**DISSENY DIGITAL BÀSIC 2016-2017**

***PRÀCTICA 0: INTRODUCCIÓ A LA SIMULACIÓ AMB QuestaSim***

L’objectiu d’aquesta pràctica introductòria és familiaritzar l’alumne amb el programa de simulació **QuestaSim** (també **ModelSim**) i veure les funcions bàsiques es poden realitzar amb ell. Per a això es simularà un primer component electrònic, un inversor (dispositiu que inverteix un senyal digital: si se li entra un “0” lògica donarà una sortida “1” lògic, i si se li entra un “1” lògic, la sortida serà un “0” lògic). Per a això, us recomanem que us mireu el començament del capítol d’Àlgebra de Boole del vostre llibre de text de referència.

1. CREACIÓ D’UN PROJECTE (Project)
2. Crear un directori propi. Això es farà al disc d:\

Per exemple, per l’alumne **13** del *Grup A*):

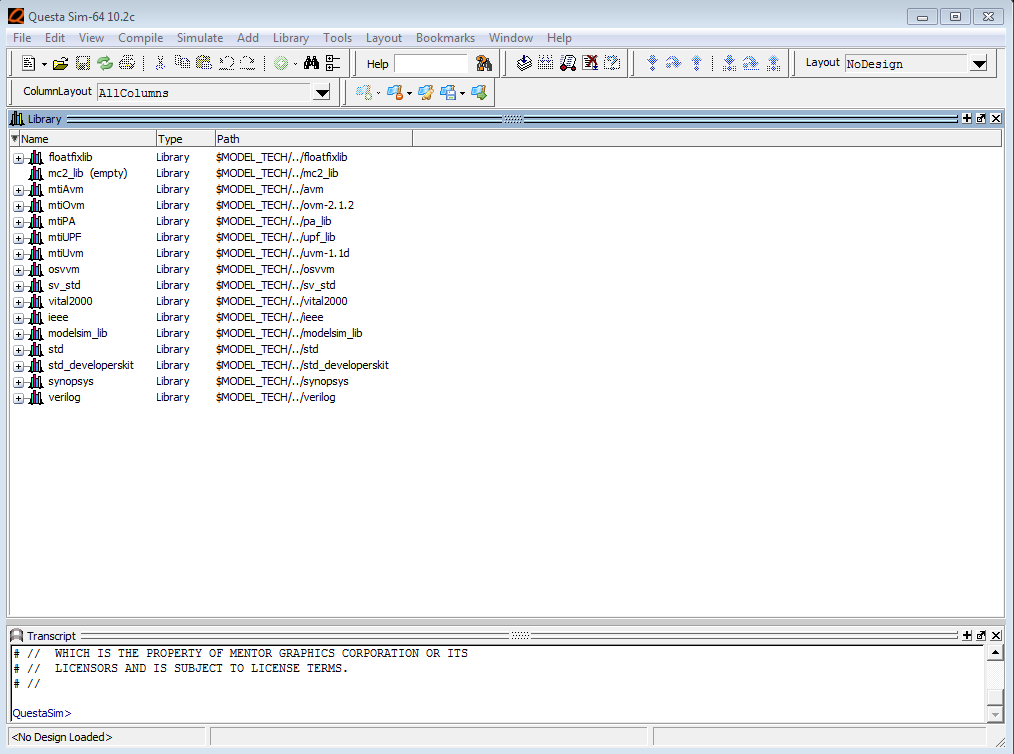
d:\DDB\A13

Dins d’aquest directori, crear un subdirectori per a la pràctica 1:

d:\DDB\A13\practica\_1

1. Executar el programa QuestaSim

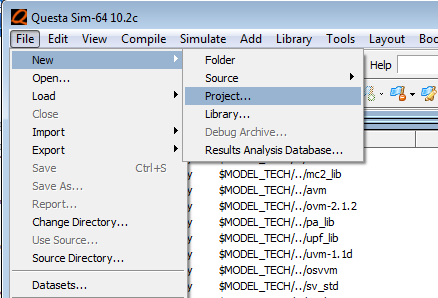
Activant l’icona d’inici, submenú QuestaSim-64 10.2c, o bé directament a través de l’accés directe de l’escriptori. Amb aquesta acció, accediríem a l’entorn següent:



1. Crear un nou Projecte

Seleccionar “*File*” → “*New*” → “*Project*”

Amb això tindrem definit un nou entorn de treball que serà únic per a cada tipus de codi.

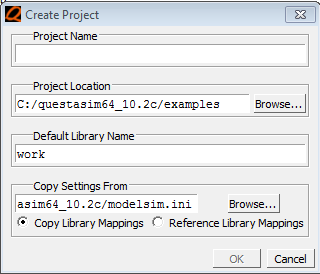


1. Apareix una nova finestra de diàleg en la qual s’ha d’especificar el nom del projecte i la seva ruta d’accés. Per exemple:

Nom del projecte (Project Name) = inversor\_basic

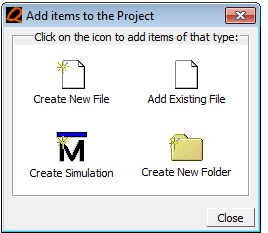
Ruta d’accés (Project Location) = d:\DDB\A13\ practica\_1\inversor\_basic

Graveu sempre els vostres fitxers al disc d.

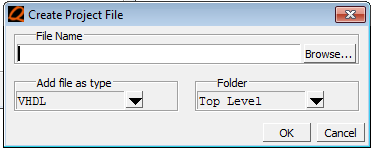


1. FITXERS VHDL. Inversor lògic
2. Creació o recuperació dels fitxers font VHDL

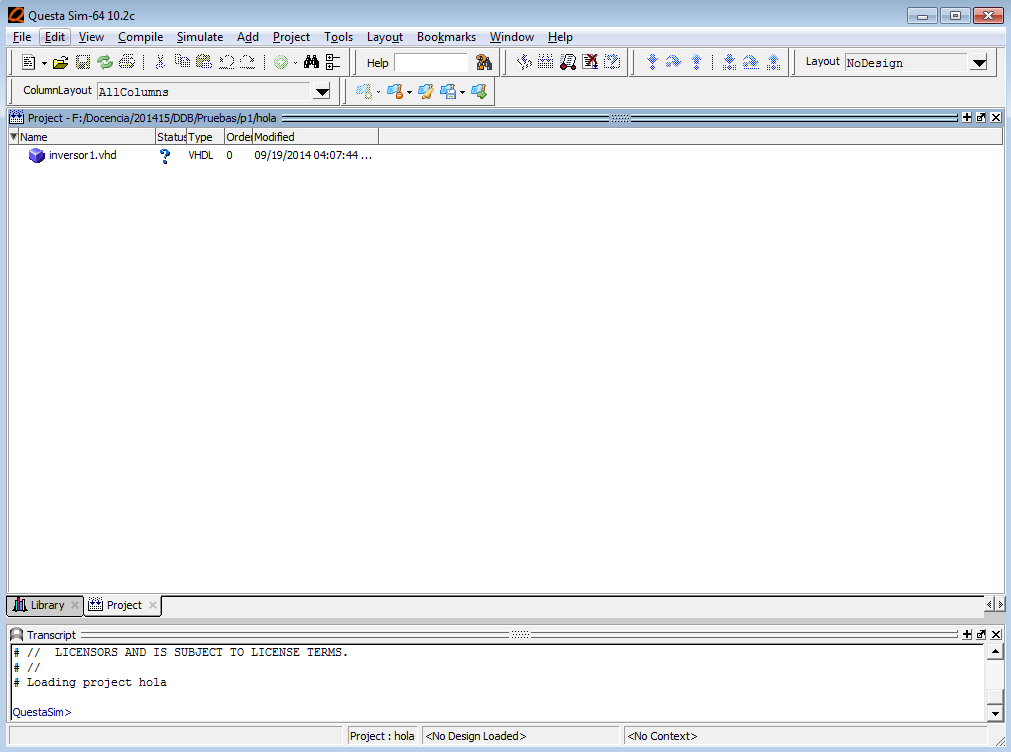
Una vegada creat el projecte, apareix una nova finestra a on podem crear un fitxer VHDL o cridar un existent:



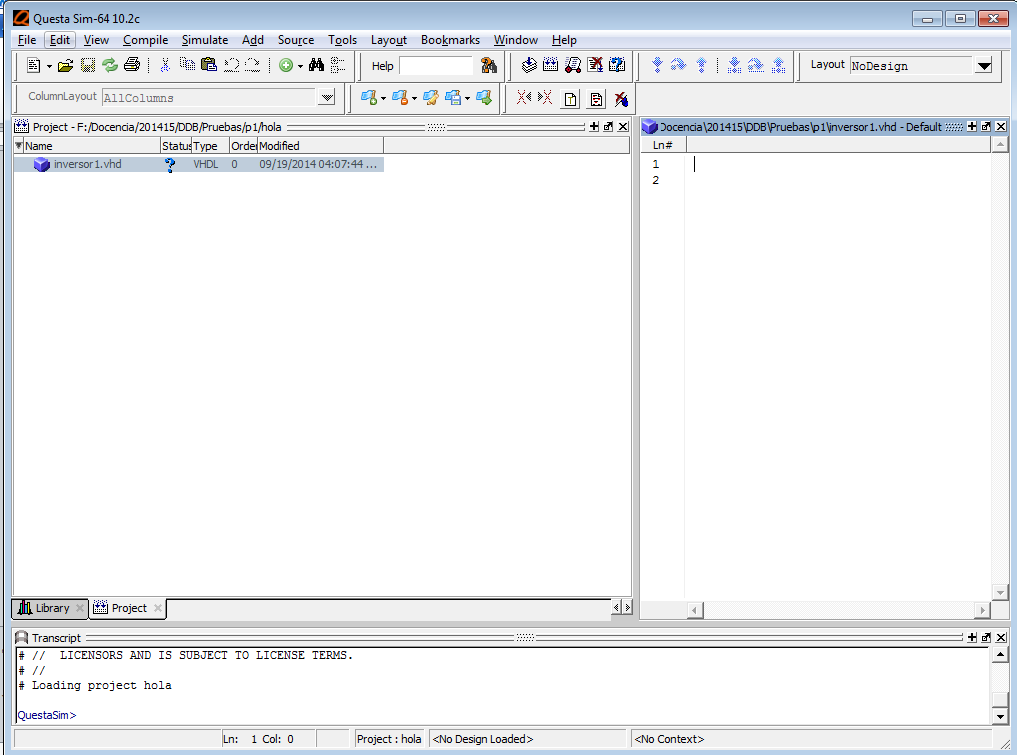
Seleccionar el símbol “*Create New File*”, indicant el nom i la ruta a on el voleu guardar. Per exemple, (*inversor1*).



1. Apareixerà una finestra amb els fitxers inclosos en el projecte (fins ara només un).



1. Podeu obrir el contingut del arxiu creat (buit fins ara), fent doble click sobre el nom del fitxer. Això obrirà la finestra de l’editor de text que ens permetrà introduir el codi a simular.



1. Ara podem entrar el codi corresponent a la funció lògica que volem implementar que, en aquest exemple, serà l’inversor. Cal definir el dispositiu (ENTITY) i la forma interna de connexió (ARCHITECTURE).

-- Inversor

-- Primer es declara l’entitat

ENTITY inversor IS

PORT (a : IN BIT;

f : OUT BIT);

END inversor;

-- Després es declara l’arquitectura

ARCHITECTURE logica OF inversor IS

BEGIN

f <= not a;

END logica;

Seleccionar el símbol “Save” per guardar el codi

El codi es guardarà a:

d:\DDB\A13\ practica\_1\inversor\_basic\inversor.vhd (és imprescindible afegir sempre l’extensió .vhd)

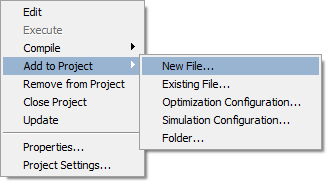
**Com que el llenguatge de programació és en anglès, cal que us aprengueu la sintaxi pròpia del programa.**

1. Compilació del disseny
2. Ara cal compilar el disseny i veure que no hi ha errors sintàctics ni de coherència en el codi generat (des del menú “*compile*”)

Si no hi ha errors sintàctics, a la pantalla de comandaments (a baix) diu que ha compilat ‘entitat i l’arquitectura.

**Si hi ha errors, podeu visualitzar el “*log*” del compilador fent doble click al “*status*” del codi en la finestra del projecte.**

1. Definició dels vectors de simulació o banc de proves (TestBench)
2. En un TestBench s’especifiquen els valors que agafen les diferents entrades que s’aplicaran a un determinat circuit lògic. Per a això, creem un nou fitxer que anomenarem testbench.vhd (botó dret dintre de la finestra del projecte, tal i com indica la figura):

A continuació es presenta un possible fitxer de vectors de simulació corresponent a la següent situació:

L’entrada inicial val 0, al cap de 100 ns val 1, després de 500 ns torna de nou a 0 i al cap de 1000ns, passa de nou a 1. A l’igual que amb el codi inversor.vhd, gravem el fitxer com a testbench.vhd.

ENTITY testbench IS

END testbench;

-- L’entitat banc de proves no té entrades ni sortides

ARCHITECTURE vectors OF testbench IS

-- Diem què volem testejar

COMPONENT inversor

port(

a: IN BIT;

f: OUT BIT);

END COMPONENT;

-- Definim senyals

SIGNAL a,f : BIT;

-- Diem quants i quins dispositius testeja el programa

FOR DUT: inversor USE ENTITY WORK.inversor(logica);

-- Relacionem el dispositiu i l’arquitectura

BEGIN

DUT: inversor PORT MAP (a,f);

-- Definim els terminals del dispositiu: la connexió entre els senyals que testejem

-- i els que senyals “muts” que porten les ENTITY

stimulus: PROCESS

BEGIN

a <= '0';

wait for 100 ns;

a <= '1';

wait for 500 ns;

a <= '0';

wait for 1000 ns;

a <= '1';

END PROCESS;

END vectors;

1. Compilació del disseny
2. Seleccionar el símbol “compile all”.

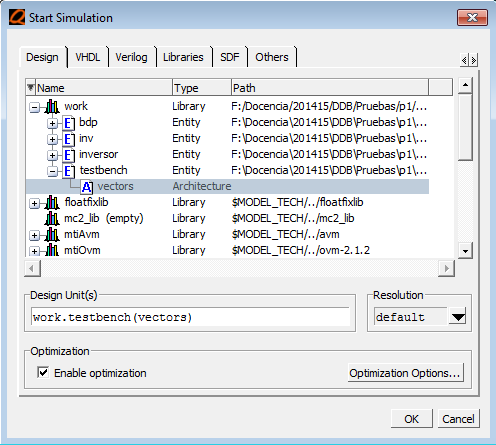
Desprès d’uns segons apareix, a la finestra de comandaments, el missatge que el fitxer ha estat compilat (sempre que no hi hagin errors de sintaxi)

Compile of inversor1.vhd was successful.

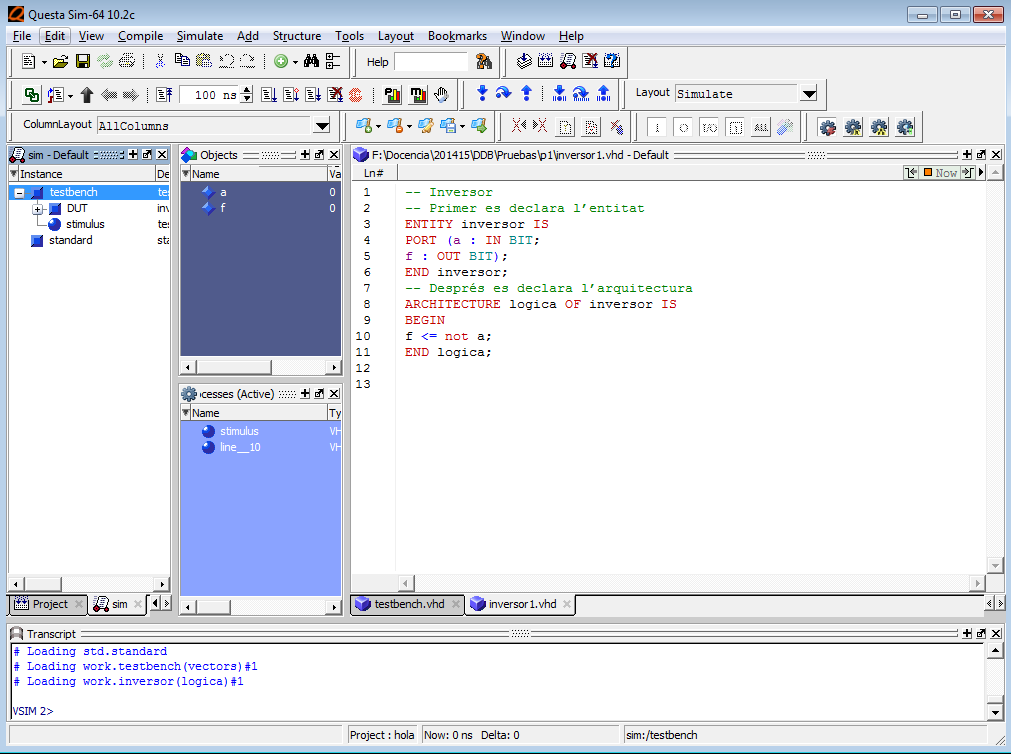
# Compile of testbench.vhd was successful.

# 2 compiles, 0 failed with no errors.

1. Activació del simulador i resultats
2. Comencem per definir l’entitat i arquitectura que volem visualitzar (menú “*simulate*” → “*start simulation*”) i cliquem OK:



1. Després d’uns pocs segons, l’entorn de finestres canviarà lleugerament, obtenint:



És ara quan podem afegir els senyals a i f (finestra blava del mig), clicar amb el botó dret i triar l’opció “*Add Wave*”. Això activarà la finestra d’ones, que ens permetrà observa l’evolució dels senyals en el temps.

1. Només cal fer corre el simulador durant un cert temps. Per fer-ho, hem d’anar al menú “*simulate*” → “*run*” → “*run 100*”.

Ara només ens cal analitzar la simulació i veure si, el que obtenim, és el resultat que esperaríem.