

Thermal Printer

Status Monitor Library Windows®

API Software User's guide

Revision 1.1

## Table of Contents

1	Introduction .....	4
1.1	<b>Overview</b> .....	4
2	System Requirements .....	5
2.1	<b>Host Hardware</b> .....	5
2.2	<b>Operating System (OS)</b> .....	5
2.3	<b>Printer</b> .....	6
2.4	<b>Interface</b> .....	6
3	Configuration .....	7
3.1	<b>Module Configuration</b> .....	7
3.2	<b>Library function list</b> .....	7
3.3	<b>Conceptual Diagram</b> .....	8
3.4	<b>Performance flow</b> .....	9
3.4.1	<b>How to obtain the status by GetStatus function</b> .....	9
3.4.2	<b>How to obtain the status by GetStatusDirect function</b> .....	10
4	Function .....	11
4.1	<b>Library Function</b> .....	11
4.1.1	<b>OpenStatus function</b> .....	11
4.1.2	<b>CloseStatus function</b> .....	13
4.1.3	<b>GetStatus function</b> .....	14
4.1.4	<b>GetStatusDirect function</b> .....	15
4.1.5	<b>GetStatusCode function</b> .....	16
4.1.6	<b>IsPrintAvailable e function</b> .....	18
4.1.7	<b>IsPrinting function</b> .....	19
4.1.8	<b>IsOffline function</b> .....	20
4.1.9	<b>IsPaperNearEnd function</b> .....	21
4.1.10	<b>IsCoverOpen function</b> .....	22
4.1.11	<b>IsPaperOut function</b> .....	23
4.1.12	<b>IsHeadHot function</b> .....	24
4.1.13	<b>IsPaperLayoutError function</b> .....	25
4.1.14	<b>IsCutterJam function</b> .....	26
4.1.15	<b>IsProtocolError function</b> .....	27
4.1.16	<b>IsHardwaerError function</b> .....	28
4.1.17	<b>IsCommunicationError function</b> .....	29
5	Others .....	30

5.1	<b>Priority of statuses</b> .....	30
5.2	<b>Status at the printer power off</b> .....	31

---

# 1 Introduction

---

## 1.1 Overview

This document describes about the Status Monitor Library which provides API to obtain the printer status. on Microsoft Windows® OS.

This document describes detailed information about input/output parameters of API function and Return values.

The contents which are not mentioned target OS are for common OS.

---

## 2 System Requirements

---

### 2.1 Host Hardware

A personal computer running with one of the following operation systems. (OS).

### 2.2 Operating System (OS)

The Status Monitor Library can be used with the following operating systems.

#### 32bit OS

- Microsoft® Windows® 2000 Professional / Server / Advanced Server
- Microsoft® Windows® XP Professional Edition / Home Edition
- Microsoft® Windows Server® 2003 Standard Edition / Enterprise Edition
- Microsoft® Windows Vista® Ultimate Edition / Business Edition / Home Premium Edition / Home Basic Edition / Enterprise Edition
- Microsoft® Embedded for Point of Service(WEPOS)
- Microsoft® Windows Server® 2008 Standard Edition / Enterprise Edition
- Microsoft® Windows® 7 Ultimate Edition / Professional Edition / Home Premium Edition / Home Basic Edition / Enterprise Edition
- Windows® 8 / Windows® 8 Pro / Windows® 8 Enterprise
- Windows® 8.1 / Windows® 8.1 Pro / Windows® 8.1 Enterprise
- Windows® 10 Home / Windows® 10 Pro / Windows® 10 Enterprise

#### 64bit OS

- Microsoft® Windows® XP Professional x64 Edition
- Microsoft® Windows Server® 2003 x64 Edition
- Microsoft® Windows Vista® Ultimate Edition / Business Edition / Home Premium Edition / Home Basic Edition / Enterprise Edition
- Microsoft® Windows Server® 2008 Standard Edition / Enterprise Edition
- Microsoft® Windows Server® 2008 R2 Standard Edition / Enterprise Edition
- Microsoft® Windows® 7 Ultimate Edition / Professional Edition / Home Premium Edition / Home Basic Edition / Enterprise Edition
- Windows® 8 / Windows® 8 Pro / Windows® 8 Enterprise
- Microsoft® Windows Server® 2012 Datacenter / Microsoft® Windows Server® 2012 Standard / Microsoft® Windows Server® 2012 Essentials
- Windows® 8.1 / Windows® 8.1 Pro / Windows® 8.1 Enterprise
- Windows® 10 Home / Windows® 10 Pro / Windows® 10 Enterprise
- Microsoft® Windows Server® 2016 Datacenter / Microsoft® Windows Server® 2016 Standard / Microsoft® Windows Server® 2016 Essentials

## 2.3 Printer

The Status Monitor Library can be used with the following printers.

Printer	Printer Driver Model Name
FP-2200	FIT FP-2200 Raster
FP-2100	FIT FP-2100 Raster
FP-2000	FIT FP-2000 Raster
FP-1100	FIT FP-1100 Raster
FP-510	FIT FP-510 Raster
FP-460	FIT FP-460 Raster
FP-360	FIT FP-360 Raster
FP-32L(※1)	FIT FP-32L Raster

※1 : If a version of the printer driver is earlier than 2.41.01, the status monitor for FP-32L needs to be activated.

## 2.4 Interface

The Status Monitor Library can be used with the following interfaces.

- Serial port
- Parallel (Centronics) port
- USB port

---

## 3 Configuration

---

### 3.1 Module Configuration

The Status Monitor Library is comprised of the following files.

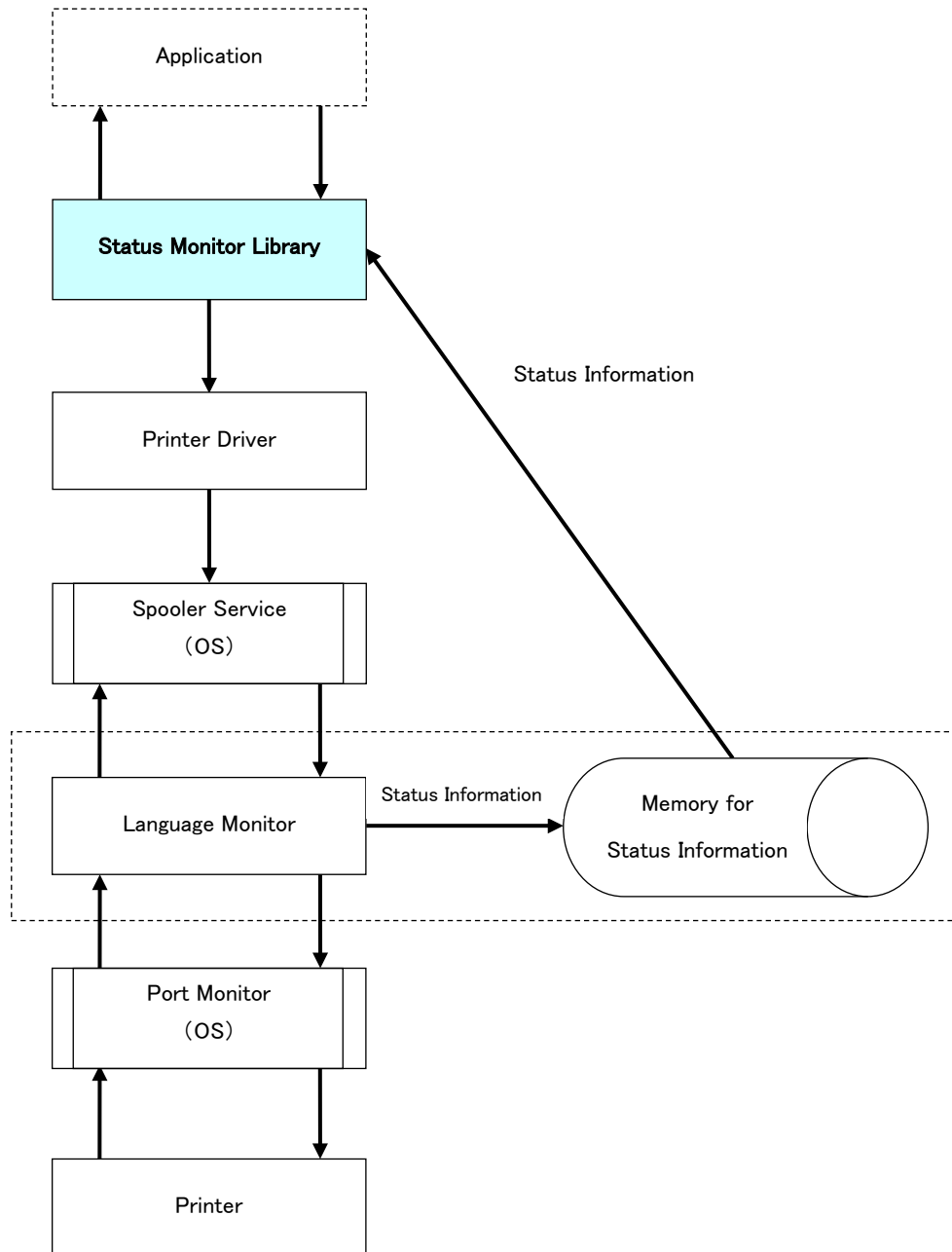
File name	Outline	Explanation
fjprnsts.dll	Main Module File	Main library file to notify a printer status
fjprnsts.lib	Library file for link	Lib file for implicit link
fjprnsts.h	Header file for link	Header file for link
How to use Status Monitor Library.docx	Guidance file	Guidance on how to use
Status Monitor Library API User's Guide.doc	API User's Guide	User's guide of the Status Monitor Library

### 3.2 Library function list

Library function list of the Status Monitor Library is as follows.

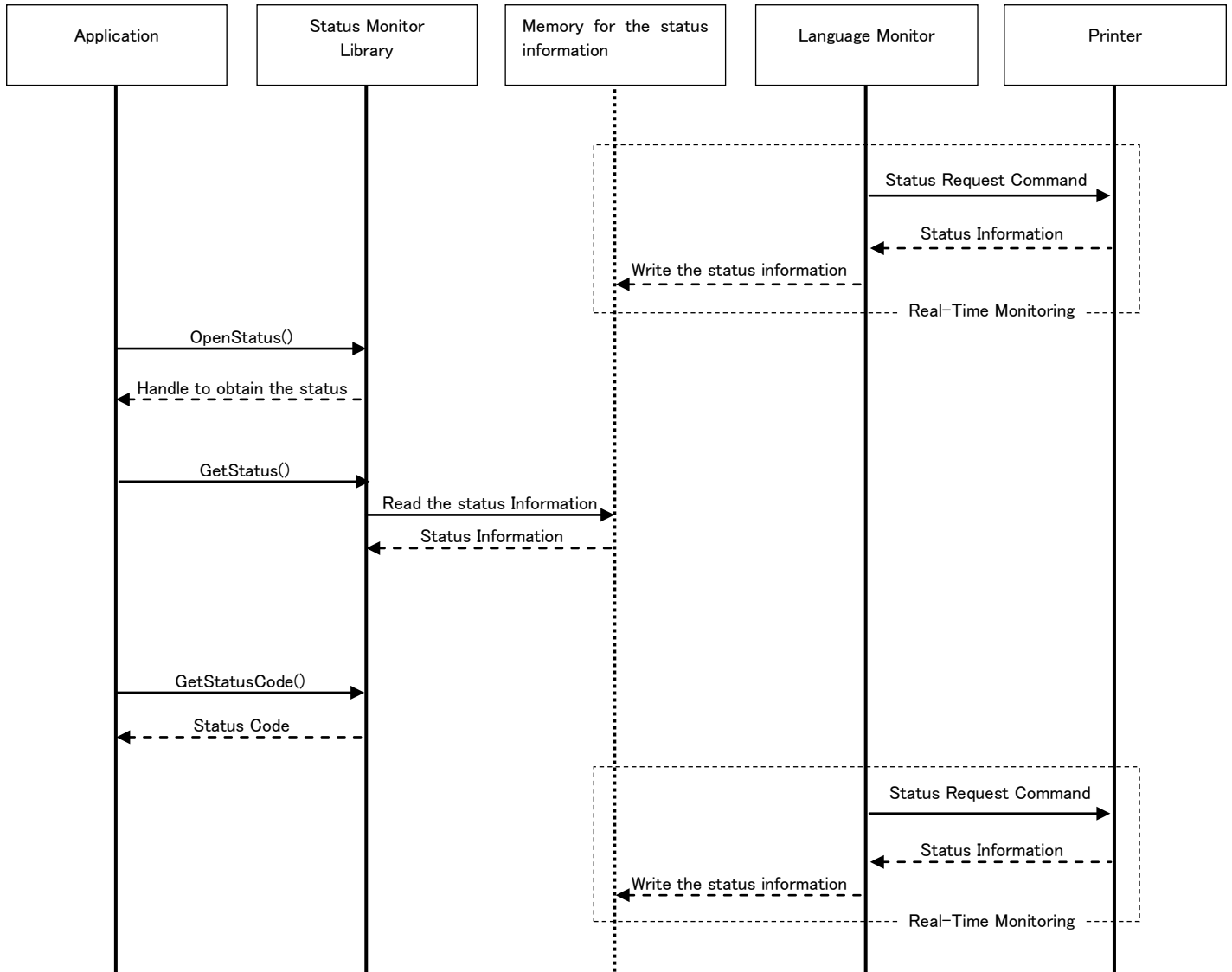
Function	Explanation
OpenStatus()	It enables to obtain a handle to receive the specified printer status.
CloseStatus()	It closes a handle to receive the printer status.
GetStatus()	It enables to obtain the status information stored in memory.
GetStatusDirect()	It enables to obtain the status information after updating the status information stored in memory.
GetStatusCode()	It enables to obtain the status code from status information.
IsPrintAvailable()	It confirms if the printer status is "Online".
IsPrinting()	It confirms if the printer status is "Printing".
IsOffline()	It confirms if the printer status is "Offline".
IsPaperNearEnd()	It confirms if the printer status is "Paper near end".
IsCoverOpen()	It confirms if the printer status is "Cover open".
IsPaperOut()	It confirms if the printer status is "Paper end".
IsHeadHot()	It confirms if the printer status is "Head hot".
IsPaperLayoutError()	It confirms if the printer status is "Paper layout error".
IsCutterJam()	It confirms if the printer status is "Cutter jam".
IsProtocolError()	It confirms if the printer status is "Protocol error".
IsHardwareError()	It confirms if the printer status is "Hardware error".
IsCommunicationError()	It confirms if the printer status is "Communication error".

### 3.3 Conceptual Diagram



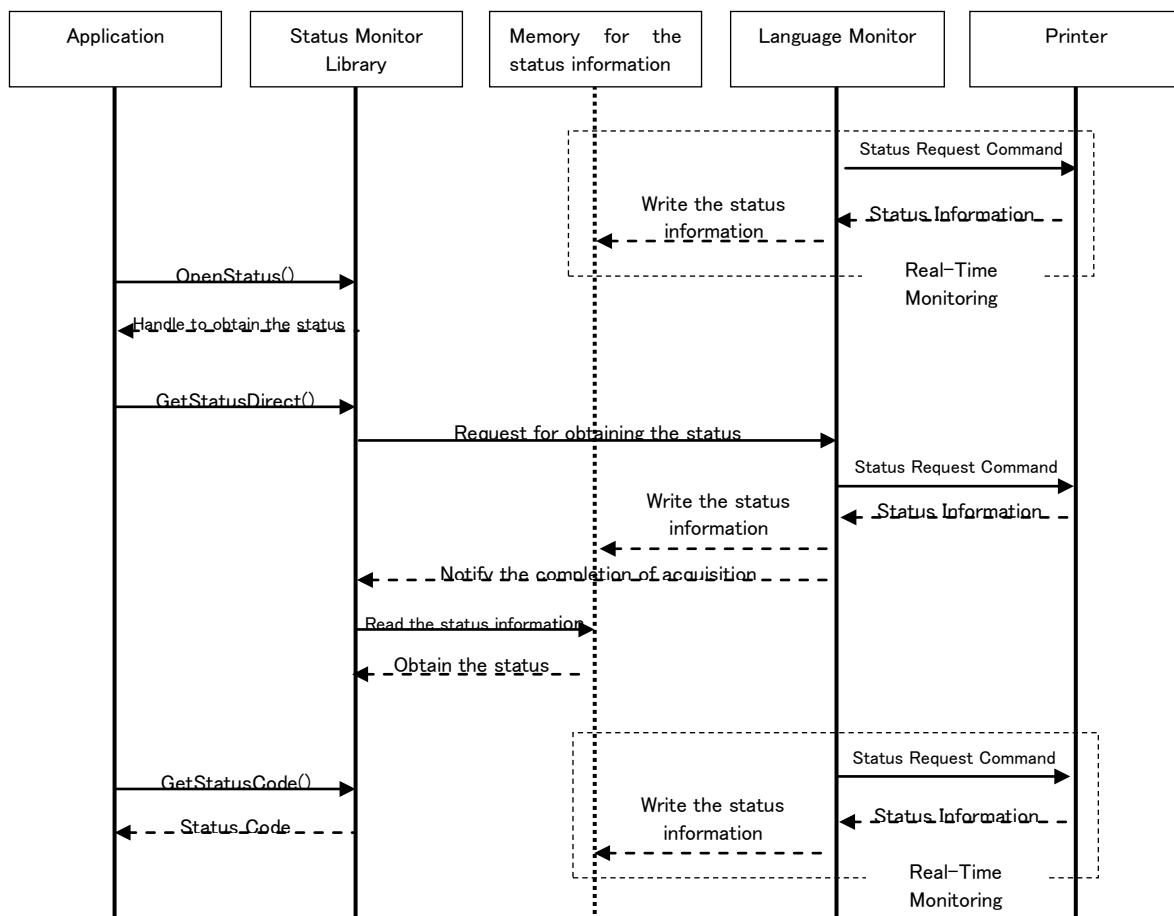
### 3.4 Performance flow

#### 3.4.1 How to obtain the status by GetStatus function



※ Interval of Real-Time Monitoring is about 500mm/s.

### 3.4.2 How to obtain the status by GetStatusDirect function



※ Interval of Real-Time Monitoring is about 500mm/s.

---

## 4 Function

---

### 4.1 Library Function

#### 4.1.1 OpenStatus function

It enables to obtain a handle to receive the specified printer status

```
HANDLE APIENTRY OpenStatus(  
    WCHAR* szPrinterName  
);
```

##### <Parameter>

*szPrinterName*

Specify a pointer of a character string ended by NULL which describes the printer name.

##### <Return Value>

Once the function succeeds, a handle is returned to obtain the printer status.

If the function fails, the return value is NULL. To get extended error information, use GetLastError function.

##### <Commentary>

After using a handle to obtain the printer status, call CloseStatus function and close this handle.

\* If you don't close the handle by CloseStatus function, a resource leak and a memory leak occur.

If you use Visual Studio 2003 ( or later versions )as the application development environment at caller side and you link implicitly, set up as follows.

Property of project => configuration property => C/C++ => Language

=> Treat wchar\_t as Built-in Type => YES

( If this setting is " NO ", the link error occurs.

If you use Visual Studio 6.0 as the application development environment at caller side and you link implicitly, use Replace API Function below.

If you don't use this function, the link error occurs.

```
HANDLE APIENTRY OpenStatusA(  
    const char* szPrinterName  
);
```

<Parameter>

*szPrinterName*

Specify a pointer of a character string ended by NULL which describes the printer name.

### 4.1.2 CloseStatus function

Close the handle for obtaining the printer status.

```
BOOL APIENTRY CloseStatus(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Return value>

Once the function succeeds, TRUE is returned.

If the function fails, the return value is FALSE. To get extended error information, use GetLastError function.

### 4.1.3 **GetStatus** function

It enables to obtain the status information stored in memory

```
BOOL APIENTRY GetStatus(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Return value>

Once the function succeeds, TRUE is returned.

If the function fails, the return value is FALSE. To get extended error information,  
use GetLastError function

<Commentary>

Because you obtain the printer status information stored in memory, the process by this function is  
returned quickly.

However, there is no guarantee that the status in the current memory is the latest one.

#### 4.1.4 **GetStatusDirect** function

It enables to obtain the status information after updating the status information stored in memory.

```
BOOL APIENTRY GetStatusDirect(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Return Value>

Once the function succeeds, TRUE is returned.

If the function fails, the return value is FALSE. To get extended error information, use GetLastError function

If “Enable bidirectional support” in “Ports “ tab in the printer properties is unchecked or if “Print directly to the printer“ in “ Advanced “ tab is checked, this function fails.

<Commentary>

In order to obtain the current printer status, this function requests the notification of the status to the printer directly.

Until obtaining the printer status is completed (Time out : Approx. 90 sec ), the process by this function is not returned.

.This function sends a print job to the printer to request the status information.

Therefore, in case the printer status is “ On Hold “,a print job for requesting the status information remains in the spooler.

#### 4.1.5 GetStatusCode function

It enables to obtain the status code from the status information.

```
BOOL APIENTRY GetStatusCode(  
    HANDLE hStatus,  
    int* piStatusCode  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

*piStatusCode*

This specifies a pointer variable to obtain the printer status.

The values stored in the variables are any of the followings:

Value		Printer Status Indication
FJPRN_STATUS_PRINTAVAILABLE	(0)	Online
FJPRN_STATUS_PRINTING	(2)	Printing
FJPRN_STATUS_REQUEST	(100)	Obtaining information
FJPRN_STATUS_OFFLINE	(200)	Offline
FJPRN_STATUS_PAPERNEAREND	(202)	Paper near end
FJPRN_STATUS_COVEROPEN	(301)	Cover Open
FJPRN_STATUS_PAPEROUT	(302)	Paper end
FJPRN_STATUS_HEADHOT	(303)	Head hot
FJPRN_STATUS_PAPERLAYOUTERROR	(304)	Paper layout error
FJPRN_STATUS_CUTTERJAM	(305)	Cutter jam
FJPRN_STATUS_PROTOCOLERROR	(500)	Protocol error
FJPRN_STATUS_HARDWAREERROR	(700)	Hardware error
FJPRN_STATUS_COMMUNICATIONERROR	(1500)	Communication error

#### <Return Value>

Once obtaining the status code succeeds, TRUE is returned.

If obtaining the status code fails, the return value is FALSE.

To get extended error information, use GetLastError function.

#### <Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.6 IsPrintAvailable e function

It confirms if the printer status is “ Online ”.

```
BOOL APIENTRY IsPrintAvailable(  
    HANDLE hStatus  
);
```

##### <Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

##### <Return Value>

If the printer status is “Online ”, “ TRUE” is returned.

If the printer status is other than “Online ”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

##### <Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.7 IsPrinting function

It confirms if the printing status is “Online”.

```
BOOL APIENTRY IsPrinting(  
    HANDLE hStatus  
);
```

##### <Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

##### <Return Value>

If the printer status is “Online”, TRUE is returned.

If the printer status is other than “Online”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

##### <Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.8 IsOffline function

It confirms if the printer status is “Offline”.

```
BOOL APIENTRY IsOffline(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Parameter>

If the printer status is “Offline”, TRUE is returned.

If the printer status is other than “Offline”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

<Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.9 IsPaperNearEnd function

It confirms if the printer status is “ Paper near end “.

```
BOOL APIENTRY IsPaperNearEnd(  
    HANDLE hStatus  
);
```

##### <Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle to obtain the printer status.

##### <Return Value>

If the printer status is “ Paper near end “., “TRUE” is returned.

If the printer status is other than “ Paper near end “., the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

##### <Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.10 IsCoverOpen function

It confirms if the printer status is “Cover open”.

```
BOOL APIENTRY IsCoverOpen(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Return Value>

If the printer status is “Cover open”, TRUE is returned.

If the printer status is other than “Cover open”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

<Commentary>>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.11 IsPaperOut function

It confirms if the printer status is “Paper end”

```
BOOL APIENTRY IsPaperOut(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns

<Return Value>

If the printer status is “Paper end ”, TRUE is returned.

If the printer status is other than “ Online ”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS,  
the error occurs.

Check the return value of GetLastError() for details.

<Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the  
status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status  
information.

#### 4.1.12 IsHeadHot function

It confirms if the printer status is “Head hot”.

```
BOOL APIENTRY IsHeadHot(  
    HANDLE hStatus  
);
```

##### <Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

##### <Return Value>

If the printer status is “Head hot”, TRUE is returned.

If the printer status is other than “Head hot”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

##### <Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.13 IsPaperLayoutError function

It confirms if the printer status is “Paper layout error”.

```
BOOL APIENTRY IsPaperLayoutError(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Return Value>

If the printer status is “Paper layout error”, TRUE is returned.

If the printer status is other than “Paper layout error”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

<Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.14 IsCutterJam function

It confirms if the printer status is “Cutter jam”

```
BOOL WINAPI IsCutterJam (  
    HANDLE hStatus  
);
```

##### <Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

##### <Return Value>

If the printer status is “Cutter jam”, TRUE is returned.

If the printer status is other than “Cutter jam”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

##### <Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.15 IsProtocolError function

It confirms if the printer status is "Protocol error"

.

```
BOOL APIENTRY IsProtocolError(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Return Value>

If the printer status is "Protocol error", TRUE is returned.

If the printer status is other than "Protocol error", the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

<Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.16 IsHardwaerError function

It confirms if the printer status is "Hardware error".

```
BOOL APIENTRY IsHardwareError(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Return Value>

If the printer status is "Hardware error", TRUE is returned.

If the printer status is other than "Hardware error", the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

<Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

#### 4.1.17 IsCommunicationError function

It confirms if the printer status is “Communication error”.

```
BOOL WINAPI IsCommunicationError(  
    HANDLE hStatus  
);
```

<Parameter>

*hStatus*

This specifies the handle to obtain the printer status.

Specify the handle that OpenStatus function returns.

<Return Value>

If the printer status is “Communication error”, TRUE is returned.

If the printer status is other than “Communication error”, the return value is FALSE.

When FALSE is returned, the return value of GetLastError() is other than ERROR\_SUCCESS, the error occurs.

Check the return value of GetLastError() for details.

<Commentary>

Call this function after calling GetStatus function or GetStatusDirect function and obtaining the status information of the printer.

The operation is not guaranteed in case you call this function before obtaining the status information.

---

## 5 Others

---

### 5.1 Priority of statuses

When a higher priority status occurs, a lower priority status is not notified.

Priority	Status
Highest	Communication error
	Protocol error
	Hardware error
	Cover open
	Cutter jam
	Paper end
	Head hot
	Paper layout error
	Offline
	Paper near end
	Printing
Lowest	Online

## 5.2 Status at the printer power off

Even if you obtain the status by GetStatus function after turning the printer power off when printing is not in process (Idling), the status information before the printer power off is returned.

(Except IsCommunicationError function when USB interface is used.)

The return value for “Communication error” when the printer power is turned off is as follows.

<If the printer power is turned off when printing is not in process>

Interface	Execution API	Return Value	Remarks
USB	After GetStatus( ), execute IsCommunicationError( )	True	The status is “Communication error” due to offline.
	GetStatusCode( )	1500	
	GetStatusDirect( ), IsCommunicationError( )	True	
Serial Parallel	After GetStatus( ), execute IsCommunicationError( )	Return value before the printer power is off.	The status information before the printer power is off is the return value due to the following control conditions. • Language monitor does not monitor the power status during idling. • The current status is not inquired to the printer, it is determined by obtaining the status information from Language Monitor.
	GetStatusCode( )	Return value before the printer power is off.	
	After GetStatusDirect( ), execute IsCommunicationError( )	True	The status is “Communication error” because the status cannot be inquired to the printer.

<If the printer power is turned off when printing is in process>

Interface	API Execution API	Return Value	Remarks
USB Serial Parallel	After GetStatus( ), execute IsCommunicationError( )	True	Because Language Monitor monitors the power status during printing the status is “Communication error” without inquiring the status.
	GetStatusCode( )	1500	
	After GetStatusDirect( ), execute IsCommunicationError( )	True	The status is “Communication error” because the status cannot be inquired to the printer.