

# Introdução ao Quartus Prime

Felipe Valencia de Almeida

Profa. Dra. Liria Sato

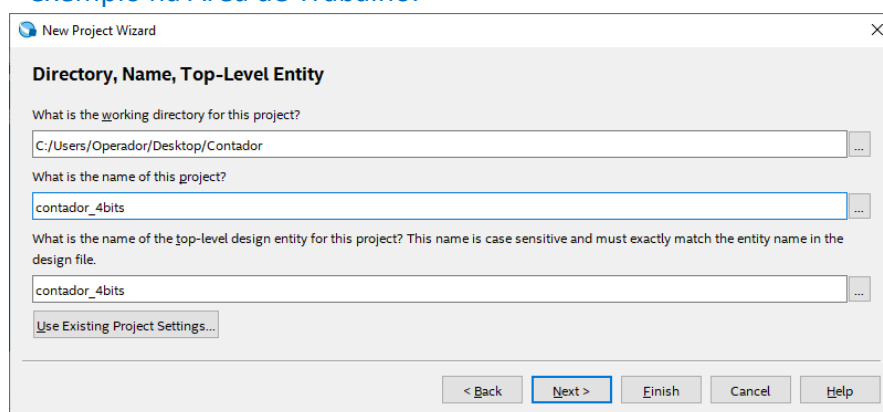
Prof. Dr. Edson Midorikawa

Esta apostila apresenta uma introdução à ferramenta Quartus Prime. Serão realizados os passos necessários para criação de um projeto, escrita e compilação do código em VHDL, simulação do código e programação na placa FPGA. Além disso, será apresentado como realizar a exportação e importação de um projeto na ferramenta. O procedimento apresentado foi feito no *Quartus Prime 20.1 Lite Edition*, porém pode ser replicado em outras versões do Quartus Prime.

## 1) Criação de um projeto no Quartus Prime

- Abra o Quartus e selecione no menu superior a opção **File -> New -> New Quartus Prime Project**.
- O Project Wizard será aberto. Siga o procedimento nas telas seguintes:
  - Na tela *Introduction*, pressione **Next** para seguir à próxima tela.
  - Na tela *Directory, Name, Top-Level Entity*, escolha um diretório e de o nome *contador\_4bits* para o projeto. Pressione **Next** para seguir à próxima tela.

OBS: O diretório padrão sugerido pelo Quartus é o "C:\intelFPGA\_lite\20.1". Não é recomendado utilizar esse diretório, pois para cada vez que ele for selecionado, serão misturados arquivos de projetos distintos. O ideal é criar um diretório para cada projeto, por exemplo na Área de Trabalho.



- Na tela *Project Type*, mantenha a opção *Empty project* selecionada e pressione **Next** para seguir à próxima tela.
- Na tela *Add Files*, existe a possibilidade de adicionar arquivos (vhdl, formas de onda, esquemáticos...) ao seu projeto, porém isso não será realizado. Pressione **Next** para seguir à próxima tela.

- Na tela *Family, Device & Board Settings*, selecione as seguintes opções para escolher a placa FPGA DE0-CV:
  - Family: Cyclone V
  - Device: All
  - Package: Any
  - Pin Count: 484
  - Core speed grade: 7

Selecione a terceira opção da tabela e pressione **Next** para seguir à próxima tela.

OBS: A FPGA selecionada (5CEBA4F23C7) é a FPGA presente na placa DE0-CV, e utilizada no Laboratório Digital.

- Na tela *EDA Tool Settings*, é possível selecionar outras ferramentas para auxiliar no seu projeto que não serão utilizadas aqui. Pressione **Next** para seguir à próxima tela.
- Na tela *Summary*, pressione **Finish** para finalizar a criação do projeto.

OBS: É possível apertar o botão **Finish** após a tela de configuração de diretório e do nome do projeto. Nesse caso, será necessário posteriormente atribuir a placa DE0-CV ao projeto.

## 2) Descrição de um sistema digital em VHDL

- Selecione no menu superior a opção **File -> New -> VHDL file** abrindo a janela para a edição do código em VHDL.
- Copie o código escrito na figura a seguir que descreve o comportamento de um contador de 4 bits.

```

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

entity contador_4bits is
    port (
        clock    : in  std_logic;
        clear    : in  std_logic;
        enable    : in  std_logic;
        Q        : out std_logic_vector(3 downto 0);
        rco      : out std_logic
    );
end contador_4bits;

architecture exemplo of contador_4bits is
    signal IQ: unsigned(3 downto 0);

begin
    process (clock, clear, enable)
    begin
        if clear = '1' then
            IQ <= "0000";
        elsif clock'event and clock = '1' then
            if enable = '1' then
                IQ <= IQ + 1;
            end if;
        end if;

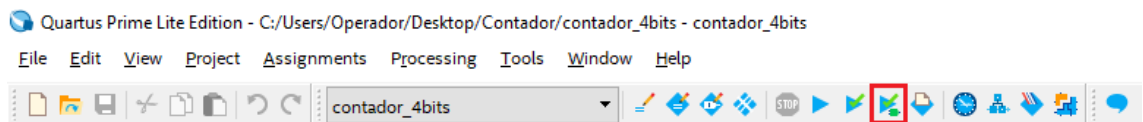
    end process;

    Q <= std_logic_vector(IQ);
    rco <= '1' when IQ = 15 else '0';
end exemplo;

```

- Em seguida, pressione o botão destacado na figura a seguir para compilar o projeto. Será apresentada uma mensagem para salvar o arquivo. Ele deve ser nomeado com o mesmo nome da entidade descrita (*contador\_4bits* neste caso), que é o mesmo nome do projeto. Realize a compilação.

OBS: O botão destacado corresponde a “compilação rápida” do projeto. Seu propósito é verificar se existe algum erro de compilação na descrição do circuito.



- A compilação será realizada com sucesso e a seguinte mensagem será apresentada:

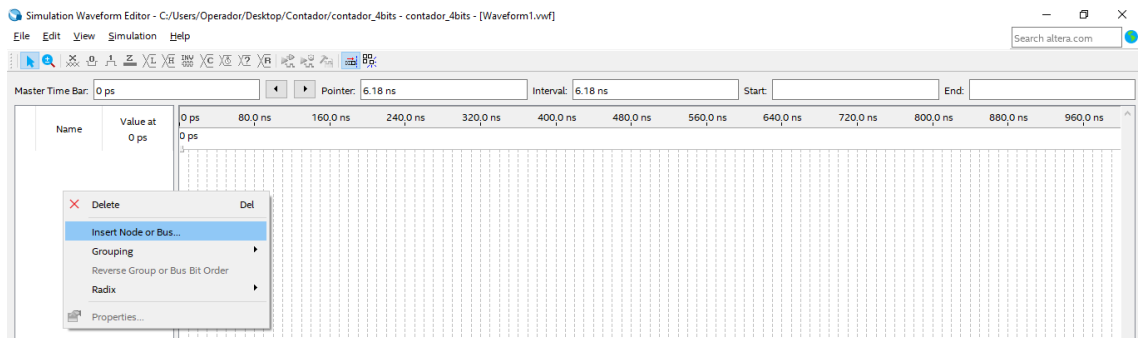
▶  Quartus Prime Analysis & Synthesis was successful. 0 errors, 2 warnings

### 3) Simulação do sistema digital

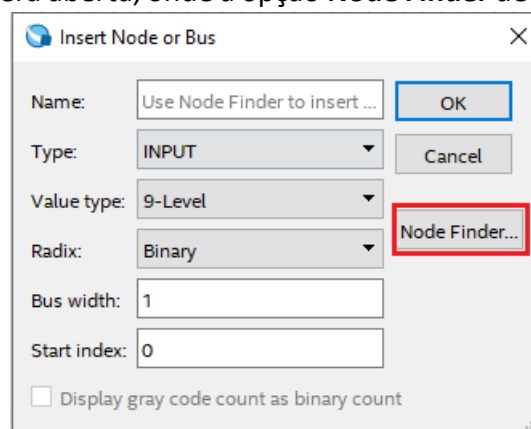
OBS: Considerando um exemplo real (fora do escopo da apostila), caso o circuito já tenha sido validado por meio de um *testbench* ou outro simulador, esta etapa não é necessária. Caso você esteja utilizando uma versão mais nova do Quartus Prime, talvez seja necessário baixar uma licença para utilizar o simulador (Questa).

- Selecione no menu superior opção **File -> New -> University Program VWF** para abrir a janela de simulação com formas de onda.

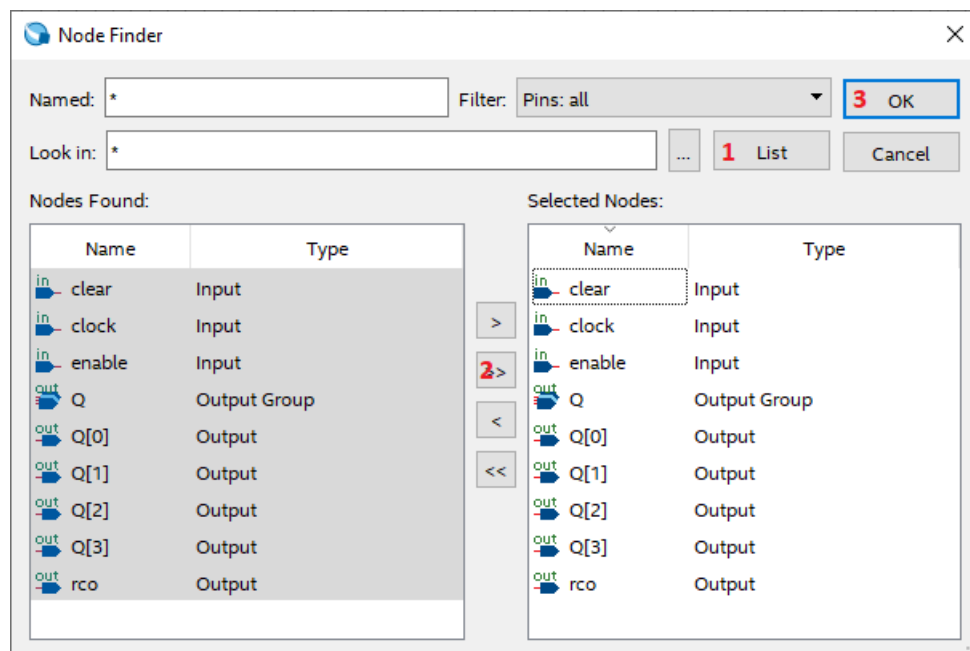
- Com um click do botão direito do mouse no canto extremo esquerdo (espaço branco abaixo de *Name*), selecione **Insert Node or Bus**.





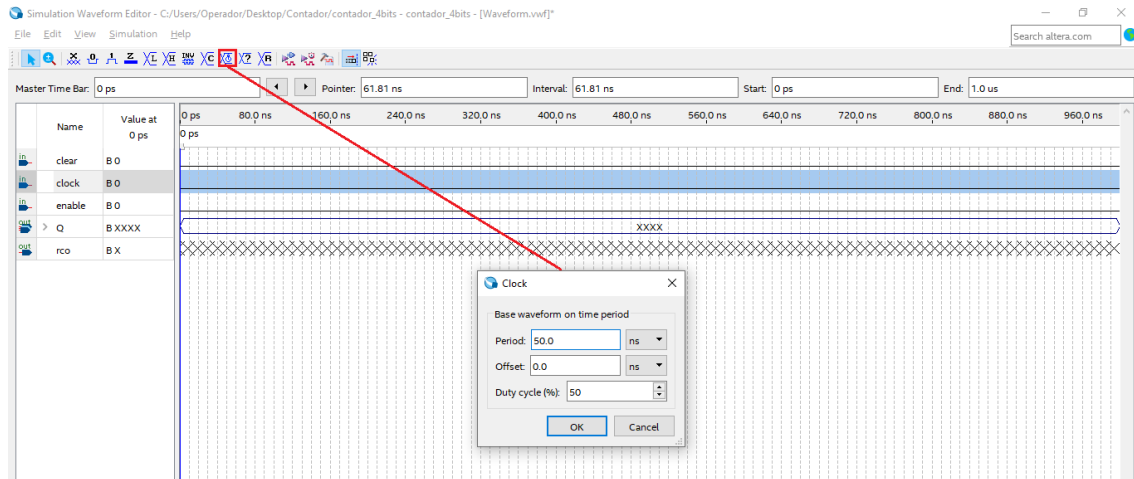
- A seguinte janela será aberta, onde a opção **Node Finder** deverá ser selecionada.





- Selecione a opção **List** para listar as entradas e saídas da sua entidade como nós. Em seguida, pressione >> para selecionar todas os nós encontrados. Aperte **OK** em seguida.



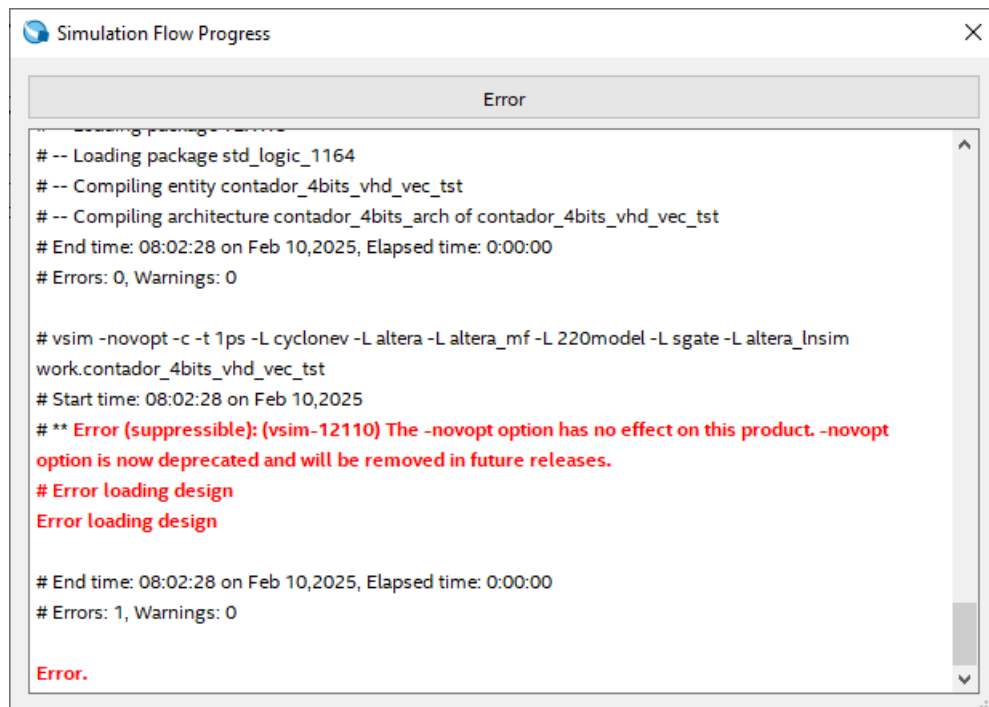
- Observe que os nós foram incorporados à simulação, sendo necessário transmitir valores aos nós de entrada. Para o *clock*, clique nele e selecione o botão **Overwrite Clock** . Em seguida escolha um período adequado para visualizar as mudanças no contador. Será utilizado aqui um período de 50ns. Você pode utilizar o botão de zoom  para aumentar ou diminuir o zoom das formas de onda com o botão esquerdo ou o direito respectivamente.



- Clique no nó *clear* e deixe em ativo baixo com o botão **Forcing Low** . Em seguida clique no nó *enable* e deixe em ativo alto com o botão **Forcing High** . Aperte **Ctrl + S** para salvar o arquivo de simulação ou selecione **File -> Save**. É recomendado salvar com o nome padrão sugerido para evitar conflitos (*Waveform.vwf*).
- Antes de realizar a simulação, é necessário alterar o script utilizado, pois ele apresenta uma *flag* que foi depreciada (*-novopt*). Selecione **Simulation -> Simulation Settings** e apague a flag no espaço grifado na figura a seguir. Pressione **Save**.



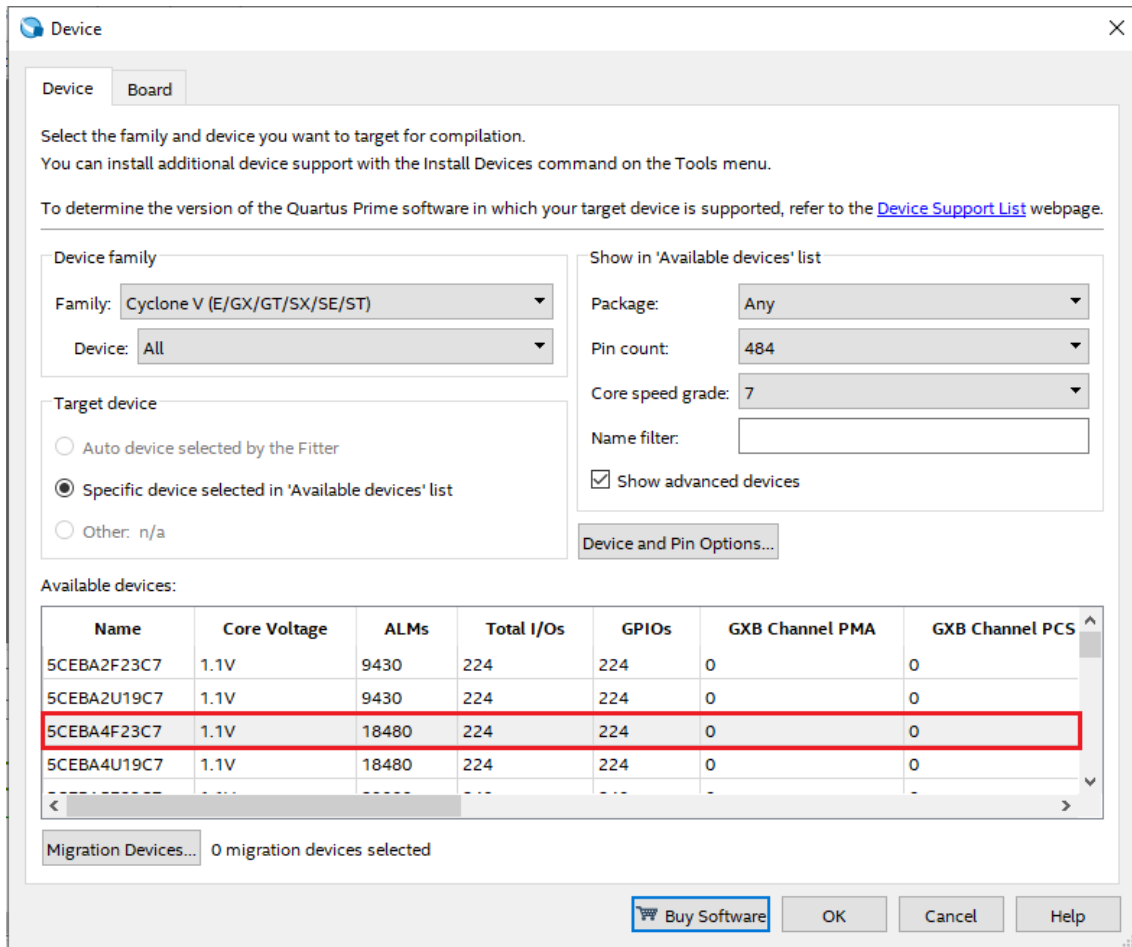
OBS: Caso o script não seja alterado (removendo a flag mencionada), o seguinte erro será obtido durante a simulação:



#### 4) Programando o projeto na placa FPGA DE0-CV

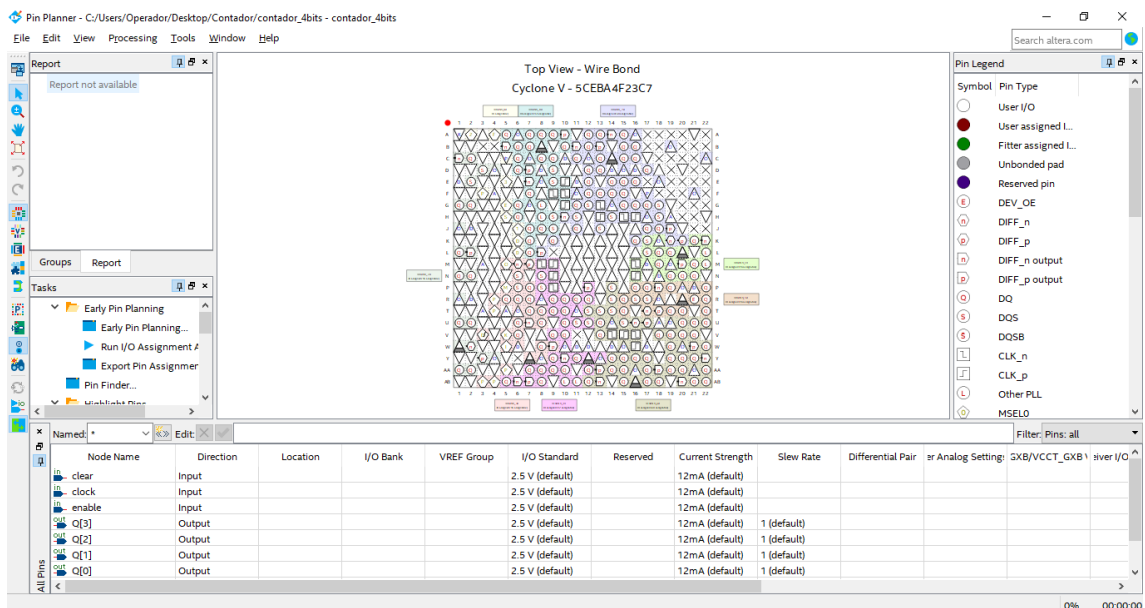
- Para programar o projeto na placa DE0-CV, caso a designação do *device* não tenha sido realizada durante sua criação, é necessário primeiro selecionar a FPGA. É possível também alterar o *device* já designado. Selecione no menu superior a opção **Assignments -> Device**. Escolha as seguintes opções:
  - Family: Cyclone V
  - Device: All
  - Package: Any
  - Pin Count: 484
  - Core speed grade: 7

Selecione a terceira opção da tabela e pressione **OK**.



- Em seguida é necessário definir a pinagem da placa. Para isso selecione no menu superior a opção **Assignments -> Pin Planner**. A seguinte janela será aberta onde os nós referentes ao contador de 4 bits estarão localizados em uma tabela no canto inferior da janela.

OBS: Caso a tabela no canto inferior esteja vazia, realize a “compilação rápida” do projeto conforme apresentado na seção 2 desta apostila.

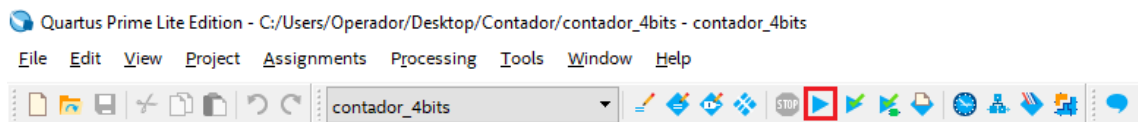




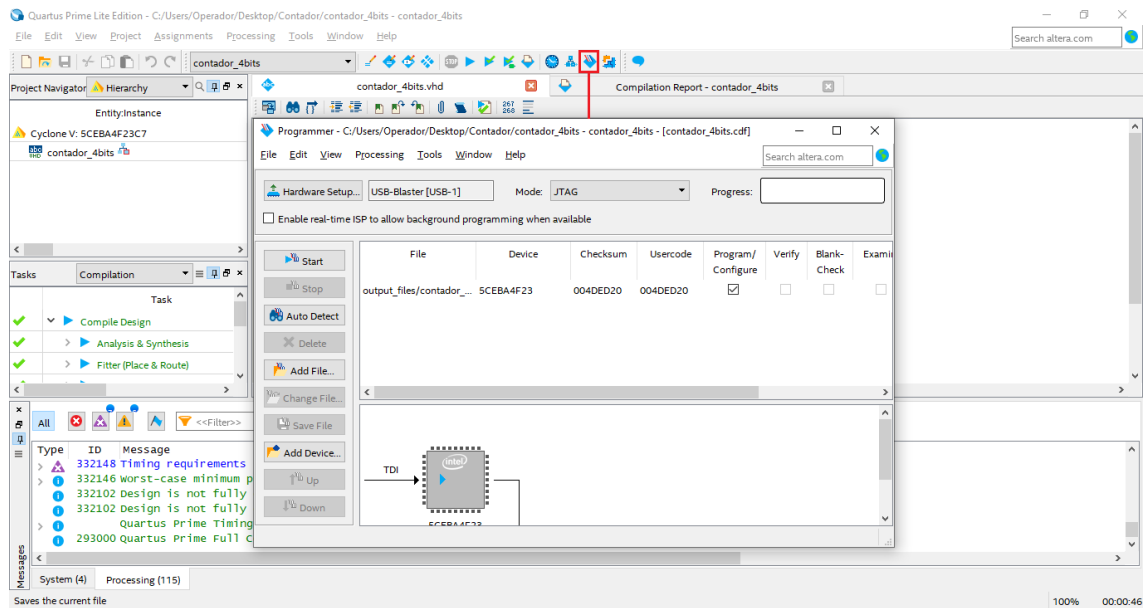
- Na coluna *Location* da tabela, selecione uma pinagem adequada da placa DE0-CV (chaves, botões, leds...) correspondente para cada nó do sistema. Para isso, utilize o arquivo contendo as informações da pinagem da placa localizado no site da disciplina. Um exemplo de pinagem para este projeto seria:

Entrada/Saída	Periférico	Pino
clock	Botão KEY0	PIN_U7
clear	Chave SW0	PIN_U13
enable	Chave SW1	PIN_V13
Q[3]	Led LEDR3	PIN_Y3
Q[2]	Led LEDR2	PIN_W2
Q[1]	Led LEDR1	PIN_AA1
Q[0]	Led LEDR0	PIN_AA2
rco	Led LEDR9	PIN_L1

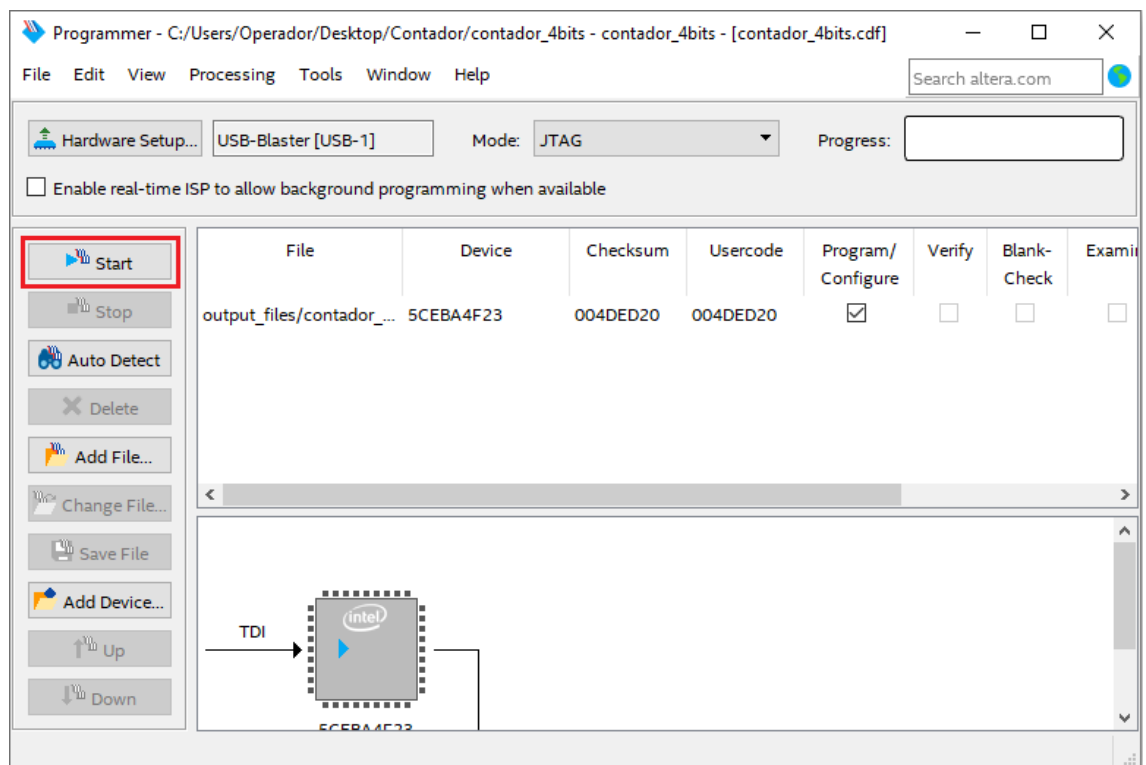
- Realize a compilação completa do projeto.  
OBS: Ao contrário da “compilação rápida” realizada anteriormente, esta compilação é mais demorada, e tem como propósito sintetizar o arquivo utilizado para programar a placa FPGA. Este é o arquivo sof.



- Pressione o botão **Programmer** para abrir uma nova janela. Em seguida, verifique se a opção *Hardware Setup* está em *USB-Blaster*. Em caso contrário, verifique se a placa está ligada e conectada ao computador e clique no botão **Hardware Setup** para adicioná-la.  
OBS: Caso você esteja utilizando um dispositivo chamado *Analog Discovery*, existe um conflito entre o uso dele e a carga no arquivo programável na placa FPGA. Este conflito faz com que não seja possível identificar o *USB-Blaster* dentro da janela *Programmer*. Para resolver o conflito, é necessário fechar o software *Waveforms* utilizado pelo *Analog Discovery*. Se você ainda não souber o que é o *Analog Discovery*, desconsidere neste momento esta mensagem.



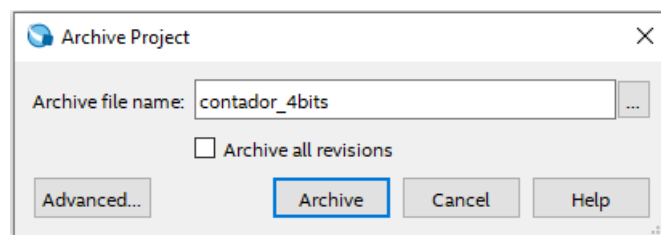
- Geralmente versões mais novas do Quartus como a versão 20.1 utilizada nesta apostila irão reconhecer automaticamente o caminho para o arquivo .sof (arquivo programável da placa FPGA), conforme imagem anterior. Caso isso não ocorra (o campo "File" não apresentará nenhum arquivo), selecione **Add Files**, vá para o diretório *output\_files* da pasta do seu projeto e selecione o arquivo *contador\_4bits.sof*.
- Pressione **Start** para programar o projeto na placa FPGA.



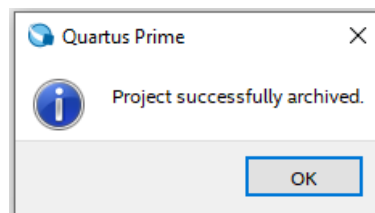
## 5) Exportando o projeto criado no Quartus

Uma maneira simples de exportar o projeto criado no Quartus é zipar a pasta onde o projeto foi criado. Este método irá servir na maioria dos casos, porém, em algumas situações o tamanho da pasta zipada pode ser um limitante. A forma de contornar essa limitação é utilizar o recurso de exportação presente no Quartus, que irá gerar um arquivo QAR. Este arquivo possui semelhanças a um zip, porém ele é específico de projeto no Quartus, e apenas armazena as informações fundamentais, como os arquivos presentes no projeto, hierarquia, pinagem dentre outros. Esta seção apresenta brevemente como exportar o projeto criado em um arquivo qar.

- No menu superior do Quartus, selecione a opção **Project -> Archive Project**, abrindo a seguinte janela.

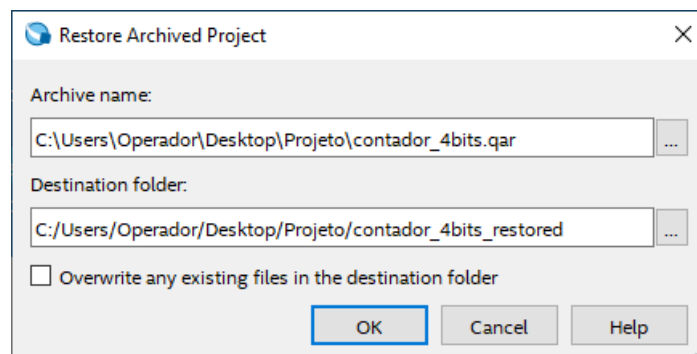


- Clique no botão **Archive** para gerar o qar, fazendo com que uma janela de sucesso seja apresentada em sequência. O arquivo qar estará localizado na pasta do projeto.

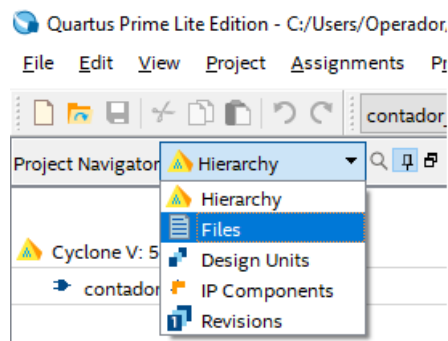


## 6) Importando um qar no Quartus

- Abra o Quartus e selecione no menu superior a opção **File -> Open**. Encontre o diretório onde o arquivo qar está localizado e coloque no canto inferior direito de filtragem de arquivos a opção de arquivo **Quartus Prime Archive Files** para visualizá-lo. Selecione o arquivo e pressione **Abrir**, gerando a seguinte janela.



- Aperte **OK** para abrir o qar. Após o término do processo, observe que no *Project Navigator*, localizado no canto esquerdo é possível selecionar a opção **Files** para visualizar todos os arquivos contidos no projeto.



- Para realizar uma simulação de um arquivo contido no qar, é necessário realizar algumas modificações no script, alterando o diretório utilizado por ele. Esse problema ocorre, pois, o qar salva no script da simulação o diretório original onde foi criado o projeto, antes de ser gerado. Abra o arquivo da simulação, selecione **Simulation -> Simulation Settings** e realize as modificações nos espaços grifados na figura abaixo, trocando o caminho do diretório original pelo caminho do seu diretório. Pressione **Save** e simule o arquivo.

