



# mamba RealTime Api Reference Manual

*Author:*  
Nicolas BEUCHER

## Contents

<b>1</b>	<b>Data Structures</b>	<b>5</b>
<b>2</b>	<b>File List</b>	<b>5</b>
<b>3</b>	<b>buffer Struct Reference</b>	<b>5</b>
<b>4</b>	<b>MB_Image Struct Reference</b>	<b>6</b>
4.1	Detailed Description	6
4.2	Field Documentation	6
4.2.1	BITPIX	6
4.2.2	NUMCOL	6
4.2.3	PIXARRAY	6
4.2.4	PLINES	6
<b>5</b>	<b>MBRT_v4l2vidT Struct Reference</b>	<b>8</b>
<b>6</b>	<b>MBRT_v4lvidT Struct Reference</b>	<b>9</b>
<b>7</b>	<b>MBRT_VideoAcq Struct Reference</b>	<b>10</b>
<b>8</b>	<b>MBRT_vidUnion Union Reference</b>	<b>11</b>
<b>9</b>	<b>mambaCommon.h File Reference</b>	<b>12</b>
9.1	Detailed Description	12
9.2	Define Documentation	12
9.2.1	MB_X_LEFT	12
9.2.2	MB_X_RIGHT	13
9.2.3	MB_Y_BOTTOM	13
9.2.4	MB_Y_TOP	13
9.3	Typedef Documentation	13
9.3.1	PIX32	13
9.3.2	PIX8	13
9.3.3	PLINE	13
9.3.4	PLINE32	13
9.3.5	Sint16	13
9.3.6	Sint32	13
9.3.7	Sint8	13
9.3.8	Uint16	13
9.3.9	Uint32	13
9.3.10	Uint8	13
<b>10</b>	<b>mambaRTApi.h File Reference</b>	<b>14</b>
10.1	Detailed Description	14
10.2	Enumeration Type Documentation	14
10.2.1	MBRT_eventcode	14
10.2.2	MBRT_vidType	14
10.3	Function Documentation	15
10.3.1	MBRT_CreateDisplay	15
10.3.2	MBRT_CreateVideoAcq	15
10.3.3	MBRT_DestroyDisplay	15
10.3.4	MBRT_DestroyVideoAcq	15
10.3.5	MBRT_GetAcqSize	15
10.3.6	MBRT_GetImageFromAcq	15
10.3.7	MBRT_PaletteDisplay	16
10.3.8	MBRT_PollDisplay	16
10.3.9	MBRT_UpdateDisplay	16
10.4	Variable Documentation	16
10.4.1	MBRT_sz_x	16

10.4.2	MBRT_sz_y	16
<b>11</b>	<b>mambaRTApi_loc.h File Reference</b>	<b>17</b>
11.1	Detailed Description	17
11.2	Function Documentation	18
11.2.1	MBRT_CreateVideoAcq_v4l	18
11.2.2	MBRT_CreateVideoAcq_v4l2	18
11.2.3	MBRT_DestroyVideoAcq_v4l	18
11.2.4	MBRT_DestroyVideoAcq_v4l2	18
11.2.5	MBRT_GetAcqSize_v4l	18
11.2.6	MBRT_GetAcqSize_v4l2	18
11.2.7	MBRT_GetImageFromAcq_v4l	19
11.2.8	MBRT_GetImageFromAcq_v4l2	19
11.3	Variable Documentation	19
11.3.1	color_palette	19
11.3.2	isPalettized	19
11.3.3	MBRT_acqDevice	19
11.3.4	MBRT_screen	19
11.3.5	standard_palette	19
<b>12</b>	<b>MBRT_Display.c File Reference</b>	<b>20</b>
12.1	Detailed Description	20
12.2	Function Documentation	20
12.2.1	MBRT_CreateDisplay	20
12.2.2	MBRT_DestroyDisplay	21
12.2.3	MBRT_PaletteDisplay	21
12.2.4	MBRT_PollDisplay	21
12.2.5	MBRT_UpdateDisplay	21
12.3	Variable Documentation	21
12.3.1	color_palette	21
12.3.2	isPalettized	21
12.3.3	MBRT_screen	21
12.3.4	MBRT_sz_x	21
12.3.5	MBRT_sz_y	21
12.3.6	standard_palette	22
<b>13</b>	<b>MBRT_error.c File Reference</b>	<b>23</b>
13.1	Detailed Description	23
13.2	Function Documentation	23
13.2.1	MBRT_StrErr	23
13.3	Variable Documentation	23
13.3.1	err_str	23
<b>14</b>	<b>MBRT_error.h File Reference</b>	<b>24</b>
14.1	Detailed Description	24
14.2	Enumeration Type Documentation	24
14.2.1	MBRT_errcode	24
14.3	Function Documentation	24
14.3.1	MBRT_StrErr	24
<b>15</b>	<b>MBRT_VideoAcq.c File Reference</b>	<b>25</b>
15.1	Detailed Description	25
15.2	Function Documentation	25
15.2.1	MBRT_CreateVideoAcq	25
15.2.2	MBRT_DestroyVideoAcq	25
15.2.3	MBRT_GetAcqSize	25
15.2.4	MBRT_GetImageFromAcq	26
15.3	Variable Documentation	26
15.3.1	MBRT_acqDevice	26

<b>16 MBRT_VideoAcq_v4l.c File Reference</b>	<b>27</b>
16.1 Detailed Description . . . . .	27
16.2 Function Documentation . . . . .	27
16.2.1 MBRT_CreateVideoAcq_v4l . . . . .	27
16.2.2 MBRT_DestroyVideoAcq_v4l . . . . .	27
16.2.3 MBRT_GetAcqSize_v4l . . . . .	27
16.2.4 MBRT_GetImageFromAcq_v4l . . . . .	27
<b>17 MBRT_VideoAcq_v4l2.c File Reference</b>	<b>28</b>
17.1 Detailed Description . . . . .	28
17.2 Function Documentation . . . . .	28
17.2.1 MBRT_CreateVideoAcq_v4l2 . . . . .	28
17.2.2 MBRT_DestroyVideoAcq_v4l2 . . . . .	28
17.2.3 MBRT_GetAcqSize_v4l2 . . . . .	28
17.2.4 MBRT_GetImageFromAcq_v4l2 . . . . .	28

## List of Figures

1	Image structure and variables . . . . .	7
---	---	---

Data Structure Index

## 1 Data Structures

Here are the data structures with brief descriptions:

<b>buffer</b>	<b>5</b>
<b>MB_Image</b>	<b>6</b>
<b>MBRT_v4l2vidT</b>	<b>8</b>
<b>MBRT_v4lvidT</b>	<b>9</b>
<b>MBRT_VideoAcq</b>	<b>10</b>
<b>MBRT_vidUnion</b>	<b>11</b>
File Index	

## 2 File List

Here is a list of all documented files with brief descriptions:

<b>mambaCommon.h</b>	<b>12</b>
<b>mambaRTApi.h</b>	<b>14</b>
<b>mambaRTApi_loc.h</b>	<b>17</b>
<b>MBRT_Display.c</b>	<b>20</b>
<b>MBRT_error.c</b>	<b>23</b>
<b>MBRT_error.h</b>	<b>24</b>
<b>MBRT_VideoAcq.c</b>	<b>25</b>
<b>MBRT_VideoAcq_v4l.c</b>	<b>27</b>
<b>MBRT_VideoAcq_v4l2.c</b>	<b>28</b>
Data Structure Documentation	

## 3 buffer Struct Reference

### Data Fields

- void \* **start**
- size\_t **length**

The documentation for this struct was generated from the following file:

- **mambaRTApi\_loc.h**

## 4 MB\_Image Struct Reference

```
#include <mambaCommon.h>
```

### Data Fields

- unsigned int **BITPPIX**
- unsigned int **NUMCOL**
- **PLINE \* PLINES**
- **PIX8 \* PIXARRAY**

### 4.1 Detailed Description

Images structure with the depth (BITPPIX); the pixels array (PIXARRAY) and entry point array to each line of the image (PLINES)

### 4.2 Field Documentation

#### 4.2.1 unsigned int MB\_Image::BITPPIX

The depth of the image

#### 4.2.2 unsigned int MB\_Image::NUMCOL

number of colors used

#### 4.2.3 PIX8\* MB\_Image::PIXARRAY

pixel array

#### 4.2.4 PLINE\* MB\_Image::PLINES

accessors to pixel lines

The documentation for this struct was generated from the following file:

- **mambaCommon.h**

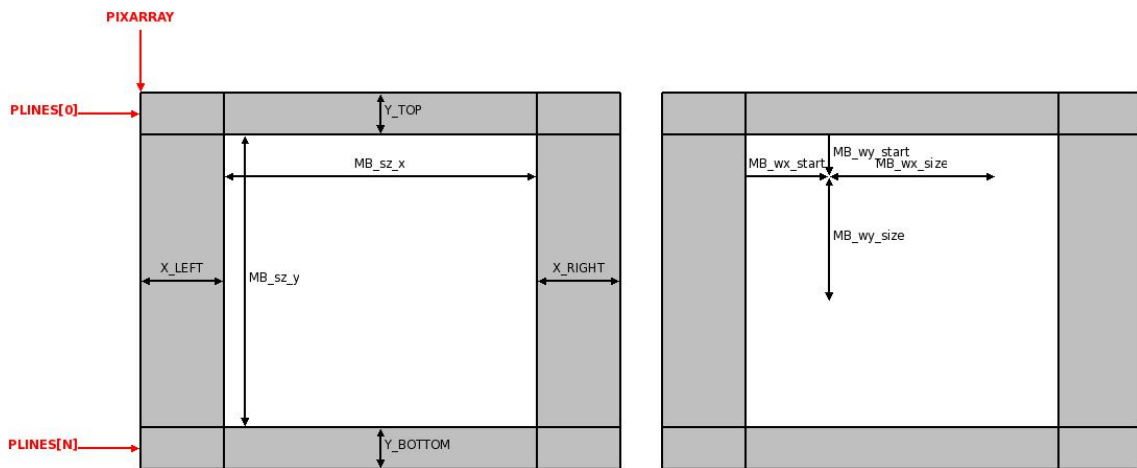


Figure 1: Image structure and variables

## 5 MBRT\_v4l2vidT Struct Reference

### Data Fields

- struct **buffer** \* **buffers**
- unsigned int **n\_buffers**
- int **w**
- int **h**

The documentation for this struct was generated from the following file:

- **mambaRTapi\_loc.h**

## 6 MBRT\_v4lvidT Struct Reference

### Data Fields

- **PIX8 \* FRAMEBUFFER**
- struct video\_capability **vcap**
- struct video\_window **vwin**
- struct video\_picture **vpic**
- struct video\_mmap **vmmmap**
- struct video\_mbuf **vmbuf**
- struct video\_channel **vchan**

The documentation for this struct was generated from the following file:

- **mambaRTapi\_loc.h**

## 7 MBRT\_VideoAcq Struct Reference

### Data Fields

- **MBRT\_vidType** type
- **int** fd
- **MBRT\_vidUnion** video

The documentation for this struct was generated from the following file:

- **mambaRTApi\_loc.h**

## 8 MBRT\_\_vidUnion Union Reference

### Data Fields

- **MBRT\_\_v4lvidT v4l**
- **MBRT\_\_v4l2vidT v4l2**

The documentation for this union was generated from the following file:

- **mambaRTapi\_loc.h**

File Documentation

## 9 mambaCommon.h File Reference

```
#include <stdint.h>
```

### Data Structures

- struct `MB_Image`

### Defines

- `#define MB_X_LEFT(im) X_LEFT`
- `#define MB_X_RIGHT(im) X_RIGHT`
- `#define MB_Y_TOP(im) Y_TOP`
- `#define MB_Y_BOTTOM(im) Y_BOTTOM`

### Typedefs

- `typedef uint8_t UInt8`
- `typedef uint16_t UInt16`
- `typedef uint32_t UInt32`
- `typedef int8_t Sint8`
- `typedef int16_t Sint16`
- `typedef int32_t Sint32`
- `typedef uint8_t PIX8`
- `typedef PIX8 * PLINE`
- `typedef int32_t PIX32`
- `typedef PIX32 * PLINE32`

### 9.1 Detailed Description

#### Date:

31-03-2009

This file contains the various definitions, macro, struct that are commons between the various modules of the library

The copyright license of Mamba is reminded here :

Copyright (c) <2009>, <Nicolas BEUCHER and ARMINES for the Centre de Morphologie Mathématique(CMM), common research center to ARMINES and MINES Paristech>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Except as contained in this notice, the names of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without their prior written authorization.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 9.2 Define Documentation

#### 9.2.1 `#define MB_X_LEFT(im) X_LEFT`

Getting image frame offset from left

**9.2.2 #define MB\_X\_RIGHT(im) X\_RIGHT**

Getting image frame offset from right

**9.2.3 #define MB\_Y\_BOTTOM(im) Y\_BOTTOM**

Getting image frame offset from bottom

**9.2.4 #define MB\_Y\_TOP(im) Y\_TOP**

Getting image frame offset from top

**9.3 Typedef Documentation****9.3.1 typedef int32\_t PIX32**

Signed 32-bit pixels value type

**9.3.2 typedef uint8\_t PIX8**

grey-scale pixels value type

**9.3.3 typedef PIX8\* PLINE**

Pixels line pointers type

**9.3.4 typedef PIX32\* PLINE32**

32-bit pixels line pointers type

**9.3.5 typedef int16\_t Sint16**

Signed 16 bit value type

**9.3.6 typedef int32\_t Sint32**

Signed 32 bit value type

**9.3.7 typedef int8\_t Sint8**

Signed 8 bit value type

**9.3.8 typedef uint16\_t Uint16**

Unsigned 16 bit value type

**9.3.9 typedef uint32\_t Uint32**

Unsigned 32 bit value type

**9.3.10 typedef uint8\_t Uint8**

Unsigned 8 bit value type

## 10 mambaRTApi.h File Reference

```
#include "mambaCommon.h"
#include "MBRT_error.h"
```

### Enumerations

- enum MBRT\_eventcode { NO\_EVENT, EVENT\_CLOSE, EVENT\_PROCESS }
- enum MBRT\_vidType { V4L\_TYPE, V4L2\_TYPE }

### Functions

- MBRT\_errcode MBRT\_CreateVideoAcq (char \*device, MBRT\_vidType type)
- MBRT\_errcode MBRT\_DestroyVideoAcq (void)
- MBRT\_errcode MBRT\_GetAcqSize (int \*acq\_w, int \*acq\_h)
- MBRT\_errcode MBRT\_GetImageFromAcq (MB\_Image \*dest)
- MBRT\_errcode MBRT\_CreateDisplay (int w, int h)
- MBRT\_errcode MBRT\_DestroyDisplay (void)
- MBRT\_errcode MBRT\_UpdateDisplay (MB\_Image \*src, double wfps, double \*ofps)
- MBRT\_errcode MBRT\_PaletteDisplay (UInt8 \*palette)
- MBRT\_errcode MBRT\_PollDisplay (MBRT\_eventcode \*event\_code)

### Variables

- UInt32 MBRT\_sz\_x
- UInt32 MBRT\_sz\_y

### 10.1 Detailed Description

#### Date:

03-27-2009

This file contains the various definitions, global variables macro, struct and functions created for the library.

### 10.2 Enumeration Type Documentation

#### 10.2.1 enum MBRT\_eventcode

Type definition for display event code

##### Enumerator:

- NO\_EVENT** No events
- EVENT\_CLOSE** Close event occure when the user close the window or press esc inside it
- EVENT\_PROCESS** Event to toggle on/off the process

#### 10.2.2 enum MBRT\_vidType

Type definition for the acquisition device

##### Enumerator:

- V4L\_TYPE** Video 4 Linux 1 API type device
- V4L2\_TYPE** Video 4 Linux 2 API type device

## 10.3 Function Documentation

### 10.3.1 MBRT\_errcode MBRT\_CreateDisplay (int *w*, int *h*)

Initialize SDL and creates the video display (SDL screen)

**Parameters:**

*w* width of the display (resolution)

*h* height of the display (resolution)

**Returns:**

An error code (NO\_ERR if successful)

### 10.3.2 MBRT\_errcode MBRT\_CreateVideoAcq (char \* *device*, MBRT\_vidType *type*)

Filled the video acquisition structure with the parameters of the given device and initialize it.

**Parameters:**

*device* the video device (usually /dev/video0)

*type* the type of device (e.g video4linux, video4linux2 ...)

**Returns:**

an error code (NO\_ERR if successful)

### 10.3.3 MBRT\_errcode MBRT\_DestroyDisplay (void)

Destroy the video display (SDL screen) and quit SDL

**Returns:**

An error code (NO\_ERR if successful)

### 10.3.4 MBRT\_errcode MBRT\_DestroyVideoAcq (void)

Close the acquisition device and reset the structure

**Returns:**

NO\_ERR if successful

### 10.3.5 MBRT\_errcode MBRT\_GetAcqSize (int \* *acq\_w*, int \* *acq\_h*)

Returns the acquisition device resolution.

**Parameters:**

*acq\_w* the width (output)

*acq\_h* the height (output)

**Returns:**

NO\_ERR if successful

### 10.3.6 MBRT\_errcode MBRT\_GetImageFromAcq (MB\_Image \* *dest*)

Obtains an image form the acquisition device

**Parameters:**

*dest* the mamba image filled by the device

**Returns:**

NO\_ERR if successful

### 10.3.7 MBRT\_errcode MBRT\_PaletteDisplay (Uint8 \* *palette*)

Change the palette associated with the display

**Parameters:**

*palette* an array containing the complete palette definition (256\*3 integers)

**Returns:**

An error code (NO\_ERR if successful)

### 10.3.8 MBRT\_errcode MBRT\_PollDisplay (MBRT\_eventcode \* *event\_code*)

Handles event that have occurred in the display

**Parameters:**

*event\_code* an integer representing a specific event (output)

**Returns:**

An error code (NO\_ERR if successful)

### 10.3.9 MBRT\_errcode MBRT\_UpdateDisplay (MB\_Image \* *src*, double *wfps*, double \* *ofps*)

Update the display with the content of a given mamba image structure

**Parameters:**

*src* the image displayed

*wfps* input the desired framerate

*ofps* output the framerate

**Returns:**

An error code (NO\_ERR if successful)

## 10.4 Variable Documentation

### 10.4.1 Uint32 MBRT\_sz\_x

Width of the display

### 10.4.2 Uint32 MBRT\_sz\_y

Height of the display

## 11 mambaRTApi\_loc.h File Reference

```
#include "mambaRTApi.h"
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <unistd.h>
#include <string.h>
#include <libv4l2.h>
#include <linux/videodev.h>
#include <linux/videodev2.h>
#include <sys/mman.h>
#include <sys/file.h>
#include <sys/ioctl.h>
#include <SDL/SDL.h>
```

### Data Structures

- struct **MBRT\_v4lvidT**
- struct **buffer**
- struct **MBRT\_v4l2vidT**
- union **MBRT\_vidUnion**
- struct **MBRT\_VideoAcq**

### Functions

- **MBRT\_errcode MBRT\_CreateVideoAcq\_v4l** (char \*device)
- **MBRT\_errcode MBRT\_DestroyVideoAcq\_v4l** (void)
- **MBRT\_errcode MBRT\_GetAcqSize\_v4l** (int \*acq\_w, int \*acq\_h)
- **MBRT\_errcode MBRT\_GetImageFromAcq\_v4l** (**MB\_Image** \*dest)
- **MBRT\_errcode MBRT\_CreateVideoAcq\_v4l2** (char \*device)
- **MBRT\_errcode MBRT\_DestroyVideoAcq\_v4l2** (void)
- **MBRT\_errcode MBRT\_GetAcqSize\_v4l2** (int \*acq\_w, int \*acq\_h)
- **MBRT\_errcode MBRT\_GetImageFromAcq\_v4l2** (**MB\_Image** \*dest)

### Variables

- **MBRT\_VideoAcq \* MBRT\_acqDevice**
- **SDL\_Surface \* MBRT\_screen**
- **SDL\_Color color\_palette** [256]
- **SDL\_Color standard\_palette** [256]
- **Uint32 isPalettized**

### 11.1 Detailed Description

#### Date:

03-27-2009

This file contains the various definitions, global variables macro, struct and functions that are shared between components of the library but are not meant to be exported to the outside world.

## 11.2 Function Documentation

### 11.2.1 `MBRT_errcode MBRT_CreateVideoAcq_v4l (char * device)`

Filled the video acquisition (V4L) structure with the parameters of the given device and initialize it.

**Parameters:**

*device* the video device (usually /dev/video0)

**Returns:**

an error code (NO\_ERR if successful)

### 11.2.2 `MBRT_errcode MBRT_CreateVideoAcq_v4l2 (char * device)`

Filled the video acquisition (V4L2) structure with the parameters of the given device and initialize it.

**Parameters:**

*device* the video device (usually /dev/video0)

**Returns:**

an error code (NO\_ERR if successful)

### 11.2.3 `MBRT_errcode MBRT_DestroyVideoAcq_v4l (void)`

Close the acquisition device (V4L) and reset the structure

**Returns:**

NO\_ERR if successful

### 11.2.4 `MBRT_errcode MBRT_DestroyVideoAcq_v4l2 (void)`

Close the acquisition device (V4L2) and reset the structure

**Returns:**

NO\_ERR if successful

### 11.2.5 `MBRT_errcode MBRT_GetAcqSize_v4l (int * acq_w, int * acq_h)`

Returns the acquisition device resolution (V4L).

**Parameters:**

*acq\_w* the width (output)

*acq\_h* the height (output)

**Returns:**

NO\_ERR if successful

### 11.2.6 `MBRT_errcode MBRT_GetAcqSize_v4l2 (int * acq_w, int * acq_h)`

Returns the acquisition device resolution (V4L2).

**Parameters:**

*acq\_w* the width (output)

*acq\_h* the height (output)

**Returns:**

NO\_ERR if successful

**11.2.7 MBRT\_errcode MBRT\_GetImageFromAcq\_v4l (MB\_Image \* *dest*)**

Obtains an image from the acquisition device (V4L)

**Parameters:**

*dest* the mamba image filled by the device

**Returns:**

NO\_ERR if successful

**11.2.8 MBRT\_errcode MBRT\_GetImageFromAcq\_v4l2 (MB\_Image \* *dest*)**

Obtains an image from the acquisition device (V4L2)

**Parameters:**

*dest* the mamba image filled by the device

**Returns:**

NO\_ERR if successful

**11.3 Variable Documentation****11.3.1 SDL\_Color color\_palette[256]**

Color palette associated to the screen

**11.3.2 Uint32 isPalettized**

indicates if the display use the color palette

**11.3.3 MBRT\_VideoAcq\* MBRT\_acqDevice**

Structure holding the acquisition device information

**11.3.4 SDL\_Surface\* MBRT\_screen**

pointer to the SDL surface (display screen)

**11.3.5 SDL\_Color standard\_palette[256]**

Standard greyscale palette associated to the screen

## 12 MBRT\_Display.c File Reference

```
#include "mambaRTapi_loc.h"
```

### Defines

- #define **MBRT\_TITLE** "Mamba RealTime"
- #define **FRAME\_COLOR** 0xffffffff
- #define **FPS\_VALUE\_COLOR** 0xffffffff
- #define **FPS\_THICKNESS** 3
- #define **FPS\_MEAN\_SIZE** 20
- #define **HISTO\_BLACKENING** 60
- #define **HISTO\_COLOR** 0xffffffff

### Functions

- **MBRT\_errcode** **MBRT\_CreateDisplay** (int *w*, int *h*)
- **MBRT\_errcode** **MBRT\_DestroyDisplay** ()
- **MBRT\_errcode** **MBRT\_UpdateDisplay** (**MB\_Image** \**src*, double *wfps*, double \**ofps*)
- **MBRT\_errcode** **MBRT\_PaletteDisplay** (**Uint8** \**palette*)
- **MBRT\_errcode** **MBRT\_PollDisplay** (**MBRT\_eventcode** \**event\_code*)

### Variables

- **SDL\_Surface** \* **MBRT\_screen** = NULL
- **Uint32** **MBRT\_sz\_x** = 0
- **Uint32** **MBRT\_sz\_y** = 0
- **SDL\_Color** **color\_palette** [256]
- **SDL\_Color** **standard\_palette** [256]
- **Uint32** **isPalettized**
- **Uint32** **last\_call**
- **Uint32** **isFpsDisplayed**
- double **old\_fps** [**FPS\_MEAN\_SIZE**]
- int **index\_fps**
- **Uint32** **histo** [256]
- **Uint32** **isHistoDisplayed**

### 12.1 Detailed Description

#### Author:

Nicolas Beucher

#### Date:

03-29-2009

### 12.2 Function Documentation

#### 12.2.1 **MBRT\_errcode** **MBRT\_CreateDisplay** (int *w*, int *h*)

Initialize SDL and creates the video display (SDL screen)

#### Parameters:

- w* width of the display (resolution)
- h* height of the display (resolution)

#### Returns:

An error code (**NO\_ERR** if successful)

### 12.2.2 MBRT\_errcode MBRT\_DestroyDisplay (void)

Destroy the video display (SDL screen) and quit SDL

**Returns:**

An error code (NO\_ERR if successful)

### 12.2.3 MBRT\_errcode MBRT\_PaletteDisplay (Uint8 \* *palette*)

Change the palette associated with the display

**Parameters:**

*palette* an array containing the complete palette definition (256\*3 integers)

**Returns:**

An error code (NO\_ERR if successful)

### 12.2.4 MBRT\_errcode MBRT\_PollDisplay (MBRT\_eventcode \* *event\_code*)

Handles event that have occurred in the display

**Parameters:**

*event\_code* an integer representing a specific event (output)

**Returns:**

An error code (NO\_ERR if successful)

### 12.2.5 MBRT\_errcode MBRT\_UpdateDisplay (MB\_Image \* *src*, double *wfps*, double \* *ofps*)

Update the display with the content of a given mamba image structure

**Parameters:**

*src* the image displayed

*wfps* input the desired framerate

*ofps* output the framerate

**Returns:**

An error code (NO\_ERR if successful)

## 12.3 Variable Documentation

### 12.3.1 SDL\_Color color\_palette[256]

Color palette associated to the screen

### 12.3.2 Uint32 isPalettized

indicates if the display use the color palette

### 12.3.3 SDL\_Surface\* MBRT\_screen = NULL

pointer to the SDL surface (display screen)

### 12.3.4 Uint32 MBRT\_sz\_x = 0

Width of the display

### 12.3.5 Uint32 MBRT\_sz\_y = 0

Height of the display

### 12.3.6 `SDL_Color standard_palette[256]`

Standard greyscale palette associated to the screen

## 13 MBRT\_error.c File Reference

```
#include "MBRT_error.h"
```

### Functions

- char \* MBRT\_StrErr (MBRT\_errcode error\_nb)

### Variables

- char \* err\_str []

### 13.1 Detailed Description

#### Author:

Nicolas Beucher

#### Date:

3-28-2009

### 13.2 Function Documentation

#### 13.2.1 char\* MBRT\_StrErr (MBRT\_errcode *error\_nb*)

Returns an explanation of the error code

### 13.3 Variable Documentation

#### 13.3.1 char\* err\_str[]

##### Initial value:

```
{  
  "No error",  
  "Initialize display error (SDL)",  
  "Locking screen for updating failure",  
  "Acquisition device error",  
  "Cannot open video device",  
  "Not a video for linux device",  
  "Not a video for linux 2 device",  
  "Device does not support streaming",  
  "Cannot obtain device resolution capabilities",  
  "Cannot set device format",  
  "Unsupported palette format",  
  "Type of the video acquisition is incorrect",  
  "Incompatible depth for realtime acquisition/display"  
}
```

Error value interpretation

## 14 MBRT\_error.h File Reference

### Enumerations

- enum MBRT\_errcode {  
NO\_ERR, ERR\_INIT\_DISPLAY, ERR\_LOCK\_DISPLAY, ERR\_VID,  
ERR\_OPEN\_VID, ERR\_V4L\_VID, ERR\_V4L2\_VID, ERR\_STRM\_VID,  
ERR\_CAP\_VID, ERR\_FMT\_VID, ERR\_PAL\_VID, ERR\_VID\_TYPE,  
ERR\_DEPTH }

### Functions

- char \* MBRT\_StrErr (MBRT\_errcode error\_nb)

#### 14.1 Detailed Description

##### Date:

3-28-2009

This file contains the complete liste of error code returned by the API functions.

#### 14.2 Enumeration Type Documentation

##### 14.2.1 enum MBRT\_errcode

Type definition for error code

##### Enumerator:

- NO\_ERR** Value returned by function when no error was encountered.
- ERR\_INIT\_DISPLAY** Init SDL display error
- ERR\_LOCK\_DISPLAY** Locking screen for updating failure
- ERR\_VID** Video acquisition module error
- ERR\_OPEN\_VID** Cannot open video device
- ERR\_V4L\_VID** Not a video for linux device
- ERR\_V4L2\_VID** Not a video for linux 2 device
- ERR\_STRM\_VID** Device does not support streaming
- ERR\_CAP\_VID** Cannot obtain device resolution capabilities
- ERR\_FMT\_VID** Cannot set device format
- ERR\_PAL\_VID** Unsupported palette format
- ERR\_VID\_TYPE** Type of the video acquisition is incorrect
- ERR\_DEPTH** The depth of the mamba image given in argument is incompatible

#### 14.3 Function Documentation

##### 14.3.1 char\* MBRT\_StrErr (MBRT\_errcode error\_nb)

Returns an explanation of the error code

## 15 MBRT\_VideoAcq.c File Reference

```
#include "mambaRTapi_loc.h"
```

### Functions

- `MBRT_errcode MBRT_CreateVideoAcq (char *device, MBRT_vidType type)`
- `MBRT_errcode MBRT_DestroyVideoAcq ()`
- `MBRT_errcode MBRT_GetAcqSize (int *acq_w, int *acq_h)`
- `MBRT_errcode MBRT_GetImageFromAcq (MB_Image *dest)`

### Variables

- `MBRT_VideoAcq * MBRT_acqDevice = NULL`

### 15.1 Detailed Description

#### Author:

Nicolas Beucher

#### Date:

03-27-2009

### 15.2 Function Documentation

#### 15.2.1 `MBRT_errcode MBRT_CreateVideoAcq (char * device, MBRT_vidType type)`

Filled the video acquisition structure with the parameters of the given device and initialize it.

#### Parameters:

*device* the video device (usually /dev/video0)

*type* the type of device (e.g video4linux, video4linux2 ...)

#### Returns:

an error code (NO\_ERR if successful)

#### 15.2.2 `MBRT_errcode MBRT_DestroyVideoAcq (void)`

Close the acquisition device and reset the structure

#### Returns:

NO\_ERR if successful

#### 15.2.3 `MBRT_errcode MBRT_GetAcqSize (int * acq_w, int * acq_h)`

Returns the acquisition device resolution.

#### Parameters:

*acq\_w* the width (output)

*acq\_h* the height (output)

#### Returns:

NO\_ERR if successful

#### 15.2.4 MBRT\_errcode MBRT\_GetImageFromAcq (MB\_Image \* *dest*)

Obtains an image from the acquisition device

**Parameters:**

*dest* the mamba image filled by the device

**Returns:**

NO\_ERR if successful

### 15.3 Variable Documentation

#### 15.3.1 MBRT\_VideoAcq\* MBRT\_acqDevice = NULL

Structure holding the acquisition device information

## 16 MBRT\_VideoAcq\_v4l.c File Reference

```
#include "mambaRTapi_loc.h"
```

### Functions

- `MBRT_errcode MBRT_CreateVideoAcq_v4l (char *device)`
- `MBRT_errcode MBRT_DestroyVideoAcq_v4l ()`
- `MBRT_errcode MBRT_GetAcqSize_v4l (int *acq_w, int *acq_h)`
- `MBRT_errcode MBRT_GetImageFromAcq_v4l (MB_Image *dest)`

### 16.1 Detailed Description

#### Author:

Nicolas Beucher

#### Date:

04-07-2009

### 16.2 Function Documentation

#### 16.2.1 `MBRT_errcode MBRT_CreateVideoAcq_v4l (char * device)`

Filled the video acquisition (V4L) structure with the parameters of the given device and initialize it.

#### Parameters:

*device* the video device (usually /dev/video0)

#### Returns:

an error code (NO\_ERR if successful)

#### 16.2.2 `MBRT_errcode MBRT_DestroyVideoAcq_v4l (void)`

Close the acquisition device (V4L) and reset the structure

#### Returns:

NO\_ERR if successful

#### 16.2.3 `MBRT_errcode MBRT_GetAcqSize_v4l (int * acq_w, int * acq_h)`

Returns the acquisition device resolution (V4L).

#### Parameters:

*acq\_w* the width (output)

*acq\_h* the height (output)

#### Returns:

NO\_ERR if successful

#### 16.2.4 `MBRT_errcode MBRT_GetImageFromAcq_v4l (MB_Image * dest)`

Obtains an image from the acquisition device (V4L)

#### Parameters:

*dest* the mamba image filled by the device

#### Returns:

NO\_ERR if successful

## 17 MBRT\_VideoAcq\_v4l2.c File Reference

```
#include "mambaRTapi_loc.h"
```

### Defines

- #define **CLEAR**(x) `memset (&(x), 0, sizeof (x))`

### Functions

- **MBRT\_errcode MBRT\_CreateVideoAcq\_v4l2** (`char *device`)
- **MBRT\_errcode MBRT\_DestroyVideoAcq\_v4l2** ()
- **MBRT\_errcode MBRT\_GetAcqSize\_v4l2** (`int *acq_w, int *acq_h`)
- **MBRT\_errcode MBRT\_GetImageFromAcq\_v4l2** (`MB_Image *dest`)

### 17.1 Detailed Description

#### Author:

Nicolas Beucher

#### Date:

04-07-2009

### 17.2 Function Documentation

#### 17.2.1 MBRT\_errcode MBRT\_CreateVideoAcq\_v4l2 (`char * device`)

Filled the video acquisition (V4L2) structure with the parameters of the given device and initialize it.

##### Parameters:

*device* the video device (usually /dev/video0)

##### Returns:

an error code (NO\_ERR if successful)

#### 17.2.2 MBRT\_errcode MBRT\_DestroyVideoAcq\_v4l2 (`void`)

Close the acquisition device (V4L2) and reset the structure

##### Returns:

NO\_ERR if successful

#### 17.2.3 MBRT\_errcode MBRT\_GetAcqSize\_v4l2 (`int * acq_w, int * acq_h`)

Returns the acquisition device resolution (V4L2).

##### Parameters:

*acq\_w* the width (output)

*acq\_h* the height (output)

##### Returns:

NO\_ERR if successful

#### 17.2.4 MBRT\_errcode MBRT\_GetImageFromAcq\_v4l2 (`MB_Image * dest`)

Obtains an image from the acquisition device (V4L2)

##### Parameters:

*dest* the mamba image filled by the device

##### Returns:

NO\_ERR if successful