

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

BBAR v.2.0.00

Bingo Balls Automatic Reader

DLL Function Reference

Table of Contents

Library Initialization	2
<i>bbarInit</i>	2
Image Loading / Setting	3
<i>bbarLoad</i>	3
<i>bbarSet</i>	4
<i>bbarSetRGB24</i>	5
Reading Bingo Balls	6
<i>bbarReadBall</i>	6
<i>bbarReadBallAuto</i>	7
Retrieving Results	8
<i>bbarGetNumber</i>	8
<i>bbarGetConfidenceFactor</i>	9
Storing User Data in the HASP dongle	10
<i>bbarWriteHASP</i>	10
<i>bbarReadHASP</i>	11
Example of Use	12

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

Library Initialization

bbarInit

Initializes the library. It must be executed once and only once before using any other function in the library.

```
bool bbarInit ( long IType );
```

Arguments

IType This value must be 1 for standard balls.
 This value must be 2 for **Tajusa** balls.

Return Value

false → Error.
true → Ok.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

Image Loading / Setting

bbarLoad

Loads an image from a BMP file. The format of the BMP file must be RGB 24 bits.

```
long bbarLoad ( char * strFilename );
```

Arguments

strFilename Filename of the BMP file to be loaded.

Return Value

false → Error.
true → Ok.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

bbarSet

Loads an image from a buffer containing the image pixels. The buffer must contain greyscale values.

```
bool bbarSet ( long IWidth,
              long IHeight,
              unsigned char * pBuffer );
```

Arguments

<i>IWidth</i>	Width of the image in pixels.
<i>IHeight</i>	Height of the image in pixels.
<i>pBuffer</i>	Buffer containing the greyscale pixels of the image (1 byte per pixel).

Return Value

`false` → Error.
`true` → Ok.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

bbarSetRGB24

Loads an image from a buffer containing the image pixels. The buffer must contain 24 bits RGB values.

```
bool bbarSetRGB24 ( long IWidth,
                  long IHeight,
                  unsigned char * pBufferRGB24 );
```

Arguments

<i>IWidth</i>	Width of the image in pixels.
<i>IHeight</i>	Height of the image in pixels.
<i>pBuffer</i>	Buffer containing the colour pixels of the image (3 bytes per pixel).

Return Value

`false` → Error.
`true` → Ok.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

Reading Bingo Balls

bbarReadBall

Analyzes the image and recognizes the ball's number.

```
bool bbarReadBall ( long IRadius,
                  long ICentreX,
                  long ICentreY );
```

Arguments

<i>IRadius</i>	Approximate radius of the ball (in pixels).
<i>ICentreX</i>	Approximate X coordinate of the ball's centre (in pixels).
<i>ICentreY</i>	Approximate Y coordinate of the ball's centre (in pixels).

Return Value

`false` → Error.
`true` → Ok.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

bbarReadBallAuto

Analyzes the image and recognizes the ball's number. This function automatically finds the ball in the image, so there is no need to indicate the ball's centre coordinates.

```
bool bbarReadBallAuto ( long IRadius );
```

Arguments

IRadius Approximate radius of the ball (in pixels).

Return Value

false → Error.
true → Ok.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

Retrieving Results

bbarGetNumber

Retrieves the number of the last bingo ball analyzed by the library.

This function must be called after **bbarReadBall** or **bbarReadBallEx**.

```
void bbarGetNumber ( long * pINumber );
```

Arguments

pINumber Pointer to a variable where the ball's number recognized by **bbarReadBall** or **bbarReadBallEx** will be stored.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

bbarGetConfidenceFactor

Retrieves a confidence factor for the last bingo ball analyzed by the library.

The higher this number, the more reliable will be the result.

This function must be called after **bbarReadBall** or **bbarReadBallEx**.

```
void bbarConfidenceFactor ( float * pfCF );
```

Arguments

pfCF Pointer to a variable where the Confidence Factor will be stored.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

Storing User Data in the HASP dongle

bbarWriteHASP

Writes data in the internal memory of the protection dongle. This capability can be used for any custom purpose. Data is encrypted automatically before being written in the HASP's memory.

A minimum of 16 bytes and a maximum of 24 bytes can be stored.

```
long bbarWriteHASP ( unsigned char * pData,
                    long lSize );
```

Arguments

pData Buffer with the data to be written in the HASP's memory.

lSize Size (in bytes) of the data to write (minimum 16 bytes a maximum 24 bytes).

Return Value

0 → Error.

1 → Ok.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

bbarReadHASP

Reads the data stored in the internal memory of the protection dongle. This capability can be used for any custom purpose. Data is decrypted automatically after being read from the HASP's memory.

```
long bbarReadHASP ( unsigned char * pData,
                  long ISize );
```

Arguments

pData Buffer where the data read from the HASP's memory will be stored.

ISize Size (in bytes) of the data to be read.

Return Value

- 0** → Error.
- 1** → Ok.

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i>	Neural Labs, s.l.
	DLL Function Reference	May, 2007

Example of Use

```

void func ()
{
    bool ok;

    // Library Initialization.
    ok = bbarInit (1); // for standard balls.
    if (ok)
    {
        long INumber;           // Number recognized.
        float fCF;             // Confidence Factor.
        unsigned char* pBuffer; // Greyscale image buffer.
        long IWidth = 640;     // Image's width.
        long IHeight = 480;    // Image's height.
        long IRadius = 156;    // Ball's radius.
        long ICX = 383;        // Ball's X centre.
        long ICY = 210;        // Ball's Y centre.

        // Memory allocation for pBuffer.
        ...

        // Acquire an image from the imaging hardware and store the image buffer in
        // pBuffer (greyscale, 1 byte/pixel).
        ...

        // Setting Image.
        ok = bbarSet (IWidth, IHeight, pBuffer);
        if (ok)
        {
            // Reading Bingo ball.
            ok = bbarReadBall (IRadius, ICX, ICY);
            if (ok)
            {
                // Get Number.
                bbarGetNumber (&INumber);

                // Get Confidence Factor.
                bbarGetConfidenceFactor (&fCF);

                // Process results.
                ...
            }
        }
    }
}

```

 www.neurallabs.net	BBAR v.2.0.00 <i>Bingo Balls Automatic Reader</i> DLL Function Reference	Neural Labs, s.l. <hr/> May, 2007
---	--	--

```
    }  
    else  
    {  
        // Error reading bingo ball.  
    }  
    }  
    else  
    {  
        // Error setting image.  
    }  
    }  
    else  
    {  
        // Error initializing BBAR.  
    }  
}
```