

تلخيص مادة

Digital Design

Part "2" ch: 5.6



إعداد وتلخيص المهندسة : دارين ياسر داود
كلية الهندسة _ النظم الذكية
2026

کتاب کد ملی یو د لاندې د وړاندوینې د کتاب
 Truth Table د افریځه پایلې stat. table

Lecture 19: Chapter 5 Synchronous sequential logic

الشارة الضوئية

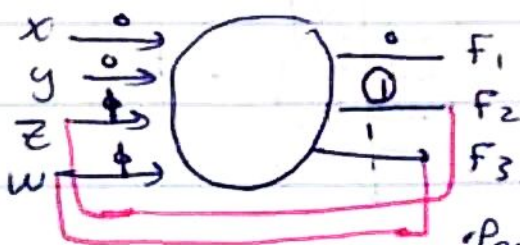
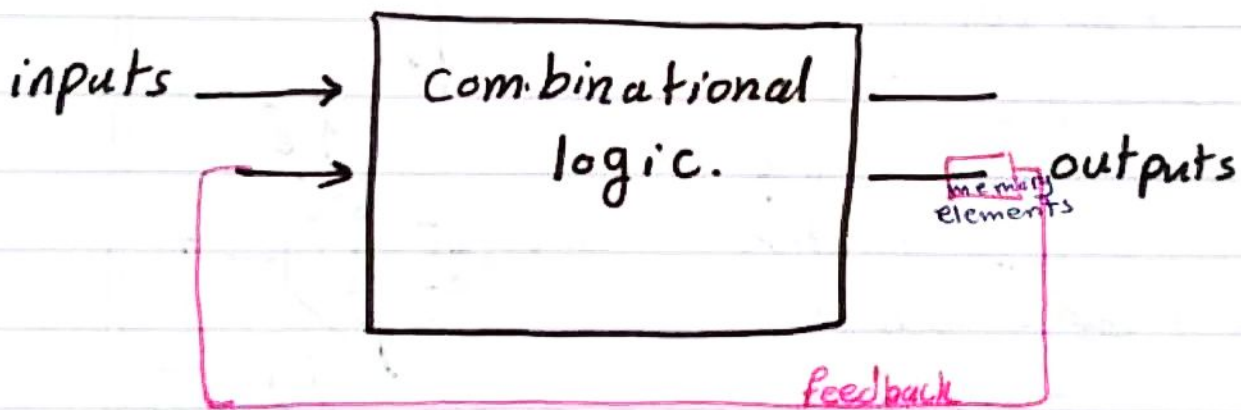
3 output
 لاندې غږېدې 3
 غږېدې

0	0	1	1
1	0	1	0
0	1	0	0

○ آفر
 ○ آفر
 ○ آفر

Stat د افریځې د افریځې
 State 3 State 2 State 1

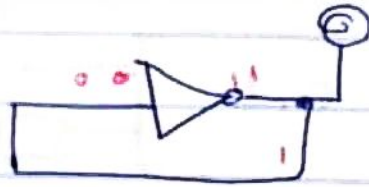
sequential logic



- في الغالب ال feedback يؤدي إلى تغيرات في inputs.

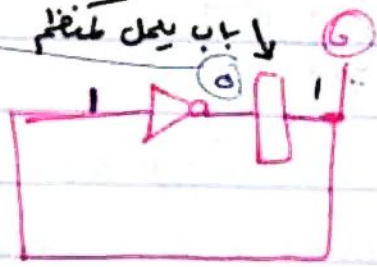
- ليس بالضرورة كل ال inputs التي داخلية مع ال circuit يكونوا راجعين من ال outputs. ولكن "feedback" لازم يكون موجود.

لهذا غيرت ال output
 لي يتغيروا ال input



حفظ نصفي، تطني رشي
رج ألقا اشوف متو منها
لذو سرعة (باللغة)

الصفر ممنوع يدخل دقود
محلته أفتح الباب لحظة
معينة واسكره بسرعة
مترجع الدائرة تستقل

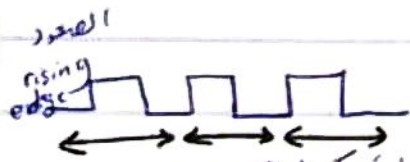
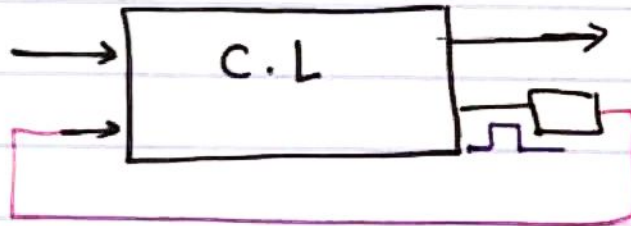


وظيفة memory element :

ال memory element هو عبارة عن بوابة لا تنسى
بال feedback إذا بالازد من ال input
"مربوطه بزمن"

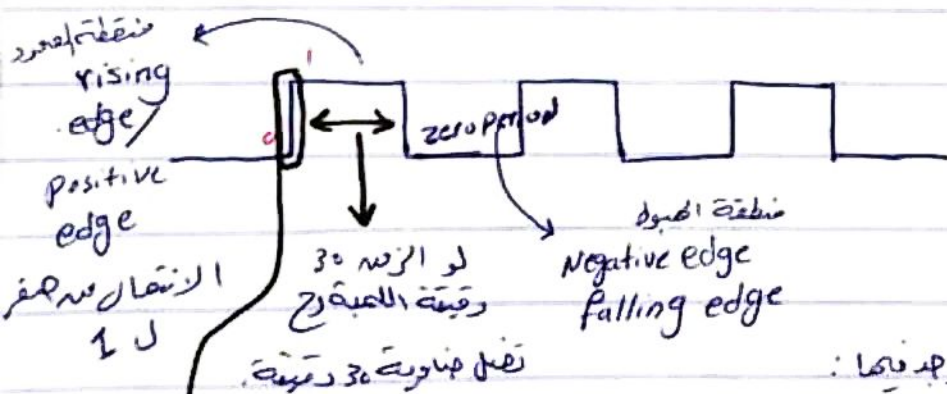
ف حفظ الدائرة عند صفر
حليف وبعد ال inverter
كلو 1 وصل عند الباب
لما أفتح الباب الدائرة تستقل
تصوي

القدرة على تخزين المعلومات



Sequential logic تتعامل مع الزمن بتبنيات

لوانا استغيت عن الزمن فالباب رج سكر للأبد رشتت الدائرة كلها هي



نقل منوية 30 دقيقة

في النصف الواحدة يوجد فيها :

Zero period / one period / rising edge / falling edge

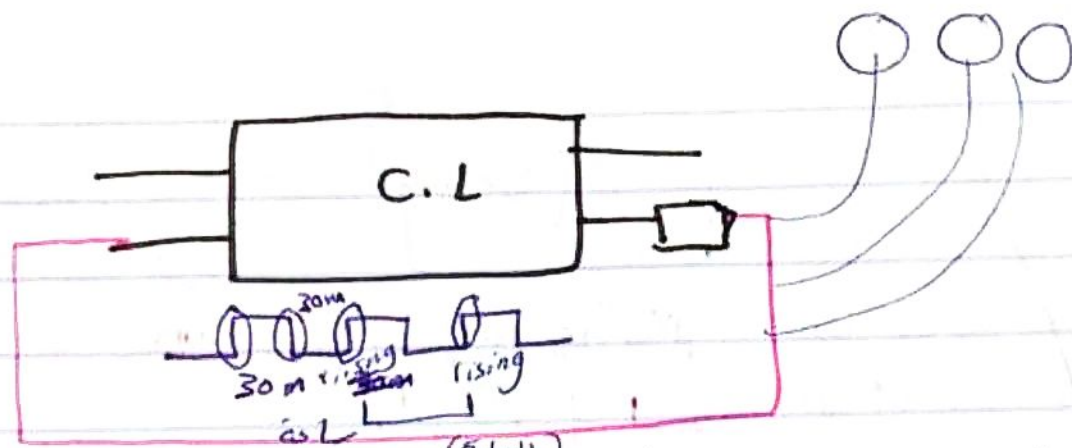
لما أضمم الدائرة ومايك ياها، إلا نفتح دسكربفة معينة كل زم آخر

لحظة زمنية معينة وأفتح اسباب عطلو واسكره عطلو على أنه أقدر اشوف

التحويل الى ريشي بالدائرة "محلها خطتها خطوة الصلوة أو الصعود"

فقط في النصف الواحدة مرة واحدة يفتح

رسم ال clock
كل ساعة



كل ساعة يجب نبضة لل memory (المسألة)

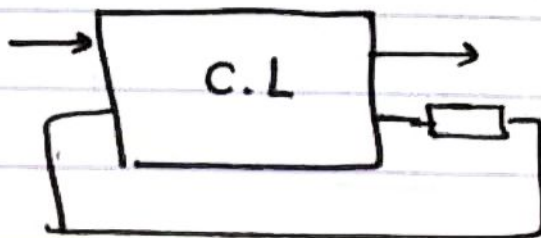
اول نبضة تأتي مع كلمة جرمية على rising في تفتح البوابات فتدخل كل المعلومات
وتتكرر فالاشياء تقترن بكانت احمر احمر اضهر مع نصير احمر مع الصفراء اضهر
ثماني مرج تفضل ثابتة ساعة اخرى لانه هاي النبضة مانعها ايا rising واحد
النبضة العادية بعد ساعة تكون عليها rising ، هذا الـ clock يؤدي الى تغير شكل
الاشارة State من لآخر.

Lecture 20 :

memory element

Flip flop:

latch

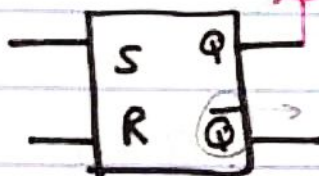


S R latch

D latch

active high

active low



active high S R latch

مخرج

فقط

set reset

S R

Q

0 0

No Change

0 1

0

1 0

1

1 1

impossible

لما اخط S ب 0 وكانت قبل

اللمبة ضاربة فلن تتغير تبقى كما هي

وعلى لو كانت طافية رجعت طافية

$$Q(t) = Q(t+1)$$

$$Q^+ = Q$$

حسينها active high لأن الوارد

التي بحد توصيل

S R

Q

0 0

P.S

0 1

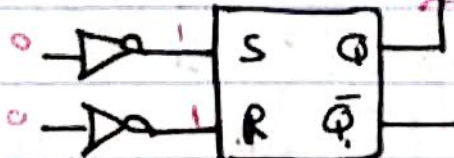
1

1 0

0

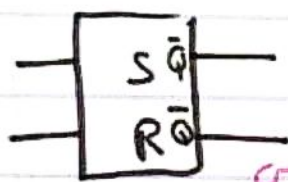
1 1

No change



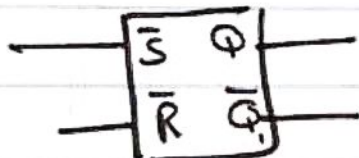
active low S R latch

الفرق بين active و law هو inverter



active
الواحد فعّال، الثاني غير

S	R	Q
0	0	No Change
0	1	0
1	0	1
1	1	!!

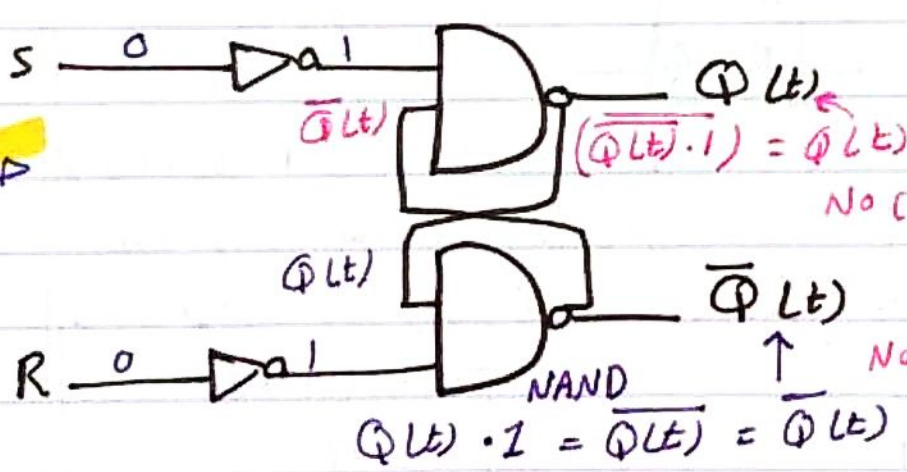


low
المنخفض فعّال، الثاني غير

S	R	Q
0	0	??
0	1	1
1	0	0
1	1	No Change

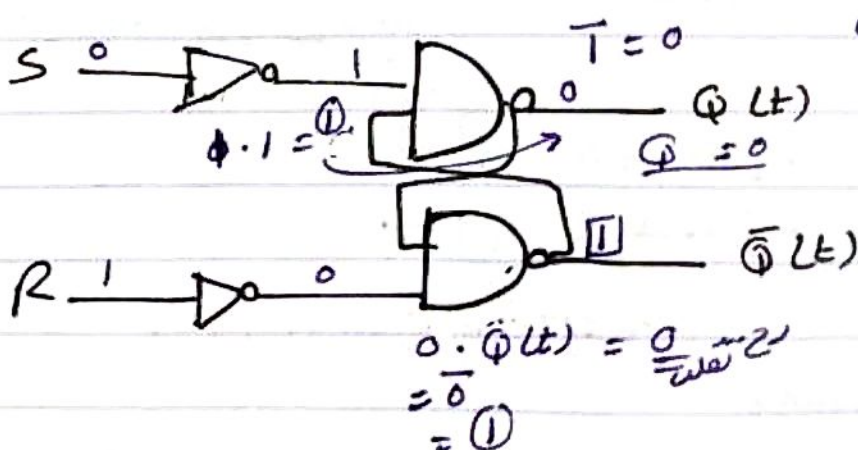
active high latch دارة

(1 0) 1



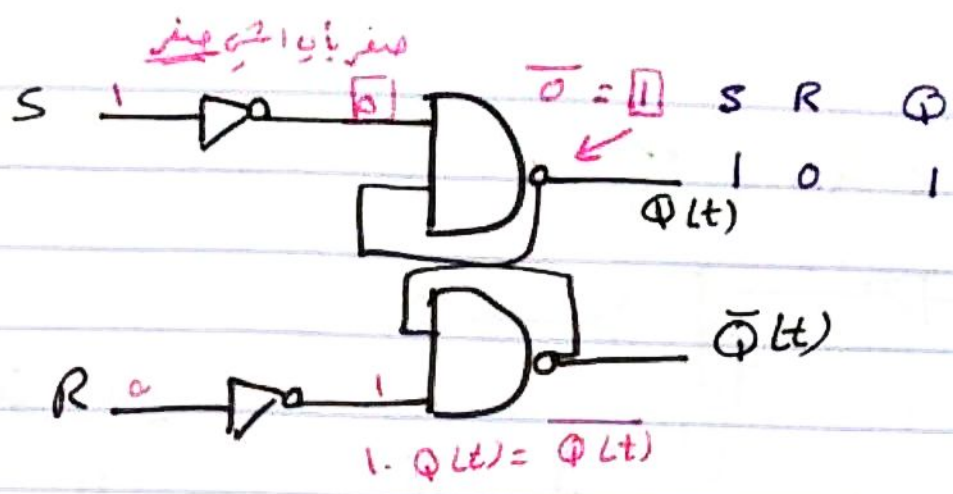
S	R	Q
0	0	No change
		$Q(t+1) = Q(t)$

(2 0) 1

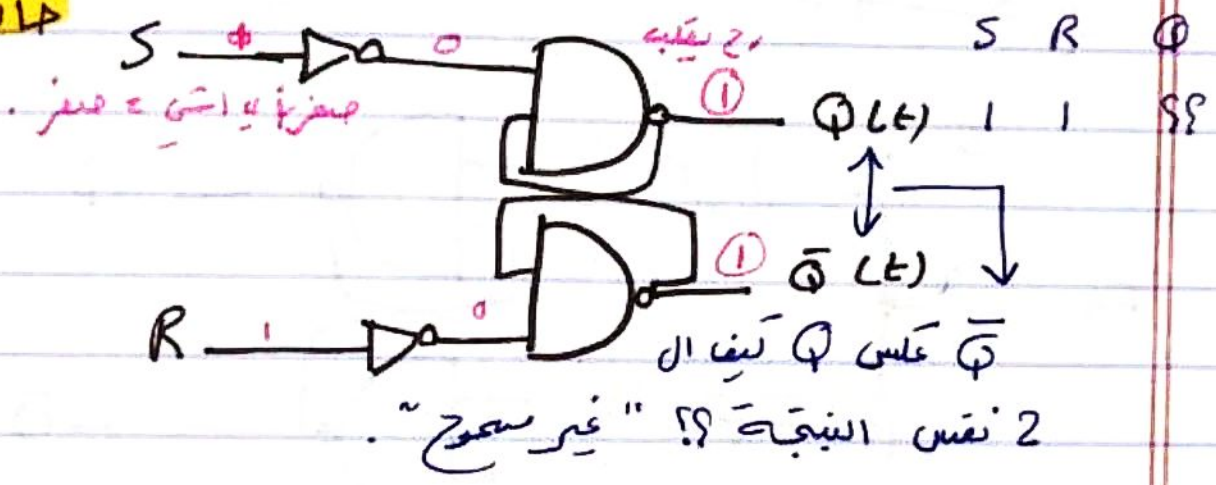


S	R	Q
0	1	0

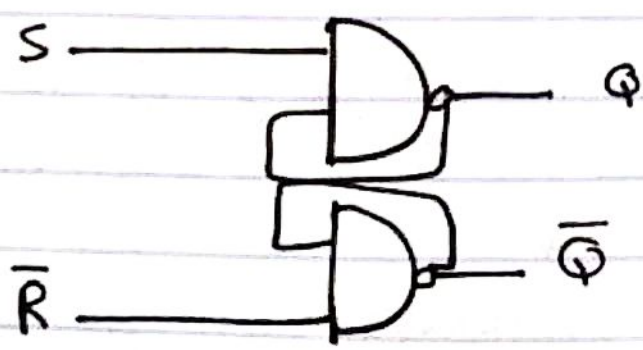
الـ 3



الـ 4

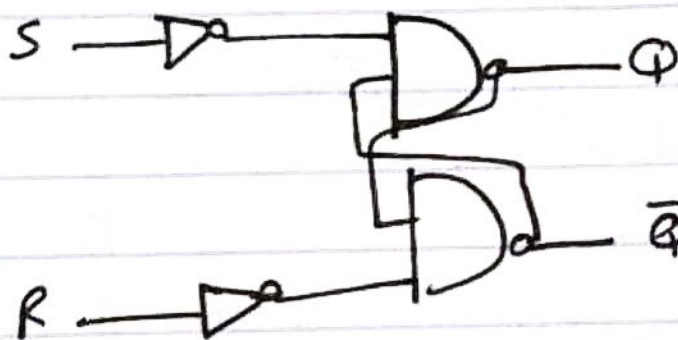


في لو بي ارجع active low : نفس high
 دالة نصف inverted . "الانفيرس بلوغا يعني"

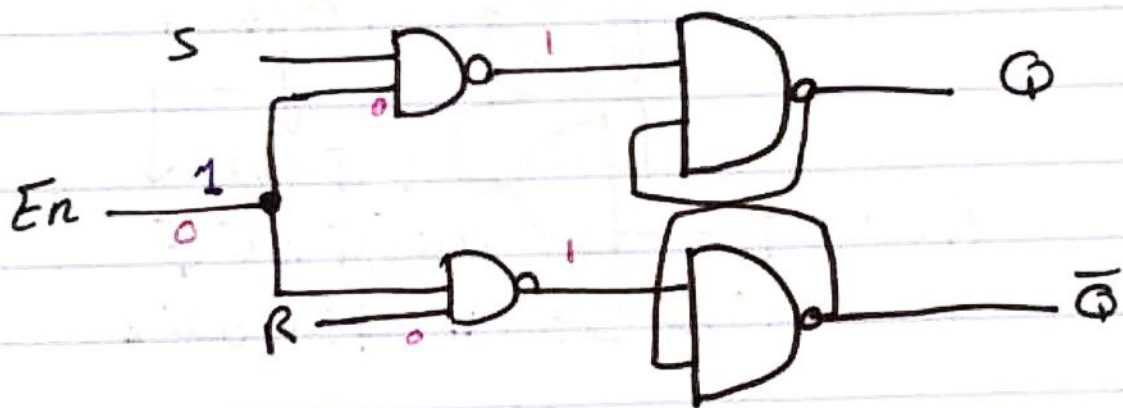


دبطه نفس الحالات

active high SR :



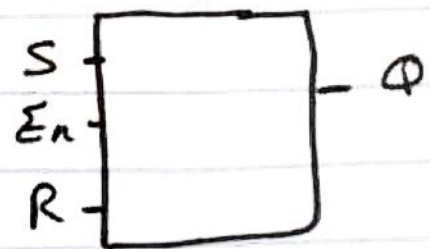
لatch
↓



لو هنتـ Enable وتنمـ واحد فله يؤثر في العملية
بـتغل كما هو.

active low
للك لو هتغير تغير فيرجع الـ S والـ R ربيـت كما عمل
No change 1 1

	S	R	En	Q
بغض، انتظار	x	x	0	No Chang
مع شغل	0	0	1	No Chang
حليـ	0	1	1	0
قالب	1	0	1	1
الـ En	1	1	1	??

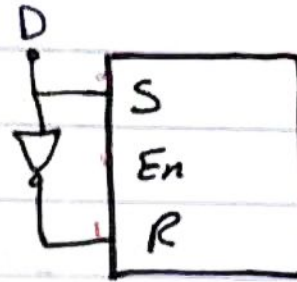


إذا En فير ونا في ابواب فله
ولـ إذا واحد كما في فتوح.

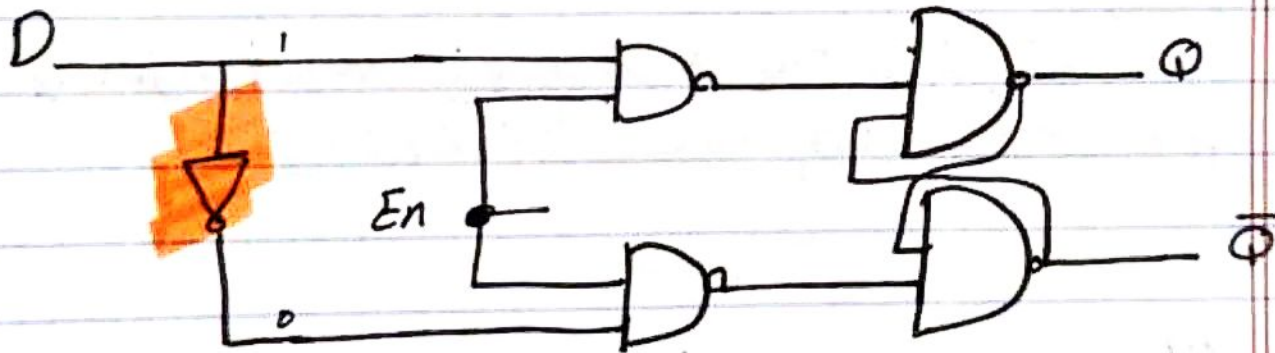
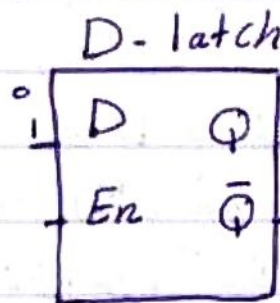
D-latch :)

D latch

D	Q
0	0
1	1



D	En	Q
x	0	No change
0	1	0
1	1	1

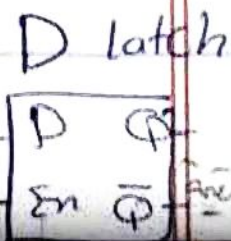
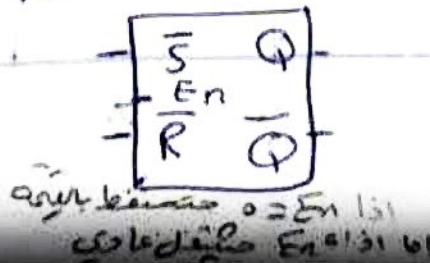
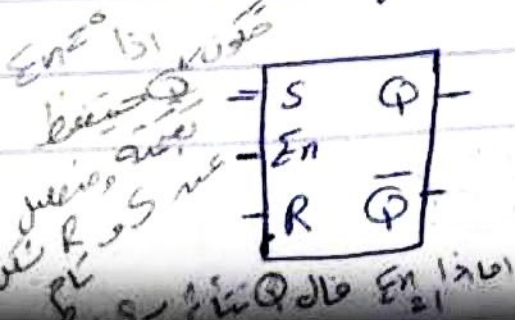


inverter $\bar{D}$ is used in latch $\bar{D}$ and $\bar{SR}$ needed

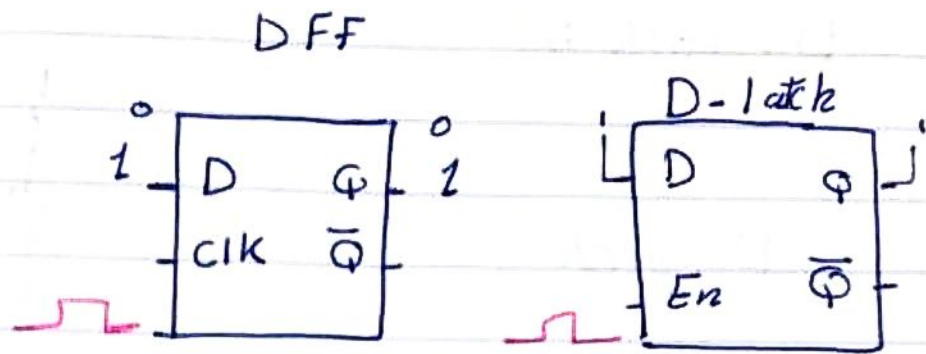
Conc:

active high SR latch

Active low SR latch



DFF



هل لاحظ ان D latch و DFF نفس العمل ولكنه هناك فرق محوري؟؟
 متى يفتح البوابة وان Q يتشوف القيمة؟؟
 شئنا ان Q يتأثر بقيمة D في الحالتين؟

في Flip flop معصم على انه يفتح لحظة معينة من الزمن

في D latch ان Q يتشوف D في فترة زمنية طولها CLK
 يتفتح البوابة لمدة زمنية (فاي تغيرات رح ياشر على ال output).

في D Flip Flop يتفتح البوابة لحظة زمنية خفة (rising)

* الفرق الاساسي بينهم هو طريقة التوقيت
 وليفيع آخر زيا البيانات *

D latch → كما في المستوى

عندما $En = 1$ → المخرج يتبع الدخل مباشرة

عندما $En = 0$ → يحتفظ بالآخر قيمة. الفكرة عانه باب مفتوح

الباب مفتوح (البيانات غير) ، الباب مغلق (آخر قيمة تبقى مخزنة).

En D Q

1 0 0

1 1 1

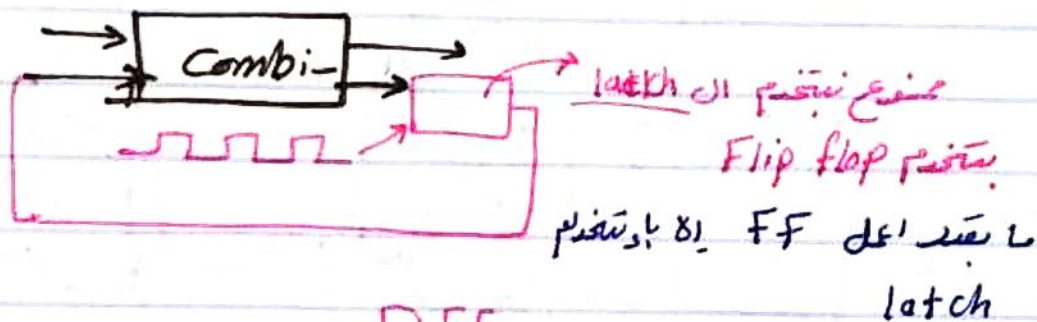
0 X 0

يحتفظ بالقيمة

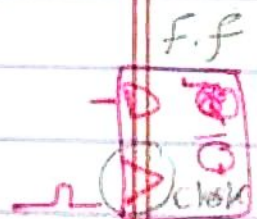
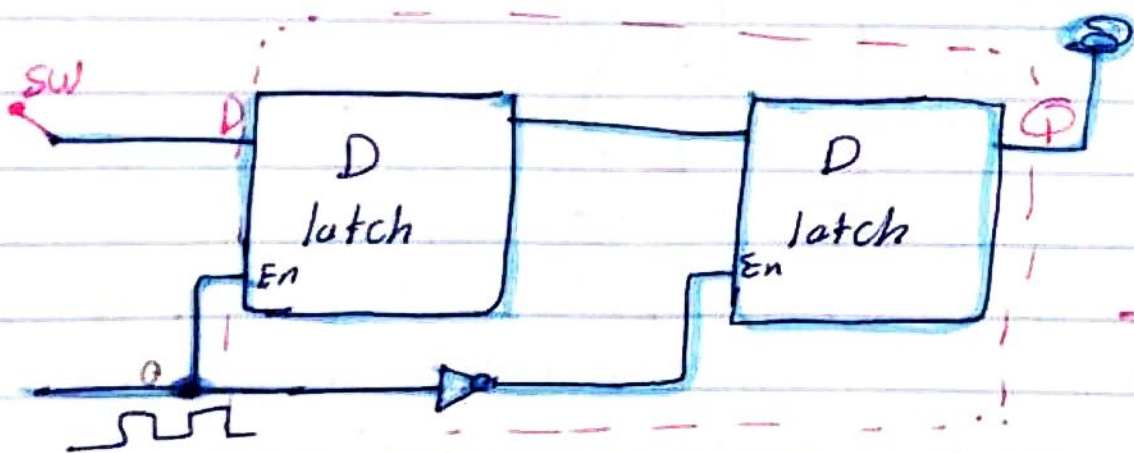
② D Flip - Flop :)

حسابي للحالة .
 يأخذ قيمة D فقط عندما : ① حالة لصاحبة ② حالة للحالة للحالة
 غير هذه يبقى ثابت .

Clock Edge	D	Q
↑	0	0
↓	1	1
بدون Edge	X	احتفظ بالحالة



DFF

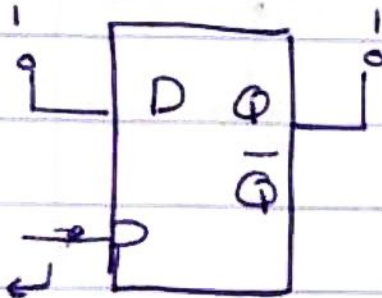


منفذ للحالة
 rising
 falling

يتم عمل جولي من نبضة واحدة وينتج الدائرة ونسبة
 تغير ال output يقترن ب المدخل D ما إذا القيمة Q يتغير ال D
 عند لحظة rising يا falling هاد يكون ال Flip-flop
 * اما لو Q يتغير ال D بالترتيب هاد يكون ال Flip-flop
 وينتج وينتج .

Lecture 21 :

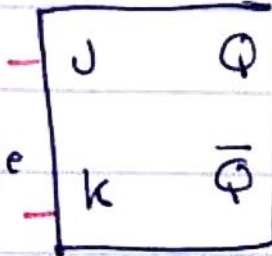
DFF



Flip-Flop

D	Q
0	0
1	1

إذا خيّننا دائرة falling
إذا خيّننا دائرة rising



JK-Flip-Flop

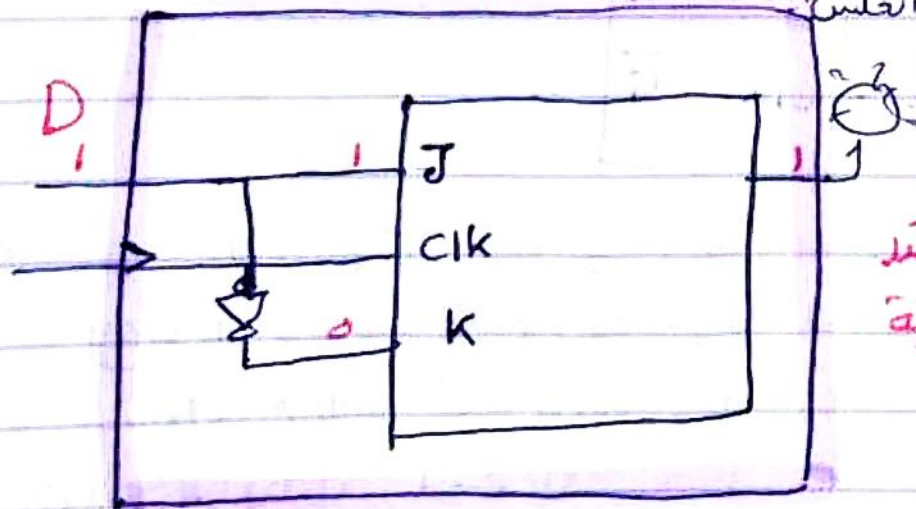
J	K	Q
0	0	No Change
0	1	0
1	0	1
1	1	Change/Toggle/Complement

Change/Toggle/Complement

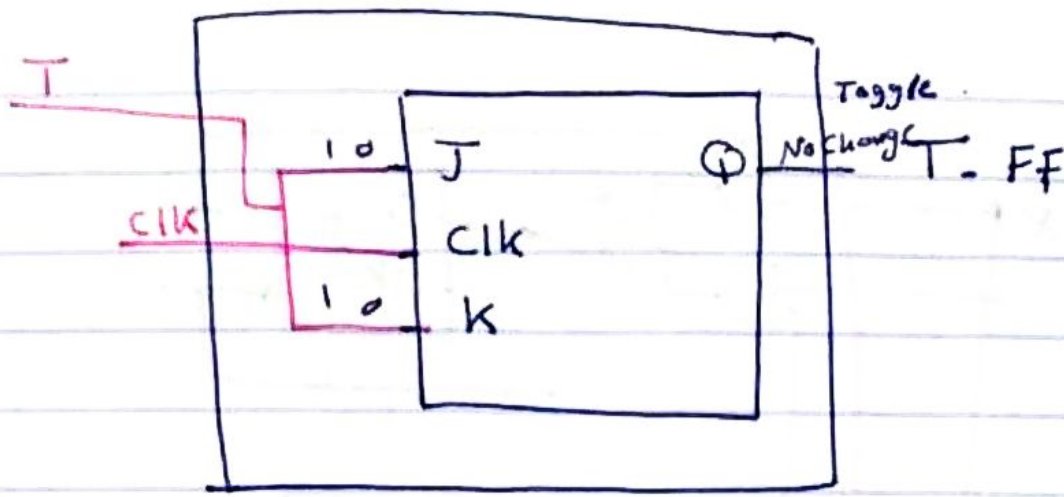
Q = J K

دائرة جايبي rising في لحظة Q يغير الباب بشرف
قيمة J و K إذا لا فهم غير رفع تفهم الدائرة التوضيحية
(القيمة مخزنة)

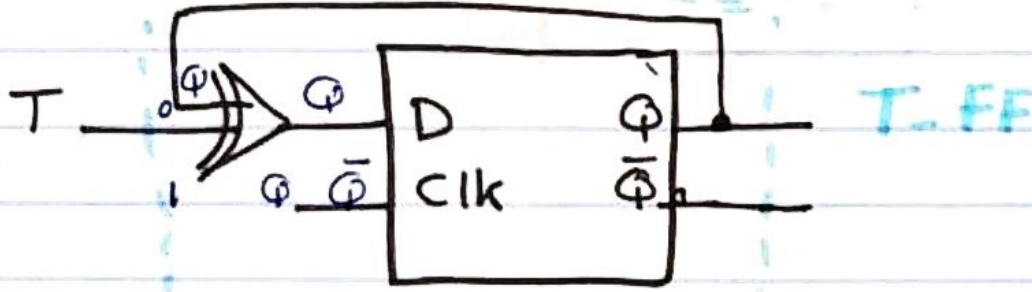
لو كان على J و K 1 في لحظة ال rising Q رفع تفهم إذا كانت
خارجية رفع تفهم داخلية



لو عرفت FF-K ال بعد
أعمله DFF بالطريقة
هنا



T-FF لود صلیت J مع ک بدون inverter ہے
 D-FF لود صلیت J مع ک ب inverter ہے



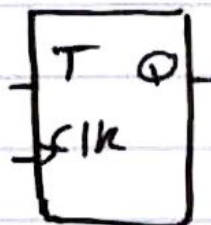
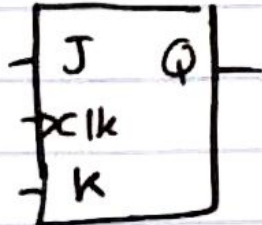
Change $\bar{Q}$ $\bar{Q}$ No change $\bar{Q}$ لو 1 رہ تبغیر (تغییر)
 لو 0 رہ تبغیر

3 type F.F

D.FF

JK.FF

T.FF



Ch table

truth table

D	Q
0	0
1	1

Ch eq $Q = D$

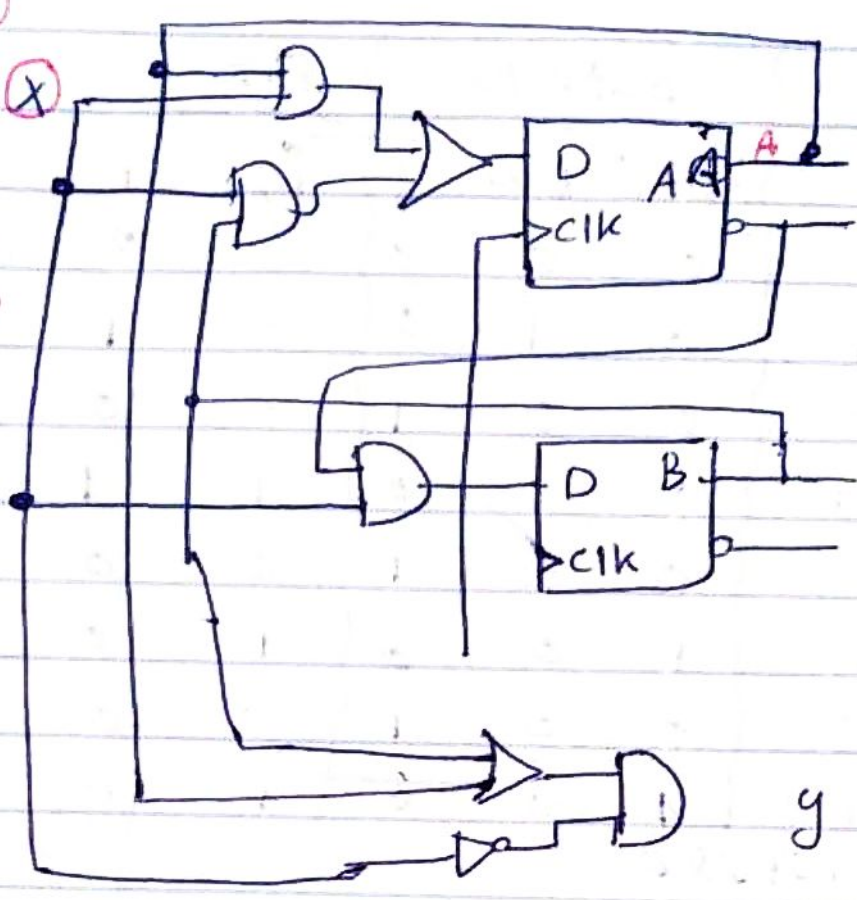
J	K	Q
0	0	No change
0	1	0
1	0	1
1	1	Change

Ch eq $Q^+ = J\bar{Q} + KQ$

T	Q
0	No change
1	change

Ch eq $Q^+ = T\bar{Q} + \bar{T}Q$

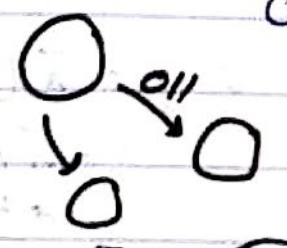
نوع نوعه من
 input
 input
 input
 feedback
 input flip flop



seq
 sequential
 flip-flop
 feedback
 seq

Circuit

اذا
 circuit



State Table
 State Diagram

input
output, $\leftarrow$ F.F

(input) 2 bit, 0, 1

inputs			outputs			F.F	
A	T	X	A(t+1)	B(t+1)	y	D _A	D _B
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1
0	1	0	0	0	1	0	0
0	1	1	1	1	0	1	1
1	0	0	0	0	1	0	0
1	0	1	1	0	0	1	0
1	1	0	0	0	1	0	0
1	1	1	1	0	0	1	0

State
table

لأنه إذا D صفر مخربها 0
مخربها 1

ما بقدر احب ان output, لا اذا عرفنا معادلات D في F-F

$$D_A = XA + XB$$

$$\begin{matrix} A=1 \\ X=1 \end{matrix}$$

$$\begin{matrix} X=1 \\ B=1 \end{matrix}$$

$$D_B = \bar{A}X$$

$$\begin{matrix} X=1 \\ A=0 \end{matrix}$$

$$y = \bar{X}(A+B)$$

$$y = \bar{X}A + \bar{X}B$$

$$\begin{matrix} A=1 \\ X=0 \end{matrix}$$

$$\begin{matrix} X=0 \\ B=1 \end{matrix}$$

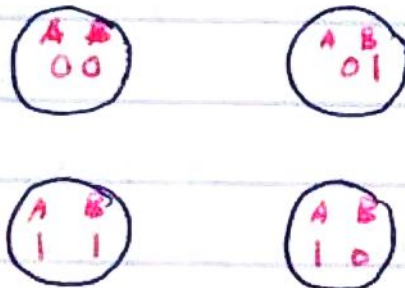
2 Flip-flop

4 حالات



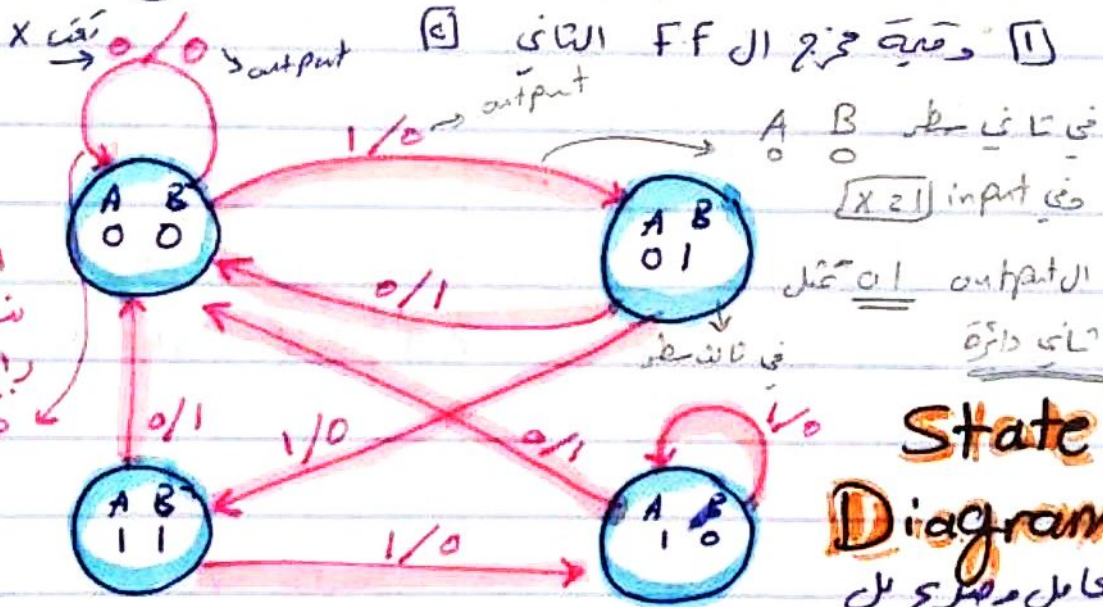
Flip-flop

عدد الساعات 2



المعنى $\Rightarrow$ شكل هاتين الدائرتين معناهما قيمه مخرج ال F.F الاول

والثاني مخرج ال F.F الثاني



State Diagram

وصف كامل وصريح لل Circuit .

توضيح: مثال في ثالث سطر بالجداول input A B X

دعنا نكمل الدائرتين $\begin{pmatrix} A & B \\ 0 & 1 \end{pmatrix}$ تمام دجيس؟؟ ال output عنا
دعنا نكمل الدائرتين $\begin{pmatrix} A & B \\ 1 & 0 \end{pmatrix}$ ال اول $\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix}$ جميع احوال
عليها دجيس input X
output Y

عملت 4 دوائر لثاني بيدي العمل للابدائيات ولتفريجات يمكنه للدائرة .

مهمة التي عملت؟؟

State Diagram

لتفرض انه ما عملنا الدوائر ، ورضنا بدنا نقول على السوية الاولى

وندخل القيمة المتزامنة ل A و B و 00 ورضنا على $3clk$

فكيف نحصر الدائرة ب $3clk$ ؟؟

في اول rising مثالي : $A \ B \ X$ بيدي الحرف هو القيم التي يكون

نيل من يفتح الباب ويقرأ X و F و A و B اصب تدقية لم يكون

قبل تدخل ال $F.F$ ، فاول rising مثلا يكون $B = A$ عليك عليك

وال rising التي وراء برهتوج اعل نفس القصة ...

" نيقدر اخلاها لله بناخذ وقت + بتاخم الراسي ن "

على اعتبار $12X$

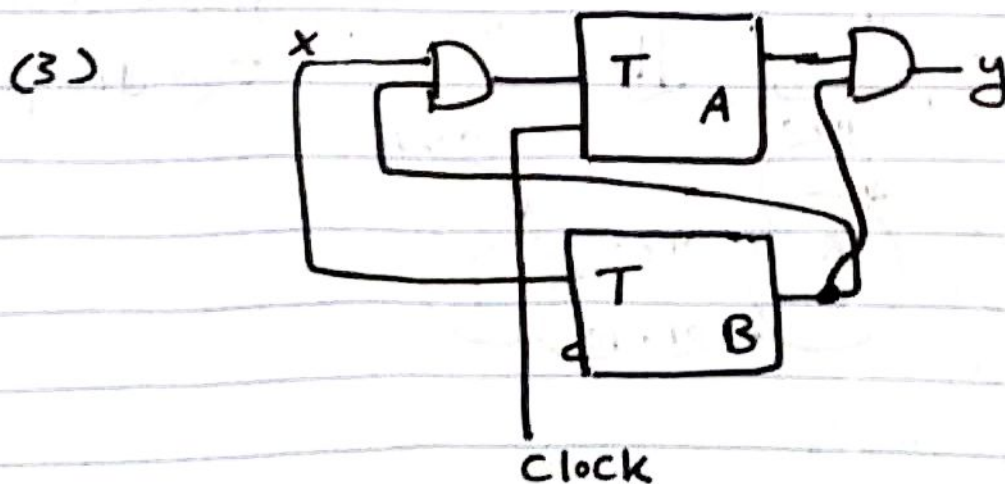
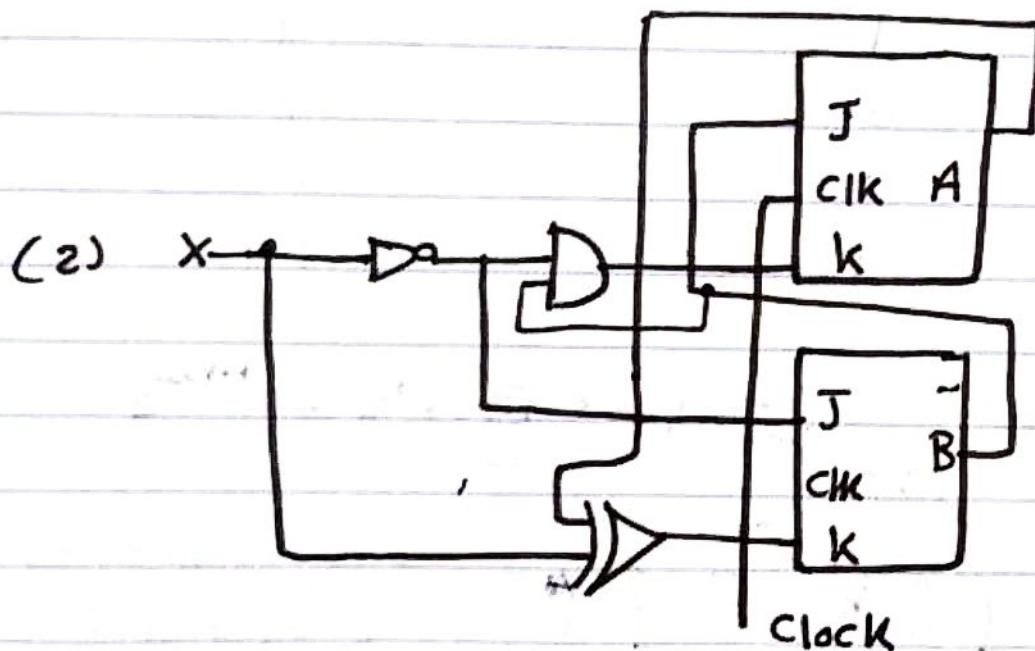
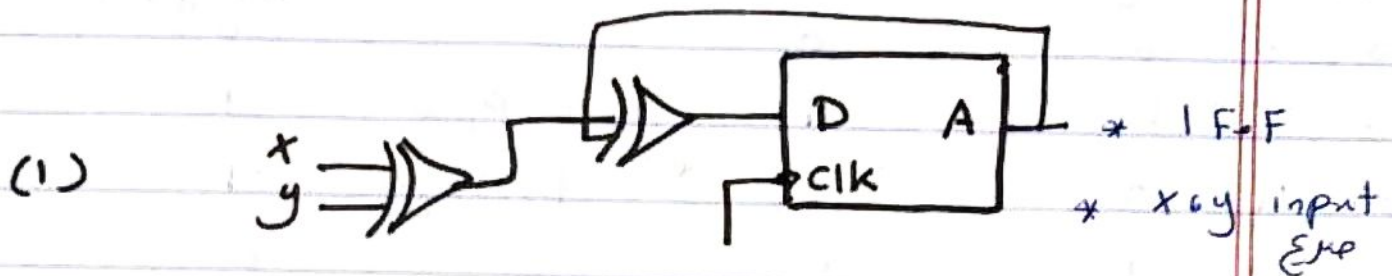
والله بالنظام التي عملت : اول دائرة مثلا 00 و X اول دائما

ما ابي اول clk $\Leftarrow A \ B \ X$ و عليك زي م شرحها

Lecture 22:

في المحاضرة السابقة بنا افل درسنا على 3 انواع F.F

D.F.F و JK.F.F و T.F.F



State Table :

(1)

input

A(t)	x	y
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

output

A(t+1)
0
1
1
0
1
0
0
1

معادلات المدخلات

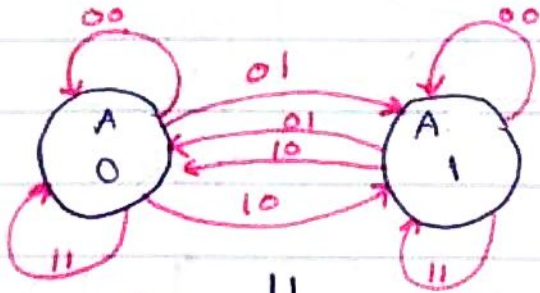
D(A)
0
1
1
0
1
0
0
1

القيم هي نفس D(A)

State diagram :

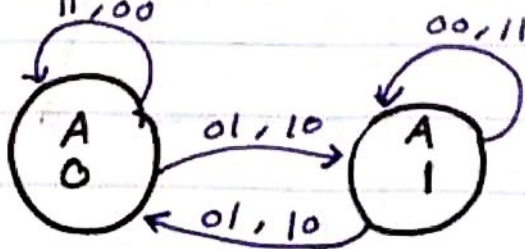
عدد المدخلات 2، 2، 2، 2

ما أعطيت بلاش /
لأنه ما في بالرسمة out



فصّل حلة رسم دائرة
على كوتبة الأرقام دائمة
الثانية عالقة !!
المهم توصل مع

يجوز لي بسبب هذا لانه



شرح صف (1) :

لأن input A = 0

وكان input المعوية 00

وكان نتيجة المخرج A(t) = 0
مع نفس ما عملنا بلاش لأنه ما في
مخرج مربع

بدأ عمل JK حسب الجدول في الشرح

المعادلات

00 ما يتغير
11 يتغير

$$JA = B$$

$$KA = B\bar{x}$$

$$JB = \bar{x}$$

$$KB = A \oplus x$$

State Table :

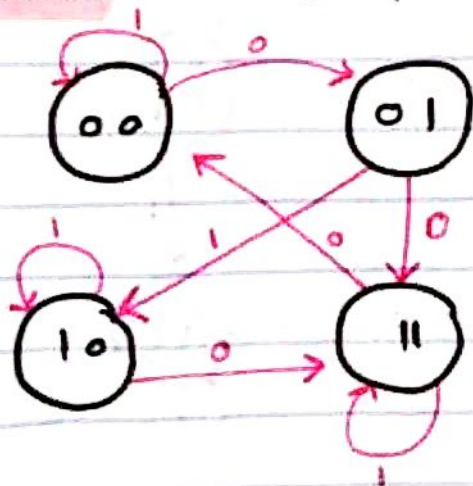
(2)

input			output		معادلات الجانبي			
A(t)	B(t)	x	A(t+1)	B(t+1)	J(A)	K(A)	J(B)	K(B)
0	0	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	1
0	1	0	1	1	1	1	1	0
0	1	1	0	0	1	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	1	0	0	0	0	0
1	1	0	0	0	1	1	1	1
1	1	1	1	1	0	0	0	0

نفس التكملة لـ B

State diagram :

عدد لدوائر 2



Mealy

T-FF مبدأ عمل

0 ← ما يتغير
1 ← يتغير

المعادلات:
 $T_A = X B$
 $T_B = X$

$$y = AB$$

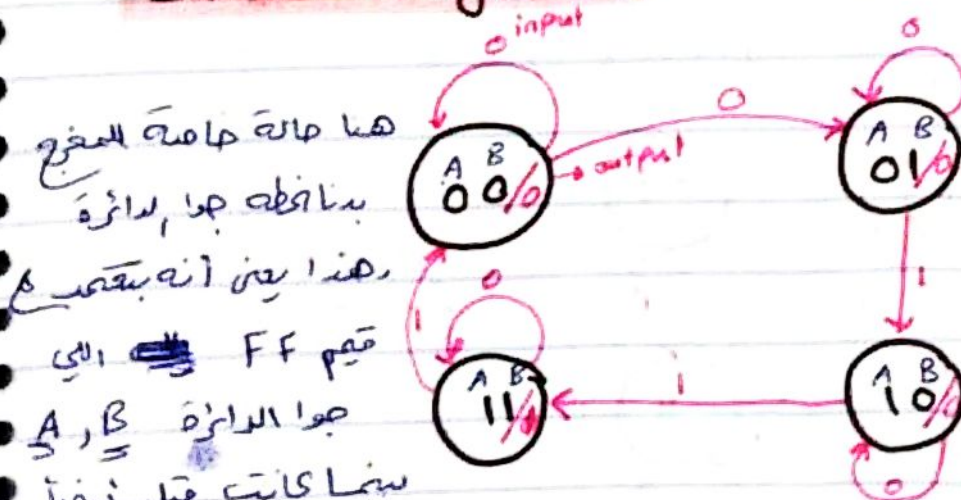
State Table:

(3)

input output معادلات مدبرية

A(t)	B(t)	X	A(t+1)	B(t+1)	y	T(A)	T(B)
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1
0	1	0	0	1	0	0	0
0	1	1	1	0	0	1	1
1	0	0	1	0	0	0	0
1	0	1	1	1	0	0	1
1	1	0	1	1	1	0	0
1	1	1	0	0	1	1	1

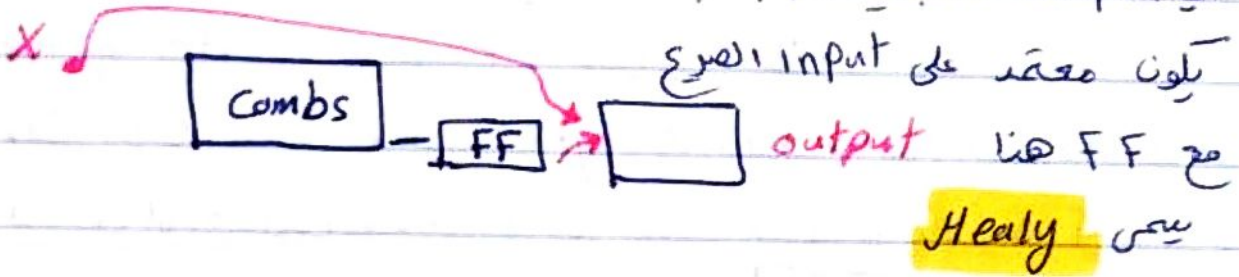
State diagram:



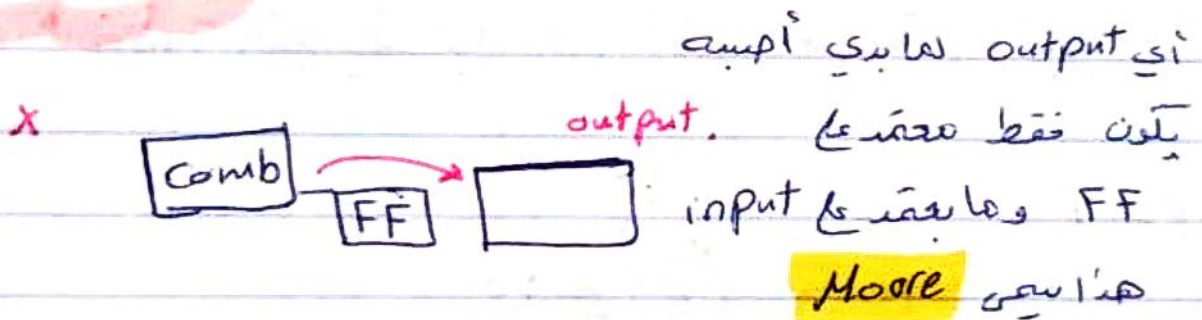
Moore

ملاحظة :-

Mealy :-



Moore :-



Lecture 23 :

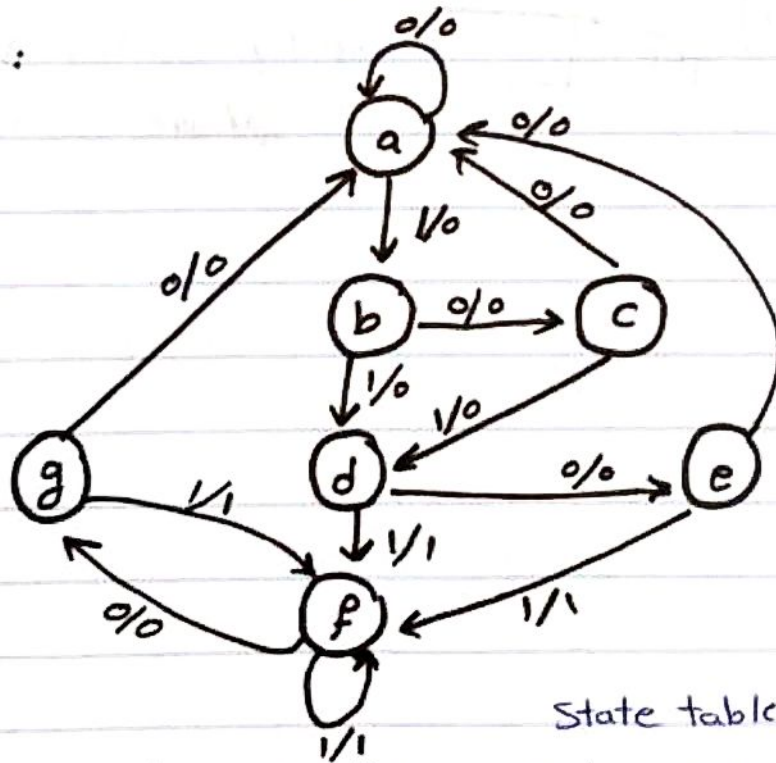
في المحاضرة السابقة كان علينا ال Flip-Flop ، وانا نجيب فيه ال Table و State diagram بطريقة مباشرة ومهلة .

هنا انكس بدو علينا state diagram وانا نجيب فيه ال F-F .

الدوائر التي يكون عليها لها بعد اجيب فيه ال F-F معي P ؟ يعني كانوا مثلا 8 دوائر عد F-F هو 2 ³ 8 ← عد F-F .

كل ما قلنا الدوائر كل ما كان لو هو ال للدائرة اهل واهل .
 " خطوة السبيل قبل الحل خطوة مهمة ورئيسية "

Example :



في نبدأ بتسجيل ال State table

في a : مثلا اذا ابي input ← قيمة 0 رج يخلي a وال output ← 0

في a : انا في a ابي input 1 رج ارجع b وال output ← 0

" عليك دائرة دائرة مرة لعلكون ضروري راجع ورجعها مرة لعلكون واحد مرة راجع ورجعها "

State Table :

Next state
inputs

outputs

X=0

X=1

X=0

X=1

a

a

b

0

0

b

c

d

0

0

c

a

d

0

0

d

e

f

0

1

e

a

f

0

1

~~f~~

g

f

0

1

~~g~~

a

f

0

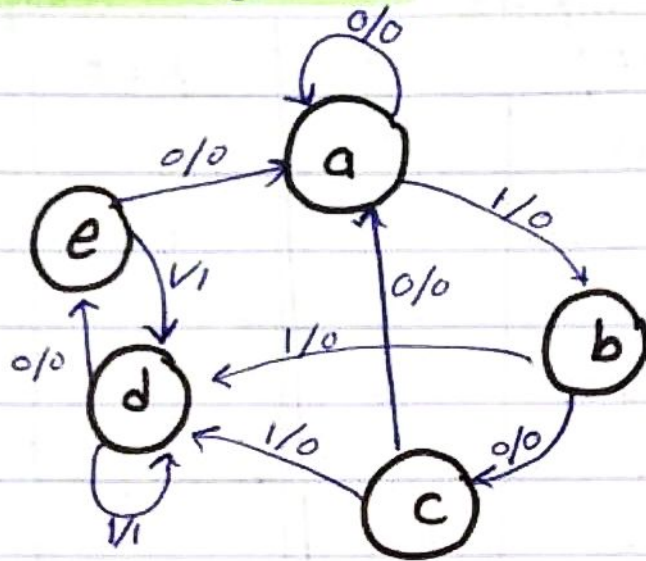
1

← ستوف فيه في ال State Table "2 rows equivalent" بلاقي عندي
مفهوم نفس ال input، ال output ^{متساوية} e, g بلقي واحد
واحد واحد. بلقي مثلاً g وكل g مكددة عنا بقلها e
لأنه أصلاً g ماع متساويات

← هلا بد من حذفنا g دعطينا e بدل g في صف f رح يصير
عنا "2 rows equivalent" أيضاً بعل نفس الطريقة وبلقي واحد
d, f نفس الاشئ ما الحمد لله الدكتور لقي f لاني ما
كنت ح الخ اللفي حرف اسمي

← ما في اي تشابهات أخرى بين الصفوف و هيك تكون أبسط
اشئ ما هلا نرجع نرسم ال State diagram مرة ثانية

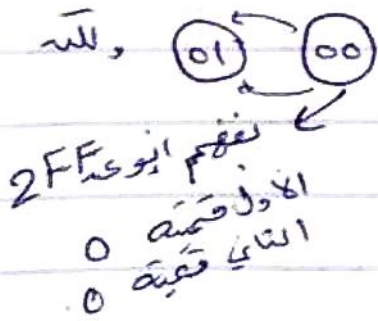
State diagram :



← احنا شو علينا؟؟ ببساطة ال State diagram بدل هذي 7 دوائر هارت 5
 أصبحت أسهل وأبسط وأسهل

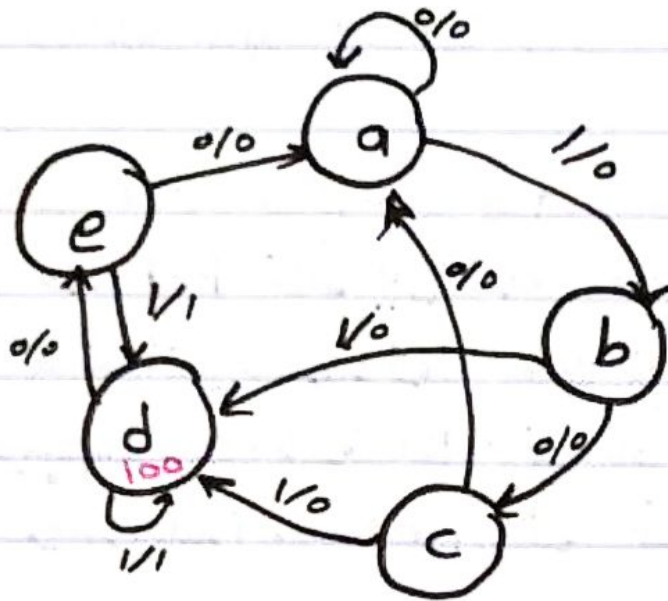
⇒ state assignment ⇐

← قبل في ال state diagram كان في الدوائر values "قيم"
 حالياً صار مكتوب بالأحرف ، عوضين diagram
 بطريقة عكسية انا بي اعرف شو القيم الي بالدوائر
 علشان ارسوم الحركة الأصلية بال F-F
 انا معسرة ابي لأشوف قيم وليس أحرف .
 ففي حاجة اسمها state assignment وهي ابدال الأحرف ب قيم بطريقة ما
 حقا أدخل للسيرة .
 طب أنا على أي أساس أعدد؟؟ الكتاب معقد 3 طرق بدينا نعتد عليهم .



① أول طريقة: استخدام طريقة "Binary numbers"

- كم دائرة عنا؟؟ 5 دوائر " 5 بال Binary كم بت بالزما
لضللها؟؟ " 3 بتات كافية $2^3 = 8$ احتمالات



000
001
010
011
100
101
110
111

طب أنا ضللت ل د أي رقم افتار؟؟ أي رقم يربط اختياره 8 احتمالات
في واعي منهم الدوائر. " عالجهم رج أعط ل د رقم 4 " ن
ولله للأسف في مساقنا لازم نعتقد كلنا حل واحد وإجابة واحدة يعني
a بتاخذ 0 b بتاخذ 1 c بتاخذ 2 وهكذا " بالترتيب "
ولله طريقة اكل الادب هيبة 100% فقط

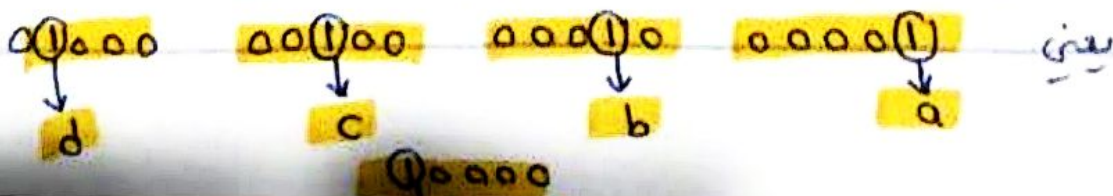
نعتقد اكل الشاي لتو عهد لحلول ..

② ثاني طريقة: "Gray assignment"

⇨ بدل من تكتب الأرقام بالترتيب بال Binary بتلبيهم بال Gray code
بأخذ أول وحدة ل a والثاني ل b وهكذا .. مثل الطريقة
السابقة.

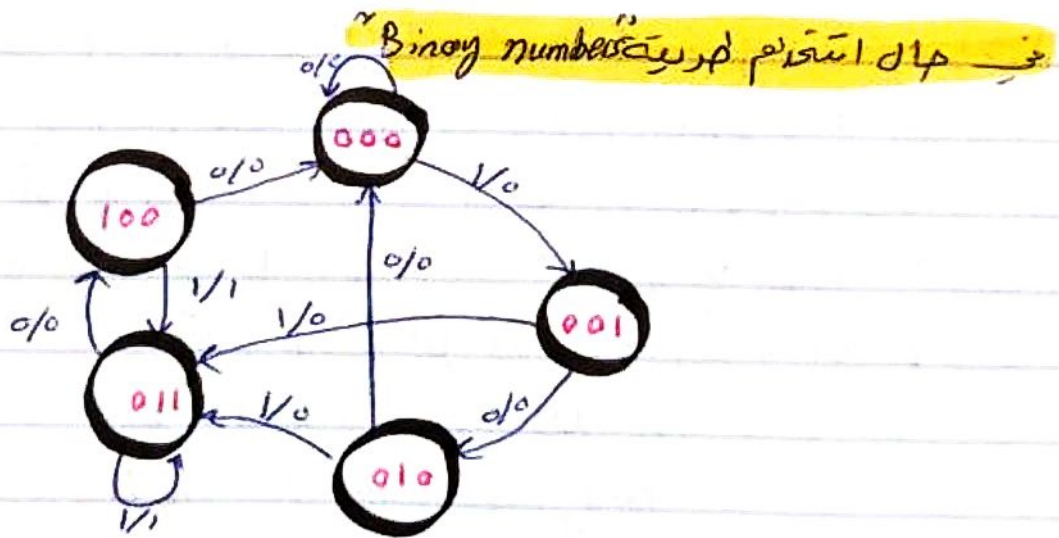
③ ثالث طريقة: "One hot assignment"

عدد الخانات م ب عدد الدوائر □□□□□ عنى 5 دوائر يعني 5 دوائر
هم كيف رح اعرف مسد a, b, c؟ كلهم يكونوا أصفار عدا خانة واحدة ①



← احنا عرسد أي طريقة نستخدم كل هذا القول ، والله لو جدد في الاختبار
 أي طريقة فلانم نلتزم بطريقة معينة "التي روح تكون مطلوبة"
 يعني لازم نعرف الـ 3 حرفه . الأسهل استخدم طريقة **Binary**
 في حالة ما جدد في الاختبار .

← ملاحظة : بلا حظ في طريقة **on-hot-ss** لما آجي اصلها **00001**
 ← عدد **F-F** قاعد يكرت عندي يعني عندي **5 F-F** فالبعض
 يعتقد انو ممكنه الدائرة تصعب فالكلام عنى منطق قد يكون **5 F-F**
 ريسيلج وسط الدائرة فلينا ، في حال ممكنه اطلعها الـ **Binary** رتوبه **3 F-F**
 ومنه المثل ان تصعب الدائرة فحدا ليسى متباين



Lecture 24 : part 1

كيف نرسم Sequential circuit
بناءً على

7 خطوات :

* The procedure for designing synchronous sequential circuits can be summarized by a list of recommended steps :

① From the word description and specifications of the desired operation , derive a state diagram for the circuit.
الخطوة 1 : من الوصف اللفظي والبيانات المطلوبة لعملية معينة ، نشتق " State diagram " أو " مخطط الحالة " .

② Reduce the number of state if necessary
تقليل عدد الحالات إذا لزم الأمر .

③ Assign binary values to the state .

④ Obtain the binary-coded state table .

⑤ choose the type of Flip-Flops to be used .

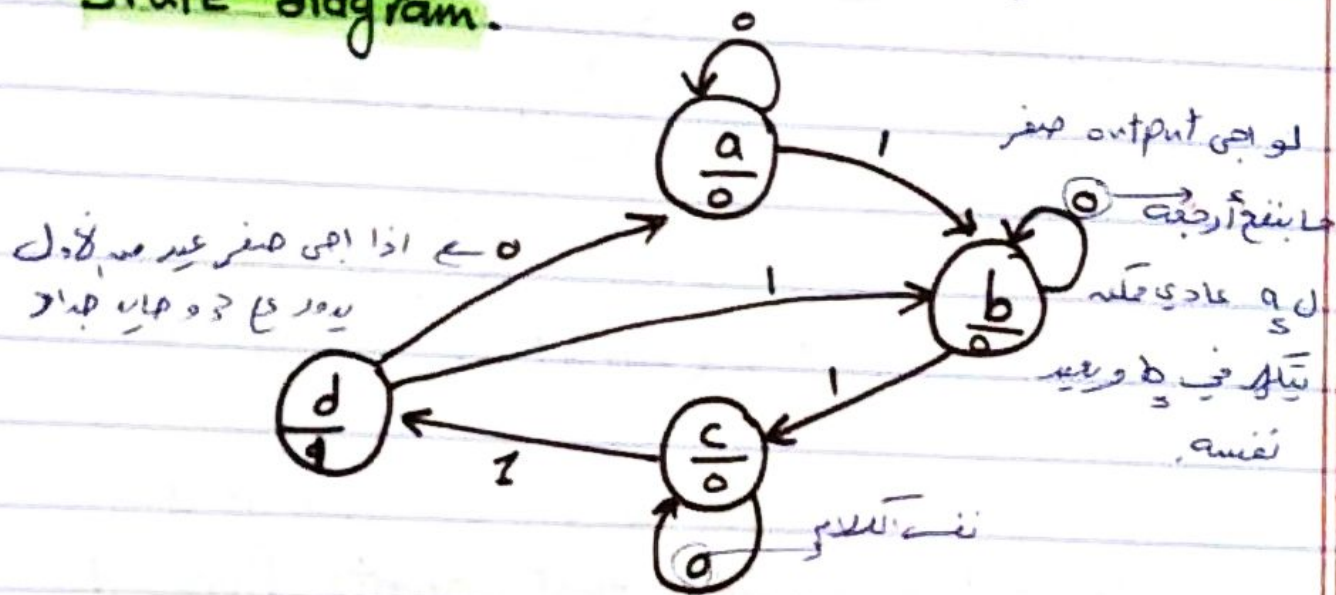
⑥ Derive the simplified Flip-Flops input equations and output equations .

⑦ Draw the logic diagram .

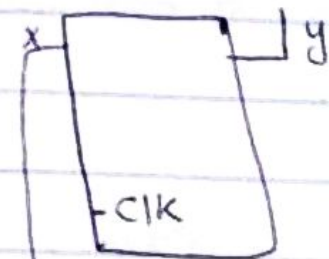
يعني في النظام السابق
 $\begin{matrix} \text{111} & \text{111} & 0 & 11 & \text{111} \\ \hline 100 & 100 & 000 & 100 \end{matrix}$

مثال (٣): نفس فكرة الأمثلة السابقة ولكن لا يوجد اشتراك ال 3 و 4
 يكونوا متتابعين (ممنوع ينبغي 1 مرين وراءه أو متتابعين بمجرد يقفوا
 ال 1 الثاني وال 3 الثالث) النظام بطول 1

State diagram.



الفكرة تلك اختصار : النظام لم يبدل
 عليه أول واحد يحفظ عنده وراها لو
 ستوما احي عادي ما بننا الوهم



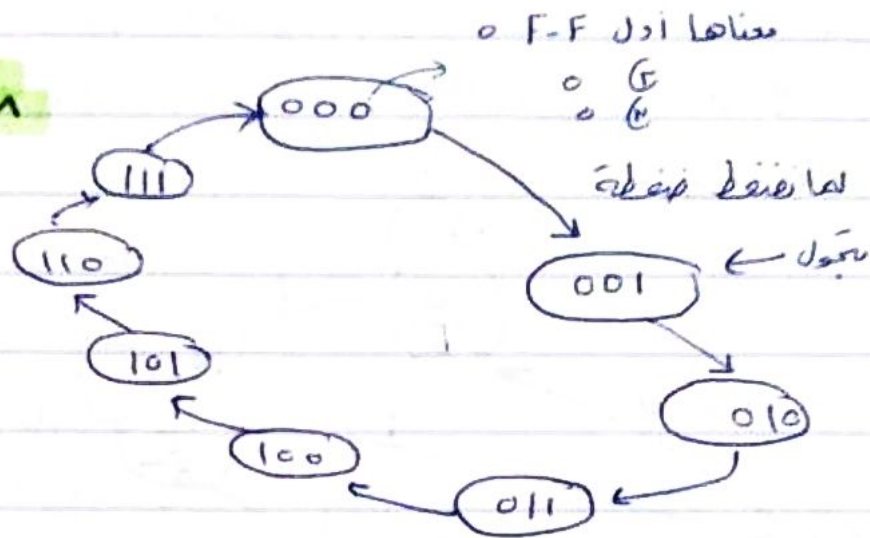
متتابعين عادي ينبغي صفر
 $\begin{matrix} 1001011011001001 \\ \hline 1000001000100000 \end{matrix}$

وراهما ينبغي واحد يحفظ
 الواحد الثاني ويقل مستدكر
 ما بنمهر 00 لها يفرجها
 رينا و ينبغي الواحد الثالث
 النظام بطولنا 1
 فهموا 99 00

مثال (4): تصميم عدد بـ 3 بتات من 0 إلى 7

Design an ascending counter that counts from 0 to 7.

State diagram

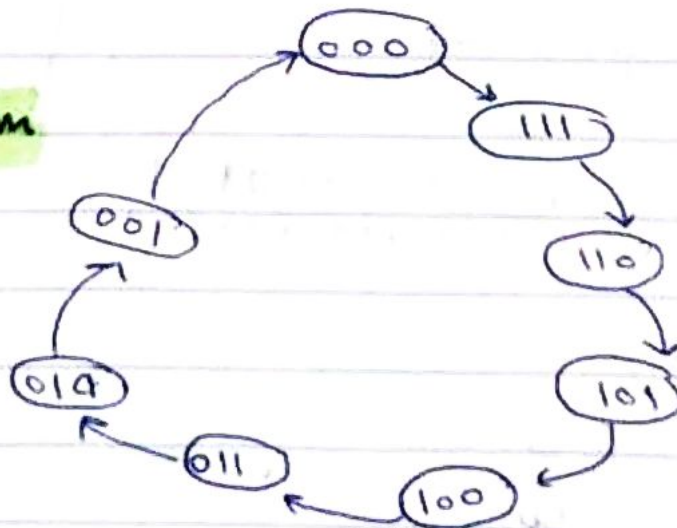


م أجدله إسوال !!

مثال (5): تصميم عدد بـ 3 بتات من 7 إلى 0

Design a descending counter that counts from 7 to 0.

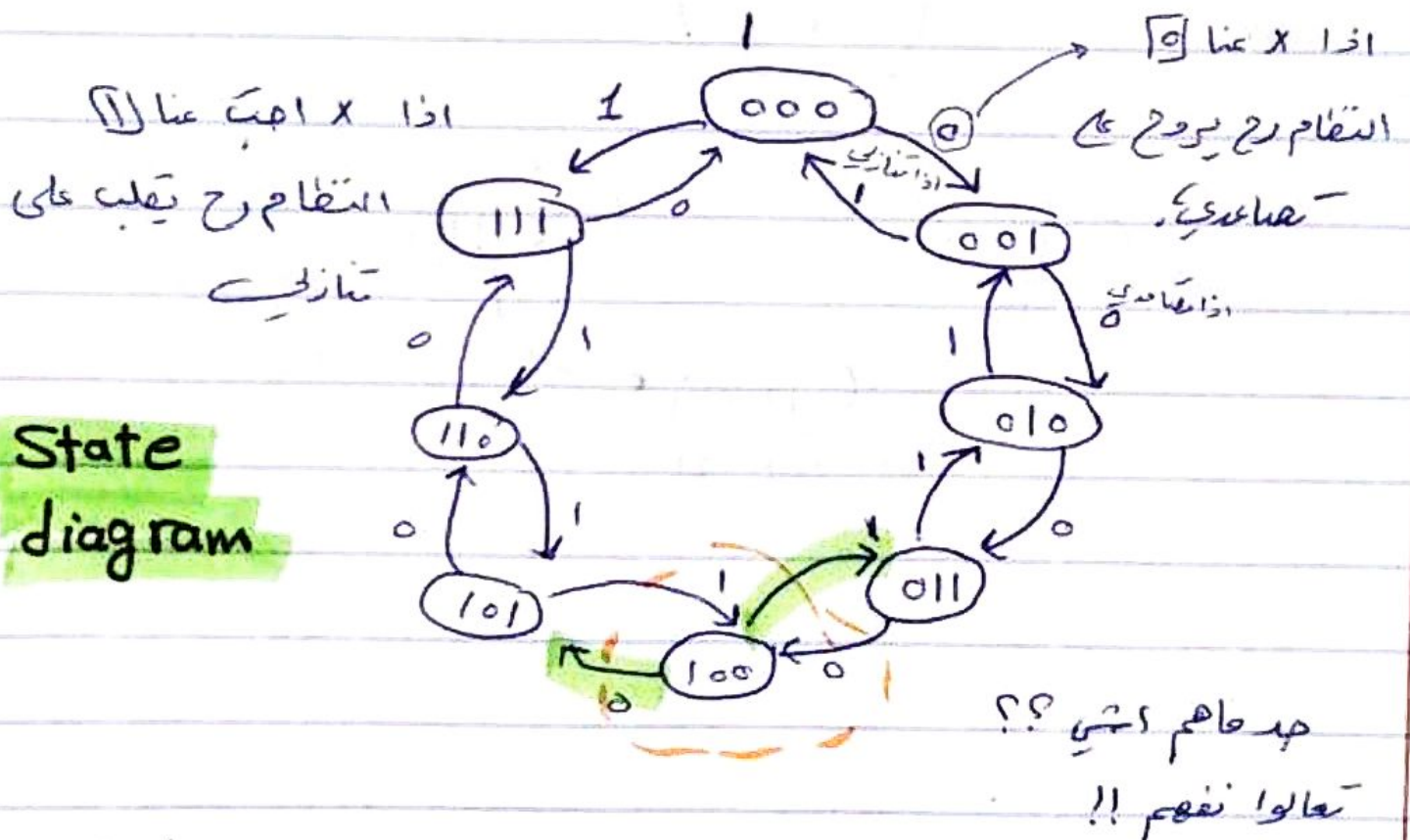
State diagram



مثال (6): تصميم عداد بعد ثنائي اوتناري بناءً على قيمة x
 اذا $x=0$ بعد ثنائي و اذا $x=1$ بعد تنازلياً.

⇒ Design an up/down counter controlled by input x :

- If $x=0$, the counter counts upward (ascending).
- If $x=1$, the counter counts downward (descending).



تعالوا نفهم !!
 هذا ناخذ رقم 4 اذا كانت $x=0$ حيثقل ل 5 فعلاً هيه
 راج ل 5 لما x صفر 6 واذا كانت $x=1$ رج يروح ل 3
 فنسبه على التسليم وسه وايه وسه وسه هالوجه :

مثال (7) : تصميم دائرة لاسمارة ضوئية " الأحمر صفوي كمال، التحريم الأصفر، الأخضر
أصفر "

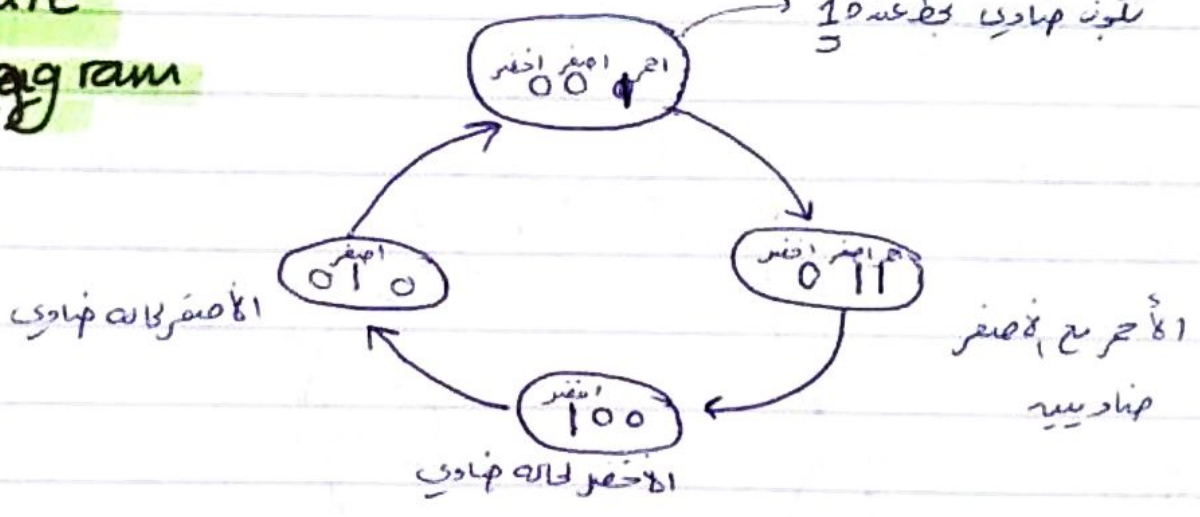
⇒ Design a sequential circuit for a traffic light with the following sequence :

Red only → Red and yellow only → green only → Yellow only , then repeats :

اول ما تبدأ الاسمارة الضوئية الأحمر

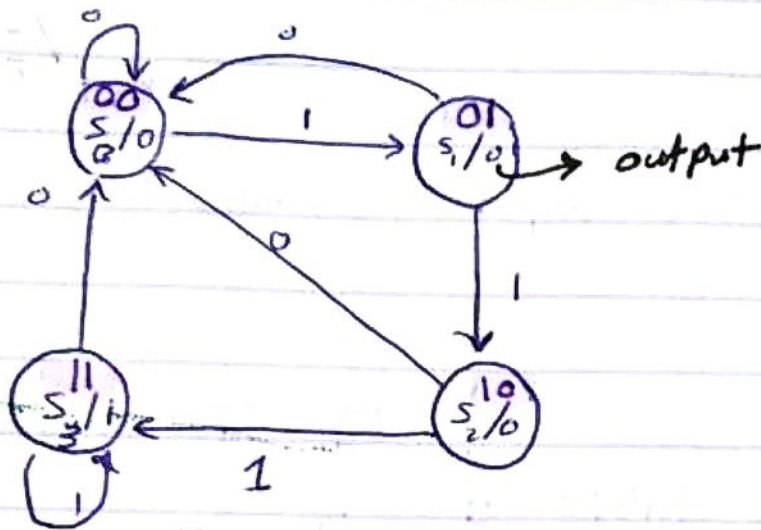
تكون ضاوي بطول 10

State diagram



Part 2 :

هذه رح تستغل على الخطوات من 4-7



بناءً على ما في الخطوات الحظوظ الأولى كانت من رسم ال State diagram

الخطوة التالية بنا فعل **State reduction** بالمدى الطريقة التالية التي

أخبرناهم [One hot assign - Gray assignment **Binary**]

ال State diagram التي معنا فافيد **State reduction** (تبسيط) آخر مرة
هيك ولأن المصروف نتأكد إذا هو أبسط صورة ولا لا

* الخطوة التي بعدها ال **State assignment** هو ما طلب بالأسفل أي طريقة

نستخدم (نستخدم أي طريقة بها ياها) **عدد f-f**

تم **f-f** بالرمز $2 \leftarrow 2 = 4 \leftarrow$ عدد الدوائر

بناظرنا 2 بت
input Present State Next State
Alt) B(t) X Alt+1) B(t+1) y **Stat table**

0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	0
1	0	1	1	0
1	0	0	0	0
1	1	1	1	1
1	1	0	0	1

أولاً أنا لسا بالدائرة
أكادف ال output
حفظ عليها وليس
لكن الدائرة الجيدة

output صريح ؟



ملاحظة :

ملاحظات بعد ال state 00

output 100

الخطوة التالية : ادرم اى نوع من flip-flop بيدى ارحم فيه
D.F.F
T.F.F
JK.F.F

أدلة : الحل ب D.F.F

State table :

$A(t)$	$B(t)$	X	$A(t+1)$	$B(t+1)$	y	D_A	D_B
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1
0	1	0	0	0	0	0	0
0	1	1	1	0	0	1	0
1	0	1	1	1	0	1	1
1	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
1	1	0	0	0	1	0	0

النتيجة النهائية : دراسة ب 2 FF في النظر على نوعهم



علیہ اذکر فی تقابل تبیین الحروف حسب الطلب و یسر
 معادلات D_A و D_B و معادلات ال output سریع
 نیز معادلات D_A و D_B k-map

D_A

	Bx	$\bar{B}x$	$\bar{B}x$	Bx	$B\bar{x}$
$\bar{A}$	0	0	1	0	
A	0	1	1	0	

→ Bx

← Ax

$$D_A = Ax + Bx$$

D_B

	Bx	$\bar{B}x$	$\bar{B}x$	Bx	$B\bar{x}$
$\bar{A}$	0	1	0	0	
A	0	1	1	0	

← $\bar{B}x$

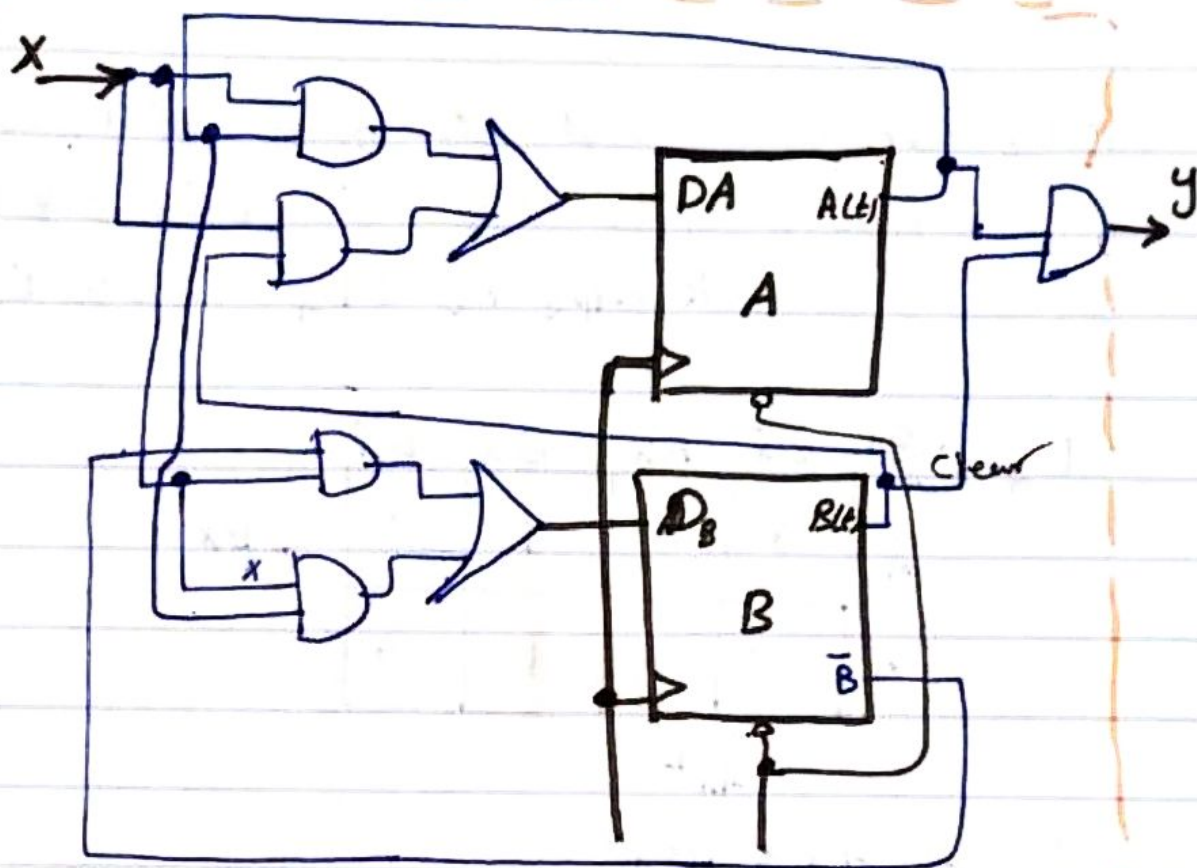
← Ax

$$D_B = Ax + \bar{B}x$$

	Bx	$\bar{B}x$	$\bar{B}x$	Bx	$B\bar{x}$
$\bar{A}$	0	0	0	0	
A	0	0	1	1	

$$y = AB$$

Seq
circuit



النظام مارج - شغل اللبنة إلا إذا أجي 3 وحدات متعاقبين 111
بويحي 1 .

ملاحظات هامة : بعد من نفل State diagram لازم يكون عنا قدرة لشغل من
اعرف لم P-P ثم input ثم output صريح . لازم امل تصور للمرجع
واحد نفع P+ .

إذا 0 تظل
No change
إذا 1 تظل 1
Change

إذا 1 1
زى بعض
0

present state

ثانياً: الـ T-ff

$A(t) \cdot B(t)$	x	$A(t+1)$	$B(t+1)$	y	T_A	T_B
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	1
0	1	1	0	0	1	0
1	0	0	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	0	1	1
1	1	1	1	1	0	0

No change

في آخر row : الحالة السابقة في A 0 ، الحالة الجديدة لـ A 0

No change

!! k-map

لـ k-map

T_A

Bx	$\bar{B}x$	$\bar{B}x$	Bx	Bx
$\bar{A}$	0	0	1	0
A	1	0	0	1

$$T_A = A\bar{x} + \bar{A}Bx$$

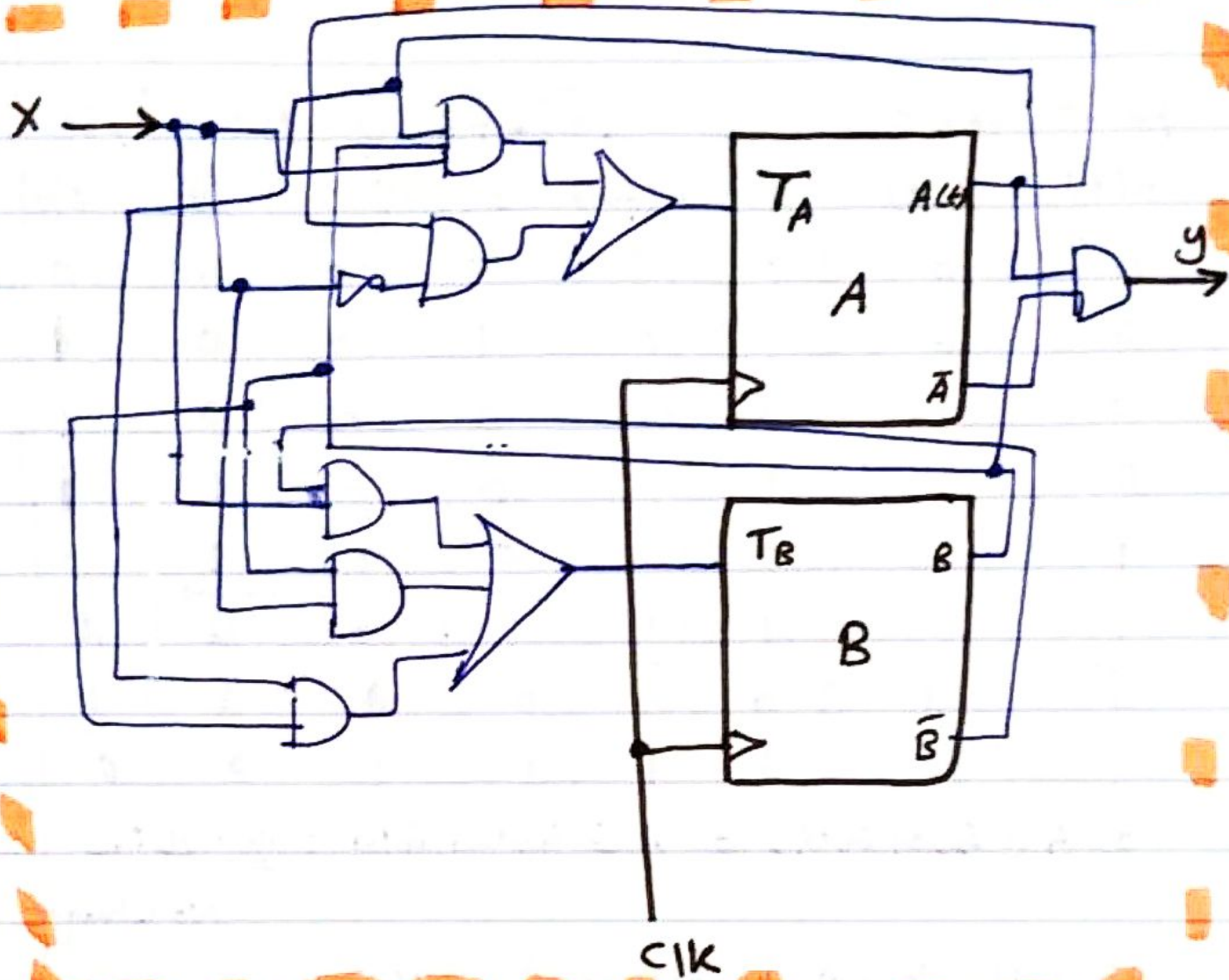
T_B

Bx	$\bar{B}x$	$\bar{B}x$	Bx	Bx
$\bar{A}$	0	1	1	1
A	0	1	0	1

$$T_B = \bar{B}x + Bx + \bar{A}B$$

y	Bx	$\bar{B}x$	$\bar{B}x$	Bx	Bx
$\bar{A}$	0	0	0	0	0
A	0	0	1	1	1

$$y = AB$$



طالبا : اكل ب ك ج

$A(t)$	$B(t)$	X	$A(t+1)$	$B(t+1)$	y	J_A	K_A	J_B	K_B
0	0	0	0	0	0	0	X	0	X
0	0	1	0	1	0	0	X	1	X
0	1	0	0	0	0	0	X	X	1
0	1	1	1	0	0	1	X	X	1
1	0	0	0	0	0	X	1	0	X
1	0	1	1	1	0	X	0	1	X
1	1	0	0	0	1	X	1	X	1
1	1	1	1	1	1	X	0	X	0

جدول لل JK

0 → 0	0	X
0 → 1	1	X
1 → 0	X	1
1 → 1	X	0

في المحاضرة
القائمة سيتم
اشتقاقه.

J_A

	Bx	$\bar{B}\bar{x}$	$\bar{B}x$	Bx	$B\bar{x}$
$\bar{A}$	0	0	1	0	
A	X	X	X	X	

K_A

X	X	X	X
1	0	0	1

$J_A = Bx$

$K_A = \bar{x}$

J_B

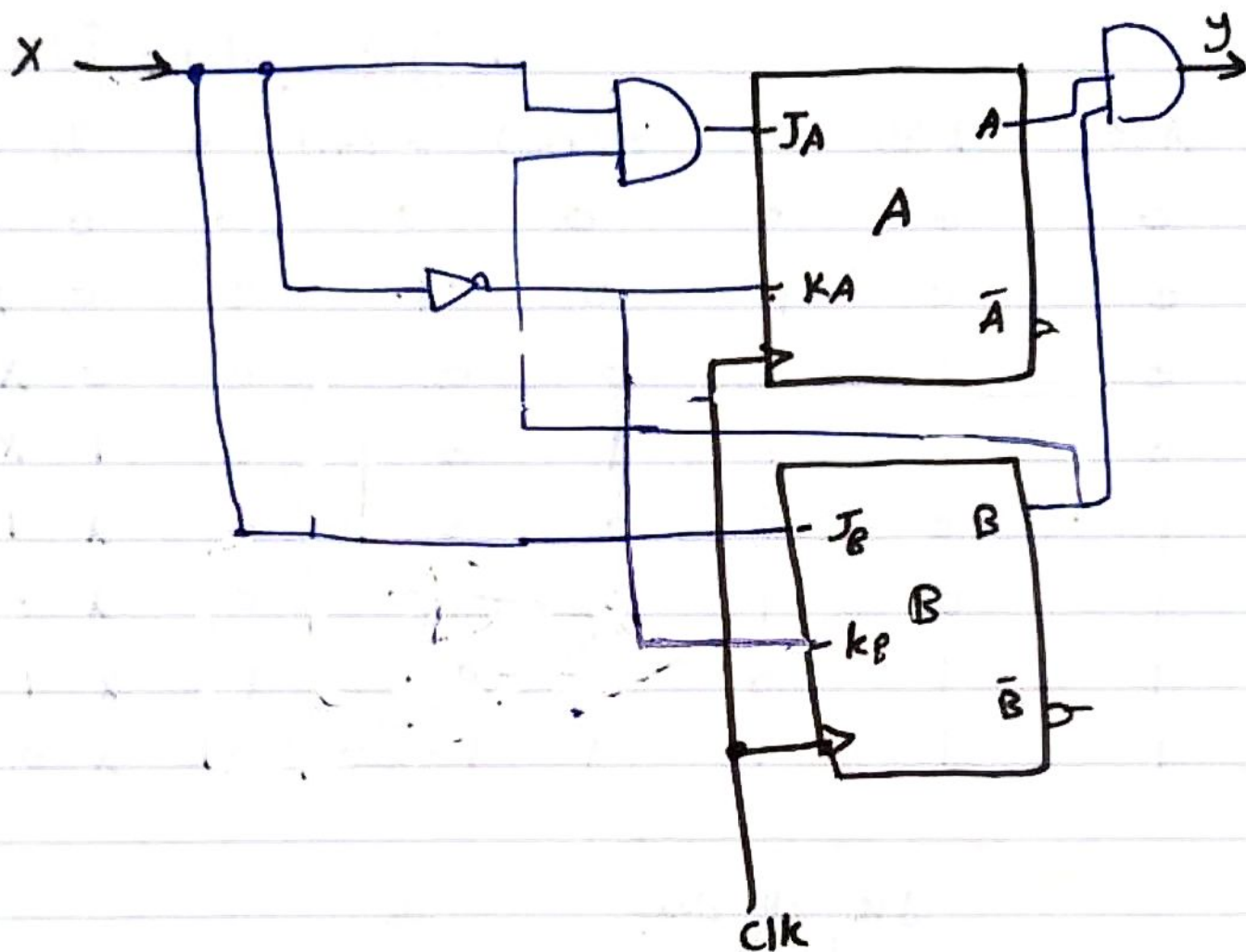
	$\bar{B}\bar{x}$	$\bar{B}x$	Bx	$B\bar{x}$
$\bar{A}$	0	1	X	X
A	0	1	X	X

K_B

	$\bar{B}\bar{x}$	$\bar{B}x$	Bx	$B\bar{x}$
$\bar{A}$	X	X	1	1
A	X	X	0	1

$J_B = x$

$K_B = \bar{x}$



Chapter 6 : Registers and Counters

Lecture 25

بينا شرحنا في السبحة 24 و نسبق الجدول التالي قبلنا عنه
الاساس

J	k
0	0
0	1
1	0
1	1

0 → 0

No change (0) لا يتغير K لا يتغير

Reset (0) لا يتغير J لا يتغير

Set (1) "No Change" لا يتغير

Change

0 → 1 ②

1 1 change

1 0 set

1 X

J k 0 → 0 ①

0 0 No change

0 1 Reset

0 X

③ 1 → 0

1 1 change

0 1 reset

X 1

④ 1 → 1

0 0 No change

1 0 set

X 0

لورفت افترفت FF من عدي :

x	y	
0	0	0 → reset
0	1	Change
1	0	No Change
1	1	??

Characteristics Table.

مطلوب ما "Execution table" تغيير

$Q(t)$	$Q(t+1)$	$\odot y$
0	0	X 0
0	1	0 1
1	0	0 X
1	1	1 0

No change
1 → 1
set

x	y
1	0
1	0

اول موجود
reset
no change
0 → 0
X 0

change
set
0 → 1

change
reset
1 → 0

x	y
0	1
0	0
0	0
0	X

مع السلكة يا جابر 5 !! ارحمنا :
ما تعدي :

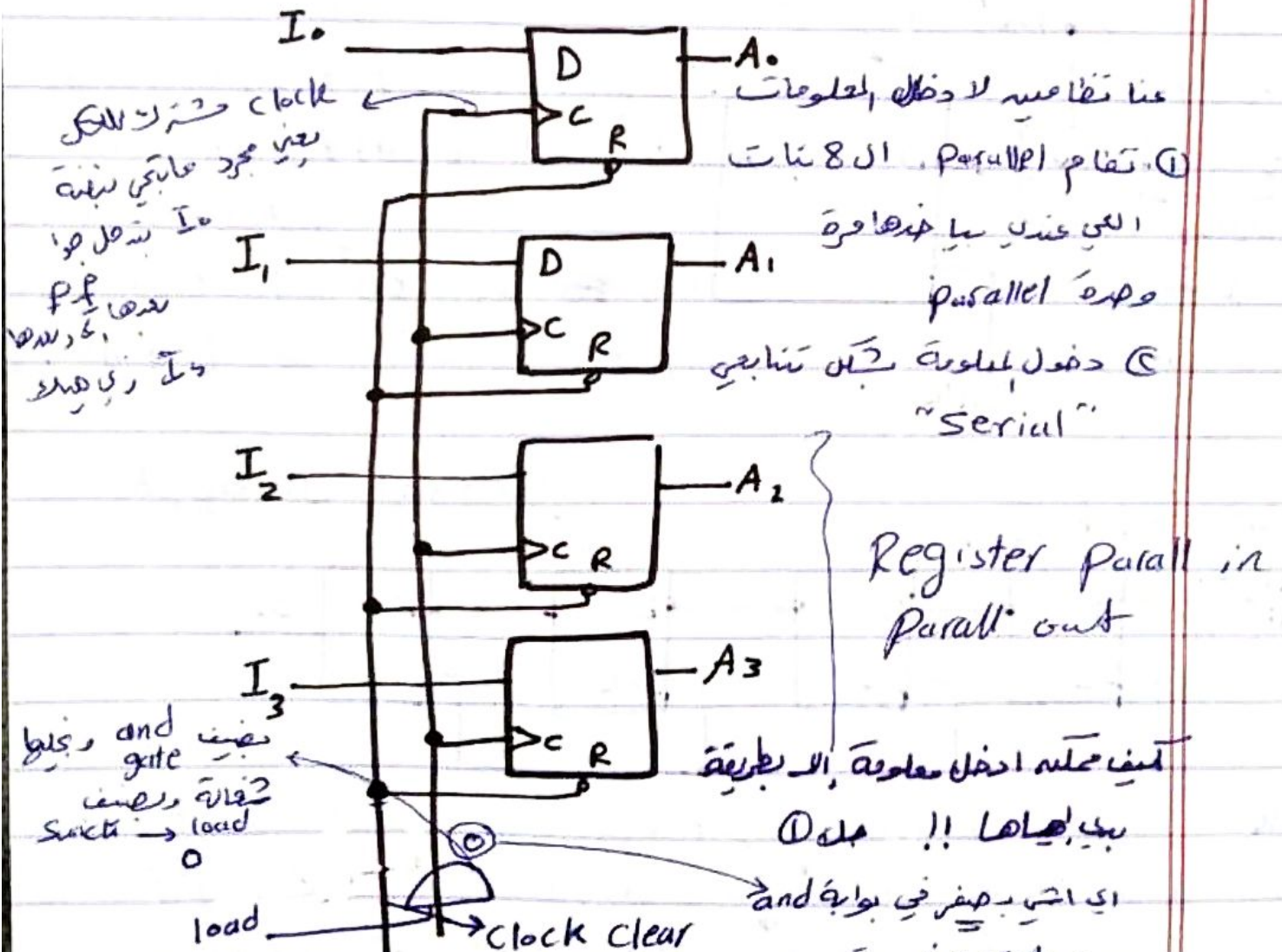
Registers and counters

في مبدأ آخر جابر !!

Registers : هي دوائر تستخدم لتخزين البيانات مؤقتاً .

مثلاً تخزين رقم ثنائي (Binary number)

الرجستر مثلاً كانه يتكون من 4 بت يعني بس اجزاه 4 بت 1 0 1 0
دستوره غالباً من محركات $f-f$ كل $f-f$ وحده اجزاه 4 بت وحده
بعض حافظه كده ما يتغير الكود او تدخل بيانات جديده .

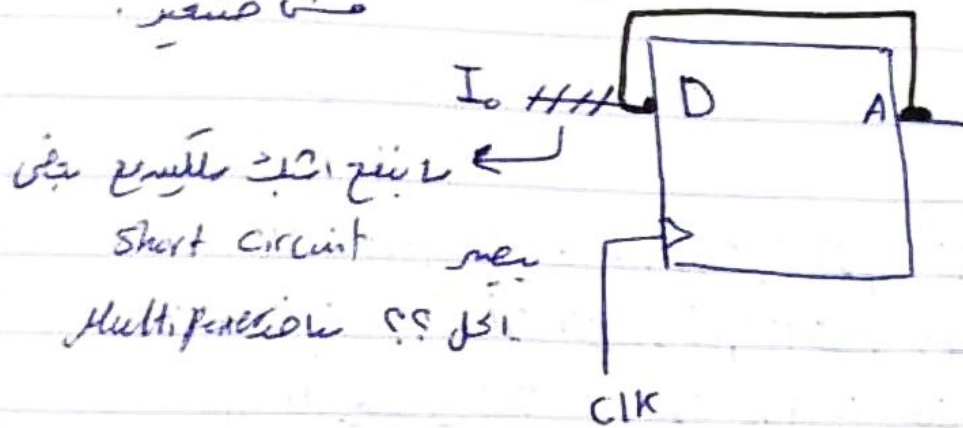


صيعلي غير في كل R داتاي بغير $f-f$

لجميع [الكود منه لو كود مثلاً معلومه من

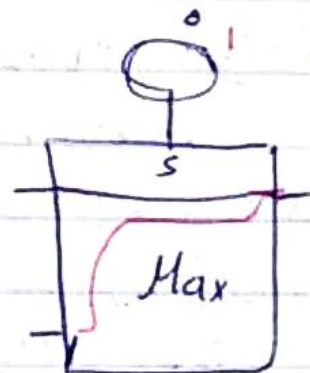
كاتب ويريد اغيرها على منظمه اغيرها
في سلك رجستر الكل [ويعني رجستر 1
منظمه

لدراسة حالت توصيلة ثمانية
من صغير

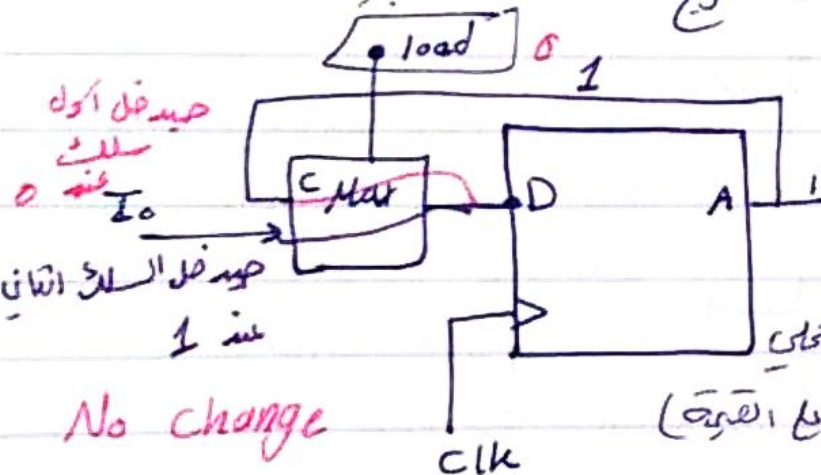


حل 2:

التيقن انك ملتصق
Short circuit
الكل؟؟ متاخره Multiplexer



ال Max فختار احد مدخله على انه يطلع فالمخرج Max نفسه على انه احدث نوعي
بيج

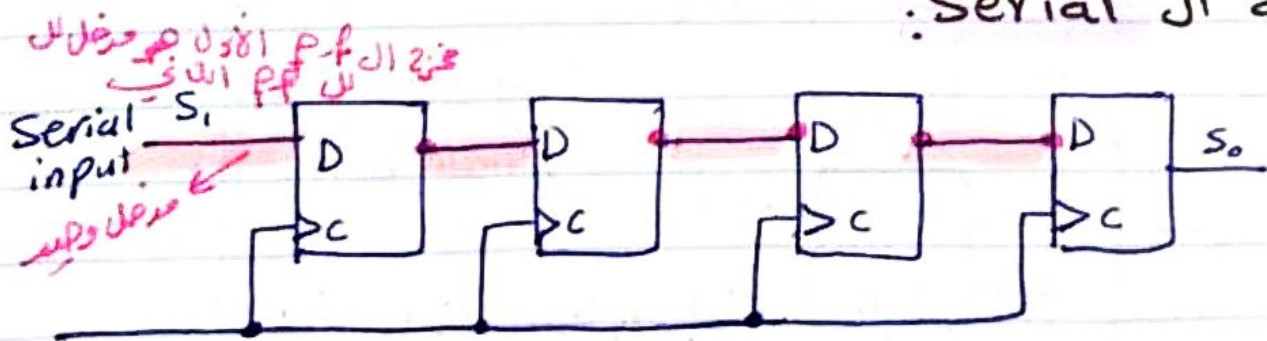


في هاد الكلام P-F واحد بدينا فيه
هاد الكلام ال P-F كن واحد
ال Max

طالما حاطيه load بغير يعني
ال P-F قرا عنه نفسه (بما حفظه القيمة)

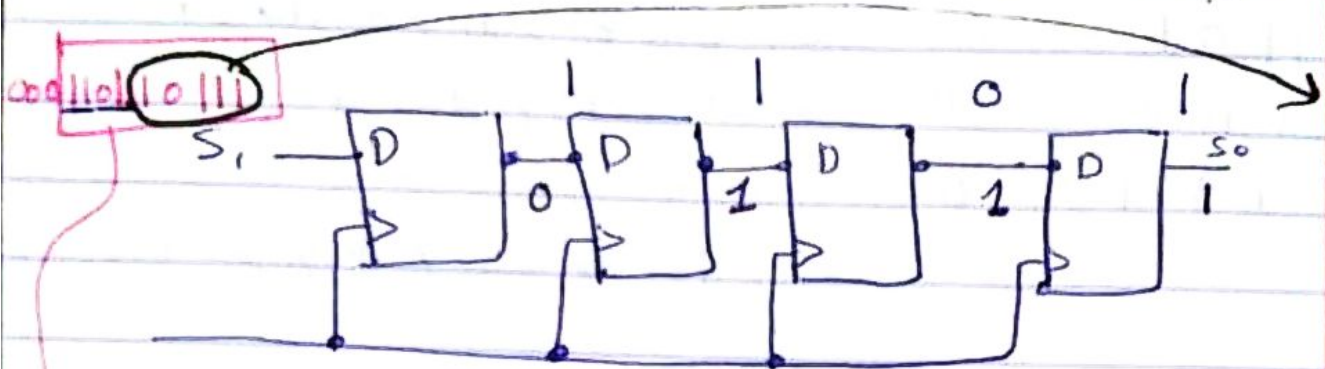
No change

طريقة ال Serial:



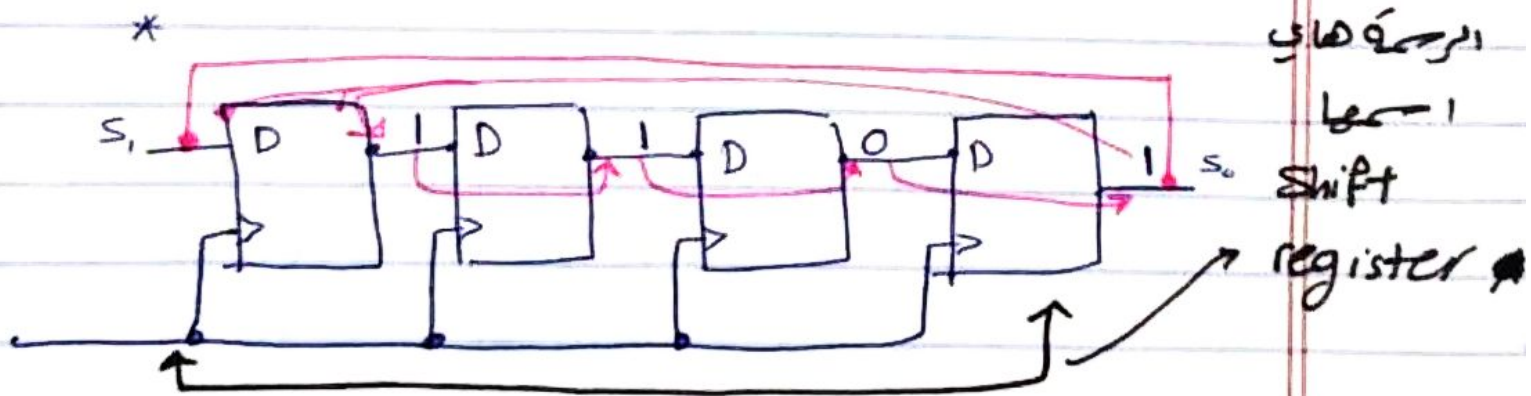
لنفترض اننا عند S1 القيمة هتغير عند اول نبضة تطلع من CLK
P-F الاول هتنتقل لمدخل ال S1 فدها يكون مثلا 1 بيخده نفسه
القيمة هتنتقل لمدخل ال S1 فدها يكون مثلا 1 بيخده نفسه
المعلومات مخزنة بـ "Serially"
ولما ازل اأهبط مثلا يكونه عنا هتغير
اول اسبي لما يوصل S0 بيروح حلفه
ما يضل

مع لو بي اترف جم المعلومة بخزينة بيط لمبة مع ال output
ويصنف 4 صفحات على ٨ اطلع كل الصفحة

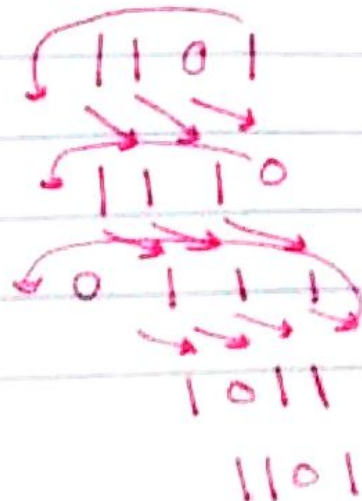


مثلاً کاغذ پر ۵۱۱ و بی ایضاً ۹ صفحات سوالی طرح استوفه
به اجلوات ۹۹

أذن ٩ منقطات المعلومات التي رح استوفينا ١١ ٥ ١
طلب ديباقي ؟؟ خلصا بلعلوا براا ماإي هقل منهم .
يعني آخر لا بي التي رح استوفهم وبى آخر لا بي ؟ لا توجد
ال P P ٤



Circular Shift Register.



- بعد حفظه
- 1 ملا حظہ : کل 4 صفحات
 - 2 ترجیع زری ماکانتہ نہ
 $f-f$ و f
 - 3 دائماً اشتغل باین لیسما
 - 4

كَيْ يَغْنِي لَوْ سَأَلَ بَعْدَ 22 مَرَّةً كَيْفَ رَجَعَتْ لَوْنَهُ ؟
وَكَيْفَ سَيَأْتِي بَعْدَ 3 حَفَظَاتٍ مَوْضِعَهُ ؟

1 1 0 1

1 1 1 0

0 1 1 1

①

②

أَمَّا إِثْرُهَا لَوْ عَلِمْنَا مَرَّةً عَادِي بِقَسَمٍ

عَلَى 4 وَبِشَقْلٍ عَلَى 5 بَاقِي الصَّفْحَةِ

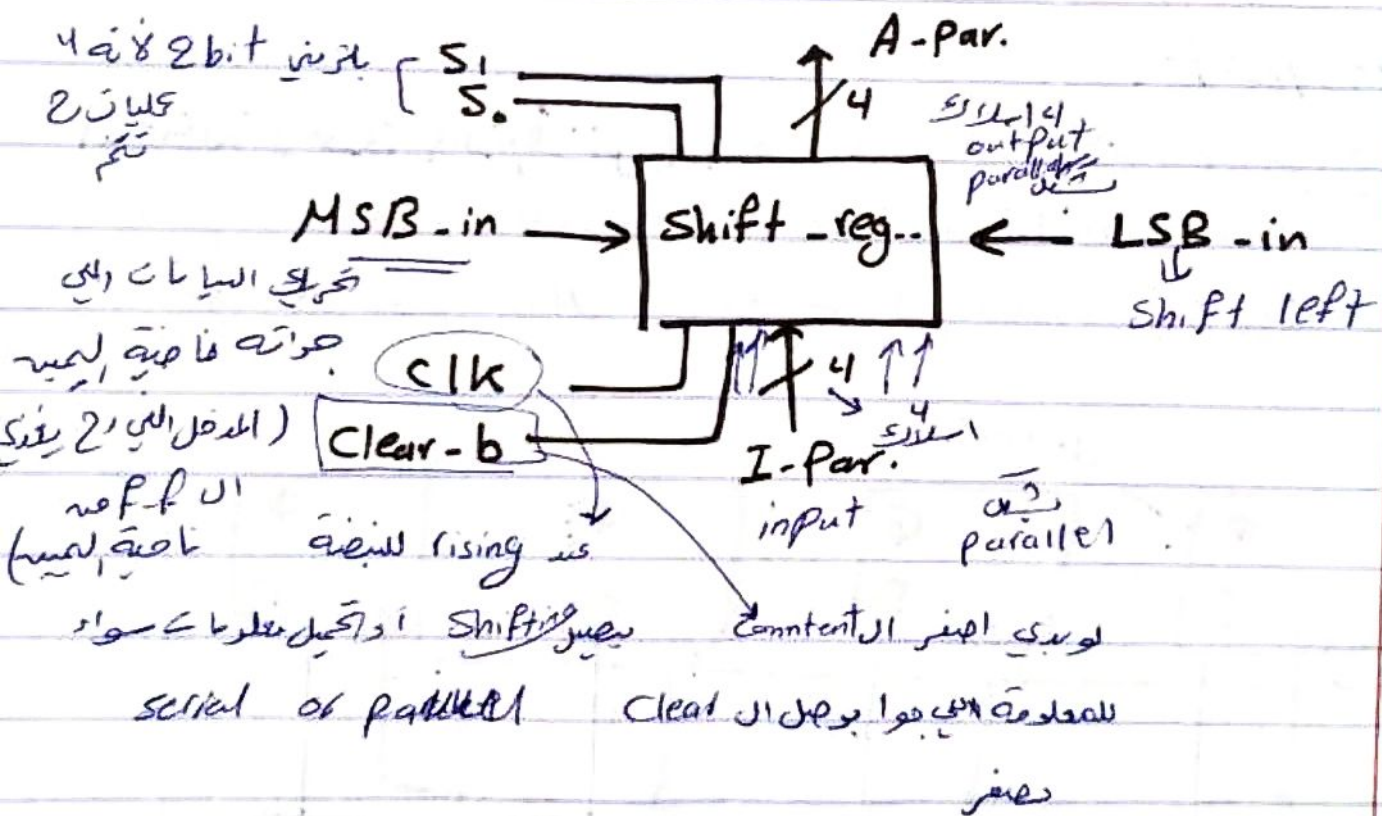
وَأَنَا بَقِيتُ عَلَى 4 عَلَى 5 بَعْدَ عَدَدِ f-f .

Lecture 26 :

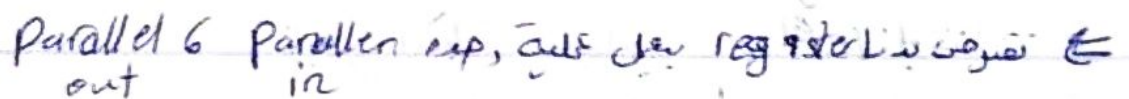
Universal Shift register.

هو سجل رقمي يعمل بـ 4 وظائف رئيسية :

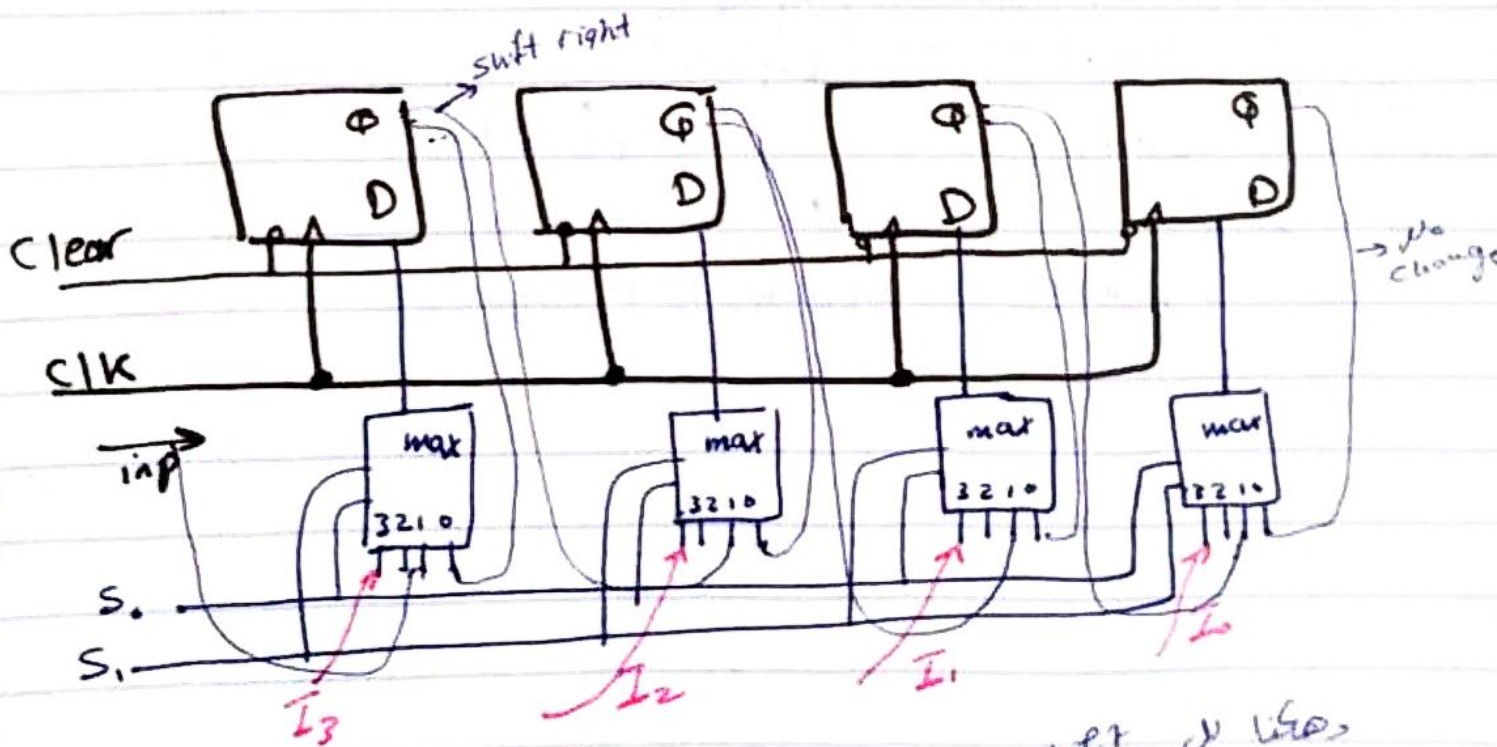
- ① Hold : تحتفظ بالقيمة دون تغيير
- ② Shift Right : ينزح البتات لليمين
- ③ shift left : ينزح البتات لليسار
- ④ parallel load : يحمل كل البتات مرة واحدة



بہ اطلاق
constant



- المشكلة (محددة للفرمان الأخضر) : في 4 احتمالات فقط واهم مفهوم يوضح على D ما ينفج انفسك ال 4 املاك واهم مفهوم ينفج (short circuit) الكل؟؟ بخط Max ينظم العملية. (يا اراحه يا لاهيت تعيس الخ...)



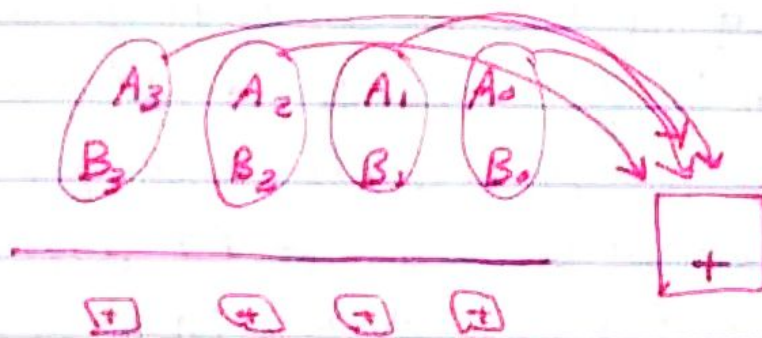
دھکا مل Shift
left

	S_1	S_0	
1	0	0	No change
	0	1	Shift right
	1	0	Shift left
	1	1	load

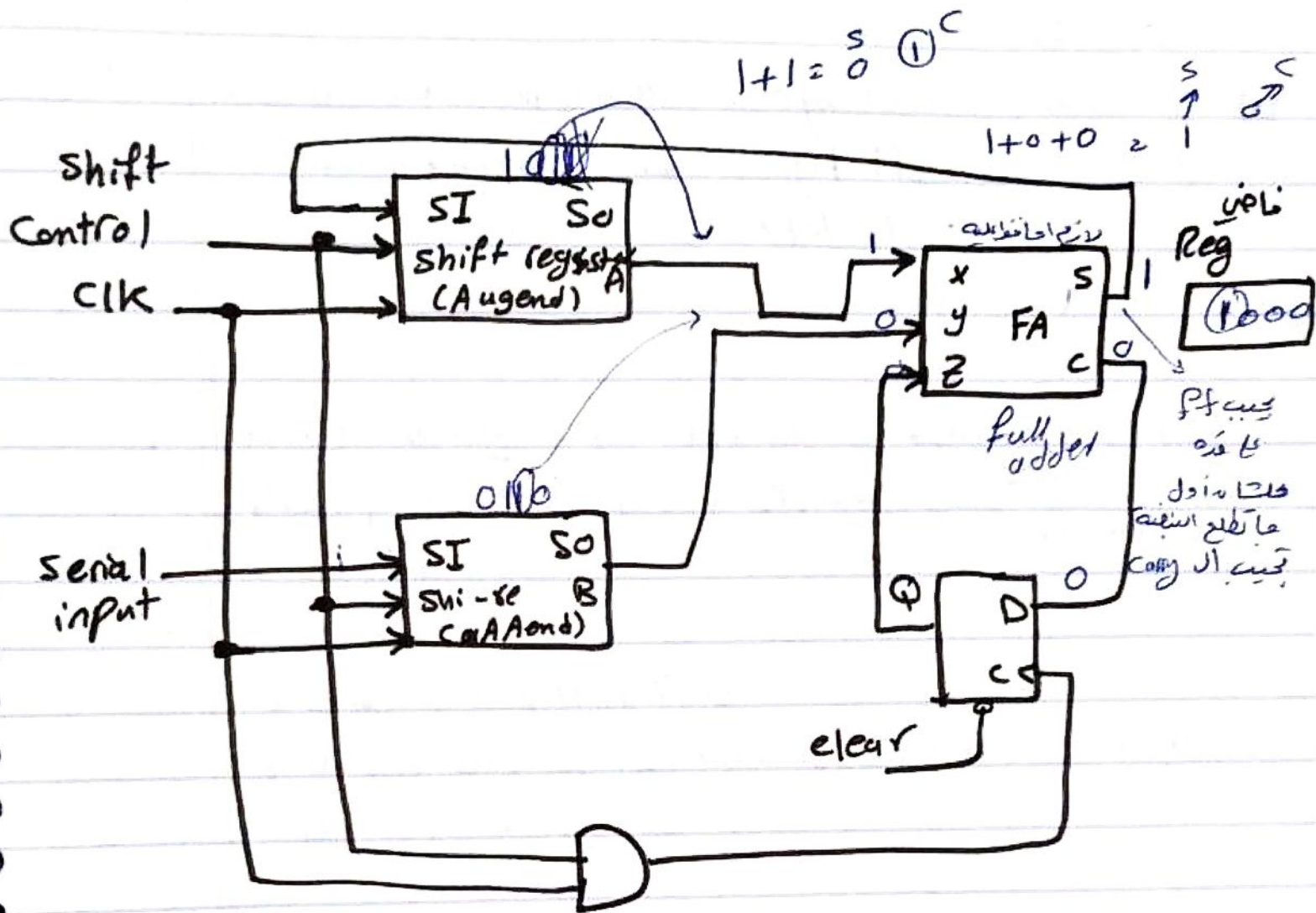
الكتابة : حالة No change : اما بنظر من صفر في S_1 الى Max
 وكيفية رفع اقلها ؟ رفع اجيب من 0 واول السلك نفسه
 بقرأ نفسه
 No change

في صفر رقم (1) لل Shift right
 الفرقية هاي من صفة فيها من تصدينا بالكتاب كمنه لل عملية
 تستعمل في كود بعد

→ Make it 4 bit adders serial



الطريقة هاي فيها كدات [اما بنظر من Full و صر لازم يكون على
 " اما انه تخزينه (يكونوا تخزينه ب 2 register)adder ، لا يطرح ال Full-adder
 Carry and Sum لازم مجموع A_0 و B_0 و ما يصير C أيضا لازم
 يتخزن لانه رفع نصف مع A_1 و B_1 [C في بي يتخزن بدينا $P-F$ و أيضا وجود register تخزينه A و B
 وعنده قدرة جعل اضافة للبيانات بت تكون السبب .



أجهزة الفيديو
الترقية (الترقية)
للقيم

بنا register معك A ونخزنها وكذلك لـ B.
لو بيدي اعمل Shift الجافشة في input 1
آلية العمل: ابي اول البتة من Box A بيجمع اول
رقم مع Box B اول رقم وبتطلع انا عندي ال Carry التي بتجري
منه من ال P-f الهم وبتدنا بتلقاها القيم بتطلع البتة الثانية وهكذا
في register الثاني التي كالمهم بتجزيه الأرقام التي بتطلع من في البتة الثانية
منه بتفوت عليها صفر وهو كانت 1000 بتجزيه اربعة البتة لـ 1
1000 هكذا بتصبح.

في هذا الكتاب: محض البتات من register فانه (أيه أنه شديك S
و SI، كل ما يفضي به في مكانه الجواب

حل آخر: حمله افتره سوال design seq circuit تسبق 2inp

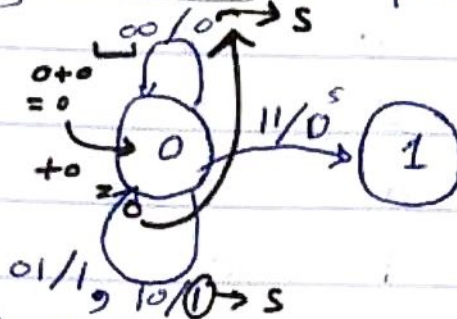
رقم عملية جهم (f-f و f) السؤال ما قبل الجمع وال C ال

بزره احاطه عليه (آخره) ملاحظه

State

مع ال 2bit

diagram



C=0 ال f-f قيه 0 معا هاتيه ال C

لنه f-f اخره تيه C

input

C ال

C A B S C

0 0 0 0 0

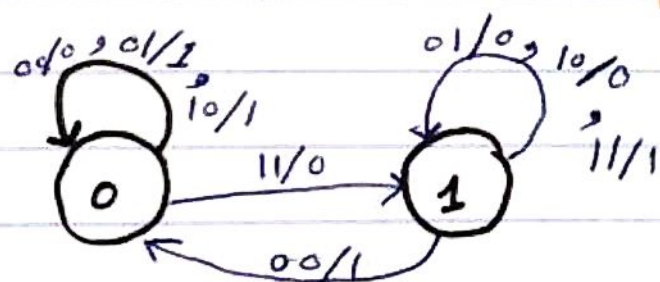
0 1 0 1 0

0 0 1 1 0

0 1 1 0 1

فرضيه

C=0



State diagram

C A B S C

1 0 0 1 0

1 0 1 0 1

1 1 0 0 1

1 1 1 1 1

فرضيه C=1

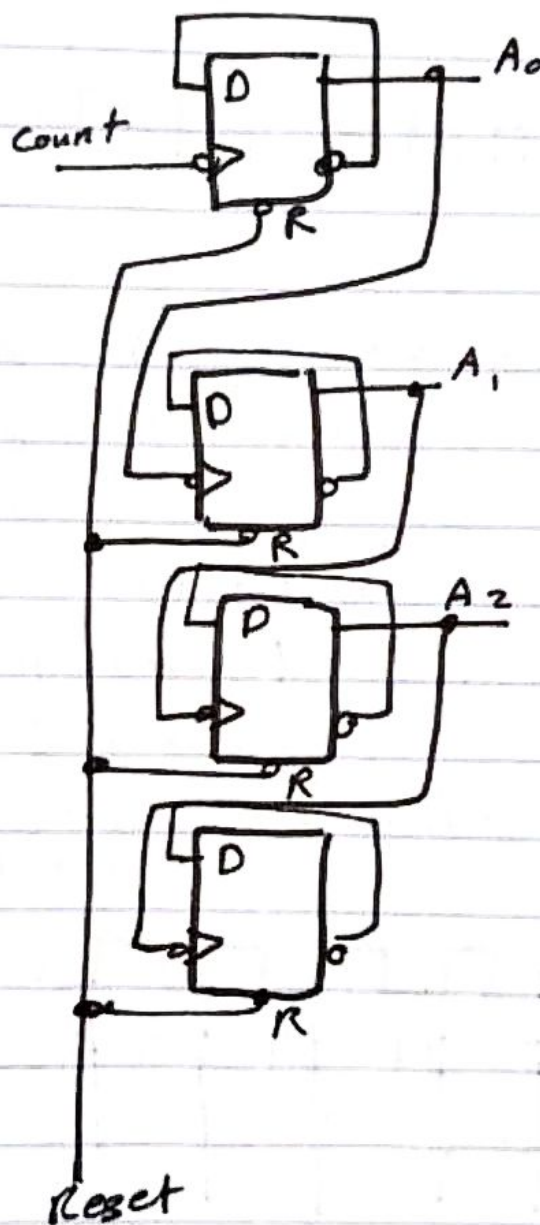
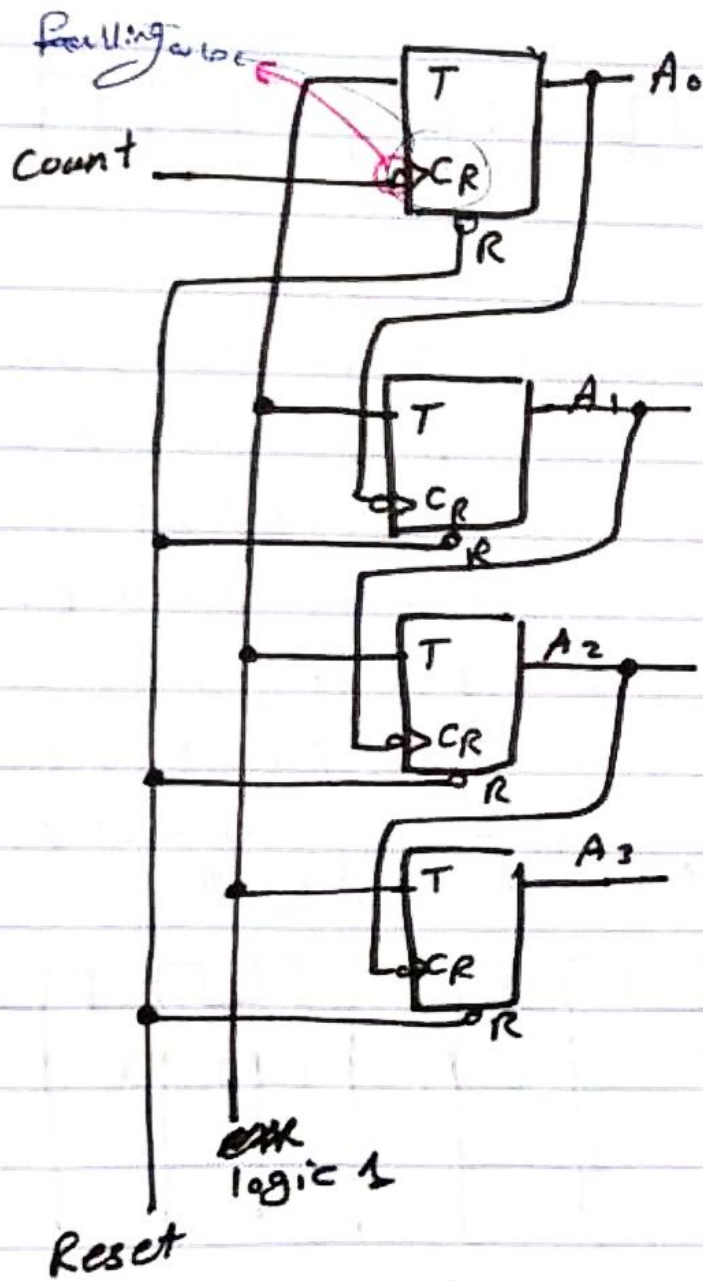
State Table

Application of f.p. of
JK

$A(t)$	x	y	$A(t+1)$	y	JK	J_A	K_A
0	0	0	0	0		0	x
0	0	1	0	1		0	x
0	1	0	0	1		0	x
0	1	1	1	0		1	x
1	0	0	0	1		x	1
1	0	1	1	0		x	0
1	1	0	1	0		x	0
1	1	1	1	1		x	0

Lecture 27:

* Ripple counters.



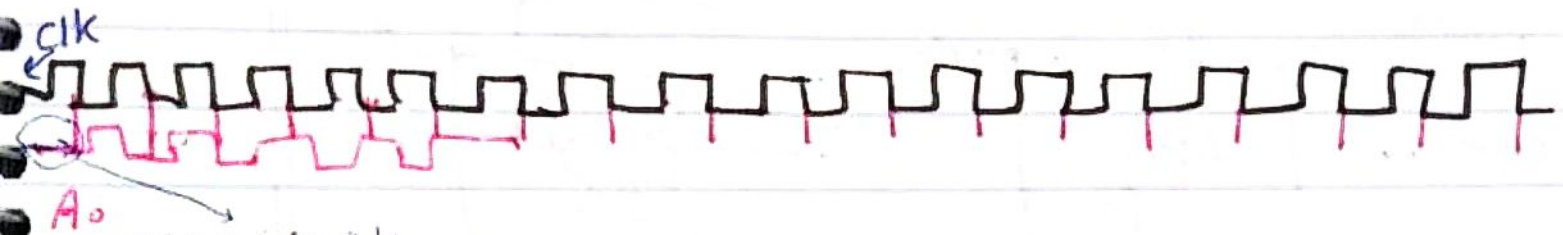
* الرخصة متوافقة بشرط $T = 1$ ، دراسة الثانية على D فاص

Chang D is 25 years old

ملاحظة: كل out_i في ال P هو مدخل لل clk التي وراءه.

(a) Q_0 to CLK and f-f js) An async counter
or
Ripple

- لو ال $p-f$ بح بعد تعداد رج بيل 6 اقل 15
 بح لو بيل استوفه كين بح بعد لازم اصطحب الاقل 6 نبضات او اكثر
 على انه يتكرر ثاني فكله اقل 6 نبضات استوفه 16



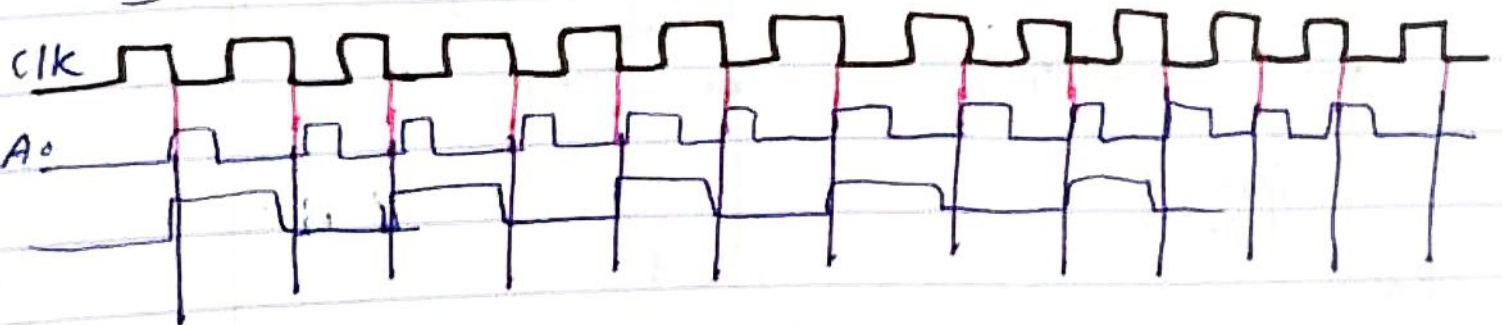
- لو عرفنا سوقيه A_0, A_1, A_2, A_3 في ال 18 نبضه كم قيمهم رج اعرف لعداد
 A_0 على انه اعرفنا لازم اعرف قيه T بح 1 ك القيمة الابتدائية ل A_0 بح صفر
 اذا ما د اعطانا قيم ابتدائية (بفرضها صفر).

⇐ ادك اشير بح قيه A_0 بصفر دتوف قيه T على انه اعرف قيه A_0 اكدرة

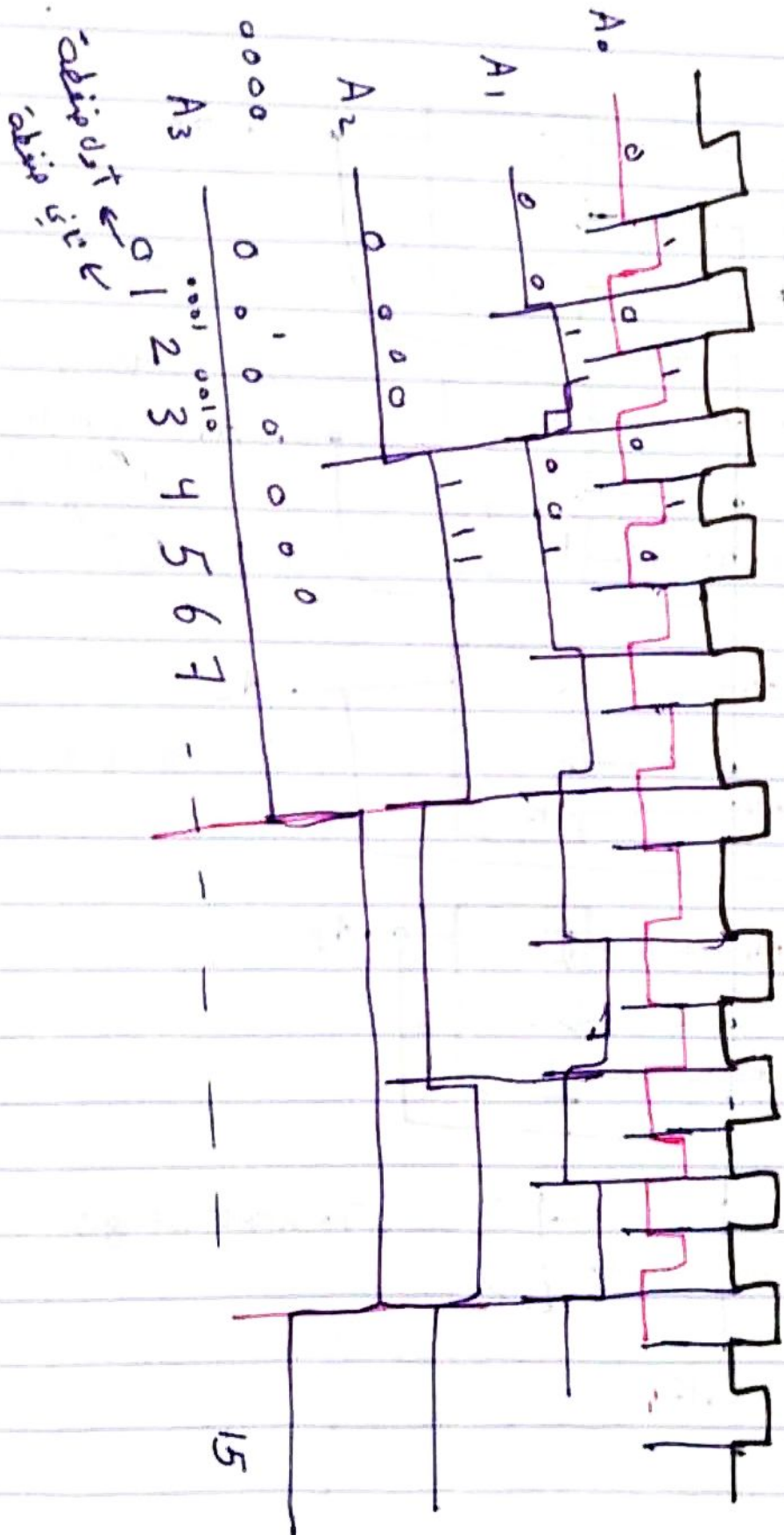
قيه T مستاهل 1 فحيسر $change$

⇐ عند كل falling نبض T عند 0 ← 1 وعند 1 ← 0

⇐ القيمة الابتدائية ل A_1 بح صفر بح لا ادخلها لازم اطلع بح falling ل T تقترن انجا
 $clk \leftarrow A_0$ لانه يعتمد عليها في $p-f$ الاول

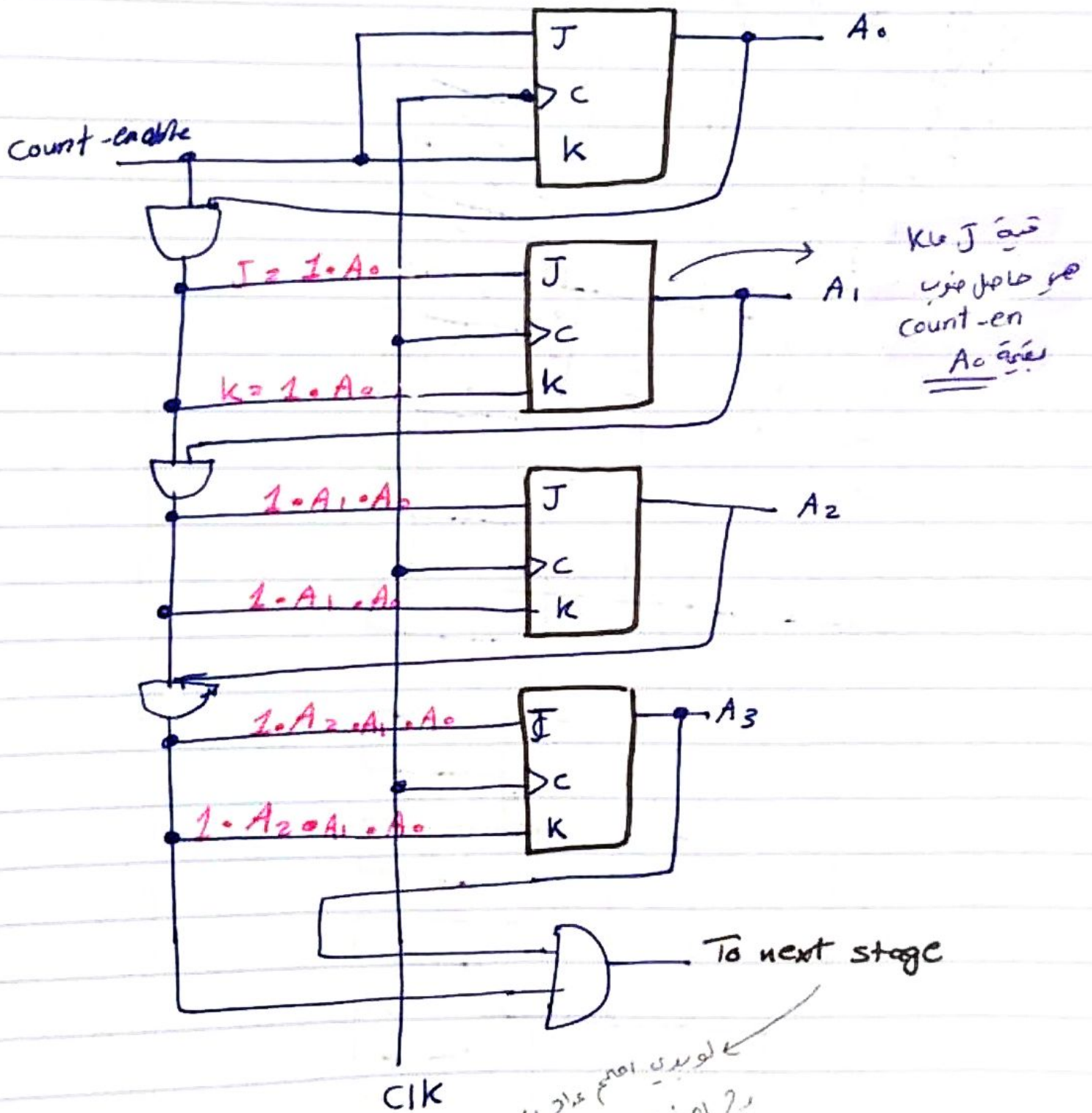


* لومتي موجود ال ۱۰ f-f اربع كيفه ۲۹ ۱۲ m ۷-۰
 وبعده يرجع بكرة ثاني ۷-۰



lecture 28.

* Synchronous Counters *



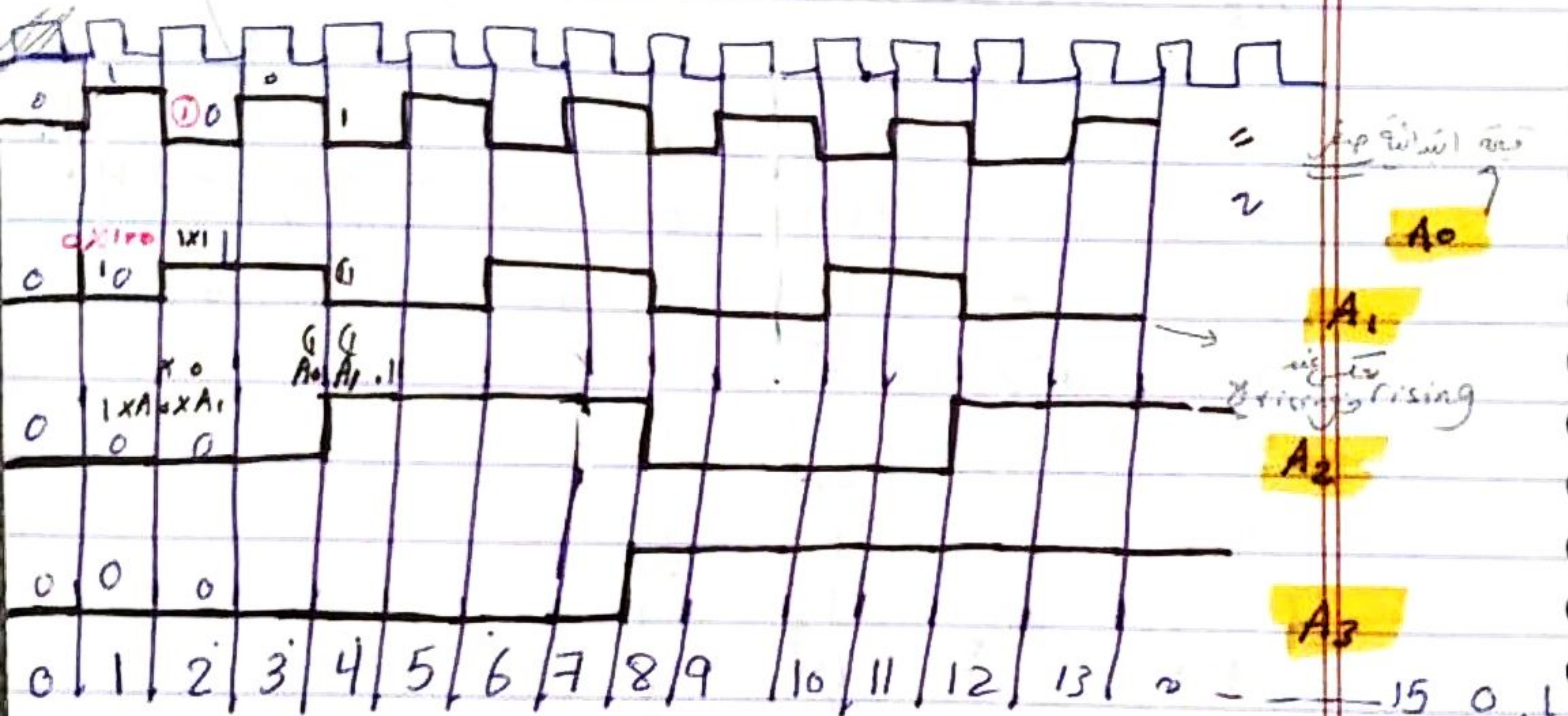
لو بيدي اوتput على 0 و 1
في الـ P-P الـ 2
نفس القارة لـ 2

لنرى لنا الخط للأخر لل rising ؟؟ ال clk هنا هي الجميع كل ال P-F
الموجودة خارج تنقلوا عند نفس ال rising

عدد P-F

في أقصاه عدد من البتات التي يمكنه يوصلها $2^4 = 16$ هي $\frac{16}{2}$
تملكه بعد التردد لكي لا ينفذ لازم يه $\frac{16}{2}$

clk للرجع



إذا بدأنا العداد بعد لازم اصله ال count-en ب 1 لأنه لو وضعت
zero فتية clk يرجع تلوته في نفس صيغة (No change)
في count-en في P-F الثاني ال كما لا يكونوا أصناف متغير
شي

في A1 لازم اعرف شو فتية J ما على n اعرف اصبها
في العداد هاد عداد منه نوع Synchronous بعد تصادياً من 0 ← 15

* Binary count sequence *

8 أعداد
0 →

A_3	A_2	A_1	A_0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
1	0	0	0

أول P-F ينقلب

1 0 1 0 1 0

J K تتغير مع P-F مع
enable

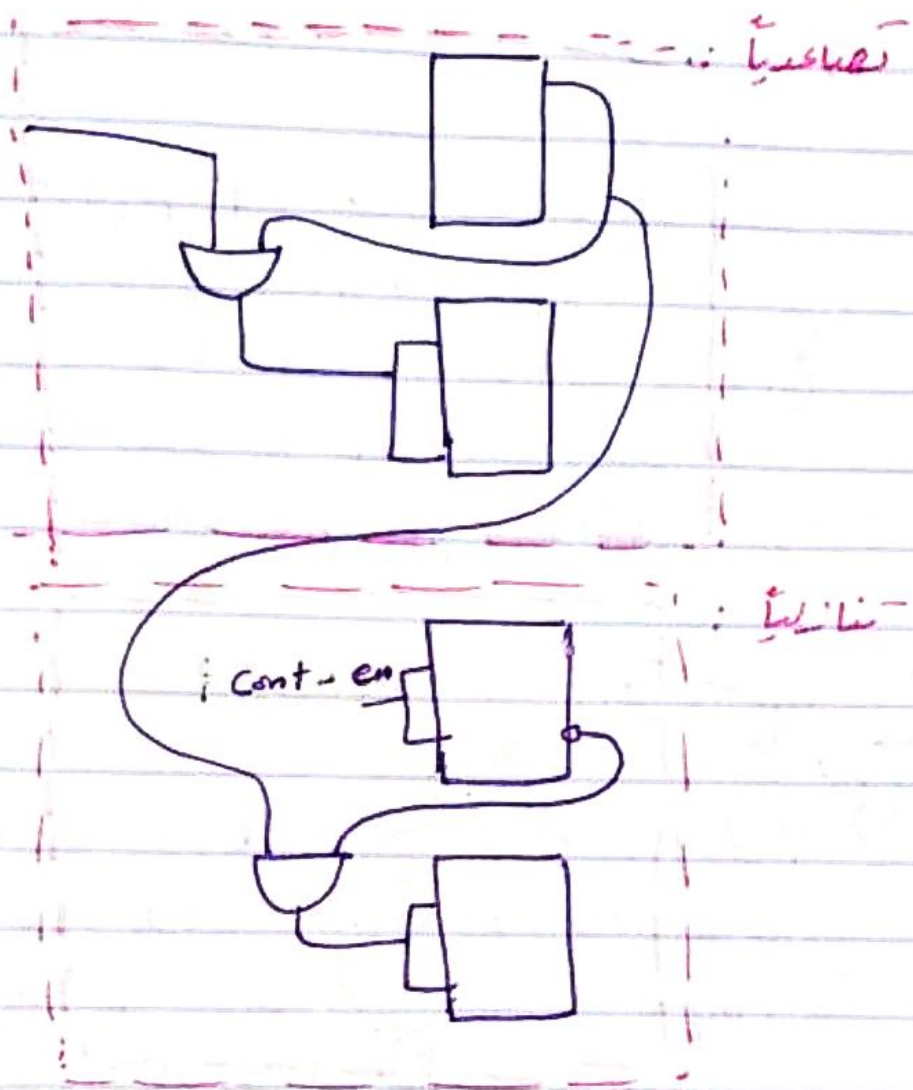
ينقلب نفسه كلما يغير نفسه

← تأتي P-F انقلب لها كان كل اللي قبله وحال يغير آخر اتي في P-F الاول
← انقلب ال P-F اتي عطل (اذا اتي $A_0 \leftarrow 1$ و count 1)
الحالة هي بتغير J و K وحال بتغير Change

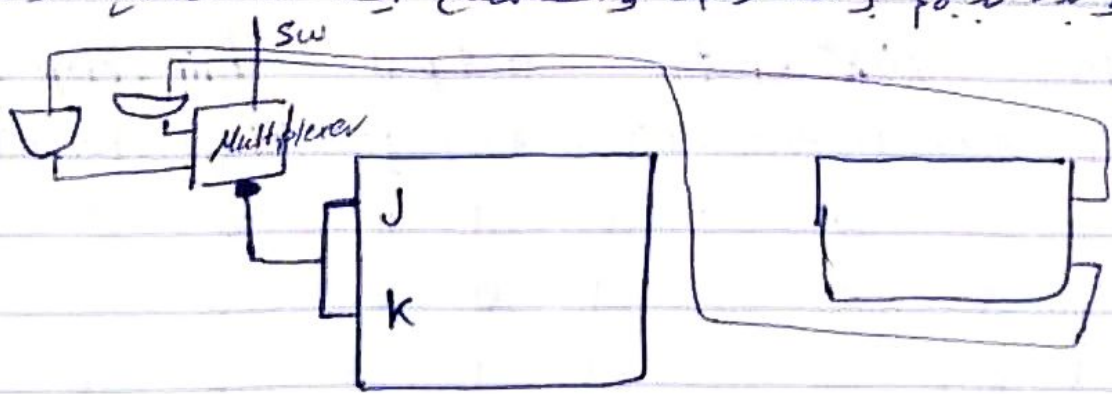
مع كل خطوة / دائماً ال P-F ما يقلب او ينقلب إلا اذا جاد قبله وحال

← لو بدأ ينقلب العملية وانقلبها بعد تنازلياً لمشي مع يقلب إلا اذا كان اللي قبله
تلفظ أصحار

← الفرق بين المتساوي و تنازلي **inverter**



لو بدنا بجمع برسمه واحدة ونخط مفتاح واحد اذا تصاعدي او تنازلي



هو عدد هو هو الطريقة التي راج شغل عليها وبقدر المفتاح م
 م هو عامله قلا كان تصاعدي و تنازلي لو مطلقا المفتاح و
 ح شغل كل شي مع التصاعدي. (ملاحظة ان Mux بيه A_0 و A_1
 مودلهم مع and) **Done** #
 ان ك ج مع مفتاح م

9 دوائر 4ff کے ساتھ

7 دوائر 3ff کے ساتھ

4 دوائر 2ff کے ساتھ

Lecture 29 :

محاضرة اليوم تتطابق مع جابر في مديته المضمون

كيف نعلم عدد بدي من 0 إلى 9

والتي أفدها في جابر في بدي من 0 إلى 7

رج نحتاج إلى 4ff وكونه 16 احتمال بنظم 10 معر

0 إلى 9 ، باقي الاحتمالات رج نغفلهم **don't care**

State Diagram

* تم output؟؟ **خداں 4ff مرفوض**

output

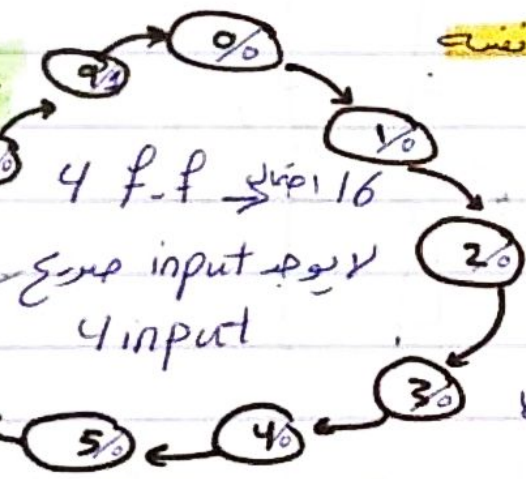
في جابر في 12 تغلنا انوفاني

output صرع بنا عدد ونعلم

في هذا المثال مطلوب منا اعتراف

وجود output صرع وما يتلح 1 لا

عند آخر حدة



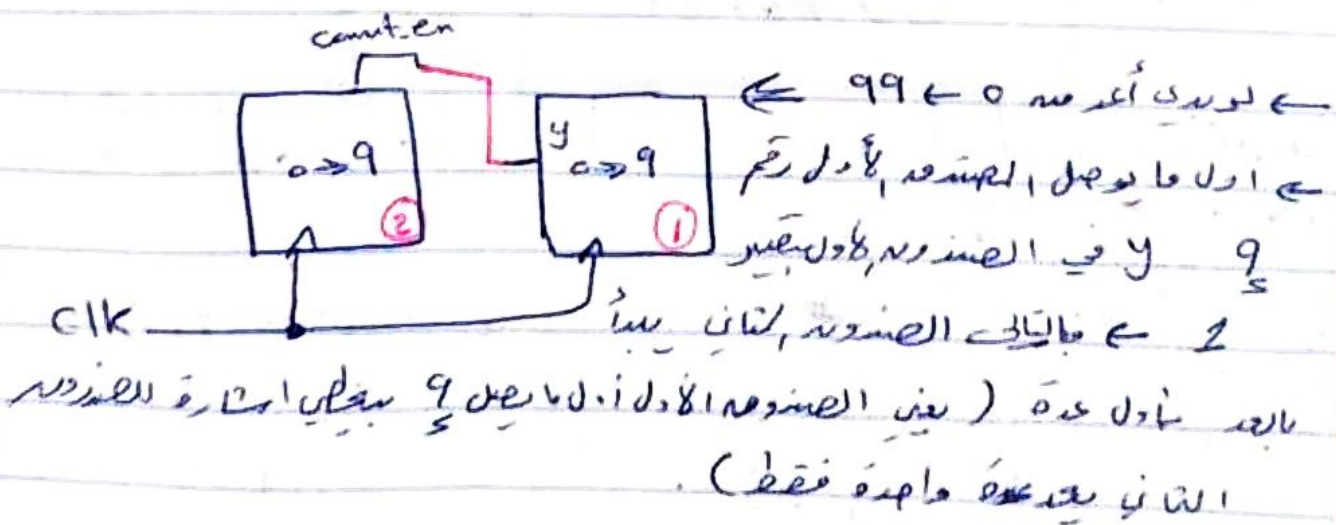
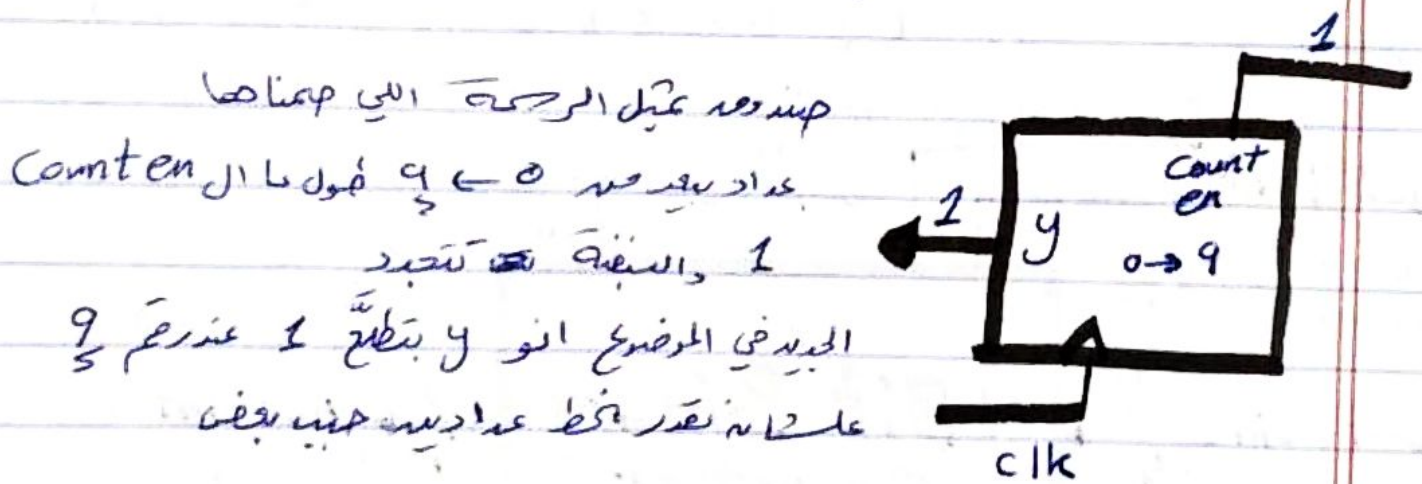
Present state Next state out put Flip-flop Inputs

D(t)	C(t)	B(t)	A(t)	D(t+1)	C(t+1)	B(t+1)	A(t+1)	y	TD(t)	TC(t)	TB(t)	TA(t)
0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	1	0	0	1	0	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	0	1
0	0	1	1	0	1	0	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	0	1
0	1	0	1	0	1	1	0	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	0	1
0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	0	0	0	0	1	1	0	0	1

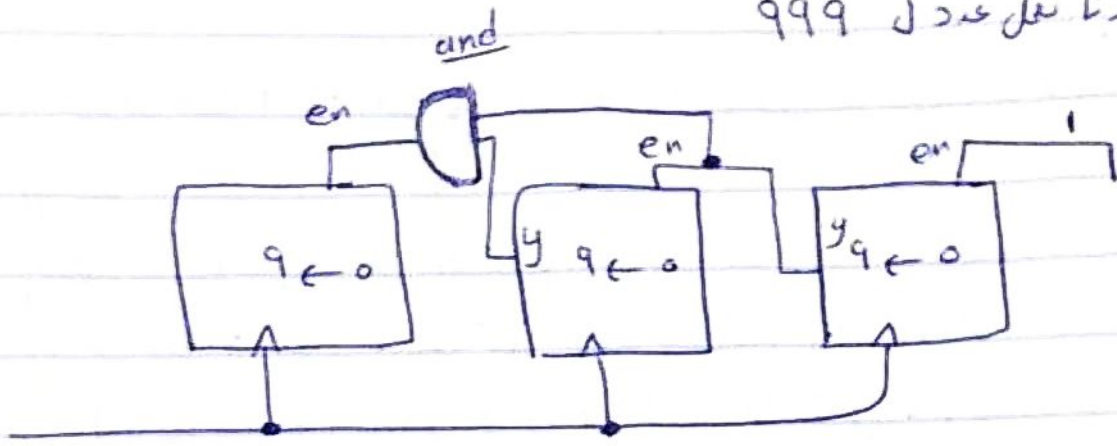
في State table طالما ان الاحتمال انه الباقي فكلنا care don't

في حالة تسمى كل الارقام لا يقيم دوائر وأخيراً كل دائرة دونه بترجع في محاضرة اليوم من كل الأرقام الموجودة إليها دوائر.
(التي لها دائرة يسجل في ال State Table تسجل قيمها دونه بترجع والتي لها دائرة في ال Stat diagram بخطها care don't).

في أول سطر (لو بدنا نستخدم نوع ال I-P-P) فتلاقي د D كانت
مفر د راعت من (No change) حفظ 0 و الباقي زيها اما 1
كانت مفردات 1 (Change) حيسر تغير 1



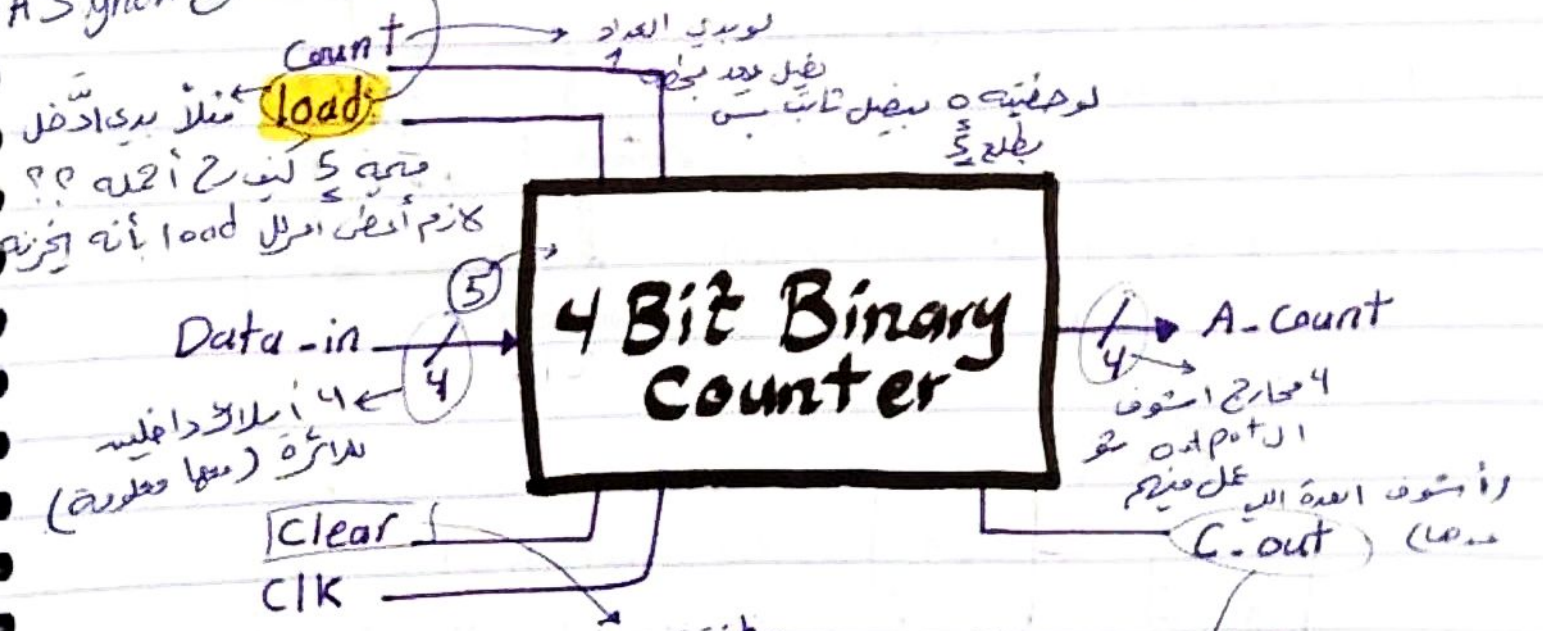
← لو دنا نقل عدل 999



← خانة المئات ممنوع تعد إلا إذا اصب 99 يعني الأحاد 9 والعشرات 9 : المئات بتزيد 1

← دهنية بوابة AND ممنوع اللي في خانة المئات يقرب إلا بشرط (يكون في خانة العشرات 9 و اللي في خانة الأحاد 9).

← من نوع ASynch



دهنية لوبي اتمه
ال content
للعداد بخطه بصفر
مؤقتا وبقيد 1
(لوظية بصفر مستخرج به
العداد) لازم ارجعه 1
علتانه بعد

4 bit → 0 → 15
3 bit → 0 → 7
5 bit → 0 → 31

4 bit Binary
نفس كل في المثال
السابق

متزامن مع ال clock (من اول ما يمس الصفر)
 ومن يعني بعد مدة شوي يمين
 Synchronous Clear

ملاحظة هامة : انواع ال Clear

A Synchronous Clear

اول ما يمس الصفر طول يقف (Clear غير) $\rightarrow$
 متزامن او مرتبط ببداية .

في هذا المثال من نوع **A Synchron** يعني ال $\uparrow$ من صفوة حتى لو
 لدينا load $\uparrow$ لا لما يجي ال rising

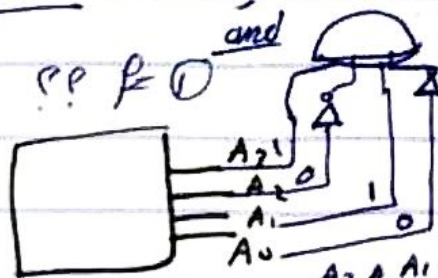
لو لدينا load $\uparrow$ count $\uparrow$ اولو عكنا متوج يمس؟؟ لازم ارجع
 ال Datasheet لل IC الموجودة عندي وانسوف لمين كذا ولور كذا

Datasheet

Clear	clk	load	count	Function
0	X	X	X	clear to 0
1	$\uparrow$	1	X	load inputs
1	$\uparrow$	0	1	count next binary state
1	$\uparrow$	0	0	No Change

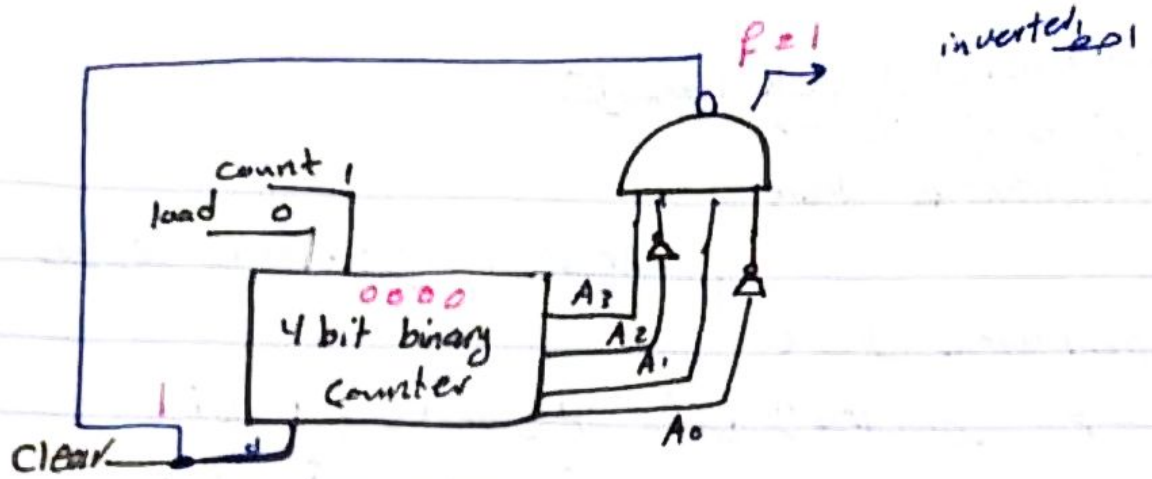
* الأولوية القصوى ل **clear** $\leftarrow$ **load**

على استه عليه : ① كيف بي افابي هاد العداد بعد مدة 5 $\leftarrow$ 9 ؟



من F بتطلع 1 ؟ اذا كانت $A_0 = 0$
 $A_1 = 1$ و $A_2 = 0$ و $A_3 = 1$

لو حطينا inverter على **and** متغير F
 بتطلع 0 اذ عند 10
 لو حطينا inverter على **and** متغير F بتطلع 1 اذ عند 10



بعد الصفر رج بعد طبيعي واصل من 0 إلى 9
 أول ما يتغير 10 حيعطي clear وبصفر ال content ويرجع 1 وبالتالي اعداد
 رج يرجع بعد 0 1 2 ... (كل ما يوصل 10 يرجع بصفر) **أول سؤال نـ**

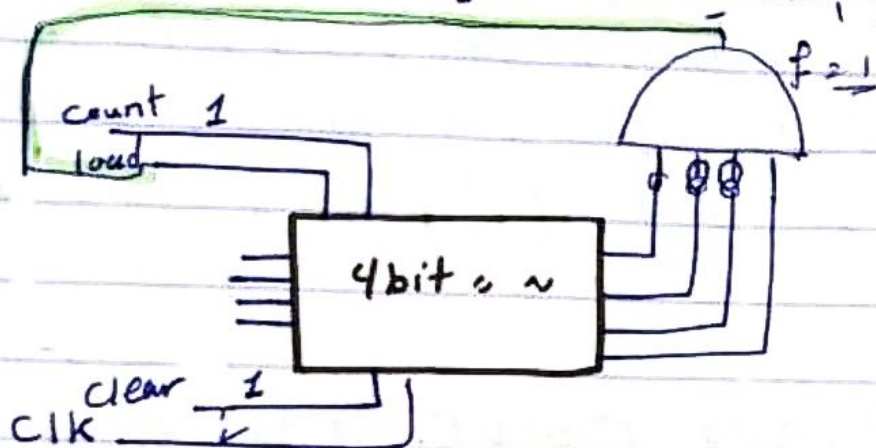
طيب اجمالي برمجتها عند 10 تصفر؟؟ ليس مت عند 9 ، فدياً لو عملناها
 على 9 تصفر انا رج استوف آخر دة 9 وبعد هيك صحت رج الحصة استوف 9 حصيفها
 خطف 8 اعداد فقط متراسمة ، لو كانت متراسمة كان عملناها عند 9 لأنه كينا ما يصفر
 على طول بعد يتن سوي بعد بصفر ولكن هنا ال clear متراسمة فما يربط
 أحطه عند 9 (الفكرة بار اختيار : لما يكون عند clear غير متراسمة
 بطعة زائدة).

ثاني سؤال نـ

هل دائماً أطق NAND أم AND ؟؟ الجواب :
 مع حسب طبيعة ال clear إذا ال clear يحتفل عند صفر بطها
 NAND علشان يطالع صفر منه 10 فقط .
 إذا ال clear يحتفل منه $\frac{1}{3}$ بطها AND

يوجد حل آخر بطريقة الكتاب يعتمد على load نـ :
 في الصفحة القادمة !! نـ

→ أخيراً إعداد بعد منه 5 ← 9 ما يستخدم في كل load



* أمثالنا : بعد منه 5 ← 9 وأول ما يصبح 10 بعد منه 5
 → أنا هنا بعد ما أدخل 9 لازم أعطى إشارة لل load حمل قيمة 0
 لازم أعطى بال input صفر ← مستش الخطة المناسبة العمل عملية loading
 ومنه أعلم عند 9 و 10 ← حسب أن load إذا كان مترامم وبتش
 زمني بطلع على آخر عدة في العداد اللي بيدي يام يعني 9
 أما إذا كان ال load غير مترامم بطلع على عدة زيادة يعني 10
 → هنا عارضة load بتقل عند 1 يعني لازم اعمل AND وليس NAND
 يعني عمرها F ما بطلع إلا إذا عند الحالة لما بيدي رقم 9
 → أدل ما يوصل 9 ← F_{load} ← load روح يصير 1 من الإدوية
 رانه إخراج منه count روح يصير القيمة اللي أمنا ← كاتبينها بعد أدخلها

الخلاصة : كانه عناعد بعد مثال منه 5 ← 15 بدنا نلعب بالأعداد اللي برا
 بحيث أحليها تقديريم أنا بيدي مثال 6 .

① روح نطلع على الرقم ونهضير 6 هبل اعمل AND لي 6 ولا وراها عدة ل 7
 اللي أجزم هبل load أو ال clear اللي روح تقاقل مع 9 منه نوع مترامم
 ثم غير مترامم ، إذا كان غير مترامم بطلع على عدة زيادة سوار
 فتاة مع load أو clear ، أما لو كان مترامم بطلع على آخر عدة عند .

③ أمطعم في AND أم NAND ، مع سوال clear و load و...

تتغير اذا عند 1 ← NAD ، اذا عند 0 ← NAND

④ شدة ما بدى ياه ؟ عند 0 ، يتعمل عند رقم عدد مثلاً 6 و 2 حاضري

طريقة واحدة وهي باستخدام load ما بنفع ايشن على clear

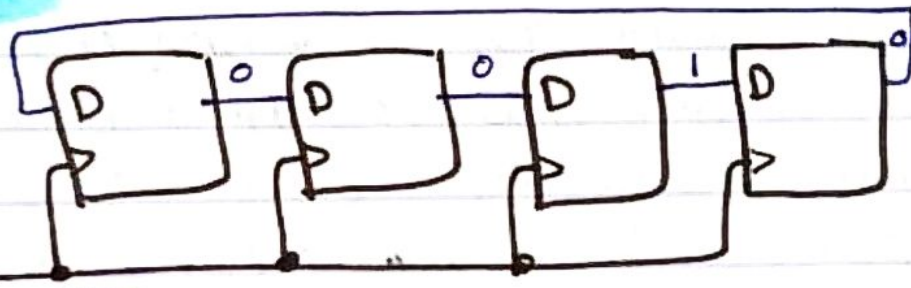
آخر سلسله
وأخيراً

* Other counters *

Ring Counter

كل الخارج أصغار
ما بعد واحدة تتكون

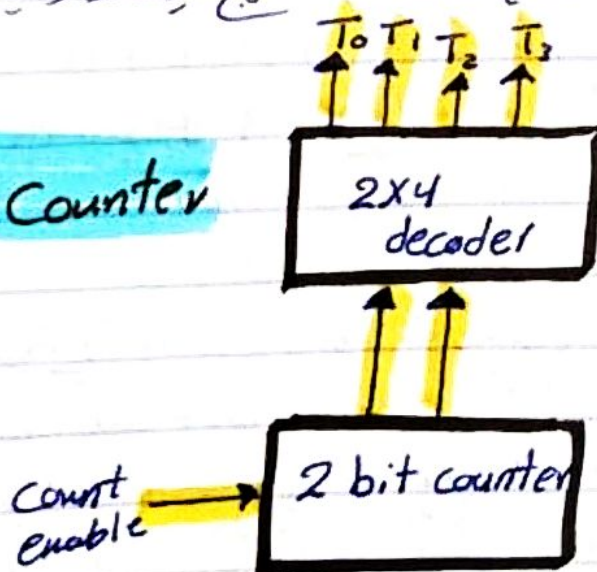
clk



4 bit

0010 عند ادخال 0010
0001 ← 0010 ← 0100 ← 1000
كل الخارج أصغار ما بعد واحد يخرج يكون 1 و يلف 2 ما بعد 3 ؟
⑤

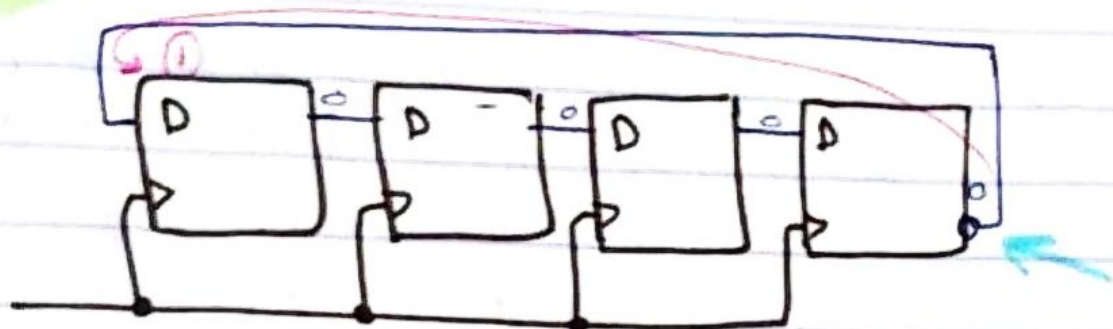
Ring Counter



Decoder 4

Johnson Counter

عدد مدار هفتی مدول میم نیو توصیل خرج
آخر P- مکتوما ای دخل اول P-
→



- ① ← 0000
- ② ← 1000
- ③ ← 1100
- ④ ← 1110
- ⑤ ← 1111
- ⑥ ← 0111
- ⑦ ← 0011
- ⑧ ← 0000

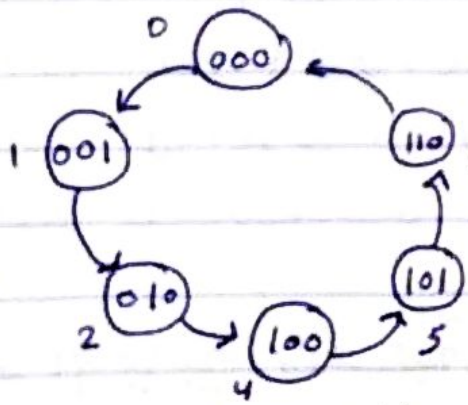
نقطه:
مدول ال States
في Johnson عدد P
 2^X

رجعت للأصل بعد 8 states

States = 4 x 2 = 8

آخر نقطة: تصميم عدد بعد بلك غير منظم عدد ما بعد حالات .
لازم أحمد عدد بلاك

input مرجع



غير منظم

	A(t)	B(t)	C(t)	A(t+1)	B(t+1)	C(t+1)
0	0	0	0	0	0	0
1	0	0	1	0	1	0
2	0	1	0	1	0	0
3	0	1	1	X	X	X
4	1	0	0	1	0	1
5	1	0	1	1	1	0
6	1	1	0	0	0	0
7	1	1	1	X	X	X

الخطوة البورد اصحا بحمد الله نوع ال
P.F. وأرسم على k-map بحمد الله
معاولات وأرسمها بالآخر .

ملاحظة عنظم في الصخرة القادرة

هكذا وانما نبلغ بالسيرت مثلاً لتعرفن حدث ضوفاً أو شَوْحاً بالخطأ وانقلنا
ل دائرة في (٥١١) السؤال كيف رح أمي بعدها؟ وسيد رح أروح؟

برجع لخريطة السيرت التي مهمناها "المجول" **٥ ١ ١ X X X**
عجب ال X إذا دخلت بالاختصار في k-map معاًها ١ وإذا ما دخلت
بالاختصار تكون ٥ ، لتعرفن مثلاً دخلت بالاختصار ستكون ١ ١ ١ انقلنا
على دائرة 7 (٥١١) ← (١١١) طيب ودائرة (١١١) سيد رح أروح؟
رضونني الغلة. عجب ال X حشوناً دخل بالاختصار k-map ولا لا؟
مثلاً **١ ١ ١ X X X** كانت X ← ٥ ١ ١ يعني كانت (١١١) ← (٥١١)
كانت 7 راحت في هم وجيد؟؟ 7 ← 3 ← 3 ← 7 ، خربت كل
عملية العد. (النظام يتعطل) بسبب خطأ بسيط.

← كيف أصلح هاي العملية من الخطأ؟؟ في حي اسمها self correct
تصلح ذاتي. الدائرة تصلح نفسها. تبطل ما بنا أنو في خطأ.
رح يصير ، بدم مضيت X X X خطينا ٣ أخطاء ٥٥٥ بيسر يصح
ذاتي يعني ٣ لو حد وصلها حشون ل صفر ٥١١ ← ٥٥٥ والصفر
نادي بابل الهدم أدلو وجيد. (بطل اصباح) انو لو صار خطأ بقي عتبه
آلوه حاطة قية ٥٥٥ بعد من أول وجيد.

← نحو الإيجابية والسلبية لما عملتا هيك؟؟
الإيجابية ← أنا صحت لو صار خطأ يرجع للنظام من أول وجيد
السلبية ← أنو خربت السبب في k-map توفر في التكاليف
← طب أنا كهندسة؟؟ شو أعمل؟ **بمسأه أهتم** (أعمل دائرة تكونه
أصغر ما علية ولو صار خطأ أخرب الدائرة ولكي أعمل ما باقي ونزيد شوية
بالتكاليف ونخلي الدائرة تصلح نفسها)؟؟ أنا بقول أهتم إجابي
أهمه نـ

الإلهامية : على حسب أناوسيه بيدي استخدم هاد النظام يعني تبدأ بيدي أمطها
في أعماص المحيطات : على سطح القمر (أمالته الوصول صعبة جداً)
وضع العبارة لا ... أخسر سوية فلوس وأفلي الدائرة تقبل نفسي
نفسيا بالله لو كانت الدائرة قديماً قريبة مني لا أخسر فلوس
: وقتاً ما يصير فيها خطأ بعمل إلها restart ، هو أنا فضيالكها ؟؟

Good luck ::

