



Lista de Exercícios – 01 - Respostas

1. Quais os motivos que levaram ao desenvolvimento da linguagem de descrição de hardware.

A linguagem VHDL deve seu desenvolvimento à necessidade de uma ferramenta de projeto e documentação padrão para o projeto VHSIC (Very High Speed Integrated Circuit) do Departamento de Defesa (DoD) dos Estados Unidos. Em 1981 o DoD patrocinou um encontro de especialistas para discutir métodos para descrição de circuitos. Em 1983, o DoD definiu os requisitos de uma linguagem padrão para descrição de circuitos e concedeu um contrato com as empresas IBM, Intermetrics e Texas Instruments para o desenvolvimento da linguagem e ferramentas.

A linguagem de descrição de hardware (HDL – Hardware Description Language) apresentava diversas vantagens em relação aos métodos tradicionais de descrição de circuitos. Os métodos utilizados eram: o diagrama lógico, com as interconexões das portas lógicas e dos módulos do circuito; e as expressões que descreviam o circuito. Para circuitos mais complexos, esses métodos não eram adequados em função das dificuldades de se representar o circuito. Além disso, o DoD adquiria muitas placas de circuitos integrados de aplicação específica (ASICs – Application Specific Integrated Circuits) de diferentes fabricantes. E cada fabricante possuía seu padrão de descrição de circuitos. Alguns fabricantes deixavam de produzir os equipamentos ou porque faliam ou mudavam de ramo de atividade. Tudo isso dificultava a manutenção e reposição de circuitos.

Esses fatores levaram ao desenvolvimento da linguagem de descrição de hardware.

2. O que significa HDL e para que serve? Dê exemplos de HDLs.

HDL significa Hardware Description Language. HDL é uma linguagem baseada na criação e utilização de textos estruturados que descrevem o comportamento de um circuito digital em diversos níveis de abstração, como, por exemplo, comportamental, transferência entre registradores e estrutural. HDL facilita o desenvolvimento de projetos, permite a execução de testes e avaliação de pontos críticos, possibilitando a análise do funcionamento do sistema a ser construído. Alguns exemplos de HDLs são: AHDL, Verilog, SystemC, Handel-C, SDL, ISP, ABEL e VHDL.

3. O que significa o acrônimo VHDL?

O acrônimo VHDL significa:

- V – Very High Speed Integrated Circuits,
- H – Hardware,
- D – Description,
- L – Language.

A tradução deste acrônimo para o português é “Linguagem de Descrição de Hardware com ênfase em Circuitos Integrados de Altíssima Velocidade”.

4. Marque nas sentenças a seguir (1) para identificar vantagens e (2) para identificar desvantagens do uso de VHDL.

- (1) Projeto independente da tecnologia;
- (2) Hardware gerado é menos otimizado;
- (1) Facilidade na atualização dos projetos;
- (2) Falta de pessoal treinado para lidar com a linguagem;
- (1) Redução do tempo de projeto e custo;
- (2) Simulações geralmente mais lentas que outras implementações;
- (1) Eliminação de erros de baixo nível;
- (1) Simplificação da documentação.



5. Das características da linguagem VHDL identifique as sentenças verdadeiras (V) e falsas (F):

(V) VHDL é análogo a uma linguagem de programação.
(F) Sintaxe em VHDL não é similar à sintaxe das linguagens Ada e Pascal.
(V) VHDL deve ser descrito após a arquitetura, e não a arquitetura após a VHDL.
(F) VHDL não provê mecanismos para modelar a concorrência e sincronização que ocorrem no nível físico no hardware.
(V) O código em VHDL é executado em um simulador, não há um “executável”.
(V) Permite, através de simulação, verificar o comportamento do sistema digital.
(V) Permite descrever hardware em níveis de abstração, tais como: Comportamental, RTL e Estrutural.
(F) Não favorece projeto “top-down”.

6. Identifique as diferenças entre VHDL (1) e linguagens de programação (2).

7.

Propósito	(1) Hardware	(2) Software
Entrada	(1,2) Texto e Ferramentas Visuais	(1,2) Texto e Ferramentas Visuais
Desenvolvimento	(2) Compilação e Ligação/interpretação	(1) Compilação para Simulação e Síntese para Hardware
Depuração	(2) Execução e Visualização dos Resultados	(1) Simulação e Visualização das formas de ondas
Instruções	(1) Concorrentes e sequenciais	(2) Somente sequenciais

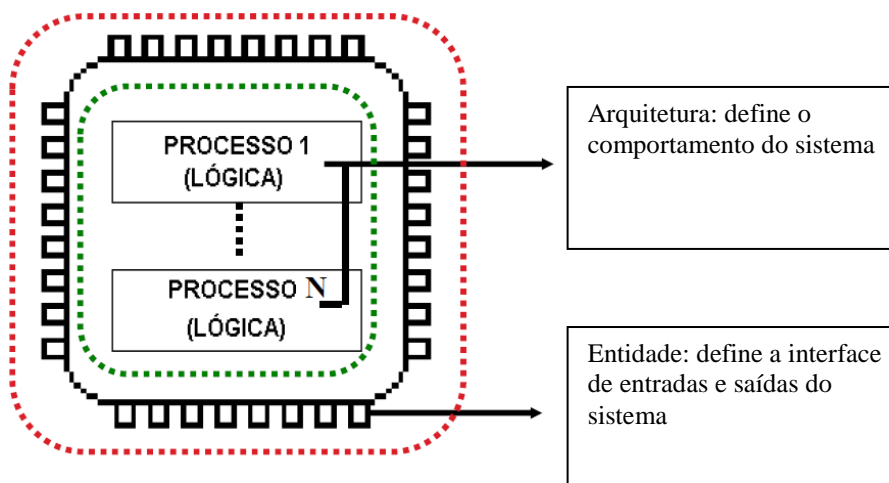
8. Quanto a síntese em VHDL marque verdadeiro (V) ou falso (F) para as sentenças a seguir:

(V) A modelagem em RTL é 100% sintetizável.
(F) A descrição comportamental não depende da ferramenta de síntese utilizada.
(F) A modelagem RTL é 50% sintetizável.
(F) A descrição comportamental não depende das construções VHDL utilizadas no código.
(V) A descrição comportamental depende da ferramenta de síntese utilizada e das construções VHDL utilizadas no código.

9. A estrutura básica de um código em VHDL consiste em três partes: (1) declarações de bibliotecas e pacotes, (2) entidade e (3) arquitetura:

(X) Verdadeiro
() Falso

10. Na figura a seguir identifique e defina quem é Entidade e quem é Arquitetura:



11. Para a implementação no nível de portas lógicas quais as arquiteturas possíveis em VHDL? Descreva cada uma delas.

Existem 3 modelagens possíveis para arquiteturas em VHDL: (a) comportamental, (b) RTL ou fluxo de dados e (c) estrutural.

- Descrição Comportamental: é o nível mais alto, descreve o projeto em termos do comportamento do sistema utilizando algoritmos.
- Descrição RTL ou Fluxo de Dados: descreve o projeto em termos da transferência de dados da entrada para a saída e entre os sinais. Esta descrição é similar à expressão booleana de um circuito.
- Descrição Estrutural: é o nível mais baixo, descreve a estrutura do projeto em termos de portas lógicas e suas interconexões. Esta descrição é similar aos diagramas de blocos lógicos de um circuito.

12. Obtenha as entidades dos circuitos apresentados nas Figuras 1, 2 e 3.

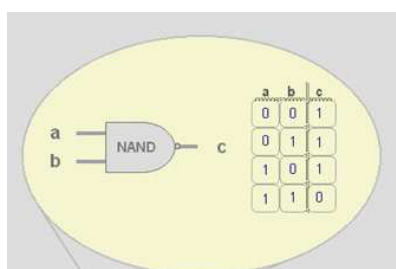


Figura 1

```
ENTITY porta_nand IS
    PORT (a, b: IN BIT;
          c: OUT BIT);
END porta_nand;
```

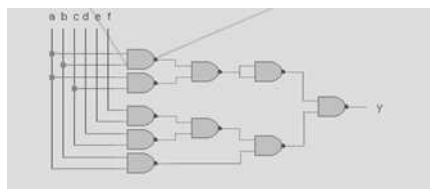


Figura 2

```
ENTITY circuito_2 IS
    PORT (a, b, c, d, e, f: IN BIT;
          y: OUT BIT);
END circuito_2;
```

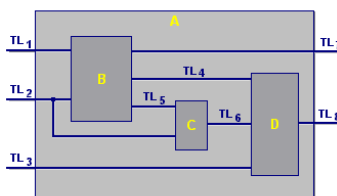


Figura 3

```
ENTITY circuito_3 IS
    PORT (TL1, TL2, TL3: IN BIT;
          TL7, TL8: OUT BIT);
END circuito_3;
```