



《单片机与接口技术》

主讲人：李刚



02

单片机的结构 和工作原理

2.1 概述

2.2 单片机的引脚

2.3 AT89S51单片机的内部组成及结构

2.4 存储器

2.5 CPU的时钟电路和时序

2.6 单片机工作方式

2.7 输入/输出端口结构

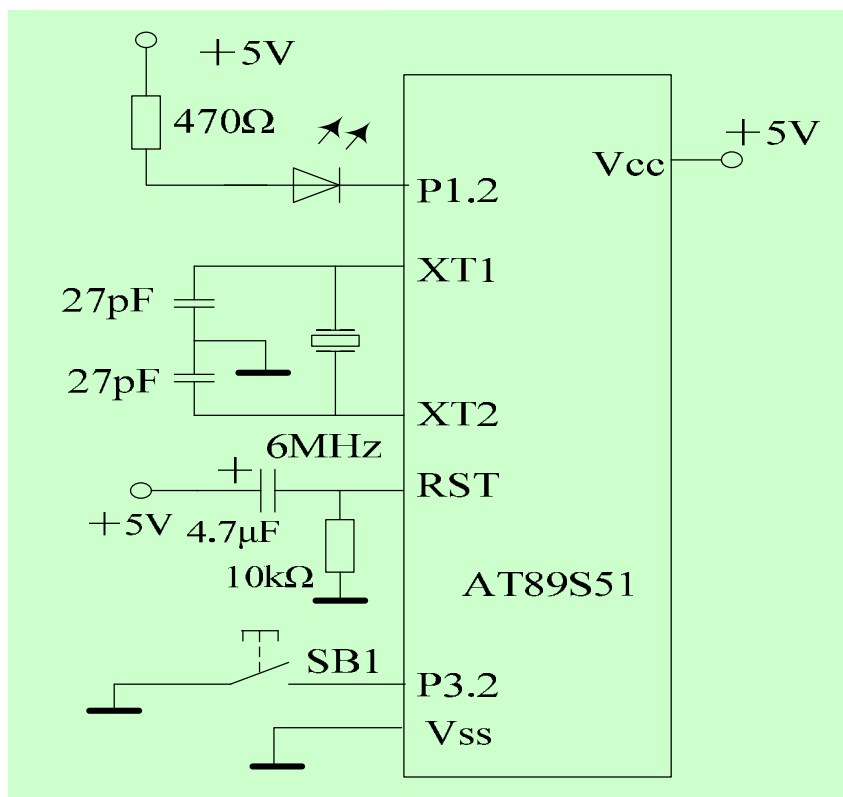
2.1 概述

学习目标：熟悉80C51单片机的内部结构，掌握80C51单片机各组成部分的功能和特点。

引入 单灯受控闪烁

任务描述：单片机上电工作时，发光二极管按1HZ左右频率闪烁，按下键S1后在原状态基础上停止5秒，再按原频率闪动。

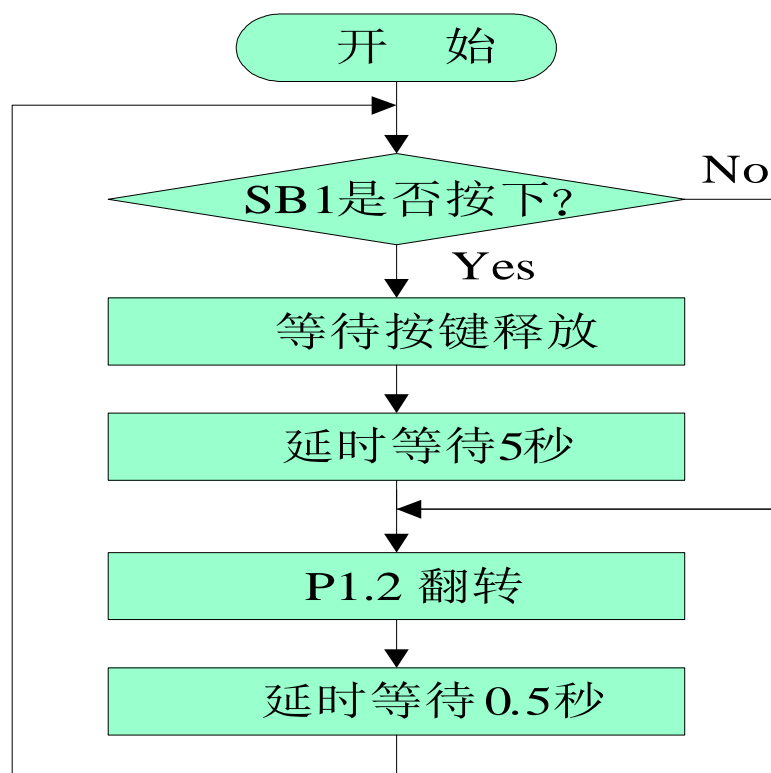
1. 硬件电路



2. 工作原理

从图上可以看出，当P1.2引脚输出低电平时，则发光二极管上有电流流过，发光二极管发光，否则，当引脚输出为高电平时，发光二极管上无电流流过，发光二极管熄灭。由于SB1按键一端接地，当按键按下时，对应的引脚被拉成低电平。单片机通过查询该输入口线的高低电平状态，就知道对应的按键是否按下。

3. 主程序流程图



4. 控制程序

```

                ORG    0000H
                CLR    P1.2
L0:             JB     P3.2, L3    ; 无键按下跳转
L1:             JNB    P3.2, L1    ; 等待按键释放
                MOV    R3, #10
L2:             LCALL  DELAY
                DJNZ   R3, L2      ; 延时5秒
L3:             LCALL  DELAY      ; 延时0.5秒
                CPL    P1.2      ; 输出反相
                SJMP   L0
DELAY:          MOV    R7, #10
D1:             MOV    R6, #255
D2:             MOV    R5, #255
D3:             DJNZ   R5, D3
                DJNZ   R6, D2
                DJNZ   R7, D1
                RET
                END
  
```

指令、伪指令
将在第三、四
章中详细讲解

```

#include "reg51.h"
sbit led1 = P3^2;
sbit key1 = P1^2;
void delay(int x)
{ int i,j;
  for (i=0;i<x;i++)
    for (j=0;j<1000;j++)
      ;
}
main()
{
  while(1)
  {
    if (key1==0)
      { while (key1==0) ;
        delay(5000);
      }
    led1=!led1;
    delay(500);
  }
}
  
```

5 源程序的编辑、编译、仿真、下载

1. 打开“Keil” 仿真软件进行程序的编辑、编译。

- 1、新建文件进行编辑
- 2、新建一个工程
- 3、选择单片机类型，进行工程设置
- 4、编译程序文件直至通过
- 5、模拟仿真

2. 将目标文件下载到实验开发板上的AT89S51单片机芯片中，观察程序运行结果。

- 1、编译成功后打开下载软件
- 2、选择好单片机AT89S51并测试
- 3、打开编译生成的.BIN或.HEX文件
- 4、选择自动写完成擦除、写、读、校验过程
- 5、观看运行结果

3. Keil 软件模拟仿真

几点说明:

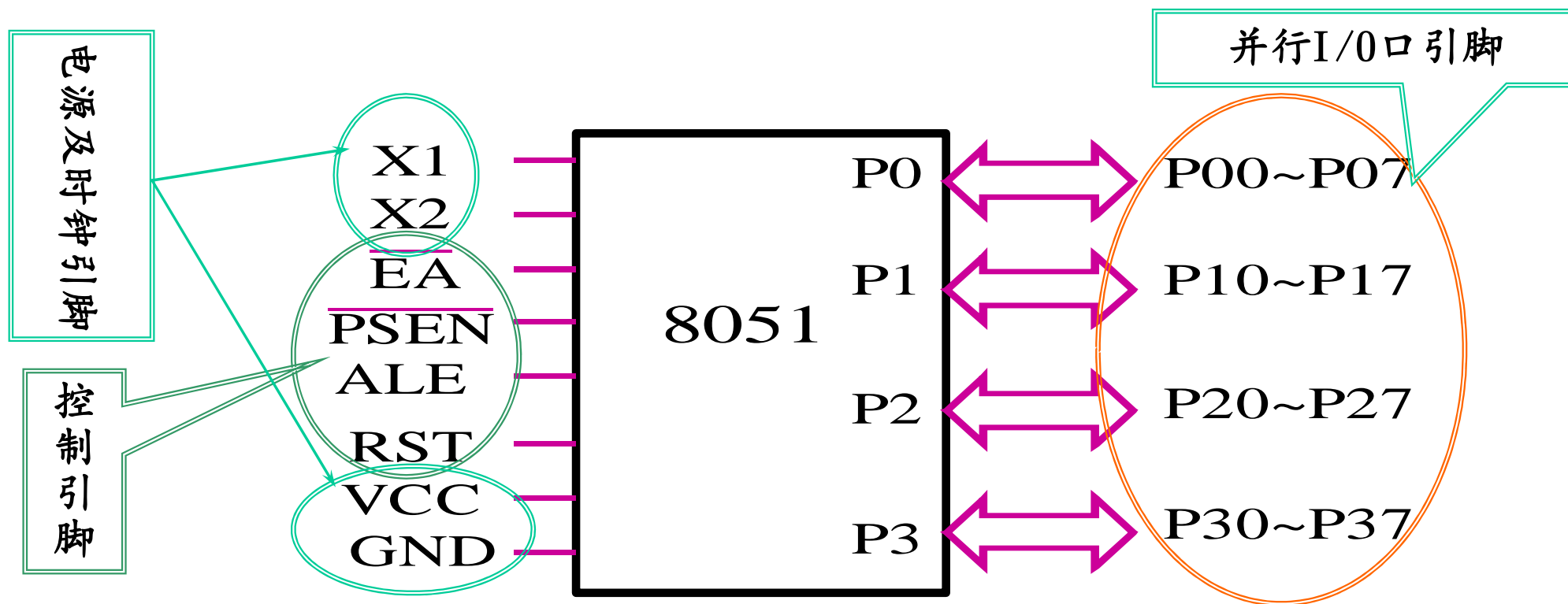
- 单片机实质上是一个芯片。在实际应用中，通常很难将单片机直接和被控对象进行电气连接，必须外加各种扩展接口电路、外部设备、被控对象等硬件和软件，才能构成一个单片机应用系统。
- 单片机应用系统是以单片机为核心，配以输入、输出、显示、控制等外围电路和软件，能实现一种或多种功能的实用系统。
- 单片机应用系统是由硬件和软件组成的，硬件是应用系统的基础，软件则在硬件的基础上对其资源进行合理调配和使用，从而完成应用系统所要求的任务，二者相互依赖，缺一不可。

思考题：

1. 单片机程序如何控制发光二极管？
2. 程序指令如何运行？
3. 如何用按键控制其它灯？
4. 要用1个按钮同时控制2个灯，在硬件与软件上要作什么改进？

2.2 单片机的引脚

引脚逻辑图：80C51单片机为40条引脚双列直插式封装，引脚可分为三个部分。



- 1. I/O口线功能

4个8位并行 I/O 接口引脚

$P_{0.0} \sim P_{0.7}$ 、 $P_{1.0} \sim P_{1.7}$ 、 $P_{2.0} \sim P_{2.7}$ 和 $P_{3.0} \sim P_{3.7}$

- 2. 控制线

ALE: 地址锁存允许信号端

PSEN: 外部程序存储器读选通信号端

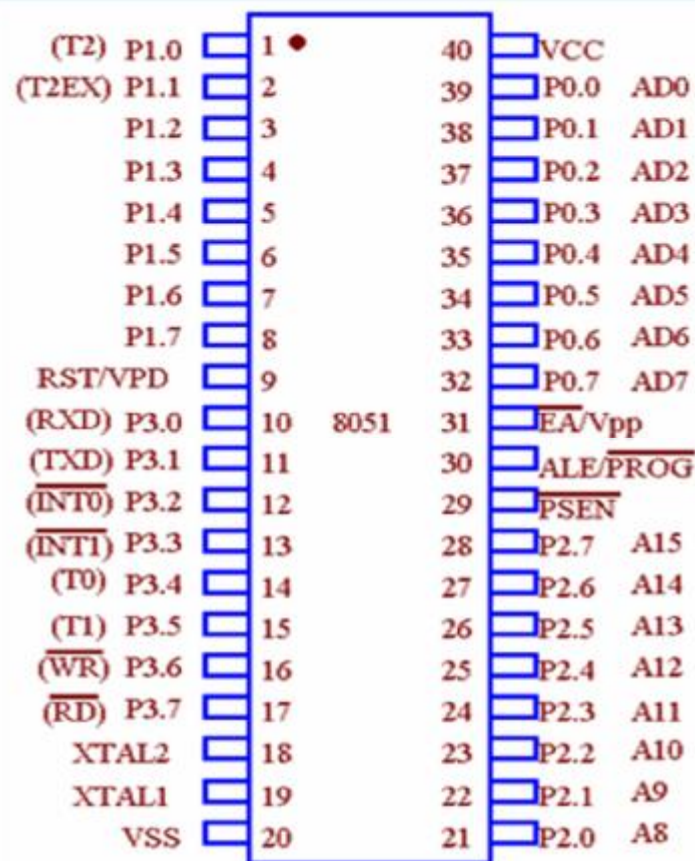
EA/ V_{pp} : 程序存储器选择信号端和编程电源输入端

RST: 复位信号端, 输入10ms以上高电平脉冲, 单片机复位。

- 3. 电源及时钟引线

工作电源: V_{CC} 、 V_{SS} 、

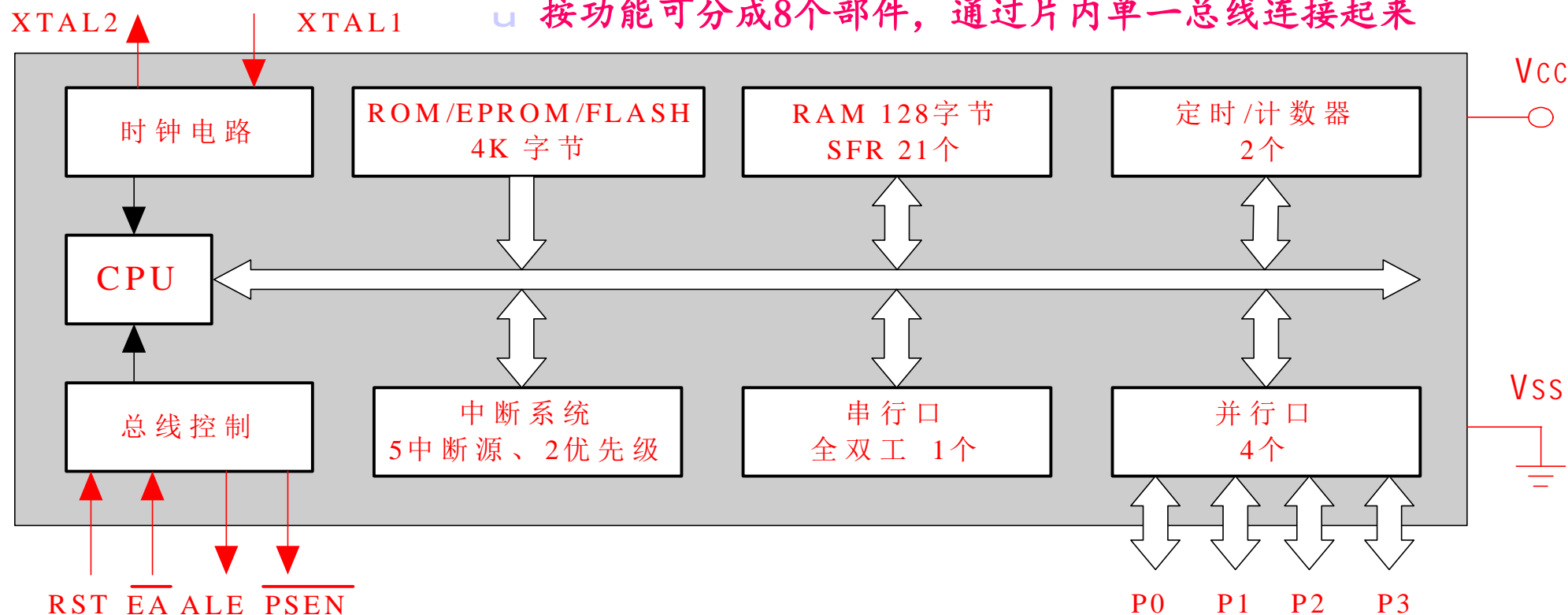
时钟输入: XTAL1、XTAL2 。



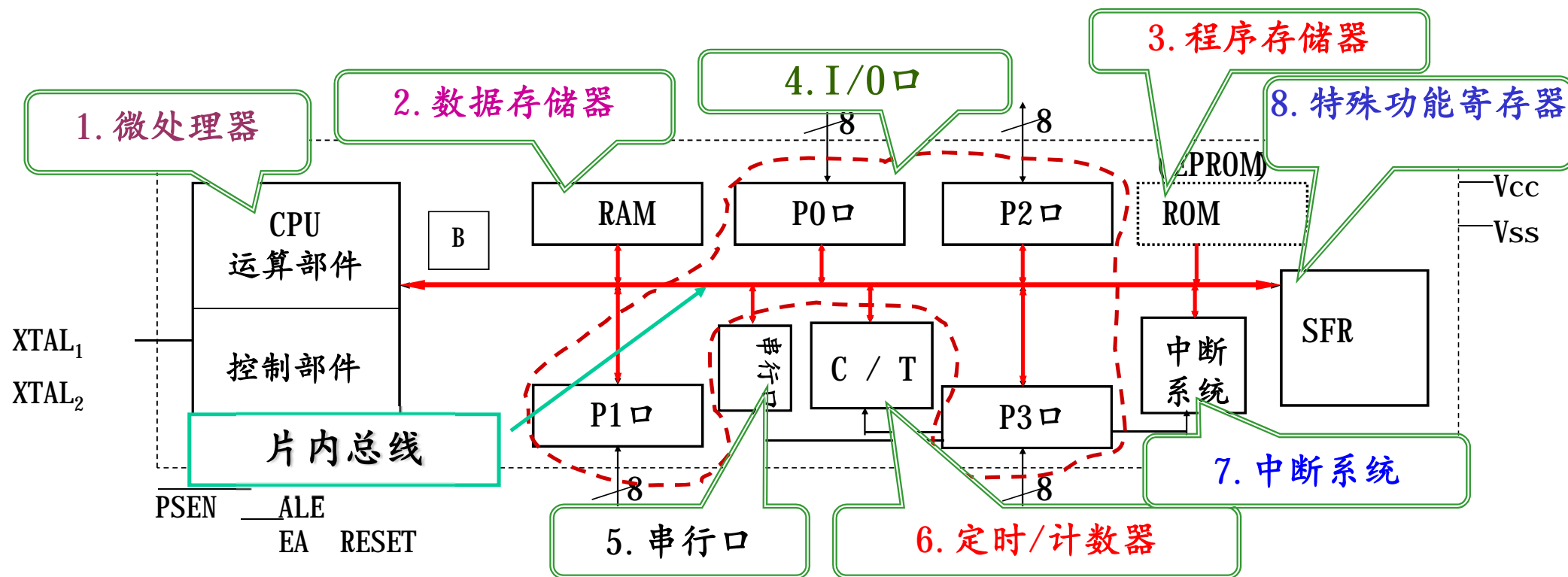
2.3 AT89S51单片机的内部组成及结构

2.3.1 AT89S51的内部结构

按功能可分成8个部件，通过片内单一总线连接起来



控制方式：SFR对各功能部件集中控制



- 1) 中央处理器（CPU）： 8位，运算和控制功能。
- 2) 内部数据存储器（内部RAM）： 共256个RAM单元，用户使用前128个单元，用于存放可读写数据，后128个单元被专用寄存器占用。
- 3) 内部程序存储器（内部ROM）： 80C51共有4 KB掩膜ROM，用于存放程序、原始数据或表格。89C和89S系列内部有不同容量的FLASH存储器，使用更方便。

- 4) 定时/计数器：80C51共有两个16位的定时/计数器，以实现定时或计数功能。
- 5) 并行I/O口：MCS-51共有4个8位的I/O口（P0、P1、P2、P3）。
- 6) 串行口：一个全双工的串行口。
- 7) 中断控制系统：共有5个中断源，即外中断两个，定时/计数中断两个，串行中断一个。

- 8) 时钟电路：时钟电路为单片机产生时钟脉冲序列。系统允许的晶振频率一般为2 MHz至12 MHz（89S系列为0 Hz至33 MHz）。
- 9) ISP在系统编程：AT89S51/52内有4/8K字节的FLASH存储器，支持在系统编程ISP和1000次的擦写周期。
- 10) 看门狗监视定时器WDT：AT89S系列单片机内部增加了一个定时器T3作为看门狗监视定时器。

2.3.2 AT89S51单片机的CPU

- CPU由运算器、控制器和若干SFR（ACC，B，PSW，SP，DPTR）构成。
- （1）运算器：实现数据的算术运算和逻辑运算。
- （2）累加器ACC：提供一个操作数并存放运算结果，其进位标志CY同时也是布尔处理器的累加器。
- （3）寄存器B：用于乘法和除法操作，也用作暂存器。

(4) 数据指针DPTR

DPTR主要是用来保存16位地址

DPTR为16位寄存器，编程时，
可以按16位寄存器来使用，也可以
按两个8位寄存器来使用。

```
MOVX  A, @DPTR
MOVX  @DPTR, A
```

```
MOVC  A, @A+DPTR
```

```
MOV DPTR,#1000H
MOV DPH,#10H
MOV DPL,#00H
```

(5) 堆栈指针SP(Stack Pointer)

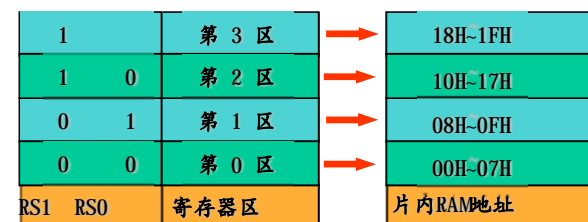
指向栈顶数据的地址，8位，可软件设置初值，复位时SP = 07H。



(6) CPU标志寄存器



- CY: 进位标志, 有进位或借位时为1, 否则为0
- AC: 辅助进位标志当D₃向D₄有进位或借位时为1, 否则为0
- F0: 用户标志
- RS1、RS0: 工作寄存器组
- OV: 溢出标志, 结果超过-128~+127时, OV=1, 否则为0
- P: 奇偶校验标志, 当ACC中“1”的个数为奇数个时, P=1



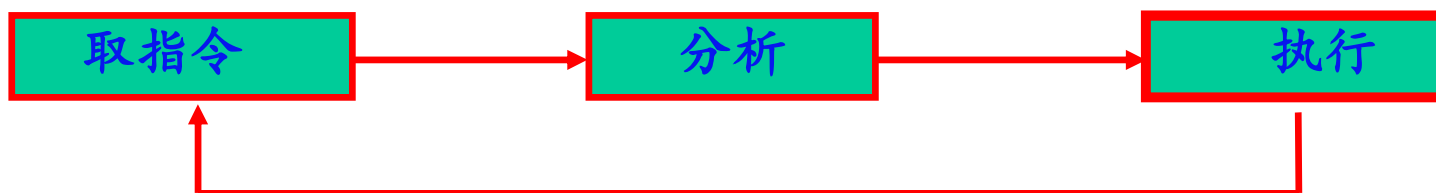
CPU的工作原理

- 程序计数器（PC）：是一个16位的计数器，其内容为将要执行指令的地址。PC有自动加1功能，从而实现程序的顺序执行。只能通过转移、调用、返回等指令改变其内容，以实现程序的转移。
- 指令寄存器：用于暂时存放指令，等待译码。
- 指令译码器：对指令进行译码，分析指令功能。

CPU执行程序一般包括两个主要过程

第一，就是从程序存储器中**取出指令**，指令的地址由PC指针提供。

第二，就是**指令执行**过程，取出的指令代码首先被送到CPU中控制器中的指令寄存器，再通过指令译码器译码变成各种电信号，从而实现指令的各种功能。



例：程序 `MOV A, #05H` 执行过程：

单片机复位后程序计数器PC的内容为0000H，故必须从0000H单元开始取指令来执行程序。0000H单元是系统的起始地址，一般在该单元存放一条无条件转移指令，用户设计的程序是从转移后的地址开始存放执行的。

假设PC当前值为2000H。

1、取指令

1. PC的内容2000H送地址寄存器。
2. PC的内容自动加1 (PC=2001H) 。
3. 地址寄存器内容选中2000H存储单元 。
4. 被选中的存储单元的内容74H送指令寄存器。

2、分析指令

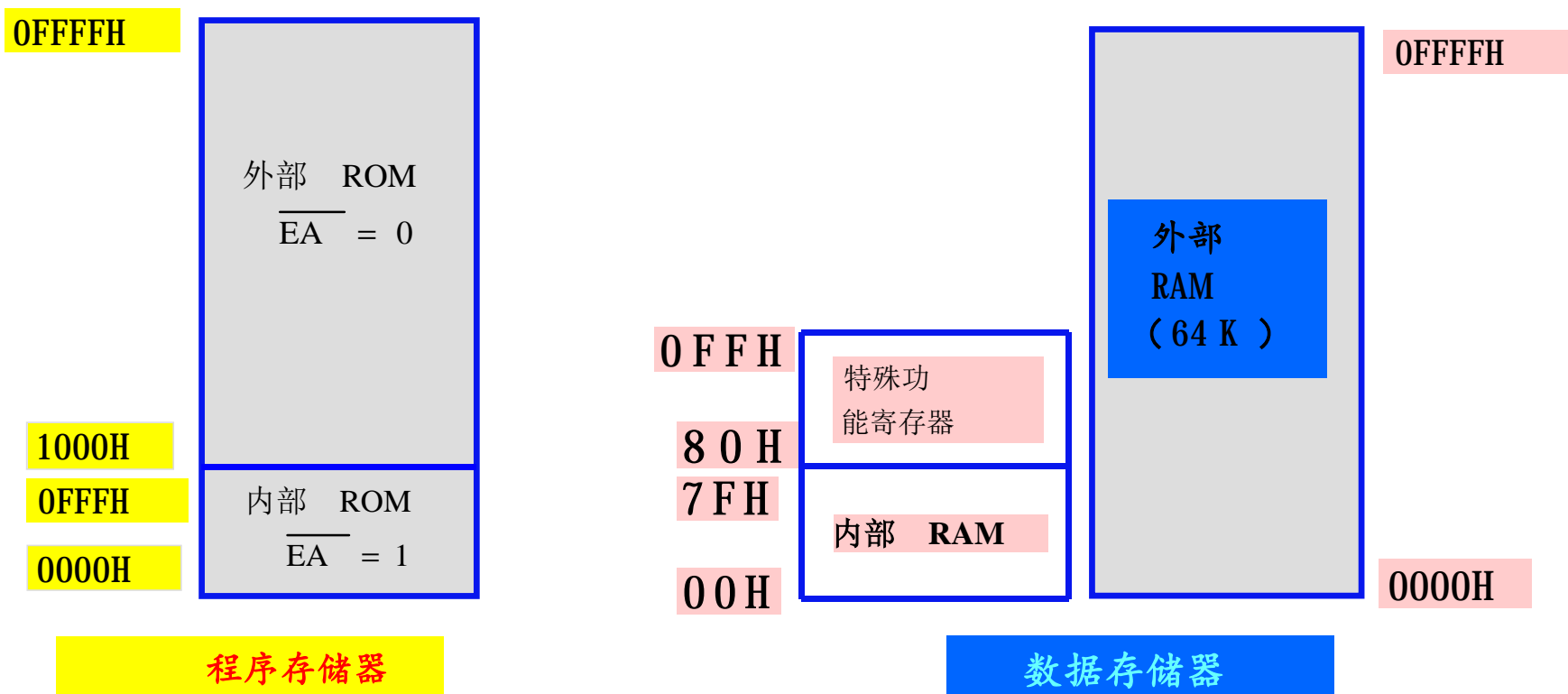
1. 操作码74H送指令译码器。
2. 指令译码器将操作码分解成一系列的微操作信号。
3. 微操作信号与时序电路相结合，产生完成指令任务的工作时序。

3、执行指令

1. 按照指令的工作时序，PC的当前值（2001H）再送地址寄存器，选取中地址为2001H的存储单元。
2. 2001H单元中的内容05H直接送累加器A。
3. 2001H指令单元数据取出后，PC值自动加1，变成2002H，为取下条指令作好准备。

2.4 存储器

单片机存储器组织结构



2.4.1 程序存储器

用途： 存放程序及程序运行时所需的常数。

寻址范围： 0000H — FFFFH 容量64KB，即地址长度：16位

$\overline{EA} = 1$ ，寻址内部ROM；

说明：当PC值超过片内ROM容量时会自动转向外部存储器空间。

$\overline{EA} = 0$ ，寻址外部ROM。

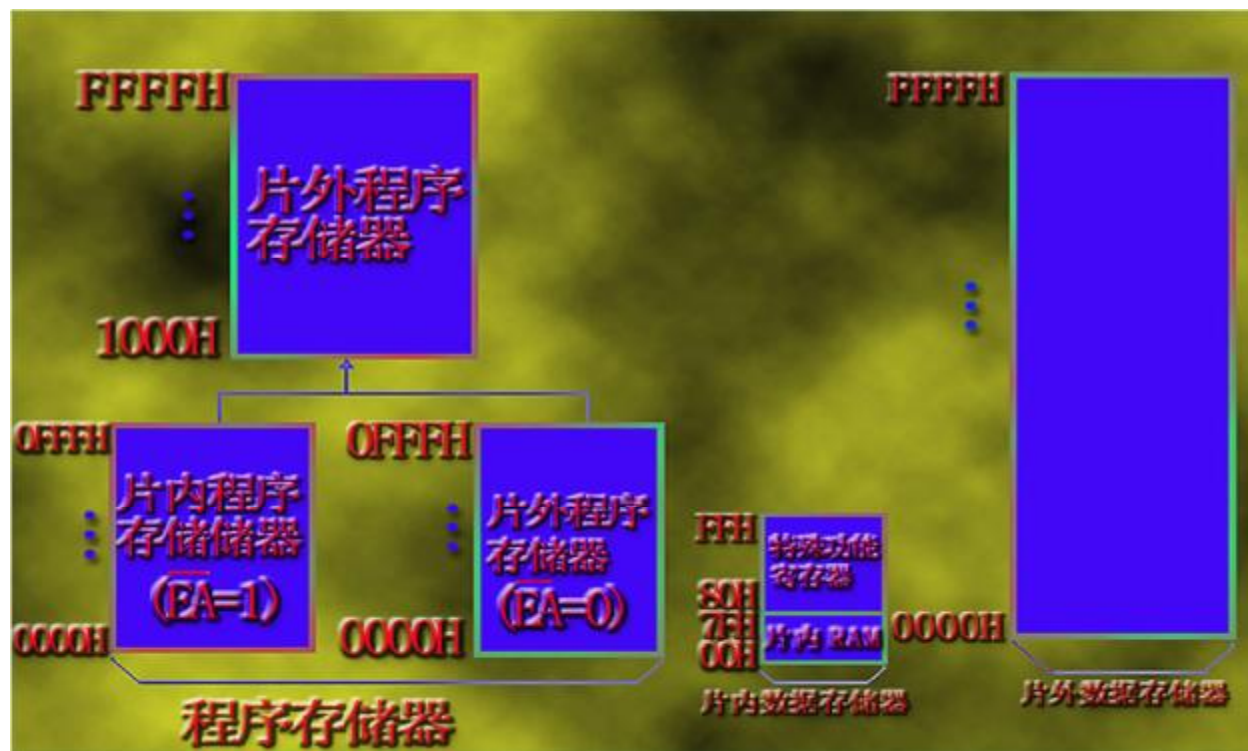
说明：80C31单片机

\overline{EA} 必须接低电平。

• 64KB 程序存储器空间，
EA不同电平选择片内或片外低位存储单元

特殊存储单元：

- 0000H —— 系统复位，PC指向此处；
- 0003H —— 外部中断0入口
- 000BH —— T0溢出中断入口
- 0013H —— 外中断1入口
- 001BH —— T1溢出中断入口
- 0023H —— 串口中断入口



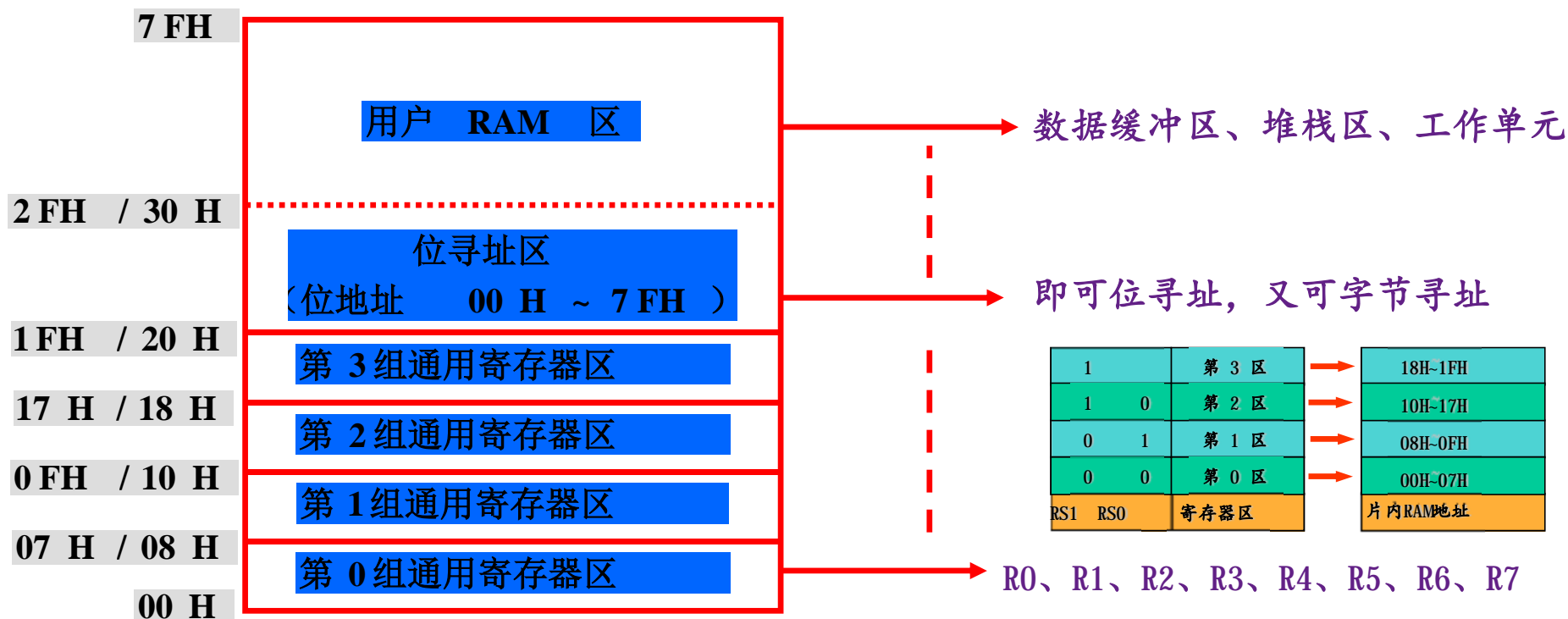
转向机器码

02 第二章 单片机的结构和工作原理

2.4.2 数据存储器

用途：存放程序执行的中间结果和过程数据的。

物理上分为两大区域：00H~7FH即128B用户RAM区。80H~FFH即特殊功能寄存器区。



RAM位寻址区位地址表(P11)

单元地址	MSB	←-----位地址-----→						-----LSB-----
2FH	7F	7E	7D	7C	7B	7A	79	78
2EH	77	76	75	74	73	72	71	70
2DH	6F	6E	6D	6C	6B	6A	69	68
2CH	67	66	65	64	63	62	61	60
2BH	5F	5E	5D	5C	5B	5A	59	58
2AH	57	56	55	54	53	52	51	50
29H	4F	4E	4D	4C	4B	4A	49	48
28H	47	46	45	44	43	42	41	40
27H	3F	3E	3D	3C	3B	3A	39	38
26H	37	36	35	34	33	32	31	30
25H	2F	2E	2D	2C	2B	2A	29	28
24H	27	26	25	24	23	22	21	20
23H	1F	1E	1D	1C	1B	1A	19	18
22H	17	16	15	14	13	12	11	10
21H	0F	0E	0D	0C	0B	0A	09	08
20H	07	06	05	04	03	02	01	00

注意： 一个单元地址对应8个位地址

MSB——Most Significant Bit (最高有效位)

LSB ——Least Significant Bit (最低有效位)

(3) 特殊功能寄存器

寄存器	位地址 / 位定义								地址
B	F7	F6	F5	F4	F3	F2	F1	F0	F0H
ACC	E7	E6	E5	E4	E3	E2	E1	E0	E0
PSW	D7	D6	D5	D4	D3	D2	D1	D0	D0H
	CY	AC	F0	RS1	RS0	OV	/	P	
IP	BF	BE	BD	BC	BB	BA	B9	B8	B8H
P3	B7	B6	B5	B4	B3	B2	B1	B0	B0H
	P3. 7	P3. 6	P3. 5	P3. 4	P3. 3	P3. 2	P3. 1	P3. 0	
IE	AF	AE	AD	AC	AB	AA	A9	A8	A8H
P2	A7	A6	A5	A4	A3	A2	A1	A0	A0H
	P2. 7	P2. 6	P2. 5	P2. 4	P2. 3	P2. 2	P2. 1	P2. 0	
SBUF									99H

SCON									98H
P1	97	96	95	94	93	92	91	90	90H
	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	
TH1									8DH
TH0									8CH
TL1									8BH
TLO									8AH
TMOD	GATE	C/T	M1	MO	GATE	C/T	M1	MO	89H
TCON	8F	8E	8D	8C	8B	8A	89	88	88H
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
PCON	SMOD	/	/	/	GF1	GF0	PD	IDL	87H
DPH									83H
DPL									82H
SP									81H
P0	87	86	85	84	83	82	81	80	80H
	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	

MCS-51三总线结构

1、数据总线 (DB)

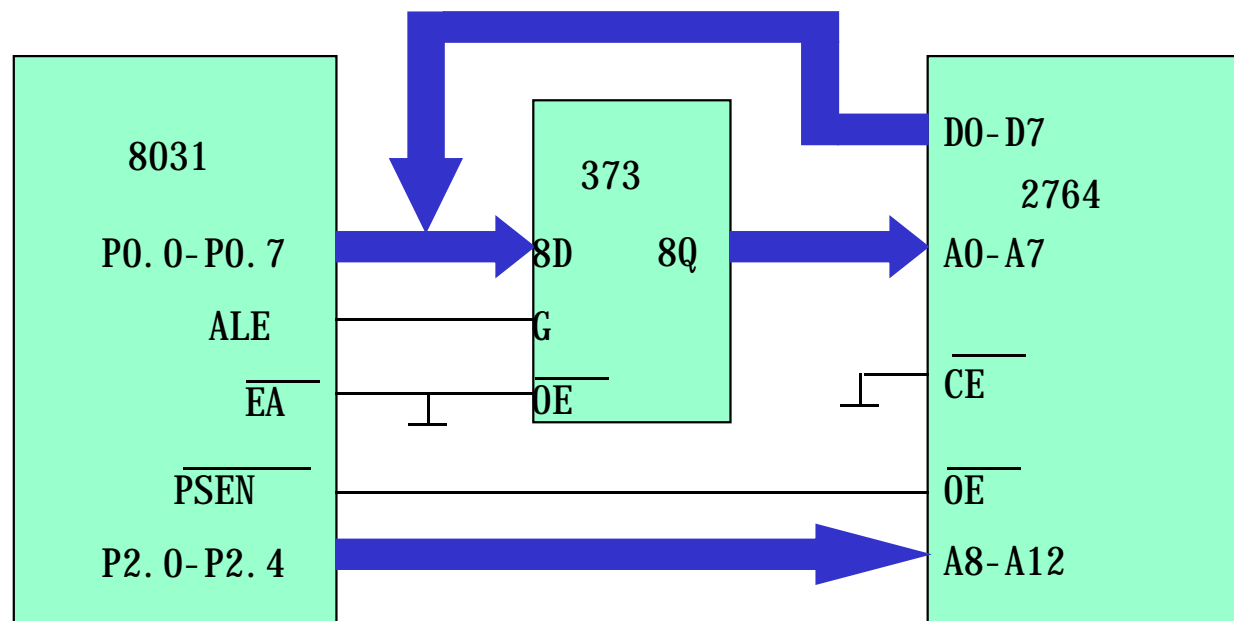
由P0口提供8位数据

2、地址总线 (AB)

由P2, P0提供16位地址

3、控制总线 (CB)

由P3口和RST、EA、ALE、PSEN组成



哈佛结构 VS 冯·诺伊曼结构

- 1、哈佛结构是一种将程序指令存储和数据存储分开的存储器结构。中央处理器首先到程序指令存储器中读取程序指令内容，解码后得到数据地址，再到相应的数据存储器中读取数据，并进行下一步的操作（通常是执行）。程序指令存储和数据存储分开。如摩托罗拉公司的MC68系列、Zilog公司的Z80系列、ATMEL公司的AVR系列和安谋公司的ARM9、ARM10和ARM11，51单片机也属于哈佛结构。
- 2、冯·诺伊曼结构也称普林斯顿结构，是一种将程序指令存储器和数据存储器合并在一起的存储器结构。程序指令存储地址和数据存储地址指向同一个存储器的不同物理位置，因此程序指令和数据的宽度相同，如英特尔公司的8086中央处理器的程序指令和数据都是16位宽。

哈佛结构 VS 冯·诺伊曼结构

3、在通用计算机系统中，应用程序的多样性使得计算机要不断地变化所执行的代码的内容，并且频繁地对数据与代码占有的存储器进行重新分配，这种情况下，冯·诺依曼结构占有绝对优势，因为统一编址可以最大限度地利用资源，而哈佛结构的计算机若应用于这种情形下则会对存储器资源产生理论上最大可达50%的浪费，这显然是不合理的。

但是在嵌入式应用中，系统要执行的任务相对单一，程序一般是固化在硬件里。当然这时使用冯·诺依曼结构也完全可以，代码区和数据区在编译时一次性分配好了而已，但是其灵活性得不到体现，所以现在大量的单片机也还在沿用冯·诺依曼结构，如TI的MSP430系列、Freescal e的HCS08系列等。

哈佛结构 VS 冯·诺伊曼结构

4、嵌入式计算机在工作时与通用计算机有着一些区别：嵌入式计算机在工作期间的绝大部分时间是无人值守的，而通用计算机工作期间一般是有人操作的；嵌入式计算机的故障可能会导致灾难性的后果，而通用计算机一般就是死死机，重新启动即可。这两点决定了对嵌入式计算机的一个基本要求：可靠性。

使用冯·诺依曼结构的计算机，程序空间不封闭，程序空间的数据在运行期理论上可以被修改，此外程序一旦跑飞也有可能运行到数据区。虽然都是一些不常见的特殊情况下，但是看看哈佛结构的计算机在这些情况下是怎样的：基于哈佛结构的处理器如MCS-51，不需要对代码段进行写操作的指令，所以不会有代码区被改写的问题；程序只能在封闭的代码区中运行，不可能跑到数据区，这也使得跑飞的几率减少并且跑飞后的行为有规律（数据区的数据是不断变化的而代码区是不变的）。

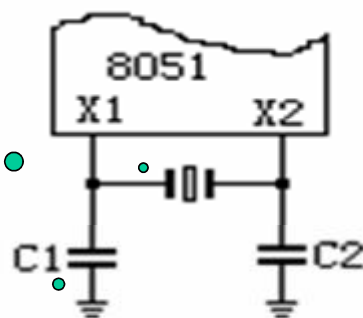
2.5 CPU的时钟电路和时序

我们知道了单片机怎样取指、执指，即怎样运行程序了。那么怎样才能保证CPU有序的工作？这就必须提到单片机的两个非常重要的外围电路：单片机的时钟电路和复位电路。

时钟电路：

1.2~12 MHz

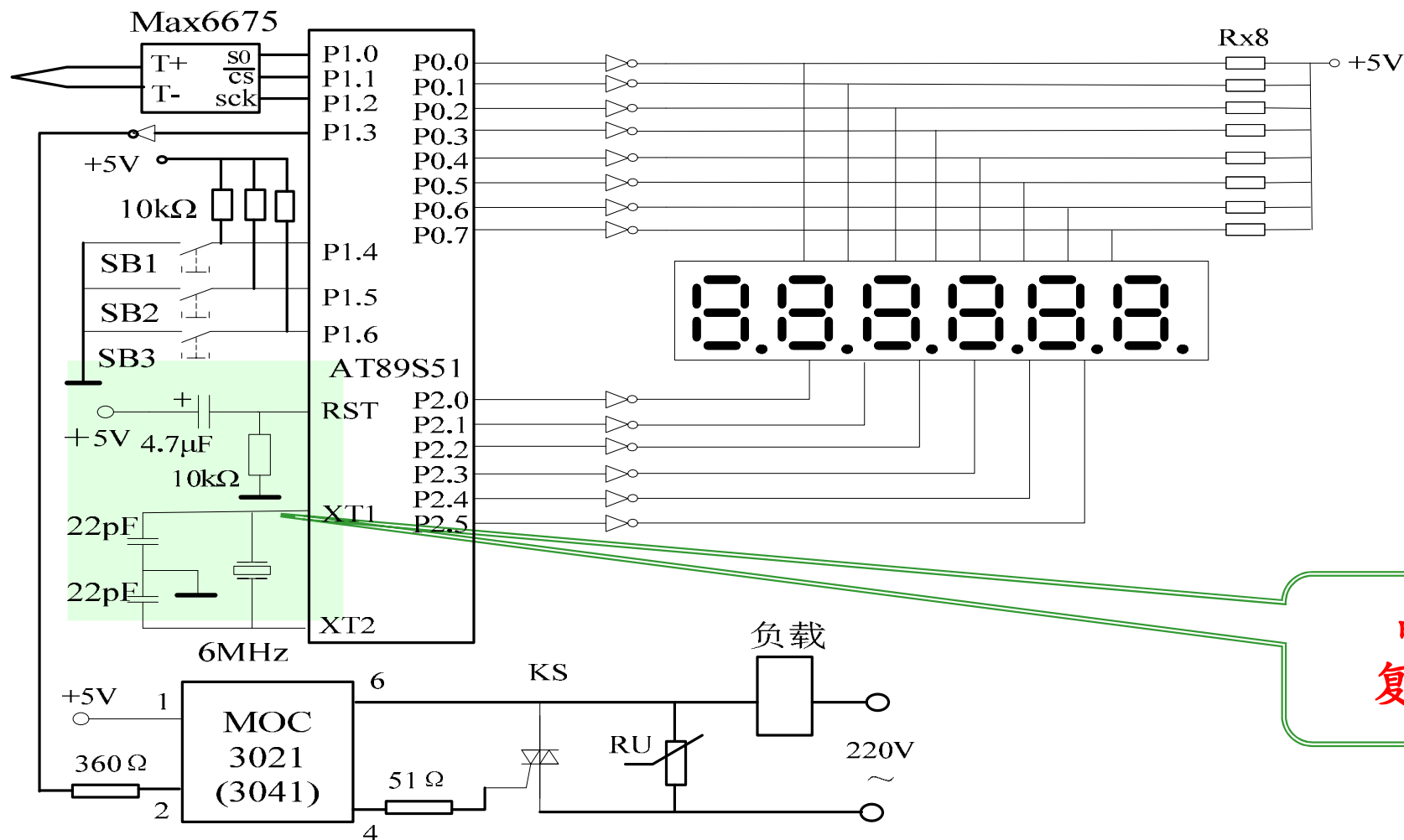
C1和C2取30 pF
左右



内部时钟方式



外部时钟方式



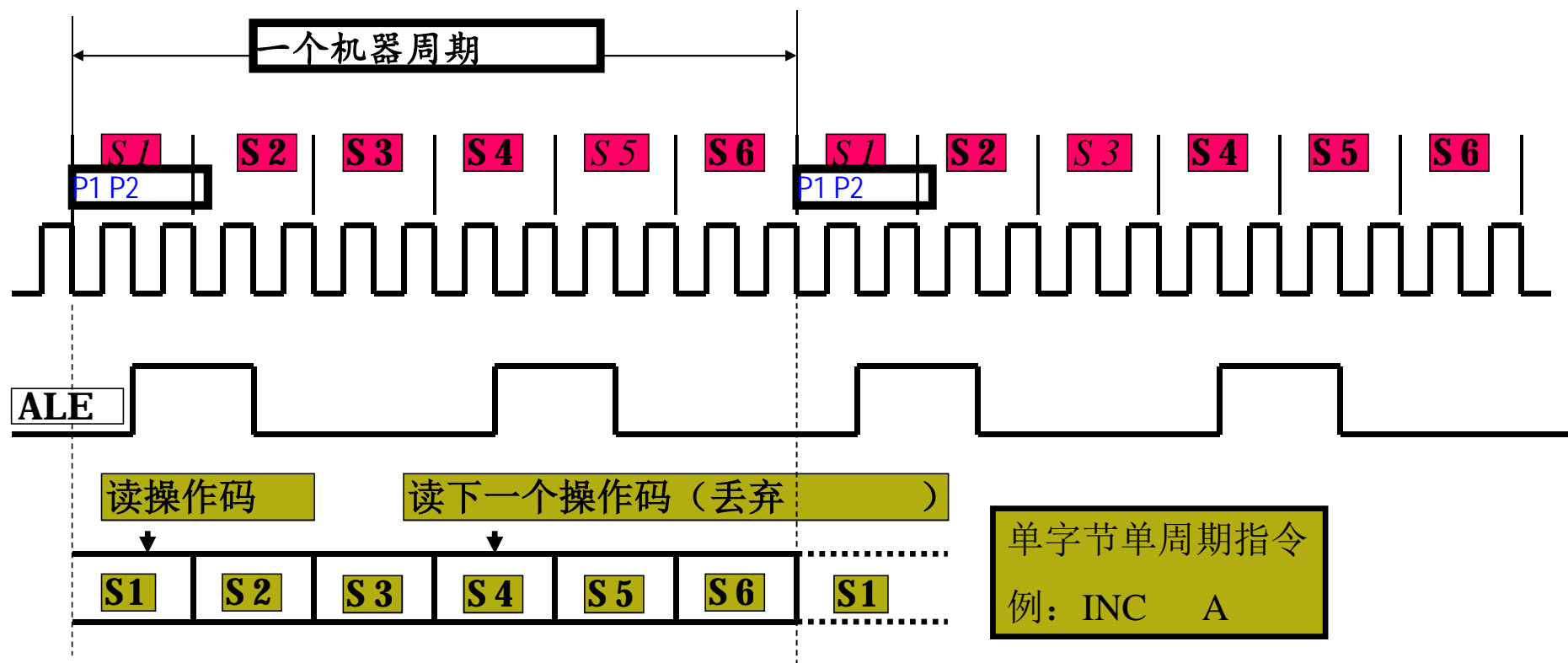
晶振、复位电路

【振荡周期】：单片机外接石英晶体振荡器的周期。如外接石英晶体的频率若为12MHz，则其振荡周期就是1/12微秒。

【状态周期】：单片机完成一个最基本的动作所需的时间周期。如扫描一次定时器T0引脚状态所需要的时间。一个状态周期=2个振荡周期。

【机器周期】：单片机完成一次完整的具有一定功能的动作所需的时间周期。如一次完整的读操作或写操作对应的时间。一个机器周期=6个状态周期。

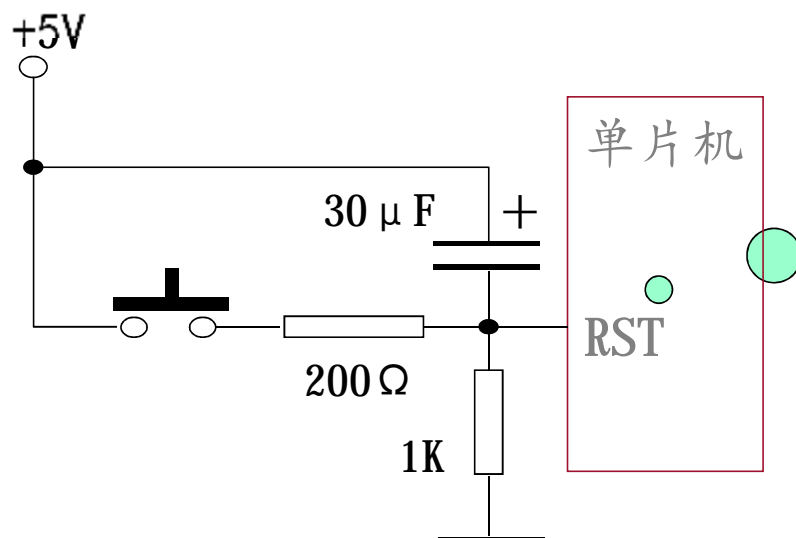
【指令周期】：执行完某条指令所需要的时间周期，一般需要1~4个机器周期，如MUL AB指令是四机器周期指令。一个指令周期=1~4个机器周期。



单周期单字节指令时序

2.6 单片机工作方式

2.6.1 复位与复位电路



时钟电路开始工作后，在RESET引脚上出现24个时钟周期以上的高电平，系统即初始复位。

单片机工作时，除了需要时钟支持外，还必须有一个初始状态，即单片机的复位状态。复位不改变RAM（包括工作寄存器R0-R7）的状态，89C51的初始状态如下表：

特殊功能寄存器	初始态	特殊功能寄存器	初始态
ACC	00H	B	00H
PSW	00H	SP	07H
DPH	00H	TH0	00H
DPL	00H	TL0	00H
IP	xxx00000B	TH1	00H
IE	0xx00000B	TL1	00H
TMOD	00H	TCON	00H
SCON	xxxxxxxxB	SBUF	00H
P0-P3	1111111B	PCON	0xxxxxxxxB

2.6.2 80C51系列单片机的低功耗方式

1、 电源控制寄存器PCON

SMOD				GF1	GF0	PD	IDL
------	--	--	--	-----	-----	----	-----

IDL=0, PD=0, 正常工作方式。

PD=1, 进入掉电工作方式;

IDL=1, PD=0, 进入待机工作方式。

GF1、GF0为用户使用的通用标志。

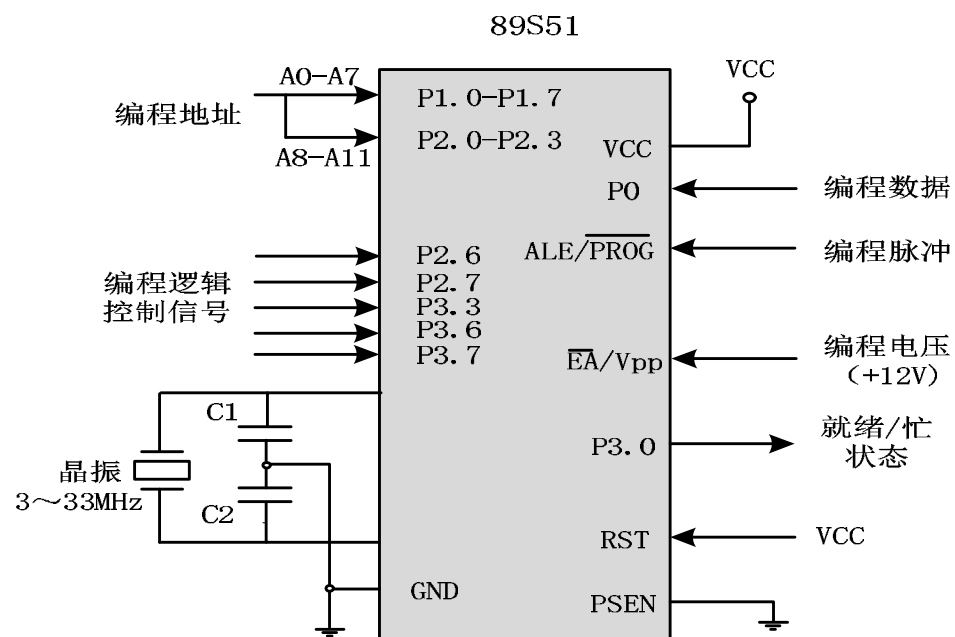
2、 待机方式

- 在待机工作方式CPU停止工作，但振荡器与中断源继续工作。
- 通过字节操作使IDL=1, 单片机进入待机工作方式。
- 一次硬件复位或一次被开放的中断源的中断申请可复位IDL，使单片机结束待机。

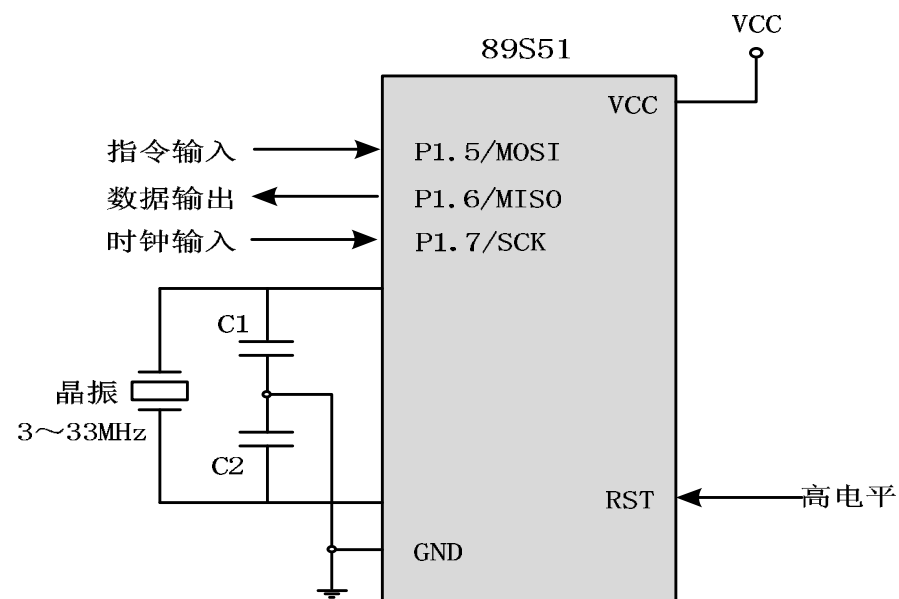
3、掉电方式

- 在掉电工作方式CPU与振荡器都停止工作，仅片内RAM部分维持供电。掉电期间，电压可低至2V。
- 通过字节操作使PD=1, 单片机进入掉电工作方式。
- 只有在电源电压正常时的硬件复位能使单片机结束掉电状态。

2.6.3 ISP编程工作方式



并行编程方式



ISP编程方式

一、并行编程方式

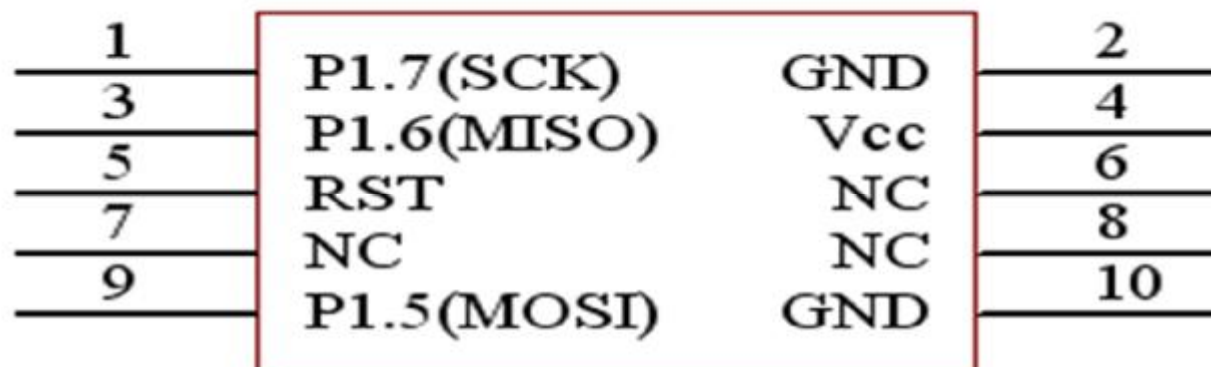
并行编程方式需要借助编程器，在EA/VPP（31引脚）上使用高电压（+12V）和协调的控制信号进行编程。

缺点：一是增加了硬件成本；二是在实际应用开发过程中，需要从目标系统电路板上拔下芯片，编程后再插上。

二、ISP编程方式（在线编程方式）

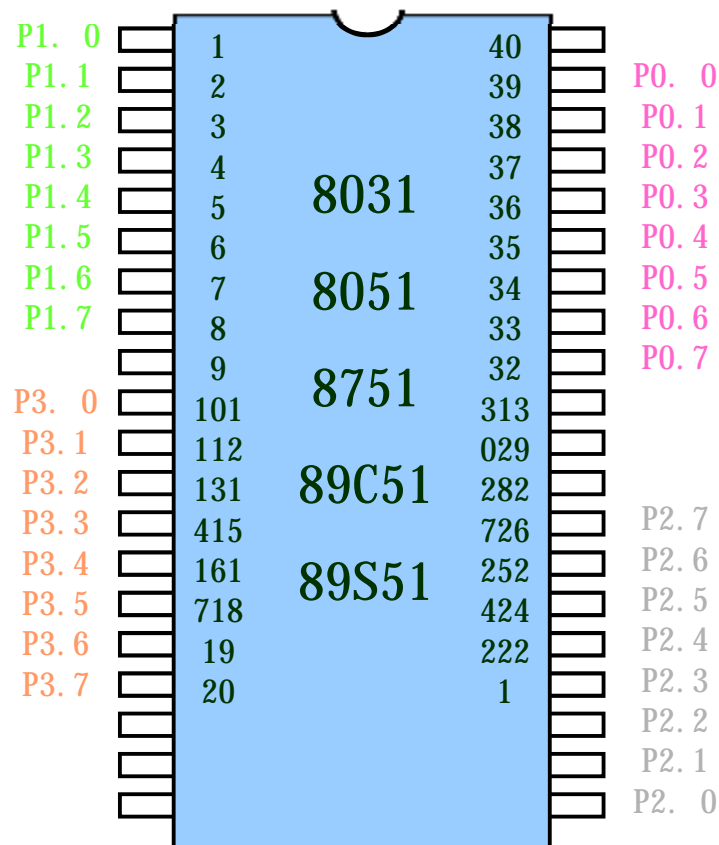
将RST 接至Vcc，程序代码存储阵列可通过串行ISP 接口进行编程，89S51的串行接口包含时钟输入SCK 线（P1.7脚）、MDSI数据输入线（P1.5脚）和MISO数据输出线（P1.6脚）。

ISP电缆和AT89S51系列单片机通过一个10针的IDC口进行连接，IDC-10的引脚定义如图所示。



Atmel公司的ISP软件可在Atmel公司的网站上免费下载，软件的使用可参阅软件的帮助说明。

2.7 输入/输出端口结构



特点:

□ 4个8位并行I/O口: P0, P1, P2, P3;

□ 均可作为双向I/O端口使用。输入时可以缓冲, 输出时可以锁存。

P0: 访问片外扩展存储器时, 复用为低8位地址线和数据线

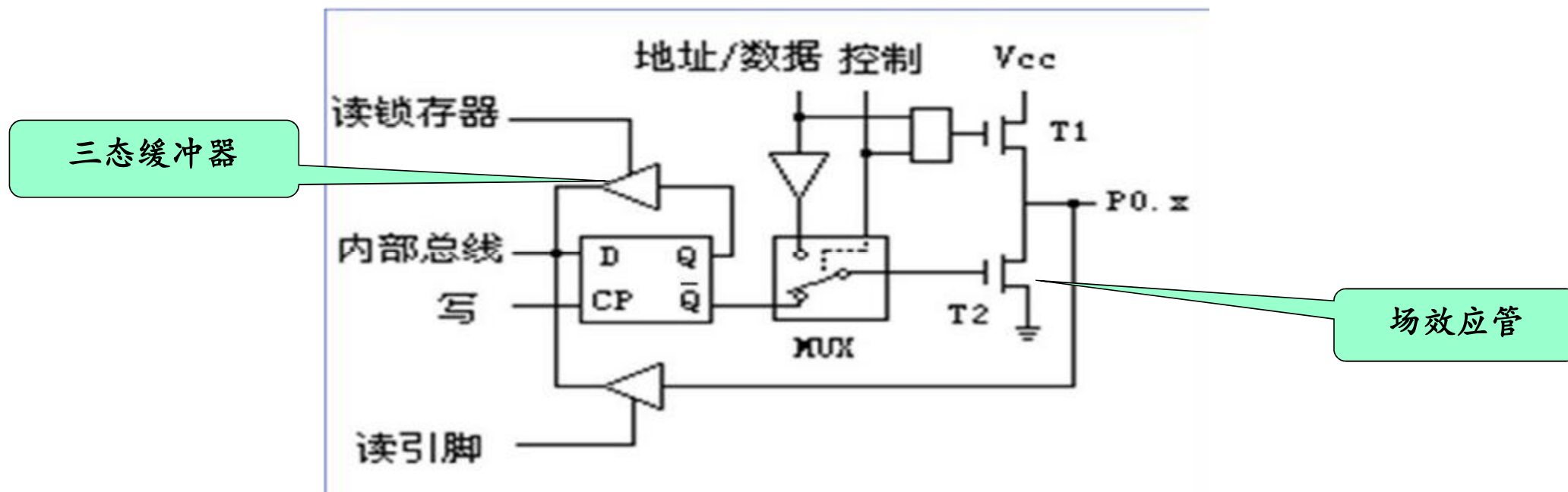
P2: 高8位地址线。

P1: 双向I/O端口

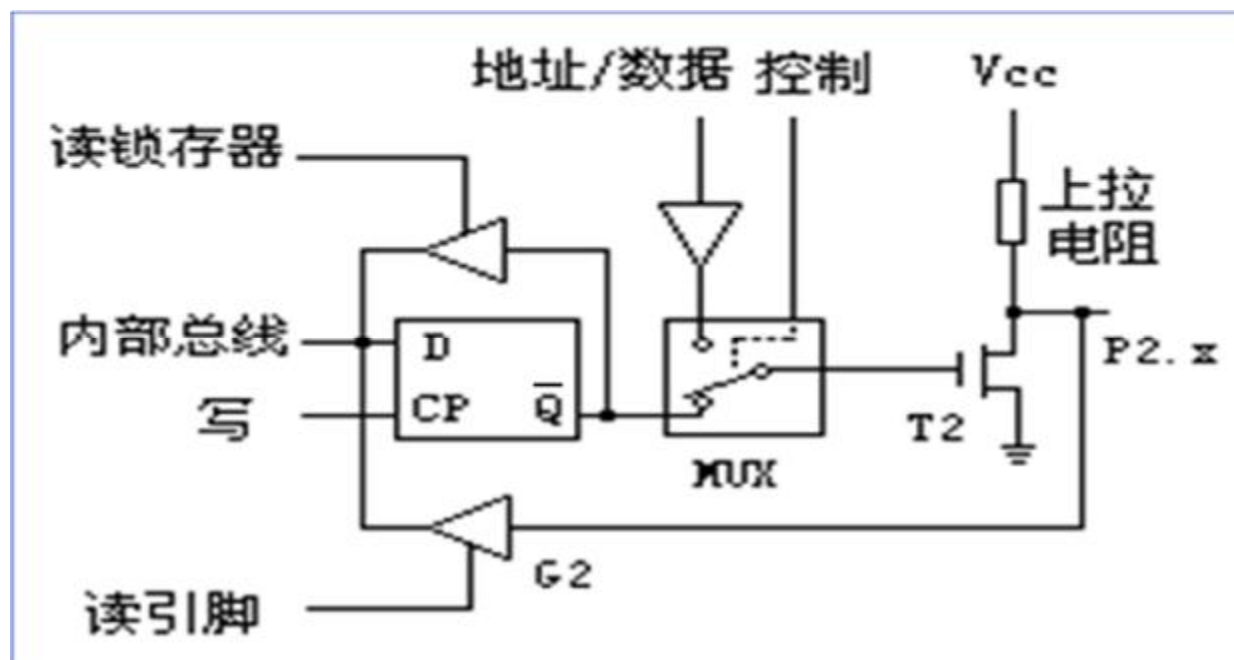
P3: 第二功能

片外扩展存储器时P0、P2口自动作为总线使用

(1) P0和P2口

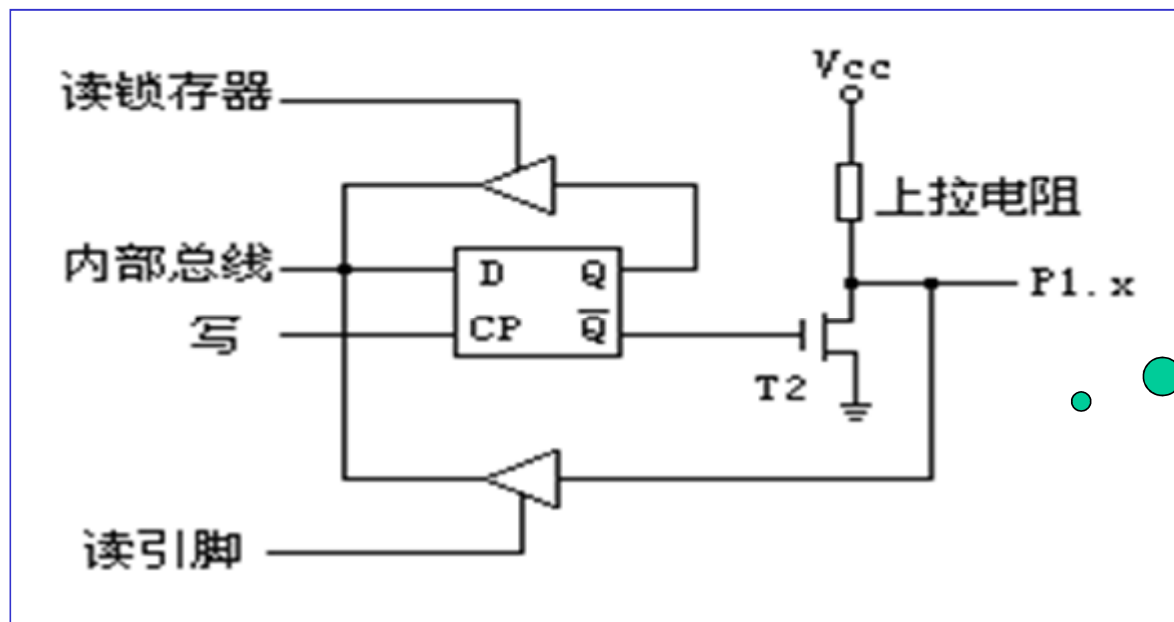


P0口既可作为I/O口，也可用作地址/数据总线，但二者不能兼用，用作I/O口时，必须在管脚上接上拉电阻。用作地址/数据总线时，地址和数据采取分时复用方式。



P2口既可作为I/O口，也可用作高8位地址总线。

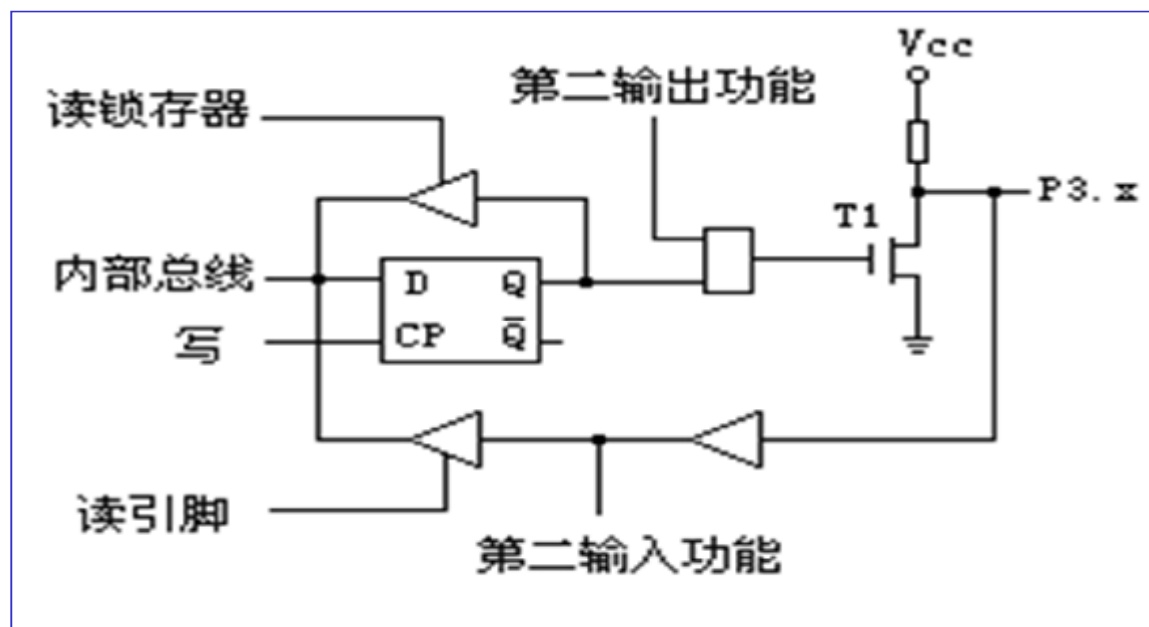
(2) P1口



P1口只用作普通
输入输出口

注意：P0、P1、P2、P3口作输入时，为防止口锁存器对输入口线造成影响必须先往口锁存器写1。

(3) P3口



P3.0 串行输入口 (RXD)

P3.1 串行输出口 (TXD)

P3.2 外中断0 (INT0)

P3.3 外中断1 (INT1)

P3.5 定时/计数器1的外部输入口 (T1)

P3.6 外部数据存储器写选通 (WR)

P3.4 定时/计数器0的外部输入口 (T0)

P3.7 外部数据存储器读选通 (RD)

读端口与读引脚：

- 读端口：读锁存器Q端的状态。适应对口进行“读—修改—写”操作指令的需要。如：ANL P0, A
- 读引脚：读输入口线的状态。

教学案例：烘箱温度控制系统任务及单片机选型

(1) 控制对象

- ① 额定工作电压：AC220V/50 Hz。
- ② 额定功率：1 kW。
- ③ 烘箱容积：20 L。

(2) 主要功能

- ① 温度控制范围与精度：控制范围20~240 °C，控制精度±10 °C。
- ② 温度显示：数码管显示设定烘箱温度和实际烘箱温度，温度测量分辨率1 °C。
- ③ 温度设定：通过按键调节设定温度，设定步进1 °C。

教学案例：烘箱温度控制系统任务及单片机选型

(3) 单片机型号选择

温度控制系统惯性大，控制周期长，运算工作量不大，选择普通的8位单片机即可满足要求。由于教学过程中主要以AT89S51/S52单片机为主来介绍单片机控制系统的硬件设计与软件设计，因此，教学案例烘箱温度控制系统的控制核心选用单片机AT89S51。



02

END



THANKS



《单片机与接口技术》

主讲人：李刚