

## دستور کار آزمایشگاه مجازی مدارهای منطقی و معماری کامپیوتر

- پیاده سازی جمع کننده چهار بیتی در محیط Xilinx ISE

### رویکرد کلی آزمایش:

ابتدا یک جمع کننده ۴ بیتی Ripple Carry را پیاده سازی می کنیم. برای آشنایی با ابزار ISE، با شبیه سازی Behavioral عملکرد کدهایمان را تست کرده و سپس تاخیرهای بحرانی آن را بررسی می کنیم. همین فرآیند را برای جمع کننده ۴ بیتی به روش Carry Look-Ahead اجرا کرده و نتایج بدست آمده از دو مدار را مقایسه می کنیم.

### بخش اول: ساخت یک جمع کننده چهار بیتی Ripple Carry :

الف- ابتدا یک پروژه به نام ripple\_adder4 بسازید و فایل‌های کد جمع کننده را به آن بیفزایید.

ب- سپس با سنتز کردن آن روی Spartan 3E-XC3S100E-VQ100-5 ، صحت Syntax آن را بسنجید.

پ- شماتیک RTL و Technology آن را استخراج و شکل آن را ذخیره کنید.

ت- یکی از LUT های میانی مدار را به دلخواه انتخاب کنید. اطلاعات تابع پیاده سازی آن را در قالب های زیر ثبت کنید:

- شماتیک
- جدول کارنو
- جدول صحت
- معادله منطقی

ث- تاخیر بحرانی مدار را با استفاده از گزارشات زمانی استخراج کنید و نتایج تاخیر بین کلیه پایه ها را در گزارش خود ثبت کنید.

ج- یک Fixture برای این مدار طراحی کنید و به ترتیب با اختلاف زمانی ۱۰۰ نانو ثانیه مقادیر زیر را اعمال کنید:

زمان (ns)	A (4bit)	B (4bit)	C <sub>in</sub> (1bit)
0	0011	1100	0
100	0011	1100	1
200	0001	1111	1

چ- صحت عملکرد مدار را در حالت شبیه سازی Behavioral بسنجید ( از مقادیر سیگنال‌های خروجی و ورودی در شبیه ساز در مقاطع زمانی مناسب عکس بگیرید و در گزارشات ثبت کنید)

ح- با استفاده از Design Summary میزان استفاده از انواع LUT ها، تعداد Slice استفاده شده و سایر منابع در سطح FPGA را برای این مدار ثبت کنید.

## بخش دوم: ساخت یک جمع کننده چهار بیتی Carry Look Ahead

مراحل الف تا ح بخش اول را برای مدار جمع کننده Carry Look Ahead تکرار کنید.

## بخش سوم: مقایسه کارآیی

الف- تأخیر عملکرد دو مدار را با هم مقایسه کنید.

ب- میزان منابع استفاده شده در این دو مدار را با هم مقایسه کنید.

\* مندرجات در گزارش :

- مطابق بند های خواسته شده در دستور کار
- تحلیل نتایج بدست آمده