

アマダミヤチ
AMADA MM400 プロバイダ

Version 1.0.0

ユーザーズ ガイド

March 5, 2019

備考：

【改版履歴】

バージョン	日付	内容
1.0.0	2019-03-05	初版.

【動作確認機種】

機種	バージョン	注意事項
MM-400	A	

目次

1. はじめに.....	5
1.1. 本書が想定している環境とバージョン.....	6
1.2. 参考となる情報源.....	6
2. アプリケーション開発のための環境セットアップ.....	7
2.1. ウェルドチェッカーとの接続.....	7
2.1.1. 双方向通信方式で接続する.....	7
2.1.2. 片方向通信方式で接続する.....	8
2.2. PC 開発環境のセットアップ.....	9
2.2.1. MM400 プロバイダの自動インストール.....	9
2.2.2. MM400 プロバイダの手動インストール.....	9
3. コマンドリファレンス.....	10
3.1. メソッド/プロパティ一覧.....	10
3.2. メソッド・プロパティ.....	10
3.2.1. CaoWorkspace クラス.....	10
3.2.1.1. AddController メソッド.....	10
3.2.2. CaoController クラス.....	12
3.2.2.1. VariableNames プロパティ.....	12
3.2.2.2. AddVariable メソッド.....	12
3.2.2.3. OnMessage イベント.....	12
3.2.3. CaoVariable クラス.....	12
3.2.3.1. Value プロパティ.....	12
3.3. 変数一覧.....	13
3.3.1. CaoController クラス変数.....	13
3.3.1.1. @MAKER_NAME.....	14
3.3.1.2. @VERSION.....	15
3.3.1.3. @MEASUREMENT.....	15
3.3.1.4. @WAVEFORM.....	16
3.3.1.5. @CURRENT_ALLCYCLE.....	19
3.3.1.6. @MEASUREMENT_HISTORY.....	21
3.3.1.7. @ALARM_HISTORY.....	22

3.3.1.8. @DISPLAY_SETTING	22
3.3.1.9. @SETTING_CONDITION	24
3.3.1.10. @EXTERNAL_IO	25
3.3.1.11. @PRINTING	27
3.3.1.12. @COMMUNICATION	28
3.3.1.13. @USB	31
3.3.1.14. @MEMORY	32
3.3.1.15. BASIC_CONDITION<??>	33
3.3.1.16. EXT_CONDITION<??>	36
3.3.1.17. SEAM_CONDITION<??>	42
3.3.1.18. RANGE_SETTING<??>	45
3.3.1.19. ENVELOPE<??>	47
3.4. メッセージ一覧	52
4. MM400 プロバイダによるプログラミング	53
4.1. 測定を自動受信するサンプルプログラミング	53
4.1.1. サンプルプログラム	54
4.1.1.1. 接続	55
4.1.1.2. メッセージ受信	57
4.1.1.3. 切断	57
4.2. 測定を取得するサンプルプログラミング	58
4.2.1. サンプルプログラム	59
4.2.1.1. 接続	60
4.2.1.2. 測定の取得	62
4.2.1.3. 切断	62
5. MM400 プロバイダエラーコード	64
6. 付録	65

1. はじめに

本書は、アマダミヤチの MM400 シリーズウェルドチェッカーに対して測定情報の受信や各種設定の送受信を行うプロバイダのユーザーズガイドです。図 1-1 が本プロバイダとウェルドチェッカーの全体構成図になります。以降本プロバイダを MM400 プロバイダ、MM400 シリーズウェルドチェッカーをウェルドチェッカーと呼称します。MM400 プロバイダは「ウェルドチェッカーMM400-A 取扱説明書」のバージョン「W120M1192557-07」に記述されている Ethernet (TCP/IP) の通信仕様を元に作成されています。以降、単に「取扱説明書」と記述した場合は上記の取扱説明書を指すものとします。

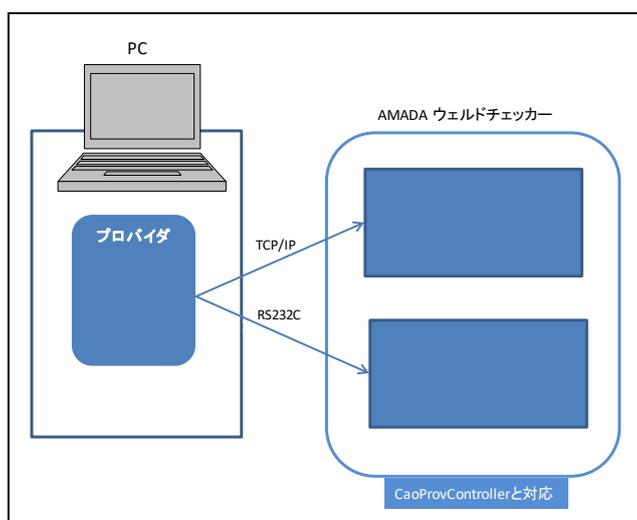


図 1-1 構成図

また、MM400プロバイダとウェルドチェッカーの対応を図 1-2に表します。
(※一例です。全てを表しているわけではありません。)

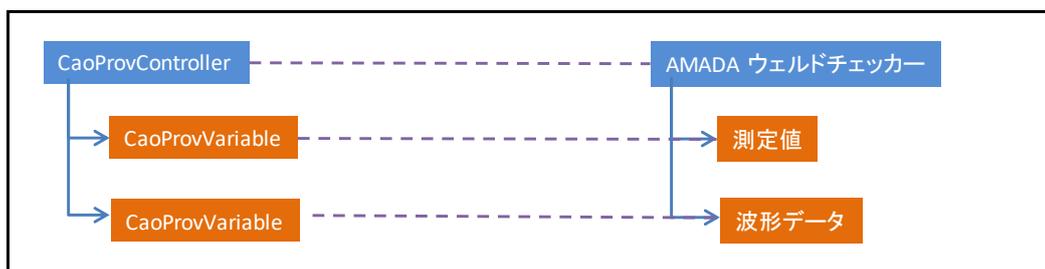


図 1-2 MM400 プロバイダの構成とウェルドチェッカー情報との対応図

1.1. 本書が想定している環境とバージョン

クライアント PC が Windows 上で動作し、対象とする CNC がイーサネット接続可能な MM400 シリーズのウェルドチェッカーであり、バージョン「W120M1192557-07」である環境を想定しています。PC の開発環境は、Component Object Model (COM, コンポーネント・オブジェクト・モデル) をサポートするプログラミング環境であれば開発が可能です。

1.2. 参考となる情報源

本書のプログラミング事例は、すべて Visual Basic 6.0 で記載していますが、C++, Java, .NET などさまざまなプログラム言語で開発が可能です。使用方法に関しては、「ORiN2 プログラミングガイド」を参照してください。

「ORiN2 プログラミングガイド」は ORiN2 SDK インストールフォルダの以下のファイルに該当します。

- ORiN2¥CA0¥Doc¥ORiN2_ProgrammersGuide_<lang>.pdf

※<lang>の部分は環境毎の言語文字列に置き換えてお読みください。

プロバイダを使ったアプリケーションを開発する上で必要となる ORiN2, COM/DCOM の基礎知識や技術に関して例を交えながら解説されています。

2. アプリケーション開発のための環境セットアップ

2.1. ウェルドチェッカーとの接続

ウェルドチェッカーとはTCP/IPにて接続することが可能です。また、以下の2つの通信方式が存在しており、ウェルドチェッカーの設定に合わせてMM400プロバイダのCaoControllerを作成する必要があります。Controllerの作成方法については3.2.1.1を参照してください。

1. 双方向通信方式

クライアントからの要求に応じてデータ通信を行う方式です。この方式を選択されているウェルドチェッカーは、クライアントから要求されたときのみデータ通信を行い、測定ごとに自動でメッセージを送信することはできません。

2. 片方向通信方式

ウェルドチェッカーがトリガーとなりデータ通信を行う方式です。この方式を選択されているウェルドチェッカーは、通信(1)画面の項目で選択されたデータを、「インターバル」、「判定外動作」の条件で測定ごとにクライアントに対して一方的にメッセージを送信し、クライアントからデータ送信要求を行うことはできません。

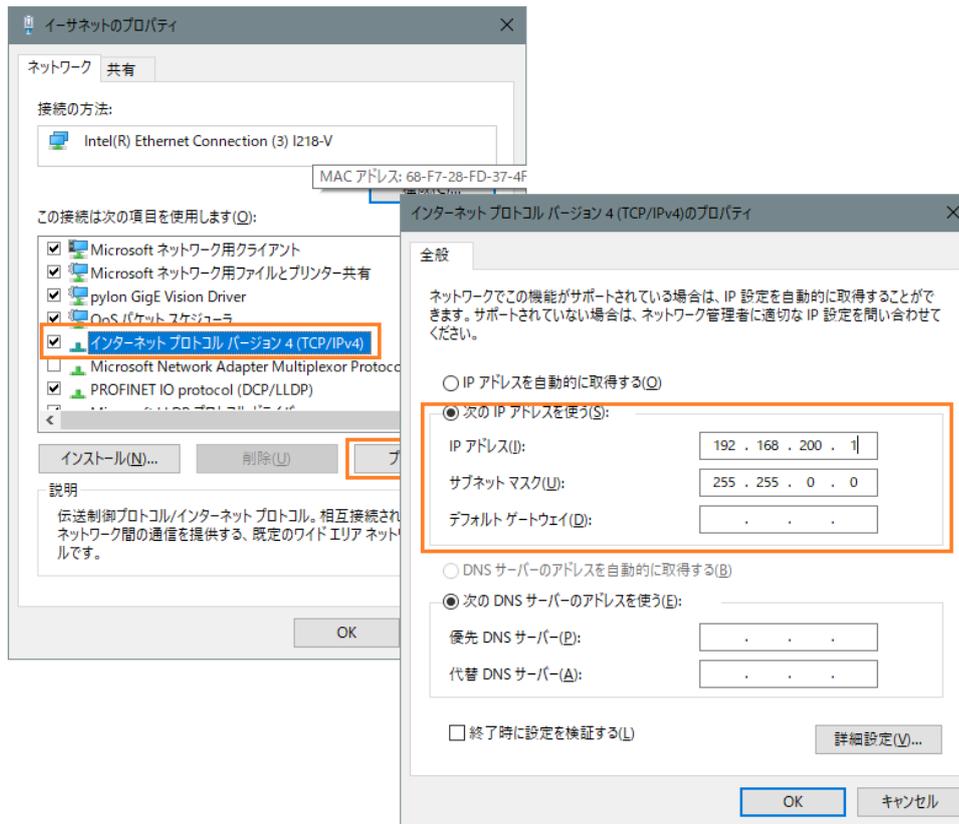
ここではWindows10のクライアントPCを使用して、ウェルドチェッカーとTCP/IPで接続する方法を記述します。

2.1.1. 双方向通信方式で接続する

- ① ウェルドチェッカーの[メニュー]→[通信]→[通信(2)]画面を表示し、方式で「ETHERNET」「双方向」を選択し、IPアドレス、サブネットマスク、ポート番号を確認します。

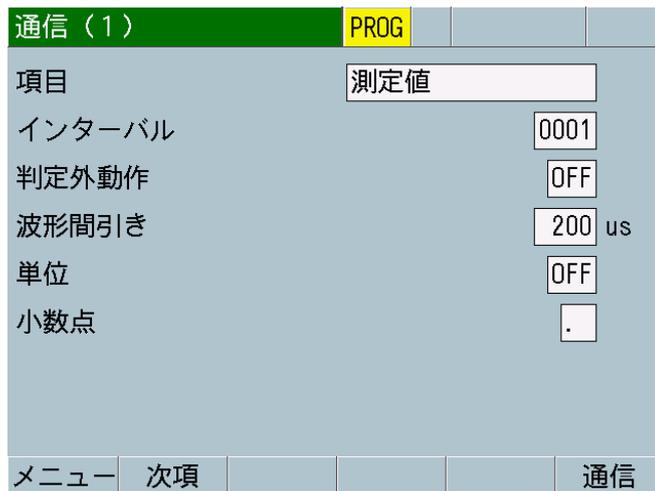
通信(2)		PROG	USB
方式	ETHERNET		
	双方向		
装置番号	01		
IPアドレス	192	168	001 . 010
サブネットマスク	255	255	255 . 000
デフォルトゲートウェイ	192	168	001 . 100
ポート番号	1024		
MACアドレス	00-60-d5-03-00-00		
メニュー	前項		

- ② クライアントPCの使用したい通信ポートのプロパティから、[インターネットプロトコルバージョン4(TCP/IPv4)]→[プロパティ]ボタンを実行し、ウェルドチェッカー側の通信設定に合わせてクライアントPCのIPアドレス、サブネットマスクの設定を行います。



2.1.2. 片方向通信方式で接続する

- ① ウェルドチェッカーの[メニュー]→[通信]→[通信(1)]画面を表示し、項目から取得したい項目を選択します。



- ② ウェルドチェッカーの[メニュー]→[通信]→[通信(2)]画面を表示し、方式で「ETHERNET」「双方向」を選択し、IPアドレス、サブネットマスク、ポート番号を確認します。
- ③ クライアントPCの使用したい通信ポートのプロパティから、[インターネットプロトコルバージョン4(TCP/IPv4)]→[プロパティ]ボタンを実行し、ウェルドチェッカー側の通信設定に合わ

せてクライアント PC の IP アドレス、サブネットマスクの設定を行います。

2.2. PC 開発環境のセットアップ

2.2.1. MM400 プロバイダの自動インストール

ORiN2 SDK がインストールされている環境であれば、ウェルドチェッカーに接続するための動作環境（ラインタイム）の準備は完了です。

開発環境のセットアップは別途、Microsoft Visual Studio 6.0, 2003/2005/2008/2010, LabVIEW など Component Object Model (COM, コンポーネント・オブジェクト・モデル) をサポートする、プログラミング環境をご準備してください。

2.2.2. MM400 プロバイダの手動インストール

MM400 プロバイダを手作業でインストールする場合は、下記レジストリ登録を行う必要があります。レジストリ登録を行う場合は、管理者権限でコマンドプロンプトを起動し、regsvr32 コマンドを実行してください。

また、CAO エンジンが動作するには予め、PC 毎に正規の ORiN2 SDK ライセンスが1つ登録されていなくてはなりません。ORiN2 SDK ユーザーズガイド内にある「ライセンスの追加と削除」の節を参照してください。

表 2-A MM400 プロバイダ

ファイル名	CaoProvAMADAMM400.dll
ProgID	CaoProv.AMADA.MM400
レジストリ登録	regsvr32 CaoProvAMADAMM400.dll
レジストリ登録の抹消	regsvr32 /u CaoProvAMADAMM400dll

3. コマンドリファレンス

3.1. メソッド/プロパティ一覧

表 3-A メソッド/プロパティ一覧

カテゴリ	メソッド/プロパティ ¹	機能	参照
CaoWorkspace			
	Addcontroller	M コントローラに接続	P. 10
CaoController			
	VariableNames	P 接続可能な変数名リストの取得	P. 12
	AddVariable	M 変数オブジェクトの作成	P. 12
	OnMessage	E メッセージ受信イベント	P. 12
CaoVariable			
	Value	P 値の取得/設定	P. 12

3.2. メソッド・プロパティ

3.2.1. CaoWorkspace クラス

3.2.1.1. AddController メソッド

CaoWorkspace に、コントローラオブジェクトを追加します。MM400 プロバイダでは、AddController メソッド実行時に渡されたパラメータを参照し、該当するウエルドチェッカーと接続を行います。以下に、AddController メソッドの仕様を示します。

書式

CaoController AddController

```
(
    "<コントローラ名>",           // コントローラ名(任意)
    "CaoProv. AMADA. MM400",     // プロバイダ名(固定)
    "<マシン名>",               // プロバイダ実行マシン名(未使用)
    "<オプション>"              // オプション文字列
)
```

使用例

```
Dim engine As CaoEngine      ' Engineオブジェクト
Dim workspace As CaoWorkspace ' WorkSpaceオブジェクト
Dim controller As CaoController ' Controllerオブジェクト
```

¹ M:メソッド, P:プロパティ, E:イベントをそれぞれ示します。

```

Set engine = New CaoEngine
Set workspace = engine.CaoWorkspaces.Item(0)
' CaoControllerオブジェクトの作成(双方向)
Set controller = workspace.AddController("AMADA", _
                                         "CaoProv. AMADA. MM400", _
                                         "", _
                                         "conn=tcp:192.168.1.10:1024, Mode=2")

```

オプション

以下にオプション文字列に指定するオプションを示します。オプション文字列は下記に示す各オプションをカンマ(,)でつなげた文字列となります。

オプション	必須	説明	値範囲	デフォルト値
Conn	○	ウェルドチェッカーに設定されている通信パラメータを指定します。 詳細は 3.2.1.1.1 を参照してください。	-	-
Timeout	--	応答待機時間 ms 単位で指定します。	1 - 2147483647	500
Mode	--	ウェルドチェッカーに設定されている通信方式を指定します。	1: 単方向 2: 双方向	1
ID	--	ウェルドチェッカーに設定されている装置 No を指定します。 本オプションは Mode オプションに双方向を指定したときのみ有効です。	1 - 255	1
DecimalPoint	--	ウェルドチェッカーに設定されている小数点種別を指定します。	1: ピリオド(.) 2: カンマ(,)	1

3.2.1.1.1. Conn オプション

前述したとおり、ウェルドチェッカーは TCP/IP による通信に対応しています。本プロバイダはこの通信方式に対応するため、Conn オプションで指定された通信方式で射出成形機との接続を試みます。

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。ここで中括弧("[]")内は省略可能なことを、各パラメータの解説中の下線部はオプションを指定しなかった時のデフォルト値をそれぞれ

示します。

- TCP/IP で接続する場合

“Conn=TCP:<接続先 IP>[:接続先ポート]”

<接続先 IP> : 接続先 IP アドレスを指定します。この項目は必ず指定してください。

<接続先ポート> : 接続先ポート番号を指定します。1024

3.2.2. CaoController クラス

3.2.2.1. VariableNames プロパティ

接続可能な変数名リストを取得します。本プロパティで取得した変数名は、後述する AddVariable メソッドの第一引数に使用することができます。

使用例

```
' ファイル名リスト取得
Dim variables as Variant
variables = controller.VariableNames
```

3.2.2.2. AddVariable メソッド

CaoController に変数オブジェクトを追加します。変数名には 3.3.13.3.1 章に示す文字列を使用できます。

以下に、AddVariable の仕様を示します。

書式

CaoVariable AddVariable

```
(
    "<変数名>",           // 変数名
    "<オプション>"       // オプション文字列(省略可能)
)
```

3.2.2.3. OnMessage イベント

コントローラのエラー通知や状態の変化を OnMessage イベントとして受け取ることが可能です。受け取れるイベントについては 3.4 を参照してください。

3.2.3. CaoVariable クラス

3.2.3.1. Value プロパティ

接続したワールドチェッカーからデータを取得/設定します。変数名によって動作が異なります。詳細は、3.3.1 CaoController クラス変数を参照してください。

3.3. 変数一覧

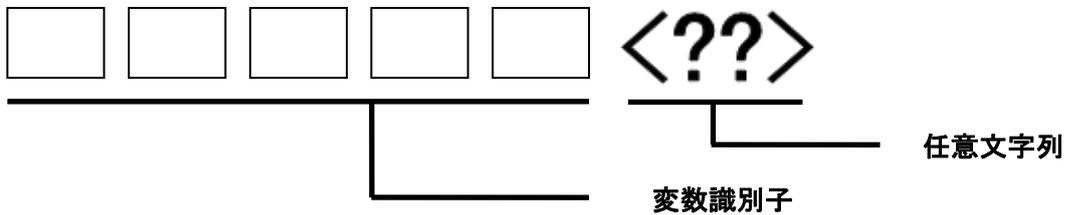
各クラスで使用可能な変数一覧を定義します。なお変数は、CaoVariable クラスのオブジェクトを指します。

3.3.1. CaoController クラス変数

複数変数を登録（オプションのみ変更したい場合等に有用）するために任意の文字列を付与することが可能です。

変数名に任意の文字列を付与するための書式を以下に示します。

複数変数共通指定書式



変数名	説明	Value		参照
		get	put	
@MAKER_NAME	メーカー名を取得します。 単方向/双方向どちらのコントローラでも使用できます。	○	-	P. 14
@VERSION	DLL バージョンを取得します。 単方向/双方向どちらのコントローラでも使用できます。	○	-	P. 15
@MEASUREMENT	測定を取得します。 双方向のコントローラのみ使用できます。	○	-	P. 15
@WAVEFORM	波形を取得します。 双方向のコントローラのみ使用できます。	○	-	P. 16
@CURRENT_ALLCYCLE	電流オールサイクルを取得します。 双方向のコントローラのみ使用できます。	○	-	P. 19
@MEASUREMENT_HISTORY	測定履歴を取得します。 双方向のコントローラのみ使用できます。	○	-	P. 21

変数名	説明	Value		参照
		get	put	
@ALARM_HISTORY	異常履歴を取得します。 双方向のコントローラのみ使用できます。	○	-	P. 22
@DISPLAY_SETTING	表示設定を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 22
@SETTING_CONDITION	条件[設定]を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 24
@EXTERNAL_IO	外部入出力を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 25
@PRINTING	印刷を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 27
@COMMUNICATION	通信を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 28
@USB	USB を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 31
@MEMORY	内部メモリを取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 32
BASIC_CONDITION<??>	条件[基本]を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 33
EXT_CONDITION<??>	条件[拡張]を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 36
SEAM_CONDITION<??>	条件[シーム]を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 42
RANGE_SETTING<??>	上下限設定を取得/設定します。 双方向のコントローラのみ使用できます。	○	○	P. 45
ENVELOPE<??>	エンベロップを取得/設定します。 双方向のコントローラのみ使用できます。	○	-	P. 47

3.3.1.1. @MAKER_NAME

メーカー名の取得をします。

データ型

型説明	
VT_BSTR	メーカー名を取得します。

使用例

```

' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("@MAKER_NAME")
' 値取得
Dim strVal As String
strVal = var.value

```

3.3.1.2. @VERSION

プロバイダバージョンを文字列で取得をします。

データ型

型説明	
VT_BSTR	DLL のバージョンを取得します。 *. *.*

使用例

```

' 変数追加
Dim var As CaoVariable
Set var = ctrl.AddVariable("@VERSION")
' 値取得
Dim value As String
value = var.value

```

3.3.1.3. @MEASUREMENT

測定【項目番号:1】を取得します。測定データが存在しない場合は VT_EMPTY となります。測定値の型は測定項目番号により異なります。測定項目番号とデータ型の対応表および判定文字列表は、「付録 B 測定項目」を参照してください。

データ型

型説明	
VT_ARRAY VT_VARIANT	
0VT_UI1	装置 No. 1 - 31
1VT_UI1	条件番号 1 - 127
2VT_DATE	測定日時
4i - 1VT_UI1	測定項目コード i ($1 \leq i \leq N$)
4i + 0VT_BSTR	判定 i ($1 \leq i \leq N$)

型説明		
4i + 2	VT_VARIANT	測定値 i ($1 \leq i \leq N$) データが存在しない場合は VT_EMPTY となります。
4i + 3	VT_BSTR	単位 i ($1 \leq i \leq N$)

N : 表示設定画面で設定項目 6~10 項目が設定されている場合は 10, 設定されていない場合は 5

使用例

```

' 変数追加
Dim measurement As CaoVariable
Set measurement = ctrl.AddVariable("@MEASUREMENT")
' 値を取得
Dim measValues As Variant
measValues = measurement.value
' 装置 No
Dim machineNo As Long
machineNo = measValues(0)
' 条件番号
Dim condition As Long
condition = measValues(1)
' 測定日時
Dim measDate As Date
measDate = measValues(2)
' 測定項目 1 ~ 最大測定項目までのループ
For i = 3 To UBound(measValues) Step 4
' 測定項目コードの取得
Dim term As Integer
term = measValues(i)
' 判定コードの取得
Dim result As String
result = measValues(i + 1)
' 測定値の取得
Dim value As Double
value = measValues(i + 2)
' 単位の取得
Dim unit As String
unit = measValues(i + 3)
Next i

```

3.3.1.4. @WAVEFORM

波形【項目番号:2】を取得します。波形データが存在しない場合は VT_EMPTY となります。波形項目番号一覧は「取扱説明書」を参照して下さい。

データ型

型説明
VT_ARRAY VT_VARIANT

型説明		
0	VT_UI1	装置 No. 1 - 31
1	VT_I4	サンプリング数
2	VT_UI1	条件番号 1 - 127
3	VT_UI1	波形項目コード 1 (255 の場合は項目なしを意味します。)
4	VT_UI1	波形項目コード 2 (255 の場合は項目なしを意味します。)
5	VT_UI1	波形項目コード 3 (255 の場合は項目なしを意味します。)
6	VT_UI1	波形項目コード 4 (255 の場合は項目なしを意味します。)
7	VT_VARIANT VT_ARRAY	
	<i>i</i>	VT_VARIANT VT_ARRAY
	0	VT_R4 サンプリング時間
	1	VT_BSTR サンプリング時間 (単位)
	2	VT_R4 測定値 1 項目なしや非表示の場合は VT_EMPTY
	3	VT_BSTR 単位 1 項目なしや非表示の場合は VT_EMPTY
	4	VT_R4 測定値 2 項目なしや非表示の場合は VT_EMPTY
	5	VT_BSTR 単位 2 項目なしや非表示の場合は VT_EMPTY
	6	VT_R4 測定値 3 項目なしや非表示の場合は VT_EMPTY
	7	VT_BSTR 単位 3 項目なしや非表示の場合は VT_EMPTY
	8	VT_R4 測定値 4 項目なしや非表示の場合は VT_EMPTY
	9	VT_BSTR 単位 4 項目なしや非表示の場合は VT_EMPTY

 $0 \leq i < \text{サンプリング数}$

使用例

```
Dim waveform As CaoVariable
Set waveform = controller.AddVariable("@WAVEFORM")
```

```
Dim readValues As Variant
readValues = waveform.value

' 装置 No
Dim machineNo As Long
machineNo = readValues(0)

' サンプル数
Dim numOfSumple As Long
numOfSumple = readValues(1)

' 条件番号
Dim condition As Long
condition = readValues(2)

' 波形項目番号取得
Dim terms(3) As Long
For i = 0 To 3
terms(i) = readValues(3 + i)
Next i

' 波形データを取得
Dim waveformCollection As Variant
waveformCollection = readValues(9)
For j = 0 To (numOfSumple - 1)
' サンプル時間の値
Dim val1 As Single
val1 = waveformCollection(j) (0)
' サンプル時間の単位
Dim unit1 As String
unit1 = waveformCollection(j) (1)

' 項目 1 の値
Dim val1 As Single
val1 = waveformCollection(j) (2)
' 項目 1 の単位
Dim unit1 As String
unit1 = waveformCollection(j) (3)

' 項目 2 の値
Dim val2 As Single
val2 = waveformCollection(j) (4)
' 項目 2 の単位
Dim unit2 As String
unit2 = waveformCollection(j) (5)

' 項目 3 の値
Dim val3 As Single
```

```

val3 = waveformCollection(j) (6)
' 項目 3 の単位
Dim unit3 As String
unit3 = waveformCollection(j) (7)

' 項目 4 の値
Dim val4 As Single
val4 = waveformCollection(j) (8)
' 項目 4 の単位
Dim unit4 As String
unit4 = waveformCollection(j) (9)
Next j

```

3.3.1.5. @CURRENT_ALLCYCLE

電流オールサイクル【項目番号:3】を取得します。電流オールサイクルデータが存在しない場合は VT_EMPTY となります。

データ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_UI1	装置 No. 1 - 31
1	VT_I4	サンプリング数
2	VT_UI1	条件番号 1 - 127
3	VT_VARIANT VT_ARRAY	
	<i>i</i>	VT_VARIANT VT_ARRAY
	0	VT_R4 サンプリング時間
	1	VT_BSTR サンプリング時間(単位)
	2	VT_UI1 測定範囲 1: 測定範囲内 0: 測定範囲外
	3	VT_R4 電流
	4	VT_BSTR 電流 単位(kA)
	5	VT_R4 電圧
	6	VT_BSTR 電圧 単位(V)
	7	VT_UI1 通電角 0 - 180 ※この項目は存在しない場合があります。

型説明			
	8	VT_BSTR	通電角 単位(deg) ※この項目は存在しない場合があります。

$0 \leq i < \text{サンプリング数}$

使用例

```
Dim currentAllcycle As CaoVariable
Set currentAllcycle = controller.AddVariable("@CURRENT_ALLCYCLE")
```

```
Dim readValues As Variant
readValues = currentAllcycle.value
```

' 装置 No

```
Dim machineNo As Long
machineNo = readValues(0)
```

' サンプリング数

```
Dim numOfSumple As Long
numOfSumple = readValues(1)
```

' 条件番号

```
Dim condition As Long
condition = readValues(2)
```

' 電流オールサイクルを取得

```
Dim currentAllcycleCollection As Variant
currentAllcycleCollection = readValues(3)
```

```
For j = 0 To (numOfSumple - 1)
```

' サンプリング時間の値

```
Dim val1 As Single
val1 = currentAllcycleCollection(j)(0)
```

' サンプリング時間の単位

```
Dim unit1 As String
unit1 = currentAllcycleCollection(j)(1)
```

' 測定範囲

```
Dim range As Byte
range = currentAllcycleCollection(j)(2)
```

```
if range = 1 Then
```

' 電流の値

```
Dim current As Single
current = currentAllcycleCollection(j)(3)
```

' 電流の単位

```
Dim currentUnit As String
currentUnit = currentAllcycleCollection(j)(4)
```

```

' 電圧の値
Dim voltage As Single
voltage = currentAllcycleCollection (j) (5)
' 電圧の単位
Dim voltageUnit As String
voltageUnit = currentAllcycleCollection (j) (6)

If UBound(currentAllcycleCollection(j)) >= 7 Then
' 通電角の値
Dim degree As Single
degree = currentAllcycleCollection (j) (7)
' 通電角の単位
Dim degreeUnit As String
degreeUnit = currentAllcycleCollection (j) (8)
End If
End if
Next j

```

3.3.1.6. @MEASUREMENT_HISTORY

測定履歴【項目番号:6】を取得します。

データ型

型説明	
VT_ARRAY VT_VARIANT	
0	VT_I4
1	VT_ARRAY VT_VARIANT
	<i>i</i> @MEASUREMENT を参照してください。

使用例

```

Dim history As CaoVariable
Set history = controller.AddVariable("@MEASUREMENT_HISTORY")

```

```

Dim readValues As Variant
readValues = history.value

```

' 履歴数

```

Dim numofSuple As Long
numofSuple = readValues(0)

```

' 測定履歴を取得

```

Dim measurementCollection As Variant
measurementCollection = readValues(1)
For j = 0 To (numofSuple - 1)

```

' 装置 No

```

Dim machineNo As Long
machineNo = measurementCollection(j) (0)

' 条件番号
Dim condition As Long
condition = measurementCollection(j) (1)

' 測定日時
Dim measDate As Date
measDate = measurementCollection(j) (2)

For i = 3 To UBound(measurementCollection(j)) Step 4
' 測定項目コード
Dim term As Long
term = measurementCollection(j) (i)

' 判定コード
Dim result As String
result = measurementCollection(j) (i + 1)

' 測定値
Dim value As Double
value = measurementCollection(j) (i + 2)

' 単位
Dim unit As String
unit = measurementCollection(j) (i + 3)
Next i
Next j

```

3.3.1.7. @ALARM_HISTORY

異常履歴【項目番号:7】を取得します。測定履歴と異なり、何らかの異常があった測定のみを取得します。

データ型

@MEASUREMENT_HISTORY を参照してください。

使用例

@MEASUREMENT_HISTORY を参照してください。

3.3.1.8. @DISPLAY_SETTING

表示設定【項目番号:11】を取得/設定します。

データ型

型説明

VT_I4 | VT_ARRAY

型説明	
0	測定値 表示 0: 5 項目 1: 10 項目
1	測定項目コード 1
2	測定項目コード 2
3	測定項目コード 3
4	測定項目コード 4
5	測定項目コード 5
6	測定項目コード 6
7	測定項目コード 7
8	測定項目コード 8
9	測定項目コード 9
10	測定項目コード 10
11	波形項目コード 1
12	波形項目コード 2
13	波形項目コード 3
14	波形項目コード 4
15	波形表示 1 0: OFF 1: ON
16	波形表示 2 0: OFF 1: ON
17	波形表示 3 0: OFF 1: ON
18	波形表示 4 0: OFF 1: ON

使用例

表示設定取得

```
Dim displaySetting As CaoVariable
```

```
Set displaySetting = controller.AddVariable("@DISPLAY_SETTING")
```

```
Dim values() As Long
values = displaySetting.value
```

```
' 測定 1 を変位量パルス後に設定
```

```
values(1) = 15
```

```
' 波形 4 を表示に設定
```

```
values(18) = 1
```

```
' 反映
```

```
displaySetting = values
```

3.3.1.9. @SETTING_CONDITION

条件[設定] 【項目番号:14】 を取得/設定します.

データ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_I4	モード 0: ノーマル 1: シーム 2: ノーマルトレース 3: シングルトレース
1	VT_I4	言語 0: 英語 1: ドイツ語 2: フランス語 3: スペイン語 4: 日本語 5: 韓国語 6: 中国語
2	VT_I4	溶接カウンタ設定 0 - 999999
4	VT_I4	良品カウンタ設定 0 - 999999
5	VT_I4	明るさ 0 - 10
6	VT_I4	明るさ 0: OFF 1: AUTO

型説明		
7	VT_R4	コイル変換係数 100 - 250
8	VT_BSTR	コイル変換係数単位 (mV/kA)
9	VT_I4	サンプリング間隔 0: 100us 1: 200us 2: 500us

使用例

'条件[設定]を取得

```
Dim settingCondition As CaoVariable
```

```
Set settingCondition = controller.AddVariable("@SETTING_CONDITION")
```

```
Dim values As Variant
```

```
values = settingCondition.value
```

'言語を英語に設定

```
values(1) = 0
```

'反映

```
settingCondition = values
```

3.3.1.10. @EXTERNAL_IO

外部入出力【項目番号:18】を取得/設定します。コードについては「取扱説明書」を参照してください。

データ型

型説明	
VT_I4 VT_ARRAY	
0	入力1(外部入力1-6コード表)
1	入力2(外部入力1-6コード表)
2	入力3(外部入力1-6コード表)
3	入力4(外部入力1-6コード表)
4	入力5(外部入力1-6コード表)
5	入力6(外部入力1-6コード表)
6	入力7(外部入力7-8コード表)
7	入力8(外部入力7-8コード表)
8	出力1-1(外部出力1-12コード表)

型説明	
9	出力 1-2 (外部出力 1-12 コード表)
10	出力 1-3 (外部出力 1-12 コード表)
11	出力 2-1 (外部出力 1-12 コード表)
12	出力 2-2 (外部出力 1-12 コード表)
13	出力 2-3 (外部出力 1-12 コード表)
14	出力 3-1 (外部出力 1-12 コード表)
15	出力 3-2 (外部出力 1-12 コード表)
16	出力 3-3 (外部出力 1-12 コード表)
17	出力 4-1 (外部出力 1-12 コード表)
18	出力 4-2 (外部出力 1-12 コード表)
19	出力 4-3 (外部出力 1-12 コード表)
20	出力 5-1 (外部出力 1-12 コード表)
21	出力 5-2 (外部出力 1-12 コード表)
22	出力 5-3 (外部出力 1-12 コード表)
23	出力 6-1 (外部出力 1-12 コード表)
24	出力 6-2 (外部出力 1-12 コード表)
25	出力 6-3 (外部出力 1-12 コード表)
26	出力 7-1 (外部出力 1-12 コード表)
27	出力 7-2 (外部出力 1-12 コード表)
28	出力 7-3 (外部出力 1-12 コード表)
29	出力 8-1 (外部出力 1-12 コード表)
30	出力 8-2 (外部出力 1-12 コード表)
31	出力 8-3 (外部出力 1-12 コード表)
32	出力 9-1 (外部出力 1-12 コード表)
33	出力 9-2 (外部出力 1-12 コード表)
34	出力 9-3 (外部出力 1-12 コード表)
35	出力 10-1 (外部出力 1-12 コード表)
36	出力 10-2 (外部出力 1-12 コード表)
37	出力 10-3 (外部出力 1-12 コード表)
38	出力 11-1 (外部出力 1-12 コード表)
39	出力 11-2 (外部出力 1-12 コード表)
40	出力 11-3 (外部出力 1-12 コード表)
41	出力 12-1 (外部出力 1-12 コード表)
42	出力 12-2 (外部出力 1-12 コード表)

型説明	
43	出力 12-3 (外部出力 1-12 コード表)
44	パリティ 0: OFF 1: ON
45	入力安定時間 0: 1ms 1: 10ms
46	異常出力設定 0: Normal close 1: Normal open
47	出力時間 0: 10ms 1: 100ms 2: HOLD
48	レベル出力 0: Level 1: Pulse

使用例

```
' 外部入出力を取得
```

```
Dim externalIO As CaoVariable
```

```
Set externalIO = controller.AddVariable("@EXTERNAL_IO")
```

```
Dim values() As Long
```

```
values = externalIO.value
```

```
' 入力 1 にパリティを設定
```

```
values(0) = 0
```

```
' 入力 7 に変位トリガを設定
```

```
values(6) = 0
```

```
' 反映
```

```
externalIO = values
```

3.3.1.11. @PRINTING

印刷【項目番号:21】を取得/設定します。

データ型

型説明

型説明	
VT_I4 VT_ARRAY	
0	項目 0: OFF 1: 測定値 2: 波形 3: 電流オールサイクル 4: 加圧オールサイクル 5: 測定値履歴 6: 異常履歴 7: 条件 8: 画面
1	インターバル 1 - 1000
2	判定外動作 0: OFF 1: ON

使用例

```
'印刷を取得
Dim printing As CaoVariable
Set printing = controller.AddVariable("@PRINTING")

Dim values() As Long
values = printing.value

'印刷項目を測定に設定
values(0) = 1

'反映
printing = values
```

3.3.1.12. @COMMUNICATION

通信【項目番号:22】を取得/設定します。

データ型

型説明
VT_I4 VT_ARRAY

型説明	
0	項目 0: OFF 1: 測定値 2: 波形 3: 電流オールサイクル 4: 加圧オールサイクル 5: 測定値履歴 6: 異常履歴 7: 条件
1	インターバル 1 - 1000
2	判定外動作 0: OFF 1: ON
3	波形間引き 0: 20us 1: 50us 2: 100us 3: 200us 4: 500us 5: 1000us
4	単位 0: OFF 1: ON
5	小数点(※データの変更は行えません。) 0: ピリオド(.) 1: カンマ(,)
6	方式(※データの変更は行えません。) 0: OFF 1: RS232C 2: RS485 3: ETHERNET
7	方式(※データの変更は行えません。) 0: 方方向 1: 双方向

型説明	
8	装置番号(※データの変更は行えません。) 1 - 31
9	速度(※データの変更は行えません。) 0: 9600 1:19200 2:38400
10	IP アドレス 1(※データの変更は行えません。) 0 - 255
11	IP アドレス 2(※データの変更は行えません。) 0 - 255
12	IP アドレス 3(※データの変更は行えません。) 0 - 255
13	IP アドレス 4(※データの変更は行えません。) 0 - 255
14	サブネットマスク 1(※データの変更は行えません。) 0 - 255
15	サブネットマスク 2(※データの変更は行えません。) 0 - 255
16	サブネットマスク 3(※データの変更は行えません。) 0 - 255
17	サブネットマスク 4(※データの変更は行えません。) 0 - 255
18	デフォルトゲートウェイ 1(※データの変更は行えません。) 0 - 255
19	デフォルトゲートウェイ 2(※データの変更は行えません。) 0 - 255
20	デフォルトゲートウェイ 3(※データの変更は行えません。) 0 - 255
21	デフォルトゲートウェイ 4(※データの変更は行えません。) 0 - 255
22	ポート番号 1024 - 5000

使用例

’通信を取得

```
Dim comm As CaoVariable
Set comm = controller.AddVariable("@COMMUNICATION")
```

```
Dim values() As Long
values = comm.value
```

```
' 単位を ON
```

```
values(4) = 1
```

```
' 反映
```

```
comm = values
```

3.3.1.13. @USB

USB【項目番号:23】を取得/設定します。

データ型

型説明	
VT_I4 VT_ARRAY	
0	項目 0: OFF 1: 測定値 2: 波形 3: 電流オールサイクル 4: 加圧オールサイクル 5: 測定値履歴 6: 異常履歴 7: 条件 8: 画面 9: エンベロープ
1	インターバル 1 - 1000
2	判定外動作 0: OFF 1: ON
3	波形間引き 0: 20us 1: 50us 2: 100us 3: 200us 4: 500us 5: 1000us

型説明	
4	単位 0: OFF 1: ON
5	小数点 0: ピリオド(.) 1: カンマ(,)

使用例

```

'USB を取得
Dim usb As CaoVariable
Set usb = controller.AddVariable("@USB")

Dim values() As Long
values = usb.value

' 単位を ON
values(4) = 1
' 反映
usb = values

```

3.3.1.14. @MEMORY

内部メモリ【項目番号:24】を取得/設定します.

データ型

型説明	
VT_I4 VT_ARRAY	
0	項目 0: OFF 1: 波形 2: 電流オールサイクル 3: 加圧オールサイクル
1	インターバル 1 - 1000
2	判定外動作 0: OFF 1: ON

型説明	
3	波形間引き 0: 20us 1: 50us 2: 100us 3: 200us 4: 500us 5:1000us

使用例

```

' 内部メモリを取得
Dim memory As CaoVariable
Set memory = controller.AddVariable("@MEMORY")

Dim values() As Long
values = memory.value

' 項目を波形に設定
values(0) = 1
' 反映
memory = values
    
```

3.3.1.15. BASIC_CONDITION<??>

条件[基本]【項目番号:12】を取得/設定します。BASIC_CONDITION の後に任意の文字列を入力して変数を作成してください。

オプション

以下にオプション文字列に指定するオプションを示します。オプション文字列は下記に示す各オプションをカンマ(,)でつなげた文字列となります。

オプション	必須	説明	値範囲	デフォルト値
Condition	○	アクセス対象の条件番号を指定します。 0 を指定した場合とそれ以外でデータ型が変化します。	0: 共通項目 1 - 127: 条件指定	0

Condition = 0 のときのデータ型

型説明	
VT_I4 VT_ARRAY	
0	電流トリガ感度 1 - 99

型説明	
1	トロイダルコイル 0: 1倍 1: 10倍
2	演算 0: オリジナル 1: ISO1767657
3	電流センサー 0: トロイダルコイル 1: ジェント抵抗
4	ジェント抵抗 0: 50mV/0.5kA 1: 50mV/1kA 2: 100mV/0.5kA 3: 100mV/1kA

Condition ≠ 0 のときのデータ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_BSTR	条件名称(A-Z, 0 - 9)
1	VT_I4	トリガ 0: 電流 1: オート 2: 加圧力 3: 外部 4: 連続 5: 変位量(外部) 6: 加圧力(外部) 7: 外部(外部)
2	VT_I4	時間 0: CYC-AC 1: ms-DC 2: CYC***Hz-AC 3: CYC-D 4: ms-AC 5: SHORT ms-DC 6: LOG CYC-AC
3	VT_I4	周波数 30: M050 31: M053 32: M056 33: M059 34: M063 35: M067 36: M071 37: M077 38: M083 39: M091 40: M100 41: M111 42: M125 43: M143 44: M167 45: M200 46: M250 47: M294 48: M417 49: M500 50 - 250: 50Hz - 250Hz

型説明		
4	VT_I4	電流レンジ トロイダルコイル 1 倍時: 0:2.000kA 1:6.00kA 2:20.00kA 3: 60.0kA 4:200.0kA トロイダルコイル 10 倍時: 0:0.2.00kA 1:0.600kA 2:2.000kA 3: 6.00kA 4:20.00kA
5	VT_I4	電圧レンジ 0: 6.00V 1: 20.0V
6	VT_R4	開始時間
7	VT_BSTR	開始時間単位 (CYC / ms)
8	VT_R4	終了時間
9	VT_BSTR	終了時間単位 (CYC / ms)
10	VT_I4	インパルス 0: 指定パルス 1: 全パルス設定あり 2: 全パルス設定なし 3: クールなし
11	VT_I4	インパルス番号 0 - 20
12	VT_R4	パルス 2 開始電流値
13	VT_BSTR	パルス 2 開始電流値単位 (kA)
14	VT_R4	クール時間
15	VT_BSTR	クール時間単位 (CYC / ms)
16	VT_R4	フォールレベル 10 - 90
17	VT_BSTR	フォールレベル単位 (%)
18	VT_R4	強制測定時間
19	VT_BSTR	強制測定時間単位 (CYC / ms)
20	VT_R4	測定休止時間
21	VT_BSTR	測定休止時間単位 (s)
22	VT_R4	終了レベル 1.5 - 15.0
23	VT_BSTR	終了レベル単位 (%)

使用例

```

'条件[基本]を取得
Dim var As CaoVariable
Set var = controller.AddVariable("BASIC_CONDITION_common")

Dim values() As Long
values = var.value

'トリガ感度を 30 に設定
values(0) = 30
'反映
var = values

```

3.3.1.16. EXT_CONDITION<??>

条件[拡張]【項目番号:13】を取得/設定します。EXT_CONDITION の後に任意の文字列を入力して変数を作成してください。

オプション

以下にオプション文字列に指定するオプションを示します。オプション文字列は下記に示す各オプションをカンマ(,)でつなげた文字列となります。

オプション	必須	説明	値範囲	デフォルト値
Condition	○	アクセス対象の条件番号を指定します。 0 を指定した場合とそれ以外でデータ型が変化します。	0: 共通項目 1 - 127: 条件指定	0

Condition = 0 のときのデータ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_I4	変位量 溶接前測定 0: 電流開始 1: 外部入力
1	VT_I4	変位量 パルス後測定 0: 電流開始 1: デイレイ時間
2	VT_I4	変位量 溶接後測定 0: デイレイ時間 1: 外部入力

型説明		
3	VT_I4	変位量 パルス 2 リセット 0: OFF 1: ON
4	VT_I4	変位量 測定方式 0: 相対値 1: 絶対値 1 2: 絶対値 2
5	VT_I4	変位量 センサー 0: ミットヨ 1: 小野測器 2: キーエンス 3: ハイデンハイン
6	VT_R4	変位量 センサーステップ 0.1 um - 10.0 um
7	VT_BSTR	変位量 センサーステップ単位 (um)
8	VT_I4	変位量 極性 0: 正方向 1: 逆方向
9	VT_I4	変位量 単位 0: mm 1: inch
10	VT_I4	変位量 出力レンジ 0: 2.047 1: 9.191 2: 32.767
11	VT_I4	加圧力 センサー 0: MA-520 1: MA-521 2: MA-522 3: MA-770 4: MA-771 5: 定格設定
12	VT_I4	加圧力 スパン 500 - 1500

型説明		
13	VT_R4	加圧力 定格出力 0.750 mV/V - 2.000 mV/V
14	VT_BSTR	加圧力 定格出力単位 (mV/V)
15	VT_R4	加圧力 定格
16	VT_BSTR	加圧力 定格単位
17	VT_I4	加圧力 小数点 0: **** 1: ***. * 2: **. **
18	VT_I4	加圧力 単位 0: N 1: kgf 2: lbf
19	VT_R4	加圧力 トリガ感度 0.2% - 99.9%
20	VT_BSTR	加圧力 トリガ感度単位 (%)
21	VT_I4	加圧力 溶接前測定 0: 電流開始 1: 外部入力
22	VT_I4	外部 入力 0: 電圧 1: 電流
23	VT_R4	外部 定格
24	VT_BSTR	外部 定格単位
25	VT_I4	外部 小数点 0: **** 1: ***. * 2: **. ** 3: *. ***

型説明		
26	VT_I4	外部 単位 0: なし 1: V 2: N 3: kgf 4: lbf 5: degC 6: degF 7: Mpa 8: bar 9: psi
27	VT_R4	外部 トリガ感度 2.0% - 99.9%
28	VT_BSTR	外部 トリガ感度 単位
29	VT_I4	溶接前測定 0: 電流開始 1: 外部入力
30	VT_R4	加圧力 オフセット
31	VT_BSTR	加圧力 オフセット単位

Condition ≠ 0 のときのデータ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_R4	変位量 レベル出力 1
1	VT_BSTR	変位量 レベル出力 1 単位 (mm / inch)
2	VT_R4	変位量 レベル出力 2
3	VT_BSTR	変位量 レベル出力 2 単位 (mm / inch)
4	VT_R4	変位量 レベル出力 3
5	VT_BSTR	変位量 レベル出力 3 単位 (mm / inch)
6	VT_R4	変位量 最終ディレイ時間 0ms - 10000ms
7	VT_BSTR	変位量 最終ディレイ時間 単位 (ms)

型説明		
8	VT_R4	変位量 パルスディレイ時間 1 0ms - 1000ms
9	VT_BSTR	変位量 パルスディレイ時間 1 単位 (ms)
10	VT_R4	変位量 パルスディレイ時間 2 0ms - 1000ms
11	VT_BSTR	変位量 パルスディレイ時間 2 単位 (ms)
12	VT_R4	変位量 リセットディレイ時間 0ms - 1000ms
13	VT_BSTR	変位量 リセットディレイ時間 単位 (ms)
14	VT_R4	加圧力 レベル出力 1
15	VT_BSTR	加圧力 レベル出力 1 単位 (N / kgf / lbf)
16	VT_R4	加圧力 レベル出力 2
17	VT_BSTR	加圧力 レベル出力 2 単位 (N / kgf / lbf)
18	VT_R4	加圧力 レベル出力 3
19	VT_BSTR	加圧力 レベル出力 3 単位 (N / kgf / lbf)
20	VT_R4	加圧力 デイレイ時間 0ms - 1000ms
21	VT_BSTR	加圧力 デイレイ時間 単位 (ms)
22	VT_R4	加圧力 開始時間 1 0ms - 10000ms (10ms ステップ)
23	VT_BSTR	加圧力 開始時間 1 単位 (ms)
24	VT_R4	加圧力 終了時間 1 0ms - 10000ms (10ms ステップ)
25	VT_BSTR	加圧力 終了時間 1 単位 (ms)
26	VT_R4	加圧力 開始時間 2 0ms - 10000ms (10ms ステップ)
27	VT_BSTR	加圧力 開始時間 2 単位 (ms)
28	VT_R4	加圧力 終了時間 2 0ms - 10000ms (10ms ステップ)
29	VT_BSTR	加圧力 終了時間 2 単位 (ms)
30	VT_R4	加圧力 ライズレベル 10% - 90%
31	VT_BSTR	加圧力 ライズレベル 単位 (%)

型説明		
32	VT_R4	加圧力 フォールレベル 10% - 90%
33	VT_BSTR	加圧力 フォールレベル 単位 (%)
34	VT_R4	外部 レベル出力 1
35	VT_BSTR	外部 レベル出力 1 単位 (V / N /kgf / lbf / degC / degF / Mpa / bar / psi)
36	VT_R4	外部 レベル出力 2
37	VT_BSTR	外部 レベル出力 2 単位 (V / N /kgf / lbf / degC / degF / Mpa / bar / psi)
38	VT_R4	外部 レベル出力 3
39	VT_BSTR	外部 レベル出力 3 単位 (V / N /kgf / lbf / degC / degF / Mpa / bar / psi)
40	VT_R4	外部 デイレイ時間 0ms - 1000ms
41	VT_BSTR	外部 デイレイ時間 単位 (ms)
42	VT_R4	外部 開始時間 1 0ms - 10000ms (10ms ステップ)
43	VT_BSTR	外部 開始時間 1 単位 (ms)
44	VT_R4	外部 終了時間 1 0ms - 10000ms (10ms ステップ)
45	VT_BSTR	外部 終了時間 1 単位 (ms)
46	VT_R4	外部 開始時間 2 0ms - 10000ms (10ms ステップ)
47	VT_BSTR	外部 開始時間 2 単位 (ms)
48	VT_R4	外部 終了時間 2 0ms - 10000ms (10ms ステップ)
49	VT_BSTR	外部 終了時間 2 単位 (ms)

使用例

'条件[拡張]を取得

```
Dim var As CaoVariable
```

```
Set var = controller.AddVariable("EXT_CONDITION_common")
```

```
Dim values() As Long
```

```
values = var.value
```

'変位量 溶接前測定を外部入力に設定

```
values(0) = 1
```

'変位量 パルス後測定を電流開始に設定

```
values(1) = 0
```

'変位量 溶接後測定をディレイ時間に設定

```
values(2) = 0
```

'反映

```
var = values
```

3.3.1.17. SEAM_CONDITION<??>

条件[シーム]【項目番号:15】を取得/設定します。SEAM_CONDITION の後に任意の文字列を入力して変数を作成してください。

オプション

以下にオプション文字列に指定するオプションを示します。オプション文字列は下記に示す各オプションをカンマ(,)でつなげた文字列となります。

オプション	必須	説明	値範囲	デフォルト値
Condition	○	アクセス対象の条件番号を指定します。 0 を指定した場合とそれ以外でデータ型が変化します。	0: 共通項目 1 - 127: 条件指定	0

Condition = 0 のときのデータ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_I4	トリガ 0: 電流 1: 電圧 「トリガ」の設定により「時間」に選択できる条件が変わります。「トリガ」が「0:電流」の時、「時間」は「0:CYC-AC」,「1:ms-AC」が選択できます。「トリガ」が「1:電圧」のとき、「時間」は「2:ms-DC」,「3:CYC-DC」,「4:SHORT ms-DC」が選択できます。
1	VT_I4	時間 0:CYC-AC 1:ms-AC 2:ms-DC 3:CYC-DC 4:SHORT ms-DC
2	VT_I4	演算 0: ピーク値 1: 実効値 2: 相加平均実効値

型説明		
3	VT_I4	電流レンジ トロイダルコイル 1 倍時: 0:2.000kA 1:6.00kA 2:20.00kA 3: 60.0kA 4:200.0kA トロイダルコイル 10 倍時: 0:0.2.00kA 1:0.600kA 2:2.000kA 3: 6.00kA 4:20.00kA
4	VT_I4	電圧レンジ 0: 6.00V 1: 20.0V
5	VT_I4	電流トリガー感度 1 - 99
6	VT_I4	電圧トリガー感度 1 - 99
7	VT_I4	トロイダルコイル 0: 1 倍 1: 10 倍
8	VT_R4	終了レベル 1.5% - 15.0%
9	VT_BSTR	終了レベル単位 (%)

Condition ≠ 0 のときのデータ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_BSTR	条件名称 (A-Z, 0 - 9)
1	VT_R4	測定開始
2	VT_BSTR	測定開始単位 (ms / CYC)
3	VT_R4	測定範囲
4	VT_BSTR	測定範囲単位 (ms / CYC)
5	VT_R4	測定間隔
6	VT_BSTR	測定間隔単位 (ms / CYC)
7	VT_R4	開始時間 1
8	VT_BSTR	開始時間 1 単位 (ms / CYC)
9	VT_R4	終了時間 1
10	VT_BSTR	終了時間 1 単位 (ms / CYC)
11	VT_R4	開始時間 2

型説明		
12	VT_BSTR	開始時間 2 単位 (ms / CYC)
13	VT_R4	終了時間 2
14	VT_BSTR	終了時間 2 単位 (ms / CYC)
15	VT_R4	開始時間 3
16	VT_BSTR	開始時間 3 単位 (ms / CYC)
17	VT_R4	終了時間 3
18	VT_BSTR	終了時間 3 単位 (ms / CYC)
19	VT_R4	電流上限 1
20	VT_BSTR	電流上限 1 単位 (kA)
21	VT_R4	電流下限 1
22	VT_BSTR	電流下限 1 単位 (kA)
23	VT_R4	電流上限 2
24	VT_BSTR	電流上限 2 単位 (kA)
25	VT_R4	電流下限 2
26	VT_BSTR	電流下限 2 単位 (kA)
27	VT_R4	電流上限 3
28	VT_BSTR	電流上限 3 単位 (kA)
29	VT_R4	電流下限 3
30	VT_BSTR	電流下限 3 単位 (kA)
31	VT_R4	電圧上限 1
32	VT_BSTR	電圧上限 1 単位 (V)
33	VT_R4	電圧下限 1
34	VT_BSTR	電圧下限 1 単位 (V)
35	VT_R4	電圧上限 2
36	VT_BSTR	電圧上限 2 単位 (V)
37	VT_R4	電圧下限 2
38	VT_BSTR	電圧下限 2 単位 (V)
39	VT_R4	電圧上限 3
40	VT_BSTR	電圧上限 3 単位 (V)
41	VT_R4	電圧下限 3
42	VT_BSTR	電圧下限 3 単位 (V)

使用例

'条件[拡張]を取得

```
Dim var As CaoVariable
Set var = controller.AddVariable("SEAM_CONDITION_common")
```

```
Dim values() As Long
values = var.value
```

```
' トリガを電圧に設定
```

```
values(0) = 1
```

```
' 反映
```

```
var = values
```

3.3.1.18. RANGE_SETTING<??>

上下限設定【項目番号:16】を取得/設定します。RANGE_SETTING の後に任意の文字列を入力して変数を作成してください。上下限コードは「取扱説明書」をご覧ください。

オプション

以下にオプション文字列に指定するオプションを示します。オプション文字列は下記に示す各オプションをカンマ(,)でつなげた文字列となります。

オプション	必須	説明	値範囲	デフォルト値
Condition	○	アクセス対象の条件番号を指定します。	1 - 127: 条件指定	1

データ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_R4 or VT_EMPTY	上限 1(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります。
1	VT_BSTR	上限 1(上下限コード表参照) 単位
2	VT_R4 or VT_EMPTY	下限 1(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります。
3	VT_BSTR	下限 1(上下限コード表参照) 単位
4	VT_R4 or VT_EMPTY	上限 2(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります。
5	VT_BSTR	上限 2(上下限コード表参照) 単位
6	VT_R4 or VT_EMPTY	下限 2(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります。
7	VT_BSTR	下限 2(上下限コード表参照) 単位

型説明		
8	VT_R4 or VT_EMPTY	上限 3(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
9	VT_BSTR	上限 3(上下限コード表参照) 単位
10	VT_R4 or VT_EMPTY	下限 3(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
11	VT_BSTR	下限 3(上下限コード表参照) 単位
12	VT_R4 or VT_EMPTY	上限 4(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
13	VT_BSTR	上限 4(上下限コード表参照) 単位
14	VT_R4 or VT_EMPTY	下限 4(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
15	VT_BSTR	下限 4(上下限コード表参照) 単位
16	VT_R4 or VT_EMPTY	上限 5(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
17	VT_BSTR	上限 5(上下限コード表参照) 単位
18	VT_R4 or VT_EMPTY	下限 5(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
19	VT_BSTR	下限 5(上下限コード表参照) 単位
20	VT_R4 or VT_EMPTY	上限 6(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
21	VT_BSTR	上限 6(上下限コード表参照) 単位
22	VT_R4 or VT_EMPTY	下限 6(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
23	VT_BSTR	下限 6(上下限コード表参照) 単位
24	VT_R4 or VT_EMPTY	上限 7(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
25	VT_BSTR	上限 7(上下限コード表参照) 単位
26	VT_R4 or VT_EMPTY	下限 7(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
27	VT_BSTR	下限 7(上下限コード表参照) 単位
28	VT_R4 or VT_EMPTY	上限 8(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
29	VT_BSTR	上限 8(上下限コード表参照) 単位

型説明		
30	VT_R4 or VT_EMPTY	下限 8(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
31	VT_BSTR	下限 8(上下限コード表参照) 単位
32	VT_R4 or VT_EMPTY	上限 9(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
33	VT_BSTR	上限 9(上下限コード表参照) 単位
34	VT_R4 or VT_EMPTY	下限 9(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
35	VT_BSTR	下限 9(上下限コード表参照) 単位
36	VT_R4 or VT_EMPTY	上限 10(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
37	VT_BSTR	上限 10(上下限コード表参照) 単位
38	VT_R4 or VT_EMPTY	下限 10(上下限コード表参照) ※ 値が設定されていない場合は VT_EMPTY となります.
39	VT_BSTR	下限 10(上下限コード表参照) 単位

3.3.1.19. ENVELOPE<??>

エンベロープ【項目番号:17】を取得します。ENVELOPE の後に任意の文字列を入力して変数を作成してください。エンベロープ上下限コードは「取扱説明書」をご覧ください。

オプション

以下にオプション文字列に指定するオプションを示します。オプション文字列は下記に示す各オプションをカンマ(,)でつなげた文字列となります。

オプション	必須	説明	値範囲	デフォルト値
Condition	○	アクセス対象の条件番号を指定します。	1 - 20: 条件指定	1

データ型

型説明		
VT_ARRAY VT_VARIANT		
0	VT_R4	OFFSET UPPER ENVE#1 SEGM#1
1	VT_BSTR	OFFSET UPPER ENVE#1 SEGM#1 単位

型説明		
2	VT_R4	OFFSET UPPER ENVE#1 SEGM#2
3	VT_BSTR	OFFSET UPPER ENVE#1 SEGM#2 単位
4	VT_R4	OFFSET UPPER ENVE#1 SEGM#3
5	VT_BSTR	OFFSET UPPER ENVE#1 SEGM#3 単位
6	VT_R4	OFFSET UPPER ENVE#2 SEGM#1
7	VT_BSTR	OFFSET UPPER ENVE#2 SEGM#1 単位
8	VT_R4	OFFSET UPPER ENVE#2 SEGM#2
9	VT_BSTR	OFFSET UPPER ENVE#2 SEGM#2 単位
10	VT_R4	OFFSET UPPER ENVE#2 SEGM#3
11	VT_BSTR	OFFSET UPPER ENVE#2 SEGM#3 単位
12	VT_R4	OFFSET LOWER ENVE#1 SEGM#1
13	VT_BSTR	OFFSET LOWER ENVE#1 SEGM#1 単位
14	VT_R4	OFFSET LOWER ENVE#1 SEGM#2
15	VT_BSTR	OFFSET LOWER ENVE#1 SEGM#2 単位
16	VT_R4	OFFSET LOWER ENVE#1 SEGM#3
17	VT_BSTR	OFFSET LOWER ENVE#1 SEGM#3 単位
18	VT_R4	OFFSET LOWER ENVE#2 SEGM#1

型説明		
19	VT_BSTR	OFFSET LOWER ENVE#2 SEGM#1 単位
20	VT_R4	OFFSET LOWER ENVE#2 SEGM#2
21	VT_BSTR	OFFSET LOWER ENVE#2 SEGM#2 単位
22	VT_R4	OFFSET LOWER ENVE#2 SEGM#3
23	VT_BSTR	OFFSET LOWER ENVE#2 SEGM#3 単位
24	VT_R4	START TIME UPPER ENVE#1 SEGM#1
25	VT_BSTR	START TIME UPPER ENVE#1 SEGM#1 単位
26	VT_R4	START TIME UPPER ENVE#1 SEGM#2
27	VT_BSTR	START TIME UPPER ENVE#1 SEGM#2 単位
28	VT_R4	START TIME UPPER ENVE#1 SEGM#3
29	VT_BSTR	START TIME UPPER ENVE#1 SEGM#3 単位
30	VT_R4	START TIME UPPER ENVE#2 SEGM#1
31	VT_BSTR	START TIME UPPER ENVE#2 SEGM#1 単位
32	VT_R4	START TIME UPPER ENVE#2 SEGM#2
33	VT_BSTR	START TIME UPPER ENVE#2 SEGM#2 単位
34	VT_R4	START TIME UPPER ENVE#2 SEGM#3
35	VT_BSTR	START TIME UPPER ENVE#2 SEGM#3 単位

型説明		
36	VT_R4	START TIME LOWER ENVE#1 SEGM#1
37	VT_BSTR	START TIME LOWER ENVE#1 SEGM#1 単位
38	VT_R4	START TIME LOWER ENVE#1 SEGM#2
39	VT_BSTR	START TIME LOWER ENVE#1 SEGM#2 単位
40	VT_R4	START TIME LOWER ENVE#1 SEGM#3
41	VT_BSTR	START TIME LOWER ENVE#1 SEGM#3 単位
42	VT_R4	START TIME LOWER ENVE#2 SEGM#1
43	VT_BSTR	START TIME LOWER ENVE#2 SEGM#1 単位
44	VT_R4	START TIME LOWER ENVE#2 SEGM#2
45	VT_BSTR	START TIME LOWER ENVE#2 SEGM#2 単位
46	VT_R4	START TIME LOWER ENVE#2 SEGM#3
47	VT_BSTR	START TIME LOWER ENVE#2 SEGM#3 単位
48	VT_R4	END TIME UPPER ENVE#1 SEGM#1
49	VT_BSTR	END TIME UPPER ENVE#1 SEGM#1 単位
50	VT_R4	END TIME UPPER ENVE#1 SEGM#2
51	VT_BSTR	END TIME UPPER ENVE#1 SEGM#2 単位
52	VT_R4	END TIME UPPER ENVE#1 SEGM#3

型説明		
53	VT_BSTR	END TIME UPPER ENVE#1 SEGM#3 単位
54	VT_R4	END TIME UPPER ENVE#2 SEGM#1
55	VT_BSTR	END TIME UPPER ENVE#2 SEGM#1 単位
56	VT_R4	END TIME UPPER ENVE#2 SEGM#2
57	VT_BSTR	END TIME UPPER ENVE#2 SEGM#2 単位
58	VT_R4	END TIME UPPER ENVE#2 SEGM#3
59	VT_BSTR	END TIME UPPER ENVE#2 SEGM#3 単位
60	VT_R4	END TIME LOWER ENVE#1 SEGM#1
61	VT_BSTR	END TIME LOWER ENVE#1 SEGM#1 単位
62	VT_R4	END TIME LOWER ENVE#1 SEGM#2
63	VT_BSTR	END TIME LOWER ENVE#1 SEGM#2 単位
64	VT_R4	END TIME LOWER ENVE#1 SEGM#3
65	VT_BSTR	END TIME LOWER ENVE#1 SEGM#3 単位
66	VT_R4	END TIME LOWER ENVE#2 SEGM#1
67	VT_BSTR	END TIME LOWER ENVE#2 SEGM#1 単位
68	VT_R4	END TIME LOWER ENVE#2 SEGM#2
69	VT_BSTR	END TIME LOWER ENVE#2 SEGM#2 単位

型説明		
70	VT_R4	END TIME LOWER ENVE#2 SEGM#3
71	VT_BSTR	END TIME LOWER ENVE#2 SEGM#3 単位

3.4. メッセージ一覧

コントローラのエラー通知や状態の変化を OnMessage イベントとして受け取ることが可能です。

番号	説明
0	受信時にエラーが発生しました。 データとしてエラーコードを保持します。
1	測定を受信しました。 データ型については 3.3.1.3 を参照してください。
2	波形を受信しました。 データ型については 3.3.1.4 を参照してください。
3	電流オールサイクルを受信しました。 データ型については 3.3.1.5 を参照してください。
6	測定履歴を受信しました。 データ型については 3.3.1.6 を参照してください。
7	異常履歴を受信しました。 データ型については 3.3.1.7 を参照してください。

4. MM400 プロバイダによるプログラミング

MM400 プロバイダでは、以下の手順でクライアント PC とウエルドチェッカーを接続することができます。

- CaoEngine の作成
- CaoWorkspace の作成
- CaoController の作成

ウエルドチェッカーに接続した後は、CaoController の OnMessage イベントを利用する、もしくは、CaoVariable オブジェクトを生成することで、ウエルドチェッカーでの測定値、設定情報にアクセスすることができます。

4.1. 測定を自動受信するサンプルプログラミング

ここでは例としてウエルドチェッカーの測定を自動で受信するサンプルプログラムを示します。表 4-A にサンプルプログラムの要件を、図 4-1 にサンプルプログラムの流れをそれぞれ記述しています。

表 4-A サンプルプログラムの要件

要件	説明
接続先	TCP/IP で接続する
	接続先 IP アドレスは 192.168.1.10
	接続先ポート番号は 1024
	通信方式は単方向
	通信項目は測定結果
	小数点方式はピリオド(.)
処理内容	ウエルドチェッカーからの測定結果を OnMessage で受信する。

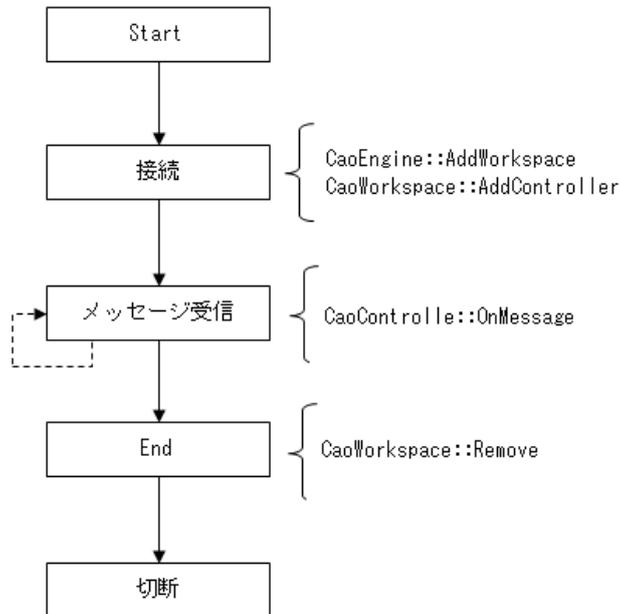


図 4-1 測定結果受信の流れ

以降の節から具体的なコードを示します。

4.1.1. サンプルプログラム

以下にサンプルプログラムの全体像を示します。

Sample	WeldCheckerFeature.vb
<pre> ' 変数宣言 Private engine As CaoEngine Private workspace As CaoWorkspace Private WithEvents controller As CaoController ' 接続メソッド Sub Connect () ' CaoEngine オブジェクトの作成 Set engine = New CaoEngine ' CaoWorkspace オブジェクトの取得 Set workspace = engine.Workspaces.Item(0) ' CaoController オブジェクトの作成(単方向, 受信項目を測定値に指定) Set controller = workspace.AddController("AMADA", _ "CaoProv. AMADA. MM400", _ "", _ "conn=tcp:192.168.1.10:1024, Mode=1") End Sub ' 切断メソッド Sub Disconnect () workspace.Controllers.Remove (controller.Index) Set controller = Nothing </pre>	

```
Call engine.Workspaces.Remove(workspace.Index)
' CaoWorkspace の消去
Set workspace = Nothing
' CaoEngine の消去
Set engine = Nothing
End Sub

' メッセージ受信メソッド
Private Sub controller_OnMessage(ByVal pICaoMess As CAOLib.ICaoMessage)
    If pICaoMess.Number = 1 Then
        ' 受信メッセージ番号が測定値であれば値を取得
        Dim measValues As Variant
        measValues = pICaoMess.Value

        ' 装置 No
        Dim machineNo As Long
        machineNo = measValues(0)

        ' 条件番号
        Dim condition As Long
        condition = measValues(1)

        ' 測定日時
        Dim measDate As Date
        measDate = measValues(2)

        ' 測定項目 1 ~ 最大測定項目までのループ
        For i = 3 To UBound(measValues) Step 4
            ' 測定項目コードの取得
            Dim term As Integer
            term = measValues(i)

            ' 判定コードの取得
            Dim result As String
            result = measValues(i + 1)

            ' 測定値の取得
            Dim value As Double
            value = measValues(i + 2)

            ' 単位の取得
            Dim unit As String
            unit = measValues(i + 3)

            Next i
        End If
    End Sub
```

4.1.1.1. 接続

ウェルドチェッカーと接続するためには、以下の手順を取ります。

- (1) オブジェクトを保持するための変数を用意します。コントローラ接続に必要なオブジェクトは、CaoEngineオブジェクトとCaoWorkspaceオブジェクトとCaoControllerオブジェクトです。VB6で

は, CaoControllerのイベントをハンドルするためにはCaoController型変数を“WithEvents”付きで宣言する必要があります。

```
' CaoEngineオブジェクト用の変数
Private engine As CaoEngine
' CaoWorkspaceオブジェクト用の変数
Private workspace As CaoWorkspace
' CaoControllerオブジェクト用の変数
Private WithEvents controller As CaoController
```

- (2) CaoEngineオブジェクトを生成します。CaoEngineオブジェクトはNewキーワードを使って生成します。

```
' CaoEngine オブジェクトの生成
Set engine = New CaoEngine
```

- (3) CaoWorkspaceオブジェクトを取得もしくは生成します。CaoEngineオブジェクトを生成すると、デフォルトでCaoWorkspacesオブジェクトとCaoWorkspaceオブジェクトを1つずつ生成していません。以下にCaoWorkspaceオブジェクトを新しく生成するコード例とデフォルトのCaoWorkspaceを示します。

```
' CaoWorkspace オブジェクトの生成
Set workspace = engine.AddWorkspace("NewWrks", "")

' デフォルトの CaoWorkspace オブジェクトの取得
Set workspace = engine.Workspaces.Item(0)
```

- (4) CaoControllerオブジェクトを生成します。CaoControllerオブジェクトを生成するには、使用するプロバイダ名と使用するためのパラメータを設定します。MM400プロバイダでは、接続先情報、通信方式(単方向/双方向)をオプションで指定します。サンプルでは以下の内容を指定してCaoControllerを作成しています。

- TCP/IP接続
- 接続先IP: 192.168.1.10, 接続先ポート: 1024
- 通信方式は単方向
- 受信項目は測定値

```
' CaoController オブジェクトの作成(単方向, 受信項目を測定値に指定)
Set controller = workspace.AddController("AMADA", _
    "CaoProv. AMADA. MM400", _
    "", _
    "conn=tcp:192.168.1.10:1024, Mode=1")
```

4.1.1.2. メッセージ受信

MM400 プロバイダの CaoController を単方向方式で作成した場合、MM400 の情報は CaoController::OnMessage イベントをハンドリングすることで取得可能となります。サンプルプログラムでは、controller_OnMessage イベントハンドラを用意して CaoController::OnMessage イベントをハンドリングしています。各メッセージの仕様については、3.4 を参照してください。

```
If pICaoMess.Number = 1 Then
    ' 受信メッセージ番号が測定値であれば値を取得
    Dim measValues As Variant
    measValues = measurement.value

    ' 装置 No の取得
    Dim machineNo As Long
    machineNo = measValues(0)
    ' 条件番号の取得
    Dim condition As Long
    condition = measValues(1)
    ' 測定日時の取得
    Dim measDate As Date
    measDate = measValues(2)
    ' 測定項目 1 ~ 最大測定項目までのループ
    For i = 3 To UBound(measValues) Step 4
        ' 測定項目コードの取得
        Dim term As Integer
        term = measValues(i)

        ' 判定コードの取得
        Dim result As String
        result = measValues(i + 1)

        ' 測定値の取得
        Dim value As Double
        value = measValues(i + 2)

        ' 単位の取得
        Dim unit As String
        unit = measValues(i + 3)
    Next i
End If
```

4.1.1.3. 切断

コントローラと切断する場合には、生成したオブジェクトを消去すると共に、オブジェクトを管理するコレクションクラスから消去するオブジェクトを削除します。以下にコード例を示します。

```
' CaoWorkspace から CaoController を削除
Call workspace.Controllers.Remove(controller.Index)
' CaoController の消去
```

```

Set ctrl = Nothing
' CaoEngine から CaoWorkspace を削除
Call engine.Workspaces.Remove(workspace.Index)
' CaoWorkspace の消去
Set workspace = Nothing
' CaoEngine の消去
Set engine = Nothing

```

4.2. 測定を取得するサンプルプログラミング

4.1 とは異なりこのサンプルプログラムでは、双方向モードで測定を取得する例を提示します。

表 4-2 サンプルプログラムの要件

要件	説明
接続先	TCP/IP で接続する
	接続先 IP アドレスは 192.168.1.10
	接続先ポート番号は 1024
	通信方式は双方向
	小数点方式はピリオド(.)
処理内容	測定結果取得変数を作成する。
	測定結果取得変数から値を取得

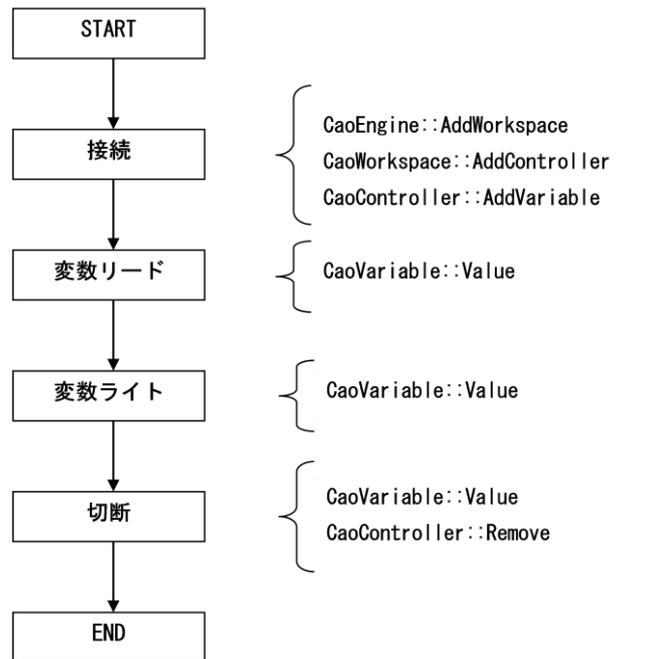


図 4-2 測定結果取得の流れ

4.2.1. サンプルプログラム

以下にサンプルプログラムの全体像を示します。

Sample	GetMeasurement.vb
<pre> ' 変数宣言 Private engine As CaoEngine Private workspace As CaoWorkspace Private controller As CaoController Private measurement As CaoVariable Sub Main() ' CaoEngine オブジェクトの作成 Set engine = New CaoEngine ' CaoWorkspace オブジェクトの取得 Set workspace = engine.Workspaces.Item(0) ' CaoController オブジェクトの作成(双方向) Set controller = workspace.AddController("AMADA", _ "CaoProv. AMADA. MM400", _ "", _ "conn=tcp:192.168.1.10:1024, Mode=2") ' 測定変数追加 Set measurement = controller.AddVariable("@MEASUREMENT") </pre>	

```
' 受信メッセージ番号が測定値であれば値を取得
Dim measValues As Variant
measValues = measurement.value

' 装置 No
Dim machineNo As Long
machineNo = measValues(0)
' 条件番号
Dim condition As Long
condition = measValues(1)
' 測定日時
Dim measDate As Date
measDate = measValues(2)
' 測定項目 1 ~ 最大測定項目までのループ
For i = 3 To UBound(measValues) Step 4
    ' 測定項目コードの取得
    Dim term As Integer
    term = measValues(i)
    ' 判定コードの取得
    Dim result As String
    result = measValues(i + 1)
    ' 測定値の取得
    Dim value As Double
    value = measValues(i + 2)
    ' 単位の取得
    Dim unit As String
    unit = measValues(i + 3)
Next i

controller.Variables.Remove(measurement.Index)
Set measurement = Nothing

workspace.Controllers.Remove(controller.Index)
Set controller = Nothing
Call engine.Workspaces.Remove(workspace.Index)
' CaoWorkspace の消去
Set workspace = Nothing
' CaoEngine の消去
Set engine = Nothing
End Sub
```

4.2.1.1. 接続

ウエルドチェッカーと接続するためには、以下の手順を取ります。

- (1) オブジェクトを保持するための変数を用意します。コントローラ接続に必要なオブジェクトは、CaoEngineオブジェクトとCaoWorkspaceオブジェクトとCaoControllerオブジェクトです。また双

方向方式で接続する場合にウェルドチェッカの情報を取得するために、CaoVariableオブジェクトも用意する必要があります。

Private engine As CaoEngine	' CaoEngineオブジェクト用の変数
Private workspace As CaoWorkspace	' CaoWorkspaceオブジェクト用の変数
Private controller As CaoController	' CaoControllerオブジェクト用の変数
Private measurement As CaoVariable	' CaoVariableオブジェクト用の変数

- (2) CaoEngineオブジェクトを生成します。CaoEngineオブジェクトはNewキーワードを使って生成します。

```
' CaoEngine オブジェクトの生成
Set engine = New CaoEngine
```

- (3) CaoWorkspaceオブジェクトを取得もしくは生成します。CaoEngineオブジェクトを生成すると、デフォルトでCaoWorkspacesオブジェクトとCaoWorkspaceオブジェクトを1つずつ生成しています。以下にCaoWorkspaceオブジェクトを新しく生成するコード例とデフォルトのCaoWorkspaceを示します。

```
' CaoWorkspace オブジェクトの生成
Set workspace = engine.AddWorkspace("NewWrks", "")
```

```
' デフォルトの CaoWorkspace オブジェクトの取得
Set workspace = engine.Workspaces.Item(0)
```

- (4) CaoControllerオブジェクトを生成します。CaoControllerオブジェクトを生成するには、使用するプロバイダ名と使用するためのパラメータを設定します。MM400プロバイダでは、接続先情報、通信方式(単方向/双方向)をオプションで指定します。サンプルでは以下の内容を指定してCaoControllerを作成しています。

- TCP/IP接続
- 接続先IP: 192.168.1.10, 接続先ポート: 1024
- 通信方式は双方向

```
' CaoController オブジェクトの作成(単方向, 受信項目を測定値に指定)
Set controller = workspace.AddController("AMADA", _
    "CaoProv. AMADA. MM400", _
    "", _
    "conn=tcp:192.168.1.10:1024, Mode=2")
```

- (5) 双方向方式の場合にウェルドチェッカーの情報を取得/設定するためにはCaoVariableオブジェクトをCaoControllerオブジェクトに追加する必要があります。サンプルプログラムでは測定の取得を行うための変数@MEASUREMENTを追加してます。

```
' 測定変数追加
Set measurement = controller.AddVariable("@MEASUREMENT")
```

4.2.1.2. 測定の取得

値を取得するためには、追加した CaoVariable 変数の Value プロパティを使用します。@MEASUREMENT 変数の Value プロパティは VT_VARIANT | VT_ARRAY 型のデータとなります。詳細は 3.3 を参照してください。

```
' 値を取得
Dim measValues As Variant
measValues = measurement.value
' 装置 No
Dim machineNo As Long
machineNo = measValues(0)
' 条件番号
Dim condition As Long
condition = measValues(1)
' 測定日時
Dim measDate As Date
measDate = measValues(2)
' 測定項目 1 ~ 最大測定項目までのループ
For i = 3 To UBound(measValues) Step 4
    ' 測定項目コードの取得
    Dim term As Integer
    term = measValues(i)
    ' 判定コードの取得
    Dim result As String
    result = measValues(i + 1)
    ' 測定値の取得
    Dim value As Double
    value = measValues(i + 2)
    ' 単位の取得
    Dim unit As String
    unit = measValues(i + 3)
Next i
```

4.2.1.3. 切断

コントローラと切断する場合には、生成したオブジェクトを消去すると共に、オブジェクトを管理するコレクションクラスから消去するオブジェクトを削除します。以下にコード例を示します。

```
' CaoController から CaoVariable を削除
Call controller.Variables.Remove(measurement.Index)
Set measurement = Nothing
' CaoWorkspace から CaoController を削除
Call workspace.Controllers.Remove(controller.Index)
' CaoController の消去
Set ctrl = Nothing
' CaoEngine から CaoWorkspace を削除
Call engine.Workspaces.Remove(workspace.Index)
' CaoWorkspace の消去
```

Set workspace = Nothing

' CaoEngine の消去

Set engine = Nothing

5. MM400 プロバイダエラーコード

MM400 プロバイダには、0x8011****でマスクした以下の独自エラーコードが存在します。(表 5-A 独自エラーコード表参照)

ORiN2 の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。

表 5-A 独自エラーコード表

エラー番号	説明
0x80110001	必須オプションが指定されていません。必須列に○と記述されているオプションは必須オプションです。必ずご指定ください。
0x80110002	通信種別が異なります。Com/Eth/Tcpのいずれかを指定してください。
0x80110003	指定された値が範囲外です。各項目に記載されている範囲内の値を指定してください。
0x80110005	受信データにヘッダ:"!"が存在しません。対応機種でない可能性があります。製造元にお問い合わせください。
0x80110006	測定データの解析に失敗しました。ワールドチェッカー側の小数点方式とプロバイダ側の小数点方式が異なる可能性があります。 CaoWorkspace::AddControllerのDecimalPointオプションの指定とワールドチェッカー側の設定が一致しているかをご確認ください。
0x80110007	指定された送信項目と異なる項目データを受信しました。ワールドチェッカーの送信項目とプロバイダの受信項目が異なる可能性があります。
0x80110008	要求した条件番号と異なる条件番号のデータを受信しました。ワールドチェッカー側で対応していない条件番号の可能性があります。 CaoController::AddVariableのConditionオプションで指定した値を見直してください。
0x80110009	送信したマシン番号と異なるマシン番号からデータを受信しました。ワールドチェッカーのマシン番号とプロバイダ側のマシン番号が異なる可能性があります。CaoWorkspace::AddControllerのIDオプションの指定とワールドチェッカー側の設定が一致しているかをご確認ください。

6. 付録

付録A. 通信プロトコル対応表

各変数と取扱説明書に記述されている通信項目との対応表をここに列挙します。

変数	通信項目
@MEASUREMENT	測定【項目番号:1】
@WAVEFORM	波形【項目番号:2】
@CURRENT_ALLCYCLE	電流オールサイクル【項目番号:3】
@MEASUREMENT_HISTORY	測定履歴【項目番号:6】
@ALARM_HISTORY	異常履歴【項目番号:7】
@DISPLAY_SETTING	表示設定【項目番号:11】
@SETTING_CONDITION	条件[設定]【項目番号:14】
@EXTERNAL_IO	外部入出力【項目番号:18】
@PRINTING	印刷【項目番号:21】
@COMMUNICATION	通信【項目番号:22】
@USB	USB【項目番号:23】
@MEMORY	内部メモリ【項目番号:24】
BASIC_CONDITION	条件[基本]【項目番号:12】
EXT_CONDITION	条件[拡張]【項目番号:13】
SEAM_CONDITION	条件[シーム]【項目番号:15】
RANGE_SETTING	上下限設定【項目番号:16】
ENVELOPE	エンベロープ【項目番号:17】

付録B. 測定項目コードとデータ型との対応表

測定項目		測定値	
コード	項目名	型	単位
0	電流 ピーク値	VT_R8	kA
1	電流 実効値		
2	電流 相加平均実効値		
3	電圧 ピーク値	VT_R8	V
4	電圧 実効値		
5	電圧 相加平均実効値		
6	通電角	VT_I4	deg
7	電力 平均値	VT_R8	kW
8	抵抗 平均値	VT_R8	mOhm
9	通電時間	VT_R8	CYC
			ms
10	通電時間 TP		VT_I4
11	通電時間 TH	VT_I4	
12	フロー時間	VT_I4	ms
13	変位置量溶接前	VT_R8	mm
14	変位置量 パルス1後	VT_R8	inch

測定項目		測定値	
コード	項目名	型	単位
15	変位量 パルス2後	VT_R8	
16	変位量 溶接後	VT_R8	
17	変位量 連続	VT_EMPTY	
18	加圧力 ピーク値	VT_R8	N
19	加圧力 平均値1	VT_R8	kgf
20	加圧力 平均値2	VT_R8	lbf
21	加圧力 溶接前	VT_R8	
22	加圧力 溶接後	VT_R8	
23	加圧力 連続	VT_EMPTY	
24	加圧力 時間	VT_I4	ms
25	外部 ピーク値	VT_R8	V
26	外部 平均値1	VT_R8	N
27	外部 平均値2	VT_R8	kgf
28	外部 溶接前	VT_R8	lbf
29	外部 溶接後	VT_R8	degC
30	外部 連続	VT_EMPTY	degF
			Mpa
			Bar
			Psi
31	外部 時間	VT_I4	ms
32	溶接カウンター	VT_I4	
33	良品カウンター	VT_I4	
34	設定なし	VT_EMPTY	

付録C. 上下限コードとデータ型との対応表

測定項目		測定値	
コード	項目名	型	単位
0	電流 ピーク値	VT_R8	kA
1	電流 実効値		
2	電流 相加平均実効値		
3	電圧 ピーク値	VT_R8	V
4	電圧 実効値		
5	電圧 相加平均実効値		
6	通電角	VT_EMPTY	deg
7	電力 平均値	VT_R8	kW
8	抵抗 平均値	VT_R8	mOhm
9	通電時間	VT_R8	CYC
			ms
10	通電時間 TP	VT_I4	ms
11	通電時間 TH	VT_I4	
12	フロー時間	VT_I4	ms
13	変位量溶接前	VT_R8	mm

測定項目		測定値	
コード	項目名	型	単位
14	変位量 パルス1後	VT_R8	inch
15	変位量 パルス2後	VT_R8	
16	変位量 溶接後	VT_R8	
17	変位量 連続	VT_EMPTY	
18	加圧力 ピーク値	VT_R8	N kgf lbf
19	加圧力 平均値1	VT_R8	
20	加圧力 平均値2	VT_R8	
21	加圧力 溶接前	VT_R8	
22	加圧力 溶接後	VT_R8	
23	加圧力 連続	VT_EMPTY	
24	加圧力 時間	VT_EMPTY	ms
25	外部 ピーク値	VT_R8	V
26	外部 平均値1	VT_R8	N
27	外部 平均値2	VT_R8	kgf
28	外部 溶接前	VT_R8	lbf
29	外部 溶接後	VT_R8	degC
30	外部 連続	VT_EMPTY	degF Mpa Bar Psi
31	外部 時間	VT_EMPTY	ms
32	溶接カウンター	VT_EMPTY	
33	良品カウンター	VT_EMPTY	
34	設定なし	VT_EMPTY	