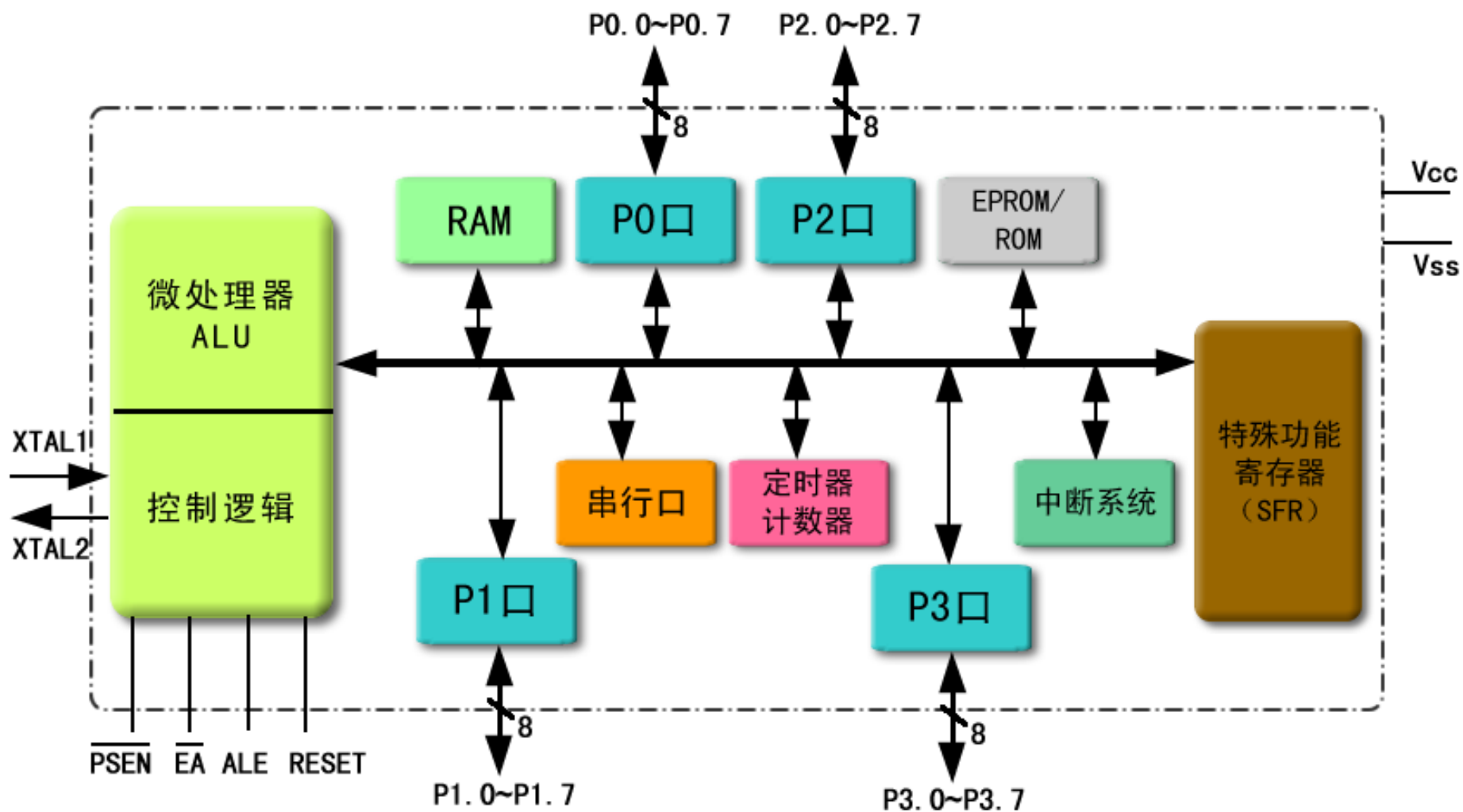


# 复习

## MCS-51单片机功能框图



# 单片机的指令系统

# 指令系统简介

- 指令——计算机能够识别和执行的命令。
- 指令系统——指令的集合或全体，因机器而异。
- 指令的表示形式
  - 二进制形式——直接为CPU执行
  - 16进制形式——阅读和书写
  - 汇编形式——编写程序

二进制形式	16进制形式(H)	汇编形式
<u>0111</u> 0100 0000 <u>1010</u>	<u>7</u> 40 <u>A</u> H	MOV A, #0AH
00100100 00000100	2404H	ADD A, #04H
10000000 11111110	80FEH	SJMP \$

# 主要学习内容

- 1、**汇编语言指令格式**
- 2、**MCS-51指令的寻址方式**
- 3、**MCS-51指令系统分析**

# 1、汇编语言指令格式

[标号:]	操作码	操作数1, 操作数2	[; 注释]
		(目标操作数) (源操作数)	
例: LOOP:	MOV	A, #40H	; 取参数
指令的 符号地址	指明 指令功能	指令操作对象: 数据、地 址、寄存器名及约定符号	说明指令在程 序中的作用

- **操作码和操作数是指令主体**
- 换行表示一条指令结束
- **常用操作码**

MOV	move	传送	ADD	rotate right	加法
XCH	exchange	交换	MUL	multiply	乘法
ANL	and logic	与逻辑运算	SJMP	short jump	短跳转
XRL	exclusive or	异或运算	RET	return	子程序返回

# 1、汇编语言指令格式

[标号:]	操作码	操作数1, 操作数2	[; 注释]
		(目标操作数) (源操作数)	
例: LOOP:	MOV	A, #40H	; 取参数
指令的 符号地址	指明 指令功能	指令操作对象: 数据、地 址、寄存器名及约定符号	说明指令在程 序中的作用

## • MSC-51指令中操作数的符号说明

Rn	R0~R7中的一个	bit	8位位地址
#data	8位立即数	Addr11	11位目标地址
#data16	16位立即数	addr16	16位目标地址
direct	8位直接地址	rel	8位地址偏移量
@Ri	R0或R1, 可以间接寻址	\$	当前指令的地址
@DPTR	可按DPTR中地址对外部存储器寻址		

## 2、MCS-51指令的寻址方式

■ **寻址方式**：寻找操作数或操作数的地址

■ **七种寻址方式（源操作数）**：

地址	数据
41H	67H
R0 → 40H	34H

- **立即寻址**                      MOV A, #40H                      ; A ← 40H
- **直接寻址**                      MOV A, 40H                      ; A ← (40H) = 34H
- **寄存器寻址**                      MOV A, R0                      ; A ← (R0) = 40H
- **寄存器间接寻址**              MOV A, @R0                      ; A ← ((R0)) = 34H
- **变址寻址(基址寄存器+变址寄存器间接寻址)**
- **相对寻址**
- **位寻址**

■ (51中目标操作数寻址方式：寄存器、直接、间接、位)

# 3、MCS-51指令系统分析

## ■指令按照功能分类

- 数据传送

- 数据操作

- 转移指令



### 3、MCS-51指令系统分析-数据传送

■ **功能：**实现寄存器、存储器之间的数据传送。

一. **内部传送指令：**片内数据存储器数据传送。

二. **外部传送指令：**片外数据存储器数据传送。

三. **交换指令：**片内数据存储器数据传送。

四. **堆栈操作指令：**片内数据存储器数据传送。

五. **查表指令：**程序存储器数据传送。

### 3、MCS-51指令系统分析-数据传送

#### ■ 内部传送指令：实现片内数据存储器中数据传送。

- 指令格式：MOV 目标操作数，源操作数
- 寻址方式：立即寻址、直接寻址、寄存器寻址、寄存器间址。

MOV A, #d ;  $A \leftarrow d$

MOV A, @Ri ;  $A \leftarrow ((Ri))$ ,  $Ri = R_0, R_1$

MOV DPTR, #d1d2 ;  $DPTR \leftarrow d1d2$

DPTR: 16  
位数据指针

#### ■ 外部传送指令：实现片外数据存储器 and A累加器之间的数据传送。

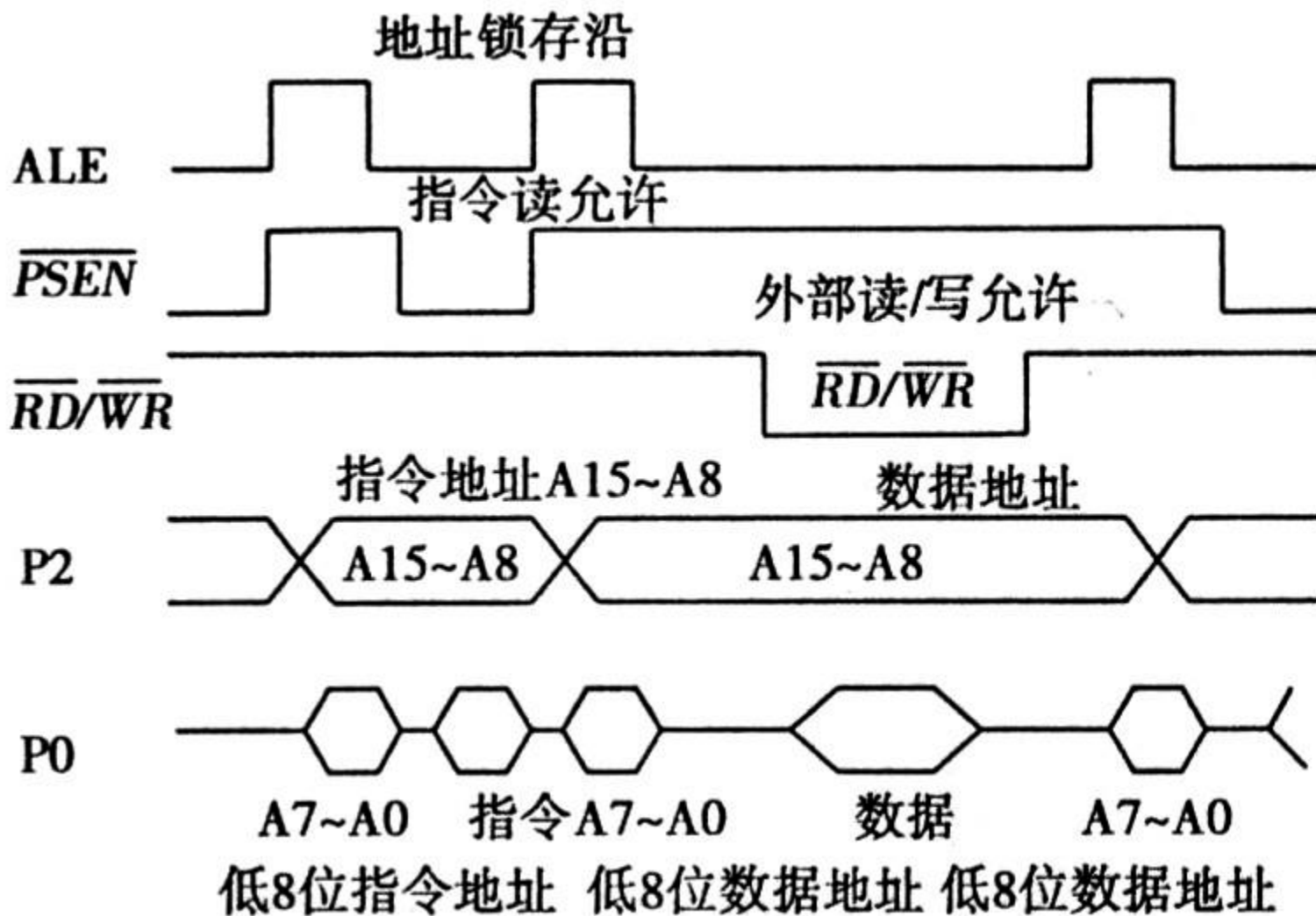
- 指令格式：MOVX 目标操作数，源操作数
- 寻址方式：片外数据存储器用寄存器间址方式。

MOVX A, @DPTR ;  $A \leftarrow ((DPTR))$

MOVX @DPTR, A ;  $(DPTR) \leftarrow (A)$

为什么对DPTR的数据  
传送使用内部传送指令？

### 3、MCS-51指令系统分析-数据传送



MOVX执行时序图

### 3、MCS-51指令系统分析-数据操作

■ **功能：**完成片内 RAM 和 A 中数据的加减乘除运算。

- **加法指令** **ADD A, R2** ;  $A \leftarrow (A) + (R2)$
- **减法指令** **SUB A, R2** ;  $A \leftarrow (A) - (R2)$
- **增量指令** **INC Rn** ;  $Rn \leftarrow (Rn) + 1$
- **减量指令** **DEC Rn** ;  $Rn \leftarrow (Rn) - 1$
- **双操作数逻辑运算指令(对位逻辑运算)** A=0000 1111
  - 与** **ANL** : **ANL A, 1111 1111** ; A=0000 1111
  - 或** **ORL** : **ORL A, 1111 1111** ; A=1111 1111
  - 异或** **XRL** : **XRL A, 1111 1111** ; A=1111 0000
- **位设置指令** **SETB、CLR**
  - SETB P0.1** ; 将P0口的第二位置1

### 3、MCS-51指令系统分析-转移指令

■ **功能**：通过改写PC的当前值，从而改变CPU执行程序  
的顺序，使**程序发生跳转**。

• **无条件转移**：执行无条件转移指令，程序无条件转移到指定处。

**长转移LJMP、绝对转移AJMP、短转移SJMP**

LJMP NEXT;      AJMP NEXT;      SJMP NEXT; NEXT是标号

• **条件转移**：指令中给出转移条件，执行指令时，先测试条件，  
若满足条件，则程序发生转移，否则，仍顺序执行程序。

**判零转移JZ、JNZ**

**判Cy转移JC、JNC**

**判位转移JB、JNB**

**判位清零转移JBC**

**比较不相等转移CJNE**

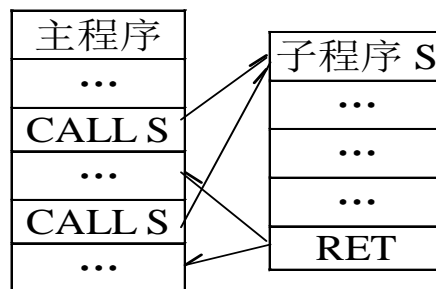
**循环转移DJNZ**

• **子程序调用和返回指令**：转移时，先用堆栈保存当前地址

**长调用LCALL**

**绝对调用ACALL**

**子程序返回RET**



# 小结

- 1、汇编语言指令格式
- 2、MCS-51指令的寻址方式
- 3、MCS-51指令系统分析

**以了解为主，将在后续课程中介绍如何应用。**