



Brother Windows DLL

Software Developer's Manual

Copyright

©2020 Brother Industries, Ltd. All rights reserved.

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement. The software may be used or copied only in accordance with the terms of those agreements. No part of this publication can be reproduced in any form or by any means without prior written permission of Brother Industries, Ltd.

Trademarks

Brother is either a trademark or a registered trademark of Brother Industries, Ltd.

Any trade names and product names of companies appearing on Brother products, related documents and any other materials are all trademarks or registered trademarks of those respective companies.

Important Note

This document provides information for the label printer (See [4.1 Supported Printers](#)) and is available for use only if you agree to the following conditions:

No Warranty

- a. The contents of this document are subject to change without notice.
- b. Regarding this document, Brother will not guarantee that there are no defects or if the purpose matches the specific purpose, regardless of whether it is explicit or implied.
- c. Brother assumes no responsibility for damages, actions of third parties, the user's intentional or negligent operation, misuse, or operation under other special conditions.

Contact Information

For more information for mobile printer developers, visit <https://support.brother.com/g/s/es/dev/en/index.html?navi=offall>

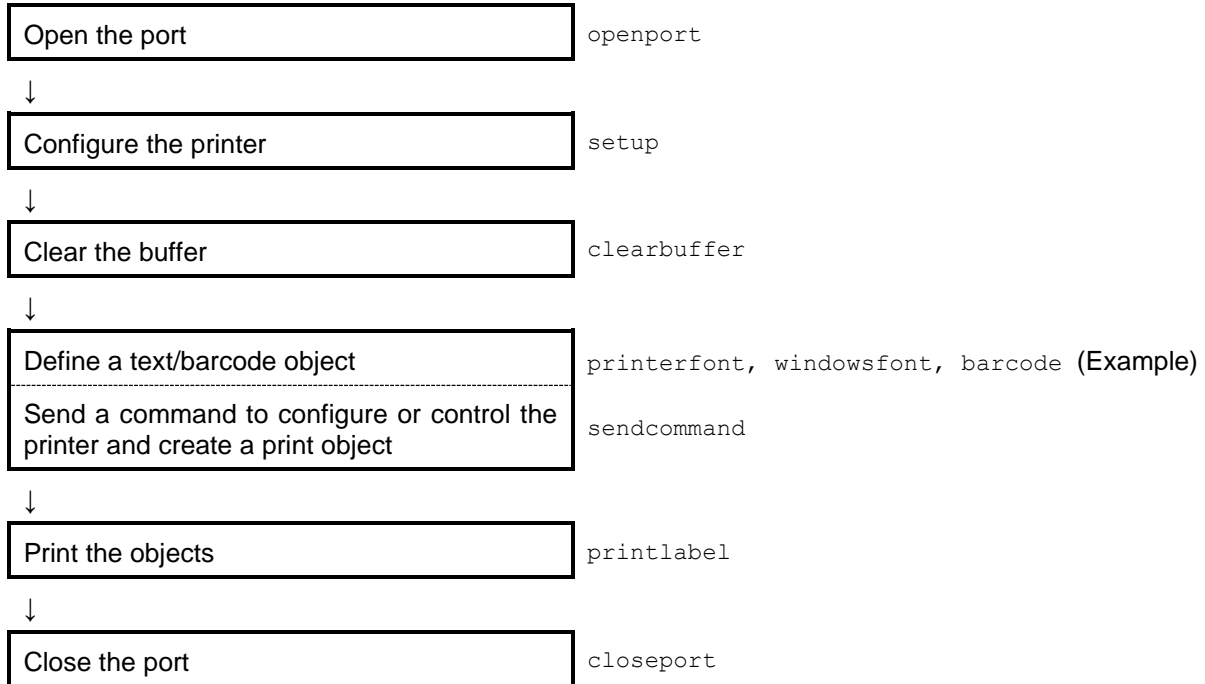
Table of Contents

1 Guide	1
1.1 Printing a Label.....	1
1.2 Printing using the Printer Commands.....	1
1.3 Printing using an FBPL Template	2
1.4 Printing an Image.....	3
1.5 Encoding an RFID Tag	4
2 API	5
2.1 openport	5
2.2 opennetport	5
2.3 closeport.....	6
2.4 setup.....	7
2.5 clearbuffer.....	7
2.6 barcode	8
2.7 printerfont	10
2.8 sendcommand	11
2.9 printlabel.....	12
2.10 downloadpcx.....	12
2.11 formfeed	12
2.12 nobackfeed.....	13
2.13 windowsfont.....	13
2.14 about	14
2.15 rfidWrite	15
2.16 rfidRead.....	16
2.17 rfidGetReadData.....	16
2.18 rfidGetReadDatavb6.....	17
2.19 usbportqueryprinter.....	17
2.20 netportqueryprinter	18
3 Sample Code.....	19
3.1 Example (1)	19
3.2 Example (2)	19
4 About	20
4.1 Supported Printers.....	20
4.2 Release Notes	20

1 Guide

1.1 Printing a Label

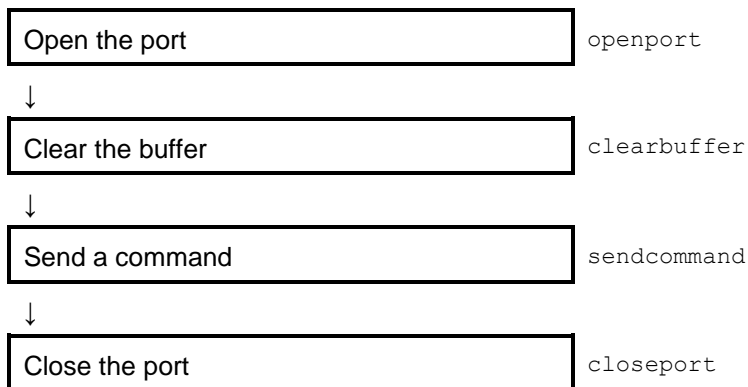
- Example:



1.2 Printing using the Printer Commands

To send printer commands (Example: FBPL, ZPL), use the `sendcommand` functions.

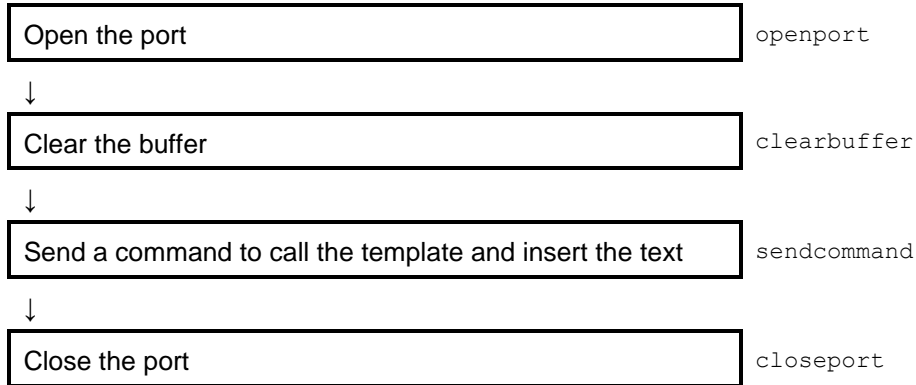
- Example:



1.3 Printing using an FBPL Template

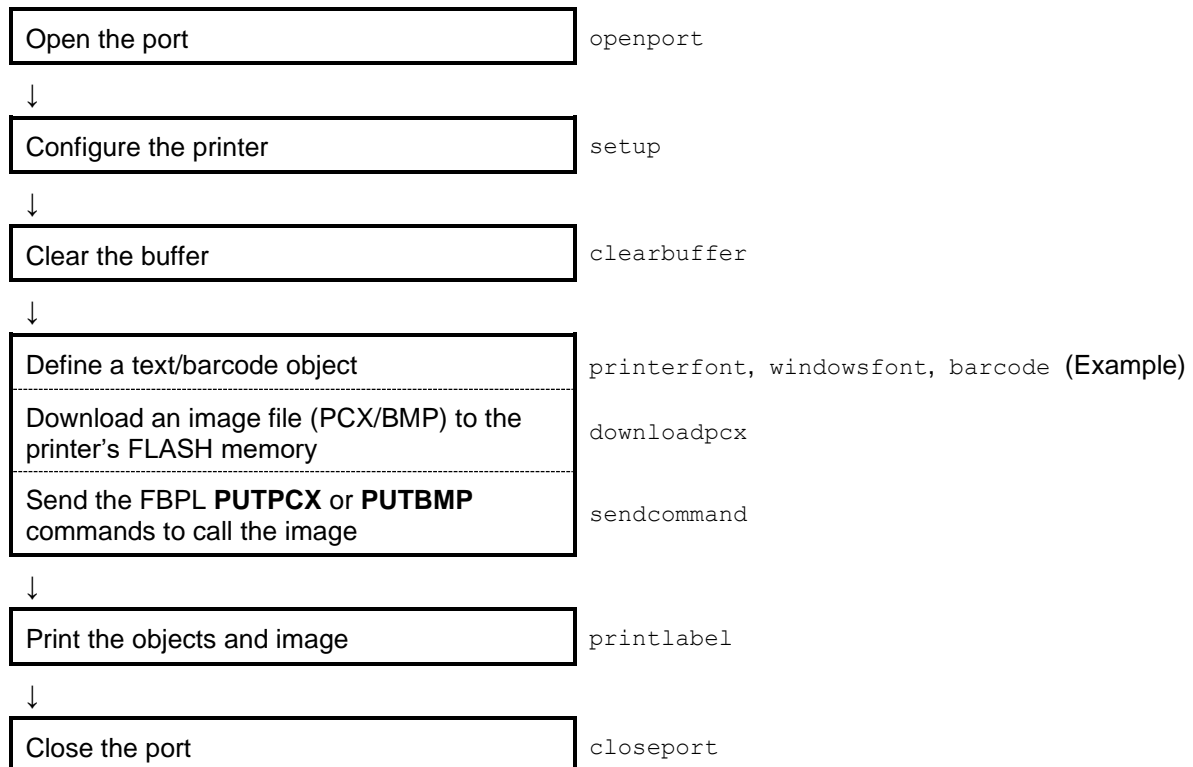
1. Create a template using FBPL commands (.BAS).
2. Transfer the template to the printer's FLASH memory using the BPM (Brother Printer Management Tool).
3. Call the template and insert the text you want using the `sendcommand` functions.

- Example:



1.4 Printing an Image

To recall and print a PCX/BMP image (Example: logo), download the file to the printer's FLASH memory, using the `downloadpcx` functions. Then call the image by sending the FBPL commands. For more information about the FBPL commands, see the **PUTPCX** or **PUTBMP** command in the *FBPL Command Reference*.



- For bitmap printing, download either a 1-bit (monochrome) or 8-bit (256-color) BMP graphic file only.
- Print the downloaded image in monochrome or grayscale using the **PUTPCX** or **PUTBMP** command. The direct thermal mode supports both monochrome printing and grayscale printing. The thermal transfer mode supports monochrome printing only.

1.5 Encoding an RFID Tag

To encode RFID tags, use the `rfidWrite` function.

- Example:



- Supported Printers:

- TD-4650TNWBR
- TD-4750TNWBR
- TJ-4021TNR
- TJ-4121TNR

2 API

2.1 openport

Start the Windows printer spool.

Declaration

```
int openport (String port);
```

Parameters

Parameter	Type	Required	Description
port	string	Yes	1) Printer driver name for local printer (Example: Brother TD-4420TN)
			2) UNC path and printer name for a network printer (Example: "\\server\TD-4420TN")
			3) LPT1 to LPT4 for Centronics Interface (Example: "LPT1")
			4) USB interface (Example: "USB")

Return Value

integer; 1 if success, otherwise 0.

2.2 opennetport

Start the Windows printer spool by designating the IP address.

Declaration

```
int opennetport (String IP address, int printer port);
```

Parameter

Parameter	Type	Required	Description
IP address	string	Yes	Printer IP address (Example: "192.168.0.1")
Printer port	integer	Yes	Printer connection port (Example : 9100)

Returned Value

integer; 1 if success, otherwise 0

2.3 closeport

Close the Windows printer spool.

Declaration

```
int closeport ();
```

Parameters

None

Return Value

integer; 1 if success, otherwise 0.

2.4 setup

Configure the printer settings.

Declaration

```
int setup (String width, String height, String speed, String density, String sensorType, String gap, String distance);
```

Parameters

Parameter	Type	Required	Description
width	string	Yes	Label width (mm)
height	string	Yes	Label height (mm)
speed	string	Yes	Print speed (inch/second) 1 / 1.5 / 2 / 3 / 4 / 6 / 8 / 10 / 12 / 13 / 14 For more information about the available print speed settings, see the SPEED command in the <i>FBPL command reference</i> .
density	string	Yes	Print density 0-15: 0 Lightest, 15 Darkest
sensorType	string	Yes	Media sensor type 0: Gap sensor, 1: Black mark sensor
gap	string	Yes	Gap distance between two labels or the height of the black mark (mm)
distance	string	Yes	Shift the distance of the gap/black mark (mm); Set this parameter to 0 when using the normal label type.

Return Value

integer; 1 if success, otherwise 0.



To set the paper type to continuous paper, specify 0 for the parameters `sensorType`, `gap`, and `distance`.

2.5 clearbuffer

Clear the buffer.

Declaration

```
int clearbuffer();
```

Parameters

None

Return Value

integer; 1 if success, otherwise 0.

2.6 barcode

Use built-in barcode formats.

Declaration

```
int barcode(String xDir, String yDir, String barcode, String height,  
String interpretation, String rotation, String narrowRatio, String  
wideRatio, String content);
```

Parameters

Parameter	Type	Required	Description
xDir	string	Yes	Starting point of the barcode in the X direction (dots) (203 dpi: 1 mm=8 dots, 300 dpi: 1 mm=12 dots, 600 dpi: 1 mm=24 dots)
yDir	string	Yes	Starting point of the barcode in the Y direction (dots) (203 dpi: 1 mm=8 dots, 300 dpi: 1 mm=12 dots, 600 dpi: 1 mm=24 dots)
barcode	string	Yes	Barcode type For more information, see the Barcode Type section.
height	string	Yes	Barcode height (dots)
interpretation	string	Yes	Human recognizable interpretation (text) 0: Not readable, 1: Human readable
rotation	string	Yes	Rotation degrees 0 : No rotation 90 : Rotate 90 degrees clockwise 180 : Rotate 180 degrees clockwise 270 : Rotate 270 degrees clockwise
narrowRatio	string	Yes	Narrow bar ratio For more information about the ratio, see the BARCODE command in the <i>FBPL Command Reference</i> .
wideRatio	string	Yes	Wide bar ratio For more information about the ratio, see the BARCODE command in the <i>FBPL Command Reference</i> .
content	string	Yes	Barcode content

Return Value

integer; 1 if success, otherwise 0.

Barcode Type

Select the `barcode` type from the list.

Barcode Type	Description
128	Code 128, switching code subset automatically
128M	Code 128, switching code subset manually
EAN128	EAN-128, switching code subset automatically
EAN128M	EAN-128M (GS1-128), switching code subset manually
25	Interleaved 2-of-5
25C	Interleaved 2-of-5 with check digit
25S	Standard 2-of-5
25I	Industrial 2-of-5
39	Code 39, switching standard and full ASCII mode automatically
39C	Code 39 with check digit
93	Code 93
EAN13	EAN-13
EAN13+2	EAN-13 with 2 digits add-on
EAN13+5	EAN-13 with 5 digits add-on
EAN8	EAN-8
EAN8+2	EAN-8 with 2 digits add-on
EAN8+5	EAN-8 with 5 digits add-on
CODA	Codabar (NW-7)
POST	POSTNET

Barcode Type	Description
UPCA	UPC-A
UPCA+2	UPC-A with 2 digits add-on
UPA+5	UPC-A with 5 digits add-on
UPCE	UPC-E
UPCE+2	UPC-E with 2 digits add-on
UPE+5	UPC-E with 5 digits add-on
MSI	MSI
MSIC	MSI with check digit
PLESSEY	Plessey
CPOST	Datalogic 2-of-5 (China post)
ITF14	ITF-14
EAN14	EAN-14
11	Code 11 (USD-8)
TELEPEN	Telepen
TELEPENN	Telepen number
PLANET	Planet
CODE49	Code 49
DPI	Deutsche Post Identcode
DPL	Deutsche Post Leitcode
LOGMARS	A special use of Code 39



To print other barcodes, send FBPL commands using the `sendcommand` function.

Example: `sendcommand("QRCODE 10,10,H,4,A,0,\"ABCabc123\"\\r\\n")`

2.7 printerfont

Define a text object in built-in fonts.

Declaration

```
int printerfont(String xDir, String yDir, String fontType, String rotation,  
String xRate, String yRate, String content);
```

Parameters

Parameter	Type	Required	Description
xDir	string	Yes	Starting point of text (character string) in the X direction (dots) (203 dpi: 1 mm=8 dots, 300 dpi: 1 mm=12 dots, 600 dpi: 1 mm=24 dots)
yDir	string	Yes	Starting point of text (character string) in the Y direction (dots) (203 dpi: 1 mm=8 dots, 300 dpi: 1 mm=12 dots, 600 dpi: 1 mm=24 dots)
fontType	string	Yes	Font type name Make sure the file name is in uppercase letters (Example: FILENAME.TTF).
rotation	string	Yes	Rotation degree of the text 0 : No rotation 90 : Rotate 90 degrees clockwise 180 : Rotate 180 degrees clockwise 270 : Rotate 270 degrees clockwise
xRate	string	Yes	Character height magnification (Range: 1-8)
yRate	string	Yes	Character width magnification (Range: 1-8)
content	string	Yes	Characters to be printed

Return Value

integer; 1 if success, otherwise 0.



To use the built-in fonts, select the fontType from the list.

Font Name	Description
0	Monotype CG Triumvirate Bold Condensed scalable font
1	8 x 12 fixed pitch dot font
2	12 x 20 fixed pitch dot font
3	16 x 24 fixed pitch dot font
4	24 x 32 fixed pitch dot font
5	32 x 48 dot fixed pitch font
6	14 x 19 dot fixed pitch font OCR-B
7	21 x 27 dot fixed pitch font OCR-B
8	14 x25 dot fixed pitch font OCR-A
ROMAN.TTF	Monotype CG Triumvirate Bold Condensed, fixed font width and height proportion
1.EFT	EPL2 font 1
2.EFT	EPL2 font 2
3.EFT	EPL2 font 3
4.EFT	EPL2 font 4
5.EFT	EPL2 font 5
A.FNT	ZPL2 font A
B.FNT	ZPL2 font B
D.FNT	ZPL2 font D
E8.FNT	ZPL2 font E8
F.FNT	ZPL2 font F

G . FNT	ZPL2 font G
H8 . FNT	ZPL2 font H8
GS . FNT	ZPL2 font GS



- To use the downloaded fonts, download your fonts to the printer using the Brother Printer Management Tool (BPM):
 - Run the BPM.
 - Click the **File Manager**.
 - Select the file type from the drop-down list, and then click **Browse** to select the font file.
 - Select **FLASH** from the **Memory Device** drop-down list.
 - Click **Download** to start downloading the file.

Make sure the file name is in uppercase letters. (Example: ~~ipaexg.ttf~~ → IPAEXG.TTF)

- The BPM is a Windows tool and it does not support the Bluetooth® connection. When using the BPM, connect the printer to your computer via USB or Wi-Fi®.



If the printer's encoding setting and the print data encoding do not match, print quality errors, such as garbled characters or missing text, will occur. If the font is not compatible with the encoding, change the encoding or the selected font. To change the printer's encoding setting, see the **CODEPAGE** command in the *FBPL Command Reference*.

2.8 sendcommand

Send a command to the printer.

Declaration

```
int sendcommand(String command);
```

Parameters

Parameter	Type	Required	Description
command	string	Yes	Built-in commands (Example: FBPL commands) For more information about FBPL commands, see the <i>FBPL Command Reference</i> .

Example: `sendcommand("PRINT 1\r\n")` `sendcommand("CLS\r\nPRINT\r\n")`

Return Value

integer; 1 if success, otherwise 0.



You can send non-FBPL commands that your printer's firmware supports. When sending non-FBPL commands with `sendcommand`, use this API only with `openport` and `closeport`.

2.9 printlabel

Print a label.

Declaration

```
int printlabel(String set, String copy);
```

Parameters

Parameter	Type	Required	Description
set	string	Yes	Number of label sets
copy	string	Yes	Number of label copies

Return Value

integer; 1 if success, otherwise 0.

2.10 downloadpcx

Download a monochrome PCX/BMP graphic file to the printer.

Declaration

```
int downloadpcx(String fileName, String name);
```

Parameters

Parameter	Type	Required	Description
fileName	string	Yes	File name (including file retrieval path)
name	string	Yes	File name (the name of the file downloaded to FLASH) Make sure the file name is in uppercase letters (FILENAME.PCX, FILENAME.BMP)

Return Value

integer; 1 if success, otherwise 0.



For more information, see [1.4 Printing an Image](#).

2.11 formfeed

Skip to the next page of a label. Make sure you set this function after calling the `setup` function.

Declaration

```
int formfeed();
```

Parameters

None

Return Value

integer; 1 if success, otherwise 0.

2.12 nobackfeed

Disable the backfeed function.

Declaration

```
int nobackfeed();
```

Parameters

None

Return Value

integer; 1 if success, otherwise 0.

2.13 windowsfont

Define a text object using Windows fonts.

Declaration

```
int windowsfont(int xDir, int yDir, int height, int rotation, int  
fontStyle, int underline, String fileType, String text);
```

Parameters

Parameter	Type	Required	Description
xDir	integer	Yes	Starting point of text (character string) in the X direction (dots) (203 dpi: 1 mm=8 dots, 300 dpi: 1 mm=12 dots, 600 dpi: 1 mm=24 dots)
yDir	integer	Yes	Starting point of text (character string) in the Y direction (dots) (203 dpi: 1 mm=8 dots, 300 dpi: 1 mm=12 dots, 600 dpi: 1 mm=24 dots)
height	integer	Yes	Font height (dots)
rotation	integer	Yes	Rotation degree of the text 0 : No rotation 90 : Rotate 90 degrees clockwise 180 : Rotate 180 degrees clockwise 270 : Rotate 270 degrees clockwise
fontStyle	integer	Yes	Font style 0 : Normal 1 : Italic 2 : Bold 3 : Bold and Italic
underline	integer	Yes	Font with underline 0 : without underline 1 : with underline
fileType	string	Yes	Font name
text	string	Yes	Characters to be printed

Return Value

integer; 1 if success, otherwise 0.

2.14 about

Display the DLL version on the screen.

Declaration

```
void about();
```

Parameters

None

Return Value

integer; 1 if success

2.15 rfidWrite

Write data to the RFID tag.

Declaration

```
int rfidWrite(String lock, String format, String blockNumber, String dataSize, String memoryBank, String data);
```

Parameters

Parameter	Type	Required	Description
lock	string	Yes	Optional parameter to lock the data to prevent it from being overwritten 0 : Write without locking 1 to FFFFFFFF in hex : Write and lock the data block
format	string	Yes	Format of the field data A = ASCII H = Hex
blockNumber	string	Yes	Specifies 0
dataSize	string	Yes	Write data size from 1 to n in decimal numbers
memoryBank	string	Yes	Memory bank EPC – EPC area (Up to 496 bits) USR – User area (Up to 8 kilobits) ACS – Access code area KIL – Kill code area PC – PC code area
data	string	Yes	Contents of data string

Return Value

integer; 1 if success, otherwise 0.



- Supported Printers: TD-4650TNWBR, TD-4750TNWBR, TJ-4021TNR, and TJ-4121TNR.
- Make sure you use the correct firmware version to set the PC bits.

Model	Firmware Version
TD-4650TNWBR / TD-4750TNWBR	V1.04.S21 or later
TJ-4021TNR / TJ-4121TNR	B2.12.S27 or later

2.16 rfidRead

Read data from the RFID tag and return the data after the `printlabel` function is called.

Declaration

```
int rfidRead(String unlock, String format, String blockNumber, String dataSize, String memoryBank);
```

Parameters

Parameter	Type	Required	Description
unlock	string	Yes	Optional parameter to unlock the data so it can be overwritten later 0 : Read without unlocking 1 to FFFFFFFF in hex : Read and unlock the data block
format	string	Yes	Format of the field data A = ASCII H = Hex
blockNumber	string	Yes	Specifies 0
dataSize	string	Yes	Read data size from 1 to n in decimal numbers
memoryBank	string	Yes	Memory bank EPC – EPC area (Up to 496 bits) TID – Tag identification area USR – User area (Up to 8 kilobits) ACS – Access code area KIL – Kill code area PC – PC code area

Return Value

integer; 1 if success, otherwise 0.



Supported Printers: TD-4650TNWBR, TD-4750TNWBR, TJ-4021TNR, and TJ-4121TNR.

2.17 rfidGetReadData

Return the received tag data specified by the `rfidRead` function.

Make sure you set this command immediately after calling the `printlabel` function.

Declaration

```
string rfidGetReadData();
```

Parameters

None

Return Value

string; tag data.



Supported Printers: TD-4650TNWBR, TD-4750TNWBR, TJ-4021TNR, and TJ-4121TNR.

2.18 rfidGetReadDatavb6

For VBA: Return the received tag data specified by the `rfidRead` function.

Make sure you set this command immediately after calling the `printlabel` function.

Declaration

```
string rfidGetReadDatavb6();
```

Parameters

None

Return Value

string; tag data.



Supported Printers: TD-4650TNWBR, TD-4750TNWBR, TJ-4021TNR, and TJ-4121TNR.

2.19 usbportqueryprinter

Return the printer status (USB port only).

For more information, see the `<ESC>!?` command in the *FBPL Command Reference*.

Declaration

```
string usbportqueryprinter();
```

Parameters

None

Return Value

string; -1 if failure, otherwise a status code is returned.

Status Code (HEX)	Printer Status
00	Normal
01	Head opened
02	Paper Jam
03	Paper Jam and head opened
04	Out of paper
05	Out of paper and head opened
08	Out of ribbon
09	Out of ribbon and head opened
0A	Out of ribbon and paper jam
0B	Out of ribbon, paper jam and head opened
0C	Out of ribbon and out of paper
0D	Out of ribbon, out of paper and head opened
10	Pause
20	Printing
80	Other error

2.20 netportqueryprinter

Return the printer status (wired LAN or wireless LAN only)

For more information, see the <ESC>!? command in the *FBPL Command Reference*.

Declaration

```
string netportqueryprinter();
```

Parameters

None

Return Value

string; -1 if failure, otherwise a status code is returned.

Status Code (HEX)	Printer Status
00	Normal
01	Head opened
02	Paper Jam
03	Paper Jam and head opened
04	Out of paper
05	Out of paper and head opened
08	Out of ribbon
09	Out of ribbon and head opened
0A	Out of ribbon and paper jam
0B	Out of ribbon, paper jam and head opened
0C	Out of ribbon and out of paper
0D	Out of ribbon, out of paper and head opened
10	Pause
20	Printing
80	Other error

3 Sample Code

3.1 Example (1)

```
BROLIB_DLL.openport("Brother TD-4750TN");
BROLIB_DLL.setup("100", "63.5", "4", "8", "0", "0", "0");
BROLIB_DLL.clearbuffer();
BROLIB_DLL.barcode("100", "100", "128", "100", "1", "0", "2", "2", "Barcode Test");
BROLIB_DLL.printerfont("100", "250", "3", "0", "1", "1", "Print Font Test");
BROLIB_DLL.windowfont(100, 300, 24, 0, 0, 0, "ARIAL", "Windows Arial Font Test");
BROLIB_DLL.downloadpcx("UL.PCX", "UL.PCX");
BROLIB_DLL.sendcommand("PUTPCX 100,400,\"UL.PCX\"");
BROLIB_DLL.printlabel("1", "1");
BROLIB_DLL.closeport();
```

3.2 Example (2)

```
BROLIB_DLL.openport("Brother TD-4750TN");
BROLIB_DLL.clearbuffer();
BROLIB_DLL.rfidWrite("0", "H", "0", "12", "EPC", "123456789012");
BROLIB_DLL.rfidRead("0", "H", "0", "12", "EPC");
BROLIB_DLL.printlabel("1", "1");
label2.Text = BROLIB_DLL.rfidGetReadData();
BROLIB_DLL.closeport();
```

4 About

4.1 Supported Printers

> [List of models supporting the Brother Windows DLL](#)

4.2 Release Notes

> [Release Notes for the Brother Windows DLL](#)

brother