WMF204C provider

For measurement made by MettlerToledo company module

Version 1.0.2

User's guide

November 1, 2018

Remarks:

This document is translated from Japanese into English by the machine translation.

【 revision history 】

Version	Date	Content
1.0.0	2017-08-08	First edition.
1.0.1	2018-02-13	Bug fixes.
1.0.2	2018-11-01	Memory leak was corrected.

【 ..confirming the operation.. equipment 】

Model	Software version	Notes
WMF204C-W/IE	1.0.1.20160629	It has confirmed the operation.

- This provider is confirming the operation in "WMF204C-W/IE".

It doesn't operate theoretical not be nor the one to guarantee operation though the real machine verification doesn't execute the cereal device.

Contents

1. Introduction

This book is an user's guide of the WMF204C provider that is the CAO provider for the measurement made by MettlerToledo company module.

The WMF204C provider receives the transmission and the response of the command to the interface unit, and notifies the event.

2. Outline of provider

2.1. Outline

The WMF204C provider is CAO provider that connects Ethernet/IP(TCP) or the cereal with the measurement made by MettlerToledo company module, and write/reads data.

When executing it, the CaoController::Execute method generates automatically, and transmits the command. Moreover, the response is analyzed, and data is acquired. It is continuous or the one that there is a response in the asynchronization notifies the command the content of the response as an event.

The file format becomes DLL(Dynamic Link Library), and when using it from the CAO engine, is loaded dynamically. When the WMF204C provider is used, it is necessary to install ORiN2SDK or register the registry by the hand work referring, refer to the Table2-1 for details.

Table 2-1 WMF204C provider

File name	CaoProvMETTLERTOLEDOWMF204C.dll
ProgID	CaoProv.METTLERTOLEDO.WMF204C
Registry registration	regsvr32 CaoProvMETTLERTOLEDOWMF204C.dll
Blotting out of registry registration	regsvr32 /u CaoProvMETTLERTOLEDOWMF204C.dll

2.2. Method property

2.2.1. CaoWorkspace::AddController method

The WMF204C provider connects the communication referring to connected parameter for the communication at AddController.

Format AddController (< bstrCtrlName:VT_BSTR >, < bstrProvName:VT_BSTR >,
<bstrPcName:VT_BSTR > ,<bstrOption:VT_BSTR>)

bstrCtrlName : The in controller name is arbitrary.
bstrProvName : Provider name . fixation value ="CaoProv.METTLERTOLEDO.WMF204C"
bstrPcName : Execution machine name of in provider
bstrOption : In optional character string

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

Table 2-2 Optional character string of CaoWorkspace::AddController

[1]Option	Explanation
Conn =< connected parameter >	Indispensability. A communication form and the connected parameter are set. Please refer to caputer 2.2.1.1 for details.
ConnTimeout =< timeout period >	The timeout period when TCP is connected is specified. (millisecond) (default: 3000)
Timeout =< timeout period >	The timeout period when sending and receiving it is specified. (millisecond) (default: 3000)

[1] In the square bracket (""), a possible omission is shown. The underlined part under the explanation of each parameter becomes a default value when the option is not specified.

2.2.1.1. Conn optional.

A communication form and connected parameter are specified.

Connected parameter character string of optional Conn is shown as follows.

Ethernet/IP device

"Conn=ETH:<Dest IP Address>[:<Dest Port No>]"

"Conn=TCP:<Dest IP Address>[:<Dest Port No>]"

< Dest IP Address >	:	Internet Protocol address TCP/IP connection ahead. Example:"127.0.0.1","192.168.0.1"
<Dest Port No>	:	TCP/IP connection port number. Example: 80,5006,5007

Cereal device

"Conn=com:<COM Port>[:<BaudRate>[:<Parity>:<DataBits>:<StopBits>]]"

<COM Port>	:	COM port number. Example:‘1'-COM1,‘2'-COM2, …
<BaudRate>	:	Baud rate. Example: 110,300,600, 1200,2400,4800,9600, 19200, and 38400
<Parity>	:	Parity. Example:‘N'-NONE,‘E'-EVEN,‘O'-ODD
<DataBits>	:	Number of data bits. Example:‘7'-7bit,‘8'-8bit.
<StopBits>	:	Number of stop bits. Example:‘1'-1bit,‘2'-2bit.

(example 1)	"com:1"	Communication port COM1 (, 57600bps, None, 8bits, 1bit)
(example 2)	"com:2:9600"	Communication port COM2 and 9600bps (, None, 8bits, 1bit)
(example 3)	"com:3:38400:N:8:2"	Communication port COM3, 38400bps, and None, 8bits, and 2bit

2.2.2. CaoController::AddVariable method

The AddVariable method of the CaoController class is a method for making the variable object write/to read data to the measurement module.

Please refer to 2.3 variable list for the variable identifier that can be specified.

Format AddVariable (< bstrVariableName:VT_BSTR > , < bstrOption: VT_BSTR >)

bstrVariableName : [in] variable identifier

bstrOption : [in] optional character string

2.2.3. CaoController:: Execute method

The Execute method of the CaoController class is a method for the execution of the command.

Please refer to capter 3 for the details.

Format Execute(<bstrCommandName:VT_BSTR>[,vntParam:VT_VARAINT>])

bstrCommandName: [in] command name

vntParam : [in] parameter

2.2.4. CaoController::get_VariableNames property

Acquires the variable identifier list shown in Table 2-3.

2.2.5. CaoController::get_Value property

The data specified by the variable identifier is acquired.

2.2.6. CaoController::put_Value property

The value passed to the data specified by the variable identifier by the argument is set.

2.2.7. CaoController::OnMessage event

The OnMessage event of the CaoController class is generated if it receives it repeating data from the measurement module.

2.3. Variable list

2.3.1. CaoController class

Table 2-3 CaoController class variable list

Variable identifier	Data type	Explanation	Attribute	
			get	put
@MAKER_NAME	VT_BSTR	Return maker name. "METTLER TOLEDO"	✓	-
@VERSION	VT_BSTR	Provider version information is returned.	✓	-
@CMDS_LIST	VT_BSTR | VT_ARRAY	All executed MT-SICS commands The list is returned.	✓	-
@MTSICS_INFO	VT_BSTR	MT-SICS level and MT-SICS version are returned.	✓	-
@DEVICE_DATA	VT_BSTR	The balance data (type and amount of the leopard) It returns it.	✓	-
@SW_VERSION	VT_BSTR	The software version and the type definition number of the balance are returned.	✓	-
@SERIALNO	VT_BSTR	The serial number is returned.	✓	-
@MATERIALNO	VT_BSTR	The SW identification number is returned.	✓	-
@WEIGHT	VT_R4 | VT_ARRAY	A present stability net weight value is returned. "Weight value and unit *"	✓	-
@WEIGHT_IMM	VT_R4 | VT_ARRAY	A present net weight value is returned regardless of the stability of the balance. "Weight value, unit *, and status (0: Stability/1: Instability)"	✓	-
@TARE	VT_R4 | VT_ARRAY	Tare and the tare weight value is returned. "Weight value and unit *"	✓	-
@TAREVALUE	VT_R4 | VT_ARRAY (get/put)	Get: A present tare weight value is returned. Put: The preset of the tare weight value is executed. "Weight value and unit *" (get/put)	✓	✓

@TARE_IMM	VT_R4 | VT_ARRAY -	Tare [hi] coming [wo] is executed, and the tare value in which tare [hi] coming [wo] is executed is returned at once. "Weight value, unit *, and status (0: Stability/1: Instability)"	✓	-

* Please refer to Table2-1.

Table2-1Table for character string ⇔ numerical value of unit system

Value	Meaning
0	Gram g
1	Kilogram kg
2	Meter ton t
3	Milligram mg
4	Microgram μg
5	Carat ct
6	Newton N
7	Pound lb of using regularly
8	Ounce oz of using regularly
9	Troy ounce ozt
10	Grain GN
11	Pennyweight dwt
12	Monme mom
13	Mesghal msg
14	Hong Kong tail tlh
15	Singapore tail tls
16	Taiwanese tail tlt
17	Tical tcl
18	Tola tola
19	Baht baht
25	Unit none--
26	It is effective in piece PCS "Number" application.

27	It is effective in percent % "Percent" application.
28	Unit 1 of the custom When unit 1 of the cu1 custom turns on, it is effective for (M22).
29	Unit 2 of the custom When unit 2 of the cu2 custom turns on, it is effective for (M22).

3. Command reference

3.1. Controller class

Table3-1CaoController::Execute command list

Command	Function	Page
Cancel	The measurement module is reset in the state detected after the switch is turned on. However, the zero out is not executed.	P.14
AllCancel	The command is canceled while executing all.	P.14
GetCommandsList	The list of all the executed MT-SICS commands is acquired.	P.15
GetMTSICSInfo	The MT-SICS level and the MT-SICS version are acquired.	P.15
GetDeviceData	The balance data is acquired.	P.16
GetSWVersion	The software version and the type definition number of the balance are acquired.	P.16
GetSerialNo	The serial number is acquired.	P.18
GetMaterialNo	The SW identification number is acquired.	P.18
GetWeight	A present stability net weight value is acquired.	P.19
GetImmediately	A present net weight value is acquired regardless of the stability of the balance.	P.19
GetImmediatelyRepeat	It acquires it repeating the net weight value regardless of the stability of the balance.	P.20
GetRepeat	A present stability weight value is acquired continuously at each weight change.	P.20
Tare	Tare is executed.	P.22
GetTareWeightValue	A present tare weight value is acquired.	P.22
PutTareWeightValue	It is a preset as for the tare weight value.	P.24
ClearTare	The tare value is cleared.	P.24
TareImmediately	Tare comes at once.	P.25
Zero	The balance is set to 0.	P.25
ZeroImmediately	The balance is set to 0 at once regardless of the stability of the balance.	P.26

3.1.1. CaoController::Execute("Cancel") command

The measurement module is reset in the state detected after the switch is turned on. However, the zero out is not executed.

Format Cancel ()

Return value : None

Usage

example

```
ctrl.Execute("Cancel");
```

3.1.2. CaoController::Execute("AllCancel") command

The command is canceled while executing all.

Format AllCancel ()

Return value : None

Usage

example

```
ctrl.Execute("AllCancel");
```

3.1.3. CaoController::Execute("GetCommandsList") command

The list of all the commands executed with present software is acquired.

Format GetCommandsList ()

Return value : List of out command (VT_BSTR | VT_ARRAY).
"Number at MT-SICS level that command belongs" and "Command name"
Example) 0 "@"
0 "C"

Usage example

```
string[] varNames = (string[])ctrl.Execute("GetCommandsList");
```

3.1.4. CaoController::Execute("GetMTSICSInfo") command

The MT-SICS level and the version are acquired.

Format GetMTSICSInfo ()

Return value : Present MT-SICS level and out version (VT_BSTR).
"<Level>"?"<V0>"?"<V1>"?"<V2>"?"<V3>"
Example), "0123", "2.30", "2.22", "2.33", and "2.20"

Name	Type	Value	Meaning
<Level>	Character string	0	MT-SICS level 0
		01	MT-SICS level 0 and 1
		012	MT-SICS level 0 and 1 and 2
		03	MT-SICS level 0 and 3
		013	MT-SICS level 0 and 1 and 3
		0123	MT-SICS level 0 and 1, 2, and 3
		3	Application device MT-SICS level 3

<V0>～<V3>	Character string		MT-SICS version at relating level (0-3)

Usage

example

```
string varName = (string)ctrl.Execute("GetMTSICSInfo")
```

3.1.5. CaoController::Execute("GetDeviceData") command

The balance data of the measurement limit etc. (type and amount of the leopard) is acquired.

Format GetDeviceData ()

Return value : Type and amount (VT_BSTR) of leopard of out balance.
"Type", "Amount of leopard", and "Unit of weight"
Example) and "WMF204C-W/IE 220.9000 g"

Usage

example

```
string varName = (string)ctrl.Execute("GetDeviceData")
```

3.1.6. CaoController::Execute("GetSWVersion") command

The software version and the type definition number are acquired.

Format GetSWVersion ()

Return value : Software version and type definition number of out balance (VT_BSTR).
"Software (firmware) version" and "Type definition number"
Example) and "1.0.1.20160629 53.0.2.3695.1603"

Usage

example

```
string varName = (string)ctrl.Execute("GetSWVersion")
```

3.1.7. CaoController::Execute("GetSerialNo") command

The serial number of the balance is acquired.

Format GetSerialNo ()

Return value : Out serial number (VT_BSTR).
Example) and "B649408468"

Usage example

```
string varName = (string)ctrl.Execute("GetSerialNo")
```

3.1.8. CaoController::Execute("GetMaterialNo") command

The identification number of software is acquired.

Format GetMaterialNo ()

Return value : SW identification number with out index (VT_BSTR).
Example) and "30131892E"

Usage example

```
string varName = (string)ctrl.Execute("GetMaterialNo")
```

3.1.9. CaoController::Execute("GetWeight") command

A present stability net weight value is acquired.

Format GetWeight ()

Return value : Present stability weight value for unit actually set by unit of out host (VT_R4 | VT_ARRAY).
Weight value and unit * return by the array.
- The conversion of the unitTable2-1Please refer to [wo].
The example) For the stability weight value 0.9915g「0.9915,0」

Usage

example

```
float[] fData = (float[])ctrl.Execute("GetWeight")
```

3.1.10. CaoController::Execute("GetImmediately") command

A present net weight value is acquired regardless of the stability of the balance.

Format GetImmediately ()

Return value : Stability weight value for unit actually set by unit of out host (VT_R4 | VT_ARRAY).
The weight value, the unit, and status (0: stability/1: instability) return by the array.
Example) "0.9953,0,0" in case of stability weight value 0.9953g.
For the weight value 0.9938g of no stability of present (dynamic) 「0.9938,0,1」.

Usage

example

```
float[] fData = (float[])ctrl.Execute("GetImmediately")
```

3.1.11. CaoController::Execute("GetImmediatelyRepeat") command

It acquires it repeating the net weight value regardless of the stability of the balance.

Format GetImmediatelyRepeat ()

Return value : None
The net weight value, the unit, and status (0: stability/1: instability) are notified in the event (VT_R4 | VT_ARRAY).
The identifier of the OnMessage event becomes 11.
When the error is received from the measurement module, ("..error code (VT_I4).. ..the event notification... Table4-1Refer to".)

Usage example

```
ctrl.Execute("GetImmediatelyRepeat")
```

3.1.12. CaoController::Execute("GetRepeat") command

It acquires it continuing a present stability weight value at each weight change.

Format GetRepeat (<PresetValue>)

< PresetValue> : In none (VT_EMPTY or VT_NULL).
The continuance acquisition is done at each weight change a present stability weight value.
When the preset value is not input, the weight change is 12.5 or more of the the last stability weight value and 30 minimum = d It should be %.
: Array of preset value and unit of in (VT_R4 | VT_ARRAY).
A present stability weight value is continuously acquired at each weight change more than a preset value and a value not steady (dynamic).

Return value : None
The stability weight value, the unit, and status (0: stability/1: instability) are notified in the event (VT_R4 | VT_ARRAY).
The identifier of the OnMessage event becomes 12.

When the error is received from the measurement module, ("..error code (VT_I4).. ..the event notification... Table4-1Refer to".)

Usage

example

```
ctrl.Execute("GetRepeat")
ctrl.Execute("GetRepeat", new float[]{10.00, 0})
```

3.1.13. CaoController::Execute("Tare") command

Tare [hi] of the balance comes.

Format Tare ()

Return value : New tare weight value of out (VT_R4 | VT_ARRAY).
The weight value and the unit return by the array.
Example) New "0.9928,0" in case of tare weight value 0.9928g.

Usage

example

```
float[] fData = (float[])ctrl.Execute("Tare")
```

3.1.14. CaoController::Execute("GetTareWeightValue") command

A present tare weight value is acquired.

Format GetTareWeightValue ()

Return value : For the unit actually set by the unit of the out host, a present tare weight value of the tare memory (VT_R4 | VT_ARRAY).
The weight value and the unit return by the array.
Example) Present "0.9928,0" in case of tare weight value 0.9928g.

Usage

example

```
float[] fData = (float[])ctrl.Execute("GetTareWeightValue")
```

3.1.15. CaoController::Execute("PutTareWeightValue") command

It is a preset as for the tare weight value.

Format PutTareWeightValue (<TarePresetValue>)

<TarePresetValue> : Array of preset value and unit of in (VT_R4 | VT_ARRAY).
Return value : For the unit actually set by the unit of the out host, a present tare weight value of the tare memory (VT_R4 | VT_ARRAY).
The weight value and the unit return by the array.

Usage

example

```
float[] fData = (float[])ctrl.Execute("PutTareWeightValue",
                                      new float[]{100.00, 0})
```

3.1.16. CaoController::Execute("ClearTare") command

The tare memory is cleared.

Format ClearTare ()

Return value : None

Usage

example

```
ctrl.Execute("ClearTare")
```

3.1.17. CaoController::Execute("TareImmediately") command

Tare [hi] coming [wo] of the balance is done to the stability of the balance regardless at once.

Format TareImmediately ()

Return value : Tare value in which out tare [hi] coming [wo] is executed (VT_R4 | VT_ARRAY).
The weight value, the unit, and status (0: stability/1: instability) return by the array.
Example) "0.993,0,0" that executes tare [hi] coming [wo] in case of stability tare value 0.9930g.
Dynamic) tare value 1.0921 ..(.. g not steady that executed tare [hi] coming [wo]
Drinking "1.0921,0,1" at the time of it is time when.

Usage

example

```
float[] fData = (float[])ctrl.Execute("TareImmediately")
```

3.1.18. CaoController::Execute("Zero") command

The balance is set to 0.

Format Zero ()

Return value : None

Usage

example

```
ctrl.Execute("Zero")
```

3.1.19. CaoController::Execute("ZeroImmediately") command

New 0 is set regardless at once as the stability of the balance.

Format ZeroImmediately ()

Return value : Condition of out zero out (VT_I2).
= 0: The zero out is executed again on a steady condition.
= Stabilize..dynamic..)..condition..zero out..execute.

Usage

example

```
int i2Data = (int)ctrl.Execute("ZeroImmediately")
```

4. Error code list

The following and peculiar the error code is defined in the WMF204C provider.

Table4-1Peculiar error code

Error name	Error number	Explanation
Receive data lack	0x80100001	When the lack (less than minimum data size etc.) is seen in receive data, it is returned.
Syntax error (general)	0x80100200	The measurement module was not able to recognize the received command.
Transmission error (general)	0x80100201	The measurement module received a for instance wrong because of broke down the parity error and the interface command.
Logical error (general)	0x80100202	The measurement module cannot execute the received command.
Overload	0x80100203	The measurement module is an overload (measurement range excess).
Loading under	0x80100204	The measurement module loads it under (The measurement plate is not installed).
Logical error (command peculiarity)	0x80100205	Parameter etc. not permitted.
Internal error	0x80100206	The measurement module is not prepared.
Reference setting error	0x80100207	The setting of the reference was interrupted.
Electronic unit (measurement module) error	0x801003XX	The error occurred in the electronic unit (measurement module) by the response of weight. The error code according to the product is inserted in XX. Example) Boot error (1) →0x80100301 Brand error (2) →0x80100302 Checksum error (3) →0x80100303 Error (9) of option →0x80100309 →0x8010030a Disagreement (11) of device →0x8010030b Hot plug out (12) →0x8010030c Measurement module/electronic unit disagreement

		(14) →0x8010030e Adjustment main point (15) →0x8010030f
Instruction meter error	0x801004XX	The error occurred in the instruction meter by the response of weight. The error code according to the product is inserted in XX. Example) Boot error (1) →0x80100401 Brand error (2) →0x80100402 Checksum error (3) →0x80100403 Error (9) of option →0x80100409 →0x8010040a Disagreement (11) of device →0x8010040b Hot plug out (12) →0x8010040c Measurement module/electronic unit disagreement (14) →0x8010040e Adjustment main point (15) →0x8010040f