Operaciones Aritméticas y Lógicas

IIC2343 - Arquitectura de Computadores

Nicolás Elliott B. (nicolas.elliott@uc.cl)

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
ESCUELA DE INGENIERÍA
PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE

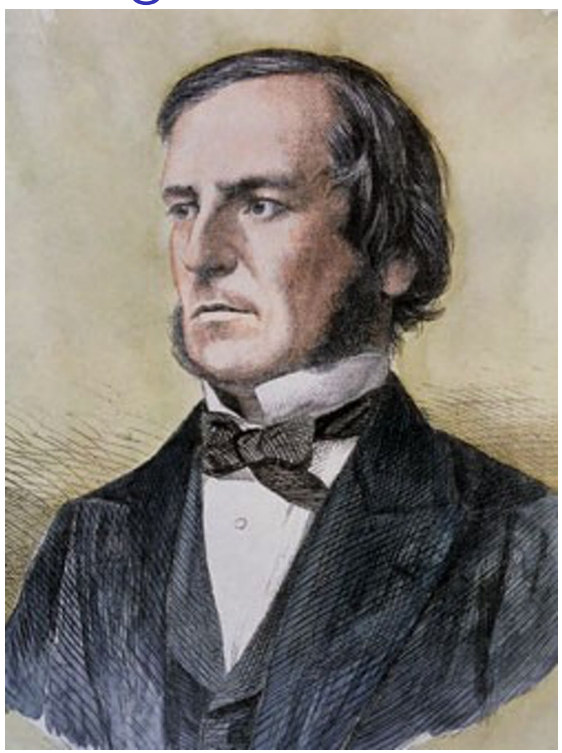
(II/2019)

Lógica Boole

George Boole

- 1815 - 1864
- Estudio matemática y la lógica
- Álgebra de Boole, usa solo variables del tipo V o F

George Boole



Álgebra de Boole

Operador No ($\neg$):

A	not(A)
F	V
V	F

Operador O ($\vee$):

A	B	A or B
F	F	F
F	V	V
V	F	V
V	V	V

Operador Y ($\wedge$):

A	B	A and B
F	F	F
F	V	F
V	F	F
V	V	V

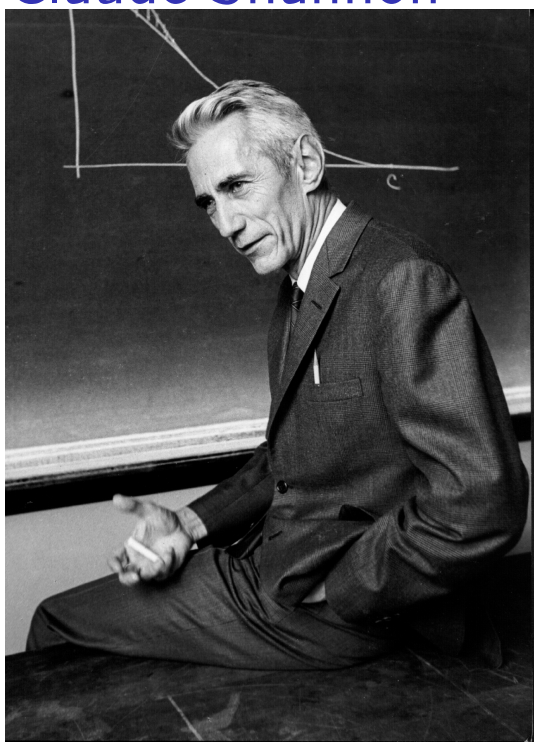
Operador O exclusivo ($\oplus$):

$$A \oplus B = (A \wedge \neg B) \vee (B \wedge \neg A)$$

Claude Shannon

- 1916 - 2001
- Aplica el álgebra de Boole al análisis y la síntesis de la conmutación y de los circuitos digitales

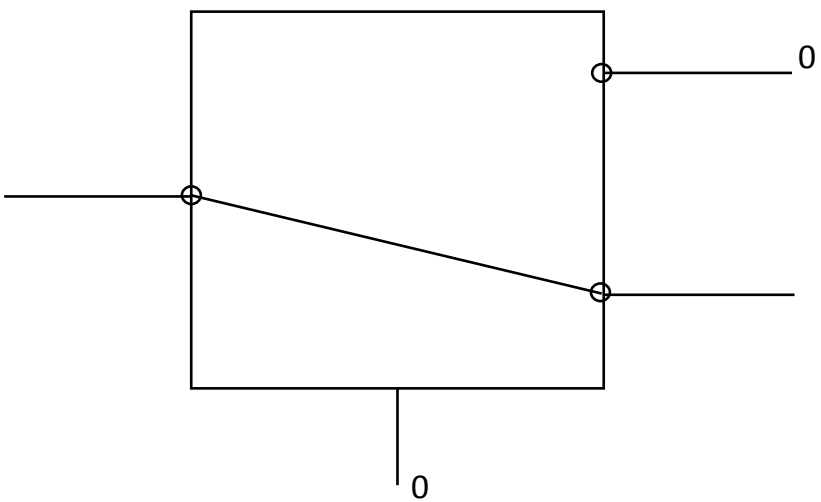
Claude Shannon



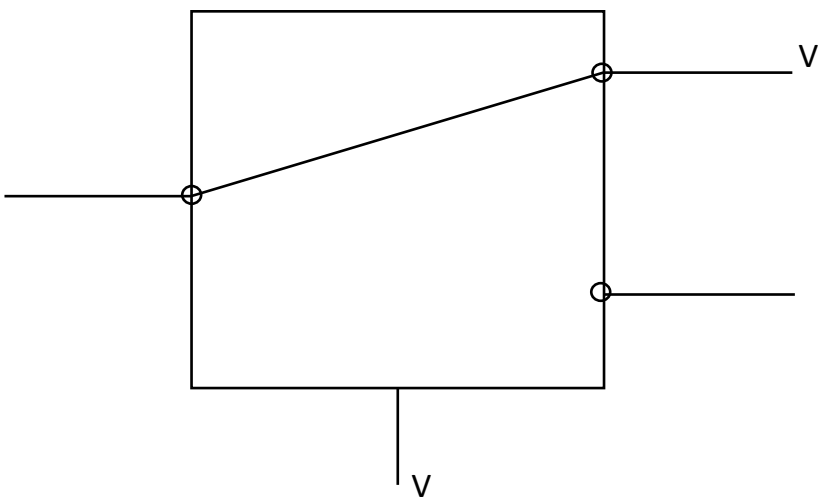
Implementación

Relé

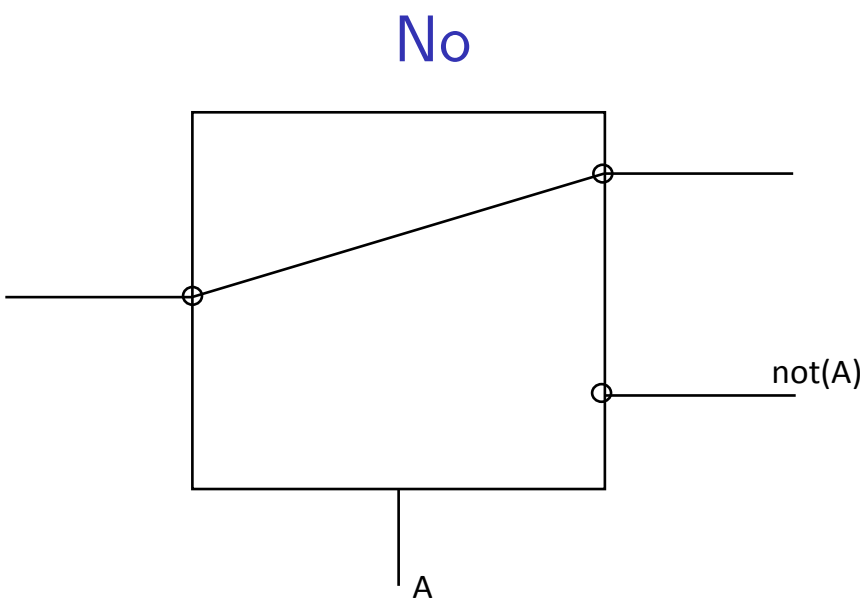
Abierto



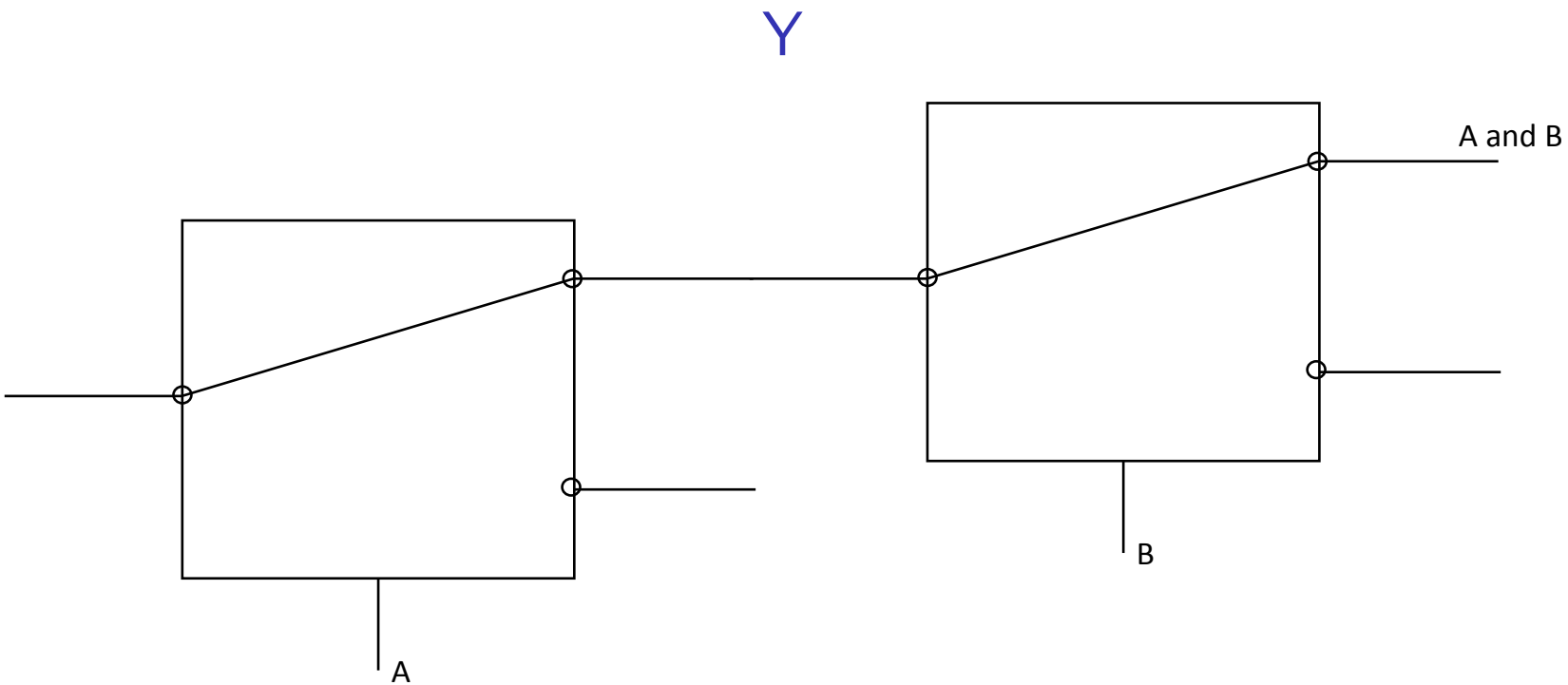
Cerrado



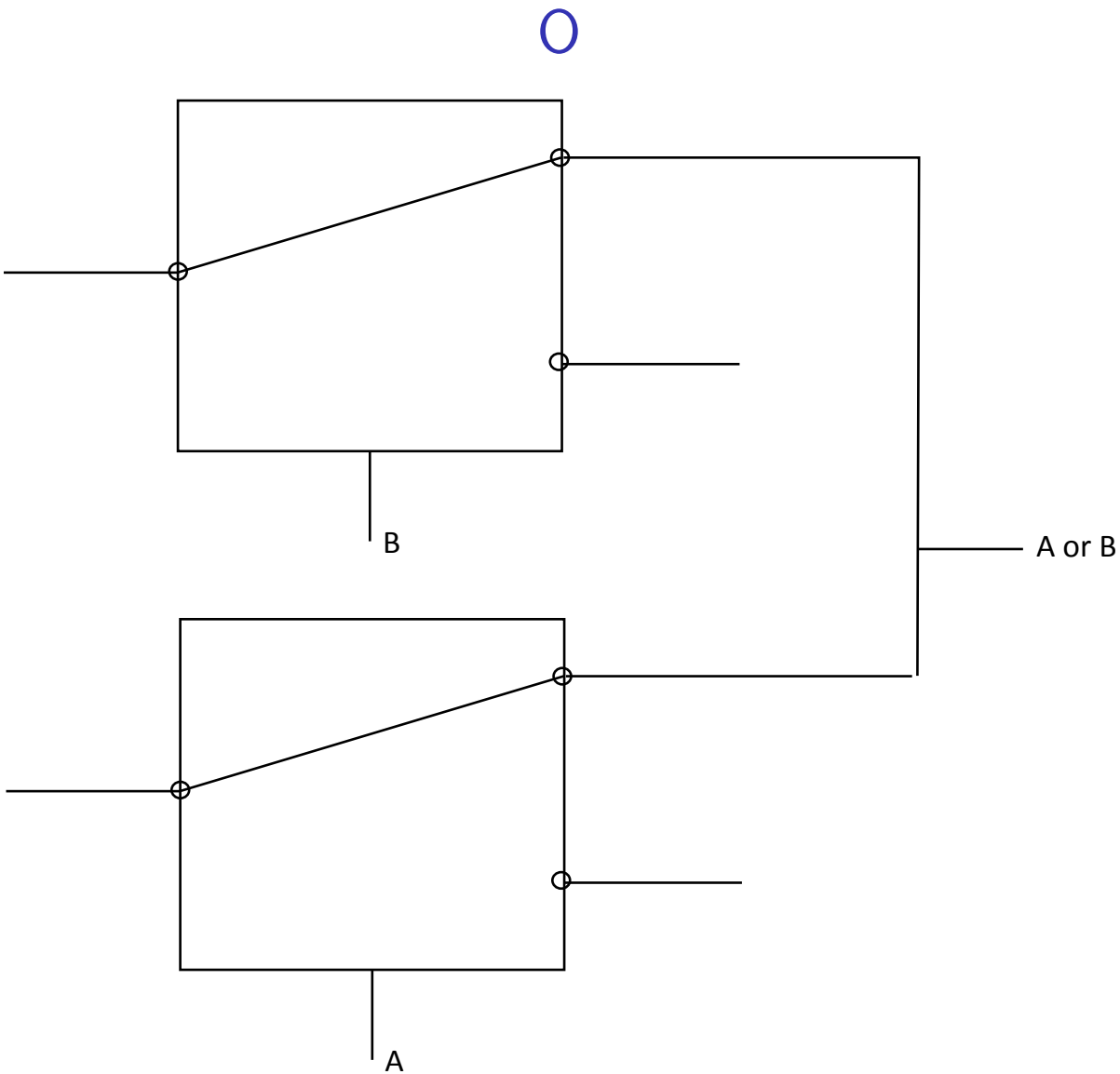
Implementación



Implementación



Implementación



Compuertas Lógicas

NOT

No ($\neg$):

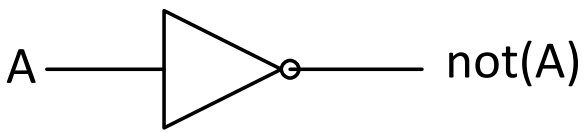


Tabla de valores:

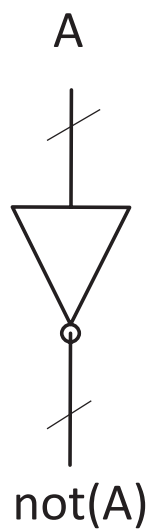
A	not(A)
0	1
1	0

C

```
not_a = ~a;
```

VHDL:

```
not_a <= not a;
```



AND

Y ($\wedge$):

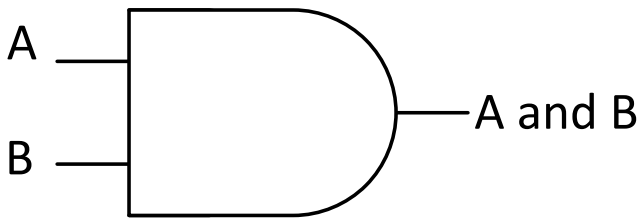


Tabla de valores:

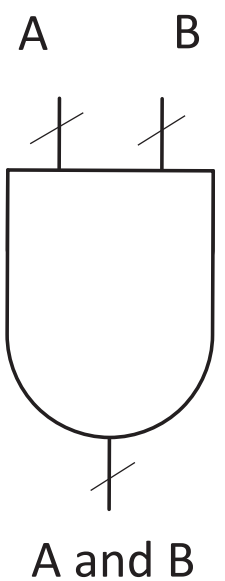
A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1

C

```
a_and_b = a & b;
```

VHDL:

```
a_and_b <= a and b;
```



OR

O ($\vee$):

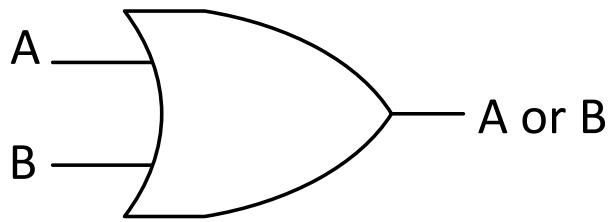


Tabla de valores:

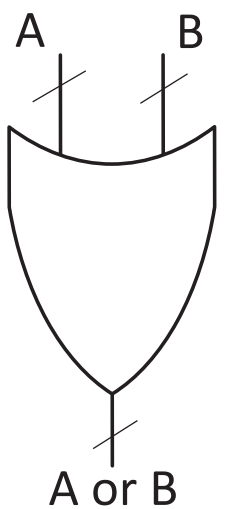
A	B	A or B
0	0	0
0	1	1
1	0	1
1	1	1

C

```
a_or_b = a | b;
```

VHDL:

```
a_or_b <= a or b;
```



XOR

O Exclusivo ($\oplus$):

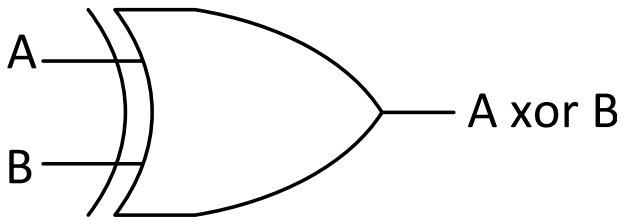


Tabla de valores:

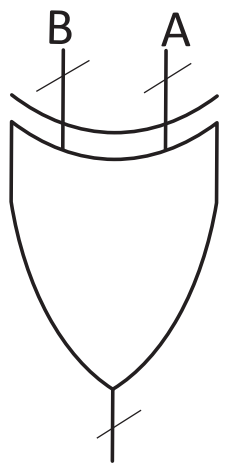
A	B	A xor B
0	0	0
0	1	1
1	0	1
1	1	0

C

```
a_xor_b = a ^ b;
```

VHDL:

```
a_xor_b <= a xor b;
```



A xor B

Sumador

Sumador de 1 Bit

HalfAdder

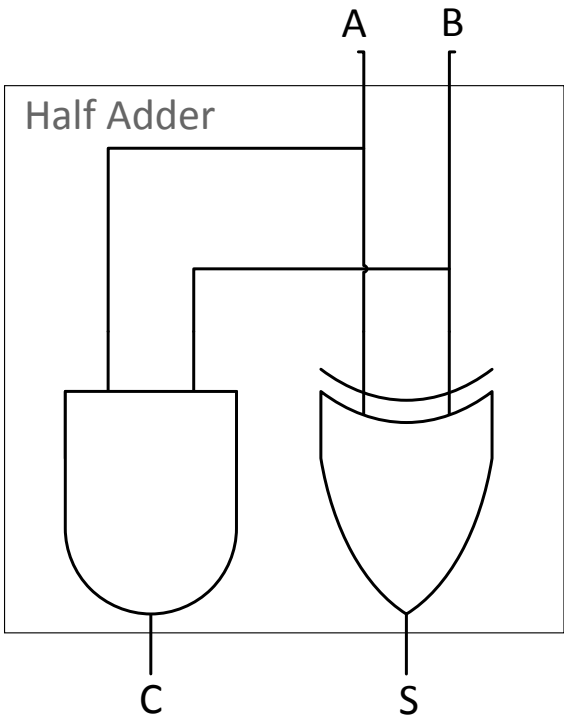


Tabla de valores

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Sumador de 1 Bit intermedio

FullAdder

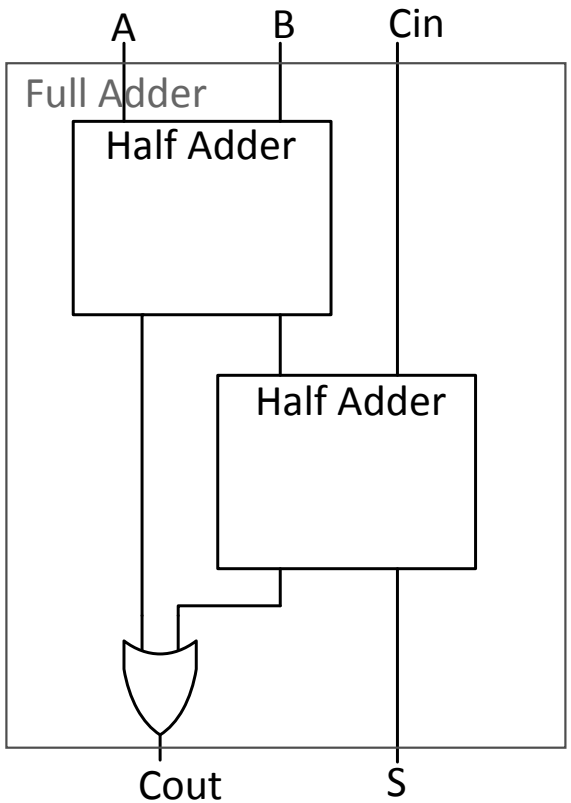
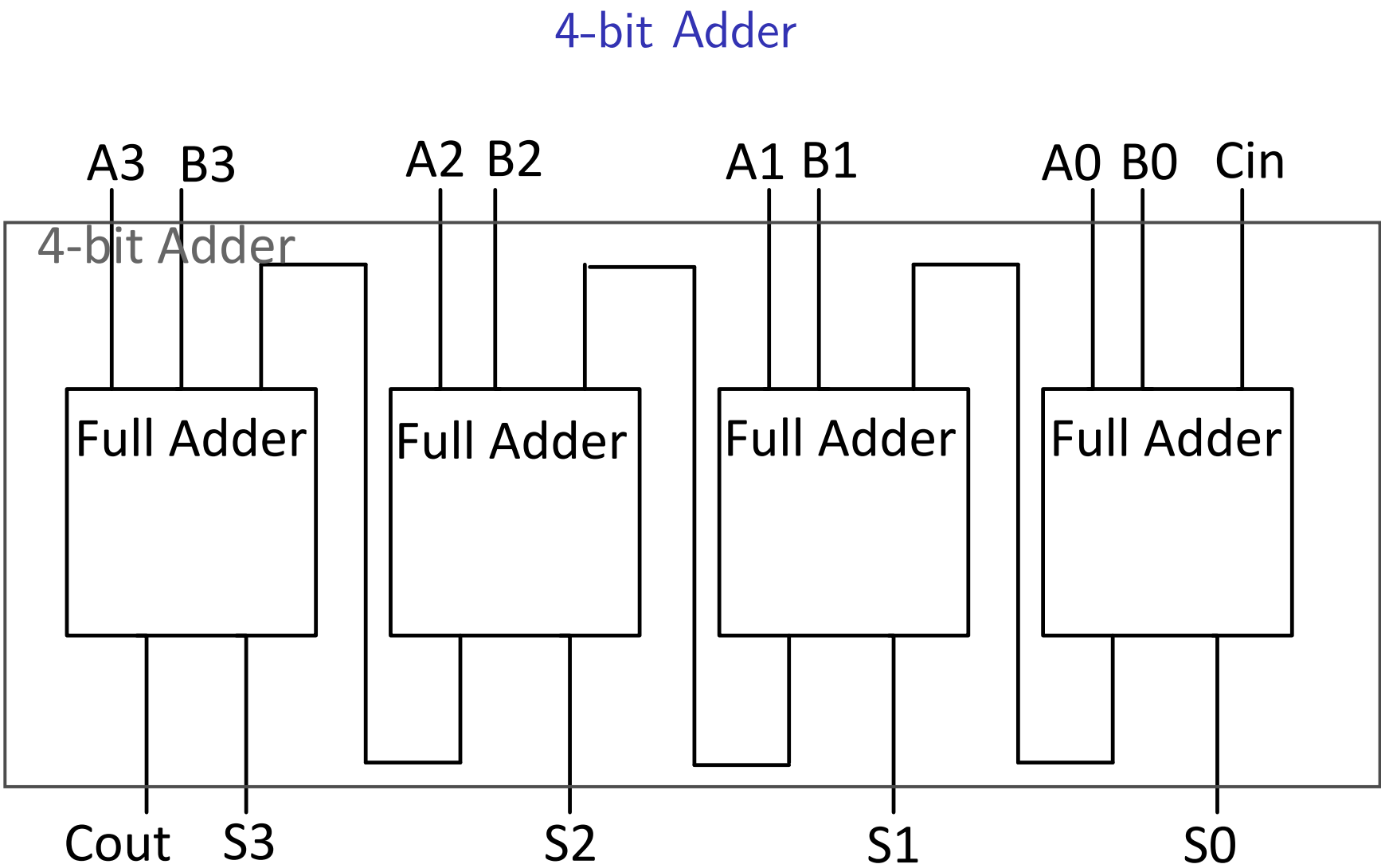


Tabla de valores

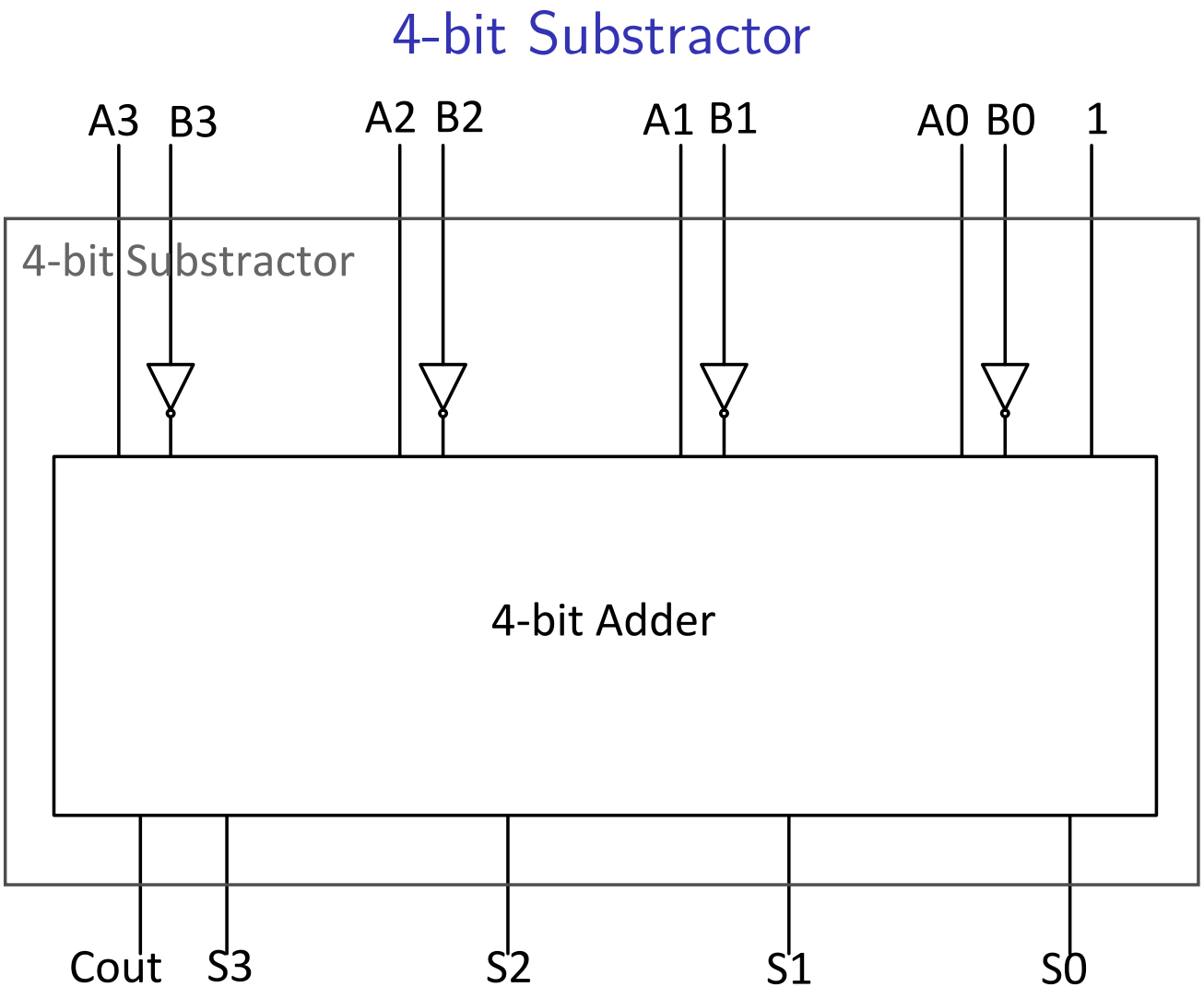
A	B	Cin	Cout	S
0	0	0	0	0
0	1	0	0	1
1	0	0	0	1
1	1	0	1	0
0	0	1	0	1
0	1	1	1	0
1	0	1	1	0
1	1	1	1	1

Sumador de 4 Bit



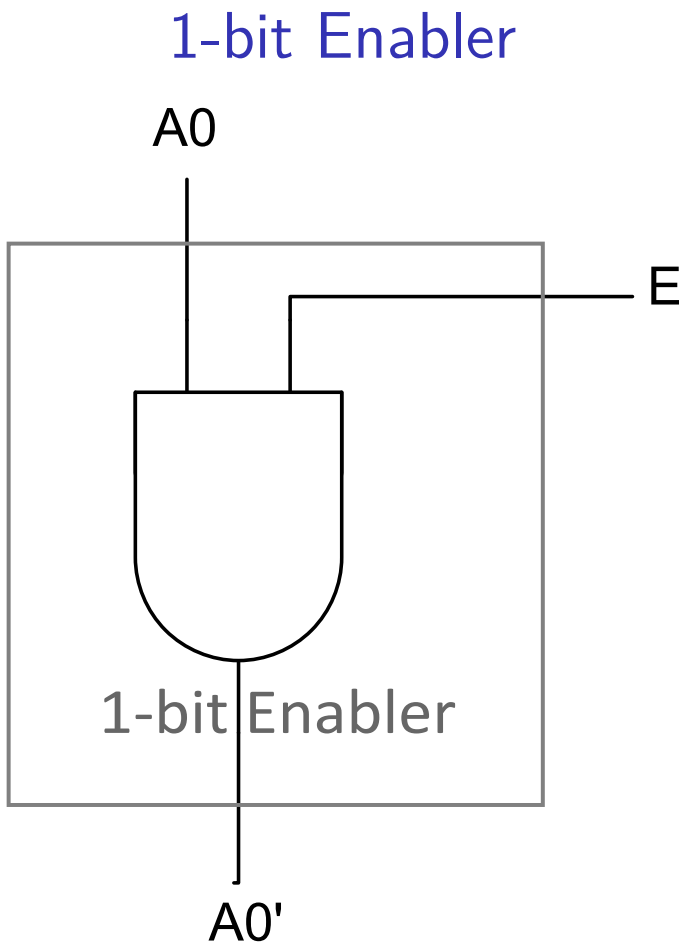
Restador

Sumador de 4 Bit

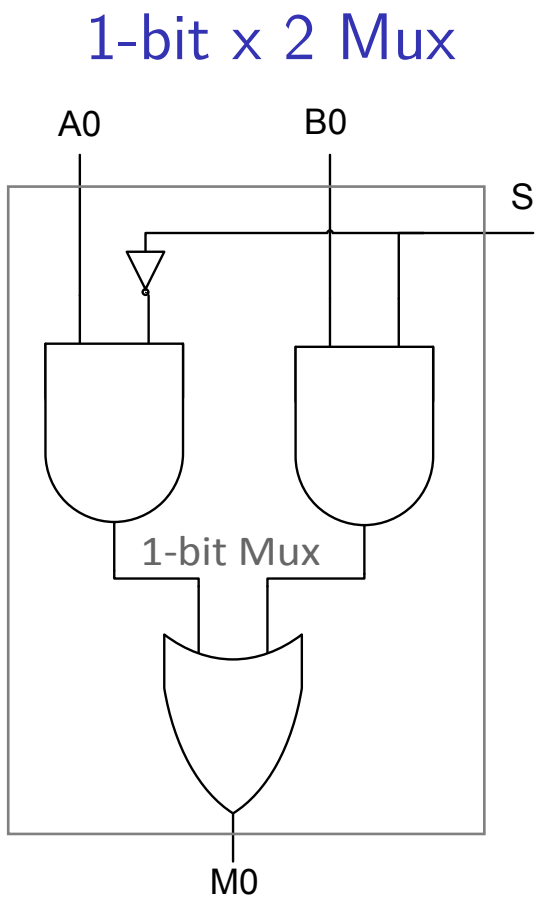


Multiplexor

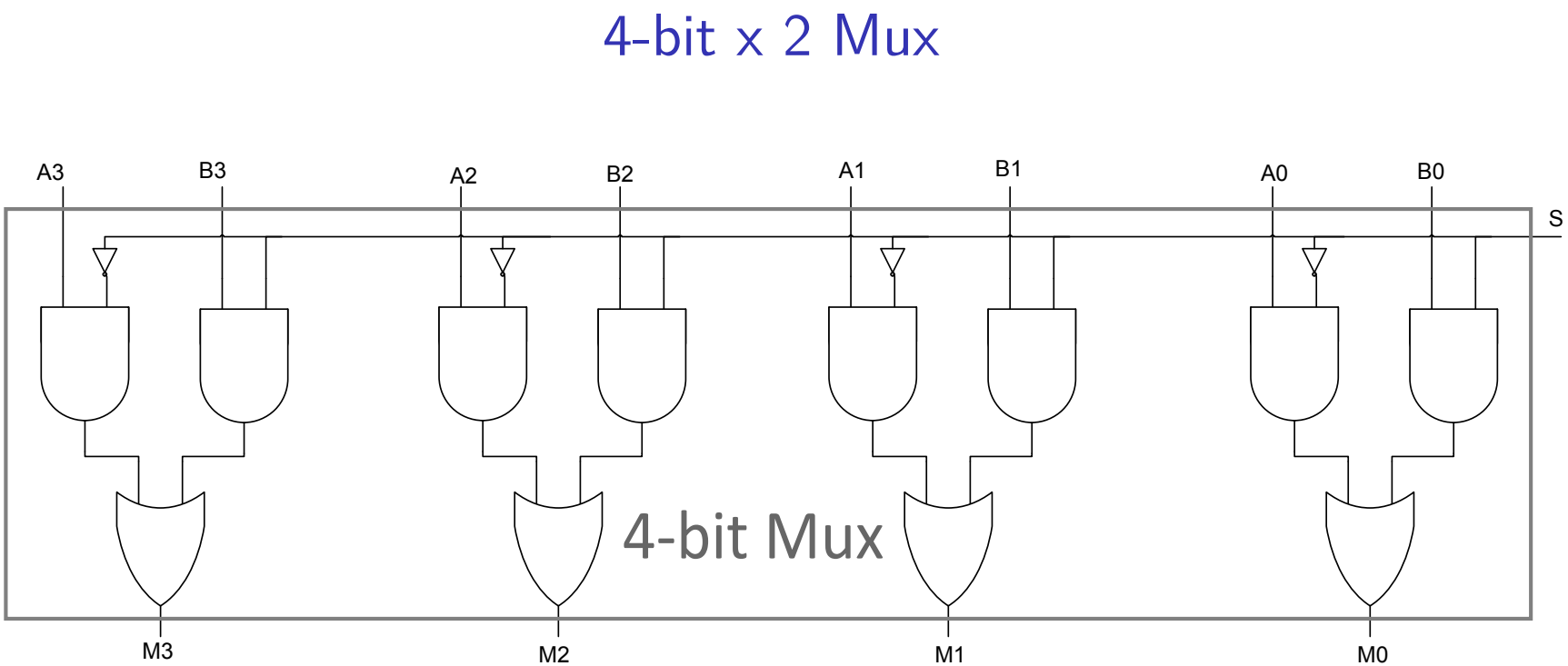
Enabler de 1 Bit



Multiplexor de 2 entradas de 1 Bit

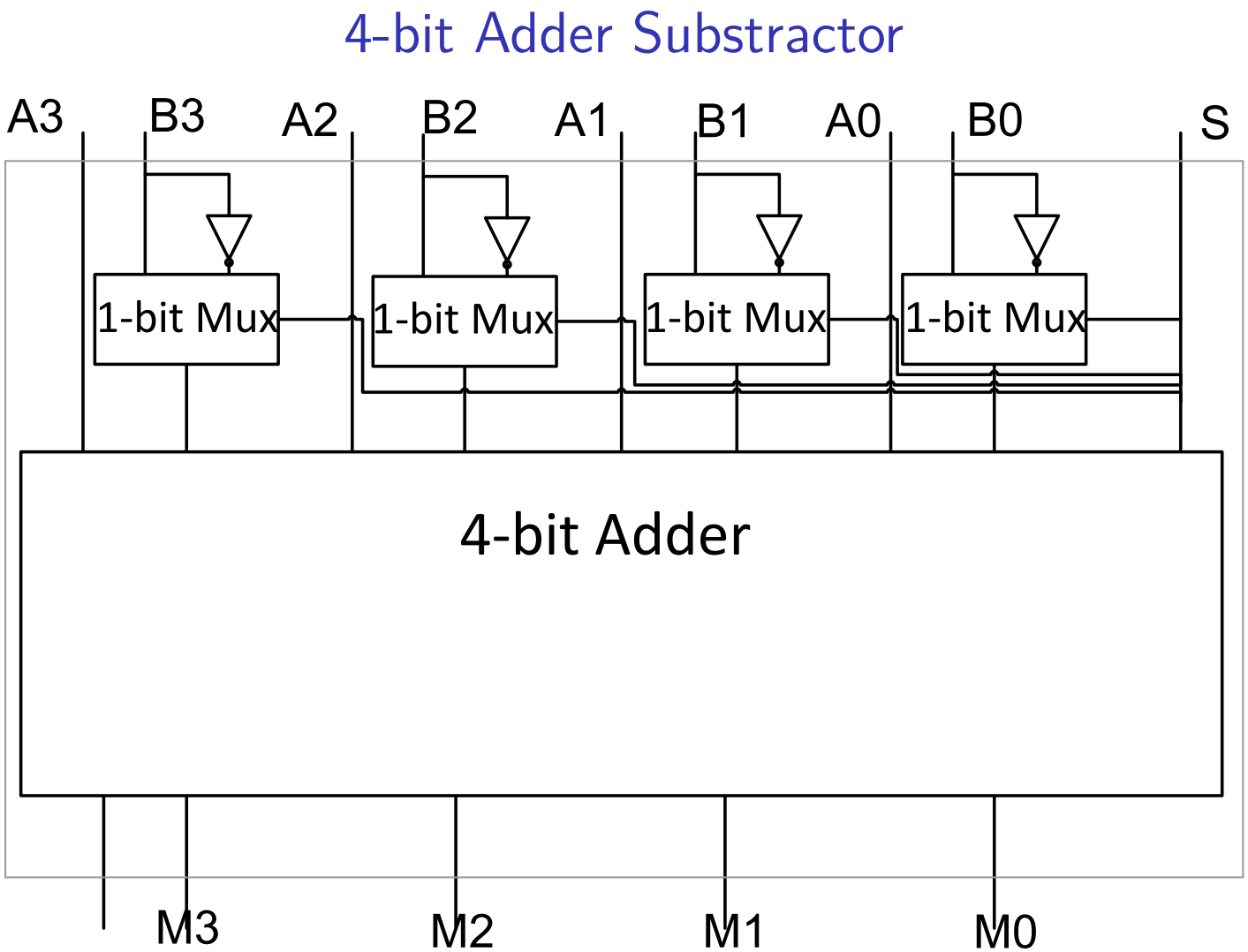


Multiplexor de 2 entradas de 4 Bit



Sumador - Restador

Sumador Restador de 4 Bit



Shifts

Operadores de Desplazamiento Lógicos

Shift Left:

shift left	0	0	1	1
	0	1	1	0

C

```
a_shl_1 = a << 1;
```

Shift Right:

shift right	0	0	1	1
	0	0	0	1

C

```
a_shr_1 = a >> 1;
```

1

[1] en C se aplica cuando el tipo de variable sobre el cual se opera no tiene signo, si lo tiene, entonces se hará un desplazamiento aritmético, que conservará el signo

ALU

Unidad Aritmético-Lógica

ALU

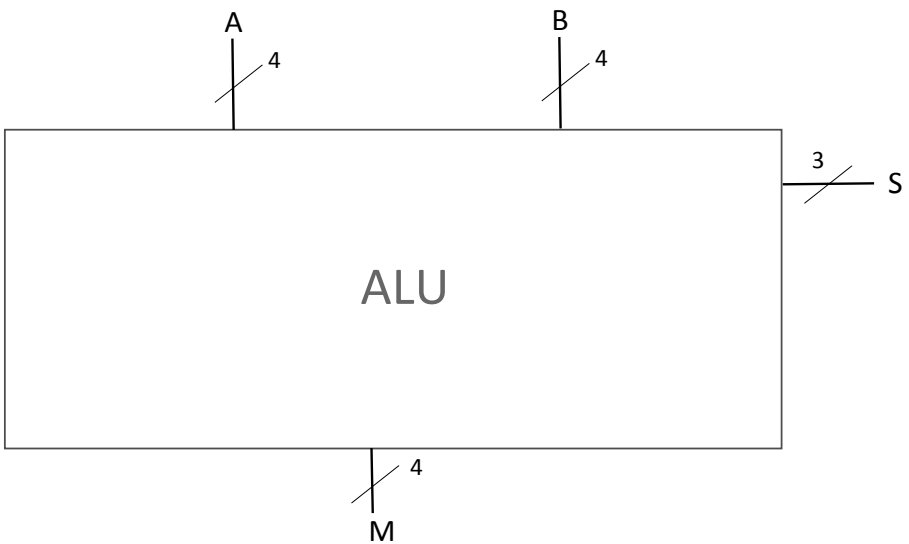


Tabla de valores

S2	S1	S0	M
0	0	0	Suma
0	0	1	Resta
0	1	0	And
0	1	1	Or
1	0	0	Not
1	0	1	Xor
1	1	0	Shift left
1	1	1	Shift right