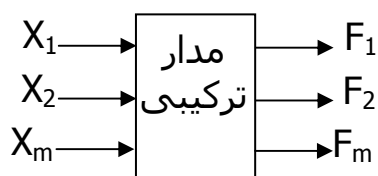


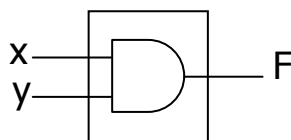
مدارهای منطقی ترکیبی :

مدار منطقی ترکیبی مداری است که در آن خروجیها در هر لحظه به ورودیها در همان



لحظه بستگی دارند .

مثال :



$$F = x.y$$

مدار
ترکیبی

مدارهای منطقی ترکیبی :

۱- SSI (Small Scale Integrated Circuits)

مدارهای ترکیبی مقیاس کوچک مانند :

AND OR NOT XOR

۲- MSI (Medium OR NOT XOR)

مدارهای ترکیبی مقیاس متوسط مانند :

- مدارهای مبدل کد ها

- مدارهای جمع کننده / تفریق کننده

- مالتی پلکسرها

- دیکودرها

۲- LSI (Large Scale Integrated Circuits)

۴- VLSI (Very Large scale Integrated Circuits) همچون ریزپردازنده ها

طراحی مدارهای ترکیبی

برای طراحی یک مدار منطقی باید مراحل زیر را دنبال نمود:

۱- نحوه ی کار

۲- تعریف عملکرد مدار

۳- تعیین ورودیها و تعداد آنها

۴- تعیین خروجیها

۵- نوشتن روابط خروجیها برحسب ورودیها - جدول صحت (

۶- ساده سازی عبارات خروجیها برحسب ورودیها

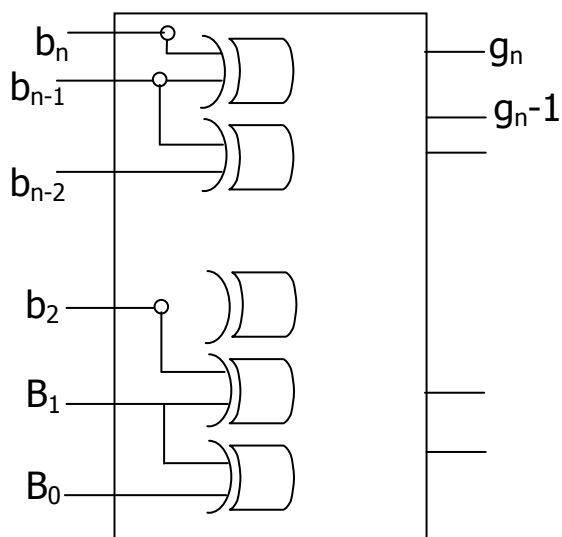
۷- پیاده سازی

دسته بندی مدارهای ترکیبی

الف (مدارهای مبدل کد :

جهت تبدیل کدهای مختلف به یکدیگر

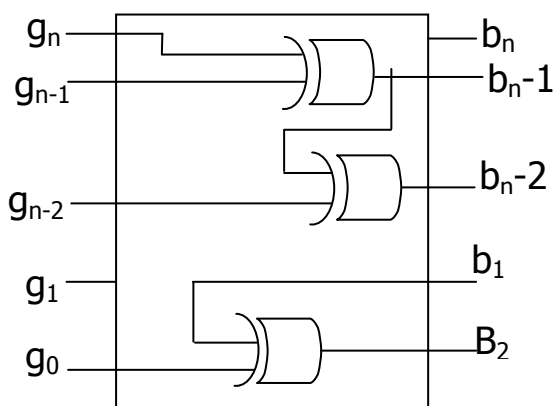
۱- مدار مبدل کد باینری به گری :



$$g_n = b_n$$

$$g_i = b_i \oplus b_{i+1} \quad 0 \leq i \leq n-1$$

تاخیر به اندازه n-1 طبقه



۲- مدار مبدل گری به باینری :

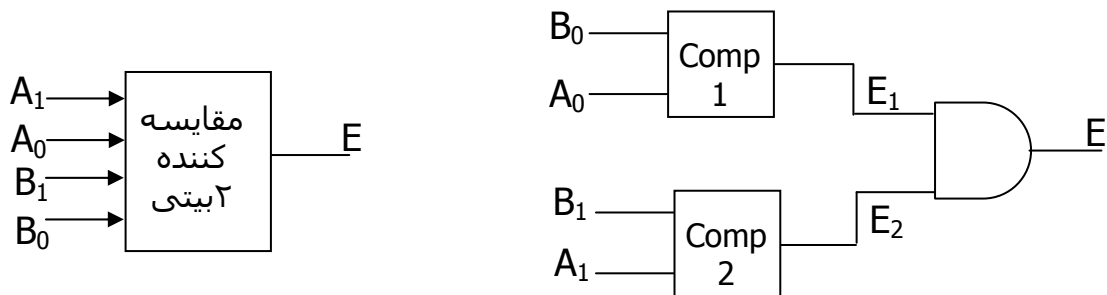
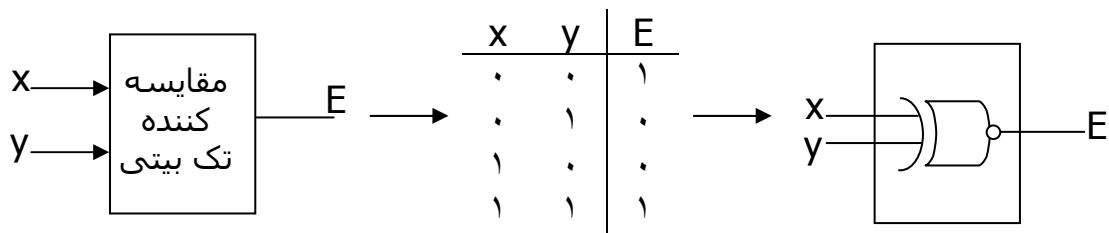
$$b_n = g_n$$

$$b_i = g_i \oplus b_{i+1} \quad 0 \leq i \leq n-1$$

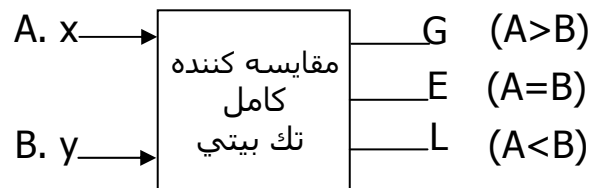
« تاخیر تجمعی به اندازه n طبقه »

ب) مدارهای مقایسه کننده :

برای مقایسه اطلاعات ورودی به کار می رود .



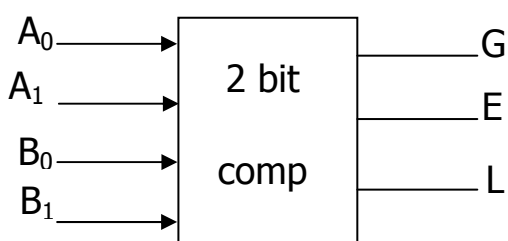
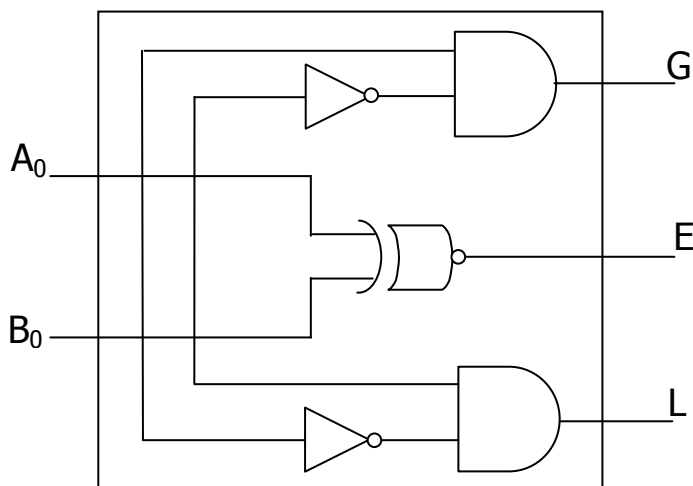
مدار مقایسه کننده کامل : ۱- ابتدا عملکرد برای یک بیت بررسی می شود.



۲- بعد جدول صحت را تنظیم کنید .

A_0	B_0	G	E	L
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

۳- مدار را ترسیم کنید .



$$A: A_1 A_0$$

$$B: B_1 B_0$$

$$E = (A_1 \oplus B_1) \circ (A_0 \oplus B_0)$$

$$G = (A_1 > B_1) + (A_1 = B_1) \circ (A_0 > B_0)$$

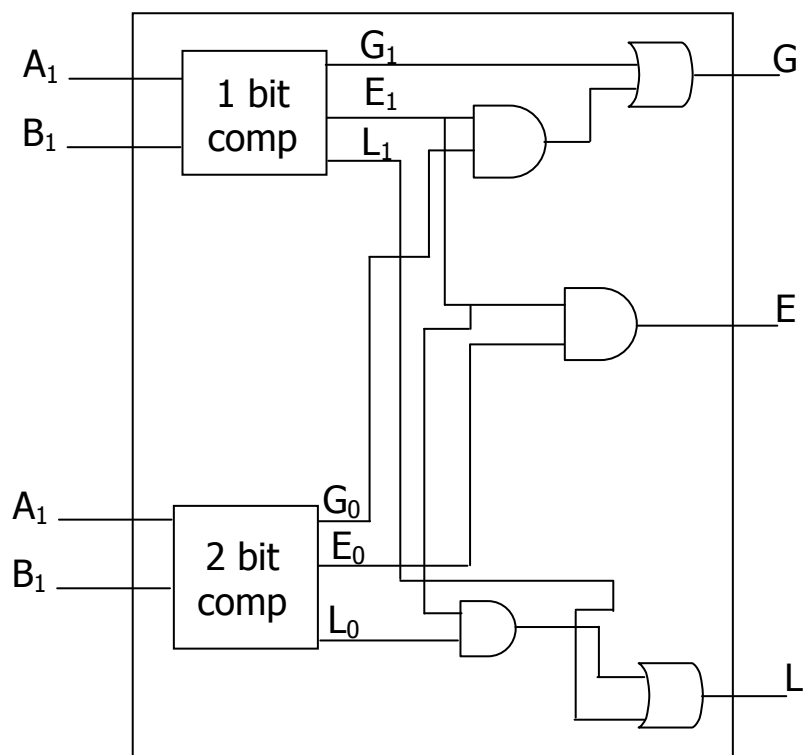
↓

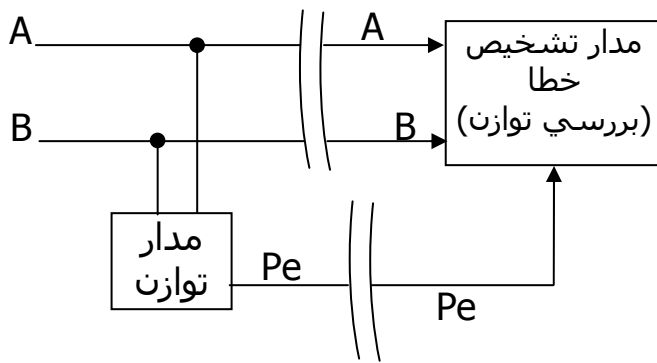
$$G = A_1 B_1' + (A_1 \oplus B_1) \circ (A_0 B_0') = (G_1 + E_1 G_0)$$

$$L = (A_1 < B_1) + (A_1 = B_1) \circ (A_0 < B_0)$$

↓

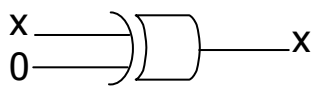
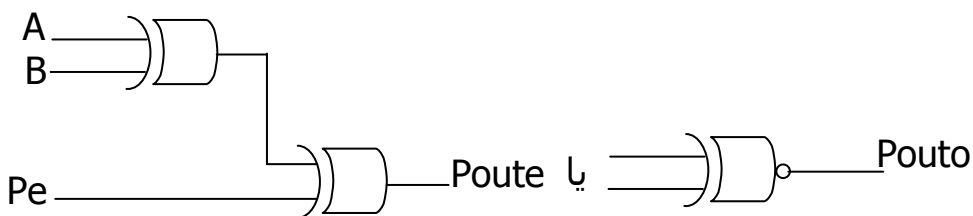
$$L = (A_1' B_1 + (A_1 \oplus B_1) \circ (A_0' B_0)) = L_1 + E_1 L_0$$



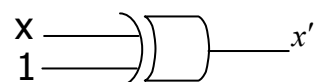
(ب) مدارهای تولید توازن و تشخیص:

زوج		فرد					
A	B	Pe	Po	A	B	Pe	Poute
۰	۰	۰	۱	۰	۰	۰	۰
۰	۱	۱	۰	۰	۱	۰	۱
۱	۰	۱	۰	۱	۰	۱	۰
۱	۱	۰	۱	۱	۱	۰	۱

$$\begin{array}{c}
 \text{A} \quad \text{B} \quad \text{Pe} \\
 \begin{array}{|c|c|c|c|}
 \hline
 ۰ & ۱ & ۰ & ۱ \\
 \hline
 ۱ & ۰ & ۱ & ۰ \\
 \hline
 \end{array}
 \Rightarrow
 \begin{array}{l}
 P_{oute} = A \oplus B \oplus p_e \\
 P_{oute} = [(A \oplus B) \oplus p_e]' = (A \oplus B) \oplus P
 \end{array}
 \end{array}$$



$$F = x.1 + x'.0 = x$$

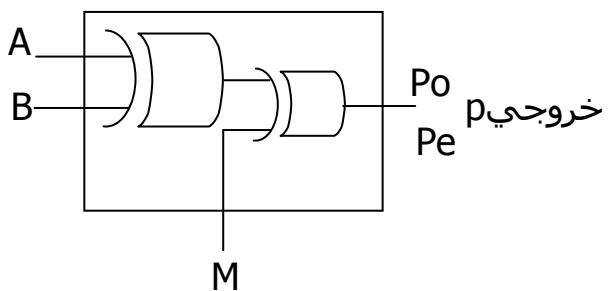


$$F = x.0 + x'.1 = x'$$

نکته :

هرگاه یکی از ورودیهای XOR ، ۰ باشد خروجی برابر با ورودی دیگر است و اگر ۱ باشد برابر با متمم خروجی دیگر است .

هدف : طراحی مداری که با یک بیت کنترل توازن زوج یا فرد ایجاد می کند .



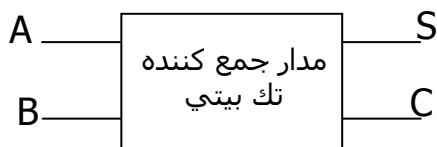
خروجی p

$$iF \quad M = 0 \text{ then } P = p_0, \quad P = A \oplus B$$

$$iF \quad M = 1 \text{ then } P = p_1, \quad P = A \oplus B$$

پ) مدارهای جمع کننده / تفریق :

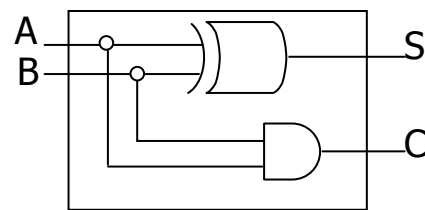
$$\begin{array}{r} A \\ + B \\ \hline CS \end{array}$$



۱- نیم-جمع کننده

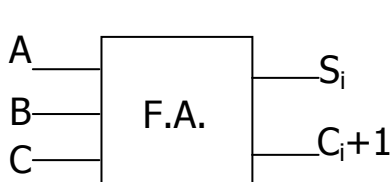
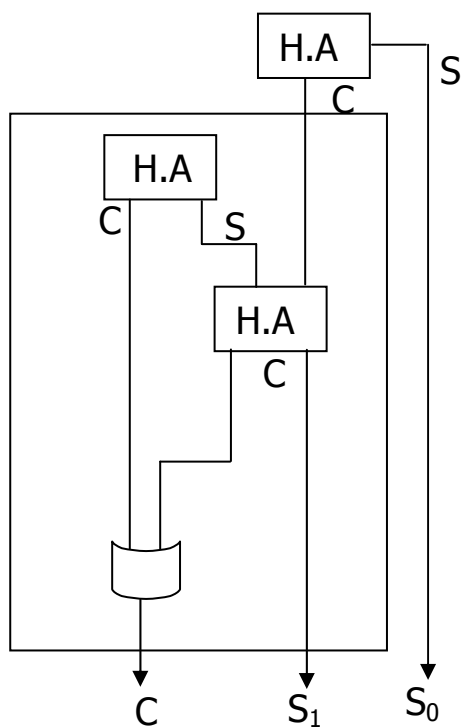
A	B	C	S
۰	۰	۰	۰
۰	۱	۰	۱
۱	۰	۰	۱
۱	۱	۱	۰

$$\rightarrow \begin{cases} S = A \oplus B \\ C = A.B \end{cases} \rightarrow$$



Half Adder (H.A) یا نیم جمع کننده

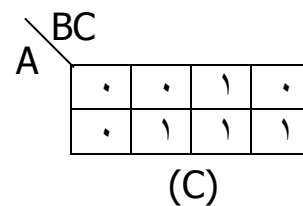
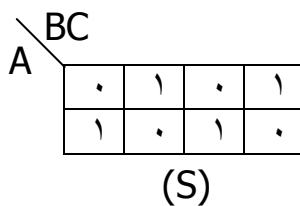
۲- مدار تمام-جمع کننده



$$\begin{array}{r} ۰۱ \\ + ۱۱ \\ \hline ۱۰۰ \\ C_2 S_1 S_0 \end{array}$$

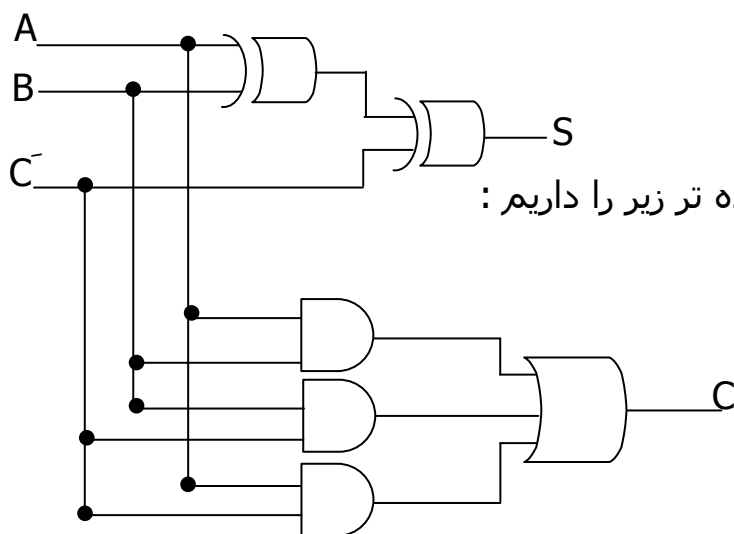
A	B	C	C	S
۰	۰	۰	۰	۰
۰	۰	۱	۰	۱
۰	۱	۰	۰	۱
۰	۱	۱	۱	۰
۱	۰	۰	۰	۱
۱	۰	۱	۱	۰
۱	۱	۰	۱	۰
۱	۱	۱	۱	۰

$$\begin{array}{r} A \\ B \\ + C \\ \hline CS \end{array}$$



$$\Rightarrow S = \sum m(1,2,4,7) = A \oplus B \oplus C$$

$$C = \sum m(3,5,6,7) = AB \oplus BC \oplus AC$$

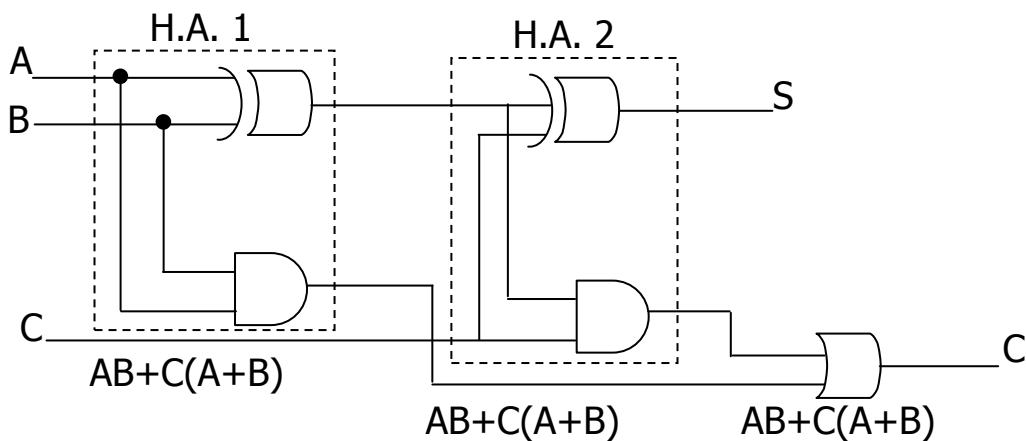


به جای این مدار شلوغ مدار ساده تر زیر را داریم :

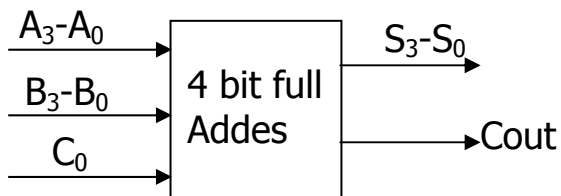
$$C_{out} = AB + BC + AC$$

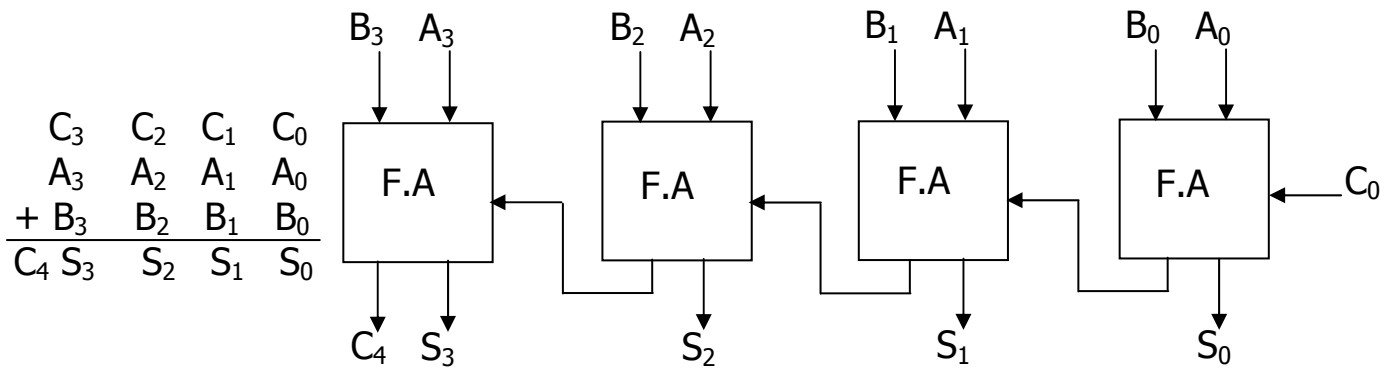
$$= AB + C(A + B)$$

لذا :

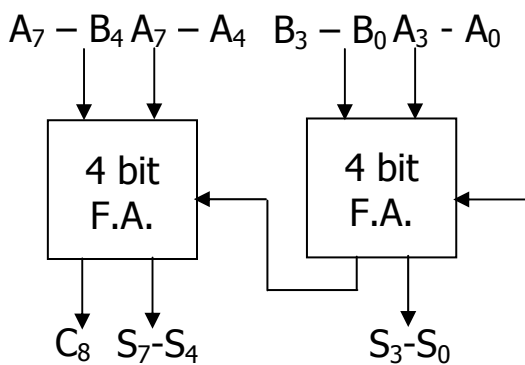


۲- جمع کننده کامل ۴ بیتی :



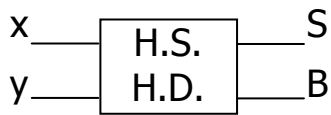


$$\begin{array}{r} C_3 \quad C_2 \quad C_1 \quad C_0 \\ A_3 \quad A_2 \quad A_1 \quad A_0 \\ + B_3 \quad B_2 \quad B_1 \quad B_0 \\ \hline C_4 \quad S_3 \quad S_2 \quad S_1 \quad S_0 \end{array}$$



8 bit Full Adder :

۲- مدار تفریق کننده :

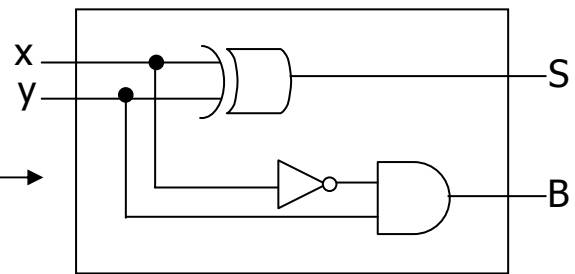


$$\begin{array}{r} x \\ -y \\ \hline BS \end{array}$$

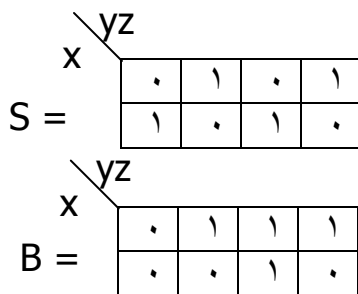
X	Y	B	S
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.

$$S = x \oplus y$$

$$B = x'y$$



X	Y	Z	B	S
.
.
.
.
.
.
.
.



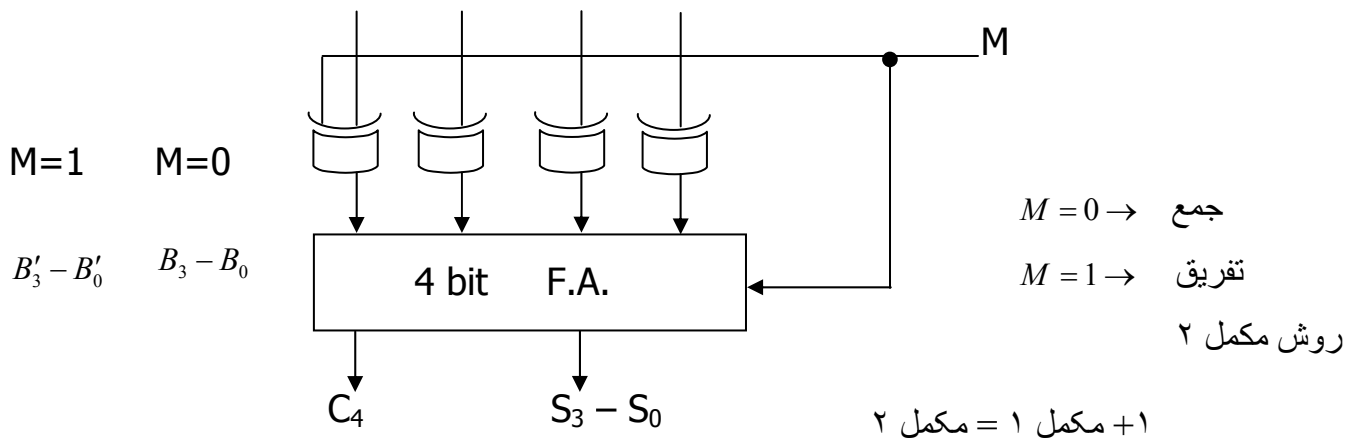
$$\begin{array}{r} x \\ -y \\ -z \\ \hline BS \end{array}$$

$$S = x \oplus y \oplus z$$

$$B = x'z + x'y + yz$$

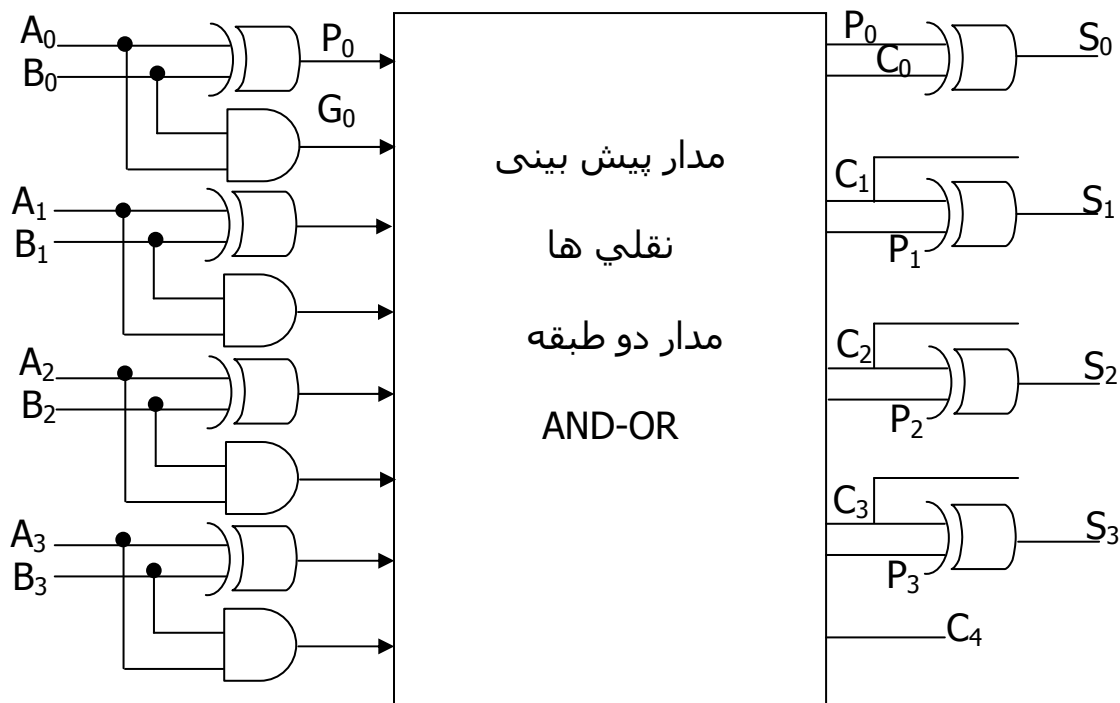
۵- طراحی مدار جمع کننده / تفریق کننده ی ۴ بیتی با روش مکمل ۲ با خط

کنترل M:



نکته : در مدارهای جمع کننده تاخیر تجمعی داریم که با افزایش تعداد بیت های ورودی افزایش می یابد . برای مدار جمع کننده ی ۲ بیتی ۵ طبقه تاخیر داشتیم پس این مدار مشکل دارد پس مدار جمع کننده با نقلی پیش بینی شده طراحی شد .

۶- مدار جمع کننده با نقلی پیش بینی شده (Look Ahead Carry Generator) :



$$S_0 = G_0 \oplus P$$

$$C_1 = G_0 + C_0 P_0$$

$$S_1 = C_1 \oplus P_1$$

$$C_2 = G_1 + C_1 P_1$$

$$C_2 = G_1 + P_1 G_0 + P_1 P_2 C_0$$

$$S_2 = G_2 \oplus P_2$$

$$C_3 = G_2 + C_2 P_2$$

$$C_3 = G_2 + P_2 G_1 + P_1 P_2 G_1 + P_1 P_2 P_0 G_0$$

$$P_i = A_i \oplus B_i$$

$$B_i = A_i \cdot B_i$$

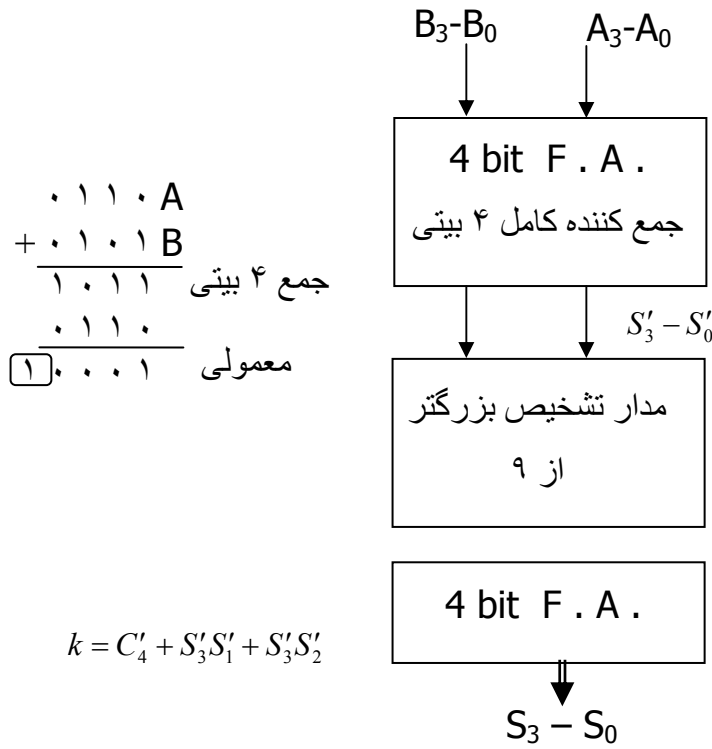
$$S_i = P_i \oplus C_i$$

$$C_i = G_{i-1} + C_{i-1} P_{i-1}, C_0$$

داریم

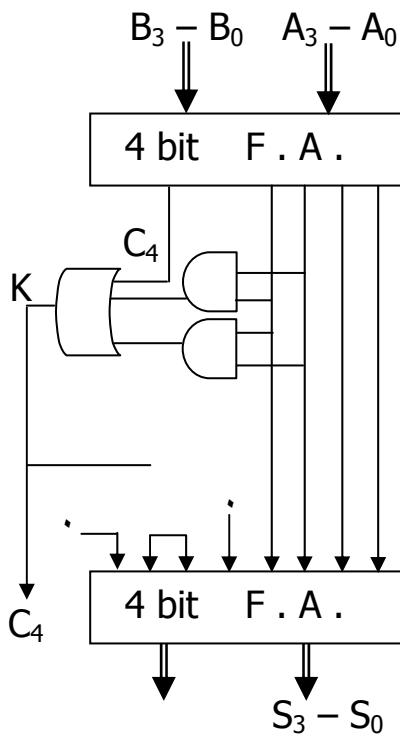
نکته : این مدار و کلیه مدارهای توسعه یافته (6,8) و ... بیتی می توانند با این روش با 4 طبقه تاخیر طراحی شوند .

۷- مدار جمع کننده BCD :

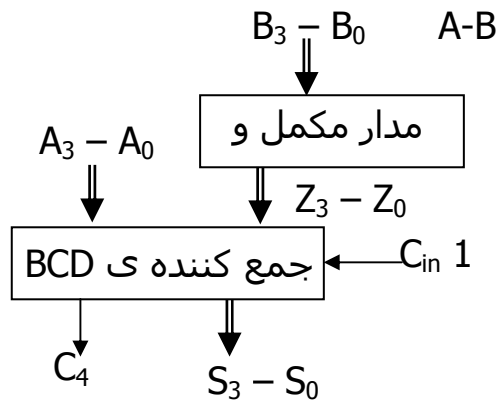


C'_4	S'_3	S'_2	S'_1	S'_0	K
۰	۰	۰	۰	۰	۰
۰	۰	۰	۰	۱	۰
۰	۰	۰	۱	۰	۰
۰	۱	۰	۰	۱	۰
۰	۱	۰	۱	۰	۱
۰	۱	۰	۱	۱	۱
۱	۰	۰	۰	۰	۱
۱	۰	۰	۰	۱	۱
۱	۰	۰	۱	۰	۱
۱	۰	۰	۱	۱	*

پیاده سازی :



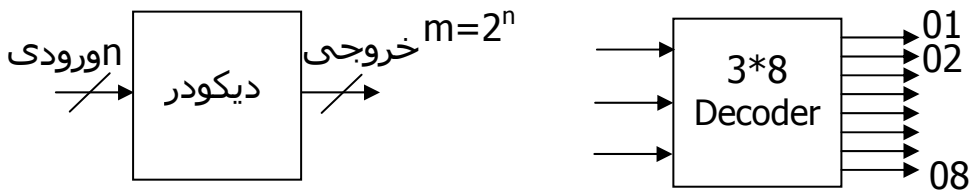
۸- مدارهای تفریق کننده ی BCD :



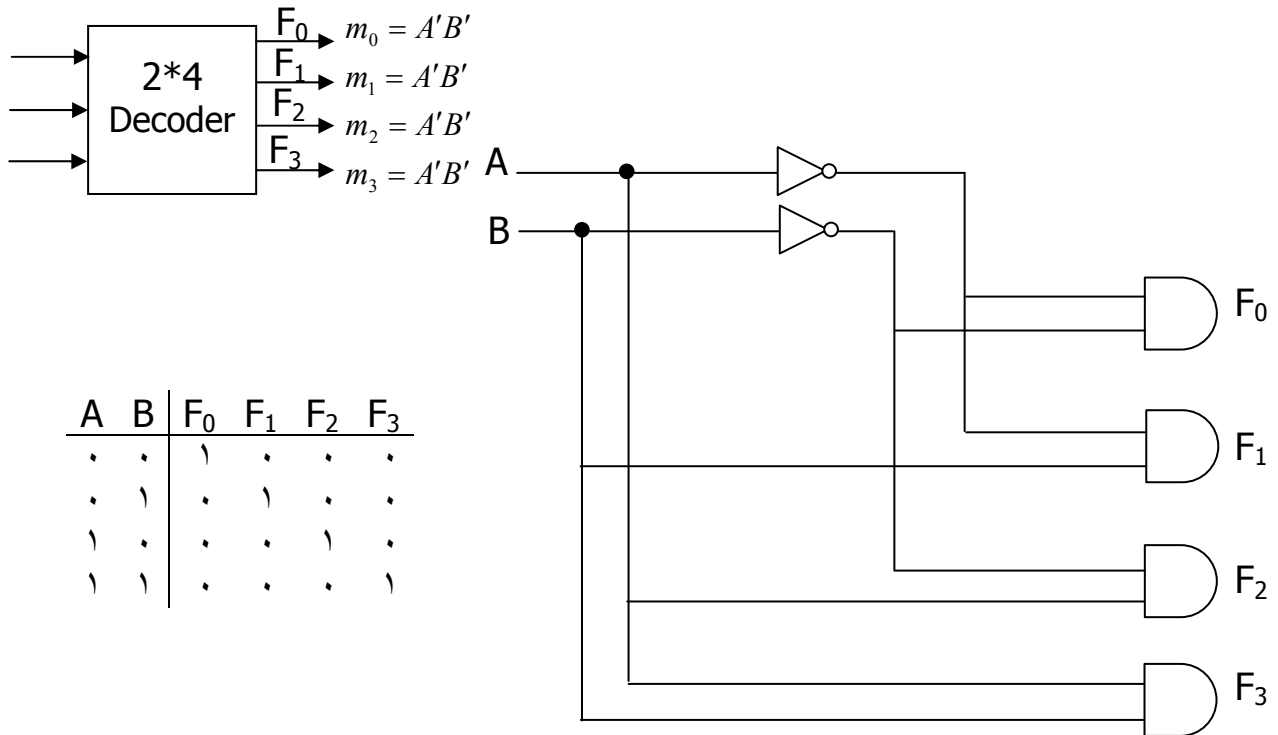
B	B	B	B	Z	Z	Z	Z
3	2	1	0	3	2	1	0
۰	۰	۰	۰	۱	۰	۰	۱
۰	۰	۰	۱	۱	۰	۰	۰
۰	۰	۱	۰	۰	۱	۱	۱
۰	۰	۱	۱	۰	۱	۰	۰
۰	۱	۰	۰	۰	۱	۰	۱
۰	۱	۰	۱	۰	۱	۱	۰
۰	۱	۱	۰	۰	۱	۰	۱
۰	۱	۱	۱	۰	۱	۱	۰
۱	۰	۰	۰	۰	۰	۰	۰
۱	۰	۰	۱	۰	۰	۰	۱
۱	۰	۱	۰	۰	۰	۱	۱
۱	۰	۱	۱	۰	۰	۱	۰
۱	۱	۰	۰	۰	۰	۰	۰
۱	۱	۰	۱	۰	۰	۰	۱
۱	۱	۱	۰	۰	۰	۱	۱
۱	۱	۱	۱	۰	۰	۱	۰

۹- مدارهای دیکودر (Decoder) :

مدار دیکودر برای n متغیر ورودی 2ⁿ حالت آنرا ایجاد می نماید.



نکته : حالات مختلف متغیرها را در خروجی می دهد .



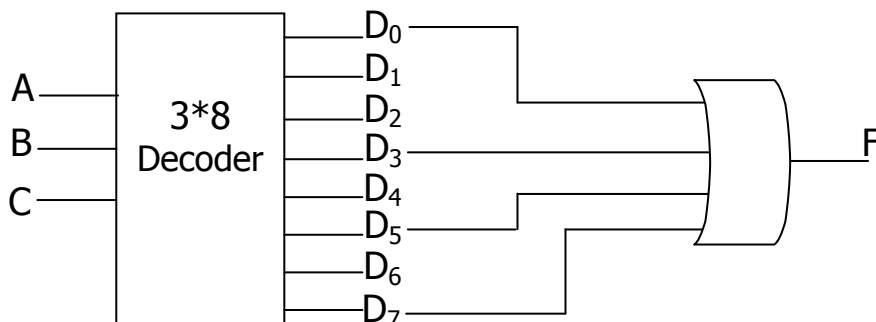
نکته : اگر به جای AND از NAND استفاده کنیم آنگاه Maxterm ها خواهیم داشت :

کاربردها :

۱- در پیاده سازی توابع :

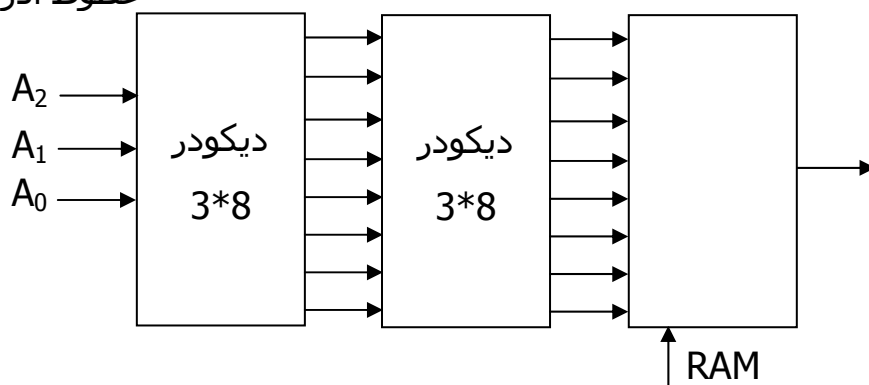
$$F = (A, B, C) = \sum m(0, 3, 5, 7)$$

تابع را به کمک يك ديکودر 3*8 پیاده سازی نمائید .

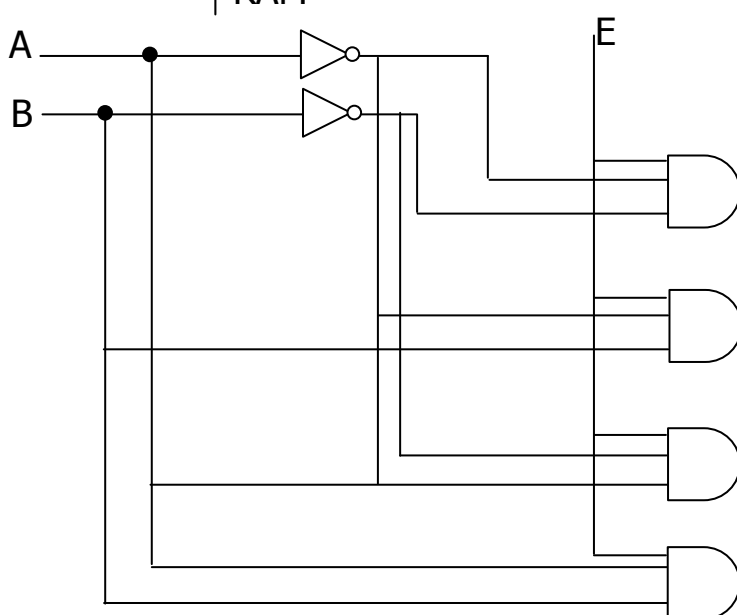
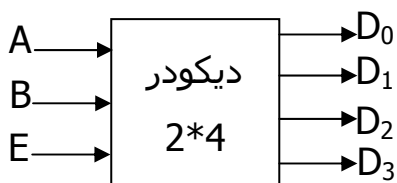


3 bit
خطوط آدرس

۲- در رمز گشایی خطوط آدرس :


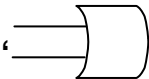


دسترسی به تمام نقاط حافظه که در این مثال 8 حافظه است .



E	A	B	D ₀	D ₁	D ₂	D ₃	
.	*	*	دیکودر عمل نمی کند E=0
\	.	.	\	.	.	.	Enable
\	.	\	.	\	.	.	Active High enable
\	\	.	.	.	\	.	
\	\	\	.	.	.	\	

نکته : در حالت ماکسترمی نیز در صورتی که E=0 ، A, B نیز Don't care می باشد .

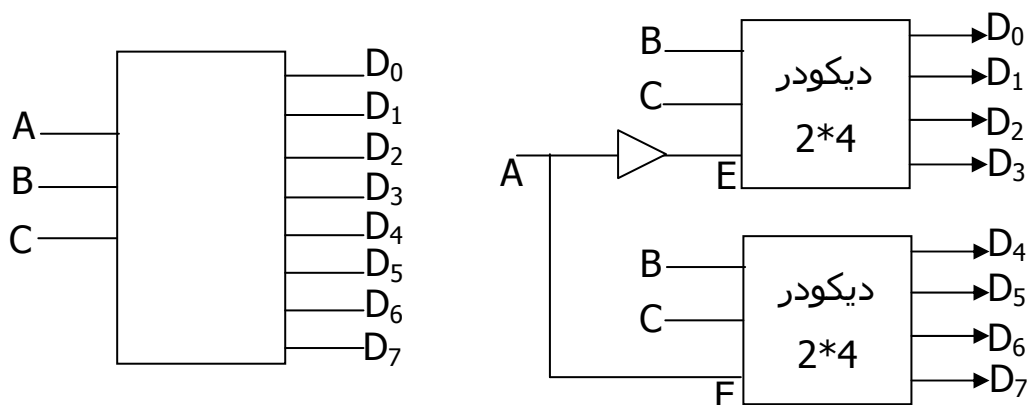
در صورتی که به جای  ،  بگذاریم جدول زیر را خواهیم داشت .

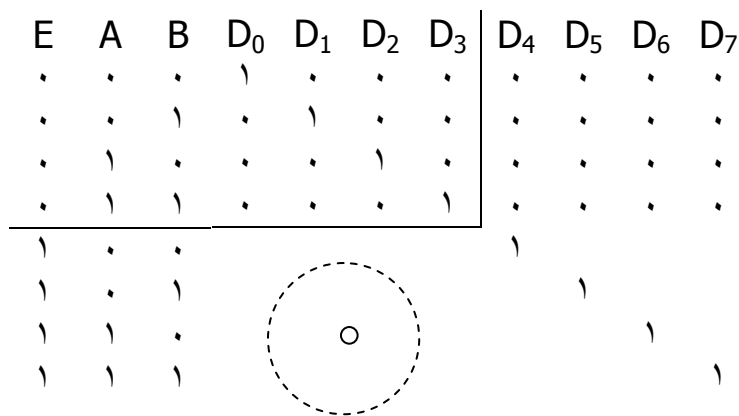
E	A	B	D ₀	D ₁	D ₂	D ₃	
1	*	*	1	1	1	1	Enable
0	0	0	0	1	1	1	
0	0	1	1	0	1	1	Active High enable
0	1	0	1	1	0	1	
0	1	1	1	1	1	0	

یعنی Enable باید NOT شود به صورت زیر

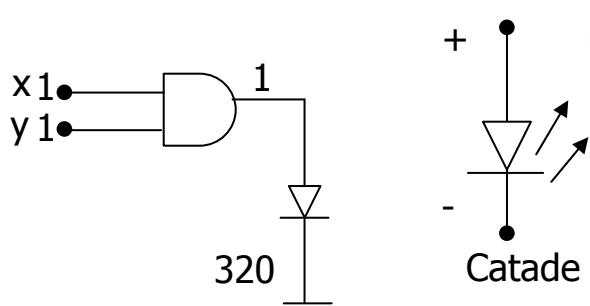


با کمک دیکودر های 2*4 ، دیکودر 3*8 بسازید .





دیکودر BCD به 7.seg (Seven Segment)



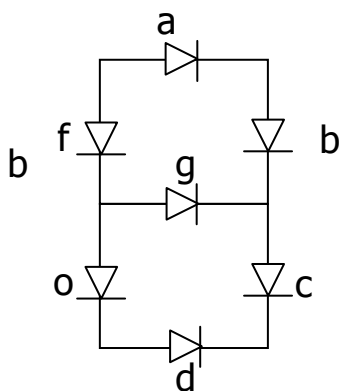
قطعه الکترونیکی LED : (دیود نوری) با نماد

(Light Emitting Diode)

اگر $2[V]$ به دو سر LED اختلاف پتانسیل اعمال شود روشن می شود .

در نتیجه :

7.seg صفحه نمایش یا خروجی یک سیستم دیجیتال می تواند باشد .

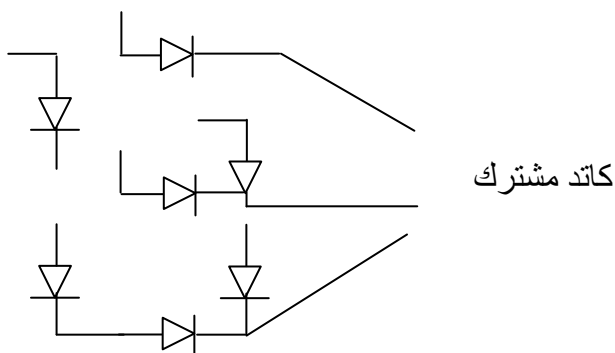


نکته :

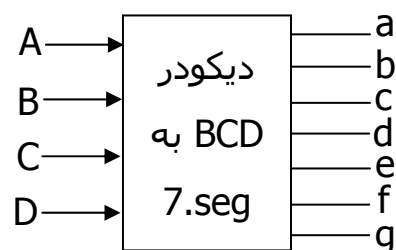
اگر کاتدها را به هم وصل کنیم و از طریق آنها فرمان بدهیم می گویند (کاتد مشترک

7.seg) و اگر آن را به هم متصل و در کل به جزای که می خواهیم روی آن کار انجام

دهیم وصل کنیم و از طریق کاتدها فرمان بدهیم (آنرا مشترک 7.seg)



A	B	C	D		a	b	c	d	e	f	g
۰	۰	۰	۰	0	۱	۱	۱	۱	۱	۱	۰
۰	۰	۰	۱	1	۰	۱	۱	۰	۰	۰	۰
۰	۰	۱	۰	2	۱	۱	۰	۱	۱	۰	۱
۰	۰	۱	۱	3	۱	۱	۱	۱	۰	۰	۱
۰	۱	۰	۰	4	۰	۱	۱	۰	۰	۱	۱
۰	۱	۰	۱	5	۱	۰	۱	۱	۰	۱	۱
۰	۱	۱	۰	6	۰	۰	۱	۱	۱	۱	۱
۰	۱	۱	۱	7	۱	۱	۱	۰	۰	۰	۰
۱	۰	۰	۰	8	۱	۱	۱	۱	۱	۱	۱
۱	۰	۰	۱	9	۱	۱	۱	۱	۰	۱	۱

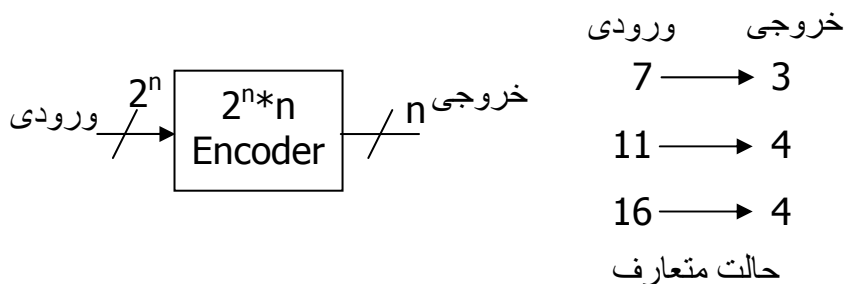


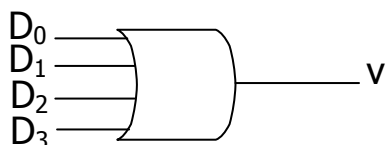
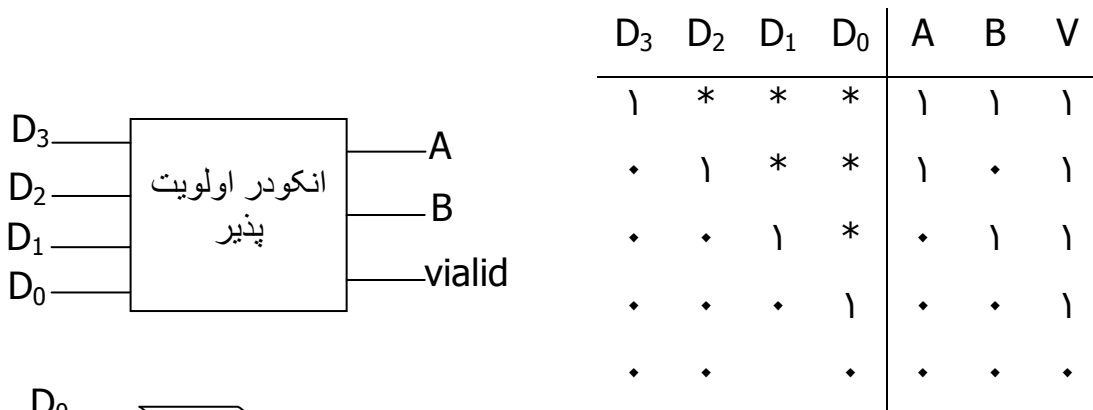
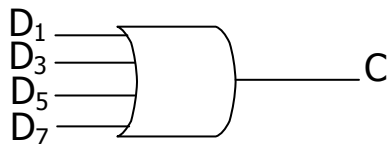
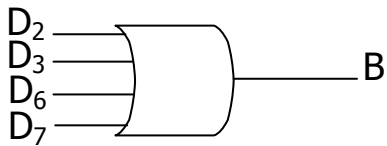
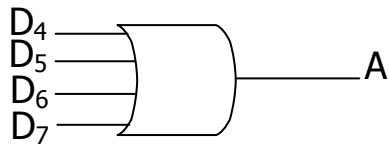
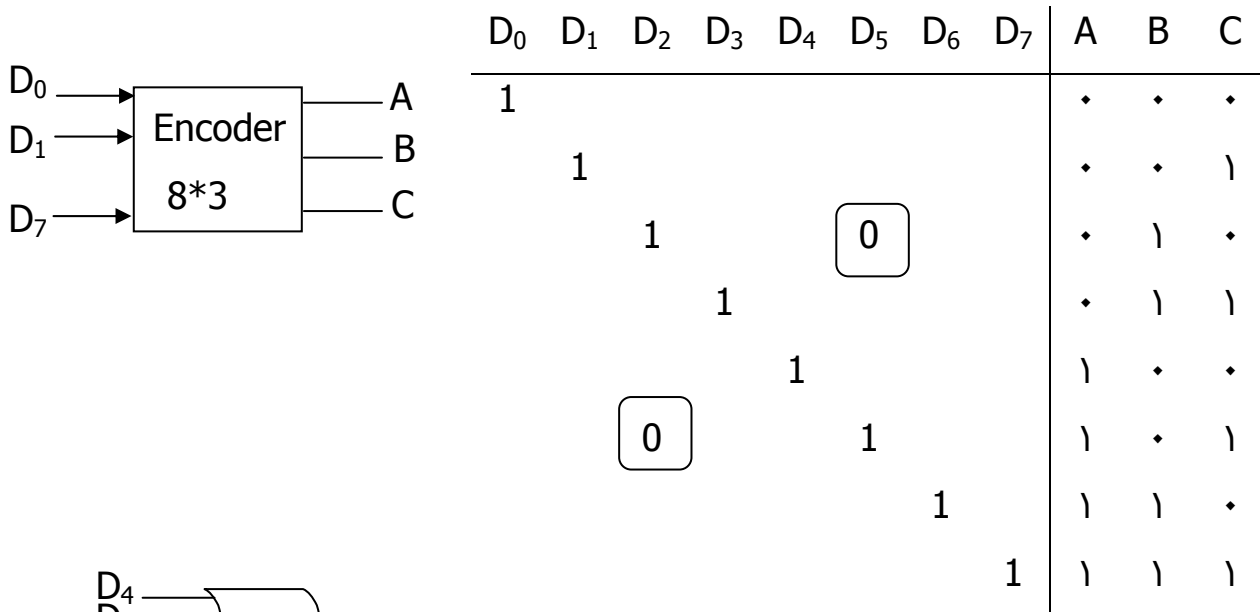
باقی مانده Don't care هستند

نکته :

۱۰- مدارهای Encoder یا رمز گذار :

مدار انکودر برای رمزگذاری یا فشرده سازی متغیرهای ورودی بکار می رود.



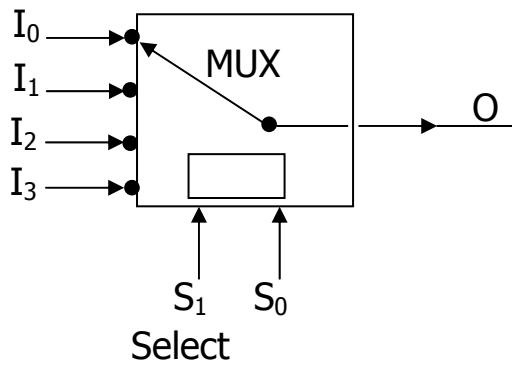


$$A = D_3 + D_2 + D_3' = (D_3 + D_2)(D_3 + D_3') = D_3 + D_2$$

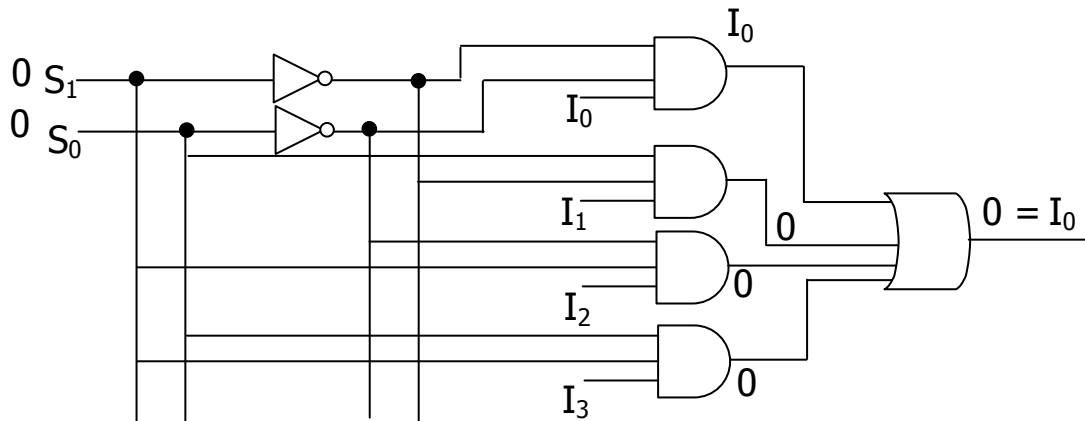
$$B = D_3 + D_3'D_2D_1 = D_3 + D_2'D_1$$

کاربرد : اولویت گذاری Interrupt مانند وقفه کیبورد ، سدی درایو ، کارت صوتی و VGA
 Multiplexer : با کمک n انتخاب 2^n ورودی را روی یک خروجی می فرستد .

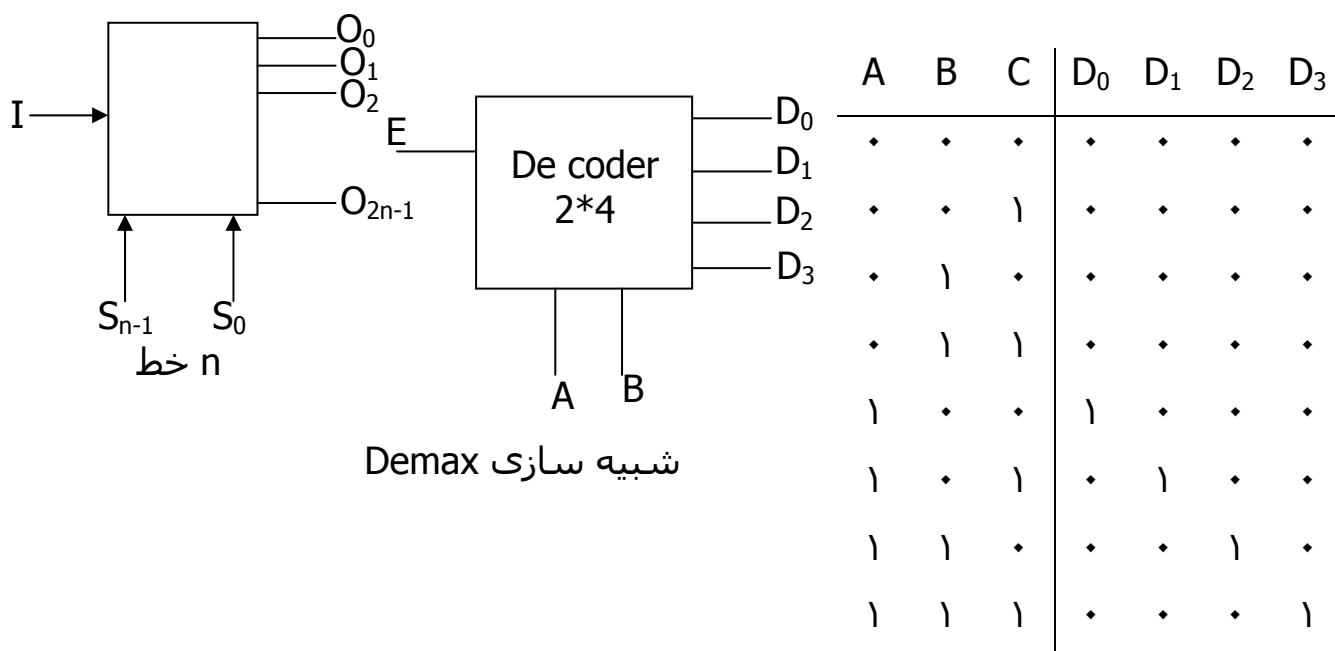
مخفف مالتی پلکسر



S_1	S_0	O
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3



۱۱- مدارهای ترکیبی - دی مالتی پلکسر Demulti plexer :

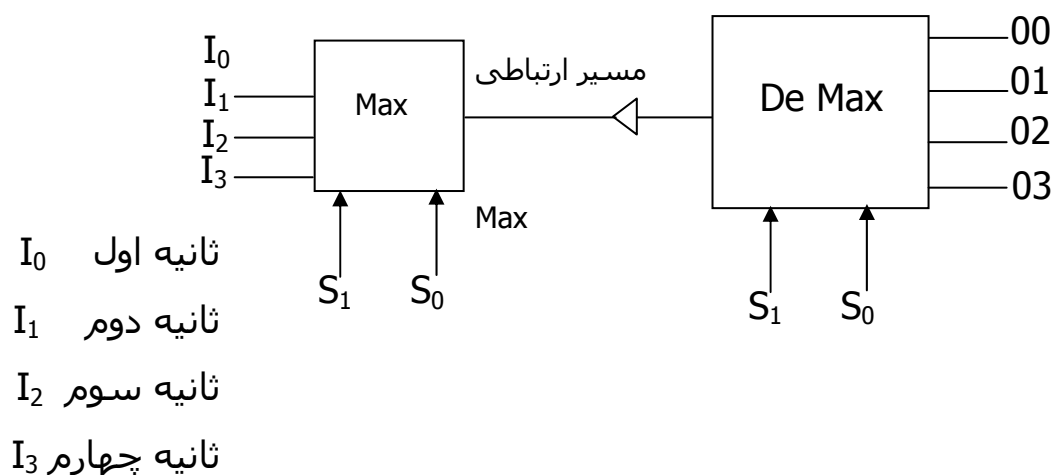


لذا مدار Demax همان دیکودر با یک ورودی Enable است.

کاربرد :

در ارتباط مخابراتی مانند تلفنهای منازل یک منطقه :

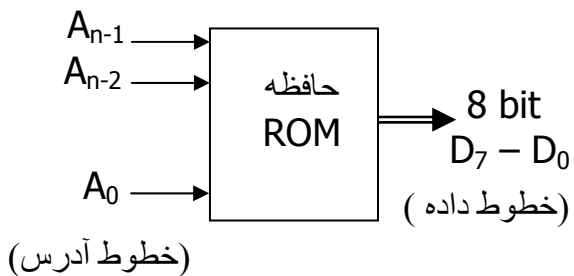
به این روش ارسال Time Domain Multiplexing (TDM) گویند .



۱۲- مدارهای ترکیبی - قطعات قابل برنامه ریزی (ROM)، (PLA)، (CAL)، (PLD)

و ... :

ROM یا Read Only Memory در واقع مدارهای ترکیبی هستند .

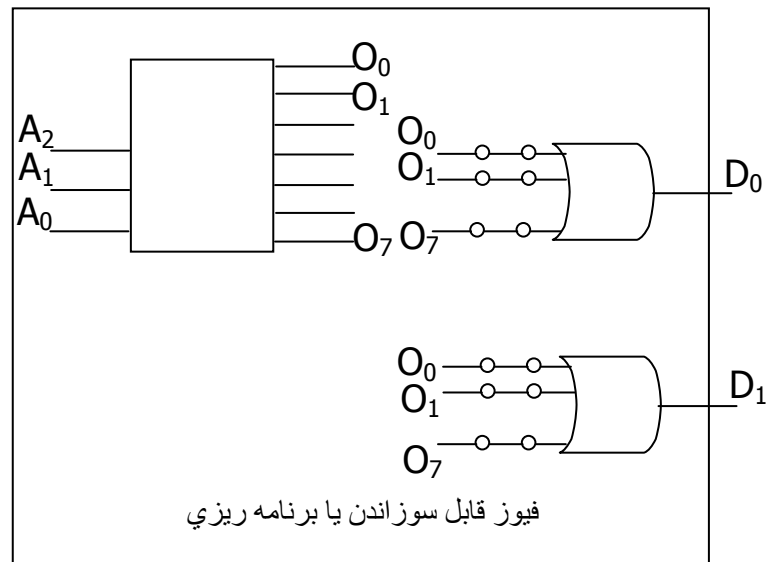


تعداد خطوط آدرس

ROM $2^n * m$

تعداد بیت‌های کلمه ی

خروجی

ROM $8 * 4$

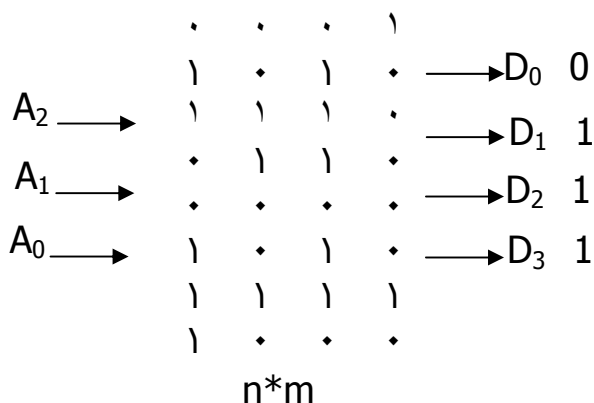
ROM : 8 خانه حافظه - هر خانه 4 بیتی یا 4 bit word

نکته : فیوز های 0,3,4 در D_3 سوزانده می شوند .

نکته : ما در ROM تمام فیوترمها را داریم و با or های قابل برنامه ریزی و با توجه به

اطلاعات مشخص می شود برای هر خروجی کدام فیوز ها سوزانده شود .

کاربرد ROM :



۱- پیاده سازی توابع

۲- یک المان حافظه

استفاده از Rom به عنوان پیاده سازی توابع :

مدارهای ترکیبی داریم که n ورودی و m خروجی دارد و بدین منظور می توان از $2^n * k$ Rom که $k \geq m$ استفاده نمود.

نکته :

چون فیوزهای سوخته شده دیگر قابل باز یافت نیستند لذا Rom ها فقط یکبار قابل خواندن می باشد . برای رفع این مشکل EPROM طراحی شده است . که با اشعه ی ماوراء بنفش پاک می شود و با مدارهای الکتریکی برنامه ریزی می شوند . ولی EPROM ها مدت زیادی را برای طراحی پاک شدن مصرف می کنند (حدود 15-16 دقیقه) لذا EEPROM را طراحی کردند که بوسیله ی مدارهای الکتریکی پاک می شوند که بسیار کم زمان می برند (میلی ثانیه).