

GINGA プロバイダ LinX 画像入力ボード

Version 1.0.1

ユーザーズ ガイド

July 17, 2012

【備考】

【改版履歴】

バージョン	日付	内容
1.0.0.0	2009-11-13	初版.
1.0.1.0	2010-20-11	Get_VariableNames 追記. 誤植修正. エラーコード追加
1.0.1.1	2011-03-11	プロバイダ登録ツールに関する追記
1.0.1	2012-07-17	ドキュメントのバージョンルールを変更

【対応機器】

機種	バージョン	注意事項
GINGA-DG/ CLe		RGB 24bit / RAW 12bit / RAW 10bit / RAW 8bit カメラ対応 1台接続のみ (動作確認は東芝テリー製 CSCU30CC18 のみ)

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::get_VariableNames プロパティ	6
2.2.3. CaoController::AddVariable メソッド	6
2.2.4. CaoVariable::get_Value プロパティ	6
2.3. 変数一覧	7
2.3.1. コントローラクラス	7
2.4. エラーコード	7
3. サンプルプログラム	8

1. はじめに

本書は LinX 社製銀河ボード用の CAO プロバイダである, GINGA プロバイダのユーザーズガイドです.

GINGA プロバイダはカメラリンクケーブルによって接続されたカメラから画像の取得を行います. このプロバイダを使用するためには, LinX 社から提供されている GINGA-API をインストールしてください. カメラの設定は LinX 社から提供されている各カメラ用の設定ファイルを準備し, GingaSysReg.exe 等のツールを用いて適切な設定を行ってください. ドライバインストール後にプロバイダをレジストリ登録する必要があります. レジストリ登録の方法は表 2-1 を参照してください.

2. プロバイダの概要

2.1. 概要

GINGA プロバイダは, GINGA-API を使用してカメラデバイスからの画像を取得します. 取得される画像データの形式は, 銀河ボードの設定に依存します.

GINGA プロバイダは ORiN SDK で ORiN をインストールした際にレジストリ登録されていません. 以下のコマンドをコマンドプロンプトで使用して, レジストリ登録を行ってから使用してください.

表 2-1 GINGA プロバイダ

ファイル名	GaoProvGINGA.dll
ProgID	GaoProv.LinX.GINGA
レジストリ登録 ¹	regsvr32 GaoProvGINGA.dll
レジストリ登録の抹消	regsvr32 /u GaoProvGINGA.dll

¹ プロバイダの登録は regsvr32.exe または RegCOM.exe ([スタート]→[ORiN2]→[Tools])で実行できます. GINGA-API をインストールしていないと, GINGA プロバイダの登録はできません.

2.2. メソッド・プロパティ

2.2.1. CaoWorkspace::AddController メソッド

GINGA プロバイダでは AddController 時にボードの検索を行い、ボードの接続処理を行います。

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

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

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

オプション	意味
ErrorDialog=True/False	GINGA-API のエラーダイアログの表示を有効にします. デフォルト=False (AddController 時のエラーダイアログは必ず表示します.)

2.2.2. CaoController::get_VariableNames プロパティ

AddVariable メソッドで指定できるシステム変数名の一覧を取得します。

2.2.3. CaoController::AddVariable メソッド

カメラデバイス及びボード・カメラ情報の変数オブジェクトを作成します。変数名には表 2-3 に示す変数名が使用できます。

書式 AddVariable(<bstrCtrlName:BSTR> [,<bstrOption:BSTR>])

bstrCtrlName : [in] 変数名
 bstrOption : [in] オプション文字列

2.2.4. CaoVariable::get_Value プロパティ

指定した変数に対応した値を取得します。詳細については表 2-3 を参照してください。

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 システム変数一覧

変数名	データ型	説明	属性											
			get	put										
@RAWImage	VT_UI1 ARRAY 又は VT_I2 ARRAY	RAW データ 取得するデータ型は, RAW データのビットによって以下ようになります. <table border="1" data-bbox="603 689 1023 947"> <thead> <tr> <th>ビット数</th> <th>データ型</th> </tr> </thead> <tbody> <tr> <td>8</td> <td>VT_UI1 ARRAY</td> </tr> <tr> <td>10</td> <td>VT_I2 ARRAY</td> </tr> <tr> <td>12</td> <td>VT_I2 ARRAY</td> </tr> <tr> <td>24</td> <td>VT_UI1 ARRAY</td> </tr> </tbody> </table>	ビット数	データ型	8	VT_UI1 ARRAY	10	VT_I2 ARRAY	12	VT_I2 ARRAY	24	VT_UI1 ARRAY	○	-
ビット数	データ型													
8	VT_UI1 ARRAY													
10	VT_I2 ARRAY													
12	VT_I2 ARRAY													
24	VT_UI1 ARRAY													
@Width	VT_I4	画像の横幅(pixel)	○	-										
@Height	VT_I4	画像の高さ(pixel)	○	-										
@BytesPerpix	VT_I4	1 ピクセル辺りのバイト数	○	-										
@FrameSize	VT_I4	画像データの大きさ(色情報のみ)	○	-										
@FrameWidth	VT_I4	画像の 1 ラインのサイズ	○	-										
@BITMAP	VT_UI1 ARRAY	BITMAP 画像データ (RGB 24bit の時のみ getvalue 可能)	○	-										
@PixDepth	VT_I4	画像の bit 数	○	-										

2.4. エラーコード

GINGA プロバイダでは, 固有のエラーコードはありません. 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.

3. サンプルプログラム

サンプルプログラムとして、VB6.0にて画像を取得し、PictureBoxに表示するコードを示します。PictureBoxは表示画像の拡大縮小に対応していませんので、表示する領域のサイズに注意してください。

List 3-1 frmMain.frm

```
Option Explicit

' ビットマップ API
Private Type BITMAPINFO ' BITMAPINFOHEADER (本当の BITMAPINFO ではない)
    data(1064) As Byte
End Type

Private Declare Function GetDIBits Lib "gdi32" (ByVal ahDC As Long, ByVal hBitmap As Long, ByVal nStartScan As Long, ByVal nNumScans As Long, lpBits As Any, lpBI As BITMAPINFO, ByVal wUsage As Long) As Long
Private Declare Function SetDIBits Lib "gdi32" (ByVal hdc As Long, ByVal hBitmap As Long, ByVal nStartScan As Long, ByVal nNumScans As Long, lpBits As Any, lpBI As BITMAPINFO, ByVal wUsage As Long) As Long
Private Declare Sub RtlMoveMemory Lib "kernel32" (Destination As Any, Source As Any, ByVal Length As Long)

Private m_Eng As CaoEngine
Private m_Ctrl As CaoController
Private m_Img As CaoVariable

Private Sub Form_Load()
    ' CAO エンジンの生成
    Set m_Eng = New CaoEngine
    ' ボードへの接続
    Set m_Ctrl = m_Eng.Workspaces(0).AddController("GINGA", "CaoProv.LinX.GINGA", "", "")
    Set m_Img = m_Ctrl.AddVariable("@BITMAP", "")
End Sub

Private Sub Timer1_Timer()
    Dim i As Long
    Dim bmp() As Byte
    Dim lAddr As Long, lWidth As Long, lHeight As Long
    Dim udtBM As BITMAPINFO

    ' 画像の取得
    bmp = m_Img.Value

    ' BMPINFO をコピー
    RtlMoveMemory lAddr, bmp(10), 4
    RtlMoveMemory lWidth, bmp(18), 4
    RtlMoveMemory lHeight, bmp(22), 4

    ' 画像の高さが-の場合がある
    If (lHeight < 0) Then
        lHeight = -lHeight
    End If

    For i = 0 To lAddr - 14 - 1
        udtBM.data(i) = bmp(i + 14)
    Next

    ' 画像表示
    With picCamera
        .Cls
    End With
End Sub
```

```
        SetDIBits .hdc, .Image, 0&, lHeight, bmp(lAddr), udtBM, 0&  
        .Refresh  
    End With  
End Sub
```

