

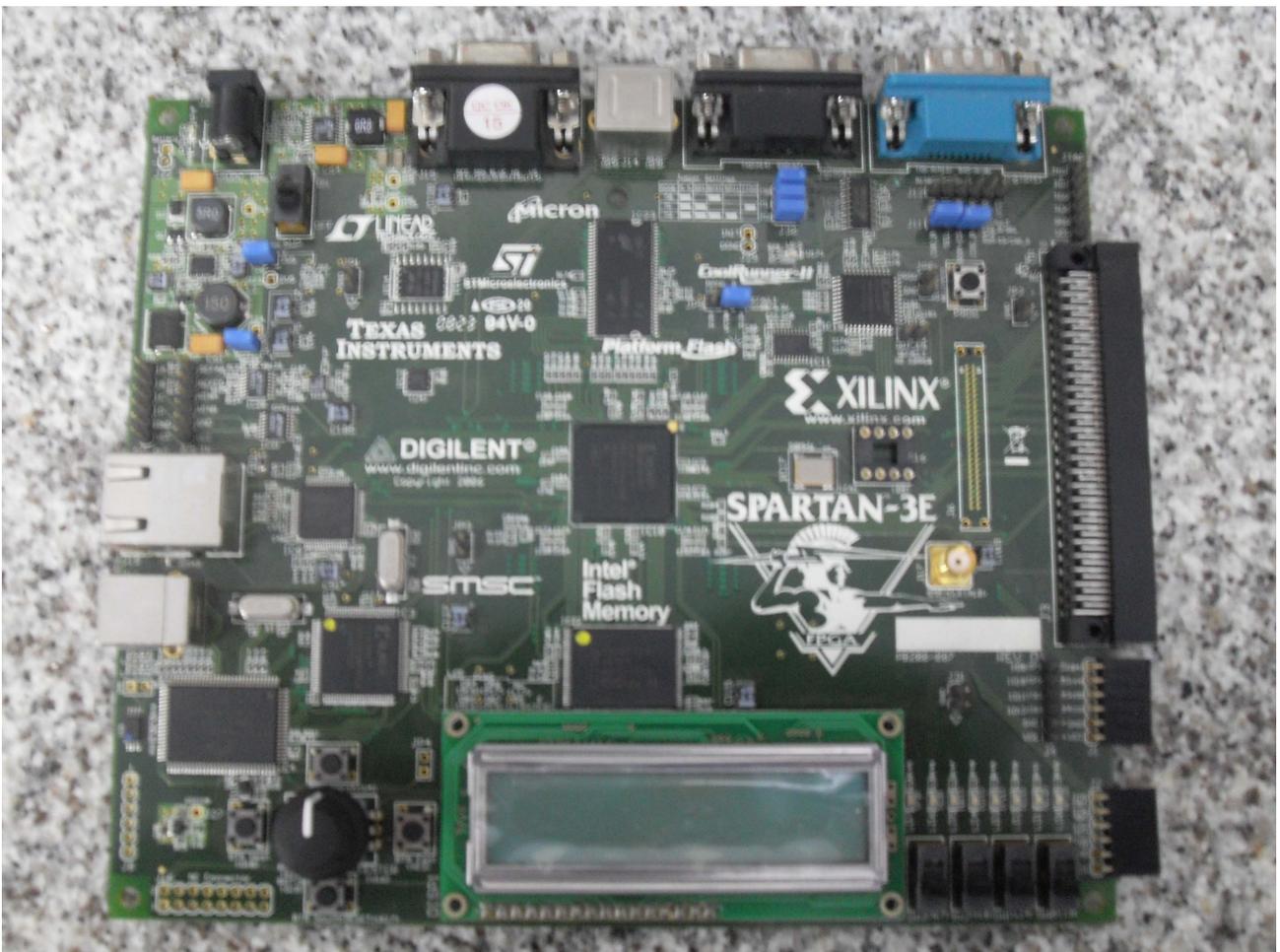
## Manual de Instalação da Plataforma Scorpion

### 1. Como conectar a Plataforma Scorpion com a Xilinx Starter-3E kit

Para começar a ler este manual, é bom que você conheça os componentes desta conexão.

– Plataforma Xilinx Starter-3E

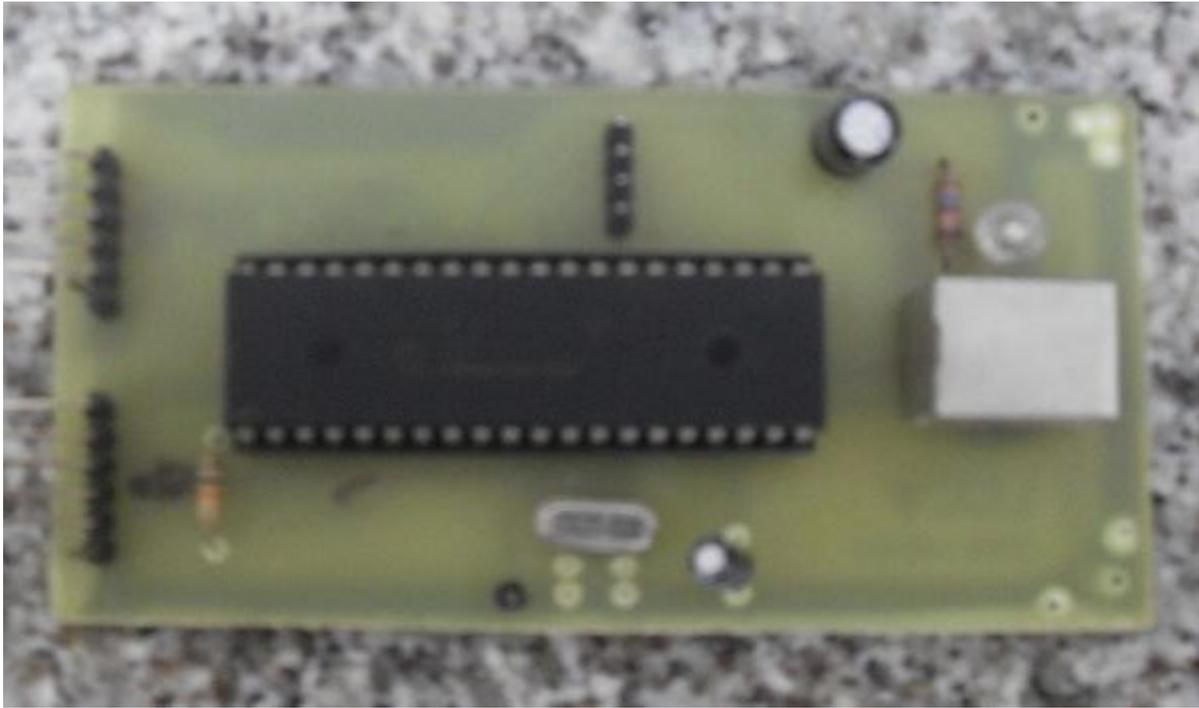
Nesta Plataforma, existe uma porta USB localiza à esquerda da figura. Esta porta deve ser conectada ao computador para que se possa configurar a FPGA. Na parte superior esquerda, possui uma conexão com a fonte de alimentação, que deve estar ligada o tempo todo na energia; pois, a FPGA trabalha como a memória RAM, e só vai ficar programada após você ter feito as configurações necessárias e até ela ser desligada da tomada.



*Figura 1: Plataforma Xilinx Starter-3E*

– Plataforma Scorpion

Esta Plataforma possui uma porta USB à direita da imagem. Esta porta deve ser conectada ao computador que irá executar o projeto, após a devida configuração da placa.



*Figura 2: Plataforma Scorpion*

– Flat Controller

O Flat Controller é composto por quatro fios, que irão fazer o controle da transmissão de dados entre as duas plataformas. Este controle é necessário porque os bytes que representam a operação e os operandos serão divididos em dois grupos de quatro bits cada. Então, estes fios farão o controle de quando uma plataforma poderá enviar/ler os bits que forem transferidos.



*Figura 3: Flat Controller*

Inicialmente, você precisa conectar as plataformas através do encaixe de pinos e ligar um Flat Controller.

A Plataforma Scorpion possui dois grupos de quatro pinos, localizados em seu lado esquerdo. Tais pinos funcionam para entrada e saída de dados. Eles devem ser encaixados à plataforma Xilinx Starter-3E através de seus conectores, localizados na parte inferior direita. Esta conexão é mostrada na figura 4.



*Figura 4: Conexão entre a Plataforma Xilinx Starter-3E e a Scorpion*

Feito isso, ligue o Flat Controller (mostrado na figura 3). Os quatro fios devem ser ligados aos pinos IO9, IO10, IO11, IO12; esta conexão é mostrada na figura 5.



*Figura 5: Conexão entre as plataformas e o Flat Controller*

## 2. Programando a FPGA

Primeiramente, é importante dar uma breve explicação sobre o FPGA. O FPGA (Field Programmable Gate Array) é um chip que suporta a implementação de circuitos lógicos relativamente grandes. Consiste de um grande arranjo de células lógicas ou blocos lógicos configuráveis contidos em um único circuito integrado. Cada célula contém capacidade computacional para implementar funções lógicas e realizar roteamento para comunicação entre elas.

Para que você possa fazer um teste de conexão entre as plataformas, estamos disponibilizando um projeto de somador-subtrator de 8 bits, que já possui um protocolo de comunicação preparado para funcionar em conjunto com a Plataforma Scorpion (o princípio de funcionamento está descrito no Anexo I). Após carregar o *bitstream* (é uma sequência de bits que representa o projeto compilado) para o FPGA, o projeto estará pronto para comunicar-se com a Scorpion, enviando e recebendo dados.

Para carregar o projeto, você deve abrir o Xilinx ISE 10.1 e ir até o menu File > Open Project. Procure a pasta onde você salvou o arquivo que disponibilizamos para teste, abra as pastas Somador-LCD > Somador-LCD > Scorpion\_ADD\_LCD > ScorpionNovo > Scorpion\_20082009 > LCD e selecione o arquivo LCD.isc.

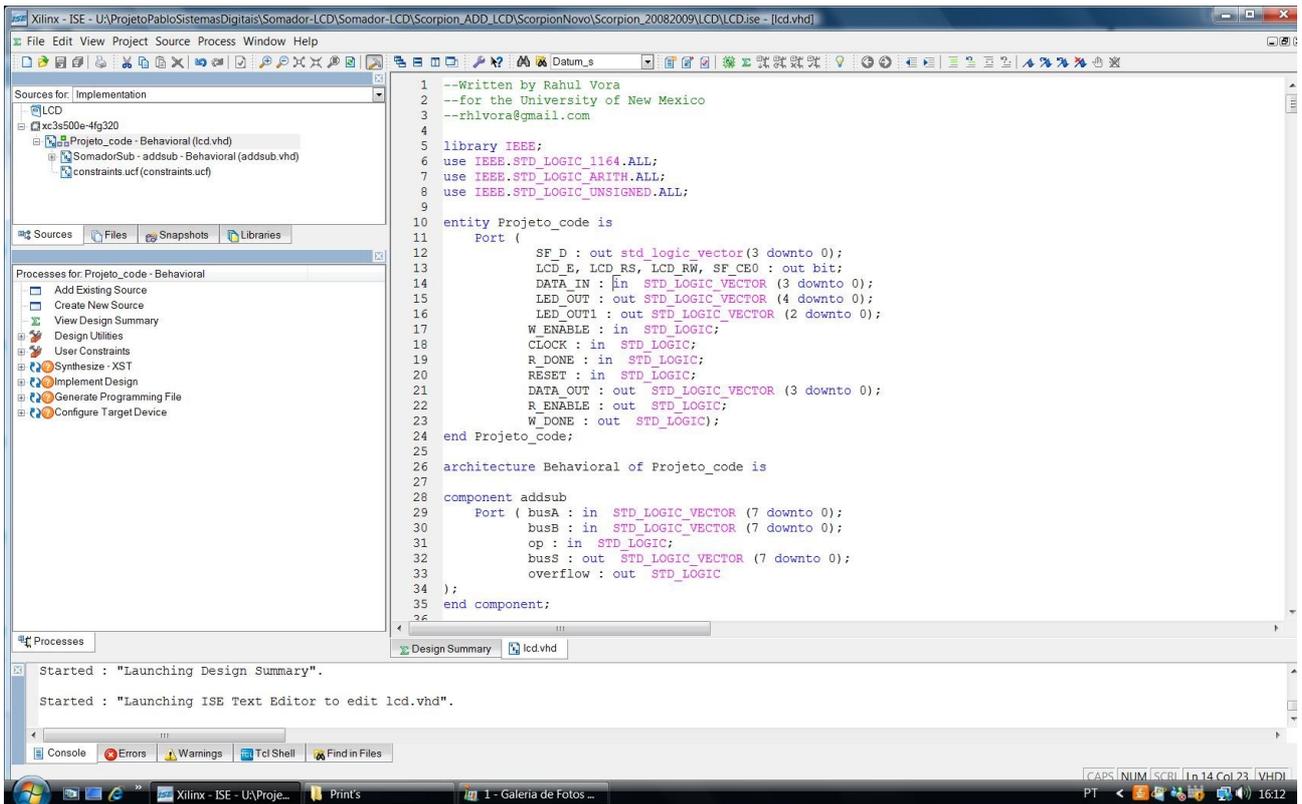


Figura 6: Arquivo do Projeto Somador-Subtrator

Com o LCD.ise aberto, você deve ir até a aba Processes e seguir quatro passos:

a) Dê dois cliques em Synthesize – XST. Aqui o programa vai sintetizar o seu projeto e buscar possíveis erros;

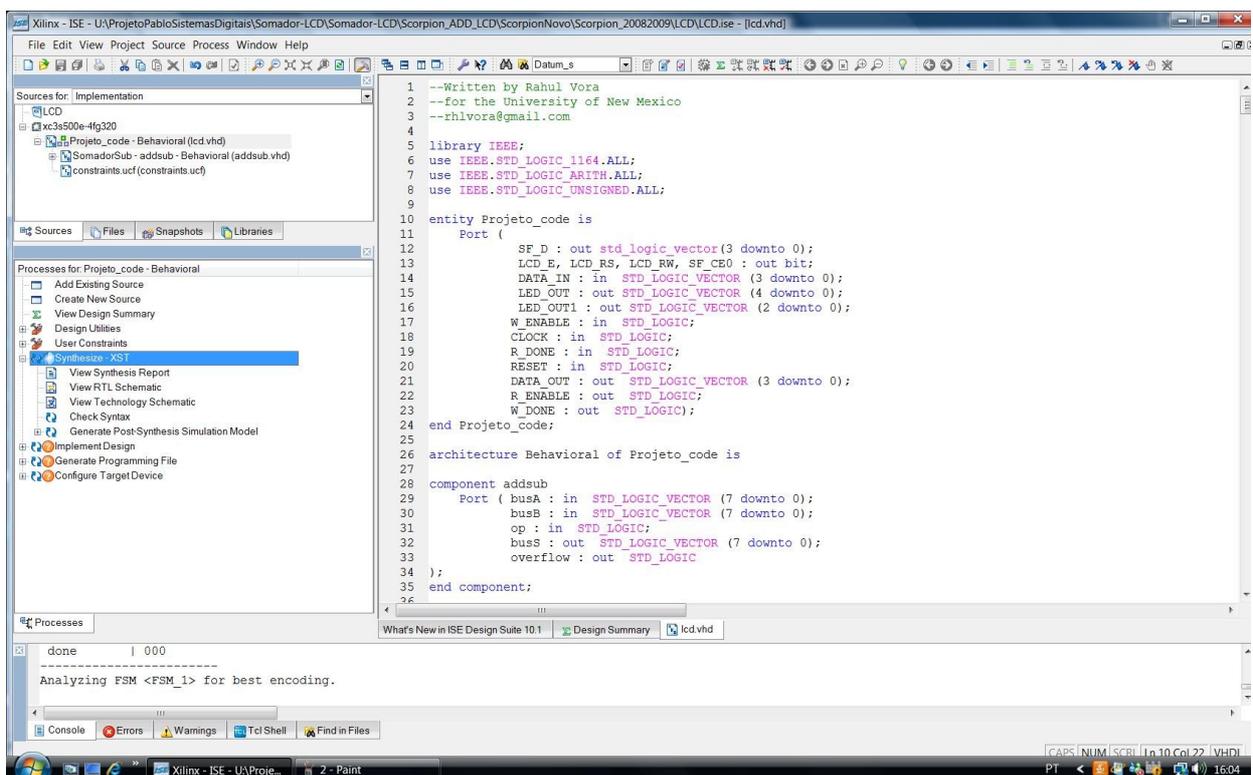


Figura 7: Synthesize - XST

b) Dê dois cliques em Implement Design;

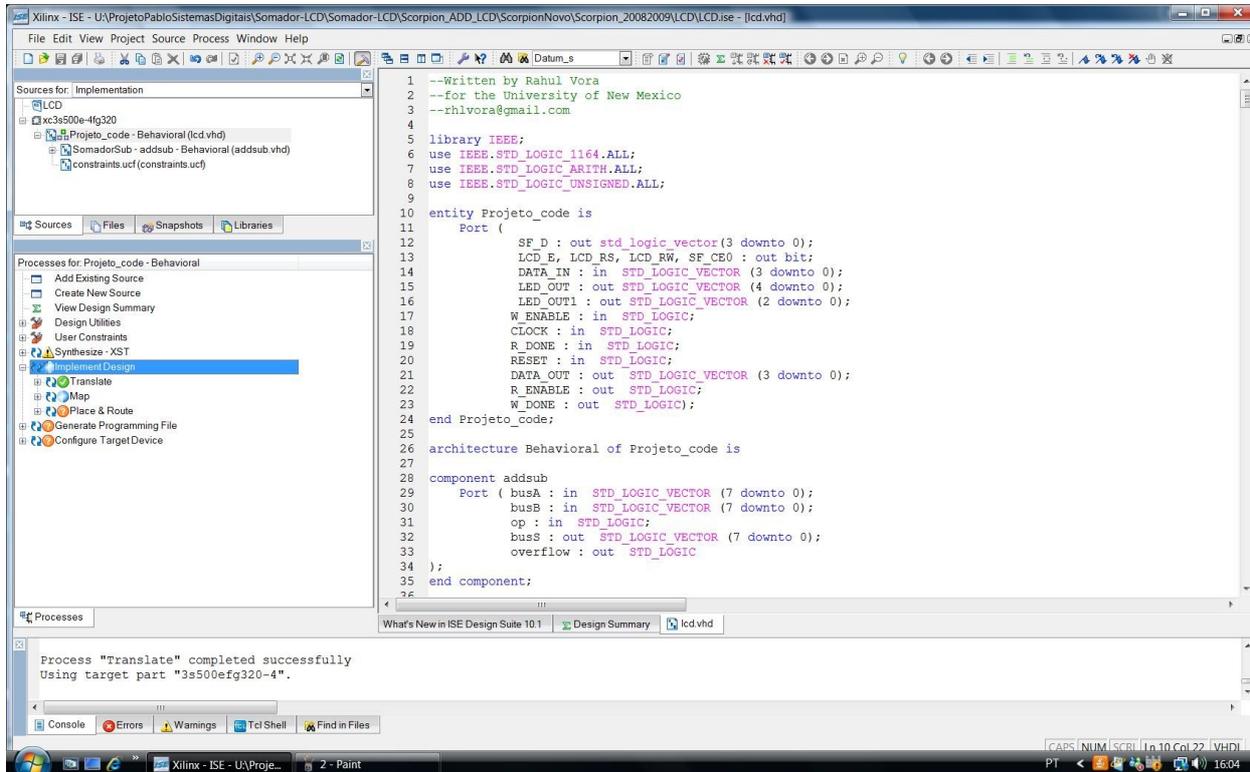


Figura 8: Implement Design

c) Dê dois cliques em Generate Programming File;

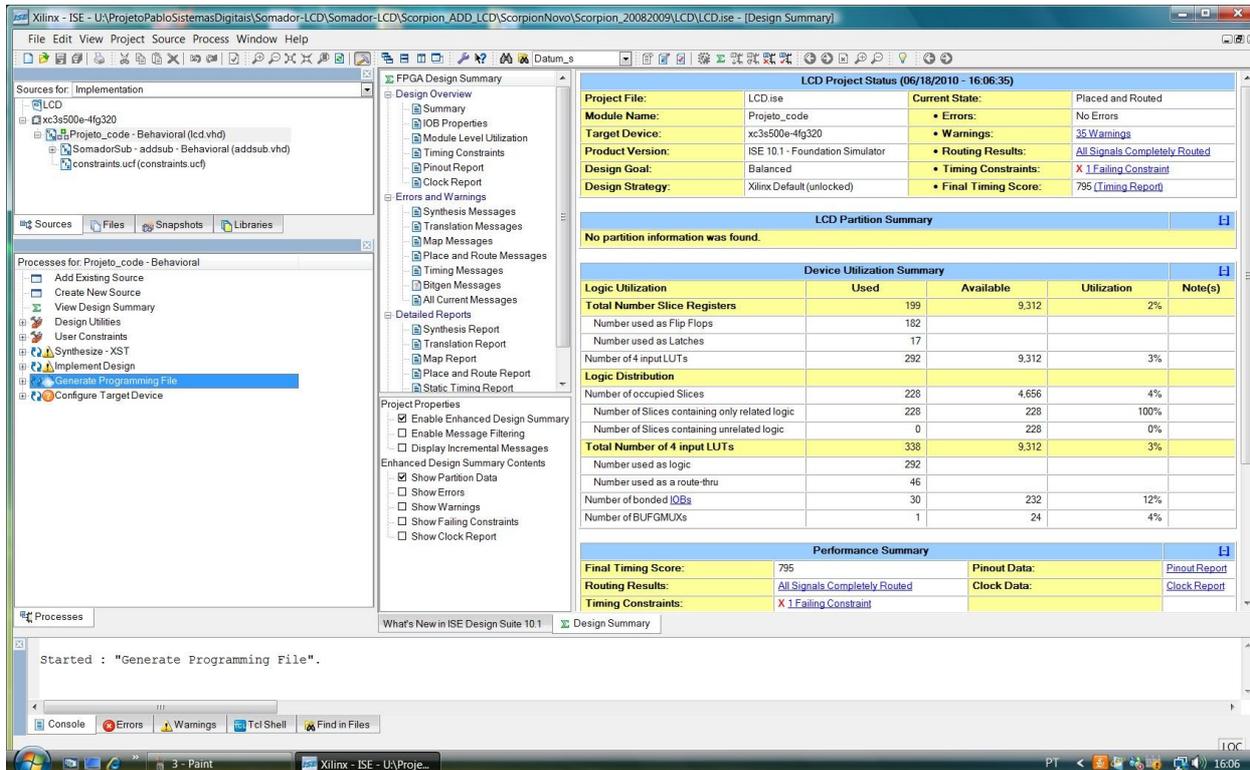


Figura 9: Generate Programming File

- d) Clique no '+' que está ao lado de Configure Target Device e dê dois cliques em Manage Configuration Project. Se aparecer o aviso que está na tela abaixo, clique em OK.

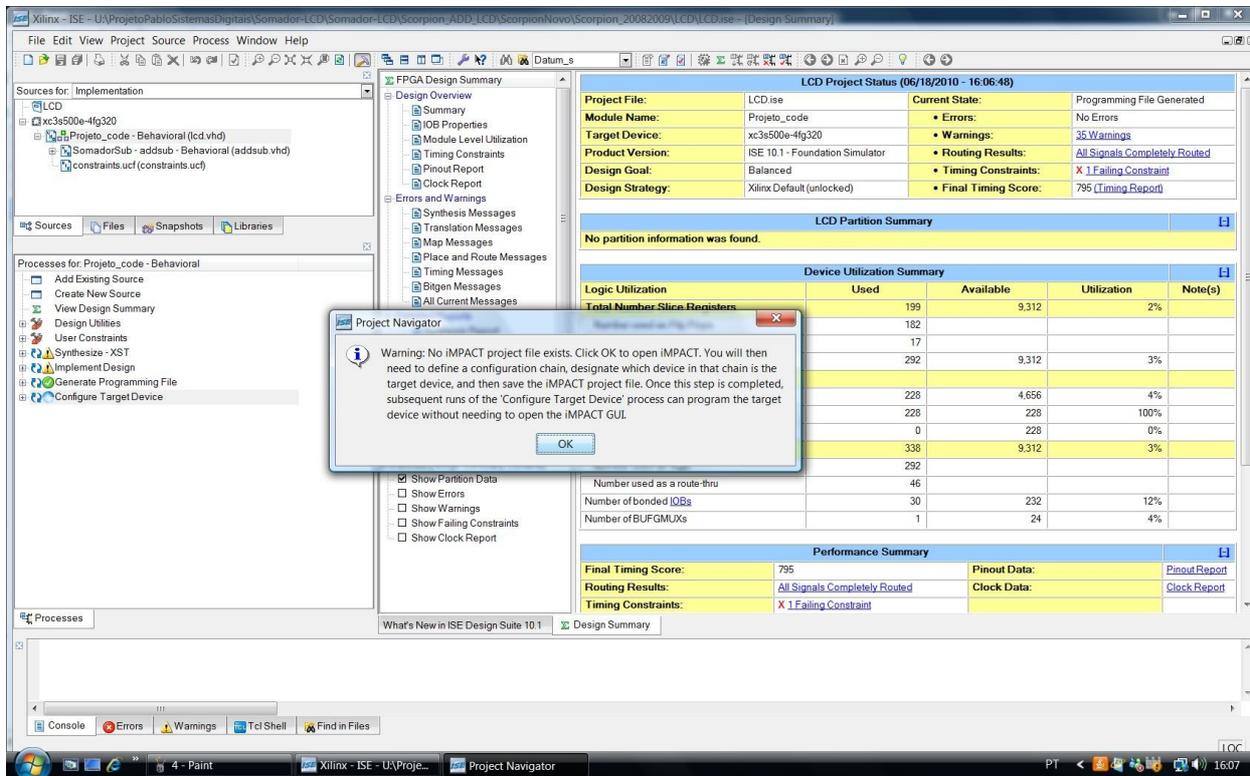


Figura 10: Manage Configuration Project

Após isso, aparecerá a tela abaixo. Basta clicar em Finish.

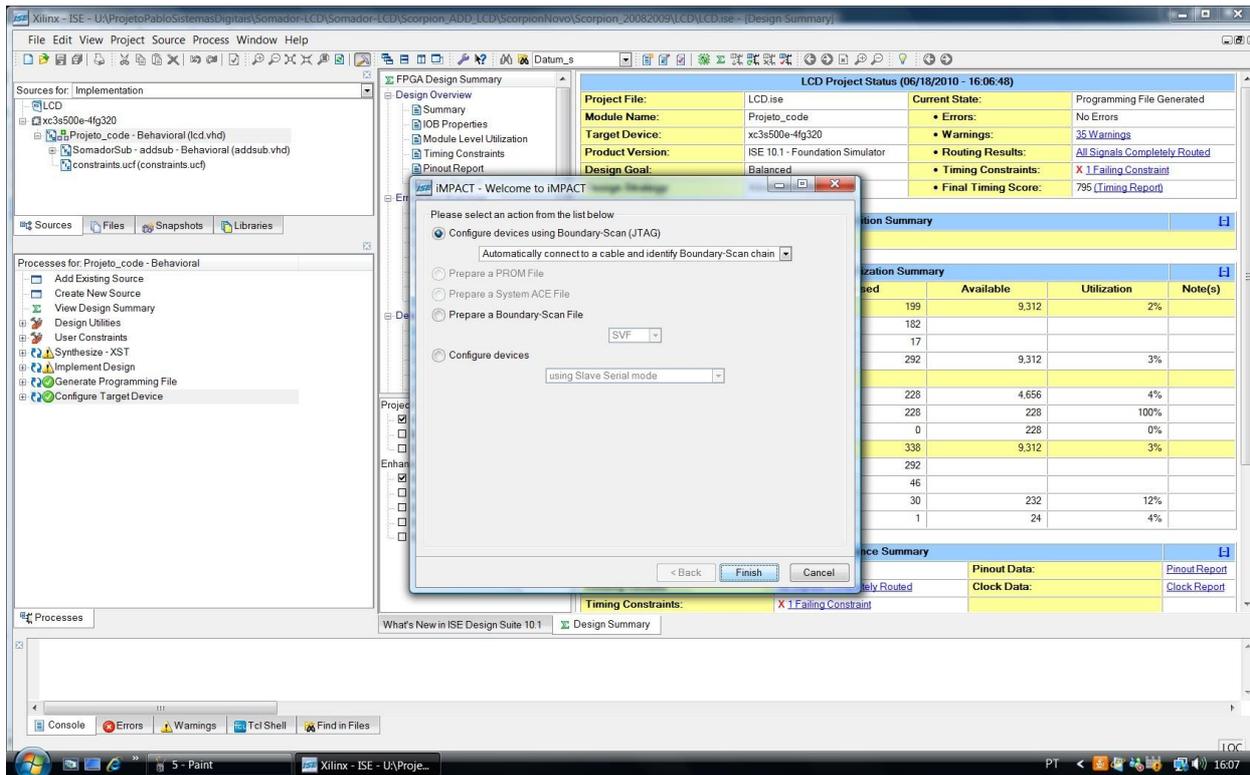


Figura 11: iMPACT

Agora, você só precisa carregar o *bitstream* para o FPGA. O quadrado verde que aparece na imagem abaixo representa a FPGA. Nesta parte, você deve abrir o arquivo `lcd.bit`, que está na mesma pasta que o `LCD.ise`.

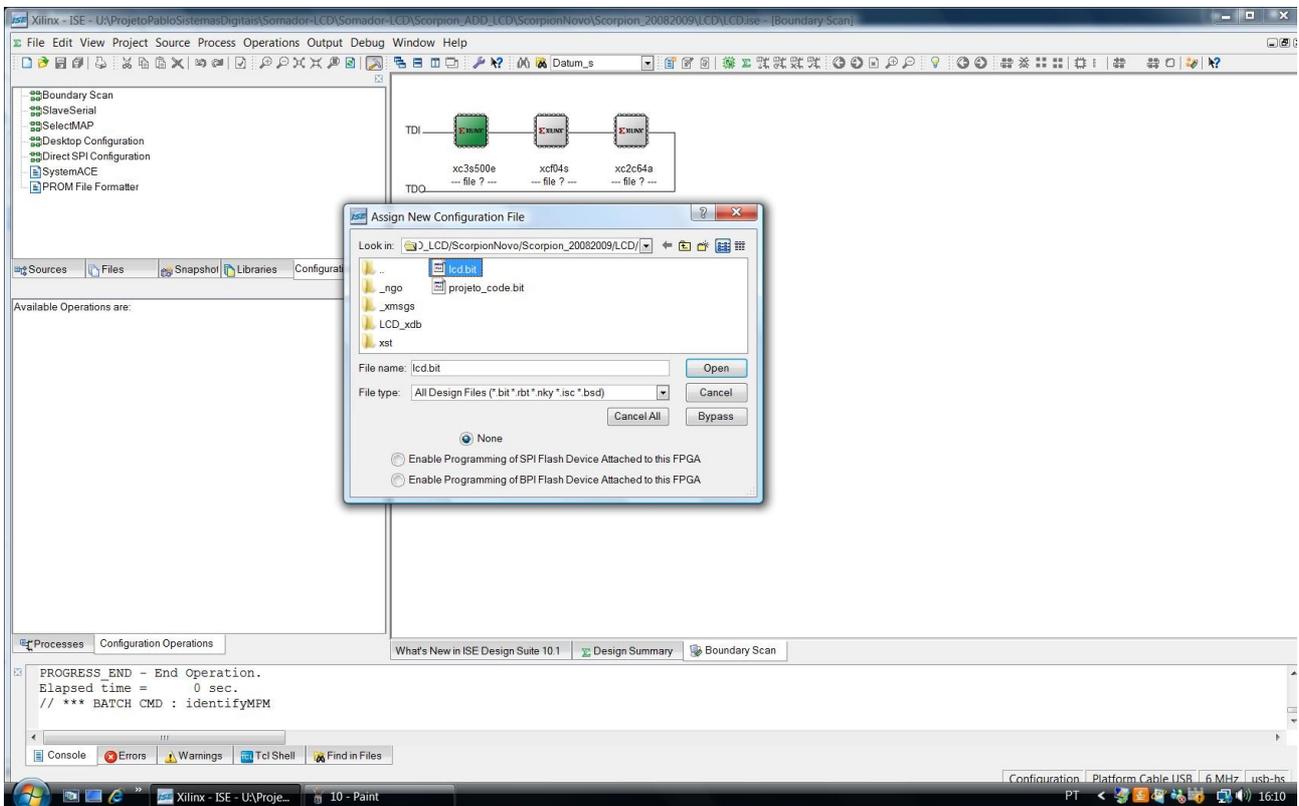


Figura 12: Assign New Configuration File (FPGA)

Os próximos quadros representam a memória PROM e o CPLD, que não serão

configurados. Então, nas próximas telas, você só precisa clicar em Cancel.

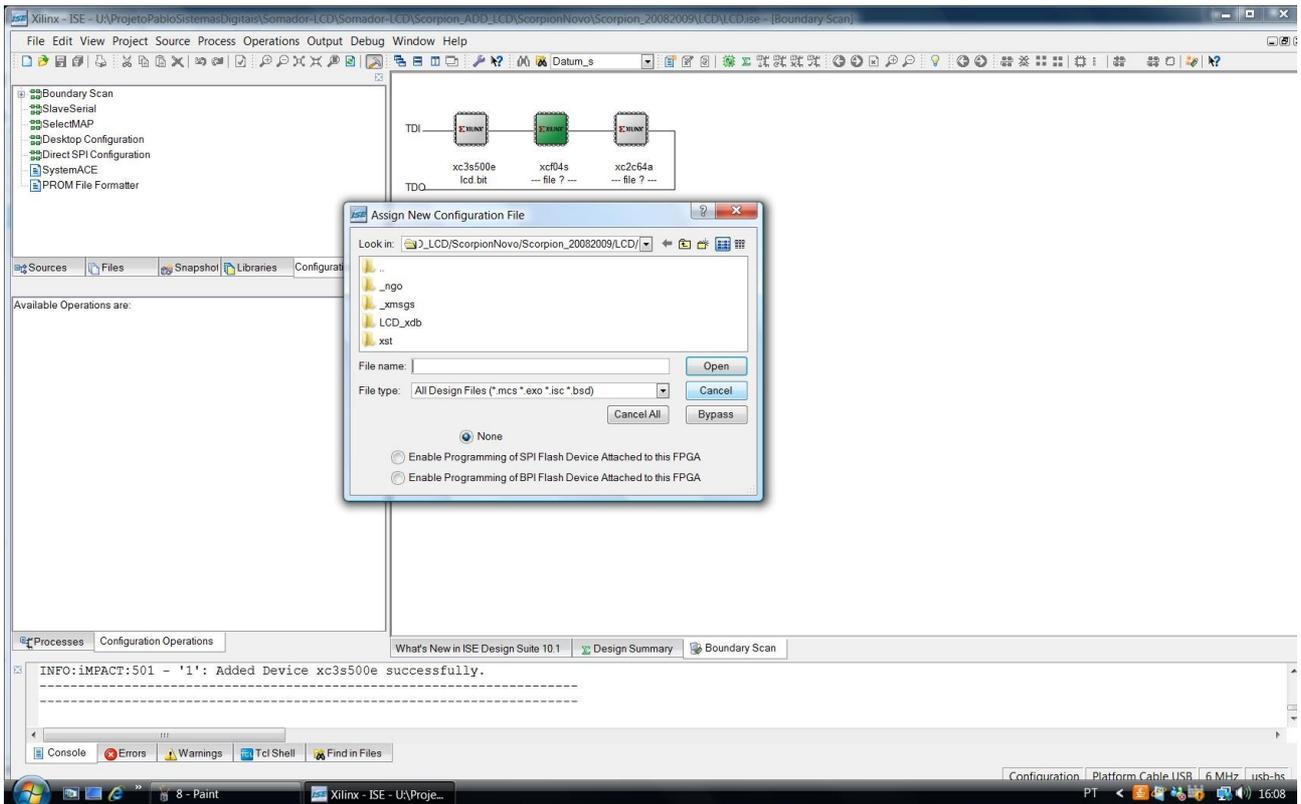


Figura 13: Assign New Configuration File (PROM)

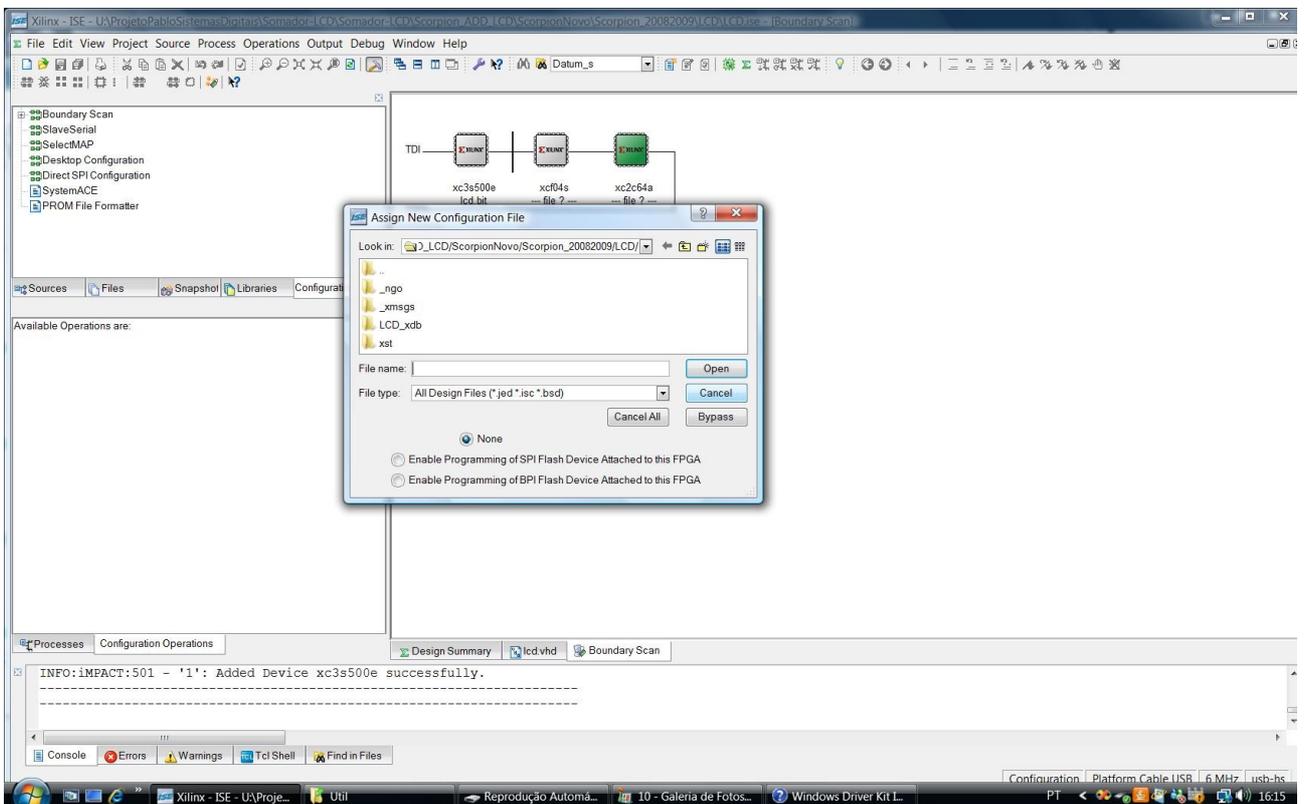


Figura 14: Assign New Configuration File (CPLD)

A próxima tela serve para configurações mais avançadas, você só precisa clicar em OK.

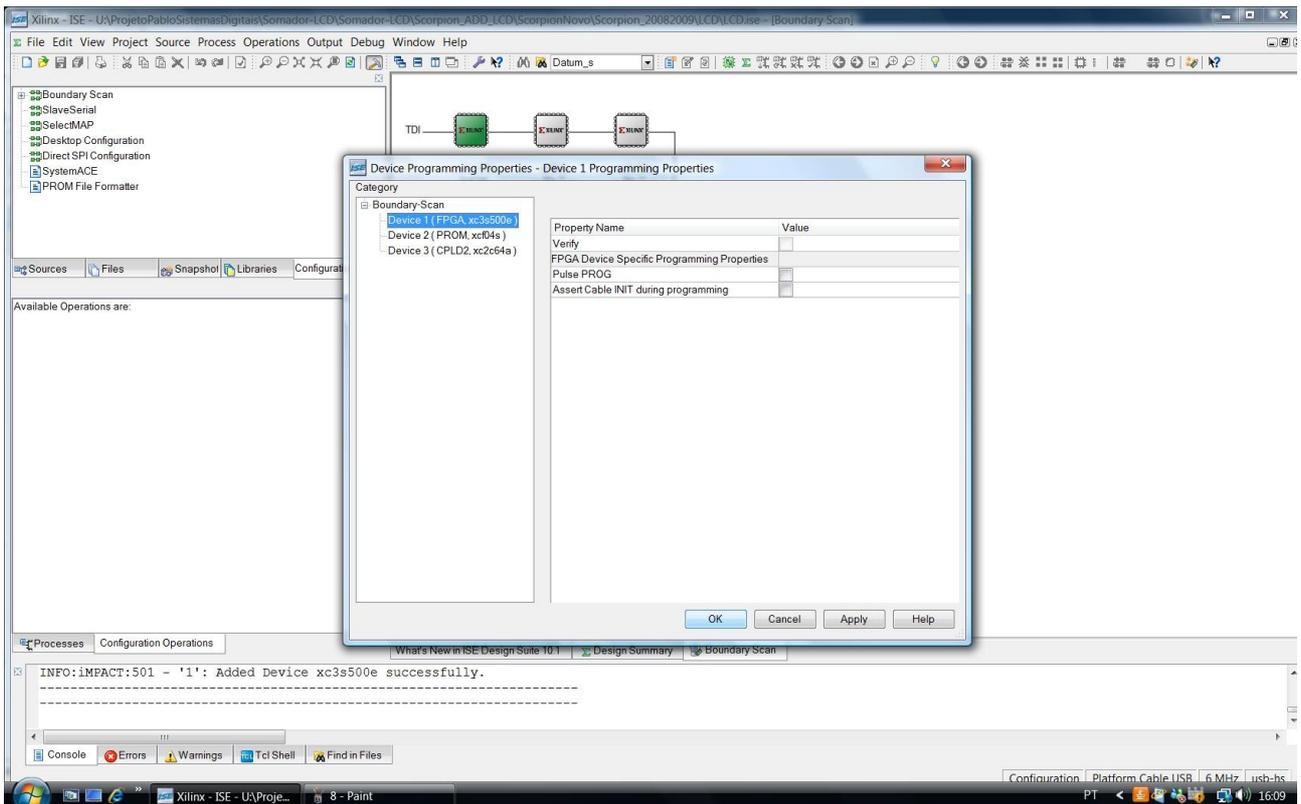


Figura 15: Device 1 Programming Properties

Agora, selecione o dispositivo FPGA (representado no primeiro quadrinho), clique com o botão direito e, em seguida, clique em Program.

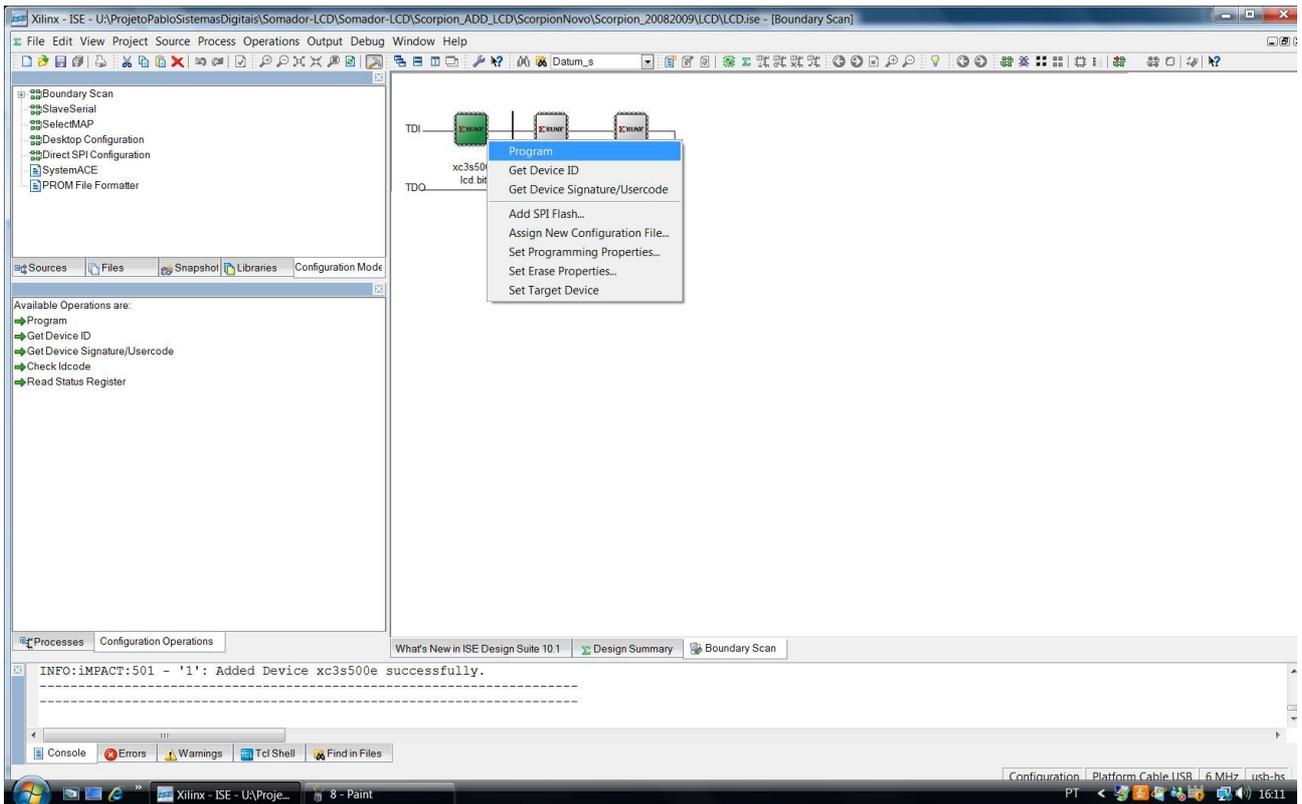


Figura 16: Program

Se tudo der certo e o programa conseguir configurar a FPGA corretamente, a próxima tela será assim:

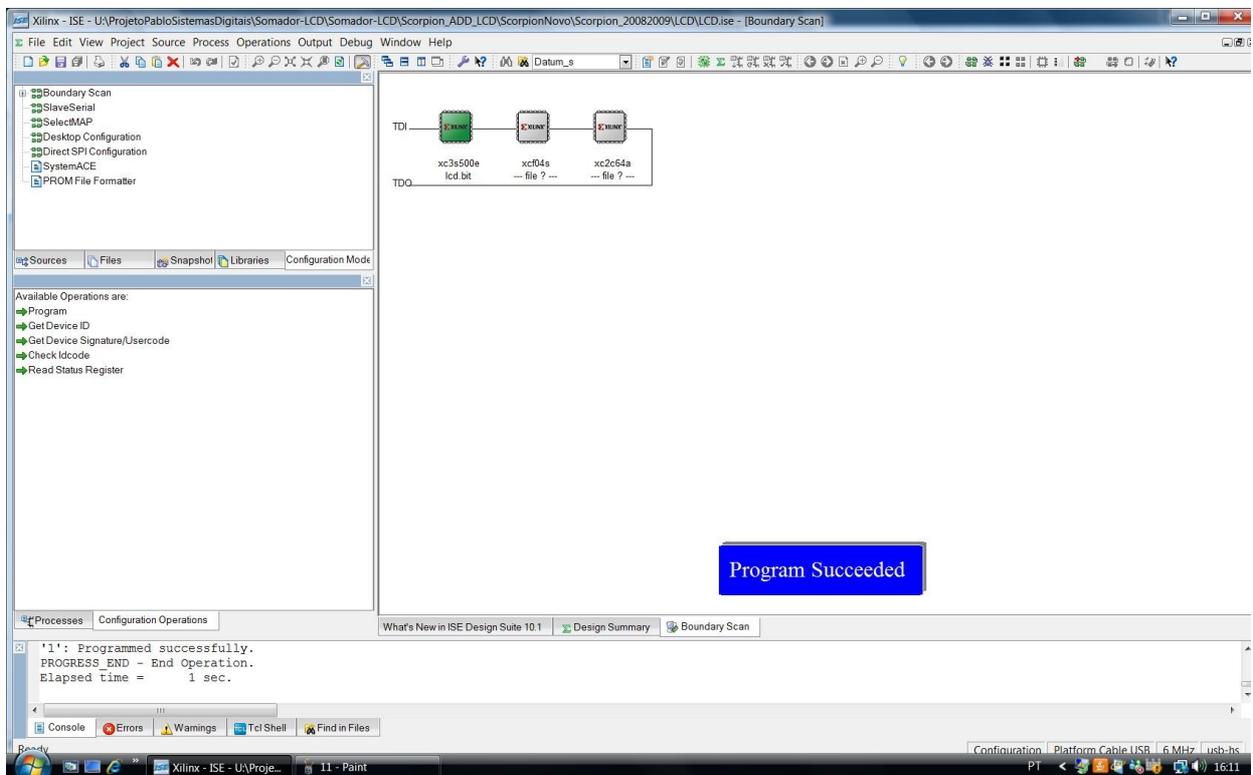


Figura 17: Program Succeeded

### 3. Ativação da Plataforma Scorpion

Até aqui, a configuração poderia ser feita usando tanto o Windows quanto o Linux. A partir de agora, todos os passos deverão ser executados em alguma plataforma Linux.

O próximo passo é ativar a Plataforma Scorpion; para isso, é bom que você entenda um pouco sobre seu funcionamento.

A Plataforma Scorpion trabalha com transferência de dados via USB, fazendo uso de funções primitivas da biblioteca LIBHID, da linguagem C++. Para que seja possível fazer o nosso programa funcionar corretamente na placa, você deve instalar esta biblioteca no Linux. Após a instalação, é necessário apenas, conectar a Plataforma ao computador através da porta USB citada na explicação da figura 2.

Você pode instalar a LIBHID de duas maneiras:

- pelo gerenciador Synaptic

Vá no menu System > Administration > Synaptic Package Manager. Em Quick Search, procure por “libhid” e instale os pacotes: libhid0 e libhid-dev.

- a partir do arquivo libhid-0.2.16.tar.gz, que é disponibilizado juntamente com este manual

Se você optar por instalar desta maneira, necessita apenas fazer a instalação via terminal no Linux. Para instalar no Ubuntu, primeiramente você deve descompactar o arquivo com o comando:

```
tar -zxvf libhid-0.2.16.tar.gz
```

Após isso, navegue até a pasta criada e execute no diretório raiz os seguintes comandos:

```
sudo ./configure  
sudo ./make  
sudo ./make install
```

Feito isso, o sistema já estará configurado para comunicar-se corretamente com a Scorpion.

#### ➤ Preparando a Scorpion

Para que a Scorpion funcione corretamente, você deve ter seguido corretamente os passos contidos neste manual: fazer as ligações dos pinos e do Flat Controller com a Xilinx Starter-3E e conectá-la ao computador através do cabo USB, e instalar a biblioteca LIBHID. Feito isso, você pode rodar o projeto que nós fornecemos – somador-subtrator de 8 (oito) bits – no terminal.

### 4. Rodando o aplicativo

Abra o terminal do seu Linux, navegue até a pasta onde o terminal (que nós disponibilizamos juntamente com este manual) está salvo (neste caso, o projeto está na Área de Trabalho, na pasta TerminalSomador) e execute o terminal.out com a seguinte linha de comando:

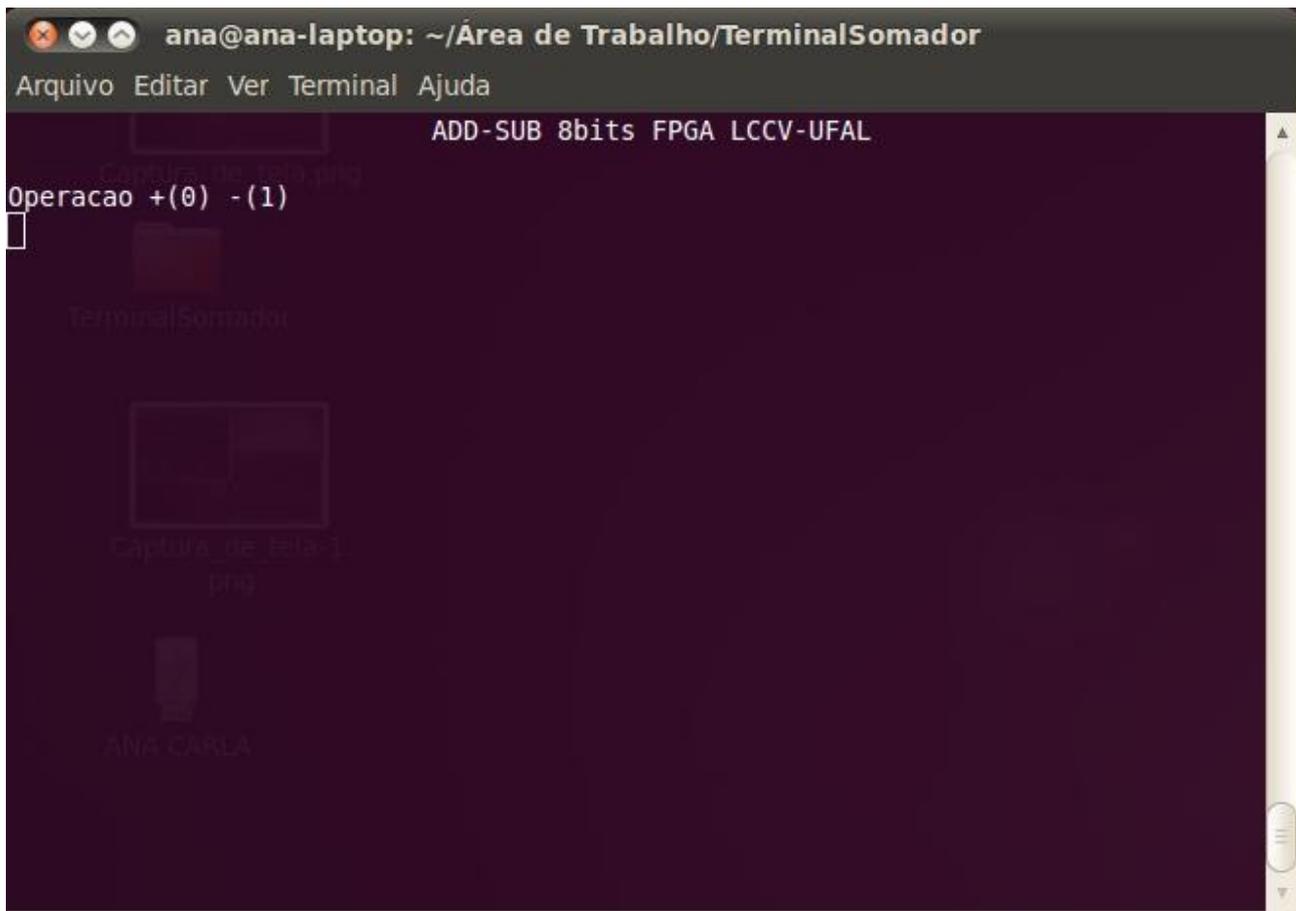
```
sudo ./terminal.out
```

Os outros comandos presentes na imagem serviram para buscar a pasta onde o terminal.out está e para listar os arquivos presentes em cada pasta.

```
ana@ana-laptop: ~/Área de Trabalho/TerminalSomador
Arquivo Editar Ver Terminal Ajuda
ana@ana-laptop:~$ cd Área\ de\ Trabalho/
ana@ana-laptop:~/Área de Trabalho$ ls
Captura_de_tela-1.png  Captura_de_tela.png  TerminalSomador
ana@ana-laptop:~/Área de Trabalho$ cd TerminalSomador
ana@ana-laptop:~/Área de Trabalho/TerminalSomador$ ls
binariochar.c      DeviceHID.h          mainADDSUB.cpp
binariochar.c~    DeviceHID.h.gch     mainADDSUB.cpp~
binariochar.cpp   Driver1.cc          terminal.out
binariochar.cpp~  DriverHID.cpp       tutorial
DeviceHID.cpp     DriverHID.cpp~     tutorial1.c
DeviceHID.cpp~   libhid0_0.2.15-1ubuntu1_i386.deb  valeu
ana@ana-laptop:~/Área de Trabalho/TerminalSomador$ sudo ./terminal.out
```

Figura 18: Terminal do Linux

Logo após, aparecerá a seguinte tela:



*Figura 19: Terminal do Somador-Subtrator de 8 bits*

O programa é bem simples: primeiramente, você escolhe a operação que deseja realizar, 0 (zero) para soma ou 1 (um) para subtração. Depois, o software solicitará que você insira os operandos; lembre-se que este projeto dá suporte à operações com palavras de até 8 (oito) bits. A entrada dos operandos é feita com caracteres ASCII, cada caractere possui um código binário correspondente; no exemplo abaixo, nós tentamos somar o caractere 3 com o caractere >, que têm como correspondentes binários, 00110011 e 00111110, respectivamente.

```
ana@ana-laptop: ~/Área de Trabalho/TerminalSomador
Arquivo Editar Ver Terminal Ajuda
ADD-SUB 8bits FPGA LCCV-UFAL
Operacao: 0 (00110000)
Operando1: 3 (00110011)
Operando2: > (00111110)
TRACE: hid_interrupt_read(): retrieving interrupt report from device 007/002[0
] ...
WARNING: hid_interrupt_read(): failed to get interrupt read from device 007/002[
0]: could not claim interface 0: Device or resource busy
FAILURE calling hid_interrupt_read
Resultado (00000000)
ADD-SUB 8bits FPGA LCCV-UFAL
Operacao +(0) -(1)
█
```

*Figura 20: Execução do Somador-Subtrator de 8 bits*

Logo após a entrada do segundo operando, o software enviará as informações para a Plataforma Scorpion via USB, que aguardará o retorno do FPGA. Quando o FPGA responder, o resultado será mostrado na tela. Após a exibição do resultado, o FPGA já estará pronto para uma nova operação.

OBS: No teste usado para o preparo deste manual, ocorreu um erro durante o envio dos dados à FPGA, portanto, não obtivemos o resultado correto da nossa soma.

## Anexo I

### Princípio de Funcionamento do Protocolo FPGA-Scorpion-computador

A Plataforma Scorpion possui dois grupos de pinos compostos por quatro pinos cada um, o grupo superior é responsável pelo envio de dados e o inferior, pelo recebimento de dados. Desta forma, a comunicação é realizada com 4 (quatro) bits de cada vez.

A Scorpion foi configurada para comunicar-se com a porta USB enviando e recebendo apenas 1 (um) byte por vez, como só possui 4 (pinos) de comunicação, é necessário dividir o byte em dois grupos de 4 (quatro) bits.

Além da divisão do byte, também é necessário o controle do tráfego de dados (envio e recebimento), para que possamos garantir a integridade da informação. Assim, faz-se necessária uma verificação realizada por cada plataforma para saber se ela está pronta para enviar ou receber algum dado. Desta maneira, podemos garantir que não haverá perda de dados, pois a transferência só vai acontecer se o dispositivo que vai receber o dado estiver pronto. Este controle de envio e recebimento de dados é realizado pelo Flat Controller. O Flat Controller é composto de 4 (quatro) fios.