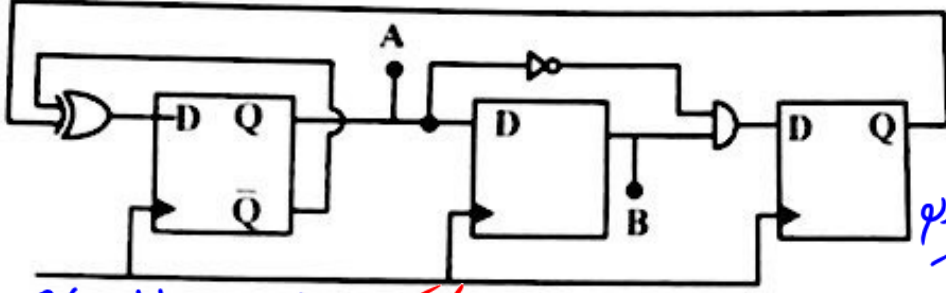


۷۰- فرض کنید تأخیر گیت‌های پایه برابر تعداد ورودی‌های آن‌ها و تأخیر گیت NOT برابر صفر باشد. اگر گیت XOR به صورت

دوسطحی NAND-NAND پیاده‌سازی شود، تأخیر یک جمع‌کننده ۴ بیتی Carry Lookahead Adder (CLA) با احتیاط به حل تمامی اساتید محترم، نظریه گترین ۳ (۱۴ ns) است. اگر با توجه به یکی از کتب مرجع چند ثانیه است؟

- ۱۸ (۱) مدار را رسم کنید و تأخیرها را لحاظ کنید گترین ۴ (۱۷ ns) بدست می‌آید اما طرح اصلاً یعنی از لیت‌های ۸۰۷ را با لیت دیگری جایگزین کرده و از ۸۰۷ها در جای دیگری استفاده کرده تا تأخیر را کمتر کند و بدین صورت به گترین ۱۲ (۲)
- ۱۴ (۳) ۱۴ (۳) رسیده که بهترین گترین است. بنده نباید دلیل حل تا بل این سؤال را در این فایل قرار می‌دهم و نظریه‌های اینجا نب این است که طرح گترین صحیح را (۳) اعلام می‌کنم هر چند اگر اعلام شده جای تکلیف نیست
- ۱۶ (۴) * سؤال جالبی نبود. بیشتر سؤال دانشگاهی بود تا کنکوری زیرا از زمان زیادی را طلب می‌کرد.

۷۱- در مدار زیر، اگر در ابتدا، محتوای هر سه فلیپ‌فلاپ برابر $ABC = 000$ باشد، به ترتیب تغییرات مدار چگونه خواهد بود؟ **بسیار آسان**

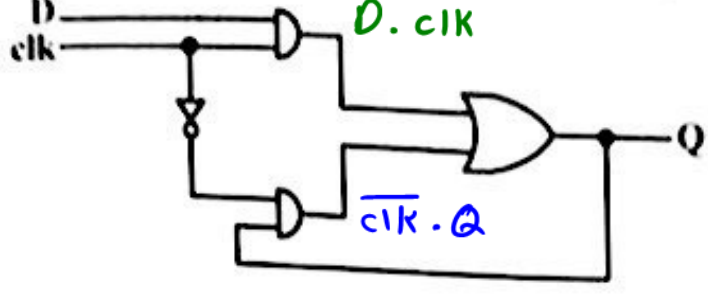


بسیار آسان است مشابه با این تست از تکنولوژی راننده و دکترای سال‌های پیش به روش سیمولیشن دستی حل کردم

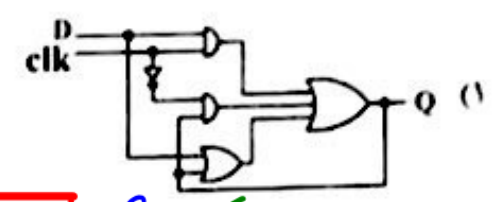
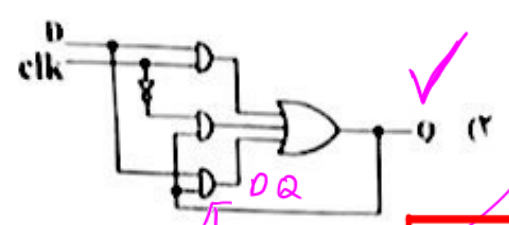
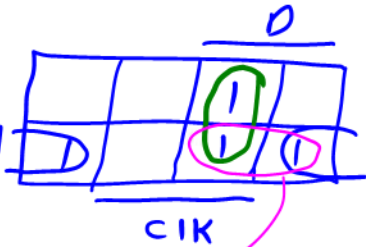
$a(t+1) |_{0 \leq t} = 0 \Rightarrow$ **بین فقط ورودی‌های ۰ را به دست می‌آوریم**

000	→	100	→	010	→	101	→	110	→	010	...
000	→	101	→	110	→	010	→	100	→	110	...
000	→	100	→	010	→	110	→	100	→	100	...
000	→	100	→	110	→	010	→	101	→	010	...

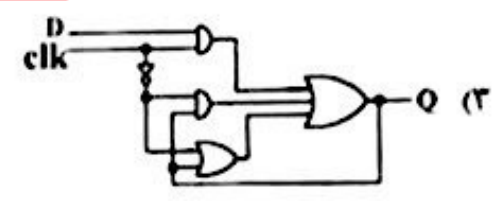
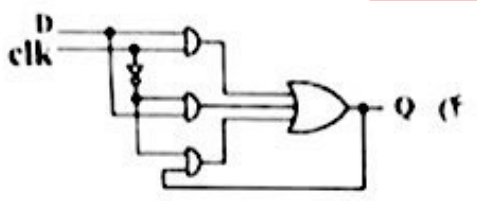
۷۲- لچ (فلیپ‌فلاپ) D شکل زیر دارای هزارد است. چگونه این هزارد را رفع کنیم؟



بسیار بسیار آسان



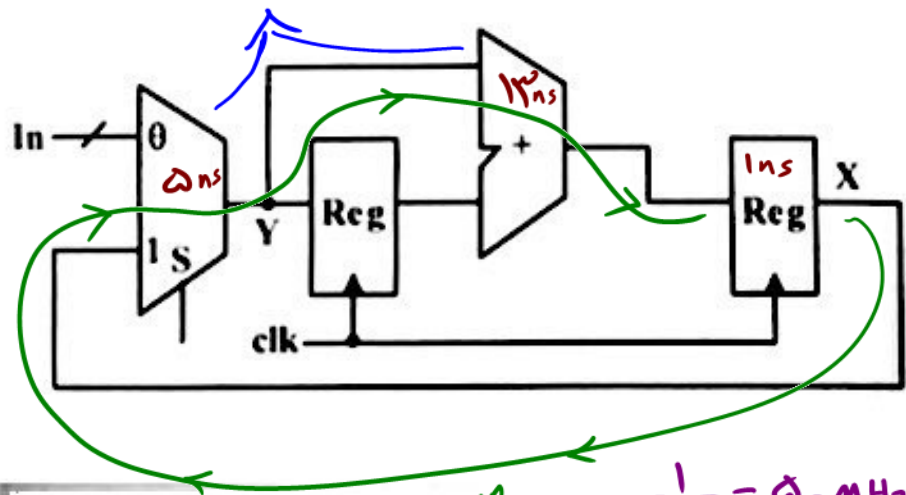
* این مثال در جزوه **عیناً** حل شده یعنی **تسا به ۱۰۰ درصد**



۷۲- فرض کنید در مدار زیر، تأخیر مالتی پلکسر برابر ۵ نانوثانیه، تأخیر جمع کننده برابر ۱۳ نانوثانیه و پارامترهای زمانی

رجیسترها به صورت $T_{clk \rightarrow q} = 1ns$ و $T_{setup} = 1ns$ باشد. ماکزیمم فرکانس کاری مدار چند مگاهرتز است؟

combinational



آسان

- ۱۴۰ (۱)
- ۵۰ (۲) ✓
- ۵۲/۵ (۳)
- ۶۶/۴ (۴)

چندین مثال از این مبحث و حساب به آن در فیزیک حل شده

T_{clk}

$$\frac{1}{T_{clk}} = 5 \text{ MHz}$$

$T_{clk \rightarrow q}$	T_{comb}	T_{setup}
1	18	

مجموعه مهندسی برق - کد (۱۳۵۱)

733E

صفحه ۲۱

میسر مورد نظر

۷۴- برای توصیف کنترلر (cu) نشان داده شده به verilog. قسمت نقطه چین در کد، با کدام مورد باید پر شود؟

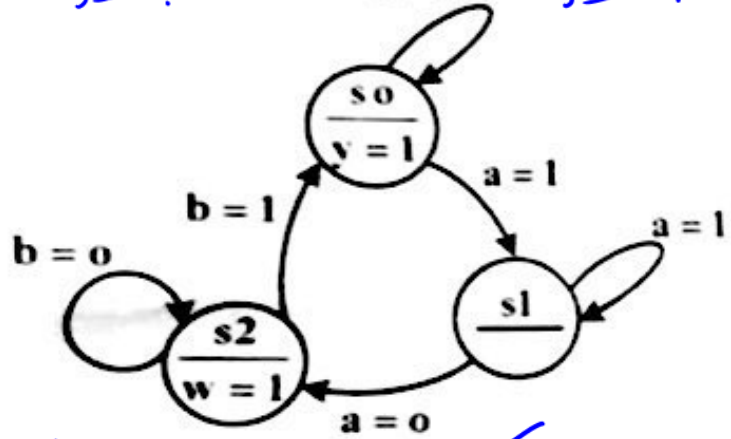
```
Module cu (input clk , rst, a, b, output reg y, w);
reg [1:0] P , N;
.....
.....
case (P)
0: begin N = a? 1: 0; y = 1; end
1: N = a? 1: 2;
2: begin N = b? 0: 2; w = 1; end
Default: N = 0;
end case
end
always@ (posedge clk, posedge rst)
if (rst) p <= 2' b 0; else p <= N;
end module
```

از مابین حالت نیز گدی بررسی شده آسان

این always مربوط به مدار ترکیبی است بارها گفتم در مدارات ترکیبی نسبت حساسیت شامل تمامی ورودی های آن است. ورودی های

کن مدار ۴ عدد است که clk و reset آن برای فلپ فلپ ها کاربرد دارد و ورودی های مدار ترکیبی a و b و نیز خروجی فلپ فلپ است. خروجی فلپ فلپ حالت فعلی یا همان P است

```
always @ (a, b) begin
y = 0 , w = 0;
end
always @ (P) begin
N = 0;
end
always @ (a, b, P) begin
{y, w} = 2'b0;
end
always @ (clk, rst) begin
N = 0;
end
```

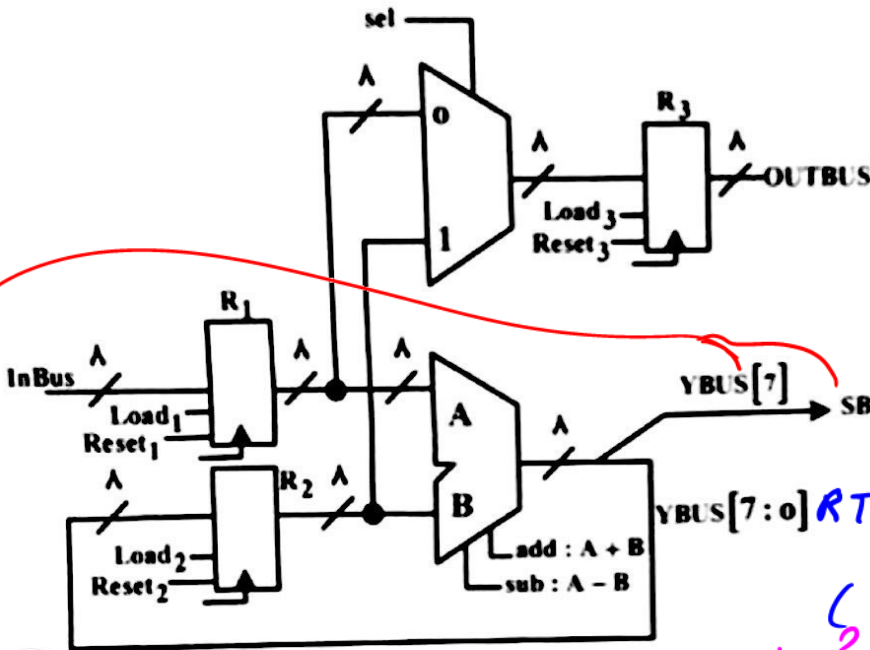


صفت ترکیبی چه موقع تغییر می کند؟ وقتی ورودی ها تغییر کنند. این نسبت حساسیت آن باید شامل P و w و a باشد

در سه کلاک پشت سر هم داده های A، B و C بر روی InBus قرار می گیرد (1:A, 2:B, 3:C). در این مدار ALU دو عمل نشان داده را انجام می دهد. برای این که پس از چهار کلاک $Min(C, (A + B))$ بر روی OUTBUS قرار گیرد. سیگنال های کنترلی کدام است؟

متوسط

حل این بخش را به صورت مداری ارائه می دهیم و به بعضی اینده بتوانیم ضمیمه حل تست های این فایل را در آپارات بارگذاری خواهیم کرد. در آن ضمیمه بازبان RTL آشنای سوئیچ در هر قطعه های دستورها با هم اجزای نمونه



$R_1 = A$ و $R_2 = 0$
 $R_1 = B$ و $R_2 = A + 0 = A$
 $R_1 = C$ و $R_2 = A + B$

- CLK1: Load 1; Reset 2;
- CLK2: Load 1; add; Load 2;
- CLK3: Load 1; add; Load 2;
- CLK4: Load 3; if(SB) sel = 0; sub;

که تعریف انجام می شود و در SB یا بیت MSB در YBUS
 یک بود یعنی حاصل منفی است لذا C کوپلتر از $A+B$
 است و با توجه به اینکه $SB=1$ است که $sel=0$ شود
 وضوح ما کس انتخاب شده و در هر R ذخیره
 می شود چون add 3 اهم فعال شده

- CLK1: Load 1; add;
- CLK2: Load 1; Load 3;
- CLK3: if(SB) sel = 0;
- CLK4: Load 3; sub;

- CLK1: Load 1;
- CLK2: Load 2; add;
- CLK3: if(SB) sel = 1;
- CLK4: Load 3; sub;

- CLK1: Load 1; Reset 1; Load 2
- CLK2: Load 2; sub
- CLK3: Load 1; add
- CLK4: Load 3; if (SB) sel = 1;