UTILIZAÇÃO DE UCLINUX EM HARDWARE RECONFIGURAVÉL COM PROCESSADOR NIOS II PARA AUTOMAÇÃO INDUSTRIAL

Gabriel Martins Buim [1], João Carlos Siqueira [2], Edson Antonio Batista [3], Alexsandro Monteiro Carneiro[4], Mauro Conti Perreira [5], Alexandre César Rodrigues da Silva [6]

Abstract - *This article describes the usage of an RTOS (Real-Time Operating System) µClinux for management and communication between FPGA's boards and supervisory software through Ethernet network. The software was developed in JAVA, in order to implement the control and automation of industrial processes and the hardware, implemented in reconfigurable hardware with NIOS II processor. This hardware was basically developed with SOPC Builder (System On a Programmable Chip) and a Linux distribution Ubuntu in Debian source-code. Inside SOPC Builder, the driver DM9000A was inserted to control the Ethernet protocol of communication. Implementing this work, it is possible to develop applications to control process using Ethernet network and a embedded module made of reconfigurable hardware.*

Index Terms — Ethernet, FPGA, RTOS.

INTRODUÇÃO

As redes ethernet estão ganhando espaço dentro da automação industrial. Segundo o diretor de redes industriais da CISCO, em alguns anos estes tipos de rede dominarão 100% das aplicações industriais. A justificativa para o crescimento das redes ethernet está baseado na sua velocidade, que pode variar de 10Mb/s até 10Gb/s, também no custo. Enquanto um controlador básico de ethernet custa de R$ 40 a R$ 100, os dispositivos proprietários custam de R$ 600 a R$1.800.

As redes ethernet são úteis no chão de fábrica, onde a aquisição de dados de sensores e atuadores operam em ordem de milisegundos.

Outra grande vantagem das redes ethernet é a utilização do protocolo TCP/IP, facilitando assim, o acesso ao sistema pela internet[1].

Um sistema automatizado precisa de um hardware capaz de executar atividades complexas e técnicas de controle avançadas. Este tipo de hardware favorece a construção de rede de transdutores inteligentes e maior flexibilidade na manutenção.

A computação reconfigurável tem a capacidade de conciliar a velocidade de um hardware específico com a flexibilidade dos microprocessadores. O grande diferencial dos sistemas reconfiguráveis é a possibilidade de alterar as instruções do sistema, sem a necessidade de construir um novo hardware, alterando suas instruções via software. Destaca-se o FPGA (Field Programable Gate Array) como hardware reconfigurável, podendo ser descrito e sintetizado em um dispositivo reprogramavél [2].

Para o controle do hardware é utilizado o processador NIOS II, devido a sua capacidade de controle de núcleo via software configurável.

O sistema atrativo para controlar um hardware reconfigurável são os RTOS (Real Time Operating System). Os RTOS são sistemas operacionais que funcionam em tempo real, utilizados para embarcá-los em microcontroladores e FPGA's, devido a sua capacidade de trabalhar com curto tempo de resposta e executar mult-threadings. O µClinux é um RTOS Open Source, inicialmente desenvolvido em linux kernel 2.0, por não possuir suporte para gerenciamento de memórias, era apenas utilizado para microcontroladores. Posteriormente foram lançadas atualizações para novos kernel como linux 2.6 e passou a ser mais explorado por desenvolvedores de hardware [10].

O objetivo de embarcar sistemas operacionais (RTOS) em FPGA está sendo estudado com mais atenção, pois, é possível realizar maiores quantidades de operações em pouco tempo de desenvolvimento. O funcionamento deste, é similar à utilização de um sistema linux em modo texto (prompt de comandos). Portanto, exige nível de conhecimento elevado de linux devido a sua complexidade de comandos existentes. A grande vantagem de utilizar o µClinux, é que este, contém um forte suporte a protocolos de comunicação, pois possui STACK TCP/IP, além de suporte a outras aplicações de rede.

A configuração e desenvolvimento do RTOS µClinux somente pode ser possível dentro de um sistema operacional Linux, este, presente em diversas distribuições como CentOS, Ubuntu, Red Hat. Deve-se optar pela distribuição de mais recursos para obtenção de pacotes e configuração dos mesmos [6].

A inserção de RTOS µClinux no FPGA Cyclone II da Altera, descreve-se neste artigo, com o intuito de conectar a placa DE2 na rede Ethernet para automação industrial. As conexões destas placas na rede Ethernet permitiram o

1 Gabriel Martins Buim, Universidade Católica Dom Bosco, GPEC, Avenida Tamandaré, 6000, Campo Grande, MS, gabrielbuim@gmail.com
2 João Carlos Siqueira, Universidade Católica Dom Bosco, GPEC, Avenida Tamandaré, 6000, Campo Grande, MS, joao1609@gmail.com
3 Dr. Edson Antonio Batista, Universidade Católica Dom Bosco, GPEC, Avenida Tamandaré, 6000, Campo Grande, MS, edson.ucdb@gmail.com
4 Msc. Alexsandro Monteiro Carneiro, Universidade Católica Dom Bosco, GPEC, Avenida Tamandaré, 6000, Campo Grande, MS, almocams@gmail.com
5 Dr. Mauro Conti Perreira, Universidade Católica Dom Bosco, GPEC, Avenida Tamandaré, 6000, Campo Grande, MS, mauro2@ieee.org
6 Eng. Eletr. Alexandre César Rodrigues da Silva, Univ Estadual Paulista, Campus de Ilha Solteira, Ilha Solteira, SP, acrsilva@dee.fies.unesp.br

controle e automação de diferentes processos utilizando os recursos da computação reconfigurável e conseqüentemente a potencialidade do processador NIOS II.

DESENVOLVIMENTO

Desenvolvimento do Hardware

Para o desenvolvimento deste projeto foram utilizadas três ferramentas de desenvolvimento: Quartus II 7.2 Service Pack 3 na sua versão Web Edition, Nios II 7.2 e SOPC Builder (System On a Programmable Chip). Como um sistema operacional base para configuração do RTOS, utilizou-se o Ubuntu, desenvolvido através de códigos fontes do Debian Linux.

O software de desenvolvimento de hardware Quartus II, tem sua principal linguagem de descrição de hardware o VHDL (VHSIC Hardware Description Language) e o Verilog. Ambos, possuindo o mesmo objetivo de descrever o hardware, ou seja, determinar quantidade de portas lógicas, números de entradas e saídas, interligá-los em um processador através de memórias e pré-determinar alguns processos principais que o hardware irá desempenhar. O software conta com algumas ferramentas no qual facilita o desenvolvimento do hardware. A ferramenta Mega-Functions, que são blocos prontos com funções específicas para determinadas aplicações, funcionando como um intermediário para o acesso do SOPC Builder, a ferramenta Pin Planner, onde são definidos os endereçamentos de cada componente e a ferramenta Programmer que realiza o envio do hardware finalizado ao FPGA.

No SOPC Builder, inseriu-se o driver DM9000A para o controle do protocolo da comunicação Ethenert. Este driver é o mais recomendável para desenvolver aplicações com rede Ethernet, inserido na placa DE2. Porém, para habilitar este sistema de comunicação nesta placa é necessário realizar o download do driver disponibilizado pela empresa [7], após o download. Inserir a pasta do DM9000A no mesmo local onde se encontra a pasta raiz do projeto, em seguida, podendo ser visualizado na ferramenta SOPC Builder.

Para o desenvolvimento de um hardware básico na implementação de um RTOS, é necessário a escolha de alguns componentes tais como:

- Processador Nios II - a sua escolha depende da analise do objetivo final do projeto. Pode-se utilizar dois tipos de processadores: Nios II/s (standard) e o Nios II/f (full). Recomenda-se a utilização do Nios II/f pela sua melhor compatibilidade com o RTOS. O Nios II/s tem a característica de ser mais lento na utilização de sistemas embarcados;
- Memória SDRAM - sua capacidade total de armazenamento de 8 MB;
- Timer - Habilitado no "full featured timer";
- JTAG UART (comunicação USB Blaster);
- Pino de Saída - representado por um LED vermelho, dispositivo pelo qual já está inserido na placa DE2;
- DM9000A.

Na Figura 1, constam todos os componentes inseridos no hardware desenvolvido na ferramenta SOPC Builder.

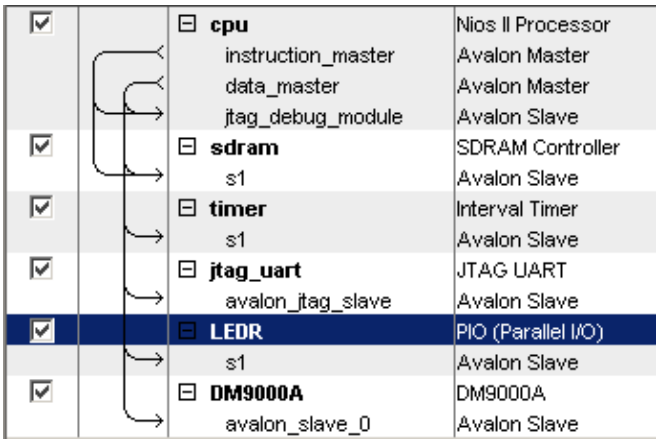

FIGURA. 1
HARDWARE DESENVOLVIDO NA FERRAMENTA SOPC BUILDER

Depois de gerado o hardware desenvolvido, é criado um bloco onde são definidas as conexões do hardware digital com o físico, e o acréscimo da mega-function "PLL" (Phase-Locked Loop), responsável pelo atraso de "-3 ns", entre o clock da memória SDRAM e o clock do processador Nios II. A inserção do Mega-Function é realizada devido o conflito de freqüências de operação entre ambos.

Nas Figuras 2 e 3, demonstram-se a configuração da PLL e o bloco do hardware com as pinagens adicionadas, respectivamente.

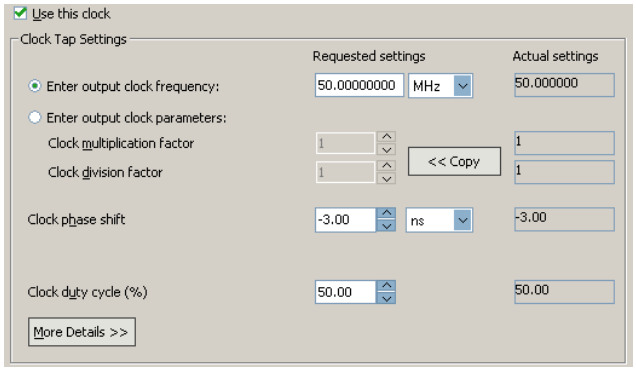

FIGURA. 2
CONFIGURAÇÃO DA PLL

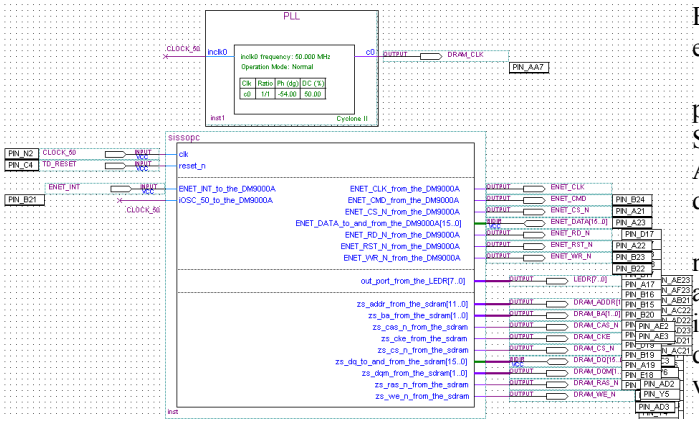

FIGURA. 3
BLOCO DO HARDWARE DESENVOLVIDO COM A IMPLEMENTAÇÂO DAS PINAGENS EA MEGA-FUNCTION INSERIDA.

Desenvolvimento do µClinux

Existem diversas distribuições µClinux onde são encontradas para download em [10], devendo-se sempre procurar a versão mais atualizada para evitar erros na implementação no hardware reconfigurável. Assim como as atualizações das ferramentas essenciais.

Utiliza-se na compilação do RTOS µClinux, o arquivo "nios2-linux-20080619.tar", encontrado para download em [10], no qual estão inclusas todas as ferramentas necessárias para realização da configuração do RTOS. Para a instalação e configuração do arquivo adquirido, são necessários possuir alguns pacotes essenciais instalados no sistema operacional Linux, podendo estes serem encontrados na referência anterior.

Após a descompactação do arquivo "nios2-linux-20080619.tar", utilizando o comando no Shell do Linux: tar xf nios2-linux-20080619.tar, é criado uma pasta nominada "nios2-linux" no local de onde foi extraído, para que os comandos executados nos passos que serão descritos não apresentem erros. Obrigatoriamente deve-se executar os comandos na pasta "nios2-linux/uClinux-dist" [6] .

Na integração do sistema embarcado com o hardware desenvolvido no SOPC Builder, deve-se integrar o arquivo gerado ao RTOS (arquivo com extensão .ptf), tarefa que pode ser realizada utilizando o comando "#make vendor_hwselect seu_arquivo.pft".

Com a execução deste comando, um modo de seleção de componentes será iniciado, dotado de três tipos de seleção. Inicialmente é definido o tipo de processador que foi determinado no hardware desenvolvido; a memória que será integrada a imagem gerada pela configuração do RTOS; posteriormente define-se o tipo de controlador de memória para realizar a execução dos processos. Nas três etapas, foram selecionados os componentes definidos no desenvolvimento do hardware utilizando a ferramenta SOPC Builder, sendo: o processador Nios II/f, o SDRAM de 8MB e o SDRAM Controller.

Na maioria dos casos são utilizados a memória FLASH para integrar a imagem gerada pelo RTOS, reservando o SDRAM Controller para a execução do Kernel µClinux. Após a realização das operações, é demonstrada a relação dos componentes selecionados anteriormente.

Através do comando "make menuconfig" é possível realizar a configuração das opções do Kernel de acordo com a necessidade do desenvolvimento do hardware e do projeto, incluindo a utilização do driver DM9000A, com as opções de uso do servidor FTP, tipos de comunicações e visualizações da execução do projeto.

A ferramenta de desenvolvimento Nios II 7.2, foi utilizada para realizar a programação de execução das funções, tais como: recebimento de um sinal de entrada; realizar o tratamento de sinais; executar processos automáticos; monitorar medições e enviá-las para o software supervisório.

Para construir a configuração optada nos passos anteriores, precisa-se executar o comando "make romfs", fazendo com que o desenvolvimento e a configuração do RTOS µClinux fique apto para a concepção da imagem na qual será integrada a memória posteriormente. Para esta etapa, necessita-se da execução do comando "make linux image", gerando uma imagem com o nome padrão de "zImage" localizado na pasta (images).

Embarcando o Sistema

Após o desenvolvimento do hardware e do RTOS µClinux, o sistema configurado está pronto para ser embarcado, sendo necessário o uso da ferramenta Quartus II e Programmer (ferramenta interna do Quartus II), responsável pelo envio do arquivo de extensão (.sof) para o FPGA, onde este arquivo detém toda a arquitetura do hardware desenvolvido.

Com o auxílio do prompt de comando do Nios II (Nios II 7.2 Command Shell), pode-se realizar o download para o FPGA da imagem gerada na configuração do µClinux, em seguida, executando o comando "nios2-terminal" o sistema embarcado será inicializado e pronto para a execução dos comandos de processos do projeto.

Os comandos utilizados para a implementação do projeto são relacionados principalmente em definição automática de IP (DHCP), Máscara de sub-rede, Gateway ou manualmente, seguidos de comandos para inicialização de servidor FTP e WEB para acesso do FPGA via browser e desempenhar as tarefas programadas no Nios II, para executar a automação desejada. Abaixo, as definições e comandos para inicializações:

- DHCP: Definição automática de IP dentro de uma rede local;
  />dhcpcd &
- IP: Endereço pelo qual o FPGA se comunica com o

software supervisório;
/>ifconfig eth0 10.1.1.2

- Máscara de sub-rede: Define em qual rede o FPGA está alocado (definido automaticamente após as definições de IP e Gateway);
- Gateway: Endereço intermediário entre o FPGA o software supervisório;
  />route add default gw 10.1.1.1
- FTP: Servidor utilizado como acesso às pastas compartilhadas do FPGA, principalmente para recebimentos de novos arquivos para execução, aquele pode ser acessado via browser ou softwares específicos para FTP.
  />inetd &

Para acessá-lo internamente apenas digite os comandos: */>ftp 10.1.1.2* seguido pelo login *"FTP"* e senha podendo ser qualquer número, no caso foi utilizado *"230"*

CONCLUSÃO

Neste artigo, pretendeu-se demonstrar as enúmeras possibilidades que o RTOS embarcado em FPGA, poderá auxiliar no desenvolvimento de um projeto com elevada complexidade, focando principalmente na utilização de protocolo TCP/IP, para substituir por completo a comunicação padrão industrial RS-232, como demonstrado na Figura 4.

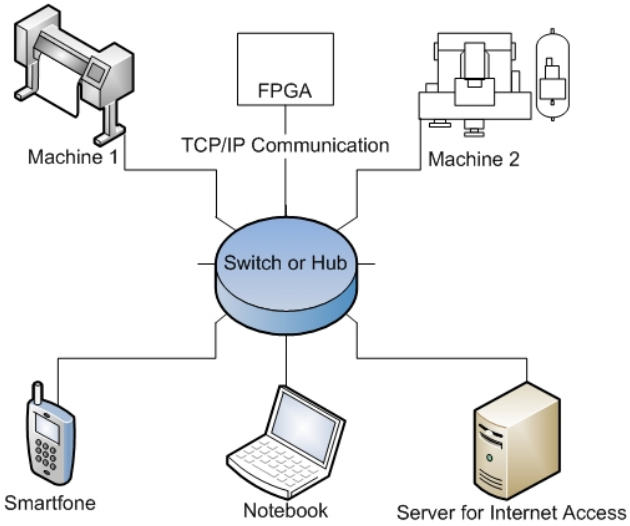

FIGURA 4
DIAGRAMA DE FUNCIONAMENTO

Algumas das dificuldades encontradas foi o fato de que o desenvolvedor do hardware e o programador, devem necessariamente possuir conhecimento da distribuição Linux, assim como, conhecimento com protocolo de comunicação TCP/IP. Espera-se aumento no tamanho da memória SDRAM, disponibilizada pelo hardware, pois assim, pode-se efetuar mais operações com apenas um dispositivo lógico reprogramável FPGA.

O principal benefício do projeto está no controle de operações e monitoramento de transdutores na indústria, com a substituição da comunicação RS-232 pela Ethernet, gerando controle num maior número de transdutores e com respectivas distâncias elevadas de comunicação.

REFERÊNCIAS

[1]Kaplan, Gadi, "Ethernet Winning Ways", *IEEE Spectrun,*Janeiro 2001, 1.

[2]Junqueira, Roberto Sant'Ana, "PDA Baseado no Microprocessador Nios II com Sistema Operacional uClinux Embarcados em FPGA",Monografia, 2007, pp. 11-22.

[3]Lutz, Philipp, "Device drivers and Test application for a SOPC solution with NiosII softcore processor and µClinux.", 2008, pp. 9-27.

[4]Sukhanov, A., Sukhanov, I., Suhyeon Kim, Shutov, A., Bazylev, S., "Online Monitoring and Remote FPGA Configuration Using JTAG Over Ethernet", Real-Time Conference, 2007 15th IEEE-NPSS, pp. 1 – 2

[5]Wang, J., Wang, H., Yang, Z., "An FPGA based slave communication controller for Industrial Ethernet, Solid-State and Integrated-Circuit Technology", 2008. ICSICT 2008. 9th International Conference on, pp. 2062 – 2065

[6]http://www.nioswiki.com/OperatingSystems/UClinux/EtherNet, acessado em 11 de mais de 2009 às 15h35min

[7]http://www.niosforum.com/, acessado em 07 de maio de às 16h12min

[8]http://www.alteraforum.com/, acessado em 12 de maio às 13h45min

[9]http://www.uclinux.org/, acessado em 19 de abril às 13h45min