



به موارد زیر توجه کنید:

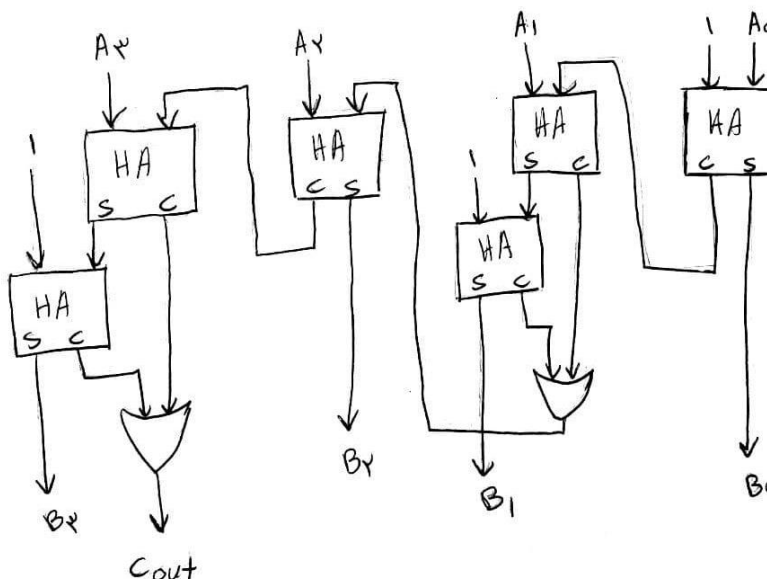
- ۱- حتما نام و شماره دانشجویی خود را روی پاسخ نامه بنویسید.
- ۲- در حل سوالات به نوشتن جواب آخر اکتفا نکنید. همه مراحل میانی را هم بنویسید.
- ۳- کل پاسخ تمرینات را در قالب یک فایل pdf با شماره دانشجویی خود نام گذاری کرده در سامانه CW بارگذاری کنید.
- ۴- در صورت مشاهده هر گونه مشابهت نامتعارف هر دو (یا چند) نفر کل نمره این تمرین را از دست خواهند داد.
- ۵- هر ساعت تاخیر در ارسال تمرین ۲ درصد از نمره آن را کم خواهد کرد و حداکثر تاخیر مجاز ۲۴ ساعت است.

### سوالات:

- ۱- (۲ نمره) با کمک شش Half adder و کمترین تعداد گیت ممکن، مداری طراحی کنید که یک عدد چهاربیتی را به عنوان ورودی بگیرد و خروجی آن، حاصل چهاربیتی تفریق عدد ۵ از عدد ورودی (در سیستم نمایش اعداد مکمل ۲) باشد.

### پاسخ سوال (۱)

برای این کار باید عدد چهار بیتی ورودی  $(A_3A_2A_1A_0)$  را با مکمل دو عدد ۵ (1011) جمع کنیم. در حالت کلی، برای جمع دو عدد ۴ بیتی، به ۴ عدد Full adder (یعنی ۸ عدد Half adder) نیاز داریم، اما در این حالت می دانیم که مقادیر Carry ورودی و بیت سوم عدد دوم (همان 1011) صفر هستند، پس این کار با دو Half adder و دو Full adder انجام می شود: (خروجی چهاربیتی تفریق به صورت  $B_3B_2B_1B_0$  نشان داده شده است)



۲- (۳ نمره) با کمک دیکودر  $3 \times 8$ ، مدارهای زیر را طراحی کنید. در طراحی خود یک بار از دیکودری با خروجی‌های active-high استفاده کنید و یک بار از دیکودرهایی با خروجی‌های active-low.

الف- مداری که یک عدد سه‌بیتی را به عنوان ورودی دریافت می‌کند و سه بیت 2's complement آن را خروجی می‌دهد.  
ب- مداری که یک عدد سه‌بیتی را به عنوان ورودی دریافت می‌کند و هفت بیت Seven-segment آن را خروجی می‌دهد.

پاسخ سوال (۲)

الف) ابتدا Truth Table مقادیر خروجی  $(B_2 B_1 B_0)$  بر اساس مقادیر ورودی  $(A_2 A_1 A_0)$  را رسم می‌کنیم:

$A_2 A_1 A_0$	$B_2 B_1 B_0$
000	000
001	111
010	110
011	101
100	100
101	011
110	010
111	001

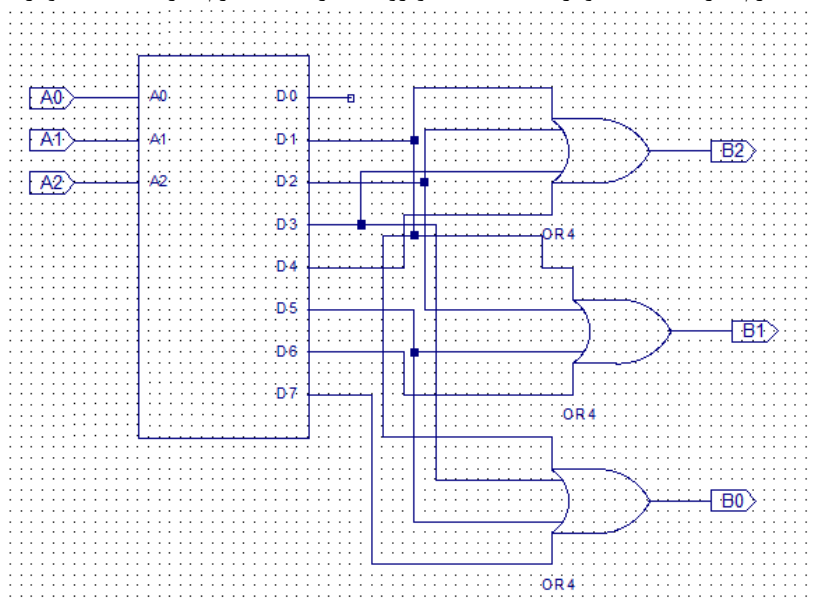
حال با توجه به جدول بالا برای هر خروجی رابطه‌ای بر اساس بیت‌های ورودی و به صورت جمع مینترمها و ضرب ماکسترها می‌نویسیم:

$$B_2 = \sum m(1.2.3.4) = \prod M(0.5.6.7)$$

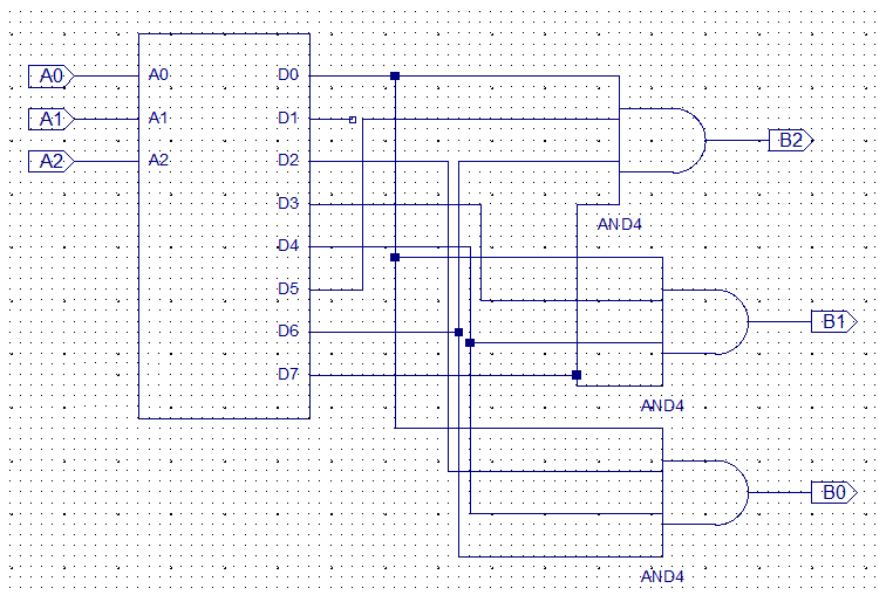
$$B_1 = \sum m(1.2.5.6) = \prod M(0.3.4.7)$$

$$B_0 = \sum m(1.3.5.7) = \prod M(0.2.4.6)$$

می‌توانیم روابط جمع مینترمها را با یک دیکودر active-high و روابط ضرب ماکسترها را با یک دیکودر active-low بسازیم:



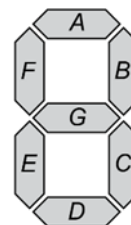
Active-high outputs



Active low outputs

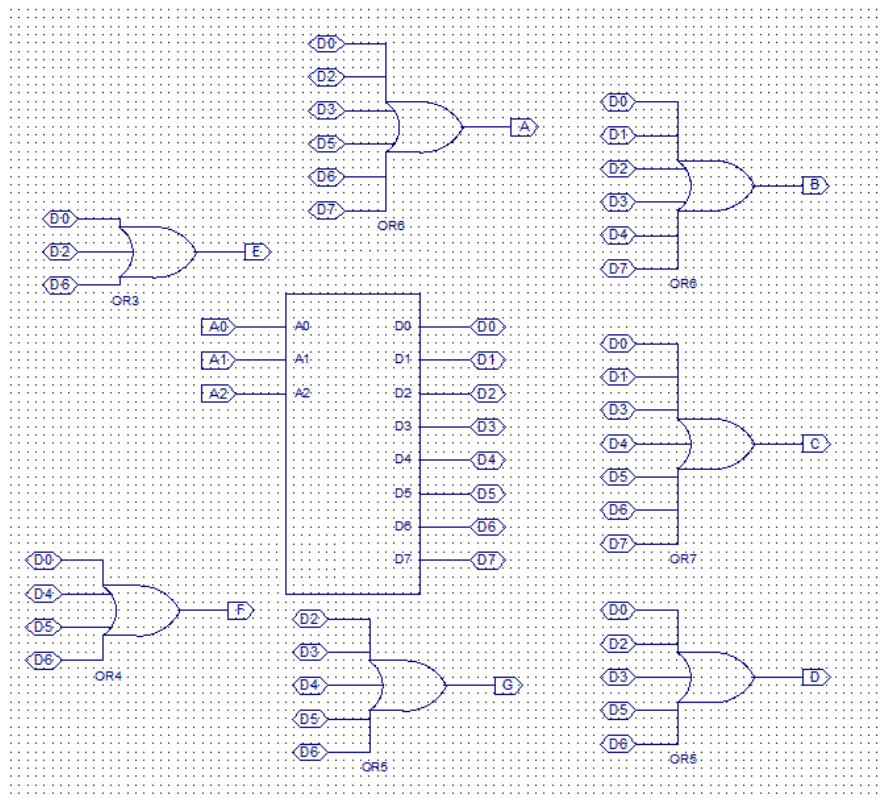
ب) خروجی‌ها ۷ بیت a تا g هستند که در صورت ۱ بودن هر بیت، سگمنت متناظر با آن بیت روشن می‌شود. ابتدا جدول درستی را رسم می‌کنیم و با توجه به جدول روابط خروجی‌ها را می‌نویسیم:

Inputs			Segments						
A	B	C	a	b	c	d	e	f	g
0	0	0	1	1	1	1	1	1	0
0	0	1	0	1	1	0	0	0	0
0	1	0	1	1	0	1	1	0	1
0	1	1	1	1	1	1	0	0	1
1	0	0	0	1	1	0	0	1	1
1	0	1	1	0	1	1	0	1	1
1	1	0	1	0	1	1	1	1	1
1	1	1	1	1	1	0	0	0	0

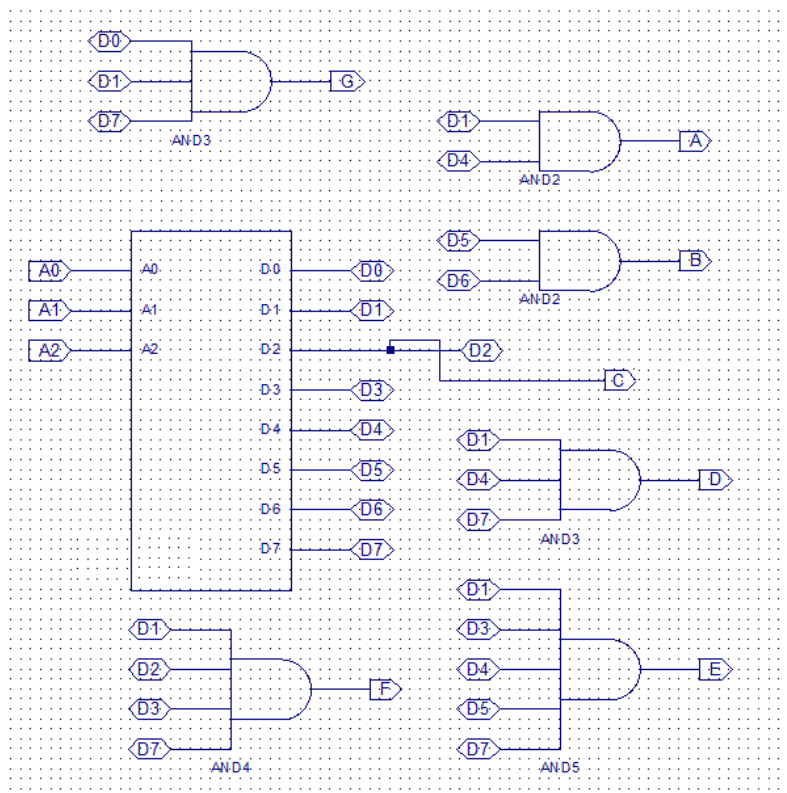


$$\begin{aligned}
 a &= \sum m(0.2.3.5.6.7) = \prod M(1.4) \\
 b &= \sum m(0.1.2.3.4.7) = \prod M(5.6) \\
 c &= \sum m(0.1.3.4.5.6.7) = \prod M(2) \\
 d &= \sum m(0.2.3.5.6) = \prod M(1.4.7) \\
 e &= \sum m(0.2.6) = \prod M(1.3.4.5.7) \\
 f &= \sum m(0.4.5.6) = \prod M(1.2.3.7) \\
 g &= \sum m(2.3.4.5.6) = \prod M(0.1.7)
 \end{aligned}$$

می‌توانیم روابط جمع مینترم‌ها را با یک دیکودر active-high و روابط ضرب ماکسترم‌ها را با یک دیکودر active-low پیاده‌سازی کنیم:

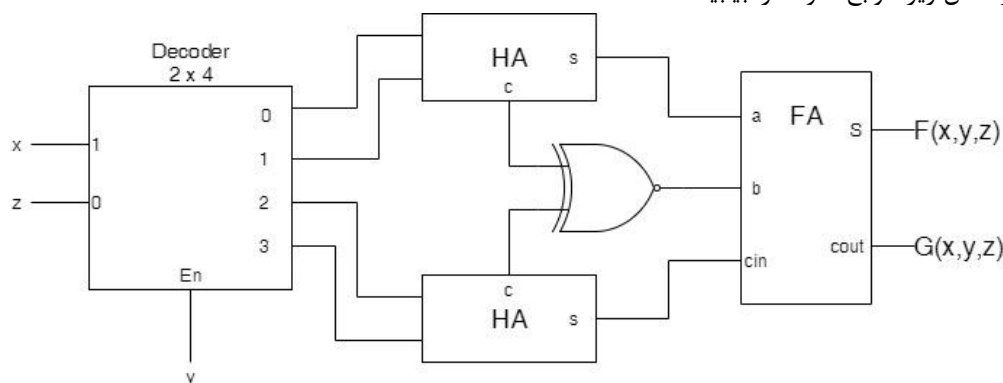


Active high Outputs



Active low outputs

۳- (۲ نمره) در شکل زیر، توابع  $F$  و  $G$  را بیابید.



پاسخ سوال ۳) ورودی  $y$ ، فعال یا غیر فعال بودن دیکودر را مشخص می کند. در صورت غیر فعال بودن، خروجی های دیکودر صفر خواهند بود و در صورت فعال بودن، با توجه به مقادیر  $x$  و  $z$ ، حالات مختلف را بررسی می کنیم

							FA					
y	x	z	خروجی دیکدر			a b c <sub>in</sub>	F=S	G=C				
0	-	-	0→0	HA	S=0	0 1 c <sub>0</sub>	1	0				
			1→0		C=0							
			2→0									
			3→0									
				HA	S=0							
				HA	C=0							
1	0	0	خروجی دیکدر				بالایی		1 1 0	0	1	
							HA					
							HL					
			0 1 2 3				S C					
	0	0	1	0	0	0	1	0	0	0	0	1
	0	1	0	1	0	0	1	0	0	0	1	1
	1	0	0	0	1	0	0	0	1	0	0	1
	1	1	0	0	0	1	0	0	1	0	0	1

این سوال را با استدلال ساده تری هم می توانیم حل کنیم:

اگر دیکودر فعال باشد ( $y=1$ )، با توجه به اینکه در هر زمان فقط یکی از خروجی های دیکودر یک و بقیه صفر هستند، یکی از دو نیم جمع کننده در حال جمع کردن صفر و یک و دیگری در حال جمع کردن صفر و صفر است، پس جمع کننده سمت راست در حال جمع کردن صفر و یک و  $b$  خواهد بود. اگر دیکودر غیرفعال باشد ( $y=0$ ) خروجی هر دو نیم جمع کننده صفر است و جمع کننده سمت راست در حال جمع کردن صفر و صفر و  $b$  است. با این توضیحات می توانیم بگوییم جمع کننده سمت راست در حال جمع کردن  $y$  و صفر و  $b$  است.

از طرفی، چه دیکودر فعال باشد و چه نباشد، بیت های نقلی خروجی دو نیم جمع کننده صفر هستند، بنابراین  $b$  همیشه یک است. بنابراین، جمع کننده سمت راست همیشه در حال جمع کردن صفر و یک و  $y$  است. پس:

$$F(x,y,z)=y \oplus 1 \oplus 0=y'$$

$$G(x,y,z)=y.1+y.0+1.0=y$$

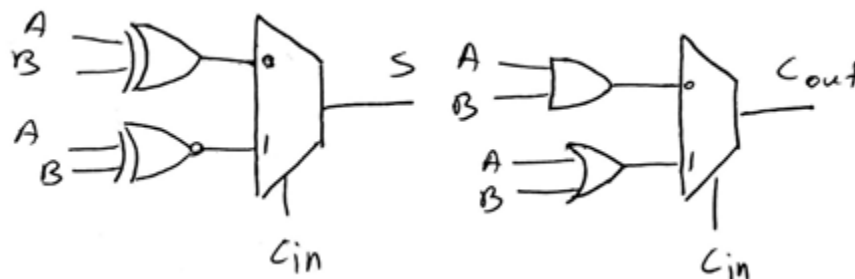
۴- (۲ نمره) با کمک دو مالتی پلکسر ۲ به ۱ و کمترین تعداد گیت ممکن یک full adder بسازید.

پاسخ سوال (۴)

ابتدا Truth Table را رسم می کنیم:

$C_{in}$	A	B	$C_{out}$	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

می توانیم برای هر خروجی یک MUX قرار دهیم و بیت  $C_{in}$  را به عنوان آدرس قرار دهیم. با توجه به جدول روابط زیر را می نویسیم:



۵- (۳ نمره) می خواهیم مداری بسازیم که یک عدد ۴ بیتی  $A = a_3a_2a_1a_0$  را بگیرد و اگر A مضرب ۴ یا ۵ بود، خروجی آن یک شود. دقت کنید، صفر مضرب همه اعداد است.

الف) این مدار را با استفاده از یک مالتی پلکسر ۸ ورودی (۳ خط آدرس) طراحی کنید.

ب) این مدار را با استفاده از یک مالتی پلکسر ۴ ورودی (۲ خط آدرس) طراحی کنید.

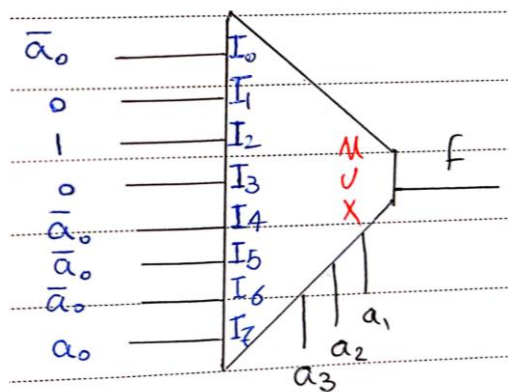
ج) این مدار را با استفاده از گیت هایی با حداکثر دو ورودی طراحی کنید.

فرض کنید مکمل هر بیت ورودی را نیز داریم. اگر علاوه بر مدارهای ذکر شده به گیت اضافه نیز نیاز است، می توانید از آن استفاده کنید اما مدار باید تا حد امکان ساده باشد.

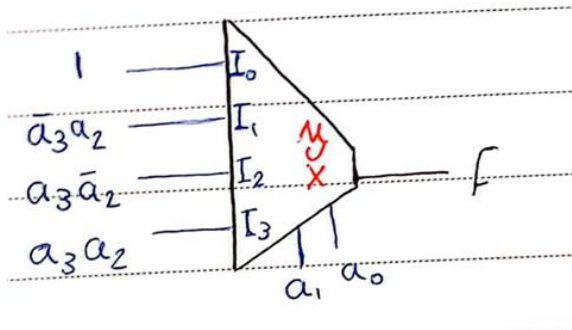
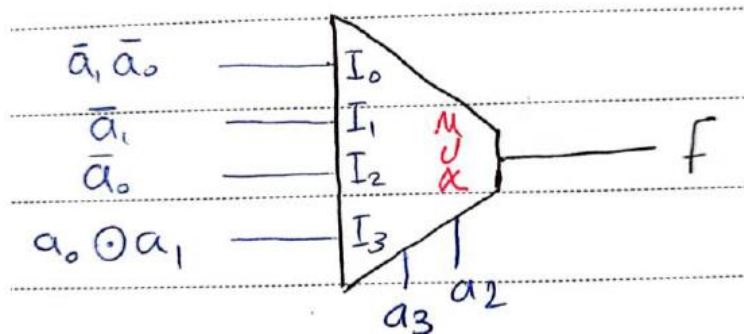
پاسخ سوال (۵)

$a_3a_2$	$a_1a_0$	00	01	11	10
00	1	1	1	1	
01	0	1	0	0	
11	0	0	1	0	
10	0	0	0	1	

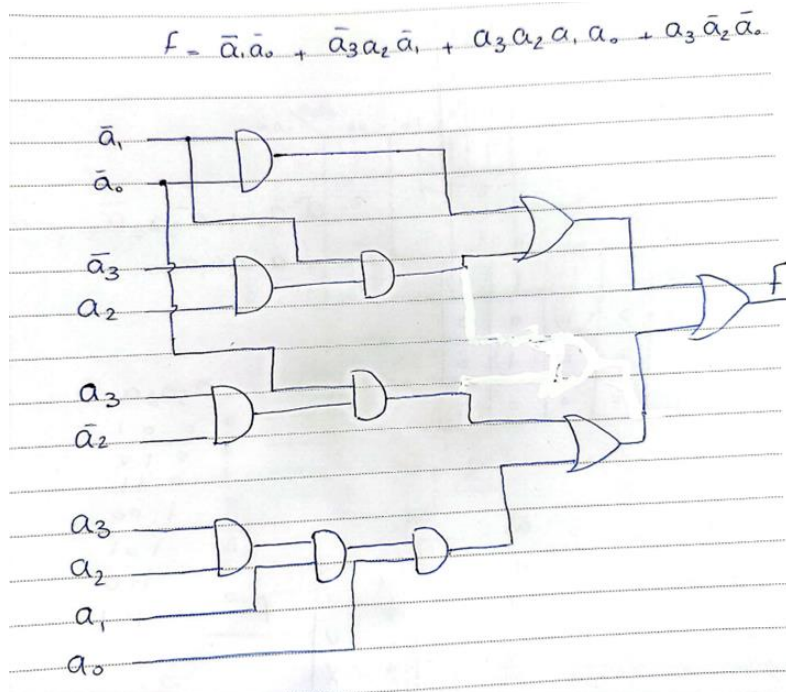
الف) اگر خطوط آدرس را  $a_3a_2a_1$  قرار دهیم، داریم:



(ب)

اگر خطوط آدرس  $a_1a_0$  باشد.اگر خطوط آدرس  $a_3a_2$  باشد.

(ج) تابع را به شکل SOP ساده می‌کنیم: ( برای این قسمت، راه‌حل‌های دیگری هم که قید استفاده از گیت دو ورودی را رعایت کرده باشد صحیح است و تعداد گیت محدودیتی ندارد.)



۶- (۲ نمره) با استفاده از یک دی‌مالتی‌پلکسر ۱ به ۴ و حداقل تعداد گیت ممکن یک مالتی‌پلکسر ۵ به ۱ بسازید.

### پاسخ سوال ۶

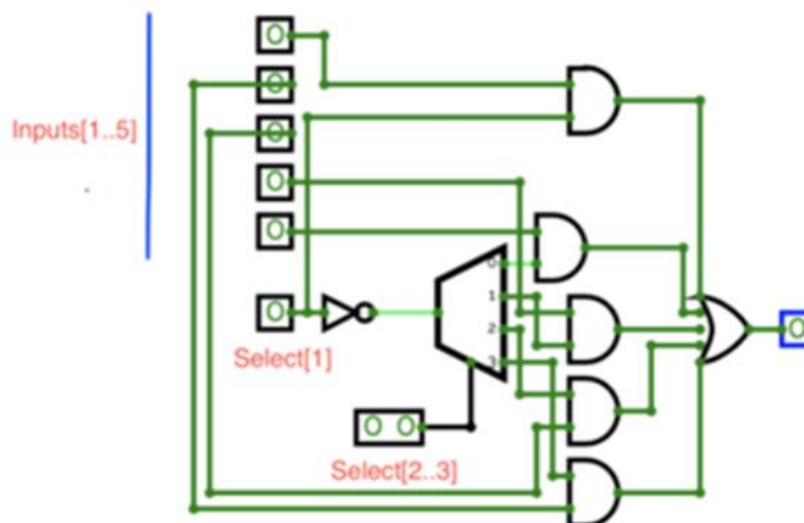
هدف این است که یک مالتی‌پلکسر ۵ به یک بسازیم، یعنی از بین ۵ ورودی، یکی را انتخاب کنیم و در خط خروجی بگذاریم. برای انتخاب بین ۵ ورودی، حداقل به سه بیت select نیاز داریم.

اگر قرار بود یک مالتی‌پلکسر ۸ به یک بسازیم باید از یک دیکودر ۳ به ۸ یا دو دیکودر ۲ به ۴ با ورودی فعال‌ساز (همان دی‌مالتی‌پلکسر ۲ به ۴) استفاده کنیم. حالا که سه ورودی آخر مالتی‌پلکسر را نداریم، پس به دیکودر (دی‌مالتی‌پلکسر) دومی هم نیازی نیست.

دو بیت Select را به یک دی‌مالتی‌پلکسر دو به چهار می‌بندیم. ورودی آن را متمم بیت چپ Select می‌گذاریم. به این ترتیب، وقتی این بیت صفر است، این دی‌مالتی‌پلکسر فعال است و یکی از چهار خروجی خود را انتخاب می‌کند. اگر این بیت یک باشد، غیرفعال شده و هیچ کدام از چهار خروجی آن فعال نمی‌باشند. در این حالت، گیت and بالایی که یک ورودی‌اش این بیت چپ Select است، برابر با ورودی دیگرش که یک خط ورودی است، خواهد شد. خروجی این گیت، چون دی‌مالتی‌پلکسر غیرفعال است، در نهایت خروجی گیت or نهایی را تعیین خواهد کرد.

وقتهایی که دی‌مالتی‌پلکسر فعال است، این گیت and بالایی همیشه غیرفعال است، و لذا خروجی گیت or نهایی برابر ورودی‌ای است که با خروجی فعال دی‌مالتی‌پلکسر and شده‌است.

پس می‌بینیم که به ازای حالت‌های مختلف بیت‌های select، خطوط ورودی مختلف به خروجی نهایی هدایت می‌شوند.

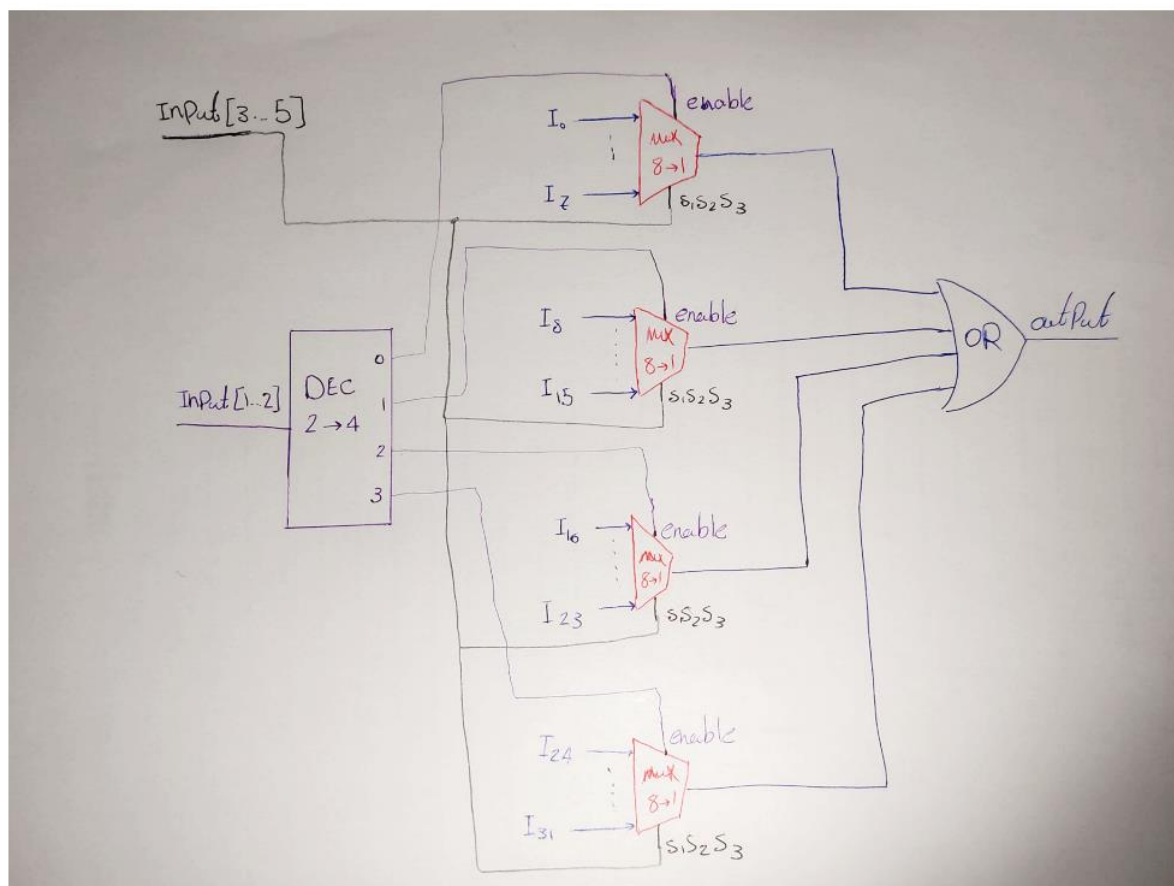


۷- (۲ نمره) با استفاده از حداقل تعداد مالتی‌پلکسر ۸ به ۱ و یک دیکودر با کمترین تعداد ورودی و خروجی یک مالتی‌پلکسر ۳۲ به ۱ بسازید. اگر لازم است برای دیکودر و مالتی‌پلکسرهایی که استفاده می‌کنید ورودی فعال‌ساز (enable) هم در نظر بگیرید.

### پاسخ سوال ۷

چهار مالتی‌پلکسر ۸ به یک را در نظر بگیرید. ورودی‌های مالتی‌پلکسر هدف که ۳۲ تا است را به ترتیب به این‌ها متصل می‌کنیم. خروجی هر چهار مالتی‌پلکسر کوچک را به یک گیت or وصل می‌کنیم که خروجی این گیت، خروجی نهایی ما است. حال لازم داریم تا با توجه به مقدار ۵ بیت خط select، در هر زمان فقط یکی از مالتی‌پلکسرهای کوچک فعال باشد. برای این کار، دو بیت چپ (دو به توان دو برابر چهار حالت می‌شود) را به یک دیکودر می‌دهیم و چهار خروجی این دیکودر را به خط enable مالتی‌پلکسرهای کوچک به ترتیب متصل می‌کنیم. سه بیت باقی‌مانده select را به همه مالتی‌پلکسرهای کوچک وصل می‌کنیم.





۸- (۳ نمره) فرض کنید می‌خواهیم دو عدد  $n$  بیتی  $A = a_{n-1}a_{n-2}\dots a_1a_0$  و  $B = b_{n-1}b_{n-2}\dots b_1b_0$  را با هم مقایسه کنیم. تعریف می‌کنیم:  $A_i = a_{n-1}\dots a_i$  و  $B_i = b_{n-1}\dots b_i$ .  $E_i$  و  $L_i$  و  $G_i$  به ترتیب وقتی یک می‌شوند که  $A_i = B_i$  و  $A_i < B_i$  و  $A_i > B_i$ . بنابراین  $E_0$  و  $L_0$  و  $G_0$  به ترتیب وقتی یک می‌شوند که  $A > B$  و  $A < B$  و  $A = B$ . یک فرمول بازگشتی برای  $E_i$  و  $L_i$  و  $G_i$  بیابید یعنی این‌ها را بر حسب  $E_{i+1}$  و  $L_{i+1}$  و  $G_{i+1}$  بنویسید. سپس شکل منطقی و بلوک دیاگرام مدار را رسم کنید. اگر این بلوک دیاگرام را  $C$  بنامیم، با کمک این بلوک  $C$  دو عدد ۴ بیتی  $A = a_3a_2a_1a_0$  و  $B = b_3b_2b_1b_0$  را مقایسه کنید. (منظور از  $E$  و  $L$  و  $G$  به ترتیب Equal و Less و Greater است).

پاسخ سوال ۸)

$$\begin{array}{ccccccc|c|ccccccc}
 A = & a_{n-1} & a_{n-2} & \dots & a_{i+1} & a_i & a_{i-1} & \dots & a_1 & a_0 \\
 B = & b_{n-1} & b_{n-2} & \dots & b_{i+1} & b_i & b_{i-1} & \dots & b_1 & b_0
 \end{array}$$
  

$$\begin{array}{c}
 \xrightarrow{E_{i+1}, L_{i+1}, G_{i+1}} \\
 \xrightarrow{E_i, L_i, G_i}
 \end{array}$$

$E_i$  وقتی یک می شود که  $A_i = B_i$  یعنی:

$$a_{n-1} \dots a_{i+1} a_i = b_{n-1} \dots b_{i+1} b_i$$

به عبارتی  $E_i$  وقتی یک می شود که  $a_i = b_i$  و  $a_{n-1} \dots a_{i+1} = b_{n-1} \dots b_{i+1}$  پس  $E_i$  وقتی یک می شود که  $E_{i+1} = 1$  و  $a_i = b_i$  بنابراین

$$E_i = E_{i+1} \cdot (a_i \odot b_i)$$

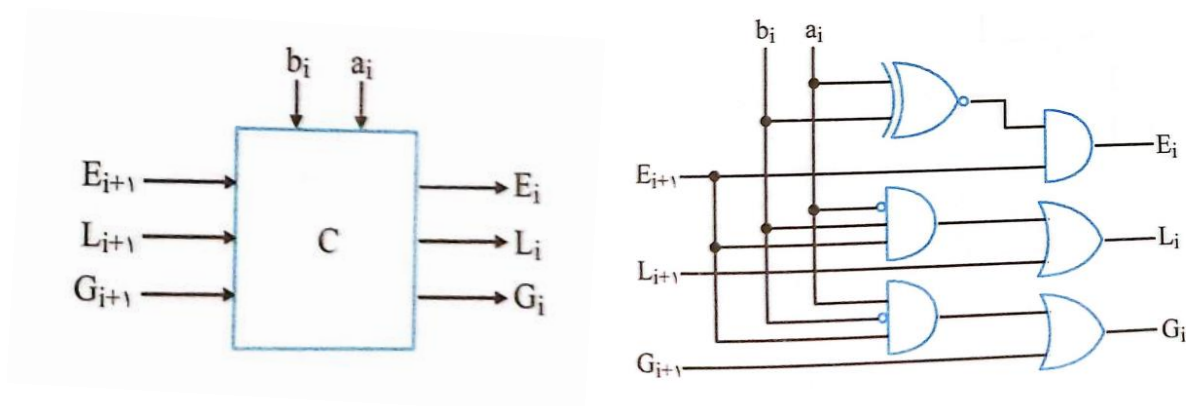
$L_i$  وقتی یک می شود که  $A_i < B_i$  یعنی  $a_{n-1} \dots a_{i+1} a_i < b_{n-1} \dots b_{i+1} b_i$  که این اتفاق وقتی رخ می دهد که یا  $a_{n-1} \dots a_{i+1} < b_{n-1} \dots b_{i+1}$  (یعنی  $L_{i+1} = 1$ ) یا  $a_{n-1} \dots a_{i+1} = b_{n-1} \dots b_{i+1}$  (یعنی  $E_{i+1} = 1$ ) و  $a_i < b_i$  پس:

$$L_i = L_{i+1} + E_{i+1} (\bar{a}_i \cdot b_i)$$

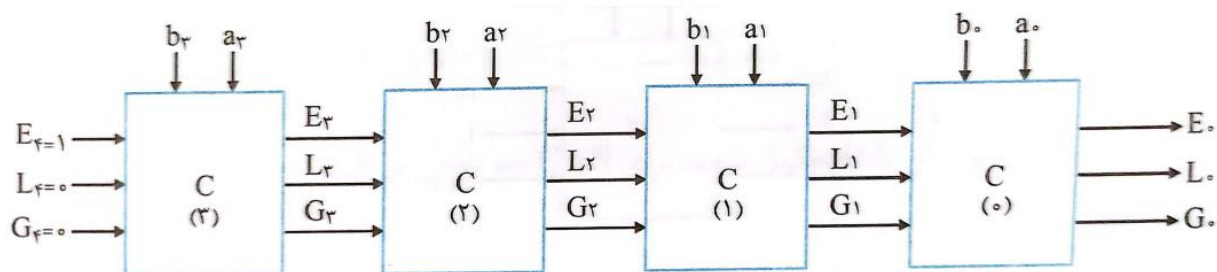
$G_i$  وقتی یک می شود که  $A_i > B_i$  یعنی  $a_{n-1} \dots a_{i+1} a_i > b_{n-1} \dots b_{i+1} b_i$  که این اتفاق وقتی رخ می دهد که یا  $a_{n-1} \dots a_{i+1} > b_{n-1} \dots b_{i+1}$  (یعنی  $G_{i+1} = 1$ ) یا  $a_{n-1} \dots a_{i+1} = b_{n-1} \dots b_{i+1}$  (یعنی  $E_{i+1} = 1$ ) و  $a_i > b_i$  پس:

$$G_i = G_{i+1} + E_{i+1} (a_i \cdot \bar{b}_i)$$

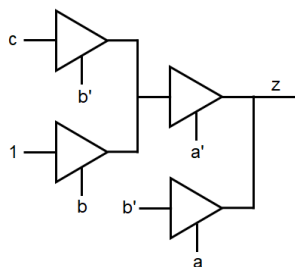
پس شکل منطقی و بلوک دیاگرام مدار، به صورت زیر است:



با کمک این بلوک دیاگرام C برای مقایسه دو عدد چهاربیتی داریم:

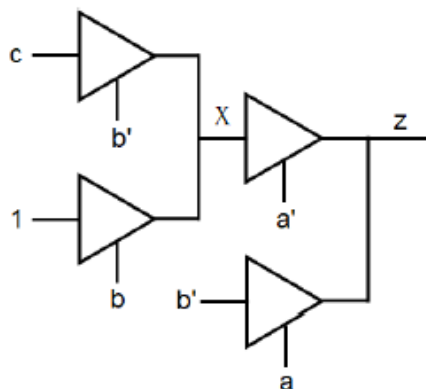


۹- (۱ نمره) در شکل زیر معادله خروجی  $z$  را به ساده‌ترین شکل ممکن بنویسید.



پاسخ سوال ۹

برای سادگی توضیح مقدار میانی  $X$  روی شکل مشخص شده:



با توجه به دو بافر سمت چپ مدار، مقدار  $X$  در صورتی که  $b$  یک باشد برابر با ۱ و در غیر این صورت برابر با  $c$  می‌شود. پس داریم:

$$X = b + b'.c = b + c$$

همچنین با توجه به دو بافر سمت راست مدار، مقدار  $z$  در صورتی که  $a$  یک باشد برابر با  $b'$  و در غیر این صورت برابر با  $X$  می‌شود.

پس داریم:

$$\begin{aligned} z &= a.b' + a'.X \\ &= a.b' + a'.(b + c) = a.b' + a'.b + a'.c \end{aligned}$$