

<<微型计算机原理及应用>> 第六版预习大纲

目录

<<微型计算机原理及应用>>第六版预习大纲.....	1
第一章	1
第二章	1
第三章	3
第四章	6
第五章	6
第六章	7
第七章	8
第八章	8

说明:

(1) 为了减轻学习负担, 本学期不布置纸面作业, 本文中列出与课程内容有关的课后习题, 各位同学根据学习情况自行安排练习, 用于强化基本概念和复习教学内容;

(2) 预习大纲的作用同样用于指导自学, 强烈建议课前预习;

(3) 课程学习中的作业和课堂练习包括: 7 次编程练习课 (共 14 道简单练习题和 1 个综合练习), 还有 5 道电路设计练习。

第一章

学习 1.1~1.3, 1.4 自学, 有助于了解后续微处理器发展的特点

本章帮助建立计算机和微型计算机的整体结构, 有很多基本概念, 重点掌握以下内容

- ✧ 数的表示: 十进制、二进制、八进制、十六进制
- ✧ 十进制、二进制、十六进制之间的转换
- ✧ 字节、字、双字的概念
- ✧ 二进制编码: BCD 码、ASCII 码的概念
- ✧ 有符号数表示法: 原码、反码、补码的概念
- ✧ 为什么计算机中采用补码表示有符号数

勘误: 有些十六进制数没有带后缀 H

注意: 非十进制数据后面要加上相应的字母表示进制, 十进制数据的 “D” 可以省略

- ✧ 冯诺依曼计算机基本组成
- ✧ 什么是汇编语言和机器码?
- ✧ 微型计算机的基本结构
- ✧ 微处理器的功能?
- ✧ 存储器的功能?
- ✧ 1KB、1MB 是什么意思?
- ✧ 什么是总线? 有哪些总线?
- ✧ 什么是 IO 设备?
- ✧ IO 接口电路的用途?

补充基本概念:

逻辑运算和逻辑门

取反、与、或、与非、或非、异或

驱动器 (缓冲器)、三态驱动器 (三态缓冲器)

输出冲突和地址选通的概念

译码器

编码器

锁存器

课后习题

1~12

13 中有关 8086 的部分

第二章

学习 2.1~2.2, 其中除以下内容外都需要掌握

1. 最大模式相关内容
2. 时钟发生器 8284A
3. 8088 处理器相关内容
4. P30 最小模式最大模式复用信号无需了解
5. 80286 等更高级 CPU 的内容可以了解, 不属于考试范围

第二章的 2.1~2.2 节涉及很多基本概念, 帮助我们了解微处理器内部结构和系统组成, 需要重点关注。

第二章引言及 2.1.1 节 微处理器内部结构

- ✧ 8086CPU 内部分为几部分?

- ✧ EU 的功能?
- ✧ BIU 的功能?
- ✧ ALU 的功能?
- ✧ 为什么要分为 EU 和 BIU 两部分?
- ✧ 8086 指令队列有多少字节? 为什么要设置指令队列?

2.1.2 节 寄存器

- ✧ 寄存器的用途? 8086 CPU 内部数据寄存器有哪些?
- ✧ 地址指针和变址寄存器有哪些? 作为地址指针的时候, 寄存器存放的值表示什么?
- ✧ BX、BP、SI、DI、SP 各是什么寄存器?
- ✧ CS、SS、DS、ES 各是什么寄存器?
- ✧ 寄存器 IP 的用途? 程序员可以直接给 IP 寄存器赋值吗?
- ✧ 标志寄存器有几个状态标志位? 各自的定义是什么?
- ✧ 标志寄存器有几个控制位? 各起什么作用?
- ✧ 微处理器中为什么要设置标志位?
- ✧ 根据哪些标志位判断两个无符号数的大小? :
答: ZF 和 CF
- ✧ 根据哪些标志位判断两个有符号数的大小?
答: ZF, OF, SF

2.1.3 节, 不含最大模式相关内容

- ✧ 需要关注以下引脚的用途
- ✧ AD0~AD15 的双向传输和复用性质
- ✧ A19:16 是地址总线高位, 不需要关注 Status 用途
- ✧ /BHE、/RD、/WR
INTR、/INTA、NMI
RESET、CLK、M/IO
ALE、/DEN、DT/R
MN/MX、VCC、GND
- ✧ 什么是最小模式? 什么是最大模式?
- ✧ 未列出的其它引脚不需要关注

2.1.4 节 存储器组织 (不需要关注 8088 内容)

- ✧ 8086 存储器空间的地址范围是?
- ✧ 什么是逻辑地址和物理地址?
- ✧ 逻辑地址和物理地址如何转换?
- ✧ 为什么 8086 指令中要采用逻辑地址表示方式? 根据逻辑地址计算物理地址的功能由什么部件完成?
- ✧ 8086 的逻辑段有几种类型? 用途是什么?

- ✧ 读取指令使用的段寄存器和偏移地址寄存器是什么?
- ✧ 访问数据默认使用的段寄存器和偏移地址寄存器是什么?
- ✧ 访问堆栈默认使用的段寄存器和偏移地址寄存器是什么?
- ✧ 想一想寄存器和存储器的区别, 8086 如何对二者进行访问
- ✧ 什么是堆栈? 主要用途?
- ✧ 进出栈的数据位宽是 16 位, 其中低字节存放在低地址高低字节如何放置?
- ✧ 进栈操作 SP 如何变化, 压栈的数据如何保存进堆栈中?
- ✧ 出栈操作 SP 如何变化, 栈顶的数据如何写入目标操作数中?
- ✧ 为什么 8086 存储器采用分体结构?
- ✧ 每个存储体的最大容量和覆盖的地址范围:
答: 每个存储体 512KB, 分别覆盖 0~FFFFH 的奇地址和偶地址
- ✧ 8086 如何通过分体结构实现字、字节访问?
- ✧ 一个 16 位的字在 8086 存储器中占据几个存储单元?
- ✧ /BHE 信号在执行什么操作时会有效? 答: 当进行字访问或者对奇地址进行字节访问的时候/BHE 信号有效, 用于选中奇地址存储体进行读或写操作;
- ✧ A0 信号在执行什么操作时会有效? 答: 当进行字访问或者对偶地址进行字节访问的时候 A0 信号有效, 用于选中偶地址存储体进行读或写操作;
- ✧ 为什么字最好存放在偶地址开始的连续两个存储单元中?

2.2 节 工作模式和总线操作 (不学最大模式)

2.2.1 不需要学习时钟发生器 8284A

- ✧ 74LS373、74LS245、74LS244 是本课程的常用器件, 需要了解其功能
- ✧ 最小模式系统组成需掌握
- ✧ 8086 系统总线是地址总线 A19:0, 数据总线 D15:0, 以及其它控制信号: 读写、M/IO、/BHE 等;
- ✧ 分析图 2.10, 并关注
- ✧ (1) 如何形成 8086 系统的 20 位地址总线 A19:0?
- ✧ (2) 如何形成 8086 系统的 16 位数据总线

D15:0?

- ✧ (3) 存储器和 IO 通过什么与 8086 进行数据交换?
- ✧ 所有能够向数据总线输出的设备,其数据输出端口都需要具有什么控制功能?为什么?
- ✧ 8086 内部的什么模块生成访问存储器或 IO 接口所需要的总线信号?
- ✧ 了解最小模式下 8086 实现读或者写操作的过程

勘误: P39 第二行 D 触发器,应为 D 锁存器

2.2.3 节 (不含 3.最大模式下的读/写总线周期)

- ✧ CPU 的工作过程受 CLK 信号同步;
- ✧ 什么是指令周期、总线周期、时钟周期?
- ✧ 掌握 8086 总线操作时序
- ✧ 总线周期至少包括几个时钟周期? T1 主要用于输出什么信号?
- ✧ 读周期的标志是什么?说明 T2、T3 周期数据总线变化的情况,执行 MOV AX, [2000H]指令时,在 T3 周期数据总线上传输的数据来自哪里?
- ✧ 写周期的标志是什么?说明 T2、T3 周期数据总线变化的情况,执行 MOV [2000H], AX 时,在 T2、T3 周期数据总线上传输的数据来自哪里?
- ✧ 执行指令 MOV AX, 2000H 和 MOV AX, BX 时是否产生总线周期?为什么?

本章的最后一节将在课堂用 Modelsim 软件采用 VerilogHDL 语言进行电路连线设计和波形观察,请带电脑,充好电,并预装 Modelsim 和自己常用的文本编辑软件

课后习题

所有题目只考虑与 8086 有关的部分

1~14 题

17 题,应改为“每秒最多执行多少条指令”,答:

8M/4=2M

18 题

第三章

以下指令不需掌握:

XLAT、LAHF、SAHF;

全部十进制调整指令: AAA、DAA、AAS、DAS、AAM、AAD;

全部串操作指令:

循环指令: LOOP、LOOPE/LOOPZ、LOOPNE/LOOPNZ、JCXZ;

ESC、WAIT、LOCK;

CMC、CLD、STD

请首先按照下面的章节提示将第三章和第四章通读一遍,对指令和汇编程序建立概括性了解,再逐小节细读预习。

请预先安装并学习虚拟机用法。

课堂编程练习安排连续 7 次课,请带笔记本电脑,充好电。

3.1 寻址方式 非常重要

- ✧ 指令包括哪两部分?操作码、操作数的定义?操作数的来源有哪些?
- ✧ 什么是寻址方式?
- ✧ 什么是立即寻址?可以用于对哪些目的操作数赋值?立即数可以作为目的操作数吗?
- ✧ 什么是寄存器寻址?举个寄存器寻址的例子
- ✧ 什么是寄存器寻址?
- ✧ 存储器寻址方式有哪些?
- ✧ 什么是有效地址?
- ✧ 什么是直接寻址?缺省情况下所寻址的存储单元在哪个段?直接寻址方式下操作数所在的物理地址如何计算?由 8086 的哪个部件进行物理地址的计算?
- ✧ 什么是符号地址?如何区分指令中的一个符号是符号地址还是立即数?
- ✧ 说明以下伪指令和指令的含义和用到的寻址方式

AA DB 12H; 定义字节变量 AA

BB DW 0FFH; 定义字变量 BB

CC EQU 12H 定义立即数 CC

MOV AX, CC 立即数给 AX

MOV [AA], AL; AL 的值写入变量 AA

MOV AX, [BB] 变量 BB 的值写入 AX

- ✧ 什么是寄存器间接寻址?可以使用哪些寄存器进行间接寻址?
- ✧ 没有段超越前缀的情况下,寄存器间接寻址指明的操作数在什么段?操作数的物理地址如何计算(分字节和字的情况说明)?
- ✧ 什么是寄存器相对寻址?有效地址如何计

算？

- ✧ 什么是基址变址寻址？有效地址如何计算？有哪些组合？
- ✧ 什么是相对基址变址寻址？有效地址如何计算？有哪些组合？
- ✧ 什么是隐含寻址？说明乘法指令中的隐含寻址。
- ✧ 什么是 IO 端口寻址？有几种类型？分别用于什么情况？

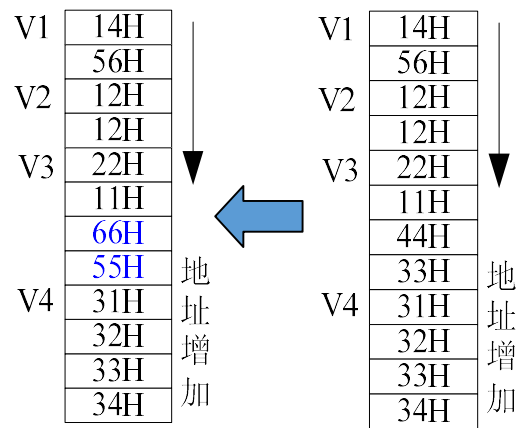
3.2 机器码，了解机器码的概念

3.3.1 传送类指令（不含 XLAT、LAHF 和 SAHF）

- ✧ 通用数据传送指令有哪几条？传送的对象是什么？
- ✧ 传送之后源操作数和目的操作数如何变化？
- ✧ 哪些不能作为 MOV 的源操作数或目的操作数？
- ✧ 解释指令的执行结果 MOV AX, 'B' 答：
AX=0042H
- ✧ 阅读指令，画出数据段中各单元内容，说明指令执行的结果

```
DD SEGMENT
V1 DB 14H, 56H
V2 DB 2 DUP(12H)
V3 DW 1122H, 3344H
V4 DB '1234'
DD ENDS
MOV AX, DD
MOV DS, AX
MOV AL, V1
MOV DI, OFFSET V3
MOV WORD PTR [DI+2], 5566H
MOV BX, 2
MOV AX, [BX+DI]
```

答：



将 DD 段首地址赋给 DS

AL=14H

DI=4

AX=5566H

- ✧ （接上题）说明以下每条指令执行的结果

MOV BX, OFFSET V1 ;**BX=0**

MOV AL, [BX] ;**AL=14H**

MOV AL, V1 ;**AL=14H**

MOV BP, OFFSET V1 ;**BP=0**

MOV AL, [BP] ;堆栈段偏移地址为 0 的单元内容赋值给 AL

- ✧ 说明 PUSH、POP 指令的执行过程
- ✧ XCHG 可以交换哪些操作数？不能交换哪些操作数？
- ✧ 判断指令的对错
 - (1) MOV DS, DATA
错，立即数不能给段寄存器赋值
 - (2) XCHG AL, 05H
错，XCHG 两个操作数均为源操作数和目的操作数，立即数不能作为目的操作数
 - (3) MOV V1, 05H
错，推测 V1 为变量，指令中变量和立即数均没有位宽信息，需要通过 WORD PTR 或者 BYTE PTR 对变量 V1 进行说明；如果 V1 为立即数
 - (4) MOV V1, V1+1
错，推测 V 为变量，传送指令不能在两个存储器类型操作数之间进行

- ✧ 不用 XLAT 指令，用 MOV 指令实现例 3.29 的查表功能
- ✧ 地址传送类指令有几条？
- ✧ 不需要了解
- ✧ 说明 PUSHF 和 POPF 执行的操作和作用
- ✧ 除了 POPF，其它传送类指令是否会影响标志位？

3.3.2 算术运算类(AAA、DAA、AAS、DAS、AAM、AAD 等十进制调整指令不需要了解)

- ✧ 算术指令可以处理哪五种类型的数据? 1000 0111 分别理解为五种类型时数值各是多少?
(注意: 我们不学习十进制调整指令, 因此, 算术指令中不会出现无符号压缩十进制数(压缩 BCD 码)和无符号非压缩十进制数(非压缩 BCD 码), 只在 8253 表示计数值中会用到压缩十进制数表示方式)
- ✧ 加法指令有哪些? 分别影响哪些标志位(注意 INC 指令不会影响 CF 标志位)? INC 用于对存储器单元操作时需要注意什么? **答: 需要指明存储单元的位宽**
- ✧ 如果将两个二进制数理解为无符号数, 如何判断其加法运算是否溢出? **答: CF=1, 则溢出**
- ✧ 如果将两个二进制数理解为有符号数, 如何判断其加法运算是否溢出? **答: OF=1 则溢出**
- ✧ 两个 8bit 无符号数相加, 结果取多少位不会溢出? 15 个 8bit 无符号数相加, 结果取多少位不会溢出? **答: 9bit, 12bit**
- ✧ 减法指令有哪几条? 分别影响哪些标志位?
(注意 DEC 指令不影响 CF)
- ✧ DEC 用于存储单元操作时需要注意什么? **答: 需要指明存储单元的位宽**
- ✧ 如果将两个二进制数理解为无符号数, 如何判断其减法运算是否溢出? 如果理解为有符号数呢? **答: 无符号溢出 CF=1, 有符号溢出 OF=1**
- ✧ 想一想, 为什么 INC 和 DEC 设计为不影响 CF?
- ✧ 8086 的乘、除法指令有哪几条? 源操作数可以是立即数吗? 有什么隐含寻址方式要求
- ✧ 乘法运算会发生溢出吗? 即乘法运算的乘积能否全部保存在结果寄存器中? **答: 不会溢出**
- ✧ 除法运算会溢出吗? 即除法运算的商能否全部保存在结果寄存器中? **答: 会溢出**
- ✧ 如果除法运算溢出会引发 8086 的什么操作? **答: 除法错中断**
- ✧ 举例说明 CBW、CWD 指令的格式和用途。**答: 用于实现有符号数位宽扩展**

3.3.3 节 逻辑运算指令

- ✧ 注意逻辑右移和算术右移的差别

- ✧ 逻辑运算指令对标志位的影响如何? **答: 逻辑运算常用于对某些位进行操作, 主要关注 ZF 标志位; 移位指令主要关注 CF 标志位, 移出位进入 CF;**
- ✧ 移位次数大于 1 时, 应如何操作?
- ✧ 说明循环移位指令 ROR、ROL、RCR、RCL 实现的操作分别是什么(注意循环时 CF 加入或者不加入移位过程, 但是移出位始终会进入 CF)

3.3.4 串操作, 不要求

3.3.5 控制转移类(条件循环控制类不要求)

- ✧ 无条件转移和过程调用指令有哪些?
- ✧ 无条件转移根据目标地址是否在另一段分为_____转移和_____转移两类, 分别通过修改哪些寄存器实现跳转? **答: 如果跳转的目标地址在当前代码段内, 称为近转移, 通过将目标地址偏移量写入 IP 实现跳转; 如果跳转的目标地址在另一个代码段, 称为远转移, 通过将目标地址段地址和偏移量分别写入 CS 和 IP 实现跳转**
- ✧ 段内转移的目标地址有哪些寻址方式?
- ✧ 段间转移的目标地址有哪些寻址方式?
- ✧ 过程的定义是什么? 过程(子程序)的重要性? **答: 过程通常用于实现相对独立的、可重复使用的功能, 合理使用过程(子程序)可以提高代码的可读性, 效率, 降低调试难度;**
- ✧ 执行近过程调用指令, CPU 自动完成哪些操作? **答: 执行近调用, CPU 自动执行以下操作: (1) 将 CALL 指令下一条指令所在偏移地址压栈, 准备用作过程调用的返回地址, (2) 过程入口地址赋值给 IP;**
- ✧ 执行远过程调用指令, CPU 自动完成哪些操作? **答: 执行远调用, CPU 自动执行: (1) 将 CALL 指令下一条指令所在段地址和偏移地址分别压栈, (2) 过程入口所在的段地址和偏移地址赋值给 CS 和 IP;**
- ✧ 执行近过程返回指令, CPU 自动完成哪些操作? **答: 执行近过程的返回指令 RET, CPU 自动执行: 从栈顶弹出 2 个字节数据进入 IP, 在堆栈操作正确的情况下, 该数据就是 CPU 之前压栈的 CALL 指令下一条指令所在偏移地址;**
- ✧ 执行远过程返回指令, CPU 自动完成哪些操作? **答: 执行远过程的返回指令 RET, CPU**

自动执行：从栈顶弹出 4 个字节数据进入 IP 和 CS，在堆栈操作正确的情况下，该数据就是 CPU 之前压栈的 CALL 指令下一条指令所在偏移地址和段地址；

- ✧ 注意：执行过程中 call 和 ret 必须成对执行，子程序内跳转指令的目标只能在本子程序内部；
- ✧ 条件转移指令判断的条件来自哪里？答：之前指令运算的结果对标志寄存器产生影响
- ✧ 如果符合条件转移指令如何执行？不符合条件指令如何执行？
- ✧ 条件转移指令允许的相对转移地址范围是多少？更远的条件转移如何实现？
- ✧ 直接标志转移指令有哪些？
- ✧ 无符号数比较间接标志转移指令有哪些？
- ✧ 有符号数比较间接标志转移指令有哪些？
- ✧ 什么是中断？什么是中断服务程序？8086 中断源有哪两类？
- ✧ 中断指令有 _____、_____、_____ 等三条
- ✧ 必须使用 _____ 指令退出中断服务程序？
- ✧ 中断类型 0~4 分别是什么？什么情况下产生该类型中断？INT 21 是什么？

3.3.6 处理器控制指令（需了解 CLC, STC, CLI, STI, HLT）

- ✧ 了解中断的概念和种类
- ✧ 清除进位位的指令是 _____，通过 _____ 指令禁止响应外中断，通过 _____ 指令允许响应外中断。

课后习题

1~3;

5~7;

8: 改为不使用 XLAT 指令（使用亦可）

9, 10

13

15, 16

17 题只考虑 CF 和 IF 的置位和清除

第四章

第四章前言和 4.1 指令与伪指令-都需掌握

4.2 了解即可，可用于编写有趣的程序

4.3 学习各种程序结构，学习程序设计方法，学习流程图的绘制方法

课后练习

1~4

6~7

12, 14, 18, 19

第五章

5.1 了解存储器的种类和基本概念

5.2 课程关注 5.2.1 SRAM，可以自学 DRAM

5.3 了解各种 ROM 的种类

5.4 课程的重点学习部分

5.5 不要求，可自学

5.1~5.3 节

- ✧ 地址线根数与可寻址空间的关系？
- ✧ 什么是 SRAM？特点是什么？什么是 DRAM？特点是什么？什么是 ROM？生活中遇到的 ROM 有哪些？
- ✧ 注意计算机存储器的层次化构成，其中最快的是内部寄存器，最慢的是外部大容量存储器（例如光盘、硬盘等）
- ✧ 计算机的主存是什么存储器构成的？计算机的高速缓存是什么存储器构成的？高速缓存用于解决什么问题？答：现代计算机的主存是 DRAM 构成的（最新到 DDR5），高速缓存又 SRAM 构成。前者容量大但是访问延迟也大，后者容量小，但是访问速度快。高速缓存用于解决 DRAM 访问延迟大的问题，利用数据的空间局部性和时间局部性来减少 CPU 访问主存的次数。需要注意的是，8086 是早期的处理器，当时还没有研制出 DRAM，所有的程序和

数据都放在 SRAM 或 ROM 中。

5.4 节

- ✧ CPU 通过哪些信号对存储器进行访问？
- ✧ 什么是地址空间？为什么需要给不同的存储器芯片分配不同的地址空间？
- ✧ 如何给不同的存储器芯片分配不同的地址空间？
- ✧ 138 译码器真值表
- ✧ 什么是线译码？它的特点是什么？
- ✧ 什么是部分译码？它的特点是什么？
- ✧ 什么是全译码？它的特点是什么？
- ✧ 存储器译码信号中，M/IO 必须为高电平还是低电平？
- ✧ 如何实现存储器的位扩展？
- ✧ 如何实现存储器的字扩展？
- ✧ 如何利用位扩展和字扩展为 8086 设计所需要的存储系统？

本章将在课后用 Modelsim 软件采用 VerilogHDL 语言通过位扩展和字扩展为 8086 设计指定地址空间的存储系统

课后习题

1~3
12~13
15~18

第六章

6.1.1~6.1.3 掌握

6.2 8255 方式 0

6.3 8255 的应用

第六章 6.1.1~6.1.2

- ✧ IO 接口用于解决 CPU 与外设数据交换中的哪些问题？
- ✧ 常用的 IO 接口种类有数据接口和状态接口，用于实现 CPU 与外设之间的数据传输和状态交互；
- ✧ IO 的寻址方法有哪两种？各有什么特点？

第六章 6.1.3

- ✧ 了解如何构造简单输入接口和简单输出接口
- ✧ 想一想什么情况下可以采用无条件输入/输出方式与外设交互？
- ✧ 查询式发送接口的组成和发送过程
- ✧ 查询式接收接口的组成和接收过程
- ✧ 了解中断方式进行数据传输的过程，其优缺点是什么？什么情况下使用中断方式传输比较好？
- ✧ 什么是 DMA？为什么 DMA 传输方式效率高？

勘误：图 6.3 和 6.5 中的触发器均应为锁存器。

本节将在课后用 Modelsim 软件采用 VerilogHDL 语言进行简单 IO 接口的电路连线设计和波形观察

6.2 节

- ✧ 什么是并行接口？
- ✧ 8255 是什么器件？有几个地址端口？端口地址由什么决定？
- ✧ 说明对 8255 控制口进行写操作时施加在 8255 引脚上的信号是怎样的
- ✧ 若为 8255 分配的基地址为 20H（设未使用的低位地址为 0），端口 D7:0 接 8086 的数据总线 D15:8，则 8255 各端口地址可以设计为：
(1) 20H, 22H, 24H, 26H
(2) 21H, 23H, 25H, 27H
(3) 22H, 24H, 26H, 28H
(4) 23H, 25H, 27H, 29H

答：（2）

- ✧ 8255 的 RESET 引脚有效时，ABC 端口的状态如何？为什么要这样设计芯片的功能？答：复位后缺省状态均为输入，在执行初始化程序前，端口如果为输出状态而所连接的外设也为输出，则会发生输出冲突。
- ✧ C 口如何实现按位输出控制？
- ✧ A, B 口能否实现按位输出控制？

此处将在课后使用 Modelsim 完成简单 IO 的电路设计

✧

6.3 8255 应用举例

- ✧ 8255 构造简单输入输出接口的电路和初始化方法；
- ✧ 了解行扫描法的工作原理，画出 P225 键盘扫描程序流程图

课后习题

1~4

7~12

13 与 8086 连接，并将端口地址改为 80H，82H，84H，86H，再进行后续的设计，需要考虑奇偶地址问题

14 与 8086 连接，并将端口地址改为 61H，63H，65H，67H，再进行后续的设计，需要考虑奇偶地址问题

第七章

定时计数器的概念

8253 方式 0 和方式 3

其它工作方式不需要掌握

8254 可以了解，不要求掌握

第 7 章引言和 7.1 节

◇ 什么是定时？什么是计数？二者的区别是什么？

◇ 8253 有几个端口？分别做什么用途？

◇ 8253 如何通过控制外部引脚实现对各个端口的读写操作？

◇ 对某个 8253 通道进行初始化需要哪些步骤？

答：步骤 1，对控制口写入该通道的控制字，设置其工作方式、计数值表示方式和读写方式，步骤 2，对该通道写入计数值，注意两个步骤中写入地址是不同的

◇ 若某个时刻 8253 端口 A1:0 接‘11’，/WR 和 /CS 为低电平，D7:0=01010011B，将执行什么操作？**答：向控制端口执行写操作，具体内容**

为对通道 1 进行初始化，设置其为工作方式 1，BCD 码表示计数值，只读写计数器的低字节

◇ 若要读取通道 0 当前的计数值应执行什么操作？**答：应首先对控制寄存器写入 00 00 0000，以锁定计数器 0 的当前计数值，再读取计数器 0，得到该计数值；**

◇ 注意方式 0 和方式 3 有哪些不同，根据其特点决定具体应用

7.2 8253 练习，

◇ 例 7.2 有光方式方式 2 个方式 5，不要求；

◇ 例 7.3 全部使用方式 3 能否实现题目的要求？

◇ 7.2.2 所示的工件计数例子需要学习

◇ 学习 7.2.3 所介绍 8253 在 PC 机中的应用，其中方式 2 实现的刷新触发是否可以用方式 3 代替？**答：如果刷新触发为边缘触发是可以的**

本章将在课后用 Modelsim 软件采用 VerilogHDL 语言进行 8253 应用电路设计和初始化编程练习

课后习题

1 回答方式 0 和方式 3 的特点

6 将端口地址改为 F1H, F3H, F5H, F7H，并回答第一问

第八章

8.1 中断的相关概念

8.2~8.3.1 8259A 相关内容不学习

8.3.2 中断向量设置和中断服务程序编写

第八章 8.1.1 节

◇ 什么是中断？理解中断在信息处理系统中的应用

◇ 了解中断源、中断类型号、中断服务程序、中断响应、中断屏蔽、中断优先级、中断嵌套等概念

◇ 8086 的中断源有哪几类？

◇ 8086 两种外中断有什么不同？其中 INTR 被 8086 响应的条件是什么？

◇ 如果通过指令控制 8086 响应或者不响应 INTR？**答：STI 指令对 IF 置位，允许响应外中断；CLI 指令对 IF 清零，禁止响应外中断**

◇ 8086 内中断由哪几种情况引发？这些中断请求可以被屏蔽么？

◇ 中断类型号的作用是什么？如何通过中断类型号找到中断服务程序入口地址？

◇ 8086 的哪些中断允许嵌套？

8.1.2 节

◇ 8086 响应中断的时刻和过程

◇ 8086 响应中断时自动执行什么堆栈操作？**答：将发生中断请求时下一条指令的段地址和偏移地址压栈（即当前的 CS 和 IP 的值），当前**

标志寄存器压栈；随后 8086 通过中断类型号找到中断向量表将中断向量设置到 CS 和 IP，这样指令指针就指向中断服务程序，并开始执行；

- ✧ 8086 退出中断需要执行什么指令，该指令执行什么堆栈操作？答：退出中断服务程序是通过 IRET 指令，该指令将当前栈顶的数据弹出，依次放入标志寄存器、IP 和 CS，则指令指针重新指向发生中断时的指令的下一条指令继续执行；
- ✧ 8086 响应外中断时如何获得中断类型号？

8.3.2 节

- ✧ 了解 8087 微处理器使用中断前需要进行哪些编程准备。答：编写中断服务程序，并在主程序开始时将该中断服务程序的入口地址写入中断向量表中该中断类型号对应连续 4 个字节的存储单元中
- ✧ 如何为某中断类型号设置中断向量表？
- ✧ 学习例 8.18，了解中断相关程序的设计方法，回答以下问题（1）此例中使用的中断类型号是多少？（2）中断服务程序中如何进行现场的保护和恢复，为什么要这么做？（3）这个例子中主程序在做什么？

本章将在课后用 Modelsim 软件采用 VerilogHDL 语言进行外中断相关电路设计和中断向量设置的编程练习

课后习题

1~4

6~8

10, 11

20