

به نام خدا

آزمایش ۲

طراحی مدارهای اولیه Multiplexer 4to1, Encoder 4to2, Decoder 2to4 و Comparator ۳ بیتی

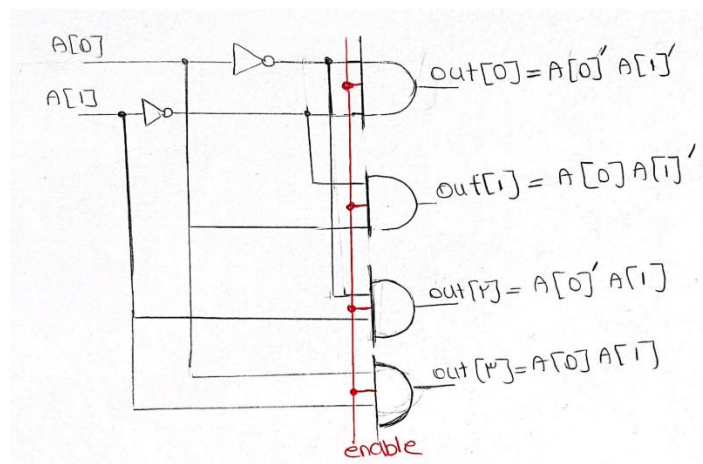
روژینا کاشفی-هلیا سادات هاشمی پور

=====

Decoder(2x4)

برای رسم دیکودر 2x4 به یک ورودی ۲ بیتی و خروجی ۴ بیتی نیاز داریم که به صورت vector تعریف کردیم و یک enable گذاشتیم. اگر enable فعال باشد مدار فعال و خروجی ها فعال می شوند در غیر اینصورت صرف نظر آنکه ورودی چی باشد تمام خروجی ها ۰ می شود.

طراحی مدار Decoder 2x4 :




در جدول زیر نحوه کار کردن دیکودر را مشاهده می کنیم:

Decimal	Enable	A[1]	A[0]	Output[0]	Output[1]	Output[2]	Output[3]
	0	X	X	0	0	0	0
0	1	0	0	1	0	0	0
1	1	0	1	0	1	0	0
2	1	1	0	0	0	1	0
3	1	1	1	0	0	0	1

خروجی کد نوشته شده:

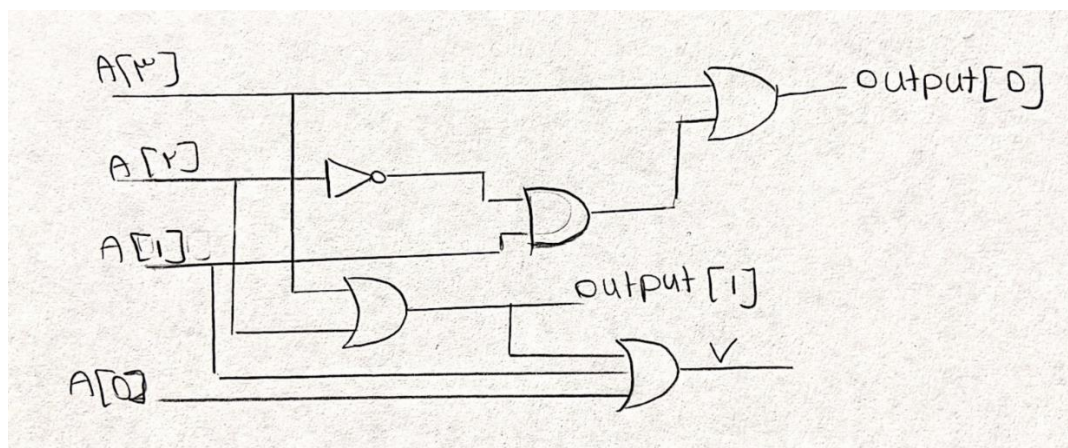
همانطور که مشاهده می کنیم خروجی کد مانند جدول فوق است.

		Msgs																								
 /decoder2_4_tb/a	2b11	00				01			10			11			00			01			10			11		
 /decoder2_4_tb/enable	1																									
 /decoder2_4_tb/output	4b1000	0000													0001			0010			0100			1000		

Encoder (4x2)

برای رسم encoder 4x2 باید یک vector ۴ بیتی به عنوان ورودی دهیم و خروجی یک vector ۲ بیتی است. ما در این قسمت یک priority encoder تعریف کردیم که زمانی که بیت v(valid) صفر است صرف نظر از آنکه ورودی چیست خروجی مدار نامشخص است و همچنین اگر v یک باشد هر بیت صرف نظر از مقدار بیتهای قبلیش مقدارش ۱ می شود و مقدار بیت های بعدیش ۰ می شود.

طراحی مدار Encoder 2x4 :



در جدول زیر نحوه کار کردن priority encoder 4x2 را مشاهده می کنیم:

Decimal	V	A[3]	A[2]	A[1]	A[0]	Output[1]	Output[0]
	0	0	0	0	0	X	X
0	1	0	0	0	1	0	0
1	1	0	0	1	X	0	1
2	1	0	1	X	X	1	0
3	1	1	X	X	X	1	1

خروجی کد نوشته شده:

همانطور که مشاهده می کنیم خروجی کد مانند جدول فوق است.

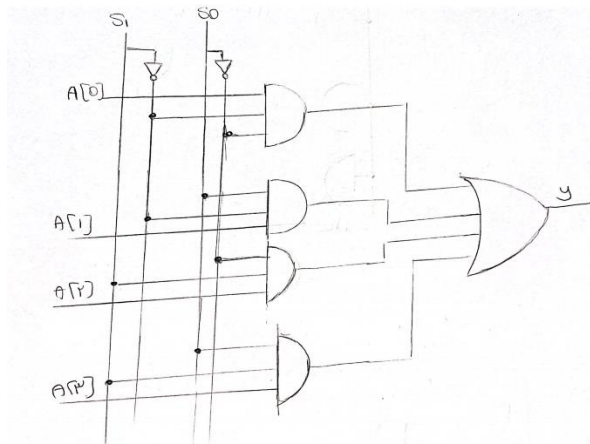
Multiplexer

هدف ما طراحی یک مالتی پلکسر 16x1 است برای اینکار ابتدا یک مالتی پلکسر 4x1 طراحی می کنیم و سپس با استفاده از 5 مالتی پلکسر 4x1 یک مالتی پلکسر 16x1 میسازیم.

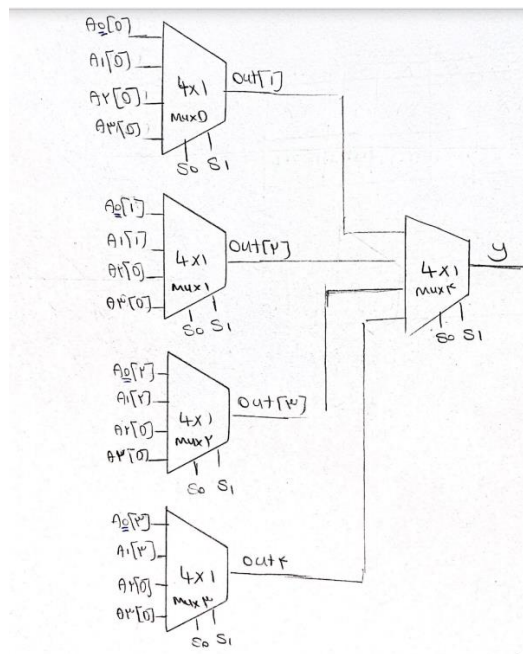
جدول یک مالتی پلکسر 4x1:

S1	S0	Y
0	0	Input[0]
0	1	Input[1]
1	0	Input[2]
1	1	Input[3]

طراحی یک مالتی پلکسر 4x1:



طراحی یک مالتی پلکسر 16x1:



- دقت شود که خط select ها یکسان است .
- ما در طراحی خود به جای دادن یک وکتور 16 بیتی از 4 vector 4 بیتی استفاده کردیم به صورتی که a_0 وکتور مربوط به بیت صفرم هر مالتی پلکسر ، a_1 وکتور مربوط به بیت 1 هر مالتی پلکسر ، a_2 وکتور مربوط به بیت 2 هر مالتی پلکسر و a_3 وکتور مربوط به بیت 3 هر مالتی پلکسر است.

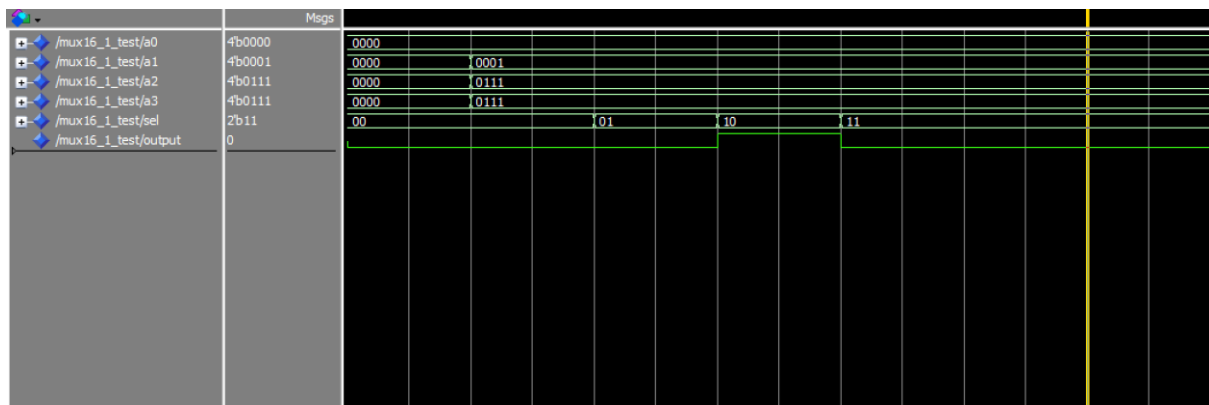
خروجی کد نوشته شده:

مشاهده می کنیم اگر خط **select** ۰ باشد بیت ۰ ام هر مالتی پلکسر به ماکس اخر منتقل می شود و در نهایت بیت ۰ **a0** به عنوان خروجی نشان داده می شود. (خروجی ۰)

اگر خط **select** ۱ باشد بیت ۱ ام هر مالتی پلکسر به ماکس اخر منتقل می شود و در نهایت بیت ۱ **a1** به عنوان خروجی نشان داده می شود. (خروجی ۰)

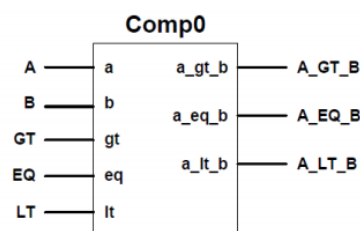
اگر خط **select** ۲ باشد بیت ۲ ام هر مالتی پلکسر به ماکس اخر منتقل می شود و در نهایت بیت ۲ **a2** به عنوان خروجی نشان داده می شود. (خروجی ۱)

اگر خط **select** ۳ باشد بیت ۳ ام هر مالتی پلکسر به ماکس اخر منتقل می شود و در نهایت بیت ۳ **a3** به عنوان خروجی نشان داده می شود. (خروجی ۰)

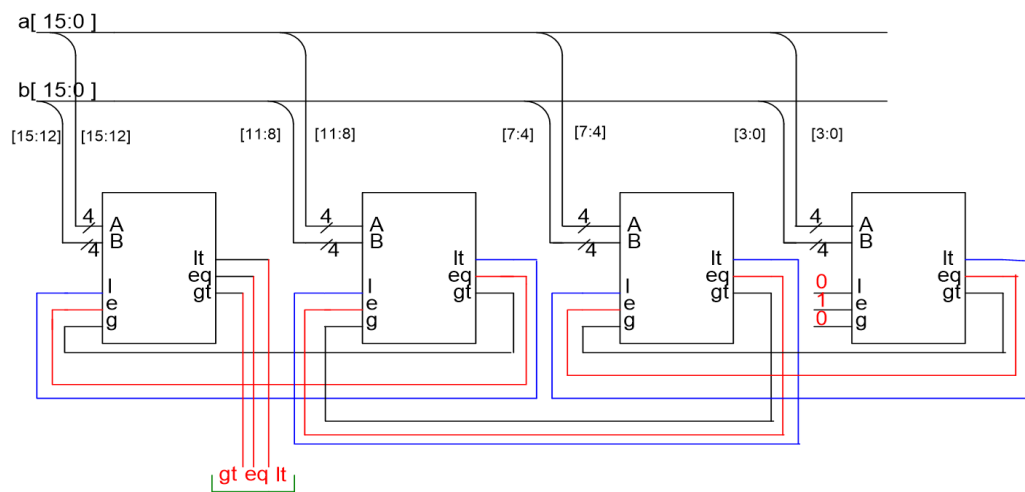


Comparator

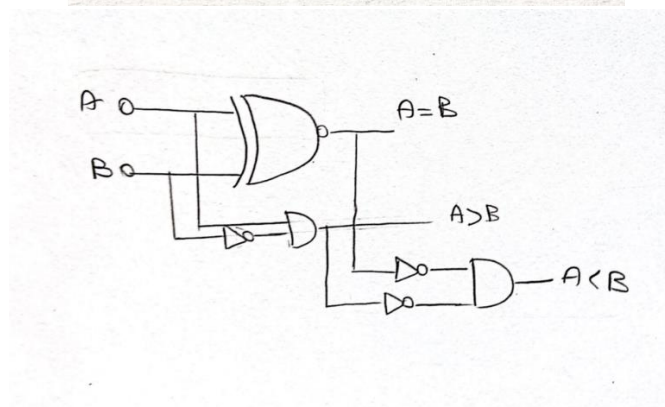
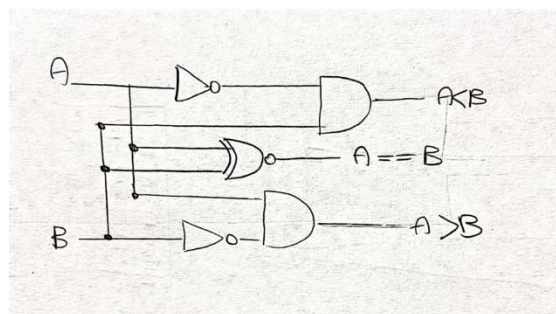
ابتدا باید یک **comparator** ۱ بیتی طراحی کنیم که ۲ ورودی دارد و نشان دهد در صورتی که اگر ورودی ها ۱ بیت بود چگونه ان را مقایسه کند اما برای طراحی **comparator** های بیش از ۱ بیت باید ۵ ورودی داشته باشیم. ۲ تا از ورودی ها بیت های برای مقایسه کردن و ۳ ورودی دیگر برای نشان دادن برابر یا بزرگتر با کوچکتر بودن بیت های قبلی است.



که مثالی از آن را میبینیم:

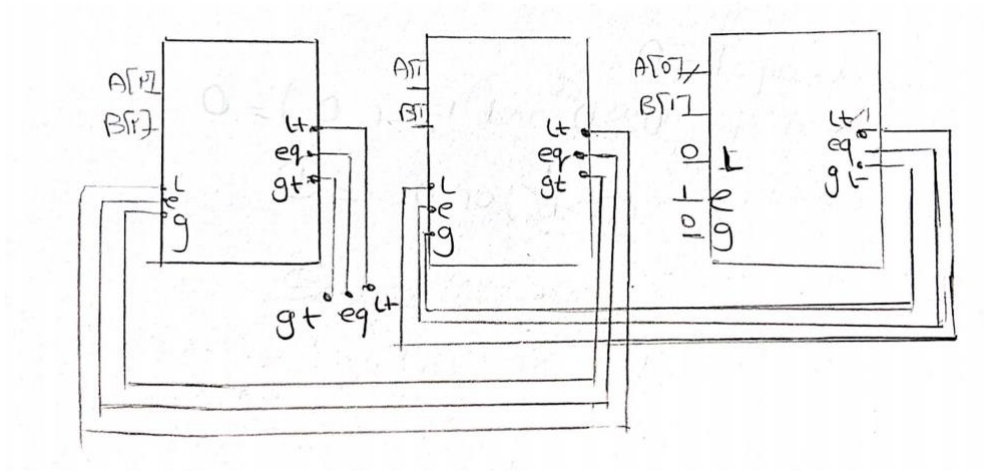


طراحی comparator ۱ بیت:

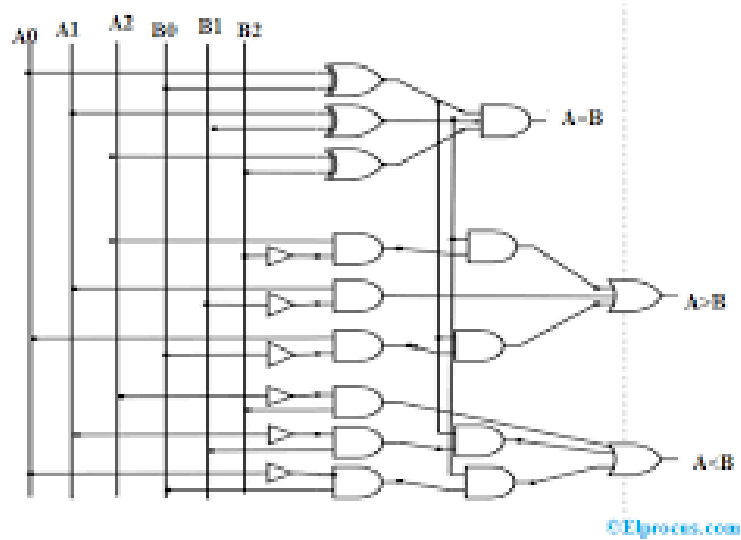


دو طراحی فوق یکسان است.

طراحی comparator ۳ بیت:



اگر بدون 1 bit comparator می خواستیم انجام دهیم به صورت زیر می بود.











©Elprocs.com

خروجی کد نوشته شده:

همانطور که مشاهده می کنیم زمانی که ورودیها ۰۰۰ و ۰۰۱ است $a_lt_b=1$

زمانی که ورودی های ۰۱۰ و ۰۱۱ است $a_gt_b=1$

زمانی که ورودی ها ۱۰۰ و ۱۰۱ است $a_eq_b=1$

	Msgs	
 /comperator3_tb/a	3'b100	000
 /comperator3_tb/b	3'b100	000
 /comperator3_tb/lt	0	010
 /comperator3_tb/gtt	0	100
 /comperator3_tb/eq	1	001
 /comperator3_tb/a_gt_b	0	100
 /comperator3_tb/a_eq_b	1	
 /comperator3_tb/a_lt_b	0	