

به نام خدا



آز معماری – دکتر سربازی آزاد

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

گزارش آزمایش ۵

اعضای گروه:

میترا قلی پور-۴۰۱۱۰۶۳۶۳

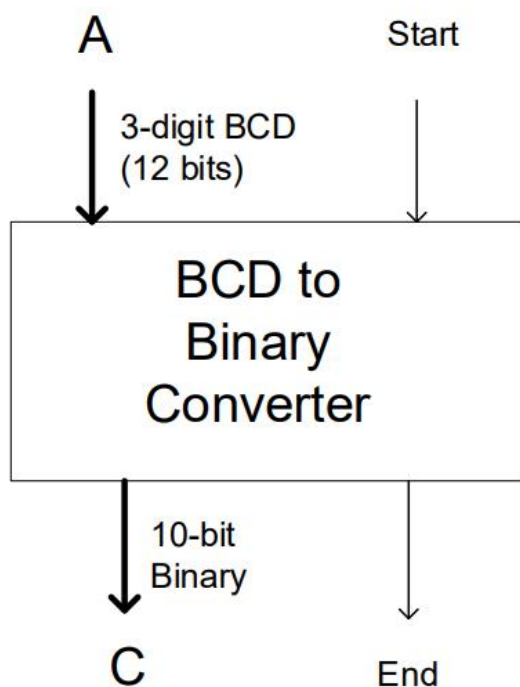
نیکا قادری-۴۰۱۱۰۶۳۲۸

ملیکا علیزاده-۴۰۱۱۰۶۲۵۵

## هدف و نتیجه مورد انتظار

در این آزمایش به طراحی و ساخت مدار مبدل دهدهی به دودویی می‌پردازیم. در این آزمایش، با فعال شدن سیگنال Start، انتظار می‌رود عدد دودویی معادل عدد سه رقمی دهدهی ورودی محاسبه شود و پایان کار با فعال شدن سیگنال End مشخص شود.

## شرح آزمایش



با فعال شدن سیگنال Start، مدار شروع به کار کرده و ورودی دهدهی که یک عدد سه رقمی است را به معادل دودویی آن تبدیل کرده و نتیجه را روی خطوط خروجی قرار می‌دهد و سیگنال End را به عنوان اعلام پایان عملیات فعال می‌کند. (شکل ۱)

شکل ۱: مبدل دهدهی به دودویی

## الگوریتم

برای تبدیل یک عدد دهدهی ۲ رقمی به معادل دودویی به صورت زیر عمل می‌کنیم:

- ۱- عدد دهدهی ورودی را یک بیت به راست شیفت می‌دهیم.
- ۲- اگر با ارزشترین بیت رقم  $i$  ام یک باشد از آن رقم ۳ تا کم می‌کنیم.
- ۳- مراحل اول و دوم را آنقدر تکرار می‌کنیم تا تمام ارقام دهدهی صفر شوند.
- ۴- در پایان، بیت‌هایی که با شیفت به راست بیرون می‌آیند، عدد دودویی معادل عدد دهدهی ورودی را تشکیل می‌دهند.

برای مثال عدد دهمی ۱۰۱ طبق این الگوریتم به این صورت به معادل دودویی تبدیل می‌شود:

عمل	خروجی	رقم ۱	رقم ۲	رقم ۳
شیفت به راست	0	0000	0001	0001
از رقم‌های ۱ و ۲ سه تا کم کن	0	1000	1000	0000
شیفت به راست	0	0101	0101	0000
از رقم ۱ سه تا کم کن	10	1010	0010	0000
شیفت به راست	10	0111	0010	0000
شیفت به راست	110	0011	0001	0000
از رقم ۱ سه تا کم کن	1110	1001	0000	0000
شیفت به راست	1110	0110	0000	0000
شیفت به راست	01110	0011	0000	0000
شیفت به راست	101110	0001	0000	0000
پایان عملیات	1101110	0000	0000	0000

## بخش اول: طراحی ASM chart و مدار اصلی

با توجه به عملکرد مبدل ددهی به دودویی مدار ما سه استیت اصلی دارد که به صورت زیر هستند:

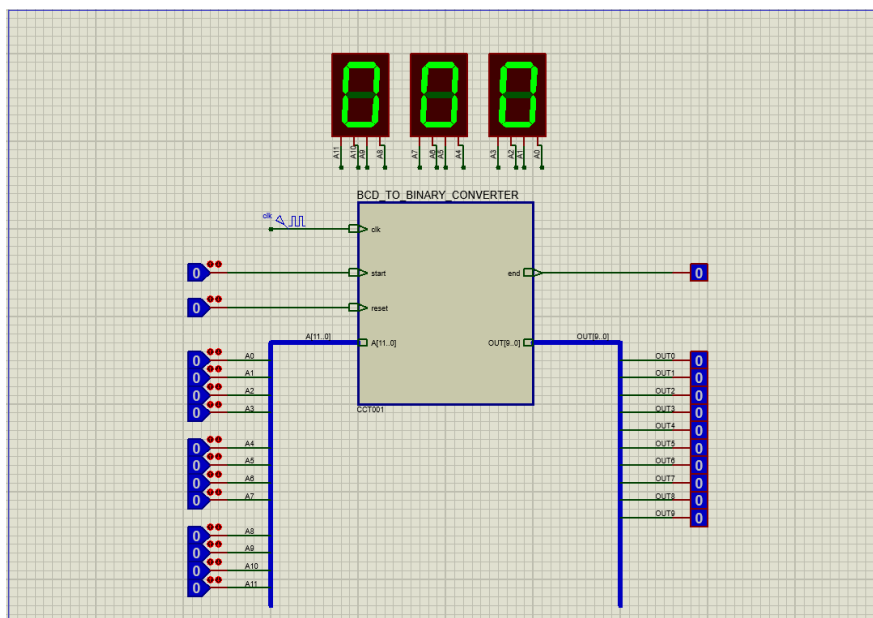
در استیت اول با فعال شدن سیگنال شروع ارقام عدد ددهی در ۳ رجیستر ریخته می‌شوند و مقادیر سیگنال پایان و بیت‌های خروجی و شمارنده‌ها 0 می‌شوند تا بعد از محاسبات مقداردهی شوند.

در استیت دوم باید طبق الگوریتم باید تا زمانی که تمام بیت‌های ددهی صفر نشده‌اند و شمارنده به ده نرسیده، عدد ددهی ورودی را یک بیت به راست شیفت دهیم. شمارنده را یک واحد افزایش داده و به استیت سوم یا تفریق می‌رویم. زمانی که تمام بیت‌های ددهی صفر شدند خروجی تا زمانی که شمارنده به عدد ۱۰ برسد شیفت می‌خورد و خروجی نهایی آماده و سیگنال پایان فعال می‌شود.

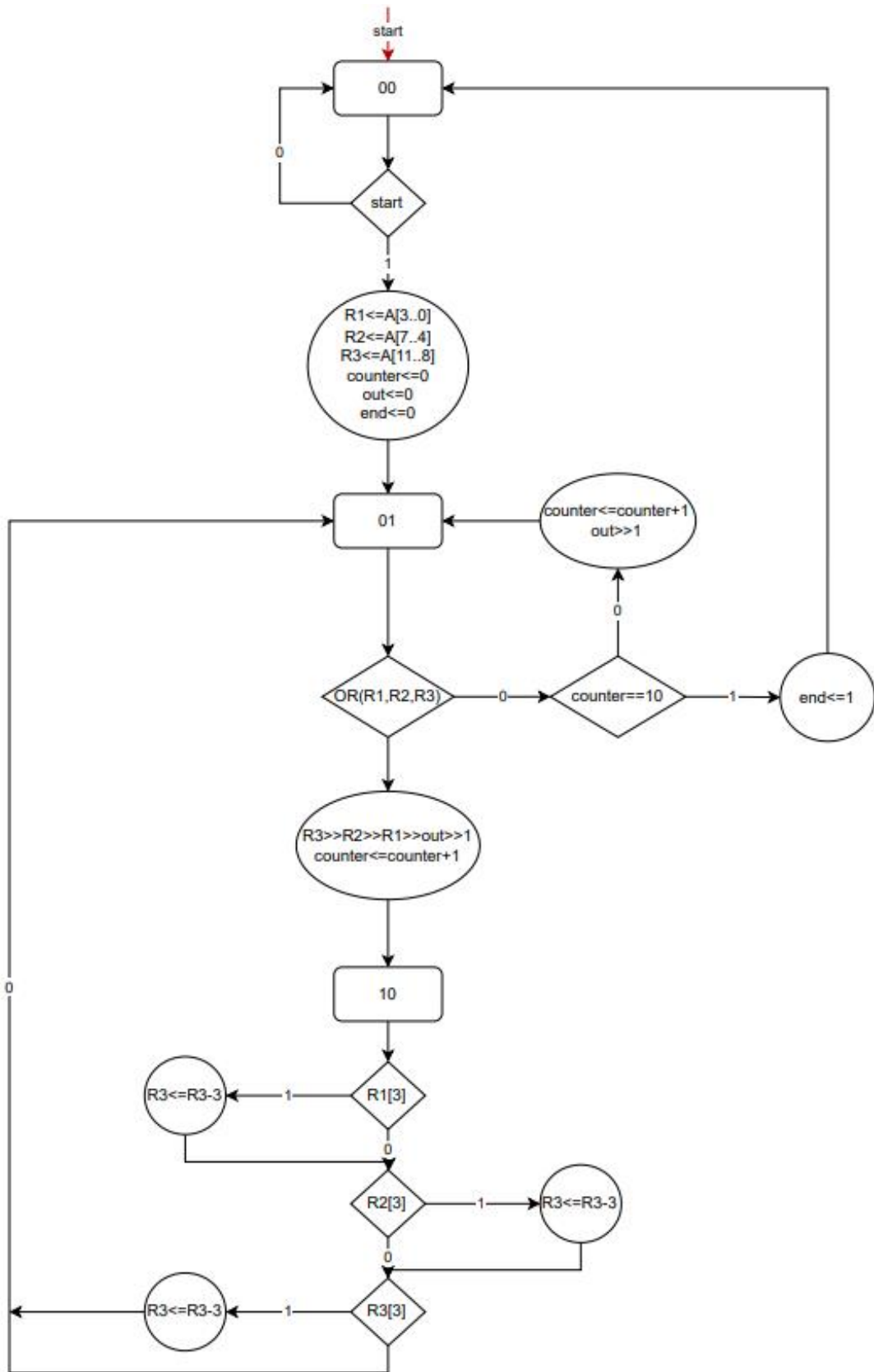
در استیت سوم که همان حالت تفریق است در صورتی که با ارزشترین بیت یک رقم یک باشد ۳ تا از آن رقم کم می‌کنیم.

چارت رسم شده در شکل ۳ نمایش داده شده است.

طبق این چارت مدار اصلی را طراحی می‌کنیم. در مدار اصلی کلاک، سیگنال شروع و ریست و عدد ددهی ۳ رقمی (به صورت BCD) را ورودی می‌گیرد و در خروجی سیگنال پایان و عدد دودویی (نمایش با ۱۰ بیت) وجود دارد. در شکل ۲ شمای کلی مدار نمایش داده شده است.



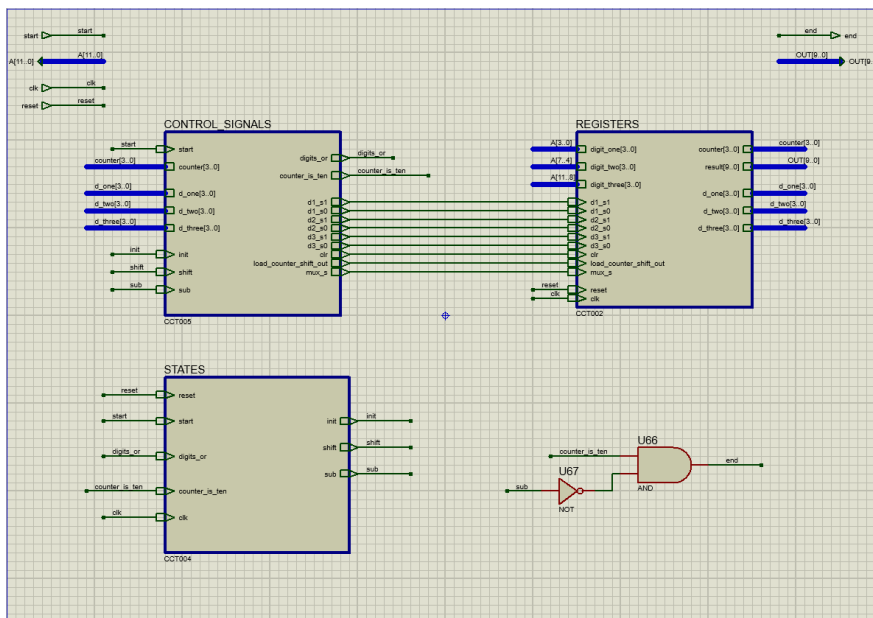
شکل ۲: مدار اصلی



شکل ۳: ASM chart

## بخش دوم: ساخت مدار

ماژول اصلی مدار ما از سه بخش اصلی رجیسترها و سیگنال‌های کنترلی و واحد کنترل حالت‌ها تشکیل شده است که هر بخش را به صورت جدا توضیح می‌دهیم. همچنین در این بخش با توجه به اینکه counter ده شده است یا نه و اینکه در استیت نهایی هستیم سیگنال end فعال می‌شود. (شکل ۴)



شکل ۴: داخل مازول اصلی

## کنترل حالت‌ها

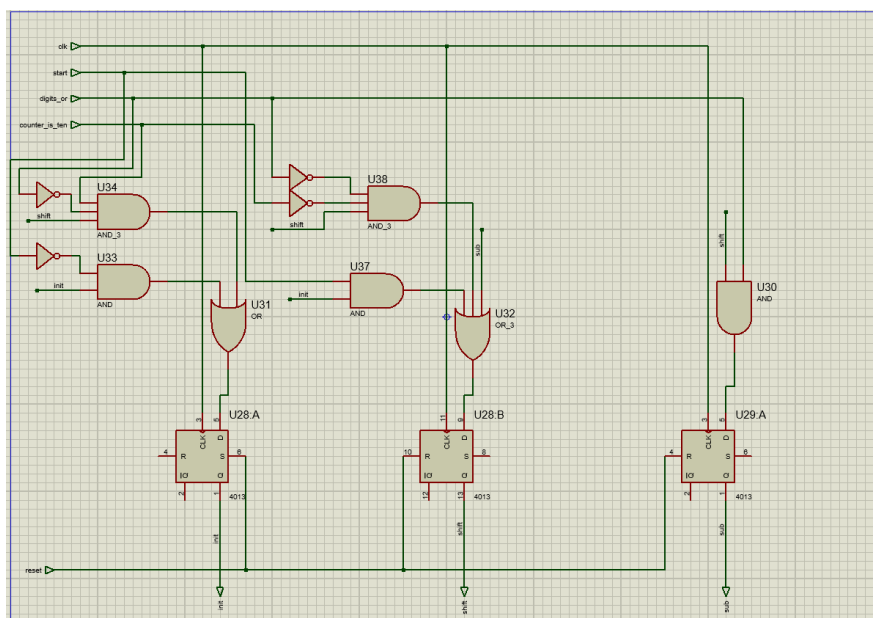
در واحد کنترل حالت‌های مختلف مدار طبق چارت بررسی می‌شود. در این بخش سیگنال‌های ریست و شروع و حاصل  $or$  بیت‌های ورودی ددهی و سیگنال کنترلی برای ده بودن شمارنده و کلاک را به عنوان ورودی گرفته و طبق روابط زیر، سه حالت چارت را به عنوان خروجی این ماژول بدست می‌آوریم. (شکل ۵)

$$S_0^+ = start.S_0 + \overline{or(A[11..0])}.S_1.counterTen$$

$$S_1^+ = start.\bar{S}_0 + \overline{or(A[11..0])}.S_1.counterTen + S_2$$

$$S_2^+ = S_1.\overline{or(A[11..0])}$$

حال با توجه به این روابط مدار مورد نظر را می‌سازیم.

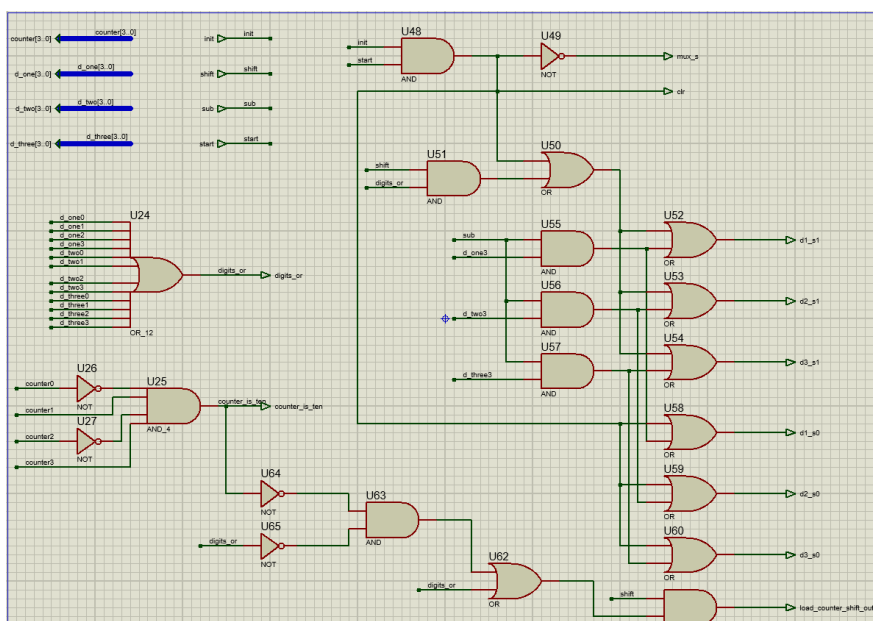


شکل ۵: واحد کنترل حالت‌ها

## سیگنال‌های کنترلی

با توجه به حالت خروجی واحد کنترل به ساخت ماژول سیگنال‌های کنترلی می‌پردازیم و در این بخش باید سیگنال‌های میانی را با توجه به حالت فعلی مدار و عدد ددهی و شمارنده را تولید کنیم. خروجی  $digits\_or$

همان **or** بیت‌های عدد دهمی است که در استیت دوم استفاده شده می‌شود و خروجی **counter\_is\_ten** نشان‌دهنده این است که شمارنده به ۱۰ رسیده یا نرسیده است. طبق چارت اگر هنوز همه‌ی بیت‌های عدد دهمی صفر نشده‌اند یا شمارنده به ۱۰ نرسیده باشد و در استیت شیفت باشیم باید **load\_counter\_shift\_out** فعال شود و که نشان می‌دهد که باید عدد دهمی را یکی شیفت دهیم و به شمارنده یکی اضافه کنیم. همچنین خروجی **mux\_s** برای زمانی است که در استیت ابتدایی باشیم و سیگنال شروع فعال شود. (شکل ۶)

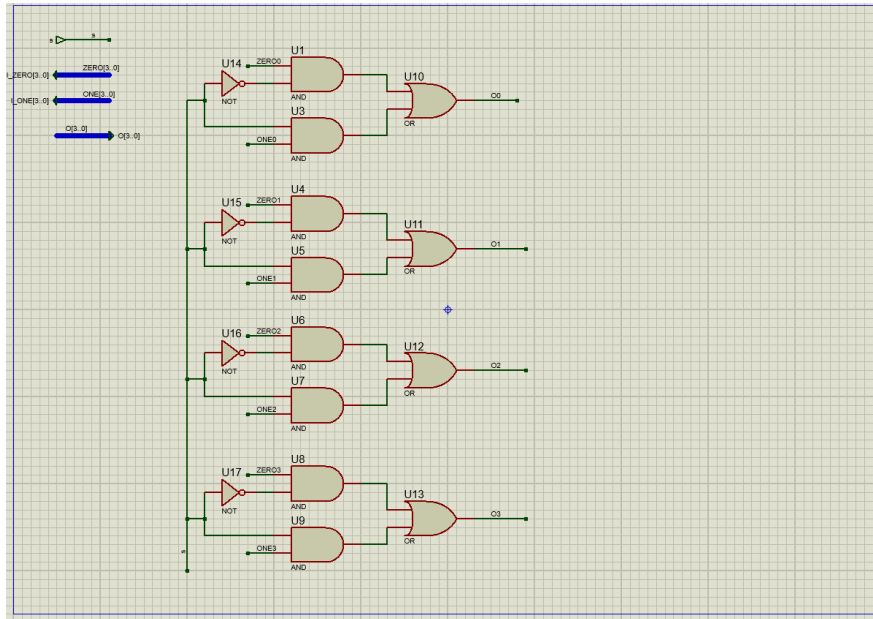


شکل ۶: سیگنال‌های کنترلی

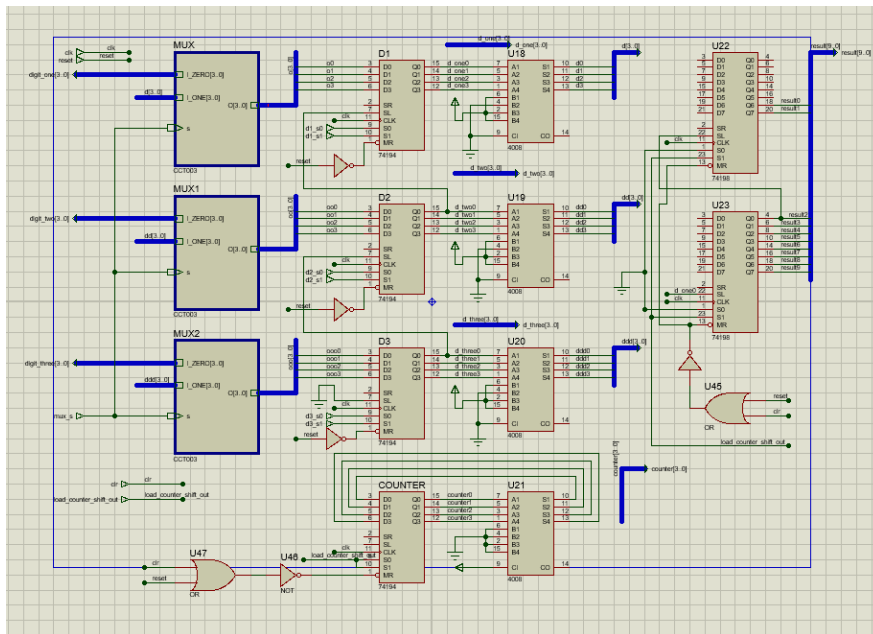
## رجیسترها

این بخش از مدار با گرفتن ارقام عدد دهمی ورودی، آنها را در رجیسترها ذخیره می‌کند. ماکس‌های دو به یک (شکل ۷) با توجه به مقدار ورودی **mux\_s** ارقام ورودی جدید را لود می‌کنند یا از ورودی قبلی استفاده می‌کنند. همچنین برای نگهداری ارقام از تراشه ۷۴۱۹۴ که شیفت رجیستر ۴بیتی است و برای کم کردن ۳ از ارقام نیز از تراشه ۴۰۰۸ که جمع کننده ۴بیتی است (ورودی **A** این جمع کننده رقم مورد نظر و ورودی **B** مکمل دو عدد ۳ است.) و از دو تراشه ۷۴۱۹۸ که شیفت رجیستر ۸بیتی است نیز برای نگهداری عدد دودویی خروجی استفاده می‌کنیم. همچنین برای شمارنده نیز از یک تراشه ۷۴۱۹۴ نیز استفاده کردیم که با وصل کردن آن به یک جمع کننده به آن یک واحد اضافه شود. (شکل ۸)





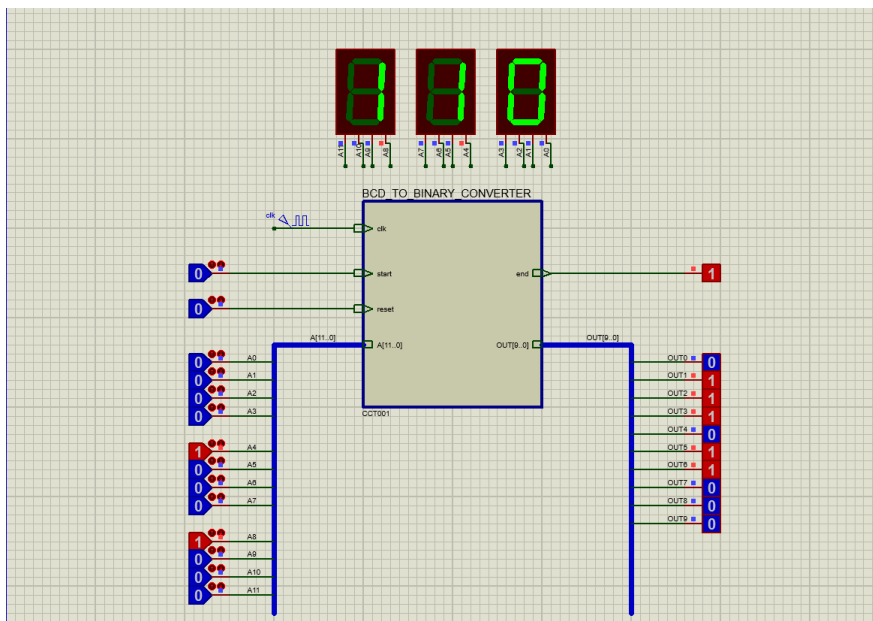
شکل ۷: ماکس های دو به یک



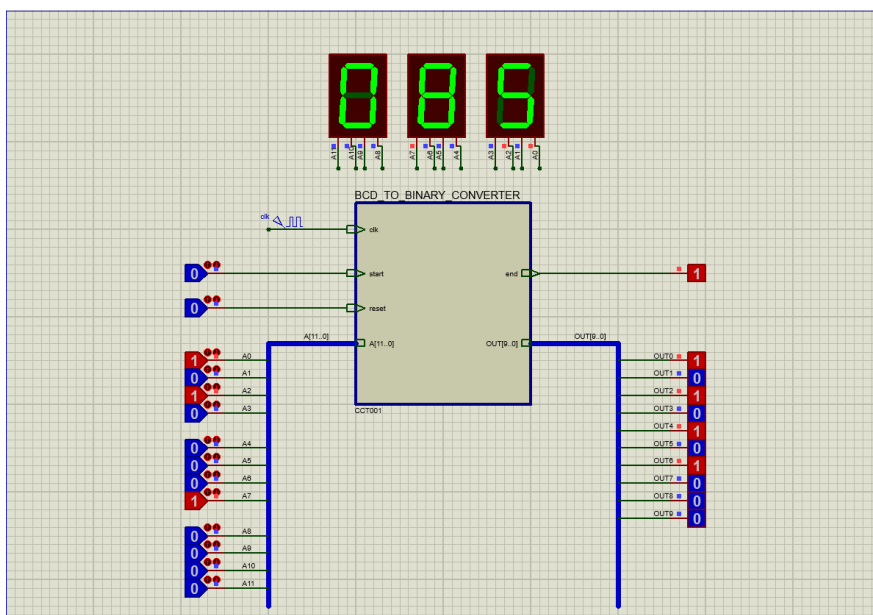
شکل ۸: رجیسترها

## بخش سوم: تست

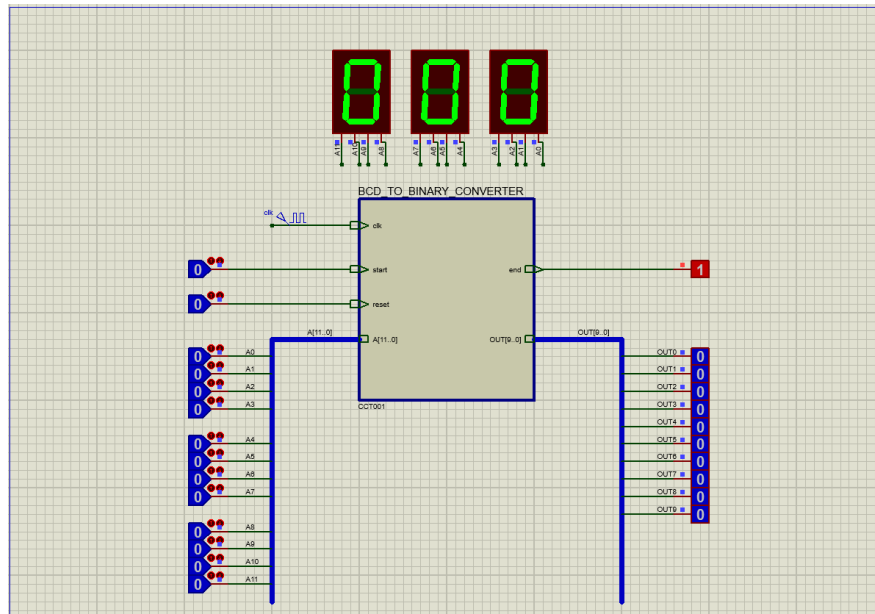
در این بخش به تست مدار طراحی شده می‌پردازیم.



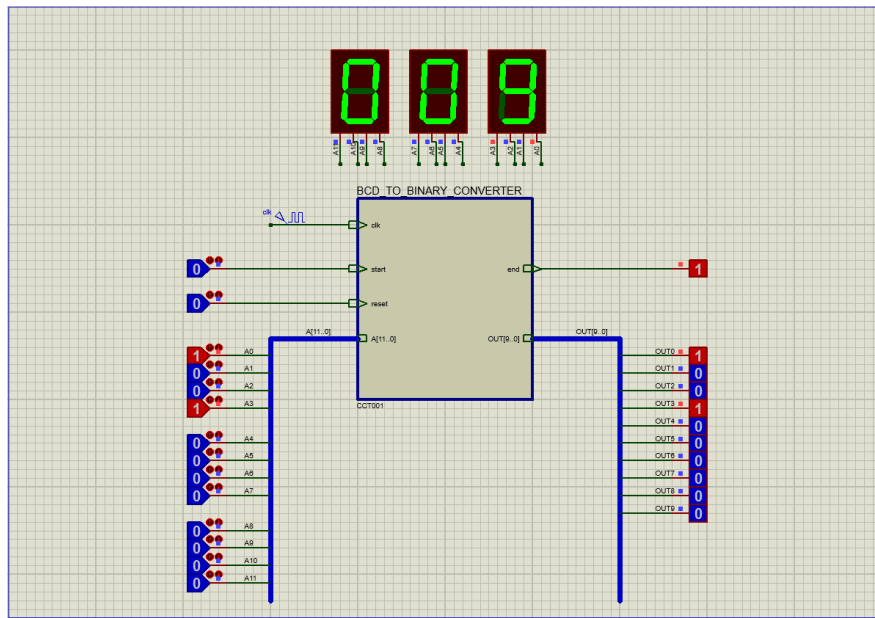
شکل ۹: تست ۱- عدد ۱۱۰



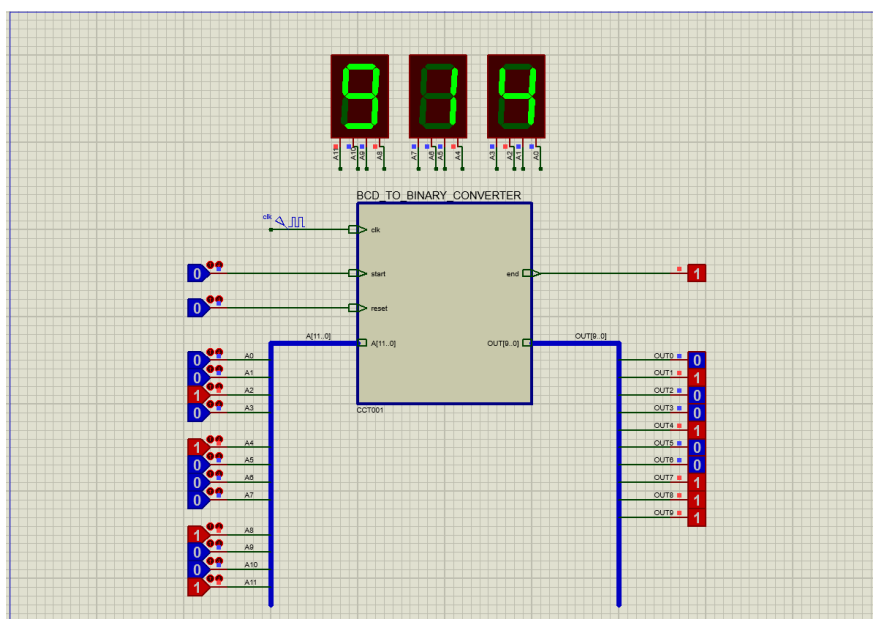
شکل ۱۰: تست ۲- عدد ۱۱۵



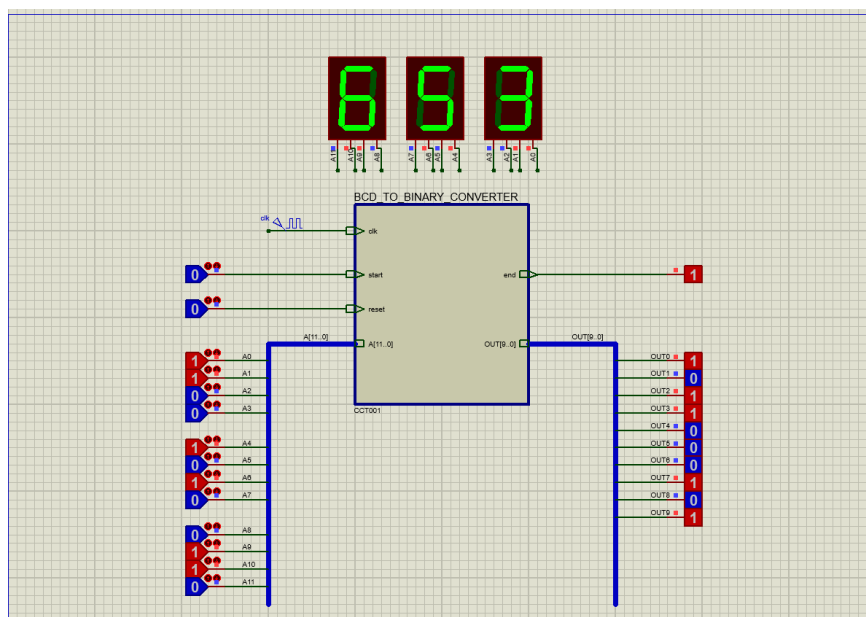
شکل ۱۱: تست ۳-عدد ۰



شکل ۱۲: تست ۴-عدد ۹



شکل ۱۳: تست ۵- عدد ۹۱۴



شکل ۱۴: تست ۶- عدد ۶۵۳