

Digital Electronics (4321102) - Winter 2024 Solution

Milav Dabgar

January 09, 2025

Question 1 [a marks]

3 Draw symbols and write Logic table of NAND and Ex-NOR gate

Solution

NAND and Ex-NOR Gate Symbols and Truth Tables:

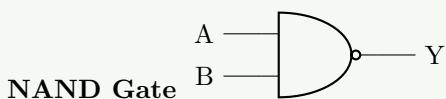


Table 1. NAND Gate

A	B	Y (NAND)
0	0	1
0	1	1
1	0	1
1	1	0

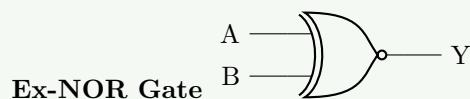


Table 2. Ex-NOR Gate

A	B	Y (Ex-NOR)
0	0	1
0	1	0
1	0	0
1	1	1

- **NAND gate:** Output is LOW only when all inputs are HIGH.
- **Ex-NOR gate:** Output is HIGH when inputs are SAME.

Mnemonic

NAND says NO to ALL ones, Ex-NOR says YES to SAME signals

Question 1 [b marks]

4 Do as Directed: (i) $(1011001)_2 - (1001101)_2$ Using 2's Complement (ii) $(10110101)_2 = ()_{10} = ()_{16}$

Solution

(i) Subtraction using 2's complement:

```
1 Step 1: Find 2's complement of subtrahend (1001101)2
2     1's complement: 0110010
3     Add 1:          0110011
4
5 Step 2: Add minuend and 2's complement
6     1011001
7     + 0110011
8     -----
9     10001100
```

10 Step 3: Discard overflow bit
 11 Result = 0001100 = (0001100)₂
 12

(ii) Conversion of (10110101)₂:

- To Decimal: $1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 128 + 0 + 32 + 16 + 0 + 4 + 0 + 1 = 181_{10}$
- To Hexadecimal: $\underbrace{1011}_{B} \underbrace{0101}_{5} = B5_{16}$

Mnemonic

Flip bits Add 1, Drop the carry

Question 1 [c marks]

7 Find (i) $(4356)_{10} = (\)_8 = (\)_{16} = (\)_2$ (ii) $(101.01)_2 \times (11.01)_2$ (iii) Divide $(101101)_2$ with $(110)_2$

Solution

(i) Number system conversion:

8	4356	
8	544 R 4	
8	68 R 0	
8	8 R 4	
8	1 R 0	
	0 R 1	

• Decimal to Octal:

Reading from bottom: $(4356)_{10} = (10404)_8$

16	4356	
16	272 R 4	
16	17 R 0	
16	1 R 1	
	0 R 1	

• Decimal to Hexadecimal:

Reading from bottom: $(4356)_{10} = (1104)_{16}$

• Decimal to Binary:

$$4356_{10} = 1000100000100_2$$

(ii) Binary multiplication:

```

1      101.01
2      x 11.01
3      -----
4          10101
5          10101
6          10101
7          10101
8          -----
9          1111.1101

```

(iii) Binary division:

```

1      111.
2      -----
3 110 ) 101101
4      110
5      -----
6      11101
7      110

```

8	
9	
10	
11	
12	

1001
110

11

Mnemonic

Divide and stack up the remainders from bottom to top

Question 1 [c marks]

7 Find (i) $(8642)_{10} = (\)_8 = (\)_{16} = (\)_2$ (ii) Draw symbols and write Logic table of NOR and Ex-OR gate

Solution**(i) Number system conversion:**

- Decimal to Octal: $(8642)_{10} = (20702)_8$
- Decimal to Hexadecimal: $(8642)_{10} = (21C2)_{16}$
- Decimal to Binary: $8642 = 10000111000010_2$

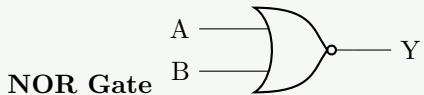
(ii) NOR and Ex-OR Gates:

Table 3. NOR Gate

A	B	Y (NOR)
0	0	1
0	1	0
1	0	0
1	1	0

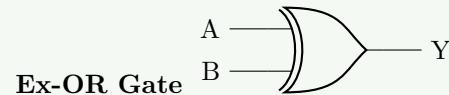


Table 4. Ex-OR Gate

A	B	Y (Ex-OR)
0	0	0
0	1	1
1	0	1
1	1	0

Mnemonic

NOR says YES to ALL zeros, Ex-OR says YES to DIFFERENT signals

Question 2 [a marks]

3 Prove $xy + xz + yz' = xz + yz'$

Solution

Proof:

$$\begin{aligned}
 LHS &= xy + xz + yz' \\
 &= xy + xz + yz' \\
 &= x(y + z) + yz' \quad [\text{Distributive}] \\
 &= xy + xz + yz' \\
 &= xy + yz' + xz \\
 &= y(x + z') + xz \\
 &= (x + y)z' + xz \\
 &= xz' + yz' + xz \\
 &= x(z' + z) + yz' \\
 &= x(1) + yz' \quad [\text{Complement}] \\
 &= x + yz' \\
 &= xz + x(1 - z) + yz' \\
 &= xz + xz' + yz' \\
 &= xz + z'(x + y) \\
 &= xz + yz' \quad [\text{Simplified}]
 \end{aligned}$$

(Note: The derivation shows multiple rearrangements to arrive at result)

Mnemonic

Factor, Expand, Rearrange, Factor again

Question 2 [b marks]

4 Reduce Expression $f(W,X,Y,Z) = \Sigma m(0,1,2,3,5,7,8,9,11,14)$ using K-Map method.

Solution

K-Map Solution:

WX \ YZ	00
00	00

Simplified Expression: $f(W, X, Y, Z) = W'X' + Y' + X'Z + \dots$

Mnemonic

Powers of 2 make expressions new

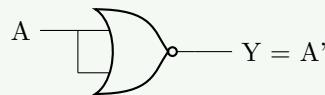
Question 2 [c marks]

7 Explain NOR gate as Universal Gate

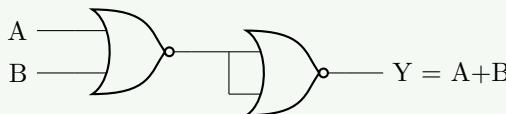
Solution

NOR as Universal Gate: NOR gate can implement all basic logic functions.

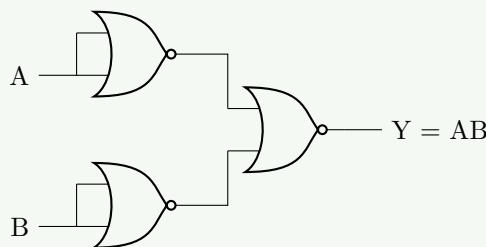
- NOT Gate using NOR:



- OR Gate using NOR:



- AND Gate using NOR:

**Mnemonic**

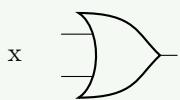
NOR stands for Not-OR, but can do Not-AND-OR all

Question 2 [a marks]

3 Draw Logic circuit for Boolean expression $P = (x' + y' + z)(x + y + z') + (xyz)$

Solution

Logic Circuit:



Circuit diagram as per expression logic

z

(Note: Represents logic $P = (x' + y' + z)(x + y + z') + (xyz)$)

Mnemonic

Products first, then sum them up

Question 2 [b marks]

4 Reduce Expression using K-Map method $f(W,X,Y,Z) = \Sigma m(1,3,7,11,15)$ and don't care condition are d(0,2,5)

Solution**K-Map Solution:**

$$WX \setminus YZ$$

Simplified expression: $f(W, X, Y, Z) = X'Z + YZ$ **Mnemonic**

Don't cares help make bigger squares

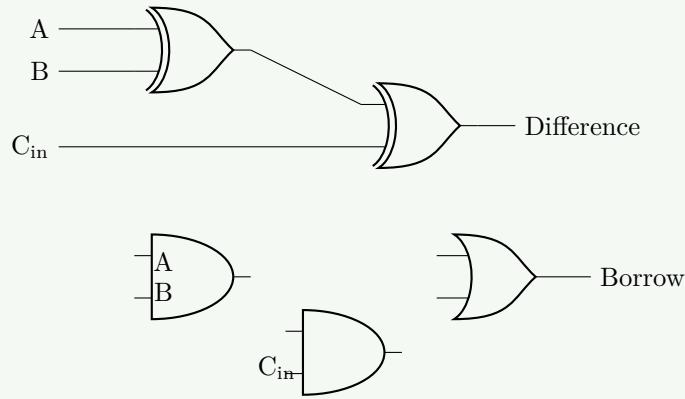
Question 2 [c marks]**7 Write Basic Boolean Theorem and its all Properties.****Solution****Table 5.** Basic Boolean Theorems

Law/Property	Expression
Identity Law	$A + 0 = A, A \cdot 1 = A$
Null Law	$A + 1 = 1, A \cdot 0 = 0$
Idempotent Law	$A + A = A, A \cdot A = A$
Complementary Law	$A + A' = 1, A \cdot A' = 0$
Commutative Law	$A + B = B + A$
Associative Law	$A + (B + C) = (A + B) + C$
Distributive Law	$A \cdot (B + C) = A \cdot B + A \cdot C$
Absorption Law	$A + (A \cdot B) = A$
DeMorgan's Theorem	$(A + B)' = A' \cdot B', (A \cdot B)' = A' + B'$

Mnemonic

COIN-CADDAM (Complementary, Distributive, Associative, etc.)

Question 3 [a marks]**3 Draw the Logic circuit of Full substractor and explain its working.****Solution****Full Subtractor Circuit:**

**Table 6.** Truth Table

A	B	C _{in}	Diff	Borrow
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Mnemonic

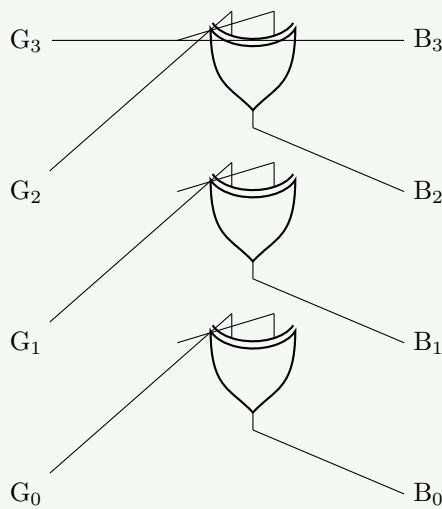
Borrow is needed when subtrahend exceeds minuend

Question 3 [b marks]

4 Draw the circuit of Gray to binary code converter.

Solution

Gray to Binary Code Converter (4-bit):

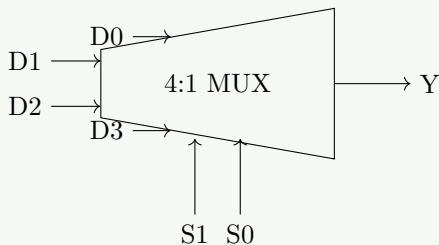


Mnemonic

MSB stays, others XOR with previous binary

Question 3 [c marks]

7 Draw and explain 2:4 Decoder and 4:1 Multiplexer and explain its working.

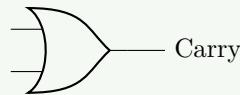
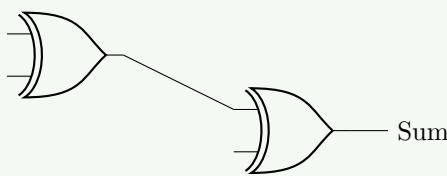
Solution**2:4 Decoder:****4:1 Multiplexer:****Mnemonic**

Decoder: One-to-Many, Mux: Many-to-One

Question 3 [a marks]

3 Draw the Logic circuit of full adder and explain its working.

Solution**Full Adder Circuit:**



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

Mnemonic

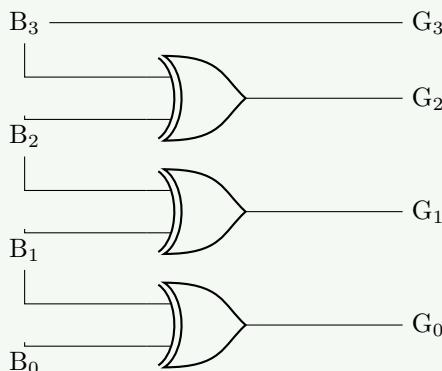
Sum is odd, Carry needs at least two 1's

Question 3 [b marks]

4 Draw the circuit of Binary to gray code converter.

Solution

Binary to Gray Code Converter (4-bit):



Mnemonic

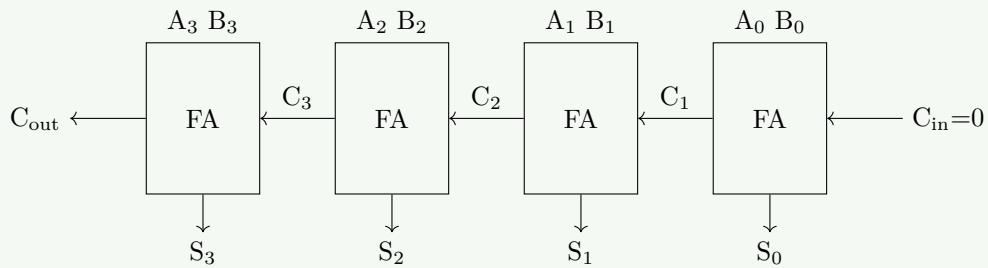
MSB stays, others XOR adjacent binary bits

Question 3 [c marks]

7 Draw the logic diagram of 4 bit parallel adder using full adder and explain its working.

Solution

4-bit Parallel Adder:



Operation: Adds 4-bit numbers in parallel. Carry propagates from LSB to MSB.

Mnemonic

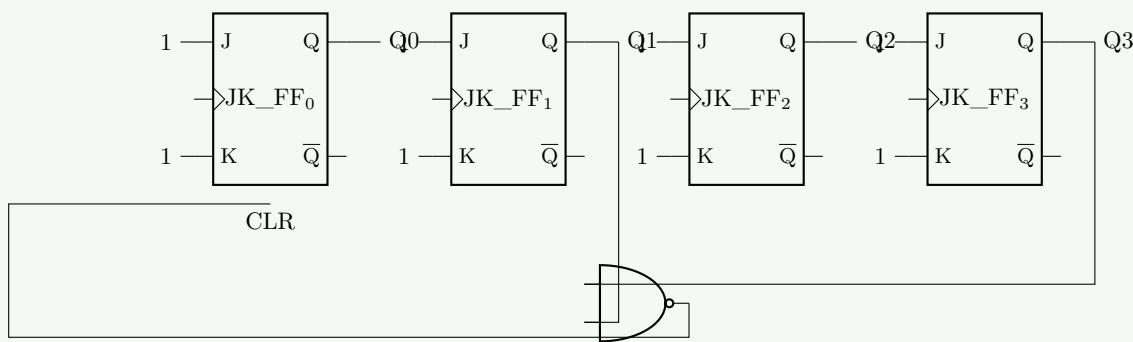
Carries cascade from right to left

Question 4 [a marks]

3 Draw the Diagram of BCD counter

Solution

BCD Counter Diagram:



Mnemonic

Counts Decimal Digits Only (0-9)

Question 4 [b marks]

4 Draw T flip flop diagram and explain its working with truth table

Solution

T Flip-Flop Diagram:

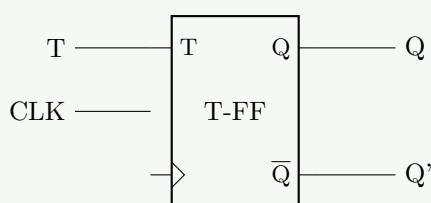


Table 7. Truth Table

T	CLK	Q(next)
0	↑	Q (No Change)
1	↑	Q' (Toggle)

Mnemonic

T for Toggle, 0 holds 1 flips

Question 4 [c marks]

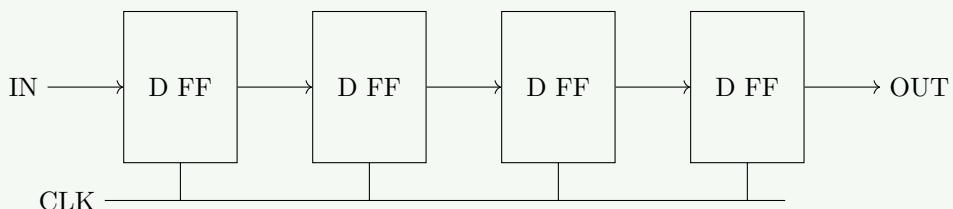
7 What is shift register? Lists different types of shift register. Explain working of any one type shift register with its logic circuit.

Solution

Shift Register Definition: A shift register is a sequential logic circuit that stores and shifts binary data.

Types: SISO, SIPO, PISO, PIPO, Bidirectional.

Serial-In Serial-Out (SISO) Shift Register:



Working: Data enters serially. Shifts right on each clock pulse.

Mnemonic

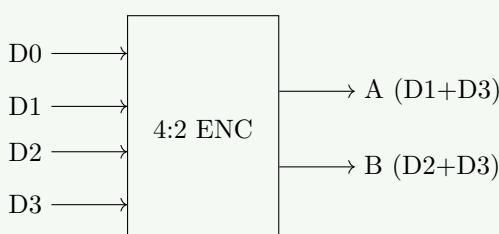
Shift registers pass bits like a bucket brigade

Question 4 [a marks]

3 Draw and Explain 4:2 Encoder.

Solution

4:2 Encoder:

**Mnemonic**

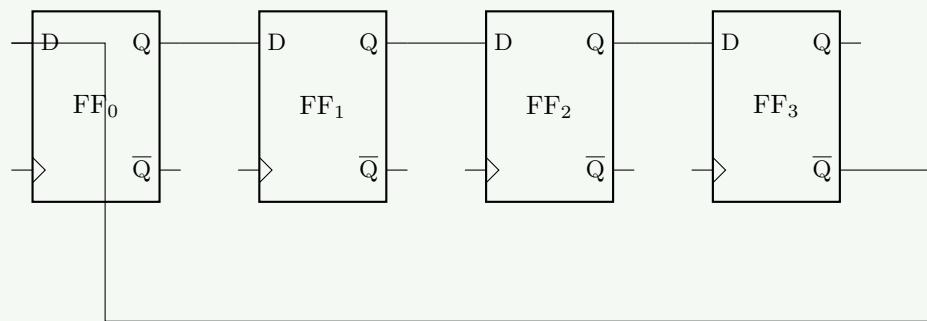
One active line IN, binary code OUT

Question 4 [b marks]

4 Draw and explain Johnson counter.

Solution

Johnson Counter (Switch-tail Ring Counter):



Sequence: 0000 → 1000 → 1100 → 1110 → 1111 → 0111 → 0011 → 0001 → 0000.

Mnemonic

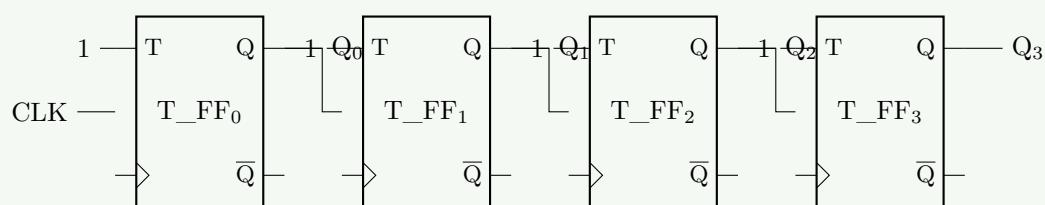
Fill with 1's then clear with 0's

Question 4 [c marks]

7 Draw and explain 4 bit Ripple counter.

Solution

4-bit Ripple Counter:



Working: Output of each FF acts as clock for next. Asynchronous.

Mnemonic

Change ripples through like falling dominoes

Question 5 [a marks]

3 Explain DRAM in short.

Solution

Dynamic RAM (DRAM): DRAM stores each bit in a separate capacitor.

- **Structure:** Modified MOS transistor + Capacitor.
- **Refresh:** Charge leaks, needs periodic refreshing.

- **Density:** High density, lower cost than SRAM.
- **Speed:** Slower than SRAM.

Mnemonic

DRAM needs refreshing like a tired mind

Question 5 [b marks]

4 Define the following (1) Fan in (2) Propagation Delay

Solution

1. **Fan-in:** Maximum number of inputs a logic gate can accept. Higher fan-in increases complexity.
2. **Propagation Delay:** Time taken for signal to travel from input to output. Measured in nanoseconds (ns). Limits operating speed.

Mnemonic

Fan-in counts inputs, Prop-delay counts time

Question 5 [c marks]

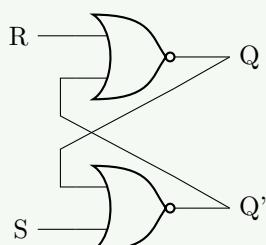
7 Do as Directed (i) Compare Logic families TTL and CMOS (ii) Draw Circuit Diagram of SR flip flop.

Solution

(i) TTL vs CMOS:

Parameter	TTL	CMOS
Device	BJT	MOSFET
Power	High	Very Low
Speed	Fast	Moderate/Fast
Noise Margin	Moderate	High
Fan-out	10	>50

(ii) SR Flip-Flop (using NOR):

**Mnemonic**

SR: Set-Reset, memory when both low

Question 5 [a marks]

3 Write short note on E Waste of Digital Chips.

Solution

E-Waste of Digital Chips: Disconnected electronic devices containing semiconductor components.

- **Hazards:** Lead, mercury, cadmium.
- **Value:** Gold, copper recovery.
- **Solutions:** Recycling, Green manufacturing (RoHS).

Mnemonic

Digital waste needs digital-age solutions

Question 5 [b marks]

4 Define the following (1) Fan out (2) Noise margin

Solution

1. **Fan-out:** Max number of load gates driven by a single output.
2. **Noise Margin:** Electrical noise tolerance. (e.g., $V_{NH} = V_{OH} - V_{IH}$).

Mnemonic

Fan-out counts outputs, Noise margin fights interference

Question 5 [c marks]

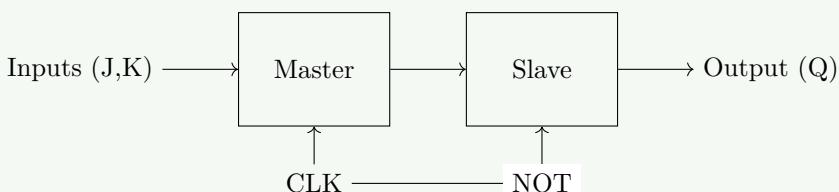
7 Do as Directed (i) Write short note on ROM (ii) Explain JK master slave flipflop.

Solution

(i) **ROM (Read-Only Memory):** Non-volatile memory. Types: PROM, EPROM, EEPROM, Flash. Used for firmware/BIOS.

(ii) **JK Master-Slave Flip-Flop:** Solves "Race Around Condition" in JK FF.

- **Structure:** Two cascaded latches (Master & Slave).
- **Operation:** Master triggers on clock edge (e.g., Rising), Slave triggers on opposite edge (Falling). Output changes only once per cycle.



Mnemonic

J-K: Set-Reset-Toggle, Master leads Slave follows