

KF8 芯片调试说明手册

1. 调试配置

1) P00、P01 须保留给调试器使用，即 P00 和 P01 脚作为调试通信脚使用，芯片代码不能使用该脚上的功能，如使用了脚上 PWM 输出，打开了通道 AD 使能，均会造成调试功能失效。简单的 IO 口控制，如调试 CLK 脚方向设置为输出，因为调试监控通信需要，监控代码会修改回 CLK 脚为输入，代码功能受到影响。同时应为 CLK 的输出低，会造成调试功能自身的暂停触发功能。也就是说本来需要无断点运行、或运行过程退出函数设定输出的函数时，会在设定了 CLK 输出为低后一定时间（触发时间）位置暂停，工具会给出错误结果并影响调试进行。

2) 外部复位 RST 脚作为调试功能的复位芯片功能，因此须要配置为复位脚功能，实际工具内部已经强制调试编译下该脚位为复位功能，正常编译时与用户选择一直。要求需要调试时该 IO 脚位的电路设计应避免冲突。通用作为 IO 的设计应用如果调试对应脚位功能不可用，代码应做屏蔽处理。

3) 配置字设置。 针对 8k 芯片引入了程序分页的概念，即 CALL、JMP 指令只能传递 4k 的指向，公司调试代码设计在了后 4k 中，工具层上自动配置字生成会根据芯片是 4k 或 8k 将配置位初始化。当使用手动配置字时，触发机制消失，因此错误的配置字造成调试入口异常，不能调试，需要手动配置调试页信息。调试机制：硬件触发调试中断，入口 0x2004 地址，内容为 CALL 指令，进入调试软监控代码。当前针对 8K 系列调试页控制在配置为的 bit8 位上，作为第 13 位地址数据。

2. 嵌汇编开发

1) C 语言调试时可以嵌入汇编代码，汇编代码不能过程调试，不建议整个函数

芯旺微电子

均为嵌汇编。如下：

```
void init_chip()
{
    __asm
        CALL    0XFFF
        MOV     OSCCAL0,R0
        CALL    0XFFE
        MOV     OSCCAL1,R0
        CALL    0XFFD
        MOV     VRECAL,R0
    __endasm;
}
```

需要修改为以下形式：

```
void init_chip()
{
    volatile unsigned char i=0;
    __asm
        CALL    0XFFF
        MOV     OSCCAL0,R0
        CALL    0XFFE
        MOV     OSCCAL1,R0
        CALL    0XFFD
        MOV     VRECAL,R0
    __endasm;
    i=1;
}
```

原因：工具缺陷，嵌入汇编代码在信息文件中缺少行号信息，也就是会作为上一个有行号的代码的补充被直接允许。

2) 嵌入汇编代码中不能有 CRET、IRET、RRET 之类的程序返回指令。单步跳过、单步跳出会分析机器代码，一个函数只能有一个返回指令。工具实现跳过函数运行将临时断点直接打到函数的最后 1 条即程序返回指令，如果前面存在条件分支并提前退出时，运行逻辑受到影响，程序监控会出错。

3. 调试差异

1) 软件调试占用额外的代码空间和数据空间，不同系列存在差异，编译提示冲突下应该代码更换使用资源，被使用的资源大概遵循规律，程序的尾部低于 256 个字节，数据区的第二区的后面少于 16 个数据，以及存在 0x70-0x7F 通用 RAM 型号的最多 2 个地址的占用。

2) 程序带操作 BEE/EE 的代码调试，外部供电调试时可能已被改写。主要因为芯片下完程序，以及未进入调试模式之前芯片已经运行。调试通过 RST 脚复位芯片重新运行，带基于 BEE/EE 条件的代码已经运行过。

3) 分支、循环语句位于函数最后时，条件不成立时光标指向异常。

分支例子：

```
return_type function_name(...)
{
    ...
    if(t)
    {
        l1;
        l2;
        l3;
        ...
        ln;
    }
}
```

当 t 不成立时光标将会指向 ln 表达式所在的行。

循环例子：

```
return_type function_name(...)
{
    ...
    while(t)
    {
        l1;
        l2;
        l3;
        ...
        ln;
    }
}
```

当 t 不成立时光标将会指向 ln 表达式所在的行。

原因：编译器产生的 coff 文件 return 指令所在行和 ln 在同一行，属于正常现象。

4. 带电调试

编程器设计支持选择外部电压，不支持外部电压调试说法属于调试默认先下载程序，编程 5V 电源和 VPP 脚 12V 高压，可能会和用户的电源系统冲突，如用户电源 3V。此时编程基本失败，无法进入调试功能。

可以 IDE 配置首选项下运行\调试中调试设置设置为手动下载。在项目属性中 ChipON 下的配置位设置中选择调试电压，如果外部供电，选择外部，选择内部时根据系统设计选择最接近的电压。调试编译后，优先断开外部电源，下载程序到芯片，切换到用户电源系统，此时芯片因为调试功能下的 hex，代码不运行，工作在等待主机状态。在程序代码编辑界面右键选择调试方式并硬件调试，随后调试功能进入。

需注意：下载程序要断开外部电源，除非外部用户电压也是 5v 的匹配。但针对 G 版本 4 线编程芯片，下载程序必须要求外部不能带电。程序更新或首次调试之前一定要下载调试版的 hex 到芯片中去。

5、 bug 反馈

调试功能还有功能不完善，可能会存在 bug，发现问题可以及时反馈。