

به نام خدا

دانشکده برق و کامپیوتر دانشگاه تهران

گزارش کار پروژه 4 سیستم های دیجیتال 1

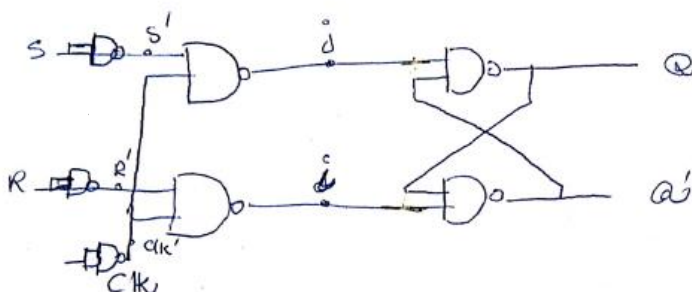
استاد : دکتر نوابی

دانشجو : فاطمه نائینیان

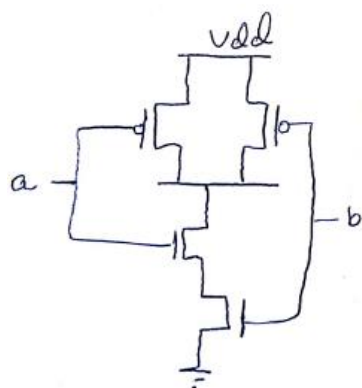
شماره دانشجویی : 810198479

بهار 1400

سوال اول



clk	S	R	Q
0	1	1	Q
0	1	0	0
0	0	1	1
0	0	0	Q
0	-	-	Q



$$t_{o1} = \max(t_{o1pmos}, t_{o2nmos}) = 1ns$$

$$t_{o0} = \max(t_{o0nmos}, t_{o2pmos}) = 1ns$$

دیلی های یک nand برای t_{o1}, t_{o0} 8 نانوثانیه به دست می آید. و worst case delay در این سوال 32 نانوثانیه خواهد بود.

```

1 | `timescale 1ns/1ns
2 | module srlatch(input s,r,clk,output q);
3 |   wire s0,r0,clk0,i,j,q0;
4 |   nand #8 T1(s0,s,s);
5 |   nand #8 T2(r0,r,r);
6 |   nand #8 T3(clk0,clk,clk);
7 |   nand #8 T4(j,s0,clk0);
8 |   nand #8 T5(i,r0,clk0);
9 |   nand #8 T6(q0,i,q);
10 |  nand #8 T7(q,j,q0);
11 | endmodule

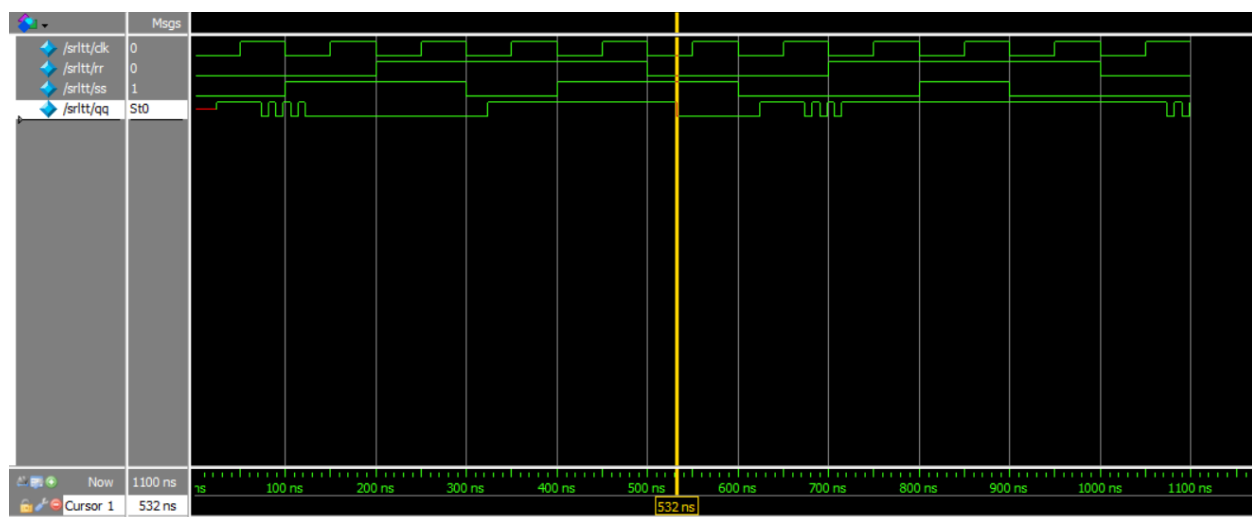
```

سوال دوم

```

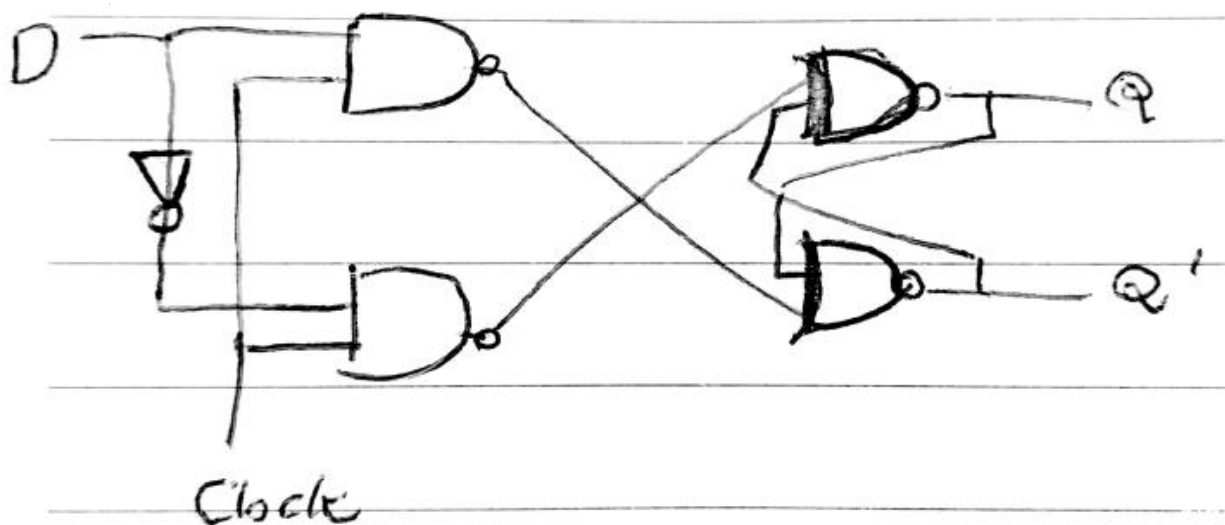
1  `timescale 1ns/1ns
2  module srltt();
3      logic rr=0;
4      logic ss=0;
5      logic clk=0;
6      wire qq;
7      srlatch T(ss,rr,clk,qq);
8      always #50 clk <= ~clk;
9      initial begin
10         #100 ss=1;
11         #100 rr=1;
12         #100 ss=0;
13         #100 ss=1;
14         #100 rr=0;
15         #100 ss=0;
16         #100 rr=1;
17         #100 ss=1;
18         #100 ss=0;
19         #100 rr=0;
20         #100 $stop;
21     end
22 endmodule

```



سوال سوم

اگر دو ورودی srlatch سوال قبل را با یک گیت نات بهم متصل کنیم dlatch به دست می آید.



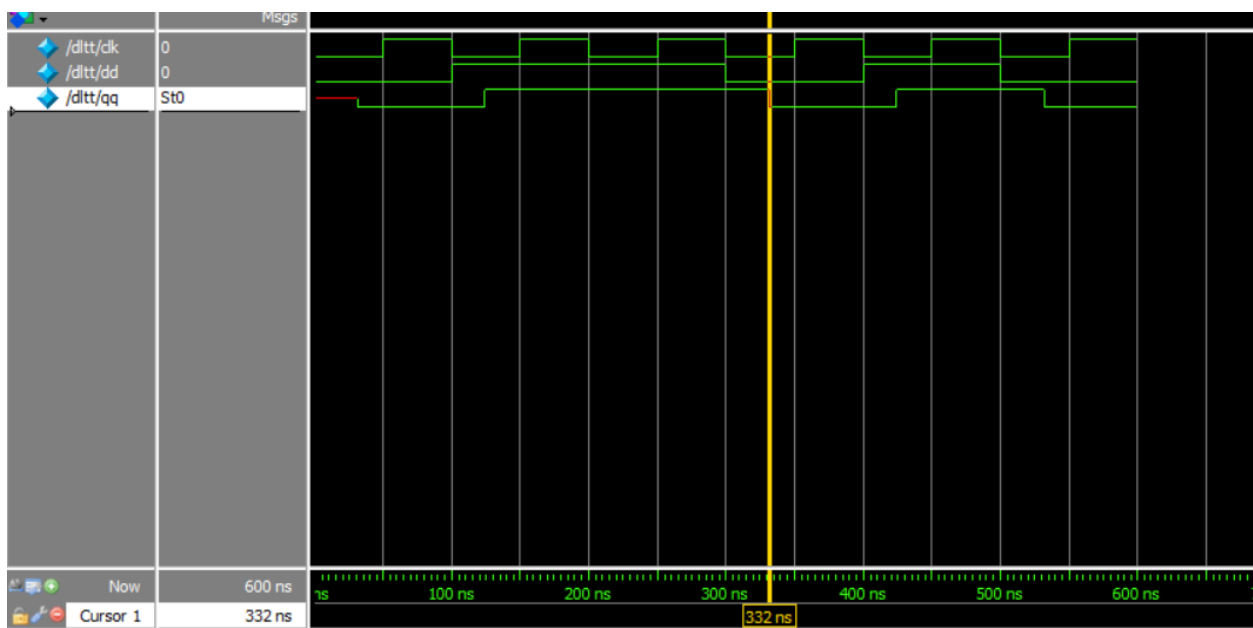
Ln#	
1	<code>`timescale 1ns/1ns</code>
2	<code>module dlatch(input d,clk,output q);</code>
3	<code> wire d0;</code>
4	<code> not n(d0,d);</code>
5	<code> srlatch T(d0,d,clk,q);</code>
6	<code>endmodule</code>
7	

تست بنچ

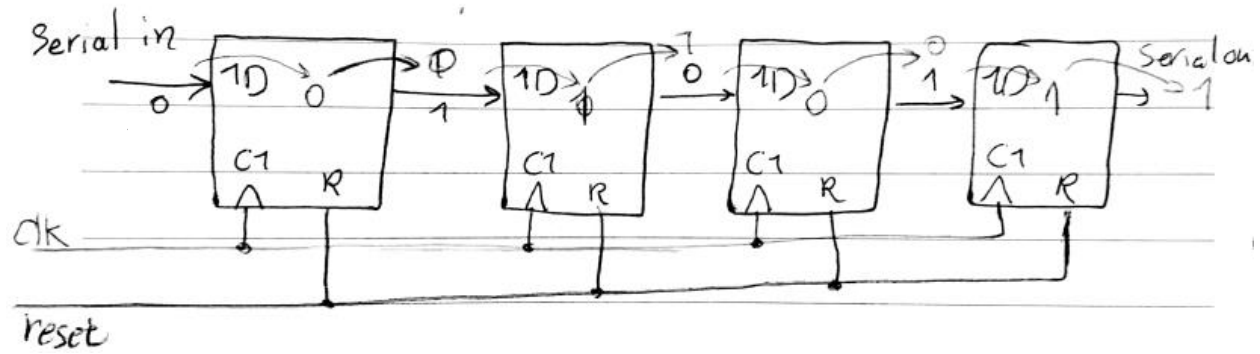
```

Ln#
1  `timescale 1ns/1ns
2  module dlтт();
3      logic dd=0;
4      logic clk=0;
5      wire qq;
6      dlatch T(dd,clk,qq);
7      always #50 clk <= ~clk;
8      initial begin
9          #100 dd=1;
10         #100 dd=1;
11         #100 dd=0;
12         #100 dd=1;
13         #100 dd=0;
14         #100 $stop;
15     end
16 endmodule

```



سوال چهارم

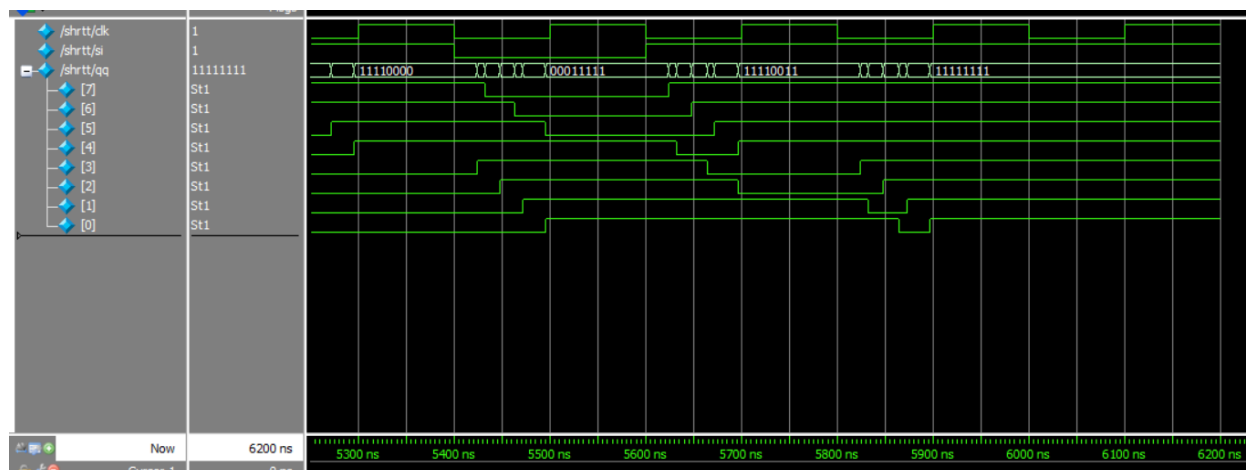


Ln#	
1	<code>`timescale 1ns/1ns</code>
2	<code>module shr(input si,clk,output [7:0]so);</code>
3	<code>wire [8:0]y;</code>
4	<code>assign y[8]=si;</code>
5	
6	<code>genvar b;</code>
7	<code>generate</code>
8	<code>for(b=8;b>0;b=b-1) begin</code>
9	<code> d latch U(y[b],clk,y[b-1]);</code>
10	<code>end</code>
11	<code>endgenerate</code>
12	<code>assign so[7:0]=y[7:0];</code>
13	<code>endmodule</code>
14	

سوال پنجم

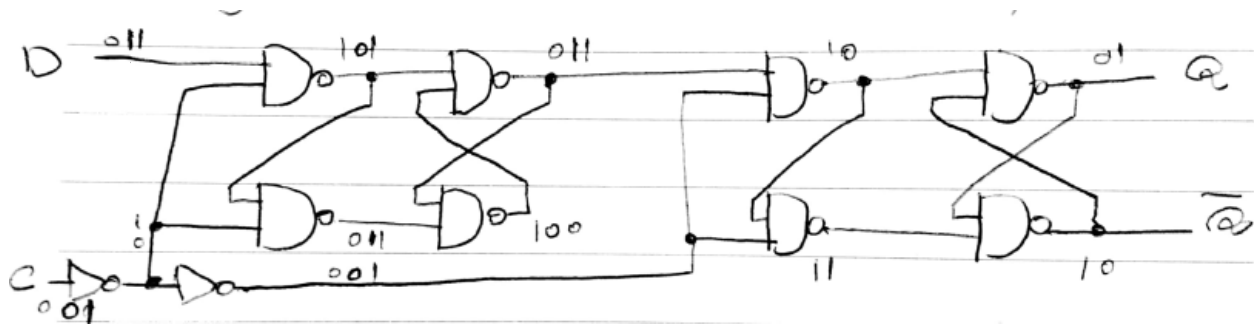
همانطور که در کلاس مطرح شد در این حالت تا زمانی که کلاک 1 شود خروجی میتواند تغییر کند و مثل همان مثال در سر کلاس تا وقتی در باز باشد نمونه ها عبور میکنند پس تا زمانی که کلاک صفر است میتواند چندین تغییر را عبور دهد که مطلوب نیست.

```
Ln# |  
1 | `timescale 1ns/1ns  
2 | module shrtt();  
3 |   logic si=0;  
4 |   logic clk=0;  
5 |   wire [7:0]qq;  
6 |   shr T(si,clk,qq);  
7 |   always #100 clk <= ~clk;  
8 |   initial begin  
9 |     repeat(30) #200 si=$random();  
10 |    #200 $stop;  
11 |   end  
12 | endmodule  
13 |
```



سوال ششم

با کمک دو dlatch میتوان شکل زیر را ساخت.



Ln#	
1	module msd(input d,clk,output q);
2	wire clk0,Q1;
3	not n(clk0,clk);
4	dlatch t1(d,clk0,Q1);
5	dlatch t2(Q1,clk,q);
6	
7	endmodule
8	

سوال هفتم

Ln#	
1	module rmsd(input d,r,clk,output q);
2	wire j;
3	and (j,d,~r);
4	msd t(j,clk,q);
5	endmodule
6	

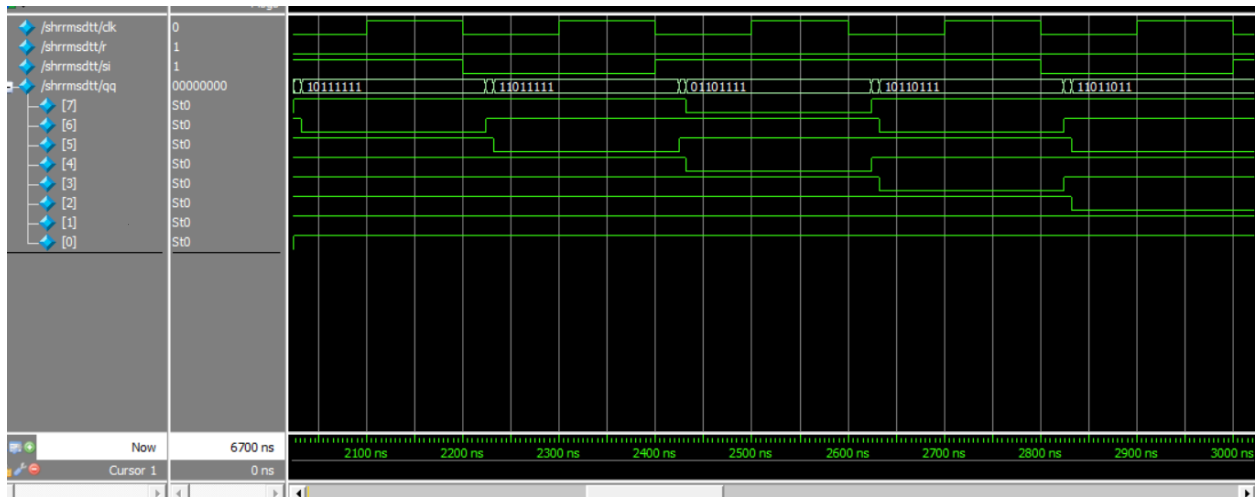
سوال هشتم

در این بخش چون از master slave استفاده کردیم با هر کلاک یک تغییر را عبور میدهد. ولی در سوال 4 هر تغییری که در زمان ان کلاک اتفاق بیوفتد را عبور میدهد.

Ln#	
1	<code>`timescale 1ns/1ns</code>
2	<code>module shrrmsd(input si,clk,r,output [7:0]so);</code>
3	<code>wire [8:0]y;</code>
4	<code>assign y[8]=si;</code>
5	
6	<code>genvar b;</code>
7	<code>generate</code>
8	<code>for(b=8;b>0;b=b-1) begin</code>
9	<code> rmsd U(y[b],r,clk,y[b-1]);</code>
10	<code>end</code>
11	<code>endgenerate</code>
12	<code>assign so[7:0]=y[7:0];</code>
13	<code>endmodule</code>
14	

تست بنچ

Ln#	
1	<code>`timescale 1ns/1ns</code>
2	<code>module shrrmsdtt();</code>
3	<code>logic si=0;</code>
4	<code>logic r=0;</code>
5	<code>logic clk=0;</code>
6	<code>wire [7:0]qq;</code>
7	<code>shrrmsd T(si,clk,r,qq);</code>
8	<code>always #100 clk <= ~clk;</code>
9	<code>initial begin</code>
10	<code> repeat(30) #200 si=\$random();</code>
11	<code> #200 r=1;</code>
12	<code> #500 \$stop;</code>
13	<code>end</code>
14	<code>endmodule</code>



سوال نهم

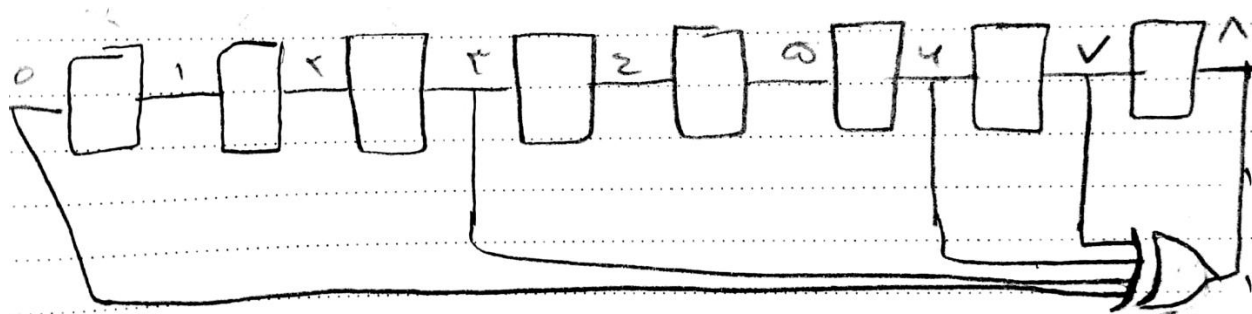
```

Ln# |
1 | `timescale 1ns/1ns
2 | module shralways(input si,clk,r,output logic [7:0]so);
3 |
4 | always @(posedge clk ) begin
5 |     if(r) so <= 8'd0;
6 |     else so<={si,so[7:1]};
7 | end
8 |
9 | endmodule

```

سوال دهم

دوره تناوب ان 63 کلاک است و بعد 63 کلاک دوباره هشت تا صفر روی خروجی می افتد. مطابق اطلاعات بدست آمده از اینترنت برای یک LFSR هشت بیت دوه تناوب حداکثر 255 می شود که با در نظر گرفتن فیدبک های مختلف این تناوب کمتر می شود. با توجه به فیدبک های این سوال تناوب 63 خواهیم داشت.



Ln#	
1	module lfsr(input r, clk , output [7:0]po);
2	wire sii;
3	assign sii=((po[0]^po[3])^po[6])^po[7]);
4	shralways t(~sii,clk,r,po);
5	endmodule
6	

تست بنچ

Ln#	
1	`timescale 1ns/1ns
2	module lfsrtt();
3	logic r=1;
4	logic clk=0;
5	wire [7:0]qq;
6	lfsr T(r,clk,qq);
7	always #100 clk <= ~clk;
8	initial begin
9	#200 r=0;
10	#5000 \$stop;
11	end
12	endmodule
13	

