Lec (07)

Registers

هما احد تطبيقات Flipflop

وله استخدامين ① تخزين البيانات data store

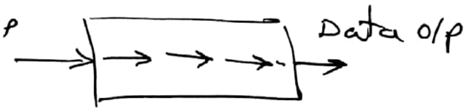
② نقل " " data movement

مثال

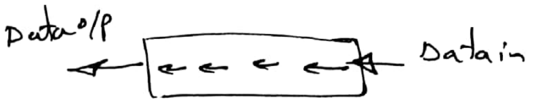
عندي مخزن عاوز أخزن فيه مجموعة من البيانات .. المخزن ده له باب بيدخل منه بيانه بيانه (0 أو 1) ثم يتم تخزينها جوه المخزن وذلك بدخولها بطريقة تسلسلية (واحد واحد) ويبقى في باب تاني (مخرج) للحصول على البيانات المخزنة جوه المخزن وتطلع تلك البيانات أيضاً بطريقة متسلسلة مع ملاحظة أول بيانه بيدخل هو نفسه أول بيانه بيخرج ويسمى هذا المخزن Register وتسمى تلك العملية

shift register = serial in/serial out

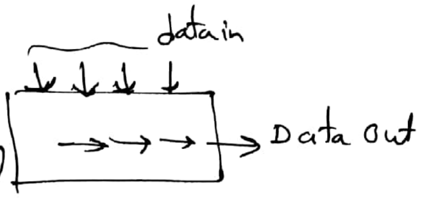
① serial in/shift right/serial out



② serial in/shift left/serial out

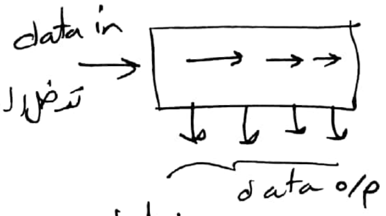


③ Parallel in/serial out



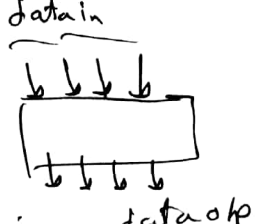
(وجود اكثر مدخل للداتا ومخرج وحيد (بطريقة تسلسلية)
كل مدخل عليه بيانات واحدة في نفس الوقت

④ عدد المخارج = عدد البيانات التي تم تخزينها داخل الـ register
تدخل البيانات بطريقة تسلسلية ثم تخرجوا مرة واحدة في المخارج



serial in/parallel out

⑤ Parallel in/Parallel out



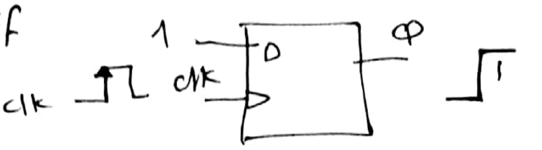
. يتم وضع البيانات على المداخل في نفس الوقت وتتخزن داخل الـ register ثم نحصل عليها مرة واحدة في نفس الوقت.

. في حالة عدم وجود data على المداخل .. يحتفظ اللي لسه متخزن في الـ register

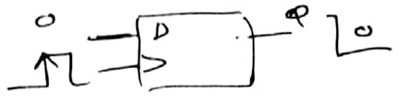
Shift Registers

تتكون من مجموعة من Flip Flops

Remember D-FF



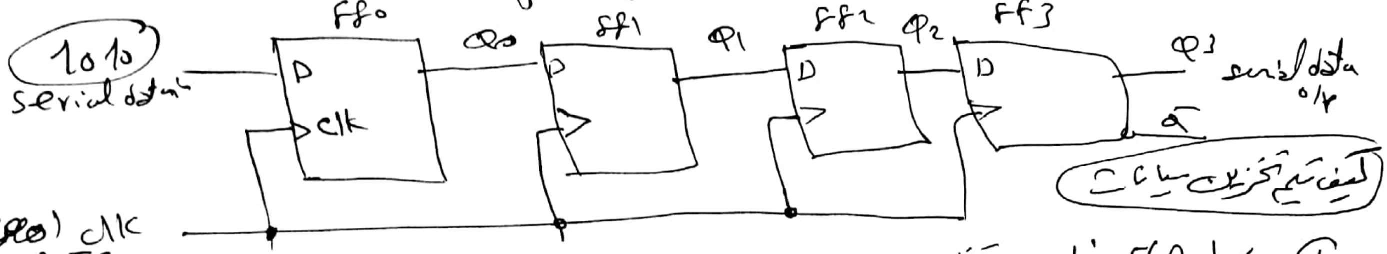
o/p high when +ve edge
Q=1



o/p low for +ve edge
Q=0

ولا يحدث تغيير حتى لو D اتغيرت الا عند وجود +ve edge فقط

① serial in/serial out shift Register (4 bit shift Register)



clk الموصل لكل FF في نفس الوقت

(كيف يتم تخزين بيانات)

① كل flipflop يتخزن فيه بيان واحد one datum

② في البداية خالص نصفر كل الـ registers (Reset) (يعني الداتا بتكون صفر) ممكن تتعمل عن طريق pin موجوده في shift register

③ نفرض عاوزين نخزن data (1010)

④ هندخل على أول ff (ff0) أول بيان من اليمين (0) ← LSB ولكن الـ0 مش هيطلع ع Q_0 إلا في حالة دخول أول clk (+ve edge)

clk	FF0 Q_0	FF1 Q_1	FF1 Q_2	FF2 Q_3
intial	0	0	0	0
1	0	0	0	0
2	1	0	0	0
3	0	1	0	0
4	1	0	1	0

⑤ ندخل بعدكده (1) ع أول (ff0) لما يدخل +ve edge clk بينما الـ0 الي كان خرج ع Q_0 انتقل لـ Q_1 يعني الـ1 أول Trigger داخل ودخل لـ Q_0 في نفس الوقت ∴ حصل ازاحه لـ 0 لـ Q_1 وحل محله 1 عند Q_0.

⑥ عند الـ Trigger 3 تدخل 0 لـ ff0 (وكذلك Q_0) في نفس الوقت يحصل إزاحه لكل من (يعني الـ Trigger في نفس الوقت)

أ- الـ0 من Q_1 لـ Q_2

ب- الـ1 من Q_0 لـ Q_1

جـ- يحل الـ0 الجديد لـ Q_0

⑦ عند الـ Trigger الرابع يحدث الآتي

أ- ازاحة الـ0 القديم من Q_2 لـ Q_3

ب- الـ1 الـ1 من Q_1 لـ Q_2

جـ- الـ0 من Q_0 لـ Q_1

د- يحل الـ1 الأخير وجديد لـ Q_0 .. وهكذا

Scanned by CamScanner

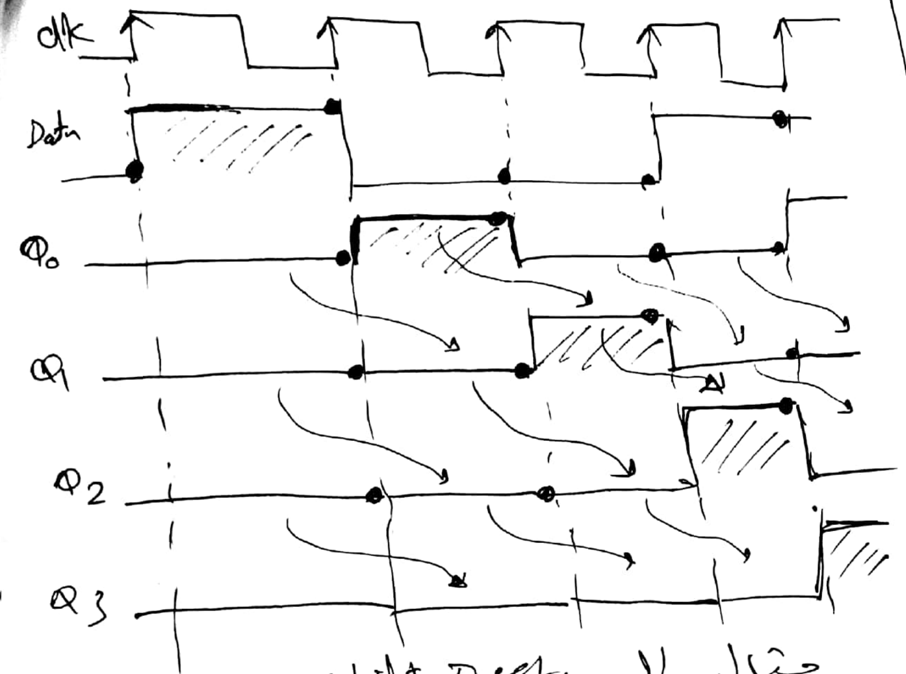
(٣)

Register دخل serial

كدة البيانات كلها دخلت واتخزنت في صورة

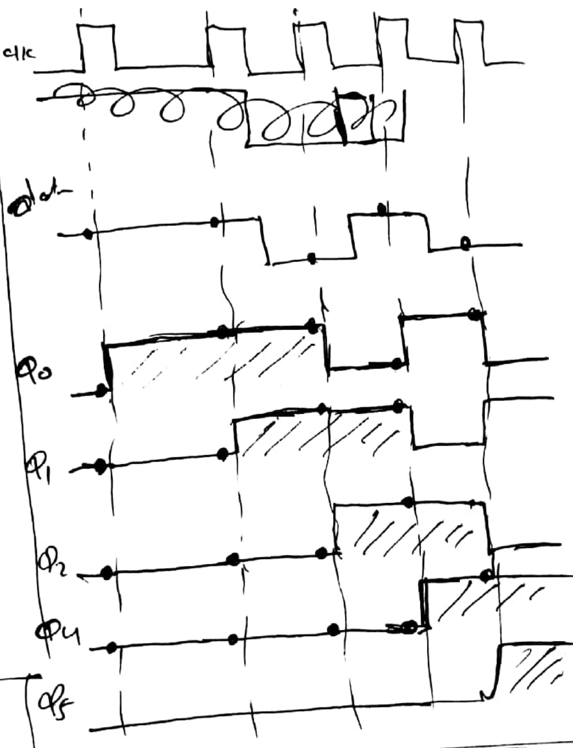
الآن كيف نقرأ (نخرج البيانات) من ال shift reg. ؟؟ (الاجابة بعد)

ex①

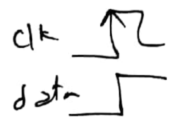


مثال لل shift reg.

ex② register 11010



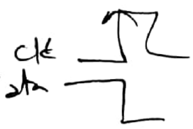
قاعدة مهمة في الاستنتاج لو clk



الداتا طالعة في نفس الوقت بتاع edge تأخذ الحالة اللي كانت قبل الطلوع لا طول يعني حالة low

يعتبر الداتا حالة صفر :. Din = 0

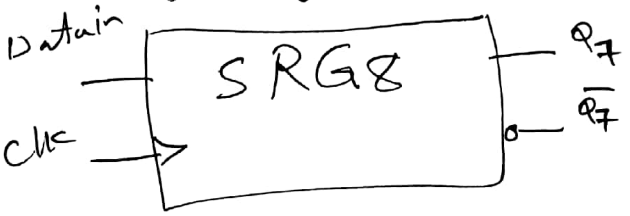
لو clk



يعتبر اخد حالة للداتا قبل ال edge high :. Din = 1

Shift Register symbol

SRG8 = shift register with 8 bit capacity



ملاحظة LSB تدخل أولاً

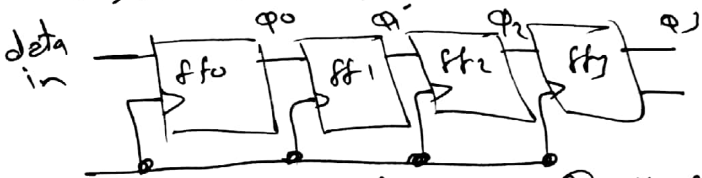
Scanned by CamScanner

كيف يتم استخراج البيانات من المخزنة Register

clk	FF0 Q_0	FF1 Q_1	FF2 Q_2	FF3 Q_3
initial	1	0	1	0
5	0	1	0	1
6	0	0	1	0
7	0	0	0	1
8	0	0	0	0

1) هنستعين ب ٤ clk تانيين عشان اطلع ال 4 داتا الموجودين (المخزنين) داخل ال shift Reg.

2) لاحظ هنا لو مطيناش حاجة لل Data in :: الهم كده مش هنخزن حاجة جديدة وانما هنطلع فقط المخزن عنده أما لو دخلنا داتا في نفس الوقت اللي بيدي فيه clk جديدة :: بيعرض داتا وكمان بخزن داتا جديدة في نفس الوقت (لاحظ البيانات 1 0 1 0 — Q_0 Q_1 Q_2 Q_3)



3) هنبدأ ب clk رقم ٥ ودي كده هاتشغل الداتا الموجودة في D تباع FF3 وتطلع في Q_3 (1) (كانت لاول 0 بقت 1)

4) هتدي بعد كده clk تانية (6) يطلع في Q_3 ال 0 (كانت 1 بقت 0)

5) وهكذا في كل clk هترحل الدخل لناحية اليمين ويطلع كله في Q_3

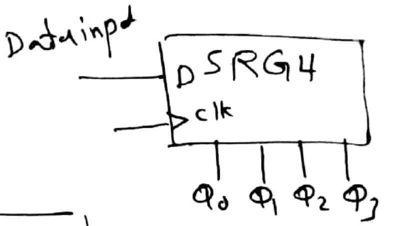
6) سؤال مهم لو احنا مدخلناش حاجة جديدة في D ايه اللي بيحصل اثناء ترحيل البيانات يعني دلوقتي عرفت مثلاً ان أول clk (5) تطلع (ترحل) ال 1 عند Q_3 :: طب الذي دخل مكانه على ال Q_0 ؟! (الحقيقة هيكون صفر اللي دخل)

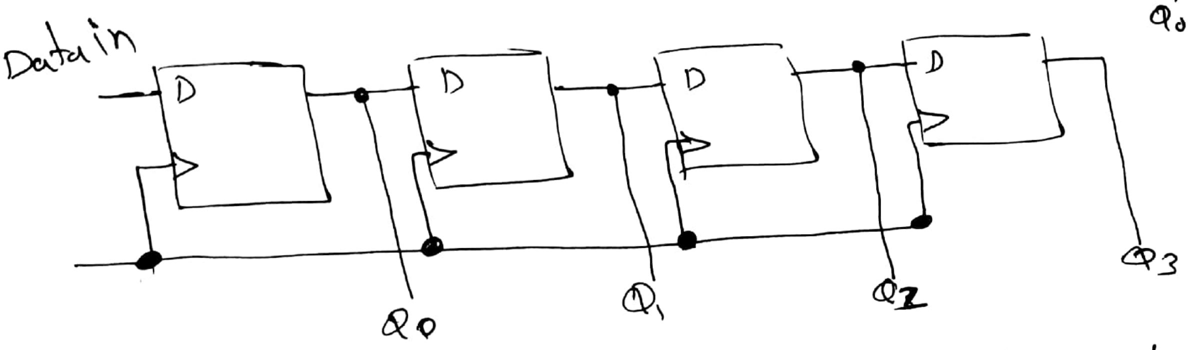
وبالتالي بعد نهاية كل ال clk تجد انتا خزنتا اصفار في كل ال Flip flop أو كأننا عملنا Reset وفي نفس الوقت طلعت الداتا اللي كانت مخزنة

FIFO First in First out — أول بيان بيدخل هو اول بيان نحصل عليه

Scanned by CamScanner

2- Serial in / parallel out





* أي data تدخل مش هتبان كدة لما توصل لآخر FF وإنما أول ما تخرج من FF Q0 تطلع مباشرة من Q0 دون انتظار لآخر FF .. ونحتاج لـ 4clk وبعد آخر clk يجمع البيانات اللي دخلت للخارج.

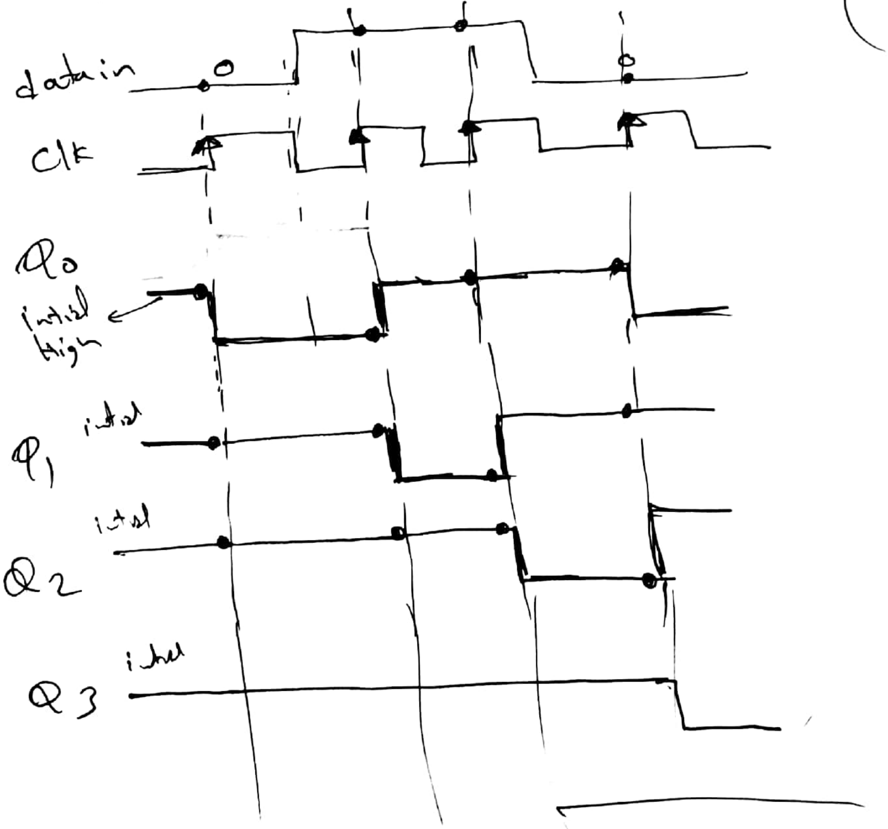
Ex 74HC164 8 bit serial in/Parallel out

14 Pin

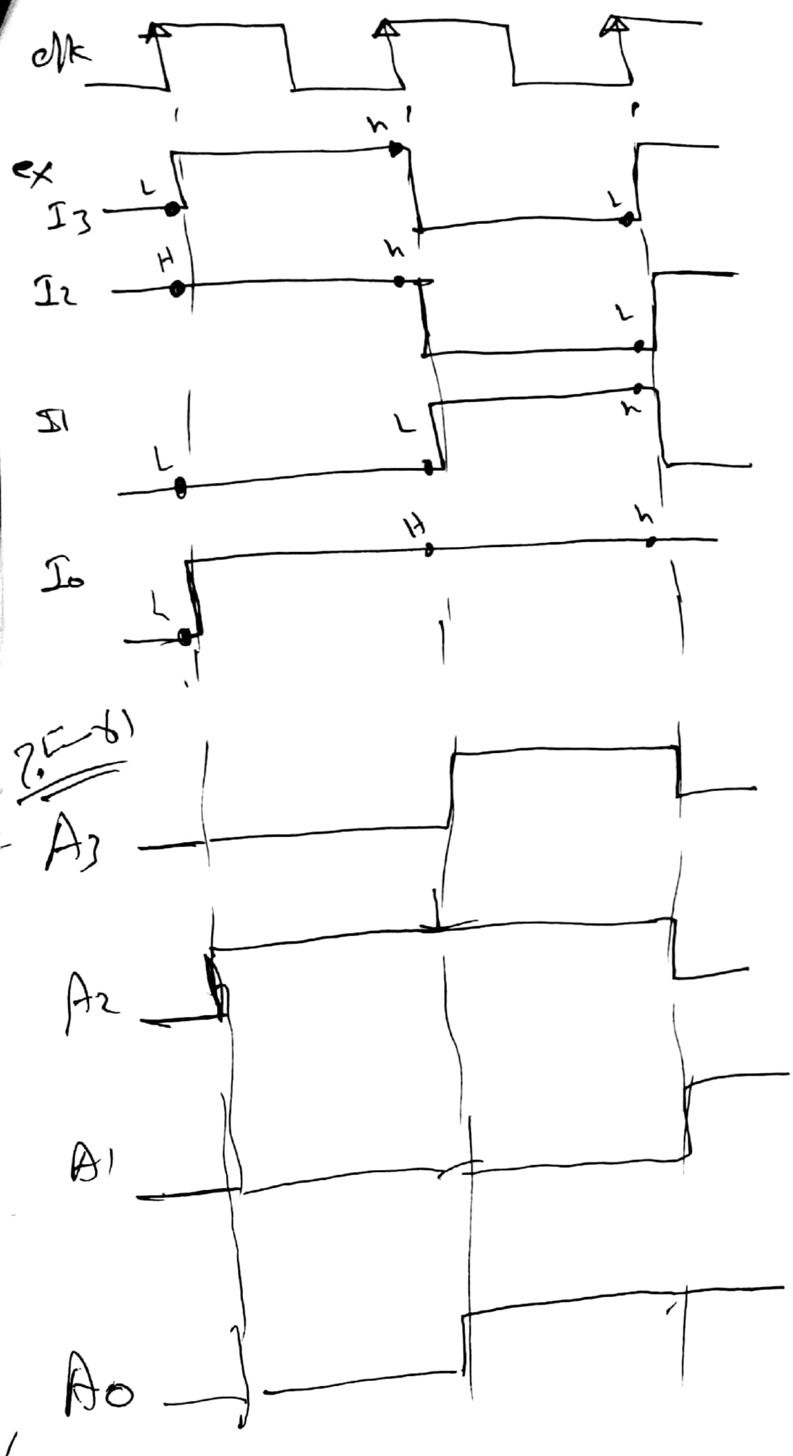
* Q0 تخرج على طول كأنها DFF عادية تقيس الدخل حسب الـ +ve edge

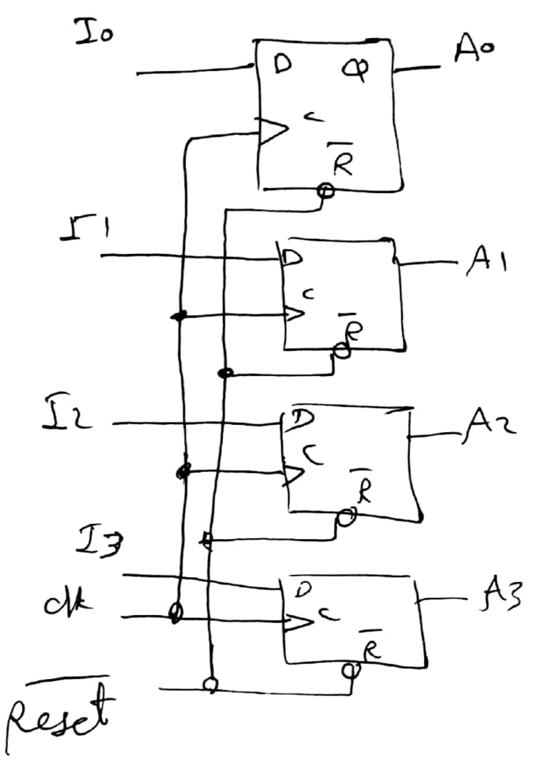
Ex (0110)

(If register contains initially 1s)



3- Parall in/ paralel out





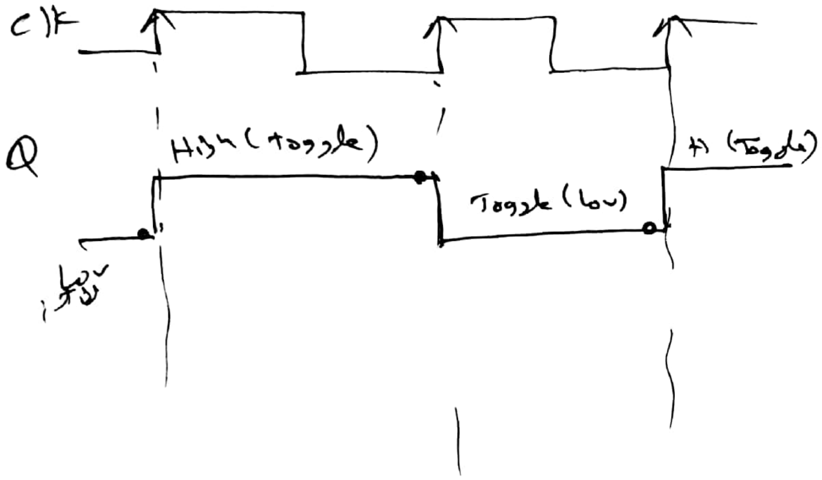
المخرج A_3 ، A_2 ، A_1 ، A_0
جملة مع I_3 ، I_2 ، I_1 ، I_0

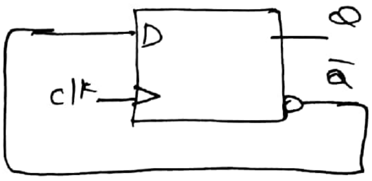
4- Frequency division

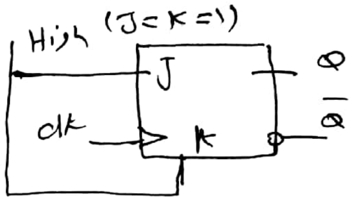
Used Registers to Reduce the Freq. of periodic wave form (Toggle FF or JK with high (J=K=1))

The o/p is Square wave with one half Freq. of the i/p

هنا Toggle يعني ما حسب الـ intial بغير الحالة (هنا لا تشترط بداية H)



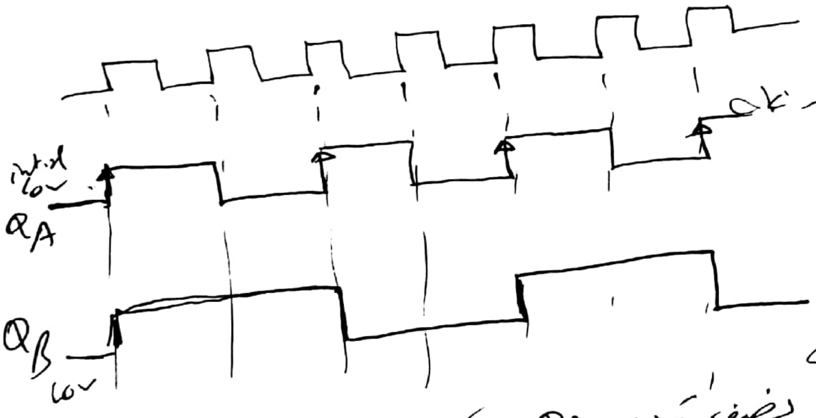


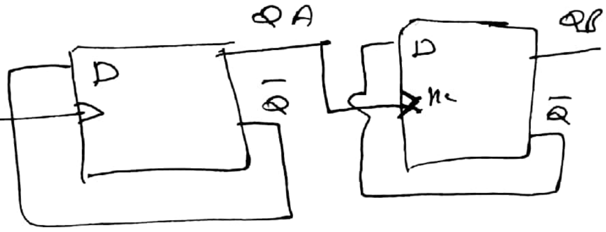


لاحظ ان D = Q̄ يعني عكس Q

initial input = Low

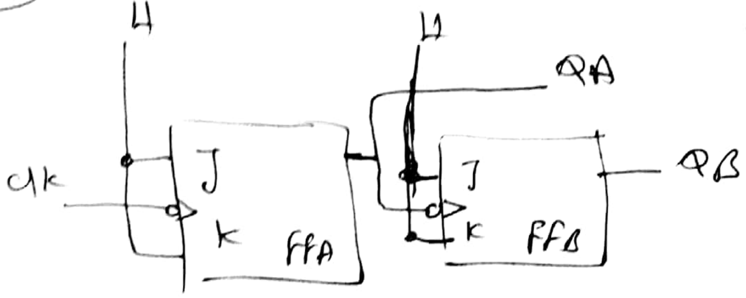
ex

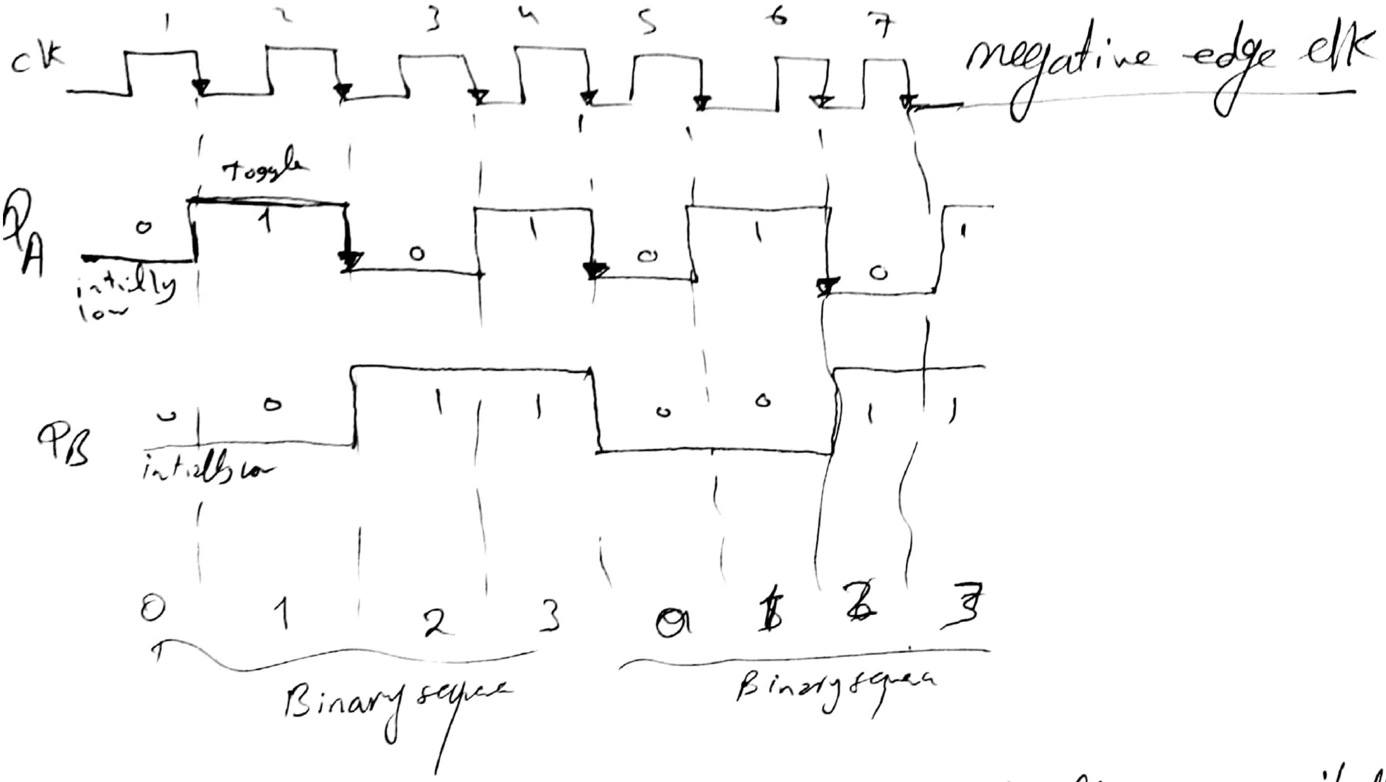




اخرج متاع FF الثاني QB نصف تردد QA و تردد QA نصف الـ clk

5- Counting





→ For negative edge Trigger JK ff. flip flops are initially Reset

→ ff A toggle on -ve going transition of each clock pulse. The Q o/p of FF A clocks FF B, So each time Q_A makes high to low transition, ff B toggles