【微机结构】计算机三大基本单元：中央处理器 CPU、存储器、输入/输出设备。按照单片微机的类型不同，单片微机应用系统可分成总线方式、非总线方式。单片微机基本结构包括中央处理器、存储器、并行 I/O 口、串行 I/O 口、定时器/计数器、定时电路及元件等。

【数据指令·MOV】内部 8 位数据传送 A Rn direct @Ri

MOV 可用作唯一的一条 16 位指令，形如 MOV DPTR，#data16:；把数据高 8 位送入 DPH 寄存器，数据低 8 位送入 DPL 寄存器。

【数据指令·MOVX】外部数据传送指令 MOVX @Ri，A MOVX @DPTR，A

【数据指令·MOVC】程序存储器数据传送指令（查表指令）MOVC A,@A+PC 为

定时，这就是定时功能。

【定时器/计数器】80C51 有两个 16 位的，T0 和 T1。定时器/计数器的核心是加

1 计数器，基本功能是计数加 1，通过软件设定。对单片微机的 T0、T1 引脚上输入的一个 1 到 0 的跳变进行计数增 1 即是计数功能，对单片微机内部的机器周期进行计数，从而得到定时，这就是定时功能。

PL N 清 零 正 数 或 零

VS V 置 位 溢 出

VC V 清 零 未 溢 出

HI C 置位 Z 清零 无符号数大于

LS C 清零 Z 置位 无符号数小于或等于

【RAM】常把寄存器（如工作寄存器、特殊功能寄存器、堆栈等）在逻辑上划分

近程查表，MOVC A,@A+DPTR 为远程查表。

【定时器/计数器】T0，T1 的工作方式，T0 可选择 4 种工作方式而 T1 只有 3 种。 GE N 等于 V 带符号数大于或等于

在片内 RAM 空间中。

【芯片引脚】RST：复位输入信号，高电平有效。EA/VPP：片外程序存储器访问

【数据指令·XCH】将累加器 A 与源操作数的字节内容互换。

【数据指令·XCHD】XCHD A，@Ri 低 4 位互换，而高 4 位不变。

方式 0:13 位定时器/计数器，此时方式寄存器 TMOD 中 M1=0，M0=0 方式 1:16

位定时器/计数器，M1=0.M0=1 方式 2：定时常数自动重装载的 8 位定时器/计数

LT N 不等于 V 带符号数小于

GT Z 清零且 N 等于 V 带符号数大于

允许信号，低电平有效。ALE/PROG：地址锁存允许信号。PSEN：片外程序存储

【数据指令·SWAP】将累加器 A 的高、低半字节交换。XCHD 和 SWAP 主要用于

器（对定时控制特别有用，适合作为串行口波特率发生器） 方式 3：T1 不工作， LE Z 置位或 N 不等于 V 带符号数小于或等于

器读选通信号，低电平有效，有效时，程序存储器内容被送上 P0 口（数据总线）。 十六进制数或 BCD 码的数位交换。

将 T0 分为一个 8 位定时器/计数器和一个 8 位定时器，通常 T1 用作波特率发生

AL 忽略 无条件执行

【CPU】中央处理器包括控制器、运算器和工作寄存器及时序电路。单片微机的

【数据指令·堆栈】入栈指令 PUSH 先执行（SP）+1 再传送数据 ；出栈指令 POP： 器时，T0 选择方式 3，这样可增加一个定时器。

【寻址方式】

核心，主要完成运算和控制功能，增设了面向控制的处理功能，增强了实时性。

先传送数据再执行（SP）-1

【定时器/计数器】定时时间t = Tc ∗ (2𝐿 − TC) = 12 (2𝐿 − 𝑇𝐶),其中 t 为定时时间； 1.寄存器寻址

𝑓𝑜𝑠𝑐

【CPU】中央控制器是识别指令，并根据指令性质控制计算机各组成部件进行工

作的部件，与运算器一起构成中央处理器 功能：控制指令的读出、译码和执行，

【算术运算】助记符 8 种：ADD、ADDC、INC、DA、SUBB、DEC、MUL、DIV

【算术运算 ADD】在计算加法中中当和的第三位有进位时，将 AC 标志置位，否

Tc 为机器周期；fosc 为晶体振荡器频率；L 为计数器的长度；TC 为定时常数。方式 0 时，L=13，方式 1 时，L=16，方式 2 时 L=8（213 = 8192，216 = 65536）

2.立即数寻址

3.寄存器移位寻址

对指令的执行过程进行定时控制，并根据执行结果决定是否分支转移。 组成：程

序计数器 PC、数据指针 DPTR、指令寄存器 IR、指令译码器、条件转移逻辑电路

则为 0；当和的第 7 位有进位时，将 CY 标志置位，否则为 0；对于带符号数运算， 定时时间的倒数即为溢出率。

当和的第 7 位与第 6 位中有一位进位而另一位不产生进位时，溢出标志 OV 置位， 【定时器时间与哪些因素有关?定时器/计数器做计数器使用时,外界输入计数频率

5 种移位操作：

（1）LSL 逻辑左移； （2）LSR 逻辑右移

及定时控制逻辑电路。

【CPU PC】中央控制器中最基本的寄存器程序计数器 PC 将其中的数作为所取指令的地址输出给程序存储器（工作过程），是一个 16 位的计数器，可对 64KB 程序存储器进行寻址。其轨迹决定程序的流程，宽度决定程序存储器可直接寻址的范围。

否则为 0，OV=1 表示两个正数相加，和为负数，或两个负数相加而和为正数的

错误结果。P 奇偶校验位

【算术运算 ADDC】带进位加法指令 标志位计算同 ADD 两数相加时需加 CY

【算术运算 INC】INC 为加一指令 不影响各个标志位

【算术运算 DA】DA 为二-十进制调整指令，根据 A 的原始数值和 PSW 的状态，

最高值】

定时器/计数器做定时器用时,定时器的定时时间与系统的振荡频率^sc,计数器的长度(如 8 位、13 位或 16 位等)和定时初始值等有关。定时器/计数器做计数器用时, 通过引脚 T0(P3.4)和 Tl(P3.5)对外部信号进行计数,由于检测一个 1 到 O 的跳变需要两个机器周期,故计数脉冲频率不能高于振荡脉冲频率的 1/24。

（3）ASR 算术右移； （4）ROR 循环右移；

1. RRX 带扩展的循环右移。

例 ： MOV R0, R1, LSL #5; R0=R1 逻辑左移 5 位

4.寄存器间接寻址

5.基址寻址

【CPU DPTR】数据指针 DPTR（16 位特殊功能寄存器）功能：作为片外数据存储

决定是否对 A 进行加 06H 或 60H 或 66H 的操作。BCD 码的 0000~1001 代表 0~9， 【串行通信数据传送】数据传送共有两种方式：并行数据传送和串行数据传送。

6.多寄存器寻址

器或 I/O 寻址用的地址寄存器（ 间接寻址）， 专门用于访问片外数据存储器

（ERAM）。

【PC 与 DPTR 不同之处】⑴PC：程序存储器的地址；DPTR：数据存储器或 I/O

而 1010~1111 位无效码，相加结果大于 9 进入无效码区，其结果比正确结果小 6.

当累加器低 4 位大于 9 或半进位标志 AC=1，则进行低 4 位加 6 修正，当累加器

高 4 位大于 9 或进位标志 CY=1，则进行高 4 位加 6 修正，

按照信号传输的方向和同时性可分为单工方式（信号沿一个方向传送）、半双工方 式（双方均具有发送接收能力但不能同时）和全双工方式（可以两个方向同时传送）。

一条指令实现一组寄存器值的传送，连续的寄存器用“-”连接，否则用“,”分隔

例 ：LDMIA R0, {R1-R5};

功能：R1=[R0], R2=[R0+4], R3=[R0+8]，R4=[R0+12], R5=[R0+16]

的地址。PC 与 DPTR 都通过 P0 口和 P2 口输出，PC 的输出与 ALE 及 PSEN 有关； 【算术运算 SUBB】带借位减法指令，其功能是从 A 中减去指定变量和进位位 CY， 【并行与串行传送异同】并行传送：计算机内部数据传送都是并行，各位同时发

从以 R0 为起始地址的存储单元中取出 5 个字的数据送到 R1 到 R5 寄存器中。

DPTR 输出与 ALE、WR、RD 信号有关。⑵PC 为 16 位寄存器，不可访问。DPTR

为 16 位寄存器，也可作两个 8 位特殊功能寄存器，可访问。

【CPU】指令寄存器 IR:存放指令操作码的专用寄存器。指令译码器对指令进行译码，译码结果送定时控制逻辑电路。程序执行过程是在控制器的控制下，将指令从程序存储器中逐条取出，进行译码，然后由定时控制逻辑电路发相应的定时控制信号，控制指令的执行。

【运算器】组成:算术逻辑运算单元 ALU（实质：全加器）、累加器 A、暂存寄存器、B 寄存器、程序状态标志寄存器 PSW 以及 BCD 码运算修正电路等。

ALU 有两个输入：1)暂存器 1 的输入 2)暂存器 2 或累加器 ACC 的输入。两个输出:1)数据运算结果通过内部总线送回累加器；2)数据运算后产生的标志位输出至程序状态字 PSW

累加器 A 功能：存放操作数，暂存运算结果。单片机中使用最频繁的 8 位专用寄存器，容易产生“瓶颈”现象。

B 寄存器：乘/除法指令中用作 ALU 的一个输入。乘法的两个输入为 A、B，运算结果，A 中放积的低 8 位，B 中放积的高 8 位。除法中，被除数取自 A，除数取自 B，商数存放于 A，余数存放于 B

程序状态字 PSW：字节地址 D0H.8 位寄存器，内容是算术逻辑运算单元（ALU） 的输出。

* 1. P－奇偶标志位表示累加器 A 中值为 1 的个数的奇偶性：若累加器值为 1 的位数是奇数，P 置位（奇校验）；否则 P 清除(偶校验) 。如(A)=00001010，则 P=0。在串行通信中，常以传送奇偶校验位来检验传输数据的可靠性。
  2. OV－溢出标志位指示运算结果是否溢出。OV=1：运算结果超出了寄存器 A 所能表示的带符号数的范围（一 128～＋127）。
  3. RS1、RS0 －工作寄存器组选择位
  4. AC －辅助进位标志位加减法运算时，低 4 位向高 4 位数进位或借位时，AC 将

结果存入 A 中。当第三位有借位时，将 AC 标志置位，否则为 0；当第 7 位有借

位时，将 CY 标志置位，否则为 0；当第 7 位与第 6 位中有一位借位而另一位不借位时，溢出标志 OV 置位，否则为 0，OV=1 表示正数减负数结果为负数，或负数减正数结果为正数的错误结果。OV 位用于带符号的整数减法。

【算术运算 DEC】DEC 为减一指令

【算术运算乘除】MUL AB 为乘法，运算结果低 8 位存于 A，高 8 位存于 B。DIV AB 为除法，运算结果商的二进制数部分存于 A，余数存于 B。

【逻辑运算】助记符 9 种：ANL、ORL、XRL、RL、RLC、RR、RRC、CPL、CLR 逻辑“与”助记符为 ANL，用符号“∧”表示，数和源地址的数按位相与

逻辑“或”助记符为 ORL，用符号“∨”表示，数和源地址的数按位相或

逻辑“异或”助记符为 XRL，用符号“⊕”表示，对应位数字相同为 0，不同为 1.

循环右移指令：RR A（累加器内容逐位循环右移一位，a0 移到 a7，不影响标志位）

带进位循环右移指令：RRC A（将累加器内容和进位位一起循环右移，a0 移入

CY，CY 移到 a7）只影响 CY 位

循环左移指令：RL A（累加器的内容逐位循环左移一位，a7 移到 a0。不影响标志位）

带进位循环左移指令：RLC A（累加器的内容和进位位一起循环左移一位，

a7 移入进位位 CY，CY 的内容移到 a0）

累加器按位取反指令：CPL A（累加器的内容逐位取反，结果仍存在 A 中。不影响标志位）

累加器清 0 指令：CLR A（对累加器进行清 0，不影响标志位）

【控制程序转移类指令】助记符 12 种：AJMP、LJMP、SJMP、JZ、JNZ、CJNE、

DJNZ、ACALL、LCALL、RET、RETI、NOP

短转移指令：SJMP rel （目标地址是由当前 PC 值即 PC 源地址值加 2 和（8 位

送，并排传输；物理通道：并行内总线或并行外总线；特点：速度快、效率高、传输距离小于 30m。

串行传送：计算机与外界数据大多是串行，各位按一定顺序一位接一位被发送和接收；物理通道为串行总线；特点为成本低、速度慢。

【串行通信同步技术】常用的同步技术有两种方式，异步传输（以字符为单位进行传输，每个字符都用起始位、停止位包装起来，在字符间允许有长短不一的间隙，单片微机中使用的都是异步方式），同步传输（对数据块进行传输）

【串行通信波特率与比特率】波特率表示每秒传输离散信号事件的个数，或每秒信号电平的变化次数，单位为波特。比特率是指每秒传送二进制数据的位数，单位为比特/秒。二进制下，波特率与比特率数值相等。

【串行通信主要问题】串行数据通信主要两个问题①数据传送（解决传送中的标准、数据帧格式及工作方式）②数据转换（解决把数据进行串、并行的转换，通常由通用异步接收、发送器电路完成）数据发送端，并行数据转换为串行数据， 数据接收端，串转并。

【串行口有几种工作方式以及特点】四种工作方式，由 SCON 中的 SM0 、SM1 进行定义：方式 0：同步移位寄存器的输入输出方式，主要用于扩展并行输入或输出口，波特率固定。方式 1：10 位数据的异步通信口，波特率可变。方式

2：11 位数据的异步通信口，波特率固定。方式 3：11 位数据的异步通信口， 波特率可变。

𝑓𝑜𝑠𝑐 2𝑆𝑀𝑂𝐷

【串行口波特率的计算】方式 0 时波特率为 fosc/12..方式 2 时波特率为 ∗

32 2

2𝑆𝑀𝑂𝐷

用定时器 T1 产生波特率时，波特率为 ∗ 定时器𝑇1 的溢出率。定时器 T1 工作

32

𝑓𝑜𝑠𝑐 1

于方式 0：此时 T1 相当于一个 13 位的计数器，溢出率= ∗ ，TC 为 13 位

12 213−𝑇𝐶

计数器初值。定时器 T1 工作于方式 1：此时定时器 T1 相当于 16 位的计数器。溢

出率= ∗ .定时器 T1 工作于方式 2 此时定时器 T1 工作于一个 8 位可重装

𝑓𝑜𝑠𝑐 1

7.堆栈寻址

8.块拷贝寻址

9.相对寻址

【ARM 指令分类】

1.数据处理指令

2.跳转指令：

B 转移指令

功能：跳转到目的地址。

跳转范围：当前指令的±32M 字节地址内（ARM 指令为字对齐，最低 2 位地址固定为 0）。

BL 带链接的转移指令

功能：PC 拷贝到链接寄存器，然后跳转到指定地址。跳转范围：当前指令的±32M 字节地址内。

BX 带状态切换的转移指令

功能：跳转到目标地址；处理器工作状态切换。

目标地址：寄存器 Rn 和 0xFFFFFFFE 相与的结果。Rn 的第 0 位拷贝到 CPSR 中 T

位，位[31∶1]移入 PC。

跳转范围：当前指令的±32M 字节地址内。

BLX 带链接和状态切换的转移指令

功能：跳转到目标地址；PC 值保存到 LR 寄存器；处理器工作状态切换。

3.Load/Store 指令

A.单寄存器的存取指令（LDR，STR）

LDR：从内存中读取单个字或字节数据存于寄存器；

LDR Rd，[Rn，Rm] ；将内存中的地址为 Rn+Rm 的字数据装入寄存器 Rd 中

LDR Rd，[Rn,Rm] ! ；将内存中的地址为 Rn+Rm 的字数据装入寄存器 Rd 中，并

被硬件置位；否则，被清除。在十进制调整指令 DA 中要用到 AC 标志位状态。

带符号）相对地址 rel 相加）

12 216−𝑇𝐶

𝑓𝑜𝑠𝑐 1

将新地址 Rn+Rm 写入 Rn

* 1. CY －进位标志位。在进行算术运算时，表示运算结果中高位是否有进位或借

绝对转移指令：AJMP addr11（目标地址由指令第 1 字节的高 3 位 a10～a8 和指

的方式用 TH1 计数，溢出率=

12

∗

28−𝑇𝐻1

LDR{cond}T <Rd>,<addr>;

位，可以被硬件置位或清除。

* 1. F0 －用户标志位。开机时该位为“0”。

用 户 可 根 据 需 要 ， 通 过 位 操 作 指 令 置 “l” 或 者 清 “0” 。 PSW 为

CY|AC|F0|RS1|RS0|OV|-|P

【时钟电路】组成：振荡器及定时控制元件、时钟发生器、地址锁存允许信号

ALE。

令第 2 字节的 a7～a0 所组成。以 11 位地址取代当前 PC 低 11 位，形成新的 PC

值。）

长转移指令：LJMP addr16（目标地址由指令第 2 字节和第 3 字节组成。目标地址为 64KB 空间）

间接转移指令：JMP @A+DPTR（目标地址是累加器 A 中的 8 位无符号数与数据指针 DPTR 的内容相加）

【系统扩展】主要有程序存储器的扩展、数据存储器的扩展、I/O 口的、中断系

统扩展。有并行扩展法和串行扩展法两种方法，并行扩展法是指利用单片微机本身具备的三组总线（AB\DB\CB）进行的系统扩展。串行扩展法是利用 SPI 和𝐼2C 双线总线等进行串行系统扩展。使用串行扩展法的优点：串行接口器件体积小， 占用电路板空间小，明显的减小了电路板空间和成本；串行接口器件与单片微机接口时需用的 I/O 线很少，减少了控制器的资源开销，简化了连接，进而提高了

T：若指令有 T，即使处理器是在特权模式下，也将存储系统访问看成是在用户模式的。

STRB R0,[R1]

STR：将寄存器 R0 中的低八位字节数据保存到 R1 表示的内存地址。

STRH R0,[R1]

STRH 将寄存器 R0 中的低 16 位数据存入 R1 表示的内存地址中

PD 置 1，振荡器停止工作，系统进入低功耗工作状态。ALE 以 1/6 振荡频率的固定速率输出。

累加器判零转移指令：JZ rel ；（ 若 (A)=0 ， 则 (PC)=(PC+2)+rel ； 若 可 靠 性 。

(A)≠0，则(PC)=(PC)+2）JNZ rel ；（ 若(A)≠0 ，则(PC)= (PC+2)+rel ；若 【并行扩展的编址技术】先片选，再线选。片选有四种方法：线选法、地址译码

STR{cond}H <Rd>,<addr>;

STRT:无论处于何种模式，该指令都将被当作一般用户模式下的内存操作。

时序定时单位: 节拍、状态、机器周期和指令周期。

(1)节拍 P:振荡脉冲的周期称为节拍。

(2)状态 S:一个状态 S 包含两个节拍。

(3)机器周期：宽度为 6 个状态，并依次表示为 S1～S6。

一个机器周期有 12 个振荡脉冲周期是单片机的最小时间单位。

(4)指令周期：执行一条指令所需要的时间。是最大的时序定时单位。80C51 的指

(A)=0，则(PC)=(PC)+2）

其中 rel=目标地址-PC 的当前地址

数值比较转移指令：CJNE A，direct，rel（指令格式为： CJNE（操作数 1），（操作数 2），rel）数值比较指令的第 1 字节为操作码(或操作码+操作数 1) ，第 2 字节为操作数 2，第 3 字节为偏移量 rel。具有比较转移和数值大小比较的功能。） 循环转移指令：DJNZ Rn，rel 或 DJNZ direct，rel。每执行一次本指令，

法、应用可编程 PAL/GAL 或 I/O 线法.

【系统扩展】AB（地址总线，16 位，可寻址 64KB），P0 口（使用最频繁的通道）提供低八位（必须要锁存器锁存，锁存控制信号来自 ALE 引脚下降沿锁存）P2 提供高八位。

DB，P0 提供，同一时间只能有一个有效的数据传送通道

CB（包括片外系统扩展用控制线和片外信号对单片微机的控制线）

STR{cond}T <Rd>,<addr>;

B.多寄存器存取指令（LDM，STM）

LDM：批量数据加载指令：

LDM {cond}{<type>} Rn{!}， reglist {^}

指令中，type 字段有以下几种：

IA 每次传送后地址加一

令周期有 1、2、4 个机器周期。6Mhz 晶振，一个机器周期 2um，12Mhz 晶振，

先将操作数减 1，判别是否为 0。不为 0，转向目标地址；为 0，则结束循环程序， 【系统并行扩展能力】①地址总线宽度为 16 位，片外可扩展存储器最大容量为

IB 每次传送前地址加一

一个机器周期 1um

【存储器两种基本结构】将程序和数据合用一个存储器空间的为普林斯顿结构； 将程序存储器和数据存储器截然分开，分别寻址的为哈佛结构。在物理上设有 4 个存储器空间：片内程序存储器、片外程序存储器、片内数据存储器、片外数据存储器。在逻辑上有 3 个存储器地址空间：片内、片外统一的 64KB 程序存储器地址空间，片内 256B 数据存储器地址空间，片外 64KB 数据存储器地址空间。

【存储器】片内数据存储器空间，在物理上包含①地址空间 0~127 位为片内数据存储区；②128~255 位为特殊功能存储区 SFR 区；三种基本寻址空间①64KB 的片内、外程序存储器寻址空间②64KB 的片外数据存储器寻址空间③256B 的片内数据存储器寻址空间，包括 SFR 寻址空间。

对于片内数据存储区 0~31 由四个通用工作寄存器（RS0~RS3）直接寻址，32~47

程序往下执行。

绝对调用指令：ACALL addr11（无条件地调用首址为 addr11 处的子程序。操作不影响标志位）

长调用指令：LCALL addr16（无条件地调用首址为 addr16 处的子程序。操作不影响标志位）

子程序返回指令：RET（表示结束子程序，返回 ACALL 或 LCALL 的下一条指令(即断点地址)，继续往下执行）

中断返回指令：RETI（中断服务程序返回，从断点处继续执行，清除内部相应的中断状态寄存器。中断服务程序必须以 RETI 为结束指令）

空操作指令：NOP 不做任何操作，仅将程序计数器 PC 加一，在时间上仅占用一个机器周期。

64KB②片外数据存储器与程序存储器允许两者地址重复，最大容量分别为 64KB

③片外数据存储器和片内数据存储器操作指令不同，允许地址重复④I/O 口扩展和片外数据存储器统一编址⑤片外数据存储器和片内程序存储器操作指令相同， 片外与片内程序存储器的选择由引脚 EA 接地或高来决定。

【I/O 两种数据传送操作】①CPU 和存储器间的数据读写（采用同步定时工作方式）②CPU 和外部设备间的数据传输（无法直接同步数据传送，需设接口电路） 接口电路功能：速度协调、三态缓冲、数据转换

【接口与端口】接口：计算机与外设之间数据传送方面的联系，通过电路实现 端口：通常把接口电路中已编址并能进行读或写操作的寄存器称为端口。一个接口对应多个端口地址。

【外扩的程序存储器和外扩/O 端口地址重叠，80C51 如何区分这些重叠地址】

DA 每次传送过后地址减一

DB 每次传送前地址减一

FD 满递减堆栈

ED 空递减堆栈

FA 满递增堆栈

EA 空递增堆栈

C.单寄存器交换指令（SWP）

D.SWP 字数据交换指令

SWP{cond} Rd,Rm,[Rn];

功能：从 Rn 所表示的内存装载一个字并把这个字放置到目的寄存器 Rd 中，然后把寄存器 Rm 的内容存储到同一内存地址中，即 Rd=[Rn],[Rn]=Rm,其中 Rm,Rn 均

用位寻址，48~127 用字节寻址。其中 16 个位寻址单元加上可位寻址的特殊功能

【布尔操作类指令】传送指令 MOV，位清除指令 CLR，位置 1 指令 SETB，位取

80C51 单片机对片外数据存储器、片内数据存储器及程 序存储器采用不同的指令， 为寄存器。

寄存器一起构成了布尔（位）处理器的数据存储器空间。

【程序存储器功能】存放程序和固定常数。PC 地址总线为 16 位,可扩展的地址空间为 64KB。 ⒈片内和片外程序存储器①EA 引脚高电平，从片内程序存储器

0000H 开始执行,当 PC 值超出 4K，自动转向片外程序存储器空间执行。②EA 引脚低电平，从片外程序存储器 0000H 开始执行.⒉程序入口地址（中断）系统复位后 PC 为 0000H，系统从 0000H 单元开始取指，执行程序。0003H～002DH 用于 5 个中断源的中断服务程序入口地址。

【存储器】片外数据存储器只能用寄存器间接寻址的方法访问，所用寄存器为

DPTR（最大寻址范围 64KB）、R0 和 R1（256B），指令助记符为 MOVX。

反指令 CPL，逻辑与指令 ANL，逻辑或指令 ORL，布尔累加器条件转移指令 JC、

JNC（布尔累加器 C 为 0 程序顺序向下执行，为 1 转向目标地址），位测试条件转移指令 JB、JNB（位变量为 0 程序顺序向下执行，为 1 转向目标地址），位测试条件转移并清 0 指令 JBC（当位标量为 1 时，该位清 0 且目标转向当前 PC 值）

【伪指令】在汇编时不产生机器指令代码，不影响程序的执行，仅指明在汇编时执行一些特殊的操作。汇编起始地址伪指令 ORG，汇编结束伪指令 END，赋值伪指令 EQU，定义标号值伪指令 DL（EQU 定义的标号，在源程序中不能更改，DL 定义的标号可以更改），定义字节伪指令 DB（一个字节 4 位），定义字伪指令 DW

（一个字 8 位），定义存储区伪指令 DS，位定义伪指令 BIT。

会产生不同的控制信号。片 外数据存储器有读 RD 和写 WR 控制信号，程序存储

器有 读 PSEN 控制信号，因此，扩展时虽然数据线和地址线重 复，但由不同的控制信号加以区别。片内数据存储器地 址采用 MOVC 指令，不会产生 RD 和写 WR 控制信号。

【I/O 编址方式】独立编址方式、统一编址方式。

【数据传送控制方式】无条件传送方式、程序查询方式、程序中断方式、直接存储器存取方式（前三种主要）

【8 位并行 D/A 转换器芯片 DAC0832】具有单缓冲、双缓冲和直通三种数据输入工作方式，单缓冲方式的接口电路中有一个处于直通方式，另一个处于受控的锁

例：SWP R0,R1,[R2];将 R2 所表示的内存单元中的字数据装载到 R0，然后将 R1 中的字数据保存到 R2 所表示的内存单元中。

4.程序状态寄存器指令

MRS 指令：对状态寄存器 CPSR 和 SPSR 进行读操作。CPSR：了解当前处理器的工作状态。SPSR：了解进入异常前的处理器状态。

MRS 指令格式

MRS{cond} Rd,CPSR/SPSR

5.协处理器指令

6.软件中断指令

【程序存储器的特点】1、存放经调试正确的应用程序和表格之类的固定常数；2、 【中断定义】程序执行过程中，允许外部或内部事件通过硬件打断程序的执行，

存方式

【访问 RAM 的寻址方式】访问外部 RAM 单元的惟一寻址方式是寄存器间接寻址

采用 16 位的程序计数器 PC 和 16 位的地址总线，可扩展的地址空间为 64 KB；3、 使其转向为处理外部或内部事件的中断服务程序中去，完成中断服务程序后， 【嵌入式系统的特点以及定义】嵌入式系统是以应用为中心，以计算机技术为基

方式。片外 RAM 的 64 KB 单元,使用 DPTR 作为间址寄存器,其形式为@DPTR,例如

64KB 地址是空间连续且统一的。

【堆栈】堆栈两种操作：进栈、出栈，对栈顶单元进行操作。两种形式：向上生成（80C51 的堆栈）、向下生成。是为子程序调用和终端操作而设立的。具体功能有两个：断电保护和保护现场。

【并行 I/O】80C51 共有 4 个 8 位的并行双向口，计有 32 根 I/O 口线。各口的每

CPU 继续原来被打断的程序，这样的过程称为中断服务响应过程。

【中断功能】中断的主要功能：1 实现 CPU 与外部设备的速度配合 2.实现实时控制 3.实现故障的及时发现及处理 4.实现人机联系.

【中断】80C51 有 2,级中断，5 个中断源（能产生中断的外部和内部事件）：两个外部中断源 INT0 和 INT1，三个内部中断源定时器/计数器 T0 和 T1 的定时/计数

础，软硬件可裁剪，适应应用系统对功能、可靠性、成本、体积和功耗等严格要

求的专用计算机系统。IEEE:嵌入式系统是用于控制、监视或辅助操作机器和设备的装置。

特点：小型化与有限资源、与应用密切相关、系统软、硬件协同设计、需要交叉开发环境和调试工具

“MOVX A,@DPTR”的功能是把 DPTR 指定的片外 RAM 单元的内容送累加器 A。片外 RAM 低 256 个单元,除了可使用 DPTR 作为间址寄存器外,也可使用:R0 或 R1 作间址寄存器。例如“MOVX A,@R0'’即把 R0 指定的片外 RAM 单元的内容送累加器

A。

【RISC 的特点】指令集：减少了指令种类，每条指令 的长度是固定的，指令格式

一位均有锁存器、输出驱动器和输入缓冲器所组成。

溢出中断源和串行口发送和接收中断源。5 个中断源可以分为三类，1.外部中断： 【嵌入式系统的特点】1.小型化与有限资源 2.与应用密切相关 3.系统软、硬件协

和寻址模式相当少，一个周 期就可以执行一条指令；2、流水线：指令的处理过

【并行 I/O】P0 口多功能 8 位口，字节访问地址: 80H，位访问地址：80H～87H。 外部中断有两种信号触发方式：电平有效方式：若引脚上采样到有效的低电平，

同设计 4.需要交叉开发环境和调试工具

程被 拆分成几个更小的。能够被流水线并行执行的单元，允 许流水线在当前指

P0 中一个多路开关：多路开关的输入有两个，地址/数据输出；输出锁存器的输出 Q。内部信号置 1，接地址/数据输出端，反之接输出锁存器的 Q 端。

【并行 I/O P0~P3 口功】P0 口的功能①I/O 口：输出锁存、输入缓冲，输入时需先将口置 1；每根口线可以独立定义为输入或输出②地址／数据复用总线：作数据总线用时，输入／输出 8 位数据 D0～D7；作地址总线用时，输出低 8 位地址

A0～A7。当 P0 口作地址／数据复用总线后就再也不能作 I/O 口使用。P1 口的操作：字节操作和位操作；读引脚操作和读锁存器操作。P2 口的工作状态是输出高

8 位地址。P2 口的功能：作 I／O 口使用；输出程序存储器或片外数据存储器的高 8 位地址，与 P0 输出的低地址一起构成 16 位地址线。P3 口的功能：可做 I/O 口使用，为准双向口；可以作为替代功能的输入、输出。

则向 CPU 提出中断请求；跳变有效方式：若引脚上采样到有效负跳变，则向 CPU

提出中断请求。各中断入口地址：INT0―0003H，T0—000BH，INT1—0013H，T1—

001BH，RI 和 TI—0023H。

⑴INT0：外部中断 0。当 IT0＝0 时，低电平有效；当 IT0=1 时，下降沿有效。

⑵INT1：外部中断 1。当 IT1＝0 时，低电平有效；当 IT1=1 时，下降沿有效。

⒉定时中断：当计数器发生计数溢出时，表明设定的定时时间到或计数值已满， 这时可以向 CPU 申请中断。80C51 有两个源，即：TF0：T0 溢出中断。TF1：T1 溢出中断。

⒊ 串行中断：每当串行口发送或接收一帧串行数据时，就产生一个中断请求。

RXD，TXD：串行中断。

【如何进行嵌入式操作系统的选型】一般根据所需的功能、处理速度和存储器寻

址能力来选择合适的处理器。

【嵌入式系统的组成】一般都是由嵌入式计算机系统和执行装置组成。其中嵌入式计算机系统由硬件层、中间层、系统软件层和应用软件层组成。在嵌入式处理器的分类中，单片机属于微处理器。

【嵌入式最小系统】硬件部分包括嵌入式系统外部连接和芯片的内部实现。在硬件的系统外部连接中，电源电路、时钟电路、存储系统、调试接口和复位电路是必须的，和嵌入式微处理器芯片以及接口电路一起构成了嵌入式最小系统。

【RISC（精简指令系统计算机）的设计准则】1.指令集：选用使用频率较高的一些 简单指令 复杂指令的功能由简单指令来组合一个时钟周期 内完成一条指令 2.

令译码器阶段去取其下一条指令；3、 寄存器：RISC 拥有更多的通用寄存器，每个寄存器都可 以存放数据或地址，寄存器可为所有的数据操作提供快 速的局部储存访问；4、Load-Store 结构：处理器只处 理寄存器中的数据。独立的 load 和

store 指令用来完成 数据在寄存器和外部存储器之间的传送（5、充分利用 VLSI 芯片的面积；6、提高计算机运行速度；7、便于设 计，可靠性高；8、有效支持高级语言程序）

【ARM 的运行模式及特点】用户（usr）:正常程序执行 模式。系统（sys）:运行操作系统的特权任务。3.快中 断（FIq）:支持高速数据传输及通道处理。中断

（irq）： 用于通用的中断处理。管理（svc）:操作系统保护模式。 中止（abt）:用于支持虚拟内存和/或存储器保护。未定 义（und）:支持硬件协处理器的软件仿

【工作方式】80C51 单片微机共有复位、程序执行、低功耗以及编程和校验四种

【外部中断 INT0 的中断响应过程】中断采样：中断采样是针对外部中断请求信

流水线：指令的处理过程被拆分成更小的、能够被流水线执行的单元。3.寄存器： 真。除用户和系统模式为异常模式。除用户模式，其余为特权模式。

工作方式。

号进行的，可以直接置位 TCON 或 SCON 中的中断请求标志；2、中断查询：若

拥有多个通用寄存器 4.LOAD/STORE 结构：实现寄存器与外存之间的数据传送。

【CPSR 与 SPSR 的关系】在处理器所有的运行模式下均 可以访问当前的程序状态

复位：单片机的初始化操作，把 PC 初始化为 0000H，单片机从 0000H 单元开始

查询到某中断标志为 1，则按优先级的高低进行处理，即响应中断；3、中断响应： 【RISC 架构的优点】1.充分利用 VLSI 芯片的面积 2.提高计算机运行速度：指令；

寄存器（CPSR）， 每一种异常模 式下又都有一个专用的程序状态保存寄存器

执行，正常初始化，也可使系统从死锁状态恢复。复位信号为 RST 引脚：高电平有效，有效时间应持续 2 个机器周期以上。复位方式有三种：上电自动复位、按键电平复位和外部脉冲复位

程序执行：单片机的基本工作方式。复位后 PC＝0000H，程序执行从 0000H 开始。一般在 0000H 开始的单元中存放一条无条件转移指令，跳转到实际主程序入口去执行。SJMP MAIN 转主程序

首先将当前程序计数器 PC 的内容压入堆栈进行保护，再将对应中断源的中断矢量地址装入 PC，执行中断服务程序。运行直到遇到 RETI 指令为止，最后恢复原程序的断点地址执行，且恢复中断触发器原先状态。

【中断矢量】当 CPU 响应中断时，由硬件产生一个固定的地址，即矢量地址，由矢量地址指出每个中断源的中断服务程序的入口。

【中断控制】串行口的中断标志存放在 SCON 寄存器中，其余存放在 TCON 寄存

通用寄存器；流水线 3.便于设计、可靠性高 4.有效支持高级语言程序

【ARM 内核 RISC 架构的特别技术】1.所有的指令可根据前面的执行结果决定是否被执行 2.可用加载/存储指令批量传输数据 3.可在一条数据处理指令中同时完成逻辑处理和移位处理 4.在循环处理中使用地址的自动增减来提高运行效率

【ARM 的流水线技术】指令的二级流水线结构：若取指和执行阶段时间上完全重叠，指令周期减半，速度提高 1 倍 ARM7:三级流水（预取，译码，执行）ARM9:

（SPSR），当 异常发生时，SPSR 用于保存当前程序状态寄存器 CPSR 的状态，以便从异常退出时，由 SPSR 来恢复 CPSR，从 而进行异常处理。

【大端、小端模式】ARM 体系结构可以用两种方法存储 字数据，称之为大端格式和小端格式。大端格式：字数 据的高字节存储在低地址中，而字数据的低字节则存放 在高地址中。小端格式：与大端储存格式相反，在小端 储存格式中，低地址中存放的是字数据的低字节，高地 址存放的是字数据的高字节。

低功耗：两种低功耗方式：待机方式和掉电保护方式。

器中。中断允许寄存器 IE（为 0 禁止为 1 允许）。80C51 有高低两个中断优先级， 五级流水（预取，译码，执行，访存，写回）ARM10:六级流水（预取，发射，译

【区分程序存储器和数据存储器地址冲突】不发生数据冲突的原因是：MCS-51

1.待机方式：IDL=1，80C51 进入待机方式。振荡器仍运行，并向中断逻辑、串行口和定时器/计数器电路提供时钟，中断功能继续存在。向 CPU 提供时钟的电路被阻断，CPU 不工作，SP、PC、PSW、ACC 及通用寄存器冻结在原状态。采用中

通过中断优先级寄存器 IP 控制（为 0 低优先为 1 高优先）

【中断撤除】1.由单片微机内部硬件自动复位 2.应用软件清楚相应标志 3.采用外加硬件结合软件清除中断请求。

码，执行，访存，写回）ARM11:八级流水（预取，预取，发射，译码，转换，执

行，访存，写回）

【ARM 处理器支持两种指令集】32 位的 ARM 指令集：处理 32 位数据时性能较

中访问程序存储器和 数据存储器指令不一样；程序存储器访问指令为 MOVC； 数据存储器访问指令为 MOVX；选通信号不同，前者为 /PSEN,后者为/WR 与/RD。

【中断响应的现场保护】所谓现场是指中断发生时单片微机中存储单元、寄存器、

断方式或硬件复位来退出待机方式。若产生一个外部中断请求信号，单片机响应

【中断响应条件】由中断源发出中断申请；中断总控制位 EA=1，即 CPU 开中断； 高；16 位的 Thumb 指令集：ARM 指令集的功能子集。特点：密度高，所占存储

特殊功能寄存器中的数据或 标志位等。因此，在编写中断服务程序时必须考虑保

中断，PCON.0 位（IDL 位）被硬件自动清“0”，单片机退出待机方式进入正常工作方式。

2.掉电保护方式：PD 位控制单片机进入掉电保护方式，当 80C51 检测到电源故障时，进行信息保护，把 PCON.1 位置“1”，进入掉电保护方式。单片机一切工作停止，只有内部 RAM 单元内容被保护。依靠复位退出掉电保护方式，当 Vcc 恢复正常后，只要硬件复位信号维持 10ms，就能使单片机退出掉电保护方式，CPU 则从进入待机方式的下一条指令开始重新执行程序。

【寻址方式】寻址方式是指令中给出的寻找操作数或操作数地址的方法。

1.立即寻址：在指令中直接给出操作数，出现在指令中的操作数称为立即数。立即数前面必需加上前缀“＃”。如：指令 MOV DPTR，＃1234H

2.直接寻址：在指令中直接给出操作数的地址例：指令 MOV A，3AH。直接寻址是访问特殊功能寄存器的唯一方法。寻址范围只限于片内数据存储器。

3.寄存器寻址：在指令中将指定寄存器的内容作为操作数，例如：指令 INC R0

申请中断源的中断允许位为 1，即中断没有被屏蔽； 无同级或更高级中断正在服务；当片指令周期已经结束；若现行指令为 RETI 或访问 IE 或 IP 指令时，该指令以及紧接着的另一条指令执行结束以后才响应。

【中断响应】响应中断后，硬件生成长调用指令 LCALL，先将程序计数器 PC 的内容压入堆栈进行保护，先压低位后压高位，堆栈指示器 SP 加 2

【中断需要解决的问题】**①**当单片机内部或外部有中断申请时,能及时响应中断,中止正在执行的任务,转去处理中断服务子程序。中断服务处理以后,能正确回到原来的断点处继续处理原先的任务。**②**当有多个中断源同时申,请中断时,单片机应能首先响应优先级高的中断源,能实现中断优先级的控制。‘**③**当低优先级中断源正在享用中断服务时,若这时优先级比它高的中断源也申请中断,则要求单片机能中止低优先级中断源的服务程序,转去执行更高优先级中断源的服务程序,实现中断嵌套,并能逐级正确返回原断点处,继续处理原先的任务。

【中断与子程序的异同】中断服务子程序与普通子程序有何异同之处？当中断产

空间较小；可以使用 32 位的寻址空间、32 位寄存器、32 位位移与逻辑运算、32

位的内存存取访问功能。

【ARM 微处理器的工作状态】ARM 状态，此时处理器执行 32 位的字对齐的

ARM 指令；Thumb 状态，此时处理器执行 16 位的、半字对齐的 Thumb 指令。可以相互调用，只要遵循一定的调用规则

【Thumb 指令与 ARM 指令的时间效率和空间效率关系】存储空间约为 ARM 代码的 60％～70％；存储器为 32 位时 ARM 代码比 Thumb 代码快约 40％；存储器为

16 位时 Thumb 比 ARM 代码快约 40～50％；使用 Thumb 代码，存储器的功耗会降低约 30％

【状态切换方法】ARM 指令集和 Thumb 指令集均有切换处理器状态的指令，并可在两种工作状态之间切换，切换不影响处理器的运行模式和寄存器内容

在开始执行代码时，应该处于 ARM 状态。

【状态寄存器】当前程序状态寄存器 CPSR，可以在任何工作模式下被访问；程

护 现场的功能。在 80C51 单片微机中，现场一般包括累加 器 A\工作寄存器

R0~R7,以及程序状态字 PSW 等。保护的 方法与子程序相同，可以有以下几种：1. 通过堆栈操作 指令 PUSH direct。2.通过工作寄存器区的切换。3.通 过单片微机内部存储器单元暂存。

【定时器/计数器程序例题】要求在 P1.0 引脚上产生周期为 2ms 的方波输出。已知晶体振荡器的频率为 fosc=6MHz。可使用 T0 作定时器，设为方法 0，设定

1ms 的定时，每隔 1ms 使 P1.0 引脚上的电平变反。

答：定时常数 TC=7692，TC 为 7692=1E0CH，转换成二进制数 TCB 为 0001 1110 0000 1100B 取低 13 位，其中高 8 位 TCH=F0H，低 5 位 TCL=0CH（方式 0 为低

13 位，方式 1、2 为 16 位）

TMOD 的设定（即控制字）GATE|C/T|M1|MO|GATE|C/T|M1|MO 前 4 个为 T1，后

4 个 为 T0. ORG 0000H

4.寄存器间接寻址：寄存器内容是操作数地址。表示形式：＠寄存器符号。例如： 生的时候进入中断服务程序，不需要调用；而普通子程序只有被调用了才能执行。 序状态备份寄存器 SPSR，只有在异常模式下，才能被访问；（各种异常模式专有

AJMP MAIN

ANL A，@R1，将 R1 中的内容作为地址进行操作。

5.相对寻址：指令给出的操作数为程序转移的偏移量。在相对转移指令中，给出地址偏移量 rel，其范围为-128B~+127B。目的地址＝(转移指令所在地址＋转移

相同：都是中断当前正在执行的程序，转去执行子程序或中断服务程序；都是由硬件自动地把断点地址压入堆栈，然后通过软件完成现场保护；执行完子程序或中断服务程序后，都要通过软件完成现场恢复，并通过执行返回指令，重新返回

的）当异常发生时，SPSR 用于保存当前程序状态寄存器 CPSR 的状态，从异常退

出时，用于恢复 CPSR，其功能包括：保存 ALU 中的当前操作信息控制允许和禁止中断，设置处理器的运行模式。

ORG 000BH//T0 的中断矢量为 000BH，T1 的中断矢量为 001BH AJMP INQP

ORG 0030H

指令字节数)＋rel。如：指令 JC 80H

到断点处，继续往下执行程序；两者都可以实现嵌套，如中断嵌套和子程序嵌套。 【S3C2410A 的 DMA 控制器】采用 DMA 方式进行数据传输的具体过程如下：1.

MAIN：MOV TMOD,#00H//写控制字，设 T0 为定时器、方式 0

6．变址寻址：以 DPTR 或 PC 为基址寄存器，累加器 A 为变址寄存器，以两者内

不同：中断请求信号可以由外部设备发出，是随机的 ，子程序调用是软件编排好

外设向 DMA 控制器发出 DMA 请求。2.DMA 控制器向 CPU 发出总线请求信号

MOV TH0，＃0F0H//写定时常数（定时 1ms）（对于 TO 是 TH0/TL0，对于 T1 是

容相加后形成的 16 位程序存储器地址作为操作数地址。又称基址寄存器＋变址

的；中断响应后由固定的矢量地址转入中断服务程序，而子程序地址由软件设定； 3.CPU 执行完总线周期，向 DMA 控制器发出相应的回答信号；4.CPU 将控制总线、 TH1/TH0）

寄存器间接寻址。变址寻址指令只有三条：MOVC A，＠A＋DPTR；MOVC A，

＠A＋PC；JMP @A+DPTR 。

7．位寻址：寻址空间为内部数据存储器中有 128 个可寻址位、特殊功能寄存器中可位寻址的位。

【数据指令·类别】数据指令按功能共有五大类：数据传送类指令、算术运算类指令、逻辑运算类指令、控制转移类指令、布尔操作类指令。符号说明：Rn 表示 8 个通用寄存器（0-7），Ri 表示可用作间接寻址的寄存器（R0R1），@为间址寄存器的前置标志，将后接寄存器中的内容作为地址进行操作，#表示后接的数为立即数。

【数据传送指令】指令助记符：MOV、MOVX、MOVC、XCH、XCHD、SWAP、

PUSH、POP。不影响标志位 C、AC、OV。

中断响应是受控的，其响应时间会受一些因素影响，子程序响应时间是固定的。

【中断响应时间】在单级中断系统中，中断响应时间：3→8 个机器周期。中断请求标志位查询占 1 个机器周期，而且是指令的最后一个机器周期，在这个机器周期结束后，CPU 即响应中断，产生硬件长调用 LCALL 指令，执行这条长调用指令需要 2 个机器周期，中断响应时间为 3 个机器周期。

【中断扩展】80C51 只有两个外部中断请求输入端 INT0 和 INT1，可以通过两种方式进行扩展：①可通过增加“OC 门”+ 软件来扩展；②当定时器/计数器有空余时，通过对计数长度的设置，使定时器/计数器的外部输入脚成为外部请求中断输入端。

【定时器/计数器】若是对单片微机的 T0、T1 引脚上输入的一个 1 到 0 的跳变进行计数加一即是计数功能。若是对单片微机内部的机器周期进行计数，从而得到

地址总线及数据总线交给 DMA 控制器控制；5.DMA 控制器向外部设备发出 DMA

请求回答信号；6.在 DMA 控制器控制下进行 DMA 传送；7.数据传送完，DMA 控制器通过中断请求线发出中断信号。CPU 接收到中断信号后，转入中断处理程序进行处理；8.中断处理结束后，CPU 返回断点继续执行，并获得总线控制权

【ARM 指令系统及编程技术】

条件码 cond 在 CPSR 中标志位 含义

EQ Z 置 位 相 等

NE Z 清 零 不 相 等

CS C 置位 无符号数大于或等于

CC C 清零 无符号数等于

MI N 置 位 负 数

MOV TL0，＃0CH

SETB TR0// 启 动 T0 （ 对 于 TO 是 TR0， 对 于 T1 是 TR1） SETB ET0//允许 TL0 中断（对于 TO 是 ET0，对于 T1 是 ET1） SETB EA//CPU 中断开放

AJMP ＄//定时中断等待

ORG 2000H//T0 中断服务程序INQP：MOV TH0，＃0F0H//重装定时常数MOV TL0，＃0CH

CPL P1.0//P1.0 变反输出

RETI//中断返回

【定时器 中断程序】

这时 T0 采用方式 3 工作，其中，TL0 产生 100μs 定时，由 P1．0 输出方波 1；

TH0 产生 200μs 定时，由 Pl．1 输出方波 2；T1 设置为方式 2，作波特率发生器用。fosc＝9．216 MHz。

ORG 0000H

AJMP MAIN

ORG 000BH ；TL0 的中断入口

AJMP ITL0

ORG 00lBH；TH0 的中断入口

AJMP ITH0 ORG 0100H

MAIN： MOV SP，＃ 60H； 设 栈 指 针

MOV TMOD，＃ 23H 设 T0 为方式 3,TI 为 2

MOV TL0，＃0B3H 设 TL0 初值(100μs 定时

MOV TH0，＃66H 设 TH0 初值(200μs 定时)

MOV TL1，＃0F6H 设 TL1 初值(波特率为 2400，第六章公式，方式二，

TH1=TC2）

MOV TH1，＃ 0F6H 设 TH1 初值

SETB TR0 ；启动 TL0

SETB TR1 ；启动 TH0

SETB ET0 ；允许 TL0 中断

SETB ET1 ；允许 TH0 中断

SETB EA ；CPU 中断开放

AJMP ＄

ORG 0200H

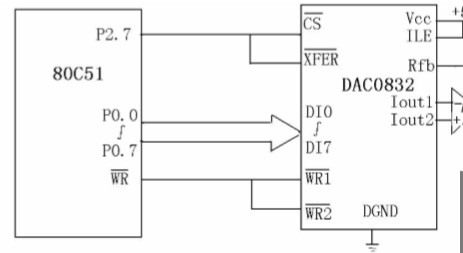
ITL0：MOV TL0，＃ 0B3H；重装定时常数CPL P1．0；输出方波 1(200μs)

RETI

ITH0：MOV TH0，＃66H；重装定时常数CPL P1．1；输出方波 2(400μs)

RETI

【例题】



产生锯齿波由上图可知，输入寄存器和 DAC 寄存器同时受控锁存，DAC0832 的地址为 7FFFH(P2.7=0)

ORG 0000H

SJMP MAIN

ORG 0030H

MAIN： MOV DPTR，#7FFFH ；同时输入输入寄存器和 DAC 寄存器地址

MOV R0，#0 ；转换数字量初值

LP： MOV A，R0

MOVX @DPTR，A； 送 出 模 拟 量LCALL DELAY;波形延时

INC R0；转换数字量加一SJMP LP；循环 D/A 转换

DELAY： ；延时子程序

RET

【并行输出 流水灯程序】

MOV SCON，#00H ；设串行口为方式 0

CLR ES ；禁止串行口中断

MOV A，#80H ；先显示最左边发光二极管

LED: MOV SBUF，A ；串行输出

JNB TI，$ ；输出等待

CLR TI ；软件清中断标志

ACALL DELAY ； 轮 显 间 隔

RR A ；发光右移

AJMP LED ；循环

DELAY：MOV R6,#200H NOP

DJNZ R6,& ；延时子程序

RET

【例题】将 2000H 单元开始的一批数据传送到从 3000H 开始的单元中，数据长度在内部 RAM 的 30H 中。

MOV DPTR，#2000H；源数据区首址PUSH DPL ；源数据区首址压栈保护PUSH DPH

MOV DPTR，#3000H；目的数据区首址

MOV R6，DPL ；目的数据区首址存入寄存器

MOV R7，DPH

LP： POP DPH；取数据区地址指针POP DPL

MOVX A，@DPTR ；取源数

INC DPTR PUSH DPL PUSH DPH

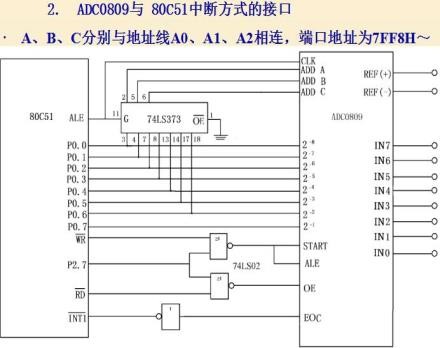
MOV DPL，R6 ；取目的数据区地址指针

MOV DPH，R7

MOVX @DPTR，A ；存入目的数据区

INC DPTR MOV R6，DPL MOV R7，DPH

DJNZ 30H，LP ；若数据块未移完，则继续

POP DPH POP DPL SJMP ＄

MOV TMOD，# 20H ；设 T1 为方式 2 MOV TL1，# 0E8H ； T1 定时常数

MOV TH1，#0E 8H

MOV S CON，# 01000000B ；设串行口为方式 1 MOV R0，#20H ； 设发送数据区首址

MOV R7，#32 ； 发送 32 个 AS CII 码数据

LOOP：MOV A， @R0 ； 取 ASCII 码 数 据

ACALLSP-OUT ；调用串行口发送子程序

INC R0 ；未发送完，则继续

DJNZ R7，LOOP

…

串行口发送子程序

SP-OUT：MOV C， P ； 设 奇 校 验 位

CPL C

MOV ACC. 7， C

MOV S BUF ，A ；带校验位发送

JNB TI， $ ；发送等待

CLR TI RET

【例题】试编写串行接口以工作方式 2 发送数据的中断服务程序

ORG 0023H AJMP SPINT

SPINT：CLR EA ； 关 中 断PUSH PSW ； 保 护 现 场PUSH ACC

SETB EA ； 开 中 断

SETB PSW.4 ；切换寄存器工作组

CLR TI ；清除发送中断请求标志

MOV A，＠R0 ；取数据，置奇偶标志位

MOV C，P ；奇偶标志位 P 送 TB8 MOV TB8，C

MOV SBUF，A ；数据写入发送缓冲器，启动发送

INC R0 ；数据地址指针加 1 CLR 0AFH ； 恢 复 现 场

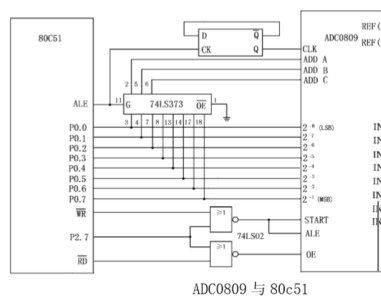
POP ACC

POP PSW SETB 0AFH

CLR PSW .4 ；切换寄存器工作组

RETI ；中断返回

【例题】



采用延时等待 A/D 转换结束方式， 分别对 8 路模拟信号轮流采样一次， 并依次把结果存入数据存储器。

ORG 0000H SJMP M AIN

ORG 0030H MAIN： MOV R1，# 20H

MOV DPTR，# 7FF8H ；指向通道 0 地址

MOV R7 ，#08H ；共需转换 8 个通道

LOOP： MOVX @DPTR， A ； 启动 A/D 转换①

LCALL D 128μ s ；延时等待 A/D 转换结束②

MOVX A， @D PTR ； 读入 A/D 转换值③

MOV @R1，A

INC DPTR ；指向下一通道地址

INC R1

DJNZ R7 ，LOOP ； 8 个通道未转换完则继续 D128μs： … ；延时 128μ s 子程序

RET

【例题】要求在 Pl.O 引脚上产生周期为 2 ms 的方波输出。已知晶体振荡器的频率为 fosc = 6MHz。可使用 T0 作定时器，设为方式 0, 设定 1ms 的定时，每隔

1ms 使 Pl.0 引脚上的电平变反。

解： (1)定时常数计算振荡器的频率 fosc = 6MHz, 机器周期为 2μs, 方式 0 计数器长度 L =13(2^13=8192, 定时时间 t=1ms=0.001s， 定时常数：TC 为 7692 =

lEOCH, 转换成二进制数 TCB，取低 13 位，其中高 8 位 TCH = FOH, 低 5 位为 TCL

=0CH

计数长度为 1EOCH=76 92, 定时为(8192 — 7692) X 2μs = 0.001s

(2) 编 程 ORG 0000H AJMP MAIN

ORG 000BH o/ ;T0 中断矢量

AJMP INQP ORG 0030 B

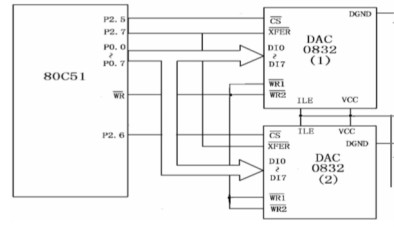
MAIN: MOV TMOD, # OOH ；写控制字，设 T0 为定时器

MOV TH0'# 0F0H ；写定时常数（定时 lms） MOV TL0, #0CH ，

SETB TR0 ；启动 T0

SETB ET0 ；允许 T0 中断

SETB EA ；开放 CPU 中断



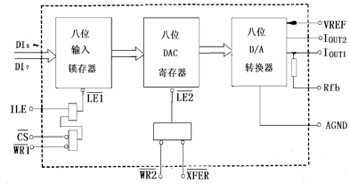
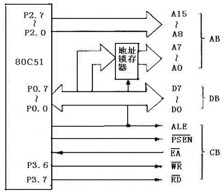
MOV DPTR，# 0DFFFH； 把数据送第一片 D AC0832 输 入锁存器 MOV A ，

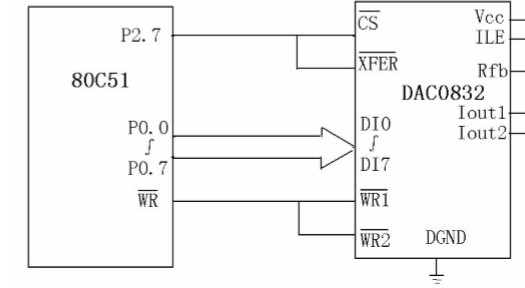
R0 MOVX @D PTR ，A MOV D PTR ，#0BFFFH ；把数据送第二片

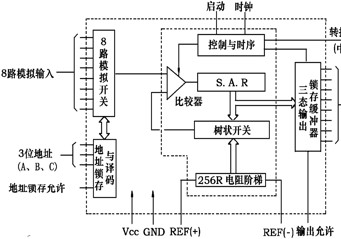
DAC0832 的输 入锁存器 MOV A， R1 MOVX @DPTR，A MOV D PTR ，

#7FFFH； 两片 0832 同时输出模拟量 MOVX @D PTR ，A

【80C51 单片微机三总线结构】







【例题】单片机与 8255A 的接口电路，PA 口作输出口，接 8 个 LED 发光二极管， AJMP ＄ ；定时中断等待

PB 作输出口，接 8 个按键开关，PC 口不用，都工作在方式 0，要实现“按下任意键，对应的 LED 发光”，相应的程序如下：

读 PB 口开关状态，送 PA 口输出控制 LED，循环MOV DPTR,#0FF7FH;指向 8255A 的控制口

MOV A,#82H; 工作方式控制字

MOVX @DPTR,A;向控制口写控制字，PA 输出，PB 输入

LOOP:MOV DPTR,#0FF7DH;指向 8255A 的 PB 口

MOVX A,@DPTR;读 PB 口按键状态

MOV DPTR,#0FF7CH;指向 8255A 的 PA 口

MOVX @DPTR,A;从 PA 口输出，驱动 LED 发光

SJMP LOOP

【例题】中断方式 AD 采样采集 8 路模拟量，并存入 20H 地址开始的内部 RAM

中

ORG 0000H

SJMP MAIN

ORG 0003H ；外部中断 0 入口地址

LJMP INTDATA

ORG 0100H ；数据采集程序 MAIN：MOV R0，＃20H ；数据缓冲区首址MOV R2，＃8 ；8 通道计数器

MOV DPTR，＃7FF8H ；指向 0 通道

START：CLR F0 ；清中断发生标志

MOVX ＠DPTR，A； 启 动 A/D (P2.7=0, /WR=0) ①

SETB IT0 ；置外部中断 0 为边沿触发

SETB EX0 ；允许外部中断 0

SETB EA ；开中断

LOOP：JNB F0，LOOP ；中断发生标志是否为 0 ②

DJNZ R2，START ；8 个通道转换是否结束

SJMP MAIN

INTDATA：MOVX A，＠DPTR ；读数据 (P2.7=0, RD=0)，；硬件撤销中断③

MOV ＠R0，A ；存数据

INC R0

INC DPTR ；指向下一通道

SETB F0 ；置中断发生标志

RETI

【例题】使用 74LS164 的并行输出端接 8 只发光二极管，利用它的串入并出功能， 把发光二极管从左向右依次点亮，并不断循环

MOV SCON，#00H ；设串行口为方式 0 CLR ES ；禁止串行口中断

MOV A，#80H ；先显示最左边发光二极管

LED: MOV SBUF，A ；串行输出

JNB TI，$ ；输出等待

CLR TI ；软件清中断标志

ACALL DELAY ； 轮 显 间 隔

RR A ；发光右移

AJMP LED ；循环

DELAY：…… ；延时子程序

RET

【例题】通过串行口发送带奇偶校验位的数据块 AS CII 码由 7 位组成，其最高位可作为奇偶校验位用。数据块通过串行口发送和接收， 采用 8 位异步通信，波特率为 1200，fo sc= 11.0592MHz 。从内部 RAM 单元 20H ～

3FH 中取出 AS CII 码加上奇偶校验位之后发出。设串行口为方式 1，T1 为方式 2 作为串行口的波特率发生器。

∵SMOD＝ 0 ∴TH 1＝ 232＝E8H ORG 0000H

ORG 2000 H ;TO 中断服务程序

INQP:MOV TH0, #0F0H ；重写定时常数

MOV TL0,#0CH

CPL Pl.0 ; Pl.0 变反输出

RETI ；中断返回

【例题】采用延时等待 A/D 转换结束方式，分别对 8 路模拟信号轮流采样一次， 并依次把结果存入数据存储器。

ORG 0000H SJMP MAIN ORG 0030H

MAIN：M OV R1，#20H

MOV DPTR ，#7FF 8H ；指向通道 0 地址

MOV R7，# 08H ； 共需转换 8 个通道

LOOP：MOVX @D PTR ，A； 启动 A/D 转换① LCALL D 128μs ；延时等待 A/D 转换结束② MOVX A， @DPTR ； 读入 A/D 转换值③

MOV @R1，A

INC DPTR ； 指向下一通道地址

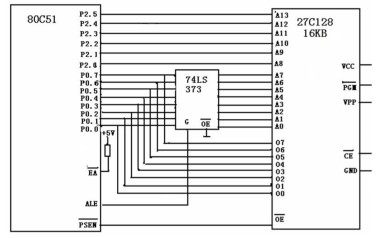
INC R1

DJNZ R7， LOOP ；8 个通道未转换完，则继续。D128μs： … ； 延 时 128

μs 子程序

RET

【例题】



对 8255A 各口作如下设置：A 口方式 0， B 口方式 0，从 A 口输入，从 B

口、C 口输出。

工作方式控制字为 10010000 ，即 90H。初始化程序段：

MOV A ，#90H ；设 A 口、B 口为方式 0 ；A 口输入，

B 口、C 口输出

MOV DPTR ，#7FFFH MOVX @DPTR ，A

MOV DPTR ，#7FF CH ；从 A 口输入

MOVX A， @DPTR

MOV DPTR ，#7FFDH ；从 B 口输出

MOVX @DPTR ，A

MOV DPTR ，#7FFEH ；从 C 口输出

MOVX @DPTR ，A

【例题】