

INATEL - Instituto Nacional de Telecomunicações

E207 – Eletrônica Digital I

3º Período / 1º Semestre de 2021

Professores: Bruno de Oliveira Monteiro

bruno@inatel.br

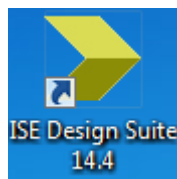
Monitores: Felipe Pereira Silveira
Carlos Daniel Borges Vilela Marques
Gualter Machado Mesquita
Isabela Rezende Barbosa da Silva
Maíra Alves Chagas
Pedro Henrique Praxedes dos Reis
Thalita Fortes Domingos

felipepereira@gea.inatel.br
carlos.marques@gea.inatel.br
machadomgualter@gmail.com
isabela.r@gec.inatel.br
mairaalves@gec.inatel.br
pedro.reis@gea.inatel.br
thalita.fortes@gec.inatel.br

Aluno: _____ **Matrícula:** _____ **Período:** ____ **Data:** ____ / ____ / ____

RELATÓRIO 5

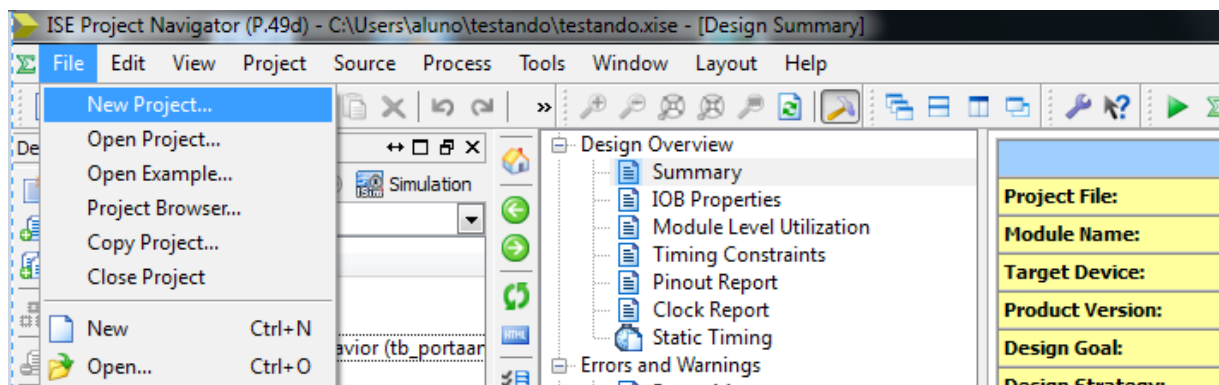
INTRODUÇÃO VHDL



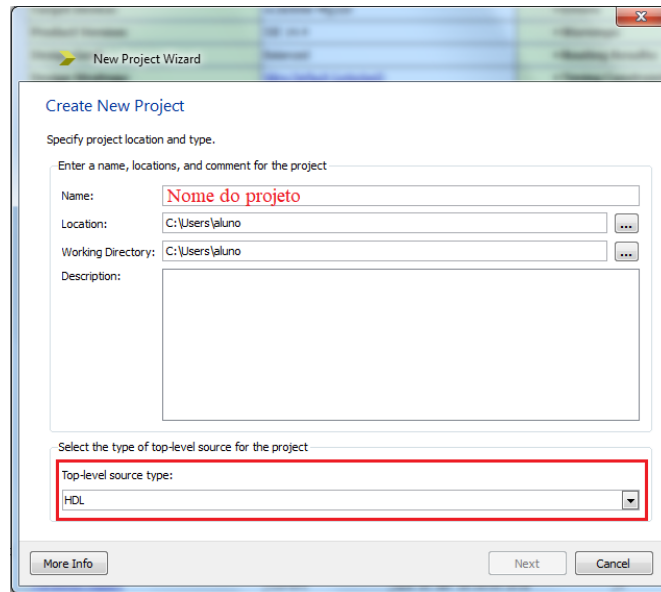
Software ISE Design Suite 14.4

Criando um Projeto no ISE

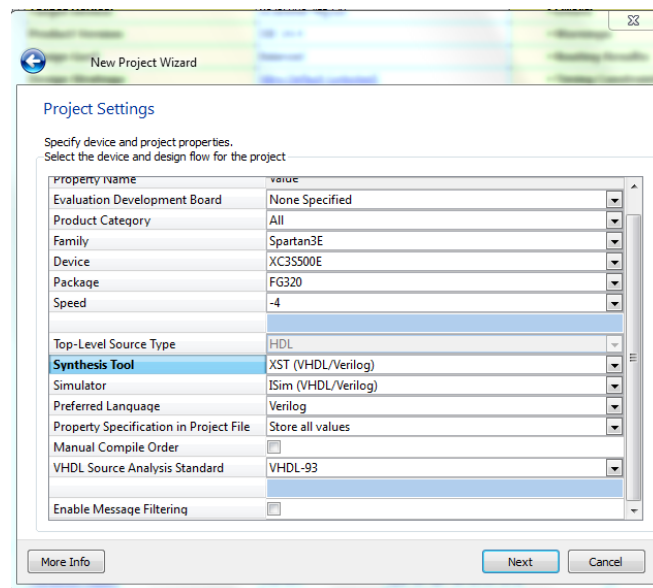
1º Passo: Clique em **File** e selecione a opção **New Project**.



2º Passo: Insira o nome do projeto e selecione o **tipo** do projeto (**HDL**).



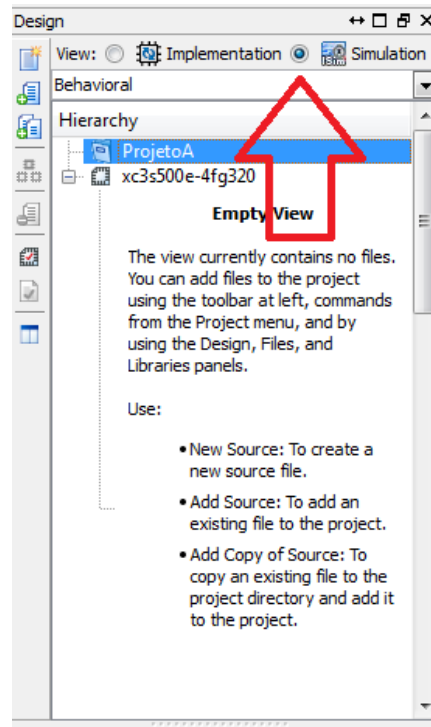
3º Passo: Após clicar em **Next**, realize as configurações do projeto de acordo com a imagem abaixo.



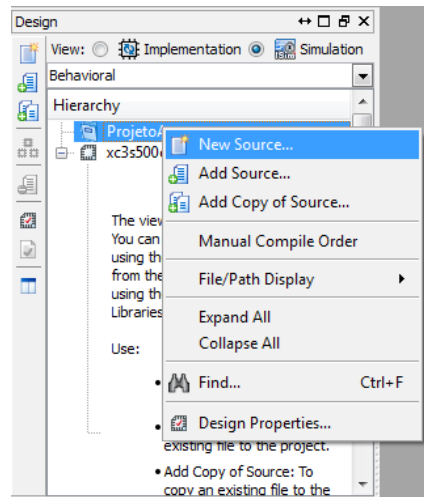
4º Passo: Clique em **Next e Finish**

Criando um código VHDL

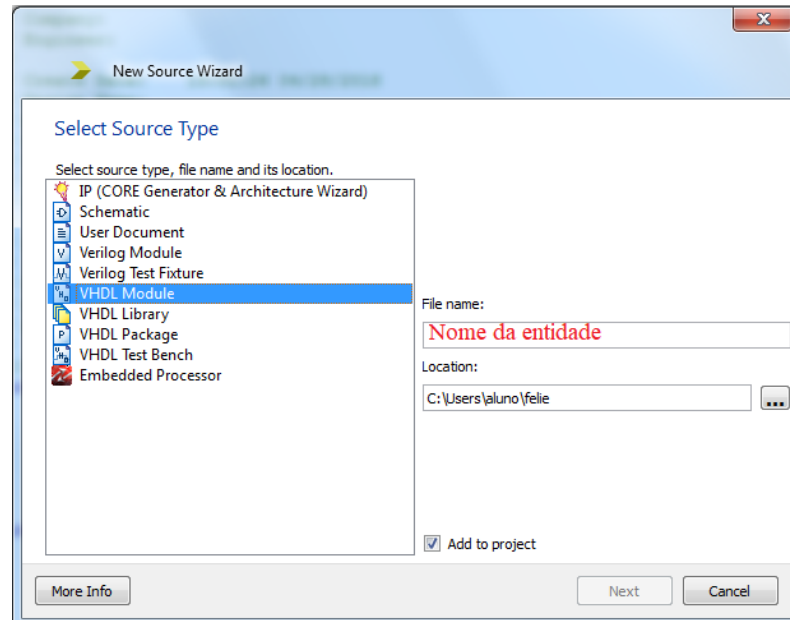
1º Passo: Selecione a opção **Simulation**.



2º Passo: Clique com botão direito no nome do seu projeto e selecione a opção **New Source**

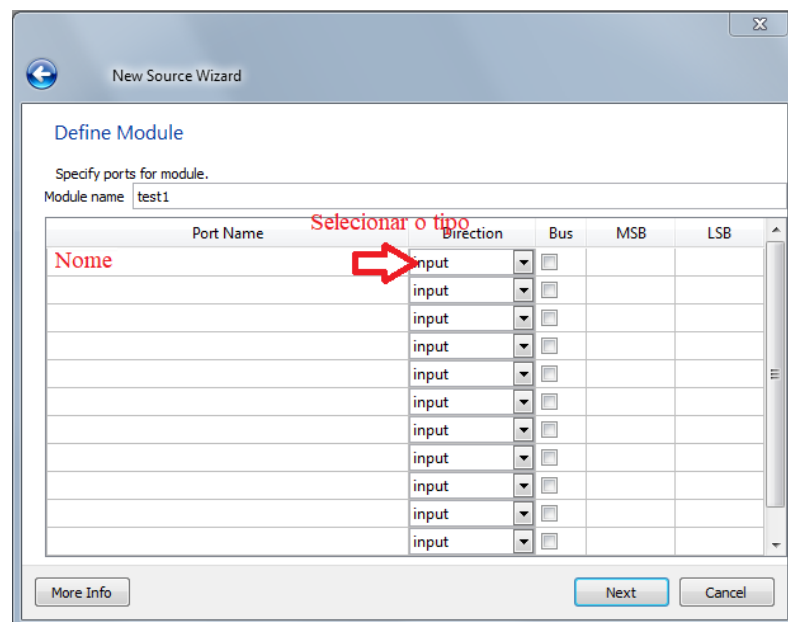


3º Passo: Marque a opção **VHDL Module**, insira o nome do arquivo/entidade e clique em **Next**.



4º Passo: Crie as variáveis de entrada e saída, selecionando seu nome e tipo (entrada ou saída).

Ex.: Vamos criar uma porta E com variáveis de entrada A e B e saída S.



Clique em **Next** e **Finish**. O software irá gerar um **código pronto**, já incluindo **a entidade** e **a arquitetura** (Caso o software não seja utilizado, essas linhas de código devem ser escritas).

Código gerado:

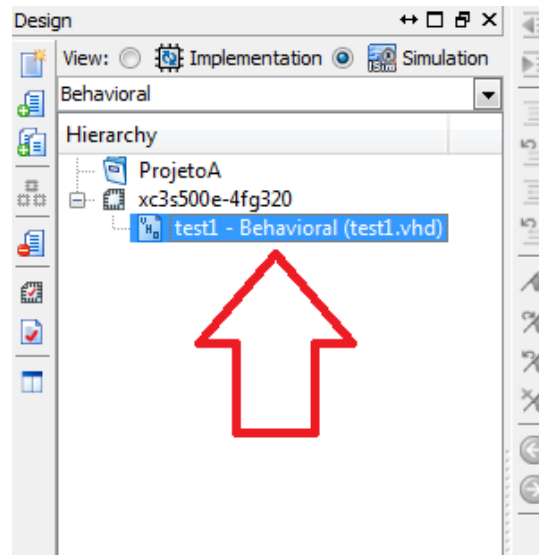
```
19 -----
20 library IEEE;
21 use IEEE.STD_LOGIC_1164.ALL;
22
23 -- Uncomment the following library declaration if using
24 -- arithmetic functions with Signed or Unsigned values
25 --use IEEE.NUMERIC_STD.ALL;
26
27 -- Uncomment the following library declaration if instantiating
28 -- any Xilinx primitives in this code.
29 --library UNISIM;
30 --use UNISIM.VComponents.all;
31
32 entity test1 is
33     Port ( a : in  STD_LOGIC;
34           b : in  STD_LOGIC;
35           s : out STD_LOGIC);
36 end test1;
37
38 architecture Behavioral of test1 is
39
40 begin
41 |
42
43 end Behavioral;
```

5º Passo: Dentro de **Begin** escreva a lógica do seu programa.

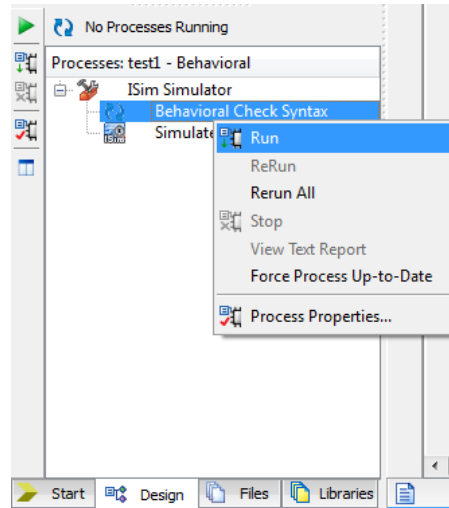
Para porta E a lógica fica: **S <= A and B;**

6º Passo: Salve o programa clicando em **File** e selecionando a opção **Save All**.

7º Passo: Selecione o arquivo de entidade gerado.

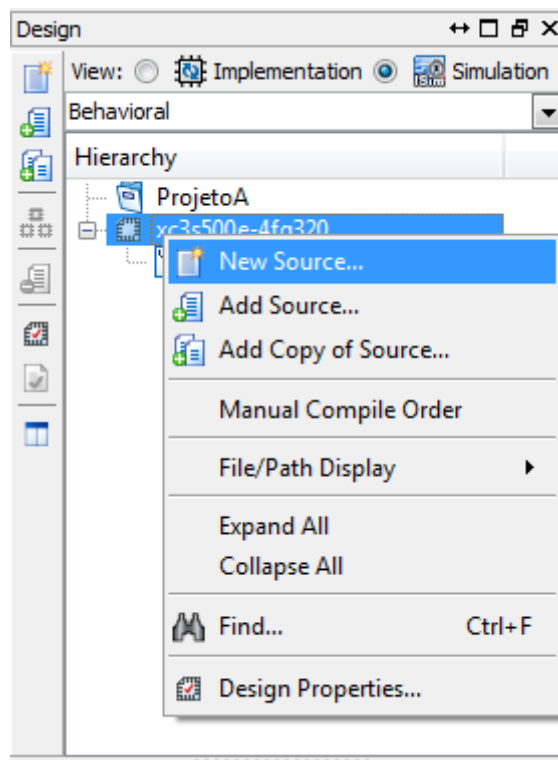


6º Passo: No item *ISim Simulator*, clique com botão direito no item *Behavioral Check Syntax* e selecione a opção *Run*. Nesse momento o compilador **verifica se há algum erro de sintaxe no código gerado**.

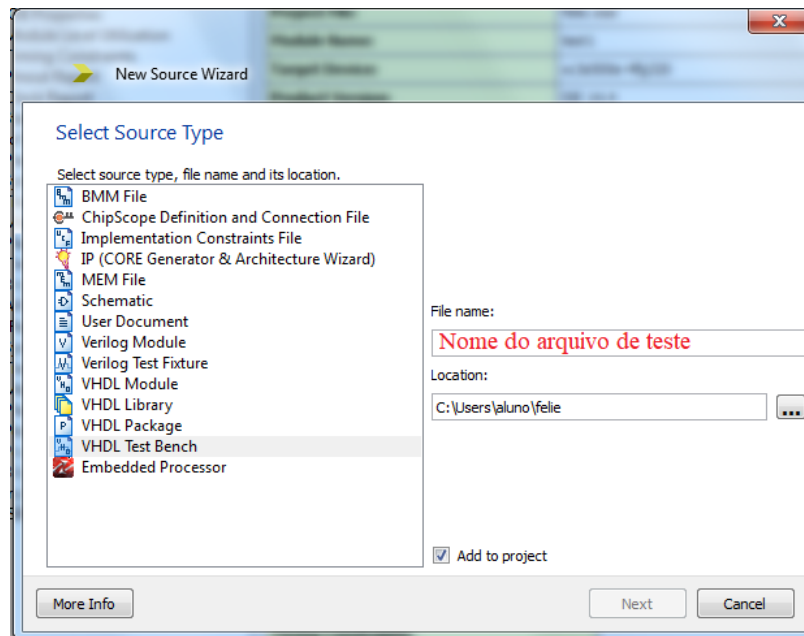


Criando um arquivo de teste

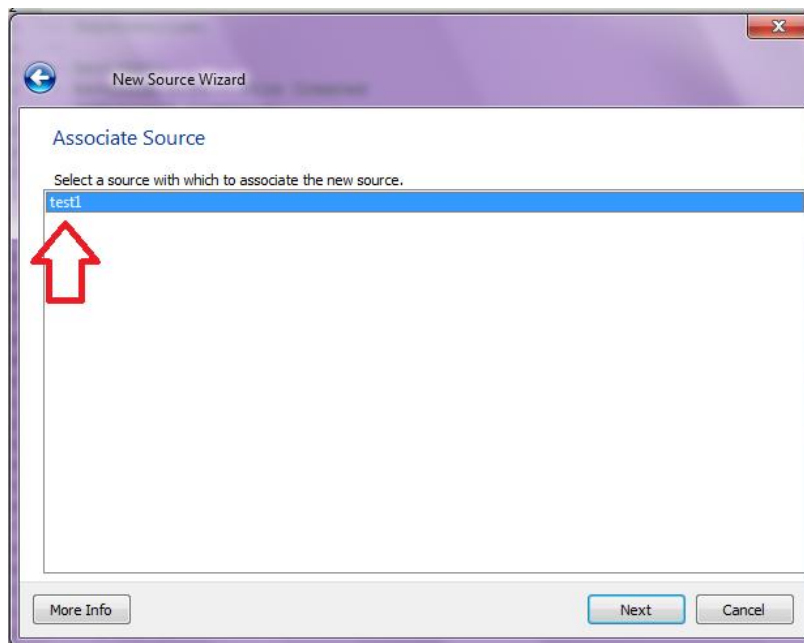
1º Passo: Clique com botão direito no item abaixo e selecione a opção *New Source* novamente.



2º Passo: Marque a opção **VHDL Test Bench**, insira o nome do arquivo de teste e clique em **Next**.



3º Passo: Associe o arquivo de *Test Bench* ao arquivo do código VHDL criado



4º Passo: Clique em **Next** e **Finish**. O software irá gerar um código pronto de teste (Caso o software não seja utilizado, essas linhas de código devem ser escritas).

5º Passo: Dentro do código gerado apague os seguintes trechos:

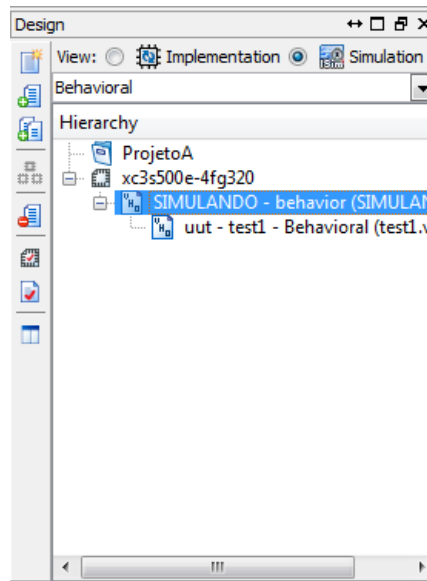
```
50
51 --Inputs
52 signal a : std_logic := '0';
53 signal b : std_logic := '0';
54
55 --Outputs
56 signal s : std_logic;
57 -- No clocks detected in port list. Replace <clock> below with
58 -- appropriate port name
59
60 constant <clock>_period : time := 10 ns;
61
62 BEGIN
63
64 -- Instantiate the Unit Under Test (UUT)
65 uut: test1 PORT MAP (
66     a => a,
67     b => b,
68     s => s
69 );
70
71 -- Clock process definitions
72 <clock>_process :process
73 begin
74     <clock> <= '0';
75     wait for <clock>_period/2;
76     <clock> <= '1';
77     wait for <clock>_period/2;
78 end process;
79
80
81 -- Stimulus process
82 stim_proc: process
83 begin
84     -- hold reset state for 100 ns.
85     wait for 100 ns;
86
87     wait for <clock>_period*10;
88
89     -- insert stimulus here
90
91     wait;
92 end process;
93
```

6º Passo: Dentro de **stim_proc**, insira as seguintes linhas de códigos. Essas linhas são referentes às variações que a entrada recebe.

```
71
72 -- Stimulus process
73 stim_proc: process
74 begin
75     a<='0';
76     b<='0';
77     wait for 100 ns;
78     a<='1';
79     b<='0';
80     wait for 100 ns;
81     a<='0';
82     b<='1';
83     wait for 100 ns;
84     a<='1';
85     b<='1';
86     wait for 100 ns;
87
88     -- insert stimulus here
89
90     wait;
91 end process;
92
93 END;
94
```

7º Passo: Salve o programa clicando em **File** e selecionando a opção **Save All**.

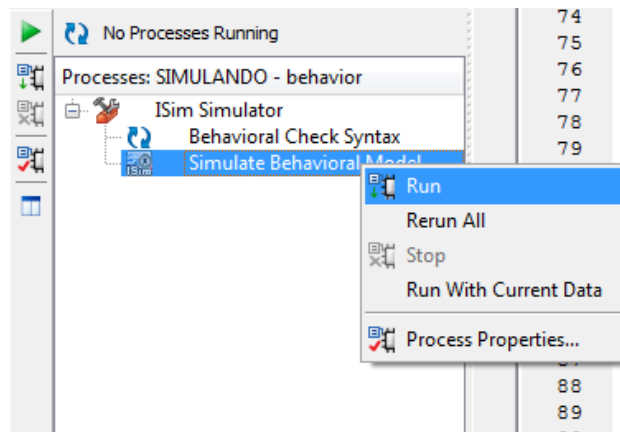
8º Passo: Selecione o arquivo de entidade gerado.



9º Passo: No item **ISim Simulator**, clique com botão direito no item **Behavioral Check Syntax** e selecione a opção **Run**.

Nesse momento o compilador **verifica se há algum erro de sintaxe no código** gerado.

Por fim, clique no item **Simulate Behavioral Model** e selecione a opção **Run**.



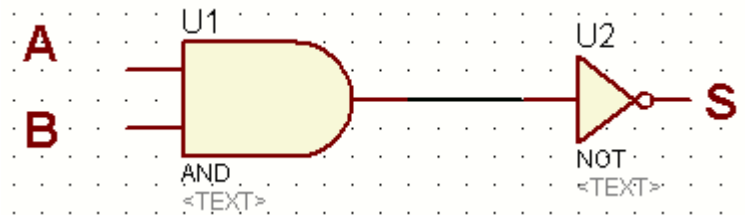
Após clicar em **Run** deve ser gerado um gráfico com os sinais de entrada e saída do circuito simulado. Para visualizar esse gráfico com maior clareza é preciso clicar na lupa!

EXERCÍCIO

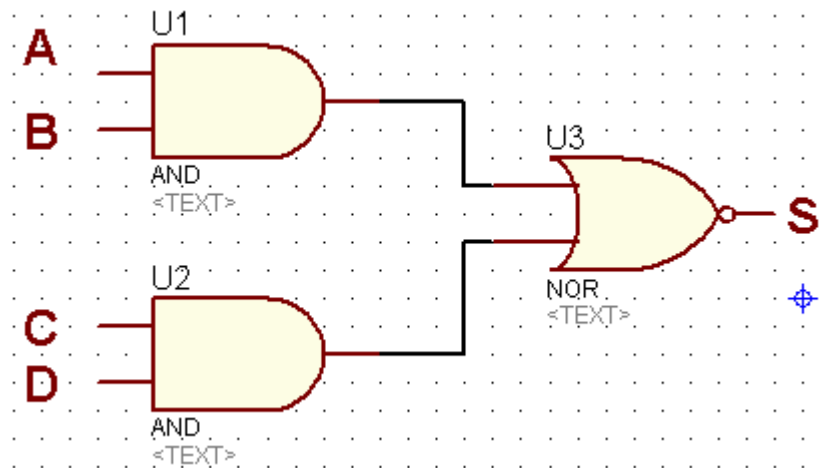
Questão 1. Crie o código e realize a simulação das seguintes portas lógicas: and, or, xor, xnor, not.

Questão 2. Crie o código dos seguintes circuitos combinacionais e realize a simulação:

a. Circuito 1



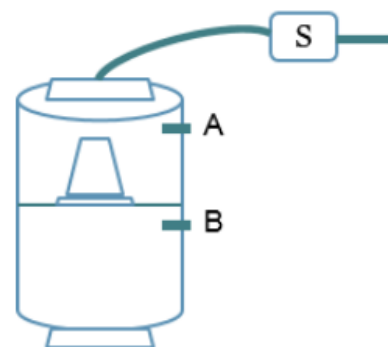
b. Circuito 2



Questão 3. Elabore um circuito lógico que permite encher automaticamente um filtro de água de dois recipientes a vela. A eletroválvula “S” permanecerá aberta quando tivermos nível “1” de saída do circuito e fechada quando nível “0”. O controle será feito por 2 sensores “A” e “B”.

Recipiente cheio, sensor = “1”;

Recipiente vazio, sensor = “0”;



OBS: O filtro possui uma divisória entre os sensores A e B, pois a água não filtrada é colocada na parte superior, e vai sendo filtrada aos poucos e pingando na parte inferior.