
ここまでの設定を行い [Build] でエラーが無ければターゲットで実行できます。

13.2 MODBUS Ethernet SERVER

ここでは MODBUS Server 機能の使用方法を解説します。

MODBUS Ethernet SERVER 通信の準備

MODBUS client 機器と同一ネットワークに接続して下さい。



CODESYS プロジェクトの準備

MODBUS Ethernet 通信を介して client と接続する手順を説明します。

■接続する Client 機器

IP アドレス : 192.168.1.10

■このプロジェクトを動作させる Server 機器 (BA1C-PAC)

LAN1 IP アドレス : 192.168.1.200 (client 機器は LAN1 側に接続)

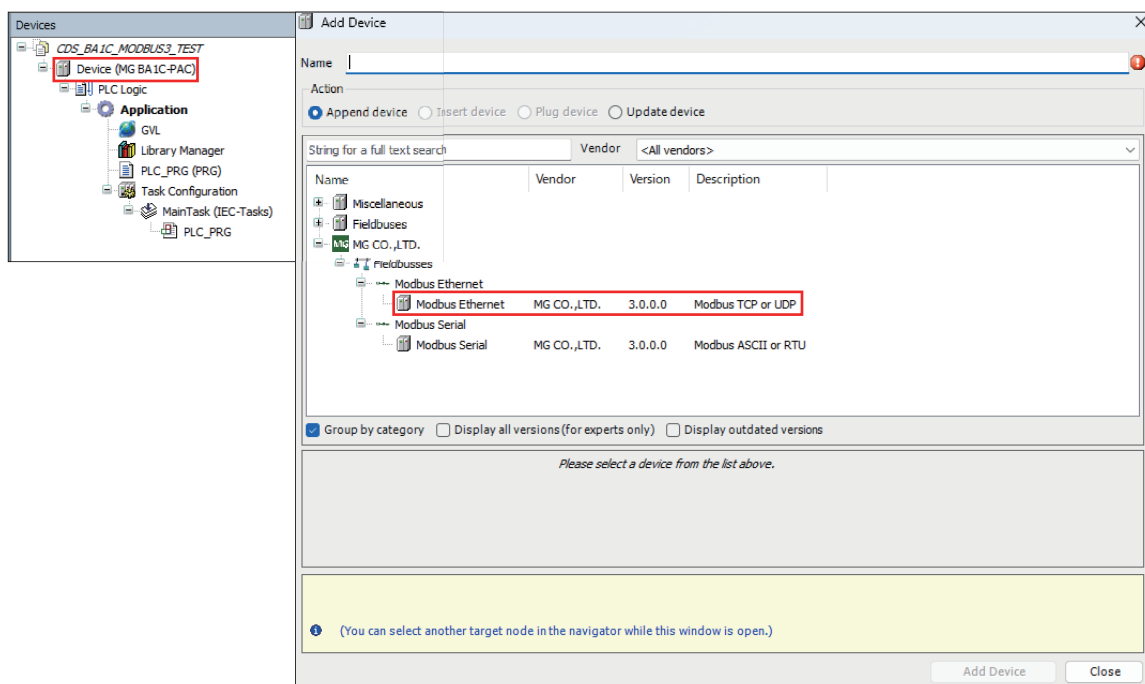
Unit ID : 255

Coil 00001 ~ 00016

Holding Register 40001 ~ 40016

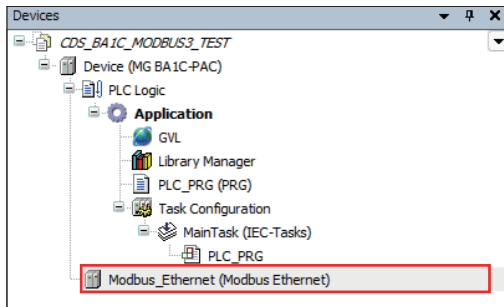
設定手順

- (1) MODBUSServer を必要とする CODESYS プロジェクトを開きます。
- (2) 「ModbusEthernet」の追加
 デバイスツリーにて「導入先のデバイスを選択」－「右クリック」－「Add Device…」を選択します。
 表示された画面で「MG CO.,LTD.」(無ければ「Miscellaneous」) から「Modbus Ethernet」フォルダ内の「Modbus Ethernet」を選択します。



C—CDS35: MglDrvModbus3

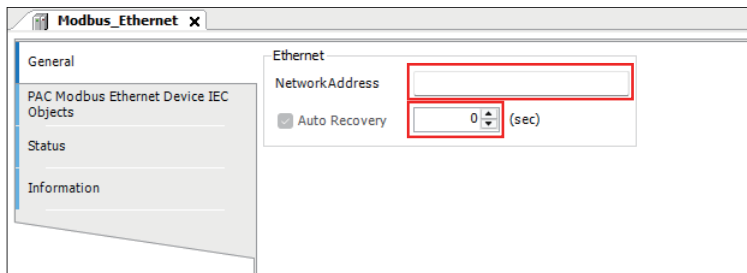
- (3) 「Modbus Ethernet」のパラメータ設定
デバイスツリーにて「追加された「Modbus_Ethernet」をダブルクリック」します。



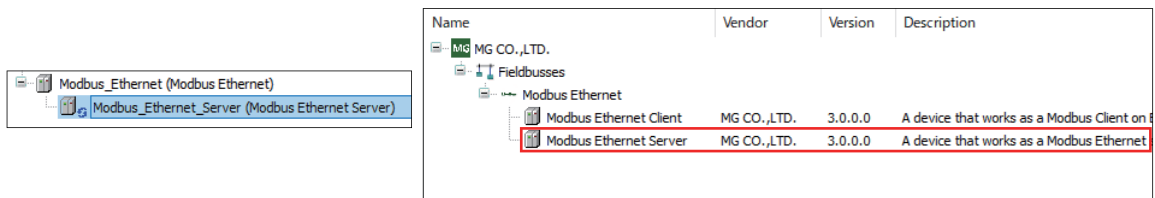
[General] タブ

Network Address : 空欄

Auto Recovery : 0



- (4) client と接続する「Modbus Ethernet Server」の追加
デバイスツリーにて「追加された「Modbus_Ethernet」インスタンスを選択」－「右クリック」－「Add Device…」を選択します。
表示された画面で「Modbus Ethernet」フォルダ内の「Modbus Ethernet Server」を選択します。



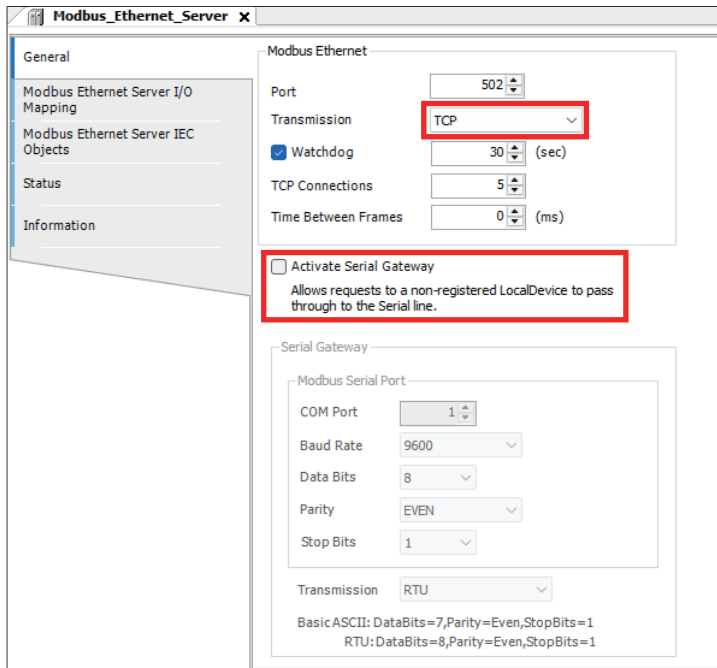
(5) 「Modbus Ethernet Server」のパラメータ設定

デバイスツリーにて「追加された 'Modbus_Ethernet_Server' をダブルクリック」します。
各タブにて次の指定を行います。

[General] タブ

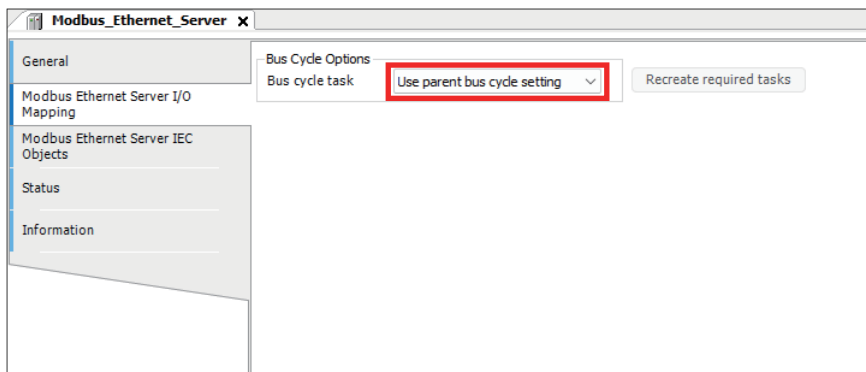
Transmission : TCP (MODBUS/TCP の場合) または UDP (MODBUS/UDP の場合)

Activate Serial Gateway : 上位プロトコルの MODBUS/TCP または UDP を MODBUS/RTU に変換してシリアル通信側の機器とデータ交換を行う場合は「有効」にして、シリアルポートの通信速度などの指定を行います。



[Ethernet Server I/O Mapping] タブ

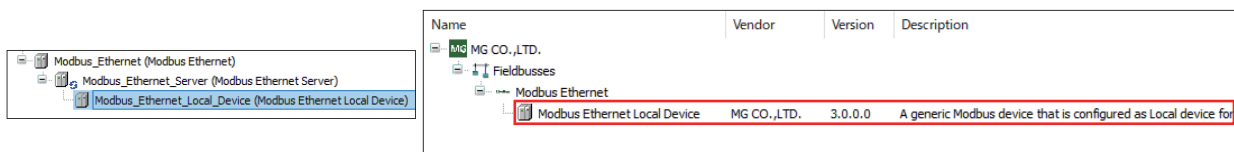
Bus cycle task : 'Use parent bus cycle setting'



(6) Memory 領域を定義する「Modbus Ethernet Local Device」の追加

デバイスツリーにて「追加された 'Modbus_Ethernet_Server' インスタンスを選択」 - 「右クリック」 - 「Add Device…」を選択します。

表示された画面で「[Modbus Ethernet] フォルダ内の「[Modbus Ethernet Local Device]」を選択します。

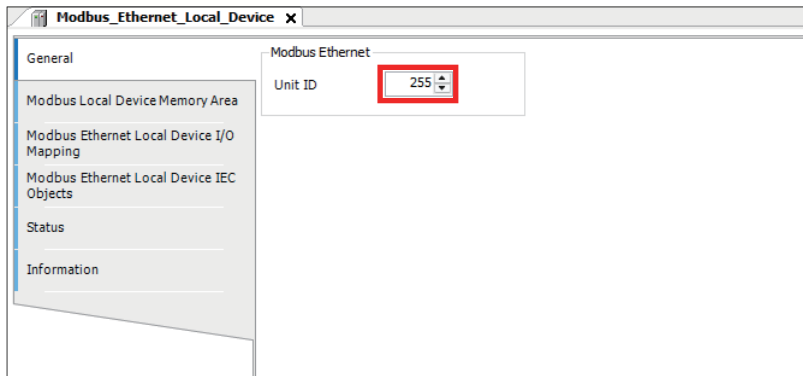


(7) 「Modbus Ethernet Local Device」のパラメータ設定

デバイスツリーにて「追加された 'Modbus_Ethernet_Local_Device' をダブルクリック」します。
各タブにて次の指定を行います。

[General] タブ

Unit ID : '255'

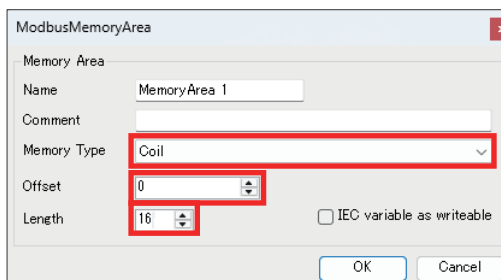


[Modbus Local Device Memory Area] タブ

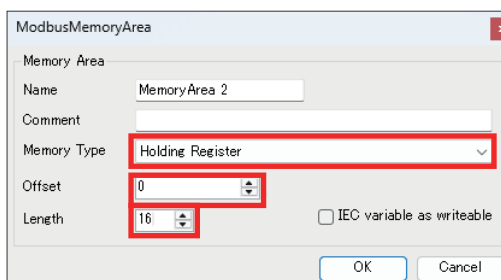


[Add...] ボタンを押し 2つの新しい領域を追加します。

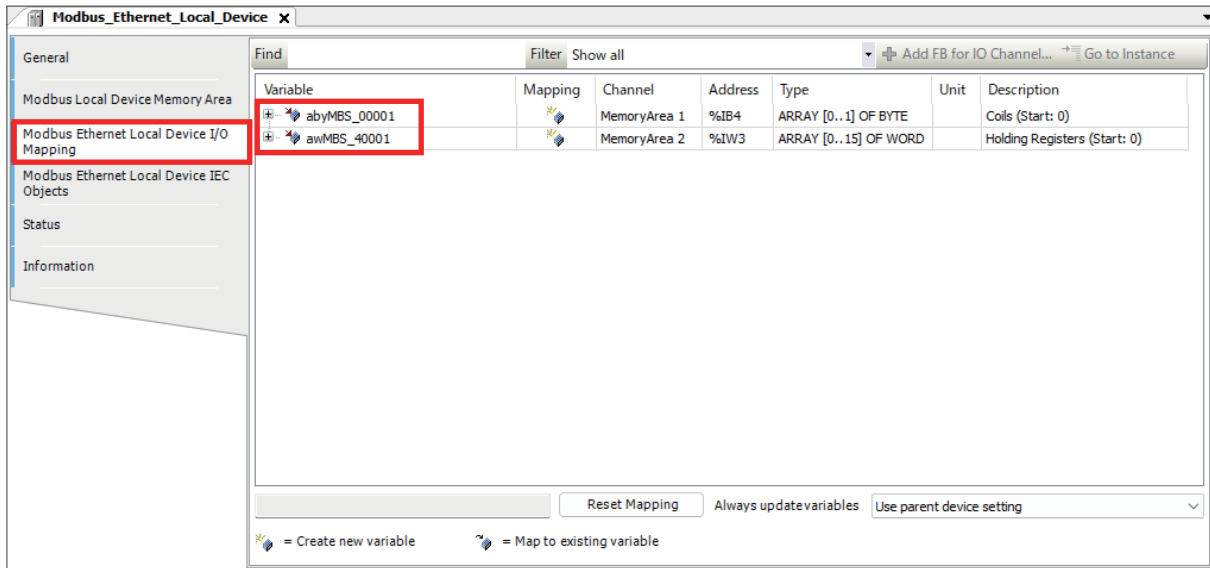
Area1 : Coil 0001 ~ 0016 (Offset : 0, Length : 16)



Area 2 : Holding Register 40001 ~ 40016 (Offset : 0, Length : 16)



[Modbus Ethernet Local Device I / O Mapping] タブ
 Variable 欄をダブルクリックして変数を割り当てます。
 ‘abyMBS_00001’
 ‘awMBS_40001’

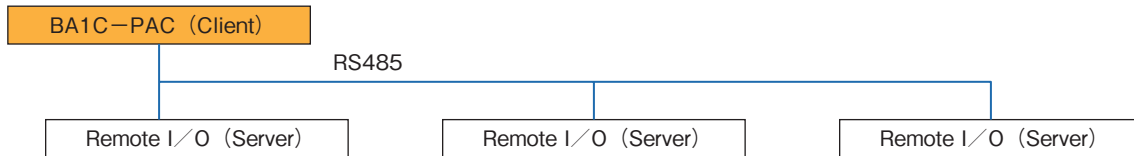


ここまでの設定を行い [Build] でエラーが無ければターゲットで実行できます。

13.3 MODBUS Serial CLIENT

ここでは MODBUS Client 機能の使用方法を解説します。

MODBUS Serial CLIENT 通信の準備



CODESYS プロジェクトの準備

MODBUS Serial 通信を介して device と接続する手順を説明します。

■接続する Server 機器

device-1

Server ID : 1

● 1 秒周期で Holding Register 40001 を読み変数 awMBC_1_40001 に値を書き込む

● 変数 bMBC_1_40002_T の立ち上がりで変数 awMBC_1_40002 の値を Holding Register 40002 に書き込む

device-2

Server ID : 2

device-3

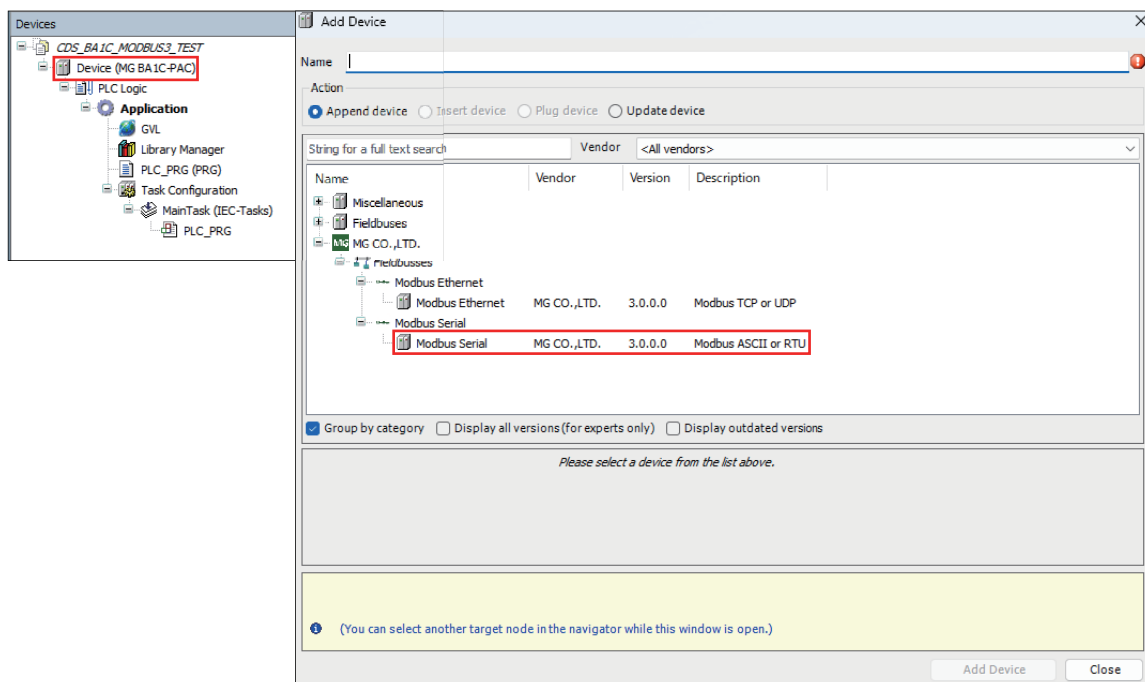
Server ID : 3

■このプロジェクトを動作させる Client 機器 (BA1C-PAC)

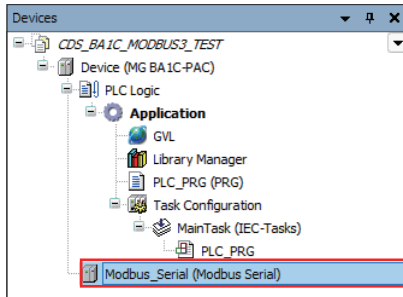
RS485-1 : COM1, 速度 38400, Data 8 bits, Parity none, Stop 1 bit

設定手順

- (1) MODBUS Client を必要とする CODESYS プロジェクトを開きます。
- (2) 「Modbus Serial」の追加
デバイスツリーにて「導入先のデバイスを選択」－「右クリック」－「Add Device…」を選択します。
表示された画面で「MG CO.,LTD.」(無ければ「Miscellaneous」)から「Modbus Serial」フォルダ内の「Modbus Serial」を選択します。

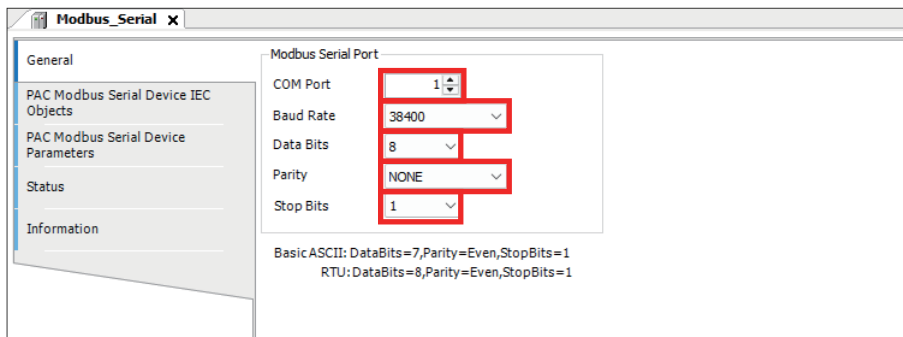


- (3) 「Modbus Serial」のパラメータ設定
 デバイスツリーにて「追加された ‘Modbus_Serial’ をダブルクリック」します。

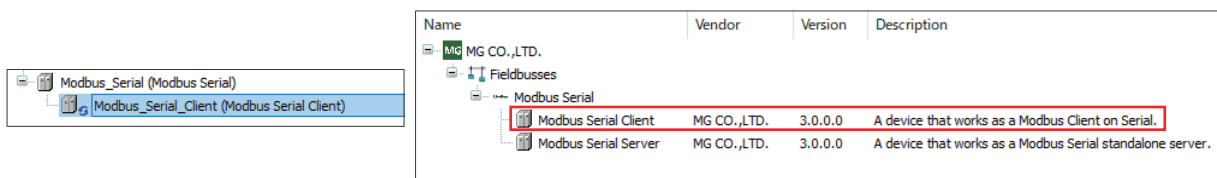


[General] タブ

COM Port : 1
 Baud Rate : 38400
 Data Bits : 8
 Parity : NONE
 Stop Bits : 1



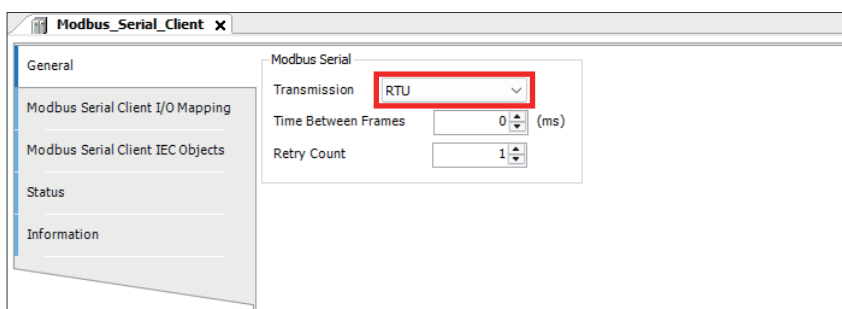
- (4) device-1に接続する「Modbus Serial Client」の追加
 デバイスツリーにて「追加された ‘Modbus_Serial’ インスタンスを選択」 - 「右クリック」 - 「Add Device…」を選択します。
 表示された画面で「Modbus Serial」フォルダ内の「Modbus Serial Client」を選択します。



- (5) 「Modbus Serial Client」のパラメータ設定
 デバイスツリーにて「追加された ‘Modbus_Serial_Client’ をダブルクリック」します。
 各タブにて次の指定を行います。

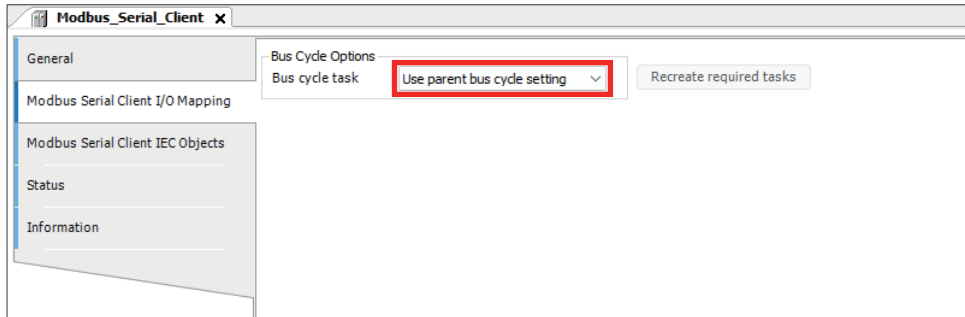
[General] タブ

Transmission : RTU (MODBUS/RTU の場合) または ASCII (MODBUS/ASCII の場合)



[Modbus Serial Client I/O Mapping] タブ

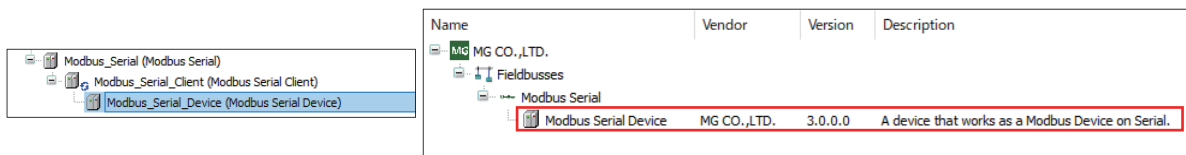
Bus cycle task : 'Use parent bus cycle setting'



(6) device への要求を定義する「Modbus Serial Device」の追加

デバイスツリーにて「追加された「Modbus_Serial_Client」インスタンスを選択」-「右クリック」-「Add Device...」を選択します。

表示された画面で「Modbus Serial」フォルダ内の「Modbus Serial Device」を選択します。



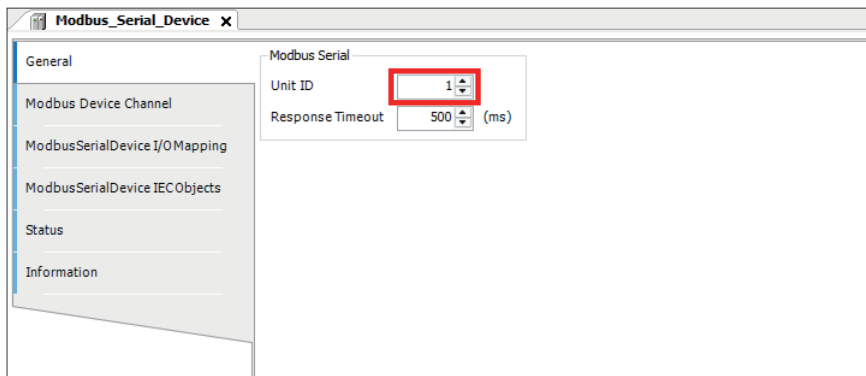
(7) 「Modbus Serial Device」のパラメータ設定

デバイスツリーにて「追加された「Modbus_Serial_Device」をダブルクリック」します。

各タブにて次の指定を行います。

[General] タブ

Unit ID : '1'



[Modbus Device Channel] タブ



C—CDS35: MglDrvModbus3

[Add...] ボタンを押し 2つの新しいチャンネルを追加します。

チャンネル 1: Cyclic, Read Holding Register 40001 (Offset: 0, Length: 1)

ModbusChannel

Channel

Name: Read 40001

Comment:

Access Type: Read Holding Registers (Function Code 3)

Trigger: CYCLIC Cycle Time (ms): 1000

READ Register

Offset: 0

Length: 1

Error Handling: Keep last value

WRITE Register

Offset: 0

Length: 0

OK Cancel

チャンネル 2: Rising Edge, Write Holding Register 40002 (Offset: 1, Length: 1)

ModbusChannel

Channel

Name: Write 40002

Comment:

Access Type: Write Single Register (Function Code 6)

Trigger: RISING_EDGE Cycle Time (ms): 1000

READ Register

Offset: 0

Length: 0

Error Handling: Keep last value

WRITE Register

Offset: 1

Length: 1

OK Cancel

[Modbus Serial Device I / O Mapping] タブ

Variable 欄をダブルクリックして変数を割り当てます。

'awMBC_1_40001'

'awMBC_1_40002' , 'bMBC_1_40002_T'

Variable	Mapping	Channel	Address	Type	Unit	Description
awMBC_1_40001		Channel 1	%IW19	ARRAY [0..0] OF WORD		Read Holding Registers (Start: 0)
awMBC_1_40002		Channel 2	%QW55	ARRAY [0..0] OF WORD		Write Single Register (Start: 1)
bMBC_1_40002_T		Channel 2	%QX112.0	BIT		Trigger Variable

Reset Mapping Always update variables Use parent device setting

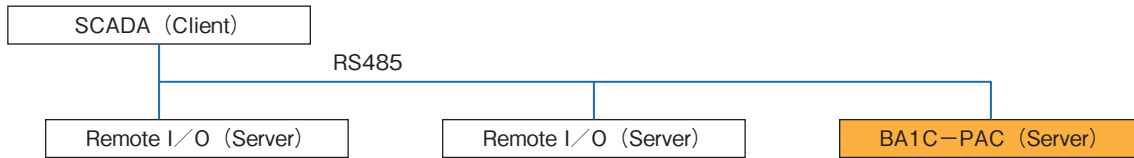
(8) 手順 (4) に戻り、同様の手順で 'device-2' と 'device-3' を追加します。

ここまでの設定を行い [Build] でエラーが無ければターゲットで実行できます。

13.4 MODBUS Serial SERVER

ここでは MODBUS Server 機能の使用方法を解説します。

MODBUS Serial SERVER 通信の準備



CODESYS プロジェクトの準備

MODBUS Serial 通信を介して client と接続する手順を説明します。

■接続する Client 機器

RS485 : 速度 38400, Data 8 bits, Parity none, Stop 1 bit

■このプロジェクトを動作させる Server 機器 (BA1C-PAC)

RS485-1 (COM1)

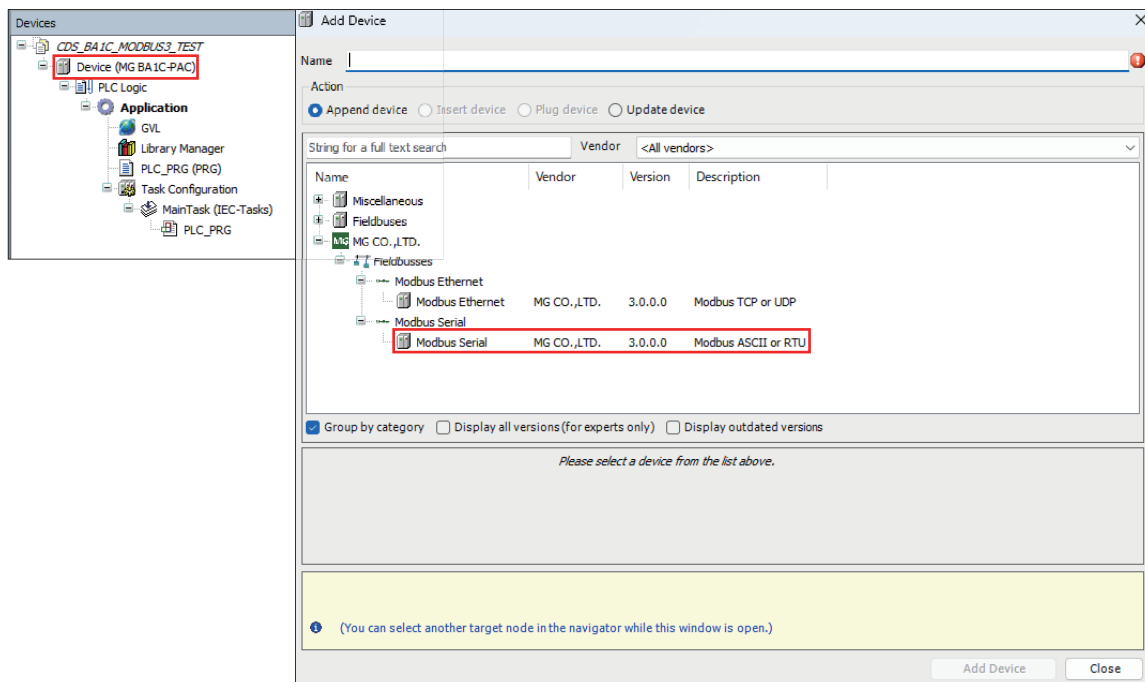
Server ID : 1

Coil 00001 ~ 00016

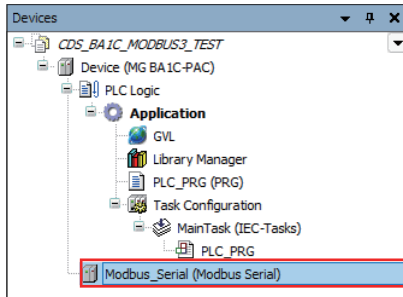
Holding Register 40001 ~ 40016

設定手順

- (1) MODBUS Server を必要とする CODESYS プロジェクトを開きます。
- (2) 「Modbus Serial」の追加
 デバイスツリーにて「導入先のデバイスを選択」→「右クリック」→「Add Device…」を選択します。
 表示された画面で「MG CO.,LTD.」（無ければ「Miscellaneous」）から「Modbus Serial」フォルダ内の「Modbus Serial」を選択します。

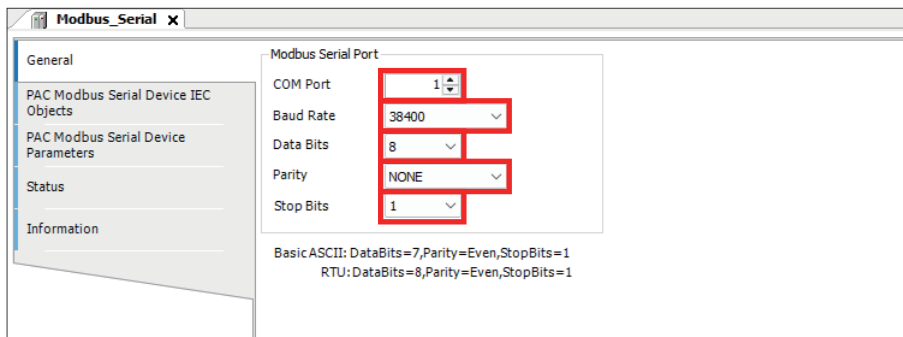


- (3) 「Modbus Serial」のパラメータ設定
 デバイスツリーにて「追加された‘Modbus_Serial’をダブルクリック」します。

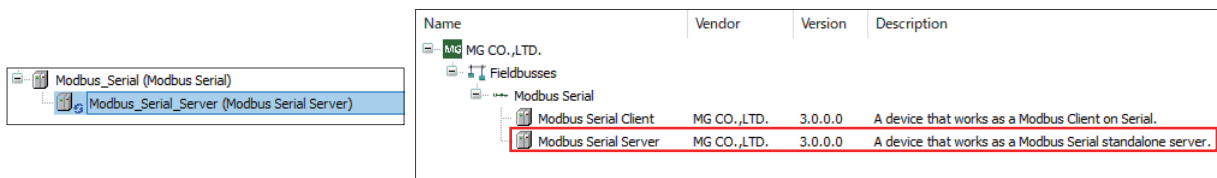


[General] タブ

COM Port : 1
 Baud Rate : 38400
 Data Bits : 8
 Parity : NONE
 Stop Bits : 1



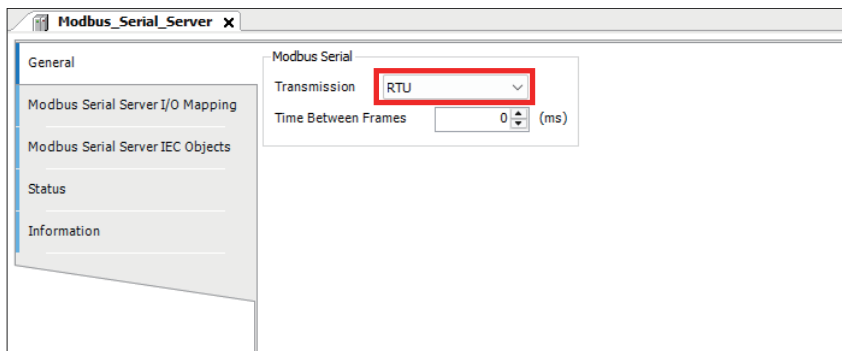
- (4) client と接続する「Modbus Serial Server」の追加
 デバイスツリーにて「追加された‘Modbus_Serial’インスタンスを選択」－「右クリック」－「Add Device…」を選択します。
 表示された画面で「Modbus Serial」フォルダ内の「Modbus Serial Server」を選択します。



- (5) 「Modbus Serial Server」のパラメータ設定
 デバイスツリーにて「追加された‘Modbus_Serial_Server’をダブルクリック」します。
 各タブにて次の指定を行います。

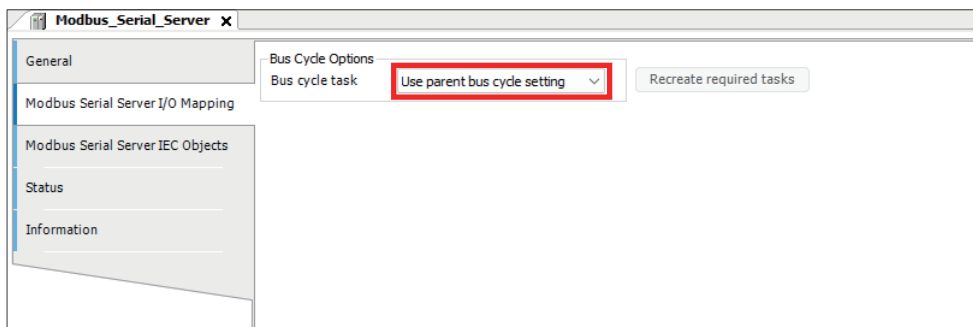
[General] タブ

Transmission : RTU (MODBUS/RTU の場合) または ASCII (MODBUS/ASCII の場合)



[Ethernet Server I / O Mapping] タブ

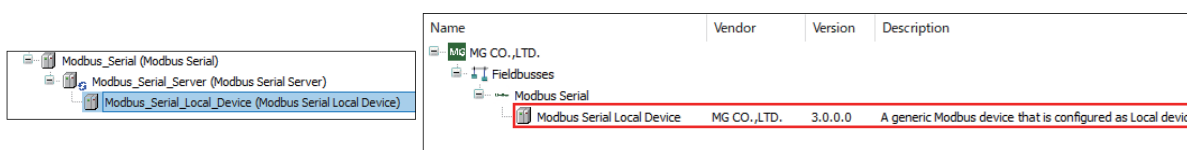
Bus cycle task : 'Use parent bus cycle setting'



(6) Memory 領域を定義する「Modbus Serial Local Device」の追加

デバイスツリーにて「追加された「Modbus_Serial_Server」インスタンスを選択」-「右クリック」-「Add Device...」を選択します。

表示された画面で「[Modbus Serial] フォルダ内の「[Modbus Serial Local Device]」を選択します。



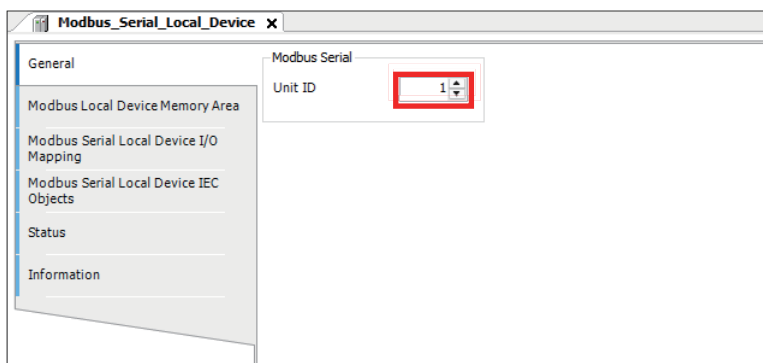
(7) 「Modbus Serial Local Device」のパラメータ設定

デバイスツリーにて「追加された「Modbus_Serial_Local_Device」をダブルクリック」します。

各タブにて次の指定を行います。

[General] タブ

Unit ID : 1



[Modbus Local Device Memory Area] タブ



[Add...] ボタンを押し 2 つの新しい領域を追加します。

Area 1 : Coil 0001 ~ 0016 (Offset : 0, Length : 16)

ModbusMemoryArea

Memory Area

Name: MemoryArea 1

Comment:

Memory Type: Coil

Offset: 0

Length: 16

IEC variable as writable

OK Cancel

Area 2 : Holding Register 40001 ~ 40016 (Offset : 0, Length : 16)

ModbusMemoryArea

Memory Area

Name: MemoryArea 2

Comment:

Memory Type: Holding Register

Offset: 0

Length: 16

IEC variable as writable

OK Cancel

[Modbus Ethernet Local Device I/O Mapping] タブ
 Variable 欄をダブルクリックして変数を割り当てます。
 ‘abyMBS_00001’
 ‘awMBS_40001’

Modbus Serial Local Device I/O Mapping

Variable	Mapping	Channel	Address	Type	Unit	Description
abyMBS_00001	⚡	MemoryArea 1	%IB38	ARRAY [0..1] OF BYTE		Coils (Start: 0)
awMBS_40001	⚡	MemoryArea 2	%IW20	ARRAY [0..15] OF WORD		Holding Registers (Start: 0)

Reset Mapping Always update variables Use parent device setting

ここまでの設定を行い [Build] でエラーが無ければターゲットで実行できます。

14. 拡張機能

本ドライバーの拡張機能について説明します。

14.1 CLIENT のチャネル状態の監視

ここでは MODBUS Client で処理される通信チャネルの状態を取得する方法を解説します。

```

VAR
  _aeMODBUS_CHANNEL_STATUS      : ARRAY [1 .. (GVL_PARAMS.gc_DEF_MAX_MODBUS_CHANNELS - 1)] OF
  ET_MODBUS_CHANNEL_STATUS;
END_VAR

Modbus_Serial_Client_1.GetChannelStatus(uiDeviceIndex:=2,
aeMODBUS_CHANNEL_STATUS=>_aeMODBUS_CHANNEL_STATUS);
// Channel-1 SUCCESS?
IF ET_MODBUS_CHANNEL_STATUS.E_SUCCESS = _aeMODBUS_CHANNEL_STATUS[0] THEN
  // SUCCESS
END_IF
// Channel-2 SUCCESS?
IF ET_MODBUS_CHANNEL_STATUS.E_SUCCESS = _aeMODBUS_CHANNEL_STATUS[1] THEN
  // SUCCESS
END_IF
// Channel-3 SUCCESS?
IF ET_MODBUS_CHANNEL_STATUS.E_SUCCESS = _aeMODBUS_CHANNEL_STATUS[2] THEN
  // SUCCESS
END_IF

```

状態は

E_UNCONFIGURED : 設定なし
 E_CONFIGURED : 設定済み (1度もデータ交換をしていない)
 E_SUCCESS : 直前の通信でデータ交換成功
 E_ERROR : 直前の通信でデータ交換失敗

14.2 MODBUS CLIENT 通知機能

ここでは MODBUS Client からアプリケーションに通知するコールバック機能の使用方法を解説します。

■コールバックされる通知種類

OnPrepareRequest

MODBUS 要求を送信する直前に通知します。

uiDevice, uiLine, eAccessType と uiRetryCounter が渡されます。

初回の呼び出しは uiRetryCounter=0 で通知されます。1 以上はリトライでの送信です。

OnSuccess

直前に処理された uiDevice (1..), uiLine (1..) が成功したことを通知します。

OnError

直前に処理された uiDevice (1..), uiLine (1..) が失敗したことを通知します。

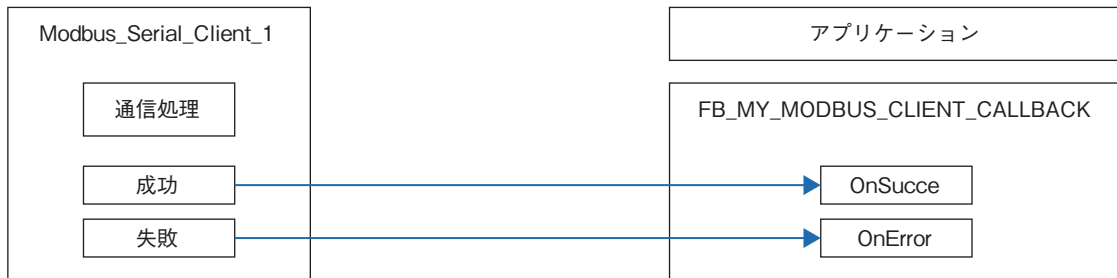
この時のエラーは eModbusError に格納される。

■コールバックされるインスタンスの宣言方法

```
FUNCTION_BLOCK FB_MY_MODBUS_CLIENT_CALLBACK IMPLEMENTS ITF_MODBUS_CLIENT_CB
```

■登録方法

```
Modbus_Serial_Client_1.SetCallback (itfCB:= GVL.fbMY_MODBUS_CLIENT_CALLBACK) ;
Modbus_Ethernet_Client_1.SetCallback (itfCB:= GVL.fbMY_MODBUS_CLIENT_CALLBACK) ;
```



14.3 MODBUS SERVER 通知機能

ここでは MODBUS Server からアプリケーションに通知するコールバック機能の使用方法を解説します。

■コールバックされる通知種類

itf_MB_INITIALIZE

回線の初期化処理完了を通知します。

itf_MB_CHK_SERVER_ID

受信した MODBUS 要求の SERVER ID が対象であることを通知します。

ここで戻り値として ET_MODBUS_ERROR.E_OK 以外を返すと、この PDU は処理されずにエラーレスポンスが返されます。特定の相手以外の要求を拒否したい場合は pstrRemoreHost で渡される相手 IP Address を判断して ET_MODBUS_ERROR.E_FRAME_NOT_EXPECTED_SERVER_ID を返すことで拒否（無応答）できます。

```
// IP Filter
IF pstrRemoreHost^ <> '192.168.2.207' THEN
    itf_MB_CHK_SERVER_ID := ET_MODBUS_ERROR.E_FRAME_NOT_EXPECTED_SERVER_ID;
END_IF
```

itf_MB_CHK_ADDRESS

受信した MODBUS 要求のアドレスが通知されます。

ここで戻り値として ET_MODBUS_ERROR.E_OK 以外を返すと、この PDU は処理されずにエラーレスポンスが返されます。例えば特定のファンクション・コード要求を拒否したい場合は ET_MODBUS_ERROR.E_ILLEGAL_FUNCTION を返すことで拒否できます。

```
// READONLY (Unsupported WRITE requests)
CASE eFC OF
ET_MODBUS_FUNCTION_CODE.E_FC05_WRITE_SINGLE_COIL,
ET_MODBUS_FUNCTION_CODE.E_FC06_WRITE_SINGLE_REGISTER,
ET_MODBUS_FUNCTION_CODE.E_FC15_WRITE_MULTIPLE_COILS,
ET_MODBUS_FUNCTION_CODE.E_FC16_WRITE_MULTIPLE_REGISTERS:
    itf_MB_CHK_ADDRESS := ET_MODBUS_ERROR.E_ILLEGAL_FUNCTION;
END_CASE
```

itf_MB_ENTER

各ファンクション・コードの処理を行う前に通知されます。

itf_MB_READ_COIL

itf_MB_READ_DI

itf_MB_READ_FILE_RECORD

itf_MB_READ_HR

itf_MB_READ_IR

itf_MB_WRITE_COIL

itf_MB_WRITE_FILE_RECORD

itf_MB_WRITE_HR

各ファンクションブロックの処理中に C,DI は 1 ビット毎、IR,HR は 1 ワード毎に要求されたカウント数分繰り返し呼び出されます。

itf_MB_LEAVE

各ファンクション・コードの処理の完了で通知されます。

itf_MB_ON_SEND_ACK

成功の MODBUS レスポンスを返送したことを通知します。

itf_MB_ON_SEND_NAK

エラーの MODBUS レスポンスを返送したことを通知します。

itf_MB_ON_ERROR

前述のエラーの MODBUS レスポンスを含むエラーの発生を通知します。

注) Serial Gateway を介した要求では以下のメソッドのコールバックはされません。

```

itf_MB_CHK_ADDRESS
itf_MB_ENTER
itf_MB_READ_COIL
itf_MB_READ_DI
itf_MB_READ_FILE_RECORD
itf_MB_READ_HR
itf_MB_READ_IR
itf_MB_WRITE_COIL
itf_MB_WRITE_FILE_RECORD
itf_MB_WRITE_HR
itf_MB_LEAVE
    
```

■コールバックされるインスタンスの宣言方法

```
FUNCTION_BLOCK FB_MY_MODBUS_SERVER_CALLBACK IMPLEMENTS ITF_MODBUS_SERVER_FC_CB
```

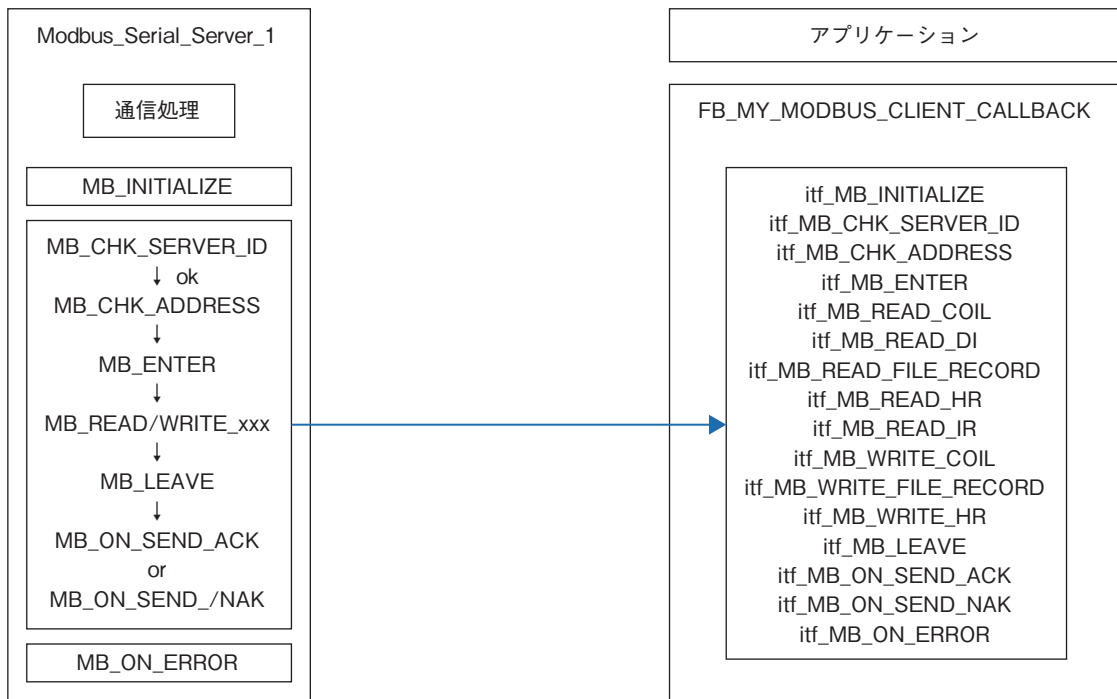
■登録方法

```
Modbus_Serial_Server_1.SetCallback (itfCB:= GVL.fbMY_MODBUS_SERVER_CALLBACK) ;
```

```
Modbus_Ethernet_Server.SetCallback (iConnIdx:= 0, itfCB:= GVL.fbMY_MODBUS_SERVER_CALLBACK) ;
```

Ethernet の場合の登録には iConnIdx の指定が必要となります。

これは複数回線で並列に処理されるので回線毎にコールバックを受け取るインスタンスを必要とします。複数の回線に同じインスタンスを登録した場合は、呼び出されたコールバックがどの回線で処理されたものか識別できません。



15. 添付

15.1 列挙型コード一覧

以降に CODESYS IDE の各 Status 画面で使用されている列挙型を示します。表示では記号の先頭文字列 'E_' は表示されません。

ET_CLIENT_STATUS

記号	値	内容
E_IDLE	0	初期状態
E_TRIGGERD	1	トリガー状態
E_SUCCESS	2	成功
E_ERROR	3	失敗
E_RETRY	4	リトライ中

ET_MODBUS_CHANNEL_STATUS

記号	値	内容
E_UNCONFIGURED	0	初期状態(未設定)
E_CONFIGURED	1	設定状態
E_SUCCESS	2	成功
E_ERROR	9	失敗

ET_MODBUS_ERROR

記号	値	内容
E_OK	0	正常
[MODBUS Error Code]		
E_ILLEGAL_FUNCTION	0x01	クエリで受信した機能コードは、サーバーが許可する操作ではありません。 これは、機能コードが新しいデバイスにのみ適用可能であり、選択されたユニットでは実装されていないためです。また、サーバーがこのタイプの要求を処理する状態にないことを示している可能性があります。例えば、サーバーが設定されていない状態でレジスタ値の返却を要求されている場合などが該当します。
E_ILLEGAL_DATA_ADDRESS	0x02	クエリで受信したデータアドレスは、サーバーが許可するアドレスではありません。具体的には、参照番号と転送長の組み合わせが無効です。
E_ILLEGAL_DATA_VALUE	0x03	クエリデータフィールドに含まれる値がサーバーで許可されていない値です。
E_SERVER_DEVICE_FAILURE	0x04	要求された処理を実行しようとしたサーバーで回復不能なエラーが発生しました。
E_ACKNOWLEDGE	0x05	プログラミングコマンドの特殊用途です。 サーバーはリクエストを受信し処理中ですが、完了までに長時間を要します。 この応答はクライアント側でタイムアウトエラーが発生するのを防ぐために返されます。クライアントは次に「Poll Program Complete」メッセージを発行し、処理が完了したかどうかを確認できます。
E_SERVER_DEVICE_BUSY	0x06	プログラミングコマンドの特殊用途です。 サーバーは長時間処理を要するプログラムコマンドの処理中です。 クライアントはサーバーが空き状態になった時点でメッセージを再送信して下さい。
E_MEMORY_PARITY_ERROR	0x08	Function Code 20 および 21、参照タイプ 6 と組み合わせて使用し、拡張ファイル領域が一貫性チェックに失敗したことを示します。 サーバーはレコードファイルの読み取りを試みましたが、メモリー内でパリティエラーを検出しました。

C – CDS35: MgloDrvModbus3

記号	値	内容
E_GATEWAY_PATH_UNAVAILABLE	0x0A	ゲートウェイを使用する用途で、ゲートウェイがリクエスト処理のため入力ポートから出力ポートへの内部通信経路を割り当てられなかったことを示します。通常、ゲートウェイの設定ミスまたは過負荷を意味します。
E_GATEWAY_TARGET_DEVICE_FAILED_TO_RESPOND	0x0B	ゲートウェイを使用する用途で、対象デバイスから応答が得られなかったことを示します。通常、デバイスがネットワーク上に存在しないことを意味します。
[Application Error Code]		
E_NO_LICENSE	0x101	ライセンスがありません。
E_OPEN_ALREADY_OPENED	0x102	すでにオープン済みです。
E_OPEN_NOT_OPENED	0x103	オープンに失敗しました。
E_BUSY	0x104	ビジー状態です。
E_WOULDBLOCK	0x105	要求が実行できない状態です。
E_PAR_NULL	0x106	NULLパラメータが指定されています。
E_PAR_INVALID	0x107	指定されたパラメータに問題があります。
[MODBUS RTU CLIENT]		
E_OPEN_FAILED_COMOPEN	0x201	COMポートのオープンに失敗しました。
E_PAR_NOT_VALID_ServerID	0x211	Out of 1-247:Server individual address
E_PAR_NOT_VALID_StartAdress	0x212	Out of 0x0000 to 0xffff
E_PAR_NOT_VALID_Quantity	0x213	Out of 1-247:Server individual address
E_FRAME_CRC_ERROR	0x214	フレームのCRCエラーを検出しました。
E_FRAME_LRC_ERROR	0x215	フレームのLRCエラーを検出しました。
E_FRAME_ILLEGAL_LENGTH	0x216	フレームの正しくない長さを検出しました。
E_FRAME_NOT_EXPECTED_SERVER_ID	0x217	フレームのサーバー識別番号が処理できません。
E_FRAME_NOT_EXPECTED_FUNC_CODE	0x218	フレームのファンクションコードが処理できません。
E_FRAME_ILLEGAL_BYTE_COUNT	0x219	フレームのバイトカウントが正しくありません。
E_FRAME_ILLEGAL_RESP_ADDRESS	0x21A	フレームの返信アドレスが正しくありません。
E_FRAME_ILLEGAL_RESP_VALUE	0x21B	フレームの返信データが正しくありません。
E_FRAME_FILE_BLOCK_PARAM_ERROR	0x21C	フレームのファイルブロックパラメータが異常です。
E_FRAME_FILE_BLOCK_RESP_ERROR	0x21D	フレームのファイルブロック返信が異常です。
E_FRAME_NOT_VALID_WRITE_RESP	0x21E	フレームの書き込み返信が正しくありません。
[MODBUS UDP/TCP CLIENT]		
E_OPEN_FAILED_ADAPTER	0x301	アダプターが間違っています。
E_OPEN_FAILED SOCK_CREATE	0x302	ソケット作成に失敗しました。
E_OPEN_FAILED_GETHOSTBYNAME	0x303	ホスト名解決に失敗しました。
E_OPEN_FAILED SOCK_BIND	0x304	ソケットのバインドに失敗しました。
E_OPEN_FAILED SOCK_SETOPT	0x305	ソケットのオプション指定に失敗しました。
E_OPEN_FAILED SOCK_CONNECT	0x306	ソケットの接続に失敗しました。
E_OPEN_FAILED SOCK_TLS_CONTEXT	0x307	ソケットのTLS処理に失敗しました。
E_FRAME_NOT_MATCH_LENGTH	0x311	not match the length in frame
E_FRAME_NOT_VALID_PROTOCOL_ID	0x312	フレームのプロトコルが正しくありません。
E_FRAME_NOT_MATCH_TRANSACTION	0x313	フレームのトランザクションが一致しません。
[TRANSMISSION]		
E_FRAME_SEND_ERROR	0x401	フレームの送信エラーを検知しました。
E_FRAME_RECV_ERROR	0x402	フレームの受信エラーを検知しました。
E_BUFFER_OVERRUN_ERROR	0x403	バッファオーバーランを検知しました。
E_SEND_TIMEOUT	0x501	送信タイムアウトを検知しました。
E_RECV_TIMEOUT	0x502	受信タイムアウトを検知しました。
E_CONN_CLOSED_BY_PEER	0x503	接続が切断を検知しました。
E_NOT_IMPLEMENTED	0xFFFFE	未実装です。
E_GENERAL	0xFFFFF	その他のエラー

ET_MODULE_STATUS

記号	値	内容
E_DISABLED	0	初期状態(無効)
E_INITIALIZE1	1000	
E_INITIALIZE2	1010	
E_INITIALIZED	1020	
E_STOPPING	2000	
E_STOPPED	2010	
E_STARTING	3000	
E_STARTED	3010	
E_CLIENT_WAIT_FOR_TRIGGER	3020	
E_CLIENT_SEND_REQUEST	3021	

記号	値	内容
E_CLIENT_WAIT_FOR_RESPONSE	3022	
E_CLIENT_WAIT_FOR_TIME_BETWEEN_FRAME	3023	
E_CLIENT_WAIT_FOR_TIME_BETWEEN_FRAME_ERR	3024	
E_CLIENT_RUNNING	3025	
E_SERVER_LISTENING	3030	
E_ERROR	9000	
E_ERROR_RECOVERING1	9010	
E_ERROR_RECOVERING2	9020	

ET_SERVER_COMM_STATUS

記号	値	内容
E_IDLE	0	初期状態
E_SUCCESS	1	成功
E_ERROR	2	失敗
E_MB_EXCEPTION	3	MODBUS エラーレスポンス受信

15.2 セキュリティ

MODBUS 通信をより安全に行うためにネットワークに Firewall を導入し不必要な通信の遮断や通信相手を限定するような設定を行われることをお勧めします。

更に本ドライバーは次の処理を組み込むことができます。

MODBUS Server を読み取り専用にする

要求ファンクション・コードの拒否を行います。
MODBUS ASCII/RTU/TCP/UDP に対して実施できます。

MODBUS Server への要求は特定の相手に限定する

要求相手 IP アドレスのフィルタリングを行います。
MODBUS TCP/UDP に対して実施できます。

これらは「MODBUS SERVER 通知機能」を使用してアプリケーションにて実装できます。

■Firewall が導入されている場合

MODBUS TCP または UDP が使用するポートあるいは IP アドレスが通信できるように設定します。

例えば


MODBUS/TCP SERVER は Destination port 502 のパケットを許可


MODBUS/UDP SERVER は Destination port 502 と Source port 502 のパケットを許可


16. 異常時の処置とメンテナンス


16.1 オンラインモードでのデバイスステータス


オンラインモードでは、デバイスエントリの前の記号でデバイスのステータスを示します。



 : デバイスが動作してデータが交換されています。

 : デバイスが停止してデータを交換していません。

 : デバイスはデータを交換していません。バスエラー、構成なし、またはシミュレーションモードです。

 : デバイスは構成されていますが、完全には機能していません。データは交換されません。

 : デバイスの説明がデバイスリポジトリに見つかりませんでした。

  : デバイス自体は実行されていますが、子デバイスが実行されていないか、診断メッセージがあります。デバイスツリーが折りたたまれているため、子デバイスは表示されません。

16.2 デバイス ログ

MODBUS ドライバーのログは CODESYS IDE の [Device] タブの [Log] ‘MgIoDrvModbus’ にて確認ができます。

表示メッセージ [Description] の形式

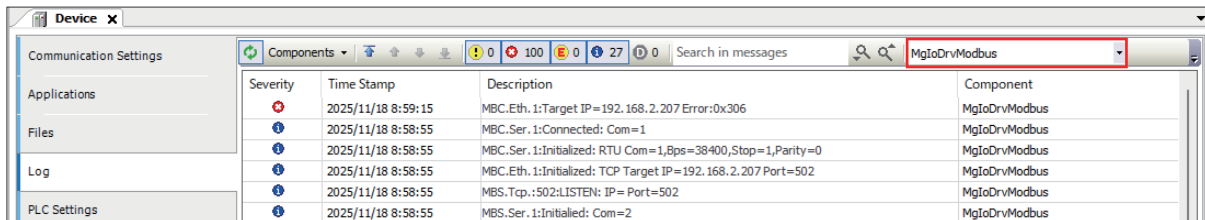
例：‘MBC.Eth.1:<message>’







(1) (2) (3)

(1) : MBC=MODBUS CLIENT, MBS=MODBUS SERVER

(2) : Eth=Ethernet, Ser=Serial

(3) : Instance No (Device Tree 表示の順に有効なオブジェクトだけを対象に 1 から始まる連番)



Severity	Time Stamp	Description	Component
	2025/11/18 8:59:15	MBC.Eth.1:Target IP=192.168.2.207 Error:0x306	MgIoDrvModbus
	2025/11/18 8:58:55	MBC.Ser.1:Connected: Com=1	MgIoDrvModbus
	2025/11/18 8:58:55	MBC.Ser.1:Initialized: RTU Com=1,Bps=38400,Stop=1,Parity=0	MgIoDrvModbus
	2025/11/18 8:58:55	MBC.Eth.1:Initialized: TCP Target IP=192.168.2.207 Port=502	MgIoDrvModbus
	2025/11/18 8:58:55	MBS.Tcp.:502:LISTEN: IP= Port=502	MgIoDrvModbus
	2025/11/18 8:58:55	MBS.Ser.1:Initialed: Com=2	MgIoDrvModbus

16.3 トラブルシューティング

MODBUS Client に関するトラブル

Client が要求しているが MODBUS Error 0x02 が返される。	Channel 設定で要求している Access Type, Offset, Length を確認して下さい。
Client が要求しているがタイムアウトとなる。	要求している Unit ID を確認して下さい。

MODBUS Sever に関するトラブル

Client が要求しているが MODBUS Error 0x02 が返される。	Memory Area の定義を確認して下さい。
Client が要求しているがタイムアウトとなる。	<ul style="list-style-type: none"> 通信速度などの通信設定が Client と一致しているか確認して下さい。 Unit ID が正しく指定されているか確認して下さい。
Client が要求しているがタイムアウトとなる。 (Serial Gateway 機能使用時)	<ul style="list-style-type: none"> MODBUS Ethernet Server で Serial Gateway を介して MODBUS RTU 機器にアクセスしている場合は通信速度やレスポンスの遅延を考慮したタイムアウト値であるか Client を確認して下さい。 また複数の Client が同時に Serial Gateway を介したアクセスを行うと、順に処理されるので Client のタイムアウトが発生しやすくなります。その場合は Client のタイムアウト時間を長めに設定して下さい。
Client との通信は成功しているが、期待する値が返されない。	<ul style="list-style-type: none"> Local Device に同一の Unit ID が定義されていないか確認して下さい。 Serial gateway を有効にしている場合は Remote I/O と同一の Unit ID を持つ Local Device が定義されていないか確認して下さい。Local Device が優先されてデータを返します。
Server の通信設定を変更後、リカバリ時間が経過しても Client との通信が受け付けられない。	Server を動作させている機器 (PLC) をリセットまたは電源の再投入を行って下さい。
Local Device の Memory Area 定義画面から移動する際に CODESYS IDE が無応答になる。	Device の追加や削除を繰り返した後のプロジェクトで Length に 10000 を超える大きな領域を定義すると、定義直後の画面移動で CODESYS IDE がかなり長い時間無応答の状態になります。その場合は中断してプロジェクトを新規に作成しなおして下さい。

17. 履歴

改訂番号	内 容
初版	初版

株式会社エムジー

〒541-0042

大阪市中央区今橋2丁目5番8号(トレードピア淀屋橋13F)

TEL 06-7525-8800

E-mail hotline@mgco.jp