

## آزمایش شماره ۳ مدار منطقی - آشنایی با مدارات Demux-Mux-Decoder-Encoder

استاد: دکتر لالی

مصطفی خوشنود و حسین برقی

چکیده :

در این آزمایش می‌خواهیم با دیکدر ، انکدر و مالتی پلکسر آشنا شویم و پس از این آشنایی اولیه به کمک تراشه ۷۴۱۵۳ جمع کننده ای که در آزمایش قبل ساختیم را دوباره بسازیم و این پیاده سازی جدید را با پیاده سازی دستور کار قبلی مقایسه کنیم و تفاوت های آن ها را درک کنیم.

قصد داریم پیاده سازی دیکدر ، انکدر و مالتی پلکسر را فرا بگیریم و در این راه از تراشه ۷۴۰۰ و گیت nand استفاده میکنیم ؛ برای این منظور هم در ابتدا جدول های درستی را رسم می کنیم و سپس به کمک روابط جبر بولی به روابط بر مبنای nand می رسمیم و سپس بر مبنای این روابط طراحی خود را انجام می دهیم.

نهایتا پس از طراحی ها به کمک nand ؛ به لطف همین گیت و استفاده از decoder یک تمام جمع کننده می سازیم سپس تمام جمع کننده ای با مالتی پلکسر می سازیم که برای رسیدن به این منظور پیاده سازی یک مالتی پلکسر 4x1 را به کمک سه مالتی پلکسر 2x1 را هم انجام می دهیم.

مقدمه:

آنچه پس از انجام آزمایش گزارش کار قبلی برای ما مشهود بود پیچیده بودن مدار پیاده سازی شده و همچنین سخت بودن عیب یابی این مدار بود .

با پیش زمینه آزمایش قبلی بر آن شدیم تا روند ساده تر و قابل فهم تری برای پیاده سازی یک جمع کننده را طی کنیم .

برای این منظور با تغییر نگرش به جدول درستی یافتیم که میتوانیم از مالتی پلکسر برای این پیاده سازی استفاده کنیم ؛ اما پیش از پرداختن به این موضوع بحث دیکدر و انکدر را دوره کردیم و پیاده سازی آن ها را انجام دادیم.

برای رسیدن به منظور بالا نیاز بود تا علاوه بر این که دستور کار های آزمایش های قبل را مرور کنیم ؛ مفاهیم مالتی پلکسر ؛ دیکدر و انکدر را نیز مرور کنیم که لینک زیر برای این مهم مفید واقع شد:

<https://www.engineersgarage.com/more-combinational-circuits-multiplexers-demultiplexers-encoders-and-decoders-de-part-14/>

روش ها و تجهیزات مورد استفاده:

در این آزمایش یک bread board استاندارد و سالم به علاوه تراشه 74153 را لازم داشتیم .

همچنین به led ، منبع قدرت و مقاومت برای پیاده سازی مدار نیاز داشتیم!

برای بحث هایی که در آزمایشگاه پیاده سازی نشد و به صورت طراحی و تئوری انجام شد علاوه بر دیتاشیت تراشه 74153 به سایت های زیر نیاز پیدا کردیم و از آن ها استفاده کردیم:

<https://circuitverse.org/>

<https://logic.ly/demo/>

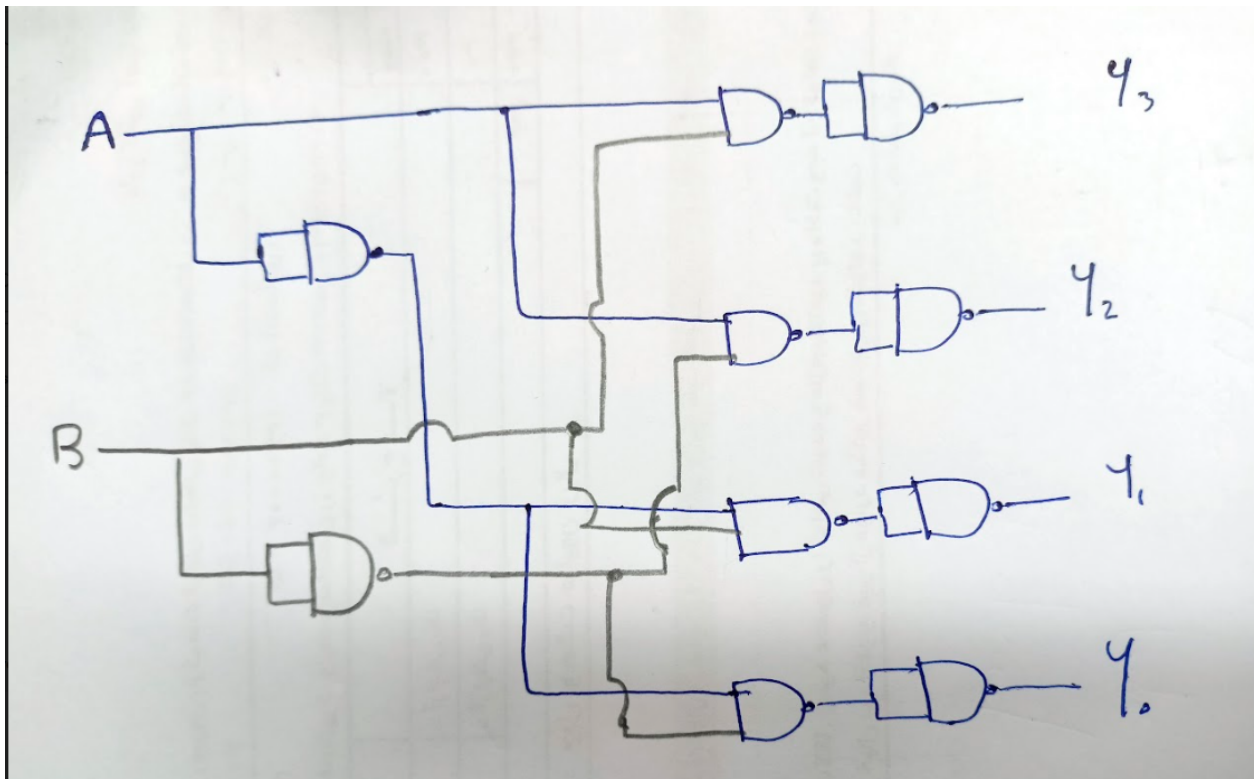
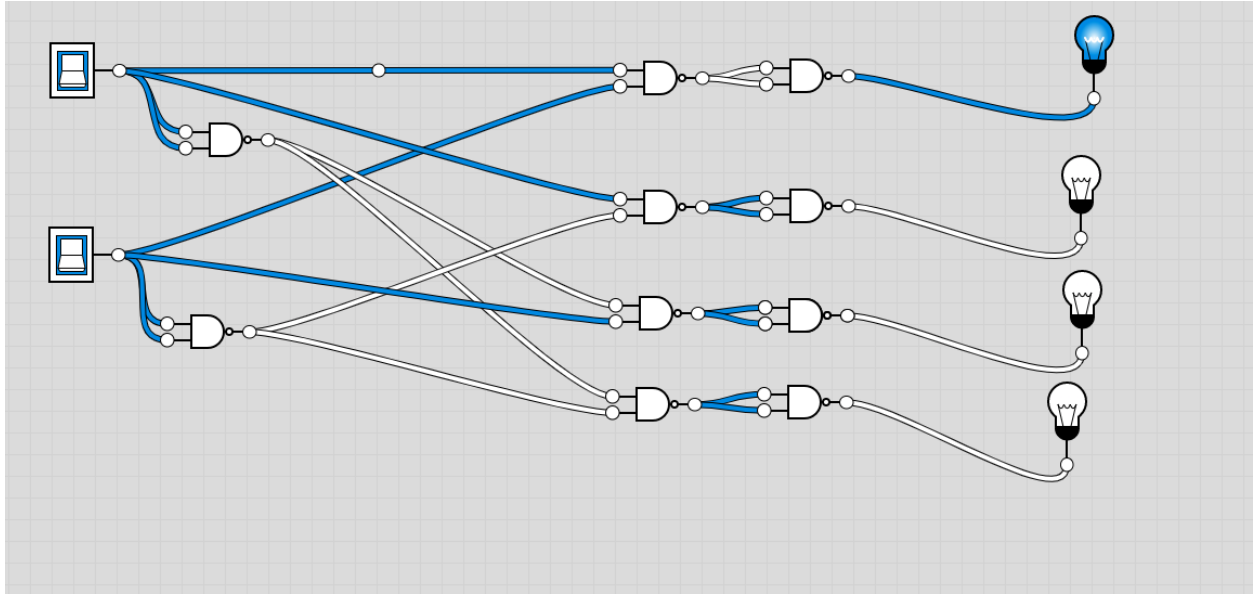
شرح آزمایش و نتایج :

آزمایش شماره ۱:

در decoder ها یک پایه E هم وجود دارد که enable است و تعیین می کنید که decoder ما فعال باشد یا خیر با توجه به active high یا active low بودن decoder ما با ست کردن این پایه میتوانیم تمامی خروجی هامون رو صفر کنیم یا مقادیر اصلی رو برگردونیم.

طراحی حاصل شده برای پیاده سازی decoder با گیت nand .

A	B	Y0	Y1	Y2	Y3
0	1	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	0	0	0	0	1

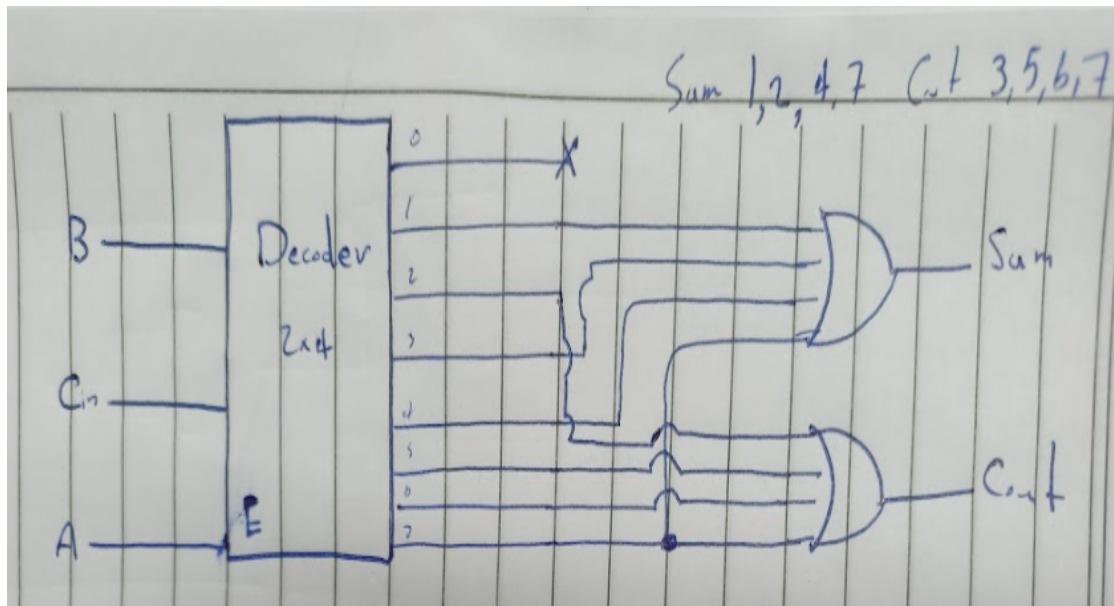


پیاده سازی تمام جمع کننده با استفاده از decoder .

برای این کار یک از ورودی ها را به عنوان enable در نظر میگیریم و ورودی دیگر به علاوه Cin را به عنوان ورودی

های decoder دو به چهار

A	B	Cin	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



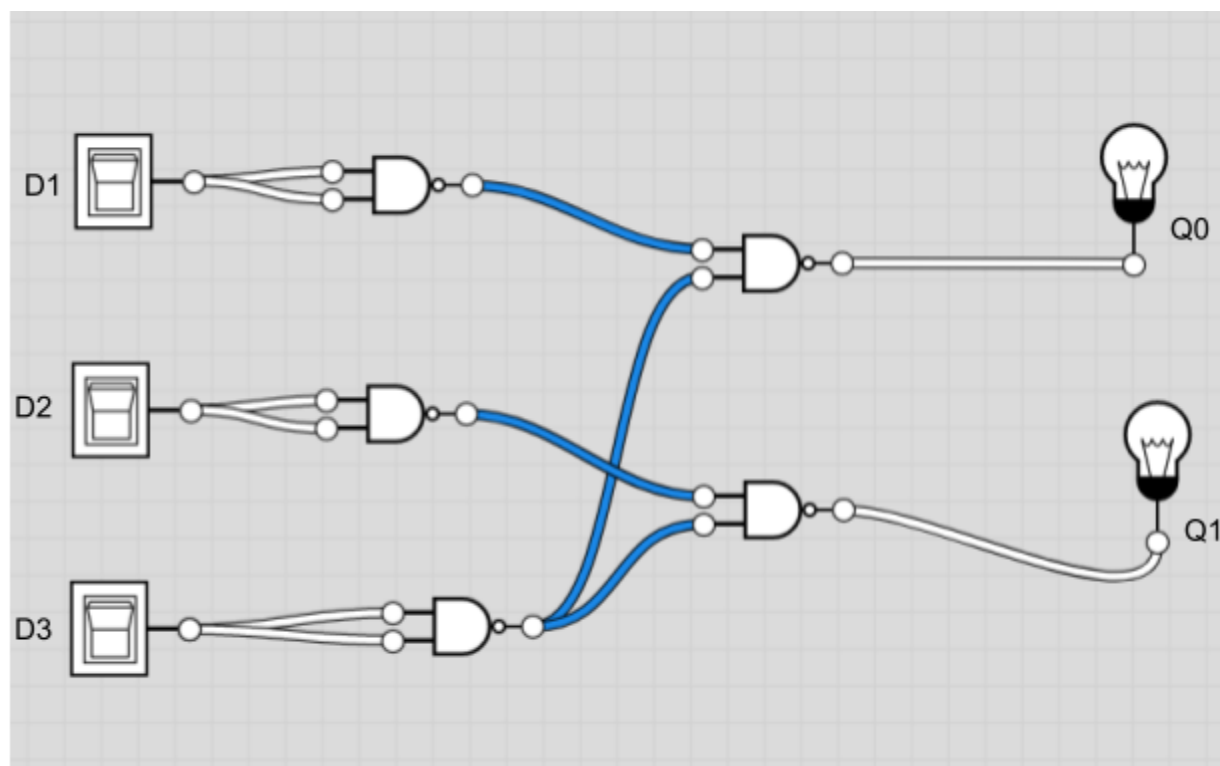
زمایش شماره ۲ :

پیاده سازی یک 4x2 encoder با مدار nand :

جدول درستی :

D3	D2	D1	D0	Q0	Q1
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

مدار طراحی شده:



آزمایش شماره ۳:

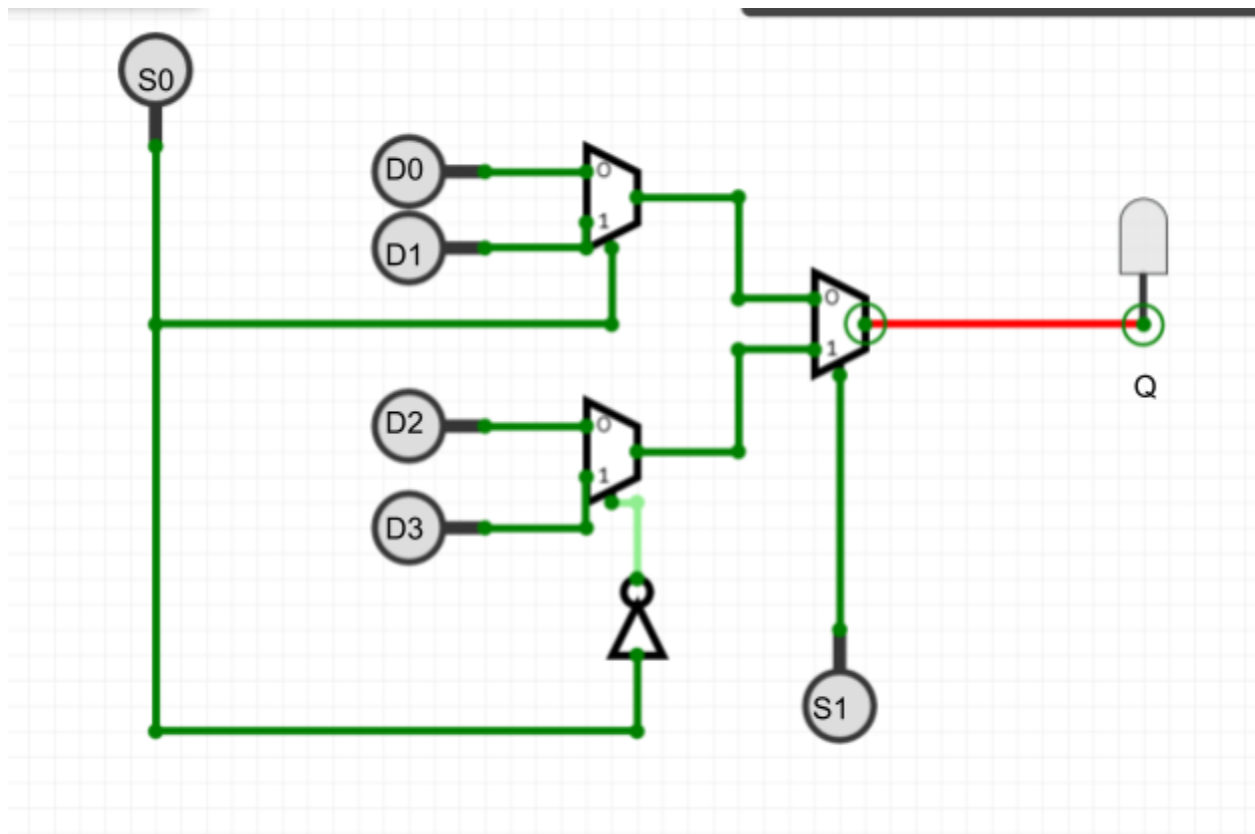
پیاده سازی یک مالتی پلکسر با دو سلکتور:

جدول درستی:

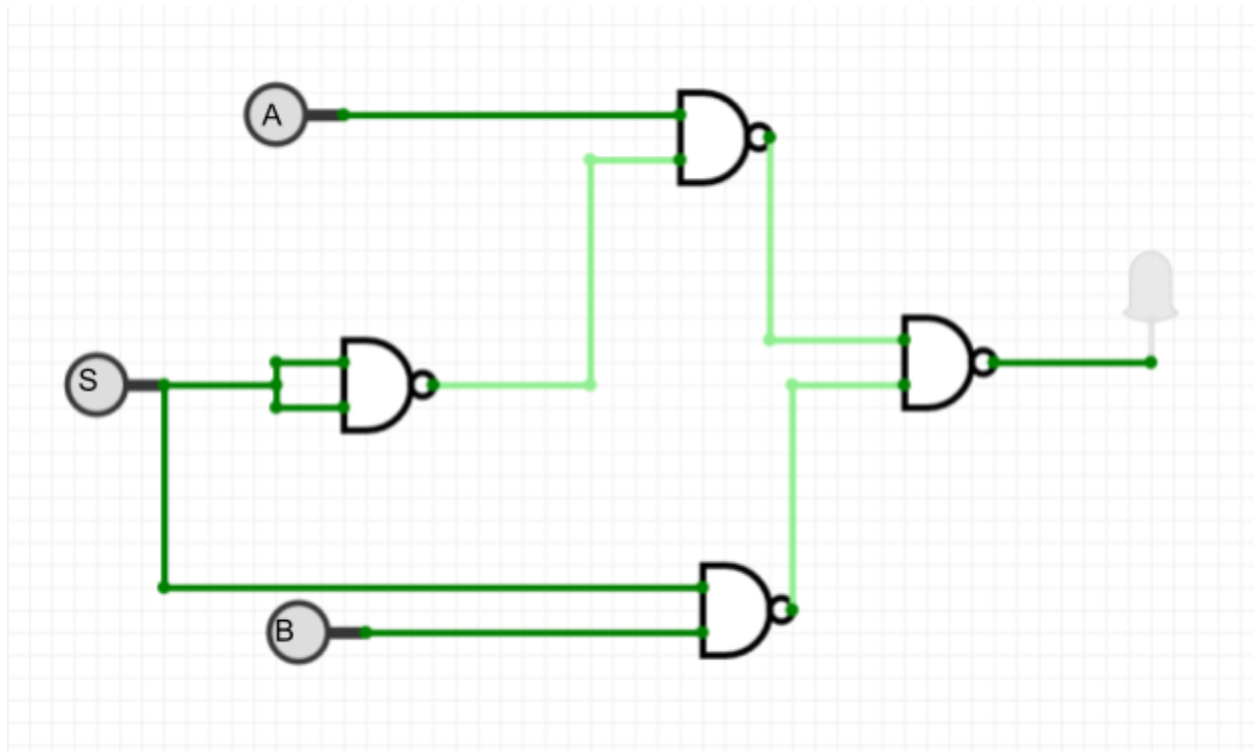
S1	S0	Out
0	0	D0
0	1	D1
1	0	D2
1	1	D3



ابتدا برای طراحی مالتی پلکسر خود ؛ مالتی پلکسر خود را به کمک مالتی پلکسر های دو به یک طراحی می کنیم:

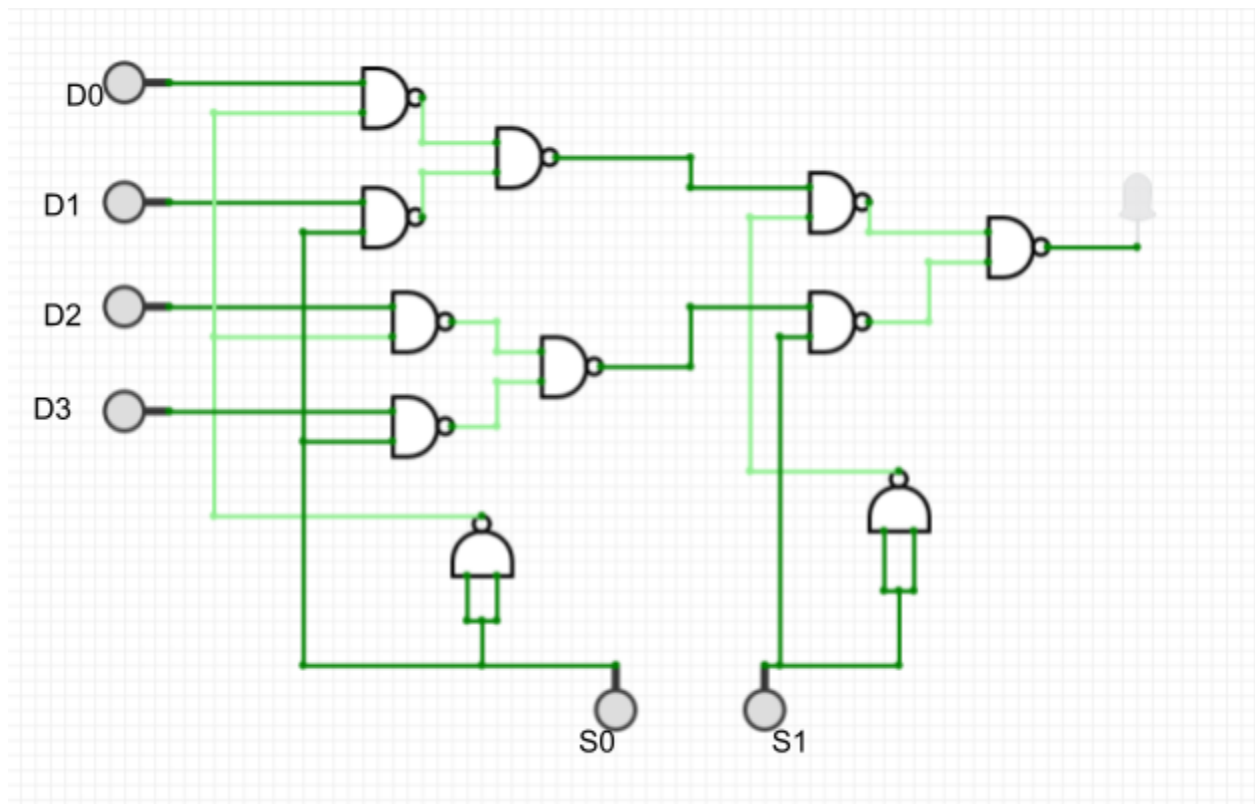


سپس مالتی پلکسر دو به یک خود را به کمک گیت nand پیاده سازی می کنیم:



حالا که توانستیم مالتی پلکسر خود را به کمک nand طراحی کنیم کافیهست که اتصال بین این مالتی پلکسر ها را هم به کمک nand پیاده سازی کنیم. (صفحه ی بعد)

به کمک طرح بالا و پیاده سازی که قبلا از مالتی پلکسر داشتیم مدار زیر را طراحی می کنیم:



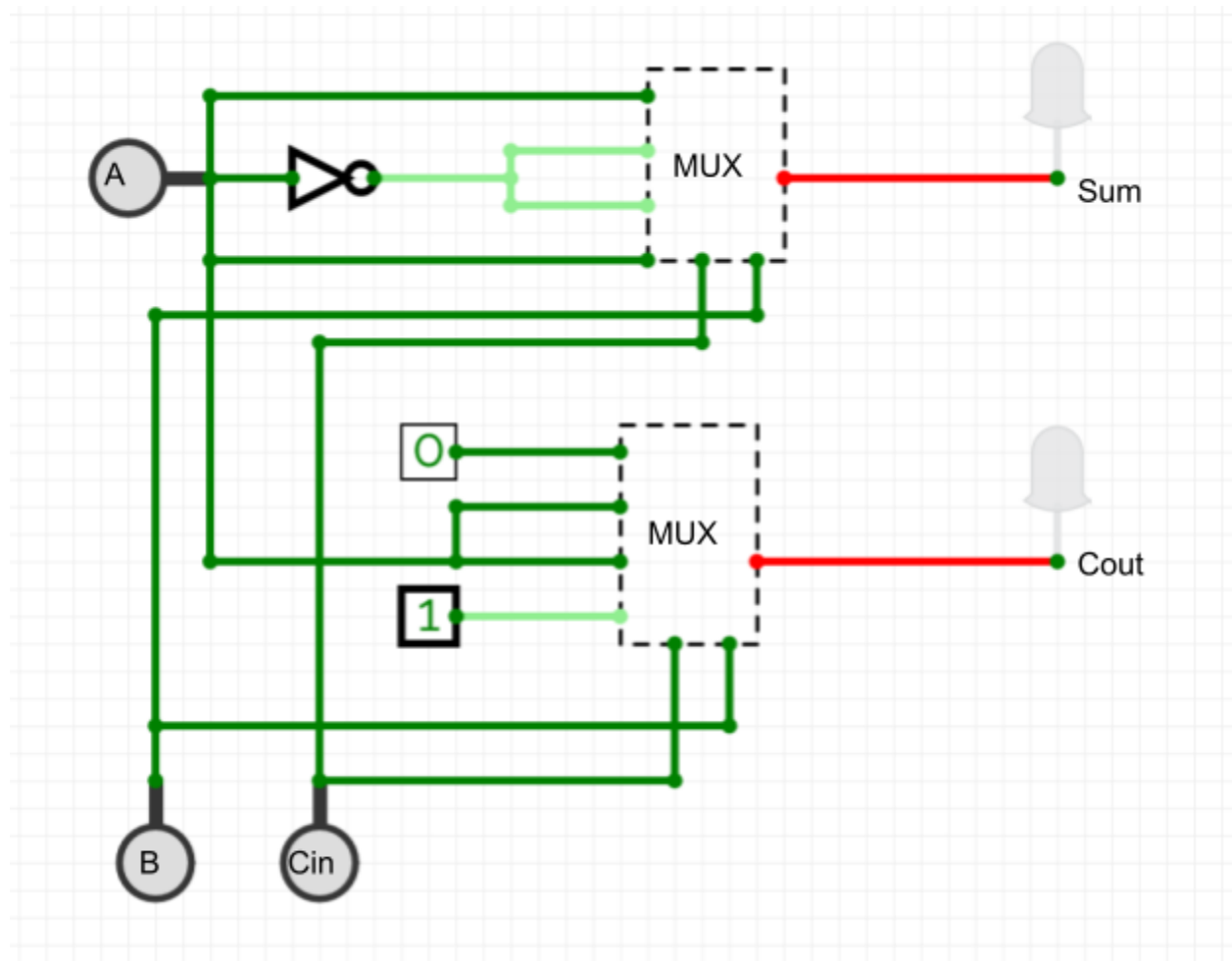
آزمایش شماره ۴:

پیاده سازی تمام جمع کننده با استفاده از مالتی پلکسر:

جدول درستی:

Cin	B	A	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

اگر از بالای جدول دو تا دو تا ردیف ها را جدا کنیم متوجه می شویم که مقادیر cout و sum با مقدار A رابطه دارند و می توانیم از B و Cin به عنوان selector های مالتی پلکسر خود استفاده کنیم که نهایتا مدار صفحه ی بعد حاصل می شود!



بخش پنجم: مدار پیاده سازی شده به کمک دیکدر مداری قابل فهم تر بر روی کاغذ بوده و به کمک یک دیکدر و چند گیت 02 پیاده سازی میشود ؛ در مقابل مدار پیاده سازی شده به کمک مالتی پلکسر به راحتی با یک تراشه و به کمک not کردن دستی (همانطور که در آزمایشگاه پیاده سازی کردیم) قابل پیاده سازی است .

همچنین به نظر می رسد با این که در طراحی پیاده سازی با دیکدر راحت تر باشد اما با توجه به نوع تراشه ها پیاده سازی با استفاده از مالتی پلکسر راحت تر باشد.

منابع:

<https://www.engineersgarage.com/more-combinational-circuits-multiplexers-demultiplexers-encoders-and-decoders-de-part-14/>

<https://www.quora.com/p/17069/design-a-full-adder-using-38-decoder-2/>

برای مطالعه بیشتر:

پیاده سازی مالتی پلکسر به کمک دیکدر

<https://www.quora.com/How-can-I-implement-4-to-1-mux-using-a-decoder>