

TMF プロバイダ

ユニパルス製 トルクメータ・荷重指示計

ユーザーズ ガイド

Version 1.0.0

July 31, 2015

【備考】

【改版履歴】

日付	版数	内容
2015-07-031	1.0.0	初版.
2015-12-16		誤記修正

【対応機器】

機種	バージョン	注意事項
TM301		
F325		
F340A		
F350		
F372A		
F377A	Ver3.05	UTM II +F377A にて動作確認

【ご注意】

本プロバイダを使用する場合は別途“ユニパルス製トルクメータ・荷重指示計”ライセンスが必要です。

目次

1. はじめに	5
1.1. ライセンスの追加	5
2. プロバイダの概要	6
2.1. 概要	6
2.2. メソッド・プロパティ	7
2.2.1. CaoWorkspace::AddController メソッド	7
2.2.2. CaoController::Execute メソッド	8
2.2.3. CaoController::AddVariable メソッド	8
2.2.4. CaoController::get_VariableNames プロパティ	8
2.2.5. CaoVariable::get_Value プロパティ	9
2.3. 変数一覧	9
2.3.1. コントローラクラス	9
2.4. エラーコード	12
3. コマンドリファレンス	13
3.1. Execute コマンド	13
3.1.1. CaoController::Execute("Raw") コマンド	13
3.1.2. CaoController::Execute("DigitalZero") コマンド	13
3.1.3. CaoController::Execute("ChangeDisp") コマンド	14
3.1.4. CaoController::Execute("SetControl") コマンド	14
3.1.5. CaoController::Execute("ChangeWork") コマンド	14
3.1.6. CaoController::Execute("ChangeCH") コマンド	14
3.1.7. CaoController::Execute("HoldMode") コマンド	15
3.1.8. CaoController::Execute("ChangeTH") コマンド	15
3.1.9. CaoController::Execute("ChangeSection") コマンド	15
4. サンプルプログラム	16
4.1. VisualBasic Ver6.0	16
4.2. PacScript	16

1. はじめに

本書はユニパルス製トルクメータ・荷重指示計用の CAO プロバイダである、TM プロバイダのユーザーズガイドです。

TMF プロバイダは、インターフェースユニットに対してコマンドの送信とレスポンスの受信を行います。

1.1. ライセンスの追加

本プロバイダを使用可能にするにはORiN2 SDKをインストール後、別途「ユニパルス製トルクメータ・荷重指示計」ライセンスを入力する必要があります。

下記に「ユニパルス製トルクメータ・荷重指示計」ライセンスの追加手順を示します。

1. CaoConfigを起動し、[Cao Provider]タブを選択する
2. Provider Listから[UNIPULSE TMF CAO Provider]項目を選択する
3. License項目の[...]ボタンをクリックする
4. ORiN2 License Managerで[Add]ボタンをクリックする
5. 入手したライセンスキーを入力後、[OK]ボタンをクリックする
6. [Close] ボタンをクリックし、CaoConfigを終了する

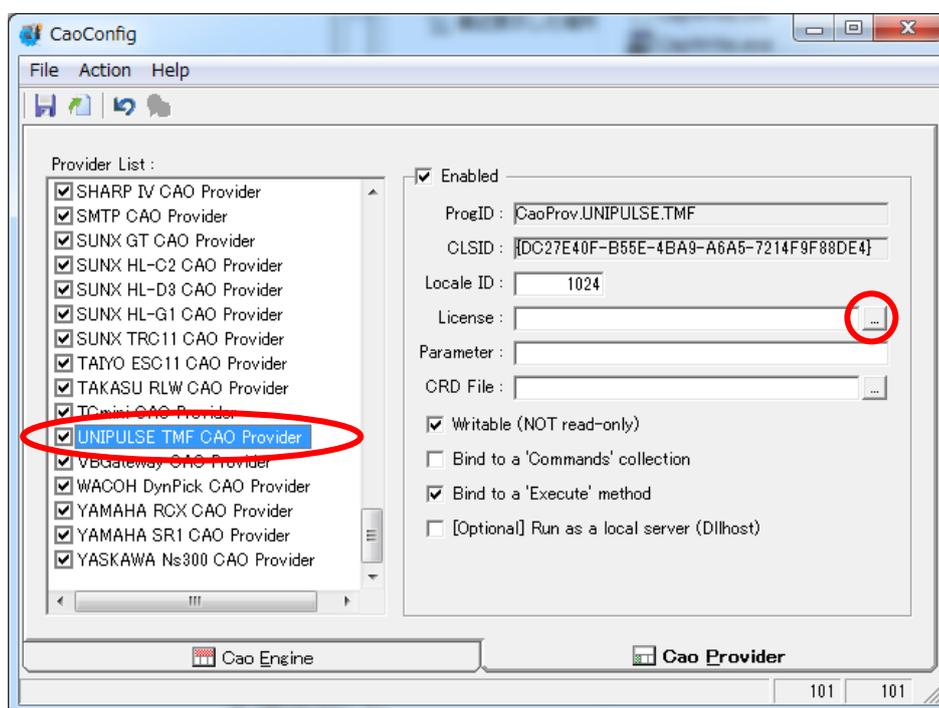


図 1-1 「ユニパルス製トルクメータ・荷重指示計」ライセンス追加

2. プロバイダの概要

2.1. 概要

TMF プロバイダは、コマンドの実行方法として `CaoController::Execute`、`CaoVariable` による 2 通りの方法を提供しています。

`CaoController::Execute` メソッド、及び `CaoVariable` は、実行時にコマンドを自動生成、送信を行います。またレスポンスの解析も行い、データを取得します。

TMF プロバイダのファイル形式は DLL(Dynamic Link Library)となっており、その詳細は表 2-1 のようになっています。

表 2-1 RLW プロバイダ

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

¹ 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

TMF プロバイダでは AddController 時に、通信用の接続パラメータを参照し、通信の接続を行います。このときオプションで通信形態、タイムアウト、接続オプション、デバイスタイプを指定します。

書式 AddController(<bstrCtrlName:VT_BSTR>,<bstrProvName:VT_BSTR>,
<bstrPcName:VT_BSTR > [,<bstrOption:VT_BSTR>])

bstrCtrlName : [in] コントローラ名 任意
 bstrProvName : [in] プロバイダ名 固定値 =” CaoProv.UNIPULSE.TMF”
 bstrPcName : [in] プロバイダの実行マシン名
 bstrOption : [in] オプション文字列

以下にオプション文字列に指定するリストを示します。

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

オプション	意味
Conn =<接続パラメータ>	必須. 通信形態とその接続パラメータを設定します. 詳細は 2.2.1.1 に示します.
Timeout[=<タイムアウト時間>]	送受信時のタイムアウト時間を指定します. (デフォルト: 500)
PacketOpt [=<デリミタ>]	デリミタを指定します. 0 : CR (デフォルト) 1 : CR+LF
Type=[<デバイスタイプ>]	センサアンプのタイプを指定します. 0 : TM301 (デフォルト) 1 : F325 2 : F340A 3 : F350 4 : F372A 5 : F377A

2.2.1.1. Conn パラメータ

以下に Parameter オプションの接続パラメータ文字列を示します。ここで角括弧(“[] ”)内は省略可能を示します。また、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ときのデフォルト値を示します。

“com:<COM Port>[:<BaudRate>[:<Parity>:<DataBits>:<StopBits>]]”

<COM Port> : COM ポート番号. ‘1’ -COM1, ‘2’ -COM2, ...
 <BaudRate> : 通信速度. 9600, 19200, 38400, 57600.
 <Parity> : パリティ. ‘N’ -NONE, ‘E’ -EVEN, ‘O’ -ODD
 <DataBits> : データビット数. ‘7’ -7bit, ‘8’ -8bit.
 <StopBits> : ストップビット数. ‘1’ -1bit, ‘2’ -2bit.

- (例 1) “com:1” 通信ポート COM1 (, 57600bps, None, 8bits, 1bit)
 (例 2) “com:2:9600” 通信ポート COM2, 9600bps (, None, 8bits, 1bit)
 (例 3) “com:3:38400:N:8:2” 通信ポート COM3, 38400bps, None, 8bits, 2bit

2.2.2. CaoController::Execute メソッド

コマンドの送受信を行います。第 1 引数にコマンド名、第 2 引数にコマンドのパラメータを指定します。各コマンドの詳細は 3 章コマンドリファレンスを参照してください。

書式 Execute (<bstrCommandName:VT_BSTR>,[<vntParam : VT_VARIANT>])

bstrCommandName: [in] コマンド名

vntParam : [in] パラメータ

2.2.3. CaoController::AddVariable メソッド

トルクセンサからのデータ読み込みを実行する CaoVariable オブジェクトを生成します。

変数名はトルクセンサのデータと対応します。指定できる変数名は 2.3.1 を参照して下さい。

書式 AddVariable(<bstrVariableName:VT_BSTR>,[<bstrOption: VT_BSTR >])

bstrVariableName : [in] 変数名

bstrOption : [in] オプション文字列 (未使用)

2.2.4. CaoController::get_VariableNames プロパティ

変数名リストを取得します。取得する変数名は 2.3.1 を参照して下さい。

2.2.5. GaoVariable::get_Value プロパティ

変数名で指定したデータを取得します。

2.3. 変数一覧

2.3.1. コントローラクラス

表 2-3 コントローラクラス 変数一覧

共通				
変数名	データ型	説明	属性	
			get	put
@Meker_Name	VT_STR	プロバイダ作成メーカー	○	-
@Version	VT_STR	プロバイダバージョン	○	-

TM301				
変数名	データ型	説明	属性	
			get	put
@Torque	VT_R4	トルク値(ホールド非連動)	○	-
@Hold1	VT_R4	トルク値(ホールド連動)	○	-
@RotSpeed	VT_R4	回転速度	○	-
@PowerValue	VT_R4	動力値	○	-
@Status1	VT_I4	ステータス 1(トルク)	○	-
@Status2	VT_I4	ステータス 2(トルク)	○	-
@Status3	VT_I4	ステータス 3(トルク)	○	-
@Status4	VT_I4	ステータス 4(トルク)	○	-
@Status5	VT_I4	ステータス 5(トルク)	○	-

F325				
変数名	データ型	説明	属性	
			get	put
@Hold1	VT_R4	指示値連動読出し	○	-
@Hold2	VT_R4	指示値非連動読出し	○	-
@Status1	VT_I4	ステータス 1	○	-
@Status2	VT_I4	ステータス 2	○	-
@Status3	VT_I4	ステータス 3	○	-

F340A				
変数名	データ型	説明	属性	
			get	put
@Hold1	VT_R4	指示値連動読出し	○	-
@Status1	VT_I4	ステータス 1	○	-

F350				
変数名	データ型	説明	属性	
			get	put
@Torque	VT_R4	指示値(リアルタイム演算値)	○	-
@Hold1	VT_R4	指示値(演算値)	○	-
@Hold2	VT_R4	指示値(CH1)	○	-
@Hold3	VT_R4	指示値(CH2)	○	-
@Status1	VT_I4	ステータス 1(演算値)	○	-
@Status2	VT_I4	ステータス 2(CH1 値, CH2 値)	○	-
@Status3	VT_I4	ステータス 3(CH1 値)	○	-
@Status4	VT_I4	ステータス 4(CH2 値)	○	-

F372A				
変数名	データ型	説明	属性	
			get	put
@Hold1	VT_R4	指示値読出し	○	-
@Status1	VT_I4	ステータス読出し	○	-
@Status2	VT_I4	ステータス読出し	○	-

F377A				
変数名	データ型	説明	属性	
			get	put
@Torque	VT_R4	リアルタイム値	○	-
@Hold1	VT_R4	ホールド A	○	-
@Hold2	VT_R4	ホールド B	○	-
@Status1	VT_I4	ステータス 1	○	-
@Status2	VT_I4	ステータス 2	○	-
@Status3	VT_I4	ステータス 3	○	-
@ Relative	VT_R4	相対値	○	
@ Difference	VT_R4	前差分値	○	
@WorkNo	VT_I4	ワーク番号	○	○

2.4. エラーコード

TMF プロバイダでは、以下の固有エラーコードが定義されています。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。

表 2-4 独自エラーコード一覧

エラー名	エラー番号	説明
E_RESPONSE_FAILED	0x80100001	無効なレスポンスを受信しました

3. コマンドリファレンス

本章では CaoController::Execute メソッドの各コマンドについて解説します。

表 3-1 CaoController::Execute コマンド一覧

対応機器	コマンド	機能
全機種	Raw	生データ送受信.
全機種	DigitalZero	デジタルゼロを設定・解除します.
F377A	ChangeDisp	画面表示を切り替えます
F377A	SetControl	制御信号を切り替えます
F377A	ChangeWork	計測ワークを切り替えます
F350	ChangeCH	ch を切換えます
F325,F340A	HoldMode	ホールド状態を切り替え, リセットを行います
TM301,F350	ChangeTH	T/H 信号を切り替えます
TM301,F350	ChangeSection	Section 信号を切り替えます

3.1. Execute コマンド

3.1.1. CaoController::Execute(“Raw”) コマンド

生データ送受信を行います。

書式 Raw (<bstrSendData>)

<bstrSendData> : [in] 送信文字列 (VT_BSTR)

戻り値 : [out] 受信文字列 (VT_BSTR)

引数で指定した文字列をセンサに送信します。受信したレスポンスを文字列で返します。
このとき送受信でコマンド及びレスポンスの内容は一切加工しません。

3.1.2. CaoController::Execute(“DigitalZero”) コマンド

デジタルゼロを設定, 解除します。

書式 DigitalZero (<iSwitch>)

<iSwitch> : [in] 切替えパラメータ (VT_I4)

0	デジタルゼロを解除します。
1	デジタルゼロを設定します。
2	[F350 のみ]デジタルゼロを解除します(両 ch 同時)。
3	[F350 のみ]デジタルゼロを設定します(両 ch 同時)。

戻り値 : なし

3.1.3. CaoController::Execute(“ChangeDisp”) コマンド

[F377A] アンプの画面表示を切替えます。

書式 ChangeDisp (<iSwitch>)

<iSwitch> : [in] 切替えパラメータ(VT_I4)

1	比較画面
2	ホールド画面
3	グラフ画面

戻り値 : なし

3.1.4. CaoController::Execute(“SetControl”) コマンド

[F377A] 制御信号を切り替えます。

書式 SetControl (<iCtrlSwitch>)

<iCtrlSwitch > : [in] 切替えパラメータ(VT_I4 | VT_ARRAY)

T/H , SECTION , GRAPH TRIG

0	OFF
1	ON

戻り値 : なし

3.1.5. CaoController::Execute(“ChangeWork”) コマンド

[F377A] 計測ワーク番号を切替えます。

書式 ChangeWork (< iWorkNo >)

< iWorkNo > : [in] 計測ワーク番号(VT_I4) (00 ~ 15)

戻り値 : なし

3.1.6. CaoController::Execute(“ChangeCH”) コマンド

[F350] ch を切替えます。

書式 ChangeCH (< iChNo >)

< iChNo > : [in] ch 番号(VT_I4) (1 ~ 2)

戻り値 : なし

3.1.7. CaoController::Execute(“HoldMode”) コマンド

[F325,F340A] ホールド状態を切り替え、リセットを行います。



HoldMode (<iMode>)

<iMode> : [in] 切替えパラメータ(VT_I4)

-1	ホールドリセットを行います。
0	[F350のみ]ホールドモードを解除します。
1	ホールドモードに切り替えます。

戻り値 : なし

3.1.8. CaoController::Execute(“ChangeTH”) コマンド

[TM301,F350] T/H 信号を切り替えます。



ChangeTH (<iSwitch>)

<iSwitch> : [in] 切替えパラメータ(VT_I4)

0	OFF
1	ON

戻り値 : なし

3.1.9. CaoController::Execute(“ChangeSection”) コマンド

[TM301,F350] Section 信号を切り替えます。



ChangeSection (<iSwitch>)

<iSwitch> : [in] 切替えパラメータ(VT_I4)

0	OFF
1	ON

戻り値 : なし

4. サンプルプログラム

センサのデータ取得、ピーク値リセットをするコードを示します。

4.1. VisualBasic Ver6.0

List 4-1 Sample.frm

```
Dim eng As CaoEngine
Dim ctrl As CaoController
Dim val As CaoVariable

Private Sub Form_Load()

    ' CAO エンジンの生成
    Set eng = New CaoEngine

    ' センサへの接続 (F377A)
    Set ctrl = eng.Workspaces(0).AddController("Sample", "CaoProv.UNIPULSE.TMF", "",
"Conn=com:1, Type=5")

    ' 変数オブジェクトの生成
    Set val = ctrl.AddVariable("@Torque")

End Sub

Private Sub Command1_Click()

    ' トルク値の取得
    Text1.Text = val.Value

End Sub

Private Sub Command2_Click()

    ' ピーク値リセット (SECTION 信号 OFF/ON)
    ctrl.Execute "SetControl", Array(0, 0, 0)
    ctrl.Execute "SetControl", Array(0, 1, 0)

End Sub
```

4.2. PacScript

List 4-2 Sample.pcs

```
Dim ctrl As Object
Dim val As Object

Sub Main()

    Dim cnt As Integer

    ' センサへの接続 (F377A)
    ctrl = cao.AddController("Sample", "CaoProv.UNIPULSE.TMF", "", "Conn=com:2, Type=5")

    ' 変数オブジェクトの生成
    val = ctrl.AddVariable("@Torque")

    ' ピーク値リセット
    ctrl.SetControl Array(0, 0, 0)
    ctrl.SetControl Array(0, 1, 0)
```

```
Do
    ' トルク値の取得
    F[0] = val.Value
Loop Until F[0] > F[1] ' トルク値が設定値を超えるまでループ

val = Nothing
ctrl.Variables.Clear
ctrl = Nothing

End Sub
```

