

项目一

单片机硬件系统

任务 单片机控制单个 LED 发光二极管闪烁

任务目标

- 了解单片机及单片机应用系统。
- 理解单片机内部组成及信号引脚。
- 掌握 MCS-51 内部数据存储器及内部程序存储器。
- 了解并行输入/输出口电路结构。
- 了解单片机复位电路。
- 理解单片机工作过程。

任务要求

通过单片机 P1.0 引脚控制发光二极管，使二极管亮一会灭一会，交替闪烁。

相关知识点

一、概述

单片微型计算机（Single Chip Microcomputer）简称单片机，是指集成在一块芯片上的计算机，是典型的嵌入式微控制器（Microcontroller Unit），常用英文字母的缩写 MCU 表示单片机，它最早是被用在工业控制领域。单片机具有结构简单、控制功能强、可靠性高、体积小、价格低等优点，在许多行业都得到了广泛应用。从航天航空、地质石油、冶金采矿、机械电子、轻工纺织到机电一体化设备、邮电通信、日用设备和器械等，单片机都发挥了巨大作用。

1. 单片机及单片机应用系统

（1）微型计算机及微型计算机系统。

微型计算机（Microcomputer）简称微机，是计算机的一个重要分支。人们通常按照计算机的体积、性能和应用范围等条件，将计算机分为巨型机、大型机、中型机、小型机和微型机等。微型计算机不但具有其他计算机快速、精确、程序控制等特点，最突出的优点是它体积小、重量轻、功

耗低、价格便宜。个人计算机（Personal Computer）简称 PC 机，是微型计算机中应用最为广泛的一种，也是近年来计算机领域中发展最快的一个分支，由于 PC 机在性能和价格方面适合个人用户购买和使用，目前它已经像普通家电一样深入到了家庭和社会生活的各个方面。

微型计算机系统由硬件系统和软件系统两大部分组成。

硬件系统是指构成微机系统的实体和装置，通常由运算器、控制器、存储器、输入接口电路和输入设备、输出接口电路和输出设备等组成。其中，运算器和控制器一般做在一个集成芯片上，统称中央处理单元（Central Processing Unit, CPU），是微机的核心部件，配上存放程序和数据的存储器、输入/输出（Input/Output, I/O）接口电路及外部设备即构成微机的硬件系统。

软件系统是指微机系统所使用的各种程序的总体。软件的主体驻留在存储器中，人们通过它对整机进行控制并与微机系统进行信息交换，使微机按照人的意图完成预定的任务。

软件系统与硬件系统共同构成实用的微机系统，两者是相辅相成、缺一不可的。

微型计算机系统组成示意图如图 1.1.1 所示。

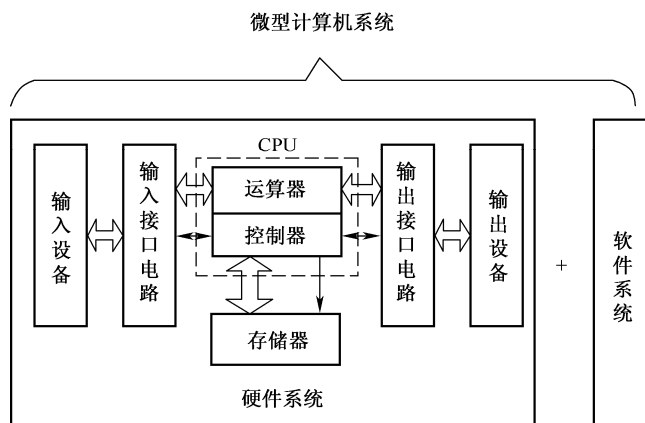


图 1.1.1 微型计算机系统组成示意图

下面对组成计算机的 5 个基本部件进行简单说明。

- 运算器：是计算机的运算部件，用于实现算术和逻辑运算，计算机的数据运算和处理都在这里进行。
- 控制器：是计算机的指挥控制部件，使计算机各部分能自动协调地工作。运算器和控制器是计算机的核心部分，常把它们合在一起称为中央处理器，简称 CPU。
- 存储器：是计算机的记忆部件，用于存放程序和数据。存储器又分为内存储器和外存储器。
- 输入设备：用于将程序和数据输入到计算机中，如键盘。
- 输出设备：用于把计算机数据计算或加工的结果以用户需要的形式显示或保存，如显示器、打印机。

通常把外存储器、输入设备和输出设备合在一起称为计算机的外部设备，简称外设。

（2）单片微型计算机。

单片微型计算机是指集成在一个芯片上的微型计算机，也就是把组成微型计算机的各种功能部件，包括 CPU（Central Processing Unit）、随机存取存储器 RAM（Random Access Memory）、只读

存储器 ROM (Read-only Memory)、基本输入/输出 (Input/Output) 接口电路、定时器/计数器等部件制作在一块集成芯片上, 构成一个完整的微型计算机, 从而实现微型计算机的基本功能。单片机内部结构示意图如图 1.1.2 所示。

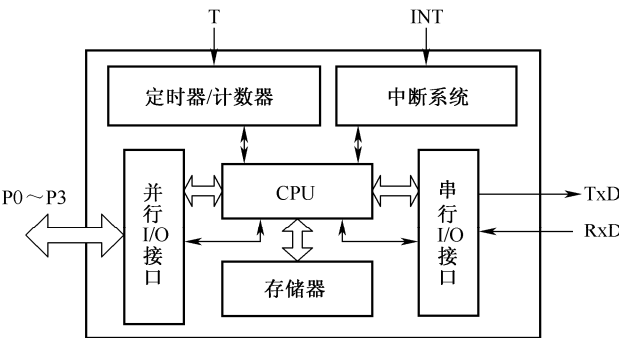


图 1.1.2 单片机内部结构示意图

单片机实质上是一个硬件的芯片, 在实际应用中, 通常很难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、外部设备等硬件和软件才能构成一个单片机应用系统。

(3) 单片机应用系统及组成。

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。所以说, 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, 软件是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。单片机应用系统的组成如图 1.1.3 所示。

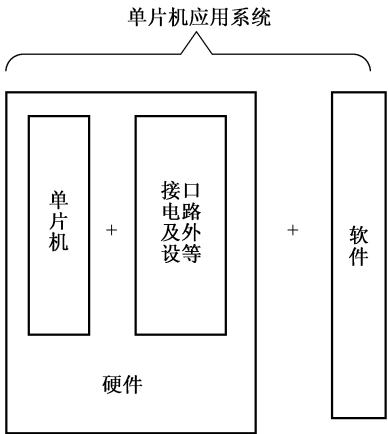


图 1.1.3 单片机应用系统的组成

由此可见, 单片机应用系统的设计人员必须从硬件和软件两个角度来深入了解单片机, 并能够将二者有机结合起来, 才能形成具有特定功能的应用系统或整机产品。

自从 1974 年美国 Fairchild 公司研制出第一台单片机 F8 之后, 迄今为止, 单片机经历了由 4 位机到 8 位机、16 位机、32 位机的发展过程。单片机制造商有很多, 主要有美国的 Intel、Motorola、ATMEL 等公司。目前, 单片机正朝着高性能、多品种方向发展, 近年来 32 位单片机已进入了实

用阶段。但是由于 8 位单片机从性能价格比上占有优势,而且 8 位增强型单片机在速度和功能上向现在的 16 位单片机挑战,因此在未来相当长的时期内 8 位单片机仍是单片机的主流机型。

2. 51 单片机系列

尽管各类单片机很多,但无论是从世界范围还是从国内范围来看,使用最为广泛的应属 MCS-51 单片机。基于这一事实,本书以应用最为广泛的 MCS-51 系列 8 位单片机(8031、8051 等)为研究对象,介绍单片机的硬件结构、工作原理及应用系统的设计。

MCS-51 单片机系列共有十几种芯片,如表 1.1.1 所示。

表 1.1.1 MCS-51 系列单片机分类表

子系列	片内 ROM 形式			片内 ROM 容量	片内 RAM 容量	寻址范围	I/O 特性			中断源
	无	ROM	EPROM				计数器	并行口	串行口	
51 子系列	8031	8051	8751	4KB	128B	2×64KB	2×16	4×8	1	5
	80C31	80C51	87C51	4KB	128B	2×64KB	2×16	4×8	1	5
52 子系列	8032	8052	8752	8KB	256B	2×64KB	3×16	4×8	1	6
	80C32	80C52	87C52	8KB	256B	2×64KB	3×16	4×8	1	6

表中列出了 MCS-51 单片机系列的芯片型号以及它们的技术性能指标,使我们对其基本情况有一个概括的了解。下面就在这个表的基础上对 MCS-51 系列单片机进行说明。

(1) 51 子系列和 52 子系列。

MCS-51 系列又分为 51 和 52 两个子系列,并以芯片型号的最末位数字作为标志。其中 51 子系列是基本型,而 52 子系列则属增强型。52 子系列功能增强的具体方面从表 1.1.1 中可以看出:

- 片内 ROM 从 4KB 增加到 8KB。
- 片内 RAM 从 128 字节增加到 256 字节。
- 定时器/计数器从 2 个增加到 3 个。
- 中断源从 5 个增加到 6 个。

(2) 片内 ROM 存储器配置形式。

MCS-51 单片机片内程序存储器有多种配置形式,即掩膜 ROM、EPROM、EEPROM 和 Flash ROM。它们各有特点,也各有其适用场合,在使用时应根据需要进行选择。一般情况下,片内带掩膜型 ROM 适用于定型大批量应用产品的生产;片内带 EPROM 型的单片机适合于研制产品样机;外接 EPROM 的方式适用于研制新产品;Intel 公司推出的片内带 EEPROM 型的单片机,可以在线写入程序;目前 Flash ROM 使用较广,闪存是电子可擦除只读存储器(EEPROM)的变种,闪存与 EEPROM 不同的是,它能在字节水平上进行删除和重写而不是整个芯片擦写,这样闪存就比 EEPROM 的更新速度快。

二、51 单片机结构

尽管单片机比较简单,但要按 5 个基本组成部件来讲单片机的硬件结构和原理也将是一件十分复杂的事。其实也没有这种必要。因此,通常讲述单片机结构原理时,总是从实际需要出发,只介绍与程序设计和系统扩展应用有关的内容。

1. 51 单片机的内部组成及信号引脚

51 单片机的典型芯片是 8031、8051、8751。8051 内部有 4KB ROM,8751 内部有 4KB EPROM,

8031 片内无 ROM，除此之外，三者的内部结构及引脚完全相同。因此以 8051 为例来说明本系列单片机的内部组成及信号引脚。

(1) 8051 单片机的基本组成。

8051 单片机的基本组成如图 1.1.4 所示。

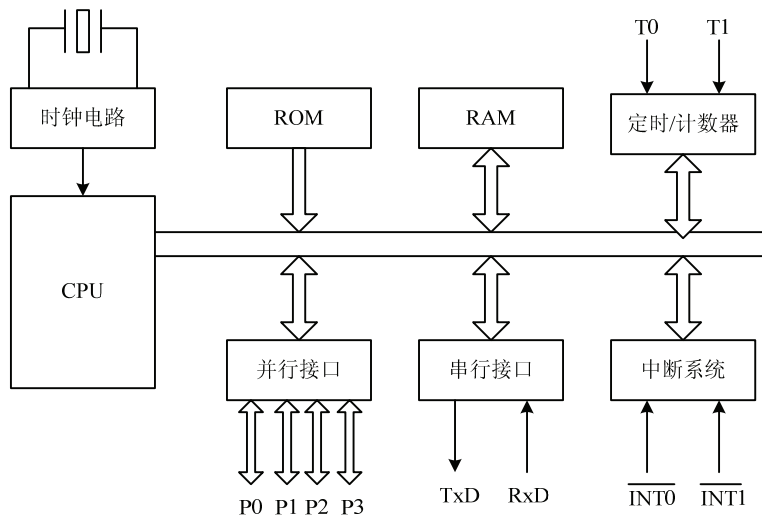


图 1.1.4 MCS-51 单片机结构框图

1) 中央处理器 (CPU)。

中央处理器是单片机的核心，完成运算和控制功能。MCS-51 的 CPU 能处理 8 位二进制数或代码。

2) 内部数据存储器 (内部 RAM)。

8051 芯片中共有 256 个 RAM 单元，但后 128 单元被专用寄存器占用，能作为寄存器供用户使用的只是前 128 单元，用于暂存中间数据，可读可写，掉电后数据会丢失。因此通常所说的内部数据存储器就是指前 128 单元，简称内部 RAM。

3) 内部程序存储器 (内部 ROM)。

8051 共有 4KB ROM，用于存放程序、原始数据或表格，因此称之为程序存储器，掉电后数据不会丢失，简称内部 ROM。

4) 定时器/计数器。

8051 共有 2 个 16 位的定时器/计数器，以实现定时或计数功能，并以其定时或计数结果对计算机进行控制。

5) 并行 I/O 口。

MCS-51 共有 4 个 8 位的 I/O 口 (P0、P1、P2、P3)，以实现数据的并行输入输出。

6) 串行口。

MCS-51 单片机有一个全双工的串行口，以实现单片机和其他设备之间的串行数据传送。该串行口功能较强，既可作为全双工异步通信收发器使用，也可作为同步移位器使用。

7) 中断控制系统。

MCS-51 单片机的中断功能较强，以满足控制应用的需要。8051 共有 5 个中断源，即外中断 2

个、定时/计数中断 2 个、串行中断 1 个。全部中断分为高级和低级共 2 个优先级别。

8) 时钟电路。

MCS-51 芯片的内部有时钟电路,但石英晶体和微调电容需要外接。时钟电路为单片机产生时钟脉冲序列。晶振频率通常选择 6MHz、11.0592MHz 或 12MHz。

从上述内容可以看出, MCS-51 虽然是一个单片机芯片,但作为计算机应该具有的基本部件它都包括,因此实际上它已是一个简单的微型计算机系统了。

(2) MCS-51 的信号引脚。

MCS-51 是标准的 40 引脚双列直插式集成电路芯片,引脚排列如图 1.1.5 所示。

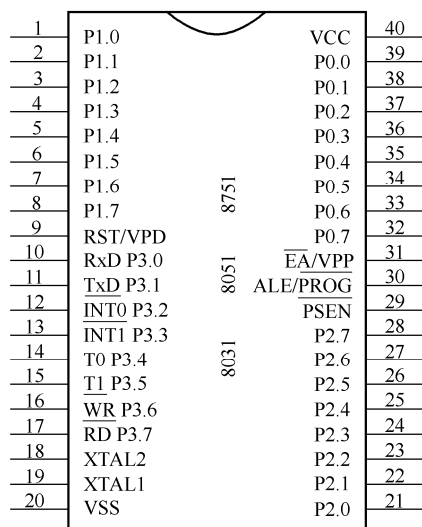


图 1.1.5 MCS-51 引脚图

1) 信号引脚介绍。

P0.0~P0.7: P0 口 8 位双向口线。

P1.0~P1.7: P1 口 8 位双向口线。

P2.0~P2.7: P2 口 8 位双向口线。

P3.0~P3.7: P3 口 8 位双向口线。

ALE: 地址锁存控制信号。

在系统扩展时, ALE 用于控制把 P0 口输出的低 8 位地址锁存器锁存起来, 以实现低位地址和数据的隔离。此外由于 ALE 是以晶振六分之一的固定频率输出的正脉冲, 因此可作为外部时钟或外部定时脉冲使用。

$\overline{\text{PSEN}}$: 外部程序存储器读选通信号。在读外部 ROM 时 $\overline{\text{PSEN}}$ 有效(低电平), 以实现外部 ROM 单元的读操作。

$\overline{\text{EA}}$: 访问程序存储控制信号。当 $\overline{\text{EA}}$ 信号为低电平时, 对 ROM 的读操作限定在外部程序存储器; 而当 $\overline{\text{EA}}$ 信号为高电平时, 则对 ROM 的读操作是从内部程序存储器开始, 并可延至外部程序存储器。

RST: 复位信号。当输入的复位信号延续 2 个机器周期以上高电平即为有效, 用以完成单片机的复位初始化操作。

XTAL1 和 XTAL2：外接晶体引线端。当使用芯片内部时钟时，这两个引线端用于外接石英晶体和微调电容；当使用外部时钟时，用于接外部时钟脉冲信号。

VSS：地线。

VCC：+5V 电源。

2) 信号引脚的第二功能。

由于工艺及标准化等原因，芯片的引脚数目是有限制的。例如 MCS-51 系列把芯片引脚数目限定为 40 条，但单片机为实现其功能所需要的信号数目却远远超过此数，因此就出现了需要与可能的矛盾。如何解决这个矛盾？“兼职”是唯一可行的办法，即给一些信号引脚赋以双重功能。如果把前述的信号定义为引脚第一功能的话，则根据需要再定义的信号就是它的第二功能。下面介绍一些信号引脚的第二功能。

P3 口线的第二功能：P3 的 8 条口线都定义有第二功能，如表 1.1.2 所示。

表 1.1.2 P3 口各引脚与第二功能表

引脚	第二功能	信号名称
P3.0	RxD	串行数据接收
P3.1	TxD	串行数据发送
P3.2	$\overline{\text{INT0}}$	外部中断 0 申请
P3.3	$\overline{\text{INT1}}$	外部中断 1 申请
P3.4	T0	定时器/计数器 0 的外部输入
P3.5	T1	定时器/计数器 1 的外部输入
P3.6	$\overline{\text{WR}}$	外部 RAM 写选通
P3.7	$\overline{\text{RD}}$	外部 RAM 读选通

EPROM 存储器程序固化所需要的信号：有内部 EPROM 的单片机芯片（如 8751），为写入程序需要提供专门的编程脉冲和编程电源，这些信号也是由信号引脚以第二功能的形式提供的，即：

编程脉冲：30 脚（ $\text{ALE}/\overline{\text{PROG}}$ ）。

编程电压（25V）：31 脚（ $\overline{\text{EA}}/\text{VPP}$ ）。

2. MCS-51 内部数据存储器

MCS-51 单片机的芯片内部有 RAM 和 ROM 两类存储器，即所谓的内部 RAM 和内部 ROM，首先分析内部 RAM。

（1）内部数据存储器低 128 单元。

8051 的内部 RAM 共有 256 个单元，通常把这 256 个单元按其功能划分为两部分：低 128 单元（单元地址 00H~7FH）和高 128 单元（单元地址 80H~FFH）。如表 1.1.3 所示为低 128 单元的配置介绍。

低 128 单元是单片机的真正 RAM 存储器，按其用途划分为 3 个区域：

1) 寄存器区。

共有四组寄存器，每组 8 个寄存单元（各为 8 位），各组都以 R0~R7 作寄存单元编号。寄存器常用于存放操作数及中间结果等，由于它们的功能及使用不作预先规定，因此称之为通用寄存器，有时也叫工作寄存器。四组通用寄存器占据内部 RAM 的 00H~1FH 单元地址。

表 1.1.3 片内 RAM 的配置

单元地址	功能
30H~7FH	数据缓冲区
20H~2FH	位寻址区 (00H~7FH)
18H~1FH	工作寄存器 3 区 (R7~R0)
10H~17H	工作寄存器 2 区 (R7~R0)
08H~0FH	工作寄存器 1 区 (R7~R0)
00H~07H	工作寄存器 0 区 (R7~R0)

在任一时刻, CPU 只能使用其中的一组寄存器, 并且把正在使用的那组寄存器称为当前寄存器组。到底是哪一组, 由程序状态字寄存器 PSW 中 RS1、RS0 位的状态组合来决定。

通用寄存器为 CPU 提供了就近数据存储的便利, 有利于提高单片机的运算速度。此外, 使用通用寄存器还能提高程序编制的灵活性, 因此在单片机的应用编程中应充分利用这些寄存器, 以简化程序设计, 提高程序运行速度。

2) 位寻址区。

内部 RAM 的 20H~2FH 单元, 既可以作为一般 RAM 单元使用, 进行字节操作, 也可以对单元中的每一位进行位操作, 因此把该区称为位寻址区。位寻址区共有 16 个 RAM 单元, 共 128 位, 位地址为 00H~7FH。MCS-51 具有布尔处理机功能, 这个位寻址区可以构成布尔处理机的存储空间。这种位寻址能力是 MCS-51 的一个重要特点。表 1.1.4 所示为位寻址区的位地址表。

表 1.1.4 片内 RAM 位寻址区的位地址

单元地址	位地址							
	MSB							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

3) 用户 RAM 区。

在内部 RAM 低 128 单元中, 通用寄存器占去 32 个单元, 位寻址区占去 16 个单元, 剩下 80 个单元是供用户使用的一般 RAM 区, 其单元地址为 30H~7FH。

对用户 RAM 区的使用没有任何规定或限制。但在一般应用中常把堆栈开辟在此区中。

(2) 内部数据存储器高 128 单元。

内部 RAM 的高 128 单元是供给专用寄存器使用的, 其单元地址为 80H~FFH。因为这些寄存器的功能已作专门规定, 所以称之为专用寄存器 (Special Function Register), 也可称为特殊功能寄存器。

8051 共有 21 个专用寄存器, 现对其中部分寄存器进行简单介绍。

1) 程序计数器 (Program Counter, PC)。

PC 是一个 16 位的计数器, 作用是控制程序的执行顺序, 内容为将要执行指令的地址, 寻址范围达 64KB。PC 有自动加 1 功能, 从而实现程序的顺序执行。PC 没有地址, 是不可寻址的。因此用户无法对它进行读写。但可以通过转移、调用、返回等指令改变其内容, 以实现程序的转移。因地址不在 SFR 之内, 一般不计作专用寄存器。

2) 累加器 (Accumulator, ACC)。

累加器为 8 位寄存器, 是最常用的专用寄存器, 功能较多, 地位重要。它既用来存放操作数, 也可用来存放运算的中间结果。MCS-51 单片机中大部分单操作数指令的操作数就取自累加器, 许多双操作数指令中的一个操作数也取自累加器。

3) B 寄存器。

B 寄存器是一个 8 位寄存器, 主要用于乘除运算。乘法运算时, B 是乘数。乘法操作后, 乘积的高 8 位存于 B 中。除法运算时, B 是除数。除法操作后, 余数存于 B 中。此外, B 寄存器也可作为一般数据寄存器使用。

4) 程序状态字 (Program Status Word, PSW)。

程序状态字是一个 8 位寄存器, 用于存储程序运行中的各种状态信息。其中有些位状态是根据程序执行结果由硬件自动设置的, 而有些位状态则使用软件方法设定。PSW 的位状态可以用专门指令进行测试, 也可以用指令读出。一些条件转移指令将根据 PSW 有些位的状态进行程序转移。PSW 的各位定义如下:

PSW 位地址	D7H	D6H	D5H	D4H	D3H	D2H	D1H	D0H
字节地址 D0H	CY	AC	F0	RS1	RS0	OV	F1	P

除 PSW.1 位保留未用外, 其余各位的定义及使用介绍如下:

- **CY (PSW.7): 进位标志位。**CY 是 PSW 中最常用的标志位, 功能有两个: 一是存放算术运算的进位标志, 在进行加或减运算时, 如果操作结果最高位有进位或借位时, CY 由硬件置 1, 否则清零; 二是在位操作中作累加位使用。位传送、位与位或等位操作, 操作位之一固定是进位标志位。
- **AC (PSW.6): 辅助进位标志位。**在进行加减运算中, 当有低 4 位向高 4 位进位或借位时, AC 由硬件置 1, 否则 AC 位被清零。

- F0 (PSW.5): 用户标志位。这是一个供用户定义的标志位, 需要利用软件方法置位或复位, 用以控制程序的转向。
- RS1 和 RS0 (PSW.4、PSW.3): 寄存器组选择位, 用于选择 CPU 当前工作的通用寄存器组。通用寄存器共有四组, 其对应关系如表 1.1.5 所示。

表 1.1.5 寄存器组对应关系

RS1	RS0	寄存器组	片内 RAM 地址
0	0	第 0 组	00H~07H
0	1	第 1 组	08H~0FH
1	0	第 2 组	10H~17H
1	1	第 3 组	18H~1FH

这两个选择位的状态是由软件设置的, 被选中的寄存器组即为当前通用寄存器组。但当单片机上电或复位后, RS1 RS0=00。

- OV (PSW.2): 溢出标志位。在带符号数加减运算中, OV=1 表示加减运算超出了累加器 A 所能表示的符号数有效范围 (-128~+127), 即产生了溢出, 因此运算结果是错误的; 否则, OV=0 表示运算正确, 即无溢出产生。在乘法运算中, OV=1 表示乘积超过 255, 即乘积分别在 B 与 A 中; 否则, OV=0 表示乘积只在 A 中。在除法运算中, OV=1 表示除数为 0 表示除法不能进行; 否则, OV=0, 除数不为 0, 除法可正常进行。
- P (PSW.0): 奇偶标志位。表明累加器 A 内容的奇偶性, 如果 A 中有奇数个 1, 则 P 置 1, 否则置 0。凡是改变累加器 A 中内容的指令均会影响 P 标志位。此标志位对串行通信中的数据传输有重要的意义。在串行通信中常采用奇偶校验的办法来校验数据传输的可靠性。

5) 数据指针 (DPTR)。

数据指针是 MCS-51 中的一个 16 位寄存器。编程时, DPTR 既可以按 16 位寄存器使用, 也可以按两个 8 位寄存器分开使用, 即:

- DPH: DPTR 高位字节。
- DPL: DPTR 低位字节。

DPTR 通常在访问外部数据存储器时作地址指针使用, 由于外部数据存储器的寻址范围为 64KB, 故把 DPTR 设计为 16 位。

6) 堆栈指针 (Stack Pointer, SP)。

堆栈是一个特殊的存储区, 用来暂存数据和地址, 它是按“先进后出”的原则存取数据的。堆栈有两种操作: 进栈和出栈。

MCS-51 单片机由于堆栈设在内部 RAM 中, 因此 SP 是一个 8 位寄存器。系统复位后, SP 的内容为 07H, 使得堆栈实际上从 08H 单元开始。但 08H~1FH 单元分别属于工作寄存器 1~3 区, 如程序中要用到这些区, 则最好把 SP 值改为 1FH 或更大的值。一般地, 堆栈最好在内部 RAM 的 30H~7FH 单元中开辟。SP 的内容一经确定, 堆栈的位置也就跟着确定下来, 由于 SP 可初始化为不同值, 因此堆栈位置是浮动的。

此处只集中讲述了 6 个专用寄存器, 其余的专用寄存器 (如 TCON、TMOD、IE、IP、SCON、

PCON、SBUF 等）将在后面陆续介绍。

（3）专用寄存器中的字节寻址和位地址。

MCS-51 系列单片机有 21 个可寻址的专用寄存器，其中有 11 个专用寄存器是可以位寻址的。各寄存器的字节地址及位地址如表 1.1.6 所示。

表 1.1.6 MCS-51 专用寄存器地址表

SFR	位地址/位定义								字节地址
B	F7	F6	F5	F4	F3	F2	F1	F0	F0H
ACC	E7	E6	E5	E4	E3	E2	E1	E0	E0H
PSW	D7	D6	D5	D4	D3	D2	D1	D0	D0H
	CY	AC	F0	RS1	RS0	OV	F1	P	
IP	BF	BE	BD	BC	BB	BA	B9	B8	B8H
	/	/	/	PS	PT1	PX1	PT0	PX0	
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
	EA	/	/	ES	ET1	EX1	ET0	EX0	
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	9F	9E	9D	9C	9B	9A	99	98	98H
	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	
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)
TL0									(8AH)
TMOD	GAT	C/T	M1	M0	GAT	C/T	M1	M0	(89H)
TCON	8F	8E	8D	8C	8B	8A	89	88	88H
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
PCON	SMOD	/	/	/	GF1	GF0	PDWN	IDLE	(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	

对专用寄存器的字节寻址问题作如下几点说明：

- 21 个可字节寻址的专用寄存器是不连续地分散在内部 RAM 的高 128 单元之中，尽管还余有许多空闲地址，但用户并不能使用。
- 程序计数器 PC 不占据 RAM 单元，它在物理上是独立的，因此是不可寻址的寄存器。
- 对专用寄存器只能使用直接寻址方式，书写时既可使用寄存器符号，也可使用寄存器单元地址。
- 表中凡字节地址不带括号的寄存器都是可进行位寻址的寄存器，而带括号的是不可位寻址的寄存器。全部专用寄存器可寻址的位共 83 位，这些位都具有专门的定义和用途。这样加上位寻址区的 128 位，在 MCS-51 的内部 RAM 中共有 $128+83=211$ 个可寻址位。

3. MCS-51 内部程序存储器

MCS-51 的程序存储器用于存放编好的程序和表格常数。8051 片内有 4KB 的 ROM。MCS-51 的片外最多能扩展 64KB 程序存储器，片内外的 ROM 是统一编址的。当 $\overline{EA}$ 端保持高电平，8051 的程序计数器 PC 在 0000H~0FFFH 地址范围内（即前 4KB 地址）是执行片内 ROM 中的程序，当 PC 在 1000H~FFFFH 地址范围时，自动执行片外程序存储器中的程序；当 $\overline{EA}$ 端保持低电平时，只能寻址外部程序存储器，片外存储器可以从 0000H 开始编址。

MCS-51 的程序存储器中有些单元具有特殊功能，使用时应予以注意。

其中一组特殊单元是 0000H~0002H。系统复位后，(PC)=0000H，单片机从 0000H 单元开始取指令执行程序。如果程序不从 0000H 单元开始，应在这三个单元中存放一条无条件转移指令，以便直接转去执行指定的程序。

还有一组特殊单元 0003H~002AH，共 40 个单元，这 40 个单元被均匀地分为 5 段，作为 5 个中断源的中断地址区。

- 0003H~000AH：外部中断 0 中断地址区。
- 000BH~0012H：定时器/计数器 0 中断地址区。
- 0013H~001AH：外部中断 1 中断地址区。
- 001BH~0022H：定时器/计数器 1 中断地址区。
- 0023H~002AH：串行中断地址区。

中断响应后，按中断种类自动转到各中断区的首地址去执行程序。因此在中断地址区中理应存放中断服务程序。但通常情况下，8 个单元难以存下一个完整的中断服务程序，因此通常也是从中断地址区首地址开始存放一条无条件转移指令，以便中断响应后通过中断地址区再转到中断服务程序的实际入口地址区。

三、并行输入/输出口电路结构

单片机芯片内还有一项主要内容就是并行 I/O 口。MCS-51 共有 4 个 8 位的并行 I/O 口，分别记作 P0、P1、P2、P3。每个口都包含一个锁存器、一个输出驱动器和输入缓冲器。实际上它们已被归入专用寄存器之列，并且具有字节寻址和位寻址功能。

在访问片外扩展存储器时，低 8 位地址和数据由 P0 口分时传送，高 8 位地址由 P2 口传送。在无片外扩展存储器的系统中，这 4 个口的每一位均可作为双向的 I/O 端口使用。

MCS-51 单片机的 4 个 I/O 口都是 8 位双向口，这些口在结构和特性上是基本相同的，但又各具特点，下面具体介绍。

1. P0 口

P0 口的口线逻辑电路如图 1.1.6 所示。

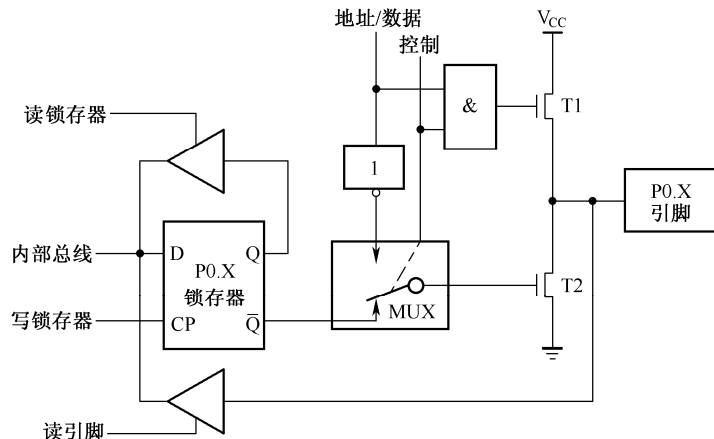


图 1.1.6 P0 口某位结构

由图可见，电路中包含有 1 个数据输出锁存器、2 个三态数据输入缓冲器、1 个数据输出驱动电路和 1 个输出控制电路。当对 P0 口进行写操作时，由锁存器和驱动电路构成数据输出通路。由于通路中已有输出锁存器，因此数据输出时可以与外设直接连接，而不需要再加数据锁存电路。

考虑到 P0 口既可以作为通用的 I/O 口进行数据的输入输出，也可以作为单片机系统的地址/数据线使用。为此在 P0 口的电路中有一个多路转接电路 MUX。在控制信号的作用下，多路转接电路可以分别接通锁存器输出或地址/数据线。当作为通用的 I/O 口使用时，内部的控制信号为低电平，封锁与门将输出驱动电路的上拉场效应管（FET）截止，同时使多路转接电路 MUX 接通锁存器 Q 端的输出通路。

当 P0 口作为输出口使用时，内部的写脉冲加在 D 触发器的 CP 端，数据写入锁存器，并向端口引脚输出。

当 P0 口作为输入口使用时，应区分读引脚和读端口两种情况。为此在口电路中有两个用于读入驱动的三态缓冲器。所谓读引脚就是读芯片引脚的数据，这时使用下方的数据缓冲器，由“读引脚”信号把缓冲器打开，把端口引脚上的数据从缓冲器通过内部总线读进来。使用传送指令（MOV）进行读口操作都是属于这种情况。

而读端口是指通过上面的缓冲器读锁存器 Q 端的状态。在端口已处于输出状态的情况下，本来 Q 端与引脚的信号是一致的，这样安排的目的是为了适应对口进行“读—修改—写”操作指令的需要。例如“ANL P0,A”就属于这类指令，执行时先读入 P0 口锁存器中的数据，然后与 A 的内容进行逻辑与，再把结果送回 P0 口。对于这类“读—修改—写”指令，不直接读引脚而读锁存器是为了避免可能出现的错误。因为在端口已处于输出状态的情况下，如果端口的负载恰是一个晶体管的基极，导通了的 PN 结会把端口引脚的高电平拉低，这样直接引脚就会把本来的 1 误读为 0。但若从锁存器 Q 端读，就能避免这样的错误，得到正确的数据。

但是要注意，当 P0 口进行一般的 I/O 输出时，由于输出电路是漏极开路电路，必须外接上拉电阻才能有高电平输出；当 P0 口进行一般的 I/O 输入时，必须先向电路中的锁存器写入 1，使 FET

截止，以避免锁存器为 0 状态时对引脚读入的干扰。

在实际应用中，P0 口绝大多数情况下都是作为单片机系统的地址/数据线使用，这要比作一般 I/O 口应用简单。当输出地址或数据时，由内部发出控制信号，打开上面的与门，并使多路转接电路 MUX 处于内部地址/数据线与驱动场效应管栅极反相接通状态，这时的输出驱动电路由于上下两个 FET 处于反相，形成推拉式电路结构，使负载能力大为提高，而当输入数据时，数据信号则直接从引脚通过输入缓冲器进入内部总线。

2. P1 口

P1 口的口线逻辑电路如图 1.1.7 所示。

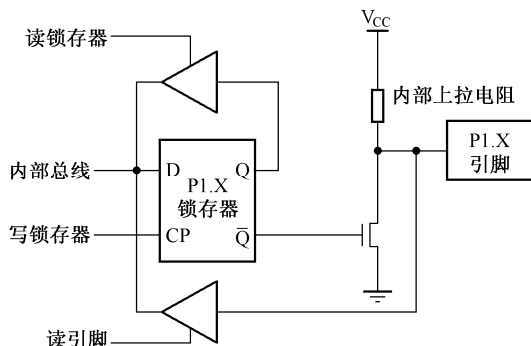


图 1.1.7 P1 口某位结构

因为 P1 口通常是作为通用 I/O 口使用的，所以在电路结构上与 P0 口有一些不同之处。首先它不再需要多路转接电路 MUX；其次电路的内部有上拉电阻，与场效应管共同组成输出驱动电路。

为此 P1 口作为输出口使用时，已能向外提供推拉电流负载，无需再外接上拉电阻。当 P1 口作为输入口使用时，同样也需要先向其锁存器写 1，使输出驱动电路的 FET 截止。

3. P2 口

P2 口的口线逻辑电路如图 1.1.8 所示。

P2 口电路中比 P1 口多了一个多路转接电路 MUX，这又正好与 P0 口一样。P2 口可以作为通用 I/O 口使用。这时多路转接开关倒向锁存器 Q 端。但通常应用情况下，P2 口是作为高位地址线使用，此时多路转接开关应倒向相反方向。

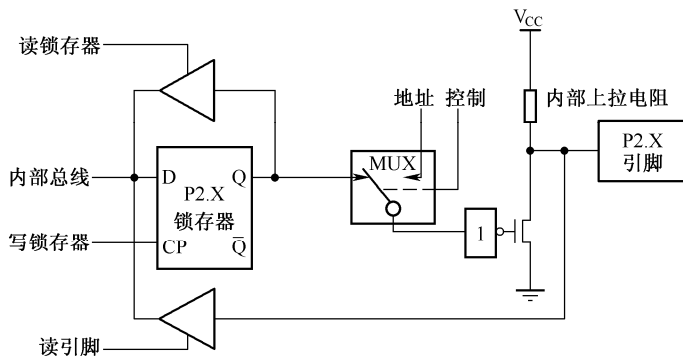


图 1.1.8 P2 口某位结构图

4. P3 口

P3 口的口线逻辑电路如图 1.1.9 所示。

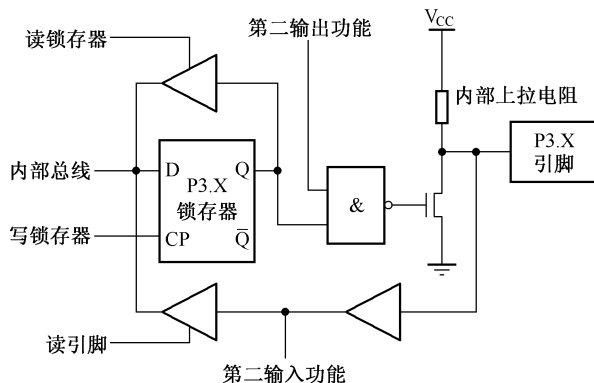


图 1.1.9 P3 口某位结构

P3 口的特点在于为适应引脚信号第二功能的需要，增加了第二功能控制逻辑。由于第二功能信号有输入和输出两类，因此分两种情况说明。

对于第二功能为输出的信号引脚，当作为 I/O 使用时，第二功能信号引线应保持高电平，与非门开通，以维持从锁存器到输出端数据输出通路的畅通。当输出第二功能信号时，该位的锁存器应置 1，使与非门对第二功能信号的输出是畅通的，从而实现第二功能信号的输出。

对于第二功能为输入的信号引脚，在口线的输入通路上增加了一个缓冲器，输入的第二功能信号就从这个缓冲器的输出端取得。而作为 I/O 使用的数据输入，仍取自三态缓冲器的输出端。不管是作为输入口使用还是第二功能信号输入，输出电路中的锁存器输出和第二功能输出信号线都应保持高电平。

四、复位电路

单片机复位是使 CPU 和系统中的其他功能部件都处在一个确定的初始状态，并从这个状态开始工作，例如复位后 $PC=0000H$ ，使单片机从第一个单元取指令。无论是在单片机刚开始接上电源时，还是断电后或者发生故障后都要复位。所以我们必须弄清楚 MCS-51 型单片机复位的条件、复位电路和复位后的状态。

单片机复位的条件是：必须使 RST 引脚（9）加上持续 2 个机器周期（即 24 个振荡周期）的高电平。例如，若时钟频率为 12MHz，每机器周期为 $1\mu s$ ，则只需 $2\mu s$ 以上时间的高电平。在 RST 引脚出现高电平后的第二个机器周期执行复位。单片机常见的复位电路如图 1.1.10 所示。

图 1.1.10 所示为按键复位电路。该电路除具有上电复位功能外，若要复位，只需按图中的 RESET 键，此时电源 VCC 经电阻 R1、R2 分压在 RST 端产生一个复位高电平。

五、单片机的工作过程

单片机自动完成赋予它的任务的过程，也就是单片机执行程序的过程，即一条条执行指令的过程。所谓指令就是把要求单片机执行的各种操作命令的形

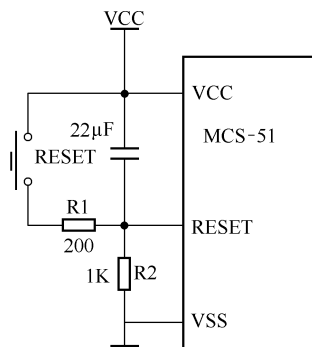


图 1.1.10 单片机常见的复位电路

式写下来,这是由设计人员赋予它的指令系统所决定的,一条指令对应着一种基本操作;单片机所能执行的全部指令就是该单片机的指令系统,不同种类的单片机,其指令系统也不同。为使单片机能自动完成某一特定任务,必须把要解决的问题编成一系列指令(这些指令必须是选定单片机能识别和执行的指令),这一系列指令的集合就称为程序,程序需要预先存放在具有存储功能的部件——存储器中。存储器由许多存储单元(最小的存储单位)组成,就像楼房由许多房间组成一样,指令就存放在这些单元里,单元里的指令取出并执行就像楼房的每个房间被分配到了唯一一个房间号一样,每一个存储单元也必须被分配到唯一的地址号,该地址号称为存储单元的地址,这样只要知道了存储单元的地址,就可以找到这个存储单元,其中存储的指令就可以被取出,然后再被执行。

程序通常是顺序执行的,所以程序中的指令也是一条条顺序存放的,单片机在执行程序时要能把这些指令一条条取出并加以执行,必须有一个部件能追踪指令所在的地址,这一部件就是程序计数器 PC(包含在 CPU 中),在开始执行程序时,给 PC 赋以程序中第一条指令所在的地址,然后取得每一条要执行的命令,PC 之中的内容就会自动增加,增加量由本条指令的长度决定,可能是 1、2 或 3,以指向下一条指令的起始地址,保证指令顺序执行。

单片机上电或复位时,PC 自动清零,即装入地址 0000H,这就保证了单片机上电或复位后程序从 0000H 地址开始执行。

项目分析

一、硬件电路分析

发光二极管控制电路如图 1.1.11 所示,R0 为限流电阻,D0 为发光二极管。根据硬件电路图可知,当 P1.0 引脚输出高电平时,发光二极管灭;当 P1.0 引脚输出低电平时,发光二极管亮。

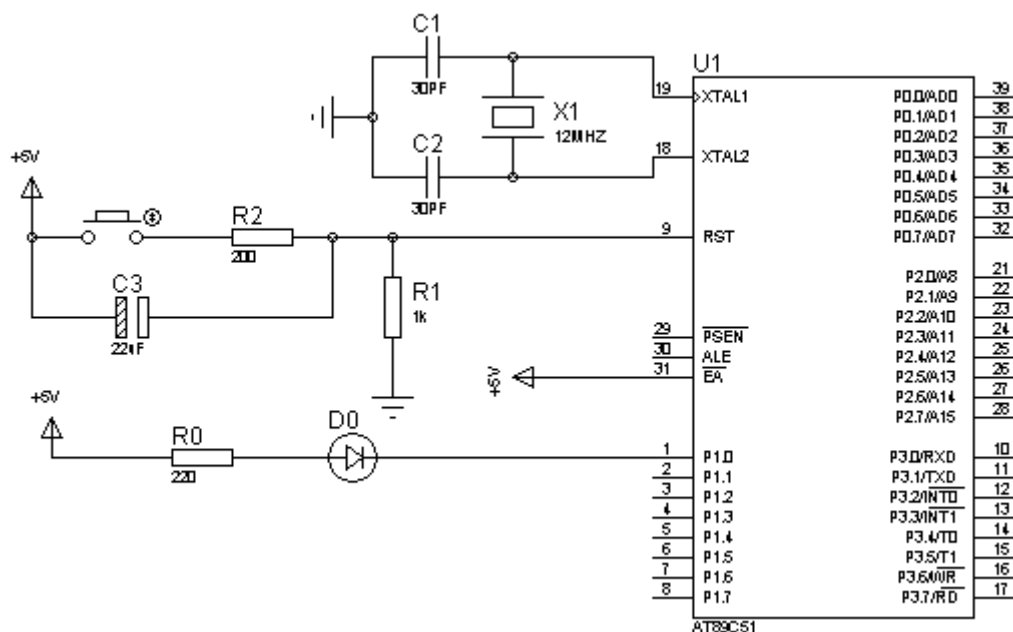


图 1.1.11 发光二极管控制电路

二、软件设计

(1) 程序流程图，如图 1.1.12 所示。

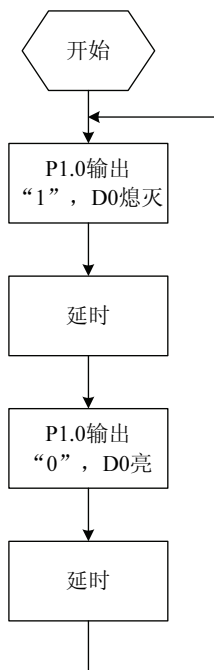


图 1.1.12 程序流程图

(2) 软件设计。

1) 汇编程序。

```

ORG 0000H
START:
SETB P1.0           ;P1.0 口置高电平
LCALL DELAY         ;调用延时子程序
CLR P1.0            ;P1.0 口置低电平
LCALL DELAY
LJMP START          ;程序跳转到标号为 START 的地方继续执行
DELAY: MOV R5,#20    ;延时子程序 D1: MOV R6,#20
D1: MOV R6,#20
D2: MOV R7,#250
DJNZ R7,$
DJNZ R6,D2
DJNZ R5,D1
RET                 ;子程序返回
END
  
```

2) C 程序。

```

#include <reg51.h>
sbit P1_0=P1^0;      //定义 P1_0 为 P1 口的第 1 位，以便进行位操作
void delay ();
void main()
  
```

```

{
    while(1)
    {
        P1_0=1;           // P1.0 口置高电平
        delay ();
        P1_0=0;           // P1.0 口置低电平
        delay ();
    }
}
void delay ()             //延时子程序
{
    unsigned char i,j,k;
    for(i=0;i<10;i++)
    {
        for(j=0;j<10;j++)
            for(k=0;k<10;k++);
    }
}

```

C 语言中，可以通过关键字 `sbit` 来定义特殊功能寄存器中的可寻址位，定义 P1 口的第 0 位可采用语句“`sbit P1_0=P1^0;`”。

项目实施

- (1) 按照图 1.1.11 在 Proteus 中连接好电路。
- (2) 在 keil c 中编写程序，生成 hex 文件。
- (3) 将生成的 hex 文件加载到单片机芯片中。
- (4) 在 proteus 中仿真，观察结果。

元件清单如表 1.1.7 所示。

表 1.1.7 元件清单

元件名称	Proteus 中的名称
单片机芯片	AT89C51
晶振	CRYSTAL
电容	CAP
电解电容	CAP-ELEC
发光二极管	LED-RED
电阻	RES
按键	BUTTON

练习题

1. 什么是中断？设置中断有何意义？
2. 8051 是几位单片机？几 KB 的 ROM？多少个 RAM 单元？
3. 8051 单片机有多少个引脚？电源和地分别对应哪个引脚？
4. 简述：

- (1) 8051 $\overline{\text{EA}}$ 引脚的功能。
- (2) 8051 RST 引脚的功能。
- (3) XTAL1 和 XTAL2 引脚的功能。
5. 简述 P3.0~P3.7 引脚的第二功能。
6. 8051 单片机 RAM 单元的地址范围、位寻址单元的地址范围、用户 RAM 区的地址范围各是多少？
7. 简述 PC 计数器的功能。
8. PSW 是什么？请简述 PSW 中各位的含义。
9. 堆栈按什么原则存取数据？SP 是什么？系统复位后 SP 的内容是多少？
10. 51 单片机 ROM 最大能扩展到多少 KB？单片机复位后 PC 中的值是多少？
11. RST 引脚复位的条件是什么？