

- **Environmental Construction Procedure**
- **SDK Guide**(Android)

● **Overview**

About SDK Overview

● **API Reference**



About API Syntax

Precaution

- Please contact us if there is anything to be corrected and missed.
- This contents may change without prior notice.
Please contact us for the latest information.
- Please do not reprint, copy, duplicate or fabricate all this contents or partially without prior consent.
- Please be aware that we will not be responsible for the result arising from the usage.
- Please be aware well that we are not responsible for damage arising from incorrect usage and handling by user on contrary to this description, and for repair or modification by the third party.

About Signage

Following signage are used in this guide. Please understand the meaning of each signage before using the product.

 Note	Vital observance in use is described. Incorrect handling against this indication may cause breakdown or operation failure of the product.
 reference	Supplemental remarks and related matters are described.

Restrictions in use

Please use the product with attention to safety design of whole system by fail safety design Or redundant design to sustain reliability and safety of whole system in case of being used Applications which require for high reliability and safety in function and accuracy such as Equipment, disaster prevention and security device directly related to operation of airplane, train, vessel, auto-mobile etc.

This product is not designed for use of aerospace hardware, main communication equipment, atomic control device, medical instrument etc. which require extremely higher Reliability and safety. Please make a decision upon user's enough acknowledgement To adequateness of this product in such applications.

Table of Contents

- Overview
 - System configuration when using SDK 4
- Environmental construction procedure
 - Environmental construction 5
 - Bar code setting 6
 - Llog output 10
- API Reference
 - How to use function 12
 - Types of API 13
 - NPrinterLib(Class) 15
 - NCarryLibrary(Class) 16
 - NEnumPrinters 17
 - NRenamePrinter 18
 - NGetPrinterInf 19
 - NAutoOpen 20
 - NOpenPrinter 21
 - NOpenResult 23
 - NClosePrinter 26
 - NClosePrinters 27
 - NPrint 28
 - NDPrint 29
 - NImagePrint 30
 - NImagePrintF 31
 - NGetStatus 32
 - NGetInformation 33
 - NResetPrinter 34
 - NStartDoc 35
 - NEndDoc 36
 - NCancelDoc 37
 - NBarcode 38
 - NFirmwareDL 40
 - NFirmwareResult 41
 - NSetUSBProtocol 41
- Error code list 42
- Extended information 44

Overview

SDK enables application to have functions such as printing and monitoring printers.

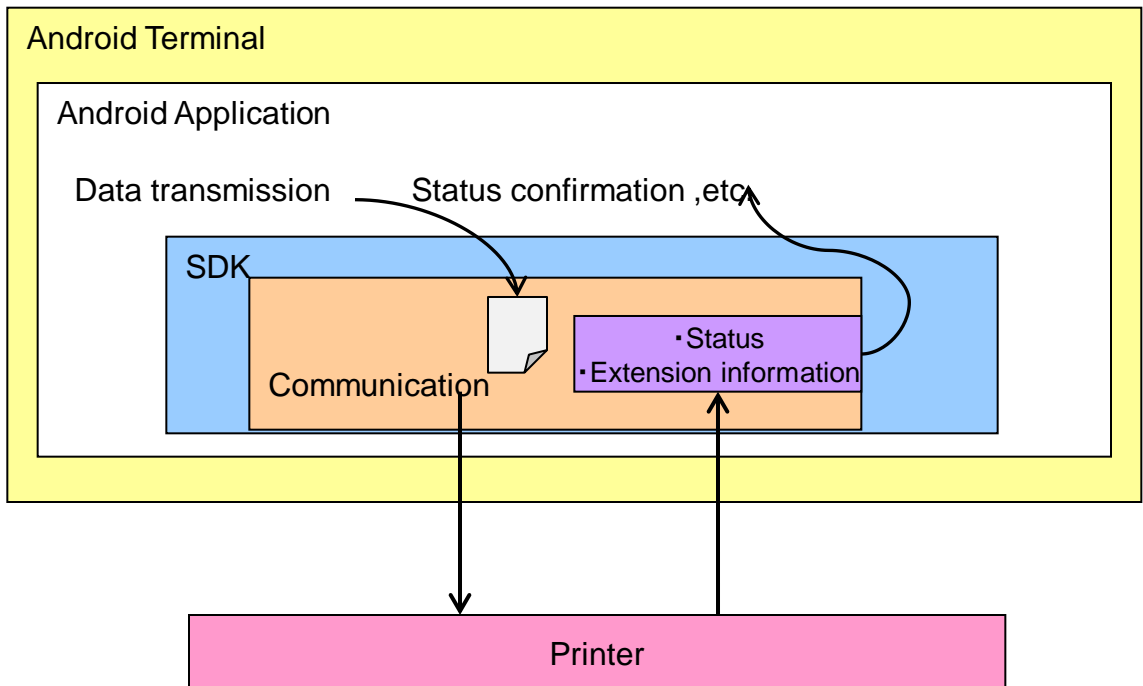
“jar” is the file distributed as SDK.

File name: NPrinterLib.jar

core.jar (For generating barcodes)

javase.jar (For generating barcodes)

System configuration when using SDK.



- **Development language**

java

- **Android compliant version**

After 4.0 (API Level 14)

- **Interface**

USB, Bluetooth(2.1)

Environmental Construction

- Store the three supplied “jar” files to “libs” folder of Android project under development. Accordingly NPrinterLib.etc.class will be available.

*Reading is available by the above method normally.

If it is unavailable, please add a corresponding “jar” file from [Project property]→[Build path of Java]→[Library tag].

- Please add the following items to AndroidManifest.xml when configuring application. (Please refer to AndroidManifest.xml of sample program for the details)

(1) Needs for reading and writing of internal storage (Please refer to the next page).

```
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE">
</uses-permission>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE">
</uses-permission>
```

(2) Needs for communication of Bluetooth.

```
<uses-permission android:name="android.permission.BLUETOOTH">
</uses-permission>
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN">
</uses-permission>
```

(3) Needs for sharing the data of NPrinterLib class.

(Transfer NPrinterLib class from main screen to another screen)

```
<application
    android:name="npi.sdk.NCarryLibrary"
```

Environmental Construction

Layout path differs depending on Android terminals.

Available by conducting **Environment.getExternalStorageDirectory()** on program.

Please confirm the configuration of same procedure **in** [Information/Setting]→[File path check] button of sample program.

Locates file in [**Environment.getExternalStorageDirectory()**]/npi/ NBarcodeInf].

Please generate npi folder by customer side.

(npi folder will be generated automatically when application is already booted.)

This guide explains the path got on [**Environment.getExternalStorageDirectory()**] described as [**Internal storage**] hereinafter.

Please make sure to acquire file path on **Environment.getExternalStorageDirectory**. Even if the same terminal, the path may differ depending on users.

About setting file

- Setting file will be explained on the next page.

On sample program, the button is set for generating each setting file inside of Android terminal(Internal storage, please refer to next page) in one time.

Each operation of sample program was made for operating with the file created by this button.

The file can be generated by [Information/Setting]→[Default setting file create] button of sample program. Please finish and reboot application once after generating file.

(Because reading the setting file is operated when application starts.)

Setting folder will be generated automatically with the name of [npi] under internal storage at the application booting. Folder will not be deleted by uninstallation of application, so please delete it manually.

About barcode setting

1. Barcode setting file layout

Generating the setting file is required for using “NBarcode” function with this SDK.

Customers need to generate this file.

Please install application for reading and writing file like file manager.

(If you have already done, unnecessary to install.)

Barcode file name: NBarcodeInf.

Layouts this file on [inter storage]/np. Please make the npi folder if the folder does not exist.

The way to describe file is shown as following page.

About barcode setting

2. How to describe barcode setting file(1D barcodes).

Please set Barcode1 ~ Barcode10 with ini file format under the following specifications.
Also, please be sure to input correctly because barcode font name and each item distinguish large and small letter.

[Barcode1] ← **Barcode1 ~ Barcode10 is assignable.(Barcode font name)**

TYPE=0
WIDTH=2
HEIGHT=100
HRI=1
HEXMODE=0
ROTATE=0
STARTBIT=0
STOPBIT=0
FILENAME=barcode01.bmp

Use the following default value when there is not NBarcodeInf file.

TYPE	: 0	STARTBIT	: 0
WIDTH	: 2	STOPBIT	: 0
HEIGHT	: 100	FILENAME	: barcodeXX. bmp
HRI	: 0		(XX is between 01 ~ 10)
HEXMODE	: 0		
ROTATE	: 0		

[Barcode2]
TYPE=1
WIDTH=3
HEIGHT=150
HRI=1
HEXMODE=0
ROTATE=0
STARTBIT=0
STOPBIT=0
FILENAME=barcode02.bmp

TYPE

0: UPC-A
1: UPC-E
2: EAN13
3: EAN8
4: CODE39
5: ITF
6: CODABAR
7: CODE128
8: CODE93

WIDTH

Assign magnification of data width between 2 ~ 4.

HEIGHT

Assign data height by dots.

HRI

0: None
1: Top (Font A)
2: Bottom (Font A)
3: Top & Bottom (Font A)
4: Top (Font B)
5: Bottom (Font B)
6: Top & Bottom (Font B)

...
...

HEXMODE

0: Input data normally (0,1,2,...)
1: Input data hexadecimally (0x30,0x31,...)

[Barcode10]

ROTATE

0: No rotation
1: 90° rotation
2: 180° rotation
3: 270° rotation

STARTBIT, STOPBIT (Only CODABAR setting is valid)

0: A
1: B
2: C
3: D

FILENAME

Assign image file name of barcode generated on [Internal storage]/npi/image.
(When blank is selected, file will not be generated)

About barcode setting

3. How to describe barcode setting file(2D barcodes)

Please set 2D-Barcode1 ~ 2D-Barcode5 with ini format and the following specifications. Also, be sure to have a right input because barcode font name and each item distinguishes large character, small character.

[2D-Barcode1] ← 2D-Barcode1 ~ 2D-Barcode5 is assignable.(Barcode font name)

TYPE=0
SIZE=100
HEIGHT=
HEXMODE=0
ERROR=L
ROTATE=0
FILENAME=2d-barcode01.bmp

[2D-Barcode2]
TYPE=1
SIZE=200
HEIGHT=8
HEXMODE=0
ERROR=
ROTATE=0
FILENAME=2d-barcode02.bmp

...
...

[2D-Barcode5]

Use the following default value if NBarcodeInf file does not exist.

TYPE : 0
SIZE : 2
HEIGHT : 1
HEXMODE : 0
ERROR : L
ROTATE : 0
FILENAME : 2d-barcodeXX.bmp (XX is between 01 ~ 05)

TYPE

0: QRCode Model2
1: PDF417

SIZE

< QRCode Model2 >
Assign magnification of data size between 2~8
< PDF417 >
Assign magnification of data size between 2~4

HEIGHT

(Only PDF417 setting is available)
Assign height ratio between 1~6

HEIGHT = height ratio × WIDTH

HEXMODE

0: Input data normally (0,1,2,...)
1: Input data hexadecimally (0x30,0x31,...)

ERROR (Error level)

< QRCode Model2 >
Assign with L,M,Q,H

< PDF417 >
Auto(Non-input), Assign with 0~8

ROTATE

0: No rotation
1: 90° Rotation
2: 180° Rotation
3: 270° Rotation

FILENAME

Assign image file name of barcode generated on [internal storage]/npi/image.
(When blank is selected, file will not be generated)

About barcode setting

4. Setting explanation of barcode file(HEXMODE).

The explanation is for the case inputting barcode data (Nbarcode byte alignment of 8th argument).

- When normal inputting (HEXMODE = 0 is selected)
If 0x31, 0x32, 0x33, 0x34 ("1234") is selected, the data will be transmitted as it is.
- When 16 hex number inputting(HEXMODE = 1 is selected)
If 0x31, 0x32, 0x33, 0x34 ("1234") is selected, ... 0x12, 0x34 will be transmitted.

5. Setting explanation of barcode file(About FILENAME)

When file name is selected on FILENAME and Nbarcode function is conducted, image file will be generated on [Internal storage]/npi/image at the timing of generation of Bitmap class.

Also, image creation path will NOT be able to change.

Please select without path (file name only).

When blank is selected/conducted, file will not be generated(Only generation of Bitmap class)

About Bluetooth connection

Pairing Android terminal and printer is required before blue tooth connecting with this SDK. Please refer to each directions of Android terminal about pairing and set.

About Setting Character Code

In this SDK, selecting character string code is available by putting a file "NCharSetInf" on [Internal storage]/npi/image. This file does not exist by default.

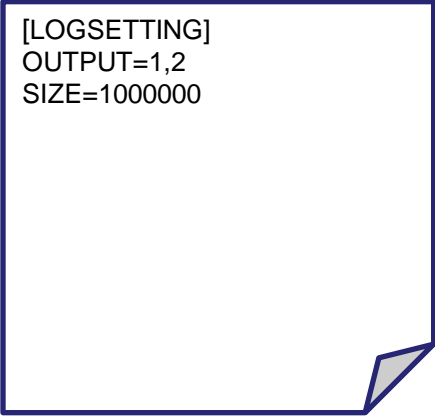
In case of handling character string of Shift_JIS code, please make "NCharSetInf" and describe "Shift_JIS" inside the file. The file will be read in at the timing of generation of "NPrinterLib."

If a Android terminal supports a character code, describing other codes to the file is available.

Moreover, loaded character codes are stored in variable number "mStrCharSet" of NprinterLid class.

About log output(Out put setting)

This SDK output log file called SDKLog under [Internal storage]/npi/log. Generating log setting file([Internal storage]/npi/NLogInf) is required for log output.
(Use the following default value when there is no file).



```
[LOGSETTING]
OUTPUT=1,2
SIZE=1000000
```

About log type (OUTPUT)

1	ERROR	: Error
2	WARN	: Warning
3	FUNC	: Function call
4	IN	: Port transmission data
5	OUT	: Port receive data

[LOGSETTING] ← Fixed

OUTPUT=1,2 ← Set log to output with comma-delimited.
(Default 1,2: output only Error, Warn)

SIZE=1000000 ← Set the maximum log file size for writing (Default 1MB).

*1 Even if file is compiled, it will not be reflected immediately.

It will be reflected at generating NPrinterLib class.

It will be reflected by rebooting application, etc.

*2 Please do not enter the space(Regardless of half size or full size)
before and after “=”.

*3 When no setting to OUTPUT, log will not be output.

*4 When incorrect value of log type(Except number 1~5, Character string) is assigned,
corresponding value will be disregarded.

Also, when incorrect value of log size(minus size value/character string),
use default value 1MB.

*5 SDK movement could be slow when the amount of data transmit is large
regarding log type4(IN), 5(OUT). (Because it takes time for log output)

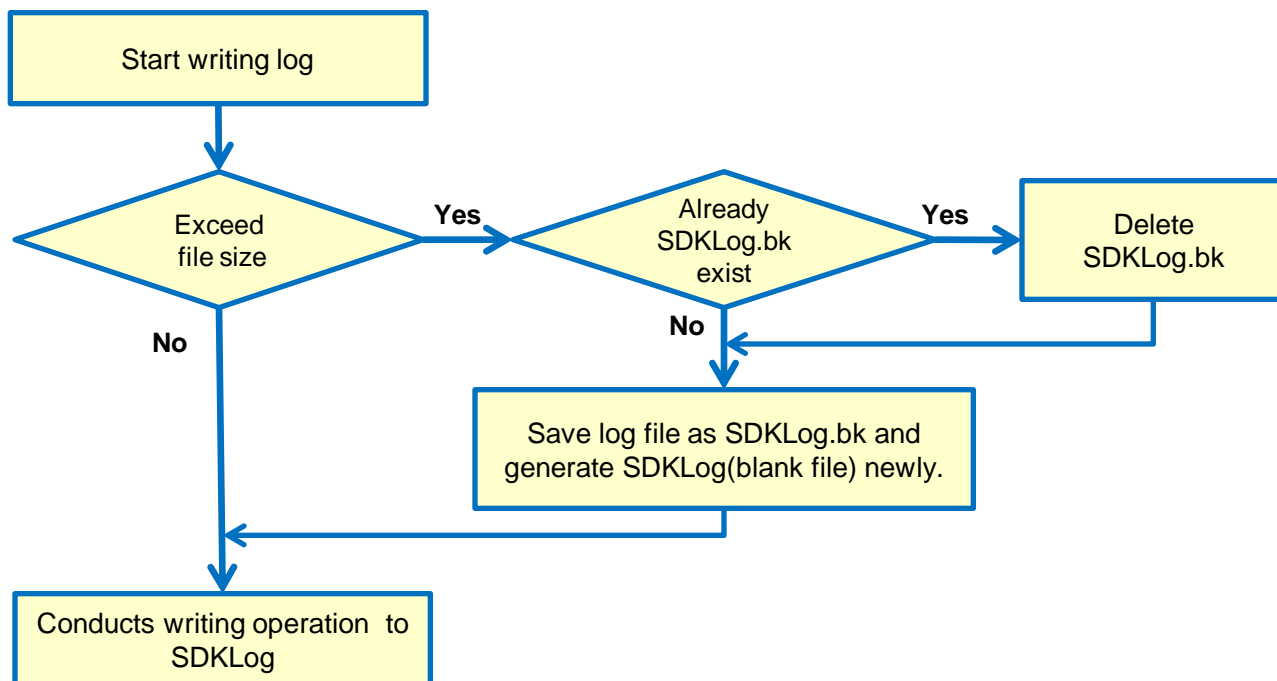
*6 Log file may not be opened when file size is assigned large.
Please change file size corresponding to the terminals to use.

Environmental construction procedure

About log output (Output file)

Also, file will be saved as past log(SDKLog.bk) and new blank log will be generated when exceeding the byte number assigned by setting file.

If there is already SDKLog.bk, it will be deleted.(Only one file for past log could be saved.)



【 Log output example】

```
2013/02/12 14:00:01.000 [FNC] xxxx.
2013/02/12 14:00:01.100 [WRN] xxxx.
2013/02/12 14:00:01.200 [ERR] xxxx.
2013/02/12 14:00:01.350 [FNC] xxxx.
2013/02/12 14:00:01.614 [I N]  xxxx.
2013/02/12 14:00:02.100 [OUT] xxxx.
2013/02/12 14:00:03.040 [I N]  xxxx.
2013/02/12 14:00:03.500 [FNC] xxxx.
2013/02/12 14:00:04.010 [FNC] xxxx.
```

Out put date by Year/Month/Day hour:
min:sec.msec .

Display log type between brackets[].

- 1 ERR : Error
- 2 WRN : Warning
- 3 FNC : Function call
- 4 I N : Port receive data
- 5 OUT : Port transmission data

How to use function

- Please generate NPrinterLib class and operate by calling each function.
Please use NCarryLibrary class when sharing NPrinterLib in multi-activities.
(Application sharing)
- When OUT parameter is assigned by SDK, use the following exclusive class.
(Except for array and objects)
Use normal format(int, boolean, etc.) when IN parameter.
Each class implements setValue (value) , getValue() functions and enables to set and get values.

(Package name: npapi.sdk.data)

NBoolean	(boolean)
NByte	(byte)
NDouble	(double)
NFloat	(float)
NInt	(int)
NLong	(long)
NShort	(short)
NString	(String)

*Please refer to sample program for details

Note

- About sequential printing and batch printing
If either Nprint, NDPrint, NImagePrint, NImagePrintF is conducted as solo, data will be transmitted to printer in each time. Batch transmission of data to printer is available by NStartDoc, NEndDoc.
Please use sequential printing and batch printing for each purpose.
- About Auto-open
This SDK implements the auto-open function to conduct processing by [Open→Process] even if output function such as NPrint・NImagePrint is called without open processing. Auto-open will not be applied by receive-related function(NGetStatus, NGetInformation). Please refer to p.20 for how to use NAutoOpen.
Please make sure to call NClosePrinter because close will not be called automatically even if auto-opened.
Receive thread will be always booted when auto-open.
Even if auto-open is failed, error value will be returned by NOpenResult function.

For example, please beware that return value of Nprint become success and return value of NOpenResult will be error when auto-open of Nprint is failed.

API Type

The following function is prepared

Application	A P I	Description
Get printer information	NEnumPrinters	Get the list of printer name
Change printer name	NRenamePrinter	Change printer name controlled by API
Get printer information	NGetPrinterInf	Get information by printer name
Auto printer setting	NAutoOpen	Set auto printer open function setting
Printer open	NOpenPrinter	Specify printer name and open
Open confirmation	NOpenResult	Get the result of NOpenPrinter function
Printer close	NClosePrinter	Close printer already opened
All printers close	NClosePrinters	Close all printers opened
Transmission of command and data	NPrint	Convert assigned hex character string data and transmit to printer
Transmission of command and data	NDPrint	Transmit assigned data to printer
Image output	NImagePrint	Transmit assigned device context to printer in raster bit image
Image output	NImagePrintF	Transmit assigned file (bmp/jpg/png) to printer in raster bit image

API Type

The following function is prepared

Application	A P I	Description
Get status	NGetStatus	Return status acquired by assigned printer
Get extended information	NGetInformation	Get information saved corresponding type ID
Reset printer	NResetPrinter	Reset printer of USB interface
Manage document	NStartDoc	Start document
Manage document	NEndDoc	Finish document
Manage document	NCancelDoc	Cancel document
Generate barcode	NBarcode	Generate barcode image
Update F/W	NFirmwareDL	Update F/W corresponding by fwf file
Remaining receive buffer setting	NSetUSBProtocol	Set confirmation of remaining receive buffer when USB transmission

Class name	NPrinterLib		
Constructor argument name	IN/OUT	Type	Description
i_context	I	Context	android.content.Context class
Return value			
Process contents			

Call each function and process after generating this class.

Log setting file and barcode setting file will be read at generating .

(This class will be required to generate again when setting file is changed)

Context class designated by argument is assigned by receiving

[Activity class name].this from generated application. (Please refer to sample program)

Ex.) Sample program

```
NPrinterLib objLib = new NPrinterLib(NPISDK.this);
```

Scraping version information (able to scrape by String)

```
objLib. SDKVERSION [Version / YYYY.MM.DD (date of release)]
```

ex)

```
SDKVERSION = "1.1.0.0 / 2014.02.14";
```


Class name	NCarryLibrary			
Constructor argument name	IN/OUT	Type	Description	
Return value				
Process contents	NPrinterLib for sharing class			

In case moving NPrinterLib class to another Activity, please use NCarryLibrary class.
Please use after setting application to AndroidManifest.xml.

Ex.) From sample program

*Set NPrinterLib class as below.

```
NPrinterLib objLib = new NPrinterLib(NPISDK.this);
```

```
NCarryLibrary objCarry = (NCarryLibrary) this.getApplication();  
objCarry.setLibrary(objLib);    // set
```

* Get class with another activity as below.

```
NCarryLibrary objCarry = (NCarryLibrary) this.getApplication();  
NPrinterLib objLib = objCarry.getLibrary();    // acquisition
```

Note

Please generate once NPrinterLib class for each application and configure to share class between each activity.

If generating class every activity, disable to succeed setting and connecting information.

Function name		NEnumPrinters	
Argument name	IN/OUT	Type	Explanation
o_printers o_size	O O	npi.sdk.data.NString npi.sdk.data.NInt	Printer name (enumerated with csv comma-delimited format) printer name byte size of o_printer (null is assignable)
Return value	int		
·Error(-Value), Normal finish(0) *Please refer to error code table(Another page)			
Process contents			
<p>·Generate printer information management file(NPrinterInf) under internal storage and save list of connectable printer name in argument [o_printers] Ex.) . PRT001,PRT002,USB-XXX,AAA</p> <p>Allocate name of “PRTxxx“ (xxx is the value between 001 ~999) and return by enumerate with csv(comma- delimited) after detecting port of USB・Bluetooth.</p> <p>Please make sure to call this function at the first time use because printer open (NOpenPrinter) is unavailable at the stage that printer information management file is not generated. Also, calling function is required when printer addition.</p> <p>In addition, o_size will not be used with Android ver. even though it exists for compatibility to Windows ver. SDK. Also, printer list acquired by this function is also available by referring to printer information management file (NPrinterInf).</p> <p>*Pairing with printer is required before calling this function when using Bluetooth.</p> <p>*Once printer name is generated, it will not be deleted by release pairing or plug off USB. In order to delete, please compile NPrinterInf file by manually. (Please delete the line which this printer name exists)</p> <p>*Maximum 999 is available for xxx allocating printer name and exceeding this number become generation error. Deletion of unnecessary data or rewriting printer name by NRenamePrinter(Described later), etc. will be required.</p>			

Function name	NRenamePrinter		
Argument name	IN/OUT	Type	Description
i_beforeprt	I	String	Printer name to change
i_afterprt	I	String	Printer name after change
Return value	int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process contents	<div>•Use when rewriting printer name in printer information management file. Although the format starting with PRT~ like “PRT001” is formed as default, printer name is changeable by this function. Also, printer name should be in the range of 50 half size alphanumeric characters and the following characters are prohibited to use. Prohibited characters to use (Including half-size space) []¥/:?*”<> ’, .</div>		

Function name	NGetPrinterInf		
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
o_ports	O	npi.sdk.data.N	Port information(Enumeration with csv
o_size	O	String npi.sdk.data.NI nt	comma-delimited format) Byte size of o_ports(null is assignable)
Return value	int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process contents			

The following information will be searched in printer information management file by printer name assigned by argument and stored.

•Connection type

USB : 1
Bluetooth : 2

•Connection information

(USB : 4 digits of vender ID+ 4digits of protocol ID= 8digits
Bluetooth : The name of paired device)

Ex.) 1,10511000
2,AAA-XX123/**0003F3598C1A**

The information acquiring by o_ports is also available by referring to printer information management file(NPrinterInf).

Also, o_size will not be used with Android ver. even though it exists for compatibility to Windows ver. (It is also available by acquiring the port information size of o_ports).

Note

In case of using the same pairing name among a number of printers with Bluetooth interface, please discern by serial ID. 12 digits character string following / of pairing name is a serial ID. (red color character string above)

Function name	NAutoOpen		
Argument name	IN/OUT	Type	Description
i_flg	I	int	Auto-printer open flag 0 : Not to auto-open (open manually) 1 : To auto-open 2 : Request to acquire status(Not to set) Other: Request to acquire status(Not to set)
Return value	boolean		
·Current(After setting) setting status of auto- printer open true : Valid auto-open, false : Invalid auto-open			
Process contents	<p>· Activate ON/OFF of auto- printer open function.</p> <p>It is available to conduct printing operation without opening in advance due to processing in the order of NOpenPrinter→NPrint→NClosePrinter when calling Nprint under the non-open condition when 1 is set.</p> <p>It is necessary to call NOpenPrinter function in activating print function, status acquisition when 0 set.</p> <p>Also, the default value of SDK is [1] and [1] is initialized by rebooting machine or etc. Please set [Not to auto-open] by this function after generation of NPrinterLib class in case not to use.</p> <p>In order to confirm the current setting, setting status is available from return value by setting [2] for argument.</p> <div><div>Note</div><p><About auto-open> Close processing will not be operated even though open processing will be operated on auto-open. Please create application to call NClosePrinter in case auto-open</p></div>		

Function name	NOpenPrinter		
Argument name	IN/OUT	Type	Description
i_prt i_statusFlg	l l	String boolean	Printer name to open Thread booting flag of status receiving true : Boot thread and receive status false : Receive at the time receiving request of status acquisition
Return value	int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process contents			

•Conduct open processing of printer

Assign printer name acquired in NEnumPrinters for argument i_prt

By assigning status receiving thread with true, boot thread and receive status at 50msec intervals. The status acquired on NGetStatus will be the updated value any time.

Receiving status will be proceeded at issuing NGetStatus without booting thread when operating with false.

Also, when the function like "Nprint" is used without calling this function, open processing will be conducted at that stage.

(Call NOpenPrinter in Nprint→Auto printer open)

However, calling NOpenPrinter function is required when [Not to auto-open] is assigned by NAutoOpen function due to [non-open error] will be occurred .

Dialog to authorize communication will be output when USB communication is not authorized to device. Outputting dialog will not be proceeded when communication is already authorized.

The result of printer open of this function (pass or fail to connect) is confirmable by calling "**NOpenResult**" function.

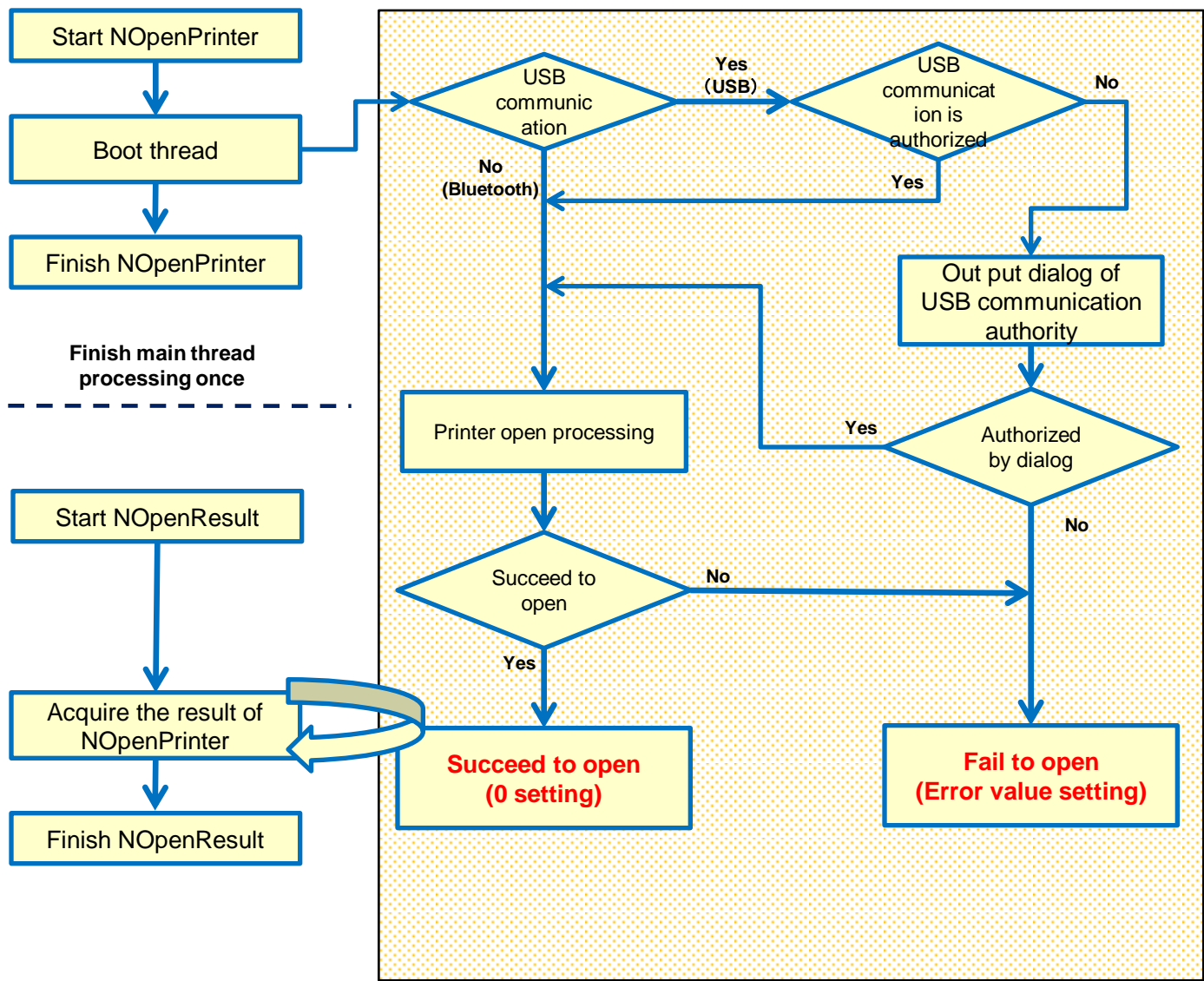
Please make sure that success value is acquired in NOpenResult and conduct printing processing such as NPrint etc.

Function name	NOpenPrinter
Process contents	<p>Also, passing through the main thread once is required after calling this function. It is impossible to confirm by sequential proceeding like NOpenPrinter→NOpenResult. Please configure application like as below. Call NOpenPrinter→Finish proceeding・・・Press button on the window→Call NOpenResult.</p> <div>Note</div> <p>Retry connection will be proceeded when turning OFF the printer opened by this function. Please press [OK] in case USB is connected when turning ON the switch again and the dialog of USB communication authority is output. when desired to stop retry process itself , please conduct stop process with “NClosePrinter” on page 26.</p> <p>Also, retry connection is only available when i_statusFlg (second argument) is true. Please use false usually.</p> <ul style="list-style-type: none">•Receive thread is always [ON] when this function is called at the auto-open.

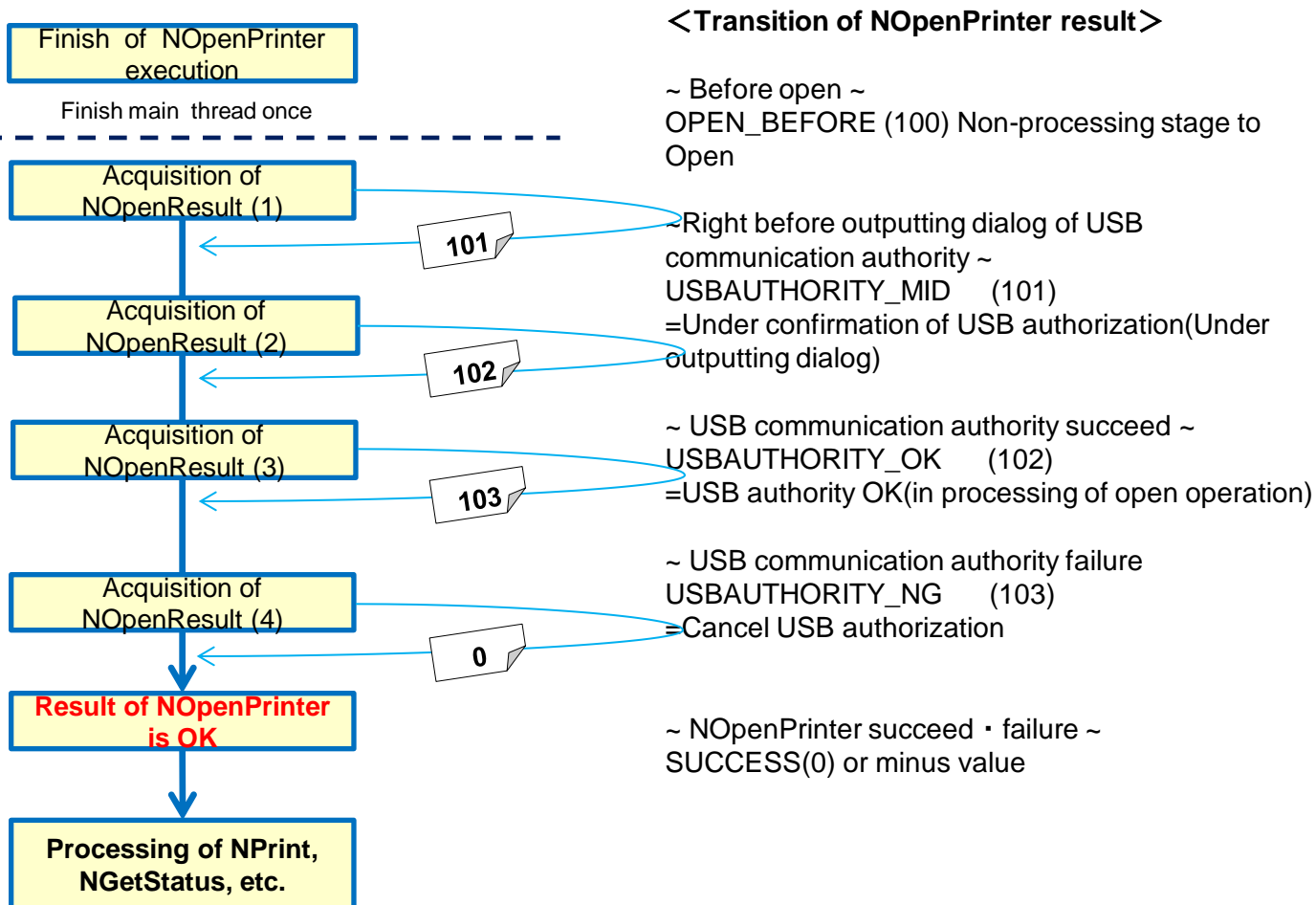
Function name	NOpenResult		
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
Return value	int		
·Error(-Value), Normal finish(0), in OpenProcess(+Value) *Please refer to error code table			
Process contents			
<div>·Acquire the result of NOpenPrinter function by this function. When the 100s value(+value) is acquired, NOpenPrinter works in progress. Please acquire until to get normal finish(0), error value(-).</div> <div><div>Reference</div><div>Checking can be made easier by calling NOpenResult in cycle with using java.util.Timer after calling NOpenPrinter. In case of returning NOpenResult to a main thread from a timer thread, please conduct it by the processing of sending message to android.os.Handler.</div></div>			

Android API Reference

NOpenPrinter / NOpenResult Outline figure of processing



Android API Reference



Configure to finish main thread once after calling NOPenPrinter.

Please operate loop calling until NOpenResult is 0 or minus value is acquired. When 100s value is acquired, it stands the status before open or proceeding USB authorization. (NOPenPrinter is not finished)

Open processing is completed at the stage SUCCESS(0) is acquired. Please follow this procedure when acquiring status or print.

Function name	NClosePrinter		
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
Return value	Int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process contents	<div>·Conduct printer close(file close) by this function. Suspend when booting thread of status receiving by NOpenPrinter. Conduct close operation by calling this function when opened by auto-open.</div>		

Function name	NClosePrinters		
Argument name	IN/OUT	Type	Description
Return value	Int		
·Normal finish(0)			
Process contents	<div>·Close all opened printers with Windows SDK. Finish all status receiving thread.</div> <div><div>Note</div><div>This function returns only succeed as return value. Please use NClosePrinter function(Close only 1pc) in order to acquire error value. since 0(Succeed) will be returned when already closed.</div></div>		

Function name		Nprint	
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
i_dat	I	String	Transmit data(Hex character string)
i_size	I	int	Output byte number
io_jobid	IO	npi.sdk.data.NInt	Print job ID(null is assignable)
Return value	int		

·Error(-Value), Normal finish(0) *Please refer to error code table

Process content

·Transmit assigned hex character string data to printer.

Also, the following three patterns will be proceeded as special data if they are detected during data analysis.

1. Character string in “(Double-quotation)
⇒ Convert as character string ("ABC" ⇒ 0x41,0x42, 0x43)
2. File name in <> (angle brackets) *including path
⇒ Out put file name (Binary data).
3. Character string which has'(Single-quotation) at the head
⇒ It will be proceeded as comment(not to be output).

Note

In case of outputting data by Shift_JIS, please put "NCharSetInf" file on [Internal storage]/npi and describe "Shift_JIS" in the file.

However, a printer must be applied Shift JIS system.

Please confirm [Japanese Kanji Code System Selection] command and memory switch with confirmation of specification. (Setting depending on printer model)

* "NCharSetInf" file is applied only NPrint functions. In addition, this file dose not exist by default.
Read in character code will be stored in "mStrCharSet" variable number as character string.

* Please calculate output byte number when using Shift_JIS.

⇒The form is String.getBytes("Shift_JIS").length.

Function name		NDPrint	
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
i_dat	I	byte[]	Transmit data(Hex character string)
i_size	I	int	Output byte number
io_jobid	IO	npi.sdk.data.NInt	Print job ID(null is assignable)
Return value		Int	
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content	·Transmit assigned data to printer(No conversion)		

Function name		NImagePrint	
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
i_bmp	I	Bitmap	Bitmap class for Android (android.graphics.Bitmap)
i_width	I	int	Width
i_height	I	int	Height
i_putType	I	byte	Output method 0x00 : Raster format line unit 0x01 : Raster format block unit 0x02 : Raster format gray scale display of block unit
io_jobid	IO	npi.sdk.data.N Int	0x10 : Bit image format Print job ID (null is assignable)
Return value	Int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content			
·Transmit assigned bitmap class to printer. It is impossible to assign width・height beyond bitmap class size			
<div>Note</div> ·It takes time to out put Raster format gray scale display of block unit in comparison with other format. Moreover, there are models which are not applied this function.			

Function name	NImagePrintF		
Argument name	IN/OUT	Type	Description
i_prt i_bmp i_putType io_jobid	I I I IO	String String byte npi.sdk.data. NInt	Printer name Image file name(Including path) Output method 0x00 : Raster format line unit 0x01 : Raster format block unit 0x02 : Raster format gray scale display of block format 0x10 : Bit image format line unit Print job ID(null is assignable)
Return value	int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content			
·Transmit assigned bmp/jpg/png file to printer.			
<div><div>Note</div><p>·It takes time to out put Raster format gray scale display of block unit in comparison with other format. Moreover, there are models which are not applied this function</p></div>			

Function name	NGetStatus		
Argument name	IN/OUT	Type	Description
i_prt o_status	I O	String npi.sdk.data.NInt	Printer name Status
Return value	int		
·Error(-Value), Normal finish(0), Alarm(+Value) *Please refer to error code table			
Process content			
<div>·Return acquired status on assigned printer. *Please refer to 《ESC v》 command on specification of corresponding printer for returned value.</div>			

Function name		NGetInformation	
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
i_id	I	byte	ID type
o_dat	O	byte[]	Storing area of extended information
o_time	O	npi.sdk.data. NLong	Update flag(Acquire the time of system with msec)(null is assignable)
Return value		Int	
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content	<div>▪Acquire the information saved in ID for corresponding type of extended information. * It is required to transmit request for necessary information from higher application to printer in advance. (There are some cases which do not require the requests such as extended status, transfer completion and request for information of print completion, etc.) * Please refer to 《ESC s》 command on specification of corresponding printers regarding return value.</div>		

Function name		NResetPrinter	
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
Return value	int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content	<div>•Reset printer. Cancel print job in proceeding. This function is only available when USB connection. Error occurs when Bluetooth connection.</div> <div>•Please confirm that return value of this API and printer will be reset and online in order to verify this API is proceeded correctly. (Confirm NGetStatus)</div> <div>*There are some models which will be offline for constant time after conducting this function by printer. Please press [OK] and reconnect when offline and the dialog of USB communication authority will be output by retry connection. Error occurs when conducting this function before reconnecting.</div>		

Function name	NStartDoc		
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
o_jobid	O	npi.sdk.data.NInt	Print job ID
Return value	Int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content	<div>· Start document. Buffer data of NPrint,NDPrint,NImagePrint,NImagePrintF in memory after issuing NStartDoc. Buffered data can be output to printer by calling NEndDoc. Also, clear buffered data by calling NCancelDoc. 1 document is available by 1 printer.</div>		

Function name	NEndDoc		
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
Return value	Int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content	<div>·Finish document. Output buffered data to printer after calling NStartDoc. Not to proceed when buffered data do not exist.</div>		

Function name	NCancelDoc		
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
Return vale	int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content	·Document will be cancelled. Not to proceed when buffered data do not exist.		

Function name	NBarcode		
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
i_fontName	I	String	Font name of barcode
i_bmp	IO	Bitmap	Bitmap class for Android (android.graphics.Bitmap)
i_x	I	int	Left
i_y	I	int	Top
i_width	I	int	Width
i_height	I	int	Height
i_dat	I	byte[]	Barcode data
i_size	I	int	Data size
Return value	int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content	<div>· Draw barcode/2D barcode specified in printer's barcode font/2D barcode font on designated bitmap.</div> <div>Please assign font name used in barcode setting file (NBarcodeInf) for font name of the second argument.</div> <div>Please refer to [About barcode setting] on this guide for NBarcodeInf.</div> <div><div>Note</div><div>· Also, barcode data deleting excessive part is generated without error even if barcode data is large and generated barcode exceed the size. In this case, please adjust data size to set all data in the size since it is impossible to read generated barcode.</div></div>		

Function name	NBarcode
Process content	Specified Image figure of argument

Bitmap class area(Third argument・the part in gray box)



*Set barcode width・height・with or without HRI character, etc. **in red box** on barcode setting file (NBarcodeInf).

*Barcode with red frame will rotate by rotate setting of barcode. Please set area with width and height after rotation since the white part in the above figure will not rotate

Function name		NFirmwareDL	
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
i_file	I	String	File name of fwf(NULL is assignable)
i_errflg	I	byte	Error check 0x00 : Invalid(Forcible transmission) 0x01 : Valid
io_chksum	IO	np.sdk.data.N Short	Check sum of fwf file (null is assignable, imperative when firm ware check)
io_jobid	IO	np.sdk.data.N Int	Print job ID(null is assignable)
Return value	int		
·Error(-value), Normal finish or check sum match (0) *Please refer to error code table			
Process content	<div>▪ Transmit assigned fwf file to printer. (firmware download)</div> <div>Also, acquire check sum from printer and compare with check sum assigned by argument when fwf file is assigned by null. (Firmware check)</div> <div>Even if error occurs on printer status, forcible transmission will be operated when error check is assigned with 0x00.</div> <div><div>Note</div><div>▪ It is unavailable during starting document. Please call NEndDoc, NCancelDoc and finish document.</div><div>▪ In casa of error check with valid, processing result must be scraped by “NFirmwareResult,” function which is mentioned below.</div></div>		

Function name		NFirmwareResult	
Argument name	IN/OUT	Type	Description
i_prt	I	String	Printer name
Return value	int		
·Error (- value), Normal End (0) 、 Firmware DL processing (+ value) * Please refer to error code table			
Process content	<div>·Result of NFirmwareDL function (except for Error Check with valid and scraping Checksum) is scraped by this function. *In case of scraping + value, NFirmwareDL is in processing. *+ value indicates status value.</div> <div>Please keep scraping until achieving normal end (0) or error value (1).</div> <div>When NFirmwareDL (error check with valid) is in practice, return value takes following process</div> <div>1. Before calling NFirmwareDL function (return vale 0) 2. Downloading Firmware (return vale 128) ···Bit 7 of status turns ON 3. Finishing downloading and reset process (return value 255) ···Turning OFFLINE temporaly 4. Reconnection dialog showing (able to reconnect printer by selecting OK in this step) 5. With ending normally after calling NFirmwareDL, return value will be 0.</div> <div>*After finishing firmware download, printer resets. Before printer resets, error is detected even if trying to call this function. Please construct program in such a way as to scrape download result by this function after reconnection.</div>		

Function name	NSSetUSBProtocol		
Argument name	IN/OUT	Type	Description
i_prt i_type	I I	String int	Printer name Confirmation flag of receive buffer remaining amount. 0 : Supervise and transmit receive buffer remaining amount(Default). 1 : Transmit receive buffer remaining amount without confirmation. 2 : Return setting value(by return value) without setting.
Return value	int		
·Current(After setting) value of confirmation flag of receive buffer amount(0,1) or error(-value) *Please refer to error code table(Another page)			
Process content			
· Flag for whether or not to transmit after confirming receive buffer remaining amount. The default value is [0] and confirm receive buffer remaining amount and transmit.. Please make sure to set [1] regarding a printer which does not respond to receiving buffer remaining amount.			

Error code list(1/2)

SUCCESS	0	Normal finish
ERR_PRTOPEN	-2	Printer open error
ERR_OFFLINE	-5	Offline
ERR_PRTCLOSE	-6	Printer close error
ERR_FILEOPEN	-10	File open error
ERR_PRTOUTPUT	-13	Printer output error
ERR_PRTINPUT	-14	Receive error of printer data
ERR_CHARSET	-15	Character code is not supported
ERR_PRTALREADYOPEN	-21	Printer is already opened
ERR_NOHANDLE	-22	Printer is not opened
ERR_LACKRESOURCE	-31	Lack of resource
ERR_NOTSUPPORTED	-40	Not supported by Android SDK
ERR_LOADFROMFILE	-50	Failed to read image file
ERR_IMAGESIZE	-51	Incorrect image size
ERR_LOADBITMAP	-52	Failed to read image data from bitmap class
ERR_RESETPRINTER	-60	Failed to reset
ERR_STARTDOC	-70	StartDoc function error
ERR_DOCNOTSTARTED	-71	Not status of start document
ERR_ALREADYSTARTED	-72	Already status of start document
ERR_FWFFILE	-80	fwf file error
ERR_FWF_CHECKSUM	-81	Not match checksum entered with argument and checksum acquired from printer
ERR_FWDL_TIMEOUT	-82	Time out of firmware download (Check of print start command is timeout)
ERR_FWCHK_TIMEOUT	-83	Time out of confirmation of firmware checksum
ERR_FWCHK_FOUNDERERROR	-84	Error is detected on the status confirmation at firmware downloading
ERR_ARGUMENT	-90	Incorrect argument value error
ERR_ARGUMENT_01	-91	1st argument is incorrect
ERR_ARGUMENT_02	-92	2nd argument is incorrect
ERR_ARGUMENT_03	-93	3rd argument is incorrect
ERR_ARGUMENT_04	-94	4th argument is incorrect
ERR_ARGUMENT_05	-95	5th argument is incorrect
ERR_ARGUMENT_06	-96	6th argument is incorrect
ERR_ARGUMENT_07	-97	7th argument is incorrect
ERR_ARGUMENT_08	-98	8th argument is incorrect
ERR_ARGUMENT_09	-99	9th argument is incorrect

Error code list(2/2)

ERR_PRTINFO_CREATE	-130	Failed to create printer information file
ERR_PRTINFO_READ	-131	Failed to read printer information file
ERR_PRTINFO_WRITE	-132	Failed to write printer information file
ERR_PRTNAME_ALLOC	-133	Failed to allocate printer name
ERR_PRTRENAME_BEFORE	-134	Printer name before change is not found
ERR_PRTRENAME_AFTER	-135	Printer name after change is already used
ERR_PRTINFO_GET	-136	Failed to acquire Printer information.
ERR_PRTINFO_ILLEGAL	-137	Incorrect printer information file
ERR_DEVICE_NOTSUPPORT	-150	Connection type is not supported
ERR_BCDSETTINGFILE	-160	Barcode setting file is incorrect
ERR_CREATEBCDFILE	-161	Failed to create barcode file
ERR_CREATEBCDDATA	-162	Failed to create barcode data
OPEN_BEFORE	100	Open is unproceeded
USBAUTHORITY_MID	101	Under confirmation of USB authorization
		(Under outputting dialog)
USBAUTHORITY_OK	102	Succeed USB authorization
		(Under open proceeding)
USBAUTHORITY_NG	103	Cancel USB authorization
NFirmwareResult return value		
	128	Downloading firmware (Bit7 of status is ON)
	255	Finishing download and reset processing (Turn OFFLINE contemporaly)

Extended information

- Type 1 : 4 byte (Fixed) : Update flag(4 byte) <Extended status> 1Byte : 7~0, 2Byte:15~8, 3Byte : 23~16, 4Byte : 31~24
- Type 2 : 32 byte (delimiter) : Update flag(4 byte) <Model name>
- Type 3 : 8 byte(Fixed) : Update flag (4 byte) <F/W version>
- Type 4 : 8 byte(Fixed) : Update flag (4 byte) <Boot version>
- Type 5 : 4 byte(Fixed) : Update flag (4 byte) <Reserve>
- Type 6 : 4 byte(Fixed) : Update flag (4 byte) <Dot line number of current head>
- Type 7 : 4 byte(Fixed) : Update flag (4 byte) <Dot line to drive>
- Type 8 : 4 byte(Fixed) : Update flag (4 byte) <Cut times>
- Type 9 : 16 byte(Fixed) : Update flag (4 byte) <User maintenance counter:
Current dot line number,
Dot line number to drive,
Cut times,
backup>
- Type 10 : 16 byte (Fixed) : Update flag (4 byte)<Reserve>
- Type 11 : 64 byte (delimiter) : Update flag (4 byte)
- Type 12 : 32 byte (delimiter) : Update flag (4 byte)
- Type 13 : 32 byte (Fixed) : Update flag (4 byte) <NV registration status>
- Type 14 : 32 byte (Fixed) : Update flag (4 byte) <Reserve>
- Type 15 : 16 byte (Fixed) : Update flag (4 byte)
- Type 16 : 16 byte (Fixed) : Update flag (4 byte)
- Type 17 : 16 byte (Fixed) : Update flag (4 byte)
- Type 18 : 16 byte (Fixed) : Update flag (4 byte)
- Type 19 : 8 byte (Fixed) : Update flag (4 byte)<Notice of print completion: Arbitrary ID and finish status will be described at proceeding finish command by assigning print start/finish command.>
- Type 20 : 8 byte (Fixed) : Update flag (4 byte) <Reserve>
- Type 21 : 8 byte (Fixed) : Update flag (4 byte)
- Type 22 : 8 byte (Fixed) : Update flag (4 byte)
- Type 23 : 8 byte (Fixed) : update flag (4 byte)
- Type 24 : 4 byte (Fixed) : Update flag (4 byte)<Reserve>
- Type 25 : 4 byte (Fixed) : Update flag (4 byte)<Notice of transfer completion: Transferred job ID>
- Type 26 : 4 byte (Fixed) : Update flag (4 byte)<Reserve>
- Type 27 : 4 byte (Fixed) : Update flag (4 byte)<Reserve>
- Type 28 : 2 byte (Fixed) : Update flag (4 byte)<F/W check sum>
- Type 29 : 2 byte (Fixed) : Update flag (4 byte)
- Type 30 : 2 byte (Fixed) : Update flag (4 byte)
- Type 31 : 2 byte (Fixed) : Update flag (4 byte) <Information of communication status:
USB: 0x0000 fixed COM : 1st byte CTS Second byte DSR
*Timestamp to acquire final signal status to update flag>

*There are no validity about acquired information which do not have the function with printer except 25, 31.

* Described contents do not mean that they are available with all of printers.