EICoM
ELECTRICAL COMPUTER MECHATRONICS

تلخيص منطق رقمي

اعداد: فيحاء الحديدي

بخط: اسراء عبد العال

إرادة - ثقة - تغيير

[1]

Bite ⇒ 0,1

Nibble ⇒ 4 bits

Byte ⇒ 8 bits

Word ⇒ 16 bits

(حفظ) 2^{10} (Kilo) ⇒ 1024

2^{20} (meya) ⇒ 1024 * 1024

2^{30} (Giga) ⇒ 1024 * 1024 * 1024

2^{40} (Tera) ⇒ 1024 * 1024 * 1024 * 1024

Exl what is the exact number of bytes?! 32 Kbyte

32 Kbyte ⇒ 32 * 1024 b = 32 . 1024 b = 32.768 b

Exl find the exact number bit ?! 32 Kbyte : 32m byte

32 Kbyte = 32 * 1024 * 8 = 262,144 bit

32 m byte = 32 * $(1024)^2$ * 8 = 268,435,456

* التحويلات :

الصغير لكبير بنضرب والكبير لصغير بنقسم ...

الكبير (Decimal) ، الصغير (any system)

* any system ⇒ Decimal (صغير ⇒ كبير) (نضرب بـ أساس النظام)

* سواء كان الرقم صحيح أو عشري طريقة الحل

Note : جزء الفاصلة نرقم (للوجب) وإذا كلهم نبدأ بـ (0) وبعد الفاصلة نبدأ بـ (السالب)

Exl $(57.24)_8$ = ( )10

⇒ $5^{1}\ 7^{0} . 2^{-1}\ 4^{-2}$

$5*8^1 + 7*8^0 + 2*8^{-1} + 2*8^{-2} = (47.3125)_{10}$

Exl $(AF.8)_{16}$ = ( )10

⇒ $A^{1}\ F^{0} . 8^{-1}$

$10*16 + 15 + \frac{8}{16} = 175 + 0,5 = (175,5)_{10}$

تذكير

10 = A

11 = B

12 = C

13 = D

14 = E

15 = F

Exl $(10110.1)_2$ = ( )10

(حفظ) نبدأ من (2) ضروري

[1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, ...]

⇒ $1^{4}\ 0^{3}\ 1^{2}\ 1^{1}\ 0^{0} . 1^{-1}$

طريقة سريعة للحل (نجمع الوزن بلي فتحته (1))

$1*2^0 + 1*2^2 + 1*2^3 + 1*2^5 + 1*2^{-1} = 1+4+8+32+0,5 = 45,5$

Exl $(12.2)_6$ = ( )10

⇒ $1^{1}\ 2^{0} . 2^{-1}$ ⇒ $2*6^0 + 1*6 + 2*6^{-1} = 8 + 0,33 = (8,333...)_{10}$

* الحكي لحتا اصغر بدي اقسمه لكن بحالة (integer) بالاساس (النظام المطلوب) ← (integer)

بحالة ال (fraction) بضرب

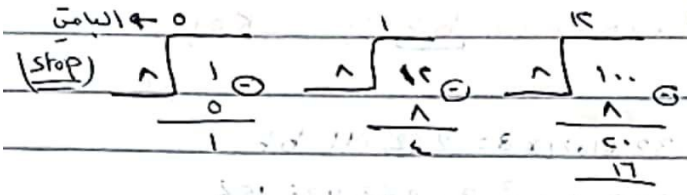
* Decimal ⇒ any system

- (integer)

بوقف عن القسمة لما الباقي = zero بحالة (integer)

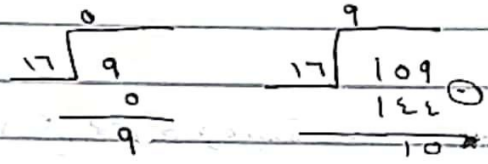
بوقف عن الضرب بحالة تكرار الرقم او الجواب = (fraction) zero

Ex) $(100)_{10} = (144)_8$



$\Rightarrow (144)_8$

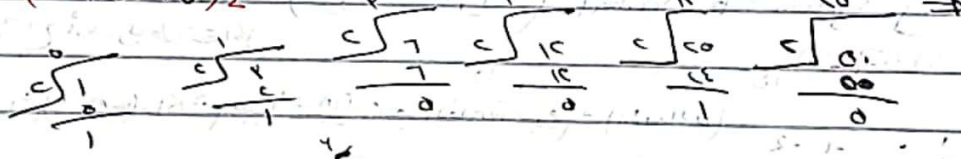
Ex) $(154)_{10} = (9F)_{16}$



$(9F)_{16}$

ط(1) جمع الاوزان التي بتساوي الرقم

Ex) $(.50)_{10} = (.11001 0)_2$



$\Rightarrow (.11001 0)_2$

ط(2) ضرب

$\frac{1}{2}$ $\frac{1}{4}$ $\frac{1}{8}$ $\frac{1}{16}$ $\frac{1}{32}$ $\frac{1}{64}$

.50 = 0 . 1 : 1 . 0, 0, 1 0

- (fraction) الاجباري

Ex) $(0,3)_{10} = (0,23146..)_8$ ← (5 precision) بحال حدد لي عدد الخانات بلتزم فيهم

$0,3 * 8 = 2,4$

$0,4 * 8 = 3,2$

$0,2 * 8 = 1,6$

$0,6 * 8 = 4,8$

$0,8 * 8 = 6,4$ ⇒ تكرار (stop)

Ex) $(0,3)_{10} = (0,4\text{C}\ldots)_{16}$ A(10), B(11), C(12), D(13), E(14), F(15)

$0,3 * 16 = 4,8$

$0,8 * 16 = 12,8$ ← تكرار (stop)

Ex) $(0,75)_{10} = (0,11)_2$

$0,75 * 2 = 1,5$

$0,5 * 2 = 1,0$ ← zero (stop)

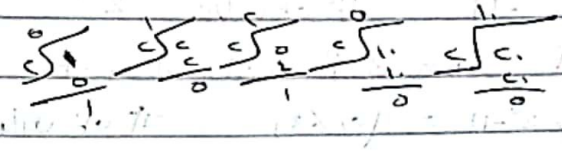
3

حالة ورود (int) مع (frac) يحول كل واحد على طريقة

Ex) $(20.25)_{10} = (10100.01)_2$

$0.25 * 2 = 0.5$

$0.5 * 2 = 1.0 \rightarrow$ zero (stop)



* Binary ⇒ octal ... [نبدأ من اليمين ونقسم إلى 3 مجموعات (1، 2، 4) ونحول كل مجموعة على حدة]

Ex) $(10\ 110\ 11.10\ 111)_2 = (133.56)_8$

Zero (على اليسار لا يؤثر)

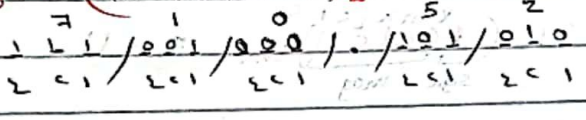
Zero (على اليمين لا تؤثر في القيمة)

001 011 011 010 111

133.56

* يجب الانتباه إن بين عشرة (1)

Ex) $(710.52)_8 = (111001000.101010)_2$



* Binary ⇒ Hexadecimal ... (نفس Bin ⇒ octal) لكن نقسم إلى 4 مجموعات (1، 2، 4، 8)

Ex) $(110100111000.11)_2 = (D38.C)_{16}$

1101 0011 1000 . 1100

D 3 8 . C

Ex) $(FC12.34)_{16} = (\quad)_2$

F C 1 2 . 3 4

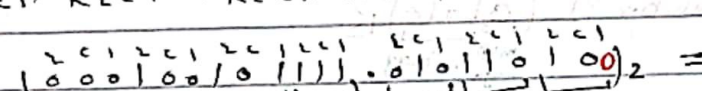
1111 1100 0001 0010 . 0011 0100

$\Rightarrow (1111110000010010.00110100)_2$

* Hexa ⇒ octal ... الوسيط (Binary) حيث من (Hex ⇒ Bin) ثم من (Bin ⇒ octal)

Ex) $(12F.5A)_{16} = (\quad)_8$

0001 0010 1111 . 0101 1010



$= (0457.264)_8$

0 4 5 7 . 2 6 4

* Binary ⇒ Gray code

① copy msB (دائماً متشابهة msB)

② G[i] = XoR [B[i+1], B[i]] i: index digits #(0 → n-1)

Gray الرقم بـ (Binary) الرقم بـ XoR = اشارة ⊕

XoR جدول		
	xy	F
[(Zero) المتشابهة	00	0
(one) المختلفة]	01	1
	10	1
	11	0

Ex] $(11011)_2$ ⇒ Gray

i = ٤ ٣ ٢ ١ ٠

i=4 ⇒ G[4] = B[4] نسخ msB دائماً

i=3 ⇒ G[3] = B[3+1] ⊕ B[3] = 1⊕1 = 0 (متشابهة)

i=2 ⇒ G[2] = B[2+1] ⊕ B[2] = 1⊕0 = 1

i=1 ⇒ G[1] = B[1+1] ⊕ B[1] = 0⊕1 = 1

i=0 ⇒ G[0] = B[0+1] ⊕ B[0] = 1⊕1 = 0

G = 10110

Gray code ⇒ Binary

① copy msB

② B[i] = XoR [B[i+1], G[i]]

Ex] $(10101)_{Gray}$ ⇒ Binary

i = ٤ ٣ ٢ ١ ٠

B[4] = G[4] = 1

B[3] = B[4] ⊕ G[3] = 1+0 = 1

B[2] = B[3] ⊕ G[2] = 1⊕1 = 0

B[1] = B[2] ⊕ G[1] = 0+1 = 1

B = 11011

B[0] = B[1] ⊕ G[0] = 1+0 = 1

* Decimal Coding (Binary Code decimal [BCD])

Ex) $(128)_{10}$ ⇒ BCD

من الباب ٩ (Hexa) [بأخذ كل منزلة لوحدها ونحولها لأربع منازل]

1	2	8
٨٤٢١	٨٤٢١	٨٤٢١
0001	0010	1000

⇒ $(0001\ 0010\ 1000)_{BCD}$

Ex) $(1\ 0011\ 1000\ 0101)_{BCD}$ ⇒ Decimal

1001	1000	0101
٨٤٢١	٨٤٢١	٨٤٢١

$(985)_{10}$

Ex) $(1011\ 1010)_{BCD}$ ⇒ Decimal

1011	1010
٨٤٢١	٨٤٢١

Decimal لا يصح ان يكون (B A) #

لكن (10) يكتب هكذا بـ (BCD)

(0001 0000)

* Unsigned Number :-

Rang $(0, 2^n - 1)$ # of value 2^n # of zero = 1

Ex) find the Rang of n=6 for unsigned number ?!

Rang = $(0, 2^6 - 1) = (0, 63)$ # of value = 64 (2^6)

If n=8 represent 520.

Rang $(0, 2^8 - 1) = (0, 255)$ ⇒ can not be represent (out of Range)

* Signed Number :- ((MSB) بعد البت الأولى) ⇒ (+) ve ⇒ MSB = 0

Ex) n=8, r=2 (sm) represent (+5) Singed magnitude ⇒ (-) -ve ⇒ MSB = 1

Signed (MSB) 0 0 0 0 0 1 0 1 — magnitude

Ex) what is this number ?! $(1\,0\,0\,1\,0\,1\,1)_{sm}$ = -11

sign mag

Rang $(-(2^{n-1}-1), 2^{n-1}-1)$ # of Value = 2^n # of zero = 2 (+0, -0)

Ex) n=6, r=2, find the Rang, for signed number ?!

Rang $(-(2^{6-1}-1), 2^{6-1}-1) \Rightarrow (-31, +31)$ # of value = $2^6 = 64$

* 1's complement ⇒ (unsigned) ① * إذا msB = 0 ⇐ معاملة عادية مثل

② * إذا msB = 1 ⇐ اطلع الرقم ← (النقص) ثم بعرفه ← (السالب) والعكس صح

0 → 1
1 → 0

Ex) If n=5, 1's complement, find (-5) ?!

③ الرقم السالب اذا معاملة رقم

+5 0 0 1 0 1

-5 1 1 0 1 0 (استناد)

للنا نكتب (1's) كالة السالب بجمع الكوبلمنت

مثال $(1\,0\,1\,0\,1)_{1's}$ ⇐ msB = 1 ∴ -ve ∴ بجمع الكوبلمنت

(Signed) msB magnitude

-(1+4) = -5

Ex) what is the following number ?! $(0\,1\,0\,1\,0)_{1's}$

msB = 0 ⇒ +ve

msB signed mag.

+ve ⇒ معاملة عادية ∴ $(0\,1\,0\,1\,0) \Rightarrow +10$ بجمع ال (1) الزيادي

Ex) what is the following number ?! $(1\,0\,1\,1\,0\,1)_{1's}$

msB = 1 ∴ سالب (-ve)

sig mag

-ve ⇒ بجمع الكوبلمنت

$\Rightarrow (1\,0\,1\,1\,0\,1) = -18$

* 1's Complement

- formal Definition (Diminished Radix)

$B + (-B) = r^n - 1$

↳ Complement of B

Ex) r=10, B=55, find -B ?! n=2

$B + (-B) = r^n - 1 \Rightarrow 55 + (-B) = 10^2 - 1 \Rightarrow -B = 99 - 55 \Rightarrow \boxed{-B = 44}$

Rang $= \left(-(2^{n-1}-1) \text{ to } 2^{n-1}-1\right)$ #(sign) بدون #of value $= 2^n$ #of zero = 2

Ex) find the Rang if n=6, 1's complement ?!

Rang $= \left(-(2^{6-1}-1) \text{ to } 2^{6-1}-1\right) = (-31 \text{ to } 31)$

* 2's Complement

① * اذا MSB = 0 يعني معاملة (1's) و (unsigned) بالعادة العادية

② * اذا MSB = 1 كيف اشارة سالب ثم ابدأ من اقرب الين نزل كل الصفر لعندك اول (one) بصادفة نزله ثم اشقلب (0→1, 1→0)

MSB = 0 ⇒ +ve

MSB = 1 ⇒ -ve

Ex) what is the following number in Decimal ?! $(0\,1\,1\,0\,1)_{2's}$ (Sig bit, mag)

MSB = 0 ⇒ +ve ⇒ معاملة عادية

$0\,1\,1\,0\,1 = +13$

Ex) $(1\,0\,1\,1\,0)_{2's}$ (Sig, mag) MSB = 1 ⇒ -ve ⇒ ⑤ معاملة رقم

① ننزل اول رقمين 10 ثم نقلب مابعد الـ (1) (0→1, 1→0)

⇒ 0 1 0 1 0 ⇒ ⑤ وضع السالب بالجواب (-)

⇒ -10 ② جمع اوزان (1)

$\therefore (1\,0\,1\,1\,0)_{2's} = (-10)_{10}$

Ex) find (-6) in 2's complement if n=7 ?! [بالة السالب] ⑤ معاملة

① اول اشي نطلع الموجب ثم نعمل معاملة ⑤ (0→1, 1→0) بحالة الموجب حل عادي

+6 0 0 0 0 1 1 0

-6 $(1\,1\,1\,1\,0\,1\,0)_{2's}$

Ex) find (-15) in 2's complement if n = 5 ?!

+15 0 1 1 1 1

-15 $(1\,0\,0\,0\,1)_{2's}$

* Formal Definition (Redix Complement)

$B + (-B) = r^n$

Ex) r=10, n=2, B=55, find -B ?!

$B + -B = r^n \Rightarrow 55 + (-B) = 100 \Rightarrow \boxed{-B = 45}$

2's Complement :-

Rang :- $(-(2^{n-1}) \text{ to } 2^{n-1}-1)$ # of value = 2^n # of zero = 1

Ex) Find the Rang for 2's complent, n=4, r=2?!

Rang = $(-(2^{4-1}) \text{ to } 2^{4-1}-1) = (-8, +7)$

* Arithmetic operation [Finit precision] :

* دائماً نفرض عدد البتات قادرة للرقم الناتج الأكبر .

اذا السؤال ما وضحلن كم عدد البتات

[1] لعمل (extension) لـ (msB) (1's/2's) (sign) .

[2] عملية جمع فقط فنعمل منفصل الـ (-) إلى (+) .

[3] لتحديد إذا كان (OV) .

حالة unsigned نضيف (0) // لحالة (sm) نجيب منزلة (msB) الإشارة ثم (0) لبقية البتات .

Unsigned (extend by adding zero)

n=8 , $(110)_2 \Rightarrow (00000110)$

1's, 2's complement (extend the sign bit (msB))

n=8, $(0110)_{1's} \Rightarrow (00000110)_{1's}$

n=8, $(1010)_{2's} \Rightarrow (11111010)_{2's}$

Signed magnitude ((نجيب الإشارة ثم البقية (0))

n=8 , -4 (1100) $\Rightarrow$ 1 0 0 0 0 1 0 0

Ex

cin 1 1 1 0

1 1 1 0 +

Cout 1 1 0 0 0

Check for overflow (OV) (الزائد عن عدد البتات للحفظ)

* unsigned addition $\Rightarrow$ cout = 1 Then (OV)

Ex) n=4 , find 9 + 12 (unsigned addition)

9 1 0 0 1

12 1 1 0 0 +

cout=1 $\Leftarrow$ 1 0 1 0 1

$\therefore$ (OV) the result is wrong

Ex) 5 to 6, n=4, unsigned addition ?!

5 0 1 0 1

6 0 1 1 0 +

cout=0 $\Leftarrow$ 1 0 1 1 $\therefore$ result is ~~wrong~~ correct

1's Complement (add / sub)

If cin ≠ cout then (OV) is true (result is wrong)

cin = cout = 1 Discard cout Add (1) to the result

Ex n=4, 5+2, 1's comp ?!

cin ☐ ^ 2 c 1

5 0 1 0 1

2 0 0 1 0 +

cout ☐ 0 1 1 1

cin = cout = 0 ∴ result is corect no (OV)

Ex n=4, 7-5, 1's comp! ?! 7 + (-5)

7 0 1 1 1 (7) 0 1 1 1

+5 0 1 0 1 ×(-1) (-5) 1 0 1 0 +

1 0 0 0 1

1 +

0 0 1 0

cout = cin = 1 ∴ Discard cout and (+1)

الزيادة: x (-1) ⇒ 0→1, 1→0

∴ (0010) the result is correct

Ex n=4, 5-7, 1's comp ?! 5 + (-7)

5 0 1 0 1

+7 0 1 1 1 *(-1) -7 1 0 0 0 +

☐ 1 1 0 1

cout = cin = 0 ∴ the result is correct no (OV)

Ex n=4, 4+4, 1's comp ?!

+4 ☐ ^ 2 c 1 : 1 0 0

+4 0 1 0 0 +

☐ 1 0 0 0

cin ≠ Cout ∴ (OV) the result is (wrong)

Ex n=4, 3-3, 1's comp. 3 + (-3)

+3 ☐ ^ 2 c 1 : 0 0 1 1

(-1) -3 1 1 0 0 +

☐ 1 1 1 1

cin = cout = zero ∴ the result is correct no (OV)

* 2's Complement (Add / Sub)

If cin ≠ cout then (OV) ∴ the result is wrong

else If (cin = cout = 1) Discard cout (تجاهلهم / (1) طرح)

Ex n=4, 5+2, 2's comp! ?!

5 ☐ ^ 2 c 1 : 0 1 0 1

2 0 0 1 0 +

☐ 0 1 1 1

cin = cout = 0 ∴ the result is true (correct)

Ex) n=4, 7-5 / 2's compl ?! 7 + (-5)

7 = 0 1 1 1

+5 = 0 1 0 1 x(-1) → 1011

1 1 1 0 1 1 1 1 + 1 0 1 1 = 0 0 1 0

cin = cout = 1 Discard Cout, stop. No(ov)

⇒ (0010) result

Ex) n=4, 5-7, 2'compl ?! 5+(-7)

+7 = 0 1 1 1 x(-1) → 1 0 0 1

+5 = 0 1 0 1

1 0 0 1 + 0 1 0 1 = 1 1 1 0

cout = cin = zero ∴ the result is correct.

Ex) n=4, 5-5, 2'comp ?! 5+(-5)

+5 = 0 1 0 1

-5 = 1 0 1 1 +

0 1 0 1 + 1 0 1 1 = 0 0 0 0

cin = cout = 1 ∴ Discard Cout, stop. No(ov)

result = 0000

* Unsigned Subtraction (ov) already did(ov) ما بتشيك عليه

by using 1's comp ⇒ If cout = 1 Discard cout, add (1) to the result.

else cout = 0 ⇒ find 1's comp and place (-) sign [لو طلع (0) كمل باكراند بنوع الحاجة اللي]

Ex) n=4, 14-3, unsigned using 1's compl ?!

14 = 1 1 1 0

3 = 0 0 1 1 (-)

1's compl

1 1 1 0 + 1 1 0 0 = 1 0 1 0

⇒ cout = 1 ∴ Discard, add (1)

the result = 1011 11 + (-5) (unsigned بنتأكد بنوع الباكراند بـ)

Ex) n=4, 3-14, unsigned using 1's comp ?!

3 = 0 0 1 1

14 = 1 1 1 0 (-)

unsigned

1's com

0 0 0 1 + 1 1 0 0 = 0 1 0 0

⇒ cout = 0 ∴ بحط (0) ثم نعوض (-)

(-) (1011) 1's comp

the result = - 1011

* unsigned Sub using 2'comp.

If Cout = 1 Discard Cout (مابيجع (1))

else Cout = 0 ⇒ find 2'comp of the result and place sign(-)

Ex) n=4, 14-3 unsigned, using 2'comp ?!

14 1 1 1 0
3 0 0 1 1 ⊖ unsigned

1 1 1 0 2'comp
1 1 0 1 ⊕
⊠ 1 0 1 1 ⇒ Cout = 1 ∴ Discard, stop.

The result = 1011

Ex) n=4, 3-14, unsigned, using 2'comp ?!

3 0 0 1 1
14 1 1 1 0 ⊖ unsigned

0 1 1 1 2'comp
0 0 1 0 ⊕
□ 0 1 0 1 ⇒ Cout = 0 ∴ (يجوم 2'comp تاني بطلع)

The result = ⊖(1 0 1 1)

* Binary Codes

→ Max of the Value (n) $x = 52 \Rightarrow x = 2^n \Rightarrow 52 = 2^n$

بالنسبة لكود الباينري طالبة فيه (n) من خلال التقريب

بالنسبة لـ (n) الأكبر أخذ $32 < 52 < 64$

$\therefore 52 = 2^n \Rightarrow \boxed{n = 6}$

* The Boolean Algebra [Combinational logic]

[1] AND Gate

* Circuit symbol ⇒ A, B → F (output) (inputs)

* Boolean Expression

↳ $F = A \cdot B \rightarrow AB$

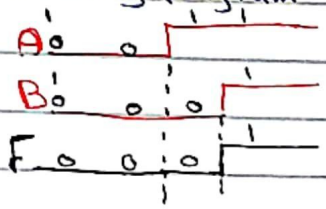
* Truth Table n: # of input ⇒ 2^n ∴ $A \cdot B \Rightarrow 2^2 = 4$

A	B	F(A·B)
0	0	0
0	1	0
1	0	0
1	1	1

⇒ الحالة الوحيدة اللي بتاخد (1) لما الكل = (1)

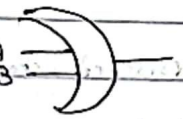
* Timing Diagram

ده اشي بينقط عند اماكن التغير وبنسحب اخر نقطة لطلع (F)



2) OR Gate

* Circuit Symbol ⇒



* Bolem Expression

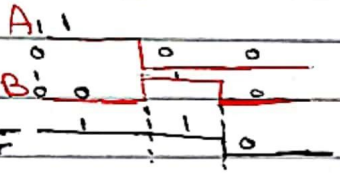
$F = A + B$

* Truth Table $F = A+B \Rightarrow 2^2 = 4$

A	B	F(A+B)
0	0	0
0	1	1
1	0	1
1	1	1

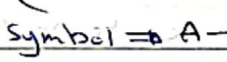
⇒ (zero) الوحيد (zero) اللي بيطلع ال

* Timing diagram



3) NOT (Inverter) ⇒ (نفي الدخول)

* Circuit Symbol ⇒



* Boolean Exp ⇒ $F = \bar{A} = A'$

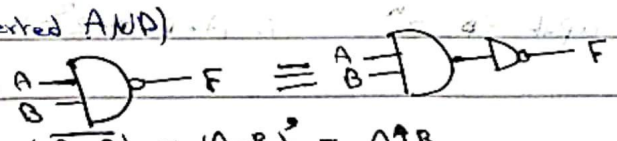
* Truth Table

A	F(Ā)
0	1
1	0

* Timing diagram

4) NAND (Inverted AND)

* Circuit Symbol →



* Boolean Exp ⇒ $\overline{(A \cdot B)} = (A \cdot B)' = A \uparrow B$

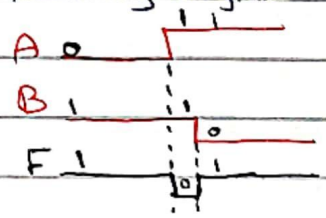
13

* Truth Table

A	B	F(A.B)‾
0	0	1
0	1	1
1	0	1
1	1	0

⇒ (عكس جدول AND لانه اكوان منفي)

* Timing diagram

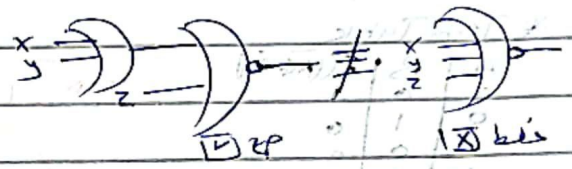


بوابات (NAND, NOR) تحتوي فقط على [2-input]

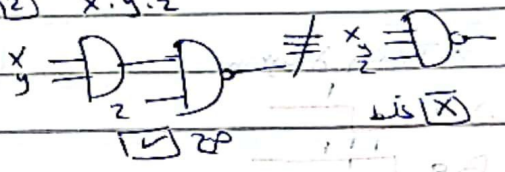
لا يوجد بوابة (NOR, NAND) تحتوي على اكثر من (2 input) سوا كل ؟!

لكن (Extention) منضبط (OR, AND)

* 1) $\overline{x+y+z}$

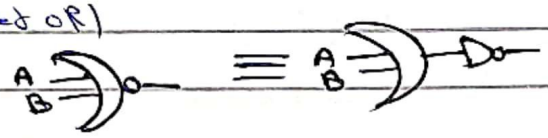


* 2) $\overline{x.y.z}$



5) NOR (Inverted OR)

* Circuit symbol



* Boolean Exp

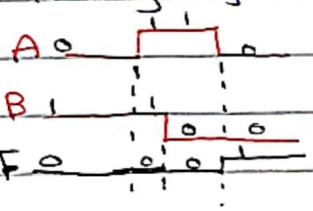
$F = \overline{(A+B)} = (A+B)' = A \downarrow B$

* Truth Table

A	B	F(A+B)‾
0	0	1
0	1	0
1	0	0
1	1	0

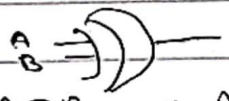
(عكس جدول OR) (ليس, ضد النفي)

* Timing diagram



6) XOR (odd function)

* Circuit symbol



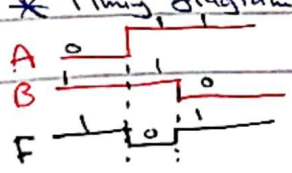
* Boolean Exp $A \oplus B = A\bar{B} + \bar{A}B$ (حفظ) ⇒ (Truth Table) لاثبات

* Truth Table

A	B	F(A⊕B)
0	0	0
0	1	1
1	0	1
1	1	0

(الاختلاف بين (zero) الاختلاف (one)

* Timing diagram



7) NOR (even function) [invert XOR]

* Circuit symbol 

* Boolean Exp →

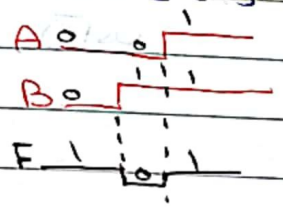
$\overline{(A \oplus B)} = AB + \bar{A}\bar{B}$ حفظ ⇐ (Truth Table) لإثبات

* Truth Table

A	B	F($\overline{A \oplus B}$)
0	0	1
0	1	0
1	0	0
1	1	1

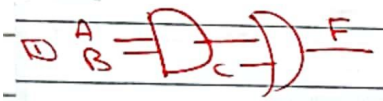
⇐ (one) الاستنتاج
(zero) الفاضية

* Timing Diagram

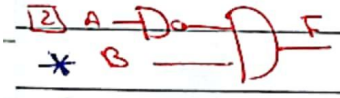


* Describing circuit (Given circuit find F)

Ex) Find F ?!

1) 

$F = A \cdot B + C$ (الأقواس ضرورية)

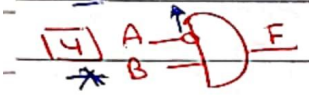
2) 

$F = \bar{A} \cdot B$

3) 

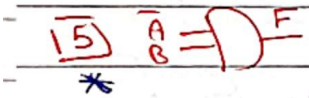
$F = (A + B) \cdot C$ (الأقواس ضرورية)

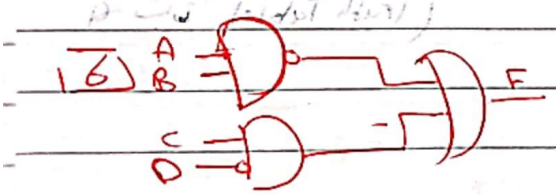
Not gate

4) 

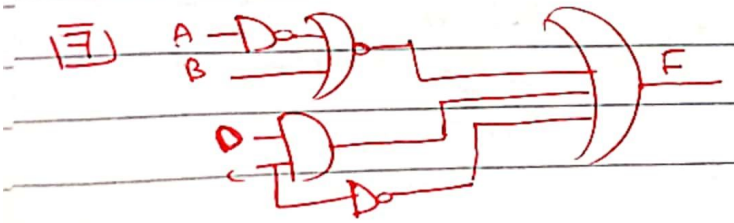
$F = \bar{A} \cdot B$

$F = \bar{A} \cdot B$

5) 

6) 

$F = (\overline{A \cdot B}) + (C \cdot \bar{D})$

7) 

$F = (\overline{\bar{A} + B}) + (D \cdot C) + \bar{C}$

اكملوا: 1) فيزيائيا القواسم إن وجدت
2) NOT
3) AND
4) OR
5) كل ما ساواة الى ولوية تبدأ من شمال لليمين

* Evaluating Circuit (given F) Find the value of F(0/1), Draw the circuit

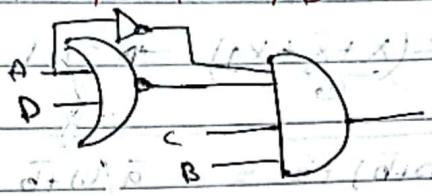
Ex) $F = \bar{A} \cdot B \cdot C \cdot (\overline{A+D})$ $A=0, B=1, C=1, D=1$

* نفس خطوات الرسم

$\bar{0} \cdot 1 \cdot 1 \cdot \overline{(0+1)}$

$1 \cdot 1 \cdot 1 \cdot \bar{1}$

$1 \cdot 1 \cdot 1 \cdot 0 = 0$

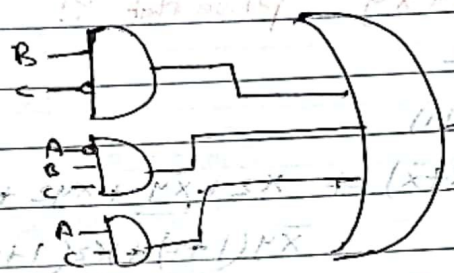


Ex) $F = Ac + B\bar{c} + \bar{A}Bc$

$0 \cdot 1 + 1 \cdot \bar{1} + \bar{0} \cdot 1 \cdot 1$

$0 \cdot 1 + 1 \cdot 0 + 1 \cdot 1 \cdot 1$

$0 + 0 + 1 = 1$

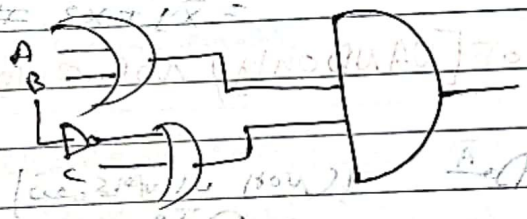


Ex) $F = (A+B) \cdot (\bar{B}+C)$ $A=1, B=1, C=0$, Find F, and Draw circuit?!

$(1+1) \cdot (\bar{1}+0)$

$(1+1) \cdot (0+0)$

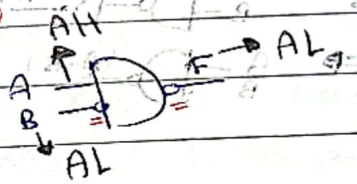
$1 \cdot 0 = 0$



* Active High / Active low

No bubble ⇒ AH

with bubble ⇒ AL



AH
If the value = 1 ⇒ Active
If the value = 0 ⇒ In Active

AL
If the value = 1 ⇒ In Active
If the value = 0 ⇒ Active

* (فنزة سريعة مسبقة ع تفسيرا لعوين)

* Boolean Algebra

1) literal :- a Single Varible in Complemented or not Complemented Form.

Ex) $F = xy + xz + \bar{x}$ ⟹ # of literal = 5 [عدد الاحرف بغض النظر عن التكرار + الـ Complement]

2) Term :- a Collection of literal.

Ex) $F = xy + xz + \bar{x}$ ⟹ # of Term = 3 term

* postulates (القواعد حفظ للاثبات)

1) $x+y \in B$, $x \cdot y \in B$ ⟹ $B = \{0,1\}$

2) $x+0 = x$, $x \cdot 1 = x$ (Identity)

3) $x+y = y+x$, $x \cdot y = y \cdot x$ (commutation)

4) $x \cdot (y+z) = x \cdot y + x \cdot z$, $x+(y \cdot z) = (x+y) \cdot (x+z)$ (Distribution) (حفظا روحه رجعه ضروري)

5) $x+\bar{x} = 1$, $x \cdot \bar{x} = 0$ (complement)

* Theorems

1) $x+x = x$, $x \cdot x = x$ (Idempotency) Ex) $abc + d + abc = abc + d$

2) $x+1 = 1$, $x \cdot 0 = 0$ (Null element) Ex)

$F = ?!$ بالرغم من المدخلات تحتوي على (1) ∴ الجواب (1) بغض النظر سوف يكون الناتج $F = 1$

3) $\bar{\bar{x}} = x = (x')'$ (Involution)

4) $(x+y)+z = x+(y+z)$, $(x \cdot y) \cdot z = x \cdot (y \cdot z)$ (Association)

5) $\overline{x+y} = \bar{x} \cdot \bar{y}$, $\overline{x \cdot y} = \bar{x}+\bar{y}$ (Demorgan law)

6) $x+xy = x$ (مهم) , $x \cdot (x+y) = x$ (Absorption)

* XOR

1) $x \oplus 0 = x$ Ex) $ab\bar{c} \oplus 0 = ab\bar{c}$

2) $x \oplus 1 = \bar{x}$

3) $x \oplus x = 0$ ((Zero) التشويش)

4) $x \oplus \bar{x} = 1$ ((one) الخلفية)

5) $x \oplus y = y \oplus x$

6) $(x \oplus y) \oplus z = x \oplus (y \oplus z)$

7) $x \cdot (y \oplus z) = x \cdot y \oplus x \cdot z$

8) $x \oplus y = \bar{x}y + x\bar{y}$ (اجابة الحفظ سابقا)

* XNOR

$\overline{x \oplus y} = xy + \bar{x}\bar{y} = x \oplus \bar{y} = \bar{x} \oplus y$ (اجابة الحفظ سابقا)

13 :

Ex) $F = \overline{ABCD}$. Find $\bar{F}$?! $\quad \bar{F} = \bar{A} + \bar{B} + \bar{C} + \bar{D}$

Ex) ... Prove Hint ?! (بجاية الدرس جمع)

1) $X + X = X \Rightarrow (X+X)\cdot 1 \Rightarrow (X+X)\cdot(X+\bar{X}) \Rightarrow X + (X\cdot\bar{X})$

$X + 0 = X$ #

2) $X + 1 = 1 \Rightarrow (X+1)\cdot 1 \Rightarrow (X+1)\cdot(X+\bar{X}) \Rightarrow X + (1\cdot\bar{X}) \Rightarrow X + \bar{X} = 1$ #

3) $X + XY = X \Rightarrow X(1+Y) \Rightarrow X\cdot 1 = X$ #

Ex) Find $\bar{F}$?!

1) $A+B+C+D \Rightarrow \overline{A+B+C+D} = \bar{A}\cdot\bar{B}\cdot\bar{C}\cdot\bar{D}$ (توزيع الـ (bar) على الكل، حرف واحد عمليات)

2) $A\cdot\bar{B} + \overline{CD} \Rightarrow \overline{A\cdot\bar{B} + \overline{CD}} = \bar{A} + B\cdot CD$

* Simplification techniques

1) آخذ حدود مشتركة (Combining term)

Ex) $X\cdot Y + X\cdot\bar{Y} = X \Rightarrow X(Y+\bar{Y}) = X\cdot 1 = X$ #

Ex) $ab\cdot cd + ab\cdot\bar{c}d \Rightarrow ab(cd+\bar{c}d) = ab\cdot(1) = ab$ #

Ex) $abcd + abc\bar{d} \Rightarrow abc(d+\bar{d}) = abc\cdot(1) = abc$ #

Ex) $abcd + ab\bar{c}\bar{d} \Rightarrow ab(cd+\bar{c}\bar{d})$ ... ابسط شيء

الفرق شيم عم = الفرق شيم عم

2) Adding Redundant term (اضافة (term) لا وصلا مشترك)

الحرف محافظ على الإشارة

Ex) $a\bar{b}c + abc + \bar{a}bc \Rightarrow abc(\ abc$ زيادة (term) مشترك)

$a\bar{b}c + abc + \bar{a}bc + abc \Rightarrow ac(\bar{b}+b) + bc(\bar{a}+a) =$

$ac(1) + bc(1) = ac + bc$ #

$\Rightarrow$ min # of liters $= c(a+b)$ ... (liters) اذا طلب اقل عدد

Ex) $X + 0 \Rightarrow X + (y\cdot\bar{y}) \Rightarrow (X+y)\cdot(X+\bar{y})$ [اوقات بنضيف من خلال القوانين السابقة]

3) Eliminating term

$X + XY = X \Rightarrow X(1+Y) = X\cdot 1 = X$

Ex) $\bar{a}b + \bar{a}bc = \bar{a}b \Rightarrow \bar{a}b(1+c) \Rightarrow \bar{a}b\cdot(1) = \bar{a}b$ #

$X + \bar{X}Y = X + Y$, $\bar{X} + XY = \bar{X} + Y$ (العرف هو الـ (bar))

Ex) $\bar{A}B + \bar{A}B\bar{C}\bar{D} + ABCD$

$\bar{A}(B + \bar{B}\bar{C}\bar{D}) + ABCD \Rightarrow \bar{A}(B + \bar{C}\bar{D}) + ABCD \Rightarrow \bar{A}B + \bar{A}\bar{C}\bar{D} + ABCD$

$B(\bar{A} + ACD) + \bar{A}\bar{C}\bar{D} \Rightarrow B(\bar{A}+CD) + \bar{A}\bar{C}\bar{D} \Rightarrow \bar{A}B + BCD + \bar{A}\bar{C}\bar{D}$

Ex) Simplify ?!

[1] $xz + z(\bar{x} + xy) \Rightarrow z(x + \bar{x} + xy) \Rightarrow z(1 + xy) = z \cdot 1 = z$ #

[2] $\overline{\bar{a}b + \bar{a}\bar{b} + \bar{b}} = \overline{\bar{a}(b+\bar{b}) + \bar{b}} = \overline{\bar{a}(1) + \bar{b}} = \overline{\bar{a} \cdot \bar{b}} = a + b$ #

Ex) $(x+y)\cdot(\bar{x}+z) = x \cdot z + \bar{x}y$. Prove that ?!

$(x \cdot \bar{x}) + xz + \bar{x}y + yz$

$0 + xz + \bar{x}y + yz \cdot (1)$

$xz + \bar{x}y + yz(x + \bar{x}) \Rightarrow xz + \bar{x}y + xyz + \bar{x}yz$

$\bar{x}y(1+z) + xz(1+y)$

$\bar{x}y \cdot (1) + xz \cdot (1)$

$= \bar{x}y + xz$ #

* Universality of [NAND only / NOR only]

[1] NAND only

[A] ① INV A → $\bar{A}$ (نتبع كما هي الـ (Not))

[B] AND A, B → A.B ⇒ A, B → A.B → A.B ($\bar{\bar{A}} = A$) [الـ (Not) لصحة السمبل]

[C] OR A, B ⇒ A, B → (A+B) (لغير صاحب السمبل (٢) عرات)

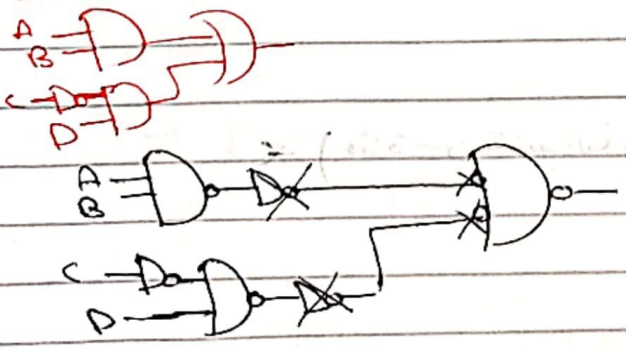
[2] NOR only

[A] INV A → $\bar{A}$

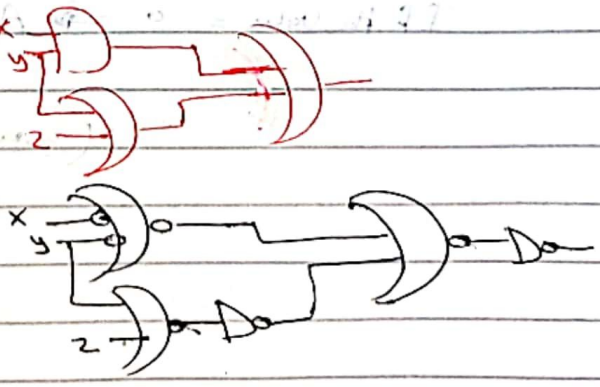
[B] OR A, B → A+B ⇒ A, B → A+B → A+B [الـ (Not) لصحة السمبل (٢) تحول عليها]

[C] AND A, B ⇒ A, B → A.B (لغير صاحب السمبل (٢) عرات)

Ex) Converted to NAND only ?



Ex) Converted to NOR only ?!

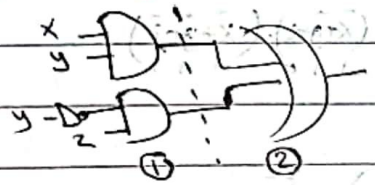


* standard Representation (2 level) هنقط

I) Sum of Product (SoP)

ANDing → ORing

Ex) $F(x,y,z) = xy + \bar{y}z$

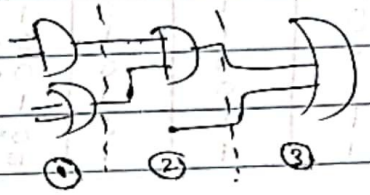


Non-standard

I) أكثر من (2-level)

II) شكل

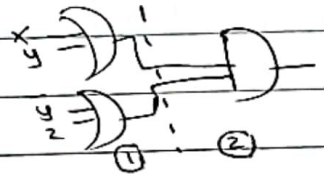
Ex)



II) Product of Sum (PoS)

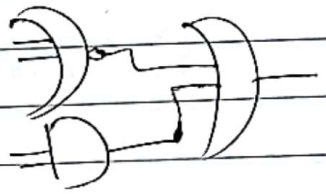
ORing → ANDing

Ex) $F(x,y,z) = (x+y)\cdot(\bar{y}+z)$



Non-standard

(NOR) لعيب وجود



* Canonical Representation (standard) حالة خاصة من

I) Sum of Product [minterm] (Variable كل الحدود يحتوي كل (term) يجب ان يكون كل

Ex) $F(x,y,z) = \underbrace{\bar{x}yz}_{\text{minterm}} + \underbrace{x\bar{y}z}_{\text{minterm}} + \underbrace{\bar{x}\bar{y}z}_{\text{minterm}}$ (Compl مكمل او)

i) → I) Standard (AND → OR)

Ex) $xy + z\bar{w}$ I) Standard (Sum of minterm) II) minterm (Canonical)

II) nonminterm (canonical)

II) Product of sum [Maxterm] [Variable كل الحدود يحتوي على كل انواع ال (term) كل مكمل او Complement]

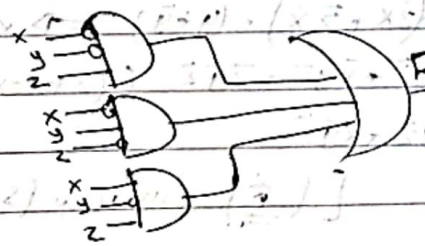
Ex) $F(x,y,z) = (x+y+\bar{z})\cdot(x+\bar{y}+z)$

I) Standard , II) Maxterm i) (Product of maxterm)

Ex) How to represent a function ?? [حالة ال (SoP) (min) بدور على (1) بـ F وبجمع كل (bar) على الـ zero]

$F = \Sigma(1,2,5)$, $F = m_1 + m_2 + m_5$, $F = \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}z$ (001 010 101)

minterm	x	y	z	F
m_0	0	0	0	0
m_1	0	0	1	1
m_2	0	1	0	1
m_3	0	1	1	0
m_4	1	0	0	0
m_5	1	0	1	1
m_6	1	1	0	0
m_7	1	1	1	0



ثم طريقة التعبير عنها ال (SoP)

بحالة التعبير بتكتب فقط ال (1)

Ex) How may to present a function ?! $F(x,y,z)$?! (F) zero لدورها (MAX) (PoS) ← حالة ال

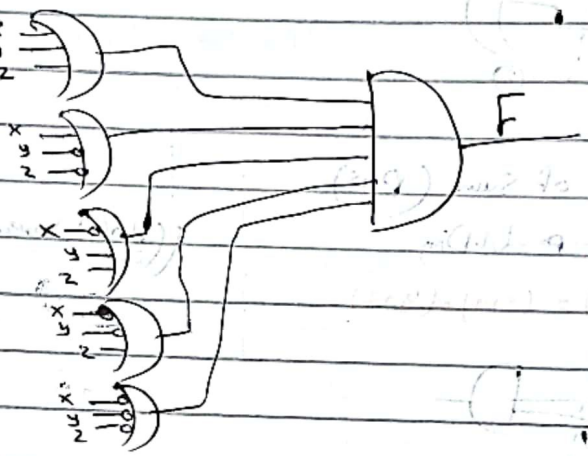
minterm	x	y	z	F
M_0	0	0	0	0
M_1	0	0	1	1
M_2	0	1	0	1
M_3	0	1	1	0
M_4	1	0	0	0
M_5	1	0	1	1
M_6	1	1	0	0
M_7	1	1	1	0

$F = \Pi(0,3,4,6,7)$ [ohne على (bar) وبوضع ال

$= M_0 \cdot M_3 \cdot M_4 \cdot M_6 \cdot M_7$

$(x+y+z)\cdot(x+\bar{y}+\bar{z})\cdot(\bar{x}+y+\bar{z})\cdot(\bar{x}+\bar{y}+z)\cdot(\bar{x}+\bar{y}+\bar{z})$

(0 0 0) (0 1 1) (1 0 0) (1 1 0) (1 1 1)



Ex) $F(x,y) = \Sigma(0,3)$, Find F in term of x,y ?!

* بالمثل (AND → OR) ∴ (Σ) تحول (bar) على (zero)

$F = \bar{x}\bar{y} + xy$

(0) (3)

$F(x,y,z,w) = \Sigma(1,2,3,4,5)$ ← bar (zero)

$\bar{x}\bar{y}\bar{z}w + \bar{x}\bar{y}z\bar{w} + \bar{x}y\bar{z}w + x\bar{y}\bar{z}\bar{w} + x\bar{y}zw$

(001) (010) (011) (100) (101)

Ex) $F = \Sigma(0,1,6,7)$, Find $\bar{F}$?!

1) Sum of minterm

2) product of Maxterm.

(لاقاء المجموعة غير الموجودة)

* اذا لم تغير الرموز Σ ← Σ وأخذ المتممة

* اذا لم تغير الرموز Σ ← Π تنزل لنفس الرقم

Σ ← Π

1) Σ → Σ ∴ متممة $\bar{F} = \Sigma(2,3,4,5)$

2) Σ → Π ∴ نقسم $\bar{F} = \Pi(0,1,6,7)$

(term) جمع الرموز بدلاً من

Ex) $F = xy + \bar{z}$, Convert to Canonical ?!

$xy + \bar{z}\cdot(x+\bar{x}) \Rightarrow xy + (\bar{z}x + \bar{z}\bar{x})\cdot(y+\bar{y}) \Rightarrow xy\cdot(z+\bar{z}) + \bar{z}xy + \bar{z}x\bar{y} + \bar{z}\bar{x}y + \bar{x}\bar{y}\bar{z}$

$\Rightarrow xyz + xy\bar{z} + xy\bar{z} + x\bar{y}\bar{z} + \bar{x}yz + \bar{x}\bar{y}\bar{z}$ (الحد يأخذ مرة $X+X=X$)

$xyz + xy\bar{z} + x\bar{y}\bar{z} + \bar{x}yz + \bar{x}\bar{y}\bar{z}$

[* ضرب ← (1) لأنه ← (SoP) بدورها (1) ← $(\bar{F})$]

Ex) $F = (x+y).(\bar{z})$ Convert to Canonical [(zero) يسمى (Pos) وكل (zero) تجيب له *]

$x+y+0 \Rightarrow (x+y+z\bar{z}) \Rightarrow (A+z).(A+\bar{z})$

$\therefore (x+y+z).(x+y+\bar{z})$ #

(zero) يسمى Pos

Ex) $F_1 = \Pi(0,1,5,6)$, $F_2 = xy+\bar{z}$, $F_3 = F_1 \oplus F_2$

Find F_3 as Sum of minterm ?! (zero) المطلوب

minterm	x	y	z	F_1	F_2	F_3
m_0	0	0	0	0	1	1
m_1	0	0	1	0	0	0
m_2	0	1	0	1	1	0
m_3	0	1	1	1	0	1
m_4	1	0	0	1	1	0
m_5	1	0	1	0	0	0
m_6	1	1	0	0	1	1
m_7	1	1	1	1	1	0

$F_3 = \Sigma(0,3,6)$

$F_3 = m_0 + m_3 + m_6$

$F_3 = \bar{x}\bar{y}\bar{z} + \bar{x}yz + xy\bar{z}$

000 011 110

* k-map

2 varible

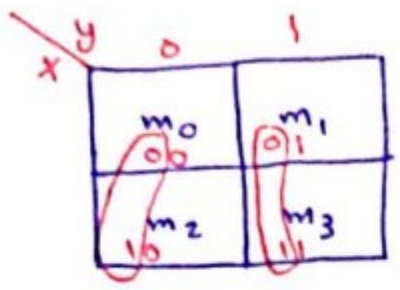
x,y → x y

0 0 m_0

0 1 m_1

1 0 m_2

1 1 m_3



* الخلايا المتجاورة تختلف بمتغير واحد فقط .

3 varible

(ترتيبها هكذا الترتيب)

x	y	z
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

x \ yz	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

* F = Σ(0,1,3), map ( ) ؟!

عدد المتغيرات

(sop)

x \ y	0	1
0	1	1
1	0	1

* $R = x \oplus y$

	x	y	R
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

x \ y	0	1
0	0	1
1	1	0

* F = π (0,1,5,7)

x \ yz	00	01	11	10
0	0	0	1	1
1	1	0	0	1

4 varible

xy \ zw	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

* $F = xyz + \bar{x}\bar{y} + \bar{x}$ (sop) (1) ← And → OR

1 1 1 0 0 0

x \ yz	00	01	11	10
0	1	1	1	1
1	0	0	1	0

* $F = x\bar{y}\bar{z}w + x\bar{y} + \bar{w}$ (sop)(1)

1 0 0 1 1 0 0

xy \ wz	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	1	0	0	1
10	1	1	1	1

* $F = (x+y)(x+\bar{z})$ (pos)(0), bar(1)

نفي x,y بغض النظر عن z — نفي y بغض النظر عن x,z

0 0 0 1

x \ yz	00	01	11	10
0	0	0	0	1
1	1	1	1	1

* of cells = 2^n → ni

عدد الخلايا

إذا كان (F) على شكل unstandard يحول لـ (standard) أو بنعمل (T.T)

* Grouping

① sop

x \ y	0	1
0	1	0
1	1	1

$(x + \bar{y})$

x \ y	0	1
0	1	1
1	1	0

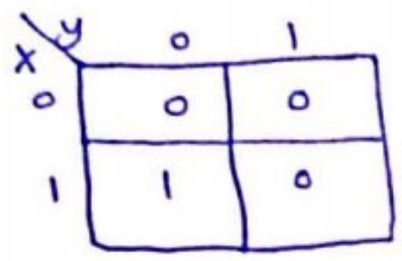
$(\bar{y} + \bar{x})$

① بندور على ① OR ② حسب النوع ③ المربعات ممكن أن تحتوي على خانات مكررة

④ اختر أقل عدد من المربعات ② اختر أكبر عدد من الخلايا ضمن هالمربع

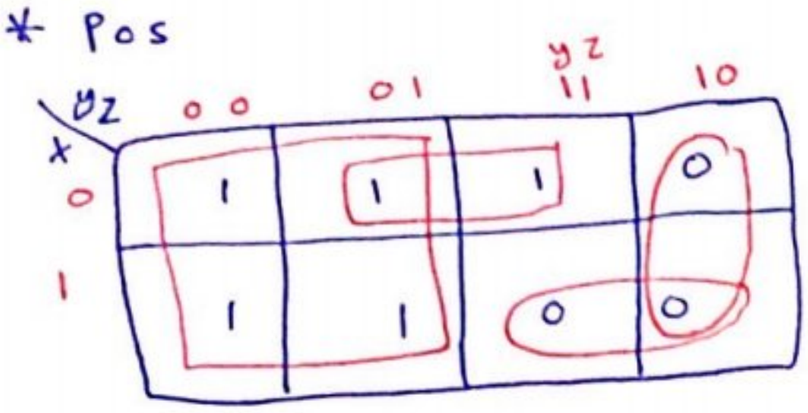
1

Pos



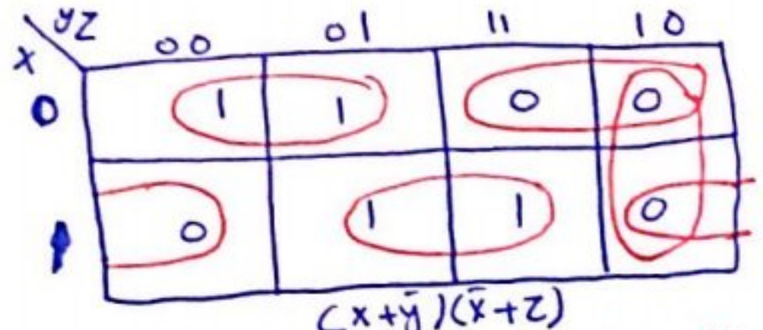
$(x.\bar{y})$

* Pos



Pos $(\bar{y}+z)(x+\bar{y})$

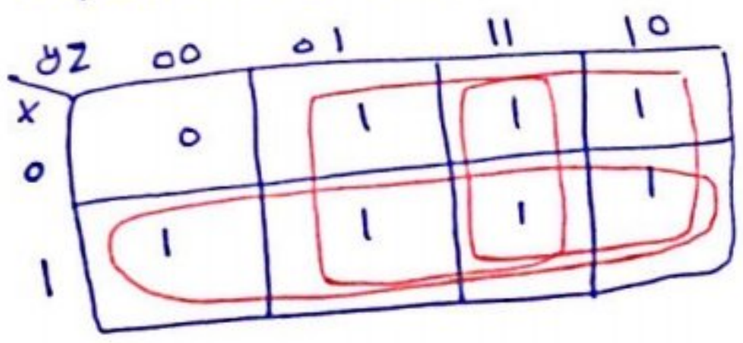
sop $(\bar{y}) + (\bar{x}z)$



$(x+\bar{y})(\bar{x}+z)$

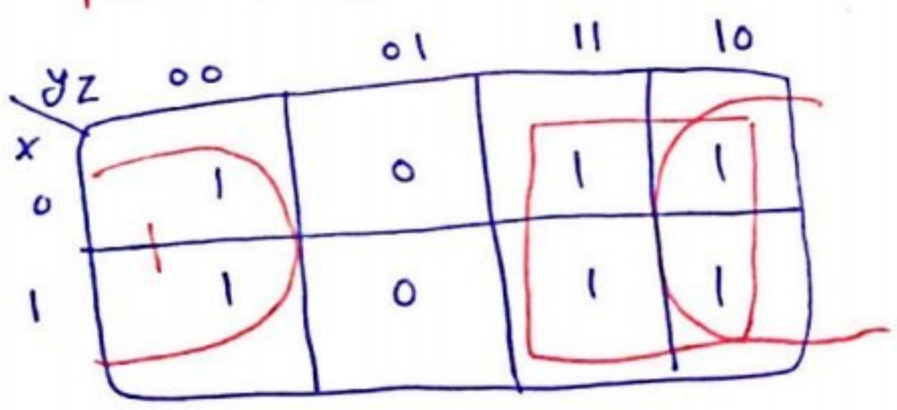
pos(0) → $(\bar{x}+z)(\bar{y}+z)(x+\bar{y})$

sop(1) → $(\bar{x}\bar{y}) + (xz)$



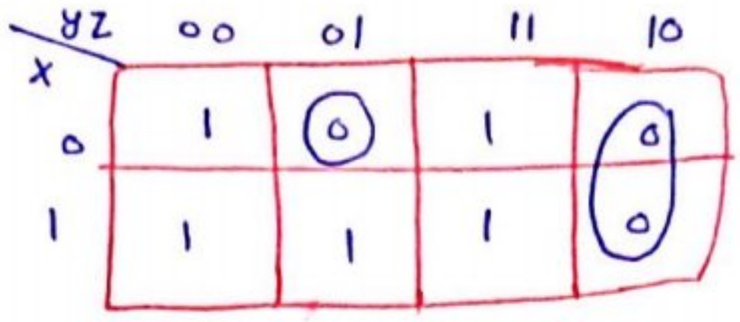
sop → $x+y+z$

pos → $x+y+z$

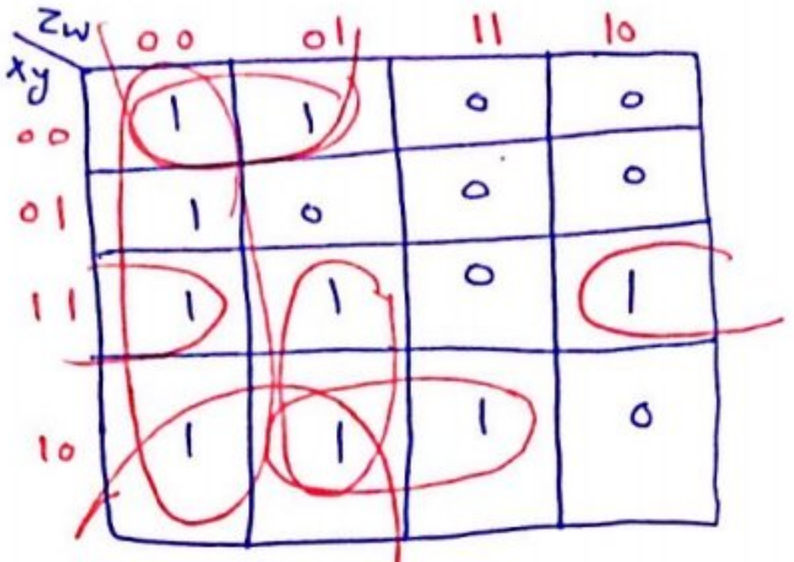


sop → $(y) + (\bar{z})$

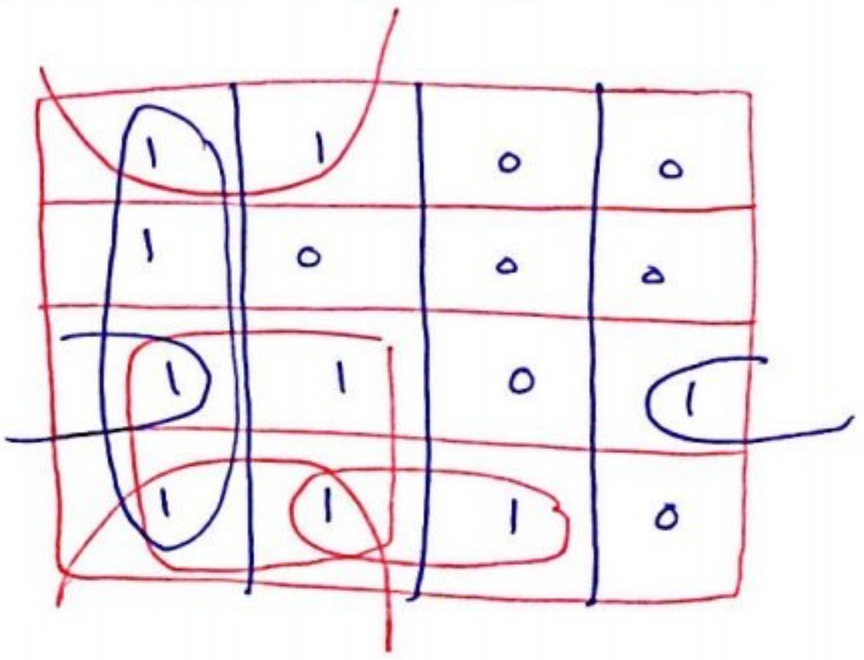
pos → $y+\bar{z}$

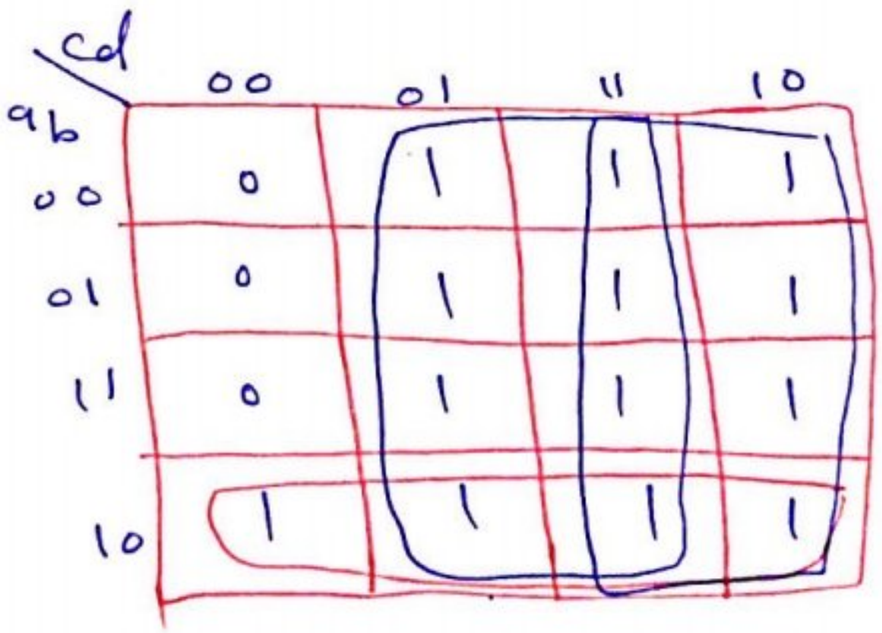


$Pos = (\bar{y}+z)\cdot(x+y+\bar{z})$



sop $= (\bar{z}\bar{w}) + (\bar{y}\bar{z}) + (x\bar{y}\bar{w}) + (x\bar{z}) + (x\bar{y}w)$

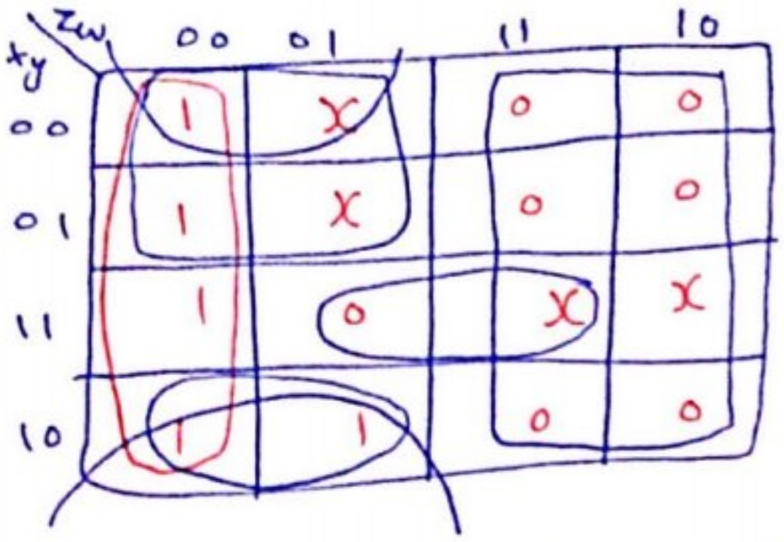




sop $= (x\bar{y}) + (z) + (w)$

$(a\bar{b}) + d + c$

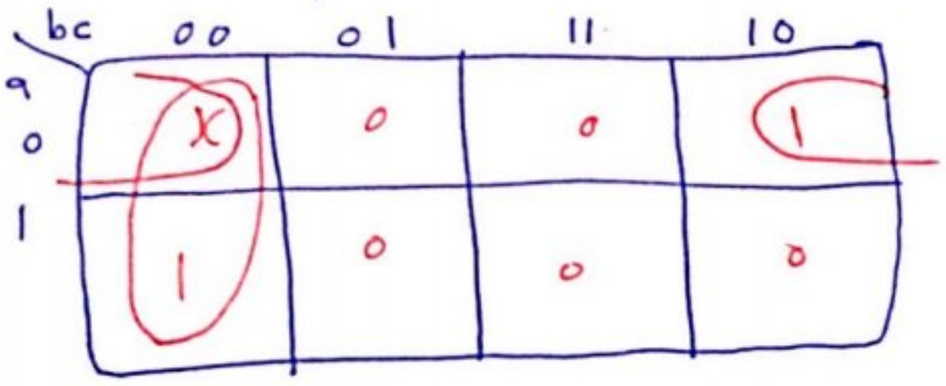
* dont care



$sop = (x\bar{z}) + (\bar{z}\bar{w}) + (x\bar{y}\bar{z})$

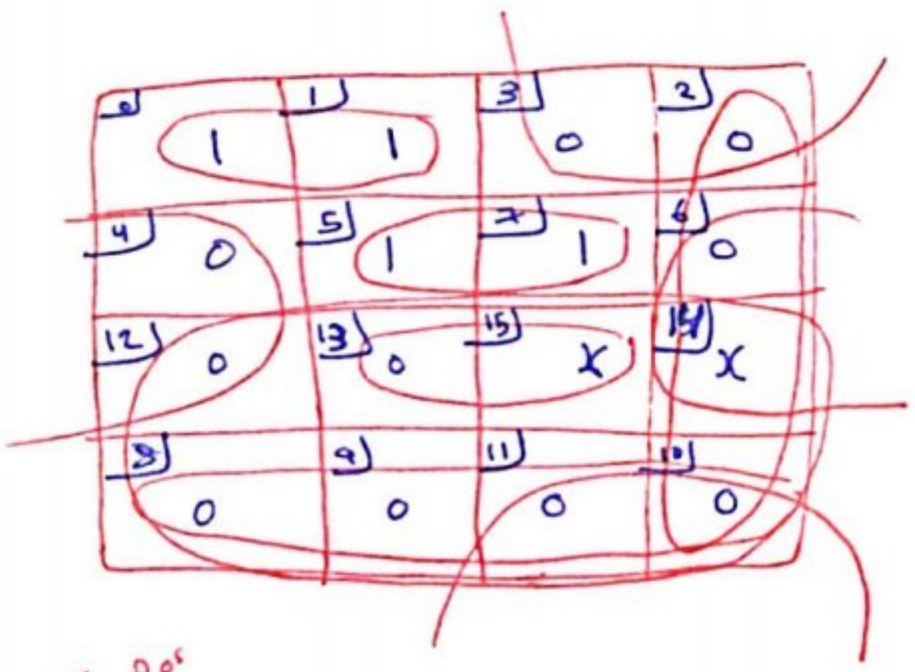
$sop = \bar{z}\bar{w} + \bar{y}\bar{z}$

$pos \rightarrow (\bar{z})(\bar{x} + \bar{y} + \bar{w})$



$sop = \bar{b}\bar{c} + \bar{a}\bar{c}$

* $F = \Sigma(0,1,5,7) + d\Sigma(14,15)$



.as35 por

* Design of Combinational circuits

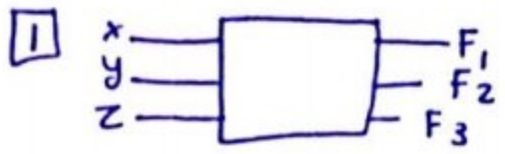
Given : description (الوصف)

Required : circuit

1. Define input, output
2. T. T
3. k-map
4. Draw circuit

Ex:- Design 3-bit binary to Gray code converter

1

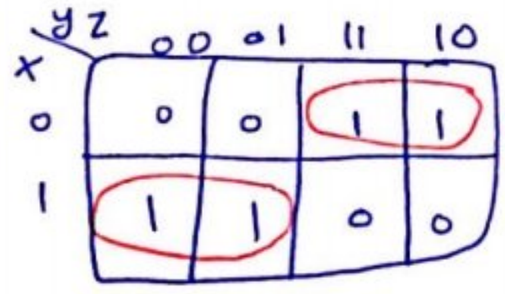


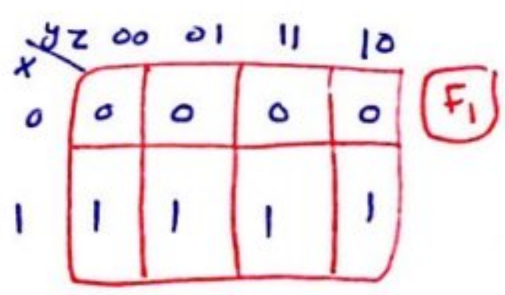
$\bar{x}\bar{y}\bar{z} \rightarrow \bar{F_1}\,\bar{F_2}\,\bar{F_3}$

[B → G (B,B)]

2

x	y	z	F1	F2	F3
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

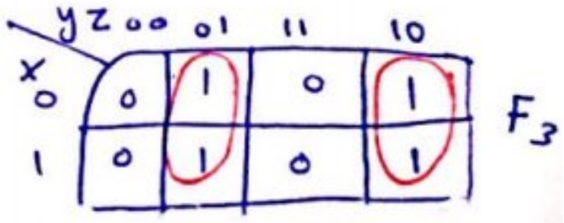




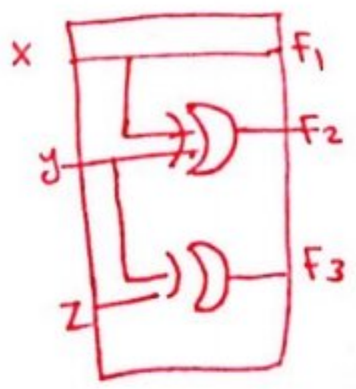
$F_1 = x$

$F_2 = (x\bar{y}) + (\bar{x}y)$

$x \oplus y$



$F_3 = \bar{y}z + y\bar{z} \rightarrow y \oplus z$



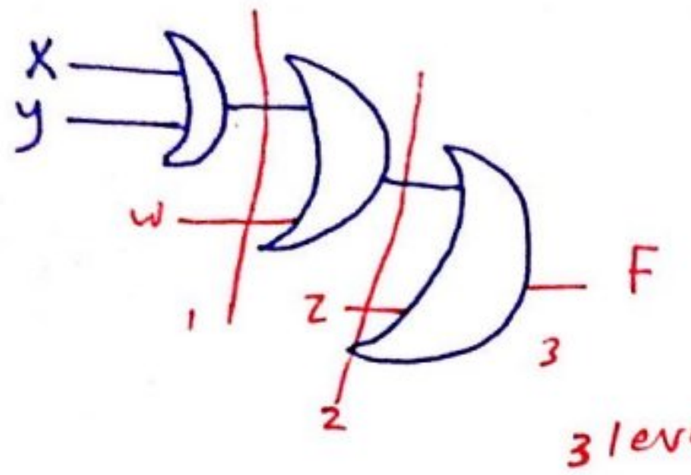
لو بدنا نرسم (k-map) لطلع الحلقة كذا

أو جيت أول Gray ⊕

$G = B[i+1] \oplus B[i]$

$B = B[i+1] \oplus G[i]$

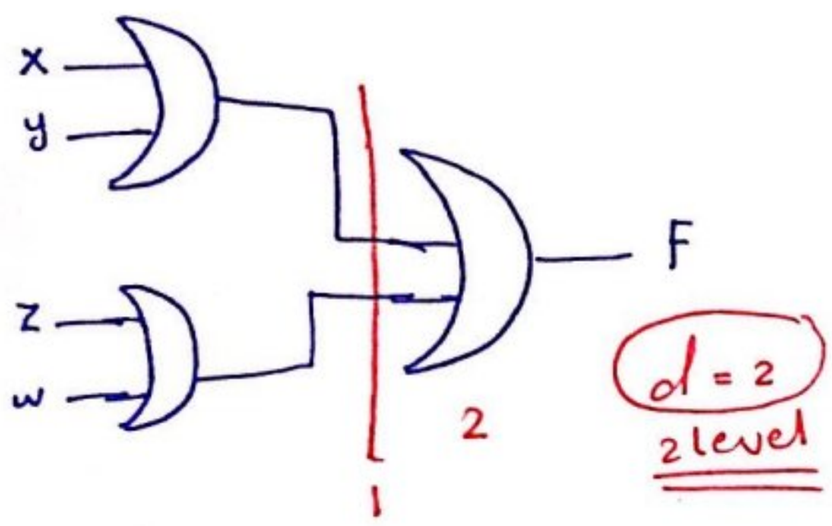
* $F = X + y + w + z$, Design the circuit if only you have 2-input OR gates



3 level

d = 3

level = d



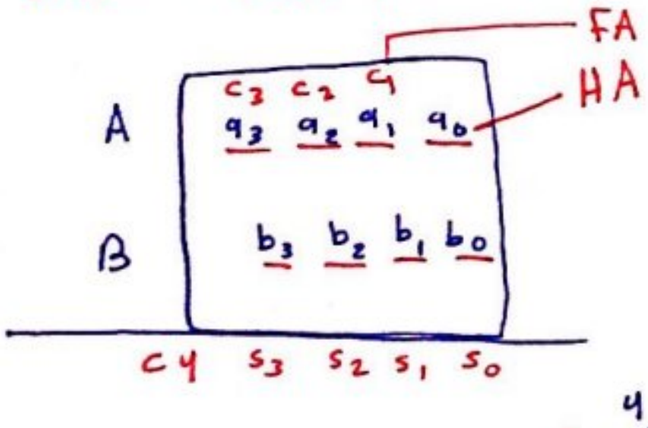
d = 2

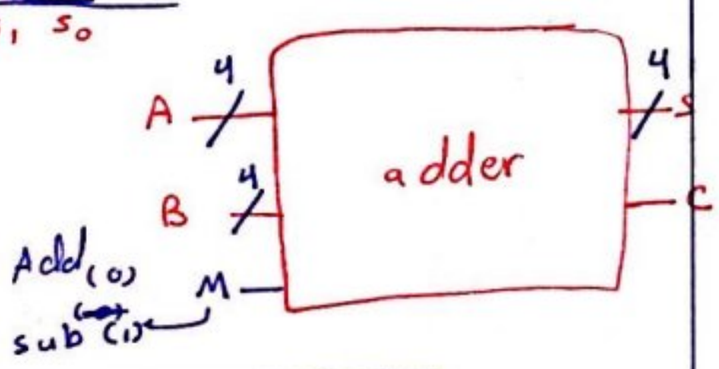
2 level

* Adder / subtracter (lect: "8")

* مناقشة العدد الأول والثاني.

Design a 4 Bit Adder / sub using 2's comp.





Half adder

2 input → 2 out

Full adder

3 input → 2 out

* Half adder

A	S
B	C

A B	cout S
0 0	0 0
0 1	0 1
1 0	0 1
1 1	1 0

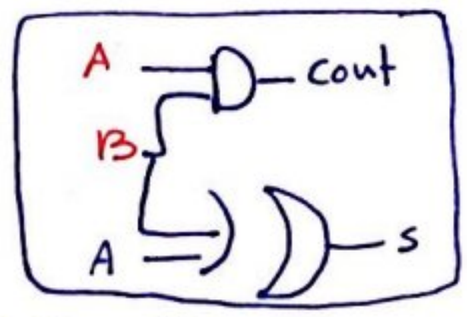
B	0	1
A 0	0	0
1	0	(1)

نأخذ cout = A · B

Q	0	1
A 0	0	(1)
1	(1)	0

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

نأخذ $S = A \oplus B$



* Full adder

inp	out
A	S
B	cout
cin	

A	B	cin	Cout	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

B cin	00	01	11	10
A 0	0	0	(1)	0
1	0	(1)	(1)	(1)

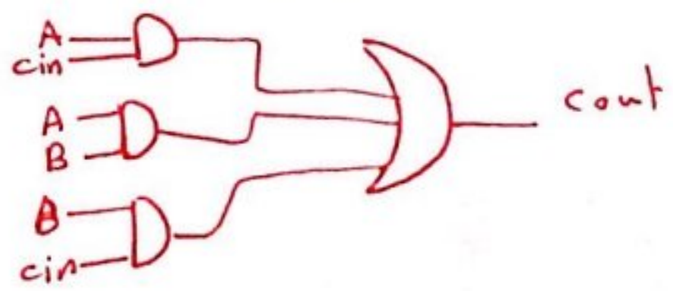
** نأخذ $cout = A\,cin + AB + B\,cin$

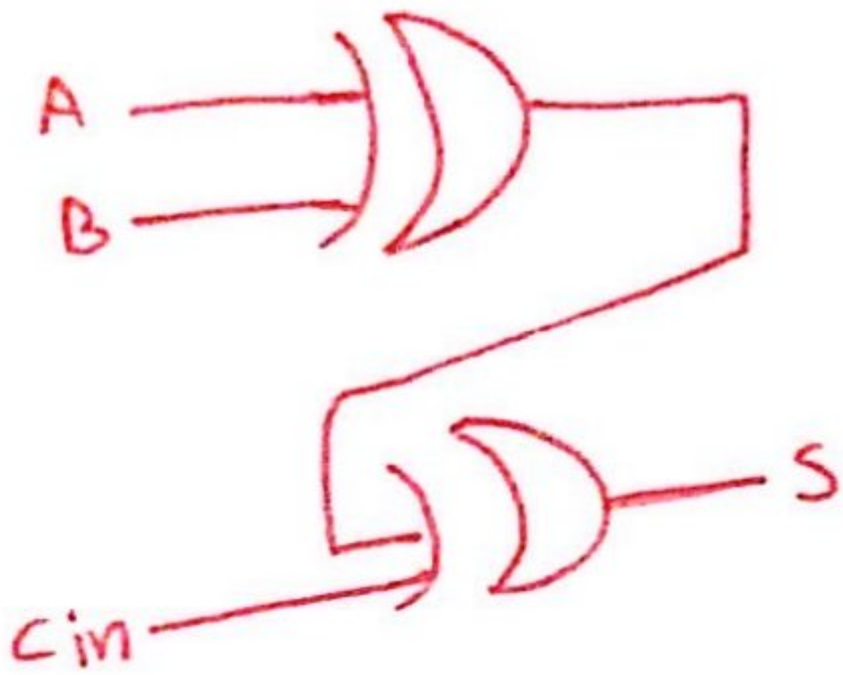
Bcin	00	01	11	10
A 0	0	(1)	0	(1)
1	(1)	0	(1)	0

*** $S = AB\bar{cin} + \bar{A}B\bar{cin} + AB\,cin + \bar{A}\bar{B}\,cin$

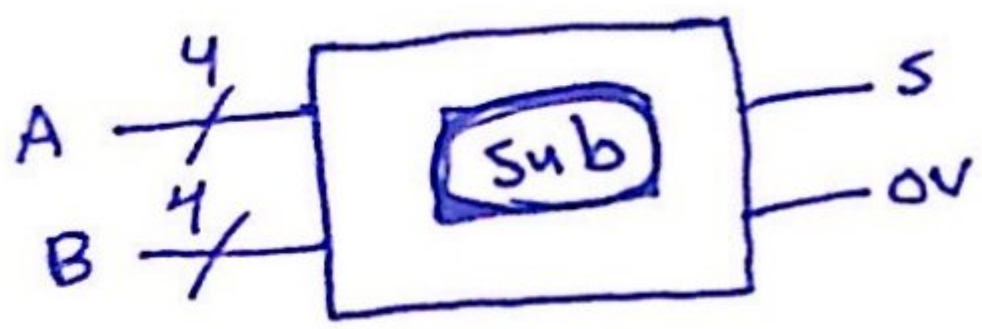
صفر — اختصار

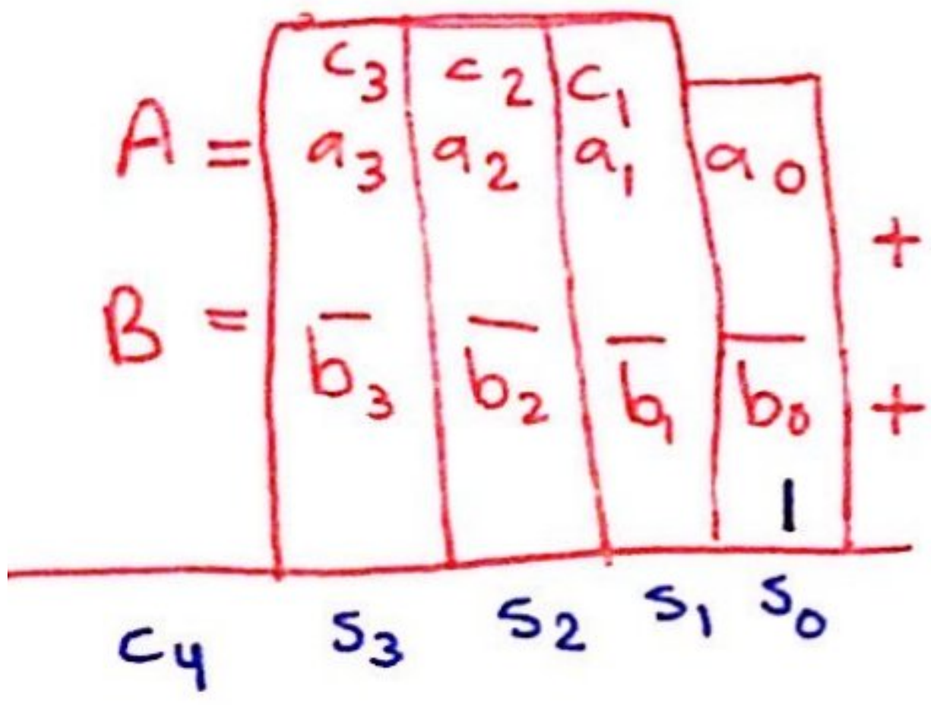
$S = A \oplus B \oplus cin$





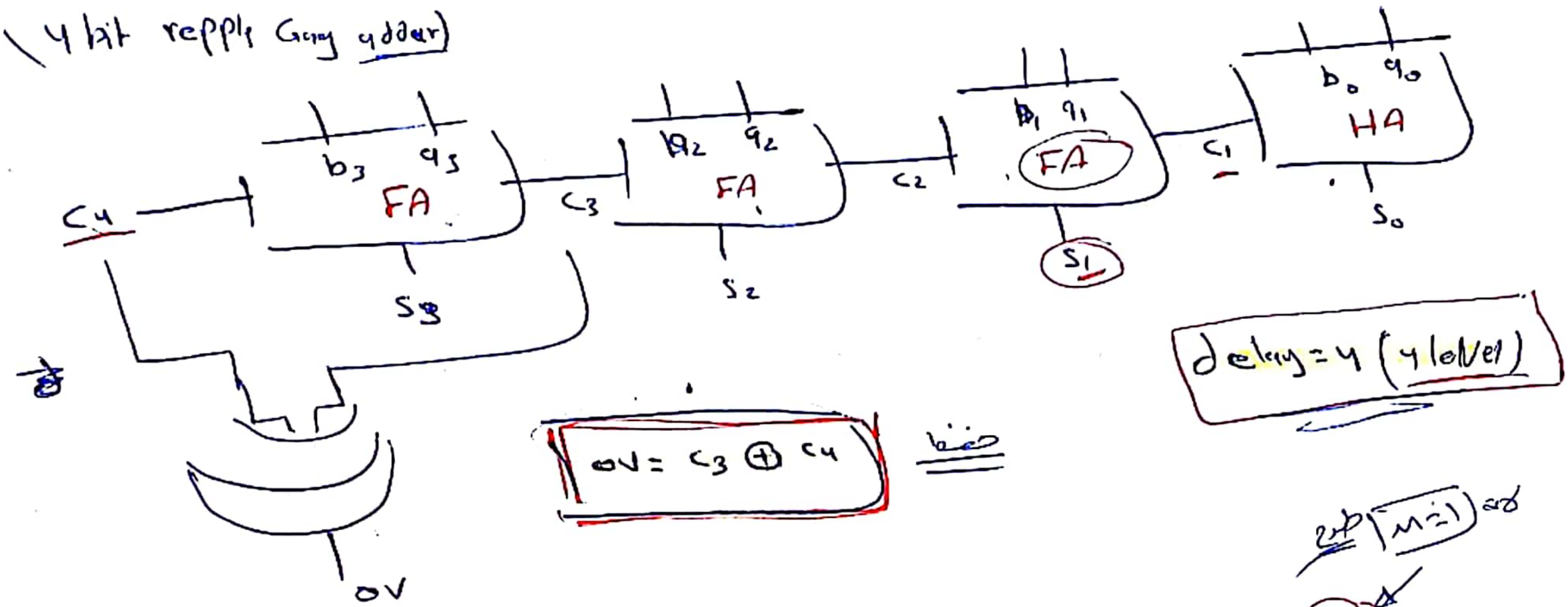
* subtracter





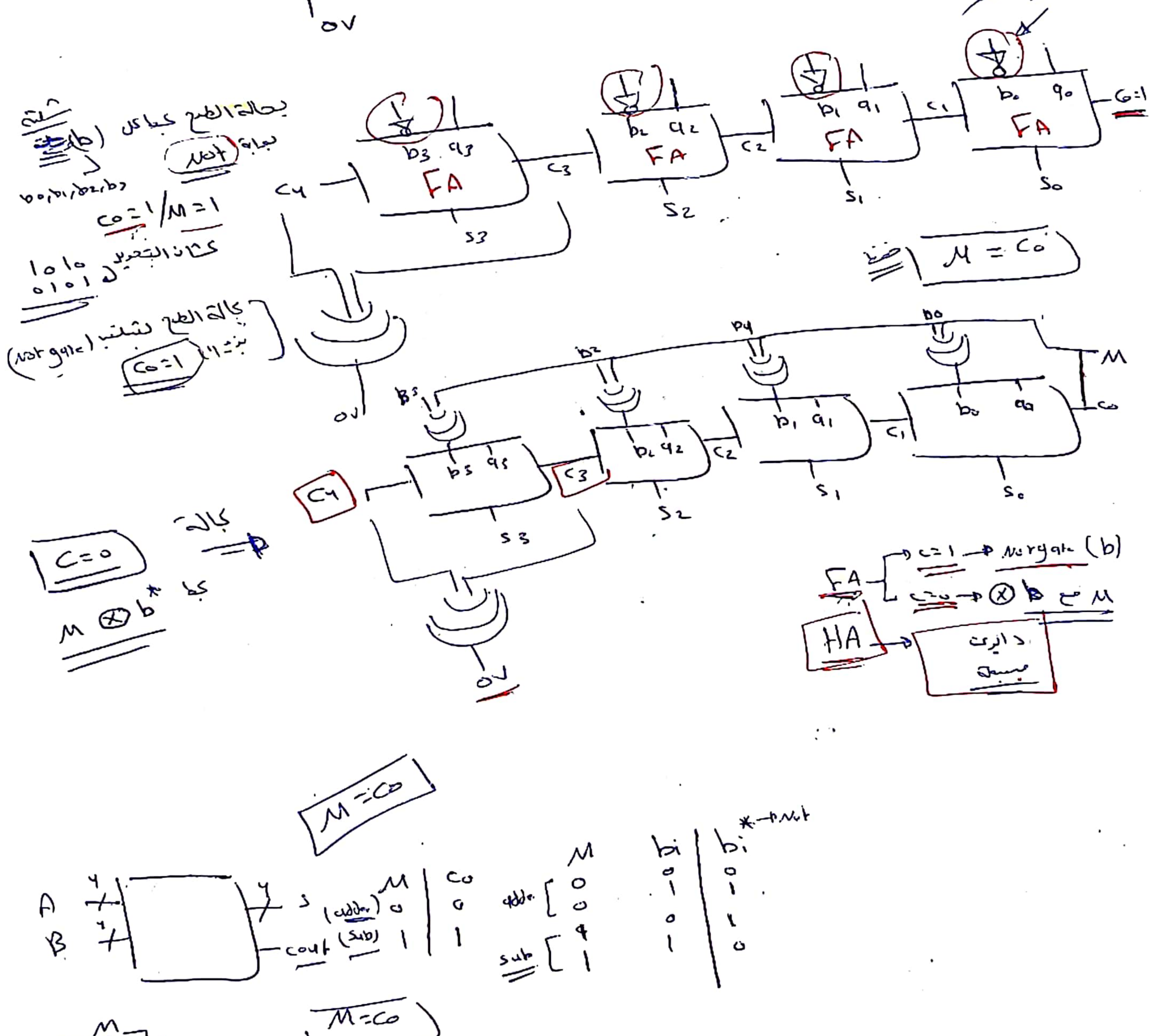

```
  1101   -
  0101
-------
  1101
  1010 +
     1 +
-------
 11000
```

4 bit ripple Carry adder



delay = 4 (4 level)

OV = C3 ⊕ C4



M = C0

C0 = 1 / M = 1

1010
0101

C=0 ← M ⊕ b*

M ⊕ b* ← C0 = 1

(b) Nergate ← C = 1

M مع b ⊕ ← 0 = C

FA

HA → دارة مبسطة

5

A = 1001
B = 0101
M = 1

Find S, OV, Cout ?!

A	1001	
B	0101	-
	1001	
	1010	
	1	+
	1 0100	

cout = 1, S = 0100

cin ≠ cout ⇒ OV

OV = 1

* Design a circuit that compares between 2 unsigned Number (4bits)

أقارن Componens

>
=
<

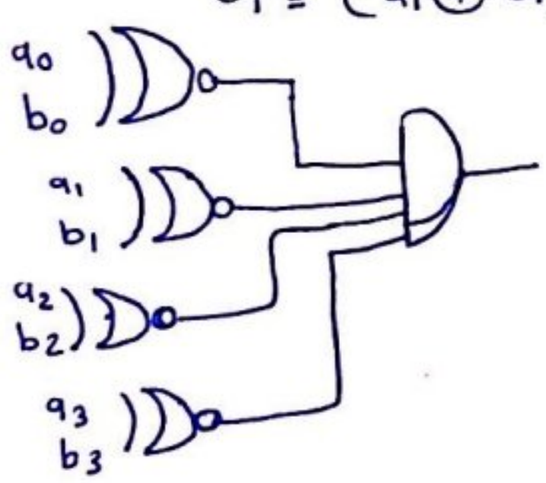
eq A $q_3\ q_2\ q_1\ q_0$

B $b_3\ b_2\ b_1\ b_0$

a_i	b_i	eq
0	0	1
0	1	0
1	0	0
1	1	1

التشابه = 1

$eq = \overline{(a_i \oplus b_i)}$



MSB $q_3 > b_3$

لطلع هو المنطقي بعد ($q_3 = b_3$) بشرط إذا $x_3 = 1$

(b<a) بعمل مقارنة

Gr A $a_3\ a_2\ a_1\ a_0$

B $b_3\ b_2\ b_1\ b_0$

المقارنة من الشمال لليمين.

a_i	b_i	Gr
0	0	0
0	1	0
1	0	1
1	1	1

(الأكبر a_i)

$$Gr = a_3\bar{b_3} + x_3 a_2 \bar{b_2} + x_3 x_2 a_1 \bar{b_1} + x_3 x_2 x_1 a_0 \bar{b_0}$$

Ls

$$Ls = \bar{a_3} b_3 + x_3 \bar{a_2} b_2 + x_3 x_2 \bar{a_1} b_1 + x_3 x_2 x_1 \bar{a_0} b_0$$

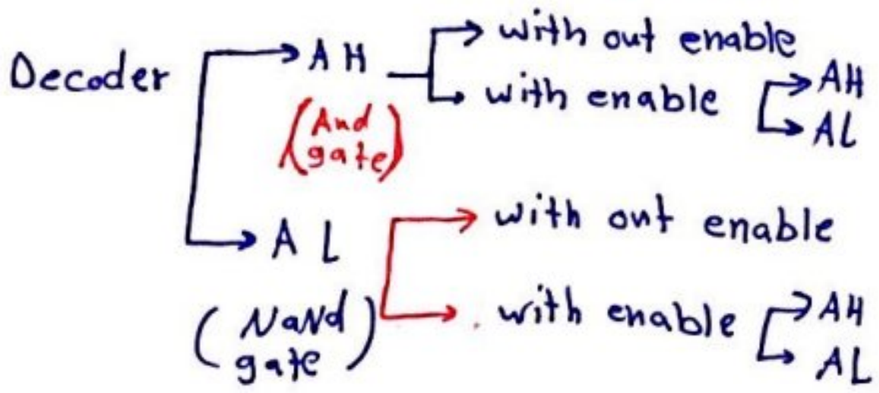
Gr	eq	Ls
0	0	1 ← لأن (1)
0	1	0
1	0	0
1	1	X

select "9"

Binary code → (يعطي) Binary value → Decoder

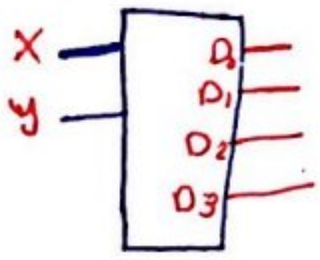
Binary value → (يعطي) Binary code → Encoder

(ضغط)



* Active High decoder without enable

2-to-4 / 3to-8 / 4-to-16 [n to 2^n]

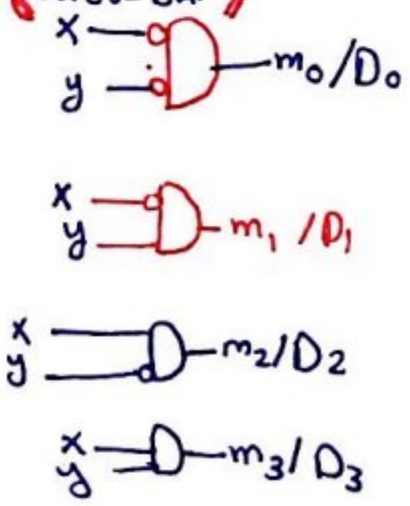


2 - 4
3 - 8
4 - 18
5 - 23

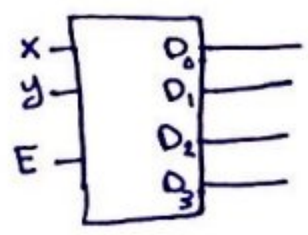
$D_0 = m_0$

	x	y	D_0	D_1	D_2	D_3
m_0	0	0	1	0	0	0
m_1	0	1	0	1	0	0
m_2	1	0	0	0	1	0
m_3	1	1	0	0	0	1

(zero de Bar)



* AH with enable (اضوي السيركت أو اطفيه) AH switch



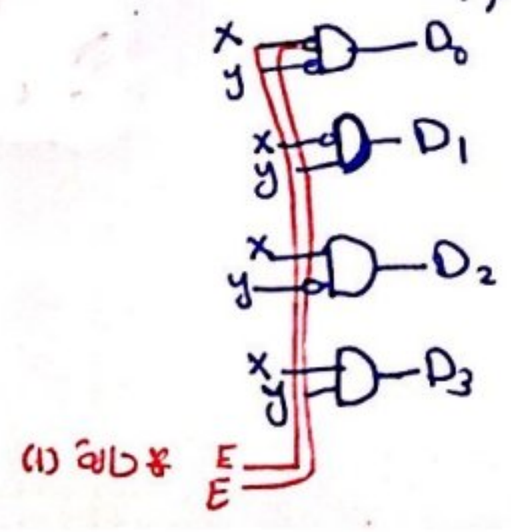
E	x	y	D_0	D_1	D_2	D_3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
0	x	x	0	0	0	0

طبيعي (1) AH E
اطفي (0)

أوقات بدي مفتاحين (E) حسب الشروط ،

① إذا بدي الاثنين صح بشبكهم مع And .

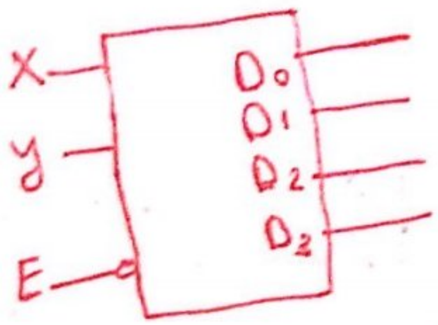
② إذا بدي واحد صح بشبكه مع OR



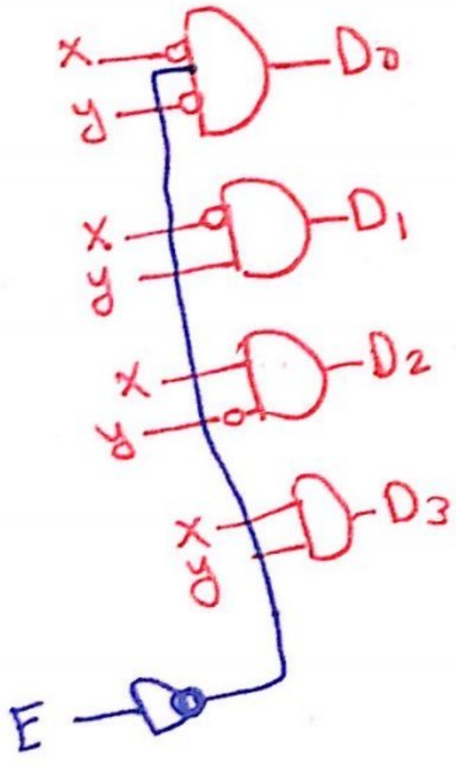
* حالة (1)

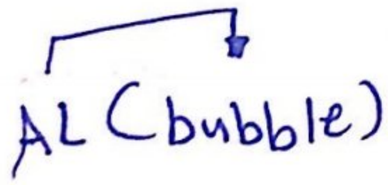
* AH with enable AL

AL
(0) طبيعي
(1) طفي

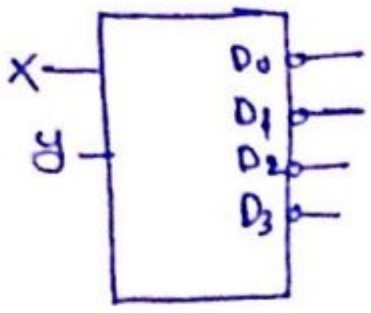


E	x	y	D_0	D_1	D_2	D_3
0	0	0	1	0	0	0
0	0	1	0	1	0	0
0	1	0	0	0	1	0
0	1	1	0	0	0	1
1	x	x	0	0	0	0





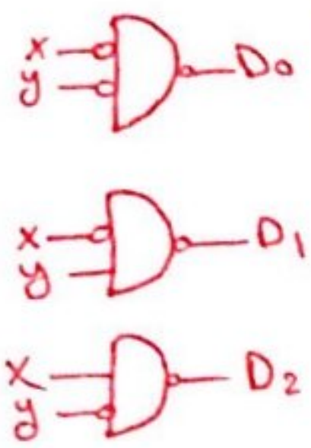
Al without enable (Decoder)



x y	D_0	D_1	D_2	D_3
0 0	0	1	1	1
0 1	1	0	1	1
1 0	1	1	0	1
1 1	1	1	1	0

عكس AH

Nand gate



$D_0 = \bar{m}_0 = \overline{\bar{x}.\bar{y}} = x + y$

$D_1 = \bar{m}_1 = \overline{\bar{x}y} = x + \bar{y}$

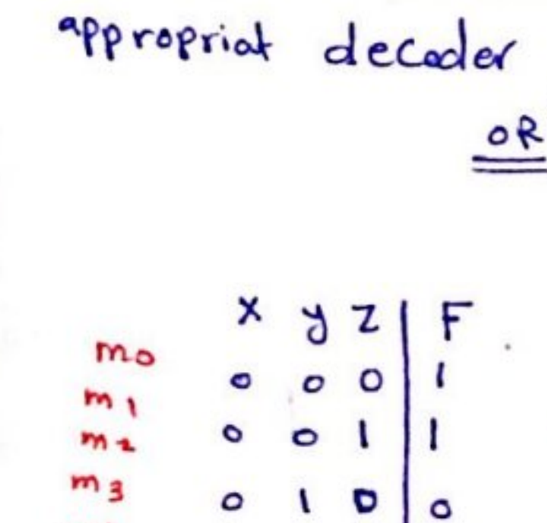
$D_2 = \bar{m}_2 = \overline{x\bar{y}} = \bar{x} + y$

$D_3 = \bar{m}_3 = \overline{xy} = \bar{x} + \bar{y}$

* AL with enable AH طبعي zero (1) بطفي

* AL with enable AL طبعي (0) ① بطفي

AL [2]



inp 101

[5] [1]

[2] AL Decoder (5) كل وطلوب ماعدا

[3] E_1, E_2

طبعي

x y z

1 0 1

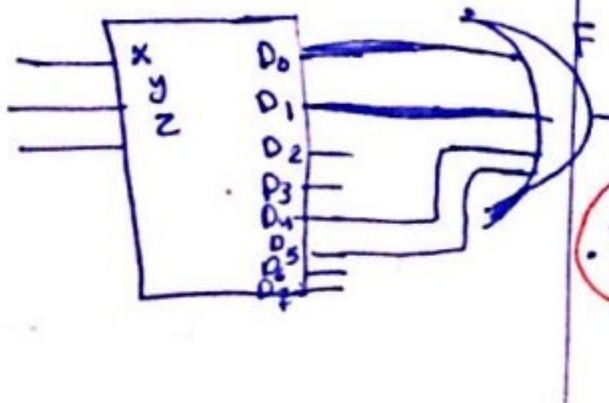
$\Sigma = 1$

Ex;- Implement the following function using the appropriat decoder $F = \Sigma(0,1,4,5)$

OR و and

and decoder بدل OR*

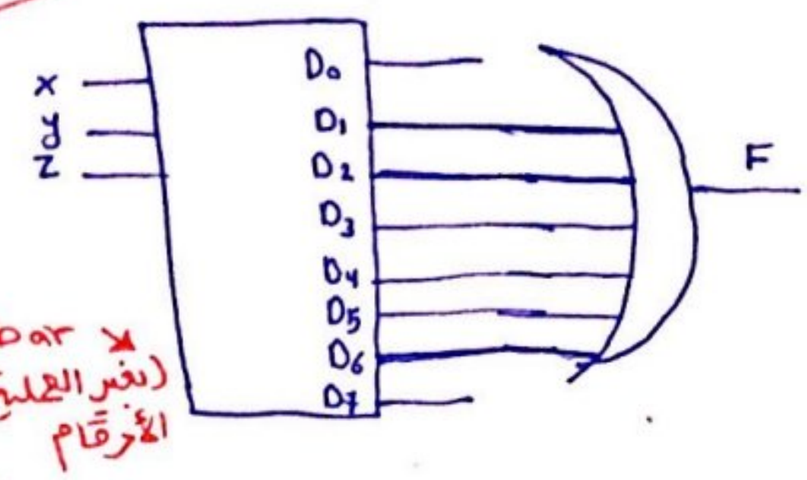
	x	y	z	F
m_0	0	0	0	1
m_1	0	0	1	1
m_2	0	1	0	0
m_3	0	1	1	0
m_4	1	0	0	1
m_5	1	0	1	1
m_6	1	1	0	0
m_7	1	1	1	0



Ex: Implement : $F = \Sigma(1,2,3,4,5,6)$ and & or

using the appropriat

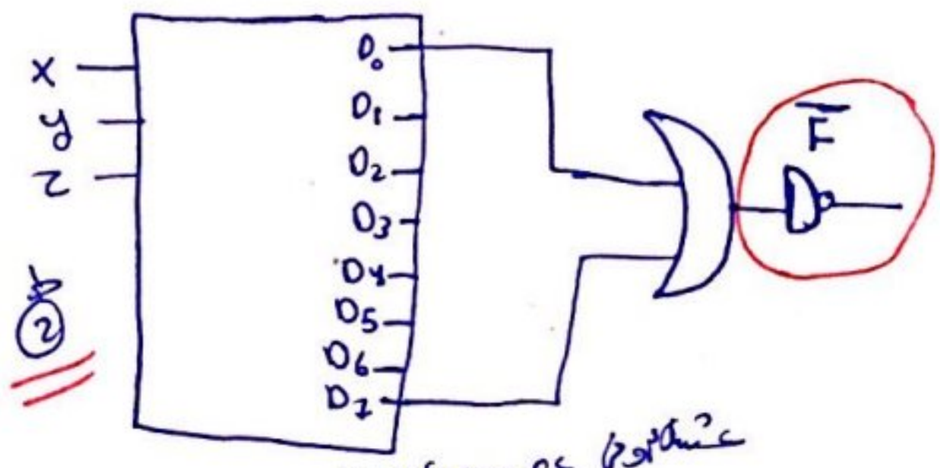
① b



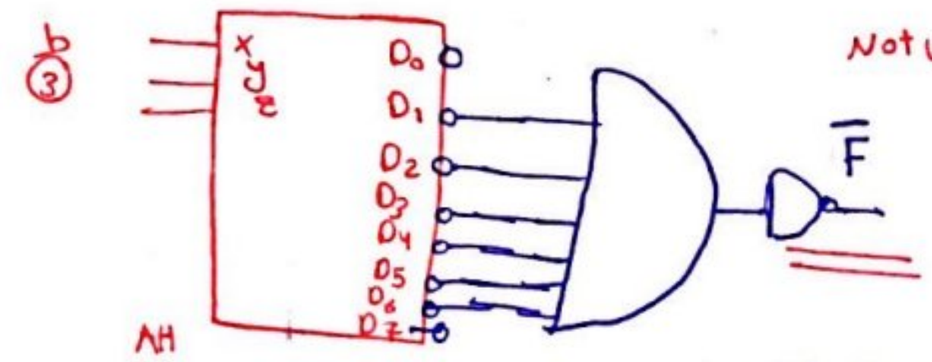
OR خلي الخلية وناخد صفقة الأرقام. حل ثاني

$F = \Sigma(1,2,3,4,5,6)$

$\bar{F} = \Sigma(0,7)$



عشانكون على حاي الأسلوب



فائتني Not

$F = \Sigma(1,2,3,4,5,6)$

$F = \Pi(1,2,3,4,5,6)$

and تم (OR)

(AL) bubble

حوفيلشو bar OR{ and

and تم OR (Π) بدو

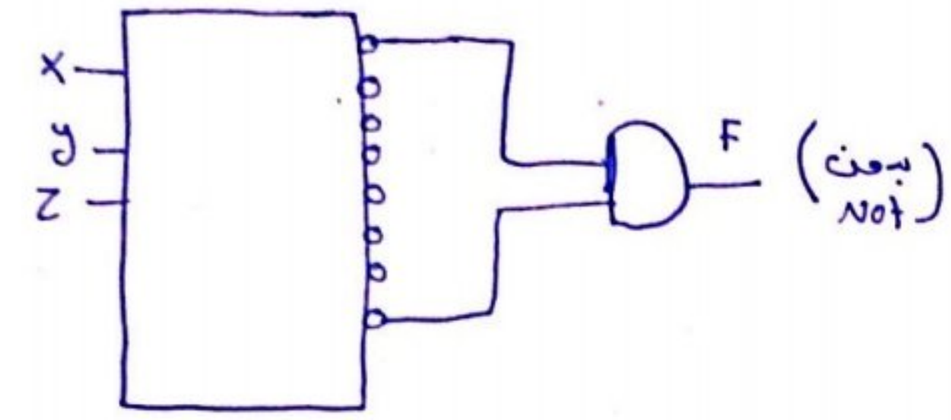
* غيرت الخلية وميزة الأرقام وما حطيت bar

F إذا غيرت وحدة من الخلية أو الأرقام كلها

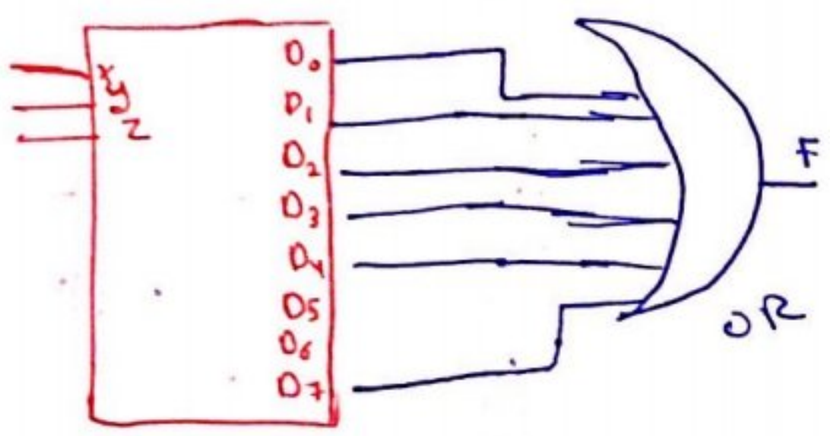
.F على bar

$F = \Sigma(1,2,3,4,5,6)$

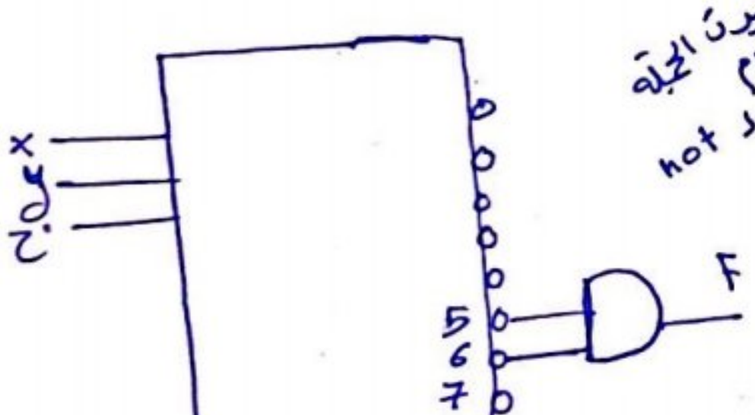
$f = (0,7)$



Ex: $F = \Sigma(0,1,2,3,4,7)$ implemant this function.



$F = \Pi(5,6)$

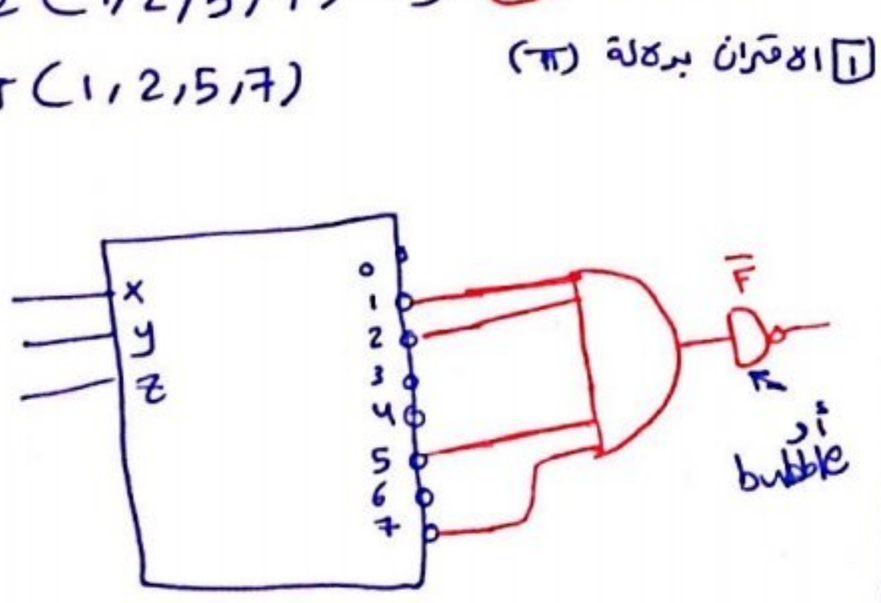


عنيزة الجلة * والأرقام not د وشلة ع

$F = \Sigma(1,2,5,7)$ usig AL decoder

AI Nand

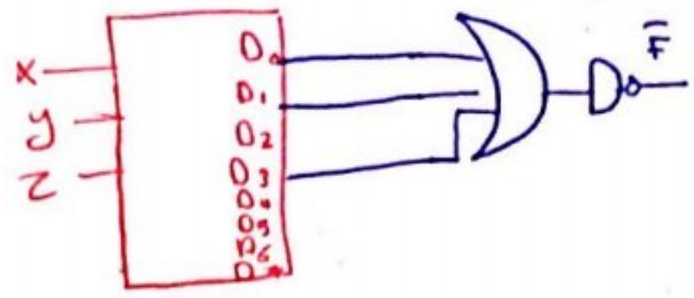
$\bar{F} = \pi(1,2,5,7)$ (π) الاقتران بروكة 1



$F = \Sigma(0,1,5)$ / Active high Decoder

$\bar{F} = \Sigma(0,1,5)$ (Σ) $\Sigma(2,3,4,6,7)$ بدون Not

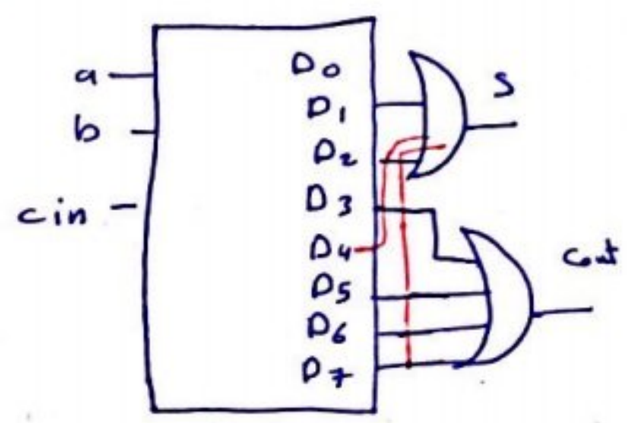
and / OR



* Design a full adder usiig OR gates and 3x8 active high decoder

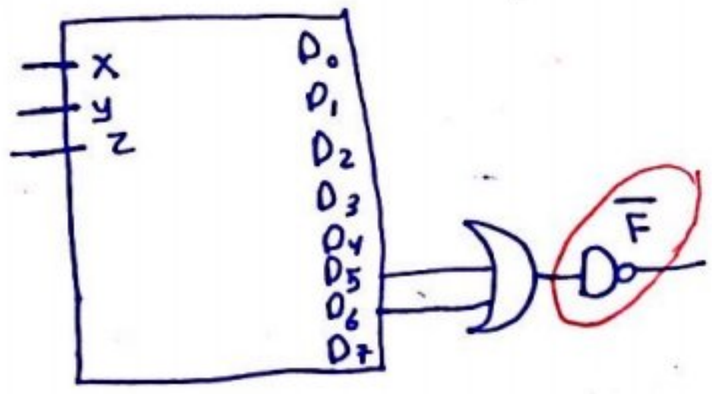
a,b,cin OR AH فقط بدون GR

	a	b	cin	cout	s
m_0	0	0	0	0	0
m_1	0	0	1	0	1
m_2	0	1	0	0	1
m_3	0	1	1	1	0
m_4	1	0	0	0	1
m_5	1	0	1	1	0
m_6	1	1	0	1	0
m_7	1	1	1	1	1

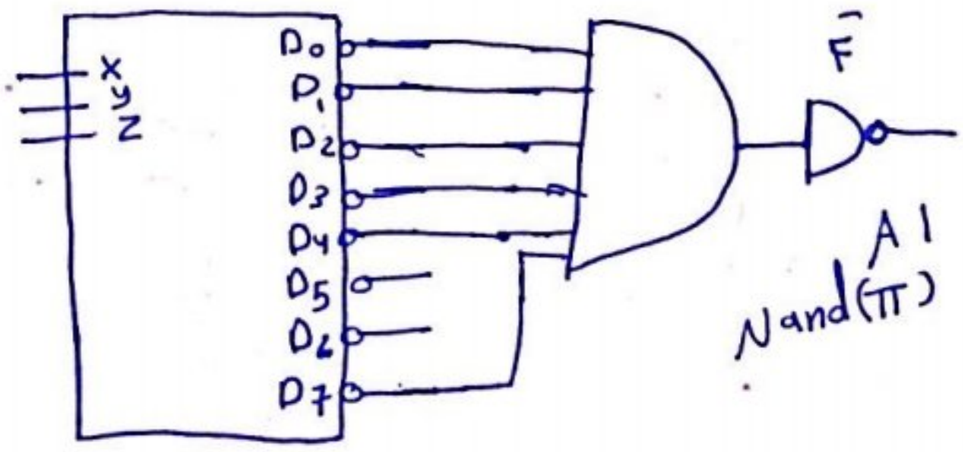


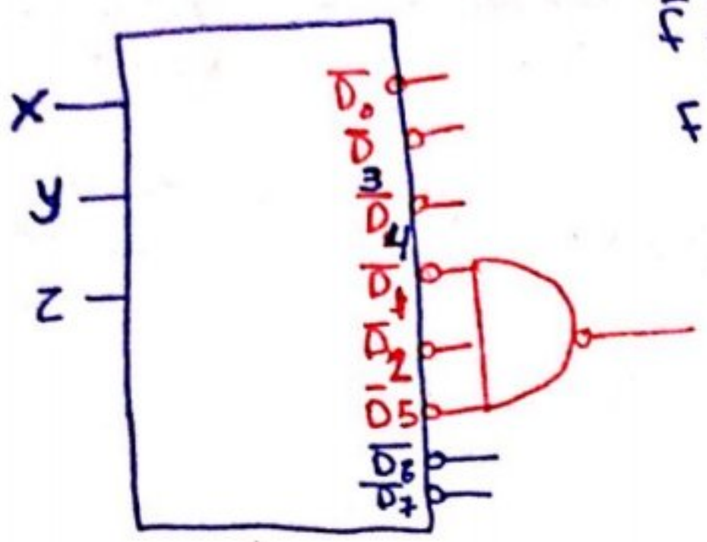
$[cout = \Sigma(3,5,6,7)$
$s = \Sigma(1,2,4,7)]$

* $\bar{F} = \Sigma(5,6)$



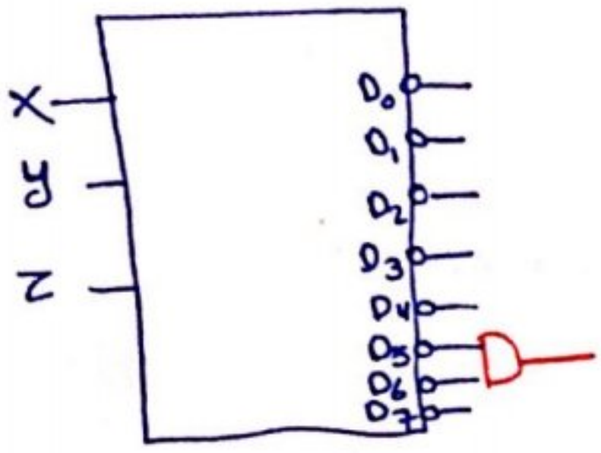
* $\bar{F} = \pi(0,1,2,3,4,7)$





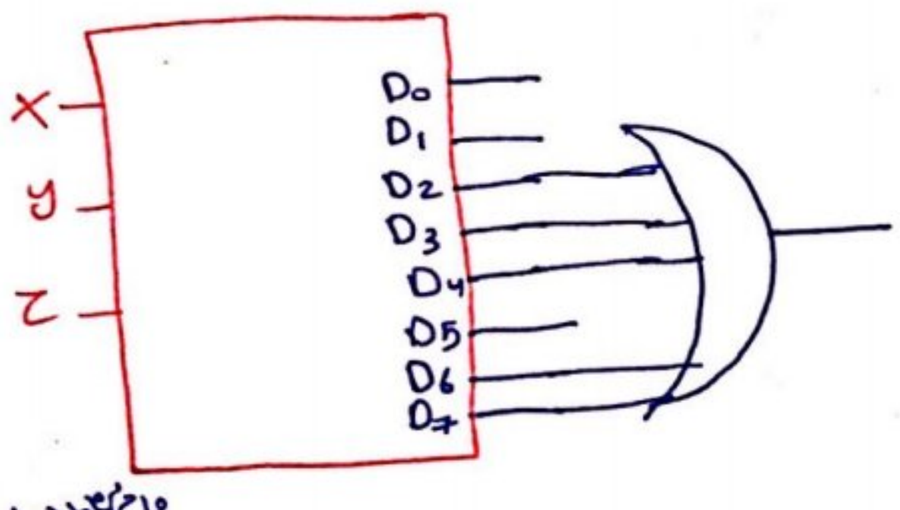
$\bar{F} = \pi(1,2,5)$

$F = \Sigma(1,2,5)$



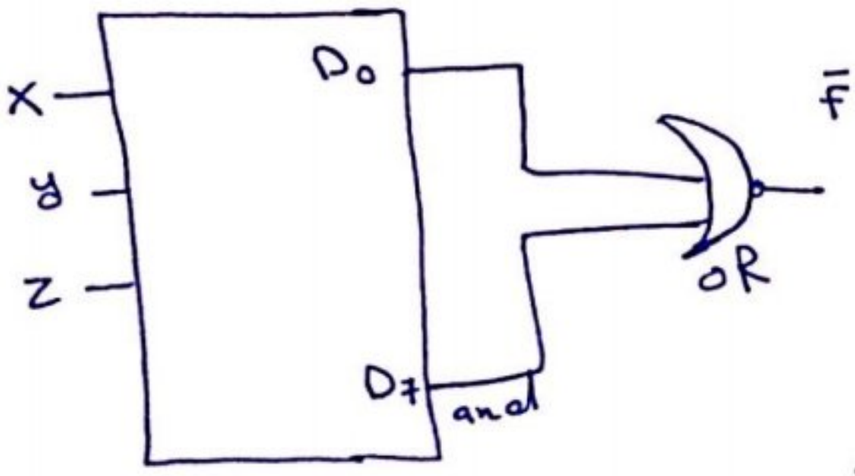
$F = \pi(5,6)$

$F = \Sigma(0,1,2,3,4,7)$



في حالة bubble إذا

$f = \Sigma(2,3,4,6,7)$



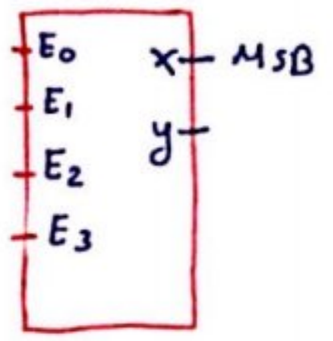
$\bar{F} = \Sigma(0,7)$

$\bar{F} = \Sigma(0,7)$ / $F = \Sigma(1,2,3,4,5,6)$

* بشبه العملية دهس الأرقام.
* غير العملية وشبه الأرقام.

(EncoDER) وعكسه decoder

[2-1 / 4-2 / 8-3 / 16-4]



E_0	E_1	E_2	E_3	x	y
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

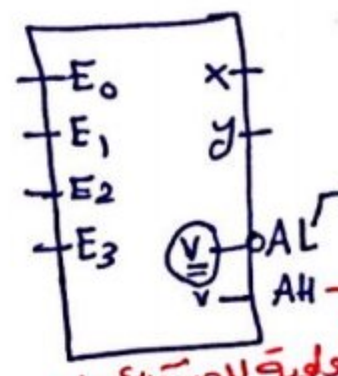
$X = E_2 + E_3$

$y = E_1 + E_3$

* output

* Priority encoder

أولوية بالكل :-



حالتين zero — أعلى الأولوية للأقل

حالتين 1 — أعلى الأولوية للصفر الأعلى

الأقل E_0	E_1	E_2	الأكبر E_3	V	x	y
1	0	0	0	1	0	0
x	1	0	0	1	0	1
x	x	1	0	1	1	0
x	x	x	1	1	1	1
0	0	0	0	0	x	x

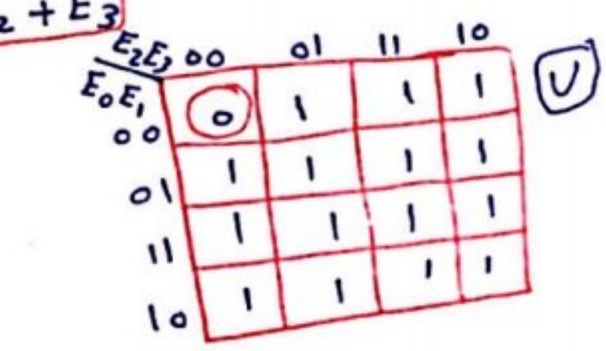
(4 صفوف)

بدور على الأعلى ← (V)

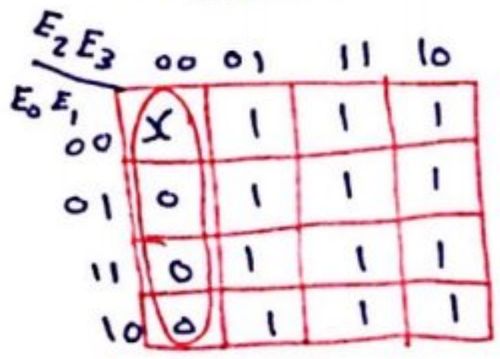
(V) بدور على الأقل

E_3 (الأقل) E_0 (الأكبر)

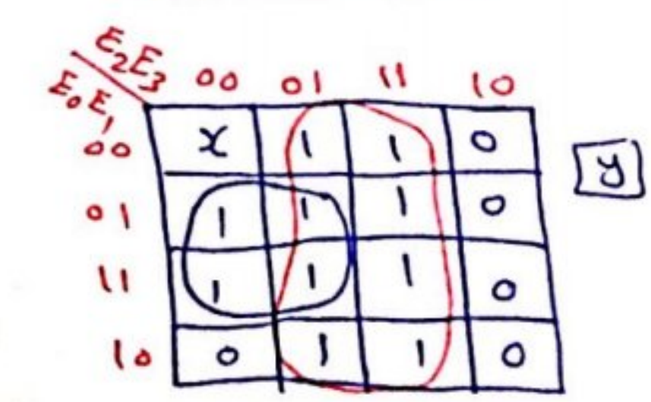
$V = E_0 + E_1 + E_2 + E_3$



$X = E_2 + E_3$



$y = E_3 + E_1 \cdot \bar{E}_2$



[١] AH Decoder :
١) اجمع الحدود لـ (F) باستخدام OR
٢) اجمع الحدود لـ ($\bar{F}$) باستخدام NOR .

[٢] AL Decoder :
١) اجمع الحدود لـ ($\bar{F}$) باستخدام AND
٢) اجمع الحدود لـ (F) باستخدام NAND

* AH / AL Decoder (output bubble)

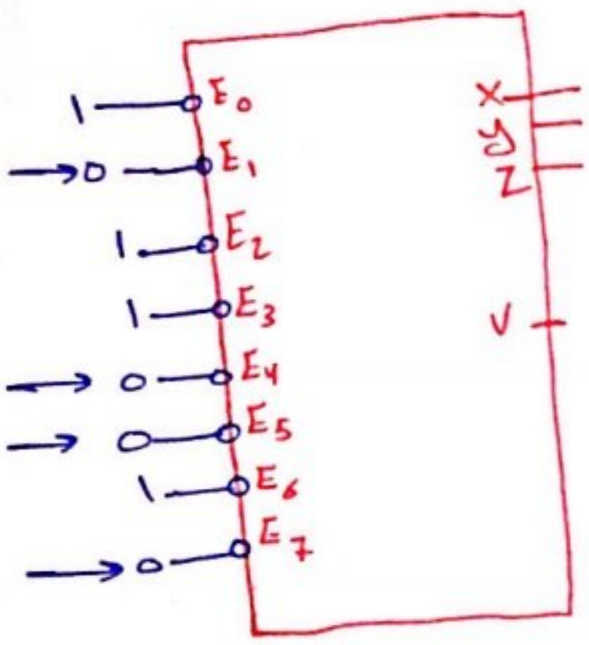
. (2×4 outputs) bubble 2×4 Decoder

4 x 2 inputs bubble Encoder
4 x 2

أو bubble كل الأكبر .

* Example Encoder

Find x, y, z, V ؟!



بالتدوير ٥

AL يعني فيه zero

AH يعني one

(V) (AH)

لأنه أنا بدور على أكبر zero

E7

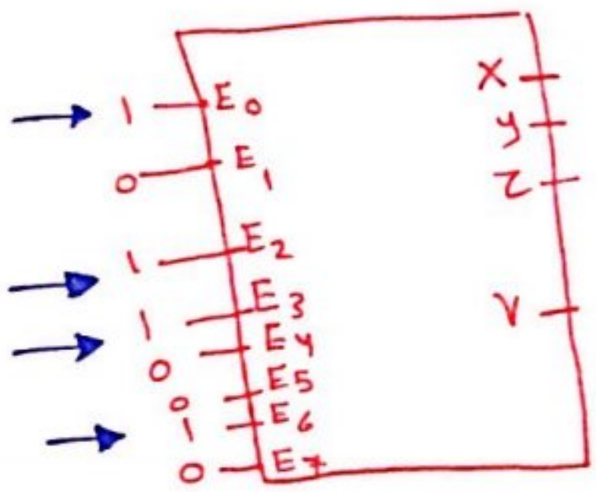
X y z

1 1 1

7

إذا كان أكبر فيه V = 1

V = 1 العادي



AH V

one

٧ أكبر

٤ ٦ ٩

x y z

١ ١ ٠

٣ < ٤

V = 1

$\bar{V} = 0$

لـ decode على الأقل.

AH oring(Σ)

AL Nand(Π)

* تجنب الأخطاء بـ Kmap تأكد من Row (كامل) Coloum

فمثلاً

$\bar{A}\bar{B}CD + A\bar{B}\bar{C}D$

0011 1101

اكتبي بهيك لتكتب الرقم نفسه.

بـ أسئلة أد (Anaylze) إذا المسألة معقدة بوزع أسامي بحيث بجعل العلاقة من خلال بوابة كل اسم وبعينه (T.T)

بـ (Analyze) سألك عن عدد الـ (using function)

عن طريق قانون 2^{2^n} عدد المتغيرات (x, y, z)

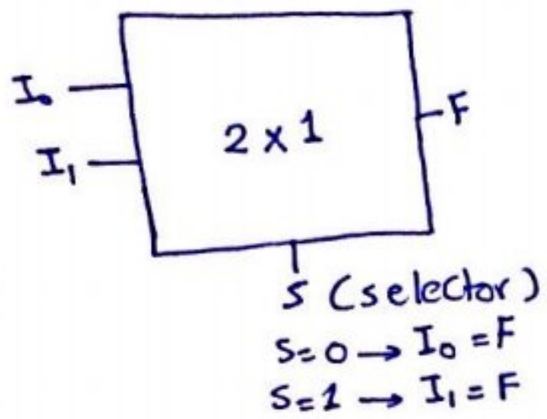
$2^{2^3} = 2^8 = 256$

combinational circuits the select one of the inputs.

2x1 , 4x1 , 8x1 , 16x1

input ← → output

(f) ← output واحد اختيار

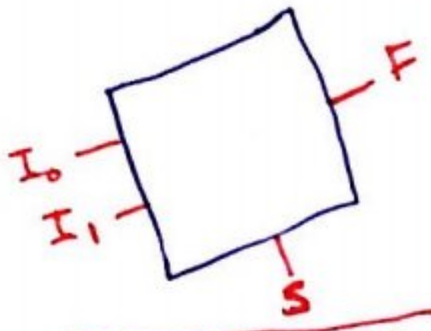


$S=0 \rightarrow I_0 = F$

$S=1 \rightarrow I_1 = F$

2input (1) select
4input (2) select
8 input (3) select
16 ---(4) select

S	f
0	I_0
1	I_1



$F = I_0 \cdot \bar{S} + I_1 S$

$F|_{S=1} = I_0 \bar{1} + I_1 1$

$= I_0 \cdot 0 + I_1$

$= I_1$

$F|_{S=0} = I_0 \bar{0} + I_1 0$

$= I_0 \cdot 1 + 0$

$= I_0$

S	I_0	I_1	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

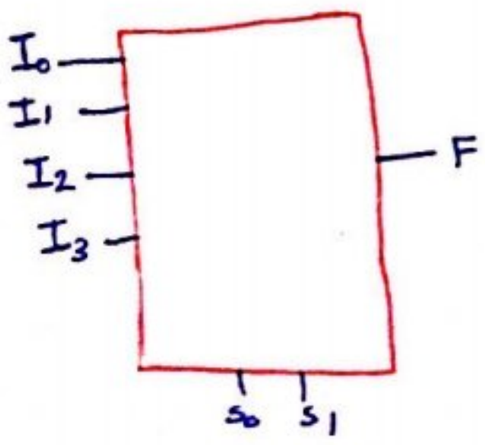
I_0I_1 / S	00	01	11	10
0	0	0	1	1
1	0	1	1	0

$f = I_0$

$F = I_1$

$f = \bar{S} I_0 + S I_1$

* Design 4x1 Mux ?!

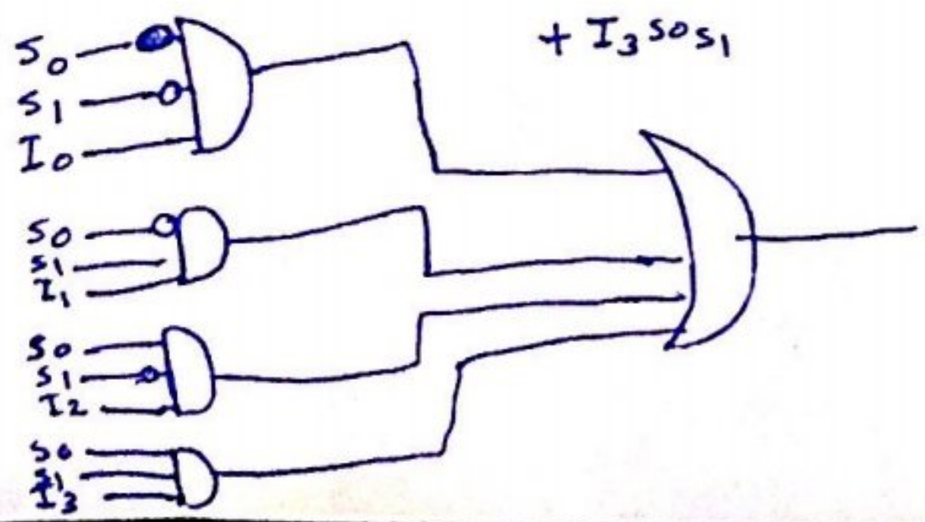


S_0	S_1	F
m_0 0	0	I_0
m_1 0	1	I_1
m_2 1	0	I_2
m_3 1	1	I_3

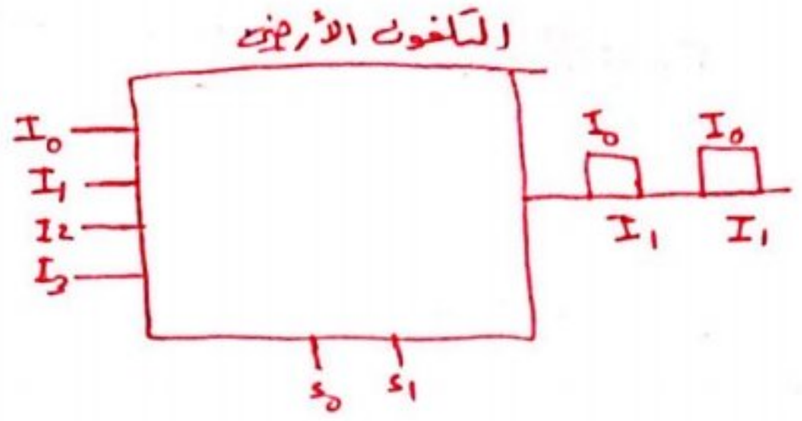
$F = I_0 m_0 + I_1 m_1 + I_2 m_2 + I_3 m_3$

$= I_0 (\bar{S_0} \bar{S_1}) + I_1 (\bar{S_0} \bar{S_1}) + I_2 S_0 \bar{S_1}$

$+ I_3 S_0 S_1$

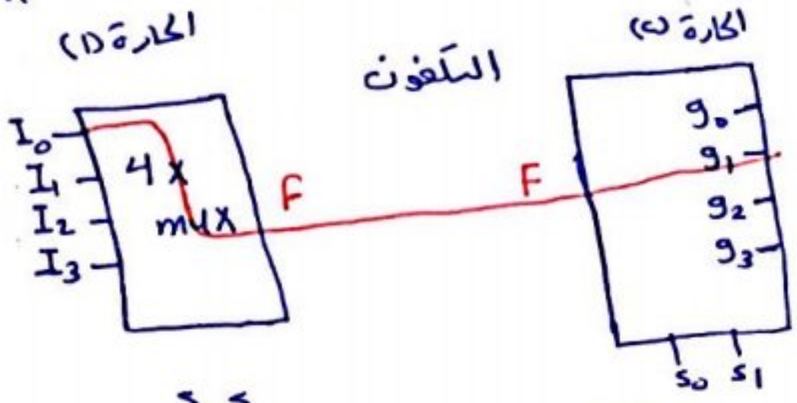


التلفون الأرضي



* DeMUX :- mux عكس

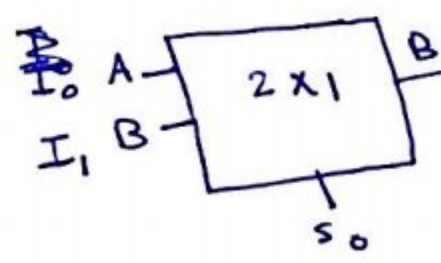
الكارة (1) التلفون الكارة (2)



S_0 S_1	
00 → 0	$g_0 = \bar{S_1} \bar{S_0} f$
01 → 1	$g_1 = \bar{S_0} S_1 f$
10 → 2	$g_2 = S_0 \bar{S_1} f$
11 → 3	$g_3 = S_1 S_0 f$

(A-B) sub-Adde ÷

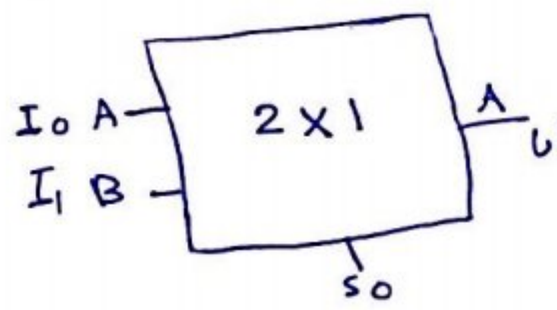
(B-A) لكن طابقد اعمل



نخلي قيمة $(I_1) B = S_1$ وشان

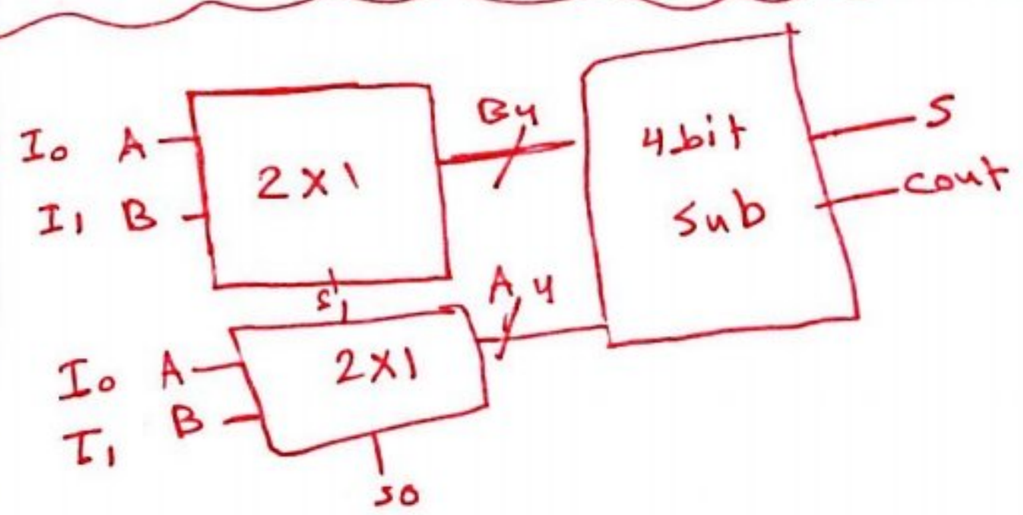
اعمل B-A واختار B

* بقتد الحل لأي رقم بس صابدي باستخدام mux



نخلي قيمة $(I_0) A = S_0$ وشان

اختار A

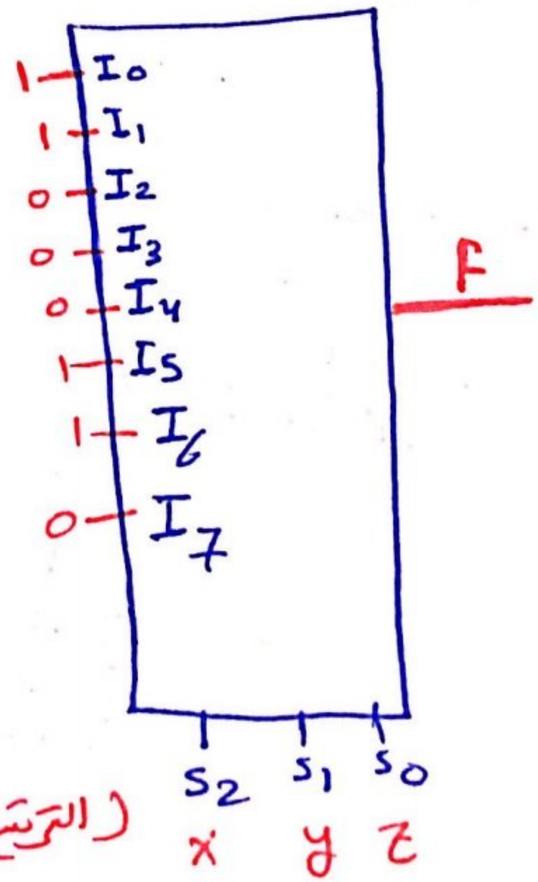


10

$F = \Sigma(0,1,5,6)$ implement this function using 8X1 mux

أفضل اشي نحل T.T بعدين نبزبر الرسمة

	X	y	z	F
m_0	0	0	0	1
m_1	0	0	1	1
m_2	0	1	0	0
m_3	0	1	1	0
m_4	1	0	0	0
m_5	1	0	1	1
m_6	1	1	0	1
m_7	1	1	1	0



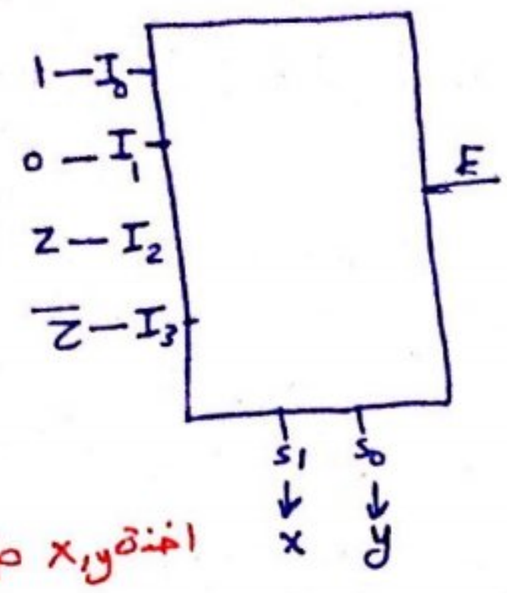
(الترتيب أفضل شي)

4X1 mux

بدنا نعمل 8 لـ 4 ÷ 2

كل صفين بعتبرهم (0)

X	y	z	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0



اخدنا x,y ضل z بدي أوجد علاقة F(z)

* بعدين بروحلهم حسب النوع بلتزم فيه وكمان لازم (S)

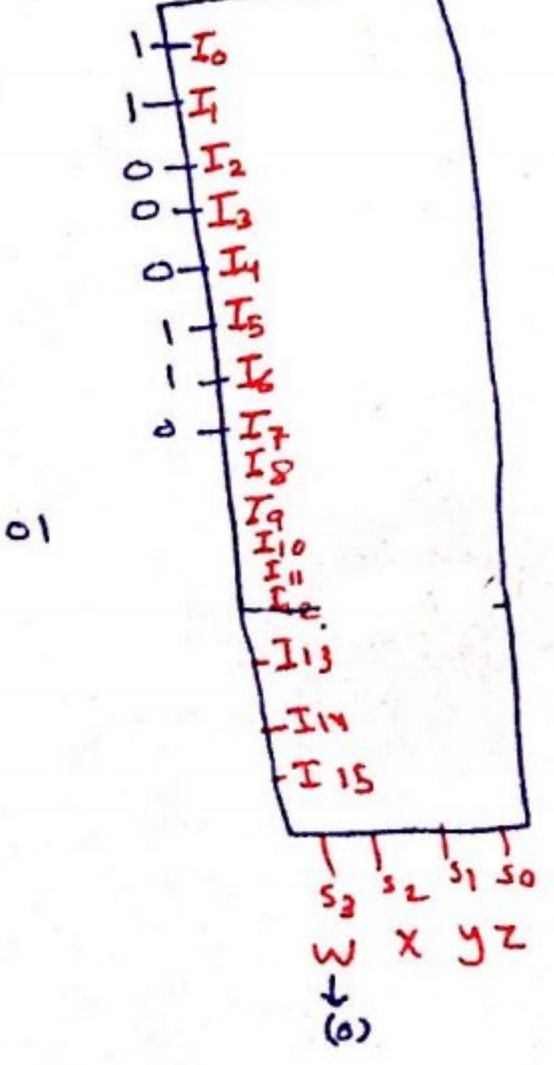
S2 S1 S0
x y z

داخل الترس

8 الي 2 بقسم كل (4)

* 2X1 mux

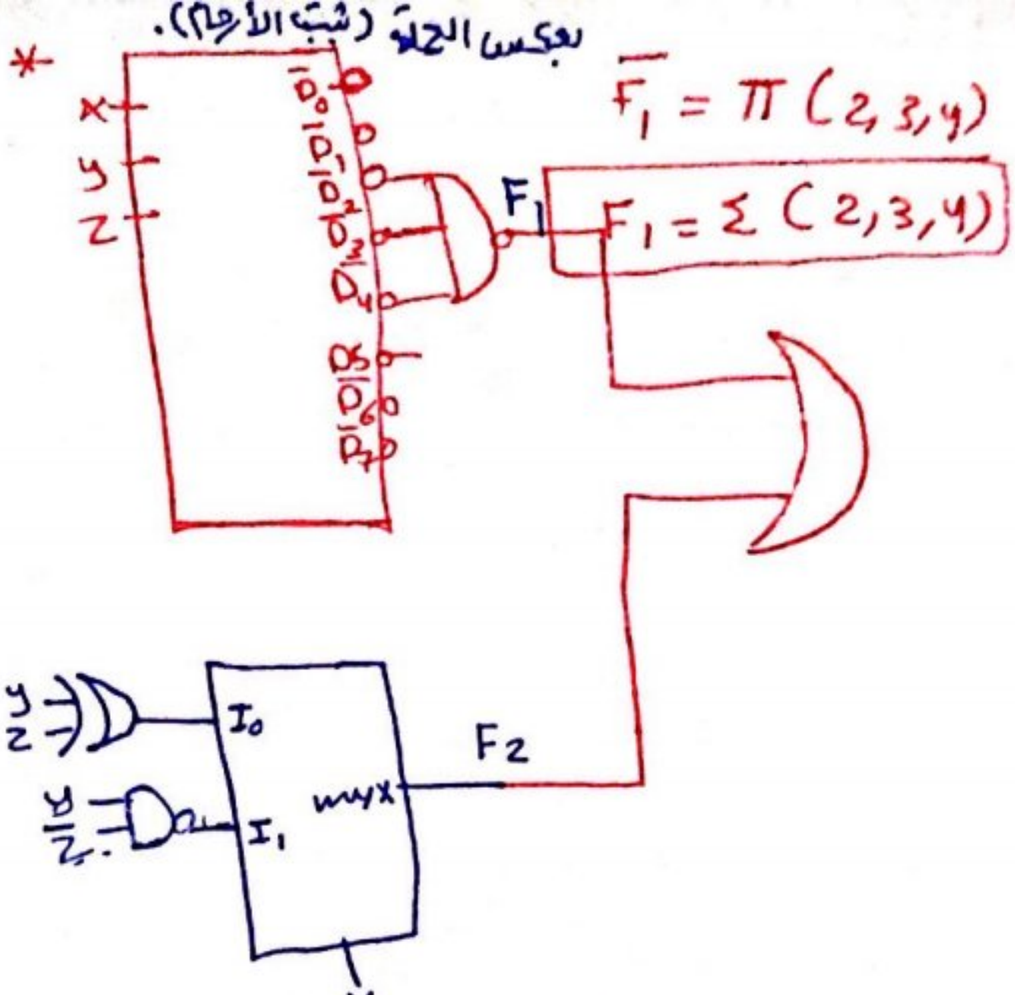
X	y	z	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0



* 16X1 MUX

x	y	z	F
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

1001 اخر زيادة فزلة بخط (0) حالياً شر



X	F_2
0	I_0
1	I_1

F_2 = في الربيعة I_0, I_1

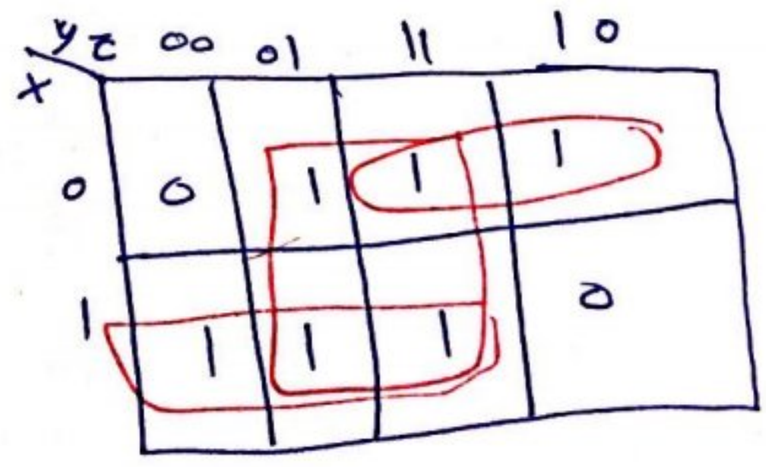
* لما اجي اعمل T.T كل شي حاطط جوا أقطيه حاطي قيم S حاطي قيم I

Find F_3 ?!

simplified sop

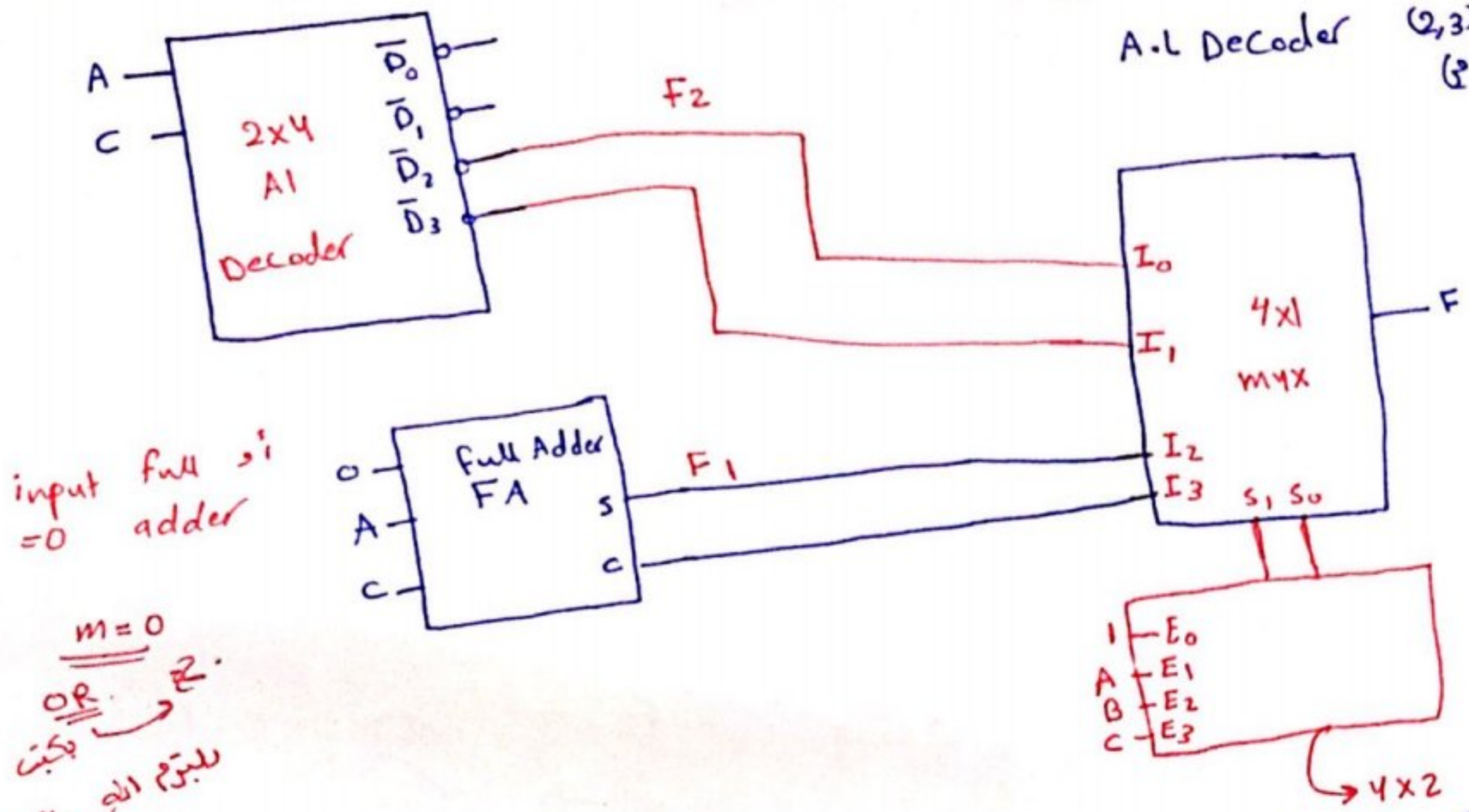
$f_3 = f_1 + f_2$

	X	y	z	F_1	F_2	F_3
I_0	0	0	0	0	0	0
	0	0	1	0	1	1
	0	1	0	1	1	1
	0	1	1	1	0	1
I_1	1	0	0	1	1	1
	1	0	1	0	1	1
	1	1	0	0	1	0
	1	1	1	0	0	1



$f_3 = (\bar{x}y).(z) + (x\bar{y})$

$F = y\bar{z} + \bar{x}z + \bar{x}y$



A.L Decoder (2,3) الشكل الباقي طامي

input full adder ؛ =0

m=0

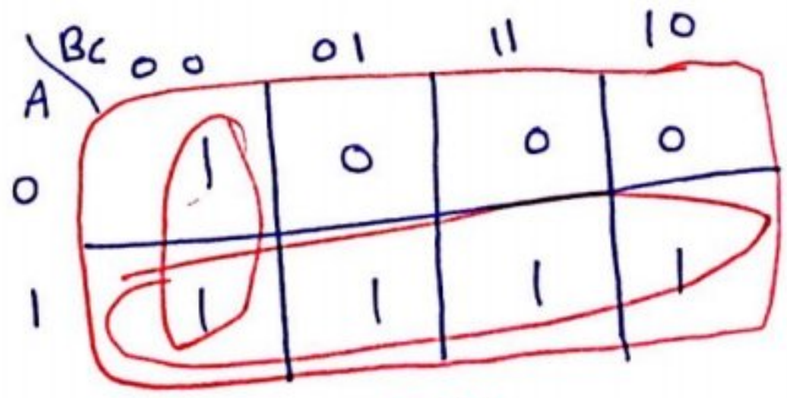
OR ← Z. بكتب للبرنامج الله full adder

Priority (أولوية) encoder

باخذ قيمة A وباخذ الأولوية الأكبر B,C

E_0, E_1, E_2, E_3

A	B	C	F_1 C	F_1 S	F_2	F_2 S_1	F_2 S_0	F (S) على شو بيشتغل
0	0	0	0	0	0	0	0	I_0 (1)
0	0	1	0	1	0	1	1	I_3 (0)
0	1	0	0	0	1	1	0	I_2 (0)
0	1	1	0	1	1	1	1 (C) أكبر	I_3 (0)
1	0	0	0	1	0	0	1 (A) اكبر	I_1 (1)
1	0	1	1	0	0	1	1 (C) اكبر	I_3 (1)
1	1	0	0	1	0	1	0 (B) أكبر	I_2 (1)
1	1	1	1	0	0	1	1 (C) اكبر	I_3 (1)



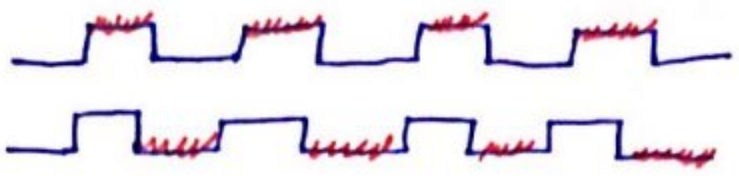
$F = (A) + (\bar{B}\,\bar{C})$

lec (10) "memory element"

[1] latch → SR, D (asynchrouns)

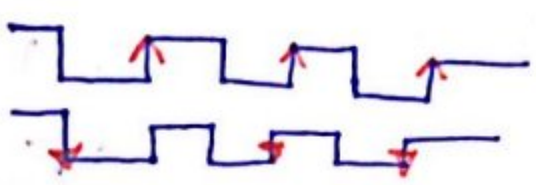
[2] Flip flop → D, JK, T (isynchrouns)

* latch high level sensetive
* latch low level sensetive



(على فترة كاملة التغير)

* Flip flop +ve edge trigged (0→1)
* flip flop -ve edge trigged (0→1)



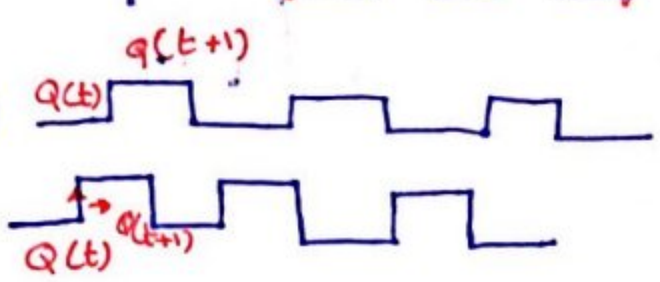
(لحظي التغير)

التيار عند الزمن الحالي .

Q(t) : current state (0 or 1)

Q(t+1) : future / next state

التيار المستقبلي



* Active high SR latch

S: ON / set / 1 / high

R: off / reset / 0 / low

	S	R	Q(t)	$\bar{Q}(t)$
set	1	0	1	0
No change	0	0	1	0
Reset	0	1	0	1
Nochange	0	0	0	1
forbidden	1	1	0	0
forbidden				

بتخيل انه بتجاهل مع حراوتر

(حرفياً)

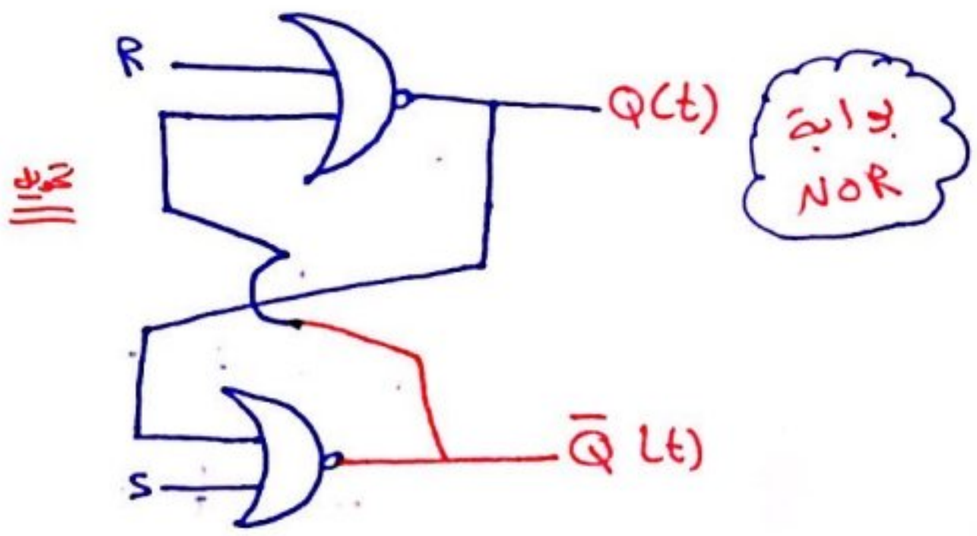
الراوتر .

التفاصيل / أنواع البوابة / الرسمة

R → Q(t)
S → $\bar{Q}$(t)

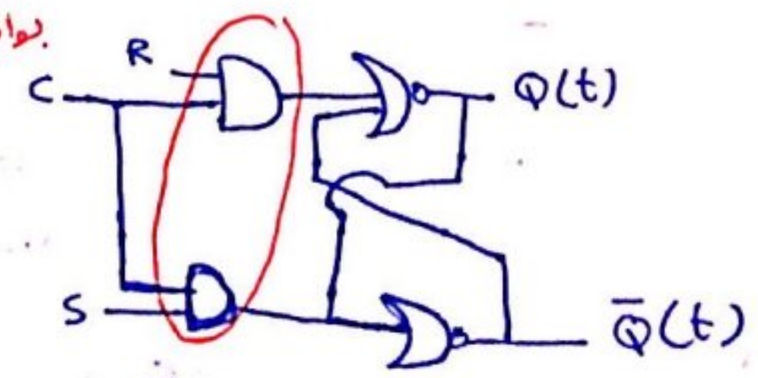
التيار الحالي

S	R	Q(t)	$\bar{Q}$(t)	Q(t+1) (future)	$\bar{Q}$(t+1)
1	0	0	1	1	0
0	0	1	0	1	0
0	1	1	0	0	1
1	1	0	1	0	0

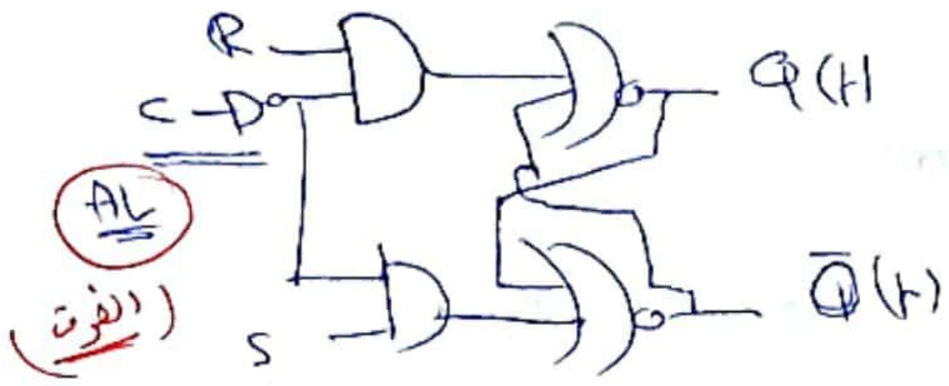


تحويل

C (clock)	S	R	Q(t) $\bar{Q}$(t)
1	0	0	Q(t) $\bar{Q}$(t) (No Change)
1	1	0	(set)
1	0	1	Reset
1	1	1	for bidden
0	x	x	No Change (التحكم بالبوابة)



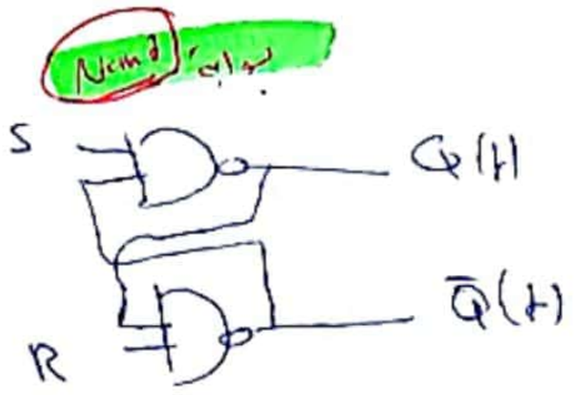
(clock) AL	S R	$\bar{Q}(H)$ $\bar{Q}(H)$
0	0 0	No chng
0	1 0	set
0	0 1	reset
0	1 1	forbidde
1	X X	No chng



12

* Active low SR latch

S R	Q(H $\bar{Q}(H)$
0 1	set
1 1	No change
1 0	reset
0 0	forbidden (11) ← AL → AH (00)



* D-latch

Clock (AH)	D	Q(H)
1	Reset(0)	0
1	Set(1)	1
0	X	No change (حفظ البيانات السابقة)

D-latch

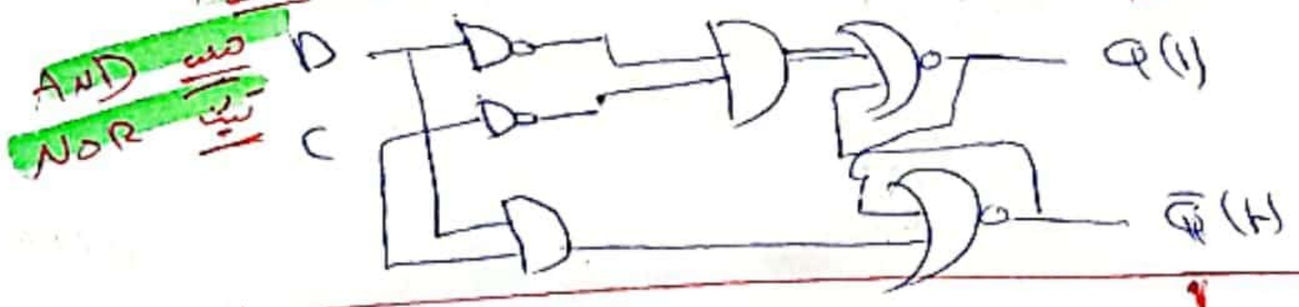
transparent (مشفرات)

اذا دخل (0) ← يطلع (0)

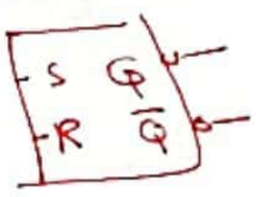
اذا دخل (1) ← يطلع (1)

* كلما (AH) او (AL) تكون لـ Clock مول (D) الـ (D) شفرة

تبنى بوابة NOR AND من



Ex $Q=0$ AH

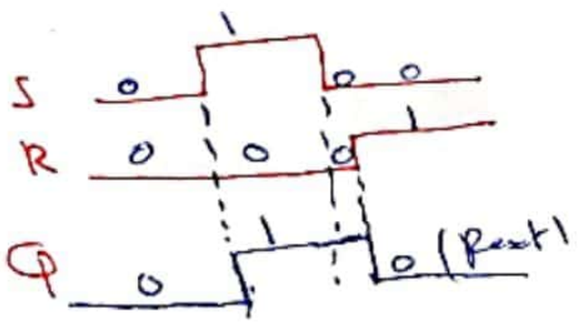


AH SR

AH = bubble واحنة R,S كل

حاضر عندي (clock)

لو طلب $\bar{Q}$ نقلب رجعه Q

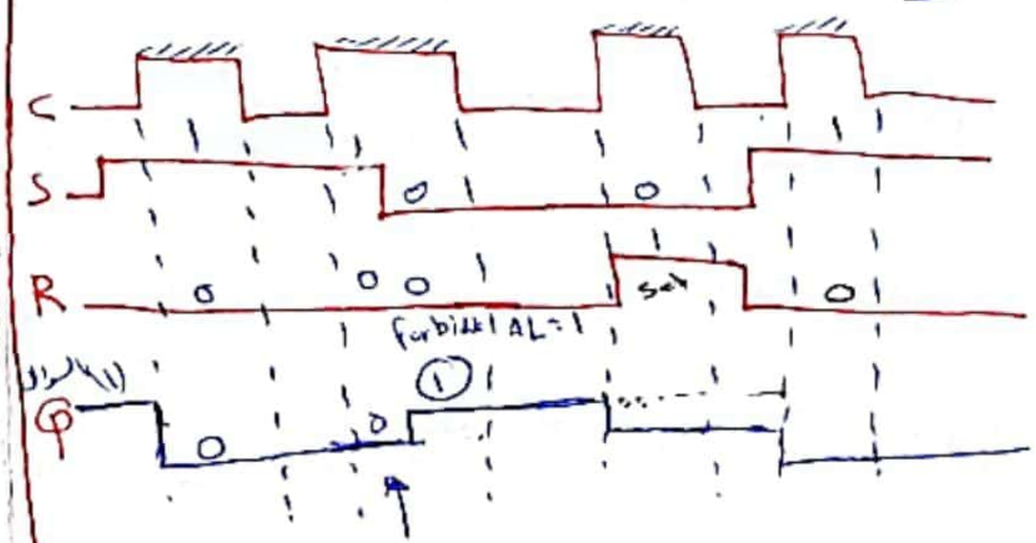


$Q=1$

AL SR

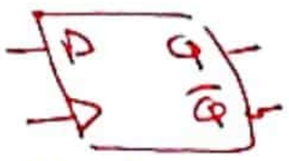
AH ⊆ (شغلنا بالنواظر العادية)

شوف كلش في SR



لو طلب $\bar{Q}$ نقلب (Q)

* D. flip flop



(flip) رجوعاً للكلمة لأصل تسمية ال

-Ve negative bar مع / +Ve Positive bar بدون

D	Q(t+1)
0	0
1	1

(+Ve) D — (-Ve) تقرأ

للأبطأ بقطع

(D latch) بسمى

بحالة ما اطلق اسم (أيهما أحدث)

D latch — D flip flop

D	Q(t)	Q(t+1)
0	0	0
0	1	0
1	0	1
1	1	1

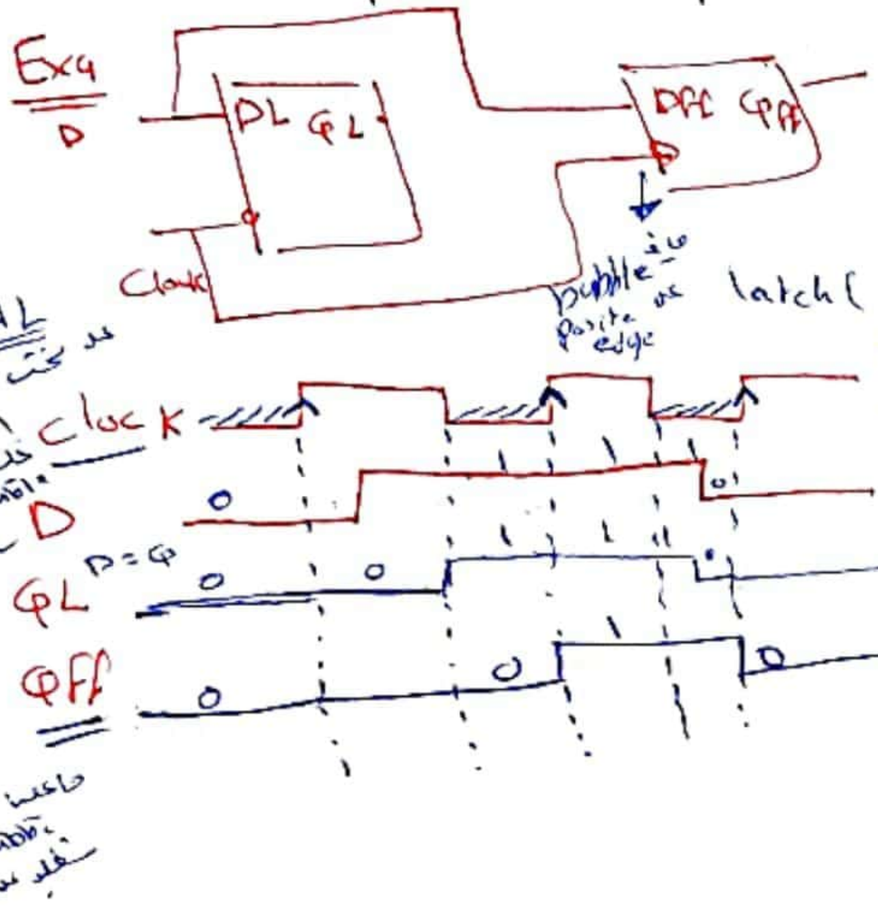
D (+Ve / -Ve)

لنفس التغير Q(t) فالنتيجة معي

$D = Q(t+1)$

صفرا

Ex
D



* قفالج عند Clock تحت D

AH
AL

* JK Flip Flop

J	K	Q(t+1)	$\overline{Q}(t+1)$
0	0	No change	
0	1	Reset	
1	0	set	
1	1	Toggle	

(SR) مثل

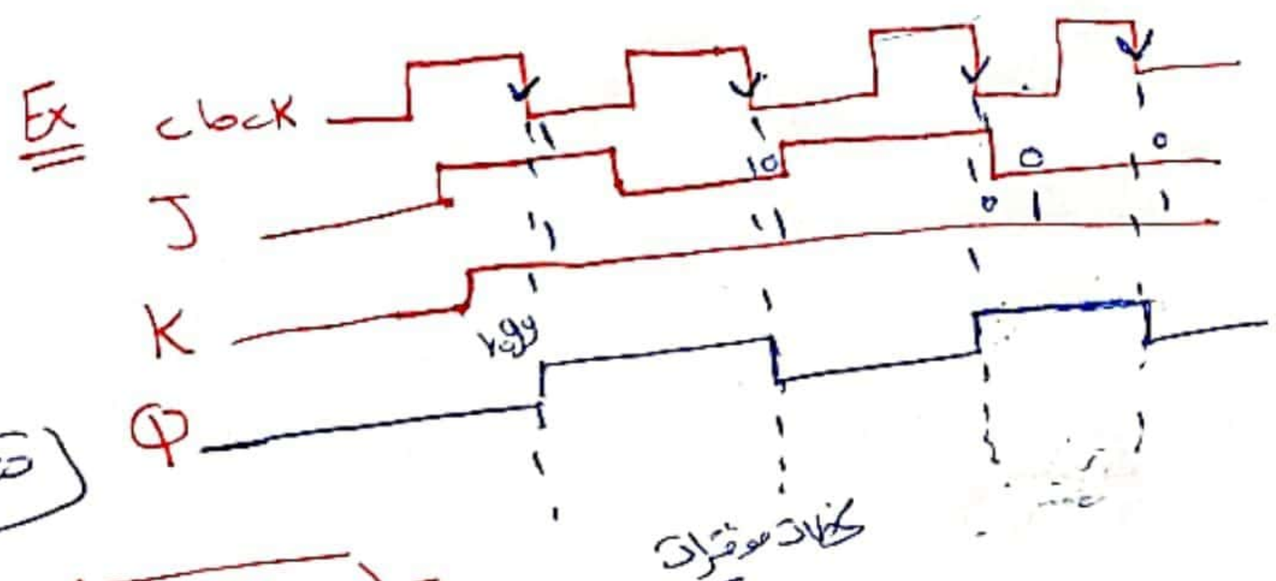
(1→0) (0→1)

* (بتعكس) الوحيدة اللي بتختلف

J	K	Q(t)	Q(t+1)
0	0	0	Nochng (0)
0	0	1	// (1)
0	1	0	Reset 0
0	1	1	Reset 0
1	0	0	set 1
1	0	1	1
1	1	0	Toggle 0
1	1	1	

لنفس التغير Q

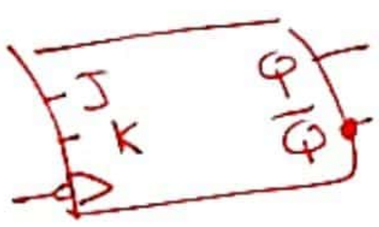
Ex



J \ K	Q(t) 00	01	11	10
0	0	1	0	0
1	1	1	0	1

$Q(t+1) = \overline{K}Q + J\overline{Q}$

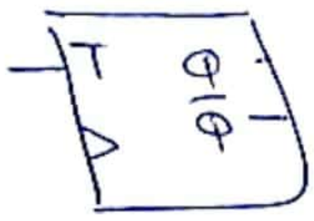
(Q=0)



-Ve bubble

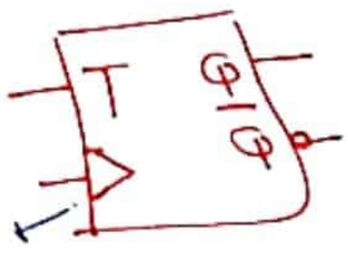
* T. flip flop

T	Q(t+1)
0	Nochng
1	toggel (عكس)

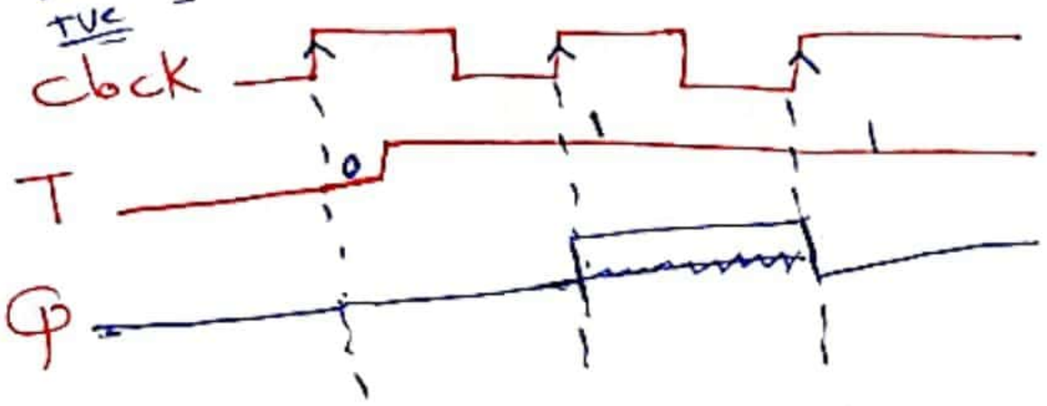


T	Q(t)	Q(t+1)
0	0	Nochng 0
0	1	1
1	0	toggel 1
1	1	0

Ex)



bubble يدل على TVE



T \ Q(t)	0	1
0	0	1
1	1	0

$Q(t+1) = T \oplus Q(t)$

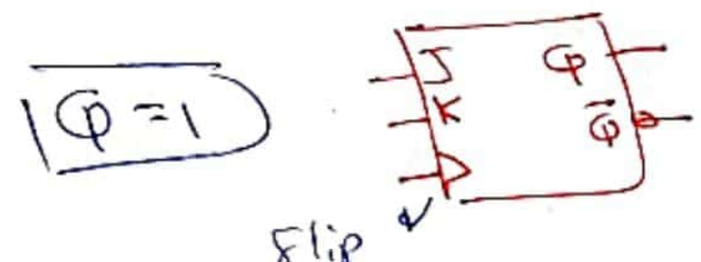
Direct inputs صمم آخر

1) Direct reset → [clear] → دائماً صفر (Zero) (حالاً تتغير) clear = 1 صفر — إذا اشتغل الـ clear دائماً بيخلي البيان يصير صفر بأي وقت (حالاً تتغير) (بالغصب عن الـ clock) ما عندهم صفة تعمل متزامن

2) Direct set → [preset] → دائماً واحد one

صمم آخر

(latch / flip)



Q = 1

flip

bubble حاجبا

+ve :

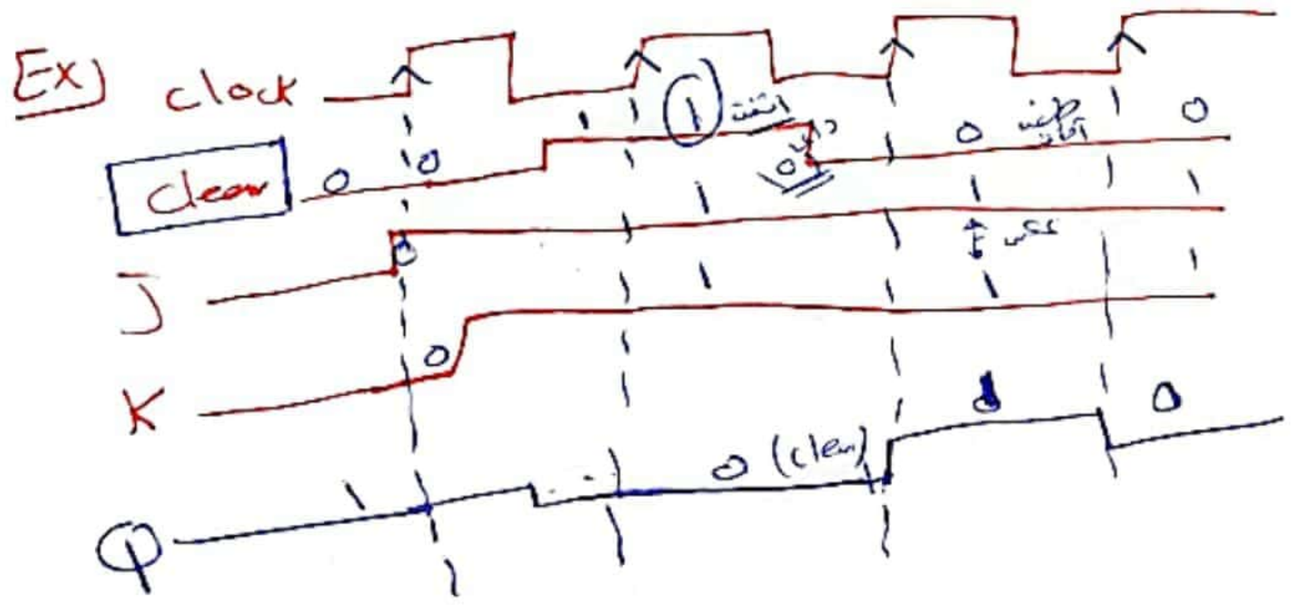
Ex) clock

clear

J

K

Q



Q = 1

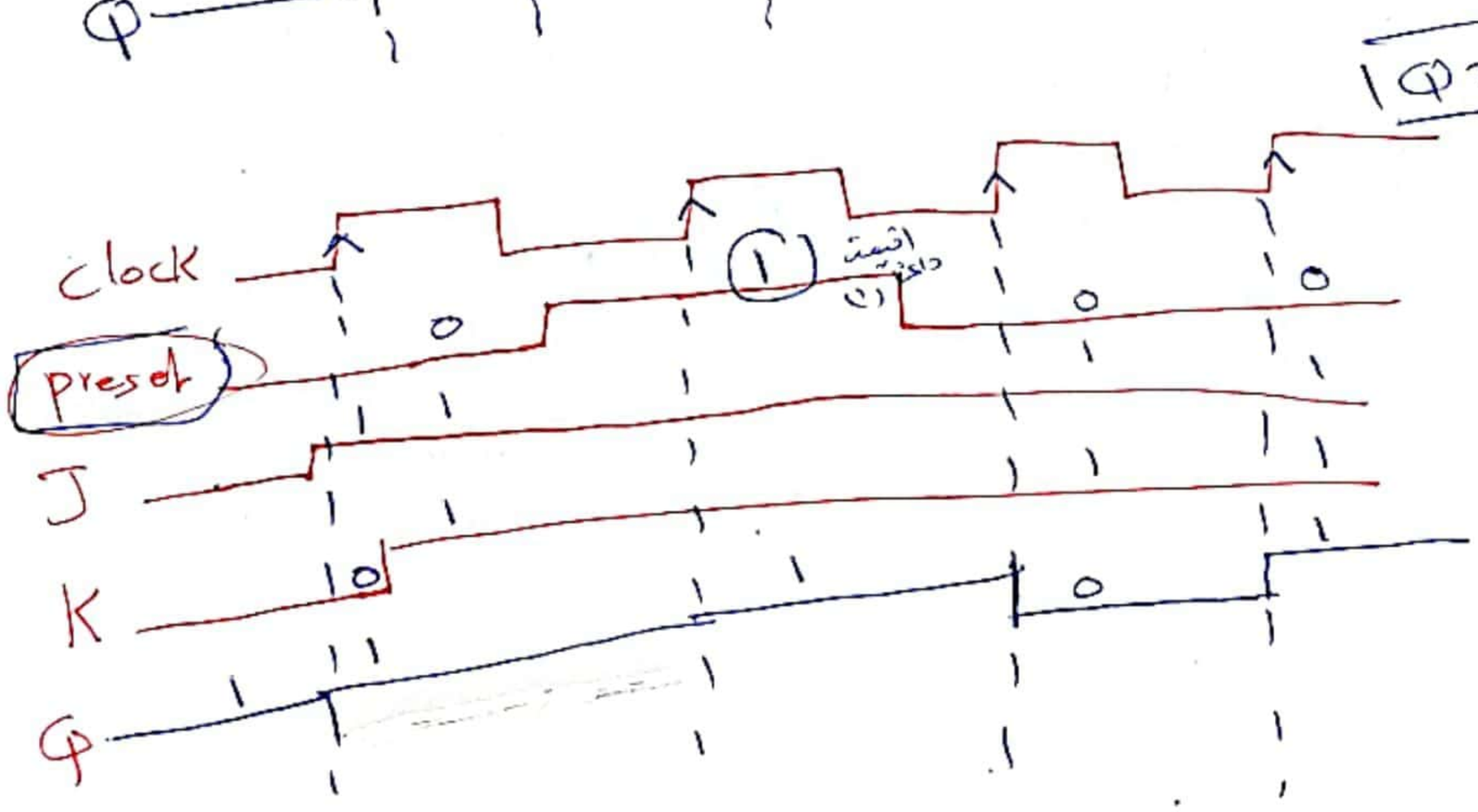
clock

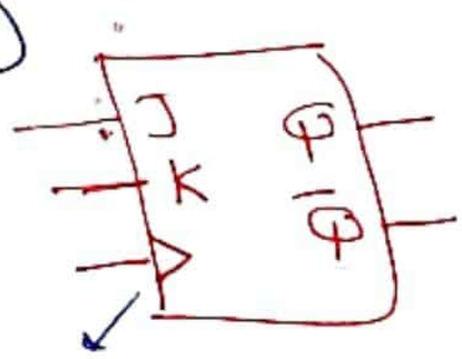
preset

J

K

Q





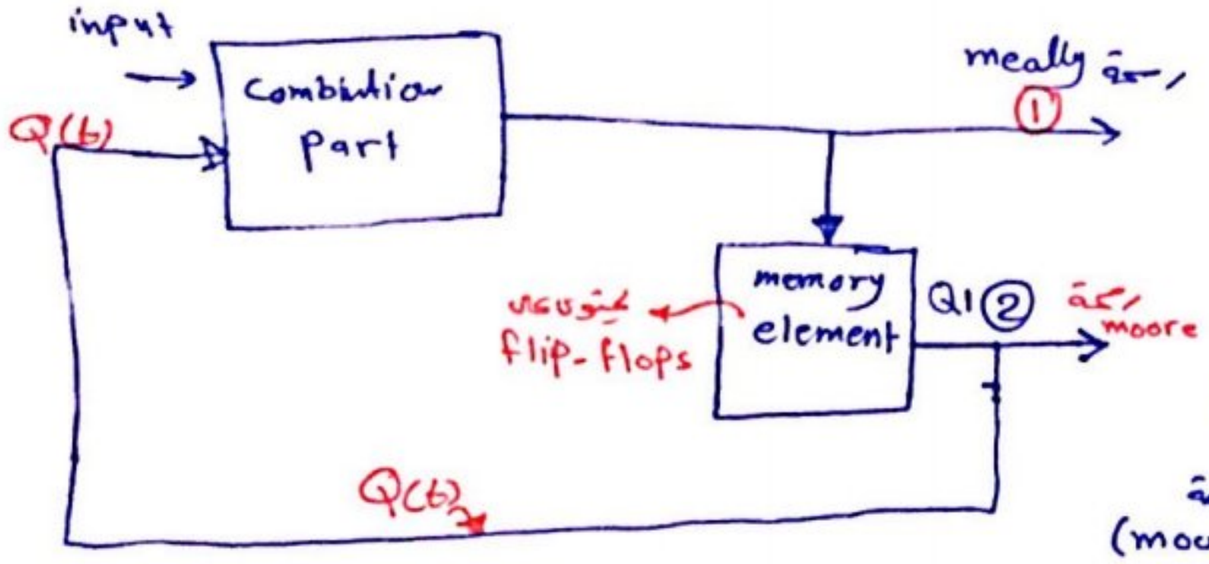
f +ve No bubble

lect 11 (Analysis of Sequential circuits)

Given : circuit, Required : - state/transition table

Input [Q(t)] → output [Q(t+1)]

* sequential circuit = combinational logic + memory element



① Meally circut : input الناتج معتمد على Q و

→ Q(t+1) ⟸ Q(t), input

→ output ⟸ Q(t), input

② moore circut : الناتج معتمد على Q فقط

→ Q(t+1) ⟸ Q(t), input

→ output ⟸ Q(t)

* أي (state) تتبع السيركت تعتمد على memory

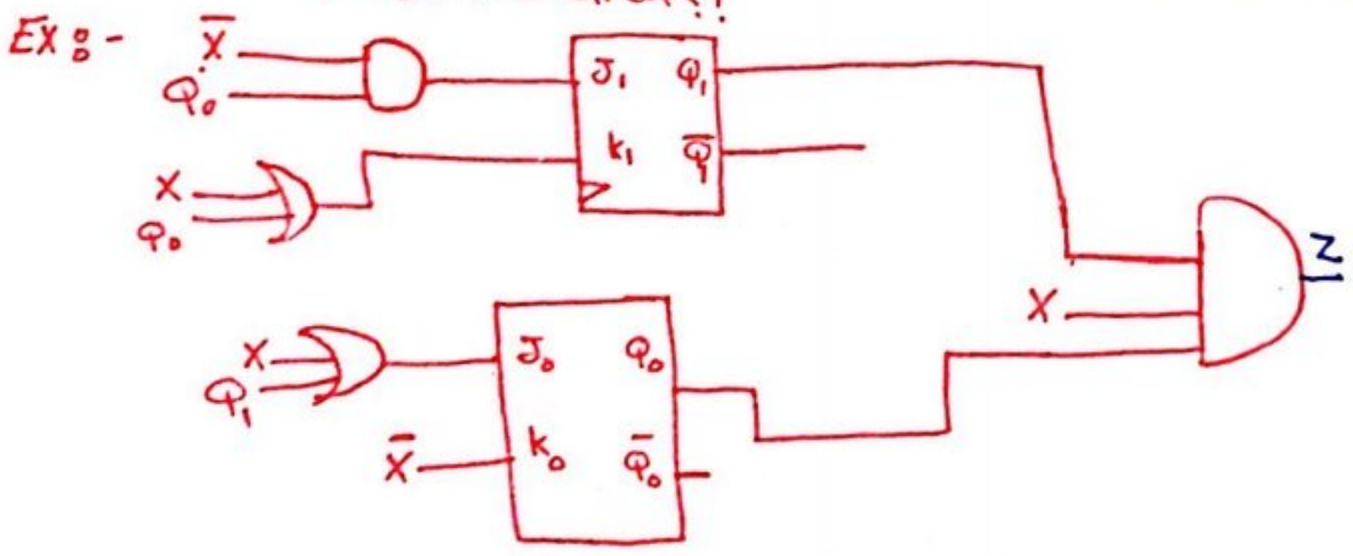
max # of state = 2^m → # of f.f

* Analysis steps :-

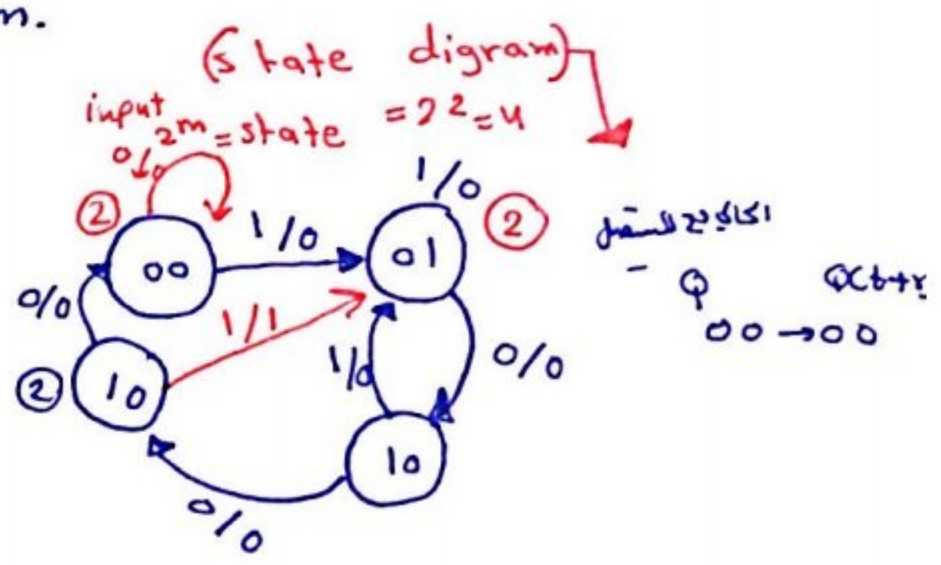
1. find boolean expression of the f.f input and output.
2. find the value of the f.f [T.T]
3. find the next state (Q(t+1) ((state بتسمى) digram.

Analysis the circut?!

EX :-



(state digram)



الكاونتر التسلسلي

Q | Q(t+1)

00 → 00

1. $J_1 = \bar{X} \cdot Q_0$, $K_1 = X + Q_0$; $J_0 = X + Q_1$, $K_0 = \bar{X}$; $Z = Q_1 \cdot X \cdot Q_0$ (Z: out, input)

(X) input عدد ال

2^n = عدد الحالات

$2^1 = 2$ # of out going

2. T.T

Q_1	Q_0	X	J_1K_1	J_0K_0	$Q_1(t+1)$	$Q_0(t+1)$	Z
0	0	0	0 0	0 1	0	0	0
0	0	1	0 1	1 0	0	1	0
0	1	0	1 1	0 1	1	0	0
0	1	1	0 1	1 0	0	1	0
1	0	0	0 0	1 1	1	1	0
1	0	1	0 1	1 0	0	1	0
1	1	0	1 1	1 1	0	0	0
1	1	1	0 1	1 0	0	1	1

lecture" Design of sequential circuit

[1] D-f.f

Q(t)	Q(t+1)	D
0	0	0
0	1	1
1	0	0
1	1	1

D = Q(t+1)
نفس المخرج اللي

[2] T.f.f

Q(t)	Q(t+1)	T	
0	0 (يضل)	0	no change
0	1 (عكس)	1	toggle
1	0 (عكس)	1	toggle
1	1 (يضل)	0	no chang

[3] Jk f.f

Q(t)	Q(t+1)	JK
0	0 no/re	0X
0	1 togg/set	1X
1	0 togg/re	X1
1	1 no/set	X0

00 01 11 10

* خطوات ال (design):

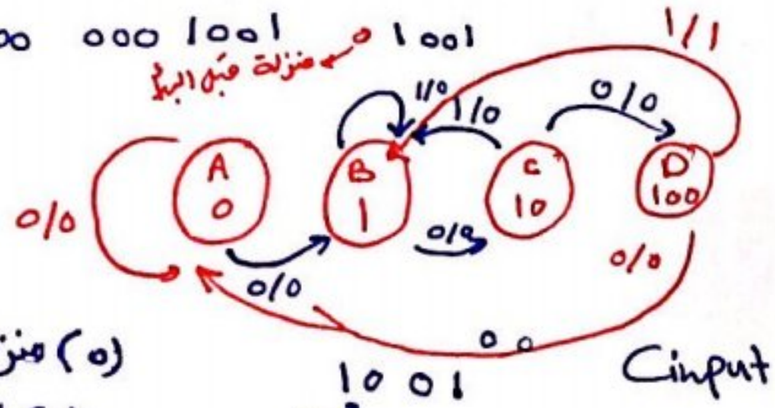
[1] state digram
[2] Assign Binary code for each state
[3] find f.f output value
[4] ,,, bolean expression
[5] Draw.

Ex: Design a circut that detect if x 1-bit input has the pattern "1001" using Jk

00 110 100 1001
00 000 000 1001

state 4 بس اذا 4bit

A: 00
B: 01
C: 10
D: 11



A: منزلة قبل البدء (0)
B: المنزلة الثانية (1)
C: المنزلة الثالثة (10)
D: 100

Cinput) التنكس
السيل
كه فاحد حالتين يا 0 يا 1

2^m = state عدد

$2^m = 4$ m=2

Q_1	Q_0	X	Q(t+1)	$Q_0(t+1)$	$J_1 k_1$	$J_0 k_0$	Z
A 0	0	0	A 0	0	0x	0x	0
A 0	0	1	B 0	1	0x	1x	0
B 0	1	0	C 1	0	1x	x1	0
B 0	1	1	B 0	1	0x	x0	0
1	0	0	1	1	x0	1x	0
1	0	1	1	1	x1	1x	0
1	1	0	0	0	x1	x1	0
1	1	1	0	1	x1	x0	1

f.f [2]

J_1

Q_1 \ Q_0x	00	01	11	10
0	0	0	0	1
1	0	x	x	x

0	0	0	0	0
1	0	0	1	0

J_0

0	1	x	x	
1	1	x	x	

K_0

x	x	0	1
x	x	0	1

K_1

	00	01	11	10
0	x	x	x	x
1	0	1	1	1

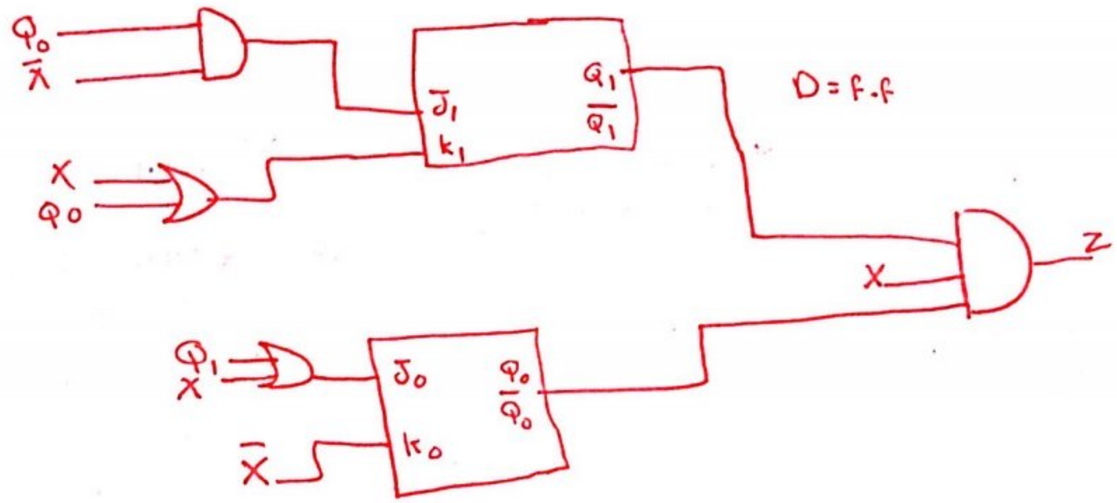
$J_1 = Q_0 \bar{X}$

$K_1 = X + Q_0$

$J_0 = Q_1 + X$

$K_0 = \bar{X}$

$Z = X . Q_0 Q_1$



*Design circut that detects 3 or more consequentions

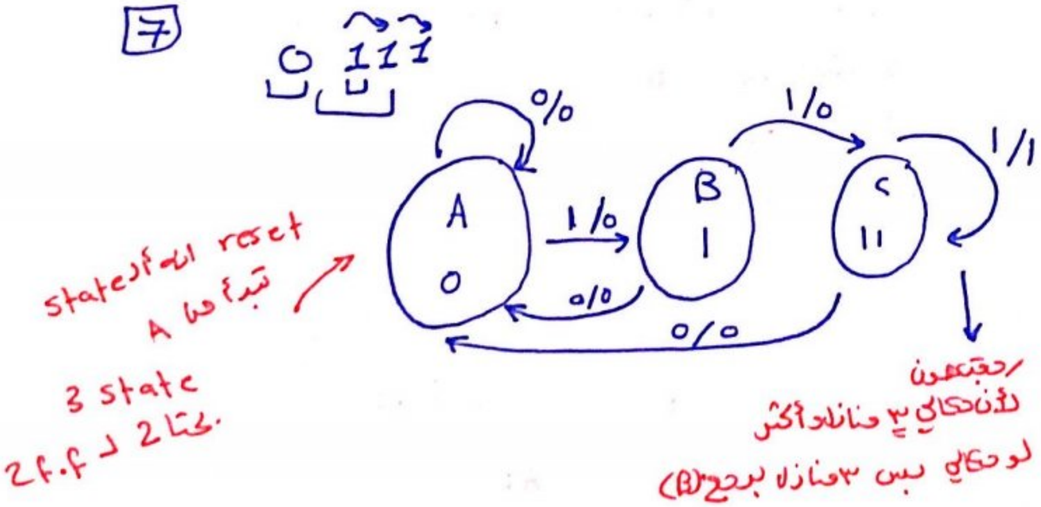
رقم تتكون من متتالية أو أكثر
كلهم عبارة عن (1)

لو كان طالب مع منازله (0)

000

1

منزلة قبل البدأ عكسه القبل.



lect 13 "Counters" (use Moore Fsm)

* Counters are **Synchronous**

- Regular (step width=1) خطوة واحدة
  - up (simply) $0 \rightarrow 2^m - 1$ ← عدد أرجل bit ← $\max (2^m - 1)$ عدد f.f
  - down (simply) $2^m - 1 \rightarrow 0$
- Irregular (next current / current +1 / current -1) يوجد simply بوجود شروط فقرة

- up / down
  - up → X=0
  - down → X=1

(شروط up / down) # of f.f = m

0 — 7

(o) * أول شيء تحقق الشرط بدأ من

* 7 تكتب 111 3 خانات

m=3

$2^3 - 1 = 7$

(regular) ∴

EX:- [1] 0,1,2,3,4,0,1,2,3,4,... 3 خانات

[2] 0,1,2,3,4,5,6,7,0,....

[3] 0,2,5,3,2,5,3,2,0

[4] mode-8 counter

0-7 0,n-1

state f.f كم عدد — عدد mealy

out عالم علاقة بالشرح (state)

moore عدد f.f كم عدد

max # اللي بدي أوصلها

f.f (2) mealy — 110 *

f.f (3) moore

(خالفنا أحد الشروط)

* بين كل رقم ورقم عدد أكثرهما (1) أو خطوات حتى

* عدهما Zero ← 8

$2^4 - 1 = 15$ ← 1000 (3 2 1)

. 0-15 المفروض regular و Irregular

✓ خطوة (Irregular)

$2^3 - 1 = 7$

✓ خطوة (1) [1] [2] (regular)

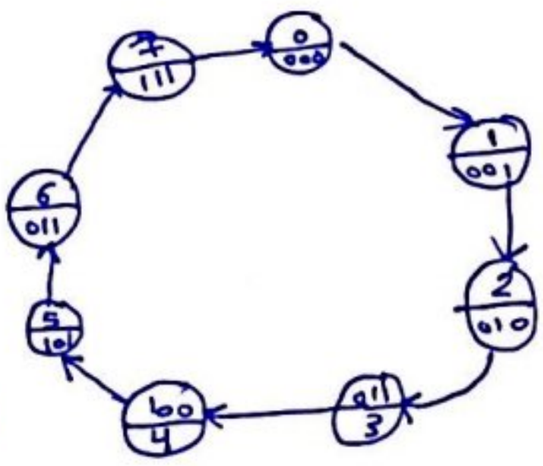
✓ 0,7

✗ مش خطوة [1] [x] (Irregular)

خطوة [1] [2] (regular)

✓ 0-7

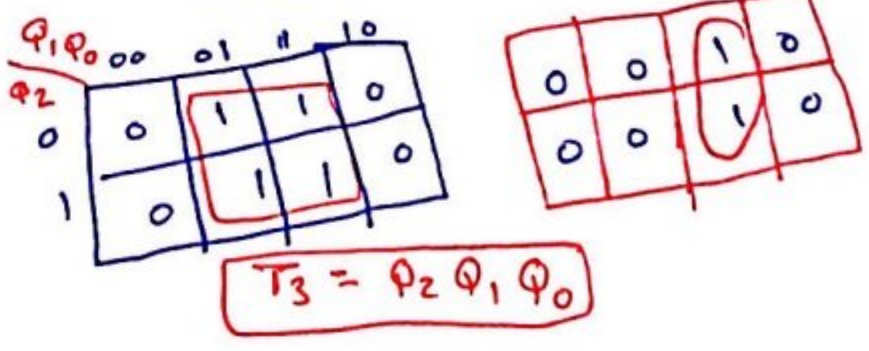
* Design Regular up counter (0-7) $Q_2 Q_1 Q_2$ (output)



up

$Q_2(t)$	$Q_1(t)$	$Q_0(t)$	$Q_2(t+1)$	$Q_1(t+1)$	$Q_0(t+1)$	T_2	T_1	T_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

$T_0 = 1$ $T_1 = Q_0$ $T_2 = Q_1 Q_0$



$T_3 = Q_2 Q_1 Q_0$

لو JK

$T_0 = 1$

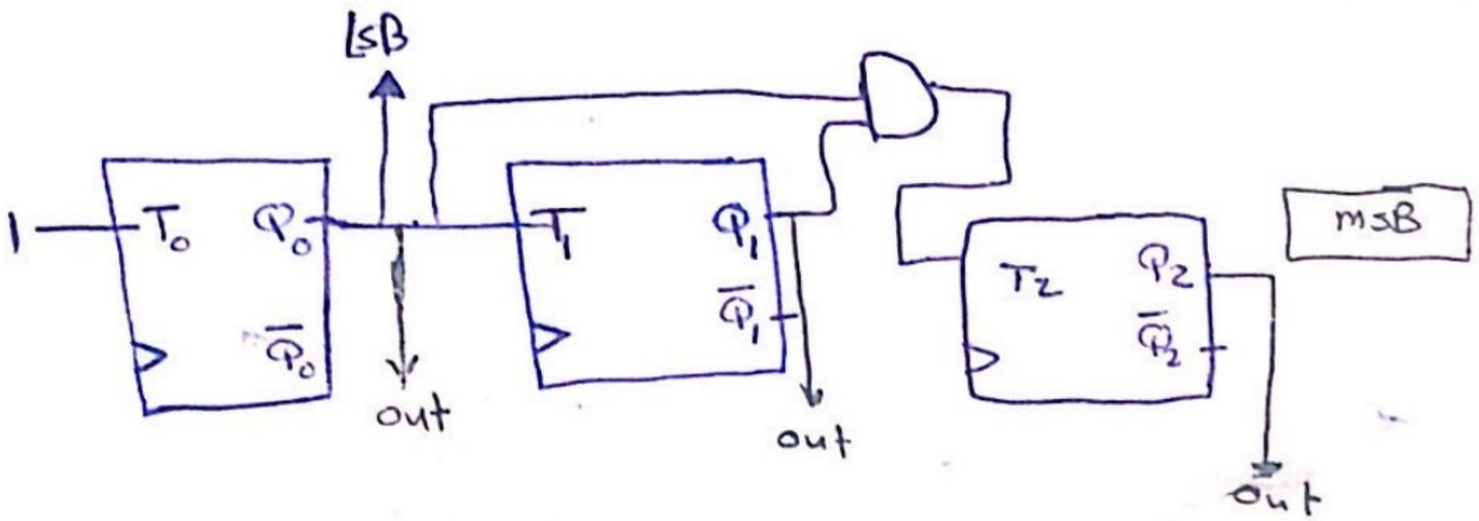
$J_0 k_0 = 1$

لـ

$J = k = T$

$J_1 J_1 = Q_0$

$J_2 k_2 = Q_2 Q_1$



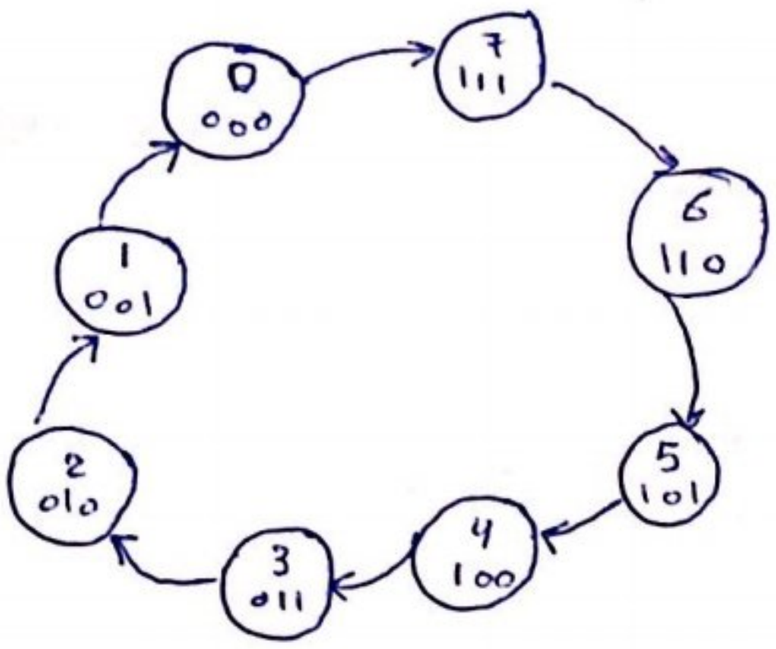
بـ T.T

لـ شطبنا الـ input (لأنه ماعندي خيارات للعد دائماً فـ أد state للي بجد منهون خيارات.

شطبنا أد output بدي out = Q

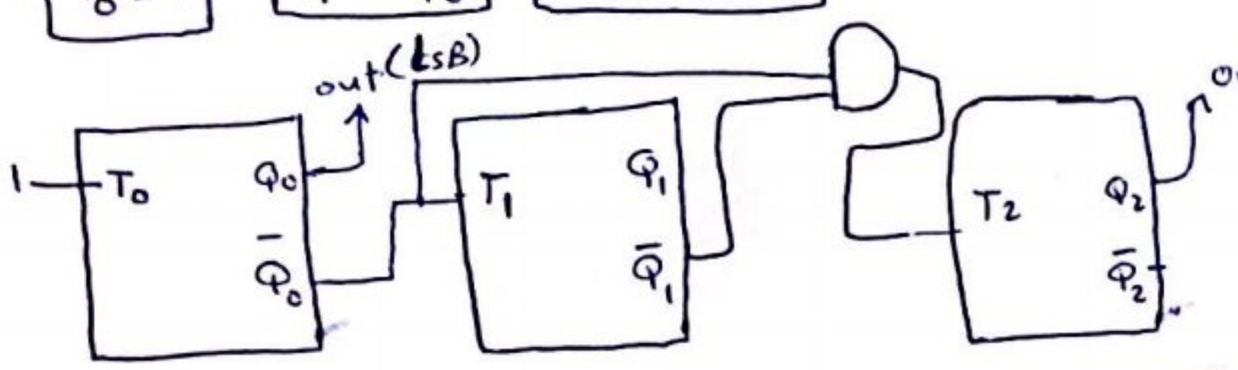
Current state.

Ex:- regular 7_0 down



Q_2	Q_1	Q_0	$Q_2(t+1)$	$Q_1(t+1)$	$Q_0(t+1)$	T_2	T_1	T_0
0	0	0	1	1	1	1	1	1
0	0	1	0	0	0	0	0	1
0	1	0	0	0	1	0	1	1
0	1	1	0	1	0	0	0	1
1	0	0	0	1	1	1	1	1
1	0	1	1	0	0	1	1	1
1	1	0	1	0	1	1	1	1
1	1	1	1	1	0	1	1	1

$T_0 = 1$ $\quad T_1 = \overline{Q_0}$ $\quad T_2 = \overline{Q_0}\,\overline{Q_1}$



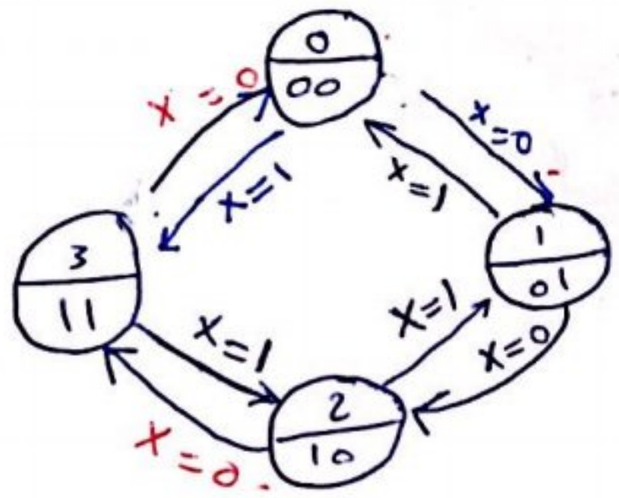
(bar) نضع الحلاقات لكن بوجود

(input) X أحطاوقيمة

* Design Counter regular up/down

0 - 3 → X = 0 (up)

3 - 0 → X = 1 (down)



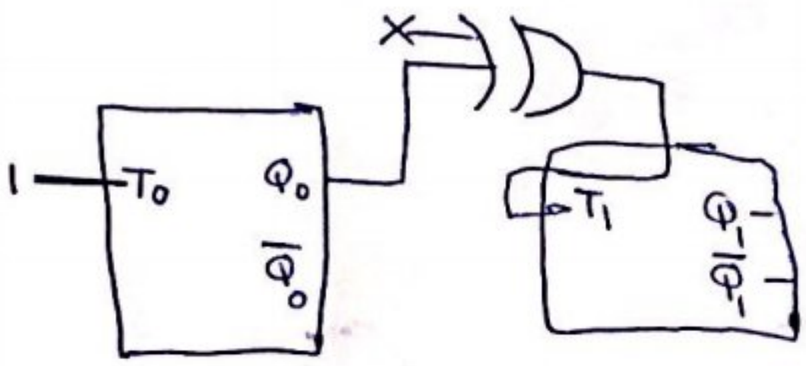
Q_1	Q_0	X	$Q_1(t+1)$	$Q_0(t+1)$	T_1	T_0	D_1	D_0
0	0	0	0	1	0	1	0	1
0	0	1	1	1	1	1	1	1
0	1	0	1	0	1	1	1	0
0	1	1	0	0	0	1	0	0
1	0	0	1	1	0	1	1	1
1	0	1	0	1	1	1	0	1
1	1	0	0	0	1	1	0	0
1	1	1	1	0	0	1	1	0

$T_0 = 1$ $\quad T_1 = \overline{Q_0}X + Q_0\overline{X}$

$T_1 = Q_0 \oplus X$

مستقابلة

0	1	0	1
0	1	0	1



$D_0 = \overline{Q_0}$

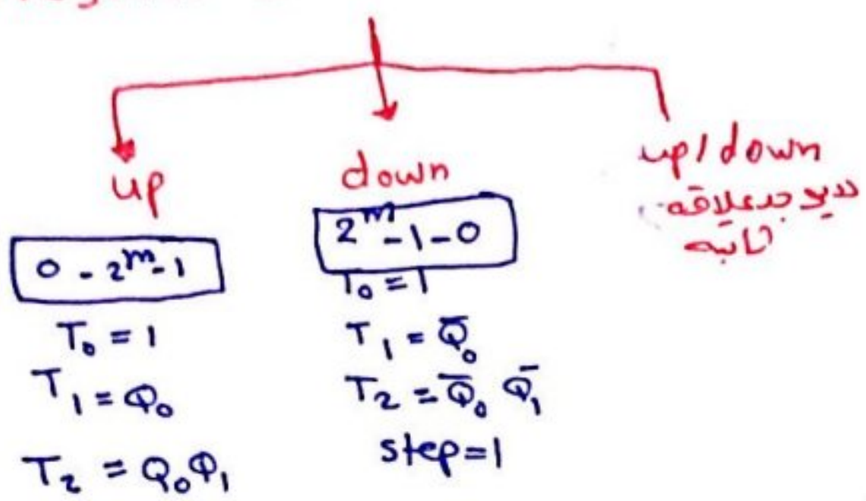
	00	01	11	10
1	0	1	0	1
0	1	0	1	0

$D_1 = Q_0 \oplus Q_1 \oplus X$

∴ up/down عدد T_0, T_1, T_2 فقط

لكن بـ (up/down) لايوجد علاقة ثابتة كما رأيتم T.T وأطلع على Kmaps

* regular (moore machine)



* Design acounter that counts 0,1,3,5 (101) → T.T /k-map/ بسم الشروط
→ simplyfy => up/down [1]
step=1 [2]

Q_2	Q_1	Q_0	$Q_2(t+1)$	$Q_1(t+1)$	$Q_0(t+1)$	T_2	T_1	T_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	1	0	1	0
0	1	0	x	x	x	x	x	x
0	1	1	1	0	1	1	1	0
1	0	0	x	x	x	x	x	x
1	0	1	0	0	0	1	0	1
1	1	0	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x

Ireg/up/step≠1

Q_1Q_0 / Q_2

Q_2 \ Q_1Q_0	00	01	11	10
0	0	0	1	x
x	x	1	x	x

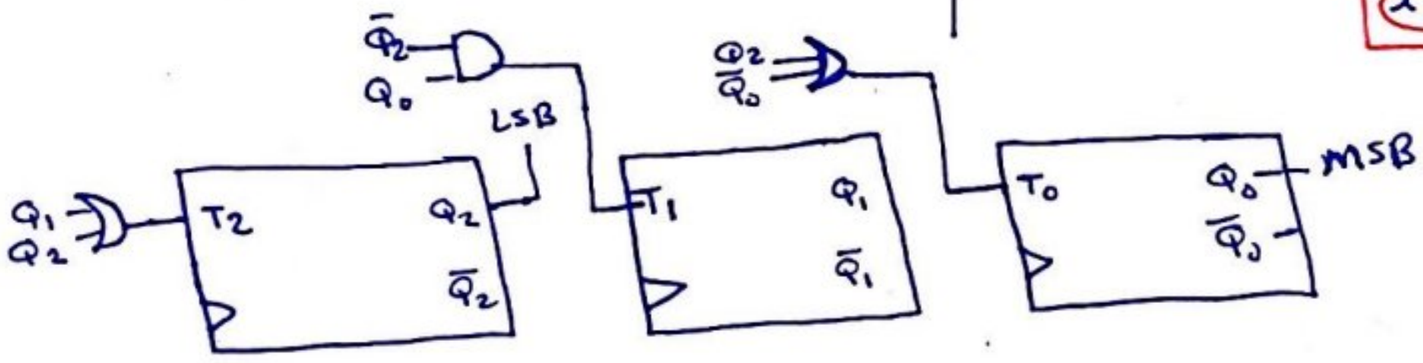
$T_2 = Q_1 + Q_2$

0	1	1	x
x	0	x	x

$T_1 = \bar{Q}_2 Q_0$

1	0	0	x
x	1	x	x

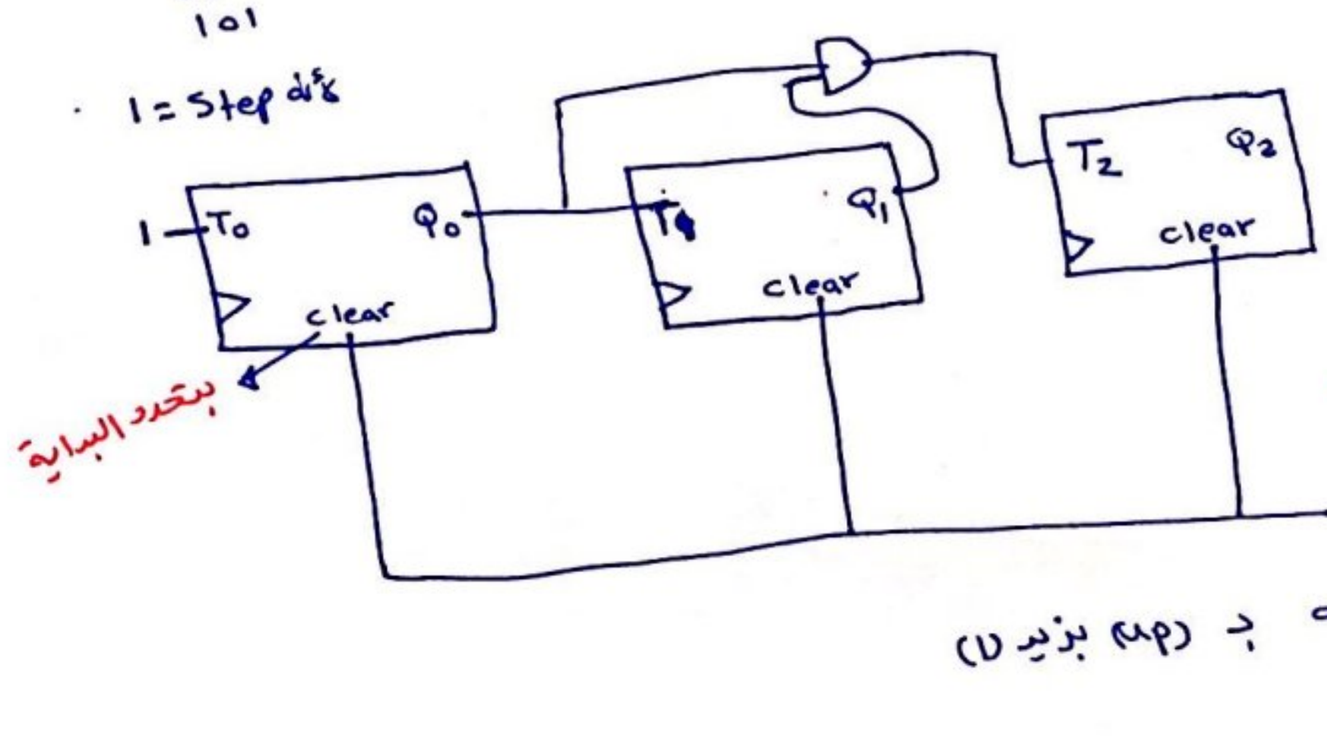
$T_0 = Q_2 + \bar{Q}_0$



Ex | 000 0 - 5 counter $2^3 - 1 = 7$ Iregular up step simplified

321
101

1 = Step دائم



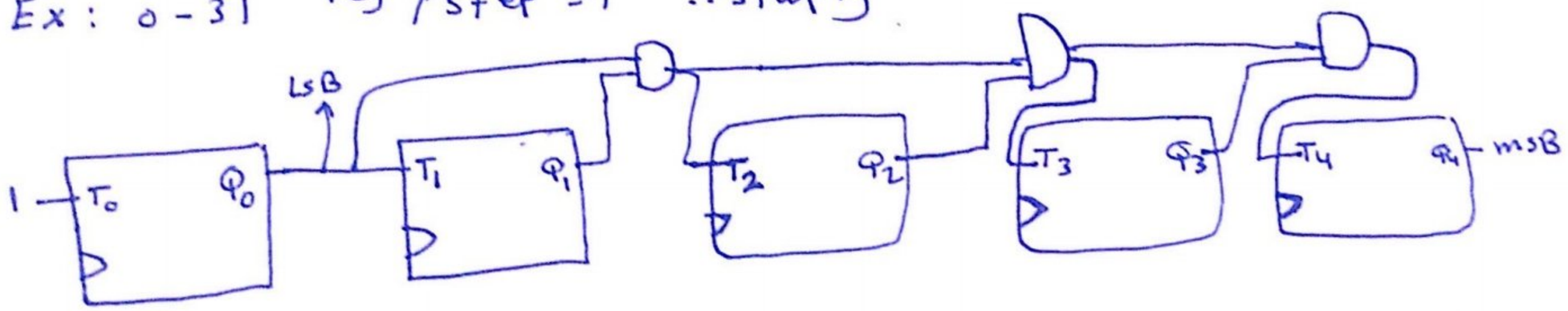
(5) للرجوع لأكبر رقم Clear(0)
set(1)

0 0 0
clear clear clear

110 = 6 بتحدد النهاية

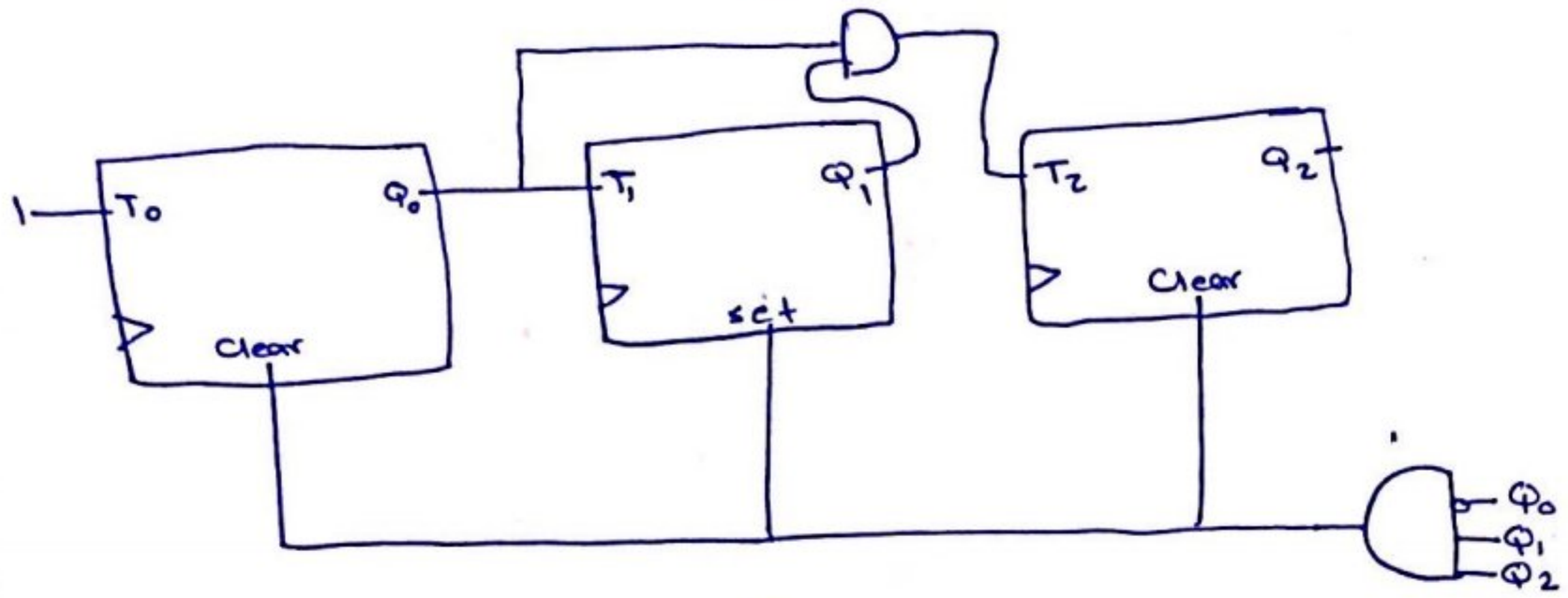
آخر رقم (5) بزيد اله (1)
وبشوف مين بتعتب وبنشبكه عند بداية and ← (NP بزيد 1)
counter لصفحه (5) عند

Ex: 0-31 reg / step = 1 ∴ simply

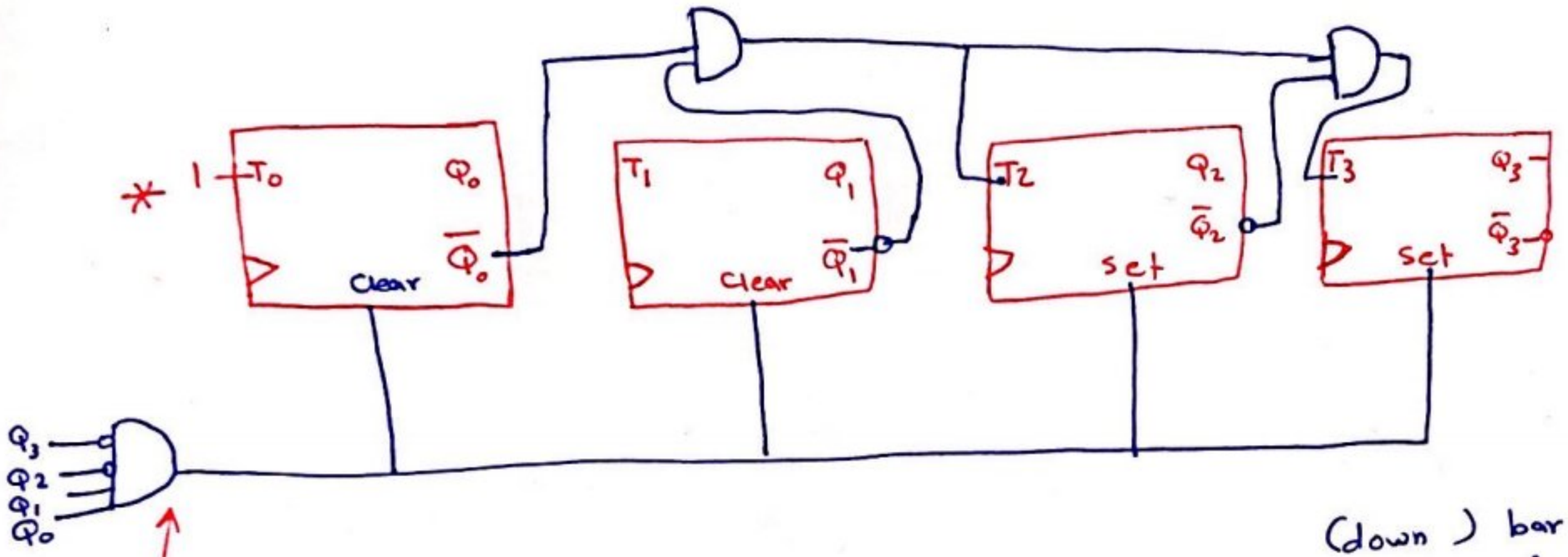


0-255 8-ff

*2-5 Counter Irregular [1] step = 1 [2] up simpifide.



2 = 2<1
010



الرقم 1 هنا
العائق + باقي بتّي
إياه
down عكسه
إذا up الرقم -1

بداية 12 بدأ من 12 1100

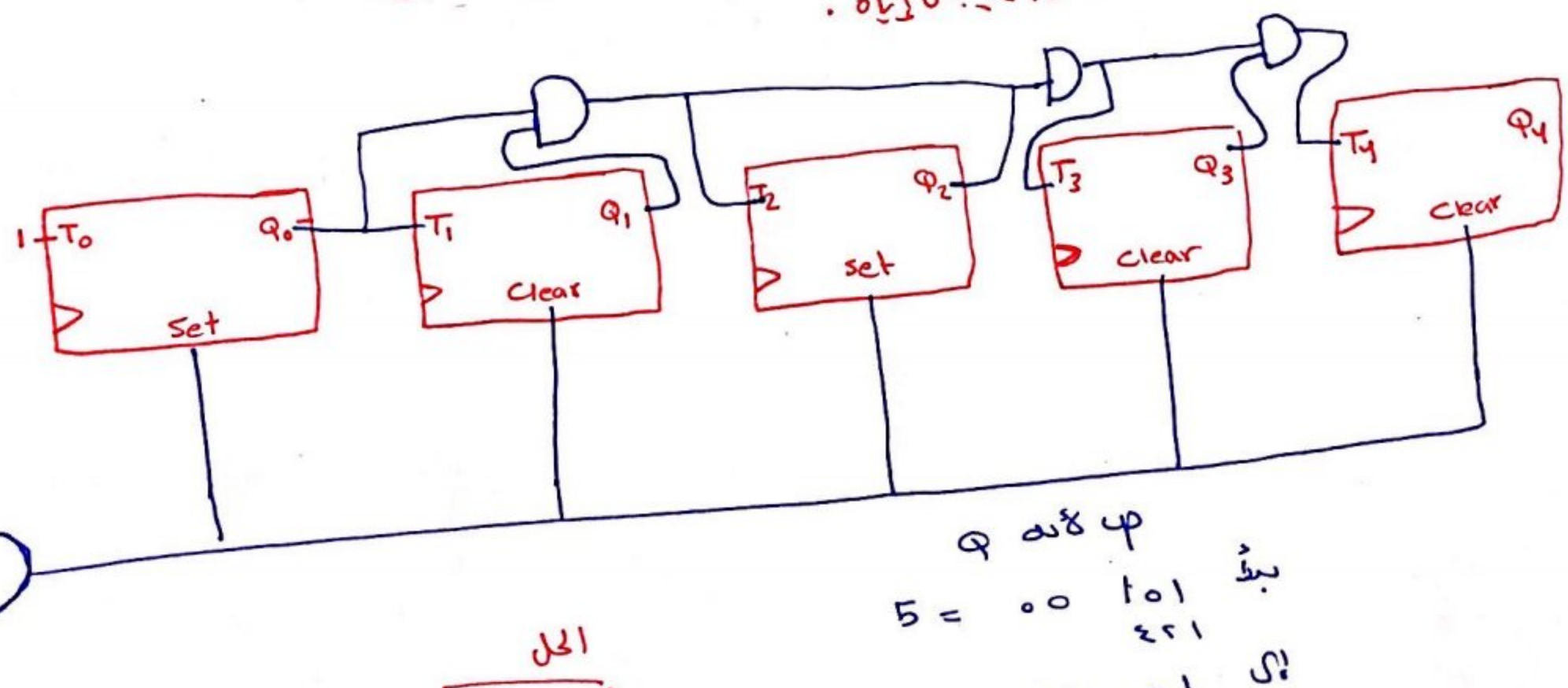
12

down +1 3 = 0011

3 + 1 = 4

12-4 down ←

لما بدّي أعد down بدّي أوصل الرقم أقل من الرقم يلي بدّي إياه . وهنا يرجع طبيعي أول وجديد



الحل

5-24

5-25

Q عكس up

بدأ 5 = 00101

4 2 1

11001

26 = 16 8 4 2 1

up-1

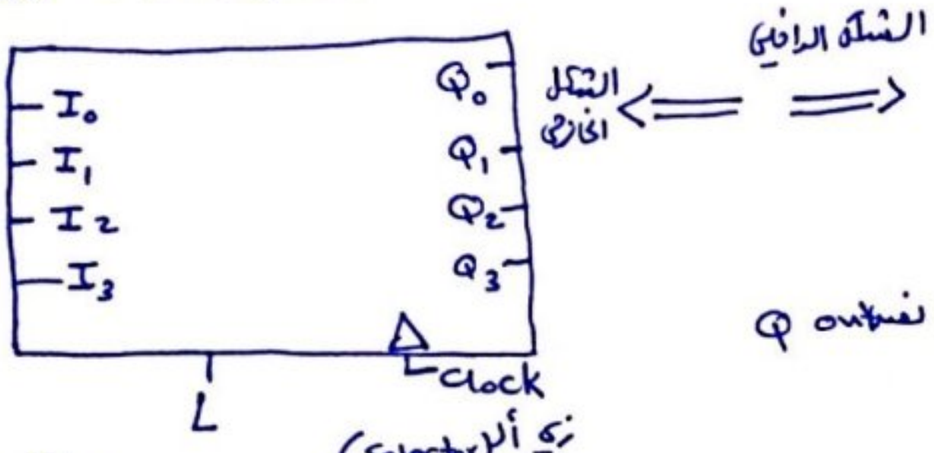
lec "14" Registers سجلات تخزن قيم binary داخلها.

(store binary information) D-f.f

- universal
- series in/out
- Parallel in/Parrllel out

* Parrallel in/out (المدخلات بدخلوا مع بعض وبطلعوا مع بعض)

4bit — — — —



الشكل الداخلي ⟸ ⟹ الشكل الخارجي

(selector زي ألـ)

(AH)

Data جديدة save تحل محل القديمة.

L=0 (store احفظ القيمة داخل registers) inactive

L=1 (Active) load / right (اكتب شيء جديد) New value (I) من بتجي

* لحتى (1) cycle وشان أكتب وبعين out put بتطلع كلها بعدين أكتب ي.

(shift Registers) (وحدة وحدة)

* series in/out

output و كذلك bit, bit بدخل (input) لـ

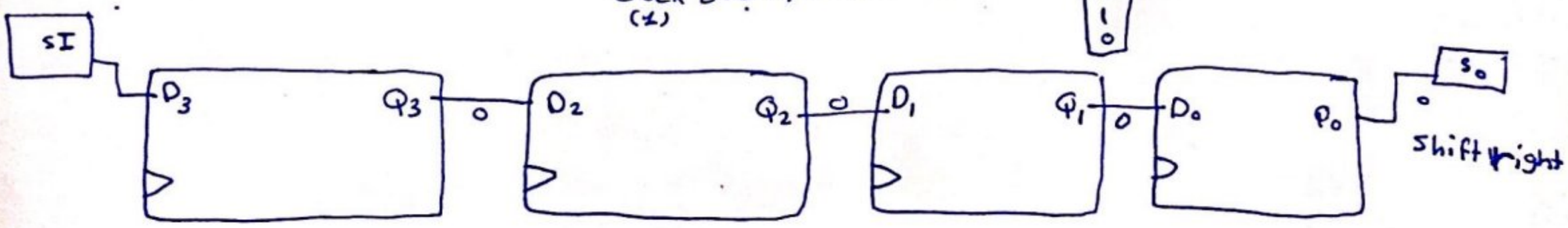
shift Right / Right shift

بدخل bit بدخل وبطلع — بدخل bit بدخل وبطلع — بدخل bit بدخل وبطلع — بدخل bit بدخل وبطلع

* حتى أكتب أي data جديدة احتجت clock 1 to write

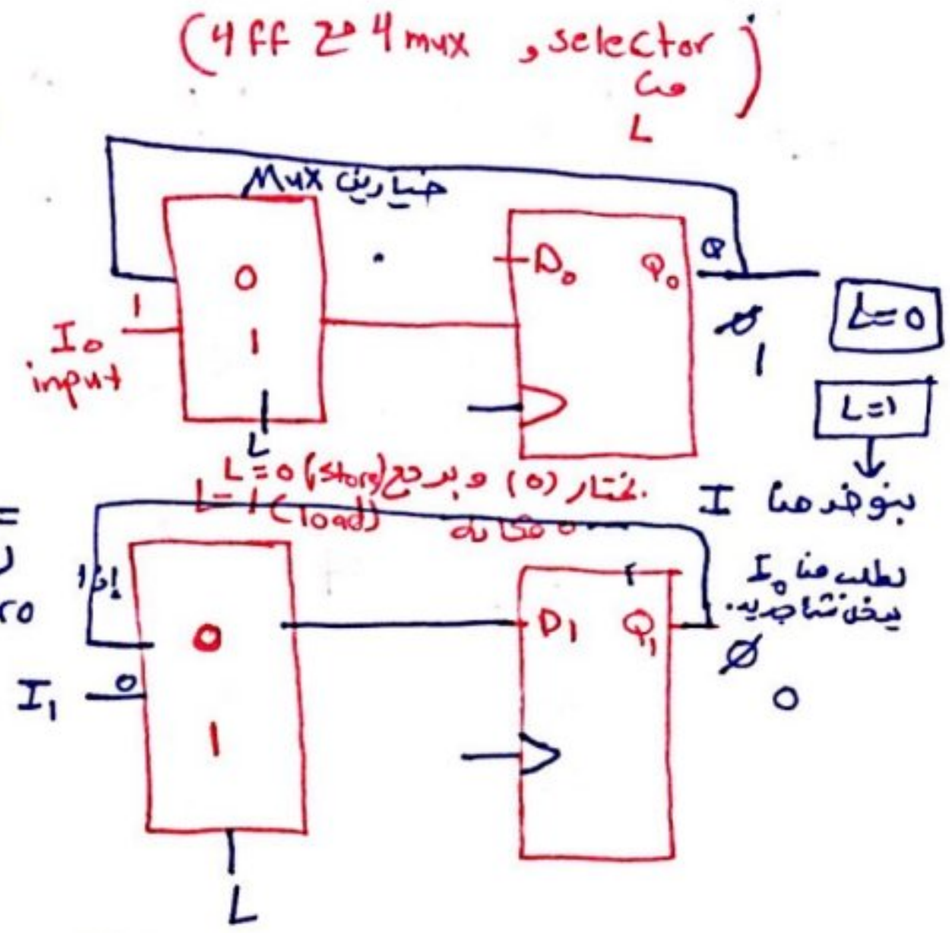
(parallel load بدنا)

لو ٤، ٨، ١٦، ٣٢ بحتاج clock (x)



L=0 كان كال.
L=1 كان كال.

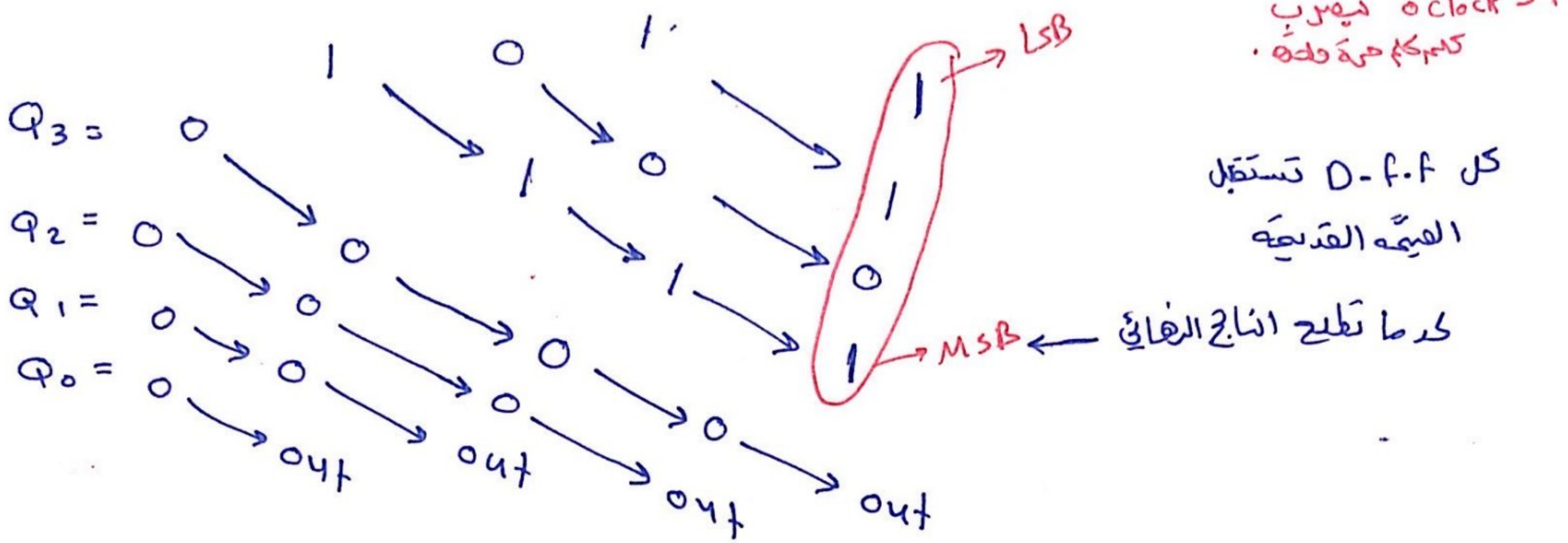
(4 ff ل 4 mux ، selector هي L)



Ex :

داعماً نبدأ Reset (0)

أو clock تنقرب كم مرة وحدة.



كل D-f.f تستقبل القيمة القديمة

كلما تطلع الناتج النهائي

* كتاح إلى cycle (4) للكتابة writing

و (3) cycle للقراءة

الكل (7) cycle.

4 bit

أول وحدة وصلت هي already نقرأ (msB)

(bit)

بالامتحان (6) متاذ كم cycle بحتاج ؟!

write
read
cycle

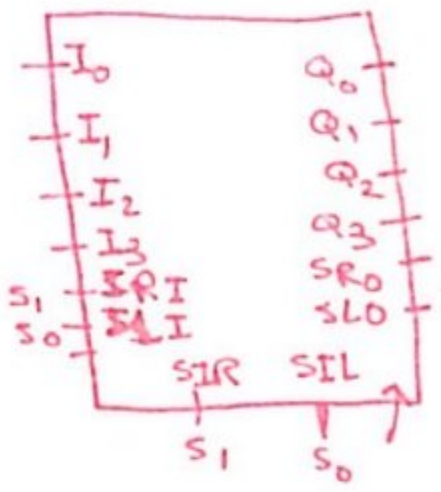
write 6
read 5

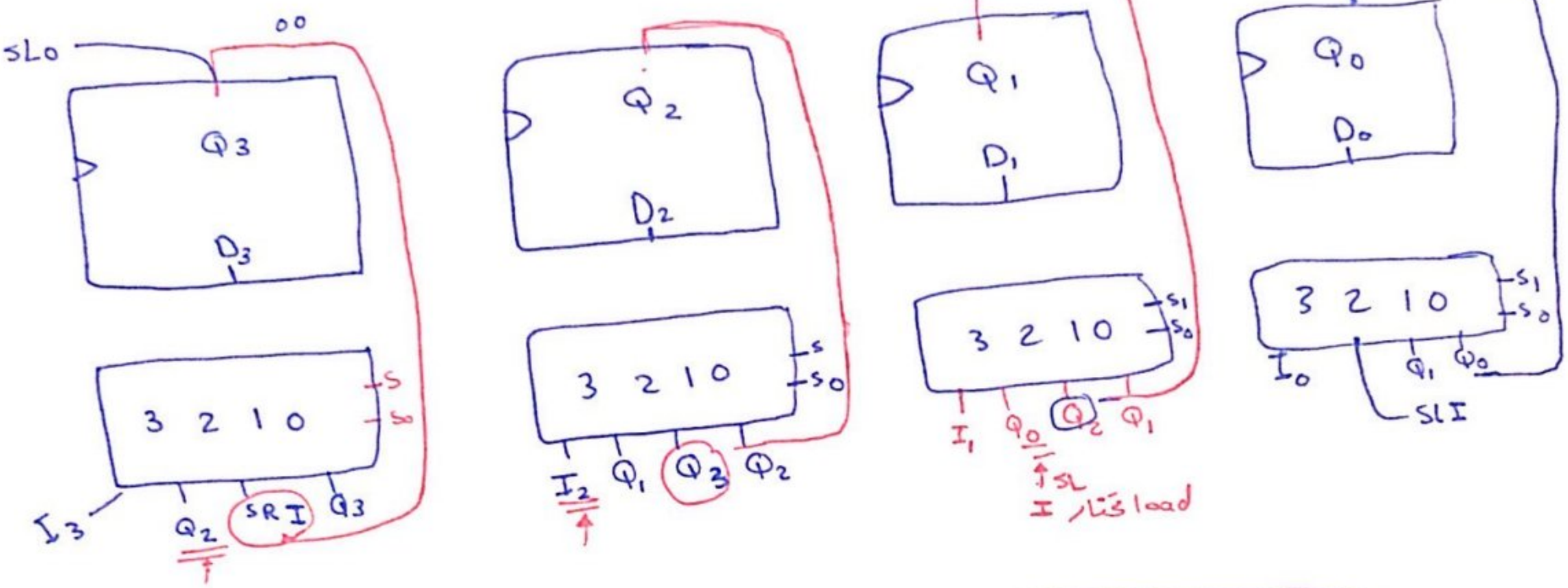
المجموع 11
cycle.

* universal Register

S_1	S_0	type
0	0	Store (No change) save
0	1	shift right (SR)
1	0	shift left (SL)
1	1	Parallel (بدخلوا هيك بطلعوا هيك)

(I تختار) load





Ex: $Q_3 Q_2 Q_1 Q_0 = 1100$ / SRI = 0 / SLI = 1

$I_3 I_2 I_1 I_0 = 1111$ معلومة غير مهمة بتجاوبها اذا أعطى

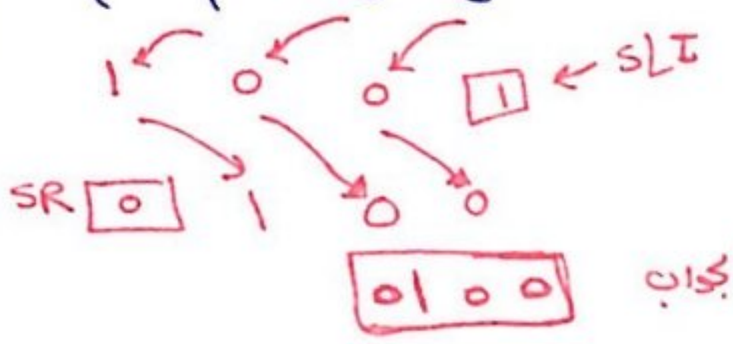
$S_1 S_0$: 0 0 cycle(1) Find
1 0 cycle(2) قبل
0 1 cycle(3) البدء

أول دورة (00) No change
ثاني دورة (10) SL
ثالث دورة 01 SR

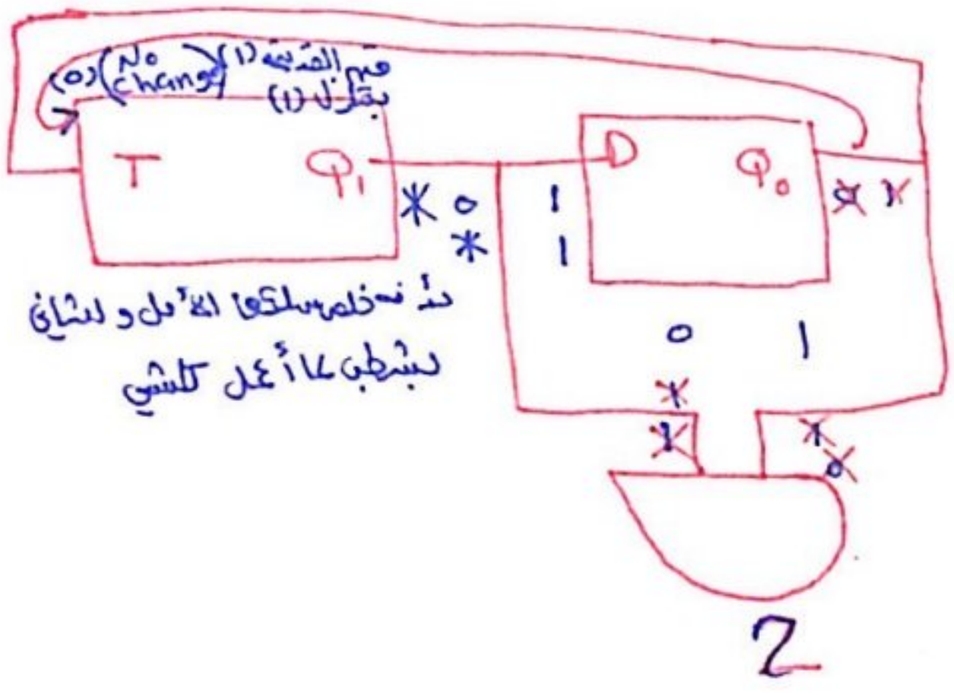
التوك للسيار

SS
00

Q_3 Q_2 Q_1 Q_0 after 3 cycle ?!
3 دورات
1 1 0 0
1 1 0 0



Ex



$Q_1 Q_0$ Find 2 and $Q_5 \rightarrow Q_1/Q_0$ after 2 cycle?!

$Q_1 = 0$
$Q_0 = 1$