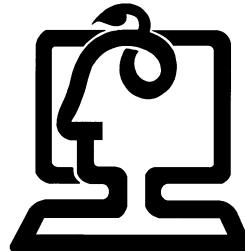




دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

دستور کار آزمایشگاه مدارهای منطقی

تهیه و تنظیم:

گروه مدرسین

۱۳۹۷ مهر

این دستور کار زیر نظر اساتید محترم آقایان دکتر مرتضی صاحب‌الزمانی و دکتر مهدی صدیقی، توسط گروه مدرسین تهیه شده است.

گروه مدرسین (به ترتیب الفبا):

- ۱- وحید امینی
- ۲- جمیله بهزادی
- ۳- جواد تلافی
- ۴- امیر حاجی‌صادقی
- ۵- سعید سیدفر جی
- ۶- رضا فانی
- ۷- عظیم فرقدان
- ۸- هانیه قاسمی
- ۹- پویا محمودی
- ۱۰- مرضیه هاشمی‌پور

• این دستور کار در شهریور ۱۳۹۷ زیر نظر اساتید مورد بازبینی قرار گرفته است.

فهرست

۵	قوانين آزمایشگاه مدارهای منطقی
۷	نکات مهم
۱۰	۱- آشنایی با ابزارهای آزمایشگاه
۳۰	۲- آشنایی با گیت‌های منطقی پایه
۳۵	۳- پیاده‌سازی توابع منطقی با استفاده از جدول کارنو
۳۷	۴- آشنایی با زبان توصیف سخت‌افزار (HDL)
۳۹	۵- آشنایی با مالتی‌پلکسر، انکدر و دیکدر
۴۰	۶- تبدیل کدهای باینری و گری به یکدیگر
۴۲	۷- پیاده‌سازی مدار جمع‌کننده-تفریق‌کننده چهار بیتی به صورت ساختاری
۴۴	۸- پیاده‌سازی واحد محاسبه و منطق
۴۶	۹- تحلیل و پیاده‌سازی مدارات ترتیبی
۴۹	۱۰- پیاده‌سازی ماشین حالت
۵۰	پیوست ۱: راهنمای پایه‌های تراشه‌ها
۵۱	پیوست ۲: روش نوشتتن Testbench
۵۳	پیوست ۳: مراحل سنتز و بارگذاری کد توصیف سخت‌افزاری روی FPGA
۶۵	پیوست ۴: کار با نمایشگر هفت قسمتی

فهرست شکل‌ها

شکل ۱-۱ نمونه‌هایی از مالتی‌مترهای مرسوم	۱۰
شکل ۲-۱ تنظیم سلکتور روی اندازه‌گیری ولتاژ مستقیم با مقدار حداقل ۲۰ ولت	۱۱
شکل ۳-۱ تنظیم سلکتور روی اندازه‌گیری ولتاژ مستقیم برای برخی مالتی‌مترها	۱۱
شکل ۴-۱ کابل‌های اتصال یک سر فیشی یک سر سوزنی	۱۲
شکل ۵-۱ تنظیم سلکتور روی تست اتصال با بوق	۱۳
شکل ۶-۱ یک منبع تغذیه‌ی نوعی	۱۴
شکل ۷-۱ تنظیم ولتاژ کانال‌های ۱ و ۲ روی ولتاژهای مورد نظر	۱۴
شکل ۸-۱ کابل‌های دو سر سوسماری	۱۵
شکل ۹-۱ بردبورد	۱۶
شکل ۱۰-۱ انواع دیگری از بردبورد	۱۷
شکل ۱۱-۱ نمونه‌هایی از سوراخ‌های به هم متصل در بردبورد	۱۸
شکل ۱۲-۱ کل اتصالات یک بردبورد ۳۰ ستونی	۱۸
شکل ۱۳-۱ نحوه‌ی ایجاد اتصالات بردبورد	۱۹
شکل ۱۴-۱ اتصال دهنده‌های ستون‌های ۵ تایی سوراخ‌ها	۲۰
شکل ۱۵-۱ نحوه‌ی قرار گرفتن یک پایه از قطعات در بین دو نوار با خاصیت فرنی	۲۰
شکل ۱۶-۱ نمونه‌هایی از قرار دادن قطعات در بردبورد	۲۱
شکل ۱۷-۱ نمونه‌ای از قرار دادن نادرست قطعات روی بردبورد	۲۱
شکل ۱۸-۱ اتصال ناخواسته‌ی پایه‌ی قطعات به هم	۲۲
شکل ۱۹-۱ اتصال تعذیه برای قطعات بردبورد	۲۲
شکل ۲۰-۱ اتصال ریل‌های بالا و پایین بردبورد به هم برای داشتن تغذیه‌ها از هر دو طرف آن	۲۳
شکل ۲۱-۱ راهی دیگر برای اتصال ریل‌های بالا و پایین تغذیه	۲۳
شکل ۲۲-۱ بردبوردهایی که ریل‌های تغذیه‌ی از وسط منقطع دارند	۲۴
شکل ۲۳-۱ دو نوع بردبورد با ریل‌های منقطع (بالا) و پیوسته (پایین)	۲۴
شکل ۲۴-۱ ایجاد اتصال در وسط ریل‌های تغذیه‌ی بردبوردهایی که ریل‌هایشان از وسط قطع می‌باشند	۲۵
شکل ۲۵-۱ سیم‌های آماده برای استفاده به منظور ایجاد اتصال بین قطعات	۲۵
شکل ۲۶-۱ مدار اتصال LED به منبع تغذیه	۲۶
شکل ۲۷-۱ دو روش اتصال مدار LED در بردبورد	۲۶
شکل ۲۸-۱ یک عکس واقعی از بستن مدارهای LED (به دو روش)	۲۶
شکل ۲۹-۱ تشخیص پایه‌های LED از هم	۲۷
شکل ۳۰-۱ کدهای مقاومتی	۲۸
شکل ۱-۲ نمونه‌هایی از تراشه‌های DIP	۳۰

شکل ۲-۲	نحوه اتصال تراشهای DIP به بردبورد	۳۱
شکل ۳-۲	بریدگی نیم دایرهای تراشهها	۳۱
شکل ۴-۲	علامت فرورفتگی بالای پایه‌ی یک	۳۲
شکل ۵-۲	اتصال LED به خروجی تراشهها	۳۲
شکل ۶-۲	یک نمونه مدار	۳۴
شکل ۱-۴	مدار ترکیبی	۳۷
شکل ۱-۶	نحوه اتصال ماظولهای	۴۰
شکل ۲-۶	مدارهای ترکیبی دارای تأخیر	۴۱
شکل ۱-۷	مدار جمع کننده-تفریق کننده تک بیتی	۴۲
شکل ۲-۷	مدار جمع کننده-تفریق کننده چهار بیتی	۴۳
شکل ۱-۸	بلوک دیاگرام یک واحد محاسبه و منطق	۴۴
شکل ۱-۱۳	صفحه شروع کار با ابزار ISE	۵۳
شکل ۲-۱۳	ایجاد یک پروژه جدید	۵۳
شکل ۳-۱۳	تعیین ویژگی‌های بورد FPGA	۵۴
شکل ۴-۱۳	ایجاد یک فایل در پروژه	۵۴
شکل ۵-۱۳	تعیین نوع و نام فایل	۵۵
شکل ۶-۱۳	تعیین پورت‌های ورودی و خروجی	۵۵
شکل ۷-۱۳	منوی شبیه‌سازی با ISim	۵۶
شکل ۸-۱۳	اعمال مقدار ثابت به سیگنال‌ها (هنگامی که testbench نداریم)	۵۶
شکل ۹-۱۳	نتیجه شبیه‌سازی	۵۷
شکل ۱۰-۱۳	منوی سنتز	۵۷
شکل ۱۱-۱۳	نتیجه سنتز	۵۸
شکل ۱۲-۱۳	اجرای برنامه تخصیص پایه‌های ورودی و خروجی	۵۸
شکل ۱۳-۱۳	تایید اجرای برنامه تخصیص پایه‌های ورودی و خروجی	۵۹
شکل ۱۴-۱۳	منوی تخصیص پایه‌های ورودی و خروجی	۵۹
شکل ۱۵-۱۳	تخصیص پایه‌های ورودی و خروجی	۶۰
شکل ۱۶-۱۳	ایجاد فایل برنامه‌ریزی	۶۰
شکل ۱۷-۱۳	ایجاد یک پروژه برنامه‌ریزی بورد FPGA	۶۱
شکل ۱۸-۱۳	تایید نوع برنامه‌ریزی	۶۱
شکل ۱۹-۱۳	انصراف از برنامه‌ریزی بخش حافظه بورد FPGA	۶۲
شکل ۲۰-۱۳	افزودن فایل برنامه‌ریزی مناسب	۶۲
شکل ۲۱-۱۳	انتخاب فایل مناسب	۶۳
شکل ۲۲-۱۳	برنامه ریزی FPGA	۶۳

شکل ۲۳-۱۳ تائید نهایی ۶۴
شکل ۲۴-۱۳ وضعیت برنامه ریزی ۶۴

قوانين آزمایشگاه مدارهای منطقی

به منظور افزایش کارآیی درس آزمایشگاه مدارهای منطقی، رعایت عدالت بین تمامی گروههای آزمایشگاه و آموزش حداکثری مطالب درس به صورت عملی، مدرسین و دانشجویان ملزم به رعایت نکات و قوانین ذیل هستند:

۱. تعداد جلسات در طول ترم حداقل ۱۲ جلسه خواهد بود (۱۰ جلسه آزمایش به همراه حداقل ۲ جلسه پروره).

۲. مدرسین و دانشجویان موظفند رأس ساعت در کلاس حاضر شوند.

۳. پس از گذشت پنج دقیقه از شروع کلاس، به ازای هر پنج دقیقه تأخیر ۱۰ درصد نمره آن جلسه کسر می‌شود.

۴. حداکثر میزان تاخیر ۳۰ دقیقه است.

۵. دانشجویان می‌توانند در طول ترم حداقل یک جلسه غیبت داشته باشند. در صورت غیبت بیش از یک جلسه مطابق قانون، درس و آزمایشگاه آن‌ها حذف غیبت خواهد شد.

۶. دانشجویان در صورت غیبت نمی‌توانند در کلاس دیگری شرکت کنند. جهت جبران نمره آزمایشی که دانشجو در آن جلسه غایب بوده است، دانشجو می‌تواند در ساعات خالی آزمایشگاه، آزمایش را انجام دهد. بدیهی است این امر غیبت را حذف نمی‌کند و در صورت غیبت مجدد درس و آزمایشگاه دانشجو حذف غیبت خواهد شد.

۷. در صورت غیبت و عدم جبران آن، نمره آن جلسه صفر منظور خواهد شد.

۸. اگر دانشجویانی نتوانند در طی یک جلسه آزمایشی را تمام کنند، در طول هفته موظفند تا در ساعات خالی آزمایشگاه، آزمایش را تکمیل کنند و پیش از شروع جلسه بعد نتایج را به مدرس خود تحويل دهند. حداکثر تعداد جلساتی که به درازا کشیده می‌شود ۳ جلسه در طول نیمسال است. دانشجو برای تکمیل آزمایش نمی‌تواند به گروههای دیگر ملحق شود.

۹. آزمایش‌ها در گروههای دو نفره انجام می‌شوند و میزان فعالیت اعضای گروه تعیین کننده نمره آن‌ها خواهد بود (در صورت عدم فعالیت یکی از اعضای گروه آن فرد نمره‌ای دریافت نخواهد کرد).

۱۰. هر آزمایش شامل یک پیش‌گزارش است. این پیش‌گزارش باید به صورت دست نویس و پیش از شروع آزمایش‌ها به مدرس تحويل داده شود. پیش‌گزارش مطلوب هر آزمایش در دستور کار آمده است.

۱۱. ارائه گزارش کار آزمایش‌ها الزامی نیست و تحويل خروجی صحیح به مدرس کفايت می‌کند.

۱۲. جهت کسب نمره قبولی در آزمایشگاه، کسب حداقل نمره قبولی در درس الزامی است.

۱۳. جهت حفظ حرمت کلاس و نظافت آزمایشگاه، خواهشمند است از خوردن و آشامیدن در طول کلاس خودداری نمایید.

۱۴. وارد آوردن هرگونه خسارت و لطمہ به تجهیزات آزمایشگاه مستلزم جبران خسارت است.

۱۵. بارمبندي تقريري نمره آزمایشگاه (که به اساتيد درس گزارش خواهد شد) به صورت زير خواهد بود:
- الف) پيشگزارشها:٪.۱۰
 - ب) انجام آزمایشها:٪.۵۵
 - پ) پروژه:٪.۳۵

نکات مهم

پیش از شروع آزمایش‌ها ضروری است تا دانشجویان گرامی این بخش را مطالعه فرمایند و در طی جلسات آزمایشگاه این نکات را رعایت نمایند.

۱. قبل از انجام هر آزمایش، مبحث تئوری مربوط به آن آزمایش به طور کامل مطالعه شود، چرا که در حین جلسه، وقت کافی برای توضیح و یادگیری قسمت تئوری وجود ندارد.
۲. در آزمایش‌های آخر، لازم است موارد ذکر شده در آزمایش‌های قبلی و ابتدایی را به خاطر داشته باشید. پس اگر آزمایشی را انجام می‌دهید، حتماً علاوه بر گزارش کاری که به مرتب خود تحويل می‌دهید، برای خود نیز یادداشت بردارید تا بعداً به مشکل بزنخوردید.
۳. هرگاه جایی در آزمایش‌ها و نقشه‌های مدارها، اتصال خروجی تراشه‌ای مشخص نشده باشد، آن را آزاد رها کنید (به جایی وصل نکنید) ولی هیچ‌گاه ورودی یک تراشه را نمی‌توان به جایی وصل نکرد چون این کار باعث تاثیرپذیری زیاد مدار نسبت به نویز می‌شود.
۴. هیچ‌گاه خروجی یک تراشه را به خروجی تراشه‌ی دیگر یا V_{CC} و GND وصل نکنید. این کار به تراشه صدمه می‌رساند (با این توضیحات متوجه می‌شویم که مهم است بدانیم کدامیک از پایه‌ها مربوط به خروجی تراشه و کدامیک مربوط به ورودی آن می‌شوند. این امر مستلزم تسلط روی عملکرد مدار است. در نتیجه باز هم اهمیت مطالعه‌ی قسمت تئوری جلوه‌گر می‌شود).
۵. هیچ‌گاه دیودهای نورانی (یا اصولاً هر قطعه‌ای که در آن LED به کار رفته مثل هفت قسمتی‌ها و ماتریس‌های LED) را مستقیماً به خروجی تراشه یا منبع تغذیه وصل نکنید، بلکه آن را با یک مقاومت بین ۱۰۰ تا ۳۳۰ اهمی سری نموده، و سپس وصل کنید.
۶. تراشه‌های TTL، نیاز به منبع تغذیه ۵ ولتی دارند ولی تراشه‌های CMOS می‌توانند با منبع تغذیه‌های ۳ تا ۱۵ ولتی کار کنند.
۷. قسمت مهم کار در این آزمایشگاه عیب‌یابی مدارها است. صرفاً وصل کردن مدار چندان مهم نیست؛ بلکه مهم، عیب‌یابی آن است. معمولاً مداری که وصل می‌کنید، دارای عیب‌هایی است و کمتر مداری بدون اشتباه وصل می‌شود. پس باید یاد بگیرید که بتوانید هر مداری را (حتی مداری که خودتان در بستن آن نقشی نداشته‌اید)، عیب‌یابی کنید. برای این کار لازم است هر آزمایشی را که انجام می‌دهید و با هر تراشه‌ای که آشنا می‌شوید، طرز کار آن را دقیقاً یاد بگیرید تا وقتی که در مدارهای بزرگتر، از این تراشه‌ها در کنار هم استفاده می‌شود، بتوانید با بی‌گیری ولتاژهای خروجی‌های قطعات مدار، به مشکل کار پی ببرید.
۸. دقت کنید باز کردن و دوباره بستن مداری که درست کار نمی‌کند، راه حل درستی نیست. سعی کنید خودتان مشکل کار را پیدا کنید.
۹. در آزمایش‌های اول با سطوح ولتاژ آشنا می‌شوید. سعی کنید که محدوده‌ی ولتاژهای '۱' منطقی و '۰' منطقی درست را به خاطر بسپارید تا در عیب‌یابی مدارها وقتی به سطح ولتاژی غیر معمول برخوردید،

- بدانید که در آنجا ممکن است مشکلی وجود داشته باشد. وقتی می‌خواهید ولتاژ خروجی تراشه‌ای را بخوانید، اگر خروجی آن به LED متصل است، اول LED را جدا کنید و بعد ولتاژ را بخوانید.
۱۰. تا مطمئن نشیدید که یک تراشه خراب است، اقدام به تعویض آن نکنید. حتماً اول به نحوی مطمئن شوید که تراشه مشکل دارد. بهترین راه برای اطمینان از خراب بودن تراشه، تست کردن جداگانه‌ی آن روی یک بردبورد سالم است. یادتان نرود که برای اطمینان بیشتر، ولتاژها را از روی پایه‌ی فلزی تراشه بخوانید.
۱۱. تراشه‌هایی که فکر می‌کنید چهار مشکل هستند را پس از اطمینان از خراب بودن به جعبه‌ی خود در آزمایشگاه بر نگردانید تا بعداً همین تراشه‌های معیوب باعث اتلاف وقت دیگران یا خودتان نشوند.
۱۲. سعی کنید ولتاژهایی که با مالتی‌متر می‌خوانید، از روی پایه‌های فلزی تراشه‌ها باشد (یعنی ترمینال فلزی مالتی‌متر را با استفاده از یک تکه سیم که محکم به آن وصل کرده‌اید دقیقاً به پایه‌ای که می‌خواهید مقدارش را بخوانید تماس دهید نه به سوراخ‌های بردبورد)؛ چون احتمال خرابی بردبورد یا قطع بودن سیم‌ها وجود دارد. حتماً می‌دانید که برای خواندن مقدار ولتاژ جایی توسط مالتی‌متر، یک سر (ترمینال به رنگ مشکی) مالتی‌متر را باید به زمین وصل کرد و ترمینال دیگر آن (به رنگ قرمز) را به جایی که می‌خواهید ولتاژ آن را اندازه بگیرید. پس برای راحتی کار می‌توانید در ابتدای آزمایش یک ترمینال مالتی‌متر را به‌طور ثابت به زمین (GND) مدار وصل کنید.
۱۳. همیشه قبل از شروع کار، با مالتی‌متر چک کنید که اصولاً ولتاژ تغذیه به مدار و تراشه‌ها می‌رسد یا خیر. با بررسی کردن پایه‌های مربوط به تغذیه‌ی تراشه‌ها می‌توانید قطعی‌های احتمالی سیم‌ها یا خرابی بردبورد (در راه رساندن ولتاژ تغذیه) را چک کنید. گاهی اوقات مداری که می‌بندید دارای اشکالی است که روی ولتاژ تغذیه اثر می‌گذارد و آن را کم می‌کند و حتی گاهی این ولتاژ را به حدود صفر می‌رساند. در این صورت باید در مدار دنبال جایی بگردید که احتمالاً خط V_{CC} ناخواسته به زمین (GND) وصل شده باشد. اگر مدار پیچیده است و پیدا کردن چنین موقعیتی مشکل است کافیست تک تک سیم‌های مدار (از سیم‌های تغذیه شروع کنید) را از جای خود جدا کنید و بعد به جای خود برگردانید تا اینکه به سیمی بررسید که با جدا کردن آن، ولتاژ تغذیه به مقدار اصلی خود برگردد. با این کار متوجه می‌شوید که این سیم مشکل را بوجود آورده است و باید چک کنید که اشکال کار در وصل کردن این سیم چه بوده است.
۱۴. همیشه فقط دو سیم از منبع تغذیه به بردبوردی که مدار را روی آن می‌بندید بکشید و در مدار هر جایی نیاز به منبع تغذیه داشتید از ریل‌هایی که این سیم‌ها به آن‌ها متصل شده استفاده کنید (دقت کنید که در برخی بردبوردها چهار ریل افقی بردبورد از وسط منقطع می‌باشند و در صورت نیاز باید آن‌ها را با سیم به هم متصل کرد). کشیدن تعداد زیادی سیم از منبع تغذیه، مدار را شلوغ و کار را مشکل می‌کند.
۱۵. سیم (یا مقاومت، خازن یا مواردی شبیه این‌ها) را از روی تراشه‌ها رد نکنید. چون در این صورت اگر آن تراشه خراب باشد جایه‌جایی آن مشکل خواهد بود و همچنین رد کردن قطعات با پایه‌های لخت از روی تراشه‌ها یا از روی هم احتمال اتصال ناخواسته آن‌ها را به هم و به تراشه افزایش می‌دهد.

۱۶. هیچ وقت مطمئن نباشید مداری که بسته‌اید عاری از خطاست و دلیل کار نکردن آن خرابی قطعات است. اشتباه ممکن است همه جا و توسط هر کسی (حتی ماهرترین افراد) بوجود آید. پس اشکالی ندارد که در مداری که بسته‌اید، اشتباه وجود داشته باشد؛ بلکه مهم این است که بتوانید از مدار جواب بگیرید. باز هم تذکر داده می‌شود که بستن مدار به تنها یک کافی نیست، باید بتوانید مداری که بسته‌اید (توسط خودتان یا دیگران) را به بهترین نحو و در کوتاه‌ترین زمان عیب‌یابی کنید و این کار جز با صبر و حوصله و همچنین تمرین و ممارست زیاد ممکن نمی‌شود. سعی کنید در این آزمایشگاه، عیب‌یابی را به بهترین نحو یاد بگیرید.

۱۷. مدار پیچیده‌ای که خود شامل قطعات مختلف است را می‌توان این گونه اشکال‌زدایی کرد که بعد از دادن ورودی‌ها و دیدن خروجی‌های نهایی، اگر این خروجی‌ها، با خروجی‌هایی که مورد نظر ماست، متفاوت باشند می‌توان یک مرحله به عقب برگشت (مثلاً اگر خروجی‌ها از تراشه‌ای گرفته می‌شوند به ورودی‌های آن تراشه رجوع کرد) و آن خطوط را چک کرد، اگر این خطوط مقادیر درست موردنظر را داشتند، متوجه می‌شویم اشکال در تراشه‌ی مذکور وجود دارد و باید آن تراشه یا قطعات متصل شده به آن با دقت بیشتری چک شوند تا دقیقاً به عامل خطا رسید. ولی اگر این خطوط هم نادرست باشند، باز هم باید به یک مرحله عقب‌تر برگشت و این کار را آنقدر تکرار کرد تا به قطعه یا قطعات یا مسیرها یا به طور کلی هر مورد (یا مواردی) که مشکل را ایجاد کرده‌اند رسید. به این نکته هم توجه داشته باشید که ممکن است مداری به ازاء یک یا چند ورودی خاص جواب بدهد ولی به ازاء یک یا چند ورودی دیگر جواب ندهد. بدیهی است که این مدار را نمی‌توان مدار سالمی فرض کرد. همچنین ممکن است به مشکلی از مدار پی ببرید و آن را مرفوع کنید ولی کماکان خروجی‌های مدار، خروجی‌های مورد نظر نباشند. از اینجا نتیجه می‌گیریم که باز هم در مدار مشکل وجود دارد که باید دوباره کارهای فوق‌الذکر را تکرار کرد تا مشکل(های) دیگر هم برطرف شوند.

۱- آشنایی با ابزارهای آزمایشگاه

در اینجا به ابزارهای اساسی مربوط به آزمایشگاه مدارهای منطقی که زیاد با آن‌ها سروکار خواهد داشت اشاره می‌شود. در صورتی که در حین انجام برخی آزمایش‌ها به صورت استثنای ابزار خاص دیگری نیاز داشته باشد، توضیحات تکمیلی در آزمایشگاه توسط مدرس داده خواهد شد.

(الف) مالتی‌متر

مالتی‌مترهای مرسوم می‌توانند سه عمل اصلی اندازه‌گیری ولتاژ، اندازه‌گیری جریان و اندازه‌گیری مقاومت را انجام دهند. در ضمن بعضی از مالتی‌مترها توانایی‌های دیگری از جمله اندازه‌گیری خازن، فرکانس، بتای ترانزیستور، و تست دیود را هم دارند.

در شکل ۱-۱ نمای ظاهری نمونه‌هایی از مالتی‌مترها را مشاهده می‌کنید. اگرچه ممکن است مالتی‌مترها شکل‌های متفاوتی داشته باشند، ولی با دانستن طرز کار یکی از آن‌ها می‌توان تا حد زیادی به نحوه کار با دیگر مالتی‌مترها پی‌برد.



شکل ۱-۱ نمونه‌هایی از مالتی‌مترهای مرسوم

همان طور که از شکل ۱-۱ پیداست قسمت اصلی مالتی متر را یک کلید سلکتور چرخان و یک صفحه نمایش تشکیل می‌دهد. در آزمایشگاه مدارهای منطقی بیشتر با قسمت اندازه‌گیری ولتاژ مستقیم مالتی متر و گاهی هم خواندن مقدار مقاومت و تست اتصال (با بوق) سروکار داریم.

برای سنجش ولتاژ مستقیم دکمه سلکتور چرخان مالتی متر را روی ۲۰V که در شکل ۲-۱ مشخص شده می‌گذاریم. در این حالت ولتاژ ۲۰ ولت حداکثر ولتاژی است که مالتی متر می‌تواند اندازه بگیرد (اصولاً در هر وضعیت، ولتاژ، جریان، مقاومت و بقیه موارد، عددی که سلکتور نشان می‌دهد، نشان‌دهنده‌ی بیشترین مقداری است که مالتی متر در آن وضعیت می‌تواند اندازه بگیرد). اینکه چرا عدد ۲۰ ولت انتخاب شده، به این دلیل است که در آزمایشگاه مدارهای منطقی با ولتاژهای حدود ۵ ولت سروکار داریم و اگر ولتاژ کمتر از ۲۰ را روی سلکتور ملاحظه کنید می‌بینید که ۲ ولت است و برای اندازه‌گیری ۵ ولت مناسب نیست. ولتاژهای بیشتر از ۲۰ ولت هم به خاطر اینکه در آن‌ها دقیقت کار پایین می‌آید معمولاً برای کار در این آزمایشگاه انتخاب نمی‌شوند.



شکل ۲-۱ تنظیم سلکتور روی اندازه‌گیری ولتاژ مستقیم با مقدار حداکثر ۲۰ ولت

البته برای نوع دیگر مالتی متر که در شکل ۱-۱ نشان داده شده است، فقط کافیست که سلکتور را روی علامت مشخص شده در شکل ۳-۱ قرار دهیم و خود مالتی متر عمل تنظیم دقیقت و حداکثر ولتاژ را به صورت خودکار انجام می‌دهد. در برخی مالتی مترهای دیگر، کلیدی به نام RANGE (یا نامی مشابه) وجود دارد که با کمک آن می‌توان دقیقت و حداکثر ولتاژ را تنظیم کرد.



شکل ۳-۱ تنظیم سلکتور روی اندازه‌گیری ولتاژ مستقیم برای برخی مالتی مترها

سپس باید کابل‌های اتصال یک سر فیشی یک سر سوزنی (شکل ۴-۱) که عموماً برای راحتی کار به دو رنگ قرمز و مشکی می‌باشند (ولی از نظر عملکرد با هم فرقی ندارند) را به ترتیب به دو سوکت $V\Omega Hz$ و COM متصل کنیم. طرف دیگر آن‌ها را باید به جایی که می‌خواهیم اختلاف پتانسیل دو سرش را اندازه بگیریم، متصل کنیم. معمولاً بیشتر اوقات نیاز به خواندن ولتاژ نقاطی از مدار داریم. برای این منظور، ترمینال قرمز (سوکت $V\Omega Hz$) را به نقطه‌ی مورد نظر از مدار وصل می‌کنیم. همچنین ترمینال مشکی (متصل شده به سوکت COM مالتی‌متر) را نیز به زمین مدار (قطب منفی منبع تغذیه) متصل می‌کنیم. در ضمن اگر جای ترمینال‌ها را جایه‌جا متصل کنید، اشکالی رخ نمی‌دهد و فقط منفی ولتاژ مورد نظر را به ما نشان می‌دهد. دقت نمایید که مالتی‌متر در حالت HOLD قرار نداشته باشد چرا که در اینصورت آخرین مقدار قبل از Hold کردن را نشان می‌دهد و تغییرات را دنبال نمی‌کند. حالت HOLD در صفحه نمایش نشان داده می‌شود و با دکمه HOLD (فشاری یا قطعی/وصلی) می‌توان آن را انتخاب یا حذف کرد.

توجه داشته باشید که باید یک تکه سیم کوتاه به سرهایی که قرار است به مدار متصل شوند وصل کنید (یک سر آن را بیشتر لخت کنید و دور فلز سوزنی سر کابل بپیچید تا محکم شود) چون سرهای سوزنی به راحتی داخل سوراخ‌های بردبورد نمی‌روند و محکم نمی‌ایستند و در صورت فشار دادن بیش از حد به سوراخ بردبورد صدمه وارد می‌کنند.



شکل ۴-۱ کابل‌های اتصال یک سر فیشی یک سر سوزنی

حالت دیگری از مالتی‌متر که گاهی به آن نیاز پیدا می‌کنیم خواندن مقدار مقاومت است. در این حالت کلید سلکتور چرخان را روی یکی از حالت‌های مقاومت (Ω) قرار می‌دهیم و ترمینال‌ها هم به همان دو سوکت

COM و $V\Omega Hz$ وصل می‌شوند. در این حالت مقداری که در نمایشگر مشاهده می‌کنید، مقدار مقاومت است. توجه داشته باشید که نباید همزمان به دو سر کابل‌های مالتی‌متر (قسمت فلزی آن‌ها) دست بزنید چرا که در این حالت مقاومت بدن شما با مقاومت مورد اندازه‌گیری سری یا موازی شده و عدد خوانده شده بسته به مورد ممکن است با واقعیت تفاوت زیادی پیدا کند.

حالت دیگر مورد نیاز، تست اتصال است که همانطور که در شکل ۵-۱ نشان داده شده سلکتور مالتی‌متر را روی علامت مشخص شده قرار می‌دهیم و ترمینال‌ها را به همان دو سوکت $V\Omega Hz$ و COM وصل می‌کنیم. در این صورت اگر دو سر ترمینال به دو نقطه وصل شوند که توسط یک رسانا (یا با یک مقاومت کم) به هم متصل هستند، دستگاه بوق می‌زند. این حالت برای جاهایی که با چشم نمی‌توان اتصال بین آن‌ها را چک کرد مثلًا سیم‌های دارای روکش یا بین سوراخ‌های بردبورد مناسب است.



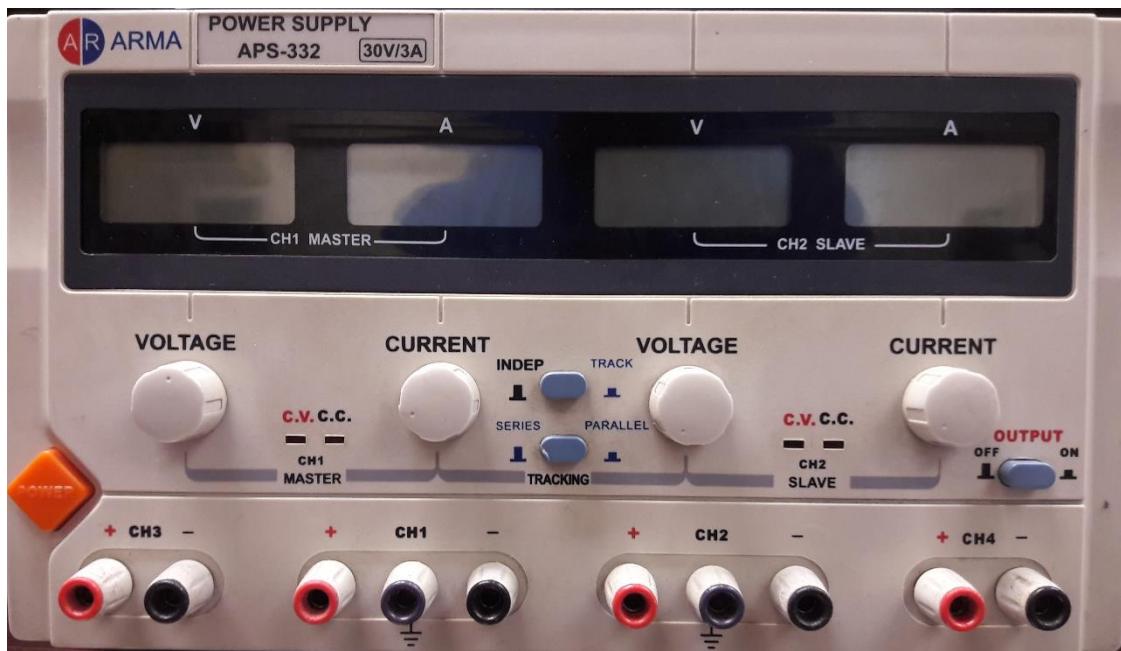
شکل ۵-۱ تنظیم سلکتور روی تست اتصال با بوق

برای اندازه‌گیری جریان باید سلکتور را روی گزینه مناسب A قرار داد و ترمینال‌های قرمز و مشکی را هم به ترتیب به mA و COM متصل کرد (جابجایی آن‌ها با هم فقط منفی جریان را نشان می‌دهد). دو ترمینال را باید به صورت سری در مسیر جریانی که قرار است اندازه‌گیری شود، قرار داد.

(ب) منبع تغذیه

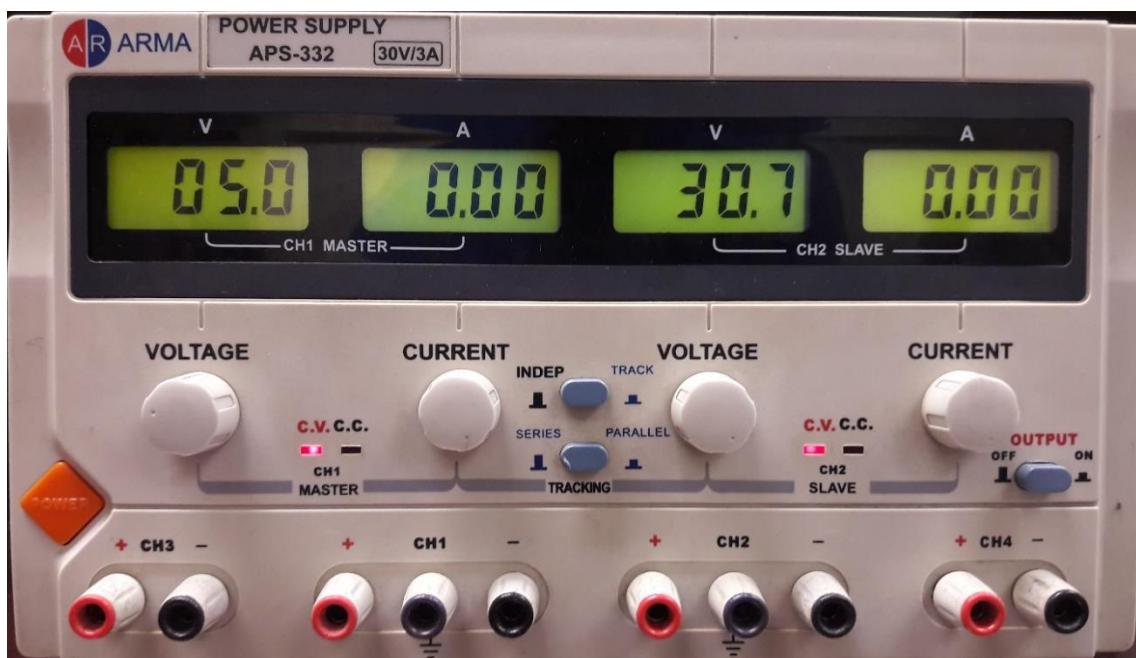
در شکل ۶-۱ نمای ظاهری یک نمونه از منابع تغذیه را مشاهده می‌کنید. ممکن است منابع تغذیه ظاهرهای مختلفی داشته باشند ولی با دانستن طرز کار یکی، می‌توان از نوع‌های دیگر منابع ولتاژ نیز استفاده کرد.

این منبع تغذیه دارای ۴ خروجی متمایز است که با CH1, CH2, CH3 و CH4 نشان داده شده‌اند. دو خروجی CH1 و CH2 (که سوکت‌هایشان در وسط منبع تغذیه قرار دارد) قابل تنظیم هستند و با ولوم‌های پیچی VOLTAGE مربوطه می‌توان ولتاژشان را تنظیم کرد و همزمان در صفحات نمایش (V) CH1 MASTER (V) و CH2 SLAVE (V) مقدارشان را مشاهده کرد (بهتر است زیاد به مقداری که می‌بینید اتنک نکنید و خود خروجی کanal را با مالتی‌متر به صورت دقیق‌تر اندازه بگیرید).



شکل ۶-۱ یک منبع تغذیه‌ی نوعی

در شکل ۷-۱، نشان داده شده که کanal ۱ روی ولتاژ ۵ ولت و کanal ۲ روی ولتاژ $\frac{30}{7}$ ولت تنظیم شده است. خروجی‌های CH3 و CH4 به ترتیب دارای ولتاژهای ثابت و غیر قابل تغییر ۵ ولت و ۱۲ ولت می‌باشند. به خاطر داشته برای آنکه خروجی‌های منبع تغذیه فعال باشند، باید کلید OUTPUT که در سمت راست قرار دارد فشرده شده باشد (حالت ON) در غیر این صورت ولتاژ تمامی خروجی‌ها صفر خواهد بود (این کلید برای نرساندن موقت ولتاژ به مداری که به منبع وصل کرده‌اید و معمولاً می‌خواهید مشکلاتش را برطرف نمایید مناسب است).



شکل ۷-۱ تنظیم ولتاژ کanal‌های ۱ و ۲ روی ولتاژهای مورد نظر

توجه داشته باشید که کلید INDEP/TRACK که در وسط منبع قرار دارد بالا باشد (فشرده نشده باشد «حالت INDEP») چون در غیر این صورت (حالت TRACK) دو کanal ۱ و ۲ با هم موازی (در صورتی که کلید پایینی آن که کلید SERIES/PARALLEL است فشرده شده باشد «حالت PARALLEL») یا سری (در صورتی که فشرده نشده باشد «حالت SERIES») خواهد شد که به ترتیب برای داشتن جریان بیشتر (حاصل جمع جریان‌های دو کanal) و ولتاژ بیشتر (حاصل جمع ولتاژ‌های دو کanal) به کار می‌آیند. در این آزمایشگاه هم ولتاژ و هم جریان هر یک از کanal‌ها به تنها بی‌برای آزمایش‌ها کفایت کرده و نیازی به این کار نیست. در نتیجه نیازی به فشردن کلید INDEP/TRACK و انجام تنظیمات مربوطه نیست).

برای اتصال هر یک از کanal‌هایی که انتخاب کرده‌اید به مدار، می‌توانید از کابل‌های دو سر سوسناری استفاده کنید که در شکل ۸-۱ نشان داده شده‌اند. معمولاً کابل قرمز را به سوکت + و کابل مشکی را به سوکت - وصل می‌کنیم (رنگ‌ها صرفاً جهت تسریع کار در نظر گرفته می‌شوند و فرقی با هم ندارند). خروجی‌های وسطی کanal‌های ۱ و ۲ که با علامت زمین مشخص شده‌اند به جایی وصل نمی‌شوند. توجه داشته باشید که باید یک تکه سیم کوتاه به سرهایی که قرار است به مدار متصل شوند وصل کنید.



شکل ۸-۱ کابل‌های دو سر سوسناری

دو ولوم CURRENT روی منبع، برای تنظیم حداکثر جریان تحویلی کanal‌های ۱ و ۲ هستند. همچنین برای اینکه به دلیل اتصال اشتباه به قطعات داخل مدار لطمehای وارد نشود، می‌توان از آن‌ها استفاده کرد. در صورتی که حداکثر جریانی که قرار است مداراتان از منبع بکشد را بدانید (در حد چند یا چندین میلی‌آمپر برای آزمایش‌های این آزمایشگاه)، می‌توانید ولوم مذکور را تنظیم کنید که منبع از آن جریان بیشتر تحویل ندهد. برای این منظور روی ولتاژ مورد نظرتان خروجی را اتصال کوتاه کنید (دو سر منفی و مثبت را به هم متصل کنید). اکنون ولوم CURRENT را آنقدر بچرخانید که صفحه نمایش مربوطه (A) CH1 MASTER و (A) CH2 SLAVE مقدار جریانی که مد نظرتان است (کمی بیشتر از حداکثر جریان مدار به خاطر احتیاط) را

نشان دهد. بعد از اینکه خروجی مورد نظر را از حالت اتصال کوتاه در آوردید منبع آماده استفاده است. اگر به هر دلیلی جریانی بیش از مقدار مشخص شده بخواهد کشیده بشود منبع، آن را محدود نموده و چراغ کوچک C.C. مربوطه را روشن می‌کند.

از این به بعد هر جا به علامت «برخوردید» یعنی باید کار خواسته شده را عملأً انجام دهید.

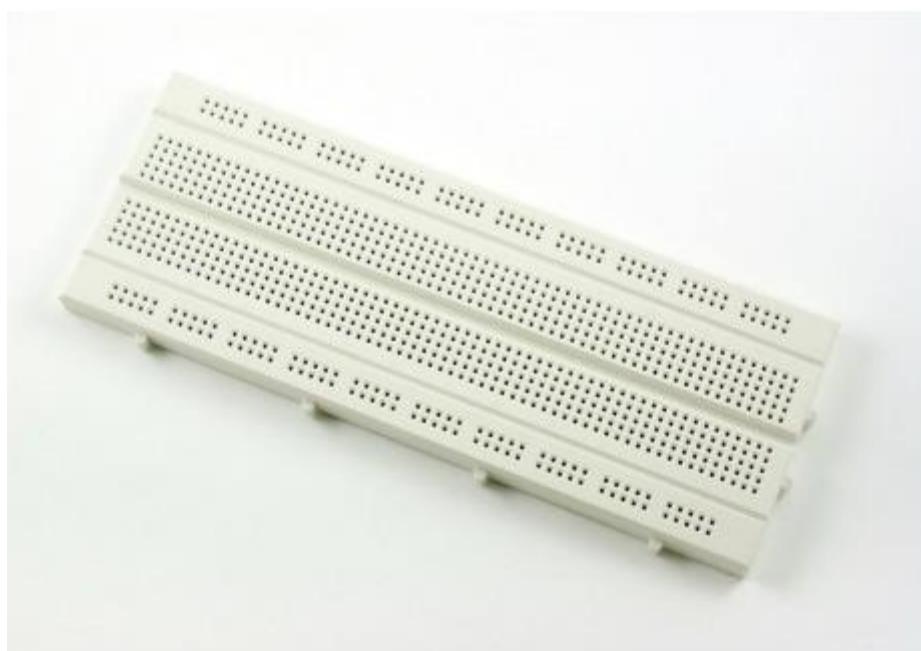
با مالتی‌متر ولتاژ خروجی‌های ثابت (CH3 و CH4) و قابل تنظیم (CH1 و CH2) منبع تغذیه را اندازه بگیرید (گاهی اصطلاحاً می‌گوییم بخوانید).

(ج) بردبورد

برای بستن و امتحان کردن مدارها راههای مختلفی وجود دارد که از بین آن‌ها بهترین گزینه برای کارهای آزمایشگاهی استفاده از بردبورد است.

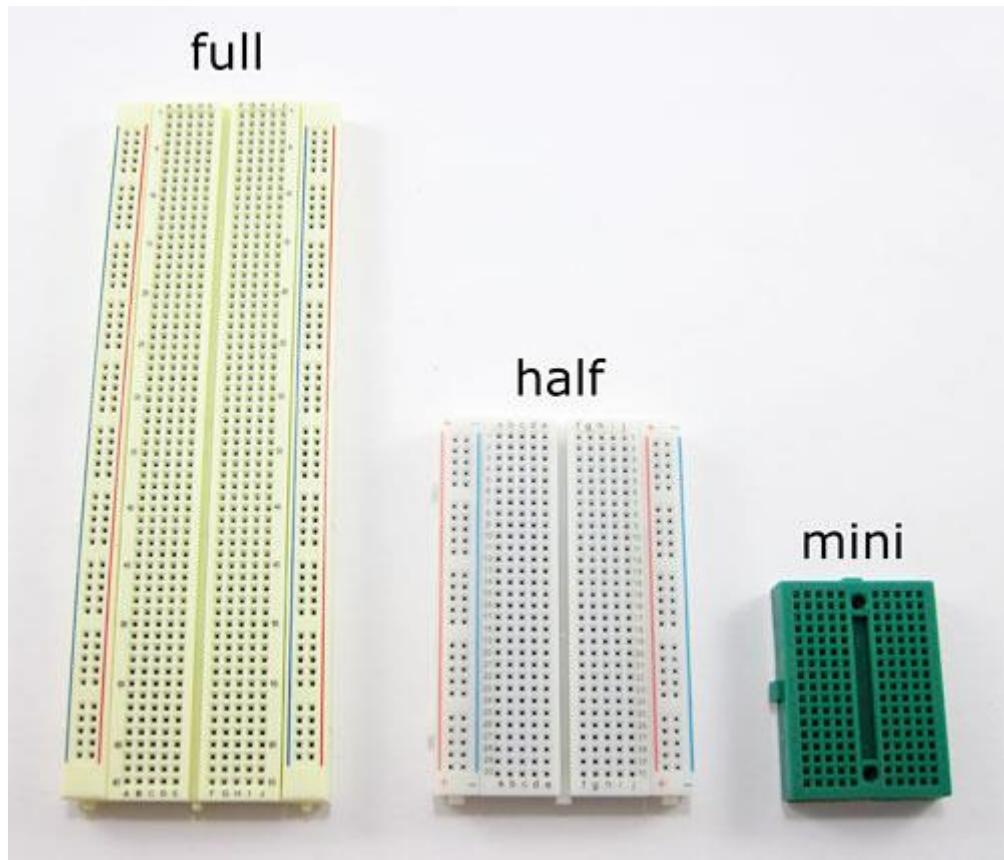
این وسیله بیشتر برای کارهای نمونه‌سازی (Prototyping) استفاده می‌شود که در آن از درستی عملکرد مداری که قرار است بعداً روی بوردهای مدار چاپی (PCB) ساخته شود اطمینان حاصل می‌شود و اگر بنا باشد تغییری در ساختار مدار برای بهبود آن داده شود در این مرحله به راحتی انجام می‌شود.

چون در بردبورد برخلاف برددهای مدار چاپی عمل لحیم‌کاری صورت نمی‌گیرد، این وسیله قابل استفاده‌ی مجدد است. یک نمونه از بردبوردهای مرسوم در آزمایشگاه را در شکل ۹-۱ می‌توانید مشاهده کنید.



شکل ۹-۱ بردبورد

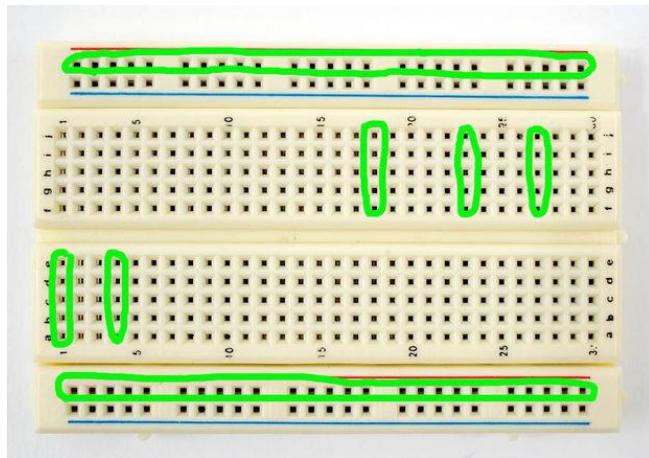
البته نمونه‌های دیگری هم از بردبورد وجود دارند که برخی از آن‌ها را در شکل ۱۰-۱ می‌بینید. اصول استفاده از همه‌ی آن‌ها یکسان است و در ادامه شرح داده خواهد شد.



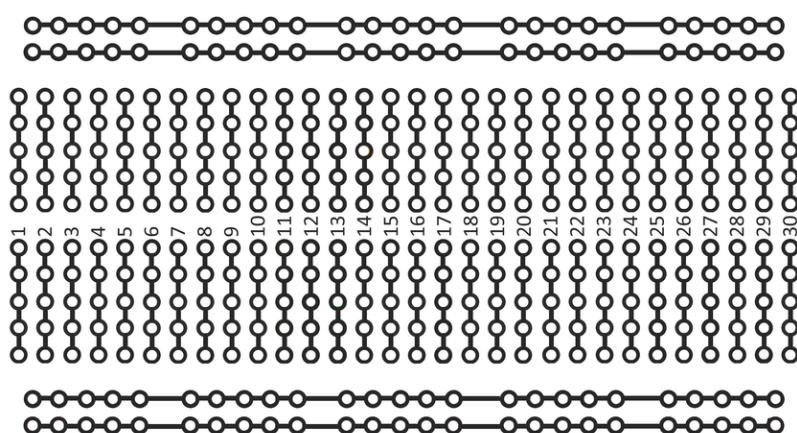
شکل ۱۰-۱ انواع دیگری از بردبورد

همانطور که در شکل‌ها مشاهده می‌کنید، روی بردبوردها تعدادی سوراخ وجود دارد که پایه‌های قطعات الکترونیکی و الکتریکی و همچنین سیم‌هایی که اتصالات آن‌ها را به هم ایجاد می‌کنند در آن‌ها قرار می‌گیرد. این سوراخ‌ها به روش خاصی که در ادامه به آن اشاره می‌شود در داخل بردبورد به هم متصل هستند.

بردبورد شامل یک تعداد (نوعاً ۳۰ و ۶۴ به طور مرسوم) ستون ۵ تایی از سوراخ‌ها است که در امتداد هم در دو ردیف چیده شده‌اند. فاصله‌ی این ستونها از هم به صورت استاندارد است. در شکل ۱۱-۱ اتصال این ستون‌های ۵ تایی به یکدیگر را ملاحظه می‌کنید (این اتصال‌ها در داخل بردبورد انجام شده است). در ضمن چهار ردیف از سوراخ‌ها (موسوم به ریل) در بالا و پایین بردبورد وجود دارند و سوراخ‌های هر کدام از این ردیف‌ها به هم متصل هستند. در شکل ۱۱-۱ دو نمونه از این ردیف‌ها مشخص شده‌اند. دقت داشته باشید که با اینکه در این ردیف‌ها دسته‌های ۵ سوراخی با یک فاصله از هم جدا شده‌اند ولی از نظر الکتریکی جدا نیستند و به هم متصلند. در شکل ۱۲-۱ هم کل اتصالات یک بردبورد ۳۰ ستونی (Half Size) نشان داده شده است. در این شکل خطوط بین دایره‌ها که نمادی از سوراخ‌ها می‌باشند، اتصالات را نشان می‌دهند.

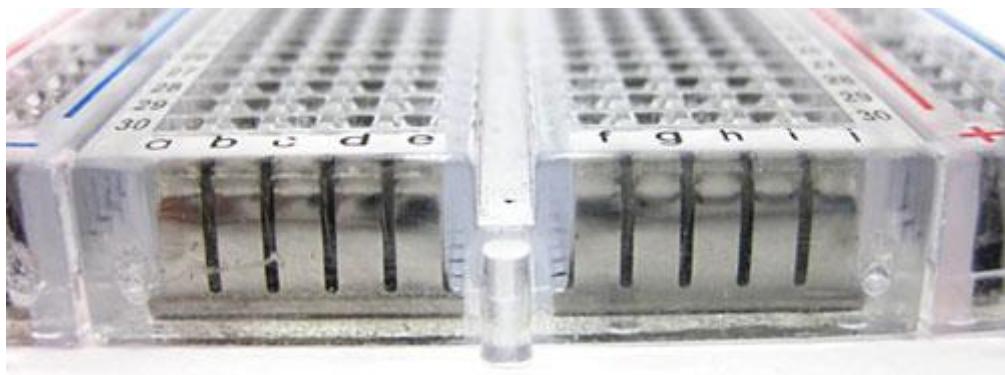
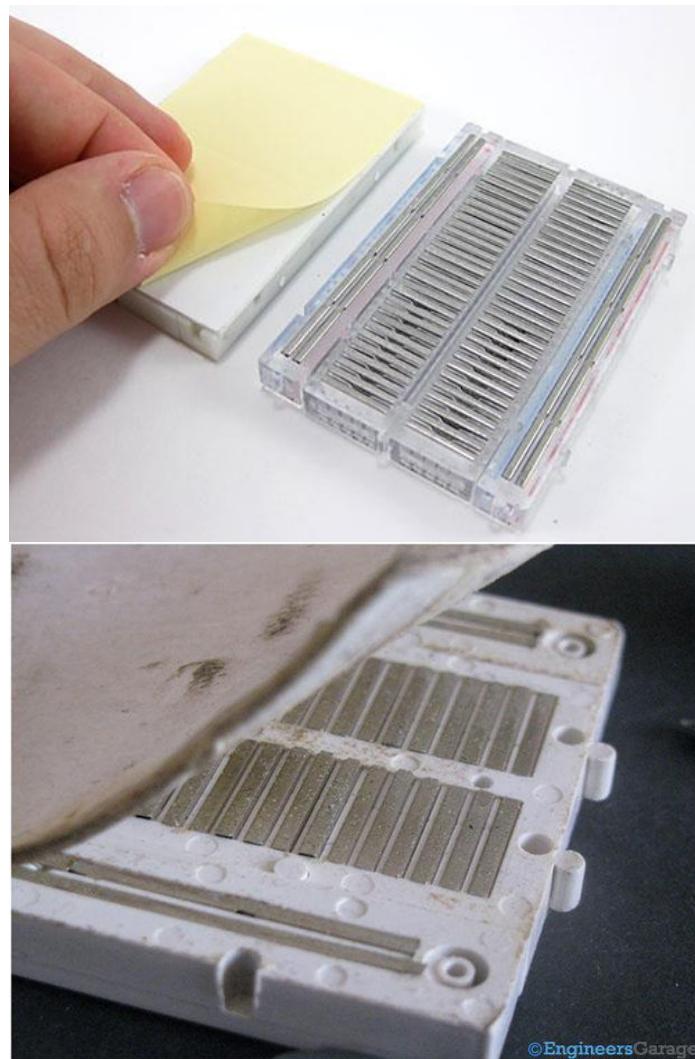


شکل ۱۱-۱ نمونه‌هایی از سوراخ‌های به هم متصل در بردبورد



شکل ۱۲-۱ کل اتصالات یک بردبورد ۳۰ ستونی

اگر پوشش پشتی بردبورد را باز کنید (عموماً با چسب محکمی چسبانده شده تا به راحتی و در حین کار جدا نگردد)، می‌توانید نحوه‌ی ایجاد اتصالات مذکور را ببینید. در شکل ۱۳-۱ این اتصالات در چند مورد و با زوایای مختلف نشان داده شده است. در این آزمایشگاه به هیچ عنوان پوشش پشت بردبوردها را باز نکنید.



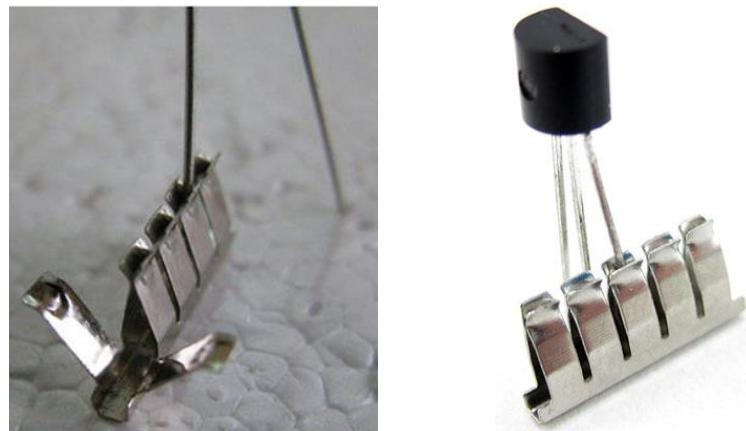
شکل ۱۳-۱ نحوه ایجاد اتصالات بردبورد

اتصالات ستون‌های ۵تایی سوراخ‌ها توسط قطعاتی فلزی که در شکل ۱۴-۱ نمونه‌ای از آن‌ها را مشاهده می‌کنید ایجاد می‌شوند. در حقیقت قسمت بالای این قطعات شامل ۵ جفت نوار فلزیست که اتصالات فنرگونه‌ای ایجاد می‌کنند و هر جفت دقیقاً در راستای یک سوراخ بردبورد قرار می‌گیرد که پایه‌های قطعات پس از وارد شدن در سوراخ بین دو نوار قرار گرفته و توسط آن جفت نوار محکم نگه داشته می‌شود و در ضمن به ۴ سوراخ دیگر از نظر الکتریکی متصل می‌شود.



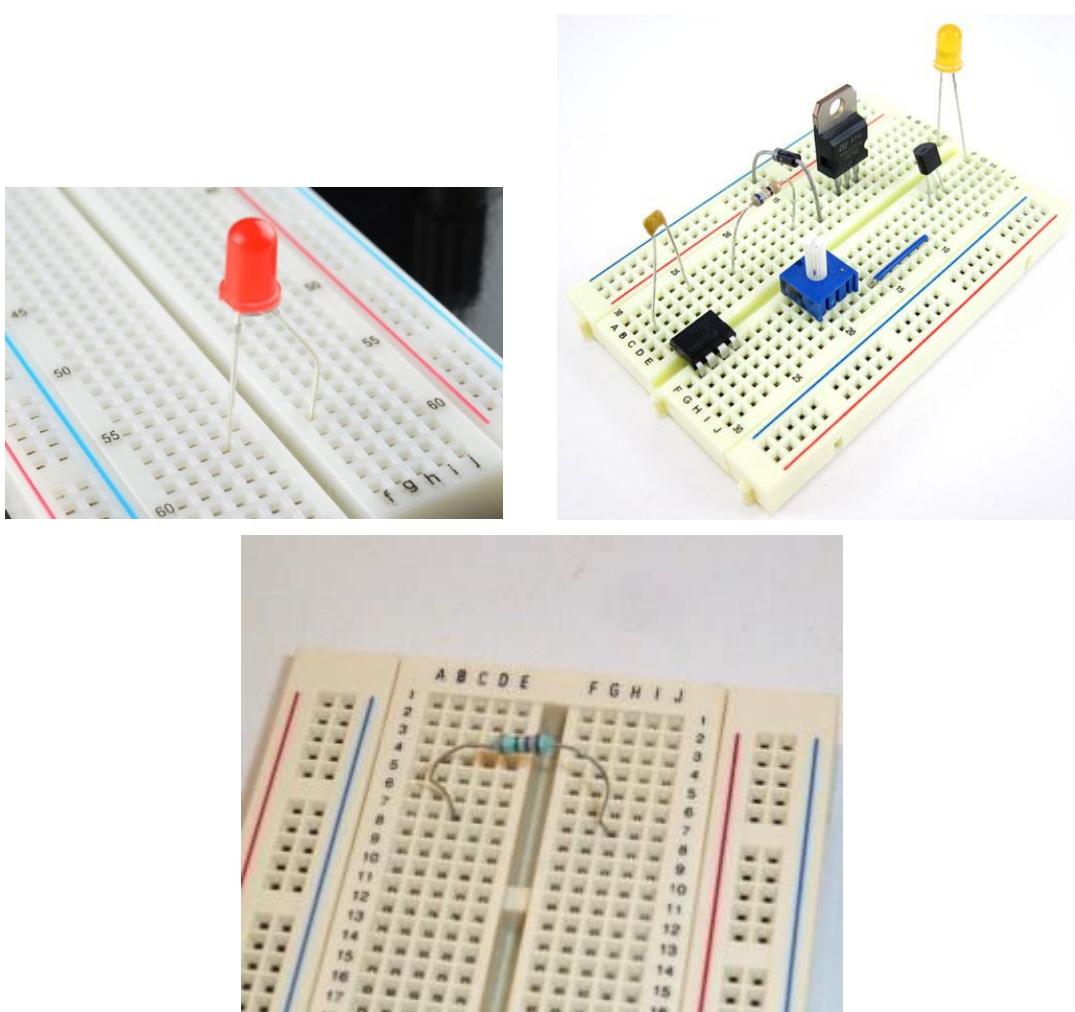
شکل ۱۴-۱ اتصال دهنده‌های ستون‌های ۵ تایی سوراخ‌ها

در شکل ۱۵-۱ نمونه‌هایی آورده شده که در آن‌ها می‌توانید ببینید که چگونه پایه‌ای از یک قطعه‌ی الکترونیکی بین دو نوار فلزی قرار گرفته شده و این دو نوار به علت خاصیت فنری که دارند آن پایه را محکم کرده‌اند تا به راحتی در حین انجام آزمایش تکان نخورد و اتصال الکتریکی آن با اجزاء دیگر برقرار بماند.



شکل ۱۵-۱ نحوه‌ی قرار گرفتن یک پایه از قطعات در بین دو نوار با خاصیت فنری

در شکل ۱۶-۱ هم نحوه‌ی قرار دادن تعدادی از قطعات در بردبورد را مشاهده می‌کنید. همانطور که مشاهده می‌شود پایه‌های قطعات باید در سوراخ‌های بردبرد فشار داده شوند تا توسط فنرهای مذکور محکم نگه داشته شوند.



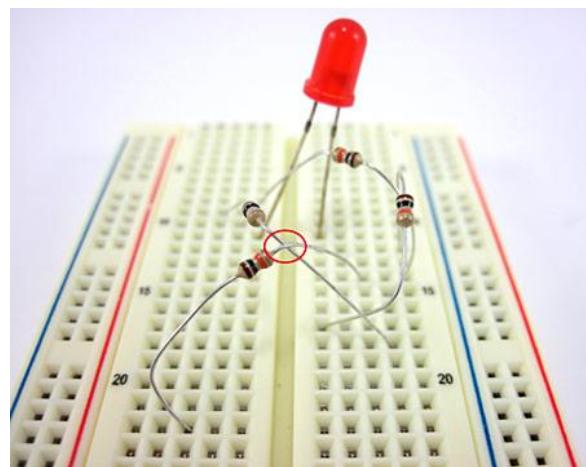
شکل ۱۶-۱ نمونه‌هایی از قرار دادن قطعات در بردبورد

باید توجه داشت که به دلیل اینکه ستون‌های ۵ تایی سوراخ‌ها از داخل به هم متصل می‌باشند، قرار دادن دو پایه یا بیشتر از یک قطعه‌ی الکترونیکی در یک ستون باعث می‌شود که عملً از نظر الکتریکی به هم متصل شوند و در نتیجه قطعه‌ی مورد نظر کار خود را انجام ندهد. در نتیجه چنینی شبیه به آنچه در شکل ۱۷-۱ می‌بینید نادرست است.



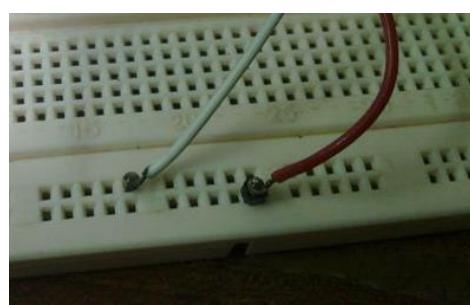
شکل ۱۷-۱ نمونه‌ای از قرار دادن نادرست قطعات روی بردبورد

نکته‌ی دیگری که باید به هنگام قرار دادن قطعات دقت کنید آن است که پایه‌های قطعات، مخصوصاً آن‌هایی که بلند می‌باشند آنقدر به هم نزدیک نشوند که در حین کار سهواً به هم برخورد کنند، چرا که در این صورت به علت هادی بودن پایه‌ها، اتصالات ناخواسته به وجود می‌آید. مخصوصاً وقتی مدار شلوغ است و قرار است تعداد زیادی قطعه در یک جای کوچک در کنار هم قرار گیرند. شکل ۱۸-۱ نمونه‌ای از این برخوردهای سهواً را نشان می‌دهد.



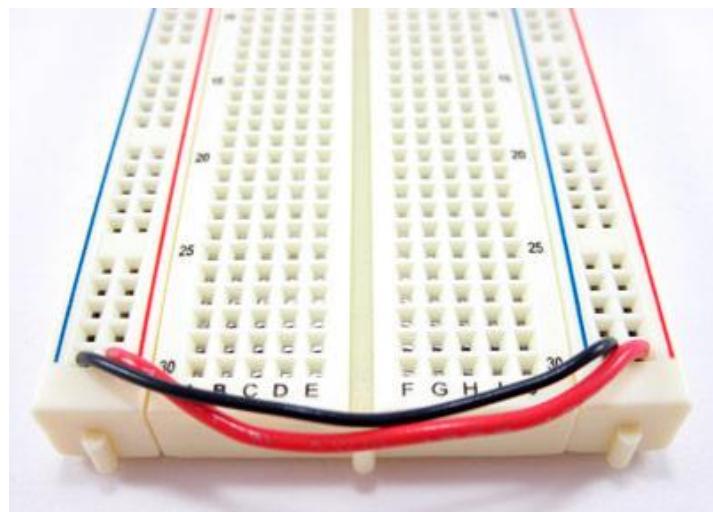
شکل ۱۸-۱ اتصال ناخواسته‌ی پایه‌ی قطعات به هم

حال به نحوه استفاده از بردبورد برای پیاده‌سازی مدارات می‌پردازیم. معمولاً در همه مدارها نیاز به یک منبع تغذیه داریم. معمولاً از چهار ردیف بالا و پایین بردبورد برای اتصال تغذیه استفاده می‌کنیم چون آن‌ها مناسب این کار هستند و کار رساندن تغذیه‌ی الکتریکی مورد نظر به قطعات را آسان می‌سازند. بدین منظور کافیست از دو قطب مثبت و منفی منبع تغذیه دو سیم به دو سطر از چهار سطر بردبورد که از این به بعد به آن‌ها ریل گفته می‌شود (و نامگذاری مرسومی هم هست) متصل کنیم. در شکل ۱۹-۱ این مطلب نشان داده شده است. توجه داشته باشید که ریل‌ها از نظر الکتریکی به ریل‌های دیگر متصل نیستند (و اینطور هم نباید باشد چون عملأً داشتن چند ریل به کار خاصی نمی‌آمد).



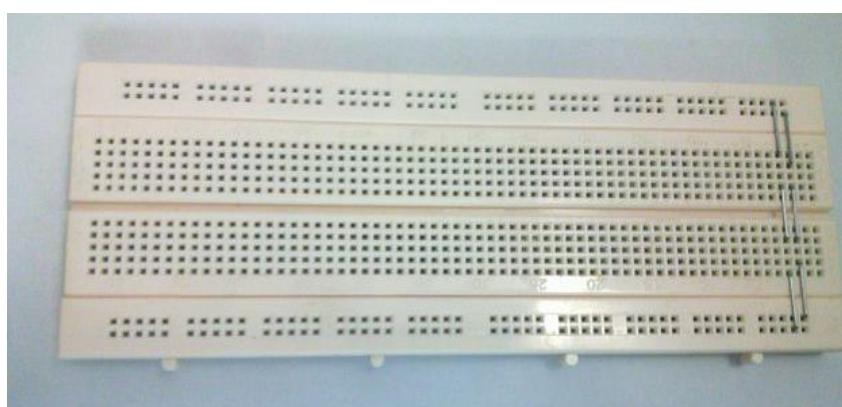
شکل ۱۹-۱ اتصال تغذیه برای قطعات بردبورد

با توجه به مورد ذکر شده چنانچه بخواهیم هم از بالای برdbورد تغذیه داشته باشیم و هم از پایین آن باید ریل‌های بالا و پایین را (نظیر به نظیر) به هم وصل کنیم. این کار مطابق شکل ۲۰-۱ انجام می‌شود. البته دقت کنید که ریل‌ها مشابه هم می‌باشند و اینکه از کدام ریل برای کدام ولتاژ (ثبت یا منفی) استفاده شود مهم نیست و فقط به قرارداد ما و اینکه به چه صورتی راحت هستیم بستگی دارد. البته در برخی برdbوردها از نوارهای رنگی قرمز و آبی (در بعضی مشکی) در کنار ریل‌ها برای کمک به این قضیه استفاده شده است.



شکل ۲۰-۱ اتصال ریل‌های بالا و پایین برdbورد به هم برای داشتن تغذیه‌ها از هر دو طرف آن

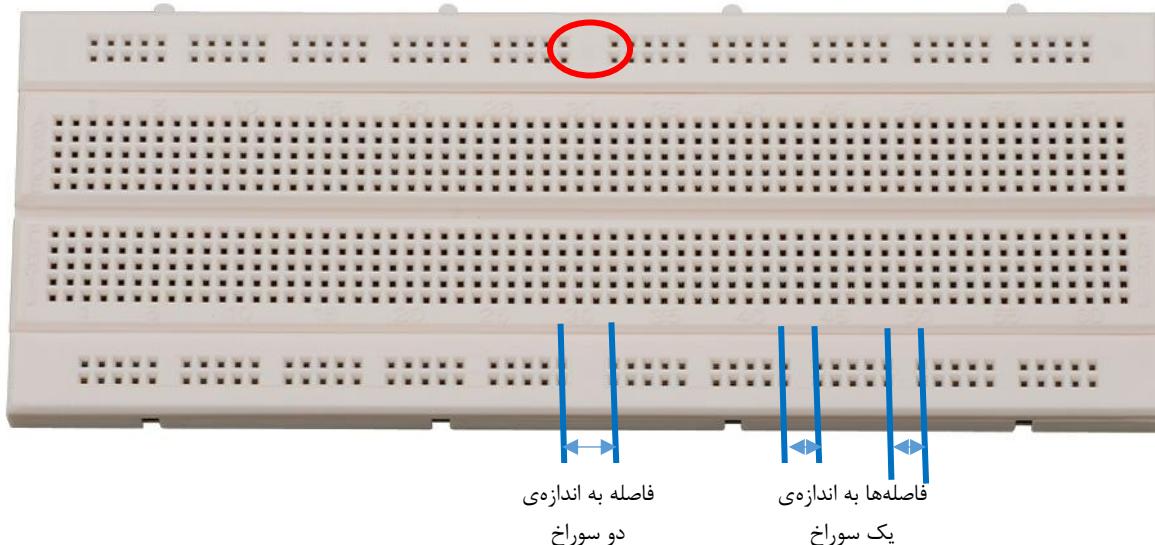
راهکار دیگری که برای این منظور می‌توان انجام داد و کمی هم مطمئن‌تر و محکم‌تر است، استفاده از سوزن‌های ماشین دوخت (مطابق شکل ۲۱-۱) است. قبل از این کار از رسانا بودن دو سر سوزن‌های مذکور و اینکه پوشش عایقی نداشته باشند اطمینان حاصل کنید (برای این کار می‌توان از تست بوق مالتی‌متر برای تشخیص رسانایی استفاده کرد).



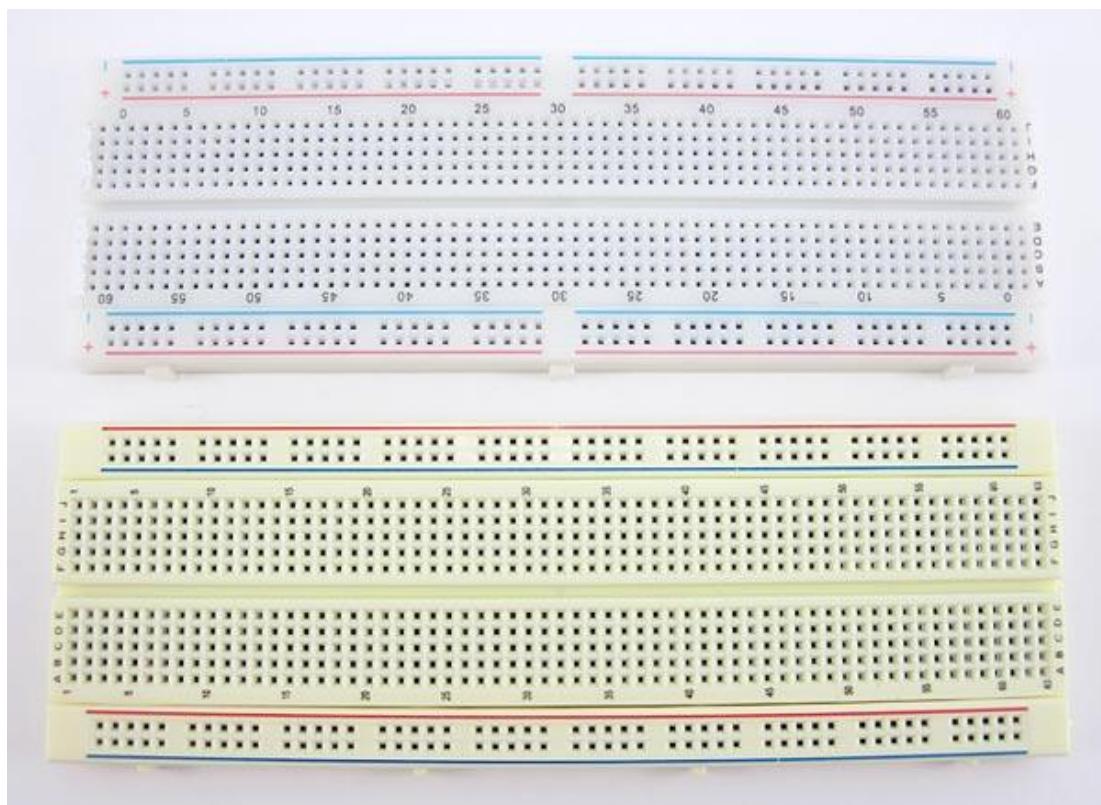
شکل ۲۱-۱ راهی دیگر برای اتصال ریل‌های بالا و پایین تغذیه

دقت داشته باشید در برخی از برdbوردهای بزرگ (Full Size) ریل‌های تغذیه از وسط از هم جدا شده‌اند و به هم متصل نیستند. با نگاه کردن به شکل ظاهری برdbوردها می‌توان این نوع‌ها را تشخیص داد. عموماً فاصله‌ی

بین سوراخ‌های وسط ریل‌ها در اینگونه بردبوردها بیشتر (به اندازه‌ی دو سوراخ) است در صورتی که در بردبوردهای با ریل‌های متصل فاصله‌ی سوراخ‌های وسط با سوراخ‌های دیگر فرقی نمی‌کند و همه به اندازه‌ی یک سوراخ فاصله‌ی دارند. این مطلب در شکل ۲۲-۱ نشان داده شده است. در بردبوردهایی که ریل‌ها با رنگ نشانه‌گذاری شده‌اند هم، برای نشان دادن این امر، خطوط رنگی در وسط منقطع شده‌اند (شکل ۲۳-۱).

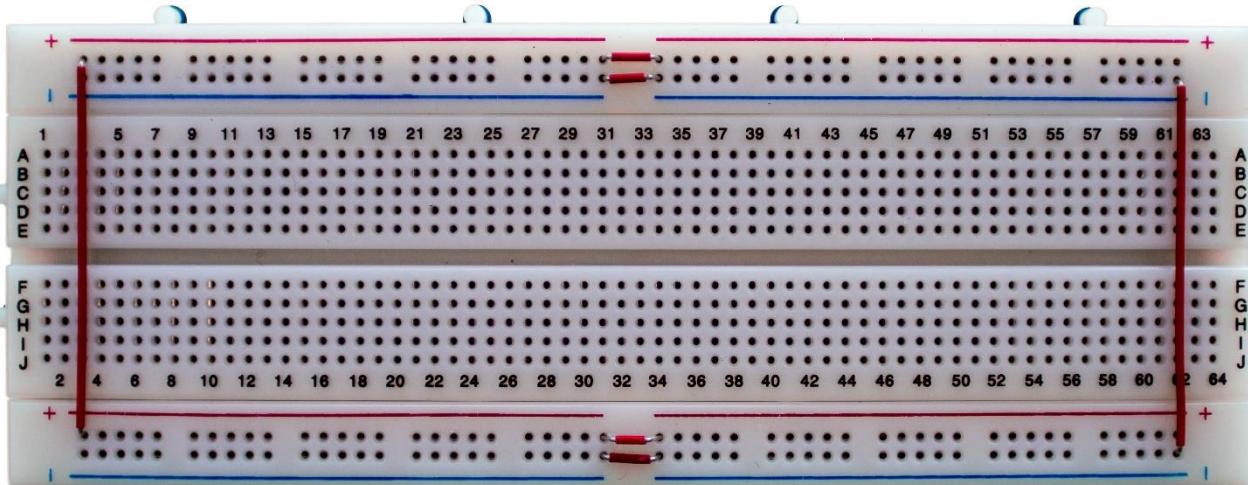


شکل ۲۲-۱ بردبوردهایی که ریل‌های تغذیه‌ی از وسط منقطع دارند



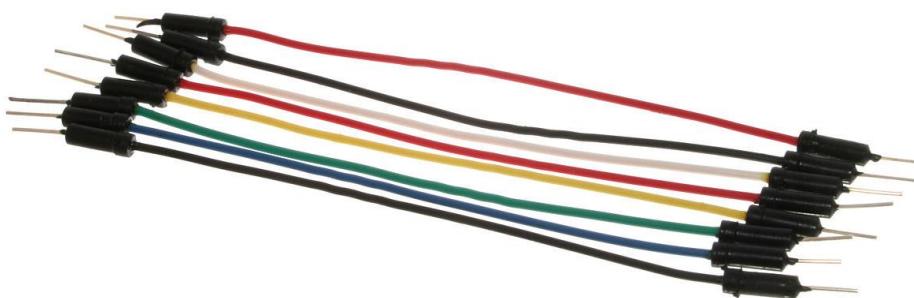
شکل ۲۳-۱ دو نوع بردبورد با ریل‌های منقطع (بالا) و پیوسته (پایین)

در اینگونه بردبوردها باید مطابق شکل ۲۴-۱ از سیم‌های کوتاهی (یا سوزن ماشین سوخت) برای اتصال بین دو نیمه‌ی ریل‌ها استفاده کرد تا تغذیه را در کل امتداد بردبورد داشته باشیم (البته در صورتی که نیاز داشته باشیم).



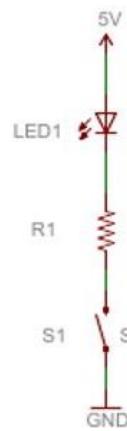
شکل ۲۴-۱ ایجاد اتصال در وسط ریل‌های تغذیه‌ی بردبوردهایی که ریل‌هایشان از وسط قطع می‌باشند

اتصال بین قطعات را می‌توان با استفاده از سیم‌ها انجام داد. نمونه‌ای از سیم‌های مخصوص این کار در شکل ۲۵-۱ نشان داده شده است. در صورت عدم وجود چنین سیم‌هایی می‌توان از سیم‌های معمولی (مفتوحی نازک) روکش دار استفاده کرد. توجه داشته باشید که این سیم‌ها نه کلفت باشند که با چند بار استفاده خاصیت فنری بردبوردها را از بین ببرند و نه چندان نازک که اتصال برقرار نگردد یا مدام قطع شود. بهترین قطر برای این سیم‌ها به اندازه‌ی قطر پایه‌ی قطعات مرسوم الکترونیکی است.



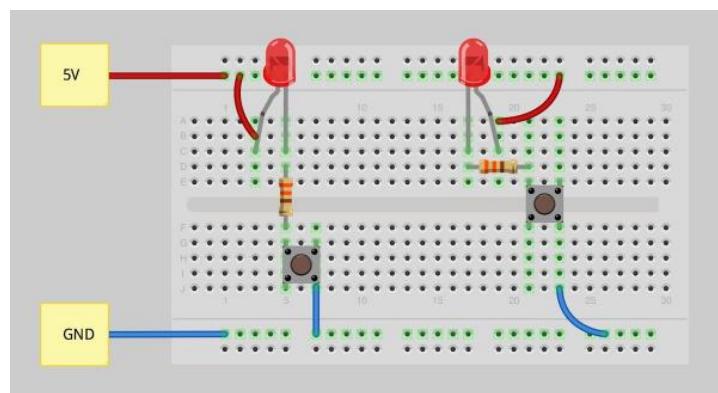
شکل ۲۵-۱ سیم‌های آماده برای استفاده به منظور ایجاد اتصال بین قطعات

می‌خواهیم مداری ببندیم که در آن یک LED (دیود نورانی) را روشن کنیم. این مدار در شکل ۲۶-۱ نشان داده شده است. داشتن مقاومت الزامی است چون بدون آن LED خواهد سوخت. می‌توان از کلید استفاده نکرد و در این صورت LED به طور دائم روشن خواهد بود.

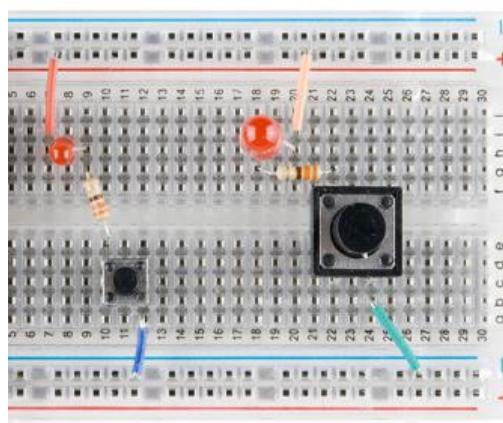


شکل ۲۶-۱ مدار اتصال LED به منبع تغذیه

این مدار را به روش‌های گوناگونی می‌توان روی بردبورد سوار کرد که در شکل ۲۷-۱ دو روش اتصال نشان داده شده است که از نظر الکتریکی هر دو دقیقاً یک کار را انجام می‌دهند. در شکل ۲۸-۱ هم یک عکس واقعی که از بسته شدن مدارهای فوق روی بردبورد گرفته شده نشان داده شده است.

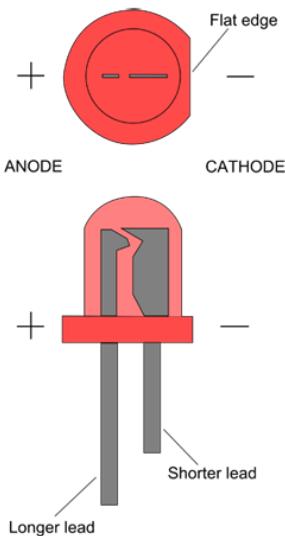


شکل ۲۷-۱ دو روش اتصال مدار LED در بردبورد



شکل ۲۸-۱ یک عکس واقعی از بستن مدارهای LED (به دو روش)

توجه داشته باشید که LED ها قطبی هستند، یعنی پایه‌هایشان با هم فرق می‌کند. در شکل ۲۶-۱ سر بالای LED سر آند (Anode) و سر پایینی آن سر کاتد (Cathode) است. شکل ۲۹-۱ سه روش شناسایی پایه‌های LED (آند و کاتد) را از هم نشان می‌دهد. اگر پایه‌های LED سالم باشند و از قبل کوتاه نشده باشند، آن پایه که بلندتر است آند است و دیگری کاتد. در غیر اینصورت به پایین بدنه‌ی آن نگاه کنید. طرفی که یک قسمت تخت روی آن ایجاد کرده‌اند کاتد است و دیگری آند. راه دیگر نگاه کردن به داخل LED است. طرفی که فلز بزرگتر (منظور با سطح مقطع بیشتر) دارد کاتد و آن دیگری آند است.



شکل ۲۹-۱ تشخیص پایه‌های LED از هم

مقاومت به کار رفته در مدار فوق ۳۳۰ اهم است که وظیفه‌ی آن ممانعت از عبور جریان بسیار زیاد و به تبع آن جلوگیری از سوختن LED است (توجه داشته باشید که LED یک دیود است و در صورت وصل شدن به یک منبع جریان بزرگ‌تر از V_t به صورت ایده‌آل جریان بی‌نهایت را از خود عبور خواهد داد. البته LEDها ایده‌آل نیستند و یک مقاومت داخلی کوچک جریان را کنترل خواهد کرد ولی معمولاً آن مقاومت داخلی آنقدر بزرگ نیست که بتواند مانع سوختن LEDها که با ولتاژ ۵ ولت کار می‌کنند شود بنابراین قرار دادن یک مقاومت در محدوده چند صد اهم می‌تواند جریان عبوری را محدود و مانع سوختن LEDها شود).

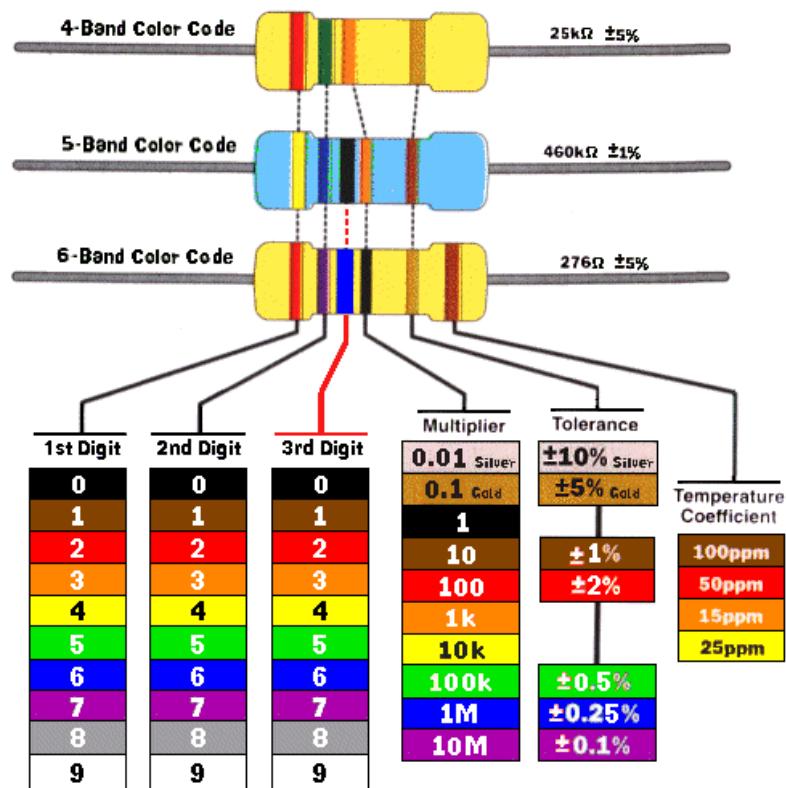
شکل ۳۰-۱ (الف) و (ب) دو نحوه‌ی نمایش کدهای مقاومتی را نشان می‌دهند. اغلب مقاومت‌های استفاده شده در آزمایشگاه‌ها دارای ۴ نوار (۴ باند) می‌باشند (مقاومت بالایی شکل ۳۰-۱(الف)) و رنگ چهارم که با کمی فاصله از بقیه قرار گرفته (یک روش تشخیص ترتیب شماره‌ی رنگ‌ها) اغلب طلازیست (تولرانس ۰.۵٪).

اگر رقم متناظر رنگ اول یک مقاومت ۴ باندی را a ، رنگ دوم آن را b و رنگ سوم آن را c بنامیم، مقدار مقاومت بر حسب اهم از رابطه‌ی زیر بدست می‌آید:

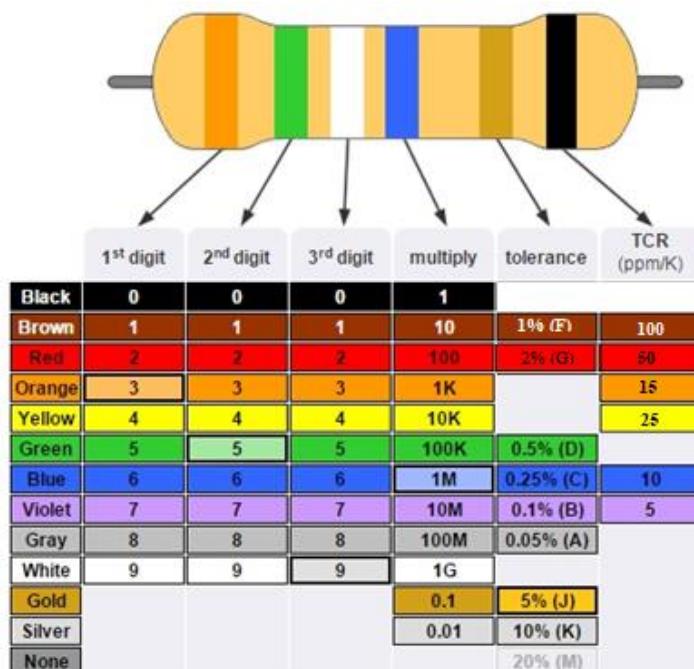
$$ab \times 10^c$$

که در آن ab به معنی در کنار هم قرار دادن دو رقم a و b است. مثلاً زرد، بنفش، سبز مقدار $4.7M\Omega$ را می‌دهد.

مقاومت‌های استاندارد با ضرایب ۱، ۱.۲، ۱.۵، ۱.۸، ۲.۲، ۲.۷، ۳.۳، ۳.۹، ۴.۷، ۵.۶ و ۶.۸ می‌باشند.



(الف)



(ب)

شکل ۳۰-۱ کدهای مقاومتی

- ۱- به جای کلید در شکل ۲۶-۱، دو ترمینال یک مالتی‌متر که در حالت آمپر‌متر قرار داده شده را قرار دهید و مقدار جریانی که از مدار می‌گذرد را حساب کنید. در صورتی که مالتی‌متر با قابلیت اندازه‌گیری جریان در اختیار ندارید یا در اختیار دارید ولی قسمت آمپر‌متر آن خراب است (معمولًاً فیوزهای محافظ قسمت آمپر‌متر به دلیل حساسیت بالا و اتصال اشتباه می‌سوزند پس زیاد با این مورد مواجه می‌شوید)، کافیست برای اندازه‌گیری جریان عبوری از مدار، ولتاژ دو سر مقاومت را بخوانید و تقسیم بر مقدار مقاومت کنید (به جای کلید در این حالت یک سیم قرار دهید).
- ۲- مقدار ولتاژ روی مقاومت و ولتاژ روی LED را با مالتی‌متر که در حالت ولت‌متر قرار داده شده بخوانید و بنویسید.
- ۳- مقاومت را با مقادیر ۱۰۰ اهم، ۴۷۰ اهم و ۱ کیلو اهم عوض کنید و در هر حالت مراحل ۱ و ۲ را انجام دهید. در ضمن در هر حالت به شدت نور LED و اینکه خاموش یا روشن است هم توجه داشته باشید و مشاهدات و نتیجه‌گیری‌ها را یادداشت کنید.

۲- آشنایی با گیت‌های منطقی پایه

پیش‌گزارش:

تراشه‌های مورد استفاده در این آزمایش هر کدام چه گیتی هستند؟ درون هر کدام از آن‌ها چند گیت منطقی وجود دارد؟

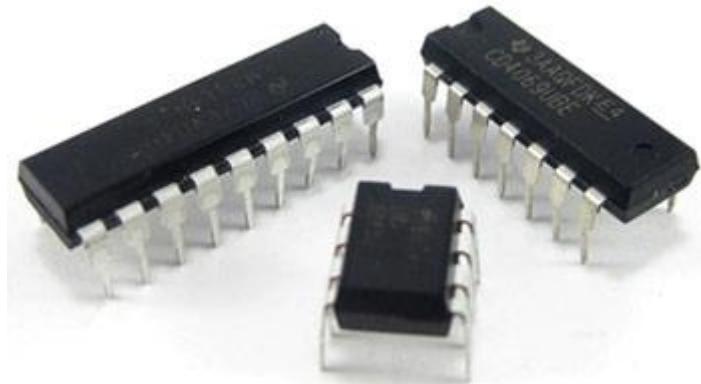
وجود پایه‌های VCC و GND در تراشه‌های منطقی به چه منظور است؟

حدود ولتاژ‌های صفر و یک منطقی چقدر است؟

وسایل مورد نیاز:

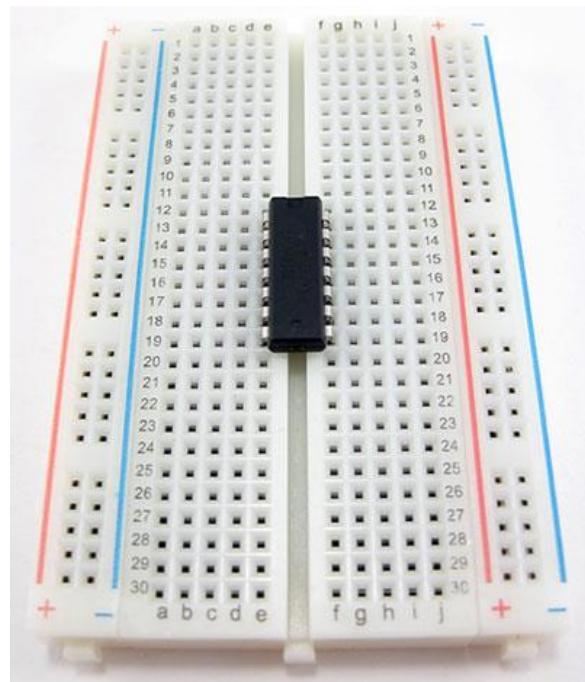
- منبع تغذیه، بردبورد، مالتی‌متر،
- مقاومت 150 اهمی، دیود نورانی (LED)،
- تراشه‌های 7400، 7404، 7406، 7408، 7432، 7486، 4011.

در شکل ۱-۲ نمونه‌هایی از پکیج (بسته‌بندی) تراشه‌های مرسوم موجود در آزمایشگاه را مشاهده می‌کنید که به آن‌ها گفته می‌شود و دارای دو ردیف پایه به موازات هم می‌باشند.



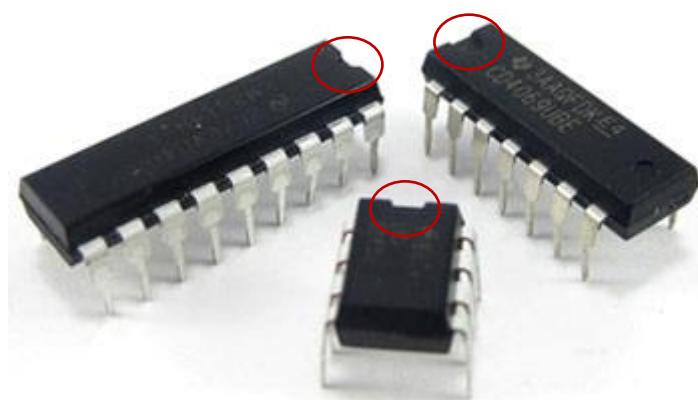
شکل ۱-۲ نمونه‌هایی از تراشه‌های DIP

برای قرار دادن این تراشه‌ها در بردبورد به روشی که در شکل ۲-۲ نشان داده شده عمل می‌کنیم. توجه داشته باشد که حتماً باید دو ردیف پایه‌ها در دو طرف شیار وسطی بردبورد قرار گیرند چرا که در غیر اینصورت همه‌ی جفت پایه‌های روبروی هم تراشه به علت اینکه ستون‌های ۵ تایی بردبورد در داخل آن به هم متصل‌اند، نظیر به نظیر به هم وصل می‌شوند.

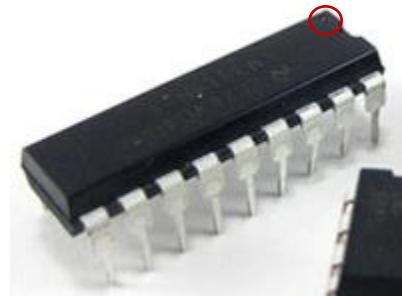


شکل ۲-۲ نحوه اتصال تراشه‌های DIP به بردبورد

هر پایه‌ی تراشه یک شماره به خود اختصاص داده و شماره‌گذاری پایه‌ها بدین صورت است که وقتی از بالا به تراشه نگاه کنید و طوری آن را در دست بگیرید که بریدگی نیم‌دایره‌ی آن (که در شکل ۳-۲ نشان داده شده است) در سمت چپ واقع شود، پایه‌ی پایینی سمت چپی (که گاهی با یک نقطه فرورفتگی در بالای آن مشخص می‌شود و در شکل ۴-۲ نشان داده شده است) پایه‌ی شماره‌ی ۱ شده و پایه‌ی کناری آن پایه‌ی شماره‌ی ۲ و به همین ترتیب تا پایه‌ی انتهای پایینی تراشه می‌شماریم و سپس از روی‌روی آن (انتهای سمت راست بالایی) ادامه می‌دهیم (شماره‌گذاری به شکل پاد ساعتگرد است). بدین صورت پایه‌ی آخر، پایه‌ی انتهای سمت چپ بالایی تراشه خواهد بود.



شکل ۳-۲ بریدگی نیم‌دایره‌ای تراشه‌ها



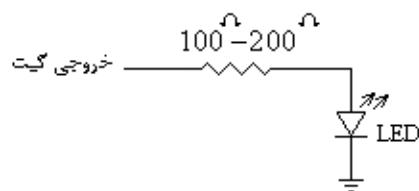
شکل ۴-۲ علامت فرورفتگی بالای پایه‌ی یک

۱- تراشه‌ی 7404 (که شامل شش گیت NOT است) را آزمایش کنید.

برای این کار این تراشه را روی بردبورد قرار دهید و پایه‌های زمین یا GND (پایه‌ی شماره‌ی ۷) و VCC (پایه‌ی شماره‌ی ۱۴) آن را به یک منبع تغذیه با خروجی ۵V متصل کنید. نقشه‌ی تراشه‌ها در انتهای آمده است

به ورودی یکی از گیتهای NOT تراشه، به ترتیب ولتاژ ۰ و ۵ ولت (یعنی ۰ و ۱ منطقی) وصل کنید و خروجی را در این دو حالت با مالتی‌متر اندازه بگیرید. ولتاژ خروجی در دو حالت صفر منطقی و یک منطقی چه مقداری است؟ حدود سطح‌های ولتاژ صفر منطقی و یک منطقی را در این آزمایش یادداشت کنید تا در آزمایش‌های بعدی به هنگام خطایابی از آن استفاده کنید.

با باز نگه داشتن ورودی (وصل نکردن به هیچ منبعی) یا به اصطلاح N.C)، خروجی چه مقداری به خود می‌گیرد؟ صفر منطقی است یا یک منطقی؟ در این صورت، ورودی صفر منطقی تعبیر می‌شود یا یک منطقی؟ همچنین خروجی تراشه 7404 را در دو حالت یک منطقی و صفر منطقی با استفاده از دیود نورانی (LED) مشاهده کنید. توجه کنید که LED را مستقیماً به خروجی وصل نکنید؛ چون ممکن است جریان زیادی کشیده شود و تراشه یا دیود بسوزد. بدین منظور LED را معمولاً با یک مقاومت بین ۱۰۰ تا ۳۳۰ اهم به خروجی مورد نظر وصل می‌کنند (شکل ۵-۲).



شکل ۵-۲ اتصال LED به خروجی تراشه‌ها

۲- تراشه‌های 7400، 7408، 7432، 7486 و 4011 را هم آزمایش کنید.

آن‌ها را به ترتیب روی بردبورد قرار دهید و طبق جدول ۱-۲ به پایه‌های ورودی، مقدار دهید و مقدار ولتاژ خروجی را در جدول یادداشت کنید.

همان طور که در درس الکترونیک دیجیتال آموزش داده خواهد شد، برای ساخت مدارات منطقی تکنولوژی ساختهای متفاوتی وجود دارد و این تکنولوژی‌ها می‌توانند باعث تغییراتی در عملکرد تراشه نسبت به دیگری شوند. برای مثال، این که اگر ورودی یک تراشه را به هیچ ولتاژی وصل نکنیم چه خواهد شد و یا این که پایه‌های خروجی تراشه تا چه حد قادر به تولید جریان هستند بستگی به تکنولوژی دارد. در ادامه ما دو خانواده رایج از تراشه‌ها (با تکنولوژی‌های CMOS و TTL) را آزمایش می‌کنیم.

تفاوت بین اندازه‌ی ولتاژهای خروجی در حالت‌های یک منطقی و صفر منطقی بین تراشه‌های TTL (سری 7400) یا CMOS تراشه‌هایی که عموماً شماره آن‌ها با 74 شروع می‌شود با حروف وسط S، H، F، ALS، LS (بدون حرف وسط) و (تراشه‌هایی که عموماً شماره آن‌ها با 4 شروع می‌شود یا شروع با 74 و حرف وسط عموماً HC و آن‌هایی که C دارند) را با توجه به اعدادی که در جدول پر کرده‌اید بیان کنید. در سطر اول جدول هم حرف وسط تراشه را در جای خالی پر کنید (در صورتی که حرف وسط نداشت با یک علامت ضربدر در جای خالی این را نشان دهید). همچنین وقتی ورودی را NC می‌کنید چه تفاوتی در خروجی این دو نوع تراشه مشاهده می‌کنید؟

در صورتی که در آزمایشگاه گونه‌های متفاوتی (نه از نظر عملکرد بلکه از نظر نحوه ساخت) از تراشه‌ها (که حروف وسط متفاوتی دارند) موجود است آن‌ها را هم آزمایش کنید و ولتاژهایشان را در ستون‌های خالی جدول با ذکر حرف وسط یادداشت کنید. لازم نیست این کار را برای همه گیت‌های مذکور و برای همه‌ی حالات ورودی امتحان کنید. بلکه از هر گیتی یک نمونه و برای آن هم دو حالت خروجی 1 و 0 منطقی را انجام دهید.

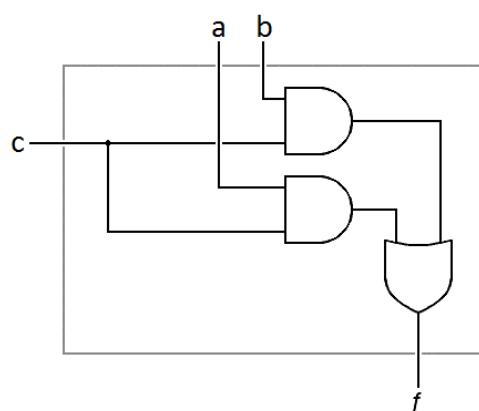
در طراحی مدارهای منطقی، هیچ‌گاه نباید ورودی را باز (NC) نگه داشت؛ چون ممکن است باعث اختلال در عملکرد مدار شود.

جدول ۱-۲ ولتاژهای ورودی و خروجی تراشه‌های مختلف

A	B	74--00	74---08	74---32	74---86	4011
0	0					
5	0					
5	5					
N.C	0					
5	N.C					
N.C	N.C					

۳- مدار شکل ۶-۲ را پیاده‌سازی کنید. و به ازای تمام ورودی‌های ممکن خروجی تولید شده در جدول زیر پر کنید.

a	b	c	f
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	



شکل ۶-۲ یک نمونه مدار

۳- پیاده‌سازی توابع منطقی با استفاده از جدول کارنو

پیش‌گزارش:

توابعی که در این آزمایش آمده‌اند را با استفاده از جدول کارنو ساده کنید.

وسایل مورد نیاز:

- منبع تغذیه، بردبورد، مالتی‌متر،
- تراشه‌های 7404، 7408، 7432.

۱. تابع منطقی زیر را با استفاده از جدول کارنو ساده کرده و مدار آن را با استفاده تراشه‌های منطقی پیاده‌سازی کنید.(SOP).

$$f(A, B, C) = [(A' + B + C)(A + C)(A + B)]$$

سپس جدول ۱-۳ را با توجه به عملکرد مداری که پیاده‌سازی کرده‌اید، تکمیل کنید (ولتاژ گره‌ها را یادداشت کنید).

جدول ۱-۳ نتیجه عملکرد مدار بخش ۱

V(A)	V(B)	V(C)	V(f)

این مدار یک رأی‌گیر اکثریت است. در این مدار هر زمان حداقل دوتا از ورودی‌ها یک (یا صفر) باشند، خروجی نیز یک (یا صفر) می‌شود. از این مدار در کاربردهایی که احتمال رخداد خطأ وجود دارد استفاده می‌شود، بدین

صورت که محاسبات به طور همزمان در سه مدار انجام می‌شود و خروجی این مدارها به این رأی گیر داده می‌شوند.

۲. تابع منطقی زیر را با استفاده از جدول کارنو ساده و مدار آن را با استفاده از تراشه‌های منطقی پیاده‌سازی کنید.

$$f(A, B, C, D) = \sum(1, 3, 4, 5, 6, 9, 11, 12, 14) + d(0, 7)$$

بعد از پیاده‌سازی تابع، جدول ۲-۳ را تکمیل کنید.

در مداری که پیاده‌سازی کرده‌اید مقادیر حالات بی اهمیت چیست؟ این رفتار را چگونه توجیه می‌کنید؟

جدول ۲-۳ نتیجه عملکرد مدار بخش ۲

۴- آشنایی با زبان توصیف سختافزار^۱ (HDL)

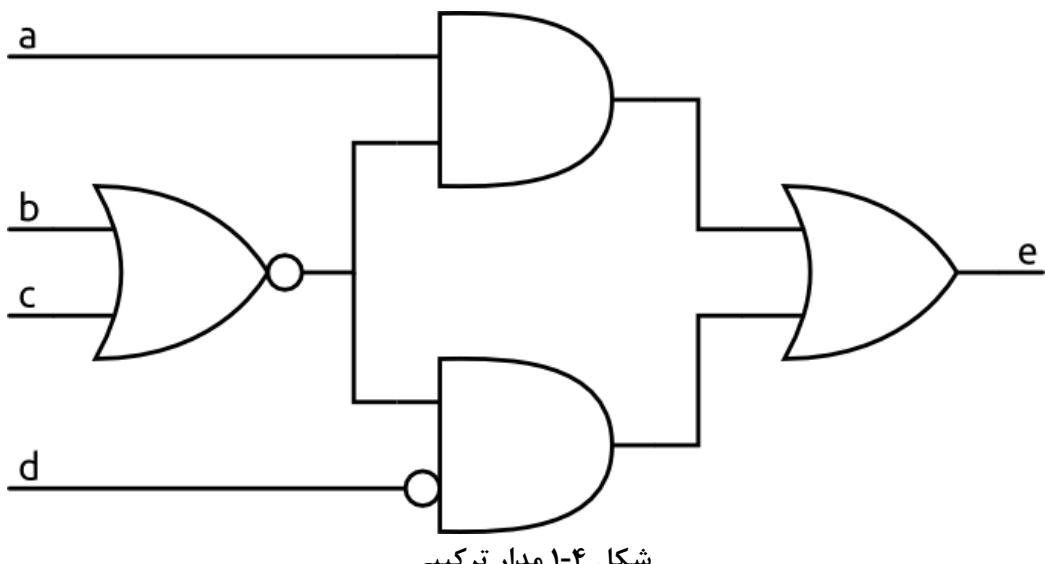
پیشگزارش:

۱. مدار شکل ۱-۴ را با زبان توصیف سختافزار Verilog توصیف کنید.
۲. برای مدار شکل ۱-۴ جدول صحت را بدست آورید.

آزمایش:

زبان توصیف سختافزار، زبانی است که از آن برای توصیف سختافزار سیستم‌های دیجیتال به صورت متنی استفاده می‌کنند. به عبارت دیگر، از زبان‌های HDL می‌توان برای توصیف رابطه‌ی منطقی بین سیگنال‌های ورودی یک مدار و سیگنال‌های خروجی آن بهره برد. از این رو، این زبان‌ها می‌توانند مدارهای منطقی، عبارت‌های بولی و یا مدارهای پیچیده را توصیف کنند. زبان‌های HDL متنوعی ارائه شده است، مانند VHDL و Verilog. در این آزمایشگاه از زبان Verilog استفاده خواهد شد.

یک شبیه‌ساز HDL توصیف مدار را دریافت و براساس مقدار ورودی‌ها و ساختار مدار یا سیستم دیجیتالی، مقدار خروجی‌ها را مشخص می‌کند. بنابراین می‌توان قبلاً از ساخت، مدار را آزمایش کرد و از صحت عملکرد آن مطمئن شد. در این آزمایشگاه شبیه‌سازی‌ها با ابزار Xilinx ISim انجام خواهد شد.



شکل ۱-۴ مدار ترکیبی

^۱ Hardware Description Language

۱. TestBench ای طراحی کنید که تمامی ورودی‌های ممکن را به مأذول بدهد. سپس مدار را شبیه‌سازی کنید.
مقادیر حاصل از شبیه‌سازی را با مقادیری که در پیش گزارش بدست آورده‌اید مقایسه کنید.
۲. اکنون که از صحت خروجی‌ها مطمئن شده‌اید، به کمک پیوست ۳: مراحل سنتز و بارگذاری کد توصیف سخت‌افزاری روی FPGA این مدار را سنتز کرده و آن را روی بورد FPGA برنامه‌ریزی کنید. مجدداً به کمک ورودی‌ها و خروجی‌های بورد، صحت عملکرد کد خود را بررسی کنید.

۵- آشنایی با مالتیپلکسر، انکدر و دیکدر

پیش‌گزارش:

مدارهای دیکدر 2×4 ، مالتیپلکسر 4×1 و انکدر اولویت 4×2 را در سطح گیت رسم کنید و توصیف آنها را با زبان Verilog در سطح گیت بنویسید.

آزمایش:

۱- مدارهای زیر را در سطح گیت و با استفاده از زبان Verilog شبیه‌سازی کنید.

- دیکدر 2×4
- مالتیپلکسر 4×1
- انکدر اولویت 4×2

۲- اکنون با استفاده از مدار دیکدری که در مرحله قبل طراحی کردید تابع زیر را پیاده‌سازی کنید (شاید در این پیاده‌سازی‌ها لازم باشد از یکی از گیت‌های پایه مانند And و Or ... استفاده کنید).

$$F(A, B, C, D) = \sum m(1, 2, 4, 7, 8, 11, 13, 14)$$

تابع $F(A, B, C, D)$ یک گیت XOR چهار ورودی است. یکی از کاربردهای این گیت‌ها ایجاد بیت‌های توازن^۲ برای داده‌ها است. استفاده از این بیت‌ها یکی از ساده‌ترین روش‌های کشف خطا در سیستم‌های کامپیوتری است. بیت‌های توازن دارای دو مدل زوج و فرد هستند. در انتقال و ذخیره اطلاعات، این بیت‌ها در کنار داده اصلی قرار می‌گیرند تا اگر داده اصلی دچار تغییر شد این تغییر کشف شود. مقدار این بیت توسط جدول ۱-۵ تعیین می‌شود:

جدول ۱-۵ تضمین‌گیری مقدار بیت افزونگی

فرد		زوج		نوع بیت توازن
فرد	زوج	فرد	زوج	تعداد یک‌های داده
.	۱	۱	.	مقدار بیت توازن

(الف) با توجه به جدول و توضیحات فوق بگویید اگر بخواهیم از این روش (پیاده‌سازی تابع بالا) برای ایجاد بیت توازن در داده‌های چهار بیتی استفاده کنیم، نوع بیت توازن ما زوج است یا فرد؟ چرا؟

(ب) اگر بخواهیم نوع دیگر از بیت توازن را بسازیم چه نوع گیتی را پیشنهاد می‌دهید؟

۳- تابع پیاده‌سازی شده توسط دیکدر را روی بورد FPGA برد و آن را تست کنید.

^۲ Parity bit

۶- تبدیل کدهای باینری و گری به یکدیگر

پیش‌گزارش:

۱. با مطالعه دستور کار این آزمایش، کدی با زبان Verilog بنویسید که تبدیل‌های کد باینری به گری و کد گری به باینری را برای اعداد ۴ بیتی انجام بدهد.

۲. توصیف مدارهای شکل ۲-۶ را با زبان Verilog و در سطح گیت بنویسید.

آزمایش:

کدگذاری‌های باینری و گری از متداول‌ترین انواع کدگذاری موجود هستند. در برخی سیستم‌ها مانند شمارنده‌های ترتیبی، می‌توان از کدگذاری گری استفاده کرد تا تغییر بیت‌ها بین هر دو حالت متوالی در ماشین حالت فقط یک بیت باشد. کاهش تعداد تغییر بیت‌ها موجب کاهش توان مصرفی سیستم خواهد شد.

- هدف این آزمایش در گام اول، تبدیل کد باینری به گری است. اعداد باینری ۱۰۱۰ و ۱۱۱۰ را طبق رابطه زیر به صورت دستی به اعداد گری تبدیل کنید. توصیف این الگوریتم را در سطح گیت با زبان Verilog بنویسید. فرض کنید تمامی گیت‌های این آزمایش تأخیر ۱۰ نانوثانیه دارند.

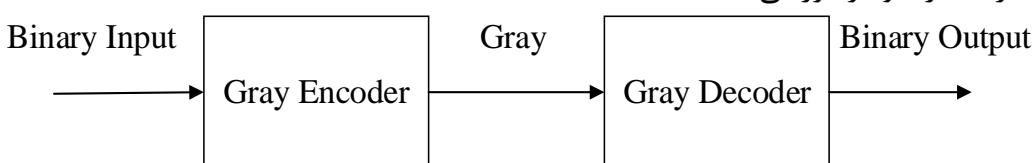
$$\begin{aligned} g_i &= b_i & i &= n; \\ g_i &= b_{i+1} \text{xor } b_i & i &= 1, 2, \dots, n-1; \end{aligned}$$

در گام دوم، کدهای گری ۱۰۱۰ و ۱۱۱۰ را مطابق رابطه زیر به صورت دستی به کدهای باینری تبدیل کنید. توصیف این الگوریتم را در سطح گیت با زبان Verilog بنویسید. فرض کنید تمامی گیت‌های این آزمایش تأخیر ۱۰ نانوثانیه دارند.

$$\begin{aligned} b_i &= g_i & i &= n; \\ b_i &= b_{i+1} \text{xor } g_i & i &= 1, 2, \dots, n-1; \end{aligned}$$

در گام سوم، صحت کدهای خود را با نوشتن مدار آزمون^۳ برای همهی حالت‌ها شبیه‌سازی کنید. در این بخش لازم است تا دانشجویان صحت هر دو تبدیل را بررسی کنند.

در گام چهارم این آزمایش لازم است تا مژویه‌هایی را که در بالا نوشته‌اید مطابق شکل ۱-۶ به یکدیگر متصل کنید. سپس آن‌ها را سنتز کرده و روی بورد FPGA پیاده‌سازی کنید. اکنون با توجه به ورودی‌ها و خروجی‌های بورد صحت عملکرد مدار خود را بررسی کنید.



شکل ۱-۶ نحوه اتصال مژویل‌ها

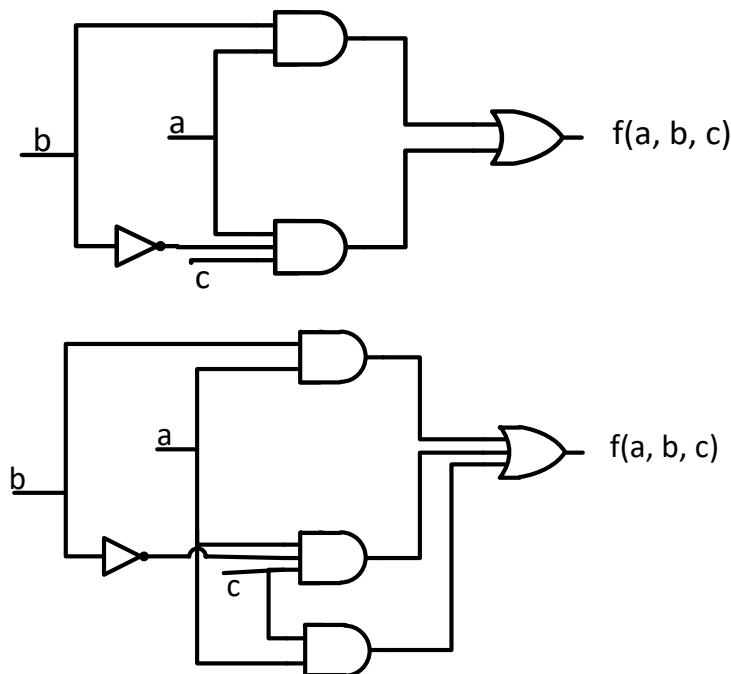
^۳Testbench

۲- مدارهای شکل ۲-۶ را شبیه‌سازی کنید و تأخیر گیت‌ها را به صورت زیر در نظر بگیرید (گیت not ۱۵ نانوثانیه، سایر گیت‌ها ۱۰ نانوثانیه).

بردار ورودی زیر را به مدار اعمال کنید:

$$abc: 111 \rightarrow 101$$

در خروجی این دو مدار چه تفاوتی مشاهده می‌کنید؟ دلیل این تفاوت را توجیه کنید.



شکل ۲-۶ مدارهای ترکیبی دارای تأخیر

۷- پیاده‌سازی مدار جمع کننده-تفریق کننده چهار بیتی به صورت ساختاری

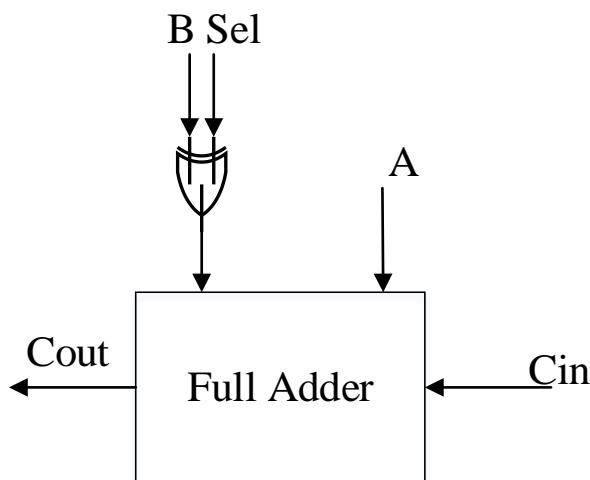
پیش‌گزارش:

پس از مطالعه دستور کار این آزمایش، توصیف‌های خواسته شده را به زبان Verilog بنویسید.

آزمایش:

۱. در گام نخست این آزمایش، هدف پیاده‌سازی مدار شکل ۱-۷ است. این مدار دو بیت A و B را دریافت و بر اساس اینکه مقدار Cin صفر یا یک باشد، به صورت متناظر دو بیت را با یکدیگر جمع یا از یکدیگر تفریق می‌کند.

از دانشجویان انتظار می‌رود تا این مدار را با زبان Verilog به صورت ساختاری^۴ توصیف کنند. توصیف مدار تمام جمع کننده‌ی این شکل نیز باید به صورت ساختاری انجام پذیرد (فرض کنید تأخیر گیت‌های AND و OR هر کدام ۵ نانو ثانیه، گیت XOR ۱۰ نانو ثانیه و گیت NOT ۲ نانو ثانیه است. تأخیرهای ذکر شده در تمامی مراحل این آزمایش برقرار هستند).

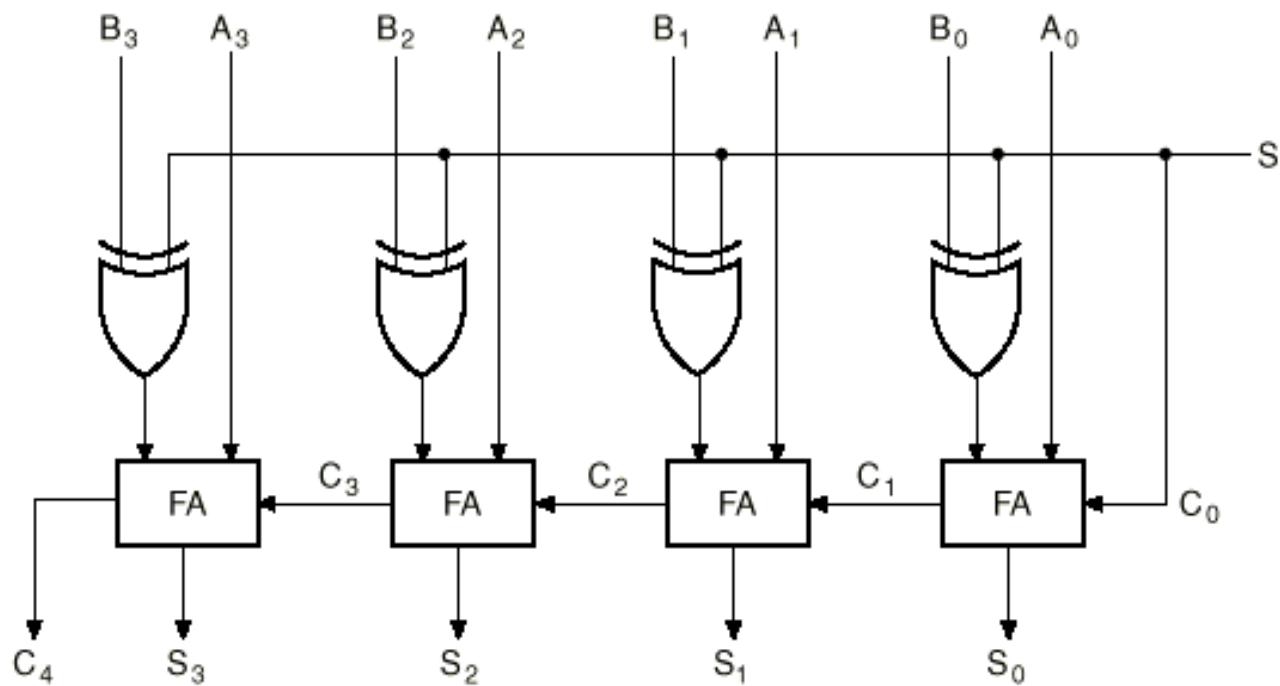


شکل ۱-۷ مدار جمع کننده-تفریق کننده تک بیتی

۲. در گام دوم این آزمایش لازم است تا دانشجویان مطابق شکل ۲-۷ مدار جمع کننده-تفریق کننده ۴-بیتی را با استفاده از مدار بخش اول توصیف کنند.

اکنون دانشجویان باید برای این مدار یک برنامه آزمون نوشته و مدار را به ازای حداقل ۳ حالت تفریق و ۳ حالت جمع آزمایش کنند. پس از اعمال ورودی تا زمانی که خروجی ثابت بشود، خروجی چه گذارهایی دارد؟ آن‌ها را یادداشت کنید. علت این تغییرات چیست؟

^۴ یعنی با ایجاد نمونه (instantiation) از گیت‌ها و اتصال آنها به یکدیگر با سیم



شکل ۲-۷ مدار جمع کننده-تفریق کننده چهار بیتی

۳. در گام آخر این آزمایش لازم است دانشجویان مطابق مراحل ذکر شده در پیوست ۳، کدهای بخش اول و دوم خود را سنتز کرده و آن را بر روی بورد FPGA بارگذاری کنند. ورودی‌های بخش ۲ را با کمک کلیدهای موجود بر روی بورد اعمال کنید و نتایج را مشاهده کنید.

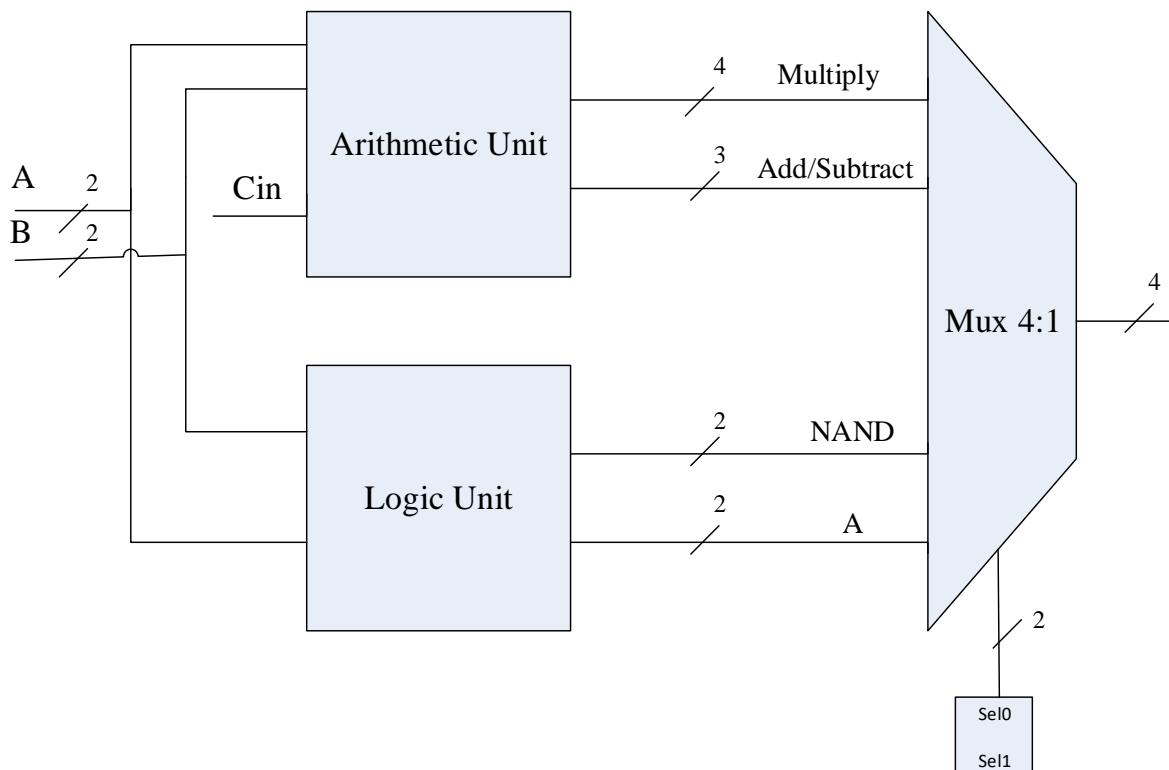
۸- پیاده‌سازی واحد محاسبه و منطق

پیش‌گزارش:

پس از مطالعه دستور کار این آزمایش، توصیف‌های خواسته‌شده را به زبان Verilog بنویسید.

آزمایش:

واحد محاسبه و منطق (ALU) یکی از مهم‌ترین قسمت‌های یک پردازنده است. ALU از سه قسمت محاسبه، منطق و کنترل تشکیل شده است. قسمت محاسباتی اعمالی نظیر جمع، تفریق، ضرب و تقسیم را انجام می‌دهد. در قسمت منطقی اعمالی نظیر NOT، OR، XOR و NOR انجام می‌شود. در نهایت، قسمت کنترل نیز وظیفه تعیین واحد عملیاتی و عملیات مورد نظر را به عهده دارد. در این آزمایش هدف پیاده‌سازی یک ALU بسیار ساده است. شکل ۱-۸ بلوک دیاگرام این ساختار را نشان می‌دهد.



شکل ۱-۸ بلوک دیاگرام یک واحد محاسبه و منطق

جدول ۱-۸ کدهای عملیاتی را که توسط واحد محاسبه و منطق در این آزمایش مورد استفاده قرار می‌گیرد، نشان می‌دهد. در این آزمایش به دلیل محدودیت زمانی از پیاده‌سازی سایر عملگرها صرف نظر می‌شود. عملکرد مورد نظر و همچنین داده‌های ورودی را می‌توان با استفاده از سوییچ‌های (DIP Switch) موجود بر روی بورد تعیین نمود.

جدول ۱-۸ کدهای تعیین‌کننده عملکرد ALU

	Sel1	Sel0
A	0	0
NAND	0	1
ADD/SUB	1	0
MULTIPLY	1	1

جدول ۱-۸ را ابتدا در سطح ساختاری توصیف کنید. اکنون برنامه آزمونی بنویسید تا صحت عملکرد طرح را بررسی کند. بعد از اطمینان از صحت نتایج، برنامه را بر روی بورد FPGA و توسط نمایشگر هفت قسمتی (7-Segment) آزمایش کنید. (ورودی‌ها ۲-بیتی و خروجی ۴-بیتی در نظر گرفته شود).

نکات مهم:

- در خطوط ورودی مالتی‌پلکسر برخی ورودی‌ها تعداد بیت کمتری دارند. برای آنکه تمامی ورودی‌ها ۴-بیتی باشند، بیت پر ارزش در این ورودی‌ها را صفر در نظر بگیرید.
- مالتی‌پلکسر باید در سطح گیت توصیف شود.

توضیحات نحوه عملکرد نمایشگر هفت قسمتی و کدهای لازم جهت راهاندازی این نمایشگر روی بورد FPGA موجود در آزمایشگاه، در پیوست ۴ آمده است.

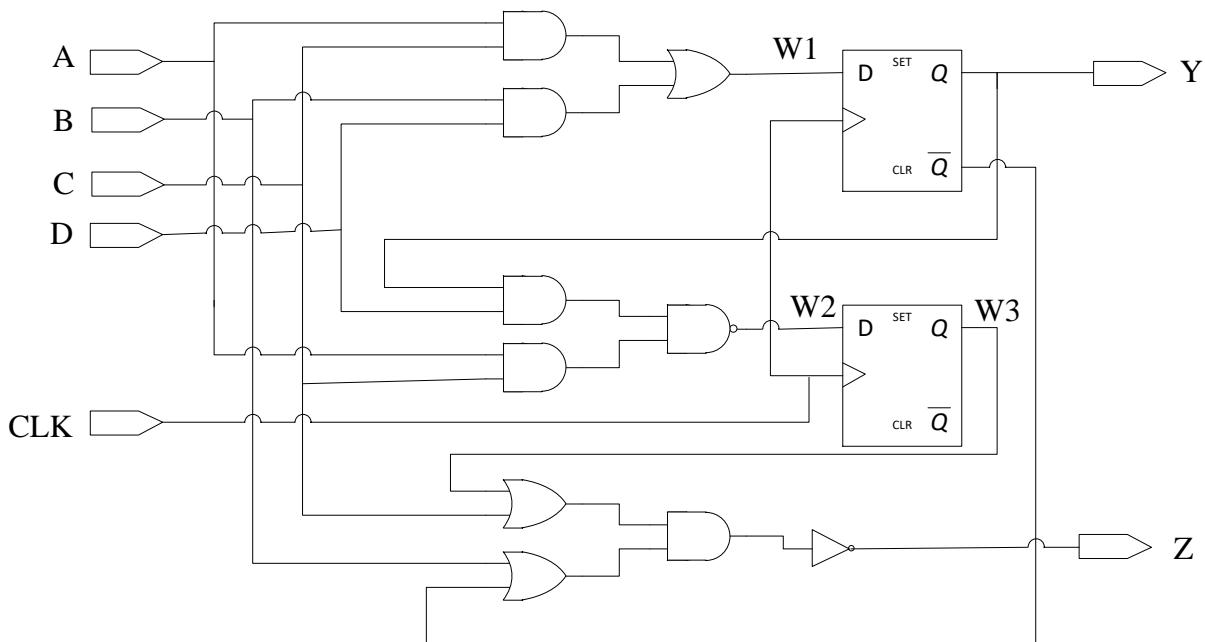
۹- تحلیل و پیاده‌سازی مدارات ترتیبی

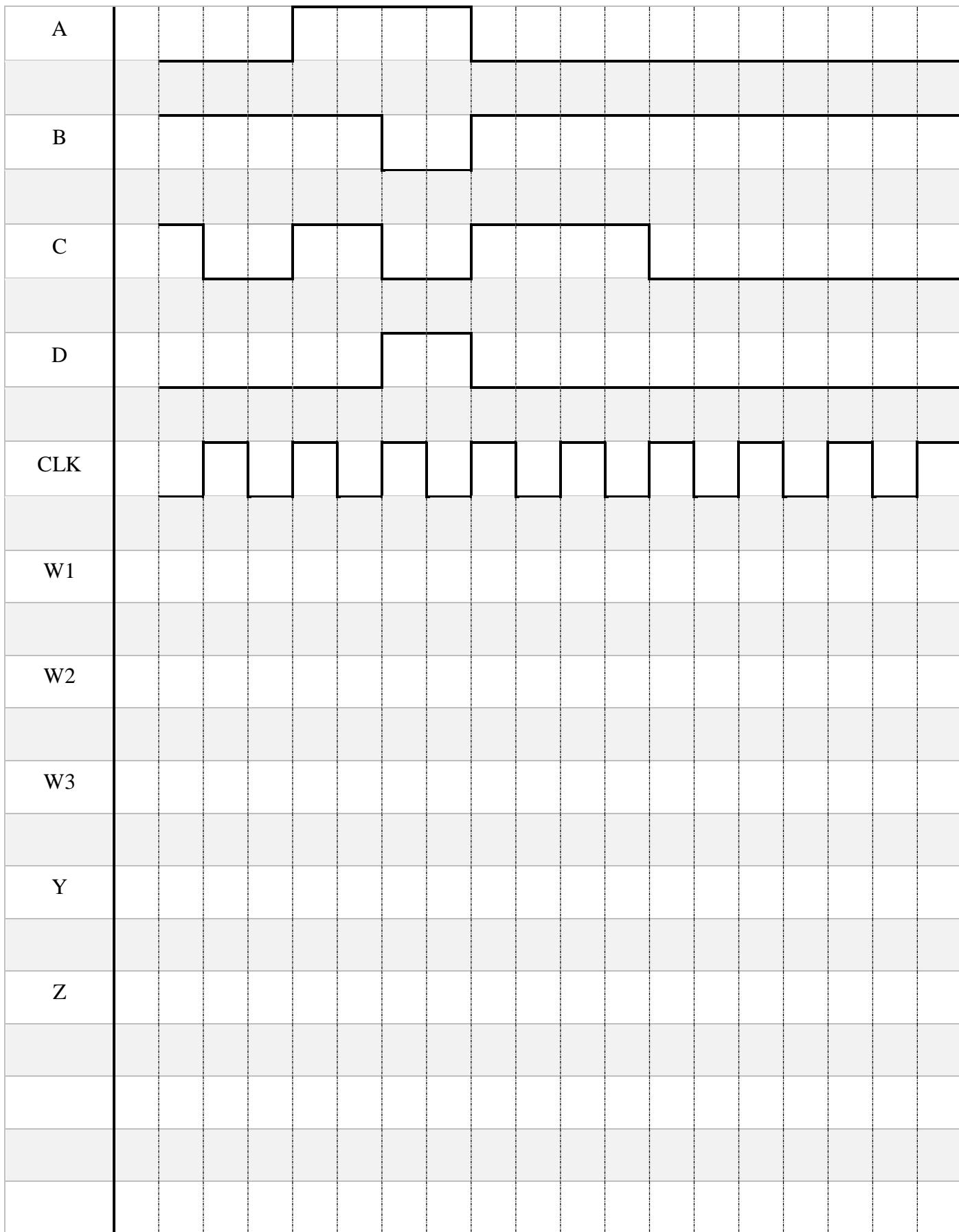
پیش‌گزارش:

پس از مطالعه دستور کار این آزمایش، توصیف‌های خواسته‌شده را به زبان Verilog بنویسید.
برای مدار این آزمایش نمودار زمانی صفحه بعد را با فرض آن که گیت‌های منطقی بدون تاخیر هستند رسم کنید.
توجه: در این آزمایش رسم نمودار زمانی مدار زیر با فرض تاخیر برای گیت‌ها در آزمایشگاه کوئیز گرفته خواهد شد.

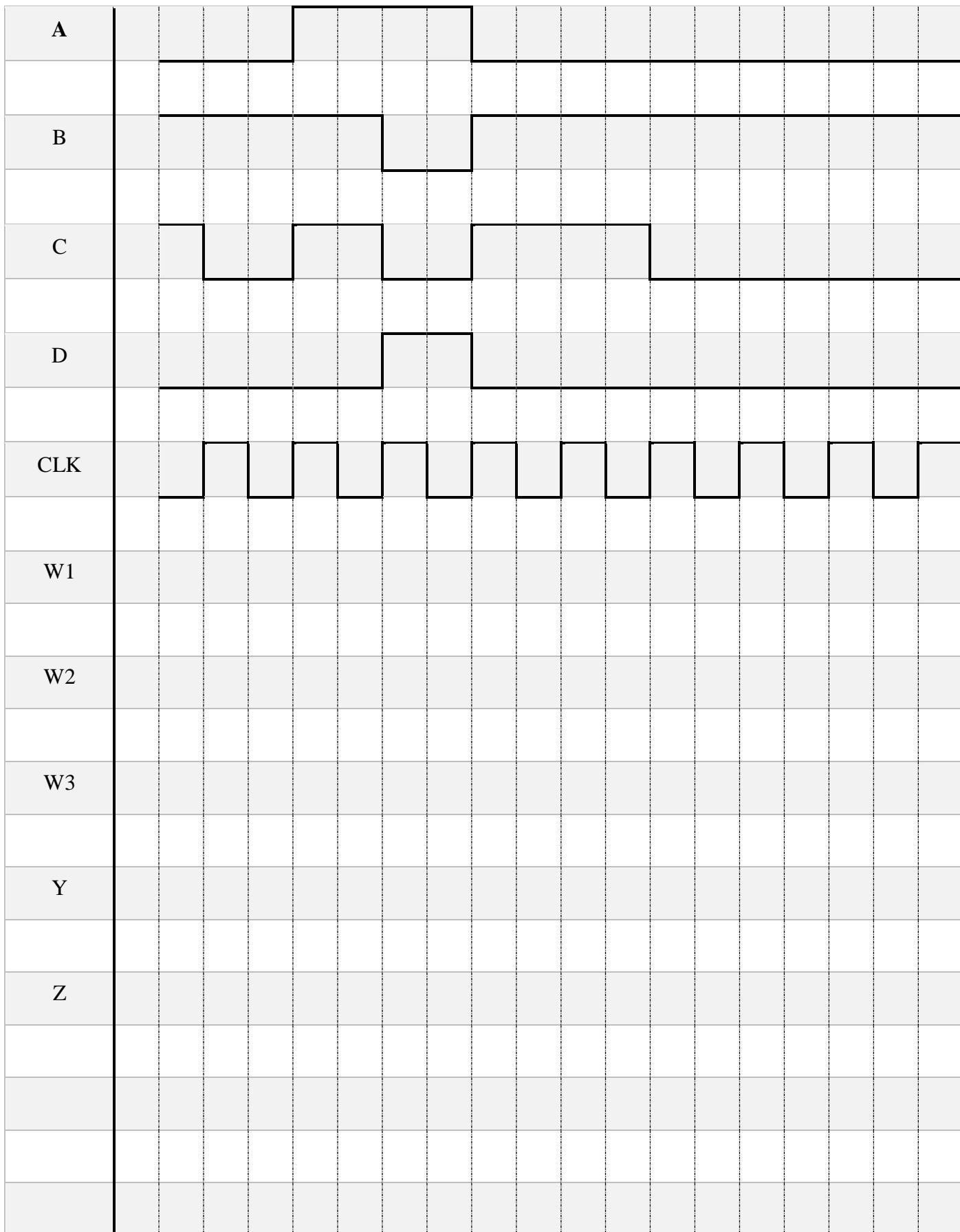
آزمایش:

۱. فلیپ‌فلاب‌ها با لبۀ پائین‌رونده سیگنال ساعت (CLK) کار می‌کنند.
 ۲. اکنون این مدار را با استفاده از زبان Verilog توصیف کنید. برنامه آزمونی بنویسید که ورودی‌هایش مشابه با نمودار داده شده در بخش اول باشد. خروجی شبیه‌سازی را با خروجی گام نخست مقایسه کنید.
 ۳. با استفاده از تأخیرهای داده شده، مدار را شبیه‌سازی کنید و نتیجه شبیه‌سازی را با نتیجه بدست آمده توسط خودتان مقایسه کنید.
- تأخیر هر یک از گیت‌ها ۱۰ نانوثانیه است. دوره تناوب سیگنال ساعت ۸۰ نانوثانیه و تأخیر ورودی ساعت تا خروجی فلیپ‌فلاب ۲۰ نانوثانیه است.





fv



48

۱۰- پیاده‌سازی ماشین حالت

پیش‌گزارش:

دیاگرام حالت ماشین زیر رارسم کنید.

آزمایش:

یک قفل هوشمند را در نظر بگیرید. این قفل دارای یک رمز سه رقمی است. کاربر با الگوی زیر رمز را وارد می‌کند:

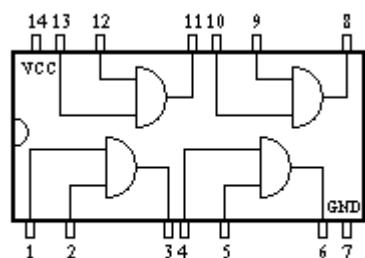
*password#

اگر رمز صحیح باشد قفل باز می‌شود، در غیر اینصورت اگر سه دفعه متوالی رمز اشتباه وارد شود قفل دیگر باز نمی‌شود.
همچنین زمانی که رمز به درستی وارد شود، با الگوی زیر می‌توان رمز آن را تغییر داد:

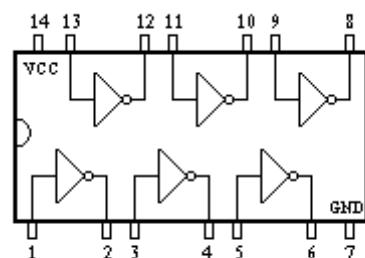
#new_password#

بر اساس دیاگرام حالت طراحی شده در پیش‌گزارش، این قفل را با استفاده از ماشین حالت محدود (FSM) طراحی و با زبان Verilog توصیف نمایید. سپس طراحی خود را شبیه‌سازی کنید.

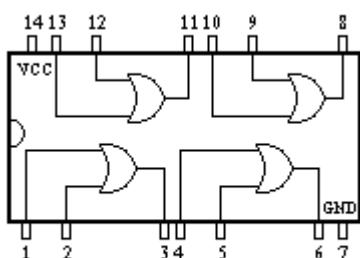
۱۱- پیوست ۱: راهنمای پایه‌های تراشه‌ها



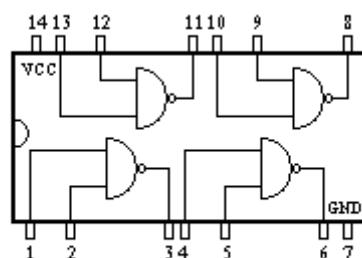
7408



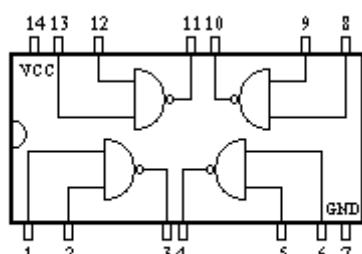
7404



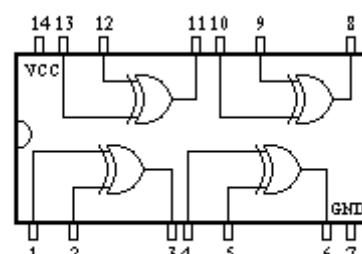
7432



7400



4011



7486

۱۲- پیوست ۲: روش نوشتتن Testbench

به طراح کمک می‌کند بدون اینکه سخت‌افزار واقعی را داشته باشد، رفتار آن را شبیه‌سازی کند. مزیت اصلی شبیه‌سازی این است که می‌توان مقادیر همه سیگنال‌ها را بررسی کرد. برای اینکه بتوانید طراحی خود را شبیه‌سازی کنید، باید برای آن یک Testbench بنویسید.

Testbench چیست؟

در واقع یک فایل HDL (در اینجا Verilog) است، اما با کدی که شما مدار مورد نظرتان را در آن توصیف کرده اید، متفاوت است. این کد در واقع برای این نوشه می‌شود که عملکرد مدار اصلی را با استفاده از آن شبیه‌سازی کرده و خروجی‌ها را بررسی کنیم. کد زیر را در نظر بگیرید.

```
module basic_and ( a, b, out );
    input a, b;
    output out;
    and GATE1(out, a, b);
endmodule
```

این کد عملکرد گیت AND را توصیف می‌کند، برای اینکه بخواهیم عملکرد این گیت را آزمایش کنیم، باید برای این کد یک Testbench بنویسیم، کد زیر نوشته شده برای این گیت است.

```
module basic_and_tb();
    reg a, b;
    wire out;

    basic_and DUT (a, b, out);

    initial begin
        a = 1'b0;
        b = 1'b0;
    #20;
        a = 1'b1;
        b = 1'b0;
    #20;
        a = 1'b1;
        b = 1'b1;
    #20;
        end

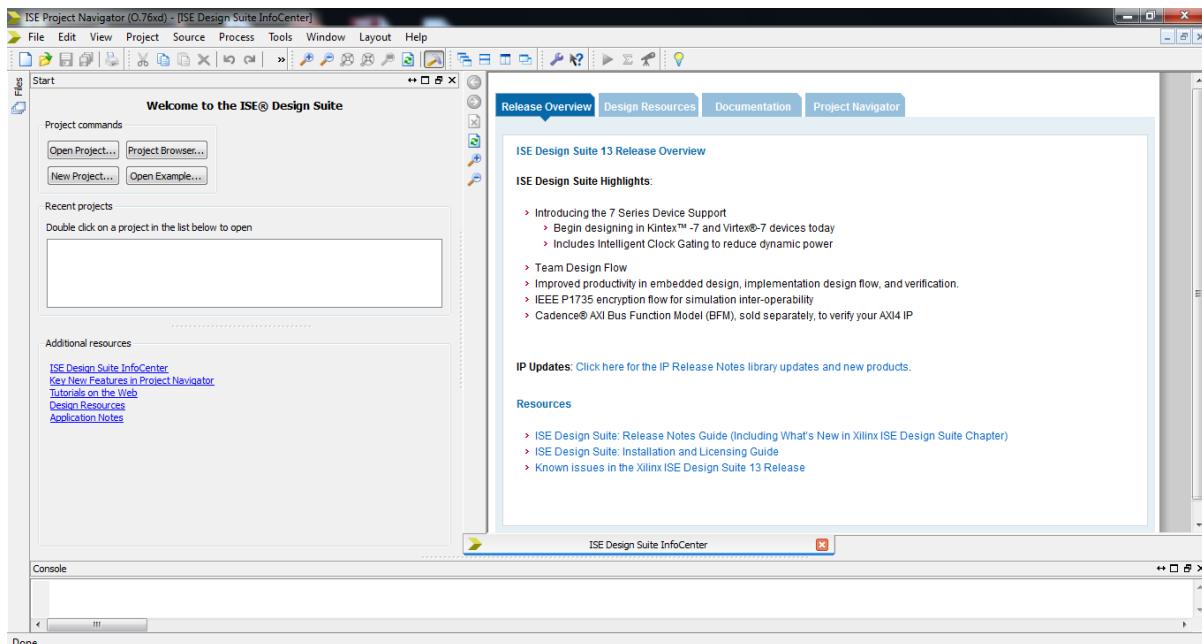
    endmodule
```

همان طور که مشاهده می شود، یک Testbench با اعلان یک ماژول شروع می شود، با این تفاوت که این ماژول هیچ ورودی و خروجی ندارد. بعد از اعلان متغیرهای لازم، ماژولی که قرار است عملکرد آن به وسیله این Testbench بررسی شود را نمونه‌گیری می‌کنیم.^۵ DUT نام رایجی است که برای یک نمونه ماژول انتخاب می‌کنیم. بعد از نمونه‌گیری از ماژول تحت تست، بعد از عبارت کلیدی initial begin، شروع به مقداردهی ورودی‌های مدار می‌کنیم. بعد از اعمال هر ترکیب ورودی یک عبارت به صورت `#20` می‌بینیم، این عبارت در واقع به این معناست که بعد از اعمال هر ترکیب ورودی، به مدت ۲۰ واحد زمانی که در ابتدای کد و با کلمه کلیدی Timescale مشخص شده است، منتظر می‌ماند و بعد از آن ترکیب بعدی را به ورودی اعمال می‌کند.

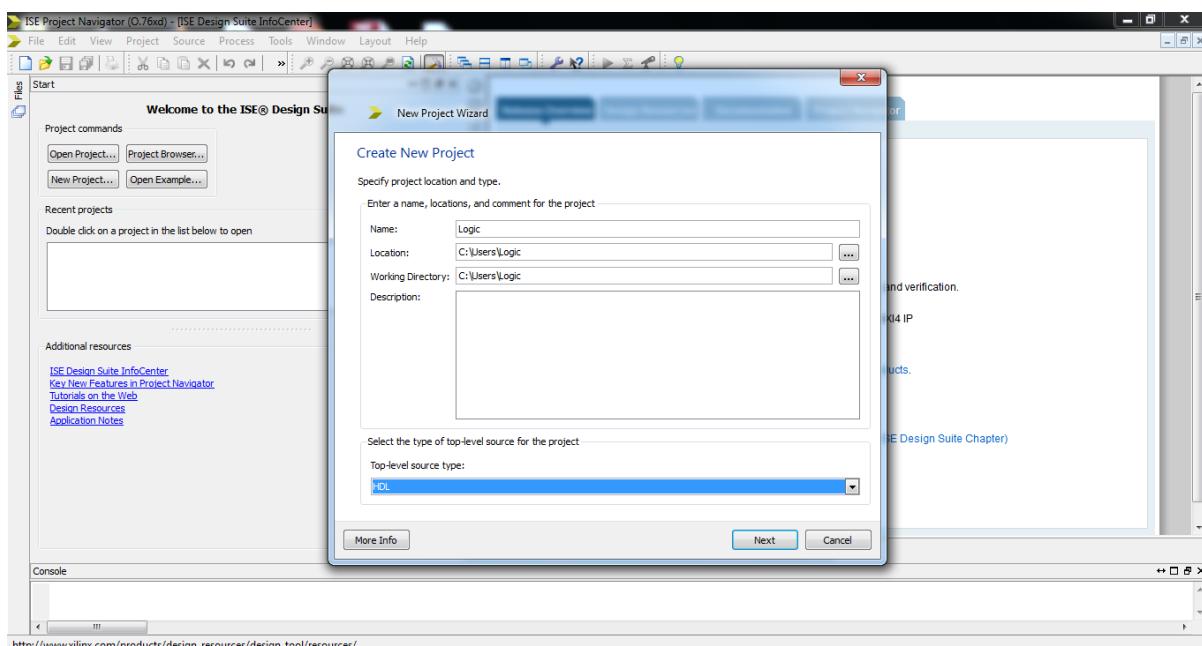
^۵ Design Under Test

۱۳- پیوست ۳: مراحل سنتز و بارگذاری کد توصیف سخت افزاری روی FPGA

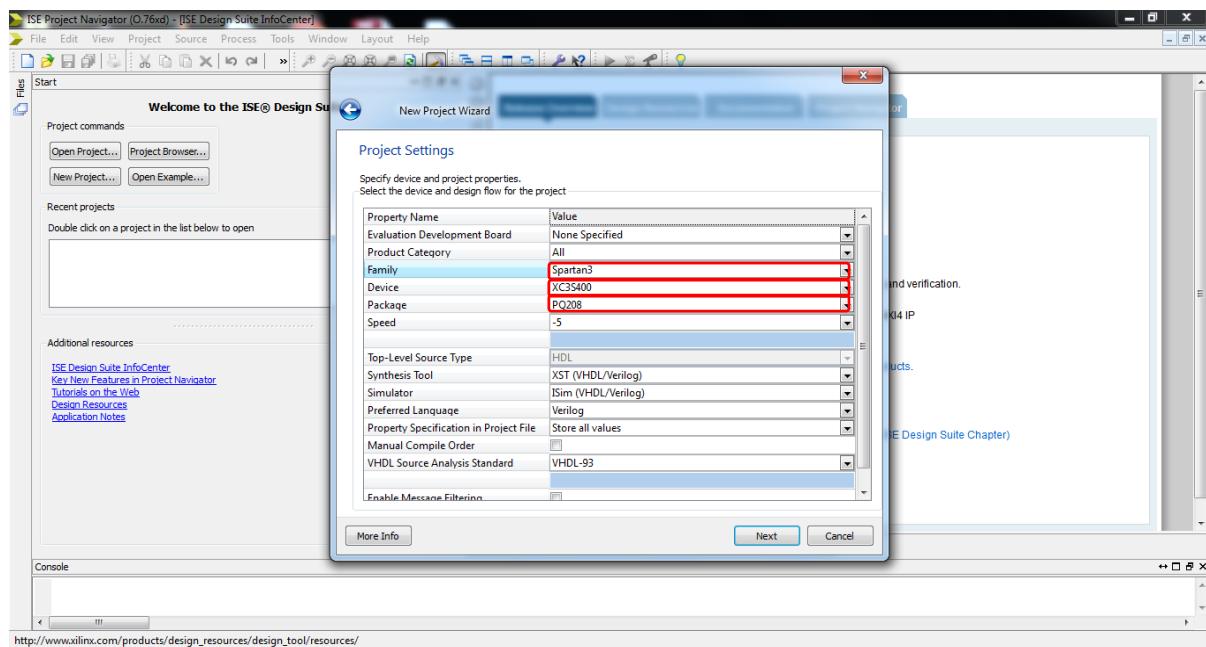
فاز اول: ایجاد پروژه



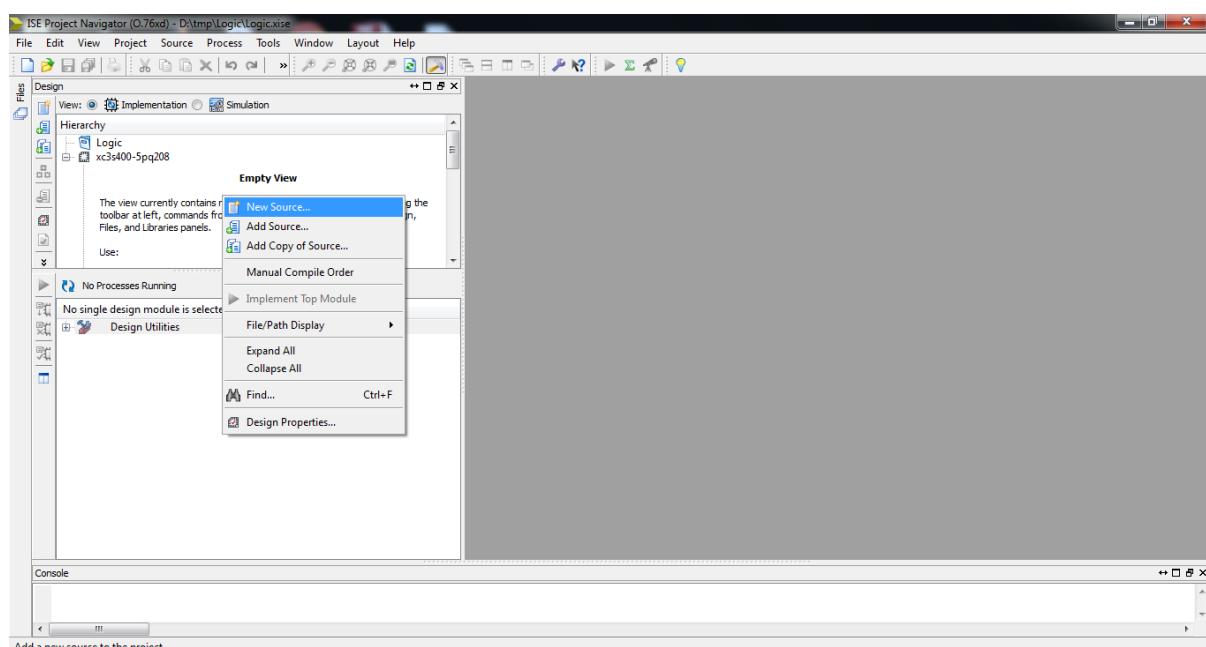
شکل ۱-۱۳ صفحه شروع کار با ابزار ISE



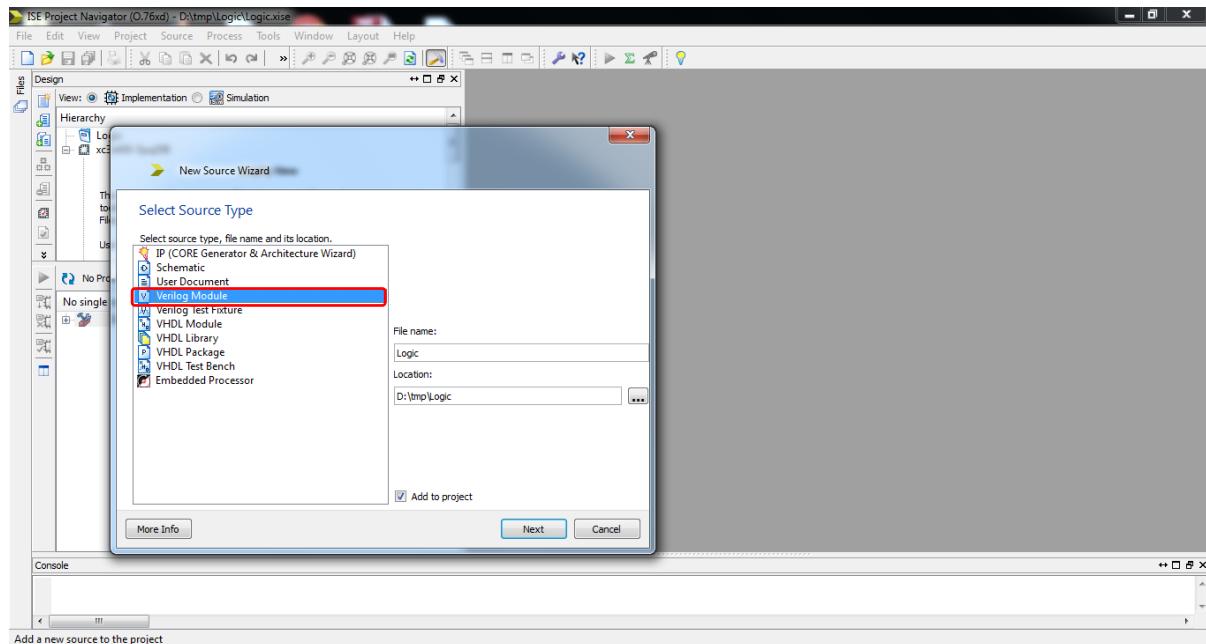
شکل ۱-۱۴ ایجاد یک پروژه جدید



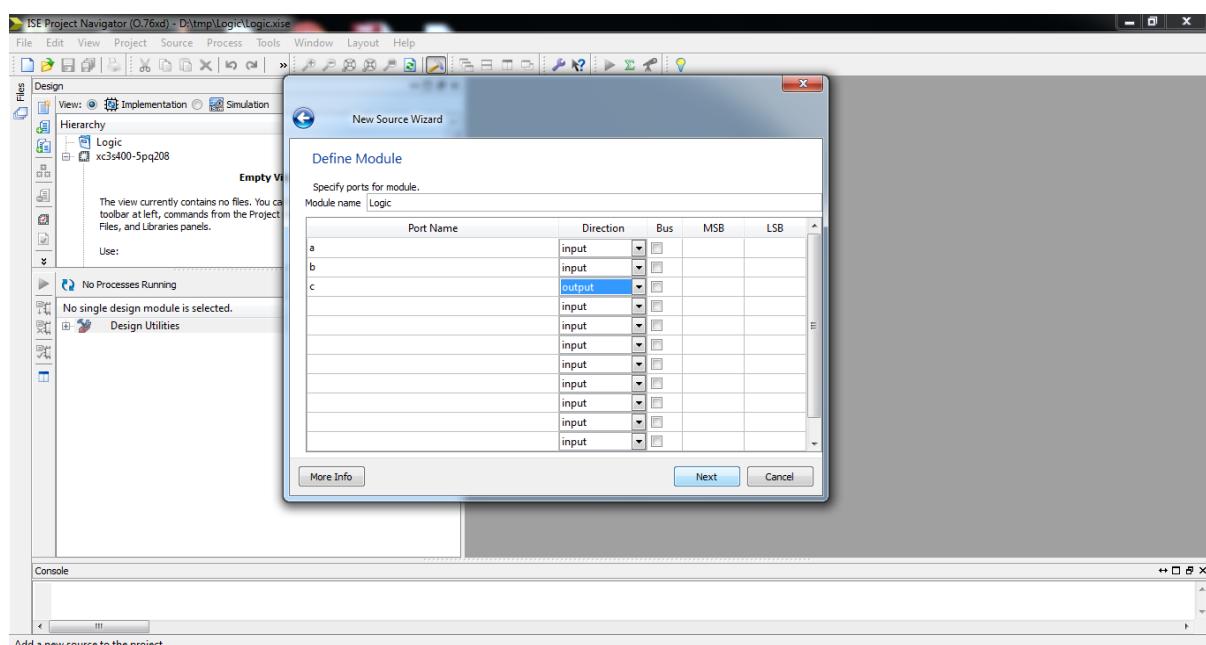
شکل ۳-۱۳ تعیین ویژگی های بورد FPGA



شکل ۴-۱۳ ایجاد یک فایل در پروژه



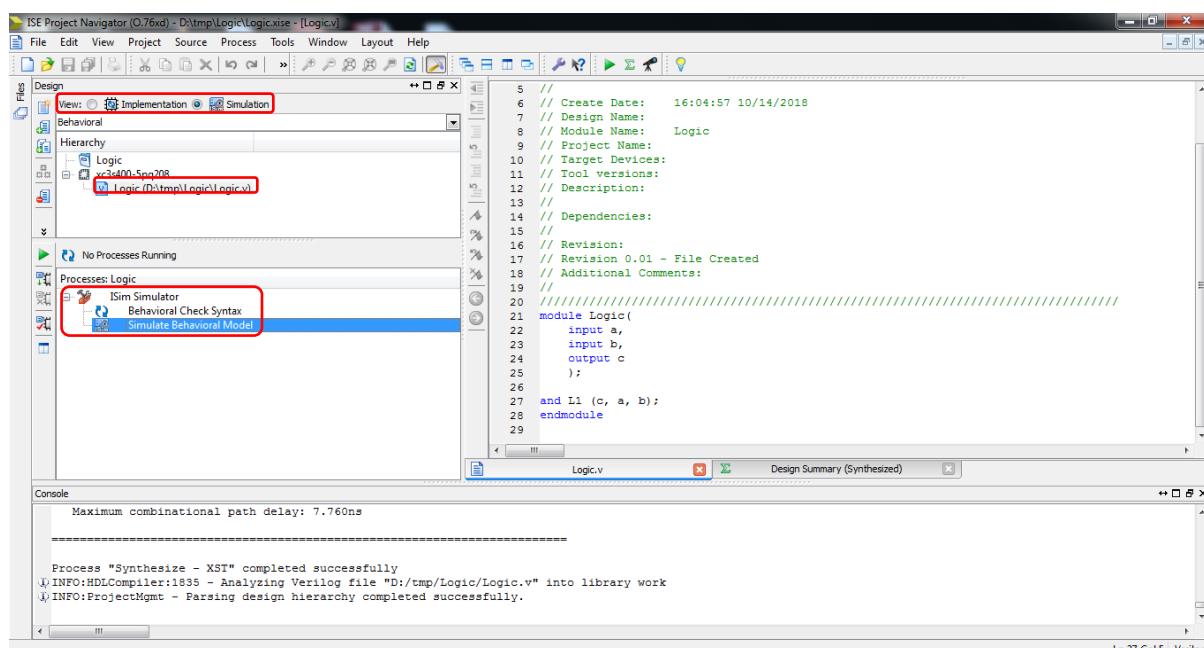
شکل ۵-۱۳ تعیین نوع و نام فایل



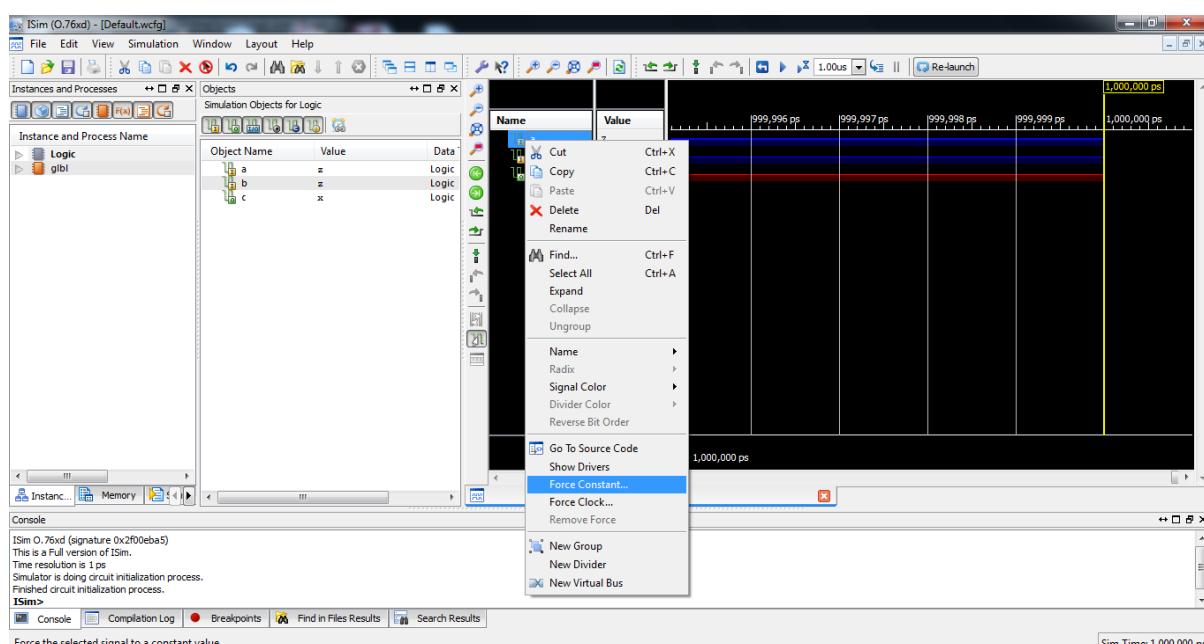
شکل ۶-۱۳ تعیین پورت‌های ورودی و خروجی

- تعیین پورت‌های ورودی و خروجی در این بخش الزامی نیست.

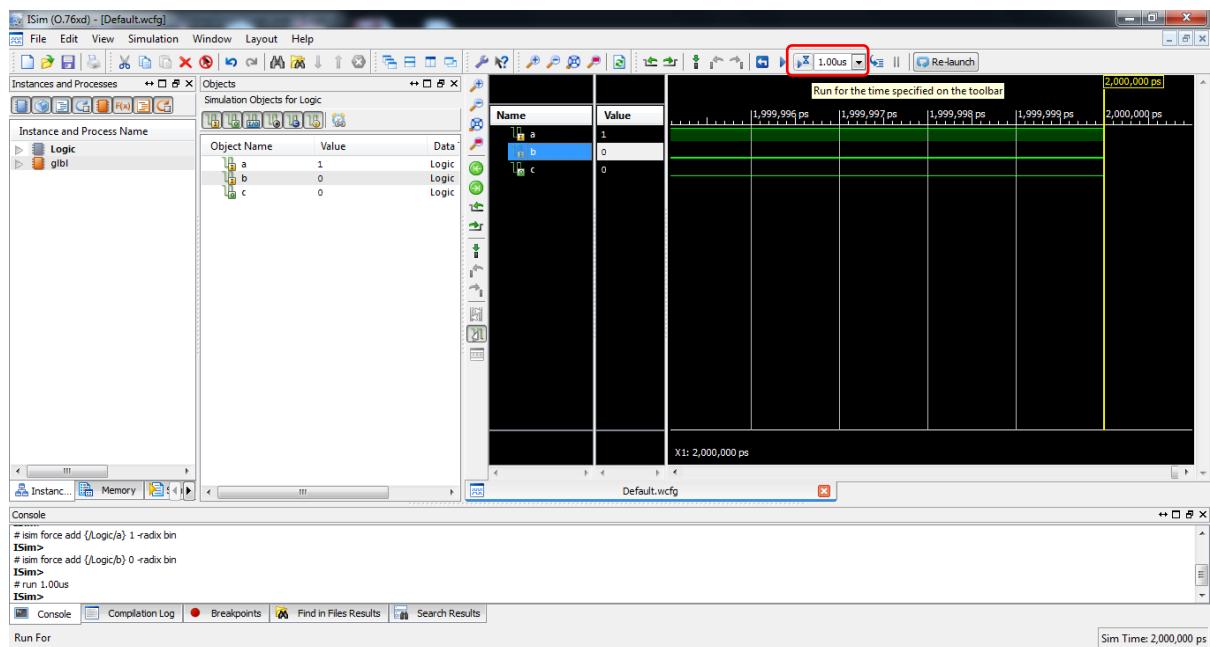
فاز دوم: شبیه‌سازی کد



شکل ۷-۱۳ منوی شبیه‌سازی با

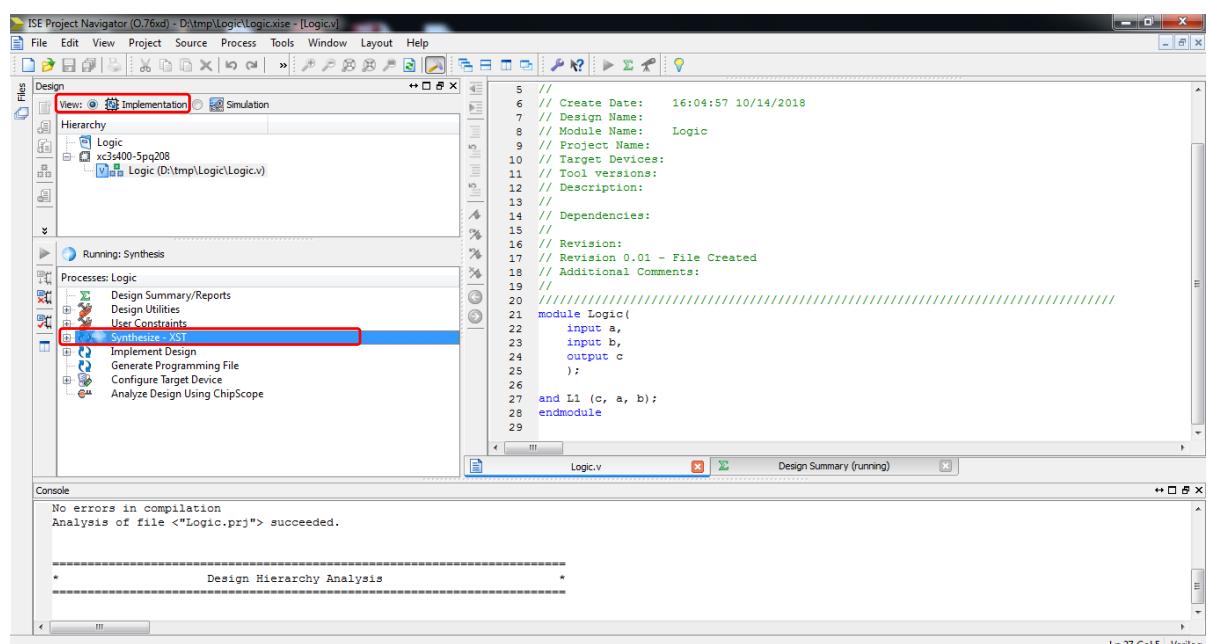


شکل ۸-۱۳ اعمال مقدار ثابت به سیگنال‌ها (هنگامی که testbench نداریم)



شکل ۹-۱۳ نتیجه شبیه‌سازی

فاز سوم: سنتز و برنامه‌ریزی بورد



شکل ۱۰-۱۳ منوی سنتز

The screenshot shows the Xilinx ISE Project Navigator interface. The project is named "Logic" and is based on the xc3s400-5pq208 device. The "Design" tab is selected, showing the "Implementation" view. In the "Processes: Logic" pane, the "Synthesize - XST" option is highlighted with a red box. The "Console" pane displays the synthesis command and its execution results.

```

5 // Create Date: 16:04:57 10/14/2018
6 // Design Name:
7 // Module Name: Logic
8 // Project Name:
9 // Target Devices:
10 // Tool versions:
11 // Description:
12 // Dependencies:
13 //
14 // Revision:
15 // Revision 0.01 - File Created
16 // Additional Comments:
17 //
18 /////////////////////////////////
19 module Logic(
20     input a,
21     input b,
22     output c
23 );
24
25     and L1 (c, a, b);
26
27 endmodule
28
29

```

Console output:

```

Fuse Memory Usage: 108436 KB
Fuse CPU Usage: 358 ms
Launching ISim simulation engine GUI...
"D:\tmp\Logic\Logic_isim_beh.exe" -intstyle ise -gui -tclbatch isim.cmd -wdb "D:/tmp/Logic/Logic_isim_beh.wdb"
ISim simulation engine GUI launched successfully

Process "Simulate Behavioral Model" completed successfully

```

شکل ۱۱-۱۳ نتیجه سنتز

The screenshot shows the Xilinx ISE Project Navigator interface. The project is named "Logic" and is based on the xc3s400-5pq208 device. The "Design" tab is selected, showing the "Implementation" view. In the "Processes: Logic" pane, the "I/O Pin Planning (PlanAhead) - Post-Synthesis" option is highlighted with a red box. The "Console" pane displays the pin planning command and its preparation.

```

5 // Create Date: 16:04:57 10/14/2018
6 // Design Name:
7 // Module Name: Logic
8 // Project Name:
9 // Target Devices:
10 // Tool versions:
11 // Description:
12 // Dependencies:
13 //
14 // Revision:
15 // Revision 0.01 - File Created
16 // Additional Comments:
17 //
18 /////////////////////////////////
19 module Logic(
20     input a,
21     input b,
22     output c
23 );
24
25     and L1 (c, a, b);
26
27 endmodule
28
29

```

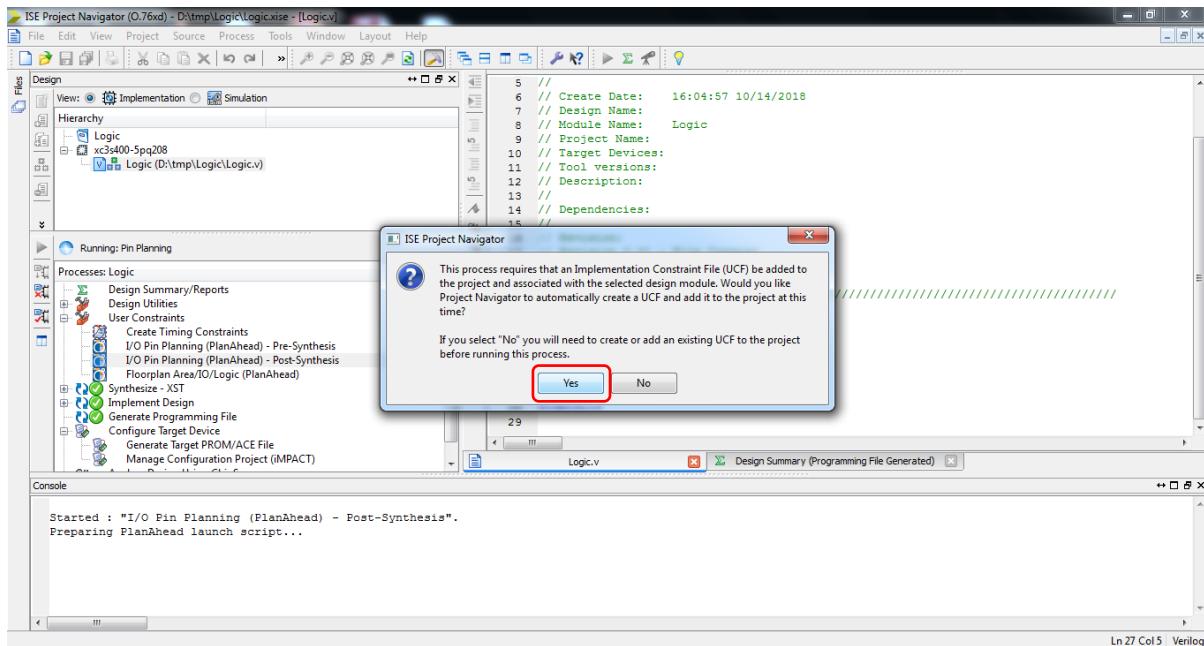
Console output:

```

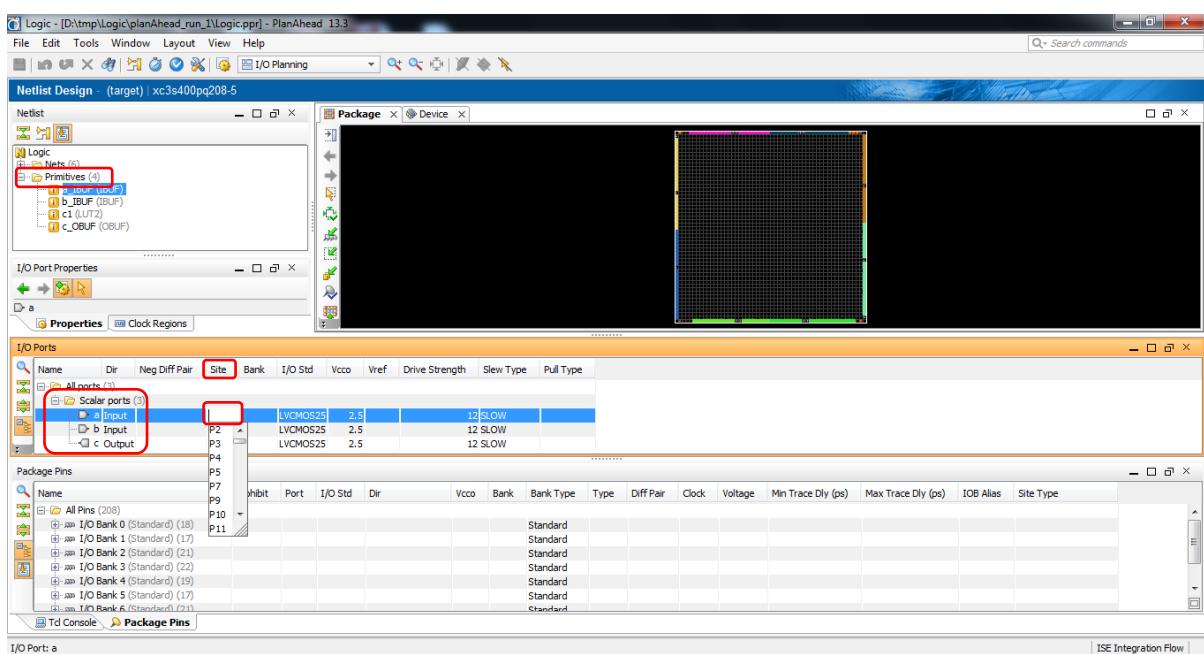
Started : "I/O Pin Planning (PlanAhead) - Post-Synthesis".
Preparing PlanAhead launch script...

```

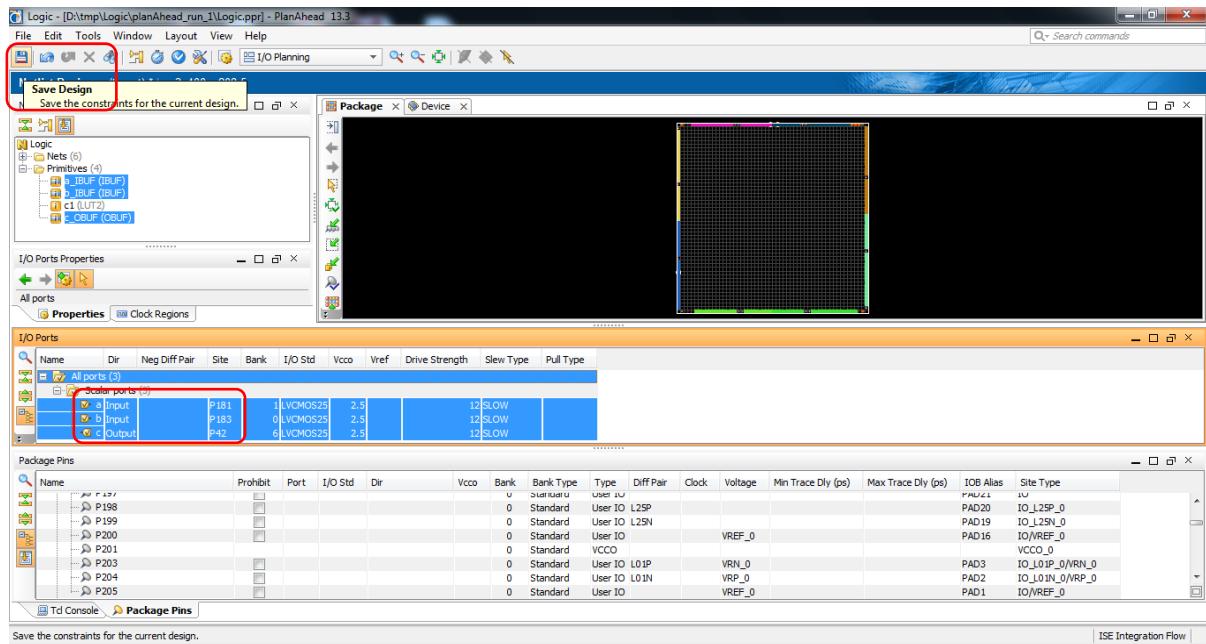
شکل ۱۲-۱۳ اجرای برنامه تخصیص پایه‌های ورودی و خروجی



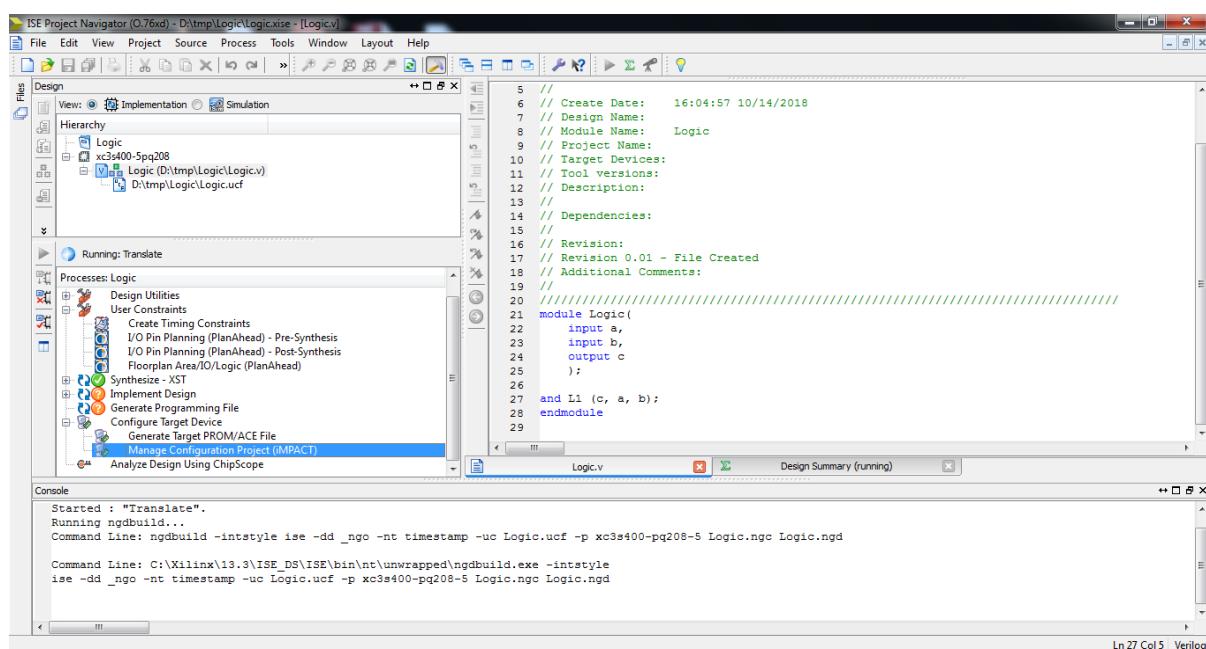
شکل ۱۳-۱۳ تایید اجرای برنامه تخصیص پایه‌های ورودی و خروجی



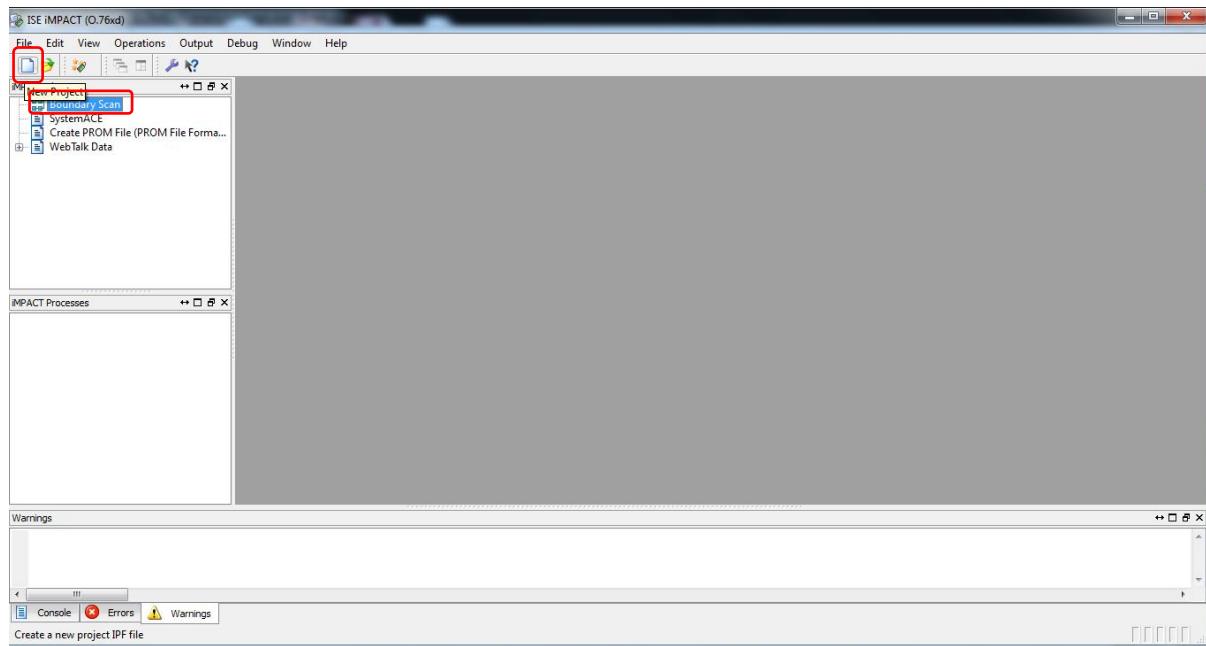
شکل ۱۴-۱۳ منوی تخصیص پایه‌های ورودی و خروجی



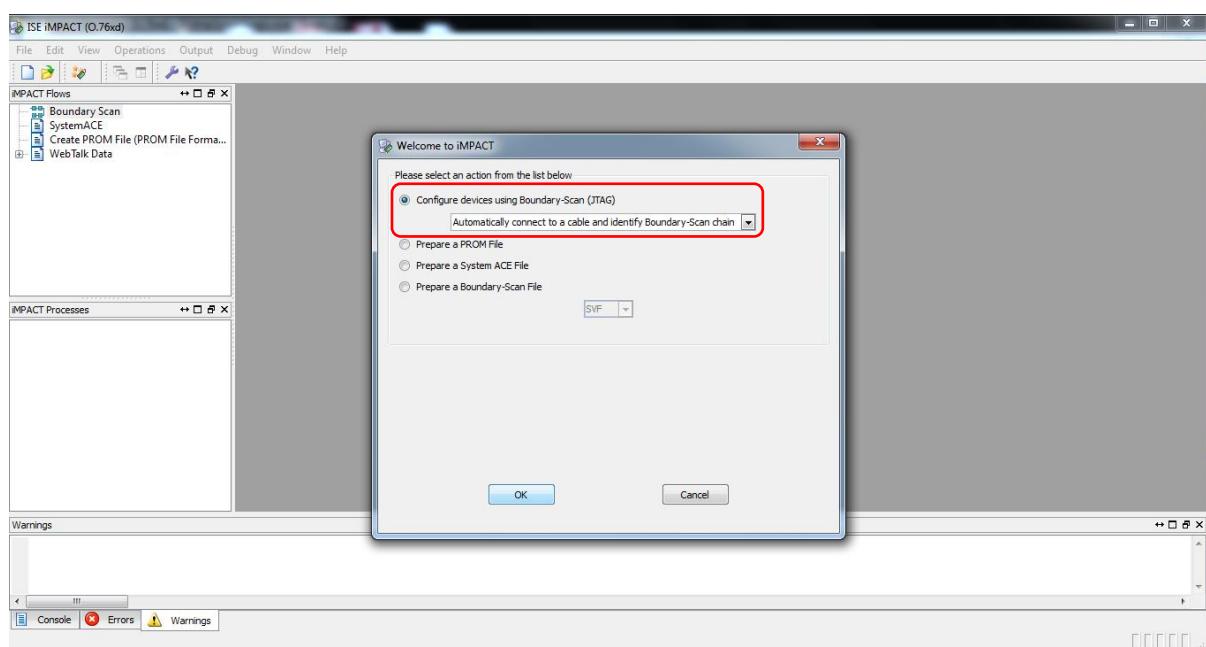
شکل ۱۵-۱۳ تخصیص پایه‌های ورودی و خروجی



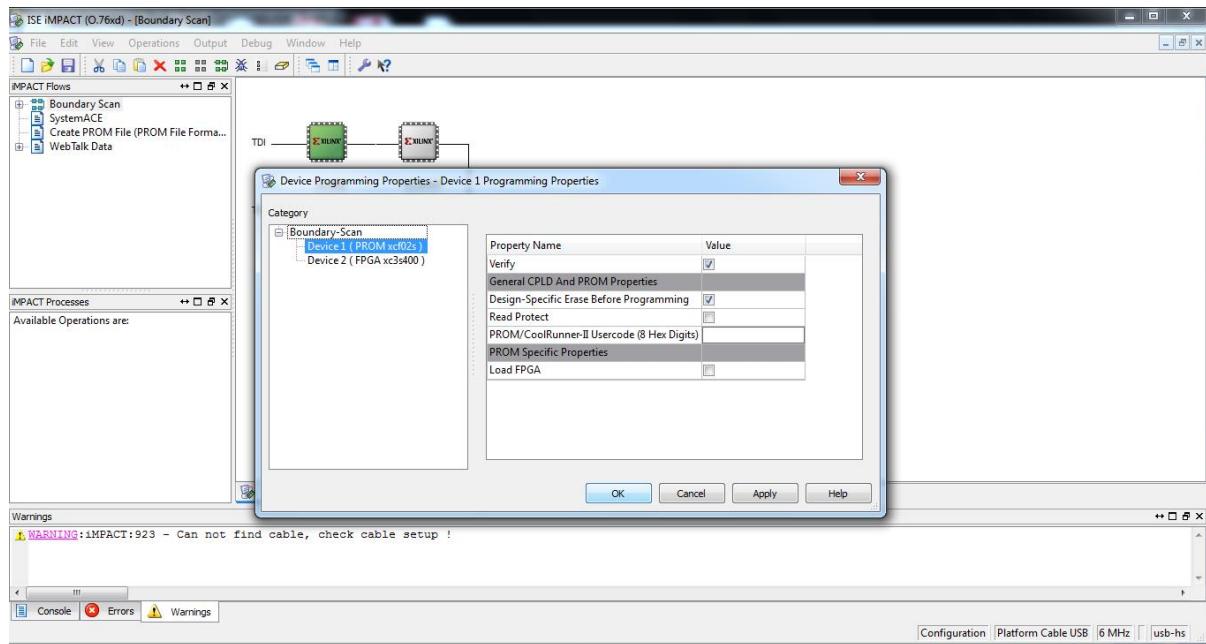
شکل ۱۶-۱۳ ایجاد فایل برنامه‌ریزی



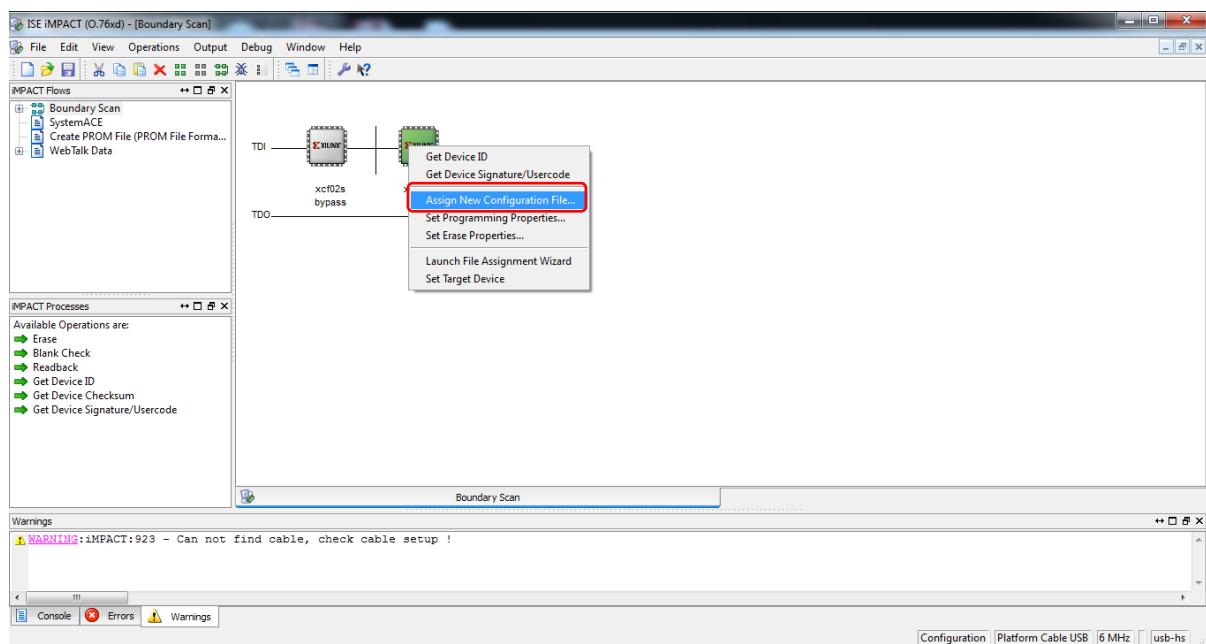
شکل ۱۷-۱۳ ایجاد یک پروژه برنامه‌ریزی بورد FPGA



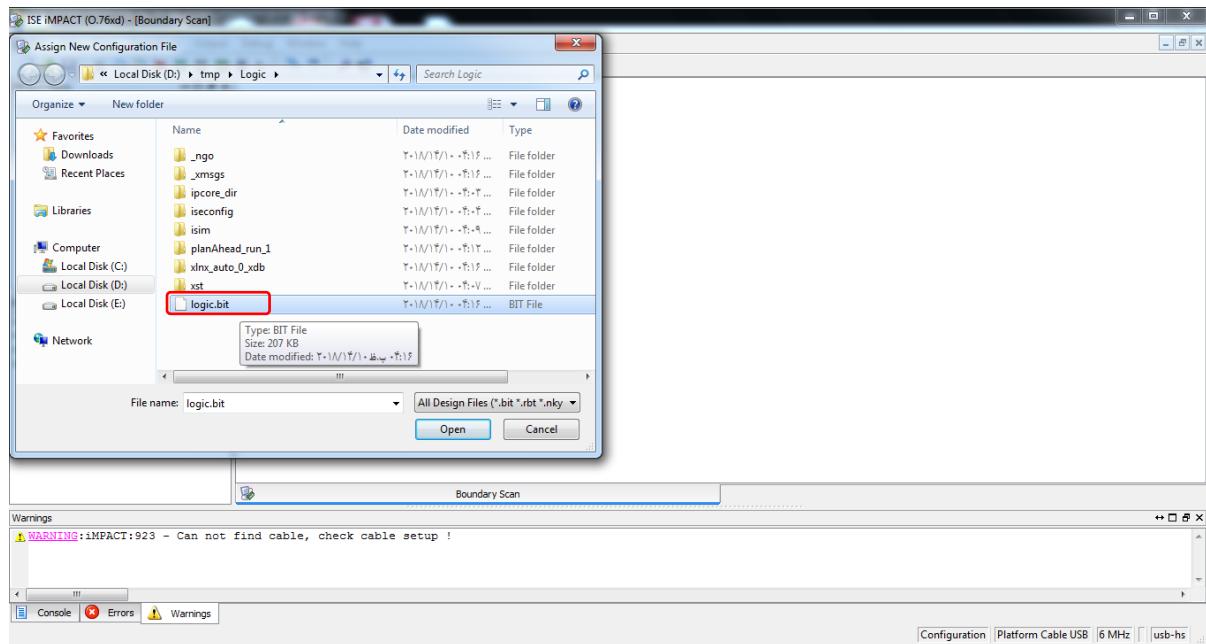
شکل ۱۸-۱۳ تایید نوع برنامه‌ریزی



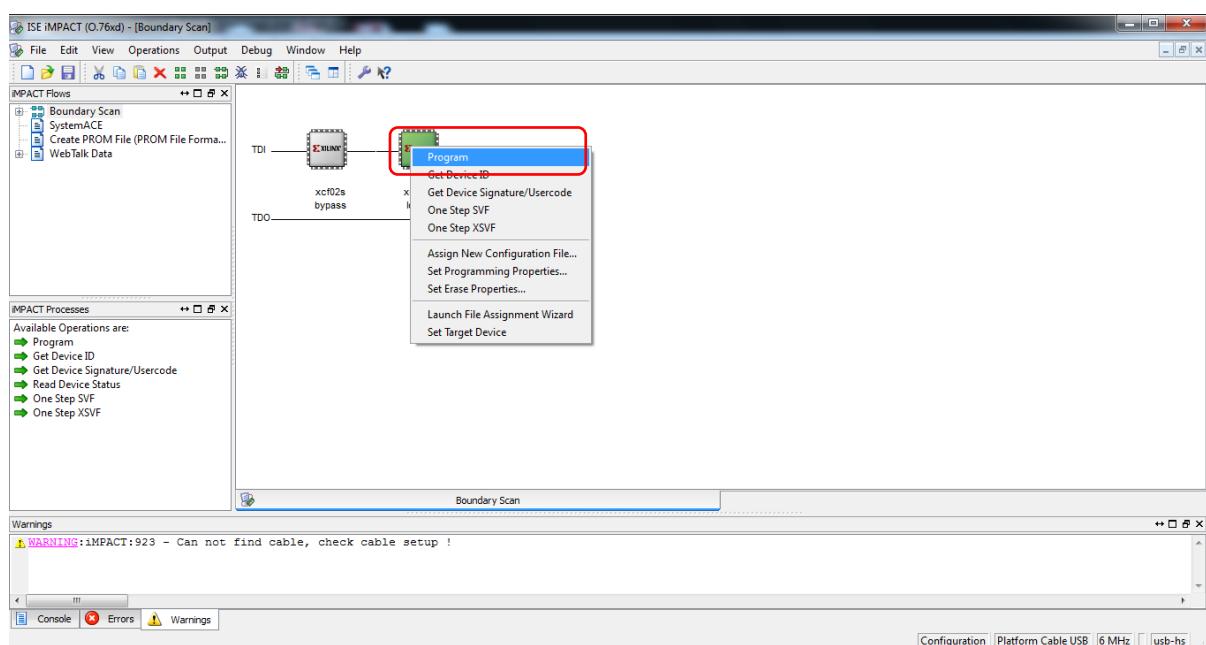
شکل ۱۹-۱۳ انصراف از برنامه‌ریزی بخش حافظه بورد FPGA



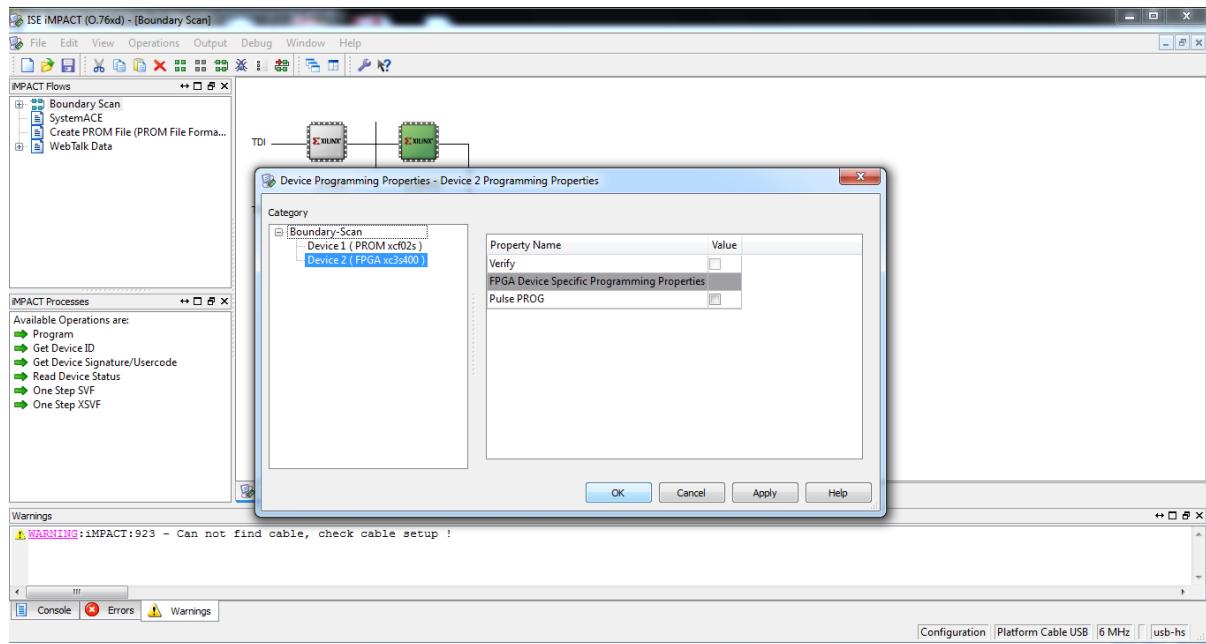
شکل ۲۰-۱۳ افزودن فایل برنامه‌ریزی مناسب



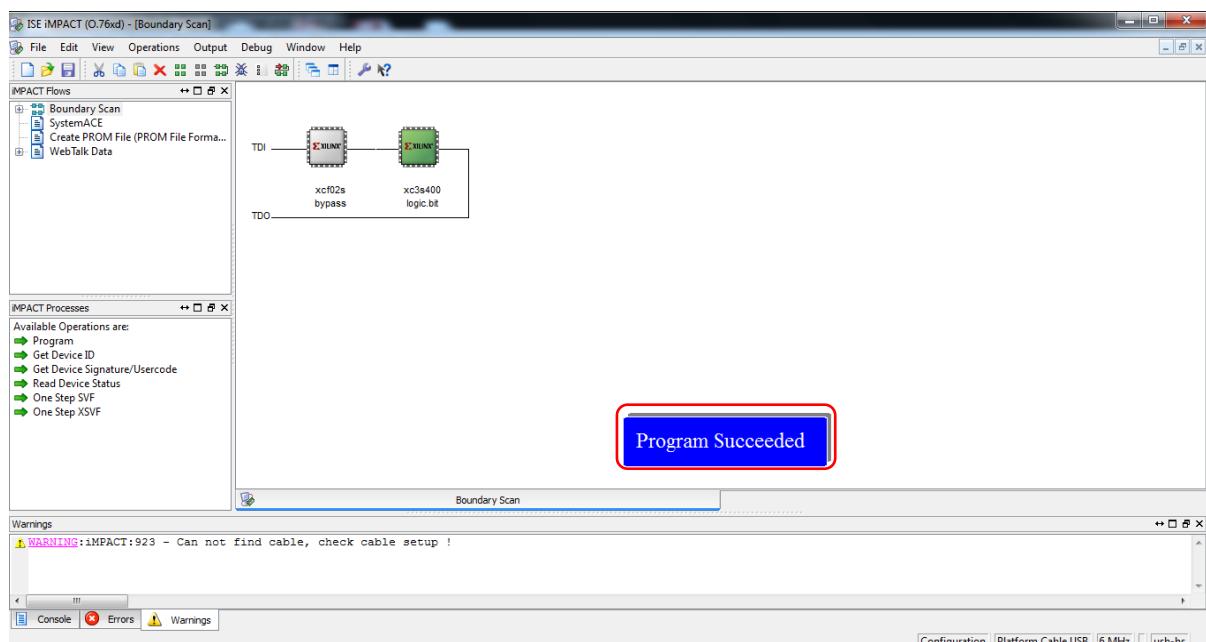
شكل ۲۱-۱۳ انتخاب فایل مناسب



شكل ۲۲-۱۳ برنامه ریزی FPGA



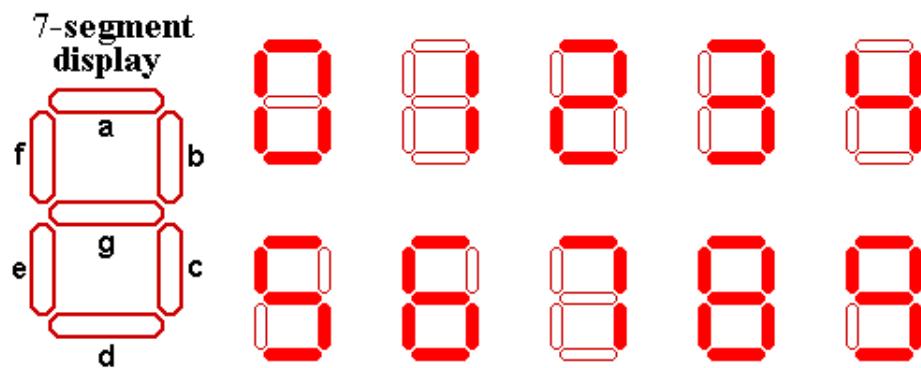
شکل ۲۳-۱۳ تأیید نهایی



شکل ۲۴-۱۳ وضعیت برنامه ریزی

پیوست ۴: کار با نمایشگر هفت قسمتی

نمایشگر هفت قسمتی، از هفت قطعه فتوالکتریک، مانند کریستال مایع یا LED ساخته می‌شود. این قطعات به گونه‌ای که در شکل ۱ مشاهده می‌شود در کنار هم قرار داده شده‌اند. با روشن کردن برخی از این LEDها می‌توان اعداد و همچنین برخی از کاراکترها را نمایش داد.



شکل ۱: چگونگی نمایش اعداد با [1] 7-seg

Binary Inputs				Decoder Outputs							7-Segment Display Outputs	
D	C	B	A	a	b	c	d	e	f	g		
0	0	0	0	1	1	1	1	1	1	0		0
0	0	0	1	0	1	1	0	0	0	0		1
0	0	1	0	1	1	0	1	1	0	1		2
0	0	1	1	1	1	1	1	0	0	1		3
0	1	0	0	0	1	1	0	0	1	1		4
0	1	0	1	1	0	1	1	0	1	1		5
0	1	1	0	1	0	1	1	1	1	1		6
0	1	1	1	1	1	0	0	0	0	0		7
1	0	0	0	1	1	1	1	1	1	1		8
1	0	0	1	1	1	1	0	1	1	1		9

شکل ۲: نگاشت مقادیر BCD به [2] 7-seg

ماژول مبدل BCD به 7-Segment

```
module bcd2seven_seg(a,b,cin, sel, SEG_SEL,SEG_DATA);

input [1:0]a;
input [1:0]b;
input [1:0]sel;
input cin;
output [4:0]SEG_SEL;
output [7:0]SEG_DATA;
reg [7:0]SEG_DATA;
wire [3:0]mux_out;

always @(mux_out)
begin
    case(mux_out)
        0:SEG_DATA = 8'b00111111;
        1:SEG_DATA = 8'b00000110;
        2:SEG_DATA = 8'b01011011;
        3:SEG_DATA = 8'b01001111;
        4:SEG_DATA = 8'b01100110;
        5:SEG_DATA = 8'b01101101;
        6:SEG_DATA = 8'b01111101;
        7:SEG_DATA = 8'b00000111;
        8:SEG_DATA = 8'b01111111;
        9:SEG_DATA = 8'b01101111;
    endcase
end

assign SEG_SEL=5'b00001;

//MODULE INSTANTIATIONS IN BASED ON TABLE 1
endmodule
```

#PlanAhead Generated physical constraints

NET "SEG_DATA[0]" LOC = P10;

NET "SEG_DATA[1]" LOC = P7;

NET "SEG_DATA[2]" LOC = P11;

NET "SEG_DATA[3]" LOC = P5;

NET "SEG_DATA[4]" LOC = P4;

NET "SEG_DATA[5]" LOC = P12;

NET "SEG_DATA[6]" LOC = P9;

NET "SEG_DATA[7]" LOC = P3;

NET "SEG_SEL[0]" LOC = P15;

NET "SEG_SEL[1]" LOC = P20;

NET "SEG_SEL[2]" LOC = P19;

NET "SEG_SEL[3]" LOC = P18;

NET "SEG_SEL[4]" LOC = P16;

NET "a[0]" LOC = P161;

NET "a[1]" LOC = P162;

NET "b[0]" LOC = P172;

NET "b[1]" LOC = P175;

NET "sel[0]" LOC = P182;

NET "sel[1]" LOC = P181;

NET "cin" LOC = P183;

منابع:

- [1] “7-Segment Display.” [Online]. Available: <http://www.physics.udel.edu/~watson/scen103/7seg.html>.
- [2] “BCD to 7-Segment Decoder.” [Online]. Available: <http://macao.communications.museum/eng/exhibition/secondfloor/MoreInfo/Displays.html>.