

DMA技术

4.1 DMA传送的特点

- 为什么要使用DMA技术?

在一般的程序控制传送方式中

- 数据从存储器送到外设或从外设到存储器都要经过CPU的中转
- 检查是否传送完毕以及修改内存地址等操作都由程序控制

采用DMA传送解决高速数据传送问题

4.1 DMA传送的特点

- 应用领域

主要应用在需要高速大批量数据传送的系统中，
以提高数据吞吐量

- 磁盘存取（硬盘和软盘I/O）
- 图象处理（对CRT屏幕送数据）
- 高速数据采集系统
- 同步通信中的收/发信号
- 快速通信通道I/O
- 多处理机和多程序数据块传送
- DRAM的刷新操作

4.1 DMA传送的特点

- DMA传送提高了传输速度
 - 存储器与外设，外设与外设之间直接交换数据，不需要经过累加器，减少了中间环节
 - 内存地址的修改，传送完毕的结束报告都由硬件完成
- 以增加系统硬件的复杂性和成本为代价
 - 硬件控制代替软件控制
 - DMA传送期间CPU被挂起，部分或完全失去了对总线的控制，影响CPU对中断请求的及时响应与处理

4.1 DMA传送的特点

- 本章主要说明DMA原理、8237的外部特性和编程使用方法

4.1 DMA传送的特点

- DMA传送方式 (Direct Memory Access)
 - 直接存储器存取方式，即 CPU暂时让出总线控制权，由DMAC完全控制存储器与I/O 之间直接进行数据传送
 - DMAC暂时行使总线控制权，执行地址修改、控制数据传送等功能，**数据传送速度主要受存储器的存取速度限制**

4.2 DMA传送过程

- CPU对DMAC进行初始化操作内容包括：
 - 通道的选择
 - DMA的操作类型及方式
 - 内存首地址
 - 地址递增/递减
 - 需要传送的字节数

4.2 DMA传送过程

1. 申请阶段

- 外设向DMAC发出DREQ申请信号
- DMAC向CPU发出HRQ总线请求信号

2. 响应阶段

- 满足条件下（总线周期结束，HRQ有效，总线锁定信号LOCK无效）

CPU给DMAC发出HLDA总线回答信号，CPU一侧总线浮空，DMAC成为系统的主控者

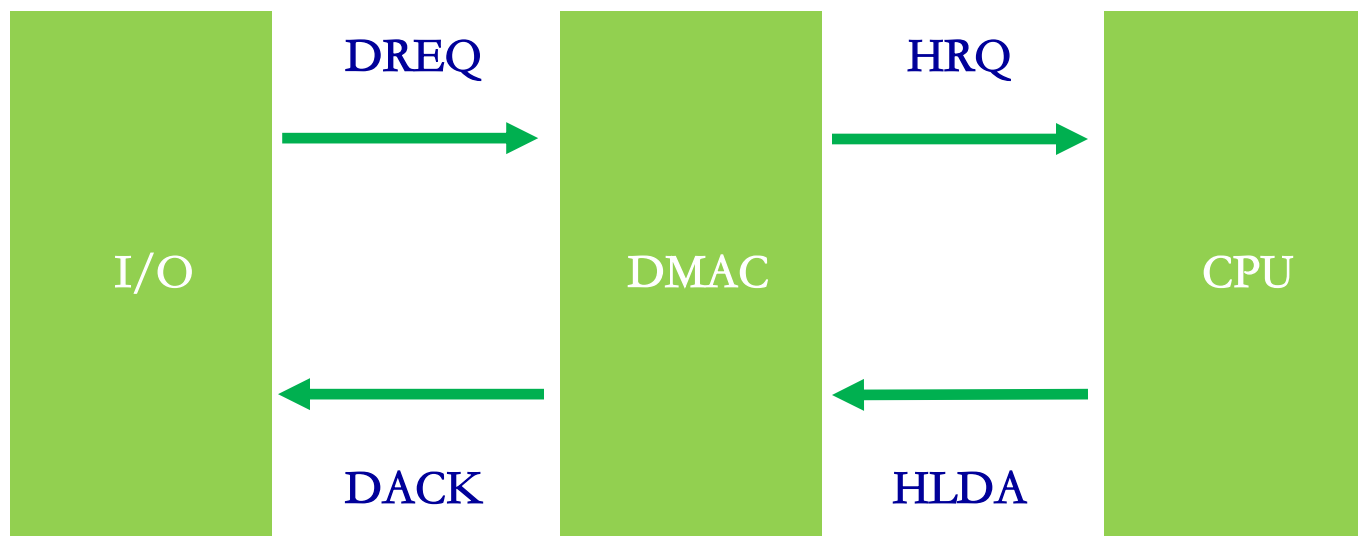
4.2 DMA传送过程

3. 数据传送阶段

- DMAC给外设发出DMACK信号
- DMAC向存储器发地址信号，向存储器和外设发读写控制信号，按初始化设定的方式（向）传送数据

4. 传送结束阶段

- DMAC产生计数已到/过程结束信号给外设
- 外设撤销DMA请求信号DREQ
- 总线请求信号HRQ和总线回答信号HLDA无效，DMAC一侧总线浮空，CPU成为系统的主控者



DMA传送过程

4.3 DMA传送的方式

一、DMA操作类型

1. 数据传送

- 源地址和目的地址都可以是存储器和I/O设备
- DMA读/写定义为从存储器读/向存储器写

2. 数据校验

- 不进行数据传输，DMA控制器不发送存储器或I/O设备的读/写信号

3. 数据检索

- 不进行数据传输，从源地址起始单元开始，逐一读出数据与比较寄存器内的内容进行比较，并在状态字中标记或申请中断

4.3 DMA传送的方式

二、DMA操作方式

1. 单字节方式

- 每传送一个字节DMAC必须重新向CPU申请占用总线

2. 连续方式

- 在整个数据块的传输过程中一直占用总线，不允许其它DMA通道参与竞争，可能产生冲突
- 即使DREQ变为无效时，DMAC不释放总线，而是等待DREQ重新有效

4.3 DMA传送的方式

二、DMA操作方式

3. 请求方式

- 有请求时，DMAC才占用总线，每传一个字节都要检测DREQ
- DREQ变为无效/数据传输结束（T/C）/检索到匹配字节/校验完毕/过程结束信号（/EOP）时，DMAC释放总线

4.4 DMA控制器

一、DMA控制器在系统中的地位

DMAC在系统中的工作状态有两种工作状态：

1. 主动态（主控制器）

- 接管并取得总线控制权，取代CPU成为系统的主控者，向存储器和外设发送命令
- 通过三总线向存储器或I/O发出地址及读/写信号，控制存储器和I/O设备之间的数据传送

4.4 DMA控制器

2. 被动态（受控器）

- 未取得总线控制时，与其他 I/O 设备相同，受 CPU 的控制
- CPU 对 DMAC 进行初始化操作或从 DMAC 读取状态信息

4.4 DMA控制器

DMAC的读/写操作

- “写入”指I/O向存储器中写入

DMA写操作：控制信号，/MEMW，/IOR

- “读出”指I/O从存储器中读出

DMA读操作：控制信号，/MEMR，/IOW

4.4 DMA控制器

二、总线控制权在DMA控制器与CPU之间的转移

DMA在主动态时接受CPU对总线的控制权

转移过程：

1. 让出：

I/O 发请求信号DREQ，DMAC发请求信号HRQ，CPU发应答信号HLDA，DMAC发应答信号DACK

2. 传输：

DMAC向I/O设备发读/写信号，向存储器发读/写信号
I/O向存储器传送数据（DREQ，HRQ，HLDA，DACK信号有效）

3. 收回：数据传送结束，DMAC 向I/O 发过程终止信号/EOP），HRQ失效，CPU重新获得总线控制权

常见 DMA 控制器

型号	内部 寄存器数	通道数	数据块 最大长度 KB	最高速率 MB/s	传送对象
8237A-5	16 2 (不可访问)	4	64	1.5	存储器/IO 存储器/存储器
Z80- DMA	21 (写) 7 (读)	1	64	2	存储器/IO 存储器/存储器 IO/IO
MC68B44	15	4	64	2	存储器/IO

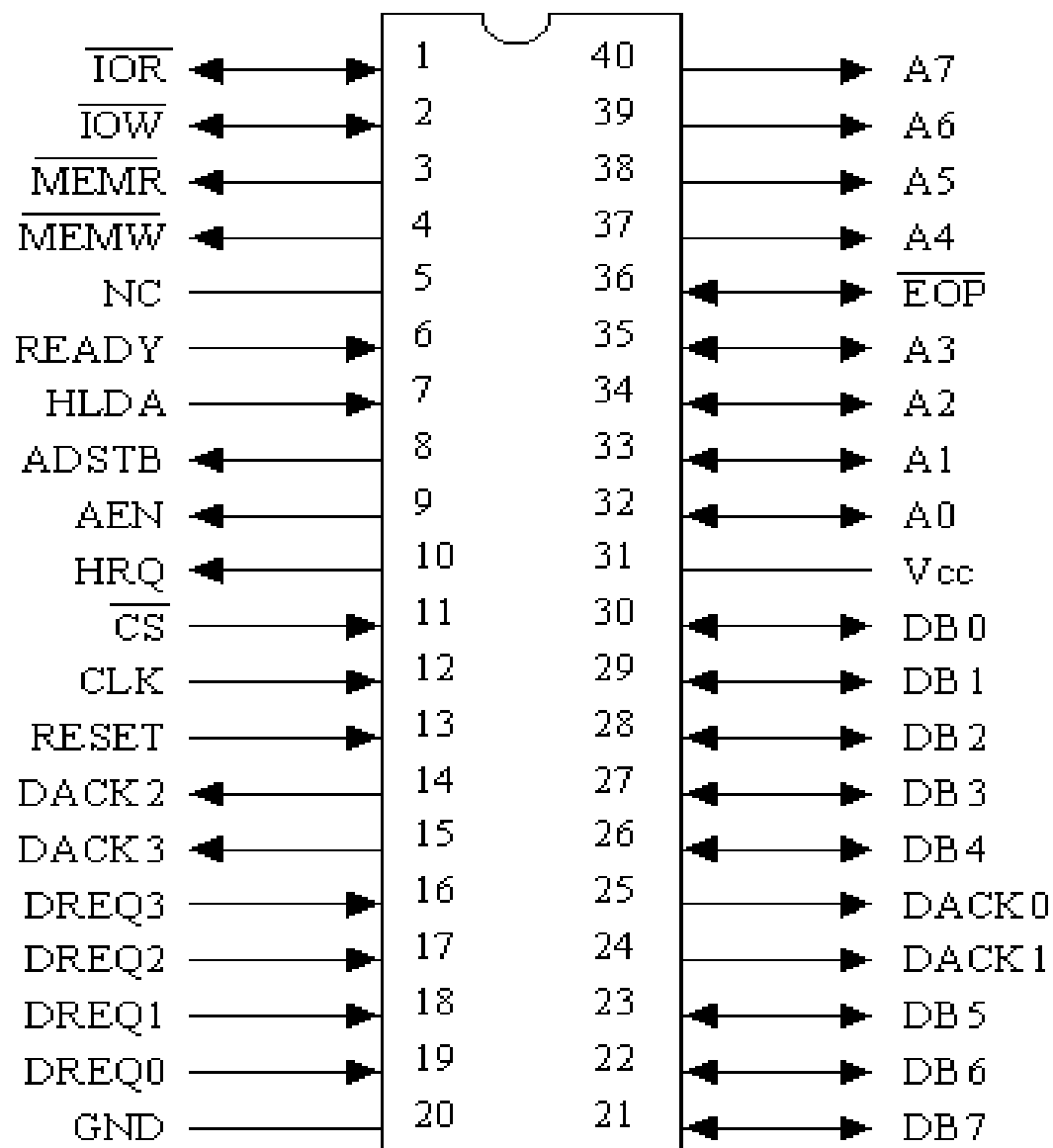
4.4 DMA控制器

三、DMA8237控制器

- 四个独立通道（可对4个I/O设备进行DMA服务）
- 64KB寻址与计数能力
- I/O设备与系统存储器，系统存储器与存储器之间交换数据
- 有级联方式和多种操作模式

4.4 DMA控制器

- CPU对DMAC进行初始化操作内容包括：
 - 通道的选择
 - DMA的操作类型及方式
 - 内存首地址
 - 地址递增/递减
 - 需要传送的字节数



8237A外部引脚

4.4 DMA控制器

- 40个引脚双列直插
 - 面向外设

DRQ0-3 来自4个独立通道外设请求 (0优先级最高)

DACK0-3 DMA控制器发给外设的应答信号 (同一时间只有一个有效)

READY 准备就绪

MEMR#/MEMW# 存储器读写

EOP# 过程结束 (双向)
 - 面向CPU

HRQ DMAC向CPU发出的接管总线请求

HLDA 总线应答

4.4 DMA控制器

- 40个引脚双列直插

- 其它

IOR#/IOW# (双向) I/O设备读写; CPU向DMAC写命令、初始化参数、读状态

CS# 允许CPU与DMAC交换信息

A0-3 (双向三态) CPU对DMAC初始化时内部寄存器和计算器寻址; 20位存储器地址的低4位

A4-7 (单向) 20位存储器地址的低8位中的高4位

DB0-7 (双向三态) CPU对DMAC初始化时传送命令, 结束时传送状态; 16位存储器地址的高8位; 数据线
(地址和数据分时复用)

ADSTB 16位地址的高8位锁存器的输入选通

AEN 高8位地址锁存器的输出允许信号; 禁止其它系统总线驱动器占用总线

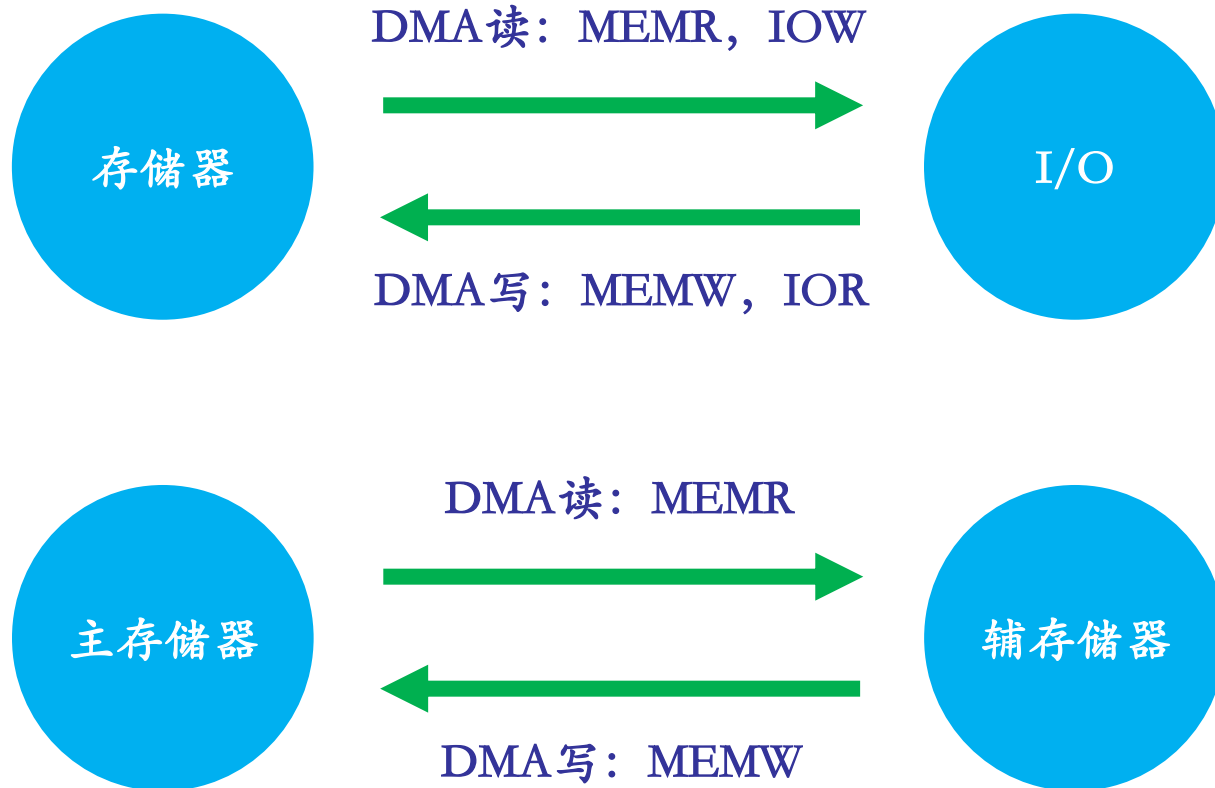
4.4 DMA控制器

一、8237的主要性能参数及引脚功能

1. 8237有4个独立的数据传送通道，每个通道64KB寻址及计数能力，还可通过级联方式进行通道扩充
2. 接收CPU对其进行的初始化操作，输入命令字和传送参数，实现级联和多种模式的操作；向CPU提供状态信息

4.4 DMA控制器

3. 接收从外设发来的DMA请求信号DREQ，向CPU发HRQ总线请求信号；
接收CPU回送的，让出总线控制权的回答HLDA；
向外设发出其对DMA请求的应答信号DACK3~0；
系统允许多个DREQ信号同时有效，即几个I/O同时提出申请，**但同一个时间，只能有一个DACK信号有效**
4. 允许I/O设备与存储器、存储器与存储器之间直接传送数据



DMA 传送

4.4 DMA控制器

1. 在进行数据传送时

- MEMR#/MEMW#为输出
- IOR#/IOW#为输出

DMAC控制对存储器或I/O设备进行读/写操作
在初始化时

- IOR#/IOW#为输入

2. 访问存储器的地址码由DMAC通过地址引脚 A₇~A₀提供地址低8位，由DMAC通过DB₇~₀及 **ADSTB**提供地址高8位

4.4 DMA控制器

3. 数据通过 $DB_{7\sim0}$ 传送（对 $DB_{7\sim0}$ 的使用采用了总线分时复用技术）
4. 如果慢速的I/O设备或存贮器与存贮器交换数据时要求等待，则可以从DMAC的READY引脚输入低电平，等待时间结束，READY引脚变为高电平，表示准备好，可以开始交换数据

4.4 DMA控制器

5. 能确定传送数据的字节数，并判断DMA传送是否结束，并发出DMA操作的结束信号
 - EOP: DMA操作结束，负脉冲有效，双向
 - DMA传送时，每传送一个字节，字节计数器自动减“1”，直至为“0”时，输出EOP传送操作终止信号，将EOP送I/O设备
 - 若从外部向EOP引脚输入一负脉冲，则迫使DMA操作中止，结束数据传送，各内部寄存器复位

4.4 DMA控制器

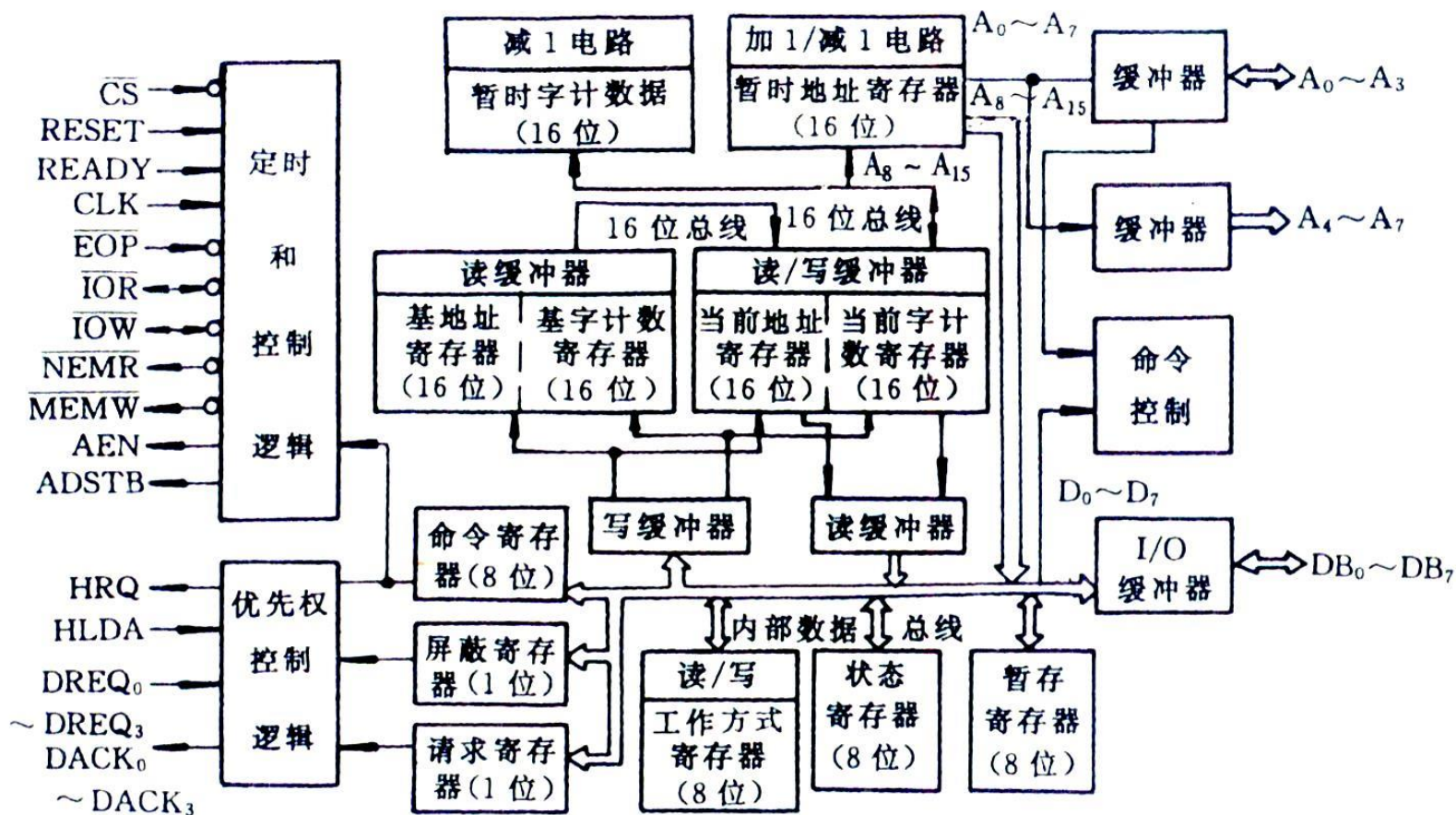
二、8237内部结构

1. 进行初始化及读状态信息涉及的主要寄存器组

- 定时与控制逻辑
- 命令控制逻辑
- 优先权控制逻辑
- 寄存器组
- 地址/数据缓冲器

2. 8237有4个独立的通道，每个通道有4个寄存器

- 基地址
- 基值字节计数器
- 当前地址
- 当前字节计数器



8237A-5 内部逻辑

4.4 DMA控制器

3. 4个通道共用寄存器

- 命令寄存器
- 状态寄存器
- 屏蔽寄存器
- 请求寄存器
- 工作方式寄存器
- 暂存寄存器

通过对这些寄存器的初始化编程，可设定8237的

- ① 3种DMA操作类型
- ② 3种DMA操作方式
- ③ 2种工作时序（主/被动）
- ④ 2种优先权排队
- ⑤ 自动预置传送地址和字节数
- ⑥ 存储器与存储器之间的数据传送

4.4 DMA控制器

三、8237的端口寻址及基本操作

- CPU由8237的 $A_{3\sim0}$ 共4位地址线输入地址信息实现对片内的16个端口寻址
- 在PC中8237占用的IO端口地址为00~0FH
p61表4.2/ p94表6.1

4.4 DMA控制器

四、内部逻辑结构

8237引脚信号的功能，它们分为以下几类

- I/O设备的请求信号DREQ_{0~3}，由I/O向DMA发出，DREQ₀优先级最高
- DMA回答I/O的信号DACK_{0~3}
- 总线请求和应答HRQ和HLDA
- #IOR/#IOW，#MEMR/#MEMW，#CS，地址线A_{0~7}
- 双向数据线DB_{0~7}，既是数据线，又是16位地址线的高8位

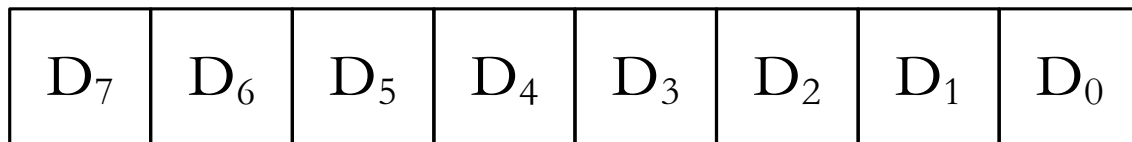
4.4 DMA控制器

五、8237的命令及对内部寄存器的编程

1. 工作方式命令字（工作方式寄存器）（DMA+11）

用于设定：DMA的操作类型/方式、地址加1/减1，自动预置以及选择通道

- 四种操作模式选择：单一传送模式、块传送模式、询问传送模式、级联模式
- 三种传送类型表示数据的传送方向，读、写和校验



模式选择

00: 询问方式
01: 单一方式
10: 块方式
11: 级联方式

1/0: 地址减1/加1

1: 自动预置

类型选择

00: DMA校验
01: DMA写
10: DMA读
11: 无效

通道选择

00: 0通道
01: 1通道
10: 2通道
11: 3通道

工作方式寄存器

4.4 DMA控制器

【注意】

- 单一方式：通道启动一次，只传送一个数据，传送完就释放系统总线交还CPU，并且：
当前地址寄存器+1（-1）（+/-由D₅决定）
当前字节计数器-1
- 块传送：启动一次可把整个数据块传送完，当前字节计数器减到0，产生EOP信号，释放总线
- 询问传送：与块传送类似，其不同点在于每传送一个字节之后，要检测（询问）DREQ是否有效，若无效，则“挂起”但不释放总线，若有效，继续传送直至字节计数器为0
- 存贮器-存贮器：为数据块传送而设置，这种传送在通道0与1之间传送（PC机未使用）
通道0（源）→通道1（目的）

4.4 DMA控制器

例如，PC系列软盘读写操作选择DMA通道2，单字节传送，地址增1，不用自动预置，其操作方式字如下：

01000110=46H 读盘 I/O → RAM (DMA写)

01001010=4AH 写盘 RAM → I/O (DMA读)

从软盘上读一个扇区的数据存放到内存区方式字为46H，写一个扇区的数据到软盘为4AH

4.4 DMA控制器

2. 基地址寄存器 (DMA+0, +2, +4, +6)
 - 16位地址寄存器 (只能写, 不能读)
 - 存放DMA传送的内存首址
 - 在初始化时, 由CPU以先低字节后高字节顺序写入
 - 在进行DMA传送时其值不变, 作用是在进行自动预置时, 将其重新装入当前地址寄存器
3. 当前地址寄存器 (DMA+0, +2, +4, +6)
 - 16位地址寄存器 (可读可写)
 - 初值与基地址寄存器相同 (两者同时由CPU写入同一端口)
 - 自动预置时, /EOP使其内容重新置为基地址

4.4 DMA控制器

4. 基字节数计数器 (DMA+1, +3, +5, +7)
 - 16位寄存器 (只能写, 不能读)
 - 存放DMA传送的总字节数
 - 在初始化时, 由CPU以先低字节后高字节顺序写入
 - 传送中, 基字节计数器不变
5. 当前字节数计数器 (DMA+1, +3, +5, +7)
 - 16位寄存器 (可读可写)
 - 当前字节数计数值, 存放DMA传送过程中没有传送完的字节数
 - 每次传送后减1, 减为0时产生/EOP
 - 初值与基字节数计数器相同 (两者同时由CPU写入同一端口)

4.4 DMA控制器

6. 屏蔽寄存器 (DMA+10, +15)

- 8位寄存器 (只能写, 不能读)
 - 禁止本通道的DMA进入, 直到执行清屏蔽寄存器命令
 - 屏蔽字的设置
- 单个通道屏蔽寄存器 (每次屏蔽单个通道)
4个通道屏蔽寄存器 (可以最多同时屏蔽4个通道)

例 开放通道2, 作为响应软的DMA请求

使用单一屏蔽

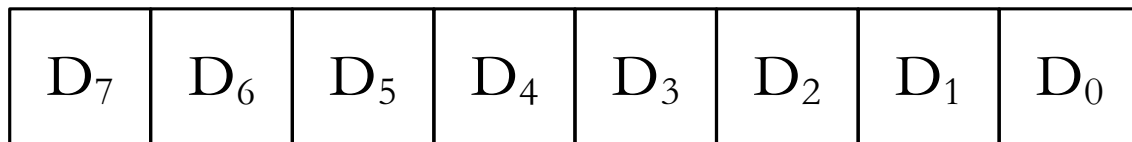
MOV AL, 0000010B ;开放通道2

OUT DMA+10, AL

使用4位屏蔽

MOV AL, 00001011B ;仅开放通道2

OUT DMA+15, AL



无效位

屏蔽位

1: 屏蔽

0: 开通

通道选择

00: 0通道

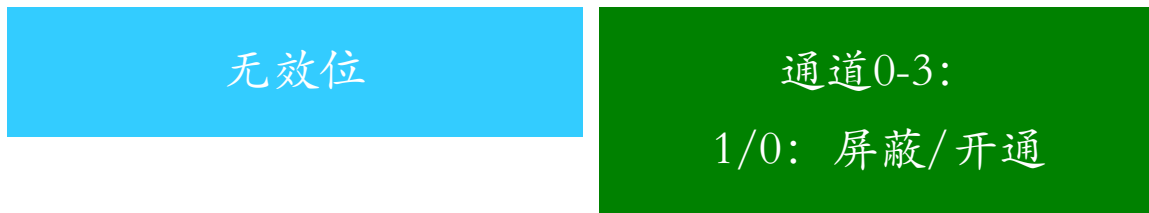
01: 1通道

10: 2通道

11: 3通道

单一屏蔽寄存器

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------



4位屏蔽寄存器

4.4 DMA控制器

7. 请求寄存器 (DMA+9)

- 8位寄存器 (只能写, 不能读)
- 用于由软件DREQ启动DMA请求 (不可屏蔽)
(DMA请求可以由I/O设备发出, 也可以由软件产生)
- 仅用于存储器到存储器的DMA传送方式
- 必须是块字节传输方式

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------



无效位

请求位

1: 有

0: 无

通道选择

00: 0通道

01: 1通道

10: 2通道

11: 3通道

请求寄存器

4.4 DMA控制器

8. 命令寄存器 (DMA+8)

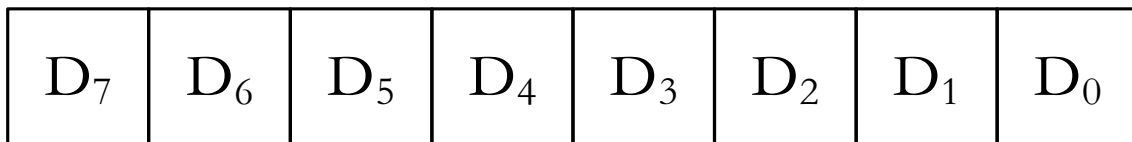
- 8位寄存器 (只能写, 不能读)
- 用于控制DMAC的操作
- 由CPU写入, RESET信号和总清除命令清除

9. 状态寄存器 (DMA+8)

- 8位寄存器 (只能读, 不能写)
- D_{0-3} 用于提供哪些通道停止计数
- D_{4-7} 用于提供哪些通道有DMA请求

10. 暂存寄存器 (DMA+13)

- 用于在存储器到存储器传送中, 暂时保存从源存储器读出的数据
- RESET命令和总清除命令可以清除暂存寄存器的内容



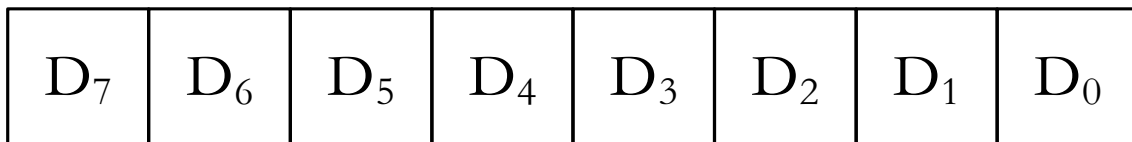
DREQ极性：0 低电平有效

DACK极性：1 低电平有效

0/1：滞后写/扩展写

0/1：固定/循环优先权

命令寄存器



0/1: 标准时序/压缩时序

0: 工作允许

1: 通道0地址不变

1: 允许存储器到存储器

命令寄存器

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------



1/0: 有/无尚未处理的DMA请求 (通道3-0)

1/0: 有/无接收到终止计数信号 (通道3-0)

状态寄存器

4.4 DMA控制器

11. 软命令

- 对特定的地址进行一次写操作（任意值），命令就生效，与写入的具体数据无关

①清先/后触发器命令（保证先写入低8位）

②总清除命令

与硬件RESET信号作用相同，使命令寄存器、状态寄存器、请求寄存器、暂存寄存器，先/后触发器清除，屏蔽寄存器置位（屏蔽所有外部DMA请求）

③清屏蔽寄存器命令

使4个通道的屏蔽位均清0，允许接受外部DMA请求

4.4 DMA控制器

六、8237的工作时序

DMAC的7种状态周期 S_I , S_0 , S_1 , S_2 , S_3 , S_4 及 S_W

- DMA的空闲周期 S_I
- DMA的过渡周期 S_0
- DMA有效周期 $S_1 \sim S_4$
- 等待状态周期 S_W

4.4 DMA控制器

1. 空闲周期 S_I

- 上电后，未编程前，还没有DMA请求，进入空闲周 S_I ，DMA处于被动工作方式，CPU可对DMA进行编程

2. 过渡状态 S_0

- 若检测到DREQ请求，DMAC即向CPU发出总线请求信号HRQ；
- DMAC从 S_I 状态跳入 S_0 状态，并重复执行，直到收到HLDA信号（CPU）进入 S_1 状态；
- DMAC从被动态进入主动态

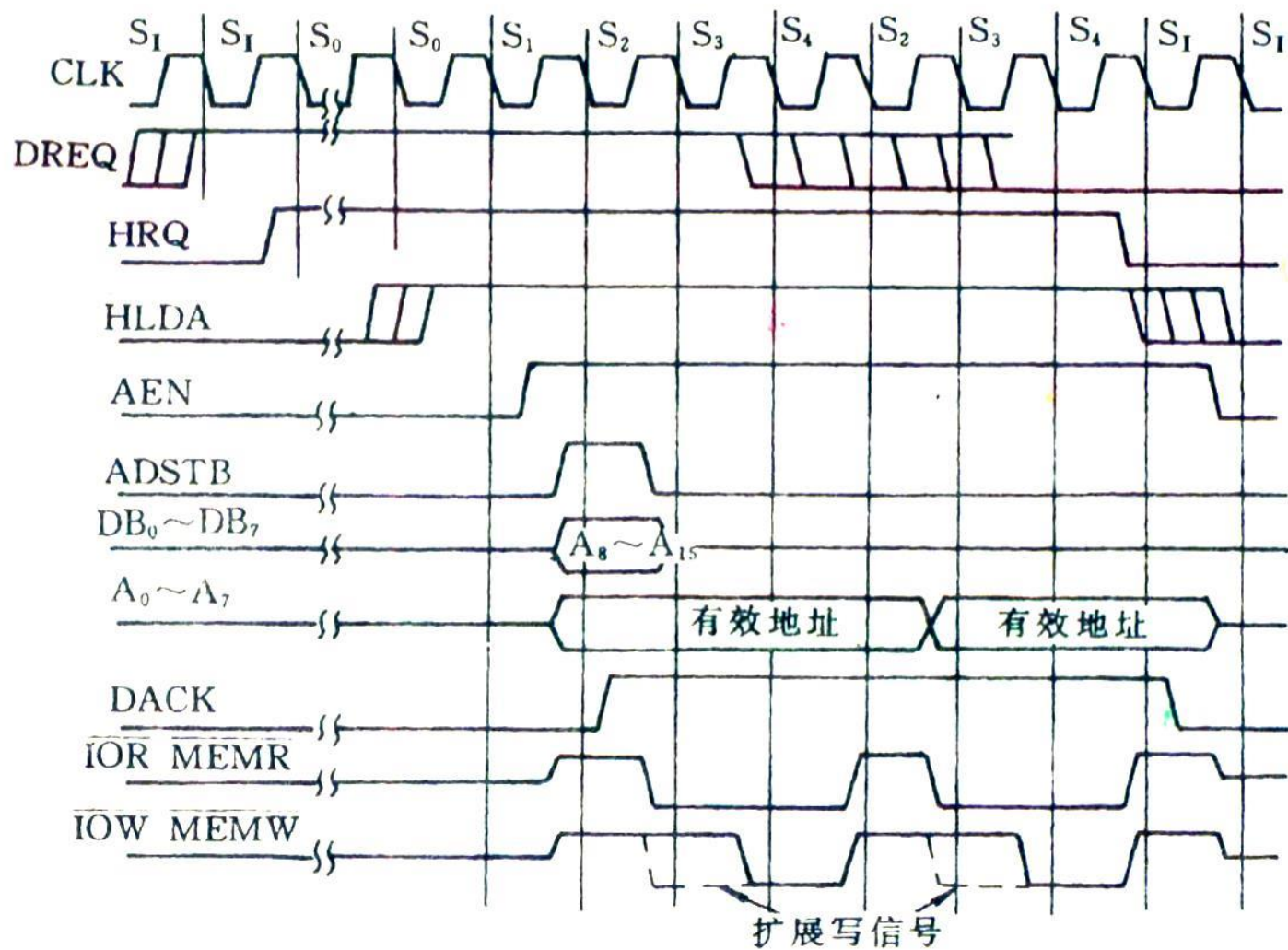
4.4 DMA控制器

3. DMA有效周期 ($S_1 \sim S_4$)

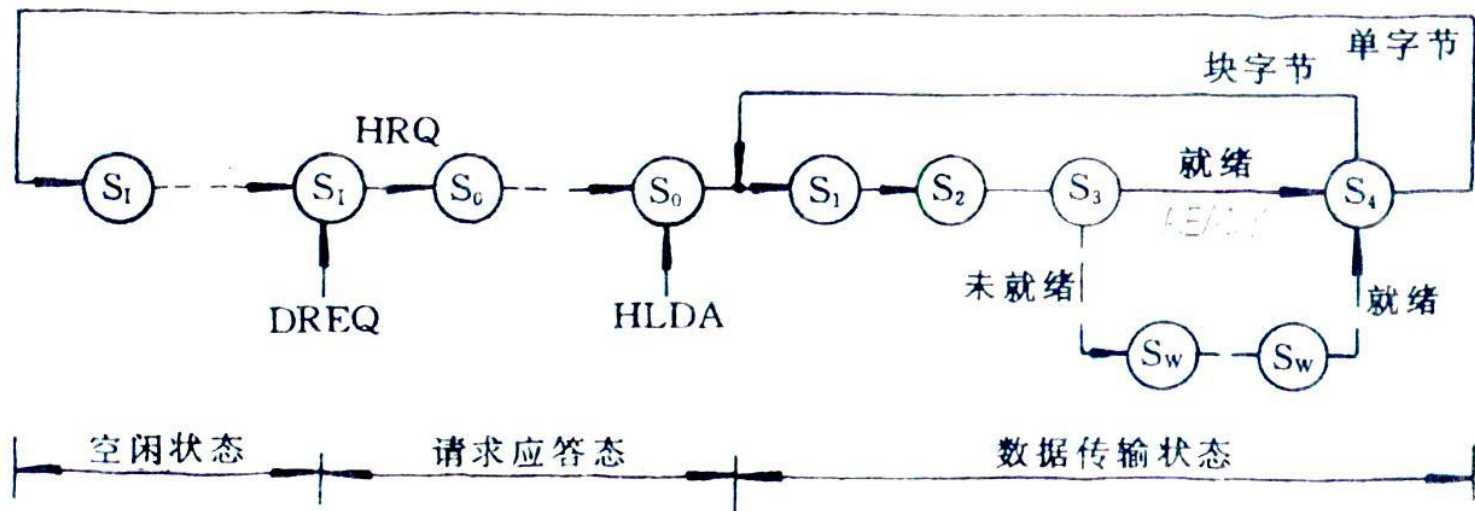
- 在CPU的回答信号HLDA到达后, DMAC进入有效周期开始传送数据;
- 一个完整的传送周期包括 S_1 , S_2 , S_3 和 S_4 四个周期;
- 如果I/O速度跟不上, 可在 S_3 和 S_4 之间插入等待周期

4.4 DMA控制器

- S_1 周期：更新高8位地址，DMAC在 S_1 状态发出地址允许AEN信号，将高8位地址 $A_8 \sim A_{15}$ 送到数据总线上 $DB_0 \sim DB_7$ 上，发地址选通信号ADSTB
- S_2 周期：
 - ① 输出16位地址到RAM（其中高8位已锁存，低8位由地址线 $A_0 \sim A_7$ 输出）
 - ② 向I/O发请求回答信号/DACK，准备传送数据
- S_3 周期：发出/MEMR或/IOR读命令，从内存或I/O准备传送数据
- S_4 周期：发/IOW或/MEMW命令（扩展写、压缩时序）



DMA 传送过程



内部状态流程

4.4 DMA控制器

例 在某一系统中，用一片8237设计DMA传输电路，8237A的基地址为00H，要求使用通道0，从外设输入一个1KB的数据块，传送到内存6000H开始区域，每传送一个字节，地址增1，采用数据块连续传送方式，禁止自动预值，外设的DMA请求信号DREQ和响应信号DACK均为高电平有效，设计**初始化**程序。

4.4 DMA控制器

DMA EQU 00H

OUT DMA+0DH, AL ;写总清命令

MOV AX, 6000H ;

OUT DMA+00H, AL ;写通道0基地址与当前地址
;寄存器低8位

MOV AL, AH

OUT DMA+00H, AL ;写通道0基地址与当前地址
;寄存器高8位

4.4 DMA控制器

MOV AX, 0400H	;1KB数据
DEC AX	
OUT DAM+01H, AL	;写通道0基字节计数与当前 ;字节计数寄存器低8位
MOV AL, AH	
OUT DMA+01H, AL	;写通道0基字节计数与当前 ;字节计数寄存器高8位
;块方式, 加1, 自动预置, DMA写, 0通道	
MOV AL, 84H	;10000100
OUT DMA+0BH, AL	;写工作方式寄存器

4.4 DMA控制器

MOV AL, 00H ;00000000 (0通道开通)
OUT DMA+0AH, AL ;写单个通道屏蔽寄存器

;DACK高电平有效, DREQ高电平有效, 滞后写
;固定优先权, 正常(标准)时序, 允许8237A-5工作
;通道0地址不保持, 禁止存储器到存储器传送

MOV AL, 80H ;10000000
OUT DMA+08H, AL ;写命令寄存器

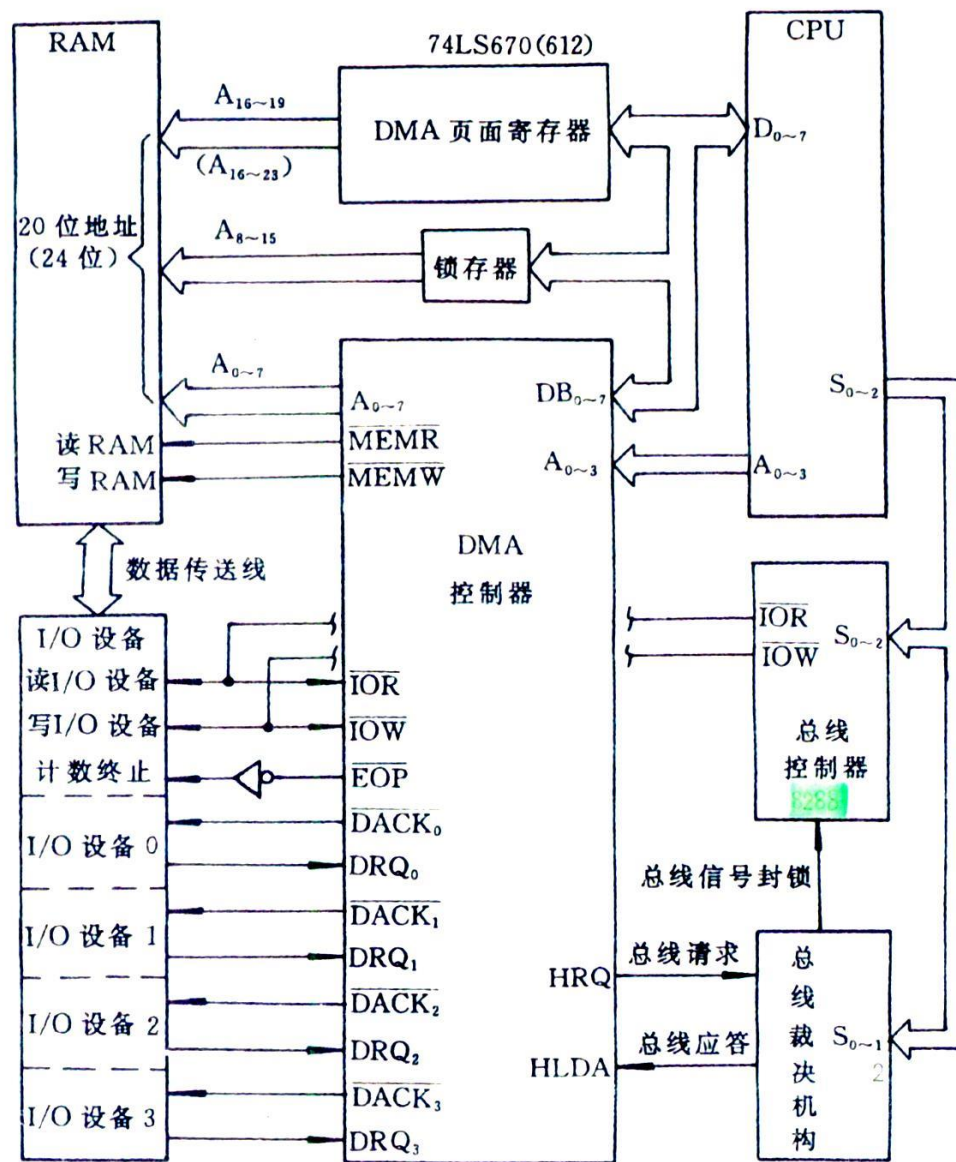
;有请求, 通道0

MOV AL, 04H ;00000100
OUT DMA+09H, AL ;写请求寄存器

4.5 DMA 系统

一、DMA 系统组成

- DMA 控制器和配套芯片组
- PC 微机中，采用 8237、DMA 页面地址寄存器、总线仲裁逻辑，支持 4 个通道（单片）或 7 个通道（两片）DMA 传输



PC系列DMA系统

4.5 DMA 系统

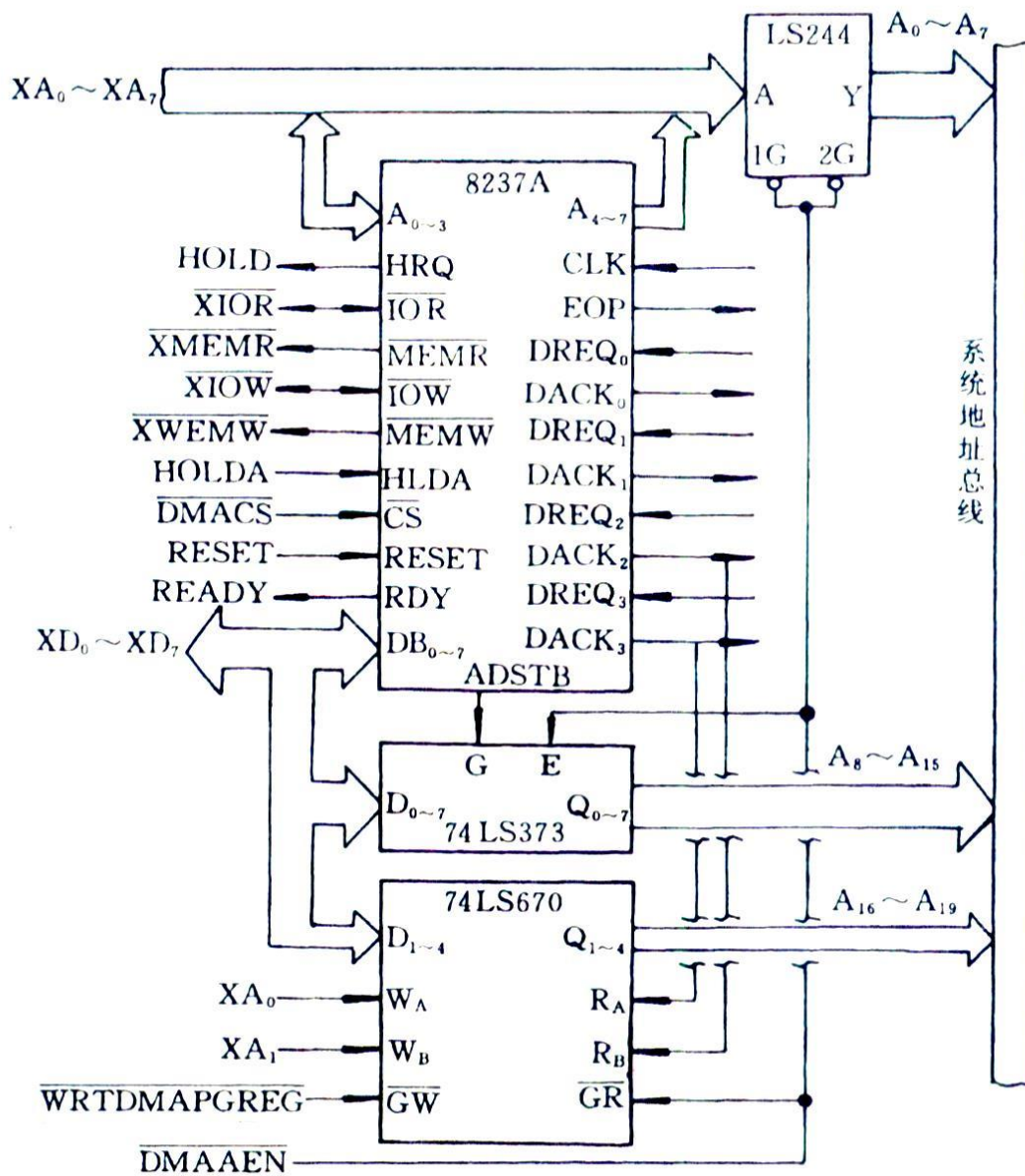
二、DMAC有效地址的生成

当DMAC取得总线控制权后，作为系统为主控制器，向**存储器**和**I/O设备**发地址信号

4.5 DMA 系统

1. 如何提供存储器的地址

- DMAC只能提供16位地址
 - $A_0 \sim A_7$ (低8位) $DB_0 \sim DB_7$ (高8位)
- PC/XT地址总线20位
 - DMAC提供16位 $A_0 \sim A_{15}$
 - 页面地址寄存器4位, $A_{16} \sim A_{19}$
- PC/AT地总24位
 - DMAC提供16位
 - 页面地址8位, $A_{16} \sim A_{23}$



有效地址生成

4.5 DMA 系统

- 地址总线20位
 - $A_0 \sim A_7$ 由DMAC地址直接产生
 - $A_8 \sim A_{15}$ 由DMAC地址直接产生 (DMA地址锁存器 $DB_{0 \sim 7}$)
 - $A_{16} \sim A_{19}$ 由页面地址产生
- AEN信号全部选通20位内存地址 (系统地址总线)

4.5 DMA 系统

- 页面地址的编码

- 写操作

地址	/GW	W_B	W_A	
80H	0	0	0	写入0号寄存器
81H	0	0	1	写入1号寄存器
82H	0	1	0	写入2号寄存器
83H	0	1	1	写入3号寄存器
	1	×	×	寄存器内容不变

【注】 /GW连接/WRTDMAPGREG

4.5 DMA 系统

— 读操作

地址	/GR	R_B	R_A	
80H	0	0	0	读出0号寄存器
81H	0	0	1	读出1号寄存器
82H	0	1	0	读出2号寄存器
83H	0	1	1	读出3号寄存器

4.5 DMA 系统

2. 如何提供I/O设备的地址?

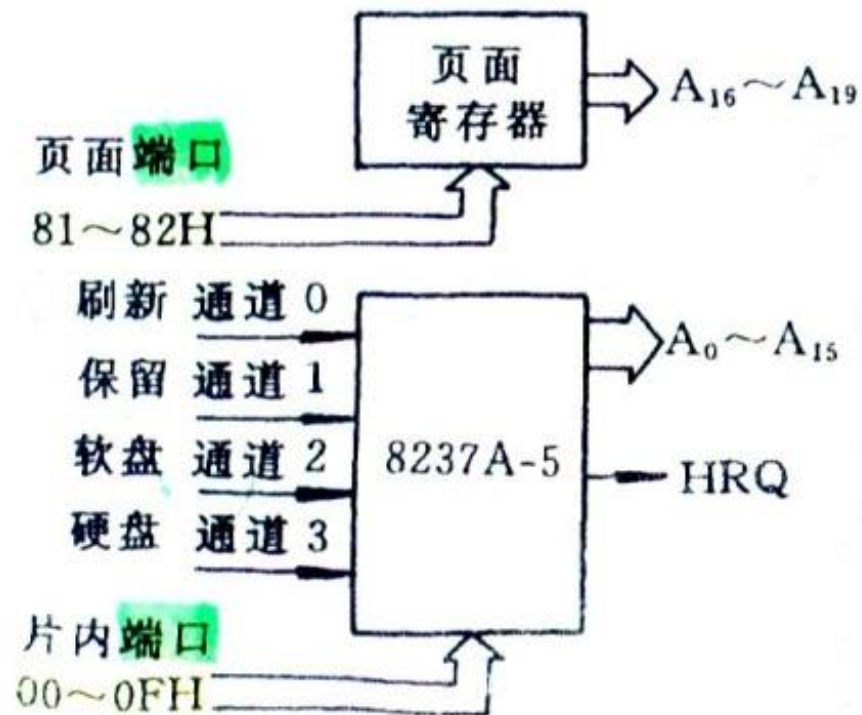
- DMA提供**DAK#**信号来取代I/O设备地址选择逻辑，申请DMA传送并被响应的设备在DMA传送过程中保持为有效设备
- 在对I/O进行读写数据时，只要DAK#信号和RD#或WR#信号同时有效，就能完成对I/O设备端口的读/写操作，与I/O设备的端口地址无关

4.5 DMA 系统

三、IBM-PC的DMA系统

1. PC/XT的DMA系统

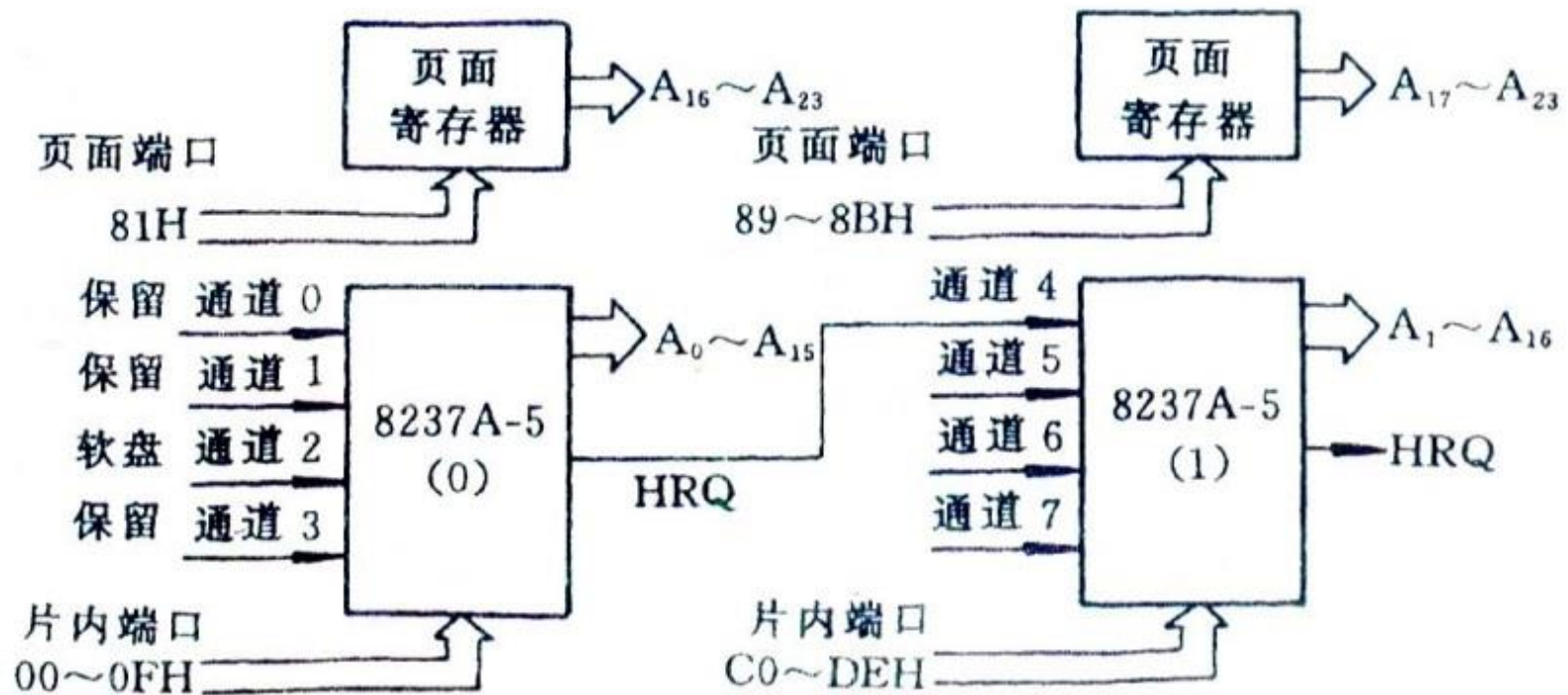
- ① 支持4个DMA通道传送
- ② 0通道用于RAM刷新
- ③ 通道1保留，用于同步通信，如网卡
- ④ 通道2用于软盘，通道3用于硬盘



4.5 DMA 系统

2. PC/AT的DMA系统

- 由2片8237DMAC构成
- 支持7个DMA通道传送
- 通道0#保留，因AT有专门的RAM刷新电路
- 通道3#保留，因硬盘驱动器（HDD）采用高速PIO传送，无须DMA支持
- 通道5#~7#均保留，8位数据传送，每次64KB，可寻址空间16MB



PC/AT DMA系统

4.5 DMA 系统

四、DMAC的初始化编程

1. 初始化编程

- 先通过命令寄存器将要编程的通道屏蔽，编程完成后再允许芯片工作；
- 所有通道（包括不使用）的方式字寄存器都要加载；
- 向基地/当前址寄存器写低16位地址，向页面寄存器写高4/8位地址；
- 上电期间，对DMA芯片进行检测

4.5 DMA 系统

例 对PC机的DMAC进行检测

分别对4个通道的8个寄存器先写入全“1”，全“0”，再读出比较看是否一致，不一致则出错，待机

4.6 DMA传送的应用

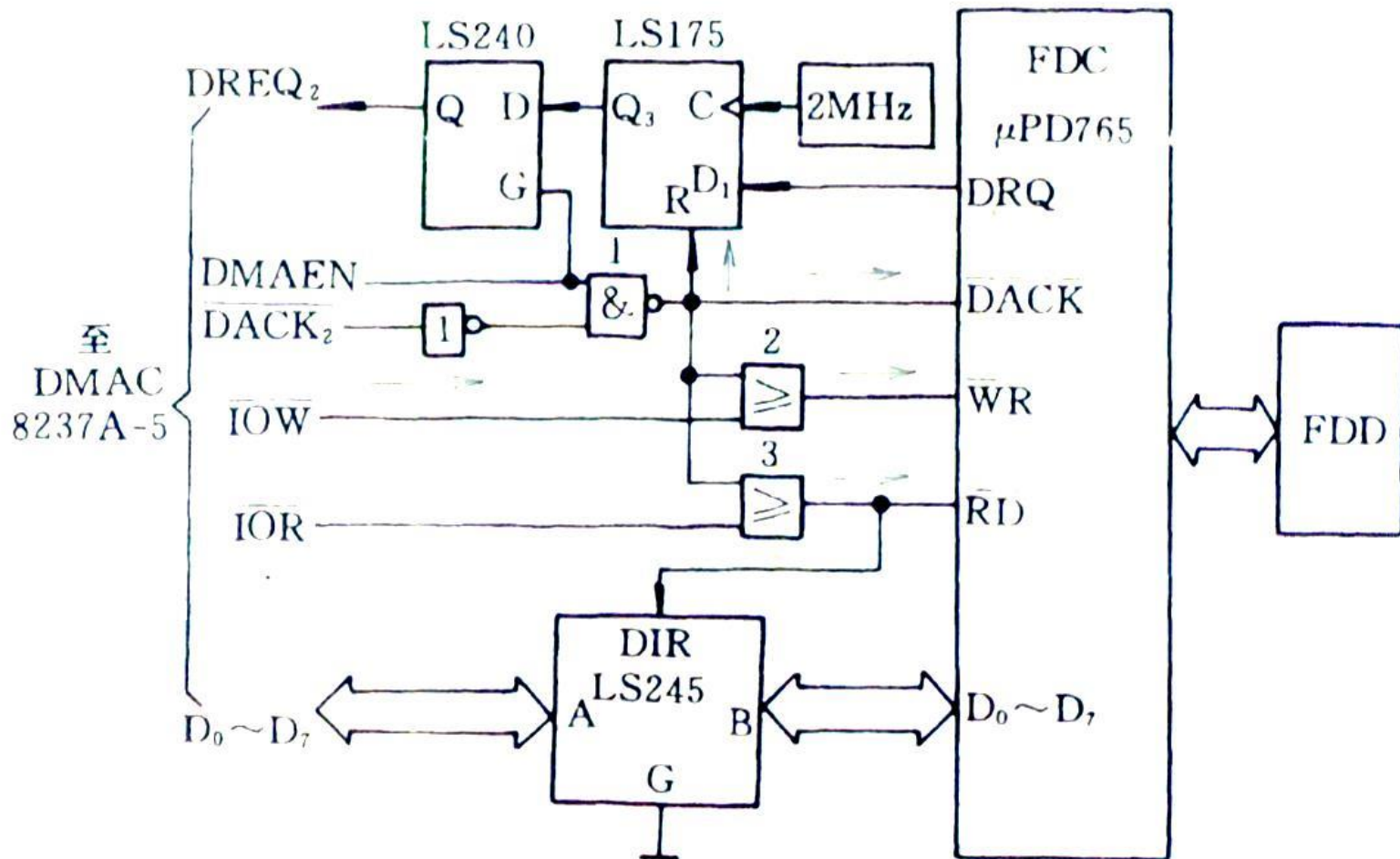
软盘控制器

- 当DMA申请允许信号DMAEN为有效，FDC软盘控制器可向DMA发送DREQ₂请求
- DMA响应请求之后，发回答信号DACK₂，允许对FDC进行IOW/IOR操作
- DMA的D₀~D₇通过LS245直接与FDC的D₀~D₇相连

4.6 DMA传送的应用

例 DMA-SETUP程序

这个程序软盘的读、写、校验等操作调用，调用前，工作方式的命令码入AL



DMA传送过程