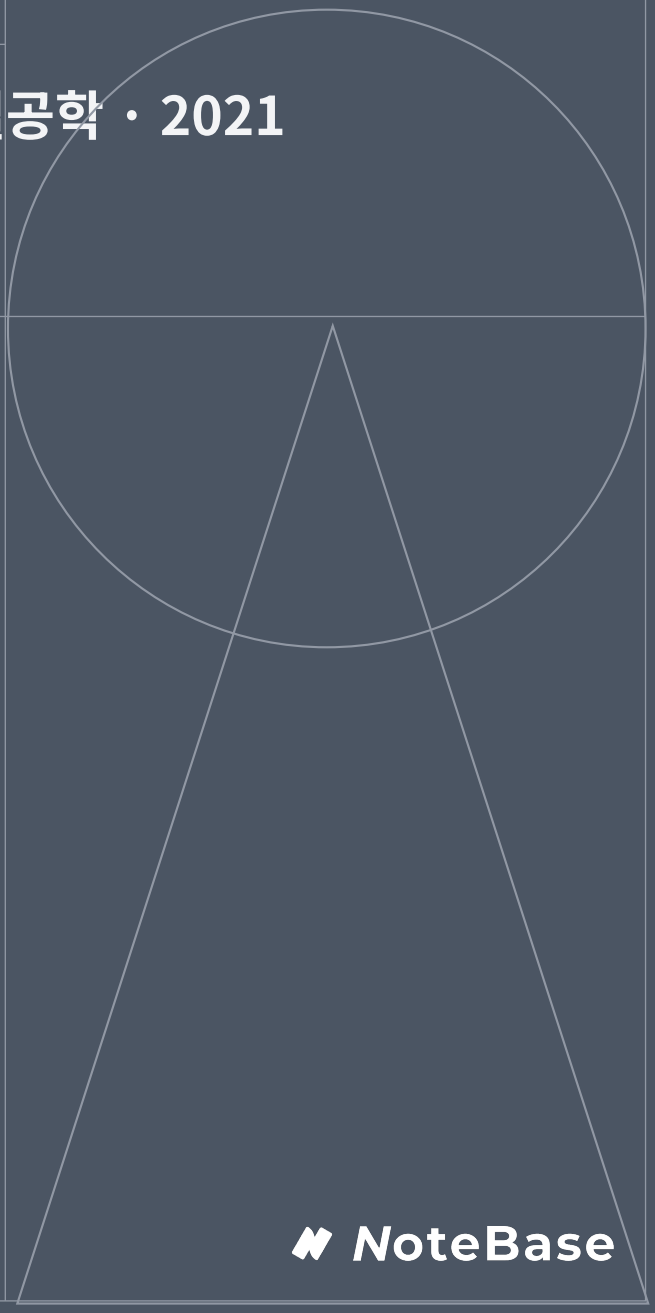
기출OX 해설집

공무원/서울시/7급 · 디지털공학 · 2021



NoteBase

기출OX 해설집

공무원/서울시/7급 · 디지털공학 · 2021

공무원/서울시/7급 · 디지털공학 · 2021

문 1. 디지털 시스템의 특징에 대한 설명으로 가장 옳지 않은 것은?

① 디지털 시스템은 하나 또는 다수의 디지털 입력과 출력을 가지며, 조합논리회로와 순서논리회로로 구별한다.

② 디지털 시스템은 아날로그 시스템에 비해 잡음의 영향을 덜 받는다.

③ 조합논리회로는 현재의 입력과 과거의 입력에 의해서 현재의 출력이 결정되는 회로이다.

④ 2진법을 사용하는 디지털 시스템은 0과 1을 2개의 전압레벨에 대응시킨다.

정답 3번

출제 포인트

이 문제는 디지털 시스템의 일반적 특징과 조합논리회로, 순서논리회로의 구분 기준을 정확히 아는지를 묻는 문제이다.

핵심 개념

디지털 시스템은 이산적인 값으로 정보를 표현하고 처리하는 시스템이다.
조합논리회로는 현재 입력만으로 현재 출력이 결정되는 회로이다.
순서논리회로는 현재 입력과 함께 과거 상태나 기억된 정보의 영향을 받아 출력이 결정되는 회로이다.
디지털 시스템은 보통 0과 1을 두 전압 레벨에 대응시키므로 아날로그 시스템보다 잡음에 강한 편이다.

NoteBase

선지별 해설

① O
디지털 시스템은 하나 또는 여러 개의 디지털 입력과 출력을 가질 수 있으며, 동작 방식에 따라 조합논리회로와 순서논리회로로 구분한다.

② O
디지털 시스템은 0과 1의 두 수준을 기준으로 신호를 판단하므로 연속값을 직접 다루는 아날로그 시스템보다 잡음의 영향을 덜 받는다.

③ X
틀린 부분은 "조합논리회로는 현재의 입력과 과거의 입력에 의해서 현재의 출력이 결정된다"는 표현이다.
조합논리회로의 출력은 현재 입력에 의해서만 결정된다.
올바르게는 "순서논리회로는 현재 입력과 과거 상태에 의해 출력이 결정되고, 조합논리회로는 현재 입력만으로 출력이 결정된다"이다.

④ O
2진 디지털 시스템은 0과 1의 두 논리값을 사용하며, 이를 실제 회로에서는 보통 두 개의 전압 레벨로 대응시켜 표현한다.

오답 포인트

③은 조합논리회로와 순서논리회로의 핵심 차이를 뒤바꾼 선지이다. "과거의 입력"이나 "기억"이 들어가면 순서논리회로를 떠올려야 하는데, 이 표현을 조합논리회로에 붙여 놓아 혼동을 유도한다.

다시 나오면 이렇게 판단

"현재 입력만"이면 조합논리회로이고, "현재 입력 + 과거 상태"이면 순서논리회로로 판단하면 된다. 디지털 시스템의 일반 특징을 묻는 선지에서는 잡음 내성, 2진 표현, 전압 레벨 대응 여부를 함께 확인하면 된다.

암기 포인트

조합은 현재만, 순서는 기억까지 포함한다.

NoteBase

문 2. <보기>의 다양한 진법으로 표현한 숫자들을 큰 숫자부터 작은 숫자의 순서대로 바르게 나열한 것은?

<보기>
(가) $F9_{(16)}$ (16진법)
(나) $256_{(10)}$ (10진법)
(다) $11111111_{(2)}$ (2진법)
(라) $370_{(8)}$ (8진법)

① (가) - (나) - (다) - (라)
② (나) - (다) - (가) - (라)
③ (다) - (라) - (가) - (나)
④ (라) - (가) - (나) - (다)

정답 2번

출제 포인트

서로 다른 진법의 수를 10진수로 변환한 뒤 크기를 비교하는 문제이다.

조건 정리

주어진 값: (가) F9(16), (나) 256(10), (다) 11111111(2), (라) 370(8)
구해야 할 값: 큰 수부터 작은 수까지의 올바른 배열
주의할 조건: 각 수를 같은 기준인 10진수로 바꾼 뒤 비교해야 한다.

사용할 공식

진법 변환은 각 자리값의 가중치를 더해 10진수로 바꾼다.
16진수는 16의 자리값, 8진수는 8의 자리값, 2진수는 2의 자리값을 사용한다.
이 공식을 쓰는 이유: 서로 다른 진법의 수는 표기만 보고 크기를 바로 비교할 수 없으므로 같은 10진수 기준으로 통일해야 하기 때문이다.

계산 과정

(가) F9(16) = 15×16 + 9 = 249
(나) 256(10) = 256
(다) 11111111(2) = 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255
(라) 370(8) = 3×64 + 7×8 + 0 = 248
따라서 크기 순서는 256 > 255 > 249 > 248이므로 (나) - (다) - (가) - (라)이다.

NoteBase

선지별 해설

① X
(가) - (나) - (다) - (라)라고 했지만 실제 최대값은 (나)=256이므로 맨 앞에 와야 한다. 틀린 표현은 '(가)가 가장 크다'는 배열이고, 올바르게는 '(나) - (다) - (가) - (라)'이다.

② O
(나)=256, (다)=255, (가)=249, (라)=248이므로 큰 수부터 작은 수까지 배열하면 (나) - (다) - (가) - (라)이다. 계산 결과와 정확히 일치한다.

③ X
(다) - (라) - (가) - (나)라고 했지만 (나)=256이 가장 큰데 맨 뒤에 배치되어 있다. 틀린 표현은 '(나)가 가장 작다'는 배열이고, 올바르게는 '(나) - (다) - (가) - (라)'이다.

④ X
(라) - (가) - (나) - (다)라고 했지만 (라)=248은 네 수 중 가장 작다. 틀린 표현은 '(라)가 가장 크다'는 배열이고, 올바르게는 '(나) - (다) - (가) - (라)'이다.

오답 포인트

16진수 F를 15가 아니라 문자로만 보고 넘어가거나, 11111111(2)를 256으로 착각하면 순서가 뒤바뀐다. 특히 255와 256이 연속된 수라는 점 때문에 (나)와 (다)의 순서를 바꾸는 실수가 자주 나온다.

다시 나오면 이렇게 판단

진법 비교 문제는 먼저 각 수의 진법을 확인하고, 모두 10진수로 변환한 다음, 변환 결과만 놓고 내림차순 또는 오름차순을 정한다.

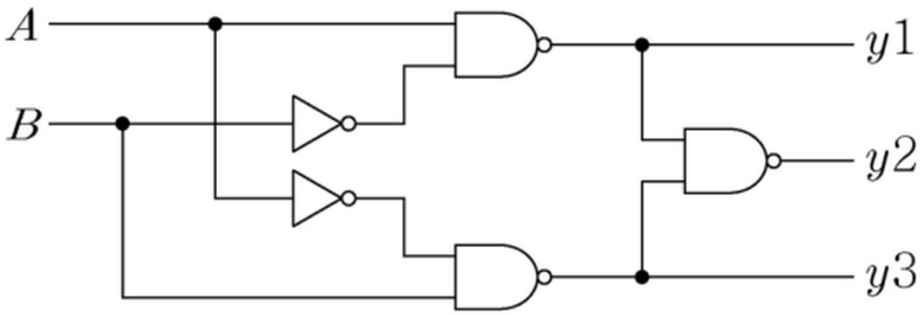
암기 포인트

F는 15이고, 11111111(2)는 255이며, 370(8)은 248이다.

NoteBase

문 3. <보기>의 회로에 대한 설명으로 가장 옳은 것은?

<보기>



① $A > B$이면, $y_1 = 1$이다.

② $A < B$이면, $y_1 = 0$이다.

③ $A = B$이면, $y_2 = 0$이다.

④ $A > B$이면, $y_3 = 0$이다.

정답 3번

출제 포인트

이 문제는 NAND와 인버터로 만든 1비트 비교 회로에서 각 출력이 어떤 입력 상태에서 0이 되는지를 판별하는 문제이다. 회로의 연결을 따라가면, 위쪽 NAND는 A와 B를 반전한 $\overline{B}$를 입력받으므로 $y1 = \overline{A \cdot \overline{B}}$이고, 아래쪽 NAND는 A를 반전한 $\overline{A}$와 B를 입력받으므로 $y3 = \overline{\overline{A} \cdot B}$이다. 가운데 NAND는 이 y1과 y3를 두 입력으로 받으므로 $y2 = \overline{y1 \cdot y3}$이 된다. 여기에 y1, y3를 대입하고 드모르간 법칙을 적용하면 $y2 = \overline{\overline{A \cdot \overline{B}} \cdot \overline{\overline{A} \cdot B}} = (A \cdot \overline{B}) + (\overline{A} \cdot B) = A \oplus B$가 되어, 두 입력이 서로 다를 때만 1이 되는 XOR 출력임을 확인할 수 있다.

NoteBase

선지별 해설

① X

$A > B$는 $(A, B) = (1, 0)$인 경우이다. 이때 $y1 = \overline{1 \cdot 1} = 0$이므로 옳지 않다. 틀린 표현은 "$y1 = 1$"이고, 올바르게는 "$A > B$이면 $y1 = 0$이다."이다.

② X

$A < B$는 $(A, B) = (0, 1)$인 경우이다. 이때 $y1 = \overline{0 \cdot 0} = 1$이므로 옳지 않다. 틀린 표현은 "$y1 = 0$"이고, 올바르게는 "$A < B$이면 $y1 = 1$이다."이다.

③ O

$A = B$는 $(0, 0)$ 또는 $(1, 1)$이다. 두 경우 모두 $y2 = A \oplus B = 0$이므로 옳다. 즉 이 출력은 두 입력이 다를 때만 1이 된다.

④ X

$A > B$는 $(1, 0)$인 경우이고, 이때 $y3 = \overline{0 \cdot 0} = 1$이다. 따라서 옳지 않다. 틀린 표현은 "$y3 = 0$"이고, 올바르게는 "$A > B$이면 $y3 = 1$이다."이다.

오답 포인트

이 회로는 비교 결과가 1이 아니라 0으로 활성화되는 NAND 출력이라는 점이 함정이다. $A > B$, $A = B$, $A < B$를 각각 직접 대입하지 않고 일반적인 비교기 출력처럼 생각하면 ①과 ④를 뒤집기 쉽다.

다시 나오면 이렇게 판단

NAND 비교 회로는 각 출력선이 어떤 입력 조합에서 0이 되는지부터 본다. 먼저 인버터를 반영해 식을 쓰고, $(A, B) = (1, 0), (0, 1), (0, 0), (1, 1)$를 필요한 경우에만 대입하면 관계식이 바로 정리된다.

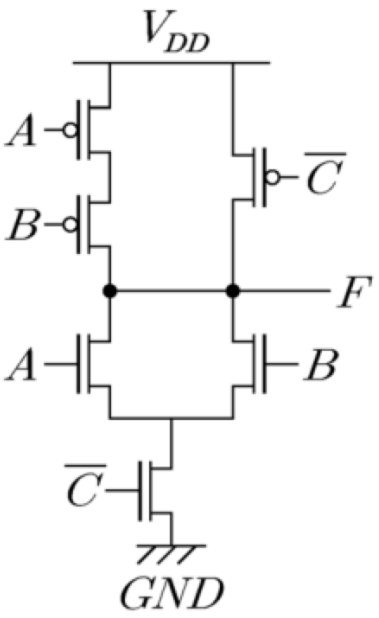
암기 포인트

이 회로는 저활성 비교기이다. $A > B$이면 $y1 = 0$, $A = B$이면 $y2 = 0$, $A < B$이면 $y3 = 0$이 된다.

NoteBase

문 4. <보기>의 CMOS 회로에서 출력 F를 가장 옳게 표현한 논리식은?

<보기>



① $F = \overline{AB} + C$

② $F = AB + C$

③ $F = AB + \overline{C}$

④ $F = \overline{AB} + \overline{C}$

정답 1번

출제 포인트

이 문제는 CMOS 회로의 pull-down(nMOS) 네트워크와 pull-up(pMOS) 네트워크를 보고 출력식 F를 구하는 문제이다. CMOS는 두 네트워크가 항상 상보적이어서, 아래 nMOS 경로가 도통하는 조건을 $F = 0$ 조건으로 읽어 여집합을 취하거나, 위 pMOS 경로가 도통하는 조건을 $F = 1$ 조건으로 직접 읽어도 같은 식이 나온다. 두 경로로 교차 검증하면 실수를 줄일 수 있다.

NoteBase

선지별 해설

① O: 아래 nMOS 네트워크에서 A와 B는 서로 병렬이고, 그 아래 게이트 $\overline{C}$의 nMOS가 직렬로 GND에 연결되므로 방전(도통) 조건은 $(A+B)\overline{C}$이다. 따라서 출력은 $F=\overline{(A+B)\overline{C}}=\overline{A+B}+C=\overline{AB}+C$가 되어 옳다. 위 pMOS 네트워크로도 확인하면, 왼쪽은 게이트 A와 B의 pMOS가 직렬이라 $A=0, B=0$일 때($\overline{AB}$) 도통하고, 오른쪽은 게이트 $\overline{C}$의 pMOS가 있어 $C=1$일 때 도통하며, 두 가지가 병렬이므로 $F=1$ 조건은 $\overline{AB}+C$로 nMOS 결과와 일치한다.
② X: $AB+C$는 nMOS 방전 조건의 여집합 관계를 반영하지 못한 식이다. 회로에서 A와 B의 nMOS는 서로 병렬이므로 방전 조건에는 $A+B$가 들어가고, 그 여집합은 $\overline{A+B}=\overline{AB}$이다. AB가 되려면 두 nMOS가 직렬이어야 하는데 이 회로는 그렇지 않다. 틀린 표현은 AB이고, 올바르게는 $\overline{AB}$이다.
③ X: $AB+\overline{C}$는 A, B 부분과 C 항의 극성이 모두 틀리다. A, B의 nMOS는 직렬이 아니라 병렬이므로 여집합은 AB가 아니라 $\overline{AB}$이고, 게이트 $\overline{C}$의 nMOS가 켜지는 것은 $\overline{C}=1$일 때이므로 출력이 0으로 내려가고 출력식에는 $+\overline{C}$가 아니라 $+C$가 들어간다. 올바르게는 $\overline{AB}+C$이다.
④ X: $\overline{AB}$ 부분은 맞지만 C 항이 틀리다. 오른쪽 pMOS의 게이트가 $\overline{C}$이므로 $C=1$일 때 이 가지가 켜져 출력이 1이 된다. 틀린 표현은 $+\overline{C}$이고, 올바르게는 $+C$이다.

오답 포인트

CMOS 문제에서 가장 흔한 실수는 아래 nMOS 네트워크가 곧바로 출력식이라고 착각하는 점이다. 이 회로에서는 아래쪽이 $(A+B)\overline{C}$일 때 출력이 0이 되므로, 이를 한 번 뒤집어 F를 써야 한다는 점이 함정이다. 또한 nMOS의 병렬/직렬을 반대로 읽어 $A+B$를 AB로 잘못 두는 실수도 잦다.

다시 나오면 이렇게 판단

먼저 GND까지 연결되는 nMOS 경로를 찾아 $F=0$ 조건을 만든다(직렬은 곱, 병렬은 합). 그다음 그 식 전체를 부정하여 F를 쓰고, 드모르간 법칙으로 정리한다. 시간이 되면 pMOS 네트워크의 $F=1$ 조건도 독립적으로 구해 두 결과가 상보적으로 일치하는지 교차 확인한다.

암기 포인트

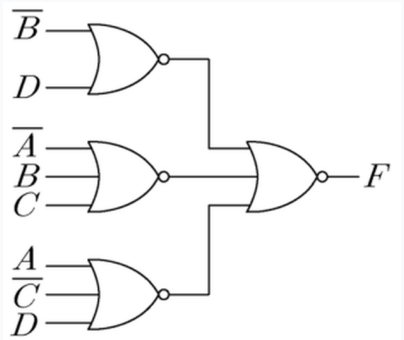
CMOS는 아래 nMOS 네트워크가 0을 만들고 위 pMOS 네트워크가 1을 만들며 둘은 항상 상보적이다. 최종 출력식은 nMOS 도통 조건의 여집합(=pMOS 도통 조건)이라는 점이 핵심이다.

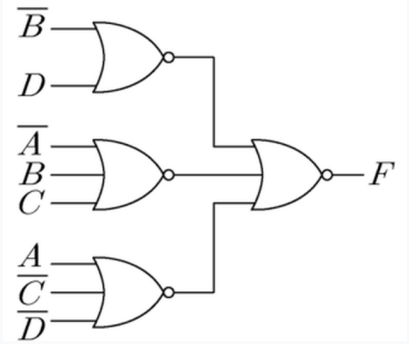
NoteBase

문 5. <보기>의 진리표를 NOR 게이트만으로 구현한 것은?

<보기>

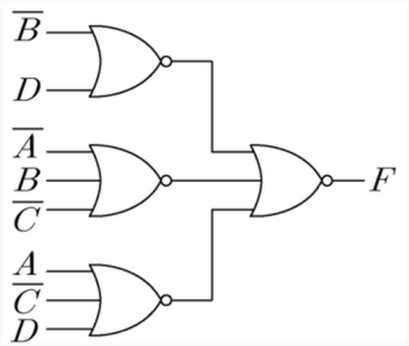
$ABCD$	F	$ABCD$	F
0000	1	1000	0
0001	1	1001	0
0010	0	1010	1
0011	1	1011	1
0100	0	1100	0
0101	1	1101	1
0110	0	1110	0
0111	1	1111	1

① 

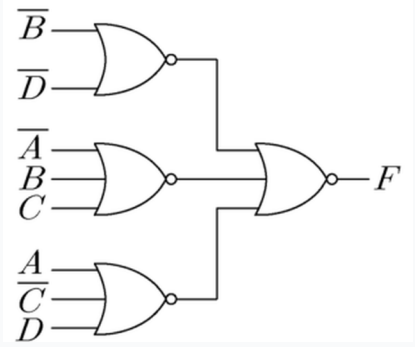
② 

NoteBase

③



④



정답 1번

출제 포인트

이 문제는 NOR-NOR 2단 회로를 드모르간 법칙으로 곱의 합 형태로 바꾸고, 그 결과가 주어진 진리표와 일치하는지를 판단하는 문제이다.

핵심 정리

NOR 게이트의 출력은 입력합의 부정이다. 1단 NOR들의 출력을 마지막 NOR에 다시 넣으면 $\overline{\overline{X_1}+\overline{X_2}+\overline{X_3}}=X_1X_2X_3$가 되어 각 입력합이 곱 형태로 복원된다.

NoteBase

선지별 해설

① O

상단, 중단, 하단 1단 NOR의 입력합은 각각 $(\overline{B}+D)$, $(\overline{A}+B+C)$, $(A+\overline{C}+D)$이다. 마지막 NOR까지 합치면 $F=(\overline{B}+D)(\overline{A}+B+C)(A+\overline{C}+D)$가 되고, 0010, 0100, 0110, 1000, 1001, 1100, 1110에서만 0이 되어 표와 정확히 일치한다.

② X

하단 1단 NOR의 입력이 $A, \overline{C}, \overline{D}$이므로 식이 $F=(\overline{B}+D)(\overline{A}+B+C)(A+\overline{C}+\overline{D})$가 된다. 이 경우 ABCD=0010에서 1이 나오고 0011, 0111에서 0이 나와 표와 어긋난다. 틀린 표현은 하단 입력의 D를 $\overline{D}$로 바꾼 부분이며, 올바르게는 $A, \overline{C}, D$이다.

③ X

중단 1단 NOR의 입력이 $\overline{A}, B, \overline{C}$이므로 식이 $F=(\overline{B}+D)(\overline{A}+B+\overline{C})(A+\overline{C}+D)$가 된다. 이 회로는 ABCD=1000, 1001에서 1이 되고 1010, 1011에서 0이 되어 표와 반대로 뒤집히는 부분이 생긴다. 틀린 표현은 중단 입력의 C를 $\overline{C}$로 둔 부분이며, 올바르게는 $\overline{A}, B, C$이다.

④ X

상단 1단 NOR의 입력이 $\overline{B}, \overline{D}$이므로 식이 $F=(\overline{B}+\overline{D})(\overline{A}+B+C)(A+\overline{C}+D)$가 된다. 이 회로는 ABCD=0100, 1100, 1110에서 1이 되고 0101, 0111, 1101, 1111에서 0이 되어 표와 많이 어긋난다. 틀린 표현은 상단 입력의 D를 $\overline{D}$로 둔 부분이며, 올바르게는 $\overline{B}, D$이다.

오답 포인트

마지막 NOR까지 보고도 각 1단 NOR의 출력을 다시 한 번 부정해야 한다는 점을 놓치면 식을 잘못 세우기 쉽다. 또 보수 표시 하나만 바뀌어도 0과 1이 뒤집히는 행이 즉시 생기므로, ②의 $\overline{D}$, ③의 $\overline{C}$, ④의 $\overline{D}$ 같은 미세한 차이를 끝까지 확인해야 한다.

다시 나오면 이렇게 판단

각 1단 NOR를 먼저 입력합 형태로 읽고, 마지막 NOR에서 드모르간 법칙을 적용해 곱의 합 식으로 바꾼 뒤, 진리표에서 0이 나오는 대표 행 몇 개만 대입해 일치 여부를 빠르게 확인하면 된다.

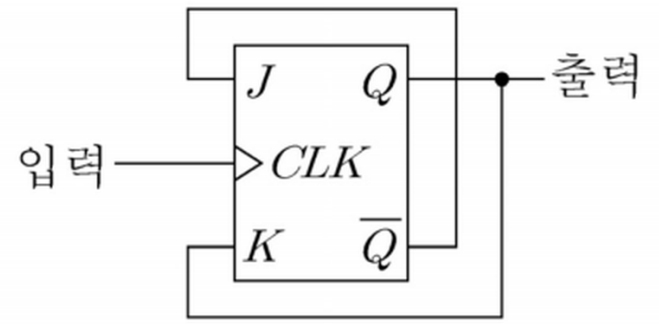
암기 포인트

NOR-NOR 2단 회로는 각 묶음의 합을 마지막에 곱으로 연결한 POS 형태로 읽는다.

NoteBase

문 6. <보기>에서 JK 플립플롭의 입력 신호 주파수(frequency)가 1[MHz]일 때 출력 신호의 주기(period)의 값[μsec]은?

<보기>



① 1

② 2

③ 4

④ 8

정답 2번

출제 포인트

이 문제는 회로도에서 JK 플립플롭의 피드백 연결을 읽어 $J = \overline{Q}$, $K = Q$가 되어 출력이 클록마다 토글되고, 따라서 2분주가 된다는 점을 판단하는 문제이다.

선지별 해설

① X: $1\mu sec$는 입력 주기와 같은 값이다. 이 회로는 출력이 클록마다 반전되므로 출력 주기는 입력 주기의 2배인 $2\mu sec$이다. 틀린 값 $1\mu sec$ → 올바르게는 $2\mu sec$이다.

② O: 회로도에서 $\overline{Q}$가 J로, Q가 K로 되먹임된다. 따라서 다음 상태는 항상 $Q^+ = \overline{Q}$가 되어 클록이 들어올 때마다 출력이 토글된다. 입력 주파수 $1MHz$의 주기는 $1\mu sec$이고, 출력 한 주기는 두 번의 클록이 필요하므로 $T_{out} = 2\mu sec$이다.

③ X: $4\mu sec$는 출력이 입력보다 4배 느려질 때의 값이다. 이 회로는 JK 플립플롭 1단의 토글 동작으로 2분주만 일어나므로 출력 주기는 $2\mu sec$이다. 틀린 값 $4\mu sec$ → 올바르게는 $2\mu sec$이다.

④ X: $8\mu sec$는 8분주로 오인한 값이다. 이 회로는 플립플롭이 한 개뿐이고 매 클록마다 상태가 한 번씩만 반전되므로 출력 주기는 $2\mu sec$이다. 틀린 값 $8\mu sec$ → 올바르게는 $2\mu sec$이다.

NoteBase

오답 포인트

입력 주파수 $1MHz$를 보고 입력 주기 $1\mu sec$만 계산한 뒤 그대로 정답으로 고르기 쉽다. 그러나 회로도에서 출력이 자기 피드백으로 매 클록마다 반전되는 토글 구조인지를 먼저 봐야 하며, 토글 플립플롭이면 주파수는 절반, 주기는 두 배가 된다.

다시 나오면 이렇게 판단

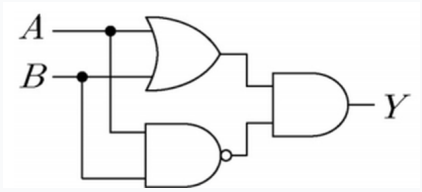
회로도에서 JK 입력이 출력의 피드백으로 연결되어 있으면 먼저 J와 K가 각각 Q와 $\overline{Q}$ 중 무엇에 연결되는지 확인한다. 다음 상태가 토글이면 $f_{out} = f_{in}/2$, $T_{out} = 2T_{in}$으로 바로 판단할 수 있다.

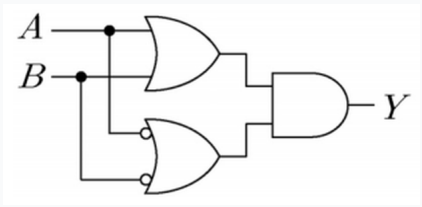
암기 포인트

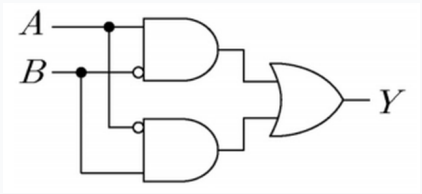
토글 동작 플립플롭 1단은 2분주이므로 주파수는 절반, 주기는 두 배이다.

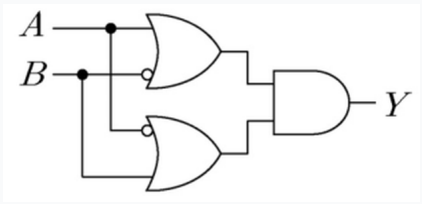
NoteBase

문 7. 논리게이트의 출력이 나머지와 다른 것은?

① 

② 

③ 

④ 

정답 4번

출제 포인트

각 회로를 입력 반전과 드모르간 법칙으로 식으로 바꾸어 XOR와 XNOR를 구별하는 문제이다.

선지별 해설

① O

위 OR 게이트는 $A+B$, 아래 NAND 게이트는 $\overline{AB}$이므로 전체 출력은 $(A+B)\overline{AB} = A\overline{B} + \overline{A}B$이다. XOR이므로 ②, ③과 같은 출력이다.

② O

위 OR 게이트는 $A+B$, 아래 게이트는 두 입력이 각각 반전된 OR이므로 $\overline{A}+\overline{B} = \overline{AB}$이다. 따라서 전체 출력은 $(A+B)(\overline{A}+\overline{B}) = A\overline{B} + \overline{A}B$이다. XOR이므로 ①, ③과 같은 출력이다.

③ O

위 AND 게이트는 $A\overline{B}$, 아래 AND 게이트는 $\overline{A}B$이고 이를 OR 하므로 $A\overline{B} + \overline{A}B$이다. 정의 그대로 XOR이다.

④ X

위 OR 게이트는 $A+\overline{B}$, 아래 OR 게이트는 $\overline{A}+B$이므로 전체 출력은 $(A+\overline{B})(\overline{A}+B) = AB + \overline{AB}$이다. 이는 XOR이 아니라 XNOR이다. 틀린 표현은 '나머지와 같은 XOR 출력'이고, 올바르게는 '입력이 같을 때 1이 되는 XNOR 출력'이다.

NoteBase

오답 포인트

입력단의 버블은 입력 반전이고 출력단의 버블은 출력 반전이다. ②는 모양이 달라도 드모르간 법칙으로 NAND와 같고, ④는 각 입력이 한 번씩 반전된 두 OR 출력을 AND하여 XNOR가 되는데 이를 XOR로 착각하기 쉽다.

다시 나오면 이렇게 판단

회로 비교 문제는 각 중간 게이트를 먼저 식으로 바꾼 뒤 마지막 게이트까지 전개하면 된다. 입력 버블은 해당 변수만 반전시키고, $(A+B)\overline{AB}$ 꼴이면 XOR, $(A+\overline{B})(\overline{A}+B)$ 꼴이면 XNOR로 판단된다.

암기 포인트

XOR는 '다르면 1'인 $A\overline{B}+\overline{A}B$, XNOR는 '같으면 1'인 $AB+\overline{A}\overline{B}$이다.

NoteBase

문 8. 8진수 $36.2_{(8)}$를 4진수로 변환시킨 값은?

① $132.0_{(4)}$

② $132.1_{(4)}$

③ $132.2_{(4)}$

④ $132.3_{(4)}$

정답 2번

출제 포인트

이 문제는 8진수를 2진수로 바꾼 뒤 2비트씩 묶어 4진수로 읽는 변환 원리를 묻는 문제이다.

조건 정리

주어진 값은 $36.2_{(8)}$이다.
구해야 할 값은 이 수를 4진수로 바꾼 결과이다.
주의할 조건은 정수부와 소수부를 모두 변환해야 하고, 소수부의 끝자리 0은 생략할 수 있다는 점이다.

사용할 공식

1자리 8진수는 3비트 2진수와 같고, 1자리 4진수는 2비트 2진수와 같다.
따라서 8진수를 2진수로 바꾼 뒤 소수점을 기준으로 양쪽을 2비트씩 묶으면 4진수로 바로 읽을 수 있다.

계산 과정

$36.2_{(8)} = 011110.010_{(2)}$이다.
앞의 불필요한 0을 빼면 $11110.010_{(2)}$이다.
소수점을 기준으로 2비트씩 묶으면 $01\ 11\ 10.01\ 00_{(2)}$이다.
각 묶음을 4진수 한 자리로 바꾸면 $132.10_{(4)}$이고, 끝자리 0은 생략할 수 있으므로 $132.1_{(4)}$이다.

선지별 해설

① X: 틀린 표현은 소수부가 0인 부분이다. $.2_{(8)} = .25_{(10)}$이므로 소수부가 없어질 수 없다. 올바르게는 $132.1_{(4)}$이다.
② O: $36.2_{(8)}$를 2진수로 바꾸면 $11110.010_{(2)}$이고, 이를 2비트씩 묶으면 $132.10_{(4)}$가 된다. 끝자리 0을 생략하면 $132.1_{(4)}$이므로 맞다.
③ X: 틀린 표현은 소수부가 2인 부분이다. $.2_{(4)} = .5_{(10)}$라서 원래 소수부 $.2_{(8)} = .25_{(10)}$보다 크다. 올바르게는 $132.1_{(4)}$이다.
④ X: 틀린 표현은 소수부가 3인 부분이다. $.3_{(4)} = .75_{(10)}$이므로 원래 값과 일치하지 않는다. 올바르게는 $132.1_{(4)}$이다.

NoteBase

오답 포인트

소수부를 빼먹으면 ①처럼 되고, 8진수의 소수 한 자리를 4진수 한 자리로 그대로 옮기면 ③처럼 틀린다. 이 문항의 함정은 소수부도 2진수로 바꿔 다시 묶어야 한다는 점이다.

다시 나오면 이렇게 판단

진법 변환에서 목표 진법이 2의 거듭제곱이면 먼저 2진수로 바꾼다. 그다음 소수점을 기준으로 목표 진법 자릿수에 맞게 묶어서 다시 읽으면 된다.

암기 포인트

8진수 1자리는 2진수 3비트, 4진수 1자리는 2진수 2비트이므로 8 → 4 변환은 2진수를 거치면 안전하다.

NoteBase

문 9. <보기>의 4변수 카르노 맵을 만족하는 최적화된 SOP(Sum of Product) 식은? (단, X는 부정의(don't care)를 뜻한다.)

<보기>

AB \ CD	00	01	11	10
00	1	1	0	1
01	0	0	0	0
11	1	1	X	X
10	1	1	X	X

① $A + \overline{BC}$

② $A + \overline{BC} + \overline{BD}$

③ $A\overline{C} + \overline{BC} + \overline{BD}$

④ $A\overline{C} + \overline{ABC} + \overline{AB}\overline{CD}$

정답 2번

출제 포인트

이 문제는 카르노 맵에서 don't care인 X를 1처럼 활용해 가장 큰 묶음을 만든 뒤 최소 SOP를 구할 수 있는지를 묻는다.

NoteBase

선지별 해설

1번 X: $A+\bar{B}\bar{C}$로는 $AB=00,\ CD=10$인 셀 m_2가 덮이지 않는다. $\bar{B}\bar{D}$ 항이 하나 더 필요하다. 올바르게는 $A+\bar{B}\bar{C}+\bar{B}\bar{D}$이다.
2번 O: 아래 두 행 $AB=11, 10$은 X를 포함해 8칸 묶음이 가능하므로 A가 된다. 또 $B=0,\ C=0$인 4칸 묶음으로 $\bar{B}\bar{C}$, $B=0,\ D=0$인 가장자리 연결 4칸 묶음으로 $\bar{B}\bar{D}$를 얻는다. 따라서 최적화된 SOP는 $A+\bar{B}\bar{C}+\bar{B}\bar{D}$이다.
3번 X: $\bar{A}C$는 $A=0,\ C=1$인 경우를 모두 1로 만들어 원래 0인 m_3, m_6, m_7까지 포함한다. 또한 아래쪽 1들과 X를 한 번에 묶어 얻는 A를 놓쳤다. 올바르게는 $A+\bar{B}\bar{C}+\bar{B}\bar{D}$이다.
4번 X: ④도 첫 항이 $\bar{A}C$라서 0이어야 할 셀을 1로 켠다. 또한 X를 이용하면 아래쪽 절반을 한 항 A로 줄일 수 있는데, ④는 그렇게 최소화되지 않았다. 올바르게는 $A+\bar{B}\bar{C}+\bar{B}\bar{D}$이다.

오답 포인트

열 10과 00이 카르노 맵에서 서로 인접한다는 점과 X를 1처럼 써서 더 큰 묶음을 만드는 점을 놓치면 ①처럼 m_2를 빼먹거나 ③, ④처럼 불필요한 항을 만든다.

다시 나오면 이렇게 판단

먼저 X를 포함해 8칸, 4칸 순으로 최대 묶음을 찾는다. 그다음 행과 열의 양 끝도 서로 인접하므로 10과 00을 이어 보고, 마지막에 남는 1이 없는지 확인한다.

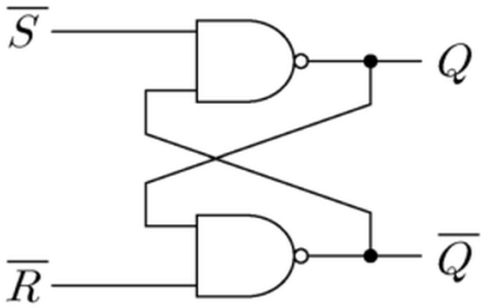
암기 포인트

카르노 맵 최소화는 X를 활용한 최대 묶음이 우선이며, 8칸은 1변수 항, 4칸은 2변수 항으로 줄어든다.

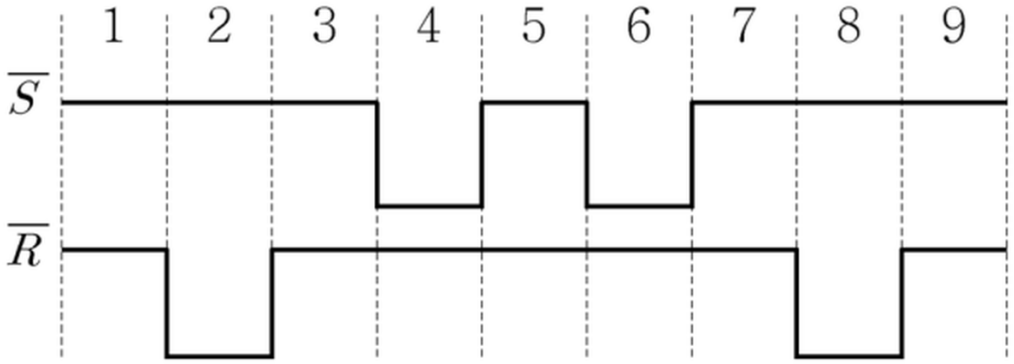
NoteBase

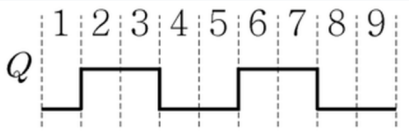
문 10. <보기 1>의 NAND 게이트 S-R 래치에 <보기 2>와 같은 파형을 인가하였을 때, 출력 Q의 파형에 가장 가까운 것은? (단, Q는 0으로 초기화되어 있으며, 게이트에서의 지연은 없다고 가정한다.)

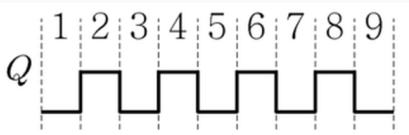
<보기 1>

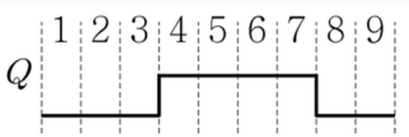


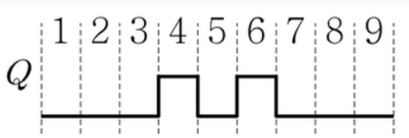
<보기 2>



① 

② 

③ 

④ 

정답 3번

NoteBase

출제 포인트

이 문제는 NAND형 S-R 래치가 활성 Low 입력이라는 점을 이용해, $\overline{S}=0$이면 Q가 1로 set되고 $\overline{R}=0$이면 Q가 0으로 reset되며 나머지 구간에서는 직전 상태를 유지한다는 점을 파형에 적용하는 문제이다.

선지별 해설

① X: 1~2 구간에서 $\overline{R}=0$이므로 Q는 1이 아니라 0으로 reset되어야 한다. 이 선지는 1에서 Q가 상승하는 점이 틀렸다. 올바른 파형은 3에서 1로 상승하고 7에서 다시 0으로 하강한다.
② X: 이 선지는 $\overline{S}$나 $\overline{R}$가 변할 때마다 Q가 매 구간 토글되는 것처럼 그렸다. S-R 래치는 토글 회로가 아니라 set, reset, hold 회로이므로 3~7 구간에서는 1이 계속 유지되어야 한다. 올바른 파형은 0-3 구간 0, 3-7 구간 1, 7-9 구간 0이다.
③ O: 초기값이 Q=0이고, 1~2 구간의 $\overline{R}=0$에서 reset이 한 번 더 걸려도 Q는 계속 0이다. 3~4 구간의 $\overline{S}=0$에서 Q가 1이 되고, 4~7 구간은 hold 상태라 1이 유지된다. 7~8 구간의 $\overline{R}=0$에서 다시 0이 되므로 제시된 파형이 맞다.
④ X: 3~4 구간과 5~6 구간에서만 $\overline{S}=0$이라는 이유로 Q를 순간 펄스로 그렸는데, NAND형 래치는 set 후 입력이 1로 돌아와도 hold 상태에서 값을 유지한다. 따라서 4~5 구간과 6~7 구간이 0이 아니라 1이어야 한다. 올바른 파형은 3~7 구간 연속 High이다.

오답 포인트

NAND형 S-R 래치는 입력에 bar가 붙은 활성 Low 회로라는 점이 함정이다. $\overline{S}=0$과 $\overline{R}=0$을 각각 set, reset으로 읽지 못하면 ①처럼 reset 구간에서 Q를 올리거나 ④처럼 set 구간을 순간 펄스로만 해석하게 된다.

다시 나오면 이렇게 판단

먼저 래치가 NOR형인지 NAND형인지와 입력의 활성 레벨을 확인한다. 그다음 시간축을 따라 $\overline{S}=0$이면 Q=1, $\overline{R}=0$이면 Q=0, 둘 다 1이면 직전 상태 유지라는 규칙만 순서대로 적용하면 된다.

암기 포인트

NAND S-R 래치는 입력이 active Low이므로 $\overline{S}=0$ set, $\overline{R}=0$ reset, 1,1 hold이다.

NoteBase

문 11. 진리표가 <보기>와 같을 때, 출력 F를 최대로 간략화한 논리식은?

<보기>

입력			출력
C	B	A	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

① $F = AB + BC$

② $F = \overline{A}B + AC$

③ $F = A\overline{C} + BC$

④ $F = AB + A\overline{C} + BC$

정답 3번

출제 포인트

이 문항은 진리표에서 출력이 1이 되는 행을 묶어 최소 곱의 합 형태로 간략화하는 문제이다.

조건 정리

주어진 값: 입력 변수는 C, B, A이고 출력 F는 진리표로 제시되어 있다.
구해야 할 값: F를 최대로 간략화한 논리식이다.
주의할 조건: 같은 출력 1이라도 묶을 때 변화하지 않는 변수만 남겨야 하며, 같은 기능을 하는 중복항은 제거해야 한다.

NoteBase

사용할 공식

$F = \sum m(1, 3, 6, 7)$에서 인접한 항을 묶어 공통 변수만 남긴다. 따라서 001,011은 $A\overline{C}$, 110,111은 BC로 묶인다.
이 공식을 쓰는 이유: 진리표에서 F가 1인 행을 최소항으로 바꾼 뒤 인접한 항을 결합하면 가장 짧은 곱의 합 형태를 얻을 수 있기 때문이다.

계산 과정

1. F가 1인 행은 $(C, B, A) = (0, 0, 1), (0, 1, 1), (1, 1, 0), (1, 1, 1)$이다.
2. $(0, 0, 1)$과 $(0, 1, 1)$은 C=0, A=1이 공통이므로 $A\overline{C}$로 묶인다.
3. $(1, 1, 0)$과 $(1, 1, 1)$은 C=1, B=1이 공통이므로 BC로 묶인다.
4. 따라서 $F = A\overline{C} + BC$이다.

선지별 해설

① X. $F = AB + BC$는 $(C, B, A) = (0, 0, 1)$일 때 0이 되지만 진리표의 F는 1이다. 틀린 표현은 AB이고, 올바르게는 $A\overline{C}$가 들어가야 하므로 $F = A\overline{C} + BC$가 된다.
② X. $F = \overline{A}B + AC$는 $(C, B, A) = (0, 0, 1)$일 때 0이 되어 진리표와 맞지 않는다. 틀린 표현은 $\overline{A}B$와 AC의 조합이고, 올바르게는 $A\overline{C}$와 BC를 남겨야 한다.
③ O. $F = A\overline{C} + BC$는 C=0인 행에서는 A값만, C=1이면서 B=1인 행에서는 B와 C값만 반영하므로 표의 1 위치와 정확히 일치한다.
④ X. $F = AB + A\overline{C} + BC$는 기능은 같지만 최대로 간략화한 식은 아니다. AB는 C=0이면 $A\overline{C}$에, C=1이면 BC에 이미 포함되어 중복항이 된다. 틀린 표현은 추가된 AB이고, 올바르게는 이를 제거한 $F = A\overline{C} + BC$이다.

오답 포인트

출력이 1인 행만 보고 식을 적은 뒤 흡수되는 항을 끝까지 지우지 않으면 ④처럼 동치이지만 최소식이 아닌 선지에 끌리기 쉽다. 또 A가 1인 두 행을 보고 AB로 잘못 읽으면 ①처럼 C 조건을 놓치게 된다.

다시 나오면 이렇게 판단

진리표 간략화 문항은 먼저 F=1인 행만 표시하고, 한 변수만 달라지는 행끼리 묶어 공통 변수만 남긴다.
마지막에는 남은 항 중 다른 항에 완전히 포함되는 중복항이 있는지 다시 확인하면 된다.

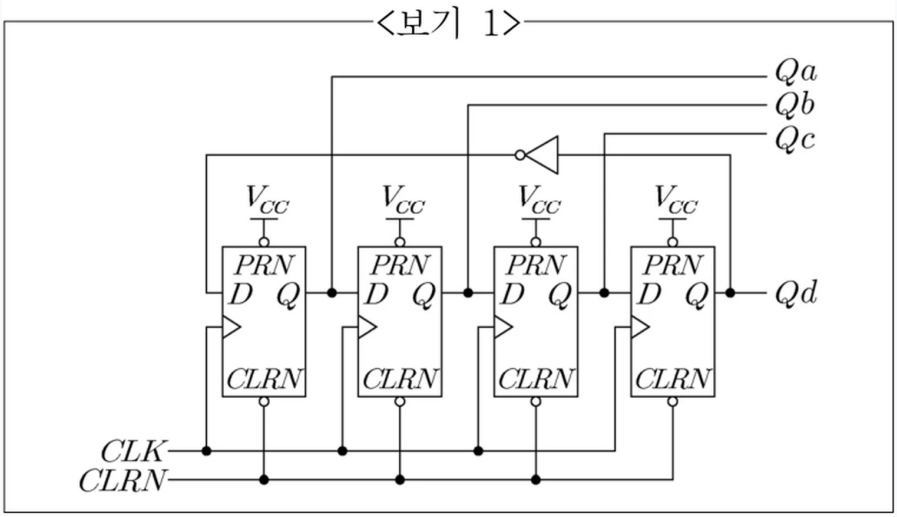
암기 포인트

진리표 최소화는 1이 되는 행을 인접 묶음으로 합치고, 묶음에서 변하는 변수는 지운다는 원리로 정리된다.

NoteBase

문 12. <보기 1>의 회로에서 $CLRN = 1$이며, 현재 출력은 $Q_aQ_bQ_cQ_d = 0000$이다. <보기 2>의 괄호 안에 들어갈 숫자는?

<보기 1>



<보기 2>

CLK의 ( )번째 상승모서리에서 $QaQbQcQd=1111$이 된다.

① 3

② 4

③ 5

④ 6

정답 2번

출제 포인트

이 문제는 4개의 D 플립플롭에 마지막 출력의 보수를 되먹임한 존슨 카운터의 상태 전이 순서를 묻는 문제이다.

선지별 해설

① X: 3번째 상승모서리 뒤 상태는 QaQbQcQd=1110이다. 1111이 되는 시점은 4번째 상승모서리이다.
② O: 이 회로는 마지막 출력 Qd의 보수를 첫 번째 D 입력으로 되먹임한 4비트 존슨 카운터이다. 초기 상태 0000에서 상승모서리마다 1000 → 1100 → 1110 → 1111로 진행하므로 4번째에서 1111이 된다.
③ X: 5번째 상승모서리에서는 첫 비트에 0이 들어와 QaQbQcQd=0111이 된다. 5번째가 아니라 4번째가 맞다.
④ X: 6번째 상승모서리에서는 QaQbQcQd=0011이 된다. 1111 상태는 이미 4번째 상승모서리에서 지난다.

NoteBase

오답 포인트

직렬 연결된 D 플립플롭만 보고 단순 시프트 레지스터로 판단하면 틀리기 쉽다. 마지막 출력의 반전이 첫 단계 입력으로 돌아가는 되먹임이 보이면 일반 링 카운터가 아니라 존슨 카운터이고, 1이 왼쪽부터 차례로 채워진다.

다시 나오면 이렇게 판단

D 플립플롭이 직렬로 연결되고 마지막 출력이 반전되어 첫 입력으로 돌아가면 존슨 카운터로 본다. 초기상태에서 한 번의 클록마다 다음 상태를 한 칸씩 적어 보면 정답 시점을 바로 찾을 수 있다.

암기 포인트

4비트 존슨 카운터는 0000에서 시작하면 1000 → 1100 → 1110 → 1111 순으로 1이 채워진다.

NoteBase

문 13. <보기>의 부울 함수를 최소항의 합으로 올바르게 나타낸 것은?

<보기>

$F(A, B, C) = A + BC$

① $F(A, B, C) = A\bar{B}C + AB\bar{C} + ABC + \bar{A}BC$

② $F(A, B, C) = A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC$

③ $F(A, B, C) = A\bar{B}\bar{C} + A\bar{B}C + ABC + \bar{A}BC$

④ $F(A, B, C) = A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC + \bar{A}BC$

정답 4번

출제 포인트

이 문제는 $F(A, B, C) = A + BC$를 모든 변수 A, B, C가 들어간 최소항의 합으로 전개할 수 있는지를 묻는다.

조건 정리

주어진 함수: $F(A, B, C) = A + BC$
구해야 할 값: 최소항의 합 형태
주의할 조건: 최소항은 각 항마다 A, B, C가 모두 한 번씩 나타나야 한다.

사용할 공식

$X = X(Y + \bar{Y}), BC = BC(A + \bar{A})$이다.
이 공식을 쓰는 이유는 빠진 변수를 $1 = Y + \bar{Y}$ 형태로 보충해야 각 항이 최소항이 되기 때문이다.

계산 과정

1. $A = A(\bar{B} + B)(\bar{C} + C)$로 전개한다.
2. 따라서 $A = A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC$이다.
3. 또 $BC = (\bar{A} + A)BC = \bar{A}BC + ABC$이다.
4. 중복된 ABC를 한 번만 남기면 $F(A, B, C) = A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC + \bar{A}BC$가 된다.

NoteBase

선지 판단

① X: $A\bar{B}\bar{C}$가 빠져 있다. A를 최소항으로 만들면 B와 C의 네 조합이 모두 나와야 하므로, 올바르게는 $A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC + \bar{A}BC$가 된다.
② X: $\bar{A}BC$가 빠져 있다. BC는 $(\bar{A} + A)BC$로 보충되어야 하므로, 올바르게는 $\bar{A}BC$가 추가되어야 한다.
③ X: $AB\bar{C}$가 빠져 있다. A를 $(\bar{B} + B)(\bar{C} + C)$로 전개하면 $AB\bar{C}$도 반드시 포함되므로, 올바르게는 $AB\bar{C}$가 추가되어야 한다.
④ O: A에서 나온 네 개의 최소항 $A\bar{B}\bar{C}, A\bar{B}C, AB\bar{C}, ABC$와 BC에서 추가되는 $\bar{A}BC$를 모두 포함하므로 올바른 최소항의 합이다.

오답 포인트

최소항의 합은 각 항에 모든 변수가 들어가야 한다는 조건을 놓치면 전개 과정에서 한 항만 빠뜨리는 실수를 하게 된다. 특히 BC에 남아 있는 변수 A를 보충하지 않으면 $\bar{A}BC$를 놓치기 쉽다.

다시 나오면 이렇게 판단

한 항에 빠진 변수가 있으면 $1 = X + \bar{X}$를 곱해 변수를 모두 보충한 뒤, 나온 조합이 중복 없이 모두 들어갔는지 확인하면 된다.

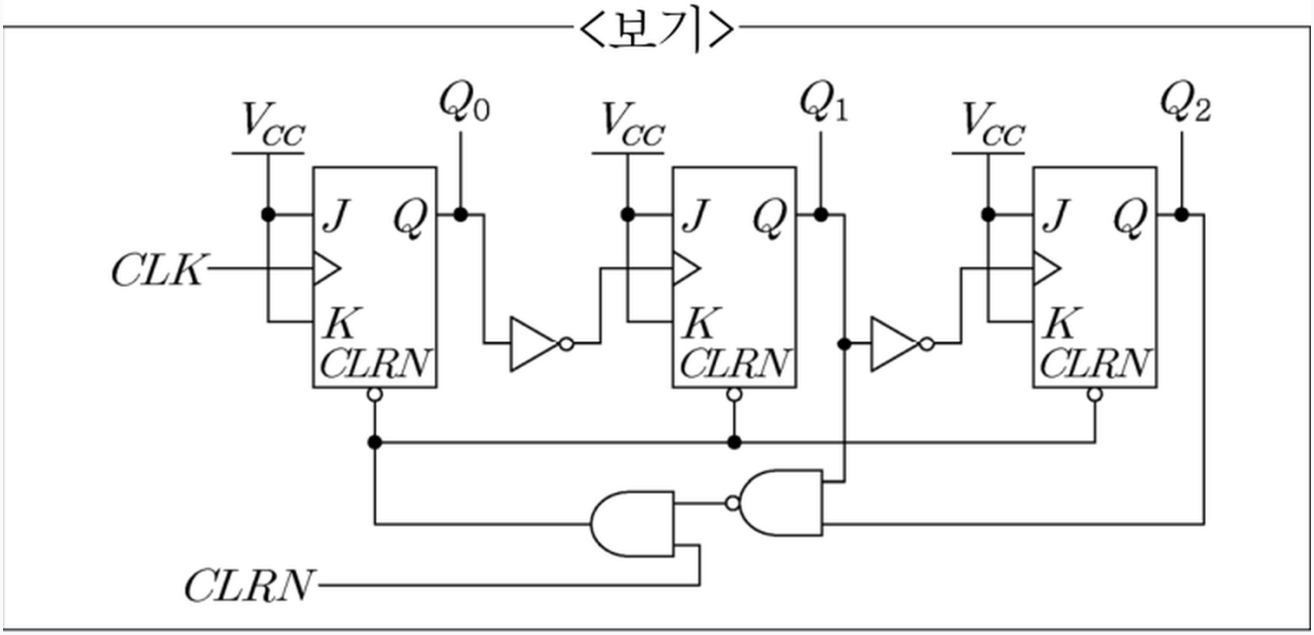
암기 포인트

단일 변수 항은 빠진 변수 두 개를 보충해 4개의 최소항이 되고, 두 변수 항은 남은 변수 하나를 보충해 2개의 최소항이 된다.

NoteBase

문 14. <보기>의 회로에서 $CLRN = 1$이며, 현재 출력은 $Q_2Q_1Q_0 = 000$이다. 이후 CLK의 cycle이 9개 발생했을 때 $Q_2Q_1Q_0$의 값은?

<보기>



① $Q_2Q_1Q_0 = 000$

② $Q_2Q_1Q_0 = 001$

③ $Q_2Q_1Q_0 = 010$

④ $Q_2Q_1Q_0 = 011$

정답 4번

출제 포인트

이 문제는 JK 플립플롭 3개가 만든 비동기 리플 카운터에서 Q_1과 Q_2가 동시에 1이 되면 CLRN이 활성화되어 즉시 000으로 초기화된다는 점을 이용해 카운트 순서를 추적하는 문제이다.

NoteBase

선지별 해설

① X

- 000은 6번째 CLK에서 $101 \rightarrow 110$으로 넘어가려는 순간 $Q_1 = Q_2 = 1$이 되어 비동기 클리어가 걸린 뒤의 값이다.
- 틀린 표현은 "9개 발생 후 000"이다. 올바르게는 6번째 CLK 직후 000으로 클리어되고, 9번째 후에는 011이 된다.

② X

- 001은 이 회로를 단순 3비트 이진 카운터로 오해했을 때 나오는 값이다. 실제로는 Q_1과 Q_2 검출 클리어 때문에 8진 카운터가 아니라 6진 카운터로 동작한다.
- 틀린 표현은 "9개 발생 후 001"이다. 올바르게는 7번째 CLK 후가 001이고 9번째 후는 011이다.

③ X

- 010은 클리어 후 두 번째 펄스까지의 상태이다. 실제 순서는
$000 \rightarrow 001 \rightarrow 010 \rightarrow 011 \rightarrow 100 \rightarrow 101 \rightarrow 000 \rightarrow 001 \rightarrow 010 \rightarrow 011$이다.
- 틀린 표현은 "9개 발생 후 010"이다. 올바르게는 8번째 CLK 후가 010이고 9번째 후는 011이다.

④ O

- J와 K가 모두 1이므로 각 플립플롭은 토글한다. 아래 조합논리가 $Q_1 = 1$이고 $Q_2 = 1$인 상태를 검출하면 공통 CLRN을 0으로 만들어 비동기 초기화한다. 따라서 전체 카운트는 mod-6이 되고, 9번째 CLK 후 상태는 011이다.

오답 포인트

가장 흔한 오답은 이 회로를 일반 3비트 리플 카운터로 보고 9를 8로 나눈 나머지만 적용해 001을 고르는 경우이다. 이 문항의 함정은 아래의 비동기 클리어 회로를 놓치게 만드는 데 있으며, Q_1과 Q_2가 동시에 1이 되는 순간 110이나 111을 유지하지 못하고 바로 000으로 돌아간다.

다시 나오면 이렇게 판단

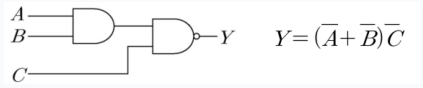
리플 카운터 문제에서는 플립플롭 연결만 보지 않고 먼저 J, K 값과 CLRN 같은 비동기 입력의 조합논리를 확인한다. 그다음 실제 상태열을 짧게 써서 mod 값이 8인지, 그보다 작은지 판단한다.

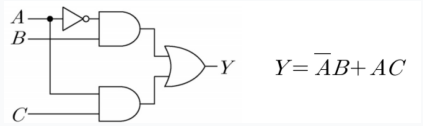
암기 포인트

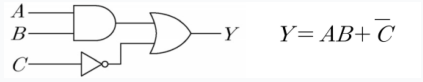
리플 카운터에 출력 검출 CLRN이 붙으면 먼저 mod 값을 본다. 이 회로는 $Q_1Q_2 = 11$ 검출로 6진 카운터가 된다.

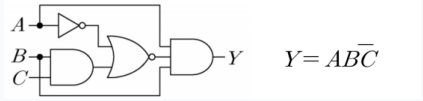
NoteBase

문 15. 논리 게이트의 출력의 논리식 Y를 나타내었을 때 가장 옳지 않은 것은?

① 

② 

③ 

④ 

정답 1번

출제 포인트

이 문제는 회로도에서 AND, OR, NAND, NOR, NOT의 연결 순서를 따라가며 최종 출력식을 정확히 세우고 드모르간 법칙으로 확인할 수 있는지를 묻는다.

선지별 해설

1번: X. 첫 번째 게이트의 출력은 AB이고, 두 번째 게이트는 출력에 버블이 있는 AND이므로 NAND이다. 따라서 최종 출력은 $Y=\overline{AB\cdot C}=\overline{ABC}$가 맞다. 제시식 $(\overline{A}+\overline{B})\overline{C}$는 전체 ABC를 한 번에 부정한 식이 아니다. 틀린 표현은 $(\overline{A}+\overline{B})\overline{C}$이고, 올바르게는 $\overline{ABC}=\overline{A}+\overline{B}+\overline{C}$이다.
2번: O. 위쪽 경로는 A가 반전되어 $\overline{A}$가 되고 이것이 B와 AND되므로 $\overline{A}B$이다. 아래쪽 경로는 A와 C가 AND되어 AC가 된다. 두 출력을 OR하므로 $Y=\overline{A}B+AC$가 맞다.
3번: O. 윗쪽 AND의 출력은 AB이고, C는 NOT을 지나 $\overline{C}$가 된다. 이 둘을 OR하므로 $Y=AB+\overline{C}$가 된다.
4번: O. B와 C는 먼저 AND되어 BC가 되고, A는 반전되어 $\overline{A}$가 된다. 가운데 게이트는 OR 출력에 버블이 있는 NOR이므로 그 출력은 $\overline{\overline{A}+BC}$이다. 마지막 게이트는 A, B, $\overline{\overline{A}+BC}$를 AND하므로 $Y=A\cdot B\cdot\overline{\overline{A}+BC}=AB\overline{C}$가 되어 제시식과 일치한다.

오답 포인트

1번은 NAND를 보고도 마지막 버블이 전체 출력에 걸린 부정이라는 점을 놓치면 틀리기 쉽다. $\overline{ABC}$를 $(\overline{A}+\overline{B})\overline{C}$처럼 일부 항만 묶어 쓰면 드모르간 법칙을 잘못 적용한 형태가 된다.

다시 나오면 이렇게 판단

회로 문제는 입력부터 차례로 각 게이트의 중간 출력을 먼저 적고, 출력 버블이 있는 게이트는 마지막에 전체 식을 한 번 부정하는 방식으로 처리하면 된다. 드모르간 법칙은 괄호 전체에 적용해야 오류가 줄어든다.

NoteBase

암기 포인트

NAND는 AND 전체를 부정한 식이고, $\overline{ABC} = \overline{A} + \overline{B} + \overline{C}$처럼 곱의 부정은 부정들의 합으로 바뀐다.

NoteBase

문 16. 부호가 있는 4비트 2의 보수(two's complement) 수체계에서 A[3:0]=1111, B[3:0]=0110이라고 할 때, A[3:0]-B[3:0]의 연산 결과를 10진수로 나타낸 값은?

① -4

② -5

③ -6

④ -7

정답 4번

출제 포인트

이 문제는 4비트 2의 보수 표현을 10진수로 해석한 뒤 뺄셈 결과를 구할 수 있는지를 묻는 문제이다.

조건 정리

주어진 값: A[3:0]=1111, B[3:0]=0110이다.
구해야 할 값: A[3:0]-B[3:0]의 10진수 값이다.
주의할 조건: 4비트 2의 보수에서 1111은 -1이고, 0110은 +6이다.

사용할 공식

2의 보수 수의 뺄셈은 각 수를 10진수로 해석해 직접 계산하거나, $A - B = A + (-B)$로 바꾸어 계산한다.
이 공식을 쓰는 이유: 문제는 부호가 있는 4비트 2의 보수 수체계에서의 뺄셈 결과를 묻기 때문이다.

계산 과정

1. A[3:0]=1111은 4비트 2의 보수 표현에서 -1이다.
2. B[3:0]=0110은 +6이다.
3. 따라서 A[3:0]-B[3:0]=-1-6=-7이다.
4. 비트 연산으로 확인하면 B의 2의 보수는 1010이고, 1111+1010=1 1001이므로 자리올림을 버리면 1001이다. 1001은 4비트 2의 보수에서 -7이다.

NoteBase

선지별 해설

① X: 10진수 결과는 -7이므로 "-4"는 틀리다. 틀린 표현은 "결과가 -4이다"이고, 올바르게는 "결과는 -7이다"이다.
② X: 10진수 결과는 -7이므로 "-5"는 틀리다. 틀린 표현은 "결과가 -5이다"이고, 올바르게는 "결과는 -7이다"이다.
③ X: 10진수 결과는 -7이므로 "-6"은 틀리다. 틀린 표현은 "결과가 -6이다"이고, 올바르게는 "결과는 -7이다"이다.
④ O: A=-1, B=+6이므로 A[3:0]-B[3:0]=-7이다. 비트 결과 1001도 10진수로 -7이므로 맞는 선택지이다.

오답 포인트

1111을 큰 양수처럼 읽거나 2의 보수에서 음수 해석을 빼먹으면 오답이 나온다. 또한 뺄셈을 할 때 B의 2의 보수를 취한 뒤 최종 4비트만 남긴다는 점을 놓치기 쉽다.

다시 나오면 이렇게 판단

2의 보수 계산 문제는 먼저 각 비트를 부호 있는 10진수 값으로 해석한 뒤 연산하고, 마지막에 비트 결과로 한 번 더 검산하는 순서로 판단한다.

암기 포인트

4비트 2의 보수에서 1111은 -1, 1001은 -7이다. 뺄셈은 피감수에 감수의 2의 보수를 더하는 방식으로도 확인할 수 있다.

NoteBase

문 17. 16진수 A8을 그레이 코드(gray code)로 변환시킨 값은?

① $(11001100)_{gray}$

② $(11111111)_{gray}$

③ $(11111100)_{gray}$

④ $(11001111)_{gray}$

정답 3번

출제 포인트

이 문제는 2진수 값을 그레이 코드로 바꾸는 원리, 즉 최상위 비트는 그대로 두고 나머지는 인접 비트의 XOR로 구하는지를 묻는다.

조건 정리

주어진 값: $A8_{16}$
구해야 할 값: 8비트 그레이 코드
주의할 조건: 먼저 $A8_{16}$를 10101000_2로 바꾼 뒤, 최상위 비트는 그대로 두고 나머지 비트는 바로 왼쪽 비트와 XOR한다.

사용할 공식

그레이 코드는 $Gray = Binary \oplus (Binary >> 1)$로 구할 수 있다.
이 공식을 쓰는 이유: 그레이 코드는 각 자리를 원래 비트와 그 왼쪽 비트의 XOR로 정하는 코드이기 때문이다.

계산 과정

1. $A8_{16}$를 2진수로 바꾸면 10101000_2이다.
2. 이를 한 비트 오른쪽으로 이동하면 01010100_2이다.
3. 두 값을 XOR하면 $10101000_2 \oplus 01010100_2 = 11111100_2$이다.
4. 따라서 그레이 코드는 11111100_2이다.

NoteBase

선지 판단

① X: 제시값은 11001100_2이다. 실제 계산 결과는 11111100_2이므로 가운데 비트 판단이 틀렸다. 틀린 표현은 11001100_2이고, 올바르게는 11111100_2이다.
② X: 제시값은 11111111_2이다. 계산 결과의 마지막 두 비트는 00이어야 하므로 맞지 않는다. 틀린 표현은 11111111_2이고, 올바르게는 11111100_2이다.
③ O: 계산 결과가 11111100_2이므로 옳다.
④ X: 제시값은 11001111_2이다. 실제 결과 11111100_2와 중간 및 마지막 비트가 다르다. 틀린 표현은 11001111_2이고, 올바르게는 11111100_2이다.

오답 포인트

그레이 코드 변환에서 최상위 비트는 그대로 두고, 나머지를 왼쪽 이웃 비트와 XOR해야 한다는 순서를 놓치면 오답이 나온다. 특히 각 자리의 XOR를 끝까지 일관되게 적용하지 않으면 ①, ④ 같은 오답이 나온다.

다시 나오면 이렇게 판단

그레이 코드 문제는 16진수를 먼저 2진수로 바꾸고, 최상위 비트 유지 -> 오른쪽으로 한 칸 민 값과 XOR -> 나온 결과를 선지와 대조하는 순서로 판단한다.

암기 포인트

그레이 코드는 맨 앞 비트는 그대로 두고, 나머지는 왼쪽 비트와 XOR한 값이다.

NoteBase

문 18. 논리게이트의 전기적 특성에 대한 설명으로 가장 옳지 않은 것은?

① 신호가 입력되어서 출력될 때까지 소요되는 시간을 지연시간이라고 하며, 이는 게이트의 동작 속도를 나타낸다.

② 게이트에서 데이터의 값에 변화를 주지 않는 최대 허용 잡음의 크기를 잡음이득이라고 한다.

③ 하나의 게이트가 구동할 수 있는 부하 게이트의 입력의 수를 팬-아웃(fan-out)이라 하며, 하나의 출력이 최대 몇 개의 입력으로 연결될 수 있는지를 나타낸다.

④ 싱크전류(sink current)는 출력 쪽으로 전류가 흘러 들어가는 것이며, 소스전류(source current)는 출력에서 바깥으로 전류가 흘러 나오는 것이다.

정답 2번

출제 포인트

논리게이트의 전기적 특성인 지연시간, 잡음여유, 팬아웃, 싱크전류와 소스전류의 정의를 정확히 구별하는 문제이다.

핵심 개념

지연시간은 입력 변화가 출력에 반영되기까지 걸리는 시간으로 게이트의 동작 속도를 나타낸다.
잡음여유는 잡음이 섞여도 논리값이 잘못 바뀌지 않는 최대 허용 범위이다.
팬아웃은 한 게이트의 출력이 구동할 수 있는 부하 게이트 입력의 최대 개수이다.
싱크전류는 출력단 안으로 들어가는 전류이고, 소스전류는 출력단 밖으로 나가는 전류이다.

NoteBase

선지별 해설

① O
신호가 입력된 뒤 출력이 반응할 때까지의 시간을 지연시간이라고 하며, 값이 작을수록 게이트가 빠르게 동작한다는 뜻이므로 옳은 설명이다.

② X
틀린 부분은 "최대 허용 잡음의 크기를 잡음이득이라고 한다"는 표현이다.
게이트에서 데이터 값에 변화를 주지 않는 최대 허용 잡음의 크기는 잡음여유(noise margin) 또는 잡음내성에 해당한다. 올바르게는 "게이트에서 데이터의 값에 변화를 주지 않는 최대 허용 잡음의 크기를 잡음여유라고 한다"이다.

③ O
팬아웃은 한 출력이 몇 개의 입력을 구동할 수 있는지를 나타내는 값이다. 하나의 출력이 연결 가능한 최대 입력 수를 뜻하므로 옳은 설명이다.

④ O
싱크전류는 출력단이 전류를 받아들이는 경우이고, 소스전류는 출력단이 외부로 전류를 내보내는 경우이다. 전류 방향 설명이 정확하므로 옳은 설명이다.

오답 포인트

잡음여유와 잡음이득처럼 용어가 비슷해 보이는 개념을 섞어 놓은 점이 함정이다. 디지털 논리회로에서는 잡음 허용 범위를 잡음여유로 기억해야 하며, 팬아웃과 전류 방향 개념은 정의만 정확히 알면 바로 걸러진다.

다시 나오면 이렇게 판단

논리게이트 특성 문제는 먼저 시간 특성인지, 잡음 허용 범위인지, 연결 가능한 입력 수인지, 전류 방향인지로 분류하면 된다. 용어를 기능별로 대응시키면 비슷한 표현을 섞은 오답을 빠르게 제거할 수 있다.

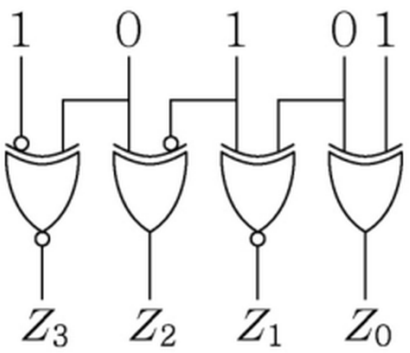
암기 포인트

지연시간은 속도, 잡음여유는 허용 잡음, 팬아웃은 연결 개수, 싱크는 안으로·소스는 밖으로이다.

NoteBase

문 19. <보기>의 논리회로에서 각 입력에 대한 출력 $Z_3Z_2Z_1Z_0$은? (단, Z_0는 LSB를 뜻한다.)

<보기>



① 1001

② 1011

③ 1111

④ 1110

정답 1번

출제 포인트

이 문제는 XOR 기호에 붙은 입력 버블과 출력 버블이 반전을 뜻한다는 점을 이용해, 각 게이트를 XOR 또는 XNOR로 바꾸어 $Z_3Z_2Z_1Z_0$를 읽는 문제이다.

핵심 정리

입력 버블 1개와 출력 버블 1개가 함께 있는 XOR 게이트는 결과적으로 일반 XOR와 같아진다.
입력 버블만 있거나 출력 버블만 있는 XOR 게이트는 결과적으로 XNOR와 같아진다.

NoteBase

선지별 해설

① O: 첫 번째 게이트는 $\overline{\overline{1} \oplus 0} = 1$, 두 번째 게이트는 $\overline{0} \oplus 1 = 0$, 세 번째 게이트는 $\overline{1 \oplus 0} = 0$, 네 번째 게이트는 $0 \oplus 1 = 1$이다. 따라서 $Z_3Z_2Z_1Z_0 = 1001$이므로 옳다.
② X: "1011"이 되려면 $Z_1 = 1$이어야 하는데, 세 번째 게이트는 출력 버블이 있는 XOR이므로 XNOR처럼 동작하여 1과 0의 결과가 0이 된다. 틀린 표현은 "세 번째 출력이 1"이라는 점이며, 올바르게는 $Z_1 = 0$이어서 1001이다.
③ X: "1111"이 되려면 $Z_2 = 1$이고 $Z_1 = 1$이어야 하는데 둘 다 틀리다. 두 번째 게이트는 입력 버블이 있는 XOR이라 XNOR처럼 동작하여 0과 1의 결과가 0이고, 세 번째 게이트도 출력 버블 때문에 0이 된다. 올바르게는 1001이다.
④ X: "1110"이 되려면 $Z_2 = 1$이고 $Z_0 = 0$이어야 하는데 둘 다 틀리다. 두 번째 게이트의 실제 출력은 0이고, 네 번째 게이트는 일반 XOR이므로 $0 \oplus 1 = 1$이다. 올바르게는 1001이다.

오답 포인트

버블을 본 뒤 바로 숫자를 대입하지 않고, 먼저 게이트 전체가 XOR인지 XNOR인지 정리해야 한다는 점이 함정이다. 특히 입력 버블만 있는 두 번째 게이트와 출력 버블만 있는 세 번째 게이트를 일반 XOR로 잘못 보면 ②, ③, ④처럼 판단하기 쉽다.

다시 나오면 이렇게 판단

각 게이트에서 버블의 개수를 먼저 본다. XOR에 반전이 홀수 번 걸리면 XNOR, 짝수 번 걸리면 XOR로 정리한 뒤 각 입력쌍 $(1, 0), (0, 1), (1, 0), (0, 1)$을 순서대로 대입하면 된다.

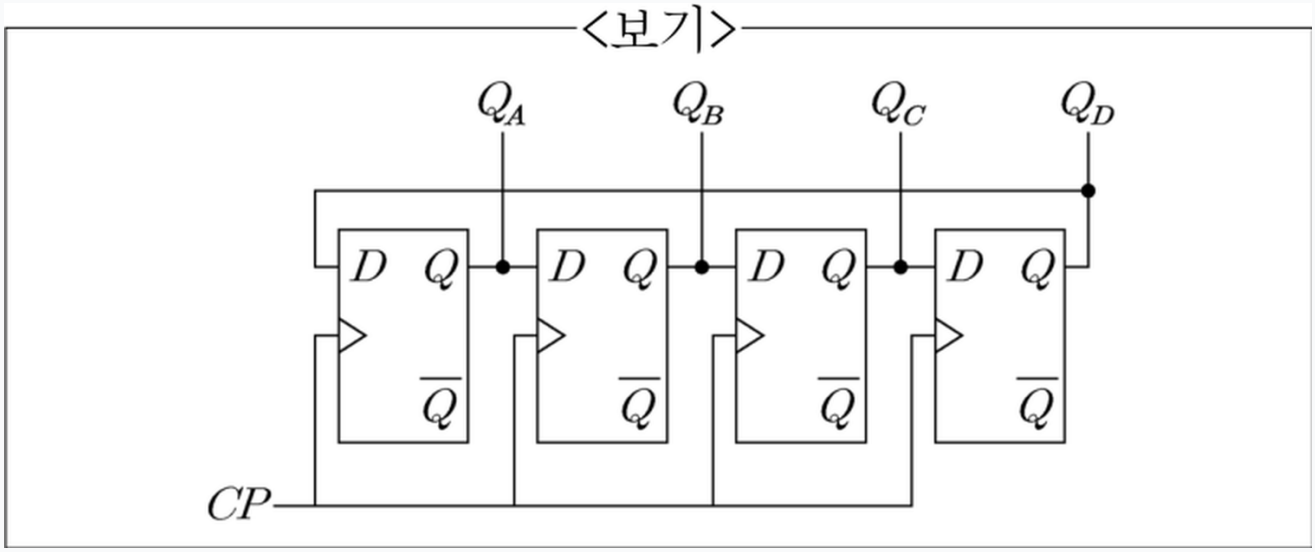
암기 포인트

XOR는 홀수 번 반전되면 XNOR, 짝수 번 반전되면 XOR로 정리된다.

NoteBase

문 20. 초기값이 $Q_AQ_BQ_CQ_D = 1000$인 <보기>의 카운터에 대한 설명으로 가장 옳은 것은?

<보기>



① 클럭펄스의 입력은 레벨 트리거 방식이다.

② n개의 플립플롭으로 구성되면 2n개의 상태를 가진다.

③ 8개의 상태가 나오며, 주파수 분주회로에 주로 사용된다.

④ 임의의 시간에 한 플립플롭만 1이 되고 나머지는 0이 되는 카운터이다.

정답 4번

출제 포인트

이 문제는 마지막 출력 Q_D를 첫 번째 D 입력으로 그대로 되먹임한 4비트 링 카운터의 상태 수와 동작 특성을 회로도에서 판별하는 문제이다.

선지별 해설

1번 X: 플립플롭의 클록 입력에 삼각형 표시가 있으므로 레벨 트리거가 아니라 에지 트리거형이다. 틀린 표현은 '레벨 트리거 방식'이며, 올바르게는 '클록 에지에서 상태가 이동하는 방식'이다.
2번 X: 이 회로는 4개의 플립플롭으로 가능한 모든 상태를 쓰는 일반 카운터가 아니라, 초기값 1000이 0100→0010→0001로 순환하는 링 카운터이다. 틀린 표현은 '2n개의 상태를 가진다'이며, 올바르게는 'n개의 유효 상태를 가진다' 또는 이 회로에서는 '4개의 유효 상태를 가진다'이다.
3번 X: 4비트 링 카운터는 8개의 상태가 아니라 4개의 상태를 순환한다. 틀린 표현은 '8개의 상태가 나오며'이고, 올바르게는 '4개의 상태가 나온다'이다. 8개의 상태를 사용하는 것은 4비트 존슨 카운터에 해당한다.
4번 O: 초기값이 1000이므로 클록이 들어올 때마다 0100, 0010, 0001, 1000으로 한 자리씩 이동한다. 따라서 임의의 시각에도 한 플립플롭만 1이고 나머지는 0인 one-hot 상태가 유지되므로 옳다.

NoteBase

오답 포인트

링 카운터와 존슨 카운터를 혼동하면 상태 수를 8개로 잘못 판단하기 쉽다. 또 일반 n비트 카운터의 2^n 상태 개념을 링 카운터에 그대로 적용하면 유효 상태 수를 틀리게 된다.

다시 나오면 이렇게 판단

되먹임 선이 마지막 Q 출력을 첫 D 입력으로 그대로 연결하면 링 카운터로 본다. 그다음 초기값이 one-hot인지 확인하고, 클록마다 1이 한 칸씩 이동하면 상태 수는 n개라고 판단한다.

암기 포인트

링 카운터는 Q를 그대로 되먹이며 one-hot 상태가 순환하고 유효 상태 수는 n개이다.

NoteBase