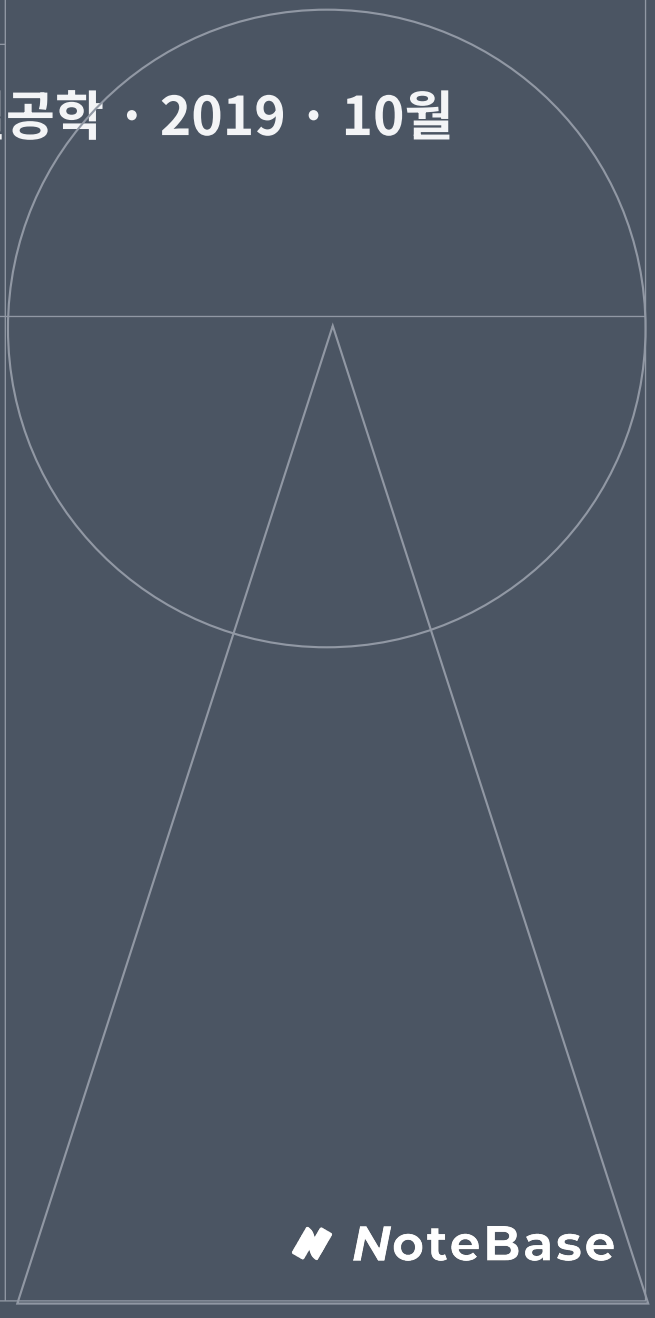
기출OX 해설집

공무원/서울시/7급 · 디지털공학 · 2019 · 10월



NoteBase

기출OX 해설집

공무원/서울시/7급 · 디지털공학 · 2019 · 10월

공무원/서울시/7급 · 디지털공학 · 2019 · 10월

문 1. 16진수 $(43)_{16}$과 $(52)_{16}$을 더한 값을 BCD(Binary Coded Decimal) 코드로 나타낸 것은?

① $(0001\ 0100\ 1001)_{BCD}$

② $(0001\ 1001\ 0101)_{BCD}$

③ $(0000\ 1001\ 0101)_{BCD}$

④ $(0000\ 0101\ 1001)_{BCD}$

정답 1번

출제 포인트

이 문제는 16진수를 10진수로 바꾼 뒤 합을 구하고, 그 10진수를 BCD로 다시 표현하는 문제이다.

조건 정리

주어진 값: $(43)_{16}$, $(52)_{16}$
구해야 할 값: 두 수의 합을 BCD 코드로 나타낸 결과
주의할 조건: BCD는 10진수의 각 자릿수를 4비트로 따로 표현하는 코드이므로, 16진수 비트를 그대로 쓰는 방식이 아니다.

사용할 공식

16진수 $(ab)_{16}$는 $a \times 16 + b$로 10진수로 바꾼다. 그다음 나온 10진수의 각 자리 숫자를 4비트 BCD로 바꾼다.

이 공식을 쓰는 이유:
문제는 16진수 덧셈 결과를 바로 묻는 것이 아니라, 덧셈 결과의 10진수 값을 BCD로 표현한 형태를 묻기 때문이다.

NoteBase

계산 과정

1. $(43)_{16} = 4 \times 16 + 3 = 67$이다.
2. $(52)_{16} = 5 \times 16 + 2 = 82$이다.
3. 두 수의 합은 $67 + 82 = 149$이다.
4. 10진수 149의 각 자리 1, 4, 9를 BCD로 바꾸면 1 → 0001, 4 → 0100, 9 → 1001이다.
5. 따라서 BCD 표현은 0001 0100 1001이다.

선지별 해설

① O: 계산 결과가 149이고, 149의 BCD는 0001 0100 1001이므로 옳다.
② X: 0001 1001 0101은 10진수 각 자리로 읽으면 195의 BCD이다. 틀린 표현은 149를 195처럼 바꾼 부분이며, 올바르게는 0001 0100 1001이어야 한다.
③ X: 0000 1001 0101은 095, 즉 95의 BCD이다. 틀린 표현은 백의 자리 1이 빠진 부분이며, 올바르게는 첫 4비트가 0001이어야 한다.
④ X: 0000 0101 1001은 059, 즉 59의 BCD이다. 틀린 표현은 149를 59처럼 적은 부분이며, 올바르게는 0001 0100 1001이어야 한다.

오답 포인트

출제자가 노린 함정은 16진수의 비트값과 BCD를 혼동하는 점이다. 16진수 합 $(43)_{16} + (52)_{16} = (95)_{16}$만 보고 95라고 착각하면 ③으로 흐르기 쉽다. 또한 149의 각 자리 1, 4, 9를 따로 코딩하지 않고 자릿수를 빠뜨리면 ③, ④ 같은 오답이 나온다.

다시 나오면 이렇게 판단

진법 변환형 문제는 원래 진수에서 먼저 10진수 값을 정확히 구한 뒤, 요구한 코드 체계가 BCD인지 순수 이진수인지 확인해서 각 자리별로 다시 표현하는 순서로 판단한다.

암기 포인트

BCD는 값 전체를 한 번에 바꾸는 코드가 아니라 10진수 각 자리를 4비트씩 따로 바꾸는 코드이다.

NoteBase

문 2. 카르노맵(Karnaugh map)이 <보기>와 같이 주어졌을 때, 출력 Y에 대한 논리식으로 가장 옳은 것은?

<보기>

Y \ AB / CD	00	01	11	10
00	1	1	1	1
01	0	1	0	0
11	0	1	1	1
10	1	1	1	1

① $Y = A + C$

② $Y = AC + \overline{B}CD$

③ $Y = ACD + \overline{A}B + D$

④ $Y = ACD + \overline{A}B + \overline{D}$

정답 4번

출제 포인트

이 문제는 카르노맵에서 1을 가장 크게 묶어 $\bar{D}$, $\bar{A}B$, ACD를 정확히 뽑아내는지를 묻는 문제이다.

NoteBase

선지별 해설

① X
$Y = A + C$이면 $AB = 01,\ CD = 01$ 칸에서 $A = 0,\ C = 0$이므로 0이 되지만, 카르노맵 값은 1이다.
틀린 식은 $A + C$이다. 올바르게는 $Y = ACD + \bar{A}B + \bar{D}$이다.

② X
$CD = 00$ 행은 네 칸이 모두 1이므로 최소항에 반드시 $\bar{D}$가 포함되어야 한다. 그런데 $AC + \bar{B}CD$는 $CD = 00$에서 항상 0이 되어 맵과 맞지 않는다.
틀린 식은 $AC + \bar{B}CD$이다. 올바르게는 $Y = ACD + \bar{A}B + \bar{D}$이다.

③ X
마지막 항이 D이면 $CD = 01$ 행의 값들이 전부 1처럼 처리된다. 예를 들어 $AB = 00,\ CD = 01$ 칸은 실제로 0인데, $ACD + \bar{A}B + D$는 $D = 1$ 때문에 1이 된다.
틀린 표현은 마지막 항 D이다. 올바르게는 D가 아니라 $\bar{D}$이므로 $Y = ACD + \bar{A}B + \bar{D}$이다.

④ O
$CD = 00, 10$ 두 행의 8개 1을 묶으면 $\bar{D}$, $AB = 01$ 한 열의 4개 1을 묶으면 $\bar{A}B$, 남는 $CD = 11$에서 $AB = 11, 10$ 두 칸을 묶으면 ACD가 된다.
따라서 $Y = ACD + \bar{A}B + \bar{D}$가 카르노맵을 가장 간단하게 만족한다.

오답 포인트

$\bar{D}$와 D를 뒤집어 읽거나, 큰 묶음을 놓치고 셀 몇 개만 따라가면 ③처럼 보이기 쉽다. 이 맵은 먼저 행 전체와 열 전체를 찾는 것이 핵심이다.

다시 나오면 이렇게 판단

카르노맵은 8칸 묶음, 4칸 묶음, 2칸 묶음 순서로 최대 묶음을 먼저 찾는다. 이 문제에서는 $D = 0$인 두 행 전체에서 $\bar{D}$가 가장 먼저 나온다.

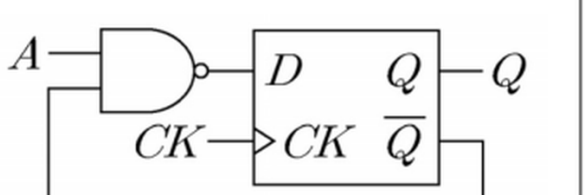
암기 포인트

카르노맵 최소화는 큰 묶음부터 본다. 행 전체는 한 변수만 남고, $D = 0$ 두 행 전체는 $\bar{D}$이다.

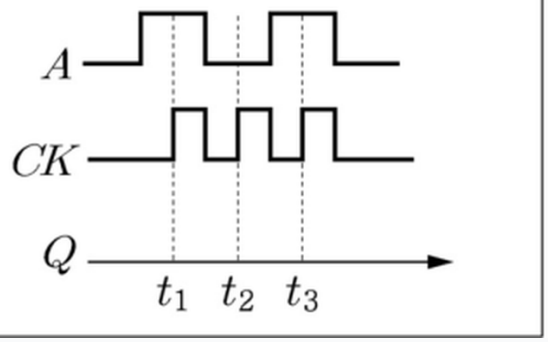
NoteBase

문 3. <보기 1>의 회로도에서 초기 Q값이 0이고, <보기 2>와 같은 신호가 입력 A와 CK에 가해졌을 때 출력 Q의 파형으로 가장 옳은 것은?

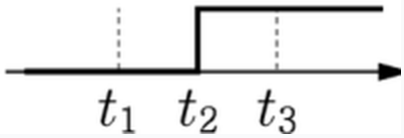
<보기 1>



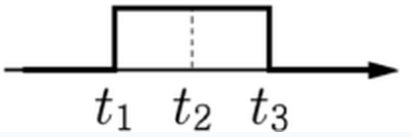
<보기 2>



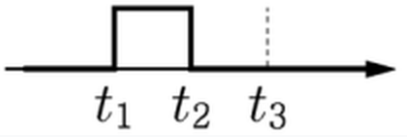
①



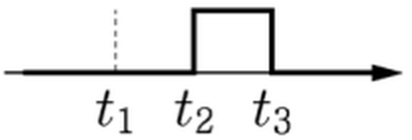
②



③



④



정답 1번

출제 포인트

이 문제는 NAND 게이트 출력이 D 입력으로 들어가는 양의 에지 트리거 D 플립플롭에서, 각 클록 시점의 A와 피드백 $\overline{Q}$를 이용해 $D = \overline{A \cdot \overline{Q}} = \overline{A} + Q$를 판단하는 문제이다.

NoteBase

선지별 해설

① O

- 초기 $Q=0$이므로 $\overline{Q}=1$이다. t_1에서 $A=1$이므로 $D=\overline{1\cdot 1}=0$이라서 Q는 0을 유지한다.
- t_2에서 $A=0$이므로 $D=\overline{0\cdot 1}=1$이 되어 Q가 1로 바뀐다.
- t_3에서는 이미 $Q=1$이므로 $\overline{Q}=0$이고, 따라서 $D=\overline{A\cdot 0}=1$이어서 Q가 계속 1이다. 따라서 t_2 이후 상승한 뒤 유지되는 ①이 맞다.

② X

- 이 파형은 t_1에서 바로 1로 올라가지만, 실제로는 t_1에서 $D=0$이므로 Q는 0이다.
- 틀린 표현은 "t_1에서 상승한다"는 부분이다. 올바르게는 Q는 t_2에서 처음 1로 바뀐다.

③ X

- 이 파형은 t_1에서 1이 되고 t_2에서 다시 0이 되는데, 실제 회로는 반대로 t_1에서는 0을 유지하고 t_2에서 1이 된다.
- 틀린 표현은 "$t_1 \sim t_2$ 구간만 1이다"는 부분이다. 올바르게는 t_2 이후 1이 유지된다.

④ X

- t_2에서 1이 되는 점은 맞지만 t_3에서 다시 0으로 내려가지 않는다. t_3에서는 피드백 때문에 $D=1$이 유지되기 때문이다.
- 틀린 표현은 "t_3에서 다시 0이 된다"는 부분이다. 올바르게는 t_2에서 1이 된 뒤 t_3 이후에도 1을 유지한다.

오답 포인트

입력 A 파형만 보고 Q가 그대로 따라간다고 오해하기 쉽다. 이 회로는 단순 전달이 아니라 $\overline{Q}$가 되먹임되는 NAND 출력이 D가 되므로, 각 시점에서 먼저 $\overline{Q}$ 값을 반영해 D를 계산해야 한다는 점이 함정이다.

다시 나오면 이렇게 판단

에지 트리거 D 플립플롭 파형 문제는 각 클록 유효 시점마다 현재 Q에서 $\overline{Q}$를 구하고, 회로식으로 D를 계산한 뒤 그 값을 다음 Q로 옮기는 순서로 판단하면 된다.

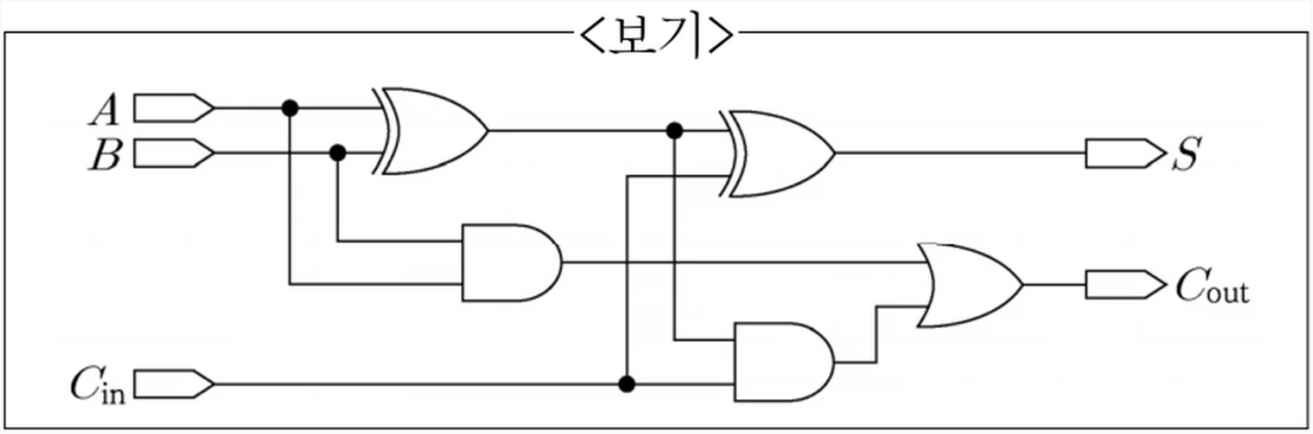
암기 포인트

D 플립플롭 되먹임 회로는 입력 파형을 바로 보지 않고, 클록 시점마다 D 식을 계산한 값이 다음 Q가 된다고 정리한다.

NoteBase

문 4. <보기>의 회로도에서 출력 S, C_{out}에 대한 설명으로 가장 옳은 것은?

<보기>



① 입력 A, B, C_{in}의 합이 홀수이면 S는 항상 1을 출력한다.

② 입력 A, B, C_{in}의 합이 홀수이면 S는 항상 0을 출력한다.

③ 입력 A, B, C_{in}의 합이 홀수이면 C_{out}는 항상 1을 출력한다.

④ 입력 A, B, C_{in}의 합이 짝수이면 C_{out}는 항상 0을 출력한다.

정답 1번

출제 포인트

이 문제는 회로도에서 전가산기의 구조를 읽어 S가 3개 입력의 패리티 출력이고 C_{out}이 캐리 출력이라는 점을 판단하는 문제이다.

NoteBase

선지별 해설

① O

- 위쪽은 XOR 두 개가 직렬로 연결되어 있어 $S = A \oplus B \oplus C_{in}$이다. XOR는 입력 1의 개수가 홀수일 때 1이 되므로 맞는 설명이다.

② X

- 입력 합이 홀수일 때 S는 0이 아니라 1이 된다. 틀린 표현은 "항상 0"이고, 올바르게는 "입력 A, B, C_{in}의 합이 홀수이면 S는 항상 1을 출력한다"이다.

③ X

- C_{out}은 아래 AND-OR 조합으로 만들어지는 캐리이므로 세 입력 중 1이 두 개 이상일 때만 1이 된다. 홀수 합이라고 해서 항상 1이 되는 것은 아니며, 합이 1일 때는 $C_{out} = 0$이다. 올바르게는 "입력 A, B, C_{in} 중 1이 두 개 이상이면 C_{out}은 1을 출력한다"이다.

④ X

- 입력 합이 짝수여도 합이 2인 경우에는 캐리가 발생하여 $C_{out} = 1$이 된다. 틀린 표현은 "짝수이면 항상 0"이고, 올바르게는 "입력 A, B, C_{in}의 합이 0이면 $C_{out} = 0$, 2이면 $C_{out} = 1$이다"이다.

오답 포인트

S와 C_{out}을 둘 다 단순한 홀짝 판단으로 처리하면 ③, ④에 걸리기 쉽다. S는 패리티 출력이고 C_{out}은 캐리 출력이므로 기준이 서로 다르다.

다시 나오면 이렇게 판단

회로에서 XOR가 직렬이면 먼저 패리티 출력 S를 떠올리고, AND들의 OR가 보이면 세 입력 중 두 개 이상이 1일 때 캐리 C_{out}이 1이 된다고 판단한다.

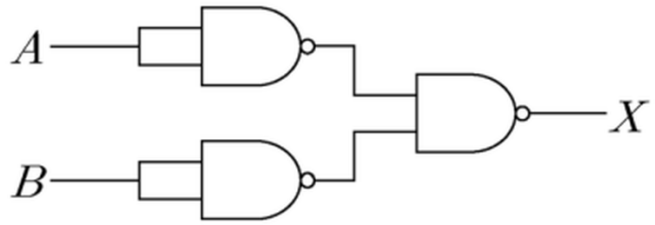
암기 포인트

전가산기에서 S는 홀수 개 1의 XOR이고 C_{out}은 두 개 이상 1의 carry이다.

NoteBase

문 5. NAND 게이트를 이용한 <보기>의 논리회로와 동일한 연산은?

<보기>



① NOR 연산

② OR 연산

③ AND 연산

④ NOT 연산

정답 2번

출제 포인트

이 문제는 NAND 게이트에 입력을 묶어 인버터로 쓰는 구조와 드모르간 법칙을 이용해 회로의 최종 연산이 무엇인지 판단하는 문제이다.

선지별 해설

① X. NOR는 $\overline{A+B}$이다. 이 회로는 각 입력을 먼저 반전시켜 $\overline{A}, \overline{B}$를 만든 뒤 마지막 NAND에서 $X = \overline{\overline{A} \cdot \overline{B}} = A + B$가 된다. 틀린 표현은 "NOR 연산"이고, 올바르게는 "OR 연산"이다.
② O. 앞단의 두 NAND는 입력을 하나로 묶어 사용했으므로 각각 NOT 게이트처럼 동작하여 $\overline{A}, \overline{B}$를 출력한다. 이를 마지막 NAND에 넣으면 드모르간 법칙으로 $A + B$가 되어 OR 연산과 같다.
③ X. AND는 $A \cdot B$이다. 이 회로의 최종 출력은 두 입력의 곱이 아니라 합이다. 틀린 표현은 "AND 연산"이고, 올바르게는 "OR 연산"이다.
④ X. NOT은 입력 하나를 반전하는 연산이다. 이 회로는 A와 B 두 입력을 함께 받아 최종적으로 $A + B$를 만든다. 틀린 표현은 "NOT 연산"이고, 올바르게는 "OR 연산"이다.

오답 포인트

앞단에 NAND가 두 개 보인다고 해서 전체를 NAND나 AND 계열로 바로 판단하면 오답이 된다. 입력이 묶인 NAND는 인버터로 동작한다는 점을 놓치거나, 마지막 단계에서 드모르간 법칙을 적용하지 않으면 ①이나 ③으로 흔들리기 쉽다.

NoteBase

다시 나오면 이렇게 판단

입력이 묶인 NAND가 보이면 먼저 NOT으로 바꿔 본다. 그다음 마지막 게이트까지 식으로 써서 $\overline{\overline{A} \cdot \overline{B}}$처럼 정리하면 OR인지 NOR인지 바로 구분된다.

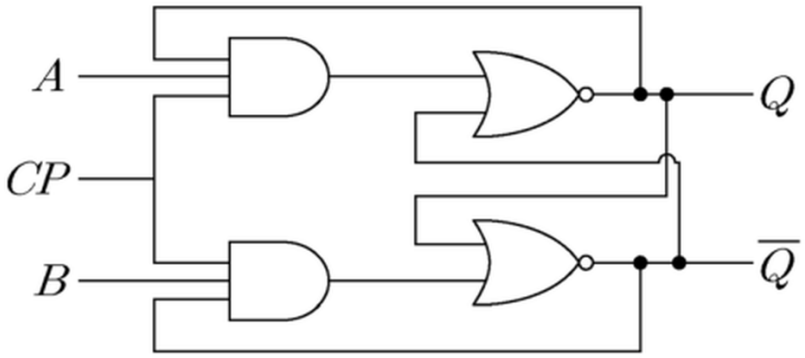
암기 포인트

입력 묶은 NAND는 NOT, $\overline{\overline{A} \cdot \overline{B}} = A + B$는 OR이다.

NoteBase

문 6. <보기>의 회로도가 나타내는 플립플롭에 대한 설명으로 가장 옳은 것은?

<보기>



① A, B가 1, 1인 경우 $Q(t+1)$의 상태는 부정이다.

② NAND 게이트를 사용하여도 구현이 가능하다.

③ A, B 값이 0, 0인 경우 $Q(t+1)$의 상태는 $\overline{Q}(t)$이다.

④ S-R 플립플롭으로 가장 많이 사용된다.

정답 2번

출제 포인트

이 문제는 출력 피드백이 걸린 AND 게이트와 NOR 래치를 보고 이 회로가 JK 플립플롭임을 식별하고, JK의 입력 조합별 다음 상태를 판단할 수 있는지를 묻는다.

핵심 개념

JK 플립플롭의 동작은 00이면 현상 유지, 01이면 리셋, 10이면 세트, 11이면 토글이다. JK는 SR 플립플롭의 11 금지 상태를 없애기 위해 출력 피드백을 건 구조이다.

선지별 해설

1번: X. JK 플립플롭에서 A, B가 1, 1이면 불정 상태가 아니라 토글 상태가 된다. 따라서 $Q(t+1)$은 현재 상태의 반전인 $\overline{Q(t)}$가 된다. 틀린 표현은 '불정이다'이고, 올바르게는 '현재 출력이 반전된다'이다.
2번: O. JK 플립플롭은 NOR 게이트로도, NAND 게이트로도 구현할 수 있다. NAND는 범용 게이트이므로 피드백 구조를 바꾸어 같은 기능의 JK 플립플롭을 만들 수 있다.
3번: X. JK 플립플롭에서 A, B가 0, 0이면 입력이 모두 비활성이라 상태를 그대로 유지한다. 따라서 $Q(t+1) = Q(t)$이다. 틀린 표현은 '$\overline{Q(t)}$이다'이고, 올바르게는 '현 상태를 유지한다'이다.
4번: X. 그림의 회로는 SR 플립플롭이 아니라 JK 플립플롭이다. JK는 SR의 금지 입력 문제를 보완한 형태이다. 틀린 표현은 'S-R 플립플롭'이고, 올바르게는 'JK 플립플롭'이다.

NoteBase

오답 포인트

SR 플립플롭의 11 불정 상태와 JK 플립플롭의 11 토글 상태를 섞으면 1번에서 흔들린다. 또 00 입력을 리셋으로 착각하면 3번을 틀리기 쉽다.

다시 나오면 이렇게 판단

출력 $Q, \overline{Q}$가 입력 게이트 쪽으로 다시 피드백되면 먼저 JK 여부를 본다. 그다음 00 유지, 01 리셋, 10 세트, 11 토글의 상태표를 대입하면 선지를 바로 가를 수 있다.

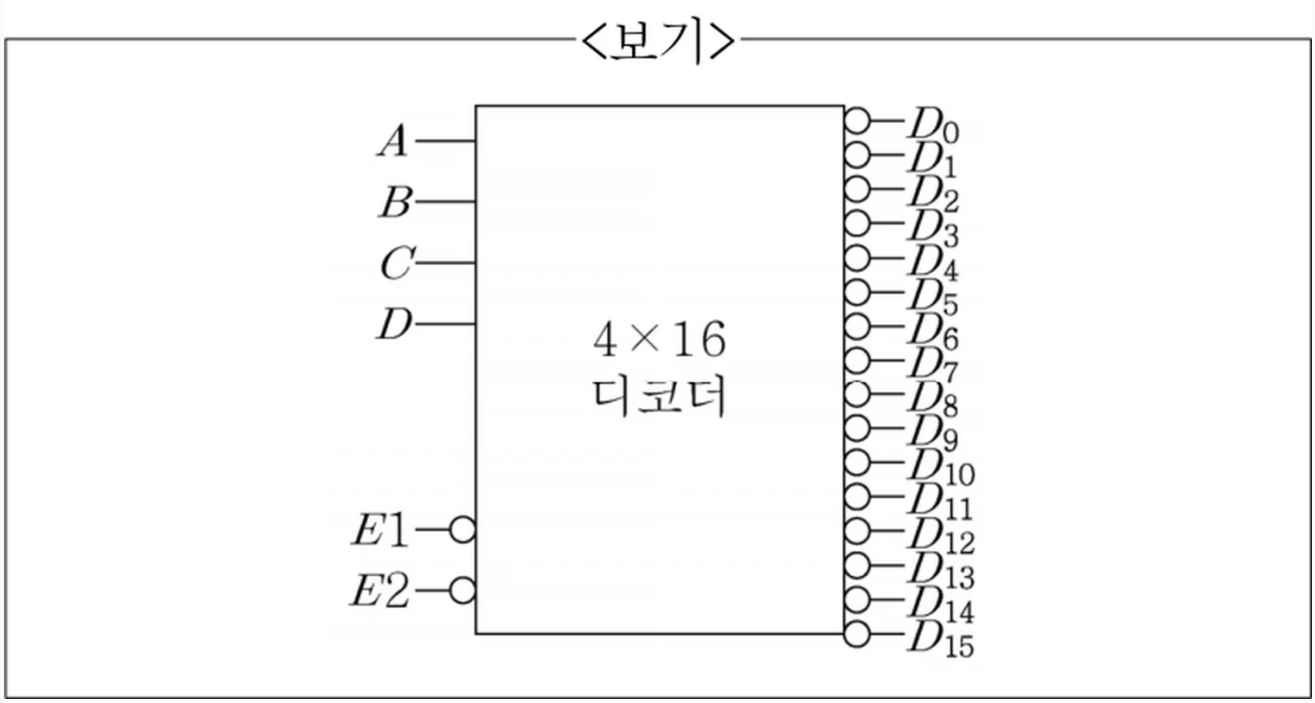
암기 포인트

JK는 SR의 금지 상태를 없앤 플립플롭이며, 11이면 토글이고 00이면 유지이다.

NoteBase

문 7. <보기>의 4×16 디코더에 E1 = E2 = 1이고, A = 0, B = 1, C = 1, D = 0이 인가되었을 경우 출력논리에 대한 설명으로 가장 옳은 것은? (단, E1, E2는 인에이블 단자이며, A가 MSB, D가 LSB를 나타낸다.)

<보기>



① 0에서 15번까지의 출력 중 6번째 출력만이 LOW 상태가 되고, 나머지 출력은 모두 HIGH가 된다.

② 0에서 15번까지의 출력 중 6번째 출력만이 HIGH 상태가 되고, 나머지 출력은 모두 LOW가 된다.

③ 0에서 15번까지의 모든 출력이 HIGH가 된다.

④ 0에서 15번까지의 모든 출력이 LOW가 된다.

정답 3번

출제 포인트

이 문제는 4×16 디코더의 도식에서 인에이블 단자의 버블과 출력단의 버블을 함께 읽어 활성 조건과 출력 극성을 판단하는 문제이다.

NoteBase

선지별 해설

① X
- 입력 $ABCD$는 0110이므로 선택 번호는 6번이 맞다. 다만 E1, E2 단자에 버블이 있으므로 두 인에이블은 활성 Low이다.
- E1=E2=1이면 디코더는 비활성 상태이므로 6번 출력만 LOW가 되지 않는다. 틀린 표현은 "6번째 출력만이 LOW"이고, 올바르게는 "비활성 상태이므로 모든 출력이 HIGH"이다.

② X
- "6번째 출력만이 HIGH, 나머지는 LOW"는 활성 High 출력 디코더를 가정한 설명이다.
- 이 도식은 출력에도 버블이 있어 활성 Low 출력이며, 게다가 E1=E2=1이라 비활성 상태이다. 틀린 표현은 "6번째 출력만이 HIGH"이고, 올바르게는 "비활성 상태이므로 모든 출력이 HIGH"이다.

③ O
- 입력값 $A=0, B=1, C=1, D=0$은 10진수로 6이지만, 인에이블 단자의 버블 때문에 E1=E2=1에서는 디코더가 동작하지 않는다.
- 비활성된 활성 Low 출력 디코더의 출력은 모두 HIGH가 되므로 ③이 옳다.

④ X
- 출력 버블이 있다고 해서 항상 LOW가 되는 것은 아니다. 버블은 선택되었을 때의 활성 극성이 Low임을 뜻한다.
- 이 문제에서는 인에이블이 비활성이므로 모든 출력이 LOW가 아니다. 틀린 표현은 "모든 출력이 LOW"이고, 올바르게는 "모든 출력이 HIGH"이다.

오답 포인트

입력 0110만 보고 곧바로 6번 출력이 선택된다고 판단하면 ①로 끌려가기 쉽다. 이 문항은 먼저 인에이블 버블을 읽어 동작 여부를 확인한 뒤, 그다음 출력 버블로 활성 극성을 판단해야 한다.

다시 나오면 이렇게 판단

디코더 문제는 먼저 인에이블 단자에 버블이 있는지 보고 활성 High인지 활성 Low인지 판단한다. 그다음 출력단 버블로 선택 출력의 활성 극성을 확인하고, 마지막에 입력 이진수를 번호로 바꿔 적용하면 된다.

암기 포인트

버블은 반전 표식이므로 입력 버블은 활성 Low, 출력 버블은 선택 시 LOW 출력을 뜻한다.

NoteBase

문 8. 플립플롭에 관한 설명으로 가장 옳은 것은?

① 플립플롭은 클럭신호가 0에서 1로 바뀌는 시점에서만 값을 변화시킬 수 있다.

② 셋업시간은 클럭신호의 에지가 발생한 다음, 동기제어신호가 검출 가능한 적정 레벨을 유지해야하는 시간을 말한다.

③ 홀드시간은 제어입력이 검출 가능한 레벨로 유지되는 동안 클럭신호의 에지가 검출되기까지의 시간을 말한다.

④ J-K 플립플롭을 토글모드로 동작시키기 위해서는 J=K=1이어야 한다.

정답 4번

출제 포인트

이 문제는 에지 트리거 플립플롭의 동작 시점, 셋업시간과 홀드시간의 기준 시점, J-K 플립플롭의 토글 조건을 구분하는 문제이다.

핵심 개념

플립플롭은 지정된 클럭 에지에서만 상태를 갱신하는 순서회로이다.
셋업시간은 클럭의 유효 에지 이전에 입력이 안정되어 있어야 하는 최소 시간이다.
홀드시간은 클럭의 유효 에지 이후에도 입력이 그대로 유지되어야 하는 최소 시간이다.
J-K 플립플롭은 $J = K = 1$일 때 출력이 반전되어 토글 모드로 동작한다.

선지별 해설

① X
- 틀린 부분은 "0에서 1로 바뀌는 시점에서만"이다.
- 올바르게는 "플립플롭은 지정된 유효 에지에서 값을 변화시킬 수 있다"이다. 상승에지형은 0→1, 하강에지형은 1→0에서 동작한다.

② X
- 틀린 부분은 "에지가 발생한 다음"이다.
- 올바르게는 "셋업시간은 클럭 에지가 발생하기 이전에 입력이 안정되어 있어야 하는 시간"이다.

③ X
- 틀린 부분은 "클럭신호의 에지가 검출되기까지"이다.
- 올바르게는 "홀드시간은 클럭 에지가 검출된 뒤에도 입력이 유지되어야 하는 시간"이다.

④ O
- J-K 플립플롭은 $J = K = 1$이면 다음 상태가 현재 상태의 보수로 바뀌므로 토글 모드라는 설명이 옳다.

NoteBase

오답 포인트

②와 ③은 셋업시간과 홀드시간의 기준 방향을 서로 바꿔 놓은 함정이다. 셋업은 에지 전, 홀드는 에지 후라는 시간축을 놓치면 두 선지를 쉽게 바꿔 고르게 된다.

다시 나오면 이렇게 판단

플립플롭 문항은 먼저 상태 변화가 레벨 구동인지 에지 구동인지부터 구분하면 된다. 시간 파라미터는 클럭 에지 이전이면 셋업, 이후면 홀드로 판단하면 되고, J-K는 $J = K = 1$이면 토글, $J = K = 0$이면 유지로 정리된다.

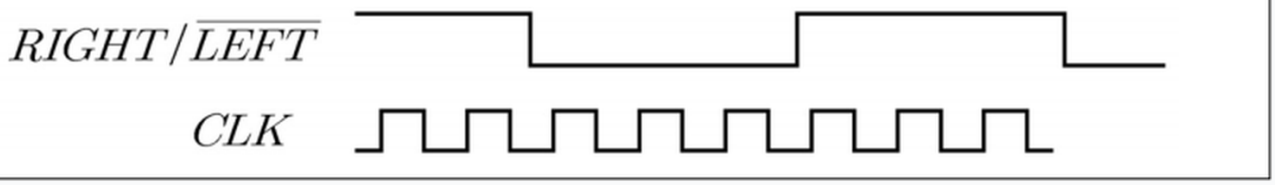
암기 포인트

셋업은 에지 전, 홀드는 에지 후, J-K의 11은 토글이다.

NoteBase

문 9. RIGHT/$\overline{\text{LEFT}}$ **제어입력이 <보기>와 같이 주어졌을 때, 4비트 양방향 시프트 레지스터의 7클럭 후** Q_0, Q_1, Q_2, Q_3**값은? (단,** $Q_0 = 1$, $Q_1 = 1$, $Q_2 = 1$, $Q_3 = 0$**으로 초기화 되어 있으며, 직렬 입력 선은 LOW이다.** Q_0**는 가장 왼쪽에 위치한 플립플롭의 출력 값을 나타낸다.)**

<보기>



① (1, 1, 1, 0)

② (1, 0, 0, 0)

③ (0, 0, 1, 0)

④ (0, 0, 0, 0)

정답 3번

출제 포인트

이 문제는 RIGHT/LEFT 제어 파형을 보고 7개의 클록을 RIGHT 2클록 → LEFT 3클록 → RIGHT 2클록으로 나눈 뒤, 4비트 양방향 시프트 레지스터의 상태 변화를 순서대로 추적하는 문제이다.

선지별 해설

① X: Q_0, Q_1, Q_2, Q_3=(1,1,1,0)은 초기값일 뿐 7클록 후 값이 아니다. HIGH 구간에서 RIGHT 시프트 2번을 적용하면 1110→0011이 되고, 이어 LOW 구간에서 LEFT 시프트 3번을 적용하면 0011→1000이 되며, 마지막 HIGH 구간에서 RIGHT 시프트 2번을 더 적용하면 1000→0010이 된다. 올바르게는 0,0,1,0이다.
② X: 1,0,0,0은 LEFT 시프트 3번까지 반영한 중간 상태에 해당한다. 문제는 여기서 끝나지 않고 마지막 RIGHT 시프트 2번을 더 적용해야 하므로 1000→0100→0010이 된다. 올바르게는 0,0,1,0이다.
③ O: RIGHT 시프트 2번, LEFT 시프트 3번, RIGHT 시프트 2번을 순서대로 적용하면 1110→0011→1000→0010이 된다. 따라서 최종 출력은 Q_0, Q_1, Q_2, Q_3=(0,0,1,0)이다.
④ X: 직렬 입력선이 LOW라고 해서 항상 7클록 후 0000이 되는 것은 아니다. 중간의 LEFT 시프트 3번에서 기존의 1이 다시 왼쪽으로 이동하므로 마지막에 Q_2에 1이 남는다. 올바르게는 0,0,1,0이다.

오답 포인트

가장 흔한 실수는 RIGHT/LEFT 제어 파형의 구간 길이를 세지 않고 한 방향 시프트만 계속 적용하는 것이다. 특히 ②처럼 LEFT 시프트 3번까지의 중간값을 최종값으로 착각하거나, ④처럼 직렬 입력이 LOW라는 이유만으로 무조건 전부 0이 된다고 판단하기 쉽다.

NoteBase

다시 나오면 이렇게 판단

양방향 시프트 레지스터 문제는 먼저 제어 파형을 클록 수로 끊어 각 구간의 방향을 확정하고, 그다음 초기값을 한 클록씩 표처럼 갱신하면 된다. 방향이 바뀌는 지점마다 중간 상태를 따로 적어 두면 최종값을 안정적으로 구할 수 있다.

암기 포인트

시프트 레지스터는 직렬입력값만 보지 않고 방향 전환 시점까지 포함해 상태를 한 클록씩 추적해야 한다.

NoteBase

문 10. 비동기식 MOD-100 카운터를 설계하려고 할 때, 필요한 플립플롭의 최소 개수는?

① 4개

② 5개

③ 6개

④ 7개

정답 4번

출제 포인트

이 문제는 비동기식 MOD-100 카운터를 구현할 때 필요한 최소 플립플롭 수를 $2^n \geq 100$ 조건으로 구하는 문제이다.

조건 정리

주어진 값: MOD-100 카운터이므로 필요한 상태 수는 100개이다.
구해야 할 값: 100개의 상태를 표현할 수 있는 최소 플립플롭 수이다.
주의할 조건: 플립플롭 n개로 표현 가능한 상태 수는 2^n개이다.

사용할 공식

사용 공식은 $2^n \geq N$이다.
이 공식을 쓰는 이유는 N진 카운터를 만들려면 최소한 N개의 서로 다른 상태를 표현할 수 있어야 하기 때문이다.

계산 과정

1. 필요한 상태 수는 100개이다.
2. $2^6 = 64$이므로 6개 플립플롭으로는 100개 상태를 모두 표현할 수 없다.
3. $2^7 = 128$이므로 7개 플립플롭이면 100개 상태를 표현할 수 있다.
4. 따라서 필요한 최소 플립플롭 수는 7개이다.

선지별 해설

① X: 4개는 $2^4 = 16$개 상태만 표현할 수 있으므로 MOD-100을 구현할 수 없다. 틀린 표현은 "4개"이며, 올바르게는 "7개"이다.
② X: 5개는 $2^5 = 32$개 상태만 표현할 수 있으므로 부족하다. 틀린 표현은 "5개"이며, 올바르게는 "7개"이다.
③ X: 6개는 $2^6 = 64$개 상태만 표현할 수 있어서 100에 미달한다. 틀린 표현은 "6개"이며, 올바르게는 "7개"이다.
④ O: 7개는 $2^7 = 128$개 상태를 표현할 수 있으므로 100개 상태가 필요한 MOD-100 카운터를 구현할 수 있다.

NoteBase

오답 포인트

MOD-100이라는 숫자만 보고 100을 2, 10, 20처럼 단순 분해해서 플립플롭 수를 추정하면 오답이 된다. 이 문제의 함정은 상태 수를 직접 세지 않고 감으로 고르는 데 있다.

다시 나오면 이렇게 판단

카운터의 플립플롭 수 문제는 먼저 필요한 상태 수 N을 확인한 뒤, 2^n이 처음으로 N 이상이 되는 n을 찾는 방식으로 판단한다.

암기 포인트

카운터 최소 비트 수는 $2^n \geq N$을 처음 만족하는 n이다.

NoteBase

문 11. 2의 보수(two's complement) 수 체계에서 -5값을 4비트 그레이 코드(gray code)로 변환한 것은?

① $(1011)_{gray}$

② $(1101)_{gray}$

③ $(1110)_{gray}$

④ $(1111)_{gray}$

정답 3번

출제 포인트

이 문제는 4비트 2의 보수 표현을 먼저 구한 뒤 그 값을 그레이 코드로 변환할 수 있는지를 묻는다.

조건 정리

주어진 값: -5의 4비트 2의 보수 표현
구해야 할 값: 그 값을 4비트 그레이 코드로 바꾼 결과
주의할 조건: 음수는 절댓값 5가 아니라 2의 보수 표현 1011을 기준으로 변환한다.

사용할 공식

그레이 코드 변환은 최상위 비트는 그대로 두고, 나머지는 왼쪽 비트와 현재 비트를 XOR하여 구한다. 즉, $g_3 = b_3$, $g_2 = b_3 \oplus b_2$, $g_1 = b_2 \oplus b_1$, $g_0 = b_1 \oplus b_0$이다.
이 공식을 쓰는 이유: 문제는 2의 보수 체계의 비트열을 그레이 코드로 바꾸는 과정이므로, 십진수 의미가 아니라 4비트 이진 패턴 자체를 변환해야 하기 때문이다.

계산 과정

1. +5의 4비트 이진수는 0101이다.
2. 이를 반전하면 1010이고 1을 더하면 -5의 4비트 2의 보수는 1011이다.
3. 1011에 대해 그레이 변환을 적용하면 $g_3 = 1$, $g_2 = 1 \oplus 0 = 1$, $g_1 = 0 \oplus 1 = 1$, $g_0 = 1 \oplus 1 = 0$이므로 결과는 1110이다.

선지 판단

1번 X: 1011은 -5의 4비트 2의 보수 표현 자체이다. 그레이 코드로 변환한 결과가 아니며, 올바르게는 1110이다.
2번 X: 1101은 1011을 그레이 코드 규칙으로 계산한 값이 아니다. 특히 g_1과 g_0 계산이 맞지 않으며, 올바르게는 1110이다.
3번 O: 1011을 그레이 코드로 바꾸면 1110이 된다. 따라서 정답이다.
4번 X: 1111은 마지막 비트 g_0를 1로 잘못 둔 경우이다. 실제로는 $1 \oplus 1 = 0$이므로 올바르게는 1110이다.

NoteBase

오답 포인트

음수 -5를 절댓값 0101로 바로 그레이 변환하거나, 2의 보수 1011을 구해 놓고 그대로 정답으로 고르면 틀린다. 그레이 코드는 인접 비트 XOR 규칙으로 다시 한 번 변환해야 한다.

다시 나오면 이렇게 판단

음수 변환 문제는 먼저 비트 패턴을 확정하고, 그다음 최상위 비트 유지와 인접 비트 XOR 순서로 그레이 코드를 만든다. 값의 부호 해석과 코드 변환 단계를 섞지 않으면 된다.

암기 포인트

2의 보수값을 먼저 만든 뒤, 그레이 코드는 최상위 비트는 그대로 두고 나머지는 왼쪽 비트와 XOR한다.

NoteBase

문 12. <보기>의 카르노맵을 만족하는 최적화된 POS(Product of Sum)식은? (단, X는 무정의(don't care)를 뜻한다.)

<보기>

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

① $(A+\overline{B})(\overline{B}+C)$

② $(A+B)(\overline{B}+C)$

③ $\overline{B}+AC$

④ $\overline{B}+A\overline{C}$

정답 1번

출제 포인트

이 문제는 카르노맵에서 0과 don't care를 묶어 최소 POS식을 만드는 문제이다. POS는 1을 묶는 것이 아니라 0을 묶어 합항을 만든다.

NoteBase

선지별 해설

① O
- $AB = 01$인 행 전체를 4개로 묶으면 $(A + \overline{B})$가 된다.
- $AB = 01, 11$과 $CD = 00, 01$을 4개로 묶으면 $(\overline{B} + C)$가 된다.
- 따라서 최소 POS식은 $(A + \overline{B})(\overline{B} + C)$이다.

② X
- 첫 번째 합항이 $(A + B)$이면 $AB = 01$ 행 전체를 0으로 만들 수 없다.
- 여기서는 $B = 1$인 행 전체를 0으로 만들기 위해 B가 아니라 $\overline{B}$가 들어가야 한다.
- 틀린 표현은 $(A + B)$이고, 올바르게는 $(A + \overline{B})$이다.

③ X
- $\overline{B} + AC$는 같은 함수를 나타낼 수 있지만 형태가 SOP식이다.
- 문제는 최적화된 POS식을 묻고 있으므로 합의 곱 형태로 써야 한다.
- 올바른 POS 표현은 $(A + \overline{B})(\overline{B} + C)$이다.

④ X
- $\overline{B} + A\overline{C}$는 SOP식일 뿐 아니라 곱항의 C 부호를 잘못 잡은 식이다.
- 이 카르노맵에서 $B = 0$인 1들을 묶으면 남는 곱항은 AC이므로 $A\overline{C}$가 될 수 없다.
- 틀린 표현은 $A\overline{C}$이고, 올바르게는 AC이다.

오답 포인트

POS와 SOP를 혼동하면 ③처럼 같은 함수를 다른 표준형으로 쓴 선택지에 흔들리기 쉽다. 또 0을 묶을 때 값이 1로 고정된 변수는 보수형으로, 0으로 고정된 변수는 원래 형태로 들어간다는 규칙을 놓치면 ②, ④처럼 부호가 뒤집힌다.

다시 나오면 이렇게 판단

카르노맵에서 POS를 묻으면 먼저 0과 don't care를 최대 크기로 묶는다. 각 묶음에서 값이 0으로 고정된 변수는 그대로, 1로 고정된 변수는 보수로 써서 합항을 만들고, 마지막에 그 합항들을 곱하면 된다.

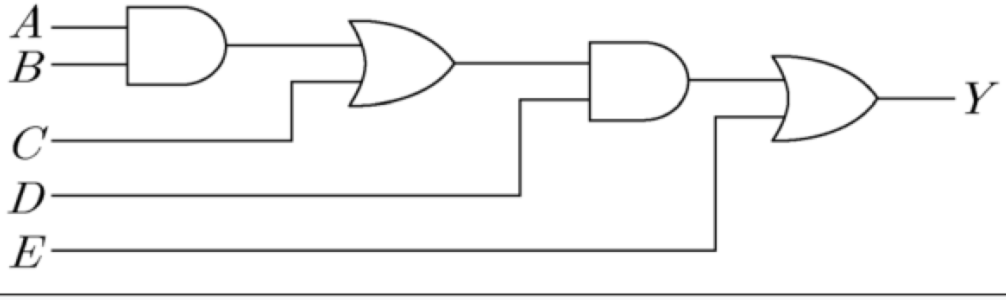
암기 포인트

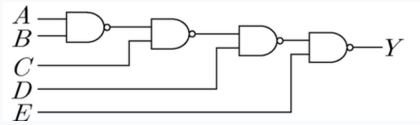
POS는 0을 묶어 합항을 만들고, SOP는 1을 묶어 곱항을 만든다.

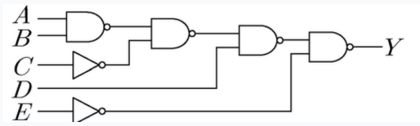
NoteBase

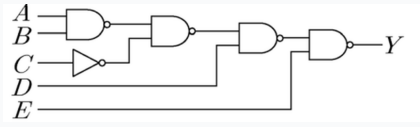
문 13. <보기>의 회로도와 동일한 출력을 가지는 회로도는?

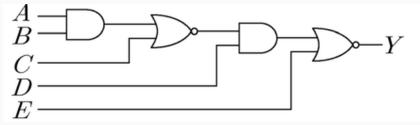
<보기>



① 

② 

③ 

④ 

정답 2번

출제 포인트

이 문제는 주어진 게이트 회로를 불 대수와 드모르간 법칙으로 바꾸어 같은 출력식인지 판단하는 문제이다. 기준 회로의 출력은 $Y = D(AB + C) + E$이다.

NoteBase

선지별 해설

① X
- 첫 번째 NAND 출력은 $(AB)'$이고, 다음 NAND 출력은 $((AB)'C)' = AB + C'$이다.
- 따라서 전체 출력은 $Y = D(AB + C') + E$가 되어 기준 회로의 $D(AB + C) + E$와 다르다.
- 틀린 부분은 C를 그대로 넣어 $AB + C$를 만들어야 하는데 1번은 C'가 반영된다는 점이다.

② O
- 첫 번째 NAND는 $(AB)'$, C 인버터는 C'를 만든다.
- 다음 NAND 출력은 $((AB)'C')' = AB + C$이므로 기준 회로의 첫 OR와 같은 기능이 된다.
- 그다음 NAND와 E 인버터를 거친 마지막 NAND 출력은 $(\{D(AB + C)\}'E')' = D(AB + C) + E$가 되어 기준 회로와 동일하다.

③ X
- 앞단은 2번과 같아서 $AB + C$까지는 맞게 만든다.
- 그러나 E를 반전하지 않고 마지막 NAND에 넣으므로 최종 출력은 $Y = D(AB + C) + E'$가 된다.
- 틀린 부분은 마지막 OR에 대응시키려면 E가 아니라 E'를 입력해야 한다는 점이다.

④ X
- 첫 OR 자리가 NOR 게이트이므로 앞단 출력이 $(AB + C)'$가 된다.
- 이어서 AND와 NOR를 거치면 전체 출력은 $Y = \{D(AB + C)' + E\}'$가 되어 기준 회로와 다르다.
- 틀린 부분은 $AB + C$를 만들어야 할 자리에 그 여집합 $(AB + C)'$를 만들고 있다는 점이다.

오답 포인트

NAND-NAND 구조는 게이트 모양만 보고 OR처럼 보인다고 판단하는 문제가 아니다. ①과 ③은 한 입력만 반전되어 C 또는 E의 극성이 바뀌고, ④는 OR 자리를 NOR로 바꾸어 전체 식의 여집합이 된다는 점이 함정이다.

다시 나오면 이렇게 판단

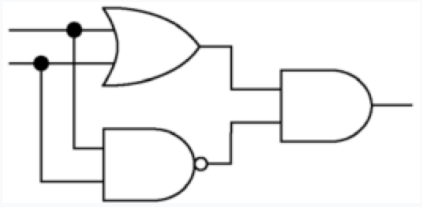
기준 회로를 먼저 $Y = D(AB + C) + E$처럼 식으로 적고, 대안 회로는 각 NAND와 NOR의 입력 반전이 쌍으로 상쇄되는지 본다. $((AB)'C')' = AB + C$처럼 드모르간 법칙으로 OR가 복원되면 같은 회로이다.

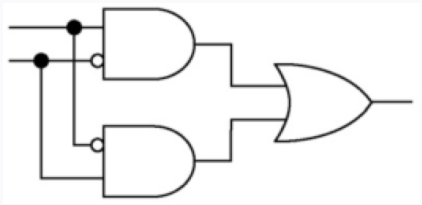
암기 포인트

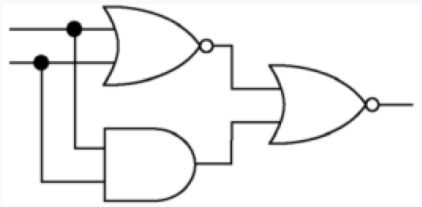
NAND에 반전된 두 입력을 넣으면 OR가 되고, OR를 NAND로 대체할 때는 대응 입력도 함께 반전되어야 원래 식이 유지된다.

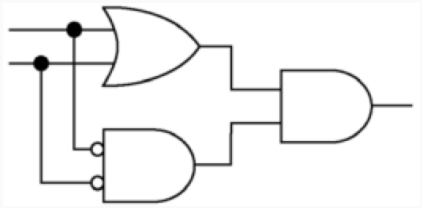
NoteBase

문 14. 반가산기의 '합'을 출력하는 용도로 사용할 수 없는 것은?

①


②


③


④


정답 4번

출제 포인트

이 문제는 각 게이트 조합을 식으로 바꾸어 반가산기의 합 출력인 $A \oplus B = \overline{A}B + A\overline{B}$가 되는지 판별하는 문제이다.

NoteBase

선지별 해설

① O

- 위 출력은 $A+B$, 아래 출력은 $\overline{AB}=\overline{A}+\overline{B}$이므로 최종 출력은 $(A+B)(\overline{A}+\overline{B})=\overline{A}B+A\overline{B}$이다. 반가산기의 합과 같으므로 사용할 수 있다.

② O

- 위 AND는 $A\overline{B}$, 아래 AND는 $\overline{A}B$이고 마지막 OR가 이를 더하므로 최종 출력은 $A\overline{B}+\overline{A}B$이다. XOR가 그대로 만들어지므로 사용할 수 있다.

③ O

- 위 NOR의 출력은 $\overline{A+B}=\overline{AB}$, 아래 AND의 출력은 AB이다. 마지막 NOR의 출력은 $\overline{\overline{AB}+AB}=(A+B)(\overline{A}+\overline{B})=\overline{A}B+A\overline{B}$이므로 역시 합 출력이 된다.

④ X

- 위 OR의 출력은 $A+B$, 아래 AND의 출력은 $\overline{AB}$이다. 최종 AND는 $(A+B)\overline{AB}=0$이므로 항상 0만 출력한다. 틀린 판단은 이 회로를 반가산기의 합 출력으로 사용할 수 있다고 보는 점이며, 올바르게는 합 출력이 아니라 상수 0 출력 회로이다.

오답 포인트

①과 ③은 게이트 수가 많아 보여도 식으로 정리하면 XOR가 된다. ④는 $(A+B)$와 $\overline{AB}$가 동시에 1이 될 수 없다는 점을 놓치면 그럴듯한 XOR 변형으로 오인하기 쉽다.

다시 나오면 이렇게 판단

합 출력 문제는 회로를 바로 식으로 바꾸어 최종식이 $\overline{A}B+A\overline{B}$ 또는 이에 동치인 형태인지 확인하면 된다. OR 출력과 보수항의 곱 형태가 보이면 분배법칙으로 XOR인지 검산할 수 있다.

암기 포인트

반가산기 합은 XOR이고, $(A+B)(\overline{A}+\overline{B})=\overline{A}B+A\overline{B}$, $(A+B)\overline{AB}=0$이다.

NoteBase

문 15. 이진수 $(1010.11)_2$를 십진수로 바꾼 값은?

① 10.2

② 10.11

③ 10.75

④ 10.3

정답 3번

출제 포인트

이 문제는 이진수의 각 자리를 2의 거듭제곱 자리값으로 풀어 십진수로 바꾸는 계산을 묻는 문제이다.

조건 정리

주어진 값: $(1010.11)_2$
구해야 할 값: 십진수 값
주의할 조건: 소수점 왼쪽은 $2^0, 2^1, 2^2, \ldots$, 소수점 오른쪽은 $2^{-1}, 2^{-2}, \ldots$ 자리값으로 계산한다.

사용할 공식

$(b_n b_{n-1} \ldots b_0 . b_{-1} b_{-2})_2 = \sum b_i 2^i$로 자리값을 전개한다.
이 문제는 정수부와 소수부가 함께 있는 이진수이므로 각 비트에 해당하는 2의 거듭제곱 값을 더하는 방식이 가장 직접적이다.

계산 과정

$(1010.11)_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2}$
$= 8 + 0 + 2 + 0 + 0.5 + 0.25 = 10.75$
따라서 십진수 값은 10.75이다.

선지별 해설

1번 X: 10.2는 소수부 $.11_2$를 십진 소수 0.2처럼 읽은 오답이다. 올바르게는 $.11_2 = 2^{-1} + 2^{-2} = 0.75$이다.
2번 X: 10.11은 이진수 표기를 십진수 표기처럼 그대로 옮긴 오답이다. 올바르게는 자리값을 계산해 10.75가 된다.
3번 O: 정수부는 $1010_2 = 8 + 2 = 10$, 소수부는 $.11_2 = 0.5 + 0.25 = 0.75$이므로 전체 값은 10.75이다.
4번 X: 10.3은 소수부의 각 비트를 십진 한 자리 값처럼 잘못 대응한 오답이다. 올바르게는 소수점 오른쪽 첫째 자리와 둘째 자리가 각각 $2^{-1}, 2^{-2}$이므로 0.75를 더해야 한다.

NoteBase

오답 포인트

이진수의 숫자 모양만 보고 10.11이나 10.2처럼 십진수 자릿값으로 읽으면 틀린다. 특히 소수점 오른쪽을 0.1, 0.01 자리가 아니라 2^{-1}, 2^{-2} 자리로 보는지가 핵심 함정이다.

다시 나오면 이렇게 판단

진법 변환 문제는 정수부와 소수부를 나눈 뒤 각 자리에 해당하는 거듭제곱 값을 적고 마지막에 모두 더하는 순서로 판단한다. 소수점 오른쪽은 항상 2^{-1}부터 시작하는지 먼저 확인하면 실수를 줄일 수 있다.

암기 포인트

이진수 소수점 오른쪽은 2^{-1}, 2^{-2}, 2^{-3} 자리이며, $.11_2 = 0.75$이다.

NoteBase

문 16. $\bar{B}\bar{C} + B\bar{C} + A\bar{B}C + \bar{A}BC$**를 최대로 간략화한 것으로 가장 옳은 것은?**

① $\bar{C} + A\bar{B}C + \bar{A}BC$

② $\bar{C} + A\bar{B} + \bar{A}BC$

③ $\bar{C} + A\bar{B} + \bar{A}B$

④ $\bar{C} + \bar{B}\bar{C} + BC$

정답 3번

출제 포인트

이 문제는 불 대수의 인수분해와 흡수법칙 $\overline{X} + XY = \overline{X} + Y$를 이용해 논리식을 최대로 간략화할 수 있는지를 묻는다.

조건 정리

주어진 식: $F = \overline{BC} + B\overline{C} + A\overline{B}C + \overline{A}BC$
구해야 할 것: 식 F를 더 이상 줄일 수 없는 형태로 바꾸는 것
주의할 조건: 먼저 $\overline{C}$를 공통인수로 묶고, 그다음 C가 붙은 항에 흡수법칙을 적용하는 순서가 핵심이다.

사용할 공식

1. $\overline{BC} + B\overline{C} = \overline{C}(\overline{B} + B) = \overline{C}$
2. $\overline{X} + XY = \overline{X} + Y$

이 문제는 앞의 두 항이 같은 $\overline{C}$를 가지므로 바로 합칠 수 있고, 남은 두 항은 C를 묶은 뒤 흡수법칙으로 C를 제거할 수 있는 구조이다.

계산 과정

1. $\overline{BC} + B\overline{C} = \overline{C}(\overline{B} + B) = \overline{C}$이다.
2. 따라서 $F = \overline{C} + A\overline{B}C + \overline{A}BC = \overline{C} + C(A\overline{B} + \overline{A}B)$이다.
3. 여기서 $\overline{X} + XY = \overline{X} + Y$에 $X = C$, $Y = A\overline{B} + \overline{A}B$를 대입하면 $F = \overline{C} + A\overline{B} + \overline{A}B$가 된다.

NoteBase

선지 판단

1번 X: $\overline{C} + A\overline{B}C + \overline{A}BC$는 첫 두 항만 정리한 중간 단계일 뿐 최종 간략형이 아니다. 잘못 남아 있는 표현은 뒤의 두 항에 붙은 C이며, 올바르게는 $\overline{C} + A\overline{B} + \overline{A}B$이다.
2번 X: $\overline{C} + A\overline{B} + \overline{A}BC$는 $A\overline{B}C$만 줄이고 $\overline{A}BC$는 그대로 둔 식이다. 잘못된 표현은 $\overline{A}BC$이며, 올바르게는 $\overline{A}B$이다.
3번 O: $\overline{C} + A\overline{B} + \overline{A}B$는 $\overline{C} + C(A\overline{B} + \overline{A}B)$에 흡수법칙을 적용한 최종식이므로 맞다.
4번 X: $\overline{C} + \overline{BC} + BC$는 $\overline{C}$가 이미 $\overline{BC}$를 포함하므로 둘째 항이 중복이고, 셋째 항 BC는 원식의 A 조건을 지워 버린다. 예를 들어 $A = 1, B = 0, C = 1$이면 원식은 1이지만 4번 식은 0이므로 동치가 아니며, 올바르게는 $\overline{C} + A\overline{B} + \overline{A}B$이다.

오답 포인트

첫 번째 축약인 $\overline{BC} + B\overline{C} = \overline{C}$까지만 하고 멈추면 1번에 끌리기 쉽다. 또 C가 붙은 두 항 중 한 항만 줄이면 2번이 되고, A가 걸린 조건을 무시한 채 BC로 바꾸면 4번처럼 다른 함수가 된다.

다시 나오면 이렇게 판단

불 대수 간략화는 공통인수를 먼저 묶고, $X + \overline{X} = 1$과 $\overline{X} + XY = \overline{X} + Y$처럼 항을 제거하는 법칙을 차례로 적용하면 된다. 각 단계 뒤에는 특정 변수 조건이 사라져도 원래 식과 동치인지 확인하면 된다.

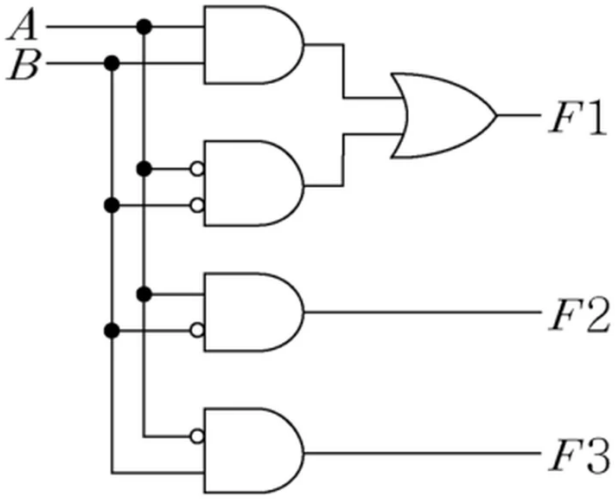
암기 포인트

$\overline{BC} + B\overline{C} = \overline{C}, \overline{C} + CY = \overline{C} + Y$ 순서로 줄어든다는 패턴을 기억해 두면 같은 유형을 빠르게 풀 수 있다.

NoteBase

문 17. <보기>의 비교기에서 논리 F1의 기능은?

<보기>



① A > B

② A = B

③ A < B

④ A ≧ B

정답 2번

출제 포인트

이 문제는 비교기 도식에서 입력 버블과 게이트 연결을 식으로 바꾸어 $F1 = AB + \overline{AB}$인지 판단하고, 그 결과가 $A = B$ 출력인지 구별하는 문제이다.

선지별 해설

① X: $A > B$ 출력은 1비트 비교기에서 $A\overline{B}$이다. F1은 위 AND의 AB와 아래 AND의 $\overline{AB}$를 OR한 값이므로 크기 비교 출력이 아니다. 올바르게는 $A > B$ 기능은 $A\overline{B}$를 내는 출력이다.

② O: F1은 AB와 $\overline{AB}$를 OR한 $AB + \overline{AB}$이다. 두 입력이 둘 다 1이거나 둘 다 0일 때만 1이 되므로 $A = B$를 나타낸다.

③ X: $A < B$ 출력은 $\overline{A}B$이다. F1은 두 입력이 같은 경우를 검출하는 식이므로 작은 경우만 따로 검출하는 출력이 아니다. 올바르게는 $A < B$ 기능은 $\overline{A}B$를 내는 출력이다.

④ X: $A \geq B$는 $A > B$ 또는 $A = B$이므로 $A\overline{B} + AB$처럼 10과 11을 포함해야 한다. F1은 00과 11에서만 1이 되는 $AB + \overline{AB}$이므로 $A \geq B$가 아니다. 올바르게는 $A \geq B$는 $A\overline{B} + AB$이다.

NoteBase

오답 포인트

입력선의 버블을 빼고 게이트 모양만 보면 OR가 붙어 있으므로 큰 경우까지 포함한다고 오해하기 쉽다. 이 문항은 각 AND 출력이 무엇인지 먼저 식으로 적어 보면 함정을 피할 수 있다.

다시 나오면 이렇게 판단

비교기 도식은 먼저 입력 버블로 보수를 반영하고, 다음에 각 AND의 출력을 항으로 적은 뒤, 마지막 OR를 적용하여 $AB + \overline{AB}$, $A\overline{B}$, $\overline{A}B$ 가운데 무엇인지 대응하면 된다.

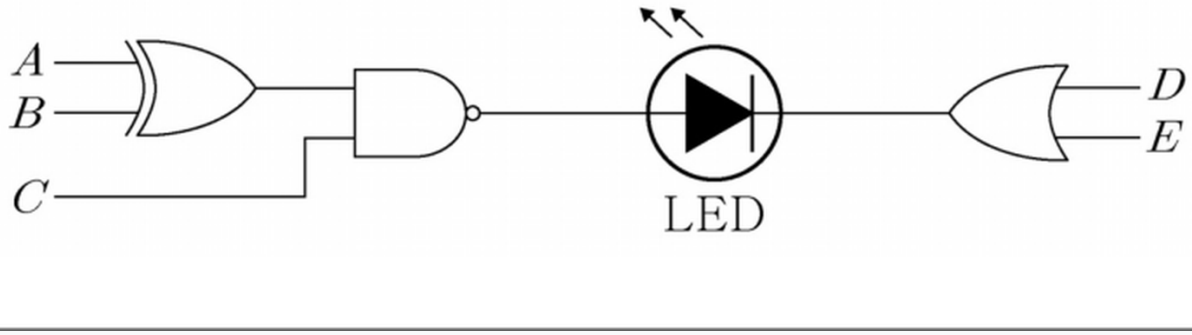
암기 포인트

1비트 비교기 기본식은 $A = B : AB + \overline{AB}$, $A > B : A\overline{B}$, $A < B : \overline{A}B$이다.

NoteBase

문 18. <보기>의 회로도에서 LED가 점등하기 위한 조건으로 가장 옳은 것은?

<보기>



① A: 1, B: 1, C: 1, D: 1, E: 0

② A: 1, B: 0, C: 1, D: 0, E: 0

③ A: 1, B: 1, C: 1, D: 0, E: 1

④ A: 0, B: 1, C: 0, D: 0, E: 0

정답 4번

출제 포인트

이 문제는 XOR 출력과 C를 NAND한 왼쪽 논리값, D와 E를 OR한 오른쪽 논리값을 각각 구한 뒤 LED의 순방향 점등 조건을 판단하는 문제이다.

선지별 해설

① X: A=1, B=1이므로 XOR 출력은 0이다. 여기에 C=1을 NAND하면 왼쪽 출력은 1이다. 오른쪽은 D OR E = 1 OR 0 = 1이므로 LED 양단이 모두 1이어서 전위차가 없다. 틀린 표현은 점등 가능하다는 판단이며, 올바르게는 오른쪽 출력이 0이어야 한다.

② X: A=1, B=0이므로 XOR 출력은 1이다. C=1이므로 NAND(1,1)=0이 되어 왼쪽 출력은 0이다. 오른쪽도 D OR E = 0이므로 LED 양단이 모두 0이어서 점등하지 않는다. 틀린 표현은 이 조건에서 LED가 켜진다는 판단이며, 올바르게는 C가 0이어야 왼쪽 출력이 1이 된다.

③ X: A=1, B=1이므로 XOR 출력은 0이고, C=1과 NAND하면 왼쪽 출력은 1이다. 그러나 오른쪽은 D OR E = 0 OR 1 = 1이므로 양단이 모두 1이어서 점등하지 않는다. 틀린 표현은 E=1이어도 점등할 수 있다는 판단이며, 올바르게는 오른쪽 출력이 0이어야 한다.

④ O: A=0, B=1이므로 XOR 출력은 1이다. C=0이므로 NAND(1,0)=1이 되어 왼쪽 출력은 1이다. 오른쪽은 D OR E = 0 OR 0 = 0이므로 LED의 왼쪽이 1, 오른쪽이 0이 되어 순방향 전류가 흘러 점등한다.

NoteBase

오답 포인트

XOR를 일반 OR로 착각하거나 NAND에서 입력 하나가 0이면 출력이 1이 된다는 성질을 놓치면 ②와 ④를 혼동하기 쉽다. 또한 LED는 한쪽 논리값만 1이라고 켜지는 것이 아니라, 방향에 맞는 전위차가 형성되어야 점등된다.

다시 나오면 이렇게 판단

왼쪽 블록과 오른쪽 블록의 출력을 따로 끝까지 계산한 뒤 LED 방향 기준으로 왼쪽 1, 오른쪽 0인지 확인하면 된다. 먼저 A와 B가 같은지 다른지로 XOR를 판단하고, 그다음 C와 함께 NAND를 계산한 뒤 마지막으로 D OR E를 본다.

암기 포인트

XOR는 다르면 1, NAND는 0이 하나라도 있으면 1, LED는 순방향 전위차가 있을 때만 켜진다.

NoteBase

문 19. 입력클럭 신호의 주파수가 2MHz이고, T 플립플롭 4개를 직렬로 연결했을 때 마지막 단에서 얻을 수 있는 신호의 주기는?

① 1μS
② 4μS
③ 8μS
④ 16μS

정답 3번

출제 포인트

이 문제는 T 플립플롭 직렬 연결에서 각 단이 입력 주파수를 2분주한다는 성질로 마지막 단의 주기를 구하는 문제이다.

조건 정리

주어진 값: 입력 클럭 주파수는 2MHz이다.
구해야 할 값: T 플립플롭 4개를 직렬 연결했을 때 마지막 단 출력의 주기이다.
주의할 조건: T 플립플롭 한 단계가 지날 때마다 출력 주파수는 절반이 되므로 4단 직렬 연결이면 전체적으로 16분주가 된다.

사용할 공식

입력 주기는 $T_{in} = 1/f_{in}$이고, T 플립플롭 4개 직렬 연결의 마지막 단 주파수는 $f_{out} = f_{in}/2^4$이다. 따라서 마지막 단의 주기는 $T_{out} = 1/f_{out} = 2^4/f_{in}$이다.
이 공식을 쓰는 이유: T 플립플롭은 토글 동작으로 입력 펄스를 받을 때마다 출력 주파수를 절반으로 만들기 때문이다.

계산 과정

1. 입력 주파수는 $f_{in} = 2MHz = 2 \times 10^6 Hz$이다.
2. 입력 주기는 $T_{in} = 1/f_{in} = 1/(2 \times 10^6) = 0.5\mu s$이다.
3. T 플립플롭 4개 직렬 연결이면 마지막 단은 $2^4 = 16$분주이므로 출력 주기는 $T_{out} = 16 \times 0.5\mu s = 8\mu s$이다.

NoteBase

선지별 해설

① X: 1μs는 입력 주기 0.5μs를 한 번만 2배 한 값으로, T 플립플롭 1개만 지난 경우에 해당한다. 이 문항은 4개 직렬 연결이므로 틀리며, 올바르게는 8μs이다.
② X: 4μs는 0.5μs × 8로 8분주, 즉 3단까지의 출력 주기에 해당한다. 마지막 4단 출력까지 반영해야 하므로 틀리며, 올바르게는 8μs이다.
③ O: T 플립플롭 4개 직렬 연결은 16분주이므로 0.5μs × 16 = 8μs가 된다. 계산 결과와 일치하므로 옳다.
④ X: 16μs는 0.5μs × 32로 한 단계 더 분주한 값이다. 4개 직렬 연결을 5개로 잘못 세면 나오는 값이므로 틀리며, 올바르게는 8μs이다.

오답 포인트

입력 주파수를 바로 분주하지 않고 주기로 바꾼 뒤 각 단계의 2분주를 누적해야 한다는 점이 함정이다. ②는 3단까지만 계산한 경우이고, ④는 플립플롭 개수를 한 단계 더 센 경우에 자주 나온다.

다시 나오면 이렇게 판단

플립플롭 직렬 연결 문제는 먼저 입력 주기를 구한 뒤, 마지막 단까지의 분주비가 2^n인지 확인하고 그만큼 주기를 곱해 판단한다.

암기 포인트

T 플립플롭 n단 직렬 연결의 마지막 단은 2^n분주, 출력 주기는 입력 주기의 2^n배이다.

NoteBase