PD3 Provider

CCS digital light control for LED lighting

User's guide

Version 1.0.0

November 15, 2012

Remarks

Revision history

Date	Version	Description
11/15/2012	1.0.0	First edition

Applicable models

Model	Version	Comments

Contents

1. Introduction

This document is a user's guide of PD3 provider that is a CAO provider for CCS digital light control unit for LED lighting (PD3 series).

PD3 provider sends a command to PD3 and receives the response from it.

2. Overview of provider

2.1. Overview

PD3 provider provides CaoController::Execute for command execution.

CaoController::Execute method generates a command and sends it at the method execution. It also analyzes the response and receives data.

The file format of PD3 provider is DLL (Dynamic Link Library) and the detail is as follows.

Table 2-1 PD3 provider

File name	CaoProv.CCS.PD3.dll
ProgID	CaoProv.CCS.PD3
Registration[1]	regsvr32 CaoProv.CCS.PD3.dll
Un-registration	regsvr32 /u CaoProv.CCS.PD3.dll

[1] You do not need to register/un-register it manually if you installed it with ORiN SDK.

2.2. Method and property

2.2.1. CaoWorkspace::AddController method

In PD3 provider, to establish a connection, connection parameters for communication is referred at the AddController execution. For this parameter's option, you need to specify the communication format, and timeout.

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

bstrCtrlName : [in] Controller name (any name)
bstrProvName : [in] Provider name. Fixed to =" CaoProv.CCS.PD3"
bstrPcName : [in] Computer name where the provider runs
bstrOption : [in] Option string

The following shows the list of option string items.

Table 2-2 CaoWorkspace::AddController option strings

Option	Description
Conn =<Connection parameter >	Required. Set the communication format and connection parameters. For details, refer to 2.2.1.1.
Timeout[=<Timeout period>]	Specify the communication timeout (sec) for sending and receiving. (Default : 500)

2.2.1.1. Conn parameter

The following table shows the connection parameter strings of Parameter option. Parameters enclosed by square brackets ("[ ] ") can be omitted. An underlined selection in each parameter is the default value when there is no entry for the parameter.

- **Ethernet device** "eth:<IP Address>[:<Port No>]

<IP Address>	:	: Required. Set an IP address. Example : "127.0.0.1"
<Port No>	:	: Specify a port number for communication. (Default : 40001) Example : "127.0.0.1:40001"

2.2.2. CaoController::Execute method

Send and receive a command. Specify "command name" as the first argument, "command parameter" as the second argument. For information about each command, refer to Chapter 3 Command reference.

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

bstrCommandName: [in] Command name

vntParam : [in] parameter

2.3. Error code

In PD3 provider, the following original error code is exist. For about ORiN2 common errors, refer to the error code section of ORiN2 programming guide.

Table 2-4 Original error code list

Error name	Error number	Description
E_RESPONSE_FAILED	0x80100001	Invalid response.

3. Command reference

This chapter describes each command of CaoController::Execute method.

Table 3-1 CaoController::Execute command list

PD3command	Command	Function
-	Raw	Send and receive raw data.
F	LightLevel	Set the lighting control data.
S	EmissionMode	Set the lighting mode.
L	LightSwitch	Turn ON/OFF the light.
M	SettingStatus	Read the setting status.
C	OvercurrentStatus	Confirm the overcurrent status.
R	AllReset	Initialize all channels.
E01	IPAddress	Set an IP address.
E02	SubnetMask	Set a subnet mask.
E03	Gateway	Set a default gateway.
E04	SendIPAddress	Set a destination IP address.
E05	ReceivePort	Set a reception port.
E06	SendPort	Set a destination port.

3.1. Executecommand

3.1.1. CaoController::Execute("Raw") command

Send and receive raw data.

Syntax Raw (<bstrSendData>)

< bstrSendData > : [In] String to send (VT_BSTR)

Return value : [out] Received string (VT_BSTR)

Send a string entered in the argument. Response is returned as a string.

The contents of command and response are not processed at the data sending and receiving.

3.1.2. CaoController::Execute("LightLevel") command

Set the lighting control data.

Syntax LightLevel (<bstrChNo>,<bstrLightLevel>)

< bstrChNo > : [In] Channel number (VT_I4) (1 to 3)

<bstrLightLevel> : [In] Lighting control level (VT_I4) (0 to 255)

Return value : None

3.1.3. CaoController::Execute("EmissionMode") command

Set the lighting mode.

Syntax EmissionMode (<bstrChNo>, <bstrMode>)

< bstrChNo > : [in]Channel number (VT_I4) (1 to 3)

<bstrMode> : [in] Lighting mode (VT_I4)

0	1	2	3	4	5	6	7	8	9	10
Anytime Mode/ON-OFF mode	Strobe mode									
	40 us	80 us	120 us	200 us	600 us	1 ms	4 ms	10 ms	20 ms	40 ms

Return value : None

3.1.4. CaoController::Execute("LightSwitch") command

Turn ON/OFF the light.

Syntax LightSwitch (<bstrChNo>, <bstrSwitch>)

< bstrChNo > : [in]Channel number (VT_I4) (0 to 3)

0 : All channels

<bstrMode> : [in] Switching data (VT_I4)

0	Turn off
1	Turn on

Return value : None

3.1.5. CaoController::Execute("SettingStatus") command

Read the setting status.

Syntax SettingStatus (<bstrChNo>)

< bstrChNo > : [in]Channel number (VT_I4) (1 to 3)

Return value : [out] Setting information (VT_I4 | VT_ARRAY)

The return value stores the result in the order of the lighting control data setting value, lighting mode setting value, ON/OFF setting value.

3.1.6. CaoController::Execute("OvercurrentStatus") command

Confirm the overcurrent status.

Syntax OvercurrentStatus ()

Argument : None

Return value : [out] Overcurrent state information(VT_BOOL)

False	Normal
True	Abnormal

3.1.7. CaoController::Execute("AllReset") command

Initialize all channels.

Syntax AllReset ()

Argument : None

Return value : None

3.1.8. CaoController::Execute("IPAddress") command

Set an IP address.

Syntax IPAddress (<bstrIPAddress>)

<bstrIPAddress> : [in] IP address (VT_BSTR)

Return value : None

3.1.9. CaoController::Execute("SubnetMask") command

Set a subnet mask.

Syntax SubnetMask (<bstrMask>)

<bstrMask > : [In] Subnet mask (VT_BSTR)

Return value : None

3.1.10. CaoController::Execute("Gateway") command

Set a default gateway.

Syntax Gateway (<bstrIPAddress>)

<bstrIPAddress> : [in] gateway address (VT_BSTR)

Return value : None

3.1.11. CaoController::Execute("SendIPAddress") command

Set a destination IP address.

Syntax SendIPAddress (<bstrIPAddress>)

<bstrIPAddress> : [in] Sending destination IP address (VT_BSTR)

Return value : None

3.1.12. CaoController::Execute("ReceivePort") command

Set a reception port number.

Syntax ReceivePort (<bstrPort>)

<bstrPort> : [in] reception port number (VT_I4) (0 to 65535)

Return value : None

3.1.13. CaoController::Execute("SendPort") command

Set a sending destination port number.

Syntax SendPort (<bstrPort>)

<bstrPort> : [In] Sending gateway port number (VT_I4) (0 to 65535)

Return value : None

4. Sample Program

This program sets the lighting control data and shows the lighting control code.

4.1. VisualBasic Ver6.0

List 4-1 Sample.frm

```
Dim eng As CaoEngine
Dim ctrl As CaoController

Private Sub Form_Load()

    'Create CAO engine
    Set eng = New CaoEngine

    'Connection to the digital light
    Set ctrl = eng.Workspaces(0).AddController("Sample", "CaoProv.CCS.PD3", "",
"Conn=eth:192.168.0.2")

    ‘Set to turn the L1 lighting mode always ON.
    Ctrl.EmissionMode Array(1, 0)

End Sub

Private Sub Command1_Click()

    'Turn the L1 light ON
    Ctrl.LightSwitch Array(1, 1)

End Sub

Private Sub Command2_Click()

    'Turn the L1 light OFF
    Ctrl.LightSwitch Array(1, 0)

End Sub

Private Sub Command3_Click()

    'Set the L1 lighting control data
    Ctrl.LightLevel Array(1, CInt(Text1.Text))

End Sub
```

4.2. PacScript

List 4-2 Sample.pcs

```
Dim ctrl As Object

Sub Main()

    Dim cnt As Integer

    'Connection to the digital light
    ctrl = cao.AddController("Sample", "CaoProv.CCS.PD3", "", "Conn=eth:192.168.0.2")

    ctrl.EmissionMode Array(1, 0)      'Set to turn the L1 lighting mode always ON.
    ctrl.LightLevel Array(1,150)       ‘Set the L1 lighting control data to 150
    ctrl.LightSwitch Array(1,1)        ‘Turn the L1 light ON.

End Sub
```