



## پیش‌گزارش آزمایش شماره ۸ آزمایشگاه مدارمنطقی

### هدف: «ساخت یک کامپیوتر ابتدایی با قابلیت انجام عملیات بولی»

امیرحسین محمدزاده ۴۰۲۱۰۶۴۳۴ \*\*\*\*\* کسری منتظری ۴۰۲۱۰۶۵۷۵

استاد مربوطه: دکتر انصاری – دستیار آموزشی: جناب آقای پورعاشوری

#### • توضیحات:

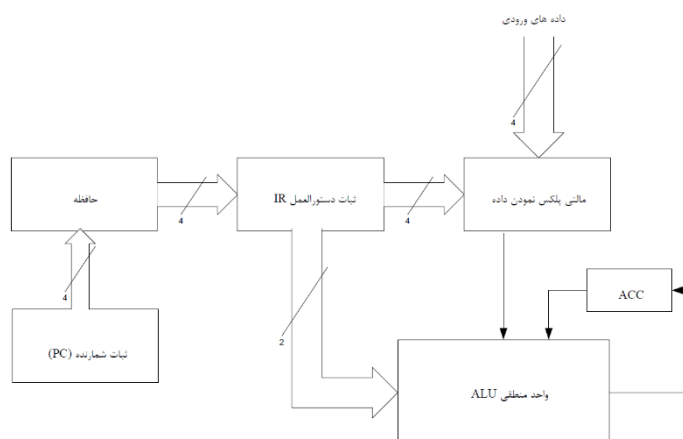
- در این آزمایش، می‌خواهیم یک کامپیوتر دودوئی ساده ۴ بیتی پیاده کنیم که دارای ۴ دستورالعمل (Instruction) می‌باشد.

ACC	←	ACC	OR	Operand
ACC	←	~ACC		
ACC	←	ACC	AND	Operand
HALT				

عملوندهای ورودی می‌تواند یکی از x, y, z, w باشد و opcode های دستورالعمل‌ها هم به فرمت زیر است.

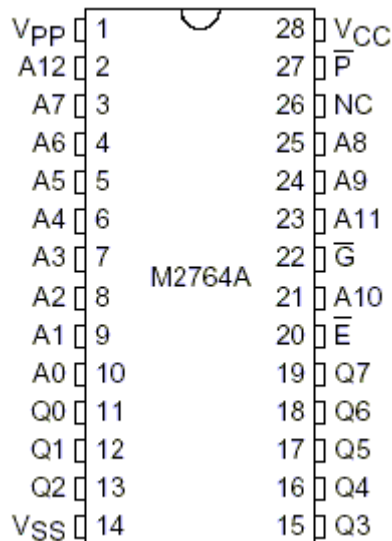
$y_1y_2 = 00$	عمل AND	$y_3y_4 = 00$	متغیر x
$y_1y_2 = 01$	عمل Complement	$y_3y_4 = 01$	متغیر y
$y_1y_2 = 10$	عمل OR	$y_3y_4 = 10$	متغیر z
$y_1y_2 = 11$	عمل HALT	$y_3y_4 = 11$	متغیر w

توضیحات کلی کامپیوتر در صورت سوال داده شده است و معماری کلی این کامپیوتر از اجزای زیر تشکیل شده است:



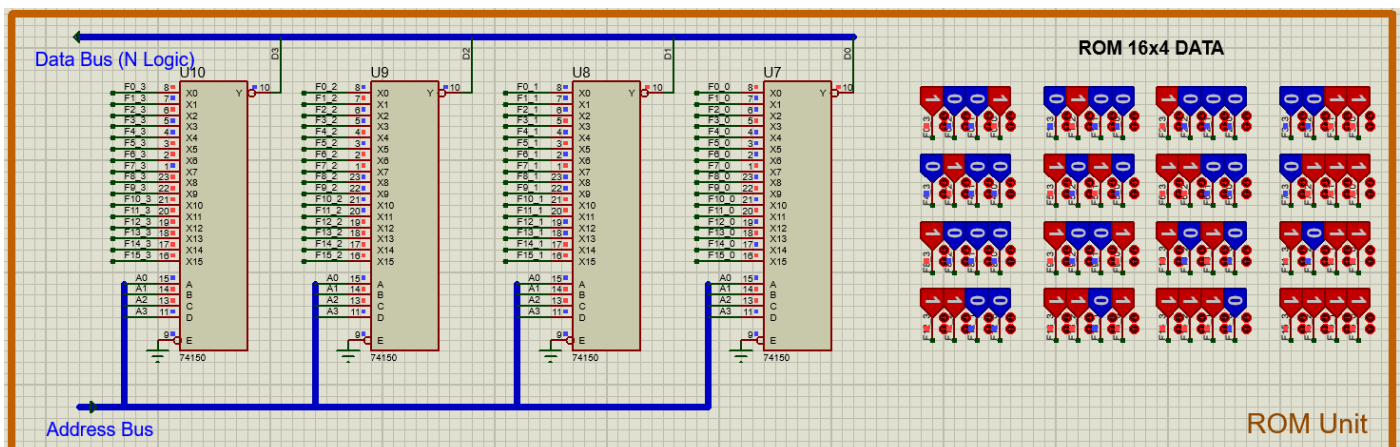
در این کامپیوتر سعی کردیم در لبه بالا رونده کلاک، PC را اضافه کنیم و نیز خروجی دستور قبلی را در ACC بنشانیم، و در لبه پایین رونده، دستور fetch شده را در IR بارگذاری کنیم. (نقیض کلاک را به تراشه مربوط به IR داده‌ایم.)

برای ساخت چنین کامپیوترهایی (که تا حدی می‌توان گفت در حد کامپیوترهای دهه ۱۹۹۰ هستند) نیاز به استفاده از برخی از انواع حافظه‌ها به نام PROM یا EPROM هستیم. حافظه‌های EPROM با نور فرابنفش Erase می‌شوند و پروگرام کردن آن‌ها نیز به شیوه الکترونیکی صورت می‌پذیرد. برای مثال، تراشه 2764 که دارای ۸ کیلوبایت حافظه است را در تصویر زیر می‌بینید:



برای استفاده از این حافظه‌ها باید از Address bus و Data bus استفاده کرد و آدرس را به آن‌ها داد و خروجی را تحویل گرفت. (البته برای نوشتن از ساز و کار دیگری استفاده می‌شود)

ما برای شبیه‌سازی یک ROM که دارای ۱۶ کلمه ۴ بیتی است، در پروتئوس، از ساز و کار زیر استفاده کردیم:



و داده‌های مربوط به machine code این کامپیوترمان را درون آن برنامه ریزی می‌کنیم.

این unit یک آدرس‌باس ۴ بیتی ورودی می‌گیرد و یک دیتا‌باس (با منطق منفی) ۴ بیتی خروجی می‌دهد.

برای شبیه‌سازی این بخش از ۴ عدد تراشه 74150 که MUX 4:16 است استفاده کرده‌ایم.

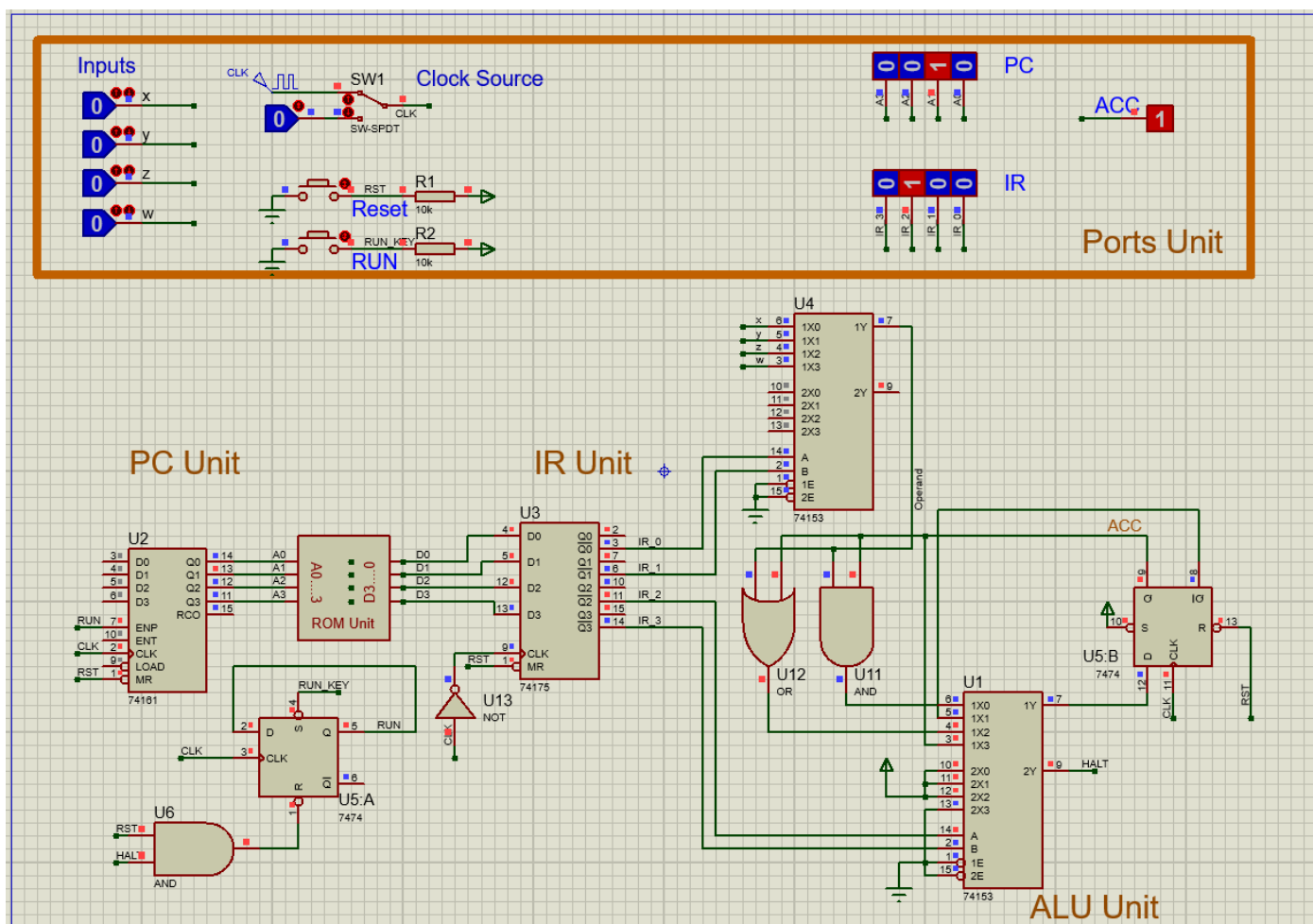
(همچنین یک نسخه پروتئوس دیگر، که در آن به جای ROM از RAM استفاده کرده‌ایم و پیش از فشردن دکمه RUN، باید ابتدا برنامه خود را به صورت دستی WRITE کنیم نیز موجود است که در آن از تراشه 2114 استفاده شده است.)

برای بخش PC (Program Counter) نیاز به یک شمارندهٔ باینری داشتیم که از تراشهٔ **74161** استفاده کرده‌ایم. همچنین برای بخش IR (Instruction Register) از یک عدد تراشهٔ **74175** که شامل ۴ D-FF است استفاده کرده‌ایم.

برای FF‌های پراکنده در مدار (که اینجا ۲ مورد استفاده شده که یکی برای نگهداری Accumulator - و دیگری برای latch کردن فرمان RUN در مدار) نیز از تراشهٔ **7474** استفاده می‌کنیم.

بخش ALU با توجه به اینکه کامپیوتر ما بسیار ساده می‌باشد صرفاً با یک عدد 4:2 MUX و نیز یک AND و یک OR ساخته می‌شود. برای MUX از تراشهٔ **74153** می‌توان استفاده کرد.

برای زیبایی و جلوگیری از شلوغ شدن مدار، با استفاده از Wire-Label‌ها در پروتئوس، به مدار نظم‌بخشی کرده‌ایم و ورودی‌ها و نشانگرها را به‌طور جداگانه در بالای مدار قرار داده‌ایم و می‌توان به‌صورت آسان به‌آنها دسترسی داشت:



مدار پروتئوس (بجز بخش ROM Unit که بالاتر عکس آن قرار داده شده) را در تصویر فوق مشاهده می‌کنید.

همچنین برای قابلیت تست آسان‌تر، می‌توان منبع کلاک را بین دو حالت **اتوماتیک** و **دستی** جابجا کرد.