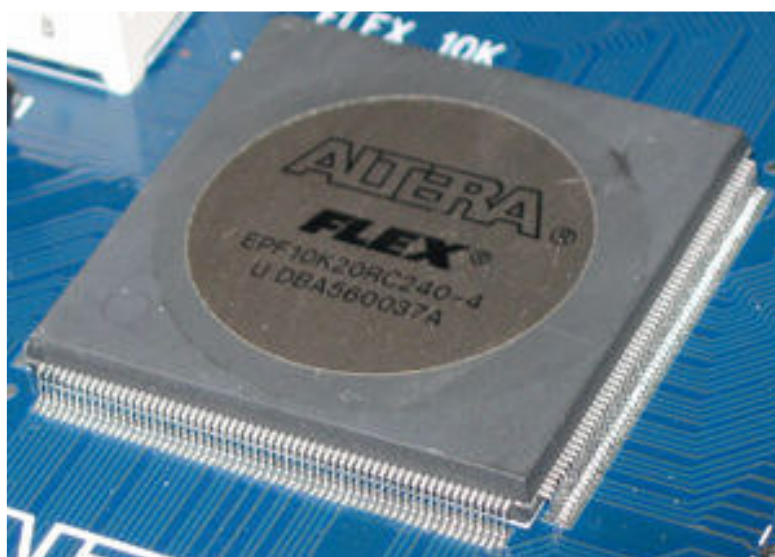


بسمه تعالی

## FPGA و تاریخچه آن



### مقدمه

سالها پیش که طراحی دیجیتال پا به عرصه وجود نهاد و IC های استاندارد چگونگیها، فلیپ فلاپ ها، لچ ها شمارنده ها و... ساخته شدند به تدریج پردازنده هایی با قدرت محدود که اولین کامپیوتر های شخصی بر اساس آنها طراحی شده بود دنیای دیجیتال را به وجود آوردند، تصور روزی که فاصله سخت افزار و نرم افزار به حد کنونی برسد به طوری که تمام مرزهای طراحی را در نوردیده و سخت افزار به نرمی و انعطاف پذیری درآید بسیار دشوار بود.

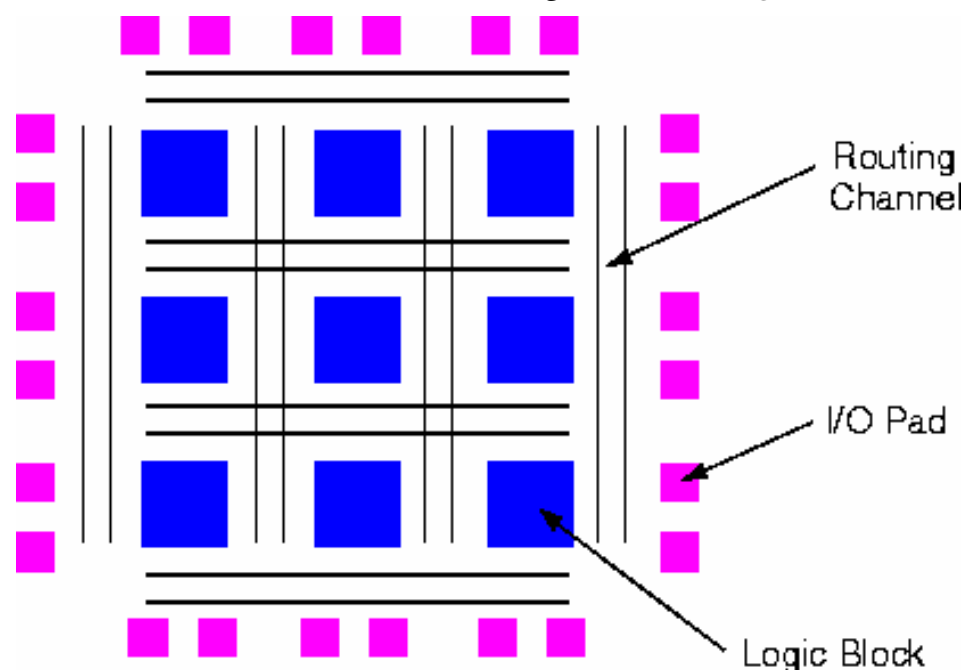
اما بعد ها با طراحی حافظه های قابل برنامه ریزی دوباره و فن آوری EPROM حافظه های پایای با قابلیت برنامه ریزی و پاک سازی و PAL آرایه های منطقی قابل برنامه ریزی، و سرانجام فن آوری آرایه های سوئیچ های فیوزهای قابل برنامه ریزی چندباره، انقلابی نوین را در عرصه طراحی دیجیتال به وجود آورد که مفهوم طراحی دیجیتال را دچار تحولی عظیم در عرصه های دیدگاه معماری، حجم طراحی، سرعت و نوع نگرش به طراحی دیجیتال نموده است.

به طوری که امروزه FPGA ها آرایه های گیتی قابل برنامه ریزی میدانییک بوم نقاشی سفید را در اختیار طراح قرار می دهند که به او اجازه می دهد تا طراحی دیجیتال خود را آنچنان که می خواهد و با هر حجم و پیچیدگی لازم، طراحی و سپس به جای انتخاب IC های استاندارد و جدا از هم و کنار هم قرار دادن آنها در روی یک مدار و وصل کردن آنها از طریق یک برد مدار چاپی (PCB)، با استفاده از یکی از زبانهای توصیف سخت افزاری نظیر VHDL، هر یک از قطعات دیجیتالی مورد نیاز را نوشته و با وصل کردن نرم افزاری آنها، سرانجام فایل کامپایل شده نهایی را از طریق یک رابط سخت افزاری بر روی یک بسته سخت افزاری خام با تعداد پایه های مورد نیاز برنامه ریزی کرده و از این IC جدید "خود ساخته" استفاده کند.

### بررسی ساختار داخلی

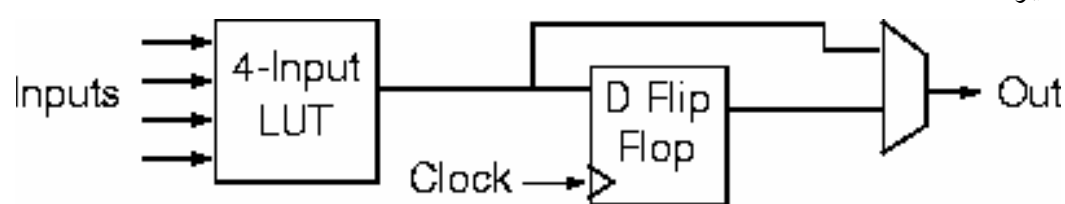
FPGA را می توان به صورت جزیره های مجزایی در نظر گرفت که توسط شاهراههایی به هم متصل می گردند. به عبارتی FPGA شامل یک سری بلوک منطقی و نیز سیم های بین آنها می گردد.

دو پد ورودی و خروجی نیز در انتهای هر یک از ردیف‌ها یا ستونها قرار داده شده است. خطوط اتصال دهنده بین بلوکها از نظر تعداد و اندازه یکسان می‌باشند که در شکل زیر نیز نشان داده شده است.

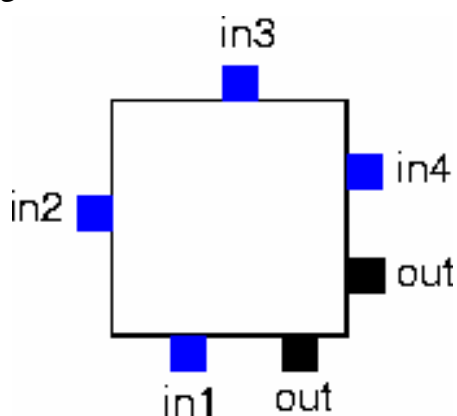


هر مدار منطقی که روی - FPGA با استفاده از پروگرامر ریخته می‌شود طوری روی FPGA پیاده می‌شود که با استفاده از بلوکهای منطقی و مسیرها بین این بلوکها بتواند تابع داده شده را اجرا نماید.

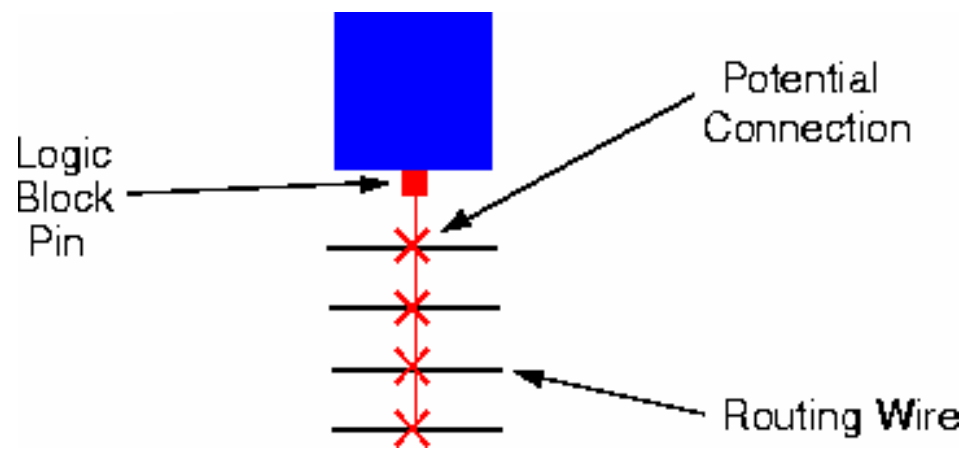
هر بلوک منطقی در FPGA دارای 4 ورودی به جدول مراجعه‌ای (Look Up Table-LUT) و یک فلیپ فلاپ، می‌باشد - در شکل زیر نشان داده شده است:



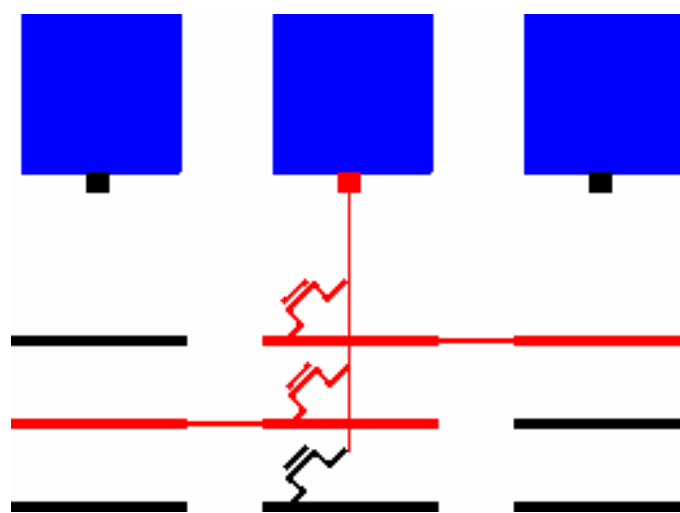
تنها یک خروجی برای این بلوک وجود دارد که می‌تواند خروجی ثبت شده (Registered) یا ثبت نشده (UnRegistered) قبلی را به خارج انتقال دهد. از آنجایی که پایه Clock به طور داخلی سیم کشی شده است - در نوع تجاری -، می‌توانید این پایه را نادیده بگیرید و در نهایت هر بلوک به شکل زیر خواهد بود.



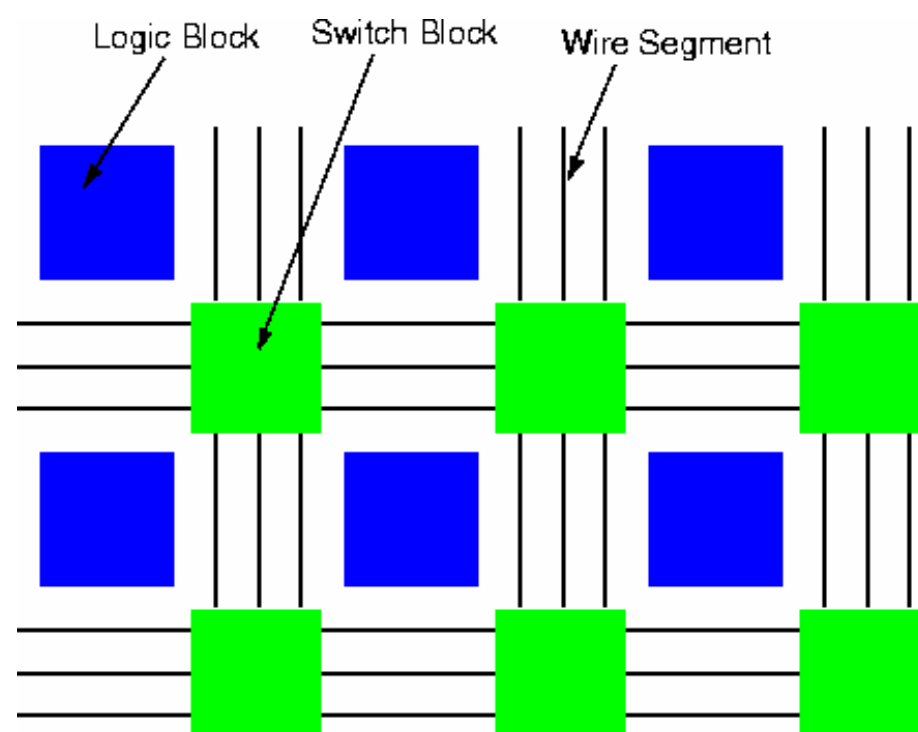
وضعیت هر پایه ورودی به گونه ای است که قابلیت اتصال به هر یک از بخش‌های کانال را داشته باشد شکل زیر به طور کامل وضعیت آن را بیان می‌دارد:



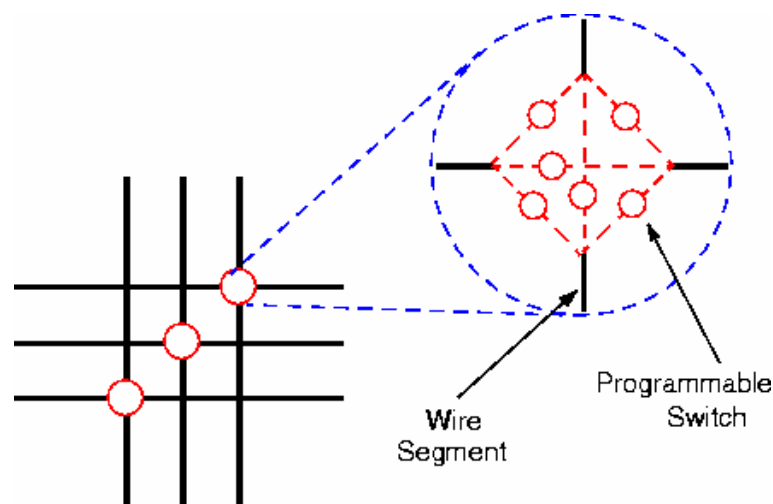
در شکل زیر نیز نمایی از طریقه برقراری این اتصالات آورده شده است. این کار را میتوان به راحتی با استفاده از سویچ های MOSFET انجام داد. با فرمان به هر قسمت می توان آن مسر را به پایه متصل نمود.



تعریف خطوط داخلی یا به عبارتی سیم کسی داخلی FPGA به طور کامل از هم مجزا نیستند بلکه با استفاده از سویچهایی این خطوط به هم راه می یابند. از این طریق می توان از تعداد خطوط کمتری برای اتصال میان بلوکها استفاده کرد. شکل زیر این مطلب را روشن تر بیان می دارد:



سویچ‌هایی که برای استفاده در کانالها استفاده می‌شوند قابلیت برنامه ریزی داشته و با توجه به برنامه و بلوک بندی آنها برنامه ریزی می‌شوند. اگر فرض کنیم که تعداد خطوط بین بلوکها 3 باشد می‌توان شکل این سوچها را به صورت زیر نشان داد:



معمولاً تکنولوژی ساخت چنین سویچ‌هایی از نوع *planer* یا *domain-based switch* می‌باشد.

توجه داشته باشد در تغییر مسیرها همان گونه که در شکل نیز نشان داده شده است سیم شماره 1 تنها می‌تواند به همان شماره از سیم (1) در مسیر بعدی متصل گردد - از 1 به 2 در این سویچها امکان پذیر نیست و مسیر اصلی همچنان حفظ می‌گردد.

#### خلاصه‌ای از چگونگی عملکرد و کاربرد:

شاید تا بحال مدارهای منطقی را بوسیله گیت‌های، *AND*, *OR*, *NOT* ساخته اید. برای ساخت چنین مدارهایی (از قبیل شمارنده‌ها کنترل کننده‌ها، و ...) ابتدا باید تعریفی از مدار در دسترس باشد سپس با توجه به منطق اعداد دودویی یک جدول صحت برای مدار تشکیل می‌شود و حالت‌های مختلف مورد بررسی قرار می‌گیرد سپس با توجه به جدول صحت مدار توسط گیت‌های منطقی مانند *AND*, *OR*, *NOT*, *NAND* طراحی می‌شود پس از این مرحله نوبت به پیاده سازی مدار بر روی برد توسط آی سی های منطقی می‌رسد و همانطور که می‌دانید یکی از وقت گیرترین و خسته کننده ترین مرحله ساخت یک مدار همین قسمت است. بعد از این مرحله نوبت به تست مدار جهت اطلاع از درستی مراحل کار کرد مدار می‌رسد. اگر در یکی از مراحل قبل دچار اشتباه شده باشیم مطمئناً در مرحله تست مدار دچار مشکل می‌شویم. در صورت اشتباه در مراحل قبل باید تمام مراحل را از آخر به اول یک به یک چک کنیم تا بتوانیم اشتباهات احتمالی موجود در نحوه بستن و سیم کشی مدار، طراحی مدار را از روی جدول صحت و درستی جدول صحت را برطرف کنیم. با توجه به مطالب گفته شده حتماً به این نکته اذعان خواهید داشت که بیشترین اشتباهات در مرحله سیم کشی و بستن مدار بر روی برد پیش خواهد آمد.

ممکن است سیمی در جای اصلی وصل نشده باشد و یا ممکن است یک پایه به هیچ جا متصل نباشد و یا اشتباهات مشابه اینها... از طرف دیگر می‌دانیم که هر چه مدار بزرگتر و پیچیده تر باشد اشتباهات بیشتر و عیب یابی مشکل تر خواهد بود. اینجاست که نقش آی سی های *FPGA* نمایان ترمی شود. آی سی هایی که با داشتن انواع گیت‌های مختلف درون خود بسیاری از مشکلات ناشی از عیب یابی مدارهای منطقی را برطرف کرده است.

قابلیت و توانایی‌ها:

اما آنچه که قابلیت و توانایی **FPGA** ها را بالا برده است توانایی هایی است که پاره ای از آنها در زیر آمده است :

1. امکان تعریف هر یک از پایه های **IC** به صورت ورودی یا خروجی یا هر دو.

2. امکان تعریف وضعیت عملکرد هر پایه در هنگام استفاده یا عدم استفاده. به عنوان مثال عملکرد **HIGH** امپدانس (**Z**) در هنگام عدم استفاده و یا قرار گرفتن در یک وضعیت منطقی صفر یا یک در هنگام عدم استفاده.

3. امکان تشخیص تغییرات سطوح یا لبه های پایین رونده یا بالا رونده منطقی اعمال شده به هر پایه.

4. امکان برنامه ریزی چند باره از طریق پایه های برنامه ریزی **JTAG** یکی از استانداردهای برنامه ریزی (**IEEE**) و تغییر معماری آن).

5. امکان تغییر متناوب معماری داخلی با استفاده از سری های **Bootable** که نقشه معماری آنها در یک حافظه خارجی نگهداری شده و با تغییر آدرس برنامه ریزی می توان **IC** را با معماری جدید **Boot** کرده و از آن استفاده کرد.

6. امکان برنامه ریزی در مدار (**ISP**) که این قابلیت را به وجود می آورد تا بدون اعمال تغییرات سخت افزاری و تنها از طریق پورت برنامه ریزی **JTAG** معماری داخلی **IC** را تغییر داد.

7. محدوده گستره‌ای از پایه های قابل استفاده در این **IC** ها که از بسته های **44** پایه تا **514** پایه و حتی بالاتر با حجم گیتی داخلی متفاوت که بسته به نیاز بر اساس میزان پیچیدگی داخلی و تعداد پایه های **IC** را تغییر داد.

8. کککش حیرت انگیز حجم مدار و مجتمع سازی در ابعادی تنها به مساحت چند سانتی متر مربع.

9. یکسان سازی عناصر طراحی و از میان بردن تمامی مشکلات ناشی از عدم تطابق استاندارد های مختلف (**LS, HC, s, AS, ...**).

10. از میان بردن تمامی نویز های ناشی از وجود قطعات مختلف و مجزا در مدار.

11. کککش چشمگیر توان مصرفی و اتلاف توان.

12. افزایش سرعت پردازش و خطکهای انتشار به دلیل استفاده از فناوری پیشرفته و دستیابی به خطکهای انتشار تا **4ns** و فرکانس کلاک فراتر از **178** مگهرتز.

13. کار با دو سطح ولتاژ **5v** و **3.3v** جهت استفاده از آنها در دستگه های قابل حمل مانند گوشی های موبایل

۱۴. ضریب ایمنی صد در صد به دلیل عدم امکان دستیابی به محتوای داخلی و عدم توان توصیف محتوای داخلی به دلیل انجام ساده سازی و فشرده سازی بسیار پیچیده.  
و بسیاری از قابلیت‌های حیرت انگیز دیگر که امکان انجام یک طراحی مجتمع، کم حجم، بهینه و سریع را فراهم می آورد.

#### کارخانهای تولید کننده محصول:

گرچه شرکتهای بسیاری بسته های FPGA را تولید می کنند اما از میان آنها در شرکت ALTERA و Xilinx از جمله عمده ترین تولید کنندگان این محصول هستند که از این میان شرکت Xilinx نوع دیگری از این بسته ها را با نام CPLD را تولید می کند که به صورت Bootable عمل می کنند، بدین معنی که داده های برنامه ریزی معماری داخلی خود را از یک حافظه ی خارجی خوانده و خود را پیکر بندی کرده و سپس آماده کار می شوند. تمامی این محصولات با توجه به تعداد پایه ها و حجم پیچیدگی قابل برنامه ریزی در آنها بر اساس تعداد گیت‌های داخلی در بازار موجود و قابل دسترس هستند. از جمله سری های پر قدرت و پر حجم آنها سری flex از محصولات شرکت ALTERA که در نمونه ای از آن می توان یک CPU مدل 486 را جای داد.

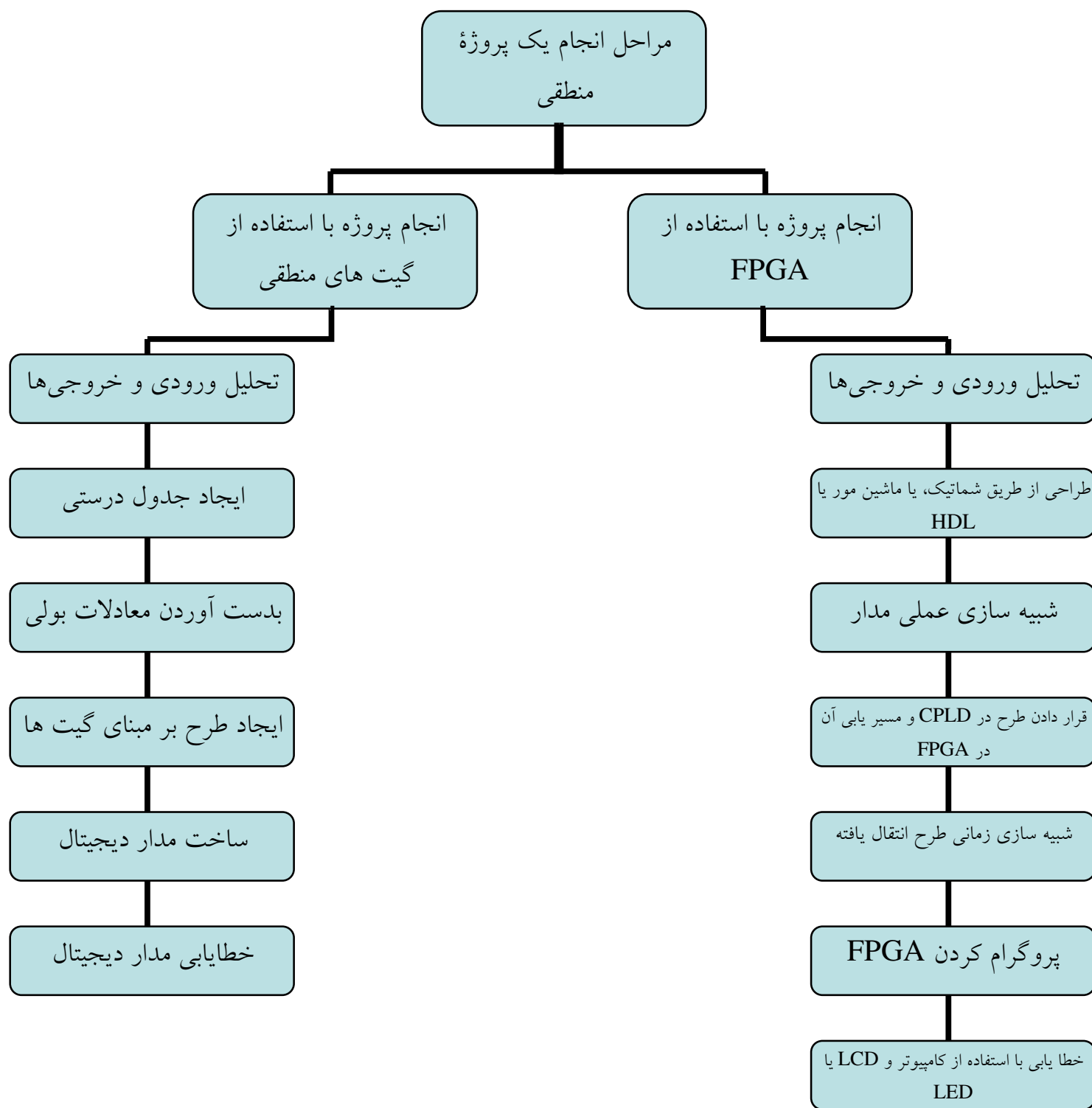
#### زبانهای برنامه نویسی:

از جمله زبانهای متداول برنامه نویسی سخت افزار ABEL, AHDL, VERILOG, VHDL هستند که هر یک با استفاده از syntax خاص خود برای توصیف سخت افزار مورد استفاده قرار می گیرند که با استفاده از هر یک از آنها می توان هر طراحی دیجیتالی را به زبان آنها نوشته و تحلیل و سپس استفاده کرد.  
نمونه ای از این آی سی در زیر آورده شده است.

#### نرم افزارهای مربوط:

معمولاً هر یک از کارخانه‌های تولید این آی سی نرم‌افزاری را جهت استفاده از تولیدات معرفی می کند. معروف ترین و پر کاربرد ترین این نرم‌افزارها مربوط به شرکت Xilinx می باشد. که کلیه مراحل اعم از شبیه سازی پردازش و پروگرام کردن در آن قابل اجرا می باشد.  
از نرم افزارهای دیگر می توان به Protel DXP اشاره کرد که همه قابلیت های فوق را دارد علاوه بر آن می تواند نقشه PCB آن را نیز ارائه دهد.

در فلوجارت زیر مقایسه بین روشهای انجام پروژه با استفاده از گیت‌های منطقی و یا FPGA با هم مقایسه شده است.



شرکت ALTRA نیز مراحل مختلف مربوط به استفاده از آی سی تولید خود را به صورت زیر بیان داشته است. برای داشتن مقایسه با Xilinx مناسب می باشد.

