



Universidade Federal de Santa Catarina
Centro Tecnológico
Departamento de Informática e Estatística
Curso de Graduação em Ciências da Computação



Lógica Programável

INE 5348

Aula 1-P

**Formas de implementação de sistemas digitais.
Introdução à Linguagem VHDL e familiarização com o
Quartus II da Altera.**

Prof. José Luís Güntzel
guntzel@inf.ufsc.br

www.inf.ufsc.br/~guntzel/ine5348/ine5348.html

Introdução à Linguagem VHDL e Uso do Quartus II

Histórico

- **Usada para modelagem, simulação, prototipação e síntese de sistemas digitais**
- **Criada sob encomenda da marinha estadunidense**
- **VHDL: VHSIC Hardware Description Language**
- **VHSIC: Very High Speed Integrated Circuits**
- **Versões padronizadas pela IEEE**

Introdução à Linguagem VHDL e Uso do Quartus II

▶ Primeiros Conceitos

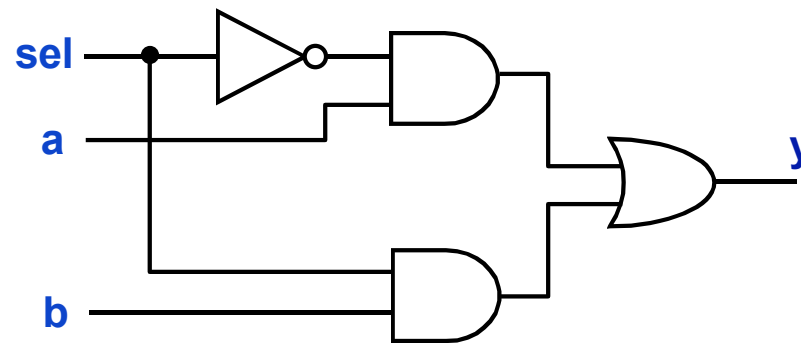
Uma descrição VHDL é dividida em **duas partes fundamentais**:

- 1) **Entidade (Entity)** – Descreve a interface do sistema digital descrito com o mundo externo. Apresenta a definição dos pinos de entrada e saída.
- 2) **Arquitetura (Architecture)** – Descreve o comportamento ou a estrutura do sistema digital. Define como a função do sistema é realizada.

Introdução à Linguagem VHDL e Uso do Quartus II

▶ Primeiros Conceitos

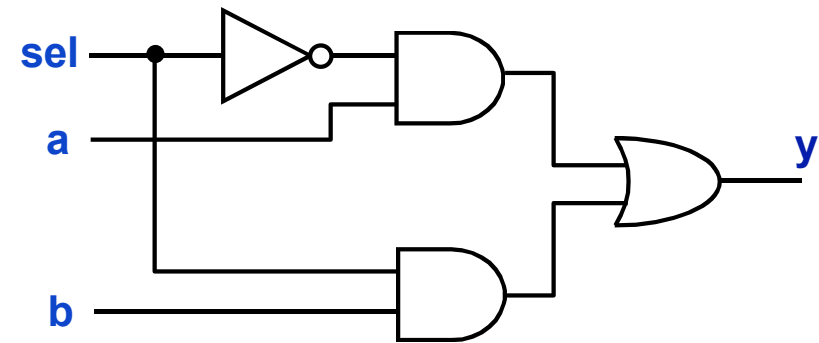
Circuito Exemplo:



Introdução à Linguagem VHDL e Uso do Quartus II

▶ Primeiros Conceitos: Entidade

```
ENTITY exemplo1 IS  
  PORT ( sel : IN BIT;  
        a : IN BIT;  
        b : IN BIT;  
        y : OUT BIT);  
END exemplo1;
```



Importante: o nome do arquivo vhdl **DEVE** ser o mesmo nome da entidade. Neste caso, o nome do arquivo seria *exemplo1.vhd*

Introdução à Linguagem VHDL e Uso do Quartus II

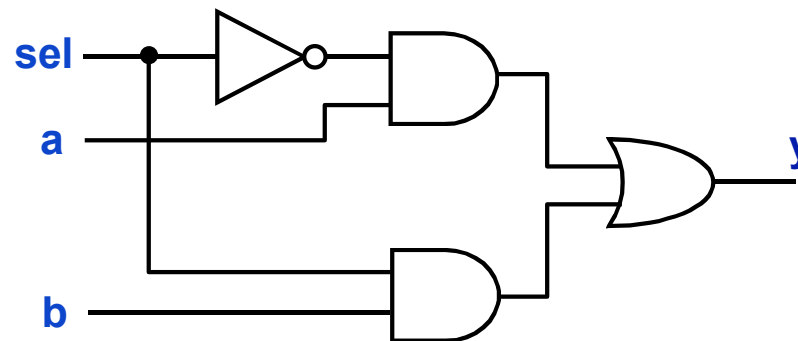
▶ Primeiros Conceitos: Arquitetura

ARCHITECTURE comportamento **OF** exemplo1 **IS**

BEGIN

y <= (a AND (NOT(sel))) OR (b AND sel);

END comportamento;



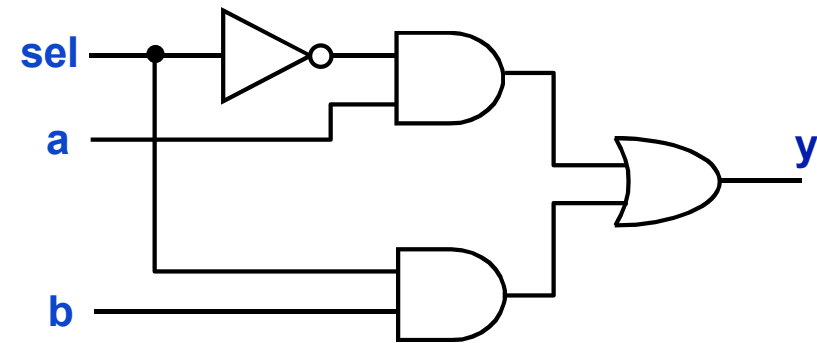
Introdução à Linguagem VHDL e Uso do Quartus II

▶ Primeiros Conceitos: circuito completo

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

```
ENTITY exemplo1 IS  
  PORT ( sel, a, b : IN BIT;  
        y : OUT BIT);  
END exemplo1;
```

```
ARCHITECTURE comportamento OF exemplo1 IS  
BEGIN  
  y <= (a AND (NOT(sel))) OR (b AND sel);  
END comportamento;
```

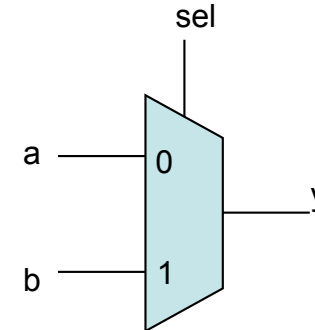


Introdução à Linguagem VHDL e Uso do Quartus II

Multiplexador 2 para 1

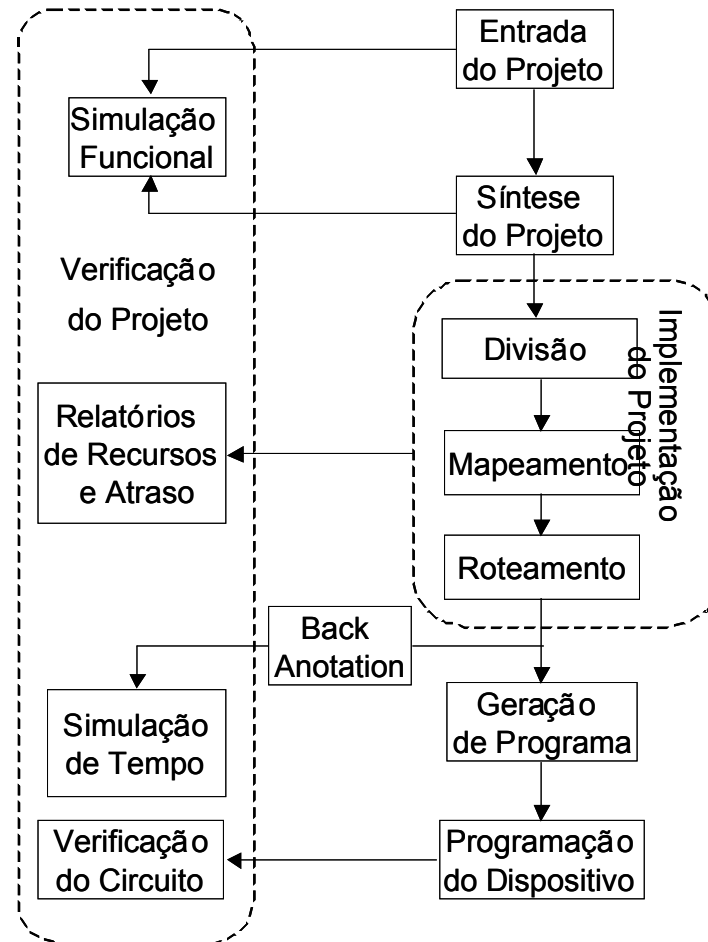
```
LIBRARY ieee;  
USE ieee.std_logic_1164.all;  
  
ENTITY mux2para1 IS  
PORT ( sel, a, b : IN STD_LOGIC;  
      y : OUT STD_LOGIC);  
END mux2para1;
```

```
ARCHITECTURE comportamento OF mux2para1 IS  
BEGIN  
  WITH sel SELECT  
    y <= a WHEN '0',  
      b WHEN OTHERS;  
END comportamento;
```



Introdução à Linguagem VHDL e Uso do Quartus II

► Fluxo de Projeto com FPGAs



slide 1P.9

Prof. José Luís Güntzel

Introdução à Linguagem VHDL e Uso do Quartus II

▶ Quartus II

- Desenvolvida pela **Altera**
- Suporta apenas FPGAs da Altera
- Possui versão para uso acadêmico (“Web Edition”), com licença grátis (www.altera.com link “licensing”)
- Versão atual é 7.1 SP1

Introdução à Linguagem VHDL e Uso do Quartus II

Quartus II

Passos para o projeto:

1. Analisar o problema e “bolar” uma solução, em termos de sistema digital
2. Descrever a solução em VHDL (criar um projeto com diversos arquivos VHDL)
3. Compilar o projeto para um determinado dispositivo FPGA
4. Anotar os dados resultantes da compilação: no de ALUTs, atraso crítico, frequência máxima...
5. Verificar o projeto
6. Gerar o arquivo de programação do FPGA
7. Baixar o arquivo para placa (gravar em EPROM na placa onde está o sistema com o FPGA)

Introdução à Linguagem VHDL e Uso do Quartus II

▶ Quartus II

4. Verificar o projeto:

1. Criar arquivo de simulação
2. Definir formas de onda de entrada para a simulação
3. Salvar arquivos de simulação
4. Rodar a simulação e verificar se o resultado está dentro do esperado

Pergunta: supondo um projeto com 3 entradas, quantas simulações precisam ser feitas para se garantir o funcionamento correto do projeto?

- Para funcionamento estático: $2^3 = 8$
- Para funcionamento dinâmico: $2^3 \times (2^3 - 1) = 56$

Introdução à Linguagem VHDL e Uso do Quartus II

Quartus II

Criar um Projeto:

1. File-> New ; escolher “VHDL File”; OK
2. Editar o arquivo VHDL (copiar o conteúdo da transparência 7)
3. File -> Save As; completar o nome com “exemplo1”; OK
4. Clicar em “create new project”
5. Next
6. Escolher dispositivo Stratix II; Next
7. “Specify the other EDA Tools” não clicar em nada; Next
8. Finish

Introdução à Linguagem VHDL e Uso do Quartus II

Quartus II

Compilar

1. Processing -> Start Compilation
2. Aguardar mensagem “Quartus II Full Compilation Succesfull” (ou mensagem de erro)
3. Anotar
 - Nome (código) do dispositivo escolhido automaticamente
 - Caminho crítico (pino de entrada/pino de saída) e seu atraso (“Longest tpd from souce pin ...”)
 - Número de ALUTs utilizadas

Introdução à Linguagem VHDL e Uso do Quartus II

▶ **Quartus II**

Preparar Arquivo para Simulação Funcional

1. Criar o arquivo de simulação: New->New->Other Files->Vector Waveform File
2. Clicar com botão direito do mouse na coluna chamada “Name”
3. Escolher Insert -> Insert Node or Bus -> Node Finder
4. No Node Finder, selecionar os nomes dos sinais a serem observados (entradas e saídas): no Filter, selecionar “Pins: all”; clicar no botão “List”
5. Selecionar a, b, sel, out, clicar em “>” e em “OK”
6. “multiple items”; OK
7. Especificar as formas de onda para os sinais de entrada (a, b, sel), usando o botão “Waveform Editing Tool” (usar a lupa para aumentar/reduzir o zoom)
8. Salvar o arquivo de forma de ondas com o mesmo nome do projeto: File-> Save As

Introdução à Linguagem VHDL e Uso do Quartus II

Quartus II

Executar a Simulação Funcional e Analisar o Resultado

1. Processing -> Start Simulation
2. Para medir tempos de atraso específicos: clicar com o botão da direita e seleccionar “Time Bar -> Insert Time Bar”