

```

;
; STARTUP.A51: 用户上电初始化程序。
;
;-----
;
; 用户定义需上电初始化的内存空间
;
; 使用以下EQU命令可定义在CPU复位时需用0进行初始化的内存空间。
;
; ; IDATA 存储器的空间的绝对起始地址总是0.
IDATALEN EQU 80H ; 需用0进行初始化的IDATA存储器空间的字节数。
;
; XDATASTART EQU 0H ; XDATA存储器空间的绝对起始地址。
XDATALEN EQU 0H ; 需用0进行初始化的XDATA存储器的空间字节数。
;
; PDATASTART EQU 0H ; PDATA存储器的空间的绝对起始地址。
PDATALEN EQU 0H ; 需用0进行初始化的PDATA存储器的空间字节数。
;
; 注意: IDATA 存储器的空间在物理上包括了8051单片机的DATA和BIT存储器空间。
; *听*说*至少要保证与C51编译器运行库有关的存储器的空间进行0初始化。不知是否?
;
;-----
;
; 再入函数模拟初始化
;
; 以下用EQU指令定义了再入函数模拟堆栈指针的初始化
;
; 使用SMALL存储器模式时再入函数的堆栈空间:
IBPSTACK EQU 0 ; 使用SMALL存储器模式再入函数时将其设置成1.
IBPSTACKTOP EQU OFFH+1 ; 将堆栈顶设置为最高地址+1.
;
; 使用LARGE存储器模式时再入函数的堆栈空间.
XBPSTACK EQU 0 ; 使用LARGE存储器模式再入函数时将其设置成1.
XBPSTACKTOP EQU OFFFH+1 ; 将堆栈顶设置为最高地址+1.
;
; 使用COMPACT存储器模式时再入函数的堆栈空间.
PBPSTACK EQU 0 ; 使用COMPACT存储器模式再入函数时将其设置成1.
PBPSTACKTOP EQU OFFFH+1 ; 将堆栈顶设置为最高地址+1.
;
;-----
;
; 使用COMPACT存储器模式时64K字节XDATA存储器空间的分页定义
;
; 以下用EQU指令定义PDATA类型变量在XDATA存储器空间的页地址。
; 使用EQU指令定义PFAGE时必须与L51连接定位器PDATA指令的控制参数一致。
;
; PPAGEENABLE EQU 0 ; 使用PDATA类型变量时将其设置成1.
; PPAGE EQU 0 ; 定义页号.
;
;-----
;
; NAME ?C_STARTUP ; 模块名为: ?C_STAUTUP
;
; ?C_C51STARTUP SEGMENT CODE ; 代码
; ?STACK SEGMENT IDATA ; 堆栈
;
; RSEG ?STACK ; 堆栈

```

DS 1

EXTRN CODE (?C_START) ; 程序开始地址
PUBLIC ?C_STARTUP

?C_STARTUP: CSEG AT 0x8000 ; 定义用户程序的起始地址,用MON51仿真器时可能有用。
LJMP STARTUP1
RSEG ?C_C51STARTUP

STARTUP1:
;
; 初始化串口

MOV SCON,#40H
MOV TMOD,#20H
MOV TH1,#0fdH
SETB TR1
CLR TI

; 单片机上电IDATA内存清零, 如果不需要上电清零IDATA, 可以注销IF到IFEDN之间的话句。
; 或者修改IDTALEN的长度。为了具有掉电保护功能, 不知IDTALEN多长为好?

IF IDATALEN <> 0
MOV R0,#IDATALEN - 1
CLR A
IDATALOOP: MOV @R0,A
DJNZ R0,IDATALOOP

ENDIF
;
; 单片机上电XDATA内存清零, 如果不需要上电清零XDATA, 可以注销IF到IFEDN之间的话句。
; 或者修改XDATALEN的长度。

IF XDATALEN <> 0
MOV DPTR,#XDATASTART
MOV R7,#LOW (XDATALEN)
IF (LOW (XDATALEN)) <> 0
MOV R6,#(HIGH (XDATALEN)) +1
ELSE
MOV R6,#HIGH (XDATALEN)
ENDIF

XDATALOOP: CLR A
MOVX @DPTR,A
INC DPTR
DJNZ R7,XDATALOOP
DJNZ R6,XDATALOOP

ENDIF
;
; 送PDATA存储器页面高位地址。

IF PPAGEENABLE <> 0
MOV P2,#PPAGE

ENDIF
;
; 单片机上电PDATA内存清零, 如果不需要上电清零XDATA, 可以注销IF到IFEDN之间的话句。
; 或者修改PDATALEN的长度

IF PDATALEN <> 0
MOV R0,#PDATASTART
MOV R7,#LOW (PDATALEN)

```

                CLR      A
PDATALOOP:    MOVX     @R0,A
                INC     R0
                DJNZ   R7,PDATALOOP

ENDIF
;
; 设置使用SMALL存储器模式时再入函数的堆栈空间.
IF IBPSTACK <> 0
EXTRN DATA (?C_IBP)

                MOV     ?C_IBP,#LOW IBPSTACKTOP
ENDIF
;
; 设置使用LARGE存储器模式时再入函数的堆栈空间.
IF XBPSTACK <> 0
EXTRN DATA (?C_XBP)

                MOV     ?C_XBP,#HIGH XBPSTACKTOP
                MOV     ?C_XBP+1,#LOW XBPSTACKTOP
ENDIF
;
; 设置使用COMPACT存储器模式时再入函数的堆栈空间.
IF PBPSTACK <> 0
EXTRN DATA (?C_PBP)
                MOV     ?C_PBP,#LOW PBPSTACKTOP
ENDIF
;
; 设置堆栈的起始地址。
                MOV     SP,#?STACK-1      ; 例如: MOV     SP,#4FH
;
; This code is required if you use L51_BANK.A51 with Banking Mode 4
; 如果你的程序使用了Mode 4 程序分组技术, 请启动下面的程序,不会吧你的程序超过64K, 利害。
; EXTRN CODE (?B_SWITCH0)
;                CALL    ?B_SWITCH0      ; init bank mechanism to code bank 0
;                                           程序从第一组bank 0 块开始执行
; 跳转到用户程序MAIN函数
                LJMP    ?C_START

                END
                ; I INSHENGFENG
                ; 2001_10_26
```