王道考研——组成原理

WWW.CSKAOYAN.COM

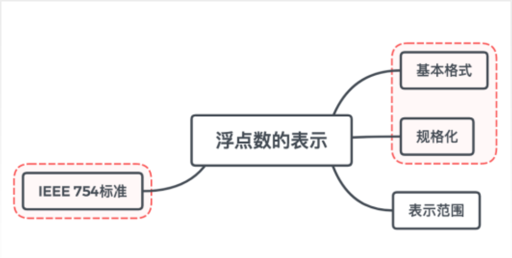
第二章 数据的表示和运算

本节内容

浮点数的表示与运算

表示

本节总览



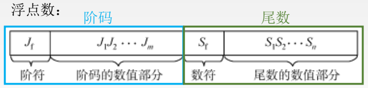
浮点数的表示

r 进制：$K_n K_{n-1} \ldots K_2 K_1 K_0 K_{-1} K_{-2} \ldots K_{-m}$

$= K_n \times r^n + K_{n-1} \times r^{n-1} + \cdots + K_2 \times r^2 + K_1 \times r^1 + K_0 \times r^0 + K_{-1} \times r^{-1} + K_{-2} \times r^{-2} + \ldots + K_{-m} \times r^{-m}$

定点数：如纯小数0.1011和纯整数11110

浮点数：



阶码：常用补码或移码表示

尾数：常用原码或补码表示

浮点数的真值：$N = \boxed{r}^E \times M$

阶码的底，通常为2

十进制：299792458m/s = 2.998×10^8m/s

阶码E反映浮点数的**表示范围**及小数点的实际位置；
尾数M的数值部分的位数n反映浮点数的**精度**。

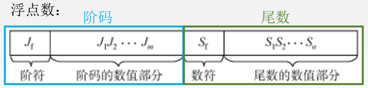
浮点数的表示

r 进制： $K_n K_{n-1} \ldots K_2 K_1 K_0 K_{-1} K_{-2} \ldots K_{-m}$

$= K_n \times r^n + K_{n-1} \times r^{n-1} + \cdots + K_2 \times r^2 + K_1 \times r^1 + K_0 \times r^0 + K_{-1} \times r^{-1} + K_{-2} \times r^{-2} + \ldots + K_{-m} \times r^{-m}$

定点数：如纯小数0.1011和纯整数11110

浮点数：



浮点数的真值： $N = \boxed{r}^E \times M$

阶码的底，通常为2

阶码E反映浮点数的**表示范围**及小数点的实际位置；
尾数M的数值部分的位数n反映浮点数的**精度**。

阶码：常用补码或移码表示

尾数：常用原码或补码表示

例：阶码、尾数均用补码表示，求a、b的真值
a = 0,01;1.1001
b = 0,01;0.01001

a： 阶码0,01对应真值+1
尾数1.1001对应真值-0.0111 = $-(2^{-2}+2^{-3}+2^{-4})$

或者理解为-111右移4位：$-\frac{7}{2^4} = -\frac{7}{16}$

所以a = $2^1 \times (-0.0111) = 2^1 \times (-\frac{7}{16}) = -\frac{7}{8}$

1B的存储空间

0	0	1	1	1	0	0	1

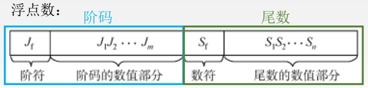
浮点数的表示

r 进制： $K_n K_{n-1} \ldots K_2 K_1 K_0 K_{-1} K_{-2} \ldots K_{-m}$

$= K_n \times r^n + K_{n-1} \times r^{n-1} + \cdots + K_2 \times r^2 + K_1 \times r^1 + K_0 \times r^0 + K_{-1} \times r^{-1} + K_{-2} \times r^{-2} + \ldots + K_{-m} \times r^{-m}$

定点数：如纯小数0.1011和纯整数11110

浮点数：



浮点数的真值： $N = \boxed{r}^E \times M$

阶码的底，通常为2

阶码E反映浮点数的**表示范围**及小数点的实际位置；
尾数M的数值部分的位数n反映浮点数的**精度**。

阶码：常用补码或移码表示

尾数：常用原码或补码表示

例：阶码、尾数均用补码表示，求a、b的真值
a = 0,01;1.1001
b = 0,01;0.01001

b： 阶码0,01对应真值+1
尾数0.01001对应真值+0.01001 = $+(2^{-2}+2^{-5})$

或者理解为+1001右移5位：$+\frac{9}{2^5} = +\frac{9}{32}$

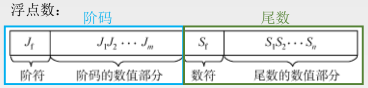
所以b = $2^1 \times (+0.01001) = 2^1 \times \frac{9}{32} = \frac{9}{16}$

1B的存储空间

0	0	1	0	0	1	0	0

浮点数的规格化

浮点数：



浮点数的真值： $N = \boxed{r}^E \times M$

阶码的底，通常为2

阶码E反映浮点数的**表示范围**及小数点的实际位置；
尾数M的数值部分的位数n反映浮点数的**精度**。

阶码：常用补码或移码表示

尾数：常用原码或补码表示

例：阶码、尾数均用补码表示，求a、b的真值
a = 0,01;1.1001
b = 0,01;0.01001

b： 阶码0,01对应真值+1
尾数0.01001对应真值+0.01001 = $+(2^{-2}+2^{-5})$

或者理解为+1001右移5位：$+\frac{9}{2^5} = +\frac{9}{32}$

所以b = $2^1 \times (+0.01001) = 2^1 \times \frac{9}{32} = \frac{9}{16}$

1B的存储空间

0	0	1	0	0	1	0	0

b = $2^1 \times (+0.01001)$
= $2^2 \times (+0.10010)$

0	1	0	0	1	0	0	1

浮点数的规格化

规格化：规定尾数的最高数位必须是一个有效值 。

左规：当浮点数运算的结果为非规格化时要进行规格化处理，
将尾数左移一位，阶码减1（基数为2时）。

右规：当浮点数运算的结果尾数出现溢出（双符号位为01或10）时，
将尾数右移一位，阶码加1（基数为2时）。

例：a = 010;00.1100，b = 010;00.1000，求a+b

a = $2^2 \times 00.1100$ ，b = $2^2 \times 00.1000$
a+b = $2^2 \times 00.1100 + 2^2 \times 00.1000$
= $2^2 \times (00.1100 + 00.1000)$
= $2^2 \times 01.0100$
= $2^3 \times 00.1010$

0	1	1	0	1	0	1	0

规格化浮点数的尾数M的绝对值应满足：$1/r \le |M| \le 1$

如果r=2，则有$1/2 \le |M| \le 1$

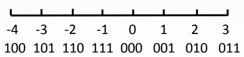
规格化浮点数的特点

规格化浮点数的尾数M的绝对值应满足：$1/r \leq |M| \leq 1$

如果$r=2$，则有$1/2 \leq |M| \leq 1$

1. 原码规格化后：
正数为0.1××…×的形式，其最大值表示为0.11…1；最小值表示为0.10…0。
尾数的表示范围为$1/2 \leq M \leq (1-2^{-n})$。
负数为1.1××…×的形式，其最大值表示为1.10…0；最小值表示为1.11…1。
尾数的表示范围为$-(1-2^{-n}) \leq M \leq -1/2$。

2. 补码规格化后：
正数为0.1××…×的形式，其最大值表示为0.11…1；最小值表示为0.10…0。
尾数的表示范围为$1/2 \leq M \leq (1-2^{-n})$。
负数为1.0××…×的形式，其最大值表示为1.01…1；最小值表示为1.00…0。
尾数的表示范围为$-1 \leq M \leq -(1/2+2^{-n})$。



王道考研/CSKAOYAN.COM

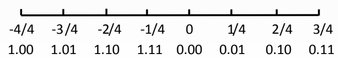
规格化浮点数的特点

规格化浮点数的尾数M的绝对值应满足：$1/r \leq |M| \leq 1$

如果$r=2$，则有$1/2 \leq |M| \leq 1$

1. 原码规格化后：
正数为0.1××…×的形式，其最大值表示为0.11…1；最小值表示为0.10…0。
尾数的表示范围为$1/2 \leq M \leq (1-2^{-n})$。
负数为1.1××…×的形式，其最大值表示为1.10…0；最小值表示为1.11…1。
尾数的表示范围为$-(1-2^{-n}) \leq M \leq -1/2$。

2. 补码规格化后：
正数为0.1××…×的形式，其最大值表示为0.11…1；最小值表示为0.10…0。
尾数的表示范围为$1/2 \leq M \leq (1-2^{-n})$。
负数为1.0××…×的形式，其最大值表示为1.01…1；最小值表示为1.00…0。
尾数的表示范围为$-1 \leq M \leq -(1/2+2^{-n})$。



王道考研/CSKAOYAN.COM

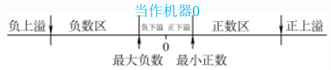
浮点数的溢出

规格化浮点数的尾数M的绝对值应满足：$1/r \leq |M| \leq 1$

如果$r=2$，则有$1/2 \leq |M| \leq 1$

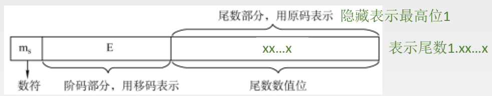
1. 原码规格化后：
正数为0.1××…×的形式，其最大值表示为0.11…1；最小值表示为0.10…0。
尾数的表示范围为$1/2 \leq M \leq (1-2^{-n})$。
负数为1.1××…×的形式，其最大值表示为1.10…0；最小值表示为1.11…1。
尾数的表示范围为$-(1-2^{-n}) \leq M \leq -1/2$。

2. 补码规格化后：
正数为0.1××…×的形式，其最大值表示为0.11…1；最小值表示为0.10…0。
尾数的表示范围为$1/2 \leq M \leq (1-2^{-n})$。
负数为1.0××…×的形式，其最大值表示为1.01…1；最小值表示为1.00…0。
尾数的表示范围为$-1 \leq M \leq -(1/2+2^{-n})$。



王道考研/CSKAOYAN.COM

IEEE 754标准



类 型	数 符	阶 码	尾数数值	总位数	偏置值	
					十六进制	十进制
短浮点数	1	8	23	32	7FH	127
长浮点数	1	11	52	64	3FFH	1023
临时浮点数	1	15	64	80	3FFFH	16383

1000 0001 1100 1010 0101 0000 1000 0000

1000 0001 1100 1010 0101 0000 1000 0000
0000 0000 0001 1111 0000 0000 0000 0000

规格化的短浮点数的真值为：$(-1)^s \times 1.M \times 2^{E-127}$
规格化长浮点数的真值为：$(-1)^s \times 1.M \times 2^{E-1023}$

王道考研/CSKAOYAN.COM

IEEE 754标准

类型	数符	阶码	尾数数值	总位数	偏置值	
					十六进制	十进制
短浮点数	1	8	23	32	7FH	127
长浮点数	1	11	52	64	3FFH	1023
临时浮点数	1	15	64	80	3FFFH	16383

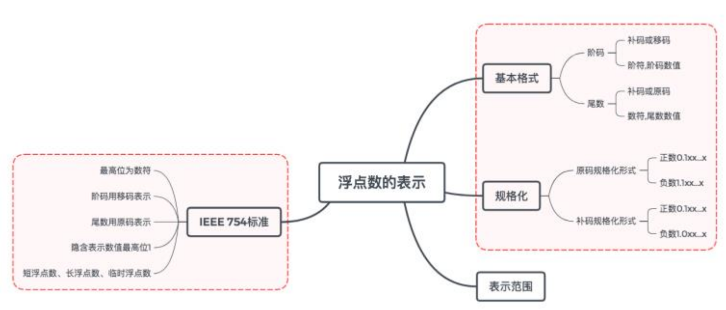
一些规定(短浮点数为例)：

1. E=0且M=0，则真值为0
2. E=0且M≠0，为非规格化数，真值 = $(-1)^s \times 0.M \times 2^{-126}$
3. 1≤E≤254时，真值 = $(-1)^s \times 1.M \times 2^{E-127}$
4. E=255且M≠0时，真值为‘NaN’（非数值）
5. E=255且M=0时，真值为正无穷或负无穷(看符号位)

格式	最小值	最大值
单精度	E=1，M=0：$1.0\times 2^{1-127}=2^{-126}$	E=254，M=.11...1：$1.11...1\times 2^{254-127}=2^{127}\times(2-2^{-23})$
双精度	E=1，M=0：$1.0\times 2^{1-1023}=2^{-1022}$	E=2046，M=.11...1：$1.11...1\times 2^{2046-1023}=2^{1023}\times(2-2^{-52})$

王道考研/CSKAOYAN.COM

本节回顾



王道考研/CSKAOYAN.COM

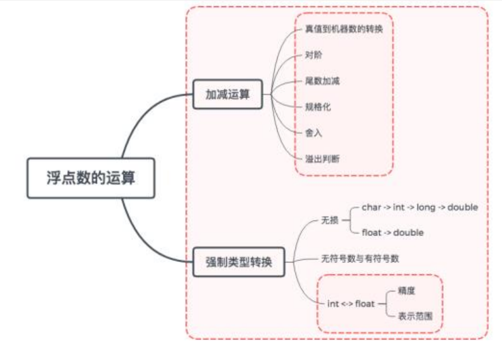
本节内容

浮点数的表示与运算

加减运算
强制类型转换

王道考研/CSKAOYAN.COM

本节总览



王道考研/CSKAOYAN.COM

浮点数的加减运算

浮点数加减运算步骤：
1. 对阶
2. 尾数加减
3. 规格化
4. 舍入
5. 判溢出

浮点数的加减运算

例：已知十进制数$X=-5/256$、$Y=+59/1024$，按机器补码浮点运算规则计算$X-Y$，结果用二进制表示，浮点数格式如下：阶符取2位，阶码取3位，数符取2位，尾数取9位

扩展：11.011000000
双符号位补码：11.011　双符号位补码：11011
补码：1.011　补码：1011

用补码表示阶码和尾数

0. 转换格式

5D = 101B，1/256 = 2^{-8} → X = - 101 × 2^{-8} = - 0.101 × 2^{-5} = - 0.101 × 2^{-101}

59D = 111011B，1/1024 = 2^{-10} → Y = + 111011 × 2^{-10} = + 0.111011 × 2^{-4} = + 0.111011 × 2^{-100}

X：11011,11.011000000　　Y：11100,00.111011000

浮点数加减运算步骤：
1. 对阶
2. 尾数加减
3. 规格化
4. 舍入
5. 判溢出

浮点数的加减运算

例：已知十进制数$X=-5/256$、$Y=+59/1024$，按机器补码浮点运算规则计算$X-Y$，结果用二进制表示，浮点数格式如下：阶符取2位，阶码取3位，数符取2位，尾数取9位

用补码表示阶码和尾数

0. 转换格式

5D = 101B，1/256 = 2^{-8} → X = - 101 × 2^{-8} = - 0.101 × 2^{-5} = - 0.101 × 2^{-101}

59D = 111011B，1/1024 = 2^{-10} → Y = + 111011 × 2^{-10} = + 0.111011 × 2^{-4} = + 0.111011 × 2^{-100}

X：11011,11.011000000　　Y：11100,00.111011000

浮点数加减运算步骤：
1. 对阶 使两个数的阶码相等，小阶向大阶看齐，尾数每右移一位，阶码加1
   ① 求阶差：$[\Delta E]_{补}$=11011+00100=11111，知$\Delta E=-1$
   ② 对阶：X：11011,11.011000000 → 11100,11. 101100000　　X = - 0.0101 × 2^{-100}
2. 尾数加减
3. 规格化
4. 舍入
5. 判溢出

浮点数的加减运算

例：已知十进制数$X=-5/256$、$Y=+59/1024$，按机器补码浮点运算规则计算$X-Y$，结果用二进制表示，浮点数格式如下：阶符取2位，阶码取3位，数符取2位，尾数取9位

用补码表示阶码和尾数

0. 转换格式

5D = 101B，1/256 = 2^{-8} → X = - 101 × 2^{-8} = - 0.101 × 2^{-5} = - 0.101 × 2^{-101}

59D = 111011B，1/1024 = 2^{-10} → Y = + 111011 × 2^{-10} = + 0.111011 × 2^{-4} = + 0.111011 × 2^{-100}

X：11011,11.011000000　　Y：11100,00.111011000

浮点数加减运算步骤：
1. 对阶 使两个数的阶码相等，小阶向大阶看齐，尾数每右移一位，阶码加1
   ① 求阶差：$[\Delta E]_{补}$=11011+00100=11111，知$\Delta E=-1$
   ② 对阶：X：11011,11.011000000 → 11100,11. 101100000　　X = - 0.0101 × 2^{-100}
2. 尾数加减　-Y：11100,11.000101000
   X-Y：11100, 10.110001000

$$
\begin{array}{r}
11.101100000 \\
+\ 11.000101000 \\
\hline
10.110001000
\end{array}
$$

X-Y
= (-0.0101 × 2^{-100})- (+0.111011 × 2^{-100})
= (-0.0101-0.111011) × 2^{-100}
= -1.001111 × 2^{-100}

3. 规格化
4. 舍入
5. 判溢出

浮点数的加减运算

例：已知十进制数$X=-5/256$、$Y=+59/1024$，按机器补码浮点运算规则计算$X-Y$，结果用二进制表示，浮点数格式如下：阶符取2位，阶码取3位，数符取2位，尾数取9位

用补码表示阶码和尾数

0. 转换格式

5D = 101B，1/256 = 2^{-8} → X = - 101 × 2^{-8} = - 0.101 × 2^{-5} = - 0.101 × 2^{-101}

59D = 111011B，1/1024 = 2^{-10} → Y = + 111011 × 2^{-10} = + 0.111011 × 2^{-4} = + 0.111011 × 2^{-100}

X：11011,11.011000000　　Y：11100,00.111011000

浮点数加减运算步骤：

1. 对阶 使两个数的阶码相等，小阶向大阶看齐，尾数每右移一位，阶码加1
   ① 求阶差：$[\Delta E]_{补}$=11011+00100=11111，知$\Delta E=-1$
   ② 对阶：X：11011,11.011000000 → 11100,11.101100000　　X = - 0.0101 × 2^{-100}
2. 尾数加减 -Y：11100,11.000101000
   X-Y：11100, 10.110001000

```
  11.101100000
+ 11.000101000
--------------
  10.110001000
```

X-Y
= (-0.0101 × 2^{-100})- (+0.111011 × 2^{-100})
= (-0.0101-0.111011) × 2^{-100}
= -1.001111 × 2^{-100}
= -0.1001111 × 2^{-011}

3. 规格化
   X-Y：11100, 10.110001000 → $\boxed{11}$101,11.011000100
4. 舍入　无舍入
5. 判溢出 常阶码，无溢出，结果真值为$2^{-3}\times(-0.1001111)_2$

王道考研/CSKAOYAN.COM

浮点数的加减运算-舍入

“0”舍“1”入法：类似于十进制数运算中的“四舍五入”法，即在尾数右移时，被移去的最高数值位为0，则舍去；被移去的最高数值位为1，则在尾数的末位加1。这样做可能会使尾数又溢出，此时需再做一次右规。

恒置“1”法：尾数右移时，不论丢掉的最高数值位是“1”还是“0”，都使右移后的尾数末位恒置“1”。这种方法同样有使尾数变大和变小的两种可能。

浮点数加减运算步骤：

1. 对阶
2. 尾数加减　如：加减结果为11100,10.110001011
3. 规格化　0舍1入：11100,10.110001011 → 11101,11.011000101 1
   → 11101,11.011000110 1
   恒置1:11100,10.110001011 → 11101,11.011000101 1
   → 11101,11.011000101 1
4. 舍入
5. 判溢出

王道考研/CSKAOYAN.COM

强制类型转换

类型	16位机器	32位机器	64位机器
char	8	8	8
short	16	16	16
int	16	32	32
long	32	32	64
long long	64	64	64
float	16	32	32
double	64	64	64

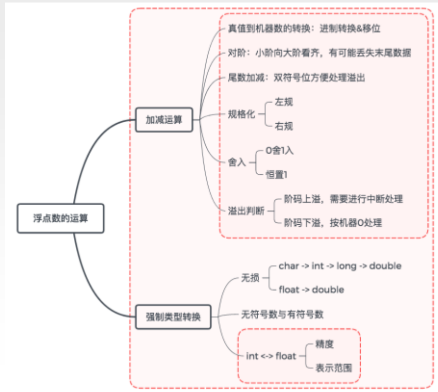
char → int → long → double
float → double
范围、精度从小到大，转换过程没有损失

int → float：可能损失精度
float → int：可能溢出及损失精度

32位
int：表示整数，范围 -2^{31} ～ $2^{31}-1$ ，有效数字32位
float：表示整数及小数，范围 $\pm[2^{-126}$ ～ $2^{127}\times(2-2^{-23})]$，有效数字23+1=24位

王道考研/CSKAOYAN.COM

本节回顾

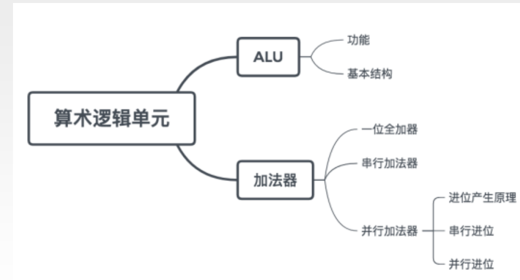


王道考研/CSKAOYAN.COM

本节内容

算术逻辑单元

本节总览



算术逻辑单元

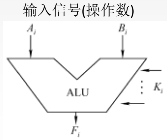
Arithmetic and Logic Unit

算术运算：加、减、乘、除等

逻辑运算：与、或、非、异或等

辅助功能：移位、求补等

输入信号(操作数)

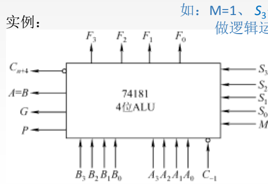


控制信号
(指令译码产生)

输出信号(运算结果)

实例：

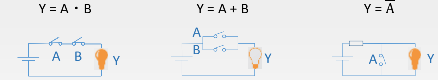
如：M=1、 $S_3 \sim S_0$=1001时，
做逻辑运算A⊕B



逻辑符号

	与	或	非
表达式	Y = A · B	Y = A + B	Y = $\bar{A}$

电路逻辑



真值表

与：

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

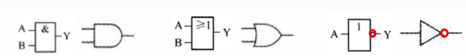
或：

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

非：

A	Y
0	0
1	1

电路符号



与门的天然逻辑：
“屏蔽”

复合逻辑

反演律：
$\overline{A+B}=\overline{A}\cdot\overline{B}$
$\overline{A\cdot B}=\overline{A}+\overline{B}$

	与非	半导体电路实现起来更方便	或非	异或
表达式	$Y=\overline{A\cdot B}$ $\overline{A}+\overline{B}$		$Y=\overline{A+B}$ $\overline{A}\cdot\overline{B}$	$Y=A\oplus B$

真值表

0 1

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

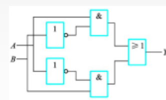
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

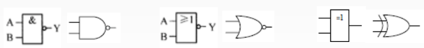
A和B不同

—> A=0且B=1或A=1且B=0

—> $\overline{A}\cdot B+A\cdot\overline{B}$



电路符号



与门的天然逻辑
“屏蔽”

异或的天然逻辑
“加法”
“奇偶”

王道考研/CSKAOYAN.COM

复合逻辑

反演律：
$\overline{A+B}=\overline{A}\cdot\overline{B}$
$\overline{A\cdot B}=\overline{A}+\overline{B}$

	与非	半导体电路实现起来更方便	或非	异或
表达式	$Y=\overline{A\cdot B}$ $\overline{A}+\overline{B}$		$Y=\overline{A+B}$ $\overline{A}\cdot\overline{B}$	$Y=A\oplus B$

真值表

0 1

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

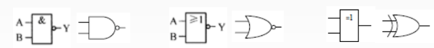
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

二进制加法

0 + 0 = 0 0
0 + 1 = 0 1
1 + 0 = 0 1
1 + 1 = 1 0

电路符号



与门的天然逻辑
“屏蔽”

异或的天然逻辑
“加法”
“奇偶”

王道考研/CSKAOYAN.COM

复合逻辑

反演律：
$\overline{A+B}=\overline{A}\cdot\overline{B}$
$\overline{A\cdot B}=\overline{A}+\overline{B}$

	与非	半导体电路实现起来更方便	或非	异或	同或
表达式	$Y=\overline{A\cdot B}$ $\overline{A}+\overline{B}$		$Y=\overline{A+B}$ $\overline{A}\cdot\overline{B}$	$Y=A\oplus B$	$Y=A\odot B$

真值表

0 1

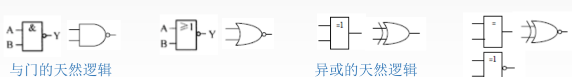
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

电路符号



与门的天然逻辑
“屏蔽”

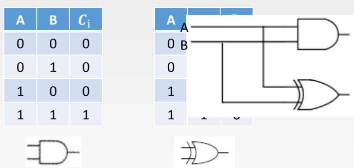
异或的天然逻辑
“加法”
“奇偶”

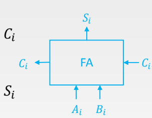
王道考研/CSKAOYAN.COM

组合逻辑电路设计-一位全加器

```
  100110...0110
+ 101100...1010
---------------
  ******...0000
```

A	B	C_i
0	0	0
0	1	0
1	0	0
1	1	1



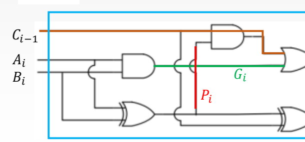


A_i
B_i
C_{i-1}

S_i：输入中有奇数个1时为1(异或)

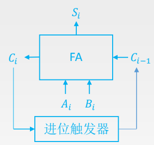
C_i：思路1. 输入中至少2个1
思路2. 进位来源
-产生(来自本级A_i和B_i)
-传递(来自前一级C_{i-1})

$C_i = G_i + P_i\ C_{i-1}$



王道考研/CSKAOYAN.COM

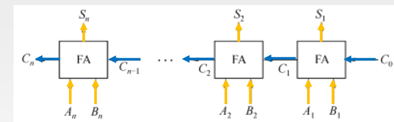
串行加法器



串行加法器：只有一个全加器，数据逐位串行送入加法器中进行运算。
进位触发器用来寄存进位信号，以便参与下一次运算。

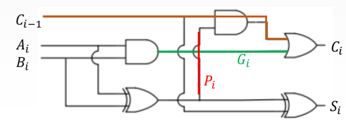
如果操作数长n位，加法就要分n次进行，每次产生一位和，并且串行逐位地送回寄存器。

并行加法器



串行进位的并行加法器：把n个全加器串接起来，就可进行两个n位数的相加。

串行进位又称为行波进位，每一级进位直接依赖于前一级的进位，即进位信号是逐级形成的。

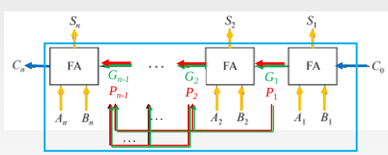


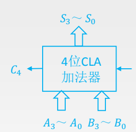
$C_i = G_i + P_i C_{i-1}$

$G_i = A_iB_i$ $P_i = A_i \oplus B_i$

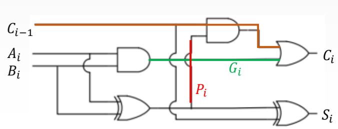
$C_1 = G_1 + P_1C_0$
$C_2 = G_2 + P_2C_1 = G_2 + P_2G_1 + P_2P_1C_0$
$C_3 = G_3 + P_3C_2 = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1C_0$

并行加法器





并行进位的并行加法器：各级进位信号同时形成，又称为先行进位、同时进位

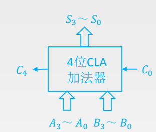


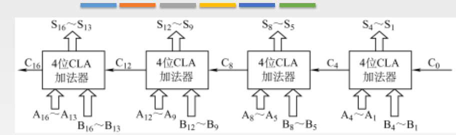
$C_i = G_i + P_i C_{i-1}$

$G_i = A_iB_i$ $P_i = A_i \oplus B_i$

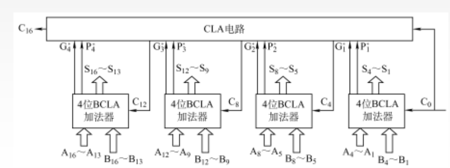
$C_1 = G_1 + P_1C_0$
$C_2 = G_2 + P_2C_1 = G_2 + P_2G_1 + P_2P_1C_0$
$C_3 = G_3 + P_3C_2 = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1C_0$

并行加法器



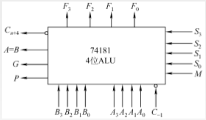


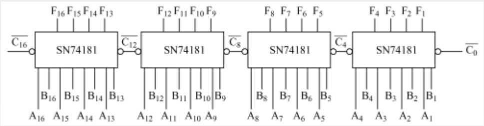
单级先行进位方式，又称为**组内并行、组间串行**进位方式。

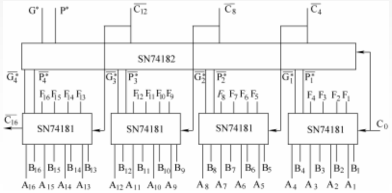


多级先行进位方式，又称为**组内并行、组间并行**进位方式

ALU芯片的组织

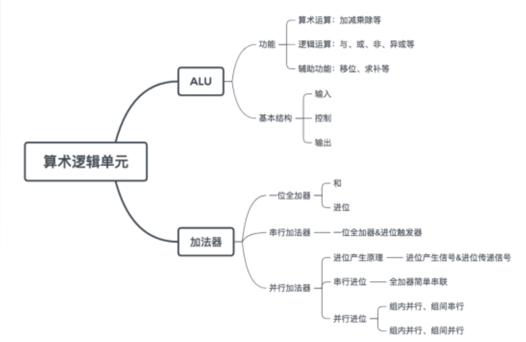






王道考研/CSKAOYAN.COM

本节总览



王道考研/CSKAOYAN.COM