



复旦微电子

FM33FT0xxA 车规级系列 MCU 软件开发注意事项

应用笔记

V1.0.3



本资料是为了让用户根据用途选择合适的上海复旦微电子集团股份有限公司（以下简称复旦微电子）的产品而提供的参考资料，不转让属于复旦微电子或者第三者所有的知识产权以及其他权利的许可。

在使用本资料所记载的信息最终做出有关信息和产品是否适用的判断前，请您务必将所有信息作为一个整体系统来进行评价。

采购方对于选择与使用本文描述的复旦微电子的产品和服务全权负责，复旦微电子不承担采购方选择与使用本文描述的产品和服务的责任。除非以书面形式明确地认可，复旦微电子的产品不推荐、不授权、不担保用于包括军事、航空、航天、救生及生命维持系统在内的，由于失效或故障可能导致人身伤亡、严重的财产或环境损失的产品或系统中。

未经复旦微电子的许可，不得翻印或者复制全部或部分本资料的内容。

今后日常的产品更新会在适当的时候发布，恕不另行通知。在购买本资料所记载的产品时，请预先向复旦微电子在当地的销售办事处确认最新信息，并请您通过各种方式关注复旦微电子公布的信息，包括复旦微电子的网站(<http://www.fmsh.com/>)。

如果您需要了解有关本资料所记载的信息或产品的详情，请与上海复旦微电子集团股份有限公司在当地的销售办事处联系。

商标

上海复旦微电子集团股份有限公司的公司名称、徽标以及“复旦”徽标均为上海复旦微电子集团股份有限公司及其分公司在中国的商标或注册商标。

上海复旦微电子集团股份有限公司在中国发布，版权所有。



目 录

1 说明	3
2 驱动库	3
3 上下电处理注意事项	3
4 独立看门狗 (IWDG)	3
5 电源电压监测 (SVD)	4
6 复位管理单元 (RMU)	4
6.1 BOR 下电复位	4
6.2 外设复位	5
7 总线与存储	5
7.1 Flash 读取等待	5
7.2 Flash 擦写/编程	5
7.2.1 擦写/编程次数	6
7.2.2 软件流程	6
7.2.3 复位	6
7.2.4 程序延迟	6
7.2.5 工作时钟源	7
7.2.6 Data Flash 应用注意事项	8
8 时钟管理单元 (CMU)	8
8.1 时钟源启动稳定时间	8
8.2 LSCLK 配置	9
8.3 RCHF 调校	9
9 电源管理单元 (PMU)	10
9.1 Deep Sleep 配置	10
9.2 模拟 BUFFER 控制寄存器(PMU_BUFGR)	11
9.3 VREFINT 输出调校	12
10 中断源多路选择 (INTMUX)	12
11 经典控制器区域网络 (FSCAN)	13
12 通用异步收发传输器 (UART)	14
13 模数转换器 (ADC)	14



13.1 基准源	14
13.2 ADC 校准	14
13.3 BUF4TST 配置寄存器(BUF4TST_CFGR)	14
13.4 单次转换 (CONT=0) 自动模式 (SEMI=0)	14
14 I/O 端口 (GPIO)	15
14.1 外部引脚中断(EXTI)	15
14.2 NRST 引脚	17
14.3 SWD 引脚	17
14.4 GPIOC 组数字功能	17
14.5 GPIO 驱动强度	17
15 直接存储访问控制器 (DMA)	18
16 电容触摸按键 (TSI)	18
17 双线串行总线和系统管理总线 (I2C-SMBus)	18
17.1 I2C_SMBus TOB 标志	18
17.2 I2S 发送暂停	18
18 调试支持	18
19 地址对齐访问	19
20 HardFault_Handler	19
21 SPI 模块	19
版本信息	20
上海复旦微电子集团股份有限公司销售及服务中心	21

1 说明

本文档为 FM33FT0xxA 系列车规级 MCU 的应用笔记,用于描述客户在使用该系列 MCU 进行开发产品时,应在嵌入软件方面注意的事项。FM33FT0xxA 系列是复旦微电子公司开发的车规级 MCU 芯片,请联系复旦微电子公司提供更多相关文档支持设计开发。

2 驱动库

用户在产品开发前请到复旦微开发者论坛更新最新版 FM33FT0xxA 驱动库,里面包含《CMSIS》和《FM33FT0xxA_FL_Driver》两个库文件。

复旦微开发者论坛地址: <http://www.fmdevelopers.com.cn>

3 上下电处理注意事项

用户在产品开发前请务必仔细阅读《FM33 系列 MCU 上下电处理注意事项》应用笔记,它能有效提高软件在上下电过程中的工作可靠性。

4 独立看门狗 (IWDG)

独立看门狗定时器 (IWDG) 的用途是当程序的个别功能以错误的顺序执行,或在过长的时间内执行时,及时检测出有缺陷的程序顺序。IWDG 被启用后,它需要定期和及时地执行清狗服务程序,清狗程序必须在配置的时间窗口内(如配置),或在看门狗定时器溢出超时前执行,当溢出超时发生时, IWDG 立即产生全局复位信号,重启系统,以避免系统锁死。

FM33FT0xxA 的独立看门狗在 MCU 上电后默认启动,且无法由软件关闭。用户软件应在进入 main() 后的第一步就配置目标溢出周期,从而确保对后面执行的软件起到监控目的。用户软件有责任按照应用程序的要求,以所需的溢出/超时周期插入程序流执行清狗服务程序,但在定时中断服务中清狗是要避免的。

如用户的程序流对时间精度有高严格要求,则建议启用独立看门狗的窗口功能,可以达到更高效的时间流监控。



如用户使用 MCU 休眠功能，则推荐用户定时唤醒 MCU 执行清狗服务程序，定时唤醒源可以使用定时器、RTC 等，对应外设的功耗也极低。当然 MCU 也提供在休眠模式下冻结 IWDG 计数功能，用户可通过对 FREEZE 寄存器位写 1 使能此功能，使能后 MCU 在休眠模式下 IWDG 停止计数。

5 电源电压监测（SVD）

用户软件在完成 IWDG 初始化后，应立即启用 SVD 以监控 MCU 电源是否上升到目标电压阈值（推荐 4.2V），在未上升到目标电压阈值前，不建议继续初始化 MCU 的其他外设或执行应用功能，以避免在低电压状态下运行所带来的风险。

MCU 上电正常运行后用户软件同样应使用 SVD 对 MCU 的电源电压进行掉电监测，当低于目标阈值时（推荐 4.2V），用户软件应有保护性设计，以防止应用程序在低电压状态下运行。

详细流程及软件实现请参考《FM33 系列 MCU 上下电处理注意事项》应用笔记和 FM33FT0xxA 标准例程。

6 复位管理单元（RMU）

6.1 BOR 下电复位

上电过程中用户软件使用 SVD 监测到 MCU 电源电压大于目标阈值后，应及时配置复位管理单元的 BOR 下电复位电压阈值（推荐 falling 2.0v reset）并保持使能，即使进入 MCU 休眠模式也禁止关闭。

先监测电源电压再配置 BOR 的目的是防止 MCU 在上电过程中振荡复位。

详细流程及软件实现请参考《FM33 系列 MCU 上下电处理注意事项》应用笔记和 FM33FT0xxA 标准例程。

6.2 外设复位

RMU 单元支持通过寄存器配置对单独某一外设执行复位，使能复位 bit 后应立即撤销复位（先写 1 再写 0），中间无需延时，必须确保退出后是撤销复位的。

某外设复位控制 bit 示意图：

17	CAN1RST	CAN1 模块复位，软件写 1 复位，写 0 撤销复位 0：不复位 1：复位
----	---------	--

7 总线与存储

7.1 Flash 读取等待

MCU 复位启动后系统时钟源默认为 RCHF 8MHz，用户软件在配置系统时钟源之前必须确保 Flash 读取控制寄存器 (FLS_RDCR) 满足如下要求：

➤ Code Flash 读等待周期配置 (CODE_WAIT)：

当目标 CPU 主频小于等于 24MHz 时，不需要开启 wait；

当目标 CPU 主频大于 24MHz 小于等于 48MHz 时，提前使能 lwait；

禁止目标 CPU 主频大于 48MHz；

➤ Data Flash 读等待周期 (DATA_WAIT)：

基于目标 CPU 主频，提前配置 DATA_WAIT 等待周期，确保 Data Flash 工作时钟小于等于 12.5MHz；

7.2 Flash 擦写/编程

如用户使用到 Flash 擦写/编程功能，用户在产品开发前请务必仔细阅读

《FM33FT0xxA Flash 擦写编程应用注意事项》应用笔记，它能有效帮助用户正确应用此功能。



7.2.1 擦写/编程次数

如用户使用到 Flash 擦写/编程功能，必须确保擦写/编程总次数小于 FM33FT0xxA 产品说明书规定的次数。

7.2.2 软件流程

软件实现流程请严格参考复旦微 FM33FT0xxA 标准例程关于 Flash 的擦写示例，为提高 Flash 数据安全，如下要求是要遵守的：

- 每次擦写/编程完毕都要及时 Flash Key 锁定；
- 每次擦写/编程完毕都要及时关闭 Flash 的操作时钟源；
- 每次擦写/编程完毕都要及时关闭 Flash 寄存器的访问时钟源；

7.2.3 复位

如在对 Flash 执行擦写/编程时刻发生全局复位，则会对 Flash 的机器码或数据产生极其严重且不可预知的后果（如误改变 Flash 机器码数据），因此如用户软件使用到 Flash 擦写/编程功能，用户有责任对软件执行如下排查，且是强制要求：

- 执行 Flash 擦写/编程时刻，不会存在看门狗复位的可能性；
- 执行 Flash 擦写/编程时刻，不会存在 NRST 被拉低复位的可能性；
- 执行 Flash 擦写/编程时刻，如系统电源掉电，软硬件方面有机制确保不会与 BOR/PDR 下电复位发生冲突；

7.2.4 程序延迟

CPU 通过 AHB 总线对 Flash 访问，当 CPU 对 Flash 执行擦写/编程操作时，Flash 外设数据总线被占用，读取即取指操作将会暂停，直至硬件的擦写/编程完成。

关于线程任务：

- 采用非 DMA 模式时，对用户程序无影响，因为软件流程设计为流水线模式，前一个功能完成后再执行下一个功能。



- 采用 DMA 模式连续编程时，CPU 和 DMA 共属主端，当访问同一个外设时，由 BusMatrix 仲裁优先级，如 DMA 抢占成功且正在执行 Flash 编程时，由于 Flash 外设数据总线被占用，CPU 会暂时停止取指，因此用户线程任务的执行会有延迟。

关于中断服务：

- 如向量表定义在 Flash 区，软件或 DMA (仅编程) 对 Flash 执行擦写/编程时，由于 Flash 外设数据总线的占用，中断服务函数响应会有延迟。

综上所述，用户对于处理时间比较关键的程序 (如高频率高精度的定时中断响应等)，应了解该延迟的存在，并做出对应的处理措施。

可行措施与应用场景强相关，且方法较多，如擦写/编程与关键任务的时序调整等。本文档在此处描述一种可通用的措施之一：对处理时间比较关键的程序或中断，可将执行代码或中断向量表定义到 RAM 中，此时 Flash 的擦写/编程操作不会影响 CPU 到 RAM 的取指。

执行 Flash 擦写/编程时，Flash 外设会根据当前 RCHF 实际频率值来产生相应的计时长，即不同频率值其需要的时长是一样的，这由 Flash 的物理介质决定，具体值如下：

扇擦时长：4ms ~ 5ms

页擦时长：1ms ~ 1.25ms

字写时长：6us ~ 7.5us

注：Flash 擦写/编程时钟使用 RCHF 时钟，但系统时钟可以是任意时钟。

7.2.5 工作时钟源

Flash 擦写/编程工作时钟源硬件固定为 RCHF，支持 8MHz、16MHz 和 24MHz，因此执行擦写/编程功能前，必须使能 RCHF 且配置目标频率应小于等于 24MHz。



如系统时钟同样选择 RCHF 且使用到 Flash 擦写/编程功能，则需要注意 RCHF 频率应小于等于 24MHz。

7.2.6 Data Flash 应用注意事项

➤ ECC

FM33FT0xxA 的 Code Flash 和 Data Flash 各自有独立的 ECC 使能；上电复位后 Data Flash ECC 默认使能，Code Flash ECC 默认关闭。

如 ECC 使能，用户软件或仿真器（如 Keil Memory 窗口）读取 Flash 擦写后但未编程过的数据时，会产生 Flash ECC 检错标志，从而进入 NMI_Handler 异常服务函数。

因上电复位后 Data Flash ECC 默认使能，因此用户软件需注意处理，如不使用 ECC 功能则可以先禁止 Data Flash ECC 再读取 Data Flash 数据，如需使用 ECC 功能则需确保软件或仿真器（如 Keil Memory 窗口）不读取未编程过的 Data Flash 数据。

➤ 存储器访问保护

如果使能了存储器访问保护 (SFU_MAPCR.MAP_EN=1)，则禁止对 Data Flash 进行访问，此时访问 Data Flash 将触发非法地址访问事件，当 MAP_SEL=0 时产生 NMI 中断，当 MAP_SEL=1 时将复位芯片。

因此当使能存储器访问保护后，如需访问 Data Flash，则应先禁止 MAP_EN。

8 时钟管理单元 (CMU)

必须在使能 IWDG，MCU 电源电压大于 SVD 监测阈值和使能 BOR 之后才允许配置 CMU 单元。

8.1 时钟源启动稳定时间

- 如使用 XTALF，必须参考元器件手册技术参数配置启动等待时间 (CMU_XTLFCR: WAIT) 和工作电流 (CMU_XTLFCR: IPW)，同时用户软件必须判断 XTALF 稳定标志 (CMU_XTLFCR: RDY) 置起后才能继续执行。



- 如使用 XTHF，必须参考元器件手册技术参数配置启动等待时间 (CMU_XTHCFR: WAIT)，同时用户软件必须判断 XTHF 稳定标志 (CMU_XTHCFR: RDY) 置起后才能继续执行。
- 如使用 PLL，使能之前必须判断 PLL 使用的时钟源已稳定工作，使能 PLL 后用户软件必须判断 PLL 锁定标志 (CMU_PLLCR: LOCKED) 置起后才能继续执行。
- 在选择系统时钟源 (CMU_SYCLKCR: SYSCLESEL) 之前，用户软件必须判读时钟源已稳定工作。

8.2 LSCLK 配置

MCU 启动后 SystemInit() 函数会禁止 LSCLK 自动切换功能 (CMU_SYCLKCR: LSCATS)，并选择 RCLP 作为 LSCLK 的时钟源 (CMU_LSCLKSEL: LSCATS)。

如用户有使用外部 XTLP，建议选择 XTLP 作为 LSCLK 的时钟源，并使能 LSCLK 自动切换功能，相关软件实施流程可参考复旦微 FM33FT0xxA 标准例程 RTC 模块中关于 XTLP 的配置部分。

使能自动切换功能后，如 XTLP 停振硬件将自动选择 RCLP 作为时钟源，提高了时钟安全性。LSCLK 主要作为 SVD、LCD、RTC 以及 Sleep 模式下外部引脚中断采样的时钟源。

8.3 RCHF 调校

RCHF 是 MCU 内部的高频 RC 振荡器，典型振荡频率为 8/16/24/32MHz，为确保全温度范围的频率精度，每个频率类型都需要调用厂内标定的调校值写到 RCHF 调校寄存器 (CMU_RCHFTR)，具体如下：

使用 RCHF 8MHz 时： CMU->RCHFTR = RCHF8M_TRIM;

使用 RCHF 16MHz 时： CMU->RCHFTR = RCHF16M_TRIM;

使用 RCHF 24MHz 时： CMU->RCHFTR = RCHF24M_TRIM;

使用 RCHF 32MHz 时： CMU->RCHFTR = RCHF32M_TRIM;

MCU 复位启动后 RCHF 缺省为 8MHz，SystemInit()函数会执行 CMU->RCHFTR = RCHF8M_TRIM 语句，当用户切换到其他频率值时，需用户软件将正确的调校值写到 RCHF 调校寄存器。

当用户软件使用到 MCU 休眠功能时，MCU 唤醒后时钟源默认使用 RCHF，但频率值由 PMU_CRWKFSEL 寄存器决定，因此如 Active 运行的 RCHF 频率与唤醒后的 RCHF 频率不一致，则需用户软件及时切换频率，并在频率切换后将正确的调校值写到 RCHF 调校寄存器

11:10	WKFSEL	Sleep/DeepSleep 唤醒后的系统频率 00: RCHF-8MHz 01: RCHF-16MHz 10: RCHF-24MHz 11: RFU
-------	--------	--

9 电源管理单元 (PMU)

9.1 Deep Sleep 配置

如用户使用到深度休眠，不要通过 CPU 系统控制寄存器 (SCB) 的 SLEEPDEEP 位实现，而应通过低功耗控制器 (PMU_CR) 的 SLPDP 位实现。

MCU 休眠配置流程如下：

- 配置 PMU_CR 寄存器的 WKFSEL 位：选择 Sleep/DeepSleep 唤醒后的系统频率；
- 配置 PMU_CR 寄存器的 DeepSleep 位：选择是 Normal Sleep 或 Deep Sleep；
- 配置 PMU_CR 寄存器的 PMOD 位：选择 Sleep mode/DeepSleep mode；
- 执行 __WFI() 或 __WFE() 指令，等待中断或事件唤醒。

低功耗工作寄存器 (PMU_CR)：

Bit	助记符	功能描述
31	RFUI	Dummy 寄存器
30:26	--	未实现，读为 0
25:24	AVREF_SH	AVREF 采样保持周期配置 (32K 时钟周期) 00: 2ms



		01: 4ms 10: 8ms 11: 16ms
23:12	--	未实现, 读为 0
11:10	WKFSEL	Sleep/DeepSleep 唤醒后的系统频率 00: RCHF-8MHz 01: RCHF-16MHz 10: RCHF-24MHz 11: RFU
9	SLPDP	DeepSleep 控制寄存器 1: DeepSleep 模式使能, 下关闭基准电压源 0: 常规 Sleep 模式 在 Sleep 下, 如果置位了 SLPDP 位即为 DeepSleep 模式; 该位仅在 Sleep 下有效
8	--	未实现, 读为 0
7:4	--	未实现, 读为 0
3:2	RFUI	Dummy 寄存器
1:0	PMOD	低功耗模式配置寄存器 00: Active mode 01: LPRUN mode 10: Sleep mode / DeepSleep mode 11: RFU

9.2 模拟 BUFFER 控制寄存器(PMU_BUFCCR)

如用户软件使用到 ADC 模块且选择 VDDA 为参考源时, 为保证采样精度一般会使用 AVREF 输出缓冲 (AVREFBUF_EN), 同样如应用软件使用到 ADC 模块的温度采样时, 会使用 Vpata Buffer (VPTATBUFFER_EN)。使能此两个 BUFFER 输出会有 7uA 左右的功耗, 因此如用户软件有 MCU 休眠功能, 在休眠之前应禁止此两个 BUFFER 输出。

模拟 BUFFER 控制寄存器 (PMU_BUFCCR)

位号	助记符	功能描述
31:6	-	RFU: 未实现, 读为 0
5	AVREFBUF_OUTE N	AVREF 输出缓冲输出使能, ADC 采样 AVREF 时建议使能
4	AVREFBUF_EN	AVREF 输出缓冲使能, ADC 采样 AVREF 时建议使能
3	VPTATBUFFER_O UTEN	Vptat Buffer 模块开关通道输出使能信号, 高电平使能有效。 (PTAT Buffer Output Enable)
2	VPTATBUFFER_E N	Vptat Buffer 模块使能信号, 高电平使能有效。(PTAT Buffer Enable)
1	-	RFU: 未实现, 读为 0
0	-	RFU: 未实现, 读为 0

9.3 VREFINT 输出调校

FM33FT0xxA 片内电路 VREFINT_VREG 可以产生 1.5V\2.0V\2.5V\3.0V 档位的基准电压，可以给内部电路使用，如 ADC 基准源、比较器输入等。

如用户软件使能该基准电压，需根据配置的输出电压档位调用厂内标定调校值写到 VEEFINT 调校寄存器（PMU_VREFINT_TR），确保输出电压精度，具体如下：

VREFINT 输出 1.5V: VREFP → TR = FL_VREFP_OUTPUT_VOLTAGE_1P5V_TRIM;

VREFINT 输出 2.0V: VREFP → TR = FL_VREFP_OUTPUT_VOLTAGE_2P0V_TRIM ;

VREFINT 输出 2.5V: VREFP → TR = FL_VREFP_OUTPUT_VOLTAGE_2P5V_TRIM;

VREFINT 输出 3.0V: VREFP → TR = FL_VREFP_OUTPUT_VOLTAGE_3P0V_TRIM;

10 中断源多路选择（INTMUX）

中断向量表中偏移第 23~第 30、第 34、第 39、第 42、第 43 和第 45 向量入口地址的中断源是需要用户软件配置中断源多路选择控制器 (INTMUX_CR) 进行选择的，具体映射关系如下：

中断向量多路映射关系：

	SEL=00	SEL=01	SEL=10	SEL=11
MUX0	FLASH	TAU00	TAU01	TAU02
MUX1	SPI0	TAU10	TAU11	TAU06
MUX2	SPI1	TAU12	TAU01	TAU02
MUX3	UART0	TAU10	TAU11	TAU01
MUX4	UART1	TAU00	TAU01	TAU10
MUX5	UART2	TAU03	TAU04	TAU11
MUX6	UART3	TAU12	TAU13	TAU02
MUX7	UART4	TAU12	TAU13	TAU04
MUX8	UART5	TAU05	TAU06	TAU12
MUX9	CCL	TAU05	TAU06	TAU03
MUX10	LUT	TAU03	TAU04	TAU07
MUX11	SPI2	TAU11	TAU04	TAU03
MUX12	SPI3	TAU13	TAU05	TAU07
MUX13	TAU00	UART1	UART2	CAN1

注 1：配置之前需通过 CMU_PLCKCR2 寄存器使能 INTMUX_PCE 总线时钟。

注 2：中断向量表请参考 FM33FT0xxA 产品说明书的中断源多路选择章节。

11 经典控制器区域网络（FSCAN）

用户在产品开发前请务必仔细阅读《FM33FT0xxA FSCAN 应用说明》应用笔记，它能确保 FSCAN 被正确配置及可靠运行。应用笔记内容其中有点非常重要，在此文档中再次重申：

- 使能自动重发功能时，禁止使用发送撤销请求功能，否则发送单元会向总线发送错误数据。
- 用户使用 FSCAN 休眠功能并借此唤醒 MCU，则需关闭 FSCAN 的工作时钟源（CMU_OPCCR3 CANxCKE）以锁定 CAN 的 Sleep Mode，但禁止关闭 FSCAN 的总线时钟（CMU_PLCKCR3 CANxPCE），以确保用户软件对 FSCAN 外设的正常访问。
- FSCAN 通过关闭工作时钟源来锁定 Sleep Mode，关闭时钟源后软件必须判断 CAN 是否处于 Sleep Mode 即是否成功锁定 Sleep Mode，如未成功锁定 Sleep Mode 则无法唤醒 MCU。
- 当用户使能自动重发 (AREN) 功能时，用户软件必须使能 Bus Off 恢复中断服务，并在服务函数中对 CAN 控制器重新初始化，否则存在概率向总线发送随机的错误帧，示例如下：

```
/* CAN0 中断服务函数 */
void CAN0_IRQHandler(void)
{
    uint32_t RXNotEmptyIT, RXNotEmptyFLAG;

    /* 必须保留 */
    if ((0x01U == FL_CAN_IsEnabledIT_BusOffRecovery(CAN0)) &&
        (0x01U == FL_CAN_IsActiveFlag_BusOffRecovery(CAN0)))
    {
        FL_CAN_ClearFlag_BusOffRecovery(CAN0);
        /* 使能自动重发功能，BusOff恢复时刻需重新初始化Controller */
        if (0x01U == FL_CAN_IsEnabledAutoResend(CAN0))
        {
            FL_CAN_Disable(CAN0);
            Can_Controller_Reg_Init(CAN0);
        }
    }
}
```



12 通用异步收发传输器 (UART)

- 优先推荐使用 UART0 和 UART1，其包含 UART 休眠唤醒和 LIN 功能。
- 如使用软件模拟 LIN 功能且采用定时器输入捕捉功能进行间隔段和同步段的时长捕捉时，推荐 UART 外设工作时钟源和定时器工作时钟源保持一致。
- 如使用 LIN 硬件外设，对软件处理有特定要求，请联系上海复旦微电子有限公司获取更多技术支持。

13 模数转换器 (ADC)

13.1 基准源

推荐正基准源选择 VDDA，负基准源选择 VSS，软件实现流程请参考复旦微 FM33FT0xxA 标准例程关于 ADC 应用示例。

13.2 ADC 校准

MCU 每次复位后必须执行一次 ADC 校准操作，以获得更好的采样精度，软件实现流程请参考复旦微 FM33FT0xxA 标准例程关于 ADC 应用示例。

13.3 BUF4TST 配置寄存器(BUF4TST_CFGR)

针对慢速通道 ADCIN14~ADCIN22，如信号驱动能力较弱，可使能 BUFFER 提供驱动能力从而提高采样精度，软件实现流程请参考复旦微 FM33FT0XXA 标准例程《ADC 内部慢速通道查询》。

需要注意的是如使能 BUFFER 功能，ADCIN14~ADCIN22 通道每次只能使用其中一个通道，因此如用户软件选择多个通道序列转换且使能 BUFFER 功能时，ADCIN14~ADCIN22 通道只能选择一个通道。如未使能 BUFFER 功能，则多个通道序列转换不受此限制。

13.4 单次转换 (CONT=0) 自动模式 (SEMI=0)

当用户使用该模式时，必须遵循以下两个要求：

- a) 使用 DMA 搬运数据功能时：



- 禁止 ADC 等待模式 (WAIT=0) ;
- 配置 DMA 的 ADC 通道优先级为 Very High;

b) 未使用 DMA 搬运数据功能, 即采用查询或中断方式时:

当所有通道即 1 个序列转换完成后, 必须先关闭 ADC, 再执行清除 EOC 标志, 代码示意如下:

```
/* 关闭 ADC */  
FL_ADC_Disable(ADC);  
/* 清除 EOC 标志 */  
FL_ADC_ClearFlag_EndOfConversion(ADC);
```

14 I/O 端口 (GPIO)

14.1 外部引脚中断(EXTI)

输入引脚选择:

FM33FT0xxA 的 4 组 GPIO (A~D) 最多可以产生 16 个 EXTI 中断, 每组 GPIO 分别可以产生 4 个 EXTI 中断标志, 最终所有的 EXTI 中断汇总到 NVIC 的#46 入口。

中断标志和引脚对应关系如下表:



GPIO	EXTI输入选择	EXTI
PA0~PA3	EXTI_ASEL[1:0]	EXTI[0]
PA4~PA7	EXTI_ASEL[3:2]	EXTI[1]
PA8~PA11	EXTI_ASEL[5:4]	EXTI[2]
PA12~PA15	EXTI_ASEL[7:6]	EXTI[3]
PB0~PB3	EXTI_BSEL[1:0]	EXTI[4]
PB4~PB7	EXTI_BSEL[3:2]	EXTI[5]
PB8~PB11	EXTI_BSEL[5:4]	EXTI[6]
PB12~PB15	EXTI_BSEL[7:6]	EXTI[7]
PC0~PC3	EXTI_CSEL[1:0]	EXTI[8]
PC4~PC7	EXTI_CSEL[3:2]	EXTI[9]
PC8~PC11	EXTI_CSEL[5:4]	EXTI[10]
PC12~PC13	EXTI_CSEL[7:6]	EXTI[11]
PD0~PD3	EXTI_DSEL[1:0]	EXTI[12]
PD4~PD7	EXTI_DSEL[3:2]	EXTI[13]
PD8~PD11	EXTI_DSEL[5:4]	EXTI[14]
PD12~PD15	EXTI_DSEL[7:6]	EXTI[15]
PE0~PE3	EXTI_ESEL[1:0]	EXTI[16]
PE4~PE7	EXTI_ESEL[3:2]	EXTI[17]
PE8~PE11	EXTI_ESEL[5:4]	EXTI[18]
PE12~PE15	EXTI_ESEL[7:6]	EXTI[19]
PF0~PF3	EXTI_FSEL[1:0]	EXTI[20]
PF4~PF7	EXTI_FSEL[3:2]	EXTI[21]

注：对应寄存器为 EXTI 输入选择寄存器 0（GPIO_EXTISEL0）和 EXTI 输入选择寄存器 1（GPIO_EXTISEL1）。

时钟源选择：

EXTI 除正确选择输入引脚外，还应正确配置工作时钟源，涉及的时钟源如下：

工作模式：

- 外设工作时钟配置寄存器 1（CMU_OPCCR1）的 EXTI 中断采样时钟源（EXTICKS）

选择： 1：外部引脚中断使用 LSCLK 采样

0：外部引脚中断使用 AHBCLK

- 外设工作时钟配置寄存器 3（CMU_OPCCR3）的 EXTI 中断采样时钟（EXTICKE）

使能： 1：使能

0：禁止

Sleep/DeepSleep 模式:

- 系统时钟配置寄存器 (CMU_SYSCLOCKER) 的 Sleep/DeepSleep 模式下 EXTI 采样时钟设置 (SLPENEXTI) :
 - 1: Sleep/DeepSleep 模式下使能外部引脚中断采样 (采样时钟为 LSCLK)
 - 0: Sleep/DeepSleep 模式下禁止外部引脚中断采样 (将无法产生 EXTI 中断)

14.2 NRST 引脚

NRST 引脚不推荐功能复用, MCU 复位启动后该引脚缺省为 NRST 数字功能、使能上拉 (5k)。

如强制要复用, 请参考 FM33FT0xxA 产品说明书关于 NRST 引脚作为 GPIO 功能的详细要求。

14.3 SWD 引脚

SWD 引脚不推荐功能复用, MCU 复位启动后该引脚缺省为 SWD 数字功能\使能上拉 (50k)。

如一定要使用复用功能, 则推荐在 MCU 复位启动后等待 100ms 再配置复用口, 以给 SWD 口连接留有窗口期。

14.4 GPIOC 组数字功能

GPIOC 组引脚支持数字功能较多, 因此数字功能的选择除了配置 GPIO 额外数字功能选择寄存器 (GPIOx_DFS) 外, 还需配置 GPIOC 定时器通道选择寄存器 (GPIOC_TMUXR)。

14.5 GPIO 驱动强度

GPIO 驱动强度配置寄存器(GPIOx_DSR)推荐配置为 0b10 (驱动次强×3), 4 个等级的驱动能力如下:

X1 – 4mA; X2 – 8mA ; X3 – 12mA; X4 – 16mA;



15 直接存储访问控制器（DMA）

如果错误的使用 DMA 从非法地址向外设搬运数据，将导致 DMA 锁死无法继续工作，此时只能通过复位 DMA 模块恢复。

因此用户软件必须确保避免使用 DMA 访问非法地址。

16 电容触摸按键（TSI）

我司有一套完整的关于 TSI 应用文档资料，请联系上海复旦微电子公司获取更多技术支持。

17 双线串行总线和系统管理总线（I2C-SMBus）

17.1 I2C_SMBus TOB 标志

当 TOB 错误标志置位后，只有主机发送 STOP 和复位 SMBUS 模块才能清除。关闭使能 EN，不能复位 TOB 计数器，也不能清零 BUSY 标志，无法使模块回到 IDLE。

因此当 TOB 发生错误后，需用户软件复位 I2C_SMBus 模块。

17.2 I2S 发送暂停

I2S 发送过程中，如果 TXBUFFER 空，则 I2S 停止发送流程，下次发送必须重新开始。因此当连续发送时用户软件应避免发送缓冲区空

18 调试支持

MCU 在 Sleep 模式下如用户要仿真调试，需用户软件配置 MCU DEBUG 配置寄存器的休眠模式下的 debug 配置位（DBG_SLEEP）：

0：正常进入休眠，关闭 HCLK 和 FCLK，debugger 无法保持连接

1：不关闭 HCLK 和 FCLK，debugger 能够在休眠模式下保持连接

注：正式发布程序建议配置为 0。



19 地址对齐访问

FM33FT0xxA 使用的 CPU 内核为 Cortex-M0+, 因此用户软件对 RAM、FLASH 和寄存器进行数据访问时, 必须遵循 ARM 要求的地址对齐访问规则, 如地址非对齐访问软件将进入 HardFault_Handler 硬件错误服务。

20 HardFault_Handler

用户软件应定义 void HardFault_Handler(void) 硬件错误服务函数, 并在服务函数内部增加软件复位命令, 以提高系统的可靠性。

21 SPI 模块

SPI 判断发送完成, TXBUF 标志不等于移位寄存器发送完成, BUSY 标志置位有个同步过程, 有 0 到 1 的一个过程, 写完 TXBUF, BUSY 没有立刻置 1 还是 0, 在 SPI 的工作时钟和系统时钟相差特别大时, 这个问题会特别明显。因此建议在开发过程中采用以下方式:

(1) 全双工模式: SPI 原理是主从移位寄存器交换, 可以判断 RXBUF 标志来判断接收完成。

(2) 半双工模式: 发送完成不能判断 RXBUF, 使用 BUSY, 先判断 BUSY 为 1, 再判断 BUSY 为 0, 发送完成。

```
FL_WriteTXBuff(SPI1,data); //写 TXBUF
```

```
while (SPI1->ISR &(1U<<8))==(1U<<8)); //等待 BUSY 置 1, 使用寄存器操作可以确保主时钟在 64MHZ 也可以保证程序逻辑没有问题
```

```
while(FL_SPI_IsActiveFlag_Busy(SPI1)); //BUSY 变 0, 发送完成
```



版本信息

版本号	发布日期	更改说明
1.0.0	2022.12	首次发布
1.0.1	2022.12	1、增加驱动库部分 2、增加 Data Flash 应用注意事项 3、增加经典控制器区域网络（FSCAN）内容
1.0.2	2022.3	1、修改 13.4 单次转换（CONT=0） 自动模式（SEMI=0） 2、SPI 模块
1.0.3	2023.12	1、增加 FSCAN Sleep Mode 的操作要求 2、优化退出 FSCAN BUS OFF 的操作要求



上海复旦微电子集团股份有限公司销售及服务中心

上海复旦微电子集团股份有限公司

地址：上海市国泰路 127 号 4 号楼

邮编：200433

电话：(86-021) 6565 5050

传真：(86-021) 6565 9115

上海复旦微电子（香港）股份有限公司

地址：香港九龙尖沙咀东嘉连威老道 98 号东海商业中心 5 楼 506 室

电话：(852) 2116 3288 2116 3338

传真：(852) 2116 0882

北京办事处

地址：北京市东城区东直门北小街青龙胡同 1 号歌华大厦 B 座 423 室

邮编：100007

电话：(86-10) 8418 6608

传真：(86-10) 8418 6211

深圳办事处

地址：深圳市华强北路 4002 号圣廷苑酒店世纪楼 1301 室

邮编：518028

电话：(86-0755) 8335 0911 8335 1011 8335 2011 8335 0611

传真：(86-0755) 8335 9011

台湾办事处

地址：台北市 114 内湖区内湖路一段 252 号 12 楼 1225 室

电话：(886-2) 7721 1889

传真：(886-2) 7722 3888



新加坡办事处

地址: 237, Alexandra Road, #07-01, The Alexcier, Singapore 159929

电话: (65) 6472 3688

传真: (65) 6472 3669

北美办事处

地址: 2490 W. Ray Road Suite#2 Chandler, AZ 85224 USA

电话: (480) 857-6500 ext 18

公司网址: <http://www.fmsh.com/>