

计算机专业导论

战德臣

哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

第2讲 符号化、计算化与自动化

战德臣

哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

本讲学习什么？

---符号化-计算化-自动化：0和1的思维

战德臣

哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



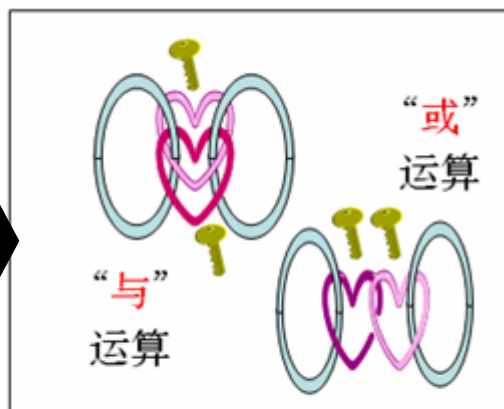
Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

符号化-计算化-自动化

0和1的思维概述



现象和思维均可表达成0和1

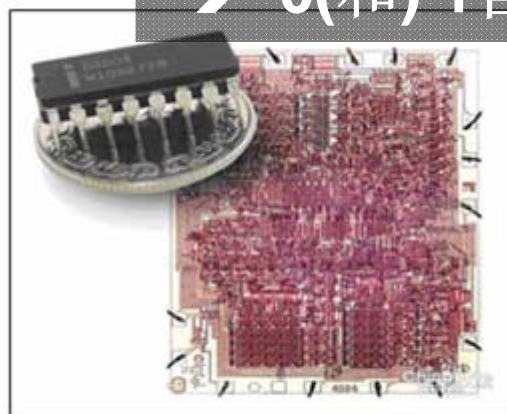


用0和1可进行算术与逻辑运算

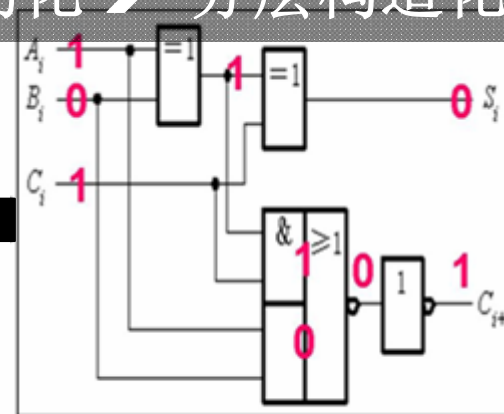


0和1可用电子技术实现

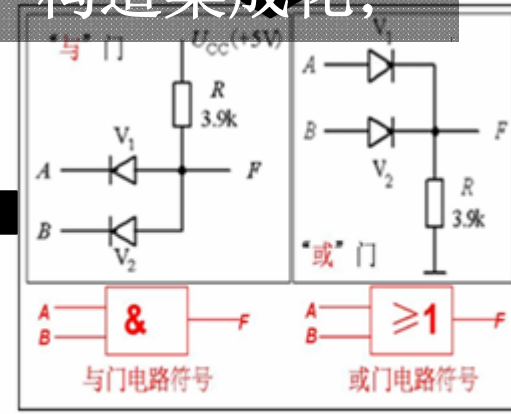
语义符号化 → 符号计算化 → 计算0(和)1化
→ 0(和)1自动化 → 分层构造化 → 构造集成化;



芯片--集成的复杂组合逻辑电路



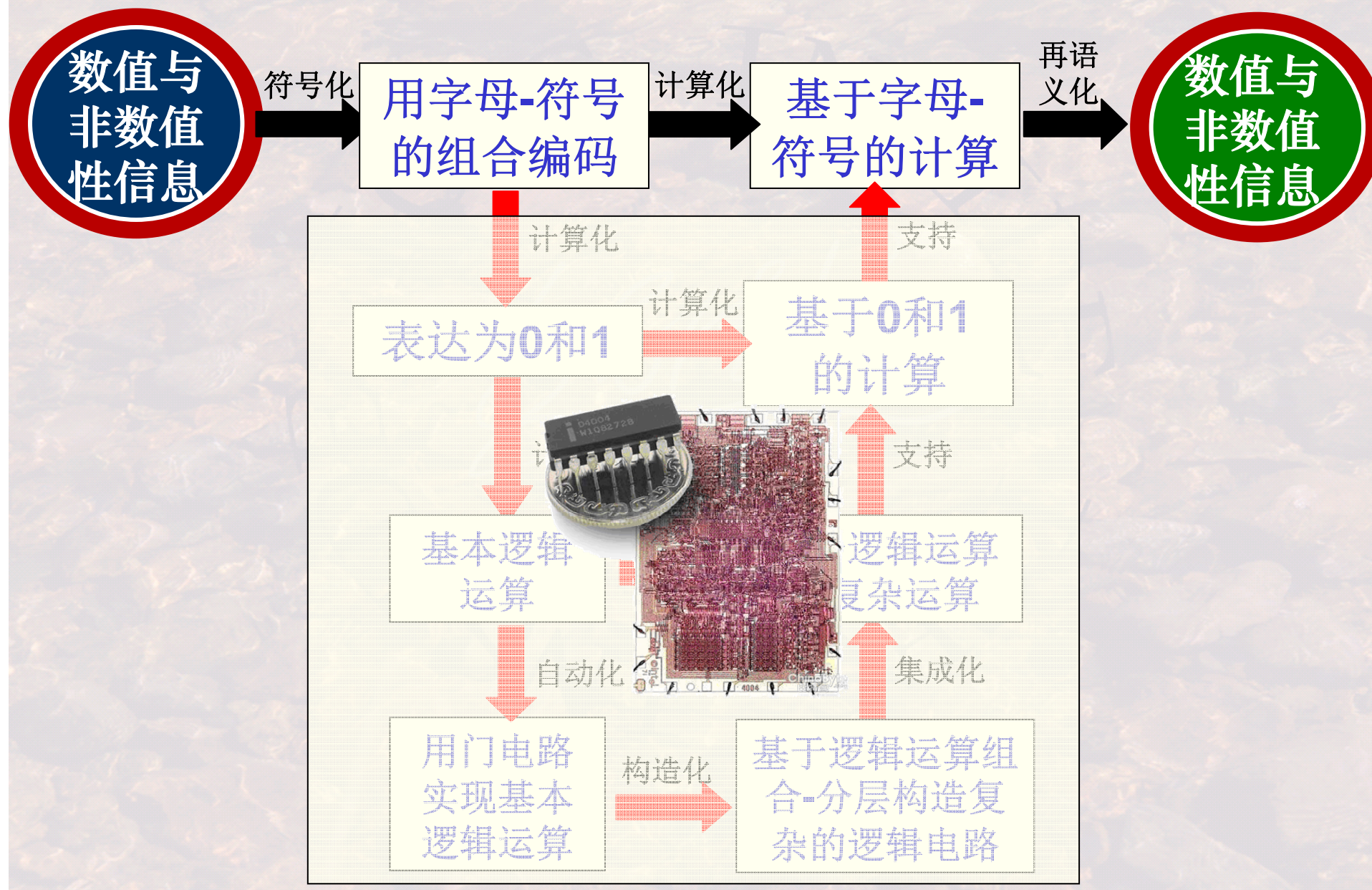
分层构造实现复杂运算



电子技术实现逻辑运算

符号化-计算化-自动化

0和1的思维概述



信息如何用0和1表示？

---数值性信息：二进制

---非数值性信息：编码

战德臣

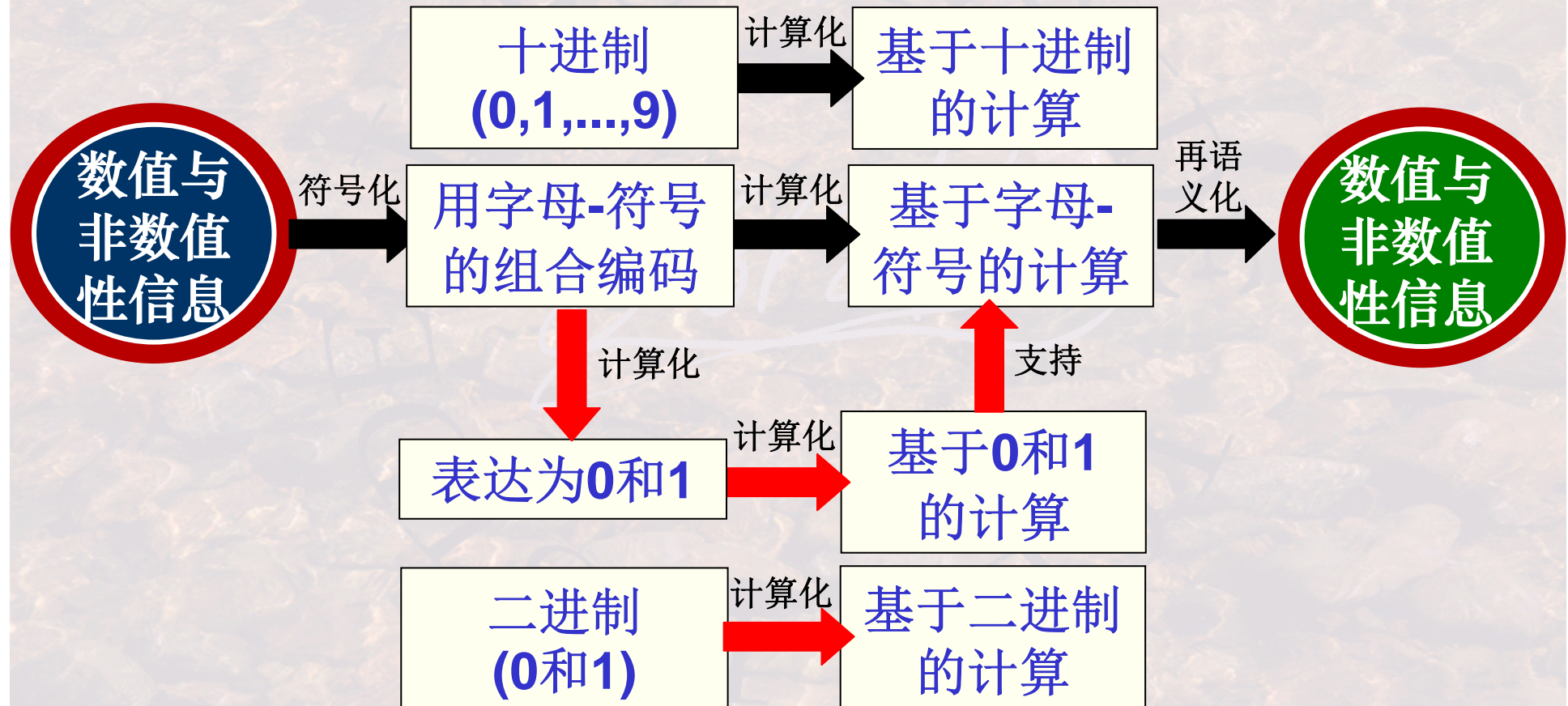
哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

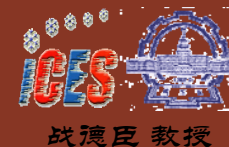
信息如何用0和1表示？

(1)基本思维



信息如何用0和1表示?

(2) 进位制?



数值性信息可以用进位制表示

◆进位制：用数码和带有权值的数位来表示有大小关系的数值性信息的表示方法。

◆十进制

◆二进制

10^3	10^2	10^1	10^0	10^{-1}	10^{-2}
3	2	1	0	.	-1 -2
(3	2	8	5	.	3 4)

2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0 . 2^{-1} 2^{-2}

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

————— 数位的权值

————— 数位

例如： (1 1 1 1 0 1 0 1 . 0 1)_二 ————— 二进制数

$$\begin{aligned} &= 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 \\ &+ 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = (245.25)_+ \end{aligned}$$

信息如何用0和1表示？

(2) 编码与ASCII码？



非数值性信息可以用编码表示

◆**编码**：编码是以若干位数码或符号的不同组合来表示非数值性信息的方法，它是人为地将若干位数码或符号的每一种组合指定一种唯一的含义。

例如：0----男， 1----女

再如：000----星期一 001----星期二
010----星期三 011----星期四
100----星期五 101----星期六
110----星期日

ASCII码----英文字母符号的编码

◆**ASCII码**是英文字母与符号的0,1型编码方法，是用7位0和1的不同组合来表示10个数字、26个英文大写字母、26个英文小写字母及其一些特殊符号的编码方法，是信息交换的标准编码。

◆ASCII码: American Standard Code for Information Interchange

B₇ B₆ B₅ B₄ B₃ B₂ B₁ B₀

0 x x x x x x x

0 0 1 1 0 0 0 1 “1”
0 1 0 0 1 1 1 0 “N”

信息如何用0和1表示?

(2) 编码与ASCII码?

信息	存储	解析规则
We are students	01010111 01100101 00100000 01100001 01110010 01100101 00100000 01110011 01110100 01110101 01100100 01100101 01101110 01110100 01110011	0/1串按8位分隔一个字符，查找ASCII码表映射成相应符号

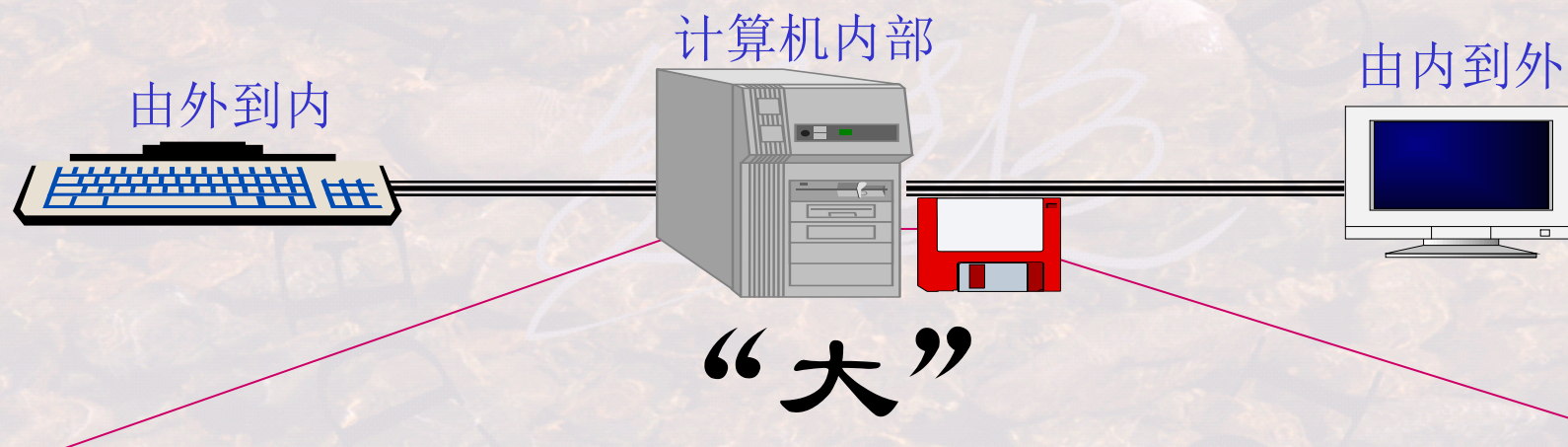
$b_3b_2b_1b_0 \backslash b_8b_5b_4$	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	.	7	G	W	g	w
1000	BS	CAN	(8	H	X	h	x
1001	HT	EM)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[k	{
1100	FF	FS	,	<	L]	l	
1101	CR	GS	-	=	M	\	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

信息如何用0和1表示?

(3) 汉字与汉字编码?

汉字的编码

◆**汉字内码**: 汉字在计算机内部采用汉字内码存储, 汉字内码是一两字节且最高位均为1的0, 1型编码



用0和1编码汉字, 每个汉字在计算机内部由 2个字节表示

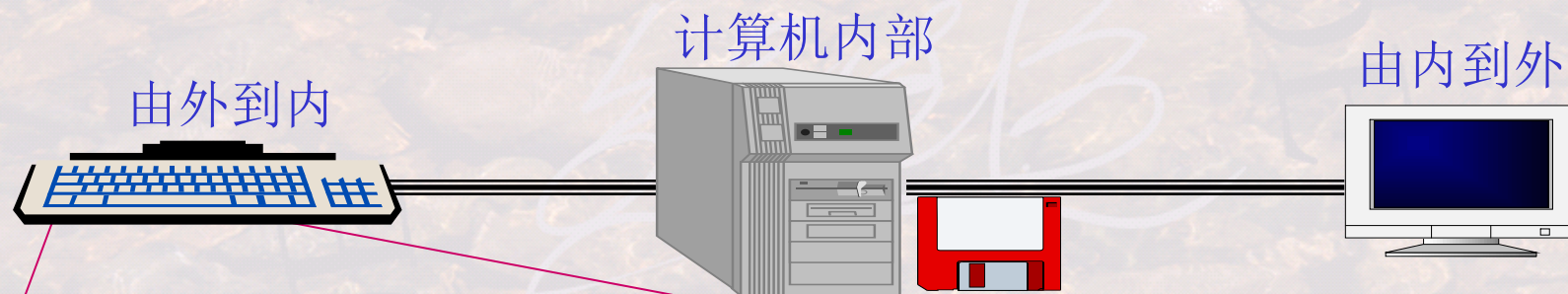
	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
国标码	0	0	1	1	0	1	0	0	0	1	1	1	0	1	1	1
	↓								↓							
(机)内码	1	0	1	1	0	1	0	0	1	1	1	0	1	1	1	1

信息如何用0和1表示？

(3) 汉字与汉字编码？

汉字的编码

◆**汉字输入码**是用键盘上的字母符号编码每一汉字的编码,它使人们通过键入字母符号代替键入汉字。



输入码有若干：拼音码、字型码、区位码… …

拼音码： xing

双拼码： x;

其中，‘x’表声母x, 而‘;’表韵母ing

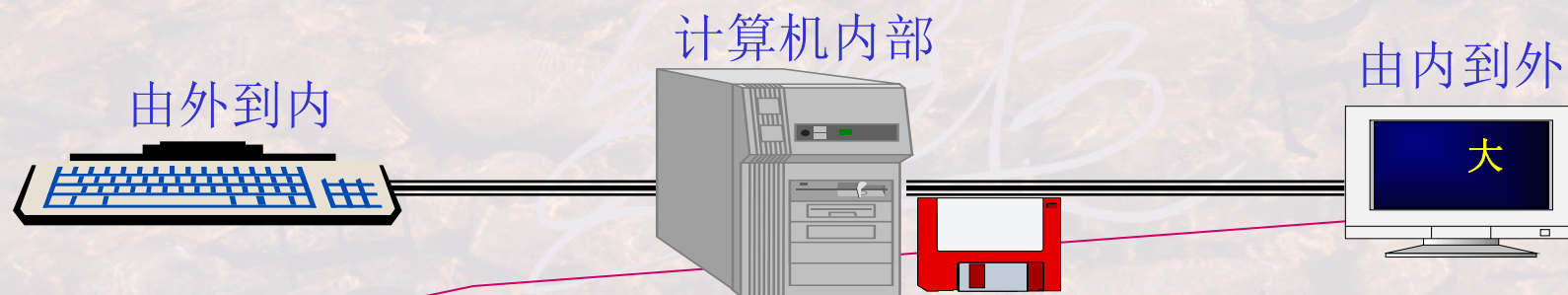
五笔字型码： gajf

其中, g表字根 “-”, a表开下的草字头, j表右侧立刀, f表下面土字

“型”

汉字的编码

◆**汉字字形码**是用0和1编码无亮点和有亮点像素,形成汉字字形的一种编码。依据字形码通过显示器或打印机输出汉字。



用0和1编码无亮点和有亮点形成字形信息, 便于显示… …

汉字字形码是一种字模点阵码。也有不同的处理汉字点阵信息的编码，如向量编码等

“大”

[illegible]

The diagram illustrates the relationship between different computer components. On the left, a keyboard is labeled "由外到内" (From outside to inside). In the center is a large grey computer tower labeled "计算机内部" (Inside the computer). To the right of the tower is a red 3.5-inch floppy disk. On the far right is a CRT monitor labeled "由内到外" (From inside to outside). Two thick black lines connect the keyboard to the tower, and another two thick black lines connect the tower to the monitor, indicating data flow or connection paths.

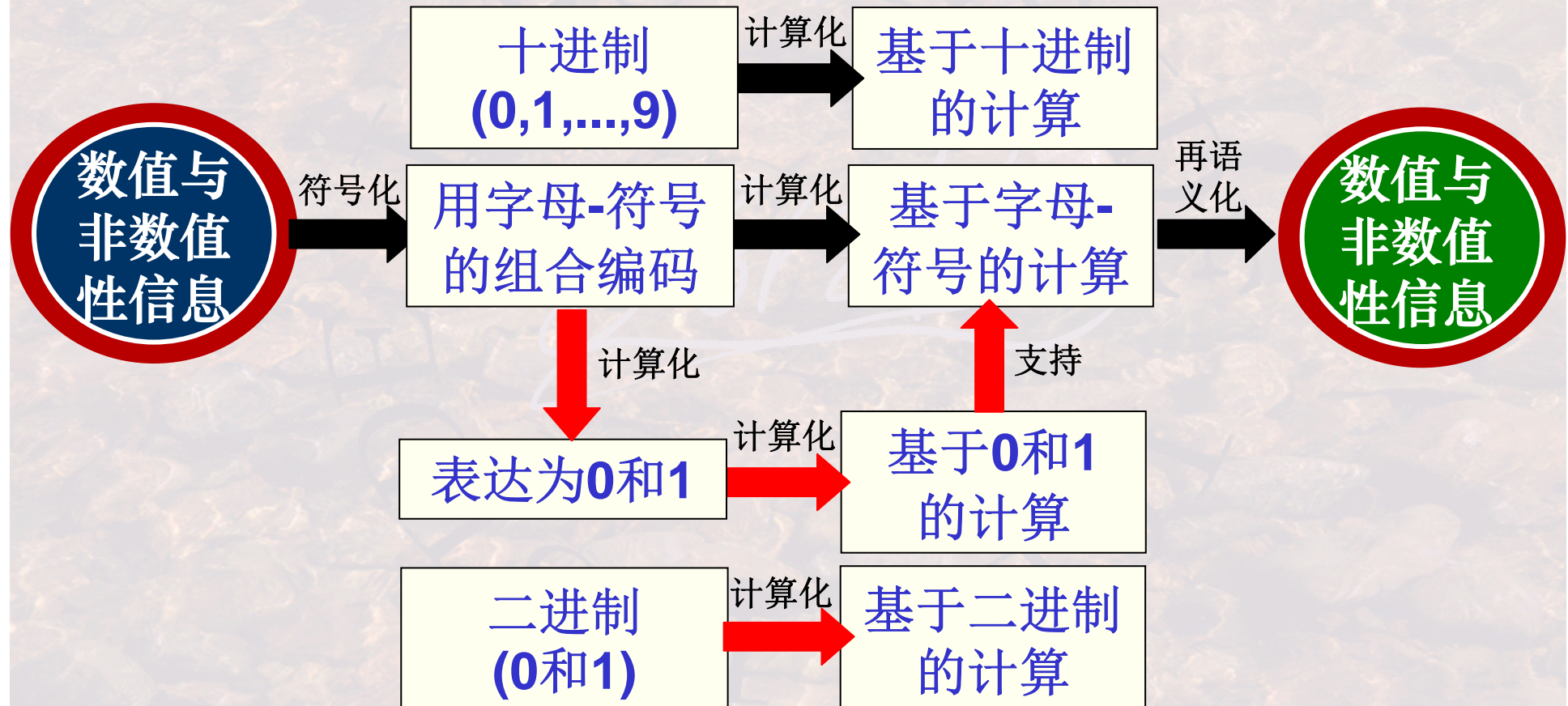
```

0000001100000000
0000001100000000
0000001100000000
0000001100000100
1111111111111111
0000001100000000
0000001100000000
0000001100000000
0000001100000000
0000001110000000
000011001000000
000110000100000
001100000110000
001000000011000
0010000000001110
1100000000000100

```


信息如何用0和1表示？

(4)基本思维回顾



如何用0和1进行计算？

---0和1的组合及其变化

---基本逻辑运算

战德臣

哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

如何用0和1进行计算?

(1)看易经如何基于0和1计算?

什么是易经?

阴(六)

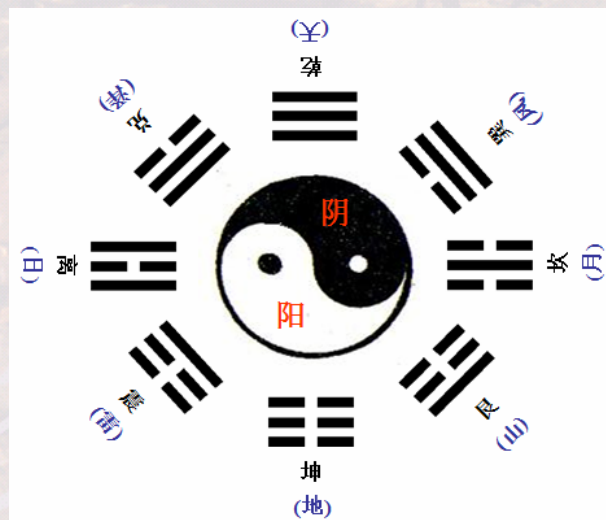
阳(九)

(九三爻)

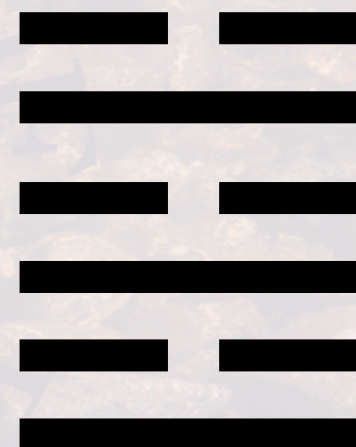
(六二爻)

(九一爻)

(三画卦)



八纯	乾	震	坎	艮	坤	巽	离	兑
上世	姤	遁	否	观	剥	大过	渐	归妹
一世	临	泰	大壮	夬	乾	姤	遁	否
二世	夬	乾	姤	遁	否	观	剥	大过
三世	否	观	剥	大过	渐	归妹	归妹	归妹
四世	归妹	归妹	归妹	归妹	归妹	归妹	归妹	归妹
五世	归妹	归妹	归妹	归妹	归妹	归妹	归妹	归妹
游魂	归妹	归妹	归妹	归妹	归妹	归妹	归妹	归妹
归魂	归妹	归妹	归妹	归妹	归妹	归妹	归妹	归妹



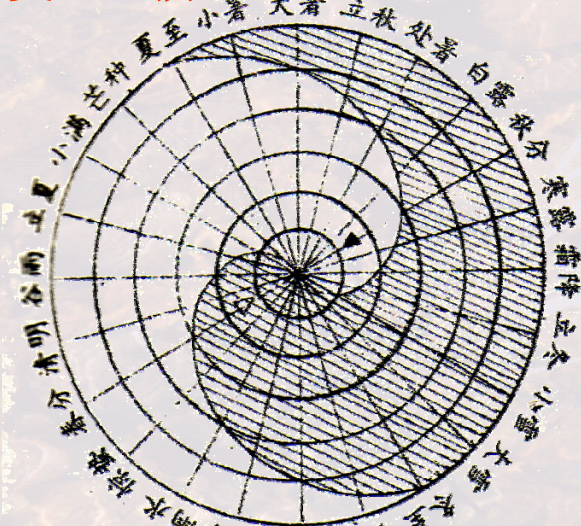
(六画卦)

如何用0和1进行计算?

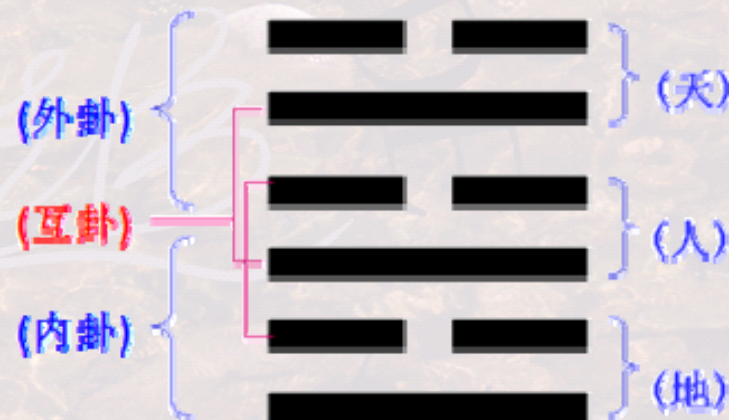
(1)看易经如何基于0和1计算?

将现象抽象为符号，进行符号组合，利用符号组合及其变化表达自然现象，进行计算！ ---卦是如何变化到另一卦的?

夏至一阴生



冬至一阳生



(六画卦)



如何用0和1进行计算?

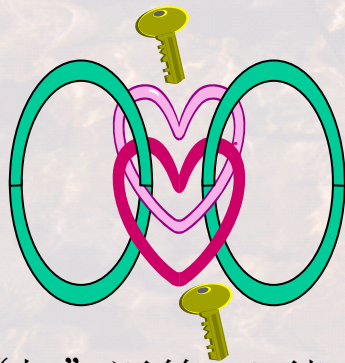
(2)看逻辑如何基于0和1计算?

基本逻辑运算

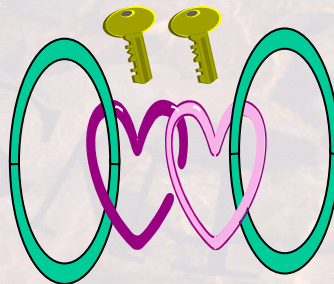
一个命题由语句表述, 即内容为“真”或为“假”的一个判断语句!

如果命题由X, Y, Z等表示, 其值可能为“真”或为“假”, 则两个命题X, Y之间是可以进行运算的:

- “与”运算(AND): 当X和Y都为真时, $X \text{ AND } Y$ 也为真; 其他情况, $X \text{ AND } Y$ 均为假。
- “或”运算(OR): 当X和Y都为假时, $X \text{ OR } Y$ 也为假; 其他情况, $X \text{ OR } Y$ 均为真。
- “非”运算(NOT): 当X为真时, NOT X为假; 当X为假时, NOT X为真。
- “异或”运算(XOR): 当X和Y都为真或都为假时, $X \text{ XOR } Y$ 为假; 否则, $X \text{ XOR } Y$ 为真。



“与”运算: 两把钥匙都有才能开门



“或”运算: 只要有任何一把钥匙便能开门

$$F = X \text{ AND } Y \text{ OR } (Z \text{ AND } (\text{NOT } Y))$$

$$E = (\text{NOT } X) \text{ AND } ((Y \text{ AND } (\text{NOT } Z)))$$

$$P = (\text{NOT } X) \text{ AND } (\text{NOT } Y)$$

如何用0和1进行计算?

(2)看逻辑如何基于0和1计算?

用0和1来表示逻辑运算

■ “与” 运算**AND**:

有0为0，全1为1

■ “或” 运算**OR**:

有1为1，全0为0

■ “非” 运算**NOT**:

非0则1，非1则0

■ “异或” 运算**XOR**:

相同为0，不同为1

注: 1表示 真, 0表示 假

	0	0	1	1
AND	0	1	0	1
	0	0	0	1

	0	0	1	1
OR	0	1	0	1
	0	1	1	1

NOT	0	1
	1	0

	0	0	1	1
XOR	0	1	0	1
	0	1	1	0

将减法变为加法？

---机器数的原码与补码

---将减法变为加法来进行

战德臣

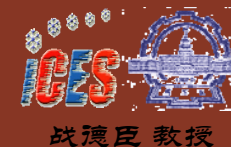
哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on Intelligent
Computing for Enterprises & Services,
Harbin Institute of Technology

将减法变为加法?

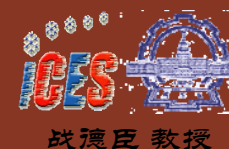
(1) 数值的符号如何表示呢—机器数及其原码、反码和补码



十进制数	+ 244	- 244
二进制数	+ 1 1 1 1 0 1 0 0	- 1 1 1 1 0 1 0 0
机器数-原码	0 1 1 1 1 0 1 0 0	1 1 1 1 1 0 1 0 0
机器数-反码	0 1 1 1 1 0 1 0 0	1 0 0 0 0 1 0 1 1
机器数-补码	0 1 1 1 1 0 1 0 0	加 0 0 0 0 0 0 0 1 1 0 0 0 0 1 1 0 0

将减法变为加法?

(1) 数值的符号如何表示呢—机器数及其原码、反码和补码



机器数的原码、反码和补码表示规则

真实数值 (带符号的 n 位 二进制数)	十进制数	机器数($n+1$ 位二进制数,其中第 $n+1$ 位表符号,0 表示正号,1 表示负号)		
		原码	反码	补码
$+11\dots11$	$+(2^n-1)$	0 11...11	0 11...11	0 11...11
$+10\dots00$	$+2^{n-1}$	0 10...00	0 10...00	0 10...00
$+00\dots00$	$+0$	0 00...00	0 00...00	0 00...00
$-00\dots00$	-0	1 00...00	1 11...11	0 00...00
$-10\dots00$	-2^{n-1}	1 10...00	1 01...11	1 10...00
$-11\dots11$	$-(2^n-1)$	1 11...11	1 00...00	1 00...01
$-100\dots00$	-2^n	-	-	1 00...00
		正数的原码、反码同补码形式是一样的。最高位为 0 表示正数		
		负数的最高位为 1,表示负数。其余同真实数值的二进制数。	负数的最高位为 1,表示负数。其余在真实数值的二进制数基础上逐位取反。	负数的最高位为 1,表示负数。其余在反码基础上最低位加 1 后形成的。它的负数不包括 0, 但包括 -2^n
		机器数由于受到表示数值的位数的限制, 只能表示一定范围内的数。超出此范围则为“溢出”		

将减法变为加法？

(2) 使用补码可使减法变加法, 你相信吗？



数值的正负符号也可和数值一样参与运算：补码运算示意

$$(+7) + (+3) = (+10)$$

$$\begin{array}{r} 0\ 0111 \\ +) 0\ 0011 \\ \hline 0\ 1010 \end{array}$$

$$(-5) + (-7) = (-12)$$

$$\begin{array}{r} 1\ 1011 \\ +) 1\ 1001 \\ \hline 1\ 0100 \end{array}$$

$$(10) + (-3) = (7)$$

$$\begin{array}{r} 0\ 1010 \\ +) 1\ 1101 \\ \hline 0\ 0111 \end{array}$$

$$(-7) + (-12) = \text{溢出}$$

$$\begin{array}{r} 1\ 1001 \\ +) 1\ 0100 \\ \hline 0\ 1101 \end{array}$$

加减乘除都可转换成加法来实现, 加法又可由与、或、非、异或等逻辑运算来实现
---只要实现了基本逻辑运算, 便可实现任何的计算

用逻辑实现加法，用硬件实现加法？

战德臣

哈尔滨工业大学 教授·博士生导师
教育部大学计算机课程教学指导委员会委员



Research Center on **I**ntelligent
Computing for **E**nterprises & **S**ervices,
Harbin **I**nstitute of **T**echnology

用逻辑实现加法，用硬件实现加法？

(0) 回顾一下基本逻辑运算？

基本逻辑运算

注：1表示 真，0表示 假

- “与” 运算 **AND**:

有0为0，全1为1

	0	0	1	1
AND	0	1	0	1
	0	0	0	1

- “或” 运算 **OR**:

有1为1，全0为0

	0	0	1	1
OR	0	1	0	1
	0	1	1	1

- “非” 运算 **NOT**:

非0则1，非1则0

NOT	0	1
	1	0

- “异或” 运算 **XOR**:

相同为0，不同为1

	0	0	1	1
XOR	0	1	0	1
	0	1	1	0

用逻辑实现加法, 用硬件实现加法?

(1) 一位二进制数的加减法?



基于二进制的算术运算

◆ 计算规则简单, 与逻辑运算能够统一起来。

	0	1	0	1
加法运	+ 0	+ 0	+ 1	+ 1
算规则	<u>0</u>	<u>1</u>	<u>1</u>	<u>0</u>

$$\begin{array}{r} + \\ A_i \\ B_i \\ \hline \end{array}$$

$$\begin{array}{cc} C_{i+1} & S_i \end{array}$$

$$\text{不考虑进位} \begin{cases} S_i = A_i \text{ XOR } B_i \\ C_{i+1} = A_i \text{ AND } B_i \end{cases}$$

$$\text{考虑进位} \begin{cases} S_i = (A_i \text{ XOR } B_i) \text{ XOR } C_i \\ C_{i+1} = ((A_i \text{ XOR } B_i) \text{ AND } C_i) \text{ OR } (A_i \text{ AND } B_i) \end{cases}$$

$$\begin{array}{r} + \\ A_i \\ B_i \\ C_i \\ \hline \end{array}$$

$$\begin{array}{cc} C_{i+1} & S_i \end{array}$$

用逻辑实现加法，用硬件实现加法？

(2) 用电路开关实现基本逻辑运算？

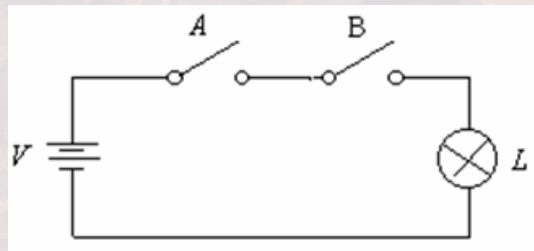
实现**0**和**1**的基本元器件：电信号和继电器开关



数字信号：高电平为1，低电平为0

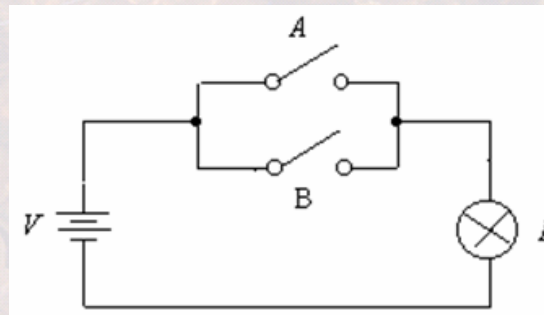
◆用继电器开关实现基本逻辑运算

A、B合上为1，断开为0；L灯亮为1，灯灭为0



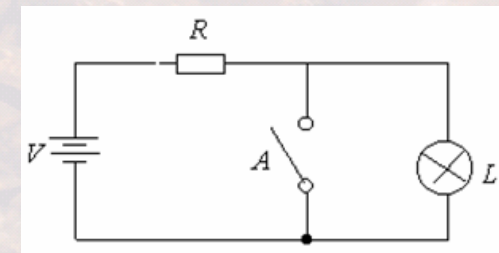
“与”运算电路

$$L = A \text{ and } B$$



“或”运算电路

$$L = A \text{ or } B$$



“非”运算电路

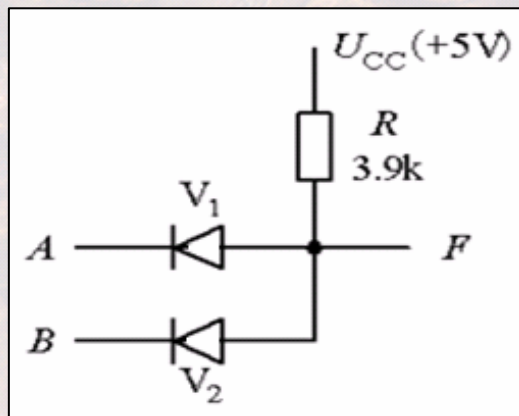
$$L = \text{not } A$$

用逻辑实现加法，用硬件实现加法？

(3)用基本电子元件实现基本逻辑运算？

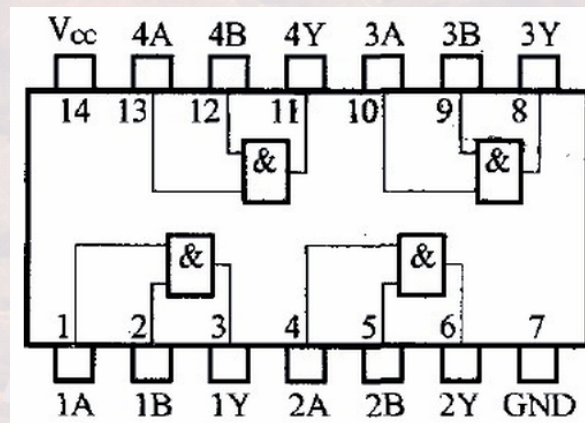
用二极管、三极管可实现基本的集成电路: 与门、或门和非门

◆ 这些电路被封装成集成电路(芯片)，即所谓的门电路。



“与” 门电路

$$F = A \text{ and } B$$

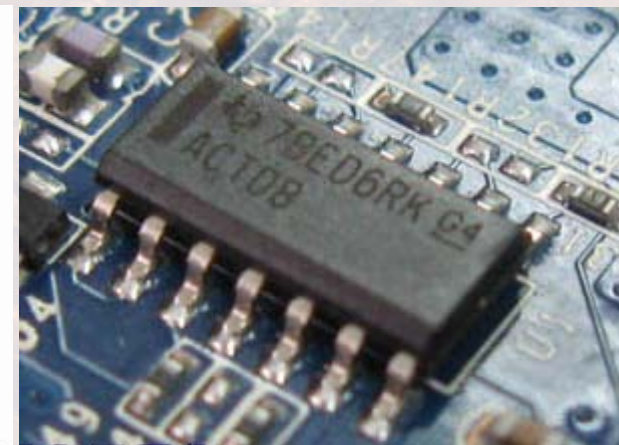
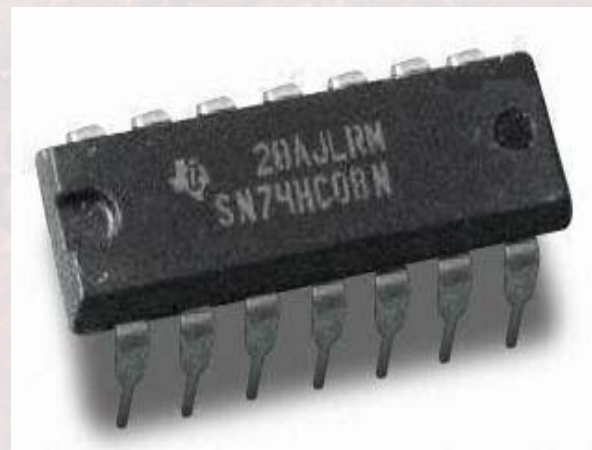


$$1Y = 1A \text{ and } 1B$$

$$2Y = 2A \text{ and } 2B$$

$$3Y = 3A \text{ and } 3B$$

$$4Y = 4A \text{ and } 4B$$

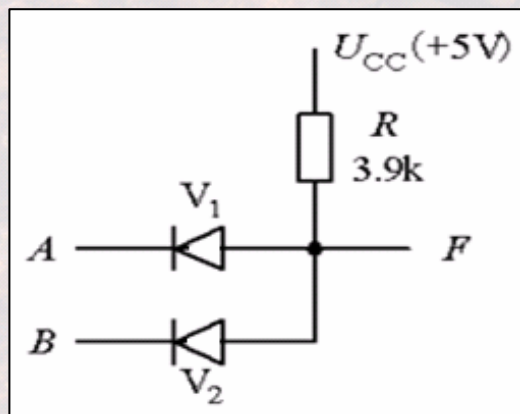


用逻辑实现加法，用硬件实现加法？

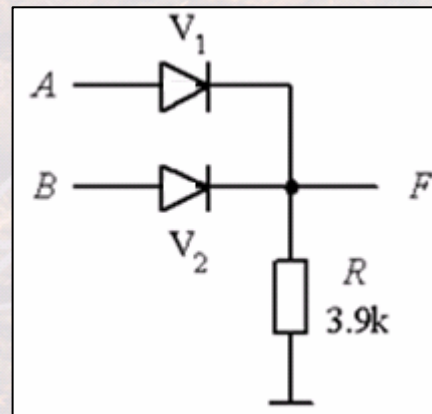
(3)用基本电子元件实现基本逻辑运算？

用二极管、三极管可实现基本的集成电路: 与门、或门和非门

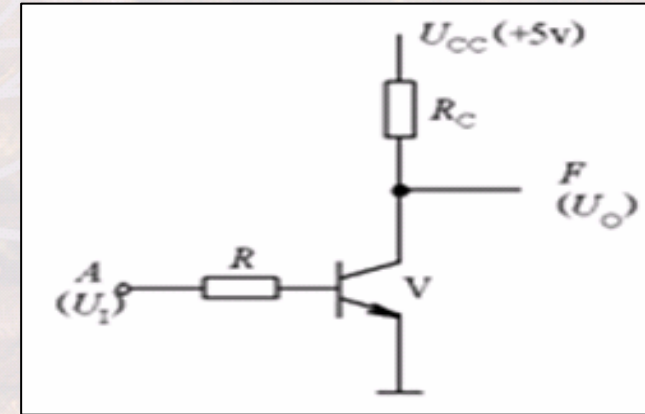
◆这些电路被封装成集成电路(芯片)，即所谓的门电路。



“与” 门电路
 $F = A \text{ and } B$



“或” 门电路
 $F = A \text{ or } B$



“非” 门电路
 $F = \text{not } A$

用逻辑实现加法，用硬件实现加法？

(4) 基本门电路的符号抽象？

基本门电路的符号表示及其特性

◆**与门电路**：是实现逻辑与运算的集成电路，即：只有当两个输入端为高电平(1)时，则输出端为高电平(1)；否则，输出端为低电平(0)。

◆**或门电路**：是实现逻辑或运算的集成电路，即：只有当两个输入端为低电平(0)时，则输出端为低电平(0)；否则，输出端为高电平(1)。

◆**非门电路**：是实现逻辑非运算的集成电路，即：当输入端为高电平(1)时，则输出端为低电平(0)；输入端为低电平(0)时，则输出端为高电平(1)。

◆**异或门电路**：是实现逻辑异或运算的集成电路，即：当两个输入端同为高电平(1)或同为低电平(0)时，则输出端为低电平(0)；否则，输出端为高电平(1)。



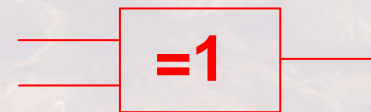
与门电路符号



或门电路符号



非门电路符号



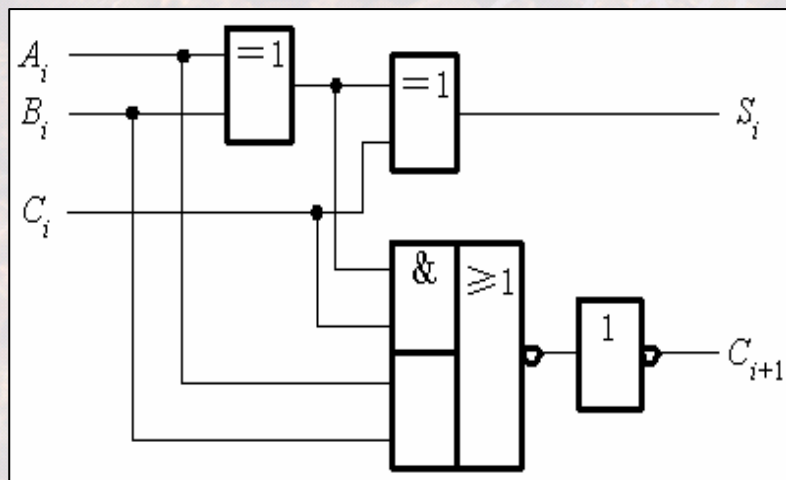
异或门电路符号

用逻辑实现加法，用硬件实现加法？

(5) 如何用已实现的基本逻辑运算(门电路)来实现更复杂的运算？

基于门电路的复杂组合逻辑电路

◆ 示例1：一位加法器的示例。

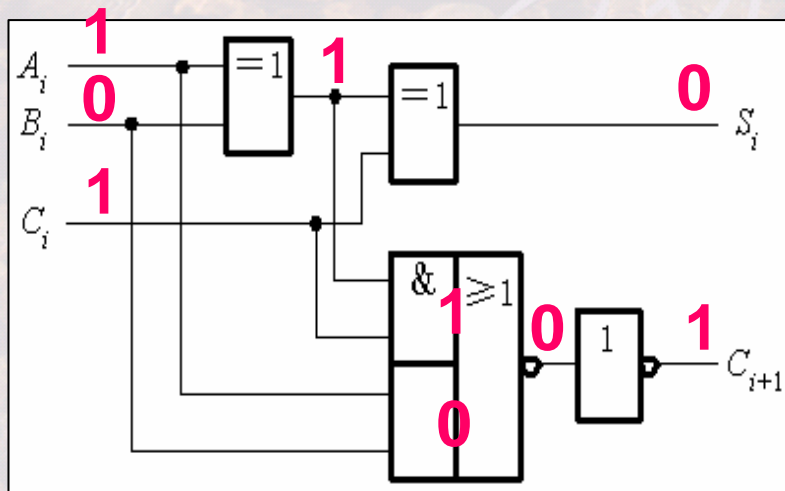


用逻辑实现加法，用硬件实现加法？

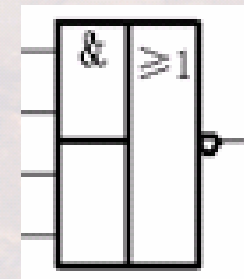
(5) 如何用已实现的基本逻辑运算(门电路)来实现更复杂的运算？

基于门电路的复杂组合逻辑电路

◆可验证一位加法器实现的正确性。



$$\begin{array}{r} A_i \\ B_i \\ C_i \\ + \\ \hline C_{i+1} \quad S_i \end{array}$$



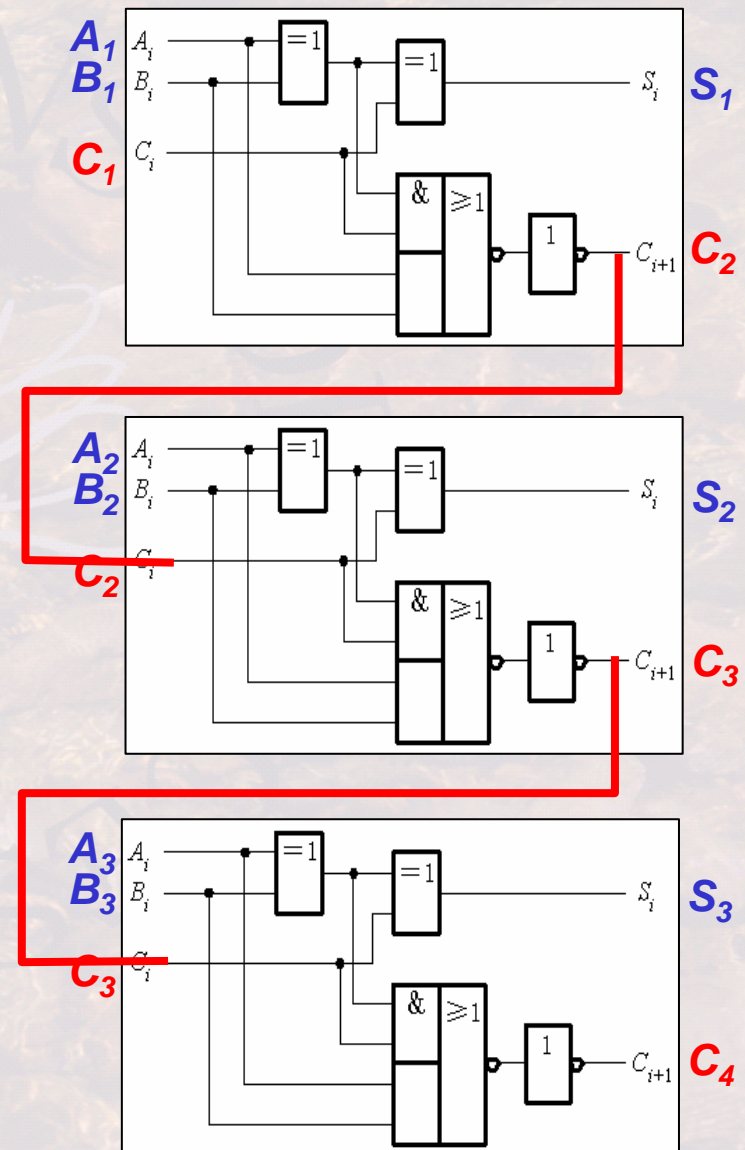
用逻辑实现加法，用硬件实现加法？

(5) 如何用已实现的基本逻辑运算(门电路)来实现更复杂的运算?

基于门电路的复杂组合逻辑电路

- ◆ 示例：多位加法器的实现
- ◆ 用已验证正确的一位加法器，来实现更为复杂的多位加法器
- ◆ 用已验证正确的多位加法器，来实现更为复杂的乘法器/除法等(略)
- ◆ 分层构造：低层电路已验证正确，可被封装起来；用已封装的已验证的低层电路可构造更为复杂的高层电路；如此一层层构造。

$$\begin{array}{r}
 A_3 \ A_2 \ A_1 \\
 + \ B_3 \ B_2 \ B_1 \\
 \hline
 C_4 \ C_3 \ C_2 \ C_1 \\
 S_3 \ S_2 \ S_1
 \end{array}
 \qquad
 \begin{array}{r}
 0 \ 0 \ 1 \\
 + \ 0 \ 1 \ 1 \\
 \hline
 1 \ 1 \ 0 \\
 1 \ 0 \ 0
 \end{array}$$



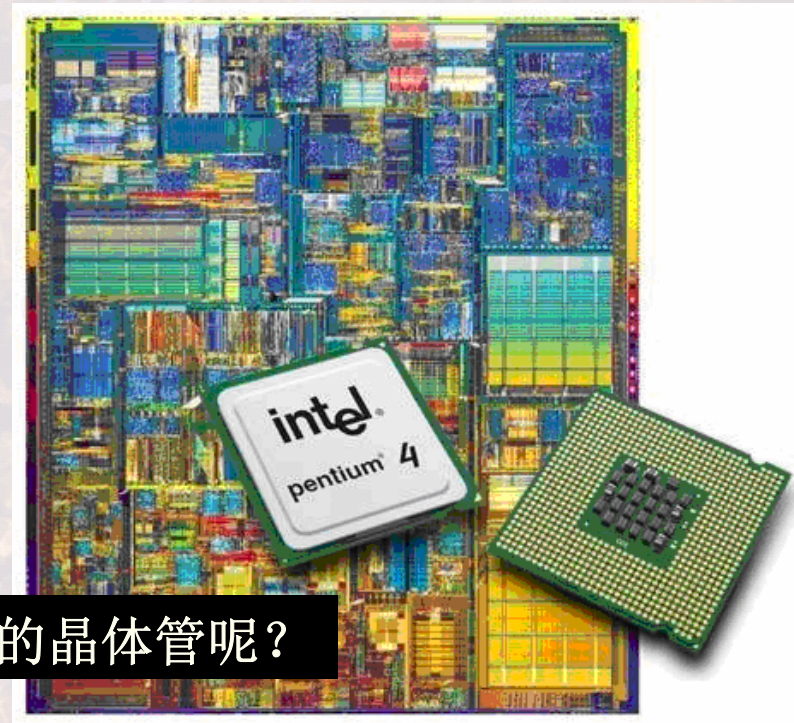
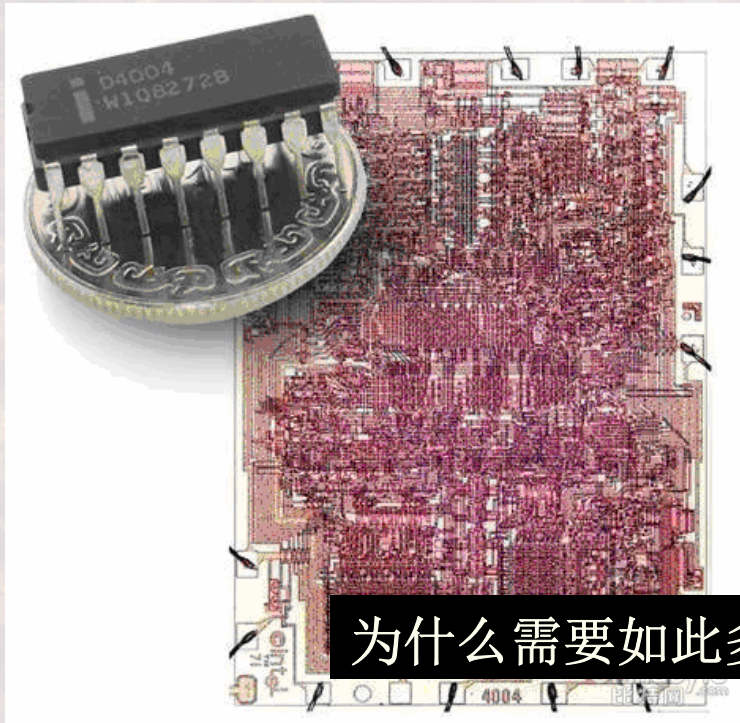
用逻辑实现加法，用硬件实现加法？

(5) 如何用已实现的基本逻辑运算(门电路)来实现更复杂的运算？

复杂部件的硬件实现(芯片、主板)

◆ **微处理器芯片**即是复杂组合逻辑集成在一块板上并封装而成的电路：

从Intel4004在**12平方毫米**的芯片上集成了**2250颗**晶体管→到Pentium 4处理器内建了**4200万颗**晶体管，以及采用**0.18微米**的电路→再到英特尔的**45纳米**Core 2至尊/至强四核处理器上装载了**8.2亿颗**晶体管。



为什么需要如此多的晶体管呢？

符号化-计算化-自动化

0和1的思维总结

