

Chapter 2: Semiconductors and Digital Logic

True/False Quiz

Quiz	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Answer	T	F	T	F	T	F	T	F	T	F	T	F	T	F	T

Answer 2.1: In digital electronics, truth tables can be used to reduce basic Boolean operations to simple correlations of inputs to outputs, without the use of logic gates or code.

Answer 2.2: Universal gates are the ones from which we can design any other gate(s). They are the NAND and NOR gates. They aid in forming uniformity in digital circuits.

Answer 2.3: Boolean addition and multiplication are the same as the OR and the AND function respectively. Boolean addition is as follows:

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 1$$

Multiplication in Boolean algebra is the same as in real-number algebra: anything multiplied by 0 is 0, and anything multiplied by 1 remains unchanged:

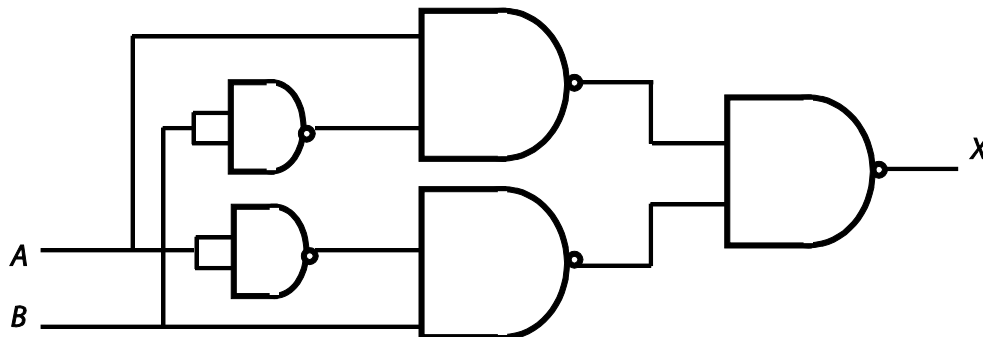
$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

Answer 2.4: Yes, it is possible to implement an XOR gate by only using NAND gate. Figure below shows the implementation.



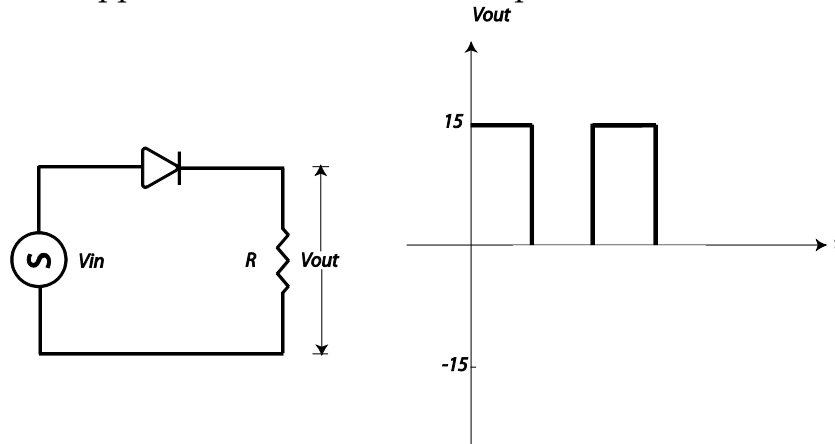
Answer 2.5: If a data-word is sent out with even parity, but has odd parity when it is received then the data has been corrupted and must be resent. As its name implies the operation of an Odd Parity generator is similar but it provides odd parity. The table shows the parity generator outputs for various 8-bit data words.

DATAWORD	EVEN O/P	ODD O/P
00000000	0	1
10000000	1	0
00010000	1	0
11101001	1	0

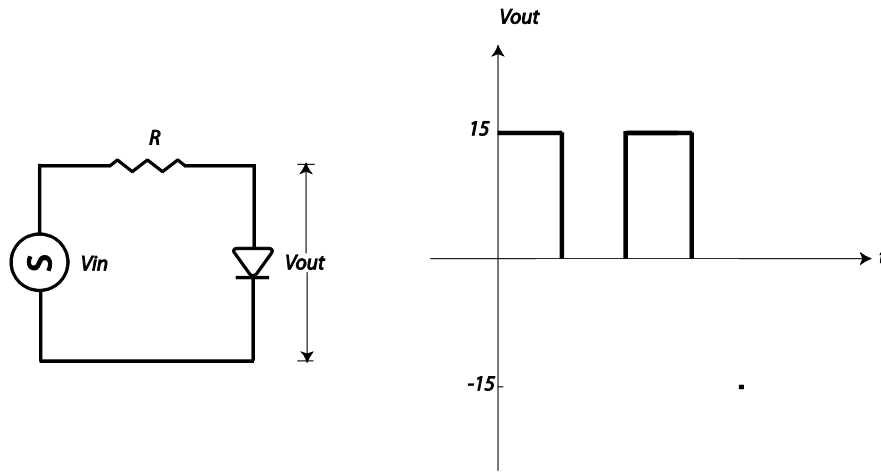
Solution 2.1: During each positive 1/2-cycle the input voltage is +20 volts. The diode acts like a closed switch. As a result, +20 volts appears across the output during each positive 1/2-cycle.

On the other hand, during each negative 1/2-cycle, the diode is reverse-biased and acts like an open switch. Therefore, no voltage is developed across the output.

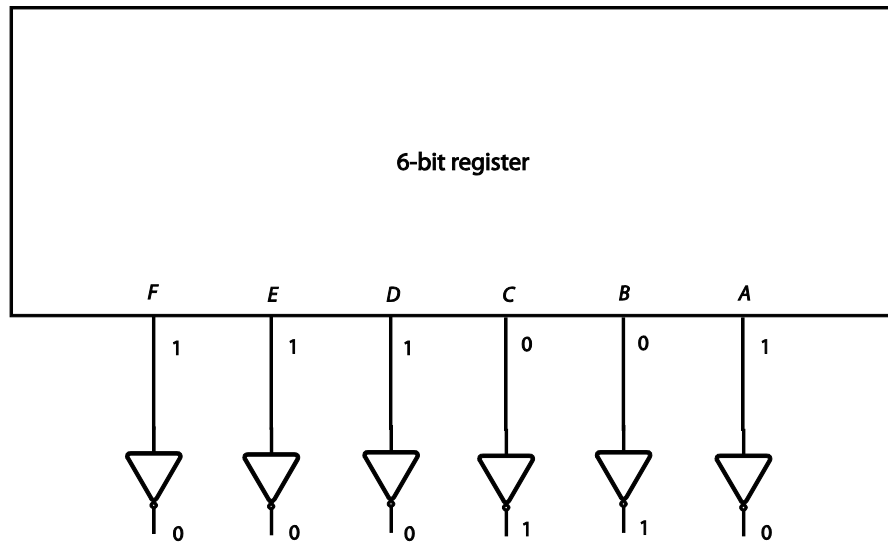
Figure below show the output voltage. Note that the negative parts of the source voltage have been clipped off. This circuit is a simple half wave rectifier.



Solution 2.2: During each positive 1/2-cycle, the diode is in forward bias and acts like a closed switch. Since no voltage can appear across a short, the output voltage must be zero throughout each positive half-cycle. On the other hand, during each negative 1/2-cycle, however, current tries to flow opposite the diode triangle, so that the diode is open. With no current in the circuit there can be no voltage drop across the resistor, and all the input voltage must appear across the output terminals. Figure below shows the output waveform. The circuit is often called a “positive clipper” due to the fact that it clips off all positive parts of the input signal.



Solution 2.3: Use an inverter on each signal line as shown in the figure below. The final output is now 011000. A HEX inverter is a commercially available IC containing six separate inverters. Given a 6-bit register, we can connect a HEX inverter to complement each bit as shown in the figure.



Solution 2.4: The input words are from 0000 0000 to 1111 1111, following the binary progression. The total number of input words is $2^n = 2^8 = 256$. The first 255 input words produce a 0 output. Only the last word, 1111 1111, results in a 1 output due to the fact that all inputs must be high to get a high output.

Solution 2.5: (a) $N = 2^n = 2^6 = 64$ inputs; (b) 000000; (c) The circuit will behave like an OR gate.

Solution 2.6: Because there are two input bits – S0 and S1, there will be four input combinations to consider.

S0	0	0	1	1
S1	0	1	0	1
SUM	0	1	1	0
CARRY	0	0	0	1

Studying the SUM output, you can see that SUM is the exclusive-OR of S0 and S1 (i.e. SUM is high only when S0 is high or S1 is high, but not both). The CARRY output should be high only when S0 AND S1 are both high. This requires an active-high input and output AND gate. This circuit is known as half adder.

Solution 2.7: SEL will be high if A3 is high and A2 is low, or if A1 and A0 are high. Below is the truth table. The SEL is high if condition 1 OR condition 2 are high.

A3	A2	A1	A0	A3 high and A2 low (condition 1)	A1 and A0 high (condition 2)	SEL
0	0	0	0			0
0	0	0	1			0
0	0	1	0			0
0	0	1	1		1	1
0	1	0	0			0
0	1	0	1			0
0	1	1	0			0
0	1	1	1		1	1
1	0	0	0	1		1
1	0	0	1	1		1
1	0	1	0	1		1
1	0	1	1	1	1	1
1	1	0	0			0
1	1	0	1			0
1	1	1	0			0
1	1	1	1		1	1

Solution 2.8: SEL will be high if A2 or A1 is high and A0 is low. Below is the truth table. The SEL is high if condition 1 AND condition 2 are high.

A2	A1	A0	A2 or A1 high (condition 1)	A0 low (condition 2)	SEL
0	0	0		1	0
0	0	1			0
0	1	0	1	1	1
0	1	1	1		0
1	0	0	1	1	1
1	0	1	1		0
1	1	0	1	1	1
0	0	0	1		0

Solution 2.9: The truth table shown below is created by arranging the input portion (from D1 to D4) in increasing binary order corresponding to decimal digit 0 through 9. The output portion (from P1 to P3) lists the segments which are to be “lit” to form the selected digit. Note that because the inputs are BCD digits, there are six unused binary combinations on the input side.

INPUT					OUTPUT						
DD	W	X	Y	Z	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	1	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	0	1	0	1	1
6	0	1	1	0	1	0	0	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1

<i>D4</i>	<i>D3</i>	<i>D2</i>	<i>D1</i>	<i>P3</i>	<i>P2</i>	<i>P1</i>
0	0	0	0	1	1	1
0	0	0	1	1	0	0
0	0	1	0	0	1	0
0	0	1	1	0	0	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	1	1
1	0	0	0	0	0	0
1	0	0	1	0	1	1
1	0	1	0	1	0	1
1	0	1	1	1	1	0
1	1	0	0	1	1	0
1	1	0	1	1	0	1
1	1	1	0	0	1	1
1	1	1	1	0	0	0

Solution 2.10: Method to re-create from Boolean equations:

1. Tabulate all possible combinations of the independent variable in ascending binary order. This will form the rows.
2. Next create an output column for each dependent variable.
3. Then from the Boolean equation, for each output column find out in which rows the 1s are located.
4. Finally, fill in the remaining output column slots with 0s.

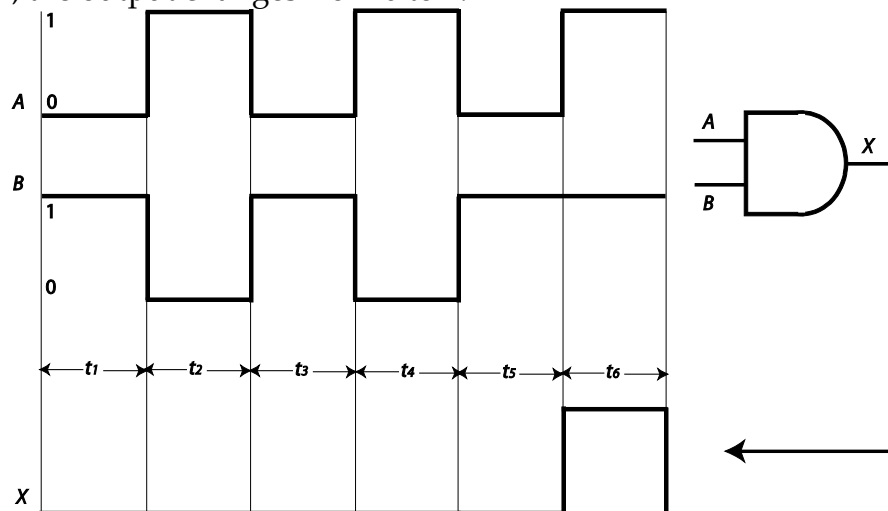
For equation

$$X = A'B'C' + A'BC' + AB'C' + AB'C$$

there are 3 independent variables A, B, C, and four AND terms. Each of these corresponds to a truth table row where the output column 'X' contains a 1. In each of these AND terms, substitute 1 for a direct variable and a 0 for a logically inverted variable to obtain the desired row codes. The four rows where $F = 1$ are thus observed to correspond to the input variable combinations $ABC = 000, 010, 100, \text{ and } 101$. After using the same process to identify the rows corresponding to $Y = 1$, the truth table may be completed.

INPUT			OUTPUT	
A	B	C	X	Y
0	0	0	1	0
0	0	1	0	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	0	1
1	1	1	0	0

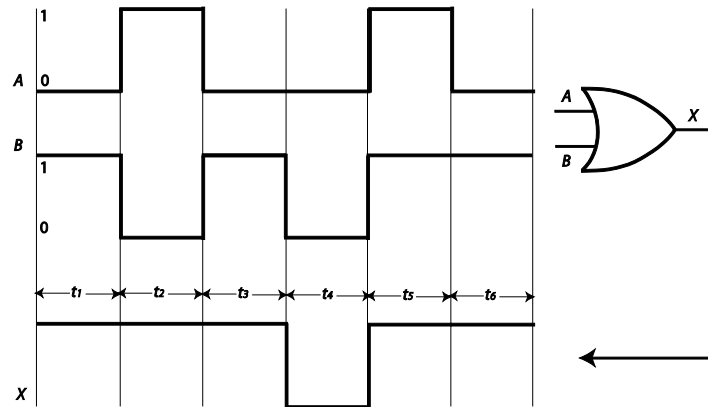
Solution 2.11: The output X will be at 1 only when A and B are both HIGH at the same time. Thus, for the first five time intervals, the output will remain at zero. In the last time interval, the output changes from 0 to 1.



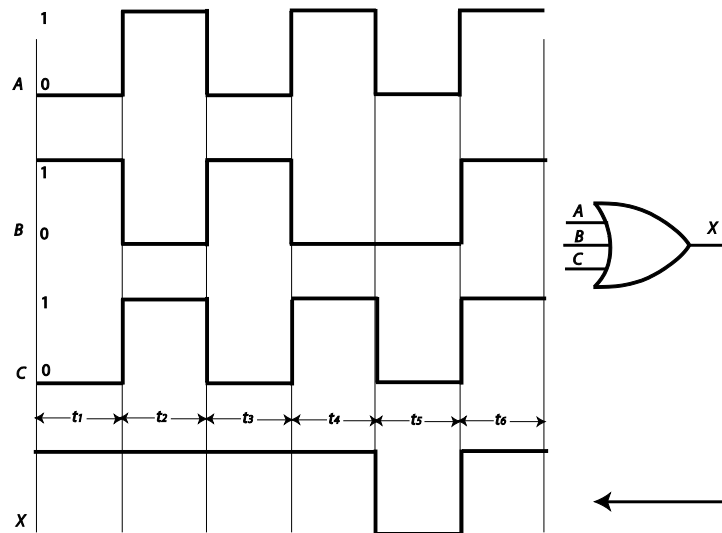
Solution 2.12: The output will always remain at 0

Solution 2.13: The total number of input words is $2^n = 2^8 = 256$. In any OR gate, 1 or more high input produce a high output. All the input words except 0000 0000 produce a high output.

Solution 2.14: The OR gate output will be HIGH whenever any input is HIGH. Thus, in the timing diagram, the fourth time interval is low, rest are high.

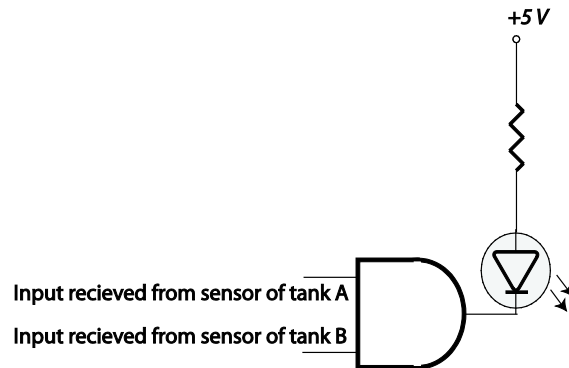


Solution 2.15: The OR gate output is determined by realizing that it will be HIGH whenever any of the three inputs is at a HIGH level. Thus, in the timing diagram, the fifth time interval is low, rest are high.



Solution 2.16: With the C input is 1, the OR gate output will remain 1 regardless of what is occurring at the other inputs. Because any HIGH input will keep an OR gate output high. Therefore, the glitch will not appear in the output.

Solution 2.17: The figure shows a NAND gate with its two inputs connected to the tank level sensors inputs and its output connected to the indicator panel.



Solution 2.18: Let the inputs be A and B, such that B is the inverter input, and Y be the output. The truth table can be constructed as:

A	B	B'	$Y = A \oplus B'$
0	0	1	$0 \oplus 1 = 1$
0	1	0	$0 \oplus 0 = 0$
1	0	1	$1 \oplus 1 = 0$
1	1	0	$1 \oplus 0 = 1$

We can observe that when $B = 1$, $Y = A$. Also, when $B=0$, $Y = A'$. The circuit transmits A when inverter input is 0 and A' when inverter input is 1.

Solution 2.19: Each XOR gate acts like that of Problem 2.1, i.e. $Y_n = A_n$ when INVERT is low, and $Y_n = A'_n$ when INVERT is high. In the latter case, each bit is either transmitted or inverted before reaching the final output. If the register contents as a word $A_7A_6A_5A_4A_3A_2A_1A_0$ and the final output as a word $Y_7Y_6Y_5Y_4Y_3Y_2Y_1Y_0$, then a low INVERT means $Y_7Y_6Y_5Y_4Y_3Y_2Y_1Y_0 = A_7A_6A_5A_4A_3A_2A_1A_0$ for low INVERT and $Y_7Y_6Y_5Y_4Y_3Y_2Y_1Y_0 = A'_7A'_6A'_5A'_4A'_3A'_2A'_1A'_0$ for high INVERT. Example for low and high INVERT for an input of 1011 0011 is 1011 0011 and 0100 1100 respectively.

Solution 2.20: Output is 1 when even number of 1s is present in the input. The output becomes 0 when odd number of inputs has 1s.

- 0000 0000 0000 0000 has even number of 1s, so output is 1.
- 0000 0000 0000 1111 has even number of 1s, so output is 1.
- 1111 0101 1111 1100 has even number of 1s, so output is 1.
- 0101 1100 0111 0011 has odd number of 1s, so output is 0.
- 1111 0000 1010 0110 has even number of 1s, so output is 1.
- 1111 1111 1111 1110 has odd number of 1s, so output is 0.