

株式会社 安川電機
YASKAWA MOTOCOMES プロバイダ

Version 1.0.0

ユーザーズ ガイド

September 20, 2018

備考：

【改版履歴】

バージョン	日付	内容
1.0.0	2018-09-20	初版.

【動作確認機種】

機種	バージョン	注意事項
DX100 SIA30D TYPE:ERDR-SIA030D-A00	---	

目次

1. はじめに.....	5
1.1. 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. YRC とクライアント PC との接続.....	7
2.2. クライアント PC 開発環境のセットアップ.....	8
2.2.1. MOTOCOMES ライブラリの準備.....	8
2.2.2. YASKAWA MOTOCOMES プロバイダの手動インストール.....	8
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
3.2.3. CaoVariable クラス.....	12
3.2.3.1. Value プロパティ.....	12
3.3. 変数一覧.....	12
3.3.1. CaoController クラス変数.....	13
3.3.1.1. @MAKER_NAME.....	14
3.3.1.2. @VERSION.....	14
3.3.1.3. @ALARM.....	15
3.3.1.4. @ALARMEX.....	15
3.3.1.5. @STATUS.....	16
3.3.1.6. @BDSP.....	16
3.3.1.7. B<??>.....	17
3.3.1.8. I<??>.....	18
3.3.1.9. D<??>.....	18
3.3.1.10. R<??>.....	19
3.3.1.11. STR<??>.....	20

3.3.1.12. POS<??>	21
3.3.1.13. BASEPOS<??>	22
3.3.1.14. EXTPOS<??>	23
3.3.1.15. IO<??>	24
3.3.1.16. REG<??>	25
3.3.1.17. JOBSTATUS<??>	26
3.3.1.18. ALARMHIST<??>	27
3.3.1.19. ALARMHISTEX<??>	27
3.3.1.20. POSITION<??>	28
3.3.1.21. DEVIATION<??>	29
3.3.1.22. TORQUE<??>	30
3.3.1.23. MONITORINGTIME<??>	31
3.3.1.24. SYSTEMINFO<??>	32
4. YASKAWA MOTOCOMES プロバイダによるプログラミング	33
4.1. 変数(I)の値を取得, 設定するサンプルプログラミング	33
4.1.1. サンプルプログラム	34
4.1.1.1. 接続	36
4.1.1.2. 変数の取得/設定	37
4.1.1.3. 切断	37
5. MOTOCOMES プロバイダエラーコード	38

1. はじめに

本書は、株式会社 安川電機の産業ロボット MOTOMAN 用制御盤 YRC1000, YRC1000micro, DX200, DX100, FS100, FS100L (以降 FS100L は FS100 とします) に対してデータの取得及び設定をするプロバイダのユーザーズガイドです。以降株式会社 安川電機の産業ロボット MOTOMAN 用制御盤を YRC と呼称します。クライアント PC が Windows 上で動作し、対象とする YRC が Ethernet 接続可能である環境を想定しています。図 1-1 が本プロバイダとデバイスの全体構成図になります。以降本プロバイダを YASKAWA MOTOCOMES プロバイダと呼称します。

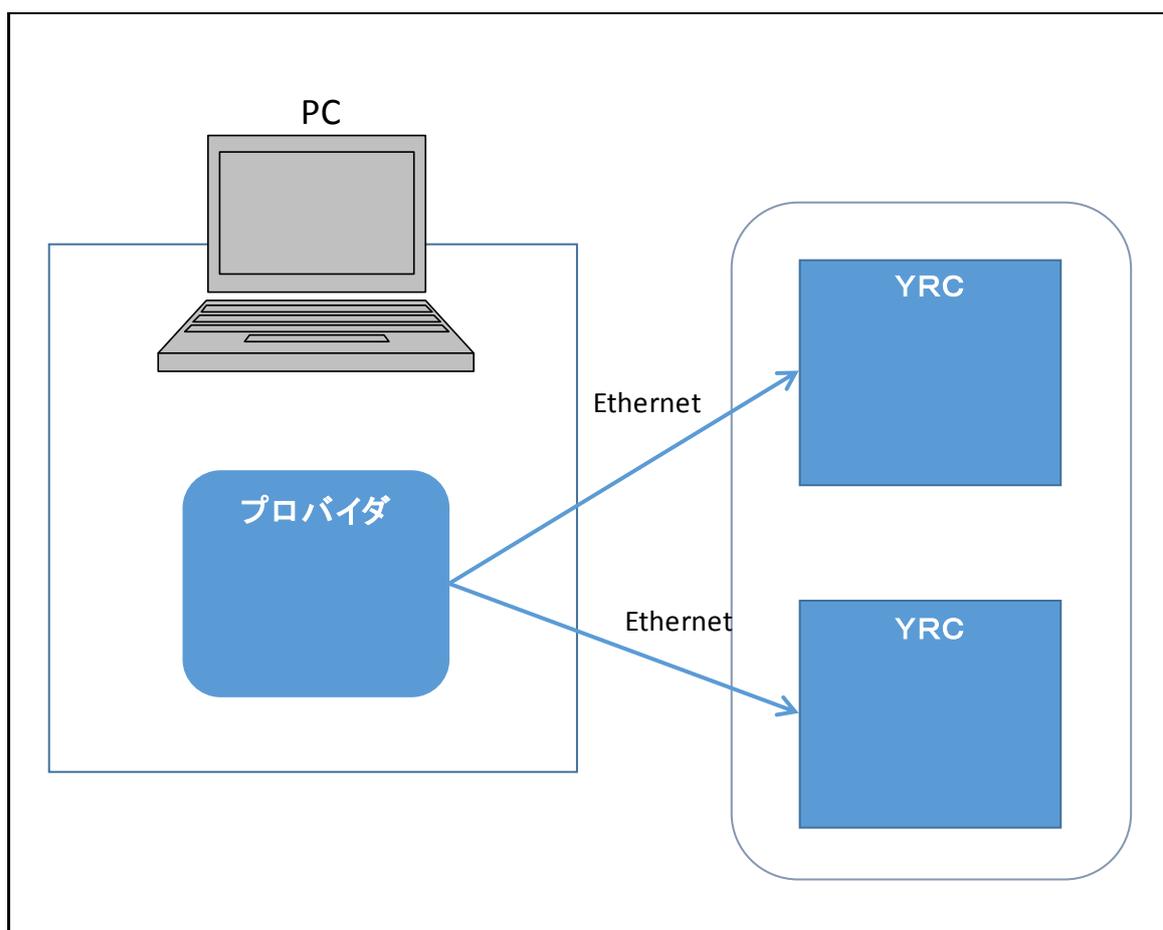


図 1-1 構成図

また、本プロバイダ及びデバイスそれぞれの対応例を図 1-2に示します。

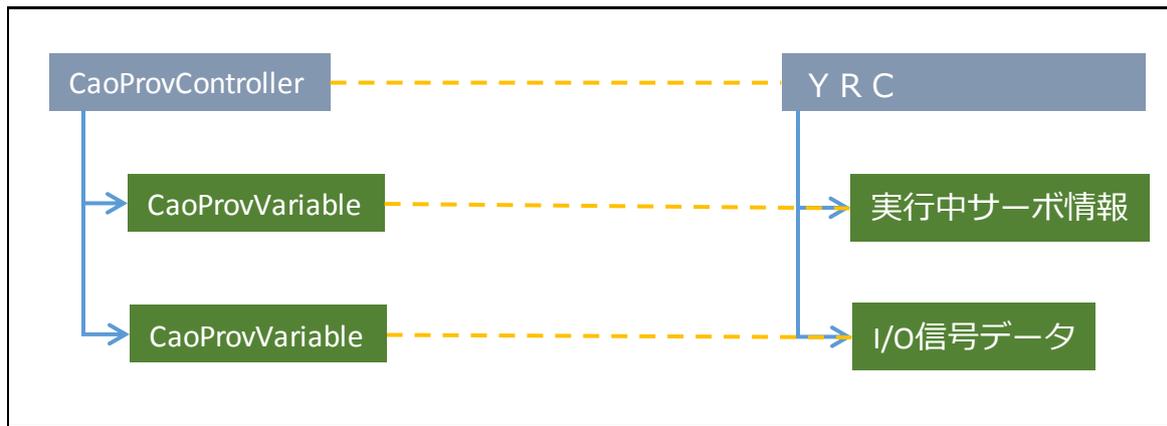


図 1-2 YASKAWA MOTOCOMES プロバイダと YRC との対応図

1.1. 参考となる情報源

YASKAWA MOTOCOMES プロバイダは、株式会社 安川電機から提供されている MOTOCOMES ライブラリに付随されている「MOTOCOMES 操作要領書（ファイル名：MOTOCOMES_jp.pdf）」を参考に開発しています。以降このマニュアルを MOTOCOMES マニュアルと呼称します。

以下参考資料

MOTOCOMES_jp.pdf （資料番号 HW1483704）

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. YRC とクライアント PC との接続

YRC とクライアント PC は MOTOCOMES ライブラリを介して Ethernet 通信にて接続されます。

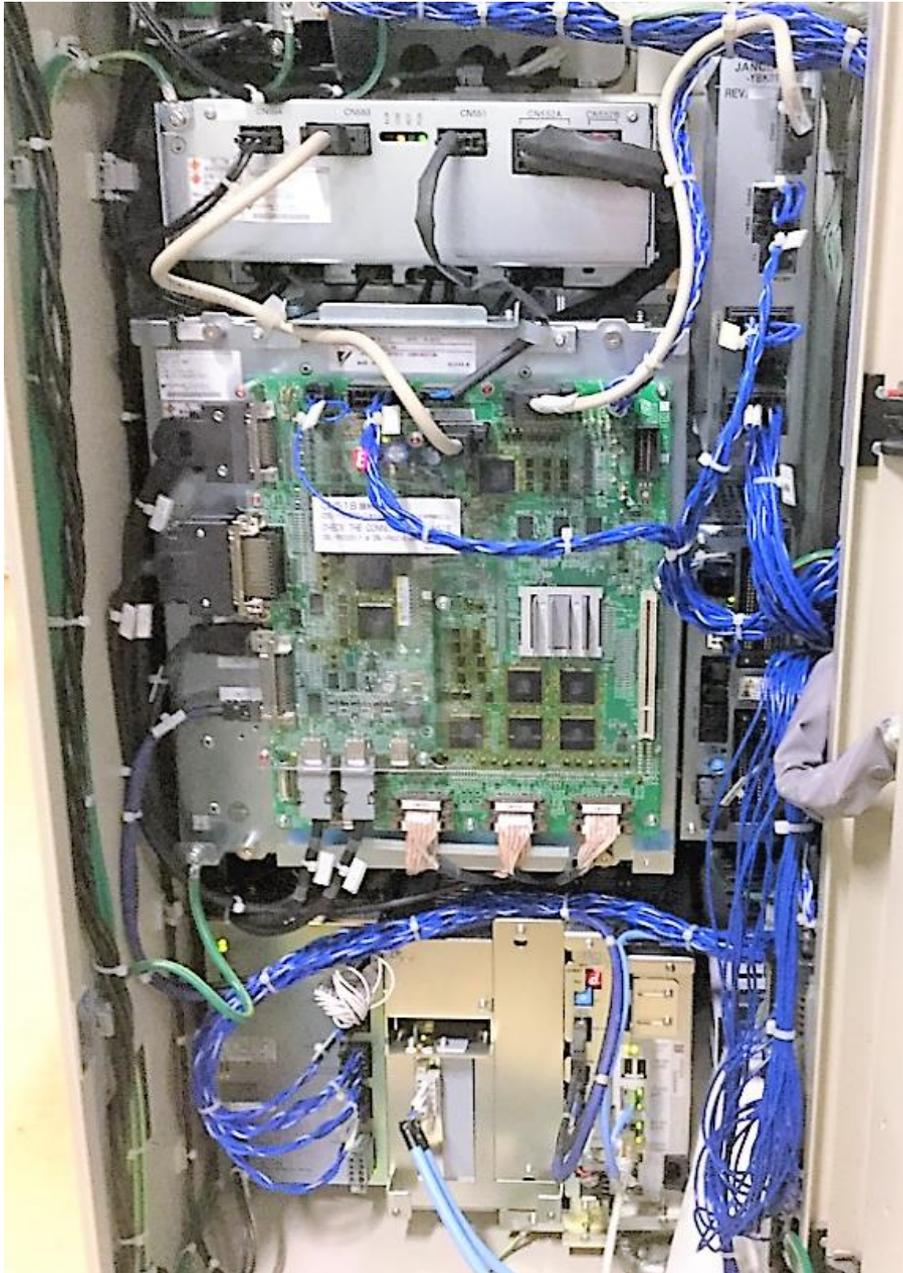


図 2-1 制御盤

LAN ケーブルを接続した後、MOTOCOMES マニュアルの「2.2 Ethernet を使用する場合の環境設定」を参照し、YRC とクライアント PC の設定を行ってください。

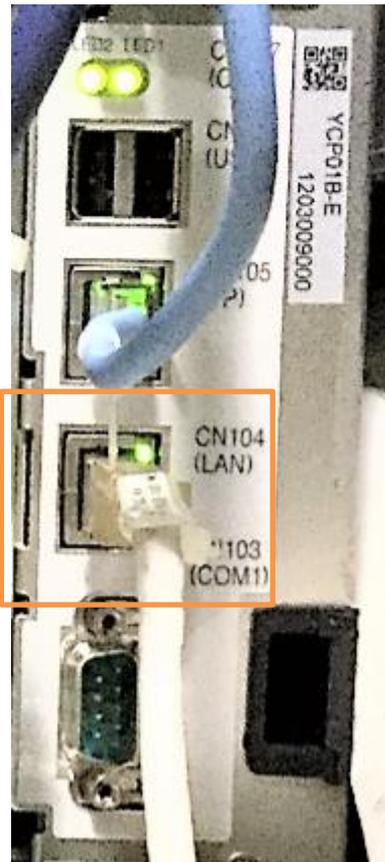


図 2-2 LAN ケーブルの接続

2.2. クライアント PC 開発環境のセットアップ

2.2.1. MOTOCOMES ライブラリの準備

YASKAWA MOTOCOMES プロバイダをクライアント PC にインストールするために、株式会社 安川電機から提供されている MOTOCOMES ライブラリを事前にクライアント PC にインストールする必要があります。MOTOCOMES ライブラリのインストール後に下記ファイルを、32bit OS の場合は C:\Windows\System32 に、64bit OS の場合は C:\Windows\SysWOW64 に配置してください。

表 2-1 必要な MOTOCOMES ライブラリ

DLL	バージョン	説明
MOTOCOMES.dll	3.1.0.0	データ伝送関数ライブラリ
MotoIkr.dll	5.0.0.0	MOTOCOMES ライブラリに必要なライブラリ
MotoLkr.dll	0.5.0.0	MOTOCOMES ライブラリに必要なライブラリ

2.2.2. YASKAWA MOTOCOMES プロバイダの手動インストール

YASKAWA MOTOCOMES プロバイダを使用するために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。

表 2-2 YASKAWA MOTOCOMES プロバイダ

ファイル名	CaoProvYASKAWAMOTOCOMES.dll
ProgID	CaoProv.YASKAWA.MOTOCOMES
レジストリ登録	regsvr32 CaoProvYASKAWAMOTOCOMES.dll
レジストリ登録の抹消	regsvr32 /u CaoProvYASKAWAMOTOCOMES.dll

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 3-1 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追加	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

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 YRC 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. YASKAWA. MOTOCOMES", // プロバイダ名(固定)
    "<マシン名>",                 // プロバイダ実行マシン名
    "<オプション>"                 // オプション文字列
)
```

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
-------	----	----	-----	--------

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。

オプション	必須	説明	値範囲	デフォルト値
Conn	○	YRC と接続するための Ethernet 接続オプションを指定します。詳細は 3.2.1.1.1 を参照して下さい。	--	--
CntIType	○	YRC と接続するためのコントローラタイプオプションを指定します。詳細は 3.2.1.1.2 を参照して下さい。	1 - 5	--

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

Set engine = New CaoEngine
Set workspace = engine.Workspaces.Item(0)
Set controller = workspace.AddController("Controller", _
    "CaoProv. YASKAWA. MOTOCOMES", _
    "", _
    "Conn=eth:192.168.255.1, CntIType=1")
```

3.2.1.1.1. 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。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。

MOTOCOMES ライブラリでは、10040、10041 のポート番号を利用しています。MOTOCOMES マニュアルの「2.3.1 ロボットコントローラ/パソコン共通」を確認して下さい。ポート番号の変更は出来ないため、接続先ポートは指定しなくても構いません。

下線はデフォルト値を示します。

"Conn=Eth:<IP>[:<Port>]"

<IP> : 接続先 IP アドレス。

<Port> : 接続先ポート。

3.2.1.1.2. CntIType オプション

以下に CntIType オプションのコントローラタイプパラメータ文字列を示します。

対象のコントローラ	オプションの値
DX100	1
FS100	2
DX200	3

YRC1000	4
YRC100micro	5

3.2.2. CaoController クラス

3.2.2.1. VariableNames プロパティ

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

```
' 接続
Call Connect

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

' 切断
Call Disconnect
```

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 3.3.1 に示すもののみ使用できます。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)
)
```

3.2.3. CaoVariable クラス

3.2.3.1. Value プロパティ

接続した YRC 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. 変数一覧を参照してください。

3.3. 変数一覧

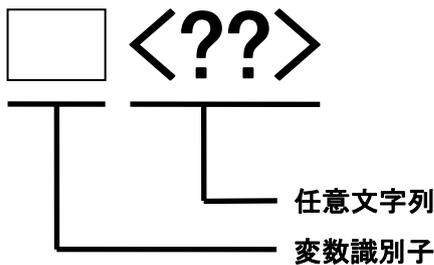
各クラスで使用可能な変数一覧を定義します。なお変数は、CaoVariable クラスのオブジェクトを指します。

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。	○	—	P. 14
@VERSION	プロバイダバージョンを取得します。	○	—	P. 14
@ALARM	現在発生中のアラームを取得します。	○	—	P. 15
@ALARMEX	現在発生中のアラームを取得（サブコード文字列対応）します。	○	—	P. 15
@STATUS	コントローラのステータスを取得します。	○	—	P. 16
@BDSP	ペンダントに文字列を表示します。	—	○	P. 16
B<??>	変数(B)の値を取得/設定します。	○	○	P. 17
I<??>	変数(I)の値を取得/設定します。	○	○	P. 18
D<??>	変数(D)の値を取得/設定します。	○	○	P. 18
R<??>	変数(R)の値を取得/設定します。	○	○	P. 19
STR<??>	S変数(文字列)の値を取得/設定します。	○	○	P. 20
POS<??>	ロボット位置型変数を取得/設定します。	○	○	P. 21
BASEPOS<??>	ベース位置型変数を取得/設定します。	○	○	P. 22
EXTPOS<??>	外部軸位置型変数を取得/設定します。	○	○	P. 23
IO<??>	I/Oの読み込み/書き込みをします。	○	○	P. 24
REG<??>	レジスタの読み込み/書き込みをします。	○	○	P. 25
JOBSTATUS<??>	ジョブステータスを取得します。	○	—	P. 26
ALARMHIST<??>	アラーム履歴を取得します。	○	—	P. 27

ALARMHISTEX<??>	アラーム履歴を取得(サブコード文字列対応) します.	○	—	P. 27
POSITION<??>	各軸位置を取得します.	○	—	P. 28
DEVIATION<??>	各軸位置偏差を取得します.	○	—	P. 29
TORQUE<??>	各軸トルクを取得します.	○	—	P. 30
MONITORINGTIME<??>	管理時間を取得します.	○	—	P. 31
SYSTEMINFO<??>	システム情報を取得します.	○	—	P. 32

3.3.1.1. @MAKER_NAME

メーカー名を取得します.

データ型

項目	型説明	
取得	VT_BSTR	メーカー名

使用例

```
' 接続
Call Connect
' 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取得します.

データ型

項目	型説明	
取得	VT_BSTR	プロバイダバージョン *. *.*

使用例

```
' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("@VERSION")
' 値取得
Dim value As String
value = var.value
```

3.3.1.3. @ALARM

現在発生中のアラームを取得します。

データ型

項目	型説明		
取得	VT_VARIANT VT_ARRAY		
	0-3	VT_VARIANT VT_ARRAY	アラームリスト 1～4
	0	VT_I4	アラームコード
	1	VT_I4	アラームデータ
	2	VT_I4	アラームデータ種別
	3	VT_BSTR	アラーム発生時刻
	4	VT_BSTR	アラーム文字列名称

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("@ALARM")
' 値取得
Dim value As Variant
value = var.value

For i = LBound(value) To UBound(value)
    MsgBox value(i)
Next

```

3.3.1.4. @ALARMEX

現在発生中のアラームを取得(サブコード文字列対応) します。

データ型

項目	型説明		
取得	VT_VARIANT VT_ARRAY		
	0-3	VT_VARIANT VT_ARRAY	アラームリスト 1～4
	0	VT_I4	アラームコード
	1	VT_I4	アラームデータ
	2	VT_I4	アラームデータ種別
	3	VT_BSTR	アラーム発生時刻
	4	VT_BSTR	アラーム文字列名称
	5	VT_BSTR	詳細データ追加情報文字列

		6	VT_BSTR	詳細データ文字列
		7	VT_BSTR	詳細データ反転表示情報

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("@ALARMEX")
' 値取得
Dim value As Variant
value = var.value

For i = LBound(value) To UBound(value)
    MsgBox value(i)
Next

```

3.3.1.5. @STATUS

コントローラのステータスを取得します。

データ型

項目	型説明		
取得	VT_VARIANT VT_ARRAY		
	0	VT_I4	ステータスデータ 1
	1	VT_I4	ステータスデータ 2

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("@STATUS")
' 値取得
Dim value As Integer
value = var.value

```

3.3.1.6. @BDSP

ペンダントに文字列を表示します。

データ型

項目	型説明	
設定	VT_BSTR	表示させたい文字列

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("@BDSP")
' 値設定
var.value = ("abc")

```

3.3.1.7. B<??>

変数(B)の値を取得/設定します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	変数番号	0~2147483647 ※1	--
Elem	○	指定した変数番号からの個数	2~2147483647 ※2	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetVarDataMB 及び ESSetVarDataMB」を参照して下さい。YRC の RS022 を確認して下さい。RS022=0 の場合は、変数番号に+1 する必要があります。

※2 API の制限があるので個数は偶数個にして下さい

データ型

項目	型説明		
取得/設定	VT_UI1 VT_ARRAY		
	n	VT_UI1	変数(B)の値

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("B1", "NO=1, ELEM=2")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
    value(i) = value(i) + (i + 1)
Next
var.value = value

```

3.3.1.8. I<??>

変数 (I) の値を取得/設定します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	変数番号	0~2147483647 ※1	--
Elem	○	指定した変数番号からの個数	1~2147483647	--
Array	--	個数に 1 を指定した際に、配列にするかのフラグ	True : する False : しない	False

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetVarDataMI 及び ESSetVarDataMI」を参照して下さい。YRC の RS022 を確認して下さい。RS022=0 の場合は、変数番号に+1 する必要があります。

データ型

項目	型説明		
取得/設定	VT_I2 VT_ARRAY		
	n	VT_I2	変数 (I) の値

使用例

```
' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("I1", "NO=1, ELEM=3")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
    value(i) = value(i) + (i + 1)
Next
var.value = value
```

3.3.1.9. D<??>

変数 (D) の値を取得/設定します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	変数番号	0~2147483647	--

			※1	
Elem	○	指定した変数番号からの個数	1~2147483647	--
Array	--	個数に 1 を指定した際に、配列にするかのフラグ	True : する False : しない	False

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetVarDataMD 及び ESSetVarDataMD」を参照して下さい YRC の RS022 を確認して下さい。RS022=0 の場合は、変数番号に+1 する必要があります。

データ型

項目	型説明		
取得/設定	VT_I4 VT_ARRAY		
	n	VT_I4	変数 (D) の値

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("D1", "NO=1, ELEM=3")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
    value(i) = value(i) + (i + 1)
Next
value = var.value

```

3.3.1.10. R<??>

変数 (R) の値を取得/設定します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	変数番号	0~2147483647 ※1	--
Elem	○	指定した変数番号からの個数	1~2147483647	--
Array	--	個数に 1 を指定した際に、配列にするかのフラグ	True : する False : しない	False

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetVarDataMR 及び ESSetVarDataMR」を参照して下さい YRC の RS022 を確認して下さい。RS022=0 の場合は、変数番号に+1 する必要があります。

ります。

データ型

項目	型説明		
取得/設定	VT_R8 VT_ARRAY		
	n	VT_R8	変数 (R) の値

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("R1", "NO=1, ELEM=3")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
    value(i) = value(i) + (i + 1)
Next
value = var.value

```

3.3.1.11. STR<??>

S 変数 (文字列) の値を取得/設定します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	変数番号	0~2147483647 ※1	--
Elem	○	指定した変数番号からの個数	1~2147483647	--
Array	--	個数に 1 を指定した際に、配列にするかのフラグ	True : する False : しない	False

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetStrDataM, ESSetStrDataM, ESGetStrDataM2 及び ESSetStrDataM2」を参照して下さい。YRC の RS022 を確認して下さい。RS022=0 の場合は、変数番号に+1 する必要があります。

データ型

項目	型説明		
取得/設定	VT_BSTR VT_ARRAY		
	n	VT_BSTR	S 変数 (文字列) の値

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("STR1", "NO=1, ELEM=3")
' 値取得
Dim value As Variant
value = var.value
' 値設定
Dim value As Variant
For i = LBound(value) To UBound(value)
    value(i) = value(i) + CStr(i + 1)
Next
value = var.value

```

3.3.1.12. POS<??>

ロボット位置型変数を取得/設定します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	変数番号	0~2147483647 ※1	--
Elem	○	指定した変数番号からの個数	1~2147483647	--
Array	--	個数に 1 を指定した際に、配列にするかのフラグ	True : する False : しない	False

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetPositionDataM 及び ESSetPositionDataM」を参照して下さい。YRC の RS022 を確認して下さい。RS022=0 の場合は、変数番号に+1 する必要があります。

データ型

項目	型説明		
取得/設定	VT_VARIANT VT_ARRAY		
	n	VT_VARIANT VT_ARRAY	n 個のデータ
	0	VT_I4	データタイプ(パルス値/直交値)
	1	VT_I4	形態
	2	VT_I4	ツール番号
	3	VT_I4	ユーザ座標番号
	4	VT_I4	拡張形態

	5	VT_R8	軸データ 1
	6	VT_R8	軸データ 2
	7	VT_R8	軸データ 3
	8	VT_R8	軸データ 4
	9	VT_R8	軸データ 5
	10	VT_R8	軸データ 6
	11	VT_R8	軸データ 7
	12	VT_R8	軸データ 8

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("POS1", "NO=1, ELEM=3")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
    For j = LBound(value(i)) To UBound(value(i))
        value(i)(j) = value(i)(j) + (j + 1)
    Next
Next
Next
value = var.value

```

3.3.1.13. BASEPOS<??>

ベース位置型変数を取得/設定します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	変数番号	0~2147483647 ※1	--
Elem	○	指定した変数番号からの個数	1~2147483647	--
Array	--	個数に 1 を指定した際に、配列にするかのフラグ	True : する False : しない	False

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetBpexPositionDataM 及び ESSetBpexPositionDataM」を参照して下さい。YRC の RS022 を確認して下さい。RS022=0 の場合は、変数番号に+1 する必要があります。

データ型

項目	型説明		
取得/設定	VT_VARIANT VT_ARRAY		
	n	VT_VARIANT VT_ARRAY	n 個のデータ
	0	VT_I4	データタイプ (パルス値/直交値)
	1	VT_R8	軸データ 1
	2	VT_R8	軸データ 2
	3	VT_R8	軸データ 3
	4	VT_R8	軸データ 4
	5	VT_R8	軸データ 5
	6	VT_R8	軸データ 6
	7	VT_R8	軸データ 7
8	VT_R8	軸データ 8	

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("BASEPOS1", "NO=1, ELEM=3")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
  For j = LBound(value(i)) To UBound(value(i))
    value(i)(j) = value(i)(j) + (j + 1)
  Next
Next
Next
value = var.value

```

3.3.1.14. EXTPOS<??>

外部軸位置型変数を取得/設定します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	変数番号	0~2147483647 ※1	--
Elem	○	指定した変数番号からの個数	1~2147483647	--
Array	--	個数に 1 を指定した際に、配列にするかのフラグ	True : する False : しない	False

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetBpexPositionDataM 及び

ESSetBpexPositionDataM」を参照して下さい。YRCのRS022を確認して下さい。RS022=0の場合は、変数番号に+1する必要があります。

データ型

項目	型説明		
取得/設定	VT_VARIANT VT_ARRAY		
	n	VT_VARIANT VT_ARRAY	n個のデータ
	0	VT_I4	データタイプ(パルス値/直交値)
	1	VT_R8	軸データ 1
	2	VT_R8	軸データ 2
	3	VT_R8	軸データ 3
	4	VT_R8	軸データ 4
	5	VT_R8	軸データ 5
	6	VT_R8	軸データ 6
	7	VT_R8	軸データ 7
8	VT_R8	軸データ 8	

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("EXTPOS1", "NO=1, ELEM=3")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
  For j = LBound(value(i)) To UBound(value(i))
    value(i)(j) = value(i)(j) + (j + 1)
  Next
Next
Next
value = var.value

```

3.3.1.15. IO<??>

I/Oの読み込み/書き込みをします。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	I/O アドレス	0~2147483647 ※1	--

Elem	○	指定した I/O アドレスからの個数	2~2147483647 ※2	--
------	---	--------------------	--------------------	----

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESReadIOM 及び ESWriteIOM」を参照して下さい

※2 API の制限があるので個数は偶数個にして下さい

データ型

項目	型説明		
取得/設定	VT_UI1 VT_ARRAY		
	n	VT_UI1	I/O の値

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("I01", "NO=1, ELEM=2")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
    value(i) = value(i) + (i + 1)
Next
value = var.value

```

3.3.1.16. REG<??>

レジスタの読み込み/書き込みをします。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	レジスタ番号	0~2147483647 ※1	--
Elem	○	指定したレジスタ番号からの個数	1~2147483647	--
Array	--	個数に 1 を指定した際に、配列にするかのフラグ	True : する False : しない	False

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESReadRegisterM 及び ESWriteRegisterM」を参照して下さい YRC の RS022 を確認して下さい。RS022=0 の場合は、変数番号に+1 する必要があります。

データ型

項目	型説明		
取得/設定	VT_I2 VT_ARRAY		
	n	VT_I2	レジスタの値

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("REG1", "NO=1, ELEM=3")
' 値取得
Dim value As Variant
value = var.value
' 値設定
For i = LBound(value) To UBound(value)
    value(i) = value(i) + (i + 1)
Next
value = var.value

```

3.3.1.17. JOBSTATUS<??>

ジョブステータスを取得します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	タスク番号	0~2147483647 ※1	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetJobStatus」を参照して下さい

データ型

項目	型説明		
取得	VT_VARIANT VT_ARRAY		
	0	VT_BSTR	ジョブ名称
	1	VT_I4	ライン番号
	2	VT_I4	ステップ番号
	3	VT_I4	速度オーバーライド値

使用例

```

' 接続
Call Connect
' 変数追加

```

```
Dim var As CaoVariable
Set var = controller.AddVariable("JOBSTATUS1", "NO=1")
' 値取得
Dim value As Variant
value = var.value
```

3.3.1.18. ALARMHIST<??>

アラーム履歴を取得します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	アラーム番号	0~2147483647 ※1	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetAlarmHist」を参照して下さい

データ型

項目	型説明		
取得	VT_VARIANT VT_ARRAY		
	0	VT_I4	アラームコード
	1	VT_I4	アラームデータ
	2	VT_I4	アラームデータ種別
	3	VT_BSTR	アラーム発生時刻
	4	VT_BSTR	アラーム文字列名称

使用例

```
' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("ALARMHIST1", "NO=1")
' 値取得
Dim value As Variant
value = var.value
```

3.3.1.19. ALARMHISTEX<??>

アラーム履歴を取得(サブコード文字列対応) します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	アラーム番号	0~2147483647 ※1	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetAlarmHistEx」を参照して下さい

データ型

項目	型説明		
取得	VT_VARIANT VT_ARRAY		
	0	VT_I4	アラームコード
	1	VT_I4	アラームデータ
	2	VT_I4	アラームデータ種別
	3	VT_BSTR	アラーム発生時刻
	4	VT_BSTR	アラーム文字列名称
	5	VT_BSTR	詳細データ追加情報文字列
	6	VT_BSTR	詳細データ文字列
	7	VT_BSTR	詳細データ反転表示情報

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("ALARMHISTEX1", "NO=1")
' 値取得
Dim value As Variant
value = var.value

```

3.3.1.20. POSITION<??>

各軸位置を取得します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	制御グループ	0~2147483647 ※1	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetPosition」を参照して下さい

データ型

項目	型説明		
取得	VT_VARIANT VT_ARRAY		
	0	VT_I4	データタイプ(パルス値/直交値)
	1	VT_I4	形態
	2	VT_I4	ツール番号

	3	VT_I4	ユーザ座標番号
	4	VT_I4	拡張形態
	5	VT_R8	軸データ 1
	6	VT_R8	軸データ 2
	7	VT_R8	軸データ 3
	8	VT_R8	軸データ 4
	9	VT_R8	軸データ 5
	10	VT_R8	軸データ 6
	11	VT_R8	軸データ 7
	12	VT_R8	軸データ 8

使用例

```
' 接続
```

```
Call Connect
```

```
' 変数追加
```

```
Dim var As CaoVariable
```

```
Set var = controller.AddVariable("POSITION1", "NO=1")
```

```
' 値取得
```

```
Dim value As Variant
```

```
value = var.value
```

3.3.1.21. DEVIATION<??>

各軸位置偏差を取得します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	制御グループ	0~2147483647 ※1	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetDeviation」を参照して下さい

データ型

項目	型説明		
取得	VT_R8 VT_ARRAY		
	0	VT_R8	軸データ 1
	1	VT_R8	軸データ 2
	2	VT_R8	軸データ 3
	3	VT_R8	軸データ 4
	4	VT_R8	軸データ 5
	5	VT_R8	軸データ 6

	6	VT_R8	軸データ 7
	7	VT_R8	軸データ 8

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("DEVIATION1", "NO=1")
' 値取得
Dim value As Variant
value = var.value

```

3.3.1.22. TORQUE<??>

各軸トルクを取得します。

オプション

オプション	必須	説明	値範囲	デフォルト値
No	○	制御グループ	0~2147483647 ※1	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetTorque」を参照して下さい

データ型

項目	型説明		
取得	VT_R8 VT_ARRAY		
	0	VT_R8	軸データ 1
	1	VT_R8	軸データ 2
	2	VT_R8	軸データ 3
	3	VT_R8	軸データ 4
	4	VT_R8	軸データ 5
	5	VT_R8	軸データ 6
	6	VT_R8	軸データ 7
	7	VT_R8	軸データ 8

使用例

```

' 接続
Call Connect
' 変数追加
Dim var As CaoVariable
Set var = controller.AddVariable("TORQUE1", "NO=1")

```

' 値取得

```
Dim value As Variant
value = var.value
```

3.3.1.23. MONITORINGTIME<??>

管理時間を取得します。

オプション

オプション	必須	説明	値範囲	デフォルト値
Type	○	管理時間の種別	0~2147483647 ※1	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetMonitoringTime」を参照して下さい
以下に MOTOCOMES マニュアルから抜粋した管理時間の種別を示します

値	説明
1	制御電源投入時間
10	サーボ電源投入時間 (TOTAL)
11~18	サーボ電源投入時間 (R1~R8)
21~44	サーボ電源投入時間 (S1~S24)
110	プレイバック時間 (TOTAL)
111~118	プレイバック時間 (R1~R8)
121~144	プレイバック時間 (S1~S24)
210	移動時間 (TOTAL)
211~218	移動時間 (R1~R8)
221~244	移動時間 (S1~S24)
301~308	作業時間 (用途 1~用途 8)

データ型

項目	型説明		
取得値	VT_BSTR VT_ARRAY		
	0	VT_BSTR	稼働開始時刻
	1	VT_BSTR	経過時間

使用例

' 接続

```
Call Connect
```

' 変数追加

```
Dim var As CaoVariable
```

```
Set var = controller.AddVariable("MONITORINGTIME1", "TYPE=1 ")
```

' 値取得

```
Dim value As Variant
value = var.value
```

3.3.1.24. SYSTEMINFO<??>

システム情報を取得します。

オプション

オプション	必須	説明	値範囲	デフォルト値
Type	○	システム情報の種別	0~2147483647 ※1	--

※1 MOTOCOMES マニュアルの「5.2 ロボット制御機能 ESGetSystemInfo」を参照して下さい
以下に MOTOCOMES マニュアルから抜粋したシステム情報の種別を示します

値	説明
11~18	機種情報 (R1~R8)
21~44	機種情報 (S1~S24)
101~108	用途情報 (R1~R8)

データ型

項目	型説明		
取得	VT_VARIANT VT_ARRAY		
	0	VT_BSTR	システムソフトウェアバージョン
	1	VT_BSTR	機種名称/用途名称
	2	VT_BSTR	パラメータ No.

使用例**' 接続**

```
Call Connect
```

' 変数追加

```
Dim var As CaoVariable
```

```
Set var = controller.AddVariable("SYSTEMINF01", "TYPE=11")
```

' 値取得

```
Dim value As Variant
```

```
value = var.value
```

4. YASKAWA MOTOCOMES プロバイダによるプログラミング

YASKAWA MOTOCOMES プロバイダでは、以下の手順でクライアント PC と YRC を接続することができます。

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

YRC に接続した後は、CaoVariable オブジェクトを生成することで、YRC の情報にアクセスすることができます。

4.1. 変数(I)の値を取得，設定するサンプルプログラミング

ここでは例として変数(I)の値を取得，設定するサンプルプログラムを示します。表 4-1 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 4-1 サンプルプログラムの要件

要件	説明
接続先	接続先 IP アドレスは 192.168.10.10
	接続先ロボットコントロールタイプは 1
処理内容	変数番号 10 の変数(I)の値を取得する
	変数番号 10 の変数(I)の値を取得した値 + 100 の値に設定する

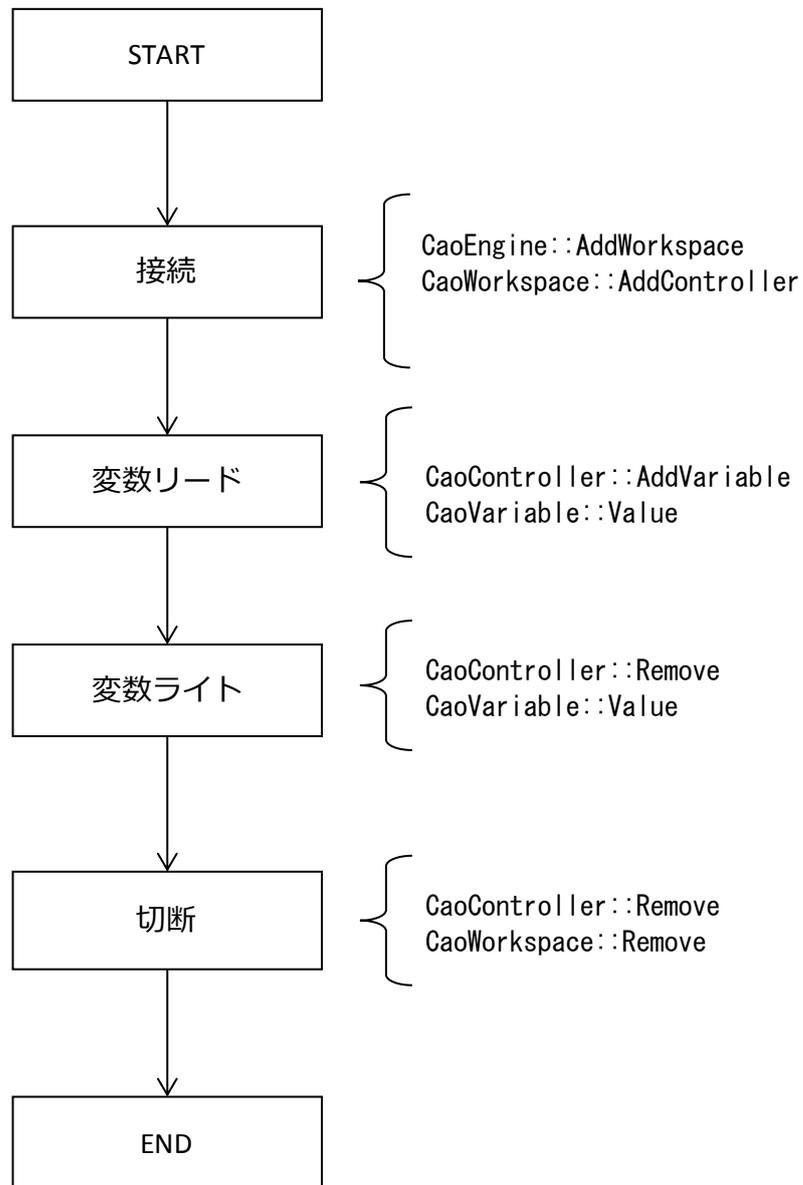


図 4-1 変数 (I) の値を取得/設定の流れ

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 **GetSetIntegerVariable.vb**

' オブジェクト

```
Dim engine As CaoEngine
Dim workspace As CaoWorkspace
Dim controller As CaoController
```

```
Private Sub Main()  
    ' 接続  
    Call Connect  
  
    ' 変数番号 10 にアクセスする CalVariable を CaoController に追加  
    Dim intVarI10 As CaoVariable  
    set intVarI10 = controller.AddVariable("I-10", "no=10, elem=1")  
  
    ' 変数番号 10 から値を取得する  
    Dim value As Integer  
    value = intVar_10.value  
  
    ' 変数番号 10 へ取得した値 + 100 の値を設定する  
    intVar_10.value = value + 100  
  
    ' CaoController から CaoVariable を削除  
    Call controller.Variables.Remove(intVar_10.Index)  
    ' CaoVariable の消去  
    Set intVar_10 = Nothing  
  
    ' 切断  
    Call Dicconnect  
End Sub  
  
' 接続メソッド  
Private Sub Connect()  
    ' CaoEngine オブジェクトの生成  
    Set engine = New CaoEngine  
    ' CaoWorkspace オブジェクトの生成  
    Set workspace = engine.AddWorkspace("NewWrks", "")  
    ' CaoController オブジェクトの生成  
    Set controller = workspace.AddController("Controller", _  
                                           "CaoProv. YASKAWA. MOTOCOMES", _  
                                           "", _  
                                           "conn=eth:192.168.10.10, cntltype=1")  
End Sub  
  
' 切断メソッド  
Private Sub Disconnect()  
    ' CaoWorkspace から CaoController を削除  
    Call workspace.Controllers.Remove(controller.Index)  
    ' CaoController の消去  
    Set controller = 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
End Sub

```

4.1.1.1. 接続

YRC 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。CaoWorkspaceオブジェクトは、CaoControllerオブジェクトをCaoWorkspacesから取得する場合には変数を用意する必要はありません。また変数にアクセスするためのCaoVariableオブジェクトも必要になります。以下にVB6でのコード例を示します。

```

Dim engine As CaoEngine      ' CaoEngineオブジェクト用の変数
Dim workspace As CaoWorkspace ' CaoWorkspaceオブジェクト用の変数
Dim controller As CaoController ' CaoController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", "")

```

- (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。YASKAWA MOTOCOMESプロバイダでは、接続先のIPアドレス及びロボットコントロールタイプをオプションで指定します。以下にコード例を示します。

```

' CaoController オブジェクトの生成
Set controller = workspace.AddController("Controller ", _
    "CaoProv. YASKAWA. MOTOCOMES", _
    "", _
    "conn=eth:192.168.10.10, cntltype=1")

```

4.1.1.2. 変数の取得/設定

変数の値を取得/設定するには、アクセスしたい変数の CaoVariable オブジェクトを生成し、CaoVariable オブジェクトの Value プロパティを参照/設定します。Value プロパティを参照/設定する場合は、Value プロパティに合わせた型変数を用意する必要があります。以下にコード例を示します。

```
' 変数番号10にアクセスするCaoVariableをCaoControllerに追加
Dim intVar_10 As CaoVariable
Set intVar_10 = controller.AddVariable("I-10", "no=10, elem=1")

' 変数番号10から値を取得する
Dim value As Integer
value = intVar_10.value

' 変数番号10へ取得した値 + 100の値を設定する
intVar_10.value = value + 100

' CaoControllerからCaoVariableを削除
Call controller.Variables.Remove(intVar_10.Index)
' CaoVariableの消去
Set intVar_10 = Nothing
```

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 controller = 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. MOTOCOMES プロバイダエラーコード

本プロバイダには、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 5-1 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 5-1 独自エラーコード表

エラー番号	説明
0x80110001	コントローラタイプオプションがありません。

また、本プロバイダは、API のエラーコードを「0x8010****」でマスクして返します。

API のエラーコード詳細は、MOTOCOMES マニュアルを参照して下さい。

付録A. API 対応表

本プロバイダの変数へ取得／設定時の使用 API を以下に示します。

変数名	取得時の使用 API	設定時の使用 API
@ALARM	ESGetAlarm	—
@ALARMEX	ESGetAlarmEx	—
@STATUS	ESGetStatus	—
@BDSP	—	ESBDSP
B<??>	ESGetVarDataMB	ESSetVarDataMB
I<??>	ESGetVarDataMI	ESSetVarDataMI
D<??>	ESGetVarDataMD	ESSetVarDataMD
R<??>	ESGetVarDataMR	ESSetVarDataMR
STR<??>	<ul style="list-style-type: none"> ■CntIType=1 ■CntIType=2 上記指定時, ESGetStrDataM ■CntIType=3 ■CntIType=4 ■CntIType=5 上記指定時, ESGetStrDataM2 	<ul style="list-style-type: none"> ■CntIType=1 ■CntIType=2 上記指定時, ESSetStrDataM ■CntIType=3 ■CntIType=4 ■CntIType=5 上記指定時, ESSetStrDataM2
POS<??>	ESGetPositionDataM	ESSetPositionDataM
BASEPOS<??>	ESGetBpexPositionDataM	ESSetBpexPositionDataM
EXTPOS<??>	ESGetBpexPositionDataM	ESSetBpexPositionDataM
IO<??>	ESReadIOM	ESWriteIOM
REG<??>	ESReadRegisterM	ESWriteRegisterM
JOBSTATUS<??>	ESGetJobStatus	—
ALARMHIST<??>	ESGetAlarmHist	—
ALARMHISTEX<??>	ESGetAlarmHistEx	—
POSITION<??>	ESGetPosition	—
DEVIATION<??>	ESGetDeviation	—
TORQUE<??>	ESGetTorque	—
MONITORINGTIME<??>	ESGetMonitoringTime	—
SYSTEMINFO<??>	ESGetSystemInfo	—