

一. 指令与伪指令的本质区别是：指令经编译后，会生成对应的机器码。而伪指令经编译后没有指令代码。

二. (1) 程序定位的作用；

(2) 为非指令代码进行定义；

(3) 为程序完整性做标注；

(4) 有条件的引导程序段。

三. 宏指令与伪指令的主要区别是：在汇编时，这些宏指令被替换成一条或两条真正的 ARM 或 Thumb 指令。而伪指令经编译后没有指令代码。

四. 伪指令 **LTORG** 用来说明某个存储区域为一个用来暂存数据的数据缓冲区，也叫文字池或数据缓冲池，大的代码段也可以使用多个数据缓冲池；其目的是，防止在程序中使用 **LDR** 之类的指令访问时，可能产生的越界。

五. MAP 8192

consta FIELD 4

constb FIELD 4

x FIELD 8

y FIELD 8

string FIELD 16

六. 对应代码如下：

```
AREA mycode, CODE, READONLY
ENTRY
```

Main

```
LDR R0,=NewItem
SUB R0,R0,#4
LDR R1,NewItem
LDR R3,Start
CMP R3,#0
BEQ Missing
LDR R4,[R0],#-4
```

Loop

```
CMP R1,R4
BEQ Done
SUBS R3,R3,#1
LDR R4,[R0],#-4
BNE Loop
```

Missing

```
MOV R3,#0xFFFFFFFF
```

Done

```
LDR R5,=Index
STR R3,[R5]
SWI 0x11
```

Start

```
DCD 0x4
DCD 0x0000138A
DCD 0x000A21DC
DCD 0x1F5376
DCD 0x9018613
```

NewItem

DCD 0x1F5376

AREA mydata, NOINIT

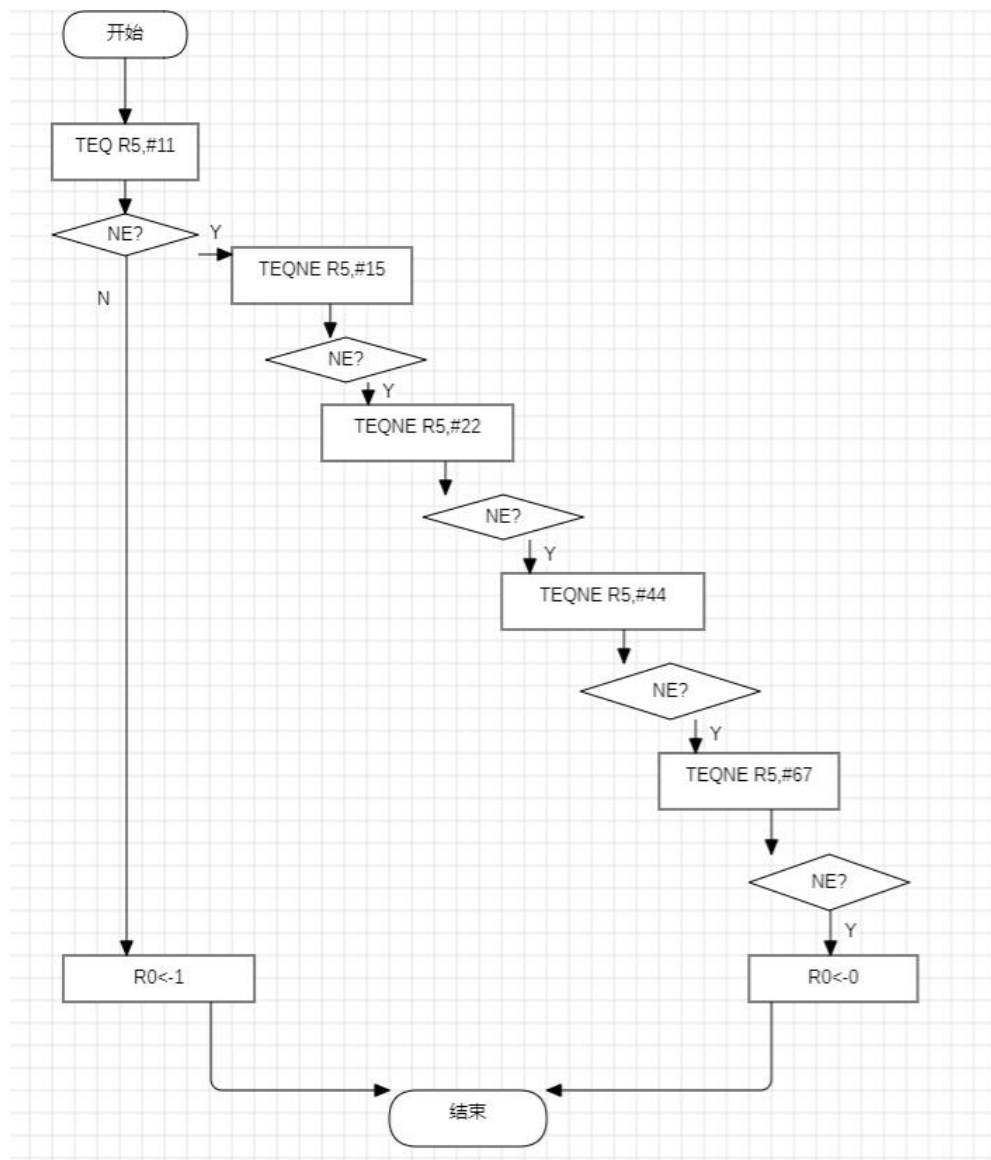
MAP 0x40000000

Index

FIELD 4

END

七. 流程图如下:



程序如下:

```
AREA Test, CODE, READONLY
ENTRY
TEQ R5, #11
TEQNE R5, #15
TEQNE R5, #18
TEQNE R5, #22
TEQNE R5, #44
TEQNE R5, #67
MOVEQ R0, #1
MOVNE R0, #0
END
```

八. 代码如下:

```
AREA Test, CODE, READONLY
EXPORT subxx
subxx
SUBS R0, R0, R1
MOV PC, LR
END
```

九. 代码如下:

```
void strcpy( char *d, char *s)
{
    while (*s != '\0')
        { *d= *s; d=d+1; s=s+1; }
    *d= *s;
}
```