Q.1(a) Perform the following mathematical operations using 2's complement method. 03m
(i) (9)10 + (-5)10
(ii) (3)10 – (8)10

1. **(9)10 + (-5)10**

- **$(9)_{10}$=(1001)2 (5)10=(0101)2**
- Find 2's complement of 5(to represent -5)
- take 1's complement of 5(0101)=1010
- Add 1 to 1010=1011
- So -5=1011
- Add 9 and -5

```
 1001
 1011
--------
1 0100
```

result = 0100 reject carry 1 so final ans for 9 + (-5)=4

2. **(3)10 - (8)10**

- (3)10= 0011 (8)10= 1000
- Find 2s complement of (to represent -8)
- So 1s complement of 8(1000)= 0111
- Add 1 to 0111= 1000
- So -8= 1000
- Add 3 and -8

```
 0011
 1000
 ------
 1011
```

- 1011 is negative(msb=1)
- 1s complement of 1011=0100
- Add 1 to 0100= 0101 that is 5
- So 1011= -5
- So adding 3 and -8 gives = -5

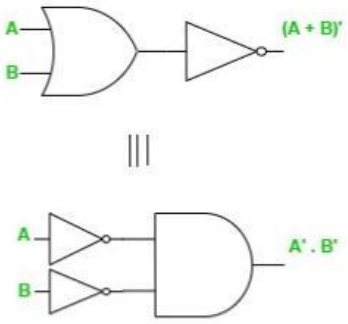
Q1(b) state and prove de morgans theorem using truth table (4m)

1. First De Morgan's law states that "The complement of OR of two or more variables is equal to the AND of the complement of each variable."

- Let A and B be two variables, then mathematically First De Morgan's Law is given as:
- **(A + B)' = A' . B'**
- Where

- + represents the OR operator between variables,
- . represents AND operator between variables, and
- ' represents complement operation on the variable.

A	B	A + B	**(A + B)'**	A'	B'	**A'. 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**



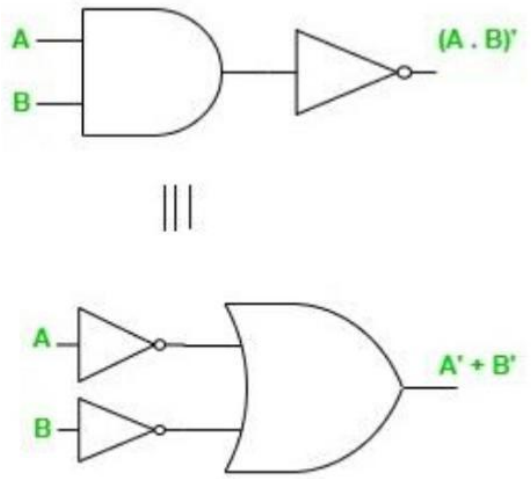
Circuit diagram of de morgan 1st law

2. Second De Morgan's law states that "The complement of AND of two or more variables is equal to the OR of the complement of each variable."

- Let A and B be two variables, then mathematically Second De Morgan's Law is given as:
- **(A . B)' = A' + B'**
- Where + represents the OR operator between variables,
- . represents AND operator between variables, and
- ' represents complement operation on variable.

A	B	A . B	**(A. B)'**	A'	B'	**A' + B'**
0	0	0	**1**	1	1	**1**
0	1	0	**1**	1	0	**1**

A	B	A . B	**(A. B)'**	A'	B'	**A' + B'**
1	0	0	**1**	0	1	**1**
1	1	1	**0**	0	0	**0**



Circuit diagram of de morgan 2nd law

Q1(c) Explain characteristics of digital ic (7m)

Digital ICs (Integrated Circuits) are defined by key characteristics such as speed, power dissipation, fan-in/fan-out, noise margin, and supply voltage compatibility. These parameters determine their performance, efficiency, and reliability in electronic systems.

Key Characteristics of Digital ICs:

- **Speed of Operation (Propagation Delay)**: Measured as the time taken for the output to respond after the input changes.
- Two delays are important:

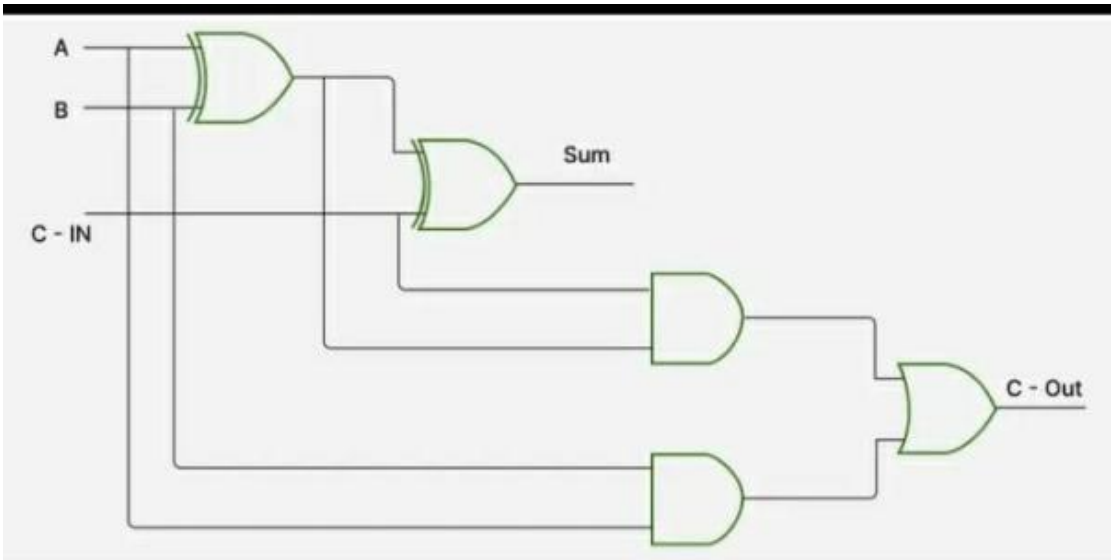
1. **tPLH:** Delay from LOW → HIGH.
2. **tPHL:** Delay from HIGH → LOW.

- **Power Dissipation:** Indicates how much power the IC consumes during operation. Expressed in milliwatts (mW) or nanowatts (nW). Calculated as the product of supply voltage and average current drawn.
- **Fan-In:** The maximum number of inputs a logic gate can handle. Example: A NAND gate with 4 inputs has a fan-in of 4.

- **Fan-Out:** The maximum number of similar gates that a logic gate output can drive without performance degradation. Example: If one gate output drives 10 inputs, its fan-out is 10.
- **Noise Margin:** Defines tolerance against unwanted electrical noise. Higher noise margin = more reliable operation.
- **Supply Voltage Compatibility:** Digital ICs are designed to operate at specific voltage levels (e.g., 5V for TTL, 3.3V for CMOS). Compatibility ensures proper logic level recognition.
- **Loading Capability:** Ability of an IC to drive other circuits without signal distortion.
- **Temperature Dependence:** Performance can vary with temperature; ICs are rated for specific ranges.

Q2 (A) Design a full adder circuit using basic logic gates (3m)

1. **Inputs**
   - A Full Adder has **3 inputs**:
     - AA (first bit)
     - BB (second bit)
     - C in (carry from previous stage)
2. **Outputs**
   - It produces **2 outputs**:
     - **Sum (S)**
     - **Carry (Cout)**
3. **Sum Logic**
   - The sum is obtained by XOR in all inputs:



$S = A \oplus B \oplus C_{in}$

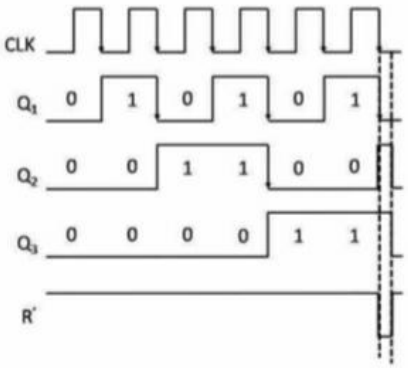
5. **Truth Table (Quick View)**

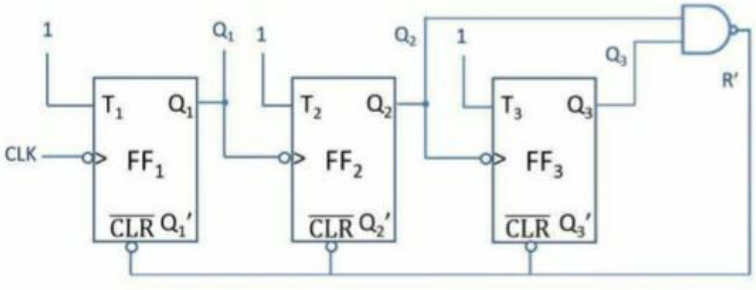
INPUT			OUTPUT	
A	**B**	**C-IN**	**Sum**	**C-OUT**
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1

Q2(b) Design modlo 6 ripple counter using T flip Flops (4m)

- Amod-6 counter has six stable states 000, 001, 010, 011, 100, 101.
- It is also known as “Divide by 6” counter and it requires 3 Flip-flops for designing.
- At the 6" clock pulse, it will again reset to 000 via feedback circuit.
- Reset signal R = 1 at time of 110, R =0 for 000 to 101 and R = X for invalid states i.e. 111.

After Pulses	State			R
	Q_3	Q_2	Q_1	
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
	↓	↓	↓	
	0	0	0	





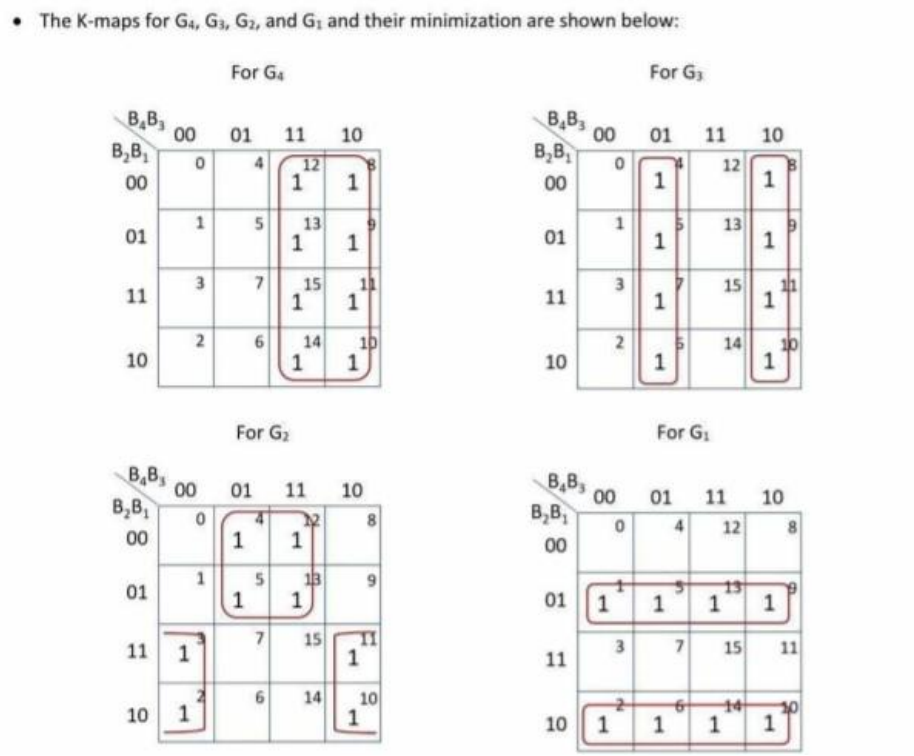
Q2(c) Design a 4-bit Binary to Gray Code Converter using K-map 07

- The input to the 4-bit binary to gray converter circuit is a 4-bit binary and the output is 4-bit binary gray code.
- There are 16 possible combinations of 4-bit binary and all of them are valid.
- The 4-bit binary and corresponding gray code are shown in below table:

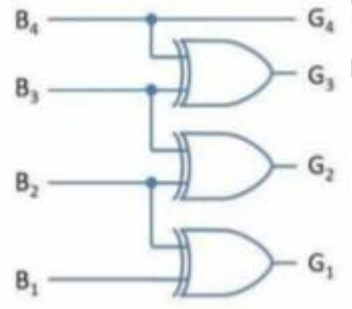
4-bit Binary				4-bit Gray			
B_4	B_3	B_2	B_1	G_4	G_3	G_2	G_1
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

- From conversion table we observe expression from output G4, G3, G2, G1 as follows:
- G1=(1,2,5,6,9,10,13,14)
- G2=(2,3,4,5,10,11,12,13)
- G3=(4,5,6,7,8,9,10,11)
- G4=(,9,10,11,12,13,14,15)

- The K-maps for G_4, G_3, G_2, and G_1 and their minimization are shown below:



- The minimal expression for the outputs obtained from k map are:
- G4=b4
- G3=b4'b3+b4'b3
- G2=b3'b2+b3b2'
- G1=b2'b1+b2b1'



OR

Q2(c) Draw a two input TTL NAND gate and explain its operation.07

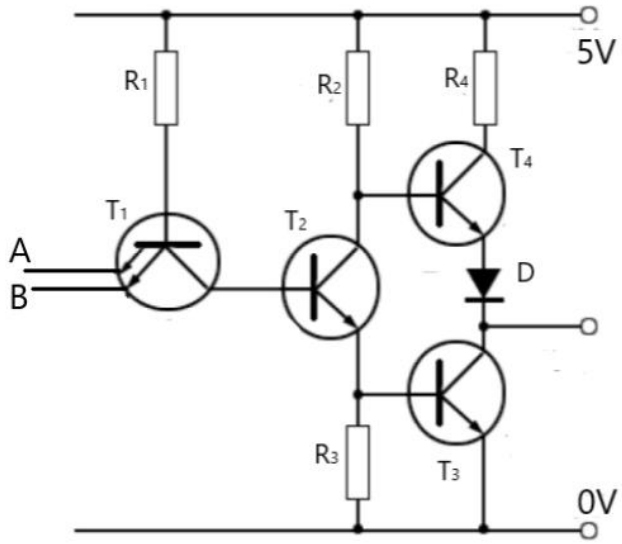
- Nand gate= And gate + Not gate
- Truth Table:

A	B	Y=(A.B)'
0	0	1
0	1	1
1	0	1
1	1	0

- Output is low when both input are high
- Nand gate uses multiple NPN transistors.
- Input translator act as steering diodes
- Output transistor act as inverter

Advantages:

- Fast switching speed
- High fan out
- Noise immunity better than CMOS in some cases



Q3(a) Explain race around condition in jk flip flop(3m)

The race-around condition in a JK flip-flop occurs when both inputs J=1J=1 and K=1K=1, and the clock pulse remains HIGH for a long time. In this case, the output keeps toggling rapidly within the same clock pulse, making the output unstable and unpredictable.

1. **Normal JK Flip-Flop Operation**

❖ Inputs: JJ, KK, Clock.
❖ Outputs: QQ, Q′Q'.
❖ When J=K=1J=K=1, the flip-flop is supposed to toggle its output at each clock edge.

2. **Race-Around Condition**

❖ If the clock pulse width is too long, the flip-flop doesn't just toggle once.
❖ Instead, the output keeps toggling multiple times during the same clock HIGH period.
❖ This repeated toggling is called the race-around condition.

3. **Why It Happens**

❖ Because the output feeds back into the input, the circuit keeps updating itself as long as the clock is HIGH.

❖ The faster the propagation delay compared to the clock pulse width, the more toggles occur.

4. **Effect**

❖ The output becomes uncertain and unstable.

❖ Designers cannot predict whether the final output will be 0 or 1 after the clock pulse ends.

5. **Solution**

❖ Reduce clock pulse width: Make it shorter than the propagation delay.

❖ Use edge-triggering: Flip-flop responds only at the rising/falling edge of the clock.

❖ Master-Slave JK Flip-Flop: Two flip-flops in series (master and slave) ensure toggling happens only once per clock cycle.

Q3 (b) Design a 1 - bit Magnitude Comparator. 04

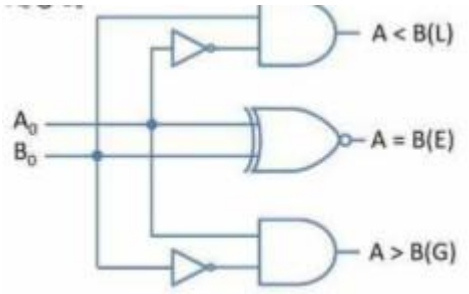
➢ Let the 1-bit numbers be A =A0 and B =B0

➢ If A0=1 and B0=0 then A>B (A>B; G=A0B0')

➢ If A0=0 and B0=1 then A>B (A<B; L=A0'B0)

➢ If A0=1 and B0=1 then A=B(A=B; E=A0'.B0')

A0	B0	L	E	G
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0



Q3 (c) Explain Hamming codes for error correction with a suitable example. 07

✧ Hamming code is error correcting code used to detect and correct single bit error in transmitted data.

✧ It was developed by richard hamming and is widely used in digital communication and memory systems.

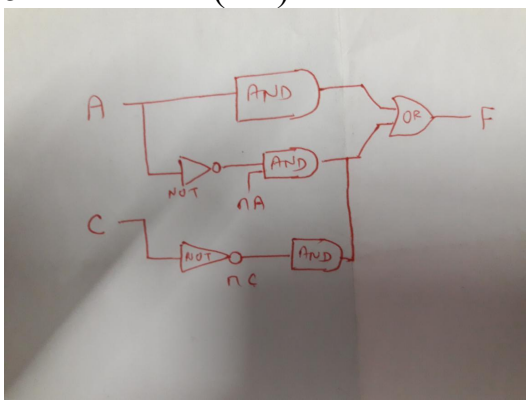
- ✧ It works by adding redundant bits (parity bits) at specific position to data words.
- ✧ These parity bits help to identify exactly which bit is in error, not just that an error occurred.
- ✧ Parity bits are placed at position that are power of 2: 1,2,4,8…
- ✧ For data word with m data bits, the number of parity bits® is calculated using: 2 ^ r >= m + r + 1
- ✧ Each parity bit checks some specific bit position based on binary positional representation.
- ✧ During transmission, the receiver recalculates parity bits and compress them with received parity bits to form a syndrome.
- ✧ The syndrome gives the binary position of bit in error.
- ✧ If syndrome = 0 then no error
- ✧ If syndrome !=0 then bit position=error location , correct by flipping that bit.
- ✧ Thus hamming code provides single bit error correction and double bit error detection making data transfer more reliable.

OR

Q.3 (a) Find expression for the following and implement using logic gates. F(A,B,C,D) = πM(0,2,3,6,7,8,9,12,13) (03)

- ✧ Given from F(A,B,C,D)= m(0,2,3,6,7,9,12,13)
- ✧ F=m0+m2+m3+m6+m7+m8+m9+m12+m13
- ✧ If you simplify sum of minterms you get:
- ✧ F=(A^C')+(A'^C)
- ✧ Circuit is using NOT, AND and OR gates:

1. nC=C'(NOT)
2. nA=A'(NOT)
3. X1=A^nC(AND)
4. X2=nA^C(AND)
5. F=X1vX2(OR)



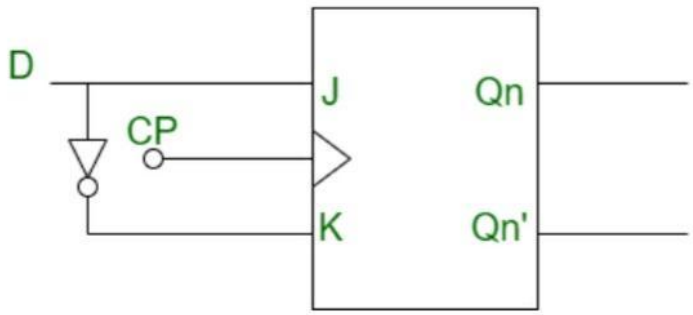
Q3 (b) Implement D flip flop using JK flip flop. (04)

Step-1: We construct the characteristic table of D flip-flop and excitation table of JK

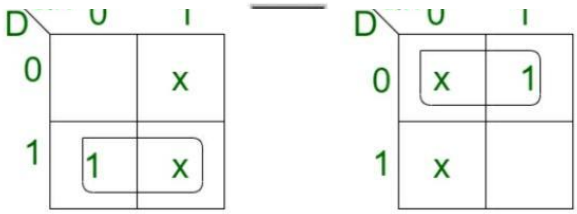
D	Qn	Qn+1	J	K
0	0	0	0	x
0	1	0	x	1
1	0	1	1	x
1	1	1	x	0

Step-2: Using the K-map we find the boolean expression of J and K in terms of D and Qn.

Step-3: We construct the circuit diagram of the conversion of JK flip-flop into D flip-flop.



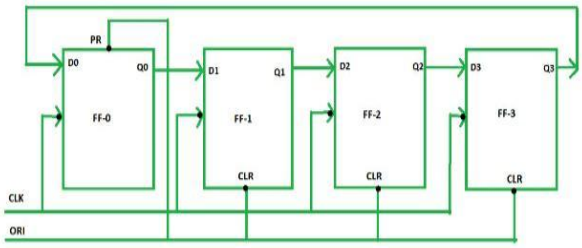
JK Flip Flop as D Flip Flop




```
J = D
K = D'
```

Q3 (c) Design a 4-bit twisted Ring Counter using JK flip flops. (07)

1. **Initial state**: 1000 (the first flip-flop is set to 1, and others are set to 0).
2. **Clock cycle 1**: The '1' bit shifts to the next flip-flop, resulting in the state 0100.
3. **Clock cycle 2**: The '1' bit shifts to the next flip-flop, resulting in the state 0010.
4. **Clock cycle 3**: The '1' bit shifts to the next flip-flop, resulting in the state 0001.
5. **Clock cycle 4**: The '1' bit shifts back to the first flip-flop, and the cycle repeats, resulting in the state 1000



Ring Counter

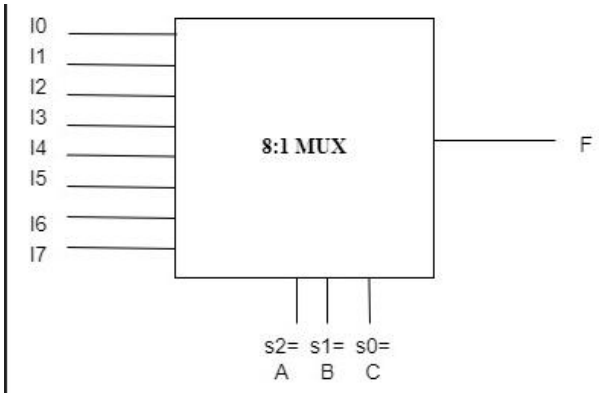
Q4 a) Differentiate Synchronous Counters and Asynchronous Counters. (03)

Feature	Synchronous Counter	Asynchronous Counter
Clocking	Common clock to all flip-flops	Clock of one flip-flop drives the next
Speed	Faster, no ripple delay	Slower due to propagation delay
Accuracy	High, synchronized state changes	Lower, outputs change sequentially
Complexity	More complex design	Simple design
Propagation Delay	Very low	High (ripple effect)
Applications	High-speed circuits, frequency division	Simple counting, low-speed applications

Q4 (b) Implement the following using 8:1 MUX. F = f(A,B,C,D) = Σm(2,4,5,7,10,14) (04)

- F(A,B,C,D)=m(2,4,5,7,10,14)
- Choose :8:1 mux selection lines: s2=A s1=B s0=C
- Mux inputs are: I0=0; I1=0, I2-D', I3=D, I4=D', I5=D, I6=0, I=D

- The remaining variable D is used to form mux data inputs.

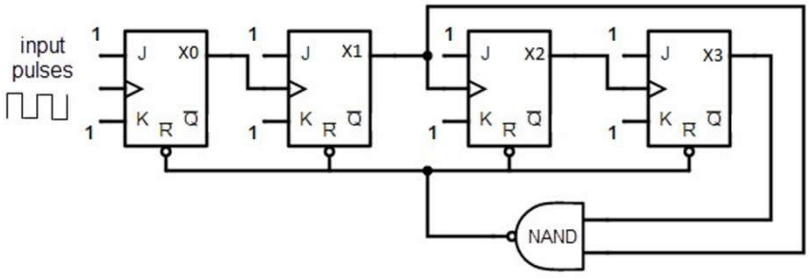


Q4 (c) Design a synchronous BCD counter using J-K flip-flops. (07)

An asynchronous BCD counter using JK flip flop is 4 bit ripple cycle that cycles 0000 to 1001(0 to 9) then resets built with four JK FFs where J and K are set to 1 for toggling the frist flip flop(ff) clocks externally and subsequent ff clock from previous Q output requiring added login to detect 9 state and reset counter back to 0 creating ripple effect.

Key components:

- **Flip flop:** four jk flip flops
- **J & K Input:** All j and k input are tied to logic 1 to make them toggle on clock edge.
- **Clock:** the first flip flop gets external clock signal
- **Ripple connections:** Q0 clocks ff1 Q1 clock ff2 Q2 clock ff3
- **Reset logic:** A circuit detect when count reaches 10 or specifically after 0 to trigger reset.



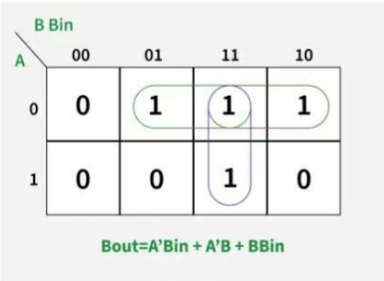
OR

Q4 (a) Implement full subtractor using 3:8 decoder and write a truth table. (03)

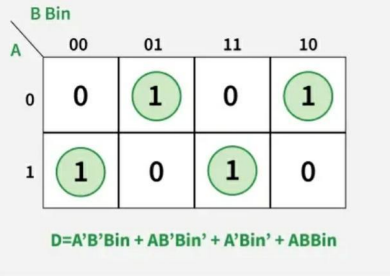
Truth Table of Full Subtractor

Input			Output	
A	B	Bin	D	Bout
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

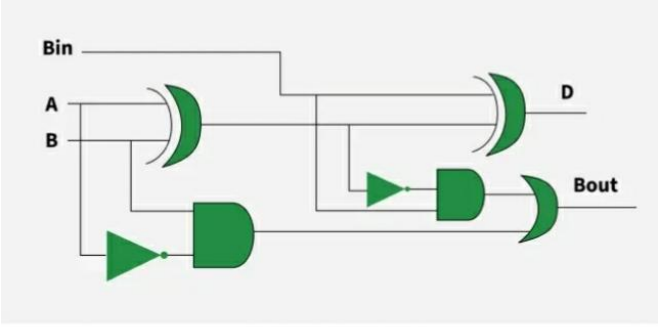
From above table we can draw the K-Map as shown for "difference" and "borrow".



K-Map for Borrow



K-Map for Difference



Logic Circuit for Full Subtractor

Full Subtractor 3:8

Q4 (b) Explain the specifications of Digital to Analog Converters. (04)

- **Resolution:** smallest change in analog output for 1 bit change in digital input.
- **Accuracy:** It indicates how close the actual output is to idea output. Usually expressed as percentage of full scale output.
- **Linearity:** A DAC should ideally increase its analog output in equal steps for each digital input increase.
- **Monotonicity:** A DAC is monotonicity if output never decrease when digital input increase.
- **Full scale output:** Max analog output DAC can produce when digital input=all 1s.
- **Setting time:** Time require for DAC output to reach and remain within specified error band after digital input change
- **Conversion time:** Time taken by DAC to convert digital value into its corresponding analog voltage.
- **Offset error:** The difference between expected output and actual output when digital input is zero.
- **Gain error:** Difference between actual slope of DAC output and the ideal slope.
- **Power consumption:** Total power required by DAC output for relliable operation.

Q4 (c) Explain Successive Approximation type A/D converter. (07)

- The Successive approximation ADC converts an analog input signal into digital output by using binary search method.
- It consist of :

i. Simple and hold circuit
ii. Successive approximation register
iii. Internal DAC
iv. Comparator
v. Control logic

- The SAR ADC works by approximating the analog input voltage step by step, starting from msb to lsb.
- The SAR register first set msb=1, converts it to analog voltage using internal DAC and compares it with input voltage.
- If DAC output is less than or equal to input the bit is kept 1; otherwise reset to 0.

- Next the SAR sets next bit to 1 and repeat compare and update process.
- This binary search process continues for N steps for N-bit DAC
- After all bits are tested th SAR register holds final digital equivalent of the analog input
- Conversion time is fixed and equal to N clock cycle which is major advantage.
- SAR ADC provides high resolution, medium to high speed, good accuracy, and low power consumption.

Q5 (a) Differentiate Static RAM and Dynamic RAM. (03)

Feature	SRAM	DRAM
Storage element	Flip-flop	Capacitor
Refreshing	Not required	Required
Speed	Fast	Slow
Cost	High	Low
Density	Low	High
Application	Cache	Main memory
Full form	Static Random Access Memory	Dynamic Random Access Memory

Q5 (b) Write a short note on FPGA. (04)

- **FPGA (Field Programmable Gate Array)** is a semiconductor device that can be programmed after manufacturing.
- **Structure:**
  - Contains **Configurable Logic Blocks (CLBs)**, **Input/Output Blocks (IOBs)**, and **Programmable Interconnects**.
  - Logic blocks can be configured to perform complex digital functions.
- **Features:**

  1. Re programmable hardware.
  2. Supports parallel processing.
  3. Flexible and fast prototyping.

- **Applications:**
  - Digital signal processing.
  - Communication systems.
  - Aerospace and defense.

- Embedded systems.

Q5 (c) Explain the operation of Dual-slope A/D converter. (07)

- **Principle:** Converts analog input into digital by integrating input voltage for a fixed time, then de integrating with a reference voltage.
- **Block Diagram Components:**
  - Integrator (Op-Amp + capacitor).
  - Comparator.
  - Counter.
  - Control logic.

Operation Steps:

1. **Integration Phase:**
   - Input voltage applied to integrator for fixed time TT.
   - Output ramps up linearly.
2. **De-integration Phase:**
   - Reference voltage of opposite polarity applied.
   - Output ramps down towards zero.
3. **Counting:**
   - Counter measures time taken for output to return to zero.
   - This count is proportional to input voltage.
4. **Digital Output:**
   - Final digital value
   - represents the analog input.

Advantages:

- High accuracy.
- Good noise rejection (especially for 50 Hz/60 Hz power line noise).
- Conversion time independent of input signal amplitude.

Applications:

- Digital voltmeters.
- Instrumentation system

OR

Q.5 (a) Explain basic structure of a CCD (Charge Coupled Device). (03)

- **CCD** is a semiconductor device used for image sensing.
- **Structure:**
    1. Array of MOS capacitors.
    2. Charges generated by incident light are stored in capacitors.
    3. Charges are transferred sequentially (coupled) to output amplifier.
- **Applications:**
    1. Digital cameras.
    2. Scanners.
    3. Astronomy instruments

Q.5 (b) Write a short note on Programmable Array Logic. (04)

- **PAL** is a programmable logic device with a **programmable AND array** and a **fixed OR array**.
- **Features:**
    1. Faster than PROM.
    2. Used for implementing combinational logic.
    3. Limited flexibility compared to PLA.
- **Applications:**
    1. Control logic.
    2. State machines.
    3. Digital circuits.

Q.5 (c) Explain various types of Read Only Memory. (07)

Ans: ROM (Read Only Memory) stores data permanently or semi-permanently. Types are:

1. **Mask ROM:**
    - Programmed during manufacturing.
    - Low cost, high volume production.
2. **PROM (Programmable ROM):**
    - User programmable once.
    - Uses fusible links.
3. **EPROM (Erasable PROM):**
    - Can be erased by UV light.
    - Reprogrammable.
4. **EEPROM (Electrically Erasable PROM):**
    - Can be erased electrically.
    - Byte-level reprogramming possible.
5. **Flash ROM:**
    - Fast electrical erasing.

- Used in USB drives, memory cards, SSDs.

Applications:

- Firmware storage.
- Embedded systems.
- Consumer electronics