BALLUFF
IO-Link provider

Version 1.1.0

User's guide

November 27, 2020

Remarks:
This document uses the machine translation.

【 revision history 】

Version	Date	Content
1.0.0	2016-08-30	First edition.
1.0.2	2016-11-16	The explanation of the unit acquisition variable to the Extension variable is added.
1.0.3	2017-06-01	It corresponds to writing the process data and the reading and writing of the parameter data.
1.0.4	2017-10-12	The data type was able to be specified for acquisition and the setting of the value.
1.1.0	2019-01-23	It corresponded to the ArrayT type. It corresponded to the ProcessDataInUnionT type. It corresponded about the ProcessDataOutUnionT type. Correspondence [suki-maba-jon] was described clearly. Trouble concerning reading the XML file was corrected.
	2019-06-17	As a precaution for connecting the client PC to IO-Link master, the procedure for setting up Windows firewall was added.
	2020-11-27	Described the description of IODD file.

【 connected confirmation ending equipment 】

■ IO-Link mastering

Model name	Firmware version	Notes
BNI EIP-507-005-Z040	4.2.1	

- The firmware of the connected IO-Link mastering cannot connect the one of **4.3.0** or less or there is a possibility that the communication of data doesn't go well from the relation of API made by the BALLUFF company.

Contents

1. Introduction

This book is an user's guide of the BALLUFF IO-Link provider (hereafter, IO-Link provider) that reads data to the IO-Link mastering made by the BALLUFF company (hereafter, master), and writes it.

It becomes easy to acquire data from mastering and the IO-Link device (hereafter, device), and to write if the IO-Link provider is used.

This book explains the function of the IO-Link provider and the mounting method.

Please refer to the following IO-Link homepages for details of the IO-Link standard.

http://www.io-link.com/en/

2. Outline of provider

2.1. Outline

The IO-Link provider reads and writes data for the device connected with the data of mastering and the port of mastering by connecting with mastering by the EtherNet/IP communication, and using API"IOLUDPIF20.dll" made by the BALLUFF company.

Moreover, "IODD file (*2)" in which information on "IODM file (*1)" in which information on mastering is described and the device is described is necessary to use this provider.

The schema version of the IODD file that corresponds in this provider is 1.1. Please note that IODD files other than Ver1.1 become operation guarantees off the subject.

(*1) The IODM file is put on the Bin¥XML folder, and refer, please.

(*2) The IODD file can be downloaded from following URL.

https://ioddfinder.io-link.com/#/

The under moreoverFigure2-1However, it becomes a whole of mastering and the device block diagram with [hon] provider.

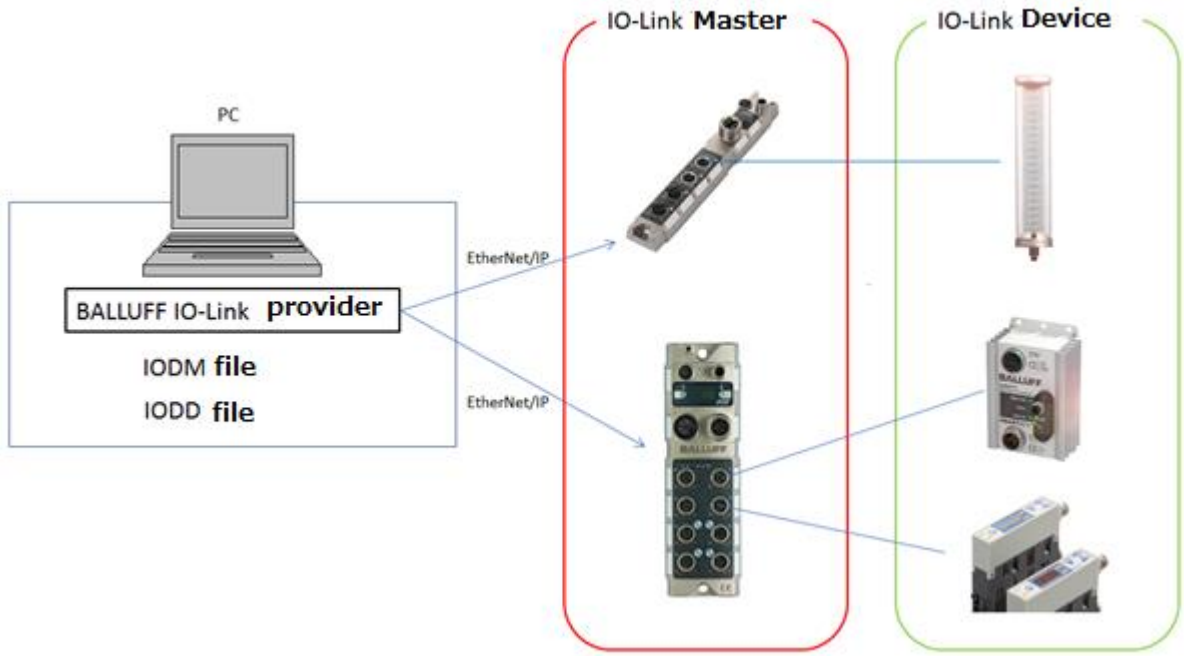


Figure2-1Whole block diagram

The file format of the IO-Link provider is DLL(Dynamic Link Library), and the details :. Table2-1It is [natteimasu] in [noyou].

Table2-1BALLUFF IO-Link provider

File name	CaoProvBALLUFFIOLink.dll
ProgID	CaoProv.BALLUFF.IOLink
Registry registration	regsvr32 CaoProvBALLUFFIOLink.dll
Blotting out of registry registration	regsvr32 /u CaoProvBALLUFFIOLink.dll

Moreover, figure where this provider, mastering, and correspondence each device were shown is a figure below. Figure2-2It becomes it.
(* It is one example. It is not because everything is shown.)

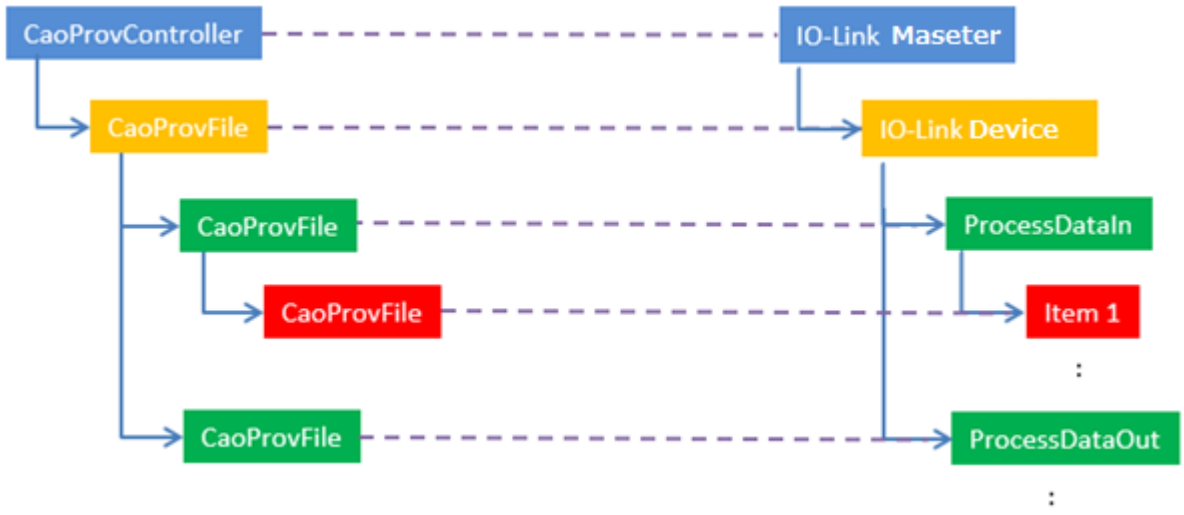


Figure2-2IO-Link provider and figure for data of IO-Link equipment

2.2. Relation of data of file class and device

This chapter describes the file class the relation of the data of the device connected with it.

The underFigure2-3It explains that each node of [ni] file class was allotted in numerical order.

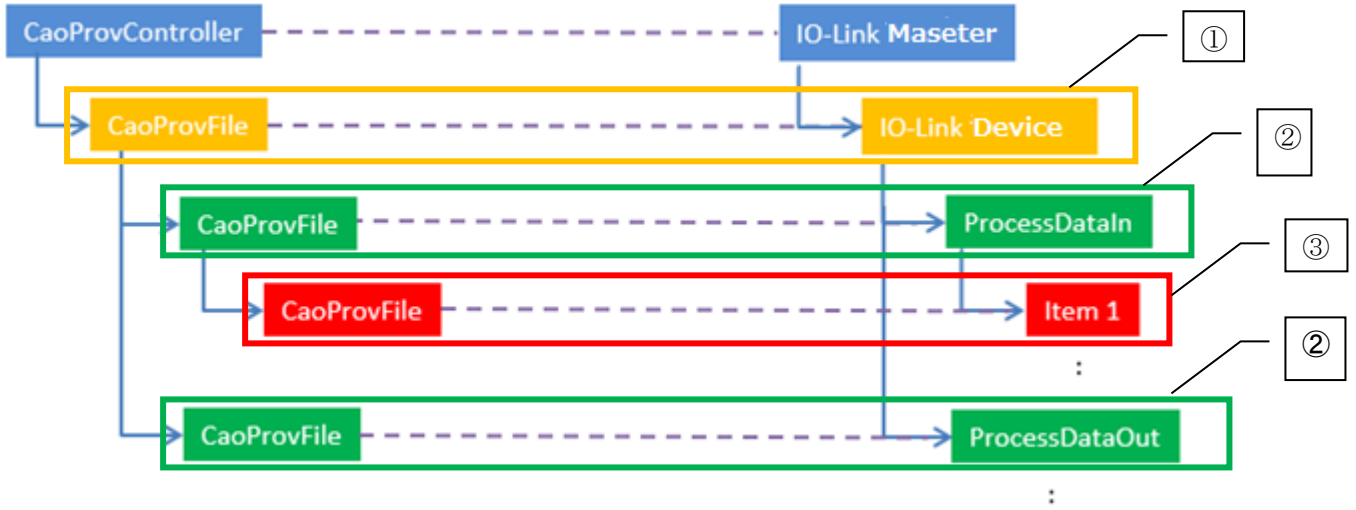


Figure2-3File class and chart related to data of device

① Device node

CaoController::AddFileThe file class added by the method is called a device node here.

In the device node, the following methods can be used.

・CaoFile::AddFile

・CaoFile::GetFileNames

The device node has the connected process data of the device and name information on each parameter. Maintained name information can be acquired in the CaoFile::GetFileNames method.

Moreover, the file name that can be added by the CaoFile::AddFile method of the device node can use only the name acquired in the CaoFile::GetFileNames method.

Moreover, there is each classification in the file name acquired with CaoFile::GetFileNames, and as follows. Table2-2The correspondence of [ni] name and the classification is described.

Table2-2File name that can be acquired in CaoFile::GetFileNames method and the classification

Name	Classification	Explanation	Remarks
“ProcessDataIn”	ProcessDataIn	It is a file class name in which ProcessDataIn of the device is exchanged with data.	This name might not exist according to the read IODD file.
“ProcessDataOut”	ProcessDataOut	It is a file class name in which ProcessDataOut of the device is exchanged with data.	This name might not exist according to the read IODD file.
Each parameter name	Parameter	It is a file class name with parametrical information on the device.	

The sample of PacScript until the device node is added is described as follows.

Usage example

```
Sub Main
        Dim caoCtrl as Object
        Dim caoSample as Object

        '①．  IO-Link Maseter(SampleMaster) and connection
        caoCtrl=cao.AddController("IOLink","caoProv.Balluff.IOLink","",
"server=192.168.1.100,Path=SampleMaster.xml")
        '②．  IO-Link device (SampleDev) and connection
        caoSample=caoCtrl.AddFile("Sample","PortNo=0, Path=SampleDevice.xml")
End Sub
```

② Data classification node

The file class added by the CaoFile::AddFile method of the device node is called a data classification node here.

The data classification node shows either about the IO-Link device among data "ProcessDataIn", "ProcessDataOut", and "Parameter". When "ProcessDataIn" is specified when the file is added by the CaoFile::AddFile method of the device node, the classification of the added data classification node becomes ProcessDataIn.

When "ProcessDataOut" is specified when ProcessDataOut also similarly adds the file by the CaoFile::AddFile method of the device node, it becomes ProcessDataOut.

When each parameter character strings other than "ProcessDataIn" and "ProcessDataOut" are specified by the CaoFile::AddFile method of the device node, all the classifications become Parameter.

Moreover, the file name that can be added by the CaoFile::AddFile method can specify only the file name acquired with CaoFile::GetFileNames for the data classification node as well as the device node.

The method that can be used whenever the data classification node is classified as followsTable2-3[Ni] is shown.

Table2-3Method list that can be used whenever classifying it

Classification	Method that can be used
ProcessDataIn	・CaoFile::GetFileNames ・CaoFile::AddFile
ProcessDataOut	・CaoFile::GetFileNames ・CaoFile::AddFile
Parameter	・CaoFile::GetFileNames ・CaoFile::AddFile ・CaoFile::Execute(“GetAccessType”)

The sample of PacScript until the classification of the data of ProcessDataIn node is added is described as follows.

Usage example

```
Sub Main
        Dim caoCtrl as Object
        Dim caoSample as Object
 Dim caoProcessDataIn as Object

        '①．  IO-Link Maseter(SampleMaster) and connection
        caoCtrl=cao.AddController("IOLink","caoProv.Balluff.IOLink","",
"server=192.168.1.100,Path=SampleMaster.xml")
        '②．  IO-Link device (SampleDev) and connection
        caoSample=caoCtrl.AddFile("Tower","PortNo=0, Path=SampleDev.xml")
        . Classification  “ProcessDataIn” ..'③..  is added.
        caoProcessDataIn = caoSample.AddFile("ProcessDataIn")
End Sub
```

③ Item node

The file class added by the CaoFile::AddFile method of the data classification node is called an item node here.

The item node shows each data item of "ProcessDataIn" that is the data of the IO-Link device and "ProcessDataOut" and "Parameter". Data with the IO-Link device is actually exchanged by the item node.

Moreover, the method that can be used changes according to from which data classification node the item node was added.

The followingTable2-4[Ni] and the method that can be used are described.

Table2-4Method list that can be used whenever classifying it

Classification of data classification node	Method by which item node can be used	Remarks
ProcessDataIn	・CaoFile::get_Value ・CaoFile::Execute(“GetParameterList”) ・CaoFile::Execute(“GetType”) ・CaoFile::Execute(“GetRange”) ・CaoFile::Execute(“GetUnit”)	---
ProcessDataOut	・CaoFile::get_Value ・CaoFile::put_Value ・CaoFile::Execute(“GetParameterList”) ・CaoFile::Execute(“GetType”) ・CaoFile::Execute(“GetRange”) ・CaoFile::Execute(“GetUnit”)	---
Parameter	・CaoFile::get_Value ・CaoFile::put_Value ・CaoFile::Execute(“GetParameterList”) ・CaoFile::Execute(“GetType”) ・CaoFile::Execute(“GetRange”) ・CaoFile::Execute(“GetUnit”)	- Get and put might not be able to be executed according to the access kind. Please refer to Fileclass for details.

The sample of PacScript until item "SampleItem" of ProcessDataIn of device "SampleDevice" is added is shown as follows.

Usage example

```
Sub Main
        Dim caoCtrl as Object
        Dim caoSampleDevice as Object
 Dim caoProcessDataIn as Object
        Dim caoSampleItem as Object

        '① .  IO-Link Maseter and connection
        caoCtrl=cao.AddController("IOLink","caoProv.Balluff.IOLink","",
"server=192.168.1.100, Path=SampleMaster.xml")
        '② .  IO-Link device and connection
        caoSampleDevice=caoCtrl.AddFile("SampleDevice","PortNo=0, Path=SampleDev.xml")
        '③ .   “ProcessDataIn” is added.
        caoProcessDataIn = caoSampleDevice.AddFile("ProcessDataIn")
        . The item ..'④.. is added.
        caoSampleItem = caoProcessDataIn.AddFile("SampleItem")

End Sub
```

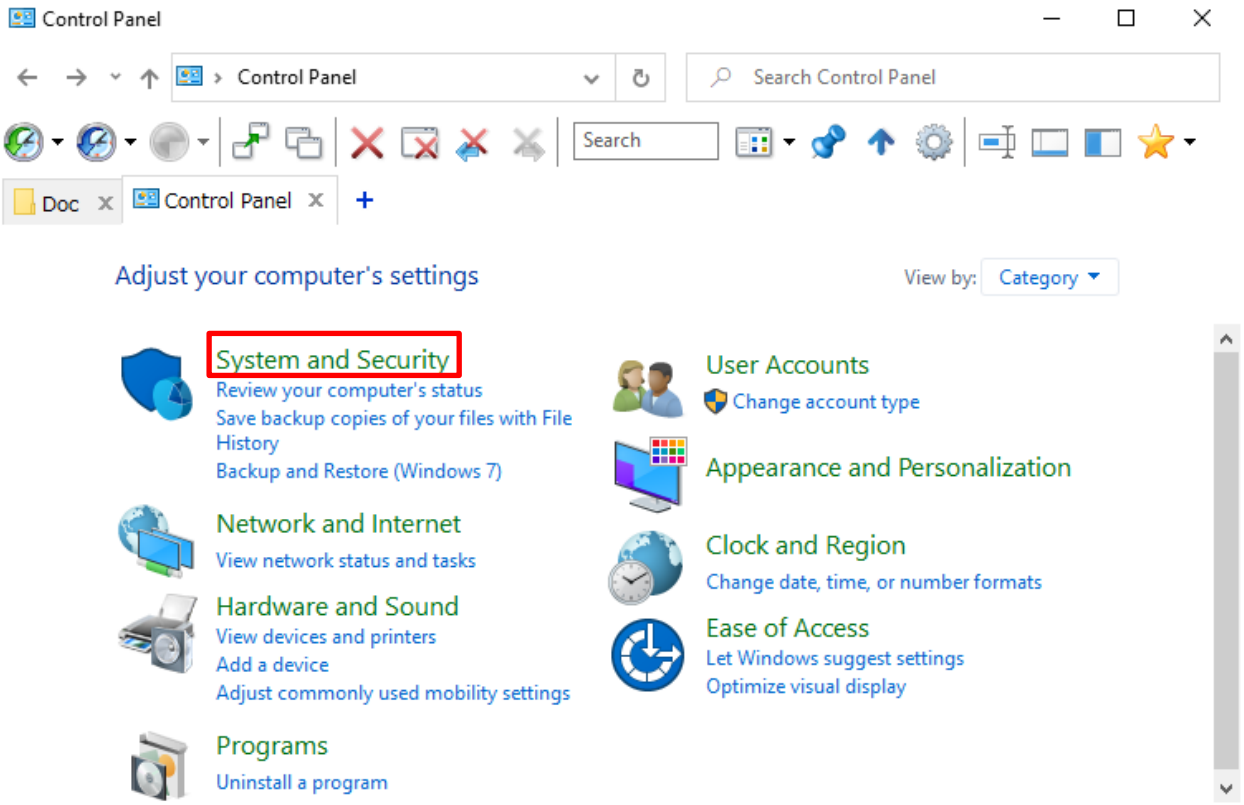
2.3. Notes on connecting IO-Link providers and IO-Link masters

If the client PC is Windows, the connection fails if Windows firewall is not configured correctly when connecting to IO-Link Master using IO-Link Provider. The following describes how to configure Windows firewalls.

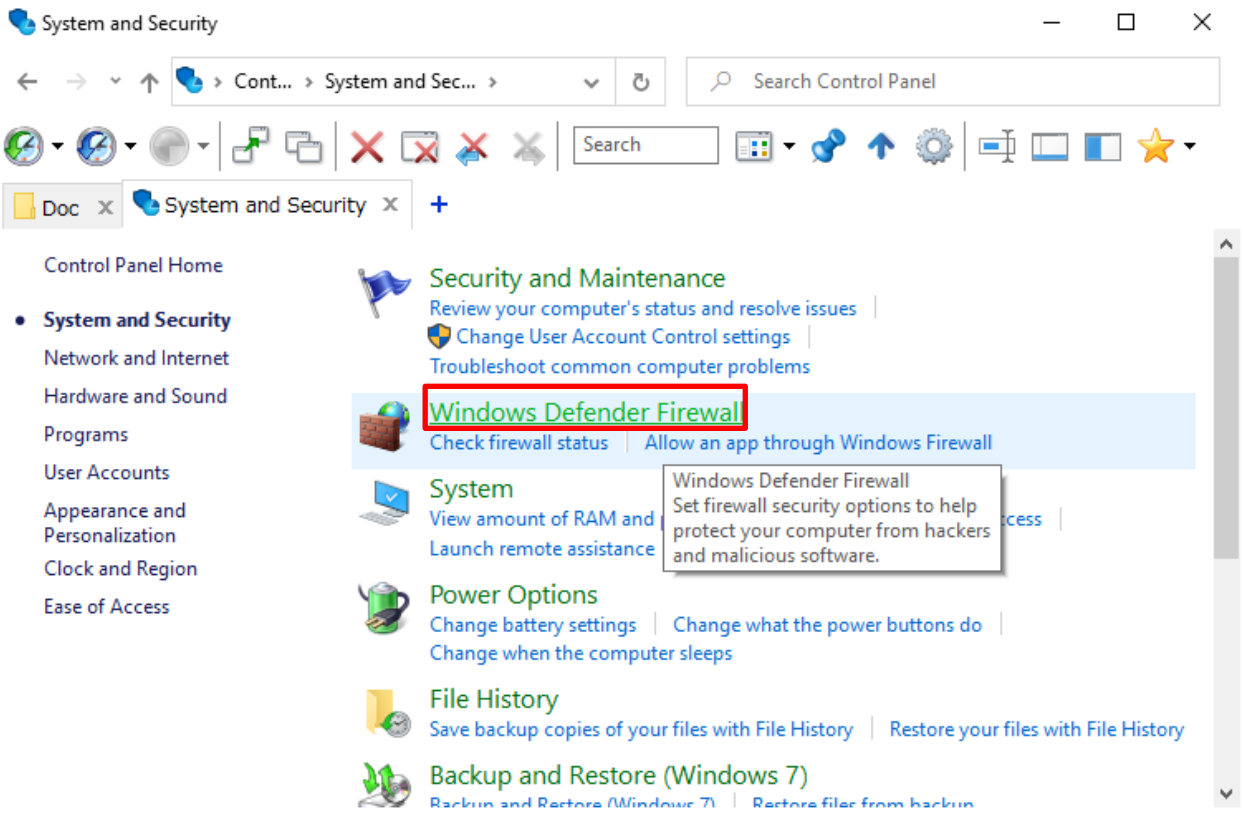
Step

① Open [Control panel] from the Start menu.

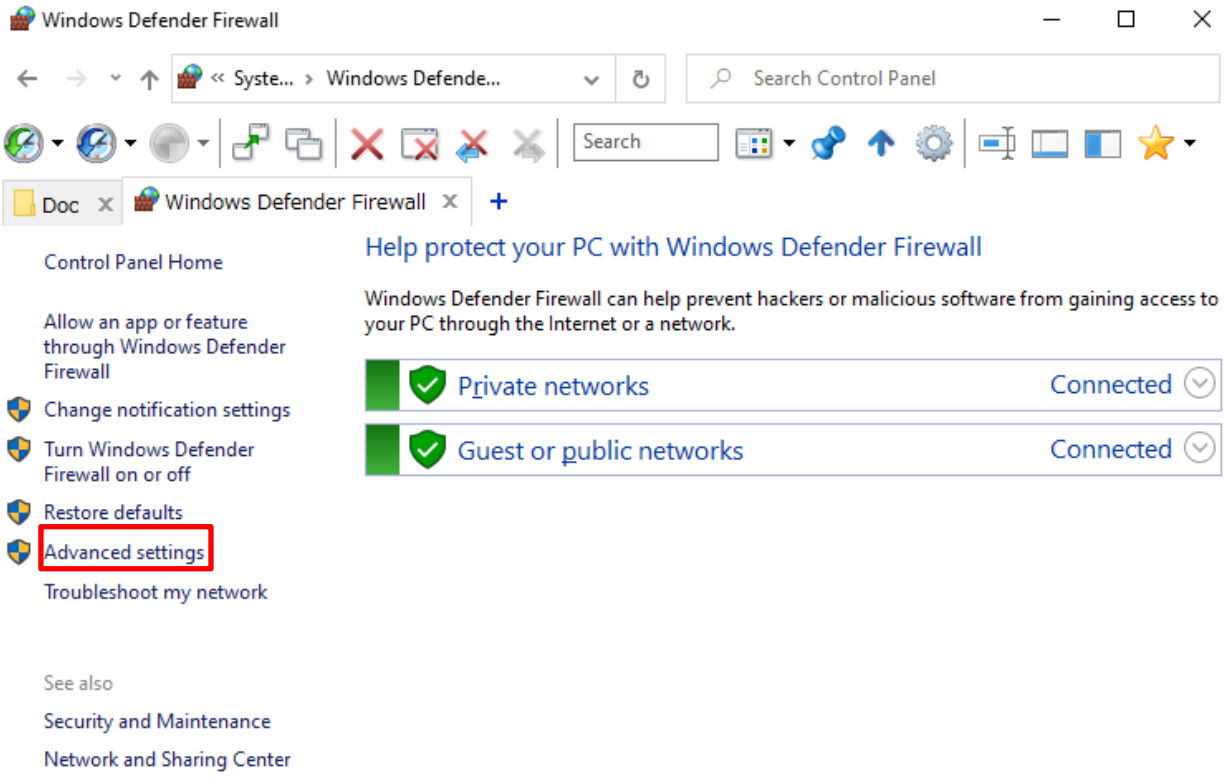
② Open [System and Security].



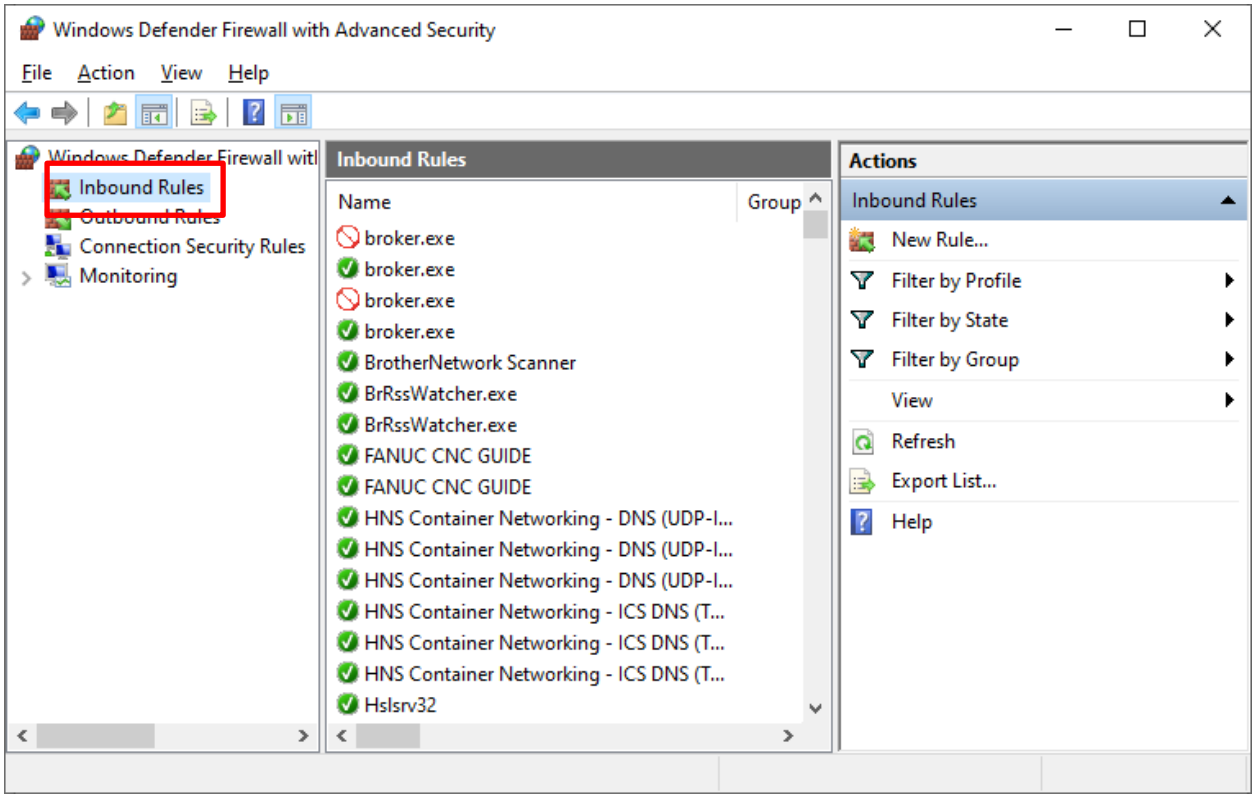
③ Open [Windows Defender Firewall].



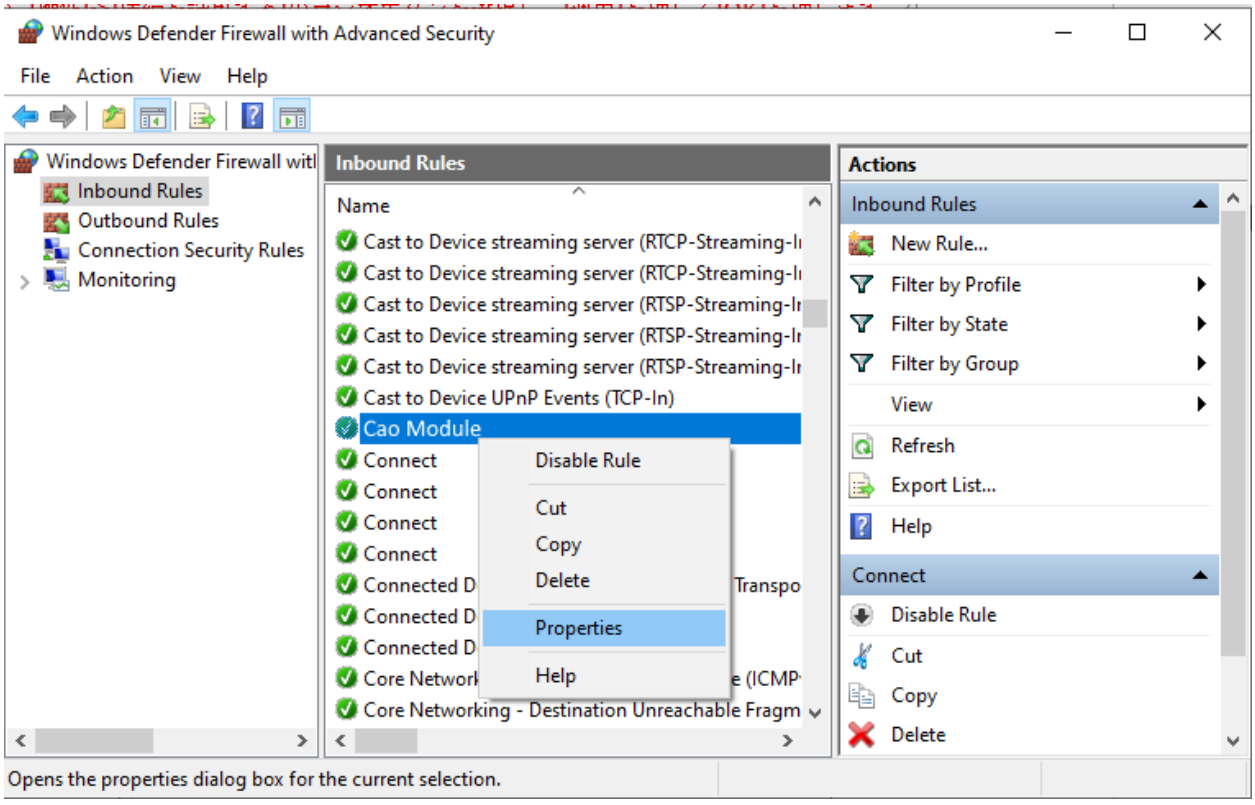
④ Open [Advanced settings].



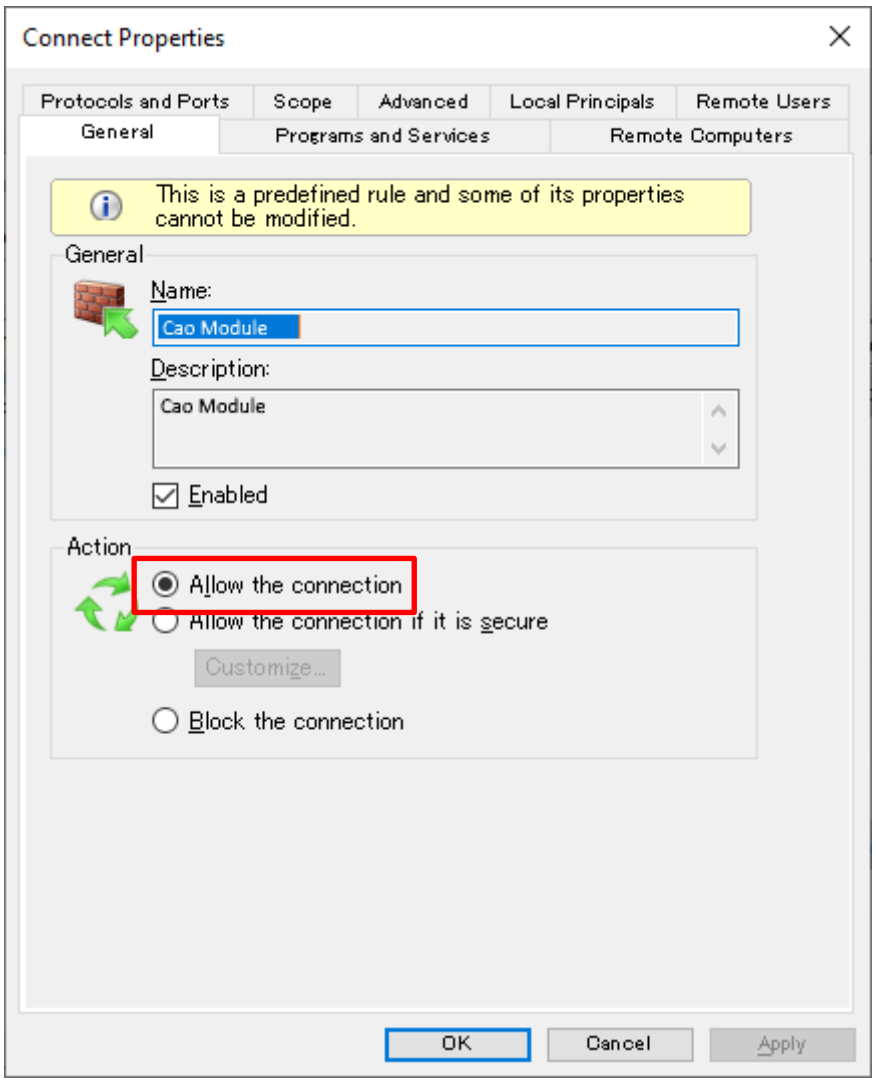
⑤ Click [Inbound Rules].



⑥ Open the [CAO Module] properties.



⑦ Select the [Action]-> [Allow the connection] radio button, press [Apply], and then press [OK]..



⑧ The setting procedure is above. When using a domain network, allow two [CAO Module] connections in [Domain] in the profile, and when using a public network, allow two [CAO Module] connections in [Public].

2.4. Method property

2.4.1. CaoWorkspace::AddController method

When the Controller object is generated, the IOLM file that describes Internet Protocol address of mastering and information on mastering is specified.

The specification of AddController is shown as follows.

Format

AddController

```
(
        "< controller name >"          // Controller name (arbitrariness)
         "CaoProv.BALLUFF.IOLink",     // Provider name (fixation)
        "< machine name >"                    //   Provider   execution
machine name (unused)
        "< option >"           // Optional character string
```

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

Table2-5Optional character string of CaoWorkspace::AddController

Option	Indispensability	Explanation	Range of value	Default value
Server= < Internet Protocol address of mastering >	-	Internet Protocol address of mastering to be connected is specified.	Up to 255 characters	----
Path= < passing of IODM file >	-	Passing to the IOLM file is specified. About details of the specification methodPlease refer to 2.4.1.1.	----	----

When 192.168.1.100 and the read IOLM file are Sample.xml, Internet Protocol address of the connected mastering inputs specification and optional a character string as follows.

Usage example

```
Dim caoCtrl as Object
caoCtrl=cao.AddController("Sample","CaoProv.Balluff.IOLink","","server=192.168.1.100,path=Sample.xml")
```

2.4.1.1. Path is optional.

The method of specifying the IOLM file set to optional Path has two (the method to specify by the relative path and the method to specify by the absolute path). Each specified details of the relative path and the absolute path is described as follows.

- Relative path

Please put the IOLM file of the read object on the following directories.

(* The IODM file for EIP being offered by the Balluff Co. in June, 2017 is pre-installed.)

·Bin/XML

The Path option specification when the IODD file "Balluff-BNI-EIP-507-005-Z040-20151110-IOLM1.3.xml" is placed in the above directory is put example is shown below.

Usage example

```
Path=Balluff-BNI-EIP-507-005-Z040-20151110-IOLM1.3.xml
```

- Absolute path

Please specify the absolute path of the read IOLM file.

Usage example

```
Path=C:/User/Desktop/Balluff-BNI-EIP-507-005-Z040-20151110-IOLM1.3.xml
```

2.4.2. CaoController::AddVariable method

The CaoVariable object is generated from CaoController. To the variable identifierOnly the shown Table2-9 variable identifier can be used.

The specification of AddVariable is shown as follows.

Format

```
AddVariable
(
        "< variable identifier >"       // Variable identifier
         "",            // Optional character string (unused)
)
```

2.4.3. CaoController::GetVariableNames method

The shown variable identifier list of Table2-9 is acquired.

2.4.4. CaoController::AddFile method

When the File object is generated, the port number of mastering from which the device is connected and passing to the IODD file are specified in the option.

The specification of AddFile is shown as follows.

Format

```
AddFile
(
        "Sample",       // File name (arbitrariness)
        "PortNo=< port number >, Path=<IODD File path >"  // Optional character string
);
```

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

Table2-6Optional character string of CaoController::AddFile

Option	Indispensability	Explanation	Range of value	Default value
PortNo=<port number>	-	The port number of mastering with which the IO-Link device equipment is connected is specified. Moreover, when a specified port doesn't correspond to IO-Link, it becomes an error.	Number of maximum ports of 0~ mastering	----
Path=<IODD file path>	-	Passing to the IODD file is specified. About details of the specification method. Please refer to 2.4.4.1.	----	----

If the port number of the device to be connected is port 0 and the IODD file "ExampleDevice.xml" to be read is placed on "C:¥Users¥user name¥Desktop", the specified option character is as shown in ② below.

Usage example

```
Sub Main
        Dim caoCtrl as Object
        Dim caoSampleDevice as Object
 Dim caoProcessDataIn as Object
        Dim caoSampleItem as Object

        '①．  IO-Link Maseter and connection
        caoCtrl=cao.AddController("IOLink","caoProv.Balluff.IOLink","",
        "server=192.168.1.100, Path=SampleMaster.xml")
        '②．  IO-Link device and connection
        caoSampleDevice=caoCtrl.AddFile("SampleDevice","PortNo=0,
        Path=C:/Users/Desktop/ExampleDevice.xml")

End Sub
```

2.4.4.1. Path is optional.

The method of specifying the IODD file set to optional Path changes according to PC or RC8 the execution environment. When the execution environment is a case of PC, and RC8, each specified details is described as follows. See Appendix B for the IODD file.

- PC environment

·Relative path specification

→ Bin¥XML

Please put the IODD file of the reading object on the above-mentioned directory, and specify the file name.

The Path option specification when the IODD file "Balluff-BNI_IOL-802-102-Z036-20150730-IODD1.1.xml" is placed in the above directory is put example is shown below.

Usage example

Path= Balluff-BNI_IOL-802-102-Z036-20150730-IODD1.1.xml

·Absolute path specification

Please specify the absolute path of the IODD file of the reading object.

Usage example

Path= C:/User/Desktop/Balluff-BNI_IOL-802-102-Z036-20150730-IODD1.1.xml

- RC8 environment

Please forward the IODD file to be read to RC8 with WINCAPS Ⅲ.

Afterwards, when the forwarded IODD file name is specified for optional Path, the forwarded IODD file is read.

The Path option specification when IODD "Balluff-BNI_IOL-802-102-Z036-20150730-IODD1.1.xml" is filed in RC8. is forwarded example is shown below.

Usage example

Path= Balluff-BNI_IOL-802-102-Z036-20150730-IODD1.1.xml

2.4.5. (Device node)CaoFile::AddFile method

The caoFile object is generated from CaoFile. The acquired value is decided from the device connected with mastering by specifying the file name.

The specification of CaoFile::AddFile is shown as follows.

Format

```
CaoFile::AddFile
(
        "< file name >" // Variable identifier
         "",                    // Optional character string (unused)
)
```

About the variable identifier that can be used CaoFile::GetFileNames Only the file name that can be acquired in the method can be used.

2.4.6. (Data classification node)CaoFile::AddFile method

The caoFile object is generated from CaoFile. The acquired value is decided from the device connected with mastering by specifying the file name.

The specification of CaoFile::AddFile is shown as follows.

Format

```
CaoFile::AddFile
(
        "< file name >" // Variable identifier
         "",                    // Optional character string (unused)
)
```

About the variable identifier that can be used CaoFile::GetFileNames Only the file name that can be acquired in the method can be used.

以下にオプション文字列に指定する文字列を示します.

Table 2-7 (Data classification node)CaoFile::AddFile option string

Option	Indispensability	Explanation	Range of value	Default value

IsGetName=<DataTpe>	-	Only the character string acquired with GetFileNames() can be specified for the AddFile method for an added file class name.	It is specified whether to apply the string to the value in XML and whether to turn over and to acquire. TRUE : BSTR FALSE : not BSTR	TRUE

2.4.7. CaoFile::getFileNames method

The list of the file name that can be added is acquired from the read IODD device file. The content of the list changes with the read IODD device file.

2.4.8. CaoFile::Execute property

Information on the item etc. can be acquired by using the Execute method.
Please refer to Chapter 2.5.

Format

```
caoFile::Execute
(
      <bstrCommand:BSTR>          // In command name
      [,<vntParam:VARIANT>]       // In parameter
      [,<pVal:VARIANT>]])         // In execution result
)
```

2.4.9. CaoFile::put_Value property

Data is set to the device by the file class of the made item node.
- Please set the set data by the character string.
About the classification of the item node and the correspondence of the attributeTable2-16Please refer to [wo].

Moreover, the CaoFile::put_Value property :. caoFile::Execute(“GetParameterList”)When the command can be executed, and the parameter acquire the list (It is not VT_EMPTY), only the acquired parameter character string can be specified. When the list of the parameter cannot be acquired in the CaoFile::Execute(“GetParameterList”) command, the value can be set by the character string. In the value that can be set, there might be a setting range according to the item. The range that can be setCaoFile::Execute(“GetRange”)It is possible to acquire it in the command. Please set the value within the range when the range exists. It becomes an error the value ..outside the range.. if it sets it.

Moreover, when the range doesn't existCaoFile::Execute(“GetType”)The type acquired in the command becomes the range of the value that can be set as it is. When the value beyond the limits of the type acquired is set, it becomes an error.

2.4.10. CaoFile::get_Value property

Data is acquired from the device, and the acquired data is returned by the character string by the file class of the made item node.

The CaoFile::get_Value method is executable also by the item node of which classification. About the classification of the item node and the correspondence of the attributeTable2-16Please refer to [wo].

2.4.11. CaoVariable::get_Value property

Data is acquired from the connected mastering according to the specified variable identifier. Details please refer to Table2-9.

2.5. Execute command list

The Execute command list that can be used in Table2-8 and the File class is described.

Table2-8Execute command list

Command name	Explanation	Details
GetAccessType	The access kind of the parameter is acquired.	P.26
GetUnit	The unit character string of the item is acquired.	P.27
GetType	The type of the item is acquired.	P.27
GetRange	The range of the value that can be set to the item is acquired.	P.28
GetParameterList	The value that can be set to the item is acquired.	P.28

GetAccessType

The access kind of the parameter is acquired. The classification of the data classification node : this command.

It is possible to use it only at Parameter. When this command is used by other nodes, it becomes an error.

Item	Type explanation	
vntParam	None	
pVal	VT_UI1	The access kind of the parameter is acquired. 0 : Read / Write 1 : Read Only 2 : Write Only

Usage example

```
CaoFile.Execute( “GetAccessType”)
```

GetUnit

The unit of the item is acquired in the character string. This command can use only the item node. It becomes an error when using it by other nodes.

Item	Type explanation	
vntParam	None	
pVal	VT_BSTR	The unit of the item is acquired in the character string. Example.) "℃" and "msec", etc.
	VT_EMPTY	The unit doesn't exist.

Usage example

```
CaoFile.Execute("GetUnit")
```

GetType

The type of the item is acquired. This command can use only the item node. It becomes an error when using it by other nodes.

Item	Type explanation	
vntParam	None	
pVal	VT_UI2	The value of the type of the item is acquired. The value of VARENUM of the acquired type is set to the value of pVal. 16 : VT_I1 2 : VT_I2 3 : VT_I4 20 : VT_I8 17 : VT_UI1 18 : VT_UI2 19 : VT_UI4 21 : VT_UI8 4 : VT_R4 5 : VT_R8 8 : VT_BSTR 11 : VT_BOOL

Usage example

```
CaoFile.Execute( “GetType")
```

GetRange

The range of the value that can be set to the item is acquired. This command can use only the item node. It becomes an error when using it by other nodes.

Item	Type explanation		
vntParam	None		
pVal	VT_ARRAY | VT_UI8		
	0	VT_UI8	The minimum value of the value that can be set is acquired.
	1	VT_UI8	The maximum value of the value that can be set is acquired.
	VT_ARRAY | VT_I8		
	0	VT_I8	The minimum value of the value that can be set is acquired.
	1	VT_I8	The maximum value of the value that can be set is acquired.
	VT_ARRAY | VT_R4		
	0	VT_R4	The minimum value of the value that can be set is acquired.
	1	VT_R4	The maximum value of the value that can be set is acquired.
	VT_EMPTY		The range doesn't exist.

Usage example

```
CaoFile.Execute("GetRange")
```

GetParameterList

The value that can be set to the item is acquired. When the value can be acquired, the value that can be set to the CaoFile::put_Value property can set only the value that was able to be acquired in the GetParameterList command.

This command can use only the item node. It becomes an error when using it by other nodes.

Item	Type explanation		
vntParam	None		
pVal	VT_ARRAY | VT_BSTR		
	i	VT_BSTR	The value that can be set to the item is acquired.
	VT_EMPTY		The parameter doesn't exist.

Usage example

```
CaoFile.Execute("GetParameterList")
```

2.6. Variable list

2.6.1. Controller class

The followingTable2-9The variable list that can be used in [ni] and the controller class is described.

Table2-9Controller class variable list

Variable identifier	Data type	Explanation	Attribute	
			get	put
@MAKER_NAME	**VT_BSTR**	Manufacturer name (BALLUFF)	✓	-
@VERSION	**VT_ARRAY | VT_VARIANT**	- The version and model number information on the firmware are acquired. - It is stored in the array in the following order. VT_BSTR: It consists of following parameter A, B, C, D, E, and F. Character string A.B.C/D.E.F VT_UI1: The main firmware version (A) VT_UI1: Detailed firmware version (B) VT_UI1: Firmware revision composition (C) VT_UI1: The main revision of mastering memory (D) VT_UI1: Detailed revision of mastering memory (E) VT_UI1: Composition of mastering memory revision (F)	✓	-
@LAST_EVENT	**VT_ARRAY | VT_VARIANT**	- Event information on mastering is acquired. - It is stored in the array in the following order. VT_UI2: Number to DLL of event multiplication VT_UI2: Port number where event is generated VT_UI2: Event code (Refer to Table2-10.) VT_UI1: Instance (of event refer to Table2-11.) VT_UI1: Event mode (Refer to Table2-12.) VT_UI1: Event type (Refer to Table2-13.) VT_UI1: (of event mode 2) VT_UI1: The event issues it from the IO-Link mastering. (whether done by all means Refer to Table2-15.)	✓	-

Usage example

When you acquire the last event of mastering

CaoController.AddVariable(“@LAST_EVENT”,””);

- The variable identifier can be input without the capital letter and the small letter relation.

AddVariable(“@last_event”,””);

Table2-10Value and explanation to be able to acquire event code

Acquisition value	Explanation
2	Frame error reception
16	Device connection release (It releases the connection, disconnect).
26	Unexpected error that detects different sensor
27	Retrying is detected.
30	It is a shunt detection on C/Q (common) line.
31	The error is in the sensor supply.
32	The error is in the actuator supply.
33	The error is in the power supply to the IO-Link master.
34	When the port resets it, the event is sent.
35	The fallback success and the device are SIO modes.
36	Use preparation completion of device
40	Because CRC was normal, nothing was done though the data storage was completed.
50	Parameter download completion
51	Parameter up-loading completion
64	The difference is in the input process data length.
65	The difference is in the output process data length.
66	The difference is in revision of the device (revision).
67	The difference is in vendorID of the V1.1 sensor (vender ID).
68	The difference is in DeviceID of the V1.1 sensor (device ID).
69	The difference is in VendorID of the V1.0 sensor.
70	The difference is in DeviceID of the V1.0 sensor.
71	The difference is in SerialNumber.
72	The cycle time is not suitable.

Table2-11Value and explanation to be able to acquire instance of event

Acquisition value	Explanation
0	Uncertain instance
1	Physical layer of instance
2	Data layer of instance
3	Instance application layer
4	Instance application

Table2-12Value and explanation to be able to acquire event mode

Acquisition value	Explanation
0	Single message or warning
1	Error generation
2	Error cancellation

Table2-13Value and explanation to be able to acquire event type

Acquisition value	Explanation
0	Message
1	Warning
2	Error

Table2-14Value and explanation to be able to acquire event mode 2

Acquisition value	Explanation
0	Abnormality
1	The process data is normal.

Table2-15Value and explanation to be able to acquire right or wrong whether event was issued from IO-Link mastering

Acquisition value	Explanation
0	It is not issued.
1	It was issued.

2.6.2. File class

The followingTable2-16The attribute is described the file class of [ni] and the item node.

Table2-16Attribute at each item node file class classification

Classification of data classification node	Access kind (*)	Explanation	Attribute	
			get	put
ProcessDataIn	-----	The data of the process data Inn of the device is acquired.	✓	-
ProcessDataOut	-----	The value of the process data out of the device is read and written.	✓	✓
Parameter	**Read / Write(0)**	It is a parameter of the device, and reading and writing is a possible item.	✓	✓
	Read Only(1)	It is a parameter of the device, and only reading is a possible item.	✓	-
	Write Only(2)	It is a parameter of the device, and only writing is a possible item.	-	✓

- About the access kind CaoFile::Execute("GetAccessType") It is possible to acquire it in the command.

2.7. Error code

In this provider, an original error code exists. Details are the following. Please refer to Table2-17.

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

Table2-17Original error code table

Error number	Explanation
0x80110001	It is not set that indispensability is optional.
0x80110002	It failed in reading the unit definition file. Please confirm whether the unit definition file exists in predetermined directory (Bin¥XML).
0x80110003	"Path =" It failed to open the XML file from the character string set in the option. Please confirm the specified file is whether it is XML file format.
0x80110004	The read XML file is not correct. Do you confirm the IODM file or the IODD file correct?
0x80110005	The specified port number doesn't exist.
0x80110006	The specified port number doesn't correspond to IO-Link.
0x80110007	The port number is not specified.
0x80110008	It is not possible to convert it into the specified type.
0x80110009	The schema version of the specified IODD file is off the subject.

Moreover, this provider does the mask with 0x8010**** and returns the error made by the Balluff company API"IOLUDPIF20". Please refer to "DLL manual 20160608. pdf" of the interface manual of the Balluff Co. for the error from API. (Appendix A, The one that the part of the error code was quoted is described to the appendix.)

3. Sample program

This chapter describes the sample of PacScript that does the data communication with the device.

Precondition:
・The IO-Link mastering used is assumed to be "BNI EIP-507-005-Z040".
・The IODM file used
It is assumed "Balluff-BNI-EIP-507-005-Z040-20151110-IOLM1.3.xml".
・Internet Protocol address of the mastering equipment is assumed to be "192.168.1.100".
・The number of the port where each device is connected is assumed to be the 0th.

① Schmalz-SCPSi_V2

Schmalz-SCPSi_V2 is communicated with this provider, and the sample program that does the data writing to item "Vacuum" of ProcessDataOut is described as follows.

Precondition: ·The IO-Link device used is assumed to be "Schmalz-SCPSi_V2". ·The IODD file used is assumed to be "Schmalz-SCPSi_V2-20140829-IODD1.1.xml".

```
Sub Main
        Controller ..'.. addition
        Dim caoCtrl as Object
        caoCtrl = cao.AddController("IOLink", "CaoProv.Balluff.IOLink", "",
"server=192.168.1.100,path=Balluff-BNI-EIP-507-005-Z040-20151110-IOLM1.3.xml")

        Device node ..'.. addition
        Dim caoFileSCPSi as Object
        caoFileSCPSi = caoCtrl.AddFile("Schmalz", "PortNo=0,
Path=Schmalz-SCPSi_V2-20140829-IODD1.1.xml")

        Data classification node ..'.. addition (ProcessDataOut)
        Dim caoFileSCPSi_ProcessDataOut as Object
        caoFileSCPSi_ProcessDataOut = caoFileSCPSi.AddFile("ProcessDataOut")

        Item node ..'.. addition
        Dim caoFileSCPSi_ProcessDataOut_Vacuum as Object
        caoFileSCPSi_ProcessDataOut_Vacuum =
        caoFileSCPSi_ProcessDataOut.AddFile( “Vacuum")

        ..'.. writing of value
        Delay 1000
        caoFileSCPSi_ProcessDataOut_Vacuum.Value = "-1"
        Delay 1000
        caoFileSCPSi_ProcessDataOut_Vacuum.Value = "0"
End Sub
```

② BNI_IOL-802-102-Z036

BNI_IOL-802-102-Z036 is communicated with this provider, the data of ProcessDataOut is written, and the sample program that makes the light of the device shine in orange is described as follows.

Precondition:
・The connected IO-Link device is assumed to be "BNI_IOL-802-102-Z036".
・The IODD file used is assumed to be
"Balluff-BNI_IOL-802-102-Z036-20150730-IODD1.1.xml".

```
Sub Main
        Controller ..'.. addition
        Dim caoCtrl as Object
        caoCtrl = cao.AddController("IOLink", "CaoProv.Balluff.IOLink", "",
"server=192.168.1.100,path=Balluff-BNI-EIP-507-005-Z040-20151110-IOLM1.3.xml")

        Device node ..'.. addition
        Dim caoFileTowerLigth as Object
        caoFileTowerLigth = caoCtrl.AddFile("TowerLight", "PortNo=0,
        Path= Balluff-BNI_IOL-802-102-Z036-20150730-IODD1.1.xml")

        Data classification node ..'.. addition (ProcessDataOut)
        Dim caoFileTowerLigth_ProcessDataOut as Object
        caoFileTowerLigth_ProcessDataOut = caoFileTowerLigth.AddFile("ProcessDataOut")

        Item node ..'.. addition
        Dim caoFileTowerLigth_ProcessDataOut_OperatingMode as Object
        Dim caoFileTowerLigth_ProcessDataOut_BackColor as Object
        caoFileTowerLigth_ProcessDataOut_OperatingMode =
        caoFileTowerLigth_ProcessDataOut.AddFile( “Operating mode")
        caoFileTowerLigth_ProcessDataOut_BackColor =
        caoFileTowerLigth_ProcessDataOut.AddFile( “Segment 1 color / Background color")

        ..'.. writing of value
        Delay 1000
        caoFileTowerLigth_ProcessDataOut_OperatingMode.Value = "Runlight mode"
        Delay 1000
        caoFileTowerLigth_ProcessDataOut_BackColor.Value = "Orange"
```

End Sub

③ BAE-PS-XA-1W

The data of ProcessDataIn as follows. this provider and communicate BAE-PS-XA-1W「Input Voltage」

The sample program that drinks and reads [atai] is described.

Precondition:
・The connected IO-Link device is assumed to be "BAE-PS-XA-1W".
・The IODD file used is assumed to be "Balluff-BAE-PS-XA-1W-24-038-607-20151119-IODD1.1.xml".

```
Sub Main
        Controller ..'.. addition
        Dim caoCtrl as Object
        caoCtrl = cao.AddController("IOLink", "CaoProv.Balluff.IOLink", "",
"server=192.168.1.100,path=Balluff-BNI-EIP-507-005-Z040-20151110-IOLM1.3.xml")

        Device node ..'.. addition
        Dim caoFileSupply as Object
        caoFileSupply = caoCtrl.AddFile("Supply", "PortNo=0,
        Path= Balluff-BAE-PS-XA-1W-24-038-607-20151119-IODD1.1.xml")

        Data classification node ..'.. addition (Parameter:Input Voltage)
        Dim caoFileSupply_InputVoltage as Object
        caoFileSupply_InputVoltage = caoFileSupply.AddFile("Input Voltage")

        Item node ..'.. addition
        Dim caoFileSupply_InputVoltage_Item as Object
        caoFileSupply_InputVoltage_Item = caoFileSupply_InputVoltage.AddFile("Item")

        ..'.. reading of value
        Dim Result as Object
        Delay 1000
        Result = caoFileSupply_InputVoltage_Item.Value
End Sub
```

Appendix A. API error codes

The one that the error code value of API was quoted is described from interface manual "DLL manual 20160608. pdf" of the Balluff Co. as follows.

RETURN_FIRMWARE_NOT_COMPATIBLE -16

It is necessary to update the firmware because there is a function that cannot be executed.

RETURN_FUNCTION_NOT_IMPLEMENTED -13

A specified function cannot be executed by the connected IO-Link mastering.

RETURN_STATE_CONFLICT -12

The specified function cannot be executed by a present IO-Link mastering setting.

RETURN_WRONG_COMMAND -11

A wrong answer returned from the IO-Link mastering to the instruction command.

RETURN_WRONG_PARAMETER -10

The function parameter is invalid.

RETURN_WRONG_DEVICE -9

The device name is connected and it is wrong or connected an unsupported device.

RETURN_NO_EVENT -8

There is no event though the event reading was ordered.

RETURN_UNKNOWN_HANDLE -7

It is processing of an uncertain function.

RETURN_UART_TIMEOUT -6

There is no answer to the command that becomes a time-out.

RETURN_CONNECTION_LOST -5

Mastering was detached while communicating.

RETURN_OUT_OF_MEMORY -4

There is no memory that can be used.

RETURN_DEVICE_ERROR -3

The error occurred while accessing the UDP driver.

RETURN_DEVICE_NOT_AVAILABLE -2

The device cannot temporarily be used.

RETURN_INTERNAL_ERROR -1

Please restart the program for the internal library error.

RETURN_OK 0

Normal completion

RESULT_STATE_CONFLICT 1

The instruction command is not correct compared with the present setting.

RESULT_NOT_SUPPORTED 2

This device doesn't correspond to the command.

RESULT_SERVICE_PENDING 3

Service is being reserved. Please do not execute new service until the service of the pending state is completed.

RESULT_WRONG_PARAMETER_STACK 4

The parameter was refused to the IO-link mastering.

Appendix B. IODD file

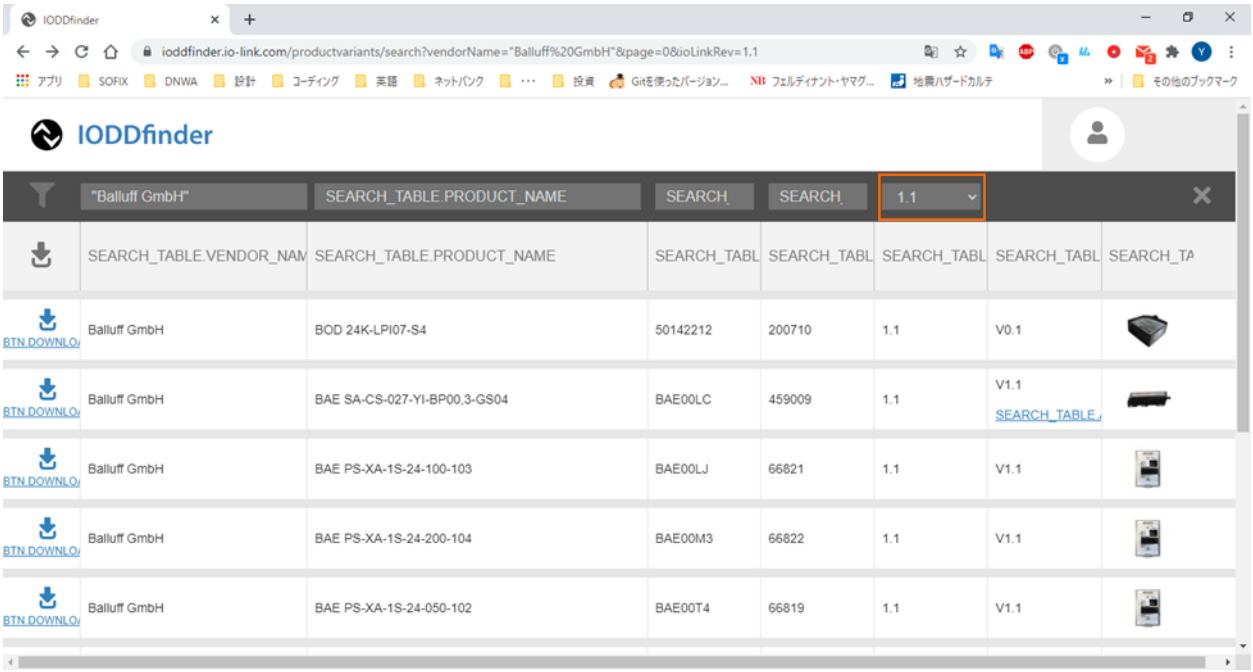
How to Obtain IODD Files

To connect to a IO-Link device, a IODD(IO Device Description) file is required. Download the specified file from the following URL and use it.

https://ioddfinder.io-link.com

Note that the schema version supported by the provider is 1.1, so be careful when downloading.

The following illustration shows an example of searching for IODD files from the above URLs.



How to distinguish IODD files

The root element of IODD file is <**IODevice**>, and the "xsi:schemaLocation" attribute of the <**IODevice**> element is described as "**http://www.io-link. com/IODD/2010/10 IODD1.1.xsd**". If an error occurs even though a IODD file is specified, check the contents of the specified file and check whether the file is normal.

Also, "IODD-StandardDefinitions.xml", which is included by default during provider installation, is not a IODD file, but a definition file referenced by all IODD files. Do not specify this as IODD option.