使用 Data Flash 模拟 EEPROM

NuMicro® 32 位系列微控制器范例代码介绍

文件信息

代码简述	用一块 Data Flash 模拟 EEPROM，并做 EEPROM 写/读
BSP 版本	NUC230/240 Series BSP CMSIS v3.01.002
开发平台	NuTiny－EVB－NUC240 V1.2

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1 功能介绍

1.1 简介

此范例程序将示范如何用一块 Data Flash 来模拟 EEPROM 做读/写。

1.2 原理

在 NuMicro® NUC240 系列 MCU 内配备了嵌入式闪存，以运行程序和储存数据。
写/读闪存操作的限制是访问地址必须是 32 位对齐地址，数据数据值也是 32 位数据。 每个位的数据值可以从 1 写成 0，并在执行擦除闪存操作完成后会变为 1。 NuMicro® NUC240系列仅支持页擦除闪存的操作，一页大小为 512 字节。

在写/读 EEPROM 时，写/读操作支持字节地址对齐，数据长度为一个字节，每个位数据是可以直接写入 1 或 0。

在此示例代码中，将预先分配指定一块 Data Flash 区域来仿真为 EEPROM 存储空间。函式 SIM_EEPROM_READ 用于从指定的 EEPROM 地址读取一笔 8 位数据，函式 SIM_EEPROM_WRITE 用于对指定的 EEPROM 地址写入一笔 8 位数据。
对 EEPROM 做写入的范例流程，如下 图 1-1 所示。

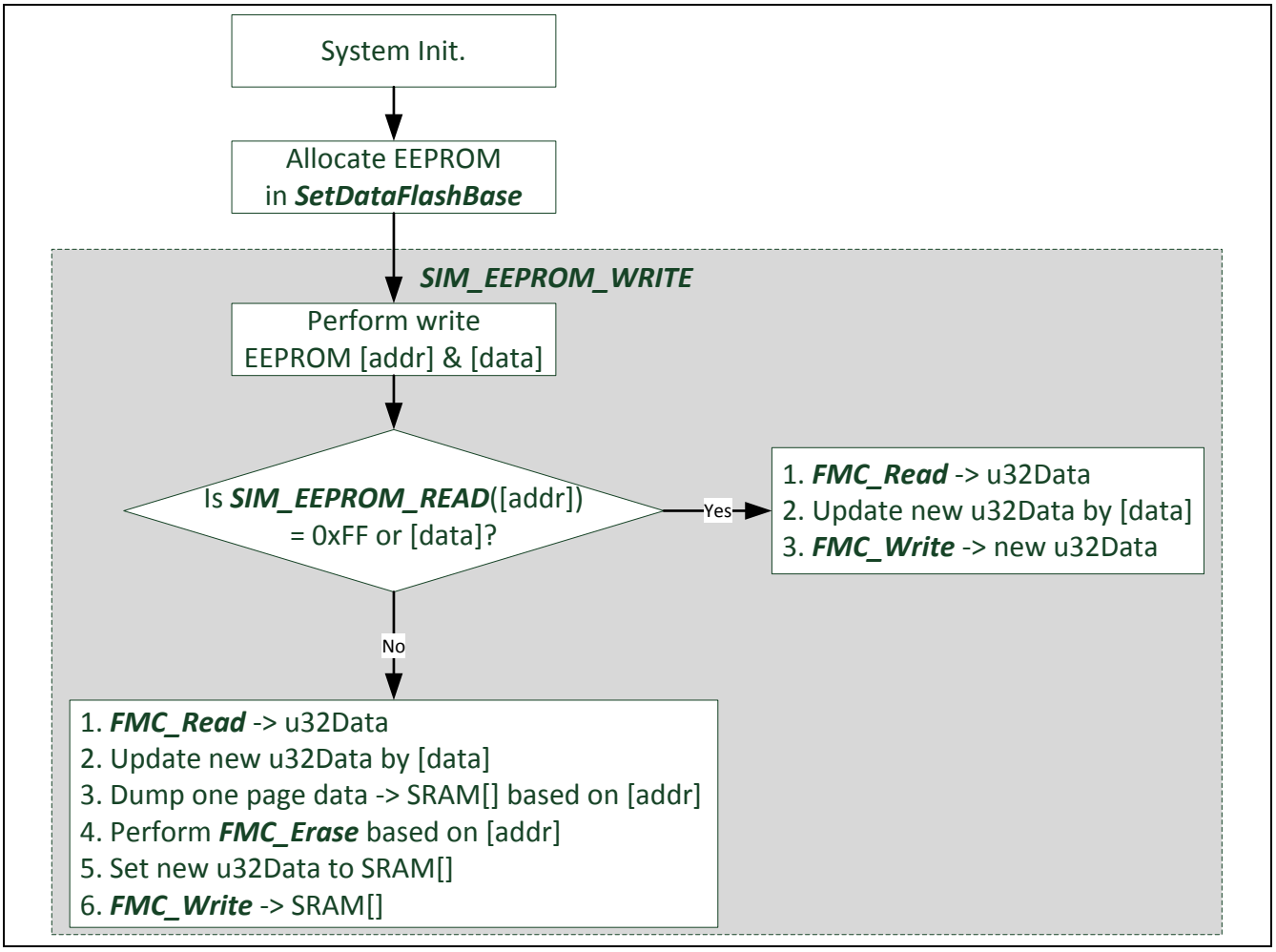

图 1-1 写入 EEPROM 的流程

1.3 执行结果

程序执行结果可经由串口鲍率 115200 的输出到 PC 做显示，如下图 1-2 所示。

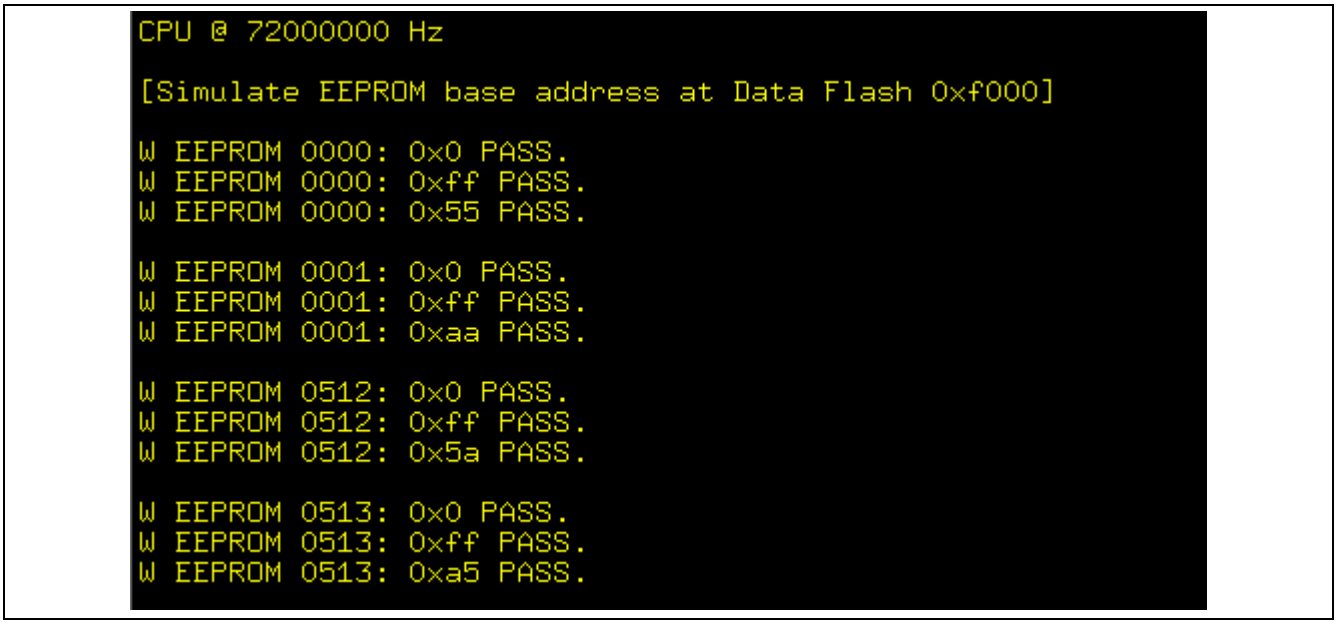

图 1-2 串口执行结果

2 代码介绍

分配一块 Data Flash 区域来仿真 EEPROM 存储空间。

```
int32_t SetDataFlashBase(uint32_t u32DFBA)
{
    uint32_t au32Config[2];

    /* Read current User Configuration */
    FMC_ReadConfig(au32Config, 1);

    /* Just return when Data Flash has been enabled */
    if(!(au32Config[0] & 0x1))
        return 0;

    /* Enable User Configuration Update */
    FMC_EnableConfigUpdate();

    /* Erase User Configuration */
    FMC_Erase(FMC_CONFIG_BASE);

    /* Write User Configuration to Enable Data Flash */
    au32Config[0] &= ~0x1;
    au32Config[1] = u32DFBA;
    if(FMC_WriteConfig(au32Config, 2))
        return -1;

    /* Perform chip reset to make new User Config take effect */
    SYS->IPRSTC1 |= SYS_IPRSTC1_CHIP_RST_Msk;

    return 0;
}
```

读 EEPROM 的函式。

```
uint8_t SIM_EEPROM_READ(uint32_t address)
{
    uint32_t u32Data;

    u32Data = FMC_Read(((address/4)*4) + EEPROM_BASE);
    return ((u32Data>>((address%4)*8)) & 0xFF);
```

```
}
```

写 EEPROM 的函式。

```
void SIM_EEPROM_WRITE(uint32_t address, uint8_t data)
{
    ……
    if((SIM_EEPROM_READ(address)==0xFF) || (SIM_EEPROM_READ(address)==data))
    {
        /* Original flash data is 0xFF, or data is matched. Do not perform flash page
erase. */

        /* Read oroginal data */
        u32Data = FMC_Read(((address/4)*4) + EEPROM_BASE);
        /* Assign 32-bit updated data */
        u32Data = ~(0xFF<<((address%4)*8));
        u32Data |= (data<<((address%4)*8));
        /* Write updated data */
        FMC_Write((((address/4)*4) + EEPROM_BASE), u32Data);
    }
    else
    {
        /* Need to update one page flash data. Perform flash page erase is mandatory
before writing updated data. */

        /* Assign 32-bit updated data */
        u32Data = FMC_Read(((address/4)*4) + EEPROM_BASE);
        u32Data &= ~(0xFF<<((address%4)*8));
        u32Data |= (data<<((address%4)*8));

        /* Dump target page data to SRAM buffer */
        u32Start = (EEPROM_BASE + ((address/FMC_FLASH_PAGE_SIZE)*FMC_FLASH_PAGE_SIZE)); //
page-size alignment
        u32End   = (u32Start + FMC_FLASH_PAGE_SIZE);
        pu32DataBuf = (uint32_t *)au32DataBuf;
        for(i=u32Start; i<u32End; i+=4)
        {
            *pu32DataBuf++ = FMC_Read(i);
        }

        /* Erase target page */
        FMC_Erase(EEPROM_BASE+((address/FMC_FLASH_PAGE_SIZE)*FMC_FLASH_PAGE_SIZE));
```

```
        /* Set updated data to SRAM buffer */
        au32DataBuf[((address%FMC_FLASH_PAGE_SIZE)/4)] = u32Data;

        /* Write target flash data from SRAM buffer */
        pu32DataBuf = (uint32_t *)au32DataBuf;
        for(i=u32Start; i<u32End; i+=4)
        {
            FMC_Write(i, *pu32DataBuf++);
        }
    }
}
```

3 软件与硬件环境

● 软件环境

- ■ BSP 版本
  - ◆ NUC230/240 Series BSP CMSIS v3.01.002
- ■ IDE 版本
  - ◆ Keil uVersion 5.28

● 硬件环境

- ■ 电路组件
  - ◆ NuTiny–EVB–NUC240 V1.2
- ■ 示意图
  - ◆ 将 NUC240 UART TX (PB.1) 引脚连接到 PC UART RX，以在 PC 上显示示例代码的执行结果。

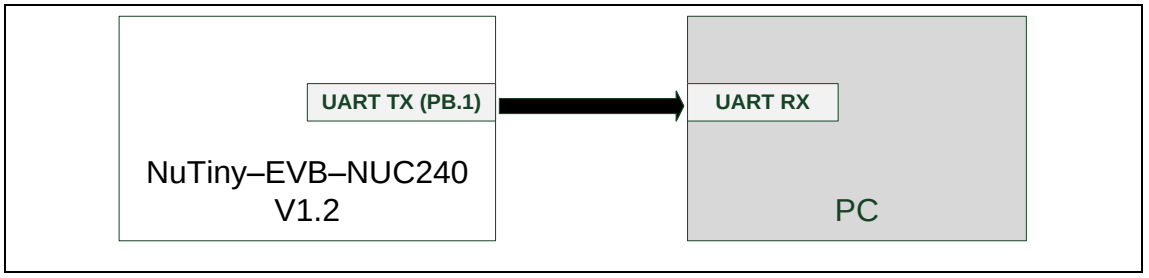

图 3-1 NUC240 UART TX 和 PC UART RX 连接

4 目录信息

- EC_NUC240_ Simulate_EEPROM_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

5 如何执行范例程序

1. 根据目录信息章节进入 ExampleCode 路径中的 KEIL 文件夹，双击 Simulate_EEPROM.uvproj。
2. 进入编译模式接口
   a. 编译
   b. 下载代码至内存
   c. 进入 / 离开除错模式
3. 进入除错模式接口
   a. 执行代码

6 修订纪录

Date	Revision	Description
Jul 30, 2019	1.00	1. 初始发布。

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.