

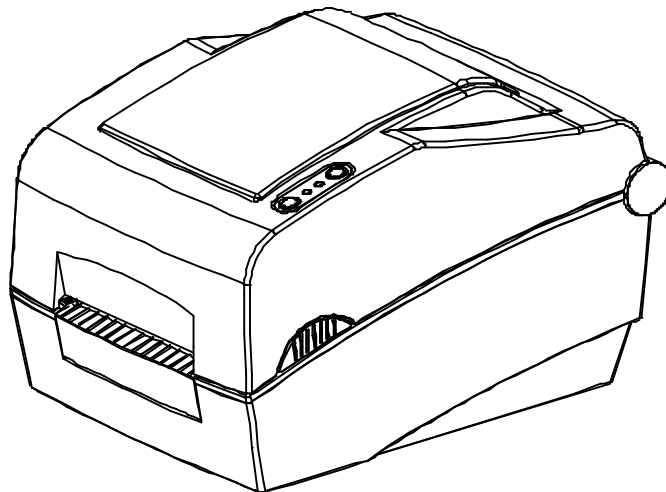
BIXOLON®

Linux SDK 레퍼런스 가이드

Label Printer

Rev. 1.00

**SLP-TX400/SLP-TX403
SLP-TX420/SLP-TX423
SLP-TX220/SLP-TX223
SLP-DX420/SLP-DX423
SLP-DX220/SLP-DX223
SLP-DL410/SLP-DL413
SRP-770II/SRP-770III**



<http://www.bixolon.com>

■ **목차**

1. 매뉴얼 안내	3
1-1 지원 Kernel & 플랫폼.....	3
1-2 지원 인터페이스.....	3
2. Property	4
2-1 CharacterSet & International CharacterSet(LONG R/W).....	4
2-2 State (LONG R).....	6
3. Method	7
3-1 ConnectToPrinter.....	7
3-2 DisconnectPrinter.....	8
3-3 InitializePrinter.....	9
3-4 FeedOneLabel.....	10
3-5 SetSpeed.....	11
3-6 SetDensity.....	12
3-7 SetOrientation.....	13
3-8 SetCutter.....	14
3-9 SetBackFeed.....	15
3-10 SetPaper.....	16
3-11 SetMargin.....	17
3-12 SetOffset.....	18
3-13 PrintDeviceFont.....	19
3-14 PrintDeviceFontW.....	21
3-15 PrintVectorFont.....	23
3-16 PrintVectorFontW.....	25
3-17 Prints.....	28
3-18 Print1DBarcode.....	29
3-19 PrintMaxiCode.....	31
3-20 PrintPDF417.....	32
3-21 PrintQRCode.....	34
3-22 PrintDataMatrix.....	36
3-23 PrintAztec.....	37
3-24 PrintCode49.....	39
3-25 PrintCODABLOCK.....	41
3-26 PrintMicroPDF.....	43
3-27 PrintGS1DataBar.....	45
3-28 PrintBlock.....	47
3-29 PrintCircle.....	49
3-30 PrintImage.....	50
3-31 DirectIO.....	51
3-32 CalibrateMedia.....	52
3-33 ResetPrinter.....	53
3-34 ClearBuffer.....	54

1. 매뉴얼 안내

이 SDK 매뉴얼에는 Linux OS용 응용프로그램 개발에 필요한 라이브러리에 대한 내용이 기술되어 있습니다.

저희 (주)빅솔론은 제품의 기능과 품질 향상을 위하여 지속적인 개선을 하고 있습니다. 이로 인하여 제품의 사양과 매뉴얼의 내용은 사전 통보 없이 변경될 수 있습니다.

1-1 지원 Kernel & 플랫폼

* 지원 kernel 버전

- Kernel 2.6.32 이후 버전

* 지원 플랫폼

- Linux 32bit

- Linux 64bit

- Raspberry PI

1-2 지원 인터페이스

- USB, Serial, Parallel, Bluetooth, Ethernet, WLAN

2. Property

제공하는 라이브러리에서 사용하는 상수 값은 BxlLabelConst.h 파일에 선언되어 있습니다. 개발환경은 C를 기반으로 하였습니다.

2-1 CharacterSet & International CharacterSet(LONG R/W)

프린터의 코드페이지와 International character Set을 정의하는 속성으로 기본 값은 CS_CP437와 ICS_USA 으로 설정되어 있습니다. SetCharacterSet()함수를 사용하여 값을 설정할 수 있습니다.

사용 가능한 코드페이지는 다음과 같습니다.

상수명	값	설 명
CS_CP437	0	U.S.A
CS_CP850	1	Latin 1
CS_CP852	2	Latin 2
CS_CP860	3	Portuguese
CS_CP863	4	Canadian French
CS_CP865	5	Nordic
CS_WPC1252	6	Latin I
CS_CP865_WCP1252	7	European Combined
CS_CP857	8	Turkish
CS_CP737	9	Greek
CS_WCP1250	10	Latin 2
CS_WCP1253	11	Greek
CS_WCP1254	12	Turkish
CS_CP855	13	Cyrillic
CS_CP862	14	Hebrew
CS_CP866	15	Cyrillic
CS_WCP1251	16	Cyrillic
CS_WCP1255	17	Hebrew
CS_CP928	18	Greek
CS_CP864	19	Arabic
CS_CP775	20	Baltic
CS_WCP1257	21	Baltic
CS_CP858	22	Latin 1 + Euro

사용 가능한 International character Set은 다음과 같습니다.

상수명	값	설 명
ICS_USA	0	USA 코드 설정
ICS_FRANCE	1	FRANCE 코드 설정
ICS_GERMANY	2	GERMANY 코드 설정
ICS_UK	3	UK 코드 설정
ICS_DENMARK_I	4	DENMARK1 코드 설정
ICS_SWEDEN	5	SWEDEN 코드 설정
ICS_ITALY	6	ITALY 코드 설정
ICS_SPAIN_I	7	SPAIN 코드 설정
ICS_NORWAY	8	NORWAY 코드 설정
ICS_DENMARK_II	9	DENMARK 2 코드 설정
ICS_JAPAN	10	JAPAN 코드 설정
ICS_SPAIN_II	11	SPAIN 2 코드 설정
ICS_LATIN	12	LATIN 코드 설정
ICS_KOREA	13	KOREA 코드 설정
ICS_SLOVENIA	14	SLOVENIA 코드 설정
ICS_CHINA	15	CHINA 코드 설정

* Example

```
int ret;  
  
ret = ConnectPrinter ("portinfo...");  
  
.....  
  
SetCharacterSet(CS_PC850, ICS_UK);  
  
.....
```

2-2 State (LONG R)

프린터 상태가 설정되는 속성입니다. **CheckPrinterStatus** 함수를 호출하여 프린터 상태를 검사하고 상태 정보를 리턴 값으로 받습니다. 상태 값은 중복되어 설정될 수 있으며 비트 연산을 통해 각각의 값을 확인 할 수 있습니다.

프린터 상태 값은 다음과 같습니다.

상수명	값	설 명
STS_NORMAL	0	프린터 상태가 정상
STS_RIBONEND	4	리본 엔드 에러
STS_GAPERROR	8	갭 인식 불량 (오토 센싱 실패)
STS_TPHOVERHEAT	16	TPH 오버히트
STS_CUTTERJAM	32	커터 잼
STS_COVEROPEN	64	커버 열림
STS_PAPEREMPTY	128	프린터 용지가 없음

* Example

```
int ret;

ret = ConnectToPrinter ("portinfo...");
.....

int state;

state = CheckPrinterStatus();

if ((state & STS_RIBONEND) == STS_RIBONEND)
    .....

if ((state & STS_GAPERROR) == STS_GAPERROR)
    .....
.....
```

3. Method

Linux SDK에서 제공하는 함수들은 BxlLabelAPI.h 파일에 선언되어 있습니다.
개발환경은 C를 기반으로 하였습니다.

3-1 ConnectToPrinter

이 함수는 프린터와 통신을 위해 연결을 설정하는 함수입니다.

```
int ConnectToPrinter(const char *port)
```

[Parameters]

* const char *port
[in] 프린터와 연결할 인터페이스 정보

인터페이스	입력 정보	예시
USB	USB:	ConnectToPrinter("USB:");
Serial	serial:(baudrate) /dev/ttyX:(baudrate)	ConnectToPrinter("serial:115200") ConnectToPrinter("/dev/tty0:115200")
Parallel	parallel /dev/lpX	ConnectToPrinter("parallel") ConnectToPrinter("/dev/lp0")
Bluetooth	장치 MAC address	ConnectToPrinter("7d:f0:7d:e4:e0:78")
Ethernet, Wifi	IP주소, port번호	ConnectToPrinter("192.168.0.10:9100")

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
PORT_OPEN_ERROR	-99	통신 포트를 열 수 없습니다.
NO_CONNECTED_PRINTER	-100	프린터가 연결되어 있지 않습니다.
NO_BIXOLON_PRINTER	-101	BIXOLON 프린터가 아닙니다.

* Example

```
int ret;  
  
// USB  
ret = ConnectToPrinter("USB:");  
  
// Serial  
ret = ConnectToPrinter("serial:115200");  
  
// Parallel  
ret = ConnectToPrinter("parallel");  
  
// bluetooth  
ret = ConnectToPrinter("7d:f0:7d:e4:e0:78");  
  
// Ethernet or WiFi  
ret = ConnectToPrinter("192.168.0.10:9100");  
.....
```

3-2 DisconnectPrinter

이 함수는 프린터와 연결을 종료합니다.

int DisconnectPrinter()

[Parameters]

없음

[Return Values]

없음

* Example

```
ConnectToPrinter("portinfo...");  
  
.....  
DisconnectPrinter();
```


3-3 InitializePrinter

이 함수는 프린터를 초기화하는 기능을 지원합니다.

```
int InitializePrinter();
```

[Parameters]

없음

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
InitializePrinter();  
  
.....
```

3-4 FeedOneLabel

이 함수는 한 라벨을 피딩합니다.

int FeedOneLabel ()

[Parameters]

없음

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
ret = FeedOneLabel();  
  
.....
```

3-5 SetSpeed

이 함수는 인쇄 속도를 설정합니다.

int SetSpeed(int speed)

[Parameters]

* int speed
[in] 인쇄 속도

상수명	값	설 명
SPEED_25	0	2.5 ips
SPEED_30	1	3.0 ips
SPEED_40	2	4.0 ips
SPEED_50	3	5.0 ips
SPEED_60	4	6.0 ips
SPEED_70	5	7.0 ips
SPEED_80	6	8.0 ips

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
SetSpeed(SPEED_60);  
  
.....
```

3-6 SetDensity

이 함수는 인쇄 농도를 설정합니다.

int SetDensity(int density)

[Parameters]

* int density
[in] 인쇄 농도 (0 ~ 20)

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
SetDensity(20);  
  
.....
```

3-7 SetOrientation

이 함수는 인쇄 방향을 설정합니다.

int SetOrientation(int orientation)

[Parameters]

* int orientation
[in] 인쇄 방향

상수명	값	설 명
TOP_TO_BOTTOM	84	위에서 아래로 인쇄
BOTTOM_TO_TOP	66	아래에서 위로 인쇄

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
SetOrientation(TOP_TO_BOTTOM);  
  
.....
```

3-8 SetCutter

이 함수는 커팅 옵션을 설정합니다.

int SetCutter(bool autoCut, int cutPeriod)

[Parameters]

- * bool autoCut
[in] 커팅 여부. false : 커팅 안함, true : 커팅
- * int cuttingPeriod
[in] 커팅 간격

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
SetCutter(false, 0);  
  
.....
```

3-9 SetBackFeed

이 함수는 back feeding 옵션을 설정합니다.

int SetBackFeed(bool backFeed)

[Parameters]

* bool backFeed
[in] 인쇄 시작 전 back-feed 여부. false : back-feed 안함, true : back-feed

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-300	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
SetBackFeed(false);  
  
.....
```

3-10 SetPaper

이 함수는 인쇄할 용지에 대한 옵션을 설정합니다.

int SetPaper(int width, int height, int mediaType, int offset, int gapLength);

[Parameters]

- * int width
[in] 용지 너비. 최대 832(4.1 inch) [dot]
- * int height
[in] 용지 높이. 최대 2432(12 inch) [dot]
- * int mediaType
[in] 용지 타입

상수명	값	설 명
MEDIA_GAP	0	Gap 용지
MEDIA_CONTINUOUS	1	연속 용지
MEDIA_BLACKMARK	2	Black Mark 용지

- * int offset
[in] Gap 또는 Blackmark의 옵셋
- * int gapLength
[in] Gap 길이 또는 Blackmark 두께 [dot]

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
SetPaper(832, 1261, MEDIA_GAP, 0, 20);  
  
.....
```


3-11 SetMargin

이 함수는 인쇄할 용지의 여백을 설정합니다.

int SetMargin(int horizontalMargin, int verticalMargin)

[Parameters]

* int horizontalMargin
[in] 가로 마진 [dot]

* int verticalMargin
[in] 세로 마진 [dot]

[Return Values]

상수명	값	설명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
SetMargin(10, 10);  
  
.....
```

3-12 SetOffset

이 함수는 Black Mark/Gap과 절취선 간에 offset 길이를 설정합니다.

int SetOffset(int offset)

[Parameters]

* int offset
[in] offset 길이 [dot] (-100 ~ 100)

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
SetOffset(10);  
  
.....
```

3-13 PrintDeviceFont

이 함수는 디바이스 폰트를 인쇄하는 기능을 지원합니다.

```
int PrintDeviceFont(const char *text, const int xPos, const int yPos, const int fontName,
                   const int xMulti, const int yMulti, const int rotation, const bool bold);
```

[Parameters]

- * const char *text
[in] 인쇄할 문자열
- * const int xPos
[in] 수평 포지션 (X) [dot]
- * const int yPos
[in] 수직 포지션 (Y) [dot]
- * const int fontName
[in] 폰트 이름

바코드	값	설명
DEVICE_ENG_9X15	0	Size 6 (9 X 15)
DEVICE_ENG_12X20	1	Size 8 (12 X 20)
DEVICE_ENG_16X25	2	Size 10 (16 X 25)
DEVICE_ENG_19X30	3	Size 12 (19 X 30)
DEVICE_ENG_24X38	4	Size 15 (24 X 38)
DEVICE_ENG_32X50	5	Size 20 (32 X 50)
DEVICE_ENG_48X76	6	Size 30 (48 X 76)
DEVICE_ENG_22X34	7	Size 14 (22 X 34)
DEVICE_ENG_28X44	8	Size 18 (28 X 44)
DEVICE_ENG_37X58	9	Size 24 (37 X 58)
DEVICE_KOR_16X16	0x61	Size 1 (16 X 16)
DEVICE_KOR_24X24	0x62	Size 2 (24 X 24)
DEVICE_KOR_20X20	0x63	Size 3 (20 X 20)
DEVICE_KOR_26X26	0x64	Size 4 (26 X 26)
DEVICE_KOR_20X26	0x65	Size 5 (20 X 26)
DEVICE_KOR_38X38	0x66	Size 6 (38 X 38)
DEVICE_CHN_GB2312	0x6D	GB2312 (24 X 24)
DEVICE_CHN_BIG5	0x6E	BIG5 (24 X 24)

- * const int xMulti
[in] 수평 확대 (1 ~ 4)
- * const int yMulti
[in] 수직 확대 (1 ~ 4)

* const int rotation

[in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

* bool bold

[in] 굵기. false : 보통, true : 굵게

[Return Values]

상수명	값	설 명
BXL_SUCCESS	0	함수가 성공할 때 리턴됩니다.
BXL_NOT_OPENED	-101	ConnectPrinter를 실행해야 합니다.
BXL_WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;

ret = ConnectToPrinter("portinfo...");

.....

PrintDeviceFont("BIXOLON LABEL SDK TEST.", 20, 10, DEVICE_ENG_24X38,
                 1, 1, ROTATE_0, false);
Prints(1, 1);

.....
```

3-14 PrintDeviceFontW

이 함수는 디바이스 폰트를 인쇄하는 기능을 지원합니다.

```
int PrintDeviceFontW(const char *text, const int xPos, const int yPos, const int fontName,  
                    const int xMulti, const int yMulti, const int rotation, const bool bold,  
                    const unsigned codePage);
```

[Parameters]

- * const char *text
[in] 인쇄할 문자열

- * const int xPos
[in] 수평 포지션 (X) [dot]

- * const int yPos
[in] 수직 포지션 (Y) [dot]

- * const int fontName
[in] 폰트 이름

바코드	값	설명
DEVICE_ENG_9X15	0	Size 6 (9 X 15)
DEVICE_ENG_12X20	1	Size 8 (12 X 20)
DEVICE_ENG_16X25	2	Size 10 (16 X 25)
DEVICE_ENG_19X30	3	Size 12 (19 X 30)
DEVICE_ENG_24X38	4	Size 15 (24 X 38)
DEVICE_ENG_32X50	5	Size 20 (32 X 50)
DEVICE_ENG_48X76	6	Size 30 (48 X 76)
DEVICE_ENG_22X34	7	Size 14 (22 X 34)
DEVICE_ENG_28X44	8	Size 18 (28 X 44)
DEVICE_ENG_37X58	9	Size 24 (37 X 58)
DEVICE_KOR_16X16	0x61	Size 1 (16 X 16)
DEVICE_KOR_24X24	0x62	Size 2 (24 X 24)
DEVICE_KOR_20X20	0x63	Size 3 (20 X 20)
DEVICE_KOR_26X26	0x64	Size 4 (26 X 26)
DEVICE_KOR_20X26	0x65	Size 5 (20 X 26)
DEVICE_KOR_38X38	0x66	Size 6 (38 X 38)
DEVICE_CHN_GB2312	0x6D	GB2312 (24 X 24)
DEVICE_CHN_BIG5	0x6E	BIG5 (24 X 24)

- * const int xMulti
[in] 수평 확대 (1 ~ 4)

- * const int yMulti
[in] 수직 확대 (1 ~ 4)

* const int rotation

[in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

* bool bold

[in] 굵기. false : 보통, true : 굵게

* const unsigned int codePage

[in] 문자열을 인코딩 방식을 설정합니다.

상수명	값	설 명
CP_EUCKR	0	한국어. 문자열을 EUC-KR로 인코딩합니다.
CP_CP949	1	한국어. 문자열을 CP949로 인코딩합니다.
CP_EUCCN	2	중국어. 문자열을 EUC-CN으로 인코딩합니다.
CP_GB18030	3	중국어. 문자열을 GB18030으로 인코딩합니다.
CP_BIG5	4	중국어. 문자열을 BIG5로 인코딩합니다.
CP_CP950	5	중국어. 문자열을 CP950으로 인코딩합니다.
CP_EUCJP	6	일본어. 문자열을 EUC-JP로 인코딩합니다.
CP_CP932	7	일본어. 문자열을 CP932로 인코딩합니다.
CP_CP874	8	태국어. 문자열을 CP874로 인코딩합니다.

[Return Values]

상수명	값	설 명
BXL_SUCCESS	0	함수가 성공할 때 리턴됩니다.
BXL_NOT_OPENED	-101	ConnectPrinter를 실행해야 합니다.
BXL_WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;

ret = ConnectToPrinter("portinfo...");

.....

PrintDeviceFont("빅솔론 라벨프린터 SDK 테스트.", 20, 10, DEVICE_ENG_24X38,
                1, 1, ROTATE_0, false, CP_CP949);
Prints(1, 1);

.....
```

3-15 PrintVectorFont

이 함수는 벡터 폰트를 인쇄하는 기능을 지원합니다.

Long PrintVectorFont(const char *text, const int xPos, const int yPos, const int font, const int fontWidth, const int fontHeight, const int rightSpace, const bool bold, const bool reverse, const bool italic, const int rotation, const int alignment, const int printDirection)

[Parameters]

- * const char *text
[in] 인쇄할 문자열
- * const int xPos
[in] 수평 포지션 (X) [dot]
- * const int yPos
[in] 수직 포지션 (Y) [dot]
- * const int font
[in] 폰트 선택

바코드	값	설 명
VECTOR_ASCII	0	ASCII (1Byte Code)
VECTOR_KS5601	1	KS5601(2Byte Code)
VECTOR_BIG5	2	BIG5(2Byte Code)
VECTOR_GB2312	3	GB2312(2Byte Code)
VECTOR_JIS	4	Shift-JIS(2Byte Code)
VECTOR_OCRA	5	OCR-A(1Byte Code)
VECTOR_OCRB	6	OCR-B(1Byte Code)

- * const int fontWidth
[in] 폰트 폭 [dot]
- * const int fontHeight
[in] 폰트 높이 [dot]
- * const int rightSpace
[in] 오른쪽 자간 [dot]. +/- 옵션을 사용할 수 있습니다.
- * const bool bold
[in] 굵기. false : 보통, true : 굵게
- * const bool reverse
[in] 문자 역상. false : 정상, true : 역상
- * const bool italic
[in] italic체 사용. false : 정상, true : italic체 사용

* const int rotation
 [in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

* const int alignment
 [in] 텍스트 정렬

상수명	값	설 명
ALIGNMENT_LEFT	0	왼쪽 정렬
ALIGNMENT_CENTER	1	가운데 정렬
ALIGNMENT_RIGHT	2	오른쪽 정렬

* int printDirection
 [in] 문자열 인쇄 방향.

상수명	값	설 명
LEFT_TO_RIGHT	0	왼쪽에서 오른쪽으로 인쇄 (ex. BIXOLON)
RIGHT_TO_LEFT	1	오른쪽에서 왼쪽으로 인쇄 (ex. NOLOXIB)

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;

ret = ConnectToPrinter("portinfo...");

.....

PrintVectorFont("VECTOR FONT.", 20, 150, VECTOR_ASCII, 10, 10, 3, false, false,
                false, ROTATE_0, ALIGNMENT_LEFT, LEFT_TO_RIGHT);
Prints(1, 1);

.....
```


3-16 PrintVectorFontW

이 함수는 벡터 폰트를 인쇄하는 기능을 지원합니다.

Long PrintVectorFontW(const char *text, const int xPos, const int yPos, const int font, const int fontWidth, const int fontHeight, const int rightSpace, const bool bold, const bool reverse, const bool italic, const int rotation, const int alignment, const int printDirection, const unsigned int codePage)

[Parameters]

- * const char *text
[in] 인쇄할 문자열
- * const int xPos
[in] 수평 포지션 (X) [dot]
- * const int yPos
[in] 수직 포지션 (Y) [dot]
- * const int font
[in] 폰트 선택

바코드	값	설 명
VECTOR_ASCII	0	ASCII (1Byte Code)
VECTOR_KS5601	1	KS5601(2Byte Code)
VECTOR_BIG5	2	BIG5(2Byte Code)
VECTOR_GB2312	3	GB2312(2Byte Code)
VECTOR_JIS	4	Shift-JIS(2Byte Code)
VECTOR_OCRA	5	OCR-A(1Byte Code)
VECTOR_OCRB	6	OCR-B(1Byte Code)

- * const int fontWidth
[in] 폰트 폭 [dot]
- * const int fontHeight
[in] 폰트 높이 [dot]
- * const int rightSpace
[in] 오른쪽 자간 [dot]. +/- 옵션을 사용할 수 있습니다.
- * const bool bold
[in] 굵기. false : 보통, true : 굵게
- * const bool reverse
[in] 문자 역상. false : 정상, true : 역상
- * const bool italic
[in] italic체 사용. false : 정상, true : italic체 사용

* const int rotation
 [in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

* const int alignment
 [in] 텍스트 정렬

상수명	값	설 명
ALIGNMENT_LEFT	0	왼쪽 정렬
ALIGNMENT_CENTER	1	가운데 정렬
ALIGNMENT_RIGHT	2	오른쪽 정렬

* int printDirection
 [in] 문자열 인쇄 방향.

상수명	값	설 명
LEFT_TO_RIGHT	0	왼쪽에서 오른쪽으로 인쇄 (ex. BIXOLON)
RIGHT_TO_LEFT	1	오른쪽에서 왼쪽으로 인쇄 (ex. NOLOXIB)

* const unsigned int codePage
 [in] 문자열을 인코딩 방식을 설정합니다.

상수명	값	설 명
CP_EUCKR	0	한국어. 문자열을 EUC-KR로 인코딩합니다.
CP_CP949	1	한국어. 문자열을 CP949로 인코딩합니다.
CP_EUCCN	2	중국어. 문자열을 EUC-CN으로 인코딩합니다.
CP_GB18030	3	중국어. 문자열을 GB18030으로 인코딩합니다.
CP_BIG5	4	중국어. 문자열을 BIG5로 인코딩합니다.
CP_CP950	5	중국어. 문자열을 CP950으로 인코딩합니다.
CP_EUCJP	6	일본어. 문자열을 EUC-JP로 인코딩합니다.
CP_CP932	7	일본어. 문자열을 CP932로 인코딩합니다.
CP_CP874	8	태국어. 문자열을 CP874로 인코딩합니다.

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
ret = ConnectToPrinter("portinfo...");  
.....  
PrintVectorFontW("빅솔론 라벨프린터 SDK 테스트.", 20, 150, VECTOR_ASCII, 10,  
10, 3, false, false, false, ROTATE_0, ALIGNMENT_LEFT,  
LEFT_TO_RIGHT, CP_CP949);  
Prints(1, 1);  
.....
```

3-17 Prints

이 함수는 버퍼의 내용을 인쇄합니다.

int Prints(const int nLabelSet, const int nCopies);

[Parameters]

- * const int nLabelSet
[in] 라벨 세트 수 (1 ~ 65535)

- * const int nCopies
[in] 라벨의 복수 매수 (1 ~ 65535)

[Return Values]

상수명	값	설명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
Prints(1, 1);  
  
.....
```

3-18 Print1DBarcode

이 함수는 1차원 바코드를 인쇄하는 기능을 지원합니다.

```
int Print1DBarcode(const char *barcodeData, const int xPos, const int yPos,  
                  const int barcodeType, const int narrowBarWidth,  
                  const int wideBarWidth, const int barcodeHeight, const int rotation,  
                  const int HRI);
```

[Parameters]

- * const char *barcodeData
[in] 바코드 출력 데이터

- * const int xPos
[in] 수평 포지션 (X) [dot]

- * const int yPos
[in] 수직 포지션 (Y) [dot]

- * const int barcodeType
[in] 바코드 타입을 정의합니다. bxlConst.h 에 정의 되어 있습니다.

바코드	값	설 명
BAR_CODE39	0	Code39
BAR_CODE128	1	Code128
BAR_I2OF5	2	Interleaved 2of5
BAR_CODABAR	3	Codabar
BAR_CODE93	4	Code93
BAR_UPCA	5	UPC-A
BAR_UPCE	6	UPC-E
BAR_EAN13	7	EAN13
BAR_EAN8	8	EAN8
BAR_EAN128	9	UCC/EAN128
BAR_CODE11	10	Code11
BAR_PLANET	11	Planet
BAR_INDUSTRIAL2OF5	12	Industrial 2of5
BAR_STANDARD2OF5	13	Standard 2of5
BAR_LOGMARS	14	logmars
BAR_EXTENSION	15	UPC/EAN Extensions
BAR_POSTNET	16	Postnet

- * int narrowBarWidth
[in] 좁은 바의 너비를 설정하는 값으로 Dot 단위를 가지고 있습니다.

- * int wideBarWidth
[in] 넓은 바의 너비를 설정하는 값으로 Dot 단위를 가지고 있습니다.

- * int barcodeHeight
[in] 바코드의 높이를 설정하는 값으로 Dot 단위를 가지고 있습니다.

* int rotation
[in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

* int HRI
[in] HRI 출력위치와 크기를 설정하는 값으로 0 ~ 8까지의 값을 가집니다.

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_CODE39, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_CODE128, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_I2OF5, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_CODABAR, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_CODE93, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_UPCA, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_UPCE, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_EAN13, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_EAN8, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
Print1DBarcode("123456789012", 220, 60, BXL_1DBAR_EAN128, 2, 5, 100, BXL_ROTATE_90, 1);  
Prints(1, 1);  
  
.....
```

3-19 PrintMaxiCode

이 함수는 2차원 바코드(Maxicode)를 인쇄하는 기능을 지원합니다.

```
int PrintMaxiCode(const char *barcodeData, const int xPos, const int yPos,  
                 const int mode);
```

[Parameters]

- * const char *barcodeData
[in] 바코드 출력 데이터

- * const int xPos
[in] 수평 포지션 (X) [dot]

- * const int yPos
[in] 수직 포지션 (Y) [dot]

- * const int mode
[in] Maxicode 모드.

상수명	값	설 명
MAXICODE_MODE0	0	모드 0
MAXICODE_MODE90	2	모드 2
MAXICODE_MODE180	3	모드 3
MAXICODE_MODE270	4	모드 4

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
PrintMaxiCode("990,840,06840,THIS IS A TEST OF MODE 0 BIXOLON LABEL  
                PRINTER", 30, 100, MAXICODE_MODE0);  
Prints(1, 1);  
  
.....
```

3-20 PrintPDF417

이 함수는 2차원 바코드(PDF417)를 인쇄하는 기능을 지원합니다.

```
int PrintBarcode(const char *barcodeData, const int xPos, const int yPos,
                const int verticalCount, const int horizontalCount, int errorLevel,
                int dataComp, bool HRI, int startPosition, int moduleWidth,
                int barHeight, int rotation);
```

[Parameters]

- * const char *barcodeData
[in] 바코드 출력 데이터
- * const int xPos
[in] 수평 포지션 (X) [dot]
- * const int yPos
[in] 수직 포지션 (Y) [dot]
- * int verticalCount
[in] 최대 가로 count : 3 ~ 90
- * int horizontalCount
[in] 최대 세로 count : 1 ~ 30
- * int errorLevel
[in] 에러 교정 레벨. bxlConst.h 에 정의 되어 있습니다.

바코드	값	설 명
PDF417_ECL0	0	EC 레벨 : 0. EC Codeword : 2
PDF417_ECL1	1	EC 레벨 : 1. EC Codeword : 4
PDF417_ECL2	2	EC 레벨 : 2. EC Codeword : 8
PDF417_ECL3	3	EC 레벨 : 3. EC Codeword : 16
PDF417_ECL4	4	EC 레벨 : 4. EC Codeword : 32
PDF417_ECL5	5	EC 레벨 : 5. EC Codeword : 64
PDF417_ECL6	6	EC 레벨 : 6. EC Codeword : 128
PDF417_ECL7	7	EC 레벨 : 7. EC Codeword : 256
PDF417_ECL8	8	EC 레벨 : 8. EC Codeword : 512

- * int dataComp
[in] 데이터 압축 방법. bxlConst.h 에 정의 되어 있습니다.

상수명	값	설 명
PDF417_COMP_TEXT	0	2 Characters per codeword.
PDF417_COMP_NUM	1	2.93 Characters per codeword.
PDF417_COMP_BINARY	2	1.2 Bytes per codeword.

- * bool HRI

[in] HRI 출력 여부를 설정

* int startPosition

[in] 0이면 바코드의 가운데가 시작 지점이 되고 1이면 바코드의 좌측상단이 시작 지점이 됩니다.

* int moduleWidth

[in] 모듈의 폭을 설정(2 ~ 9).

* int barHeight

[in] 바의 높이를 설정(4 ~ 99).

* int rotation

[in] 회전 값을 설정

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
PrintPDF417("BIXOLON Label Printer, This is Test Printing.", 30, 100, 30, 5,  
             PDF417_ECL0, PDF417_COMP_TEXT, true, 1, 3, 10, ROTATE_0);  
Prints(1, 1);  
  
.....
```

3-21 PrintQRCode

이 함수는 2차원 바코드(QR Code)를 인쇄하는 기능을 지원합니다.

```
int PrintQRCode(const char *barcodeData, const int xPos, const int yPos, int model,
                const int eccLevel, int barSize, int rotation);
```

[Parameters]

- * const char *barcodeData
[in] 바코드 출력 데이터
- * const int xPos
[in] 수평 포지션 (X) [dot]
- * const int yPos
[in] 수직 포지션 (Y) [dot]
- * int model
[in] 모델 선택. 1 : 모델1, 2 : 모델2
- * int eccLevel
[in] ECC 레벨

상수명	값	설 명
QRCODE_ECC7	0	복구 비율 7%
QRCODE_ECC15	1	복구 비율 15%
QRCODE_ECC25	2	복구 비율 25%
QRCODE_ECC30	3	복구 비율 30%

- * int barSize
[in] 바코드 크기 설정(1 ~ 4).
- * int rotation
[in] 회전 값을 설정

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
ret = ConnectToPrinter("portinfo...");  
  
.....  
PrintQRCode("ABCDEFGHJKLMN1234567890", 30, 100, 1, QRCODE_ECC7,  
             4, ROTATE_0);  
Prints(1, 1);  
  
.....
```

3-22 PrintDataMatrix

이 함수는 2차원 바코드(Data Matrix)를 인쇄하는 기능을 지원합니다.

```
int PrintDataMatrix(const char *barcodeData, const int xPos, const int yPos,  
                  int barSize, bool reverse, int rotation);
```

[Parameters]

- * const char *barcodeData
[in] 바코드 출력 데이터
- * const int xPos
[in] 수평 포지션 (X) [dot]
- * const int yPos
[in] 수직 포지션 (Y) [dot]
- * int barSize
[in] 바코드 크기 설정(1 ~ 4)
- * bool reverse
[in] 바코드 역상. false : 정상 이미지, true : 역상 이미지
- * int rotation
[in] 회전 값을 설정

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
PrintDataMatrix("BIXOLON Label Printer", 30, 100, 4, false, ROTATE_0);  
Prints(1, 1);  
  
.....
```

3-23 PrintAztec

이 함수는 2차원 바코드(Aztec)를 인쇄하는 기능을 지원합니다.

```
int PrintAztec(const char *barcodeData, const int xPos, const int yPos, int barSize,  
              int interpretation, int errCodeNSymbolSize, bool menuSymbol,  
              int numofSymbol, int optID, int rotation);
```

[Parameters]

* const char *barcodeData
[in] 바코드 출력 데이터

* const int xPos
[in] 수평 포지션 (X) [dot]

* const int yPos
[in] 수직 포지션 (Y) [dot]

* int barSize
[in] 바코드 크기 설정(1 ~ 10).

* int interpretation
[in] ECI(Extended Channel Interpretation) 코드 설정. 0 : 해제, 1 : 설정.

* int errCodeNSymbolSize
[in] 에러 코드와 심볼 사이즈/타입

값	설 명
0	디폴트 error collection 레벨
1 ~ 99	error collection 퍼센트
101 ~ 104	1 ~ 4 레이어 컴팩트 심볼
201 ~ 232	1 ~ 32 레이어 full range 심볼
300	심플 Aztec "Rune"

* bool menuSymbol
[in] 메뉴 심볼

* bool numofSymbol
[in] Number of symbols for structured append (1 ~ 26)

* int optID
[in] Optional ID filed for structured append : ID field string (최대 24글자)

* int rotation
[in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
PrintAztec("THIS IS AZTEC BARCODE TESTTHIS IS AZTEC BARCODE TEST",  
           30, 100, 5, 0, 0, true, 1, 1, ROTATE_0);  
Prints(1, 1);  
  
.....
```

3-24 PrintCode49

이 함수는 2차원 바코드(Code49)를 인쇄하는 기능을 지원합니다.

```
int PrintCode49(const char *barcodeData, const int xPos, const int yPos,  
               const int narrowWidth, const int wideWidth, const int barHeight,  
               int HRI, int startingMode, int rotation);
```

[Parameters]

- * const char *barcodeData
[in] 바코드 출력 데이터

- * const int xPos
[in] 수평 포지션 (X) [dot]

- * const int yPos
[in] 수직 포지션 (Y) [dot]

- * const int narrowWidth
[in] 좁은 바의 폭 [dot]

- * const int wideWidth
[in] 넓은 바의 폭 [dot]

- * const int barHeight
[in] 바코드 높이 [dot]

- * int HRI
[in] HRI 출력. 0 : 인쇄 없음, 1 : 바코드 아래, 2 : 바코드 위

- * int startingMode
[in] starting mode

값	설 명
0	Regular Alphanumeric Mode
1	Multiple Read Alphanumeric
2	Regular Numeric Mode
3	Group Alphanumeric Mode
4	Regular Alphanumeric Shift 1
5	Regular Alphanumeric Shift 2
7	Automatic Mode

* int rotation
[in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
PrintCode49("12345ABC", 30, 100, 2, 7, 22, 2, 7, ROTATE_0);  
  
Prints(1, 1);  
  
.....
```


3-25 PrintCODABLOCK

이 함수는 2차원 바코드(CODABLOCK)를 인쇄하는 기능을 지원합니다.

```
int PrintCODABLOCK(const char *barcodeData, const int xPos, const int yPos,  
                  const int narrowWidth, const int wideWidth, const int barHeight,  
                  const bool security, int dataColumns, int mode, int encodeRow);
```

[Parameters]

- * const char *barcodeData
[in] 바코드 출력 데이터

- * const int xPos
[in] 수평 포지션 (X) [dot]

- * const int yPos
[in] 수직 포지션 (Y) [dot]

- * const int narrowWidth
[in] 좁은 바의 폭 [dot]

- * const int wideWidth
[in] 넓은 바의 폭 [dot]

- * const int barHeight
[in] 바코드 높이 [dot]

- * const bool security
[in] 보안기능

- * int dataColumns
[in] 라인 당 글자 수 (2 ~ 62)

- * int mode
[in] 모드

상수명	값	설 명
CODABLOCK_A	0	Code 39 문자 집합 사용
CODABLOCK_E	1	Code 128 문자 집합 사용
CODABLOCK_F	2	Code 128 문자 집합과 Function 1 (FNC1) 자동 추가

- * int encodeRow
[in] 인코드 할 라인 수

값	설 명
A	1 ~ 18
E	2 ~ 4
F	2 ~ 4

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
PrintCODABLOCK("BIXOLON BARCODE TEST 123BIXOLON BARCODE TEST 123",  
                 30, 100, 2, 5, 30, false, 30, CODABLOCK_E, 4);  
Prints(1, 1);  
  
.....
```

3-26 PrintMicroPDF

이 함수는 2차원 바코드(Micro-PDF417)를 인쇄하는 기능을 지원합니다.

```
int PrintMicroPDF(const char *barcodeData, const int xPos, const int yPos,  
                 int moduleWidth, int barHeight, int mode, int rotation);
```

[Parameters]

- * const char *barcodeData
[in] 바코드 출력 데이터

- * const int xPos
[in] 수평 포지션 (X) [dot]

- * const int yPos
[in] 수직 포지션 (Y) [dot]

- * int moduleWidth
[in] 모듈 폭 (2 ~ 8)

- * int barHeight
[in] 바코드 높이 (1 ~ 99) [dot]

- * int mode
[in] 모드 (0 ~ 33), 세부 내용은 커맨드 매뉴얼 참조

- * int rotation
[in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
ret = ConnectToPrinter("portinfo...");  
.....  
PrintMicroPDF("ABCDEFGHJKLMN1234567890", 30, 100, 2, 6, 8, ROTATE_0);  
Prints(1, 1);  
.....
```

3-27 PrintGS1DataBar

이 함수는 GS1 DataBar 바코드를 인쇄하는 기능을 지원합니다.

Long PrintGS1DataBar(const char *barcodeData, const int xPos, const int yPos,
int barcodeType, int expand, int separatorHeight,
int barHeight, int segmentWidth, int rotation);

[Parameters]

* const char *barcodeData
[in] 바코드 출력 데이터

* const int xPos
[in] 수평 포지션 (X) [dot]

* const int yPos
[in] 수직 포지션 (Y) [dot]

* int barcodeType
[in] 바코드 타입

상수명	값	설명
GS1DATABAR	0	GS1 DataBar
GS1DATABAR_TRUNCATED	1	GS1 DataBar Truncated
GS1DATABAR_STACKED	2	GS1 DataBar Stacked
GS1DATABAR_STACKED_OMNIDIRECTIONAL	3	GS1 DataBar Stacked Omnidirectional
GS1DATABAR_LIMITED	4	GS1 Limited
GS1DATABAR_EXPANDED	5	GS1 Expanded

* int expand
[in] 확대 (1 ~ 10)

* int separatorHeight
[in] Separator 높이 (1 ~ 2)

* int barHeight
[in] barcode 높이

* int segmentWidth
[in] segment 너비 (0 ~ 22. 짝수만 가능)

* int rotation
[in] 회전 값을 설정합니다.

상수명	값	설 명
ROTATE_0	0	0도 회전
ROTATE_90	1	90도 회전
ROTATE_180	2	180도 회전
ROTATE_270	3	270도 회전

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
.....  
  
PrintGS1DataBar("0102005190000570031512291036310300050091320000050304",  
                 30, 100, GS1DATABAR_EXPANDED, 2, 2, 17, 10, ROTATE_0);  
Prints(1, 1);  
.....
```

3-28 PrintBlock

이 함수는 라인, 블록, 박스 및 사선을 인쇄하는 기능을 지원합니다.

```
int PrintBlock(const int xStart, const int yStart, const int xEnd, const int yEnd,  
              const int option, const int thickness);
```

[Parameters]

- * const int xStart
[in] 가로 시작 좌표 [dot]
- * const int yStart
[in] 세로 시작 좌표 [dot]
- * const int xEnd
[in] 가로 끝 좌표 [dot]
- * const int yEnd
[in] 세로 끝 좌표 [dot]
- * const int option
[in] 모드 (0 ~ 33), 세부 내용은 커맨드 매뉴얼 참조

상수명	값	설 명
BLOCK_OVERWRITE	0	라인 Overwriting
BLOCK_EXCLUSIVEOR	1	라인 Exclusive OR
BLOCK_DELETE	2	라인 삭제
BLOCK_SLOPE	3	사선
BLOCK_BOX	4	박스

- * int thickness
[in] 선 굵기. 사선 또는 박스일 경우에만 적용

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
ret = ConnectToPrinter("portinfo...");  
.....  
PrintBlock(20, 20, 300, 300, BLOCK_BOX, 10);  
PrintBlock(400, 20, 20, 500, BLOCK_SLOPE, 10);  
  
Prints(1, 1);  
.....
```


3-29 PrintCircle

이 함수는 원을 인쇄하는 기능을 지원합니다.

```
int PrintCircle(const int xPos, const int yPos, int size, int multi);
```

[Parameters]

- * const int xPos
[in] 수평 포지션 (X) [dot]
- * const int yPos
[in] 수직 포지션 (Y) [dot]
- * int size
[in] 원의 크기 (1 ~ 6)
- * int multi
[in] 확대 (1 ~ 4)

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
.....  
PrintCircle(150, 150, 4, 1);  
  
Prints(1, 1);  
  
.....
```

3-30 PrintImage

이 함수는 이미지 파일(bmp, jpg, gif)을 선택하여 인쇄하는 기능을 지원합니다.

```
int PrintImage(const char *imagePath, const int xPos, const int yPos,  
              const bool compress);
```

[Parameters]

- * const char *imagePath
[in] 이미지 파일 경로

- * const int xPos
[in] 수평 포지션 (X) [dot]

- * const int yPos
[in] 수직 포지션 (Y) [dot]

- * const bool compress
[in] 압축 여부

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴됩니다.
IMAGE_OPEN_ERROR	-118	이미지 파일을 열수가 없습니다.
MEM_ALLOC_ERROR	-120	내부 메모리 할당을 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
char *imgPath = "....";  
  
PrintImage(imgPath, 150, 150, false);  
  
Prints(1, 1);  
  
.....
```

3-31 DirectIO

이 함수는 SLCS 명령어를 직접 전송하고 응답을 읽어 올 수 있습니다.

int DirectIO(const char *writeData, const writeLen, char *readData, int *readLen);

[Parameters]

* const char *writeData
[in] 전송할 데이터

* const int writeLen
[in] 전송할 데이터 길이

* char *readData
[in] 데이터를 읽어올 버퍼

* int *readLen
[in] 읽어 들일 길이를 전달하고 읽어 들인 길이를 받음

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.
READ_ERROR	-106	데이터 수신에 실패하였습니다.

* Example

```
int ret;  
char cmd = "T20,20,3,1,1,0,0,N,N,\BIXOLON Label Printer\";  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
DirectIO(cmd, strlen(cmd), NULL, 0);  
  
.....
```

3-32 CalibrateMedia

이 함수는 오토 칼리브레이션 기능을 지원합니다.

int CalibrateMedia();

[Parameters]

없음

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
CalibrateMedia();  
  
.....
```

3-33 ResetPrinter

이 함수는 프린터를 리부팅하는 기능을 지원합니다.

```
int ResetPrinter();
```

[Parameters]

없음

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
ResetPrinter();  
  
.....
```

3-34 ClearBuffer

이 함수는 프린터의 이미지 버퍼에 있는 데이터를 삭제합니다.

```
int ClearBuffer();
```

[Parameters]

없음

[Return Values]

상수명	값	설 명
SUCCESS	0	함수가 성공할 때 리턴 됩니다.
WRITE_ERROR	-105	데이터 전송이 실패하였습니다.

* Example

```
int ret;  
  
ret = ConnectToPrinter("portinfo...");  
  
.....  
  
ClearBuffer();  
  
.....
```