

Tutorial Quartus – Modelsim-Altera

Este tutorial descreve os passos elementares para a descrição de um circuito em VHDL no Quartus e sua simulação utilizando Modelsim-Altera.

1. No Quartus, crie um projeto chamado soma e entre com a descrição do somador abaixo, armazenada no arquivo *soma.vhd*:

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity soma is
    port (
        a      : in std_logic_vector(7 downto 0);
        b      : in std_logic_vector(7 downto 0);
        result : out std_logic_vector (7 downto 0)
    );
end entity;

architecture rtl of soma is
begin
    result <= std_logic_vector(signed(a) + signed(b));
end rtl;
```

2. Crie um novo arquivo VHDL contendo um *test bench* para o circuito acima, *soma_tb.vhd*:

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
use ieee.numeric_std.all;

ENTITY soma_tb IS
END soma_tb;
ARCHITECTURE soma_arch OF soma_tb IS
-- signals
SIGNAL a : STD_LOGIC_VECTOR(7 DOWNTO 0);
SIGNAL b : STD_LOGIC_VECTOR(7 DOWNTO 0);
SIGNAL result : STD_LOGIC_VECTOR(7 DOWNTO 0);

COMPONENT soma
    PORT (
        a : IN STD_LOGIC_VECTOR(7 DOWNTO 0);
        b : IN STD_LOGIC_VECTOR(7 DOWNTO 0);
        result : OUT STD_LOGIC_VECTOR(7 DOWNTO 0)
    );
END COMPONENT;

BEGIN
    i1 : soma
    PORT MAP (
        a => a,
        b => b,
        result => result
    );
```

```

init : PROCESS
-- variable declarations
BEGIN
    a <= X"04"; b <= X"05";
    wait for 4 ps;
    a <= X"32"; b <= X"05";
    wait for 4 ps;
    a <= X"04"; b <= X"FF";
    wait for 4 ps;
END PROCESS init;
END soma_arch;

```

3. Configurando a ferramenta e as definições de NativeLink no Quartus II

a) Na opção **Assignments**, clicar **Settings**.

b) Na lista Category, selecionar Simulation.

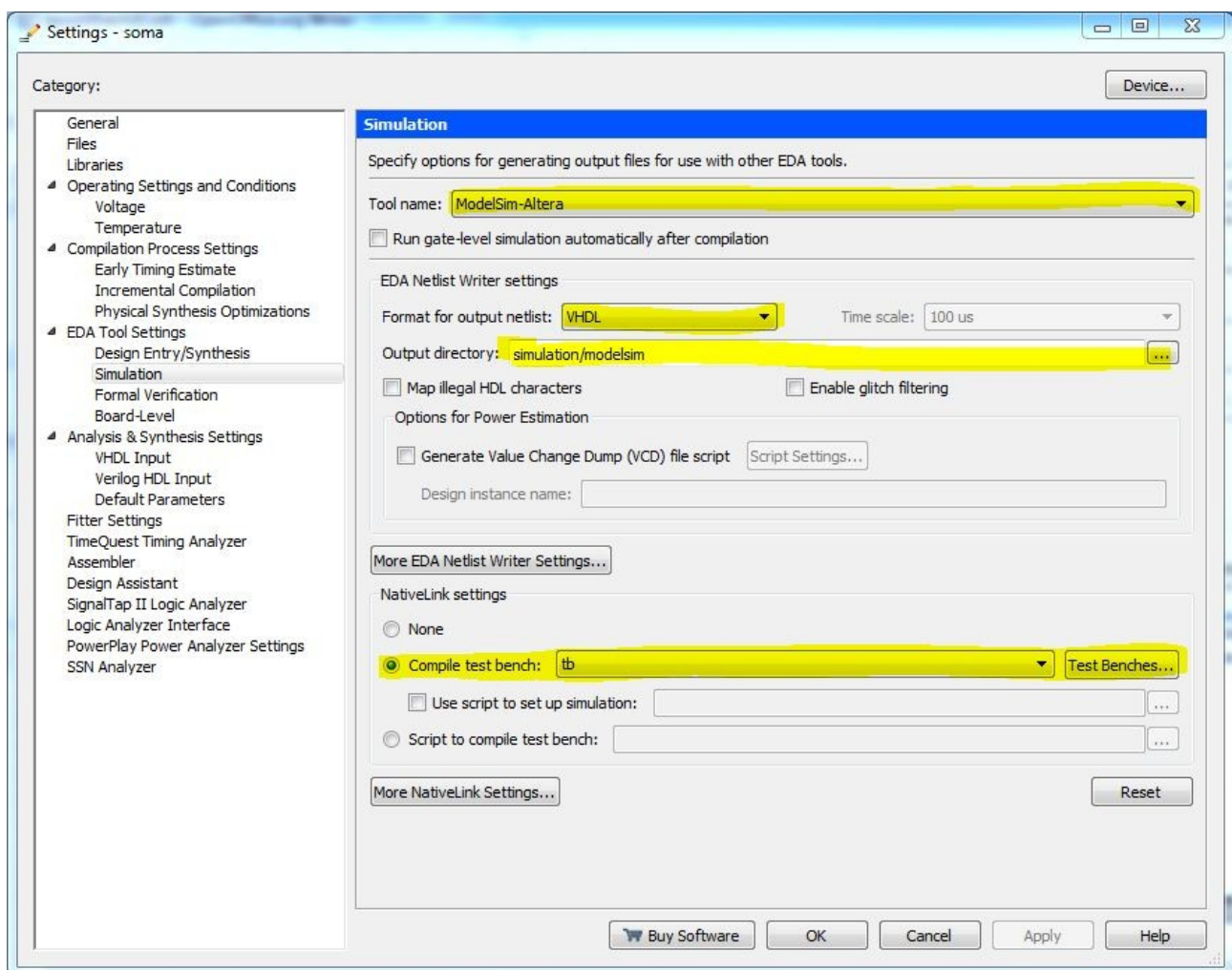


Figura 1. Configuração do ModelSim-Altera

c) Na lista Tool Name, selecionar Modelsim-Altera.

a) Se precisar localizar a ferramenta:

1. Cardápio **Tools** clicar **Options...**
2. Em **General** selecionar **EDA Tool Options**
3. No painel à direita, opção ModelSim-Altera, localizar o diretório onde foi instalado o ModelSim-Altera

4. Ex: C:\altera\12.1\modelsim_ase\win32aloem
- d) Se desejar executar simulação em nível de portas lógicas no Modelsim de forma automática depois da compilação do Quartus II, selecionar a opção **Run gate-level simulation automatically after compilation**.
- e) **Format output netlist** deve ser VHDL
- f) **Output directory** pode ser o default: **simulation/modelsim**

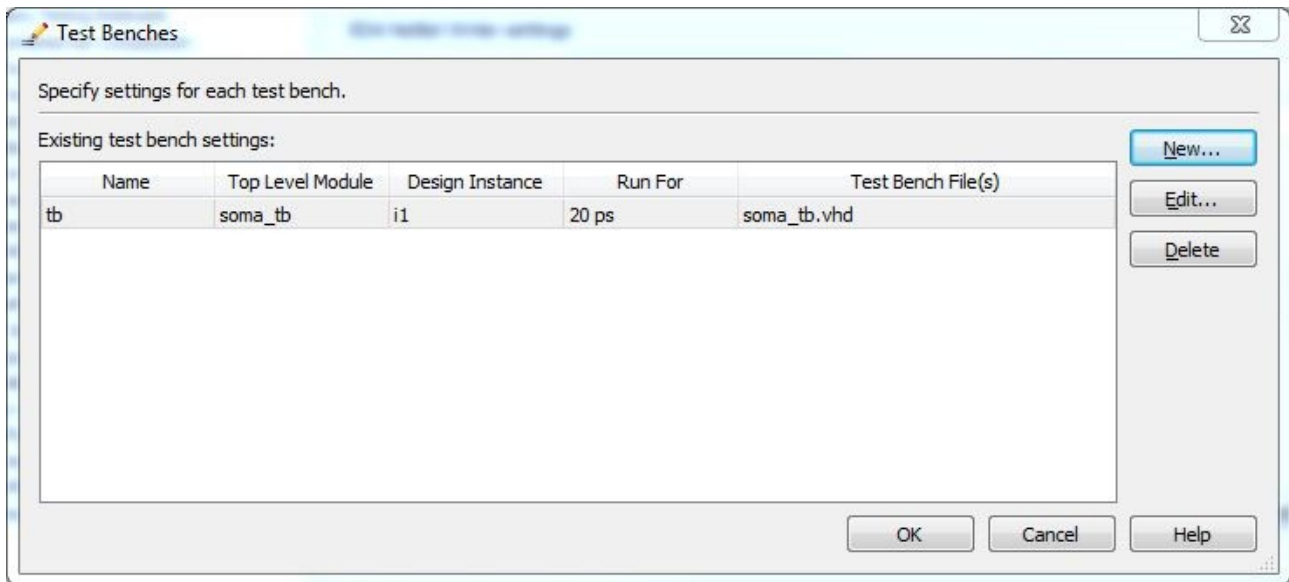


Figura 2. Configurando um test bench.

- g) Criar um *test bench* para a simulação em **NativeLink settings**
- Selecionar **Compile testbench**
 - Clicar botão **Test Benches**. Uma caixa de diálogo deve aparecer (figura 2).
 - Clicar **New**. A caixa de diálogo **New Test Bench Settings** aparece (figura 3).
 - Escolha um nome para o **test bench**: **tb**, por exemplo.
 - Defina o nome do módulo que contém o *test bench*: **soma_tb**
 - Marcar a caixa “Use test bench to perform VHDL timing simulation”
 - No campo **Design instance name in test bench** colocar o *label* do comando *port map* que instancia o módulo a ser simulado: **i1**.
 - Selecione um valor adequado de tempo para encerrar a simulação em **End simulation**: neste caso, 20 ps.
 - Em **Test bench files** selecione o arquivo que contém o *test bench*, **soma_tb.vhd**, e inclua-o no test bench (botão **Add**)
 - Clique ok em cada janela aberta.

Lance a simulação a partir do comando:

Tools → Run Simulation Tool → RTL Simulation

O ModelSim-Altera deverá ser automaticamente lançado. Para visualizar a forma de onda do resultado, selecionar a janela *Wave* para exibição:

View → Wave

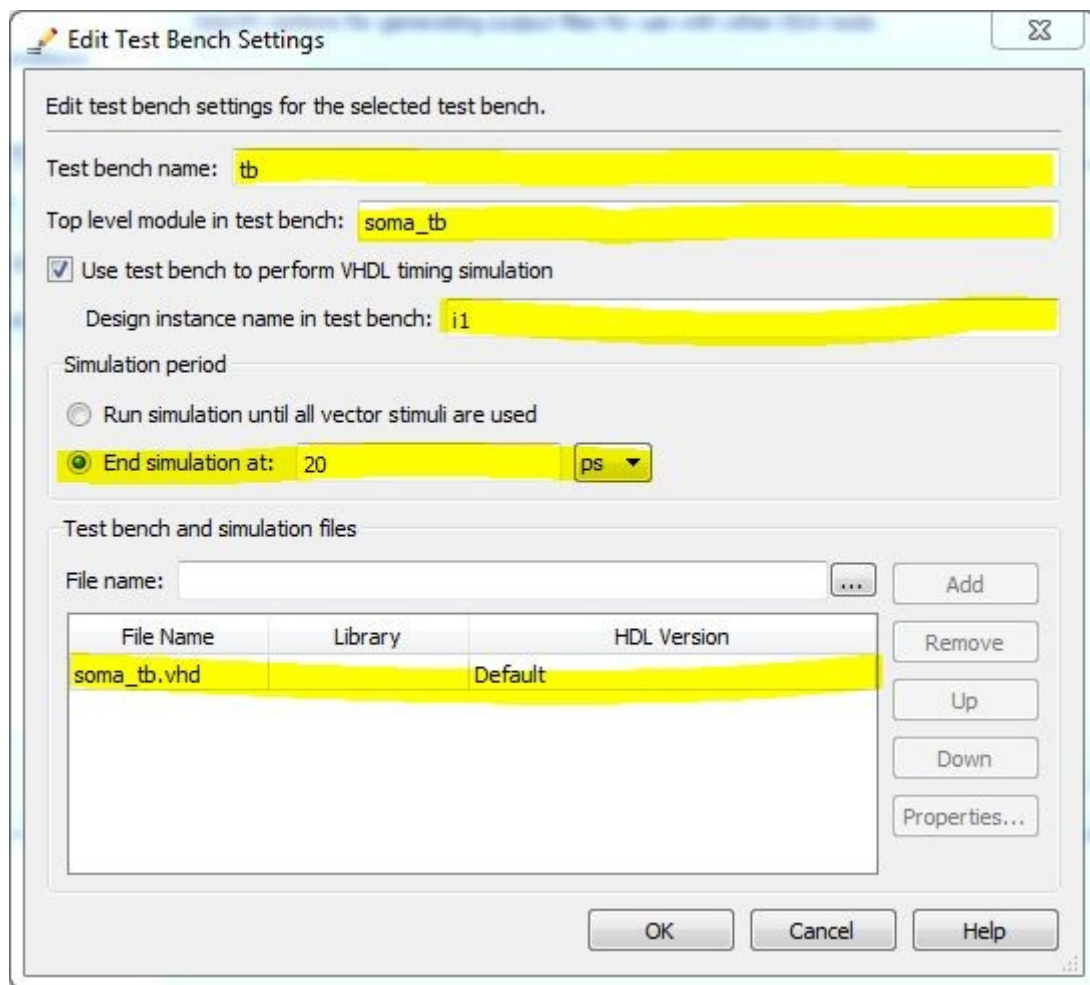


Figura 3. Especificando o test bench.

A janela exibida na figura 4 deve ser visualizada.

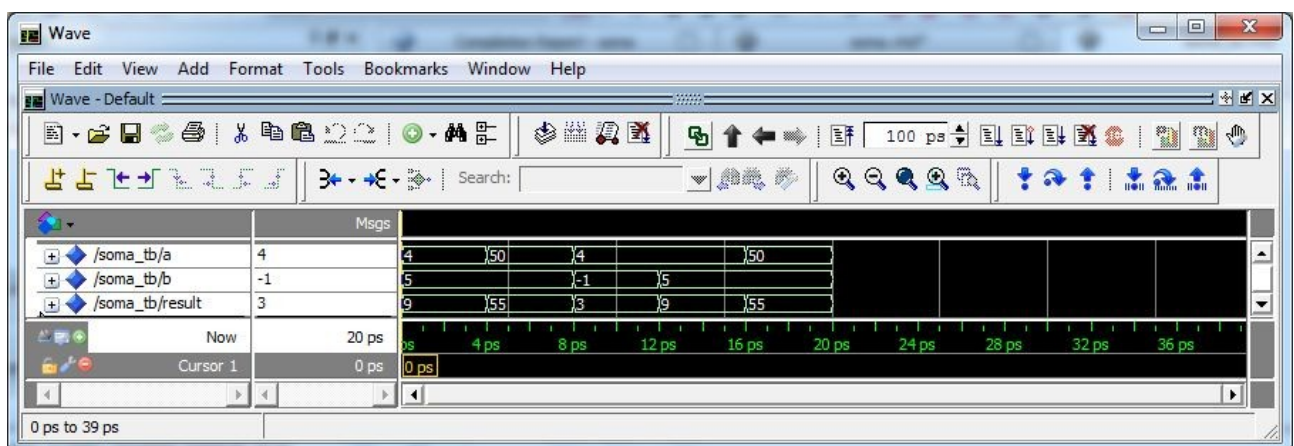


Figura 4. Formas de onda geradas pelo ModelSim-Altera – janela wave.