

考点一 了解微机并行通信的基本概念，了解在并行通信过程中联络信号及其作用，理解并行通信的实现过程。

1. 基本概念

(1) 并行传送和串行传送

在数据传送过程中，一个字或一个字节各位同时被传送，这种传送方式称为并行传送。而在数据传送过程中，一个字或一个字节的数据一位一位被传送，这种传送方式称为串行传送。

并行通信由并行接口来完成，在并行数据传送中，并行接口连接 CPU 与并行外设的通道。并行接口中各位数据都是并行传输的。他以一个字或一个字节为单位与 I/O 设备或被控对象进行数据交换，并行通信以同步方式传输，特点是传输速度快，硬件开销大，只适合近距离传输。一个并行接口中包括状态信号控制信号和数据信号。

串行通信是微机和外部设备交换信息的方式之一，所谓串行通信是通过一位一位地进行数据传输来实现通信。与并行通信相比，串行通信具有传输线少，成本低等优点，适合远距离传送，缺点是速度慢，若并行传送 n 位数据需要时间 T ，则串行传送的时间最少为 nT 。

(2) 并行接口和串行接口

I/O 接口与 I/O 设备之间数据传送采用并行传送，则称为并行接口。而 I/O 接口与 I/O 设备之间数据传送采用串行传送，则称为串行接口。

注: 不论并行接口还是串行接口，其与系统总线的数据传送都采用并行传送。

考点二 掌握并行接口 8255A 的内部结构与引脚结构，了解 8255A 的功能特性。

8255A 的内部结构

主机与外围设备之间传送数据有多种方法并且每一种方法有基本的接口电路。

随着集成电路技术的不断发展，接口电路早已集成化，并出现了许多可编程接口

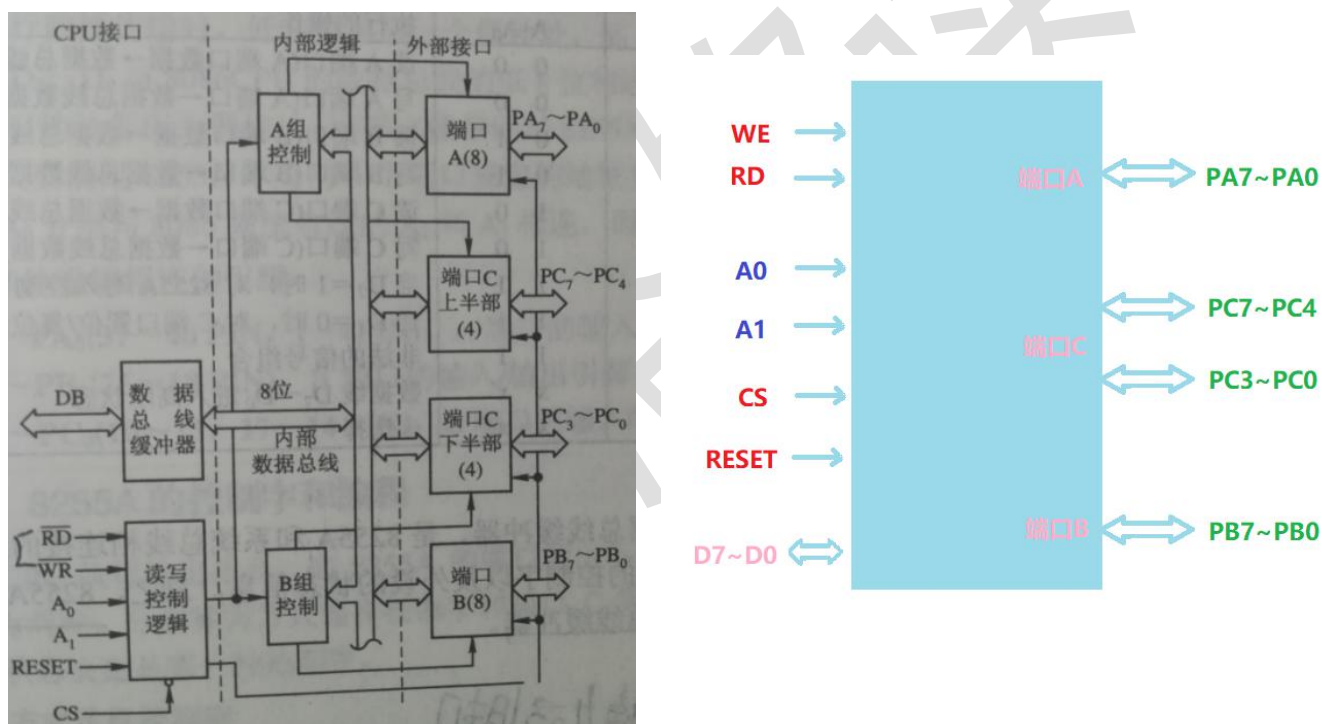
芯片。

所谓可编程接口芯片，是指可通过程序设定其功能或工作方式的接口芯片。

为了具备通用性，通常这些芯片被设计成具有多项功能或多种工作方式，用户在使用时通过编程来选择自己所需要的功能或工作方式。

8255A 就是 Intel 生产的一种典型的可编程并行通信接口芯片，其功能与通用性都较强，使用也很灵活。通过学习要理解 8255 的编程方法以及应用。

8255A 的内部结构框图如图所示，其内部由以下四部分组成。



(1).端口 A、端口 B 和端口 C

端口 A、端口 B 和端口 C 都是 8 位端口，可以选择作为输入或输出,用于 8255A 向外设输入输出 8 位并行数据。

可以将端口 C 的高 4 位和低 4 位分开使用，分别作为输入或输出。

当端口 A 和端口 B 作为输入或输出的数据端口时，端口 C 的指定位与端口 A 和端口 B 配合使用，用作控制信号或状态信号。

(2) A 组和 B 组控制电路

这是两组根据 CPU 送来的工作方式控制字控制 8255A 工作方式的电路。它们的控制寄存器接收 CPU 输出的方式控制字，由该控制字决定端口的工作方式，还可以根据 CPU 的命令对端口 C 实现按位置位或复位操作。

置位:使该位输出为“1”

复位:便该位输出为“0”

(3).数据总线缓冲器

这是一个 8 位三态数据缓冲器，8255A 正是通过它与系统数据总线相连，实现 8255A 与 CPU 之间的数据传送。

输入数据、输出数据、CPU 发给 8255A 的控制字等都是通过该部件传递的。

(4).读/写控制逻辑

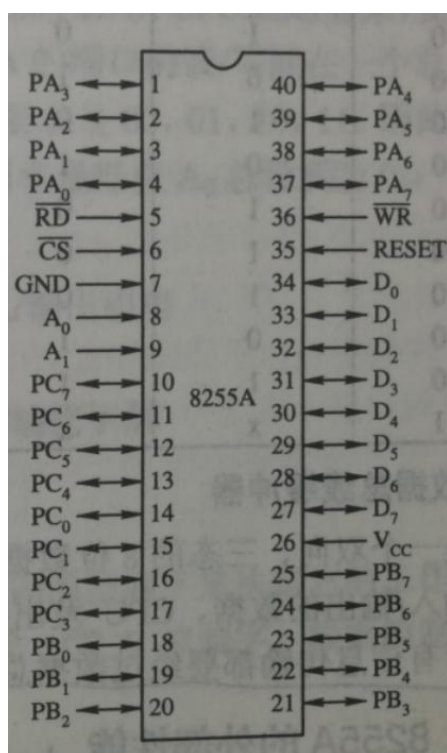
读/写控制逻辑电路的功能是负责管理 8255A 与 CPU 之间的数据传送过程。它接收 CS#及地址总线的信号 A1、A0 和控制总线的控制信号 RESET、RD#和 WR#，将它们组合后，得到对 A 组控制部件和 B 组控制部件的控制命令，并将命令送给这两个部件，再由它们控制完成对数据、状态信息和控制信息的传送。

各端口读/写操作与对应的控制信号之间的关系如表所示:

由 CS#、A1、A0、RD#、WR#引脚的不同组合，实现各种不同的功能。

\overline{CS}	A1	A0	\overline{RD}	\overline{WR}	操作
0	0	0	0	1	读端口A
0	0	1	0	1	读端口B
0	1	0	0	1	读端口C
0	0	0	1	0	写端口A
0	0	1	1	0	写端口B
0	1	0	1	0	写端口C
0	1	1	1	0	写控制字
0	1	1	0	1	非法状态
1	X	X	X	X	数据总线高阻状态
0	X	X	1	1	数据总线高阻状态

8255A 引脚



8255A 是可编程的并行输入/输出接口芯片，它具有三个 8 位并行端口(A 口、 B 口和 C 口)，具有 40 个引脚，双列直插封装，由+5V 供电。

8255A 芯片引脚信号除电源 V_{CC} 和接地引脚(GND)外，其他引脚可分成两组：

(1)8255A 与外设相连的引脚

8255A 与外设连接的有 24 个双向、三态引脚，分成三组，分别对应于 A、B、C 三个端口:PA7~PA0,PB7~PB0,PC7~PC0。

(2)8255A 与 CPU 相连的引脚

D7~D0:双向、三态数据线

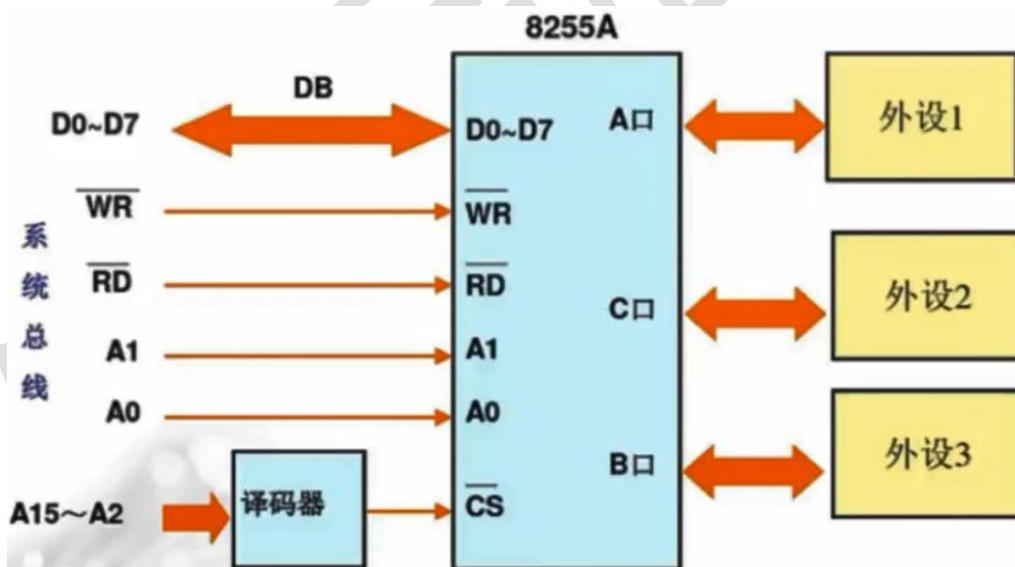
RESET:复位信号, 高电平有效。复位时所有内部寄存器清除, 同时其 3 个数据端口被自动设为输入端口。

CS#:芯片选择信号, 低电平有效。该信号有效时, 8255A 被选中。

RD#: 读信号, 低电平有效。该信号有效时, CPU 可从 8255A 读取输入数据或状态信息。

WR#:写信号, 低电平有效。该信号有效时, CPU 可向 8255A 写入控制字或输出数据。

A1、A0 片内端口选择信号。8255A 内部有三个端口和一个控制端口。规定当 A1、A0 为 00 时, 选中 A 端口, 为 01 时选中 B 端口;为 10 时, 选中 C 端口;为 11 时, 选中控制端口。



8255A 与系统的连接示意图

(1)8255A 的数据引脚只有 8 个。使用在 8 位数据总线的微机系统(如 8088 系统)中, 一般将 8255A 的两个地址引脚 A1 和 A0 分别接至系统地址总线的 A1 和 A0

上, 因而占用连续的 4 个 I/O 端口地址。具体分配如图:



(2) 使用在 16 位数据总线的微机系统(如 8086 系统)中, 一般将 8255A 的两个地址引脚 A₁ 和 A₀ 分别接至系统地址总线的 A₂ 和 A₁ 上, 若 8255A 的数据引脚在系统数据总线的低 8 位上(D₇~D₀), 此时 A₀=0, 则占用 4 个连续的 I/O 偶地址, 如图:



(3) 若 8255A 的数据引脚接在系统数据总线的高 8 位上(D₁₅~D₈), 则占用 4 个连续的 I/O 奇地址, 具体如图:



习惯上将 8255A 的 8 位数据线接到 8086CPU 的低 8 位数据总线上:D₀~D₇ 上, 硬件连接要求 8255A 的 4 个端口必须是偶地址端口。

考点三 掌握 8255A 的方式控制字和 C 口的按位置位/复位控制的格式及使用。

(1) 最高位 D7 为 1, 是方式选择控制字的标志。

(2) 紧接着的 4 位, 即 D6、D5、D4 和 D3 位用来对 A 组进行设置。

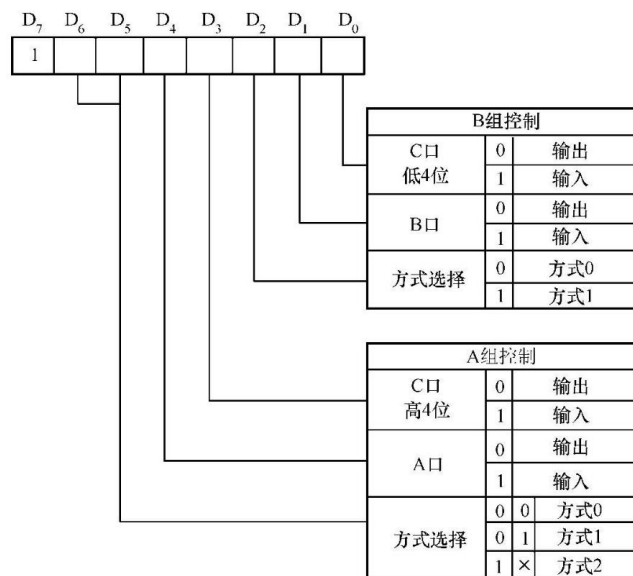
(3) 最低的 3 位, 即 D2、D1 和 D0 位用来对 B 组进行设置。

(4) A 组可设置成三种工作方式中的任意一种, 因此使用两位(D6、D5)来规定工作方式

(4) B 组只能选择方式 1 和方式 0, 所以规定工作方式只需要一位(D2)。

(5) D4 和 D3 位分别规定 A 组的 8 位口(A 口)和 4 位口(C 高 4 位口)是输入还是输出: “1”表示输入, “0”表示输出。

(6) D1 和 D0 位分别规定 B 组的 8 位口(B 口)和 4 位口(C 低 4 位口)是输入还是输出: “1”表示输入, “0”表示输出。



1、可以利用软件编程确定 8255A 的 3 个端口工作于何种方式。

(1) 8255A 的各种工作方式由 CPU 对 8255A 写入控制命令字来设定, 这个过程称为“初始化”

(2) 8255A 在初始化编程时, 是利用 OUT 指令由 CPU 输出一个控制字到控制端口的控制寄存器来控制其工作的。根据具体控制要求的不同, 可使用两种不同类型的控制字: 即方式选择控制字和端口 C 置位/复位控制字。

2、8255A 有两种控制字, 即:

(1) 方式选择控制----确定三个端口的工作方式。即方式控制字决定了 8255A 的工作方式。8255A 工作之前必须初始化,也就是将方式控制字写入控制寄存器中,以指定端口的工作方式。

(2)C 口置位 /复位控制字-----确定 C 口某一位的初始状态,或用于设置 INTE(InterruptRequest,中断请求信号)位。

对 8255A 的工作方式的选择:可以分别选择端口 A、端口 B 和端口 C 上下两部分的工作方式。

3、8255A 工作方式初始化步骤:

(1)根据 8255A 与 CPU 接口硬件电路图确定 8255A 四个端口地址。

(2)根据 8255A 的具体应用,确定 8255A 工作方式控制字内容。

(3)利用输出指令,将工作方式控制字送至控制命令寄存器中。

C 口按位置位/复位控制字(置位/复位字)



1. 8255A 的 C 口具有位控功能,即端口 C 的 8 位的任一位都可通过 CPU 向 8255A 的控制寄存器写入一个按位置位/复位控制字来置 1 或清 0,而 C 口中其他位的状态不变。

(1) 该控制字用最高位 D7 为 0 作为标志。

- (2) D3~D1 用来进行行位选择, 即指定对哪一位进行操作。
- (3) D0 用来表示置位(D0 为 1)还是复位(D0 位 0)
- (4) 另有 3 位未用到, 一般取值为 0。
- (5)注意:端口 C 的按位置位/复位控制字必须跟在方式选择控制字之后写入控制字寄存器, 即使仅使用该功能, 也应先选送一个方式控制字。
- (6) 方式选择控制字只需写入一次, 之后就可多次使用端口 C 按位置位/复位控制字对端口 C 的某些位进行置 1 或清 0 操作。

2. 8255A 的 C 口置位/复位操作步骤:

- (1)根据 8255A 与 CPU 接口硬件电路图确定 8255A 4 个端口地址。
- (2)根据对 PC 口具体哪一位操作, 确定 8255A 的按位置位/复位控制字内容。
- (3)利用输出指令, 将按位置位/复位控制字送至 8255A 控制命令寄存器中。

考点四 重点掌握 8255A 的工作方式 0 和方式 1, 掌握 8255A 与系统总线及外设的连接方法。

8255A 有 3 种工作方式

- ①方式 0:基本输入/输出, 适用于无条件传送和查询方式的接口电路;
- ②方式 1:选通输入/输出, 适应于查询和中断方式的接口电路 (需要 C 口提供联络信号)
- ③方式 2:选通双向输入/输出 , 仅适合于 A 口, 适应于双向传送数据的外设, 也适应于查询和中断方式的接口电路。(需要 C 口提供联络信号)

C口联络信号定义 (状态字)				
C口位线	方式1 (A、B口)		方式2 (只有A口)	
	输入	输出	输入	输出
PC ₇		\overline{OBF}_A		\overline{OBF}_A
PC ₆		$\overline{ACK}_A (INTE_A)$		$\overline{ACK}_A (INTE_A)$
PC ₅	IBF_A		IBF_A	
PC ₄	$\overline{STB}_A (INTE_A)$		$\overline{STB}_A (INTE_A)$	
PC ₃	$INTR_A$	$INTR_A$	$INTR_A$	$INTR_A$
PC ₂	$\overline{STB}_B (INTE_B)$	$\overline{ACK}_B (INTE_B)$		
PC ₁	IBF_B	\overline{OBF}_B		
PC ₀	$INTR_B$	$INTR_B$		

A口可选择方式0、方式1和方式2;

B口只能选择方式0和方式1;

C口只能用方式0;

当选择方式0和方式1时, C口通常都是配合A口或B口工作, 作为A口、B口与外设联络用的输出控制信号或输入状态信号, 而C口的其余各位仍用方式0工作。

1、方式0

方式0是基本的输入输出工作方式, 它只能完成简单的并行输入/输出操作。

方式0具有以下特点:

(1) 方式0是一种基本输入/输出工作方式, 通常不用联络信号, 只能无条件传送或按查询方式传送, 而不能采用中断方式来和CPU交换数据。任何一个数据端口(A口, B口, C口的高4位和低4位)都可用方式0作简单的数据输入或输出。

(2) 由A口、B口两个8位并口和C口高4位和C口低4位并口, 共有4个独立的并口, 它们可组合成16种不同的输入/输出组态。同时, 它们也是一种单向的输入/输出传送, 一次初始化只能使所指定的某个端口或者作输入或者作输出。

(3) 8255A 在方式 0 下不设置专用联络信号线。

2、方式 1

方式 1 为选通输入输出方式或应答方式。它在使用端口 A 和端口 B 进行输入/输出时，一定要利用端口 C 所提供的选通信号和应答信号来配合输入/输出操作。

方式 1 的工作特点：

- ①端口 A (PA) 和端口 B (PB) 为数据口，而 PC 口分配作联络信号用，用户不能再指定其他用途。
- ②如果 A 口和 B 口都工作在方式 1，则 C 口中有 6 位(分为两组 3 位)被规定为配合方式 1 工作的信号，剩下两位，仍可以作为输入或输出。
- ③如果 8255A 的端口 A 和端口 B 中只有一个端口工作于方式 1，那么，端口 C 中有 3 位被规定为配合方式 1 工作的信号，此时，另一个端口可以工作在方式 0。端口 C 中的其它数位也可以工作在方式 0，即作为输入或输出。

3、方式 1 输入情况：

(1) 此时 PC3~PC5 和 PC0~PC2 分别用作 A 口和 B 口的状态和控制线，PC6 和 PC7 用作 I/O 线。

(2) STB#: 外设送到 8255A 的“输入选通”信号，低电平有效。当它变为低电平时，将数据锁存到 8255A 端口的输入数据寄存器。

(3) IBF: 8255A 给外设的应答信号，高电平有效。当 IBF=1，说明 8255A 的“输入缓冲器满”，数据尚未被 CPU 取走，通知外设不能送新数据；当 IBF=0 时，说明输入缓冲器已空，允许外设送新数据。

(4) INTR: 8255A 送到 CPU 的“中断请求”信号，高电平有效。当 INTR=1 时，

请求 CPU 从 8255A 读数。使 INTR 变为高电平的条件是: STB#=1 (即数据已输入 8255A 的输入锁存器), IBF=1 (输入缓冲器满), INTE=1 (中断请求被允许), 三个条件都具备时, 才使 INTR 变高, 向 CPU 发出中断请求。

(5) INTE: “中断允许”信号 INTE 是 8255A 为控制中断而设置的内部控制信号。当 INTE=1 时, 允许中断, 当 INTE=0 时, 禁止中断。这要通过向 C 口写入按位置位/复位命令来设置, 内部不能自动产生该控制信号。

4、方式 1 输出情况:

这时 PC3、PC6 和 PC7 作 A 口的应答信号, PC0、PC1 和 PC2 作 B 口的应答信号。余下的 PC4 和 PC5 可作输入或输出线。

各应答信号的含义:

(1) OBF#: 输出缓冲满信号, 输出, 低电平有效。表示 CPU 已输出数据到指定端口。

(2) ACK#: 外设的应答信号, 低电平有效, 由外设送给 8255A。表示 CPU 送到指定端口的数据已被外设接受。

(3) INTE: 中断允许信号。INTEA、INTEB 是由用户对 PC6、PC2 按位置位实现的。

(4) INTR: 中断请求信号, 高电平有效。ACK#=1、OBF#=1 且 INTE=1 时, INTR=1。

考点五 了解计数/计时的具体应用, 理解计数/计时的实现原理。

1. 基本概念

(1) 定时/计数应用场合

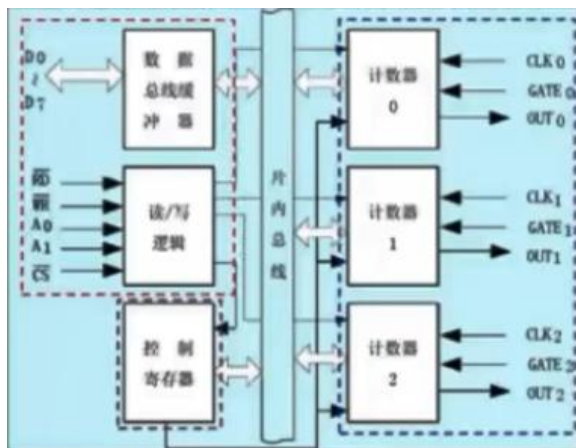
在微机应用系统中, 常会遇到二类问题。一类问题是需要外部实时时钟, 以实现定时功能。如微机中的时钟的定时信号、SRAM 的定时刷新信号、控制中的定时采

样信号等。另一类问题是需要计数器,以实现计数功能。如外部事件发生的次数、生产产品的件数等。

(2) 定时/计数的实现

实现定时,计数的基本方法:利用软件定时(延时子程序)使用硬件电路(数字逻辑电路——如单稳态定时电路等)。使用可编程定时器/计数器芯片。

考点六 掌握定时计数器 8253 的内部结构与引脚结构。掌握其工作原理。



与系统总线相连

数据引脚 D7~D0

地址引脚 A0, A1

控制引脚 RD#, WE#, CS#

通道引脚

CLK

外部计数或定时脉冲的输入端。每个通道工作时, 都是对输入到CLK引脚上的脉冲按二进制或十进制(BCD码)进行减1计数。

GATE

门控信号输入端。用于启动或禁止计数通道工作, 以使计数通道和计测对象同步。

OUT

计数通道的输出端。以相应的电平指示计数完成, 或输出脉冲波形表示定时时间。

可编程计数器/定时器 8253 用软、硬技术相结合的方法来实现定时和计数控制, 其特点主要有以下几个:

- (1) 有 3 个独立的 16 位计数器，每个计数器均以减法计数。
- (2) 每个计数器都可按二进制计数或十进制(BCD 码)计数。
- (3) 每个计数器都可以由程序设置 6 种工作方式。
- (4) 每个计数器计数速度可达 2.6MHz。
- (5) 所有 I/O 都可与 TTL 兼容。

6.1 8253 内部结构

8253 由数据总线缓冲、读/写逻辑、控制字寄存器以及三个独立的 16 位计数器组成。

(1)数据缓冲器:

- ① 三态、双向 8 位缓冲器。
- ② 数据缓冲器用于 8253 和系统数据总线的连接。
- ③ CPU 通过数据缓冲器 将控制命令字和计数值写入 8253 计数器，或者从 8253 计数器中读取当前的计数值。

(2)读/写逻辑

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
SC ₁	SC ₀	RL ₁	RL ₀	M ₂	M ₁	M ₀	BCD

读/写逻辑的任务是接收来自 CPU 的控制信号，完成对 8253 内部操作的控制。

这些控制信号包括读信号 RD#、写信号 WR#、片选信号 CS#以及用于片内寄存器寻址的地址信号 A0 和 A1。

当片选信号有效，即 CS#=0 时，读/写逻辑才能工作。

该控制逻辑根据读/写命令及送来的地址信息，决定三个计数器和控制寄存器中

的哪一个工作，并控制内部总线上的数据传送的方向。

8253 共占用 4 个 IO 地址。

- ①当 A1A0=00 时， 为计数器 0 中的 CR 和 OL 寄存器的共用地址。
- ②A1A0=01 和 10 时， 分别为计数器 1 和 2 的 CR 和 OL 的共用地址。
- ③当 A1A0=11 时， 是三个计数器内的三个控制寄存器的共用的地址号。
- ④CPU 给哪一个计数器送控制字，由控制字格式中最高两位(计数器选座位) SC1、SC0 的编码来决定。
- ⑤计数工作单元 CE 不能用指令访问，它们不占用 I/O 地址。

(3) 控制寄存器

- ① 控制寄存器用来保存来自 CPU 的控制字。
- ② 每个计数器都有一个控制命令寄存器，用来保存该计数器的控制信息。
- ③ 控制字将决定计数器的工作方式、计数形式及输出方式，也决定如何装入计数初值。
- ④ 8253 的 3 个控制寄存器只占用一个地址号，而靠控制字的最高两位来确定将控制信息送入哪个计数器的控制寄存器中保存。
- ⑤ 控制寄存器只能写入，不能读出。

(4)三个独立的 16 位计数器

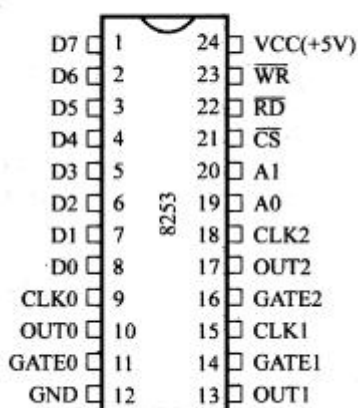
- 1) 每个计数器具有相同的内部结构。
- 2) 包括一个 8 位的控制寄存器、一个 16 位的计数初值寄存器 CR、一个 16 位的减 1 计数器 CE 和一个 16 位的输出锁存寄存器 OL。
- 3) 16 位的计数初值寄存器 CR 和 16 位的输出锁存寄存器 OL 共同占用一个 I/O 端口地址，CPU 用输出指令向 CR 预置计数初值，用输入指令读回 OL 中的数

值，这两个寄存器都没有计数功能，只起锁存作用。

- 4) 16 位的减 1 计数器 CE 执行计数操作，其操作方式受控制寄存器控制，最基本的操作是:接受计数初值寄存器的初值，对 CLK 信号进行减 1 计数，把计数结果送输出锁存寄存器中锁存。

6.2 8253 引脚功能

8253、TMP82C53 可编程间隔计数器



8253 是 24 脚双列直插式芯片，用+5V 电源供电。

芯片内有三个相互独立的 16 位定时器/计数器。

- (1)数据引脚 D7~D0:数据线，双向三态，与系统数据总线连接。
- (2) 片选信号 CS#:输入信号，低电平时选中此片。由 CPU 输出的地址经地址译码器产生。
- (3)地址线 A0、A1: 这两根线接到系统地址总线的 A0、A1 上，当 CS#为低电平，且 8253 被选中时，用它们来选择 8253 内部的 4 个寄存器。
- (4)读信号 RD#:输入信号，低电平有效。由 CPU 发出，用于控制对选中的 8253 内寄存器的读操作。
- (5)写信号 WR#:输入信号，低电平有效。由 CPU 发出，用于控制对选中的 8253

内部寄存器的写操作。

(6)时钟脉冲信号 CLK0~CLK2:计数器 0、计数器 1 和计数器 2 的时钟输入端。由 CLK 引脚输入的脉冲可以是系统时钟或其他任何脉冲源所提供的脉冲。该脉冲可以是均匀的、连续的并具有精确周期的,也可以是不均匀的、连续的、周期不确定的脉冲。时钟脉冲信号的作用是在 8253 进行定时或计数时,每输入一个时钟信号,就使计数值减 1。若 CLK 是由精确的时钟脉冲提供,则 8253 作为定时器使用;若 CLK 是由外部事件输入的脉冲,则 8253 作为计数器使用。

(7)门控脉冲信号 GATE0~GATE2,计数器 0、计数器 1 和计数器 2 的门控脉冲输入端,是由外部送入的门控脉冲,该信号的作用是控制启动定时器/计数器工作。

(8)输出信号 OUT0~OUT2:计数器 0、计数器 1 和计数器 2 的输出端。当计数器计数到 0 时,该端输出一个标志信号,从而产生不同工作方式时的输出波形。

6.3 8253 的工作过程

1、8253 计数器在工作之前,用户必须对其进行初始化编程:

首先,CPU 用输出指令向控制寄存器送控制字;

然后,在用输出指令向计数初值寄存器 CR 预置计数/定时的初值。

启动工作后,CR 中的初值就送入减 1 计数器 CE 对 CLK 输入的计数/定时脉冲信号进行减 1 计数。当 CE 中的内容减为 0,表示计数/定时到,则 OUT 端输出信号。输出信号的波形形式由工作方式决定。

2、8253 的级联

8253 的通道间,一个通道的输出 OUT 作为其它通道的输入 CLK,这种情况称为级联。

当定时 或计数初值大于一个通道的最大计数初值时，需要通道级联来扩大计数范围。

3、8253 定时计数器与 CPU 的关系

8253 定时计数器操作过程与 CPU 相互独立，并行操作。

(1)数据总线的连接

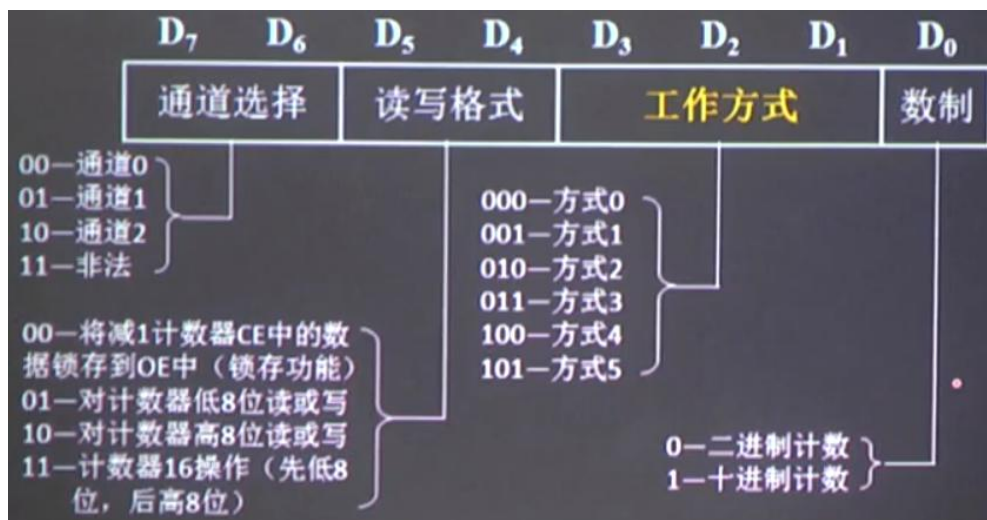
- ① 若 CPU 的数据总线是 8 位，则 8253 将数据线按位与系统数据总线相连;
- ② 若 CPU 的数据总线是 16 位则需要根据 CPU 在使用数据总线时的具体规定来连接。
- ③ 8086 在访问 I/O 端口时，当按 8 位来访问时，如果 IO 端口地址是偶地址，则使用低 8 位数据线;如果 IO 端口地址是奇地址，则使用高 8 位数据线。因此连接时，需要根据 8253 端口的地址类型，来确定 8253 是与系统的低 8 位相连，还是与系统的高 8 位相连。

(2)地址总线的连接

- ① 如果 CPU 的数据总线是 8 位，由于不需要区分奇偶地址，所以 8253 的端口地址可以连续，一般直接将 8253 的 A1、A0 系统地址总线的 A1、A0 同名端相连。
- ② 如果 CPU 的数据总线是 16 位)由于需要区分奇偶地址，连接时，一般就将系统地址总线的 A2、A1 与 8253 的 A1、A0 相连，而系统地址总线的 A0 参与地址译码来确定片选信号。

6.4 8253 的初始化编程

1. 8253 控制字



(1) D0: 数制选择指示计数器按哪一种数制计数 D0=0, 按二进制计数, 计数范围 16 位二进制数 0000H~FFFFH。而 0000H 是最大数, 代表 65536 故按二进制计数值的实际范围为 1~65536D, D0=1, 按二-十进制(BCD)计数, 其计数范围为 4 位十进制数, 数值范围是 0000D~9999D, 而 0000 是最大数, 代表 10000D, 其实际计数范围是 1~10000D。

(2) D3、D2、D1 计数器工作方式选择, 6 种工作方式。

(3) D5、D4 数据读/写格式, 决定 CPU 对计数器的读/写情况。

(4) D7、D6 选择计数器, 选择哪个计数器工作

2. 初始化流程

先写入工作模式控制字, 接着写入计数初始值

初始化命令

(1) 对计数器设置初值前必须先写入控制字, 用来选择计数器、设定工作模式和计数格式;

(2) 设置初值命令用来给出计数的初始值, 初值可以是 8 位, 也可以是 16 位;

设置计数初始值的说明

计数初始值写入对应计数器的计数初值寄存器;

① 计数初值为 8 位则控制字中的 RW1、RW0。应取 01, 初值只写入 CR 的低 8 位, 高 8 位会自动置 0;

② 若是 16 位初值, 而低 8 位是 0, 则 RW1、RW0。应取 10, 初值高 8 位写入 CR 的高 8 位, CR 的低 8 位会自动置 0;

③ 若是一般 16 位初值, 则 RW1, RW0 应取 11, 应分两次写入初值, 先写低 8 位、再写入高 8 位(此顺序不能反)

写各计数器的控制字

由 OUT 指令向控制字寄存器写入, 用来选定计数通道、规定该计数器的工作方式和计数方式。

```
MOV AL, 74H      ;控制字
MOV DX, 123H     ;控制字寄存器地址
OUT DX, AL
```

设置计数初始值

由 OUT 指令向某计数器送一个计数初值, 可以是 8/16 位数据。

若是 8 位数据, 一条 OUT 指令即可完成初值的设置;

若为 16 位数据, 必须用两条 OUT 指令来完成, 且先送低 8 位数据, 后送高 8 位数据。

```
MOV DX, 121H     ;计数器 1 地址
MOV AL, 00H      ;低 8 位字节送 AL
OUT DX, AL       ;输出到计数器
```


MOV AL, 20H ;高 8 位字节送 AL
OUT DX, AL ;输出到计数器

计数初值的计算有两种方法:

- (1)计数时钟频率除以定时频率
- (2)定时周期除以计数时钟周期

因此要先计算定时频率或计数时钟周期。因为频率和时钟互为倒数

例: 将 8253 的计数器 1 作为 5ms 定时器, 设输入时钟频率为 200KHz, 试编写 8253 的初始化程序。(工作于方式 0, BCD 码计数)

(1)计数初值 n 计算

已知输入时钟 CLK 频率为 200KHz, 则时钟频率为 $T=1/f=1/200\text{KHz}=5\mu\text{s}$, 于是计数初值 n 为: $n=5\text{ms}/T=5\text{ms}/5\mu\text{s}=1000$ 。

(2)确定控制字

按题意选计数器 1, 按 BCD 码计数, 工作于方式 0 由于计数初值 $n=1000$, 控制字 D5D4 应为 11, 于是 8253 控制字为: 01110001B=71H。

(3)选择 8253 各端口地址

设计数器 1 的端口地址为 3F82H, 控制口为 3F86H

初始化流程的注意事项

计数器的顺序是任意的, 不必一定按照计数器 0、1、2 的顺序初始化;

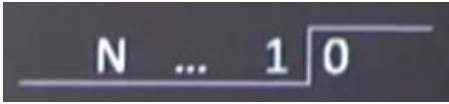
可先写所有计数器的工作模式控制字, 再装入各计数器的计数值(但先控制字再计数值的顺序不能错);

计数值先写低再写高的顺序不能错;其他顺序无关紧要

6.5 8253 的工作方式

8253 有六种工作方式，也即六种输出波形;

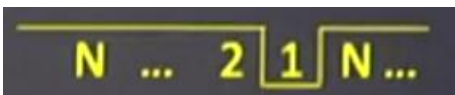
①方式 0:计数结束中断方式（软件控制）



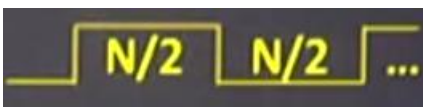
②方式 1:可编程单稳态输出方式（硬件控制）



③方式 2:比率发生器(分频器)（软件控制）



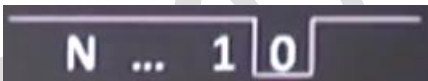
④方式 3:方波发生器（软件控制）



⑤方式 4:软件触发选通（软件控制）



⑥方式 5:硬件触发选通（硬件控制）



工作方式	功能	N与输出波形关系	GATE 作用	计数过程中改变N
0	计数结束中断方式	写入N后开始计数 计数器为0, OUT输出H	0: 禁止计数 1: 允许计数	立即有效
1	可编程单稳态	外部GATE触发 OUT输出 单拍负脉冲宽度N个CLK	GATE上升沿启动 下一个CLK下降沿 开始计数, 计数值为0 输出上升沿	GATE 触发后有效
2	速率发生器	每N个CLK输一个宽度为 一个CLK周期的负脉冲 不断重复	GATE=0, 禁止计数 GATE=1, 允许计数 GATE为上升沿重新 装入N启动计数	计数过程中可改变 计数值但需计数到1 后有效
3	方波发生器	N=偶, N/2个1, N/2个0 N=奇, (N+1)/2个1, (N-1)/2个0 不断重复	GATE=0, 禁止计数 GATE=1, 允许计数 GATE为上升沿重新 装入N启动计数	计数过程中可改变 计数值外触发 或计数到0有效
4	软触发选通	写入N后开始计数 计数器为0, OUT输出一个 宽度为CLK周期的负脉冲	GATE=0, 禁止计数 GATE=1, 允许计数	计数过程中可改变 计数值外触发 或计数到0有效
5	硬触发选通	写入N后GATE上升沿 开始计数计数器为0 OUT输出一个宽度为CLK 周期的负脉冲	GATE上升沿 启动计数	GATE上升沿 触发有效