PATLITE.
PNS provider

Version 1.0.2

User's guide

May 16, 2022

Remarks:
This document uses the machine translation.

【 revision history 】

Version	Date	Content
1.0.1	2016-12-20	First edition.
1.0.2	2020-06-29	Relaxed command argument type specification.
	2020-10-26	Fixed an error in the default port number
	2022-05-16	Corrected because WriteChannel and WriteLightChannel channel number ranges were incorrect.

【 ..confirming the operation.. model 】

Model	Version	Notes
PATLITE NHP-FV1		
PATLITE NHS-5FB1		WriteLightChannel, WriteChannel, and the WriteDigitalOutput command cannot be used.

1. Introduction

This book is an user's guide of the ORiN provider (PATLITE PNS provider) that does the control and data acquisition by using the network monitoring made of PATLITE light (NH series) and the PNS command.

This book explains the function of this PATLITE PNS provider and the mounting method.

2. Outline of provider

2.1. Outline

The file format of this provider is DLL(Dynamic Link Library). Please refer to Table2-1 for the details.

Table2-1 PATLITE PNS provider

File name	CaoProvPATLITEPNS.dll
ProgID	CaoProv.PATLITE.PNS
Registry registration	regsvr32 CaoProvPATLITEPNS.dll
Blotting out of registry registration	regsvr32 /u CaoProvPATLITEPNS.dll

Figure2-1 The correspondence chart of the NH series is shown each class of the provider.

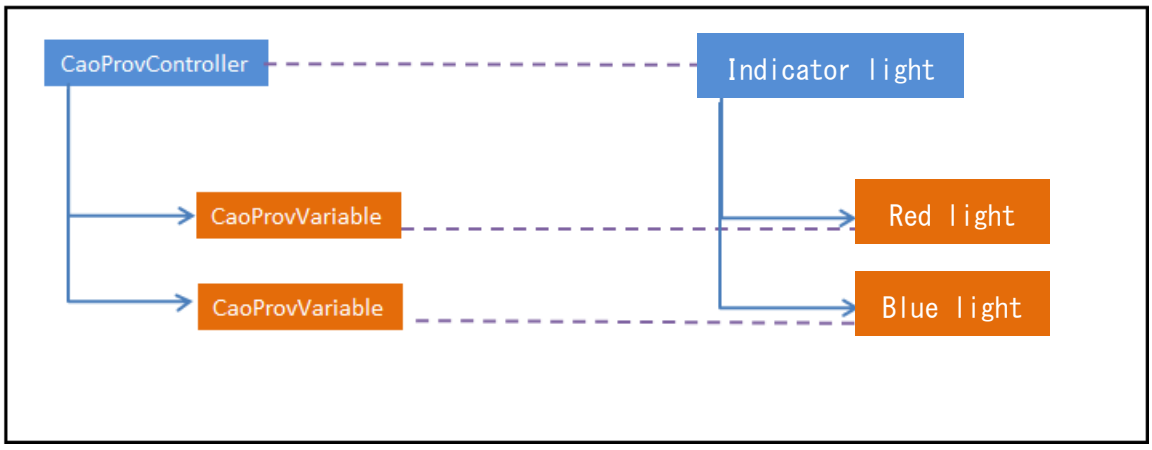

Figure2-1 Each class of provider and correspondence chart of NH series

2.2. Method property

2.2.1. CaoWorkspace::AddController method

In this provider, connected parameter is set with CaoWorkspace::AddController, and the communication is connected. The specification of AddController is shown as follows.

Format

```
AddController
(
      <bstrCtrlName:BSTR>,        // In controller name (unique name)
      <bstrProvName:BSTR>,        // In provider name (fixation ="CaoProv.PATLITE.PNS")
      <bstrPcName:BSTR>           // In provider execution machine name (unused)
      [,bstrOption:BSTR>]         // In optional character string
)
```

The character string specified for an optional character string is shown as follows.

Table2-2 Optional character string of CaoWorkspace::AddController

Option	Required	Explanation	Range of value	Default value
CONN=<Eth optional connection >	√	Internet Protocol address of the connected device is specified. About the specification method, please refer to 2.2.1.1.	-----	Destination IP: -- Destination port: 10000 IP: -- Port: --
WatcherPort=<Device port number for application program watch >	-	Please specify the device port number for the application program watch set to the device now. When this option and optional WatchingCycle are specified by numerical values other than 0, the application program monitoring function is executed on the back ground.	0 - 65535	0
WatchingCycle=< application program watch cycle>	-	Please specify the application program watch cycle set to the device by the integer	0 - 65535	0

		(second) now. When this option and optional WatcherPort are specified by numerical values other than 0, the application program monitoring function is executed on the back ground.		
TIMEOUT=<Response standby time>	-	The response standby time is specified (ms).	1 - 65535	500

2.2.1.1. Conn is optional.

Connected parameter character string of optional Conn is shown as follows. It is shown to omit it here in the brace) and the underlined part under the explanation of each parameter shows the default value when the option is not specified respectively.

TCP is connected when specifying ETH in this provider. Whether to specify UDP or TCP (ETH) depends on the setting value of the device.

Eth

"Conn=Eth:< connection destination IP > connection destination port of :< > :< local IP > :< local port >"

< connection destination IP > : Connection destination Internet Protocol address. "192.168.0.1" etc..

< connection destination port > : Connection destination port number.

10000

< local IP > : Local Internet Protocol address.

< local port > : Local port number.

2.2.2. CaoController::AddVariable method

Generate a CaoVariable object from CaoController. Only the variables listed in Table2-3 can be used in the variable name.

The specification of AddVariable is shown as follows.

Format

```
AddVariable
(
      <bstrVariableName:BSTR>          // In variable identifier
      [,<bstrOption:BSTR>])            // In optional character string
)
```

2.2.3. CaoController::get_VariableNames method

Get the list of variable names in Table 2-3.

2.2.4. CaoController::Execute method

In this provider, only "SendCommand" can be used as a command. SendCommand transmits the character string specified for vntParam (the second argument) to the device as it is, and returns the received character string to pVal (the third argument). Details of each command, pleae refer to Chapter 3.

Format

```
Execute
(
      <bstrCommand:BSTR>         // [In] command name ("SendCommand fixation =")
      [,<vntParam:VARIANT>]      // [In] parameter (transmission character string to
device)
      [,<pVal:VARIANT>]])        // [In] execution result (reception character string
```

2.3. Variable list

2.3.1. CaoController

Table2-3 shows the variable list that can be used in CaoController class.

Table2-3 CaoController class variable list

Variable name	Data type of acquisition value	Explanation	Attribute	
			get	put
@MAKER_NAME	VT_BSTR	The manufacturer name is returned.	√	-
@VERSION	VT_BSTR	The version of the provider is returned.	√	-

@RED	VT_UI1		The value of a red lamp setting/is acquired. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change (Only at put).	√	√
@YELLOW	VT_UI1		The value of a yellow lamp setting/is acquired. Please refer to @RED for each value.	√	√
@GREEN	VT_UI1		The value of a green lamp setting/is acquired. Please refer to @RED for each value.	√	√
@BLUE	VT_UI1		The value of a blue lamp setting/is acquired. Please refer to @RED for each value.	√	√
@WHITE	VT_UI1		The value of a white lamp setting/is acquired. Please refer to @RED for each value.	√	√
@BUZZER	VT_UI1		The value of the buzzer setting/is acquired. 0: Stop 1: Buzzer pattern 1 2: Buzzer pattern 2 3: Buzzer pattern 3 4: Buzzer pattern 4	√	√
@LIGHTBUZZER	VT_ARRAY | VT_UI1		Red-white and the value of the buzzer setting/are acquired. Please refer to @RED and @BUZZER for each value.	√	√
	0	VT_UI1	Red lamp		
	1	VT_UI1	Yellow lamp		
	2	VT_UI1	Green lamp		
	3	VT_UI1	Blue lamp		
	4	VT_UI1	White lamp		
	5	VT_UI1	Buzzer		

2.4. Error code

In this provider, the following and original the error code exists. (Table2-4 Reference)

Please refer to the chapter of the error code of "ORiN2 programming guide" for the ORiN2 commonness error.

Moreover, the code is assumed about the error code received from the device and the value in which the mask is done with 0x80100000 is assumed to be an error code of the provider. Please mean the error code received from the device, and refer to the manual of PATLITE.

Table2-4 Original error code table

Error number	Explanation
0x80110001	TCP/ ETH/ The communication kinds other than UDP were specified.
0x80110002	The value of WatcherPort is illegal.
0x80110003	The value of WatchingCycle is illegal.

3. Command reference

This chapter explains each command of the CaoController::Execute method.

Table3-1Command list

Command	Function	
WriteLightBuzzer	The value of the light and the buzzer is written at the same time.	P.11
ReadLightBuzzer	The value of the light and the buzzer is read now.	P.12
WriteChannel	The channel control is written. This command can be executed only with the device that can reproduce MP3.	P.13
WriteDigitalOutput	The value of the digital output is written. It is effective only in the model that maintains the digital output.	P.13
WriteLightChannel	The light and the channel control are written at the same time. It is effective only in the model that can control the channel.	P.13
SendPacket	The specified packet data is transmitted to the device.	P.14

3.1. WriteLightBuzzer command

The value of the light and the buzzer is written at the same time.

Table3-2 WriteLightBuzzer command argument

Argument	Function		
[in] vntParam	VT_ARRAY | VT_UI1		
	0	VT_UI1	The value of a red lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	1	VT_UI1	The value of a yellow lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	2	VT_UI1	The value of a green lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	3	VT_UI1	The value of a blue lamp is specified. 0: Turning off 1: Lighting always

			2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	4	VT_UI1	The value of a white lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	5	VT_UI1	The value of the buzzer is specified. 0: Stop 1: Buzzer pattern 1 2: Buzzer pattern 2 3: Buzzer pattern 3 4: Buzzer pattern 4 9: There is no value change.
[out] pVal	--		

3.2. ReadLightBuzzer command

Read the current value of light and buzzer.

Table3-3 ReadLightBuzzer command argument

Argument	Function		
[in] vntParam	VT_EMPTY		
[out] pVal	VT_ARRAY | VT_UI1		
	0	VT_UI1	Value of red lamp 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2
	1	VT_UI1	Value of yellow lamp 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2
	2	VT_UI1	Value of green lamp Please refer to a red lamp for the value. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2
	3	VT_UI1	Value of blue lamp Please refer to a red lamp for the value. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2
	4	VT_UI1	Value of white lamp Please refer to a red lamp for the value.

			0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2
	5	VT_UI1	Value of buzzer 0: Stop 1: Buzzer pattern 1 2: Buzzer pattern 2 3: Buzzer pattern 3 4: Buzzer pattern 4

3.3. WriteChannel command

The channel control is written. The MP3 file set to the channel can be beforehand, be reproduced, and be stopped for this command. This command can be executed only with the device that can reproduce MP3.

Table3-4WriteChannel command argument

Argument	Function		
	0	VT_UI1	The reproduction pattern is specified. 0: Reproduction stop 1: Reproduction
	1	VT_UI1	The repetition frequency is specified. 0: None of repetition 1 - 254: Specified frequency repetition reproduction 255: Infinite reproduction
	2	VT_UI1	The channel number is specified (1 - 70).
[out] pVal	--		

3.4. WriteDigitalOutput command

The value of the digital output is written.

It is effective only in the model that maintains the digital output.

Table3-5 WriteDigitalOutput command argument

Argument	Function		
[in] vntParam	0	VT_UI1	Value of digital output 0: OFF 1: ON 9:State maintenance
[out] pVal	--		

3.5. WriteLightChannel command

The light and the channel control are written at the same time.
It is effective only in the model that can control the channel.

Table3-6 WriteLightChannel command argument

Argument	Function		
[in] vntParam	VT_ARRAY | VT_UI1		
	0	VT_UI1	The value of a red lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	1	VT_UI1	The value of a yellow lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	2	VT_UI1	The value of a green lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	3	VT_UI1	The value of a blue lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	4	VT_UI1	The value of a white lamp is specified. 0: Turning off 1: Lighting always 2: Blinking pattern 1 3: Blinking pattern 2 9: There is no value change.
	5	VT_UI1	The reproduction pattern is specified. 0: Reproduction stop 1: Reproduction
	6	VT_UI1	The repetition frequency is specified. 0: None of repetition 1 - 254: Specified frequency repetition reproduction 255: Infinite reproduction
	7	VT_UI1	The channel number is specified (1 - 70).
[out] pVal	--		

3.6. SendPacket command

The specified packet data is transmitted to the device.

Table3-7 SendPacket argument

Argument	Function	
[in] vntParam	VT_ARRAY | VT_UI1	Please specify the packet data that transmits by the binary.
[out] pVal	VT_ARRAY | VT_UI1	The binary data received from the device after it transmits is maintained.