复习范围（题型：选择、简答、编程、扩展设计）

51是哈弗结构的，ram,rom分开编址

1. AT89C51的基本参数（CPU，RAM, ROM等）和内部的组成结构。

（1）8位微处理器（CPU）；

（2）数据存储器（128B RAM）+128B 特殊功能寄存器；

（3）程序存储器（4KB Flash ROM）；

（4）4个8位并行I/O口（P0口~P3口）；

（5）1个全双工异步串口；

（6）2个16位定时器/计数器；

（7）1个看门狗定时器；

（8）中断系统：5个中断源、5个中断向量；

（9）特殊功能寄存器（SFR）26个(答案)；

（10）低功耗的空闲模式和掉电模式；

（11）在线可编程功能ISP（In System Program）；

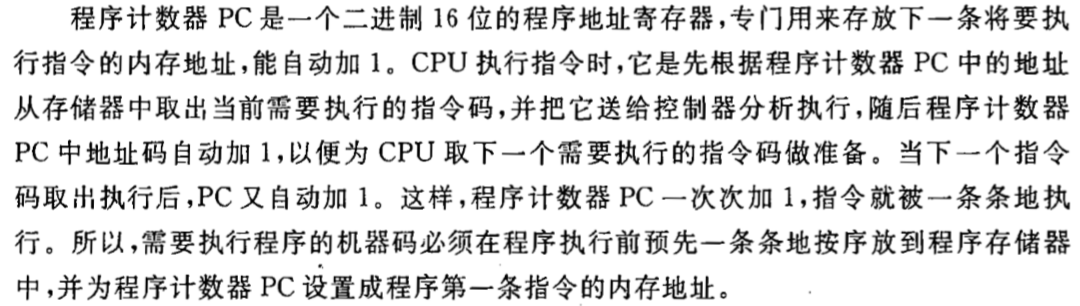
（12）数据指针2个，方便对片外RAM的访问。

（13）12MHz or 11.0592MHz

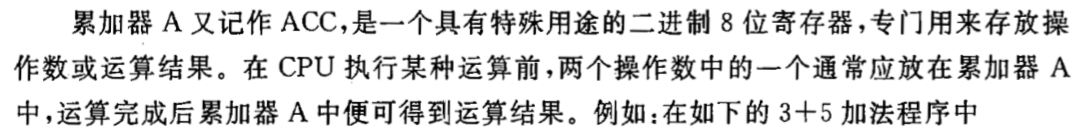
各功能部件通过片内单一总线连接而成，基本结构是CPU+外围芯片的传统微机结构。CPU对各种功能部件控制采用特殊功能寄存器的集中控制方式。

1. 重要的特殊寄存器的作用(PC, SP, DPTR,累加器A等)，复位后各重要寄存器的状态？PC/SP/P0~P3

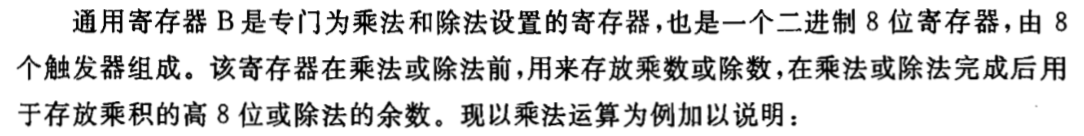
(1).程序计数器PC 复位后0000H



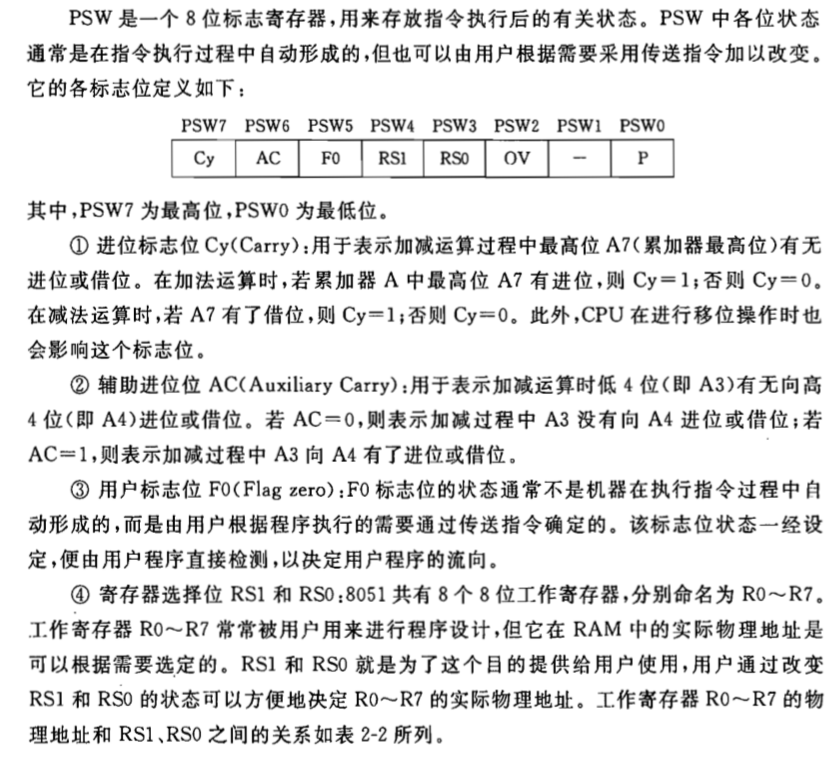
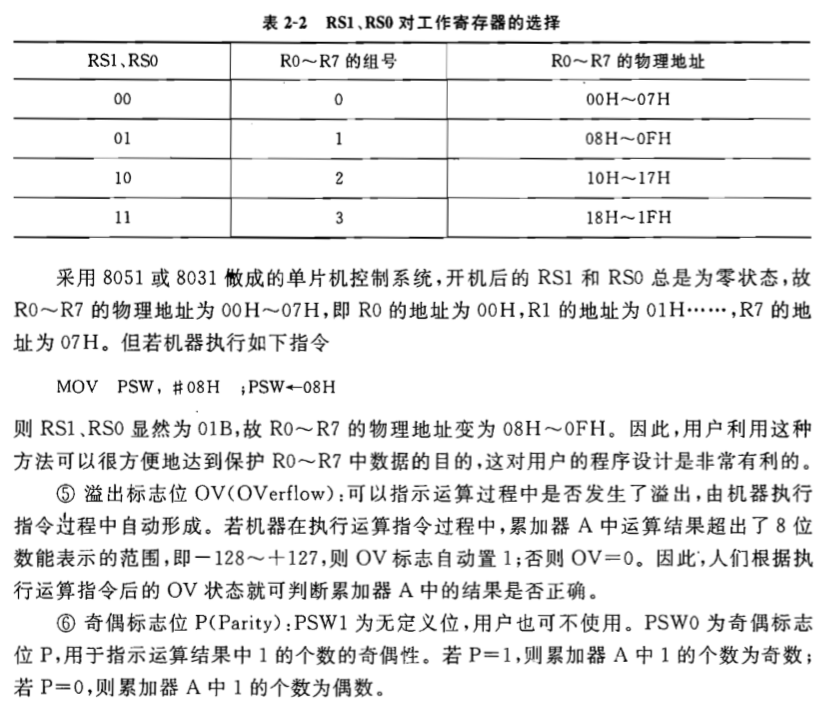
(2).累加器A=00H



(3).通用寄存器B=00H



(4).程序状态字PSW=00H

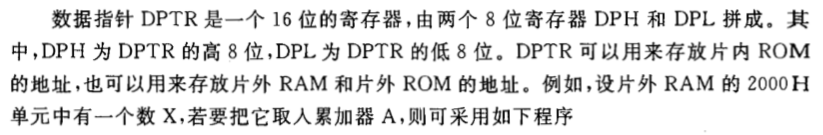
 

(5).堆栈指针SP 复位后07H#for r0~r7 occupied 00H~07H



即指示堆栈栈顶在内部RAM中的位置

(6).数据指针DPTR=0000



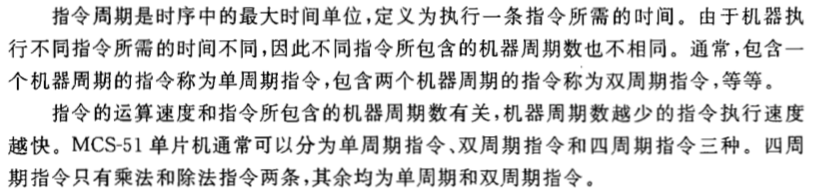
(7).P0-P3 复位后FFH

1. 单片机内部时序计算（机器周期、振荡周期\时钟周期、状态周期）

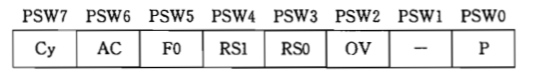
（1）机器周期：CPU完成一个基本操作所需时间为机器周期。执行一条指令分为几个机器周期。每个机器周期完成一个基本操作，如取指令、读或写数据等。每12个时钟周期为1个机器周期。

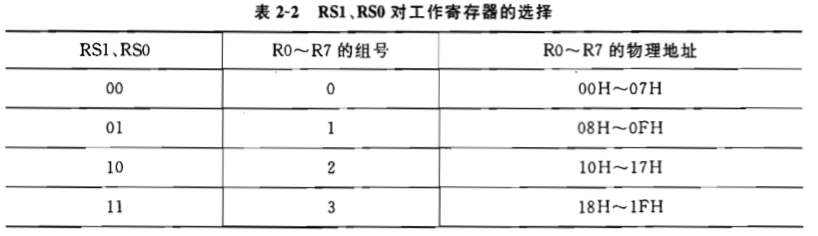
（2）时钟周期，状态周期：1个机器周期包括12个时钟周期（也称振荡周期），分6个状态：S1～S6。每个状态又分两拍：P1和P2。因此，一个机器周期中的12个时钟周期表示为S1P1、S1P2、S2P1、S2P2、…、S6P2

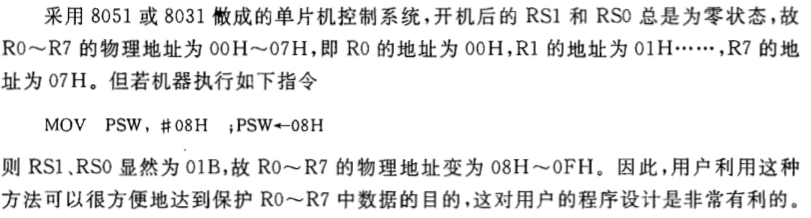
（3）指令周期：



1. 通用工作寄存器R0~R7(0-3组)的选择方法？（RS0和RS1的设置）







1. MCS51单片机五大寻址的逻辑地址范围？（内部数据存储器，程序存储器，外部数据存储器，特殊功能寄存器，位地址空间）

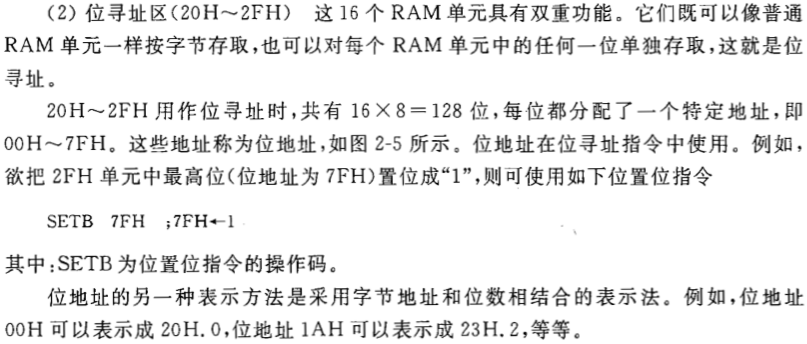
内部数据存储器：00H-FFH;实则00H-7FH

程序存储器：0000H-FFFFH;实则0000H-0FFFH

外部数据存储器：0000H-FFFFH

特殊功能寄存器：80H-FFH,需要直接寻址访问,(看似与ram的物理地址相同)实则与ram的物理地址不同,寄存器地址可以被8整除的才可以位寻址.

位地址空间：20H-2FH



(注：RAM是数据存储器，ROM是程序存储器)

1. MCS-51支持位寻址的存储空间有哪些？。 ram中20H~2FH,寄存器地址可以被8整除的才可以位寻址
2. I/O口基本性质。

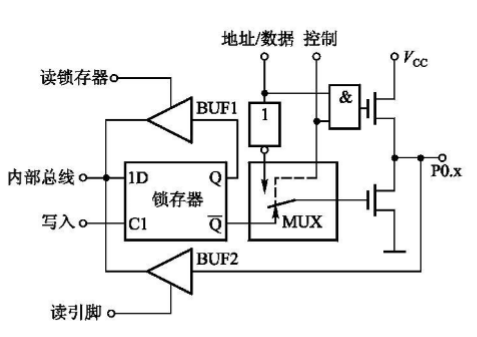
P0：在CPU访问片外存储器时先是用于传送片外存储器的低8位地址，然后传送CPU对片外存储器的读写数据（数据/地址总线，数据/地址复用口）

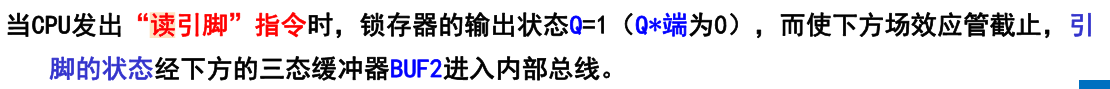
P2: 高8位

1。p0口灌电流12ma，其余口最大灌电流6ma

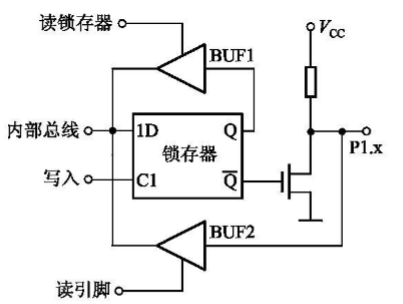
2。p0-p3口初始化均为正

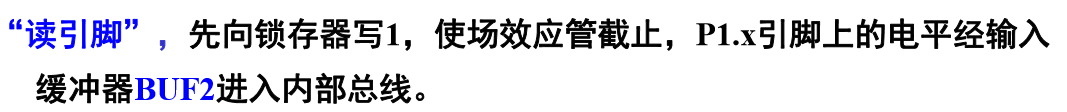
1. 四个I/O口P0/P1/P2/P3在片外扩展总线中的作用。I/0口读引脚的方法。
   * 都是向锁存器写1，使场效应管截至，在BUF2读取（p3也可以BUF3）
   * P3的功能复用较多
   * 作用：P0：在CPU访问片外存储器时先是用于传送片外存储器的低8位地址，然后传送CPU对片外存储器的读写数据（数据/地址总线，数据/地址复用口），先传输地址后传输数据。



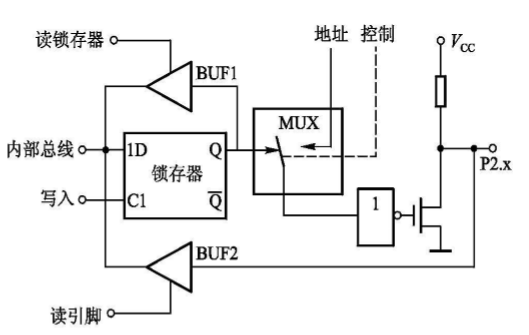


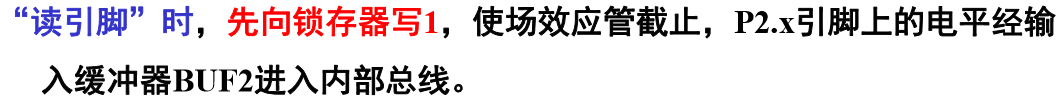
P1：只作为通用I/O口



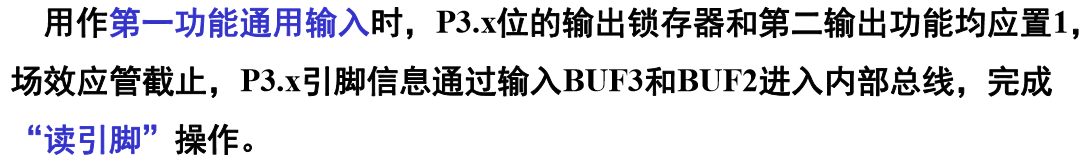
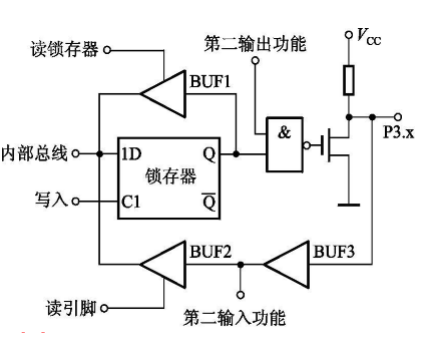


* + P2：配合P1传送片内EPROM12位地址中的高4位地址（地址总线）





* + P3：控制总线





1. 掌握单片机汇编指令（MOV, MOVX, MOVC, 跳转类JZ,JC, 运算类，CJNE, DJNZ）：
   * 命令+操作码 + 操作数
   * MOV :common operation
   * MOVX: operation used on outter ram
   * MOVC used on ROM 转移方向都是由右到左
   * JZ:累加器为0跳转
   * JC:C为1跳转
   * CJNE：ram单元与立即数不等转移
   * DJNZ：直接寻址单元或寄存器减一不为0转移
2. 汇编指令的七种寻址方式。（能举例，会区分）：
   * 寻址指的单个数据，而一个操作符可以包含多种寻址方式，一般讨论的后面一个
   * 立即寻址

直接给出运算所用的立即数：eg：MOV A,#30;

* + 直接寻址

直接给出地址，只可以作用于内部ram,特殊功能寄存器，位地址空间，其中特殊功能寄存器，位地址空间只可以直接寻址：eg:ANL 70H，#30；把70H中的数与操作30，并保存于70H

* + 间接寻址

间接寻址在r0或r1面前加上@进行寻址，eg：MOV A，@r0

* + 寄存器寻址

对r0-r7,累加器A，通用寄存器B，地址寄存器DPTR，进位C中的数进行操作，是一种隐含寻址方式，eg：INC r0；++r0

* + 相对寻址

用于转移指令中PC中基址加偏移量的寻址方式，偏移量+127～-128，eg：

JC 80H;若果c=1，则pc加80H作为偏移量

* + 变址寻址

变址寄存器加上基址寄存器，基址寄存器有pc与dptr，eg:

MOVC A,@A+DPTR

* + 位寻址

是对内部ram与特殊功能寄存器中的位寻址空间进行位寻址。借助进位寄存器C作为位操作累加器，指令操作数直接给出位地址，根据操作码的性质进行位操作。位操作与字节操作完全相同，主要由操作码进行区分。

Eg:MOV C,20H

1. 访问程序存储器、片外数据存储器、内部数据存储器、特殊功能寄存器各用什么指令？
   * 程序存储器：

MOVC

* + 片外数据存储器：

MOVX

* + 内部数据存储器：

MOV ,直接寻址，位寻址，

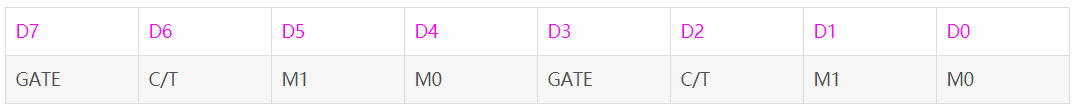
* + 特殊功能寄存器：

直接寻址，位寻址

1. 定时器的相关控制寄存器TMOD, TCON的比特位定义？定时器有哪几种工作方式，怎么配置？
   * 当定时器1作为波特率发生器时，定时器1可以作为两个8位定时器用

，用TMOD中控制位C/T决定作为定时器还是计数器，核心部件是一个加法器，本质都是对脉冲进行计数，只是脉冲源来源不同。当为定时模式时，系统时钟每6个或12个脉冲计数一次

* + TMOD

复位值：00H

只可以字节寻址

D0~D3为T0定时/计数器的设置，D4~D7为T1定时/计数器的设置 。

//GATE ：为门控位，GATE=0时，只要在编写程序时，使TCON中的TRO或TR1为1，就可以启动定时器/计数器工作。GATE=1时，不仅要在编写程序时，使TCON中的TRO或TR1为1，且需要外部引脚也为高电平，才能 工作。

C/T ：定时/计数模式切换，C/T=0时为定时模式，C/T=1时为计数模式。

M1，M0：用来选择定时计/计数器的工作方式，一般使用都是采用16位的计时计数器。



* + TCON

T0,T1控制寄存器

作用是控制定时器的启、停，标志定时器溢出和中断情况

可位寻址



//TF1 ：TF1=1表示T1有中断产生。（Timer Flag，定时器标志位）

//TR1 ：TR1=1表示T1开始运行。（单片机中T0引脚，需要高低电平的驱动）

TF0 ：TF0=1表示T0有中断产生。

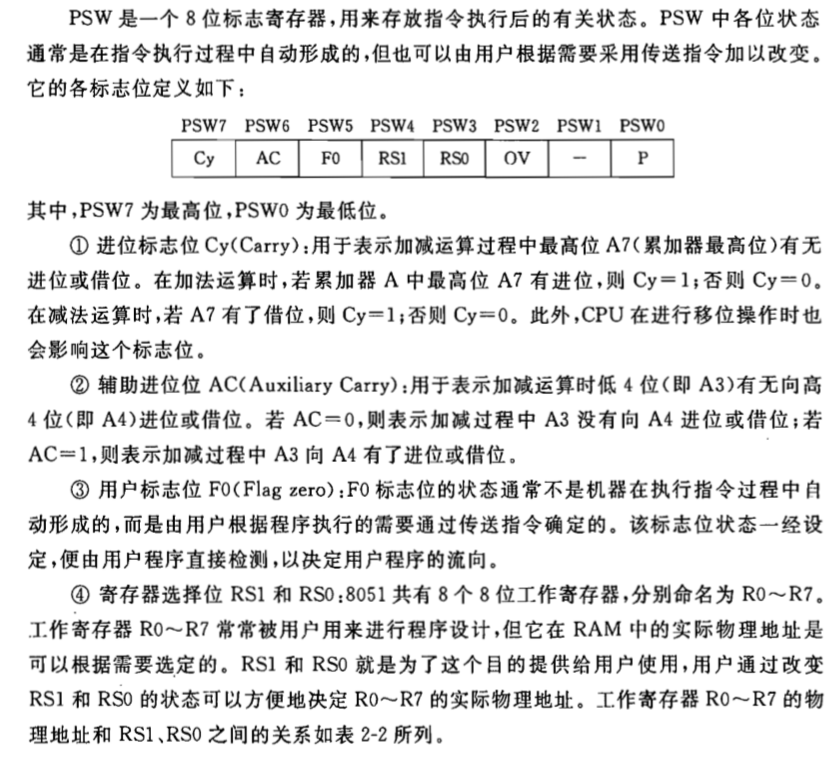
//TR0 ：TR0=1表示T0开始运行。（单片机中T1引脚，需要高低电平的驱动）

//IE1 ：IE1=1表示INT1有中断产生。

IT1 ：IT1=1表示INT1为下降沿触发，IT1=0表示INT1为低电平触发。

IE0 ：IE0=1表示INT0有中断产生。

IT0 ：IT0=1表示INT0为下降沿（负跳变）触发，IT0=0表示INT0为低电平触发。

1. 程序状态字PSW的每个比特位的定义。重点是加法乘法指令ADD/ADDC对标志位的影响
   * 可位寻址
   * ADD 两数相加，不加进位位。
   * ADDC 两数相加，加进位位。进位当时为1就加1，进位为0就加0。
   * 一般在多字节数相加时，低字节相加用ADD ，高字节相加就用ADDC。
   * 实现低字节产生的进位加到高字节相加当中，最终完成多字节数的相加。
   * 比如例一：
   * 0290H + 0190H
   * 低字节相加用 ADD
   * 90H+90H=120H 用ADD指令得到20H，并溢出产生进位 C=1
   * 高字节相加用 ADDC
   * 02H+01H=03H 用ADDC指令两数相加结果03H会再加上进位位1，得到 04H
   * 0290H+0190H=0420H
   * 
2. 中断控制寄存器IE/IP的比特位定义、5种中断源入口地址表/中断标志位
   * C51有5个中断

0, 外部中断0, IE0(P3.2)

1, 定时器0溢出中断, TF0

2, 外部中断1, IE1(P3.3)

3, 定时器1溢出中断, TF1

4, 串行口中断, RI

* + 0003H->外部中断0入口地址，之后每间隔8bytes一个中断入口
  + 

IE中断允许控制寄存器（可位寻址）

EX0：外部中断0允许位；  
ET0：定时/计数器T0中断允许位；  
EX1：外部中断1允许位；  
ET1：定时/计数器T1中断允许位；  
ES ：串行口中断允许位；  
EA ：CPU中断允许（总允许）位。

* + IP中断优先级寄存器

可位寻址

c51只有两级优先



PX0：外部中断0优先级设定位  
PT0：定时/计数器T0优先级设定位  
PX1：外部中断1优先级设定位  
PT1：定时/计数器T1优先级设定位  
PS ：串行口优先级设定位

* + 中断响应条件

中断请求+中断允许位为1+中断总允许位为1

* + 中断标志位有scon与tcon

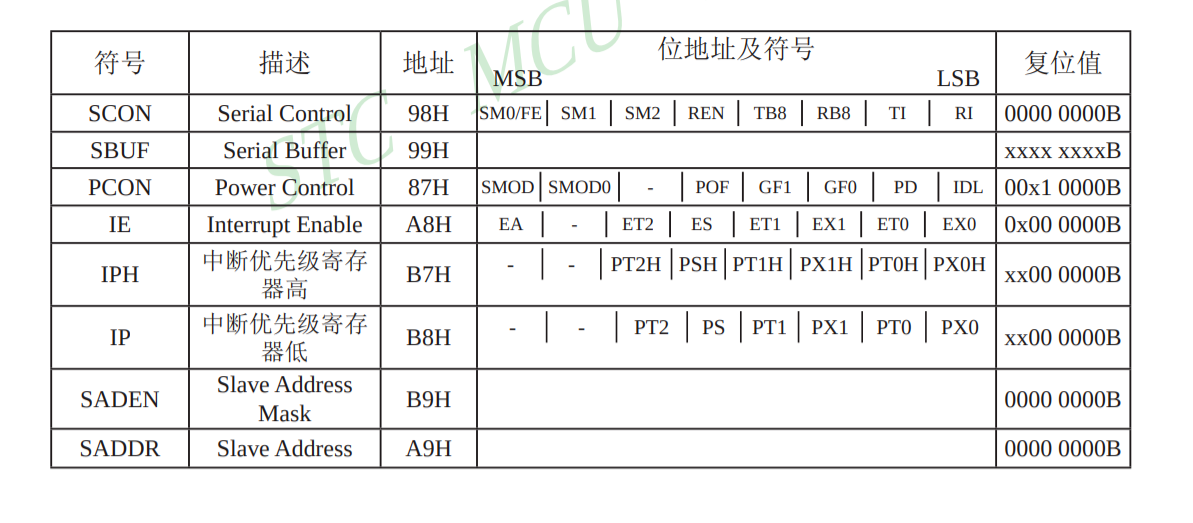
TCON参照上面的，SCON如下：

SCON可位寻址



RI：串行口接收中断标志位。当允许串行口接收数据时，每接收完一个串行帧，由硬件置位RI。注意，RI必须由软件清除。

TI：串行口发送中断标志位。当CPU将一个发送数据写入串行口发送缓冲器时，就启动了发送过程。每发送完一个串行帧，由硬件置位TI。CPU响应中断时，不能自动清除TI，TI必须由软件清除。

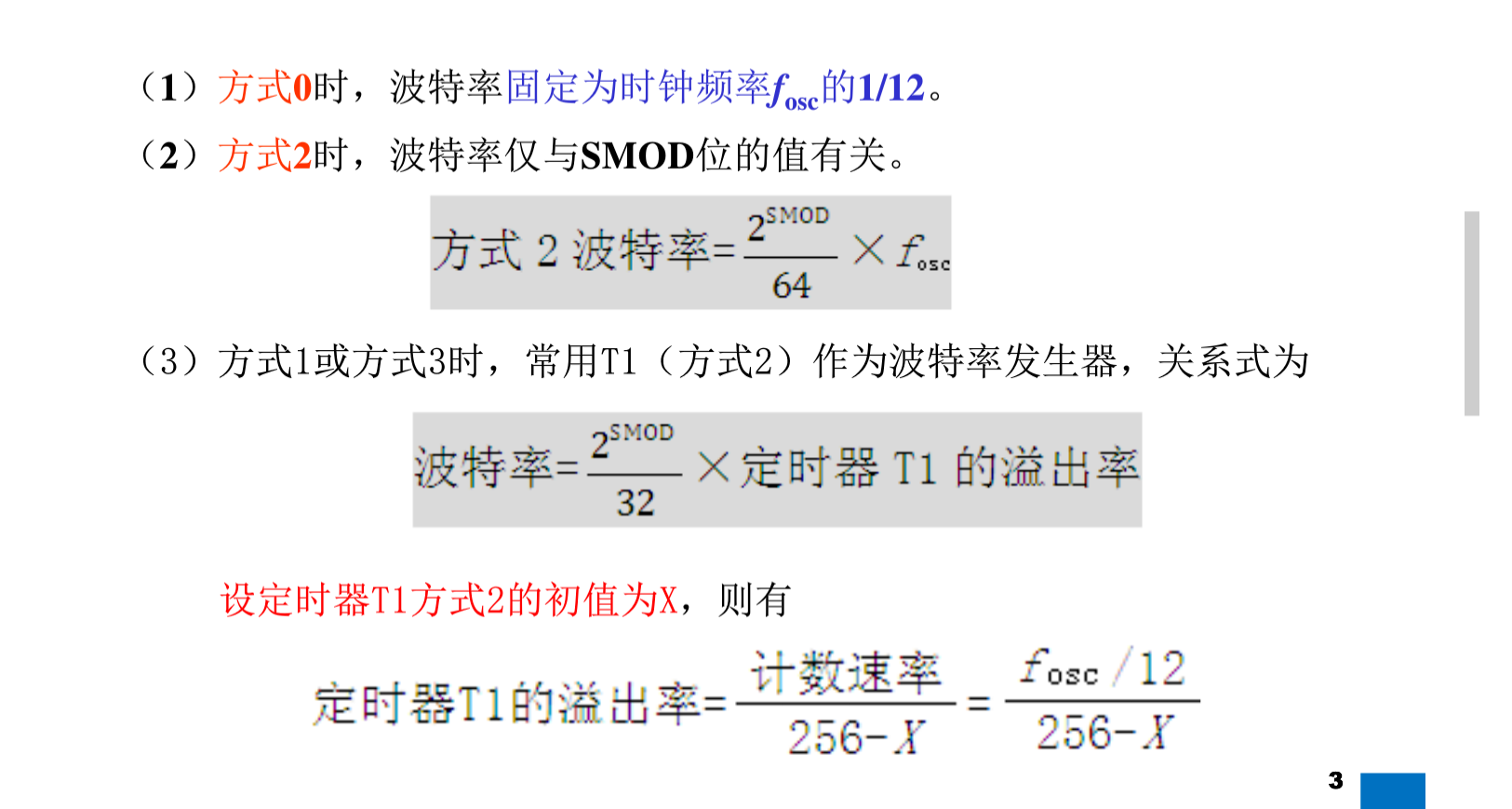
1. 5种中断源的自然优先级
   * 优先级参考上面例举的顺序
2. 何谓同步通信、异步通信？两种串口异步通信的数据帧格式
   * 同步通信指的有同步时钟信号的通信
   * 8051中的串口是全双工的
   * 有两个缓冲器，发送缓冲器只可写与接受缓冲器只可读，因此共用地址99H，统一称作SBUF
   * P3.0RX,P3.1TX,波特率由计数器给出
   * 帧格式：
   * 数据帧格式：同步：即检测到规定的同步字符（1~2个）后，下面就连续按顺序传送数据，直到一块数据传送完毕。字符之间不需要间隙，也不要起始位和停止位。
   * 异步：一个数据帧包括：1个起始位（低电平），8个数据位，1个校验位，1个停止位（高电平，表示数据帧结束）
3. 串口方式控制字SCON比特位定义; 根据波特率参数，对串口初始化计算和编程
   * SCON(串口控制寄存器)
   * 
   * M0、SM1, 工作方式, 功能描述, 波特率

0 0, 方式0, 8位移位寄存器, Fosc/12

0 1, 方式1, 10位UART, 可变

1 0, 方式2, 11位UART, Fosc/64或fosc/32

1 1, 方式3, 11位UART, 可变（ fosc ）

* + SM2：多机通讯控制位。在方式0时，SM2一定要等于0。在方式1中，当（SM2）=1则只有接收到有效停止位时，RI才置1。在方式2或方式3当（SM2）=1且接收到的第九位数据RB8=0时，RI才置1。
  + REN：接收允许控制位。由软件置位以允许接收，又由软件清0来禁止接收。
  + TB8: 是要发送数据的第9位。在方式2或方式3中，要发送的第9位数据，根据需要由软件置1或清0。例如，可约定作为奇偶校验位，或在多机通讯中作为区别地址帧或数据帧的标志位。
  + RB8：接收到的数据的第9位。在方式0中不使用RB8。在方式1中，若（SM2）=0，RB8为接收到的停止位。在方式2或方式3中，RB8为接收到的第9位数据。
  + TI：发送中断标志。在方式0中，第8位发送结束时，由硬件置位。在其它方式的发送停止位前，由硬件置位。TI置位既表示一帧信息发送结束，同时也是申请中断，可根据需要，用软件查询的方法获得数据已发送完毕的信息，或用中断的方式来发送下一个数据。TI必须用软件清0。
  + RI：接收中断标志位。在方式0，当接收完第8位数据后，由硬件置位。在其它方式中，在接收到停止位的中间时刻由硬件置位（例外情况见于SM2的说明）。RI置位表示一帧数据接收完毕，可用查询的方法获知或者用中断的方法获知。RI也必须用软件清0。
  + 1，方式0为移位寄存器输入/输出方式。可外接移位寄存器以扩展I/O口，也可以外接同步输入/输出设备。8位串行数据者是从RXD输入或输出，TXD用来输出同步脉冲。  
    （1）输出串行数据从RXD引脚输出，TXD引脚输出移位脉冲。CPU将数据写入发送寄存器时，立即启动发送，将8位数据以fos/12的固定波特率从RXD输出，低位在前，高位在后。发送完一帧数据后，发送中断标志TI由硬件置位。  
    （2）输入当串行口以方式0接收时，先置位允许接收控制位REN。此时，RXD为串行数据输入端，TXD仍为同步脉冲移位输出端。当（RI）=0和（REN）=1同时满足时，开始接收。当接收到第8位数据时，将数据移入接收寄存器，并由硬件置位RI。
  + 2，方式1为波特率可变的10位异步通讯接口方式。发送或接收一帧信息，包括1个起始位0，8个数据位和1个停止位1。  
    （1）输出当CPU执行一条指令将数据写入发送缓冲SBUF时，就启动发送。串行数据从TXD引脚输出，发送完一帧数据后，就由硬件置位TI。  
    （2）输入在（REN）=1时，串行口采样RXD引脚，当采样到1至0的跳变时，确认是开始位0，就开始接收一帧数据。只有当（RI）=0且停止位为1或者（SM2）=0时，停止位才进入RB8，8位数据才能进入接收寄存器，并由硬件置位中断标志RI；否则信息丢失。所以在方式1接收时，应先用软件清零RI和SM2标志。
  + 3，方式2  
    方式2为固定波特率的11位UART方式。它比方式1增加了一位可程控为1或0的第9位数据。  
    （1）输出: 发送的串行数据由TXD端输出一帧信息为11位，附加的第9位来自SCON寄存器的TB8位，用软件置位或复位。它可作为多机通讯中地址/数据信息的标志位，也可以作为数据的奇偶校验位。当CPU执行一条数据写入SUBF的指令时，就启动发送器发送。发送一帧信息后，置位中断标志TI。  
    （2）输入: 在（REN）=1时，串行口采样RXD引脚，当采样到1至0的跳变时，确认是开始位0，就开始接收一帧数据。在接收到附加的第9位数据后，当（RI）=0或者（SM2）=0时，第9位数据才进入RB8，8位数据才能进入接收寄存器，并由硬件置位中断标志RI；否则信息丢失。且不置位RI。再过一位时间后，不管上述条件时否满足，接收电路即行复位，并重新检测RXD上从1到0的跳变。
  + 4，工作方式3  
    方式3为波特率可变的11位UART方式。除波特率外，其余与方式2相同。
  + 

1. 存储器芯片容量与地址线数量的关系？比如2732是什么芯片，容量是多少？6264是什么芯片，容量又是多少？
   * 储存芯片容量<=2^(地址线线数)\*(数据线线数)/8 （KB）
   * 后两位表示总存储量：4Kx8bits=4KB
   * 后两位表示总存储量：8Kx8bits=8KB
2. 单片机外部并行扩展的基本方法（线选法等）的概念。能用线选法扩展数据存储器和确定其寻址地址。
   * 单片机

编程题的范围

编程题的范围

19、 能编写对内存单元中的数进行奇数和偶数判断的汇编程序，算法原理是除以2后判断余数是否为0.

ORG 0100H

MOV A, #00H; dividend

MOV B, #2;divisor

DIV AB

MOV A, B;B(remainder)，A(quotient）

JZ even\_num

SETB P1.0

even\_num:

CLR P1.0

END

20、 能编写两数进行加减乘除的汇编程序。

ORG 0100H

MOV A, #00H

MOV B, #00H

ADD A,B;stored A

SUBB A,B; stored A

MUL AB; stored H:B,L:A

DIV AB; stored A

21、 能编写通过定时器中断在I/O引脚输出方波的完整汇编程序。（定时器和中断初始化）

ORG 0000H

LJMP MAIN

ORG 000BH

CPL P1.0

RETI

ORG 0100H

MAIN: MOV SP, #60H;设置堆栈指针

MOV TMOD, #02H；设置定时器T0为方式2

MOV TL0, #206;设置计数初值，计50次后溢出，正好50微秒

MOV TH0,#206

SETB TR0;启动定时器

SETB EA;开中断开关

SETB ET0;开定时器T0中断

SJMP $;原地循环

END

22、 能编写通过CJNE和JC的指令组合，编写比较两个数大小的汇编程序

MOV A,#00H

MOV B,#00H

SUBB AB

JC, LOOP(A<B)

CJNE Cy,#00H,START(A>B)

SETB P1.0

SETB P1.1

LOOP:

CLR P1.1(A<B)

START:

CLR P1.0(A>B)

END,

23、 能编写通过串口发送数据的程序。

ORG 0000H

LJMP MAIN

ORG 0023H

LJMP SENDSTR

ORG 0100H

MAIN: MOV SP, #60H;设置堆栈指针

MOV TMOD ,#20H

MOV TL1,#0F4H

MOV TH1,#0F4H

SETB TR1

MOV SCON, #40H;串口通信方式1

MOV PCON,#80H

SETB EA;开中断开关

SETB ES；开串口中断

MOV R7,#9

MOV SBUF,#30H

SJMP $；原地循环

ORG 0200H

SENDSTR: CLR TI

MOV SBUF, #30H

DJNZ R7, EXIT

CLR ES

EXIT: RETI

END

24、 能编写子程序、循环结构体程序。

DELAY:

MOV R1,#255

LOOP:

MOV R2,#255

DJNZ R2,$

DJNZ R1,LOOP

RET

END

(LCALL DELAY)

25、 能编写延时子程序。

ORG 0000H

TIME EQU 60000;us

LJMP START

ORG 000BH;timer1 interrupt address

MOV TH0,#(65536-TIME)/256 ;set time

MOV TL0,#(65536-TIME)MOD 256 ;set time

LJMP T0\_INT

START:

SETB C

MOV P1,#0FEH

MOV P2,#0FFH

MOV R0,#0

MOV TMOD,#01H;set timer0 work mode

MOV TH0,#(65536-TIME)/256 ;set time

MOV TL0,#(65536-TIME)MOD 256 ;set time

MOV IE,#10000010B;enable all interrupt and timer0 interrupt

SETB TR0;start counter0

MOV R2,#3

SJMP $ //wait

T0\_INT:

MOV A,P1

RLC A

MOV P1,A

MOV A,P2

RLC A

MOV P2,A

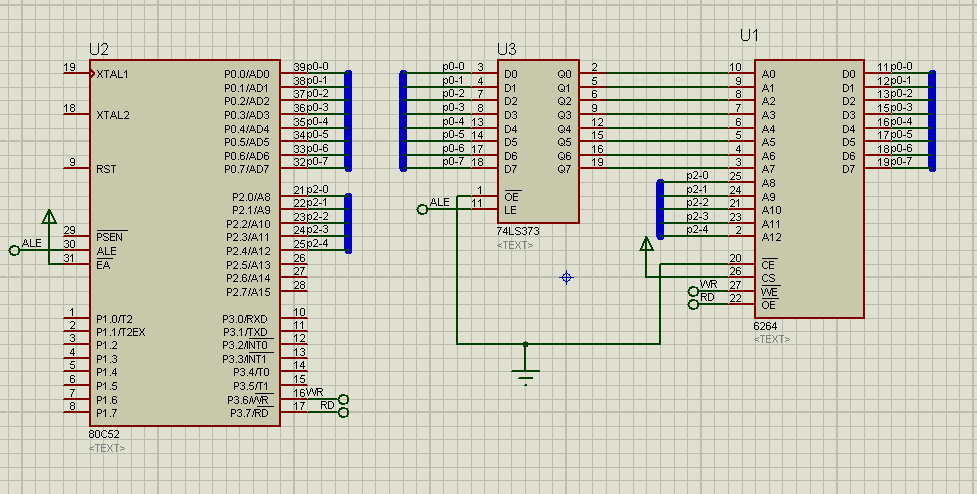
RETI //RETURN

END

设计题范围

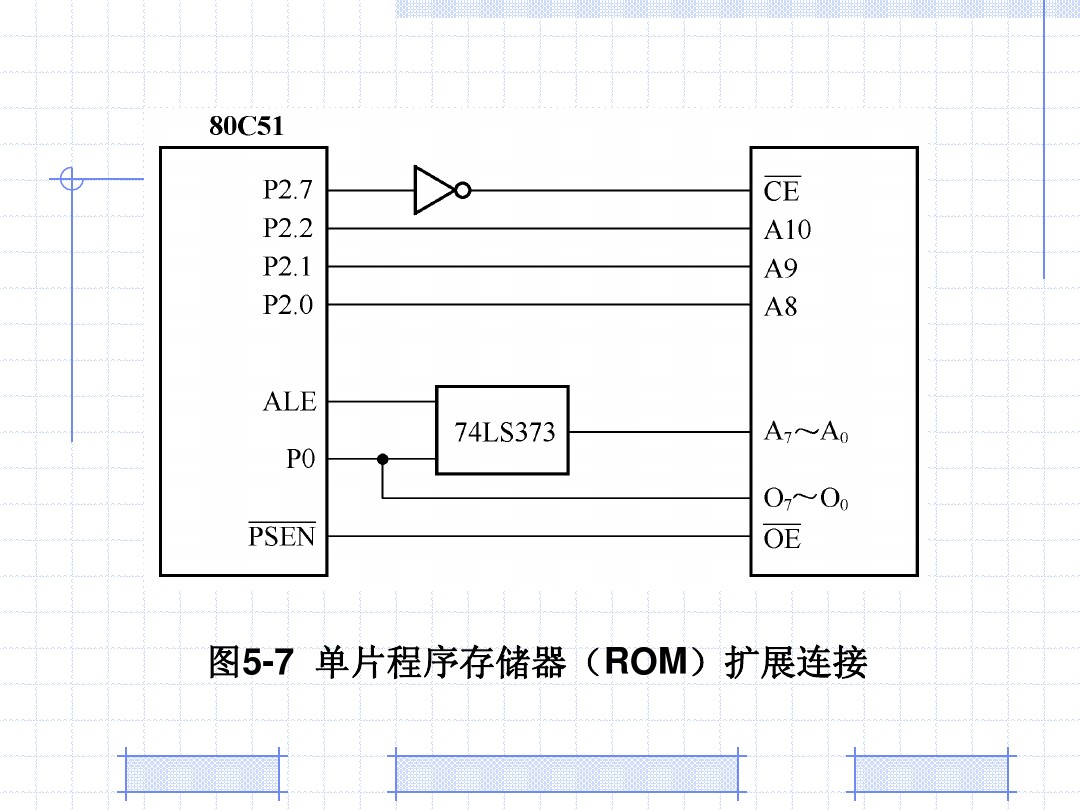
1. 能画出对单片机扩展程序存储器、数据存储器的电路逻辑框图。能准确连接扩展中的对应控制引脚。能依据自己画得电路图，确定扩展芯片的寻址范围。

**RAM拓展**

* + MOVX A,@DPTR //读外扩ram，产生读时序，由WR/p3.6产生
  + MOVX @DPTR,A //写外扩ram，产生写时序，由RD/p3.7产生
  + 我们发出一条MOVX指令的时候，先发出16位地址，产生控制时序，然后在写（读）数据，这个时候，我们低8位地址总线就与数据总线冲突了（因为都是用的P0口），使用如果我们要时候低8位地址去限定的时候，就可以就一个锁存器，来锁存低8位地址（这个时候要用到锁存信号引脚ALE），再比如我们外围器件不多的时候，或者外扩ram不超过2^8=256B的时候，完全可以不管低8位
  + 
  + P0用锁存器74373
  + 拓展ram芯片选择6264，8KB
  + 寻址范围：0000H-1FFFH

**ROM拓展**

* + 8051程序存储器用于存放编好的程序和表格常数。程序存储器通过16位程序计数器PC寻址，寻址能力为64K字节。8051，8751的64程序存储器片内ROM为4K字节，地址为0000H—0FFFH，片外最多可扩至64K字节ROM，地址1000H—FFFFH,片内外是统一编址的。
  + 当引脚EA接高电平时，8051程序计数器PC在0000H—0FFFH范围内，即前4K字节地址执行片内ROM中的程序；当指令地址超过0FFFH后，就自动地转向片外ROM指令
  + 当引脚EA接低电平时，8051片内ROM不起作用，CPU只能从片外ROM中取指令，地址可以从000H开始编址。这种接法特别适用于采用8031单片机的场合，由于8031片内不带ROM，所以使用使必须使EA=0.以便能够从外部扩展EPROM中取指令。



CE：片选，A0～A10：地址，O0～O7：数据，ALE：锁存，OE：输出允许，地址范围：f000-f7ff