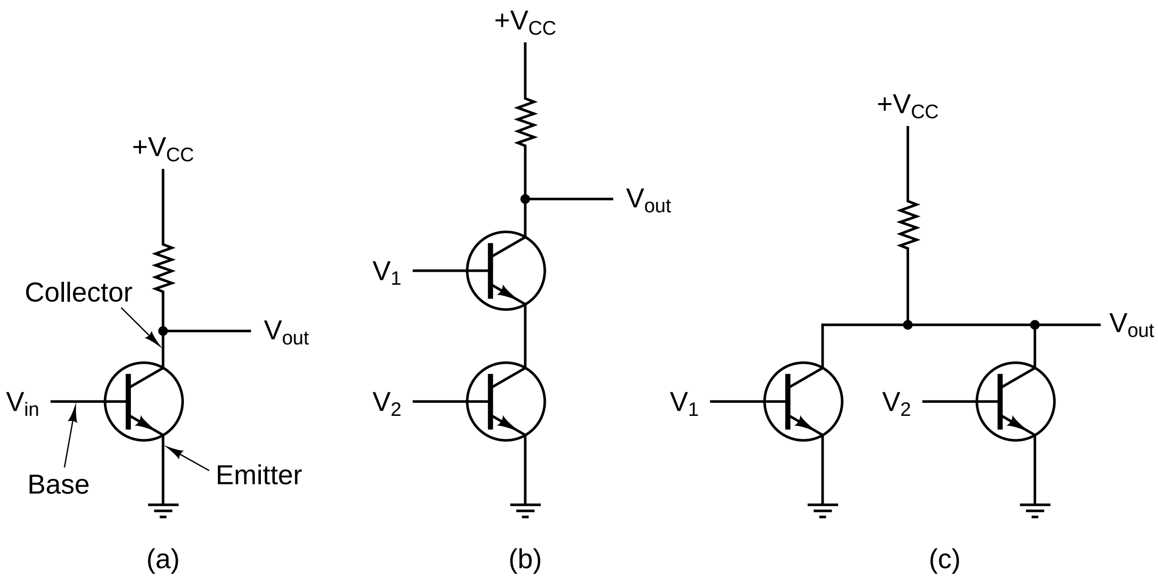
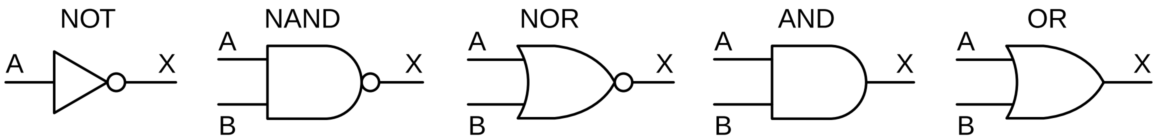


Figure 3-1. (a) A transistor inverter. (b) A NAND gate. (c) A NOR gate.



A	**X**
0	1
1	0

(a)

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

(b)

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

(c)

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

(d)

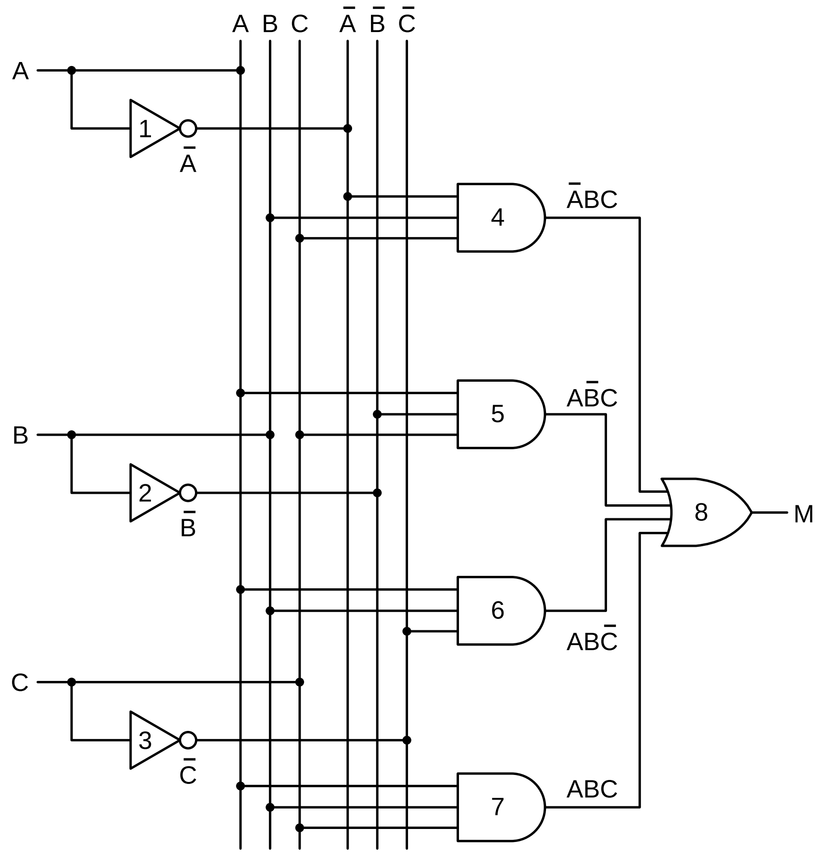
A	**B**	**X**
0	0	0
0	1	1
1	0	1
1	1	1

(e)

Figure 3-2. The symbols and functional behavior for the five basic gates.

A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

(a)



(b)

Figure 3-3. (a) The truth table for the majority function of three variables. (b) A circuit for (a).

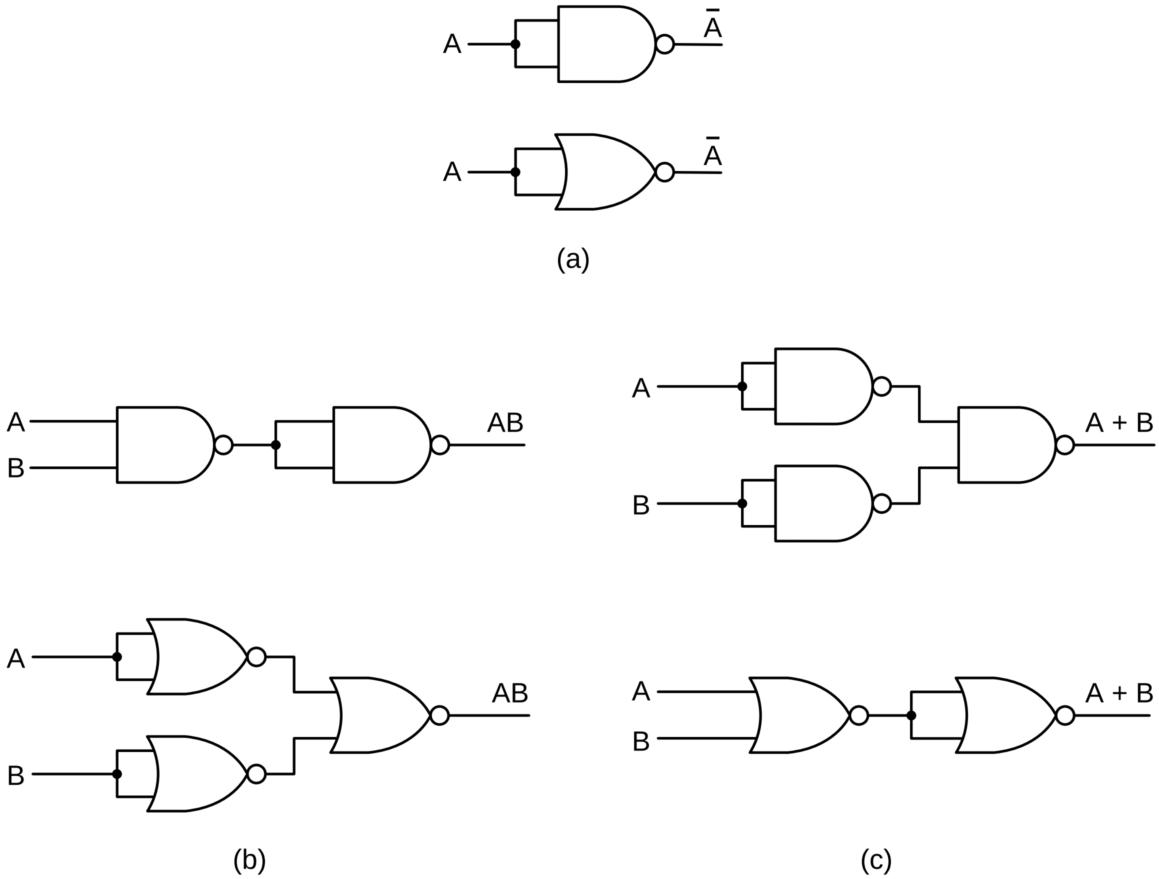


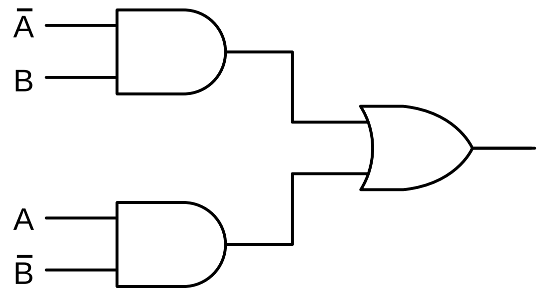
Figure 3-4. Construction of (a) NOT, (b) AND, and (c) OR gates using only NAND gates or only NOR gates.

Name	AND form	OR form
Identity law	1A = A	0 + A = A
Null law	0A = 0	1 + A = 1
Idempotent law	AA = A	A + A = A
Inverse law	$A\overline{A} = 0$	$A + \overline{A} = 1$
Commutative law	AB = BA	A + B = B + A
Associative law	(AB)C = A(BC)	(A + B) + C = A + (B + C)
Distributive law	A + BC = (A + B)(A + C)	A(B + C) = AB + AC
Absorption law	A(A + B) = A	A + AB = A
De Morgan's law	$\overline{AB} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A}\,\overline{B}$

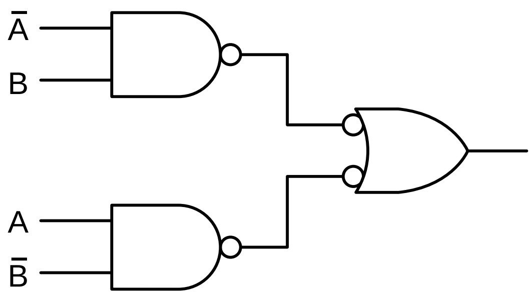
Figure 3-6. Some identities of Boolean algebra.

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

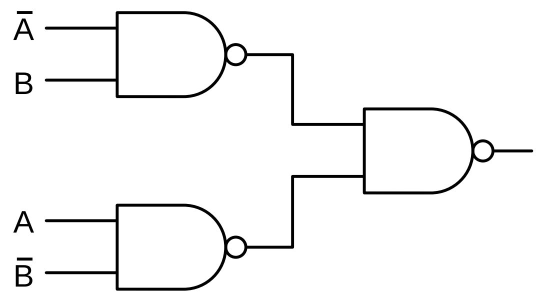
(a)



(b)



(c)



(d)

Figure 3-8. (a) The truth table for the XOR function. (b)-(d) Three circuits for computing it.

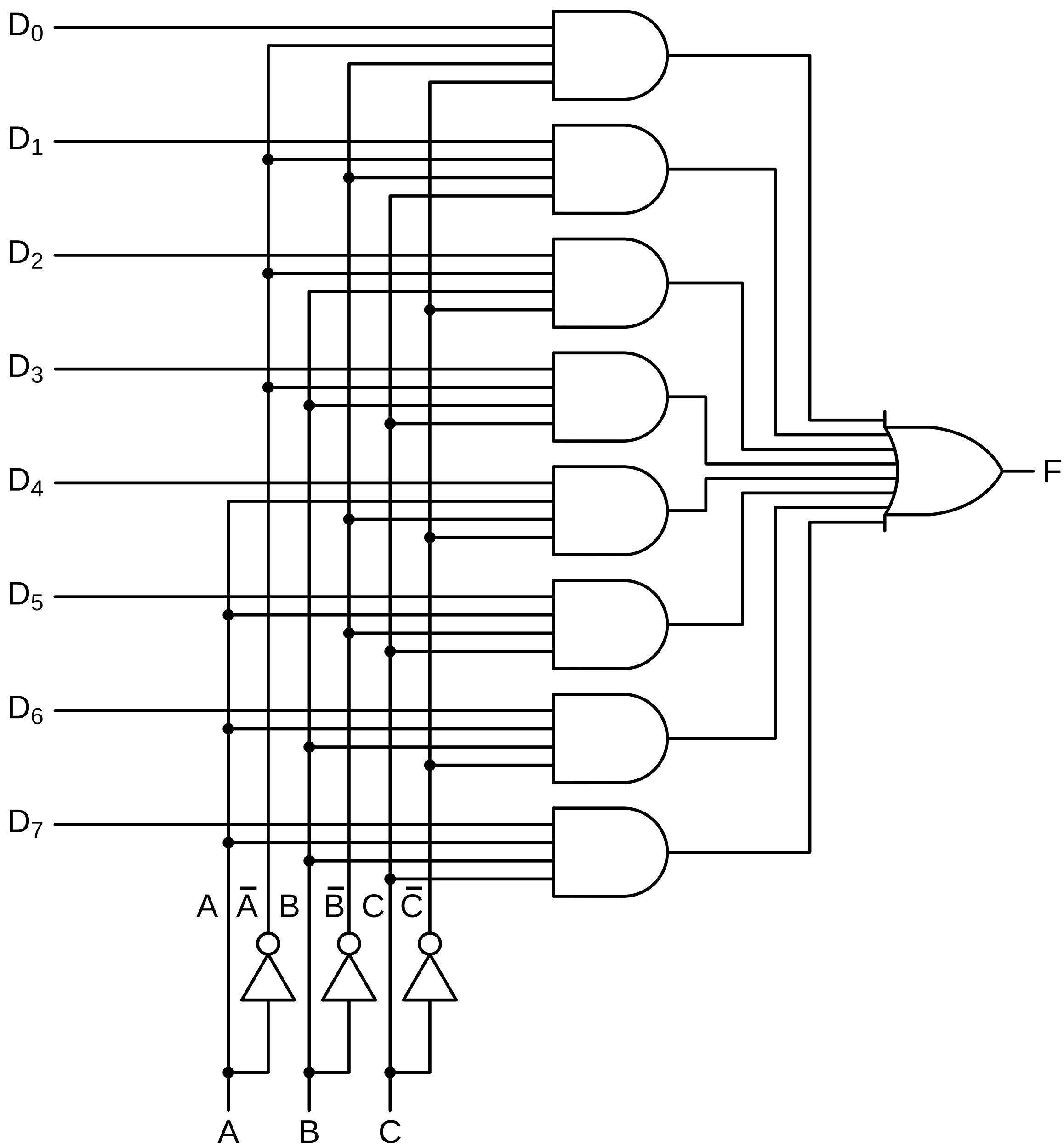


Figure 3-11. An eight-input multiplexer circuit.

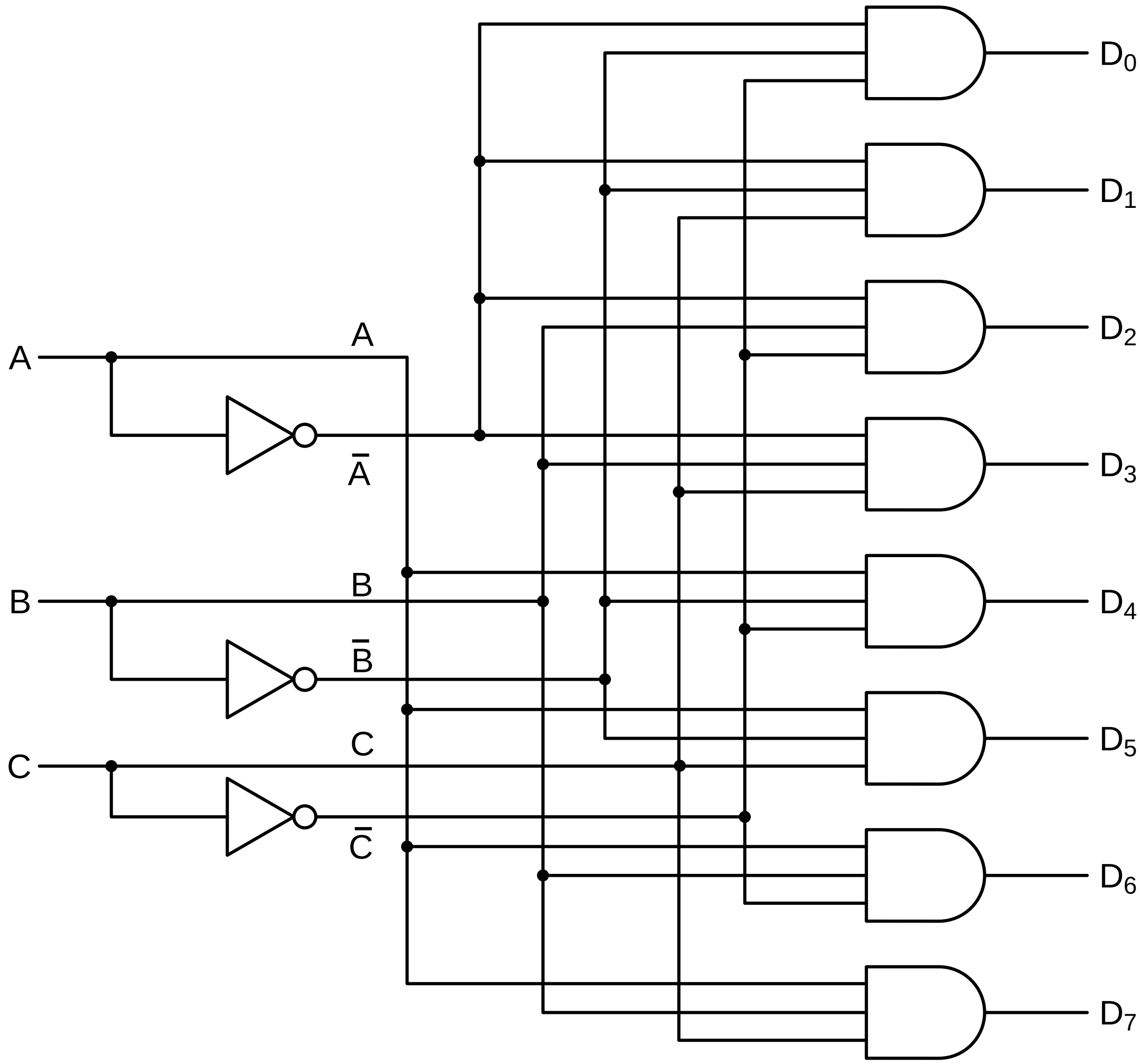


Figure 3-13. A 3-to-8 decoder circuit.

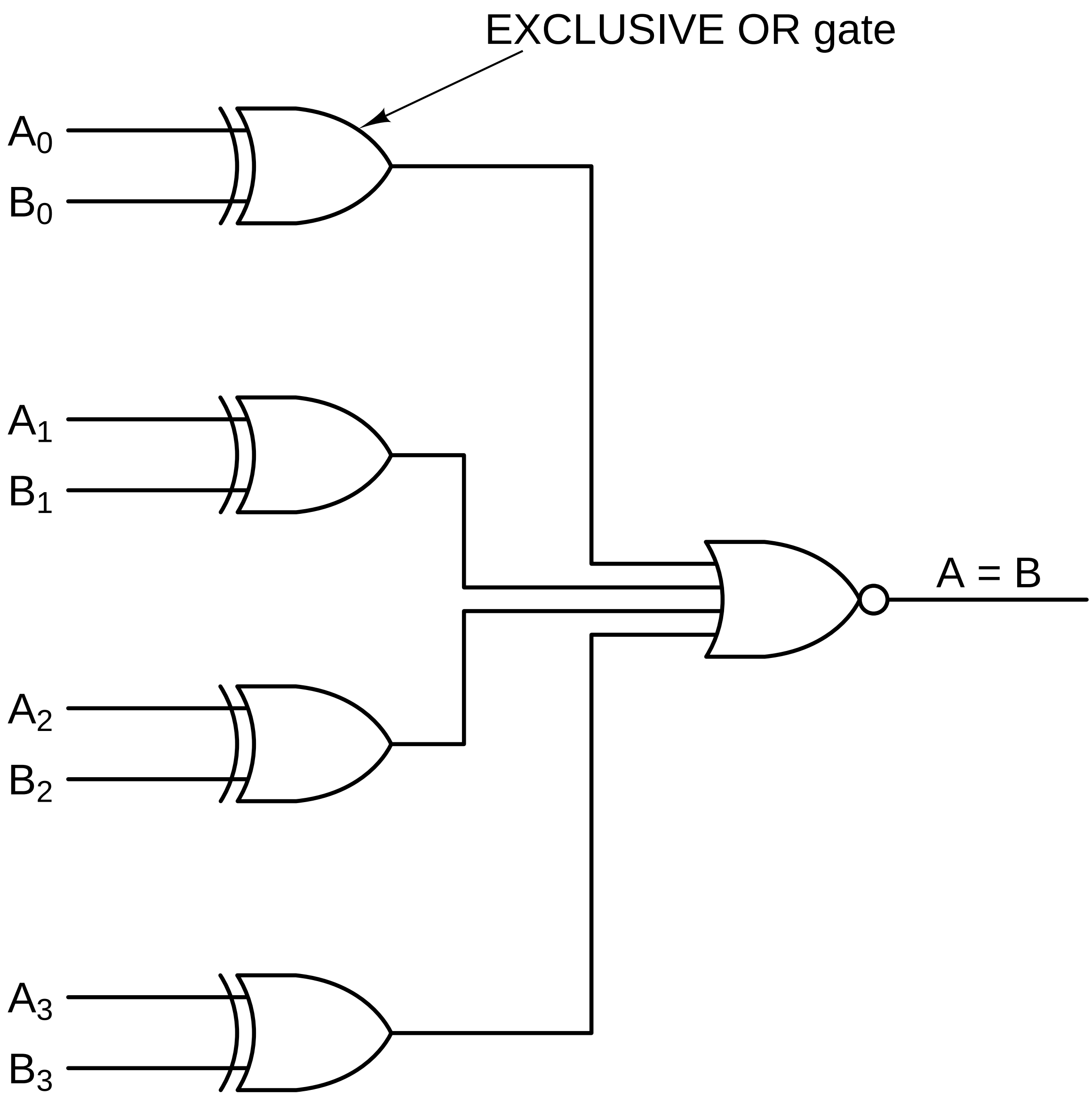


Figure 3-14. A simple 4-bit comparator.

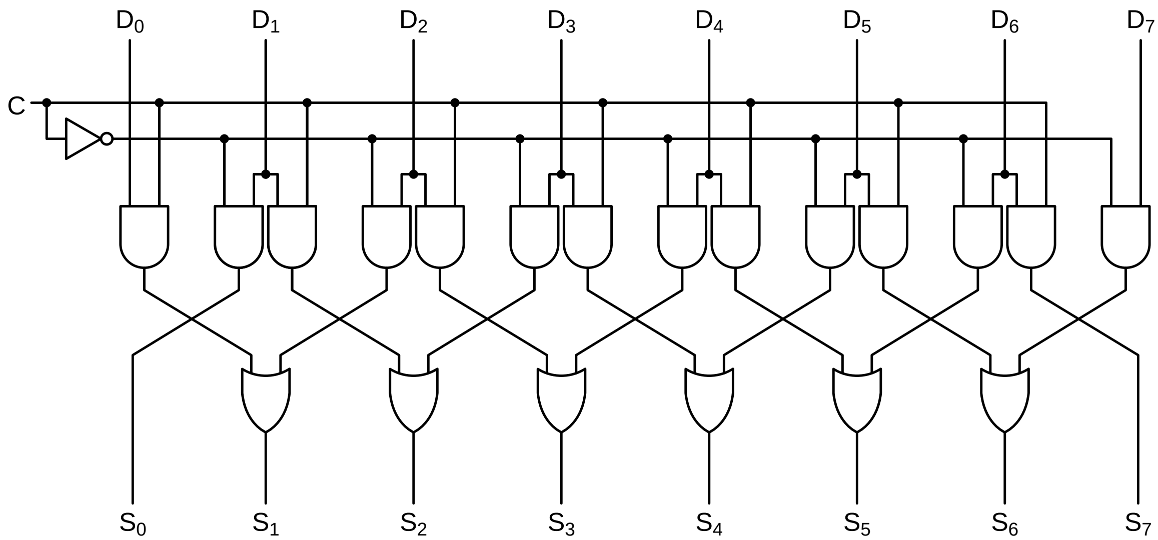


Figure 3-16. A 1-bit left/right shifter.

A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

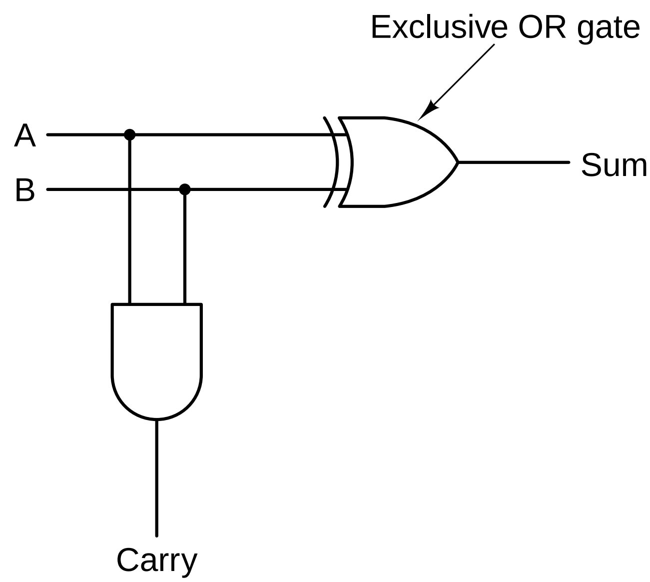
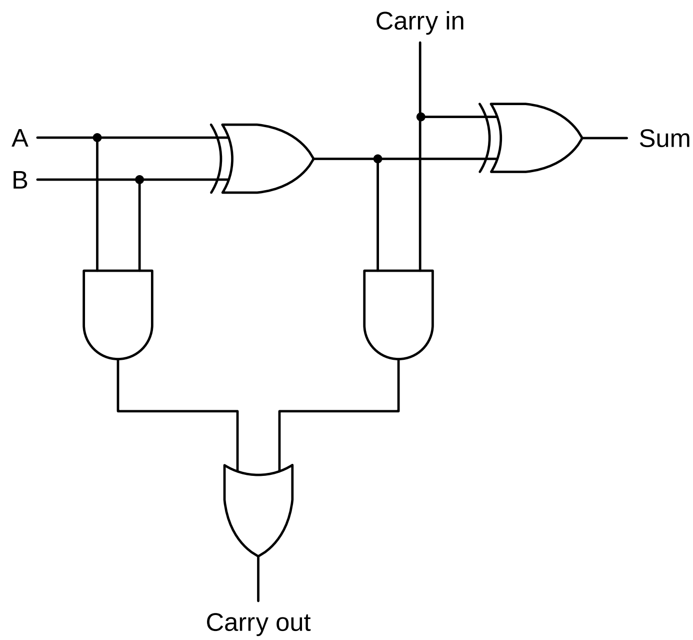


Figure 3-17. (a) Truth table for 1-bit addition. (b) A circuit for a half adder.

A	B	Carry in	Sum	Carry out
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

(a)



(b)

Figure 3-18. (a) Truth table for full adder. (b) Circuit for a full adder.

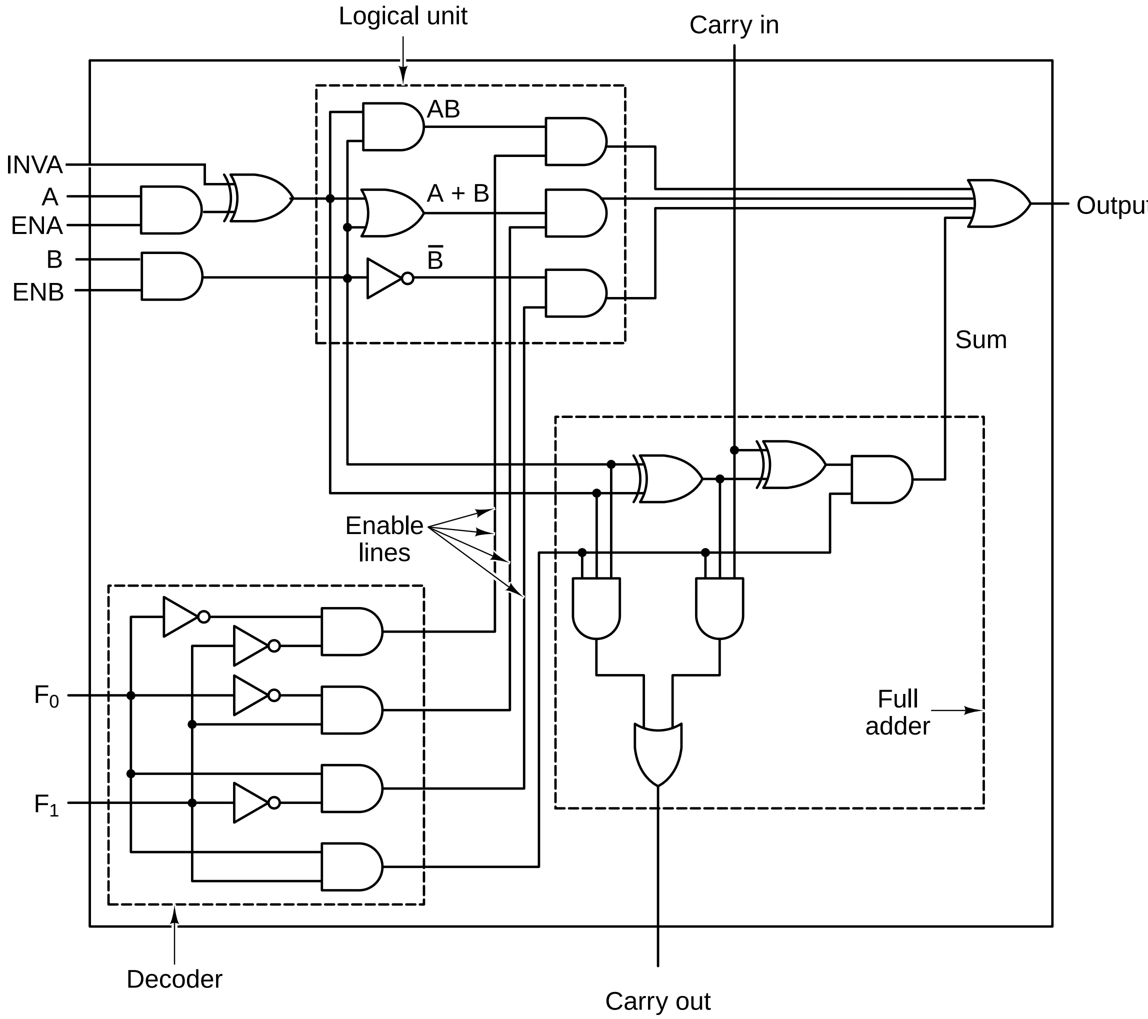


Figure 3-19. A 1-bit ALU.

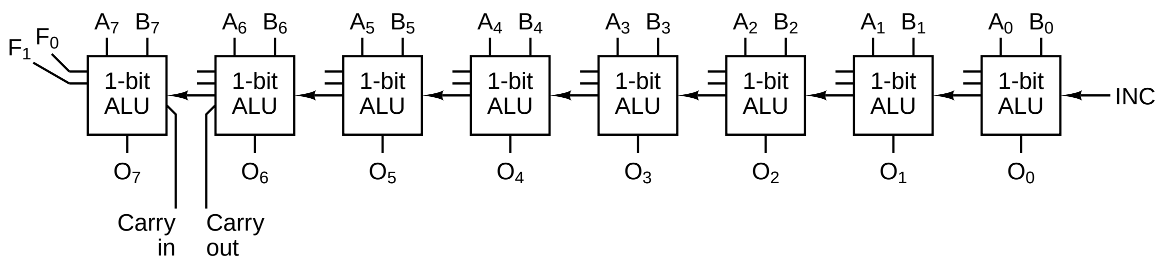


Figure 3-20. Eight 1-bit ALU slices connected to make an 8-bit ALU. The enables and invert signals are not shown for simplicity.

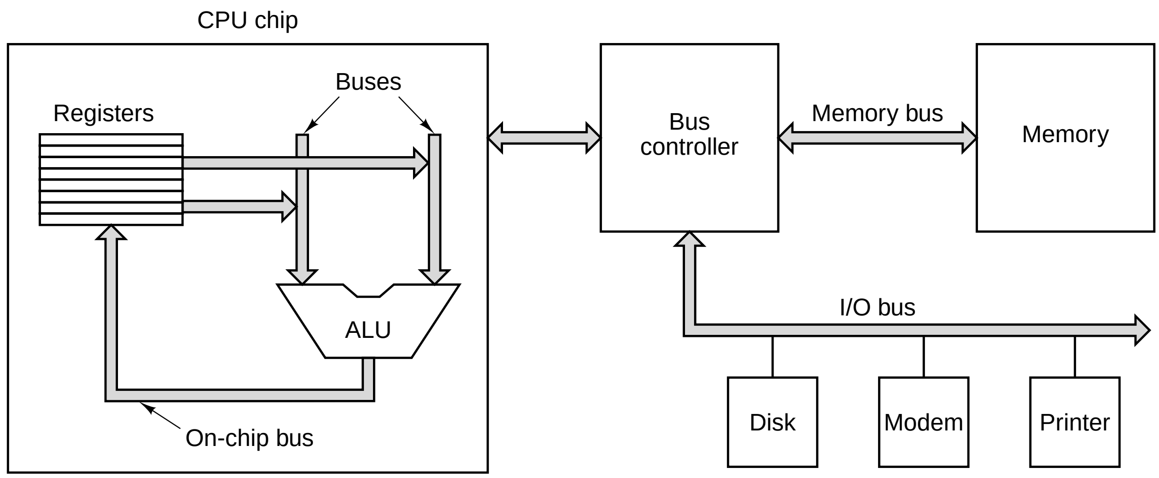


Figure 3-34. A computer system with multiple buses.