

## Pràctica 4. Introducció al disseny en VHDL con Xilinx ISE.

### 1. Introducció.

Esta pràctica serve de introducció a la descripció de sistemes mitjançant el llenguatge VHDL. Per a ell, emplearem el simulador de VHDL Modelsim que acompanya les eines de disseny de Xilinx. Este simulador admet el mateix conjunt de VHDL capaç de sintetitzar-se en els dispositius programables Xilinx donat que incorpora llibrerías específiques per a aquests dispositius. Però a pesar de ell, és un simulador de ús molt estès donat que a través de les llibrerías és possible un ús més general.

A primera vista, Modelsim resulta un programa no massa intuïtiu, però si se segueixen els passos apropiats, es pot realitzar la simulació de forma senzilla. Esta complexitat ve donada per la possibilitat de controlar el simulador mitjançant comandos, i la seua capacitat per a poder simular tot tipus de característiques VHDL, inclosos aquelles no sintetitzables; ademés de admetre descripcions en Verilog.

**Si se realitzen modificacions en el còdigo, és necessari compilar de nou el fitxer** per a que els canvis tinguin efecte. Un projecte crea un directori on s'introdueixen tots els fitxers relacionats, i un o diversos subdirectoris amb les llibrerías creades (per defecte sempre es crea la llibrería **work**).

### 2. Objectiu de la pràctica.

En esta pràctica es realitza una aproximació al maneig del simulador mitjançant la realització de diversos elements funcionals que es descriuen a continuació.

### 3. Primer apartat.

Simular un comptador de 8 bits. Com a entrades, té una de rellotge i una de reset, la sortida és el valor del comptador. Realitzar tots els passos necessaris per a crear un projecte nou, introduir el còdigo VHDL, compilar, afegir al projecte i simular. Utilitzar el dispositiu Spartan2 XCS2200E-6pq208. Per a simular, fer-ho de les següents formes:

1. Introduint vectors de test des de Modelsim (este procediment ja és conegut).
2. Creant els vectors de test amb HDL Bench (este procediment ja és conegut).
3. Creant un fitxer de banc de proves. Per a este mètode, es crea un nou fitxer font (**New Source**), del tipus Test Bench associat a l'entitat que volem testar. Una vegada programat el fitxer, tenint-lo seleccionat, en la finestra de processos es executa la simulació. Amb este fitxer creat, executar les diferents opcions de simulació (**behavioral, post-translate, post-map, post place&route**) analitzant i justificant les possibles diferències que poden existir en la simulació.

### 4. Segon apartat.

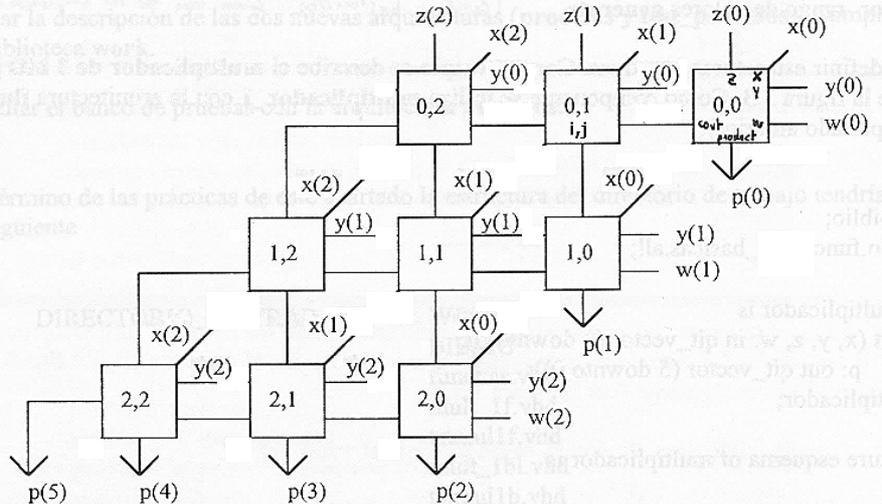
Crear un nou projecte on s'inclueixen els fitxers "**MULT1.VHD**" i "**TSTMUL1F.VHD**" que es baixen des de la pàgina Web de l'assignatura. Analitzar les descripcions que se donen en cada un dels fitxers, com es veurà, existeix una única entitat i dues arquitectures en el mateix fitxer, amb lo que tractarem de veure com es pot anomenar a una u altra a l'hora de simular i implementar. A l'hora de compilar, assegurar-se que en la descripció del component que referencia a l'entitat **multiplicador\_1**, s'utilitza l'arquitectura **flujo**. Si tot ha compilat correctament, es pot començar la simulació, però esta vegada no és necessari definir el comportament de les senyals perquè este ve definit en el fitxer de simulació (banc de proves o *testbench*) **TSTMUL1F.VHD**.

Comprovar que els resultats de la simulació corresponen amb els de la descripció de funcionament proporcionada.

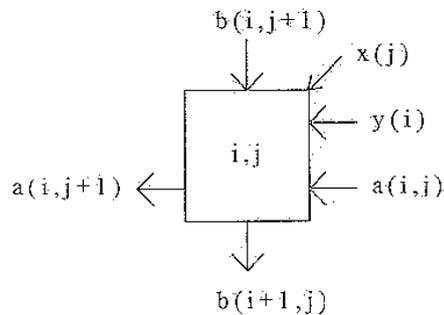
Una vegada comprovat el correcte funcionament (en cas contrari, revisar el disseny fins que funcione correctament), editar el fitxer "**TSTMUL1F.VHD**" per a emprar l'arquitectura **procesos** en lloc de **flujo**. Compilar el disseny (realment basta amb simular el fitxer de test que és el únic que s'ha modificat) i simular de nou tot el sistema per a comprovar si coincideix amb el resultat de la simulació anterior, cerciorant-se que ara l'arquitectura que es està simulant és l'arquitectura **procesos**, per a ell, revisar els missatges que es emeten en la finestra principal de ModelSim.

### 5. Tercer apartado.

En este apartado deseamos crear un multiplicador de 3x3 bits basándonos en la descripción del módulo de 1 bit del apartado anterior. La estructura que se debe seguir es la siguiente:



Para facilitar la tarea de descripción, resulta recomendable realizar la declaración basándose en sentencias **generate** que realicen de forma general el interconexionado. Con esto, expandir el multiplicador a cualquier número de bits resultaría inmediato. Como guía de descripción, bastaría con declarar unas señales **a** y **b** de interconexión de tipo array de tal modo que la interconexión de un módulo con otro sería del siguiente modo:



Realizar la descripción VHDL de este sistema, y crear su banco de pruebas asociado. Recordad que el resultado para números sin signo debe ser  $(x*y)+z+w$ , siendo  $x,y,z,w$  números de 3 bits.

### 6. Documentación a entregar.

Se entregará un pequeño dossier que incluya el código VHDL con los diseños propuestos y los resultados de la simulación, así como la opinión personal acerca de la práctica realizada y dificultades encontradas en el desarrollo de la misma.