

每一种CPU模式下对应有一组可以操作的寄存器。

任何模式下都有以下寄存器可以存取：

1）R0~R12

2）R13，the Stack Pointer,SP,栈指向

3）R14，the Link Register，LR,链路寄存器

4）R15，the Program Pointer，PC,程序指针

5）CPSR,the Current Program Status Register，当前状态寄存器

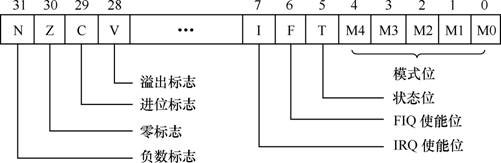
另外，特权模式(除了系统模式)还有一个SPSR,the Saved Program Pointer，保存状态寄存器。

**CPSR 程序状态寄存器**

用来保存当前程序的状态。

SPSR,程序状态保存寄存器，每一种CPU异常模式(除了system和user)下对应有一个自己的SPSR寄存器，用来在异常产生时保存CPSR寄存器的值，以便在异常之后恢复CPSR的值。所以格式和CPSR一模一样。

好，那么我们来看看CPSR/SPSR寄存器各个bit的具体含义：



组成如下：

1. ALU(Arithmetic Logic Unit，算术逻辑单元)结果状态标志备份
2. 当前的处理器模式
3. 中断使能标志位
4. 设置处理器的状态

**ALU结果状态标志位[31:28]：**

**N = Negative result from ALU**

经过ALU运算之后结果为负数N=1，如果为正数或者0情况，N=0。

例子：2-3 = -1，N = 1 （SUB）

0000 0000 0000 0000 0000 0000 0000 0010

+ 1111 1111 1111 1111 1111 1111 1111 1101 (-3负数以补码形式存在)

1111 1111 1111 1111 1111 1111 1111 1111 (-1的补码)

**Z = Zero result from ALU**

经过ALU运算之后结果为0时，Z=1，否则Z=0。

例子：2 - 2 = 0，Z = 1

0000 0000 0000 0000 0000 0000 0000 0010

+ 1111 1111 1111 1111 1111 1111 1111 1110 (-2补码形式存在)

0000 0000 0000 0000 0000 0000 0000 0000

**C = ALU operation Carried out or borrow**

分四种情况：

1. 在加法指令中，当结果产生进位，则C=1，表示无符号数运算发生溢出，其它情况下C=0。

例子：0xffffffff + 1 C=1

1111 1111 1111 1111 1111 1111 1111 1111

+ 0000 0000 0000 0000 0000 0000 0000 0001

0000 0000 0000 0000 0000 0000 0000 0000

1. 在减法指令中，当运算发生借位(即无符号数运算发生下溢)，则C=0，其它情况C=1。

2 - 3 = -1 C=0

1. 在对于移位操作的指令，C的值为最后一次被移出位的值。

例子，0XA5 >> 2 0xa5右移两位

0xa5 == 0000 0000 0000 0000 0000 0000 1010 0101

最后一次被移出的位是倒数第二位0，所以C = 0

1. 对于其他非加、减运算指令一般不会影响C的值。

**V = ALU operation Overflowed**

超出补码的范围 V=1

没有超出补码的范围 V=0

32位寄存器补码表示的范围是-0x7fffffff ~ +0x7fffffff

例: 0x7fffffff + 1 = 0x80000000(-2^31) V = 1

0111 1111 1111 1111 1111 1111 1111 1111

+ 0000 0000 0000 0000 0000 0000 0000 0001

1000 0000 0000 0000 0000 0000 0000 0000

例子：-1 -1 = -2 V=0

1111 1111 1111 1111 1111 1111 1111 1111

1111 1111 1111 1111 1111 1111 1111 1111

1 1111 1111 1111 1111 1111 1111 1111 1110

**中断禁止位：**

I = 1: 禁止 IRQ.

F = 1: 禁止 FIQ

**T 位：**

T = 1; 处理器处于 Thumb状

T = 0; 处理器处于 ARM 状态

**Mode位(M0~M4)**：处理器模式位

10000 User mode;

10001 FIQ mode;

10011 SVC mode;

10111 Abort mode;

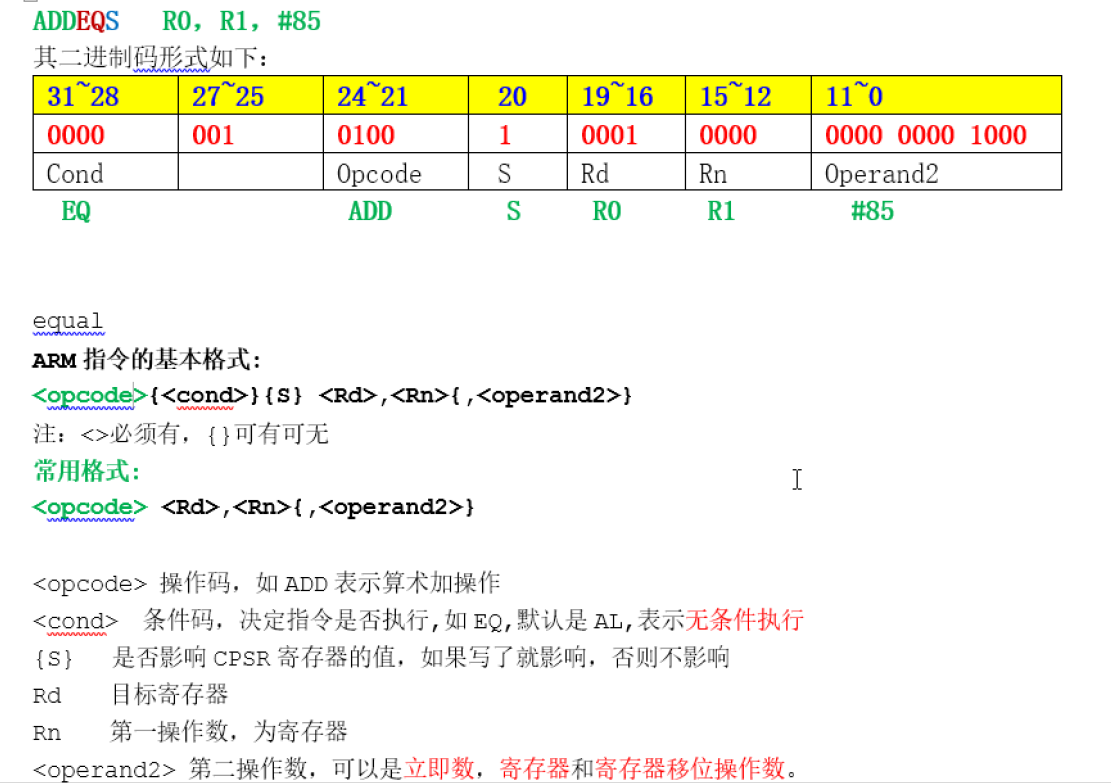
11011 Undfined mode;

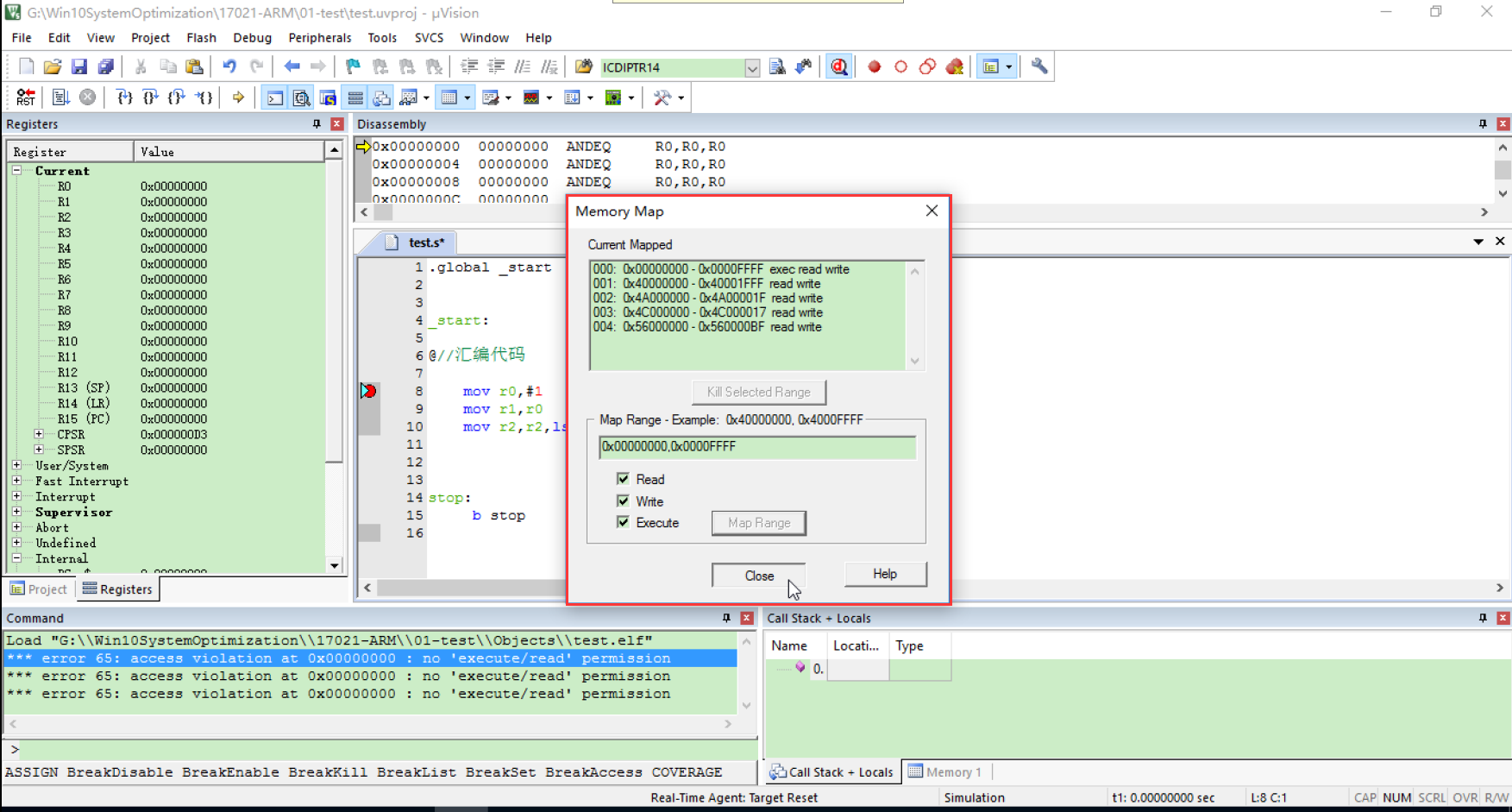
11111 System mode;

10110 Monitor mode;

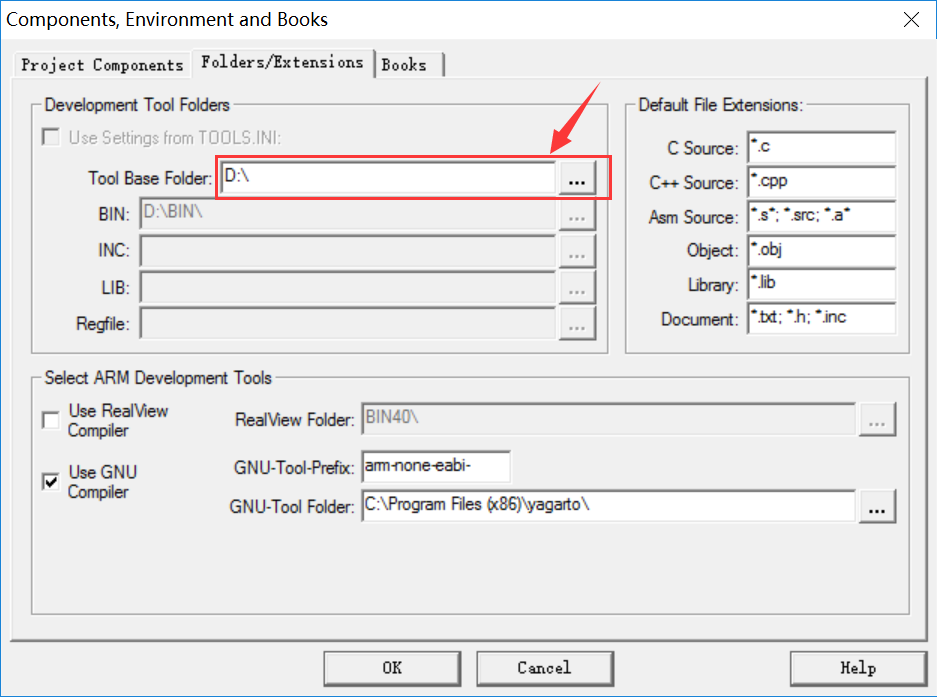
10010 IRQ mode;



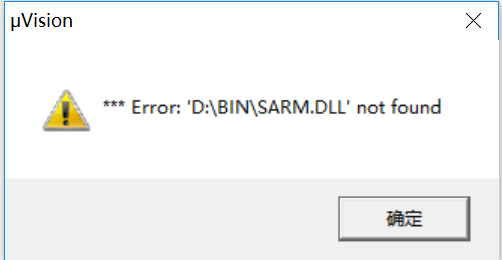


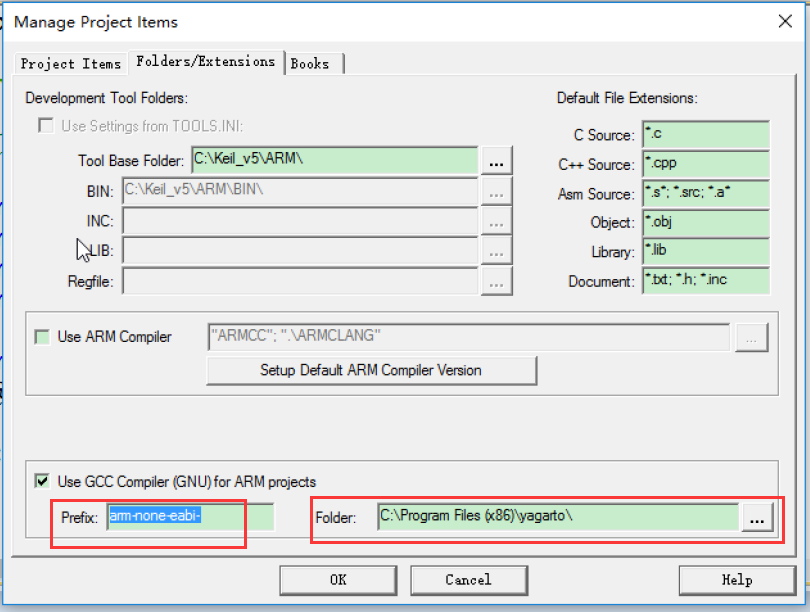


keil 编译提示can't open Files for reading,no such file or dictionary.

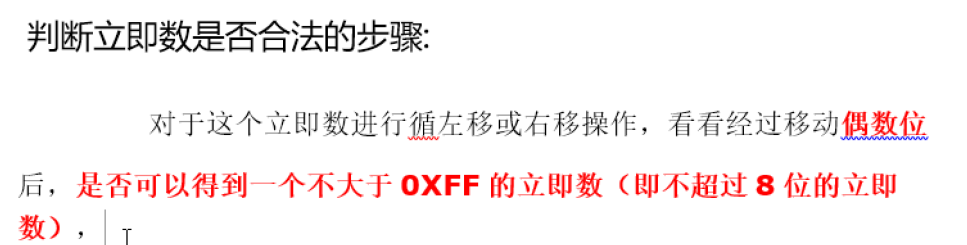


这里改成任意路径；

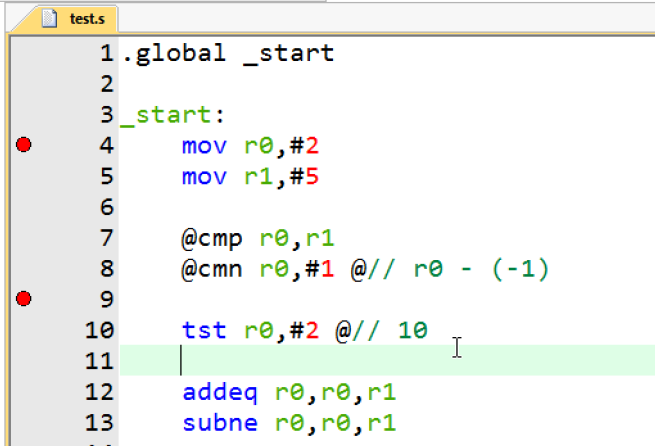


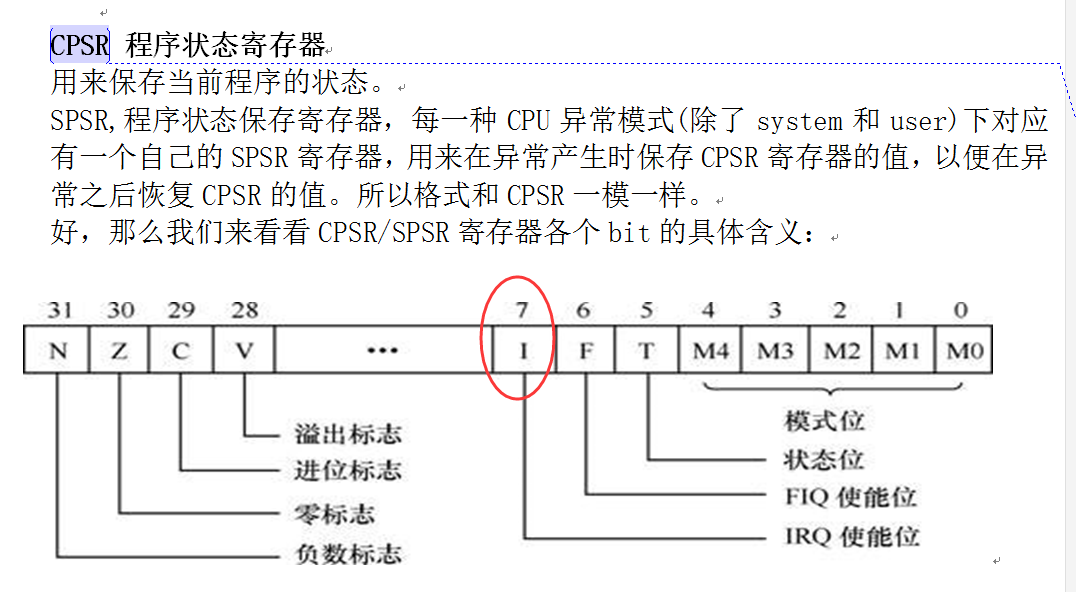


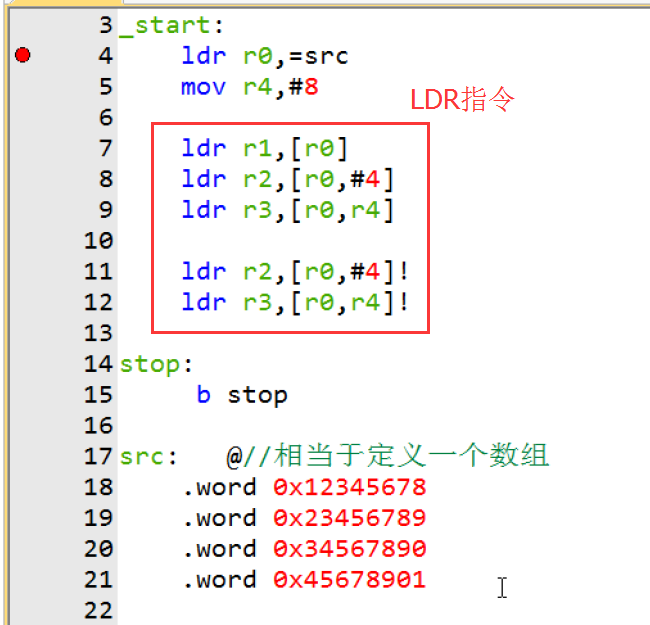
交叉工具链

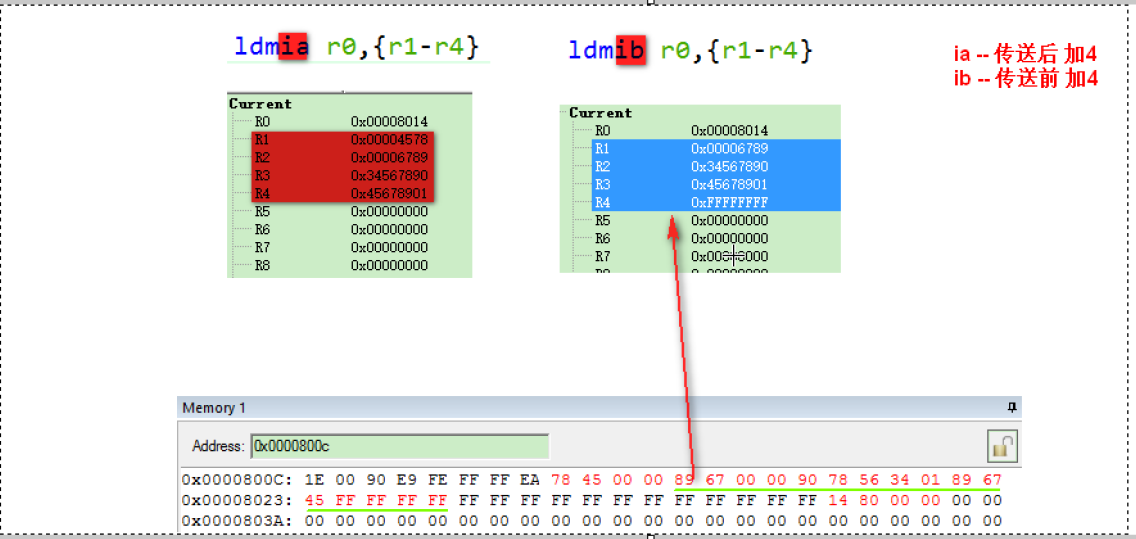


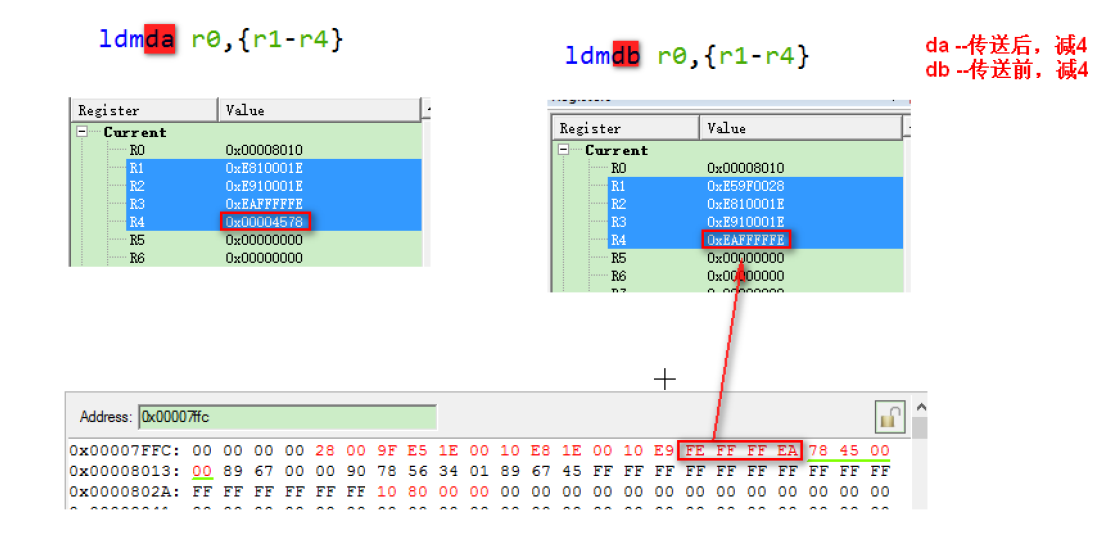
比较指令











生成.bin文件的方法

a).使用makefile生成。

