



دانشگاه سیستان و بلوچستان  
دانشکده مهندسی برق و کامپیوتر

## آموزش استفاده از نرم افزار Xilinx ISE Design Suite v14.7

تهیه و تنظیم:

علی عباسی

این آموزش در دانشگاه سیستان و بلوچستان جهت اهداف آموزشی، غیر تجاری و استفاده‌ی رایگان تهیه گردیده است. تمامی علائم تجاری، لوگوها، و نرم‌افزارهای مورد استفاده در این آموزش جزء دارایی‌های شرکت Xilinx یا دیگر شرکای تجاری آن می‌باشد.

نسخه‌ی فایل	تاریخ آخرین ویرایش
۱/۸	بهمن ۱۳۹۹

با تشکر ویژه از دکتر محمدحسین سرگلزائی، مدیریت گروه مهندسی کامپیوتر دانشگاه سیستان و بلوچستان، جهت کمک در تهیه و انتشار این محتوای آموزشی.

## فهرست مطالب

۱.....	مقدمه و معرفی نرم افزار
۲.....	نصب نرم افزار
۹.....	آشنایی با محیط نرم افزار
۱۰.....	ایجاد یک پروژه جدید
۱۳.....	اضافه کردن Source به پروژه
۱۷.....	سنتز
۲۲.....	پیاده سازی (ترجمه - جایابی - مسیریابی)
۲۳.....	شبیه سازی رفتاری
۲۶.....	شبیه سازی زمانی
۲۷.....	استفاده از نمایشگر شکل موج نرم افزار Xilinx ISIM
۳۰.....	گزارش طراحی
۳۱.....	تولید فایل برنامه ریزی
۳۲.....	برنامه ریزی تراشه به کمک نرم افزار ISE iMPACT

## ۱. مقدمه و معرفی نرم افزار

شرکت های تولید کننده تراشه های برنامه پذیر، برای سنتز طرح ها و برنامه ریزی تراشه های خود، نرم افزارهای خاصی را تولید و معرفی می کنند. دو مورد از معروف ترین شرکت های تولید کننده تراشه های برنامه پذیر<sup>۱</sup> FPGA، شرکت های Altera و Xilinx هستند که هر کدام از آن ها نرم افزارهای انحصاری خود را برای این کار دارند. به عنوان مثال، اگر شما تصمیم به برنامه ریزی یک تراشه ی FPGA از سری Cyclone 4 که توسط شرکت Altera تولید می شود را دارید، می بایست از نرم افزار تولید شده توسط این شرکت یعنی Quartus استفاده نمایید.

مشابه شرکت Altera، شرکت Xilinx نیز نرم افزار انحصاری خودش را برای برنامه ریزی تراشه های تولید کرده است. نرم افزار ISE<sup>۲</sup> Design Suite، توسط شرکت Xilinx برای برنامه ریزی تراشه های FPGA، CPLD<sup>۳</sup>، و برخی دیگر تراشه های برنامه پذیر این شرکت از سال ۲۰۱۰ میلادی به بازار عرضه شد و آخرین نسخه ی آن در اواخر سال ۲۰۱۳ میلادی با شماره نسخه ی ۱۴.۷ در دسترس عموم قرار گرفت. از سال ۲۰۱۲، شرکت Xilinx نرم افزار دیگری مشابه ISE تحت عنوان Vivado را نیز عرضه کرد و پس از آن در سال ۲۰۱۳ توسعه ی نرم افزار ISE را متوقف نمود. تفاوت اصلی این دو نرم افزار، جدا از توانمندی بیشتر نرم افزار Vivado به عنوان یک ابزار سنتز، در تراشه هایی است که توسط آن ها پشتیبانی می شود. به صورت کلی، می توان گفت نرم افزار Vivado تنها از سری هفتم تراشه های FPGA شرکت Xilinx مانند Spartan 7، Kintex 7 و برخی اعضای خانواده های UltraScale پشتیبانی می کند (در سال ۲۰۱۹ که این متن نوشته شده است)، در حالی که پشتیبانی از خانواده های قدیمی تر FPGA های این شرکت همچنان بر عهده ی نرم افزار ISE می باشد.

در آموزش پیش رو به توضیح موارد ابتدایی استفاده از نرم افزار Xilinx ISE Design Suite پرداخته خواهد شد. اگرچه این آموزش بر اساس نسخه ی ۱۴.۷ تهیه گردیده است، با این حال می تواند برای استفاده از نسخه های پیشین این نرم افزار نیز مفید باشد. نسخه فوق قابلیت پشتیبانی از Xilinx System Generator for DSP v14.7 را نیز داشته که به کاربران امکان تولید بلوک های آماده ی مربوط به پردازش سیگنال های دیجیتال را می دهد. بلوک های تولید شده با این ابزار امکان شبیه سازی و تحلیل در نرم افزار MathWorks MATLAB و در نسخه های 2012a، 2012b، 2013a، و 2013b از آن را خواهند داشت. توجه داشته باشید که برای استفاده از این قابلیت باید ابزارهای Simulink و Simulink Fixed-Point Designer به همراه نرم افزار MATLAB بر روی کامپیوتر نصب شده باشند.

<sup>۱</sup> Field-programmable Gate Array

<sup>۲</sup> Integrated Synthesis Environment

<sup>۳</sup> Complex Programmable Logic Device

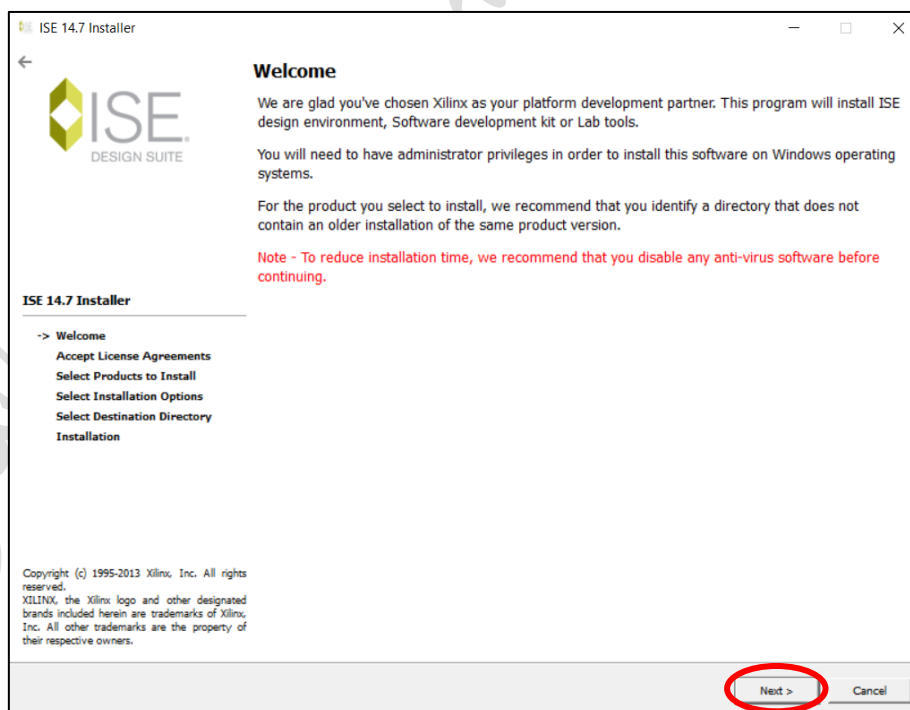
## ۲. نصب نرم افزار

نرم افزار ISE دارای نسخه های متعددی می باشد که می توان آن ها را از سایت شرکت Xilinx خریداری کرده و فایل نصب آن را دانلود و یا در قالب دیسک دریافت نمود. همچنین این نرم افزار دارای نسخه ای رایگان با عنوان WebPACK نیز می باشد که امکانات کمتری نسبت به دیگر نسخه ها در اختیار کاربر می گذارد، اما برای اهداف آموزشی در دانشگاه ها کاملاً مناسب است.

در لیست زیر سیستم های عاملی که به صورت رسمی از نسخه ی ۱۴.۷ این نرم افزار پشتیبانی می کنند آورده شده است (البته امکان اجرای نرم افزار فوق بر روی سیستم های عامل دیگر نیز با اعمال برخی تغییرات وجود دارد).

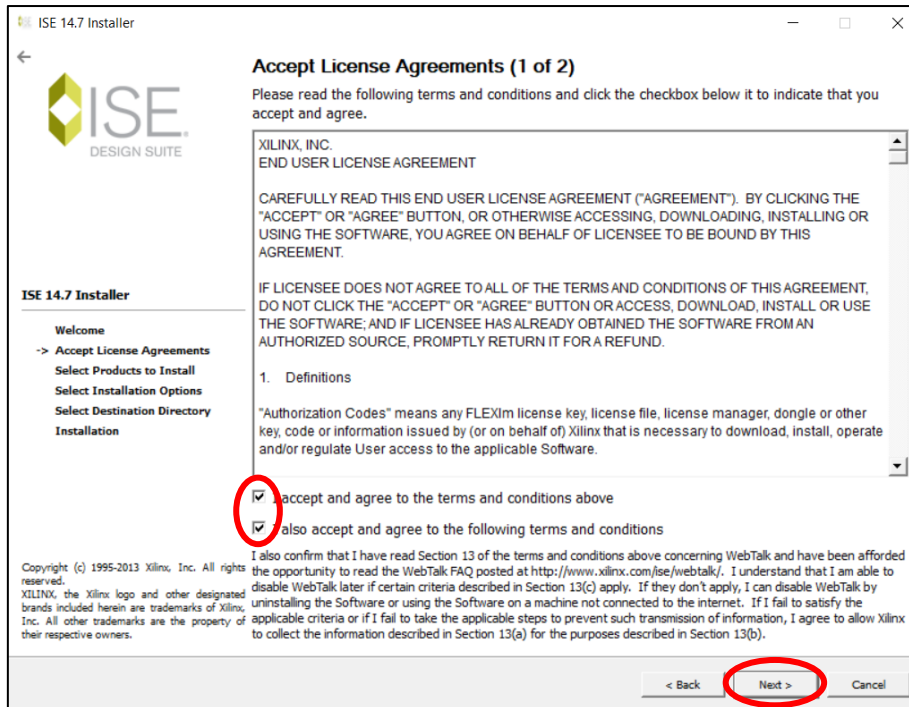
- Microsoft Windows 10 (32 and 64 bits)
- SUSE Linux Enterprise 11 (32 and 64 bits)
- Red Hat Enterprise Workstation 4, 5, and 6 (32 and 64 bits)

پس از دریافت فایل های نصب نرم افزار و اجرای فایل اصلی نصب که معمولاً "xsetup.exe" نام گذاری شده است، پنجره ی شکل ۱ برای شما باز خواهد شد. برای ادامه، کلید "Next" را انتخاب کنید.

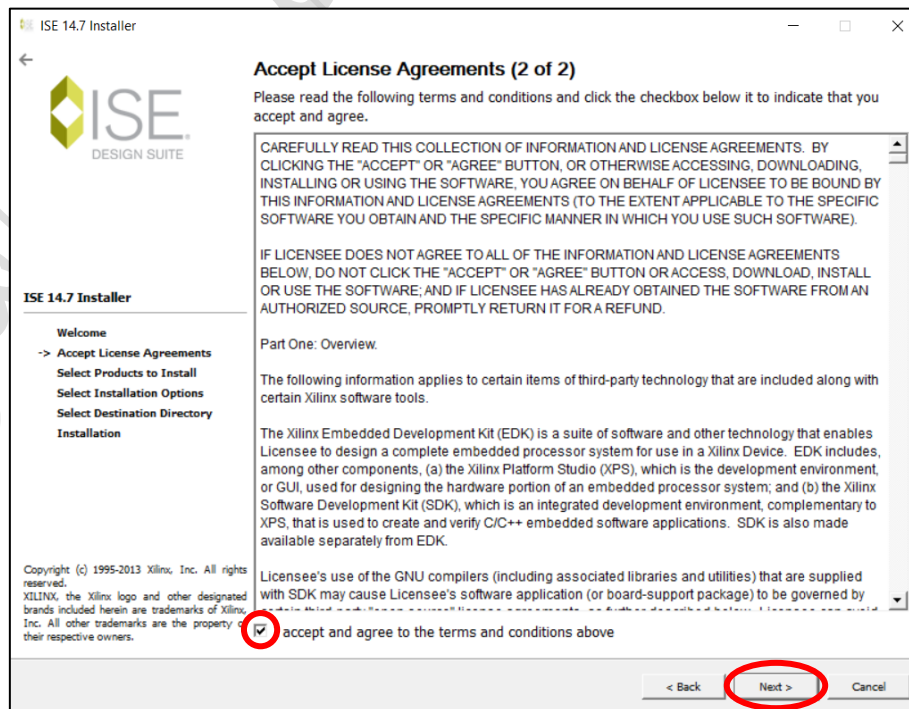


شکل ۱. صفحه ی ابتدایی نصب نرم افزار

با فشردن کلید "Next"، شما وارد صفحه‌های پذیرفتن قوانین در دو صفحه (شکل‌های ۲ و ۳) خواهید شد که برای نصب نرم‌افزار، ملزم به پذیرفتن آن‌ها هستید. پس از انتخاب موارد مشخص شده، گزینه‌ی "Next" را انتخاب کنید.

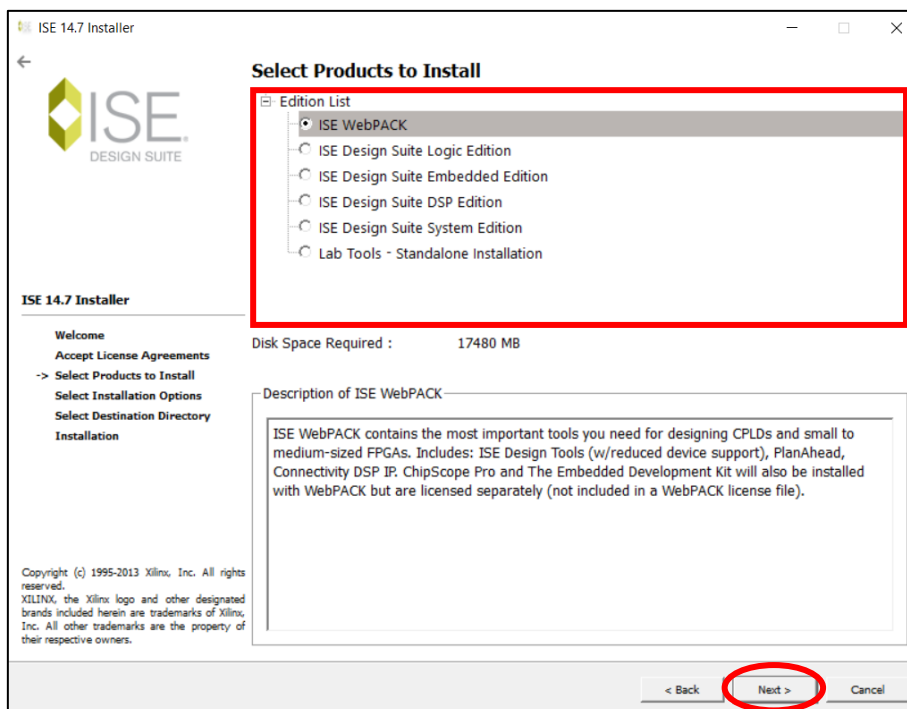


شکل ۲. صفحه‌ی اول پذیرفتن مقررات استفاده از نرم‌افزار



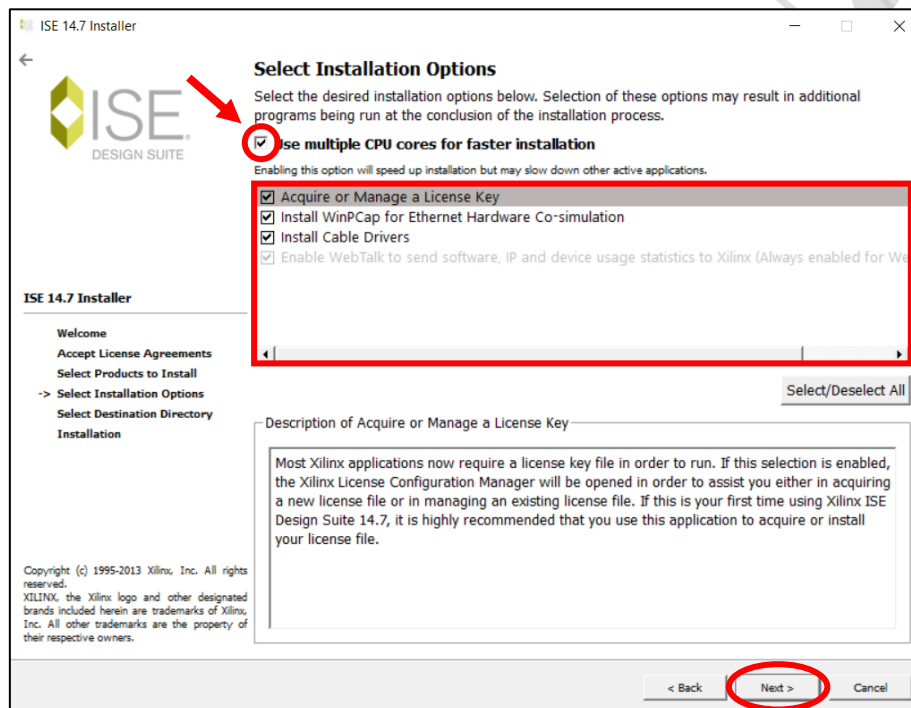
شکل ۳. صفحه‌ی دوم پذیرفتن مقررات استفاده از نرم‌افزار

در صورتی که قوانین را پذیرفته باشید، وارد صفحه‌ی شکل ۴ خواهید شد تا نسخه‌ای از نرم‌افزار ISE که قصد نصب آن را دارید انتخاب کنید. در این آموزش ما از نسخه‌ی WebPack نرم‌افزار که یک نسخه‌ی رایگان است، استفاده خواهیم نمود.



شکل ۴. انتخاب نسخه‌ی نرم‌افزار برای نصب

پس از انتخاب نسخه‌ی نرم‌افزار، مشابه شکل ۵ از شما جهت نصب تعدادی افزونه سؤال پرسیده خواهد شد که بسته به نیازتان آن‌ها را انتخاب می‌کنید. (در صورتی که با این افزونه‌ها آشنایی ندارید، بدون تغییر آن‌ها ادامه دهید). همچنین با انتخاب گزینه‌ای که با فلش در شکل نشان داده شده است، نرم‌افزار هنگام نصب از چند هسته‌ی پردازنده به صورت همزمان استفاده خواهد کرد که روند نصب را تسریع می‌بخشد. در صورت انتخاب این گزینه پیشنهاد می‌شود تمامی نرم‌افزارهایی که باز هستند را ببندید؛ زیرا توان پردازشی سیستم به شدت پایین خواهد آمد.

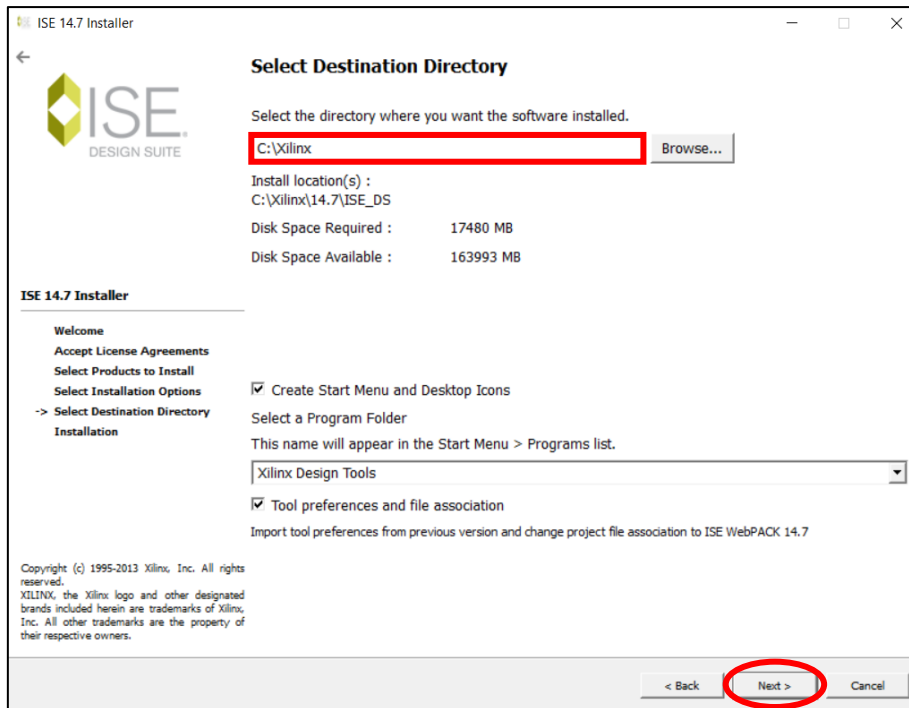


شکل ۵. انتخاب گزینه‌های جانبی برای نصب نرم‌افزار

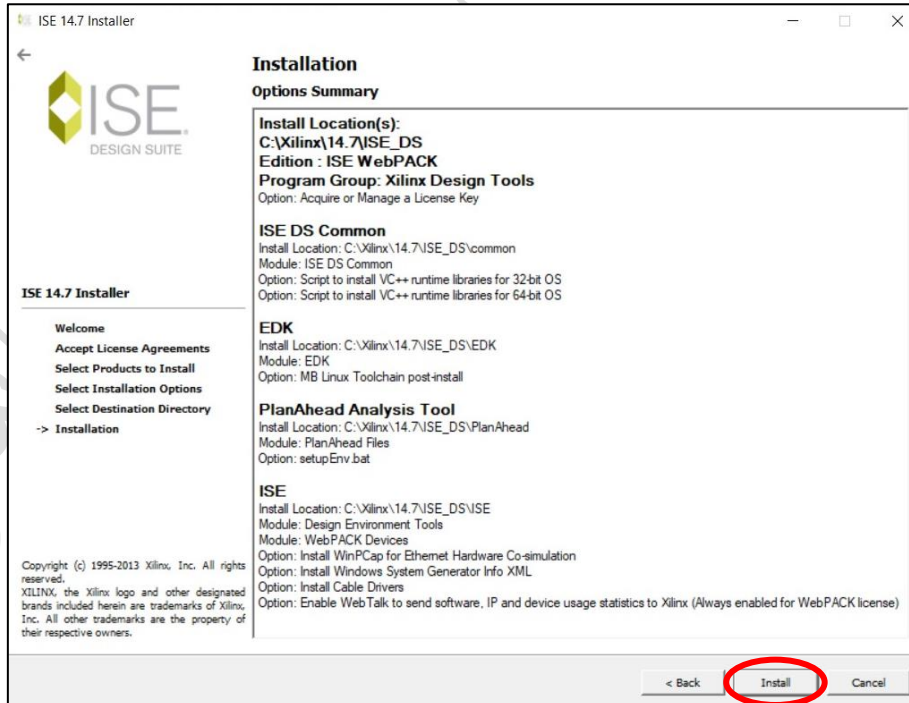
در ادامه‌ی نصب، مشابه شکل ۶، راجع به آدرس نصب نرم‌افزار و برخی موارد مانند ایجاد میانبر<sup>۱</sup> از شما پرسیده خواهد شد که با گذر از این صفحه و فشردن کلید "Next"، خلاصه‌ای از تنظیماتی که در فرآیند نصب انتخاب کرده‌اید، مانند شکل ۷، برایتان نمایان خواهد شد. پس از تأیید نهایی، نرم‌افزار شروع به نصب خواهد کرد (شکل ۸).

<sup>۱</sup> Shortcut





شکل ۶. انتخاب آدرس نصب نرم افزار و میانبرها



شکل ۷. خلاصه‌ای از تنظیمات انتخاب شده برای نصب نرم افزار



شکل ۸. صفحه‌ی نصب نرم‌افزار

در حین نصب نرم‌افزار، ممکن است بسته به سیستم شما، صفحه‌ی نصب تعدادی نرم‌افزارهای جانبی مانند Microsoft Visual C++، که برای اجرای نرم‌افزار ISE اجباری هستند، باز شوند. فرآیند نصب این موارد را نیز طی کنید.

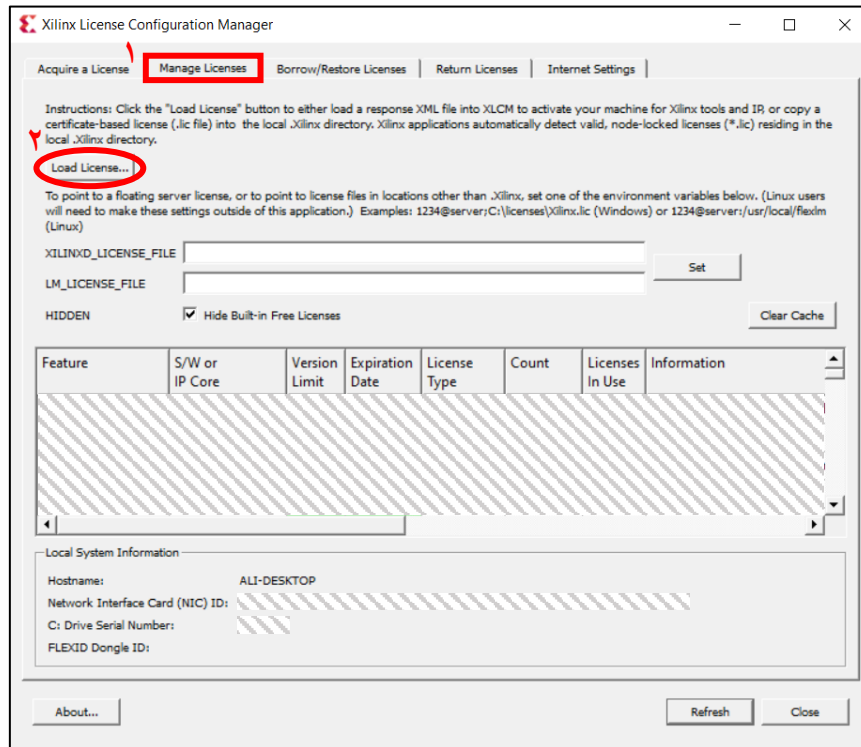
پس از پایان مراحل فوق و نصب کامل نرم‌افزار، در صورتی که خطایی وجود نداشته باشد، پنجره‌ای جدید با نام "Xilinx License Configuration Manager" برای شما باز خواهد شد تا فایل مجوز<sup>۱</sup> که یک فایل با فرمت ".lic" و یا ".xml" می‌باشد را برای نرم‌افزار مشخص کنید (این کار برای جلوگیری از سوء استفاده‌های احتمالی حتی برای نسخه‌های رایگان اجباری است<sup>۲</sup>). همچنین در صورتی که این پنجره به صورت خودکار باز نشد، می‌توانید عبارت "Manage Xilinx Licenses" را در میان نرم‌افزارهای موجود در کامپیوتر خود جستجو کنید.

برای ثبت مجوز، ابتدا فایل مجوز را در یک محل دلخواه در همان درایوی که نرم‌افزار را نصب کرده‌اید کپی نمایید. سپس همانطور که در شکل ۹ نشان داده شده است، وارد پنجره‌ی "Manage Licenses" در نرم‌افزار مدیریت مجوز شده و بر روی

<sup>۱</sup> License

<sup>۲</sup> فایل مجوز را می‌توانید از نماینده‌های شرکت Xilinx در محل زندگی خود و یا با پر کردن فرم مربوطه به صورت آنلاین در سایت این شرکت دریافت نمایید.

کلید "Load License..." کلیک کنید. با کلیک بر روی این گزینه، یک File Dialog برای انتخاب فایل مجوز باز خواهد شد. فایل را انتخاب کرده و بر روی گزینه‌ی "Open" کلیک نمایید. در صورت صحیح بودن فایل مجوز، پیغامی مبتنی بر موفقیت‌آمیز بودن فرآیند به شما نشان داده خواهد شد. پس از این مرحله می‌توانید تمام پنجره‌ها را بسته و شروع به استفاده از نرم‌افزار نمایید.



شکل ۹. نرم‌افزار مدیریت مجوز شرکت Xilinx

در صورتی که از سیستم عامل ویندوز ۱۰ استفاده می‌کنید، ممکن است همچنان پس از نصب نرم‌افزار قادر به اجرای آن نباشید. برای رفع این مشکل مراحل زیر را طی کنید:

۱- وارد آدرس زیر در محل نصب نرم‌افزار شوید.

`<install_path>\Xilinx\14.7\ISE_DS\ISE\lib\nt64\`

۲- نام فایل "libPortability.dll" را به "libPortability.dll.orig" تغییر دهید.

۳- از فایل "libPortabilityNOSH.dll" یک کپی ایجاد کرده و نام آن را "libPortability.dll" قرار دهید.

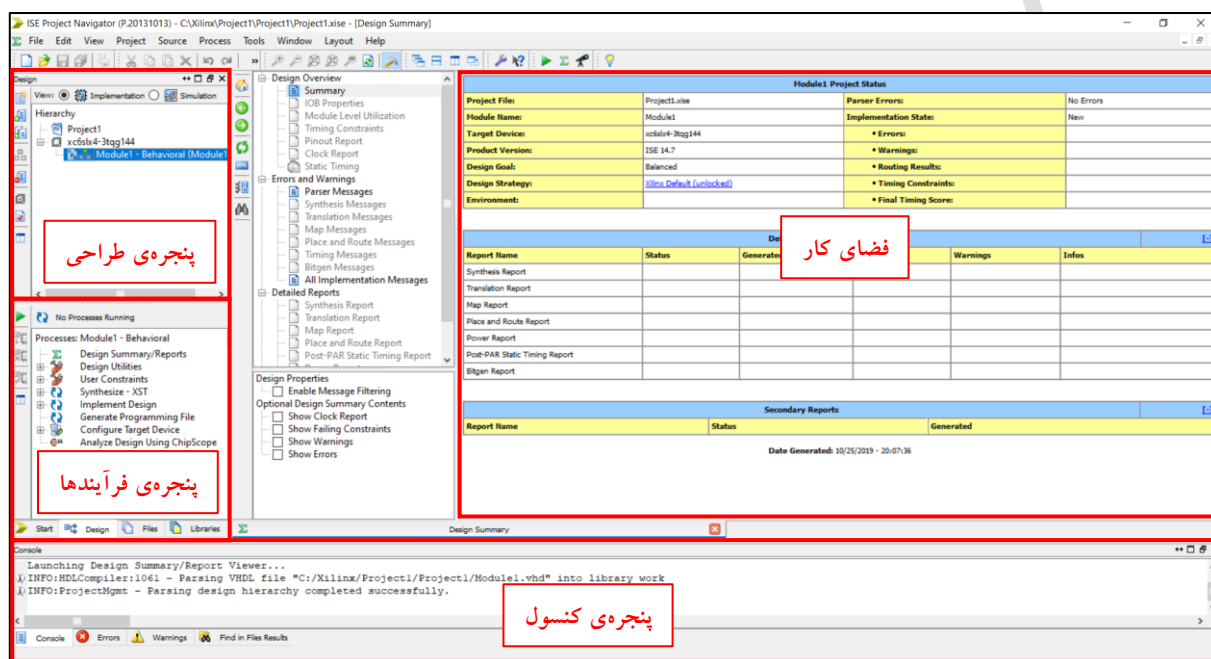
۴- این بار وارد آدرس زیر شده و مراحل ۲ و ۳ را برای فایل‌های داخل این پوشه نیز تکرار کنید.

`<install_path>\Xilinx\14.7\ISE_DS\common\lib\nt64\`

حال سیستم خود را Restart کرده و از استفاده از نرم‌افزار لذت ببرید.

### ۳. آشنایی با محیط نرم افزار

پس از انجام مراحل فوق و اجرای نرم افزار وارد صفحه‌ای مشابه شکل ۱۰ خواهید شد. محیط اصلی نرم افزار از چهار بخش اصلی تشکیل شده است که در ادامه هر کدام را به صورت کامل توضیح خواهیم داد.



شکل ۱۰. محیط نرم افزار ISE

**پنجره طراحی:** در این بخش تمامی قسمت‌های پروژه (به عنوان مثال ماژول‌ها) نمایش داده می‌شوند.

**پنجره فرآیندها:** در این بخش وضعیت انجام هر یک از فرآیندهای طراحی و تولید طرح مانند سنتز، پیاده‌سازی، و یا شبیه‌سازی نمایش داده خواهد شد.

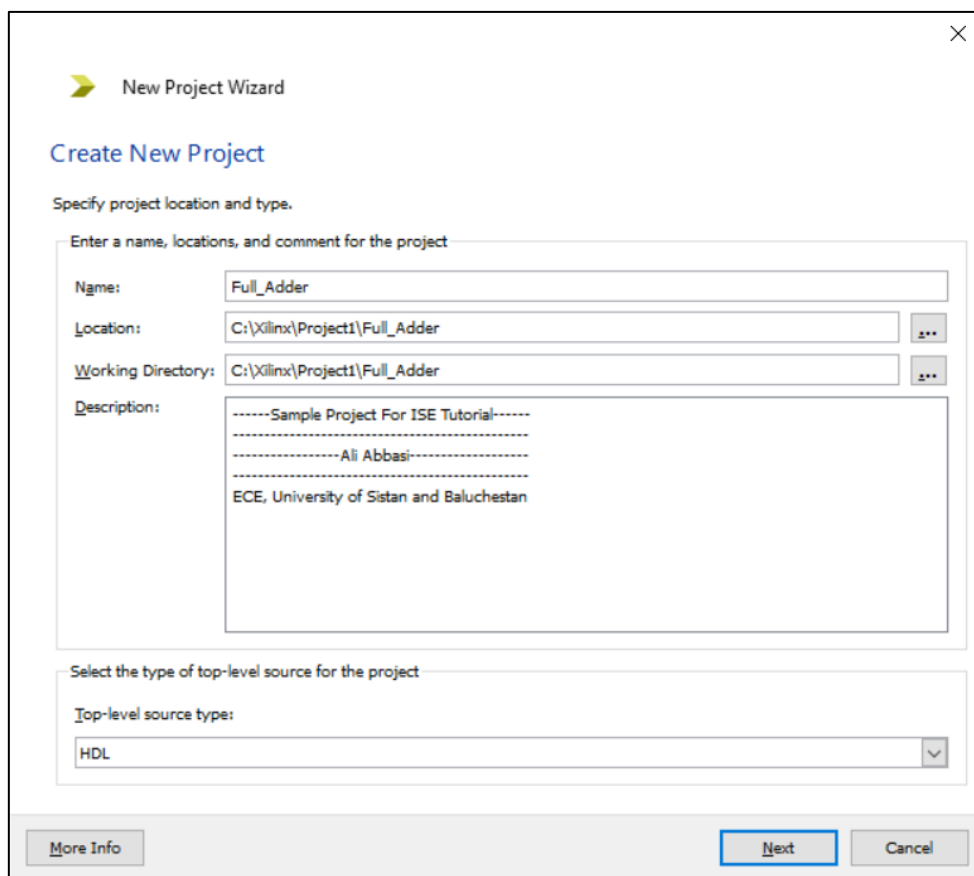
**پنجره کنسول:** تمامی اعلانات در زمان کار با نرم افزار، مانند وضعیت نرم افزار، خطاها و اخطارها، در این قسمت به کاربر اعلام می‌شود.

**فضای کار:** بخش اصلی نرم افزار فضای کار است که توصیف‌ها و فعالیت‌های مرتبط با تولید طرح در آن انجام می‌شود. به عنوان مثال، در شکل ۱۰ پنجره‌ای تحت عنوان خلاصه‌ی طراحی<sup>۱</sup> (به عنوان گزارش طراحی نیز شناخته می‌شود) در بخش فضای کار باز می‌باشد. این پنجره اطلاعات مفیدی از فرآیند کار مانند بهینه‌سازی‌ها، میزان تأخیر و شرح منابع مصرفی به کاربر می‌دهد که در بخش ۱۱ دسترسی به آن را توضیح خواهیم داد.

<sup>۱</sup> Design Summary

## ۴. ایجاد یک پروژه جدید

برای شروع به کار با نرم‌افزار، اولین کار ایجاد یک پروژه جدید خواهد بود. برای این کار از منوی “File” گزینهی “New Project” را انتخاب کنید. پس از انتخاب این گزینه، صفحه‌ی شکل ۱۱ باز خواهد شد.



شکل ۱۱. انتخاب نام و محل ذخیره‌ی پروژه

**Name:** یک نام برای پروژه‌ی خود انتخاب کنید.

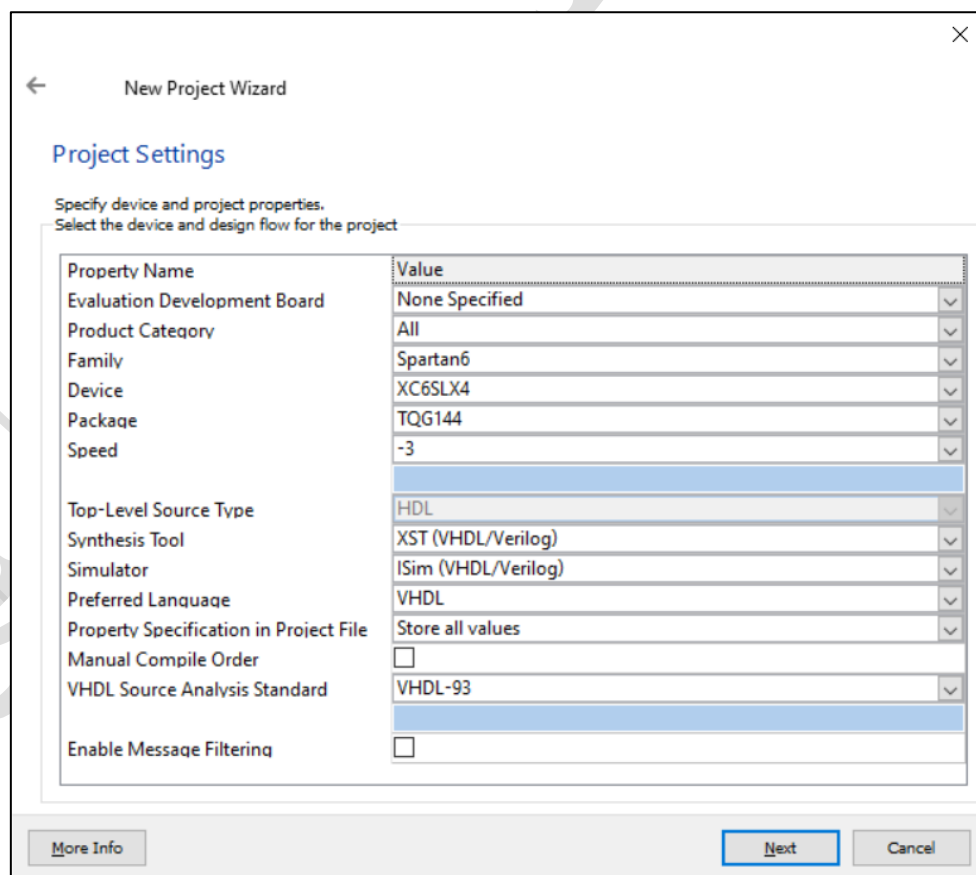
**Location:** آدرس ذخیره‌ی پروژه را مشخص کنید.

**Working Directory:** آدرس پوشه‌ای که قصد اضافه کردن ماژول‌ها و تمامی فایل‌های پروژه‌ی خود را دارید تعیین کنید. (آدرس این پوشه معمولاً با آدرس بخش قبلی یکسان است.)

**Description:** در صورت نیاز، توضیحاتی راجع به پروژه‌ی خود در این قسمت وارد نمایید.

**Top-level source type:** در آخرین بخش، باید نوع بالاترین ماژول پروژه‌ی خود را مشخص کنید که می‌تواند به شکل‌های مختلفی مانند شماتیک<sup>۱</sup> و یا کد توصیف سخت‌افزار<sup>۲</sup> باشد. برای روشن‌تر شدن بیشتر این مفهوم، فرض کنید قصد ساخت یک جمع‌کننده‌ی چهار بیتی به کمک چهار جمع‌کننده‌ی تک بیتی را دارید. در این طراحی، بالاترین سطح (هاژول) پروژه، بخشی است که نحوه‌ی اتصال بخش‌های کوچکتر یعنی جمع‌کننده‌های تک بیتی به یکدیگر را نشان می‌دهد. حال در این قسمت باید مشخص کنید که بالاترین سطح طراحی شما به چه صورتی خواهد بود؛ اگر این طراحی را به کمک کدهای توصیف سخت‌افزار انجام می‌دهید، گزینه‌ی مربوطه را انتخاب کنید. اگر هم قصد توصیف مدار با استفاده از شمای مدار و سیم‌بندی بخش‌های آن به صورت دستی را دارید، گزینه‌ی شماتیک را انتخاب کنید. انتخاب ما در این آموزش، زبان توصیف سخت‌افزار می‌باشد و به مباحث طراحی شماتیک پرداخته نخواهد شد.

پس از انتخاب گزینه‌ی "Next"، وارد صفحه‌ای خواهید شد که باید برخی اطلاعات پایه، مانند زبان توصیف و مشخصات تراشه‌ای که قصد برنامه ریزی آن را دارید، انتخاب کنید (شکل ۱۲). در صورتی که قصد پیاده‌سازی واقعی طرح را ندارید، انتخاب نوع تراشه اهمیتی ندارد و می‌توانید بدون تغییر دادن مقادیر مربوط به این بخش ادامه دهید.



شکل ۱۲. انتخاب مشخصات پایه‌ی پروژه

<sup>1</sup> Schematic

<sup>2</sup> HDL: Hardware Description Language

**Evaluation Development Board**: اگر از بوردهای آموزشی مورد تأیید شرکت Xilinx استفاده می‌کنید، نوع آن را در این بخش انتخاب کنید.

**Product Category**: دسته‌بندی تراشه‌ی خود را انتخاب کنید. (به عنوان مثال همه‌منظوره و یا نظامی)

**Family**: خانواده‌ی تراشه‌ی مورد نظر را در این بخش مشخص کنید.

**Device**: شناسه‌ی دقیق تراشه را انتخاب کنید.

**Package**: ساختار بسته‌بندی تراشه را مشخص کنید.

**Speed**: رده‌ی سرعتی تراشه را انتخاب کنید.

**Top-Level Source Type**: نحوه‌ی توصیف بالاترین سطح پروژه، که در بخش قبل مشخص شد و قابل تغییر نمی‌باشد.

**Synthesis Tool**: ابزار سنتزی که قصد استفاده از آن را دارید انتخاب نمایید.

**Simulator**: شبیه ساز پروژه‌ی خود را انتخاب کنید. (به عنوان مثال شبیه ساز ISim از شرکت Xilinx و یا ModelSim از شرکت Mentor Products)

**Preferred Language**: زبانی که تصمیم دارید توصیف را به کمک آن انجام دهید.

**Property Specification in Project File**: تعیین کنید که کدام نوع از خواص در فایل پروژه‌ی شما ذخیره شود.

**Manual Compile Order**: در صورتی که می‌خواهید سلسه مراتب ماژول‌ها برای سنتز و شبیه‌سازی را به صورت دستی مشخص کنید، این گزینه را انتخاب کنید.

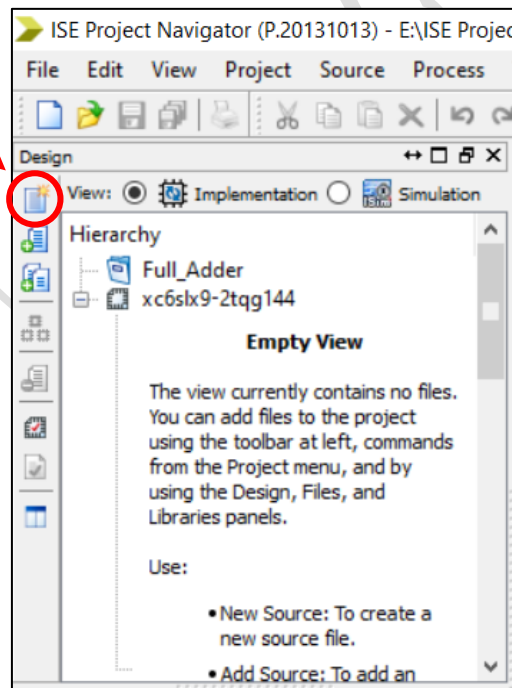
**VHDL Source Analysis Standard**: استاندارد زبان VHDL که قصد استفاده از آن را دارید، مشخص کنید.

با فشردن کلید "Next"، وارد صفحه‌ای خواهید شد که خلاصه‌ی انتخاب‌های شما در روند ایجاد پروژه را نشان می‌دهد. در صورت صحیح بودن اطلاعات، کلید "Finish" را فشار دهید تا پروژه‌ی شما ایجاد شود. حال به صفحه‌ی اول باز خواهید گشت و باید source های مورد نیاز خود را به پروژه اضافه نمایید. در این آموزش، برای توضیح مراحل کار با نرم‌افزار، ابتدا یک تمام‌جمع‌کننده در سطح گیت توصیف خواهد شد، سپس درستی عملکرد آن سنجیده شده، و در نهایت بر روی یک تراشه‌ی Spartan 6 پیاده‌سازی می‌شود.

## ۵. اضافه کردن Source به پروژه

یک پروژه‌ی توصیف سخت‌افزار می‌تواند از بخش‌های مختلفی تشکیل شده باشد؛ به عنوان مثال، یک پروژه‌ی ساخت تمام‌جمع‌کننده، شامل یک فایل برای توصیف عملکرد مدار، یک فایل به نام میز آزمایش<sup>۱</sup> برای بررسی صحت عملکرد مدار، و یک فایل برای تعریف محدودیت‌های پیاده‌سازی<sup>۲</sup> مثل تعیین پین‌های ورودی و خروجی خواهد بود. به هر یک از این بخش‌ها، یک source می‌گویند که در ادامه راجع به نحوه‌ی ایجاد آن‌ها در نرم‌افزار ISE صحبت خواهیم کرد.

اولین source که باید به این پروژه اضافه شود، source توصیف تمام‌جمع‌کننده می‌باشد. برای این کار، بر روی کلیدی که در شکل ۱۳ نشان داده شده است کلیک کنید.



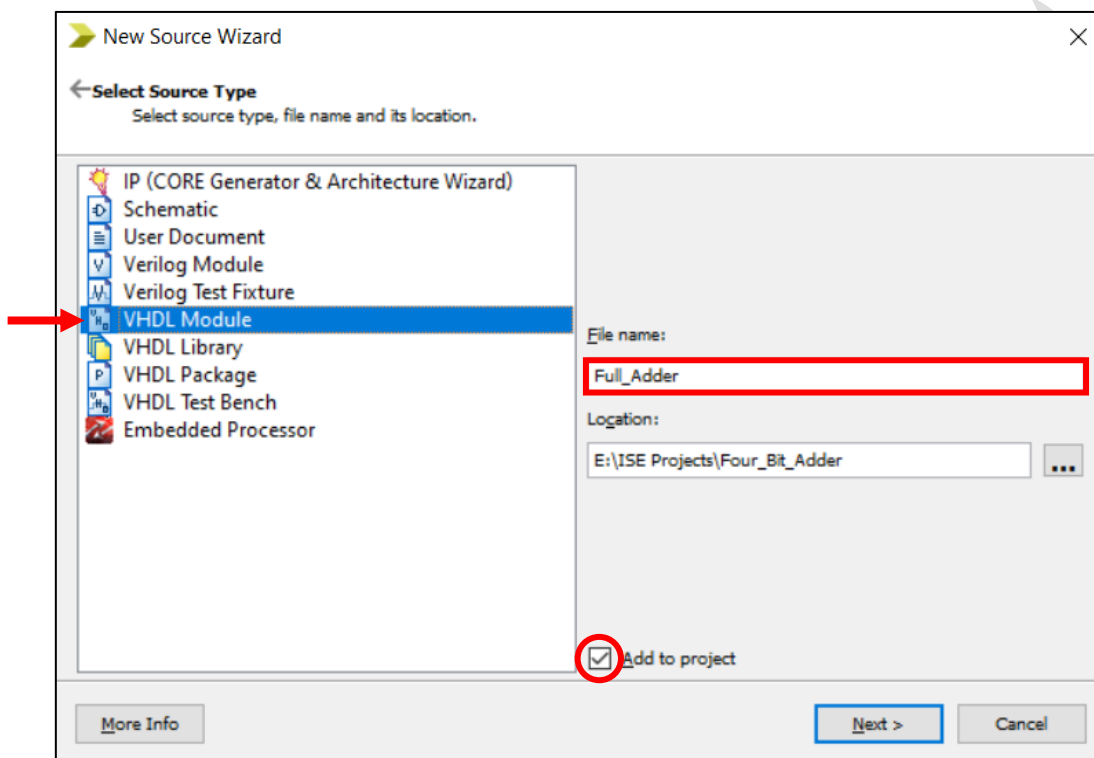
شکل ۱۳. ساخت یک source جدید برای پروژه

<sup>1</sup> Test Bench

<sup>2</sup> Implementation Constraints File

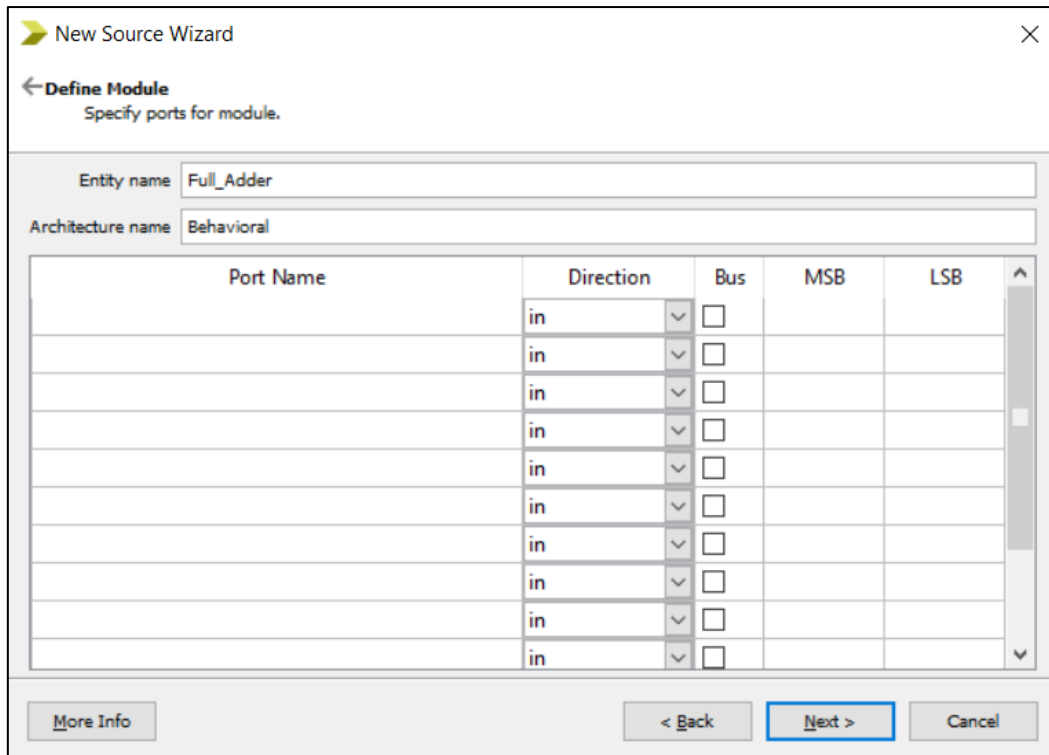


پس از انتخاب گزینه‌ی فوق، پنجره‌ی شکل ۱۴ برای شما نمایش داده خواهد شد که باید با توجه به نیاز خود یک گزینه را انتخاب کنید. در این مرحله ما قصد توصیف یک ماژول به کمک زبان VHDL را داریم، پس گزینه‌ی مربوطه را انتخاب می‌کنیم. همچنین باید برای این source یک نام نیز انتخاب شود که با توجه به ماژول مورد نظر، بهترین نام "Full\_Adder" خواهد بود. توجه داشته باشید که حتماً گزینه‌ی "Add to project" انتخاب شده باشد.



شکل ۱۴. انتخاب نوع source و نام‌گذاری آن (از نوع ماژول)

با فشردن کلید "Next" به صفحه‌ی بعد خواهید رفت. با توجه به اینکه در مرحله‌ی قبل، نوع source یک ماژول انتخاب شده است، صفحه‌ای به شما نمایش داده خواهد شد که می‌توانید پورت‌های ورودی و خروجی ماژول خود را مشخص کرده و نرم‌افزار یک کد اولیه‌ی مناسب به صورت خودکار برای شما تولید کند (شکل ۱۵). از آنجایی که پورت‌ها را هنگام توصیف به صورت دستی در کد خواهیم نوشت از این بخش عبور می‌کنیم.

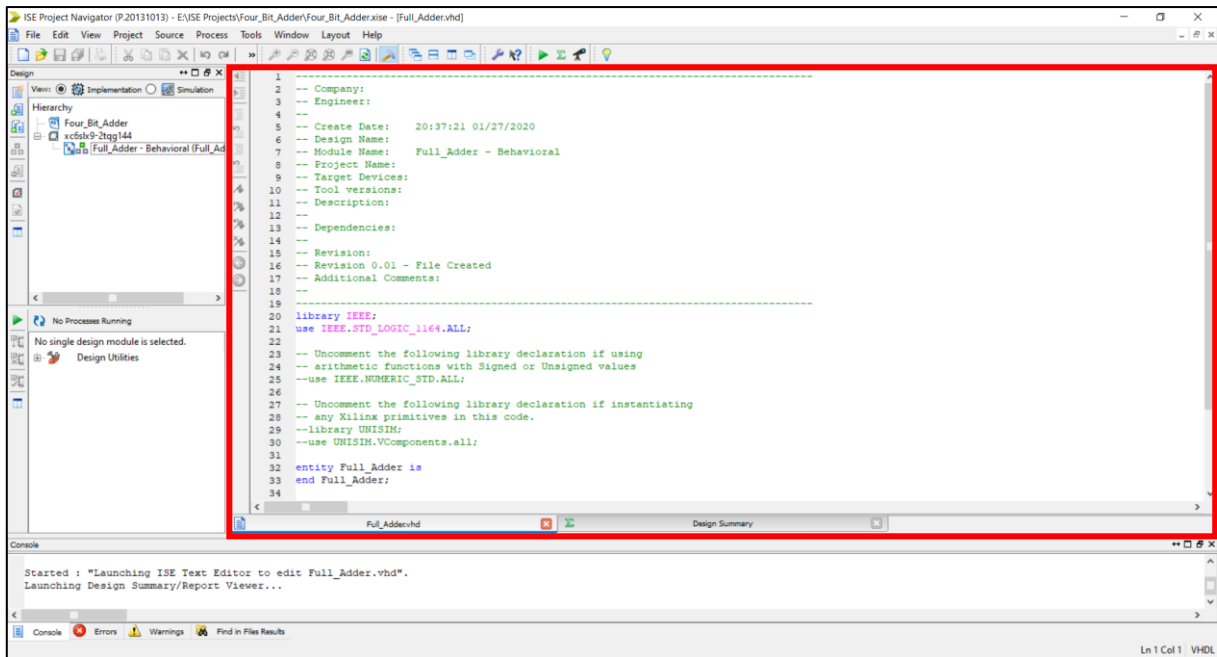


شکل ۱۵. مشخص کردن پورت‌های ماژول در حال ساخت

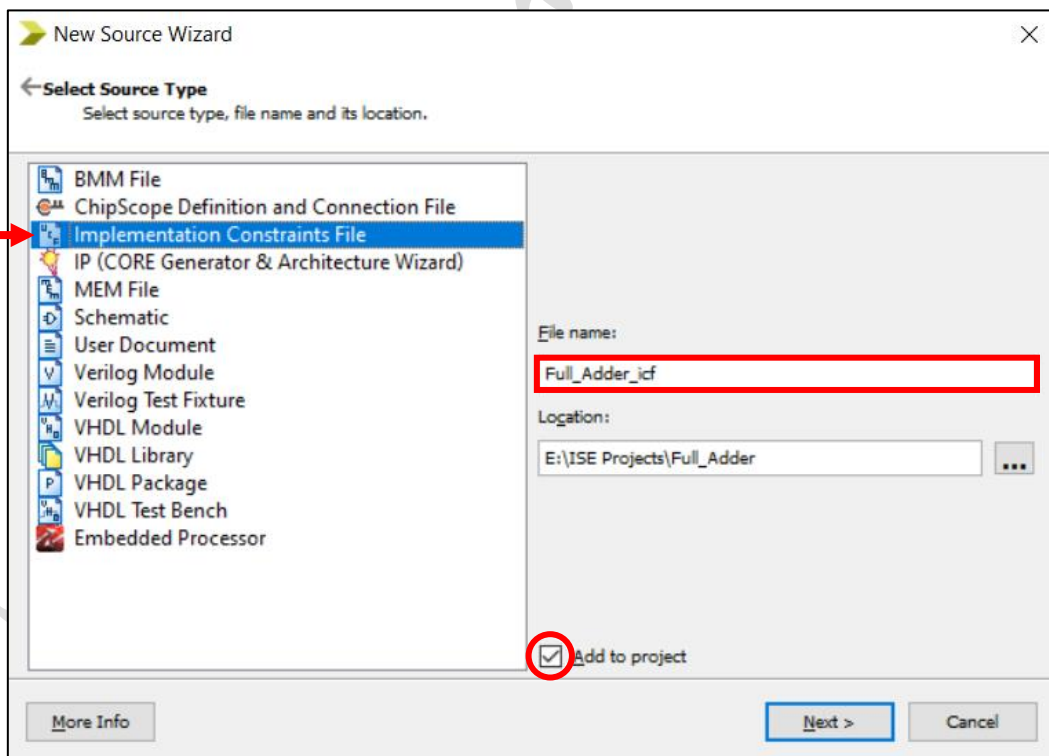
با گذر از این صفحه، در صفحه‌ی بعد، خلاصه‌ای از source ایجاد شده به شما نمایش داده خواهد شد. در صورت تأیید کلیک "Finish" را انتخاب کنید تا source مورد نظر ایجاد و به پروژه اضافه شود. پس از ایجاد source، فایل آن به پنجره‌ی طراحی پروژه اضافه خواهد شد که با دو بار کلیک بر روی آن می‌توانید آن را باز کنید. توجه شود که پس از ایجاد source، فایل به صورت خودکار باز خواهد شد و می‌توانید شروع به توصیف طرح کنید (شکل ۱۶).

دومین source که احتمالاً به آن نیاز پیدا خواهید کرد، source آزمایش و شبیه‌سازی مدار است که در بخش شبیه‌سازی رفتاری (بخش ۸)، نحوه‌ی ایجاد و استفاده از آن به صورت کامل توضیح داده خواهد شد و فعلاً از آن گذر می‌کنیم.

پس از آنکه توصیف طرح انجام شد و به کمک شبیه‌سازی‌های رفتاری و زمانی از صحت عملکرد آن مطمئن شدید، باید فایل محدودیت‌های پیاده‌سازی را به پروژه اضافه کنید. برای این کار، دوباره از همان محل شکل ۱۳ یک source جدید به پروژه‌ی خود اضافه کنید، با این تفاوت که نوع آن را باید از نوع "Implementation Constraints File" انتخاب نمایید (مطابق شکل ۱۷).



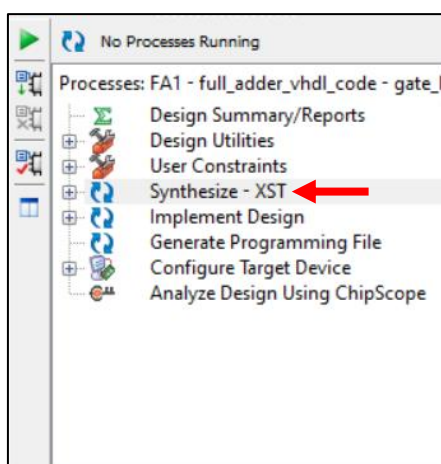
شکل ۱۶. شروع به توصیف ماژول



شکل ۱۷. انتخاب نوع source و نام گذاری آن (از نوع محدودیت پیاده‌سازی)

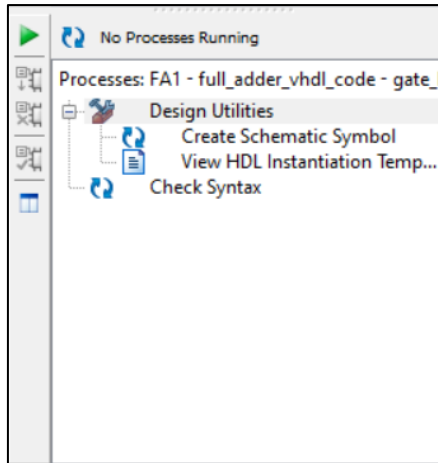
## ۶. سنتز

پس از پایان توصیف طرح، زمان سنتز آن است. برای این کار sourceای که قصد سنتز آن را دارید از پنجره طراحی انتخاب نمایید. سپس در پنجره فرآیند بر روی گزینه سنتز (شکل ۱۸) کلیک راست کرده و گزینه "Run" را انتخاب نمایید. در صورت موفقیت آمیز بودن فرآیند سنتز، یک علامت سبز در سمت چپ این گزینه نمایان خواهد شد. در صورتی که این علامت به رنگ نارنجی یا قرمز بود، به ترتیب به معنای اخطار و خطا می باشد.



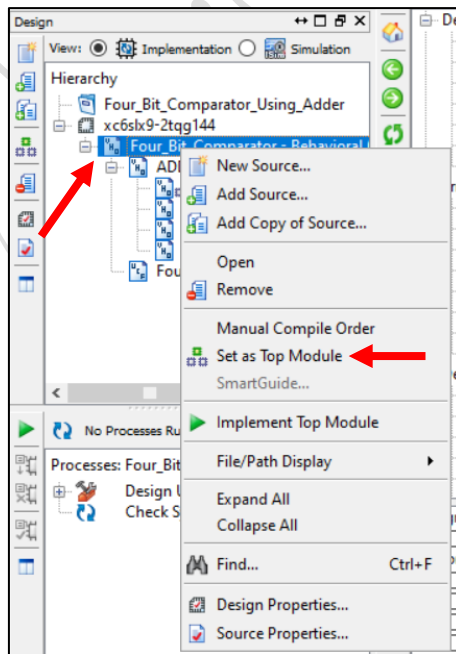
شکل ۱۸. سنتز طرح در پنجره فرآیند

در نظر داشته باشید که برای سنتز و پیاده سازی یک source، باید آن source به عنوان Top Module طرح شما انتخاب شده باشد. در غیر این صورت، گزینه های سنتز و پیاده سازی در پنجره فرآیند نمایش داده نخواهند شد و تنها گزینه هایی مشابه شکل ۱۹ خواهید دید.



شکل ۱۹. پنجره‌ی فرآیند برای source غیر Top Module

برای انتخاب یک source به عنوان Top Module، مطابق شکل ۲۰ بر روی آن source کلیک راست کرده و گزینه‌ی "Set as Top Module" را انتخاب نمایید. (مفهوم Top Module تنها مربوط به زمانی است که چندین source از نوع ماژول در طرح شما وجود داشته باشد.)

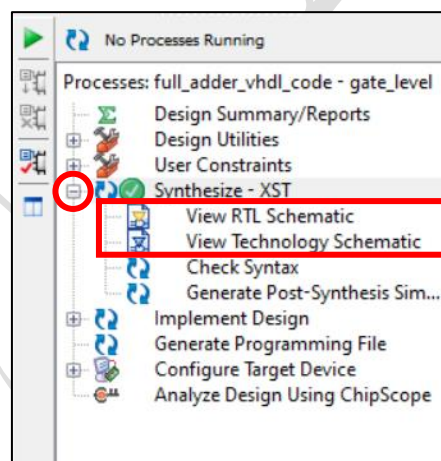


شکل ۲۰. انتخاب یک source به عنوان Top Module

همانطور که می‌دانید، خروجی و نتیجه‌ی عمل سنتز، اصطلاحاً یک Netlist می‌باشد که نمایشی از نحوه‌ی پیاده‌سازی طرحی است که کد آن به یکی از زبان‌های توصیف سخت‌افزار نوشته شده است. Netlist در نرم‌افزار ISE به دو شکل مختلف در دسترس است: شماتیک سطح انتقال ثبات و شماتیک تکنولوژی.

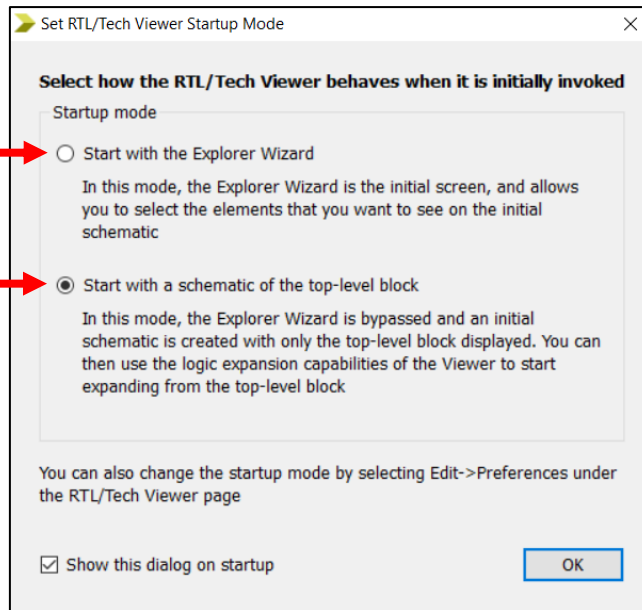
در شماتیک سطح انتقال ثبات، طراحی سطح گیت متناظر با کد نمایش داده می‌شود. در حالی که در شماتیک تکنولوژی، پیاده‌سازی واقعی طرح بر روی یک تراشه‌ی FPGA یا CPLD (یا هر تراشه‌ای که در ابتدای ساخت پروژه انتخاب کردید) نمایش داده خواهد شد. به این معنی که اگر طرح برای یک FPGA بر مبنای LUT سنتز شده است، آن طرح چگونه به کمک LUT ها پیاده‌سازی می‌شود. توجه کنید که شماتیک تکنولوژی، نمایشی از پیاده‌سازی نهایی بر روی FPGA نیست و فرآیندهایی مانند جزءبندی، جایابی، و مسیریابی بر روی آن انجام نشده است. بلکه تنها مشخص شده است که این پیاده‌سازی نیاز به چه منابعی و با چه اتصالاتی دارد.

برای نمایش Netlist طرح، ابتدا بر روی علامت مثبت در کنار گزینه‌ی سنتز در پنجره‌ی فرآیند کلیک کنید تا گزینه‌های بیشتر برای شما نمایش داده شود. سپس بر روی نوع شماتیک مورد نظران دو بار کلیک کنید (شکل ۲۱).



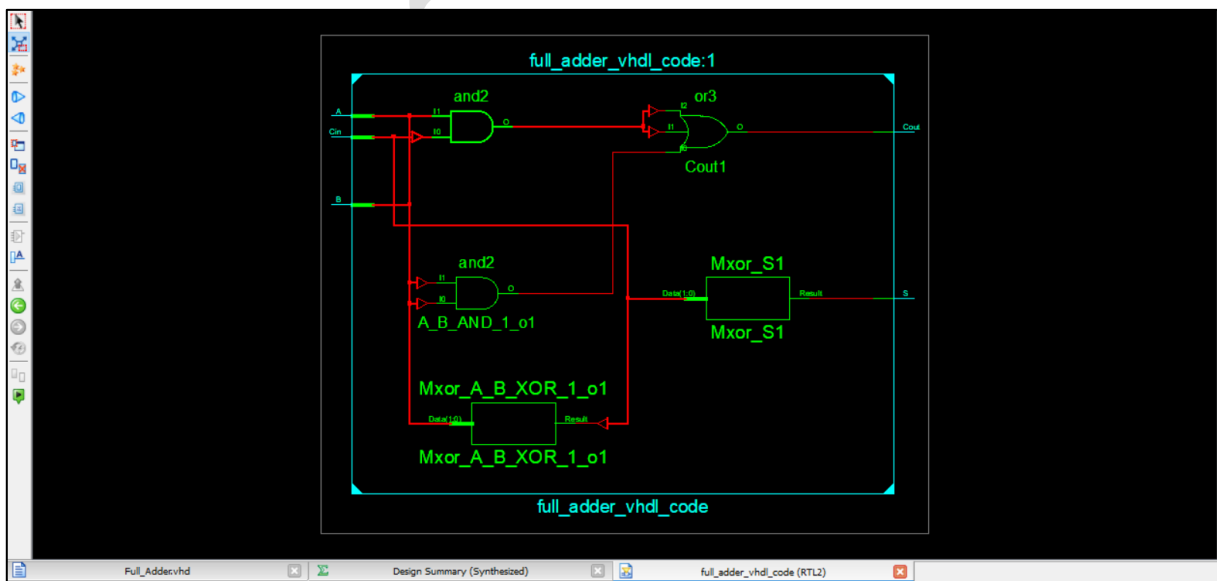
شکل ۲۱. نمایش Netlist طرح پس از سنتز

پس از دو بار کلیک بر روی هرکدام از دو نوع شماتیک، پنجره‌ی شکل ۲۲ نمایان خواهد شد. در این پنجره دو انتخاب دارید که می‌توانید مشخص کنید آیا شماتیک را برای کل طرح ببینید یا برای قسمتی محدود و مشخص شده از آن.



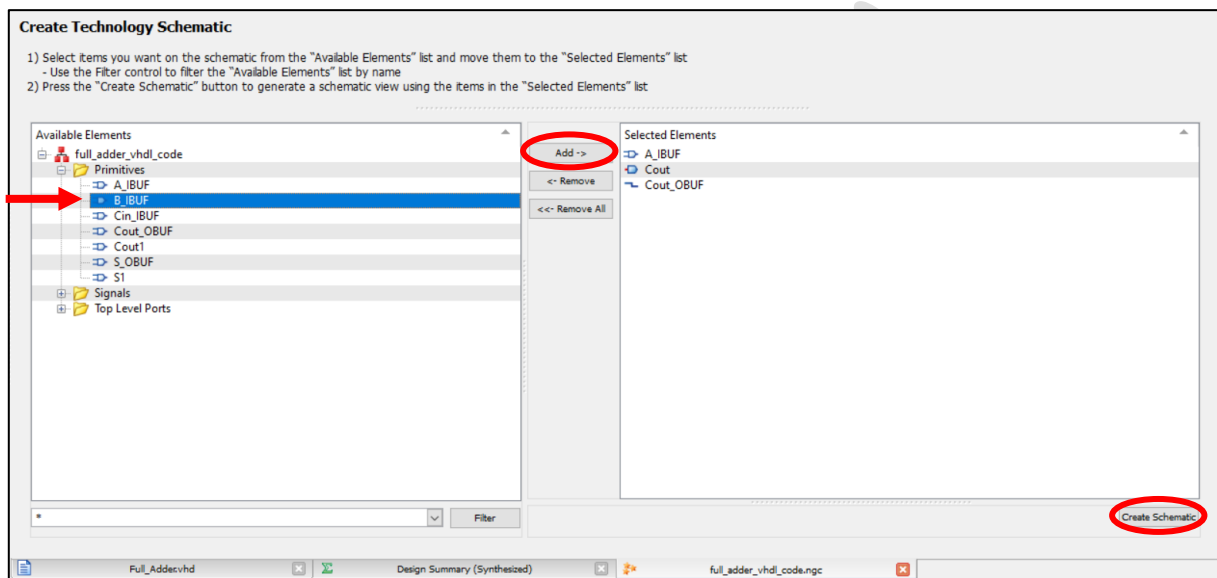
شکل ۲۲. انتخاب نوع نمایش شماتیک

در صورتی که گزینه‌ی دوم را انتخاب کنید، مستقیماً صفحه‌ی نمایش شماتیک باز خواهد شد و شما تمامی اجزاء را به صورت کامل، مشابه شکل ۲۳ که برای یک تمام‌جمع‌کننده است، مشاهده خواهید کرد. ممکن است بخش‌هایی از طرح را به صورت یک بلوک ببینید، برای دقیق‌تر شدن و باز کردن هر یک از این بخش‌ها، می‌توانید بر روی آن بخش دو بار کلیک کنید.



شکل ۲۳. نمایش شماتیک کامل سطح انتقال ثبات برای یک تمام‌جمع‌کننده

اما در صورت انتخاب گزینه‌ی اول، صفحه‌ی دیگری به شما نشان داده می‌شود تا انتخاب کنید که کدام یک از اجزاء طرح در شماتیک نمایش داده شوند (کدام سیگنال‌ها یا پورت‌ها یا...). برای این کار، هر کدام از اجزائی را که نیاز دارید در شماتیک نشان داده شوند از سمت چپ (شکل ۲۴) انتخاب کرده و با فشردن کلید "Add ->" آن‌ها را به شماتیک اضافه کنید. پس از اینکه تمام اجزاء را اضافه کردید بر روی گزینه‌ی "Create Schematic" کلیک کنید تا شماتیک برای شما نمایش داده شود. در این نوع نمایش نیز می‌توانید بر روی هر بخش که به صورت بلوک نمایش داده شده است، دو بار کلیک کنید تا به صورت دقیق‌تر نشان داده شود. همچنین اگر بر روی جزئی که در صفحه‌ی شکل ۲۴ به شماتیک اضافه نکرده بودید دو بار کلیک کنید، آن جزء نیز به صورت خودکار به شماتیک اضافه خواهد شد.



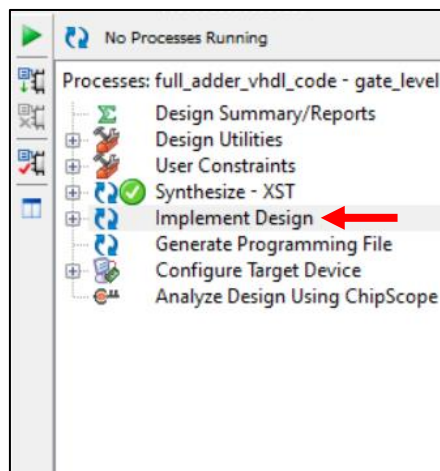
شکل ۲۴. انتخاب اجزاء برای نمایش در شماتیک محدود تکنولوژی

در زمان استفاده از نمایش شماتیک تکنولوژی، می‌توانید با دو بار کلیک بر روی هر یک از LUT ها، معادله، جدول درستی، جدول کارنو، و مدار سطح گیت تابعی که در آن LUT پیاده‌سازی شده است را مشاهده کنید.



## ۷. پیاده‌سازی (ترجمه - جایابی - مسیریابی)

پس از آنکه Netlist طرح توصیف شده، توسط مرحله‌ی سنتز مشخص شد، باید پیاده‌سازی بر روی تراشه‌ی مورد نظر صورت گیرد. برای این کار، مانند سنتز کردن، ابتدا Source مورد نظر را از پنجره‌ی طراحی انتخاب نمایید. سپس بر روی گزینه‌ی "Implement Design" در پنجره‌ی فرآیند (شکل ۲۵) دو بار کلیک کرده و منتظر ظاهر شدن علامت تیک سبز در سمت چپ آن بمانید.



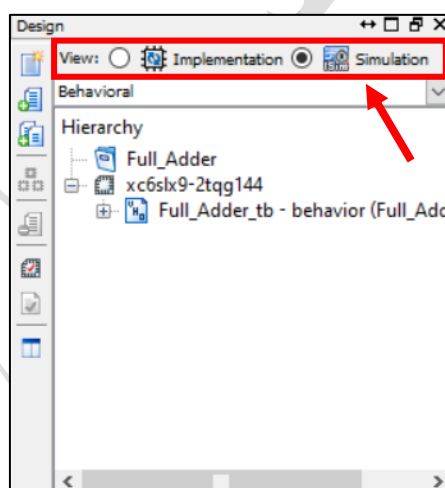
شکل ۲۵. پیاده‌سازی طرح در پنجره‌ی فرآیند

با انجام عمل پیاده‌سازی به صورت فوق، تمامی مراحل پیاده‌سازی به صورت خودکار توسط نرم‌افزار انجام خواهد شد. این مراحل می‌توانند به صورت دستی توسط نرم‌افزارهایی مثل FPGA Editor و PlanAhead که همزمان با نصب نرم‌افزار ISE نصب می‌شوند نیز انجام پذیرند که در این نسخه از آموزش به آن‌ها اشاره‌ای نخواهد شد.

## ۸. شبیه‌سازی رفتاری

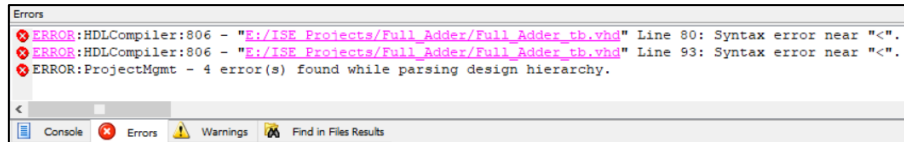
اولین قدم برای آنکه بتوانید طرح خود را شبیه‌سازی کنید، نوشتن فایل آزمایش طرح، که معمولاً به عنوان میز آزمایش نیز شناخته می‌شود، خواهد بود. برای این کار، باید یک source از نوع “VHDL Test Bench” برای ماژول‌های توصیف شده به زبان VHDL، و یا از نوع “Verilog Test Fixture” برای ماژول‌های توصیف شده به زبان Verilog در پروژه‌ی خود ایجاد کنید (در صورتی که نمی‌دانید چگونه این کار را انجام دهید به بخش ۵ مراجعه نمایید). پس از فشردن کلید “Next”، از شما درخواست می‌شود تا ماژولی که قصد نوشتن Test Bench برای آن را دارید انتخاب کنید. با انتخاب ماژول مورد نظر، بخش‌هایی از کد آزمایش به صورت خودکار تولید خواهد شد و نیازی به نوشتن کامل آن از ابتدا نخواهید داشت.

در پایان این مراحل، فایل آزمایش برای ماژول مورد نظر ایجاد شده و در پنجره‌ی فضای کار باز می‌گردد، اما در پنجره‌ی طراحی دیده نخواهد شد. برای مشاهده و دسترسی به فایل‌های آزمایش و شبیه‌سازی پروژه، باید مشابه شکل ۲۶، بخش “View” در پنجره‌ی طراحی را از “Implementation” به “Simulation” تغییر دهید. (برای دیدن فایل‌های اصلی پروژه مانند ماژول‌ها، دوباره گزینه‌ی “Implementation” را انتخاب کنید).



شکل ۲۶. بخش شبیه‌سازی از پنجره‌ی طراحی

از آنجایی که پنجره‌ی کنسول تنها با هر بار ذخیره‌ی فایل، خطاها و هشدارها را نمایش می‌دهد، لازم است فایل‌های طرح خود را مرحله به مرحله به کمک  $Ctrl + S$  ذخیره کنید تا بتوانید بازخوردهای نرم‌افزار را ببینید. اگر فایل آزمایشی که به صورت خودکار توسط نرم‌افزار برای ماژول انتخابی شما تولید شد ذخیره نمایید، احتمالاً مشاهده خواهید کرد که در پنجره‌ی کنسول، خطاهای شکل ۲۷ نمایش داده شده است.



شکل ۲۷. خطاهای موجود در فایل آزمایش ایجاد شده به صورت خودکار توسط نرم‌افزار برای ماژول تمام‌جمع‌کننده

دلیل وجود این خطاها، تولید خودکار قطعه کدهای مربوط به ایجاد سیگنال ساعت در فایل آزمایش است. در صورتی که ماژول در حال آزمایش دارای سیگنال ساعت باشد، پورت ساعت به صورت خودکار توسط نرم‌افزار شناخته خواهد شد و خطایی در کد آزمایش وجود نخواهد داشت (برای پورت‌هایی با نام‌های مشابه clock یا clk). اما در صورتی که این طرح فاقد پورت ساعت باشد (مانند تمام‌جمع‌کننده)، باید قطعه کدهای مربوط به سیگنال ساعت به صورت دستی حذف شوند. برای این کار قطعه کدهای زیر را از فایل آزمایش حذف کنید:

```

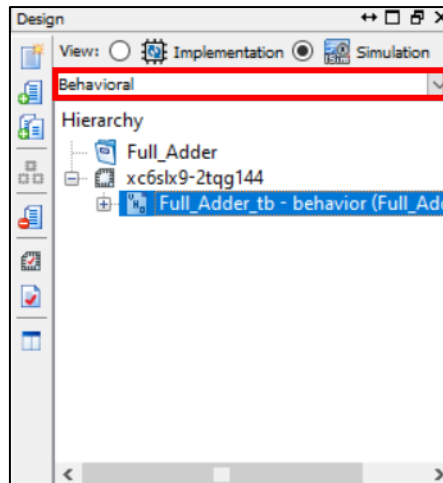
---First One---
constant <clock>_period : time := 10 ns;

---Second One---
<clock>_process :process
begin
    <clock> <= '0';
    wait for <clock>_period/2;
    <clock> <= '1';
    wait for <clock>_period/2;
end process;

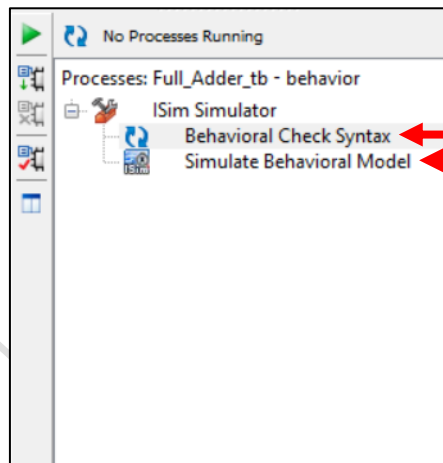
---Third One---
wait for <clock>_period*10;

```

پس از انجام مراحل فوق و نوشتن متن فایل آزمایش، باید مدار را شبیه‌سازی و صحت خروجی‌ها را به کمک شکل موج آن‌ها بررسی کنید. مانند انجام عملیات سنتز و پیاده‌سازی، فایل آزمایش را از پنجره طراحی انتخاب کنید (این بار در حالت شبیه‌سازی که در شکل ۲۶ نشان داده شد). سپس برای آنکه این شبیه‌سازی یک شبیه‌سازی رفتاری باشد، بخش مشخص شده در شکل ۲۸ را بر روی "Behavioral" قرار داده و سپس در پنجره فرآیند، قسمت‌های مشخص شده در شکل ۲۹ را به ترتیب اجرا کنید.



شکل ۲۸. انتخاب نوع شبیه‌سازی

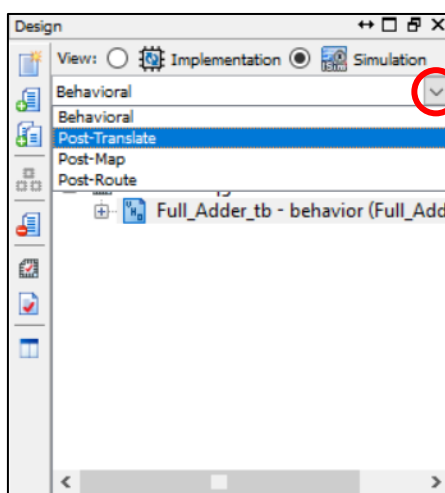


شکل ۲۹. شبیه‌سازی طرح در پنجره‌ی فرآیند

در صورت موفق‌آمیز بودن شبیه‌سازی، پنجره‌ای جدید بر روی سیستم شما باز خواهد شد که شکل موج را نشان می‌دهد. این پنجره همان نرم‌افزاری است که آن را هنگام ساخت پروژه به عنوان شبیه‌ساز انتخاب کردید (که در این پروژه، نرم‌افزار ISIM انتخاب شد). استفاده از امکانات نمایشگر شکل موج ISIM در بخش مربوطه (بخش ۱۰) توضیح داده خواهد شد.

## ۹. شبیه‌سازی زمانی

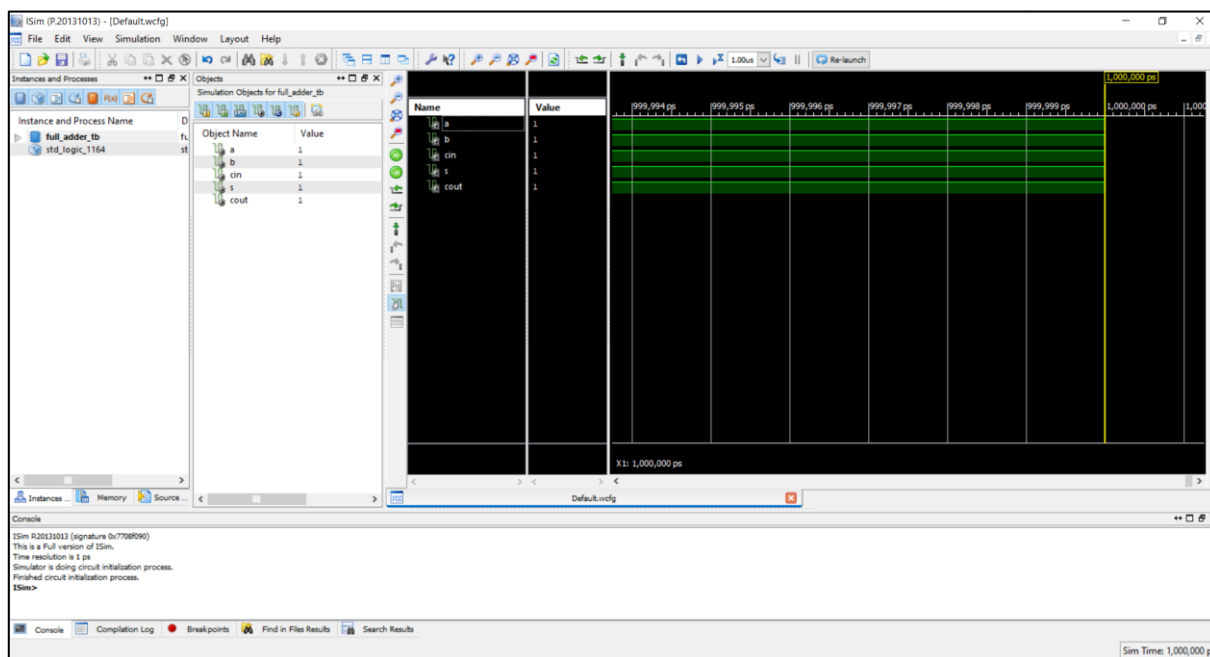
شبیه‌سازی زمانی در نرم‌افزار ISE، بعد از هر سه مرحله از پیاده‌سازی یعنی Map، Translate و Route در دسترس است و می‌توانید بسته به نیازتان، شکل موج را پس از هر یک از این مراحل مشاهده کنید. برای انجام شبیه‌سازی زمانی باید کاملاً مشابه شبیه‌سازی رفتاری عمل کنید. با این تفاوت که هنگام مشخص کردن نوع شبیه‌سازی، به جای انتخاب گزینه‌ی “Behavioral”، یکی از گزینه‌های دیگری که در شکل ۳۰ نمایش داده شده است انتخاب کنید.



شکل ۳۰. انتخاب یکی از انواع شبیه‌سازی

## ۱۰. استفاده از نمایشگر شکل موج نرم افزار Xilinx ISIM

هنگامی که هر کدام از انواع شبیه سازی را انجام می دهید، در صورتی که در زمان ایجاد پروژه، نرم افزار ISIM را به عنوان شبیه ساز انتخاب کرده باشید، پنجره ای مشابه شکل ۳۱ برای شما باز خواهد شد.



شکل ۳۱. محیط کار نرم افزار ISIM

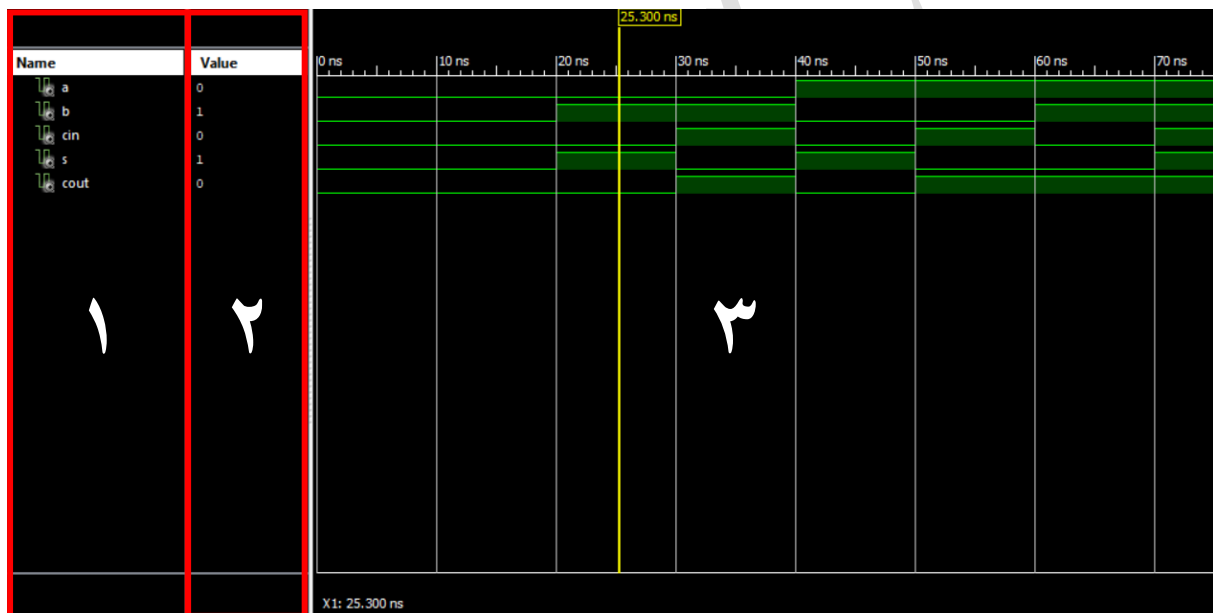
از آنجایی که معمولاً مدت زمانی که برای هر مقدار ورودی در هنگام نوشتن کد آزمایش در نظر گرفته می شود بسیار اندک است، لازم است بر روی شکل موج زوم کنید تا بتوانید تغییرات مقادیر را مشاهده کنید. شکل ۳۲ بخشی از نوار ابزار در این نرم افزار شبیه ساز را نشان می دهد که می توانید از آن برای کاربری راحت تر از نرم افزار استفاده کنید.



شکل ۳۲. بخشی از نوار ابزار نرم افزار ISIM

با استفاده از کلیدهای مشخص شده در شکل ۳۲، می‌توانید به ترتیب از سمت چپ، بر روی شکل موج زوم کنید، زوم را کاهش دهید، و شکل موج را به صورت خودکار به اندازه‌ای در بیاورید که تمام مقادیر ورودی‌ای که برای آن در نظر گرفته‌اید در پنجره‌ی نمایش شکل موج جای بگیرند.

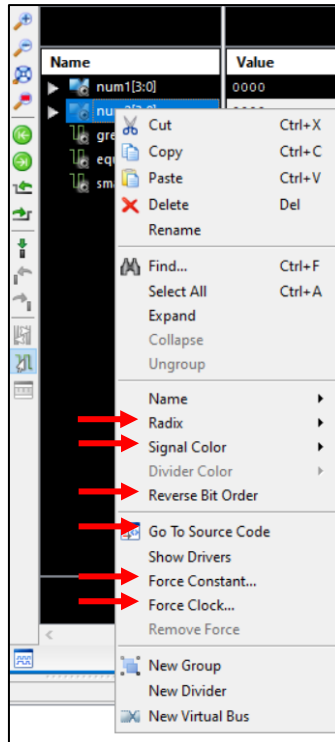
پس از فشردن کلید سوم، شکل موج به صورت شکل ۳۳ نمایش داده خواهد شد. در بخش شماره‌ی ۱ از این شکل، سیگنال‌های ماژول<sup>۱</sup> در حال شبیه‌سازی قرار گرفته‌اند. همچنین در بخش شماره‌ی ۲، مقادیر لحظه‌ای هر یک از این سیگنال‌ها در زمانی که خط زمانی زرد رنگ در بخش ۳ بر روی آن قرار گرفته است نشان داده شده است.



شکل ۳۳. نمونه‌ای از یک شبیه‌سازی توسط نرم‌افزار ISIM

شما می‌توانید برای مشاهده و بررسی راحت‌تر شکل موج، تغییراتی در نحوه‌ی نمایش آن برای هر سیگنال ماژول اعمال نمایید (تغییراتی مانند تغییر رنگ، مبنای نمایش مقدار پورت، و...). برای دسترسی به این امکانات، بر روی سیگنال مورد نظر (از بخش ۱ در شکل ۳۳) کلیک راست کرده و مطابق با نیاز خود تغییرات را اعمال کنید. (در ادامه و در شکل ۳۴ به توضیح موارد کاربردی در منوی تنظیمات سیگنال‌ها پرداخته شده است.)

<sup>۱</sup> منظور از سیگنال در این بخش، الزاماً سیگنال‌های درونی که در زبان VHDL استفاده می‌شوند نمی‌باشد و این مفهوم شامل پورت‌های ورودی، خروجی و حتی متغیرها نیز می‌شود.



شکل ۳۴. منوی تنظیمات یک سیگنال در نرم‌افزار ISIM

**Radix:** مبنای ریاضی نمایش مقدار سیگنال (قابل استفاده تنها برای سیگنال‌های چند بیتی)

**Signal Color:** رنگ موج سیگنال

**Riverse Bit Order:** تغییر ترتیب ارزش گذاری (Big-endian یا Little-endian)

**Go To Source Code:** مراجعه به قسمتی از کد آزمایش که این سیگنال در آن تعریف شده است.

**Force Constant:** تغییر مقدار سیگنال به یک مقدار ثابت به صورت اجباری (بدون تغییر در کد آزمایش)

**Force Clock:** تغییر مقدار سیگنال به یک سیگنال پالس ساعت به صورت اجباری (بدون تغییر در کد آزمایش)



## ۱۱. گزارش طراحی

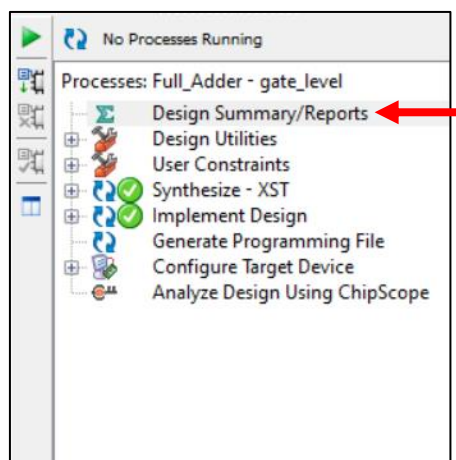
نرم افزار ISE این امکان را به کاربر می دهد که همواره در هر کدام از مراحل طراحی و یا پیاده سازی، خلاصه و گزارشی از طرح خود دریافت کند. در این گزارش اطلاعات بسیار مفیدی راجع به طراحی به کاربر داده می شود که نمونه هایی از این اطلاعات عبارتند از: شرح منابع مصرف شده در تراشه، تأخیر بین هر دو واحد، مسیر حیاتی و میزان تأخیر آن، و غیره (شکل ۳۵). معمولاً در هنگام ساخت پروژه، این پنجره به صورت خودکار در بخش فضای کار باز می گردد. با این حال برای مشاهده ی این گزارش، در هر زمان تنها کافیست بر روی گزینه ی “Design Summary/Reports” در پنجره ی فرآیند (شکل ۳۶) دو بار کلیک کنید.

Full_Adder Project Status (02/14/2020 - 10:16:19)				
Project File:	Full_Adder.xise	Parser Errors:	No Errors	
Module Name:	Full_Adder	Implementation State:	Placed and Routed	
Target Device:	xc6slx9-2tqg144	Errors:		
Product Version:	ISE 14.7	Warnings:		
Design Goal:	Balanced	Routing Results:	All Signals Completely Routed	
Design Strategy:	Xilinx Default (unlocked)	Timing Constraints:		
Environment:	System Settings	Final Timing Score:	0 [Timing Report]	

Device Utilization Summary				
Slice Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Registers	0	11,440	0%	
Number of Slice LUTs	1	5,720	1%	
Number used as logic	1	5,720	1%	
Number using O6 output only	0			
Number using O5 output only	0			
Number using O5 and O6	1			
Number used as ROM	0			
Number used as Memory	0	1,440	0%	
Number of occupied Slices	1	1,430	1%	
Number of MUXCYs used	0	2,860	0%	
Number of LUT Flip Flop pairs used	1			
Number with an unused Flip Flop	1	1	100%	
Number with an unused LUT	0	1	0%	
Number of fully used LUT-FF pairs	0	1	0%	
Number of slice register sites lost to control set restrictions	0	11,440	0%	

شکل ۳۵. پنجره ی گزارش طراحی در فضای کار

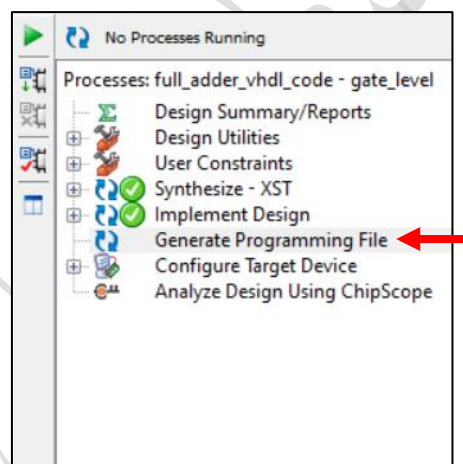


شکل ۳۶. مشاهده ی گزارش طراحی از طریق پنجره ی فرآیند

## ۱۲. تولید فایل برنامه‌ریزی

تمامی اطلاعات مورد نیاز برای برنامه‌ریزی یک تراشه‌ی برنامه‌پذیر (FPGA یا CPLD)، در یک فایل دودویی، که به عنوان فایل جریان بیتی<sup>۱</sup> نیز شناخته می‌شود، ذخیره می‌گردد. پس از انجام تمامی مراحل طراحی و آزمایش، لازم است که این فایل را تولید و برای انتقال به تراشه ذخیره کنید.

برای تولید فایل جریان بیتی، پس از ایجاد source محدودیت‌های پیاده‌سازی (همانطور که در بخش ۵ گفته شد)، تنها کافی است بر روی گزینه‌ی "Generate Programming File" در پنجره‌ی طراحی دو بار کلیک کنید (شکل ۳۷). در صورت موفقیت‌آمیز بودن، یک فایل هم‌نام با نام Top Module پروژه و با پسوند ".bit" در محل پروژه ایجاد و علامت سبز در کنار این گزینه ظاهر خواهد شد.



شکل ۳۷. تولید فایل دودویی برنامه‌ریزی از طریق پنجره‌ی فرآیند

<sup>۱</sup> Bitstream

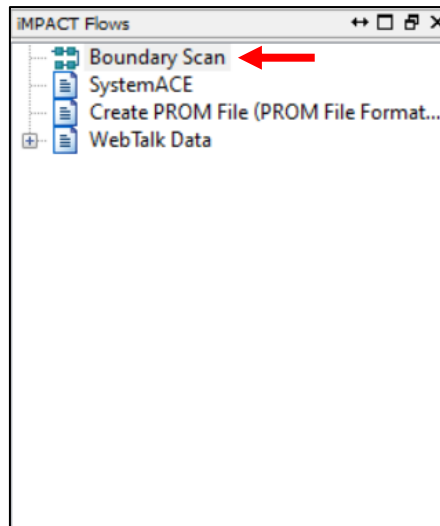
## ۱۳. برنامه‌ریزی تراشه به کمک نرم‌افزار ISE iMPACT

برای انتقال فایل جریان بیتی به تراشه، می‌توانید از نرم‌افزار iMPACT که همزمان با نصب نرم‌افزار ISE نصب می‌شود، استفاده کنید. برای این کار، ابتدا تراشه‌ی خود را با استفاده از پروگرامر به کامپیوتر متصل نموده و در صورت نیاز درایورهای آن را نصب کنید (معمولاً درایورهای رایج همزمان با نصب نرم‌افزار نصب می‌شوند). پس از آنکه از شناخته شدن تراشه توسط کامپیوتر اطمینان حاصل کردید، نرم‌افزار iMPACT را باز کنید. قبل از باز شدن نرم‌افزار، از شما راجع به ذخیره‌ی خودکار پروژه پرسیده خواهد شد که با توجه به نیازتان باید یک گزینه را انتخاب نمایید. پس از گذشتن از این صفحه، بسته به انتخابتان، صفحه‌ی دومی برای شما باز می‌شود. با فشردن کلید "Cancel" آن‌ها را ببندید (این تنظیمات در داخل برنامه نیز در دسترس هستند و نیازی به انجام این تنظیمات در مرحله‌ی اول نیست). پس از انجام مراحل قبلی، وارد صفحه‌ی اصلی نرم‌افزار می‌شوید که مشابه شکل ۳۸ خواهد بود.

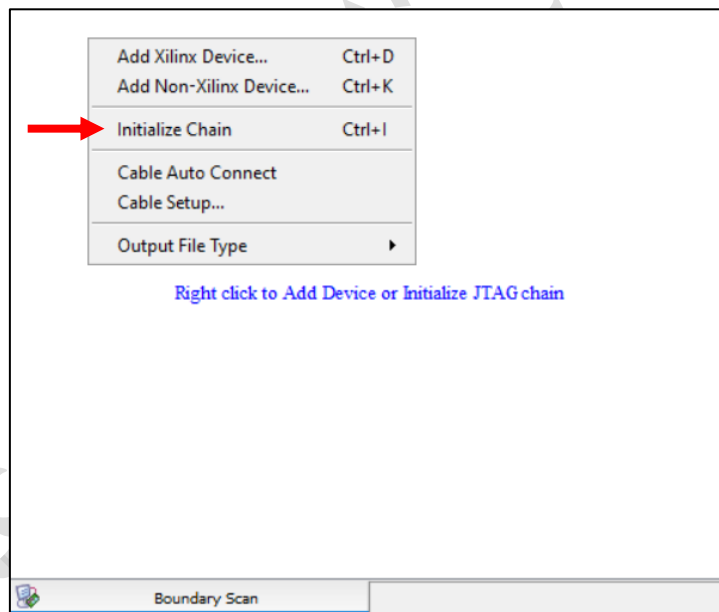


شکل ۳۸. صفحه‌ی اصلی نرم‌افزار iMPACT و بخش‌های آن

برای شروع به برنامه‌ریزی تراشه، از قسمت جریان iMPACT بر روی گزینه‌ی "Boundary Scan" دو بار کلیک کنید (شکل ۳۹) تا پنجره‌ی مربوط به آن در بخش فضای کار باز شود. حال مشابه شکل ۴۰ بر روی پنجره‌ی باز شده در فضای کار کلیک راست کرده و گزینه‌ی "Initialize Chain" را انتخاب کنید.

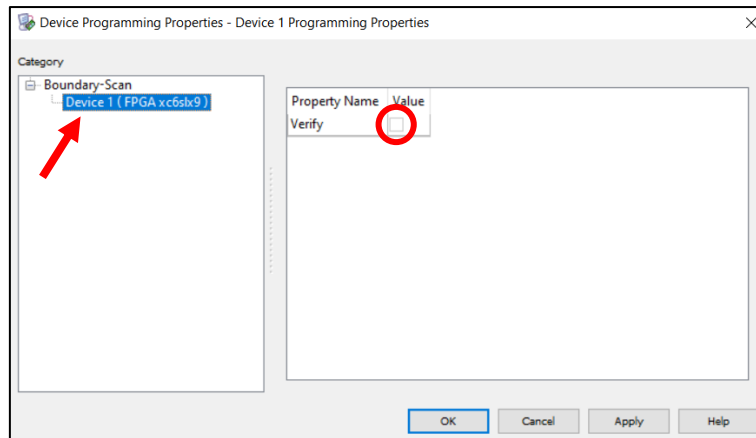


شکل ۳۹. پنجره‌ی جریان iMPACT در محیط نرم‌افزار iMPACT



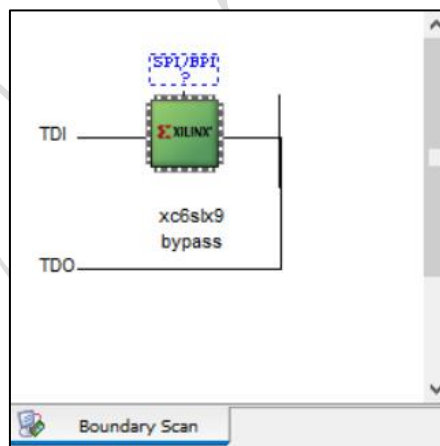
شکل ۴۰. جستجوی زنجیره در برنامه‌ی iMPACT

پس از انتخاب گزینه‌ی “Initialize Chain”، در صورت صحیح بودن اتصالات و نصب بودن درایورها، صفحه‌ی شکل ۴۱ نمایش داده خواهد شد که باید انتخاب کنید آیا قصد Verify شدن تراشه پس از برنامه‌ریزی را دارید یا خیر. به عنوان مثال، در این شکل تنها یک تراشه‌ی اسپارتان ۶ به زنجیره‌ی برنامه‌ریزی متصل شده است و گزینه‌ی “Verify” برای آن انتخاب نشده است (می‌توان چند تراشه را در یک زنجیره به هم متصل نمود و آن‌ها را به صورت همزمان برنامه‌ریزی کرد. برای اطلاعات بیشتر، درباره‌ی زنجیره‌ی JTAG مطالعه کنید).



شکل ۴۱. تنظیمات برنامه‌ریزی تراشه‌های متصل

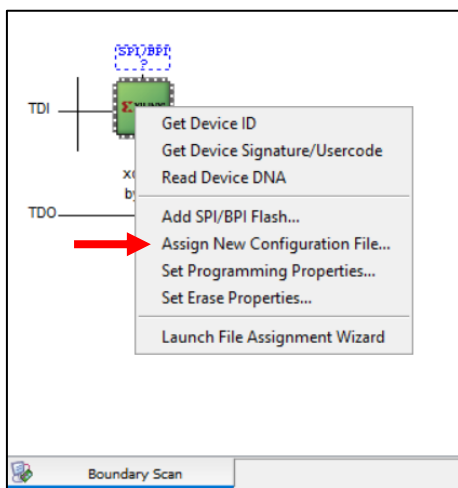
پس از انتخاب گزینه‌ی تأیید و خروج از پنجره‌ی شکل ۴۱، دوباره به صفحه‌ی اصلی نرم‌افزار باز خواهید گشت که در آن دستگاه/دستگاه‌هایی که توسط پروگرامر به کامپیوتر متصل است، مانند شکل ۴۲ نمایش داده خواهند شد. همانطور که در این شکل مشاهده می‌کنید، در این لحظه یک تراشه با شناسه‌ی XC6SLX9 به دستگاه متصل است.



شکل ۴۲. نمایش تراشه‌های متصل به زنجیره

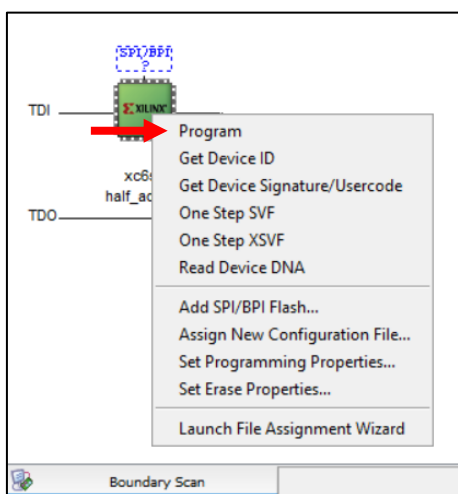
می‌دانیم که اکثر تراشه‌های FPGA برای برنامه‌ریزی از حافظه‌های فرار استفاده می‌کنند، به همین دلیل همواره در کنار آن‌ها از یک حافظه‌ی غیر فرار نیز استفاده می‌شود تا فایل برنامه‌ریزی را ذخیره کرده و در هر بار روشن شدن دستگاه، آن را بر روی تراشه بارگذاری کند. در نتیجه در هنگام برنامه‌ریزی تراشه، فایل جریان بیتی باید هم به تراشه و هم به حافظه‌ی غیر فرار منتقل شود (انتقال همزمان به این دو تراشه، به کمک زنجیره‌ی JTAG امکان‌پذیر است). در این نسخه از آموزش به برنامه‌ریزی حافظه‌ی غیر فرار پرداخته نخواهد شد و فایل جریان

بیتی تنها به خود تراشه منتقل می‌شود. برای این کار، بر روی تراشه‌ی مورد نظر کلیک راست کرده و مطابق شکل ۴۳ گزینه‌ی “Assign New Configuration File” را انتخاب کنید. حال یک File Dialog برای شما باز می‌شود. به آدرس ذخیره‌ی پروژه رفته و فایل جریان بیتی را انتخاب کنید، از شما راجع به برنامه‌ریزی حافظه‌ی غیر فرار پرسیده می‌شود که آن را ببندید.



شکل ۴۳. انتخاب فایل برنامه‌ریزی تراشه

پس از اینکه فایل برنامه‌ریزی را مشخص کردید، دوباره بر روی تراشه‌ی مورد نظر کلیک راست کرده و این بار مطابق شکل ۴۴ گزینه‌ی “Program” را انتخاب کنید. در پایان در صورتی که خطایی رخ نداده باشد، پیغام “Program Succeed” بر روی فضای کار نمایش داده خواهد شد که نشان دهنده‌ی موفقیت‌آمیز بودن برنامه‌ریزی می‌باشد.



شکل ۴۴. برنامه‌ریزی تراشه

This learning material has been developed for educational purposes provided to the students of the University of Sistan and Baluchestan, Iran at no cost. The trademarks, logos, service marks displayed, and the software used on this tutorial are the property of Xilinx or other third parties.

<b>Last Edit</b>	<b>Version</b>
February 2021	1.8

Special thanks to Dr. Mohammad Hossein Sargolzaei, Head of the Department of Computer Engineering, for his endeavor to help finalize and publish this document.



**University of Sistan and Baluchestan**  
**Faculty of Electrical and Computer Engineering**

## **Xilinx ISE Design Suite v14.7 Tutorial**

**Author:**  
Ali Abbasi