Yokogawa Electric Corporation
PCLinkASCII provider

Version 1.0.2

User's guide

August 08, 2022

```
Remarks:
This document uses the machine translation.
```

【 revision history 】

Version	Date	Content
1.0.0	2016-09-28	First edition.
1.0.1	2019-08-23	Fixed internal logic.
1.0.2	2022-08-08	Fixed internal logic.

【 ..confirming the operation.. model 】

Model	Version	Notes
YOKOGAQA F3LE01	01	
	–	

Contents

1. Introduction

This book is an user's guide of the ORiN provider (YOKOGAWA PCLinkASCII provider) that acquires data by using the high-ranking link function made by Yokogawa Electric Works, Ltd. PLC.

If this provider is used, the high-ranking link function made of a horizontal river PLC can be used. This provider responds only to the communication by ASCII code.

This book explains the function of this YOKOGAWA PCLink provider and the mounting method.

2. Outline of provider

2.1. Outline

  The YOKOGAWA PCLinkASCII provider acquires data by using the high-ranking link function (TCP/IP communication) made by Yokogawa Electric Works, Ltd. PLC. The file format of this provider is DLL (Dynamic Link Library), and the details.

Table2-1YOKOGAWA PCLinkASCII provider

File name	CaoProvYOKOGAWAPCLinkASCII.dll
ProgID	CaoProv.YOKOGAWA.PCLinkASCII
Registry registration	regsvr32 CaoProvYOKOGAWAPCLinkASCII.dll
Blotting out of registry registration	regsvr32 /u CaoProvYOKOGAWAPCLinkASCII.dll

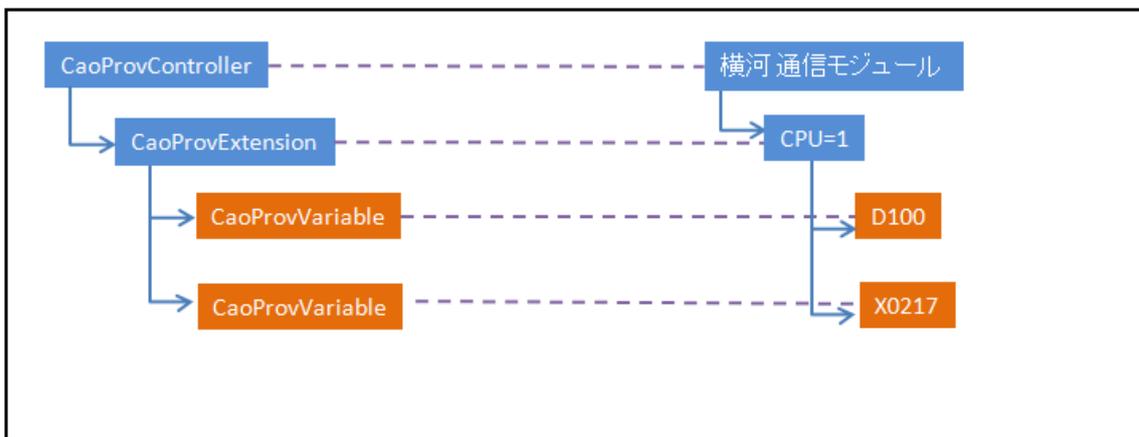Figure2-1The correspondence chart of F3LE01 is shown each class of provider.

Figure2-1Each class of provider and correspondence chart of F3LE01

2.2. Method property

2.2.1. CaoWorkspace::AddController method

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

Format

```
AddController
(
      <bstrCtrlName:BSTR>,        // [in]controller name (unique name)
      <bstrProvName:BSTR>,        // [in]provider name ("CaoProv.YOKOGAWA.PCLink")
      <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.

Table2-2Optional character string of CaoWorkspace::AddController

Option	Indispensability	Explanation	Range of value	Default value
Conn=<connection option>	-	Specify Internet Protocol address of the connected device. About the specification method, please refer to 2.2.1.1.	-----	Destination IP: -- Destination port: 12289 IP: -- Port: --
Timeout=<Response standby time of < >	--	Specify the response standby time (ms).	1 - 65535	500
Delay=< delay time >	--	Specify the DeLay time between the commands (ms).	0 - 2147483647	10

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.

Connect TCP when specifying ETH in this provider.

  Eth

   "Conn=Eth:<IP>[:<Port>[:<LocalIP>[:LocalPort]]]"

      < IP >   : Connection destination Internet Protocol address. "192.168.0.1" etc..
      < Port >   : Connection destination port number. (default:12289)
      < Local IP >   : Local Internet Protocol address.
      < Local port > : Local port number.

2.2.2. CaoController::AddExtension method

Generate the CaoExtension object from CaoController. The specification of AddExtension is shown as follows.

Format

```
AddExtension
(
        <bstrExtensionName:BSTR>            //[in] variable name (arbitrariness)
        [,<bstrOption:BSTR>])               //[in] optional character string
)
```

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

Option	Indispensability	Explanation	Range	Default value
CPU=<CPU number>	✓	Specify CPU number corresponding to added CaoExtension. If the value is specified ..outside the range.., it becomes an error.	1 – 4	--

2.2.3. CaoController::AddVariable method

Generate the CaoVariable object from CaoController. To the variable identifierTable2-3Only the variable from which is enumerated can be used.

The specification of AddVariable is shown as follows.

Format

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

2.2.4. CaoController::get_VariableNames property

Table2-3Drink and acquire the variable name list.

2.2.5. 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).

Format

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

2.2.6. CaoExtension::AddVariable method

Generate the CaoVariable object from CaoExtension. To the variable identifier Table2-4 Only the variable from which is enumerated can be used.

2.2.7. CaoExtension::get_VariableNames property

Table2-4 drink and acquire the variable name list.

2.2.8. CaoVariable::get_Value property

Acquire the value of the variable that corresponds to the object.
Please refer to 2.3 for the mounting situation and the data type of the variable.

2.2.9. CaoVariable::put_Value property

Set the value to the variable that corresponds to the object.
Please refer to 2.3 for the mounting situation and the data type of the variable.

2.3. Variable list

2.3.1. CaoController

Table2-3The variable list that can be used in CaoController class is described.

Table2-3CaoController class variable list

Variable identifier	Data type of acquisition value	Explanation	Attribute get	put
@MAKERNAME	VT_BSTR	Return the manufacturer name.	✓	–
@VERSION	VT_BSTR	Return the version of the provider.	✓	–

2.3.2. CaoExtension

Table2-4 the variable list that can be used in CaoExtension class is described. Only the get operation is possible by all variables.

Table2-4CaoExtension class variable list

Variable name	Explanation	Attribute get	put	Details
DV<??>	Acquisition/set data from the sequence device.	✓	✓	Refer to DV<??>.
SP<??>	Acquisition/set data from a special module.	✓	✓	Refer to SP<??>.
PG<??>	Acquire program information.	✓	–	Refer to PG<??>.
PB<??>	Acquire program block information.	✓	–	Refer to PB<??>.
IF<??>	Acquire various information.	✓	–	Refer to IF<??>.
MN<??>	Acquire the module name.	✓	–	Refer to MN<??>.
ER<??>	Acquire the error history.	✓	–	Refer to ER<??>.
UL<??>	Acquire the user log.	✓	–	Refer to UL<??>.

2.3.2.1. DV<??>

The variable from the sequence device from which data acquisition/is set. Make the variable specifying an arbitrary character string since DV. Table2-5Enumerate the option for which can be specified.

Table2-5DV<> is optional.

Option	Indispensability	Explanation	Range of value	Default value
Device=<Device Name>	✓	Specify the read device name. See Yokokawa's manual for the device that can be read.	––	––
UserVarWidth=<Bit length >	––	Specify one point's worth of reading length. The bit length that can be specified is different according to the read device. It operates by the default value when the specified bit length is outside the range of the value.	Link: 1/16/32 Register: 16/32 Timer: 16	Link device: 1 Register device: 16 Timer: 16
Elem=<number of size>	––	Specify the read point. It operates by the default value when the specified point is outside the value range.	1 – 65535	1
Array=< Array existence >	––	Specify whether to acquire it in the array when one is specified for optional Elem.	True/False	False

Data type of acquisition/setting:

It becomes VT_VARIANT with the specified number of elements though it differs according to optional UserVarWidth ..optional Elem it...

1. At one bit specification

It is VT_UI1 of which 0 1, and is turning off as for turning on. | Return VT_VARIANT.

2. At 16 bit specification

VT_UI2 | Return VT_VARIANT.

3. At 32 bit specification

VT_UI4 | Return VT_VARIANT.

2.3.2.2. SP<??>

The variable from a special module from which data acquisition/is set. Make the variable specifying an arbitrary character string since SP. Table2-6The option for which can be specified is described.

Table2-6SP<> is optional.

Option	Indispensability	Explanation	Range of value	Default value
Unit=< unit number >	✓	Specify the unit-number of a special module. If the value is specified ..outside the range.., it becomes an error.	0 - 7	--
Slot=< slot number >	✓	The slot number of the specified unit is specified optional Unit it. If the value is specified ..outside the range.., it becomes an error.	1 - 16	--
Pos=< number of potision >	✓	Specify the position of reading data or writing data. If the value is specified ..outside the range.., it becomes an error.	0 - 9999	--
UserVarWidth=< data length>	--	Specify one point's worth of reading or writing length bitting. It operates by the default value when the specified bit length is outside the range of the value.	16/32	16
Elem=<number of size >	--	Specify reading or writing point. It operates by the default value when the specified point is outside the value range.	1 - 65535	1
Array=< Array existence >	--	Specify whether to acquire it in the array when one is specified for optional Elem.	True/False	False

Data type of acquisition/setting:

It becomes VT_VARIANT with the specified number of elements though it differs according to optional UserVarWidth ..optional Elem it...

1. At 16 bit specification

VT_UI2 | VT_VARIANT

2. At 32 bit specification

VT_UI4 | VT_VARIANT

2.3.2.3. PG<??>

It is a variable to acquire registered program information. Make the variable specifying an arbitrary character string since PG. Table2-7The data type of acquisition value is described.

Table2-7Data type of PG<> acquisition value

Data type		Explanation	Range of value
VT_ARRAY | VT_VARIANT		--	--
0	VT_BSTR	Store the program name.	--
1	VT_UI4	Store the number of steps of programs.	0 - 999999
2	VT_DATE	Store the date and time of creation of the program.	--

2.3.2.4. PB<??>

It is a variable to acquire program information on a certain block. Make the variable specifying an arbitrary character string since PB. Table2-8The option for which can be specifiedTable2-9, Table2-10 data type is described.

Table2-8PB<> is optional.

Option	Indispensability	Explanation	Range of value	Default value
Block=< number of Block >	✓	Specify the reading start program block number. It becomes an error, except for the value range.	1 - 999	--
Elem=< number of size >	--	Specify the number of reading block numbers. It operates by the default value, except for the value range.	1 - 65535	1
Array=< Array existence >	--	Specify whether to acquire it in the array when one is specified for optional Elem.	True/False	False

Table2-9Data type when PB<> Elem is plurals or Array=True

Data type			Explanation	Range of value
VT_ARRAY | VT_VARIANT			--	--
0	VT_UI2		Store the number of reading blocks. Because the number of elements that can be acquired the specified number of blocks and actually might be different, this item is seen, and arrange and judge the read number of blocks optional Elem it.	0 - Elem
1	VT_ARRAY | VT_VARIANT			--
	2i	VT_BSTR	Store the block name.	--
	2i + 1	VT_UI4	Store the number of steps in the block.	0 - 999999

*I is a value in which the specified block number is assumed to be 0.

Table2-10Data type when PB<> Elem is 1 and Array=False

Data type			Explanation	Range of value
0	VT_ARRAY | VT_VARIANT			--
	0	VT_BSTR	Store the block name.	--
	1	VT_UI4	Store the number of steps in the block.	0 - 999999

2.3.2.5. IF<??>

It is a variable to acquire various information. Make the variable specifying an arbitrary character string since IF. Table2-11The option for which can be specifiedTable2-12The data type of acquisition value is described.

Table2-11IF<> is optional.

Option	Indispensability	Explanation	Range of value	Default value
No =< Information Kind>	--	Specify the read information kind. The data type acquired according to the value specified here changes. Moreover, it operates by the default value, except for the value range.	1: CPU information reading 2: System ID, CPU type, and area size information reading	1

Table2-12Data type of IF<> acquisition value

No	Data type		Explanation	Range of value
1	VT_ARRAY | VT_VARIANT		Data type at No=1	--
	0	VT_UI1	Store CPU status.	0: Ready 1: Not Ready
	1	VT_UI1	Store the program status.	1: Driving 2: Stop 3: Debugging 4: ROM writer
2	VT_ARRAY | VT_VARIANT		Data type at No=2	--
	0	VT_BSTR	Store system ID.	--
	1	VT_BSTR	Store [ribijon].	--
	2	VT_UI1	Store CPU type.	1: Sequence 2: BASIC
	3	VT_UI2	Store the program area size. For the sequenceNumber of steps For BASICNumber of bytes It means it in each.	--

2.3.2.6. MN<??>

It is a variable to acquire the module name. Make the variable specifying an arbitrary character string since MN. Table2-13The option for which can be specified is described and the data type of acquisition value is described.

Table2-13MN<> is optional.

Option	Indispensability	Explanation	Range of value	Default value
Unit=< unit-number >	–	Specify the unit-number that wants to read the module name. If the value is specified ..outside the range.., it becomes an error.	0 – 7	--

Table2-14Data type of MN<> acquisition value

Data type		Explanation	Range of value
VT_ARRAY | VT_VARIANT		Slot 1 Store module name information to – slot 16.	--
0	VT_BSTR	Store the module name of slot 1.	--
1	VT_UI1	Store the I/O type of slot 1.	0: There is no I/O relay. 1: Only the input relay 2: Only the output relay 3: There is I/O.
2	VT_UI2	Store the I/O point of slot 1.	
:	:	:	:
:	:	:	:
44	VT_BSTR	Store the module name of slot 16.	
45	VT_UI1	Store the I/O type of slot 16.	
46	VT_UI2	Store the I/O point of slot 16.	

2.3.2.7. ER<??>

It is a variable to acquire the error history. Make the variable specifying an arbitrary character string since ER. Table2-15The option for which can be specifiedTable2-16The data type of acquisition value is described.

Table2-15ER<> is optional.

Option	Indispensability	Explanation	Range of value	Default value
Pos=< position >	✓	Specify the error history reading position. If the value is specified ..outside the range.., it becomes an error.	0: The latest error data 1 - 999: Error data of N turn eyes from the latest	--

Table2-16Data type of ER<> acquisition value

Data type			Explanation	Range of value
VT_ARRAY | VT_VARIANT			--	--
0	VT_UI1		Store the error kind. The data type of additive information (element number 3) changes depending on this value.	0: System error 1: BASIC error 2: Sequence error 3: I/O is abnormal.
1	VT_UI2		Store the error code.	--
2	VT_DATE		Store the generation date.	--
3	VT_ARRAY | VT_VARIANT		It is additive information when the sequence makes an error.	--
	0	VT_BSTR	Store the block name.	--
	1	VT_UI4	Store the instruction number.	--
3	VT_ARRAY | VT_VARIANT		It is additive information when BASIC makes an error.	--
	0	VT_BSTR	Store the program name.	--
	1	VT_BSTR	Store the subprogram name.	--
	2	VT_UI4	Store the line number.	--
3	VT_ARRAY | VT_VARIANT		It is additive information on abnormal circumstances of I/O.	
	0	VT_UI2	Store the slot number.	--
	1	VT_UI2	Store the detailed error number.	--
3	VT_EMPTY		It is additive information when the system makes an error.	

2.3.2.8. UL<??>

It is a variable to acquire the user log. Make the variable specifying an arbitrary character string since UL. Table2-17The option for which can be specified is described.

Table2-17UL<> is optional.

Option	Indispensability	Explanation	Range of value	Default value
Pos=<position >	✓	Specify the user log reading position. If the value is specified ..outside the range.., it becomes an error.	0: The latest data 1 - 999: Data of N turn eyes from the latest	--

The data type acquired in UL<?? > becomes VT_BSTR.

2.4. Error code

In this provider, the following and original the error code exists. (Table2-18Reference)

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. Mean the error code received from the device, and refer to specifications of Yokokawa's high-ranking link command.

Table2-18Original error code table

Error number	Explanation
0x80110001	TCP/ The communication kinds other than ETH were specified.
0x80110002	CPU is not specified.
0x80110003	The value specified with CPU is outside the range.
0x80110004	Device is not specified.
0x80110005	The character string specified with Device is illegal.
0x80110006	Unit is not specified.
0x80110007	The value specified with Unit is outside the range.
0x80110008	Slot is not specified.
0x80110009	The value specified with Slot is outside the range.
0x8011000A	Pos is not specified.
0x8011000B	The value specified with Pos is outside the range.
0x8011000C	Block is not specified.
0x8011000D	The value specified with Block is outside the range.

3. Sample program

An easy sample to acquire the data for 500 words from D200 is shown as follows.

```
Precondition:
・Internet Protocol address of the device is adjusted to 192.168.0.2.
```

List 3-1	Sample.frm

```vb
Dim eng As CaoEngine
Dim ctrl As CaoController
Dim ext As CaoExtension
Dim var As CaoVariable
Private Sub Form_Load()
        Set eng = New CaoEngine
        ' Create controller object .
        Set ctrl = eng.Workspaces(0).AddController(
                "YOKOGAWA", "CaoProv.YOKOGAWA.PCLinkASCII", "",
                "CONN=ETH:192.168.0.2")
        ' Create Extension object
        Set ext = ctrl.AddExtension(
                "Ext00", "CPU=1")
        ' Create variable object
        Set var = ext.AddVariable(
                "DV00", "Device=D200, Elem=500")
End Sub


' get value
Private Sub Command1_Click()
        ' acquisition of value of a variable
        Dim d200 As Variant
        Set d200 = var.Value
End Sub
```