

MELSEC プロバイダ 三菱電機 PLC

Version 1.0.1

ユーザーズ ガイド

July 17, 2012

【備考】

【改版履歴】

バージョン	日付	内容
1.0.0.0	2006-02-24	初版
1.0.1.0	2008-02-06	マニュアル修正
1.0.1.1	2010-02-12	エラーコード追加
1.0.1.2	2011-03-11	プロバイダ登録ツールに関する追記
1.0.1	2012-07-17	ドキュメントのバージョンルールを変更

【対応機器】

機種	バージョン	注意事項

目次

1. はじめに	4
2. プロバイダの概要	5
2.1. 概要	5
2.2. メソッド・プロパティ	6
2.2.1. CaoWorkspace::AddController メソッド	6
2.2.2. CaoController::AddCommand メソッド	6
2.2.3. CaoController::AddTask メソッド	7
2.2.4. CaoController::AddVariable メソッド	7
2.2.5. CaoCommand::Execute メソッド	8
2.2.6. CaoCommand::put_Parameter プロパティ	8
2.2.7. Caocommand::get_Parameter プロパティ	8
2.2.8. CaoTask::Start メソッド	8
2.2.9. CaoTask::Stop メソッド	8
2.2.10. CaoVariable::put_Value プロパティ	8
2.2.11. CaoVariable::get_Value プロパティ	9
2.3. 変数一覧	10
2.3.1. コントローラクラス	10
2.4. エラーコード	10
2.5. CAO-MELSEC API 対応表	11
3. サンプルプログラム	12

1. はじめに

本書は、MELCO 製 MELSEC データリンクライブラリの API (以降 MELSEC API とします) を使用して通信を行うプロバイダである、MELSEC プロバイダのユーザーズガイドです。

詳細については、MELSEC データリンクライブラリのヘルプを参照して下さい。

注意: MELSEC プロバイダを使用するには、MELSEC データリンクライブラリをインストールしなければなりません。ドライバインストール後にプロバイダをレジストリ登録する必要があります。レジストリ登録の方法は表 2-1 を参照してください。

2. プロバイダの概要

2.1. 概要

MELSEC プロバイダは, CAO API を実行するときに対応する MELSEC API を実行します. 以下に CAO API と MELSEC API の対応については表 2-7 を参照してください.

プロバイダの概要について記述します.

表 2-1 MELSEC プロバイダ

ファイル名	CaoProvMELSEC.dll
ProgID	CaoProv.MELSEC
レジストリ登録 ¹	regsvr32 CaoProvMELSEC.dll
レジストリ登録の抹消	regsvr32 /u CaoProvMELSEC.dll

¹ プロバイダの登録は regsvr32.exe または RegCOM.exe ([スタート]→[ORiN2]→[Tools])で実行できます. MELSEC データリンクライブラリをインストールしていないと, MELSEC プロバイダの登録はできません.

2.2. メソッド・プロパティ

2.2.1. GaoWorkspace::AddController メソッド

MELSEC プロバイダでは Controller オブジェクトの生成時に MELSEC API の mdOpen を用いて接続処理を行います。接続時にオプション文字列でデバイスを指定します。

```
AddController
(
    "<コントローラ名>", // コントローラ名.
    "GaoProv. MELSEC", // プロバイダ名. 固定.
    "<マシン名>", // プロバイダの実行マシン名.
    "<オプション>" // オプション文字列.
)
```

以下にオプション文字列に指定するリストを示します。

表 2-2 GaoWorkspace::AddController のオプション文字列

オプション	意味
CH=<チャンネル番号>	通信媒体, 通信先との接続形態(必須) 詳細については MELSEC API のヘルプを参照してください。

2.2.2. GaoController::AddCommand メソッド

MELSEC API を実行するコマンドオブジェクトを生成します。

```
AddCommand
(
    "<コマンド名>", // コマンド名.
    "<オプション>" // オプション文字列. (未使用)
)
```

以下に使用できるコマンド名の一覧を示します。

表 2-3 コマンド名一覧

コマンド名	意味	パラメータ	結果
Init	シーケンサデバイス番地テーブルのリフレッシュ	-	-
Reset	自ボードをリセットする	-	-
BitSet	ビットデバイスのセット	VT_I2: 局番 VT_I2: デバイスタイプ VT_I2: 指定デバイス No.	-
BitReset	ビットデバイスのリセット	VT_I2: 局番	-

		VT_I2: デバイスタイプ	
		VT_I2: 指定デバイス No.	

各コマンド名と実行 MELSEC API の関係については表 2-7 を参照してください。

2.2.3. CaoController::AddTask メソッド

MELSEC API の mdControl を実行する Task オブジェクトを生成します。

タスク名には、局番を 10 進数で指定します。これ以外のタスク名を指定したときはこのメソッドは、エラーを返します。

```
AddTask
(
  "<局番>",           // コマンド名.
  "<オプション>"      // オプション文字列. (未使用)
)
```

2.2.4. CaoController::AddVariable メソッド

MELSEC API を実行する変数オブジェクトを生成します。

```
AddVariable
(
  "<局番>:<デバイスタイプ>:<(先頭)デバイス番号>" // 変数名.
  "<オプション>" // オプション文字列.
)
```

使用できるシステム変数名に付いては 2.3.1 を参照してください。

ここで、ユーザ変数名は以下のように指定します。

<局番>:<デバイスタイプ>:<(先頭)デバイス番号>

局番、デバイス名、デバイス番号は 10 進数字で指定します。各値の設定値に付いては MELSEC API のヘルプを参照してください。

例) 局番:255, デバイスタイプ:1, デバイス番号:0 の場合

```
Set Var = AddVariable("255:1:0")
```

このメソッドでは、オプション文字列に以下のものを指定することができます。

表 2-4 CaoController::AddVariable のオプション文字列

オプション	意味
Elements=<バイト数>	ユーザ変数で扱うサイズ(デフォルト:2) システム変数のときは使用しません。

2.2.5. GaoCommand::Execute メソッド

コマンド名に対応する MELSET API を実行します。対応については、表 2-7 を参照してください。
パラメータが不正なとき、実行結果にエラーを返します。

2.2.6. GaoCommand::put_Parameter プロパティ

MELSEC API の実行時に使用するパラメータを設定します。各コマンドに必要なパラメータについては、表 2-7 を参照してください。

このプロパティでは、パラメータが適切かどうかのチェックは行いません。パラメータが不適切なときは、2.2.5 の実行結果がエラーになります。

2.2.7. GaoCommand::get_Parameter プロパティ

現在設定されているパラメータを取得します。
何も設定されていないときは VT_EMPTY を取得します。

2.2.8. GaoTask::Start メソッド

MELSEC API の mdControl をリモート RUN で実行します。

```
Start
(
  "<開始モード>" // 開始モード (未使用)
  "<オプション>" // オプション文字列. (未使用)
)
```

第 1, 2 引数はともに無視されます。しかし、第 1 引数の開始モードは必須入力のため、常に 0 を設定してください。

2.2.9. GaoTask::Stop メソッド

MELSEC API の mdControl をリモート PAUSE またはリモート STOP で実行します。

```
Stop
(
  "<停止モード>" // 停止モード
  "<オプション>" // オプション文字列. (未使用)
)
```

停止モードに“2”(ステップ停止)を指定した場合はリモート PAUSE を、それ以外の値を指定したときはリモート STOP を実行します。

2.2.10. GaoVariable::put_Value プロパティ

変数名に対応する MELSEC API を実行します。対応については、表 2-7 を参照してください。

2.2.11. GaoVariable::get_Value プロパティ

変数名に対応する MELSEC API を実行します。対応については、表 2-7 を参照してください。

2.3. 変数一覧

2.3.1. コントローラクラス

表 2-5 コントローラクラス ユーザ変数一覧

変数名	データ型	説明	属性	
			get	put
<局番>:<デバイスタイプ>: <(先頭)デバイス番号>	VT_ARRAY VT_I2	データの送受信. MELSEC API の mdSend/mdReceive を実行します.	○	○

表 2-6 コントローラクラス システム変数一覧

変数名	データ型	説明	属性	
			get	put
@LED	VT_ARRAY VT_I2	自ボードの LED 情報. MELSEC API の mdBdLedRead を実行します.	○	-
@MODE	VT_I2	自ボードのモード情報. MELSEC API の mdBdModRead / mdBdModSet を実行します.	○	○
@SW	VT_ARRAY VT_I2	自ボードのスイッチ状態. MELSEC API の mdBdSwRead を実行します.	○	-
@VER	VT_ARRAY VT_I2	自ボードのバージョン情報. MELSEC API の mdBdVerRead を実行します.	○	-
@TYPE??	VT_I2	シーケンサ CPU の形名. 変数名の後ろに局番を 10 進数で指定します. 例) “TYPE255” MELSEC API の mdTypeRead を実行します.	○	-

2.4. エラーコード

MELSEC プロバイダでは、固有のエラーコードとして MELSEC API で規定されたエラー番号を“0x8010000”でマスクした値を返します。

例) MELSEC API のエラー:0x64 → CAO API のエラー:0x80100064

MELSEC の詳細については、MELSEC データリンクライブラリを参照してください。

ORiN2 共通エラーについては、「[ORiN2 プログラミングガイド](#)」のエラーコードの章を参照してください。

2.5. CAO-MELSEC API 対応表

表 2-7 CAO API と MELSEC API の対応表

CAO API		MELSEC API
CaoController::FinalConstruct		mdOpen
CaoController::FinalRelease		mdClose
CaoCommand::Execute	(コマンド名: Init)	mdInit
	(コマンド名: Reset)	mdBdRst
	(コマンド名: BitSet)	mdDevSet
	(コマンド名: BitReset)	mdDevRst
CaoVariable::put_Value	(変数名: ユーザ変数)	mdSend
	(変数名: @Mode)	mdBdModSet
CaoVariable::get_Value	(変数名: ユーザ変数)	mdReceive
	(変数名: @LED)	mdBdLedRead
	(変数名: @Mode)	mdBdModRead
	(変数名: @SW)	mdBdSwRead
	(変数名: @VAR)	mdBdVerRead
	(変数名: @TYPE)	mdTypeRead

3. サンプルプログラム

以下に設定でデータを読み書きするサンプルを示します。

チャンネル : 81
局番 : 255
デバイスタイプ : 2
アドレス : 24
データサイズ : 2

List 3-1 Sample.frm

```
Private eng As CaoEngine
Private ctrl As CaoController
Private ext As CaoExtension
Private var As CaoVariable

Private Sub Form_Load()

Dim ws As CaoWorkspace

Set eng = New CaoEngine
Set ws = eng.Workspaces(0)

' MELSEC と接続
Set ctrl = ws.AddController("Sample", "CaoProv.MELSEC", "", "CH=81")

' 変数の生成
Set var = ctrl.AddVariable("255:2:24", "Elements=2")

End Sub

' データの設定
Private Sub Command1_Click()
Dim Val(0) As Integer
Val(0) = 10
var = Val
End Sub

' データの取得
Private Sub Command2_Click()
Dim Val() As Integer
Val = var
Text1.Text = Val(0)
End Sub
```