

Digital Electronics (4321102) - Summer 2023 Solution (Gujarati)

Milav Dabgar

August 07, 2023

પ્રશ્ન 1(ા) [3 ગુણ]

બુલિયન એલ્જીબ્રા માટેના ડે-મોર્ગનના નિયમ સમજાવો

જવાબ

ડે-મોર્ગનના નિયમમાં બે કાયદા છે જે AND, OR અને NOT કિયાઓ વચ્ચેના સંબંધને દર્શાવે છે:

કાયદો 1: સરવાળાના પૂરકની કિંમત પૂરકના ગુણાકાર બરાબર હોય છે

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

કાયદો 2: ગુણાકારના પૂરકની કિંમત પૂરકના સરવાળા બરાબર હોય છે

$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

કોષ્ટક 1. ડે-મોર્ગનના નિયમની ચકાસણી

A	B	A+B	$\overline{A + B}$	\overline{A}	\overline{B}	$\overline{A} \cdot \overline{B}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

મેમરી ટ્રીક

"OR પર NOT થાય AND, AND પર NOT થાય OR"

પ્રશ્ન 1(બ) [4 ગુણ]

નીચેના ડેસિમલ નંબરને બાયનરી અને ઓક્ટલ નંબરમાં ફેરવો (i) 215 (ii) 59

જવાબ

બાયનરી રૂપાંતર:

215 માટે:

- 2 વડે ભાગ કરો: $215/2 = 107$ શેષ 1
- $107/2 = 53$ શેષ 1
- $53/2 = 26$ શેષ 1
- $26/2 = 13$ શેષ 0
- $13/2 = 6$ શેષ 1
- $6/2 = 3$ શેષ 0

- $3/2 = 1$ શેષ 1
- $1/2 = 0$ શેષ 1
- તેથી, $(215)_{10} = (11010111)_2$

59 માટે:

- 2 વડે ભાગ કરો: $59/2 = 29$ શેષ 1
- $29/2 = 14$ શેષ 1
- $14/2 = 7$ શેષ 0
- $7/2 = 3$ શેષ 1
- $3/2 = 1$ શેષ 1
- $1/2 = 0$ શેષ 1
- તેથી, $(59)_{10} = (111011)_2$

ઓકટલ રૂપાંતર:**215 માટે:**

- 8 વડે ભાગ કરો: $215/8 = 26$ શેષ 7
- $26/8 = 3$ શેષ 2
- $3/8 = 0$ શેષ 3
- તેથી, $(215)_{10} = (327)_8$

59 માટે:

- 8 વડે ભાગ કરો: $59/8 = 7$ શેષ 3
- $7/8 = 0$ શેષ 7
- તેથી, $(59)_{10} = (73)_8$

કોષ્ટક 2. સંખ્યા રૂપાંતર સારાંશ

ડેસિમલ	બાયનરી	ઓકટલ
215	11010111	327
59	111011	73

મેમરી ટ્રીક

"આધાર વડે ભાગો, શેષ નીચેથી ઉપર વાંચો"

પ્રશ્ન 1(ક)(I) [2 ગુણ]

ડેસિમલ, બાયનરી, ઓકટલ અને હેકજાડેસિમલ નંબર સિસ્ટમનો બેઝ લખો

જવાબ**કોષ્ટક 3.** સંખ્યા પદ્ધતિના આધાર

સંખ્યા પદ્ધતિ	આધાર
ડેસિમલ	10
બાયનરી	2
ઓકટલ	8
હેકજાડેસિમલ	16

મેમરી ટ્રીક

"ડ-બા-અ-હે: 10-2-8-16"

પ્રશ્ન 1(ક)(II) [2 ગુણ]

$$(147)_{10} = (\underline{\hspace{2cm}})_2 = (\underline{\hspace{2cm}})_{16}$$

જવાબ

ડિસિમલથી બાયનરી રૂપાંતર:

- $147/2 = 73$ શેષ 1
- $73/2 = 36$ શેષ 1
- $36/2 = 18$ શેષ 0
- $18/2 = 9$ શેષ 0
- $9/2 = 4$ શેષ 1
- $4/2 = 2$ શેષ 0
- $2/2 = 1$ શેષ 0
- $1/2 = 0$ શેષ 1
- તેથી, $(147)_{10} = (10010011)_2$

ડિસિમલથી હેકાડિસિમલ રૂપાંતર:

- બાયનરી અંકોને 4ના સમૂહમાં વિભાજિત કરો: 1001 0011
- દરેક સમૂહને હેકસમાં રૂપાંતરિત કરો: $1001 = 9, 0011 = 3$
- તેથી, $(147)_{10} = (93)_{16}$

કોષ્ટક 4. રૂપાંતર પરિણામ

ડિસિમલ	બાયનરી	હેકાડિસિમલ
147	10010011	93

મેમરી ટ્રીક

"હેકસ માટે જમણેથી 4ના સમૂહમાં વિભાજિત કરો"

પ્રશ્ન 1(ક)(III) [3 ગુણ]

નીચેના બાયનરી કોડનું ગ્રે કોડમાં રૂપાંતર કરો (i) 1011 (ii) 1110

જવાબ

બાયનરીથી ગ્રે કોડ રૂપાંતર પ્રક્રિયા:

1. ગ્રે કોડનો MSB (ડાબી બાજુનો બિટ) બાયનરી કોડના MSB જેવો જ હોય છે
2. ગ્રે કોડના અન્ય બિટ્સ બાયનરી કોડના આસપાસના બિટ્સને XOR કરીને મેળવવામાં આવે છે

1011 માટે:

- ગ્રે કોડનો MSB = બાયનરી કોડનો MSB = 1
- બીજો બિટ = $1 \oplus 0 = 1$
- ત્રીજો બિટ = $0 \oplus 1 = 1$
- ચોથો બિટ = $1 \oplus 1 = 0$
- તેથી, $(1011)_2 = (1110)_{gray}$

1110 માટે:

- ગ્રે કોડનો MSB = બાયનરી કોડનો MSB = 1
- બીજો બિટ = $1 \oplus 1 = 0$
- ત્રીજો બિટ = $1 \oplus 1 = 0$
- ચોથો બિટ = $1 \oplus 0 = 1$
- તેથી, $(1110)_2 = (1001)_{gray}$

કોષ્ટક 5. બાયનરીથી ગ્રે કોડ રૂપાંતર

બાયનરી	રૂપાંતર પદ્ધતિ	ગ્રે કોડ
1011	$1, 1 \oplus 0 = 1, 0 \oplus 1 = 1, 1 \oplus 1 = 0$	1110
1110	$1, 1 \oplus 1 = 0, 1 \oplus 1 = 0, 1 \oplus 0 = 1$	1001

મેમરી ટ્રીક

"પહેલો રાખો, બાકીના XOR કરો"

પ્રશ્ન 1 [OR] (I) [2 ગુણ]

BCD અને ASCII નું ફૂલક્રોમ્બ લખો

જવાબ

કોષ્ટક 6. BCD અને ASCII નું પૂર્ણ નામ

સંક્ષિપ્ત રૂપ	પૂર્ણ નામ
BCD	Binary Coded Decimal
ASCII	American Standard Code for Information Interchange

મેમરી ટ્રીક

"બાયનરી કોડેડ ડેસિમલ, અમેરિકન સ્ટાન્ડર્ડ કોડ ફોર ઇન્ફોર્મેશન ઇન્ટરચેન્જ"

પ્રશ્ન 1 [OR] (II) [2 ગુણ]

નીચેના બાયનરી નંબરના 1's અને 2's કોમ્પલમેન્ટ શોધો (i) 1010 (ii) 1011

જવાબ

1's કોમ્પલમેન્ટ: બધા બિટ્સ ઉલટાવો (0 ને 1 અને 1 ને 0 માં બદલો)

2's કોમ્પલમેન્ટ: 1's કોમ્પલમેન્ટ લો અને 1 ઉમેરો

1010 માટે:

- 1's કોમ્પલમેન્ટ: 0101
- 2's કોમ્પલમેન્ટ: $0101 + 1 = 0110$

1011 માટે:

- 1's કોમ્પલમેન્ટ: 0100
- 2's કોમ્પલમેન્ટ: $0100 + 1 = 0101$

કોષ્ટક 7. કોમ્પલમેન્ટ પરિણામો

બાયનરી	1's કોમ્પલમેન્ટ	2's કોમ્પલમેન્ટ
1010	0101	0110
1011	0100	0101

મેમરી ટ્રીક

"1's માટે બધા બિટ ઉલટાવો, 2's માટે એક ઉમેરો"

પ્રશ્ન 1 [OR] (III) [3 ગુણ]

2's કોમ્પિલમેન્ટ મેથડથી બાદબાકી કરો (i) $(110110)_2 - (101010)_2$

જવાબ

2's કોમ્પિલમેન્ટ પદ્ધતિથી બાદબાકી માટે:

- બાદ થનાર સંખ્યાનો 2's કોમ્પિલમેન્ટ શોધો
- તેને મૂળ સંખ્યામાં ઉમેરો
- બિટ વિડ્યથની બહારના કેરીને છોડી દો

બાદબાકી: $(110110)_2 - (101010)_2$

પગલું 1: 101010 નો 2's કોમ્પિલમેન્ટ શોધો

- 101010 નો 1's કોમ્પિલમેન્ટ = 010101
- 2's કોમ્પિલમેન્ટ = $010101 + 1 = 010110$

પગલું 2: $110110 + 010110$ ઉમેરો

$$\begin{array}{r} 11111 \\ 110110 \\ + 010110 \\ \hline 001100 \end{array}$$

પગલું 3: પરિણામ $001100 = (12)_{10}$

કોષ્ક 8. બાદબાકી પ્રક્રિયા

પગલું	કિયા	પરિણામ
1	101010 નો 2's કોમ્પિલમેન્ટ	010110
2	$110110 + 010110$ ઉમેરો	001100
3	અંતિમ પરિણામ (ડેસિમલ)	12

મેમરી ટ્રીક

"બાદનારનો કોમ્પિલમેન્ટ લો, ઉમેરો, કેરી ભૂલી જાઓ"

પ્રશ્ન 2(અ) [3 ગુણ]

NAND ગેટનો જ ઉપયોગ કરી AND, OR અને NOT ગેટની લોજિક સર્કિટ બનાવો

જવાબ

AND ગેટ NAND ગેટથી:

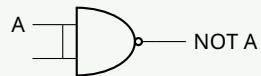
- AND ગેટ = NAND ગેટ પછી NOT ગેટ (NAND ગેટ)

OR ગેટ NAND ગેટથી:

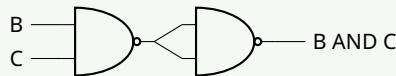
- OR ગેટ = બંને ઇનપુટને NOT (NAND ગેટ) લાગુ કરો, પછી તે પરિણામોને NAND કરો

NOT ગેટ NAND ગેટથી:

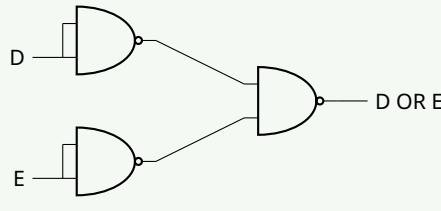
- NOT ગેટ = NAND ગેટ જેમાં બંને ઇનપુટ જોડાયેલા હોય



NOT Gate



AND Gate



OR Gate

આકૃતિ 1. Universal Gates implementation

મેમરી ટ્રીક

"NOT માટે એક NAND, AND માટે બે NAND, OR માટે ત્રણ NAND"

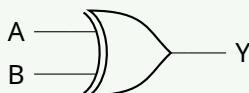
પ્રશ્ન 2(બ) [4 ગુણ]

નીચેના લોજિક ગેટનો લોજિક સિમ્બોલ, ટુથ ટેબલ અને સમીકરણ લખો/દોરો (i) XOR ગેટ (ii) OR ગેટ

જવાબ

XOR ગેટ:

લોજિક સિમ્બોલ:



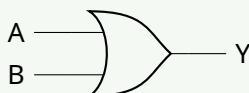
ટુથ ટેબલ:

A	B	Y ($A \oplus B$)
0	0	0
0	1	1
1	0	1
1	1	0

બુલિયન સમીકરણ: $Y = A \oplus B = A'B + AB'$

OR ગેટ:

લોજિક સિમ્બોલ:



ટુથ ટેબલ:

A	B	Y (A+B)
0	0	0
0	1	1
1	0	1
1	1	1

બુલિયન સમીકરણ: $Y = A + B$

મેમરી ટ્રીક

"XOR: એક્સક્લુસિવ OR - એક અથવા બીજું પણ બંને નહીં; OR: એક અથવા બીજું અથવા બંને"

પ્રશ્ન 2(ક)(િ) [૩ ગુણ]

બુલિયન સમીકરણ $Y = A + B[AC + (B + C\bar{D})D]$ ને algebraic મેથડથી સરળ બનાવો

જવાબ

પગલાંવાર સરળીકરણ:

$$\begin{aligned}
 Y &= A + B[AC + (B + \bar{C})D] \\
 Y &= A + B[AC + BD + \bar{C}D] \\
 Y &= A + BAC + BBD + B\bar{C}D \\
 Y &= A + ABC + BD + B\bar{C}D && (\text{કારણ કે } BB = B) \\
 Y &= A + AC + BD + B\bar{C}D && (\text{અવશોષણ: } A + AB = A) \\
 Y &= A + BD + B\bar{C}D && (\text{અવશોષણ: } A + AC = A) \\
 Y &= A + BD(1 + \bar{C}) \\
 Y &= A + BD && (\text{કારણ કે } 1 + X = 1)
 \end{aligned}$$

અંતિમ સમીકરણ: $Y = A + BD$

કોષ્ટક 9. સરળીકરણ પગલાં

પગલું	સમીકરણ	લાગુ પડેલ નિયમ
1	$A + B[AC + (B + \bar{C})D]$	મૂળ
2	$A + B[AC + BD + \bar{C}D]$	વિતરણ
3	$A + BAC + BBD + B\bar{C}D$	વિતરણ
4	$A + ABC + BD + B\bar{C}D$	આઇડેમ્પોટન (BB = B)
5	$A + AC + BD + B\bar{C}D$	અવશોષણ
6	$A + BD + B\bar{C}D$	અવશોષણ (A + AC = A)
7	$A + BD(1 + \bar{C})$	ફેક્ટરિંગ
8	$A + BD$	પૂર્ક નિયમ

મેમરી ટ્રીક

"આઇડેમ્પોટનસ, અવશોષણ, અને પૂર્ક પેટર્ન માટે હંમેશા તપાસો"

પ્રશ્ન 2(ક) (II) [4 ગુણ]

બુલિયન સમીકરણ $F(A,B,C) = \bar{m}(0, 2, 3, 4, 5, 6)$ ને Karnaugh Map ની મદદથી સરળ બનાવો

જવાબ

$F(A,B,C) = \bar{m}(0, 2, 3, 4, 5, 6)$ માટે K-map બનાવો:

		BC	00	01	11	10	
		A	0	1	0	0	1
A	0	1					
	1	1	1		0		1

1 ની ગુપ્તિ કરો:

- ગુપ્ત 1: $m(0,4)$ - $B'C'$ સાથે સંબંધિત
- ગુપ્ત 2: $m(2,6)$ - BC' સાથે સંબંધિત
- ગુપ્ત 3: $m(4,5)$ - AB' સાથે સંબંધિત

સરળ સમીકરણ: $F(A, B, C) = B'C' + BC' + AB'$

વધુ સરળ કરીએ: $F = C'(B' + B) + AB' = C' + AB'$

મેમરી ટ્રીક

"2ની પાવરમાં આસપાસના 1 ને ગુપ્ત કરો"

પ્રશ્ન 2 [OR] (અ) [3 ગુણ]

NOR ગેટનો જ ઉપયોગ કરી AND, OR અને NOT ગેટની લોજિક સર્કિટ બનાવો

જવાબ

NOT ગેટ NOR ગેટથી:

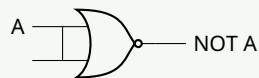
- NOT ગેટ = NOR ગેટ જેમાં બંને ઇનપુટ જોડાયેલા હોય

AND ગેટ NOR ગેટથી:

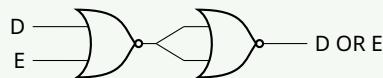
- AND ગેટ = બંને ઇનપુટને NOT (NOR ગેટ) લાગુ કરો, પછી તે પરિણામોને ફરીથી NOR કરો

OR ગેટ NOR ગેટથી:

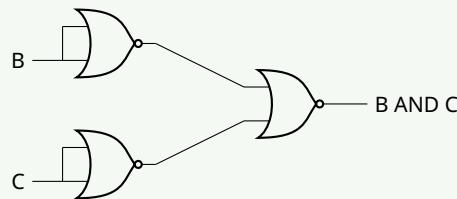
- OR ગેટ = NOR ગેટ પછી NOT ગેટ (NOR ગેટ)



NOT Gate



OR Gate



AND Gate

આકૃતિ 2. Universal Gates implementation (NOR)

મેમરી ટ્રીક

"NOT માટે એક NOR, દરેકને NOT કરીને NOR કરો AND માટે, વે વાર NOR કરો OR માટે"

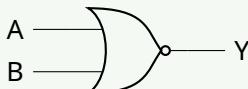
પ્રશ્ન 2 [OR] (બ) [4 ગુણ]

નીચેના લોજિક ગેટનો લોજિક સિમ્બોલ, ટુથ ટેબલ અને સમીકરણ લખો/દોરો (i) NOR ગેટ (ii) AND ગેટ

જવાબ

NOR ગેટ:

લોજિક સિમ્બોલ:



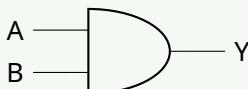
ટુથ ટેબલ:

A	B	$Y (A+B)'$
0	0	1
0	1	0
1	0	0
1	1	0

બુલિયન સમીકરણ: $Y = (A + B)' = A'B'$

AND ગેટ:

લોજિક સિમ્બોલ:



ટુથ ટેબલ:

A	B	Y (A·B)
0	0	0
0	1	0
1	0	0
1	1	1

બુલિયન સમીકરણ: $Y = A \cdot B$

મેમરી ટ્રીક

"NOR: NOT OR - ન તો એક કે ન તો બીજું; AND: બંને 1 હોવા જ જોઈએ"

પ્રશ્ન 2 [OR] (ક) [૭ ગુણ]

ઉપરની લોજિક સક્રિય માટે બુલિયન સમીકરણ લખો. આ સમીકરણને સરળ બનાવો અને આ સરળ સમીકરણની લોજિક AND-OR-Invert મેથડથી દોરો

જવાબ

પગલું 1: સક્રિયમાંથી બુલિયન સમીકરણ લખો: $Q = (A + B) \cdot (B + C \cdot ((B + C)'))$

$$Q = (A + B) \cdot (B + C \cdot (B' \cdot C'))$$

$$Q = (A + B) \cdot (B + C \cdot B' \cdot C')$$

પગલું 2: સમીકરણને સરળ બનાવો:

- નોંધ કરો કે $C \cdot C' = 0$
- તેથી, $C \cdot B' \cdot C' = 0$
- એટલે $Q = (A + B) \cdot (B + 0) = (A + B) \cdot B = A \cdot B + B \cdot B = A \cdot B + B = B + A \cdot B = B(1 + A) = B$

પગલું 3: અંતિમ સરળ સમીકરણ: $Q = B$

પગલું 4: AND-OR-Invert દ્વારા $Q = B$ નું અમલીકરણ:

- આ ફક્ત ઇનપુટ B થી આઉટપુટ Q સુધીનો એક તાર છે

B ————— Q

કોષ્ટક 10. સરળીકરણ પગલાં

પગલું	સમીકરણ	સરળીકરણ
1	$(A + B) \cdot (B + C \cdot ((B + C)'))$	મૂળ સમીકરણ
2	$(A + B) \cdot (B + C \cdot B' \cdot C')$	ડી મોર્ગનનો નિયમ લાગુ કરવો
3	$(A + B) \cdot (B + 0)$	$C \cdot C' = 0$
4	$(A + B) \cdot B$	સરળીકરણ
5	$A \cdot B + B \cdot B$	વિતરણ ગુણધર્મ
6	$A \cdot B + B$	આઇડેમ્પોટન્ટ ગુણધર્મ ($B \cdot B = B$)
7	$B(1 + A)$	ફેક્ટરિંગ
8	B	$1 + A = 1$

મેમરી ટ્રીક

"જ્યારે પૂરક ચલ ગુણાકાર કરે, તેઓ શૂન્ય થાય"

પ્રશ્ન 3(અ) [3 ગુણ]

કોમ્બીનેશનલ સર્કિટની વ્યાખ્યા લખો. કોમ્બીનેશનલ સર્કિટના બે ઉદાહરણ લખો

જવાબ

કોમ્બીનેશનલ સર્કિટ: એક ડિજિટલ સર્કિટ જેનું આઉટપુટ માત્ર વર્તમાન ઇનપુટ મૂલ્યો પર આધારિત હોય છે અને અગાઉના ઇનપુટ અથવા સ્થિતિઓ પર નહીં. કોમ્બીનેશનલ સર્કિટમાં કોઈ મેમરી અથવા ફીડબેક હોતા નથી.

મુખ્ય લક્ષણો:

- આઉટપુટ ફક્ત વર્તમાન ઇનપુટ પર આધારિત હોય છે
- કોઈ મેમરી એલિમેન્ટ નથી
- કોઈ ફીડબેક પાથ નથી

કોમ્બીનેશનલ સર્કિટના ઉદાહરણો:

- મલ્ટિપ્લેક્સર (MUX)
- ડિકોડર
- એડર/સબટ્રેક્ટર
- એનકોડર
- કમ્પોરેટર

કોષ્ટક 11. કોમ્બીનેશનલ vs સિક્વેન્શિયલ સર્કિટ

લક્ષણ	કોમ્બીનેશનલ સર્કિટ	સિક્વેન્શિયલ સર્કિટ
મેમરી	ના	હા
ફીડબેક	ના	સામાન્ય રીતે
આઉટપુટ આધારિત	માત્ર વર્તમાન ઇનપુટ	વર્તમાન અને અગાઉના ઇનપુટ
ઉદાહરણો	મલ્ટિપ્લેક્સર, એડર	ફિલાપ-ફલોપ, કાઉન્ટર

મેમરી ટ્રીક

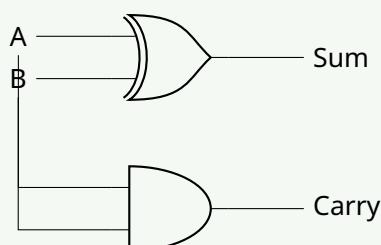
"કોમ્બીનેશનલ સર્કિટ: વર્તમાન આવે, વર્તમાન જાય - કોઈ યાદ નહીં"

પ્રશ્ન 3(બ) [4 ગુણ]

લોજિક સર્કિટ અને ટુથ ટેબલની મદદથી હાફ એડર સમજાવો

જવાબ

હાફ એડર: એક કોમ્બીનેશનલ સર્કિટ જે બે બાયનરી અંકો ઉમેરે છે અને સમ અને કેરી આઉટપુટ ઉત્પન્ન કરે છે.
લોજિક સર્કિટ:



ટુથ ટેબલ:

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

બુલિયન સમીકરણ:

- $\text{Sum} = A \oplus B = A'B + AB'$
- $\text{Carry} = A \cdot B$

મર્યાદાઓ:

- ત્રણ બાયનરી અંકો ઉમેરી શકતા નથી
- અગાઉના તબક્કામાંથી કેરી ઇનપુટ સમાવી શકતા નથી

મેમરી ટ્રીક

"XOR સમને માટે, AND કેરીને માટે"

પ્રશ્ન 3(ક)(I) [3 ગુણ]**મલિટિપ્લેક્સર ટૂકમાં સમજાવો****જવાબ**

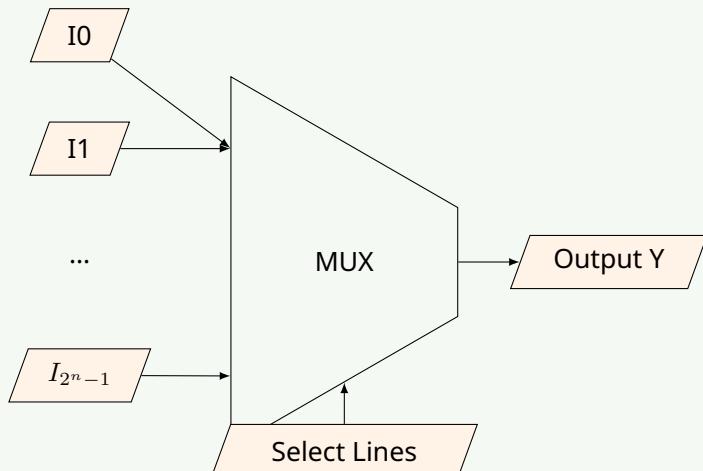
મલિટિપ્લેક્સર (MUX): એક કોમ્પ્યુનેશનલ સર્કિટ જે સિલેક્ટ લાઇન આધારે અનેક ઇનપુટ સિગ્નલ્સમાંથી એકને પસંદ કરે છે અને તેને એક આઉટપુટ લાઇન પર મોકલે છે.

મુખ્ય લક્ષણો:

- ડિજિટલ સ્વિચ તરીકે કાર્ય કરે છે
- 2^n ડેટા ઇનપુટ, n સિલેક્ટ લાઇન, અને 1 આઉટપુટ ધરાવે છે
- સિલેક્ટ લાઇન નક્કી કરે છે કે કે કિ કિંદું ઇનપુટ આઉટપુટથી જોડાયેલું છે

સામાન્ય મલિટિપ્લેક્સર:

- 2:1 MUX (1 સિલેક્ટ લાઇન)
- 4:1 MUX (2 સિલેક્ટ લાઇન)
- 8:1 MUX (3 સિલેક્ટ લાઇન)

મૂળભૂત રૂથના:**ઉપયોગો:**

- ડેટા રાઉટિંગ
- ડેટા પસંદગી
- પેરેલલથી સીરિયલ રૂપાંતર

- બુલિયન ફંક્શનનું અમલીકરણ

મેમરી ટ્રીક

"ઘણા ઇન, સિલેક્શન પસંદ કરે, એક આઉટ"

પ્રશ્ન 3(ક)(II) [4 ગુણ]

8:1 મલ્ટિપ્લિક્સર ડિઝાઇન કરો. તેનું ટૃથ ટેબલ લખો અને લોજિક સર્કિટ દોરો

જવાબ

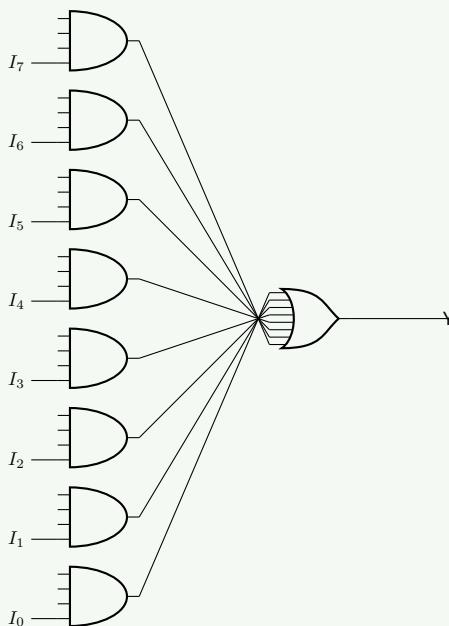
8:1 મલ્ટિપ્લિક્સર ડિઝાઇન:

- 8 ડૉટા ઇનપુટ (I_0 થી I_7)
- 3 સિલેક્ટ લાઇન (S_2, S_1, S_0)
- 1 આઉટપુટ (Y)

ટૃથ ટેબલ:

S_2	S_1	S_0	આઉટપુટ Y
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7

બુલિયન સમીક્ષણ: $Y = S'_2 S'_1 S'_0 I_0 + S'_2 S'_1 S_0 I_1 + S'_2 S_1 S'_0 I_2 + S'_2 S_1 S_0 I_3 + S_2 S'_1 S'_0 I_4 + S_2 S'_1 S_0 I_5 + S_2 S_1 S'_0 I_6 + S_2 S_1 S_0 I_7$



મેમરી ટ્રીક

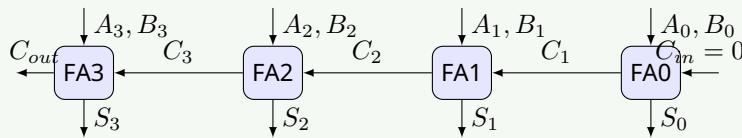
"આદ ઇનપુટ, ત્રણ સિલેક્ટ, ડિકોડ કરો અને આઉટપુટ મેળવવા OR કરો"

પ્રશ્ન 3 [OR] (અ) [3 ગુણ]

4-bit બાયનરી પેરેલલ એડરનો બ્લોક ડાયાગ્રામ દોરો

જવાબ

4-bit બાયનરી પેરેલલ એડર: બે 4-bit બાયનરી નંબર ઉમેરતી અને 4-bit સરવાળો અને એક કેરી આઉટપુટ ઉત્પન્ન કરતી સર્કિટ.



ઘટકો:

- ચાર કૂલ એડર (FA) કેરક્કડમાં જોડાયેલા
- દરેક FA સંબંધિત બિટ્સ અને અગાઉના તબક્કાની કેરી ઉમેરે છે
- પ્રારંભિક કેરી-ઇન (Cin) સામાન્ય રીતે 0 હોય છે

મેમરી ટ્રીક

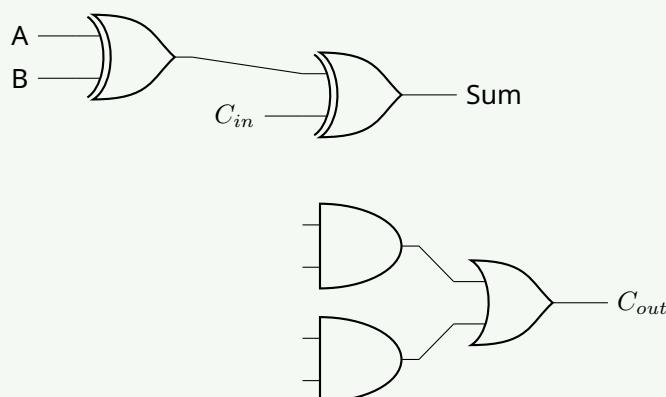
"ચાર FA જોડાયેલા, કેરીઓ વચ્ચેથી પસાર થાય છે"

પ્રશ્ન 3 [OR] (બ) [4 ગુણ]

લોજિક સર્કિટ અને ટુથ ટેબલની મદદથી કૂલ એડર સમજાવો

જવાબ

કૂલ એડર: એક કોમ્બીનેશનલ સર્કિટ જે ત્રણ બાયનરી અંક (બે ઇનપુટ અને એક કેરી-ઇન) ઉમેરે છે અને સરવાળો અને કેરી આઉટપુટ ઉત્પન્ન કરે છે.



ટુથ ટેબલ:

A	B	C_{in}	Sum	C_{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

બુલિયન સમીકરણ:

- $\text{Sum} = A \oplus B \oplus C_{in}$
- $C_{out} = AB + C_{in}(A \oplus B)$

મેરી ટ્રીક

"ત્રણોય XOR કરો સમ માટે, ANDને OR કરો કેરી માટે"

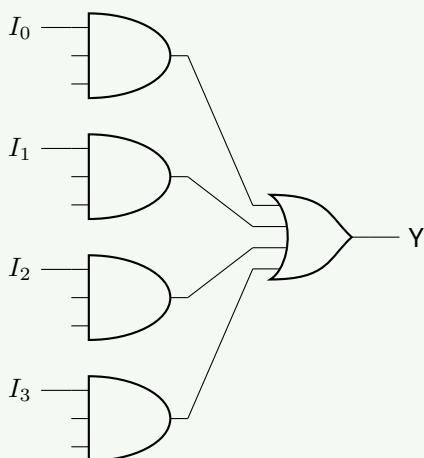
પ્રશ્ન 3 [OR] (ક) (I) [3 ગુણ]

લોજિક સર્કિટ અને ટુથ ટેબલની મદદથી 4:1 માલ્ટિપ્લિક્સર સમજાવો

જવાબ

4:1 માલ્ટિપ્લિક્સર: એક ડિજિટલ સ્વિચ જે બે સિલેક્ટ લાઇન્સના આધારે ચાર ઇનપુટ લાઇન્સમાંથી એકને પસંદ કરે છે અને તેને આઉટપુટથી જોડે છે.

લોજિક સર્કિટ:



ટુથ ટેબલ:

S_1	S_0	આઉટપુટ Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

બુલિયન સમીકરણ: $Y = S'_1 S'_0 I_0 + S'_1 S_0 I_1 + S_1 S'_0 I_2 + S_1 S_0 I_3$

મેમરી ટ્રીક

"બે સિલેક્ટ લાઇન ચાર ઇનપુટમાંથી એક પસંદ કરે છે"

પ્રશ્ન 3 [OR] (ક) (II) [4 ગુણ]

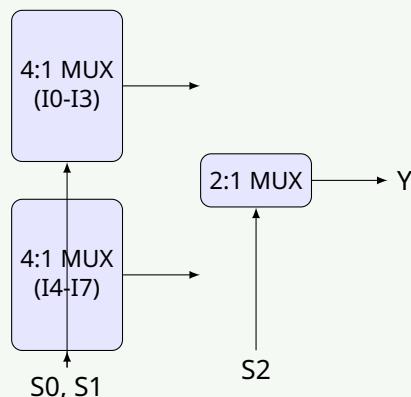
બે 4:1 મલ્ટિપ્લેક્સરનો ઉપયોગ કરીને 8:1 મલ્ટિપ્લેક્સર ડિઝાઇન કરો.

જવાબ

ડિઝાઇન અભિગમ: 8:1 MUX બનાવવા માટે બે 4:1 MUX અને એક 2:1 MUX વાપરો.

1. પ્રથમ 4:1 MUX ઇનપુટ $I_0 - I_3$ સંભાળે છે, સિલેક્ટ લાઇન S_0, S_1 નો ઉપયોગ કરીને
2. બીજો 4:1 MUX ઇનપુટ $I_4 - I_7$ સંભાળે છે, સિલેક્ટ લાઇન S_0, S_1 નો ઉપયોગ કરીને
3. 2:1 MUX બે 4:1 MUXના આઉટપુટ વર્ષે S_2 નો ઉપયોગ કરીને પસંદગી કરે છે

બ્લોક ડાયાગ્રામ:



ટુથ ટેબલ:

S_2	S_1	S_0	આઉટપુટ Y
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7

મેમરી ટ્રીક

" S_0, S_1 દરેક 4:1 MUXમાંથી પસંદ કરે છે, S_2 તેમની વર્ષે પસંદ કરે છે"

પ્રશ્ન 4(અ) [3 ગુણ]

સિક્વન્સીયલ સર્કિટની વ્યાખ્યા લખો. તેના બે ઉદાહરણ લખો

જવાબ

સિક્વન્સીયલ સર્કિટ: એક ડિજિટલ સર્કિટ જેનું આઉટપુટ માત્ર વર્તમાન ઇનપુટ પર જ નહીં પણ ઇનપુટના ભૂતકાળના ક્રમ (ઇતિહાસ/અગાઉની સ્થિતિ) પર પણ આધારિત હોય છે.

મુખ્ય લક્ષણો:

- મેમરી એલિમેન્ટ્સ (ફિલ્પ-ફલોપ) ધરાવે છે
- આઉટપુટ વર્તમાન ઇનપુટ અને અગાઉની સ્થિતિઓ બંને પર આધારિત છે
- સામાન્ય રીતે ફીડબેક પાથ સમાવે છે
- સિંકોનાઇઝેશન માટે કલોક સિશ્વલની જરૂર પડે છે (સિંકોન્સ સર્કિટ માટે)

સિક્વન્સીયલ સર્કિટના ઉદાહરણો:

1. ફિલ્પ-ફલોપ (SR, JK, D, T)
2. રજિસ્ટર (શિફ્ટ રજિસ્ટર)
3. કાઉન્ટર (બાયનરી, ડિક્રેડ, રિંગ કાઉન્ટર)
4. સ્ટેટ મશીન
5. મેમરી ચુનિટ

કોષ્ટક 12. સિક્વન્સીયલ vs કોમ્બીનેશનલ સર્કિટ

લક્ષણ	સિક્વન્સીયલ સર્કિટ	કોમ્બીનેશનલ સર્કિટ
મેમરી	હા	ના
ફીડબેક	સામાન્ય રીતે	ના
આઉટપુટ આધારિત	વર્તમાન અને અગાઉના ઇનપુટ	માત્ર વર્તમાન ઇનપુટ
કલોક જરૂરી	સામાન્ય રીતે	ના
ઉદાહરણો	ફિલ્પ-ફલોપ, કાઉન્ટર	માન્યલ કલોકસર, એડર

મેમરી ટ્રીક

"સિક્વન્સીયલ ઇતિહાસ યાદ રાપે છે, કોમ્બીનેશનલ માત્ર વર્તમાન જાણે છે"

પ્રશ્ન 4(બ) [4 ગુણ]

ડિક્રેડ કાઉન્ટર ડિઝાઇન કરો

જવાબ

ડિક્રેડ કાઉન્ટર: એક સિક્વન્સીયલ સર્કિટ જે 0 થી 9 (ડેસિમલ) સુધી ગણે છે અને પછી 0 પર રીસેટ થાય છે.

JK ફિલ્પ-ફલોપનો ઉપયોગ કરી ડિઝાઇન:

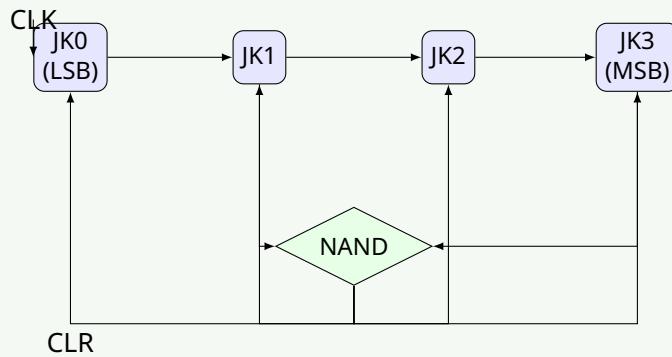
- 4 બિટ બાયનરી નંબર રજૂ કરવા માટે 4 JK ફિલ્પ-ફલોપ (Q_3, Q_2, Q_1, Q_0) જરૂરી છે
- 0000 થી 1001 (0-9 ડિસિમલ) સુધી ગણે છે પછી રીસેટ થાય છે

J-K ઇનપુટ સમીકરણ:

- $J_0 = K_0 = 1$ (દરેક કલોક પર ટોગલ)
- $J_1 = K_1 = Q_0 \cdot \overline{Q_3}$
- $J_2 = K_2 = Q_1 \cdot Q_0$
- $J_3 = K_3 = Q_2 \cdot Q_1 \cdot Q_0 + Q_3 \cdot Q_0$

રીસેટ સ્થિતિ: જ્યારે $Q_3 \cdot Q_1 = 1$ (સ્થિતિ 1010), બધા ફિલ્પ-ફલોપ રીસેટ કરો

બ્લોક ડાયાગ્રામ (Asynchronous/Ripple):



મેમરી ટ્રીક

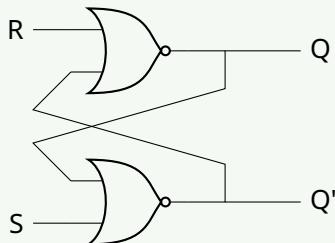
"BCD ગણો, 9 પછી રીસેટ"

પ્રશ્ન 4(ક)(૧) [૩ ગુણ]

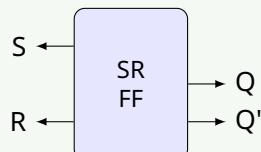
NOR ગેટની મદદથી S-R ફિલિપ-ફલોપ સમજાવો. તેનો લોજિક સિમ્બોલ દોરો અને ટુથ ટેબલ લખો.

જવાબ

NOR ગેટથી S-R ફિલિપ-ફલોપ: બે કોસ-કપણ નોર ગેટમાંથી બનેલું એક મૂળભૂત ફિલિપ-ફલોપ જે એક બિટની માહિતી સંગ્રહિત કરી શકે છે. લોજિક સક્રિયા:



લોજિક સિમ્બોલ:



ટુથ ટેબલ:

S	R	Q (આગામી)	Q' (આગામી)	ઓપરેશન
0	0	Q (અગાઉની)	Q' (અગાઉની)	મેમરી (કોઈ ફેરફાર નહીં)
0	1	0	1	રીસેટ
1	0	1	0	સેટ
1	1	0	0	અમાન્ય (ટાળો)

મેમરી ટ્રીક

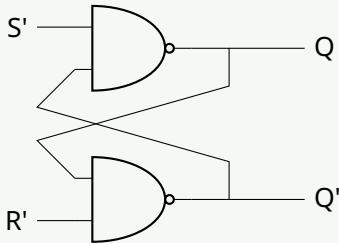
"S થી 1 સેટ થાય, R થી 0 રીસેટ થાય, બંને એકસાથે અમાન્ય સ્થિતિ આપે"

પ્રશ્ન 4(ક)(II) [4 ગુણ]

NAND ગેટની મદદથી S-R ફિલ્પ-ફ્લોપ સમજાવો. S-R ફિલ્પ-ફ્લોપની મર્યાદા લખો

જવાબ

NAND ગેટથી S-R ફિલ્પ-ફ્લોપ: બે કોસ-કપડા NAND ગેટમાંથી બનેલું એક મૂળભૂત ફિલ્પ-ફ્લોપ.



SR ફિલ્પ-ફ્લોપની મર્યાદાઓ:

- અમાન્ય સ્થિતિ: જ્યારે $S=1, R=1$ (NOR માટે) અથવા $S=0, R=0$ (NAND માટે), આઉટપુટ અનિશ્ચિત રહે છે
- રેસ કન્ડિશન: જ્યારે ઇનપુટ એકસાથે બદલાય છે, ત્યારે અંતિમ સ્થિતિ અનિશ્ચિત હોઈ શકે છે
- કલોકિગ મેકેનિઝમ નથી: અન્ય ડિજિટલ ઘટકો સાથે સિંકોનાઇઝ થઈ શકતું નથી
- એજ-ટ્રિગાઈડ નથી: ટુંકા પદ્ધતને વિશ્વસનીય રીતે પ્રતિક્રિયા આપી શકતું નથી
- અનિષ્ટનીય ટોગલિંગ: નોઇજ કે ટિલ્યાને પ્રતિક્રિયા આપી શકે છે

કોષ્ટક 13. NAND vs NOR SR ફિલ્પ-ફ્લોપ

લક્ષણ	NAND SR ફિલ્પ-ફ્લોપ	NOR SR ફિલ્પ-ફ્લોપ
સહ્ય ઇનપુટ	લો (0)	હાઈ (1)
નિષ્ટિક્ય ઇનપુટ	હાઈ (1)	લો (0)
અમાન્ય સ્થિતિ	$S=0, R=0$	$S=1, R=1$

મેમરી ટ્રીક

"NAND: ઇનપુટ એકિટિવ-લો, NOR: ઇનપુટ એકિટિવ-હાઇ; બંનેમાં એક અમાન્ય સ્થિતિ છે"

પ્રશ્ન 4 [OR] (અ) [3 ગુણ]

ફિલ્પ-ફ્લોપની વ્યાખ્યા લખો. ફિલ્પ-ફ્લોપના પ્રકાર લખો

જવાબ

ફિલ્પ-ફ્લોપ: એક મૂળભૂત સિકવન્સીયલ ડિજિટલ સર્કિટ જે એક બિટની માહિતી સંગ્રહિત કરી શકે છે અને બે સ્થાયી સ્થિતિઓ (0 અથવા 1) ધરાવે છે. તે ડિજિટલ સિસ્ટમમાં મૂળભૂત મેમરી એલિમેન્ટ તરીકે કામ કરે છે.

મુખ્ય લક્ષણો:

- બાયસ્ટેબલ મલ્ટિવાયબ્રેટર (બે સ્થાયી સ્થિતિઓ)
- જ્યાં સુધી બદલવાનો નિર્દેશ ન અપાય ત્યાં સુધી પોતાની સ્થિતિ અનિશ્ચિત સમય સુધી જાળવી રાખી શકે છે
- રજિસ્ટર, કાઉન્ટર અને મેમરી સર્કિટ માટે મૂળભૂત બિલ્ડિંગ બ્લોક બને છે

ફિલ્પ-ફ્લોપના પ્રકાર:

ફિલેપ-ફલોપ પ્રકાર	વર્ણન
SR (સેટ-રીસેટ)	સૌથી મૂળભૂત ફિલેપ-ફલોપ જેમાં સેટ અને રીસેટ ઇનપુટ હોય છે
JK	SR ફિલેપ-ફલોપની સુધારેલી આવૃત્તિ જે અમાન્ય સ્થિતિ દૂર કરે છે
D (ડેટા)	ઇનપુટ D પરનો મૂલ્ય સંગ્રહિત કરે છે, ડેટા સ્ટોરેજ માટે વપરાય છે
T (ટોગલ)	ટ્રિગર થથે સ્થિતિ બદલે છે, કાઉન્ટર માટે ઉપયોગી
માસ્ટર-સ્લેવ	રેસ કન્ડિશન અટકાવતું બે-તબક્કાનું ફિલેપ-ફલોપ

મેમરી ટ્રીક

"એક સિંગલ સ્ટેટ સ્ટોરેજ: SR, JK, D, T"

પ્રશ્ન 4 [OR] (બ) [4 ગુણ]

3-bit રિંગ કાઉન્ટર ડિઝાઇન કરો

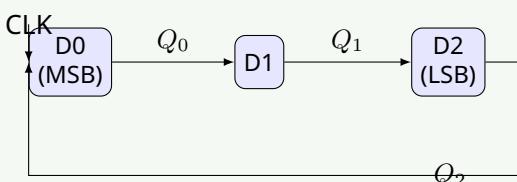
જવાબ

રિંગ કાઉન્ટર: એક સર્ક્યુલર શિફ્ટ રજિસ્ટર જેમાં ફક્ત એક બિટ સેટ (1) હોય છે અને બાકી બધા રીસેટ (0) હોય છે. એકમાત્ર સેટ બિટ કલોક થતાં રજિસ્ટરમાં "ફરે" છે.

D ફિલેપ-ફલોપનો ઉપયોગ કરી ડિઝાઇન:

- 3-bit કાઉન્ટર માટે 3 D ફિલેપ-ફલોપ જરૂરી છે
- પ્રારંભિક સ્થિતિ: 100, પછી 010, 001, અને પાછા 100 પર જાય છે

બ્લોક ડાયાગ્રામ:



મેમરી ટ્રીક

"એક હોટ બિટ વર્તુળમાં ફરે છે"

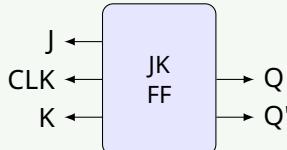
પ્રશ્ન 4 [OR] (ક)(િ) [3 ગુણ]

લોજિક સિમ્બોલ અને ટૂથ ટેબલની મદદથી J-K ફિલેપ-ફલોપ સમજાવો

જવાબ

J-K ફિલેપ-ફલોપ: SR ફિલેપ-ફલોપની સુધારેલી આવૃત્તિ જે અમાન્ય સ્થિતિ દૂર કરે છે અને બધા ઇનપુટ સંયોજનોમાં સચોટ વર્તન દર્શાવે છે.

લોજિક સિમ્બોલ:



ટૂથ ટેબલ:

J	K	Q (આગામી)	ઓપરેશન
0	0	Q (અગાઉની)	કોઈ ફેરફાર નહીં
0	1	0	રીસેટ
1	0	1	સેટ
1	1	Q' (અગાઉની)	ટોગલ

મેમરી ટ્રીક

"J સેટ કરે, K રીસેટ કરે, બંને ટોગલ કરે, કોઈ નહીં યાદ રાખે"

પ્રશ્ન 4 [OR] (ક)(II) [4 ગુણ]

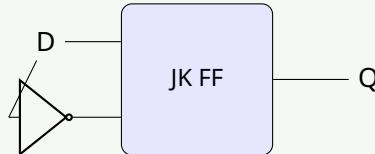
J-K ફિલ્પ-ફલોપનો ઉપયોગ કરી D ફિલ્પ-ફલોપ અને T ફિલ્પ-ફલોપની લોજિક સક્રિયાઓ

જવાબ

JK ફિલ્પ-ફલોપનો ઉપયોગ કરી D ફિલ્પ-ફલોપ:

- D ઇનપુટને J સાથે જોડો
- D' (NOT D)ને K સાથે જોડો

લોજિક સક્રિયા:



JK ફિલ્પ-ફલોપનો ઉપયોગ કરી T ફિલ્પ-ફલોપ:

- T ઇનપુટને J અને K બંને સાથે જોડો

લોજિક સક્રિયા:



મેમરી ટ્રીક

"D સીધું અનુસરે, T સાચું હોય ત્યારે ટોગલ થાય"

પ્રશ્ન 5(અ) [3 ગુણ]

RAM અને ROM-ની સરખામણી કરો

જવાબ

RAM (Random Access Memory) vs ROM (Read-Only Memory):

કોષ્ટક 14. RAM vs ROM સરખામણી

લક્ષણ	RAM	ROM
પૂર્જ નામ	Random Access Memory	Read-Only Memory
ડેટા નિભાવણી	અસ્થાયી (પાવર બંધ થતાં ડેટા ગુમાવે)	સ્થાયી (પાવર વિના પણ ડેટા જળવાય)
વાંચન/લેખન ક્ષમતા	વાંચન અને લેખન બંને	મુખ્યત્વે માત્ર વાંચન
ગતિ	વધુ ઝડપી	ધીમી
બિટ દીઠ ખર્ચ	વધુ	ઓછો
ઉપયોગો	અસ્થાયી ડેટા સ્ટોરેજ	બૂટ સૂચનાઓ, ફર્મવેર

મેમરી ટ્રીક

"RAM વાંચે અને સુધારે (પણ ભૂલી જાય), ROM શટડાઉન પર યાદ રાખે (પણ નિશ્ચિત)"

પ્રશ્ન 5(બ) [4 ગુણ]

સિરિયલ ઇન સિરિયલ આઉટ શિફ્ટ રજિસ્ટર સમજાવો

જવાબ

સિરિયલ ઇન સિરિયલ આઉટ (SISO) શિફ્ટ રજિસ્ટર: એક સિકવન્સીયલ સર્કિટ જે ઇનપુટ અને આઉટપુટ બંને પર ડેટાને એક સમયે એક બિટ શિફ્ટ કરે છે.

કાર્યપદ્ધતિ:

- ડેટા સિરિયલી એક બિટ એક વખતે દાખલ થાય છે
- દરેક કલોક પદ્સ પર દરેક બિટ રજિસ્ટરમાંથી શિફ્ટ થાય છે
- ડેટા સિરિયલી એક બિટ એક વખતે બહાર નીકળે છે

બ્લોક ડાયાગ્રામ:



મેમરી ટ્રીક

"બિટ્સ લાઇનમાં પ્રવેશો, શ્રેણીમાં આગળ વધે, કમમાં બહાર નીકળે"

પ્રશ્ન 5(ક) [7 ગુણ]

લોજિક ફેમિલિઝ પર ટૂંક નોંધ લખો

જવાબ

લોજિક ફેમિલિઝ: સમાન ઇલેક્ટ્રિકલ લક્ષણો, ફેબ્રિકેશન ટેકનોલોજી અને લોજિક અમલીકરણ સાથેના ડિજિટલ ઇન્ટિગ્રેટેડ સર્કિટના સમૂહો.

મુખ્ય લોજિક ફેમિલિઝ:

1. TTL (ટ્રાન્ઝિસ્ટર-ટ્રાન્ઝિસ્ટર લોજિક):

- બાયપોલર જંકશન ટ્રાન્ઝિસ્ટર પર આધારિત
- Supply voltage: 5V

2. CMOS (કોમ્પ્લિમેન્ટરી મેટલ-ઓક્સાઇડ-સેમિકન્ડક્ટર):

- MOSFETs (P-ટાઇપ અને N-ટાઇપ) પર આધારિત
- ખૂલ્ય ઓછો પાવર વપરાશ

3. ECL (ઇમિટર-કપલ લોજિક):

- અત્યંત તોચી ઝડપ (સૌથી ઝડપી લોજિક ફેમિલી)

કોષ્ટક 15. સરખામણી કોષ્ટક

પેરામીટર	TTL	CMOS	ECL
જડપ	મદ્યમ	ઓછી થી ઊંચી	ખૂબ ઊંચી
પાવર વપરાશ	મદ્યમ	ખૂબ ઓછો	ઊંચો
નોઇજ ઇમ્પુનિટી	ઊંચી	ખૂબ ઊંચી	ઓછી
ફેન-આઉટ	10	50+	25

મેમરી ટ્રીક

"TTL ટ્રાન્ઝિસ્ટર ટેકનોલોજી, CMOS કરંટ ઓછો વાપરે છે, ECL એક્સ્ટ્રીમ જડપે કામ કરે છે"

પ્રશ્ન 5 [OR] (અ) [3 ગુણ]

SRAM અને DRAMની સરખામણી કરો

જવાબ

SRAM (સ્ટેટિક RAM) vs DRAM (ડાયનમિક RAM):

કોષ્ટક 16. SRAM vs DRAM સરખામણી

લક્ષણ	SRAM	DRAM
પૂર્ણ નામ	Static Random Access Memory	Dynamic Random Access Memory
સ્ટોરેજ એલિમેન્ટ	ફિલાપ-ફલોપ	કેપેસિટર
રિફેશિંગ	જરૂરી નથી	સમયાંતરે જરૂરી (ms)
જડપ	વધુ જડપી	ધીમી
ડેન્સિટી	ઓછી (મોટો સેલ સાઈઝ)	ઊંચી (નાનો સેલ સાઈઝ)
બિટ દીઠ ખર્ચ	વધુ	ઓછો
ઉપયોગો	કેશ મેમરી	મુખ્ય મેમરી (RAM)

મેમરી ટ્રીક

"સ્ટેટિક સ્થિર રહે છે છ ટ્રાન્ઝિસ્ટર સાથે, ડાયનમિક ફ્રેઇન થાય અને નિયમિત રિફેશ જોઈએ"

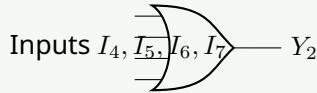
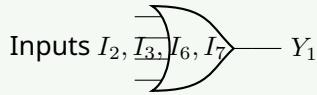
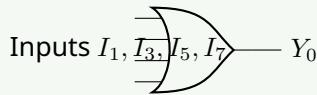
પ્રશ્ન 5 [OR] (બ) [4 ગુણ]

8:3 એનકોડર સમજાવો

જવાબ

8:3 એનકોડર: એક કોમ્બિનેશનલ સર્કિટ જે 8 ઇનપુટ લાઇન્સને 3 આઉટપુટ લાઇન્સમાં રૂપાંતરિત કરે છે, મૂળભૂત રીતે સર્કિય ઇનપુટ લાઇન્સને તેની બાયનરી પોઝિશનમાં રૂપાંતરિત કરે છે.

લોજિક સર્કિટ:



બુલિયન સમીકરણાઃ

- $Y_0 = I_1 + I_3 + I_5 + I_7$
- $Y_1 = I_2 + I_3 + I_6 + I_7$
- $Y_2 = I_4 + I_5 + I_6 + I_7$

મેમરી ટ્રીક

"આઠ ઇનપુટ તેમના સ્થાન ત્રણ બિટમાં બને"

પ્રશ્ન 5 [OR] (૫) [૭ ગુણ]

લોજિક ફેબ્રિલિઝ માટે નીચેની વ્યાખ્યાઓ લખો (i) ફેન-ઇન (ii) ફેન-આઉટ (iii) નોઇસ માર્જિન (iv) પ્રોપેશન ડિલે (v) પાવર ડિસિપેશન

જવાબ

લોજિક ફેબ્રિલિઝના મુખ્ય પેરામીટ્ર્સ:

- ફેન-ઇન:** લોજિક ગેટ સ્વીકારી શકે તેવા ઇનપુટની મહત્વમાં સંખ્યા
- ફેન-આઉટ:** એક ગેટ આઉટપુટ દ્વારા વિશ્વસનીય રીતે દ્રાઇવ થઈ શકતા સમાન ગેટની મહત્વમાં સંખ્યા
- નોઇઝ માર્જિન:** અનિયાનીય ઇલેક્ટ્રિકલ નોઇઝ/સિશ્વલ સહન કરવાની ક્ષમતા
- પ્રોપેશન ડિલે:** ઇનપુટ ચેન્જ અને તેના તરત પછીના આઉટપુટ ચેન્જ વરચેનો સમય વિલંબ
- પાવર ડિસિપેશન:** ગેટ દ્વારા વપરાતી પાવરની માત્રા

મેમરી ટ્રીક

"પાંચ ફેક્ટર: ફેન-ઇન ઇનપુટ ગણો, ફેન-આઉટ ગેટ ચલાવે, નોઇઝ માર્જિન દખલ સામે લડે, પ્રોપેશન ડિલે ઝડપ માપે, પાવર ડિસિપેશન ગરમી ઉત્પત્ત કરે"