

- **Environmental Construction Procedure**
- **SDK Guide**

Overview

About SDK Over

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
- API Reference
 - Types of API 7
 - NEnumPrinters 9
 - NDeletePrinter 10
 - NRenamePrinter 11
 - NGetPrinterInf 12
 - NOpenPrinter 13
 - NClosePrinter 14
 - NPrint 15
 - NDPrint 16
 - NImagePrint 17
 - NImagePrintF 18
 - NGetStaus 19
 - NGetInformation 20
 - NResetPrinter 21
 - NStartDoc 22
 - NEndDoc 23
 - NCancelDoc 24
 - NBarcode 25
 - NFirmwareDL 26
 - NSetUSBProtocol 27
 - NDeleteJob 28
- Error code list 29
- Extended information 31

Overview

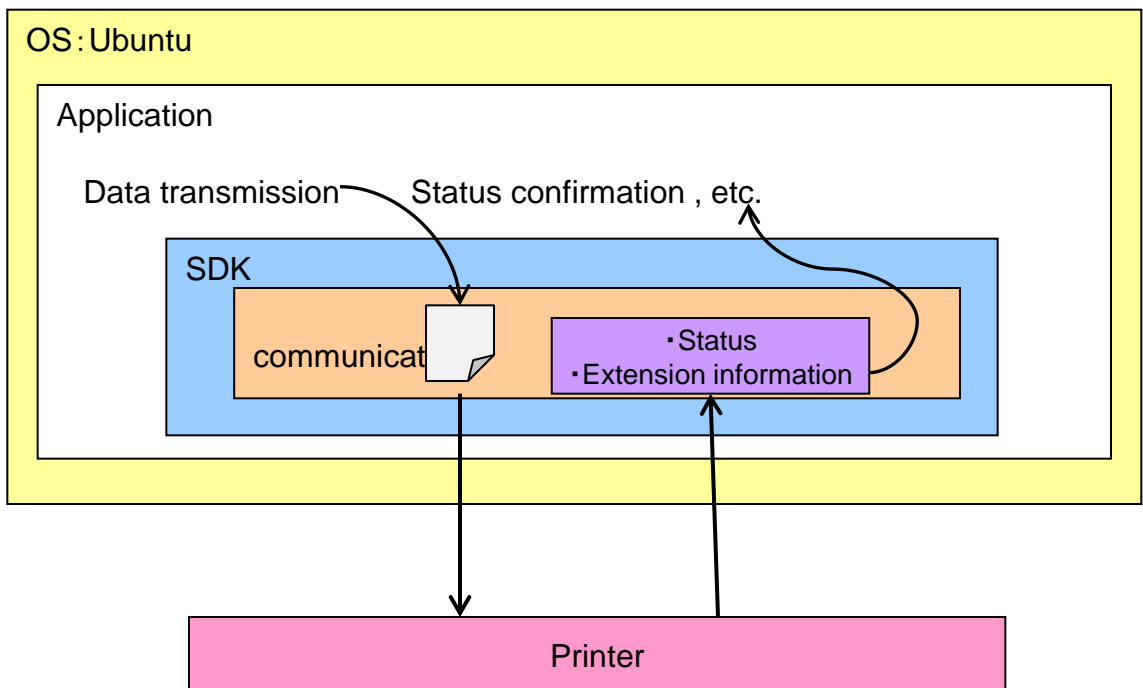
SDK enables application to have function such as printing and monitoring printers.

“libNPreint.so.1.2 is the file distributed as SDK.

Caution

Since SDK uses libusb and OpenCV library, installation of the above library is necessary.

System configuration when using SDK.



- **Development Language**
C Language
- **Ubuntu compliant version**
14.04, 16.04
- **Interface**
USB, COM

Environmental Construction

- Install the following library

- libusbLibrary

EX: Install at command line

sudo apt-get install libusb-dev

- OpenCV Library

EX: Install at command line

sudo apt-get install libopencv-dev

- Folder for saved SO library

- Please save in `"/usr/lib/` OR `"/usr/local/lib/`

※ Please update a dependence relationship information of the shared library by
“ldconfig”

command regarding saved library.

Caution

**SDK cannot be used since dependency is not solved with “ldconfig”
without install libusb,OpenCV library.**

Character code setting

SDK enables to specify character string conversion code inside of SDK by allocating a file `NCharSetInf` under `“usr/np”`. This file does not exist by default.

In case of using character string of `Shift_JIS`, please use `NCharSetInf` file and enter `“Shift_JIS”` in the file. The file is read when executing method `NOpenPrinter`.

Serial connection

Monitoring power supply's state is judged by CTS signal line, however allocating a file `COM` under `“usr/np”` enables not to monitor power supply's state by CTS signal line and to become always online state setting.

Please note that serial connection is specification only for hardware flow control.

Sample code compile

- Please execute the following command for sample code compile.

```
$gcc libNPrint_sample.c /usr/local/lib/libNPrint.so.1 -o test `pkg-config --cflags --libs opencv` -lm -lpthread
```

Ref.

Specify /usr/local/lib/libNPrint.so.1 for compile.
Also, link by specifying OpenCV for -libs.

API Type

The following function is prepared

Application	A P I	Description
Get printer information	NEnumPrinters	Get the list of printer name
Delete printer name	NDeletePrinter	Delete the printer name (specific or all)
Change printer name	NRenamePrinter	Change printer name controlled by API
Get printer information	NGetPrinterInf	Get information by printer name
Printer open	NOpenPrinter	Specify printer name and open
Open confirmation	NOpenResult	Get the result of NOpenPrinter function
Printer close	NClosePrinter	Close printer already 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
Get status	NGetStatus	Return status acquired by assigned printer
Get extended information	NGetInformation	Get information saved corresponding type ID

API Type

The following function is prepared

Application	A P I	Description
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
Document control	NDeleteJob	Delete document, check the remaining document

Function name		NEnumPrinters	
Argument name	IN/OUT	Type	Explanation
o_printers	O	char*	Printer name (enumerated with csv comma-delimited format)
o_size	O	unsigned int*	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 “usr/npil/” and save list of connectable printer name in argument [o_printers]</p> <p>Ex.) PRT001,0,/dev/ttyS0</p> <p> PRT002,1,/dev/usb/lp0</p> <p> PRT003,1,/dev/usb/lp2</p> <p> PRT004,1,/dev/ttyS1</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.</p> <p>Please make sure to call this function for the first time use because printer open (NOpenPrinter) is unavailable at the stage where printer information management file is not generated. Also, to call function is required when adding printer.</p> <p>*Once printer name is generated, it will not be deleted by plug off USB.</p> <p>In order to delete, please compile NPrinterInf file by manually.</p> <p>(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> <p>*0: Com port(/dev/ttyS*) 1: USB(/dev/usb/lp*)</p>			

Function name	NDeletePrinter		
Argument name	IN/OUT	Type	Description
i_prt	I	Char*	Printer name to delete
Return value	Int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process contents	<div>·Use when deleting printer name in printer information management file. Specify printer name to delete at “i_prt” of argument Delete printer information management file itself by specifying blank or NULL.</div> <div>*Please do not use this function if NOpenPrinter method is executed.</div>		

Function name	NRenamePrinter		
Argument name	IN/OUT	Type	Description
i_beforeprt	I	char*	Printer name to change
i_afterprt	I	char*	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) []¥/ : ? * ” < > ’ , . *Please do not use this function if NOpenPrinter method is executed and it is in open state. Closing process will not be executed by NClosePrinter method.</div>		

Function name	NGetPrinterInf		
Argument name	IN/OUT	Type	Description
i_prt o_ports o_size	I O O	char* char* Unsigned long*	Printer name Port information(Enumeration with csv 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	<p>The following information will be searched in printer information management file by printer name assigned by argument and stored.</p> <p>·Connection type COM : 0 USB : 1</p> <p>Ex.) 0,/dev/ttyS0 1,/dev/usb/lp1</p> <p>The information acquiring by o_ports is also available by referring to printer information management file(NPrinterInf).</p>		

Function name	NOpenPrinter		
Argument name	IN/OUT	Type	Description
i_prt i_statusFlg	I I	char* unsigned char	Printer name to open Thread booting flag of status receiving 0: Receive at the time receiving request of status acquisition 1: Boot thread and receive status (without automatic reconnecting process when sending error) 2: Boot thread and receive status (with automatic reconnecting process when sending error)
Return value	Int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process contents			
<p>•Conduct open processing of printer Assign printer name acquired in NEnumPrinters for argument i_prt</p> <p>By assigning status receiving thread with “1” or “2”, 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 “0”. (*”0” not supported)</p>			

Function name	NClosePrinter		
Argument name	IN/OUT	Type	Description
i_prt	I	char*	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.</div>		

Function name		Nprint	
Argument name	IN/OUT	Type	Description
i_prt	I	char*	Printer name
i_dat	I	char*	Transmit data(Hex character string)
i_size	I	unsigned int	Output byte number
io_jobid	IO	unsigned int*	Print job ID(null is assignable)
Return value	int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content			
<div>·Transmit assigned hex character string data to printer. Also, the following two patterns will be proceeded as special data if they are detected during data analysis.</div> <div>1. Character string in “(Double-quotation) => Convert as character string ("ABC" => 0x41,0x42, 0x43)</div> <div>2. File name in <> (angle brackets) *including path =>Out put file name (Binary data).</div> <div>3. File name in [] (Bracket) =>Out put Image for Bit map file in raster block format.</div> <div><div>Note</div><div>Please store NCharSetInf file under “usr/npi/” in case of outputting in Shift_JIS. Please enter “Shift_JIS” in the file. And, a printer has to be in Shift_JIS format. Please confirm a printer specification and check “Japanese Kanji Code Selection” command/memory switch setting(varies depending on printers).</div></div> <div>*NCharSetInf file is applied only for Nprint function. And this file does not exist by default.</div>			

Function name		NDPrint	
Argument name	IN/OUT	Type	Description
i_prt	I	char*	Printer name
i_dat	I	unsigned char*	Transmit data(Hex character string)
i_size	I	unsigned int	Output byte number
io_jobid	IO	unsigned int*	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_bmp i_width i_height i_putType	I	char* IpImage unsigned int unsigned int Unsigned char	Printer name Address of image (*OpenCV library) Width Height Output method 0x00 : Raster format line unit 0x01 : Raster format block unit 0x02 : Raster format gray scale display of block unit 0x10 : Bit image format
o_jobid	O	unsigned int*	Print job ID (null is assignable)
Return value	Int		
·Error(-Value), Normal finish(0) *Please refer to error code table			
Process content			
·Transmit assigned image to printer. It is impossible to assign width·height beyond image size			
<div>Note</div> ·It takes time to out put Raster format gray scale display of block unit in comparison with other format. Some printers do not support the function. ·Please install and use OpenCV library.			

Function name	NImagePrintF		
Argument name	IN/OUT	Type	Description
i_prt i_bmp i_putType o_jobid	I O	char* char* unsigned char unsigned int*	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><div>·It takes time to out put Raster format gray scale display of block unit in comparison with other format. Some printers do not support the function. ·Please install and use OpenCV library.</div></div>			

Function name	NGetStatus		
Argument name	IN/OUT	Type	Description
i_prt	I	char*	Printer name
o_status	O	unsigned long*	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.</div> <div>*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	char*	Printer name
i_id	I	Unsigned char	ID type
o_dat	O	void*	Storing area of extended information
o_time	O	unsigned long*	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			

- 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.

Function name		NResetPrinter	
Argument name	IN/OUT	Type	Description
i_prt	I	char*	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.</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><div>Note</div><div>·Please install and use libusb library.</div></div>		

Function name	NStartDoc		
Argument name	IN/OUT	Type	Description
i_prt	I	char*	Printer name
o_jobid	O	unsigned int*	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	l	char*	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	l	char*	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	char*	Printer name
i_fontName	I	char*	Font name of barcode
i_bmp	IO	IpImage*	Address of image (*Open CV libaray)
i_x	I	unsigned int	Left
i_y	I	unsigned int	Top
i_width	I	unsigned int	Width
i_height	I	unsigned int	Height
i_dat	I	unsigned char*	Barcode data
i_size	I	unsigned int	Data size
Return value	Int		
· Error(-Value), Normal finish(0) *Please refer to error code table			
Process content	· Not supported (always returns “SUCCESS”)		

Function name		NFirmwareDL	
Argument name	IN/OUT	Type	Description
i_prt	I	char*	Printer name
i_file	I	char*	File name of fwf(NULL is assignable)
i_errflg	I	unsigned char	Error check 0x00: Invalid(Forcible transmission) 0x01: Valid
io_chksum	IO	Unsigned int*	Check sum of fwf file (null is assignable, imperative when firm ware check)
o_jobid	O	Unsigned 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>			

Function name	NSSetUSBProtocol		
Argument name	IN/OUT	Type	Description
i_prt i_type	I I	char* unsigned 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		
<div>·Current(After setting) value of confirmation flag of receive buffer amount(0,1) or error(-value)</div> <div>*Please refer to error code table(Another page)</div>			
Process content			
<div>▪ Not supported (always returns “SUCCESS”)</div>			

Function name	NDeleteJob		
Argument name	IN/OUT	Type	Description
i_prt i_cmp	I I	char* unsigned char	Printer name Document flag 0: Delete queued first document 1: Delete first document (including in transmitting document) 2: Check a number of queued documents after deleting.
Return value	int		
<div>·Delete and check a number of pending document s or error(-value) *Please refer to error code table(Another page)</div>			
Process content	<div>· Document deletion or check a number of documents.</div> <div><div>Note</div><div>· Print data enclosed by StartDoc method/EndDoc method becomes one document. · About document In this SDK, print data executed by Nprint/NDPrint/NImagePrint/NImagePrintF method is stored once as a file(document) under “usr/npi/spool” and is processed sequentially.</div></div>		

Error code list(1/2)

SUCCESS	0	Normal finish
ERR_HANDLE	-1	Handle error
ERR_PRTOPEN	-2	Printer open error
ERR_OFFLINE	-5	Offline
ERR_UNEXPLAINED_OFFLINE	-6	Offline (Output error<transmission error when submitting write method>) * This error is not released until read method succeed. * Using write method only (All read method error is ERR_OFFLINE(-5))
ERR_FILEOPEN	-10	File open error
ERR_PRTOUTPUT	-13	Printer output error
ERR_FILEWRITE	-15	File write error
ERR_FILEREAD	-16	File read error
ERR_MAKEDIR	-18	Directory create error
ERR_PRTALREADYOPEN	-21	Printer is already opened
ERR_NOHANDLE	-22	Connecting information does not exist
ERR_LACKRESOURCE	-31	Lack of resource
ERR_LOADFROMFILE	-50	Failed to read image file
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_ENDDOC	-73	EndDoc function error
ERR_FWFFILE	-80	fwf file error
ERR_FWF_CHECKSUM	-81	Not match checksum entered with argument and checksum acquired from printer
ERR_FWCHK_TIMEOUT	-83	Time out of confirmation of firmware checksum
ERR_FWCHK_FOUNDERRO R	-84	Error is detected on the status confirmation at firmware downloading

Error code list(2/2)

ERR_ARGUMENT	-90	Incorrect argument value error
ERR_ARGUMENT_01	-91	1st argument error
ERR_ARGUMENT_02	-92	2nd argument error
ERR_ARGUMENT_03	-93	3rd argument error
ERR_ARGUMENT_04	-94	4th argument error
ERR_ARGUMENT_05	-95	5th argument error
ERR_ARGUMENT_06	-96	6th argument error
ERR_ARGUMENT_07	-97	7th argument error
ERR_ARGUMENT_08	-98	8th argument error
ERR_ARGUMENT_09	-99	9th argument error
ERR_PRTINFO_GET	-136	Failed to acquire Printer information.
ERR_PRTINFO_ENTRYMAX	-138	Disable to process due the printer information reaching to MAX value

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.