

KEBA AG  
**Active Contact Flange** プロバイダ

Version 1.0.0

ユーザーズ ガイド

July 18, 2012

備考:

**【改版履歴】**

バージョン	日付	内容
1.0.0	2012-7-18	初版

**【対応機器】**

機種	バージョン	注意事項
ACF 110-04		ACF TCP/IP イーサネット通信 I/F Ver 1040SV_03.00.00
ACF 110-10		
ACF 120-05		
ACF 120-10		

## 目次

1. はじめに .....	4
2. プロバイダの概要 .....	5
2.1. 概要 .....	5
2.2. メソッド・プロパティ .....	6
2.2.1. CaoWorkspace::AddController メソッド .....	6
2.2.2. CaoController::Execute メソッド .....	6
3. コマンドリファレンス .....	8
3.1. コントローラクラス .....	8
3.1.1. CaoController::Execute("Send") コマンド .....	8
4. サンプルプログラム .....	10

## 1. はじめに

本ドキュメントでは KEBA Active Contact Flange (ACF) デバイスを簡単に制御できる KEBA ACF プロバイダに関して記載しています。KEBA ACF プロバイダは Windows 搭載のパソコンの汎用言語、あるいは DENSO の RC8 ロボットコントローラの PacScript 言語から直接使用することができます。

ACF デバイスとの通信はイーサネット上の TCP/IP 通信で行われます。

本プロバイダはコマンド待ちをするサーバーである ACF デバイスに対してクライアントとして振る舞います。

## 2. プロバイダの概要

### 2.1. 概要

本プロバイダである ACF プロバイダではイーサネット接続 (TCP/IP) に対応した KEBA Active Contact Flange (ACF) を制御するための機能を提供しています。この機能は CaoController::Execute メソッドで実装されているただ一つのコマンドである Send コマンドに必要なパラメータを指定し、実行、その戻り値である現在値を使うことにより実現できます。

本プロバイダは ACF の TCP/IP 通信 I/F バージョン 1040SV\_03.00.00 に対応しています。

表 2-1 ACF プロバイダ

ファイル名	CaoProvACF.dll
ProgID	CaoProv.KEBA.ACF
レジストリ登録	regsvr32 CaoProvACF.dll
レジストリ登録の抹消	regsvr32 /u CaoProvACF.dll

## 2.2. メソッド・プロパティ

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

ACF プロバイダでは AddController 時にデバイスに接続するパラメータを次の書式で指定します。

#### 書式

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

bstrCtrlName : [in] コントローラ名. 任意文字列を指定.

bstrProvName : [in] プロバイダProgID. "CaoProv.KEBA.ACF"の固定文字列を指定.

bstrPcName : [in] 実行PC名. 通常ローカル接続の空白文字列 ( "" ) を指定.

bstrOption : [in] オプション文字列

以下に<bstrOption>オプション文字列に指定するオプションリストを示す.

オプション	意味
Conn=< 接続パラメータ >	接続のためのパラメータを指定
ID=< ID 情報 >	ACF 型式とバージョンの情報を指定

#### Conn オプション

イーサネット接続時の接続パラメータの書式は次の通りです.

```
"conn=eth:<IP Address>[:<Port No>]"
```

<IP Address> : IP アドレスを指定. 必須.

例: "conn=eth:192.168.0.1"

<Port No> : ポート番号を指定.

例: "conn=eth:192.168.0.1:7070"

(ACF のポート番号は常に 7070)

#### ID オプション

<ID 情報> : ACF 型式とバージョンの情報を指定.

例: ID=104003000

### 2.2.2. CaoController::Execute メソッド

Execute メソッドの引数は, 一番目の引数にコマンド名を文字列 (VT\_BSTR), 二番目の引数のコマンド名に対応するパラメータの意味で VARIANT 型 (VT\_VARIANT) に必要な情報を格納して渡します. 各コマンドの詳細に関しては Chapter 3 コマンドリファレンスを参照して下さい.



Execute ( <bstrCommandName:VT\_BSTR>, [<vntParam: VT\_VARIANT]> )

bstrCommandName : [in] コマンド名. (VT\_BSTR)

vntParam : [in] パラメータ. (VT\_VARIANT)

戻り値 : [out] コマンドに依存した値が返ります. (VT\_VARIANT)

## 3. コマンドリファレンス

### 3.1. コントローラクラス

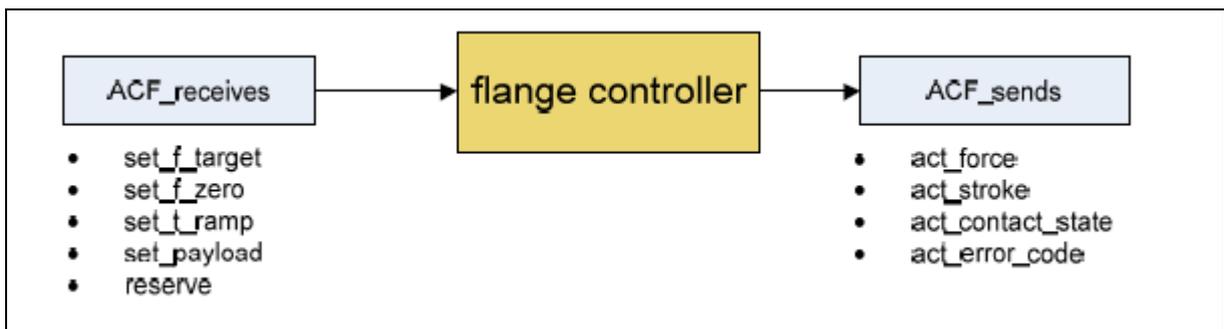
表 3-1 CaoController::Execute のコマンド一覧

コマンド名	機能	ページ
Send	ACF に新しい値を送信します	8

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

**書式** Send ( <set\_f\_target>, <set\_f\_zero>, <set\_t\_ramp>, <set\_payload> )

戻り値: 単精度実数の配列[要素数=4] (VT\_R4|VT\_ARRAY[4])

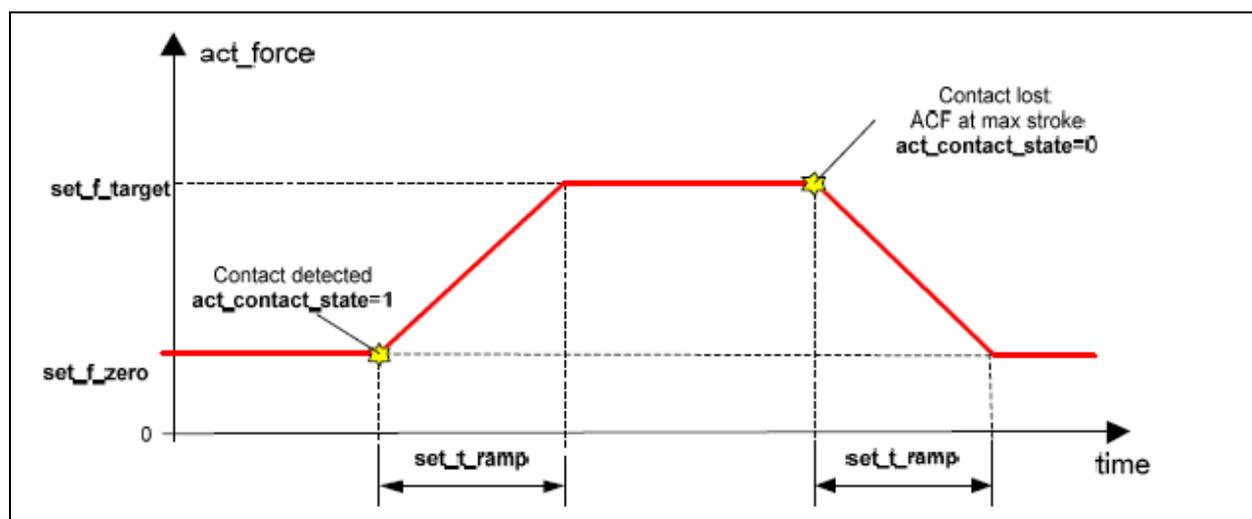


<引数 (VT\_R4|VT\_ARRAY[4])>

- set\_f\_target > : [in] 目標の力量 (単精度実数 VT\_R4)  
単位: ニュートン N
- <set\_f\_zero> : [in] 接触と判定する最少の力量 (単精度実数 VT\_R4)  
単位: ニュートン N
- <set\_t\_ramp> : [in] set\_f\_zero から set\_f\_target に到達するまでの時間  
(単精度実数 VT\_R4)  
単位: 秒 s  
値の範囲: 0 から 10
- <set\_payload> : [in] ACF デバイスのツール重量 (単精度実数 VT\_R4)  
単位: キログラム Kg  
値の範囲: ACF 110 = max. 10, ACF 120 = max. 50

<戻り値 (VT\_R4|VT\_ARRAY[4])>

- [0]:<act\_force> : [out] ACF デバイスの現在適応中の力量
- [1]:<act\_stroke> : [out] ACF デバイスの現在のストローク
- [2]:<act\_contact\_state> : [out] ACF デバイスの接触検出状態の値  
 0 未検出  
 1 検出
- [3]:<act\_error\_code> : [out] エラーコード  
 各ビットの意味:  
 0 valve pressure error  
 1 position sensor error  
 2 license error  
 3 set\_f\_target cannot be reached  
 4 warning: set\_f\_zero set to 0  
 (set\_f\_target and set\_f\_zero must have the same sign)  
 5 one or more input values out of range



#### 使用例

```
float[] input = new float[] { 1.2f, 3.6f, 1.5f, 1.8f };
CaoController.Execute("Send", input);
```

## 4. サンプルプログラム

### List 4-1 Example.cs

```
public class Program
{
    static void Main(string[] args)
    {
        CaoEngine _engine;
        CaoController _controller;
        float[] sendValues;
        float[] receivedValues;

        _engine = new CaoEngine();
        _controller = _engine.Workspaces.Item(0).AddController("KEBA",
"CaoProv.KEBA.ACF", "", "Conn=eth:192.168.0.1:7070, ID=1040030300");
        sendValues = new float[] { 1.6f, 5.2f, 2.2f, 1.3f };
        receivedValues = new float[4];

        for (int i = 0; i < 100; i++)
        {
            receivedValues = (float[])_controller.Execute("Send", sendValues);

            Console.WriteLine(string.Format(
                "Result: {0}, {1}, {2}, {3}",
                receivedValues[0],
                receivedValues[1],
                receivedValues[2],
                receivedValues[3]));

            Thread.Sleep(20);
        }
    }
}
```

**List 4-2 PacScriptExample.pcs**

```
!TITLE " PacScriptExample.pcs "  
#define VT_R4          4  
Sub Main  
    TakeArm Keep = 0  
    Dim SendValues As Variant  
    Dim ReceiveValues As Variant  
    Dim KebaController As Object  
  
    ' Connect to KEBA ACF  
    KebaController = CAO.AddController("KEBA", "CaoProv.KEBA.ACF", "",  
"Conn=eth:127.0.0.1:7070, ID=1040030300")  
  
    SendValues = CreateArray(4, VT_R4)  
    SendValues(0) = 1.2  
    SendValues(1) = 1.4  
    SendValues(2) = 1.5  
    SendValues(3) = 1.7  
  
    ReceiveValues = CreateArray(4, VT_R4)  
    ReceiveValues(0) = 0  
    ReceiveValues(1) = 0  
    ReceiveValues(2) = 0  
    ReceiveValues(3) = 0  
  
    for I1 = 0 to 5  
        DELAY 20  
        ReceiveValues = KebaController.Execute("Send", SendValues)  
        Debug.Print "Received Values"  
        Debug.Print ReceiveValues(0)  
        Debug.Print ReceiveValues(1)  
        Debug.Print ReceiveValues(2)  
        Debug.Print ReceiveValues(3)  
    next  
End Sub
```