सार्थक

प्राविधिक शिक्षा परिषद् उत्तर प्रदेश द्वारा
स्वीकृत नवीनतम पाठ्यक्रमानुसार

डिजिटल इलैक्ट्रॉनिक्स

प्रयोगात्मक कार्य सहित

[DIGITAL ELECTRONICS]

With Practicals

[द्वितीय वर्ष (तृतीय सेमेस्टर) इलैक्ट्रिकल इंजीनियरिंग एवं इलैक्ट्रॉनिक्स इंजीनियरिंग के विद्यार्थियों के लिए]

राहुल मणि उपाध्याय
(M. Tech.)
प्रवक्ता इलैक्ट्रॉनिक्स
बुद्धा इन्सटीटयूट ऑफ टेक्नोलॉजी
गोरखपुर

जय प्रकाश नाथ पब्लिकेशन्स
गाँधी आश्रम चौराहा
नौचन्दी रोड, मेरठ – 250 002 (यू० पी०)

Scanned by CamScanner

विषय सूची

क्रम अध्याय	पृष्ठ संख्या
1. डिजिटल इलेक्ट्रॉनिक्स	1–25
2. लॉजिक गेट्स	26–55
3. लॉजिक परिवार	56–71
4. डिस्प्ले युक्तियाँ, बाइनरी एडर सर्किट, एनकोडर्स डिकोडर्स, मल्टीप्लैक्सर्स तथा डिमल्टीप्लैक्सर्स	72–88
5. फ्लिप-फ्लॉप	88–104
6. मल्टीवाइब्रेटर तथा A/D, D/A कनवर्टर्स	105–114
7. सेमीकन्डक्टर मेमोरी	115–118
परिशिष्ट	119–139
प्रयोगात्मक कार्य	140–149

❑

Scanned by CamScanner

अध्याय 1

डिजिटल इलेक्ट्रॉनिक्स (Digital Electronics)

1.1 परिचय (Introduction)

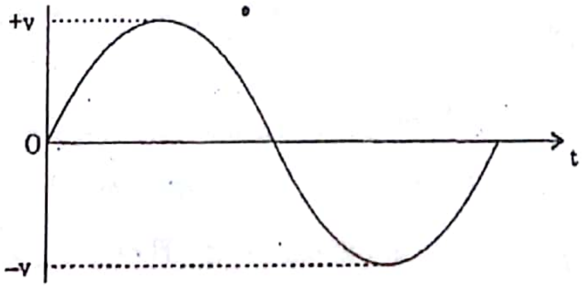
- डिजिटल इलेक्ट्रॉनिक्स का उपयोग हम डाटा प्रोसेसिंग, कन्ट्रोल सिस्टम, संचार तथा मापन में करते हैं।
- एनेलॉग इलेक्ट्रॉनिक्स में आउटपुट वोल्टेज इनपुट वोल्टेज के अनुसार परिवर्तित होते रहते हैं अर्थात् आउटपुट वोल्टेज के अनन्त मान होते हैं।
- वह इलेक्ट्रॉनिक सर्किट जो दो स्टेट ऑपरेशन के लिए डिजाइन किया जाता है, डिजिटल सर्किट कहलाता है। इस प्रकार डिजिटल सर्किट में आउटपुट वोल्टेज की केवल दो अवस्थाएँ होती है।
- इलेक्ट्रॉनिक्स की वह ब्रांच जिसमें हम डिजिटल सर्किट का अध्ययन करते हैं, डिजिटल इलेक्ट्रॉनिक्स (digital electronics) कहलाती है।

सिग्नल को दो भागों में बाँटा गया है—

1. एनेलॉग सिग्नल,
2. डिजिटल सिग्नल।

1.2 एनेलॉग सिग्नल (Analog Signal)

- एनेलॉग सिग्नल के अनन्त मान होते हैं तथा ये सिग्नल समय के साथ-साथ लगातार परिवर्तित होते रहते हैं।
- Sine wave, Triangular wave इत्यादि एनेलॉग सिग्नल के उदाहरण हैं।

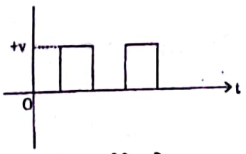


चित्र : 1.1 एनेलॉग सिग्नल

Scanned by CamScanner

1.3 डिजिटल सिग्नल (Digital Signal)

- डिजिटल सिग्नल के दो डिस्क्रीट स्तर (discrete level) होते हैं। यह सिग्नल सदैव Low अथवा High में से किसी एक स्तर पर होता है।



चित्र : 1.2 डिजिटल सिग्नल

- डिजिटल सिग्नल समय के साथ-साथ लगातार परिवर्तित नहीं होते हैं।

1.4 एनेलॉग सिग्नल तथा डिजिटल सिग्नल की तुलना (Comparison of Analog Signal and Digital Signal)

(UPBTE 2013, 14, 16)

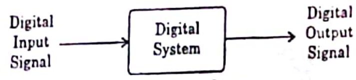
क्रम सं०	पैरामीटर	एनेलॉग सिग्नल	डिजिटल सिग्नल
1.	आउटपुट वोल्टेज के मान	अनन्त (Infinite)	दो (Low अथवा High)
2.	नेचर	कॉन्टीनुअस (Continuous)	डीस्क्रीट (Discrete)
3.	सोर्स	सिग्नल जनरेटर, ट्रांसड्यूसर	कम्प्यूटर
4.	उदाहरण	Sine wave, Triangular wave	बाइनरी सिग्नल

1.5 एनेलॉग सिस्टम (Analog Systems)

- यह सिस्टम (System) जो एनेलॉग सिग्नल (analog signals) को प्रोसेस करता है, एनेलॉग सिस्टम कहलाता है।
- एनेलॉग सिस्टम के निम्न उदाहरण हैं—
  1. फिल्टर (Filters),
  2. एम्प्लीफायर (Amplifiers),
  3. सिग्नल जेनरेटर (Signal Generators),
  4. पावर सप्लाई (Power Supplied)।
- एनेलॉग सिस्टम की परिशुद्धता (Accuracy) कम होती है जिस कारण यह कम रिलायबल (reliable) होता है।

1.6 डिजिटल सिस्टम (Digital Systems)

- यह सिस्टम जो डिजिटल सिग्नल को प्रोसेस करता है, डिजिटल सिस्टम कहलाता है।
- डिजिटल सिस्टम के इनपुट एवं आउटपुट दोनों सिग्नल, डिजिटल होते हैं।



चित्र : 1.3 डिजिटल सिस्टम

- डिजिटल सिस्टम के निम्न उदाहरण हैं—
  1. फ्लिप-फ्लाप (Flip-Flops),
  2. काउन्टर (Counter),
  3. माइक्रोप्रोसेसर (Microprocessors),
  4. डिजिटल कैलकुलेटर (Digital Calculators),
  5. कम्प्यूटर (Computers)।

1.7 डिजिटल सिस्टम के लाभ (Advantages of Digital Systems)

1. डिजिटल सिस्टम की परिशुद्धता (accuracy) उच्च होती है।
2. इन पर शोर (noise) का प्रभाव कम पड़ता है।
3. डिजिटल सिस्टम, एनेलॉग सिस्टम की अपेक्षा ज्यादा विश्वसनीय (reliable) होते हैं।
4. डिजिटल सिस्टम के बीच कम्यूनिकेशन (communication) आसानी से हो जाता है।
5. डिजिटल सिस्टम की डिजाइन (design) आसानी से हो जाती है।
6. इन पर ताप का प्रभाव बहुत कम पड़ता है।

एनेलॉग तथा डिजिटल सिग्नल का तुलनात्मक चार्ट
(Comparison of Analog and Digital Systems)

क्रम सं०	पैरामीटर	एनेलॉग सिग्नल	डिजिटल सिग्नल
1.	सिग्नल प्रोसेस्ड (Signal Processed)	एनेलॉग सिग्नल	डिजिटल सिग्नल
2.	डिस्प्ले (Display)	एनेलॉग मीटर	डिजिटल डिस्प्ले (LED एवं LCD)
3.	परिशुद्धता (Accuracy)	कम	ज्यादा
4.	डिजाइन (Design)	कठिन	आसान
5.	मैमोरी (Memory)	मैमोरी नहीं होती	मैमोरी होती है
6.	ताप का प्रभाव (Effect of temperature)	अधिक	कम
7.	शोर का प्रभाव (Effect of noise)	अधिक	कम
8.	सिस्टम के बीच कम्यूनिकेशन (Communication between system)	आसानी से नहीं होता	आसानी से होता है
9.	उदाहरण (Examples)	फिल्टर, एम्प्लीफायर पावर सप्लाई, सिग्नल जनरेटर	काउन्टर, रजिस्टर्स, माइक्रोप्रोसेसर, कम्प्यूटर

Scanned by CamScanner

1.8 नम्बर सिस्टम (Number System)

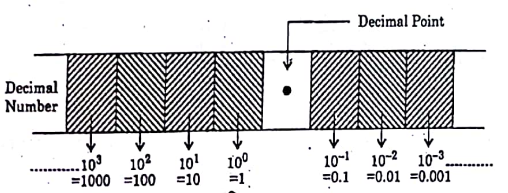
- डिजिटल इलेक्ट्रॉनिक्स में स्थिर संख्याओं को युग्मित कर अन्य संख्याओं को प्रदर्शित किया जाता है।
- नम्बर सिस्टम (number system) को निम्नलिखित चार भागों में बाँटा गया है—
  1. डेसिमल नम्बर सिस्टम (Decimal number system)
  2. बाइनरी नम्बर सिस्टम (Binary number system)
  3. ऑक्टल नम्बर सिस्टम (Octal number system)
  4. हेक्साडेसिमल नम्बर सिस्टम (Hexadecimal number system)
- प्रत्येक नम्बर सिस्टम के लिए एक बेस (base) होता है और प्रत्येक नम्बर सिस्टम का अधिकतम मान (largest value) उसके बेस से एक कम होता है।

नम्बर सिस्टम और उसका बेस

S.No.	Name of Number System	Base
1.	Decimal	10
2.	Binary	2
3.	Octal	8
4.	Hexadecimal	16

1.9 डेसिमल नम्बर सिस्टम (Decimal Number System)

- वह नम्बर सिस्टम जो हम प्रारम्भ से ही स्कूलों में पढ़ते चले आ रहे हैं, उसे डेसिमल नम्बर सिस्टम कहते है।
- डेसिमल नम्बर सिस्टम के 10 विशेष प्रतीक होते हैं, 0, 1, 2, 3, 4, 5, 6, 7, 8 तथा 9.
- डेसिमल नम्बर का बेस 10 होता है तथा इसकी अधिकतम संख्या 9 होती है।



चित्र : 1.4

- ऊपर दिए गए चित्र के द्वारा हम डेसिमल नम्बर को प्रदर्शित करने का अध्ययन करेंगे।
- डेसिमल प्रणाली में लिखी गई प्रत्येक संख्या के प्रत्येक गुणक का अपना एक स्थानीय मान (weighted value) होता है। प्रत्येक संख्या के स्थानीय मान को चित्र में प्रदर्शित किया गया है।

MSD (Most Significant Digit)

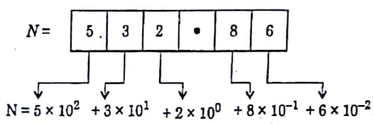
- बायीं तरफ की वह संख्या जिसका स्थानीय मान अधिकतम होता है, MSD कहलाती है।

LSD (Least Significant Digit)

- दायीं तरफ की वह संख्या जिसका स्थानीय मान सबसे कम होता है, LSD कहलाती है।

उदाहरण—डेसिमल नम्बर 532.86 को प्रदर्शित करना।

हल—



चित्र : 1.5

इस प्रकार दी गई संख्या 532.86 को प्रदर्शित किया जा सकता है।

1.10 बाइनरी नम्बर सिस्टम (Binary Number System)

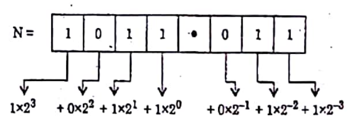
- बाइनरी नम्बर सिस्टम के दो विशेष प्रतीक होते हैं—0 तथा 1.
- बाइनरी नम्बर सिस्टम, डेसिमल नम्बर सिस्टम की तरह कार्य करता है। इनमें विशेष अन्तर केवल बेस का है। बाइनरी नम्बर का बेस 2 होता है, जबकि डेसिमल नम्बर का बेस 10 होता है।

1.10.1 बाइनरी संख्या को डेसिमल में परिवर्तित करना

बाइनरी संख्या को डेसिमल में बदलने के लिए बाइनरी संख्या के प्रत्येक अंक को 2^{n-1} से गुणा किया जाता है, जहाँ n बाइनरी संख्या के उस अंक की स्थिति को प्रदर्शित करता है। तत्पश्चात् सभी गुणनफलों को जोड़कर बाइनरी संख्या के तुल्य डेसिमल संख्या ज्ञात की जा सकती है।

उदाहरण— बाइनरी नम्बर 1011.011 का डेसिमल तुल्यांक ज्ञात करना।

हल—**Step 1 :** दी गयी बाइनरी संख्या को 2 के पावर के रूप में प्रदर्शित करने पर



चित्र : 1.6

Step 2 : डेसिमल तुल्यांक

$$N = (1\times 2^{3)} + (0\times 2^2) + (1\times 2^1) + (1\times 2^0) + (0\times 2^{-1}) + (1\times 2^{-2}) + (1\times 2^{-3})$$

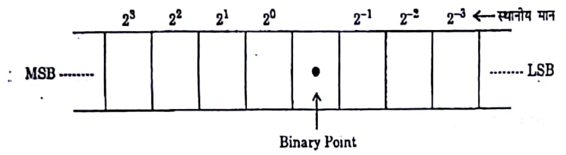
$$N = (1\times 8) + (0\times 4) + (1\times 2) + (1\times 1) + (0\times 0.5) + (1\times 0.25) + (1\times 0.125)$$

$$N = 8 + 2 + 1 + 0.5 + 0.25 + 0.125$$

$$N = (11.375)_{10}$$ **Ans.**

Scanned by CamScanner

- बाइनरी नम्बर का बेस 2 होता है, अत: इसकी अधिकतम संख्या 1 होती है।
- नीचे दिए गए चित्र के द्वारा हम बाइनरी नम्बर को प्रदर्शित करने का अध्ययन करेंगे।



चित्र : 1.7

- डेसिमल प्रणाली की तरह बाइनरी प्रणाली में लिखी गई प्रत्येक संख्या के प्रत्येक गुणक का अपना एक स्थानीय मान होता है। प्रत्येक संख्या के स्थानीय मान को चित्र में प्रदर्शित किया गया है।
- बाइनरी डिजिट (0 तथा 1) को बिट्स (bits) भी कहते हैं। अतः हम कह सकते हैं कि बाइनरी सिस्टम 2-बिट सिस्टम है।
- बायीं तरफ की वह बिट जिसका स्थानीय मान अधिकतम होता है, Most Significant Bit (MSB) कहलाती है।
- दायीं तरफ की वह बिट जिसका स्थानीय मान सबसे कम होता है, Least Significant Bit (LSB) कहलाती है।

1.10.2 डेसिमल संख्या को बाइनरी में परिवर्तित करना

- डेसिमल संख्या को बाइनरी में परिवर्तित करने के लिए डेसिमल संख्या को लगातार 2 से विभाजित किया जाता है।
- प्रत्येक विभाजन के पश्चात् शेषफल को रिकॉर्ड किया जाता है।
- बाइनरी संख्या प्राप्त करने के लिए सभी शेषफलों को उल्टे क्रम में लिखते हैं।

उदाहरण— $(105)_{10}$ को बाइनरी में परिवर्तित करना।

हल—

		शेष	
2	105 ← दी हुई संख्या		
2	52	1	LSB
2	26	0	↑
2	13	0	↑
2	6	1	↑
2	3	0	↑
2	1	1	↑
	0	1	MSB

शेषफलों को उल्टे क्रम में लिखने पर $(105)_{10} = (1101001)_2$

1.10.3 डेसिमल भिन्न को बाइनरी में परिवर्तित करना

- डेसिमल भिन्न को बाइनरी में परिवर्तित करने के लिए दशमलव भिन्न को लगातार 2 से गुणा किया जाता है।
- प्राप्त गुणनफल के दशमलव से बाएँ के अंक नोट कर लिये जाते हैं, जिसे कैरी (carry) कहते हैं।
- कैरी अंकों को सीधे क्रम में लिखने पर बाइनरी संख्या प्राप्त हो जाती है।

उदाहरण—डेसिमल संख्या $(0.42)_{10}$ को बाइनरी में परिवर्तित करना।

हल—

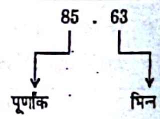
डेसिमल भिन्न	× बेस	गुणनफल	कैरी
0.42	×2	0.84	0 MSB
0.84	×2	1.68	1
0.68	×2	1.36	1
0.36	×2	0.72	0
0.72	×2	1.44	1 LSB

अत: $(0.42)_{10} = (0.01101)_2$

नोट—

हम यहाँ इस प्रक्रिया को आगे भी बढ़ा सकते हैं लेकिन कनवर्ज़न (Conversion) करते समय इस प्रक्रिया को 5 डिजिट तक ही करते हैं।

उदाहरण– $(85.63)_{10}$ को बाइनरी में परिवर्तित कीजिए।



चित्र : 1.8

हल—

Step 1 : पूर्णांक (integer) और भिन्न को अलग करने पर

2	85		
2	42	1	LSB
2	21	0	↑
2	10	1	↑
2	5	0	↑
2	2	1	↑
2	1	0	↑
	0	1	MSB

Step 2 : पूर्णांक को बाइनरी में बदलना

Scanned by CamScanner

$(85)_{10} = (1010101)_2$

Step 3 : भिन्न को बाइनरी में बदलना

$0.63 \times 2 = 1.26$	1	MSB
$0.26 \times 2 = 0.52$	0	↓
$0.52 \times 2 = 1.04$	1	↓
$0.04 \times 2 = 0.08$	0	↓
$0.08 \times 2 = 0.16$	0	LSB
	↑ शेष	

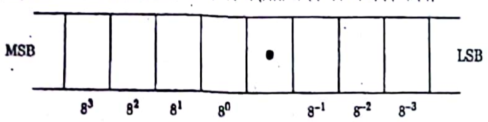
$(0.63)_{10} = (10100)_2$

Step 4 : Step 2 और Step 3 के परिणाम को मिलाने पर

$(85.63)_{10} = (1010101.10100)_2$ Ans.

1.11 ऑक्टल नम्बर सिस्टम (Octal Number System)

- ऑक्टल नम्बर सिस्टम का बेस "8" है।
- ऑक्टल नम्बर सिस्टम के प्रत्येक डिजिट के विभिन्न मान है जो 0 से 7 तक होते हैं। (0, 1, 2, 3, 4, 5, 6, and 7)
- ऑक्टल नम्बर की अधिकतम डिजिट का मान 7 होता है।
- नीचे दिए गए चित्र के द्वारा हम ऑक्टल नम्बर को प्रदर्शित करने का अध्ययन करेंगे।



चित्र : 1.9

- डेसिमल तथा बाइनरी संख्या की तरह ऑक्टल प्रणाली में भी प्रत्येक संख्या के प्रत्येक गुणांक का अपना एक स्थानीय मान होता है। प्रत्येक संख्या के स्थानीय मानों को चित्र में प्रदर्शित किया गया है।

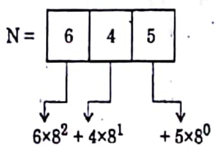
1.11.1 ऑक्टल संख्या को डेसिमल में परिवर्तित करना

ऑक्टल संख्या को डेसिमल में बदलने के लिए ऑक्टल संख्या के प्रत्येक अंक को 8^{n-1} से गुणा किया जाता है, जहाँ n ऑक्टल संख्या के उस अंक की स्थिति को प्रदर्शित करता है। तत्पश्चात् सभी गुणनफलों को जोड़कर ऑक्टल संख्या के तुल्य डेसिमल संख्या प्राप्त की जाती है।

उदाहरण—ऑक्टल संख्या 645 का डेसिमल तुल्यांक ज्ञात करना।

हल—

Step 1 : दिए गए ऑक्टल संख्या को 8 के पावर के रूप में प्रदर्शित करने पर



चित्र : 1.10

Step 2 : डेसिमल तुल्यांक

$$N = (6\times8^2)+(4\times8^1)+(5\times8^0)$$

$$N = (6\times64)+(4\times8)+(5\times1)$$

$$N = (421)_{10}$$

1.11.2 डेसिमल संख्या को ऑक्टल में परिवर्तित करना

डेसिमल संख्या को ऑक्टल संख्या में परिवर्तित करने के लिए डेसिमल संख्या को 8 से लगातार भाग दिया जाता है तथा प्रत्येक विभाजन की क्रिया के पश्चात् प्राप्त शेषफलों को उल्टे क्रम में लिखने पर हमें ऑक्टल संख्या प्राप्त हो जाती है।

उदाहरण— डेसिमल संख्या $(204)_{10}$ को ऑक्टल में परिवर्तित करना।

हल—

8	204	4	LSD
8	25	1	↑
	3	3	MSD
		↑ शेष	

चित्र : 1.11

- शेषफलों को उल्टे क्रम में लिखने पर $(204)_{10} = (314)_8$.

1.11.3 ऑक्टल संख्या को बाइनरी में परिवर्तित करना

- ऑक्टल संख्या का बेस "8" होता है। अतः इसकी अधिकतम मान की संख्या 7 होती है।
- ऑक्टल संख्या के प्रत्येक डिजिट (0, 1, 2, ..., 7) के लिए बाइनरी तुल्यांक निश्चित होता है, जो इस प्रकार है—

Octal Number	Equivalent Binary Number
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

चित्र : 1.12 ऑक्टल नम्बर का बाइनरी तुल्यांक

उदाहरण— ऑक्टल संख्या $(364)_8$ को बाइनरी में परिवर्तित कीजिए।

हल— दी हुई ऑक्टल संख्या = 364

- प्रत्येक संख्या का बाइनरी तुल्यांक लिखने पर

Scanned by CamScanner

3	6	4
011	110	100

- अत: दी हुई ऑक्टल संख्या का बाइनरी तुल्यांक

$$(364)_8 = (011110100)_2$$

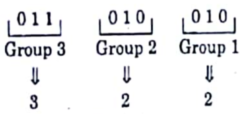
1.11.4 बाइनरी संख्या को ऑक्टल संख्या में परिवर्तित करना

- बाइनरी संख्या को ऑक्टल में परिवर्तित करने के लिए LSB से प्रारम्भ कर 3-3 बिट्स के समूह बनाए जते हैं उसके बाद प्रत्येक समूह के लिए उसके तुल्य ऑक्टल संख्या लिख दी जाती है।

उदहारण— बाइनरी संख्या $(11010010)_2$ को ऑक्टल संख्या में परिवर्तित कीजिये।

हल— दी हुई बाइनरी संख्या = 11010010

3-3 बिट का समूह बनाने पर



चित्र : 1.13

यहां दी हुई संख्या के ग्रुप 3 में तीन का समूह बनाने के लिए 0 को आगे लगा देते हैं।

ऑक्टल तुल्यांक = 322

$$(11010010)_2 = (322)_8$$

1.12 हेक्साडेसिमल नम्बर सिस्टम (Hexadecimal Number System)

- हेक्साडेसिमल नम्बर सिस्टम का बेस "16" होता है।
- हेक्साडेसिमल नम्बर की अधिकतम डिजिट का मान 15 होता है।
- हेक्साडेसिमल नम्बर में कुल 16 संख्याएँ आती है जिनमें 0 से 9 तक सभी डिजिट तथा अंग्रेजी वर्णमाला के अक्षर A, B, C, D, E और F आते हैं।
- नीचे हेक्साडेसिमल नम्बर की सभी डिजिट एवं उसके मान को दर्शाया गया है—

Hexadecimal Digit	Value
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12
D	13
E	14
F	15

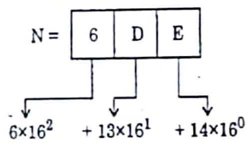
चित्र : 1.14 हेक्साडेसीमल डिजिट का मान

1.12.1 हेक्साडेसिमल को डेसिमल में परिवर्तित करना

- हेक्साडेसिमल को डेसिमल में परिवर्तित करने के लिए हैक्साडेसिमल के प्रत्येक अंक को 16^{n-1} से गुणा किया जाता है जहां n हेक्साडेसिमल नम्बर की उस अंक की स्थिति को प्रदर्शित करता है उसके बाद सभी गुणनफलों को जोड़कर हेक्साडेसिमल नम्बर के तुल्य डेसिमल नम्बर प्राप्त हो जाता है।

उदहारण—हेक्साडेसिमल नम्बर 6DE का डेसिमल तुल्यांक ज्ञात कीजिए।

हल— Step 1 : दी गयी हेक्साडेसिमल संख्या को 16 के पावर के रूप में प्रदर्शित करना।



चित्र : 1.15

Step 2 : डेसिमल तुल्यांक

$$N = 6DE = (6\times16^2)+(132\times16^1)+(14\times16^0)$$

$$N = (6\times256)+(13\times16)+(14\times1)$$

$$N = (1758)_{10}$$

अत: $(6DE)_{16} = (1758)_{10}$

1.12.2 डेसिमल संख्या को हेक्साडेसिमल में परिवर्तित करना

- डेसिमल संख्या को हेक्साडेसिमल में परिवर्तित करने के लिए डेसिमल संख्या को 16 से लगातार भाग दिया जाता है तथा प्रत्येक विभाजन की क्रिया के पश्चात् शेषफलों को उल्टे क्रम में लिखने पर हमें हेक्साडेसिमल संख्या प्राप्त हो जाती है।

Scanned by CamScanner

उदाहरण—डेसिमल संख्या 259 का हेक्स तुल्यांक ज्ञात कीजिए।

हल—

		शेष	
16	259	3	LSD ↑
16	16	0	
16	1	1	MSD

चित्र : 1.16

शेषफलों को उल्टे क्रम में लिखने पर

$$(259)_{10} = (103)_{16}$$

1.12.3 हेक्साडेसिमल संख्या को बाइनरी में परिवर्तित करना

- हेक्साडेसिमल के प्रत्येक अंक के लिए (0,1,2,......,9) तथा A, B, C, D, E, F का बाइनरी तुल्यांक निश्चित होता है जो नीचे दर्शाया गया है—

Hexadecimal Number	Equivalent Binary Number
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

चित्र : 1.17 हेक्साडेसिमल का बाइनरी तुल्यांक

उदाहरण—हेक्स संख्या AFB2 को बाइनरी में परिवर्तित कीजिए।

हल—दी हुई संख्या के प्रत्येक अंक का बाइनरी तुल्यांक लिखने पर

A	F	B	2
1010	1111	1011	0010

अतः दी हुई हेक्स संख्या का बाइनरी तुल्यांक

$$(AFB2)_{16} = (1010111110110010)_2$$

1.12.4 बाइनरी संख्या को हेक्साडेसिमल में परिवर्तित करना

- बाइनरी संख्या को हेक्साडेसिमल में परिवर्तित करने के लिए LSB से प्रारम्भ करके 4-4 बिट्स का समूह बनाया जाता है उसके बाद प्रत्येक समूह के लिए उसके तुल्य हेक्स संख्या लिख दी जाती है।

उदाहरण—बाइनरी संख्या $(1100010111100010)_2$ का हेक्स तुल्यांक ज्ञात कीजिए।

हल— दी हुई संख्या के अंकों का 4-4 बिट का समूह बनाने पर

1100	0101	1110	0010
Group 4	Group 3	Group 2	Group 1
⇓	⇓	⇓	⇓
C	5	E	2

अतः दी हुई संख्या का हेक्स तुल्यांक = $(C5E2)_{16}$

$$(1100010111100010)_2 = (C5E2)_{16}$$

1.12.5 हेक्साडेसिमल संख्या को ऑक्टल में परिवर्तित करना

- हेक्साडेसिमल संख्या को ऑक्टल संख्या में परिवर्तित करने के लिए पहले हेक्साडेसिमल संख्या को बाइनरी में परिवर्तित करते है, उसके बाद प्राप्त बाइनरी संख्या को ऑक्टल संख्या में परिवर्तित करते है।

उदाहरण—हेक्साडेसिमल $(25B)_{16}$ का ऑक्टल तुल्यांक ज्ञात कीजिए।

हल—Step 1 : दी हुई हेक्साडेसिमल संख्या का बाइनरी तुल्यांक

2	5	B
0010	0101	1011

प्राप्त बाइनरी संख्या = 001001011011

Step 2 : बाइनरी संख्या का ऑक्टल तुल्यांक

001	001	011	011
1	1	3	3

अतः दी हुई हेक्साडेसिमल संख्या का ऑक्टल तुल्यांकं = $(1133)_8$

$$(25B)_{16} = (1133)_8$$

1.12.6 ऑक्टल संख्या से हेक्साडेसिमल संख्या प्राप्त करना

आक्टल संख्या से हेक्साडेसिमल संख्या प्राप्त करने के लिए सर्वप्रथम ऑक्टल संख्या को बाइनरी में परिवर्तित करते हैं तथा प्राप्त बाइनरी संख्या से हेक्साडेसिमल संख्या प्राप्त की जाती है।

उदाहरण—ऑक्टल संख्या $(615)_8$ का हेक्स तुल्यांक ज्ञात करना।

हल— Step 1 : दी हुई ऑक्टल संख्या का बाइनरी तुल्यांक

6	1	5
110	001	101

Scanned by CamScanner

प्राप्त बाइनरी संख्या = 110001101

Step 2 : बाइनरी संख्या का हेक्स तुल्यांक

0001	1000	1101
1	8	D

अतः दी गई बाइनरी संख्या का हेक्स तुल्यांक $(18D)_{16}$

$(615)_8 = (18D)_{16}$

1.13 बाइनरी अर्थमेटिक (Binary Arithmetic)

- कम्प्यूटर सर्किट डेसिमल नम्बर को प्रोसेस नहीं करता, यह केवल बाइनरी नम्बर को प्रोसेस करता है।
- बाइनरी एडिशन (Binary Addition) सभी प्रकार के अन्य ऑपरेशन जैसे—बाइनरी सबट्रेक्शन, मल्टीप्लिकेशन तथा डिविजन के लिए कुंजी (key) का कार्य करती है। अतः हम सर्वप्रथम बाइनरी एडिशन (Binary Addition) का अध्ययन करेंगे।

बाइनरी एडिशन के नियम (Rules for Binary Addition)

बाइनरी संख्या को जोड़ने पर हमें आउटपुट SUM तथा CARRY प्राप्त होता है जो नीचे सत्य तालिका में दर्शाया गया है—

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

उदाहरण—बाइनरी संख्या $(10111)_2$ तथा $(11001)_2$ का योग प्राप्त कीजिए।

हल—दी हुई बाइनरी संख्या A तथा B का योग

Carry→	1	1	1	1	
A→	1	0	1	1	1
B→	1	1	0	0	1
Answer→ 1	1	0	0	0	0

अतः दी हुई बाइनरी संख्या का योग = $(110000)_2$

उदाहरण—बाइनरी संख्या $(11000)_2$ तथा $(1101)_2$ का योग प्राप्त कीजिए।

हल—दी हुई बाइनरी संख्या का योग

```
      1 1 0 0 0
  +     1 1 0 1
  -------------
    1 0 0 1 0 1
```

अतः दी हुई बाइनरी संख्या का योग = $(100101)_2$

1.14 बाइनरी सबट्रेक्शन (Binary Subtraction)

बाइनरी सबट्रेक्शन के लिए दी गई तालिका के नियमों का पालन करते हैं।

A	B	SUB	BORROW
0	0	0	0
1	0	1	0
1	1	0	0
0	1	1	1

उदाहरण—बाइनरी संख्या A = $(11011)_2$ तथा B = $(10110)_2$ को सबट्रेक्ट कीजिए।

हल—दी गई बाइनरी संख्या

A = $(11011)_2$

B = $(10110)_2$

A→	1	1	0	1	1
B→	1	0	1	1	0
Answer	0	0	1	0	1

अतः दी गई बाइनरी संख्या का सबट्रेक्शन = $(00101)_2$

1.15 बाइनरी कोडेड दशमलव [Binary Coded Decimal (BCD) Code]

BCD Code की प्रत्येक डेसिमल डिजिट चार बिट बाइनरी नम्बर को दर्शाती है।

अतः BCD Code प्रत्येक डेसिमल डिजिट को बाइनरी में प्रदर्शित करने का एक माध्यम है।

नीचे तालिका में डेसिमल नम्बर के तुल्यांक BCD कोड को दर्शाया गया है—

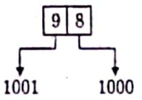
Decimal Number	BCD Code
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

उदाहरण—दिए हुए डेसिमल नम्बर को BCD Code में परिवर्तित कीजिए।

(i) 98
(ii) 78
(iii) 164

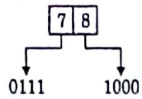
Scanned by CamScanner

हल— (i) Decimal Number



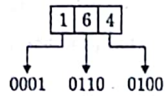
अत: 98 का BCD तुल्यांक = 10011000

(ii) Decimal Number



अत: 78 का BCD तुल्यांक = 01111000

(iii) Decimal Number



अत: 164 का BCD तुल्यांक = 000101100100

1.16 एक्सेस-3 कोड (Excess -3 Code)

- Excess-3 Code BCD Code का मॉडिफाइड फॉर्म है। BCD Code के प्रत्येक कोड में 3 को जोड़ कर Excess-3 Code प्राप्त किया जाता है।
- नीचे तालिका में डेसिमल नम्बर के तुल्यांक Excess-3 को दर्शाया गया है।

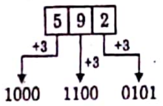
Decimal Digit	Excess-3 Code
0	0011
1	0100
2	0101
3	0110
4	0111
5	1000
6	1001
7	1010
8	1011
9	1100

- डेसीमल नम्बर 12 को BCD Code में परिवर्तित करने पर 00010010 लिखते हैं जबकि Excess-3 code प्राप्त करने के लिए डेसीमल नम्बर में 3 जोड़ दिया जाता है, अत: प्राप्त आउटपुट 01000101 होगा।

उदाहरण—दिए गए डेसीमल नम्बर को Excess-3 कोड में परिवर्तित कीजिए।

(i) 592
(ii) 403

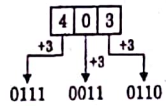
हल—(i) Decimal Number



अत: 592 का Excess-3 तुल्यांक = 100011000101

$(592)_{10}$ = 100011000101

(ii) Decimal Number



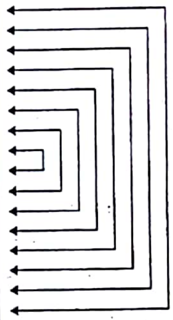
अत: 403 का Excess-3 तुल्यांक = 0111 0011 0110

$(403)_{10}$ = 0111 0011 0110

1.17 ग्रे कोड (Gray Code)

- Gray Code को Cyclic code भी कहा जाता है।

Decimal code	Gray code
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1101
9	1101
10	1111
11	1110
12	1010
13	1011
14	1001
15	1000



बाइनरी से Gray Code में कनवर्जन (Binary to Gray Conversion)

- माना यहाँ कुछ बाइनरी नम्बर $B_1, B_2, B_3, B_4, \ldots, B_n$ दिए गए हैं।
- दिए गए बाइनरी कोड का ग्रे कोड (Gray Code) में परिवर्तन

$$G_1 = B_1$$
$$G_2 = B_1 \oplus B_2$$
$$G_3 = B_2 \oplus B_3$$
$$G_4 = B_3 \oplus B_4$$
$$\vdots \quad \vdots \quad \vdots$$

Scanned by CamScanner

$G_n = B_{n-1} \oplus B_n$

उदाहरण—बाइनरी नम्बर 10111011 को ग्रे (Gray Code) में परिवर्तित कीजिए।

हल—

Binary Code =

	⊕	⊕	⊕	⊕	⊕	⊕	⊕
1	0	1	1	1	0	1	1
↓	↓	↓	↓	↓	↓	↓	↓
1	1	1	0	0	1	1	0

Gray Code = 11100110

1.18 ग्रे कोड का बाइनरी कोड में परिवर्तन (Conversion of Gray Code into Binary Code)

दिए गए ग्रे कोड 101010 को बाइनरी में परिवर्तित करने के लिए निम्न का पालन करते है—

Binary Code = 110011

Gray code ⇒	1	0	1	0	1	0
	↓ ⊕	↓ ⊕	↓ ⊕	↓ ⊕	↓ ⊕	↓
Binary code ⇒	1	1	0	0	1	1

उदाहरण 1. दी गई संख्याओं का आउटपुट ज्ञात करें—

साधित उदाहरण (Solved Examples)

(i) $(DD\,CC)_{16} + (BBAA)_{16} = (\ )_{16}$
(ii) $(723)_8 + (237)_8 = (\ )_8$
(iii) $(10101100)_2 - (00111000)_2 = (\ )_2$

हल—

(i)

DDCC	⟶	1101	1101	1100	1100
BBAA	⟶	1011	1011	1010	1010
19976	1	1001	1001	0111	0110

Result → $(19976)_{16}$ **Ans.**

(ii)

723	⟶	111	010	011
237	⟶	010	011	111
1162	1	001	110	010

Result → $(1162)_8$ **Ans.**

(iii)


```
  10101100
- 00111000
----------
  01110100
```

Result → $(01110100)_2$ **Ans.**

उदाहरण 2. बाइनरी संख्या $(11000)_2$ तथा $(1101)_2$ का योग ज्ञात करें।

हल—

```
   11000
+   1101
--------
  100101
```

Ans.

उदाहरण 3. बाइनरी संख्या $(11100)_2$ तथा $(01111)_2$ का योग ज्ञात कीजिए।

हल—

```
   11100
   01111
--------
  101011
```

Ans.

उदाहरण 4. बाइनरी संख्या $(100110)_2$ तथा $(11101)_2$ का अन्तर ज्ञात कीजिए।

हल—

```
Borrow  11  1
     100110
+     11101
-----------
     001001
```

Ans.

उदाहरण 5. डेसीमल संख्या 214 को ऑक्टल में परिवर्तित कीजिए।

हल—

8	214	6	↑ LSD
8	26	2	
8	3	3	MSD
	0		

ऑक्टल तुल्यांक = $(326)_8$ **Ans.**

उदाहरण 6. डेसीमल संख्या 3509 को हेक्साडेसीमल में परिवर्तित कीजिए।

हल—

16	3509	5 → 5	↑ LSD
16	219	11 → B	
16	13	13 → D	MSD
	0		

Scanned by CamScanner

हेक्स तुल्यांक = $(DB\,5)_{16}$ **Ans.**

उदाहरण 7. डेसीमल संख्या 35 का ऑक्टल तुल्यांक ज्ञात कीजिए।

हल—

8	35	3	↑ LSD
8	4	4	
	0		MSD

ऑक्टल तुल्यांक = $(43)_8$ **Ans.**

उदाहरण 8. बाइनरी संख्या 1101.1 का डेसीमल तुल्यांक ज्ञात कीजिए।

हल—डेसीमल संख्या $(N) = 1\times 2^3 + 1\times 2^2 + 0\times 2^1 + 1\times 2^0 + 1\times 2^{-1}$

$N = 8+4+0+1+0.5$

$N = (13.5)_{10}$ **Ans.**

उदाहरण 9. $(475.25)_8$ को डेसीमल में परिवर्तित कीजिए।

हल— $(N) = 4\times 8^2 + 7\times 8^1 + 5\times 8^0 + 2\times 8^{-1} + 5\times 8^{-2}$

$N = 256+56+5+0.25+0.078125$

$N = (317.32813)_{10}$ **Ans.**

उदाहरण 10. $(9B2\,.\,1A)_H$ को डेसीमल में परिवर्तित कीजिए।

हल— $N = 9\times 16^2 + B(11)\times 16^1 + 2\times 16^0 + 1\times 16^{-1} + A(10)\times 16^{-2}$

$N = 2304+176+2+0.0625+0.039$

$N = (2482.1)_{10}$ **Ans.**

उदाहरण 11. दिए गए बाइनरी नम्बर को डेसीमल, हेक्साडेसीमल तथा ऑक्टल में परिवर्तित कीजिए—

(i) $(101101)_2$

(ii) $(11011011)_2$

हल—(i) डेसीमल कनवर्जन

$= 1\times 2^5 + 0\times 2^4 + 1\times 2^3 + 1\times 2^2 + 0\times 2^1 + 1\times 2^0$

$= 32+8+4+1 = (45)_{10}$ **Ans.**

हेक्साडेसीमल कनवर्जन

ऑक्टल कनवर्जन =

$= \underbrace{10}\ \ \underbrace{1101}$

$\underbrace{0010}_{2}\underbrace{1101}_{D} = (2D)_H$

$\underbrace{101}_{5}\ \underbrace{101}_{5} = (55)_8$ **Ans.**

(ii) डेसीमल कनवर्जन $= 1\times 2^7 + 1\times 2^6 + 0\times 2^5 + 1\times 2^4 + 1\times 2^3 + 0\times 2^2 + 1\times 2^1 + 1\times 2^0$

$= 219$ **Ans.**

हेक्सा कनवर्जन = $\underbrace{1101}_{D}\underbrace{1011}_{B} = (DB)_{16}$ **Ans.**

ऑक्टल कनवर्जन = $\underbrace{011}_{3}\ \underbrace{011}_{3}\ \underbrace{011}_{3} = (333)_8$ **Ans.**

उदाहरण 12. हेक्स संख्या $(2AC5)_H$ को डेसीमल, ऑक्टल तथा बाइनरी में परिवर्तित करना।

हल—हेक्स से डेसीमल $(2AC5)_{16} = 2\times 16^3 + 10\times 16^2 + 12\times 16^1 + 5\times 16^0$

$= (10949)_{10}$ **Ans.**

हेक्स से बाइनरी $(2AC5)_{16} = (0010\ 1010\ 1100\ 0101)_2$ **Ans.**

हेक्स से ऑक्टल—हेक्स से ऑक्टल के लिए सर्वप्रथम हेक्स संख्या को बाइनरी में परिवर्तित कर प्राप्त बाइनरी संख्या से ऑक्टल संख्या प्राप्त करते हैं।

$\underbrace{000}_{0}\ \underbrace{010}_{2}\ \underbrace{101}_{5}\ \underbrace{011}_{3}\ \underbrace{000}_{0}\ \underbrace{101}_{5}$

$(2AC5)_{16} = (025305)_8$

$(2AC5)_{16} = (25305)_8$

उदाहरण 13. दी गई डेसीमल संख्याओं को BCD में परिवर्तित कीजिए—

(a) 35 (b) 174 (c) 2479

हल—

(a)	Decimal	3	5		
		↓	↓		
	BCD	0011	0101		**Ans.**
(b)	Decimal	1	7	4	
		↓	↓	↓	
	BCD	0001	0111	0100	**Ans.**
(c)	Decimal	2	4	7	9
		↓	↓	↓	↓
	BCD	0010	0100	0111	1001 **Ans.**

उदाहरण 14. निम्न को BCD नियमानुसार जोड़िए—

(a) $(57)_{10}$ तथा $(26)_{10}$

(b) $(83)_{10}$ तथा $(34)_{10}$

(c) $(569)_{10}$ तथा $(687)_{10}$

Scanned by CamScanner

```
Decimal        BCD
  57          0101 0111
+ 26          0010 0110
Carry →            [1][1]
Sum → 83   [0] 0111 (1101) ← Invalid BCD and Carry = 0
         Final      └→ Valid BCD
         carry
```

हल—(a)

```
        0111   1101    Incorrect sum
      + 0000   0110    We add 6 for correction
Carry   1111   1
        1000   0011    Correct result
          8      3
```

We have to add 6 to the sum to correct it.

Therefore, $(57)_{10} + (26)_{10} = (83)_{10}$ Ans.

(b)

```
Decimal                  BCD
(83)10               1000 0011
+(34)10             +0011 0100
             Carry
(117)10      Sum :   1011  0111     Sum > 9, Carry = 0
                  Invalid  Valid    Hence, correction is required
                    BCD     BCD
```

Thus, we add 6 to the invalid BCD, for correction.

```
          1011   0111
  +       0110   0000    6 is added only to the invalid BCD.
Carry
  1       0001   0111
  ↓        ↓      ↓
0001      0001   0111    ← Correct BCD result.
  1         1      7
```

Therefore, $(83)_{10} + (34)_{10} = (117)_{10}$ Ans.

(c)

```
  569    0101     0110     1001
+ 687    0110     1000     0111
            0        1      111
 1256    1011     1111↖   10000      Incorrect answer
       Invalid  Invalid  Valid BCD
         BCD      BCD      with
                          carry 1
```


We add $(0110)_2$ to only the invalid BCD numbers to get correct answer.

```
        1011      1111      0000      Incorrect sum
 +      0110      0110      0110
        1111 ↖    11
 [1]    0010    ↖ 10100     0110
  ↓
0001    0010      0101      0110      Correct BCD answer
  1       2         5         6
```

Therefore, $(569)_{10} + (687)_{10} = (1256)_{10}$ Ans.

उदाहरण 15. निम्न का जोड़ Excess-3 कोड में ज्ञात कीजिए—

(a) 5 तथा 4

(b) 16 तथा 29

हल—(a)

```
    5       1 0 0 0     Excess-3 for 5
 +  4     + 0 1 1 1     Excess-3 for 4
    9       1 1 1 1     No carry
            0 0 1 1     subtract 3
            1 1 0 0     Excess-3 for 9
```

(b)

```
   16      0 1 0 0      1 0 0 1     Excess-3 for 16
 + 29    + 0 1 0 1      1 1 0 0     Excess-3 for 29
   45      1 0 0 1   1  0 1 0 1
                 1 ←┘               Propagate carry
           1 0 1 0      0 1 0 1
                     +  0 0 1 1     Add 3 to correct 0101
           1 0 1 0      1 0 0 0
         - 0 0 1 1                  Subtract 3 to correct 1010
```

उदाहरण 16. परिवर्तित कीजिए—

1. $(10111011)_2 = (?)_{16} = (?)_8$

2. $(01010101)_2 = (?)_{Gray} = (?)_{10}$ *(UPBTE 2016)*

हल—1. (a) $(10111011)_2 = (?)_{16} = (?)_8$

```
1011 1011
  B    B
```

$(10111011)_2 = (BB)_{16}$ Ans.

(b)
```
010 111 011
 2   7   3
```

Scanned by CamScanner

$(010111011)_2 = (273)_8$ **Ans.**

2. (a) $(01010101)_2 = (?)_{Gray}$

```
0 1 0 1 0 1 0 1
↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓
0 1 1 1 1 1 1 1
```

$(01010101)_2 = (01111111)_{Gray}$ **Ans.**

(b) $(01010101)_2 = (?)_{10}$

$\Rightarrow (0 \times 2^7) + (1 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0)$

$\Rightarrow 0 + 64 + 0 + 16 + 0 + 4 + 0 + 1$

$\Rightarrow 85$

$(01010101)_2 = (85)_{10}$ **Ans.**

प्रश्नावली (Exercise)

1. डिजिटल इलेक्ट्रॉनिक को संक्षेप में समझाइये।
2. एनेलॉग तथा डिजिटल सिग्नल से क्या समझते है? उनका तुलनात्मक चार्ट बनाइये। *(UPBTE 2013, 14, 16)*
3. एनेलॉग तथा डिजिटल सिस्टम की व्याख्या कीजिए।
4. डिजिटल सिस्टम के लाभ तथा उदहारण लिखिए।
5. एनलॉग तथा डिजिटल सिस्टम का तुलनात्मक चार्ट बनाइये।
6. विभिन्न नम्बर सिस्टम क्या है? संक्षेप में समझाइये।
7. डेसिमल नम्बर सिस्टम से क्या समझते है?
8. बाइनरी नम्बर सिस्टम की व्याख्या कीजिये।
9. दिए गए नम्बरों को बदलिये–

   (a) $(1101011)_2 = (?)_{16}$ (b) $(357)_8 = (?)_{10}$ (c) $(1359)_{10} = (?)_8$
10. निम्न हेक्साडेसिमल संख्याओं को बाइनरी में बदलिये–

    (a) $(A14)_{16}$ (b) $(5C8)_{16}$
11. निम्न डेसिमल संख्याओं को ऑक्टल में बदलिए–

    (a) $(247)_{10}$ (b) $(0.6874)_{10}$ (c) $(3287.5198)_{10}$
12. बाइनरी संख्या $(101.101)_2$ को डेसिमल में बदलिए–
13. दी हुई डेसिमल संख्या को बाइनरी में बदलिये—

    (a) 46 (b) 109 (c) 121 (d) 17 (e) 135
14. निम्न को हल कीजिए–

    (a) $(1011011 - 1101)_2 = (?)_{10} = (?)_{10} = (?)_8 = (?)_{16}$

    (b) $(3287)_{10} = (?)_8$ (c) $(200)_{10} = (?)_2$

15. $(153.7)_{10}$ और $(4BD)_{16}$ को बाइनरी में बदलिये। *(UPBTE 2012)*
16. $(11011)_2$ को डेसिमल तथा ऑक्टल में बदलिये। *(UPBTE 2012)*
17. $(138.4)_{10}$ और $(5CE)_{16}$ को बाइनरी में बदलिये। *(UPBTE 2011)*
18. निम्न को बदलिये– *(UPBTE 2010)*

    (a) $(247)_{10}$ से ऑक्टल (b) $(0.6875)_{10}$ से ऑक्टल

    (c) $(3287.5100098)_{10}$ से ऑक्टल (d) $(95.5)_{10}$ से हेक्साडेसिमल

    (e) $(675.625)_{10}$ से हेक्साडेसिमल
19. निम्नलिखित को बदलिये–

    (a) $(11011011)_2 = (?)_8$ (b) $(5697)_{10} = (?)_2$

    (c) $(479)_{10} = (?)_{16}$ (d) $(10111011)_2 = (?)_{10}$
20. बाइनरी संख्या को डेसिमल में परिवर्तित करने की विधि का वर्णन कीजिए।
21. डेसिमल संख्या को बाइनरी में परिवर्तित करने की विधि का वर्णन कीजिए।
22. ऑक्टल नम्बर सिस्टम को समझाइए।

बहुविकल्पीय प्रश्न

1. एक किलोबाइट में कितने बाइट होते हैं?

   (a) 1000 (b) 500 (c) 1024 (d) 2000
2. बाइनरी संख्या का बेस होता है—

   (a) 4 (b) 2 (c) 16 (d) 8
3. डेसिमल संख्या 214 का ऑक्टल तुल्यांक है—

   (a) 214 (b) 300 (c) 326 (d) 200
4. संख्या $(614)_7$ का डेसीमल तुल्यांक होगा—

   (a) 1000 (b) 614 (c) 305 (d) 250
5. एक बाइट में कितने बिट होते हैं?

   (a) 2 (b) 4 (c) 6 (d) 8
6. एक निबल में कितने बिट होते हैं?

   (a) 2 (b) 4 (c) 6 (d) 8
7. संख्या 1001 का 1S काम्प्लीमेन्ट होगा—

   (a) 1001 (b) 0110 (c) 1010 (d) 1000
8. 1111.01 + 0111.10 = ...

   (a) 11110.11 (b) 10110.11 (c) 1011.11 (d) 1011.10

उत्तर (Answers)

1. (c), 2. (b), 3. (c), 4. (c), 5. (d), 6. (b), 7. (b), 8. (b)

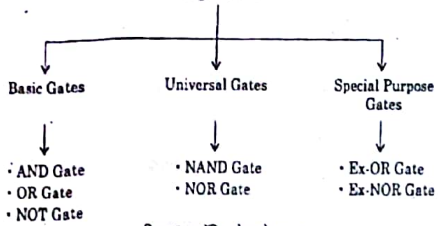
Scanned by CamScanner

अध्याय 2

लॉजिक गेट्स (Logic Gates)

2.1 परिचय (Introduction)

- लॉजिक गेट्स एक प्रकार का डिवाइस है जो डिजिटल सर्किट्स में प्रयोग किया जाता है।
- AND, OR, NOT बेसिक लॉजिक गेट्स हैं। इसके साथ ही NOR, NAND, EX-OR भी लॉजिक गेट्स है।



चित्र : 2.1 लॉजिक गेट्स के प्रकार

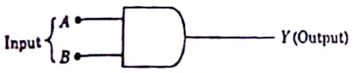
2.2 AND गेट (AND Gate)

- AND गेट एक प्रकार का लॉजिकल ऑपरेटर है जो गुणा को दर्शाता है। यह दो वैरियेबल (variable)के बीच मे डॉट (dot) लगाकर प्रदर्शित किया जाता है।

$$A \text{ AND } B = A \cdot B \rightarrow \text{Logical Multiplication}$$

- यदि लॉजिक सर्किट में A को B से गुणा करना है तो उसे $A.B$ या A AND B से दर्शाते है।
- AND गेट में दो या उससे अधिक इनपुट होते हैं तथा केवल एक आउटपुट होता है।

Symbol of AND Gate



चित्र : 2.2 AND गेट

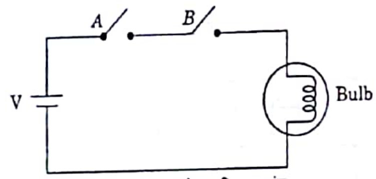
Boolean Equation

$$Y = AB$$

Truth Table

Input		Output
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

AND गेट का विद्युत तुल्यांक



चित्र : 2.3 AND गेट का विद्युत तुल्यांक

- सत्य तालिका से स्पष्ट है कि हमें आउटपुट तभी High मिलेगा जब सभी इनपुट High हों तथा आउटपुट तब Low होगा जब कम से कम एक इनपुट Low हो।

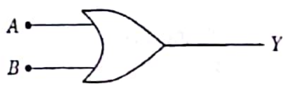
2.3 OR गेट (OR Gate)

- OR गेट एक प्रकार का लॉजिकल ऑपरेटर है जो जोड़ (addition) को दर्शाता है। यह दो वैरियेबल के बीच में "+" चिह्न लगाकर प्रदर्शित किया जाता है।

$$(A \text{ OR } B) = (A + B) \rightarrow \text{Logical Addition}$$

- यदि लॉजिक सर्किट में A को B से जोड़ना है तो उसे $(A + B)$ या $(A$ OR $B)$ से दर्शाते हैं।
- OR गेट में दो या दो से अधिक इनपुट होते हैं तथा केवल एक आउटपुट होता है।

Symbol of OR Gate



चित्र : 2.4 OR गेट

Boolean Equations

$$Y = A + B$$

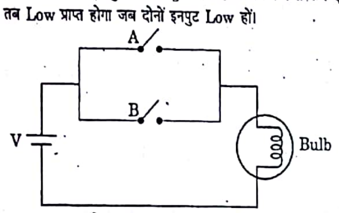
Scanned by CamScanner

Truth Table

Input		Output
A	*B*	*Y*
0	0	0
0	1	1
1	0	1
1	1	1

OR गेट का विद्युत तुल्यांक

- सत्य तालिका से स्पष्ट है कि हमें आउटपुट तभी High प्राप्त होगा जब कम से कम एक इनपुट या दोनों इनपुट High हों तथा आउटपुट तब Low प्राप्त होगा जब दोनों इनपुट Low हों।



चित्र : 2.5 OR गेट का विद्युत तुल्यांक

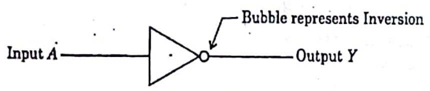
2.4 NOT गेट (NOT Gate)

- NOT गेट या इनवर्टर एक लॉजिक गेट है जिसमें एक इनपुट तथा एक आउटपुट होता है।
- NOT गेट के ऑपरेशन को बार (—) द्वारा दर्शाया जाता है तथा NOT गेट कॉम्पलीमेन्ट (complement) आउटपुट देता है।
- यदि कोई वैरियेबल *A* है तो इसके कॉम्पलीमेन्ट को $\overline{A}$ से दर्शाते हैं।

$$\overline{A} = \text{NOT } A$$

- NOT गेट को इनवर्टर (inverter) भी कहा जाता है।

Symbol of NOT Gate



चित्र : 2.6 NOT गेट

Boolean Equation

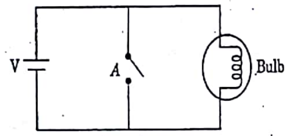
$$Y = \overline{A}$$

Truth Table

Input (A)	Output (Y)
0	1
1	0

NOT गेट का विद्युत तुल्यांक

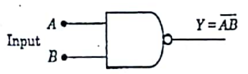
- NOT गेट के प्रतीक में बबल (bubble) का चिह्न इनवर्जन (inversion) तथा त्रिभुज का चिह्न एम्पलीफायर को प्रदर्शित करता है।



चित्र : 2.7 विद्युत तुल्यांक

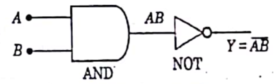
2.5 NAND गेट (NAND Gate)

- AND गेट को NOT के साथ कास्केड (cascade) करने पर हमें NAND गेट प्राप्त होता है अर्थात् जब एक AND गेट को NOT गेट के साथ कास्केड किया जाता है तो परिणामी गेट NAND गेट कहलाता है।
- इस प्रकार स्पष्ट है कि NAND गेट AND गेट तथा NOT गेट का कॉम्बिनेशन है।
- नीचे चित्र में दो इनपुट NAND गेट के संकेत तथा उसकी सत्य तालिका को दर्शाया गया है।



चित्र : 2.8 Two Input NAND Gate

≡



चित्र : 2.9 Equivalent Circuit

Truth Table

Input		Output
A	*B*	*Y*
0	0	1
0	1	1
1	0	1
1	1	0

- आउटपुट केवल तभी शून्य प्राप्त होगा जब सभी के सभी इनपुट "1" हों। अन्य सभी स्थितियों में आउटपुट सदैव High ("1") प्राप्त होगा।

Boolean Equations

- NAND गेट के लिए बूलियन इक्वेशन

Scanned by CamScanner

$$Y = \overline{A.B}$$

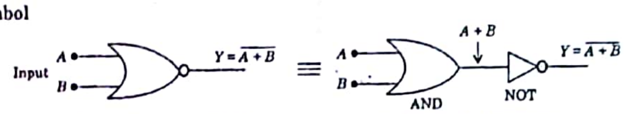
जहाँ $A.B$ AND गेट को दर्शाता है तथा बार (Bar) NOT गेट को दर्शाता है।

- NAND गेट को यूनिवर्सल गेट (universal gate) भी कहते है, क्योकि इसकी सहायता से सभी AND, OR व NOT गेटों को बनाया जा सकता है।

2.6 NOR गेट (NOR Gate)

- OR गेट को NOT गेट के साथ कास्केड करने पर हमे NOR गेट प्राप्त होता है अर्थात् जब एक OR गेट को NOT गेट के साथ कास्केड किया जाता है तो परिणामी गेट NOR गेट कहलाता है ।
- इस प्रकार स्पष्ट है की NOR गेट OR गेट तथा NOT गेट का कॉम्बिनेशन है।
- चित्र मे NOR गेट का संकेत तथा सत्य तालिका को दर्शाया गया है।

Symbol



चित्र : 2.10 Two Input NOR Gate

चित्र : 2.11 Equivalent Circuit

Truth Table

Input		Output
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

- सत्य तालिका से स्पष्ट है कि NOR गेट का आउटपुट तभी High होगा जब सभी इनपुट Low होगे। अन्य सभी स्थितियों मे आउटपुट सदैव Low होगे।

Boolean Equations

- NOR गेट के लिए बूलियन समीकरण

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

यहाँ $A+B$, OR गेट को दर्शाता है तथा बार (Bar) NOT गेट को दर्शाता है।

- NAND गेट की तरह NOR गेट भी यूनिवर्सल गेट है जिसकी सहायता से AND, OR तथा NOT गेटों को बनाया जाता है।

2.7 Ex-OR गेट (Ex-OR Gate)

- Exclusive-OR गेट को Ex-OR गेट या X-OR गेट कहते हैं।
- Ex-OR गेट मे दो या दो से अधिक इनपुट होते है तथा एक आउटपुट होता है।
- नीचे चित्र मे Ex-OR गेट के संकेत तथा सत्य तालिका को दर्शाया गया है।

Symbol

चित्र : 2.12 EX-OR गेट

Truth Table

Input		Output
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

Boolean Equations

$$Y = A \oplus B$$

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

$$Y = A \oplus B = A\overline{B} + \overline{A}B$$

- सत्य तालिका से स्पष्ट है कि Ex-OR गेट में जब दोनो इनपुट एकसमान होंगे तो आउटपुट सदैव Low प्राप्त होगा। अन्य सभी स्थितियो मे आउटपुट High प्राप्त होगा।
- Ex-OR गेट का उपयोग दो डिजिटल सिग्नल की तुलना करने के लिए किया जाता है।

2.8 यूनिवर्सल गेट (Universal Gate)

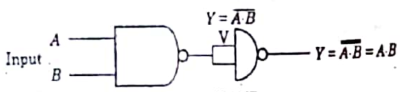
(UPBTE 2015)

- NAND तथा NOR गेट को यूनिवर्सल गेट कहते हैं, क्योंकि इनकी सहायता से AND, OR, NOT सभी प्रकार के गेटों का निर्माण किया जा सकता है।

2.8.1 NAND गेट की सहायता से विभिन्न गेटों का निर्माण

(a) NAND गेट से AND गेट

- AND गेट के लिए $Y = A.B$
- डबल इनवर्जन करने पर $Y = \overline{\overline{A.B}}$



AND gate using NAND

चित्र : 2.13

(b) NAND की सहायता से OR गेट

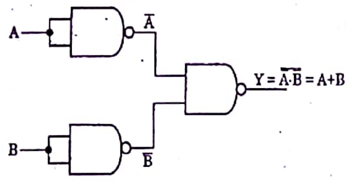
- OR गेट के लिए

$$Y = A + B$$

- डबल इनवर्जन RHS पर लिखने पर

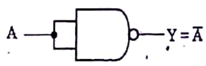
$$Y = \overline{\overline{A+B}} = \overline{\overline{A}.\overline{B}} = A + B$$

Scanned by CamScanner



OR gate using NAND gates
चित्र : 2.14

(c) NAND गेट की सहायता से NOT गेट



चित्र : 2.15

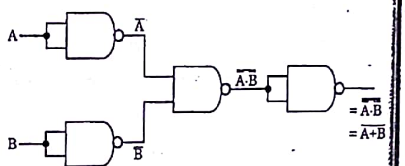
- इनपुट $A = B = A$
- आउटपुट $Y = \overline{A.B} = \overline{A.A}$

  $Y = \overline{A}$

(d) NAND गेट की सहायता से NOR गेट

- NOR गेट के लिए $Y = \overline{A+B} = \overline{A}.\overline{B}$
- RHS पर डबल इनवर्जन लेने पर

  $Y = \overline{\overline{\overline{A}.\overline{B}}} = \overline{A+B}$



NOR gate using only NAND gates
चित्र : 2.16

(e) NAND गेट की सहायता से Ex-OR

- Ex-OR गेट के लिए

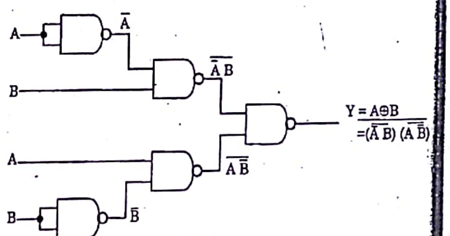
  $Y = A \oplus B$

  $= A\overline{B} + B\overline{A}$
- RHS पर डबल इनवर्जन लेने पर

  $Y = \overline{\overline{A\overline{B} + B\overline{A}}}$
- माना $X = AB$ तथा $Z = A\overline{B}$

  $Y = \overline{\overline{X + Z}}$
- डिमॉर्गन प्रमेय से $\overline{X+Z} = \overline{X}.\overline{Z}$

  $Y = \overline{\overline{X}.\overline{Z}} = \overline{(\overline{\overline{A}B}).(\overline{A\overline{B}})}$

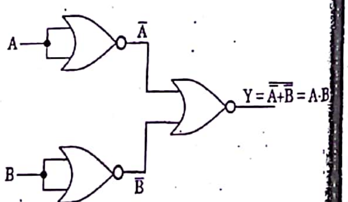


EX-OR using NAND gates
चित्र : 2.17

2.8.2 NOR गेट की सहायता से विभिन्न गेटों का निर्माण

(a) NOR गेट की सहायता से AND गेट

(UPBTE 2013)



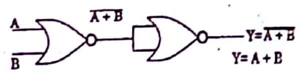
चित्र : 2.18 AND gate using only NOR gates

- AND गेट के लिए $Y = A.B$
- RHS पर डबल इनवर्जन लेने पर

  $Y = \overline{\overline{A.B}} = \overline{\overline{A}+\overline{B}}$

(b) NOR गेट की सहायता से OR गेट

(UPBTE 2013)



OR using only NOR gates
चित्र : 2.19

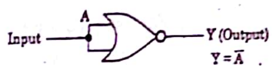
- OR गेट के लिए

  $Y = A+B$
- RHS पर डबल इनवर्जन लेने पर

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

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

(c) NOR गेट की सहायता से NOT गेट



NOT gate using NOR gates
चित्र : 2.20

- इनपुट $A = B = A$
- आउटपुट $Y = \overline{A+B} = \overline{A+A}$

  $Y = \overline{A}$

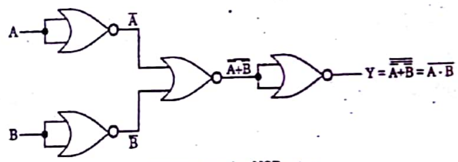
(d) NOR गेट की सहायता से NAND गेट

- NAND गेट के लिए

  $Y = \overline{A.B}$

  $= \overline{A} + \overline{B}$
- डबल इनवर्जन लेने पर

  $Y = \overline{\overline{\overline{A}+\overline{B}}} = \overline{A.B}$



NAND gate using NOR gates
चित्र : 2.21

(e) NOR गेट की सहायता से Ex-OR गेट

- Ex-OR गेट के लिए

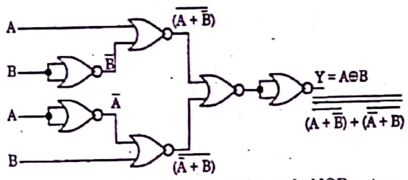
  $Y = \overline{A}B + A\overline{B}$
- माना $X = \overline{A}B,\ Z = A\overline{B}$

  $Y = X + Z$
- डबल इनवर्जन लेने पर

  $Y = \overline{\overline{X+Z}} = \overline{\overline{X}.\overline{Z}}$

  $= \overline{(\overline{\overline{A}B})\,(\overline{A\overline{B}})}$

  $= (\overline{A}B + A\overline{B})$



Ex–OR using only NOR gates
चित्र : 2.22

Scanned by CamScanner

2.9	बूलियन नियम (Boolean Law)

बूलियन प्रणाली के प्रमुख नियम हैं—

1. क्रम विनिमेय नियम (Commutative Law)
2. साहचर्य नियम (Associative Law)
3. वितरण नियम (Distributive Law)
4. AND नियम (AND Law)
5. OR नियम (OR Law)
6. इनवर्जन नियम (Inversion Law)

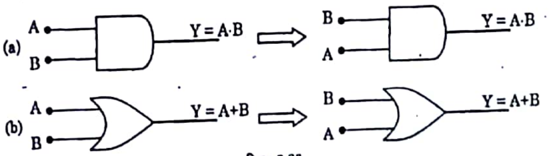
2.9.1 क्रम-विनिमेय नियम (Commutative Law)

- कोई बाइनरी ऑपरेशन जो दिए गए समीकरण को सेटिस्फाई (Satify) करता है, क्रमविनिमेय नियम कहलाता है।

$1 \rightarrow A.B = B.A$

$2 \rightarrow A + B = B + A$

- इस प्रकार हम कह सकते हैं कि क्रम-विनिमेय नियम में दिए गए वैरियेबल का क्रम बदलने पर लॉजिक गेट के आउटपुट पर कोई प्रभाव नहीं पड़ता है।



चित्र : 2.23

- इस प्रकार हम कह सकते है कि लॉजिकल चर राशियाँ किसी भी क्रम में हों, परिणाम सदैव समान होता है।

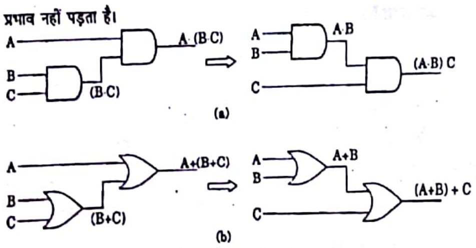
2.9.2 साहचर्य नियम (Associative Law)

- कोई बाइनरी ऑपरेशन जो दिए गए शून्य (Expression) को संतुष्ट करता है साहचर्य नियम कहलाता है।

$1 \rightarrow (A.B).C = A.(B.C)$

$2 \rightarrow (A + B) + C = A + (B + C)$

- इस प्रकार हम कह सकते हैं कि साहचर्य नियम में दिए गए वैरियेबल किसी भी प्रकार से संयोजित किए जाएँ आउटपुट पर कोई प्रभाव नहीं पड़ता है।



चित्र : 2.24 क्रम विनिमय

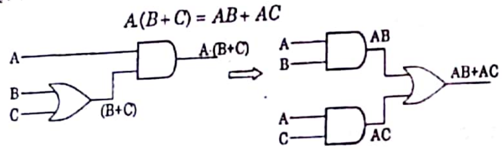
- इससे सिद्ध होता है कि लॉजिकल चर राशियाँ किसी भी प्रकार संयोजित हो, परिणाम सदैव समान होता है।

2.9.3 वितरण नियम (Distributive Law)

- सामान्य बीजगणित की भाँति लॉजिक व्यंजकों को किसी पद से गुणा कर बढ़ाया जाना वितरण का नियम कहलाता है।

उदाहरण –

$A(B + C) = AB + AC$



चित्र : 2.25

- इस प्रकार वितरण का नियम प्रयुक्त कर व्यंजकों को लॉजिक परिपथों में बदला जा सकता है।

2.9.4 AND नियम (AND Law)

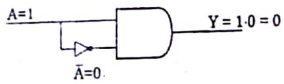
- AND निय्म में AND ऑपरेशन का प्रयोग किया जाता है, अत: इस नियम को AND नियम कहते हैं।
- बूलियन बीजगणित पर आधारित AND के कुछ प्रमुख नियम इस प्रकार है—

$1 \rightarrow A.0 = 0$

$2 \rightarrow A.1 = A$

$3 \rightarrow A.A = A$

$4 \rightarrow A.\overline{A} = 0$



चित्र : 2.26

- नियम $A.\overline{A} = 0$ प्रदर्शित करता है कि यदि इनपुट (*A*) लॉजिक 1 है तो $\overline{A}$ शून्य होगा जिससे आउटपुट शून्य होगा।
- इस प्रकार चित्र द्वारा इस नियम की पुष्टि होती है।

2.9.5 OR नियम (OR law)

- OR नियम OR ऑपरेशन का प्रयोग करता है। अत: इस नियम को OR नियम कहते हैं।
- बूलियन बीजगणित पर आधारित OR के कुछ प्रमुख नियम इस प्रकार हैं—

$1 \rightarrow A + 0 = A$

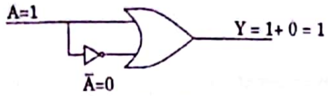
$2 \rightarrow A + 1 = 1$

$3 \rightarrow A + A = A$

$4 \rightarrow A + \overline{A} = 1$

- नियम $(A + \overline{A} = 1)$ प्रदर्शित करता है कि यदि इनपुट (*A*) लॉजिक 1 है तो $\overline{A}$ शून्य होगा तथा $A + \overline{A} = $ '1' प्राप्त होगा।

Scanned by CamScanner



चित्र : 2.27

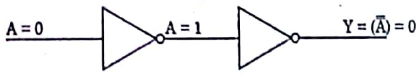
- इस प्रकार चित्र द्वारा इस नियम की पुष्टि होती है।

2.9.6 इन्वर्जन नियम (Inversion Law)

- Inversion Law "NOT" ऑपरेशन का प्रयोग करता है। नियम से आशय है कि यदि किसी वैरियेबल का डबल इन्वर्जन किया जाए तो हमें मूल वैरियेबल प्राप्त होगा।

$$\bar{\bar{A}} = A$$

- यदि $A = 0$ तो $\bar{A} = 1$ तथा $\bar{\bar{A}} = 0$, अतः आउटपुट $Y = A$.



चित्र : 2.28

- इस प्रकार चित्र द्वारा इस नियम की पुष्टि होती है।

2.9.7. बूलियन नियम का सारांश

S.No.	Name of Law	Statement
1.	Commutative law	$A \cdot B = B \cdot A$
		$A + B = B + A$
2.	Associative Law	$(A \cdot B) \cdot C = A \cdot (B \cdot C)$
		$(A + B) + C = A + (B + C)$
3.	Distributive Law	$A \cdot (B + C) = AB + AC$
4.	AND Laws	$A \cdot 0 = 0$
		$A \cdot 1 = A$
		$A \cdot A = A$
		$A \cdot \bar{A} = 0$
5.	OR Laws	$A + 0 = A$
		$A + 1 = 1$
		$A + A = A$
		$A + \bar{A} = 1$
6.	Inversion Law	$\bar{\bar{A}} = A$
7.	Other Important Laws	$A + BC = (A + B)(A + C)$
		$\bar{A} + AB = \bar{A} + B$
		$\bar{A} + A\bar{B} = \bar{A} + \bar{B}$
		$A + \bar{A}B = A$
		$A + \bar{A}B = A + B$

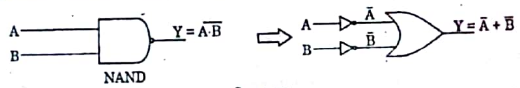
2.10 डि-मॉर्गन प्रमेय (De-Morgan's Theorem)

- डि-मॉर्गन द्वारा दो प्रमेय लागू किए गए थे जिनका प्रयोग बूलियन व्यंजकों को उनके तुल्यांक व्यंजकों में परिवर्तित करने में होता है।

प्रमेय-1—

$$\overline{AB} = \bar{A} + \bar{B} \qquad \text{NAND = Bubbled OR}$$

- इस प्रमेय के अनुसार किसी गुणनफल का कॉम्पलीमेन्ट, राशियों के अलग-अलग कॉम्पलीमेन्ट के योग के बराबर होता है।
- दिए गए समीकरण को चित्र में NAND तथा (NOT-OR) गेटों द्वारा प्रदर्शित किया गया है।

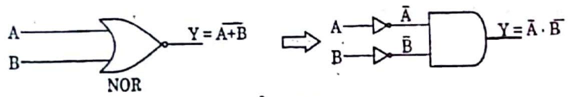


चित्र : 2.29

प्रमेय-2 —

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

- इस प्रमेय के अनुसार किसी योग (Sum) का कॉम्पलीमेन्ट, उनमें प्रयुक्त राशियों के अलग-अलग कॉम्पलीमेन्ट के गुणनफल के बराबर होता है।
- दिए गए समीकरण को चित्र में NOR तथा NOT-AND (Bubbled-AND) द्वारा प्रदर्शित किया गया है।



चित्र : 2.30

2.11 डुअलिटी प्रमेय (Duality Theorem)

- इस प्रमेय द्वारा किसी बूलियन सम्बन्ध को दूसरे बूलियन सम्बन्ध में परिवर्तित करना संभव है।
- डुअलिटी प्रमेय की निम्न विधियाँ हैं—

1. प्रत्येक AND ऑपरेशन को OR आपरेशन में बदलिये।
2. प्रत्येक OR ऑपरेशन को AND ऑपरेशन में बदलिये।
3. व्यंजक के 0s तथा 1s को कॉम्पलीमेन्ट कीजिए।

उदाहरण—दिए गए बाइनरी फंक्शन का डुअल (Dual) ज्ञात कीजिए।

Scanned by CamScanner

(a) $A + AB = A$

(b) $A + \overline{A}B = A + B$

(c) $A + \overline{A} = 1$

(d) $(A + B)(A + C) = A + BC$

हल—(a) सर्वप्रथम सभी (+) को (·) से तथा सभी (.) को (+) से रिप्लेस करने पर

$$A \cdot (A + B) = A$$

इसी प्रकार

S.No.	Given Expression	Dual of Expression
(b)	$A + \overline{A}B = A + B$	$A \cdot (\overline{A} + B) = A \cdot B$
(c)	$A + \overline{A} = 1$	$A \cdot \overline{A} = 0$
(d)	$(A + B)(A + C) = A + BC$	$AB + AC = A(B + C)$

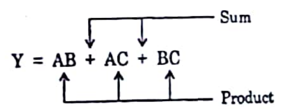
2.12 लॉजिक फंक्शन को सरल करने की विधि (Representation for Logical Functions)

- किसी लॉजिक फंक्शन को सरल करने की दो विधियाँ हैं—

1. Sum-of-Product (SOP)
2. Product-of-Sum (POS)

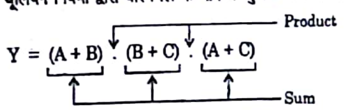
Sum-of-Product (SOP)

- SOP में व्यंजक को बूलियन नियमों द्वारा वैरियेबल के गुणनफल के योग के रूप में दर्शाया जाता है।



Product of Sum (POS)

- POS में व्यंजक को बूलियन नियमों द्वारा वैरियेबल के योग के गुणक के रूप में दर्शाते हैं।



2.12.1 स्टैन्डर्ड तथा नॉन-स्टैण्डर्ड SOP, POS फंक्शन (Standard and Non-standard SOP, POS Function)

S.No.	Logical Expression	Type of Expression
1.	$Y = AB + AB\overline{C} + \overline{A}BC$	Non-standard SOP
2.	$Y = AB + A\overline{B} + \overline{A}\overline{B}$	Standard SOP
3.	$Y = (\overline{A} + B)(A + B)(A + \overline{B})$	Standard POS
4.	$Y = (A + B)(A + B + C)$	Non-standard POS

उदाहरण—दिए गए व्यंजकों को उनके स्टैण्डर्ड SOP या POS फार्म में बदलें—

(a) $Y = AB + AC + BC$

(b) $Y = (A + B)(\overline{B} + C)$

(c) $Y = (A + B)(A + C)(B + \overline{C})$

हल—(a)

$$Y = AB + AC + BC$$

$$Y = AB(C + \overline{C}) + AC(B + \overline{B}) + BC(A + \overline{A})$$

$$= ABC + AB\overline{C} + ACB + AC\overline{B} + BCA + BC\overline{A}$$

$$= \underbrace{ABC + ACB + BCA} + AB\overline{C} + AC\overline{B} + BC\overline{A}$$

$$Y = ABC + AB\overline{C} + A\overline{B}C + \overline{A}BC \quad (\because A + A = A)$$ **Ans.**

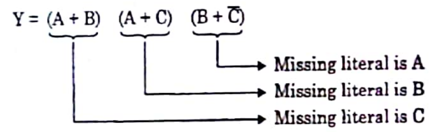
(b)

$$Y = (A + B)(\overline{B} + C) = (A + B + C\overline{C})(\overline{B} + C + A\overline{A})$$

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

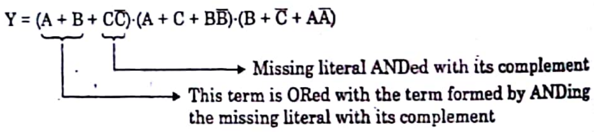
$$Y = (A + B + C)(A + B + \overline{C})(\overline{B} + C + A)(\overline{B} + C + \overline{A})$$ **Ans.**

(c) (i) First, let us find the missing literal for each term as under :



(ii) Secondly, we OR each term with (Missing literal or its complement).

Therefore, we have



(iii) Lastly, we simplify the expression to get standard POS as under :

$$Y = (A + B + C\overline{C})(A + C + B\overline{B})(B + \overline{C} + A\overline{A})$$

But $\quad A + BC = (A + B)(A + C)$

Therefore, $\quad Y = (A + B + C)(A + B + \overline{C})(A + C + B)(A + C + \overline{B})(B + \overline{C} + A)(B + \overline{C} + \overline{A})$

But $\quad A \cdot A = A$

Therefore, $(A + B + C)(A + C + B) = (A + B + C)$

Scanned by CamScanner

and $(A+B+\overline{C})(B+\overline{C}+A)=(A+B+\overline{C})$

$\therefore \quad Y = (A+B+C)(A+B+\overline{C})(A+\overline{B}+C)(\overline{A}+B+\overline{C})$

Each term consists of all the literals. Hence, this is the standard POS form.

2.12.2 Minterm तथा Maxterm

Minterm : स्टैण्डर्ड SOP फंक्शन का प्रत्येक इन्डिविजुअल टर्म (term) Minterm कहलाता है।

Standard SOP form $Y = ABC \quad A\overline{B}\overline{C} \quad \overline{A}BC$

Each individual term is called minterm

Standard POS form $Y = (A+B)\cdot(A+\overline{B})$

Each individual term is called maxterm

चित्र : 2.31 Concept of maxterm and minterm.

Maxterm : स्टैण्डर्ड POS फंक्शन का प्रत्येक इन्डिविजुअल टर्म (term) Maxterm कहलाता है।

Table - Minterms and Maxterms for three variables

Variables			Minterms	Maxterms
A	B	C	m_i	M_i
0	0	0	$\overline{ABC}=m_0$	$A+B+C=M_0$
0	0	1	$\overline{A}\overline{B}C=m_1$	$A+B+\overline{C}=M_1$
0	1	0	$\overline{A}B\overline{C}=m_2$	$A+\overline{B}+C=M_2$
0	1	1	$\overline{A}BC=m_3$	$A+\overline{B}+\overline{C}=M_3$
1	0	0	$A\overline{B}\overline{C}=m_4$	$\overline{A}+B+C=M_4$
1	0	1	$A\overline{B}C=m_5$	$\overline{A}+B+\overline{C}=M_5$
1	1	0	$AB\overline{C}=m_6$	$\overline{A}+\overline{B}+C=M_6$
1	1	1	$ABC=m_7$	$\overline{A}+B+C=M_7$

उदाहरण—दो वैरियबल के लिए Minterm तथा Maxterm लिखिए।

हल—दो वैरियेबल के लिए Minterm तथा Maxterm

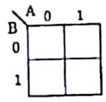
S.No.	Variables/literals		Minterms	Maxterms
	A	B	m_i	M_i
1.	0	0	$m_0=\overline{AB}$	$M_0=A+B$
2.	0	1	$m_1=\overline{A}B$	$M_1=A+\overline{B}$
3.	1	0	$m_2=A\overline{B}$	$M_2=\overline{A}+B$
4.	1	1	$m_3=AB$	$M_3=\overline{A}+B$

2.13 कारनॉफ मैप (Karnaugh Map)

- कारनॉफ मैप (K-map) बूलियन समीकरण के हल करने का एक ग्राफिकल मैथड है।
- K-map 2 वैरियेबल, 3 वैरियेबल, 4 वैरियेबल, 5 वैरियेबल तथा 6 वैरियेबल के होते हैं।
- n- वैरियेबल K-map के लिए 2^n ब्लॉक बनाए जाते है जिसकी सहायता से बूलियन फंक्शन को सरल किया जाता है।
- इस प्रकार 2 वैरियेबल के लिए 4 ब्लॉक, 3 वैरियेबल के लिए 8 ब्लॉक तथा 4 वैरियेबल के लिए 16 ब्लॉक K-map विधि से सरल करने के लिए बनाए जाते हैं।

2.13.1 2-वैरियेबल के लिए K-map

माना दो वेरियेबल A तथा B है जिनके लिए कोष्ठकों की संख्या $2^2 = 4$ होगी।

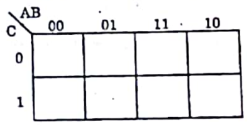


चित्र : 2.32 Structure of 3-variable K-map

- दो वैरियेबल के लिए कोष्ठकों को ऊपर चित्र में दर्शाया गया है।

2.13.2 3-वैरियेबल K-map

- माना A, B तथा C तीन वैरियेबल है जिनके लिए कोष्ठकों की संख्या $2^2 = 8$ होगी।

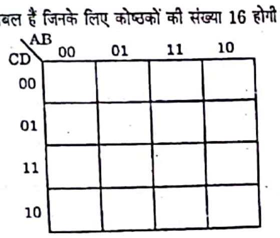


चित्र : 2.33 Structure of 3-variable K-map

- A, B तथा C की स्थिति को बदला जा सकता है।

2.13.3 4- वैरियेबल K-map

- A, B, C तथा D चार वैरियेबल हैं जिनके लिए कोष्ठकों की संख्या 16 होगी



चित्र : 2.34 Structure of 4-variable K-map

- K-map द्वारा बूलियन फंक्शन को सरल करने के लिए K-map में "युग्म", क्वाड तथा अष्टक बनाए जाते हैं।

Scanned by CamScanner

(a) युग्म (Pairs)

- एक युग्म, एक वेरियेबल तथा उसके पूरक (complement) को लुप्त करता है।
- दिए गए K-map में युग्म को दर्शाया गया है।

A \ BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
$\bar{A}$	0	0	1	1
A	0	0	0	0

(pair: the 1s in row $\bar{A}$)

- इस युग्म द्वारा बूलियन फंक्शन को लिखा जा सकता है—

$$Y = \bar{A}B$$

(b) क्वाड (Quad)

- क्वाड (quad) दो वेरियेबल को तथा उनके पूरक को लुप्त (eleminate) करता है—

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

(Quad: row 10)

- दिए गए K-map में Quad को दर्शाया गया है जिसके लिए बूलियन फंक्शन को निम्न प्रकार से लिखा जा सकता है—

$$Y = A\bar{B}$$

(c) अष्टक (Octet)

- अष्टक (octet) तीन चर राशियों तथा उनके पूरक को लुप्त (eleminate) करता है।

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

(Octet: rows 00 and 01)

- दिए गए K-map में अष्टक को दर्शाया गया है जिसके लिए बूलियन फंक्शन को निम्न प्रकार से लिखा जा सकता है—

$$Y = \bar{A}$$

2.14 K-मैप द्वारा बूलियन फंक्शन का सरलीकरण (Simplification of Boolean Function Using K-Map)

- चित्र मे तीन चर राशियों के SOP व्यंजक के लिए K-map प्रदर्शित किया गया है।
- K-map चित्र में 1's आउटपुट के लिए तीन युग्म P, Q तथा R बनाये जा सकते हैं।

C \ AB	$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$
$\bar{C}$	0	1	1	1
C	0	0	1	0

(P, Q, R: groups of 1s)

- युग्म P का आउटपुट

$$P = AB\bar{C} + A\bar{B}\bar{C}$$
$$= A\bar{C}(B + \bar{B})$$
$$= A\bar{C} \qquad \{B + \bar{B} = 1\}$$

- युग्म Q का आउटपुट

$$Q = \bar{A}B\bar{C} + AB\bar{C}$$
$$= B\bar{C}(\bar{A} + A)$$
$$= B\bar{C} \qquad \{A + \bar{A} = 1\}$$

- युग्म R का आउटपुट

$$R = AB\bar{C} + ABC$$
$$= AB(\bar{C} + C)$$
$$= AB$$

- अतः बूलियन फंक्शन को निम्न प्रकार सरल रूप में लिखा जा सकता है—

$$Y = P + Q + R$$
$$Y = A\bar{C} + B\bar{C} + AB$$

- इस प्रकार प्राप्त व्यंजक को लॉजिक गेट द्वारा भी प्रदर्शित किया जा सकता है।

उदाहरण 1—*दिए गए बूलियन फंक्शन को K-map द्वारा सरल कीजिए।*

$$Y = A\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC$$

हल :

A \ BC	$\bar{B}\bar{C}$	$\bar{B}C$	BC	$B\bar{C}$
$\bar{A}$			1	1
A		1		

(P: pair in row $\bar{A}$; Q: single 1 in row A)

Scanned by CamScanner

- दिया गया फंक्शन SOP फॉर्म में है, अतः प्रत्येक पद स्टैण्डर्ड प्रोडक्ट "1" उत्पन्न करेगा ।
- P के लिए आउटपुट = $\overline{A}BC+\overline{A}B\overline{C}=\overline{A}B$
- Q के लिए आउटपुट = $A\overline{B}C$

$$Y=\overline{A}B+A\overline{B}C$$

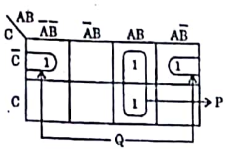
- प्राप्त आउटपुट दिए गए फंक्शन का सरलतम रूप है।

उदाहरण 2—*दिए गए बूलियन फंक्शन को सरल कीजिए ।*

$$Y=\overline{A}\overline{B}\overline{C}+AB\overline{C}+ABC+A\overline{B}\overline{C}$$

हल : फंक्शन को K-मैप में व्यवस्थित करने पर

- P के लिए आउटपुट = $AB\overline{C}+ABC=AB$
- Q के लिए आउटपुट = $\overline{A}\overline{B}\overline{C}+A\overline{B}\overline{C}=\overline{B}\overline{C}$



$$Y=AB+\overline{B}\overline{C}$$

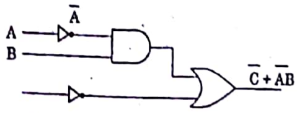
- प्राप्त आउटपुट फंक्शन का सरलतम रूप है।

उदाहरण 3—बूलियन नियमों द्वारा सिद्ध कीजिए—

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

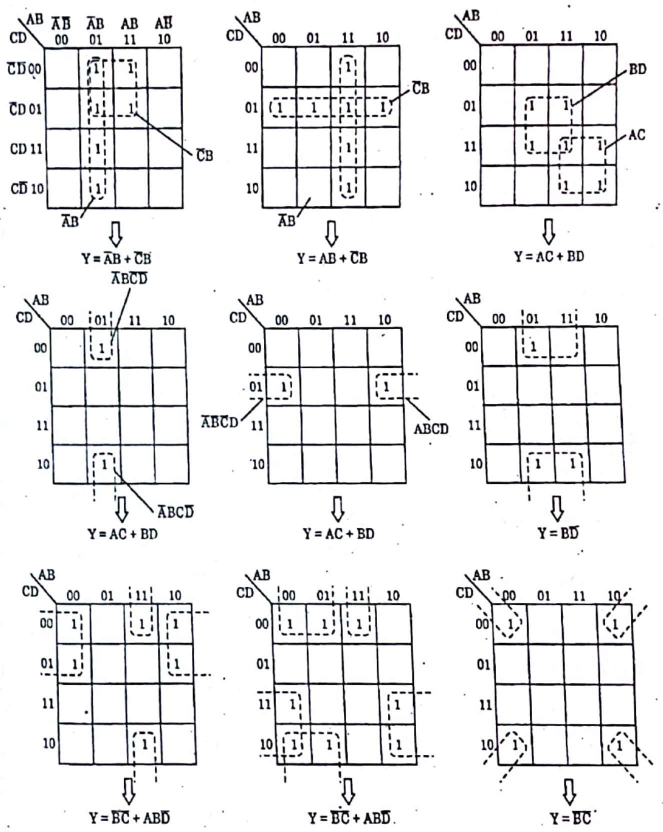
हल :

$$\overline{A}B+B\overline{C}+\overline{B}\overline{C}$$
$$=\overline{A}B+\overline{C}(B+\overline{B})$$
$$=\overline{A}B+\overline{C} \qquad [\because B+\overline{B}=1]$$
$$=\overline{C}+\overline{A}B$$



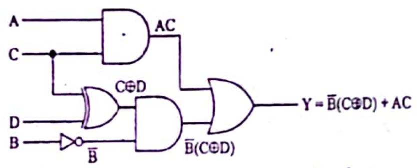
- प्राप्त फंक्शन का लॉजिक डायग्राम चित्र में दर्शाया गया है।

2.14.1 दिए गए K-map का आउटपुट लिखने का तरीका



चित्र : 2.35 Some additional K-map grouping possibilities

Scanned by CamScanner



चित्र : 2.45 (b) : Realization with minimum number of gates

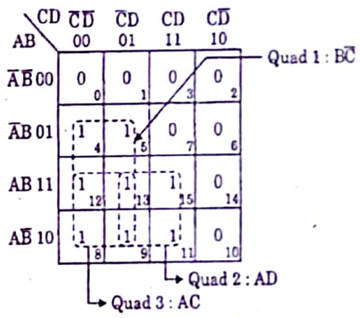
Minimized expression : $Y = \overline{B}\,\overline{C}\,D + \overline{B}\,C\,\overline{D} + AC = \underbrace{B(\overline{C}\,D + C\,\overline{D})}_{\text{Ex-OR gate}}\,AC$

Therefore, $Y = \overline{B}\,(C \oplus D) + AC$

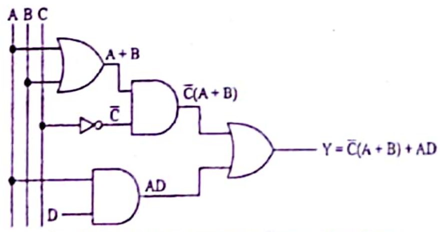
उदाहरण 10. K-map द्वारा दिए गए फंक्शन को सरल कीजिए।

$Y = \Sigma m\,(4, 5, 8, 9, 11, 12, 13, 15)$

हल—



चित्र : 2.46 (a) : K-map simplification



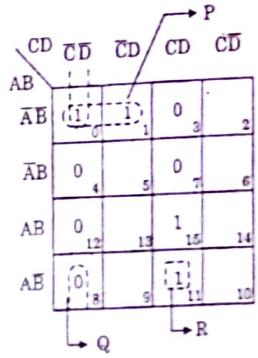
चित्र : 2.46 (b) : Realization using minimum number of gates

Minimized expression : $Y = B\overline{C} + AD + A\overline{C} = \overline{C}\,(A + B) + AD$

उदाहरण 11. सरलीकृत कीजिए— *(UPBTE 2016)*

$f\,(A, B, C, D) = \Sigma m\,(0, 1, 8, 11)$

हल—फंक्शन को K-Map में व्यवस्थित करने पर



P के लिए आउटपुट = $\overline{A}\,\overline{B}\,\overline{C}$

Q के लिए आउटपुट = $\overline{B}\,\overline{C}\,\overline{D}$

R के लिए आउटपुट $A\overline{B}\,C\,D$

$Y = \overline{A}\,\overline{B}\,\overline{C} + \overline{B}\,\overline{C}\,\overline{D} + A\overline{B}\,C\,D$

प्राप्त आउटपुट फंक्शन का सरलतम रूप है।

प्रश्नावली (Exercise)

1. लॉजिक गेट को समझाते हुए उनका वर्गीकरण कीजिए।
2. AND गेट की सत्य तालिका को बनाते हुए व्याख्या कीजिए।
3. यूनिवर्सल गेट से क्या तात्पर्य है? ये कितने प्रकार के होते है? *(UPBTE 2015)*
4. NAND तथा NOR गेट को सहायता से AND, OR तथा NOT गेट को समझाइये।
5. Ex-OR गेट तथा OR गेट मे क्या अन्तर है? Ex-OR गेट की सत्य तालिका बनाइये।
6. NOT गेट को समझाते हुए सत्य तालिका बनाइये।
7. बूलियन के नियमों की व्याख्या कीजिए।
8. डि-मॉर्गन प्रमेय से क्या तात्पर्य है? व्याख्या कीजिए।
9. SOP तथा POS को समझाये।
10. K-मैप में विभिन्न वैरियेबल के ब्लॉकों की संख्या को लिखिए।
11. निम्न का अर्थ स्पष्ट कीजिए—

Scanned by CamScanner

(a) Universal गेट (b) Exclusive OR गेट

12. OR गेट की सत्य तालिका को बनाते हुए समझाइये।

13. NAND गेट की सहायता से Ex-OR गेट को समझाइये।

14. निम्न बूलियन फंक्शन्स को सिद्ध कीजिए—

(a) $(A+B)(A+B)(A+C) = AC$

(b) $\overline{AB}C + \overline{A}BC + A\overline{BC} + \overline{A}B\overline{C} = \overline{A}C$

(c) $ABC + A\overline{B}C + AB\overline{C} = A(B+C)$

15. निम्न बूलियन फलन को केवल यूनिवर्सल गेट द्वारा निर्मित करें— *(UPBTE 2012)*

$$Y = AB\overline{C} + \overline{A}B\overline{C} + \overline{A}BC$$

16. K-map की सहायता से निम्न बूलियन फलन को सिद्ध कीजिए— *(UPBTE 2011)*

$$Y = \overline{ABC} + \overline{A}B\overline{C} + A\overline{BC} + AB\overline{C}$$

17. निम्न गेट को Ex-OR गेट की मदद से सिद्ध करें— *(UPBTE 2010)*

$$Y = A \oplus B \oplus C \oplus D$$

18. यूनिवर्सल गेट क्या है? AND,OR एवं NOT गेट को सिर्फ NAND एवं सिर्फ NOR से बनाकर दिखाएँ। *(UPBTE 2010, 13)*

19. दिए गए गेटों के संकेत चिह्न तथा सत्य तालिका बनाइये— *(UPBTE 2009)*

AND, NOR तथा Ex-OR

20. K-map का उपयोग करते हुए दिए गए बूलियन फलन को बनाइये तथा गेटों के प्रयोग से दर्शाइये— *(UPBTE 2009)*

$$f = ABC + AB\overline{C} + A\overline{BC} + \overline{A}B\overline{C} + \overline{A}B\overline{C} + \overline{ABC}$$

बहुविकल्पीय प्रश्न

1. एक बाइट में कितने बिट होते हैं?
(a) 2 (b) 4 (c) 8 (d) 16

2. एक निबल में बिट्स की संख्या होती है—
(a) 2 (b) 4 (c) 6 (d) 8

3. ऑक्टल संख्या $(567)_8$ का डेसीमल तुल्यांक है—
(a) $(567)_{10}$ (b) $(887)_{10}$ (c) $(375)_{10}$ (d) 501

4. हेक्साडेसिमल संख्या प्रणाली का रैडिक्स है—
(a) 6 (b) 8 (c) 16 (d) 10

5. AND, OR, NOT आदि डिजिटल ऑपरेशन में प्रयुक्त किये जाते हैं—
(a) Switch (b) रेक्टीफायर्स (c) ऑसिलेटर्स (d) एम्पलीफायर्स

6. किस गेट की I/P Low होने पर O/P Low होती है—
(a) AND (b) NAND (c) NOR (d) OR

7. एक गेट की O/P उस अवस्था में High होती है जब कम से कम उसकी एक I/P High होती है यह गेट है—
(a) Ex–OR (b) AND (c) OR (d) NAND

8. किस गेट की सभी I/P केवल low होने पर उसकी O/P low होती है?
(a) Ex–OR (b) NOR (c) OR (d) AND

9. निम्न में से कौन-सा ऑपरेशन 'associative' नहीं है—
(a) NOR (b) AND (c) OR (d) Ex-OR

10. बूलियन व्यंजक $\overline{\overline{A} \cdot \overline{B} \cdot \overline{C}}$ तुल्य है—
(a) $\overline{A}+\overline{B}+\overline{C}$
(b) $A\overline{B}C$
(c) $A+B+C$
(d) $\overline{A \cdot B \cdot C}$

11. कम्यूनिकेशन प्रणाली में प्रयुक्त की जाने वाली विधि है—
(a) डिजिटल
(b) एनेलॉग
(c) (a) तथा (b) दोनों
(d) इनमें से कोई नहीं

12. पॉजिटिव लॉजिक में Digital signal का 'high voltage' लेवल है—
(a) 0
(b) 1
(c) 2
(d) इनमें से कोई नहीं

13. आक्टल संख्या का बेस होता है—
(a) 2 (b) 8 (c) 10 (d) 16

14. व्यंजक '$A+BC+AB$' का कॉम्प्लीमेन्ट है—
(a) $\overline{A}+\overline{BC}$
(b) $\overline{B}+\overline{AC}$
(c) $\overline{ABC}$
(d) इनमें से कोई नहीं

15. निम्न Identity प्रदर्शित करती है—
$\overline{A+B+C+\cdots+N} = \overline{A}\cdot\overline{B}\cdot\overline{C}\cdots\overline{N}$
(a) डि-मॉर्गन का प्रथम नियम
(b) डि-मॉर्गन का द्वितीय नियम
(c) OR ऑपरेशन
(d) इनमें से कोई नहीं

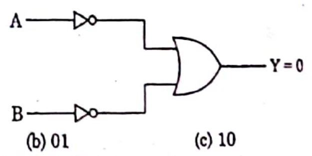
16. व्यंजक $\overline{ABC\cdots N} = \overline{A}+\overline{B}+\overline{C}+\cdots+\overline{N}$ प्रदर्शित करता है—
(a) डि-मॉर्गन का प्रथम नियम
(b) डि-मॉर्गन का द्वितीय नियम
(c) OR ऑपरेशन
(d) इनमें से कोई नहीं

17. बूलियन व्यंजक $(\overline{A}+B)(A+\overline{C})(\overline{B}+\overline{C})$ को सरल करने पर प्राप्त होगा—
(a) $(A+B)\cdot\overline{C}$
(b) $A+\overline{B})\overline{C}$
(c) $(\overline{A}+B)\overline{C}$
(d) इनमें से कोई नहीं

18. किस फ्लिप-फ्लॉप को लैच के समान प्रयुक्त किया जाता है?
(a) 1K फ्लिप-फ्लॉप
(b) D फ्लिप-फ्लॉप
(c) RS फ्लिप-फ्लॉप
(d) T फ्लिप-फ्लॉप

19. MOS परिवार के सदस्यों PMOS, NMOS, तथा CMOS में से किसमें न्यूनतम Power क्षय होती है?
(a) NMOS
(b) CMOS
(c) PMOS
(d) (a) व (b) दोनों

20. निम्न सर्किट मे O/P $Y = 0$ के लिए I/P होगी—



(a) 00 (b) 01 (c) 10 (d) 11

21. बूलियन व्यंजक $Y = (A+\overline{B}+\overline{A}B)\overline{C}$ को सरल करने पर प्राप्त होगा—

Scanned by CamScanner

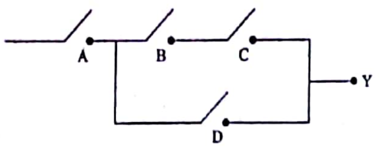
(a) $A\bar{C}$ (b) $B\bar{C}$

(c) $\bar{C}$ (d) इनमें से कोई नहीं।

22. डेसीमल संख्या 2 के लिए ग्रे कोड है—

(a) 0010 (b) 00011 (c) 1000 (d) 0101

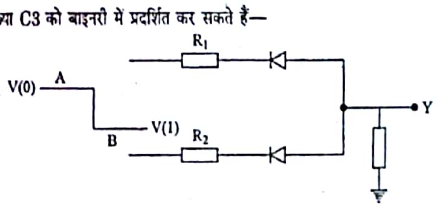
23. निम्न सर्किट में switch open होने पर शून्य (0) को तथा close होने पर 1 को प्रदर्शित करता है। सर्किट में Y के लिए व्यंजक होगा—



(a) $A+(B+C)D$ (b) $A+BC+D$

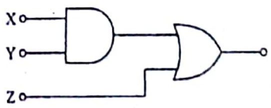
(c) $A+(BC+D)$ (d) इनमें से कोई नहीं।

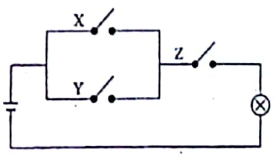
24. हेक्साडेसीमल संख्या C3 को बाइनरी में प्रदर्शित कर सकते हैं—



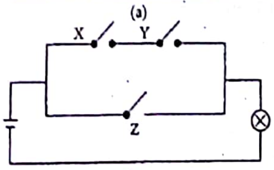
(a) 1111 (b)110011 (c) 111100 (d) 1100 0011

25. निम्न में से कौन सा वैद्युत सर्किट दिए गये सर्किट परिपथ के तुल्य है?

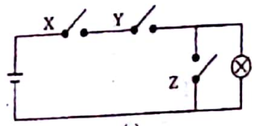




(a)



(b)



(c)

(d) इनमें से कोई नही।

उत्तर (Answers)

1. (c), 2. (b), 3. (c), 4. (c), 5. (a), 6. (a), 7. (c), 8. (c), 9. (a), 10. (c), 11. (c), 12. (b), 13. (b), 14. (b), 15. (b), 16. (a), 17. (b), 18. (c), 19. (c), 20. (b), 21. (d), 22. (a), 23. (b), 24. (c), 25. (d)

Scanned by CamScanner

अध्याय 3

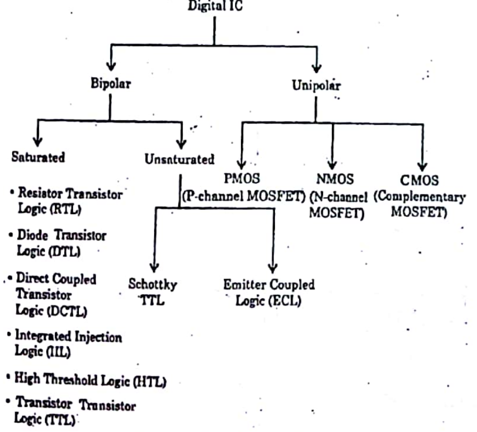
लॉजिक परिवार (Logic Families)

3.1 परिचय (Introduction)

- आजकल बाजार में विभिन्न प्रकार की IC उपलब्ध है। ये विभिन्न प्रकार की IC एक परिवार (family) के नाम से जानी जाती है।
- कम्पोनेन्ट एवं डिवाइसों की आन्तरिक जरूरतों को देखते हुए इस IC परिवार को दो भागों में बाँटा गया है—
  1. बाइपोलर (Bipolar), 2. यूनिपोलर (Unipolar)।
- डिजिटल ICs बाइपोलर अथवा यूनिपोलर तकनीकी पर फेब्रिकेट (fabricate) किये जाते हैं तथा उसी आधार पर उन्हें बाइपोलर लॉजिक परिवार (Bipolar Logic Family) अथवा यूनिपोलर लॉजिक परिवार (Unipolar Logic Family) कहा जाता है।

3.2 लॉजिक परिवार का वर्गीकरण (Classification of Logic Families)

- लॉजिक परिवार का वर्गीकरण नीचे दर्शाया गया है—



चित्र : 3.1 लॉजिक परिवार का वर्गीकरण

- बाइपोलर लॉजिक परिवार के मुख्य एलिमेन्ट प्रतिरोध, ट्रांजिस्टर तथा डायोड हैं।
- यूनिपोलर लॉजिक परिवार को MOS परिवार (MOS Family) भी कहा जाता है।

3.3 कैरेक्टरस्टिक्स ऑफ डिजिटल ICS (Characteristics of Digital ICs)

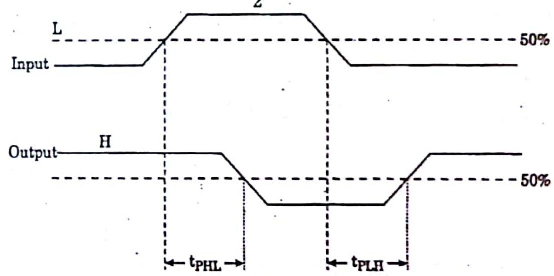
- डिजिटल ICs के प्रमुख पैरामीटरों को नीचे दर्शाया गया है जो इस प्रकार है—

1. Speed of Operation (Propagation Delay)
2. Power Dissipation
3. Fan-In
4. Fan-Out
5. Noise Immunity
6. Operating Temperature
7. Current and Voltage Parameter

3.3.1 स्पीड ऑफ ऑपरेशन (Speed of Operation)

- डिजिटल ICs की Speed of Operation उसके Propagation Delay के पदों में व्यक्त की जाती है।
- किसी लॉजिक के इनपुट में परिवर्तन होने के कितने समय बाद उसका आउटपुट परिवर्तित होता है, उस समय को उस लॉजिक का प्रोपेगेशन डिले (propagation delay) कहते हैं।
- प्रत्येक लॉजिक का HIGH से LOW अवस्था में आते समय प्रोपेगेशन डिले (t_{PHL}) तथा Low से HIGH अवस्था में आते समय प्रोपेगेशन डिले (t_{PLH}) का मान भिन्न-भिन्न होता है।
- प्रोपेगेशन डिले t_{PHL} तथा t_{PLH} के औसत को मध्यमान प्रोपेगेशन डिले (Average Propagation Delay) कहते हैं।

$$t_P = \frac{t_{PHL} + t_{PLH}}{2}$$



चित्र : 3.2

3.3.2 पावर-क्षय (Power Dissipation)

- जब लॉजिक गेट इनपुट के द्वारा ड्राइव किया जाता है तब लॉजिक गेट कुछ पावर कनज्यूम (Power consume) कर लेता है, जिसे power dissipation कहते हैं।

Scanned by CamScanner

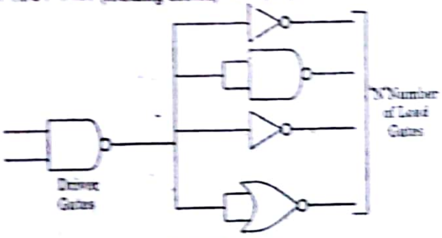
- यह IC में कम शक्ति को मिलीवाट (milliwatts) में मापा है।
- किसी लॉजिक युक्ति में पॉवर क्षय जितना कम होगा वह युक्ति उतनी ही अच्छी होगी।

3.3.3 फैन-इन (Fan-In)

- लॉजिक गेट के इनपुट की वह अधिकतम संख्या जो लॉजिक गेट को दी जा सकती है, Fan-In कहलाती है।
- इस परिभाषा के अनुसार एक 2-इनपुट गेट के लिए Fan-In दो है।

3.3.4 फैन-आउट (Fan-Out)

- गेटों की वह संख्या जिन्हें एक लॉजिक गेट ड्राइव कर सकता है, Fan-Out कहलाता है।
- फैन-आउट को लोडिंग फैक्टर (loading factor) भी कहते हैं।



चित्र : 3.3 Fan-Out

3.3.5 Noise Immunity

- Noise Immunity से तात्पर्य है किसी लॉजिक गेट को दी जाने वाली वह इनपुट वोल्टेज जो एक दी गई स्थायी वोल्टेज पर आउटपुट की स्टेट को बदलती है।
- Noise Immunity के मात्रात्मक मापन (Quantitative Measure) को Noise Margin कहते हैं।

3.3.6 Operating Temperature

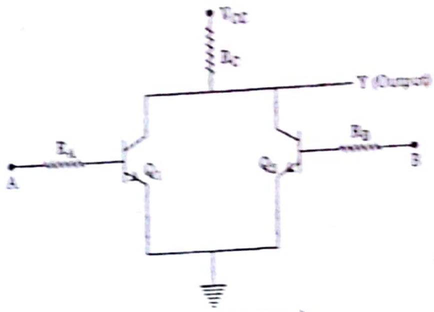
- ICs के ऑपरेटिंग टेम्परेचर का रेंज 0°C से +70°C तक इण्डस्ट्रियल एप्लीकेशन में तथा –55°C से +125°C तक मिलिट्री एप्लिकेशन में होता है।
- सभी लॉजिक गेटों को अधिकतम संचालित करने के लिए उनकी टेम्परेचर रेंज निर्धारित होती है।

3.3.7 Current and Volatge Parameter

- डिजिटल डिजाइन में करेन्ट और वोल्टेज पैरामीटर का मुख्य योगदान होता है।
- यदि आउटपुट करेन्ट आउटपुट टर्मिनल पर फ्लो हो रही हो तो उसे Sinking Current कहते हैं।
- डिजिटल सर्किट में वोल्टेज पैरामीटर को नहीं दर्शाते हैं। यहाँ वोल्टेज LOW अथवा HIGH के द्वारा दर्शाया जाता है।

3.4 प्रतिरोध ट्रांजिस्टर लॉजिक (RTL) (Resistor Transistor Logic)

- RTL से तात्पर्य है—प्रतिरोध ट्रांजिस्टर लॉजिक, जिससे पता चलता है कि RTL में प्रतिरोध तथा ट्रांजिस्टर का उपयोग होता है।



चित्र : 3.4 2-इनपुट RTL NOR गेट

चित्र : 3.5 NOR गेट का सिंबल

- RTL में NOR फंक्शन प्राप्त किये जाते हैं। इस प्रकार हम कह सकते हैं कि RTL मुख्य रूप से NOR गेट है।
- जब इनपुट $A = B = 0$, इस समय दोनों ट्रांजिस्टर ऑफ होंगे और R_C का भी करेन्ट शून्य होगा जिससे आउटपुट वोल्टेज V_{CC} के बराबर होगा अर्थात् $A = B = 0$ के लिए $Y = 1$ होगा।
- जब $A=0$ और $B=1$ तब ट्रांजिस्टर Q_1 ऑफ होगा और Q_2 सेचुरेशन की स्थिति में होगा जिससे आउटपुट वोल्टेज शून्य होगा अर्थात् $A = 0$ तथा $B = 1$ के लिए $Y = 0$ होगा।
- जब $A=1$ तथा $B=0$ तब ट्रांजिस्टर Q_1 सेचुरेशन की स्थिति में होगा तथा Q_2 ऑफ होगा जिससे आउटपुट वोल्टेज शून्य होगा अर्थात् $A=1$ तथा $B=0$ के लिए $Y=0$ होगा।
- जब इनपुट $A=B=1$ तब दोनों ट्रांजिस्टर सेचुरेशन की स्थिति में होंगे जिससे आउटपुट शून्य होगा अर्थात् $A = B = 1$ के लिए आउटपुट शून्य होगा।

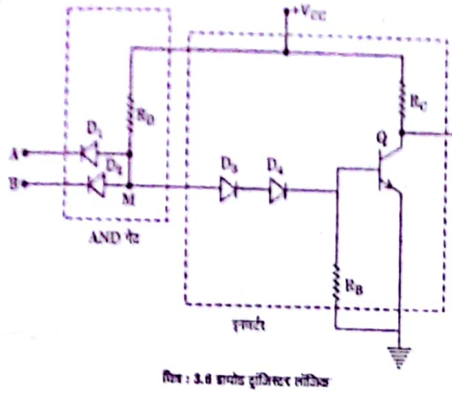
RTL के लिए पैरामीटर

1. फैन-इन 4
2. फैन-आउटपुट = 5
3. प्रोपेगेशन डिले =10 ns
4. TTL की अपेक्षाकृत कम स्पीड
5. पॉवर क्षय अधिकतम

3.5 डायोड ट्रांजिस्टर लॉजिक (DTL) (Diode Transistor Logic)

- DTL सर्किट NAND गेट की तरह कार्य करता है।
- DTL सर्किट RTL की तुलना में अधिक जटिल होते हैं।

Scanned by CamScanner



चित्र : 3.6 डायोड ट्रांजिस्टर लॉजिक

- DTL सर्किट के इनपुट स्टेज में डायोड D_1, D_2 तथा रजिस्टर R_D मिलकर AND गेट बनाते है। सर्किट का आगे का भाग इनवर्टर को दर्शाता है।
- जब A तथा B दोनों LOW होंगे तब डायोड D_1 तथा D_2 फॉरवर्ड बायस में होंगे तथा आउटपुट HIGH होगा अर्थात् $A=B=0$ के लिए आउटपुट Y=1 होगा।
- जब A या B में से कोई एक LOW होगा तब इस स्थिति में हमें आउटपुट HIGH प्राप्त होगा।

यदि

$$\left.\begin{array}{lll} A=0 & \text{तथा} & B=1 \\ & \text{या} & \\ A=1 & \text{तथा} & B=0 \end{array}\right\} Y=1$$

- जब A तथा B दोनों HIGH होंगे तब डायोड D_1 तथा D_2 रिवर्स बॉयस में होंगे तथा हमें LOW आउटपुट प्राप्त होगा अर्थात् $A=B=1$ के लिए आउटपुट $Y=0$ होगा।

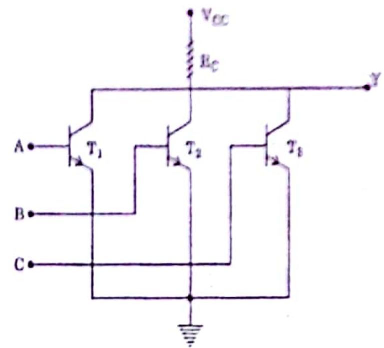
DTL के पैरामीटर

1. TTL की अपेक्षा कम स्पीड
2. प्रोपेगेशन डिले = 25 ns
3. फैन-इन = 5
4. फैन-आउट = 8
5. पॉवर क्षय अधिकतम = 11 mW

3.6 Direct Coupled Transistor Logic (DCTL)

- DCTL सर्किट RTL की तुलना में सरल होते हैं परन्तु Current Hogging के दोष के कारण DCTL का व्यापक प्रयोग नहीं किया जाता है।

- DCTL सर्किट NOR गेट की तरह कार्य करता है।



चित्र : 3.7 DCTL

- जब सभी इनपुट LOW होंगे इस स्थिति में ट्रांजिस्टर T_1, T_2 तथा T_3 किसी में कोई धारा नहीं प्रवाहित होगी। अतः आउटपुट (Y) HIGH होगा।
- इस प्रकार सभी LOW होने पर आउटपुट HIGH स्टेट में होगा।

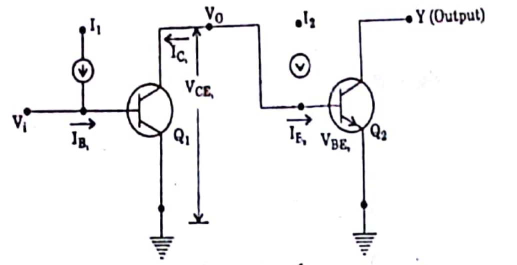
3.7 इन्टीग्रेटेड इन्जैक्शन लॉजिक (IIL) (Integrated Injection Logic)

- IIL (Integrated Injection Logic) DCTL के सिद्धान्त पर आधारित है।
- IIL का फेब्रिकेशन आसानी से हो जाता है। इसके लिए सिलिकॉन चिप का बहुत कम क्षेत्र (space) प्रयोग होता है तथा इसमें शक्ति व्यय बहुत कम होता है।
- Integrated Injection Logic LSI के लिए अकेला सेचुरेटेड बाइपोलर लॉजिक है।
- IIL का सिद्धान्त कम्पोनेन्टं को परस्पर मिलाना है अर्थात् एक अर्धचालक क्षेत्र दो या दो से अधिक युक्तियों से मिलकर बना होता है।
- इस प्रकार परस्पर मिलाने (merging) के कारण यह लॉजिक Merged Transistor Logic भी कहा जाता है।
- इस तकनीकी के प्रयोग से सिलिकॉन चिप पर काफी स्थान (space) की बचत होती है।

3.8 IIL इनवर्टर (IIL Inverter)

- IIL इनवर्टर को सर्किट चित्र में दर्शाया गया है।
- जब $V_i = 0$ होता है तब Q_1 ऑफ स्टेट में होता है तथा धारा IB_1 शून्य होती है।
- इस समय इनपुट सोर्स करेन्ट I_1 के लिए सिंक (sink) का कार्य करता है। अतः करेन्ट I_2, Q_2 के बेस में से प्रवाहित होगी जिससे ट्रांजिस्टर Q_2 सेचुरेशन में ड्राइव हो जाता है।

Scanned by CamScanner



चित्र : 3.8 IIL इन्वर्टर

- इस प्रकार जब Q_1 "OFF" होता है तब Q_2 "ON" होता है तथा आउटपुट LOW प्राप्त होती है।
- जब V_i HIGH होती है तब बेस धारा के दो घटक होते है—एक I_1 तथा दूसरा सोर्स V_i के कारण, इस प्रकार ट्रांजिस्टर Q_1 सेचुरेट हो जाता है तथा Q_2 कट-ऑफ हो जाता है।
- इस समय Q_2 धारा I_2 के लिए सिंक (sink) की भाँति कार्य करता है। इससे ज्ञात होता है कि V_0 का लॉजिक स्तर V_i का कॉम्प्लीमेन्ट है।

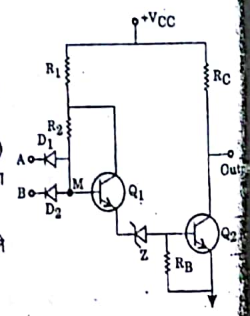
IIL के पैरामीटर

1. प्रोपेगेशन डिले चार्जिंग करेन्ट के व्युत्क्रमानुपाती होता है।
2. पॉवर क्षय चार्जिंग करेन्ट के समानुपाती होता है।
3. Figure of merit = 0.1 to 0.7 J
4. पैकिंग डेन्सिटी (packaging density) = 120 से 200 गेट/mm^2
5. सिलिकॉन के कम क्षेत्र (space) की आवश्यकता।

3.9 उच्च थ्रेशोल्ड लॉजिक (High Threshold Logic) (HTL)

(UPBTE 201

- HTL परिपथ NAND गेट की भाँति व्यवहार करता है।
- HTL परिवार उच्च सप्लाई वोल्टेज पर कार्य करता है।
- HTL परिवार में V_{th} का मान लगभग 7 V होता है जहाँ V_{th} Threshold Voltage को दर्शाता है।
- DTL सर्किट में डायोड की जगह पर जीनर डायोड (zener diode) तथा उच्च सप्लाई वोल्टेज प्रयोग में लाने पर HTL का निर्माण होता है।
- HTL में DTL के समान धारा प्राप्त करने के लिए अधिक मान वाले प्रतिरोध का उपयोग करते है।

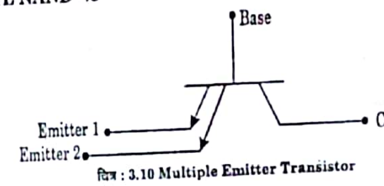


चित्र : 3.9 High Threshold Logic

3.10 ट्रांजिस्टर-ट्रांजिस्टर लॉजिक (TTL) (Transistor-Transistor Logic)

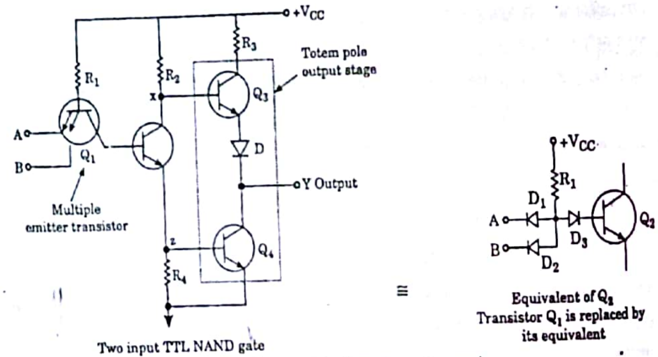
- DTL की सीमित गति (limited speed) होने के कारण DTL प्रयोग क्षेत्र से लगभग बाहर हो गया है तथा इसके स्थान पर TTL प्रयोग में लाया जाता है।
- पॉवर क्षय के अतिरिक्त TTL लॉजिक फैमिली में एक अच्छे लॉजिक के सभी गुण विद्यमान है।
- TTL सर्किट में इनपुट पर एक से अधिक ऐमीटर (emitter) वाले ट्रांजिस्टर का उपयोग किया जाता है।

3.10.1 2-इनपुट TTL NAND गेट



चित्र : 3.10 Multiple Emitter Transistor

- एक 2-इनपुट TTL NAND गेट नीचे चित्र में दर्शाया गया है जहाँ A तथा B इनपुट को प्रदर्शित करते है तथा Y आउटपुट को प्रदर्शित करता है।



चित्र : 3.11 दो इनपुट TTL NAND गेट तथा उसका तुल्यांक

- इस परिपथ में DTL परिपथो में प्रयोग किये जाने वाले इनपुट डायोड के स्थान पर मल्टी एमिटर प्रणाली (multiple emitter configuration) प्रयुक्त की गई है जिसके कारण उच्च स्विचिंग गति (high switching speed) प्राप्त होती है।

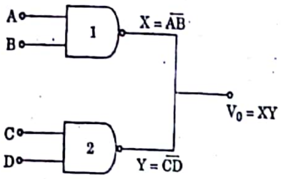
3.10.2 Totem-Pole Output Stage

- चित्र में आउटपुट की तरफ Q_3 तथा Q_4 से प्राप्त आउटपुट स्टेज को Totem-Pole Output Stage कहते हैं।
- Totem-Pole आउटपुट स्टेज में कोई एक ट्रांजिस्टर Q_3 अथवा Q_4 अवश्य "ON" रहता है। जब Q_3 ON होता है तब आउटपुट HIGH होती है तथा जब Q_4 "ON" होता है तब आउटपुट LOW होती है।

Scanned by CamScanner

3.10.3 Wired-AND TTL गेट

- Wired- AND प्रचालन में कुछ गेट्स की आउटपुट परस्पर संयोजित कर दी जाती है। इस संयोजन से AND प्रचालन प्राप्त होता है तथा एक अतिरिक्त लॉजिक फंक्शन उपलब्ध होता है।

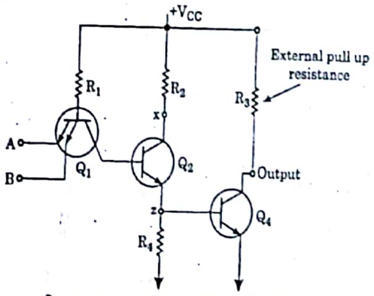


चित्र : 3.12 Wired-AND Operation

$$Vo = X.Y = (\overline{AB}), (\overline{CD})$$

3.10.4 Open Collector Output

- चित्र में Open Collector 2 इनपुट NAND गेट का परिपथ दर्शाया गया है।
- यह परिपथ TTL NAND गेट के समान है, अन्तर केवल इतना है कि इस परिपथ में से R_3 तथा Q_3को हटा दिया गया है।
- चित्र में ट्रांजिस्टर Q_4 के कलेक्टर टर्मिनल से आउटपुट प्राप्त होता है, अतः इसे Open Collector Output भी कहते हैं।



चित्र : 3.13 Open collector 2 input NAND gate

- प्रोपर ऑपरेशन के लिए यह आवश्यक है कि V_{cc} तथा Open Collector Output के बीच में बाह्य प्रतिरोध (external resistance) R_3 लगाया जाए। यह प्रतिरोध Pull up Resistance कहलाता है।
- Open Collector गेट्स का मुख्य गुण यह है कि उन्हें एक Common Pull up Resistance की सहायता से परस्पर Wired किया जा सकता है, अतः एक AND गेट की अवश्यकता नहीं होती।

3.10.5 Totem-Pole आउटपुट तथा Open Collector आउटपुट में तुलना

Comparison of Totem-Pole and Open Collector Outputs

Parameter	Totem-Pole	Open Collector
पॉवर क्षय	पुल अप (Pull up) ट्रांजिस्टर Q_3 के कारण निम्न	उच्च
स्पीड	उच्च	निम्न
पुल अप रजिस्टर (Pull up Resistor)	नहीं होता है	होता है
आउटपुट साइट में सर्किट कम्पोनेन्ट	पुल अप ट्रांजिस्टर (Q_3) तथा पुल डाउन टांजिस्टर (Q_4) होता है।	केवल पुल डाउन ट्रांजिस्टर (Q_4) होता है।

3.10.6 TTL Subfamilies (TTL सबफैमिली)

- TTL सीरीज का अध्ययन हम कर चुके है। TTL परिवार कुछ Sub-series में बँटी हुई है जिसे TTL Sub-families के नाम से जाना जाता है।
- TTL Sub-families के कुछ सदस्य इस प्रकार हैं—

1. HIGH Speed तथा LOW Power TTL
2. Schottky TTL
3. Advanced Schottky TTL
4. Advanced LOW Power Schottky TTL
5. Fast TTL

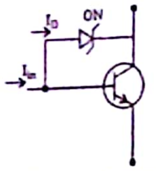
3.10.6.1 HIGH Speed तथा LOW Power TTL

- 74 Series के 74L00 Series LOW Power TTL को तथा 74H00 Series HIGH Power TTL को प्रदर्शित करते हैं।
- यहाँ "L" निम्न शक्ति व्यय को तथा "H" उच्च गति को प्रदर्शित करता है।
- TTL 74H00 Series की युक्तियाँ उच्च गति की होती हैं तथा TTL 74L00 Series की युक्तियों का आन्तरिक प्रतिरोध अधिक होता है जिससे शक्ति व्यय निम्न होता है।

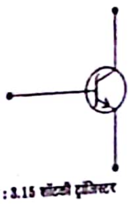
3.10.6.2 Schottky TTL (शॉटकी TTL)

- 74S00 Series शॉटकी TTL को प्रदर्शित करता है।
- TTL परिवार में Schottky TTL की गति अधिकतम होती है।
- इसका औसत प्रोपेगेशन डिले टाइम केवल 3 ns तथा औसत शक्ति व्यय 20 mW होता है।
- 3 ns प्रोपेगेशन डिले प्राप्त करने के लिए LOW Voltage Drop Dipole प्रयोग होता है जिसे Schottky Dipole कहते हैं।
- Schottky TTL में प्रोपेगेशन डिले घट जाता है लेकिन इसमें कम मान के प्रतिरोध को लगाने के कारण शक्ति व्यय बढ़ जाता है।

Scanned by CamScanner



चित्र : 3.14 ट्रांजिस्टर के साथ शॉटकी डायोड



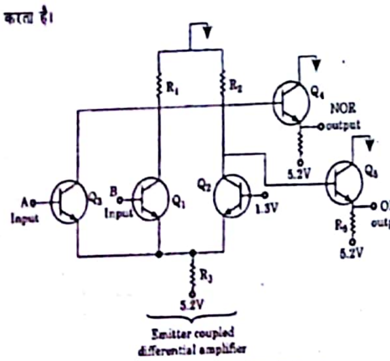
चित्र : 3.15 शॉटकी ट्रांजिस्टर

- ट्रांजिस्टर को संतृप्त अवस्था से रोकने के लिए शॉटकी ट्रांजिस्टर का प्रयोग किया जाता है। इस प्रकार के ट्रांजिस्टर निर्माण के लिए, ट्रांजिस्टर के बेस एवं कलेक्टर के मध्य एक शॉटकी डायोड को फैब्रिकेट किया जाता है। शॉटकी डायोड एवं ट्रांजिस्टर का फैब्रिकेशन शॉटकी ट्रांजिस्टर कहलाता है।
- शॉटकी TTL युक्तियों में प्रयुक्त सभी ट्रांजिस्टर अधिकतर शॉटकी ट्रांजिस्टर होते है।

3.11 एमिटर कपल्ड लॉजिक (ECL) (Emitter Coupled Logic)

(UPBTE 2015)

- ECL युक्तियों की स्पीड सबसे अधिक होती है, क्योंकि इसमें प्रयुक्त ट्रांजिस्टर न पूर्ण संतृप्त और न पूर्ण कट-ऑफ अवस्थाओं में प्रचलित रहते है।
- ECL का प्रोपेगेशन डिले लगभग 1 n sec प्रति गेट होता है। ECL मे पॉवर क्षय का मान लगभग 30 mW होता है।
- ट्रांजिस्टर एक्टिव रीजन (active region) में अधिक पावर कनज्यूम (consume) करता है। अतः ECL गेट अधिक पॉवर कनज्यूम करता है।



चित्र : 3.16 Two input ECL OR/NOR gate

- चित्र में 2-इनपुट ECL OR / NOR गेट को दर्शाया गया है।
- Q_1 तथा Q_2 ट्रांजिस्टर, Q_3 के एमिटर टर्मिनल एक साथ जोड़े गए है। अतः इसे एमिटर कपल्ड लॉजिक कहते है।
- ट्रांजिस्टर Q_1 तथा Q_2 मिलकर एमिटर कपल्ड डिफ्रेंशियल एम्प्लीफायर (emitter coupled differential amplifier) बनाते हैं।

- Q_1 तथा Q_3 परस्पर समान्तर में कनेक्ट किए गए है। Q_4 तथा Q_5 दो एमिटर फॉलोवर स्टेज है जिससे NOR तथा OR आउटपुट प्राप्त होता है।

ECL के गुण

1. सभी लॉजिक परिवार में अधिकतम गति।
2. प्रोपेगेशन डिले 1 n sec से भी कम।
3. फैन-आउटपुट (fan-out) अधिकतम लगभग 25 से अधिक।
4. एक साथ दो आउटपुट प्रदान करना।

ECL के दोष

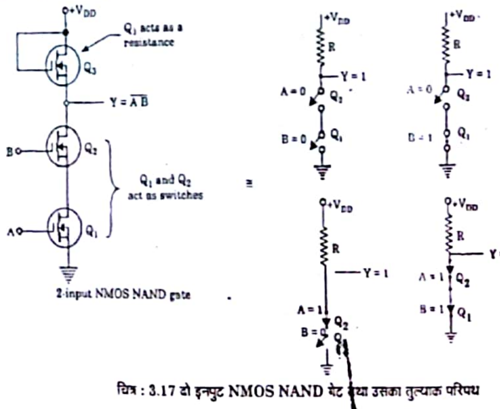
1. पावर क्षय अधिकतम लगभग 30 mW।
2. अधिक स्पीड होने के कारण वोल्टेज तथा करेन्ट ट्रांजिएन्ट उत्पन्न हो जाता है।

3.12 MOSFET सर्किट्स (MOSFET Circuits)

- MOSFET सर्किट्स को तीन भागों में विभाजित किया गया है—
- 1. PMOS2. NMOS3. CMOS
- MOS लॉजिक परिवार मे लोड (load) एवं स्विचों के लिए Enhancement MOSFET का प्रयोग होता है।
- NMOS लॉजिक के लिए *n*-channel Enhancement MOSFET का प्रयोग होता है तथा PMOS लॉजिक के लिए *p*-channel Enhancement MOSFET का प्रयोग होता है।
- CMOS लॉजिक के लिए *n*-channel तथा *p*-channel दोनों डिवाइसों का प्रयोग होता है।
- MOS सर्किट्स एम्प्लीफायर (amplifier) तथा लोड रजिस्टेंस (load resistance) दोनों की भाँति व्यवहार करता है।
- MOS सर्किट्स की मैनुफैक्चरिंग आसानी से हो जाती है तथा इसमें पावर क्षय कम होता है।

3.12.1 NMOS लॉजिक (NMOS Logic)

- NMOS लॉजिक मे N-Channel Enhancement MOSFET का प्रयोग होता है।
- NMOS डिवाइसों का इनपुट प्रतिरोध उच्च होता है।



चित्र : 3.17 दो इनपुट NMOS NAND गेट तथा उसका तुल्यांक परिपथ

Scanned by CamScanner

- NMOS का प्रयोग मेमोरी डिवाइसों, इनपुट/आउटपुट डिवाइसों तथा माइक्रोप्रोसेसर्स में होता है।
- चित्र 3.17 में 2-इनपुट NMOS NAND गेट का सर्किट दर्शाया गया है।
- MOSFET Q_1 तथा Q_2 एक स्विच की तरह कार्य करते हैं। Q_1 तथा Q_2 का गेट टर्मिनल NAND गेट के इनपुट A तथा B को प्रदर्शित करता है।
- MOSFET Q_3 एक रजिस्टेंस की भाँति व्यवहार करता है।

2- इनपुट NAND गेट ऑपरेशन

- Q_1 तथा Q_2 सीरीज में जोड़े गए स्विच की भाँति कार्य करते हैं।
- आउटपुट $Y = 1$ तब प्राप्त होगा जब कम से कम एक या दोनों इनपुट OFF होंगे।
- आउटपुट $Y = 0$ तब प्राप्त होगा जब दोनों इनपुट A तथा B HIGH होंगे।

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

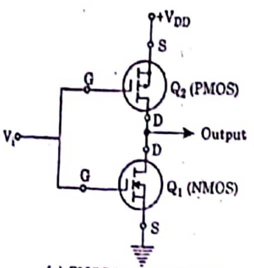
2 इनपुट NAND गेट का ऑपरेशन

3.12.2 CMOS लॉजिक

- CMOS से तात्पर्य है कॉम्पलीमेन्ट्री (complementary) MOSFET.
- CMOS लॉजिक में एक ही चिप पर p-टाइप तथा n-टाइप MOSFETs फेब्रिकेट किए जाते हैं।
- CMOS अन्य सभी लॉजिक परिवारों की तुलना में धीमा (slow) है परन्तु इसमें शक्ति व्यय लगभग शून्य होता है चाहे आउटपुट HIGH अथवा LOW किसी भी अवस्था में हो।

3.12.2.1 CMOS इनवर्टर

- नीचे चित्र में CMOS इनवर्टर को दर्शाया गया है।
- जब इनपुट V_i LOW होगा तब ट्रांजिस्टर Q_1 OFF होगा तथा Q_2, ON होगा जिससे आउटपुट HIGH प्राप्त होगी।



(a) CMOS inverter positive voltage

V_i	Q_1	Q_2	Output
0V (logic 0)	OFF	ON	V_{DD}(logic 1)
V_{DD}(logic 1)	ON	OFF	0V (logic 0)

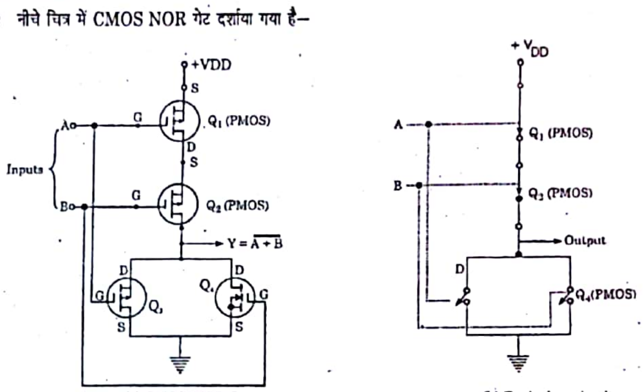
(b) Summary of operation

चित्र : 3.18 CMOS इनवर्टर तथा उसका ऑपरेशन

- जब V_i HIGH होगा तब ट्रांजिस्टर Q_1 ON होगा तथा Q_2 OFF होगा जिससे आउटपुट LOW प्राप्त होगा।

3.12.2.2 CMOS NOR गेट

- नीचे चित्र में CMOS NOR गेट दर्शाया गया है—



(a) CMOS, NOR Gate (b) Equivalent circuit

चित्र : 3.19 CMOS, NOR गेट तथा तुल्यांक परिपथ

- Q_1 तथा Q_2 p-channel MOSFET है जो सीरीज में कनेक्ट किए गए हैं तथा Q_3 और Q_4 n-channel MOSFET हैं जो समान्तर मे कनेक्ट किए गए है।
- इनपुट A को Q_1 तथा Q_3 के गेट से जोड़ा गया है तथा इनपुट B को Q_2 तथा Q_4 के गेट से जोड़ा गया है।
- आउटपुट $Y = 1$ तब प्राप्त होगा जब Q_1 तथा Q_2 ON हों तथा Q_3 और Q_4 OFF हों।
- आउटपुट $Y = 0$ तब प्राप्त होगा जब Q_1 तथा Q_2 OFF हों तथा Q_3 और Q_4 ON हों।

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

CMOS के लाभ

1. पॉवर क्षय निम्न
2. Fan-Out उच्च (लगभग 50)
3. अधिक सप्लाई वोल्टेज पर ऑपरेट किया जा सकता है।
4. Packaging Density उच्च होती है जिससे CMOS युक्तियाँ बहुत कम स्थान (space) लेती हैं।

Scanned by CamScanner

CMOS के दोष

1. प्रोपेगेशन डिले अधिक (लगभग 25 से 100 ns)
2. TTL से कम स्पीड

प्रश्नावली (Exercise)

1. निम्नलिखित में से किन्हीं दो तार्किक परिवारों के विवरण लिखिये— *(UPBTE 2009, 15)*
   (a) TTL
   (b) ECL
   (c) MOS Logic
2. बाइपोलर तथा यूनिपोलर से क्या तात्पर्य है?
3. बाइपोलर परिवार को सूचीबद्ध कीजिए।
4. डिजिटल IC की कैरेक्टरस्टिक्स को लिखिए।
5. निम्न पर टिप्पणी लिखिए—
   (a) Speed of Operation
   (b) Power dissipation
   (c) Fan In तथा Fan Out
   (d) Noise Immunity
6. RTL से क्या समझते हैं? विवेचना कीजिए।
7. DTL की व्याख्या कीजिए।
8. DCTL को समझाते हुए टिप्पणी लिखिए।
9. ITL को समझाते हुए IIL इनवर्टर की व्याख्या कीजिए।
10. HTL परिपथ को NAND गेट की भाँति व्यवहार करते हुए समझाइये। *(UPBTE 2015)*
11. TTL की व्याख्या कीजिए। Totem Pole Output स्टेज से क्या अभिप्राय है?
12. TTL सबफैमिली के विभिन्न सदस्यों को लिखिए।
13. ECL को समझाते हुए उसके गुण तथा दोषों की व्याख्या कीजिए।
14. MOSFET सर्किट से क्या अभिप्राय है? व्याख्या कीजिए।
15. CMOS तथा NMOS को समझाइये।

बहुविकल्पीय प्रश्न

1. निम्न में कौन सी बाइपोलर टेक्नालोजी डिजिटल IC में अधिकतम प्रयोग की जाती है?
   (a) DTL (b) TTL
   (c) ECL (d) इनमें से कोई नहीं
2. निम्न में यूनिपोलर डिवाइस है—
   (a) NMOS (b) RTL (c) TTL (d) ECL

3. लॉजिक परिवार में किसकी गति अधिकतम होती है?
   (a) TTL (b) ECL (c) NMOS (d) RTL
4. DTL सर्किट निम्न की तरह कार्य करता है—
   (a) AND (b) OR (c) NAND (d) NOT
5. HTL का पूरा नाम है—
   (a) High Transistor Logic (b) High Threshold Logic
   (c) High Transistor Level (d) High Threshold Level

उत्तर (Answers)

1. (b), 2. (a), 3. (b), 4. (c), 5. (b).

Scanned by CamScanner

अध्याय 4

डिस्प्ले युक्तियाँ, बाइनरी एडर सर्किट एनकोडर्स, डिकोडर्स तथा मल्टीप्लैक्सर्स, डिमल्टीप्लेक्सर्स

(Display Devices, Binary Adder Circuit Encoders, Decoders and Multiplexers Demultiplexers)

4.1 डिजिटल डिस्प्ले युक्तियाँ (Digital Display Devices)

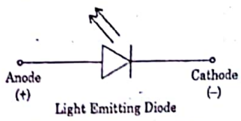
(UPBTE 2013)

- डिजिटल सर्किट के इनपुट एवं आउटपुट, बाइनरी डिजिट के कोड में होते हैं। ये बाइनरी डिजिट के कोड हमें अक्षर (letters) तथा आंकिक संख्या (numericals) के रूप में डिस्प्ले किये जाते है, जिन्हें डिजिटल डिस्प्ले युक्तियाँ (digital display devices) कहते हैं।
- डिजिटल डिस्प्ले युक्तियाँ दो प्रकार की होती है—

  1. प्लेनर (Planar)
  2. नान-प्लेनर (Non-planar)
- प्लेनर यूनिट में सभी कैरेक्टर्स एक तल (same plane) में होते हैं, जबकि नान-प्लेनर यूनिट में सभी कैरेक्टर्स विभिन्न तलों में होते है।
- प्लेनर डिस्प्ले यूनिट प्रकाशित सेगमेन्टल (illuminated segmental) अथवा डॉट मैट्रिक्स (dot matrix) टाइप के होते हैं।
- प्लेनर डिस्प्ले युक्तियाँ हैं—7-सेगमेन्टल डिस्प्ले, 14-सेगमेन्टल डिस्प्ले, 3×5 डॉट मैट्रिक्स, 5×7 डॉट मैट्रिक्स, LED, LCD इत्यादि।
- नॉन-प्लेनर डिस्प्ले यूनिट प्राय: गैसीय डिस्चार्ज पर आधारित होती हैं।
- निक्सी ट्यूब (Nixie tube) एक नॉन-प्लेनर डिस्प्ले यूनिट है।

4.2 प्रकाश उत्सर्जक डायोड (Light Emitting Diode)

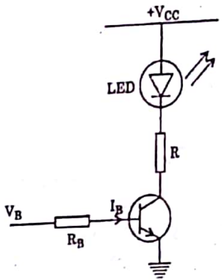
- LED एक प्रकार का डायोड है जो फॉरवर्ड बायस (forward bias) में प्रकाश उत्सर्जित करता है।
- LED सिलिकॉन तथा जर्मेनियम से न बनकर गैलियम, आर्सेनिक तथा फॉस्फोरस से मिलकर बना होता है।
- जब LED गैलियम आर्सेनाइड की सहायता से बना होता है तब लाल रंग उत्सर्जित करता है तथा जब गैलियम फॉस्फाइड से बना होता है तब हरा (green) रंग उत्सर्जित करता है।



Light Emitting Diode

चित्र : 4.1 Light Emitting Diode

- LED का प्रयोग डिजिटल डिस्प्ले युक्तियों जैसे 7-सेगमेन्टल डिस्प्ले, 14-सेगमेन्टल डिस्प्ले अथवा डॉट मैट्रिक्स डिस्प्ले में होता है।
- LED को नीचे चित्र में दिए गए सर्किट की भाँति संयोजित किया जाता है। इस सर्किट में LED को ट्रांजिस्टर के कलेक्टर टर्मिनल में प्रतिरोध R के श्रेणी क्रम में संयोजित किया जाता है।
- जब ट्रांजिस्टर का बेस इनपुट $(V_B) = 0$ (LOW) होता है, ट्रांजिस्टर ऑफ होता है जिससे LED की थ्रो धारा नहीं प्रवाहित होती है।
- जब ट्रांजिस्टर का बेस इनपुट $(V_B) = 1$ (HIGH) होता है तो ट्रांजिस्टर सैचुरेशन (saturation) अवस्था में होता है जिससे LED में धारा प्रवाहित होने से प्रकाश उत्सर्जित होता है।



चित्र : 4.2 LED का संयोजन

- सामान्यत: LED को बनाने के लिए जर्मेनियम तथा सिलिकॉन का प्रयोग नहीं किया जाता है क्योंकि इन सेमीकण्डक्टरों से बने जंक्शन पर एनर्जी हीट के रूप में अधिक तथा लाइट के रूप में कम डिसीपेट होती है।
- जब LED फॉरवर्ड बायस में होता है, उससे प्रकाश उत्सर्जित होता है। यह क्रिया डायोड में PN जंक्शन के समीप मुक्त इलेक्ट्रॉन तथा होल्स के रिकॉम्बिनेशन के कारण होती है।
- LED डिस्प्ले डिवाइसेस में सबसे महत्वपूर्ण युक्ति है।
- LED की प्रचालन वोल्टता 1.2 volt तथा धारा 20 mA है। अधिक विद्युत खपत के कारण बड़े प्रदर्शन युक्तियों में इनका उपयोग नहीं किया जाता है। अत: ऐसी आवश्यकता की पूर्ति के लिए गैसयुक्त प्लाज्मा उपयोग में लाए जाते हैं। इनकी विद्युत खपत साधारण LED की अपेक्षा कम होती है।

LED के लाभ

- LED के विभिन्न लाभ हैं—
  1. कम वोल्टेज (1 या 2 V) तथा कम करेन्ट (5-20 mA)
  2. कम पॉवर कनज्यूम करना (10-150 mW)
  3. ऑन-ऑफ स्विचिंग फास्ट
  4. अधिक जीवन TTL (20 साल से ज्यादा)
  5. तीव्र गति से काम करना

LED के दोष

1. LCD की अपेक्षा अधिक पॉवर व्यय (power loss)।
2. LCD की अपेक्षा अधिक कीमत (महँगा)।
3. तीव्र प्रकाश में LED का प्रकाश आसानी से दिखाई नहीं देता।

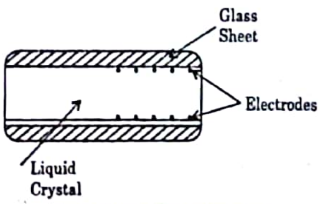
4.3 द्रव क्रिस्टल डिस्प्ले (Liquid Crystal Display)

- LED की भाँति ही LCD का उपयोग अक्षरों एवं अंकों को डिस्प्ले करने में किया जाता है।

Scanned by CamScanner

- LCD में LED की अपेक्षा पॉवर लॉस (power loss) कम होता है। अत: LCD को उस स्थान पर प्रयोग किया जा सकता है जहाँ Power Loss की समस्या हो । जैसे—कैलकुलेटर, डिजिटल घड़ियाँ इत्यादि।
- LCD तीव्र प्रकाश (bright light) में और अधिक चमकते है, जबकि LED प्रकाश तीव्र प्रकाश में आसानी से दिखाई नहीं देता है।
- LCD का जीवन काल (life time) कम होता है तथा इन्हें सर्किट्स में प्रयोग करना कठिन होता है।
- LCD, AC वोल्टेज द्वारा प्रचालित होते हैं जिसकी आवृत्ति 30 Hz से 1000 Hz (1 kHz) के मध्य होती है।
- LCD दो प्रकार के होते हैं—

  1. डायनेमिक स्कैटरिंग टाइप (Dynamic scattering type)
  2. फील्ड इफेक्ट टाइप (Field effect type)
- डायनेमिक स्कैटरिंग टाइप LCD में दो ग्लास शीटों के मध्य में द्रव क्रिस्टल की एक परत होती है।
- ग्लास शीटों में अन्दर की तरफ ट्रान्सपेरेन्ट कनडक्टिव कोटिंग की जाती है जो इलेक्ट्रोड (electrodes) का काम करते है।
- डायनेमिक स्कैटरिंग टाइप LCD में प्रयुक्त किए जाने वाला द्रव क्रिस्टल पदार्थ ऐसा आर्गेनिक कम्पाउण्ड (organic compound) होता है जो द्रव अवस्था में होने पर प्रकाशीय गुण प्रदर्शित करता है।
- जब सेल के टर्मिनल पर विद्युतवाहक बल प्रयुक्त किया जाता है तब विद्युत क्षेत्र के प्रभाव से उत्पन्न परमाणुवीय स्कैटरिंग के कारण प्रकाश सभी दिशाओं में scatter हो जाता है तथा LCD सेल चमकने लगता है। यह प्रक्रिया डायनेमिक स्कैटरिंग कहलाती है।
- फील्ड इफेक्ट टाइप LCD सेल की संरचना डायनेमिक स्कैटरिंग टाइप से जटिल होती है।
- फील्ड इफेक्ट डिस्प्ले में डायनेमिक स्कैटरिंग डिस्प्ले की अपेक्षा पॉवर लॉस (power loss) कम होता है तथा इसका प्रचालन वोल्टेज भी कम होता है।



चित्र : 4.3 Liquid Crystal Display

- LCD का उपयोग उन सभी कार्यों के लिए होता है जहाँ LED का उपयोग होता है।
- लिक्विड क्रिस्टल प्रकाश परावर्तक अथवा ट्रांसमीटर होते हैं। इसलिए बहुत कम विद्युत शक्ति व्यय करते है।
- LCD में जब दिष्ट धारा का उपयोग किया जाता है तब इनका जीवन काल कम होता है। इसलिए इन्हें 500 Hz से कम आवृत्ति की A.C. पर प्रयोग किया जाता है।

LCD के लाभ

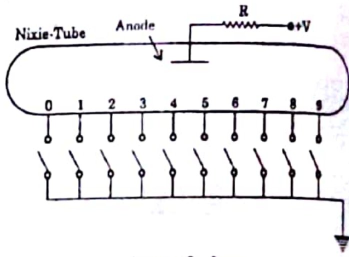
1. LCD में शक्ति व्यय कम होता है।
2. LED की अपेक्षा LCD सस्ते होते हैं।

LCD के दोष

1. Turn-on तथा Turn-off समय अधिक होता है जिससे ये युक्तियां प्रचालन में सुस्त होती है।
2. DC सप्लाई पर प्रयोग करने पर ये शीघ्र खराब हो जाते है, अत: इन्हें AC सप्लाई पर ही प्रयुक्त किया जाता है।

4.4 निक्सी ट्यूब डिस्प्ले (Nixie Tube Display)

- निक्सी ट्यूब एक नान-प्लेनर युक्ति है जो प्रत्येक अंक को डिस्प्ले कर सकती है।
- निक्सी ट्यूब में एक एनोड (anode) तथा विभिन्न कैथोड होते है।
- कैथोड का आकार डिस्प्ले करने वाले अंक एवं अक्षर के समान हो सकता है।
- निक्सी ट्यूब कैरेक्टर (अंक एवं अक्षर) तब तक पारदर्शी होते है जब तक उन्हें Activate न किया जाए।

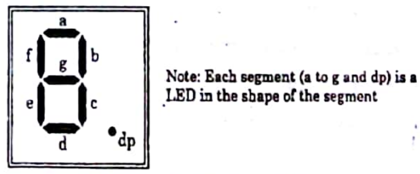


चित्र : 4.4 निक्सी-ट्यूब

- किसी कैथोड को ग्राउन्ड करने पर, उस कैथोड एवं एनोड के मध्य निऑन (neon) गैस आयनित हो जाती है।
- इस आयनन के कारण Anode एवं Cathode के मध्य एक चमक (glow) उत्पन्न होती है तथा इस Glow से इनपुट के अनुसार कोई विशेष अंक डिस्प्ले हो जाता है।

4.5 7-सेगमेन्ट डिस्प्ले (Seven Segment Display)

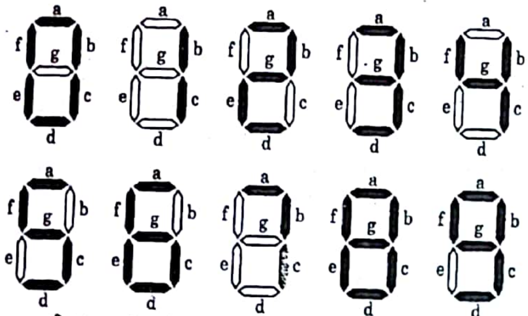
- 7-सेगमेन्ट डिस्प्ले में 7 LEDs *a, b, c, d, e, f* तथा *g* प्रयुक्त किए जाते हैं।
- LED फॉरवर्ड बायस में प्रकाश उत्सर्जित करता है। अत: विभिन्न LEDs को फॉरवर्ड बायस में कर 0 से 9 तक कोई अंक डिस्प्ले कर सकते है।
- उदाहरणत: 8 को डिस्प्ले करने के लिए *a, b, c, d, e, f* तथा *g* सभी को प्रकाशित किया जाता है तथा 0 को डिस्प्ले करने के लिए *a, b, c, d, e* तथा *f* को प्रकाशित किया जाता है।
- कॉमन एनोड 7-सेगमेन्ट डिस्प्ले में LED तथा ग्राउन्ड के मध्य एक धारा सीमित करने के लिए प्रतिरोध लगाया जाता है। प्रतिरोध के मान द्वारा LED में प्रवाहित होने वाली धारा निश्चित होती है।



Standard form of seven segment display

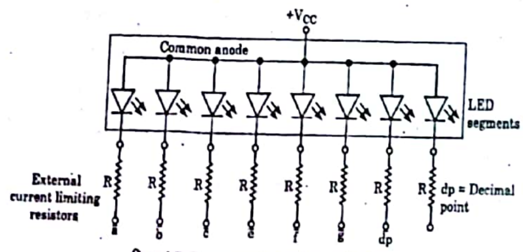
चित्र : 4.5

Scanned by CamScanner

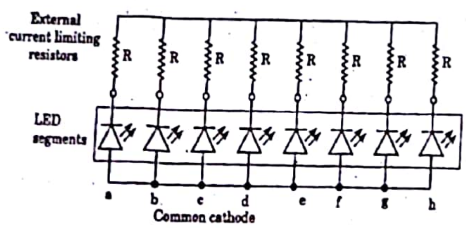


चित्र : 4.6 Various digits displayed with seven segment display

- 7-सेगमेन्ट डिस्प्ले दो प्रकार के होते हैं—
    1. कॉमन एनोड टाइप (Common Anode Type)
    2. कॉमन कैथोड टाइप (Common Cathode Type)
- कॉमन एनोड तथा कॉमन कैथोड 7-सेगमेन्ट डिस्प्ले को क्रमशः चित्र 4.7 तथा 4.8 में प्रदर्शित किया गया है।
- कॉमन एनोड टाइप में सभी LEDs के एनोड को परस्पर संयोजित करके एक टर्मिनल को बाहर निकालते हैं, जबकि कॉमन कैथोड टाइप में सभी LEDs के कैथोड एक साथ संयोजित रहते हैं।



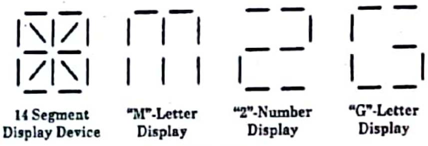
चित्र : 4.7 Common Anode LED Display



चित्र : 4.8 Common Cathode LED Display

4.6 14 सेगमेन्ट डिस्प्ले (Fourteen Segment Display)

- 14-सेगमेन्ट डिस्प्ले का प्रयोग गणितीय अंकों तथा अक्षरों दोनों को डिस्प्ले करने में होता है।
- चित्र 4.9 में 14-सेगमेन्ट डिस्प्ले प्रदर्शित किया गया है जिससे कुछ अंकों तथा अक्षरों को दिखाया गया है।

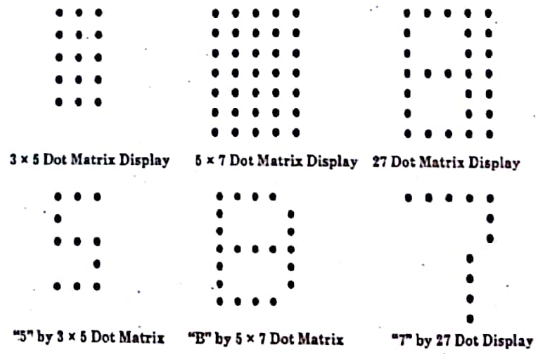


चित्र : 4.9

- 14-सेगमेन्ट डिस्प्ले में 14 LED अवस्थित किये जाते हैं। इनमें से विभिन्न आवश्यक डिस्प्ले युक्तियों को एक्टिव करके हम वांछित अंक (0 से 9 तक कोई भी) अथवा अक्षर (A से Z तक कोई भी) प्राप्त कर सकते हैं।

4.7 डॉट मेट्रिक्स डिस्प्ले (Dot Matrix Display)

- डॉट मेट्रिक्स डिस्प्ले में LEDs अथवा LCDs को मेट्रिक्स के फॉर्म में व्यवस्थित करते हैं तथा अंकों एवं अक्षरों को डॉट मेट्रिक्स के रूप में डिस्प्ले करते हैं।
- नीचे चित्र में 3 × 5 डॉट मेट्रिक्स, 5 × 7 डॉट मेट्रिक्स तथा 27 डॉट मेट्रिक्स डिस्प्ले की व्यवस्था को दर्शाया गया है।
- Dot matrix Display में हम एक कॉलम से अधिक पर वोल्टेज प्रयुक्त कर तथा एक से अधिक रो (row) को ग्राउन्ड कर कोई भी डेसीमल अंक अथवा अक्षर डिस्प्ले कर सकते हैं।



चित्र : 4.10

4.8 बाइनरी एडर सर्किट (Binary Adder Circuits)

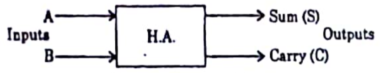
- डिजिटल कम्प्यूटर द्वारा बाइनरी संख्याओं को जोड़ना, डिजिटल कम्प्यूटर का एक महत्वपूर्ण ऑपरेशन है।

Scanned by CamScanner

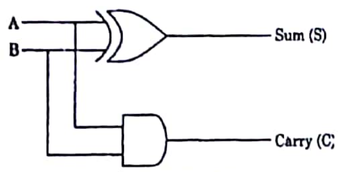
- बाइनरी एडर को दो भागों में बाँटा गया है—
  1. हाफ एडर (Half adder)
  2. फुल एडर (Full adder)

4.8.1 हाफ एडर (Half Adder) *(UPBTE 2012, 13, 14, 16)*

- हाफ एडर एक कॉम्बिनेशनल (combinational) लॉजिक सर्किट है जिसके दो इनपुट तथा दो आउटपुट होते हैं।
- हाफ एडर दो सिंगल बिट संख्याओं को जोड़ता है जिसके दो आउटपुट मिलते है—"सम" (sum) तथा कैरी (carry)।
- दो बाइनरी संख्याओं A तथा B के लिए हाफ एडर का ब्लॉक डायग्राम नीचे दर्शाया गया है।
- हाफ एडर सर्किट में एक Ex-OR गेट तथा दूसरा AND गेट प्रयुक्त किया गया है।



चित्र : 4.11 Block diagram of Half Adder



चित्र : 4.12 Half Adder Circuit

- Ex-OR गेट की आउटपुट इनपुट संख्याओं का योग होता है तथा AND गेट का आउटपुट योग में प्राप्त कैरी (carry) होती है।
- Ex-OR के प्रचालन से योग (sum)

$$S = A\bar{B} + B\bar{A}$$

- AND गेट के प्रचालन से कैरी (carry)

$$C = AB$$

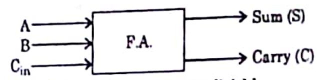
- हॉफ एडर द्वारा बाइनरी संख्याओं को जोड़ने के नियम से सत्य तालिका—

Inputs		Outputs	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

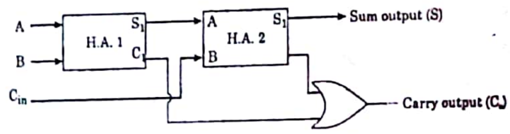
- तालिका से स्पष्ट है कि Ex-OR गेट केवल तब HIGH आउटपुट देता है जब दोनों इनपुट में से कोई एक HIGH हो तथा AND गेट का आउटपुट तब HIGH आउटपुट देता है जब दोनों इनपुट HIGH हो।

4.8.2 फुल एडर (Full Adder)

- हाफ एडर में केवल दो इनपुट टर्मिनल होते हैं, अत: यह केवल दो बाइनरी सख्या को जोड़ता है।
- हाफ एडर की इस कमी को पूरा करने के लिए तथा दो से अधिक बाइनरी संख्याओं को जोड़ने के लिए फुल एडर का विकास हुआ।
- फुल एडर सर्किट एक कॉम्बिनेशनल लॉजिक सर्किट है जिसमें तीन इनपुट टर्मिनल होते हैं तथा दो आउटपुट टर्मिनल होते हैं।

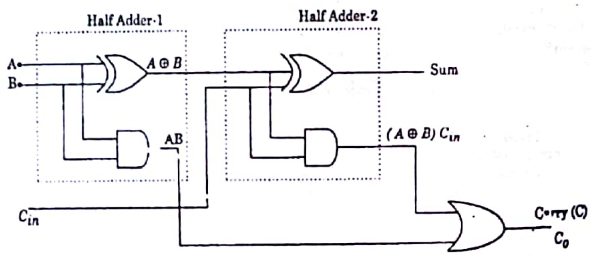


चित्र : 4.13 Diagram of Full Adder



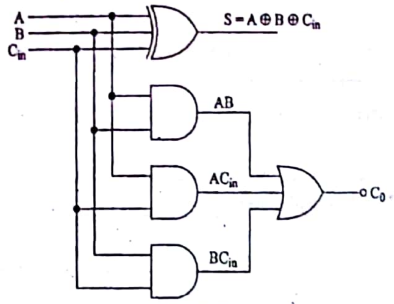
चित्र : 4.14 Full adder using half adders

- फुल एडर दो हाफ एडर की सहायता से बनाया जाता है जिसका चित्र ऊपर दर्शाया गया है।
- फुल एडर द्वारा बाइनरी संख्याओं को जोड़ने के लिए सत्य तालिका चित्र 4.16 में दिखाया गया है।
- सत्य तालिका से स्पष्ट है कि Ex-OR गेट को ड्राइव करने वाली इनपुट में 1s की संख्या Odd है, तब आउटपुट Sum भी High प्राप्त होगी तथा जब $A\ B\ C$ में से दो या उससे अधिक इनपुट High होगी तब Carry High होती है।



चित्र : 4.15 (a) Full adder using two half adders

Scanned by CamScanner



चित्र : 4.15 (b)

- Full Adder दो से अधिक बाइनरी बिट को जोड़ता है। चित्र में तीन इनपुट A, B तथा C_{in} को दिखाया गया है। परिप के दो आउटपुट है—Sum तथा Carry। दूसरी आउटपुट कैरी कहलाती है क्योंकि इसे अगले उच्च क्रम के कॉलम प्रयोग करते है।

$$Sum(S) = A \oplus B \oplus Cin$$

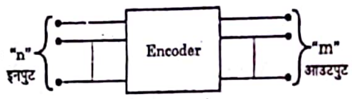
$$Carry(C_0) = AB + AC_{in} + BC_{in}$$

Input			Outputs	
A	B	C_{in}	Sum	Carry
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

4.9 एन्कोडर्स (Encoders)

(UPBTE 2015)

- एन्कोडर एक कॉम्बिनेशनल लॉजिक सर्किट है जो प्रत्येक इनपुट सिग्नल को एक विशेष बाइनरी कोड (जैसे बाइन नम्बर, BCD etc.) के आउटपुट सिग्नल में परिवर्तित करता है।
- एक एन्कोडर में "n" नम्बर की इनपुट लाइन तथा "m" नम्बर की आउटपुट लाइनें होती है।

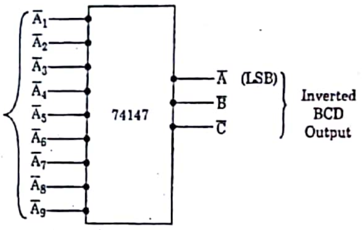


चित्र : 4.17 एन्कोडर

- एन्कोडर इनपुट पर "n" नम्बर की डिजिटल वर्ड को एक्सेप्ट करता है तथा उसे बदल कर "m" बिट की डिजिटल वर्ड आउटपुट पर देता है।
- Encoder एक ऐसा लॉजिक परिपथ है जिसमें एक से अधिक इनपुट लाइनें तथा आवश्यक संख्या युक्त आउटपुट लाइनें होती हैं। परन्तु एक समय में केवल एक इनपुट लाइन ही Active होती है।
- Encoder प्रत्येक इनपुट सिग्नल को एक विशेष बाइनरी कोड (जैसे—Binary Number, BCD code, Excess-3 code) में से एक की उचित संख्या में परिवर्तित करता है।

4.9.1 एन्कोडर के प्रकांर (Types of Encoders)

- एन्कोडर को निम्नलिखित प्रकार से बाँटा गया है—
  1. प्रायोरिटी एन्कोडर (Priority Encoder)
  2. डेसिमल से BCD एन्कोडर (Decimal to BCD Encoder)
  3. ऑक्टल से बाइनरी एन्कोडर (Octal to Binary Encoder)
  4. हेक्साडेसिमल से बाइनरी एन्कोडर (Hexadecimal to Binary Encoder)
- Priority Encoder में इनपुट लाइन को Priority दी जाती है। यदि दो या दो से अधिक लाइन एक समय पर "1" है तो अधिक Priority वाली इनपुट लाइन को कन्सीडर किया जाएगा।
- डेसिमल से BCD Encoder को 10:4 Encoder भी कहते हैं।



चित्र : 4.18 Logic Symbol of IC 74147

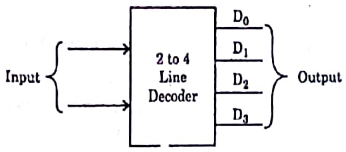
- IC 74147 Decimal से BCD Encoder का उदाहरण है जो नीचे चित्र में दर्शाया गया है।
- ऑक्टल से बाइनरी एन्कोडर में 8-इनपुट लाइनें होती हैं तथा 3-आउटपुट लाइनें होती हैं।
- IC 74148 एक ऑक्टल से बाइनरी एन्कोडर है।
- हेक्साडेसिमल से बाइनरी एन्कोडर दो ऑक्टल से बाइनरी एन्कोडर को मिलाकर बनाया जाता है।
- दोनों ऑक्टल से बाइनरी एन्कोडर के आउटपुट को 2:1 मल्टीप्लेक्सर को देने पर हेक्साडेसिमल से बाइनरी एन्कोडर प्राप्त होता है।

Scanned by CamScanner

4.10 डिकोडर्स (Decoders)

(UPBTE 2015)

- डिकोडर एक कॉम्बिनेशनल सर्किट है जो n-बिट् बाइनरी सूचना को 2^n आउटपुट लाइन में परिवर्तित करता है।
- उदहारणत: यदि डिकोडर के इनपुट में दो बाइनरी लाइनें हैं तब डिकोडर की आउटपुट पर चार आउटपुट लाइनें प्राप्त होंगी।



चित्र : 4.19 Block Diagram of 2 Line to 4 Line Decoder

- 2:4 लाइन डिकोडर के लिए A तथा B दो इनपुट लाइने हैं तथा D_0, D_1, D_2 तथा D_3 आउटपुट है।

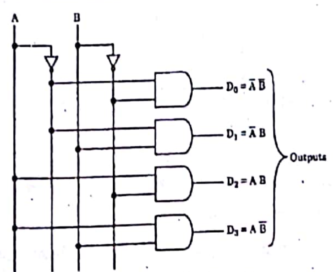
Inputs		Outputs			
A	B	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

चित्र : 4.20 Truth Table of 2 Line to 4 Line Decoder

- डिकोडर के चारों आउटपुट के लिए बूलियन एक्सप्रेशन

$$D_0 = \bar{A}\ \bar{B}$$
$$D_1 = \bar{A}\ B$$
$$D_2 = A\ \bar{B}$$
$$D_3 = A\ B$$

- 2 से 4 लाइन डिकोडर का लॉजिक डायग्राम-



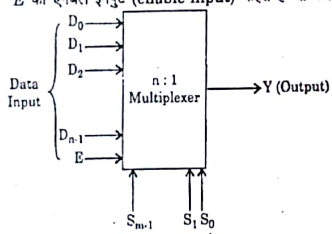
चित्र 4.21 : 2 to 4 Line Decoder

- IC 74138 एक 3:8 डिकोडर है जिसमें इनपुट में 3 लाइन तथा आउटपुट की तरफ 8 लाइनें होती हैं। इस प्रकार 4:16 डिकोडर में 4 इनपुट लाइनें तथा 16 आउटपुट लाइनें होती हैं।
- डिजिटल प्रणाली में डिकोडर बाइनरी सन्देश को डेसीमल अथवा अन्य नम्बर प्रणाली में परिवर्तित करता है।
- इसका उपयोग डिजिटल परिपथों में डाटा प्रवेश कराने के लिए तथा डाटा बाहर लाने के लिए किया जाता है।
- प्राय: डेकोडर दो प्रकार के होते हैं—
  1. डिस्प्ले डिकोडर
  2. लॉजिक डिकोडर।

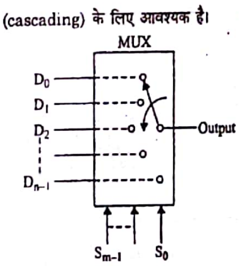
4.11 मल्टीप्लेक्सर (Multiplexers)

(UPBTE 2013, 15, 16)

- मल्टीप्लेक्सर एक विशेष कॉम्बिनेशनल सर्किट है जिसमें अनेक इनपुट तथा केवल एक आउटपुट होता है।
- यह एक डेटा सैलेक्टर सर्किट है।
- मल्टीप्लेक्सर एक डिजिटल सर्किट है जो केवल "n" इनपुट में से एक को सेलेक्ट कर आउटपुट पर देता है।
- "n" इनपुट को सेलेक्ट करने के लिए हमे "m" सेलेक्ट लाइन की आवश्यकता होती है, जहाँ $2^m = n$.
- E को एनेबल इनपुट (enable input) कहते है जो कैसकेडिंग (cascading) के लिए आवश्यक है।



चित्र : 4.22 मल्टीप्लेक्सर



चित्र : 4.22.1 An Equivalent Circuit

- मल्टीप्लेक्सर डिजिटल सर्किट की विश्वसनीयता को बढ़ा देता है, क्योंकि यह आउटपुट पर कनेक्शन को कम कर देता है।
- Multiplexer का अर्थ है बहुत-से इनपुटों में से एक को सेलेक्ट करना।
- Multiplexer चिप को Enable/Disable करने के लिए प्राय: चिप में एक Enable पिन होती है जोकि प्राय: Low सक्रिय होती है, अर्थात् $E = 0$ अवस्था में होने पर Multiplexer चिप पर वांछित प्रचालन करती है।

मल्टीप्लेक्सर के गुण

1. वायर (wire) की संख्या को घटा देता है।
2. वायर (wire) की संख्या घटाने से सर्किट अधिक जटिल नहीं होता है।
3. मल्टीप्लेक्सर की सहायता से हम विभिन्न कॉम्बिनेशनल सर्किट बना सकते है।

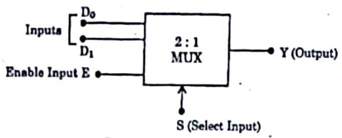
Scanned by CamScanner

4.11.1 मल्टीप्लेक्सर के प्रकार (Types of Multiplexer)

मल्टीप्लेक्सर के प्रकार हैं—

1. 2 : 1 मल्टीप्लेक्सर
2. 4 : 1 मल्टीप्लेक्सर
3. 8 : 1 मल्टीप्लेक्सर
4. 16 : 1 मल्टीप्लेक्सर

- 2 : 1 मल्टीप्लेक्सर में दो इनपुट, एक सेलेक्ट इनपुट तथा एक एनेबल (enable) इनपुट होता है। आउटपुट पर हमें



चित्र : 4.23 2 : 1 मल्टीप्लेक्सर

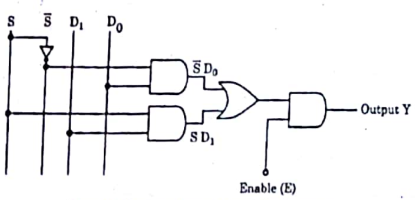
केवल एक आउटपुट प्राप्त होता है।

- 2 : 1 Multiplexer में आउटपुट $Y = 1$ केवल दो अवस्थाओं मे होगा।

जब $Y = E\,\overline{S}\,D_0 + E\,S\,D_1$

$Y = E\,(\overline{S}\,D_0 + S\,D_1)$

- गेट्स की सहायता से 2 : 1 मल्टीप्लेक्सर का चित्र

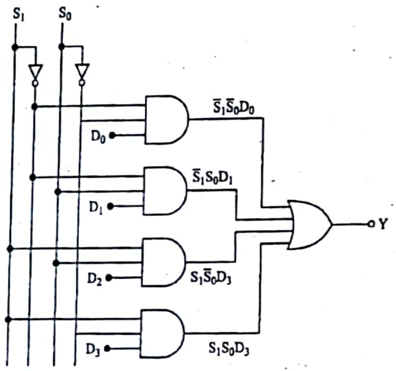


चित्र : 4.24 Realization of 2 : 1 MUX using gates

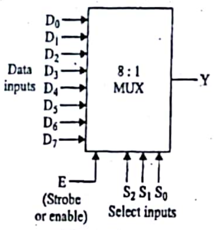
- 2 : 1 मल्टीप्लेक्सर की भाँति 4 : 1 मल्टीप्लेक्सर में 4 डाटा इनपुट 2 सेलेक्ट लाइन तथा एक आउटपुट होता है।
- 4 : 1 मल्टीप्लेक्सर में हमें आउटपुट तब High प्राप्त होगा जब सेलेक्ट इनुपट 1 होगा।

0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

चित्र : 4.25.1



चित्र : 4.25.2 Realization of 4 : 1 multiplexer using basic gates



(a) Block diagram

चित्र : 4.25.3 8 : 1 multiplexer

Enable	Select inputs			Output
0	S_2	S_1	S_0	Y
1	x	x	x	0
1	0	0	0	D_0
1	0	0	1	D_1
1	0	1	0	D_2
1	0	1	1	D_3
1	1	0	0	D_4
1	1	0	1	D_5
1	1	1	0	D_6
1	1	1	1	D_7

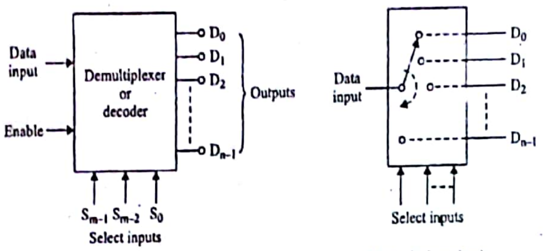
(b) Truth table

Scanned by CamScanner

4.12 डिमल्टीप्लेक्सर (Demultiplexers)

(UPBTE 2013, 15)

- डिमल्टीप्लेक्सर एक कॉम्बिनेशनल सर्किट है।
- डिमल्टीप्लेक्सर में एक इनपुट तथा अनेक आउटपुट होते हैं।
- डिमल्टीप्लेक्सर मल्टीप्लेक्सर के विपरीत ऑपरेशन को परफॉर्म (perform) करता है।
- एक डिमल्टीप्लेक्सर में 1 इनपुट "n" आउटपुट तथा m सेलेक्ट इनपुट होते हैं जो नीचे चित्र में दर्शाये गये हैं।



(a) 1 : n demultiplexer (b) Equivalent circuit

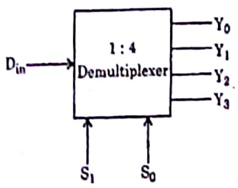
चित्र : 4.26

- एनेबल इनपुट (enable input) डिमल्टीप्लेक्सर को एनेबल (enable) करने का कार्य करता है।
- डिमल्टीप्लेक्सर के "n" आउटपुट तथा "m" सेलेक्ट इनपुट के मध्य $n = 2^m$ का सम्बन्ध होता है।
- Demultiplexer सर्किट की प्रक्रिया प्राय: multiplexer सर्किट के विपरीत होती है।
- Demultiplexer में multiplexer की भाँति कुछ कन्ट्रोल इनपुट टर्मिनल होते हैं जिन पर विभिन्न बाइनरी बिटों के कॉम्बिनेशन प्रयुक्त करके आउटपुट की वांछित लाइन पर डाटा प्राप्त कर सकते है।

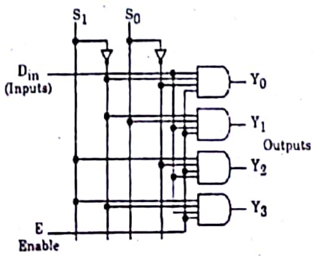
4.12.1 डिमल्टीप्लेक्सर के प्रकार (Types of Demultiplexer)

डिमल्टीप्लेक्सर के निम्न प्रकार हैं—

1. 1 : 2 डिमल्टीप्लेक्सर
2. 1 : 4 डिमल्टीप्लेक्सर



चित्र : 4.27 Block Diagram of 1 : 4 Demultiplexer



चित्र : 4.28 1 : 4 demultiplexer

3. 1 : 8 डिमल्टीप्लेक्सर
4. 1 : 16 डिमल्टीप्लेक्सर

- 1 : 2 डिमल्टीप्लेक्सर में 1 डाटा इनपुट, 1 सेलेक्ट इनपुट एवं एक आउटपुट होता है।
- 1 : 2 डिमल्टीप्लेक्सर की भाँति 1 : 4 में डिमल्टीप्लेक्सर 1 डेटा इनपुट, दो सेलेक्ट इनपुट तथा 4 आउटपुट होते हैं, जो नीचे चित्र में दर्शाये गये है।
- इस प्रकार 1 : 8 डिमल्टीप्लेक्सर में 1 डाटा इनपुट, 8 आउटपुट तथा 3 सेलेक्ट इनपुट होते हैं तथा 1 : 16 डिमल्टीप्लेक्सर में 1 डाटा इनपुट, 16 आउटपुट तथा 4 सेलेक्ट इनपुट होते है।
- IC 74138, IC 74139, IC 74154 तथा IC 74155 कुछ डिमल्टीप्लेक्सर ICs हैं।

4.13 मल्टीप्लेक्सर तथा डिमल्टीप्लेक्सर में तुलना (Comparison of Multiplexer and Demultiplexer)

	पैरामीटर	मल्टीप्लेक्सर	डिमल्टीप्लेक्सर
1.	लॉजिक सर्किट	कॉम्बिनेशनल	कॉम्बिनेशनल
2.	डाटा इनपुट की संख्या	n	1
3.	सेलेक्ट इनपुट की संख्या	m	m
4.	डाटा आउटपुट की संख्या	1	n
5.	इनपुट/आउटपुट लाइन तथा सेलेक्ट लाइन में सम्बन्ध	$n = 2^m$	$n = 2^m$
6.	ऑपरेशन सिद्धान्त	Many to 1 या डेटा सेलेक्टर (Data Selector)	1 to Many या डाटा डिस्ट्रीब्यूटर (Data Distributor)
7.	एप्लीकेशन	टाइम डिविजन मल्टीप्लेक्सिंग (TDM) में सेन्डिंग (Sending) End पर	TDM में Receiving End पर

प्रश्नावली (Exercise)

1. अर्द्ध-जोड़क की कार्य प्रणाली के बारे में व्याख्या कीजिए। *(UPBTE 2012, 16)*
2. पूर्ण जोड़क (full adder) परिपथ, उसकी सत्य तालिका एवं कार्य के बारे में विस्तार से समझाइये। *(UPBTE 2011)*

Scanned by CamScanner

3. Full Adder का सर्किट दो Half Adder से बनाएँ एवं सत्यापित टेबिल लिखें। *(UPBTE 2010)*
4. डिजिटल परिपथों में डिस्प्ले के लिये प्रयोग की जाने वाली युक्तियों का विस्तारपूर्वक वर्णन कीजिए।
5. LEDs तथा LCDs में अन्तर स्पष्ट कीजिए तथा उनकी संरचना दीजिए।
6. 7-Segment डिस्प्ले की कार्य प्रणाली को बताइये तथा 14-Segment का भी वर्णन करें।
7. Nixie7-ट्यूब से आप क्या समझते है?
8. Common-Anode एवं Common-Cathode 7-सैगमैन्ट में अन्तर स्पष्ट कीजिए। प्रत्येक का उपयोग बताइये तथा परिपथ बनाइये।
9. मल्टीप्लेक्सर का कार्य समझाइये। एक 2 line -to- 1 line मल्टीप्लेक्सर का लॉजिक परिपथ बनाइये तथा सत्य-तालिका को भी बनाइये।
10. Liquid Crystal display पर एक संक्षिप्त टिप्पणी लिखिये।
11. Multiplexers पर एक टिप्पणी लिखिये? *(UPBTE 2013, 15, 16)*
12. Digital निकाय में Encoder तथा Decoder के कार्यों को समझाइये। *(UPBTE 2015)*
13. Encoder से आप क्या समझते है?
14. Decoder का वर्णन कीजिये।
15. मल्टीप्लेक्सर तथा डिमल्टीप्लेक्सर में अन्तर स्पष्ट कीजिये। *(UPBTE 2013, 15)*
16. Demultiplexers का सचित्र वर्णन कीजिये। *(UPBTE 2013, 15)*

बहुविकल्पीय प्रश्न

1. LED किस बायस में कार्य करता है?
   (a) Forward (b) Reverse (c) दोनों (d) इनमें से कोई नहीं
2. Half Adder में कम से कम इनपुट होंगे—
   (a) 0 (b) 3 (c) 2 (d) 1
3. Full Adder में आउटपुट की संख्या होती है—
   (a) 4 (b) 5 (c) 2 (d) 1
4. LCD प्रदर्शित करता है—
   (a) Liquid Crystal Device (b) Line Crystal Device
   (c) Liquid Crystal Display (d) Line Crystal Display
5. 4 : 1 MUX में कितने select इनपुट होंगे?
   (a) 1 (b) 2 (c) 3 (d) 4

उत्तर (Answers)

1.(a), 2. (c), 3. (c), 4. (c), 5. (b)

अध्याय 5

फ्लिप फ्लॉप (Flip-Flops)

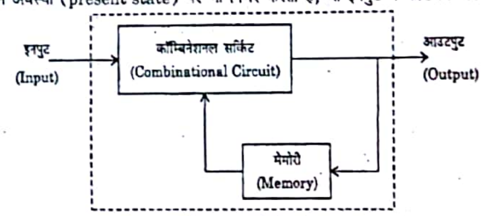
5.1 कॉम्बिनेशनल तथा सिक्वैन्शियल सर्किट (Combinational and Sequential Circuits)

5.1.1 कॉम्बिनेशनल सर्किट्स

- कॉम्बिनेशनल सर्किट में फीडबैक का उपयोग नही होता है। अत: इसकी आउटपुट केवल वर्तमान के इनपुट पर निर्भर करती है।
- कॉम्बिनेशनल सर्किट में मेमोरी नहीं होती है, अत: इसका प्रीवियस स्टेट वर्तमान स्टेट पर कोई प्रभाव नहीं डालता है।
- पाँच बेसिक गेट्स AND, OR, NOT, NAND तथा NOR कॉम्बिनेशनल सर्किट के उदाहरण हैं।
- ADDER तथा SUBTRACTOR भी इसके उदाहरण है।

5.1.2 सिक्वैन्शियल सर्किट

- सिक्वैन्शियल सर्किट में फीडबैक का उपयोग होता है। अत: इसकी आउटपुट वर्तमान के इनपुट के साथ-साथ आउटपुट की वर्तमान अवस्था (present state) पर भी निर्भर करती है, जो इनपुट में फीडबैक की जाती है।



चित्र : 5.1 ब्लॉक डायग्राम ऑफ सिक्वैन्शियल सर्किट

- इस प्रकार हम कह सकते है कि सिक्वैन्शियल सर्किट में मेमारी होती है।
- Flip-Flops, Shift Registers तथा Counters सभी सिक्वैन्शियल सर्किट के उदाहरण हैं।

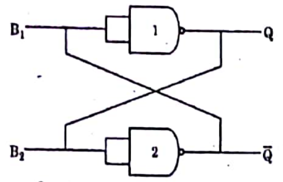
5.2 फ्लिप-फ्लॉप (Flip-Flop)

(UPBTE 2014)

- फ्लिप-फ्लॉप एक बेसिक डिजिटल मेमोरी परिपथ है।
- इसकी दो अवस्थाएँ होती है—लॉजिक 0 स्टेट तथा लॉजिक 1 स्टेट।

Scanned by CamScanner

- फ्लिप-फ्लॉप को डिजाइन करने के लिए NAND तथा NOR गेटों का उपयोग किया जाता है।

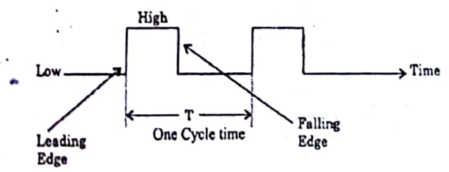


चित्र : 5.2 Cross coupled Inverter

- ऊपर चित्र में NAND गेट की सहायता से फ्लिप-फ्लॉप की डिजाइन को दर्शाया गया है।
- यहाँ दोनों NAND गेट इनवर्टर की भाँति कार्य कर रहे हैं। अत: इसे क्रॉस कपल्ड इनवर्टर (cross coupled invertor) भी कहते हैं।
- चित्र में गेट 1 का आउटपुट गेट 2 के इनपुट में तथा गेट 2 का आउटपुट गेट 1 के इनपुट में जोड़ा गया है।
- क्रॉस कपल्ड इनवर्टर सूचनाओं को लॉक करने या लैच करने में सक्षम है, अत: इस सर्किट को लैच (latch) भी कहते है।
- लैच एक बाइस्टेबल एलीमेन्ट है जिसके दो स्टेबल स्टेट होते है।

5.3 क्लॉक सिग्नल (Clock Signal)

- क्लॉक एक रैक्टेंगुलर सिग्नल होता है जो कि चित्र में दर्शाया गया है।

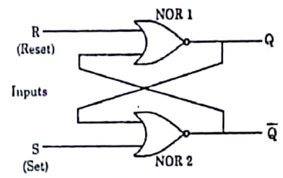


चित्र : 5.3 क्लॉक सिग्नल

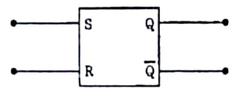
- क्लॉक सिग्नल प्रत्येक T second बाद खुद को दोहराता है। अधिकांश फ्लिप-फ्लॉप क्लॉक टाइप (clock type) होते हैं जिनकी अवस्था एक नियत दर पर परिवर्तित होती है।
- क्लॉक फ्लिप-फ्लॉप में अवस्था परिवर्तन केवल क्लॉक के द्वारा किया जाता है।
- कुछ परिस्थितियों में अवस्था परिवर्तन क्लॉक के चढ़ते सिरे (leading edge) तथा कुछ में उतरते सिरे (falling edge) पर होता है।
- यदि एक क्लॉक द्वारा उत्पन्न फ्लिप-फ्लॉप की निर्गत Q_n से इंगित की जाये तो अगले क्लॉक द्वारा उत्पन्न निर्गत Q_{n+1} द्वारा इंगित होगी।

5.4 S-R फ्लिप-फ्लॉप (S-R Flip-Flop)

- चित्र मे NOR गेट प्रयुक्त कर SR फ्लिप-फ्लॉप का परिपथ प्रदर्शित किया गया है।
- यहाँ दो NOR गेट को क्रॉस कपल्ड करके जोड़ा गया है, इससे स्पष्ट है कि NOR गेट 1 का आउटपुट NOR गेट 2 के इनपुट के साथ तथा NOR गेट 2 का आउटपुट NOR गेट 1 के इनपुट के साथ जोड़ा गया है।



चित्र : 5.4 NOR गेट SR फ्लिप-फ्लॉप



चित्र : 5.5 Symbol of SR flip-flop

R	S	Q	Action
0	0	Last value	No change
0	1	1	Set
1	0	0	Reset
1	1	×	Prohibited State

चित्र : 5.6 S-R फ्लिप-फ्लॉप की सत्य तालिका

Case 1. $R = 0$, $S = 0$

- चूँकि NOR इनपुट गेट के इनपुट पर 0 आउटपुट को प्रभावित नहीं करता, अत: फ्लिप-फ्लॉप अपनी पूर्व अवस्था में ही रहेगा अर्थात् आउटपुट Q पर कोई प्रभाव नहीं पड़ेगा।

Case 2. $R = 0$, $S = 1$

- चूँकि S = 1, अत: NOR गेट 2 का आउटपुट $\overline{Q} = 0$.
- इससे NOR गेट 1 के दोनों इनपुट 0 हो जाएँगे।
- NOR गेट 1 के दोनों इनपुट 0 होने से आउटपुट $Q = 1$.
- इस समय NOR गेट 2 के दोनों इनपुट 1 हो गए हैं, अत: आउटपुट $\overline{Q} = 0$.

Case 3. $R = 1$, $S = 0$

- चूँकि $R = 1$, अत: NOR 1 का आउटपुट $Q = 0$.

Scanned by CamScanner

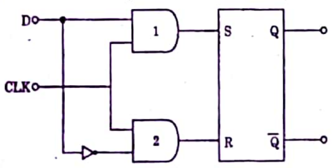
- अब NOR-2 के दोनों इनपुट 0 हो जाएँगे।
- अत: NOR-2 का आउटपुट $\overline{Q}$ = 1.

Case 4. $R = 1$, $S = 1$

- जब $S = R = 1$ होता है तब दोनों आउटपुट Q तथा $\overline{Q}$ का मान 0 हो जाता है अर्थात् दोनों Low स्टेट में आ जाते हैं जोकि फ्लिप-फ्लॉप की मूल भाषा के विपरीत हैं।
- फ्लिप की आउटपुट Q दूसरी आउटपुट $\overline{Q}$ का कॉम्प्लीमेन्ट होना चाहिए। अत: $R = S = 1$ कभी प्रयुक्त नहीं की जा सकती है।
- जब $S = R = 1$ होता है तो इस स्थिति को RACE Condition कहते हैं।

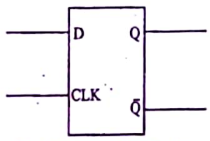
5.5 D फ्लिप-फ्लॉप (D Flip-Flop)

- RS फ्लिप-फ्लॉप में दो डाटा इनपुट R तथा S लाइन की आवश्यकता होती है।
- RS फ्लिप-फ्लॉप में HIGH बिट् को स्टोर करने के लिए, S इनपुट HIGH रखी जाती है तथा LOW बिट् को स्टोर करने के लिए R इनपुट HIGH रखी जाती है। इस प्रकार फ्लिप-फ्लॉप के लिए दो सिग्नलों की आवश्यकता होती है।



चित्र : 5.7 D -फ्लिप-फ्लॉप

- इसके अतिरिक्त R तथा S दोनों इनपुट के एक साथ HIGH होने की स्थिति मान्य नहीं है।
- यह दोष D फ्लिप-फ्लॉप में उत्पन्न नहीं होते हैं। D फ्लिप-फ्लॉप मे केवल एक डाटा इनपुट की आवश्यकता होती है।



चित्र : 5.8 Symbol of D flip-flop

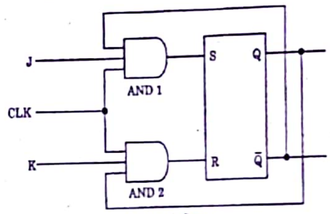
- D-फ्लिप-फ्लॉप में डाटा D तब तक आउटपुट Q पर प्राप्त नहीं होता जब तक क्लॉक पल्स CLK टर्मिनल पर नहीं आती।

CLK	D	Q_{n+1}
0	×	Q_n (Last value)
1	0	0
1	1	1

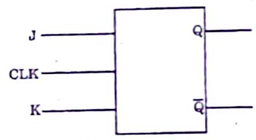
- जब क्लॉक (CLK) Low होती है तब दोनों AND गेट्स Disable रहते हैं। अत: D की स्टेट में परिवर्तन से आउटपुट प्रभवित नहीं होती।
- जब क्लॉक (CLK) High होती है तब दोनों AND गेट्स Enable हो जाते हैं। इस स्थिति में आउटपुट Q इनपुट D के समान हो जाती है।

5.6 J-K फ्लिप-फ्लॉप (J-K Flip-Flop)

- J-K फ्लिप-फ्लॉप एक अत्यन्त उपयोगी फ्लिप-फ्लॉप है जिसका उपयोग काउन्टर्स में किया जाता है।
- नीचे चित्र मे J-K फ्लिप-फ्लॉप का परिपथ एवं संकेत दिखाया गया है।



चित्र : 5.9 परिपथ



चित्र : 5.10 Symbol of JK Flip-Flops

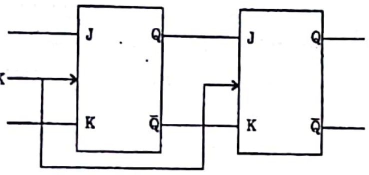
- चित्र में J तथा K कन्ट्रोल इनपुट हैं। परिपथ का प्रचालन क्लॉक पल्स द्वारा होता है।

CLK	J	K	Q_{n+1}	Action
0	×	×	Q_n	No change
1	0	0	Q_n	No change
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	$\overline{Q}_n$	Toggle

चित्र : 5.11 J-K फ्लिप-फ्लॉप की सत्य तालिका

Scanned by CamScanner

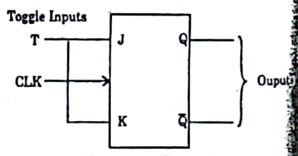
- सत्य तालिका में दिखायी गई $J=K=1$ की स्थिति को रेस एराउन्ड कन्डीशन (race around condition) कहते हैं। इस स्थिति में फ्लिप-फ्लॉप toggle mode में होता है।
- फ्लिप-फ्लॉप को Race Around Conditions से रोकने के लिए *J-K* मास्टर-स्लेव फ्लिप-फ्लॉप का उपयोग किया जाता है।
- मास्टर स्लेव फ्लिप-फ्लॉप के परिपथ में दो फ्लिप-फ्लॉप प्रयुक्त किये जाते हैं, एक मास्टर (master) तथा दूसरा स्लेव (slave)।



चित्र : 5.12 *J-K* Master-Slave फ्लिप-फ्लॉप

5.7 "*T*" फ्लिप-फ्लॉप ("*T*" Flip-Flop)

- *T* फ्लिप-फ्लॉप को टॉगल (toggle) फ्लिप-फ्लॉप भी कहते हैं।
- Toggle Flip-Flop एक प्रकार का *JK* फ्लिप-फ्लॉप है जिसमें *J* तथा *K* टर्मिनल परस्पर जुड़े होते हैं। इस प्रकार इसमें केवल एक इनपुट टर्मिनल होता है, जिसे "*T*" से दर्शाया जाता है।



चित्र : 5.13 *T*-फ्लिप-फ्लॉप

Q_n	T	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	1

चित्र : 5.14 *T* फ्लिप-फ्लॉप की सत्य तालिका

- *T* फ्लिप-फ्लॉप में इनपुट पर Trigger Pulse दी जाती है। प्रत्येक Trigger Pulse आने पर फ्लिप-फ्लॉप की आउटपुट बदलती है।
- *T* फ्लिप-फ्लॉप का उपयोग रिपल काउन्टर (ripple counter) में किया जाता है।

5.8 काउन्टर (Counter)

(UPBTE 2012, 13,

- डिजिटल सर्किट में पल्सों को काउन्ट करने वाली युक्ति को काउन्टर कहते हैं। यह एक प्रकार का सिक्वैन्शियल सर्किट है।

- काउन्टर कुछ फ्लिप-फ्लॉप का समूह होता है जिसमें क्लॉक सिग्नल एप्लाई की जाती है।
- काउन्टर क्लॉक पल्सों की गणना करता है। अतः हम कह सकते हैं कि इसका आगे उपयोग आवृत्ति तथा आवर्तकाल के मापन में हो सकता है।

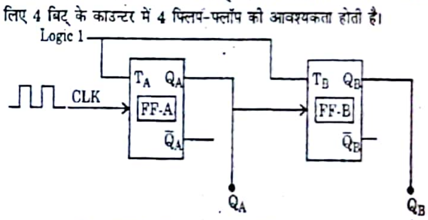
5.9 काउन्टर के प्रकार (Types of Counter)

(UPBTE 2015, 16)

- काउन्टर दो प्रकार के होते हैं–
  1. एसिन्क्रोनस या रिपल काउन्टर (Asynchronous or Ripple Counter)
  2. सिन्क्रोनस काउन्टर (Synchronous Counter)

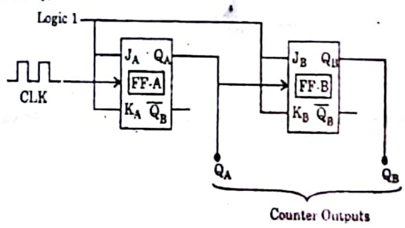
5.9.1 एसिन्क्रोनस काउन्टर (रिपल काउन्टर)

- एसेन्क्रोनस काउन्टर में बाहरी क्लॉक सिग्नल केवल एक ही फ्लिप-फ्लॉप पर एप्लाई की जाती है तथा उस फ्लिप-फ्लॉप के आउटपुट को दूसरे फ्लिप-फ्लॉप के इनपुट पर एप्लाई किया जाता है।
- चित्र में 2 बिट् का रिपल काउन्टर दर्शाया गया है। यहाँ हम 2 बिट् के काउन्टर की बात कर रहे हैं, अतः इसमें 2 फ्लिप-फ्लॉप का उपयोग होगा।
- इन प्रकार हम कह सकते हैं कि काउन्टर में जितने बिट्स होंगे उतने फ्लिप-फ्लॉप की आवश्यकता काउन्टर में होगी। उदाहरण के लिए 4 बिट् के काउन्टर में 4 फ्लिप-फ्लॉप की आवश्यकता होती है।



चित्र : 5.15 बिट् एसिन्क्रोनस काउन्टर यूजिंग *T*-फ्लिप-फ्लॉप

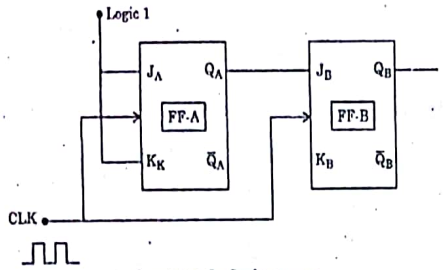
- चित्र में Toggle (*T*) फ्लिप-फ्लॉप का उपयोग करते हुए काउन्टर बनाया गया है। *T* फ्लिप-फ्लॉप की जगह *JK* फ्लिप-फ्लॉप को भी उपयोग करते हुए भी काउन्टर डिजाइन किया जा सकता है जो नीचे चित्र में दर्शाया गया है।
- चित्र में दोनों फ्लिप-फ्लॉप के *J* एवं *K* इनपुट को Logic 1 से जोड़ा गया है तथा दोनों आउटपुट को क्रमशः Q_A एवं Q_B में दर्शाया गया है।



चित्र : 5.16 बिट् एसिन्क्रोनस काउन्टर यूजिंग *J-K*-फ्लिप-फ्लॉप

5.9.2 सिन्क्रोनस काउन्टर (Synchronous Counter)

- सिन्क्रोनस काउन्टर में बाहरी क्लॉक पल्स सभी फ्लिप-फ्लॉप पर एक समय पर एप्लाई की जाती है।
- रिंग काउन्टर सिक्रोनस काउन्टर का एक उदाहरण है।
- नीचे चित्र में 2 बिट् का सिन्क्रोनस काउन्टर *JK*-फ्लिप-फ्लॉप का उपयोग करके डिजाइन किया गया है।
- सिन्क्रोनस काउन्टर में सभी फ्लिप-फ्लॉप पर क्लॉक पल्स एक साथ एप्लाई की जाती है जिससे प्रोपेगेशन डिले (propagation delay) की समस्या कम हो जाती है।



चित्र : 5.17 2 बिट् सिन्क्रोनस काउन्टर

- प्रोपेगेशन डिले में कमी के कारण सिन्क्रोनस काउन्टर, एसिन्क्रोनस काउन्टर की तुलना में अधिक क्लॉक आवृत्ति पर ऑपरेट होता है।

5.10 काउन्टर का मॉडयुलस (Modulus of Counter)

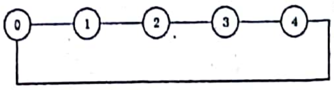
- 2 बिट् रिपल काउन्टर को MOD-4 काउन्टर कहते हैं तथा 3 बिट् रिपल काउन्टर को MOD-8 काउन्टर कहते हैं।
- इस प्रकार *n*-बिट् रिपल काउन्टर को मॉडुलो-*N* काउन्टर कहते हैं,

  जहाँ MOD number = 2^n
- काउन्टर का माडुलस उस स्टेट की संख्या को व्यक्त करता है जिससे काउन्टर अपने ऑपरेशन को प्रोग्रेस करता है।

उदाहरण— *माडुलो-5 काउन्टर को डिजाइन कीजिए।*

हल—मॉडुलो-5 काउन्टर को डिजाइन करने के लिए 3 बिट् रिपल काउन्टर का प्रयोग करते हैं जिसके लिए निम्न steps का प्रयोग करते हैं—

Step 1 : MOD-5 काउन्टर का स्टेट डायग्राम खींचना।



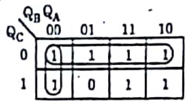
चित्र : 5.18 State Diagram of MOD-5 Ripple Counter

Step 2 : रिसेट (reset) लॉजिक का Truth Table खींचना।

- सत्य तालिका में 0 से 4 तक के स्टेट वैलिड (valid) स्टेट हैं तथा रिसेट लॉजिक का आउटपुट उनके लिए Inactive (1) है।

State	Flip-Flop Output			Output (Y)	
	Q_C	Q_B	Q_A		
0	0	0	0	1	Valid States
1	0	0	1	1	
2	0	1	0	1	
3	0	1	1	1	
4	1	0	0	1	
5	1	0	1	0	Invalid States
6	1	1	0	0	
7	1	1	1	0	

- स्टेट 5, 6 तथा 7 इनवैलिड स्टेट हैं।



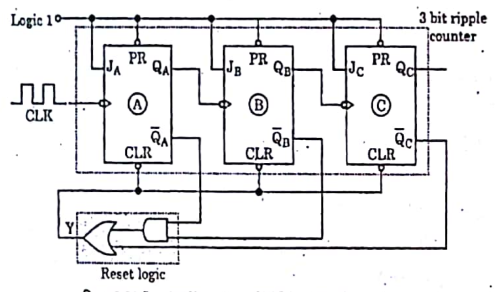
चित्र : 5.19

Step 3 : आउटपुट *Y* को सरल करना।

$$Y = \bar{Q}_C + \bar{Q}_B + \bar{Q}_A$$

Step 4 : लॉजिक डायग्राम

- MOD-5 रिपल काउन्टर का लॉजिक परिपथ चित्र में दर्शाया गया है।



चित्र : 5.20 Logic diagram of MOD-5 ripple counter

Scanned by CamScanner

5.11 सिन्क्रोनस एवं एसिन्क्रोनस काउन्टर में तुलना (Comparison of Synchronous and Asynchronous Counters)

क्र० सं०	पैरामीटर	एसिन्क्रोनस काउन्टर	सिन्क्रोनस काउन्टर
1.	परिपथ	सरल	जटिल
2.	कनेक्शन पैटर्न	पहले फ्लिप-फ्लॉप का आउटपुट दूसरे फ्लिप-फ्लॉप के लिए क्लॉक पल्स का कार्य करता है।	पहले फ्लिप-फ्लॉप का आउटपुट एवं दूसरे फ्लिप-फ्लॉप की क्लॉक में कोई कनेक्शन नहीं होता।
3.	क्लॉक इनपुट	सभी फ्लिप-फ्लॉप पर एक साथ क्लॉक इनपुट नहीं दी जाती।	सभी फ्लिप-फ्लॉप पर एक साथ क्लॉक इनपुट प्राप्त करते हैं।
4.	प्रोपेगेशन डिले	अधिक	कम
5.	फ्रीक्वेंसी ऑफ ऑपरेशन	अधिक प्रोपेगेशन डिले के कारण कम होती है।	कम प्रोपेगेशन डिले के कारण अधिक होती है।

5.12 काउन्टर एप्लीकेशन (Counter Application)

- विभिन्न प्रकार के काउन्टर्स के कई उपयोग हैं जो नीचे दर्शाये गये हैं—

1. फ्रीक्वेंसी काउन्टर में
2. डिजिटल क्लॉक में
3. डिजिटल वोल्टमीटर में
4. एनलॉग से डिजिटल कनवर्टर में
5. डिजिटल डिवाइडर सर्किट में
6. फ्रीक्वेंसी डिवाइडर सर्किट में
7. टाइम मेजरमेन्ट में

5.13 शिफ्ट रजिस्टर्स (Shift Registers)

- क्लॉक पल्स के कारण रजिस्टर्स में बाइनरी डाटा एक फ्लिप-फ्लॉप से दूसरे फ्लिप-फ्लॉप में मूव करते हैं। वह रजिस्टर जिसमें इस प्रकार का डाटा ट्रान्सफर होता है, शिफ्ट रजिस्टर्स (shift register) कहलाता है।
- शिफ्ट रजिस्टर का उपयोग डाटा को स्टोर करने में, डाटा को ट्रांस्फर करने में तथा कुछ अर्थमैटिक एवं लॉजिक (logic) ऑपरेशन में होता है।
- रजिस्टर कुछ फ्लिप-फ्लॉप का समूह होता है। प्रत्येक फ्लिप-फ्लॉप एक कॉमन (common) क्लॉक को शेयर करते हैं तथा 1 बिट को स्टोर करते हैं।
- n-बिट रजिस्टर में n-फ्लिप-फ्लॉप होते हैं तथा यह n-बिट सूचना को स्टोर करते हैं।

S.No.	Mode of operation	Illustration	Comment
1.	Serial input serial output (serial shift right)	I/P → FF3 FF2 FF1 FF0 → O/P	Data bits shift from left to right by 1 position per clock cycle.
2.	Serial input serial output (serial shift left)	O/P ← FF3 FF2 FF1 FF0 ← I/P	Data bits shift from right to left by 1 position per clock.
3.	Serial input parallel output.	I/P → FF3 FF2 FF1 FF0, Outputs	All output bits are made available simultaneously after 4-clock pulses.
4.	Parallel input serial output.	Inputs, FF3 FF2 FF1 FF0 → O/P	All inputs are loaded simultaneously but output bit-by-bit.

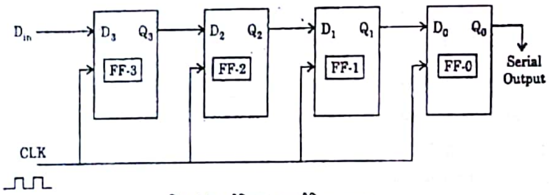
5.14 मोड ऑफ ऑपरेशन (Mode of Operation)

- शिफ्ट रजिस्टर को ऑपरेट करने के विभिन्न मोड निम्नवत् हैं—

1. सीरियल इनपुट सीरियल आउटपुट (Serial Input Serial Output)
2. सीरियल इनपुट पैरलल आउटपुट (Serial Input Parallel Output)
3. पैरलल इनपुट सीरियल आउटपुट (Parallel Input Serial Output)
4. पैरलल इनपुट पैरलल आउटपुट (Parallel Input Parallel Output)

5.14.1 Serial Input Serial Output (SISO)

- सीरियल इनपुट सीरियल आउटपुट टाइप का शिफ्ट रजिस्टर चित्र में दर्शाया गया है।

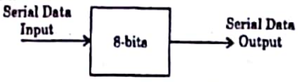


चित्र : 5.21 सीरियल इनपुट सीरियल आउटपुट

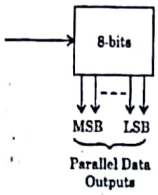
- मान लीजिए प्रारम्भ में सभी फ्लिप-फ्लॉप रिसेट अवस्था में हैं, *i.e.*, $Q_3 = Q_2 = Q_1 = Q_0 = 0$
- इसके बाद हमने एक 4 बिट् की बाइनरी संख्या 1111 को रजिस्टर के इनपुट पर एप्लाई किया।
- बाइनरी संख्या का एक-एक बिट् फ्लिप-फ्लॉप 3, फ्लिप-फ्लॉप 2, फ्लिप-फ्लॉप 1 तथा फ्लिप-फ्लॉप 0 से होता हुआ शिफ्ट हो जाएगा।
- यहाँ दिया गया इनपुट फ्लिप-फ्लॉप के साथ सीरियल में जोड़ा गया है, अत: इसे सीरियल शिफ्ट रजिस्टर भी कहते हैं।
- शिफ्ट रजिस्टर के ऑपरेशन के विभिन्न मोड को चित्र की सहायता से दर्शाया गया है।

Scanned by CamScanner

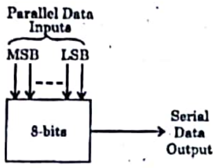
(a) Serial In Serial Out



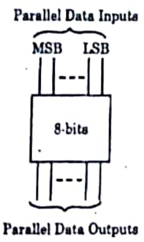
(b) Serial In Parallel Out



(c) Parallel In Serial Out

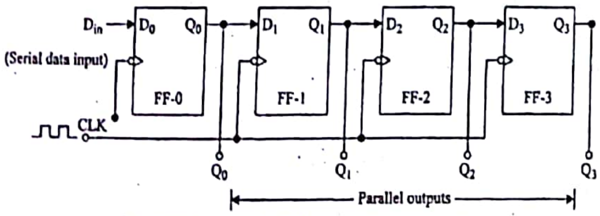


(d) Parallel In Parallel-Out



5.14.2 Serial Input Parallel Output (SIPO)

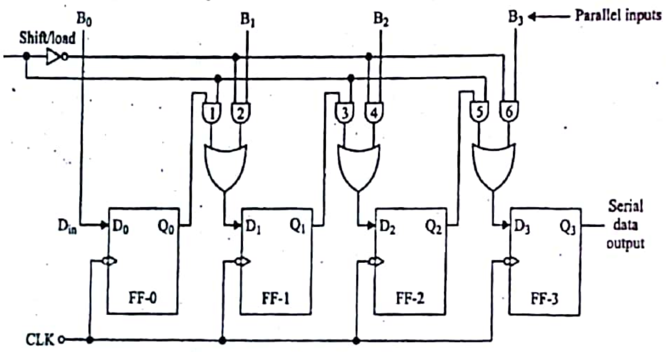
SIPO में डाटा को सीरियल से प्रवेश कराया जाता है परन्तु हमें पैरलल डाटा आउटपुट प्राप्त होता है। SIPO का स्पीड ऑफ ऑपरेशन SISO की भाँति होता है।



चित्र : 5.22 Illustration of serial input parallel output mode.

5.14.3 Parallel In Serial Out (PISO)

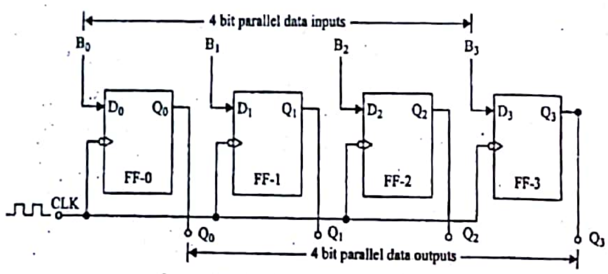
PISO में डाटा को पैरलल में इनपुट किया जाता है। यहाँ हमें प्राप्त आउटपुट डाटा सीरियल में मिलता है।



चित्र : 5.23 Parallel in serial out shift register.

5.14.4 Parallel In Parallel Out (PIPO)

नीचे चित्र में Parallel In Parallel Out मोड को दिखाया गया है।



चित्र : 5.24 Parallel in parallel out shift register.

PIPO में डाटा को पैरलल में प्रवेश कराते है तथा आउटपुट डाटा पैरलल में प्राप्त होता है।

5.15 शिफ्ट रजिस्टर के एप्लीकेशन (Application of Shift Registers)

- शिफ्ट रजिस्टर के अनुप्रयोग निम्नवत् है–
  1. टेम्पररी डाटा को स्टोर करने में
  2. सीरियल से पैरलल कनवर्टर में
  3. पैरलल से सीरियल कनवर्टर में

Scanned by CamScanner

4. रिंग काउन्टर में
5. डिले लाइन में
6. डाटा रिसिविंग में।

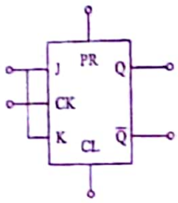
प्रश्नावली (Exercise)

1. काउन्टर और रजिस्टर के अनुप्रयोगों के बारे में बताइये। (UPBTE 2012)
2. शिफ्ट रजिस्टर की कार्य विधि के बारे में व्याख्या कीजिए। (UPBTE 2011)
3. निम्नलिखित को सत्य तालिका बनाकर समझाइये– (UPBTE 2010)
   (a) D-Flip-Flop
   (b) S-R Flip-Flop
   (c) J-K Flip-Flop
   (d) T-Flip-Flop
4. Combinational तथा Sequential Circuits से आप क्या समझते है?
5. फ्लिप-फ्लॉप को समझाइये। (UPBTE 2014)
6. क्लॉक सिग्नल का सचित्र वर्णन करें?
7. फ्लिप-फ्लॉप क्या है? तथा S-R फ्लिप-फ्लॉप की सभी स्थितियों का वर्णन कीजिये।
8. D फ्लिप-फ्लॉप को विस्तार से समझाइये।
9. J K फ्लिप-फ्लॉप क्या है? J K Master-Slave फ्लिप-फ्लॉप को समझाइए।
10. टॉगल फ्लिप-फ्लॉप (T-Flip-Flop) का सत्य तालिका द्वारा समझाइये।
11. काउन्टर किसे कहते है?
12. काउन्टर कितने प्रकार के होते है? प्रत्येक काउन्टर का वर्णन कीजिये। (UPBTE 2012)
13. रिपल काउन्टर से आप क्या समझते है?
14. काउन्टर के मॉडुलस को उदहारण सहित समझाइये। तथा सभी Steps का वर्णन कीजिये।
15. Synchronous तथा Asynchronous काउन्टर में अन्तर स्पष्ट कीजिए। (UPBTE 2015, 16)
16. Shift Register से आप क्या समझते है? इसके Mode of Operation का वर्णन कीजिए।

बहुविकल्पीय प्रश्न

1. रिपल काउन्टर ________ के समान कार्य करता है—
   (a) $2^n:1$ काउन्टर (b) 2 : 1 काउन्टर (c) 2 n :1 काउन्टर (d) n : 1 काउन्टर
2. Race condition उत्पन्न होती है—
   (a) कॉम्बिनेशन सर्किट में (In combinational circuits)
   (b) एसिन्क्रोनस सर्किट में (In asynchronous circuits)
   (c) सिन्क्रोनस सर्किट में (In synchronous circuits)
   (d) डिजिटल सर्किट में (In digital circuits)

3. CMOS इनवर्टर में—
   (a) एक ट्रांजिस्टर N चैनल तथा एक P चैनल होता है।
   (b) दोनों ऐनहैन्समेन्ट टाइप होते हैं।
   (c) दोनों N चैनल होते है परन्तु एक ऐनहैन्समेन्ट टाइप तथा दूसरा डिप्लीशन टाइप होता है।
   (d) एक ऐनहैन्समेन्ट टाइप तथा दूसरा डिप्लीशन टाइप होता है।
4. एस्टेबिल (astable) मल्टीवाइब्रेटर की .... होती है—
   (a) 2 स्थायी अवस्थाएं (2-stable states) (b) 1 स्थायी तथा 1 quasistable states
   (c) 2 quasistable states (d) इनमें से कोई नहीं।
5. डिप्लीशन टाइप तथा ऐनहैन्समेन्ट टाइप NMOS में मुख्य अन्तर है—
   (a) डिप्लीशन टाइप में एक +V थ्रेशोल्ड होता है।
   (b) डिप्लीशन टाइप में एक V_P निगेटिव होता है।
   (c) ऐनहैन्समेन्ट टाइप को दोनों मोड्स में प्रचालित किया जा सकता है।
   (d) डिप्लीशन टाइप को दोनों मोड्स में प्रचालित किया जा सकता है।
6. निम्न परिपथ में प्राप्त होता है—



   (a) मास्टर-स्लेव एक्शन (b) डिले गेट
   (c) टॉगल स्विच (d) इनमें से कोई नहीं।
7. फ्लिप-फ्लॉप एक .... होता है—
   (a) 1-bit स्टोरेज सैल (b) 2- bit स्टोरेज सैल
   (c) 3- bit स्टोरेज सैल (d) 4- bit स्टोरेज सैल
8. D फ्लिप-फ्लॉप तथा T फ्लिप-फ्लॉप का उपयोग क्रमशः .... के समान होता है—
   (a) टॉगल स्विच तथा डिले गेट (b) डिले गेट एव टॉगल स्विच
   (c) दोनों डिले गेट के समान (d) दोनों टॉगल स्विच के समान
9. सर्वाधिक तीव्र (fastest A/D) कनवर्टर है—
   (a) कम्पेरेटर कनवर्टर (b) काउन्टर टाइप कनवर्टर
   (c) सक्सैसिव एप्रॉक्सिमेशन टाइप कनवर्टर (d) डुएल स्लोप कनवर्टर।
10. 60 पल्स काउन्टर करने के लिए प्रयुक्त काउन्टर में फ्लिप-फ्लॉप की संख्या होगी—
   (a) 4 (b) 6 (c) 8 (d) 10

Scanned by CamScanner

11. एस्टेबिल मल्टी वाइब्रेटर की होती है—
(a) 2 स्टेबिल स्टेट्स (b) 1 स्टेबिल स्टेट्
(c) 2 quasistable स्टेट्स (d) इनमें से कोई नहीं।

12. एक SR फ्लिप-फ्लॉप में $S = R = 1$ है। फ्लिप-फ्लॉप की आउटपुट होगी—
(a) 1 (b) 0
(c) Infinite (d) Ambiguous or indeterminate

13. यदि एक JK फ्लिप-फ्लॉप में K इनपुट, J इनपुट का इनवर्जन है तब परिणामी फ्लिप-फ्लॉप होगा—
(a) SR फ्लिप-फ्लॉप (b) JK फ्लिप-फ्लॉप
(c) D फ्लिप-फ्लॉप (d) T फ्लिप-फ्लॉप

14. बाइनरी काउन्टर डिजाइन करने के लिए उपयुक्त फ्लिप-फ्लॉप है—
(a) D टाइप (b) SR टाइप (c) लैच (d) JK टाइप

15. एक-डिग्री JK फ्लिप-फ्लॉप द्वारा निर्मित रिपल काउन्टर में .... पर पल्स इनपुट दी जाती है—
(a) सभी फ्लिप-फ्लॉप के क्लॉक इनपुट पर
(b) एक फ्लिप-फ्लॉप के क्लॉक इनपुट पर
(c) एक फ्लिप-फ्लॉप के J तथा K इनपुट पर
(d) सभी फ्लिप-फ्लॉप के J तथा K इनपुट पर

16. एक JK फ्लिप-फ्लॉप की आउटपुट उस समय 'toggle' करती है, जब—
(a) $J = 1 ; K = 0$ (b) $J = 0 ; K = 1$ (b) $J = 1 ; K = 1$ (d) $J = 0 ; K = 0$

17. एक इलेक्ट्रॉनिक घड़ी (clock) की क्लॉक फ्रीक्वेन्सी 32 kHz है। इस फ्रीक्वेन्सी को डिवाइड कर 1Hz तक लाने के लिए आवश्यक है—
(a) दो डिकेड काउन्टर्स, एक 2-bit बाइनरी काउन्टर तथा एक T फ्लिप-फ्लॉप
(b) तीन डिकेड काउन्टर्स, एक 4-bit बाइनरी काउन्टर तथा एक T फ्लिप-फ्लॉप
(c) एक 4-bit बाइनरी काउन्टर तथा एक T फ्लिप-फ्लॉप तथा एक डिकेड काउन्टर
(d) चार 4-bit बाइनरी काउन्टर तथा एक डिकेड काउन्टर

18. निम्न में सीक्वेन्शियल सर्किट है—
(a) AND गेट (b) NAND गेट
(c) बाइस्टेबिल मल्टी वाइब्रेटर (d) Ex-OR गेट

उत्तर (Answers)

1. (a), 2. (b), 3. (b), 4. (c), 5. (c), 6. (c), 7. (a), 8. (b), 9. (b), 10. (a), 11. (b), 12. (c), 13. (d), 14. (c), 15. (a), 16. (b), 17. (b), 18. (b).

❑

अध्याय 6

मल्टीवाइब्रेटर्स तथा A/D, D/A कनवर्टर्स (Multivibrators and A/D, D/A Converters)

6.1 मल्टीवाइब्रेटर (Multivibrators)

- मल्टीवाइब्रेटर एक इलेक्ट्रॉनिक सर्किट है जो स्क्वायर वेव (square wave) को जनरेट करता है।
- मल्टीवाइब्रेटर एक स्विचिंग परिपथ है जो पॉजिटिव फीडबैक (positive feedback) के ऑपरेशन पर निर्भर करता है।
- मल्टीवाइब्रेटर निम्न तीन प्रकार के होते है—
  1. *एस्टेबिल या फ्री रनिंग मल्टीवाइब्रेटर (Astable or Free Running Multivibratlor)*
  2. *मोनोस्टेबिल या वन-शॉट मल्टीवाइब्रेटर (Mono stable or One-shot Multivibrator)*
  3. *बाइस्टेबिल या फ्लिप-फ्लॉप मल्टीवाइब्रेटर (Bistable or Flip-Flop Multivibrator)*

6.2 एस्टेबिल मल्टीवाइब्रेटर (Astable Multivibrator)

- वह मल्टीवाइब्रेटर जो बिना बाह्य ट्रिगरिंग पल्स (triggering pulse) के स्क्वायर वेव (square wave) जेनरेट करता है, एस्टेबिल मल्टीवाइब्रेटर कहलाता है।
- एस्टेबिल मल्टीवाइब्रेटर में कोई स्टेबिल स्टेट नहीं होती। इसमें दो क्वासी स्टेबिल (quasi-stable) स्टेट होते है।

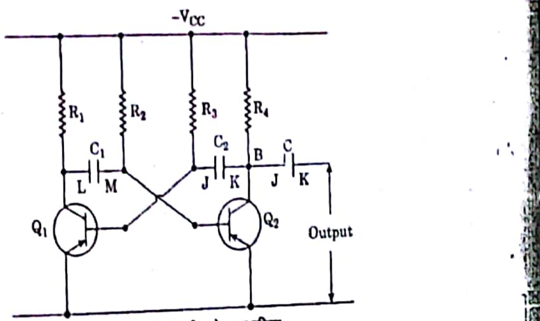
चित्र : 6.1 Astable Multivibrator

- एस्टेबिल मल्टीवाइब्रेटर लगातार स्क्वायर वेव (square wave) जेनरेट करता है।

कार्यप्रणाली (Principle of Operation)

- नीचे चित्र में दो ट्रांजिस्टर Q_1 तथा Q_2 का प्रयोग करते हुए एस्टेबिल मल्टीवाइब्रेटर को दर्शाया गया है।
- मल्टीवाइब्रेटर के सर्किट में दो CE एम्पलीफायर है जो एक-दूसरे के लिए फीडबैक का कार्य करते है।
- CE एम्पलीफायर के दोनों स्टेज बराबर है जिसके कारण $R_1 = R_4$ तथा बायसिंग रजिस्टर $R_2 = R_3$.
- ट्रांजिस्टर Q_1 का आउटपुट Q_2 के इनपुट के साथ तथा Q_2 का आउटपुट Q_1 के साथ युग्मित (coupled) होता है।

Scanned by CamScanner



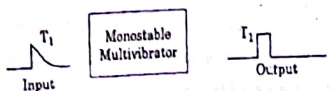
चित्र : 6.2 एस्टेबिल मल्टीवाइब्रेटर का परिपथ

- जब V_{cc} को सप्लाई जाता है तो कलेक्टर करेन्ट Q_1 तथा Q_2 में फ्लो होती है जिसके कारण कैपेसिटर C_1 तथा C_2 चार्ज होना प्रारम्भ कर देता है।
- माना प्रारम्भ में ट्रांजिस्टर Q_1 में अधिक चालन होता है जिसके कारण इसकी कलेक्टर वोल्टेज कम होने लगती है।
- Q_1 के कलेक्टर पर इस गिरती हुई वोल्टेज का निगेटिव सिग्नल (negative signal) कैपेसिटर C_1 के द्वारा Q_2 के बेस को प्राप्त होता है जिससे Q_2 कट-ऑफ (cut-off) पर आ जाता है तथा Q_1 की कलेक्टर वोल्टेज बढ़कर V_{cc} के लगभग बराबर हो जाती है।
- Q_2 के कलेक्टर पर बढ़ती हुई पॉजिटिव वोल्टेज कैपेसिटर C_1 के द्वारा ट्रांजिस्टर Q_1 के बेस पर पहुँचती है जिससे Q_1 सैचुरेशन (saturation) की स्थिति में आ जाता है।
- कैपेसिटर C_1 के अनावेशित क्रिया प्रारम्भ होने पर Q_2 की बेस एमीटर संधि फॉरवर्ड बायस में आ जाती है तथा Q_2 में चालन प्रारम्भ हो जाता है।
- इस क्रिया के बाद Q_1 कट-ऑफ (cut-off) अवस्था में, Q_2 चालन अवस्था में तब तक रहता है जब तक C_2 अनावेशित नही हो जाता।
- यह क्रिया निरन्तर चलती रहती है तथा हमें वर्गाकार तरंग (square wave) प्राप्त होती रहती है।

6.3 मोनोस्टेबिल मल्टीवाइब्रेटर (Monostable Multivibrator)

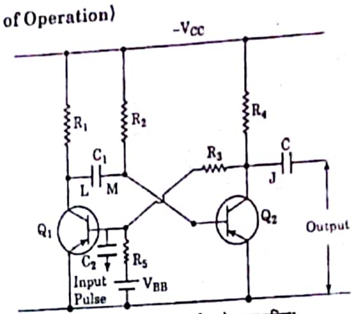
(UPBTE 2012)

- वह मल्टीवाइब्रेटर जिसका एक ट्रांजिस्टर ON स्टेट मे होता है तथा दूसरा OFF स्टेट में होता है, मोनोस्टेबिल मल्टीवाइब्रेटर कहलाता है।
- मोनोस्टेबिल मल्टीवाइब्रेटर मे एक क्वासी-स्टेबल (half stable) स्टेट होता है।
- मोनोस्टेबिल मल्टीवाइब्रेटर अपने प्रत्येक इनपुट ट्रिगर पल्स के लिए एक सिंगल आउटपुट देता है।



चित्र : 6.3 Monostable Multivibrator

कार्य प्रणाली (Principle of Operation)

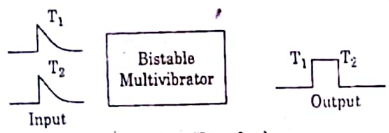


चित्र : 6.4 मोनोस्टेबिल मल्टीवाइब्रेटर का परिपथ

- मोनोस्टेबिल मल्टीवाइब्रेटर का परिपथ चित्र 6.4 में दर्शाया गया है जिसमें दो ट्रांजिसटर Q_1 तथा Q_2 प्रयुक्त किए गए हैं।
- परिपथ स्टेबिल अवस्था में तब तक रहता है जब तक ट्रांजिस्टर Q_1 OFF हो तथा Q_2 ON हो।
- ट्रांजिस्टर Q_1 के बेस को कोई इनपुट पल्स देने पर परिपथ स्थायी अवस्था मे नही रहता।
- ट्रांजिस्टर Q_1 के बेस को कोई इनंपुट पल्स देने पर Q_2 चालन अवस्था मे आ जाती है तथा Q_2 की बेस बायस धीरे-धीरे कम होने लगती है। अतः ट्रांजिस्टर Q_2 की कलेक्टर धारा कम होने लगती है तथा Q_2 लगभग कट-ऑफ (cut-off) अवस्था की ओर बढ़ने लगता है।
- इस क्रिया के बाद ट्रांजिस्टर Q_1 की फॉरवर्ड बायस अधिक हो जाती है तथा वह ON स्टेट मे आ जाता है। इस समय परिपथ अपनी Quasi-stable स्टेट में होता है।
- परिपथ की यह अवस्था एक निश्चित समय लिए होती है तथा कुछ समय पश्चात् परिपथ अपने पूर्व स्टेबिल स्टेट में आ जाता है।
- परिपथ इस स्थिति में तब तक रहता है जब तक कोई अन्य इनपुट पल्स ट्रांजिस्टर (Q_1) के बेस को नहीं दी जाती।

6.4 बाइस्टेबिल मल्टीवाइब्रेटर (Bistable Multivibrator)

(UPBTE 2015, 16).



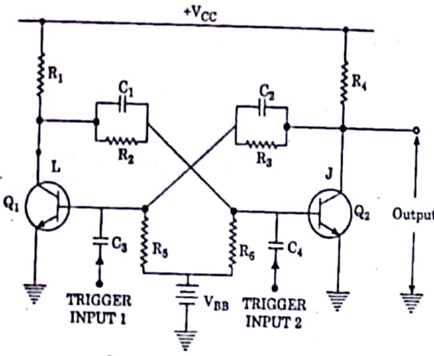
चित्र : 6.5 बाइ-स्टेबिल मल्टीवाइब्रेटर

- वह मल्टीवाइब्रेटर जिसके पास दो स्टेबिल स्टेट (stable state) होते हैं, बाइस्टेबिल मल्टीवाइब्रेटर कहलाता है।
- पहले इनपुट पल्स का प्रयोग स्क्वायर वेव (square wave) के अर्धचक्र (half cycle) को जेनरेट करने में होता है तथा दूसरी इनपुट पल्स का प्रयोग स्क्वायर वेव के अगले अर्धचक्र को जेनरेट करने में होता है।

Scanned by CamScanner

कार्य-प्रणाली (Principle of Operation)

- बाइस्टेबिल मल्टीवाइब्रेटर का परिपथ चित्र 6.6 में दर्शाया गया है जिसमें दो ट्रांजिस्टर Q_1 तथा Q_2 प्रयोग किए गए हैं।
- ट्रांजिस्टर Q_1 का आउटपुट Q_2 के बेस को तथा Q_2 का आउटपुट Q_1 के बेस को पुनः निविष्ट (feedback) की गई है।



चित्र : 6.6 बाइस्टेबिल मल्टीवाइब्रेटर का परिपथ

- प्रवर्धकों का संयोजन इस प्रकार किया जाता है कि एक ट्रांजिस्टर सेचुरेशन में हो तथा दूसरा कट-ऑफ में हो।
- माना किसी समय ट्रांजिस्टर Q_1 तथा Q_2 के कलेक्टर में I_1 तथा I_2 धाराएं प्रवाहित हो रही हैं।
- यदि किसी समय I_1 में क्षणिक कमी होती है तब ट्रांजिस्टर Q_1 का कलेक्टर विभव बढ़ने लगता है। इससे Q_2 के बेस बायस में वृद्धि होगी तथा Q_1 का बेस-बायस कम हो जाएगा।
- यह क्रिया तब तक चलती है जब तक Q_2 सेचुरेशन में तथा Q_1 OFF अवस्था में नहीं आ जाते।
- बाइस्टेबिल मल्टीवाइब्रेटर की दो स्थायी अवस्था इस प्रकार हैं—

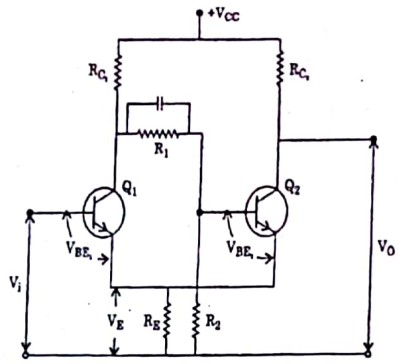
  1. ट्रांजिस्टर Q_1 "OFF" तथा Q_2 "ON"

  2. ट्रांजिस्टर Q_1 "ON" तथा Q_2 "OFF"

- बाइस्टेबिल मल्टीवाइब्रेटर सर्किट का वह समय जिसमें एक ट्रांजिस्टर से चालन दूसरे ट्रांजिस्टर को स्थानान्तरित होता है, ट्रांजिशन समय (transition time) कहलाता है।

6.5 श्मिठ ट्रिगर (Schmitt Trigger)

- श्मिट ट्रिगर एक स्विचिंग परिपथ है जो डिजिटल प्रणालियों में प्रयोग किया जाता है।
- श्मिट ट्रिगर की स्थायी अवस्था इनपुट वोल्टेज के आयाम पर निर्भर करती है।
- श्मिट ट्रिगर को एमिटर कपल्ड बाइनरी (emitter coupled binary) भी कहते हैं।



चित्र : 6.7 Schmitt Trigger

- श्मिट ट्रिगर धीमी गति से परिवर्तनशील तरंग को ऐसी वर्गाकार तरंग में बदलता है जिसका Rise तथा Fall समय बहुत कम होता है।
- इसकी एक स्थायी अवस्था होती है तथा इनपुट देने पर तेजी से दूसरी अवस्था में आ जाती है।

कार्य-प्रणाली (Principle of Operation)

- चित्र 6.7 में श्मिट ट्रिगर का परिपथ प्रदर्शित किया गया है।
- जब परिपथ को कोई इनपुट (V_i) नहीं दी जाती है तब ट्रांजिस्टर Q_1 "OFF" स्टेट में तथा ट्रांजिस्टर Q_2 "ON" में होता है।
- श्मिट ट्रिगर के परिपथ में इनपुट वोल्टेज के दो निश्चित मान Upper Trigger Potential (UTP) तथा Lower Trigger Potential (LTP) होते हैं जो परिपथ को एक स्थिर अवस्था से दूसरी अवस्था में स्विच करते हैं।
- UTP के लिए इनपुट वोल्टेज V_i का मान V_E से इतना अधिक होना चाहिये कि यह ट्रांजिस्टर Q_1 के एमिटर बेस जंक्शन को फॉरवर्ड बायस में ले आए तथा Q_1 में बेस धारा प्रवाहित होने लगे।
- इनपुट वोल्टेज V_i का मान, जिस पर परिपथ की अवस्था में परिवर्तन होता है, UTP के लिए अलग तथा LTP के लिए अलग होती है।
- श्मिट ट्रिगर में परिपथ की अवस्था परिवर्तन के लिए इनपुट की बढ़ती हुई स्थिति तथा गिरती हुई स्थिति का अन्तर Hysteresis कहलाता है।

6.6 मल्टीवाइब्रेटर तथा श्मिट ट्रिगर के अनुप्रयोग (Application of Multivibrator and Schmitt Trigger)

क्रम संख्या	परिपथ	विशेषता	अनुप्रयोग
1.	बाइस्टेबिल मल्टीवाइब्रेटर	दो स्थायी अवस्थाएं	मैमोरी, गणितीय ऑपरेशन
2.	मोनो स्टेबिल मल्टीवाइब्रेटर	एक स्थायी अवस्था तथा एक क्वासी-स्थायी अवस्था	वेव जेनरेटर, गेटिंग परिपथ

Scanned by CamScanner

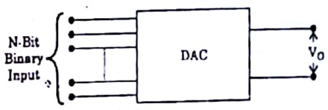
3.	एस्टेबिल मल्टीवाइब्रेटर	दो क्वासी-स्थायी अवस्था	स्क्वायर वेव जेनरेटर, दोलित्र
4.	श्मिट ट्रिगर	एक अस्थायी अवस्था तथा दूसरी अवस्था का नियंत्रण UTP तथा LTP	A/D कनवर्टर, पल्स काउन्टर

6.7 A/D तथा D/A कनवर्टर्स (A/D and D/A Converters)

- माइक्रोप्रोसेसर, इन्सट्रक्शन एवं डाटा को केवल बाइनरी फॉर्म (डिजिटल फार्म) में स्वीकार करता है तथा डिजिटल फॉर्म में एड्रेस और डाटा प्रदान करता है। इसके विपरीत कुछ सिग्नल एनेलॉग प्रकृति के होते हैं।
- अत: वास्तविक सिग्नलों को डिजिटल प्रोसेसिंग के लिए एनेलॉग सिग्नलों को डिजिटल सिग्नलों में परिवर्तित किया जाता है तथा डिजिटल प्रोसेसिंग के पश्चात् प्राप्त डिजिटल सिग्नलों को वास्तविक उपयोग में लाने के लिए एनेलॉग सिग्नल में परिवर्तित किया जाता है।
- एनेलॉग सिग्नलों से डिजिटल में परिवर्तित करने की क्रिया Analog to Digital Conversion कहलाती है तथा इस प्रयोजन के लिए प्रयुक्त प्रणाली Analog to Digital Converter (A/D कनवर्टर) कहलाती है.
- डिजिटल सिग्नलों से एनेलॉग में परिवर्तित की क्रिया Digital to Analog Conversion कहलाती है तथा इसके लिए प्रयुक्त प्रणाली Digital Analog Converter कहलाती है।

6.8 डिजिटल से एनेलॉग कनवर्टर्स (Digital to Analog Converters)

- डिजिटल से एनेलॉग कनवर्टर्स का मूल फंक्शन N-बिट् बाइनरी इनपुट को एनेलॉग आउटपुट में बदलना है।
- D/A कनवर्टर्स N : 1 मल्टीप्लेक्सर के समान होता है। अन्तर केवल यह है कि D/A Converter सभी डिजिटल इनुपट को एक साथ ही एक ही समय पर एनेलॉग में परिवर्तित कर देता है जबकि मल्टीप्लेक्सर में आउटपुट कन्ट्रोल लाइन पर डिपेन्ड करती है।



चित्र : 6.8 D/A Converters

6.9 डिजिटल से एनेलॉग में बदलने की विधियाँ (Digital to Analog Conversion Techniques)

- डिजिटल से एनेलॉग में परिवर्तित करने की दो विधियाँ है—

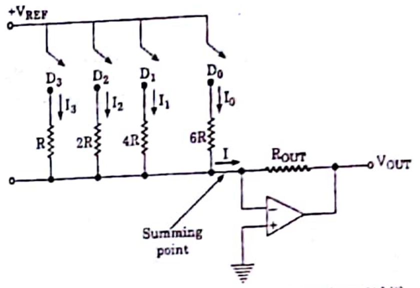
  1. परिवर्ती प्रतिरोधी D/A कनवर्टर

  2. R-2R लेडर D/A कनवर्टर

6.9.1 परिवर्ती प्रतिरोधी D/A कनवर्टर (Weighted Resistor D/A Converter)

- चित्र 6.9 में परिवर्ती प्रतिरोधी D/A कनवर्टर का परिपथ दर्शाया गया है जिसमें V_{REF} रैफ्रेन्स वोल्टेज को दर्शाता है।

- परिपथ में प्रयुक्त रजिस्टर प्रीसिजन रजिस्टर (precision resistor) है जिससे एकुरेट (accurate) इनपुट करेन्ट प्राप्त होता है।



चित्र : 6.9 D/A converter with binary weighted resistor (4-bit)

- परिपथ में प्रयुक्त स्विच खुली या बन्द दो स्थितियों में हो सकती है।
- जब सभी स्विचें खुली होंगी तब सभी इनपुट करेन्ट शून्य होंगी तथा जब सभी स्विचें बन्द होंगी तब Weighted Input Current द्वारा आउटपुट करेन्ट जनरेट होगा।
- सभी स्विचें बन्द होने की स्थिति मे $D_3 = D_2 = D_1 = D_0 = 1$ होगा।

इस समय

$$I_3 = \frac{V_{REF}}{R}$$

$$I_2 = \frac{V_{REF}}{2R}$$

$$I_1 = \frac{V_{REF}}{4R}$$

$$I_0 = \frac{V_{REF}}{8R}$$

- सभी स्विचों के बन्द होने की स्थिति में आउटपुट करेन्ट सभी इनपुट करेन्ट के योग के बराबर होता है।

$$I = I_0 + I_1 + I_2 + I_3$$

$$I = \frac{V_{REF}}{R}(1 + 0.5 + 0.25 + 0.125)$$

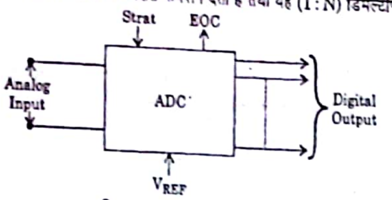
$$I = 1.875\frac{V_{REF}}{R}$$

- स्विचें को बन्द करने तथा खुले रखने से 16 विभिन्न आउटपुट करेन्ट प्राप्त होंगी जिनका मान 0 से 1.875 V_{REF}/R के बीच मे होगा।

Scanned by CamScanner

6.10 एनेलॉग से डिजिटल कनवर्टर्स (Analog to Digital Converters)

- एनेलॉग से डिजिटल कनवर्टर DAC के विपरीत फंक्शन देता है तथा यह (1 : N) डिमल्टीप्लेक्सर के समान होता है।



चित्र : 6.10 ADC Converters

- ADC में दो कन्ट्रोल लाइन होती है, START तथा EOC।
- START कन्ट्रोल लाइन ADC को कनवर्जन स्टार्ट होने की सूचना देता है तथा EOC (End of Conversion) कनवर्जन समाप्त होने की सूचना देता है।

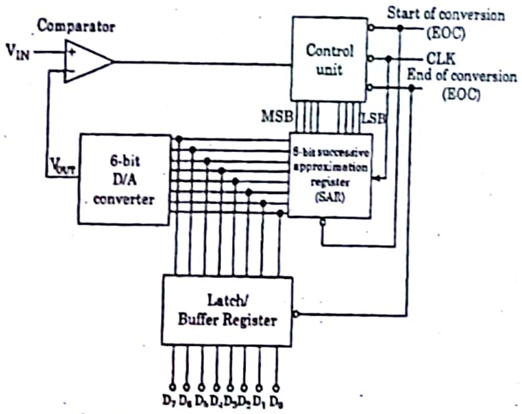
6.11 एनेलॉग से डिजिटल में बदलने की विधियाँ (Analog to Digital Conversion Techniques)

(UPBTE 2013, 14)

- एनेलॉग से डिजिटल में बदलनें की कुछ विधियाँ निम्न है–
  1. सर्वो विधि
  2. डिजिटल रैम्प A/D कनवर्टर
  3. सक्सेसिव एप्रॉक्सिमेशन A/D कनवर्टर
  4. सिग्नल स्लोप D/A कनवर्टर
  5. इन्टिग्रेटिंग A/D कनवर्टर
  6. डुअल स्लोप A/D कनवर्टर
  7. पैरलल A/D कनवर्टर

6.11.1 सक्सेसिव एप्रॉक्सिमेशन D/A कनवर्टर (Successive Approximation A/D Converter)

- A/D कनवर्टर की यह विधि सबसे अधिक प्रचलित है। इस विधि को Bit Weighting Conversion भी कहते हैं।



चित्र : 6.11 Successive approximation A/D converter

- प्रारम्भ में START पल्स कनवर्जन को स्टार्ट करता है तथा सभी बिट्स को क्लिअर (clear) करता है।
- सक्सेसिव एप्रॉक्सिमेशन कनवर्टर के तीन मुख्य भाग है–
  1. D/A Converter
  2. Successive Approximation Register (SAR)
  3. Comparator
- इस विधि द्वारा एनेलॉग को डिजिटल में परिवर्तन करने के लिए D/A कनवर्टर की आउटपुट को एनेलॉग इनपुट सिग्नल के साथ कम्पेयर (Compare) किया जाता है।
- जब D/A कनवर्टर की आउटपुट एनेलॉग सिग्नल से मैच कर जाती है तब D/A कनवर्टर की इनपुट डिजिटल सिग्नल के तुल्य होती है।
- Successive Approximation विधि में सबसे पहले MSB को Turn On किया जाता है तथा D/A कनवर्टर के आउटपुट की तुलना एनेलॉग इनपुट से करते है।
- यदि कम्परेटर की स्टेट बदलती है तब इसका अर्थ है कि MSB द्वारा जेनरेट की गई आउटपुट से अधिक है। इस समय SAR में पहले MSB को ऑफ करके दूसरे MSB को Turn On करते हैं। यह क्रिया तब तक चलती है जब तक इनपुट के लिए सभी LSB का प्रयोग नहीं किया जाता है। तब D/A कनवर्टर की आउटपुट एनेलॉग इनपुट के बराबर हो जाती है।
- इस प्रकार Successive Approximation क्रिया MSB से प्रारम्भ कर एक बार में केवल एक बिट् Turn On किया जाता है। सभी बिट्स के चेक (check) होने के बाद EOC आउटपुट अधिक हो जाती है तथा A/D कनवर्जन समाप्त हो जाती है।

प्रश्नावली (Exercise)

1. एक स्थायी मल्टीवाइब्रेटर परिपथ की कार्यविधि समझाइये। *(UPBTE 2012)*
2. Schmitt Trigger प्रवर्धक की कार्य-विधि के बारे में व्याख्या कीजिए। *(UPBTE 2011)*
3. बाइस्टेबल मल्टीवाइब्रेटर को समझाएं। *(UPBTE 2010)*
4. मल्टीवाइब्रेटर से आप क्या समझते है? *(UPBTE 2016)*
5. एस्टेबिल मल्टीवाइब्रेटर की कार्य प्रणाली को विस्तार से समझाएं।
6. मोनोस्टेबिल मल्टीवाइब्रेटर को समझाइए तथा इसकी कार्य-विधि की व्याख्या कीजिए। *(UPBTE 2012)*
7. बाइस्टेबिल मल्टीवाइब्रेटर को विस्तार से समझाइये। *(UPBTE 2015, 16)*
8. स्मिट ट्रिगर की कार्य-विधि का वर्णन कीजिए।
9. मल्टीवाइब्रेटर तथा स्मिट ट्रिगर के अनुप्रयोगों का वर्णन कीजिए।
10. Analog to Digital और Digital to Analog को विस्तार से समझाइये।
11. Digital से Analog में बदलने की विधियाँ बताइये।
12. Digital to Analog Converters से आप क्या समझते है?
13. Analog to Digital Converters से आप क्या समझते है? *(UPBTE 2013, 14)*
14. Analog to Digital Converters की विधियों को विस्तार से समझाएं।
15. सक्सेसिव एप्रॉक्सिमेशन A/D Converter क्या है?

Scanned by CamScanner

बहुविकल्पीय प्रश्न

1. एस्टेबिल मल्टीवाइब्रेटर में स्टेबिल स्टेट होते हैं—

   (a) 1 (b) 0 (c) 2 (d) 4

2. बाइस्टेबिल मल्टीवाइब्रेटर में स्टेबिल स्टेट होते हैं—

   (a) 1 (b) 0 (c) 2 (d) 4

3. मल्टीवाइब्रेटर का कार्य है—

   (a) एम्पलीफिकेशन करना (b) Square wave जेनरेट करना

   (c) सूचना को भेजना (d) प्रकाश का उत्सर्जन करना।

4. डिजिटल से एनेलॉग में बदलने के लिए प्रयोग होगा—

   (a) Successive approximation (b) Dual slope

   (c) Servo mechanism (d) R-2R Ladder

5. मोनोस्टेबिल में स्थायी अवस्था होगी—

   (a) 0 (b) 1 (c) 2 (d) 3

उत्तर (Answers)

1. (b), 2. (c), 3. (b), 4. (d), 5. (b).

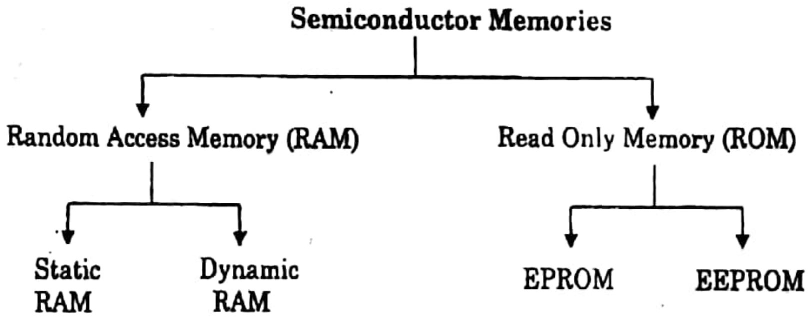
❑

Scanned by CamScanner

अध्याय 7 सेमीकन्डक्टर मेमोरीज (Semiconductor Memories)

7.1 सेमीकन्डक्टर मेमोरी (Semiconductor Memory)

- मेमोरी सर्किट मेमोरी सेल पर आधारित है तथा यह एक ऐसी डिवाइस है जिसका कार्य इन्फारमेशन बिट (Information Bit) को स्टोर करना है।
- मेमोरी सेल का लॉजिकल सिक्वेंस (Logical Sequence) मेमोरी सर्किट के कॉन्सेप्ट की जानकारी देता है।



7.2 Random Access Memory (RAM)

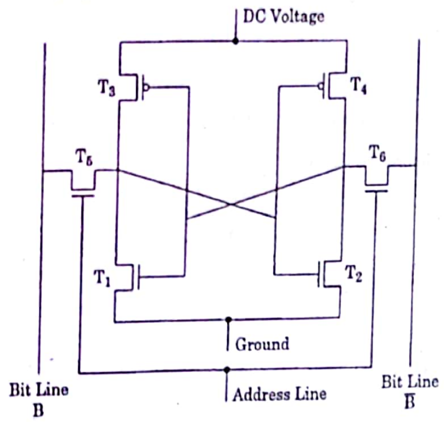
- RAM एक कॉमन टाइप मेमोरी है जिसको Rapid-Access Memory की भाँति उपयोग करते हैं तथा इसका कार्य मशीन कोड में कम्प्यूटर डाटा को स्टोर करना है।
- RAM एक वोलाटाइल टाइप (Volatile Type) मेमोरी है जिसमें Stored डाटा पावर ऑफ होने पर लॉस (Loss) हो जाता है। इसे Temporary Storage भी कहते हैं।
- RAM के मुख्यत: दो प्रकार हैं—
  (a) Static Random Access Memory (SRAM)
  (b) Dynamic Random Access Memory (DRAM)

7.2.1 Static Random Access Memory (SRAM)

- SRAM में डाटा लम्बे समय तक स्टोर रहता है जब तक पावर सप्लाई दी जाती है।
- इसमें जब पॉवर सप्लाई दी जाती है तो Refresh करने की आवश्यकता नहीं होती है।

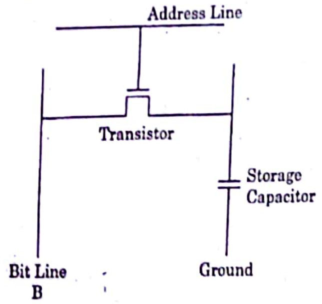
Scanned by CamScanner

- इसकी संरचना जटिल होती है तथा यह महँगा होता है।
- SRAM एक Faster Memory है तथा इसमें सर्किट को refresh करने की आवश्यकता नहीं होती।
- इसका उपयोग Flip-flop में होता है।



7.2.2 Dynamic Random Access Memory (DRAM)

- DRAM में लगातार Power Supply देने के बाद भी सर्किट को Refresh करने की आवश्यकता होती है।
- इसकी संरचना सरल होती है तथा यह Slower डिवाइस है।
- SRAM में कोई Charge लीक नहीं होता जबकि DRAM में Charge लीक होते हैं।



7.2.3 SRAM तथा DRAM की तुलना

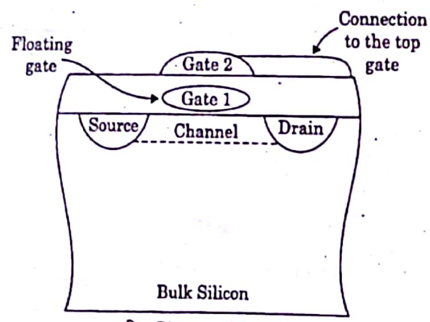
Basis for Comparison	SRAM	DRAM
Speed	Faster	Slower
Size	Large	Small
Cost	Expensive	Cheap
Used in	Cache Memory	Main Memory
Density	High	Less
Construction	Complex	Simple
Memory requires	6 Transistor	Only One Transistor
Charge Leakage	Not Present	Present
Power Consumption	Low	High

7.3 Read Only Memory (ROM)

- यह एक Non-volatile memory है जिसमें डाटा Power off होने पर भी Loss नहीं होता है। इसे Permanent Storage भी कहते हैं।
- ROM के मुख्यत: दो प्रकार है–
  (a) Erasable Programmable Read Only Memory (EPROM)
  (b) Electrically Erasable Programmable Read Only Memory (EEPROM)

7.3.1 Erasable Programmable Read Only Memory (EPROM)

- EPROM एक प्रोग्रामेबल मेमोरी है जो पावर ऑन होने पर भी डाटा को बरकरार रखती है।
- EPROM में डाटा को इरेज (Erase) करके फिर से उपयोग (reuse) में लाया जा सकता है।
- EPROM में डाटा को इरेज करने के लिए Ultraviolet Light (UV Light) का उपयोग करते हैं।



चित्र : Diagram of EPROM

Scanned by CamScanner

- EPROM टेस्टिंग एवं डिबगिंग (Testing and Debugging) के लिए Convenient होता है। इसमें किसी External Memory की आवश्यकता नहीं होती है।
- EPROM का स्टैटिक पावर कन्जपशन (Static Power Consumption) उच्च होता है तथा PROM की तुलना में इसकी लागत भी अधिक है।
- EPROM का अधिकतर उपयोग Personal Computers में होता है।

7.3.2 Electrically Erasable Programmable Read Only Memory (EEPROM)

- EEPROM एक Programmable Memory है जिसमें डाटा को इरेज (Erase) तथा Reprogramme करने के लिए इलेक्ट्रिकल वोल्टेज का उपयोग करते हैं।
- EPROM की भाँति EEPROM को मोडिफाई करने हेतु कम्प्यूटर से रिमूव नहीं करना होता है।
- EEPROM का एक विशेष रूप Flash Memory है जिसका उपयोग Personal Computer में डाटा को Erase तथा Reprogramming में होता है।
- EEPROM कम से कम 100000 टाइम डाटा को इरेज कर सकता है।

7.3.3 EPROM तथा EEPROM में अन्तर

EPROM	EEPROM
EPROM = Erasable Programmable Read Only Memory	EEPROM = Electrically Erasable Programmable Read Only Memory
इसमें डाटा को इरेज करने के लिए UV light का उपयोग करते हैं।	इसमें डाटा को इरेज करने के लिए Electrical Signal का उपयोग करते है।
कम्प्यूटर से डाटा को इरेज तथा रिप्रोग्रामिंग करने हेतु EPROM Chip को Remove किया जा सकता है।	इसमें चिप को Remove करने की आवश्यकता नहीं होती।
EPROM एक पुरानी टैक्नोलॉजी है।	EEPROM एक मॉडर्न टैक्नोलॉजी है।

प्रश्नावली (Exercise)

1. सेमीकण्डक्टर मेमोरी के टाइप का वर्णन कीजिए।
2. RAM तथा ROM में अन्तर बताइये।
3. SRAM का विस्तृत वर्णन करें।
4. Dynamic RAM का चित्र बनाकर व्याख्या कीजिए।
5. EPROM तथा EEPROM में अन्तर बताइये।
6. SRAM तथा DRAM की तुलना कीजिए।
7. EPROM की कार्यप्रणाली समझाइये।
8. वोलाटाइल तथा नॉन-वोलाटाइल मेमोरी को बताइये।

परिशिष्ट

Example 1. Represent the decimal number 532.86 in terms of powers of 10.

Sol. The required representation is shown in fig.

N = | 5 | 3 | 2 | · | 8 | 6 |

$$N = 5 \times 10^2 + 3 \times 10^1 + 2 \times 10^0 + 8 \times 10^{-1} + 6 \times 10^{-2}$$

Example 2. Represent the octal number 645 in power of 8 and find its decimal equivalent.

Sol. Step 1 : Representation in power of 8.

N = | 6 | 4 | 5 |

6×8^2 4×8^1 5×8^0

Step 2 : Decimal equivalent

$$N = (6 \times 8^2) + (4 \times 8^1) + (5 \times 8^0)$$

$$= (6 \times 64) + (4 \times 8) + (5 \times 1) = (421)_{10}$$ **Ans.**

Example 3. Represent the hexadecimal number 6DE in the power of 16 and obtain its decimal equivalent.

Sol. Step 1 : Representation in the power of 16.

N = | 6 | D | E |

6×16^2 13×16^1 14×16^0

Step 2 : Decimal equivalent

$$N = 6DE = (6 \times 16^2) + (13 \times 16^1) + (14 \times 16^0)$$

$$= (6 \times 256) + (13 \times 16) + (14 \times 1) = (1758)_{10}$$

$\therefore$ $(6DE)_{16} = (1758)_{10}$ **Ans.**

Example 4. Express the number 423.6 in powers of 7 and find its decimal equivalent.

Sol. $N = (423.6)$, *i.e.*, the base is 7.

Step 1 : Representation in powers of 7.

N = | 4 | 2 | 3 | · | 6 |

4×7^2 2×7^1 3×7^0 6×7^{-1}

Step 2 : Decimal equivalent

$$N = (423.6)_7 = (4 \times 7^2) + (2 \times 7^1) + (3 \times 7^0) + (6 \times 7^{-1})$$

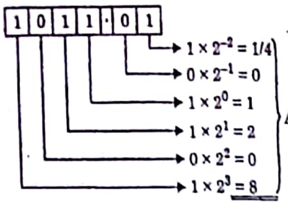
$$= (213.857)_{10}$$ **Ans.**

Example 5. Convert the binary number 1011.01 into its decimal equivalent.

Sol.

Scanned by CamScanner

Step 1, 2 and 3 :



Step 4 : Addition :

$\therefore \quad (1011.01)_2 = (11.25)_{10}$ **Ans.**

Example 6. Convert the octal number $(314)_8$ into its decimal equivalent.

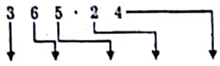
Sol.

Step 1 : Get the octal number :	3	1	4
Step 2 : Write corresponding weights :	8^2	8^1	8^0
Step 3 : Multiply (columnwise) :	192	8	4
Step 4 : Add contents of row-3	192 + 8 + 4 = 204		

$\therefore \quad (314)_8 = (204)_{10}$ **Ans.**

Example 7. Convert the octal number $(365.24)_8$ into its equivalent decimal number.

Sol.



Decimal number,

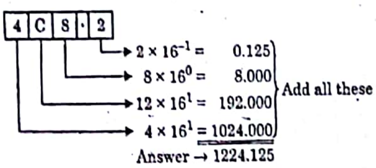
$$D = (3 \times 8^2) + (6 \times 8^1) + (5 \times 8^0) + (2 \times 8^{-1}) + (4 \times 8^{-2})$$

$$= 192 + 48 + 5 + 0.25 + 0.0625$$

$\therefore \quad D = 245.3125$ **Ans.**

Example 8. Convert the hex number $(4C8.2)_{16}$ into its equivalent decimal number.

Sol.



1. Given Hex Number :
2. Multiply each digit by its positional weight

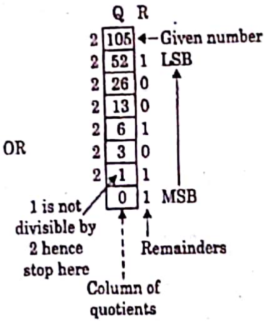
$\therefore \quad (4C8.2)_{16} = (1224.125)_{10}$ **Ans.**

Example 9. Convert $(105)_{10}$ to the equivalent binary number.

Sol. We divide the given number by the radix or base of binary system which is 2.

Step	Divide	Integer Quotient	Remainder
(1)	105/2 =	52	1 (LSB)
(2)	52/2 =	26	0
(3)	26/2 =	13	0
(4)	13/2 =	6	1
(5)	6/2 =	3	0
(6)	3/2 =	1	1
(7)	1/2 =	0	1 (MSB)

OR



$\therefore$ Binary Number = 1101001 (a) $\quad$ $\therefore$ Binary Number = 1101001 (b)

Decimal to binary conversion

Thus $\quad (105)_{10} = (1101001)_2$ **Ans.**

Example 10. Convert $(204)_{10}$ into its equivalent octal number.

Sol.

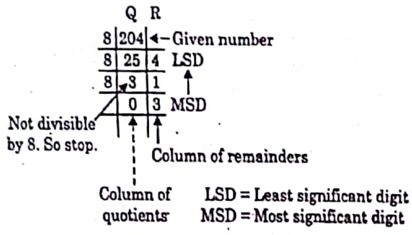
- We divide the decimal number by the radix (base) of octal system which is 8.
- The required conversion is as shown in Table P.1.12.6.

Divide	Integer Quotient	Remainder
204/8 =	25	4 (LSD)
25/8 =	3	1
3/8 =	0	3 (MSD)

Decimal to octal conversion

$\therefore \quad (204)_{10} = (314)_8$ **Ans.**

A simpler method, to carry out the decimal to octal conversion is illustrated in fig. P.1.12.6.



$\therefore \quad (204)_{10} = (314)_8$ **Ans.**

Example 11. Convert the decimal number 259 into its hex equivalent.

Sol. The conversion takes place as follows :

The base is 16, so we divide the given number by 16.

Scanned by CamScanner

Step	Divide	Integer Quotient	Remainder	
(1)	259/16 =	16	3	LSD
(2)	16/16 =	1	1	↑
(3)	1/16 =	0	0	MSD

OR

	Q	R	
16	259		
16	16	3	LSD
16	1	0	↑
	0	1	MSD

Number not divisible by 16 so stop; ↑ Remainder

$\therefore \quad (259)_{10} = (103)_H$ **Ans.**

Example 12. Convert the decimal number 35 into radix 5 equivalent.

Sol. The base is 5, so we will divide the given number by 5 as shown in fig.

Step	Divide	Integer Quotient	Remainder	
(1)	35/5 =	7	0	LSD
(2)	7/5 =	1	2	↑
(3)	1/5 =	0	1	MSD

	Q	R	
5	35		
5	7	0	LSD
5	1	2	↑
	0	1	MSD

Decimal to radix 5

Hence the answer is : $(35)_{10} = (120)_5$

Example 13. Convert the decimal number $(0.42)_{10}$ into binary.

Sol.

Decimal fraction	× Base	Product	Recorded carry	
0.42	× 2	0.84	0	MSB
0.84	× 2	1.68	1	↑
0.68	× 2	1.36	1	
0.36	× 2	0.72	0	
0.72	× 2	1.44	1	LSB

So $(0.42)_{10} = (0.01101)_2$ **Ans.**

Example 14. Convert $(0.8)_{10}$ to equivalent binary number.

Sol.

Decimal fraction	Base	Product	Recorded carry	
0.8	× 2	1.6	1	MSB
0.6	× 2	1.2	1	↑
0.2	× 2	0.4	0	
0.4	× 2	0.8	0	
0.8	× 2	1.6	1	LSB

$\therefore \quad (0.42)_{10} = (0.11001)_2$ **Ans.**

Example 15. Convert $(0.6234)_{10}$ into its equivalent octal number.

Sol. Table : Conversion of fractional decimal to octal

Decimal fraction	Base	Product	Recorded carry	
0.6234	× 8	4.9872	4	MSD
0.9872	× 8	7.8976	7	↓
0.8976	× 8	7.1808	7	
0.1808	× 8	1.4484	1	
0.4484	× 8	3.5712	3	LSD

$\therefore \quad (0.6234)_{10} = (0.47713)_8$ **Ans.**

Example 16. Convert the decimal fraction $(0.122)_{10}$ to its equivalent hex number.

Sol.

Decimal fraction	× Base	Product	Recorded decimal	Hex	
0.122	× 16 =	1.952	1	1	MSD
0.952	× 16 =	15.232	15	F	↑
0.232	× 16 =	3.712	3	3	
0.712	× 16 =	11.392	11	B	
0.392	× 16 =	8.272	8	6	
0.272	× 16 =	4.352	4	4	LSD

$\therefore \quad (0.122)_{10} = (0.1F3B64)_{16}$ **Ans.**

Example 17. Convert $(85.63)_{10}$ into its equivalent binary number.

Sol. Step 1 : Separate integer and fractional parts.

85 . 63 → Integer (85), Fractional (63)

Step 2 : Convert the integer

2	85		
2	42	1	LSB
2	21	0	↑
2	10	1	
2	5	0	
2	2	1	
2	1	0	
	0	1	MSB

$\therefore \quad (85)_{10} = (1010101)_2$

Step 3 : Convert the fractional part

$0.63 \times 2 = 1.26 \quad 1$ MSB

$0.26 \times 2 = 0.52 \quad 0$

$0.52 \times 2 = 1.04 \quad 1$

$0.04 \times 2 = 0.08 \quad 0$

$0.08 \times 2 = 0.16 \quad 0$ LSB

$\therefore \quad (0.63)_{10} = (10100)_2$

Step 4 : Combine the results of steps 2 and 3.

$\therefore \quad (85.63)_{10} = (1010101.10100)_2$ **Ans.**

Scanned by CamScanner

Example 18. (a) Convert the octal number $(364)_8$ into equivalent binary number.

Sol.

Step 1 : Given octal number	:	3	6	4
Step 2 : Convert each digit into binary	:	011	110	100

$\therefore \quad (364)_8 = (011110100)_2$ **Ans.**

Example 18. (b) Convert $(364.25)_8$ into its equivalent binary number.

Sol. Follow the same procedure explained in the previous example.

Step 1 : Given octal number	:	3	6	4	.	2	5
Step 2 : Convert each digit into binary	:	011	110	100	.	010	101

$\therefore \quad (364.25)_8 = (011\ 110\ 100.010\ 101)_2$ **Ans.**

Example 19. Convert the hex number AFB2 into equivalent binary number.

Sol. Each digit in the given hex number is converted into 4-bit binary numbers as shown in fig.

Given hex number	A	F	B	2
Each digit converted to its four bit binary equivalent	1010	1111	1011	0010

Fig. : Hex to binary conversion

Hence $(AFB2)_{16} = (1010\ 1111\ 1011\ 0010)_2$ **Ans.**

Example 20. Convert $(0.12E)_{16}$ into equivalent octal number.

Sol. Step 1 : Convert each hex digit into 4-bit binary word :

Given hex number	.	1	2	E
		↓	↓	↓
Binary number	.	0001	0010	1110

} Hex to binary conversion

Step 2, 3 : Group the binary bits into groups of 3 bits and convert each group into an octal number :

. 000 100 101 110 ← Groups of 3 bits

. 0 4 5 6 ← Octal digits

$\therefore \quad (0.12E)_{16} = (0.0456)_8$ **Ans.**

Example 21. Determine the values of base x if :

(i) $(211)_x = (152)_8$ (ii) $(193)_x = (623)_8$

Sol. (i) $(211)_x = (152)_8$

$$2\times x^2 + 1\times x^1 + 1\times x^0 = 1\times 8^2 + 5\times 8^1 + 2\times 8^0$$
$$2x^2 + x + 1 = 64 + 40 + 2$$
$$2x^2 + x + 1 = 106$$
$$2x^2 + x - 105 = 0 \qquad \cdots(1)$$

Solving Equation (1) we get,

$$x = 7$$

$\therefore \quad (211)_7 = (152)_8$ **Ans.**

(ii) $(193)_x = (623)_8$

$$1\times x^2 + 9\times x^1 + 3\times x^0 = 6\times 8^2 + 2\times 8^1 + 3\times 8^0$$
$$x^2 + 9x + 3 = 384 + 16 + 3$$
$$x^2 + 9x + 3 = 403$$
$$x^2 + 9x - 400 = 0 \qquad \cdots(2)$$

Solving Equation (2) we get,

$$x = 16$$

$\therefore \quad (193)_{16} = (623)_8$ **Ans.**

Example 22. (a) Obtain the 2's complement of $(1011)_2$.

Sol. Step 1 : Obtain 1's complement of the given number :

1's complement of 1011 is 0100.

Step 2 : Add 1 to 1's complement :

1's complemnt	:	0 1 0 0
Add 1	:	+ 1
2's complement	:	0 1 0 1

Hence the 2's complement of 1011 is 0101.

Example 22. (b) Obtain the 2's complement of $(10110010)_2$.

Sol.

Given number	:	1 0 1 1 0 0 1 0
1's complemnt		0 1 0 0 1 1 0 1
Add 1	:	+ 1
2's complement	:	0 1 0 0 1 1 1 0

Example 23. Perform $(9)_{10} - (5)_{10}$ using 2's complement method.

Sol. Step 1 : Obtain 2's complement of $(5)_{10}$:

Decimal	Binary	2's complement
$(5)_{10}$	$(0101)_2$	1011

Step 2 : Add $(9)_{10}$ to 2's complement of $(5)_{10}$:

$(9)_{10}$	1 0 0 1
2's complement of $(5)_{10}$	1 0 1 1
Carry	1 1
Discard Carry →	1 0 1 0 0 ⇒ $(4)_{10}$

Final carry indicates that the answer is positive and in its true form. → Answer

$(9)_{10} - (5)_{10} = (4)_{10}$ **Ans.**

Example 24. Perform $(4)_{10} - (9)_{10}$ using the 2's complement method.

Sol. Convert both the numbers to binary.

$$(4)_{10} = 0100_2 \text{ and } (9)_{10} = 1001_2$$

Scanned by CamScanner

Step 1 : Obtain 2's complement of $(9)_{10}$:

Decimal	Binary	2's complement
$(9)_{10}$	$(1001)_2$	(0111)

Step 2 : Add $(4)_{10}$ to 2's complement of $(9)_{10}$:

$(4)_{10}$:		0	1	0	0
2's complement of $(9)_{10}$:	+	0	1	1	1
Carry :	[1]				
Final carry →	[0]	1	0	1	1

→ Answer is negative and in the 2's complement form.

→ "0" carry indicates that the result is negative and in its 2's complement form.

Step 3 : Convert the answer into its true form :

Answer : 1 0 1 1 In 2's complement form 1

Subtract 1 : − 1

1 0 1 0

Invert all bits : 0 1 0 1 Answer in true form.

Thus the answer is $-(0101)_2$ *i.e.*, $(-5)_{10}$.

Example 25. Add $(10111)_2$ and $(11001)_2$.

Sol.

- The binary addition will take place column by column. The carry generated in the addition of bits in the previous column is to be transferred to the next column.
- Then in that column the addition of carry and the bits corresponding to the two given numbers is to be performed.

Carry		1	1	1	1	
A	+	1	0	1	1	1
B		1	1	0	0	1
Answer →	1	1	0	0	0	0

So the result of addition is $(110000)_2$.

Example 26. Add 24 and 13 in binary.

Sol. Step 1 : Convert the given numbers into binary numbers.

2	24		
2	12	0	LSB
2	6	0	
2	3	0	
2	1	1	
	0	1	MSB

2	13		
2	6	1	LSB
2	3	0	
2	1	1	
	0	1	MSB

∴ $(24)_{10} = (11000)_2$ ∴ $(13)_{10} = (1101)_2$

Step 2 : Perform the binary addition.

Carry			1				
$(24)_{10}$			1	1	0	0	0
$+(13)_{10}$		+		1	1	0	1
$(37)_{10}$	=	1	0	0	1	0	1	← Answer

Example 27. Subtract the decimal numbers $(38)_{10}$ and $(29)_{10}$ by converting them into binary.

Sol. Step 1 : Convert $(38)_{10}$ and $(29)_{10}$ into their binary equivalents :

2	38		
2	19	0	LSB
2	9	1	
2	4	1	
2	2	0	
2	1	0	
	0	1	MSB

2	29		
2	14	1	LSB
2	7	0	
2	3	1	
2	1	1	
	0	1	MSB

∴ $(38)_{10}(100110)_2$ ∴ $(29)_{10} = (11101)_2$

Step 2 : Perform the subtraction :

Borrow	1	1			1		
$(38)_{10}$	1	0	0	1	1	0	
$-(29)_{10}$	−	1	1	1	0	1	
$(09)_{10}$ →	0	0	1	0	0	1	← Answer

Example 28. Add $(569)_{10}$ and $(687)_{10}$ in BCD.

Sol.

569 →	0101	0101		1001
+687 →	0110	1000		0111
11	1	1		111
1256	1011	1111	[1]	0000
	Invalid BCD	Invalid BCD		Valid BCD with carry 1 ← Incorrect answer

Add $(0110)_2$ to only the invalid BCD numbers to get correct answer.

	1011		1111	0000	← Incorrect answer
+	0110		0110	0110	
	1111		11		
[1]	0010	[1]	0101	0110	
↓	↓		↓	↓	
0001	0010		0101	0110	← Correct BCD answer
1	2		5	6	

∴ $(569)_{10} + (687)_{16} = (1256)_{10}$

Scanned by CamScanner

Example 29. Perform the subtraction $(4)_{10} - (7)_{10}$ using the 9's complement.

Sol. Step 1 : Obtain 9's complement of $(7)_{10}$:

9's complement of 7 is $(9 - 7) = (2)_{10}$

Step 2 : Add $(4)_{10}$ and 9's complement of $(7)_{10}$:

$(4)_{10}$ →		0	1	0	0	← BCD of 4
9's complement of $(7)_{10}$ →	+	0	0	1	0	
[0]		0	1	1	0	← Result

Final carry is 0. Hence result is negative, hence take 9's complement of the result.

Step 3 : Take 9's complement of the result :

9 →		1	0	0	1	
−6 →	−	0	1	1	0	← Result obtained in step 2
3		0	0	1	1	← Final answer

$$(4)_{10} - (7)_{10} = (-3)_{10}$$

Example 30. Perform the subtraction $(9)_{10} - (4)_{10}$ in the BCD using the 10's complement.

Sol. Step 1 : Obtain the 10's complement of $(4)_{10}$:

9's complement of 4 = 9 − 4 = $(5)_{10}$

Add 1 + 1

10's complement of 4 → 6

Step 2 : Add $(9)_{10}$ and 10's complement of $(4)_{10}$:

$(9)_{10}$ →		1	0	0	1	
$+(6)_{10}$ →		0	1	1	0	
Carry →						
		1	1	1	1	← Invalid BCD and carry = 0
Step 3 : Add $(6)_{10}$	+	0	1	1	0	← Add $(0110)_2$ for correction
		1	1			
Discard final carry → [1]		0	1	0	1	← Answer is positive and in true BCD form
			5			

$\therefore$ $(9)_{10} - (4)_{10} = (5)_{10}$

Example 31. Perform $(3)_{10} - (8)_{10}$ in BCD using 10's complement method.

Sol. Step 1 : 10's complement of $(8)_{10}$:

10's complement of $(8)_{10}$ is $(9 - 8) + 1 = 2$

Step 2 : Add 3 and 10's complement of 8 :

$(3)_{10}$ →		0011	
$+(2)_{10}$ →	+	0010	← 10's complement of 8
Carry →		1	
		[0101]	← Sum is negative

Final carry is 0. Hence sum is negative and not in its true form.

Step 3 : Obtain 10's complement of the sum :

Subtract the sum from $(9)_{10}$ →		1	0	0	1	
	−	0	1	0	1	← Sum
		1				
		0	1	0	0	
Add 1 →	+				1	
Final Result :		0	1	0	1	← BCD 5

The result is negative.

$\therefore$ $(3)_{10} - (8)_{10} = (-5)_{10}$

Example 32. Perform $(54)_{10} - (22)_{10}$ in BCD using 10's complement.

Sol. Step 1 : 10's complement of 22 :

9's complement of 22	→	99−22	=	77
Add 1	→		+	1
10's complement of 22	→			78

Step 2 : Add $(54)_{10}$ and 10's complement of $(22)_{10}$:

$(54)_{10}$ →		0 1 0 1	0 1 0 0	
10's complement of $(22)_{10}$ →	+	0 1 1 1	1 0 0 0	← $(78)_{10}$
Carry →		1 1 1		
Invalid BCD numbers →		1 1 0 0	1 1 0 0	
Step 3 : Add $(0110)_2$ →		0 1 1 0	0 1 1 0	
Carry →		1 1	1	
Discard final carry → [1]		0 0 1 1	0 0 1 0	← Answer is positive and in true BCD form
		3	2	

$\therefore$ $(54)_{10} - (22)_{10} = (32)_{10}$

Example 33. Add $(7)_{10}$ and $(6)_{10}$ in excess-3.

Sol. Convert $(7)_{10}$ and $(6)_{10}$ in excess-3.

$(7)_{10} = (1010)_{xs-3}$ and $(6)_{10} = (1001)_{xs-3}$

Step 1 : Add the two excess-3 numbers :

		1	0	1	0	← Excess – 3 for $(7)_{10}$
	+	1	0	0	1	← Excess – 3 for $(6)_{10}$
Final carry →	[1]	0	0	1	1	← Sum

Step 2 : Carry is 1 so add 0011 to the sum :

0 0 0 1	0 0 1 1	← Sum
0 0 1 1	0 0 1 1	← Add (3)
0 1 0 0	0 1 1 0	← Final sum in excess-3
1	3	

$\therefore$ $(7)_{10} + (6)_{10} = (13)_{10}$

Example 34. Add $(2)_{10}$ and $(3)_{10}$ in excess-3.

Sol. Convert the given decimal numbers into excess-3.

$$(2)_{10} = (0101)_{xs-3}, (3)_{10} = (0110)_{xs-3}$$

Step 1 : Add the two Excess-3 numbers :

	0 1 0 1	← Excess – 3 for $(2)_{10}$
+	0 1 1 0	← Excess – 3 for $(3)_{10}$
Carry	1	
Final carry → 0	1 0 1 1	

Step 2 : Carry is 0, so subtract (0011) from the sum :

	1 0 1 1	← Sum
–	0 0 1 1	← Subtract (3)
	1 0 0 0 = $(5)_{10}$	← Final sum in excess-3

$\therefore$ $(2)_{10} + (3)_{10} = (5)_{10}$

Example 35. Prove that $(A + \overline{B} + AB)(A + \overline{B})(\overline{A}B) = 0$

Sol. $LHS = (A + \overline{B} + AB)(A + \overline{B})(\overline{A}B)$

But $A + AB = A$ ... Refer to Ex. 2.6.1.

$LHS = (A + \overline{B})(A + \overline{B})(\overline{A}B)$

$= (AA + A\overline{B} + A\overline{B} + \overline{B}\overline{B})(\overline{A}B)$

But $A \cdot A = A$ and $\overline{B} \cdot \overline{B} = \overline{B}$ and $A\overline{B} = A\overline{B} = \overline{B}(A + A) = A\overline{B}$

$\therefore$ $LHS = (A + A\overline{B} + \overline{B})(\overline{A}B)$

$= [A(1 + \overline{B}) + \overline{B}](\overline{A}B)$

But $(1 + \overline{B}) = 1$ $\therefore LHS = [(A \cdot 1) + \overline{B}](\overline{A}B)$

$\therefore$ $LHS = (A + \overline{B})(\overline{A}B)$ ... since $A \cdot 1 = A$

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

But $A\overline{A} = 0$ and $B\overline{B} = 0$

$\therefore$ $LHS = 0 + 0 = 0$ ... Proved

Example 36. Simplify : $ABCD + A\overline{B}CD$.

Sol. $Y = ABCD + A\overline{B}CD = ACD(B + \overline{B})$

But $(B + \overline{B}) = 1$

$\therefore$ $Y = ACD$ Ans.

Example 37. Simplify the following expression :

$$Y = \overline{(\overline{AB} + \overline{A} + AB)}$$

Sol. $Y = \overline{(\overline{AB} + \overline{A} + AB)}$

But $\overline{AB} = \overline{A} + \overline{B}$ ... De-Morgan's first theorem

$\therefore$ $Y = \overline{(\overline{A} + \overline{B} + \overline{A} + AB)}$

But $\overline{A} + \overline{A} = \overline{A}$

$\therefore$ $Y = \overline{(\overline{A} + \overline{B} + AB)}$

Now use De-Morgan's second theorem which states that,

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

$\therefore$ $Y = \overline{\overline{A}} \cdot \overline{\overline{B}} \cdot \overline{AB}$

But $\overline{\overline{A}} = A$ and $\overline{\overline{B}} = B$

$\therefore$ $Y = A \cdot B \cdot \overline{AB}$

But $\overline{AB} = (\overline{A} + \overline{B})$ ... De-Morgan's second theorem

$\therefore$ $Y = A \cdot B(\overline{A} + \overline{B}) = A\overline{A}B + AB\overline{B}$

But $A\overline{A} = 0$ and $B\overline{B} = 0$

$\therefore$ $Y = 0 \cdot B + A \cdot 0$

$= 0 + 0$ ... since $0 \cdot B = 0$ and $A \cdot 0 = 0$

$\therefore$ $Y = 0$

Example 38. Simplify the following Boolean expressions :

1. $A\overline{B} + \overline{A}B + AB + \overline{A}\overline{B}$
2. $A\overline{B}C + \overline{A}BC + ABC$

Sol.

1. $Y = A\overline{B} + \overline{A}B + AB + \overline{A}\overline{B}$

$= \overline{B}(A + \overline{A}) + B(\overline{A} + A)$

$= \overline{B} + B$ ... since $(\overline{A} + A) = 1$

$\therefore$ $Y = 1$ ... since $\overline{B} + B = 1$

2. $Y = A\overline{B}C + \overline{A}BC + ABC$

$= AC(\overline{B} + B) + \overline{A}BC$

$= AC + \overline{A}BC$

$= AC + \overline{A}BC$

$= C(A + \overline{A}B)$

$= Y = C(A + B)$ ... since $A + \overline{A}B = A + B$

Example 39. Prove the following Boolean identifies :

(i) $A + \overline{A}B + AB = A + B$

(ii) $A\overline{B} + \overline{A}B + AB + \overline{A}\overline{B} = 1$

Sol.

(i) $LHS = A + \overline{A}B + AB = A + B(\overline{A} + A)$

$= A + B(1)$ ... since $A + \overline{A} = 1$

$\therefore$ $LHS = A + B = RHS.$

(ii) $LHS = A\overline{B} + \overline{A}B + AB + \overline{A}\overline{B} = A\overline{B} + \overline{A}\overline{B} + \overline{A}B + AB$

$= (A + \overline{A})\overline{B} + (\overline{A} + A)B$

$= \overline{B} + B$ ... since $\overline{A} + A = 1$

$= 1$ ... since $\overline{B} + B = 1$

$\therefore$ $LHS = 1$ ... Proved.

Example 40. Simplify the expression given below.

$$Y = AB + (A + B)(\overline{A} + B).$$

Sol. Step 1 : Bring the given expression in SOP form.

Given expression : $Y = AB + (A + B)(\overline{A} + B)$

$= AB + (A\overline{A} + AB + \overline{A}B + BB)$

Step 2 : Search for common factors and simplify :

$Y = AB + A\overline{A} + AB + \overline{A}B + BB$

$= AB + AB + BB + A\overline{A} + \overline{A}B$

But $AB + AB = AB, BB = B$ and $A\overline{A} = 0$

$\therefore$ $Y = AB + B + \overline{A}B = B(A+1) + \overline{A}B$

But $(A+1) = 1$

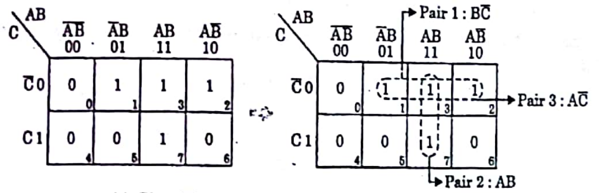
$\therefore$ $Y = B + \overline{A}B = B(1+\overline{A}) = B$ ... since $(1+\overline{A}) = 1$

$\therefore$ $Y = B$ Ans.

This is simplified expression.

Example 41. For the K-map shown in figure or write the simplified Boolean expression.

Sol.



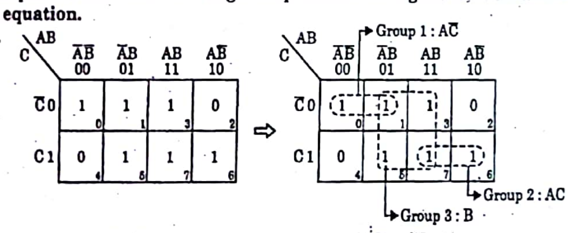
(a) Given K-map (b) Grouping

Simplified Boolean expression : $Y = B\overline{C} + AB + A\overline{C}$ Ans.

(1) (2) (3)

Example 42. For the Karnaugh map shown in figure obtain the simplified Boolean equation.

Sol.



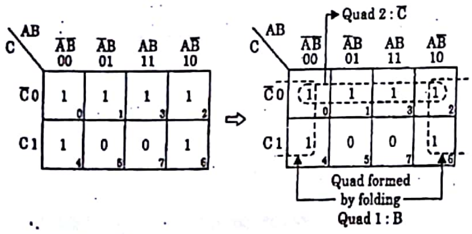
(a) Given K-map (b) Simplification

Simplified Boolean expression : $Y = \overline{A}\overline{C} + AC + B$ Ans.

(1) (2) (3)

Example 43. Write the simplified Boolean expression from the K-map given in fig.

Sol.

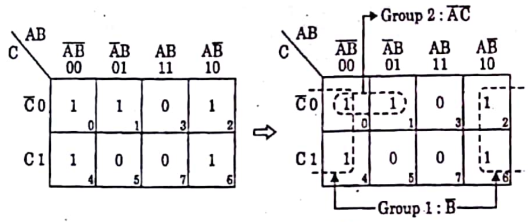


(a) Given K-map (b) Simplification

Simplified Boolean expression : $Y = \overline{B} + \overline{C}$

Example 44. Write the simplified Boolean equation from the K-map shown in figure.

Sol.



(a) Given K-map (b) Simplification

Simplified Boolean equation : $Y = \overline{B} + \overline{A}\overline{C}$

(1) (2)

Ans.

Example 45. A logical expression in the standard SOP form is as follows :

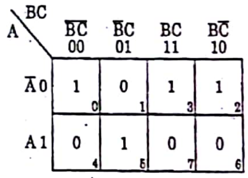
$$Y = \overline{A}B\overline{C} + \overline{A}\overline{B}\overline{C} + \overline{A}BC + A\overline{B}C$$

Minimize it using the K-map technique.

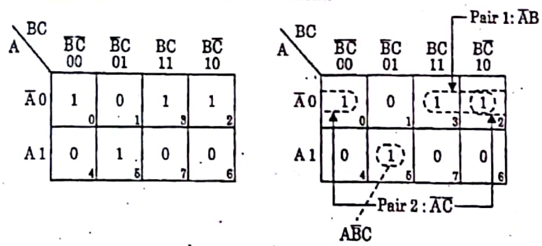
Sol.

Step 1 : Prepare the K-map and place 1's and 0's as shown in fig.

$$\overline{A}B\overline{C} = m_2, \overline{A}\overline{B}\overline{C} = m_0, \overline{A}BC = m_3, A\overline{B}C = m_5$$



Step 2 : Locate the isolated 1's and encircle them.



Step 3 : Identify the pairs.

$\therefore$ $Y = \overline{A}B + \overline{A}\overline{C} + A\overline{B}C$

Pair 1 Pair 2 Isolated 1

This is the minimized expression.

Scanned by CamScanner

Example 46. For the logical expression given below draw the K-map and obtain the simplified logical expression.

$Y = \Sigma m(1, 5, 7, 9, 11, 13, 15)$.

Realize the minimized expression using the basic gates.

Sol. The given expression is,

$$Y = m_1 + m_5 + m_7 + m_9 + m_{11} + m_{13} + m_{15}$$

It can be expressed on K-map as shown in fig.

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	0 (0)	1 (1)	0 (3)	0 (2)
$\bar{A}B$ 01	0 (4)	1 (5)	1 (7)	0 (6)
AB 11	0 (12)	1 (13)	1 (15)	0 (14)
$A\bar{B}$ 10	0 (8)	1 (9)	1 (11)	0 (10)

Quad 1 : $\bar{C}D$; Quad 2 : BD; Quad 3 : AD

The simplified logical expression is given by,

$$Y = \underbrace{\bar{C}D}_{\text{Quad 1}} + \underbrace{BD}_{\text{Quad 2}} + \underbrace{AD}_{\text{Quad 3}} \quad \ldots(1)$$

Minimized expression $\quad Y = D(\bar{C} + B + A) \quad \ldots(2)$

Example 47. Minimize the following Boolean expression using K-map and realize it using the basic gates.

$$Y = \Sigma m(1, 3, 5, 9, 11, 13).$$

Sol. The given expression can be expressed in terms of the minterms as,

$$Y = m_1 + m_3 + m_5 + m_9 + m_{11} + m_{13}$$

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	0 (0)	1 (1)	1 (3)	0 (2)
$\bar{A}B$ 01	0 (4)	1 (5)	1 (7)	0 (6)
AB 11	0 (12)	1 (13)	1 (15)	0 (14)
$A\bar{B}$ 10	0 (8)	1 (9)	1 (11)	0 (10)

Quad 1 : $\bar{B}D$; Quad 2 : $\bar{C}D$

The corresponding K-map is shown in fig.

The logical expression is given by,

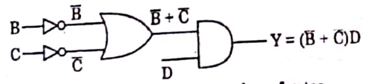
$$Y = \underbrace{\bar{B}D}_{\text{Quad 1}} + \underbrace{\bar{C}D}_{\text{Quad 2}} \quad \ldots(1)$$

$$Y = D(\bar{B} + \bar{C})$$

∴ This is the minimized logical expression.

Realization

The minimized expression is the realized as shown in fig. P.2.13.4(b).



Relation with minimum number of gates.

Example 48. Minimize the following expression using K-map and realize using the basic gates.

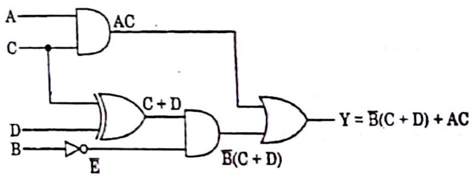
$$Y = \Sigma m(1, 2, 9, 10, 11, 14, 15)$$

Sol.

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	0 (0)	1 (1)	0 (3)	1 (2)
$\bar{A}B$ 01	0 (4)	0 (5)	0 (7)	0 (6)
AB 11	0 (12)	0 (13)	1 (15)	1 (14)
$A\bar{B}$ 10	0 (8)	1 (9)	1 (11)	1 (10)

Group 1; Group 2 : $\bar{B}C\bar{D}$; Group 3 : AC; Group 2; Group 2; Group 1 : $\bar{B}\bar{C}D$

(a) K-map simplification.



(b) Realization with minimum number of gates.

Minimized expression : $\quad Y = \bar{B}\bar{C}D + \bar{B}C\bar{D} + AC$

$$= B\underbrace{(\bar{C}D + C\bar{D})}_{\text{EX OR gate}} + AC$$

$$\therefore \quad Y = \bar{B}(C \oplus D) + AC$$

Scanned by CamScanner

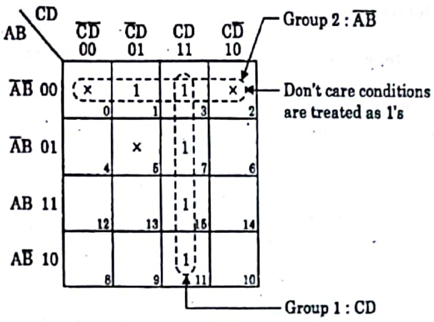
Example 49. Simplify the expression given below using K-map. The don't care conditions are indicated by $d(\,)$.

$$Y = \Sigma m(1, 3, 7, 11, 15) + d(0, 2, 5).$$

Sol. The given equation is,

$$Y = \underbrace{m_1 + m_3 + m_7 + m_{11} + m_{15}}_{\text{Regular minterms so enter 1's}} + \underbrace{d(0, 2, 5)}_{\text{Don't care conditions so enter x marks}}$$

The required K-map is shown in fig. P.2.13.11.

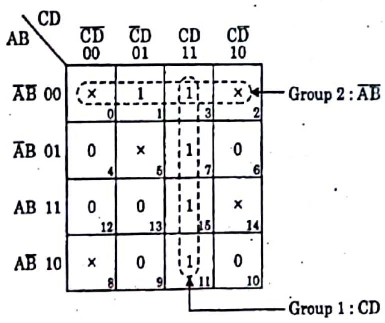


$\therefore$ Simplified equation is $\quad Y = CD + \overline{A}\,\overline{B}.$

Example 50. Minimize the SOP expression given below using K-map.

$$Y = \Sigma m(1, 3, 7, 11, 15) + \Sigma d(0, 2, 5, 8, 14).$$

Sol. The required K-map with don't care conditions.

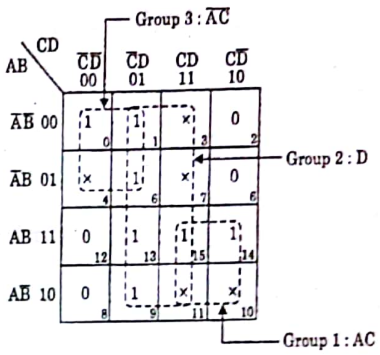


$$\therefore \qquad Y = CD + \overline{A}\,\overline{B}$$

Example 51. Minimize the following expression using K-map.

$$Y = \Sigma m(0, 1, 5, 9, 13, 14, 15) + d(3, 4, 7, 10, 11).$$

Sol. Refer fig. for the required K-map.



Simplified equation $\qquad Y = AC + D + \overline{A}\,\overline{C}$
(1) (2) (3)

$$Y = D + \underbrace{(AC + \overline{A}\,\overline{C})}_{\text{EX-NOR}}$$

$$\therefore \qquad Y = D + (A \odot C) \qquad \textbf{Ans.}$$

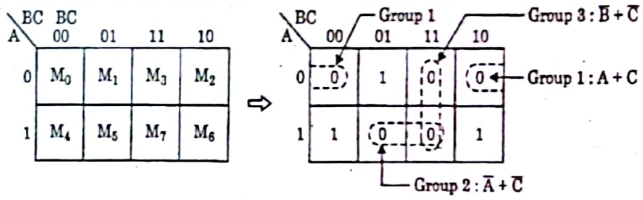
Example 52. Minimize the following standard POS expression using K-map.
$Y = \Pi M(0, 2, 3, 5, 7)$

Sol.

1. The given expression is in the POS form. It can be written in terms of the maxterms as follows :

$$Y = M_0 M_2 M_3 M_5 M_7$$

2. The three variable K-map for POS form is shown in fig.



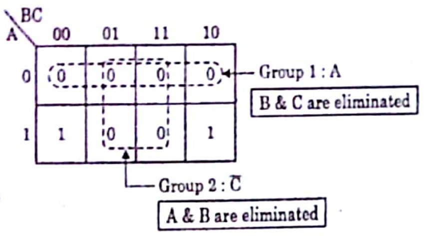
3. Enter a 0 corresponding to each maxterm in the given expression as shown in fig. P.2.15.2(b).
4. Three groups are formed as shown in fig. P.2.15.2(b).

$\therefore$ Minimized expression is as follows :

$$Y = (A + C)(\overline{A} + \overline{C})(\overline{B} + \overline{C}) \qquad \textbf{Ans.}$$

Example 53. Find the expression for the output in the POS form for the K-map shown in fig.

Scanned by CamScanner



Sol.

- Note that two quads have been encircled in fig. P.2.15.3.
- When a quad is encircled, two variables which change will get eliminated.
- Expression for output in the POS form is as follows :
-

$$Y = A \cdot \overline{C}$$

(Group 1: A; Group 2: $\overline{C}$)

Ans.

Example 54. Write the simplified expression for output in the POS form for the following expression :

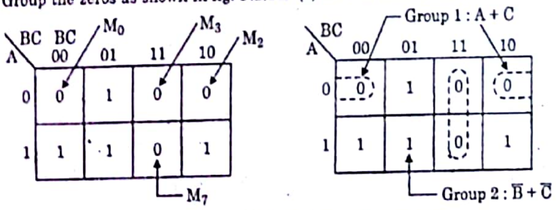
$$Y = \Pi M(0, 2, 3, 7)$$

Sol.

1. The given expression is,

$$Y = M_0 M_2 M_3 M_7$$

2. In the K-map enter 0's corresponding to these maxterms and enter 1's in the remaining cells as shown in fig. P.2.15.4(a).
3. Group the zeros as shown in fig. P.2.15.4(b) for the further simplifications.



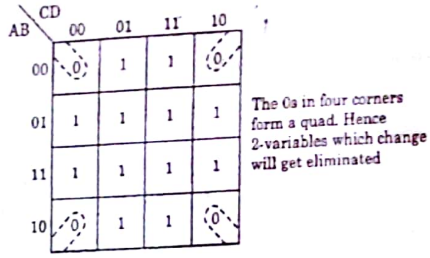
(a) (b) Simplification using K-map.

4. Simplified equation using K-map is as follows :

$$Y = (A + C)(\overline{B} + \overline{C})$$

Ans.

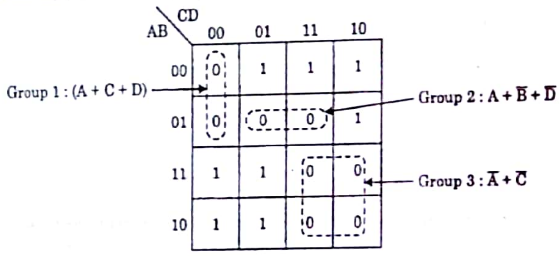
Example 55. Write the expression for ouput in the POS form for the K-map shown in fig.



Sol. Expression for output $Y = B + D$

Example 56. For the K-map shown in fig. write the expression for output in the POS form.

Sol. The expression for output is



$$Y = (A + C + D)(A + \overline{B} + \overline{D})(\overline{A} + \overline{C})$$

(1) (2) (3)

□

Scanned by CamScanner

विधि (Procedure)

(1) NAND गेट की सत्य तालिका का सत्यापन करने के लिए, IC-7400 के एक गेट के साथ चित्र 1.1 के अनुसार, परिपथ को ब्रेड-बोर्ड पर संयोजित करते हैं।

(2) अब इस IC की पिन-14 पर ++ V_{cc} (= +5V) तथा पिन-7 पर ग्राउन्ड प्रयुक्त करते हैं।

(3) द्विपथ स्विच Sw_1 द्वारा A- इनपुट टर्मिनल पर तथा स्विच Sw_2 द्वारा B- इनपुट टर्मिनल पर सत्य तालिका की कोई एक अवस्था के अनुसार इनपुट प्रयुक्त करके, NAND गेट के आउटपुट का प्रेक्षण करते हैं।

इसी प्रकार सत्य-तालिका को चारों अवस्थाओं का सत्यापन करते हैं। जब जिस input(s) एवं output पर विभव की उच्च अवस्था होगी, उसका LED प्रकाशित हो जाएगा तथा इसके विपरीत निम्न अवस्था होने पर LED ऑफ रहता है।

❑.

प्रयोग संख्या-2

उद्देश्य (Object)

XOR तथा NAND गेटों को प्रयुक्त कर अर्द्ध योगकारी (half-adder) एवं अर्द्ध-अन्तरकारी (half subtractor) की संरचना करना तथा उनकी सत्य-तालिका का सत्यापन करना।

उपकरण एवं युक्तियाँ (Apparatus and Devices)

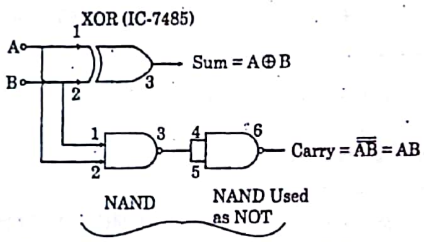
ब्रेड-बोर्ड, रेगुलेटेड पॉवर-सप्लाई- 5V, 500 mA, ICs—7400 तथा 7486, डुअल बीम CRO, मल्टीमीटर अथवा LED इन्डीकेटर इत्यादि।

सिद्धान्त (Theory)

हाफ-एडर परिपथ केवल 2-बिट का योग (sum) तथा कैरी (carry) प्रदान कर सकता है। इसी प्रकार हाफ सब्ट्रेक्टर केवल 2-बिट का अन्तर तथा बॉरो प्रदान कर सकता है।

हाफ-एडर तथा हाफ सब्ट्रेक्टर की सत्य-तालिकायें क्रमश: चित्र 2.2 तथा चित्र 2.4 तालिका में प्रकाशित की गई हैं।

परिपथ-आरेख (Circuit Diagram)



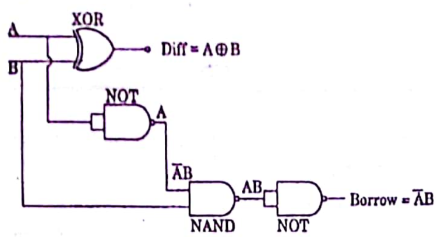
चित्र : 2.1-Half Adder circuit Using XOR and NAND gates.

हाफ-एडर की सत्य तालिका तथा लॉजिक व्यंजक

Input		Output		Logic Relations
A	*B*	Sum	Carry	
0	0	0	0	(i) Sum = $A\overline{B} + \overline{A}B = A \oplus B$
0	1	1	0	where
1	0	1	0	$\oplus$ = Symbol of Ex-OR
1	1	0	1	(ii) Carry $= A \cdot B = \overline{\overline{A \cdot B}} =$ NAND and NOT

चित्र : 2.2- Truth Table of Half-Adder Circuit

Scanned by CamScanner



चित्र : 2.3- Half-Subtractor Circuit.

हाफ-सब्ट्रेक्टर की सत्य-तालिका तथा लॉजिक-व्यंजक

Input		Output		Logic Relations
A	B	Diff.	Carry	
0	0	0	0	Difference $= \bar{A}\cdot B + A\cdot\bar{B} = A\oplus B$
0	1	1	1	Borrow $= \bar{A}\cdot B = \bar{A}\cdot B$
1	0	1	0	= Not A,1 NAND ($\bar{A}$ and B) and NOT
1	1	0	0	

चित्र : 2.4 – Truth Table of Half-Subtractor Circuit

विधि (Procedure)

(1) Half-Adder की संरचना और उसकी सत्य-तालिका का सत्यापन—

(i) चित्र के अनुसार परिपथ को ब्रेड-बोर्ड पर संयोजित करते हैं।

(ii) अब दोनों ICs-7486 एवं 7400 की पिन-14 पर +5V तथा पिन-7 पर ग्राउन्ड प्रयुक्त करते हैं।

(iii) चित्र की सहायता से LEDs इन्डीकेटर बना कर सत्य-तालिका का सत्यापन करते है।

(2) Half-Subtractor की संरचना और उसकी सत्य-तालिका का सत्यापन—

(iv) चित्र के अनुसार परिपथ को ब्रेड-बोर्ड पर संयोजित करते हैं।

(v) अब विधि (ii) से (iii) तक की पुनरावृत्ति करते हैं।

❑

प्रयोग संख्या-3

उद्देश्य (Object)

XOR तक NAND गेट प्रयुक्त कर Full Adder परिपथ की संरचना करना।

उपकरण एवं युक्तियाँ (Apparatus and Devices)

ब्रेड-बोर्ड, रेगुलेटेड पावर सप्लाई– 5V, 500 mA;

ICs-7486 एवं 7400, LED इन्डीकेटर, डुअल-बीम CRO अथवा मल्टीमीटर।

सिद्धान्त (Theory)

फुल-एडर एक साथ तीन अंकों का योग करते है। इस फुल-एडर के आउटपुट पर एक योग (sum) का अंक तथा दूसरा कैरी अंक प्राप्त होता है।

पूर्ण योगकारी की सत्य-तालिका तथा लॉजिक-व्यंजक

Inputs			Outputs	
A	B	C	Sum	Carry
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

Logic Relations

$$\text{Sum} = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$$
$$= \bar{A}\bar{B}C + \bar{C}(\bar{A}B + A\bar{B}) + ABC$$
$$= C(AB + AB) + \bar{C}[A \ominus B]$$
$$= C(AA + B\bar{B} + \bar{A}\bar{B} + AB) + \bar{C}[A \ominus B]$$
because $\overline{AA} = B\bar{B} = 0$

So Sum

$$= C\,[A(\bar{A} + B) + \bar{B}(\bar{A} + B)] + \bar{C}\,[A \ominus B]$$
$$= C\,[(A.\bar{B})(\bar{A}.B)] + \bar{C}[A \ominus B]$$
$$= C(A + \bar{B})(\overline{A.B}) + \bar{C}[A \ominus B]$$
by De Morgan's Theorem
$$= C[A\bar{B} + \bar{A}B] + \bar{C}[A \ominus B]$$
by De Morgan's Theorem
$$\text{Sum} = C\,\overline{[A \ominus B]} + C[A \ominus B]$$
$$= C \ominus [A \ominus B]$$

जहाँ, C इनपुट— पहले वाले कॉलम की carry है।

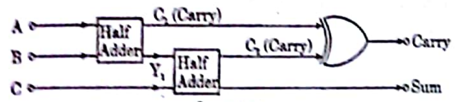
अत: $\text{Sum} = C \oplus [A \ominus B]$

इसी प्रकार $\text{Carry} = A\bar{B}C + A\bar{B}C + AB\bar{C} + ABC$

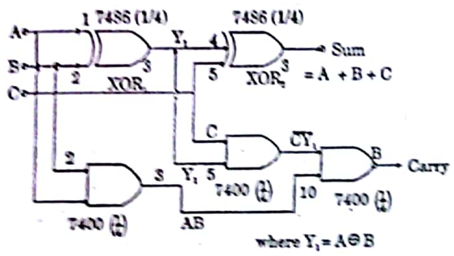
$$= C(AB + AB) + AB(C + \bar{C})$$
$$= C(A \ominus B) + A.B \qquad (\text{क्योंकि } C + \bar{C} = 1)$$
$$= \overline{\overline{[C(A \oplus B)}.\overline{[A\,B]}}$$

Scanned by CamScanner

परिपथ आरेख (Circuit Diagram)



चित्र : 3.1 (a)



where $Y_1 = A \oplus B$

चित्र : 3.1 (b) Circuit Diagram of a Full-adder using XOR and NAND gates

विधि (Procedure)

(1) पूर्ण-योजकार्ती परिपथ को चित्र के अनुसार संयोजित करके, दोनों ICs-7486 एवं 8400 को पिन-14 पर +5V तथा पिन-7 पर ग्राउन्ड विभव प्रयुक्त करते हैं।

(2) अब वोल्टमीटर या CRO द्वारा दी गई सत्य-तालिका का सत्यापन करते हैं।

❑

प्रयोग संख्या-4

उद्देश्य (Object)

JK फ्लिप-फ्लॉप की सत्य-तालिका (Truth-table) को सत्यापित करना।

उपकरण एवं युक्तियाँ (Apparatus and Devices)

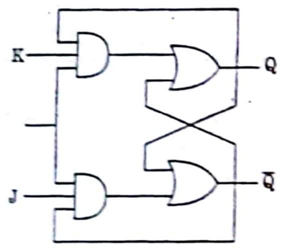
डिजिटल ट्रेनिंग एवं डिजिटल इलेक्ट्रॉनिक सेट IC 7404, IC 7408, IC 7411, IC 7474, IC 7476

सिद्धान्त (Theory)

डिजिटल सिस्टम में लॉजिक सर्किट या तो कम्बीनेशनल होती है या Sequential होती है। Combinational Circuit अपने वर्तमान वैल्यू पर निर्भर करती है जबकि Sequential Circuit में ऐसा नहीं है। Flip-Flop, Sequential Circuit की मुख्य कड़ी है।

JK Flip-Flop

JK Flip-flop, SR Flip-flop का मोडिफाइड वर्जन है।



चित्र 4.1 : Logic Diagram of JK Flip flop

Characteristic Table

CLK	J	K	Q (t + 1)	Comment
0	X	X	Q (t)	NC
1	0	0	Q (t)	NC
1	0	1	0	Reset
1	1	0	1	Set
1	1	1	Q̄ (t)	Toggle

विधि (Procedure)

(1) I.C. एवं ब्रेडबोर्ड की सहायता से JK Flip-flop को इम्प्लीमेन्ट करते हैं।

(2) इसके बाद Truth table को सत्यापित करते हैं।

❑

Scanned by CamScanner

प्रयोग संख्या-5

उद्देश्य (Object)

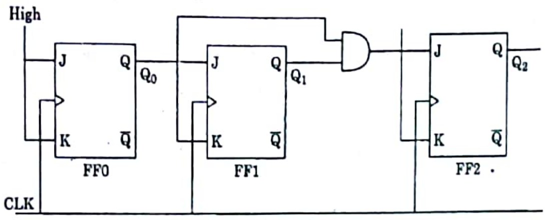
Counter को डिजाइन कर उसकी सत्य-तालिका का सत्यापन करना।

उपकरण एवं युक्तियाँ (Apparatus and Devices)

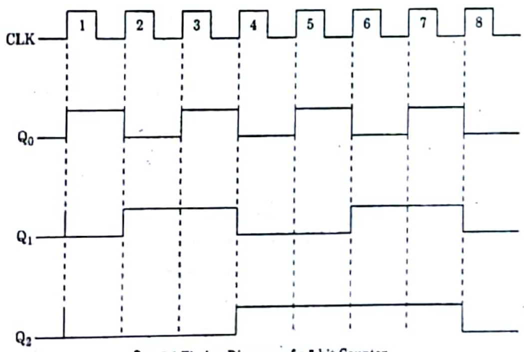
डिजिटल I.C. ट्रेनर किट, IC 7473, 74LS08, 74LS32, 74LS04

सिद्धान्त (Theory)

डिजिटल सर्किट मे पल्सों को काउन्ट करने वाली युक्ति को काउन्टर कहते है। काउन्टर कुछ फ्लिप-फ्लॉप का समूह होता है जिसमें क्लॉक पल्स सिग्नल एप्लाई की जाती है।



चित्र : 5.1 Logic Diagram of a 3-bit Synchronous Counter.



चित्र : 5.2 Timing Diagram of a 3-bit Counter.

विधि (Procedure)

1. चित्र में दिए गए लॉजिक सर्किट को कन्सट्रक्ट करना।
2. Up/Down इनपुट का उपयोग करते हुए Up Counter तथा Down Counter का चयन करना।
3. दिए गए Count Sequence को सत्यापित करना।

प्रयोग संख्या-6

उद्देश्य (Object)

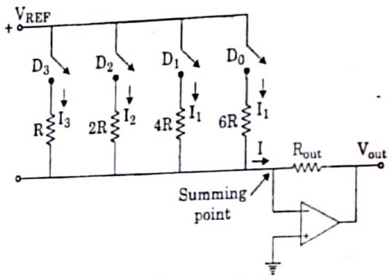
8 bit D/A कनवर्टर का ऑपरेशन तथा सत्यापन करना।

उपकरण एवं युक्तियाँ (Apparatus and Devices)

ट्रेनर किट D/A कनवर्टर

सिद्धान्त (Theory)

D/A कनवर्टर का परिपथ चित्र मे दर्शाया गया है। इसमें V_{REF} Reference Voltage है।



चित्र : 6.1 D/A Converte.

विधि (Procedure)

1. परिपथ में प्रयुक्त स्विच खुली या बंद दो स्थितियों में हो सकती है।
2. जब सभी स्विचें खुले होंगे तब इनपुट करेन्ट शून्य होगी।
3. सभी स्विच बन्द होने की स्थिति में $D_3 = D_2 = D_1 = D_0 = 1$ होगा।

Scanned by CamScanner

ODD SEMESTER EXAMINATION (U.P.), DECEMBER–2019

डिजिटल इलैक्ट्रॉनिक्स

(Digital Electronics)

Code : 2110

Third Semester

Time : 2.30 Hours] **[Maximum Marks : 50**

Notes :

(i) Attempt all questions.

(ii) Students are advised to specially check the Numerical Data of question paper in both versions. If there is any difference in Hindi translation of any question, the students should answer the question according to the English version.

(iii) Use of Pager and Mobile Phone by the students is not allowed.

नोट—सभी प्रश्नों के उत्तर दीजिए।

1. निम्नलिखित में से किन्हीं दो भागों के उत्तर दीजिए— [2 × 5 = 10]

(अ) निम्नलिखित का परिवर्तन कीजिए—

(i) $(6475)_8 = (\ )_{16}$ (ii) $(456)_8 = (\ )_2$

(iii) $(10110011)_2 = (\ )_{gray}$ (iv) $(8AD)_{16} = (\ )_2$

(v) $(497.5)_{10} = (\ )_2$

(ब) 1's कम्प्लीमेन्ट विधि द्वारा निम्न प्रश्न को हल कीजिए—

(i) 1011.11 – 1001.01 (ii) 1100.10 – 10111.01

(स) निम्नलिखित नम्बरों को हल कीजिए—

(i) $(6243)_8 - (4465)_8$ (ii) $(1ACD)_{16} + (5CDB)_{16}$

2. निम्नलिखित में से किन्हीं चार भागों के उत्तर दीजिए— [4 × 3 = 12]

(अ) NOR गेट को प्रयुक्त कर AND, OR तथा NOT प्रचालन हेतु परिपथ बनाइए।

(ब) डायोड एवं प्रतिरोध का प्रयोग कर 3–इनपुट का OR गेट का परिपथ आरेख खीचिए। सत्य तालिका की मदद से कार्यप्रणाली समझाइये।

(स) पाजिटिव लाजिक में एक NAND गेट का परिपथ बनाइए। यदि निगेटिव लॉजिक माना जाये तब यह गेट किस गेट के समतुल्य होगा?

(द) निम्न फंक्शन को k-map द्वारा सरल कीजिए तथा न्यूनतम गेट प्रयुक्त कर Realise कीजिए।
$f(A, B, C, D) = \Sigma m\ (3, 4, 5, 7, 9, 13, 14, 15)$

(य) व्यंजक $y = AB + A(B + C) + B(B + C)$ को बूलियन बीजगणित से सरल कीजिए एवं सरलीकृत व्यंजक की लॉजिक डायग्राम बनाइये।

3. निम्नलिखित में से किन्हीं तीन भागों के उत्तर दीजिए— [3 × 4 = 12]

(अ) दो हाफ सबट्रैक्टर एवं एक OR गेट की मदद से फुल सबट्रैक्टर का परिपथ आरेख खींचिए। फुल सबट्रैक्टर की सत्य तालिका भी बनाइए।

(ब) कामन एनोड एवं कामन कैथोड–7 सेगमेन्ट डिस्प्ले युक्तियों में क्या अन्तर है? प्रत्येक का उपयोग देते हुए परिपथ बनाइए।

(स) BCD to Decimal डिकोडर पर टिप्पणी लिखिए।

(द) J.K. फ्लिप फ्लाप का लाजिक परिपथ खींचिए तथा J.K. फ्लिप-फ्लाप की सत्य तालिका लिखिए।

Scanned by CamScanner

4. निम्नलिखित में से किन्हीं दो भागों के उत्तर दीजिए— [2 × 4 = 8]

(अ) 4 बिट सिन्क्रोनस काउन्टर का परिपंथ आरेख खींचकर कार्यविधि समझाइये।

(ब) अप/डाउन गणक की कार्यविधि परिपथ आरेख की मदद से समझाइये।

(स) लाजिक डायग्राम की मदद से पैरलल-इन सीरियल आउट शिफ्ट रजिस्टर की कार्यविधि समझाइये।

5. निम्नलिखित में से किन्हीं दो भागों के उत्तर दीजिए— [2 × 4 = 8]

(अ) Successive Approximation प्रकार का ADC परिपथ बनाइये तथा इसकी कार्यप्रणाली समझाइये।

(ब) RAM तथा ROM में अन्तर स्पष्ट कीजिए।

(स) EPROM मेमोरी पर टिप्पणी लिखिए।

O

Scanned by CamScanner