

# YASKAWA

## 拡張 MEMOBUS プロバイダ

Version 1.0.2

## ユーザーズ ガイド

January 21, 2022

【備考】

### 【改版履歴】

バージョン	日付	内容
1.0.0	2020-06-30	初版.
	2020-10-20	レジスタ種別の説明, データ種別の詳細, データ種別ごとに指定可能な VT オプション一覧を追加
1.0.1	2021-05-19	ランダムアクセスの不具合を修正
1.0.2	2021-05-27	put_Value のデータ型による動作不具合を修正
	2021-06-30	VT オプションの説明を修正
	2022-01-21	Reference オプションのアドレス値範囲を修正

### 【対応機器】

機種	バージョン	注意事項
MP900 シリーズ		
MP2000 シリーズ		
MP3000 シリーズ		

### 【動作確認機器】

機種	バージョン	注意事項
MP3300		

## 目次

1. はじめに.....	4
2. プロバイダの概要 .....	4
2.1. 概要 .....	4
2.2. メソッド・プロパティ.....	6
2.2.1. CaoWorkspace::AddController メソッド .....	6
2.2.1.1. Conn オプション .....	7
2.2.2. CaoController::get_VariableNames プロパティ .....	8
2.2.3. CaoVariable:: AddVariable メソッド .....	9
2.2.3.1. Reference オプション .....	11
2.2.3.2. Elem オプション .....	15
2.2.3.3. VT オプション .....	16
2.2.3.4. Array オプション .....	17
2.2.3.5. 戻り値の型 .....	18
2.2.4. CaoVariable::get_Value プロパティ .....	19
2.2.5. CaoVariable::put_Value プロパティ .....	20
2.3. エラーコード .....	21
3. サンプルプログラム .....	23
4. 通信プロトコルコマンド対応表 .....	24
5. 付録 アクセス上限 .....	25

## 1. はじめに

本書は拡張 MEMOBUS プロトコルに対応した機器(株式会社安川電機社製 MP3000/2000/900 シリーズ等. 以降, 拡張 MEMOBUS 機器と呼称します.) に対してレジスタ内容の一括書込/一括読込/ランダム書込/ランダム読込を行う CAO プロバイダのユーザーズガイドです.

本書で扱う CAO プロバイダ(CaoProvYaskawaExMemobus.dll)を拡張 MEMOBUS プロバイダと呼びます. 本プロバイダは株式会社安川電機から提供された「マシンコントローラ MP3000 シリーズ 通信機能 ユーザーズマニュアル」「マシンコントローラ MP2000 シリーズ 通信モジュール ユーザーズマニュアル」「マシンコントローラ MP900 シリーズ ユーザーズマニュアル 通信モジュール編」(以降, 拡張 MEMOBUS 通信マニュアルと呼称します.) に沿って開発しています. 本書は拡張 MEMOBUS プロバイダの機能と, 実装されているメソッドについて説明します.

## 2. プロバイダの概要

### 2.1. 概要

図 2-1 にプロバイダの各クラスとレーザマーカの対応を示します. CaoProvController は拡張 MEMOBUS 機器本体に, CaoProvVariable は, アクセス先レジスタに対応しています.

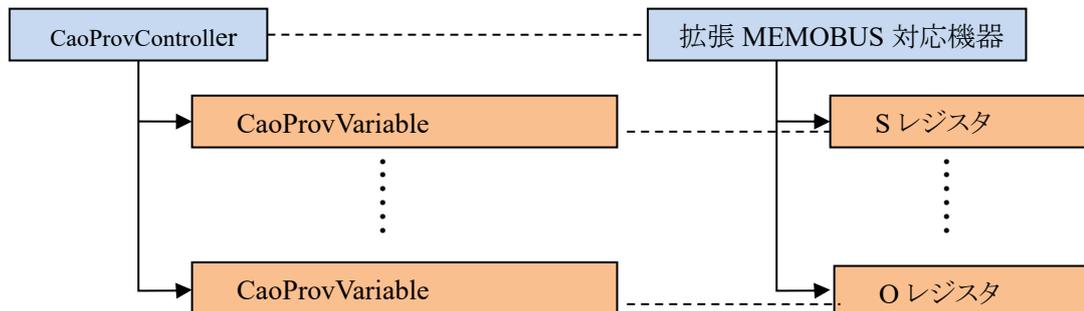


図 2-1 プロバイダの各クラスと拡張 MEMOBUS 機器の対応図

図 2-2 にプロバイダと拡張 MEMOBUS 機器との接続を示します. 接続対象とする機器には, Ethernet で接続してください.

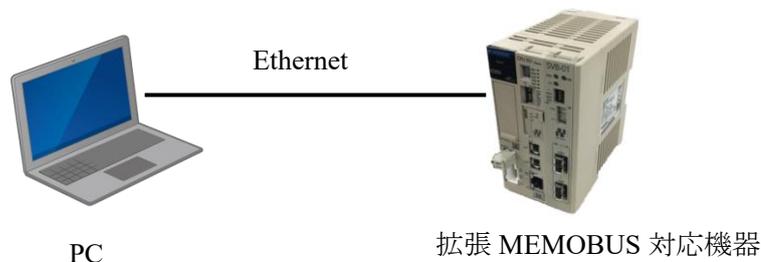


図 2-2 プロバイダと拡張 MEMOBUS 機器との接続図

拡張 MEMOBUS プロバイダのファイル形式は DLL(Dynamic Link Library)であり, CAO エンジンから使用時に動的にロードされます. 拡張 MEMOBUS プロバイダを使用するにあたっては ORiN2SDK をインストールするか, 表 2-1 を参照して手作業でレジストリ登録を行う必要があります.

表 2-1 拡張 MEMOBUS プロバイダのファイル形式

ファイル名	CaoProvYaskawaExMemobus.dll
ProgID	CaoProv.YASKAWA.ExMEMOBUS
レジストリ登録 <sup>1</sup>	regsvr32 CaoProvYaskawaExMemobus.dll
レジストリ登録の抹消	regsvr32 /u CaoProvYaskawaExMemobus.dll

<sup>1</sup> ORiN SDK でインストールした場合は手動で登録/抹消する必要はありません.

## 2.2. メソッド・プロパティ

### 2.2.1. CaoWorkspace::AddController メソッド

本プロバイダでは、CaoWorkspace::AddController で接続パラメータを設定して TCP 通信の接続を行います。以下に AddController の仕様を示します。

また、接続機器により AddController メソッドで同時接続可能な数が異なります。

#### 書式

```
AddController(
    BSTR bstrCtrlName,           // [in] コントローラ名(任意のコントローラ名を入力して下さい))
    BSTR bstrProvName,         // [in] プロバイダ名. 固定値 = " CaoProv.YASKAWA.ExMEMOBUS "
    BSTR bstrPcName,          // [in] プロバイダの実行マシン名
    BSTR bstrOption           // [in] オプション文字列
);
```

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

設定項目	オプション文字列	必須	説明	設定範囲	デフォルト値
接続先	Conn	○	接続先の IP アドレスとポート番号を指定してください。記述方法については、2.2.1.1Conn オプションを参照してください。	2.2.1.1Conn オプション参照	2.2.1.1Conn オプション参照
タイムアウト	Timeout	-	接続先とのタイムアウトを ms で指定してください。	1~4294967295	3000
レジスタアドレス容量	Bit	○	接続機器のアドレス容量を 16 or 32 で指定してください。 詳細は各機器のマニュアルを参照してください。 (例)MP3000 シリーズ:32, MP2000/900 シリーズ:16	16, 32	-
接続先 CPU 番号	CPUNo	-	接続機器の CPU 番号を指定してください。 2020 年 4 月以前に発売された機器は 1 固定, それ以外の機器に関しては各機器のマニュアルを参照してください。	1~15	1

### 2.2.1.1. Conn オプション

以下に Conn オプションの接続パラメータ文字列を示します.

"tcp:<A>:[<B >]"<sup>2</sup>

パラメータ	意味	必須	設定範囲	デフォルト値
A	接続先 IP アドレス	○	0.0.0.0~255.255.255.255	-
B	接続先ポート番号	-	1~65535	10001

例 1)MP3000 シリーズ( IP アドレス 192.168.1.1, ポート番号 10001)へ接続する場合

"Conn = tcp:192.168.1.1, Bit = 32", "Conn = tcp:192.168.1.1:10001, Bit = 32"

例 2) MP2000/900 シリーズ( IP アドレス 192.168.1.1, ポート番号 10001)へタイムアウト 1000 ms で接続する場合

"Conn = tcp:192.168.1.1, Timeout = 1000, Bit = 16"

"Conn = tcp:192.168.1.1:10001, Timeout = 1000, Bit = 16"

#### 使用例(CaoScript)

##### ・MP3000 シリーズへの接続

Dim ctrl

```
Set ctrl = cao.AddController("ExMemobus", "CaoProv.YASKAWA.ExMEMOBUS", "", "Conn = tcp:192.168.1.1:10001, Bit=32")
```

##### ・MP2000/900 シリーズへの接続

Dim ctrl

```
Set ctrl = cao.AddController("ExMemobus", "CaoProv.YASKAWA.ExMEMOBUS", "", "Conn = tcp:192.168.1.1:10001, Bit=16")
```

<sup>2</sup>角括弧("[ ]")内は省略可能なパラメータです.

## 2.2.2. CaoController::get\_VariableNames プロパティ

本プロバイダで使用可能な変数リストを取得します。表 2-3 のリストが取得できます。

戻り値の型	説明
VT_ARRAY   VT_BSTR	CaoController の AddVariable の変数名に使用できる変数一覧が取得できます。

### 使用例(CaoScript)

```
Dim vntNamesList
vntNamesList = dat.ToVar(ctrl.VariableNames)
dbg.output vntNamesList(0)           '⇒表示結果例:@MAKER_NAME
dbg.output vntNamesList(1)           '⇒表示結果例:@VERSION
dbg.output vntNamesList(2)           '⇒表示結果例:@LAST_PACKET_SIZE
```

表 2-3 CaoController クラス システム変数一覧

変数名	データ型	説明	属性	
			get	put
@MAKER_NAME	VT_BSTR	メーカー名「YASKAWA Electric Corporation」を返します。	○	-
@VERSION	VT_BSTR	プロバイダのバージョンを返します。	○	-
@LAST_PACKET_SIZE	VT_UI4	最後に送信したパケットサイズを返します。	○	-

### 2.2.3. CaoVariable:: AddVariable メソッド

CaoControllerからCaoVariableオブジェクトを生成します。システム変数を使用する場合は表 2-3 の各変数名に対応した文字列を、ユーザー変数の場合は任意文字列を変数名として入力してください。オプション文字列に関しては、システム変数を使用する場合は空白を、ユーザー変数を使用する場合は表 2-4 のオプション文字列に対応した文字列を入力してください。

#### 書式

```
AddVariable (  
    BSTR bstrVarName, // [in] 変数名.  
    BSTR bstrOption   // [in] オプション文字列  
);
```

#### 使用例(CaoScript)

```
Dim varMW0  
Set varMW0 = ctrl.AddVariable("MW0", "Reference=MW0, Elem=30, Array=TRUE")  
Dim varRandom  
Set varRandom = ctrl.AddVariable("Random", "Reference=MW10:SL30:GF78:ODFF")
```

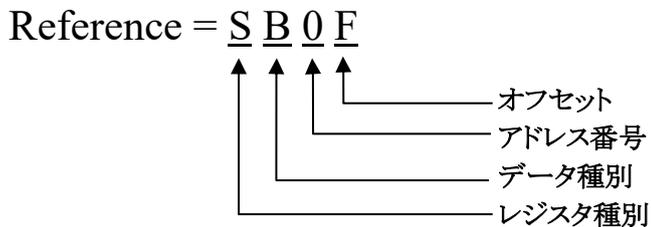
表 2-4 CaoController::AddVariable のオプション文字列

設定項目	オプション文字列	必須	説明	設定範囲	デフォルト値
参照レジスタ	Reference	○	読込/書込を行うレジスタ種別, データ種別, アドレス番号を指定します. 記述方法については, 2.2.3.1.Reference オプションを参照してください.	2.2.3.1.Reference オプション参照	-
要素数	Elem	-	読込/書込を行う要素数を指定します. 記述方法については, 2.2.3.2. Elem オプションを参照してください.	参照レジスタに依存	1
変数型	VT	-	データを文字列で扱いたい場合に指定します.記述方法については, 2.2.3.3.VT オプションを参照してください.	2.2.3.3.VT オプション参照	参照レジスタに依存
配列型	Array	-	一要素(Elem=1)の読み込み時に配列型で値を取得するかどうかを指定します. 記述方法については, 2.2.3.4.Array オプションを参照してください.	FALSE:非配列 TRUE:配列	FALSE
コードページ	CodePage	-	読込/書込を行う際に使用するコードページを指定します.  以下, 代表的なコードページを記述します. 0:現在の WindowsANSI コードページ 932:Shift-JIS 1200:UTF-16 12000:UTF-32 65000:UTF-7 65001:UTF-8	0 ~ 65535	0

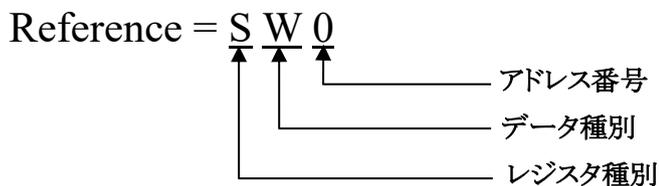
### 2.2.3.1. Reference オプション

読込/書込を行うレジスタ種別, データ種別, アドレス番号, オフセットを以下の形式で指定します.

[データ種別:B の場合]



[データ種別:B 以外の場合]



Reference オプションで指定可能なレジスタ種別, データ種別などは CaoWorkspace::AddController で指定したレジスタ容量によって異なります. レジスタ容量を 16 ビットに指定している場合は表 2-5, 32 ビットに指定している場合は表 2-6 を参照してください. また, 各レジスタ種別の説明は表 2-7 を, データ種別の詳細は表 2-8 を参照してください.

また, 各表のランダム可否が「○」の Reference オプションを[:]区切りで複数指定(以降, ランダムアクセスと呼称します.) することも可能です.

以下に Reference オプションの使用例を示します.

"Reference=<アクセス先 1>[:<アクセス先 2>]…[:<アクセス先 n>]"<sup>3</sup>

- |                             |   |
|-----------------------------|---|
| 例 1) "Reference = MW10"     | MW10 の値を整数値として参照します.                              |
| 例 2) "Reference = MB10A"    | MB10A の値をビット値として参照します.                            |
| 例 3) "Reference = SL10"     | SL10 の値を倍長整数値として参照します.                            |
| 例 4) "Reference = GQ10"     | GQ10 の値を 4 倍長整数値として参照します.                         |
| 例 5) "Reference = IFAB"     | IFAB の値を実数値として参照します.                              |
| 例 6) "Reference = OD20"     | OD20 の値を倍精度実数値として参照します.                           |
| 例 7) "Reference = MW10:SL3" | MW10 の値を整数値で, SL3 の値を倍長整数値で同時参照します.<br>(ランダムアクセス) |

<sup>3</sup> 角括弧("[ ]")内は省略可能なパラメータです.

表 2-5 16ビットアドレス Reference オプション指定方法

1文字目 レジスタ種別	2文字目 データ種別	3文字目～最終文字		ランダム 可否	データ 型	属性	
		アドレス番号 <sup>4</sup>	オフセット(16進) <sup>5</sup>			get	put
M	B	0~4095(10進)	0~F	×	VT_I2	○	○
	W	0~65535(10進)	-	○	VT_I2	○	○
	L				VT_I4		
	Q				VT_I8		
	F				VT_R4		
	D				VT_R8		
I	B	0 ~ FFF(16進)	0~F	×	VT_I2	○	×
	W	0 ~ FFFF(16進)	-	×	VT_I2	○	×
	L				VT_I4		
	Q				VT_I8		
	F				VT_R4		
	D				VT_R8		

<sup>4</sup> アドレス番号の上限は拡張 MEMOBUS プロバイダとしての値です。実際の上限幅については各機器のマニュアルを参照してください。

<sup>5</sup> オフセットは指定したアドレスから何ビット目を指します。1アドレスのサイズは1ワードです。

表 2-6 32 ビットアドレス Reference オプション指定方法

1 文字目 レジスタ種別	2 文字目 データ種別	3 文字目～最終文字		ランダム 可否	データ 型	属性	
		アドレス番号 <sup>4</sup>	オフセット (16 進) <sup>5</sup>			get	put
S	B	0~268435455(10 進)	0~F	×	VT_I2	○	○
	W	0~4294967295(10 進)	-	○	VT_I2		
	L				VT_I4		
	Q				VT_I8		
	F				VT_R4		
	D				VT_R8		
M	B	0~268435455(10 進)	0~F	×	VT_I2	○	○
	W	0~4294967295(10 進)	-	○	VT_I2		
	L				VT_I4		
	Q				VT_I8		
	F				VT_R4		
	D				VT_R8		
G	B	0~268435455(10 進)	0~F	×	VT_I2	○	○
	W	0~4294967295(10 進)	-	○	VT_I2		
	L				VT_I4		
	Q				VT_I8		
	F				VT_R4		
	D				VT_R8		
I	B	0 ~ FFFFFFFF(16 進)	0~F	×	VT_I2	○	×
	W	0 ~ FFFFFFFF(16 進)	-	○	VT_I2		
	L				VT_I4		
	Q				VT_I8		
	F				VT_R4		
	D				VT_R8		
O	B	0 ~ FFFFFFFF(16 進)	0~F	×	VT_I2	○	○
	W	0 ~ FFFFFFFF(16 進)	-	○	VT_I2		
	L				VT_I4		
	Q				VT_I8		
	F				VT_R4		
	D				VT_R8		

表 2-7 レジスタ種別の説明

レジスタ種別	レジスタ種別名称	説明
S	システムレジスタ	システムで準備されているレジスタです。マシンコントローラなどが報告されます。 システム始動時, SW00000 ~ SW00049 は 0 クリアされます。 バッテリーバックアップされます。
M	データレジスタ	プログラム間の I/F などに使用するレジスタです。 バッテリーバックアップされます。
G	グローバルレジスタ	プログラム間の I/F などに使用するレジスタです。 バッテリーバックアップされません。
I	入力レジスタ	入力データに使用するレジスタです。
O	出力レジスタ	出力データに使用するレジスタです。

表 2-8 データ種別の詳細

データ種別	データ型	データサイズ	備考
B	ビット	ビット	リレー回路やオン/オフ条件の判定に使用します。
W	整数	1 ワード	数値演算で使用します。
L	倍長整数	2 ワード	数値演算で使用します。
Q	4倍長整数	4 ワード	数値演算で使用します。
F	実数	2 ワード	高度な数値演算で使用します。
D	倍精度実数	4 ワード	高度な数値演算で使用します。

### 2.2.3.2. Elem オプション

読込/書込を行う要素数を 10 進数で指定します。16 進数で指定する場合は「0x~」、「&H~」または「~H」の形式で指定してください。Elem オプションを省略した場合のデフォルトの値は 1 となります。Reference オプションが複数指定(ランダムアクセス)の場合、Elem オプションは無視されます。

以下に Elem オプションのパラメータ文字列を示します。

"[Elem = [<要素数>]]"<sup>6</sup>

<要素数 20 を指定する場合>

- |                                   |                                   |
|-----------------------------------|-----------------------------------|
| 例 1) "Reference=MW10, Elem=20"    | MW10 から MW29 の値を整数値として参照します。      |
| 例 2) "Reference=SL20, Elem=0x14"  | SL20 から SL29 の値を倍長整数値として参照します。    |
| 例 3) "Reference=GF1A, Elem=&H14"  | GF1A から GF2D の値を実数値として参照します。      |
| 例 4) "Reference=OB9001, Elem=14H" | OB9001 から OB9014 の値をビット値として参照します。 |

<sup>6</sup> 角括弧("[ ]")内は省略可能なパラメータです。

### 2.2.3.3. VT オプション

Get\_Value 実行時, 戻り値の型は Reference オプションで指定した表 2-5 または表 2-6 のデータ型となります。VT オプションに「BSTR」を指定することで, データを文字列として取得することが可能です。文字列のコードページは CodePage オプションで指定します。VT オプションの指定方法は, ランダムアクセスの場合と, 非ランダムアクセスの場合とで異なります。

- 非ランダムアクセスの場合

VT オプションは省略するか, BSTR を指定することが可能です。ただし, Reference オプションで指定したデータ種別が「B」の場合は指定することができません。

"[VT=<VT オプション文字列>]"<sup>7</sup>

例 1) "Reference=MW10"                      戻り値の型: VT\_I2

例 2) "Reference=MW10, VT=BSTR"        戻り値の型: BSTR

**表 2-9 非ランダムアクセス時に指定可能な VT オプションの一覧**

Reference オプション データ種別	指定可能な VT オプション	データ型
B	-	-
W	BSTR	VT_BSTR
L	BSTR	VT_BSTR
Q	BSTR	VT_BSTR
F	BSTR	VT_BSTR
D	BSTR	VT_BSTR

- ランダムアクセスの場合

VT オプションは省略するか, [:] で区切り Reference オプションと同数指定する必要があります。一部のみの指定や省略はできませんのでご注意ください。指定可能な VT オプションは表 2-10 となります。

"[VT=<VT オプション文字列>[:<VT オプション文字列>]…[:<VT オプション文字列>]"<sup>7</sup>

例 1) "Reference=MW10:IL45, VT=I2:BSTR"        戻り値の型: VT\_VARIANT|VT\_ARRAY

例 2) "Reference=MW10:IL45"                      戻り値の型: VT\_VARIANT|VT\_ARRAY

例 3) "Reference=MW10:IL45, VT=BSTR:BSTR"    戻り値の型: VT\_VARIANT|VT\_ARRAY

<sup>7</sup> 角括弧("[ ]")内は省略可能なパラメータです。

表 2-10 ランダムアクセス時に指定可能な VT オプションの一覧

Reference オプション データ種別	指定可能な VT オプション	データ型
W	I2	VT_I2
	BSTR	VT_BSTR
L	I4	VT_I4
	BSTR	VT_BSTR
Q	I8	VT_I8
	BSTR	VT_BSTR
F	R4	VT_R4
	BSTR	VT_BSTR
D	R8	VT_R8
	BSTR	VT_BSTR

#### 2.2.3.4. Array オプション

Elem オプションが 1 の場合に読み込んだ値を配列形式で取得するかを指定します。Array オプションを省略した場合のデフォルトの値は False となります。ランダムアクセスまたは VT オプションで BSTR 指定した場合、Array オプションは無視されます。

以下に Array オプションのパラメータ文字列を示します。

"[Array=[< True or False >]]"

(例) "Reference=MW10"

戻り値の型: VT\_I2

(例) "Reference=MW10, Array=FALSE"

戻り値の型: VT\_I2

(例) "Reference=MW10, Array=TRUE"

戻り値の型: VT\_I2 | VT\_ARRAY

### 2.2.3.5. 戻り値の型

Get\_Value 実行時, 戻り値の型は Reference オプションで指定した表 2-5 または表 2-6 のデータ型となります。VT オプションを指定した場合, VT オプションの型が優先されます。

戻り値が配列形式になるかはランダムアクセス, Elem オプション, Array オプションの組み合わせに依存します。

配列形式の有無を表 2-11 に示します。

**表 2-11 配列形式の有無**

ランダムアクセス <sup>8</sup>	Elem	Array	配列形式有無
無	1	FALSE	×
		TRUE	○
	2 以上	-	○
有	-	-	○

例 1) "Reference=MW10"

戻り値の型: VT\_I2

例 2) "Reference=MW10, Array=FALSE"

戻り値の型: VT\_I2

例 3) "Reference=MW10, Array=TRUE"

戻り値の型: VT\_I2 | VT\_ARRAY

例 4) "Reference=MW10, Elem=3, Array=FALSE"

戻り値の型: VT\_I2 | VT\_ARRAY

例 5) "Reference=MW10:MW20, Array=FALSE"

戻り値の型: VT\_VARIANT | VT\_ARRAY

<sup>8</sup> Reference オプションの複数指定有無

## 2.2.4. CaoVariable::get\_Value プロパティ

AddVariable メソッドで作成した変数の状態を取得します。戻り値の型は AddVariable で指定したオプションデータによって異なります。詳細は 2.2.3.CaoWorkspace::AddController メソッドを参照ください。

### 使用例(CaoScript)

```
Dim varMW0
```

```
Set varMW0 = ctrl.AddVariable("MW0", "Reference=MW0")
```

```
dbg.Output varMW0.value           '⇒表示結果例:100
```

```
Dim varMultiMW
```

```
Dim vntMultiMW
```

```
Set varMultiMW = ctrl.AddVariable("MultiMW", "Reference=MW0, Elem=3")
```

```
vntMultiMW = DAT.ToVar (varMultiMW.Value)
```

```
dbg.Output vntMultiMW(0)         '⇒表示結果例:100
```

```
dbg.Output vntMultiMW(1)         '⇒表示結果例:50
```

```
dbg.Output vntMultiMW(2)         '⇒表示結果例:200
```

```
Dim varRandom
```

```
Dim vntRandom
```

```
Set varRandom = ctrl.AddVariable("Random", "Reference=MW10:SL30:GF78:ODFF")
```

```
vntRandom = DAT.ToVar (varRandom.Value)
```

```
dbg.Output vntRandom(0)         '⇒表示結果例:1000
```

```
dbg.Output vntRandom(1)         '⇒表示結果例:100000
```

```
dbg.Output vntRandom(2)         '⇒表示結果例:-4.316021E+008
```

```
dbg.Output vntRandom(3)         '⇒表示結果例:-3.386099378622854E+216
```

### 2.2.5. CaoVariable::put\_Value プロパティ

AddVariable メソッドで作成した変数に値を設定します。AddVariable で表 2-5, 表 2-6 で put に「○」がついているオプションを指定した場合、put\_Value プロパティを使用可能です。設定可能なデータサイズは Reference オプションのデータ種別により異なります。表 2-12 にデータ種別毎に設定可能なデータサイズを示します。また、設定するデータ数に関しては Elem オプションの要素数またはランダムアクセス数より多く指定された場合、超過分のデータは無視されます。

表 2-12 設定可能なデータサイズ

データ種別	MIN	MAX
B	0(False)	0 以外(True)
W	-32788	32767
L	-2147483648	2147483647
Q	-9223372036854775808	9223372036854775807
F	-3.402823E+038	3.402823E+038
D	-1.797693E+308	1.797693E+308

#### 使用例(CaoScript)

```
Dim varMW0
```

```
Set varMW0 = ctrl.AddVariable("MW0", "Reference=MW0")
```

```
varMW0.value = 100
```

```
Dim varMultiMW
```

```
Dim arrMultiMW
```

```
Set varMultiMW = ctrl.AddVariable("MultiMW", "Reference=MW0, Elem=3")
```

```
arrMultiMW = Array(100, 50, 200)
```

```
varMultiMW.Value = DAT.ChangeType(arrMultiMW, 2)
```

```
Dim varRandom
```

```
Dim arrRandom
```

```
Set varRandom = ctrl.AddVariable("Random", "Reference=MW10:SL30:GF78:ODFF")
```

```
arrRandom = Array(1, 2, 3, 4)
```

```
varRandom.Value = DAT.ChangeType(arrRandom, 2)
```

### 2.3. エラーコード

本プロバイダでは固有エラーコードを定義しています。固有エラーコードを表 2-13 に示します。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。<sup>9</sup>

表 2-13 固有エラーコード

エラー番号	説明
0x80100000	AddController メソッドで指定したオプション文字列 Conn が不正です。Conn = tcp:<IP アドレス>[:<ポート番号>]の形式で指定してください。IP アドレスを 0.0.0.0 ~ 255.255.255.255 の範囲、ポート番号を 1 ~ 65535 の範囲で指定してください。
0x80100001	AddController メソッドで指定したオプション文字列 Timeout が不正です。1 ~ 4294967295 ms の範囲で指定してください。
0x80100002	AddController メソッドで指定したオプション文字列 Bit が不正です。16 または 32 を指定してください。
0x80100003	AddController メソッドで指定したオプション文字列 CPUNo が不正です。0 ~ 15 の範囲で指定してください。
0x80100004	AddVariable メソッドで指定したオプション文字列 Reference が不正です。Reference=<アクセス先 1>[:<アクセス先 2>]…[:<アクセス先 n>]の形式で指定してください。 表 2-5 および表 2-6 に従って指定してください。 ランダムアクセスの場合、データ種別で'B'は指定できません。データ種別に'B'を指定している場合、オフセットを指定してください。
0x80100005	AddVariable メソッドで指定したオプション文字列 VT が不正です。VT=[<VT オプション文字列>][:<VT オプション文字列>]…[:<VT オプション文字列>]の形式で指定してください。 表 2-10 に従って指定してください。 Reference オプションの指定数と同数であることを確認してください。
0x80100006	AddVariable メソッドで指定したオプション文字 Elem が不正です。1 ~ 4294967295 の範囲で指定してください。 16 進表記の場合"0x~","&H~","~H"を使用してください。
0x80100007	AddVariable メソッドで指定したオプション文字列 Array が不正です。"TRUE"または"FALSE"を指定してください。

<sup>9</sup> 通信が確立されている場合にタイムアウトエラー(0x80000900)が発生した場合、システム変数[@LAST\_PACKET\_SIZE]から、機器に送信したパケットサイズを取得し、接続機器の最大パケットサイズを超えていないか確認してください。

0x80100008	AddVariable メソッドで指定したオプション文字列 Reference のランダムアクセス数が上限を超過しました。付録を参照し、指定数を見直してください。
0x80100009	AddVariable メソッドで指定したオプション文字列 Elem の指定数が上限を超過しました。付録を参照し、指定数を見直してください。
0x8010000A	Reference オプションで指定したレジスタ種別が'I'の場合、Put_Value は実行できません。
0x8010000B	Put_Value 実行時に指定されたデータ型が不正です。ランダムアクセス時は、配列型を指定してください。
0x8010000C	Put_Value 実行時に指定されたデータ数が不正です。アクセス先の要素数と同数を指定してください。
0x8010000D	読出データの型変換で範囲外の値を検出しました。 Get_Value 実行時に発生した場合、VT オプションで指定するデータ型が Reference オプションで指定するデータ種別のアクセス点数より小さい可能性があります。 Put_Value 実行時に発生した場合、指定したデータが Reference オプションで指定するデータ種別のアクセス点数より大きい可能性があります。 データ種別毎のアクセス点数は表 5-1 を参照してください。
0x8010000E	不正なパケットを受信したときに出力されるエラーコードです。WireSharkなどで応答パケットをキャプチャし、使用するユーザーズマニュアルからエラー原因を調査してください。
0x8010000F	AddVariable メソッドで指定したオプション文字列 CodePage が不正です。 1 ~ 65535 の範囲で指定してください。
0x801100xx	拡張 MEMOBUS 機器からエラーコードを受信しました。下 2 桁「xx」が受信したエラーコードです。使用するユーザーズマニュアルからエラー原因を調査してください。

### 3. サンプルプログラム

AddVariableのサンプルプログラムを以下に示します.

#### 使用例(CaoScript)

例) 32 ビットアドレスの拡張 MEMOBUS 機器 MW0 の値を書き換える.

```
' CaoController の生成
Dim ctrl
Set ctrl = cao.AddController("ExMemobus", "CaoProv.YASKAWA.ExMEMOBUS", "",
                                                                    "Conn = tcp:192.168.1.1:10001, bit=32")

' CaoVariable の生成
Dim varMW0
Set varMW0 = ctrl.AddVariable("MW0", "Reference=MW0 ")

' MW0 値を読み
dbg.Output varMW0.value           '⇒表示結果例:50

' MW0 値を書込
varMW0.value = 100
dbg.Output varMW0.value           '⇒表示結果例:100

' CaoVariable の破棄
If Not varMW0 Is Nothing Then
    ctrl.Variables.remove varMW0.Index
    Set varMW0 = Nothing
End If

' CaoController の破棄
If Not ctrl Is Nothing Then
    cao.Controllers.remove ctrl.Index
    Set ctrl = Nothing
End If
```

## 4. 通信プロトコルコマンド対応表

本プロバイダで実装している Variable 変数と拡張 MEMOBUS 通信マニュアルの通信コマンドの対応表を表 4-1 に示します。

表 4-1 通信コマンド対応表

レジスタ 容量	Reference オプション			get/put	通信コマンド		
	ランダム アクセス	レジスタ 種別	データ 種別		MFC <sup>10</sup>	SFC <sup>11</sup>	機能内容
16	○	-	-	get	20	0D	保持レジスタの不連続読み出し(拡張)
				put	20	0E	保持レジスタの不連続書き込み(拡張)
	×	M	B	get	20	01	コイルの状態読み出し
				put	20	0F	複数コイルの状態変更
			WLQFD	get	20	09	保持レジスタの内容読み出し(拡張)
				Put	20	10	複数保持レジスタへの書き込み
		I	B	get	20	02	入力リレーの状態読み出し
				put	-	-	-
			WLQFD	get	20	0A	入力レジスタの内容読み出し(拡張)
				Put	-	-	-
32	○	-	-	get	43	4D	レジスタの不連続読み出し
				put	43	0E	レジスタの不連続書き込み
	×	-	B	get	43	41	ビットデータの状態読み出し
				put	20	4F	複数ビットデータの状態変更
			WLQFD	get	20	49	レジスタの内容読み出し
				Put	20	4B	複数レジスタへの書き込み

<sup>10</sup> メジャーファンクションコード

<sup>11</sup> サブファンクションコード

## 5. 付録 アクセス上限

Reference オプションと Elem オプションで指定した、ランダムアクセス数とデータ型一要素あたりの点数、要素数により、アクセス点数を以下のように決定します。データ型毎の一要素あたりの点数を表 5-1 に示します。また、拡張 MEMOBUS プロトコルのアクセス点数の上限を表 5-2 に示します。機器毎のアクセス上限に関してはそれぞれのマニュアルを参照してください。

[ランダムアクセス以外の場合]

$$\text{アクセス点数} = (\text{データ型点数}) \times (\text{要素数})$$

[ランダムアクセスの場合]

$$\text{アクセス点数} = \text{各データ型点数の総和}$$

例 1) "Reference=MW10"	アクセス点数:2(=2×1)
例 2) "Reference=MF10, Elem=10"	アクセス点数:40(=4×10)
例 3) "Reference=MD10, Elem=100"	アクセス点数:800(=8×100)
例 4) "Reference=MW10:ML20:MQ30"	アクセス点数:14(=2+4+8)

表 5-1 データ型毎の一要素あたりの点数

データ型	点数
B	1
W	2
L	4
Q	8
F	4
D	8

表 5-2 アクセス上限値(拡張 MEMOBUS プロトコル)

Reference			アクセス上限			
ランダムアクセス 有無	1 文字目	2 文字目	16 ビットアドレス		32 ビットアドレス	
	レジスタ種別	データ種別	get	put	get	put
○	-	-	32757	16378	10919	8189
×	S	B	-		524113	524049
		W			32756	32754
		L			16378	16377
		Q			8189	8188
		F			16378	16377
		D			8189	8188
	M	B	524144	524105	524113	524049
		W	32757	32756	32756	32754
		L	16378	16378	16378	16377
		Q	8189	8189	8189	8188
		F	16378	16378	16378	16377
		D	8189	8189	8189	8188
	G	B	-		524113	524049
		W			32756	32754
		L			16378	16377
		Q			8189	8188
		F			16378	16377
		D			8189	8188
	I	B	524144	-	524113	-
		W	32757		32756	
		L	16278		16378	
		Q	8189		8189	
		F	16278		16378	
		D	8189		8189	
O	B	-		524113	524049	
	W			32756	32754	
	L			16378	16377	
	Q			8189	8188	
	F			16378	16377	
	D			8189	8188	