应用笔记

使用MMU进行多用户场景下的FLASH分区保护

简介

在嵌入式产品研发过程中，有时会存在单个 MCU 内部需要多个用户分阶段进行应用软件开发的场景，在此场景中，各用户的代码及数据可能出于版权或安全考虑，不方便公开给其它几个用户共享。那如何解决这类问题呢？

本文档主要针对国民技术 MCU 系列产品在上述应用场景，指导用户如何使用国民技术的 MCU，通过内置的存储器管理单元（Memory Management Unit，简称 MMU）实现 FLASH 主存储区的多用户区域划分及访问权限管理，从而解决多用户开发过程中的代码版权保护及数据安全问题。因此，可以广泛应用于各种版权保护、敏感数据和多用户代码保护等场景中。

本文档仅适应于内置 MMU 的国民技术 MCU 产品，目前支持的产品系列有 N32G05x，N32A052 系列产品，本文以 N32G05x 系列作为示例。

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900　　传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

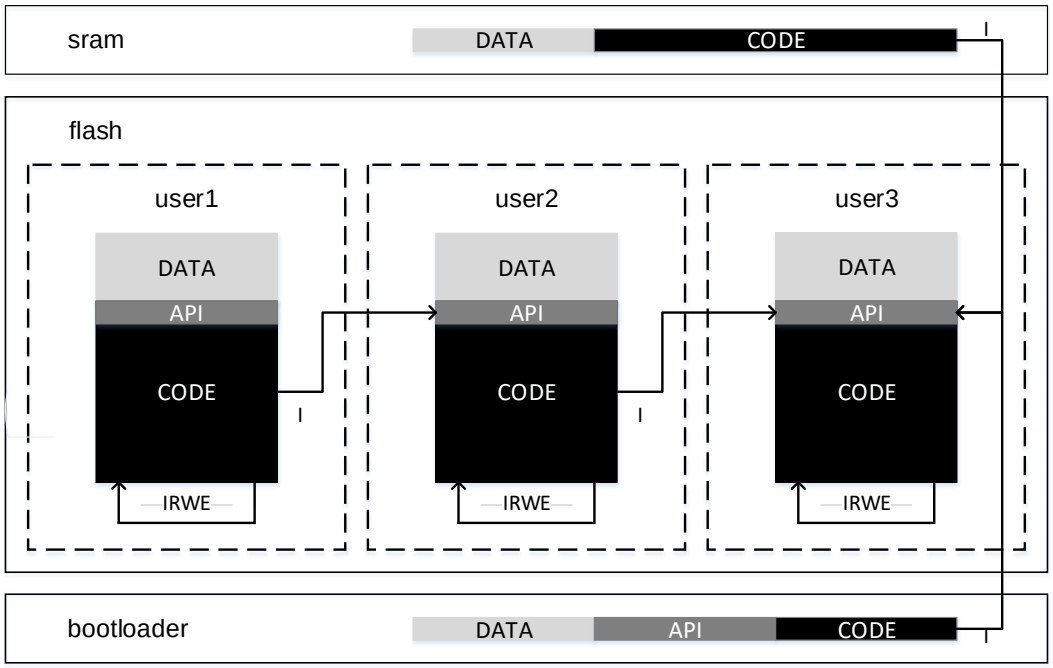
目录

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

1 分区保护实现机制

通常 MCU 片内的闪存（FLASH）挂接在内存总线上，CPU 可以无限制的访问 FLASH 内的任何区域。要实现对单颗 MCU 内片 FLASH 进行多个用户区域划分并保护，避免在片内不同用户通过 CPU 指令直接读取或修改其它用户区的 FLASH 内容。我们可以使用国民技术 MCU 内置的 MMU，将 FLASH 主存储区的区域进行划分和设置访问权限，同时可保护各个应用存储区域内的代码与数据不被非法访问及篡改，并指示出存储器及受保护的寄存器的非法访问错误，所有越权操作都将触发 MMU 异常报警，从而实现多用户下的 FLASH 分区保护功能。

图 1-1 MMU 分区保护实现机制

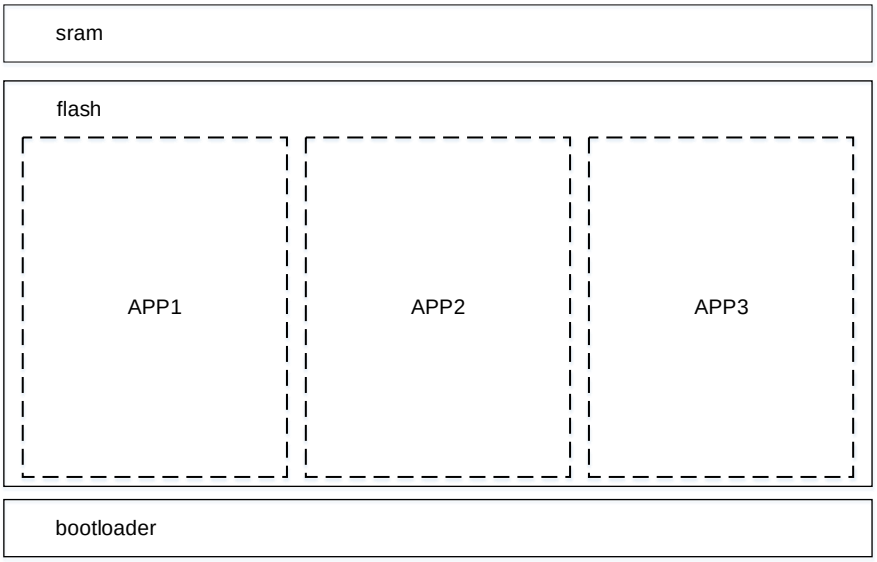


国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

2 MMU功能说明

MMU 可实现 FLASH 主存储区的区域划分和访问权限管理，可为 MCU 的不同应用划分独立的存储空间（见图 2-1 ），并对访问权限进行管理。

图 2-1 存储器区域划分



2.1 用户区域划分

FLASH 主存储区至多可划分为 USER1（默认）、USER2 和 USER3 三个区域。在实际使用中，用户区域划分有以下几种情况，各种情况的设置说明可参考表 2-1。

表 2-1 用户分区设置说明

情况描述	分区设置顺序	涉及权限管理的 FLASH 空间大小(1)			说明
		user1	user2	user3	
没有分区，默认为 user1 区域不封口	-	-	-	-	user1 区域大小为 flash_size，没有访问权限管理功能
没有分区，USER1 分区封口	user1	user1_size	-	-	user1 区域大小为 flash_size
两个分区，USER1 不封口、USER3 分区封口	user3→user2	-	0	user3_size	user1、user3 空间大小和为 flash_size，user1 没有访问权限管理功能
两个分区，USER1 封口、USER3 分区封口	user3→user2→user1	user1_size	0	user3_size	user1、user3 空间大小和为 flash_size
三个分区，USER1 不封口、USER2&USER3 分区封口	user3→user2	-	user2_size	user3_size	user1、user2 和 user3 空间大小和为 flash_size，user1 没有访问权限管理功能
三个分区，都分区封口	user3→user2→user1	user1_size	user2_size	user3_size	user1、user2 和 user3 空间大小和为 flash_size

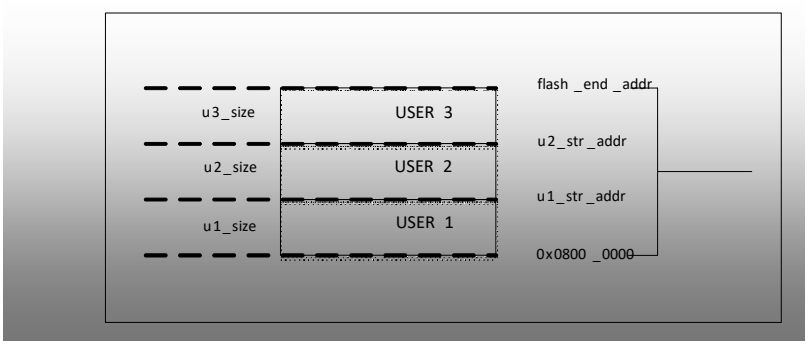
国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

说明：

1. “涉及权限管理的 FLASH 空间”指的是已设置分区大小的 FLASH 主存储区空间。

当 FLASH 主存储区划分为 3 个区域时，如图 2-2 所示，分别为 USER1（默认）、USER2 和 USER3，分区的颗粒度为 4KB。

图 2-2 FLASH 主存储区域划分关系



FLASH 主存储区的用户分区设置说明详见表 **2-2**。通过设置各用户分区的大小实现区域划分。分区设置属于静态设置，一旦设置，MCU 每次上电会自动加载配置。特别指出，分区设置只能操作一次，且操作不可逆。

表 2-2 FLASH 主存储区分区设置说明

分区用户	存储区域	分区大小范围
USER1	0x0800_0000 ~ (0x0800_0000 + u1_size - 1)	4KB[1] ~ (flash_size) KB
USER2	(0x0800_0000 + u1_size) ~ (flash _end_addr - u3_size)	0 KB ~ (flash_size - 8)KB
USER3	(flash _end_addr - u3_size + 1) ~ (flash _end_addr)[2]	0 KB ~ (flash_size - 4)KB

说明：

(1) 分区的颗粒度为 4KB；

(2) 不同型号的 flash _end_addr 会有差异，对应 flash_size 也不同，flash_size 应为 USER1、USER2 和 USER3 区域大小之和，其大小为(flash_end_addr – 0x0800_0000 + 1)。

注意：用户分区设置无法重置

2. 2 访问权限管理

通过用户区域划分来管理 FLASH 主存储区各区域的操作权限，实现存储器访问控制，表 2-3 提供了 FLASH 主存储区分区前后各用户区域的访问权限。

表 2-3 用户权限表

	被访问区域		
	user1	user2	user3
程序归属/	是否分区	是否分区	是否分区

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

访问方式	N[1]	Y	N	Y	N	Y
user1 code	IRWE[2, 3]	IRWE	IRWE	I	IRWE	I
user2 code	IRWE	I	IRWE	IRWE	IRWE	I
user3 code	IRWE	I	IRWE	I	IRWE	IRWE
SRAM1/2 code	IRWE	I	IRWE	I	IRWE	I
DMA	RW	-	RW	-	RW	-
JTAG/SWD	IRWE	I	IRWE	I	IRWE	I

说明：

(1) 分区前，USER1、USER2 和 USER3 视为同一个区域，所有 FLASH 空间默认为 USER1；

(2) I 表示取址，R 表示读，W 表示写，E 表示擦除；

(3) 写保护(WRP)使能与 MMU 分区的访问权限管理同级。

注意：如果不设置 USER1 区域大小（设置“操作步骤”见 3.2.1~3.2.3 小节），则 USER1 区域不具备访问权限管理功能。

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

3 操作说明

对 MCU 内置的 FLASH 主存储区进行分区操作，可通过国民技术提供的 PC 端的 Nsing MCU Download Tool 工具实现，关于工具的使用方法可参考《UG_通用 MCU 下载工具使用指南》。

3.1 操作环境

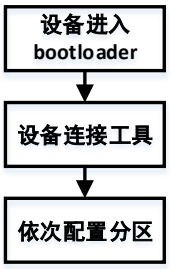
- 硬件环境：PC（系统 Windows XP/7/10）、开发板 N32G05XRBQ7-STB V1.0（含 N32G05XRBQ7 芯片）
- 目标设备： N32G05XRBQ7 芯片
- 软件环境：下载工具(Nsing MCU Download Tool.exe)、USB 转串口驱动（可选）

注意：Bootloader 支持 USB 接口或 USART 接口下载，使用前请确认已安装 USB DFU 驱动或 USB 转串口驱动。同时，确认目标设备已进入 Bootloader 状态，以便设备与下载工具正常连接。关于如何让目标设备进入 Bootloader 状态，可详细参考目标设备芯片的用户手册。本文档以 N32G05xRBQ7 芯片使用 UART 接口下载为例进行举例说明。

3.2 操作步骤

FLASH 主存储区用户区域划分流程如图 3-1，以下详细介绍分区设置操作步骤。

图 3-1 分区设置步骤



3.2.1 设备进入Bootloader

N32G05XRBQ7 BOOT0(PD0)引脚接 VDD，芯片上电进入 Bootloader。

注意：对于开发板 N32G05XRBQ7-STB V1.0，使用 UART 接口，则连接 USB Debug Port 接口供电。

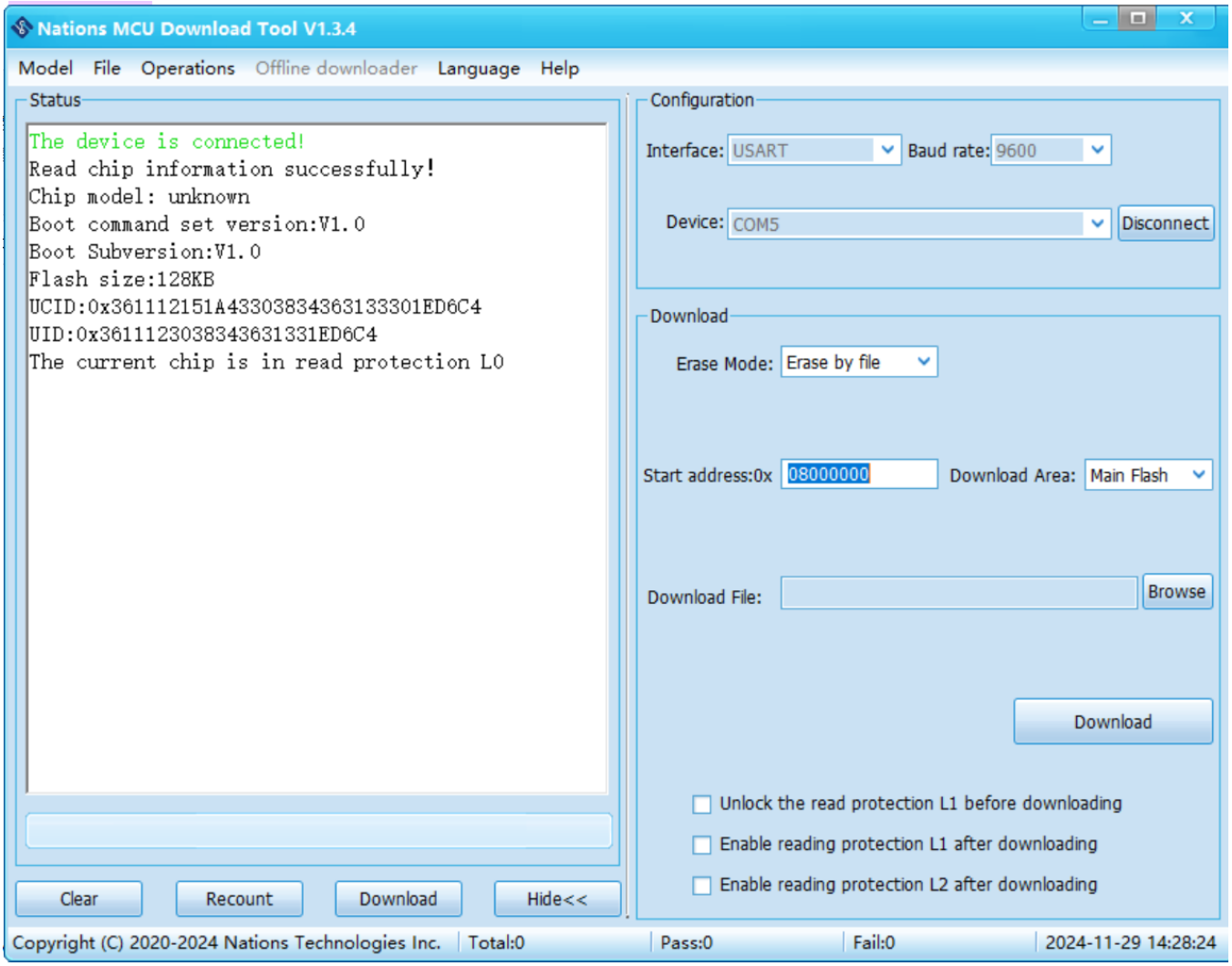
3.2.2 设备连接工具

双击 Nsing MCU Download Tool.exe，打开下载工具，界面如图 3-2 所示。此处，将重点关注“选择设备”区域。接口默认选择“USART”。选择匹配的端口号，作为设备。“COM 端口号”可通过 PC 的“设备管理器”查看，图 3-2 中连接 MCU 的串口被识别为“COM5”。同时，设置 USART 的波特率（如配置“9600”），单击“连接设备”按键，左边显示界面会提示“设备已连接”。

注意：N32G05XRBQ7 中 Bootloader 中默认使用 PA9 与 PA10 分别作为 UART 的 TX 与 RX，请确保 PA9 与 PA10 与串口的 TX 与 RX 已正确连接，可以通过修改选项字节修改 UART 引脚。

图 3-2 下载工具界面

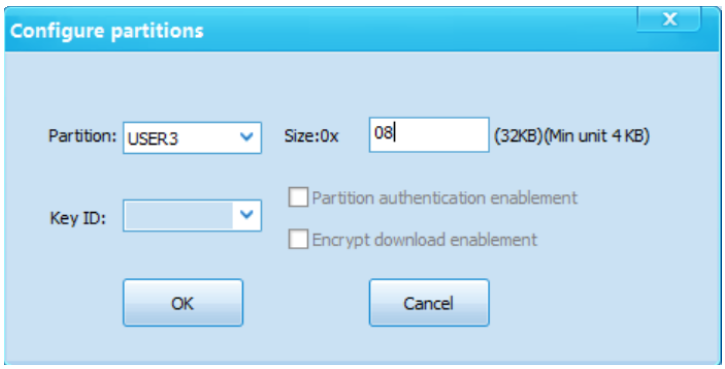
国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057



3.2.3 配置分区

单击“常用操作”下拉菜单中的“配置分区”按键，弹出配置分区对话框，按需选择分区用户 ID（USER1、USER2 或 USER3），并输入分区的 FLASH 大小（数值以分区颗粒度 4KB 为单位设置）。如图 3-3 所示，假设需为 USER3 划分 32KB 区域，则分区选择“USER3”，大小输入 0x08。点击“配置分区”，确认配置分区，完成当前用户 ID 的区域划分。

图 3-3 配置分区界面



国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

注意：

(1) 分区配置操作不可逆，请慎重操作；

(2) 如需设置多个分区，各用户可分别进入 Bootloader 配置，配置的大小、顺序等注意事项请参考表 2-1 及《UG_通用 MCU 下载工具使用指南》，操作不当可能导致配置失败。

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

4 示例工程

示例工程位于“Nations.N32G05x_Library.1.2.0\projects\n32g05x_EVAL\applications\MMU”。

为了展示 FLASH 主存储区分区后程序的执行方式，例如不同分区区域间的函数调用方法、正常或异常读取数据的不同效果以及中断处理方式等，将提供三个示例工程分别作为 USER1、USER2 和 USER3 三个分区用户 ID 的工程。

以下小节将重点介绍工程的 section 地址配置、生成 bin 文件、用户分区间的访问操作等内容。

4.1 Section地址配置

以 N32G05XRBQ7 芯片 128KB 的 FLASH 主存储区大小为例，假设 USER1、USER2、USER3 三个用户区域大小分别为 64KB、32KB 和 32KB。此时，FLASH 主存储区的区域划分关系如图 4-1 所示。各用户可根据各分区应用的实际代码量协商、划分 FLASH 主存储区。

图 4-1 Flash 主存储区区域划分关系示例

除了划分 FLASH 主存储区，为避免不同分区程序的全局变量存储空间冲突，同时也可以划分 N32G05XRBQ7 的 16KB SRAM 空间。各用户的 SRAM 可存储对应程序中的全局变量，其中 USER2 与 USER3 的全局变量需手动初始化。由于芯片程序执行的起始地址为 0x08000000，USER1 作为终端用户，同时负责处理堆栈与中断响应，所以 USER1 的 SRAM 还可用作堆栈空间。全局变量指定的地址需避开堆栈（栈顶地址详见工程的.map 文件）。

SRAM 区域划分是可选操作，原因是 N32G05XRBQ7 的 MMU 只管理 FLASH 主存储区的分区访问权限。SRAM 实际由 USER1、USER2、USER3 三个用户共享使用。将 SRAM 划分为多块区域，只是出于程序执行的稳定性考虑，防止不同分区的全局变量空间重叠，并不提供“保护用户 SRAM 中数据安全”的功能。根据实际应用，也可以不划分 SRAM，由用户相互协商分配全局变量的空间。本例中，USER1、USER2、USER3 共用 16KB SRAM。

用户区域划分后，各用户的应用程序需要下载到不同的地址空间，因此对应的工程需要分别配置各自的 section 地址，以免因为程序分配的地址空间与下载地址不一致造成程序下载失败或者运行异常。

4.1.1 Sct分散加载文件

KEIL 链接器根据 sct 分散加载文件的配置，分配各个 section 地址，生成分散加载代码，因此通过修改 sct 分散加载文件可以定制某 section 的存储位置。

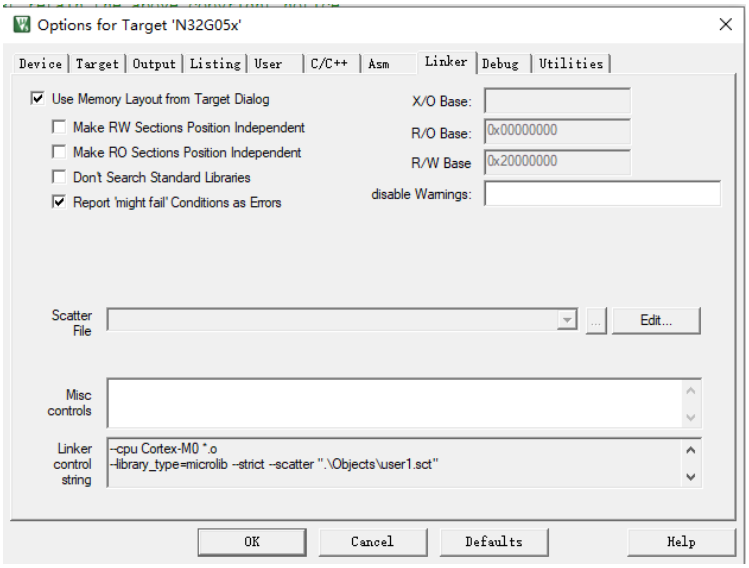
- **选择 sct 文件的生成方式**

Sct 文件可以使用 MDK 自动生成，也可以使用用户自定义的 sct 文件。通过 MDK 的“Options for Target -> Linker->Use Memory Layout from Target Dialog”选项，即可配置该选择，见

图 4-2。

图 4-2 选择 sct 文件的产生方式

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057



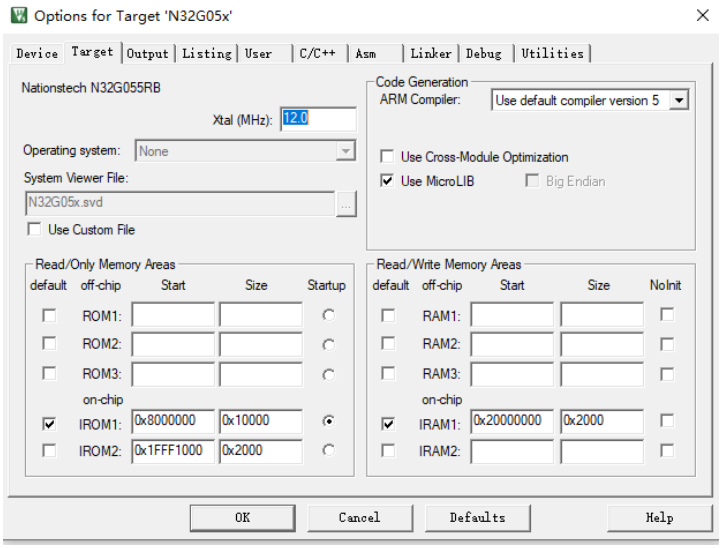
勾选“Use Memory Layout from Target Dialog”选项（SDK 默认勾选），则使用“Options for Target -> Target”页面的存储器分布配置选项生成 sct 文件，此时“Options for Target -> Linker-> Scatter File”是失效的，无法手动打开生成的 sct 文件进行编辑。工程构建完成，MDK 会生成新的 sct 文件覆盖旧文件。

如果需要手动编辑 sct 文件，则取消勾选“Use Memory Layout from Target Dialog”选项，同时在“Options for Target -> Linker-> Scatter File”框中指定 sct 文件的路径。之后，点击“Edit”则自动打开 sct 文件，用户可手动编辑该文件。

■ 通过 Target 控制配置存储器分布

勾选 MDK 的“Options for Target -> Linker->Use Memory Layout from Target Dialog”选项后，“Options for Target -> Target”页面存储器分布配置则自动生效。SDK 中默认的配置是在“Options for Target -> Device”页面选择芯片型号后自动加载的，FLASH 设置分区后，需重置存储器配置。

图 4-3 Target 存储器分布配置



以本例中 USER1 为例，工程的“Options for Target -> Target”页面的存储器分布配置如图 4-4 所示。其中，“on-chip”（片内存储器）部分 IROM1 起始地址为 0x08000000，大小为 0x10000，正好是 USER1 的 FLASH 起始地址和大小；而 IRAM1 起始地址为 0x20000000，大小为 0x2000，则分别是 USER1 的 SRAM 区域的起始地址与

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

大小。图中默认已勾选 IROM1 及 IRAM1，表示当前配置信息且会被使用；如果取消勾选，则该存储配置信息不会被采用。

而 USER2 与 USER3 的工程可以类似方式重置存储器配置，具体可参考对应示例工程的配置情况。

MDK 通过图 4-3 的 Target 存储器分布配置生成的 sct 文件的路径为“.\Objects\user1.sct”（SDK 默认设置），Sct 文件内容如图 4-4。用户可参考该文件格式，手动编辑 sct 文件。

图 4-4 Sct 文件内容

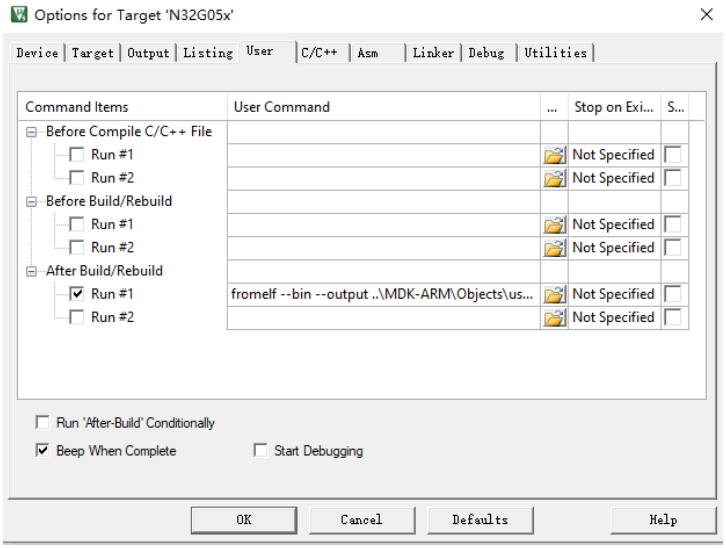
```
; *************************************************************
; *** Scatter-Loading Description File generated by uVision ***
; *************************************************************

LR_IROM1 0x08000000 0x00010000  {    ; load region size_region
  ER_IROM1 0x08000000 0x00010000  {  ; load address = execution address
   *.o (RESET, +First)
   *(InRoot$$Sections)
   .ANY (+RO)
   .ANY (+XO)
  }
  RW_IRAM1 0x20000000 0x00002000  {  ; RW data
   .ANY (+RW +ZI)
  }
}
```

4. 2 生成bin文件

通过 Nsing MCU Download Tool 工具下载程序，需要下载程序的 bin 文件。这里，介绍利用 fromelf 指令生成 bin 文件的方法。用户也可以自行编写 python 脚本，并输入用户指令执行该脚本。

在 MDK 的“ Options for Target->User” 配置页面中，“After Build/Rebuild”一栏添加调用 fromelf 工具形成生成 bin 文件指令（根据 axf 文件生成 bin），见图 4-5。

图 4-5 User 配置页面



生成 bin 文件指令首先调用 fromelf 工具，随后是工具的选项及输出文件名、输入文件名。假如将 bin 文件和 axf 文件生成在相同文件夹“..\MDK-ARM\Objects”内，则示例工程的用户指令可写为“fromelf --bin --output ..\MDK-ARM\Objects\user1.bin ..\MDK-ARM\Objects\user1.axf”。

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

4.3 分区访问操作

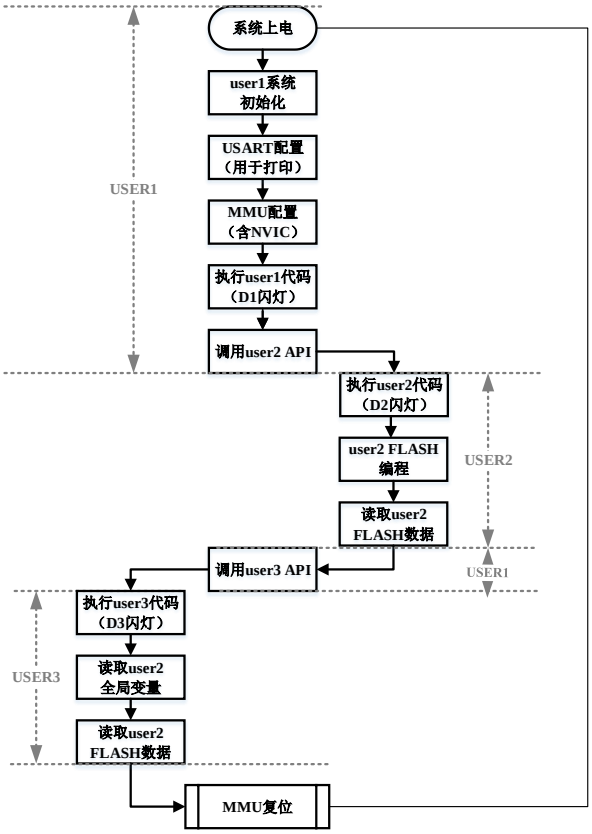
USER1、USER2 和 USER3 的示例工程配合演示了不同分区间的相互访问操作。将 USER1、USER2 和 USER3 的示例工程分别下载到 N32G05XRBQ7 芯片上，然后按照配置分区进行分区后，重新上电后，已设置 3 个分区的芯片将会按

图 4-6 展示的流程执行代码（示例工程参考 4.1 小节“Section 地址配置”的分区大小配置）。芯片程序执行起始地址为 0x08000000，因此 USER1 作为终端用户负责控制整个应用流程，包括系统初始化、堆栈处理、中断处理等操作。

MMU 限制了 FLASH 不同分区间的读、写操作，分区间的相互访问通过调用 API 实现。例如，USER2 与 USER3 各自实现具有一定功能的应用（以 API 形式封装），USER1 则通过调用 API 访问 USER2 或 USER3 的应用功能。所有涉及 MMU 的越权操作（调试接口/程序读或写越权、中断向量表地址读或写越权等），都将触发 MMU 异常报警，并以复位或者中断的方式及时告知用户。

以下将重点介绍三个分区访问操作：跨分区调用 API、跨分区读写数据以及中断处理。

图 4-6 示例工程执行流程



4.3.1 调用API

跨分区调用 API 本质是通过跳转至函数入口地址，执行程序。函数可以由编译器自动为其分配地址（详见工程的.map 文件），也可以由各分区用户指定地址（推荐）。在提供多个跨分区访问功能的 API 场景中，为函数指定固定地址，显然更有优势。而 MDK 中“ __attribute__”关键字可实现指定地址的功能。

本例中，USER1 分别调用 USER2 和 USER3 的 API。这里介绍“USER1 调用 USER2 API”的操作，以供参考。

USER2 的 FLASH 范围为 0x0801_0000~0x0801_7FFF，SRAM 范围为 0x2000_2000~0x2000_2FFF。在示例工程

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

user2 中，将 user2_demo.c 文件中的示例 demo（函数“void Test_User2(void)”）定义在 0x08016000 地址（见图 4-7）。

图 4-7 指定函数地址

```
uint32_t test_data __attribute__((at(0x20002A00)));
void Test_User2(void) __attribute__((section(".ARM.__at_0x08016000")));

void Test_ProgramFlashWord(uint32_t Address, uint32_t Data)
{
void Test_InitData(void)
{
void Test_User2(void)
{
    uint32_t flash_write_data = 0x76543210;
    Test_InitData();
    /* USART Configuration */
    // log_init();
    /* Output a message on Hyperterminal using printf function */
    printf("\n\rHello! Here is USER2 Example!\n\r");

    /* LED2 Blinks */
    Test_LedBlink(LED2_PORT, LED2_PIN);

    /* Program USER2 FLASH */
    Test_ProgramFlashWord(0x08017800, flash_write_data);

    /* Read USER2 FLASH */
    printf("USER2 Get USER2 FLASH *0x08017800 = 0x%X\r\n", *(__IO uint32_t*)(0x08017800));

    /* Read USER2 FLASH */
    printf("USER2 Get USER2 SRAM *0x20002A00 = 0x%X\r\n", *(__IO uint32_t*)(0x20002A00));
}
```

USER2 将函数的跳转地址提供给其它分区用户，以便其通过跳转至该地址，实现 API 功能调用。为了方便多个用户共同开发，USER2 可在 user2_demo.h 文件中利用宏定义函数的跳转地址以及跳转操作（如图 4-8 所示）。之后，不同用户可通过头文件获知应用程序的跳转信息。示例工程演示的是无参数函数的跳转，用户可进一步扩展 API 函数定义（如有参数函数，返回指定类型函数等）。

图 4-8 跳转地址和函数指针

```
44  typedef void (*pFunction)(void);
45
46  #define USER2_FUNC_ADDR (0x08016001)
47  #define API_FuncEntry2 ((pFunction)(USER2_FUNC_ADDR))
```

对于 USER1，可以选择将 user2_demo.h 添加进示例工程 user1 中（更名 user2_demo_api.h）。之后，USER1 的程序通过调用 API“API_FuncEntry2();”即可跳转至 USER2 执行函数，实现 D2 闪灯等操作。

4. 3. 2 读写数据-MMU异常报警

FLASH 分区配置生效后，跨分区的数据读取和 FLASH 编程、SRAM 代码访问用户分区、DMA 或调试接口访问用户分区等操作均会触发 MMU 异常报警（默认报警方式为 MMU 复位，详见 4.3.3 小节“MMU 复位”）。USER2 与 USER3 的示例工程分别演示了正常及异常读写数据两种情况。

在示例工程 user2 的 user2_demo.c 文件中，示例 demo 演示了 USER2 读写所属分区区域内的数据，代码详见图 4-7，将 USER2 SRAM 中变量 flash_write_data 的值写入 USER2 FLASH 指定位置 0x0801_7800，并确认写入地址的数据是否正确。以上操作均为常规操作，不再赘述具体操作方法。需特别指出，USER2 的示例工程没有执行启动流程，因此如果 USER2 有定义全局变量，其初值不一定是 0，用户使用全局变量前请注意初始化变量。

USER3 可以读写 USER2 SRAM。但由于 MMU 的分区权限管理功能，USER3 却无法写 USER2 FLASH 或读取 USER2 FLASH 的数据。在 USER3 的示例工程 user3 中，文件 user3_demo.c 包含示例 demo 代码，如图 4-9 中 66

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

行的操作将触发 MMU 异常复位报警（默认）。

图 4-9 USER3 读取数据

```
void Test_User3(void) __attribute__((section(".ARM.__at_0x08019000")));

void Test_User3(void)
{
    /* Output a message on Hyperterminal using printf function */
    printf("\n\rHello! Here is USER3 Example\n\r");

    /* LED3 Blinks */
    Test_LedBlink(LED3_PORT, LED3_PIN);

    /* Read USER2 SRAM */
    printf("USER3 Get USER2 SRAM *0x20002A00 = 0x%X\r\n", *(__IO uint32_t*)(0x20002A00));

    /* Read USER2 FLASH */
    printf("USER3 Get USER2 FLASH *0x08017800 = 0x%X\r\n", *(__IO uint32_t*)(0x08017800));
}
```

4. 3. 3 MMU复位

MMU 异常报警的方式有两种：复位（默认）或中断。本例中，我们将演示 MMU 复位。

通过打印 RCC->CTRLSTS 寄存器查看复位标志。

```
void Test_User1(void)
{
    /* Output a message on Hyperterminal using printf function */
    printf("\n\rHello! Here is USER1 Example\n\r");

    printf("\n\rRCC_CTRLSTS Value: 0x%x\n\r",RCC->CTRLSTS);

    /* LED(PB0) Blinks */
    Test_LedBlink(LED1_PORT, LED1_PIN);

    /* Jump to API of USER2 */
    API_FuncEntry2();

    /* Jump to API of USER3 */
    API_FuncEntry3();

    printf("USER1 Example End\r\n");
}
```

通过测试会发现 USER3 在读取 USER2 数据时发生了复位，MMURSTF 标志置起。

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

```
Hello! Here is USER1 Example


RCC_CTRLSTS Value: 0x18


Hello! Here is USER2 Example!

USER2 Get USER2 FLASH *0x08017800 = 0x76543210
USER2 Get USER2 SRAM *0x20002A00 = 0x1234567

Hello! Here is USER3 Example

USER3 Get USER2 SRAM *0x20002A00 = 0x1234567
U□

Hello! Here is USER1 Example


RCC_CTRLSTS Value: 0x1c
```


国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

5 结论

利用国民技术 MCU 芯片内置的 MMU，可将其 FLASH 至多划分为 3 个区域（USER1、USER2 或 USER3），并为各个用户区域提供访问权限控制功能。它既能防护存储器内部的攻击（如不同用户区域间相互访问、SRAM 访问等），也能抵御部分外部攻击（如调试接口访问、DMA 访问等）。

用户通过 Bootloader 可以设置分区，也可以下载程序。一旦分区设置成功，用户区域划分及权限管理功能将及时生效。同时，分区配置只能设置一次、无法重置且操作不可逆。这些特点使得 MMU 可以防止对 FLASH 的非法访问，有效保护存储在 FLASH 中的数据和代码。从而，在版权保护、敏感数据保护等应用场景中发挥安全作用。

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900　　传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

6 历史版本

版本	日期	备注
V1.0.0	2024.5.22	新建文档
V1.1.0	2026.1.6	1. 更新页眉/页脚 2. 更新示例工程到 SDK 中 3. 添加 N32A052 系列

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057

7 声明

国民技术股份有限公司（下称“国民技术”）对此文档拥有专属产权。依据中华人民共和国的法律、条约以及世界其他法域相适用的管辖，此文档及其中描述的国民技术产品（下称“产品”）为公司所有。

国民技术在此并未授予专利权、著作权、商标权或其他任何知识产权许可。所提到或引用的第三方名称或品牌（如有）仅用作区别之目的。

国民技术保留随时变更、订正、增强、修改和改良此文档的权利，恕不另行通知。请使用人在下单购买前联系国民技术获取此文档的最新版本。

国民技术竭力提供准确可信的资讯，但即便如此，并不推定国民技术对此文档准确性和可靠性承担责任。

使用此文档信息以及生成产品时，使用者应当进行合理的设计、编程并测试其功能性和安全性，国民技术不对任何因使用此文档或本产品而产生的任何直接、间接、意外、特殊、惩罚性或衍生性损害结果承担责任。

国民技术对于产品在系统或设备中的应用效果没有任何故意或保证，如有任何应用在其发生操作不当或故障情况下，有可能致使人员伤亡、人身伤害或严重财产损失，则此类应用被视为“不安全使用”。

不安全使用包括但不限于：外科手术设备、原子能控制仪器、飞机或宇宙飞船仪器、所有类型的安全装置以及其他旨在支持或维持生命的应用。

所有不安全使用的风险应由使用人承担，同时使用人应使国民技术免于因为这类不安全使用而导致被诉、支付费用、发生损害或承担责任时的赔偿。

对于此文档和产品的任何明示、默示之保证，包括但不限于适销性、特定用途适用性和不侵权的保证责任，国民技术可在法律允许范围内进行免责。

未经明确许可，任何人不得以任何理由对此文档的全部或部分进行使用、复制、修改、抄录和传播。

国民技术股份有限公司 NSING TECHNOLOGIES INC.
地址：深圳市南山区高新北区宝深路109号国民技术大厦
电话：+86-755-86309900 传真：+86-755-86169100
网址：https://www.nsingtech.com 邮编：518057