MangoC64Boards

Host API Guide

Table of Contents

MangoC64Boards

1 Introduction

MangoC64Boards is designed to replace the board-specific libraries for PCI cards. It is capable of detecting and opening any combination of supported PCI boards and to access their DSPs.

It is currently compatible with the following cards: Seagull PMC; Seagull PCI; Phoenix (PC-104).

Upon a request to open access to a particular card, the library will scan the available devices and determine whether the requested board type is available and then open it.

The library receives a list of devices created by the MangoBIOS function MANGOBIOS_getDeviceHandles. It recognizes card types by the PCI bus structure of the DSP devices. See MangoC64Boards_Open (see page 7) for more information.

Note that some card combinations can confuse the library. Known examples are:

- Two Seagull PMCs, mounted without mezzanine (video) cards, can appear as one Seagull PCI board.

In the supplied program examples, the application attempts to open all supported card types until successful. This is because the examples are generic and are intended for many board types. On the other hand, user applications are normally intended for one board type. Therefore, in your application you should only attempt to open the card type corresponding to the boards installed in your system. This will eliminate any possible confusion arising from the examples above.

2 Symbol Reference

Files

File	Description
MangoC64BoardsExp.h (see page 21)	MangoC64Boards library exported header file

Functions

Function	Description
MangoC64Boards_Get_Version (see page 6)	Gets MANGOBIOS_version_t
MangoC64Boards_Open (see page 7)	Opens access to a single PCI board of specified type.

Types

Type	Description
MangoC64Boards_attrs_t (see page 9)	typedef of struct MANGOBIOS_dummy_t
MangoC64Boards_FXN_close (see page 10)	Close MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_h2d_interrupt (see page 11)	Sends a PCI interrupt to MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_init_emif (see page 12)	Initializes an emif of one dsp of a MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_load_from_file (see page 13)	Loads one dsp of a MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_read_config (see page 14)	Reads a PCI configuration word from a dsp
MangoC64Boards_FXN_read_memory (see page 15)	Reads from memory on a dsp
MangoC64Boards_FXN_reset (see page 16)	Places a dsp on MangoC64Boards_handle_t (see page 19) in warm reset
MangoC64Boards_FXN_write_config (see page 17)	Writes a PCI configuration word to a dsp
MangoC64Boards_FXN_write_memory (see page 18)	Writes to memory on a dsp
MangoC64Boards_handle_t (see page 19)	typedef of struct MangoC64Boards_handle_s (see page 4)

Structs, Records, Enums

Struct, Record, Enum	Description
MangoC64Boards_handle_s (see page 4)	structure for using the MangoC64Boards

2.1 Structs, Records, Enums

Structs

Struct	Description
MangoC64Boards_handle_s (see page 4)	structure for using the MangoC64Boards

2.1.1 MangoC64Boards_handle_s

```
struct MangoC64Boards_handle_s {
};
```

File

MangoC64BoardsExp.h (see page 21)

Description

structure for using the MangoC64Boards

2.2 Functions

Functions

Function	Description
MangoC64Boards_Get_Version (see page 6)	Gets MANGOBIOS_version_t
MangoC64Boards_Open (see page 7)	Opens access to a single PCI board of specified type.

2.2.1 MangoC64Boards_Get_Version

```
MANGOERROR_error_t MangoC64Boards_Get_Version(MANGOBIOS_version_t * version);
```

Summary

Gets MANGOBIOS_version_t

File

MangoC64BoardsExp.h (see page 21)

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success

Description

Gets version information for MangoC64Boards Library

Remarks

None

Example

```
int errorCode;
MANGOBIOS_version_t version;

errorCode = MangoC64Boards_Get_Version(
 &version
 );
```

2.2.2 MangoC64Boards_Open

```
MANGOERROR_error_t MangoC64Boards_Open(MangoC64Boards_handle_t * handle,
MANGOBIOS_deviceHandle_t * devices, int num_devices, const board_footprint_t * footprint,
const MangoC64Boards_attrs_t * attrs);
```

Summary

Opens access to a single PCI board of specified type.

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
`MangoC64Boards_handle_t * handle`	Pointer for handle
`MANGOBIOS_deviceHandle_t * devices`	Array of MANGOBIOS_deviceHandle_t devices as generated by MANGOBIOS_getDeviceHandles.
`int num_devices`	Length of device array, as given by MANGOBIOS_getNumDevices.
`const board_footprint_t * footprint`	A "footprint" corresponding to the board type to open. You should use the footprints supplied in the MangoBoards.c file.
`const MangoC64Boards_attrs_t * attrs`	Normally NULL, but can control bus_scan_start. Normally the first board that will be opened is the one containing the lowest bus number, but you can force a different order by setting bus_scan_start to the minimal bus number that will be scanned (any board containing a lower bus number will be ignored).

Returns

- MANGOERROR_SUCCESS - Success
- MANGOERROR_INVALID_CONFIGURATION - Could not find a device arrangement corresponding to the requested board type.
- Other value - Error from MANGOBIOS_deviceOpen or MANGOBIOS_deviceGetProperty.

Description

This function receives a device list as created by MANGOBIOS_getDeviceHandles and the type of card requested. It will then scan the list and, if it finds a group of devices matching this board type's "footprint" it will mark those devices as open and initialize a handle for the application to use for accessing the board.

Remarks

- The 'devices' array given should always be the start of the device array. There is no need to increment this pointer as opened devices are internally marked.

Example

```
MangoC64Boards_handle_t card;
int num_dev;
MANGOBIOS_deviceHandle_t * devices;
if(MANGOBIOS_getNumDevices(NULL, &num_dev) != MANGOERROR_SUCCESS)
    return -1;
devices = (MANGOBIOS_deviceHandle_t *)malloc(sizeof(MANGOBIOS_deviceHandle_t) * num_dev);
if (MANGOBIOS_getDeviceHandles(NULL, devices) != MANGOERROR_SUCCESS)
    return -1;
if (MangoC64Boards_Open(&card, devices, num_dev, &SEAGULL_PMC_BOARD, NULL) !=
MANGOERROR_SUCCESS)
    return -1;
```

2.3 Types

Types

Type	Description
MangoC64Boards_attrs_t (see page 9)	typedef of struct MANGOBIOS_dummy_t
MangoC64Boards_FXN_close (see page 10)	Close MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_h2d_interrupt (see page 11)	Sends a PCI interrupt to MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_init_emif (see page 12)	Initializes an emif of one dsp of a MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_load_from_file (see page 13)	Loads one dsp of a MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_read_config (see page 14)	Reads a PCI configuration word from a dsp
MangoC64Boards_FXN_read_memory (see page 15)	Reads from memory on a dsp
MangoC64Boards_FXN_reset (see page 16)	Places a dsp on MangoC64Boards_handle_t (see page 19) in warm reset
MangoC64Boards_FXN_write_config (see page 17)	Writes a PCI configuration word to a dsp
MangoC64Boards_FXN_write_memory (see page 18)	Writes to memory on a dsp
MangoC64Boards_handle_t (see page 19)	typedef of struct MangoC64Boards_handle_s (see page 4)

2.3.1 MangoC64Boards_attrs_t

```
typedef struct {
} MangoC64Boards_attrs_t;
```

File

MangoC64BoardsExp.h (see page 21)

Description

typedef of struct MANGOBIOS_dummy_t

2.3.2 MangoC64Boards_FXN_close

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_close)(MangoC64Boards_handle_t * handle);
```

Summary

Close MangoC64Boards_handle_t (see page 19)

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success

Description

Closes 'handle.'

Remarks

None

Example

```
int errorCode;
errorCode = sgl_pmc.close(
 &sgl_pmc
 );
```

2.3.3 MangoC64Boards_FXN_h2d_interrupt

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_h2d_interrupt)(MangoC64Boards_handle_t *
handle, int dsp);
```

Summary

Sends a PCI interrupt to MangoC64Boards_handle_t (see page 19)

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board
dsp	Dsp number

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success
Other value	Error from MANGOBIOS_deviceWrite

Description

Sends 'dsp' on 'handle' a PCI interrupt

Remarks

None

Example

```
int errorCode;
errorCode = sgl_pmc.h2d_interrupt(
 &sgl_pmc,
 0
 );
```

2.3.4 MangoC64Boards_FXN_init_emif

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_init_emif)(MangoC64Boards_handle_t *
handle, emif_init_t * emif, int dsp);
```

Summary

Initializes an emif of one dsp of a MangoC64Boards_handle_t (see page 19)

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board
dsp	Dsp number

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success
Other value	Error from MANGOBIOS_deviceWrite

Description

Initializes 'dsp' on 'handle' 's EMIFs with a valid EMIF configuration for SDRAM access

Remarks

None

Example

```
int errorCode;
errorCode = sgl_pmc.init_emif(
 &sgl_pmc,
 0
 );
```

2.3.5 MangoC64Boards_FXN_load_from_file

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_load_from_file)(MangoC64Boards_handle_t *
handle, int dsp, const char * file);
```

Summary

Loads one dsp of a MangoC64Boards_handle_t (see page 19)

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board
dsp	Dsp number
file	Path to a (COFF formatted) DSP .out file

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success
MANGOERROR_ERR_INVALID_PARAMETER	Failed fopen on 'file' Coff_file2writes parses it as having no write sections
MANGOERROR_INSUFFICIENT_RESOURCES	Failed malloc
MANGOERROR_FAILURE	Failed fseek or fread on 'file'
Other value	Error from Coff_file2writes Error from *MangoC64Boards_FXN_write_memory (see page 18) Error from MANGOBIOS_deviceWrite

Description

Loads 'dsp' on 'handle' with the COFF formatted 'file.'

Remarks

None

Example

```
int errorCode;
errorCode = sgl_pmc.load_from_file(
 &sgl_pmc,
 0,
 "fpga_load.out"
 );
```

2.3.6 MangoC64Boards_FXN_read_config

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_read_config)(MangoC64Boards_handle_t *
handle, int dsp, int offset, void * data, int size);
```

Summary

Reads a PCI configuration word from a dsp

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board
dsp	Dsp number
offset	Offset in bytes from start of configuration space
data	Pointer for data
size	Length in bytes of the read (1,2,4)

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success
MANGOERROR_ERR_INVALID_PARAMETER	'size' is not equal to 1, 2, or 4
Other value	Error from MANGOBIOS_devicePciRegRead

Description

Reads a configuration word from 'dsp' on 'handle'

Remarks

None

Example

```
int errorCode;
int dev_ven_id;
errorCode = sgl_pmc.read_config(
 &sgl_pmc,
 0
 0x0, (offset for Device/Vendor ID)
 &dev_ven_id,
 0x4
 );
```

2.3.7 MangoC64Boards_FXN_read_memory

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_read_memory)(MangoC64Boards_handle_t *
handle, int dsp, void * hst_adr, unsigned int dsp_adr, unsigned int bytes);
```

Summary

Reads from memory on a dsp

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board
dsp	Dsp number
hst_adr	Pointer to memory on host
dsp_adr	Location of memory on dsp
bytes	Length in bytes of the write

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success
Other value	Error from MangoC64Boards_FXN_read_memory Error from MANGOBIOS_deviceRead

Description

Reads from memory on 'dsp' on 'handle'

Remarks

None

Example

```
int errorCode;
int dev_ven_id;
int buffer[0x100];

errorCode = sgl_pmc.read_memory(
 &sgl_pmc,
 0
 &buffer,
 0x80000000, (beginning of SDRAM)
 0x100
 );
```

2.3.8 MangoC64Boards_FXN_reset

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_reset)(MangoC64Boards_handle_t * handle,
int dsp);
```

Summary

Places a dsp on MangoC64Boards_handle_t (see page 19) in warm reset

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board
dsp	Dsp number

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success
Other value	Error from MANGOBIOS_deviceWrite

Description

Places 'dsp' on 'handle' in warm reset

Remarks

A PCI interrupt from the host will take a dsp out of warm reset

Example

```
int errorCode;
errorCode = sgl_pmc.reset(
 &sgl_pmc,
 0
 );
```

2.3.9 MangoC64Boards_FXN_write_config

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_write_config)(MangoC64Boards_handle_t *
handle, int dsp, int offset, const void * data, int size);
```

Summary

Writes a PCI configuration word to a dsp

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board
dsp	Dsp number
offset	Offset in bytes from start of configuration space
data	Pointer to data
size	Length in bytes of the write (1,2,4)

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success
MANGOERROR_ERR_INVALID_PARAMETER	'size' is not equal to 1, 2, or 4
Other value	Error from MANGOBIOS_devicePciRegRead

Description

Writes a configuration word to 'dsp' on 'handle'

Remarks

None

Example

```
int errorCode;
int dev_ven_id;
int bar0 = 0xffa00000;
errorCode = sgl_pmc.write_config(
 &sgl_pmc,
 0
 0x10, (offset for base address register 0)
 &bar0,
 0x4
 );
```

2.3.10 MangoC64Boards_FXN_write_memory

```
typedef MANGOERROR_error_t (* MangoC64Boards_FXN_write_memory)(MangoC64Boards_handle_t *
handle, int dsp, const void * hst_adr, unsigned int dsp_adr, unsigned int bytes);
```

Summary

Writes to memory on a dsp

File

MangoC64BoardsExp.h (see page 21)

Parameters

Parameters	Description
handle	Handle to MangoC64Boards board
dsp	Dsp number
hst_adr	Pointer to memory on host
dsp_adr	Location of memory on dsp
bytes	Length in bytes of the write

Returns

Status

Return Values

Return Values	Description
MANGOERROR_SUCCESS	Success
Other value	Error from MangoC64Boards_FXN_write_memory Error from MANGOBIOS_deviceRead

Description

Writes to memory on 'dsp' on 'handle'

Remarks

None

Example

```
int errorCode;
int dev_ven_id;
int buffer[0x100];

memset(buffer, 0, sizeof(int) * 0x100);
errorCode = sgl_pmc.write_memory(
 &sgl_pmc,
 0
 &buffer,
 0x80000000, (beginning of SDRAM)
 0x100
 );
```

2.3.11 MangoC64Boards_handle_t

```
typedef struct MangoC64Boards_handle_s MangoC64Boards_handle_t;
```

File

MangoC64BoardsExp.h (see page 21)

Description

typedef of struct MangoC64Boards_handle_s (see page 4)

2.4 Files

Files

File	Description
MangoC64BoardsExp.h (see page 21)	MangoC64Boards library exported header file

2.4.1 MangoC64BoardsExp.h

MangoC64Boards library exported header file

Description

MangoC64Boards library exported api declarations

History

Author	Change Description
Nachum Kanovsky	Created

Functions

Function	Description
MangoC64Boards_Get_Version (see page 6)	Gets MANGOBIOS_version_t
MangoC64Boards_Open (see page 7)	Opens access to a single PCI board of specified type.

Structs

Struct	Description
MangoC64Boards_handle_s (see page 4)	structure for using the MangoC64Boards

Types

Type	Description
MangoC64Boards_attrs_t (see page 9)	typedef of struct MANGOBIOS_dummy_t
MangoC64Boards_FXN_close (see page 10)	Close MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_h2d_interrupt (see page 11)	Sends a PCI interrupt to MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_init_emif (see page 12)	Initializes an emif of one dsp of a MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_load_from_file (see page 13)	Loads one dsp of a MangoC64Boards_handle_t (see page 19)
MangoC64Boards_FXN_read_config (see page 14)	Reads a PCI configuration word from a dsp
MangoC64Boards_FXN_read_memory (see page 15)	Reads from memory on a dsp
MangoC64Boards_FXN_reset (see page 16)	Places a dsp on MangoC64Boards_handle_t (see page 19) in warm reset
MangoC64Boards_FXN_write_config (see page 17)	Writes a PCI configuration word to a dsp
MangoC64Boards_FXN_write_memory (see page 18)	Writes to memory on a dsp
MangoC64Boards_handle_t (see page 19)	typedef of struct MangoC64Boards_handle_s (see page 4)

Index