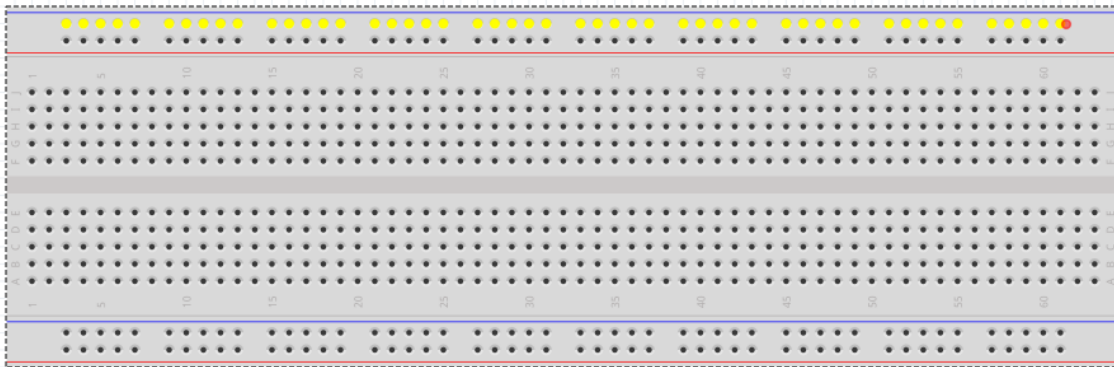


آزمایش اول:

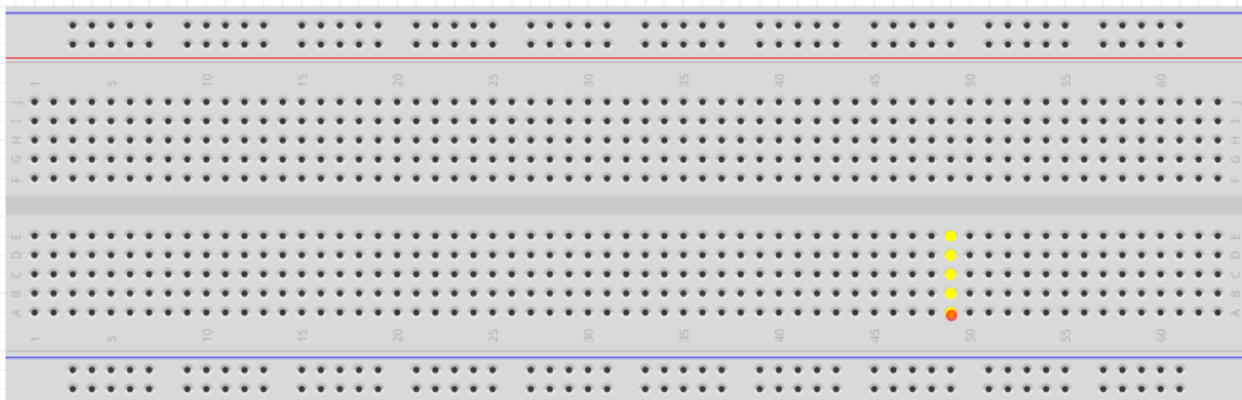
هدف آزمایش:
آشنایی با محیط های نرم افزاری شبیهسازی مدارهای منطقی.

رسم مدار با Fritzing :

همانطور که در شکل زیر مشاهده خواهید کرد، برد مورد به یک قسمت بالایی دارد و یک قسمت پایینی که در این بخش ها اتصال نقاط به هم دیگر به صورت افقی است.

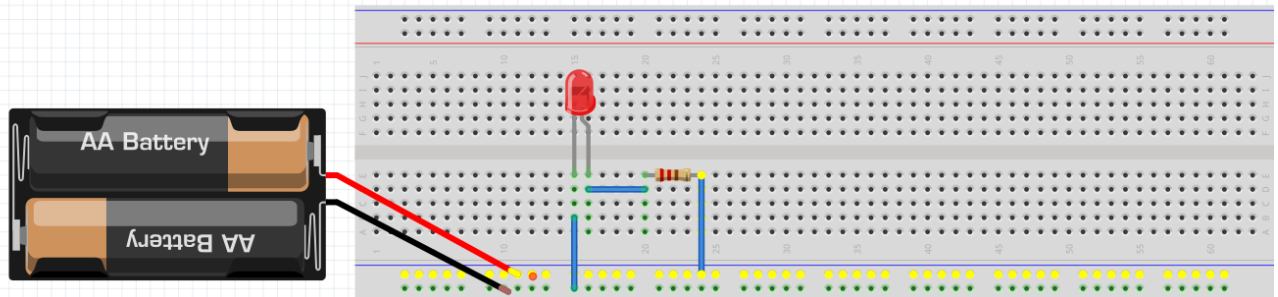


در دو قسمت میانی برد مورد، اتصالات به صورت عمودی هستند، هر 5 نقطه در یک ستون با هم در یک اتصال قرار دارند.

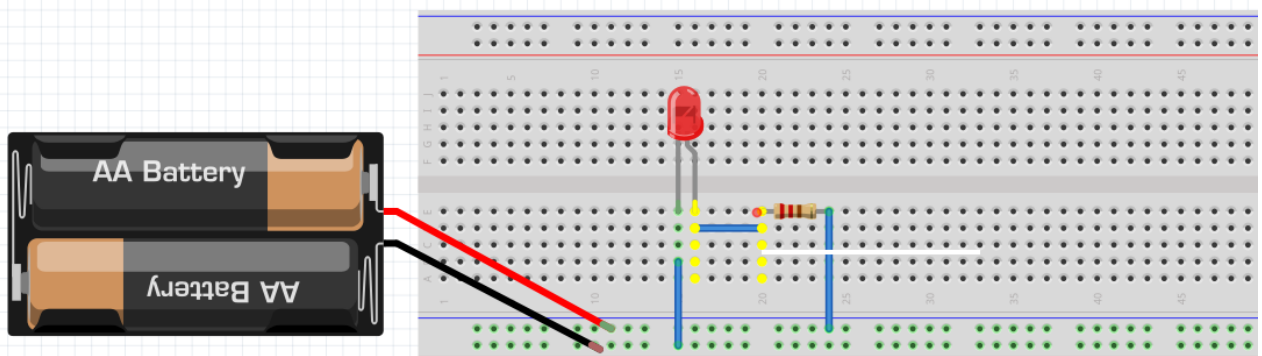


در بخش دوم این آزمایش، یک مقاومت و LED و باتری را به هم می‌پندیم.

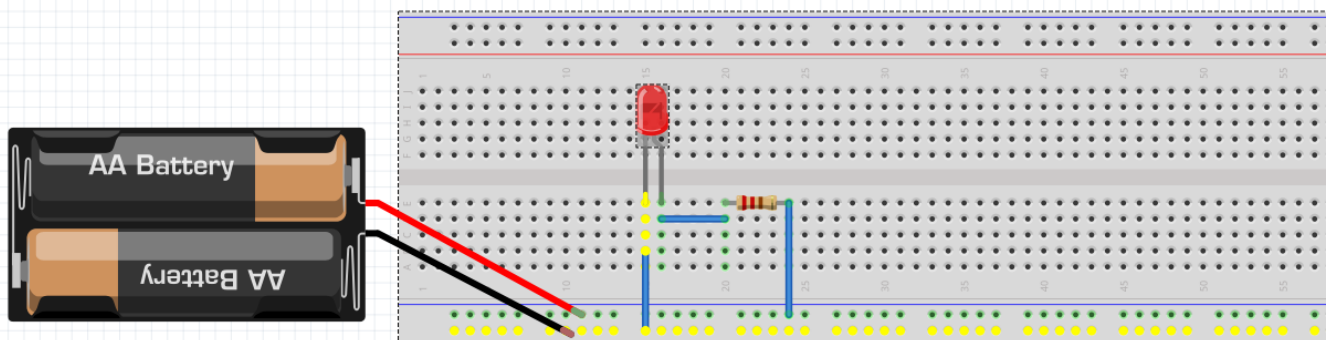
اتصال قطب مثبت باتری به مقاومت.



اتصال مقاومت به قطب مثبت LED .

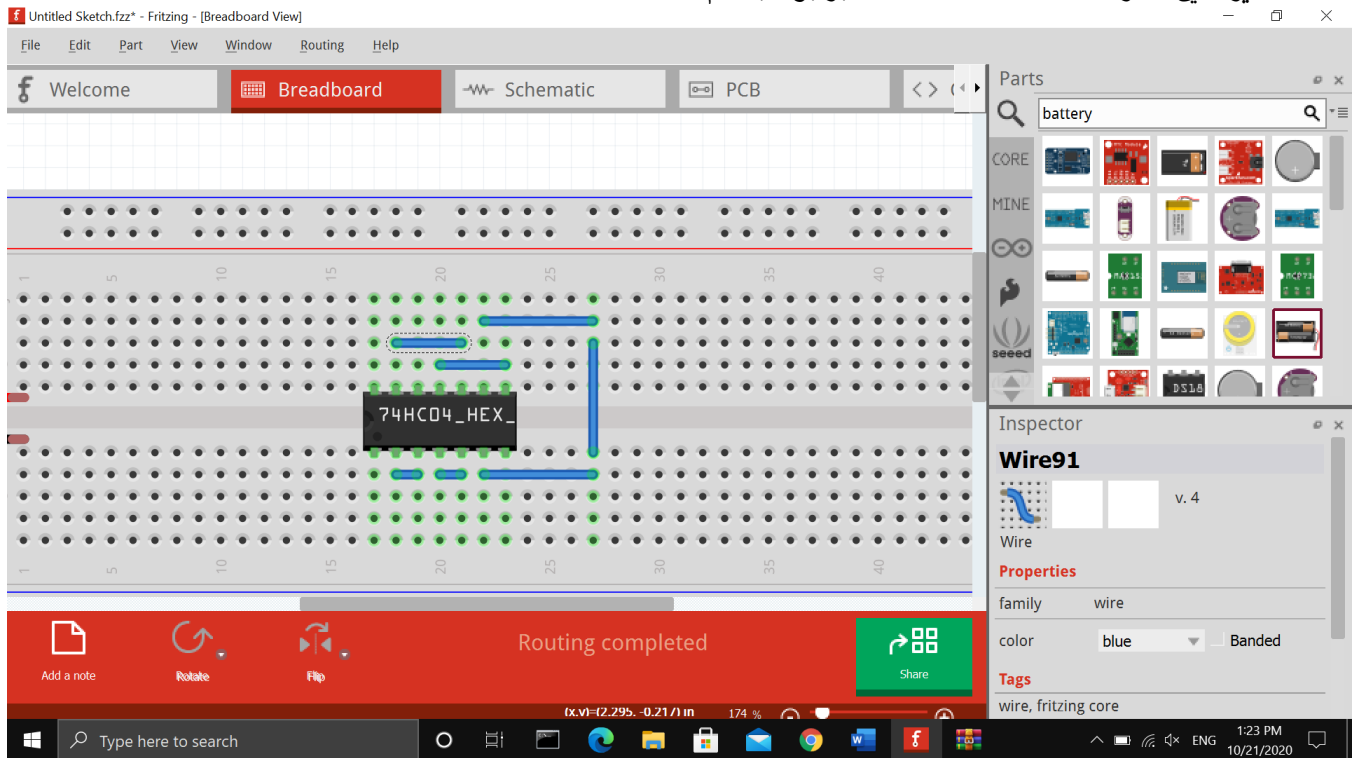


اتصال قطب منفی LED به قطب منفی باتری

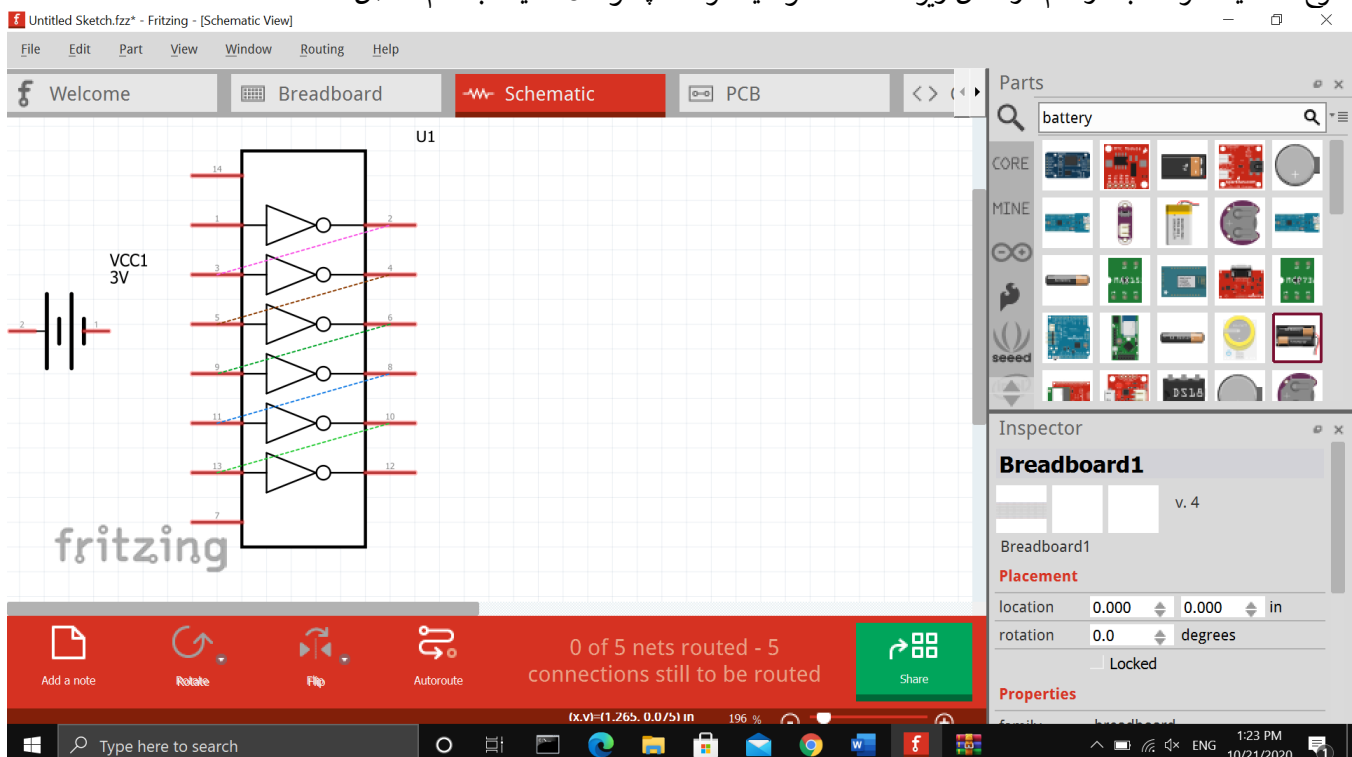


در بخش سوم این آزمایش یک مقدار با 6 بیت NOT رسم خواهیم کرد.

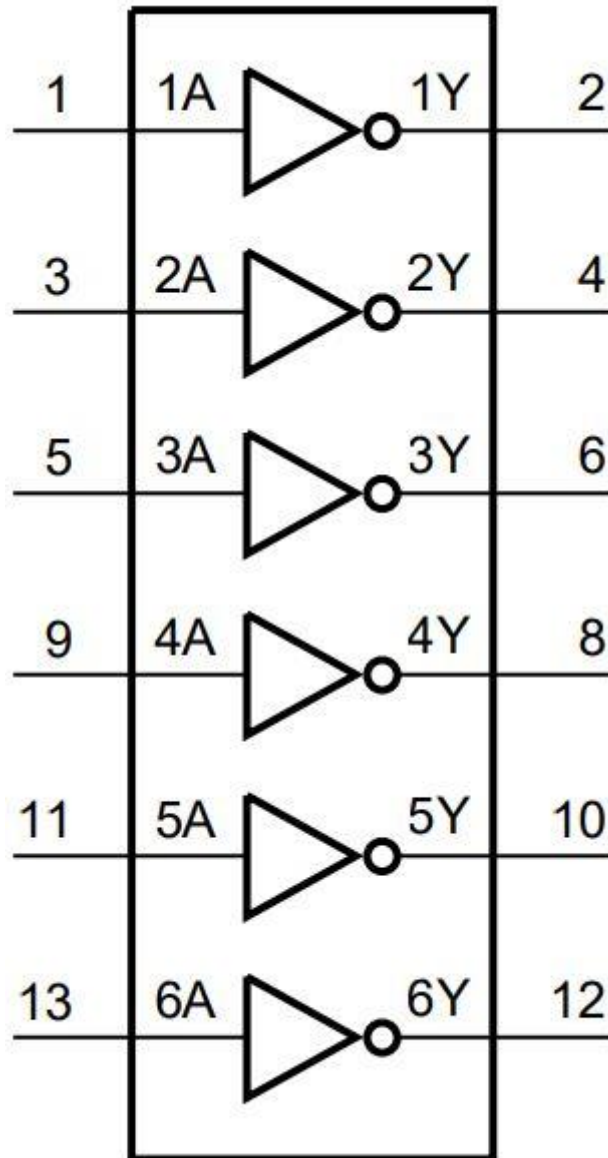
در شکل زیر نمایی از تراشه که 6 تا بیت NOT در بردبرد به هم متصل شده اند را ملاحظه میکنید.



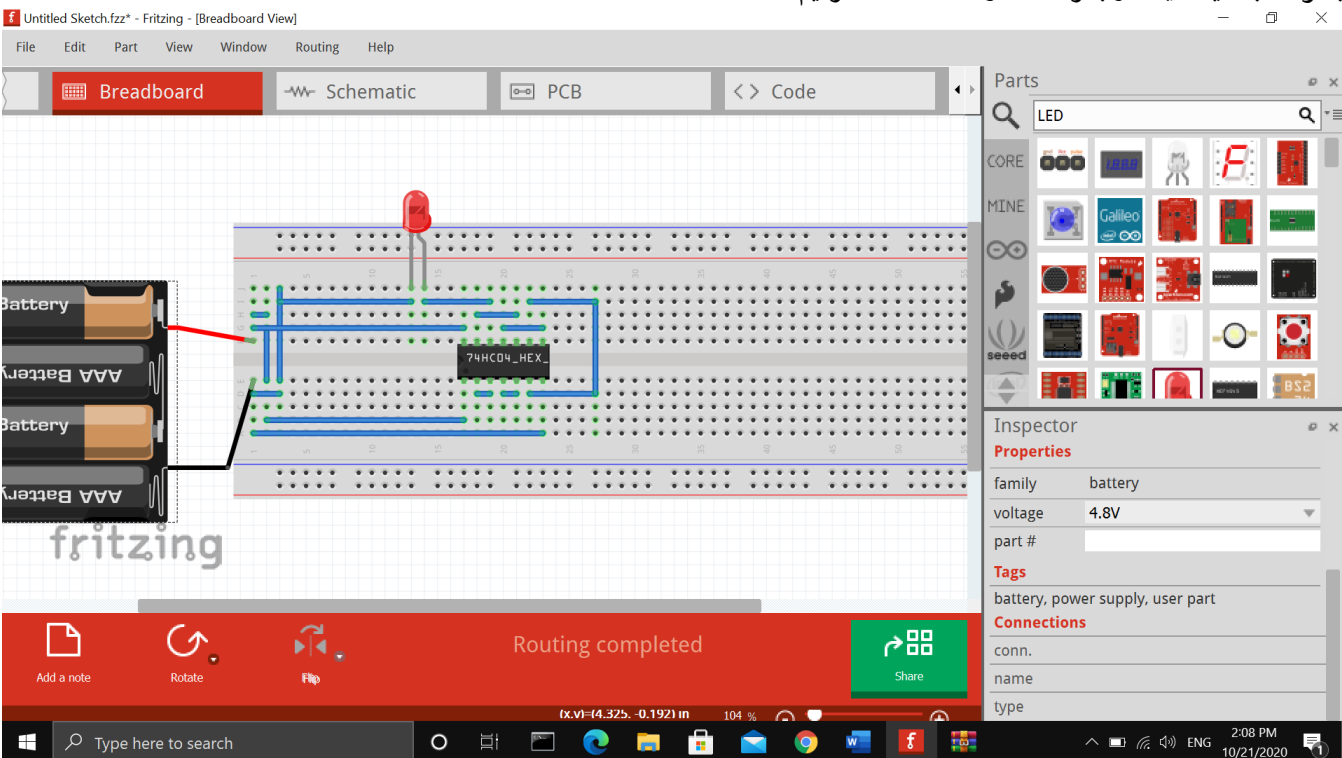
طرح شماتیک تراشه بالا را هم در شکل زیر مشاهده خواهید کرد که چگونه 6 تا بیت به هم متصل اند.



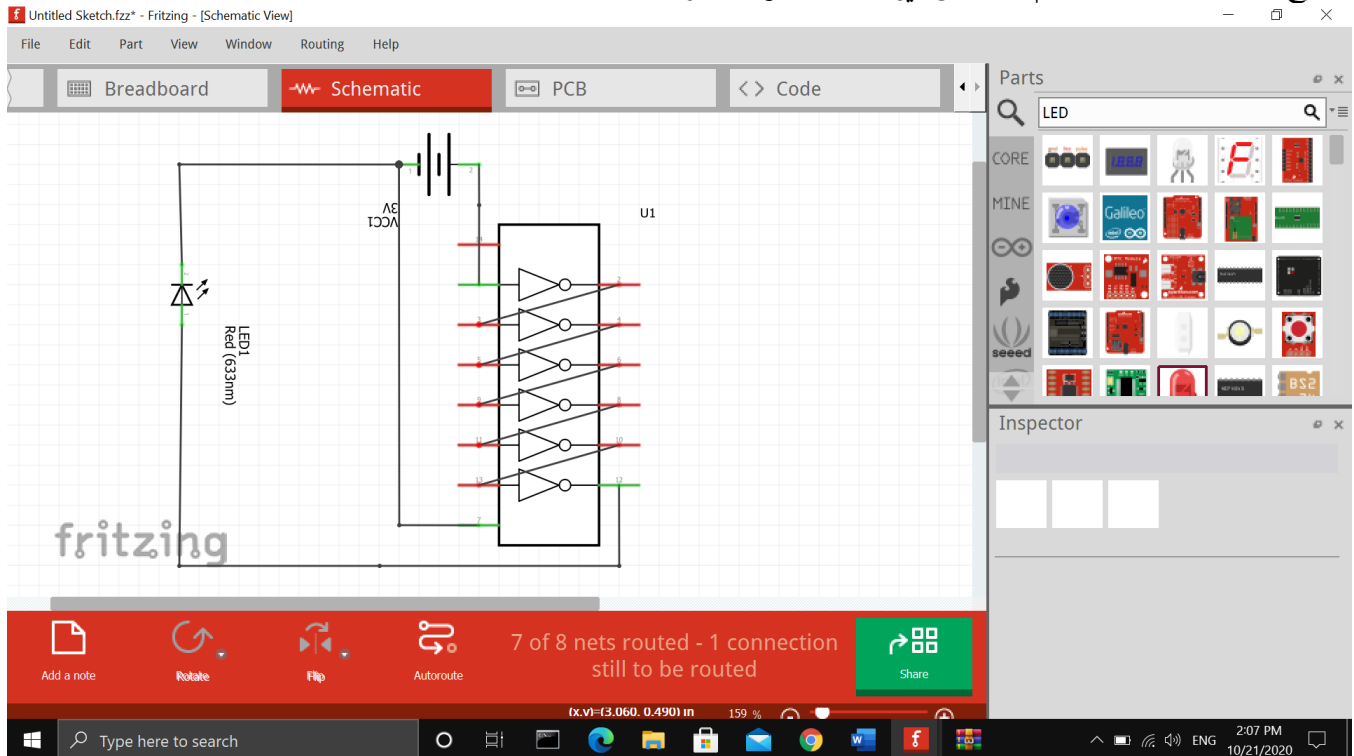
نمایی از طرح شماتیک تراشه:



در شکل زیر ما تراشه را با یک باتری و یک LED به هم متصل کردیم، نحوه اتصال پایه ها در فیلم توضیح داده شده است. با توجه به دیتاشیت از باتری 4.8 ولت استفاده کردیم.



و طرح شماتیک شکل بالا را هم در عکس زیر مشاهده خواهید کرد.



آزمایش دوم:

هدف آزمایش:

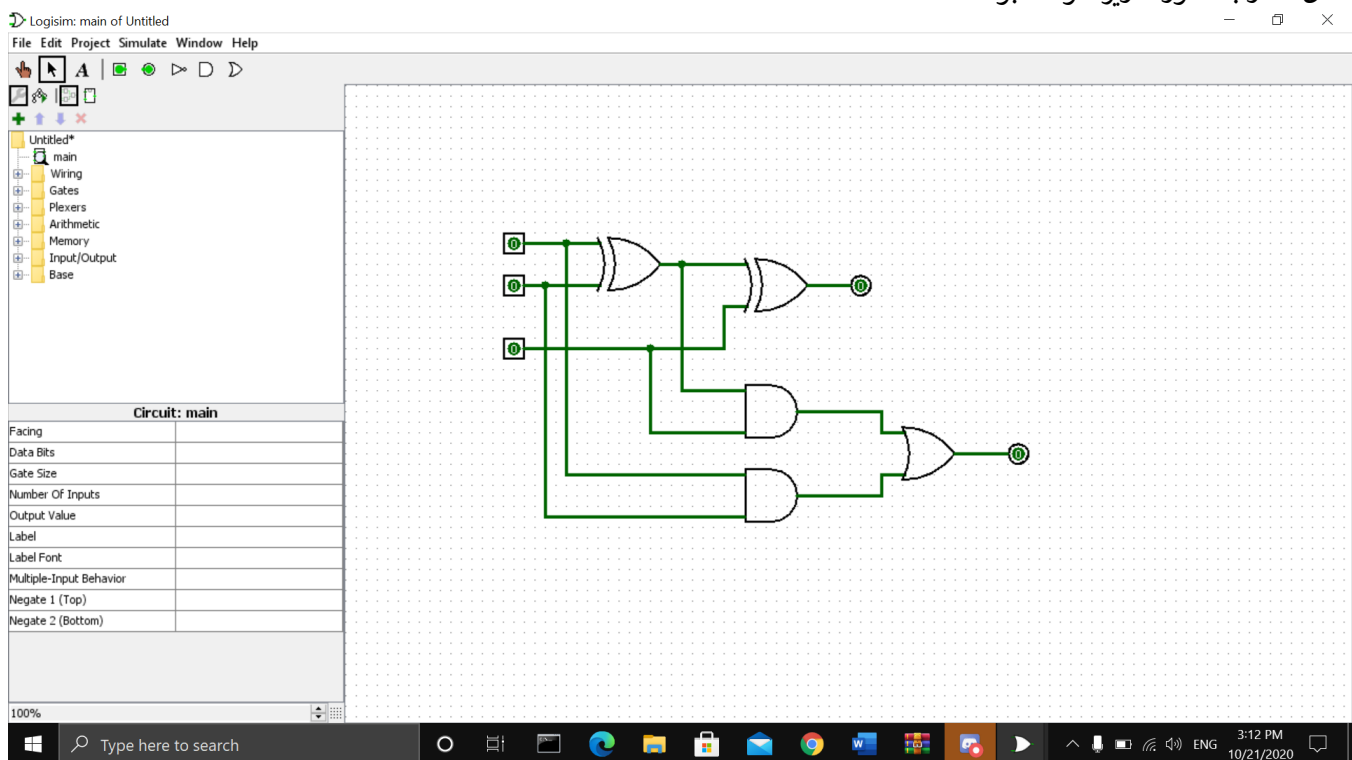
ساخت مدار با Logisim :

در مدار Full adder ورودی های a, b, cin هستند.
 $S, cout$ خروجی های ما هستند که به شکل زیر طبق فرمول های مدارهای منطقی محاسبه میشوند.

$$S = Cin \oplus (A \oplus B)$$

$$Cout = Cin \oplus (A \oplus B) + AB$$

شکل مدار به صورت زیر خواهد بود:



در عکس زیر شکل را با وجود جدول کارنو مشاهده خواهید کرد:

Logisim: main of Untitled

File Edit Project Simulate Window Help

Combinational Analysis

File Edit Project Simulate Window Help

Inputs Outputs Table Expression Minimized

a	b	c	x	y
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

Build Circuit

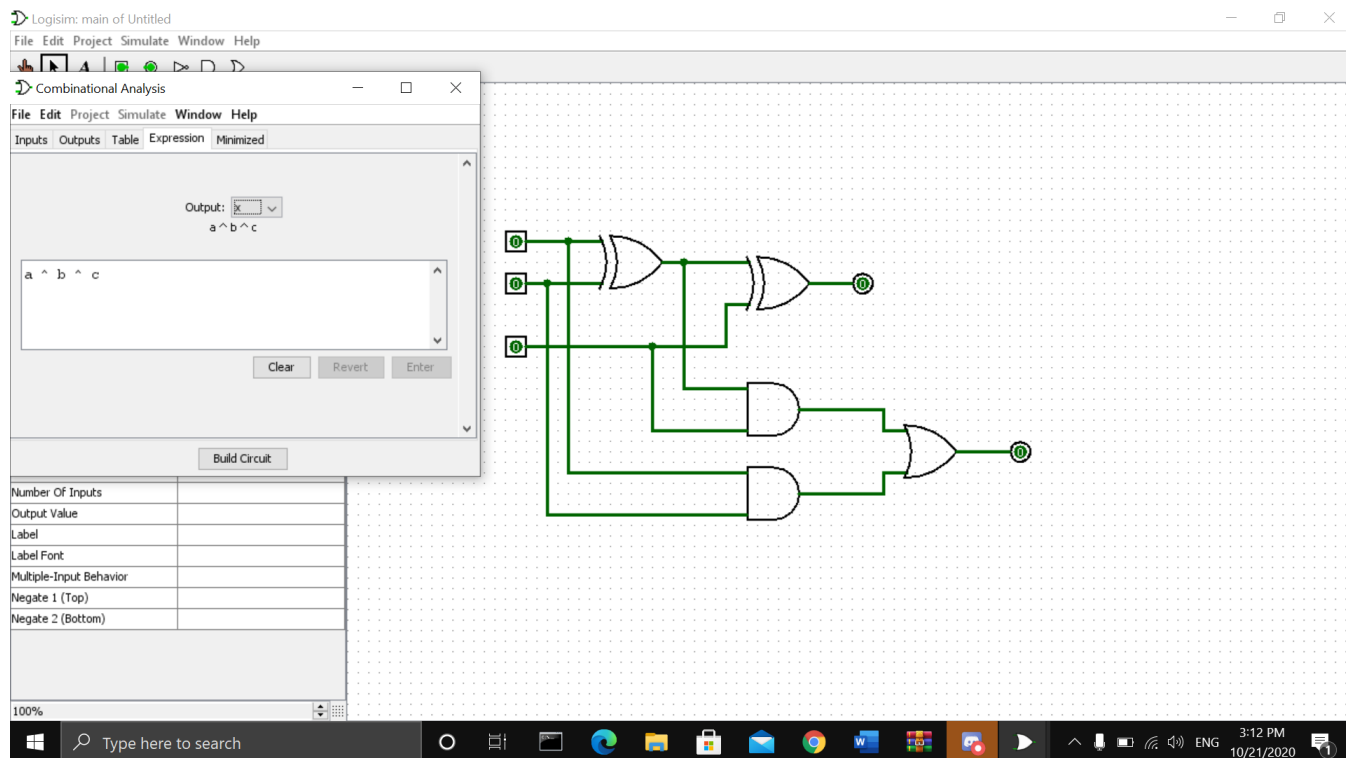
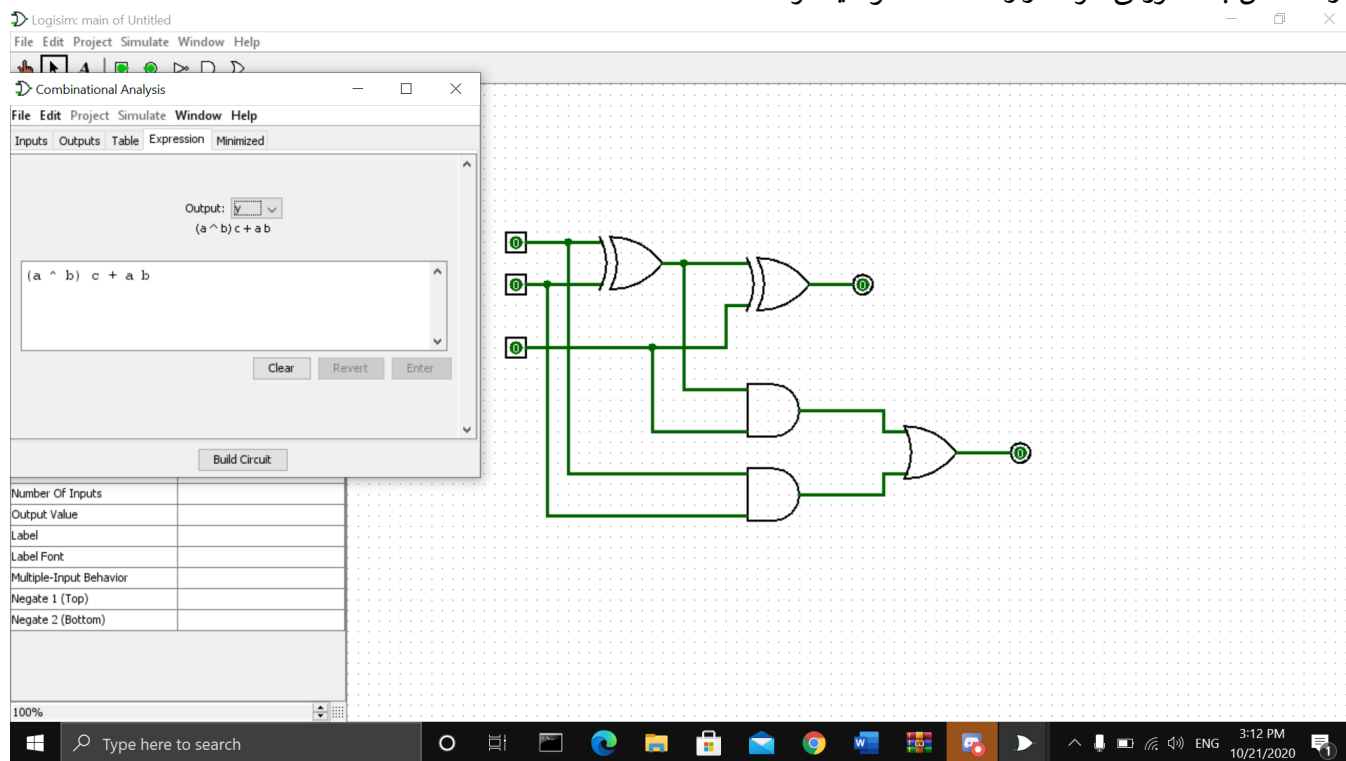
Number Of Inputs
Output Value
Label
Label Font
Multiple-Input Behavior
Negate 1 (Top)
Negate 2 (Bottom)

100%

Type here to search

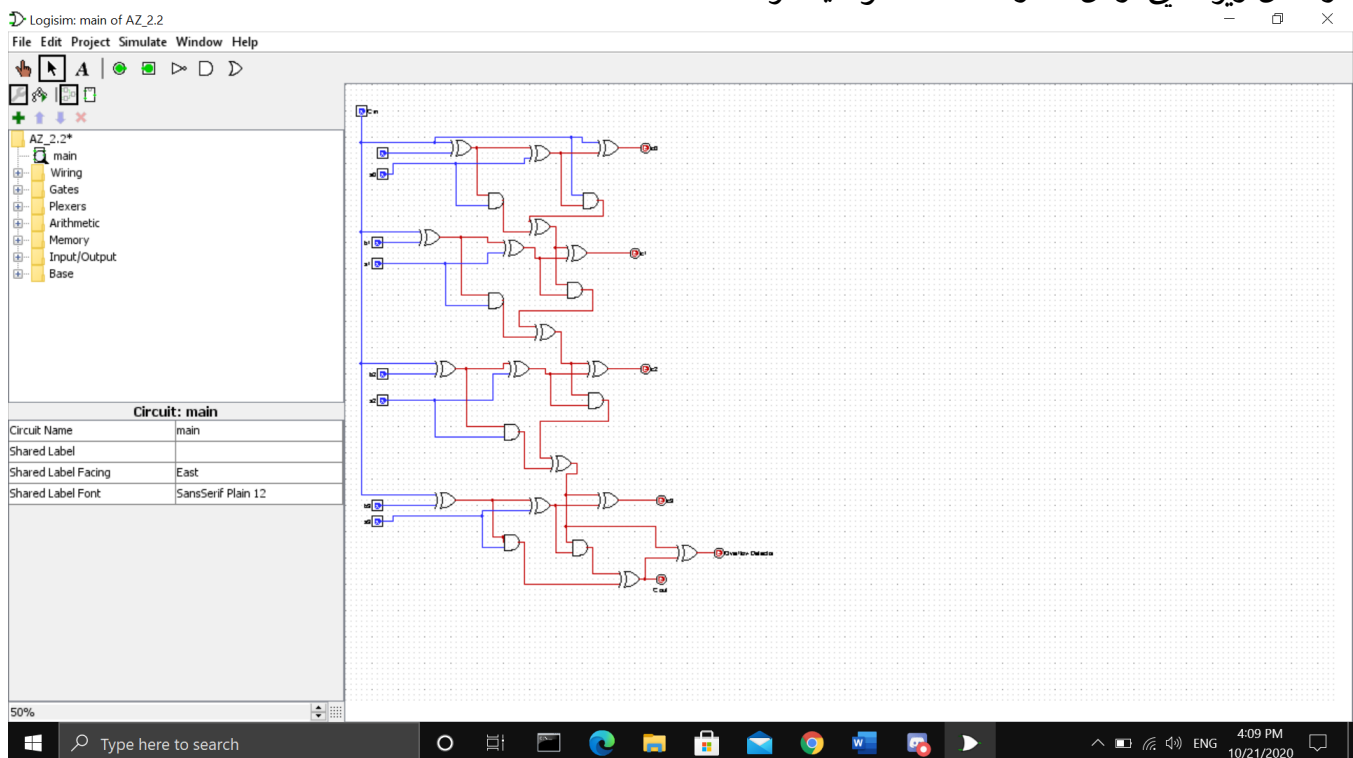
3:12 PM
10/21/2020

در 2 عکس بعد خروجی هر مدار را مشاهده خواهید کرد:

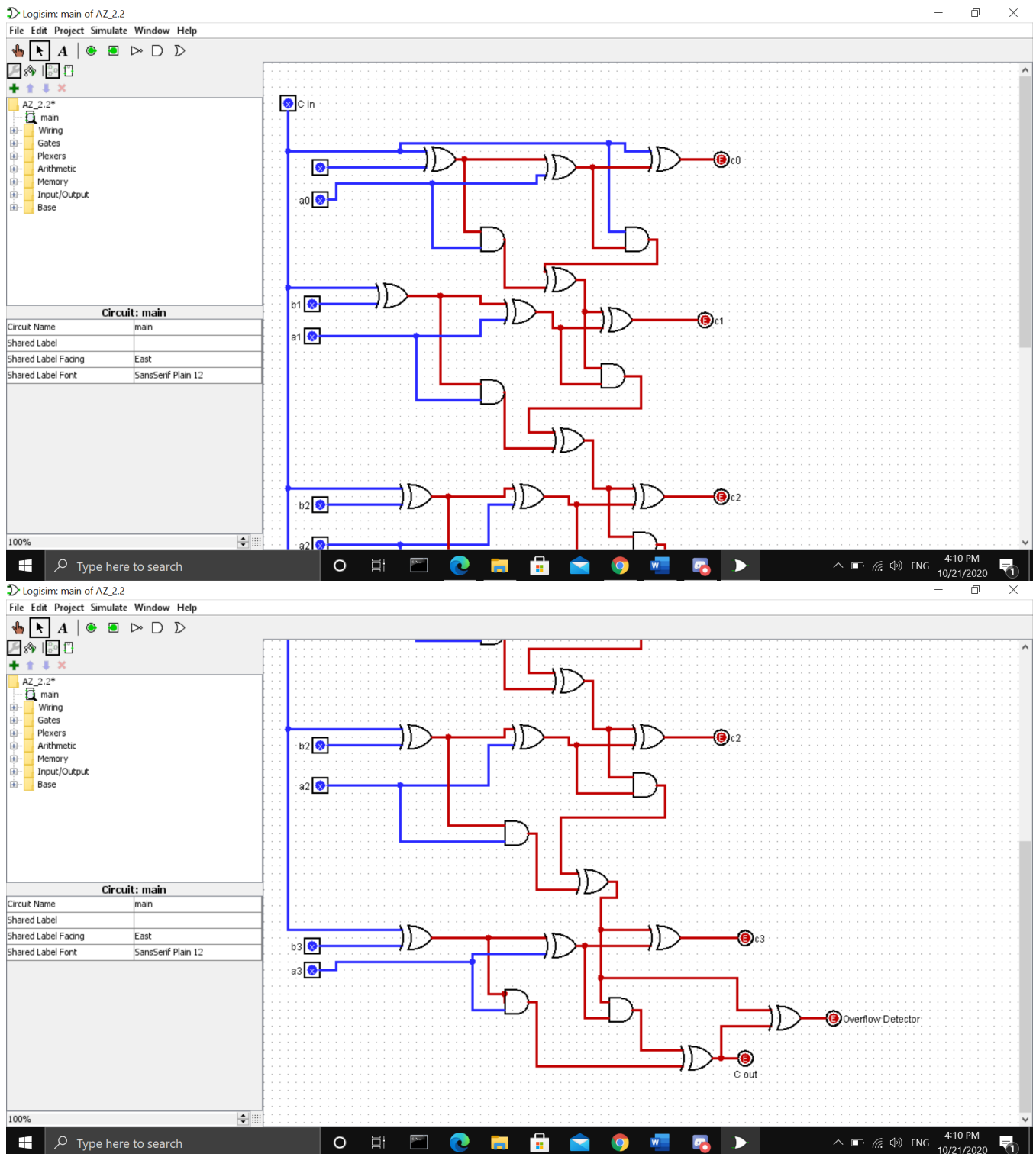


در بخش دوم آزمایش ما یک 4 bit adder میسازیم:
نحوه ی کار این Adder به این طریق است که 2 عدد 4 بیتی A و B و بیت کنترل Cin گرفته میشوند. و اگر Cin برابر با 0 باشد مدار باید $B - A$ را تولید کند و در غیر این صورت باید $B + A$ را محاسبه کند. برای این کار Cin با تمام بیت های عدد B XOR میشوند. تا در صورت 1 بودن عدد B تمام بیت هایش نقیض شوند و برای تبدیل کردن عدد B به مکمل 2 Cin به عنوان ورودی به Adder Full اول داده میشود. دقت کنید در صورت 0 بودن Cin تغییری در بیت های اعداد به وجود نمیاید و آن ها به صورت عادی جمع میشوند. در انتها نیز بیت کرای آخر با یکی مانده به آخر XOR میشوند تا بتوانیم از وجود Overflow با خبر شویم.

در شکل زیر نمایی از کل مدار مشاهده خواهید کرد:



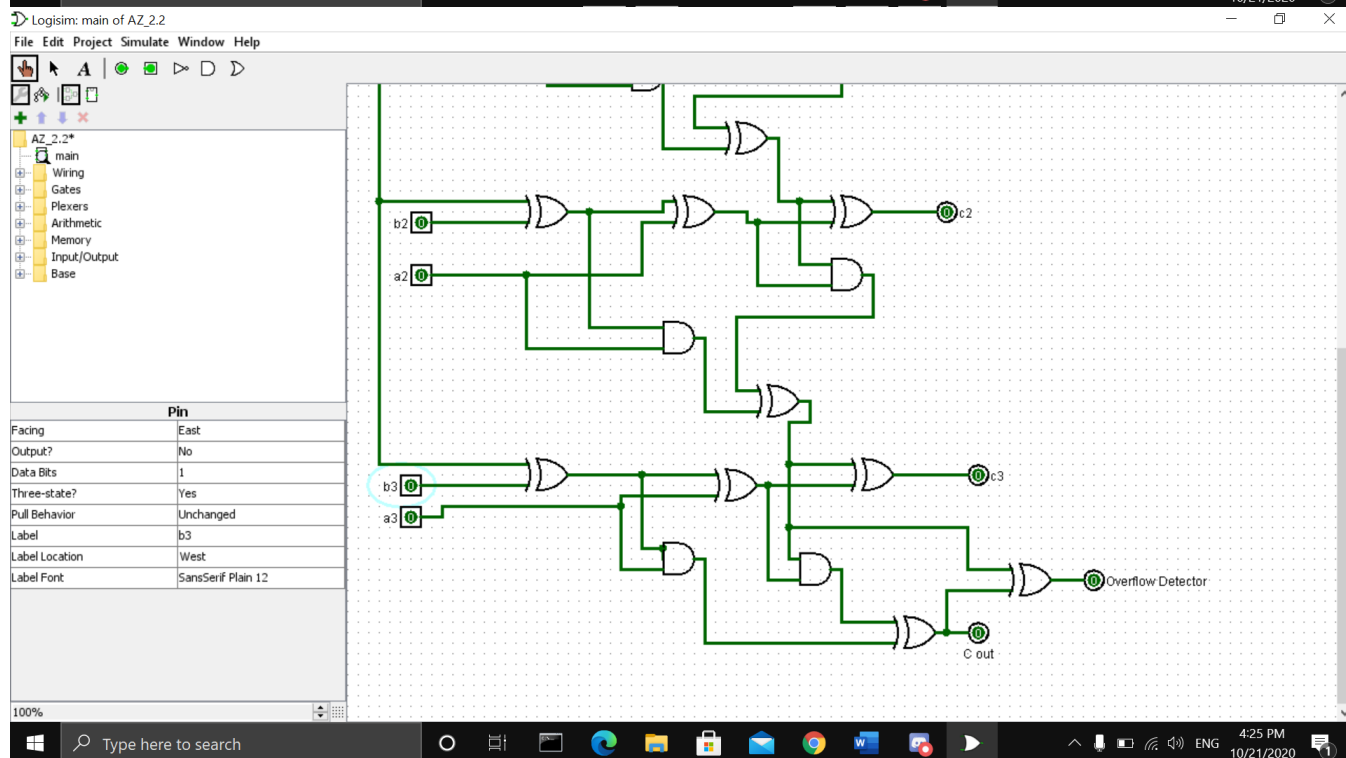
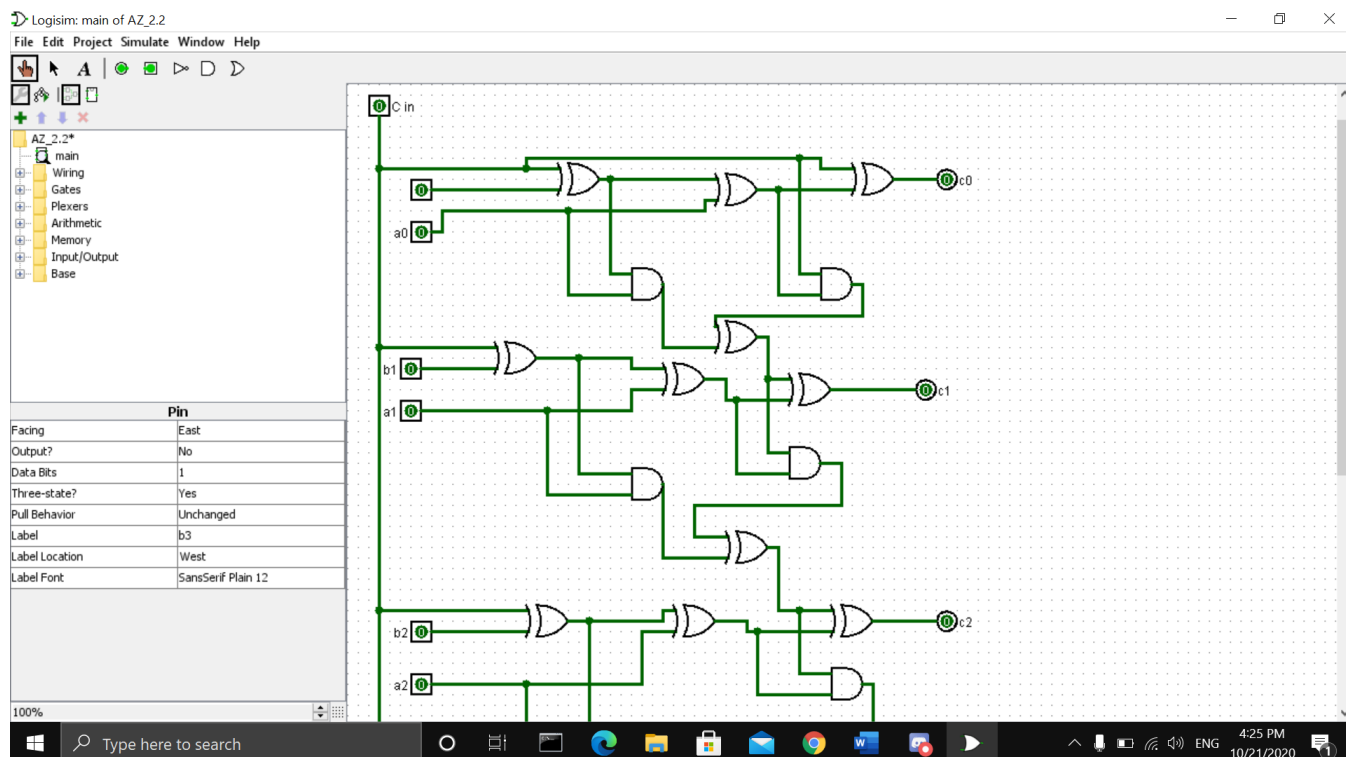
در شکل های زیر تمام مدار را به صورت بزرگ تر مشاهده خواهید کرد:



در شکل زیر جدول صحت را ملاحظه خواهید کرد:

[illegible]

در شکل زیر نمایی از مدار را میبینید که مقدار دهی شده است:



آزمایش سوم:

هدف آزمایش:

ساخت یک جمع کننده چهاربیتی از نوع C-look-ahead :

این Adder با نوع قبلی فرق دارد، و به گونه ای سریع تر از آن است.
فرمول آن را در زیر می آوریم:

$$C_{i+1} = G_i + P_i + C_i$$

$$S_i = P_i \text{ XOR } C_i$$

طبق فرمول های بالا در هر مرحله C_i میتواند به طور بازگشتی و مستقیم محاسبه شود، و نیازی به منتظر ماندن هر Adder برای Adderهای قبل نیست.

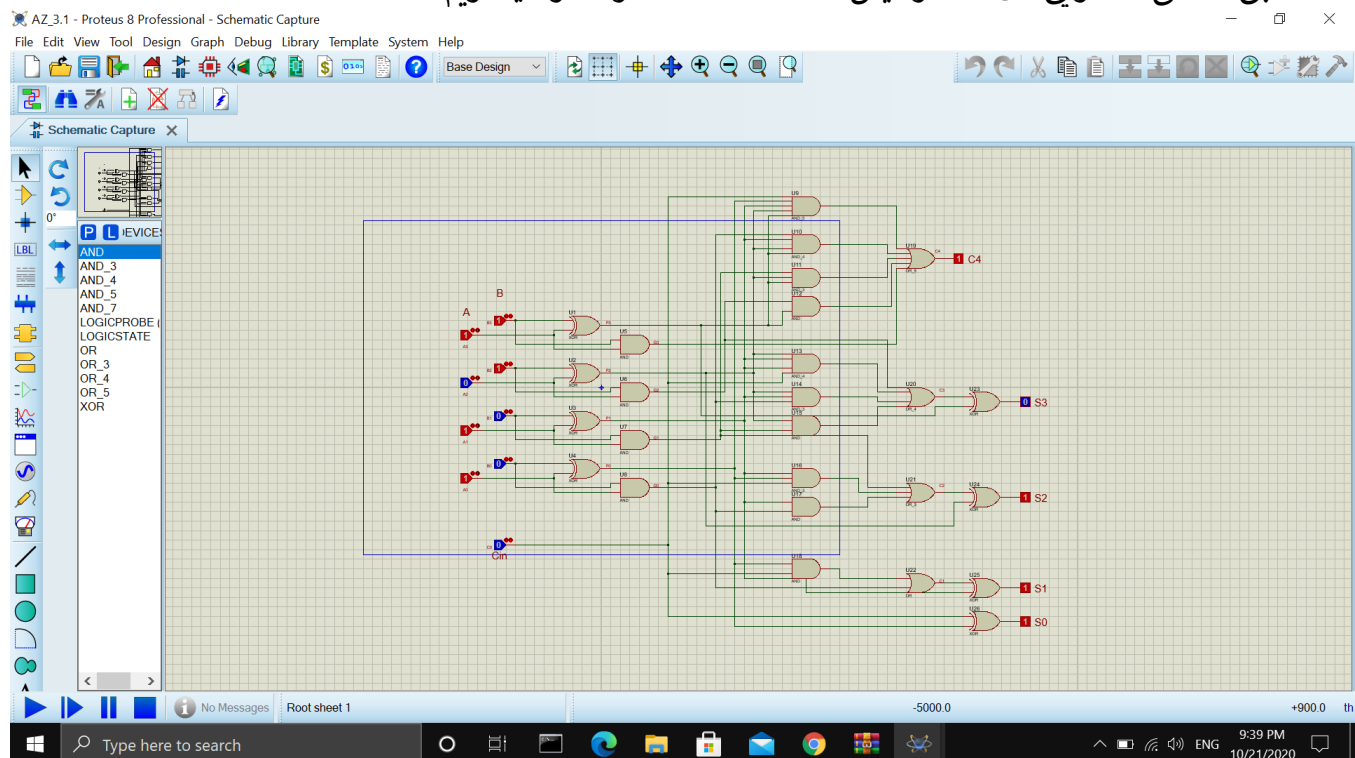
$$C_4 = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0 + P_3P_2P_1P_0C_{in}$$

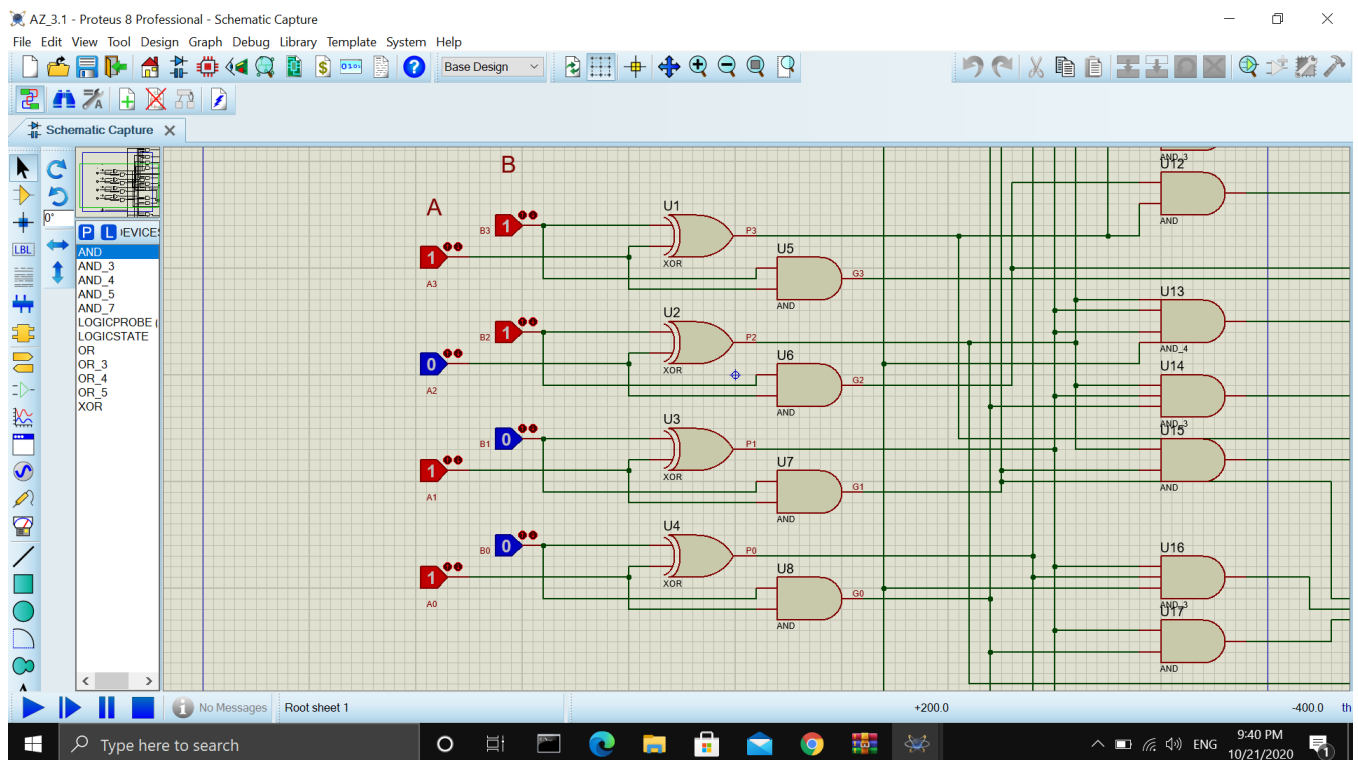
$$C_3 = G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0C_{in}$$

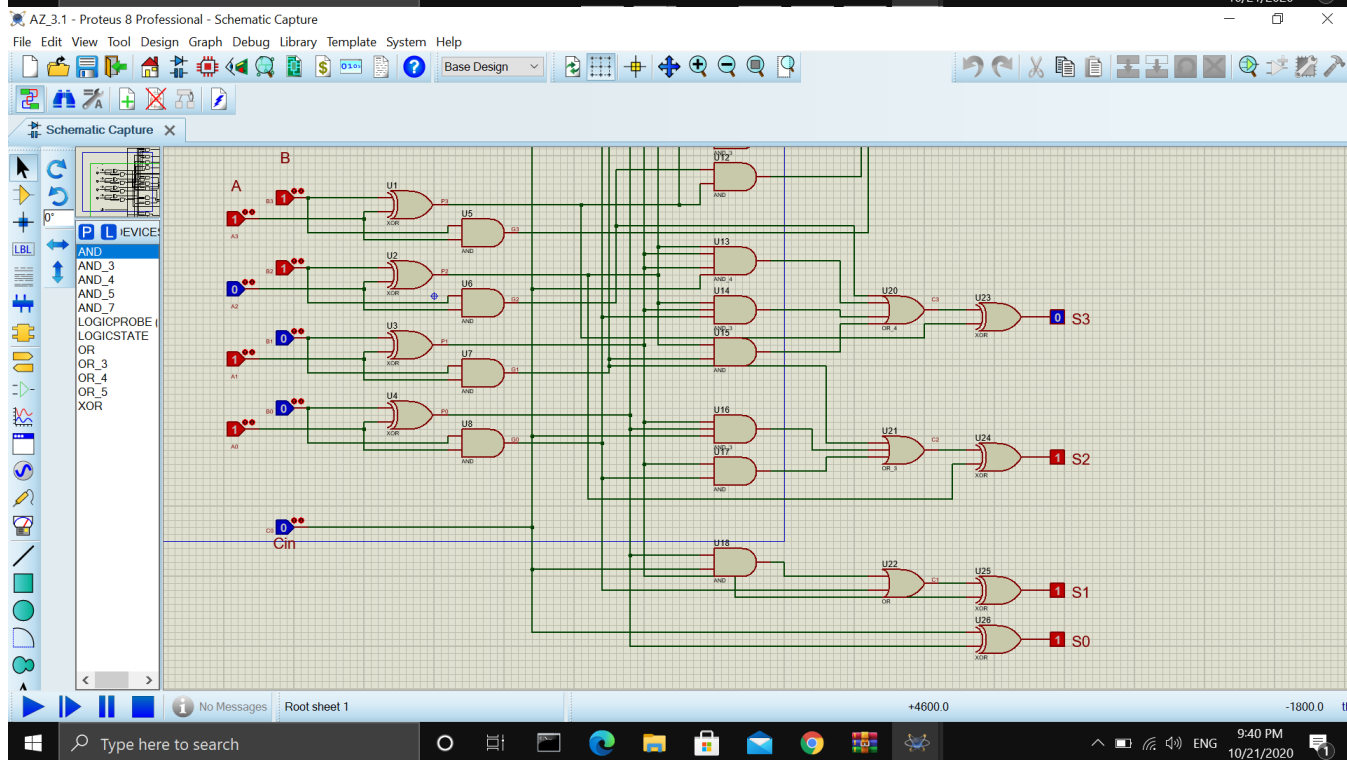
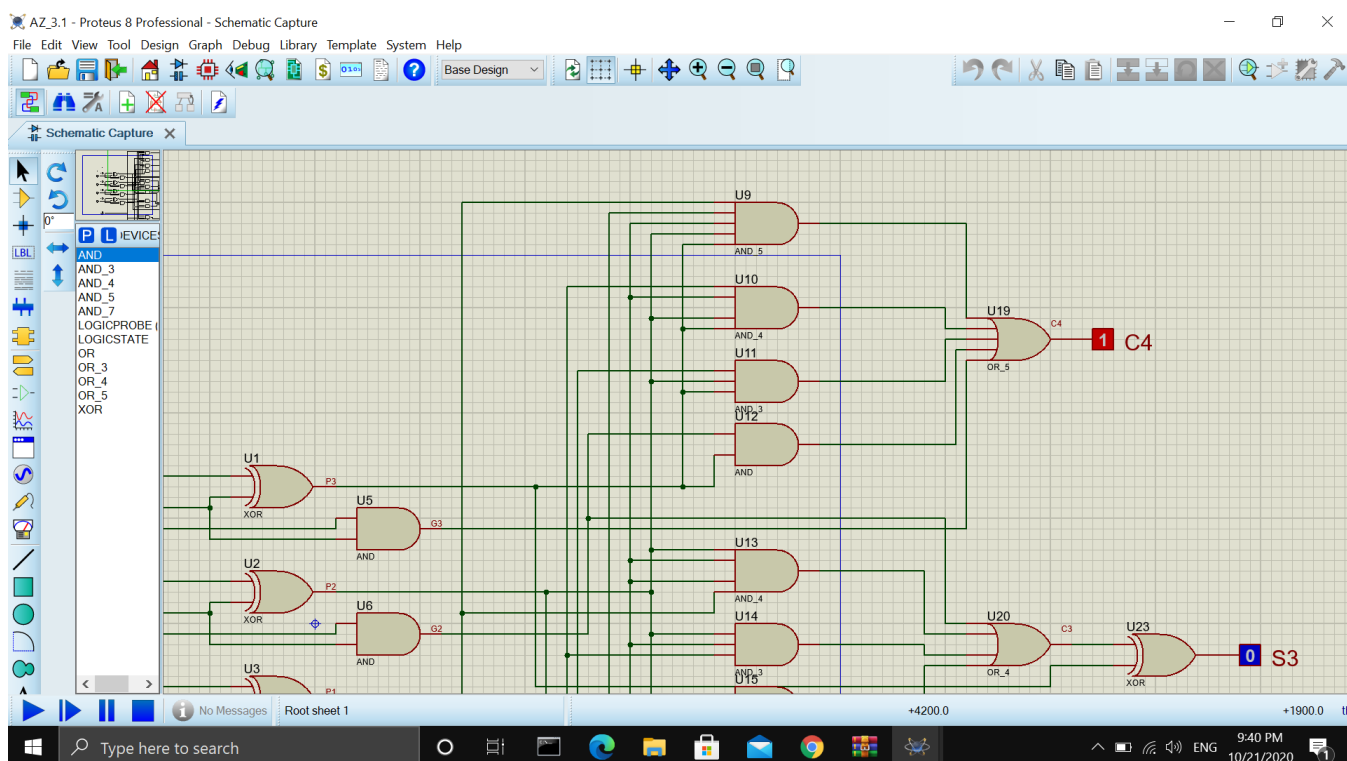
$$C_2 = G_1 + P_1G_0 + P_1P_0C_{in}$$

$$C_1 = G_0 + P_0C_{in}$$

حالا طبق عکسی که در پی دی اف آزمایش ها آمده است مدارمان را میسازیم:







در سوال آزمایش 2 بخش 2 از دوستم و از اینترنت کمک گرفته شده است.
در سوال 3 هم از اینترنت کمک گرفته شده است.