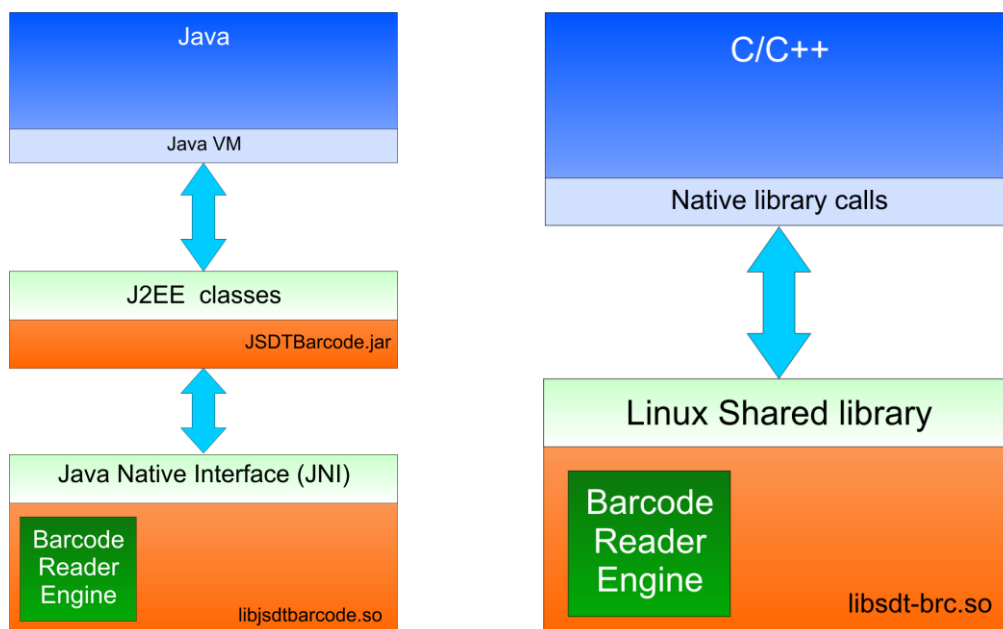


Introduction

The SD-TOOLKIT® Barcode Reader SDK 2.1 for Linux is a developer kit for Linux platforms starting from Ubuntu 10. This document and sample applications are provided to help you to integrate high performance barcode recognition functionality into your linux C/C++ or Java application or solution.



Overview

To access barcode reader functionality from your application code do the following steps

1. Add SD-TOOLKIT SDK *include* folder path to your application project or makefile include folders search path.
2. Add SD-TOOLKIT SDK *lib/x86* or *lib/x64* folder path to your application project or makefile library search path.
3. Create instance of BarcodeReader by calling SDTCreateBarcodeReader.
4. Specify if needed scan area on the input image in pixels where engine will look for barcode symbols by calling SDTSetActiveScanRect. By default will be used whole image.
5. Specify barcode symbol types the BarcodeReader will look for by calling SDTSetReadInputTypes. By default the input image will be analyzed for all types which could decrease performance in most cases.
6. Call either SDTReadImageFileA, SDTReadImageFileW or SDTReadRGBImageBuffer to process the image file or image rgb data buffer respectively.
7. Check count of recognized barcode symbols by calling SDTGetResultsCount
8. Obtain recognized barcode symbol value and properties by calling SDTGetResultValue and SDTGetResultProperties
9. To destroy the BarcodeReader and its allocated memory call SDTDestroyBarcodeReader.

The following C++ code demonstrate how to read pdf file.

A complete code you can find in sdt-brc-sample sample application included into SD-TOOLKIT Barcode SDK for Linux.

```
LPSPDBARREADER rdr = SDTCreateBarcodeReader(L"UNKNOWN");

if( rdr != NULL)
{
    unsigned long nRes = SDTSetReadInputTypes( rdr, SDTBARCODETYPE_ALL_1D);
    if(nRes == 0)
    {
        nRes = SDTSetReadInputDirections( rdr, SDTREADDIRECTION_LTR);
    }
    if(nRes == 0 )
    {
        nRes = SDTReadImageFileA(rdr, "/xfiles/auto/test.pdf", 0);
    }

    if(nRes == 0)
    {
        int iResCnt = SDTGetResultsCount(rdr);

        for(int pos = 0; pos < iResCnt; pos++)
```

```
        {
            const char* pchValue = SDTGetResultValueA(rdr, pos);
            if(pchValue != NULL)
            {
                printf("%d. ResultA: %s\n", pos, pchValue);
            }
        }
    }
    SDTDestroyBarcodeReader(rdr);
    rdr = NULL;
}
```

BarcodeReader API Functions Reference

Library	libsdt-brc.so
Declared in	sdtbarcode.h
Related sample code	sdt-brc-sample

Overview

The ST-TOOLKIT Barcode SDK API functions are available from libsdt-brc.so shared library included into the SDK. Both 32 bit and 64 bit version are available and are located in x86 and x64 folders.

API Functions

SDTCreateBarcodeReader

Function Initialize new barcode reader instance.

```
LPSDBARREADER SDTCreateBarcodeReader (const wchar_t* lpLicenseKey);
```

Parameters

license

The purchased Individual Developer or Team Developer license. For purchase details visit <http://www.sd-toolkit.com/order.php>

Return Value

If the function succeeds, the return value is a reader handle. Otherwise NULL. To get extended error information, check errno value.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTSetLicenseUpgradeKey

Function sets the License upgrade key. This key is used with developer key for access to newer version of runtimes.

```
void SDTSetLicenseUpgradeKey(LPSDBARREADER hReader, const wchar_t* lpLicenseUpgradeKey);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

license

The purchased Upgrade license. Note that upgrade license is used as an extension of Developer License and cannot be used without it.

Discussion

The purchased Developer License give access to all new SDK versions released within next 12 months starting from purchase date. After that period of time it is possible to extend such period for 12 months more by purchasing Upgrade License.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTDestroyBarcodeReader

Function destroy barcode reader instance.

```
void SDTDestroyBarcodeReader(LPSDBARREADER hReader);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

Declared In

sdtbarcode.h

Library

libsdt-brc.so

setReadInputDirections:

The function specifies barcode reading directions the barcode reader will use to scan input image in next call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTSetReadInputDirections(LPSDBARREADER hReader, unsigned long flags);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

flags

Specifies barcode reading directions. This parameter can be one or combination of the following values.

Value	Integer Value	Description
SDTREADDIRECTION_LTR	0x00000001	Left-to-right direction. Used when input image contains normally oriented barcodes.
SDTREADDIRECTION_RTL	0x00000002	Right-to-left direction. Used when input image contains barcodes rotated to 180 degrees.
SDTREADDIRECTION_TTB	0x00000004	Top-to-bottom direction. Used when input image contains barcodes rotated to 90 degrees clockwise.
SDTREADDIRECTION_BTT	0x00000008	Top-to-bottom direction. Used when input image contains barcodes rotated to 90 degrees counter-clockwise.
SDTREADDIRECTION_ALL	0x0000000f	All above direction.

Discussion

In order to achieve better performance it is recommended to set only expected directions.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTSetReadInputTypes:

The method specifies barcode types the Barcode Reader will look for in next call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTSetReadInputTypes(LPSDBARREADER hReader, unsigned long flags);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

flags

Specifies barcode types. This parameter can be one or combination of the following values.

Value	Integer Value	Description
SDTBARCODETYPE_CODABAR	0x00000001	Codabar barcode
SDTBARCODETYPE_CODE128	0x00000002	Code 128 barcode
SDTBARCODETYPE_CODE32	0x00000004	Code 32 (also known as Base 32, Pharma 32/39, and Italian Pharmacode)
SDTBARCODETYPE_CODE39	0x00000008	Code 39 barcode
SDTBARCODETYPE_CODE93	0x00000010	Code 93 barcode
SDTBARCODETYPE_CODE11	0x00000020	Code 11 barcode
SDTBARCODETYPE_EAN13	0x00000040	EAN13 barcode
SDTBARCODETYPE_EAN8	0x00000080	EAN8 barcode
SDTBARCODETYPE_EAN5	0x00000100	EAN5 barcode
SDTBARCODETYPE_I2OF5	0x00000200	Interleaved 2 of 5 barcode
SDTBARCODETYPE_PATCH_CODE	0x00000400	Patch code barcode
SDTBARCODETYPE_POSTNET	0x00000800	Postnet barcode
SDTBARCODETYPE_PLUS2	0x00001000	Plus2 (2-digit supplementals assoc. with EAN and UPC)
SDTBARCODETYPE_PLUS5	0x00002000	Plus5 (5-digit supplementals assoc. with EAN and UPC)
SDTBARCODETYPE_UPCA	0x00004000	UPC-A barcode
SDTBARCODETYPE_UPCB	0x00008000	UPC-B barcode
SDTBARCODETYPE_MSI	0x00100000	MSI barcode
SDTBARCODETYPE_UPCB	0x00200000	UPC-E barcode
SDTBARCODETYPE_PHARMACODE	0x00400000	Pharma code barcode
SDTBARCODETYPE_ALL_1D	0x0070ffff	All above 1-D Barcodes currently supported by SD-TOOLKIT Barcode Reader
SDTBARCODETYPE_INTELLIMAIL	0x00010000	Intelligent Mail barcode
SDTBARCODETYPE_DATAMATRIX	0x00020000	DataMatrix 2D Barcode

SDTBARCODETYPE_PDF417	0x00040000	PDF417 2D Barcode
SDTBARCODETYPE_QRCODE	0x00080000	QR Code 2D Barcode
SDTBARCODETYPE_ALL_2D	0x000f0000	All above 2-D Barcodes currently supported by SD-TOOLKIT Barcode Reader

Discussion

In order to achieve better performance it is recommended to minimize the set of barcode types by specifying only types the final application is interested in..

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTSetActiveScanRect

The function defines a square area within the image where the reader will look for barcodes in next call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTSetActiveScanRect(LPSDBARREADER hReader, int left, int top, int right, int bottom);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

left

[in] positive integer value specified left boundary of the scan area.

top

[in] positive integer value specified top boundary of the scan area.

right

[in] positive integer value specified right boundary of the scan area.

bottom

[in] positive integer value specified bottom boundary of the scan area.

Discussion

If one of the boundary values passed to the function exceeds input image size then image boundary will be used instead.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTReadImageFileA

The function reads the image file located at specified path and recognize barcodes on it. The recognition results values are stored internally and can be obtained later by call to [SDTGetResultValueW](#) or [SDTGetResultValueA](#) function.

```
unsigned long SDTReadImageFileA(LPSDBARREADER hReader, const char* lpFileName, int iPageIndex);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

lpFileName

Path to image or pdf file as ASCII string

iPageIndex

index of page in multipage image file or 0 for single page image files

Return Value

If function succeeds, the return value is zero. If method fails, the return value is nonzero and contain extended error information.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTReadImageFileW

The function reads the image file located at specified path and recognize barcodes on it. The recognition results values are stored internally and can be obtained later by call to [SDTGetResultValueW](#) or [SDTGetResultValueA](#) function.

```
unsigned long SDTReadImageFileW(LPSDBARREADER hReader, const wchar_t* lpFileName, int iPageIndex);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

lpFileName

Path to image or pdf file as Unicode string

iPageIndex

index of page in multipage image file or 0 for single page image files

Return Value

If function succeeds, the return value is zero. If method fails, the return value is nonzero and contain extended error information.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTReadRGBImageBuffer

The function reads RGB image data from allocated memory block and recognize barcodes on it. The recognition results values are stored internally and can be obtained later by call to [SDTGetResultValueW](#) or [SDTGetResultValueA](#) function.

```
- unsigned long SDTReadRGBImageBuffer(LPSDBARREADER hReader,
                                     unsigned char* lpImageBuffer,
                                     int wImageWidth,
                                     int wImageHeight,
                                     int wBpp);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

lpImageBuffer

Pointer to allocated memory block containing RGB image data

wImageWidth

Image width

wImageHeight

Image Height

wBpp

Bpp of the image data. Supported values are 8 - for gray data, 16, 24 and 32 – for color data

Return Value

If function succeeds, the return value is zero. If method fails, the return value is nonzero and contain extended error information.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultsCount

The method returns count of successfully recognized barcodes in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
int SDTGetResultsCount(LPCTSTR hReader);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

Return Value

Returns count of recognized barcodes or 0 in case no barcodes were detected.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultValueA

The function returns pointer to ASCII string with value of the barcode symbol successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
const char* SDTGetResultValueA(LPSDBARREADER hReader, int iPos);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

position

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method

Return Value

Function returns ASCII value of barcode symbol.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultValueW

The function returns pointer to Unicode string with value of the barcode symbol successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
const char* SDTGetResultValueW(LPSDBARREADER hReader, int iPos);
```

Parameters*hReader*Reader handle previously created with [CreateSDTBarcodeReader](#).*position*index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method**Return Value**

Function returns Unicode value of barcode symbol.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultType

The function returns integer value of barcode type successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTGetResultType(LPSDBARREADER hReader, int position);
```

Parameters*hReader*Reader handle previously created with [CreateSDTBarcodeReader](#).*position*index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method**Return Value**

Function returns one of the following possible barcode type flags:

Value	Integer Value	Description
SDTBARCODETYPE_CODABAR	0x00000001	Codabar barcode
SDTBARCODETYPE_CODE128	0x00000002	Code 128 barcode
SDTBARCODETYPE_CODE32	0x00000004	Code 32 (also known as Base 32, Pharma 32/39, and Italian Pharmacode)

SDTBARCODETYPE_CODE39	0x00000008	Code 39 barcode
SDTBARCODETYPE_CODE93	0x00000010	Code 93 barcode
SDTBARCODETYPE_CODE11	0x00000020	Code 11 barcode
SDTBARCODETYPE_EAN13	0x00000040	EAN13 barcode
SDTBARCODETYPE_EAN8	0x00000080	EAN8 barcode
SDTBARCODETYPE_EAN5	0x00000100	EAN5 barcode
SDTBARCODETYPE_I2OF5	0x00000200	Interleaved 2 of 5 barcode
SDTBARCODETYPE_PATCH_CODE	0x00000400	Patch code barcode
SDTBARCODETYPE_POSTNET	0x00000800	Postnet barcode
SDTBARCODETYPE_PLUS2	0x00001000	Plus2 (2-digit supplementals assoc. with EAN and UPC)
SDTBARCODETYPE_PLUS5	0x00002000	Plus5 (5-digit supplementals assoc. with EAN and UPC)
SDTBARCODETYPE_UPCA	0x00004000	UPC-A barcode
SDTBARCODETYPE_UPCB	0x00008000	UPC-B barcode
SDTBARCODETYPE_MS1	0x00100000	MSI barcode
SDTBARCODETYPE_UPCE	0x00200000	UPC-E barcode
SDTBARCODETYPE_PHARMACODE	0x00400000	Pharma code barcode
SDTBARCODETYPE_INTELLIMAIL	0x00010000	Intelligent Mail barcode
SDTBARCODETYPE_DATAMATRIX	0x00020000	DataMatrix 2D Barcode
SDTBARCODETYPE_PDF417	0x00040000	PDF417 2D Barcode
SDTBARCODETYPE_QRCODE	0x00080000	QR Code 2D Barcode

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultTypeNameA

The function returns a name of barcode type as ASCII string successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
const char* SDTGetResultTypeNameA(LPSDBARREADER hReader, int iPos);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) function

Return Value

Function returns Unicode name of barcode type, for example “CODE128” for Code 128 barcode symbol.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultTypeNameW

The function returns a name of barcode type as Unicode string successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
const wchar_t* SDTGetResultTypeNameA(LPSDBARREADER hReader, int iPos);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) function

Return Value

Function returns Unicode name of barcode type, for example “CODE128” for Code 128 barcode symbol.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultReadDirection

The function returns integer value of read direction of the barcode successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTGetResultReadDirection(LPCTSTR hReader, int iPos);
```

Parameters*hReader*

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) function

Return Value

Function returns one of the following possible barcode read direction flags:

Value	Integer Value	Description
SDTREADDIRECTION_LTR	0x00000001	Left-to-right direction. Used when input image contains normally oriented barcodes.
SDTREADDIRECTION_RTL	0x00000002	Right-to-left direction. Used when input image contains barcodes rotated to 180 degrees.
SDTREADDIRECTION_TTB	0x00000004	Top-to-bottom direction. Used when input image contains barcodes rotated to 90 degrees clockwise.
SDTREADDIRECTION_BTT	0x00000008	Top-to-bottom direction. Used when input image contains barcodes rotated to 90 degrees counter-clockwise.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultPositionLeft

The function returns left boundary of the barcode which was successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTGetResultPositionLeft(LPSDBARREADER hReader, int iPos);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method

Return Value

Function returns position in pixels relative to the left top corner of the image.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultPositionTop

The function returns top boundary of the barcode which was successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTGetResultPositionTop(LPSDBARREADER hReader, int iPos);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method

Return Value

Function returns position in pixels relative to the left top corner of the image.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultPositionRight

The function returns right boundary of the barcode which was successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTGetResultPositionRight(LPSDBARREADER hReader, int iPos);
```

Parameters*hReader*

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method

Return Value

Function returns position in pixels relative to the left top corner of the image.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultPositionBottom

The function returns bottom boundary of the barcode which was successfully recognized in the previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTGetResultPositionBottom(LPSDBARREADER hReader, int iPos);
```

Parameters*hReader*

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method

Return Value

Function returns position in pixels relative to the left top corner of the image.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultValueAsBinaryBufferSize

The function returns size of byte buffer of recognition result obtained in previous call to [SDTReadImageFileA](#), [SDTReadImageFileW](#) or [SDTReadRGBImageBuffer](#) function.

```
unsigned long SDTGetResultValueAsBinaryBufferSize(LPCTSTR hReader, int iPos);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method

Return Value

Function returns size in bytes of byte array or zero if buffer is binary value is not available.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so

SDTGetResultValueAsBinaryBuffer

The function returns pointer to allocated memory buffer containing recognized barcode value as byte array.

```
unsigned char* SDTGetResultValueAsBinaryBuffer(LPSDBARREADER hReader, int iPos);
```

Parameters

hReader

Reader handle previously created with [CreateSDTBarcodeReader](#).

iPos

index of recognized barcode result. The count of available barcode results is obtained by call to [SDTGetResultsCount](#) method

Return Value

Function returns pointer to buffer which contains recognized barcode value or null if such buffer is not available.

Availability

Available in Ubuntu 10 and later.

Declared In

sdtbarcode.h

Library

libsdt-brc.so