

オプテックスエフエー株式会社  
OPPD プロバイダ

Version 1.0.0

ユーザーズ ガイド

January 31, 2022

備考：

**【改版履歴】**

バージョン	日付	内容
1.0.0	2017-04-25	初版.
	2022-01-31	使用例の誤記訂正 WriteDeviceLock, ResetLightingSequence, ReadEmissionTimeUnit

**【対応機器】**

機種	バージョン	注意事項
OPPD 30E	-	

## 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 .....	5
2.2.1. CaoWorkspace::AddController メソッド .....	5
2.2.2. CaoController::AddVariable メソッド .....	6
2.2.3. CaoController::get_VariableNames メソッド .....	6
2.2.4. CaoController::Execute メソッド .....	7
2.2.5. CaoVariable::get_Value メソッド .....	7
2.2.6. CaoVariable::put_Value メソッド .....	7
2.3. Execute コマンド一覧.....	8
2.4. 変数一覧 .....	28
3. サンプルプログラム.....	29

## 1. はじめに

本書は、オプテックス・エフエー株式会社製 イーサネット対応 LED 照明コントローラ (OPPD 30E) 用 ORiN プロバイダ (OPTEX-FA OPPD プロバイダ) のユーザーズガイドです。

本書は、この OPTEX-FA OPPD プロバイダの機能と実装されているメソッドについて説明します。

## 2. プロバイダの概要

### 2.1. 概要

OPTEX-FA OPPD プロバイダは、デバイスから Telnet 通信を利用してデータを取得します。本プロバイダのファイル形式は DLL (Dynamic Link Library) となっており、その詳細を表 2.1 に示します。

表 2.1 OPTEX-FA OPPD プロバイダ

ファイル名	CaoProvOPTEX-FAOPPD.dll
ProgID	CaoProv. OPTEX-FA.OPPD
レジストリ登録	regsvr32 CaoProvOPTEX-FAOPPD.dll
レジストリ登録の抹消	regsvr32 /u CaoProvOPTEX-FAOPPD.dll

### 2.2. メソッド・プロパティ

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

Controller オブジェクトの生成時に必要なオプションを指定します。

以下に、AddController の仕様を示します。

#### 書式

##### AddController

```
(
    "<コントローラ名>",           // コントローラ名(任意)
    "CaoProv. OPTEX-FA.OPPD",     // 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
)
```

以下にオプション文字列に指定する文字列を示します。

表 2.2 CaoWorkspace::AddController のオプション文字列

オプション	必須	説明	値範囲	デフォルト値
CONN=<接続オプション> ※1	○	通信パラメータを指定します。指定方法については、2.2.1.1 を参照してください。	-----	-----
TIMEOUT=<応答待機時間>	--	応答待機時間を指定します (ms)。指定されない場合、指定値が範囲外の場合はデフォルト値となります。	0 - 65535	500

※1 CONN オプションについては、ETH 接続オプション、TCP 接続オプション のいずれかが指定できます。

#### 使用例

```
CONN=ETH:192.168.10.1
```

### 2.2.1.1. 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。

**Ethernet** 下線はデフォルト値を示します。

```
“Conn=ETH:<IP>[:<Port>[:<Local IP>:<Local Port>]]”
```

<IP> : 接続先 IP アドレス。  
<Port> : 接続先ポート。 (61440)  
<Local IP> : ローカル IP アドレス。  
<Local Port> : ローカルポート。

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

CaoController から CaoVariable オブジェクトを生成します。変数名には表 2.4 の変数名のみ使用できます。

以下に、AddVariable の仕様を示します。

**書式**

**AddVariable**

```
(  
    “<変数名>”, // 変数名  
    “”, // オプション文字列  
)
```

### 2.2.3. CaoController::get\_VariableNames メソッド

表 2.4 の変数名リストを取得します。

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

Execute メソッドを使用することでデバイスのデータを取得/設定することができます。  
詳細は、表 2.3Execute コマンド一覧を参照してください。

##### 書式

```
Execute  
(  
    <bstrCommand:BSTR>           // [in] コマンド名  
    [, <vntParam:VARIANT>]       // [in] パラメータ  
    [, <pVal:VARIANT>]])         // [in] 実行結果  
)
```

#### 2.2.5. CaoVariable::get\_Value メソッド

指定した変数名によって、デバイスからデータを取得します。  
詳細は表 2.4 コントローラクラス変数一覧を参照して下さい。

#### 2.2.6. CaoVariable::put\_Value メソッド

指定した変数名によって、デバイスにデータを設定します。  
詳細は表 2.4 コントローラクラス変数一覧を参照してください。

## 2.3. Execute コマンド一覧

表 2.3 にコントローラクラスで使用できる Execute コマンド一覧を記述します。

表 2.3 Execute コマンド一覧

コマンド名	説明	詳細
WriteOnOffForced	ランプを強制的に点灯または消灯させます。設定などは無視されます。	<a href="#">WriteOnOffForced</a>
Light	現在のシーケンス番号で、指定された点灯制御入力を選択しているランプを点灯値通りに点灯させます。	<a href="#">Light</a>
WriteDeviceLock	デバイスがツマミによる操作を受け付けるか否かを書き込みます。	<a href="#">WriteDeviceLock</a>
ResetLightingSequence	点灯制御シーケンスの内部状態をリセットします。	<a href="#">ResetLightingSequence</a>
WriteCommonConfiguration	ランプ 1, ランプ 2 共通の設定を書き込みます。ランプ 1, ランプ 2 共通の設定 (PWM 周波数, 点灯制御入力フィルタ時間, 点灯制御シーケンス回数) を書き込みます。	<a href="#">WriteCommonConfiguration</a>
ReadCommonConfiguration	ランプ 1, ランプ 2 共通の設定を読み出します。	<a href="#">ReadCommonConfiguration</a>
WriteLightingDelayTime	ランプ毎の点灯遅延時間を書き込みます。	<a href="#">WriteLightingDelayTime</a>
ReadLightingDelayTime	ランプ毎の点灯遅延時間を読み出します。	<a href="#">ReadLightingDelayTime</a>
WriteEmissionTimeUnit	ランプ毎の発光幅単位を書き込みます。	<a href="#">WriteEmissionTimeUnit</a>
ReadEmissionTimeUnit	ランプ毎の発光幅単位を読み出します。	<a href="#">ReadEmissionTimeUnit</a>
WriteMonitorAlarmThreshold	ランプ毎のモニタ輝度アラームの閾値を書き込みます。	<a href="#">WriteMonitorAlarmThreshold</a>
ReadMonitorAlarmThreshold	ランプ毎のモニタ輝度アラームの閾値を読み出します。	<a href="#">ReadMonitorAlarmThreshold</a>
WriteFeedbackSetting	ランプ毎のフィードバック機能の設定を書き込みます。	<a href="#">WriteFeedbackSetting</a>
ReadFeedbackSetting	ランプ毎のフィードバック機能の設定を読みだします。	<a href="#">ReadFeedbackSetting</a>
WriteLightingValueSetting	シーケンス毎の点灯値の設定 (調光値, 発行幅) を行います。	<a href="#">WriteLightingValueSetting</a>
ReadLightingValueSetting	シーケンス毎の点灯値の設定を読み出します。	<a href="#">ReadLightingValueSetting</a>
WriteLightingControlInput	シーケンス毎の点灯制御入力の設定を行います。	<a href="#">WriteLightingControlInput</a>
ReadLightingControlInput	シーケンス毎の点灯制御入力の設定を読み出します。	<a href="#">ReadLightingControlInput</a>
ReadCurrentData	ランプ毎の現在値 (モニタ値, 絶対輝度, 内部温度) を読み出します。	<a href="#">ReadCurrentData</a>
ReadCorrectionDimmingValue	簡易フィードバックによる補正後の調光値を読み出します。	<a href="#">ReadCorrectionDimmingValue</a>
GetTimeout	設定されているタイムアウト時間を取得します。	-----
SetTimeout	タイムアウト時間を設定します。	-----

## WriteOnOffForced

ランプを強制的に点灯または消灯させるコマンドです。設定などは無視されます。

項目	型説明		備考
vntParam	VT_ARRAY   VT_I2		書き込み保存フラグを指定しない場合は、VT_I2 型でも指定することができます。
	0	ランプの消灯/点灯を指定します。 0: 消灯 1: 点灯	
	1	書き込み保存フラグ 0: 書き込み値を保存しない(デフォルト) 1: 書き込み値を保存する	
pVal	なし		

### 使用例

```
CaoController.Execute("WriteOnOffForced", Array(1, 0))
```

## Light

現在のシーケンス番号で、指定された点灯制御入力を選択しているランプを点灯値通りに点灯させるコマンドです。

本メソッドを実行すると、指定したランプのシーケンス番号が1つ進みます。

ランプの点灯制御入力選択を変更したい場合は **WriteLightingControlInput** コマンドを、ランプの点灯値を変更する場合は **WriteLightingValueSetting** コマンドを実行してください。

項目	型説明	
vntParam	VT_UI2	点灯させるランプ番号を指定します。 1: ランプ1 2: ランプ2
pVal	なし	

### 使用例

```
CaoController.Execute("Light", 1)
```

## WriteDeviceLock

デバイスがツマミによる操作を受け付けるか否かを書き込むコマンドです。

項目	型説明	備考
vntParam	VT_ARRAY   VT_I2	書き込み保存フラグを指定しない場合は、VT_I2 型でも指定することができます。
	0 ランプの消灯/点灯を指定します。 0: 0: 1: 点灯	
	1 書き込み保存フラグ 0: 書き込み値を保存しない(デフォルト) 1: 書き込み値を保存する	
pVal	なし	

### 使用例

```
CaoController.Execute("WriteDeviceLock", 1)
```

## ResetLightingSequence

点灯制御シーケンスの内部状態をリセットさせるコマンドです。

項目	型説明
vntParam	なし
pVal	なし

### 使用例

```
CaoController.Execute("ResetLightingSequence")
```

## WriteCommonConfiguration

ランプ 1, ランプ 2 共通の設定を書き込みます。ランプ 1, ランプ 2 共通の設定 (PWM 周波数, 点灯制御入力フィルタ時間, 点灯制御シーケンス回数) を書き込むコマンドです。

項目	型説明		備考	
vntParam	VT_ARRAY   VT_VARIANT			
	0	VT_ARRAY   VT_I2		
	0.0	VT_I2	PWM 周波数を指定します。 0: 50 kHz 1: 100 kHz 2: 99 kHz 3: 98 kHz 4: 97 kHz	
	0.1	VT_I2	点灯制御入力フィルタ時間を指定します。 0: 1 $\mu$ s 1: 6 $\mu$ s 2: 25 $\mu$ s 3: 100 $\mu$ s	
	0.2	VT_I2	点灯制御シーケンス回数を指定します。 0: 1 回 1: 2 回 2: 3 回 3: 4 回	
1	VT_I2	書き込み保存フラグ 0: 書き込み値を保存しない(デフォルト) 1: 書き込み値を保存する	書き込み保存フラグは省略することができます。	
pVal	なし			

### 使用例

```
GaoController.Execute("WriteCommonConfiguration", Array(0, 0, 3))
```

## ReadCommonConfiguration

ランプ 1, ランプ 2 共通の設定 (PWM 周波数, 点灯制御入力フィルタ時間, 点灯制御シーケンス回数) を読み出すコマンドです。

項目	型説明		
vntParam	なし		
pVal	VT_ARRAY   VT_VARIANT		
	0	VT_ARRAY   VT_I2	
	0.0	VT_I2	PWM 周波数を指定します。 0: 50 kHz 1: 100 kHz 2: 99 kHz 3: 98 kHz 4: 97 kHz
	0.1	VT_I2	点灯制御入力フィルタ時間を指定します。 0: 1 $\mu$ s 1: 6 $\mu$ s 2: 25 $\mu$ s 3: 100 $\mu$ s
0.2	VT_I2	点灯制御シーケンス回数を指定します。 0: 1 回 1: 2 回 2: 3 回 3: 4 回	

### 使用例

```
CaoController.Execute("ReadCommonConfiguration")
```

## WriteLightingDelayTime

ランプ毎の点灯遅延時間を書き込むコマンドです。設定したいランプ番号と設定値をセットにして値を設定して下さい。

項目	型説明	備考	
vntParam	VT_ARRAY   VT_VARIANT	i の範囲は 0 ~ 1 です。	
	0 VT_ARRAY   VT_VARIANT	範囲を超えた部分に関しては無視されます。	
	0. i VT_ARRAY   VT_I2		
	0. i. 0 VT_I2	設定ランプを指定します。 1: ランプ 1 2: ランプ 2	
	0. i. 1 VT_I2	点灯遅延時間を指定します。単位は、0. i. 2 で指定する単位となります。 0 ~ 999	
0. i. 2 VT_I2	点灯遅延時間単位を指定します。 0: 10 $\mu$ s 1: 1 ms		
1	VT_I2	書き込み保存フラグ 0: 書き込み値を保存しない(デフォルト) 1: 書き込み値を保存する	書き込み保存フラグは省略することができます。
pVal	なし		

### 使用例

```
CaoController.Execute("WriteLightingDelayTime", Array(Array(Array(1, 0, 0), Array(2, 10, 0))))
```

## ReadLightingDelayTime

ランプ毎の点灯遅延時間を読み出すコマンドです。読出したいランプ番号を指定してください。

項目	型説明		備考	
vntParam	VT_I2	読み出すランプを指定します。 1: ランプ 1 2: ランプ 2 3: 全てのランプ		
pVal	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1 です。	
	0	VT_ARRAY   VT_VARIANT		
	0. i	VT_ARRAY   VT_I2		
	0. i. 0	VT_I2		ランプ (i+1) を取得します。 1: ランプ 1 2: ランプ 2
	0. i. 1	VT_I2		点灯遅延時間を取得します。単位は、0. i. 2 で取得した単位となります。 0 ~ 999
0. i. 2	VT_I2	点灯遅延時間単位を取得します。 0: 10 $\mu$ s 1: 1 ms		

### 使用例

```
GaoController.Execute("ReadLightingDelayTime", 3)
```

## WriteEmissionTimeUnit

ランプ毎の発光幅単位を書き込むコマンドです。設定したいランプ番号と設定値をセットにして値を設定して下さい。

項目	型説明	備考
vntParam	VT_ARRAY   VT_VARIANT	i の範囲は 0 ~ 1 です。
	0 VT_ARRAY   VT_VARIANT	範囲を超えた部分に関しては無視されます。
	0. i. 0 VT_ARRAY   VT_I2	設定ランプを指定します。 1: ランプ 1 2: ランプ 2
	0. i. 1 VT_I2	発行幅単位を指定します。 1: 10 $\mu$ s 2: 1 ms
	1 VT_I2	書き込み保存フラグ 0: 書き込み値を保存しない(デフォルト) 1: 書き込み値を保存する
pVal	なし	

### 使用例

```
GaoController.Execute("WriteEmissionTimeUnit", Array(Array(Array(1, 1), Array(2, 1)), 0))
```

## ReadEmissionTimeUnit

ランプ毎の発光幅単位を読み出すコマンドです。読出したいランプ番号を指定してください。

項目	型説明		備考	
vntParam	VT_I2	読み出すランプを指定します。 1: ランプ 1 2: ランプ 2 3: 全てのランプ		
pVal	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1 です。	
	0	VT_ARRAY   VT_VARIANT		
	0. i	VT_ARRAY   VT_I2		
	0. i. 0	VT_I2		ランプ (i+1) を取得します。 1: ランプ 1 2: ランプ 2
	0. i. 1	VT_I2		発行幅単位を取得します。 1: 10 $\mu$ s 2: 1 ms

### 使用例

```
GaoController.Execute("ReadEmissionTimeUnit", 3)
```

**WriteMonitorAlarmThreshold**

ランプ毎のモニタ輝度アラームの閾値を書き込むコマンドです。設定したいランプ番号と設定値をセットにして値を設定して下さい。

項目	型説明	備考	
vntParam	VT_ARRAY   VT_VARIANT	i の範囲は 0 ~ 1 です。 範囲を超えた部分に関しては無視されます。	
	0 VT_ARRAY   VT_VARIANT		
	0. i VT_ARRAY   VT_I2		
	0. i. 0 VT_I2		設定ランプを指定します。 1: ランプ 1 2: ランプ 2
	0. i. 1 VT_I2		下限値 (%) を指定します。 設定値範囲: 0 ~ 200
	0. i. 2 VT_I2		上限値 (%) を指定します。 設定値範囲: 0 ~ 255
1	VT_I2	書き込み保存フラグ 0: 書き込み値を保存しない(デフォルト) 1: 書き込み値を保存する	書き込み保存フラグは省略することができます。
pVal	なし		

**使用例**

```
CaoController.Execute("WriteMonitorAlarmThreshold", Array(Array(Array(1, 50, 100), Array(2, 200, 255)), 0))
```

## ReadMonitorAlarmThreshold

ランプ毎のモニタ輝度アラームの閾値を読み出すコマンドです。読出したいランプ番号を指定してください。

項目	型説明		備考
vntParam	VT_I2	読み出すランプを指定します。 1: ランプ 1 2: ランプ 2 3: 全てのランプ	
pVal	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1 です。
	0	VT_ARRAY   VT_VARIANT	
	0. i	VT_ARRAY   VT_I2	
	0. i. 0	VT_I2 ランプ(i+1) を取得します。 1: ランプ 1 2: ランプ 2	
	0. i. 1	VT_I2 下限値(%)を指定します。 設定値範囲: 0 ~ 200	
0. i. 2	VT_I2 上限値(%)を指定します。 設定値範囲: 0 ~ 255		

### 使用例

```
CaoController.Execute("ReadMonitorAlarmThreshold", 3)
```

## WriteFeedbackSetting

ランプ毎のフィードバック機能の設定を書き込むコマンドです。設定したいランプ番号と設定値をセットにして値を設定して下さい。

項目	型説明	備考	
vntParam	VT_ARRAY   VT_VARIANT	i の範囲は 0 ~ 1 です。 範囲を超えた部分に関しては無視されます。	
	0 VT_ARRAY   VT_VARIANT		
	0. i VT_ARRAY   VT_I2		
	0. i. 0 VT_I2		設定ランプを指定します。 1: ランプ 1 2: ランプ 2
	0. i. 1 VT_I2		フィードバック機能の設定を指定します。 0: OFF 1: ON 2: cPb(6V OFF)
1	VT_I2	書込み保存フラグ 0: 書込み値を保存しない(デフォルト) 1: 書込み値を保存する	書込み保存フラグは省略することができます。
pVal	なし		

### 使用例

```
CaoController.Execute("WriteFeedbackSetting", Array(Array(Array(1, 0), Array(2, 1)), 0))
```

## ReadFeedbackSetting

ランプ毎のフィードバック機能の設定を読み出すコマンドです。読出したいランプ番号を指定してください。

項目	型説明		備考
vntParam	VT_I2	読み出すランプを指定します。 1: ランプ 1 2: ランプ 2 3: 全てのランプ	
pVal	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1 です。
	0	VT_ARRAY   VT_VARIANT	
	0. i	VT_ARRAY   VT_I2	
	0. i. 0	VT_I2 ランプ(i+1) を取得します。 1: ランプ 1 2: ランプ 2	
	0. i. 1	VT_I2 フィードバック機能の設定を取得します。 0: OFF 1: ON 2: cPb(6V OFF)	

### 使用例

CaoController.Execute("ReadFeedbackSetting", 3)

## WriteLightingValueSetting

ランプ、シーケンス毎の点灯値の設定を書き込むコマンドです。設定したいシーケンス番号と設定値をセットにして値を設定して下さい。

項目	型説明		備考	
vntParam	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1, j の範囲は 1 ~ 4 です。範囲を超えた部分に関しては無視されます。	
	0	VT_ARRAY   VT_VARIANT		
	0. i	VT_ARRAY   VT_VARIANT		
	0. i. 0	VT_I2		設定ランプを指定します。 1: ランプ 1 2: ランプ 2
	0. i. j	VT_ARRAY   VT_I2		
	0. i. j. 0	VT_I2		設定シーケンス番号を指定します。 1: シーケンス 1 2: シーケンス 2 3: シーケンス 3 4: シーケンス 4
	0. i. j. 1	VT_I2		調光値を指定します。 値範囲: 0 ~ 999
0. i. j. 2	VT_I2	発行幅を指定します。 値範囲: 0 ~ 999		
1	VT_I2	書き込み保存フラグ 0: 書き込み値を保存しない(デフォルト) 1: 書き込み値を保存する	書き込み保存フラグは省略することができます。	
pVal	なし			

### 使用例

```
CaoController.Exec("WriteLightingValueSetting", Array(Array(Array(1, Array(1, 300, 10), Array(3, 1, 10)),
Array(2, Array(1, 200, 10), Array(4, 50, 10))), 0))
```

## ReadLightingValueSetting

ランプ、シーケンス毎の点灯値の設定を書き込むコマンドです。設定したいシーケンス番号と設定値をセットにして値を設定して下さい。

項目	型説明		備考	
vntParam	VT_ARRAY   VT_I2			
	0	VT_I2 ランプを指定します。 1: ランプ 1 2: ランプ 2 3: 全てのランプ		
	1	VT_I2 シーケンス番号を指定します。 1: シーケンス 1 2: シーケンス 2 3: シーケンス 3 4: シーケンス 4 5: 全てのシーケンス番号		
pVal	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1, j の範囲は 1 ~ 4 です。 範囲を超えた部分に関しては無視されます。	
	0	VT_ARRAY   VT_VARIANT		
	0. i	VT_ARRAY   VT_VARIANT		
		0. i. 0		VT_I2 ランプ (i+1) を取得します。
	0. i. j	VT_ARRAY   VT_I2		
		0. i. j. 0		VT_I2 シーケンス番号 (j) を取得します。
		0. i. j. 1		VT_I2 調光値を取得します。 値範囲: 0 ~ 999
0. i. j. 2	VT_I2 発行幅を取得します。 値範囲: 0 ~ 999			
pVal	なし			

### 使用例

```
CaoController.Execute("ReadLightingValueSetting", Array(3, 5))
```

## WriteLightingControlInput

シーケンス毎の点灯制御入力の設定を書き込むコマンドです。設定したいシーケンス番号と設定値をセットにして値を設定して下さい。

項目	型説明		備考	
vntParam	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1, j の範囲は 1 ~ 4 です。範囲を超えた部分に関しては無視されます。	
	0	VT_ARRAY   VT_VARIANT		
	0. i	VT_ARRAY   VT_VARIANT		
	0. i. 0	VT_I2		設定ランプを指定します。 1: ランプ 1 2: ランプ 2
	0. i. j	VT_ARRAY   VT_I2		
	0. i. j. 0	VT_I2		設定シーケンス番号を指定します。 1: シーケンス 1 2: シーケンス 2 3: シーケンス 3 4: シーケンス 4
0. i. j. 1	VT_I2	点灯制御入力を指定します。 1: 点灯制御入力 1 2: 点灯制御入力 2		
1	VT_I2	書込み保存フラグ 0: 書込み値を保存しない(デフォルト) 1: 書込み値を保存する	書込み保存フラグは省略することができます。	
pVal	なし			

### 使用例

```
CaoController.Execute("WriteLightingControlInput", Array(Array(Array(1, Array(1, 1), Array(3, 2)), Array(2, Array(1, 2), Array(4, 1))), 0))
```

## ReadLightingControlInput

ランプ、シーケンス毎の点灯制御入力の設定を書き込むコマンドです。設定したいシーケンス番号と設定値をセットにして値を設定して下さい。

項目	型説明		備考	
vntParam	VT_ARRAY   VT_I2			
	0	VT_I2 ランプを指定します。 1: ランプ 1 2: ランプ 2 3: 全てのランプ		
	1	VT_I2 シーケンス番号を指定します。 1: シーケンス 1 2: シーケンス 2 3: シーケンス 3 4: シーケンス 4 5: 全てのシーケンス番号		
pVal	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1, j の範囲は 1 ~ 4 です。 範囲を超えた部分に関しては無視されます。	
	0	VT_ARRAY   VT_VARIANT		
	0. i	VT_ARRAY   VT_VARIANT		
		0. i. 0		VT_I2 ランプ (i+1) を取得します。
	0. i. j	VT_ARRAY   VT_I2		
		0. i. j. 0		VT_I2 シーケンス番号 (j) を取得します。
		0. i. j. 1		VT_I2 点灯制御入力を取得します。 1: 点灯制御入力 1 2: 点灯制御入力 2
pVal	なし			

### 使用例

```
CaoController.Execute("ReadLightingControlInput", Array(3, 5))
```

## ReadCurrentData

ランプ毎の現在値を読み出すコマンドです。読出したいランプ番号を指定してください。

項目	型説明		備考	
vntParam	VT_I2	読み出すランプを指定します。 1: ランプ 1 2: ランプ 2 3: 全てのランプ		
pVal	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1 です。	
	0	VT_ARRAY   VT_VARIANT		
	0. i	VT_ARRAY   VT_I2		
	0. i. 0	VT_I2		ランプ (i+1) を取得します。
	0. i. 1	VT_I2		モニタ値を取得します。 値範囲: 0 ~ 4095
	0. i. 2	VT_I2		絶対輝度を取得します。 値範囲: 0 ~ 4095
0. i. 3	VT_I2	内部温度 (°C) を取得します。 値範囲: -100°C ~ 155°C		

### 使用例

```
CaoController.Execute("ReadCurrentData", 3)
```

## ReadCorrectionDimmingValue

簡易フィードバックによる補正後の調光値を読み出すコマンドです。読出したいランプ番号を指定してください。

項目	型説明		備考	
vntParam	VT_I2	読み出すランプを指定します。 1: ランプ 1 2: ランプ 2 3: 全てのランプ		
pVal	VT_ARRAY   VT_VARIANT		i の範囲は 0 ~ 1 です。	
	0	VT_ARRAY   VT_VARIANT		
	0. i	VT_ARRAY   VT_I2		
	0. i. 0	VT_I2		ランプ (i+1) を取得します。
	0. i. 3	VT_I2		補正調光値を取得します。 値範囲: 0 ~ 999

### 使用例

GaoController.Execute("ReadCorrectionDimmingValue", 3)

## GetTimeout

設定されているタイムアウト時間 (ms) を取得するコマンドです。

項目	型説明	
vntParam	なし	
pVal	VT_UI2	タイムアウト値 (ms) 0 ~

### 使用例

```
CaoController.Execute("GetTimeout")
```

## SetTimeout

タイムアウト時間 (ms) を設定するコマンドです。

項目	型説明	
vntParam	VT_UI2	タイムアウト値 (ms) 0 ~
pVal	なし	

### 使用例

```
CaoController.Execute("SetTimeout", 500)
```

## 2.4. 変数一覧

表 2.4 にコントローラクラスで使用できる変数一覧を記述します。

表 2.4 コントローラクラス変数一覧

変数名	説明	型	属性	
			get	put
@MAKER_NAME	製造元名を取得します。	VT_BSTR	○	—
@VERSION	デバイスバージョンを取得します。	VT_BSTR	○	—
@STATUS	<p>ステータスをビットフラグ形式で取得します。 各ビットの意味は以下の通りです。(＃はビットを表しています。)</p> <ul style="list-style-type: none"> <li>#0: 内部回路通信異常</li> <li>#1: 電源温度以上</li> <li>#2: EEPROM アクセス失敗</li> <li>#3: 入力電圧 18V 異常</li> <li>#4: IP アドレスが重複</li> <li>#5: DHCP エラー</li> <li>#6: LAMP1 フィードバックエラー</li> <li>#7: LAMP2 フィードバックエラー</li> <li>#8: LAMP1 6V 供給</li> <li>#9: LAMP1 モニタ輝度アラーム</li> <li>#10: LAMP1 過電流</li> <li>#11: LAMP1 出力停止</li> <li>#12: LAMP2 6V 供給</li> <li>#13: LAMP2 モニタ輝度アラーム</li> <li>#14: LAMP2 過電流</li> <li>#15: LAMP2 出力停止</li> </ul>	VT_I4	○	—

### 3. サンプルプログラム

以下にデバイスのステータス状態を取得する簡単なサンプルを示します。

前提条件：

- ・ デバイスのアドレスは 192.168.10.1 とする。

#### List 3-1 Sample.frm

```
Dim eng As CaoEngine
Dim ctrl As CaoController
Dim var As CaoVariable
Private Sub Form_Load()
    Set eng = New CaoEngine
    ' コントローラーオブジェクト作成
    Set ctrl = eng.Workspaces(0).AddController(
        "",
        "CaoProv. OPTEX-FA. OPPD",
        "",
        "CONN=TCP:192.168.10.1 ")
    ' コントローラー変数オブジェクト作成
    Set var = ctrl.AddVariable("@STATUS", "")
End Sub

' 値の取得
Private Sub GetValue_Click()
    ' コントロール変数値の取得
    Dim value As Variant
    Set value = ctrl.Value
End Sub
```