

# Printer Browser Programming manual

NOV 1, 2023 Rev1.0.0

This manual provides information on the design guidelines concerning Printer Browser that customers need to build Web applications.



History of revision of this manual

Revision	Date	APP ver	Description of revision
Rev1.0.0	November 1, 2023	1.00	Release of 1st edition

## Caution

- Printer Browser is a copyrighted work of Sanei Electric Co., Ltd. (hereinafter referred to as Sanei Electric). Copyright and other rights concerning this product belong to Sanei Electric.
- Sanei Electric does not guarantee the absence of defects in the Printer Browser and is not liable for any damages resulting from the use of the information contained in this manual.
- Sanei Electric shall never be liable under any circumstances for any direct or indirect loss or damage in connection with the use of Printer Browser.

Sanei Electric Co., Ltd.

Unauthorized reproduction prohibited.

The contents of this document are subject to change without notice.

Android is a trademark of Google Inc.

# Table of Contents

<b>1. INTRODUCTION .....</b>	<b>3</b>
<b>2. OPERATING ENVIRONMENT .....</b>	<b>3</b>
<b>3. SUPPORTED PRINTERS .....</b>	<b>3</b>
<b>4. ABOUT API .....</b>	<b>4</b>
<b>5. API LIST .....</b>	<b>5</b>
5-1. discoverUsbPrinter.....	7
5-2. discoverBlePrinter.....	7
5-3. discoverBthPrinter.....	8
5-4. connectUsbPrinter .....	8
5-5. connectBlePrinter.....	9
5-6. connectBthPrinter .....	9
5-7. connectNetworkPrinter .....	10
5-8. disconnectPrinter .....	10
5-9. isPrinterConnected .....	10
5-10. printString.....	10
5-11. printString (Encoding specified) .....	11
5-12. setFontStyle .....	11
5-13. setFontStyle (Underline thickness specified) .....	11
5-14. setFontType .....	11
5-15. setFontMagnification.....	12
5-16. setFontColor .....	12
5-17. setFontSmoothing.....	12
5-16. setAlignment .....	12
5-17. setCodePage .....	13
5-18. setInternationalChar .....	14
5-19. printBarcode.....	14
5-20. printQR.....	15
5-21. printPDF417.....	15
5-22. setBarcodeModification .....	16
5-23. printBitmap.....	16
5-24. printBitmap (Print Position and dithering designation).....	16
5-25. linefeed .....	16
5-26. paperFeed.....	17
5-27. printStringAndroidFont.....	17
5-28. initPrinter.....	17

5-29. startPageMode..... 17  
5-30. endPageMode..... 18  
5-31. setPageAttribute ..... 18  
5-32. getStatus ..... 19  
5-33. outputRawDataByte ..... 20  
5-34. outputRawDataBase64 ..... 20  
5-35. inputRawData ..... 20  
5-36. inputRawData ..... 20

## 1. Introduction

The Printer Browser is a browser app supportable to print directly from a web program (Javascript) to our printer.

To prepare for using this browser app, you can download the sample program in the following URL.

<https://www.sanei-elec.co.jp/wp/wp-content/uploads/2021/02/sample.zip>

This is a simple print sample using Javascript.

This manual describes the command syntax for Javascript required for printing, referring to this sample program.

## 2. Operating Environment

Runtime version: 1.8.0 or later

Android OS: Ver 7.1.1 or later

## 3. Supported Printers

Supported printer models		F/W version	Compatible interface	Supported command
KIOSK printers	SK4-21	V1.02.00 or later	USB	MODE-A
	SK4-31	V1.02.00 or later	USB	MODE-A
	SK1-2x1	V2.60.00 or later	USB	MODE-A
	SK1-3x1	V2.60.00 or later	USB	MODE-A
	SK1-21H	V2.60.00 or later	USB	MODE-A
	SK1-31H	V2.60.00 or later	USB	MODE-A
Desktop printers	SD3-22	V1.05.00 or later	USB	MODE-A
	SD3-21	V1.05.00 or later	USB	MODE-A
Mobile printer	SM4-21	V1.01.00 or later	USB, BLE, WLAN	MODE-A
	SM4-31	V1.01.00 or later	USB, BLE, WLAN	MODE-A

### ★ Memory switch settings

Set the memory switch settings for all printer models as follows.

OFFLINE BUSY = OFF

Act. For Driver = INVALID

## 4. About API

### (1) type of data

The data handled by the Javascript API is number, strings, Boolean, and object.

Character strings that are difficult to handle depending on the type of API are converted to JSON format or Base64 and used.

### (2) Calling API

The API can be called easily by adding "Printer." at the top of the call.

No external library is needed to build this extension, and no preparation is required to call the API.

### (3) Get Return Values

The return value can be obtained directly from the API.

The return value is mainly used after conversion because it is returned in JSON format or Base64.

The Boolean can be used without conversion.

## 5. API List

The list of printer device functions is as follows.

Category	API	Function
Connect Disconnect	<b>discoverUsbPrinter</b>	Get the name of the USB printer port that can be connected.
	<b>discoverBlePrinter</b>	Get the name of the Bluetooth (BLE) printer port that can be connected.
	<b>discoverBthPrinter</b>	Get the name of the Bluetooth (SPP) printer port that can be connected.
	<b>connectUsbPrinter</b>	Connect to the specified USB printer port.
	<b>connectBlePrinter</b>	Connect to the specified Bluetooth (BLE) printer port.
	<b>connectBthePrinter</b>	Connect to the specified Bluetooth (SPP) printer port.
	<b>connectNetworkPrinter</b>	Connect to the specified network printer port.
	<b>disconnectPrinter</b>	Disconnect the connected printer port.
	<b>isPrinterConnected</b>	Check the printer port connection.
Standard printing	<b>printString</b>	Specifies the character string and prints out the printer font.
	<b>setFontStyle</b>	Sets character modifiers for the printer fonts.
	<b>setFontType</b>	Sets the printer font type (font A or font B).
	<b>setFontMagnification</b>	Sets the printer font size (vertical multiple and horizontal multiple).
	<b>setFontColor</b>	Specifies or cancels white/black inversion for printer fonts.
	<b>setFontSmoothing</b>	Specifies or cancels smoothing of printer fonts.
	<b>setAlignment</b>	Specifies the print position for print data (left alignment, centering, and right alignment).
	<b>setCodePage</b>	Specifies the code page for the printer font.
	<b>setInternationalChar</b>	Specifies the international characters for the printer font.
	<b>printBarcode</b>	Prints out the 1D barcode.
	<b>printQR</b>	Prints out the QR code.
	<b>printPDF417</b>	Prints out PDF417.
	<b>setBarcodeModification</b>	Sets the 1D bar code modification information (height, HRI character).
	<b>printBitmap</b>	Prints out the Bitmap.
	<b>lineFeed</b>	Inserts the specified number of carriage return (line feeds).
	<b>paperFeed</b>	Executes paper feeding the specified number of dot lines.
	<b>printStringAndroidFont</b>	Converts the specified character string into Android font and prints in bit image format.
<b>initPrinter</b>	Initializes the modification information set in the printer.	
Page printing	<b>startPageMode</b>	Shifts from standard mode into page mode.
	<b>endPageMode</b>	Prints the page memory and shifts from page mode to standard mode.
	<b>setPageAttribute</b>	Sets the page mode attribution information (printing direction and printing area).
Status	<b>getStatus</b>	Gets the printer status.



Binary	<b>outputRawDataByte</b>	Sends binary data (Byte) to the printer port.
	<b>outputRawDataBase64</b>	Sends binary data (Base64) to the printer port.
	<b>inputRawdata</b>	Gets binary data from the printer port.

## 5-1. discoverUsbPrinter

Get the name of the USB printer port that can be connected.

function : Printer.discoverUsbPrinter(timeoutMillis)  
Argument: timeoutMillis Timeout period (milliseconds)  
Return value: USB printer port name in JSON format (array)  
Remarks: Return the obtained USB printer port name to its original form with "JSON.parse".

Example:

```
// Get USB Printer Port Name
var result = Printer.discoverUsbPrinter(0);
// Converts the USB printer port name in JSON format to its original format.
var printers = JSON.parse(result);
// Get the first USB printer port name.
var first_printer = printers[0];
```

## 5-2. discoverBlePrinter

Get the name of the Bluetooth (BLE) printer port that can be connected.

function : Printer.discoverBlePrinter(timeoutMillis)  
Argument: timeoutMillis Timeout period (milliseconds)  
Return value: Bluetooth (BLE) printer port name in JSON format (array)  
Remarks: Return the obtained Bluetooth (BLE) printer port name to its original form with "JSON.parse".

Example:

```
// Get Bluetooth (BLE) Printer Port Name
var result = Printer.discoverBlePrinter(0);
// Converts the Bluetooth (BLE) printer port name in JSON format to its original format.
var printers = JSON.parse(result);
// Get the first Bluetooth (BLE) printer port name.
var first_printer = printers[0];
```

### 5-3. discoverBthPrinter

Get the name of the Bluetooth (SPP) printer port that can be connected.

function : Printer.discoverBthPrinter(timeoutMillis)  
Argument: timeoutMillis      Timeout period (milliseconds)  
Return value: Bluetooth(SPP) printer port name in JSON format (array)  
Remarks: Return the obtained Bluetooth (SPP) printer port name to its original form with "JSON.parse".

Example:

```
// Get Bluetooth (SPP) Printer Port Name
var result = Printer.discoverBthPrinter(0);
// Converts the Bluetooth (SPP) printer port name in JSON format to its original format.
var printers = JSON.parse(result);
// Get the first Bluetooth (SPP) printer port name.
var first_printer = printers[0];
```

### 5-4. connectUsbPrinter

Connect to the specified USB printer port.

function : Printer.connectUsbPrinter(printer)  
Argument: printer      USB printer port name in JSON format.  
Return value: Returns the success or failure of the connection as a Boolean.  
          true      Successful connection  
          false      Connection failed  
Remarks: Use "JSON. stringify" to convert the USB printer port name to JSON format.

Example:

```
// Converts USB printer port names to JSON format.
var jprinter = JSON.stringify(printer);
// Connect to USB printer port
var status = Printer.connectUsbPrinter(jprinter)
```

## 5-5. connectBlePrinter

Connect to the specified Bluetooth (BLE) printer port.

function: Printer.connectBlePrinter(printer)

Argument: printer Bluetooth (BLE) printer port name in JSON format.

Return value: Returns the success or failure of the connection as a Boolean.

true Successful connection

false Connection failed

Remarks: Use "JSON.stringify" to convert the Bluetooth (BLE) printer port name to JSON format.

Example:

```
// Converts Bluetooth (BLE) printer port names to JSON format.  
var jprinter = JSON.stringify(printer);  
// Connect to Bluetooth (BLE) USB printer port  
var status = Printer.connectBlePrinter(jprinter)
```

## 5-6. connectBthPrinter

Connect to the specified Bluetooth (SPP) printer port.

function: Printer.connectBthPrinter(printer)

Argument: printer Bluetooth (SPP) printer port name in JSON format.

Return value: Returns the success or failure of the connection as a Boolean.

true Successful connection

false Connection failed

Remarks: Use "JSON.stringify" to convert the Bluetooth (SPP) printer port name to JSON format.

Example:

```
// Converts Bluetooth (SPP) printer port names to JSON format.  
var jprinter = JSON.stringify(printer);  
// Connect to Bluetooth (SPP) printer port  
var status = Printer.connectBthPrinter(jprinter)
```

### 5-7. connectNetworkPrinter

Connect to the specified network printer port.

function : Printer.connectNetworkPrinter (address,port)  
Argument: address IP address (string)  
port Port number (number)  
Return value: Returns the success or failure of the connection as a Boolean.  
true Successful connection  
false Connection failed

### 5-8. disconnectPrinter

Disconnect the connected printer port.

function : Printer.disconnectPrinter()  
Argument: None  
Return value: None

### 5-9. isPrinterConnected

Check the printer port connection.

function : Printer.isPrinterConnected()  
Argument: None  
Return value: Returns the current connection state as a Boolean  
true Connected  
false Not connected

### 5-10. printString

Specifies the character string and prints out the printer font.

function : Printer.printString(text)  
Argument: text Specify the character string to be printed in Unicode.  
Return value: None

### 5-11. printString (Encoding specified)

Prints a character string with encoding specified in the printer font.

function : Printer.printString(text,charname)

Argument: text           Character string  
          charname     String encoding name

Return value: None

Remarks: Charname specifies the encoding name to use in Java.  
          Example. Specify an encoding name such as "UTF-8", "SJIS", etc. as a string.

### 5-12. setFontStyle

Sets character modifiers for the printer fonts.

function : Printer.setFontStyle(bold,italic,underline)

Argument: bold           Specification (true) or cancellation (false) of bold typeface  
          italic         Specification (true) or cancellation (false) of italic typeface  
          underline     Specification (true) or cancellation (false) of underlining

Return value: None

### 5-13. setFontStyle (Underline thickness specified)

Sets character modifiers for the printer fonts. (Underline thickness specified)

function : Printer.setFontStyle(bold,italic,underline,dotpitch)

Argument: bold           Specification (true) or cancellation (false) of bold typeface  
          italic         Specification (true) or cancellation (false) of italic typeface  
          underline     Specification (true) or cancellation (false) of underlining  
          dotpitch      Underline thickness specifies 0 to 7 dots.

Return value: None

### 5-14. setFontType

Sets the printer font type.

function : Printer.setFontType(compact)

Argument: compact     Specification (true) or cancellation (false) of small font (Font B).

Return value: None

### 5-15. setFontMagnification

Sets the printer font size (vertical multiple and horizontal multiple).

function: Printer.setFontMagnification (horizontalRatio, verticalRatio)  
Argument: horizontalRatio Specification of the horizontal multiple between 1 to 8  
            verticalRatio Specification of the vertical multiple between 1 to 8.  
Return value: None

### 5-16. setFontColor

Specifies or cancels white/black inversion for printer fonts.

function: Printer.setFontColor(reverse)  
Argument: reverse Specification (true) or cancellation (false) of white/black inversion  
Return value: None

### 5-17. setFontSmoothing

Specifies or cancels smoothing of printer fonts.

function: Printer.setFontSmoothing(smoothing)  
Argument: smoothing Specification (true) or cancellation (false) of smoothing  
Return value: None

### 5-16. setAlignment

Specifies the print position for print data (left alignment, centering, and right alignment).

function: Printer.setAlignment(alignment)  
Argument: alignment

alignment	
Left alignment	0
Centering	1
Right alignment	2

Return value: None

## 5-17. setCodePage

Specifies the code page for the printer font.

function : Printer.setCodePage(codePage)

Argument: codepage

CodePage	
PC437	0
katakana	1
PC850	2
PC852	3
PC857	4
PC858	5
PC863	6
PC865	7
PC866	8
WPC1252	9
PC860	10
WPC1252_2	11
PC862	12
WPC1254	13
WPC1250	14
WPC1251	15
PC864	16
PC737	18
PC1253	20

Return value: None



### 5-18. setInternationalChar

Specifies the international character for the printer font.

function : Printer.setInternationalChar(internationalCharset)

Argument: internationalCharset

InternationalCharset	
USA	0
France	1
Germany	2
England	3
Denmark	4
Sweden	5
Italy	6
Spain	7
Japan	8

Return value: None

### 5-19. printBarcode

Prints the 1D barcode.

function : Printer.printBarcode(barcodeSystem,barcodeData)

Argument: barcodeSystem

BarcodeSystem	
UPCA	65
UPCE	66
JAN13	67
JAN8	68
CODE39	69
ITF	70
CODABAR	71
CODE93	72
CODE128	73

barcodeData                      Barcode data

Return value: None

## 5-20. printQR

Prints the QR code.

function : Printer.printQR(size,eccLevel,barcodeData)  
Argument: size Specify the symbol size (1 to 40).  
eccLevel Specify the error control level (1 to 4).  
1: L (7%)  
2: M (15%)  
3: Q (25%)  
4: H (30%)  
barcodeData Barcode data  
Return value: None

## 5-21. printPDF417

Prints the PDF417.

function : Printer.printPDF417(truncate,binaryEncode,eccLevel,size,barcodeData)  
Argument: truncate Specify (true) of not specify (false) as (compact) PDF417  
binaryEncode Byte encoding mode (true) or automatic encoding mode (false)  
eccLevel The error correction level (0 to 7) is specified.  
size The barcode size is specified as shown in the combination table.

size	Details (X: Row / Y = Step)	size	Details (X: Row / Y = Step)
0	X 2: Y 4	8	X 12: Y 4
1	X 2: Y 9	9	X 12: Y 9
2	X 2: Y 15	10	X 12: Y 15
3	X 2: Y 20	11	X 12: Y 20
4	X 7: Y 4	12	X 20: Y 4
5	X 7: Y 9	13	X 20: Y 9
6	X 7: Y 15	14	X 20: Y 15
7	X 7: Y 20	15	X 20: Y 20

barcodeData Barcode data  
Return value: None

## 5-22. setBarcodeModification

Sets the 1D barcode modification information (height, HRI characters).

function : Printer.setBarcodeModification (hri,width,height)

Argument: hri HRI character modification.

hri	
HRI characters are not printed.	0
HRI characters are printed above the barcode.	1
HRI characters are printed below the barcode.	2
HRI characters are printed above and below the barcode.	3

Width The barcode module width (1 to 4) is specified.

Height The barcode height (1 to 255 dot pitch) is specified.

Return value: None

## 5-23. printBitmap

Converts the bitmap data specified by Base64 character string to bit image command and prints it.

function : Printer.printBitmap(data)

Argument: data Specify the bitmap data (Base64 string).

Return value: None

## 5-24. printBitmap (Print Position and dithering designation)

Converts the bitmap data specified by Base64 character string to bit image command and prints it.

function : Printer.printBitmap(data,align,dither)

Argument: data Specify the bitmap data (Base64 string).

align Specify the print position (Left, Center, Right).

dither Specify dithering (true) or not (false)

Return value: None

## 5-25. linefeed

Specified number of carriage return (line feeds) is executed.

function : Printer.lineFeed(lines)

Argument: lines Number of carriage returns (line feeds) is specified (1 to 255).

Return value: None

## 5-26. paperFeed

Executes paper feeding the specified number of dot lines.

function : Printer.paperFeed(lines)

Argument: lines Paper feed is specified by a dot pitch value of -255 to 255.  
If the parameter is negative, the command executes printing and reverse paper feeding.  
If the parameter is an integer, printing and forward paper feeding are executed.

Return value: None

Remarks: In the standard mode, the data in the line buffer is printed, and the paper is fed forward by the specified dot pitch.  
In page mode, the data is shifted in the y-axis direction by the specified dot pitch.

## 5-27. printStringAndroidFont

Converts the specified character string into Android font and prints in bit image format.

function : Printer.printStringAndroidFont(text, x, y, textSize)

Argument: text Character string is specified.  
x Horizontal print position  
y Vertical print position  
textSize Specify Character Size (Unit: SP).

Return value: None

Remarks: SP = Scale-independent pixel

## 5-28. initPrinter

Modification information set in the printer is initialized.

function : Printer.initPrinter()

Argument: None

Return value: None

## 5-29. startPageMode

Shift from standard mode to page mode.

function : Printer.startPageMode()

Argument: None

Return value: None

### 5-30. endPageMode

The specified area of page memory is printed and the page mode is shifted to standard mode.

function : Printer.endPageMode()

Argument: None

Return value: None

### 5-31. setPageAttribute

Sets the page mode attribution information (printing direction and printing area).

function : Printer.setPageAttribute(x,y,width,height,direction)

Argument: x Starting point of the print area (x axis)

y Starting point of the print area (y axis)

width Horizontal width of the starting point of the print area (x axis)

height Height of the starting point of the print area (y axis)

direction Direction of expansion within the print area

direction	
Forward direction	0
90-degree rotation to the left	1
Reverse direction	2
90-degree rotation to the right	3

Return value: None

Remarks: The print area can be specified any number of times until the page mode is printed. When printing in page mode, the printer automatically prints within the maximum print area at that time.

The specified range of the print area depends on the printer model and memory switch settings.

## 5-32. getStatus

Gets the printer status.

function : Printer.getStatus()

Argument: None

Return value: JSON format Status.

Remarks: The acquired status is restored to its original format by "JSON.parse" and used. Refer to the command reference GS R1 command for the specification of the status value.

Example:

```
//Get the status
var result = Printer.getStatus();
// Converting a Retrieved JSON Status to Original Format
var status = JSON.parse(result);

switch (status.status1) {
case 82://Ascii code 'R'
    buffer+= "Status1: WAITING<br>";
    break;
case 66:// Ascii code 'B'
    buffer+= "Status1: MOVING<br>";
    break;
case 69:// Ascii code 'E'
    buffer+= "Status1: ERROR<br>";
    break;
default:
    buffer+= "Status1: " + status.status1 + "<br>";
}
```

### 5-33. outputRawDataByte

Sends binary data (1-Byte) to the printer port.

function : Printer.outputRawDataByte(byte)  
Argument : byte            1-byte binary data  
Return value : None

### 5-34. outputRawDataBase64

Sends binary data (Base64 strings) to the printer port.

function : Printer.outputRawDataBase64(base64)  
Argument : base64        Base64 strings binary data  
Return value : None

### 5-35. inputRawData

Gets binary data (1-Byte) from the printer port.

function : Printer.inputRawData(timeoutMillis)  
Argument : timeoutMillis    Specify the timeout period (in milliseconds)  
Return value : 1-byte binary data

### 5-36. inputRawData

Gets binary data (Base64 strings) from the printer port.

function : Printer.inputRawData(timeoutMillis,size)  
Argument : timeoutMillis    Specify the timeout period (in milliseconds)  
              Size                Specify the size to receive  
Return value : Base64 strings binary data