EPSON
EXCEED YOUR VISION

Advanced Printer Driver for TM-T88IV ReStick Ver.4

Devmode API / PRINTERINFO Manual

M00015001
Rev.B

Cautions

- No part of this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Seiko Epson Corporation.
- The contents of this document are subject to change without notice. Please contact us for the latest information.
- While every precaution has taken in the preparation of this document, Seiko Epson Corporation assumes no responsibility for errors or omissions.
- Neither is any liability assumed for damages resulting from the use of the information contained herein.
- Neither Seiko Epson Corporation nor its affiliates shall be liable to the purchaser of this product or third parties for damages, losses, costs, or expenses incurred by the purchaser or third parties as a result of: accident, misuse, or abuse of this product or unauthorized modifications, repairs, or alterations to this product, or (excluding the U.S.) failure to strictly comply with Seiko Epson Corporation's operating and maintenance instructions.
- Seiko Epson Corporation shall not be liable against any damages or problems arising from the use of any options or any consumable products other than those designated as Original EPSON Products or EPSON Approved Products by Seiko Epson Corporation.

Trademarks

EPSON® and ESC/POS® are registered trademarks of Seiko Epson Corporation in the U.S. and other countries.

MS-DOS®, Microsoft®, Win32®, Windows®, Windows Vista®, Visual Studio®, Visual Basic®, Visual C++®, and Visual C#® are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.

ESC/POS® Command System

EPSON leads the printer business by ESC/POS, the original POS printer command system. ESC/POS implements a lot of original commands, some of them have patents, and makes it possible to build a high-expandability and versatile POS system. It also has a compatibility for all the EPSON POS printers (excluding TM-C100) and display devices and a flexibility of unique control system so that it is easy to upgrade the system in the future easily. The function and user-friendliness is valued all over the world.

For Safety

Key to Symbols

The symbols in this manual are identified by their level of importance, as defined below. Read the following carefully before handling the product.

CAUTION	Provides information that must be observed to avoid damage to your equipment or a malfunction.
NOTE	Provides important information and useful tips.

Restriction of Use

When this product is used for applications requiring high reliability/safety such as transportation devices related to aviation, rail, marine, automotive etc.; disaster prevention devices; various safety devices etc; or functional/precision devices etc, you should use this product only after giving consideration to including fail-safes and redundancies into your design to maintain safety and total system reliability. Because this product was not intended for use in applications requiring extremely high reliability/safety such as aerospace equipment, main communication equipment, nuclear power control equipment, or medical equipment related to direct medical care etc, please make your own judgment on this product's suitability after a full evaluation.

About this Manual

Aim of the Manual

This manual is aimed to provide all the necessary information for development engineers to develop, design, and install POS system, or to develop and design printer applications.

Manual Content

The manual is made up of the following sections:

Contents

Overview

This manual describes Devmode API and PrinterInfo.

Devmode API for APD provides a function to acquire/set up device-unique Devmode from user's application. Parameters and setting range of command IDs (functions of Devmode API) are acquired from the device, and the parameter is set to the device by the application.

PRINTERINFO is one of Windows structures. In addition to Status API provided by EPSON to acquire TM printer status by the user application, Windows functions can be used for that purpose. For PRINTERINFO information, refer to "Acquiring TM Printer Status Using PRINTERINFO_2" on page 27. For details, refer to the Microsoft website.

Contents of This Manual

Install manual

Descriptions of the procedures from installing the APD to performing test print, adding printer drivers, changing port, and the silent installation that is an automated APD installation.

TM Printer Manual

Descriptions of how to use the APD and its functions.
Descriptions of the specifications of TM-T88IV ReStick.

Status API Manual

Descriptions of how to acquire the status of the TM printer from the user application by using the Status API.

Devmode API / PRINTERINFO Manual

This manual. Descriptions of how to set some functions of printer from the user application by using Devmode API. Descriptions of PrinterInfo, which is part of Windows functions.

Devmode API Overview

This section explains the functions of Devmode API and applicable devices.

Devmode API Functions

Devmode API provides the following functions (command IDs). (Available functions differ depending on the printer driver. See "Functions for TM-T88IV ReStick" on page 17.)

- Changing paper size
- Changing print direction
- Changing margin setting
- Changing input unit
- Changing paper feeding mode
- Changing print speed
- Setting back-feed after cutting
- TrueType font substitution

Provided File

EPSON provides the following files:

Type	File Name	Description
Header File for API Win32	APDDM_API.H	In this file, API is defined during creation of Win32 32-bit application. <destination of the header file> C:\Program Files\EPSON \EPSON Advanced Printer Driver 4\Sample\US \Src\C++\SingleFunction\Program13\Program13
Devmode API module	EAPDM32.DLL	This executable file is called by the Win32 32-bit application in Windows\System32.
.NET Wrapper module	EAPDM32W.DLL	This executable file is called by the application in Windows\assembly.

Sample Program

Devmode API is used for "Changing paper feeding mode" in the sample program.

Operational Environment

OS

Conforming to the APD environment. Refer to "Install Manual".

On Windows XP, no .NET modules are installed without .NET Framework 2.0 during APD installation. If .NET Framework 2.0 was installed later, install Devmode API .NET using custom installation of APD.

CAUTION	In Terminal Service / Citrix XenApp environment, the status of TM printers cannot be acquired using PRINTER_INFO_2.

.NET Framework Version

Conforming to the APD environment. Refer to "Install Manual".

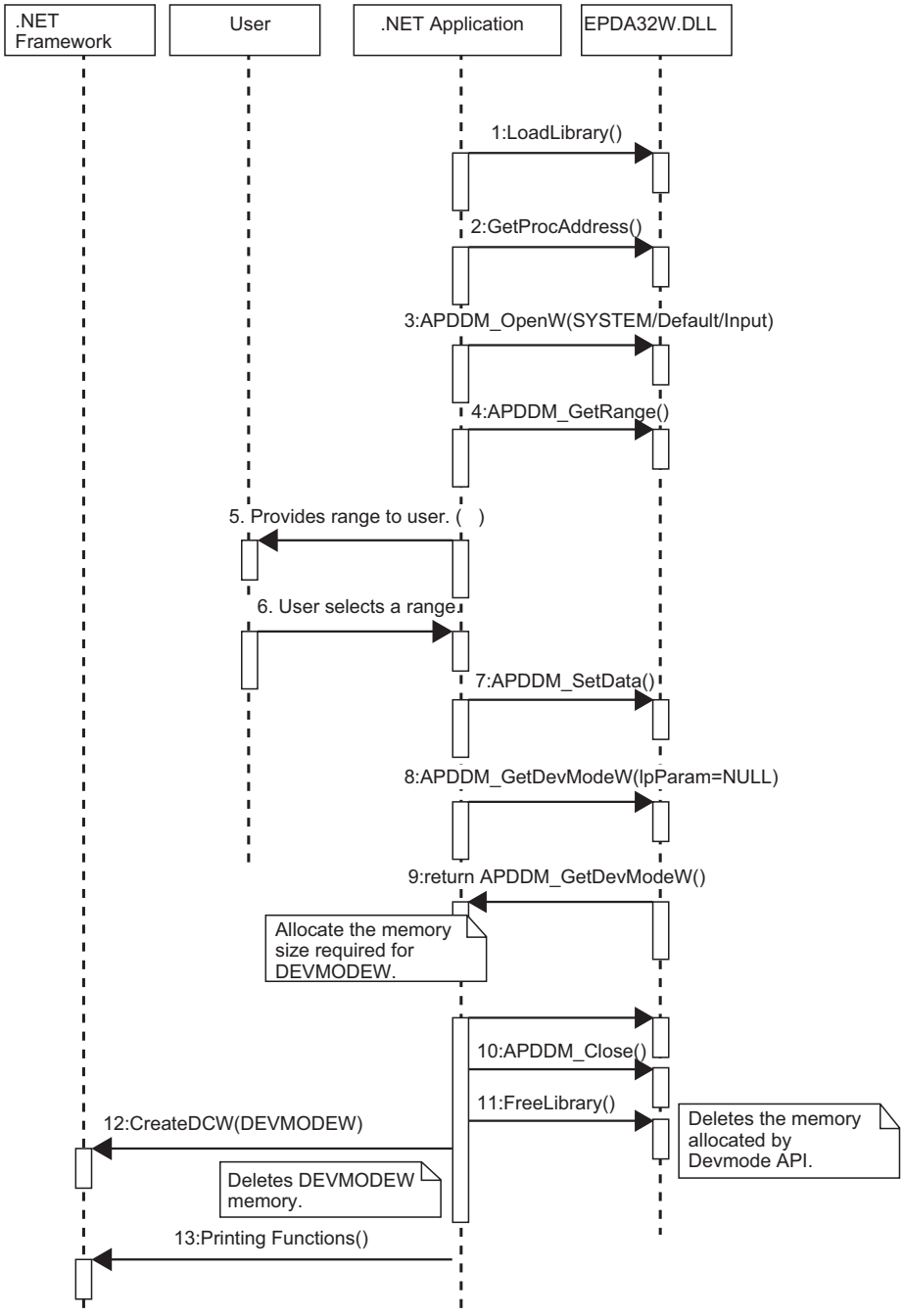
If you use Devmode API .NET Wrapper in Windows XP, install .NET Framework 2.0 or later before installing APD.

How to Use Devmode API

This chapter describes how to use Devmode API.

Sequence

Recommended sequence for Win32 is as shown below.



Explanation

1. LoadLibrary()

2. GetProcAddress()

3. APDDM_OpenW(SYSTEM/Default/Input)
   Opens Devmode API. When using Devmode API of the device, use "System Devmode" to open.

4. APDDM_GetRange()
   Specifies a command ID and acquires the setting range that can be set for the ID from the device.

5. Provides the acquired command ID range to the user.

6. The user makes selection from the range.

7. APDDM_SetData()
   Sets a value to the device.

8. APDDM_GetDevModeW()
   Acquires the set value of the device. It is recommended that the application acquire/store this value.

9. return APDDM_GetDevModeW()
   Acquires the memory size required for DEVMODEW. The application allocates memory.

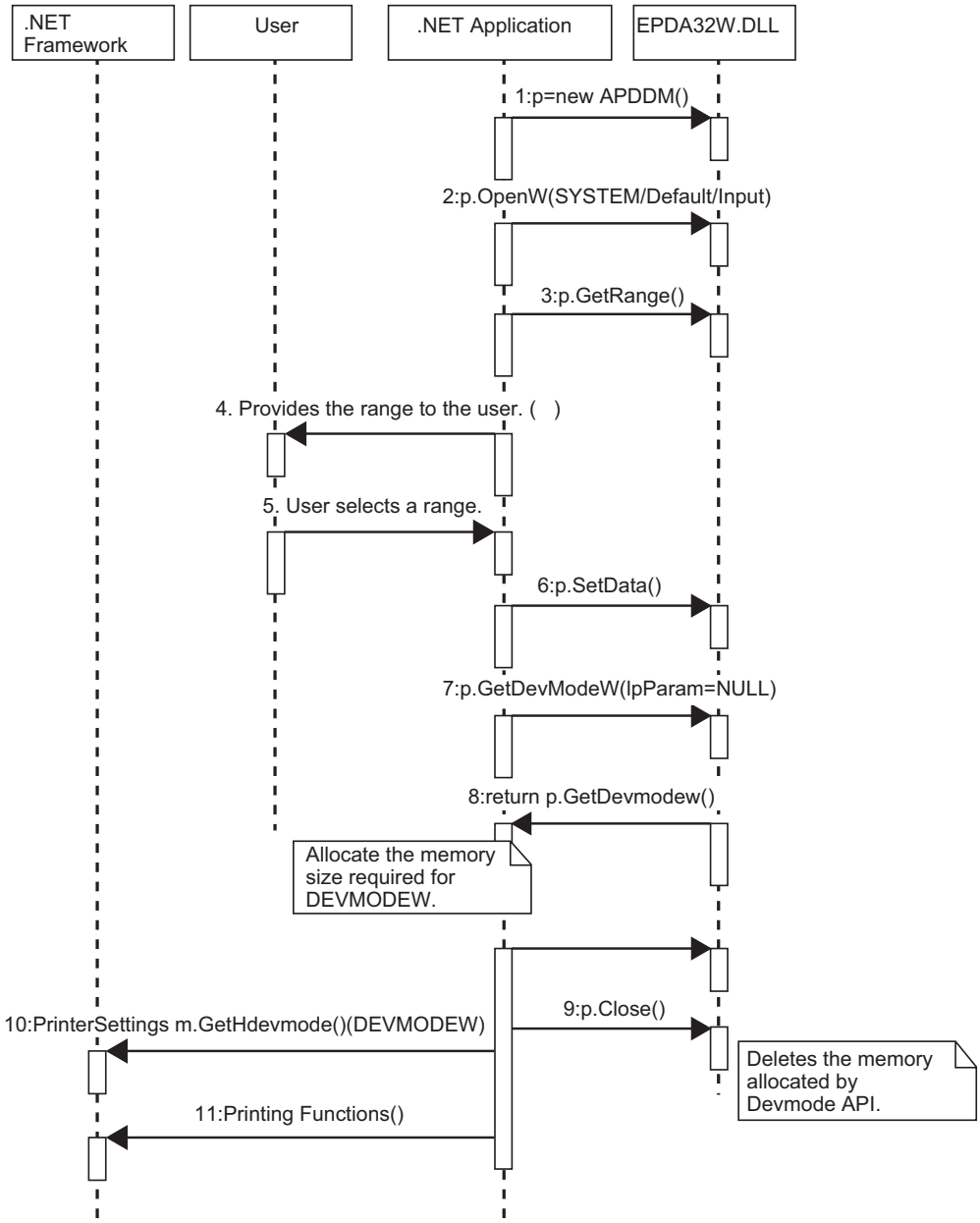
10. APDDM_Close()
    Closes Devmode API.

11. FreeLibrary()
    Deletes the memory allocated by Devmode API.

12. CreateDCW(DEVMODEW)
    Deletes DEVMODEW memory.

13. Printing Functions()
    Printing with the set contents is available.

Recommended sequence for .NET is as shown below.



Devmode API Development Information

This section describes the information required for application development.

Character Code

Unicode is used for all strings.

Unicode is used for all strings. It is required to set UNI_CODE and _UNICODE in the compiler macro definition. Use Wide-character for all Windows APIs.

Devmode Type

The following three Devmode types are available when opened:

- System DEVMODE
  DEVMODE set in the printer driver is used as the input parameter. In this case, set NULL because DEVMODEW address is not used for input.
- Default DEVMODE
  The default DEVMODE for the printer driver is used as the input parameter. In this case, set NULL because DEVMODEW address is not used for input.
- Specified DEVMODE
  DEVMODE acquired independently by the application is used as the input parameter.
  Can be acquired from DocumentPropertySheet and Windows API.
  The DEVMODEW area that is allocated by the application needs to be released by the application.

Memory Allocation

The Devmode API module does not need to release the memory area that returns as a parameter. FreeLibrary () of the Devmode API module releases the memory area. That is why you should copy the memory area in advance when the data needs to be stored.

DEVMODEW acquired by GetDevmodeW(.NET) or APDDM_GetDevModeW(Win32) is used in a later printing process and is not released by the Devmode API module. So, release it by the application.

Multi-process/Multi-thread/Multi-user

Because the Devmode API can handle the instances of DEVMODEW at the same time, it supports multi-process/multi-thread/multi-user.
For applications, .NET version can distinguish the instances with the object, and Win32 version can distinguish the instances with the printer device information handle. The Devmode API module does not need to release the memory area that returns as a parameter.

Error Code

The error codes returned by each API are identical to the Windows error codes (defined in WinError.h).

Reference for Win32

This chapter describes the functions of TM-T88IV ReStick, API reference and command ID.

Functions for TM-T88IV ReStick

- Changing paper size ("APD_DM_PAPER_SIZE" on page 21)
- Changing print direction ("APD_DM_ORIENTATION" on page 21)
- Changing margin setting ("APD_DM_PAPERSAVE" on page 21)
- Changing input unit ("APD_DM_INPUT_UNIT" on page 21)
- Changing paper feeding mode ("APD_DM_PAPER_BIN" on page 22)
- Changing print speed ("APD_DM_SPEED_DEVICE" on page 22)
- TrueType font substitution ("APD_DM_REPLACEFONT" on page 22)

APDDM_OpenW

Opens the specified printer device and enables device information acquisition/setting. lphAPDDM for the specified DEVMODE, it is required to allocate and delete memory by the application.

Syntax

DWORD ***APDDM_OpenW*** (LPCWSTR lpwDevName,
DWORD dwType, LPDEVMODEW lpDM,
LPHANDLE lphAPDDM)

Argument

lpwDevName :	Printer device name address. This is LPCWSTR type.
dwType:	DEVMODE type when opened. Select among the following: System DEVMODE/Default DEVMODE/Specified DEVMODE. This is DWORD type.
lpDM :	DEVMODEW address. This is LPDEVMODEW type.
lphAPDDM :	Printer device information handle address. This is LPHANDLE type.

APDDM_GetRange

Acquires the setting range of the specified command ID from the device.

Syntax

DWORD ***APDDM_GetRange*** (HANDLE hAPDDM,
DWORD nCommad,
LPVOID* lpParam,
LPDWORD pdwCount,
LPDWORD pdwSize)

Argument

hAPDDM :	Printer device information handle. This is Handle.
n Commad	Command ID (function) to acquire the execution level. This is DWORD type.
lpParam:	The setting range storage parameter. This is LPVOID* type.
pdwCount:	The number of elements. This is LPDWORD type.
pdwSize:	The parameter size. The number of parameters divided by the number of elements indicates the number of arrays of Param. This is LPDWORD type.

APDDM_GetData

Acquires the data of the specified command ID from the device.

Syntax

DWORD ***APDDM_GetData*** (HANDLE hAPDDM,
DWORD nCommad,
LPVOID* lpParam, LPDWORD pdwSize)

Argument

hAPDDM :	Printer device information handle. This is Handle.
n Commad	Command ID (function) to acquire the execution level. This is DWORD type.
lpParam:	Obtained data storage parameter. This is LPVOID* type.
pdwSize:	Obtained data size. This is LPDWORD type.

APDDM_SetData

Configures the specified data to the device.

Syntax

DWORD ***APDDM_SetData*** (HANDLE hAPDDM, DWORD nCommad,
LPVOID lpParam, LPDWORD pdwSize)

Argument

hAPDDM :	Printer device information handle. This is Handle.
n Commad	Command ID (function) to configure the data. This is DWORD type.
lpParam:	Configured data storage parameter. This is LPVOID type.
pdwSize:	Configured data size. This is LPDWORD type.

APDDM_GetDevModeW

Acquires DEVMODE information, to which the printer device information is applied, to the specified area. For lpDM of the obtained DEVMODE, it is required to allocate and delete memory by the application.

Syntax

DWORD ***APDDM_GetDevModeW*** (HANDLE hAPDDM,
LPDEVMODE lpDM,
LPDWORD pdwSize)

Argument

hAPDDM :	Printer device information handle. This is Handle.
lpDM	DEVMODE address. This is LPDEVMODE type.
pdwSize:	Obtained DEVMODEW size. This is LPDWORD type.

APDDM_Close

Closes the specified printer device.

Syntax

DWORD ***APDDM_Close*** (HANDLE hAPDDM)

Argument

hAPDDM : Printer device information handle. This is Handle.

Command ID

APD_DM_PAPER_SIZE

Acquires or changes the paper size. User-defined papers are not supported.

For paper size name acquisition, the following two methods are available:

- Enumeration type:Enumerates the paper size names. The paper size names correspond to the array of paper sizes.
  The command ID can be specified with the following macro:
  GET_ENUM_STRING_COMMAND_ID(COMMAND)
- Single type:Acquires a paper name by specifying a single paper size.
  The command ID can be specified with the following macro:
  GET_STRING_COMMAND_ID(COMMAND, ID)

APD_DM_ORIENTATION

Acquires or changes the print direction. Select the print direction among the following: 0 degree, 90 degree, 180 degree, and 270 degree.

To change the direction to any option other than 0 degree, it is required to set "Yes" to "Print all documents as image" in formatting.

APD_DM_PAPERSAVE

Acquires or changes the margin reduction setting. Select the setting among the following: No margin reduction/Upper margin reduction/Bottom margin reduction/Upper and bottom margin reduction. Available options depend on the printer model.

APD_DM_INPUT_UNIT

Acquires or changes the driver input unit. Specify either mm or inch.

APD_DM_PAPER_BIN

Acquires or changes the paper feeder. Available options depend on the printer model.
For acquisition of paper feeder name, enumeration-type and single-type macros are available as with changing paper size.

APD_DM_SPEED_DEVICE

Acquires or changes the print speed. Available options depend on the printer model.

APD_DM_REPLACEFONT

Specifies whether or not to substitute TrueType fonts to device fonts. It is required to assign TrueType fonts to device fonts manually in advance.

Reference for .NET

This chapter describes the API reference in .NET environment. For the functions of each printer and command ID, refer to "Reference for Win32" on page 17.

OpenW

Calls APDDM_OpenW for Win32.

Opens the specified printer device and enables device information acquisition/setting.

The printer information handle retained by the wrapper.

To acquire the DEVMODE structure pointer, acquire the handle to the DEVMODE structure pointer from the application first and then acquire the pointer using GlobalLock which is an unmanaged method. After use, call GlobalUnLock,GlobalFree which are unmanaged methods, and release the pointer by the application.

Syntax

OpenW (String strDevName, UInt32 un32Type, IntPtr npDM)

Argument

strDevName :	Printer device name string. This is String type.
un32Type	DEVMODE type when opened. Select among the following: System DEVMODE/Default DEVMODE/Specified DEVMODE. This is UInt32 type.
npDM :	DEVMODE structure pointer. This is IntPtr type.

GetRange

Calls APDDM_GetRange for Win32.

Acquires the setting range of the specified command ID from the device.

Syntax

GetRange (UInt32 un32Commad, out IntPtr npParam,
out UInt32 un32Count, out UInt32 un32Size)

Argument

un32Commad	Command ID (function) to acquire the execution level. This is UInt32 type.
npParam:	The setting range storage parameter. Contents differ depending on the command ID. This is IntPtr type.
un32Count:	The number of elements. This is UInt32 type.
un32Size	The parameter size. The number of parameters divided by the number of elements indicates the number of arrays of Param. This is UInt32 type.

GetData

Calls APDDM_GetData for Win32.

Acquires the data of the specified command ID from the device.

Syntax

GetData (UInt32 un32Commad, out IntPtr npParam,
out UInt32 un32Size)

Argument

un32Commad	Command ID (function) to acquire the execution level. This is UInt32 type.
npParam:	Obtained data storage parameter. This is IntPtr type.
un32Size:	Data size of the obtained parameter. This is UInt32 type.

SetData

Calls APDDM_SetData for Win32.

Configures the specified data to the device.

Syntax

SetData (UInt32 un32Commad, IntPtr npParam, UInt32 un32Size)

Argument

un32Commad	Command ID (function) to acquire the execution level. This is UInt32 type.
npParam:	Configured data storage parameter. This is IntPtr type.
un32Size:	Specifies the parameter data size. This is UInt32 type.

GetDevModeW

Calls APDDM_GetDevModeW for Win32.

Acquires DEVMODE information, to which the printer device information is applied, to the specified area.

To acquire the DEVMODE structure pointer, acquire the handle to the DEVMODE structure pointer from the application first and then acquire the pointer using GlobalLock which is an unmanaged method. After use, call GlobalUnLock,GlobalFree which are unmanaged methods, and release the pointer by the application.

Syntax

GetDevModeW (IntPtr npDM, out UInt32 un32Size)

Argument

npDM	DEVMODEW structure pointer. This is IntPtr type.
un32Size	DEVMODEW structure size. This is UInt32 type.

Close

Calls APDDM_Close for Win32.

Closes the specified printer device.

Syntax

Close ()

PRINTERINFO

Acquiring TM Printer Status Using PRINTERINFO_2

Status API is provided for your applications to obtain the status of the TM printer. The status can be obtained also using Windows functions.

The status of PRINTER_INFO_2 structure can be obtained using GetPrinter function. For details, refer to Microsoft homepage.

> **CAUTION** In Terminal Service / Citrix XenApp environment, the status of TM printers cannot be acquired using PRINTER_INFO_2.

The statuses of PRINTER_INFO_2 structure and the TM printer are shown below.

PRINTERINFO_2 Status	TM Printer
PRINTER_STATUS_BUSY	Not supported.
PRINTER_STATUS_DOOR_OPEN	Printer cover is open.
PRINTER_STATUS_ERROR	Not supported.
PRINTER_STATUS_INITIALIZING	Initializing the TM printer.
PRINTER_STATUS_IO_ACTIVE	Not supported.
PRINTER_STATUS_MANUAL_FEED	Feeding paper with the Feed switch.
PRINTER_STATUS_NO_TONER	Ink cartridge is not installed/requires replacement.
PRINTER_STATUS_NOT_AVAILABLE	Unable to access the TM printer. (TM printer is not powered on/Cables are disconnected)
PRINTER_STATUS_OFFLINE	TM printer is off-line.
PRINTER_STATUS_OUT_OF_MEMORY	Not supported.
PRINTER_STATUS_OUTPUT_BIN_FULL	Not supported.
PRINTER_STATUS_PAGE_PUNT	Not supported.
PRINTER_STATUS_PAPER_JAM	Not supported.
PRINTER_STATUS_PAPER_OUT	No roll paper.
PRINTER_STATUS_PAPER_PROBLEM	Not supported.
PRINTER_STATUS_PAUSED	Not supported.
PRINTER_STATUS_PENDING_DELETION	Not supported.
PRINTER_STATUS_POWER_SAVE	Not supported.
PRINTER_STATUS_PRINTING	Printing.
PRINTER_STATUS_PROCESSING	Not supported.
PRINTER_STATUS_SERVER_UNKNOWN	Not supported.
PRINTER_STATUS_TONER_LOW	Low ink level.