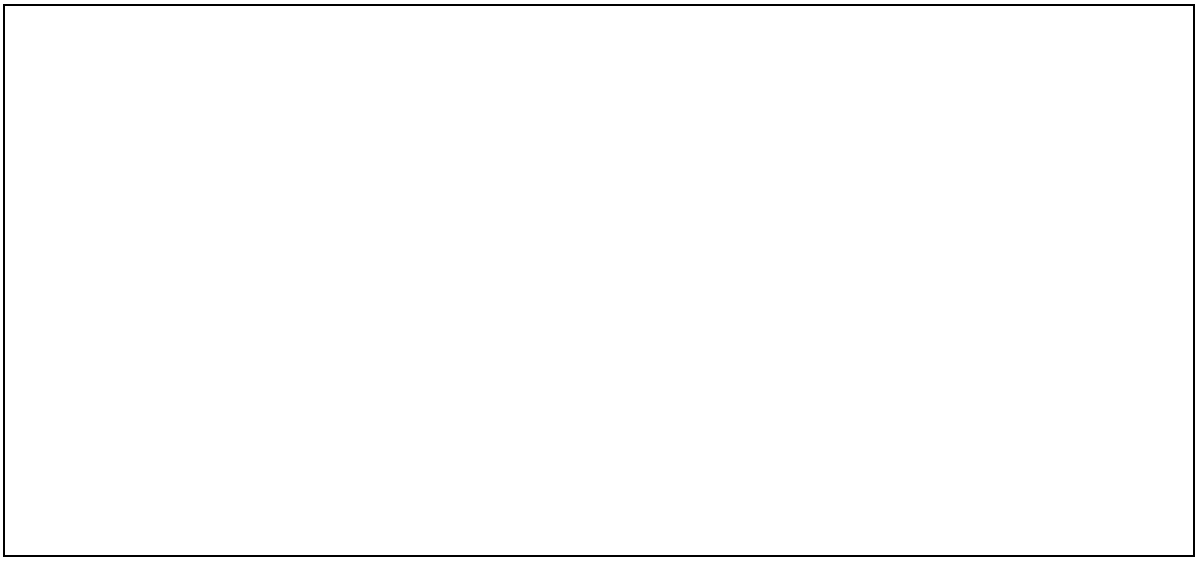
DENSO
RFID Scanner UR40 providers

Version 1.2.0

User's Guide

August 23, 2024



Revision history

Version	Date	Content
1.0.0	2019-12-02	First edition.
1.0.1	2022-07-13	Corrected the message content of the Error message event. Corrected a typo.
1.1.0	2022-10-18	Add Write,WriteUII,Lock,LockUII,Kill,KillUII command. Correction of typographical errors.
	2024-07-23	Tag specifications are added in the appendix. Corrected errors.
1.2.0	2024-08-23	Add ConnectionLog option.

[Target model]

Model	Remark
UR40-H-ERU	
UR50-H-ERU	

[Operation Verification Model]

Model	Version	Note
UR50-H-ERU	Ver. 1.11.00	

Table of Contents

1. Introduction

This document is a user's guide for providers obtaining RFID tags from RFID Scanner's UR40/UR50 family. Figure 1-1 Configuration Diagram shows the overall configuration of this provider and device. Hereafter, this provider is referred to as the UR40 provider, and RFID Scanner is referred to as the scanner. The UR40 provider communicates with the scanner.

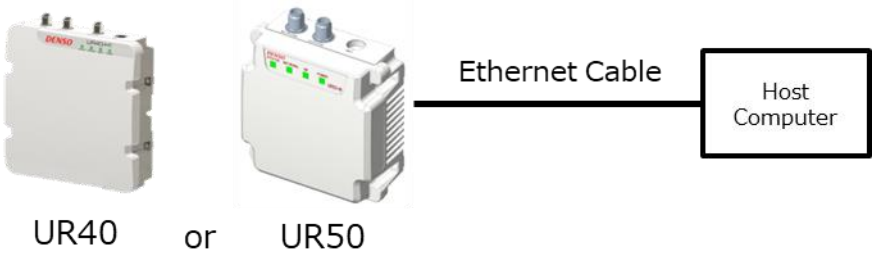


Figure 1-1 Configuration Diagram

When controlling commands to the scanner, set the scanner to TCPServer.
Set TCPClient for transferring scanned data. Providers act as TCPClient for command-control and TCPServer for read-data transfers. Figure 1-2 Connection overview provides an overview of the connections.

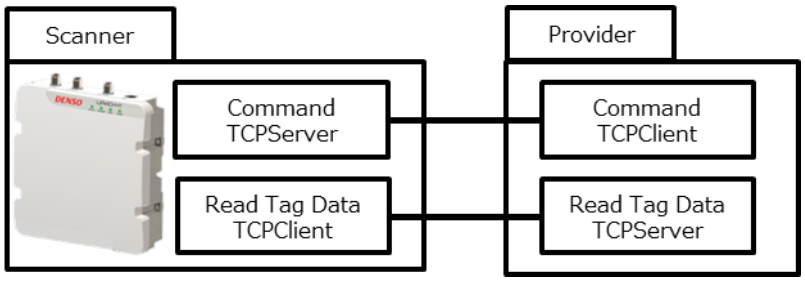


Figure 1-2 Connection overview

Figure 1-3 shows the correspondence between this provider and device.
(※For example: Not all are shown.)

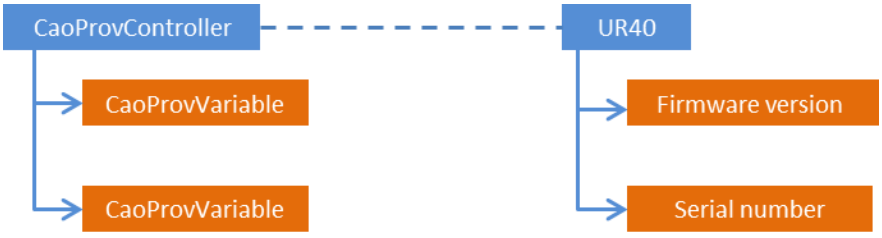


Figure 1-3 Correspondence between provider configuration and device information

1.1. Development environment assumed by this document

The development environment of Windows and PC can be developed in any programming environment that

supports Component Object Model (COM-Component Object Model).

1.2. Source of informative information

All programming examples in this document are presented in Visual Basic for Applications, but they can be developed in a variety of programming languages, including C++, Java, and .NET. See the ORiN2 Programming Guide for instructions.

The ORiN 2 Programming Guide corresponds to the following files in the ORiN2 SDK installation folder:

- ORiN2¥CAO¥Doc¥ORiN2_ProgrammersGuide_<lang>.pdf

※Replace <lang> with the language strings for each language environment.

This chapter provides examples of the basic ORiN2, COM/DCOM skills and techniques required to develop applications that use providers.

2. Setting Up the Environment for Application Development

2.1. Connecting the Scanner to the Client PC

Before connecting to the scanner, connect to the client PC while the scanner is turned on. Connect the scanner after making the settings shown in The scanner setting should be performed from Scanner Setting F of DENSO WAVE. Refer to 5.Appendix C for the correspondence between the Scanner Setting F screen and the settings. Figure 2-1.

The scanner setting should be performed from Scanner Setting F of DENSO WAVE. Refer to 5.Appendix C for the correspondence between the Scanner Setting F screen and the settings.

Figure 2-1 Scanner Settings for Provider Use

Classification				Setting item
RFID	RF tag communication operation mode			Upper level control mode
Communication	RF tag read data transmission interface			Ethernet
	Ethernet	IP Address Setting	IP address	Ethernet
			Subnet mask	Optional
			Default Gateway	Optional
		Command Control	Protocol	TCP Server
		Read Data Transfer	Protocol	TCP Client
Trigger	Trigger type			Command
	Trigger Command Setting	Trigger command response		Enabled
		Trigger ON command		TRGON
		Trigger OFF command		TRGOFF
	Fail Notification Data Output	Fail Notification Data Output		Checked
		Failure notification data		ERROR
Output format	Separator			Comma
	Scanner ID			Enabled
	Time			Enabled
	Optional String 1			No data
	Scanning Condition Number			Enabled
	Communication identifier			Enabled
	Response method			ASCII response
	PC			Enabled
	RSSI			Enabled

	Antenna number	Enabled
	Polarization	Enabled
	Optional String 2	No data

2.2. Read Data Transfer

Immediately after the provider starts up, the connection with the scanner may not be complete and read data may not be received. Get the @READ_CONNECTED variable and make sure it is set to True before reading tags.

2.3. Connection Settings

The correspondence between the scanner's IP address and port settings and the provider's settings is as follows.

scanner		provider
IP address setting		IP address for Conn option
command and control	standby port number	Port number for Conn option
Read data transmission	IP address	There is no support by the provider. It corresponds to the IP address of the PC running the provider.
	destination port number	ReadPort Option

The following diagram shows the relationship between the configuration software, the PC, and the provider.

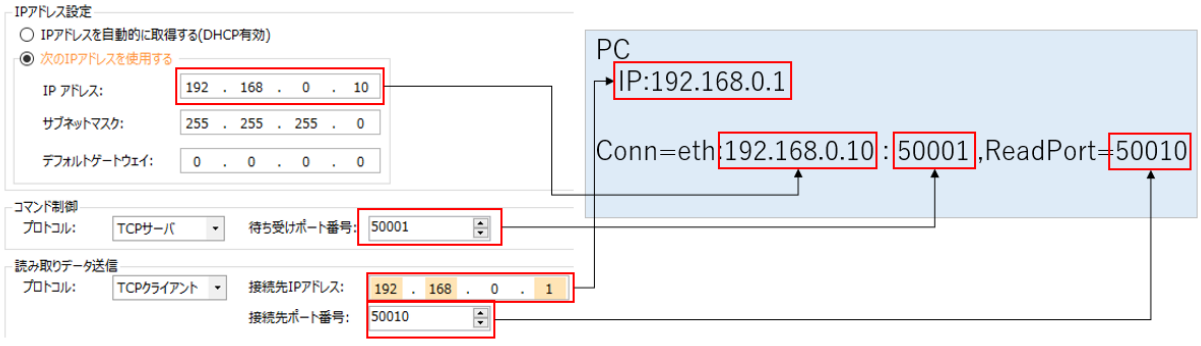


Figure 2-2 Correspondence diagram with configuration software

2.4. Setting Up the PC Development Environment

2.4.1. Installing UR40 Providers Automatically

If ORiN2 SDK is installed, the operating environment (runtime) is ready to connect to the UR40/UR50.

To set up the development environment, prepare a programming environment that supports Component Object Model (COM, Component Objects Model), such as Microsoft Visual Studio, 2003/2005/2008/2010, LabVIEW, etc.

2.4.2. Installing UR40 Providers Manually

To install the UR40 providers manually, you must register the following registry. To register the registry, start the command prompt with administrator privileges and execute the regsvr32 command.

In addition, a valid ORiN2 SDK license must be registered for each computer before the CAO engine can run. See the "Adding and Removing Licenses" section in the ORiN2 SDK User's Guide.

Figure 2-3 UR40 provider

File name	CaoProvDENSOUR40.dll
ProgID	CaoProv.DENSO.UR40
Registry registration	regsvr32 CaoProvDENSOUR40.dll
Unregistering the Registry	regsvr32 /u CaoProvDENSOUR40.dll

3. Command reference

3.1. Method/Property List

Table 3-1 List of methods and properties

Category	Methods/Properties[1]		Facility	Reference
CaoWorkspace				
	AddController	M	Connect to Controller	P.11
CaoController				
	VariableNames	P	Get a list of connectable variable names	P.12
	Variables	P	Retrieving Variable Collections Held by a Controller	P.13
	AddVariable	M	Adding Variable Objects	P.13
	Execute	M	Executing Extended Commands	P.13
	OnMessage	E	Message reception event	P.14
CaoVariable				
	Value	P	Get/Set Value	P.14

3.2. Method Properties

3.2.1. CaoWorkspace classes

3.2.1.1. AddController method

Add a controller object to the CaoWorkspace. UR40 providers refer to the parameters passed when the AddController method is executed, and connect with the relevant scanners. The following are the specifications for the AddController method:

Format

CaoController AddController

```
(
        Controller name          // Controller name (optional)
        "CaoProv.DENSO.UR40",          // Provider name (fixed)
        Machine name                   // Provider Running Machine Name
        Option                   // Option string
)
```

Option

The options specified in the option string are as follows: The option string is a string in which each option

[1] M: Indicates a method, P: property, or E: event, respectively.

shown below is connected with a comma (,).

Option	Required	Description	Values range	Default value
CONN =<Eth connection option>	○	Specify the IP address of the device to connect to.	--	--
ReadPort =<read data receive port>	--	Specifies the port for receiving read data.	1-65535	50010
MessageMax =<Create Message objects>	--	Sets a limit on the number of Message objects that UR40 providers generate.	1-4294967295	100
ConnectionLog =<Connection Log Output Flag>	--	Sets whether to output a log indicating that communication between the read data port and UR40 has not been established.	True or False	True

Examples of uses

```
Dim engine As CaoEngine          ' Engine objects
Dim workspace As CaoWorkspace          ' Workspace objects
Dim controller As CaoController   ' Controller objects

Set engine = New CaoEngine
Set workspace = engine.Workspaces.Item(0)
Set controller = workspace.AddController("SampleController",_
                                          "CaoProv.DENSO.UR40",_
                                          "",_
                                          "conn=eth:192.168.0.10,ReadPort=50010")
```

3.2.1.1.1. CONN options

The following are the connect parameter strings for the Conn options: Here, square brackets ("[]") indicate that it is optional, and the underlined part in the explanation of each parameter indicates the default value when the option is not specified. In this provider, if ETH is specified, TCP connection is performed.

ETH

"Conn=Eth:<Destination IP>[:<Destination Port>[:<Local IP>[:<Local Port>]]""

<Connection IP> : Destination IP address. "192.168.0.10" etc..

Connection port : Destination port number 50001

<Local IP> : The local IP address.

Local port : Local Port Number

3.2.2. CaoController classes

3.2.2.1. VariableNames Properties

Gets a list of connectable variable names. The variable name obtained by this property can be used as the first

argument of the AddVariable method described later.AddVariable method

Examples of uses

```
' Connect
Call Connect

' Get variable name list
Dim variables as Variant
Variables = controller.VariableNames

' Disconnect
Call Disconnect
```

3.2.2.2. Variables Properties

Gets the variable collection held by the controller.

Examples of uses

```
' Connect
Call Connect

' Variable collection
Dim variables as CaoVariables
Set variables = controller.Variables

' Get variable
Dim variable as CaoVariable
Set variable = variables.Item(0)

' Disconnect
Call Disconnect
```

3.2.2.3. AddVariable method

Add variables to the CaoController. Only the variables listed in 3.4.1 can be used.

The AddVariable specifications are shown below.

Format

CaoVariable AddVariable

```
(
        Variable name           // Variable name
        Option          // Optional String
)
```

3.2.2.4. Execute method

Executes CaoController extended commands. The Execute specifications are shown below.

Format

Variant Execute

```
(
        "<extended command name>",              // Extended command name
        "<option string>"               // Optional String
)
```

For a list of extended commands that can be specified with Execute, see 3.3 Execute commands. Examples of use are described in detail in Extended Commands.

3.2.3. CaoVariable classes

3.2.3.1. Value Properties

Acquires/sets data from the connected scanner. The behavior depends on the variable name. For details, see 3.4Variable List.

3.2.4. OnMessage classes

Data to be notified when the scanner detects the RF tag during the RF tag control is issued as a message event. For details on events, refer to 3.5 Event list.

For the RF tag control status of the scanner, refer to Section 3.3.5.

3.3. Execute commands

3.3.1. Outline

Configuration change commands, tag control commands to enable communication with RF tags, and trigger commands are implemented in this provider.

To execute the read operation from the provider to the scanner, send the tag control command, register the command, and then activate it by the trigger command. The scanned RF tags can be obtained by OnMessage. Figure 3-1 is an example of using the Execute command to control tags from providers.

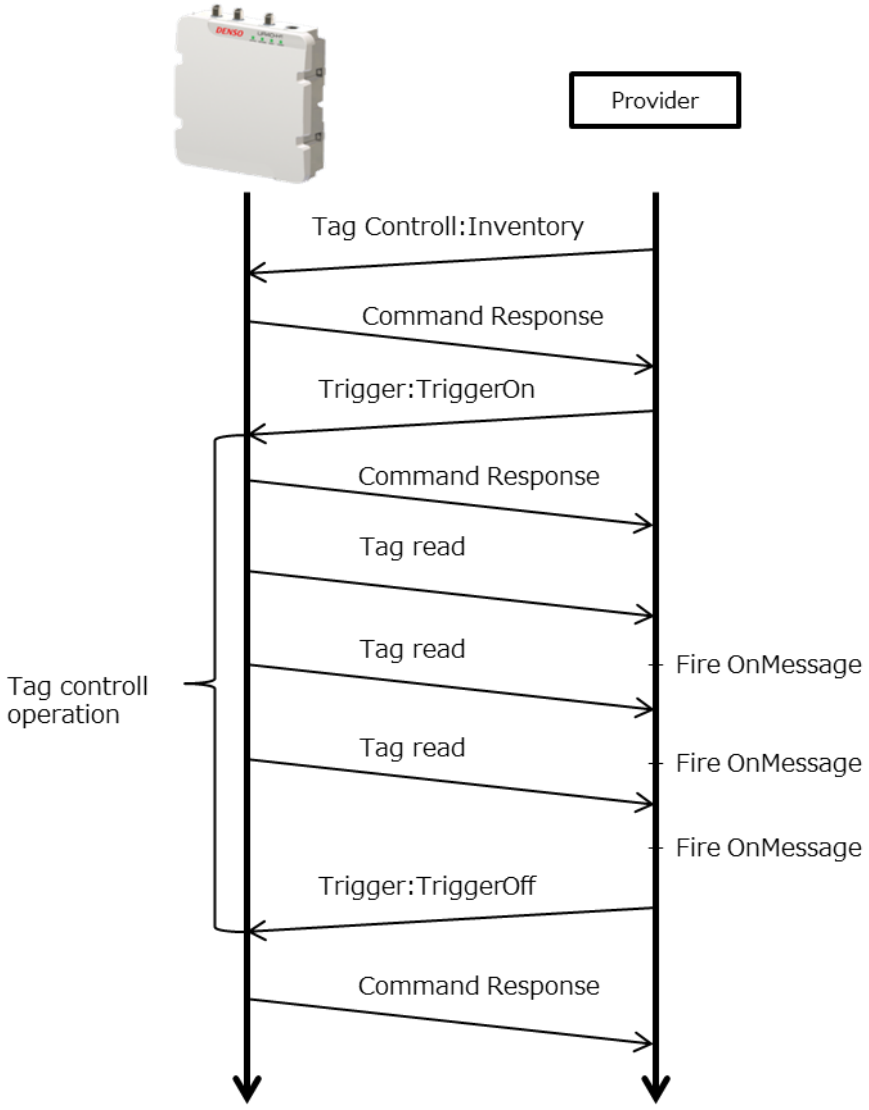


Figure 3-1 Tag Control by Execute Commands (Inventory)

* Tag control operation is also started in the order of trigger command Trriger : TriggerOn → Tag control : Inventory.

3.3.2. List of Execute Commands

Commanded	Type	Description	Reference
TriggerContinuous	Setting change	Switch the trigger mode to continuous.	P.17
TriggerSingle	Setting change	Switches the trigger mode to standalone mode.	P.17
Inventory	Tag control	Registers Inventory commands to acquire UIIs of RF tags.	P.18
InventoryUII	Tag control	Registers Inventory commands that specify the UII of the RF tag and obtain the UII of the RF tag.	P.18
Read	Tag control	Register the Read commands to read the memory data of the RFtag.	P.19
ReadUII	Tag control	Specify the UII of the RF tag and register the Read commands to read the memory data of the RF tag.	P.20
Write	Tag control	Register the Write commands to write the memory data of the RFtag.	P.21
WriteUII	Tag control	Specify the UII of the RF tag and register the Write commands to write the memory data of the RF tag.	P.21
Lock	Tag control	Register the Lock commands to change the lock status of the RFtag.	P.22
LockUII	Tag control	Specify the UII of the RF tag and register the Lock commands to change the lock status of the RF tag.	P.23
Kill	Tag control	Register the Kill commands to unavailable the RFtag.	P.24
KillUII	Tag control	Specify the UII of the RF tag and register the Kill commands to unavailable the RF tag.	P.25
CommandRelease	Tag control	Cancels the registered command.	P.25
TriggerOn	Trigger	Activate the registered command.	P.26
TriggerOff	Trigger	Stops an active command.	P.26
Reset	Other	Restart the scanner.	P.27
Raw	Other	Send any command to the scanner.	P.27

3.3.3. Command Tag Control

To read an RF tag, execute the tag control command, and then execute the TriggerOn command. To switch the command midway, send the CommandRelease command, and then send the command that you want to execute.

The following describes how to send a command when reading RFID tags using the Inventory command.

※The scanner is in "upper control mode" and the trigger type is "command".

1. Execute the Inventory from the power-on.

Steps	Operation

1	Send Inventory commands
2	Send TriggerOn commands

2. Execute Read commands from status 1.

Steps	Operation
1	Send TriggerOff commands
2	Send CommandRelease commands
3	Send Read commands
4	Send TriggerOn commands

※If you execute step 3 instead of sending the TriggerOff command in step 1, the Read operation is executed at the stage where the Read command is sent.

3.3.4. Setting change command

The parameter command acquires the settings of the scanner's parameters. An error is returned during tag control operation or after a TriggerOn command. Execute the setting change command after executing the TrggerOff and CommandRelease commands.

3.3.4.1. TriggerContinuous

Switch trigger mode to continuous mode. The one-shot time can be set from the argument.

Executes the specified time reading operation in one-shot time.

To re-read the tags after the one-shot operation ends, send the TriggerOff command and then send the TriggerOn command.

Item	Type	Description
Argument	VT_UI1	Specifies the one-shot time. 0: One-shot disabled. 1-255: With one-shot. Time of input value ×100ms.
Returned value	Without	

Examples of uses

```
' Connect
Call Connect
' Change Trigger mode
Controller.Execute("TriggerContinuous", 10)
' Disconnect
Call Disconnect
```

3.3.4.2. TriggerSingle

Switch the trigger mode to the standalone mode. The one-shot time can be set from the argument.

Executes the specified time reading operation in one-shot time. In the stand-alone mode, the operation ends after one-shot scanning or one-shot time has elapsed.

To re-read tags after operation is completed, send the TriggerOff command and then send the TriggerOn command.

Item	Type Description	
Argument	VT_UI1	Specifies the one-shot time. 0: One-shot disabled. 1-255: With one-shot. Time of input value ×100ms.
Returned value	Without	

Examples of uses

```
' Connect
Call Connect
' Change Trigger mode
Call controller.Execute("TriggerSingle", 10)
' Disconnect
Call Disconnect
```

3.3.5. RF tag control command

The RF tag control command registers commands to the scanner by executing the command.

Reading is executed by sending TriggerOn commands after registering.

If the scanner detects an RF tag during communication control, a message event is issued each time the information is notified to notify the information of the detected RF tag.

See below for details of each command.

3.3.5.1. Inventory

When the scanner detects an RF tag, it notifies the UII data detected.

When UII data is notified from scanners, a Inventory message event (see 3.5.2) is issued.

If this command is executed while RF tag control is started, an error is returned.

Item	Type Description
Argument	Without
Returned value	Without

Examples of uses

```
' Connect
Call Connect
' Regist Inventory command
Call controller.Execute("Inventory")
```

3.3.5.2. InventoryUII

When the scanner detects an RF tag with the specified UII, it notifies the detected UII data.

When UII data is notified from scanners, a Inventory message event (see 3.5.2) is issued.

If this command is executed while RF tag control is started, an error is returned.

Item	Type	Description
Argument	VT_BSTR	Specifies the UII data. Digit range: 4 - 124 (set according to UII data length, hexadecimal character string of up to 62 bytes)
Returned value	Without	

Examples of uses

```
' Connect
Call Connect
Dim UII As String
' UII
UII = "AAAA00001234"
' Regist InventoryUII command
Call controller.Execute("InventoryUII", UII)
```

3.3.5.3. Read

When the scanner detects an RF tag, it will report the contents of the specified range of memory data. A Read message event (see Section 3.5.3) is issued when the scanner notifies the scanner of the read data. It cannot be executed when the scanner has started RF tag control.

Item			Description
Argument	VT_ARRAY | VT_VARIANT		
	1	VT_I4	Specify Bank Value Range: 0-3 0:Reserved Bank 1:UII Bank 2:TID Bank 3:User Bank
	2	VT_I4	Specify the read start position Value range: 0-65535[2]
	3	VT_I4	Specify the read size Value range: 2-2562[2]
	4	VT_BSTR	The access password is specified as a hexadecimal character string (authentication processing is omitted when "00000000" is specified). Value-range: "00000000"-"FFFFFFFF"
Returned value	Without		

Examples of uses

```
' Connect
Call Connect
' Set Parameter
```

[2] Check the specifications of your RF Tag for the actual specifiable range.

```
Dim readParam As Variant
ReadParam = Array(0, 4, 4, "00000000")
' Regist Read command
Call controller.Execute("Read", readParam)
```

3.3.5.4. ReadUII

When the scanner detects an RF tag with the specified UII, it will report the contents of the specified range of memory data. A Read message event (see Section 3.5.3) is issued when the scanner notifies the scanner of the read data. It cannot be executed when the scanner has started RF tag control.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	1	VT_I4	Specify Bank Value Range: 0-3 0:Reserved Bank 1:UII Bank 2:TID Bank 3:User Bank
	2	VT_I4	Specify the read start position Value range: 0-65535[3]
	3	VT_I4	Specify the read size Value range: 2-2563[3]
	4	VT_BSTR	The access password is specified as a hexadecimal character string (authentication processing is omitted when "00000000" is specified). Value-range: "00000000"-"FFFFFFFF"
	5	VT_BSTR	Specifies the UII data. Digit range: 4 - 124 (set according to UII data length, hexadecimal character string of up to 62 bytes)
Returned value	Without		

Examples of uses

```
' Connect
Call Connect
' Set Parameter
Dim readParam As Variant
ReadParam = Array(0, 4, 4, "00000000", "AAAA00001234")
' Regist ReadUII command
Call controller.Execute("ReadUII", readParam)
```

[3] Check the specifications of your RF Tag for the actual specifiable range.

3.3.5.5. Write

When the scanner detects a RF tag, the specified value is written to the specified range of memory data. When data is written, a Write message event (see 3.5.4) is issued. It cannot be executed when the scanner has started RF tag control.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	1	VT_I4	Specify Bank Value Range: 0-3 0:Reserved Bank 1:UII Bank 2:TID Bank 3:User Bank
	2	VT_I4	Specify the write start position Value range: 0-65535[4]
	3	VT_BSTR	The access password is specified as a hexadecimal character string (authentication processing is omitted when "00000000" is specified). Value-range: "00000000"-"FFFFFFFF"
	4	VT_BSTR	Data to be written is specified as a hexadecimal character string. Specify in 1-byte (2-character) units.
Returned value	Without		

Examples of uses

```
' Connect
Call Connect
' Set Parameter
Dim writeParam As Variant
writeParam = Array(1, 4, "00000000", "ABCDEF12")
' Regist Read command
Call controller.Execute("Write", writeParam)
```

3.3.5.6. WriteUII

When the scanner detects a RF tag with the specified UII, the specified value is written to the specified range of memory data. When data is written, a Write message event (see 3.5.4) is issued. It cannot be executed when the scanner has started RF tag control.

Item	Type Description
Argument	VT_ARRAY | VT_VARIANT

[4] Check the specifications of your RF Tag for the actual specifiable range.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	1	VT_I4	Specify Bank Value Range: 0-3 0:Reserved Bank 1:UII Bank 2:TID Bank 3:User Bank
	2	VT_I4	Specify the write start position Value range: 0-65535[5]
	3	VT_BSTR	The access password is specified as a hexadecimal character string (authentication processing is omitted when "00000000" is specified). Value-range: "00000000"-"FFFFFFFF"
	4	VT_BSTR	Data to be written is specified as a hexadecimal character string. Specify in 1-byte (2-character) units.
	5	VT_BSTR	Specifies the UII data. Digit range: 4 - 124 (set according to UII data length, hexadecimal character string of up to 62 bytes)
Returned value	Without		

Examples of uses

```
' Connect
Call Connect
' Set Parameter
Dim writeParam As Variant
writeParam = Array(1, 4, "00000000", "ABCDEF12", "AAAA00001234")
' Regist ReadUII command
Call controller.Execute("WriteUII", writeParam)
```

3.3.5.7. Lock

When the scanner detects a RF tag, the lock status of the specified area is changed. When a lock state change is executed, a Lock message event (see 3.5.5) is issued. It cannot be executed when the scanner has started RF tag control.

Item	Type Description
Argument	VT_ARRAY | VT_VARIANT

[5] Check the specifications of your RF Tag for the actual specifiable range.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	1	VT_UI1 or VT_BSTR	Specify the area to be locked by the sum of the following values When specifying with VT_BSTR, specify it as a two-digit hexadecimal character string. 01h: kill password 02h: Access password 10h:UII Bank 20h:TID Bank 40h:User Bank
	2	VT_UI1 or VT_BSTR	Specify lock type When specifying with VT_BSTR, specify it as a two-digit hexadecimal character string. 00h: Unlock 01h: Locking 10h: Permanent unlock 11h: Permanent Lock
	3	VT_BSTR	The access password is specified as a hexadecimal character string (authentication processing is omitted when "00000000" is specified). Value-range: "00000000"-"FFFFFFFF"
Returned value	Without		

Examples of uses

```
' Connect
Call Connect
' Set Parameter
Dim lockParam As Variant
lockParam = Array(&H11, &H00, "ABCDEF0123")
' Regist Read command
Call controller.Execute("Lock", lockParam)
```

3.3.5.8. LockUII

When the scanner detects a RF tag with the specified UII, the lock status of the specified area is changed. When a lock state change is executed, a Lock message event (see 3.5.5) is issued. It cannot be executed when the scanner has started RF tag control.

Item	Type Description
Argument	VT_ARRAY | VT_VARIANT

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	1	VT_UI1 or VT_BSTR	Specify the area to be locked by the sum of the following values When specifying with VT_BSTR, specify it as a two-digit hexadecimal character string. 01h: kill password 02h: Access password 10h:UII Bank 20h:TID Bank 40h:User Bank
	2	VT_UI1 or VT_BSTR	Specify lock type When specifying with VT_BSTR, specify it as a two-digit hexadecimal character string. 00h: Unlock 01h: Locking 10h: Permanent unlock 11h: Permanent Lock
	3	VT_BSTR	The access password is specified as a hexadecimal character string (authentication processing is omitted when "00000000" is specified). Value-range: "00000000"-"FFFFFFFF"
	4	VT_BSTR	Specifies the UII data. Digit range: 4 - 124 (set according to UII data length, hexadecimal character string of up to 62 bytes)
Returned value	Without		

Examples of uses

```
' Connect
Call Connect
' Set Parameter
Dim lockParam As Variant
lockParam = Array(&H11, &H00, "ABCDEF12", "AAAA00001234")
' Regist ReadUII command
Call controller.Execute("LockUII", lockParam)
```

3.3.5.9. Kill

Places RF tag unavailable when the scanner detects a RF tag. When RF tag becomes unavailable, an Kill message event (see 3.5.6) is issued. It cannot be executed when the scanner has started RF tag control.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	1	VT_BSTR	Specify the kill password as a hexadecimal string Range: "00000000"-"FFFFFFFF"
Returned value	Without		

Examples of uses

```
' Connect
Call Connect
' Set Parameter
Dim killParam As Variant
killParam = Array("ABCDEF0123")
' Regist Read command
Call controller.Execute("Kill", killParam)
```

3.3.5.10. KillUII

Sets RF tag unavailable when the scanner detects a RF tag with the specified UII. When RF tag becomes unavailable, an Kill message event (see 3.5.6) is issued. It cannot be executed when the scanner has started RF tag control.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	1	VT_BSTR	Specify the kill password as a hexadecimal string Range: "00000000"-"FFFFFFFF"
	2	VT_BSTR	Specifies the UII data. Digit range: 4 - 124 (set according to UII data length, hexadecimal character string of up to 62 bytes)
Returned value	Without		

Examples of uses

```
' Connect
Call Connect
' Set Parameter
Dim killParam As Variant
killParam = Array("ABCDEF12", "AAAA00001234")
' Regist ReadUII command
Call controller.Execute("KillUII", killParam)
```

3.3.5.11. CommandRelease

Releases the registered RF tag control command. Send a new tag control command before registering it.

Item	Type Description
Argument	Without
Returned value	Without

Examples of uses

```
' Connect
Call Connect
' Regist Inventory command
Call controller.Execute("Inventory")
' Unregist command
Call controller.Execute("CommandRelease")

' Regist Read command
Dim readParam As Variant
ReadParam = Array(0, 4, 4, "00000000")
Call controller.Execute("Read", readParam)
```

3.3.6. Trigger command

With the trigger command, the scanner executes the command registered with the tag control command. The message event corresponding to each command is issued for the tag read by the executed tag control command.

3.3.6.1. TriggerOn

Execute the registered command operation. If you send the TriggerOn command again after sending the TirggerOn command, an error is returned.

Item	Type Description
Argument	Without
Returned value	Without

Examples of uses

```
' Connect
Call Connect
' Regist Inventory command
Call controller.Execute("Inventory")
' Start RF-tag controll(Inventory)
Call controller.Execute("TriggerOn")
```

3.3.6.2. TriggerOff

Stops the registered command operation.

Item	Type Description
Argument	Without
Returned value	Without

Examples of uses

```
' Connect
Call Connect
' Regist Inventory command
Call controller.Execute("Inventory")
' Start RF-tag controll(Inventory)
Call controller.Execute("TriggerOn")
' Stop RF-tag controll(Inventory)
Call controller.Execute("TriggerOff")
```

3.3.7. Other Commands

3.3.7.1. Reset

Restart the connected scanner. A CaoController reconnect is required to connect after a reboot.

Item	Type Description
Argument	Without
Returned value	Without

Examples of uses

```
' Connect
Call Connect
' Reset scanner
Call controller.Execute("Reset")
' Connect
Call Disconnect
Call Connect
```

3.3.7.2. Raw

Sends an arbitrary command character string to the scanner.

Refer to the scanner's manual for the command format.

Item	Type Description	
Argument	VT_BSTR	Any command string
Returned value	VT_BSTR	Command response

Examples of uses

```
' Connect
Call Connect
' Send raw command
Controller.Execute("Raw","RFUW,03,0000,00000000,11223344")
```

3.4. Variable List

Defines a list of variables that can be used in each class. Variables are objects of CaoVariable classes.

3.4.1. CaoController Classes Variables

Variable name	Acquisition Data Type	Description	Value		Reference
			Get	Put	
@MAKER_NAME	VT_BSTR	Obtain the provider manufacturer name.	✓	-	P.28
@VERSION	VT_BSTR	Get the provider version.	✓	-	P.28
@DEVICE_VERSION	VT_BSTR	Get the scanner firmware version.	✓	-	P.29
@ID	VT_BSTR	Get the ID of the scanner.	✓	-	P.29
@READ_CONNECTED	VT_BOOL	This function acquires the connection status between the read data reception port and the scanner.	✓	-	P.29

3.4.1.1. @MAKER_NAME

Get the manufacturer's name.

Data type

Type	Description
VT_BSTR	Get the manufacturer name.

Examples of uses

```
' Connect
Call Connect
' Add Variable
Dim var As CaoVariable
Set var = controller.AddVariable("@MAKER_NAME")
' Get value
Dim strVal As String
StrVal = var.value
```

3.4.1.2. @VERSION

Get the provider version.

Data type

Type	Description
VT_BSTR	Get the provider version.

Examples of uses

```
' Connect
Call Connect
' Add Variable
Dim var As CaoVariable
Set var = controller.AddVariable("@VERSION")
' Get value
Dim value As String
Value = var.value
```

3.4.1.3. @DEVICE_VERSION

Obtain the scanner firmware version.

Data type

Type Description	
VT_BSTR	Get the scanner firmware version.

Examples of uses

```
' Connect
Call Connect
' Add Variable
Dim var As CaoVariable
Set var = controller.AddVariable("@DEVICE_VERSION")
' Get value
Dim value As String
Value = var.value
```

3.4.1.4. @ID

Obtain the scanner serial number.

Data type

Type Description	
VT_BSTR	Get the scanner firmware version.

Examples of uses

```
' Connect
Call Connect
' Add Variable
Dim var As CaoVariable
Set var = controller.AddVariable("@ID")
' Get value
Dim value As String
Value = var.value
```

3.4.1.5. @READ_CONNECTED

This function acquires the connection status between the read data reception port and the scanner. Immediately after the provider starts up, the connection with the scanner is not completed and read data may not be received. Please make sure that the @READ_CONNECTED variable is set to True before reading tags.

Data type

Type Description	
VT_BOOL	True: Connecting: False: Disconnecting.

Examples of uses

```
' Connect
Call Connect
' Add Variable
Dim var As CaoVariable
Set var = controller.AddVariable("@READ_CONNECTED")
' Get value
Dim value As String
Value = var.value
```

3.5. Event list

Notifies the information of the RF tag detected by the scanner. Messages are issued once per tag, and if multiple tags are read, they are notified as many times as they have been read.

The contents of the message differ for each RF tag control command. For details, see below.

Message number	Description	Reference
0	Error code is notified.	P.30
1	Indicates the RF tag detected during RF tag control by the Inventory.	P.31
2	This reports the RF tag information detected during RF tag control by the Read and the read data of the specified memory area.	P.31
99	Notifies other data types.	P.35

3.5.1. Error messages

Notifies the error code during tag control and the received data. The content of the received data depends on the tag control command being executed. For the content of the error code, refer to "5.UR40 provider error code".

Item	Description		
Message number	0		
Message content	VT_BSTR | VT_ARRAY		
	0	VT_BSTR	Error code. (HRESULT format)
	1 - n	VT_BSTR	Data received from the scanner(see supplement below).

[Supplementation]

The output contents differ depending on the setting of "Failure Notification Data" -> "Apply Output Format Setting" of the scanner.

- Apply output format settings: No

  The failure notification data character (ERROR) is output.
- Apply output format settings: Yes

The serial number, date, and failure notification data character (ERROR) are output.

3.5.2. Inventory messages

The scanner notifies the scanner of the detected RF tag data during RF tag control by Inventory (see Section 3.3.5.1).

Item	Description		
Message number	1		
Message content	VT_BSTR | VT_ARRAY		
	1	VT_BSTR	Scanner ID
	2	VT_BSTR	Time
	3	VT_BSTR	Scanning Condition Number
	4	VT_BSTR	Communication identifier
	5	VT_BSTR	UII
	6	VT_BSTR	PC
	7	VT_BSTR	RSSI
	8	VT_BSTR	Antenna number
	9	VT_BSTR	Polarization

3.5.3. Read messages

When the communication result of RF tag communication is normal, the specified memory area is also notified when the scanner is controlling RF tag by Read (see 3.3.5.3).

Item	Description		
Message number	2		
Message content	VT_BSTR | VT_ARRAY		
	1	VT_BSTR	Scanner ID
	2	VT_BSTR	Time
	3	VT_BSTR	Scanning Condition Number
	4	VT_BSTR	Communication identifier
	5	VT_BSTR	UII
	6	VT_BSTR	PC
	7	VT_BSTR	RSSI
	8	VT_BSTR	Antenna number
	9	VT_BSTR	Polarization

<table>
<tr><td>Item</td><td colspan="3">Description</td></tr>
<tr><td>Message number</td><td colspan="3">2</td></tr>
<tr><td>Message content</td><td colspan="3">VT_BSTR | VT_ARRAY</td></tr>
<tr><td></td><td>10</td><td>VT_BSTR</td><td>Error-code
<table>
<tr><th>Error-code</th><th>Content</th></tr>
<tr><td>0000</td><td>Normal termination</td></tr>
<tr><td>0203</td><td>Access out of memory range.
Check the specified address, size, and RF tag specifications.</td></tr>
<tr><td>0204</td><td>Access to locked memory.
Either specify an access password to access the card or unlock the card in advance.</td></tr>
<tr><td>020F</td><td>An unspecified error was returned from the RF tag.
Check the specifications of the RF tag.</td></tr>
</table></td></tr>
<tr><td></td><td>11</td><td>VT_BSTR</td><td>Scanned data
If the error code is not normal termination, an empty character is stored.</td></tr>
</table>

3.5.4. Write messages

Informs the scanner of the detected RF tag during RF tag control by Write (see 3.3.5.5).

Item	Description		
Message number	3		
Message content	VT_BSTR | VT_ARRAY		
	1	VT_BSTR	Scanner ID
	2	VT_BSTR	Time
	3	VT_BSTR	Scanning Condition Number
	4	VT_BSTR	Communication identifier
	5	VT_BSTR	UII
	6	VT_BSTR	PC
	7	VT_BSTR	RSSI
	8	VT_BSTR	Antenna number
	9	VT_BSTR	Polarization

<table>
<tr><td>Item</td><td colspan="3">Description</td></tr>
<tr><td>Message number</td><td colspan="3">3</td></tr>
<tr><td>Message content</td><td colspan="3">VT_BSTR | VT_ARRAY</td></tr>
<tr><td rowspan="5"></td><td>10</td><td>VT_BSTR</td><td>Error-code<table><tr><th>Error-code</th><th>Content</th></tr><tr><td>0000</td><td>Normal termination</td></tr><tr><td>0203</td><td>Access out of memory range.
Check the specified address, size, and RF tag specifications.</td></tr><tr><td>0204</td><td>Access to locked memory.
Either specify an access password to access the card or unlock the card in advance.</td></tr><tr><td>020F</td><td>An unspecified error was returned from the RF tag.
Check the specifications of the RF tag.</td></tr></table></td></tr>
</table>

3.5.5. Lock messages

Informs the scanner of the detected RF tag during RF tag control by Lock (see 3.3.5.7).

Item	Description		
Message number	4		
Message content	VT_BSTR | VT_ARRAY		
	1	VT_BSTR	Scanner ID
	2	VT_BSTR	Time
	3	VT_BSTR	Scanning Condition Number
	4	VT_BSTR	Communication identifier
	5	VT_BSTR	UII
	6	VT_BSTR	PC
	7	VT_BSTR	RSSI
	8	VT_BSTR	Antenna number
	9	VT_BSTR	Polarization

<table>
<tr><td>Item</td><td colspan="3">Description</td></tr>
<tr><td>Message number</td><td colspan="3">4</td></tr>
<tr><td>Message content</td><td colspan="3">VT_BSTR | VT_ARRAY</td></tr>
<tr><td></td><td>10</td><td>VT_BSTR</td><td>Error-code
<table><tr><th>Error-code</th><th>Content</th></tr><tr><td>0000</td><td>Normal termination</td></tr><tr><td>0203</td><td>Access out of memory range.
Check the specified address, size, and RF tag specifications.</td></tr><tr><td>0204</td><td>Access to locked memory.
Either specify an access password to access the card or unlock the card in advance.</td></tr><tr><td>020F</td><td>An unspecified error was returned from the RF tag.
Check the specifications of the RF tag.</td></tr></table></td></tr>
</table>

3.5.6. Kill messages

Informs the scanner of the detected RF tag during RF tag control by Kill (see 3.3.5.9).

Item	Description		
Message number	4		
Message content	VT_BSTR | VT_ARRAY		
	1	VT_BSTR	Scanner ID
	2	VT_BSTR	Time
	3	VT_BSTR	Scanning Condition Number
	4	VT_BSTR	Communication identifier
	5	VT_BSTR	UII
	6	VT_BSTR	PC
	7	VT_BSTR	RSSI
	8	VT_BSTR	Antenna number
	9	VT_BSTR	Polarization

<table>
<tr><td>Item</td><td colspan="3">Description</td></tr>
<tr><td>Message number</td><td colspan="3">4</td></tr>
<tr><td>Message content</td><td colspan="3">VT_BSTR | VT_ARRAY</td></tr>
<tr><td rowspan="5"></td><td>10</td><td>VT_BSTR</td><td>Error-code
<table><tr><th>Error-code</th><th>Content</th></tr><tr><td>0000</td><td>Normal termination</td></tr><tr><td>0203</td><td>Access out of memory range.
Check the specified address, size, and RF tag specifications.</td></tr><tr><td>0204</td><td>Access to locked memory.
Either specify an access password to access the card or unlock the card in advance.</td></tr><tr><td>020F</td><td>An unspecified error was returned from the RF tag.
Check the specifications of the RF tag.</td></tr></table></td></tr>
</table>

3.5.7. Other Messages

Scanner notifies the scanner of the content of the scanned data other than Inventory,Read data...

<table>
<tr><td>Item</td><td colspan="3">Description</td></tr>
<tr><td>Message number</td><td colspan="3">99</td></tr>
<tr><td rowspan="2">Message content</td><td colspan="3">VT_BSTR | VT_ARRAY</td></tr>
<tr><td>1 - n</td><td>VT_BSTR</td><td>Arrays are allocated n times the number of elements notified from the scanner, and each element is stored in VT_BSTR type.</td></tr>
</table>

4. Programming by UR40 providers

With UR40 providers, you can connect scanners to clients using the following procedures.

- Creating a CaoEngine
- Creating a CaoWorkspace
- Creating a CaoController

Once connected to the scanner, the scanner's info can be accessed using the CaoController's Execute method or by creating CaoVariable objects.

4.1. Sample Programming to Connect to Scanner and Read UII Data of RF Tag

Here are examples of sample programs that set scanner parameters and save them in flash memory. Table 4-1 Sample Program Requirements describes the requirements for sample programs, and 4.1.1 describes the flow of sample programs.

Table 4-1 Sample Program Requirements

Requirement	Description
Destination	ETH
	The connection destination IP is 192.168.0.10, the read data transfer destination port is 50010. Other items are omitted.
Process Description	Starts communication control in Inventory.
	Displays a message when UII data is read from the scanner.
	Disconnect after displaying a message.

4.1.1. Sample program

The following shows an overview of the sample program.

Examples are provided in VB.Net.

Sample **SampleInventory.vb**

```
 ' Object
 Dim engine As CaoEngine
Dim workspace As CaoWorkspace
Dim WithEvents controller As CaoController
Dim MessageBuf As Object

' Connect
Private Sub Connect()
      ' Create CaoEngine
    Engine = New CaoEngine
    ' Create CaoWorkspace
    Workspace = engine.AddWorkspace("Newwrks", "")
    ' Create CaoController
    Controller = workspace.AddController("SampleController", _
                                         "CaoProv.DENSO.UR40",_
                                         "",_
```

```
                                                    "conn=eth:192.168.0.10,ReadPort=50010")
        ' Regist OnMessage handler
        AddHandler controller.OnMessage, AddressOf controller_OnMessage
    End Sub

    ' Disconnect
    Private Sub Disconnect()
            ' Send TriggerOff
        If controller IsNot Nothing Then
            Call controller.Execute("TriggerOff")
        End If

        ' Delete OnMessage handler
        RemoveHandler controller.OnMessage, AddressOf controller_OnMessage
          ' Delete CaoController from CaoWorkspace
        Call workspace.Controllers.Remove(controller.Index)
        ' Delete CaoController
        System.Runtime.InteropServices.Marshal.ReleaseComObject(controller)
        Controller = Nothing
          ' Delete CaoWorkspace from CaoEngine
          Call engine.Workspaces.Remove(workspace.Index)
          ' Delete CaoWorkspace
        System.Runtime.InteropServices.Marshal.ReleaseComObject(workspace)
          Workspace = Nothing
          ' Delete CaoEngine
        System.Runtime.InteropServices.Marshal.ReleaseComObject(engine)
        Engine = Nothing
    End Sub

    ' Inventory
    Private Sub StartInventory()
        ' Connect
        Call Connect()
        ' Stop current command
        Controller.Execute("TriggerOff")
        ' Unregist current command
        Controller.Execute("CommandRelease")

        ' Regist Inventory command
        Controller.Execute("Inventory")
        ' Start Inventory
        Controller.Execute("TriggerOn")

    End Sub

    ' Message event from provider.
    Private Sub controller_OnMessage(ByVal pICaoMess As CAOLib.CaoMessage)
        MessageBuf = pICaoMess.Value
        ' UII data
        Dim uiiData As String
        UiiData = MessageBuf(4)

        ' PC
        Dim pc As String
        Pc = MessageBuf(5)
```

```
    ' RSSI
    Dim rssi As String
    Rssi = MessageBuf(6)

    ' Antenna
    Dim antenna As String
    Antenna = MessageBuf(7)

    ' Display UII data
    MsgBox(uiiData)

End Sub
```

4.1.1.1. Connection

Follow the procedure below to connect the scanner.

(1) Prepare a variable to hold the object. The objects required for connecting controllers are the CaoEngine object, the CaoWorkspace object, and the CaoController object.The CaoWorkpace object does not need to have variables when retrieving the CaoController object from the CaoWorkspaces. You can also prepend a WithEvents to the variable name to allow OnMessage events to be received from the CaoController.

```
Dim engine As CaoEngine                         ' Variable CaoEngine object
Dim workspace As CaoWorkspace                   ' Variable CaoWorkspace object
Dim WithEvents controller As CaoController      ' Variable CaoController object
```

(2) Creates a CaoEngine object. A CaoEngine object is created using the New keyword.

```
' Create CaoEngine object
Set engine = New CaoEngine
```

(3) Gets or creates a CaoWorkspace object. Creating a CaoEngine object defaults to creating one CaoWorkspaces object and one CaoWorkspace object at a time. The following is a sample of code that creates new CaoWorkspace objects and the default CaoWorkspace of the code.

```
' Create CaoWorkspace object
Set workspace = engine.AddWorkspace("NewWrks", "")
```

(4) Generate a CaoController object.To generate a CaoController object, set the provider name to use and the parameters to use.In the UR40 provider, specify the response format beforehand when connecting to the scanner... The following shows a code example.

```
' Create CaoController object
Set controller = workspace.AddController("SampleController", _
                                         "CaoProv.DENSO.UR40", _
                                         "", _
                                         "conn=eth:192.168.0.10,ReadPort=50010")
```

4.1.1.2. Execute scanner in UII-data-retrieval mode.

Starts the scanner in UIIacquisition mode (Inventory).

```
'Regist Inventory command
Controller.Execute("Inventory")
'Start Inventory
Controller.Execute("TriggerOn")
```

4.1.1.3. Receive Message when Reading Tag

When tagging information is received from the scanner, a OnMessage event is issued. When a OnMessage event is received, UII data is displayed as a MsgBox from the message content.

```
Private Sub controller_OnMessage(ByVal pICaoMess As CAOLib.CaoMessage)
    MessageBuf = pICaoMess.Value
    ' Display UII data
    MsgBox MessageBuf(4)
End Sub
```

4.1.1.4. Disconnection

When you disconnect from the controller, you clear the objects that you have created and delete the objects that you want to clear from the collection classes that manage the objects. The following shows a code example.

```
    ' Delete OnMessage handler
RemoveHandler controller.OnMessage, AddressOf controller_OnMessage
    ' Delete CaoController from CaoWorkspace
Call workspace.Controllers.Remove(controller.Index)
' Delete CaoController
System.Runtime.InteropServices.Marshal.ReleaseComObject(controller)
Controller = Nothing
    ' Delete CaoWorkspace from CaoEngine
    Call engine.Workspaces.Remove(workspace.Index)
    ' Delete CaoWorkspace
System.Runtime.InteropServices.Marshal.ReleaseComObject(workspace)
    Workspace = Nothing
    ' Delete CaoEngine
System.Runtime.InteropServices.Marshal.ReleaseComObject(engine)
Engine = Nothing
```

5. UR40 provider error code

The following proprietary error codes are masked with 0x8010****: (See Table 5-1 Unique error code table.)
For ORiN2 common errors, refer to the error code chapters in the ORiN2 Programming Guide (Link).

Table 5-1 Unique error code table

Error name	Error Number	Description
E_INVALID_PACKET	0x80100001	The read data and command response packets are corrupted. Check the connection and scanner settings.
E_OPT_NONCONN	0x80100002	Either eth was not specified in the Conn option, or the option setting is different from the specification. Check the AddController option.
E_NOCLIENT	0x80100003	The read data transfer destination port is not connected to the scanner. Check the connection.
E_MISMATCH_FORMAT	0x80100004	The device format setting does not match the provider setting. Set the device format referring to 2.1.
E_UNREAD_TAG	0x80100005	The tag could not be read during tag control operation. * To detect this error in the provider, it is necessary to enable "Output failure notification data" in the scanner settings and set the failure notification data character to "ERROR".
Error from the device	0x8010A300	Operation mode error
	0x8010A301	Execution condition error
	0x8010A303	Format error
	0x8010A304	Argument out of range error
	0x8010A400	Execution error
	0x8010AA01	Trigger mode error
	0x8010****	Unexpected error

Appendix A. Communication protocol command correspondence table

CaoController::Execute

Commanded	Communications command
TriggerContinuous	RFUU7
TriggerSingle	RFUU6
Inventory	RFUI
InventoryUII	RFUIU
Read	RFUR
ReadUII	RFURU
Write	RFUW
WriteUII	RFUWU
Lock	RFUL
LockUII	RFULU
Kill	RFUK
KillUII	RFUKU
CommandRelease	RFUS
TriggerOn	TRGON
TriggerOff	TRGOFF
Reset	RESET

Appendix B. Communicatable RF Tags

RF tags that can be communicated with the scanner are ISO/IEC 18000-63 (GS1 EPS Gen2) compliant RF tags. Please refer to the RF tag manual or contact the tag manufacturer for detailed information on the RF tag's memory and size.

Appendix B.1. RF Tag Memory Structure

Bank area	Start sddress[byte]	size[byte]	Contents	Description
User Bank	0	n[6]	user memory	Area where users can store data
TID Bank	0	n[6]	TID	RF tag manufacturer identifier, serial, expansion method, etc. (set by tag manufacturer)
UIIBank	4	n[7]	UII	RF Tag Unique Identifier
	2	2	PC(StoredPC)	Protocol control information. ● 5 bits from 10h-14h to specify UII length
	0	2	CRC-16	Checkdigit for Inventory communication (not user-writable)
Reserved Bank	8	n[6]	No stipulations	-
	4	4	access password	Used for access authentication for Read, Write, and Lock processes
	0	4	kill password	Used to authenticate Kill process

[6] Defined by RF tag specifications.
[7] The address is determined by the first 5 bits (bit address: 10h-14h) of the PC.

Appendix C. Scanner Settings

The following is an example of using Scanner Setting F to set the scanner settings that are set in Section 2.1, "Connecting the Scanner to the Client PC".

- RFID

Classification		Setting item
RFID	RF tag communication operation mode	Upper level control mode

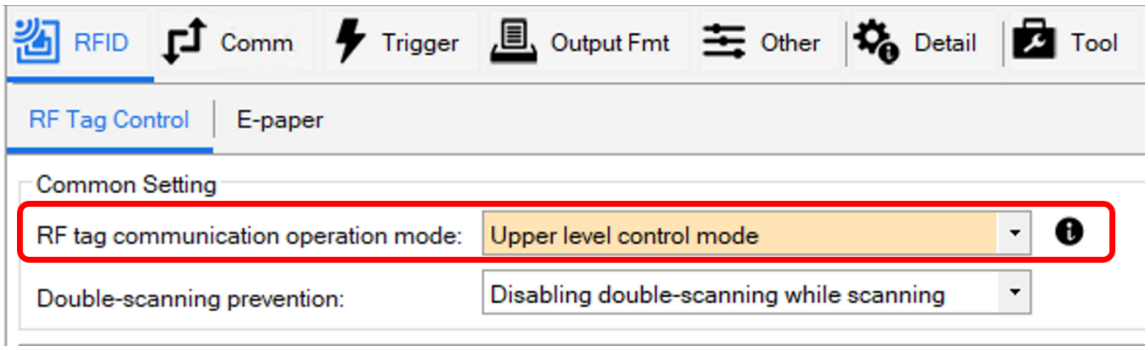


Figure 5-1 RFID Settings

- Communication

Classification				Setting item
Communication	RF tag read data transmission interface			Ethernet
	Ethernet	IP Address Setting	Use the following IP address	Any addresses other than DHCP
			Subnet mask	Optional
			Default gateway	Optional
		Command Control	Protocol	TCP Server
		Read Data Transfer	Protocol	TCP Client

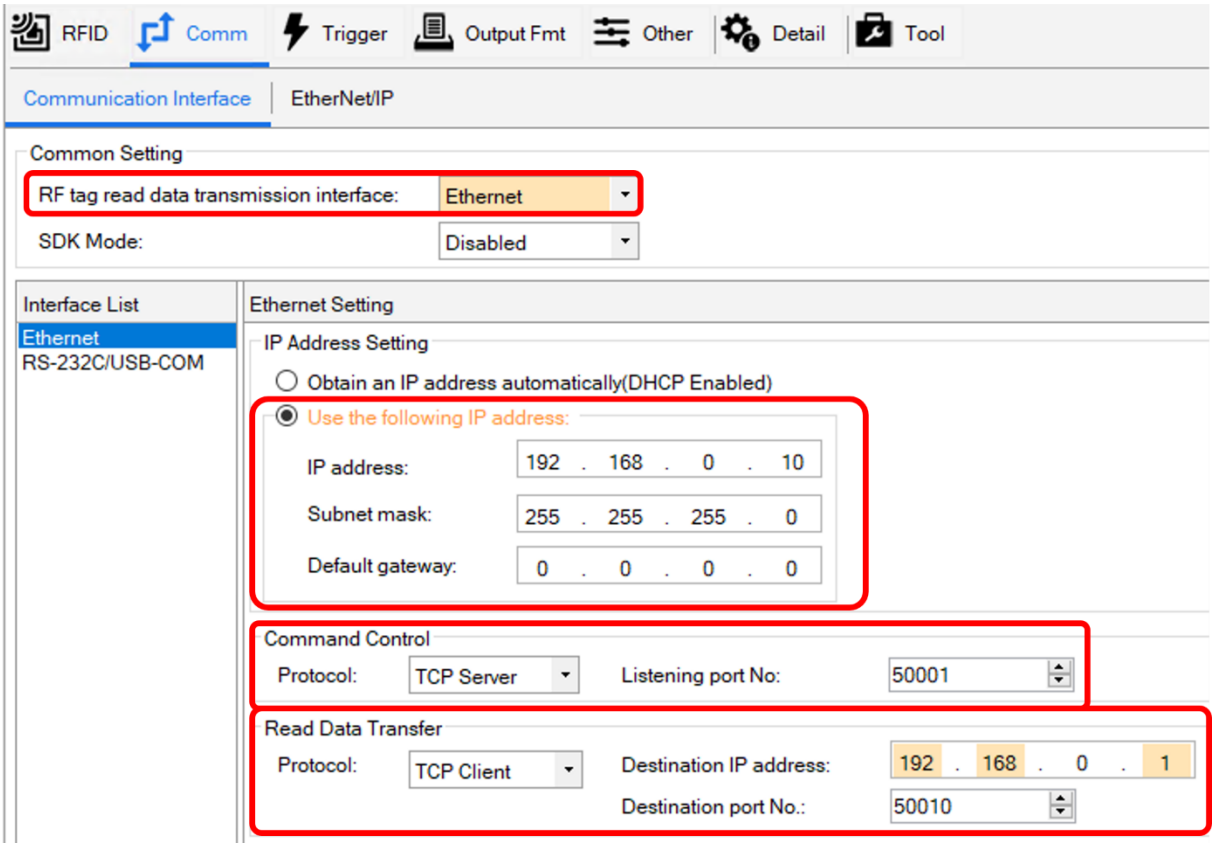


Figure 5-2 Comm settings

- Trigger

Classification			Setting item
Trigger	Trigger type		Command
	Trigger Command Setting	Trigger command response	Enabled
		Trigger ON command	TRGON
		Trigger OFF command	TRGOFF
	Fail Notification Data Output	Fail Notification Data Output	Checked
		Failure notification data	ERROR

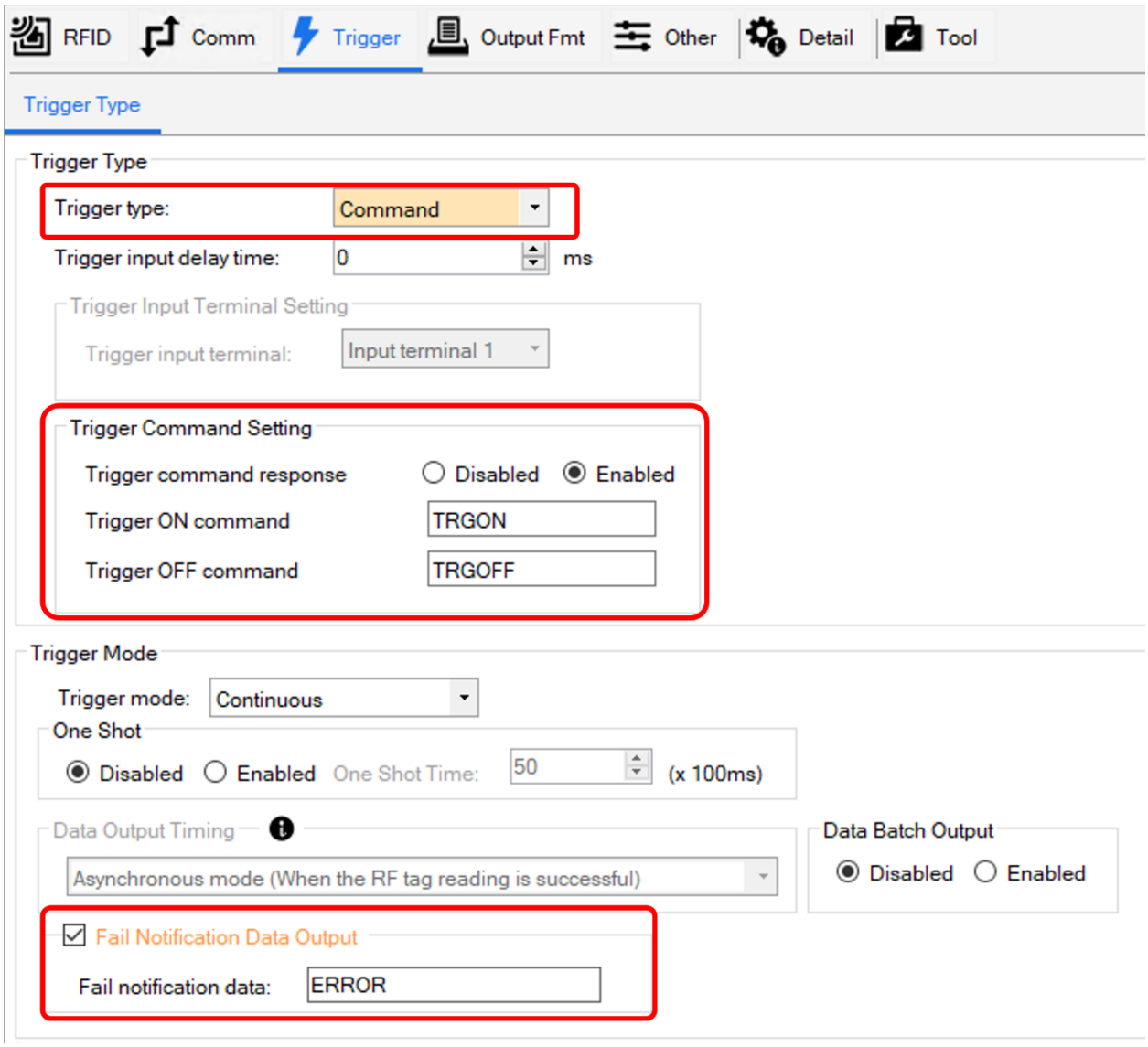


Figure 5-3 Trigger settings

- Output format

Classification		Setting item
Output format	Separator	Comma
	Scanner ID	Enabled
	Time	Enabled
	Optional String 1	No data
	Scanning Condition Number	Enabled
	Communication identifier	Enabled
	Response method	ASCII response

	PC	Enabled
	RSSI	Enabled
	Antenna number	Enabled
	Polarization	Enabled
	Optional String 2	No data