به نام خدا

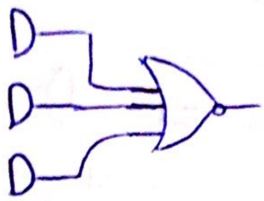
IC se14 Dr.Shaabaany

سؤال کوئیز:

تابع زیر را در نظر بگیرید:

$f = \sum m(۴,۶,۷,۸,۱۴,۱۵)$

این تابع را به صورت AND-NOR پیاده‌سازی کنید.



پیاده‌سازی بر اساس AOI و OAI:

- در مدارات مجتمع:
- پیاده‌سازی AND-OR-Invert ساده‌تر از And-OR است.
- پیاده‌سازی AOI همان AO دو طبقه است که در آن به جای گیت OR از گیت NOR استفاده شده.
- پیاده‌سازی OAI در OA " " " " " AND " NAND " ".
- لذا برای پیاده‌سازی تابع f به صورت AOI یا (AND-NOR) هم آن را به صورت AND-OR (SOP) پیاده‌سازی می‌کنیم.
- در چینش standard cell ها معمولاً از چینش AOI و OAI استفاده می‌شود.

AOI221 OAI321

پیاده سازی براساس NAND-AND , NOR-OR :

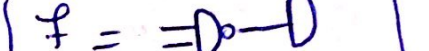
- برای پیاده سازی f براساس NAND-AND تابع f را بصورت SOP می نویسیم.

$$(\overline{ab}).(\overline{cd}) = \overline{(ab+cd)}$$

and-OR-NOT
SOP

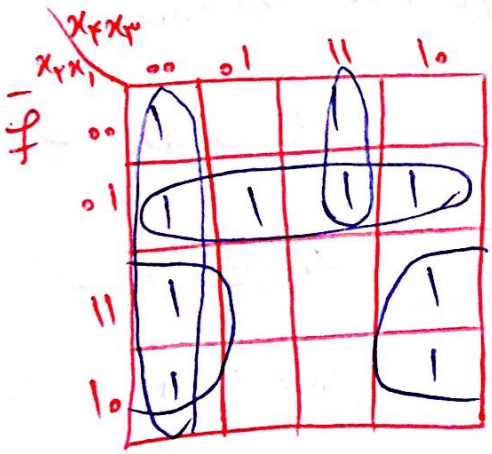
$\bar{f}$  $\Rightarrow$

$\Rightarrow \bar{f}$  $\Rightarrow$

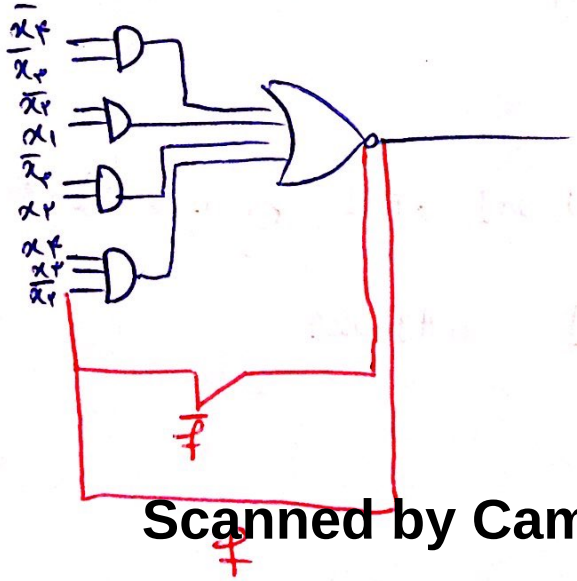
$f =$ 
NAND-AND

پاسخ سوال کوییز : $f = \sum m(4,6,7,8,14,15) \rightarrow$ AND-NOR

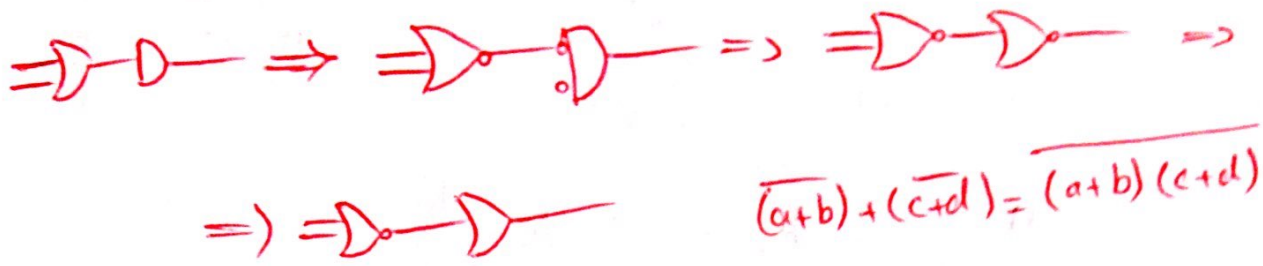
ابتدا باید $\bar{f}$ را به صورت AND-OR که همان SOP است بنویسیم سپس آن را Not کنیم.



$$\bar{f} = \bar{x}_4\bar{x}_3 + \bar{x}_2x_1 + \bar{x}_3x_2 + x_4x_3\bar{x}_2$$



← برای پیاده‌سازی براساس NOR-OR تابع f را بصورت POS می‌نویسیم.



$$\overline{(a+b)}+\overline{(c+d)}=\overline{(a+b)(c+d)}$$

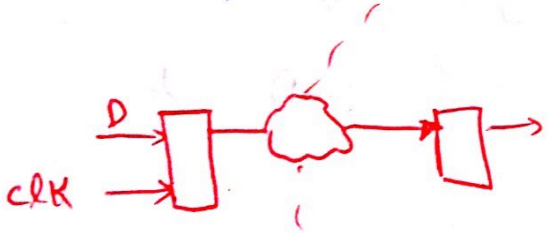
مثال:

تابع زیر را به صورت NAND-AND ، NOR-OR پیاده‌سازی کنید.

$$f=\sum m(0,1,2,3,4,8,9,12)$$

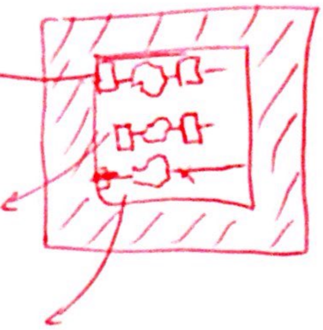
مسیر بحرانی : critical path

← طولانی‌ترین مسیر برای رسیدن به clk ورودی آخرین flip-flop را که به حافظه متصل است مسیر بحرانی می‌گویند که میزان زمان اتلافی را نشان می‌دهد. (مثال ۳۹ نفر در یک تورنامنت آماده‌ی حرکت و منتظر نفر چهلم می‌مانند تا سوار شود پس همه با هم حرکت کنند. نفر چهلم مانند مسیر بحرانی است)



تا n flip-flop (مسیر بحرانی رویدادی است که دیرتر می‌رسد.)

← فرض کنید یک IC داشته باشیم. بدترین حالت طراحی حالتی است که دسترسی برای delay در خارج آن توسط user وجود نداشته باشد که باعث محدود شدن وظایف شود.



دسترسی به بیرون دارد ممکن است یک delay توسط user به وجود آید

این مدل گرافی بهینه است (ایزوله کردن)

این مدل کلاً بدون کلاک وارد می‌شود!!!

* مثال خوب برای critical path : یک کارخانه‌ی خط تولید اتومبیل را در نظر بگیرید.

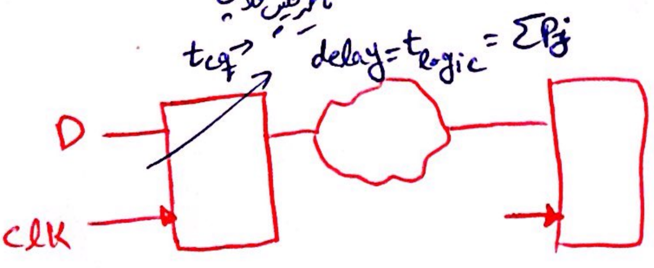
اگر مثلاً اتومبیل‌ها با سرعت خوب مثلاً در هر ساعت ۱۵ تا وارد شود و با همان سرعت درهای آن‌ها گذاشته شود و با همان سرعت رنگ شود ولی سیستم برق آن در هر ساعت مثلاً ۲ خودرو انجام شود، تأمین سیستم برق خودرو در این مثال critical path است.

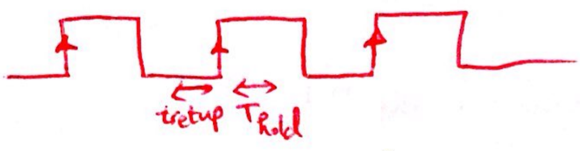
---

flip-flop :

* قبل و بعد از لبه‌ی بالارونده باید حالت data برایمان stable باشد.

(D-: ⇐ clk مثالی از رئیس جمهور است که قبل و بعد (ورود) آن همه چیز مرتب و stable است برای بازدید از دانشگاه مثلاً! D-:)





$$t_{clk} > t_{cq} + t_{logic} + t_{setup}$$

$$t_{cq,cd} + t_{logic,cd} \geq T_{hold}$$

t_{cd} ها مسیرهایی هستند که زود data ارسال می‌کنند پس باید زودتر به پایان برسد تا data های قبلی ارسال T_{hold} شوند سپس data های جدید برسند. ← قبل از T_{hold} data های پشت flip-flop نباید تغییر کنند.