

[首页](#) >> [AVR 单片机开发工具](#) >>

AVR 系列单片机 (MCU) 问答

问: 我想使用 AVR 单片机中的 Mega 系列, 有哪些开发工具支持这种单片机?

答: AVR 单片机中的 Mega 系列有一套完善的开发工具评估/编程板,

ATMEL 的 STK500(980 元/套)起步工具包是适用于开发 Atmega103 芯片的对芯片的编程是通过在系统编程(ISP)接口完成的仿真器

ATMEL 的 ATICE30 可适用于实时仿真

C 编译器:

所有的 C 编译器均已在 ATMEL 网站上有关第三方工具供应商的网页上列出;ATMEL 公司在它的网站:<http://www.atmel.com> 上还提供了许多自由软件它们可用于对 AVR 单片机的程序进行汇编和模拟这些软件可以自由下载, 上述产品也可通过 ATMEL 授权的地区分销商和销售代理获得。

问: 有哪些支持 AVR 的起步工具包

答: STK100,SL-AVR,SL-OKAVR,SL-AVRL 支持 AVR 的 Tiny 系列单片机

STK200,SL-AVR,SL-OKAVR,SL-AVRL 支持 AVR 单片机即 AT90S 系列

STK300,SL-AVR,SL-OKAVR,SL-AVRL 还支持 AVR 的 Mega103 单片机.

STK500,SL-AVR,SL-OKAVR,SL-AVRL 还支持 AVR 的 Mega103 单片机

所有的起步工具包都包括按钮 LED 指示灯 RS232 通信接口 I/O 口通过分布在 PCB 板边沿的连接插头可以使用 STK200/300 起步工具包还有一条在线编程 ISP 电缆它可用以对 AVR 芯片编程 STK300 还提供了 AVR 应用程序生成器用来生成外围设备的初始化代码

问: 即使看门狗没有使用我也有时在 AVR Studio 调试软件的状态栏中看到“WDT reset” WDT 复位的信息这是为什么?

答: 当目标器件运行在较高的频率大于 6 MHz)时来自扁平电缆 FPC 电缆的噪声信号可能错误地触发 WDT 复位检测电路这对程序的仿真和执行没有影响它们不会由于噪声的原因而产生复位噪声仅仅影响 AVR Studio 调试软件的信息处理在这种情况下不要理睬这个信息当目标器件运行在较低的频率小于或等于 4 MHz 时应该不会出现这个问题

问: 我刚买了一台 ICE 200 仿真器在使用不同的芯片和不同的时间运行诊断程序时都是失败的而其它一切正常这是什么原因?

答: 在 ICE200 仿真器的第一批产品中是一个固件监控程序的问题而导致这个现象这个现象可以通过升级新版本的监控程序来修正当使用 2.02 版或更高版本的 AVR Studio 软件时 AVRStudio 将读取固件版本并检查这个问题如果你的 ICE200 需要升级它会引导你完成升级处理这个检查工作只有在装载应用程序时才能被执行而当运行诊断程序时是不被启动的如果你首先尝试运行诊断程序诊断仿真器诊断程序不会检查固件版本并且可能诊断失败如果你碰巧有这种情况不要理睬诊断失败而直接去装载应用程序这时你将可以升级 ICE200 的监控程序在升级后再反过来运行诊断程序诊断结果应该全部是正确了(注现在销售的 ICE200 已没有这个问题了)

问: 如何检查我的 AVR 仿真器的版本号?

答: 在仿真器被连接的时候打开 AVR Studio 程序选择菜单中 Help->About->Info 来获得你的仿真器的有关信息 AT90ICEPRO 在 1.20 之前的版本和 ATmegaICE 在 1.11 之前的版本是不能被 AVR Studio 软件检测到的较新的版本已经有了而且可以从 Atmel 网站自由下载.

问: 程序在仿真器或单片机中不工作,而在模拟器中看来工作很正常这是什么原因?

答: 这是堆栈指针设置中的常见问题对所有带 SRAM 的 AVR 单片机堆栈指针必须设置在 SRAM 最后的地址在汇编程序中这样做是正确的

```
ldi R16 low(RAMEND) 装载堆栈指针的低字节到 R16
```

```
out SPL R16 输出这个数值到堆栈指针寄存器
```

```
ldi R16 high(RAMEND) 装载堆栈指针的高字节到 R16
```

```
out SPH R16 如果 AVR 的 SRAM 小于 256 字节就将这行注释掉
```

在 C 程序中堆栈指针是在启动代码中自动设置的注意链接文件(XCL 文件)定义了堆栈指针的位置请参考应用笔记 AVR032

---- C 编译器的链接文件”

问：从哪里我能找到完整的 AVR 指令集？

答：指令集在 Atmel 的网站的 Products->AVR 8 bit RISC->Datasheets 下面的网页上或者直接连接 <http://www.atmel.com/atmel/products/prod200.htm> 网页；<<AVR 嵌入式高速单片机原理与应用>>附录

3 AVR 指令速查表

问：当开始做一个使用 AVR 单片机的新项目时最常见最共同问题有哪些？

答：最常见最共同的错误是忘记对不带硬件堆栈的芯片设置堆栈指针对于 AT90S8515 单片机上这样做是妥当的

```
LDI R16 low(RAMEND)
```

```
OUT SPL R16
```

```
LDI R16 high(RAMEND)
```

```
OUT SPH R16
```

注在 ATMEL 的汇编语言源程序中经常将 R16 寄存器用 temp 符号来代替

另一个常见的错误是将端口作为输出使用时忘记设置端口在 AVR 单片机中设置端口作为输出使用是通过写一个 1 到数据方向寄存器(DDR)中完成的比如设置端口 PORTB 输出高电平

```
LDI R16 0xFF 装载十六进制数 FF 到寄存器 R16
```

```
OUT DDRB R16 设置端口 PORTB 作为输出使用
```

```
OUT PORTB R16 设置端口 PORTB 输出高电平
```

使用 AVR 汇编器时第三个常见的错误是在使用装载程序存储器(LPM)指令时指向了程序存储器中错误的地址 AVR 的程序存储器是组织成字 16 bit 的形式而 LPM 指令是读取字节 8bit LPM 指令可以读 16 位字的高位字节或低位字节中的任意一个因为这个原因必须将 16 位的程序存储器地址分解成两个指向你所希望装入的字节地址

问：我在 AT90ICEPRO 中使用回溯(Trace 功能选项感觉很困难好象每次都不能启动回溯功能为什么？

答：让我们假设你在地址 0x20 处停止仿真如果你现在设置启动回溯功能的触发地址是同一个地址即也是 0x20 那么在程序重新到达 0x20 之前的回溯功能是无效的原因是触发条件与断点信息应该在指令之前装入 AVR 核中所以尽管指令和触发条件在 0x20 地址但对在这之前装入 AVR 核中的指令没有任何效果如果你设置触发条件为一个程序还没有到达的地址应该不会有任问题

问：Mega103/603 微控制器的哪些引脚用于在线编程(ISP)？

答：Mega103/603 微控制器用于在线编程界面的引脚如下列

```
SCK - SCK (PB1 pin 11)
```

```
MISO - PDI (PE1 pin 3)
```

```
MOSI - PDI (PE0 pin 2)
```

```
RESET- RESET (RESET pin 20)
```

PEN(编程使能)引脚和 RESET 引脚可以选择其中一个使用这两个引脚有同样的功能它们都激活内部 iRESET 信号)但 PEN 引脚只有在上电时(VCC 大约 1.2V)被读取如果 PEN 引脚不使用推荐将此引脚接 VCC 所有其它型号的 AVR 单片机都是使用 RESET 引脚来允许使能串行编程的

问：我不能使用 AVR 仿真器正确地驱动外部负载当我使用电压表来测量仿真器 POD 上的引脚时好象引脚不能驱动负载这是什么错误？

答：这是 1.21 版的 ICEPRPO 和 1.10 版的 MegaICE 存在的问题，可用于仿真器升级的新版本软件可以从 <http://www.atmel.com> 网站的 products->AVR 8bit RISC->software 网页中下载

为升级 ICEPRPO 的版本到 1.22 版下载 UPGRD122.ZIP 文件

为升级 MegaICE 的版本到 1.11 版下载 M111UPGR.ZIP 文件

这些文件是压缩文件它包括了一个适用于仿真器的新的程序文件和一个 PC 程序 PC 程序可以用来对代码检查和对仿真器自动升级

问：我使用了带 A/D 转换器的 AVR 单片机单片机工作非常好我也喜欢 AVR 的速度但我有一个问题当我改变 A/D 转换器的输入通道时好象 A/D 转换器读取的数据仍然是上一个通道的，如果我第二次读我又得到正确的数值为什么会发生这件事？

答：A/D 控制和状态寄存器 ADCSR 中有两个位是用来检测 A/D 转换的结束的当一个新的转换启动时 ADSC 启动转换控制位是用于测试的不管是否准备就绪前一次的转换结果仍然被送入 A/D 数据寄存器当读取 A/D 数据寄存器时读到的自然仍是上

一次的转换结果

为了得到最近的转换结果应该测试 A/D 中断标志位 ADIF 或使能全局中断并且在中断服务程序中读取

例子代码

```
ldi R16 1 选择通道
out ADMUX R16
sbi ADCSR adif 复位中断标识
sbi ADCSR adsc 开始 A/D 转换
wait sbis ADCSR adif 等待直到 ADIF 已经置位
rjmp wait
```

问： 当将一个文件装入 AVR STUDIO 时它始终使用我第一次使用过的文件工作这是什么错误？

答：AVD 文件包含了有关窗口设置和当前目标文件的信息有时由 AVR STUDIO 创建的 AVD 文件可能被破坏根据文件破坏部位的不同会出现不同的现象当运行 AVR STUDIO 过程中出现奇怪问题时首先应尝试删除位于存放源代码的文件夹中扩展名为 avd 的文件

问： 在 AVR 系列中如 AT90S2313 单片机全部数据存储单元不超过 256 字节而且它不能访问外部存储器能否忽略 X Y 和 Z 指针的高位字节而仅使用指针的低位字节来访问可以使用的存储器？

答：在全部数据存储单元寄存器 I/O 寄存器和 SRAM 少于 256 字节的 AVR 单片机中当使用

LD/ST 指令访问数据存储单元时 CPU 将忽略指针的高位字节这时允许每个指针的高位字节(R27、R29 和 R31)用于普通的储存用途自增和预减指令(如 LD -Z Rd)不会影响高位字节注意当使用 Z 指针访问 FLASH 程序存储器和使用 ADIW/SBIW 指令时将影响使用的 Z 指针的高位字节

问： 在 IAR C 编译器中怎样才能访问 AVR 内部的 EEPROM 数据存储单元

答：AVR 内部的 EEPROM 是 AVR 微控制器 I/O 存储器的一部分以 I/O 接口形式访问不能使用普通变量进行访问 IAR 在标准库例程“ina90.h”中定义了特殊的宏来读写 EEPROM

读写 EEPROM 的宏有下列原形

```
_EEGET(VAR ADR) /* 从 EEPROM 的地址 ADR 读取数值*/
_EEPUT(ADR VAL) /* 写 VAL 到 EEPROM 的地址 ADR *
```

IAR C 编译器不支持将常量定义到 EEPROM 中不能生成 EEPROM 的初始化数据文件

用 AVR 汇编器来做这件事是很容易的先写一些定义 EEPROM 中常量的代码

```
.ESEG
.org 0x50
datatable: .db $01,$02,$03,$04,$05
dtableend: .db $00
```

当这段代码被汇编后它产生一个.EEP 文件这个文件可以被下载到微控制器 MCU 或仿真器中这个.org 声明给出数据的地址这个变量在 C 编译器中可以用_EEGET()和_EEPUT()进行访问

问： 我无法使 MCU 在掉电模式时消耗的电流下降到数据手册中的值我应该怎么做？

答：当 AVR 进入掉电模式时 I/O 引脚的状态应该是不变化的如果你有一些未连接引脚是三态输入并且没有上拉这些引脚的电平是浮动的并且很可能引起振荡从而导致额外的电流消耗为解决这种问题应该激活所有未连接的输入引脚的上拉电阻这会使这些引脚具有一个明确的逻辑电平在由于引脚振荡而产生这个问题的例子中在活动模式非休眠状态打开未使用引脚的上拉是一个好方法

其次你再查看一下连接到输出引脚的接线任何输出引脚上的负载都会在掉电模式时消耗额外的电流

问： 我观察到现有的微控制器在上电过程中 I/O 端口是活动的在我的应用中有一个问题系统一旦设定 I/O 引脚中的一个为高电平 MCU 可以关闭电源而即使用一个电阻把该引脚电平拉低在上电过程中它有时仍然出现高电平因此当电源打开时系统立刻关闭在使用者看来系统不能工作如果我换成 AVR 我猜想会有一些同样的问题？

答：大多数微控制器是同步复位比如在 RESET 尚未完全完成时 MCU 需要一定数量的有效时钟在上电过程中当 VCC 上升到启动晶体振荡器的数值之前上电复位是有效的但不能影响 I/O 引脚的数值而且 I/O 引脚是处于随机状态 AVR 的 I/O 端口与复位是异步的而且保证在整个上电过程中是处于三态换成 AVR 单片机将可以解决你的问题

问： 怎样处理 AVR 微控制器的 NC 引脚空引脚？

答：不要将任何线路连接到 NC 引脚 NC 引脚是为将来的需用而保留的

问：在 AVR 汇编器中怎样定义字符串常量

答：字符串可以作为常量定义在 FLASH 存储器或 EEPROM 数据存储器中

例如在 FLASH 中定义一个字符串常量

```
.CSEG
```

```
fstring .db "This is a string in flash" 0x00
```

在 EEPROM 中规定一串常量：

```
.ESEG
```

```
eestring .db "This is a string in EEPROM" 0x00
```

问：我正在编写一个很大的汇编程序当我往程序中加入新的行时文件末尾的字符好象消失了怎样才能防止这种现象？

答：汇编器的编辑器有 30K 字节的文件尺寸限制如果要克服这个限制文件必须被分成几个模块并用 “.include” 链接还有一种方法使用其它编辑器如 NotePad UltraEdit 程序等和命令行方式工作的汇编器(DOS 版本)

问：AT90S8515 有 8KB 的程序存储器由于 RJMP 和 RCALL 指令只能在 2K 的相对范围内跳转而这个部件的指令集没有 JMP 或 CALL 指令我怎样做才能跳转到全部的程序存储器中

答：程序存储器是组织成 4Kx16 的形式的所以只有 4K 的程序存储器地址空间在汇编器中选择 “Options >> Wrap Relative Jumps” 选项这时将允许你跳过程序存储器的边界例如如果你从 \$FFE 至 \$00A 做一个相对跳转程序计数器将被增加 12 并且隐藏了程序存储器的边界这个功能只能用在 8K 的单片机 4K 的单片机不需要隐藏边界 16K 的单片机需要使用 JUMP 和 CALL 指令。

问：当我使用 UART 时定时/计数器的任意一个是否都可用于产生波特率？

答：不对有一个专用的定时器被 UART 用来产生波特率所有定时/计数器只可以用作普通用途

问：怎样才能从外部将 AT90S1200 从掉电模式下唤醒？

答：要从掉电模式唤醒 AT90S1200 你必须使能外部 INTO 中断并且是低电平触发中断当你以掉电模式进入休眠状态时如果 INTO 引脚上低电平的保持时间大于 16384 个内部 RC 振荡周期时 AT90S1200 将被唤醒内部 RC 振荡器是用于延时 MCU 的启动直到 XTAL 振荡器工作稳定参考 AT90S1200 数据簿 RC 振荡器的频率是受 VCC 电压影响的

问：当我使用 SBI 和 CBI 指令来设置或清除 I/O 端口的一个信号位时是否会影响同一端口的其它位？

答：不会不像多数单片机那样在 AVR 单片机中允许你 100%安全地操作 I/O 端口的信号位这也适用于整个端口的操作有疑问可参考每个 I/O 端口的三个地址

问：为什么每个 I/O 端口有三个地址？

答：为使你能建立 100%安全的系统 AVR 支持真正的读-修改-写 I/O 端口如果你希望读取 I/O 引脚的物理电平读 PIN 寄存器当你希望改变输出时读 PORT 锁存能确保正确的数据写回到端口保证全部输出这种方式始终能给出你所希望的结果而不依赖于引脚的物理电平这个特点使你省去了为建立一个安全的系统而拷贝你的端口数据进入存储器的所有工作这个工作使用了许多指令当你使用 SBI 和 CBI 指令来设置/清除 I/O 端口的信号位时必须始终使用 PORT 的地址

问：为什么 SBI 和 CBI 指令只能对 \$00 - \$1F 的 I/O 寄存器进行操作？

答：所有 AVR 指令中除少数外都为两个字节长度这也意味着只有 65,536(64K)种可能的组合来安排指令集当我们指定 AVR 指令集时采取一些折衷的办法是为了尽可能充分地利用这 64K 种组合不像 CISC 微控制器指令可以是一个两个三个或更多个字节在 AVR 结构上我们不能实现全部指令都象这样作为例子在一个立即寻址而且包含全部 32 个寄存器地址的指令中常数需要 8 位而寄存器地址需要另外的 5 位这种指令将占用全部指令组合空间中的 8K 换句话说我们只能安排八个这样的指令而没有更多的指令能够实现如果做一个 17 位长度的指令是不经济和不便的方案不会被考虑在设计 AVR 指令集的过程中设置我们听取了許多建议我们请教了 C 编译器专家他们在如何调整指令集来适应 C 编译器方面提出了许多意见作为例子编译器专家建议我们为 SBCI 带进位的立即数减法牺牲 ADDI 指令

对那些缺少的指令操作也是方便的 AVR 的代码效率应该能证明我们已经找到一种在实现的指令和省略的指令之间进行折衷的好办法

问：为什么立即寻址指令在寄存器 R0-R15 中不能工作？

答：所有 AVR 指令中除少数外都为两个字节长度这也意味着只有 65,536(64K)种可能的组合来安排指令集当我们指定 AVR 指令集时采取一些折衷的办法是为了尽可能充分地利用这 64K 种组合不像 CISC 微控制器指令可以是一个两个三个或更多个字节在 AVR 结构上我们不能实现全部指令都象这样作为例子在一个立即寻址而且包含全部 32 个寄存器地址的指令中常

数需要 8 位而寄存器地址需要另外的 5 位这种指令将占用全部指令组合空间中的 8K 换句话说我们只能安排八个这样的指令而没有更多的指令能够实现而做一个 17 位长度的指令是不经济和不方便的方案不会被考虑在设计 AVR 指令集的过程中设置我们听取了許多建议我们请教了 C 编译器专家他们在如何调整指令集来适应 C 编译器方面提出了许多意见作为例子编译器专家建议我们为 SBCI 带进位的立即数减法牺牲 ADDI 指令

对那些缺少的指令操作也是方便的 AVR 的代码效率应该能证明我们已经找到一种在实现的指令和省略的指令之间进行折衷的好办法

问：为什么没有 EORI 指令？

答：所有 AVR 指令中除少数外都为两个字节长度这也意味着只有 65,536(64K)种可能的组合来安排指令集当我们指定 AVR 指令集时采取一些折衷的办法是为了尽可能充分地利用这 64K 种组合不像 CISC 微控制器指令可以是一个两个三个或更多个字节在 AVR 结构上我们不能实现全部指令都象这样作为例子在一个立即寻址而且包含全部 32 个寄存器地址的指令中常数需要 8 位而寄存器地址需要另外的 5 位这种指令将占用全部指令组合空间中的 8K 换句话说我们只能安排八个这样的指令而没有更多的指令能够实现而做一个 17 位长度的指令是不经济和不方便的方案不会被考虑在设计 AVR 指令集的过程中设置我们听取了許多建议我们请教了 C 编译器专家他们在如何调整指令集来适应 C 编译器方面提出了许多意见作为例子编译器专家建议我们为 SBCI 带进位的立即数减法牺牲 ADDI 指令

对那些缺少的指令操作也是方便的 AVR 的代码效率应该能证明我们已经找到一种在实现的指令和省略的指令之间进行折衷的好办法

问：我用 "sbr r30,3"指令在寄存器 30 中设置 D3 位好象对 D3 位不起作用但 D0 位与 D1 位可以被设置我的做法在哪里有错误？

答：“SBR”可以用来在一个寄存器中同时设置多位(象“CBR”可以用来同时清除寄存器中的多位)指令的第二个参数不是指定位号而是用来与寄存器中的数作逻辑或运算在这个例子中\$03 与 R30 的内容相或所以导致最低的两位置位为了只设置 D3 位应尝试下列方法的一种

```
sbr r30 $08
sbr r30 0b00001000
sbr r30 (1<<3)
```

上面三行都会产生同样的结果你可以选择一个你最习惯的用法

问：在我的多任务系统中除了中断标志以外我还需要定义可以设置和清除的标志我应该怎样做？

答：每个标志使用一个寄存器为了设置标志用

```
ldi flag 01
```

或者如果你想使用低位的寄存器(R0-R15)在复位后先清除它然后用

```
inc flag
```

来设置标志

当检查和清除标志时这样做

```
lsl flag 移动标志到进位标志位(标志清除)
```

```
brne flag_was_set 如果标志被设置就跳转
```

象在你所有的中断例程中一样在退出之前要恢复状态寄存器这种方式十分安全

问：在 AT90S1200 中如何实现常数表功能？

答：由于 AT90S1200 没有 LPM 指令比较有效的方法是将一个较小的常数表放置在 EEPROM 中你可以使用 EEPROM 地址寄存器和 EEPROM 数据寄存器有效地直接访问常数表如

```
.ESEG 定义 EEPROM 段
.db my_var00=$45
.db my_var01=$4c
.db my_var02=$5f
.CSEG 定义通用代码段
```

如果你的表不适合于放置在 EEPROM 中你可以象下面这样放置在 FLASH 中

```
.def tp =r16 表指针表索引
.def output=r17 输出表的中常数
```

```

table
ldi output,$45
cpi tp,01
breq end
ldi output,$4c
cpi tp,02
breq end
ldi output,$5f
cpi tp,03
breq end
end
ret

```

为了访问表中的 2 号元素可以这样做

```

ldi tp 2
rcall table

```

当从表格子程序返回时“output”将包括“\$4c”

问：在没有 ADDI 指令的情况下怎样将一个常数立即数和寄存器的内容相加？

答：一个较简单的方法是减去一个负数下面的代码是将寄存器 R20 中和 5 的相加的例子

```
subi r20 -5
```

注意无论如何在这种情况下进位标志位不能用于检测溢出如果你已经将常数放入另一个寄存器你可以使用通用的 ADD 指令如果常量为 1 你能使用 INC 指令来增加寄存器的值 16 位的加法是同样的下列代码是将 r17 r16 和 \$0b3c 相加的例子

```
subi r16 low(-$0b3c) 减去低位字节
sbci r17 high(-$0b3c) 减去高位字节

```

问：为实现指令的单周期运行是否对外部时钟倍频？

答：没有这个时钟是 MCU 真正使用的没有使用时钟分频或倍频电路

问：AVR 的一个指令周期使用了多少个时钟周期？

答：只有一个时钟周期例如在寄存器中加入两个数全部执行时间等于在 XTAL1 引脚上的时钟信号正半周加负半周的时间

问：为什么 AT90S8515 称 8K 的 MCU？

答：大多数 CISC 微控制器的指令是变长度的一些指令由单字节组成其它的指令也可以是四个字节的长度大多数 AVR 指令是两个字节长度只有很少也是四个字节的所以 CPU 在启动运行时不能每次只读取和解释一个字节而我们的代码尺寸基准始终是和字节相比较的因此在 80C51 上填满 8K 的一个 C 程序肯定也适合的 8K 的 AVR

问：我从哪里能找到数据簿？

答：你可以从网站 www.atmel.com 有选择地下载它们你也可从当地分销商和 ATMEL 办事处得到数据簿你还可以发电子邮件到 literature@atmel.com 索取

问：AVR 有哪些超过我现有 MCU 的附加性能？

答：十分紧凑的代码，特别是对 C 程序 4-10 倍的高速度可编程 FLASH 存储器可在系统编程芯片内含 EEPROM 存储器可在系统编程低功耗。

1. 欢迎浏览得技通公司网站 www.8051faq.com.cn

2. 文章自网络转载，版权属原作者。