

Pesquisa Sobre o uso de Operadores Binários em Linguagem C

Para Início das Tarefas de Iniciação Científica 2017/01
Engenharia de Controle e Automação

Autores: Mateus Braga e Matheus Cardoso

Orientador: Márcio Assunção

Professor: Lúcio Passos

4 de junho de 2017

1 Tabela de Operadores Binários

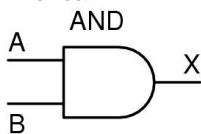
Device Features	
Operador	Nome
&	AND
	OR inclusivo
. ^	OR exclusivo
<<	Deslocamento à esquerda
>>	Deslocamento à direita
. ~	Complemento

2 Operador

2.1 And &

O operador And funciona comparando bit-a-bit de um número com outro, de forma que funciona semelhante a uma porta lógica And.

Porta AND:



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

Exemplo em Código:

C (gcc 4.8, C11) **EXPERIMENTAL!**
see [known bugs](#) and report to philip@pgbovine.net

```
1 int main() {  
2   int a=12;  
3   int b=25;  
4   int c;  
5   c = a&b;  
→ 6   return 0;  
→ 7 }
```

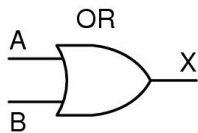
Stack

main	
a	int 12
b	int 25
c	int 8

2.2 OR inclusivo ||

O operador OR inclusivo funciona comparando bit-a-bit de um número com outro, de forma que se um dos bits comparados for 1 a saída já é automaticamente 1. Funciona como uma porta OR

Porta OR:



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

Exemplo em Código:

C (gcc 4.8, C11) **EXPERIMENTAL!**
see [known bugs](#) and report to philip@pgbovine.net

```
1 int main() {  
2   int a=12;  
3   int b=25;  
4   int c;  
5   c = a|b;  
→ 6   return 0;  
→ 7 }
```

[Edit code](#)

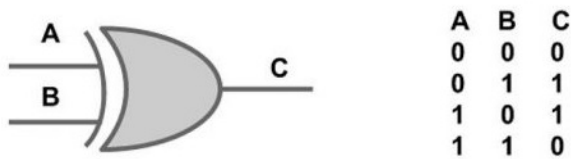
Stack

main	
a	int 12
b	int 25
c	int 29

2.3 OR exclusivo ||

O operador OR exclusivo funciona comparando bit-a-bit de um número com outro, de forma que se os bits comparados forem diferentes a saída será 1. Funciona como a porta XOR.

Porta XOR:



Exemplo em Código:

C (gcc 4.8, C11) **EXPERIMENTAL!**
see [known bugs](#) and report to philip@pgbovine.net

```
1 int main() {  
2 int a=12;  
3 int b=25;  
4 int c;  
5 c = a^b;  
→ 6 return 0;  
→ 7 }
```

[Edit code](#)

Stack

main	
a	int 12
b	int 25
c	int 21

2.4 Deslocamento a Esquerda <<

O operador Deslocamento a esquerda funciona deslocando todos os bits do número para esquerda e completando a última casa com zero. Funciona como uma multiplicação por 2.

Exemplo em Código:

C (gcc 4.8, C11) **EXPERIMENTAL!**
see [known bugs](#) and report to philip@pgbovine.net

```
1 int main() {  
2 int a=12;  
3 printf("%d",a<<1);  
→ 4 return 0;  
→ 5 }
```

[Edit code](#)

Print output (drag lower right corner to resize)

24

Stack

Heap

main	
a	int 12

2.5 Deslocamento a Direita >>

O operador Deslocamento a direita funciona deslocando todos os bits do número para direita e completando a primeira casa com zero. Funciona como uma divisão por 2.

Exemplo em Código:

C (gcc 4.8, C11) **EXPERIMENTAL!**
see [known bugs](#) and report to philip@pgbovine.net

```
1 int main() {  
2 int a=12;  
→ 3 printf("%d",a>>1);  
→ 4 return 0;  
5 }
```

[Edit code](#)

Print output (drag lower right corner to resize)

6

Stack

Heap

main	
a	int 12

2.6 Complemento

O operador complemento funciona realizando a técnica de inverter bit-a-bit de um número. Dessa forma, assim como o operador de deslocamento a direita e deslocamento a esquerda, atua sobre um único número. O seu resultado sempre será o número negativo com seu módulo acrescido de uma unidade (estará escrito na forma de complemento de 2). Aplicar essa operação duas vezes em um número sempre resultará nele mesmo.

Exemplo em Código:

C (gcc 4.8, C11) **EXPERIMENTAL!**
see [known bugs](#) and report to philip@pgbovine.net

```
1 int main() {  
2   int a=21;  
3   printf("%d\n", ~a);  
4   printf("%d", ~~a);  
5   return 0;  
6 }
```

[Edit code](#)

Print output (drag lower right corner to resize)

-22
21

Stack

Heap

main

a | int
21

Referências

- [1] <https://www.embarcados.com.br/xor/>
- [2] http://www.dpi.inpe.br/~carlos/Academicos/Cursos/ArqComp/aula_5bn1.html
- [3] <http://www.pythontutor.com/c.html#mode=edit>
- [4] https://en.wikipedia.org/wiki/Bitwise_operations_in_C
- [5] http://www.tutorialspoint.com/cprogramming/c_bitwise_operators.htm