

## Procedimiento para simular proyectos en Verilog con muchos archivos de diseño usando el programa Questa Altera Starter FPGA Edition

En este documento se expone el procedimiento a seguir para obtener una simulación funcional de un sistema digital que ha sido especificado con el lenguaje de descripción de hardware Verilog donde se encuentran presentes el tope de jerarquía y múltiples archivos de diseño. La herramienta de simulación es Questa Altera Starter FPGA Edition para Quartus Prime Lite Edition 25.1 de Altera (ahora Intel). El ejemplo que se hace a continuación es con solo archivos de Verilog, pero el procedimiento es el mismo si todo el diseño ha sido elaborado en VHDL. Los pasos que se indican es para una máquina Linux con Ubuntu 22.04 LTS, donde previamente se ha tenido que preparar el archivo **.bashrc** con las siguientes líneas:

```
export QROOT="/home/uniccass/altera_lite/25.1std"
export QUARTUS_ROOTDIR="$QROOT/quartus"
export PATH="$PATH:$QUARTUS_ROOTDIR/bin:$QUARTUS_ROOTDIR/linux64:$QROOT/questa_fse/bin"
export SALT_LICENSE_SERVER=/home/uniccass/.altera.quartus/questa_lic.data
export MGLS_LICENSE_SERVER=/home/uniccass/.altera.quartus/questa_lic.data
export LM_LICENSE_SERVER=/home/uniccass/.altera.quartus/questa_lic.data
```

1. Colocarse en el directorio donde se encuentra el testbench y una copia de los archivos de diseño

```
$ cd /home/uniccass/unic-cass/2025/quartus/fp32adder/serial/verilog/testbenches/
```

2. Ejecutar el Questa desde la línea de comandos, y automáticamente se genera la GUI:

```
$ vsim
```

3. Crear un proyecto. Seleccionar: **File → New → Project**

4. Especificar lo siguiente:

**Project Name:** prueba (escoger un nombre de acuerdo al diseño a simular)

**Project Location:** /home/uniccass/unic-cass/2025/quartus/fp32adder/serial/verilog/testbenches/  
(ejemplo de ubicación de archivos \*.v)

**Default Library Name:** work

**Copy Settings From:** /home/uniccass/altera\_lite/25.1std/questa\_fse/modelsim.ini

Seleccionar **"Copy Library Mappings"**

Presionar el botón **OK**

**Nota:** lo indicado para "Copy Settings From" corresponde a la ruta por defecto de la instalación estándar de ModelSim para Quartus II de Altera. No modificar la ruta que aparezca en su caso en particular.

5. En la pequeña ventana que aparece (**"Add items to the Project"**) dar un click al ícono **"Add Existing File"**
6. En la pequeña ventana que aparece (**"Add file to Project"**) hacer una selección múltiple de todos los archivos \*.v que forman parte del proyecto de Quartus Prime incluyendo el archivo de testbench apretando el botón **"Browse"**. Mantener las opciones por defecto de esta ventana:

**Add file as type:** default

**Folder:** Top Level

**Reference from current location**

y apretar el botón **OK**

7. Cerrar la pequeña ventana **"Add items to the Project"** apretando el botón **"Close"**
8. En el tab **"Project"** aparecen ahora todos los archivos seleccionados en el paso 6. Proceder a compilar todos los archivos: Seleccionar: **Compile → Compile All**

Si hubiera algún error de compilación en algún archivo en particular, seleccionarlo individualmente y compilarlo: **Compile → Compile selected**

La compilación de todos los archivos debe ser exitosa para proceder al siguiente paso. De presentarse "warnings", puede proceder con el siguiente paso. De presentarse un error, en la ventana inferior darle un doble click al mensaje de error y aparecerá una nueva ventana explicativa del error y en la línea del archivo donde posiblemente se encuentre el error.

9. Preparación para una simulación funcional del proyecto. Seleccionar:

**Simulate → Start Simulation**

En la ventana que aparece (**"Start Simulation"**), aparece una serie de tabs:  
Design, VHDL, Verilog, Libraries, SDF, Others

En el tab **"Design"**, expandir la librería **"work"** o el nombre que se colocó en el paso 4 (cajón **"Default Library Name"**) y seleccionar el archivo de testbench (no es necesario expandir el archivo de testbench), con lo cual el cajón **"Design Unit(s)"** se completa con la palabra **"work.tb"**, asumiendo que el nombre del archivo de testbench es **"tb.v"**. Y en la sección **"Optimization"**, dar un check a **"Enable optimization"** y apretar el botón **"Optimization Options"**. En la ventana que se abre en el tab **"Visibility"**, seleccionar la opción **"Apply full visibility to all modules (full debug mode)"**, desde que estamos tomando todos los archivos fuente y no estamos tomando el diseño sintetizado, el cual optimiza eliminando nets. Apretar el botón **OK**, y al regresar a la ventana anterior presionar el botón **OK**.

- Después de unos cuantos segundos, el entorno de Questa cambia, generándose una serie de ventanas/tabs. Se crea el tab **"sim"** donde se identifican las arquitecturas a simular, y las ventanas **"Objects"** y **"Processes"**.
- En el tab **"sim"** seleccionar **"tb"** (asumiendo el nombre del testbench es **"tb.v"**), asegurándose que el color sea azul oscuro (no azul tenue) y proceder a lo siguiente:

**Add → To Wave → All items in region**

con lo que se genera una nueva ventana llamada **"Wave"**, la cual mostrará los resultados de la simulación aplicando como archivo de entrada el archivo de testbench. La ventana **"Wave"** se completa con las señales que se van a simular. Si el diseño instancia el estilo de diseño tipo FSM+Datapath, seleccionar la instanciación de FSM y seleccionar al igual que antes **Add → To Wave → All items in region**, y del mismo modo con la instanciación del Datapath, para poder visualizar todas las señales de interés.

- Cambiar la raíz (Radix) de la representación de todas las señales. Seleccionar lo siguiente:

**Simulate → Runtime Options**

En la ventana que aparece (Runtime Options), y en el tab **"Defaults"** seleccionar como **"Default Radix"** la opción Hexadecimal, apretar el botón **"Apply"** y luego el botón **OK**.

- Proceder a la simulación funcional usando el archivo de testbench. Seleccionar lo siguiente:

**Simulate → Run → Run –All**

En este momento sale una pequeña ventana emergente que dice **"Are you sure you want to finish?"**. Entonces, apretar el botón **No**, de otra forma, se cierran todas las ventanas. Entonces se generan todas las señales en función del tiempo en la ventana **"Wave"**. Si la simulación es exitosa, se obtendrá el mensaje **"SIMULATION FINISHED!!!"**.

- Proceder a salvar la simulación de la ventana **"Wave"**, con la opción **File → Save Format... Ctrl+S**, y darle un nombre, la extensión es siempre **\*.do**. Por defecto, el nombre del archivo es **wave.do**. Luego se puede cerrar la ventana con la opción **File → Close Window**.
- Para terminar la simulación, en el menú principal, seleccionar: **Simulate → End Simulation**  
En la ventana que aparece (**"End Current Simulation Session"**), confirmar la terminación de la simulación apretando el botón **"Sí"**
- Seleccionar el tab **"Project"** y ejecutar lo siguiente para cerrar el proyecto: **File → Close Project**  
En la ventana que aparece (**"Close Project"**), confirmar apretando el botón **"Sí"**
- Salir del Questa para Altera seleccionando: **File → Quit**  
En la ventana que aparece (**"Quit Vsim"**), confirmar apretando el botón **"Sí"**

- Al salir de Questa, se puede verificar que se han generado los archivos **prueba.cr.mti**, **prueba.mpf**, **transcript**, **vsim.wlf** y **wave.do**.

- Para ver una simulación, ejecutar el siguiente comando: `$ vsim –view vsim.wlf –do wave.do`

Pero si se tienen varios testbenches individuales en una misma carpeta, entonces, por cada simulación, salvar los archivos del paso 18 con un nombre diferente, excepto transcript, que es un histórico de los comandos ejecutados en vsim.

Lima, 29 de Enero de 2026

Aurelio Morales Villanueva.