



中微 BAT32A 系列

应用注意事项

Rev. 1.0.0

请注意以下有关CMS知识产权政策

* 中微半导体（深圳）股份有限公司（以下简称本公司）已申请了专利，享有绝对的合法权益。与本公司MCU或其他产品有关的专利权并未被同意授权使用，任何经由不当手段侵害本公司专利权的公司、组织或个人，本公司将采取一切可能的法律行动，遏止侵权者不当的侵权行为，并追讨本公司因侵权行为所受的损失、或侵权者所得的不法利益。

* 中微半导体（深圳）股份有限公司的名称和标识都是本公司的注册商标。

* 本公司保留对规格书中产品在可靠性、功能和设计方面的改进作进一步说明的权利。然而本公司对于规格内容的使用不负责任。文中提到的应用其目的仅仅是用来做说明，本公司不保证和不表示这些应用没有更深入的修改就能适用，也不推荐它的产品使用在会由于故障或其它原因可能会对人身造成危害的地方。本公司的产品不授权适用于救生、维生器件或系统中作为关键器件。本公司拥有不事先通知而修改产品的权利，对于最新的信息，请参考官方网站 www.mcu.com.cn

目录

1. 概要	3
1.1 目的	3
1.2 章节概要	3
1.3 参考文档	3
2. 复位	4
3. 电压检测电路 (LVD)	5
3.1 电压检测电路 (LVD) 概述	5
3.2 LVD 中断&复位模式注意事项	5
3.3 LVD 使能	6
4. GPIO	7
5. TIMER4 PWM 输出管脚	8
6. SWD 调试引脚	9
7. SPI	10
8. DMA	11
9. 串口	13
10. 低功耗测试	13
11. FLASH	14
12. DATA FLASH	15
13. 时钟运行模式控制	16
14. 副系统时钟起振稳定等待与看门狗复位	17
14.1 推荐操作	17
15. 版本修订说明	18

1. 概要

1.1 目的

本应用笔记介绍了中微半导体以 Cortex-M0+ 为内核的 BAT32A 系列单片机的应用注意事项。

1.2 章节概要

本文档主要包括以下几个章节：

第 2 章 介绍了复位的应用注意事项。

第 3 章 介绍了电压检测电路（LVD）的应用注意事项。

第 4 章 介绍了 GPIO 的应用注意事项。

第 5 章 介绍了 TIMER4 PWM 输出管脚的应用注意事项。

第 6 章 介绍了 SWD 调试引脚的应用注意事项

第 7 章 介绍了 SPI 的应用注意事项。

第 8 章 介绍了 DMA 的应用注意事项。

第 9 章 介绍了串口的应用注意事项。

第 10 章 介绍了低功耗测试的应用注意事项。

第 11 章 介绍了 Flash 的应用注意事项。

第 12 章 介绍了 Data Flash 的应用注意事项。

第 13 章 介绍了时钟运行模式控制的应用注意事项。

第 14 章 介绍了副系统时钟起振稳定等待与看门狗复位的应用注意事项。

1.3 参考文档

BAT32A 系列用户手册

BAT32A 系列 Datasheet

2. 复位

芯片 RESETB 复位引脚不能悬空，可直接或者通过电阻连接 VDD，一种简单的复位电路如下图所示：

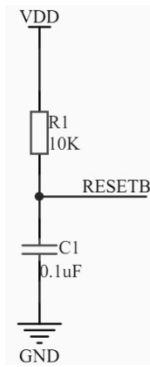


图 2-1：复位电路

3. 电压检测电路（LVD）

3.1 电压检测电路（LVD）概述

电压检测电路(LVD)通过选项字节设定运行模式和检测电压 (V_{LVDH} 、 V_{LVDL} 、 V_{LVD})，运行模式可以设置中断模式、复位模式、中断&复位模式。具体运行过程如下表：

模式	中断&复位模式	复位模式	中断模式
运行过程	$VDD \geq V_{LVDH}$ ，解除内部复位； $VDD < V_{LVDH}$ ，产生中断请求信号； $VDD < V_{LVDL}$ ，产生内部复位。	$VDD \geq V_{LVD}$ ，解除内部复位； $VDD < V_{LVD}$ ，产生内部复位。	上电 $VDD \geq V_{LVD}$ ，解除内部复位； 在解除内部复位后： $VDD < V_{LVD}$ 或者 $VDD \geq V_{LVD}$ ，产生中断请求信号（INTLVI）。

3.2 LVD 中断&复位模式注意事项

1. 当VDD开始下降时，LVD稳定检测到 $VDD < V_{LVDL}$ 后，产生复位信号将系统复位。 $VDD < V_{LVDL}$ 的稳定时间为3个 F_{IL} 时钟（约200 μ s），如果 $VDD < V_{LVDL}$ 的时间小于3个 F_{IL} 时钟，则无法产生复位信号。
2. 当LVD检测到 $VDD < V_{LVDH}$ 产生中断后，LVIMD和LVILV标志会被置‘1’，需要在中断程序中将LVIMD和LVILV标志清零，否则当VDD下次降低时将无法产生中断，具体操作流程详见图3-1。
3. 当LVD检测到 $VDD < V_{LVDL}$ 发生系统复位后，LVIMD和LVILV标志会被置‘1’，芯片重新工作后需要在初始化程序中将LVIMD和LVILV标志清零，否则当VDD下次降低时将无法产生系统复位信号，具体操作流程详见图3-2。

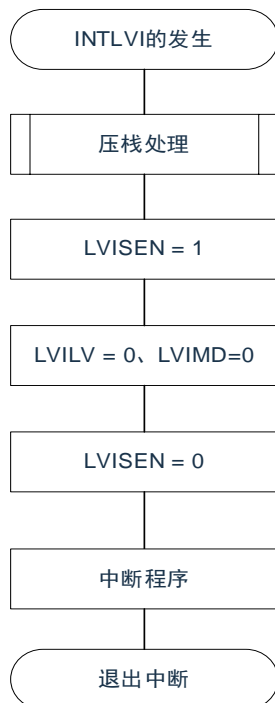


图3-1

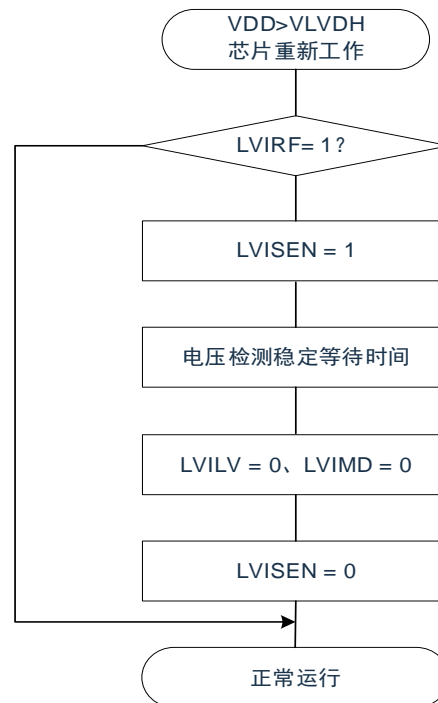


图3-2

3.3 LVD 使能

建议打开芯片 LVD 功能，样例工程默认开启 LVD 功能为复位模式，如果需要修改 LVD 运行模式或检测电压，请按照如下的步骤进行配置：

1. 打开 system_BAT32Axxxx.c 文件
2. 找到 user_opt_data 数组
3. 修改 user_opt_data[1] 的值即可修改 LVD 电压（例如 user_opt_data[1] = 0x37 表示系统电压跌落到 2.04V 产生复位，相应地当电压爬升到高于 2.09V 时，再次产生复位信号）

```
//----- <<< Use Configuration Wizard in Context Menu >>> -----
const uint8_t user_opt_data[4] __attribute__((used)) __attribute__((section(".ARM.__AT_0x00000C0"))) =
{
```

图 3-3: user_opt_data 数组

```
// <h> LVD Control Option Byte (C1H)
//   <o.0..7> Voltage detection setting (VLVD)
//   <0xFF=> ( LVD OFF )
//   <0x3D=> VLVD = 1.88V/1.84V ( interrupt mode )
//   <0x39=> VLVD = 1.98V/1.94V ( interrupt mode )
//   <0x35=> VLVD = 2.09V/2.04V ( interrupt mode )
//   <0x5D=> VLVD = 2.50V/2.45V ( interrupt mode )
//   <0x59=> VLVD = 2.61V/2.55V ( interrupt mode )
//   <0x55=> VLVD = 2.71V/2.65V ( interrupt mode )
//   <0x7D=> VLVD = 2.81V/2.75V ( interrupt mode )
//   <0x79=> VLVD = 2.92V/2.86V ( interrupt mode )
//   <0x75=> VLVD = 3.02V/2.96V ( interrupt mode )
//   <0x3F=> VLVD = 1.88V/1.84V ( reset mode )
//   <0x3B=> VLVD = 1.98V/1.94V ( reset mode )
//   <0x37=> VLVD = 2.09V/2.04V ( reset mode )
//   <0x5F=> VLVD = 2.50V/2.45V ( reset mode )
//   <0x5B=> VLVD = 2.61V/2.55V ( reset mode )
//   <0x57=> VLVD = 2.71V/2.65V ( reset mode )
//   <0x7F=> VLVD = 2.81V/2.75V ( reset mode )
//   <0x7B=> VLVD = 2.92V/2.86V ( reset mode )
//   <0x77=> VLVD = 3.02V/2.96V ( reset mode )
//   <0x3A=> VLVDH = 1.98V/1.94V, VLVDL = 1.84V ( interrupt, reset mode )
//   <0x36=> VLVDH = 2.09V/2.04V, VLVDL = 1.84V ( interrupt, reset mode )
//   <0x32=> VLVDH = 3.13V/3.06V, VLVDL = 1.84V ( interrupt, reset mode )
//   <0x5A=> VLVDH = 2.61V/2.55V, VLVDL = 2.45V ( interrupt, reset mode )
//   <0x56=> VLVDH = 2.71V/2.65V, VLVDL = 2.45V ( interrupt, reset mode )
//   <0x52=> VLVDH = 3.75V/3.67V, VLVDL = 2.45V ( interrupt, reset mode )
//   <0x7A=> VLVDH = 2.92V/2.86V, VLVDL = 2.75V ( interrupt, reset mode )
//   <0x76=> VLVDH = 3.02V/2.96V, VLVDL = 2.75V ( interrupt, reset mode )
//   <0x72=> VLVDH = 4.06V/3.98V, VLVDL = 2.75V ( interrupt, reset mode )
//
//   <i> Please setting the item for interrupt & reset mode
// </h>
0x37,
```

图 3-4: user_opt_data[1]

详细的 LVD 操作请参考用户手册选项字节章节：电压检测电路。

适用产品：本应用注意事项适用于以下产品型号：所有 BAT32A 系列产品。

4. GPIO

1. BAT32A 系列芯片除了 P130 以外的 GPIO 都是默认输入态，P130 默认输出低电平，这对于多芯片的系统，可以接到从芯片的 RESET 脚，用于控制从芯片的工作时序。
2. 对于需要用到开漏输出的场合，比如 I2C 通信，就需要使用芯片的 P60~P63 引脚，这 4 个引脚是真正意义上的开漏输出，即上拉电阻连接的 VCC 电压可以高于芯片工作电压（耐压为 6V）
3. P20~P27 既可以做 GPIO，也可以作为 ADC 输入通道，为了保证 ADC 采集外部信号的精度，因而弱化了这几个 IO 口的驱动能力（ $I_{oh} < 2.5\text{mA}$, $I_{ol} < 10\text{mA}$ ）。
4. P121~P124 为外部高速、低速晶振管脚，用作 GPIO 时，仅作输入使用。

使用 GPIO 时，需要仔细检查，详见用户手册引脚功能章节：使用的端口功能和复用功能的寄存器设置例子，里面有详细的 GPIO 复用功能列表。此外，需要注意不同封装之间的差异。

5. TIMER4 PWM 输出管脚

1. TIMER4 一共有 4 个 16 位定时器，各 16 位定时器称为通道，即通道 0，通道 1，通道 2 和通道 3
2. 当配置为两主两从模式的时候，通道 0 和通道 2 作为周期计数器，用于控制 PWM 的周期，通道 1 和通道 3 作为占空比寄存器，用于设置 PWM 的占空比。该模式输出 2 路 PWM 波。
3. 当配置为一主三从模式的时候，通道 0 作为周期计数器，通道 1~3 作为占空比寄存器。该模式最多输出 3 路 PWM 波。
4. PWM 的输出管脚分别为 TO01，TO02 和 TO03。

6. SWD 调试引脚

SWDIO 和 SWCLK 两个调试引脚需要复用成其他功能（普通 IO 或者外设功能）时，需要将 DBGSTOPCR 寄存器的 SWDIS 位置 1，并且在芯片处于 debug 模式时，引脚优先用于调试功能。

在 SWDIS 位为 0 时，SWDIO 引脚为高电平，同时在 SWCLK 引脚输入 50 个以上的脉冲信号，会将芯片置为 debug 模式，在实际应用时，需要注意以下几个场景：

1. 使用 SWD 接口烧录完芯片，此时芯片仍处于 debug 模式，需要对芯片断电（外部引脚 reset 也无效），才能将 SWDIO 和 SWCLK 两个调试引脚复用成其他功能。
2. 程序启动时，考虑需要使用调试口进行烧录，SWDIS 位保持为默认 0 的状态，在程序运行过程中，再将 SWDIS 位置 1，将调试引脚复用成其他功能（一般见于 Bootloader 未复用调试口，APP 程序复用调试口），此时需要注意在 SWDIS 位为 0 状态时需保证 SWCLK 引脚的稳定，避免误输入 50 个以上的脉冲，导致芯片进入 debug 状态，调试引脚无法复用成其他功能。
3. 通过选项字节将 Flash 设置成保护状态时，尝试进入 debug 模式会导致芯片进入保护状态，无法正常运行程序（外部引脚 reset 也无法恢复），在需要使用 Flash 保护功能时，为避免误进入 debug 模式，建议在程序运行后，将 SWDIS 位置 1。

使用第三方烧录器进行烧写，因不确定烧录器的信号状态，SWDIO 和 SWCLK 两个调试引脚建议串联电阻，推荐值 500Ω（我司官方烧录器内部已串有电阻）。

7. SPI

- 1) 不同封装所具备的 SPI 个数不同，请参考数据手册中的通用串行通信单元章节。
- 2) SPI 主机模式最大工作频率为系统时钟的 2 分频，比如系统时钟设置为 48MHz，那么 SPI 的工作速度为 24MHz。
- 3) SPI 与 DMA 模块的联动
 - a) 在一些对通信速度有较高要求的场合，需要 SPI 发送的数据之间的间隔尽量小，这时候需要和 DMA 模块联动起来使用。
 - b) SPI 的 DMA 通道由 SPI 的中断来触发传输，SPI 的中断源有两种：一种是传输结束，一种是缓冲器空。（传输结束作为中断源表示数据真实在 SPI 对应的 IO 上发送完毕后产生中断，而缓冲器空作为中断源表示 SPI 的数据寄存器传输到 SPI 内部的移位寄存器模块后产生中断，这个时候 SPI 对应的 IO 上还没有产生任何波形）
 - c) 如果用传输结束作为 SPI 中断源，波形会不连续，中间有间隔。
 - d) 使用缓冲器空作为 SPI 中断源，波形连续，数据发送效率提高。对通信速度有较高要求的场合，建议使用缓冲器空作为 SPI 中断源。

8. DMA

1. DMA 和周边外设的框图如下图所示：

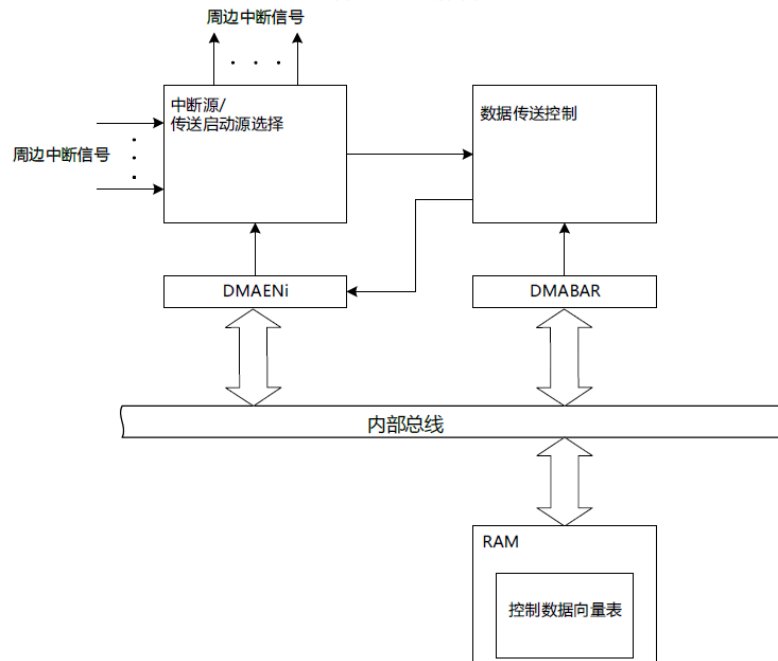


图 8-1：DMA 框图

2. ADC 与 DMA 联动使用

- 例如希望通过 DMA 将 ADC 转换完成的结果存放到内存中，16 次转换完成后产生中断以读取存储在内存中的 ADC 转换结果。
- 将 DMA 配置为重复模式，传输计数器设置为 16，启动 ADC 和 ADC 对应的 DMA 通道。
- 如图 8-1 中的中断信号一样，ADC 的中断信号传输到 DMA 模块触发 DMA 传输，每完成一次 ADC 转换，传输计数器减 1（初始值为 16），当传输计数器从 1 减到 0 的时候，响应 ADC 中断服务程序，用户程序可从内存读取这 16 次的 ADC 转换结果。

3. DMA 模块需要使用一部分 SRAM。用户需要根据实际应用对工程做配置。以 keil 为例，如果只使用一个 DMA 传输，占用的内存大小为 64+16 字节。修改默认工程生成的 scatter 文件中的 SRAM 起始地址至少偏移 80 字节。（DMA 使用的最大 SRAM 大小 = 64 字节 DMA 向量指针+16 字节 x DMA 使用的通道个数）。

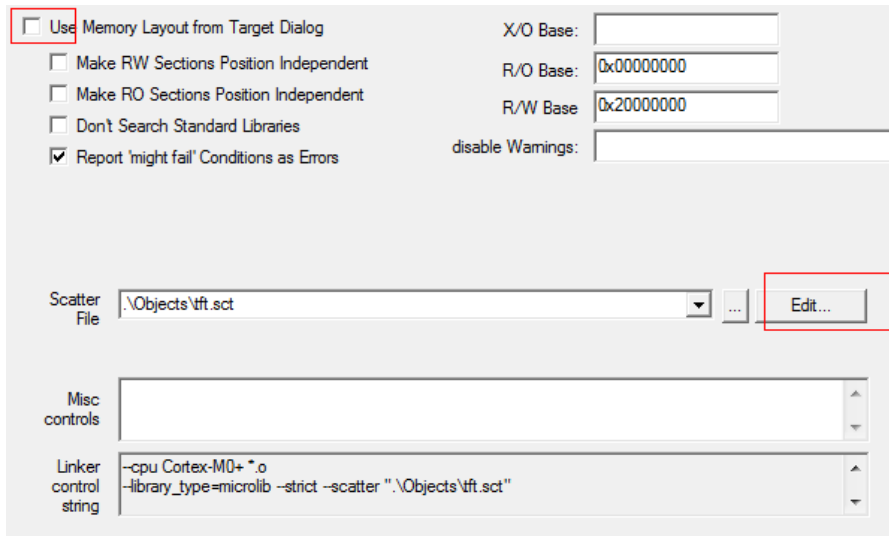


图 8-2：链接定位文件配置

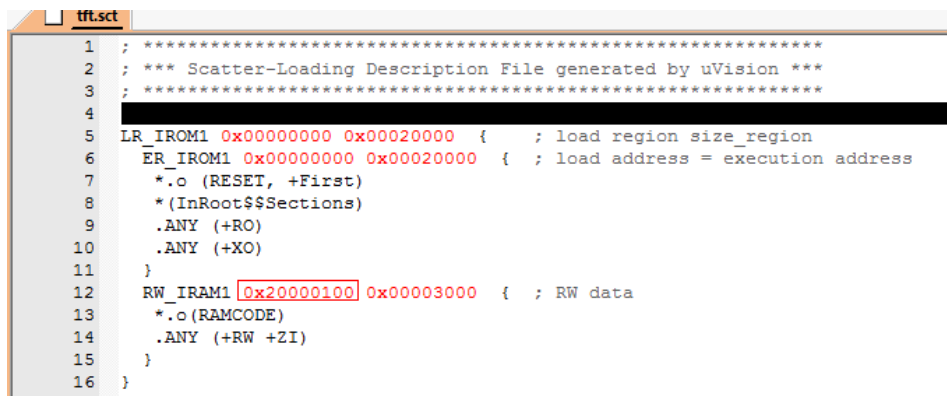


图 8-3：配置内存偏移

4. 若有多个通道的 DMA 需用到重复模式，应分时使用重复模式的 DMA 通道。

9. 串口

UART 模块支持常用的 7 位，8 位数据传输，对于某些需要用到 9 位传输的场合，用户可使用 UART0。

10. 低功耗测试

调试模式下，芯片不会进入低功耗。测试低功耗功能及低功耗电流时，需将程序烧写到芯片，断开调试器，然后脱机运行。

1. 为了测试到准确的低功耗电流，进入深度休眠前，应关闭所有外设，同时请注意芯片的各个 GPIO 口进行了正确的配置：输入不悬空，输出不能高低电平冲突。（包括封装未引出的引脚）
2. P121~P124 不使用时，接地。

11. Flash

BAT 系列芯片内置了 Flash 存储器，在对 flash 单元（包含 Data Flash 单元）进行编程时，即使对 Flash 写入 0xFF 这样的值，也会对 Flash 单元施加一次时长为写入周期的高电平。基于 flash 工作原理和制造工艺，要求 flash 单元在执行擦除前施加的累积高电平时长不超过一定值。

经过实际芯片测试验证，在执行一定次数的重复写入操作后（实测为整页按字节写入 0xFF，整页重复写入数百次），会发生该 Flash 页内的某个 bit 误读出低电平现象，执行擦除后，该 bit 的误读现象消失。

1. 避免对相同页地址的 Flash 区域超常规地多次重复写入数据操作。
2. 当程序设计需要重复写入相同数据时，在一定的写入操作后执行擦除操作再次重新写入，从而避免累积高电平时长过长引发异常。

12. Data Flash

芯片提供了 Data Flash，大小为 1.5KB **【注 1】**，Data Flash 的寻址范围 (0x00500000H~0x005005FFH)。

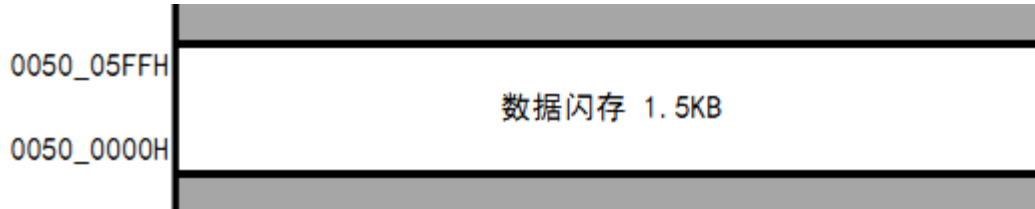


图 12-1: Data Flash 地址范围

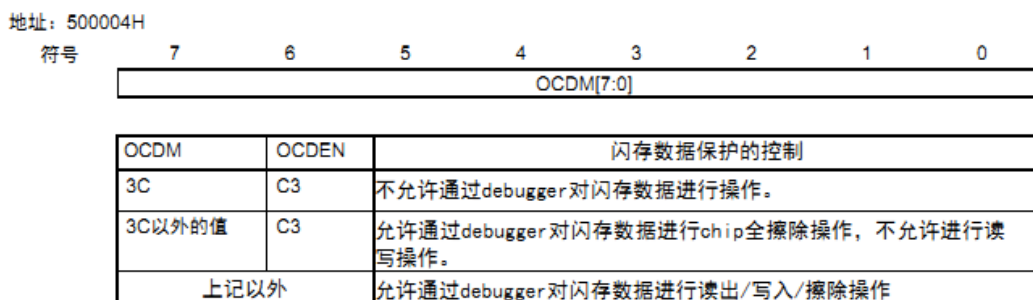


图 12-2: 闪存控制信息

0x00500004 地址是闪存控制信息配置字节，0x00500005 地址是交换功能配置字节。

这两个字节地址在 Data Flash 区域地址范围内，用户如果从 0x00500000 开始做存储数据用，容易误操作触发 swap 功能，或者造成闪存配置信息改变。

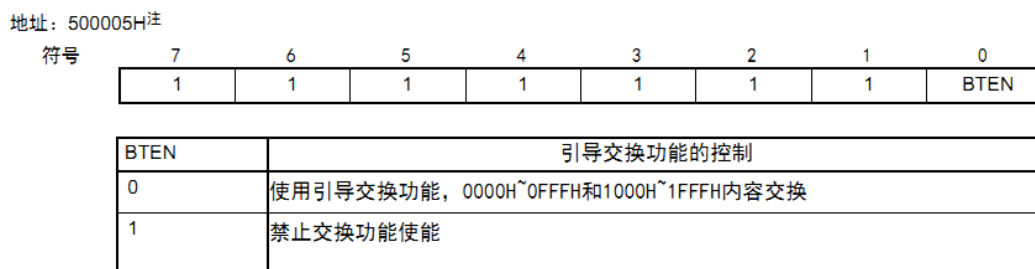


图 12-3: 交换功能

处理方式：写 Data Flash 时要跳过这两个地址所在的块，直接从下一个块开始操作。

如果需要存储在 0x500000 开始的块，要注意回写 0x500004, 0x500005 的配置信息，评估项目中是否会存在配置字节被 Data Flash 擦写改变的情况。

【注 1】：仅适用于 BAT32A237/A6700/A6703

13. 时钟运行模式控制

图 13-1: 时钟运行模式控制寄存器 (CMC) 的格式

地址: 40020400H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMC	EXCLK	OSCSEL	EXCLKS 注	OSCSELS 注	0	AMPHS1 注	AMPHS0 注	AMPH

EXCLK	OSCSEL	高速系统时钟引脚的运行模式	X1/P121引脚	X2/EXCLK/P122引脚
0	0	输入端口模式	输入端口	
0	1	X1振荡模式	连接晶体或者陶瓷谐振器。	
1	0	端口模式	输入端口	
1	1	外部时钟输入模式	输入端口	外部时钟输入

EXCLKS	OSCSELS	副系统时钟引脚的运行模式	XT1/P123引脚	XT2/EXCLKS/P124引脚
0	0	输入端口模式	输入端口	
0	1	XT1振荡模式	连接晶体谐振器。	
1	0	输入端口模式	输入端口	
1	1	外部时钟输入模式	输入端口	外部时钟输入

AMPHS1	AMPHS0	XT1振荡电路的振荡模式选择
0	0	低功耗振荡 (默认)
0	1	通常的振荡
1	0	超低功耗振荡
1	1	禁止设置。

AMPH	X1时钟振荡频率的控制
0	$1\text{MHz} \leq f_x \leq 10\text{MHz}$
1	$10\text{MHz} < f_x \leq 20\text{MHz}$

注: EXCLKS 位, OSCSELS 位, AMPHS1 位和 AMPHS0 位只在上电复位时被初始化, 而在其他复位时保持不变。

上电复位后, CMC 寄存器只能写入一次, 其他复位 (软复位、低压复位、看门狗复位) 后无法再次写入。

当使用 PLL 时钟时, 若内部高速时钟频率低于 PLL 时钟频率, 由内部高速时钟切换到 PLL 时钟的代码应放到 SRAM 执行。【注 1】

【注 1】: 仅适用于 BAT32A239/279

14. 副系统时钟起振稳定等待与看门狗复位

芯片提供了多种时钟：主系统时钟（X1 振荡电路，高速内部振荡器（高速 OCO）），副系统时钟（XT1 振荡电路 32.768KHz），低速内部振荡器时钟（低速 OCO），各时钟可进行切换，其中副系统时钟的起振稳定时间（约 2s）。

若设置看门狗定时器功能为复位系统，那么设置副系统时钟使能后，在等待时钟稳定的过程中，需要注意及时清除看门狗计数器，否则将产生图 14-1 所示的非期望的看门狗复位：虽然看门狗的溢出时间可配置，但是大部分看门狗溢出时间为 ms 量级，小于副系统时钟的稳定时间（约 2s）；如果进行了等待超时控制，那么超时控制的时间也是 $\geq 2s$ ；在正常等待副系统时钟稳定的过程中，ms 量级的看门狗溢出将产生非期望的看门狗复位。

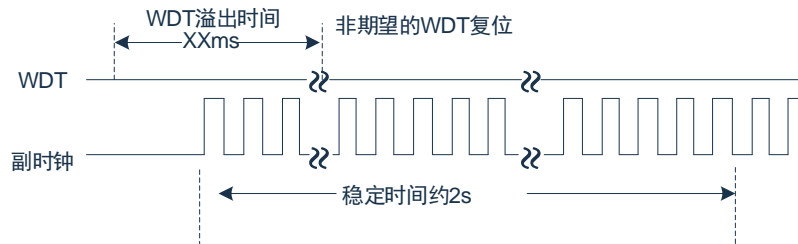


图 14-1：等待副系统时钟稳定过程中的 WDT 复位

14.1 推荐操作

如图 14-2 b 所示为推荐的操作流程，在等待副系统时钟稳定的过程中及时清除看门狗计数器。

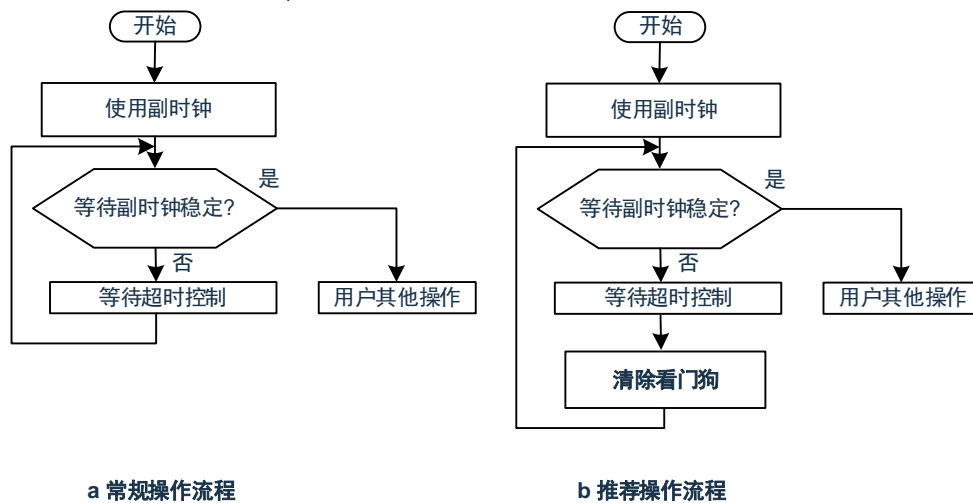


图 14-2：等待副系统时钟稳定过程中的 WDT 操作

15. 版本修订说明

版本号	时间	修改内容
V1.0.0	2024 年 1 月	初始版本