Sugisaki Meter Co., Ltd.

Torque Tester Provider

Version 1.0.1

User's Guide

July 13, 2020

NOTE:

This document was created by automatic translation from Japanese.

[Revision History]

Version	Date	Description
1.0.1	2016-09-30	First edition
1.0.1	2020-07-13	Delete operation check device. Add compatible device.

[Compatible device]

Model	Version	Notes
DI-9M Series		

Table of Contents

1. Introduction

This manual is a user's guide for ORiN providers (SUGISAKI Torque providers) that acquire measured and statistical data from Sugisaki Instrument's Torque Tester.

This document describes the functionality of SUGISAKI Torque provider and the methods it implements.

2. Provider Overview

2.1. Introduction

SUGISAKI Torque provider obtains measured data and statistical data from Sugisaki Instrument's Torque Tester via USB-Serial Communication. The file format of this provider is DLL (Dynamic Link Library), which is detailed in Table 2-1. In addition to the DLL below, a driver for USB serial communication is required. For details, refer to the Sugisaki Instrument User's Manual.

Table 2-1 Tabular SUGISAKI Torque Providers

File name	CaoProvSUGISAKITorque.dll
ProgID	CaoProv.SUGISAKI.Torque
Registry registration	Regsvr32 CaoProvSUGISAKITorque.dl
Deletion of Registry Registration	Regsvr32 /u CaoProvSUGISAKITorque.dl

This provider is broadly divided into two communication modes.

1. Statistical data mode

   In this mode, a message corresponding to the operation is sent at the timing when the user performs some operation, not during the Torque measurement.

2. Real-time mode

   In this mode, the measured value is automatically transmitted as a message during the Torque measurement.

These modes are configured on the device side. When using providers, options for AddController described later must be specified according to the mode of the device.

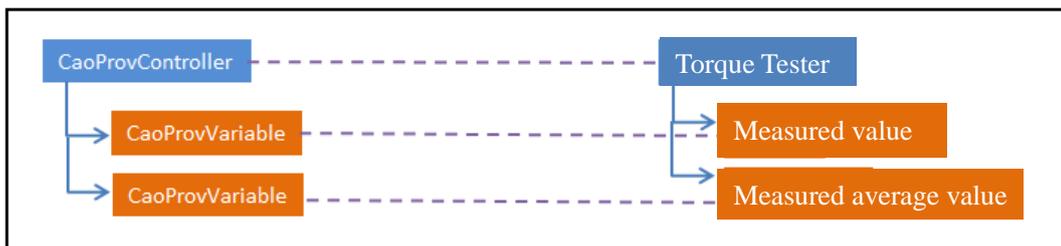Fig. 2-1 shows the correspondence diagram between each class of provider and the torque tester.

Fig. 2-1 Correspondence diagram of each class of provider and torque tester

2.2. Method properties

2.2.1. CaoWorkspace::AddController method

In this provider, the connection parameters are set in CaoWorkspace::AddController and communication is connected. AddController is specifi ed as follows.

SYNOPSIS

```
AddController
(
        <bstrCtrlName:BSTR>,          // [in] Controller name (unique name)
        <bstrProvName:BSTR>,          // [in] provider-name (fixed = "CaoProv.SUGISAKI.Torque")
        <bstrPcName:BSTR>             // [in] Provider execution machine name (unused)
        [,bstrOption:BSTR>]           // [in] Option string
)
```

The following strings are specified in the option string:

Table 2-2 CaoWorkspace::AddController Option-String

Option	Required	Description	Value Range	Default Value
CONN = <Com connection option>	✓	Specifies the port number to be used for USB serial communication. If this option is not specified, an error occurs.	-----	COM number:-- Baud rate: 19200 Parity: NOPARITY Data bit: 8 Stop bit: TWOSTOPBIT Flow control: 0
TIMEOUT = <response wait time>	--	Specifies the response wait time (ms).	1 - 65535	500
MODE = <Operating mode>	--	Specifies the operating mode of the controller. The default behavior is when the specified operation mode is out of range.	1: Measurement mode 2: Real-time mode	1

2.2.1.1. CONN Optional

The following is a CONN optional connection parameter string: Here, braces ("[]") are optional, and the underlined part in the description of each parameter indicates the default value when no options are specified.

Eth

"CONN = COM:<COM number>[:<baud rate>[:<parity>:<data bit>:<stop bit>:<flow control>]]"

<COM number>	:	
<Baud Rate>	:	<u>19200</u>
<Parity>*	:	<u>NOPARITY(0)</u>, ODDPARITY(1), EVENPARITY(2), MARKPARITY(3), SPACEPARITY(4)
<Data bit>	:	4, 5, 6, 7, <u>8</u>
<Stop bit>*	:	ONESTOPBIT(0), ONE5STOPBIT(1), <u>TWOSTOPBIT(2)</u>
<Flow Control>	:	<u>0</u>, 1, 2, 3

2.2.2. CaoController::AddVariable method

Creates a CaoVariable from a CaoController. Only the variables listed in Table 2-3 can be used in variable names.

AddVariable is specifi ed as follows.

SYNOPSIS

AddVariable

(

        <bstrVariableName:BSTR>   // [in] Variable name

        [,<bstrOption:BSTR>])   // [in] Option string

)

2.2.3. CaoController::method get_VariableNames

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

---

* You can also specify the number in parentheses.

2.3. Variable list

2.3.1. CaoController

Table 2-3 lists the variables that can be used in CaoController class in statistical data mode. In addition, only the get operation is possible for all variables.

Table 2-3 List of variables in statistical data mode

Variable		Acquired data		Attribute	
NAME	Introduction	Data Type	Description	Get	Put
@MAKER_NAME	Holds the manufacturer name.	VT_BSTR	String holding the manufacturer's name	✓	-
@VERSION	Keeps the version of the provider.	VT_BSTR	String that holds the version (*. *. *)	✓	-
@LAST_MEAS	Holds the last measured value of the measured value packets received from the device.	VT_ARRAY | VT_VARIANT	--	✓	-
		(VT_UI2	Data No. (It will be 0 when cleared.)		
		VT_R4	Measured value		
		VT_BSTR	Units of Measurement		
@LAST_NUMOFDATA	Number of data packets received from the device The last number of data packets is retained.	VT_UI2	Number of data	✓	-
@LAST_MAXMEAS	Holds the last maximum measured value of the maximum measured value packets received from the device.	VT_ARRAY | VT_VARIANT	--	✓	-
		(VT_R4	Measured value		
		VT_BSTR	Units of Measurement		
@LAST_MINMEAS	Holds the last minimum measurement value of the minimum measured value packets received from the device.	VT_ARRAY | VT_VARIANT	--	✓	-
		(VT_R4	Measured value		
		VT_BSTR	Units of Measurement		
@LAST_AVGMEAS	Holds the last average measurement of the average measured packets received from the device.	VT_ARRAY | VT_VARIANT	--	✓	-
		(VT_R4	Measured value		
		VT_BSTR	Units of Measurement		

Table 2-4 shows the list of variables that can be used in CaoController class in real time mode. In addition, only the get operation is possible for all variables.

Table 2-4 List of variables in real-time mode

Variable		Acquired data		Attribute	
NAME	Introduction	Data Type	Description	Get	Put
@MAKER_NAME	Holds the manufacturer name.	VT_BSTR	String holding the manufacturer's name	✓	-
@VERSION	Keeps the version of the provider.	VT_BSTR	String that holds the version (*. *. *)	✓	-
@LAST_REALTIME	Holds the last real-time of the real-time packets received from the device.	VT_R4	Received single precision floating point value	✓	-

2.4. List of Messages

Table 2-5 shows the list of messages sent from the provider to the engine.

Table 2-5 List of messages

Message		Acquired data	
No.	Meaning	Data Type	Description
3	Measured value reception	VT_ARRAY | VT_VARIANT	--
		VT_UI2	Data No. (It will be 0 when cleared.)
		VT_R4	Measured value
		VT_BSTR	Units of Measurement
4	Receive number of data	VT_UI2	Number of data
5	Maximum measured value reception	VT_ARRAY | VT_VARIANT	--
		VT_R4	Measured value
		VT_BSTR	Units of Measurement
6	Minimum measurement value reception	VT_ARRAY | VT_VARIANT	--
		VT_R4	Measured value
		VT_BSTR	Units of Measurement
7	Mean value received	VT_ARRAY | VT_VARIANT	--
		VT_R4	Measured value
		VT_BSTR	Units of Measurement
8	Real-time data reception	VT_R4	Received single precision floating point value
9	Unexpected packet reception This message is also sent in the following cases. ・Real-time data is received in statistical data mode. ・Statistical data is received to the real-time mode self.	VT_BSTR	Received packet

2.5. Error code

The provider has the following unique error codes: (Refer to Table 2-6 Unique error codes.)

For ORiN2 common errors, refer to the Error Codes section of ORiN2 Programming Guide.

Table 2-6 Unique error codes

Error Number	Description
0x80110001	Conn option is not specified.
0x80110002	Conn option-specification is invalid.
0x80110003	An attempt was made to add a variable that cannot be added.

3. Sample program

The following describes a sample program for acquiring measured value data.

```
Prerequisites:
 ・The Com port shall be 1.
```

List 3-11	Sample.frm

```
Dim eng As CaoEngine
Dim ctrl As CaoController
Dim var As CaoVariable
Private Sub Form_Load()
        Set eng = New CaoEngine
        ' Create Controller Object
        Set ctrl = eng.Workspaces(0).AddControlle(
                "Sugisaki", "CaoProv.SUGISAKI.Torque", "", "CONN=COM:1)
        ' Create Variable Object
        Set var = ext.AddVariable(
                "@LAST_MEAS", "")
End Sub


' Obtain 'Value'
Private Sub Command1_Click()
        ' Get variable value
        Dim lastMeas As Variant
        Set lastMeas = var.Value
End Sub
```