

FM33A0XXEV 开发前需知

版本：V1.1

FM33A0XXEV 是一款 M0 内核的 ARM 芯片，在客户开发中通常会遇到一些普遍的问题，本文将分硬件、编译环境、软件驱动排查简单阐述这些问题，加快客户的开发过程。

1. 硬件

电源

FM33A0XXEV 系列芯片有电源 VDD 和 VDD15。VDD 为 MCU 的电源，VDD15 为 VDD 通过芯片内部 LDO 输出的 1.5V 电压用于数字电路的电源。

在硬件设计时 VDD 和 VDD15 不能直接连接，VDD15 只需要外接 0.1uF 电容就可以，详细见推荐原理图。

编程接口

FM33A0XXEV 系列芯片的编程接口使用的是 SWD 接口，芯片在处于低功耗状态时编程口不能正常工作，所以在通常的 SWD 接口的基础上增加了 RST 引脚。通过 RST 引脚去复位芯片，才能进行仿真器的操作。

假如客户需要使用 RTC，并对时间的精度要求比较高，还需要增加 FOUT 接口，使用 FOUT 接口可以进行外接晶体的温度调校。

复位电路

FM33A0XXEV 系列芯片，内部集成了上下电复位电路，不需要在 RST 引脚外部接复位电路。为防止干扰可以接滤波电容。

FM33A0XXEV 的 NRST 拉低芯片不会处于复位状态，程序仍然正常执行，只有保持低电平超过 8ms 后经过上升沿芯片才会复位。

外接低速晶体

XTLF 外接 32768K 晶体，FM33A0XXEV 在没有特殊配置下，外部晶体停振会自动切换到内部 32K 晶体 RCLP。假如不接外部晶体，为了防止外部干扰需要将 XTLFIN 接地。

在对 RTC 时间要求比较高时，需要使用一致性较好，精度比较高的外部晶体。匹配电容需要按照晶体手册进行设计。

GPIO

FM33A0XXEV 的 I2C 管脚不是真开漏管脚

PCB 布板

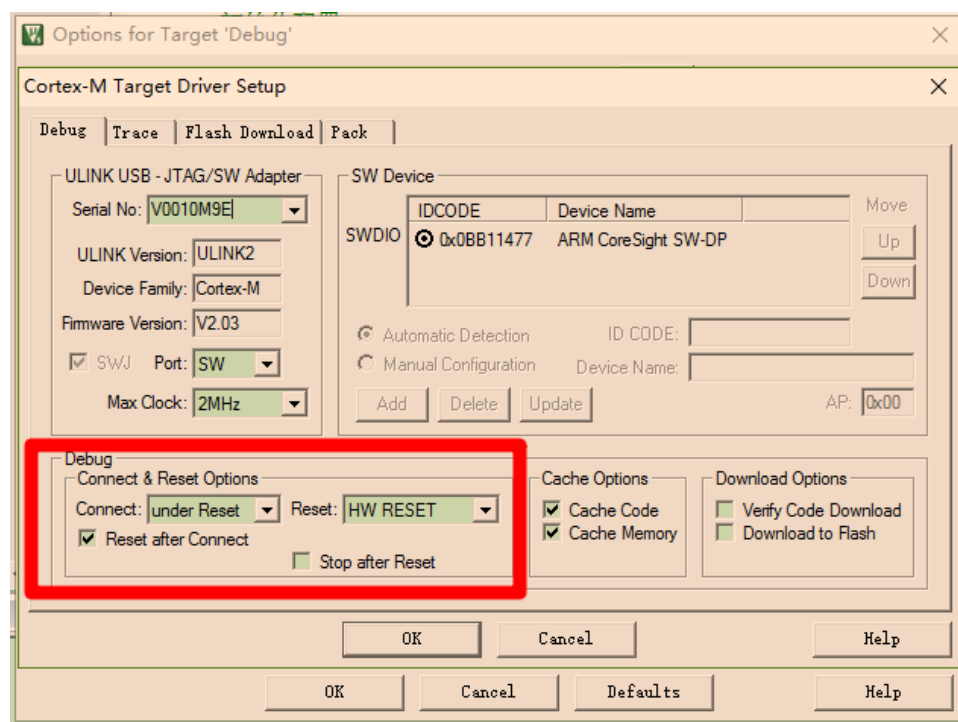
以 80pin 为例，为了保证更好的静电特性 pin61~pin80 这一侧放在最上方向

2. 编译器环境配置

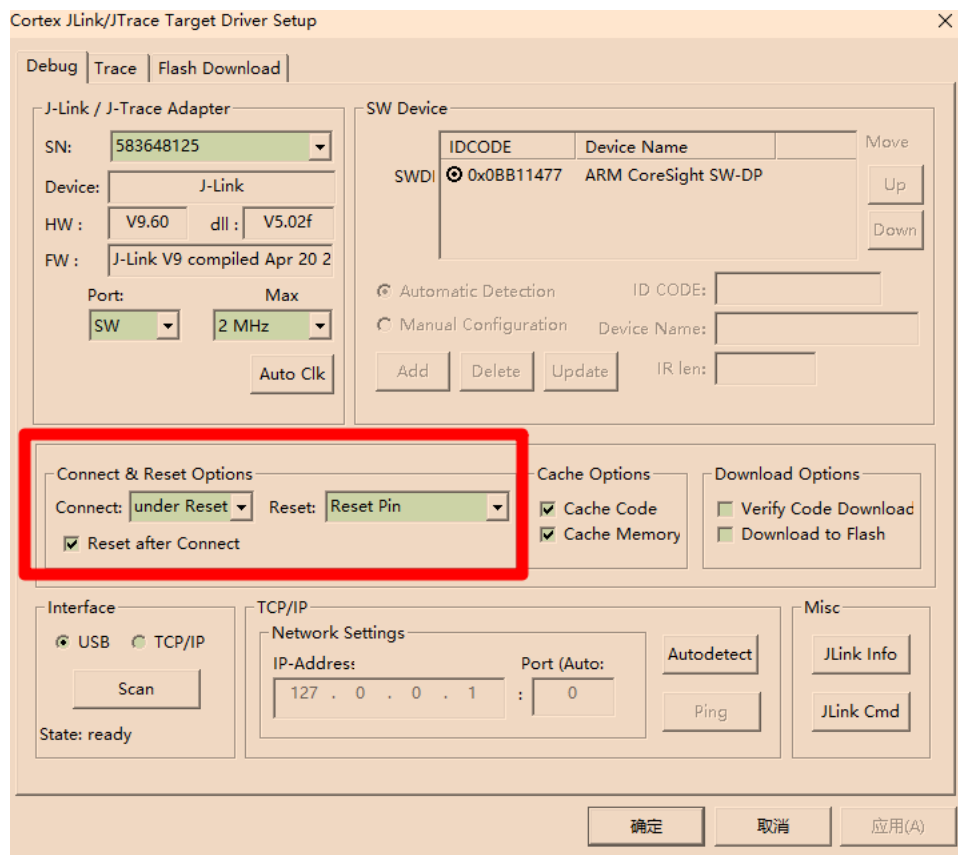
Keil 和 IAR 两种编译器 FM33A0XXEV 都可以支持。为了在休眠下能够顺利连接仿真器，除了在硬件上编程口需要连接 RST 脚以及在编译器的配置上需要如下设置。修改 SWD 状态之前或休眠时有足够的延时，保证仿真器能顺利连接芯片。

Keil

Keil 在使用 ULINK 时 debug 选项中 Connect 和 Reset, 建议按下图配置



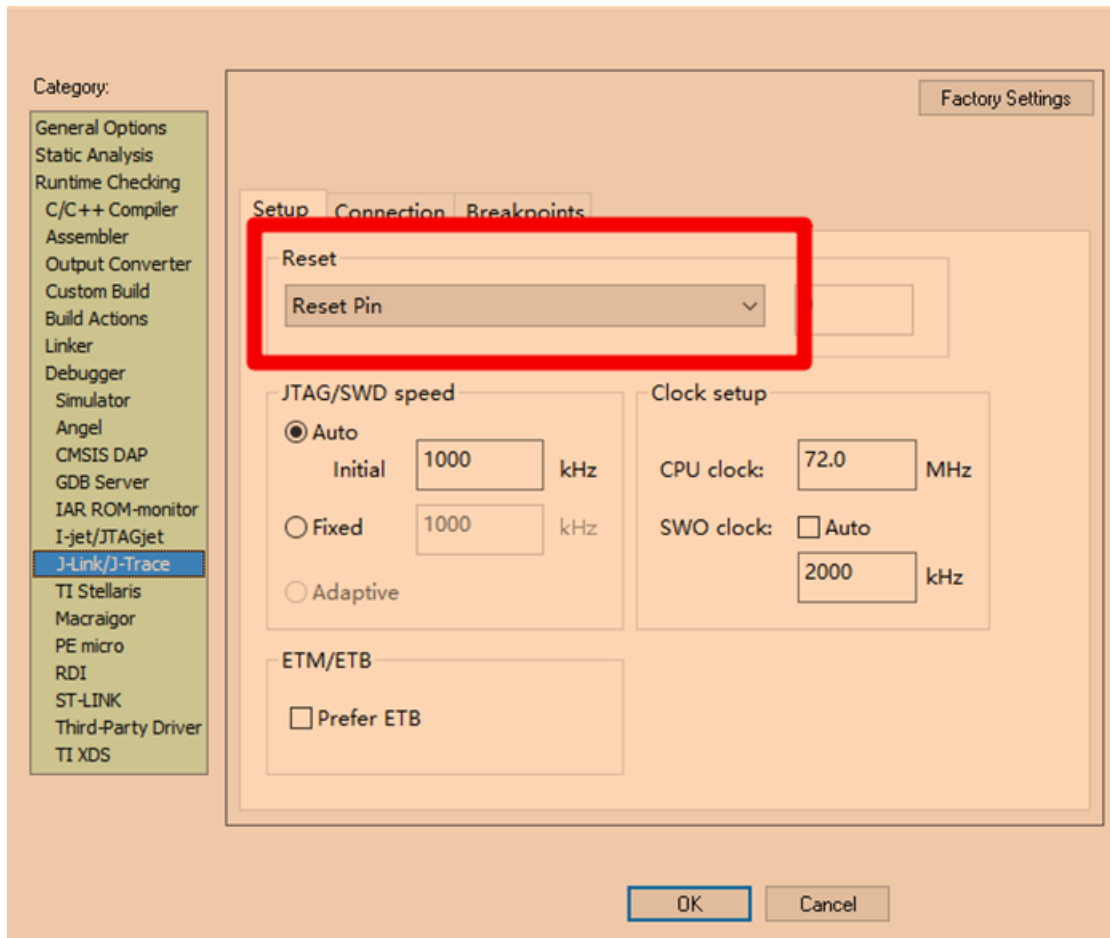
Keil 在使用 JLINK 时 debug 选项中 Connect 和 Reset, 建议按下图配置



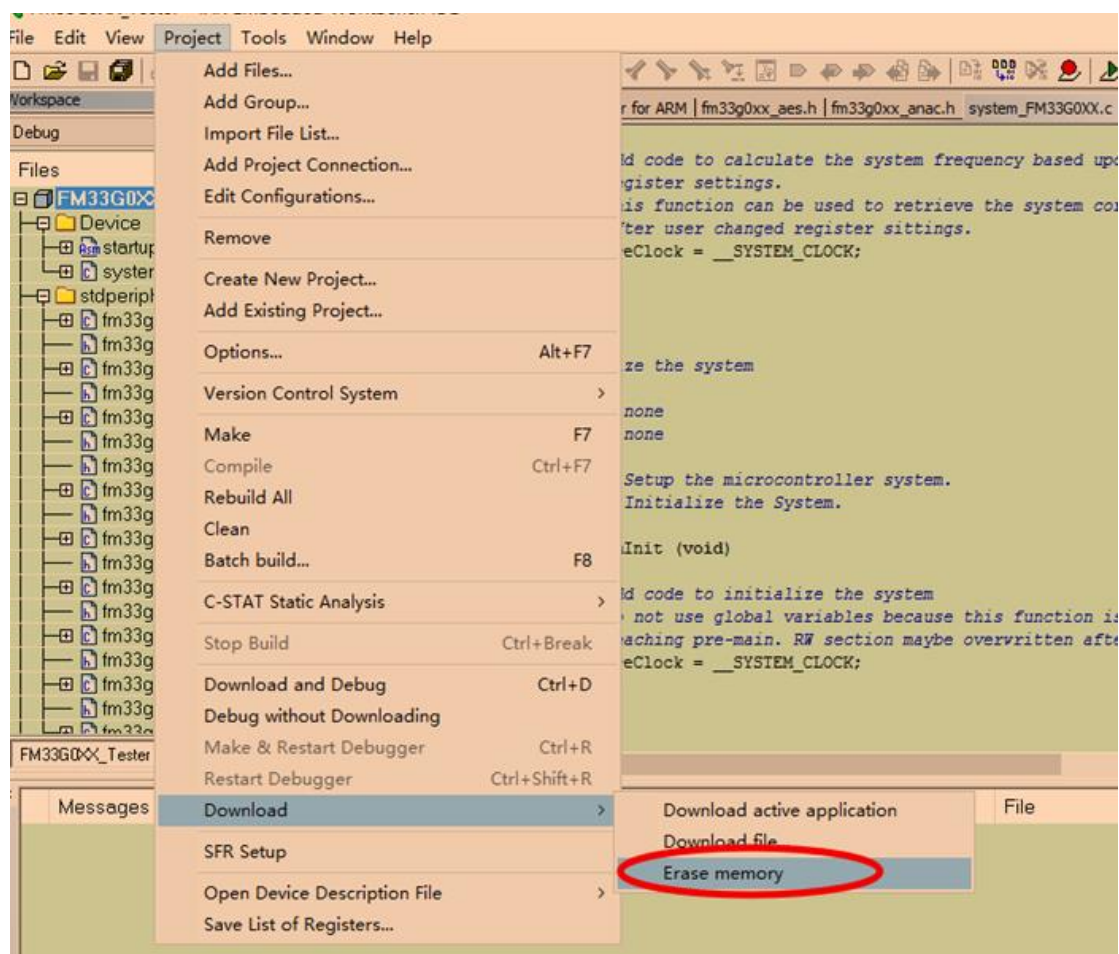
在 Flash Download 中最好确认下 Flash 的配置信息和芯片选型是否一致，假如不一致需要点击 Add 手动添加。

IAR

IAR Reset 选择 Reset pin 如下图



在实际使用中，假如这种配置不能成功烧写，需要先按照下图先擦写再编程



3. 软件

系统主时钟

FM33A0XXEV 系列可以选择多种时钟源做为主时钟，常用的为 RCHF、XTHF 和 PLL。RCHF 的频率为 8MHZ~48MHZ，其中 8MHZ 和 16MHZ 全温区的变化范围小于 $\pm 2\%$ 。XTHF 可外接 4-32MHZ 晶体。PLL 的时钟源必须是 RCHF 和 XTHF 分频后的 1MHZ，PLL 频率的范围 32MHZ-64MHZ。

RCHF 和 PLL (时钟源为 RCHF 或 XTHF) 做系统主时钟静电特性较好，XTHF 做系统主时钟的静电特性稍差。从功耗看 RCHF 的功耗较低。

RCHF 的每一档频率出厂前都有校准值，保证在常温时的精度，8MHZ 的 TRIM 为上电自动加载，其余频率需要程序手动加载。可以参看例程。

仿真定时器与看门狗

在仿真状态下定时器和看门狗默认为关闭状态，可以操作 DBGCR 来改变状态。

复位

FM33A0XXEV 有上电复位和下电复位电路，在程序中下电复位必须要打开。下电复位有 BOR 和 PDR，至少选择打开其中的一个。BOR 阈值比较准，但功耗高大约 2ua。PDR 阈值有 100~200MV 的偏差，但功耗低小于 100na。

UART

FM33A0XXEV 有 6 路独立 UART，每一路 UART 都具有红外调制输出功能。

UART 的中断标志位有 buffer 空产生和移位寄存器空产生两种，建议使用移位寄存器空。

UART 有 6bit、7bit、8bit、9bit 四种模式

UART0 和 UART1 支持接收超时和发送延时功能

清除标志

FM33A0xxEV 的大部分标志位默认是写 1 清 0，写 0 清除手册中会特别注释。

在写 1 清 0 的情况下，注意避免使用与等于和或等于的方式，这种用法可能会把寄存器中的已经置位的标志清除，可以采用直接赋值的方法，比如清除 bit0 的标志位，可以用寄存器直接等于 1 的方式。

RTC

如果应用代码中没有对时钟调校寄存器操作，在每次上电时 RTC 计时精度将因为时钟调校寄存器的随机值发生变化。在需要 RTC 时钟精度的应用中注意时钟调校寄存器的复位值问题，或者写 0，或者按温度进行相关的数据填充。

FLASH

FLASH 的最小编程单位是 32bit。编程前必须对 FLASH 进行擦除，FLASH 支持：全擦、扇区擦、页擦。

在操作 FLASH 时必须注意 RCHF 需要使能。

LCD

最大支持 8×40 、 6×42 、 4×44 的显示段数, 可以在 Active 模式、Sleep 模式和 DeepSleep 模式下工作。

选择 4COM 模式时只能选择 COM0-3, 6COM 模式选择 COM0-5, 8COM 选择 COM0-7。在和 LCD 连接时 COM 和 SEG 可以不必要按照一一对应连接, 如在 4COM 模式连接 LCD 的 COM0, MCU 的 COM0-3 都可以和 LCD 的 COM0 连接, SEG 和 COM 一样。

特别需要注意的是, 没有和 LCD 连接的 COM 和 SEG, GPIO 不能配置为 LCD 功能, 相应的 COM 和 SEG 也不能使能, 否则会影响 LCD 的正常显示。

LPUART

FM33A0XXEV 的 LPUART 是低功耗串口最高波特率为 9600。LPUART 的工作时钟为 32K, 需要注意的是 LPUART 的波特率的每一个 bit 不是等长的, 为了消除波特率的累计误差, 波特率的每一 bit 的长度是通过特别调制, 可以参看例程。

低功耗模式

FM33A0XXEV 的低功耗有 SLEEP 和 DEEPSLEEP 两种模式, SLEEP 模式为 3.6ua, DEEPSLEEP 模式 (RTC 走时+全部 RAM 保持+CPU 内核保持) 为 1.5ua。

内核和外设的都进入低功耗, MCU 才能得到理想的低功耗, PMU_CR.PMOD 的配置使得外设进入低功耗, SCR 和 WFI 的执行使得内核进入低功耗。具体的实现方法可以参看例程。

获得理想功耗的注意点:

- 1 在休眠时不使用的 GPIO 配置高阻态, 高阻态的配置有两种方式: GPIO 配置为输入模式并且输入禁止, 或者 GPIO 配置为输出模式, 开漏输出高电平;
- 2 SWD 的 SWCLK 和 SWIO 配置上拉电阻使能;
- 3 避免有输入管脚浮空;
- 4 注意 ADC 的 BUFFER 使能是否关闭;
- 5 低功耗下不需要工作的电路, 设计为无电或者电源可控状态;

休眠时外设和 GPIO 的寄存器都能保持不变。

MCU 休眠唤醒后默认的主时钟为 RCHF，可以通过 PMU_CR.WKFDEL 配置唤醒后的频率为 8MHZ、16MHZ、24MHZ 或 32MHZ。

唤醒后先进入中断，再进入主循环。也可以通过屏蔽中断(void __disable_irq(void);)，唤醒后直接执行 WFI 后的第一条指令。

低功耗定时器

FM33A0XXEV 除了 LPTIME 可以作为低功耗定时器，BSTIM 把工作时钟设置 LSCLK 同样可以在低功耗下定时。

修订记录:

V1.1 修改低功耗模式下 ADC 的 BUFFER 使能是否关闭的相关表述。