第 3 章　计算机的硬件

计算机是由电子元件来控制电流的电子设备。第 2 章介绍了计算机是采用二进制表示信息的，本章将探讨计算机如何使用电信号来表示和操作这些二进制数，分析计算机中最基础的硬件元件——门。门对电信号执行基本运算。一个门接收一个或多个输入信号，生成一个输出信号。门的类型很多，每种类型的门执行一个特定的逻辑函数。

电路是由门组合而成的，可以执行更加复杂的任务。在电路中，一个门的输出值通常作为另一个门或多个门的输入值。电路中的电流由经过精心设计的相互关联的门来控制。简单电路可以逐步集成，构成大规模和超大规模集成电路。

门和电路有 3 种等价的描述方法：布尔表达式、逻辑框图和真值表。虽然它们互不相同，但效果却是一样的。

电路可以进一步构成计算机的部件，这些部件常常被看作计算机最基本的组成部分，可以组合成各种各样的计算机。现代计算机的基本工作原理主要源于冯·诺依曼思想，由五大部件构成，五大部件通过总线和接口连接起来，总线和接口分布在计算机的主板上。计算机的运行过程就是连续自动地执行一条条指令的过程，但现在也出现了超越该思想的新的计算机体系结构。

学习目标

- 掌握布尔代数的基本运算。
- 了解布尔代数和门的关系。
- 掌握用基本门组合成电路。
- 掌握用布尔表达式、真值表和逻辑框图描述门或电路的行为。
- 了解门的构造。
- 掌握常见的组合电路和时序电路的工作原理。
- 了解集成电路的特征。
- 掌握冯·诺依曼结构计算机的五大部件及其功能。
- 熟悉个人计算机的主板、接口和插槽。
- 掌握指令的含义、格式和寻址方式。
- 了解指令的执行过程。
- 了解非冯·诺依曼结构的计算机。

3.1　布尔代数

乔治·布尔长期关注逻辑过程形式化与机械化，出版过著作 *The Laws of Thought*，书中介绍了以他的名字命名的布尔代数。他认为，可将代数形式表示的逻辑关系应用于实际。布尔代数是数字系统逻辑设计的基础。

3.1.1 布尔代数的基本运算

布尔代数中只有0和1两个常数，不能作为算术中的数来比较它们之间的大小关系，所以称为逻辑0和逻辑1，统称逻辑值。逻辑1和0可以用来表示两种截然不同的状态，如日常生活中灯的亮与灭、开关的开与关、磁性材料的S极与N极等。布尔代数中有3种基本运算。

1. 逻辑与

在布尔代数中，逻辑与运算用点(·)来表示，或者省略，写做 $X=A\cdot B$ 或者 $X=AB$。逻辑与的运算规则如表3.1所示，只有 A、B 都为1时，$A\cdot B$ 的结果才为1；当 A、B 中有一个不为1时，结果就为0。

因为有两个输入值，每个输入有两种可能的值，所以与运算有4种0和1的组合。因此，在布尔表达式中，与运算有4种情况：$0\cdot 0=0$；$0\cdot 1=0$；$1\cdot 0=0$；$1\cdot 1=1$。

逻辑与运算的意义类似于串联开关电路，如图3.1所示，设开关接通用1表示，断路用0表示，灯亮的条件是开关 A 和 B 同时连通；否则灯不会亮。灯亮用1表示，灯灭用0表示。开关状态和灯的亮灭就表示了输入与输出的关系。与电路接收两个输入信号，这两个输入信号的值决定了输出信号，如果两个输入值都为1，输出为1；否则输出为0。

表3.1 逻辑与的运算规则

A	B	$X=A\cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

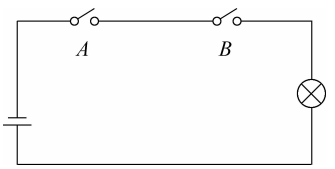


图3.1 与电路示意图

2. 逻辑或

在布尔代数中，逻辑或运算用加(+)来表示，$X=A+B$。逻辑或的运算规则如表3.2所示，只有 A、B 都为0时，$A+B$ 的结果才为0；当 A、B 中有一个为1时，结果都为1。或运算也有4种情况：$0+0=0$；$0+1=1$；$1+0=1$；$1+1=1$。

逻辑或运算的意义类似于并联开关电路，如图3.2所示，灯亮的条件是开关 A 和 B 只要有一个连通，只有当开关 A 和 B 都不连通时，灯才不会亮。

表3.2 逻辑或的运算规则

A	B	$X=A+B$
0	0	0
0	1	1
1	0	1
1	1	1

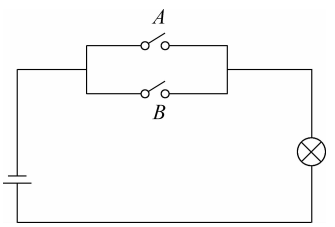


图3.2 或电路示意图

3. 逻辑非

逻辑非就是取反，用上面加一小横线来表示，$X=\overline{A}$，逻辑非的运算规则如表3.3所示。如果输入是0，输出是1；如果输入是1，则输出是0，即$\overline{0}=1$，$\overline{1}=0$。

表3.3 逻辑"非"的运算规则

A	$X=\overline{A}$
0	1
1	0

3.1.2 布尔函数与表达式

在数学中，布尔函数用于描述用输入布尔量来确定输出布尔量的逻辑关系，有时也称为逻辑函数，经常用在逻辑电路中。布尔函数通常是形如$F=\varphi(b_1,b_2,\cdots,b_n)$的函数。输入变量$b_1,b_2,\cdots,b_n$取任意一组确定的值后，输出变量$F$的值也就唯一确定了。布尔函数的右部称为布尔表达式，是指由布尔量和布尔运算符按一定的语法规则组成的表达式。组成布尔表达式的时候，以上3种运算的优先级不同，逻辑非最高，逻辑与次之，逻辑或最低。也就是说，在一个同时含有上述3种运算的表达式中，应先进行逻辑非运算，再进行逻辑与运算，最后进行逻辑或运算。可以通过括号来改变运算次序。

布尔变量b_i的取值只能是0或1，F的取值也只能是0或1，所以对有n个布尔变量的布尔函数而言，变量的所有取值有且仅有2^n种情况。对一组变量的各种可能取值，唯一地求出对应函数的值，并列成一个表，该表就称为函数的真值表。真值表可全面反映出一组输入变量与输出变量之间的关系。

如布尔函数$F=A(B+C)$，输出F和输入变量A、B、C之间的真值表如表3.4所示，具有3个输入的布尔函数有$2^3=8$种输入组合，所以真值表一共有8行。

要证明两个布尔函数是否相等，最原始的办法就是列出它们的真值表，看看它们是否相同。如两个布尔函数$F=A(B+C)$和$G=AB+AC$，列出它们的真值表，如表3.4和表3.5所示，针对输入变量的每一组取值，输出F和G都是相等的，证明$F=G$成立。此时，称这两个布尔函数是等价的。布尔函数的等价说明每个输入值的组合都生成相同的输出。

表3.4 函数$F=A(B+C)$的真值表

A	B	C	$F=A(B+C)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

表3.5 函数$G=AB+AC$的真值表

A	B	C	$G=AB+AC$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

3.1.3 布尔表达式的化简

从布尔表达式的等价可以看出，相同的一组变量，存在多个函数值相同的布尔函数。怎样寻找一个相同而又最简单或是最佳的布尔表达式来表示该函数呢？如何将一个烦琐的布尔表达式化简成一个等价而又最简单的布尔表达式呢？除了最基本的真值表外，在布尔代数中，还有更好的方法和简化工具，从而为数字电路的设计和实现奠定了理论基础。

其实，上面这个例子证明了布尔代数的一个重要定律——分配律。

$$A(B+C)=AB+AC$$

表3.6列出了布尔代数的一些常用的定律，这些都是可证明的数学法则，可以用真值表来验证，以后可以利用这些定律来简化和设计布尔表达式。

表3.6 布尔代数的几个定律

属　性	与	或
交换律	$AB=BA$	$A+B=B+A$
结合律	$(AB)C=A(BC)$	$(A+B)+C=A+(B+C)$
分配律	$A(B+C)=AB+AC$	$A+BC=(A+B)(A+C)$
恒等	$A1=A$	$A+0=A$
余式	$A\cdot\overline{A}=0$	$A+\overline{A}=1$
德·摩根定律	$\overline{A\cdot B}=\overline{A}+\overline{B}$	$\overline{A+B}=\overline{A}\cdot\overline{B}$

3.2 门

数字逻辑电路中最基本的电子元件是门，也称逻辑门。通过控制高、低电平来实现逻辑运算。逻辑门可以组合使用，实现更为复杂的逻辑运算。简单的逻辑门由晶体管组成。

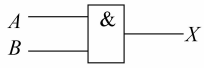
3.2.1 基本门

最基本的门有3种，用于实现布尔代数中的3种基本运算，实现与运算的门称为与门，实现或运算的门称为或门，实现非运算的门称为非门。此外，常用的门还有异或门、与非门和或非门。这些基本的门是构造计算机或者数字电路的基本元器件。

1. 与门

与门实现布尔代数中的与运算，接收两个输入值，生成一个输出值。图3.3是与门的3种表示方法。在逻辑框图中，输入和输出由流入和流出门的连接线表示。

$X=A\cdot B$



A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

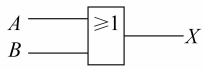
(a) 布尔表达式　(b) 逻辑框图符号　(c) 真值表

图 3.3　与门的各种表示方法

2. 或门

或门实现布尔代数中的或运算，接收两个输入值，生成一个输出值。图 3.4 是或门的 3 种表示方法。

$X=A+B$



A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

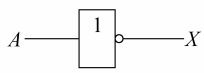
(a) 布尔表达式　(b) 逻辑框图符号　(c) 真值表

图 3.4　或门的各种表示方法

3. 非门

非门实现布尔代数中的非运算。接收一个输入值，生成一个输出值。非门有时又叫做逆变器，因为它对输入值求逆。图 3.5 是非门的 3 种表示方法。

$X=\overline{A}$



A	X
0	1
1	0

(a) 布尔表达式　(b) 逻辑框图符号　(c) 真值表

图 3.5　非门的各种表示方法

4. 异或门

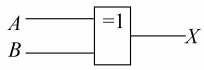
异或门实现异或运算，异或运算的运算规则是：当两个输入相同时，输出为 0；否则，输出为 1。用布尔代数符号(⊕)表示异或运算。图 3.6 是异或门的 3 种表示方法。

注意异或门和或门之间的区别。当两个输入都是 1 时，或门生成 1 而异或门生成 0。异或门的逻辑框图符号和或门也相似，只是多了一条贯穿两个输入信号的连接线的曲线。

5. 与非门

与非门是由与门和非门组合得到的门，接收两个输入信号。也就是说，与非门是对与

$X=A\oplus B$



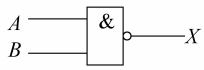
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

(a) 布尔表达式　　(b) 逻辑框图符号　　(c) 真值表

图 3.6　异或门的各种表示方法

运算求逆。同样地，如果让与门的结果经过一个逆变器(非门)，得到的输出和与非门的输出一样。图 3.7 是与非门的表示法。

$X=\overline{A\cdot B}$



A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

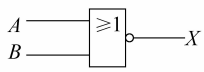
(a) 布尔表达式　　(b) 逻辑框图符号　　(c) 真值表

图 3.7　与非门的各种表示方法

6. 或非门

或非门是由或门和非门组合得到的门，接收两个输入信号。也就是说，或非门是对或运算求逆。图 3.8 是或非门的各种表示法。

$X=\overline{A+B}$



A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

(a) 布尔表达式　　(b) 逻辑框图符号　　(c) 真值表

图 3.8　或非门的各种表示方法

“与非”门和“或非”门的逻辑框图符号与“与”门和“或”门的相似，只是多了一个求逆的小圆圈(说明求逆运算)。比较与门和与非门的真值表，输出的每一行都是相反的。或门和或非门的真值表也是如此。

3.2.2　具有更多输入的门

前面介绍了 6 种类型的门，下面总结各种门的特性。

(1) 非门将对它的唯一输入值求逆。

(2) 如果两个输入都是 1，与门输出 1。

(3) 如果两个输入中任何一个为1,或门输出1。

(4) 如果两个输入不同,异或门输出1。

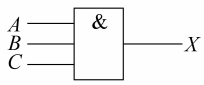
(5) 与非门的输出和与门的输出相反。

(6) 或非门的输出和或门的输出相反。

这些规则也可以用在有多个输入的门中。因此门也可以设计为接受3个或更多个输入。如具有3个输入值的与门,只有当3个输入都是1时,才得到值为1的输出;具有3个输入的或门,如果任何一个输入为1,则输出都是1。这些定义和具有两个输入值的门的定义一致。图3.9展示了具有3个输入信号的与门。

$X=A\cdot B\cdot C$

(a) 布尔表达式



(b) 逻辑框图符号

A	B	C	$X=A\cdot B\cdot C$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

(c) 真值表

图3.9 3个输入的与门的各种表示方法

对于逻辑框图符号,只需要在两个输入的与门符号上加入第三个输入信号即可。但对应布尔表达式,则需要重复一次与操作,以表示第三个值。

3.2.3 门的构造

门是由电子元器件及其电路连接来实现的。构造门就是用二极管或三极管的组合来建立输入值和输出值之间的映射。

1. 二极管、三极管的工作原理

在计算机中,可以用高、低电平来表示1和0,如图3.10(a)所示。

二极管是一种可表示0和1的电子元器件,在半导体硅或锗中,一部分区域掺入微量的3价元素硼成为P型半导体,另一部分区域掺入微量的5价元素磷成为N型半导体。在P型和N型半导体的交界处就形成一个PN结。一个PN结就是一个二极管,P区的引线称为阳极,N区的引线称为阴极,如图3.10(b)所示。

二极管的导通和断开即可表示1和0,如图3.10(c)所示。

三极管是在二极管的基础上做成的,通常也是由半导体材料硅构成的。它有两个作用:一是用作传导电流的电线(称为导通);二是用作阻止电流的电阻器(称为截止)。

三极管在二极管上增加了一个栅极(基极)b,输入信号的电压电平决定了三极管的

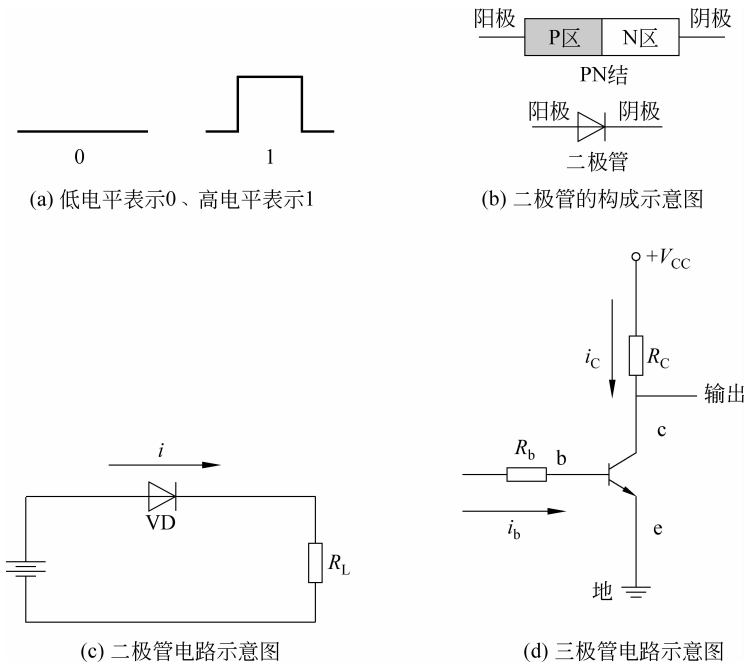


图 3.10　数字电平、二极管和三极管电路示意图

作用，可以作为开关使用。三极管有 3 个接线端，即发射极(emitter，e)、基极(base，b)和集电极(collector，c)，如图 3.10(d)所示。在计算机中，发射极 e 通常接地，c 接高电压，约为 5V。当 b 施加高电平时，c 极和 e 极连通，当 b 施加低电平时，c 极和 e 极断开，这样就可以由 b 来控制三极管的输出信号是高电平(1)还是低电平(0)。

2. 用二极管和三极管构造门

根据二极管和三极管的工作原理，用它很容易构成与门、或门和非门，如图 3.11 所示。用二极管构造的与门，当 A、B 端均为高电平(1)时，F 端为高电平(1)，否则 F 为低电平(0)。用二极管构造的或门，当 A、B 端均为低电平(0)时，F 端为低电平(0)，否则 F 端为高电平(1)。用三极管构造的非门，当 A 端为低电平(0)时，F 端为高电平(1)，否则 F 端为低电平(0)。

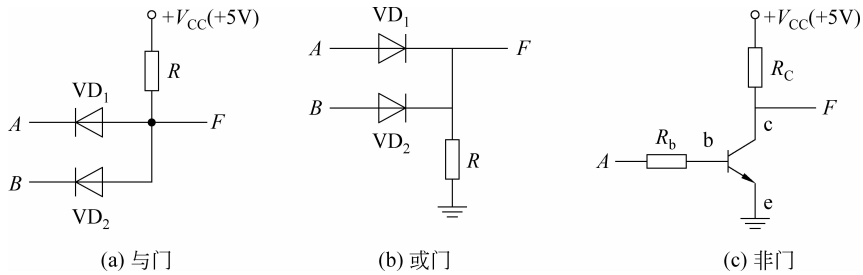


图 3.11　用二极管和三极管构成的门

有了与门、或门和非门以后，就可以利用这些门构造复杂的门。如构造与非门，就是用一个与门和一个非门形成的门，也可以用两个三极管构成，一个与或非门是由一个与门、一个或门和一个非门构成。

3.3 电路

电路由门组合而成。电路有两类：一类是组合电路，输入值明确决定输出，如加法器和选择器；另一类是时序电路，它的输出由输入值和电路现有状态共同决定。因此，时序电路通常涉及信息存储，如锁存器和计数器。和门一样，可以用3种方法描述电路的动作，即真值表、布尔表达式和逻辑框图。它们是不同的表示法，但同样有效。电路比门的真值表更复杂，需要用足够的行和列来说明对任何一组输入值，整个电路如何动作。布尔表达式是演示电路活动的最好方式，能够直接用数学符号定义和操作电路逻辑。在逻辑框图中，通过用不同方式把这些门连接在一起，就可以真实地表示出整个电路逻辑。

3.3.1 用门构成电路

把一个门的输出作为另一个门的输入，就可以把门组合成电路，如图3.12所示。

这个逻辑框图的含义是什么呢？首先从输出来看，最终输出 X 是 A 和 D 进行与运算，A 和 D 同时为1时，结果为1；若希望 D 为1，B 或者 C 有一个为1即可。或门的两个输入是 B 和 C，其输出为 $D=B+C$，作为与门的一个输入，与门的另一个输入是 A，与门的输出是 $X=A(B+C)$。因此，图3.12所示电路所表示的布尔表达式就是 $A(B+C)$。

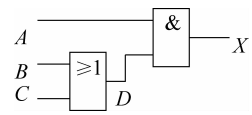


图3.12 $A(B+C)$的逻辑框图

也可以用真值表来表示该电路的动作，如表3.7所列。因为整个电路有3个输入，所以需要8行来描述所有可能的输入组合。D 列显示了电路的中间值。

表3.7 布尔表达式 $A(B+C)$ 的真值表

A	B	C	$D=B+C$	$X=A(B+C)$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

电路是一组互连的门，所以表示电路的布尔表达式是布尔运算的组合。只需要将这些布尔运算用门组织成正确的形式，就可以得到一个有效的逻辑电路。例如，设计一个逻

辑电路来实现布尔表达式 $f=x+\bar{y}\cdot z$。其中有 3 个布尔变量 x、y、z，布尔表达式表示成门电路的组合，如图 3.13 所示。

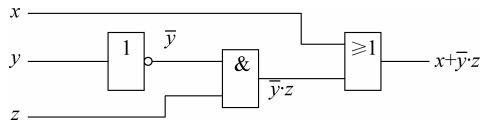


图 3.13　$f=x+\bar{y}\cdot z$ 的逻辑图

3.3.2　加法器

计算机能执行的最基本运算是把两个数相加，用二进制执行。加法运算是由专用电路——加法器来执行的。完成两个 1 位二进制数相加的加法器，称为 1 位加法器，完成两个 8 位二进制数相加的加法器，称为 8 位加法器。还可以构成 16 位、32 位加法器。

1. 1 位加法器

两个二进制数字求和可能会生成进位值。例如，在二进制中，1+1=10。计算两个二进制数字的和，不考虑低位的进位，称为半加，实现半加功能的电路称为半加器。

半加器有两个输入，生成两个输出。两个输入是需要相加的数 A 和 B；两个输出，一个表示本位的求和结果 S，另一个表示向上一位的进位 C。如果 A 和 B 都为 0，那么和为 0，进位为 0；如果 A 为 0，B 为 1，则和为 1，进位为 0；如果 A 为 1，B 为 0，则和为 1，进位为 0；如果 A 和 B 都为 1，那么和为 0，进位为 1。相应的真值表如表 3.8 所示。

表 3.8　半加器的真值表

A	B	和 S	进位 C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

如果把真值表中的和与进位列同各种逻辑运算进行比较可以知道，和操作实际上就是对输入的两个二进制数进行异或(XOR)运算，而进位等效于对输入的两个二进制数进行与(AND)运算，即

$$和\ S=A\oplus B$$

$$进位\ C=AB$$

将一个异或门和一个与门组合在一起，就得到了图 3.14 所示的半加器逻辑框图。

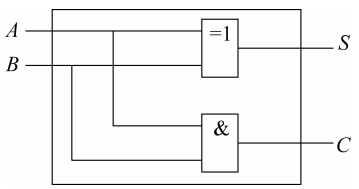


图 3.14　半加器的逻辑框图

半加器只能计算两个二进制数位的和，没有考虑低位的进位，不能实现多位数相加。实际上，在计算多位数的和时，每一位相加时都要考虑前一位是否有进位。如两个 n 位数 $A_nA_{n-1}\cdots A_i\cdots A_1A_0$ 和 $B_nB_{n-1}\cdots B_i\cdots B_1B_0$ 进行加法运算时，第 i 位 A_i 和 B_i 相加时，要考虑 $i-1$ 位是否产生了进位 C_{i-1}，需要把产生的进位也加进去。

如果将低位的进位也作为输入，和本位半加和再进行半加，就能得到一位全加和，两次半加中，将任何一次产生的进位作为本位进位。可以用两个半加器构造一个全加器。全加器的逻辑图如图 3.15 所示。

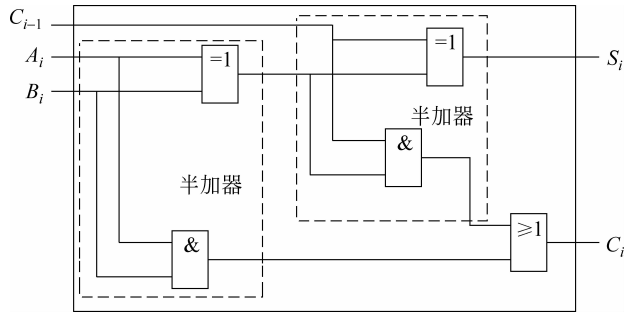


图 3.15　全加器的逻辑框图

这个电路有 3 个输入，即原始的数位 A_i 和 B_i 以及低位的进位 C_{i-1}，两个输出，即本位全加和 S_i 和进位输出 C_i，因此，真值表具有 8 行，如表 3.9 所示。

表 3.9　全加器的真值表

A_i	B_i	C_{i-1}	和 S_i	进位输出 C_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

2. 8 位加法器

1 位全加器仅仅完成一位二进制数的加法运算，要实现 8 位二进制数的加法运算，就需要用 8 个 1 位全加器电路。考虑一种最简单的方法，和手工进行两个 8 位数相加类似，采取逐位进位，将低位相加的进位输出作为高位的进位输入，即将每个全加器的进位输出端连接到左边相邻的全加器的进位输入端，最右边位的进位输入是 0，最左边位的进位输出将被舍弃（通常会生成溢出错误），就构成 8 位二进制数加法器。图 3.16 所示是 8 位加法器。除了逐位进位外，还可以有更复杂的进位逻辑，读者可阅读相关参考资料。

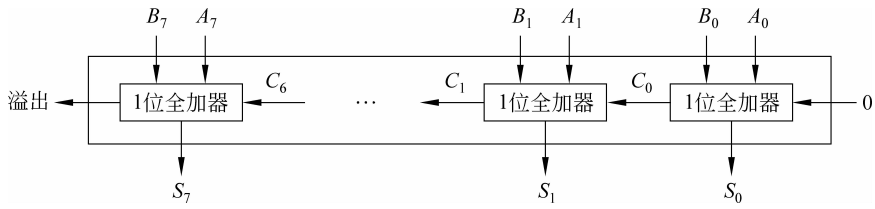


图 3.16　8 位加法器的逻辑框图

3.3.3　选择器

选择器又称为多路复用器，是由多个输入生成单个输出信号的通用电路。输出值等于该电路的多个输入值之一。选择器根据选择信号决定输出信号等于哪个输入信号。

图 3.17 是一个选择器的逻辑框图。选择控制线 S_0、S_1 和 S_2 用来决定输出信号 F 等于 D_0 到 D_7 的 8 条输入线中的哪一条的信号。

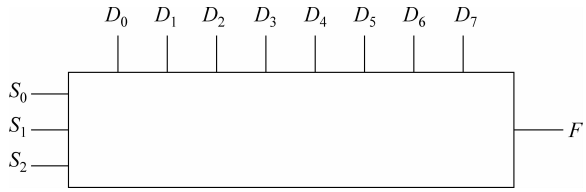


图 3.17　具有 3 条控制线的选择器的逻辑框图

3 条控制线的值译为一个二进制数，决定了发送输出信号的输入线。在第 2 章中介绍过 3 位二进制数字可以表示 8 个不同的值。注意，这些值只能从 0～7，对应的输出值为 D_0～D_7 中的某一个。因此，如果 S_0、S_1 和 S_2 都是 0，那么选择器的输出就是 D_0。如果 S_0、S_1 和 S_2 的值是 1、0、1，那么输出就是 D_5。

表 3.10 所示的功能表列出了 3 条输入控制线如何决定选择器的输出。

表 3.10　选择器的功能表

S_0	S_1	S_2	F
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	0	0	D_4
1	0	1	D_5
1	1	0	D_6
1	1	1	D_7

图 3.17 中的框图隐藏了执行选择器逻辑的复杂电路。用 8 个三输入的与门和一个输入的或门可以表示这个电路。

选择器可能有任意多条输入线和相应的控制线。一般来说，n 条输入控制线决定了选择 2^n 条数据线中的哪一条作为输出。

分配器是执行和选择器相反操作的电路。也就是说，它只有一个输入，根据 n 条控制线的值，决定将这个输入信号发送到 2^n 个输出中的某一个。

3.3.4 锁存器

数字电路的一个重要作用是存储信息，其存储功能是由时序电路来实现的，这种电路的输出信号反过来又用做电路的输入信号。也就是说，电路的下一个状态部分地由当前状态决定。时序电路有很多种，本节只分析 S-R 锁存器。

一个 1 位 S-R 锁存器存储一个二进制数字(1 或 0)。用不同的门，可以以不同的方式设计 S-R 锁存器。图 3.18 展示了一种用与非门设计的 S-R 锁存器。

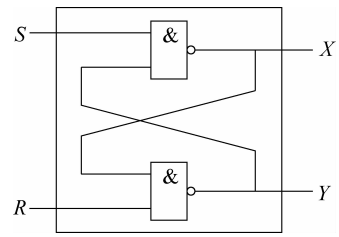


图 3.18 用与非门设计的 S-R 锁存器

这个电路的设计使 X 和 Y 总是互补的。即 $X=0$ 时，$Y=1$；反之亦然。把 X 的值看作电路的当前状态。因此，如果 $X=1$，电路存储的就是 1；如果 $X=0$，电路存储的就是 0。

用与非门设计的 S-R 锁存器的功能表如表 3.11 所示。

表 3.11 1 位 S-R 锁存器的功能表

S	R	X	Y	操 作
1	1	X	Y	保持
0	1	1	0	置位
1	0	0	1	复位
0	0	1	1	不用

锁存器有 4 种操作状态。第一个状态“保持”，意味着只要 $S=1$，$R=1$，电路的状态就维持不变(即 X 的值维持不变)。由与非门的性质可知，只有两个输入值都为 1 时，输出值才为 0，其他情况都为 1。假设现在 $S=1$，$R=1$ 被应用到输入，如果 $X=1$，根据与非门可计算出 $Y=0$。这样 Y 就和 S 一起反作用于上方的与非门，求得 $X=1$，确保 $X=1$ 保持不变。同理，如果 $X=0$，和 $R=1$ 一同作用于下方的与非门，求得 $Y=1$，Y 又和 S 一起反作用于上方的门，确保 $X=0$ 保持不变。

为了给该锁存器“置位”，S 由 1 变为 0，X 被置成 1。如果保持 $R=1$，S 又从 0 变为 1，那么此时锁存器的值 1 将保持不变。

当 $S=1$，$R=1$ 时，如果 R 从 1 变成 0，Y 被置成 1，反作用于上方的与非门，X 为 0，实

现了锁存器的“复位”功能。当 $S=1$ 时，R 又从 0 变成 1，此时锁存器的值保持为 0。

当 S、R 同时被置为 0 时，X、Y 都被强置为 0。由于这种情形下两种输出不互补，因此 $S=0$，$R=0$“不用”。使用与非门设计的 S-R 锁存器时，应避免 S、R 同时被置为 0。

只要合理控制 S 和 R 的值，S-R 锁存器电路就可以存储 0 或 1。把这个思想扩展到较大的电路，就可以设计出容量较大的存储设备。如把 8 个 S-R 锁存器组合在一起，就可以设计出 8 位锁存器。

3.3.5 计数器

计数是一种简单基本的运算，从 0、1、2、3、…，一直往下，计数器就是实现这种运算的逻辑电路，用以统计输入时钟脉冲 CP 的个数，以实现测量、计数和控制的功能。在数字系统中，计数器应用非常广泛，如计算机的控制器中对指令地址进行计数，以便顺序取出下一条指令，在运算器做乘法、除法运算时记下加法、减法次数等。

为了实现计数器的功能，先分析 3 位的计数器状态变化的规律，如表 3.12 所示。

表 3.12　3 位计数器的状态变化表

CP 顺序	Q_2	Q_1	Q_0	等效十进制数
0	0	0	0	0
1	0	0	1	1
2	0	1	0	2
3	0	1	1	3
4	1	0	0	4
5	1	0	1	5
6	1	1	0	6
7	1	1	1	7
8	0	0	0	0

从表 3.12 可以看出，最低位 Q_0 在每个脉冲 CP（0 到 1 或 1 到 0 的变化）到来时都会改变。Q_1 在 Q_0 为 1 后的下一个 CP 脉冲到来时会改变，即 $Q_{n+1}=\overline{Q_n}$，Q_2 在 Q_1 两次改变后的下一个 CP 脉冲到来时发生变化。

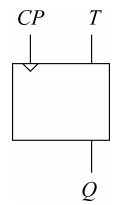


图 3.19　T 触发器的逻辑框图

可以用 T 触发器很简单地组成计数器。在数字电路中，凡在 CP 时钟脉冲控制下，根据输入信号 T 取值的不同，具有保持和翻转功能的电路，即当 $T=0$ 时能保持状态不变、$T=1$ 时一定翻转的电路，都称为 T 触发器。T 触发器的逻辑框图如图 3.19 所示，功能表如表 3.13 所示。

用 T 触发器实现 3 位计数器的原理如图 3.20 所示，最低位 Q_0 的 T 端始终接 1，这样不管上一次 Q_0 是什么，CP 到来时它都会翻转，输出端接 Q_1 的 T 端，只有当 Q_0 输出为 1

后的下一个 CP 到来时 Q_1 才会翻转。

表 3.13 T 触发器的功能表

T	Q_n	Q_{n+1}	功　　能
0	0	0	$Q_{n+1}=Q_n$ 保持
0	1	1	
1	0	1	$Q_{n+1}=\overline{Q_n}$ 翻转
1	1	0	

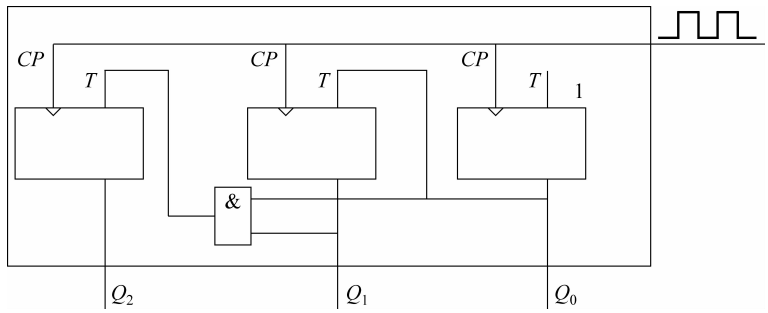


图 3.20　用 T 触发器实现 3 位计数器的原理图

3.3.6　集成电路

集成电路(Integrated Circuit,IC)又称芯片是一种微型电子器件,采用一定的工艺,把一个电路所需的晶体管、二极管、电阻、电容和电感等元件及布线互连起来,制作在一块半导体硅片上,硅片封装在塑料或陶瓷管壳中,边缘有引脚,可以焊接在电路板上或插入合适的插座中。每个引脚和一个门的输入或输出连接,也可能和电源或接地连接。

集成电路是根据它包含的门数分类的。一个小规模集成电路(SSI 芯片)只有几个独立的门。图 3.21 是一个有 14 个引脚的 SSI 芯片,其中 8 个引脚用做芯片的输入(1、2、4、5、9、10、12、13),4 个用做输出(3、6、8、11),1 个接地(7),1 个接电源(14)。

市面上有各种集成电路,74 系列集成电路 74LS32 双输入四或门是常见的 SSI 芯片,其引脚如图 3.21 所示。

超大规模集成电路(Very Large Scale Integration Circuit,VLSI)中一般多于 100 000 个门。这是否就意味着需要 300 000 个引脚呢? VLSI 芯片上的门不像小规模集成电路中的门一样,它们一般不是完全独立的。VLSI 芯片上嵌入的电路具有很高的门引脚比。也就是说,许多门被组合在一起,创建的复杂电路只需要很少的输入和输出值。前面介绍的选择器是这种电路的一个例子。

计算机中最重要的集成电路是中央处理器(Central Processing Unit,CPU)。每个 CPU 芯片有大量的引脚,计算机系统的所有通信都是通过这些引脚完成的,它们把 CPU 和本身也是集成电路的存储器与 I/O 设备连接在一起。

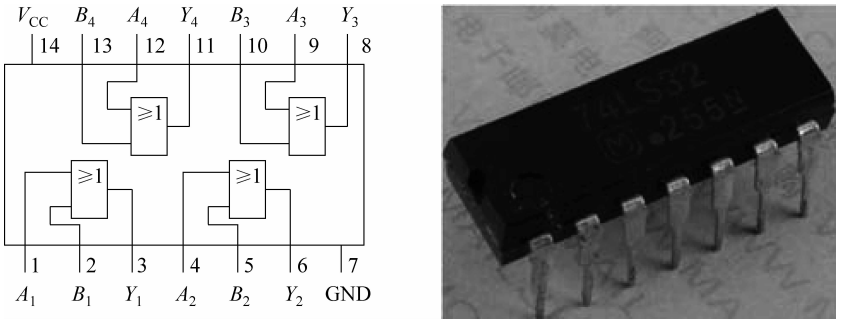


图 3.21　74LS32 双输入四或门 SSI 芯片引脚

3.4　计算机硬件系统

硬件是计算机运行的物质基础，计算机硬件系统也是不断发展的。早期的计算机都是基于冯·诺依曼基本思想构成的，计算机由运算器、控制器、存储器、输入和输出设备五大部件以及它们之间的连线构成。随着输入设备、输出设备、存储器数量的增多，现代计算机在冯·诺依曼计算机的基础上将各个部件通过总线和主板连接起来，有效地减少了部件之间的连线。单个处理器的能力逐步趋于极限，为了提高计算机的处理能力，将多个处理器以各种方式组合在一起构成并行计算机。

3.4.1　计算机体系结构

平常看到的计算机硬件大多如图 3.22 所示，可以分为 3 部分：输入设备、输出设备和主机箱。

图 3.22　计算机的三大部分

图 3.22 中的输入设备包括键盘、鼠标、摄像头，计算机还可以有其他的输入设备，输入设备的功能是将人们熟悉和使用的信息转换成计算机能识别的形式，输入计算机中。图中的输出设备包括显示器、打印机和音箱，计算机还可以有其他的输出设备，输出设备的功能是将计算机处理的结果转换成人们熟悉的文字、声音或图形图像的形式，反馈给

人。计算机的主要功能由主机箱中的部件来完成，所以计算机的主机箱中的部件必须完成信息处理、控制和存储。从表面上看，计算机的硬件结构如图 3.23 所示。实线反映了数据传送方向，虚线反映了控制。

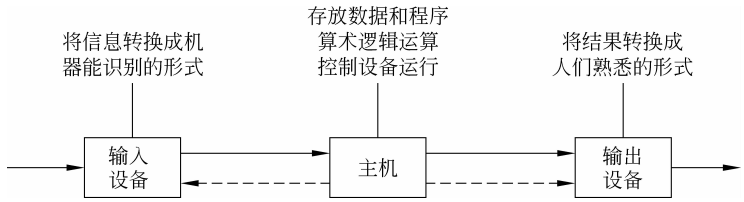


图 3.23　计算机硬件结构简化示意图

主机箱中到底都有哪些部件呢？如图 3.24 所示。打开主机箱，能够看见的部分包括硬盘、光驱、电源、CPU 风扇、各种插槽和主板等。硬盘和光驱是外部存储器，用于永久存储信息；电源为整个计算机供电；CPU 风扇（含散热片）将 CPU 的热量快速传导出来；各种插槽用来插入扩展卡，连接设备。主板是最大的那块电路板，它是微型计算机的主要部件，其上装有计算机的核心部件中央处理器（Central Processing Unit，CPU，在图 3.24 中 CPU 风扇下面）和内存，以及与各部件之间连接的线路。CPU 由运算器和控制器组成，集成在一块芯片上，运算器进行算术逻辑运算，控制器对整个计算机的运行进行有条不紊的控制；内存用于存放计算机正在执行的程序（指由若干条操作命令构成的序列）和数据。因此，典型的计算机硬件逻辑结构如图 3.25 所示。

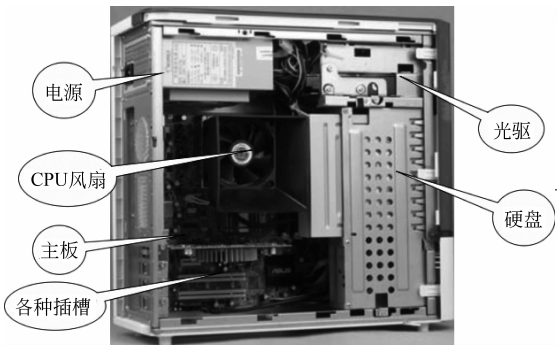


图 3.24　主机箱内部构成

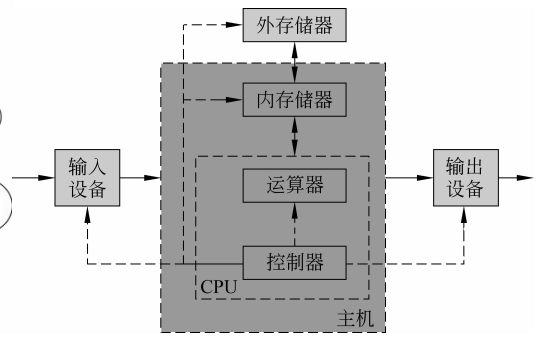


图 3.25　计算机硬件逻辑结构图

虽然有不同类型、不同规模和不同性能的计算机，但从本质上说，按照这种方式构成的计算机都称为冯·诺依曼体系结构的计算机，它是按照冯·诺依曼基本思想构造的，其核心是存储程序。其主要特征可以归纳如下。

（1）计算机中的数据和指令都用二进制表示。

（2）采用存储程序的思想。要执行的程序和数据以二进制形式存放在存储器中，自动顺序执行程序。

（3）计算机硬件由运算器、控制器、存储器、输入设备和输出设备五大部件组成。各部件的主要功能可以归纳如下。

① 运算器。它是计算机对数据进行加工的主要部件，完成算术运算和逻辑运算。

② 控制器。用于控制计算机的各部件协调工作。控制器负责从内存储器中取出指令并根据该指令向有关部件发出控制命令，从而使整个处理过程有条不紊地进行。

③ 存储器。它是计算机的记忆装置，用于存放程序和数据，分为内存和外存。

④ 输入设备。用于从外界将程序和数据输入计算机，供计算机处理。

⑤ 输出设备。用于将计算机处理后的结果转换成外界能够识别和使用的数字、文字、声音、电压等形式的信息并输出给用户。

3.4.2 总线、主板和设备接口

早期的冯·诺依曼计算机的各个部件之间是直接连接的，但随着各种外部设备的出现，需要连接到计算机中的外设逐步增多，因此使用总线和主板来连接计算机中的各种设备。

1. 总线

计算机内主要有3种信息在流动：数据信息、地址信息和控制信息。数据信息包括源程序、原始数据、中间结果和最终结果，它们从输入设备或存储器读入运算器进行运算，计算结果再送入存储器或传送到输出设备，地址信息用来指示各种设备的地址，一般从CPU送入外部设备和存储器，在图3.25中用实线箭头表示。控制信息是由控制器向各部分发出的控制命令，指挥各部件协调工作，在图3.25中用虚线箭头表示。

计算机硬件系统的各个部件之间需要传送数据、地址和控制信息，需要用电线连接起来，目前流行的互连方式是使用总线。总线是计算机内部传输信息的公共信息通道，是计算机系统的骨架，如图3.26所示。根据总线上传输的信息，可以将总线分为数据总线、地址总线和控制总线。数据总线上传输数据和指令，地址总线上传输内存或外设的地址，控制总线上传输控制命令。它们都不是一根线，而是一组线。

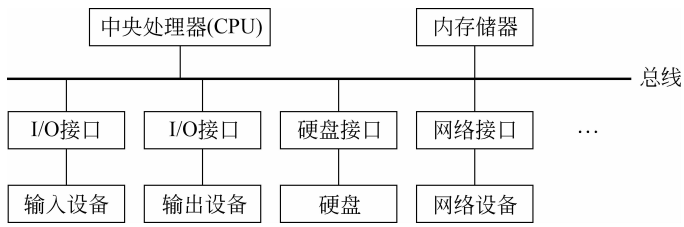


图3.26 总线及其各部件的互连

用来连接计算机各功能部件的总线有时又称为系统总线、内总线或板级总线，因为该总线将各部件连接起来，构成一个完整的计算机系统。

总线的特征是共享传送信号，多个设备连接到总线上，一个设备传送的信号可以被其他所有连接到总线上的设备所接收，可以理解为图3.26所示的公共通道。如果两个设备同时传送信号，它们传送的信号就会重叠，从而造成混乱。因此，某一时刻只能有一个设

备利用总线传输信息。计算机中必须设置总线使用的仲裁和管理机制。

2. 主板

从物理角度看，总线是一组电线，把计算机的各个部件连接在一起。这组电线直接印制在电路板上，延伸到各部件，在个人计算机上，这个电路板称为主板。打开微型计算机的机箱，可以看到一块大的矩形板，就是主板，又叫主机板（Mainboard）、系统板（Systemboard）或母板（Motherboard），是微型计算机最基本的也是最重要的部件之一。主板上面安装了组成计算机的主要电路系统。主板上一般有 BIOS 芯片、CMOS、CPU 插座、内存条插槽、南北桥芯片等芯片组、各种 I/O 接口、扩充插槽、主板及插卡的直流电源供电接插件等元件。

计算机的各个部件通过主板承载和连接，同时通过南桥芯片和北桥芯片提供常用外存和 I/O 设备的接口，然后通过总线把各个部件连接起来。主板在整个微机系统中扮演着举足轻重的角色，主板的性能影响着整个微型计算机系统的性能。主板上大都有 6～15 个扩展插槽，供 PC 外围设备的控制卡（适配器）插接。通过更换这些插卡，可以对微型计算机的相应子系统进行局部升级，使厂家和用户在配置机型方面有更大的灵活性。图 3.27 所示为某型号的主板示意图。

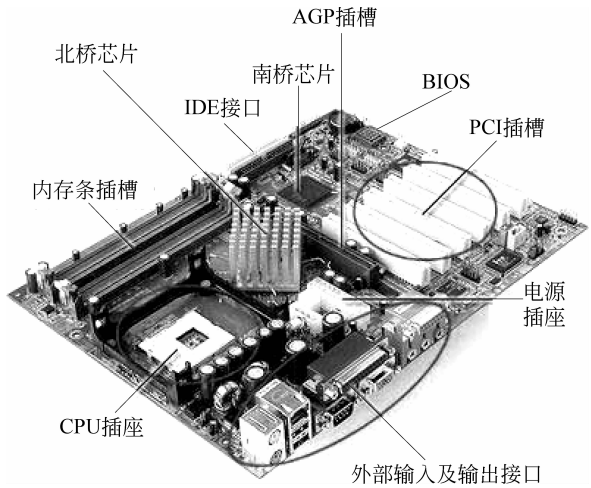


图 3.27　某微型计算机的主板示意图

个人计算机中的主板有以下作用。

（1）提供总线。将计算机的不同部件连接在一起，使它们相互传递信息。

（2）提供支撑计算机运行的各种芯片。主板上提供的芯片主要有 CMOS 芯片、BIOS 芯片、北桥芯片和南桥芯片等。

（3）提供各种插座和插槽。计算机中的许多部件可以用“插入”方式与计算机的其他部件连接。主板上提供的插座和插槽主要有 CPU 插座、内存插槽和一些扩展插槽。

（4）提供外置 I/O 设备接口。包括键盘鼠标接口、USB 接口、外存接口、音频接口、显示器接口、网络接口等，用于连接相应的外部设备。

3. 主板上的主要芯片

1）BIOS 芯片

BIOS(Basic Input Output System)是计算机系统的基本输入输出系统，是主板上的一个 ROM 芯片，它保存着计算机最重要的基本输入输出程序、系统设置信息、开机后自检程序和系统自启动程序，实现计算机对输入输出设备的基本控制。BIOS 是主板的核心，负责从计算机开始加电到完成操作系统引导之前的各个部件和接口的初始化、检测和运行管理，存放在 Flash ROM 中，不需要供电就可保持数据不丢失，容量一般多为 1MB、2MB 或 4MB。用户可以直接使用厂商提供的升级程序来修改内容。

2）CMOS 芯片

CMOS(Complementary Metal Oxide Semiconductor)芯片是主板上的一块可读写的 RAM 芯片，用来保存 BIOS 对计算机硬件配置信息和用户对某些参数的设定(如日期、时间、启动设置等)。CMOS 芯片一般由主板的电池供电，即使计算机关机，信息也不会丢失。CMOS 的内容是 BIOS 中的有关程序设置的结果。早期的 CMOS 芯片是一块单独的芯片 MC146818A(DIP 封装)，共有 64 个字节存放系统信息。现在大多集成到南桥芯片中，容量达到 128～256B。如果 CMOS 中数据损坏，计算机将无法正常工作，为了确保 CMOS 数据不被损坏，主板厂商都在主板上设置了开关跳线，一般默认为关闭。当要对 CMOS 数据进行更新时，可将它设置为可改写。进入 BIOS 对计算机参数进行设置的程序界面如图 3.28 所示。

图 3.28　CMOS 参数设置

3）芯片组

北桥芯片和南桥芯片合称为芯片组，是主板的核心组成部分，在很大程度上决定了主板的功能和性能。北桥芯片主要负责 CPU、内存、显卡三者间的数据传输，同时还通过特定的数据通道和南桥芯片相连接。由于发热量较大，因而需要散热片散热。南桥芯片负责 CPU 和 I/O 设备接口之间的通信(I/O 总线)，如 PCI 总线、USB、LAN、ATA、SATA、音频控制器、键盘控制器、实时时钟控制器、高级电源管理等，这些技术一般比较稳定，所以不同芯片组中的南桥芯片差别不大，不同的只是北桥芯片。芯片组的构成及与各部分