MangoX

Table of Contents

MangoX

1 Introduction

Please refer to the MangoX User Manual for a full introduction to MangoX and its capabilities. This document is a reference guide to the host side of the library.

2 Symbol Reference

Files

File	Description
MangoXExp.h (see page 10)	Copyright 2005 Mango DSP, Inc. All rights reserved. Software and information contained herein is confidential and proprietary to Mango DSP, and any unauthorized copying, dissemination or use is strictly prohibited. Filename: MangoXExp.h General Description: Exported header for MangoX library Created: 21/2/2005 Author: Itay Chamiel

Functions

Function	Description
MangoX_CardDisable (see page 3)	Frees all resources associated with a previous successful call to MangoX_CardEnable (see page 3)() and disables all enabled DSPs.
MangoX_CardEnable (see page 3)	Grant library permission to use a card and some (or all) of its DSPs; initializes DSPs.
MangoX_Close (see page 4)	Closes access to the library and frees all associated resources.
MangoX_ConfigVideo (see page 5)	Configure video chips.
MangoX_DeleteGraph (see page 6)	Closes a graph on a DSP.
MangoX_DspCustomCommand (see page 6)	Send user-definable command and parameter to DSP (regardless of whether there are any open graphs).
MangoX_GetVersion (see page 7)	Returns lib version.
MangoX_GetVersionString (see page 7)	Receive version information string from DSP.
MangoX_Open (see page 8)	Opens access to the library.
MangoX_SubmitGraph (see page 8)	Opens a stream on a dsp.

Legend

	Method

Types

Type	Description
MANGOX_ERR_CB (see page 9)	This is a prototype for a user callback function that is called whenever a DSP reports an error.

2.1 Functions

The following table lists functions in this documentation.

Functions

Function	Description
MangoX_CardDisable (see page 3)	Frees all resources associated with a previous successful call to MangoX_CardEnable (see page 3)() and disables all enabled DSPs.
MangoX_CardEnable (see page 3)	Grant library permission to use a card and some (or all) of its DSPs; initializes DSPs.
MangoX_Close (see page 4)	Closes access to the library and frees all associated resources.
MangoX_ConfigVideo (see page 5)	Configure video chips.
MangoX_DeleteGraph (see page 6)	Closes a graph on a DSP.
MangoX_DspCustomCommand (see page 6)	Send user-definable command and parameter to DSP (regardless of whether there are any open graphs).
MangoX_GetVersion (see page 7)	Returns lib version.

MangoX_GetVersionString (see page 7)	Receive version information string from DSP.
MangoX_Open (see page 8)	Opens access to the library.
MangoX_SubmitGraph (see page 8)	Opens a stream on a dsp.

Legend

	Method

2.1.1 MangoX_CardDisable Function

```
MANGOERROR_error_t MangoX_CardDisable(IN int card_num);
```

Summary

Frees all resources associated with a previous successful call to MangoX_CardEnable (see page 3)() and disables all enabled DSPs.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
`IN int card_num`	the card number that was returned by MangoX_CardEnable (see page 3).

Returns

- MANGOERROR_ERR_INVALID_PARAMETER - Invalid card number
- MANGOERROR_FAILURE - Failure communicating with DSP or closing PCI streams.
- MANGOERROR_RESOURCE_NOT_READY - DSP reported failure when attempting to communicate.
- MANGOERROR_INSUFFICIENT_RESOURCES - Failure freeing shared memory.
- MANGOERROR_SUCCESS - Success freeing resources.

Remarks

- Calling this function without first deleting all graphs will result in memory leaks. The library does not keep track of open graphs and cannot close them for you.
- After calling this function the card number is invalid.
- All DSPs that were enabled for MangoX on this card are now reset and free for use by your application. (You must load them if you wish to use them.)

2.1.2 MangoX_CardEnable Function

```
MANGOERROR_error_t MangoX_CardEnable(IN MangoC64Boards_handle_t * handle, IN char**
coff_fnames, IN char** fpga_fnames, OUT int* card_num, OUT h2d_pci_device_t *** pciDevArr,
IN MANGOX_ERR_CB * err_cbs, IN void ** err_args);
```

Summary

Grant library permission to use a card and some (or all) of its DSPs; initializes DSPs.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
`IN MangoC64Boards_handle_t * handle`	Handle to the card to be used. May be of any valid Mango card type.
`IN char** coff_fnames`	Array indicating which DSPs may be used, and if so, which executable (out file) to load. The length of the array should be equal to the number of DSPs on the card, and each entry should contain a character string containing the path and file name to load, or NULL if the DSP is not to be accessed by the library at all.
`IN char** fpga_fnames`	Array of FPGA filenames. The length of the array should be equal to the number of DSPs on the card, and each entry should contain a character string containing the path and file name of an RBF-format FPGA configuration data file. If this DSP does not need to load an FPGA, or if the FPGA is not necessary for your application, set the value of the appropriate entry to NULL. Set the parameter itself to NULL if you do not need to load any FPGA.
`OUT int* card_num`	An integer value will be placed in this pointer. This value must be used for all subsequent calls that pertain to this card.
`OUT h2d_pci_device_t *** pciDevArr`	A pointer to a table of PCI stream device pointers, one per DSP. This pointer is filled by this function and should be given to any filter that needs to open PCI streams.
`IN MANGOX_ERR_CB * err_cbs`	Array of callback functions to be used in case of an error, one per DSP. See typedef MANGOX_ERR_CB (see page 9) for a description of this function. This function can be the same or different for each DSP. The function can be NULL if you do not need this functionality for a specific DSP. Or, if you do not need this functionality for any DSP, you may set this parameter as NULL. A NULL value will cause no callback to be called, in which case reported errors will be ignored.
`IN void ** err_args`	Array of void* values, one per DSP, that will be given as the first parameter (arg) to the callback function. Use NULL instead of an array pointer if you don't need this functionality; in this case the value of arg shall always be NULL.

Returns

Since this function performs many tasks, there are several possible error conditions:

- MANGOERROR_INVALID_CONFIGURATION - File not found (rbf or out); Make sure FPGA (.rbf) and executable (.out) files are in the locations given by the fpga_fnames and coff_fnames parameters.
- MANGOERROR_ERR_INVALID_HANDLE - card handle is problematic or card not supported.
- MANGOERROR_INSUFFICIENT_RESOURCES - can't allocate shared memory.
- MANGOERROR_RESOURCE_NOT_READY - error loading fpga.
- MANGOERROR_TIMEOUT - DSP did not respond to initial request within a reasonable time frame.
- MANGOERROR_FAILURE - other unexpected error.
- MANGOERROR_SUCCESS - Card opened successfully.

Description

This function receives a handle to a card (which had been initialized using MangoC64Boards_Open) and an array indicating which DSPs the library is allowed to use, as well as indicating what DSP executable (out) file to load on each. The function will load the DSPs and perform relevant initializations such as FPGA loading where applicable.

Remarks

After calling this function, the user application must not access the DSPs that are granted to the library in any way other than via MangoX calls. The DSPs marked as NULL are untouched, and may be freely used.

2.1.3 MangoX_Close Function

```
MANGOERROR_error_t MangoX_Close();
```

Summary

Closes access to the library and frees all associated resources.

File

MangoXExp.h (see page 10)

Returns

- MANGOERROR_FAILURE - Failure communicating with DSP or closing PCI streams.
- MANGOERROR_RESOURCE_NOT_READY - DSP reported failure when attempting to communicate.
- MANGOERROR_INSUFFICIENT_RESOURCES - Failure freeing shared memory.
- MANGOERROR_SUCCESS - Success freeing resources.

Remarks

- This function will call MangoX_CardDisable (see page 3) to disable all cards that haven't yet been disabled. All remarks from that function apply here as well.
- After calling this function no MangoX function may be called except MangoX_Open (see page 8).
- It is good practise to always call this function before exiting your application, even if aborting in case of an error. Depending on your OS, exiting the program without calling this function may cause an application crash.

2.1.4 MangoX_ConfigVideo Function

```
MANGOERROR_error_t MangoX_ConfigVideo(IN int card_num, IN int dsp, IN enConfigVideoCmds
cmd, IN enVideoStandard std, IN int chip_id);
```

Summary

Configure video chips.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
`IN int card_num`	Card number, given by MangoX_CardEnable (see page 3).
`IN int dsp`	Dsp number to use. This DSP must have been enabled in the coff_fnames parameter of MangoX_CardEnable (see page 3).
`IN enConfigVideoCmds cmd`	Indicate whether you wish to access a video-in or video-out chip.
`IN enVideoStandard std`	The video standard (PAL/NTSC) you wish to use with this chip.
`IN int chip_id`	The ID of the chip you wish to access, or -1 to access all chips of the requested type.

Returns

- MANGOERROR_ERR_INVALID_PARAMETER - attempted to use DSP not enabled in MangoX_CardEnable (see page 3).
- MANGOERROR_RESOURCE_NOT_READY - DSP failed to configure chip or chip is not accessible from this DSP.
- MANGOERROR_FAILURE - Other fatal error.
- MANGOERROR_SUCCESS - Chip configured successfully.

Description

This function initializes the video input/output chips.

Remarks

- This function is only needed on the Seagull PMC, Lark and Raven-C.
- See the MangoAV_HW User's Manual for details on chip IDs.

2.1.5 MangoX_DeleteGraph Function

```
MANGOERROR_error_t MangoX_DeleteGraph(IN int graph_num, IN int card_num, IN int dsp);
```

Summary

Closes a graph on a DSP.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
IN int graph_num	Graph number given by MangoX_SubmitGraph (see page 8).
IN int card_num	Card number given by MangoX_CardEnable (see page 3).
IN int dsp	DSP number on card.

Returns

- MANGOERROR_ERR_INVALID_PARAMETER - attempted to use DSP not enabled in MangoX_CardEnable (see page 3).
- MANGOERROR_FAILURE - Fatal error.
- MANGOERROR_RESOURCE_NOT_READY - DSP failed to delete the stream. Perhaps the graph number is incorrect or was not opened on this DSP?
- MANGOERROR_SUCCESS - stream successfully deleted.

Description

This call tells the DSP to closes a graph and free all associated resources, including all filters and tasks. The given graph number is invalidated by this function. Note that this call does not free the graph object on the host; the user must delete the graph object in the application after calling this function. (Deleting the graph object will delete all associated filters and resources on the host side.)

2.1.6 MangoX_DspCustomCommand Function

```
MANGOERROR_error_t MangoX_DspCustomCommand(IN int card_num, IN int dsp, IN unsigned int
cst_cmd, IN unsigned int cst_param0, IN unsigned int cst_param1, OUT int* ret_val);
```

Summary

Send user-definable command and parameter to DSP (regardless of whether there are any open graphs).

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
IN int card_num	Card number, given by MangoX_CardEnable (see page 3).

IN int dsp	Dsp number to use. This DSP must have been enabled in the coff_fnames parameter of MangoX_CardEnable (see page 3).
IN unsigned int cst_cmd	Custom command.
IN unsigned int cst_param0	First parameter for command.
IN unsigned int cst_param1	Second parameter for command.
OUT int* ret_val	Pointer for return value array; may be NULL if none is expected. (The size of this array must be NUM_CUSTOM_CMD_RETVALS.)

Returns

- MANGOERROR_ERR_INVALID_PARAMETER - attempted to use DSP not enabled in MangoX_CardEnable (see page 3).
- MANGOERROR_RESOURCE_NOT_READY - Command sent, but custom function responded with failure.
- MANGOERROR_FAILURE - Other fatal error.
- MANGOERROR_SUCCESS - Command sent and acknowledged.

Remarks

User must implement this command on the DSP side.

2.1.7 MangoX_GetVersion Function

```
MANGOERROR_error_t MangoX_GetVersion(OUT MANGOBIOS_version_t * version);
```

Summary

Returns lib version.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
OUT MANGOBIOS_version_t * version	Pointer for returned version number.

Returns

- MANGOERROR_ERR_INVALID_PARAMETER - if null pointer was received
- MANGOERROR_SUCCESS - successful.

Remarks

Version is returned as an unsigned int containing major and minor version numbers in this format: ((major << 16) | minor)

2.1.8 MangoX_GetVersionString Function

```
MANGOERROR_error_t MangoX_GetVersionString(IN int card_num, IN int dsp, OUT char**
ver_string);
```

Summary

Receive version information string from DSP.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
`IN int card_num`	Card number given by MangoX_CardEnable (see page 3).
`IN int dsp`	DSP number.
`OUT char** ver_string`	Empty pointer, will point to string after this function call.

Returns

- MANGOERROR_ERR_INVALID_PARAMETER - attempted to use DSP not enabled in MangoX_CardEnable (see page 3).
- MANGOERROR_FAILURE - Fatal error.
- MANGOERROR_RESOURCE_NOT_READY - DSP reported failure.
- MANGOERROR_SUCCESS - successful.

2.1.9 MangoX_Open Function

```
MANGOERROR_error_t MangoX_Open(IN MangoX_Attrs_T * attrs);
```

Summary

Opens access to the library.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
`IN MangoX_Attrs_T * attrs`	Reserved; use NULL.

Returns

MANGOERROR_SUCCESS - Library opened successfully.

Remarks

This function must be called before any other functions. (Undefined results may occur if this function if not called.)

2.1.10 MangoX_SubmitGraph Function

```
MANGOERROR_error_t MangoX_SubmitGraph(OUT int* graph_num, IN int card_num, IN int dsp, IN
CMangoXGraph* pGraph);
```

Summary

Opens a stream on a dsp.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
`OUT int* graph_num`	Pointer to graph number, which will be filled in by the library.
`IN int card_num`	Card number, given by MangoX_CardEnable (see page 3).

IN int dsp	Dsp number to use. This DSP must have been enabled in the coff_fnames parameter of MangoX_CardEnable (see page 3). Make sure that the DSP you are trying to use has the capability you are requesting; for example, on the Seagull PMC, only DSP 0 has the ability to output an image to a video monitor.
IN CMangoXGraph* pGraph	A fully configured and connected CMangoXGraph object.

Returns

- MANGOERROR_ERR_INVALID_PARAMETER - attempted to use DSP not enabled in MangoX_CardEnable (see page 3).
- MANGOERROR_RESOURCE_NOT_READY - DSP failed to create graph.
- MANGOERROR_FAILURE - Fatal error, or graph serialize buffer is too large.
- MANGOERROR_SUCCESS - Graph submitted successully.

Description

This function passes a new graph to the DSP.

2.2 Types

The following table lists types in this documentation.

Types

Type	Description
MANGOX_ERR_CB (see page 9)	This is a prototype for a user callback function that is called whenever a DSP reports an error.

2.2.1 MANGOX_ERR_CB Type

```
typedef void (* MANGOX_ERR_CB)(void * arg, int card_num, int dsp_num, Error_Severity_E
err_type, int err_num, char * err_string);
```

Summary

This is a prototype for a user callback function that is called whenever a DSP reports an error.

File

MangoXExp.h (see page 10)

Parameters

Parameters	Description
%PAR0%	This is the value given by the user in MangoX_CardEnable (see page 3), which you may use for any purpose.
%PAR1%	The card number (same as returned by MangoX_CardEnable (see page 3)).
%PAR2%	The DSP number that reported the error.
%PAR3%	Severity of the error, as reported by the DSP. Numerical error code as reported by the DSP.
%PAR4%	Text string containing human-readable text indicating the name of the class (filter) that reported the error and a description of the problem.

Description

A pointer to this function should be given in the call to MangoX_CardEnable (see page 3). The function will receive information regarding the source of the error, its severity and a human-readable string indicating the nature of the problem. Note that it is very likely that the DSP will not be able to gracefully recover from an error reported here, so it is best to find

the cause of the problem and prevent it from ever happening in your production system.

Remarks

- The context from which this function is called is a separate thread, created specifically for this purpose. There is one such thread for each enabled DSP.
- The DSP will report errors of three possible types:

1. Warning - minor problems
2. Error - The DSP could not complete a task and has probably not recovered gracefully, but other tasks can probably continue to operate. This is still considered a problem.
3. Fatal - A problem has occurred that could only happen in case of a severe problem, such as corruption of a stack or heap. It is recommended to halt the program immediately in this case.

2.3 Files

The following table lists files in this documentation.

Files

File	Description
MangoXExp.h (see page 10)	Copyright 2005 Mango DSP, Inc. All rights reserved. Software and information contained herein is confidential and proprietary to Mango DSP, and any unauthorized copying, dissemination or use is strictly prohibited. Filename: MangoXExp.h General Description: Exported header for MangoX library Created: 21/2/2005 Author: Itay Chamiel

2.3.1 MangoXExp.h

Copyright 2005 Mango DSP, Inc. All rights reserved. Software and information contained herein is confidential and proprietary to Mango DSP, and any unauthorized copying, dissemination or use is strictly prohibited.

Filename: MangoXExp.h

General Description: Exported header for MangoX library

Created: 21/2/2005

Author: Itay Chamiel

Functions

Function	Description
MangoX_CardDisable (see page 3)	Frees all resources associated with a previous successful call to MangoX_CardEnable (see page 3)() and disables all enabled DSPs.
MangoX_CardEnable (see page 3)	Grant library permission to use a card and some (or all) of its DSPs; initializes DSPs.
MangoX_Close (see page 4)	Closes access to the library and frees all associated resources.
MangoX_ConfigVideo (see page 5)	Configure video chips.
MangoX_DeleteGraph (see page 6)	Closes a graph on a DSP.
MangoX_DspCustomCommand (see page 6)	Send user-definable command and parameter to DSP (regardless of whether there are any open graphs).
MangoX_GetVersion (see page 7)	Returns lib version.

MangoX_GetVersionString (see page 7)	Receive version information string from DSP.
MangoX_Open (see page 8)	Opens access to the library.
MangoX_SubmitGraph (see page 8)	Opens a stream on a dsp.

Types

Type	Description
MANGOX_ERR_CB (see page 9)	This is a prototype for a user callback function that is called whenever a DSP reports an error.

Legend

	Method

3 Macros

The following table lists %CATEGORYL% in this documentation.

Index