دانشکده برق، کامپیوتر و فناوری های پیشرفته

گروه مهندسی برق- الکترونیک

دستورکار آزمایشگاه مدارمنطقی

تهیه کننده:

خانم دکتر بابایی صداقت

تاریخ تنظیم:

مهرماه 1403

فهرست شماره صفحه

1. آشنایی با اصول مدارهای منطقی 3
2. آشنایی با بردبورد و اصول بستن گیتهای پایه بر روی برد 5
3. بدست آوردن جدول صحت برخی از گیتهای منطقی بر روی برد 11
4. طراحی توابع مختلف تنها با گیت پایه NAND 12
5. مقایسه‌گرها 18
6. جمع‌کننده و تفریق کننده 19
7. مالتی‌پلکسر 23
8. انکدر 26
9. سون‌سگمنت + دیکودر + شمارنده 30
10. دیکدر BCD و نمایشگر 7 قسمتی 32
11. لچ و فلیپ‌فلاپ 36
12. فلیپ‌فلاپ T و شیفت رجیستر 40

با تشکر از دانشجویان عباس نژاد، پورسلیمانی و تاج مراغه

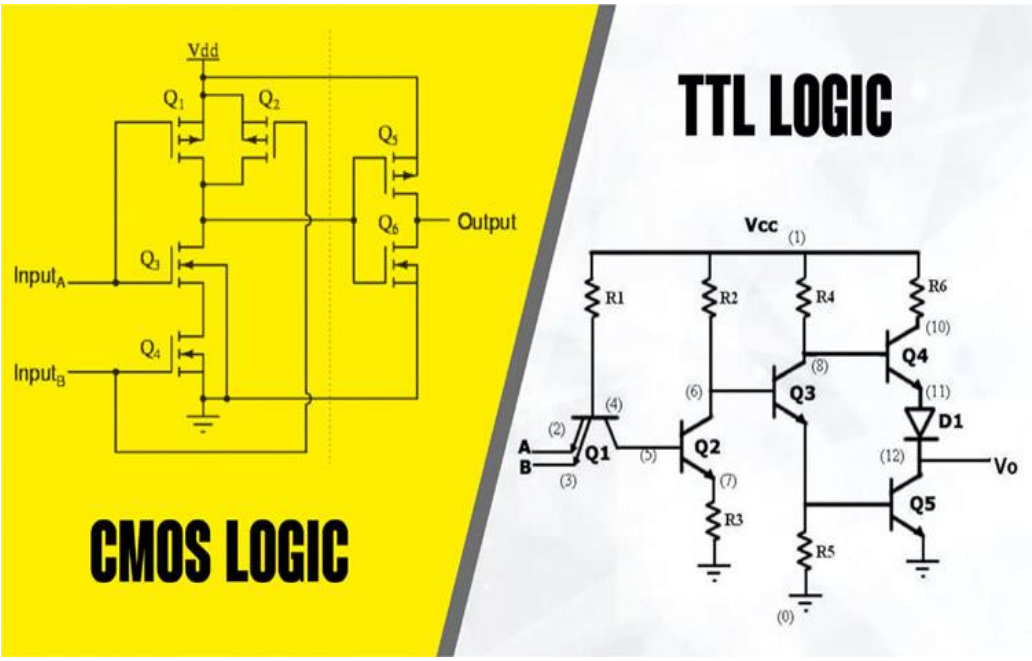
1. آشنایی با اصول مدارهای منطقی

آزمایشگاه مدارهای منطقی یکی از بخش های اساسی در رشته های مهندسی برق و کامپیوتر است که دانشجویان را با اصول و کاربردهای مدارهای دیجیتال آشنا می‌کند. مدارهای دیجیتال با مدارهای مجتمع ساخته می‌شوند. مدار مجتمع یا IC ، یک کریستال نیمه‌هادی است که قطعات الکترونیکی لازم برای گیت های دیجیتال را در خود دارد. تراشه در داخل یک محفظه پلاستیکی یا سرامیکی جای می‌گیرد. اتصالات آن با سیم‌های نازک به پایه های خارجی جوش داده می‌شود تا مدار مجتمع به وجود آید. تعداد پایه های یک IC می‌تواند از 14 پایه در بسته های کوچک تا 64 پایه باشد.

در آزمایشگاه، به منظور آشنایی عمیق تر با مدارهای دیجیتال و نحوه کار با آن‌ها، از فناوری‌های مختلفی همچون TTL و CMOS استفاده می‌شود.

TTL مخفف کلمه Transistor to Transistor Logic است که از ترانزیستورهای BJT ساخته شده‌اند. این خانواده منطقی بر روی پالس‌های جریان مستقیم عمل می‌کند. به طور کلی، اکثر ICهای مربوط به گیت‌های منطقی جزو این دسته هستند. مثالی از ICهای منطقی TTL عبارتند از 74XX

CMOS مخفف کلمه Complementary Metal Oxide Semiconductor است و متداول‌ترین تکنولوژی ساخت MOSEFT (Metal-Oxide-Semiconductor Field-Effect Transistor) می‌باشد.



برخی از تفاوت‌های TTL و CMOS

- اجزای CMOS معمولا گران‌تر از TTL هستند، اما در سطح تراشه ارزان‌تر هستند، زیرا اندازه آنها از اندازه TTL کوچکتر است.
- سطح تغذیه در خانوادهTTL ، 5 ولت و در خانواده CMOS بین 3 تا 15 ولت است.

- مدارهای CMOS نسبت به TTL ایمنی بهتری در برابر نویز دارند.
- گیت‌های اساسی در ساخت TTL ، NAND و در ساخت CMOS گیت‌های NOR و NAND هستند.
- به دلیل اینکه در CMOS از FET و مدارهای TTL از BJT استفاده شده‌است، تراشه‌های CMOS بسیار سریع‌تر و کارآمدتر هستند.
- توان مصرفی مدارهای TTL بیشتر از CMOS است.

خروجی	TTL	CMOS
HIGH – سطح منطقی 1	بیش از 2V	بیش از 2/3 تغذیه
LOW – سطح منطقی 0	کمتر از 0.8V	کمتر از 1/3 تغذیه
نامعتبر	بین 0.8V و 2V	بین 1/3 و2/3 تغذیه

- در زمان استفاده از خانواده TTL توجه داشته باشید که برخی از انواع آن به قرار زیر است:

TTLاستاندارد (سری74)

TTLسرعت بالا (سری74H)

TTLکم مصرف (سری74L)

TTLشاتکی (سری74S)

TTLشاتکی کم مصرف (سری74LS)

TTLشاتکی کم مصرف پیشرفته (سری74ALS)

مشخصه خانواده‌های منطقی دیجیتال معمولاً با تحلیل گیت مبنا در هر خانواده با یکدیگر مقایسه میگردند. برای این کار باید پارامترهای توان مصرفی، تأخیر در انتشار و مصونیت در مقابل نویز را در نظر داشت.

	DTL	TTL						ECL	MOS	CMOS
		74	74H	74L	74S	74LS	74 ALS			
تاخیر درانتشار(ns)	۳۰	۱۰	۶	۳۳	۳	۹	۴	۲	۱۰۰تا۵۰	۲۵
توان‌مصرفی‌هردروازه(mw)	۱۵	۱۰	۲۲/۵	۱	۲۰	۲	۱	۲۵	۰/۲۵	^{nw}۱۰
برون دهی (f.o)	۸	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۲۵	>۵۰	>۵۰
مصونیت در مقابل نویز	خوب	خوب	خوب	خوب	خوب	خوب	خوب	متوسط	عالی	عالی

هر آی‌سی در یک بسته با پایه‌های خاص بسته‌بندی میشود. نحوه صحیح خواندن پایه‌های یک تراشه و نیز روش تشخیص اولین پایه آن و مشخصات دیگری که برای شناسایی و کار با تراشه لازم است، ابزارهای اولیه کار در آزمایشگاه مدارهای منطقی هستند. چنانکه در شکل زیر نیز مشخص است در محل پایه 1 آی‌سی علامت خاصی وجود دارد. از جمله مشخصاتی که بر روی تراشه حک می‌گردد، علامت کارخانه سازنده است که میتواند یک یا چند حرف و یا شکل باشد. نام تراشه نیز توسط شماره مشخص می‌شود. البته باید توجه داشت که روش شماره‌گذاری در شرکت‌های مختلف یکسان نبوده و در رابطه با شرکت‌های مختلف باید به برگه اطلاعات (data sheet) منتشر شده

همان شرکت مراجعه نمود. اطلاعات دیگری که ممکن است بر روی تراشه مشاهده گردد، شامل تاریخ ساخت تراشه (دو رقم برای سال و دو رقم برای هفته)، نام کشور سازنده و شماره انبار باشد.

M: سازنده آی‌سی/ 4021: شماره آی‌سی / B: Buffered CMOS

- : نشانه شروع شمارش پایه/ CP: Data Code/ 78: 1978

24_{th} week:24

به هر آی‌سی شماره‌ای اختصاص می‌یابد که بر روی آن به‌منظور شناسایی چاپ شده است. تولیدکنندگان کتابچه‌هایی را چاپ نموده‌اند که حاوی توضیحات و کلیه اطلاعات مربوط به آی‌سی‌های تولید شده آنها است. به‌عنوان مثال، 6 گیت NOT را داخل یک آی‌سی قرار داده‌اند و به آن شماره 7404را اختصاص داده‌اند.

نکات ایمنی

همیشه دقت کنید که ولتاژ و جریان مورد استفاده در محدوده ایمن باشد تا از آسیب به قطعات و تجهیزات جلوگیری شود.

قبل از اتصال منبع تغذیه، مطمئن شوید که اتصالات صحیح و محکم هستند.

مراقب باشید که هیچ‌گاه دو پایه مجاور را به طور ناخواسته به هم وصل نکنید، زیرا این امر ممکن است به قطعات آسیب برساند.

نحوه نتیجه‌گیری از آزمایش‌ها

برای نتیجه‌گیری از هر آزمایش، مراحل زیر را دنبال کنید:

مشاهده و ثبت نتایج: نتایج حاصل از پیاده‌سازی مدارها را با دقت مشاهده و ثبت کنید.

تحلیل نتایج: نتایج به دست آمده را تحلیل کنید و با تئوری‌های مطرح شده در درس مطابقت دهید.

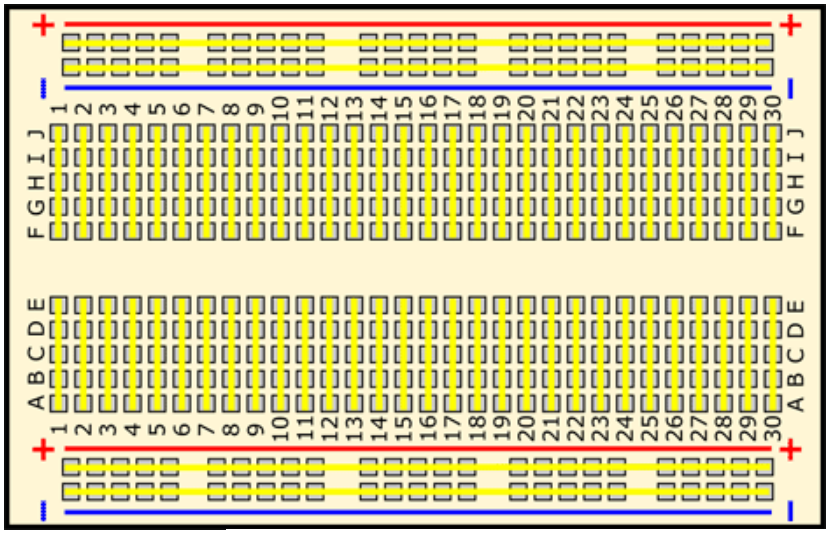
مقایسه با پیش‌بینی‌ها: نتایج عملی را با پیش‌بینی‌های تئوریک مقایسه کنید و دلیل اختلاف‌ها را بررسی کنید.

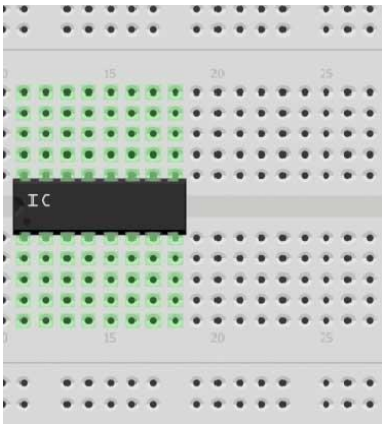
ارائه گزارش: به‌صورت گروهی و با هماهنگی بین اعضا، گزارشی کامل از مراحل انجام آزمایش، نتایج و تحلیل‌های خود تهیه کنید و در انتهای ترم به استاد مربوطه ارائه دهید.

2. آشنایی با بردبورد و اصول بستن گیتهای پایه بر روی برد

خلاصه آزمایش: در این آزمایش قصد داریم با ساختار بردبورد و نحوه ی بستن مدار بر روی برد آشنا شویم. سپس گیت‌های NOT ، AND و OR بر روی بردبورد می‌بندیم و جدول صحت آن‌ها را بدست می‌آوریم.

ابتدا شرح مختصری از بردبورد داریم. بردبورد (Bread Board) یک تخته معمولا پلاستیکی است که سوراخ‌های کوچکی دارد و به راحتی می‌توان قطعات الکترونیکی را در آن قرار داد و نمونه‌های اولیه مدارها را پیاده‌سازی کرد. خطوط طویل رنگی روی بردبوردها با دو رنگ قرمز و آبی به ترتیب با علامت‌های مثبت و منفی مشخص می‌شود. این خطوط باس (Bus) یا شین نامیده می‌شوند و برای اتصال زمین یا منبع تغذیه به مدار، مورد استفاده قرار می‌گیرند. معمولا ترمینال قرمز با علامت مثبت و ترمینال آبی با علامت منفی نشان داده می‌شود. لازم به ذکر است که بین باس‌های مثبت و منفی تفاوت فیزیکی وجود ندارد و تعیین علامت آن‌ها تنها برای مدیریت ساده‌تر مدار است. هر یک از ردیف‌های درون بردبورد به صورت مجموعه‌های پنج‌تایی به هم متصل هستند (در واقع هم پتانسیل می‌باشند). این بدین معنی است که هر پنج سوراخ، یک نیم ردیف را تشکیل می‌دهند.





تعداد ردیف‌ها ۳۰ تا می‌باشد و هر نیم ردیف دارای پنج حرف انگلیسی است.

در ادامه به کاربرد بردبورد بپردازیم. برد بورد وسیله‌ای برای بستن سریع مدارهای مختلف الکترونیکی جریان و فرکانس پایین است و از آنجا که میتوان به راحتی و به سرعت بسیاری از IC ها و قطعات مختلف الکترونیکی را روی آن سوار و ارتباط‌های لازم را با سیم و بدون لحیم کاری برقرار کرد، کاربرد وسیعی در آزمایشگاه‌ها و مراکز تحقیقاتی دارد.

❋ هنگام نصب آی سی روی بورد باید توجه کنیم که فرو رفتگی آن سمت چپ قرار داشته باشد.

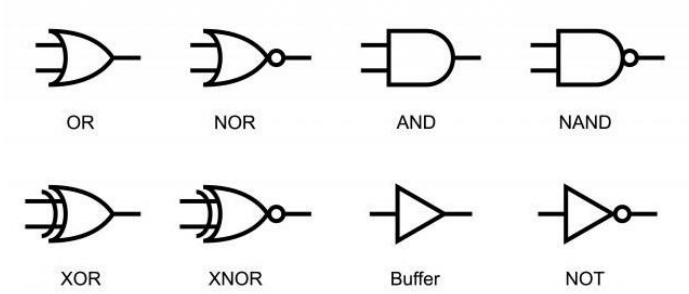
گیت‌های منطقی، به عنوان عناصر اصلی در الکترونیک دیجیتال، برای انجام عملیات منطقی بر روی سیگنال‌های دیجیتال استفاده می‌شوند. این گیت‌ها معمولا با استفاده از ترانزیستورها و مقاومت‌ها ساخته می‌شوند و وظیفه انجام عملیات منطقی مانند جمع، ضرب، نمایش، تغییر و ترکیب سیگنال‌های دیجیتال را بر عهده دارند. یک گیت منطقی ورودی‌های خود را دریافت کرده و طبق استانداردهای منطقی، خروجی مشخصی را تولید می‌کند.

اساس کار گیت منطقی، اعداد باینری (0 و 1) است. این گیت‌ها روی یک یا چند ورودی عملیات منطقی انجام می‌دهند تا در نهایت یک خروجی حاصل شود. سطح ولتاژ ورودی و خروجی گیت‌ها، با دو مؤلفه High Voltage و Low Voltage سنجیده می‌شود. زمانی که سطح ولتاژ High باشد، به آن 1 منطقی و زمانی که سطح ولتاژ Low باشد، 0 منطقی به آن اطلاق می‌شود.

هر گیت منطقی از یک جدول صحت (Truth Table) پیروی می کند که ترکیب های ممکن ورودی و خروجی به دست آمده مربوطه را نشان می دهد. هفت گیت منطقی اصلی وجود دارد: AND، OR، XOR، NOT، NAND، NOR و XNOR

که همه ی این گیت ها را می توان با سه گیت پایه AND، OR، و NOT تشکیل داد.

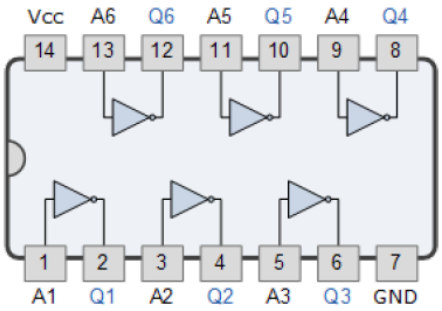
تابع منطقی	نماد بولی
AND	A.B
OR	A+B
NOT	$\overline{A}$
NAND	$\overline{A.B}$
NOR	$\overline{A+B}$
EX-OR	$(A.\overline{B})+(\overline{A}.B)$ or $A \oplus B$
EX-NOR	$(A.B)+(\overline{A}.\overline{B})$ or $\overline{A \oplus B}$



IC: تراشه، مدار یکپارچه، مدار مجتمع یا آی‌سی (Integrated Circuit یا Chip) به مجموعه‌ای از مدارات الکترونیکی اطلاق می‌گردد که با استفاده از موادنیمه‌رسانا (عموماً سیلیکون همراه با میزان کنترل شده‌ای ناخالصی) در ابعادی کوچک (معمولاً کمتر از یک سانتی متر مربع) ساخته می‌شود. هر تراشه معمولاً حاوی تعداد بسیار زیادی ترانزیستور می‌باشد که با استفاده از فناوری پیچیده‌ای در داخل یک لایه از سیلیکن همگون و با ضخامتی یکنواخت و بدون ترک تزریق شده‌اند. هرIC دارای تعدادی پایه می باشد که هر کدام دارای شماره مشخص می باشد.

مرحله 1) بدست آوردن جدول صحت گیت NOT

آی سی 7404 از انوع آی سی های دیجیتال است که شامل 6 عدد گیت NOT می باشد. پین دیاگرام این آی سی به‌صورت زیر است. همانطور که مشهود است پایه 7 به زمین و پایه 14 به تغذیه متصل می شود. پایه های A ورودی و پایه های Q خروجی می باشند.



نکته: قبل از بستن مدار از سالم بودن تک تک سیم های مورد استفاده و LED با آوومتر اطمینان حاصل کنید.

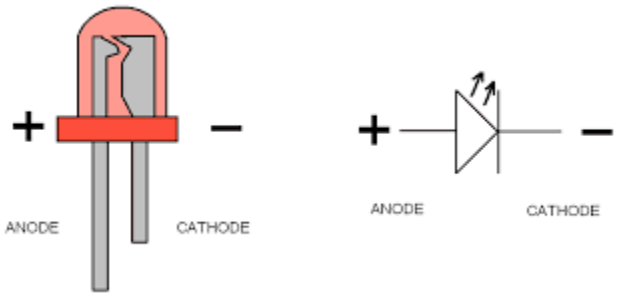
بعنوان یک قرارداد، نحوه بدست آوردن جدول صحت اولین IC مورد بحث بطور کامل توضیح داده می شود.IC7404 در مرکز بردبورد قرار می گیرد(بطوریکه هیچکدام از پایه ها هم پتانسیل نباشند). سر مثبت منبع تغذیه به ردیف بالایی و سر منفی تغذیه به ردیف پایینی برد متصل می شود. سپس پایه 7 آی سی به ردیف پایینی و پایه 14 به ردیف بالایی برد متصل می شوند (ردیف بالایی برد معرف منطق 1 و ردیف پایین برد معرف منطق 0 است که می توان از آنها برای قسمتهای مختلف مدار استفاده کرد). حال ورودی را بعنوان مثال به پایه 1 وصل می کنیم و با توجه به پین دیاگرام باید خروجی را از پایه 2 بگیریم. برای ارزیابی منطقی خروجی از یک LED سری با مقاومت 220 اهمی استفاده می شود که سر آند آن به پایه مورد نظر و سر کاتد آن به مقاومت وصل می‌شود (عدم استفاده از مقاومت سری و اعمال ولتاژ مستقیم به LED، بسته به مقدار ولتاژ اعمالی باعث آسیب دیدن و حتی سوختن LED خواهد شد).

تست سلامت LED:

1- ابتدا اهم متر را روی حالت بوق قرار دهید.
۲- رابط‌های اهم متر را به پایه‌های ال‌ای دی وصل کنید.
۳- اگر اهم متر صدا داد به معنای این است که ال‌ای دی از داخل اتصال کوتاه می‌باشد وخراب است.

روش های تشخیص پایه های آند و کاتد LED:

عموما پایه‌ی بلندتر معرف آند می باشد، اما درصورتیکه پایه های آن کوتاه شده باشند، می توان با استفاده از شکل زیر پایه های LED را تشخیص داد.

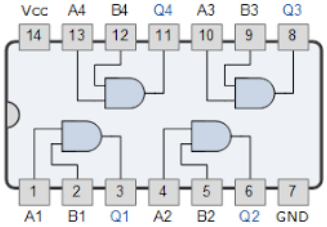


1- جدول زیر را بر اساس ورودی اعمالی پر کنید. اثر مدار باز بودن ورودی را چگونه تفسیر می کنید؟

Input	Output
Low	
High	
Open Circuit(OC)	

مرحله2) بدست آوردن جدول صحت گیت **AND**

آی سی 7408 شامل 4 عدد گیت AND دو ورودی می باشد. پین دیاگرام این آی سی به‌صورت زیر است. همانطور که مشهود است پایه 7 به زمین و پایه 14 به تغذیه متصل می شود. پایه های A, B ورودی و پایه های Q خروجی می باشند.

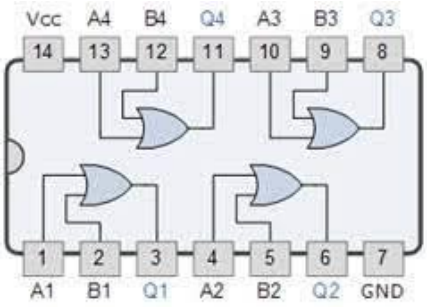


2- جدول زیر را براساس ورودی های اعمال شده پر کنید.

A	B	Output
L	L	
L	H	
H	L	
H	H	
OC	OC	
L	OC	
OC	L	
H	OC	
OC	H	

مرحله**3**) بدست آوردن جدول صحت گیت **OR**

آی سی 7432 دارای 14 پایه است که شامل 4 عدد گیت **OR** دو ورودی می باشد. یونیت شامل 8 ورودی و 4 خروجی می باشد. پین دیاگرام این آی سی به‌صورت زیر است. همانطور که مشهود است پایه 7 به زمین و پایه 14 به تغذیه متصل می شود. پایه های A, B ورودی و پایه های Q خروجی می باشند.



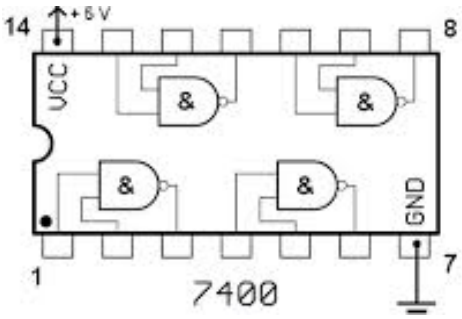
3- جدول زیر را براساس ورودی های اعمال شده پر کنید.

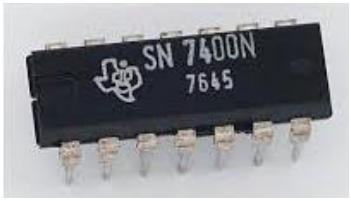
A	B	Output
L	L	
L	H	
H	L	
H	H	
OC	OC	

L	OC	
OC	L	
H	OC	
OC	H	

مرحله4) بدست آوردن جدول صحت گیت **NAND**

آی سی 7400، دارای 14 پایه و شامل چهار گیت NAND دو ورودی می باشد که پین دیاگرام آن در زیر نشان داده شده است. گیت NAND منطقی به عنوان گیت کلی (UNIVERSAL GATE) طبقه‌بندی می‌شود زیرا یکی از متداول‌ترین انواع گیت منطقی است.





4- جدول زیر را براساس ورودی های اعمال شده پر کنید.

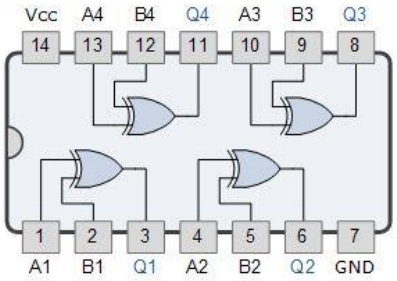
A	B	Output
L	L	
L	H	
H	L	
H	H	

3. بدست آوردن جدول صحت برخی از گیتهای منطقی بر روی برد

خلاصه آزمایش: در این آزمایش آی سی مربوط به گیت‌های NAND، NOR و XOR بر روی بردبورد بسته می‌شوند و جدول صحت و پین دیاگرام آن‌ها را بدست می‌آورید.

مرحله1) بدست آوردن جدول صحت گیت XOR

آی سی 7486 از سری آی سی های TTL شامل 4 عدد گیت XOR دو ورودی می باشد که دارای 8 ورودی و 4 خروجی است. وضعیت خروجی بر اساس حالت ورودی ها تعیین می شود. در شکل زیر نمای شماتیک پایه های IC7486 نشان داده شده است.



1- جدول زیر را براساس ورودی های اعمال شده پر کنید.

A	B	Output
L	L	
L	H	
H	L	
H	H	

مرحله2) بدست آوردن جدول صحت IC7402

2- آی‌سی 7402 را بر روی برد ببندید. ورودی ها را به پایه‌های 2 و3 اعمال کنید و خروجی را از پایه‌ی 1 بگیرید. با استفاده از جدول صحت بدست آمده، تابع را بدست آورید. سپس پین دیاگرام آن را بدست آورید.

پایه 2	پایه 3	خروجی- پایه 1
0	0	
0	1	
1	0	
1	1	

مرحله 3) بدست آوردن پین دیاگرام و جدول صحت **IC7410**

3- آی‌سی 7410 را بر روی برد ببندید. ورودی ها را به پایه‌های 1, 2, 13 اعمال کنید و خروجی را از پایه‌ی 12 بگیرید. با استفاده از جدول صحت بدست آمده، تابع عملکرد آن را بدست آورید. سپس پین دیاگرام این آی‌سی را بدست آورید.

A	B	C	Output
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

مرحله4) بدست آوردن پین دیاگرام و جدول صحت **IC7420**

4- آی‌سی 7420 را بر روی برد ببندید. ورودی ها را به پایه‌های 1, 2, 4, 5 اعمال کنید و خروجی را از پایه‌ی 6 بگیرید. با استفاده از جدول صحت بدست آمده، تابع عملکرد آن را بدست آورید. سپس پین دیاگرام این آی‌سی را بدست آورید.

A	B	C	D	Output
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

4. طراحی توابع مختلف تنها با گیت پایه‌ی NAND

خلاصه آزمایش: مدارهای دیجیتال اغلب به جای اینکه با گیت‌های AND, OR ساخته شوند، با گیتهای NAND, NOR ساخته می‌شوند. ساختن گیت‌های NAND, NOR ساده‌تر بوده و به‌عنوان گیت‌های پایه در تمام خانواده‌های آی‌سی‌های منطقی به‌کار می‌روند.

بدلیل مزایای موجود، قاعده‌ی کلی برای بدست آوردن دیاگرام منطقی NAND از یک تابع بولی به شرح زیر است:

1- تابع را ساده کرده و آن را به فرم جمع حاصلضرب‌ها بنویسید.
2- برای هر جمله ضرب موجود در تابع که حداقل دارای دو متغیر است یک گیت NAND بکشید. ورودی هر گیت NAND متغیرهای آن جمله هستند. این مجموع، گیت‌های طبقه اول را تشکیل می‌دهند.
3- در طبقه دوم یک گیت NAND با ورودی‌هایی که از خروجی‌های طبقه اول می‌آیند، بکشید.
4- یک جمله تک‌متغیری در طبقه اول، نیازمند یک معکوس کننده است.

تابع NOR دوگان تابع NAND است. به همین دلیل همه روال‌ها و قواعد منطقی NOR، دوگان روال و قواعد منطقی NAND می‌باشد. از گیت‌های NAND برای تولید هر نوع عملکرد گیت منطقی نیز استفاده کرد و در عمل گیت NAND اساس بیشتر مدارهای منطقی عملی است.

مرحله1) پیاده‌سازی گیت‌های **NOT, AND, OR, XOR** با استفاده از **IC7400**

1- با استفاده از چمدان آماده بر روی میزکارتان و IC7400، عملکرد گیت‌های NOT, AND, OR, XOR را پیاده‌سازی کنید: ابتدا به‌صورت شماتیک طراحی کنید و سپس بعد از بستن مدار، جدول صحت آن را تکمیل کرده و درستی مدار بسته شده را بررسی کنید.

الف) NOT

Input	Output
0	
1	

ب) AND

A	B	Output
0	0	
0	1	
1	0	
1	1	

ج) OR

A	B	Output
0	0	
0	1	
1	0	
1	1	

د) XOR

A	B	Output
0	0	
0	1	
1	0	
1	1	

مرحله2) پیاده‌سازی عملکرد گیت‌های **AND, NAND** سه ورودی با استفاده از **IC7400**

2-الف) ابتدا شماتیک مداری AND سه ورودی را با استفاده از AND دو ورودی ترسیم کنید و سپس جدول صحت آن را تکمیل کنید.

A	B	C	Output
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

2-ب) ابتدا شماتیک مداری NAND سه ورودی را با ا ستفاده از NAND دو ورودی تر سیم کنید و سپس جدول صحت آن را تکمیل کنید.

A	B	C	Output
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

مرحله3) پیاده‌سازی توابع بولی داده شده با استفاده از **IC7400**

3-الف) تابع بولی $F_1(A,B,C,D) = \sum(0,1,4,5,8,9,10,12,13,14)$ را فقط با اســتفاده از گیت NAND دو ورودی پیاده‌ســازی کنید و بعد از بستن مدار، جدول درستی را تکمیل کرده و درستی مدار پیاده سازی شده را بررسی کنید.

A	B	C	D	Output
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

3-ب) تابع بولی $F_2(A,B,C,D) = \sum(0,1,2,3,4,5,10,11)$ را فقط با استفاده از گیت NAND دو ورودی پیاده‌سازی کنید و بعد از بستن مدار، جدول درستی را تکمیل کرده و درستی مدار پیاده سازی شده را بررسی کنید.

A	B	C	D	Output
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

3-ج) تابع $F_3 = CB + AD$ را با استفاده از گیت NAND دو ورودی طراحی کنید و بعد از بستن مدار، جدول درستی را تکمیل کرده و درستی مدار پیاده سازی شده را بررسی کنید.

A	B	C	D	Output
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	

1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

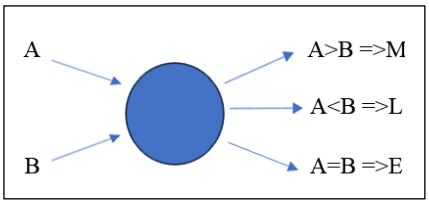
5. مقایسه‌گرها

خلاصه آزمایش: مدار یک مقایسه‌گر تک‌بیتی طراحی و پیاده‌سازی می‌شود و در مرحله‌ی بعد عملکرد IC7485 بررسی می‌شود.

مرحله1) طراحی و پیاده‌سازی مقایسه‌گر تک‌بیتی

1- با توجه عملکردی که از یک مقایسه کننده تک بیتی انتظار داریم، بعد از ترسیم جدول صحت، مدار آن را طراحی کنید و سپس با استفاده از IC7486, IC7400 مدار آن را پیاده‌سازی کرده و صحت مدار طراحی شده را چک کنید.

عملکرد مورد انتظار:



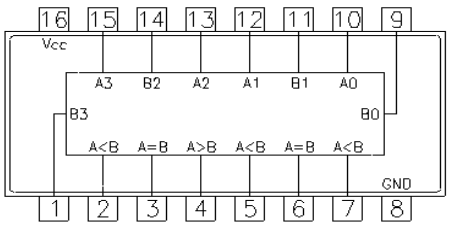
A	B	M	L	E
0	0			
0	1			
1	0			
1	1			

مرحله2) بررسی عملکرد مقایسه‌گر چهاربیتی

در مقایسه اعداد باینری یا BCD بزرگ. همواره برای کاهش زمان ابتدا با ارزش‌ترین بیت‌ها (MSB) با یکدیگر مقایسه می‌شوند. اگر این دو بیت مساوی بودند، به ترتیب تا رسیدن به کم ارزش‌ترین بیت (LSB) بیت‌های کم ارزش‌تر با یکدیگر مقایسه می شوند. در صورت برابری همه‌ی بیت‌ها، دو عدد مساوی بوده و خروجی A=B فعال می‌شود. اما اگر در هر مرحله‌ای از این فرآیند دو بیت با یکدیگر برابر نبوده و مثلاً $A > B$یا $A < B$ باشد، رابطه‌ی بین دو عدد نیز همان‌گونه خواهد بود.

2-الف) با استفاده از بلوک‌های مقایسه‌گر تک‌بیتی، یک مقایسه‌گر چهاربیتی طراحی کنید.

2-ب) IC7485 یک آی‌سی 16پایه است که در واقع مقایسه‌گر چهاربیتی می‌باشد که دو عدد چهاربیتی A3A2A1A0و B3B2B1B0 را با هم مقایسه می کند. 4 بیت A و 4 بیت B را ورودی بدهیم 3 بیت خروجی می گیریم که یا A>B یا A=B یا A<B. پین دیاگرام



7485
4-Bit Magnitude Comparator

این آی‌سی در شکل نشان داده شده‌است. تفاوت پایه‌های 2,3,4 با پایه‌های 5,6,7 چیست و چه کاربردی دارند؟ با استفاده از IC7485 چگونه می توان مقایسه گر هشت بیتی طراحی کرد؟

2-ج) با اعمال ورودی‌ها متناسب با جدول داده شده، جدول را تکمیل کنید.

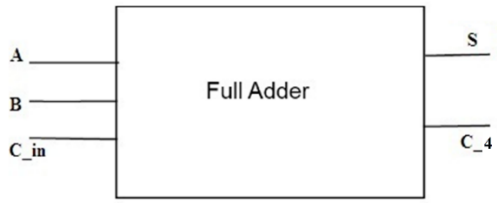
A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0	E	M	L
0	0	1	0	1	0	0	0			
1	0	0	0	0	0	1	0			
1	1	0	0	0	0	1	1			
1	1	0	0	1	1	1	0			
0	0	1	1	1	1	1	0			
0	1	1	1	1	1	1	0			
0	1	1	0	0	0	0	1			
1	1	1	1	1	1	1	1			
1	0	0	0	0	0	0	0			
1	0	0	0	1	1	1	1			
1	0	0	0	0	0	1	1			
1	1	0	0	0	0	1	1			
1	1	0	1	1	1	0	1			
0	1	0	1	1	1	1	1			
0	1	0	0	1	1	0	1			
0	1	0	0	0	1	0	0			

6. جمع کننده و تفریق کننده

خلاصه آزمایش: ابتدا با آی سی‌های 7400 و 7486 یک جمع‌کننده تک بیتی طراحی می‌کنیم و سپس با استفاده از تراشه 7483 که یک جمع‌کننده 4 بیتی است، جدول صحت درستی را برای آن تعیین می‌کنیم.

کامپیوترهای دیجیتال اعمال پردازش اطلاعات گوناگونی را انجام می‌دهند که از آن میان، اعمال مختلف حسابی جز اعمال اصلی به شمار می‌روند و بدون شک اساسی‌ترین عمل حساب، جمع دو رقم دودویی است. این جمع ساده از چهار عمل مقدماتی 0=0+0 و 1=1+0 و 1=0+1 و 10=1+1 تشکیل می‌شود، که حاصل سه جمع اول عددی تک رقمی و آخرین عمل حاصل دو رقمی دارد که رقم مرتبه‌ی بالای آن را نقلی یا carry می‌نامند. مداری که جمع دو رقم دودویی ورودی را انجام داده و حاصلجمع و رقم نقلی را تولید کند، نیم‌جمع‌کننده یا Half-adder نامیده می‌شود. و مداری که جمع سه رقم دودویی(دو رقم دودویی و رقم نقلی حاصل از ستون قبل) را انجام داده و حاصلجمع و نقلی تولید می‌کند، تمام جمع‌کننده یا Full-adder نامیده می شود.

یک تمام جمع‌کننده، جمع مقادیر وارد شده را حساب و همچنین مقادیر نقلی‌شده خروجی را نمایش می‌دهد. یک تمام جمع‌کننده یک بیتی، سه عدد یک بیتی را جمع می‌کند که اغلب به صورت A و B و C_{in} نوشته می‌شود. A و B عملوندها هستند و C_{in} که بیت نقلی که از جمع قبلی (در صورت وجود) است.



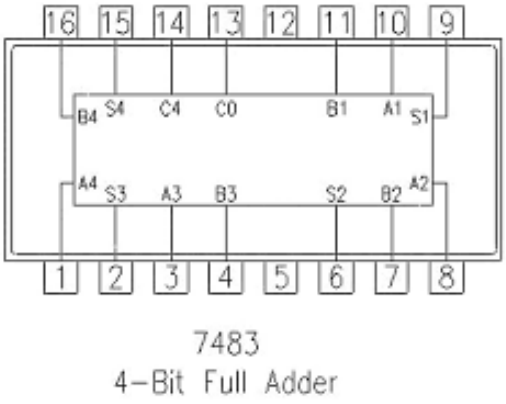
مرحله1) پیاده‌سازی تمام‌جمع کننده تک‌بیتی

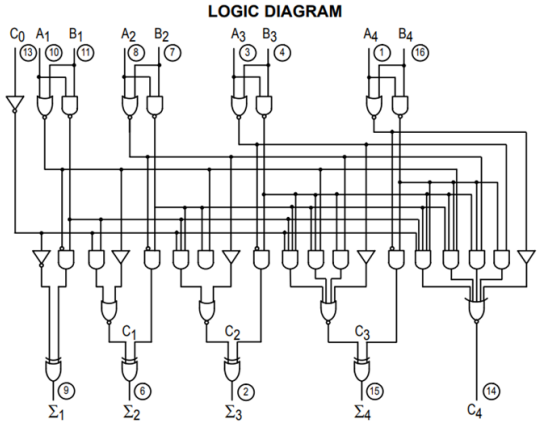
1- با استفاده از IC7486, IC7410 مدار یک تمام جمع کننده‌ی تک‌بیتی را طراحی کرده و سپس پیاده‌سازی کنید.

A	B	C_{in}	C_4	S
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

مرحله2) پیاده‌سازی تمام‌جمع کننده چهاربیتی

آی‌سی 7483 به‌عنوان تمام‌جمع‌کننده استفاده می‌شود. پین دیاگرام و مدار داخلی آن در زیر نشان داده شده است.





2- با استفاده از چمدان‌های روی میزکارتان وIC7483 عملکرد یک جمع‌کننده‌ی چهاربیتی را بررسی کنید. برای این‌کار، مطابق مقادیر داده شده در جدول زیر، جدول را تکمیل کنید.

A_4	A_3	A_2	A_1	B_4	B_3	B_2	B_1	C_{in}	C_4	S_4	S_3	S_2	S_1
0	0	0	0	0	0	0	0	0					
0	0	0	1	0	0	0	1	0					
0	0	1	1	0	0	0	1	0					
1	1	1	1	0	0	0	1	0					
1	1	1	1	0	1	0	0	0					
0	1	1	1	1	0	0	0	0					
0	1	1	0	0	0	0	1	0					
0	1	1	1	0	1	1	1	0					
0	1	1	0	0	0	1	0	1					
0	1	0	0	0	0	1	0	1					
1	0	0	0	0	0	1	1	1					
1	1	0	0	0	0	1	1	1					
1	0	1	0	0	1	0	0	1					

مرحله**3**) پیاده‌سازی تمام‌جمع‌کننده- تفریق‌کننده چهاربیتی با یک بیت کنترلی

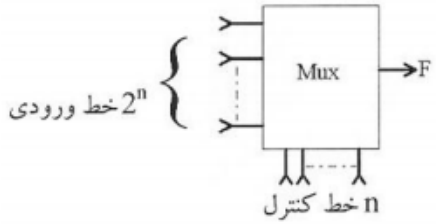
3- با استفاده از IC7483 یک تمام جمع کننده و تفریق کننده همرمان طراحی کنید، به‌صورتی‌که با استفاده از یک بیت کنترلی M بطوریکه اگر M=0 باشد، عمل جمع انجام شود و اگر M=1 باشد، عمل تفریق انجام شود. ابتدا ساختار را طراحی کرده و سپس جدول زیر را تکمیل کنید و صحت مدار خود را بررسی کنید.

A_4	A_3	A_2	A_1	B_4	B_3	B_2	B_1	M	C_4	S_4	S_3	S_2	S_1
0	0	0	0	0	0	0	0	0					
0	0	0	1	0	0	0	1	0					
0	0	1	1	0	0	0	1	0					
1	1	1	1	0	0	0	1	0					
1	1	1	1	0	1	0	0	0					
0	1	1	1	1	0	0	0	0					
0	1	1	0	0	0	0	1	1					
0	1	1	1	0	1	1	1	1					
0	1	1	0	0	0	1	0	1					
0	1	0	0	0	0	1	0	1					
1	0	0	0	0	0	1	1	1					
1	1	0	0	0	0	1	1	1					
1	0	1	0	0	1	0	0	1					

7. مالتی‌پلکسر

خلاصه آزمایش: ابتدا مالتی پلکسر 2 در 1 با استفاده از گیت‌های منطقی طراحی و پیاده سازی می‌شود. سپس توابع 3 و 4 متغیره را با استفاده از IC74151 که یک مالتی پلکسر 8 در 1 است، پیاده‌سازی خواهند شد و در نهایت طراحی مالتی‌پلکسر 4 در یک و طراحی یک مالتی‌پلکسر 16 در 1 با استفاده از MUX4×1 خواسته شد.

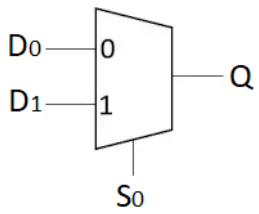
مالتی پلکسر یا به اختصار MUX قطعه‌ای است که با استفاده از انتخاب‌گر یا selector، یکی از چند سیگنال ورودی را انتخاب کرده و سپس آن ورودی انتخاب شده را به یک خط خروجی هدایت می‌کند. تسهیم‌کننده‌ای با 2^n ورودی، n خط انتخاب‌گر دارد که این خطوط انتخاب مشخص می‌کنند که کدام خط ورودی باید به خط خروجی هدایت شود. از تسهیم‌کننده‌ها بیشتر برای افزایش مقدار اطلاعاتی که می‌توان در یک مدت زمان مشخص، با یک پهنای باند مشخص بر روی شبکه ارسال کرد، استفاده می‌شوند. یک تسهیم‌کننده الکترونیکی به چند سیگنال امکان می‌دهد تا از یک دستگاه یا منبع به صورت اشتراکی استفاده کنند. برای مثال می‌توان یک تبدیل‌کننده آنالوگ به دیجیتال یا یک خط ارتباطی را در بین خطوط ورودی به اشتراک گذاشت، به جای اینکه برای هر ورودی یک دستگاه مجزا در نظر گرفته شود.



مرحله1) پیاده‌سازی مالتی پلکسر 2×1

1- برای شروع ابتدا باید مسئله را به شکل بلوک دیاگرامی در بیاوریم.

S_0	Q
0	D0
1	D1



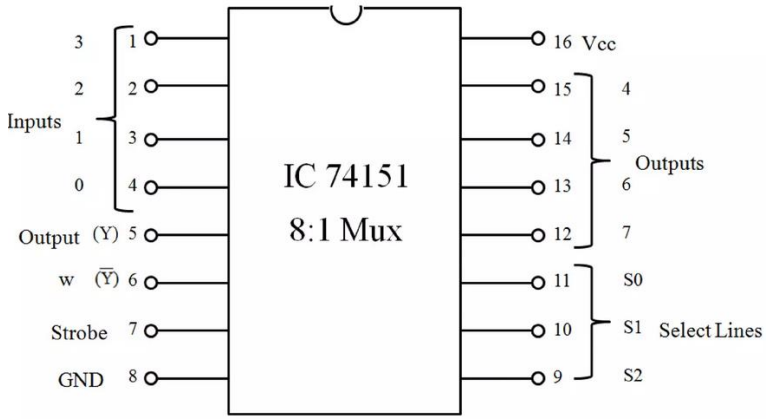
با توجه به عملکردی که از یک مالتی پلکسر 2 در 1 انتظار دارید، جدول زیر را تکمیل کرده و سپس تابع بولی خروجی را بدست آورید. سپس برای پیاده‌سازی تابع به کمک گیت‌های NAND، فقط با استفاده از IC7400 یک مالتی پلکسر 2×1 طراحی و پیاده‌سازی کنید و در نهایت صحت طراحی خود را بررسی کنید.

D_1	D_0	S_0	Q
0	0	0	
0	0	1	
0	1	0	
0	1	1	

1	0	0	
1	0	1	
1	1	0	
1	1	1	

مرحله2) پیاده‌سازی تابع با استفاده از **MUX 8×1**

توابع مختلف را با کمک مالتی پلکسرها می توان پیاده‌سازی کرد. آی‌سی 74151 یک مالتی‌پلکسر است که با استفاده از 3 خط آدرس، یکی از 8 سیگنال ورودی را انتخاب و به خروجی انتقال می‌دهد. پین دیاگرام آن در زیر نشان داده شده است.



2-الف) با استفاده از IC74151 که در چمدان‌های روی میزکارتان است، توابع بولی $F_1(C,B,A)=\sum(1,3,5,7)$ و $F_2(C,B,A)=\sum(0,2,5,6,7)$ را پیاده‌سازی کنید و سپس با پر کردن جدول زیر، صحت عملکرد مدار طراحی شده را بررسی کنید. توجه شود که در این آی‌سی A کم‌ارزش‌ترین بیت و C با ارزش‌ترین بیت می‌باشد.

S_2=C	S_1=B	S_0=A	F_1	F_2
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

2-ب) به نظر شما کدام یک از خروجی‌های 5,6 سریعتر هستند. چرا؟

2-ج) نقش پایه‌ی هفت IC را توضیح دهید و اگر یک باشد، چه تاثیری بر خروجی مدار دارد؟

2-د) تابع $F_3(C,B,A,D)=(0,1,3,4,8,9,15)$ را با استفاده از مالتی‌پلکسر 8 در 1 طراحی و پیاده‌سازی کنید. راهنمایی: ورودی‌های C,B,A بعنوان سلکتور و D را به‌عنوان ورودی در نظر بگیرید.

C	B	A	D	F_3
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

مرحله3)طراحی با استفاده از **MUX 4×1**

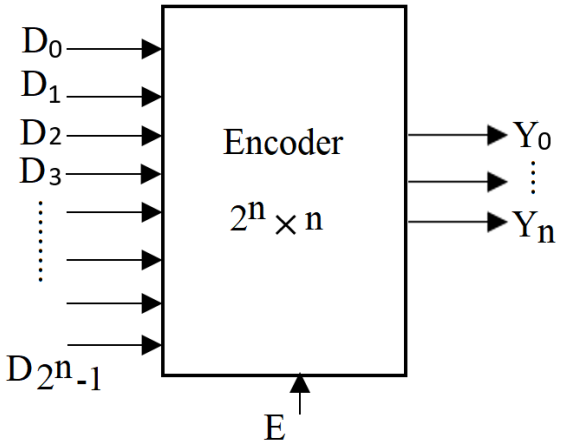
3-الف) با استفاده از مالتی‌پلکسر 4 در1 یک مالتی‌پلکسر 1 در 16 طراحی کنید.

3-ب) مدار داخلی یک مالتی‌پلکسر 4 در 1 را طراحی کنید.

8. انکدر Encoder

خلاصه آزمایش: ابتدا با استفاده از گیتهای منطقی، انکدرهای 4 در 2 و 8 در 3 طراحی و پیاده سازی می‌شود. سپس عملکرد IC 74148 بررسی می شود و سپس انکدر الویت 4 در 2 با استفاده از گیتهای منطقی طراحی و پیاده سازی خواهد شد.

«انکدرهای دیجیتالی» (Digital Encoders) مدارهایی هستند که یک داده‌ی ورودی را گرفته و کد باینری معادل آن را در خروجی تولید می‌کنند. برخلاف مالتی‌پلکسرها که تنها داده‌ی یک خط ورودی را انتخاب کرده و آن را به یک خط خروجی ارسال یا سوئیچ می‌کنند، انکدرهای دیجیتالی یا باینری در هر لحظه از زمان بر اساس مقدار تمامی ورودی‌ها تصمیم‌گیری کرده و یک خروجی باینری معادل تولید می‌کند. لذا می توان گفت انکدر باینری یک مدار منطقی ترکیبی با چند ورودی است که بسته به شماره‌ی پایه‌ای از ورودی که مقدار آن 1 منطقی است، در خروجی مقدار باینری معادل آن شماره را قرار می‌دهد. انکدرهای دیجیتالی بسته به تعداد خطوط داده‌ی ورودی، ممکن است خروجی‌های 2 بیتی، 3 بیتی و یا 4 بیتی تولید کنند. به طور کلی یک انکدر n بیتی دارای 2^n خط ورودی و n خط خروجی است. مرسوم‌ترین انکدرها عبارتند از انکدرهای 4 به 2، 8 به 3 و 16 به 4.



مرحله 1) پیاده‌سازی انکدر به کمک گیت‌های منطقی

1-الف) با توجه به عملکردی که از یک انکدر 4 در 2 انتظار دارید، ابتدا جدول صحت آن را تکمیل کنید و سپس توابع متناظر با خروجی را بدست آورید. در نهایت فقط با استفاده از IC7400 یک انکدر 4 در 2 پیاده‌سازی کنید. راهنمایی: در هر حالت تنها یکی از ورودی ها می-تواند یک باشد.

D_3	D_2	D_1	D_0	Y_1	Y_0

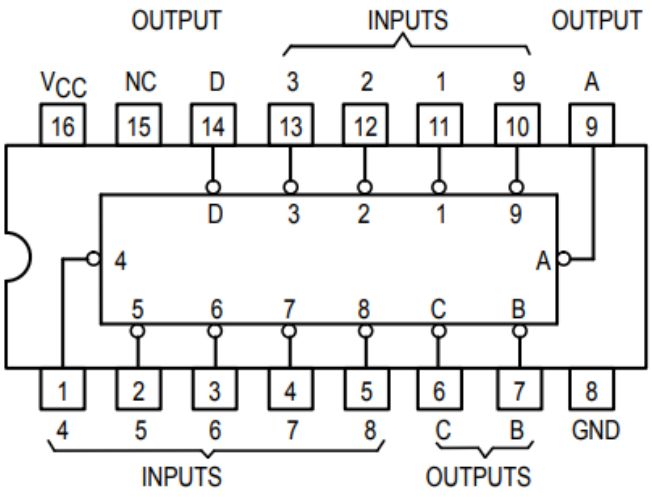
1-ب) با توجه به عملکردی که از یک انکدر 8 در 3 انتظار دارید، ابتدا جدول صحت آن را تکمیل کنید و سپس توابع متناظر با خروجی را بدست آورید. در نهایت فقط با استفاده از IC7432 یک انکدر 8 در 3 پیاده‌سازی کنید.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Y_2	Y_1	Y_0

مرحله2) بررسی عملکرد انکدر الویت **74148**

یکی از معایب اصلی انکدرهای استاندارد دیجیتال، این است؛ که درصورت وجود بیش از یک ورودی در سطح منطقی "1"، می‌توانند کد خروجی اشتباه، تولید نمایند. برای مثال، اگر ما دو ورودی و را، به‌صورت همزمان، در حالت HIGH و سطح منطقی "1" قرار دهیم؛ خروجی نهایی ما نه "01" و نه"10" ، بلکه برابر با "11" خواهد بود؛ که یک خروجی با یک عدد باینری متفاوت از ورودی واقعی ما است. هم‌چنین، کد خروجی برای تمام منطق‌های "0"، زمانی می‌تواند تولید شود؛ که تمام ورودی‌های آن در حالت "0" یا ورودی برابر با "1" باشد.

یک راه ساده برای غلبه بر این مشکل، "اولویت‌بندی" هر پین ورودی است. بنابراین، اگر بیش از یک ورودی، به‌طور همزمان، در سطح منطقی"1" ، موجود باشد؛ کد خروجی واقعی، فقط با ورودی دارای بالاترین اولویت تعیین‌شده، تطابق خواهدداشت. درنتیجه، این نوع از انکدر دیجیتال، معمولا انکدر اولویت یا برای اختصار انکدر -P نامیده می‌شود. انکدر اولویت، با اختصاص سطح اولویت به هر ورودی، مشکلات ذکرشده در بالا را حل می‌کند. خروجی انکدرهای اولویت، مطابق با ورودی فعالی است؛ که بالاترین اولویت را دارد. پس زمانی‌که، یک ورودی با اولویت



بالاتر موجود باشد؛ تمام ورودی‌های دیگری که دارای اولویت‌های پایین‌تری می‌باشند؛ نادیده گرفته می‌شوند. انکدرهای اولویت به شکل آی سی‌های استاندارد تولید می‌شوند؛ مثلاً TTL 74LS148 یک انکدر اولویت 8 به 3 بیتی است که ورودی‌های آن active LOW هستند. این آی‌سی با توجه به ورودی با بالاترین اولویت یک کد 3 بیتی در خروجی تولید می‌کند. شکل پین دیاگرام این آی‌سی در شکل نشان داده شده است.

2-الف) ورودی‌ها را متناسب با جدول به آی‌سی اعمال کنید و خروجی‌ها را بدست آورید.D7 با ارزش‌ترین ورودی می باشد.

(توجه شود همانطور که در شرح آی سی گفته شد، برای گرفتن خروجی درست، باید خروجی را به LED های active low وصل کنید)

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	Y_2	Y_1	Y_0	$\overline{E_I}$	$\overline{E_O}$
X	X	X	X	X	X	X	X				1	
1	1	1	1	1	1	1	1				0	
X	X	X	X	X	X	X	0				0	
0	0	0	0	0	0	0	0				1	
1	0	1	0	1	0	1	0				0	
X	X	X	X	X	X	0	1				0	
X	X	X	X	X	0	1	1				0	
X	X	X	X	0	1	1	1				0	
X	X	X	0	1	1	1	0				0	
X	X	X	X	1	0	0	1				0	
X	X	X	0	1	1	1	1				0	
X	X	X	0	1	1	0	1				0	
X	X	0	1	1	1	1	1				0	
X	X	0	0	1	1	1	1				0	
X	0	1	1	1	1	1	1				0	
X	0	0	0	1	1	1	1				0	
0	1	1	1	1	1	1	1				0	
0	1	0	0	1	1	0	1				0	
0	0	0	0	0	0	0	0				0	
0	1	0	1	0	1	0	0				0	
0	1	0	1	0	1	0	1				0	
0	1	0	1	0	1	0	1				1	

2-ب) نقش پایه‌های E_i, E_o را شرح دهید.

2-ج) یک انکدر الویت 4 در 2 با منطق high active طراحی کنید، بطوری‌که اگر ورودی‌ها اعمال شوند، در خروجی کد متناسب با باارزش‌ترین ورودی ایجاد شود.

راهنمایی1: در هر حالت تعدادی یا همه‌ی ورودی ها می‌توانند مقدار یک باشد.

راهنمایی2: برای خروجی‌ها بطور جداگانه، جدول کارنو تشکیل دهید و بر حسب ورودی‌ها توابع را بدست آورید.

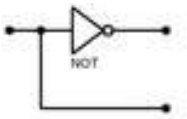
راهنمایی3: در حالتی که همه ورودی‌ها صفر است، خروجی نا معتبر خواهد بود، در آن‌صورت نیاز به تعریف خروجی V داریم که نشان‌دهنده‌ی این حالت باشد.

D_3	D_2	D_1	D_0	Y_1	Y_0	V
0	0	0	0			
0	0	0	1			
0	0	1	0			
0	0	1	1			
0	1	0	0			
0	1	0	1			
0	1	1	0			
0	1	1	1			
1	0	0	0			
1	0	0	1			
1	0	1	0			
1	0	1	1			
1	1	0	0			
1	1	0	1			
1	1	1	0			
1	1	1	1			

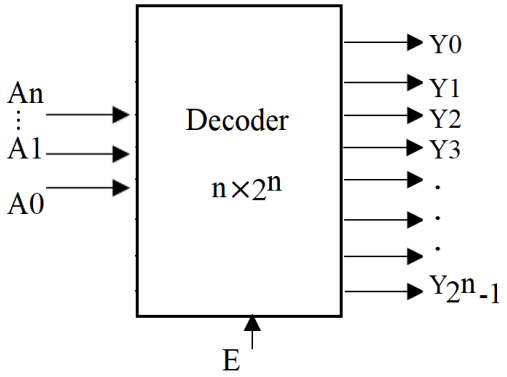
9. دیکدر (Decoder)

خلاصه آزمایش: ابتدا با استفاده از گیتهای منطقی، دیکدر 2 به 4 طراحی و پیاده سازی می‌شود. سپس عملکرد IC 74138 بررسی می‌شود و سپس با استفاده از دیکدر 3 به 8(IC74138)، چند تابع بولی طراحی و پیاده‌سازی خواهد شد.

دیکدرهای باینری (Binary Decoders) دسته‌ای از مدارهای منطقی ترکیبی هستند که در ساخت آنها از گیت‌های منطقی پایه استفاده شده و عملکرد آنها دقیقاً عکس انکدرهاست. کلمه‌ی Decoder به معنای ترجمه یا رمزگشایی اطلاعات رمزنگاری شده از یک قالب به قالبی دیگر است. یک دیکدر 1 به 2 را می توان با استفاده از یک گیت NOT مطابق شکل زیر طراحی کرد.



دیکدرها دارای انواع دو به چهار، سه به هشت، چهار به شانزده و ... هستند. کاربرد اصلی دیکودر در مدارهای دیجیتال، دسترسی به خانه‌های حافظه است. مدار دیکدر می‌تواند شامل یک سیگنال فعال ساز (En) باشد. اگر سیگنال فعال‌ساز وجود نداشته باشد، مدار دیکودر غیرفعال خواهد شد و عمل نخواهد کرد. در دیکودر مذکور اگر n ورودی داشته باشیم 2^n تا خروجی دریافت خواهد شد.



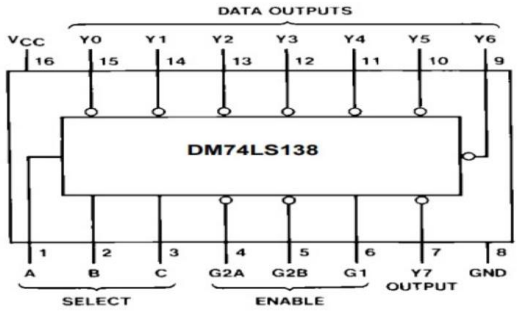
مرحله1) پیاده‌سازی دیکدر 2 به 4 با استفاده از گیت‌های منطقی

1-با توجه به عملکردی که از یک دیکدر 2 به 4 انتظار دارید، ابتدا جدول صحت آن را تکمیل کنید و سپس توابع متناظر با خروجی را بدست آورید. در نهایت با استفاده از IC7400، دیکدر 2 به 4 را پیاده‌سازی کنید.

A	B	Y_3	Y_2	Y_1	Y_0
0	0				
0	1				
1	0				
1	1				

مرحله2) بررسی عملکرد دیکدر 3 به 8 **(IC74138)**

IC74138 دارای 16 پایه است که سه ورودی برای کد و سه ورودی برای فعالسازی آن‌ها و هشت خروجی دارد. پین دیاگرام آن در شکل نشان داده شده است. **توجه شود که** در این آی‌سی،A کم ارزش‌ترین و C باارزش‌ترین بیت است.و همانطور که درنمای شماتیک پین دیاگرام مشاهده می شود، تمامی خروجی ها not دارند، پس برای ارزیابی خروجی‌ها به LED های active low وصل کنیم.



2- با استفاده از IC74238 جدول زیر را تکمیل کنید.

C	B	A	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
0	0	0								
0	0	1								
0	1	0								
0	1	1								
1	0	0								
1	0	1								
1	1	0								
1	1	1								

مرحله3) پیاده‌سازی توابع مختلف با دیکدر **IC74138**

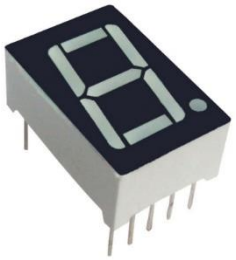
3- برای اجرای توابع $F_1 = \Sigma(1,3), F_2 = \Sigma(3,4,7)$ با استفاده از دیکدر 3 به 8 بگونه ای طراحی را انجام دهید که فقط با کمک گیت 7400 قابل اجرا باشد و بعد از پیاده سازی، صحت مدار بسته شده را چک کرده و جدول را تکمیل کنید.

C	B	A	F_1	F_2
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

10. دیکدر BCD و نمایشگر 7قسمتی(Seven Segment)

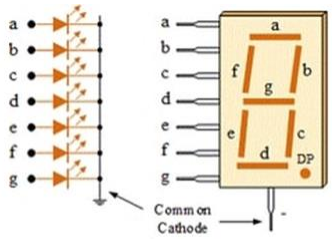
خلاصه آزمایش: در این جلسه با سون‌سگمنت و دیکدر BCD آشنا شده و به کمک آن‌ها اعداد را بر روی نمایشگر، نمایش خواهیم داد.

آشنایی با سون سگمنت (Seven Segment) : از هفت LED تشکیل شده‌است که این هفت LED در کنار هم قرار گرفته اند و یک آی‌سی را می‌سازند. برای نمایش هریک از اعداد 0 تا 9 و کارکترهای هگزا دسیمال A تا F بر روی آن، لازم است تا ترکیب خاصی از این LED ها روشن شوند. از کاربردهای اصلی نمایش‌گر سون سگمنت می‌توان در دستگاه‌های الکترونیکی، ابزارهای اندازه‌گیری، تجهیزات ارتباطی، ساعت‌های دیجیتال، تابلوهای اعلانات و سایر سیستم‌های نمایش اطلاعات مشابه، اشاره کرد.یک Seven Segment استاندارد، 8پایه ورودی دارد که 7تای آن متعلق به هفت LED بوده و هشتمین پایه به‌عنوان پایه مشترک به‌کار می‌رود که مخصوص تمامی سگمنت‌های داخلی نمایشگر است. برخی از نمایشگرها یک پایه‌ی ورودی دیگر نیز دارند که به‌منظور نمایش نقطه‌ی اعشار در گوشه‌ی پایین سمت راست به‌کار می‌رود.



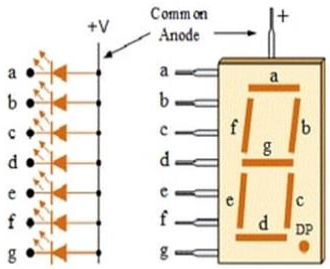
همان‌طور که می‌دانیم، هر LED دو پایه دارد که یکی از آن‌ها آند و دیگری کاتد نامیده می‌شود. بر همین اساس، دو دسته سون سگمنت خواهیم داشت: کاتد مشترک (Common Cathode) و آند مشترک(Common Anode) .

1- **سون سگمنت کاتد مشترک:** همه کاتدها به یکدیگر متصل شده و با هم به "0" یا زمین وصل می‌شوند. بنابراین، هر سگمنت را می‌توان با اعمال سیگنال High یا "1" منطقی به ترمینال آند، روشن کرد.



کاتد مشترک

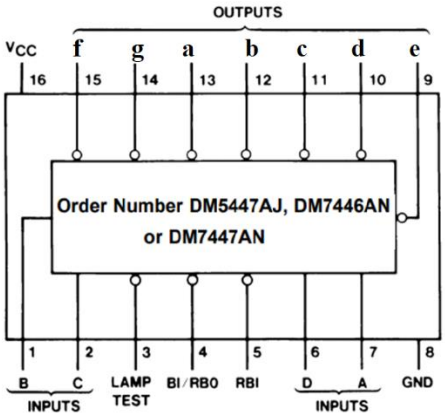
2- **سون سگمنت آند مشترک:** همه آندها به یکدیگر متصل شده و با هم به ولتاژ High یا "1" منطقی وصل می‌شوند. بنابراین برای روشن شدن هر کدام از سگمنت‌ها، تنها لازم است که به پایه‌ی ورودی آن(کاتد)، ولتاژ Low یا "0" منطقی اعمال کنیم.



آند مشترک

برای مثال برای نمایش عدد3، باید سگمنت‌های a,b,g,c,d باید روشن شوند. یا برای نمایش حرفF، باید سگمنت‌هایa,f,g,e باید روشن شوند.

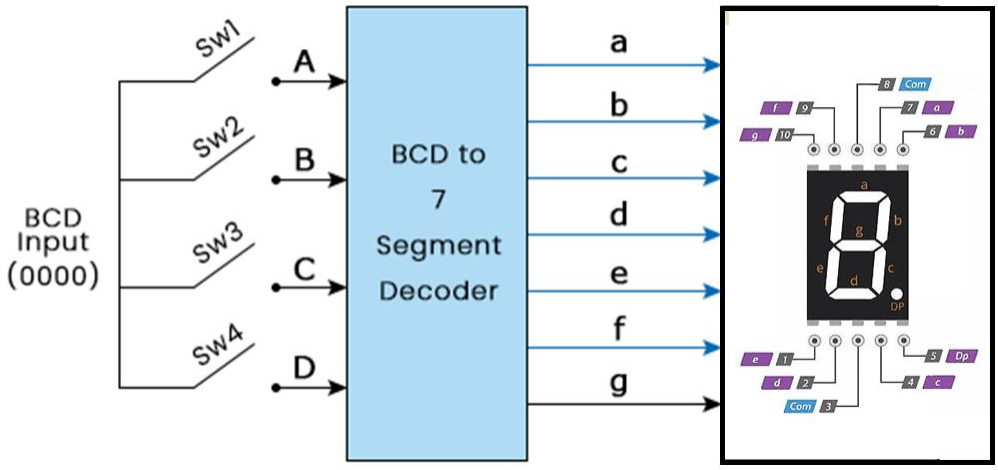
آشنایی با IC7447: یکی از انواع مختلف دیکدرها، دیکدر BCD به دهدهی است. ورودی آن کدها و خروجی آن اعداد در محدوده 0 تا 9 هستند. آی سی 7447 از سری آی سی های TTL شامل یک دیکدر BCD به سون سگمنت بوده که برای راه اندازی سون سگمنت **آند مشترک** به کار می رود. آی سی 7447 دارای 4 ورودی BCD و 8 خروجی مناسب سون سگمنت دارد. از ویژگی های این آی سی میتوان به پایه lamp test و پایه RBI, RBO اشاره کرد. پین دیاگرام آن در شکل نشان داده شده است. توجه شود که D با ارزش‌ترین بیت کد ورودی و A کم ارزش‌ترین بیت کد ورودی است.



مرحله1) مبدل BCD به Seven Segment

با استفاده از IC7447 که یک دیکدر چهار به هشت است، به سون سگمنت ورودی می‌شود و خروجی آن را ثبت می‌گردد.

شرح آزمایش: برای اینکار به‌وسیله چهار کلید، کد متناسب را به ورودی دیکدر اعمال می‌شود و خروجی دهدهی متناسب با کد، از خروجی دیکدر، به ورودی سون سگمنت متصل می‌شود. نحوه‌ی اتصال به‌صورت نمای شماتیک نشان داده شده در شکل زیر است.



1-الف) ورودی‌ها را مطابق جدول زیر به IC7447 اعمال کنید، شکل خروجی نشان داده شده در خروجی را ثبت کنید.

D	C	B	A	Output
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

1-ب) نمایشگر تا چه عددی را درست نمایش داد؟چرا؟

1-ج) فرق IC 7447 و IC 7448 چیست؟ برای طراحی این آزمایش با IC 7448 به چه صورت باید عمل می‌شد و چه تغییری لازم بود؟

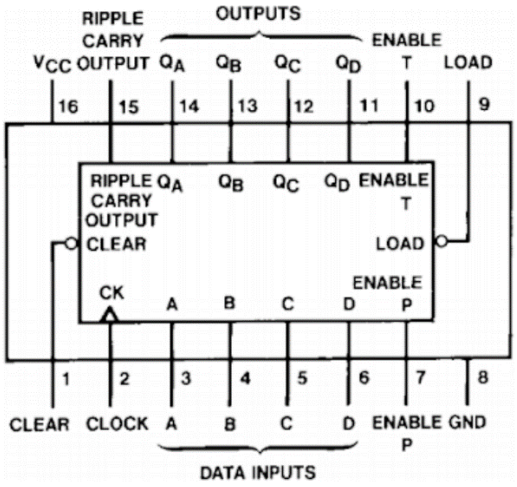
1-د) وظیفه ی پایه‌های شماره 3و4و5 آی‌سی 7447 چیست؟

مرحله2) اتصال شمارنده به Seven Segment

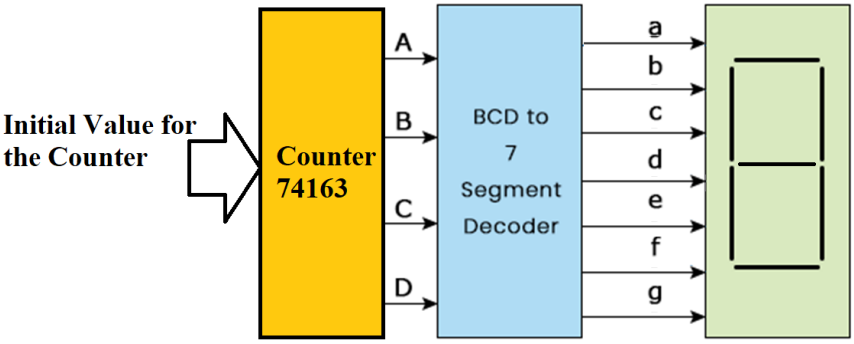
شمارنده‌ها گروهی از مدارات منطقی هستند که یک سری از پالس‌های ورودی را می‌شمارند.

در این مرحله تصمیم است تا با اتصال دیکدر ، سون سگمنت و شمارنده (74163) بتوان شمارش اعداد به‌صورت نزولی و صعودی و یا با شروع از اعداد مختلف را نمایش داد.

IC74163 در واقع یک شمارنده است که دارای 16 پایه امی‌باشد، که چهار ورودی و چهار خروجی دارد. چهار خروجی را می‌توان به ورودی دیکدرBCD متصل کرد و سپس خروجی دیکدر را به Seven Segment متصل کرد تا بتوان شاهد شمارش اعداد دهدهی بود. **توجه شود که** D ورودی باارزش‌ترین بیت و A ورودی کم ارزش‌ترین بیت است.



نحوه‌ی اتصال آن به‌صورت بلوک دیاگرامی در شکل زیر نشان داده شده است.



2-الف) بعد از بستن مدار، نحوه‌ی شمارشی را که بر روی seven segment می‌بینید، شرح دهید.

2-ب) شرح دهید که چگونه می‌توان شمارنده را طراحی کرد تا از 0 تا 9 را به‌صورت بالاشمار بشمارد.

2-ج) شرح دهید که چگونه می‌توان شمارنده را طراحی کرد تا از 9 تا 0 را به‌صورت پایین‌شمار بشمارد.

2-د) شرح دهید که چگونه می‌توان شمارنده را طراحی کرد تا از 3 تا 9 بشمارد.

2-ه) نقش پایه‌ی carry out را تحقیق کنید.

11. لچ و فلیپ فلاپ

خلاصه آزمایش: در این جلسه با مفاهیم لچ و فلیپ فلاپ و مسیر رفع اشکال ساختارها آشنا می‌شویم. با استفاده از گیت‌های پایه لچهای SR, JK, D پیاده‌سازی می‌شود و سپس با کمک لچها، ساختار فلیپ‌فلاپ‌های SR,JK پیاده‌سازی خواهد شد.

مدارهای ترتیبی، اطلاعات دودویی رااز ورودی‌ها دریافت می‌کنند و این ورودی‌ها به‌همراه حالت عناصر حافظه، یک مقدار دودویی را در پایانه‌های خروجی مشخص می‌نماید. مدارهای ترتیبی از نظر مسائل زمانی سیگنال‌هایشان به دو نوع اساسی سنکرون و آسنکرون طبقه‌بندی می‌شوند. در سیستم سنکرون کلاک همگی عناصر حافظه همگام با هم هستند. درحالی‌که در آسنکرون کلاک‌ها غیرهمگام هستند.

عناصر پایه‌ی حافظه لچها هستند. که اگر ورودی enable باشد، داده‌ی ورودی روی خروجی می‌تواند تاثیر بگذارد، پس در کل بازه‌ی فعال بودن، هر تغییری در ورودی بر روی خروجی تغییر ایجاد می‌کند پس لچها حساس به سطح (سطح + یا سطح -) هستند. درحالی‌که در مدارات دیجیتال نیاز داریم تنها در بازه‌ی خاصی اطلاعات را ذخیره کنیم نه در یک بازه‌ی طولانی. علاوه بر این، در لچها مشکل همگام سازی هم داریم، یعنی به‌راحتی نمی‌توانیم همه حافظه‌ها را با هم سنکرون کنیم.

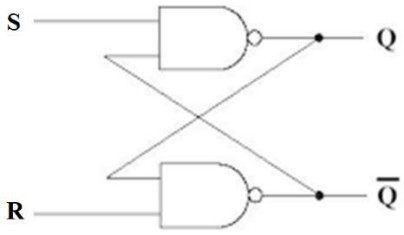
می‌توان از فلیپ‌فلاپ‌ها به جای لچ استفاده کرد که فقط در زمانهای خاصی، خروجی‌ها به‌روز می‌شوند. درواقع FF ها از لچها ساخته می‌شوند و حافظه‌های حساس به لبه (لبه پایین‌رونده یا لبه بالارونده) هستند. یک FF قادر است تا وقتی‌که تحت تاثیر یک سیگنال ورودی برای تغییر حالت قرار نگرفته، یک حالت دودویی را به‌طور نامحدود در خود نگهداری کند. مدار یک FF دارای دو خروجی است(یکی برای مقدار طبیعی بیت ذخیره شده و دیگری برای مکمل آن).

- مرسوم‌ترین فلیپ‌فلاپ‌ها عبارتند از: D, T, JK, SR

- FF ها وردی‌هایی برای clear یا reset دارند. این ورودی‌ها توسط یک سیگنال مثبت یا منفی روی FF اثر می‌کنند و برای این‌کار نیاز به پالس ساعت ندارند.

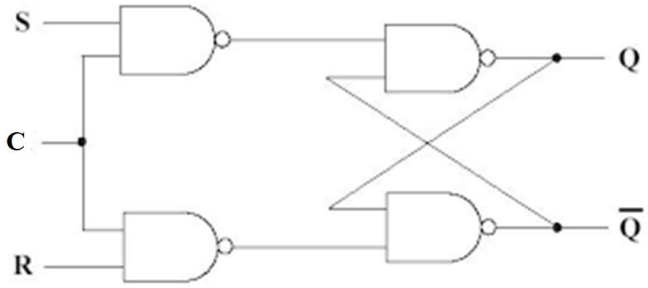
مرحله**1**) طراحی و پیاده‌سازی **SR- Latch**

1-الف) مدار SR-latch زیر را مطابق شکل زیر ببندید. با اعمال ورودی‌ها مطابق جدول، خروجی‌ها را ثبت کنید. مشکل مدار حاضر چیست؟ طرز کار مدار را شرح دهید.



S	R	Q	$\overline{Q}$
0	1		
0	0		
1	0		
0	0		
1	1		
0	0		
1	0		
1	1		
1	0		

1-ب) اگر به طرح قبلی، یک ورودی کنترلی به‌صورت زیر اضافه کنیم چه اتفاقی می افتد؟ نقش ورودی C چیست؟



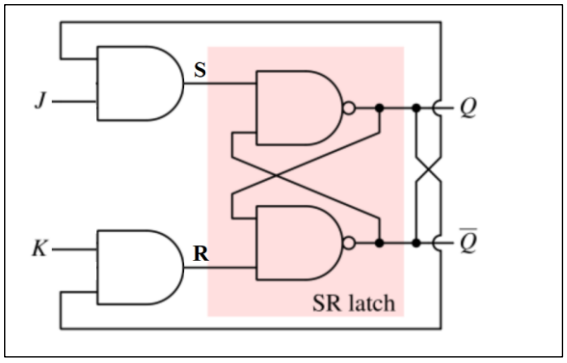
S	R	C	Q	$\overline{Q}$
0	1	1		
0	0	1		
1	0	1		
0	0	1		
1	1	1		
0	0	1		
0	1	0		
0	0	0		
1	0	0		
1	0	1		

مرحله2) طراحی و پیاده‌سازی D- Latch و JK-Latch

2-الف) در مدار قبلی (SR-latch کنترلی) ، ورودی S را با یک Not به ورودی R وصل کنید. بعد از بستن مدار، جدول را تکمیل کنید و عملکرد آن را شرح دهید. مزیت این ساختار نسبت به SR-latch کنترلی چیست؟

D	En	Q
1	0	
0	0	
0	1	
1	1	

2-ب) مطابق مدار شکل زیر، یک JK-Latch را با SR-Latch ساده، پیاده‌سازی کنید.

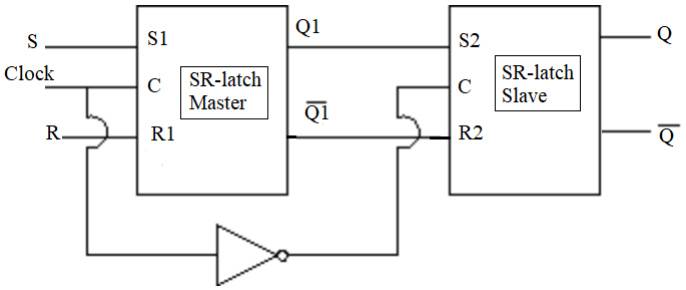


J	K	Q	$\overline{Q}$
0	1		
0	0		
1	0		
0	0		
1	1		
0	0		
0	1		
1	1		
1	0		
1	1		

2-ج) مشکل مدار فوق چیست؟

مرحله3) طراحی و پیاده‌سازی فلیپ فلاپ‌ها

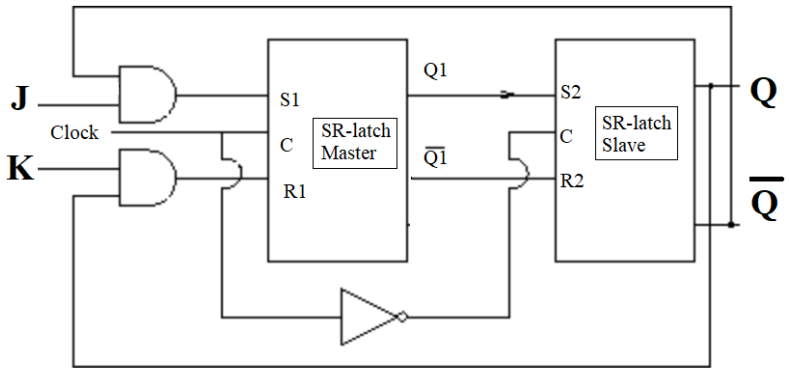
3-الف) با استفاده از SR-Latch با آرایش master-slave، فلیپ فلاپ SR را پیاده‌سازی کرده و جدول زیر را تکمیل کنید.



S	R	Q	$\overline{Q}$
0	1		
0	0		
1	0		
0	0		
1	1		
0	0		
0	1		

3-ب) در شکل‌بندی فوق فلیپ فلاپ به کدام لبه حساس است؟

3-ج) با استفاده از SR-Latch با آرایش master-slave، فلیپ فلاپ JK را پیاده‌سازی کرده و جدول زیر را تکمیل کنید. آیا مشکل موجود در JK-latch همچنان وجود دارد؟



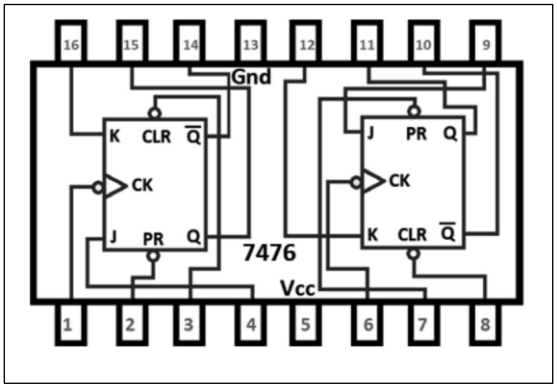
J	K	Q	$\overline{Q}$
0	1		
0	0		
1	0		
0	0		
1	1		
0	0		
0	1		
1	1		
1	0		
1	1		

12. فلیپ فلاپ T و شیفت رجیستر

خلاصه آزمایش: با IC7476 و IC74194 آشنا شده و عملکرد این دو آی‌سی بررسی می‌شود.

مرحله 1) ساخت **T-FF** با استفاده از **JK-FF**

فلیپ فلاپ JK یکی از پرکاربردترین و مهم‌ترین فلیپ‌فلاپ‌ها در طراحی مدارات منطقی است. این نوع فلیپ‌فلاپ دارای دو ورودی است که به طور سنتی با نمادهای J و K برچسب‌گذاری می‌شوند. IC7476 شامل دو JK-FF با پایه‌ی پاک‌کننده است و شکل پین دیاگرام آن در شکل زیر نشان داده شده است.



1-الف) عملکرد فلیپ‌فلاپ JK را بررسی کنید و جدول را تکمیل کنید.

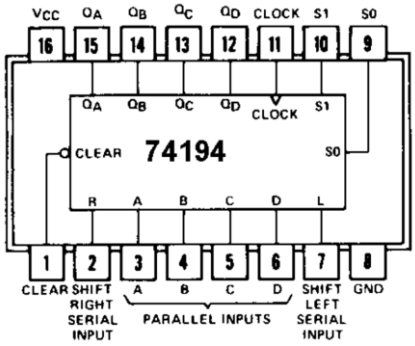
J	K	Q	$\bar{Q}$
0	1		
0	0		
1	0		
0	0		
1	1		
0	0		
0	1		
1	1		
1	0		
1	1		

1-ب) با استفاده از IC7476، T-FF طراحی کنید. ابتدا فلیپ فلاپ را reset کنید و سپس جدول را پر کنید.

1-ج) T-FF دارای چند حالت است؟

مرحله2) بررسی عملکرد شیفت رجیستر

IC74194 یک شیفت رجیستر چهاربیتی است که دارای 16 پایه است و در دو جهت قابلیت شیفت ایجاد می‌کند. پین دیاگرام آن در شکل زیر نشان داده شده است.



2-الف) در مورد نقش پایه‌های 2 و 7 تحقیق کنید. در آزمایشات بعدی این پایه‌ها را به یک وصل کنید.

2-ب) اگر S1S0= 11 باشد. چه عملکردی صورت می‌پذیرد؟ (برای بررسی عملکرد در ورودی 0100 اعمال شود و خروجی را ثبت کنید)

Input	Output
0100	

2-ج) اگر S1S0= 01 باشد. به ترتیب مقدار خروجی را که می‌بینید، ثبت کنید. چه عملکردی صورت می‌پذیرد؟

Input	Output
0100	

2-د) اگر S1S0= 10 باشد. به ترتیب مقدار خروجی را که می‌بینید، ثبت کنید. چه عملکردی صورت می‌پذیرد؟

Input	Output
0100	