

**GigaDevice Semiconductor Inc.**

**Arm<sup>®</sup> Cortex<sup>®</sup>- M3/M4/M23/M33 32-bit MCU**

**应用笔记**

**AN022**

## 目录

目录.....	2
图索引.....	3
表索引.....	4
1. 简介.....	5
2. FLM 文件.....	6
2.1. 自定义 SDRAM_256Mb.FLM 文件.....	6
2.2. 将 FLM 文件拷贝到 keil 5 安装路径.....	6
3. APP_GPIO_Running_LED 工程.....	7
3.1. 修改工程分散加载文件.....	7
3.2. 修改中断向量表入口地址.....	7
3.3. 在 Keil 工程中添加 SDRAM_256Mb.FLM 文件.....	8
4. BOOT_EXMC_SDRAM 工程.....	9
5. 版本历史.....	11

## 图索引

图 2-1. FLM 文件地址范围 .....	6
图 3-1. Keil 工程 Flash Algorithm 配置 .....	8
图 4-1. 调试模式下 0xC0000000 地址的值 .....	9
图 4-2. APP_GPIO_Running_LED 工程编译出的 bin 文件.....	10

## 表索引

表 3-1. APP_GPIO_Running_LED 分散加载文件 .....	7
表 3-2. 设置中断向量表入口地址 .....	8
表 4-1. 设置 boot mode 为 EXMC SDRAM .....	9
表 4-2. 跳转到 APP_GPIO_Running_LED 工程所在的地址运行代码 .....	10
表 5-1. 版本历史 .....	11

## 1. 简介

SDRAM (Synchronous Dynamic Random Access Memory), 同步动态随机存储器。同步是指 Memory 工作需要同步时钟, 内部的命令的发送与数据的传输都以它为基准; 动态是指存储阵列需要不断的刷新来保证存储的数据不丢失, 因为 SDRAM 中存储数据是通过电容来工作的, 由于电容在自然放置状态是会有放电的, 如果电放完了, 也就意味着 SDRAM 中的数据丢失了, 所以 SDRAM 需要在电容的电量放完之前进行刷新; 随机是指数据不是线性依次存储, 而是自由指定地址进行数据的读写。

由于 SDRAM 具有某些和 SRAM 一样的特性, 如果需要在 SDRAM 上跑代码的话, 除了使用 MPU 以外, 还可以将 SDRAM Device0 地址(0xC0000000)映射到 0x00000000 来运行存储在 SDRAM 的代码。本文介绍了如何在 GD32F450 上将 SDRAM Device0 地址(0xC0000000)映射到 0x00000000 运行其中的代码的方法, 其中, APP\_GPIO\_Running\_LED 是烧录到 SDRAM Device0 地址(0xC0000000)的 APP 程序。BOOT\_EXMC\_SDRAM 是烧录到 0x08000000 地址的 BOOT 程序。

## 2. FLM 文件

### 2.1. 自定义 SDRAM\_256Mb.FLM 文件

本 AN 基于 GD32F450Z-EVAL 开发板，搭载一颗 MT48LZ16M16 A2P-6AIT SDRAM，容量为 256Mb。

在制作 FLM 文件的时候，起始地址为 0x00000000，具体如[图 2-1. FLM 文件地址范围](#)所示。

图 2-1. FLM 文件地址范围

Description	Device Size	Device Type	Address Range
SDRAM_256Mb	256M	Unknown Device	00000000H - 0FFFFFFFH

### 2.2. 将 FLM 文件拷贝到 keil 5 安装路径

将 SDRAM\_256Mb.FLM 文件拷贝到 Keil 5 安装路径，一般为“C:\Keil\_v5\ARM\Flash”。

### 3. APP\_GPIO\_Running\_LED 工程

#### 3.1. 修改工程分散加载文件

由于 APP\_GPIO\_Running\_LED 代码是要运行在 0x00000000 地址上的，所以需要修改工程的分散加载文件，将其加载到 0x00000000 地址上，分散加载文件如 [表 3-1. APP\\_GPIO\\_Running\\_LED 分散加载文件](#) 所示。

表 3-1. APP\_GPIO\_Running\_LED 分散加载文件

```
LR_IROM1 0x00000000 0x00100000 { ; load region size_region
  ER_IROM1 0x00000000 0x00100000 { ; load address = execution address
    *.o (RESET, +First)
    *(InRoot$$Sections)
    .ANY (+RO)
    .ANY (+XO)
  }
  RW_IRAM1 0x20000000 0x00030000 { ; RW data
    .ANY (+RW +ZI)
  }
}
```

#### 3.2. 修改中断向量表入口地址

由于代码本身运行 0x00000000 地址，所以需要修改 APP 工程的中断向量表入口地址，需要在合适的位置调用 `nvic_vector_table_set`，设置中断向量表入口地址为 0x00000000，偏移为 0，具体如 [表 3-2. 设置中断向量表入口地址](#) 所示。

表 3-2. 设置中断向量表入口地址

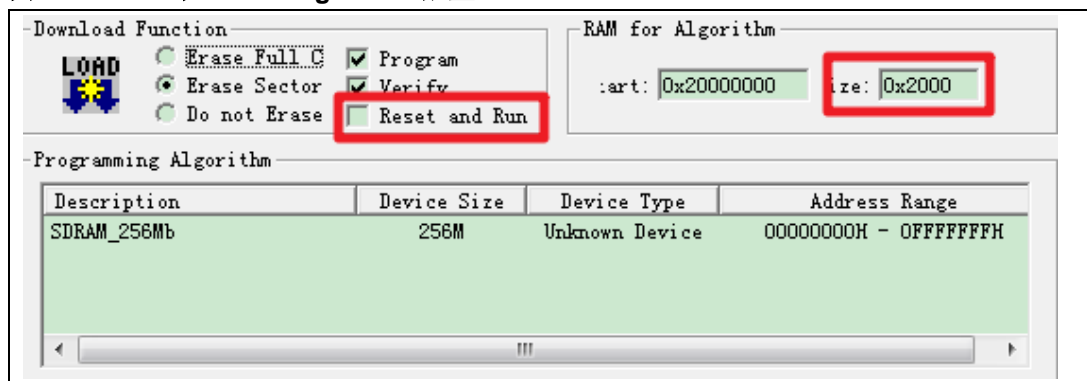
```
/*!
 *brief      main function
 *param[in]  none
 *param[out] none
 *retval     none
 */
int main(void)
{
    nvic_vector_table_set(0, 0);
    /* configure systick */
    systick_config();

    .....
}
```

### 3.3. 在 Keil 工程中添加 SDRAM\_256Mb.FLM 文件

打开 Keil 工程，在 Keil Flash download 界面，添加 SDRAM\_256Mb.FLM 文件，并将 RAM for Algorithm 的 Size 改为 0x2000，具体 [图 3-1. Keil 工程 Flash Algorithm 配置](#) 如所示。然后将程序烧录到 SDRAM 上。

图 3-1. Keil 工程 Flash Algorithm 配置





## 4. BOOT\_EXMC\_SDRAM 工程

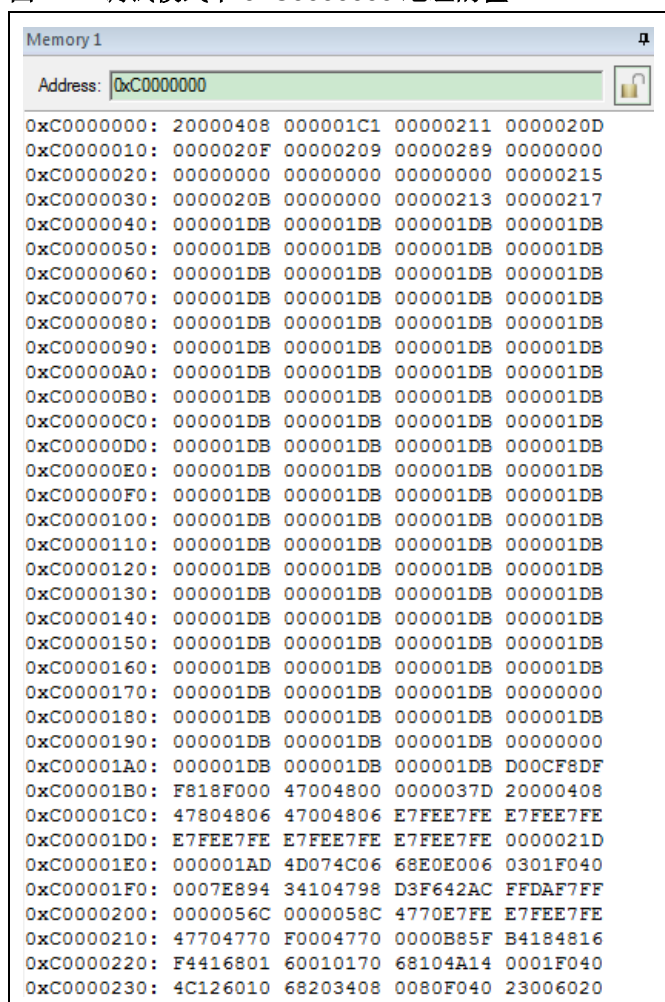
在 BOOT\_EXMC\_SDRAM 工程里,需要将 SYSCFG\_CFG0 寄存器 BOOT\_MODE 改为 0b`100。具体实现如[表 4-1. 设置 boot mode 为 EXMC SDRAM](#)所示。

**表 4-1. 设置 boot mode 为 EXMC SDRAM**

```
.....
rcu_periph_clock_enable(RCU_SYSCFG);
syscfg_bootmode_config(SYSCFG_BOOTMODE_EXMC_SDRAM);
.....
```

在 BOOT\_EXMC\_SDRAM 中使能了 EXMC SDRAM, 可以通过调试观察 SDRAM 的值, 是否为 APP\_GPIO\_Running\_LED 代码。具体如[图 4-1. 调试模式下 0xC0000000 地址的值](#)和[图 4-2. APP GPIO Running LED 工程编译出的 bin 文件](#)所示。

**图 4-1. 调试模式下 0xC0000000 地址的值**



Address	Value
0xC0000000	20000408 000001C1 00000211 0000020D
0xC0000010	0000020F 00000209 00000289 00000000
0xC0000020	00000000 00000000 00000000 00000215
0xC0000030	0000020B 00000000 00000213 00000217
0xC0000040	000001DB 000001DB 000001DB 000001DB
0xC0000050	000001DB 000001DB 000001DB 000001DB
0xC0000060	000001DB 000001DB 000001DB 000001DB
0xC0000070	000001DB 000001DB 000001DB 000001DB
0xC0000080	000001DB 000001DB 000001DB 000001DB
0xC0000090	000001DB 000001DB 000001DB 000001DB
0xC00000A0	000001DB 000001DB 000001DB 000001DB
0xC00000B0	000001DB 000001DB 000001DB 000001DB
0xC00000C0	000001DB 000001DB 000001DB 000001DB
0xC00000D0	000001DB 000001DB 000001DB 000001DB
0xC00000E0	000001DB 000001DB 000001DB 000001DB
0xC00000F0	000001DB 000001DB 000001DB 000001DB
0xC0000100	000001DB 000001DB 000001DB 000001DB
0xC0000110	000001DB 000001DB 000001DB 000001DB
0xC0000120	000001DB 000001DB 000001DB 000001DB
0xC0000130	000001DB 000001DB 000001DB 000001DB
0xC0000140	000001DB 000001DB 000001DB 000001DB
0xC0000150	000001DB 000001DB 000001DB 000001DB
0xC0000160	000001DB 000001DB 000001DB 000001DB
0xC0000170	000001DB 000001DB 000001DB 00000000
0xC0000180	000001DB 000001DB 000001DB 000001DB
0xC0000190	000001DB 000001DB 000001DB 00000000
0xC00001A0	000001DB 000001DB 000001DB D00CF8DF
0xC00001B0	F818F000 47004800 0000037D 20000408
0xC00001C0	47804806 47004806 E7FEE7FE E7FEE7FE
0xC00001D0	E7FEE7FE E7FEE7FE E7FEE7FE 0000021D
0xC00001E0	000001AD 4D074C06 68E0E006 0301F040
0xC00001F0	0007E894 34104798 D3F642AC FFDAF7FF
0xC0000200	0000056C 0000058C 4770E7FE E7FEE7FE
0xC0000210	47704770 F0004770 0000B85F B4184816
0xC0000220	F4416801 60010170 68104A14 0001F040
0xC0000230	4C126010 68203408 0080F040 23006020

图 4-2. APP\_GPIO\_Running\_LED 工程编译出的 bin 文件

Address	0	4	8	C	ASCII
0000	20000408	000001C1	00000211	0000020D	.....
0010	0000020F	00000209	00000289	00000000	.....
0020	00000000	00000000	00000000	00000215	.....
0030	0000020B	00000000	00000213	00000217	.....
0040	000001DB	000001DB	000001DB	000001DB	.....
0050	000001DB	000001DB	000001DB	000001DB	.....
0060	000001DB	000001DB	000001DB	000001DB	.....
0070	000001DB	000001DB	000001DB	000001DB	.....
0080	000001DB	000001DB	000001DB	000001DB	.....
0090	000001DB	000001DB	000001DB	000001DB	.....
00A0	000001DB	000001DB	000001DB	000001DB	.....
00B0	000001DB	000001DB	000001DB	000001DB	.....
00C0	000001DB	000001DB	000001DB	000001DB	.....
00D0	000001DB	000001DB	000001DB	000001DB	.....
00E0	000001DB	000001DB	000001DB	000001DB	.....
00F0	000001DB	000001DB	000001DB	000001DB	.....
0100	000001DB	000001DB	000001DB	000001DB	.....
0110	000001DB	000001DB	000001DB	000001DB	.....
0120	000001DB	000001DB	000001DB	000001DB	.....
0130	000001DB	000001DB	000001DB	000001DB	.....
0140	000001DB	000001DB	000001DB	000001DB	.....
0150	000001DB	000001DB	000001DB	000001DB	.....
0160	000001DB	000001DB	000001DB	000001DB	.....

在配置完 SYSCFG\_CFG0 寄存器 BOOT\_MODE 改为 0b`100 后，需要将程序跳转到 APP\_GPIO\_Running\_LED 工程所在的地址运行代码，具体配置如 [表 4-2. 跳转到 APP GPIO Running LED 工程所在的地址运行代码](#) 所示。

表 4-2. 跳转到 APP\_GPIO\_Running\_LED 工程所在的地址运行代码

```

typedef void (*pFunction)(void);
#define ApplicationAddress          0xC0000000

pFunction Jump_To_Application;
uint32_t JumpAddress = 0;

.....

if (((*(__IO uint32_t*)ApplicationAddress) & 0x2FFE0000) == 0x20000000) {
    JumpAddress = (*(__IO uint32_t*) (ApplicationAddress + 4));
    Jump_To_Application = (pFunction) JumpAddress;
    __set_MSP(*(__IO uint32_t*) ApplicationAddress);
    Jump_To_Application();
}

.....
    
```

复位芯片后，可观察到 APP 代码已经在正常运行了。

## 5. 版本历史

表 5-1. 版本历史

版本号.	说明	日期
1.0	首次发布	2021 年 11 月 30 日

## Important Notice

This document is the property of GigaDevice Semiconductor Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the "Product"), is owned by the Company under the intellectual property laws and treaties of the People's Republic of China and other jurisdictions worldwide. The Company reserves all rights under such laws and treaties and does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

The Company makes no warranty of any kind, express or implied, with regard to this document or any Product, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company does not assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Except for customized products which has been expressly identified in the applicable agreement, the Products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only. The Products are not designed, intended, or authorized for use as components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, atomic energy control instruments, combustion control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or Product could cause personal injury, death, property or environmental damage ("Unintended Uses"). Customers shall take any and all actions to ensure using and selling the Products in accordance with the applicable laws and regulations. The Company is not liable, in whole or in part, and customers shall and hereby do release the Company as well as its suppliers and/or distributors from any claim, damage, or other liability arising from or related to all Unintended Uses of the Products. Customers shall indemnify and hold the Company as well as its suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Products.

Information in this document is provided solely in connection with the Products. The Company reserves the right to make changes, corrections, modifications or improvements to this document and Products and services described herein at any time, without notice.