**به نام خدا**

****

**گزارشکار دستورکار شماره 3**

**ارائه دهندگان:**

**زهره ابوعلی شمشیری**

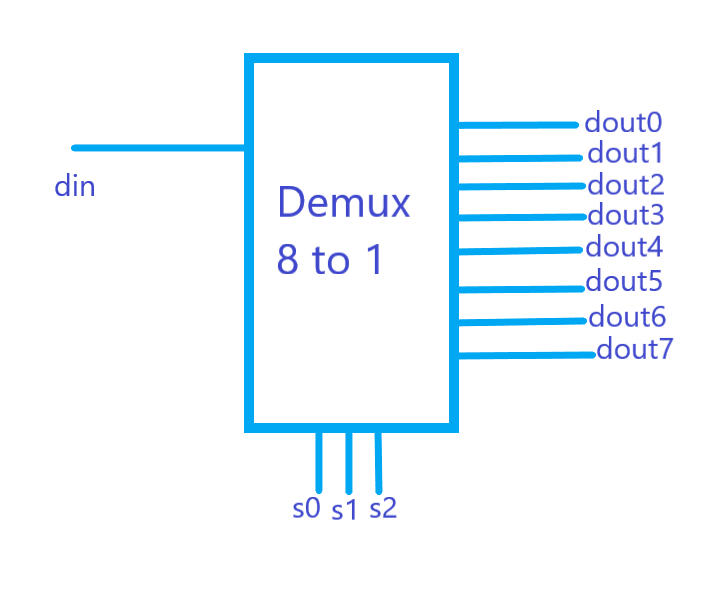
**امین چیت سازان**

**نام استاد: مهندس سید مجتبی موسوی**

**اردیبهشت 1402**

**آزمایش اول: پیاده سازی دی مالتی پلکسر 1 به 8**

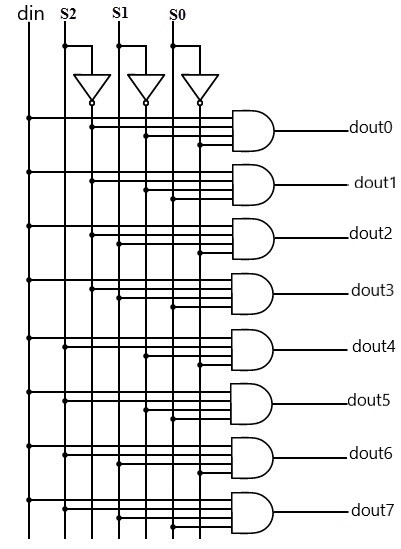
دی مالتی پلکسر یک مدار ترکیبی است که شامل یک ورودی و چندین خروجی است. خط(خطوط) انتخاب این مدار تعیین می کند که ورودی به کدام خروجی منتقل شود.دی مالتی پلکسر 8 به 1 شامل یک ورودی و 8 خروجی و سه خط انتخاب است.

****

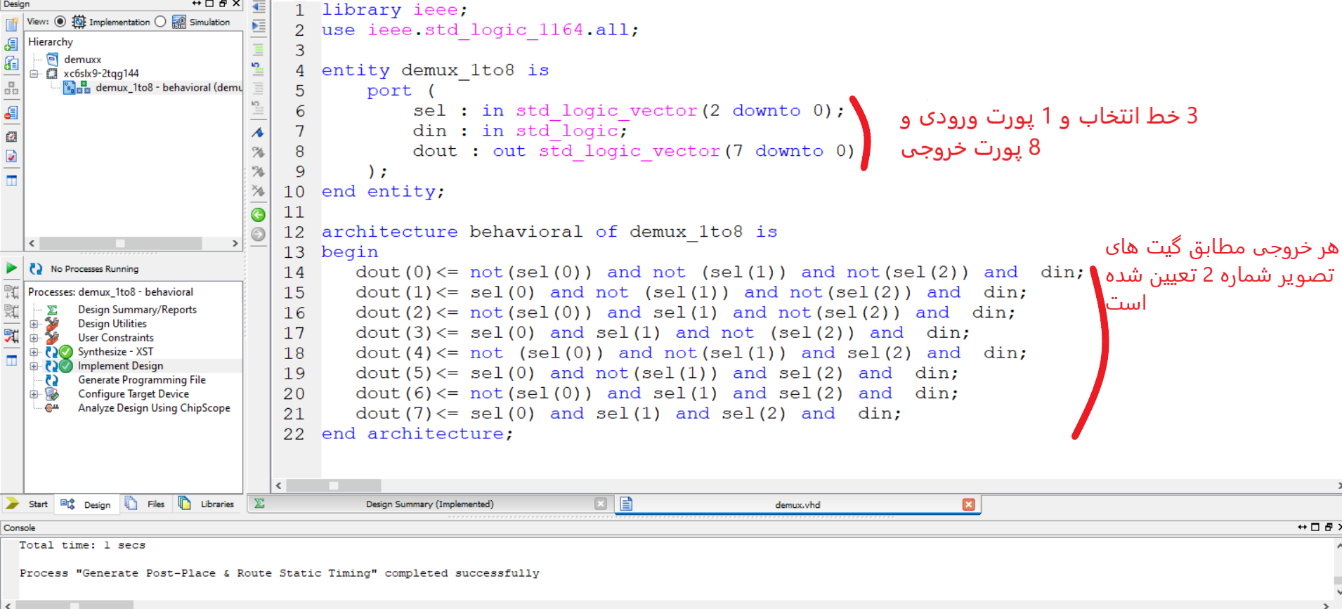
**تصویر 1- شماتیک دی مالتی پلکسر 1 به 8**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Dout71** | **Dout61** | **Dout51** | **Dout41** | **Dout31** | **Dout21** | **Dout11** | **Dout01** | **Sel2** | **sel1** | **Sel0** |
| 0 | **0** | **0** | **0** | **0** | **0** | **0** | **din** | **0** | **0** | **0** |
| 0 | **0** | **0** | **0** | **0** | **0** | **din** | **0** | **1** | **0** | **0** |
| 0 | **0** | **0** | **0** | **0** | **din** | **0** | **0** | **0** | **1** | **0** |
| 0 | **0** | **0** | **0** | **din** | **0** | **0** | **0** | **1** | **1** | **0** |
| 0 | **0** | **0** | **din** | **0** | **0** | **0** | **0** | **0** | **0** | **1** |
| 0 | **0** | **din** | **0** | **0** | **0** | **0** | **0** | **1** | **0** | **1** |
| 0 | **din** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **1** | **1** |
| din | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **1** | **1** | **1** |

**جدول1- جدول درستی دی مالتی پلکسر1 به 8**

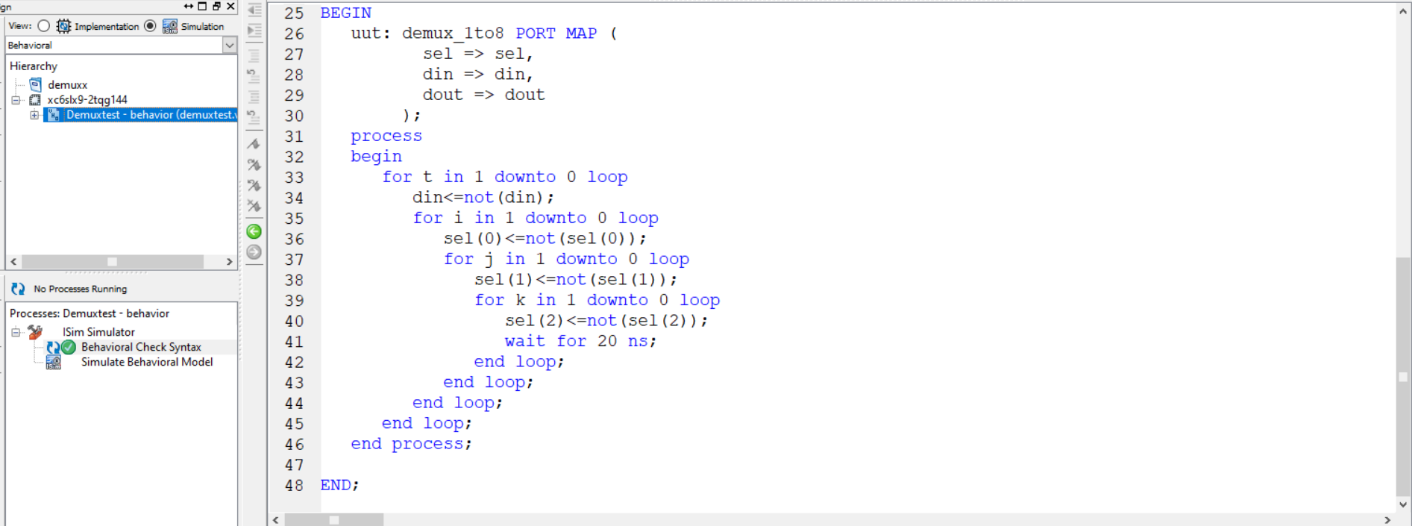
****

**تصویر 2- شماتیک مالتی پلکسر 8 به 1 با گیت های منطقی**

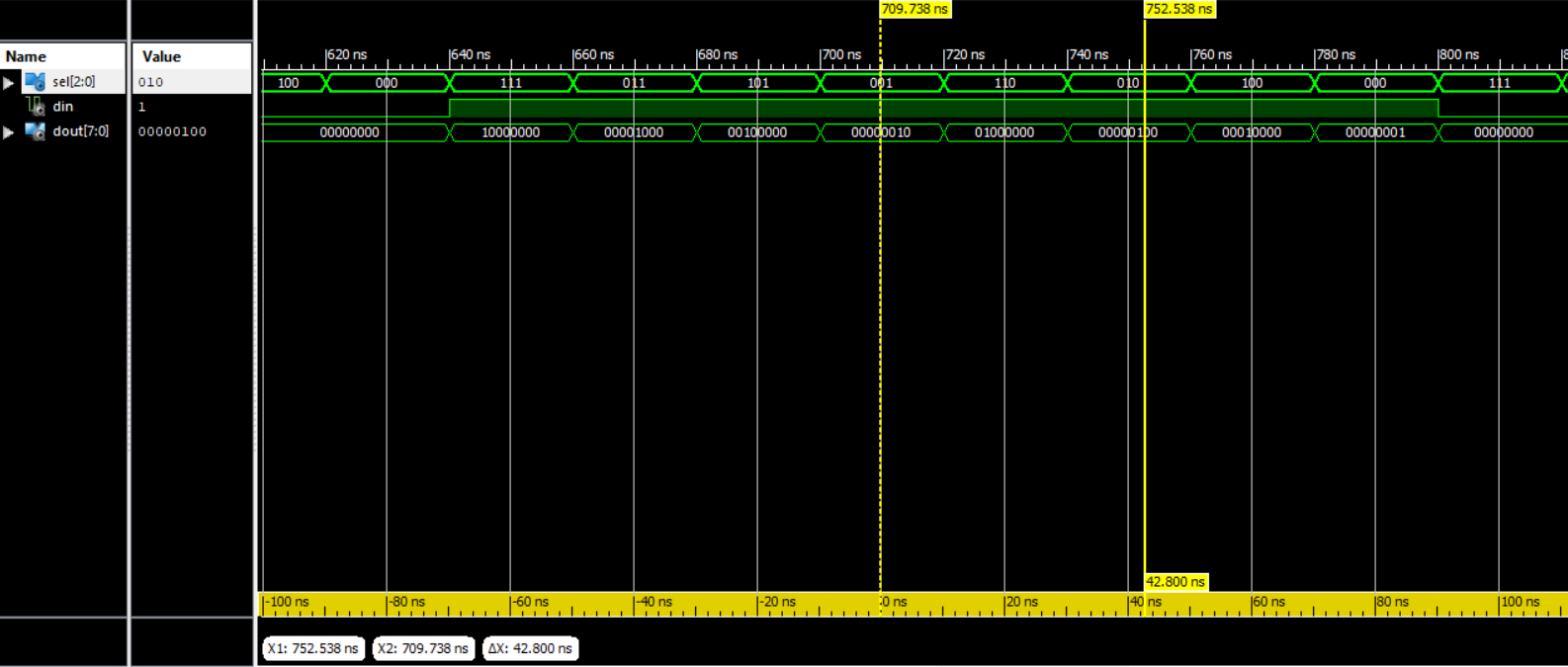
****

**تصویر 2- کد VHDL دی مالتی پلکسر 1 به 8**

**تست بنچ دی مالتی پلکسر 1 به 8:** کد تست بنچ برای ایجاد تست کیس ها جهت بررسی عملکرد مدار طراحی شده، باید در بخش process و پایین بخش begin نوشته شود. با استفاده از حلقه های تو در تو، در هر حلقه یکی از خط های انتخاب را با عملگر not نقیض کرده و تمامی حالت ها را برای خطوط انتخاب متفاوت بررسی می کنیم. همچنین میتوان din را نیز نقیض کرد.

****

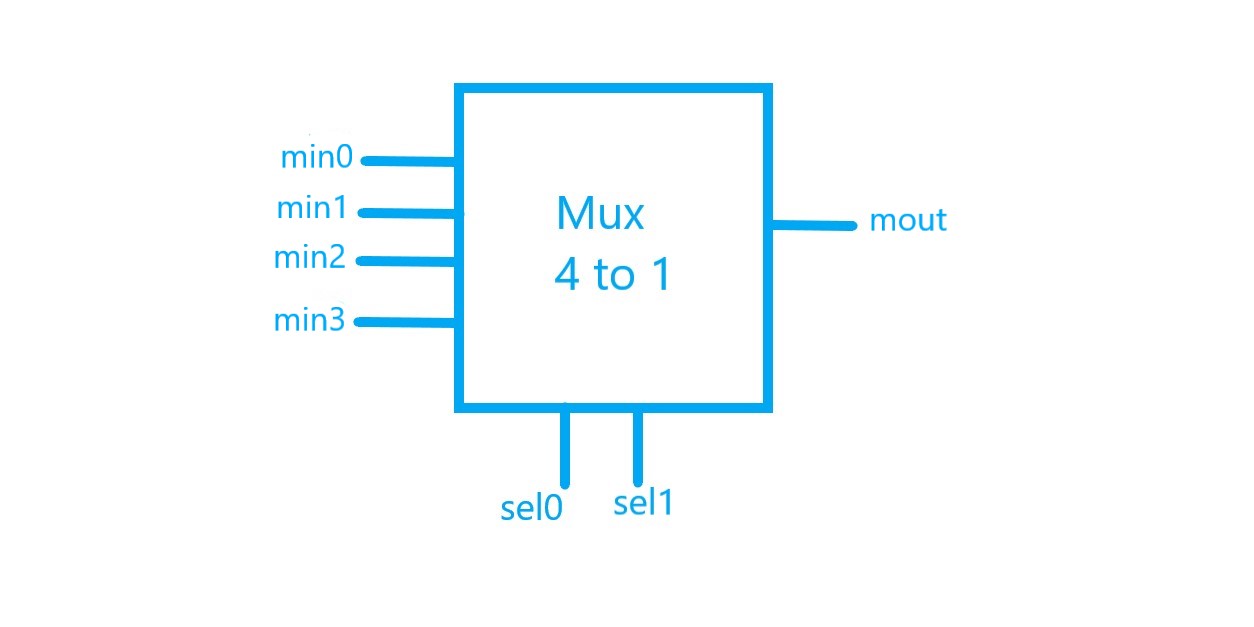
**تصویر 3 – تست بنچ دی مالتی پلکسر 1 به 8**

****

**تصویر 4-شبیه سازی دی مالتی پلکسر 8 به 1 با استفاده از کد تست بنچ و اجرای تست کیس های متفاوت**

**آزمایش دوم : پیاده سازی مالتی پلکسر 4 به 1 تک بیتی با استفاده از ساختار case**

مالتی پلکسر یک مدار ترکیبی است که شامل چندین ورودی و یک خروجی است.خط(خطوط) انتخاب می کند کدام ورودی به خروجی منتقل شود.مالتی پلکسر 4 به یک شامل 4 ورودی و 1 خروجی و 2 خط انتخاب است.

****

**تصویر 5- شماتیک مالتی پلکسر 4 به 1**

|  |  |  |
| --- | --- | --- |
| **mout** | **Sel1** | **Sel0** |
| min0 | **0** | **0** |
| min1 | **1** | **0** |
| min2 | **0** | **1** |
| min3 | **1** | **1** |

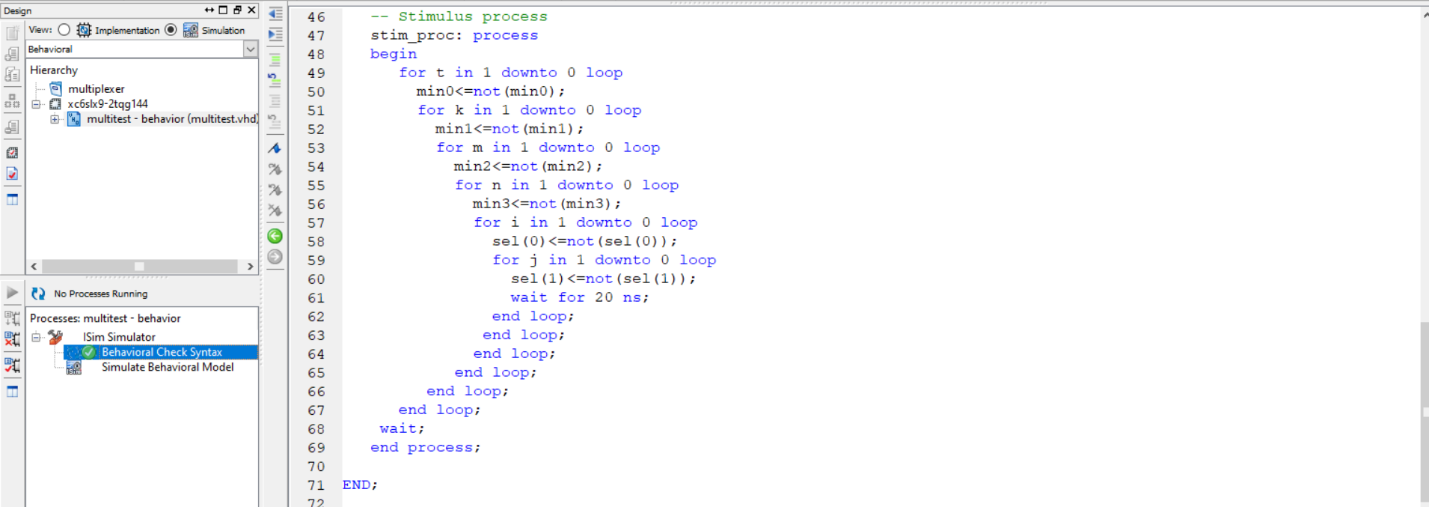
**جدول 2- جدول درستی مالتی پلکسر 4 به 1**

در مالتی پلکسر با توجه به خطوط انتخاب،یکی از 4 ورودی انتخاب میشود به همین دلیل از از ساختار case برای ساختن این مدار استفاده میکنیم. ساختار case باید در process باشد که با هر بار تغییر خط انتخاب یا ورودی خروجی مطلوب انتخاب شود. اگر خط سلکت "00" باشد،min0 (اولین پورت ورودی) به خروجی منتقل میشود.اگر خط سلکت "01"باشد،min1 (پورت دوم ورودی) به خروجی منتقل میشود.اگر خط سلکت "10" باشد min2 (پورت سوم ورودی) به خروجی منتقل میشود.اگر خط سلکت "11" باشد،min3 (پورت چهارم ورودی) به خروجی منتقل میشود و اگر به دلایلی مثل وصل نبودن پورت به سخت افزار و ... مقادیر خطوط انتخاب غیر از این ها بودند مقداری را برای از کار نیفتادن مدار به خروجی منتقل میکنیم.(تصویر 6 و جدول 2)

****

**تصویر 6 – کد VHDL مالتی پلکسر 4 به 1**

**تست بنچ مالتی پلکسر 4 به 1:** با استفاده از حلقه های تو در تو و نقیض کردن مقادیر خطوط انتخاب و همچنین ورودی های مدار با عملگر not،حالات مختلف را بررسی میکنیم.

****

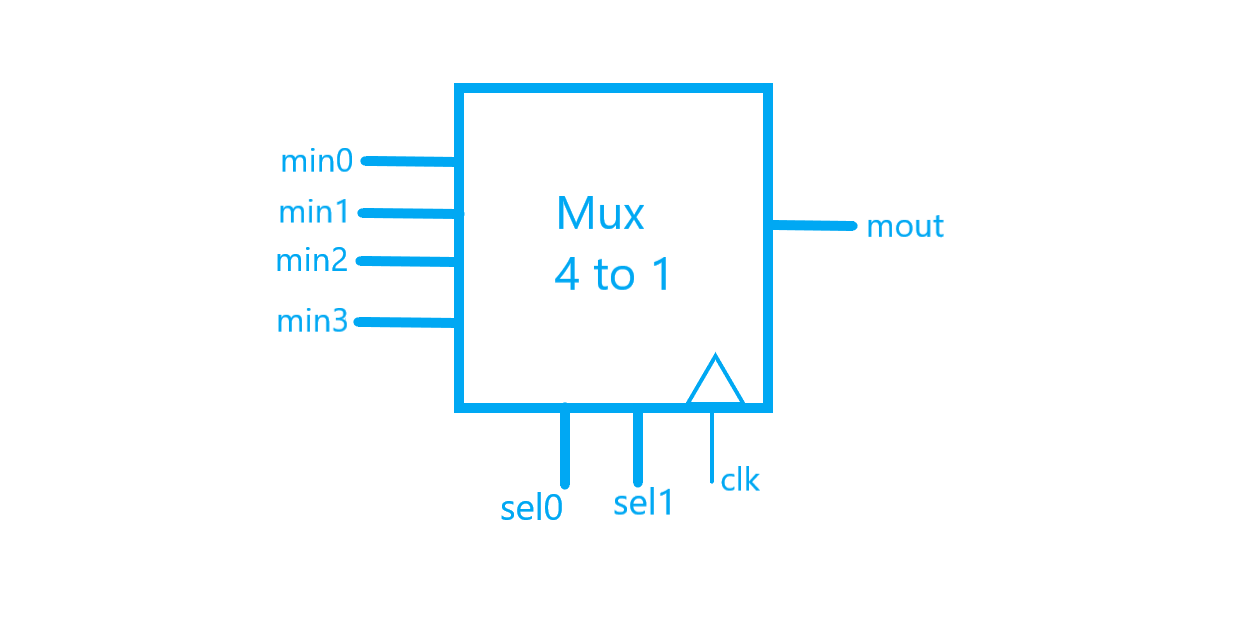
**تصویر 7- تست بنچ مالتی پلکسر 4 به 1**

****

**تصویر 8-شبیه سازی مالتی پلکسر 4 به 1 با استفاده از کد تست بنچ و اجرای تست کیس های متفاوت**

**آزمایش سوم : پیاده سازی مالتی پلکسر 4 به 1 تک بیتی با کلاک با استفاده از ساختار if**

این مدار مانند نوع بدون کلاک آن است با این تفاوت که در نوع با کلاک، فقط در زمانی که کلاک 1 (بالارونده) باشد خروجی مالتی پلکسر تغییر میکند.

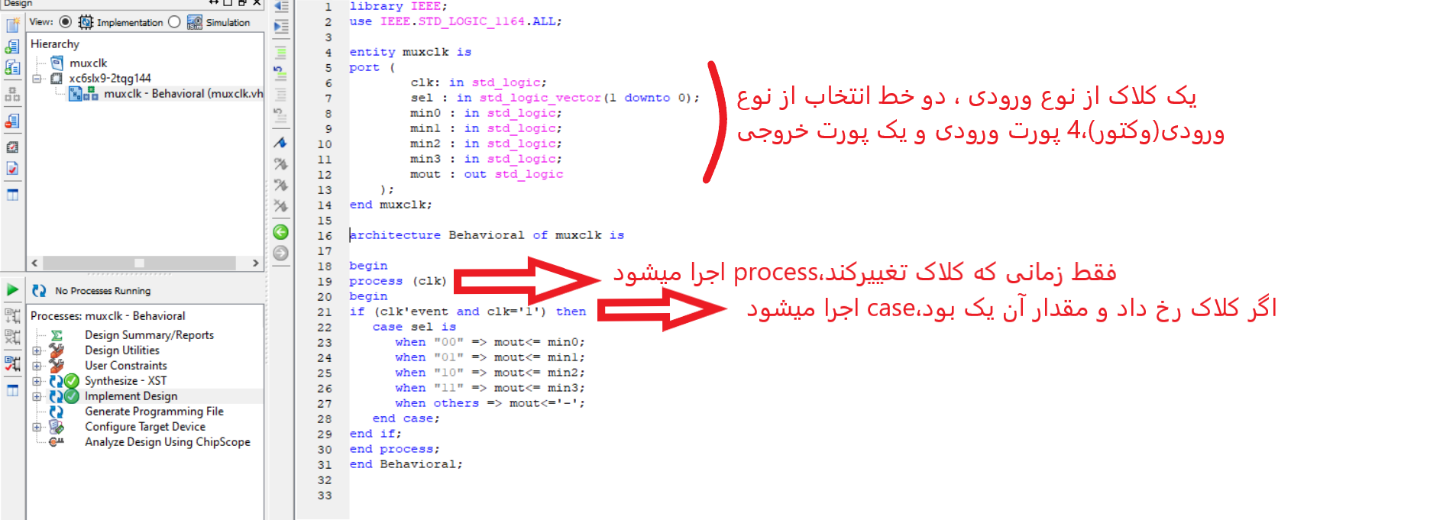
****

**تصویر 9 – شماتیک مالتی پلکسر 4 به 1 با کلاک**

|  |  |  |  |
| --- | --- | --- | --- |
| **mout** | **sel1** | **sel0** | **clk** |
| min0 | **0** | **0** | **1** |
| min0 | **1** | **0** | **1** |
| min0 | **0** | **1** | **0** |
| Min3 | **1** | **1** | **1** |
| min3 | **1** | **1** | **0** |

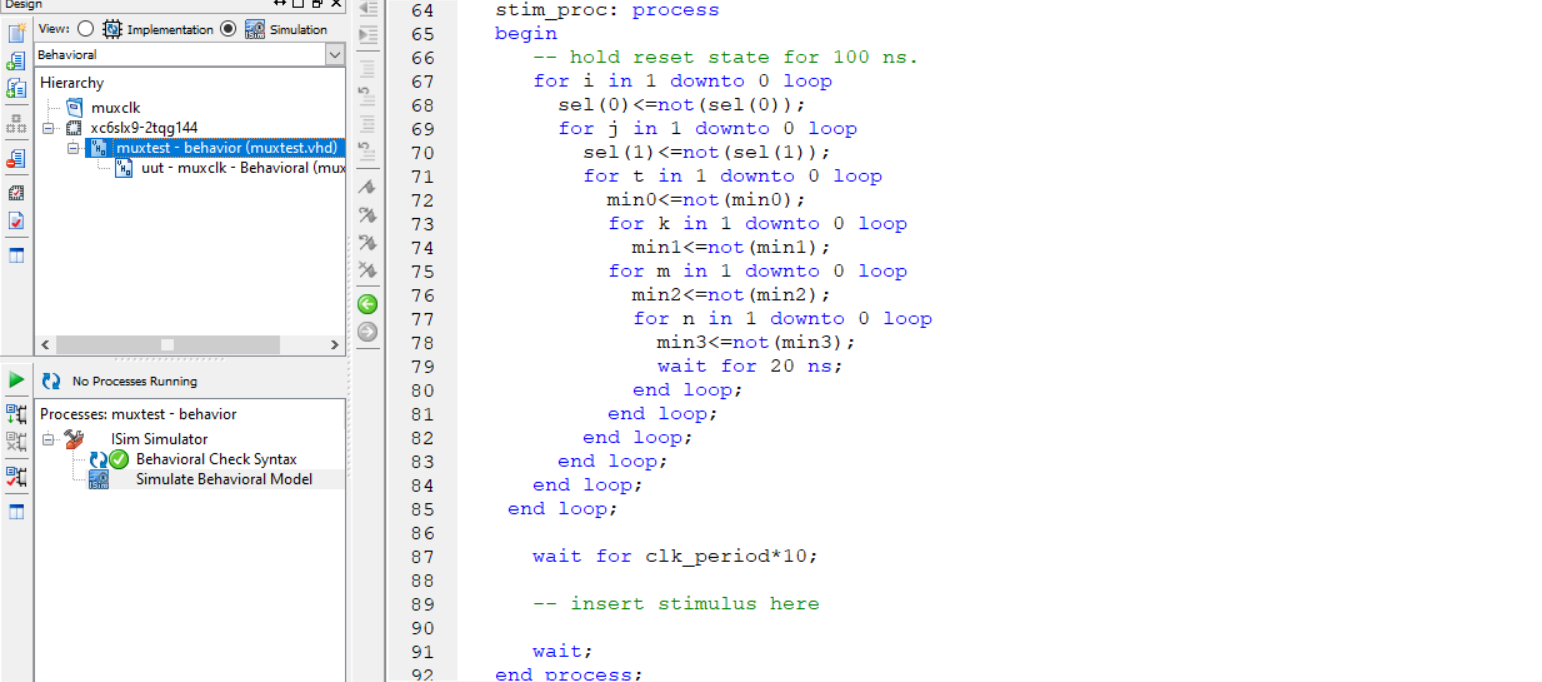
**جدول 3 – جدول درستی مالتی پلکسر 4 به 1 با کلاک**

پیاده سازی کد مالتی پلکسر 4 به 1 با کلاک مانند بی کلاک است با این تفاوت که یک پورت ورودی کلاک به کد اضافه می کنیم و کلاک را در بخش حساسیت process میگذاریم چون در این مدار تنها با تغییر کلاک خروجی تغییر می کند.

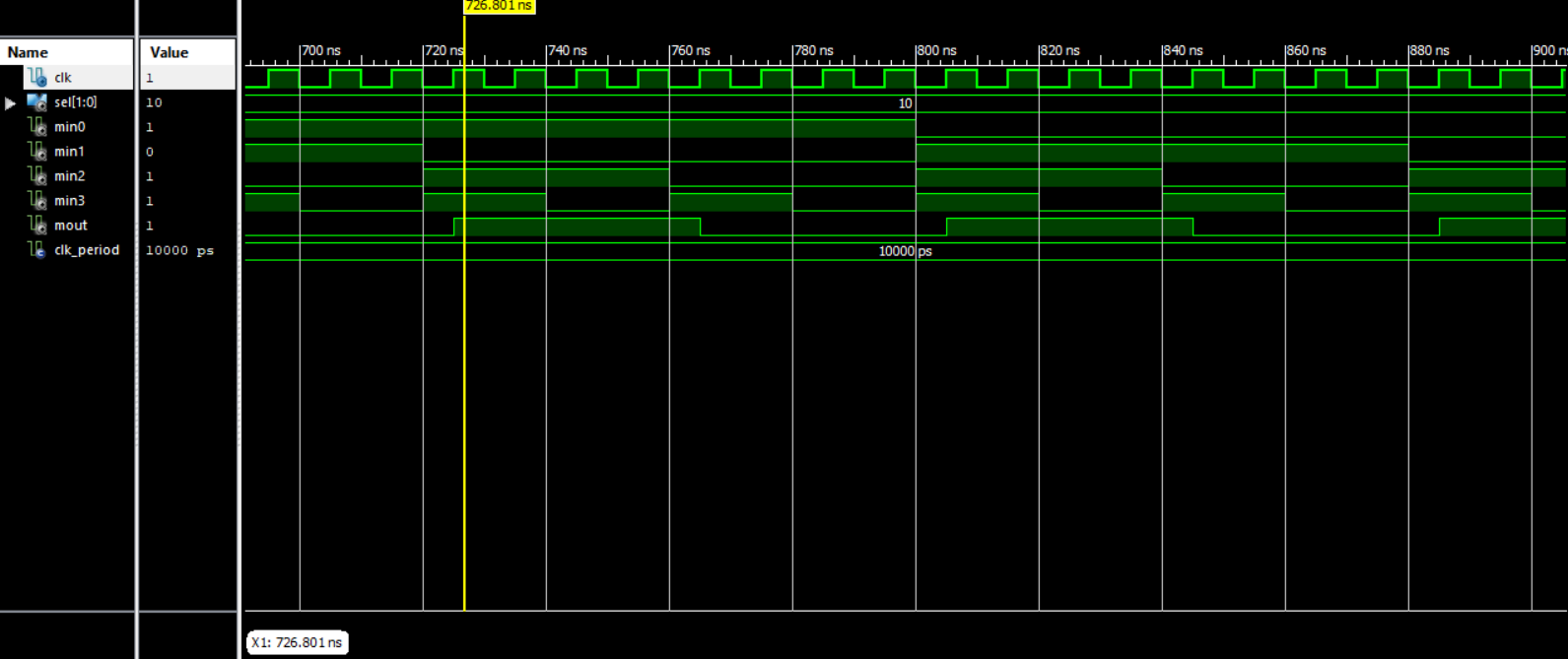
****

**تصویر 10 – کد VHDL مالتی پلکسر 4 به 1 با کلاک**

**تست بنچ مالتی پلکسر 4 به 1 با کلاک:** پیاده سازی این قسمت هم به مانند بدون کلاک است با این تفاوت که بعد از ساخت فایل تست بنچ از ماژول با کلاک،قسمت های مربوط به کلاک را در این فایل پاک نمی کنیم.

****

**تصویر 11 – تست بنچ مالتی پلکسر 4 به 1 با کلاک**

****

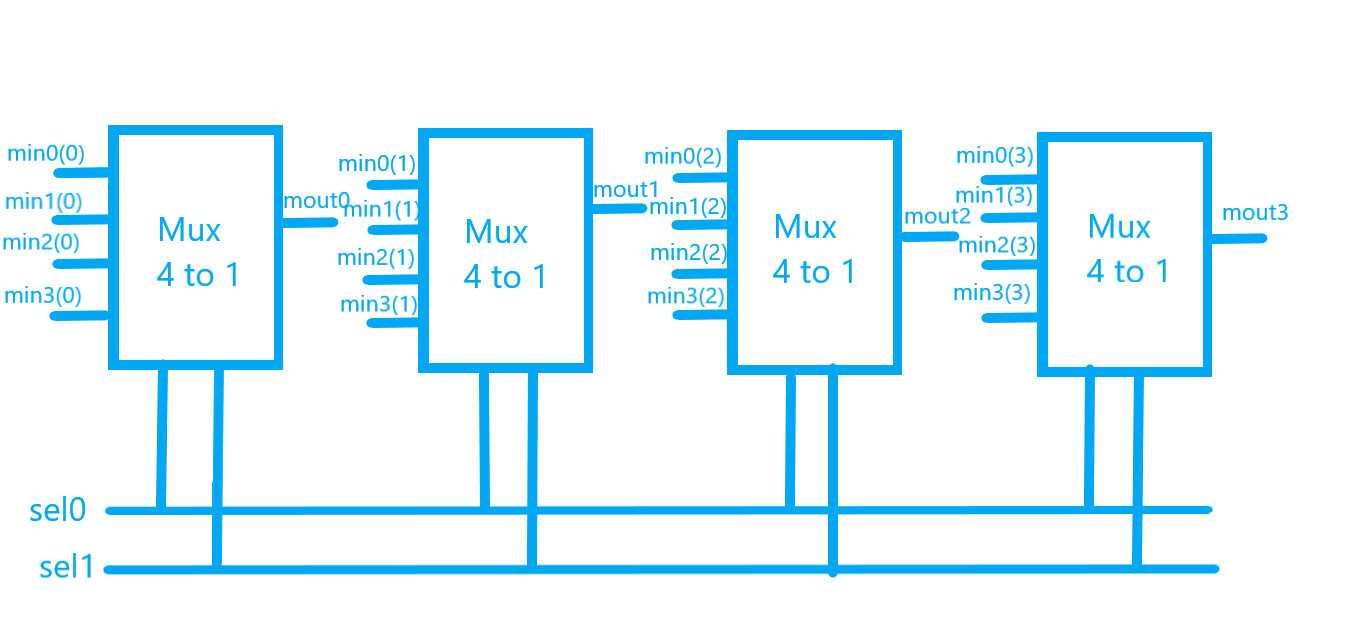
**تصویر 12 - شبیه سازی مالتی پلکسر 4 به 1 با کلاک با استفاده از کد تست بنچ و اجرای تست کیس های متفاوت**

**پاسخ به سوالات:**

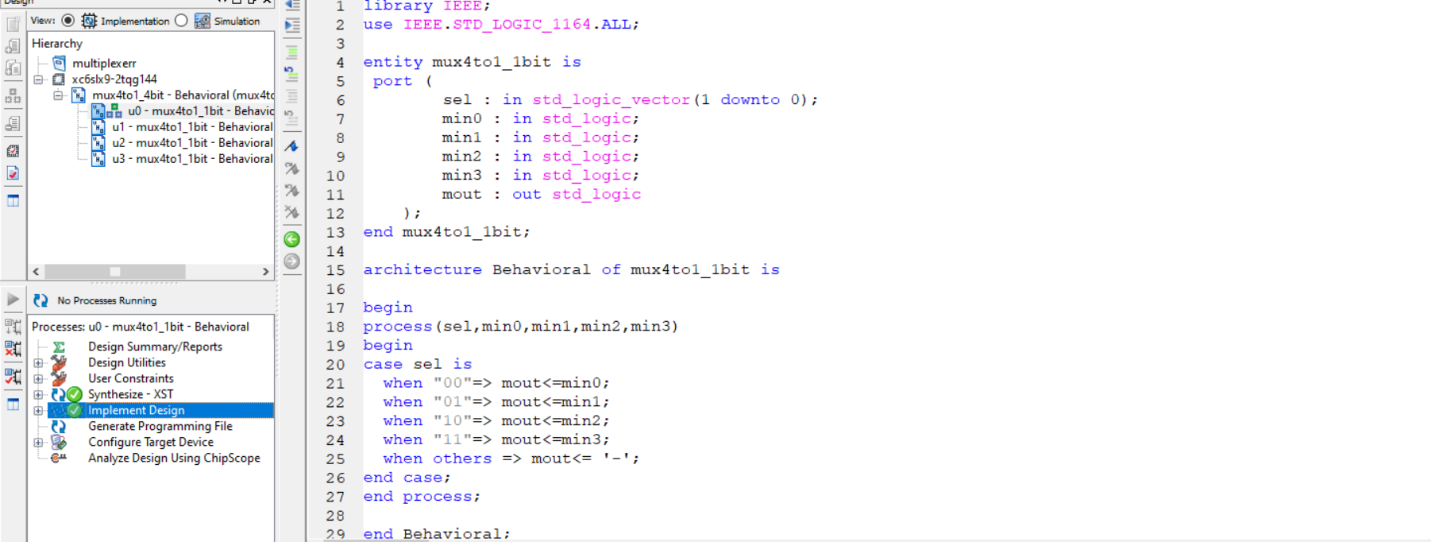
**1)پیاده سازی مالتی پلکسر 4 به 1 چهار بیتی با استفاده از ماژول آزمایش دوم(بدون کلاک):**

در این مدار باید 4 عدد چهاربیتی وارد کرده و با توجه به مقادیر خط انتخاب یکی از این اعداد 4 بیت به خروجی منتقل می شوند.

پیاده سازی این مدار با استفاده از 4 مالتی پلکسر 4 به 1 است که خطوط انتخاب هر 4 مالتی پلکسر با همدیگر مشترک است به گونه ای که بیت اول عدد چهار بیتی اول(min0(0)) در مالتی پلکسر اول و بیت دوم آن (min0(1))در مالتی پلکسر دوم،بیت سوم (min0(2)) در مالتی پلکسر سوم و بیت چهارم(min0(3)) در مالتی پلکسر چهارم و اعداد 4 بیتی دیگر نیز به همین صورت قرار میگیرند و با توجه مقادیر خط انتخاب، بیت های یکی از اعداد 4 بیت به خروجی مالتی پلکسر نظیرشان منتقل می شوند.(تصویر 13)

****

**تصویر 13 – شماتیک مالتی پلکسر 4 به 1 چهار بیتی**

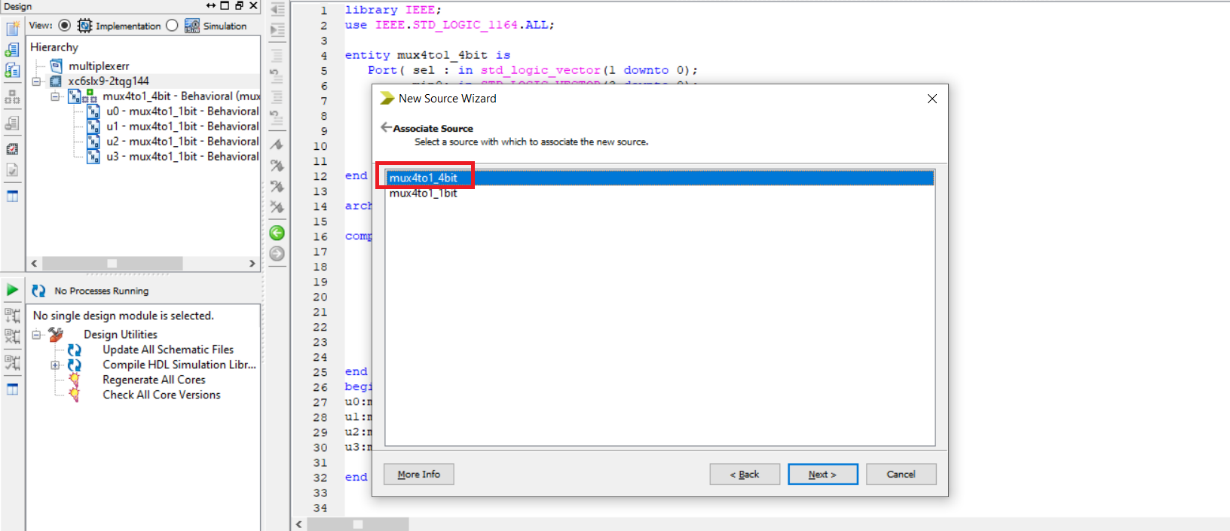
****

**تصویر 14 – کد VHDL مالتی پلکسر 4 به 1 تک بیتی**

****

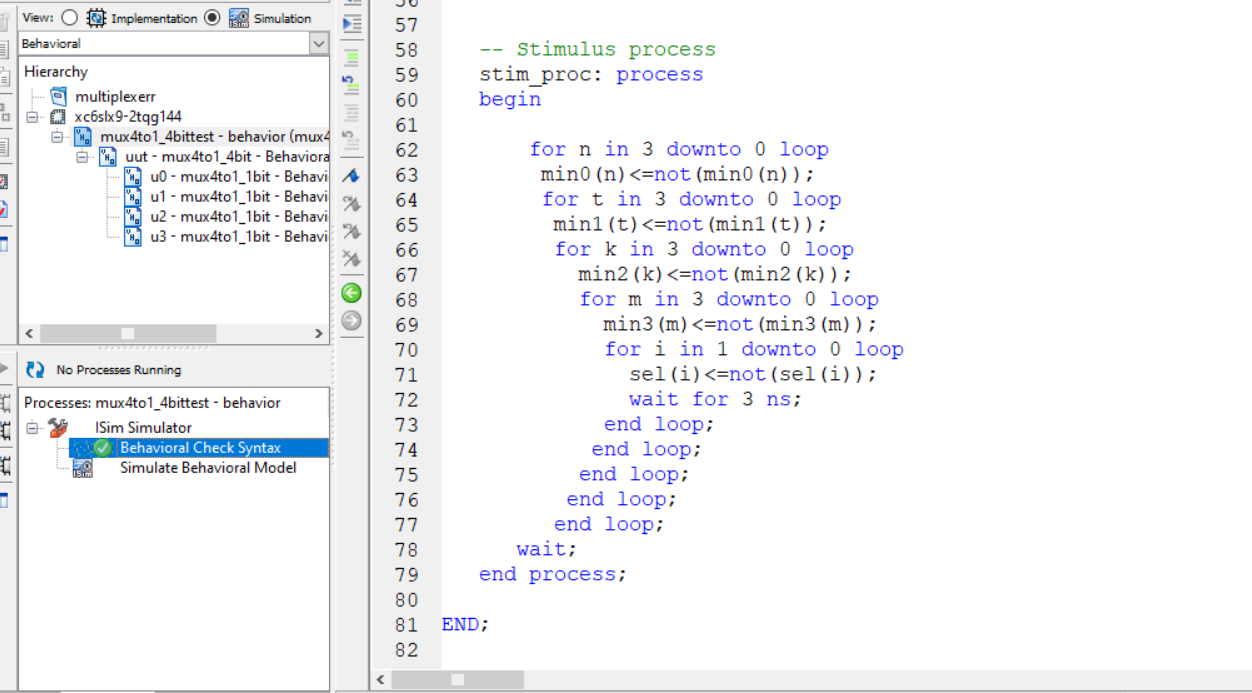
**تصویر 15 – کد VHDL مالتی پلکسر 4 به 1 چهار بیتی**

**تست بنچ مالتی پلکسر 4 به 1 چهار بیتی:** هنگام ساخت فایل تست بنچ،باید از ماژول 4 بیت تست بنچ را بسازیم(تصویر 16)

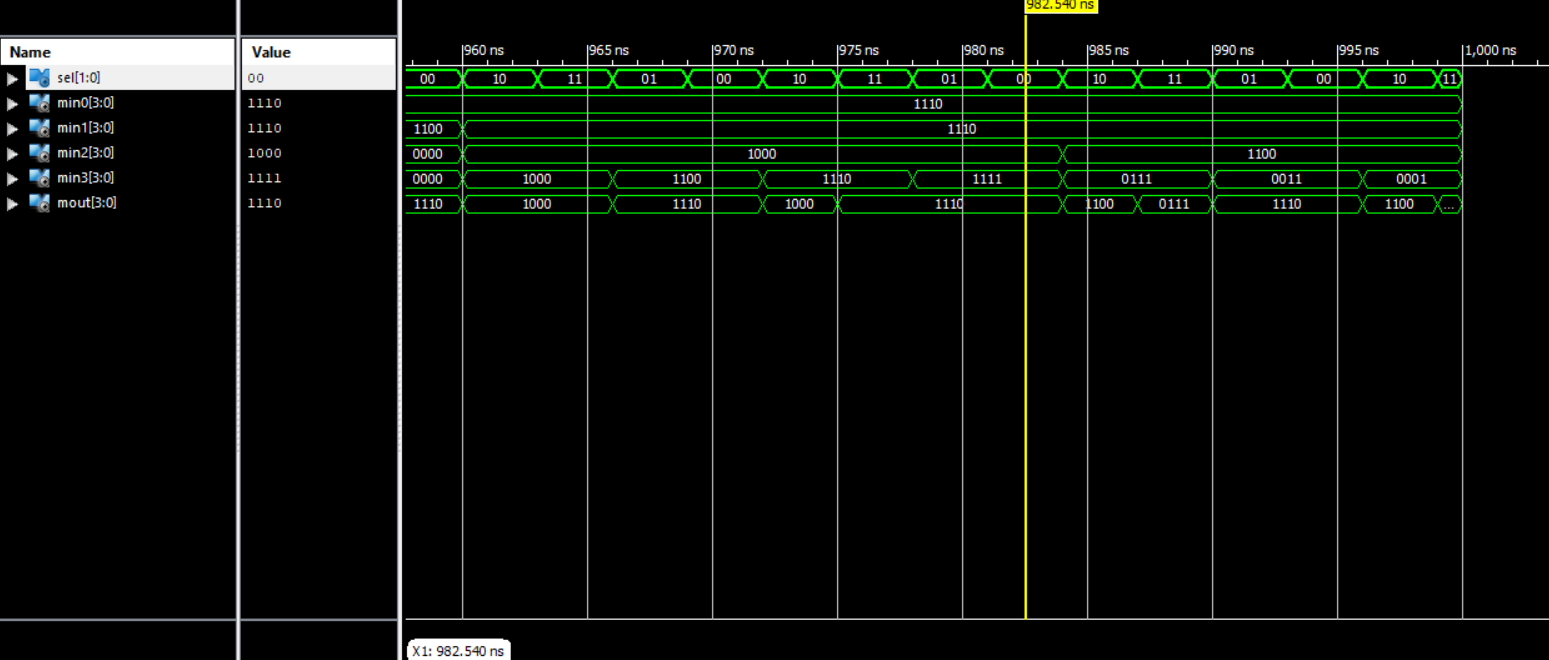
****

**تصویر 16**

برای تست بنچ ملاتی پلکسر 4 به 1 چهار بیت مانند تست بنچ های قبل، با استفاده از حلقه های تو در تو و نقیض کردن خطوط انتخاب و ورودی ها،حالت های ممکن را بررسی و تست می کنیم. از آنجایی که محدودیت تست بنچ 1000 نانوثانیه است و در این ماژول 4 بیت نمیتوان همه تست کیس ها را بررسی کرد حلقه نقیض کردن sel را در آخر میگذاریم تا بتوانیم بیشترین حالت ها را بررسی کنیم.

****

**تصویر 17 – تست بنچ مالتی پلکسر 4 به 1 چهار بیتی**

****

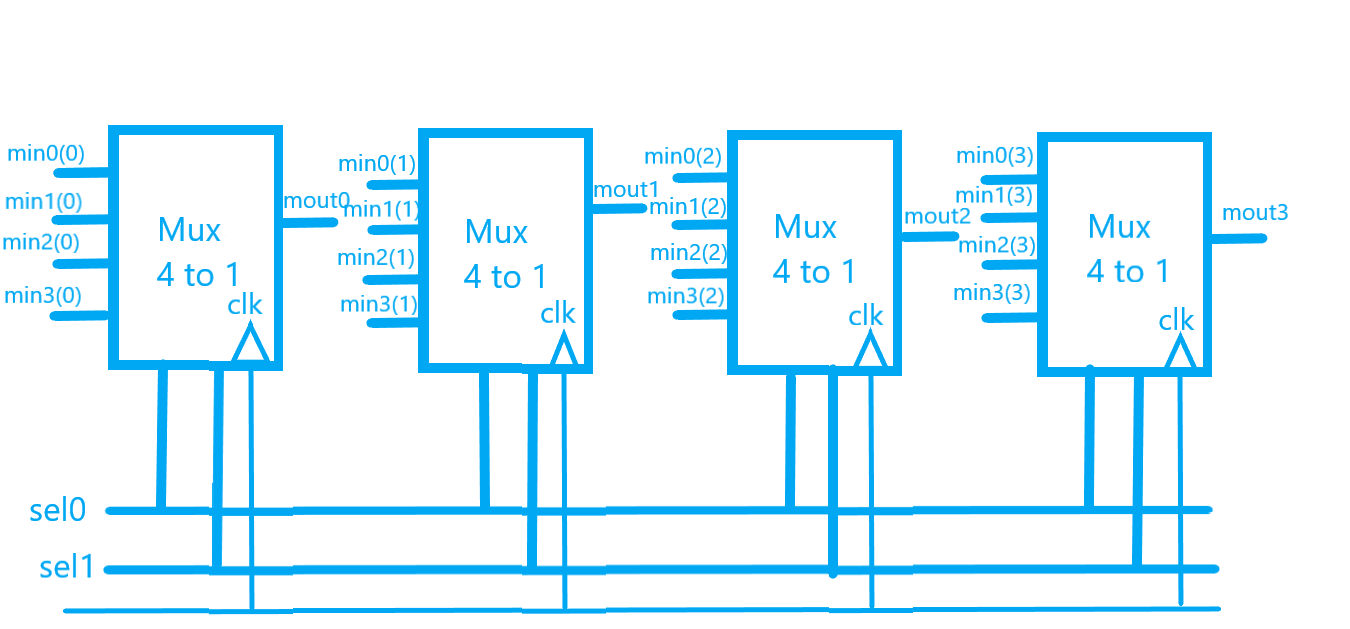
**تصویر 19 - شبیه سازی مالتی پلکسر 4 به 1 چهار بیتی با استفاده از کد تست بنچ و اجرای تست کیس های متفاوت**

**2)تفاوت ماژول ساخته شده در آزمایش 2 و 3:**

در ماژول آزمایش دوم ، از کلاک استفاده نشده و خروجی مدار به تغییرات خطوط انتخاب و ورودی ها حساس است و در ماژول آزمایش سوم از کلاک استفاده شده و خروجی مدار تنها به تغییرات کلاک حساس است و تنها در صورت تغییر کلاک خروجی مالتی پلکسر ما تغییر میکند که به این مدار ها که خروجی آن ها وابسته به تغییرات کلاک هستند سنکرون یا همزمان می گویند بنابراین ماژول سوم آزمایش بر خلاف ماژول آزمایش دوم سنکرون است.

**3)پیاده سازی مالتی پلکسر 4 به 1 چهار بیتی با استفاده از ماژول آزمایش سوم(با کلاک):**

پیاده سازی این ماژول مانند نوع بی کلاک آن(سوال قبل) است با این تفاوت که هر 4 مالتی پلکسر دارای خط کلاک مشترک هستند.(تصویر 20)

****

**تصویر 20 – شماتیک مالتی پلکسر 4 به 1 چهاربیتی با کلاک**

برای پیاده سازی کد مالتی پلکسر 4 به 1 با کلاک مانند بی کلاک است و از ساختار case استفاده می شود با این تفاوت که یک پورت کلاک تعریف میکنیم و پروسه خود را حساس به تغییرات کلاک میکنیم که با تغییرات کلاک خروجی ما نیز تغییر کند.

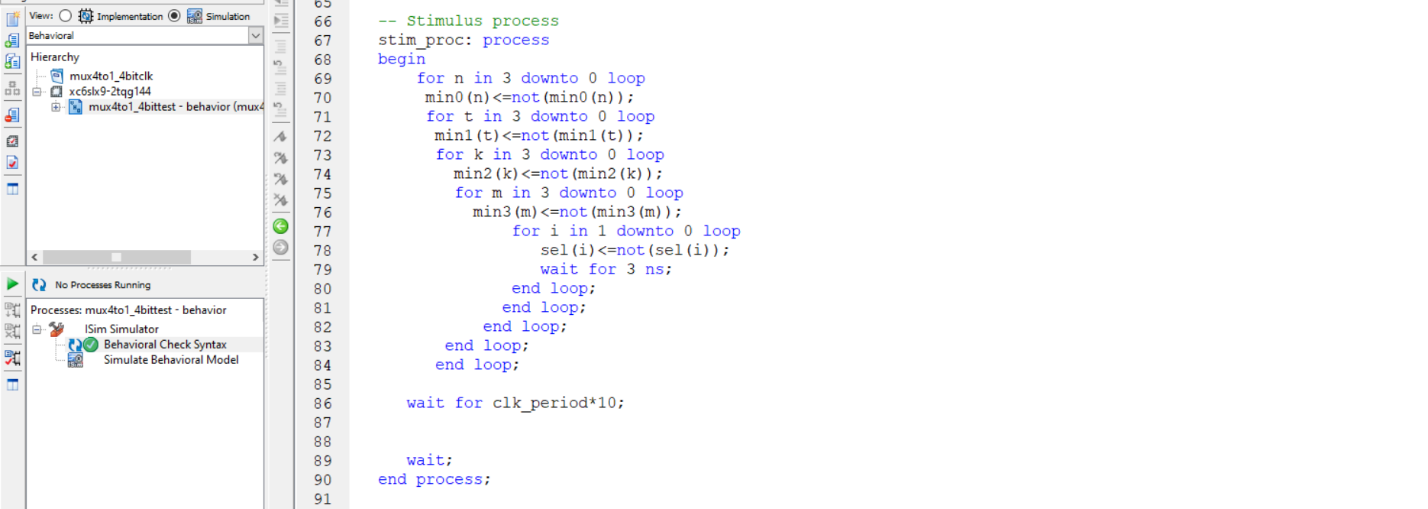
****

**تصویر 21 – کد VHDL مالتی پلکسر 4 به 1 تک بیتی**

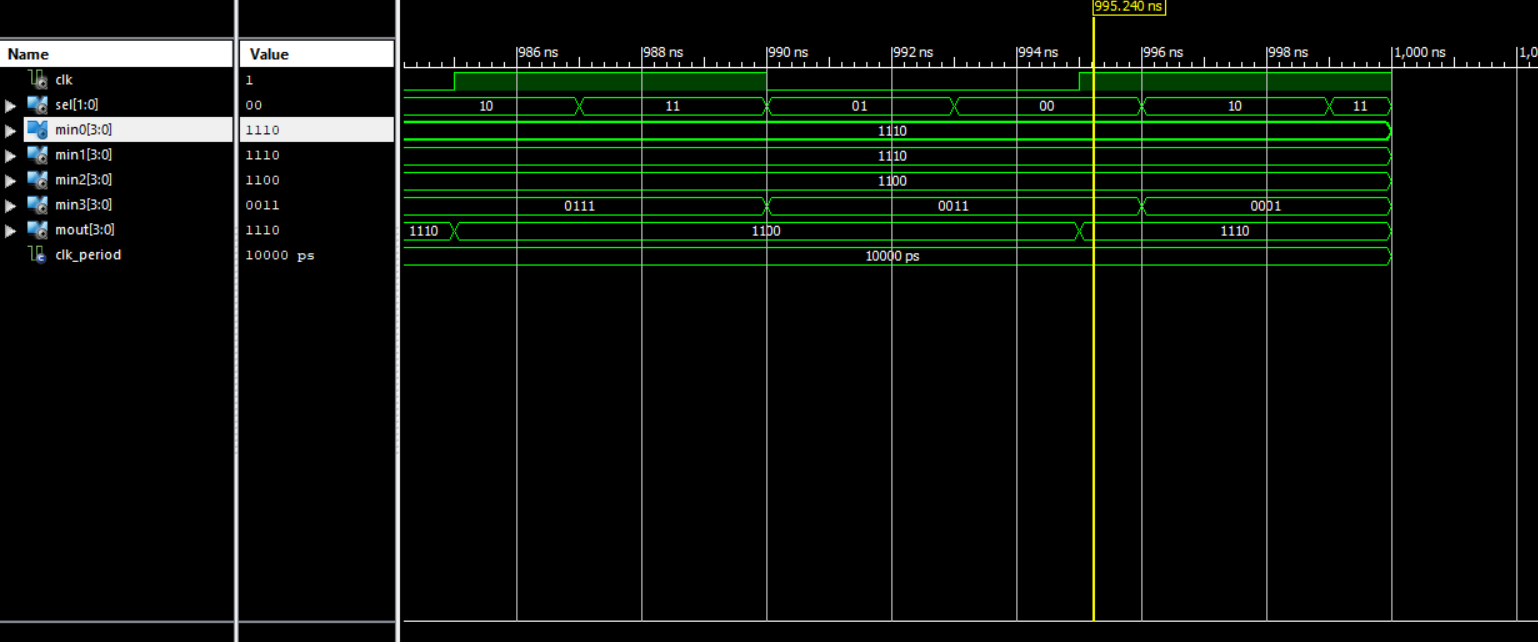
****

**تصویر 22 – کد VHDL مالتی پلکسر 4 به 1 چهار بیتی**

برای تست بنچ مالتی پلکسر 4 به 1 چهار بیتی با کلاک مانند بدون کلاک عمل می کنیم با این تفاوت که بعد از ساخته شدن فایل تست بنچ بخش کلاک را حذف نمی کنیم. مانند ماژول بدون کلاک با نقیض کردن خطوط سلکت و ورودی ها،تست کیس های مختلف را ایجاد و حالت های مختلف را بررسی می کنیم.

****

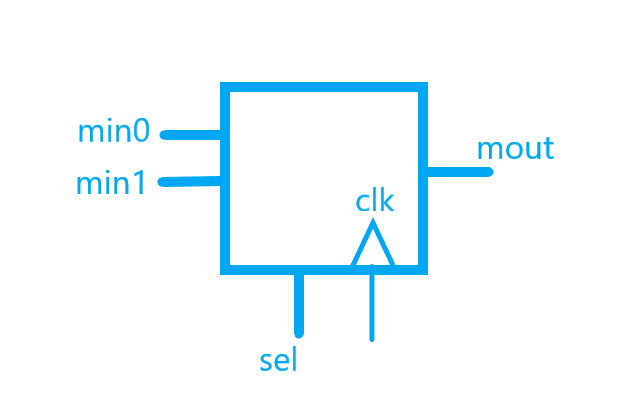
**تصویر 23 - تست بنچ مالتی پلکسر 4 به 1 چهار بیتی**

****

**تصویر 24 - شبیه سازی مالتی پلکسر 4 به 1 چهار بیتی با کلاک با استفاده از کد تست بنچ و اجرای تست کیس های متفاوت**

**4)پیاده سازی مالتی پلکسر 2 به 1 با کلاک:**

مالتی پلکسر 2 به 1 با کلاک شامل دو پورت ورودی و یک خط کلاک و یک خط انتخاب و یک پورت خروجی است.

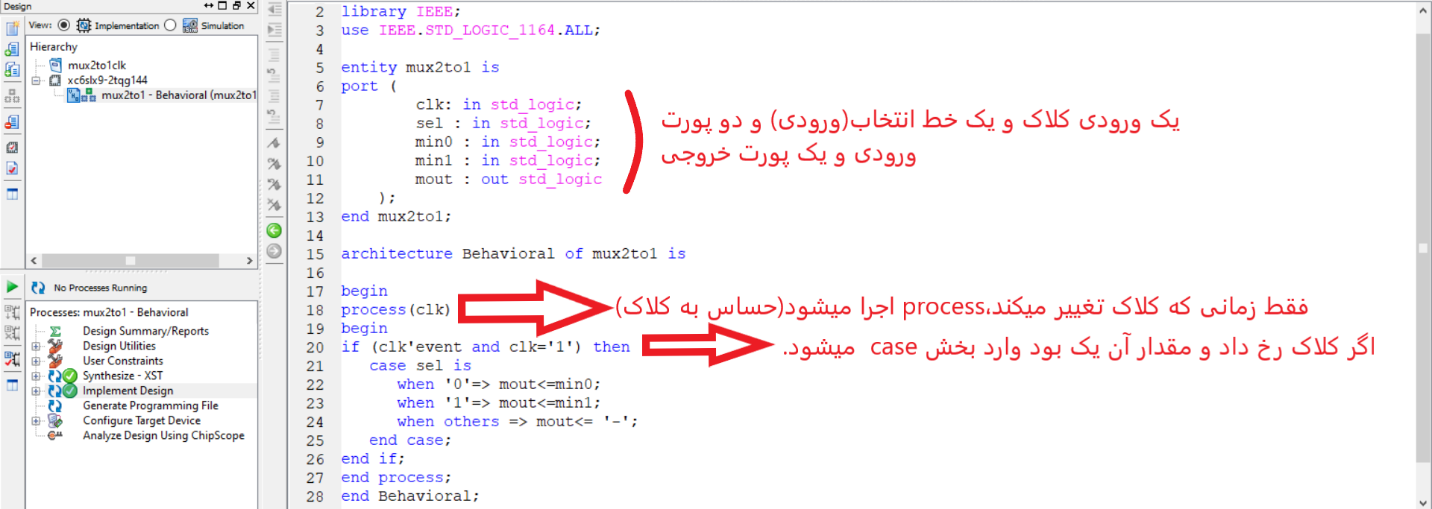
****

**تصویر 25 – شماتیک مالتی پلکسر 2 به 1 با کلاک**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **mout** | **min1** | **min0** | **sel** | **clk** |
| - | **0** | **0** | **0** | **0** |
| 0 | **1** | **0** | **0** | **1** |
| 0 | **0** | **1** | **1** | **1** |
| 0 | **1** | **1** | **1** | **0** |
| 1 | **1** | **1** | **1** | **1** |

**جدول 4 – جدول درستی مالتی پلکسر 2 به 1**

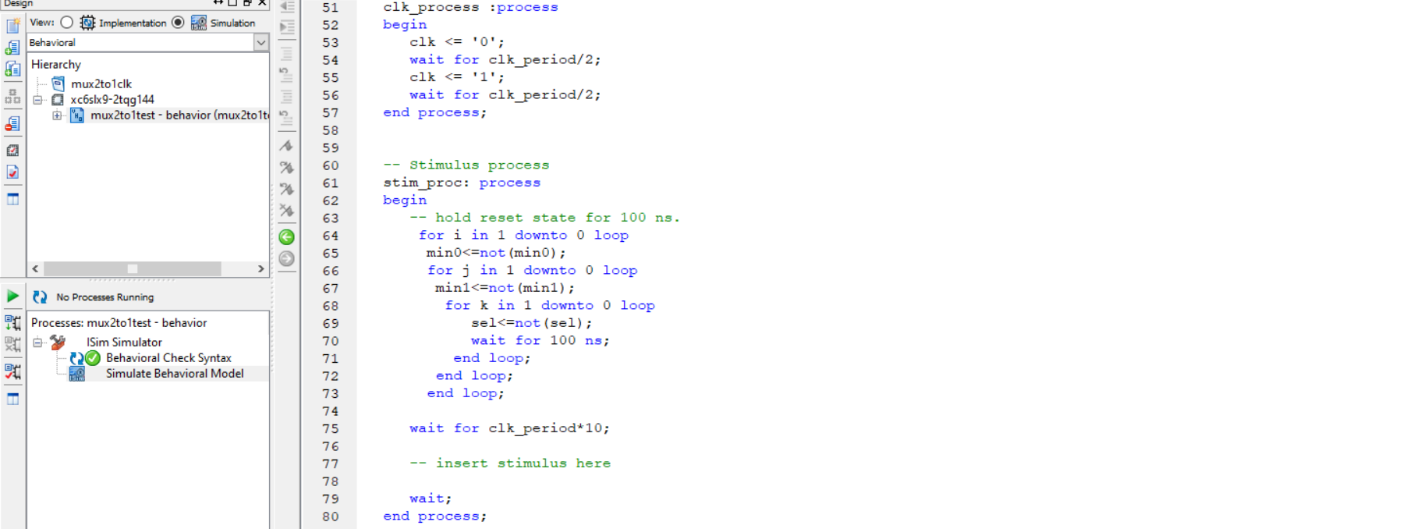
پیاده سازی کد مالتی پلکسر 2 به 1 تک بیتی مانند 4 به 1 است با این تفاوت که در 2 به 1، دو پورت ورودی و یک خط انتخاب داریم و در بخش case،فقط دو شرط را بررسی میکنیم.زمانی که خط انتخاب برابر"0" باشد،خروجی برابر min0 است و زمانی که خط انتخاب برابر"1" باشد،خروجی برابر min1 است

****

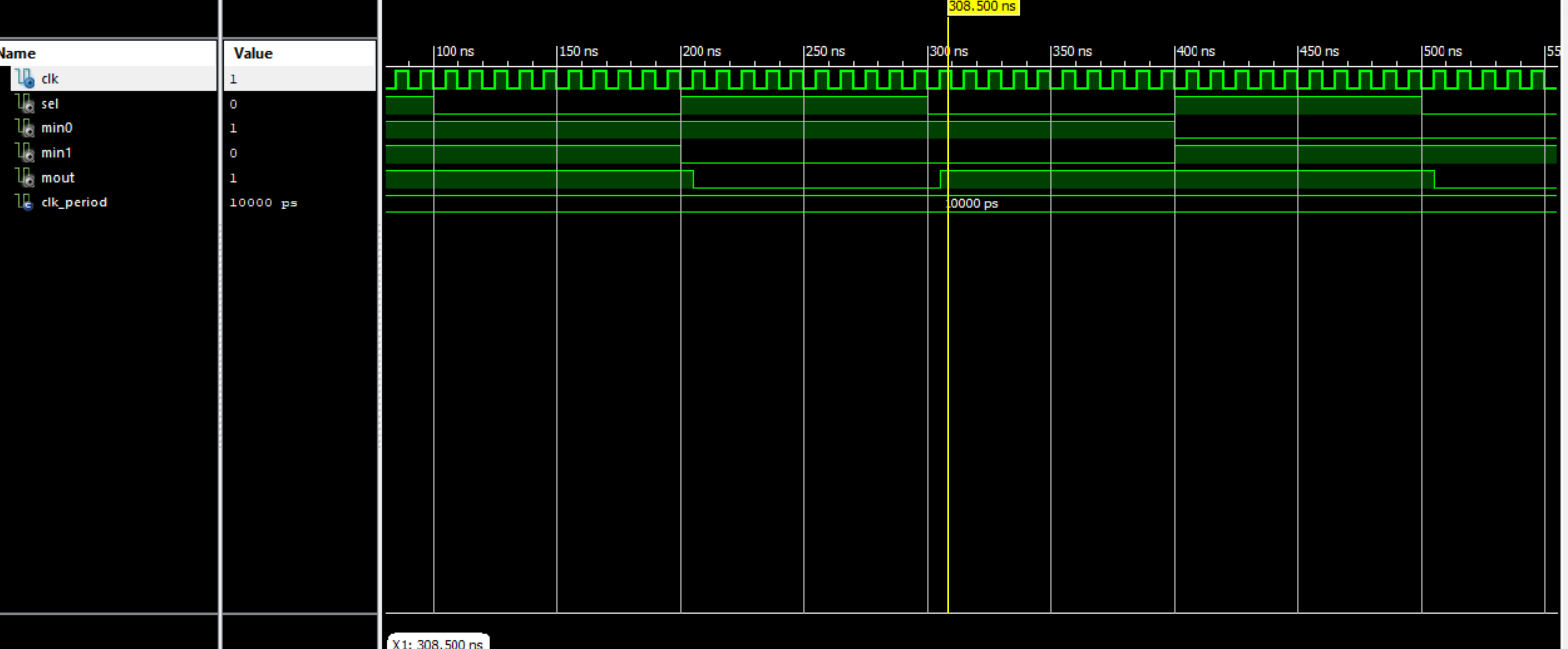
**تصویر 26 – کد VHDL مالتی پلکسر 2 به 1 تک بیتی**

**تست بنچ مالتی پلکسر 2 به 1 تک بیتی:**

مانند ماژول های قبل با حلقه های تو در تو و نقیض کردن خط انتخاب و ورودی ها،حالت های مختلف را بررسی میکنیم.به دلیل زیاد نبودن تعداد حالت ها زمان انتظار بین هر تست کیس را 100 نانو ثانیه میزاریم.



**تصویر 27 – تست بنچ مالتی پلکسر 2 به 1 تک بیتی**



**تصویر 28 - شبیه سازی مالتی پلکسر 4 به 1 چهار بیتی با کلاک با استفاده از کد تست بنچ و اجرای تست کیس های متفاوت**