USB Mass Storage Firmware

User Guide

Section 1

Introduction 1-1

1. References 1-1
2. Abbreviations 1-1
3. Supported Controllers 1-1
4. Introduction 1-1

Section 2

Memory General Management 2-3

5. Example: add the DataFlash memory support 2-4

Section 3

Configure the Nand Flash driver 3-5

Section 4

Configure the Data Flash driver 4-7

Section 1

Introduction

1. References

- Universal Serial Bus Specification, revision 2.0
- Universal Serial Bus Class Definition for Communication Devices, version 1.1
- USB Mass Storage Overview, revision 1.2
- USB Mass Storage Bulk Only, revision 1.0

2. Abbreviations

- USB: Universal Serial Bus
- VID: Vendor Identifier
- PID: Product Identifier
- LUN: Logical Unit Number

3. Supported Controllers

- AT89C5130/31A & AT8xC5122D

4. Introduction

The aim of this document is to describe the way to use different memories with the USB Mass Storage firmware.

Refer to the USB Mass Storage device application note for more information.

Section 2

Memory General Management

By default, the memory supported is the Nand Flash memory (2K). You can use either the DataFlash memory or the virtual memory (internal memory) instead. Each memory has its own LUN (Logic Unit Number), so to select the memory that you want to use you have to enable the related LUN and disable the others.

You will find the LUN definition In the **conf_access.h** file:

- enable or disable the memory:

```
// Active the Logical Unit
#define  LUN_0                DISABLE // virtual  memory
#define  LUN_1                ENABLE          // NF (2k)  memory
#define  LUN_2                DISABLE         // NF (512) memory
#define  LUN_3                DISABLE         // DF memory
#define  LUN_4                DISABLE
#define  LUN_5                DISABLE
#define  LUN_6                DISABLE
#define  LUN_7                DISABLE
```

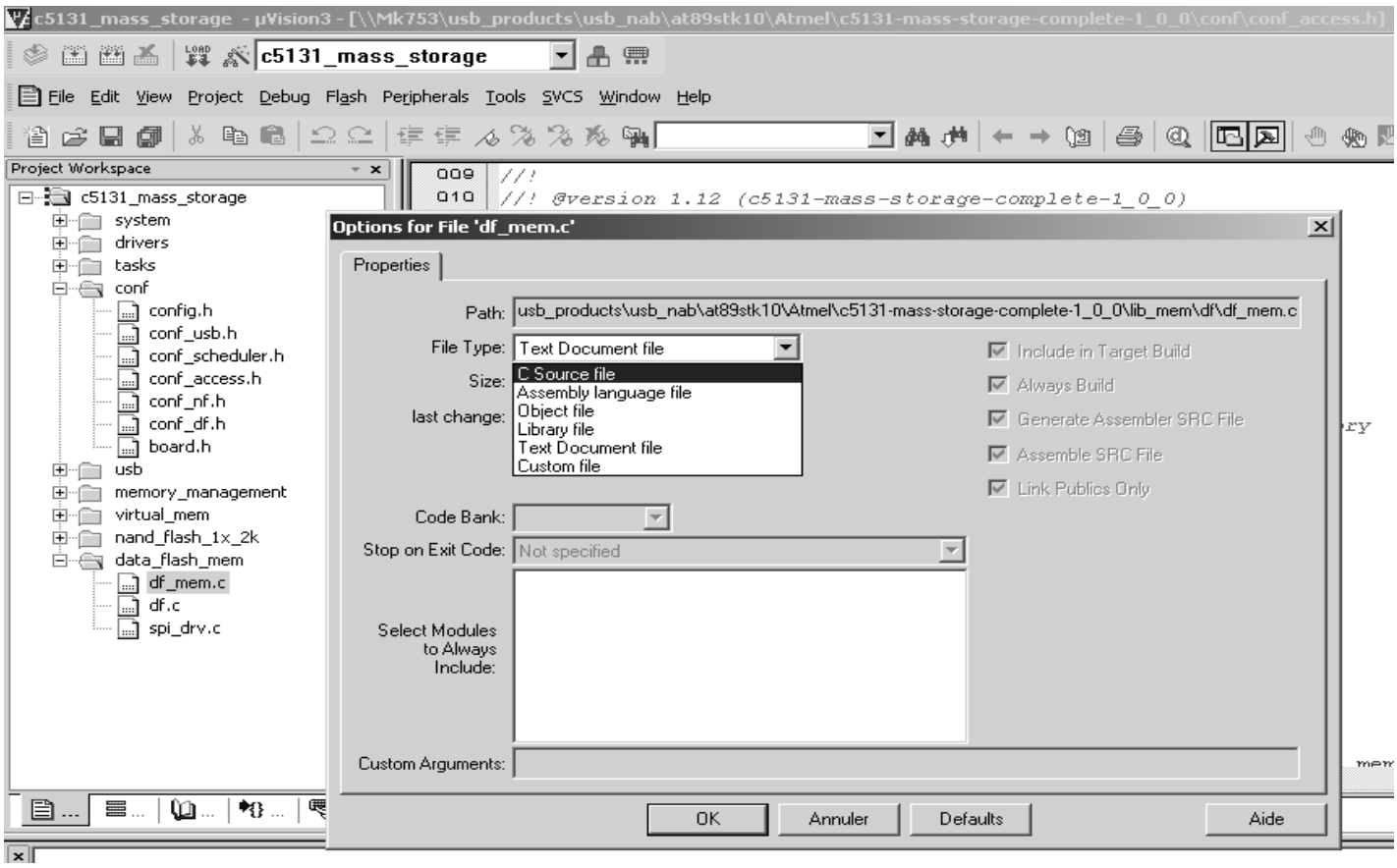
- define the memory interface:

```
// LUN 0 DEFINE
#if (LUN_0 == ENABLE)
#define  VIRTUAL_MEM                              ENABLE
#else
#define VIRTUAL_MEM                              DISABLE
#endif
#define  LUN_0_INCLUDE "lib_mem\virtual_mem\virtual_mem.h"
#define  Lun_0_test_unit_ready()                  virtual_test_unit_ready()
#define  Lun_0_read_capacity(nb_sect) virtual_read_capacity(nb_sect)
#define  Lun_0_wr_protect()                       virtual_wr_protect()
#define  Lun_0_removal()                          virtual_removal()
#define  Lun_0_read_10(ad, sec)                   virtual_read_10(ad, sec)
#define  Lun_0_usb_read()                         virtual_usb_read()
#define  Lun_0_write_10(ad, sec)                  virtual_write_10(ad, sec)
#define  Lun_0_usb_write()                        virtual_usb_write()
```

5. Example: add the DataFlash memory support

In this example, we add the DataFlash memory.

1. Include the drivers in the project.
   Check if the DataFlash drivers (df_mem.c, df.c, spi_drv.c) are included in the project. Then, check if they are **declared as C files** (or A51) in order for them to be compiled (file options > file type > C source file):



2. Put the Nand Flash files (nf_mem.c, nf.c, nf_drv.c, nf_drv_load.a51) in “text document file” state in order to avoid it to be compiled(file options > file type > text document file).
3. In the conf_access.h file, enable the LUN 3 and disable the others:

```
// Active the Logical Unit
#define  LUN_0                  DISABLE // virtual  memory
#define  LUN_1                  DISABLE          // NF (2k)  memory
#define  LUN_2                  DISABLE         // NF (512) memory
#define  LUN_3                  ENABLE         // DF memory
#define  LUN_4                  DISABLE
#define  LUN_5                  DISABLE
#define  LUN_6                  DISABLE
#define  LUN_7                  DISABLE
```

Section 3

Configure the Nand Flash driver

The following Nand Flash files have to be included into the project :

- nf_mem.c
- nf.c
- nf_drv.c
- nf_drv_load.a51

There is no configuration for the Nand Flash driver. Just select the correct one corresponding to the page size of the Nand Flash: 512 Bytes or 2 kBytes.

Nand Flash Size Auto-detection

All drivers support nand flash size autodetection. This feature can be activated by defining NF_CAPACITY_AUTO_DETECT to TRUE, in board.h. If you want to optimize code generation, define NF_CAPACITY_AUTO_DETECT to FALSE. In this case, total memory size must be defined in board.h from NF_16 (16Mbytes) to NF_512 (512Mbytes)

Reserved Zone

This one permits to divided the NF memory in two memory space :

- normal zone

- reserved zone

It's possible to define a reserved zone for application inside nand-flash memory. This reserved zone can not be modified by an operating system when application is in mass storage mode, except by set application in update mode.

To set application in update mode, set flag reserved_disk_space (see nf_mem.c for definition) to TRUE.

The "MEM_RESERVED_SIZE" value define the size of reserved zone (unit sector = unit 512B). This one must be modulo a cluster (max cluster size = 32*64).

Size of reserved space is defined in config.h file. For example following line

#define MEM_RESERVED_SIZE1024

will keep a reserved space on nand flash with a size of 1024 sectors (512bytes for all drivers).

Logical starting address of this reserved space is stored in the variable nf_reserved_space_start.

Reserved zone can be read/write with all low level functions (nf_read_open, nf_read_write, ...).

normal zone size = total memory space - reserved zone size.

To access at these zones, you must change the value "nf_reserved_zone_selected" and reinit the memory.

Sample in tool_task.c :

```
case TOOL_SELECT_RSVD:
{
  nf_reserved_zone_selected = TRUE;
  mem_chip_select();
  break;
}

case TOOL_SELECT_NORMAL:
{
  nf_reserved_zone_selected = FALSE;
  mem_chip_select();
  break;
}
```

Full Chip Erase

Define NF_FULL_CHIP_ERASE determines how the function nf_erase_all_block will erase block during a format.

If NF_FULL_CHIP_ERASE is set to TRUE, then all physicals blocks, except bad block, specific block and reserved block will be erased.

If NF_FULL_CHIP_ERASE is set to FALSE, then only blocks that contains LUT will be erased.

Section 4

Configure the Data Flash driver

The following Data Flash files have to be included into the project :

- df_mem.c
- df.c
- spi_drv.c

In the **conf_df.h** file, select the memory type to support:

```
#define DF_TYPE DF_X_MB_TYPE // DF_4_MB_TYPE , DF_8_MB_TYPE , DF_X_MB_TYPE
```

The memory type can be:

- DF_4_MB for AT45DB321 memories,
- DF_8_MB for AT45DB642 memories,
- DF_X_MB for AT45DB002, AT45DB004 and AT45DB008 removable memories

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

7537A–USB–07/05 /xM