



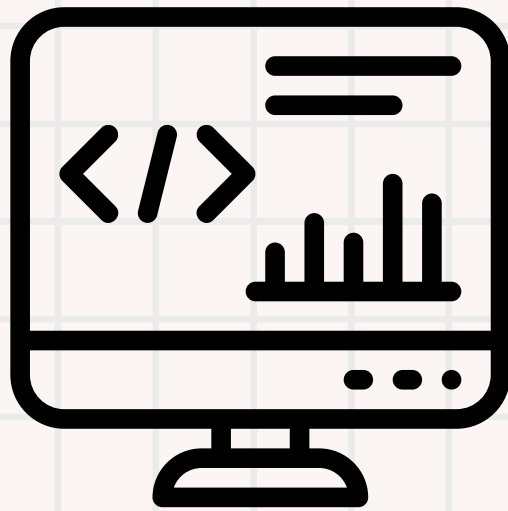
Tutorial

Turbo Encoder

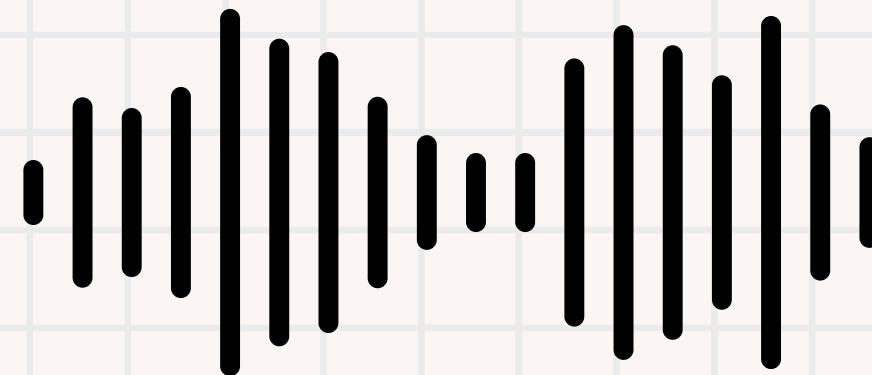
Laboratórios Didáticos para Ensino de Sistemas de
Comunicação

Materiais

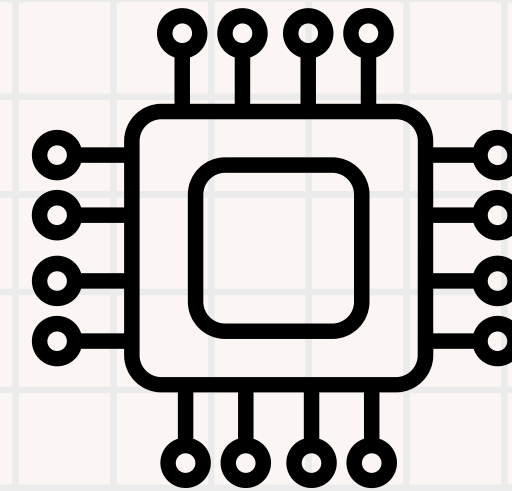
Pré-requisitos



QUARTUS instalado



**Conceito rápido de
modulação OFDM**



FPGA DE2-115

Introdução

Quartus

○ sistema é composto por três blocos principais:

Codificador RSC 1

Interleaver

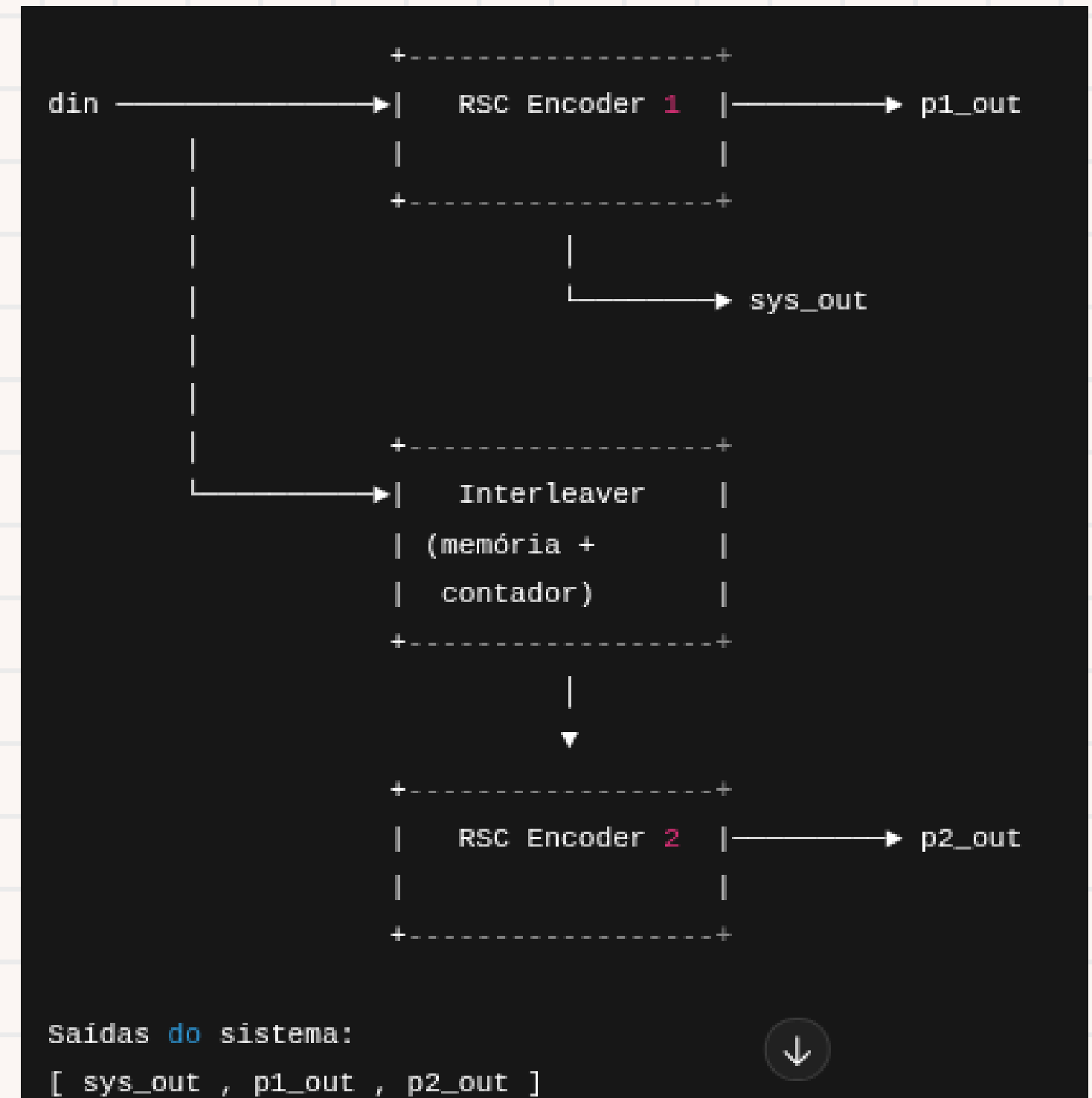
Codificador RSC 2

○ fluxo de dados ocorre em dois caminhos paralelos:

Caminho direto, sem permutação

Caminho entrelaçado, com permutação dos bits

As saídas desses blocos formam o código Turbo



Desenvolvimento

Passo a Passo





Desenvolvimento

Simulink

Arquivo do projeto

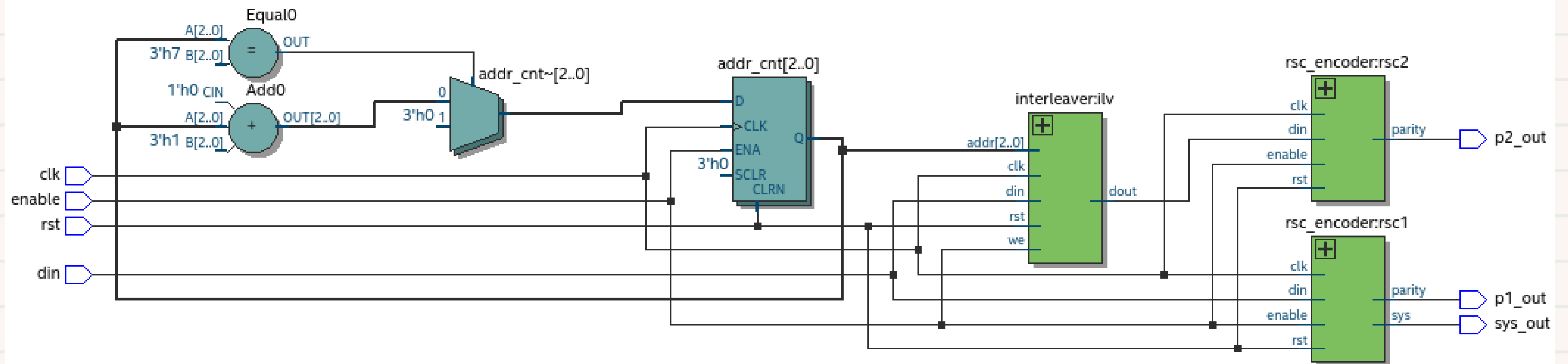
Para facilitar, é disponibilizado arquivo do modelo em Simulink no Github:



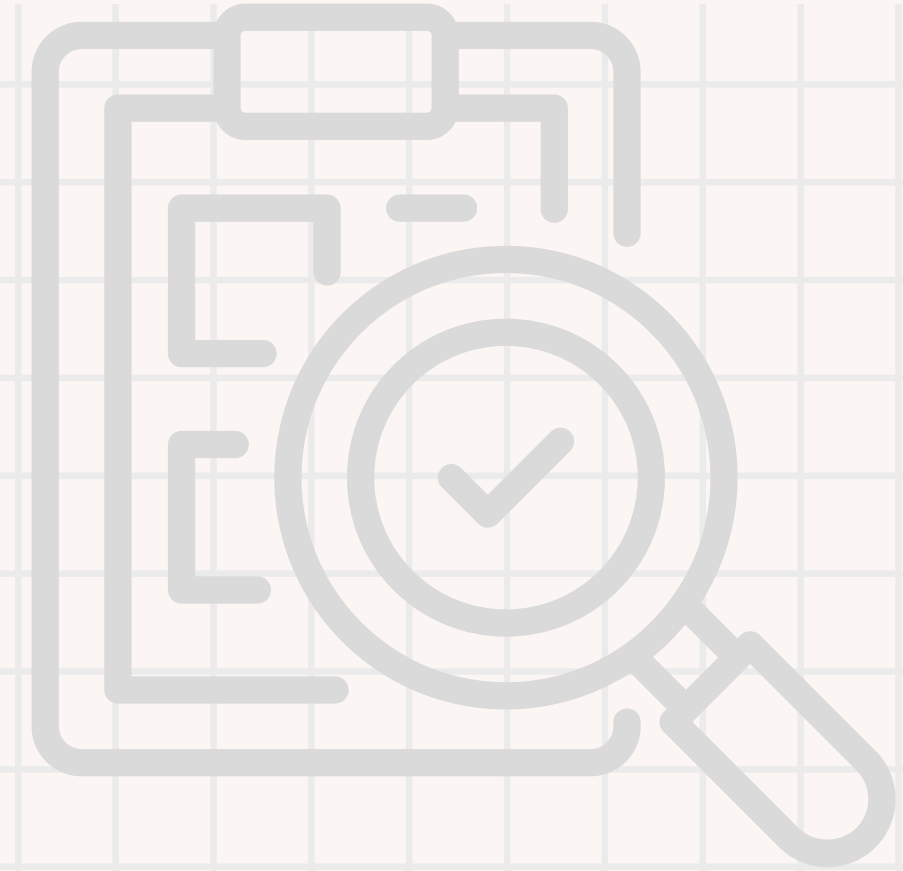
[Link Download](#)

Desenvolvimento

Turbo Encoder

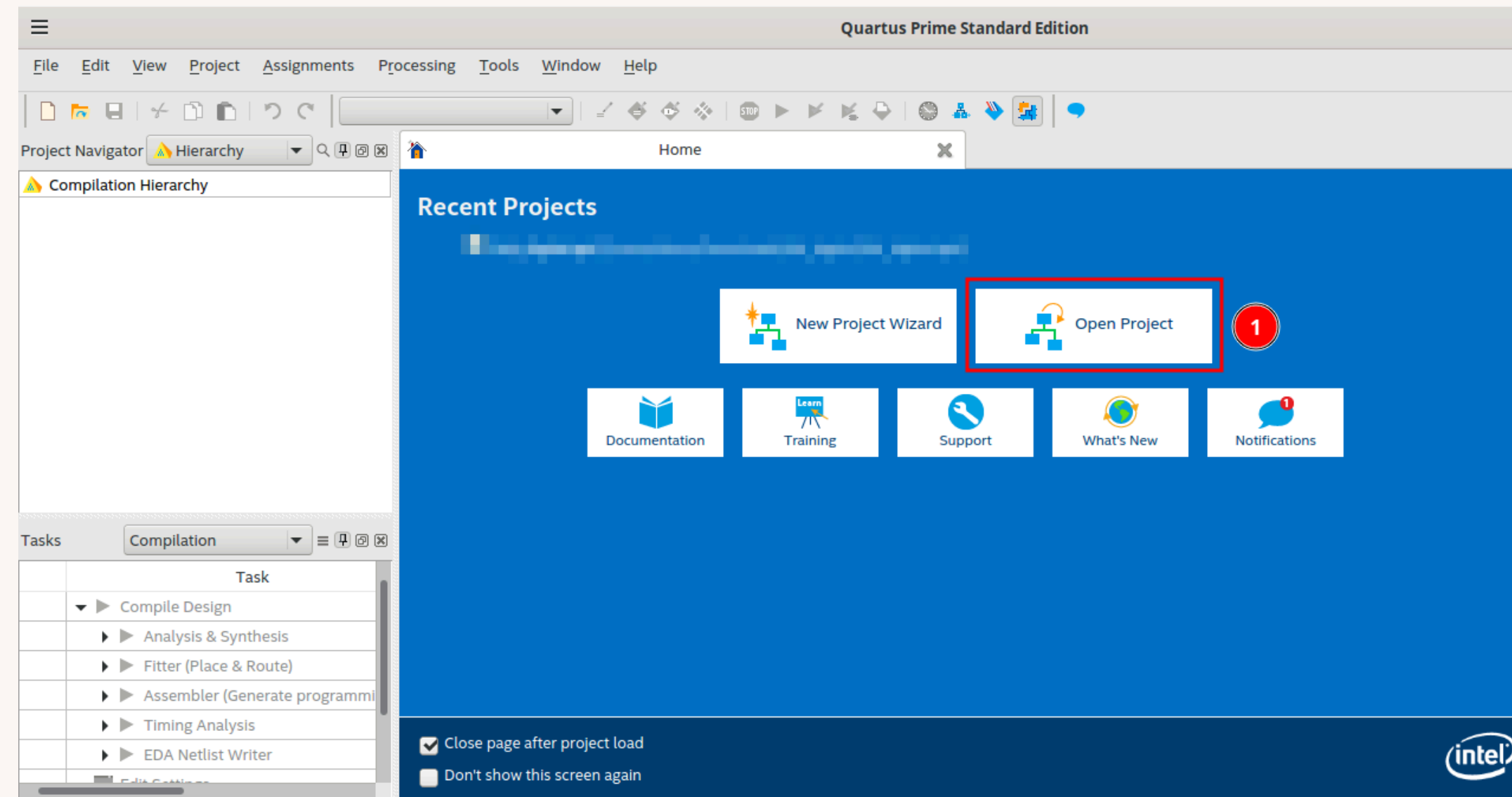


Desenvolvimento



A seguir a demonstração de como usar o QUARTUS...

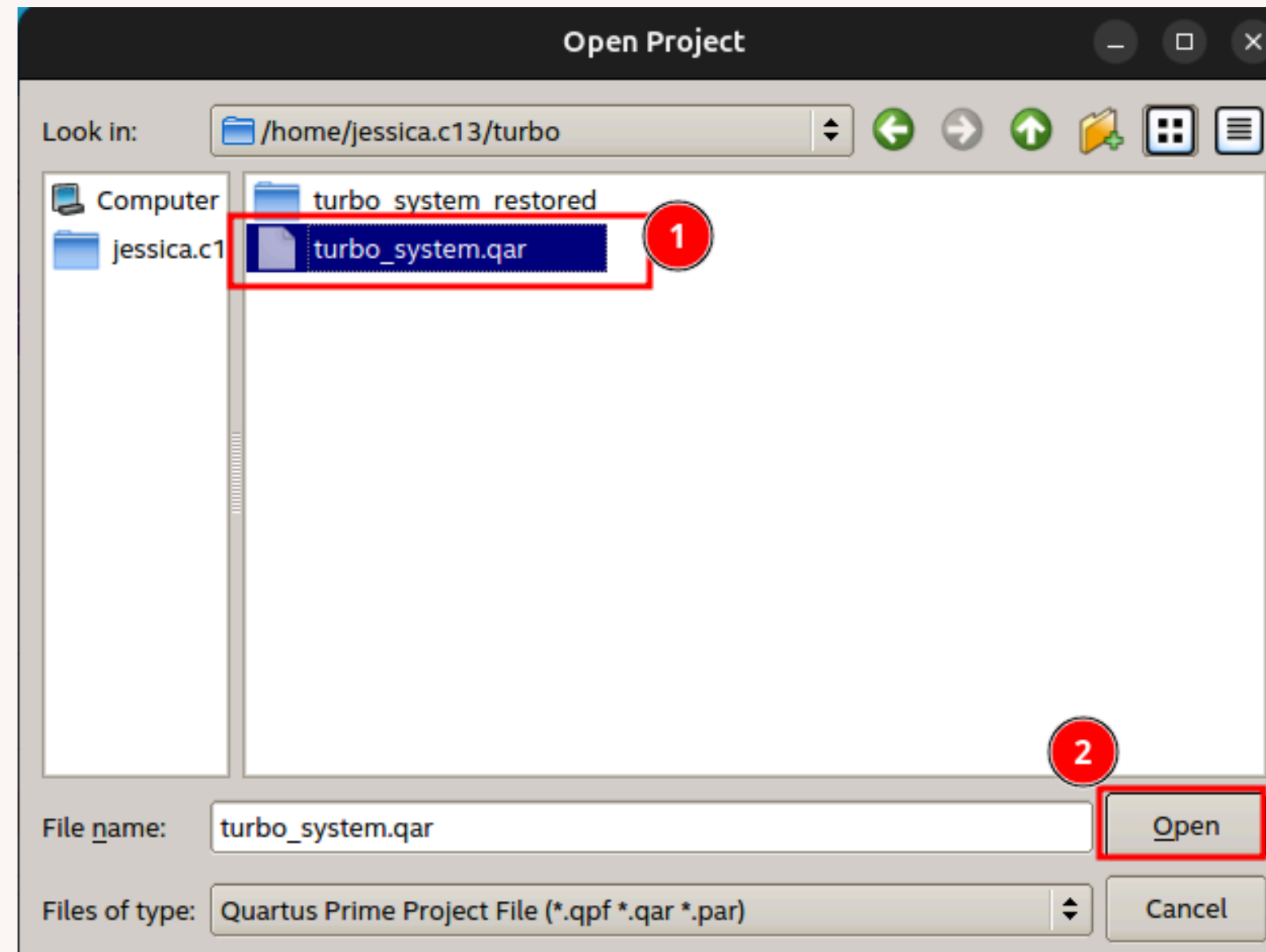
Desenvolvimento



Abra o Quartus

1 - Clique em **Open Project**

Desenvolvimento

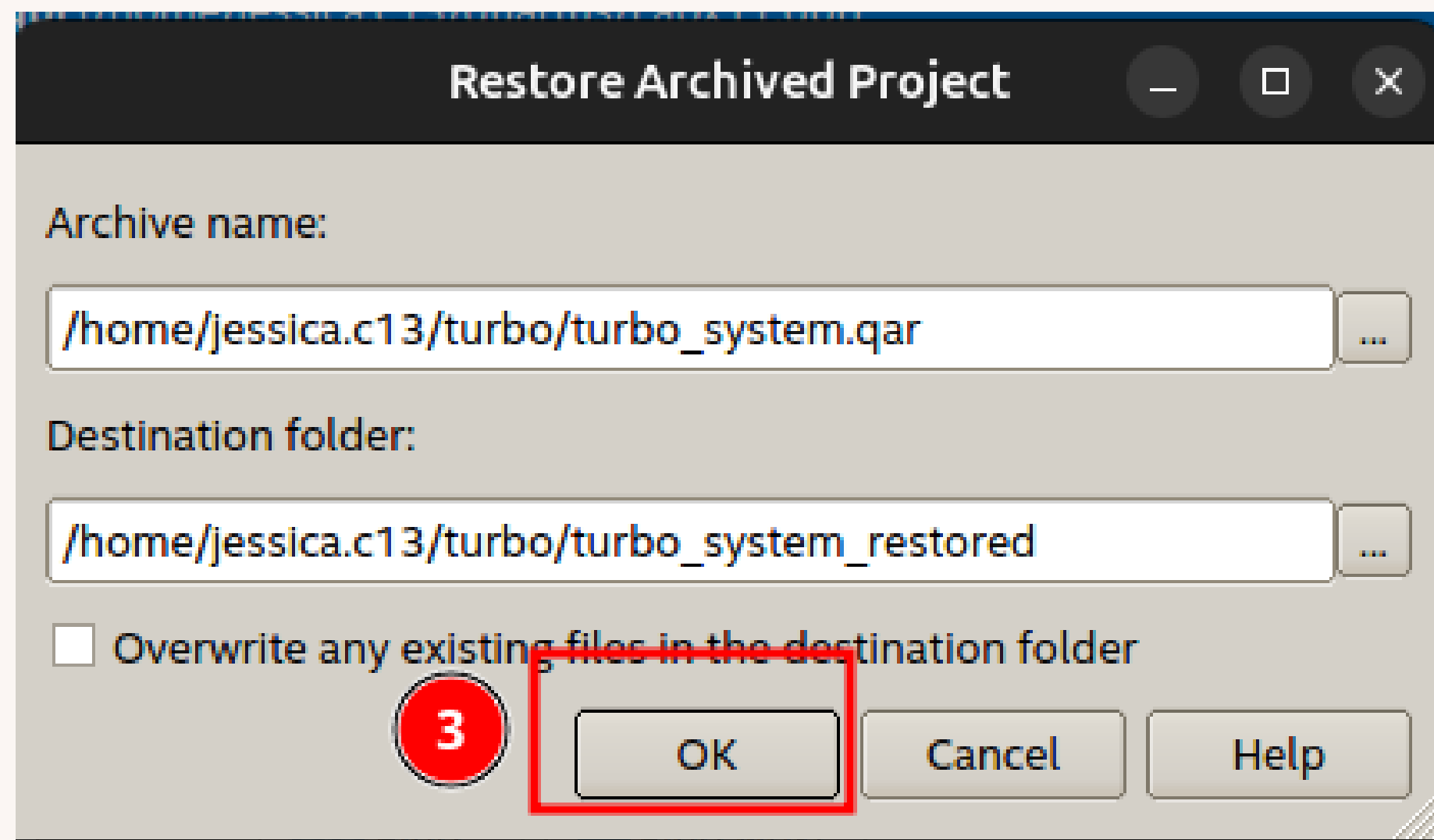


Vá até ao diretório onde foi salvado o arquivo turbo_system.qar

2 - Clique no arquivo

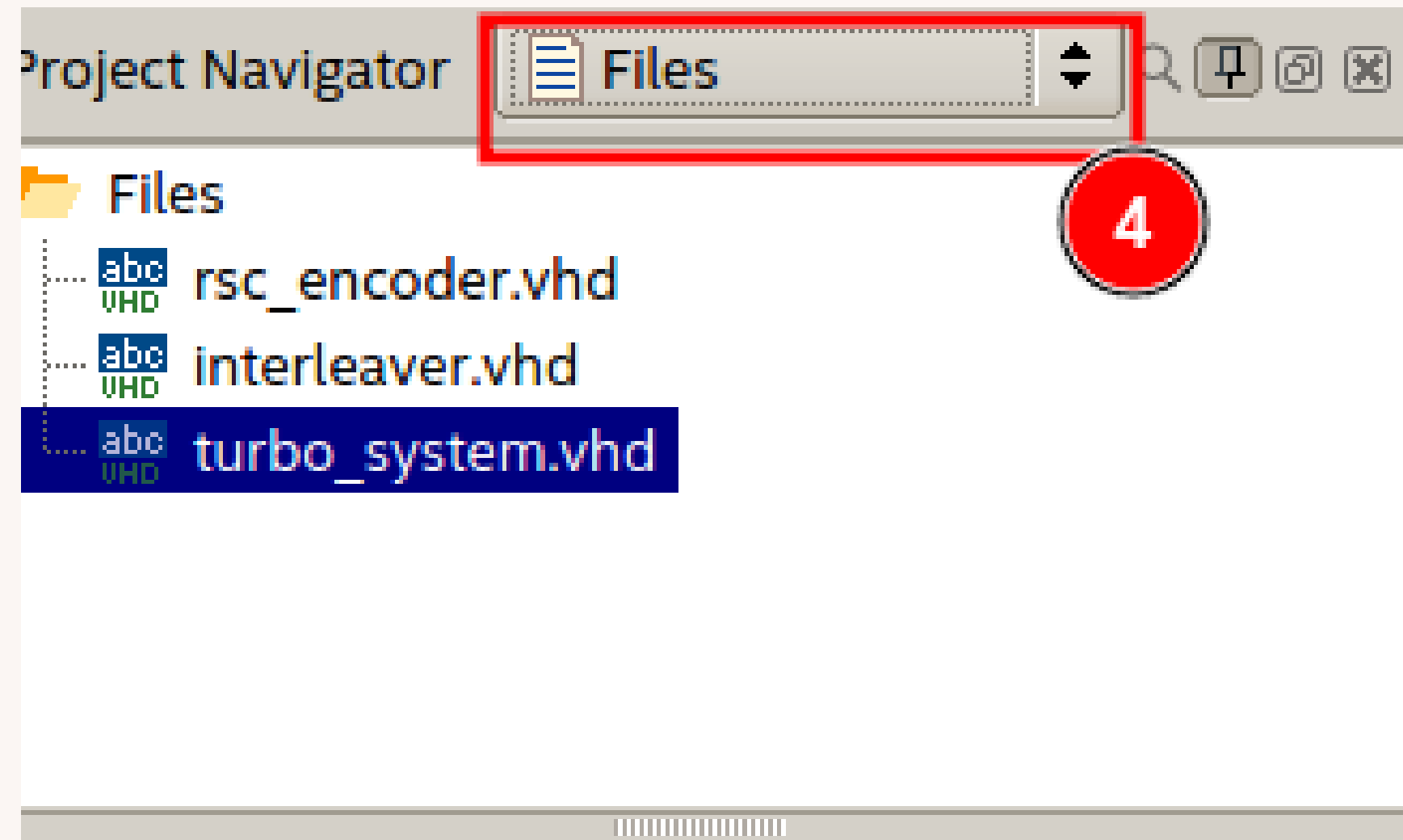
3 - Clique em Open para abri-lo

Desenvolvimento



3 - Clique em Ok
E arquivo será aberto.

Desenvolvimento



Vá em Project Navigator

4 - Selecione **Files** Com isso, você consegue visualizar todos os códigos



Desenvolvimento

Entidade turbo_system

A entidade **turbo_system** representa o bloco de mais alto nível do Turbo Encoder implementado em VHDL.

Função Principal

- Receber o bit de entrada din
- Controlar a codificação em paralelo
- Coordenar RSC 1, interleaver e RSC 2
- Entregar as saídas sincronizadas

Entradas

- clk: clock do sistema
- rst: reset
- enable: habilita a codificação
- din: bit de dados de entrada

Saídas

sys_out: bit sistemático
p1_out: paridade do RSC 1
p2_out: paridade do RSC 2

Configuração de código Turbo com taxa 1/3



Desenvolvimento

Entidade `rsc_encoder`

A entidade **`rsc_encoder`** implementa um codificador convolucional recursivo e sistemático, constituindo o bloco básico do Turbo Encoder, baseado em registradores de deslocamento e realimentação.

Função Principal

- Utiliza dois registradores internos
- A realimentação é definida por operações XOR
- Polinômios utilizados:
- Feedback: 7
- Paridade: 5

Entradas

- `clk`: clock do sistema
- `rst`: reset
- `enable`: habilita a codificação
- `din`: bit de dados de entrada

Saídas

- `sys`: bit sistemático (cópia direta de `din`)
- `parity`: bit de paridade gerado pelo codificador



Entidade interleaver

Desenvolvimento

A entidade interleaver reorganiza a sequência de bits por meio de uma memória endereçada, escrevendo e lendo dados sincronizados ao clock, recebendo din e fornecendo dout conforme o endereço selecionado.

Função Principal

- Reorganizar a sequência de bits de entrada
- Introduzir permutação temporal dos dados
- Fornecer bits entrelaçados ao segundo codificador

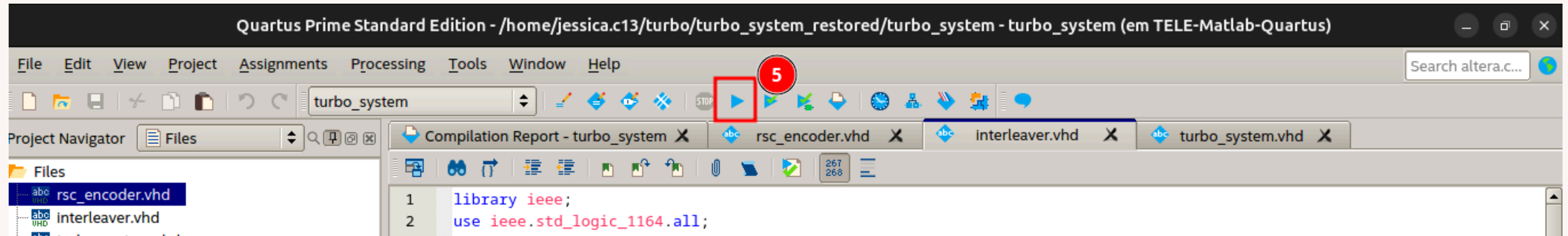
Entradas

- clk: clock do sistema
- rst: reset
- we: habilita escrita na memória
- addr: endereço de acesso à memória
- din: bit de dados de entrada

Saídas

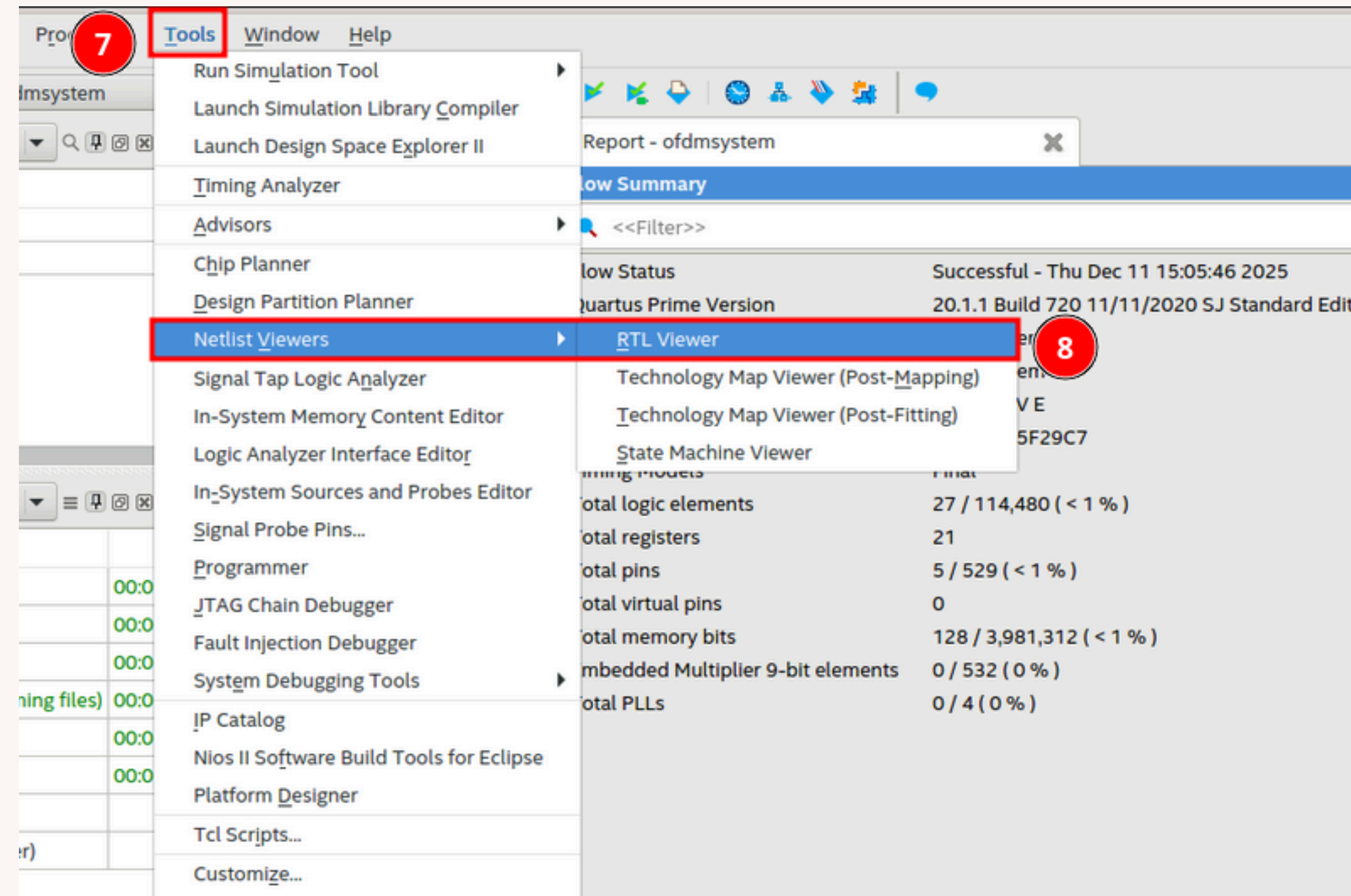
dout: bit armazenado no endereço selecionado

Desenvolvimento



5 - Clique no botão RUN para realizar a compilação completa do projeto

Desenvolvimento

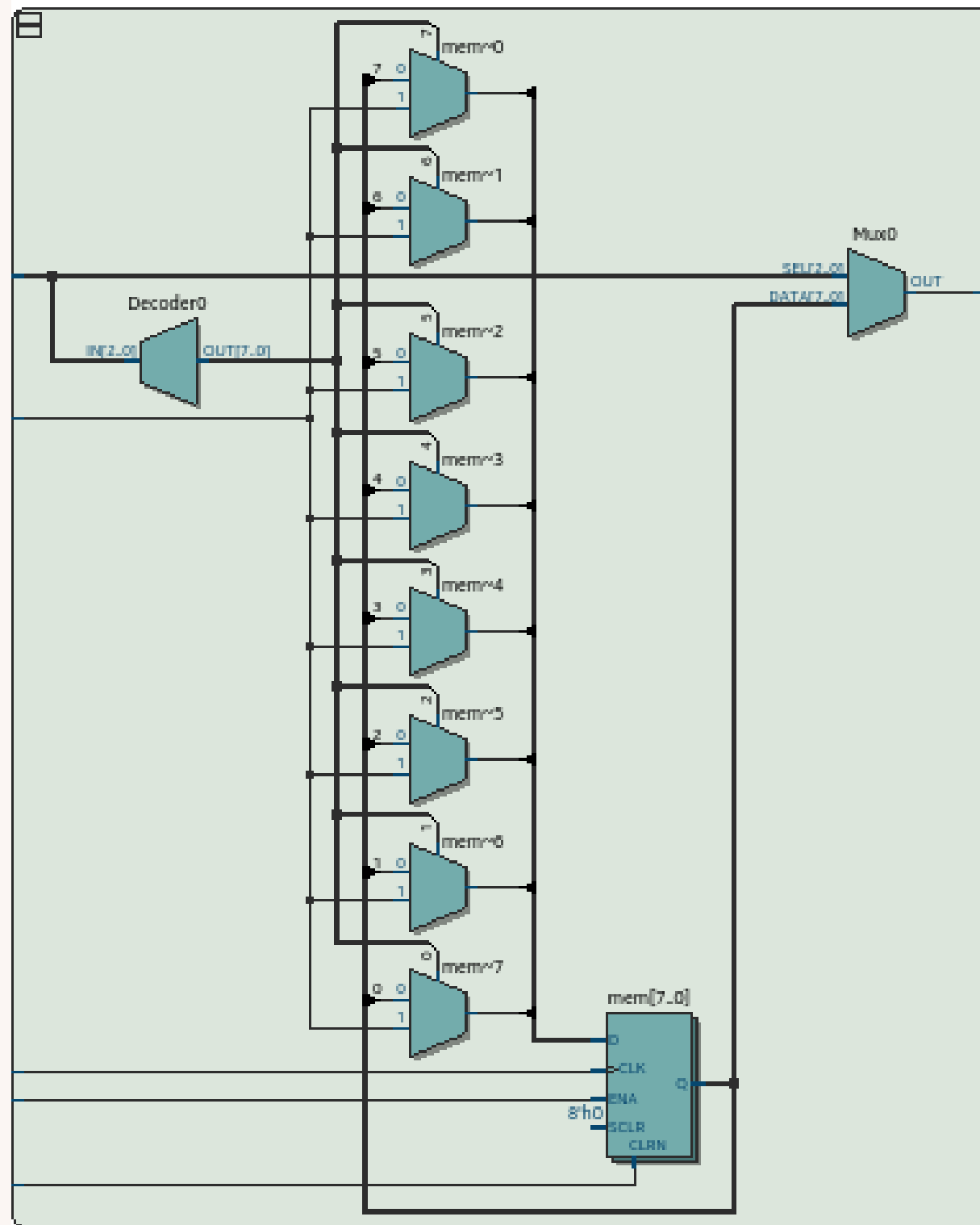


7 - Vá em Tools

8 - Cliente em: Netlist Viewers → RTL Viewer para visualização do sistema

Desenvolvimento

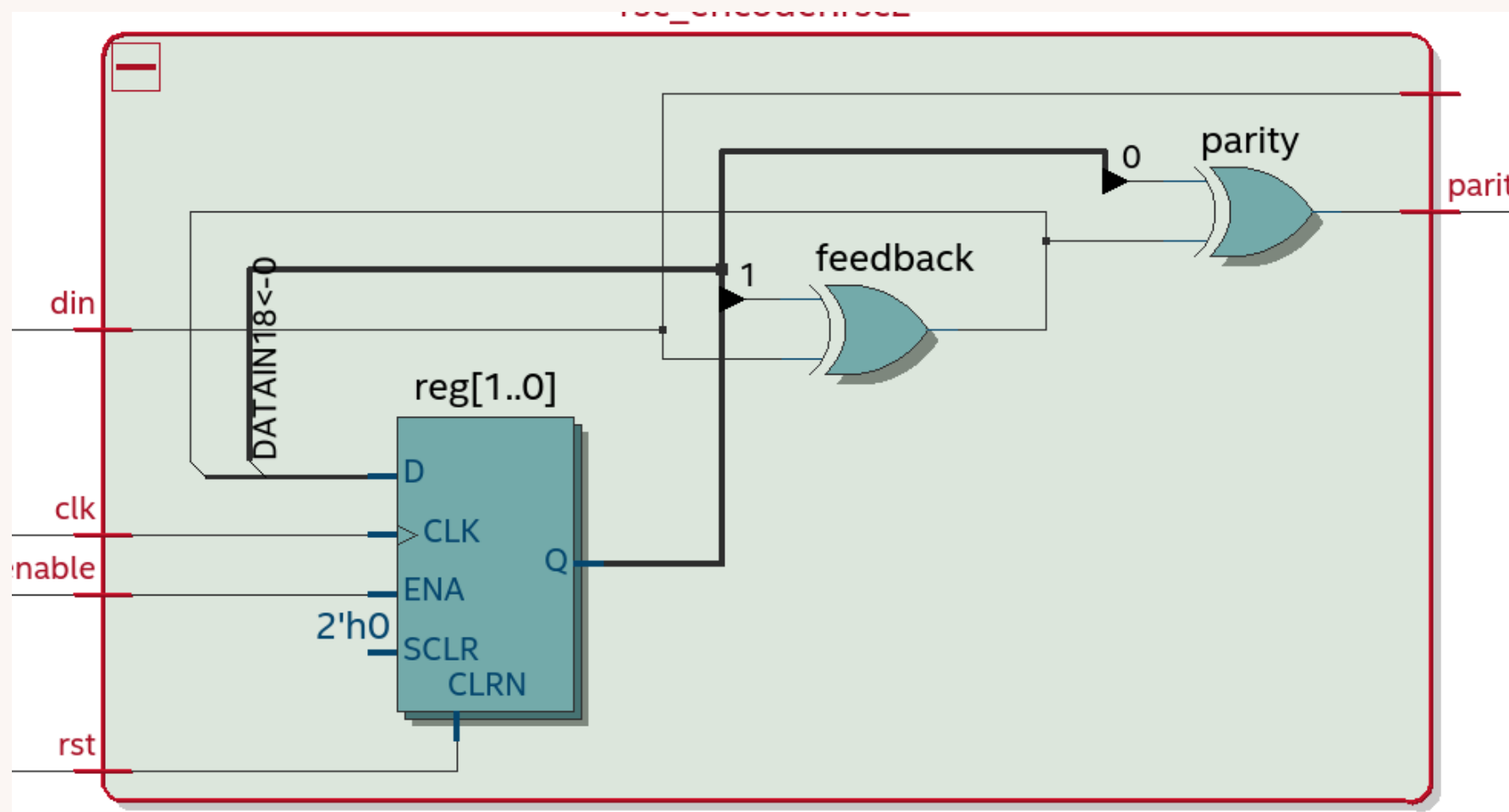
Após dar duplo clique no interleave



- Memória com múltiplas posições de 1 bit
- Endereço seleciona escrita e leitura
- Decodificador ativa a célula correta
- MUX envia o bit selecionado para a saída

Desenvolvimento

Após dar duplo clique no RSC



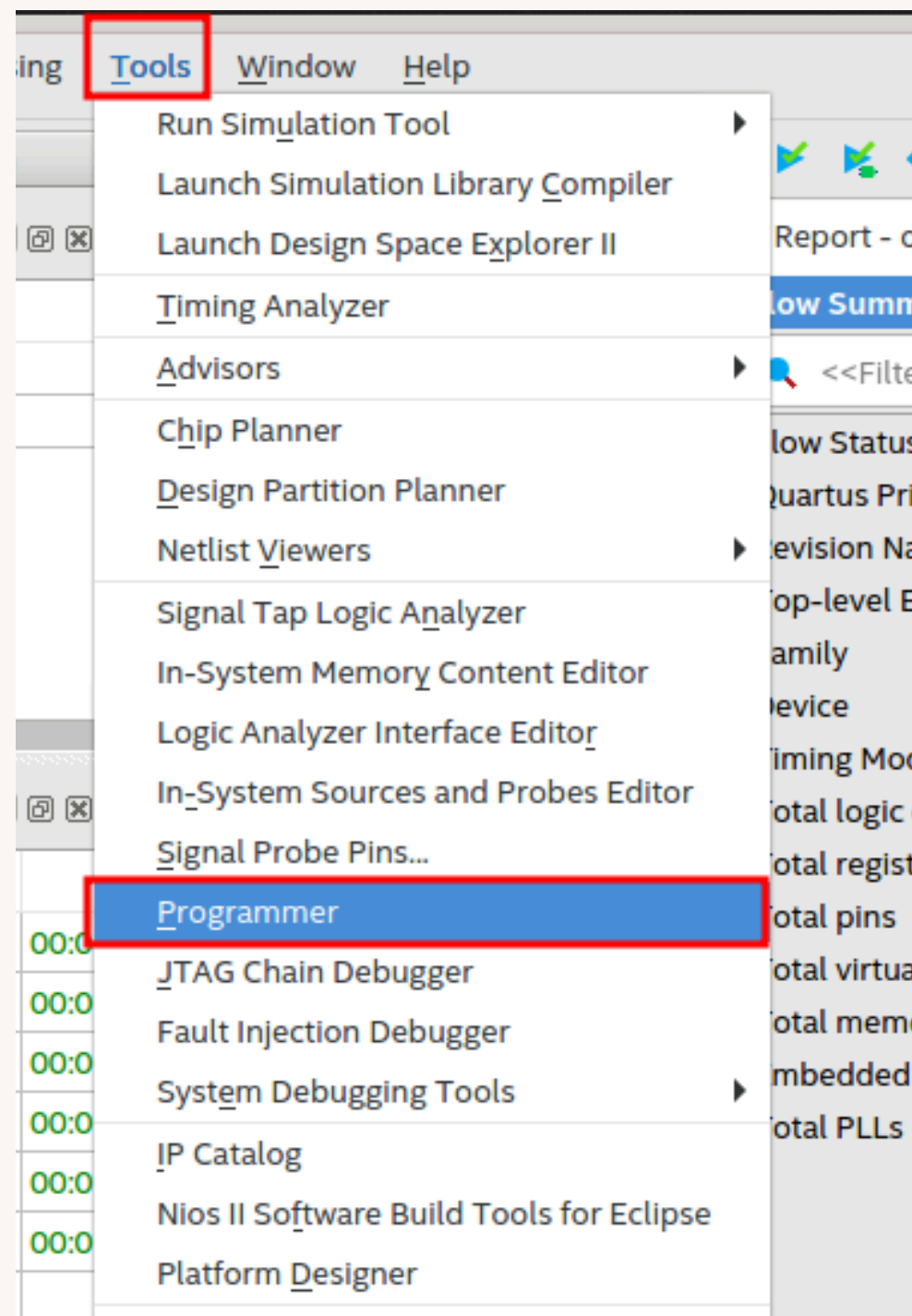
- Registradores de deslocamento armazenam o estado interno
- Realimentação é gerada por operação XOR
- Bit sistemático é a própria entrada
- Paridade é calculada a partir do estado e do feedback

Para transferir o seu projeto de modulador OFDM do software Quartus para a placa DE2-115, siga este roteiro. Este processo é chamado de programação ou gravação da FPGA:

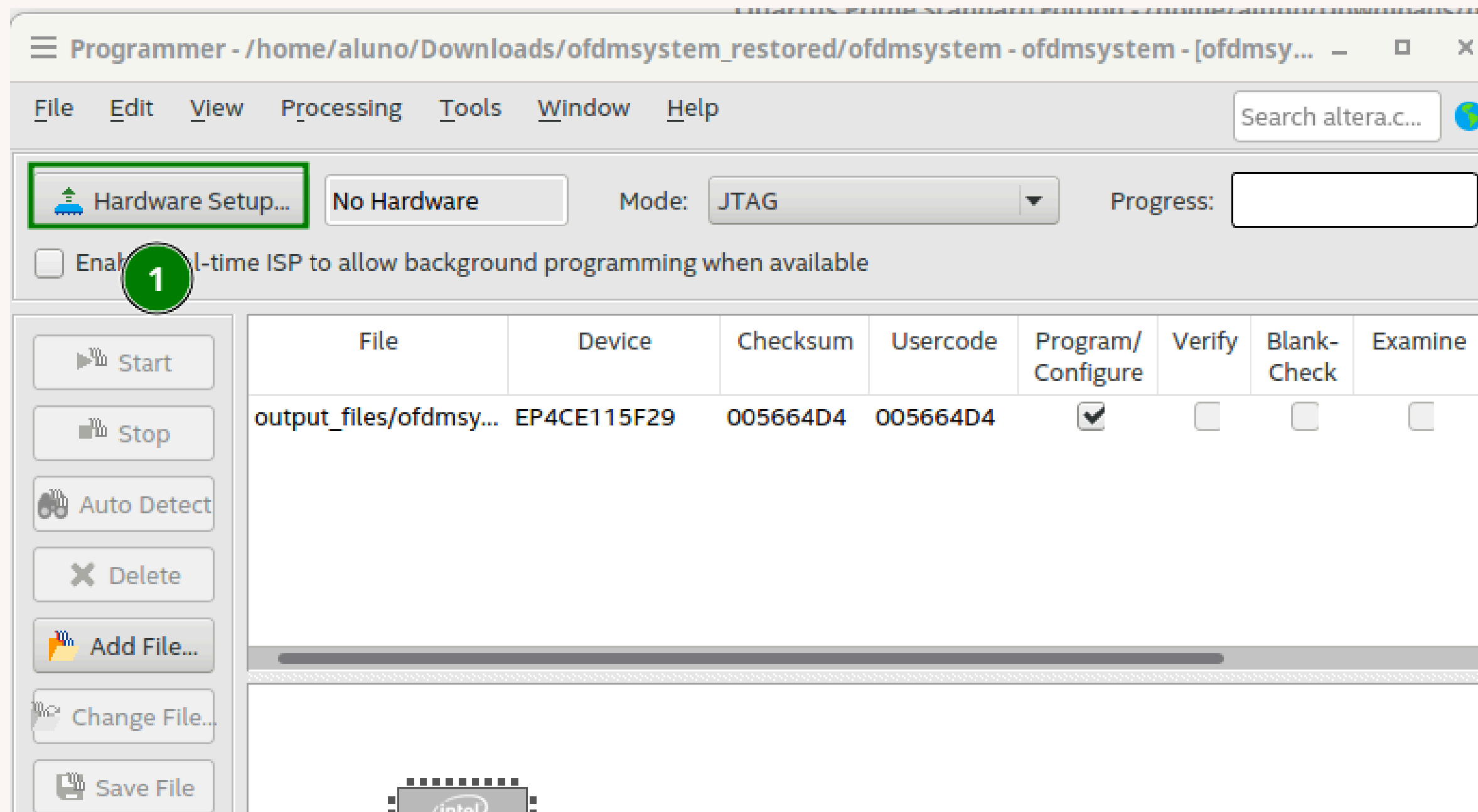
1. **Conexão USB:** Conecte o cabo USB da porta **USB BLASTER** da placa DE2-115 ao seu computador.
2. **Alimentação:** Ligue a fonte de energia da placa e acione o botão de Power.

Desenvolvimento

Gravar Código na Placa



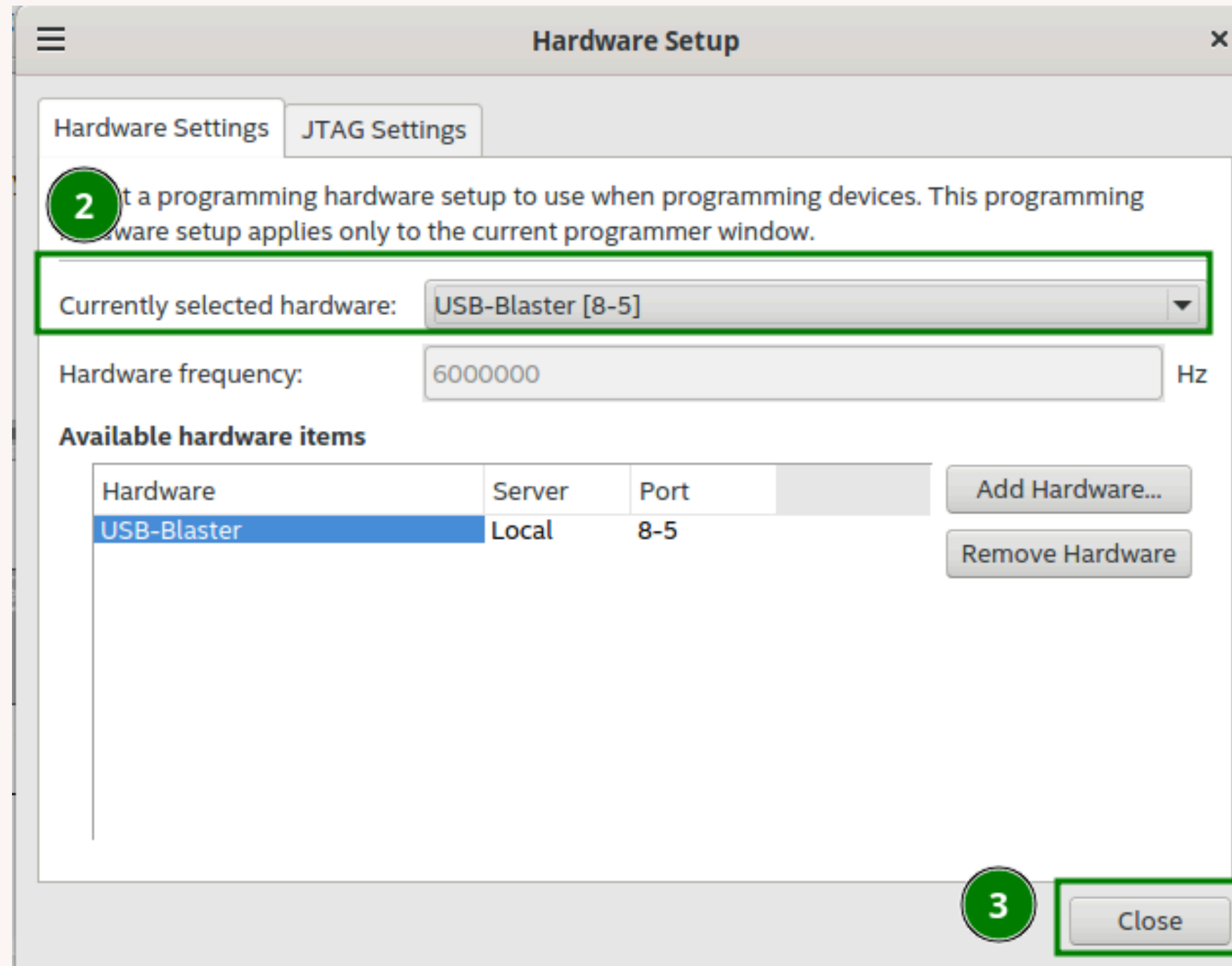
- Vá em **Tools** → **Programmer**



1 - Clique em **Hardware Setup**

Desenvolvimento

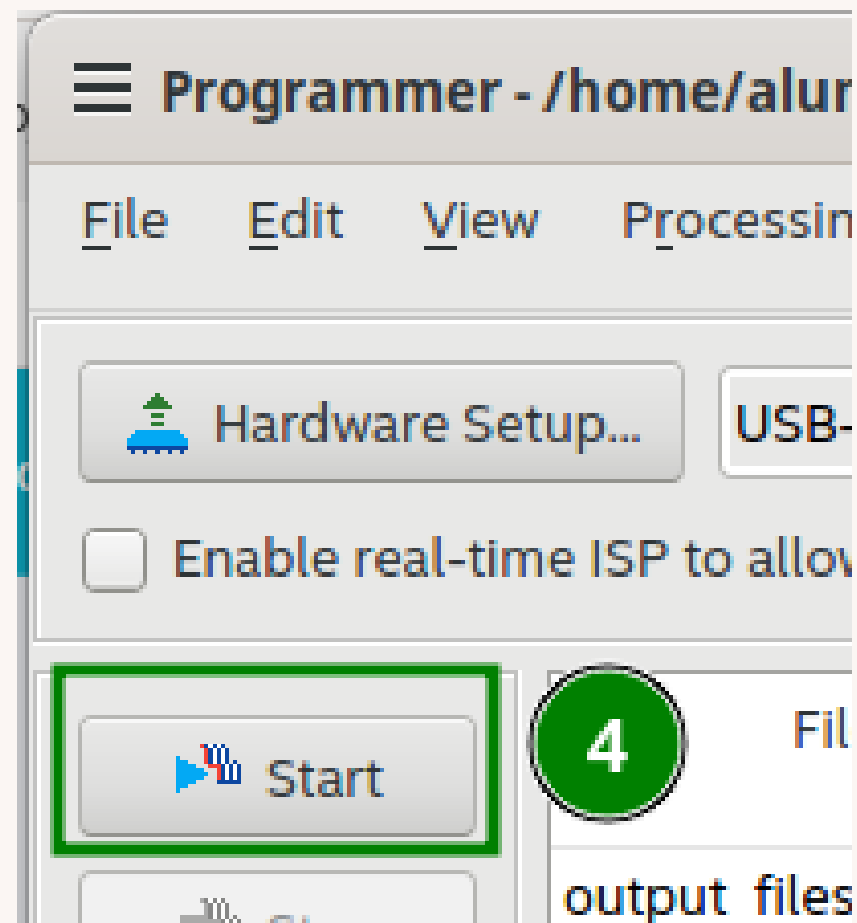
Gravar Código na Placa



Em Currently selected hardware:
2 - Selecione **USB-Blaster**

3 - Clique em **Close**

Gravar Código na Placa



4 - Cliente em **Start**

Fazendo esses passos o código será gravado na placa



**Muito
obrigado!**