

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTOS ACADÊMICOS DE ELETRÔNICA E  
INFORMÁTICA

GEOVANA FRANCO SANTOS  
JIMMY YUJI TANAMATI SOARES

PROJETO FINAL

CURITIBA  
2018

# 1 INTRODUÇÃO

Este documento tem o objetivo de apresentar uma aplicação para o kit de desenvolvimento da família Cyclone II do fabricante Altera, utilizando FPGAs (Field Programmable Gate Array).

Para isso foi desenvolvido um sistema que requisita de um servidor, utilizando comunicação TCP/IP através de sockets, um arquivo de texto e exibe os dados recebidos através do LCD 16x2 presente no kit de desenvolvimento. Utilizou-se o software QUARTUS II como ambiente de desenvolvimento para a criação dos código e na programação do dispositivo lógico estudado.

Aqui serão descritas as características e especificações utilizadas no desenvolvimento do projeto.

## 1.1 Especificações do projeto

Para o desenvolvimento do projeto as seguintes especificações foram feitas:

- O kit se comporta como cliente em uma comunicação TCP/IP através de sockets com o servidor.
- Interfaceamento por software com o componente LCD.
- Controle de leitura utilizando os botões KEY3 e KEY2.
- Escolha de arquivos utilizando duas chaves do tipo switch.
- Requisição do arquivo através do botão KEY1.
- Recepção de um arquivo texto a ser processado e exibido.

## 2 DESENVOLVIMENTO E FUNCIONAMENTO

As seções a seguir descrevem o desenvolvimento do projeto e como proceder o funcionamento.

### 2.1 Desenvolvimento

O desenvolvimento do projeto se iniciou utilizando um projeto exemplo fornecido pela Altera que utiliza o Qsys para o desenho da plataforma e conexões entre os componentes existentes no kit de desenvolvimento, dessa forma o tempo gasto criando as conexões e pinagem de cada componente foi reduzido. Utilizou-se o NIOS II como processador, dessa forma é possível utilizar um projeto em linguagem de programação C para realizar o interfaceamento e processamento dos dados utilizados. No NIOS II utilizou-se um projeto que implementa uma comunicação socket com um servidor. O usuário pode através das chaves SW0 e SW1 escolher quais arquivos deseja requisitar seguindo a seguinte sequência:

- ☐ 00 - ArquivoA.txt
- ☐ 01 - ArquivoB.txt
- ☐ 10 - ArquivoC.txt
- ☐ 11 - ArquivoInválido.txt

Após a seleção o usuário pode acionar o botão KEY1 que fará a requisição. Após a recepção do arquivo o programa divide a string recebida em partes de 16 caracteres, e no LCD do kit são exibidos duas dessas partes. Quando acionado o botão KEY3 ou KEY3 o usuário pode visualizar as demais linhas, indo para cima ou para baixo no texto.

## 2.2 Funcionamento

Para a compilação e execução do projeto desenvolvido é necessária a utilização do Quartus 13.0sp1 e seguir os seguintes passos:

1. Utilize o Quartus II versão 13.0sp1
2. Descompacte PROJ\_FINAL\_GEOVANA\_JIMMY.zip para obter a pasta interna DE2\_NET
3. Abra o DE2\_NET.qpf
4. Vá na aba Files do Project Navigator e delete o arquivo system\_0/synthesis/system\_0.qip
5. Abre o Qsys em Tools->Qsys
6. Selecione o arquivo system\_0.qsys e clique em Abrir
7. De OK ou Close no que aparecer
8. Va para aba Generation e clique em Generate
9. Feche o Qsys
10. No Project Navigator, aba Files, clique com o botão direito em Files->Add/Remove Files in Project
11. Clique no [...], entre em system\_0/synthesis, mude o tipo de arquivo para todos, selecione o arquivo system\_0.qip
12. Clique em Add-> Apply-> OK
13. Abra o Programmer, aperte Auto Detect e aperte Start, caso o Progress esteja verde, continue caso contrário verifique conexão da JTAG
14. Deixe aberto o Programmer e a janela "OpenCore Plus Status" e inicie o Eclipse em Tools -> Nios II Software Build Tools for Eclipse
15. No Workspace selecione SeuCaminhoAtéaPasta\DE2\_NET\software
16. Agora existem 2 opções:
  - 16.1. Se os projetos simple\_socket e simple\_socket\_bsp estão aparecendo e são expandíveis (pastinha cor amarela):

- 16.1.1. Clique com o botão direito em cada projeto e clique em Clean Project
- 16.1.2. Abra o arquivo iniche\_init.c, dentro da função SSSInitialTask irá encontrar 2 linhas de código e fazer alterações:
  - sa.sin\_port = htons(7777); // ALTERAR PORTA
  - res = inet\_pton(AF\_INET, "192.168.1.110", &sa.sin\_addr);
- 16.1.3. Vá em Project -> Build All
- 16.1.4. Se tudo comilar sem erros, é possível fazer Botão direito em simple\_socket -> Run As...->Nios II Hardware
- 16.1.5. Se der algum erro na hora de rodar apenas, tente denovo, o Eclipse tem dessas
- 16.1.6. Se rodar como deve, algo assim deve aparecer no Nios II Console:

```
InterNiche Portable TCP/IP, v3.1
                                Copyright 1996-2008 by InterNiche
Technologies. All rights reserved.
Your Ethernet MAC address is 00:07:ed:ff:cd:15
prepped 1 interface, initializing...
dm9ka_init
Created "Inet main" task (Prio: 2)
Created "clock tick" task (Prio: 3)
mctest init called
IP address of et1 : 10.3.2.175
Acquired IP address via DHCP client for interface: et1
IP address : 192.168.1.101
Subnet Mask: 255.255.255.0
Gateway : 192.168.1.1
Simple Socket Server starting up
Socket criado
```
- 16.1.7. Se deu certo, precisa testar com um server agora. Primeiro pare o programa do Eclipse.

- 16.1.8. Pegue o tcpserver.py (fazendo alterações de ip e porta dentro) fornecido e rode ele fazendo o comando python .\tcpserver.py na linha de comando no local que estiver o arquivo
  - 16.1.9. Rode novamente com Run As->Nios II Hardware
  - 16.1.10. Se o roteador não bloqueou nada, está pronto! O LCD da placa deve estar mostrando o arquivo atual, SW0 e SW1 mudam o arquivo pedido e clicar com o KEY1 pede o arquivo. Tendo sucesso o arquivo aparece no LCD, podendo ser movido para cima ou para baixo com KEY3 e KEY2. Clicar KEY1 novamente retorna a seleção de arquivo
- 16.2. Se os projetos simple\_socket e simple\_socket\_bsp estão aparecendo e NÃO são expandíveis (pastinha cor azul):
- 16.2.1. Delete ambos com botão direito -> Delete (sem deletar do computador)
  - 16.2.2. Botão direito no Project Explorer->Import...->Nios II Software Build Tools Project->Import Custom Makefile for Nios II Software Build Tools Project-> Next->Browse...->Vá para a pasta do projeto DE2\_NET/software/simple\_socket\_bsp-> OK/Abrir  
Project name: simple\_socket\_bsp  
Faça a mesma coisa para o simple\_socket (deixando o Project name SEM o \_bsp)
  - 16.2.3. Retome a partir de 16.1.1

Importante notar que as plataformas utilizam caminhos absolutos para as referências de arquivos e projetos, dessa forma, mesmo seguindo os passos, é possível que a compilação reproduza erros.

### 3 CONCLUSÕES

Esse projeto permitiu à equipe um maior contato com o uso de FPGAs. Foram observadas ao longo do projeto grandes dificuldades com relação à documentação e compilação de exemplos fornecidos pelo fabricante. Grande parte do tempo de desenvolvimento foi gasto tentando entender como as partes do kit se comportam, pois os componentes e alguns dos exemplos fornecidos já estão sem suporte ou sem correções de possíveis bugs.