Toshiba Tec Corporation

TPCL Provider

Version 1.0.0

User’s Guide

September 28, 2021

Remarks:

[Revision History]

Version	Date	Contents
1.0.0	Sep. 28, 2021	1st edition

[Operation Checked Printers]

Model	Precautions
B-EX4T1	

Table of Contents

1. Introduction

This document is the User’s Guide of the CAO provider which performs data writing, reading out and printing to label printers manufactured by Toshiba Tec Corporation (hereafter, called "Toshiba Tec"). The CAO provider (CaoProvTECTPCL.dll) dealing with in this document is called the "TPCL provider".
TPCL providers use BCP Print For.NET to communicate with Toshiba Tech label printers.
BCP Print For.NET is shipped with ORiN2SDK and does not need to be installed separately.
In addition, TPCL providers only support label formats created with BCP print system V2 or Label Format Generation Tool.
This provider was developed using the version of BCP Print For.NET shown below.
Different versions may cause the provider to function incorrectly. If you install BCP Print For.NET separately, install the same version of the software.
BCP Print For .NET V2.0.18

1.1. Setting Up of the PC Environment

To use TPCL providers, register TPCL providers according to 2.1 Overview.

Fig. 1-1 shows the connection example of the TPCL provider and label printer. Use the Ethernet to connect the label printer with the PC.

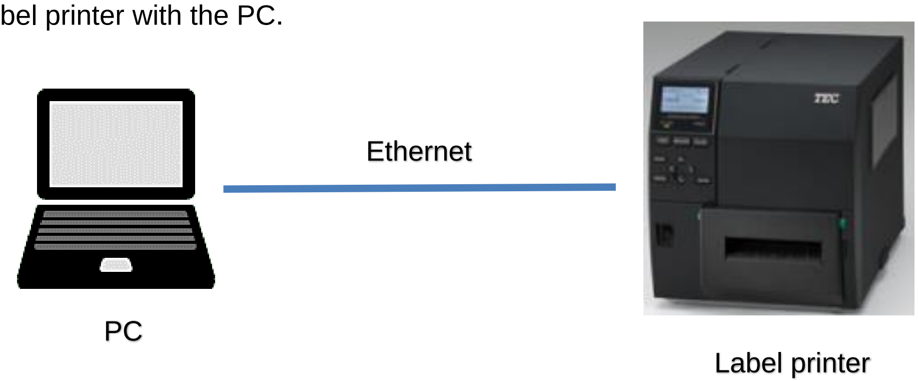


Fig. 1-1: Connection image

2. Provider Overview

2.1. Overview

The TPCL provider is a CAO provider which performs data writing, reading out and printing to label printers manufactured by Toshiba Tec, using the BCP Print For .NET. Its file format is DLL (Dynamic Link Library) and is dynamically loaded from the CAO engine when the TPCL provider is used. To use TPCL providers, you must install ORiN2SDK or register the registry by referring to the table below.

Table 2-1: TPCL provider

File name	CaoProvTECTPCL.dll
ProgID	CaoProv.TEC.TPCL
Registry registration[1]	RegistAsm.bat CaoProvTECTPCL.dll
Deletion of registered registry	UnregistAsm.bat CaoProvTECTPCL.dll

[1] Patch files such as RegistAsm.bat or UnregistAsm.bat are stored under {ORiN2InstallationFolder}¥DotNet¥BAT.

2.2. Method Property

2.2.1. CaoWorkspace::AddController method

A controller object to perform data writing, reading out and printing by connecting with a label printer can be created. Parameters when connecting with the label printer can be specified by the option character string.

Format AddController (<bstrCtrlName:BSTR>,<bstrProvName:BSTR>,
<bstrPcName:BSTR>, [<bstrOption:BSTR>])

< bstrCtrlName >	:	[in] Controller name
< bstrProvName >	:	[in] Provider name (Fixed value: "CaoProv.TEC.TPCL")
<bstrPcName>	:	[in] Provider execution PC name (not used)
<bstrOption>	:	[in] Option character string

Example

```
// Create engine
var eng = new CCaoEngine();
// Create workspace
var ws = eng.AddWorkspace("SampleWorkSpace","");

// Connection option settings
var option = @"Conn=TCP:192.168.1.1:8000, Timeout=5, LfmFilePath=C:¥TestLablel.lfm ";

// Connect to printer
this.ctrl = ws.AddController("Sample","CaoProv.TEC.TPCL","",option);
```

The table below shows the list to be specified for the option character string.

Table 2-2: Option character string for CaoWorkspace::AddController

Option	Meaning
Conn=<Connection parameter>	Mandatory. Sets the communication type and its connection parameter. (Reference: 2.2.1.1)
Timeout[=<Timeout interval>]	Arbitrarily. Specifies the waiting time (seconds) for the response from the label printer. (Default: 30) 1 to 9999 : Waits for a response from the label printer for the specified time (seconds). 0 : No timeout has occurred. * The value is available only when Return upon the Completion of Printing is set for Issue Mode.

IssueMode[=<Issue mode>]	Arbitrarily. Specifies the mode for label issuing. (Default: 1) 1 : Return upon the Completion of Sending 2 : Return upon the Completion of Printing
LfmFilePath[=<Label format file path>]	Arbitrarily. Specifies the label format file name by the absolute path. * In the case of the Issue command, make sure to specify this.
LoadPCIndex[=<PC save ID number>]	Arbitrarily. Specifies the save ID number (1 to 99) read out from the label printer.
SystemPath[=<Printer information files stored path setting>]	Arbitrarily. Specifies the storage path for the printer information files. (Default: {ORiN2SDK installation folder}¥CAO¥ ProviderLib¥ToshibaTec¥TPCL¥Bin)
UsePrinter=<Printer number>	Arbitrarily. Specifies the printer number to be used. (Default: 99) 5 : B-458-TS 7 : B-858 8 : B-SX5 10 : B-SA4T 11 : B-EP2G 12 : B-EP4T 15 : B-458TR 16 : B-858TR 17 : B-SX8T 18 : B-EV4-T 19 : B-EV4-G 21 : B-EX4T1-T 22 : B-SX8R 23 : B-LV4-T 24 : B-LV4-G 28 : B-FP2D-G 32 : B-EX4T3-H 34 : B-EX6T1-T 37 : BA400T-G 38 : BA400T-T 39 : BV400-G 40 : BV400-T 99 : LFM defined

GraphicType[=<Graphic data type>]	Arbitrarily. Sets the graphic data type. (Default: 0) 0 : Nibble mode (4 dots/byte) 1 : Hex mode (8 dots/byte) 2 : TOPIX compression mode
ControlCode[=<Command control code system>]	Arbitrarily. Sets the command control code system. (Default: 0) 0 : ESC (1BH), LF (0AH), NULL (00H) system 1 : (7BH), | (7CH), } (7DH) system

2.2.1.1. Conn option

The character string of the connection parameter for the Conn option is as follows.

[Ethernet device]

TCP/IP

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

< Dest IP Address >	:	IP address of the connection destination
< Dest Port No >	:	Port number of the connection destination

2.2.2. CaoController::GetVariableNames property

The list of the system variable names shown in 2.3 can be obtained.

Example

```
var varNames = this.ctrl.get_VariableNames();
```

2.2.3. CaoController::AddVariable method

A variable object to perform writing and reading out of the communication settings of the label printers and the setting values of a label format can be created.

Format AddVariable (<bstrVariableName:BSTR>, [<bstrOption:BSTR>])

<bstrVariableName >	:	[In] Variable name
<bstrOption>	:	[in] Option character string (Not used for this command)

Example

```
var controlCode = this.ctrl.AddVariable("@CONTROL_CODE", null);
```

2.2.4. CaoController::Execute method

The Execute method of the CaoController class is to execute the command. For details about each command, see "Command Reference".

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

<bstrCommandName> : [in] Command name

<vntParam> : [in] Parameter

2.2.5. CaoController::OnMessage event

When a status is received from a label printer, it is notified as an OnMessage event of the CaoController class.

This event will be generated in the following cases.

Table 2-3: OnMessage generation timing

Generation timing	Details
When issuing is completed and an error occurs (Status auto-transmission)	This will be generated in the following timing: - When the label has been fed - When the printing of the specified number of labels has been completed for batch printing - When the printing of one sheet of label has been completed for peel-off printing - When an error has occurred
When a status upon request	- When a status is requested by the GetStatus command and it is returned from a label printer - When a 13-byte status is obtained
When communication has failed	When writing into a port has failed due to a buffer-full situation

* No OnMessage event will be generated if the communication setting of a label printer is COM, Socket, USB, Bluetooth, Return upon the Completion of Printing, Driver, or File.

2.2.5.1. Details of the messages

The following tables show the type of message and the property which can be obtained by the message.

- **Printer status reception**

Number	:	1
Value	:	0x800A044E
Description	:	Status character string received from a label printer (see 5.1)
DateTime		Time stamp
Destination		Null
Source		IP address of the connection label printer destination

- **Other message reception**

Number	:	2
Value	:	Message code received from a label printer
Description	:	Null
DateTime		Time stamp
Destination		Null
Source		IP address of the connection label printer destination

2.2.6. CaoVariable::get_Value property

Values which correspond to the variable for the system and users shown in 2.3 can be obtained.

Example

```
var val = (int)this.value.Value;
```

2.2.7. CaoVariable::put_Value property

Values which correspond to the variable for the system and users shown in 2.3 can be set.

Example

```
int setValue = 0;
this.value.Value = tempValue;
```

2.3. Variable List

2.3.1. CaoController class

Table 2-4: CaoController class system variable list

Variable name	Data type	Description	Attribute	
			get	put
@MAKER_NAME	BSTR	Returns "TOSHIBA TEC".	✓	-
@VERSION	BSTR	Returns the version of the provider.	✓	-
@PORT_SETTING	BSTR	Returns the character string of the communication parameter setting.	✓	-
@RECV_TIMEOUT	I4	Obtains and specifies the receive timeout period (seconds).	✓	✓
@ISSUE_MODE	I4	Returns the issue mode. 1 : Return upon the Completion of Sending 2 : Return upon the Completion of Printing	✓	-
@LFM_FILE_PATH	BSTR	Obtains and specifies the absolute path of the format file.	✓	✓
@LOAD_PC_INDEX	I4	Obtains and specifies the PC save ID number.	✓	✓
@SYSTEM_PATH	BSTR	Returns the storage path for the printer information files.	✓	-
@USE_PRINTER	I4	Returns the setting value of the printer number to be used. 5 : B-458-TS 7 : B-858 8 : B-SX5 10 : B-SA4T 11 : B-EP2G 12 : B-EP4T 15 : B-458TR 16 : B-858TR 17 : B-SX8T 18 : B-EV4-T 19 : B-EV4-G 21 : B-EX4T1-T	✓	-

		22 : B-SX8R 23 : B-LV4-T 24 : B-LV4-G 28 : B-FP2D-G 32 : B-EX4T3-H 34 : B-EX6T1-T 37 : BA400T-G 38 : BA400T-T 39 : BV400-G 40 : BV400-T 99 : LFM defined		
@GRAPHIC_TYPE	I4	Obtains and specifies the graphic data type. 0 : Nibble mode (4 dots/byte) 1 : Hex mode (8 dots/byte) 2 : TOPIX compression mode	✓	✓
@CONTROL_CODE	I4	Obtains and specifies the command control code system. 0 : ESC (1BH), LF (0AH), NULL (00H) system 1 : (7BH), | (7CH), } (7DH) system	✓	✓
@PRINTER_STATUS	BSTR	Returns the character string of the printer status obtained by means of the OnMessage event or GetStatus command. * If the character string of the printer status is not obtained since the OnMessage event has not been generated or the GetStatus command has not been carried out, Null will be returned.	✓	-
@STATUS_AUTO_SEND_NO	BSTR	Returns the status auto-transmission portion of the character string of the printer status. * If the character string of the printer status has not been obtained, Null will be returned.	✓	-
@STATUS_REQ_NO	BSTR	Returns the status requirement portion of the character string of the printer status. * If the character string of the printer status has not been obtained, Null will be returned.	✓	-

@STATUS_TYPE	I4	Returns the status type portion of the character string of the printer status. 1 : When a status upon request 2 : When a status auto-transmission * If the character string of the printer status has not been obtained, Null will be returned.	✓	-
@STATUS_ISSUE_EXPECTED_COUNT	BSTR	Returns the number of labels that are yet to be printed of the character string of the printer status.	✓	-
@RESULT_CODE	I4	Returns the result number returned from the printer.	✓	-

* When @LFM_FILE_PATH or @LOAD_PC_INDEX is specified, the label format linked with the value set at the last time will take precedence at the time of the execution of the Issue command.

Table 2-5: CaoController class user variable list

The setting of the print data can be made by specifying the object name in the label format for which print data are specified in the variable name.

Variable name	Data type	Description	Attribute	
			get	put
Object number in the label format for which print data are specified.	BSTR	Creates the item to obtain and specify the print data to be set to the label format object. * Up to 2000 digits can be specified. If the object is a bitmap, specify the absolute path of the file.	✓	✓

2.4. Error Code

There are no error codes unique for the TPCL provider. However, an error code defined by the BCP Print For .NET may be returned. For details about the error of the BCP Print For .NET, refer to "Chapter 6: Details about the Printer - 2. Messages" in the Reference Manual co-packed with the BCP Print For .NET.

Moreover, as for the common errors with ORiN2, refer to "2.2.7 Error code" in the ORiN2 Programming Guide.

3. Command Reference

3.1. CaoController class

Table 3-1: CaoController class command list

Command	Function	Page
Label printer operation		
Issue	Sends the label issue command to a label printer.	14
Feed	Sends the feed command to a label printer.	15
GetMessage	Obtains the character string defined for the code.	15
GetStatus	Sends the printer status get command to a label printer.	16
SendCommand	Sends the command character string to a label printer.	16
Reset	Sends the command to clear the send buffer and reset the printer to its initial state to a label printer.	17

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

The label issue command is sent to a label printer. To use this command, specification of the label format file is required. Specify LfmFilePath by the connection option of AddController. Alternatively, specify the label format file by @LFM_FILE_PATH of the system variable before the command is executed.

Format Issue(<ISSUE_COUNT>,［<CUT_INTERVAL>])

ISSUE_COUNT : [in] Mandatory. Specifies the number of labels issued.
1 to 9999 (VT_I4)

CUT_INTERVAL [in] Arbitrarily. Specifies the cut interval. 0 to 100 (Default: 0) (VT_I4)
When "0" is set, no cutting is performed.
Specify "0" for B-EP2G and B-EP4G because they do not have the cut function. The print method (whether to cut) in the label format setting is ignored.

Return value : [out] If the following process is successful, true is returned. If not, false is returned. (VT_BOOL)

Communication setting	Description
COM/Socket (Return upon the Completion of Sending)	Creates the command data in the send buffer.
COM/Socket (Return upon the Completion of Printing)	Completes the label issue.

Example

```
var param = new int[] { 1,0 };

// If the second argument is omitted, "param" may be int.
// int param = 1;
var result = this.ctrl.Execute("Issue", param);
```

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

The feed command is sent to a label printer.

Format Feed([<CUTTING>])

CUTTING : [in] Arbitrarily. Specifies whether to cut after label feeding.

0 : Cut (Default)

1 : Not cut

(VT_I4)

Return value : [out] If the following process is successful, true is returned. If not, false is returned. (VT_BOOL)

Communication setting	Description
COM/Socket (Return upon the Completion of Sending)	Creates the command data in the send buffer.
COM/Socket (Return upon the Completion of Printing)	Completes the label feed. The label feed completion status is received from a label printer.

Example

```
var result = this.ctrl.Execute("Feed", 1);
```

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

The command to obtain the character string defined for an error or status code is sent to a label printer.

Format GetMessage(<PRINT_STATUS>)

PRINT_STATUS : [in] Mandatory. Specifies the status character string received from a label printer. (VT_BSTR)

Return value : [out] Returns the character string corresponding to the status. If applicable information cannot be found, Null is returned. (VT_BSTR)

Example

```
var result = this.ctrl.Execute("GetMessage ", "testStatusBSTR");
```

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

The printer status get command is sent to a label printer.

Format GetStatus()

Return value : [out] Returns the status character string received from a label printer. (VT_BSTR)

Communication setting	Description
COM/Socket (Return upon the Completion of Sending)	Always returns Null. Obtain the printer status in the OnMessage event.
COM/Socket (Return upon the Completion of Printing)	Returns the status character string received from a label printer.

Example

```
var result = this.ctrl.Execute("GetStatus", null);
```

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

The command character string is sent to a label printer. If the print command is executed by using SendCommand, it is necessary to call the command data registered as the PC save in advance. Specify LoadPCIndex by the connection option of AddController. Alternatively, specify the PC save ID number by @ LOAD_PC_INDEX of the system variable before the command is executed. Moreover, it is necessary to set the PC save ID number again by @ LOAD_PC_INDEX after the setting of the label printer is cleared.

Format SendCommand(<COMMAND>)

COMMAND : [in] Mandatory. Specifies the command character string to send (excluding the command control code). (VT_BSTR)

Return value : [out] If the following process is successful, true is returned. If not, false is returned. (VT_BOOL)

Communication setting	Description
COM/Socket (Return upon the Completion of Sending)	Creates the command data in the send buffer.
COM/Socket (Return upon the Completion of Printing)	Outputs the command data to the port. Checking is performed until the number of yet-to-be output blocks reaches 0.

Example

```
var result = this.ctrl.Execute("SendCommand", "J1;C");
```

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

The Reset command is sent to a label printer to clear the send buffer and reset the printer to its initial state.

Format Reset(< RESET_TYPE>, [<WAIT>])

RESET_TYPE : [in] Arbitrarily. Specifies how to reset the printer to its initial state. (Default: 0) (VT_I4)

0 : Clears the send buffer only.

1 : Clears the send buffer and resets the printer to its initial state.

2 : Clears the receive buffer and the received status buffer.

WAIT : [in] Arbitrarily. The period of time to wait for the label printer to be reset to its initial state is specified between 1 to 100 (seconds). (Default: 3) (VT_I4)

The guide period of time varies depending on the printer model.
For B-EX4T1-T, the guide period time is approx. 3 seconds.
However, if "0" (Clears the send buffer only) is set to RESET_TYPE or the communication setting is other than COM (Return upon the Completion of Sending) or Socket (Return upon the Completion of Printing), wait will not take place, and therefore, this setting will be ignored.

Return value : [out] If the following process is successful, true is returned. If not, false is returned. (VT_BOOL)

Communication setting	Description
COM/Socket (Return upon the Completion of Sending)	RESET_TYPE = 0: Clears the send buffer. RESET_TYPE = 1: Creates the command data in the send buffer.
COM/Socket (Return upon the Completion of Printing)	RESET_TYPE = 0: Clears the send buffer. RESET_TYPE = 1: Resets the label printer to its initial state.

Example

```
var param = new int[] { 1,3 };

// If the second argument is omitted, "param" may be int.
// int param = 1;
var result = this.ctrl.Execute("Reset", param);
```

4. Sample Program

The sample of C# to perform printing by using a label format and obtain the print status is described. The settings to be used for the sample program are as below. Arrange each item depending on the actual use environment.

IP address	192.168.10.10	
Port	8000	
Label format file path	C:¥tpcl_test¥Ws_PD-A_TPCL.lfm	
BCP Print For .NET install folder	C:¥Program Files (x86)¥TOSHIBA TEC¥BCP Print For .NET 2.0 * When brackets "(" and ")" are included in the path, enclose the entire path with square brackets "[" and "]" in the option character string.	
Controller	Connection provider of the controller	CaoProv.TEC.TPCL
	Controller name	TPCL_Sample
Item	Result number returned from the printer	@ RESULT_CODE

List 4-1

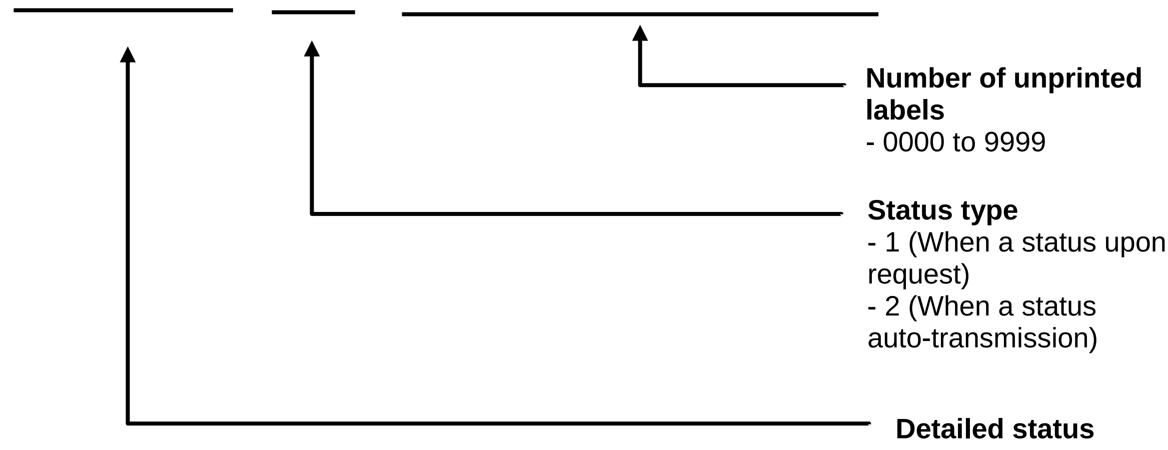
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading;
using ORiN2.ManagedCAO;

namespace Sample
{
    class Program
    {
        static void Main(string[] args)
        {
            CCaoEngine eng = null;

            try
            {
                // Create Cao objects
                eng = new CCaoEngine();
                CCaoWorkspaces wss = eng.Workspaces;
                CCaoWorkspace ws = wss[0];

                var connOpt = @"Conn=TCP:192.168.10.10:8000,
                        LfmFilePath=C:¥tpcl_test¥Ws_PD-A_TPCL.lfm,
                        SystemPath=[C:¥Program Files (x86)¥TOSHIBA TEC¥BCP Print For .NET 2.0],
                            UsePrinter=21,Timeout=5";
                // Connect to printer
                CCaoController ctrl = ws.AddController("TPCL_Sample",
                                                       "CaoProv.TEC.TPCL",
                                                       null,
                                                       connOpt);
                // Set event handler for message reception
                ctrl.OnMessage += new OnMesssageEventHandler(OnMessageEvent);

                var param = new int[] { 1, 0 };

                // If the second argument is omitted,"param" may be int.
                // int param = 1;
                var result = ctrl.Execute("Issue", param);


                // Get result code
                CCaoVariable caoResultCode = ctrl.AddVariable("@RESULT_CODE", null);
                var resultCode = caoResultCode.Value;
                var code = (resultCode != null) ? Convert.ToString(resultCode) : string.Empty;
                Console.WriteLine("@RESULT_CODE:" + code);

                // Wait 10 seconds to accept OnMessage event
                Thread.Sleep(10000);
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            finally
            {
                // Disconnect
                if (eng != null)
                {
                    eng.Dispose();
                    eng = null;
                }
            }
        }

        static void OnMessageEvent(object sender, OnMessageEventArgs e)
        {
            var message = e.Message;
            var number = message.Number;
            var value = (int)message.Value;
            Console.WriteLine(string.Format("Message Number:{0} Value:{1} Description:{2}",
              message.Number, value, message.Description));
        }
    }
}
```

5. Appendix

5.1. Printer Status

The following shows the configuration of the status character string returned from the label printer. As for "Number of unprinted labels", the number of labels that are yet to be printed is set.

Status			Number of unprinted labels			
X	X	X	X	X	X	X



Number of unprinted labels
- 0000 to 9999

Status type
- 1 (When a status upon request)
- 2 (When a status auto-transmission)

Detailed status

The following table shows the detailed status codes for the B-EX series. As for the codes for other models, refer to the Reference Manual co-packed with the BCP Print For .NET.

Table 5-1: Detailed status for the B-EX series

Printer status	Detailed status	
	Status auto-transmission	When a status upon request
The print head was closed when the head was opened.	00	00
The print head was opened in the online mode.	01	01
The printer is in operation. (A command analysis in progress, rendering in progress, printing in progress feed in progress.)	-	02
The printer is in a pause state.	-	04
Waiting for the printed label to be removed.	-	05
A command error was found during a command analysis.	06	06
Any of the following errors occurred during communication via RS-232C: parity error, overrun error or framing error.	07	07
Label jammed while being fed.	11	11

An error occurred with the cutter.	12	12
Label ended.	13	13
Feeding or printing was attempted with the print head opened (except for the FEED key).	15	15
A broken element was found in the thermal print head.	17	17
The thermal print head was overheated.	18	18
An abnormal condition occurred in the sensor for determining the torque for the ribbon motor.	21	21
An overflow error has occurred in the rewinder.	22	22
Ribbon near end state (Online) *1	-	27
Ribbon near end state (In a pause state) *2	-	28
Ribbon near end state (Operating) *1	-	29
Label printing ended successfully.	40	-
Label feed ended successfully.	41	-
Broken element check ended successfully.	00	-
An error occurred in writing to the registration memory.	50	50
An erase error occurred in formatting the registration memory.	51	51
Entry cannot be allowed due to the lack of the available free space of the registration memory.	54	54
User-defined character, PC command entry mode	-	55
A void pattern printed.	60	-
RFID write error	61	61
Communication with RFID module failed.	62	62
Countries which can be used an RFID module are not set up in it.	65	65
The peel-off cover opened in the online mode.	83	83

*1) The Issue command does not return the status.
To obtain this status, execute the GetStatus command in advance.

*2) The Issue command does not return the status in the Completion of Sending mode (when "1" is set for IssueMode in the connection option of AddController).
To obtain this status, call the GetStatus command in advance.