



## 3.14 输入输出指令

- **8086/8088CPU**采用两个地址空间，即存储器地址空间和I/O端口地址空间，这就是说，访问存储器与访问I/O端口的指令是不同的，因此，我们不能采用**MOV**指令来访问I/O端口。
- **8086/8088CPU**提供了访问I/O端口的专门指令，由于系统至多可以设计**65536**个端口寄存器，因此端口地址只需要用**16**位地址表示，这一点与存储器不同。



## 3.14 输入输出指令

### 3.14.1 端口输入指令 (IN)

格式: **IN DST, SRC**

- 表示**CPU**从端口读取数据
- **DST**用于指定存储读取数据的寄存器，这里只能取**AL**或**AX**寄存器：
  - 当从**8**位端口中读取一个字节时，采用**AL**寄存器；
  - 当从**16**位端口中读取一个字时，采用**AX**寄存器。



## 3.14 输入输出指令

- **SRC**用于指定端口地址，它有两种寻址方式：
  - 当端口地址可以用**8**位地址值表示时，**SRC**可以直接用地址值（**port**）表示，这时的寻址方式为直接寻址；
  - 当端口地址不能用**8**位表示时，**SRC**取**DX**，即将端口地址存放在**DX**中，通过**DX**找到端口地址，这时的寻址方式为寄存器间接寻址；
  - 对可以用**8**位地址表示的端口访问也可以采用寄存器间接寻址方式。



## 3.14 输入输出指令

### 3.14.2 端口输出指令（OUT）

格式： **OUT DST, SRC**

表示**CPU**将数据送给**I/O**端口。

其说明类似与**IN**指令，只是**SRC**用于指定存储读取数据的寄存器（**AL**或**AX**），**DST**用于指定端口地址。



## 3.14 输入输出指令

操作类型	端口输入指令	端口输出指令
字节	<b>IN AL, PORT</b>	<b>OUT PORT, AL</b>
字	<b>IN AX, PORT</b>	<b>OUT PORT, AX</b>
字节	<b>IN AL, DX</b>	<b>OUT DX, AL</b>
字	<b>IN AX, DX</b>	<b>OUT DX, AX</b>



## 3.14 输入输出指令

例: (1) IN AL, 28H

若 (28H端口) = 1010 1111B

执行后 (AL) = (28H端口) = 0AFH

(2) IN AX, 28H

若 (28H端口) = 1010 1111B

(29H端口) = 0101 0000B

执行后 (AL) = (28H端口) = 0AFH

(AH) = (29H端口) = 50H

(3) MOV DX, 300H

IN AL, DX

若 (300H端口) = 69H

执行后 (AL) = (300H端口) = 69H



## 3.14 输入输出指令

例: (1) **OUT 21H, AL**

若(AL)= 1100 1100B

执行后 **(21H端口) = 0CCH**

(2) **MOV DX, 21BH**

**OUT DX, AL**

若 (AL) = 1010 0110B

执行后 **(21BH端口) = (AL) = 0A6H**



## 3.14 输入输出指令

输入/输出指令（IN、OUT）特点：

(1) 累加器AL、AX的专用指令

对输入指令IN，目的操作数只能为AL，或AX

IN AL, 21H ✓

IN AX, DX ✓

IN BX, 21H ✗

对输出指令OUT，源操作数只能为AL，或AX

OUT 20H, AL ✓

OUT [DX], AX ✗

OUT DX, CX ✗





## 3.14 输入输出指令

(2) 端口地址可为直接寻址方式或间接寻址方式

①当端口号在**0~FFH**，即**0~255**时，

可在指令中直接指定端口号

例： IN       AL, **21H**

      OUT   **20H**, AL

②当端口号**>FFH**，即**>255**时，需把端口号放在

**DX**寄存器中，**DX**是一个**16**位寄存器，范围在**0~FFFFH**之间。