



## **TECNOLOGIA EMBARCADA: PROCESSADOR NIOS II APLICADO A UM ROBÔ MÓVEL**

**Felipe Alves** – e-mail: alves.f.souza@gmail.com

Universidade Estadual Paulista – UNESP - Departamento de engenharia Elétrica  
Av. Brasil n.56 - Centro  
15385-000 – Ilha Solteira – São Paulo

**Suely Cunha Amaro Mantovani** – e-mail: suely@dee.feis.unesp.br

Universidade Estadual Paulista – UNESP - Departamento de engenharia Elétrica  
Av. Brasil n.56 - Centro  
15385-000 – Ilha Solteira – São Paulo

**Resumo:** *É crescente o interesse e o uso de tecnologias e recursos de kits de desenvolvimento nos vários trabalhos encontrados na literatura que tratam de robótica e sua aplicação para o ensino de graduação em engenharia elétrica. A utilização de sistemas microprocessados de forma popular e lucrativa aumenta a demanda por microprocessadores cada vez mais rápidos. Para algumas empresas a fabricação de novos chips e a inclusão de uma nova tecnologia, pode significar um aumento considerável nos custos da empresa, repassado depois ao consumidor. Surgem então os kits de desenvolvimento ou single board e os processadores implementados em software- softcores. Os softcores podem ser uma opção para reduzir os custos de produção e tornar mais prático e rápido o desenvolvimento de controles e processos. Neste trabalho usa-se um softcore do kit de desenvolvimento da ALTERA, o Nios II, para implementar o controle embarcado, de um robô móvel. Obtêm-se com isso, além do domínio da técnica, um sistema modular, confiável e reconfigurável.*

**Palavras-chave:** *Processador embarcado, Controle, Dispositivos Lógicos programáveis, Robôs Móveis, Softcores*

### **1. INTRODUÇÃO**

A utilização de novas tecnologias nos cursos de Engenharia Elétrica, Mecatrônica ou Ciência da Computação vai de encontro a atualização necessária para formar alunos que estejam conectados com o mercado de dispositivos e softwares, mais recentes. No contexto da globalização e competência profissional é necessário que os estudantes tenham acesso a tecnologias emergentes, através das disciplinas obrigatórias e/ou optativas da maioria desses cursos onde devem ser dadas as informações básicas necessárias nesta área, envolvendo principalmente os dispositivos programáveis (ATOCHÉ *et al*, 2008).

A evolução dos dispositivos programáveis trouxe principalmente a realização de projetos mais elaborados em circuitos digitais, oferecendo ferramentas amigáveis de síntese,



simulação e a implementação de projetos, diminuindo o tempo para o seu desenvolvimento. Hoje, os dispositivos programáveis já conquistaram seu nicho no mercado de componentes entre esses os FPGA (*Field-Programmable Gate Array*). Sua disponibilidade e suporte oferecidos pelas empresas tornaram-se uma ferramenta importante na flexibilidade de sistemas, cuja arquitetura é frequentemente atualizada. São vários os trabalhos encontrados na literatura (DO ROSÁRIO & BRANDÃO, 2010), (BROWN & VRANESIC, 2000) (DE SÁ & MANTOVANI, 2013) que usam os recursos de software e hardware programáveis.

Os dispositivos FPGAs possuem uma arquitetura interna que permite sua fácil utilização em variados propósitos. Podem ser configuradas várias vezes usando diagramas esquemáticos ou uma linguagem de descrição de hardware- HDL (*Hardware Description Language*). Com a linguagem é possível descrever um hardware digital, um microprocessador, por exemplo, e fazê-lo executar em um FPGA. Se houver necessidade de acréscimo de alguma tecnologia nova ou reparar alguma falha, é feita somente uma atualização no código HDL. Com esta tecnologia pode-se implementar em sala de aula ou em trabalhos de IC, circuitos mais elaborados e em pouco tempo.

Cresce também a procura e o interesse pelo uso de dispositivos que permitam, ao nível de hardware, alterar a arquitetura de um processador sem substituir o circuito integrado, assim como ocorre com uma atualização de um software, em um computador pessoal.

Atualmente existem vários processadores implementados em HDL como o, OpenRISC (CPU desenvolvida pela OpenCores community), Nios II, MicroBlaze, entre outros. Excluindo o OpenRISC, os outros dois processadores são de propriedade intelectual da Altera e Xilinx, respectivamente.

Ao projetista cabe utilizar as opções que o fabricante disponibiliza para adequar o seu projeto ao processador. Para o projeto ser executado em um processador também é necessário escolher um sistema operacional adequado e uma linguagem na qual o seu projeto será construído.

Apesar do grande avanço na área de computadores, as arquiteturas reconfiguráveis têm conseguido o seu espaço na engenharia, devido a flexibilidade que podem proporcionar. Em todos os estudos associados com projetos de sistemas embarcados, as tecnologias de software e hardware, particularmente, a dos processadores, apresentam uma rápida evolução. Em função dessa evolução tecnológica acelerada, torna-se muito atraente a ideia de se utilizar estruturas abertas e reconfiguráveis que possam adaptar-se às necessidades do ambiente. Estas facilidades servem de motivação para escolha do FPGA e um processador embarcado, como plataforma de hardware deste trabalho.

Sistemas de desenvolvimento baseados em computação reconfigurável apresentam, dentre outras vantagens, as características de baixo consumo, além da flexibilidade de integração, facilidade de programação e operação modular, por isso são utilizados largamente em sistemas embarcados.

O Nios II é um processador implementado em FPGA da empresa Altera® , cujas características fundamentais foram configuradas, cabendo ao projetista fazer as configurações de desempenho. A Altera® fornece os softwares Quartus® II e o NIOS®II *Embedded Design Suite*, responsáveis, respectivamente, pela configuração da camada de hardware e da camada de software do sistema Nios II.

Nesse trabalho usa-se o processador embarcado Nios II no controle e acionamento por sinal PWM, de um protótipo de um robô móvel implementado com dois motores de corrente contínua (CC) .

Para isso apresentam-se na seção dois as ferramentas e os dispositivos usados no desenvolvimento do projeto. Na seção 3, mostram-se o diagrama bloco do projeto, e as etapas seguidas no desenvolvimento da estrutura para constituir um núcleo de processamento e o protótipo do robô. Os resultados e discussões fazem parte da seção quatro, seguido pelas considerações finais e as referências.

## 2. FERRAMENTA DE PROJETO

Foi grande a evolução dos circuitos programáveis, passando por dispositivos mais simples como os PLA (*Programmable Logic Array*) até os mais sofisticados como os FPGAs (*Field Programmable Gate Array*), e CPLDs (*Complex Programmable Logics Devices*). Estes dispositivos são constituídos de blocos lógicos configuráveis – CLBs, blocos de entradas/saídas e de matrizes de chaves programáveis. Com estas características proporcionam a sua aplicação em projetos mais simples como circuitos combinacionais, somadores, contadores, multiplexadores, até um núcleo de um processador, tratamento de imagem, etc.

Na realização deste projeto utiliza-se como ferramenta um FPGA – família Cyclone II contido no kit DE2 (*Development and Education board*) da Altera® e o software de desenvolvimento, Quartus II integrado com SOPC Builder - Altera.

A placa DE2, Figura 1, contém o dispositivo FPGA Cyclone® II 2C35F672C6 (pacote de 672-pinos), mostrado na Figura 2 (ALTERA CORPORATION, 2012).

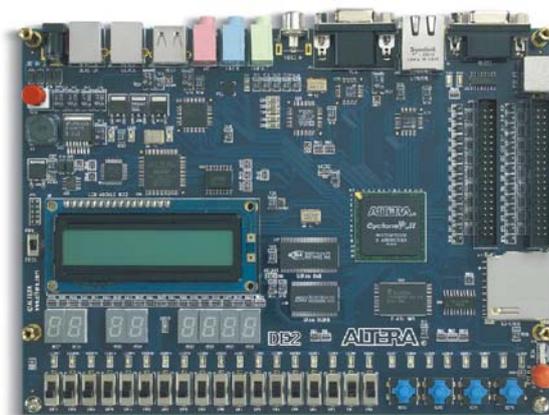


Figura 1. Placa de Desenvolvimento DE2 – Altera

Todos os componentes importantes da placa são conectados aos pinos deste dispositivo, permitindo ao usuário ter o controle completo de sua operação. Inclui um número razoável de chaves, LEDs, displays de 7 segmentos, servindo para os experimentos mais simples. Para experimentos avançados apresenta uma quantidade suficiente de dispositivos de memória SRAM (512kbyte), SDRAM(8Mbyte) e Flash(4Mbyte) e ainda um display LCD de 16X2 caracteres. Acesso ao processador Nios II da Altera e interfaces padrões simples de I/O (entrada/saída), tais como, RS-232 e PS/2. Tem conectividade para USB 2.0 e oferece a alternativa de conexão com outras placas de usuário por meio de terminais de expansão.

Depois do projeto ser programado na placa, pode-se verificar o funcionamento desejado através de seus recursos.

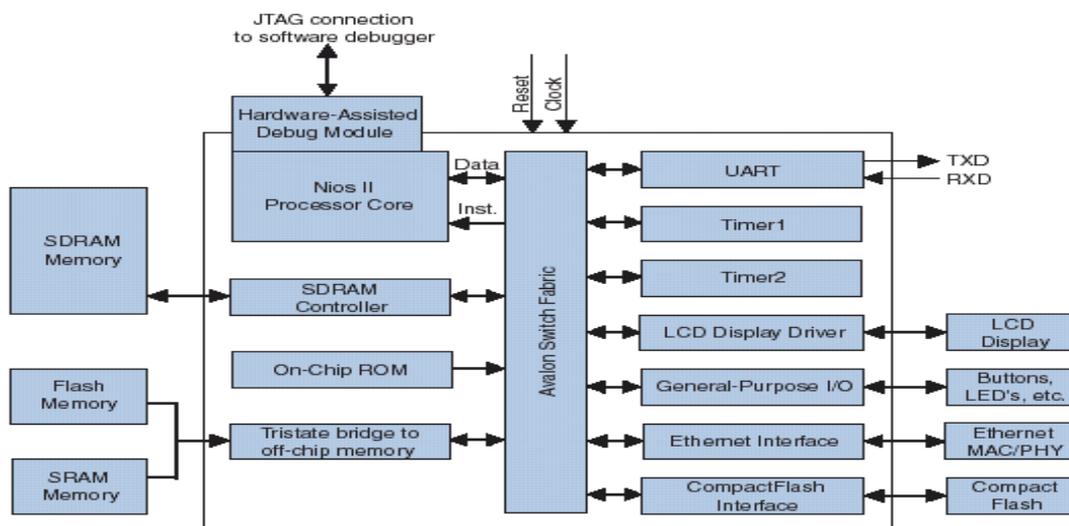


Figura 2. Arquitetura do Cyclone® II 2C35F672C6 – Altera

No ambiente de projeto, Quartus II são usados as IDEs (*Integrated Development Environment*) Quartus® II 64 bit version 9.1 e NIOS®II *Integrated Design Environment*.

O SOPC Builder é utilizado para o desenvolvimento do sistema microprocessado baseado no padrão Nios II. O computador utilizado tem que estar conectado ao kit por um cabo usb-blaster. A configuração e o sistema do processador são programados na placa. O software Nios II IDE é utilizado para o desenvolvimento do programa em linguagem C/C++ que, depois de compilado e transmitido a placa, é executado na CPU NIOS implementada no FPGA. O software gravador (integrado ao Quartus e Nios IDE) grava a configuração no FPGA via cabo usb-blaster.

Nios II Processor System, é considerado um computador em um dispositivo - “Computer on a chip”. Tem como características:

- 32 registradores de 32 bits de propósito geral;
- 32 fontes de interrupção externa habilitadas individualmente;
- Instruções dedicadas para cálculo de multiplicações de 32, 64 e 128 bits;
- Acesso a periféricos internos / externos, e mapa de endereçamento flexíveis;
- Módulo de debug JTAG;
- Interface para instruções personalizadas;
- Ambiente de desenvolvimento baseada no GNU C/C++ e no Eclipse IDE;
- Geração automatizada de sistemas (SOPC Builder).

É considerado um processador reconfigurável porque as características do sistema podem ser adicionadas ou removidas com objetivo de performance ou mesmo de custo. Possui ainda em sua arquitetura ULA, Controlador de Exceção, Controlador de Interrupção, Barramento de instruções e de dados, Memória cache de instruções e de dados, conforme se mostra na Figura 3.

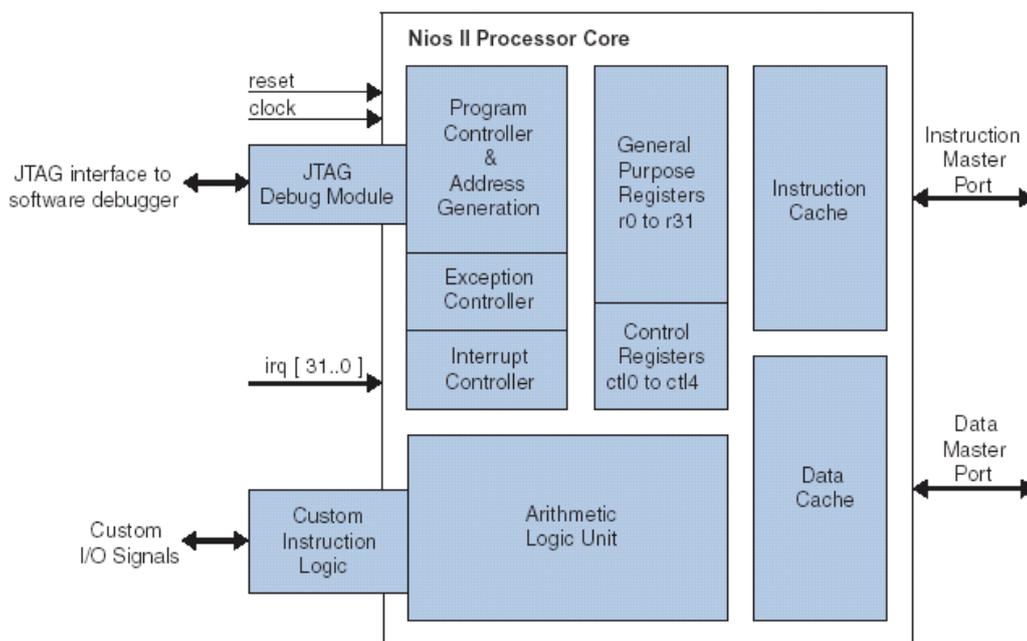


Figura 3- Arquitetura do processador Nios II

O processador Nios II é a interface necessária para conectar os outros dispositivos na DE2 implementados no dispositivo Cyclone II - FPGA. Esses componentes são interconectados por meio do barramento Avalon (Figura 2). O bloco de memória no dispositivo Cyclone II pode ser usado para fornecer uma memória on-chip.

A camada de hardware é constituída pelo processador, periféricos e a maneira como se conectam. A camada de software se constitui nos softwares que serão executados no sistema Nios II. Essa interface também serve de retorno para os resultados. Programando os sistemas em linguagem de hardware, podem-se integrar componentes como CPU's (*Central Process Unit* ou Unidade de Processamento Central), controladores de memórias, controladores de vídeo e quaisquer outros periféricos, tendo como vantagem o fato de que todos os pontos importantes das placas já estão previamente conectados e descritos, podendo ser utilizado em diversos projetos.

Os dispositivos FPGA por enquanto não substituem os processadores atuais, mas cresce a sua utilização como co-processadores, aplicados na aceleração de hardware e processamento paralelo, como processamento de imagens, DSP, criptografia, compressão de arquivos e diversas outras aplicações. Apresenta como limitação frequência de operação baixa, obtendo larga vantagem na implementação de algoritmos de tarefas executadas em modo paralelo. Destinadas a aplicações industriais, placas mãe tem sido implementadas com chips Intel Atom e FPGA Altera (DELAI, 2013), (DO ROSÁRIO & BRANDÃO, 2010)

No trabalho, além dessa plataforma, são usados componentes discretos como, motores CC e seus dispositivos de acionamento ou *drivers* dos motores- ponte H, sensores ópticos e seu condicionamento, baterias, etc.

### 3. DESENVOLVIMENTO DO PROJETO

O projeto completo pode ser representado por um diagrama de blocos, conforme mostrado na Figura 4 onde se observa o controlador (processador Nios II) que atua no motor, usando para isso, as informações provenientes da realimentação dada pelo *encoder*. Para a realização do trabalho descreve-se a seguir, os detalhes de configuração do processador, assim como o protótipo completo (Figura 5) e as demais placas, visando a sua implementação.

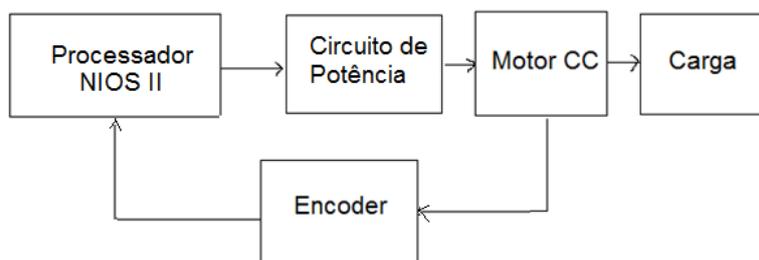


Figura 4. Diagrama de Blocos do projeto.

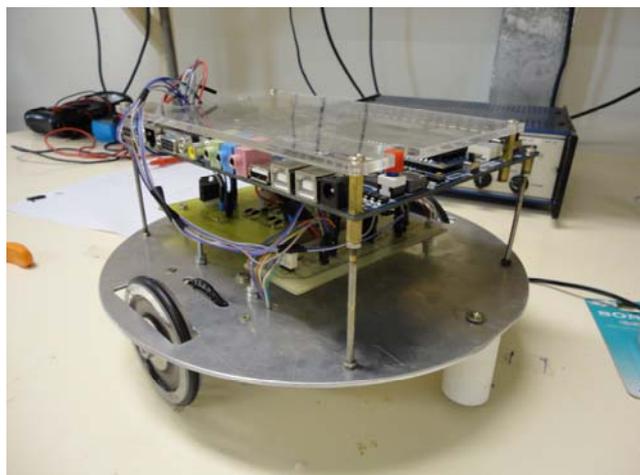


Figura 5. Protótipo Completo para o robô

No protótipo do robô foi usada uma placa leve de alumínio, com mobilidade dada por duas rodas feitas de acrílico e acabamento emborrachado, e uma roda boba para sua estabilização. Em cada roda são colocados dois motores CC com redução dada por 25:1, tensão de alimentação típica de 6V e 470 RPM.

#### 3.1. Processador Nios II

Para configurar o FPGA e o processador é necessário projetar uma camada de hardware onde serão definidos os periféricos do processador, o próprio processador e qualquer outro dispositivo ou circuito lógico combinacional necessário ao projeto.

Com a camada de hardware pronta, é necessário montar a camada de software onde é definido o sistema operacional ( Micro C/OS-II é padrão, sendo apenas necessário selecioná-lo na opção de RTOS- Sistema operacional de tempo real) e o programa na linguagem C que é executado pelo processador.

A Altera fornece os softwares Quartus II e o NIOS II *Embedded Design Suite*, responsáveis, respectivamente, pela configuração da camada de hardware e *software* do sistema microprocessado.

Portanto, os programas executados no processador constituem a camada de software. A camada de hardware é constituída pela ligação dos periféricos ao processador. Neste trabalho o processador Nios II gera o sinal PWM (*Pulse Width Modulation*) programado na placa de desenvolvimento DE2 , para o controle de velocidade e acionamento de um motor CC. Um *encoder* óptico fornece a realimentação necessária, conforme diagrama de blocos do projeto na Figura 4.

### 3.2. Encoder óptico

Para controlar a velocidade do robô foi implementado um *encoder* de posição incremental, de 50 mm de diâmetro, Figura 6 a. Neste *encoder* observam-se dois anéis de informação: o anel externo para a leitura da velocidade, dimensionado para dar uma resolução de 32 pulsos/volta e o anel de furos interno para detectar a direção do movimento, mas defasados de 90 graus. Na Figura 6b o disco incremental é acoplado ao eixo de um dos motores. Este mesmo esquema foi feito para o outro motor. O *encoder* óptico implementado junto com sensores interruptores de feixe IR (*Infrared Sensitive Photoresistor*) constituído por um sensor dotado de um diodo emissor de luz e um transistor bipolar de junção, cuja a base é sensível ao sinal luminoso do diodo, gerando um sinal pulsado, tratado pelo processador Nios II. O sensor usado no projeto é o TCST1103 (BALBINOT & BRUSAMARELLO, 2007).

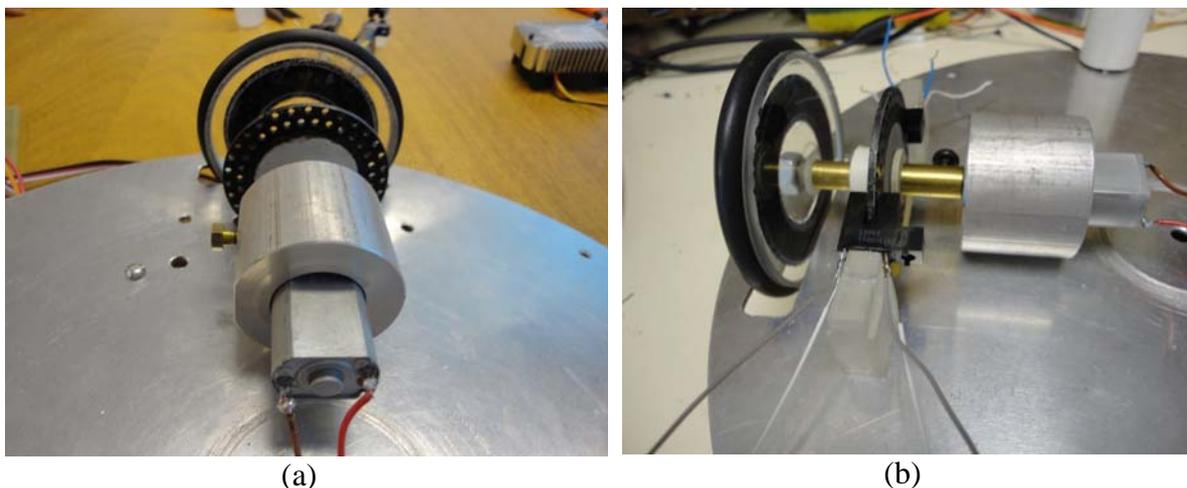


Figura 6. Montagem do *encoder* óptico para uma roda

### 3.3. Drive para os Motores CC

O circuito integrado L298 constituído de duas pontes H é usado no acionamento para os motores CC, junto com quatro diodos roda livre para proteção do retorno da corrente. Uma palavra de dois bits (uma para cada circuito H) controla o sentido de giro do motor, Figura 7.

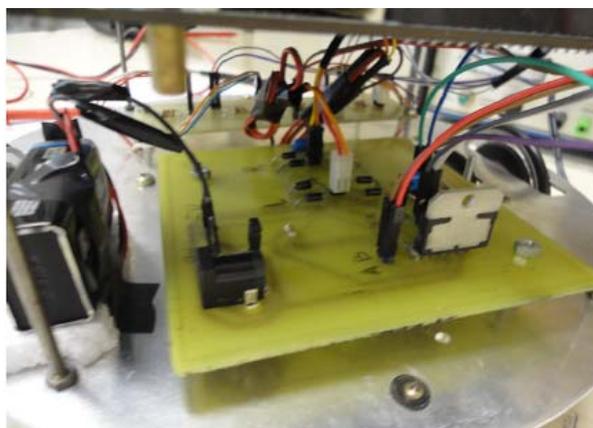


Figura 7- Drive para os dois motores CC (DE SÁ & MANTOVANI, 2013).

### 3.4. Fluxogramas

Apresenta-se na Figura 8, o fluxograma geral do programa para a camada de hardware. Este fluxograma foi dividido em tarefa 1 que executa durante três segundos um sinal de PWM com *duty cycle* selecionado via chaves de entrada (pelo usuário) e tarefa 2, que segue à tarefa 1 depois desse tempo e executa o cálculo da velocidade em RPM e mostra no display LCD esta informação.

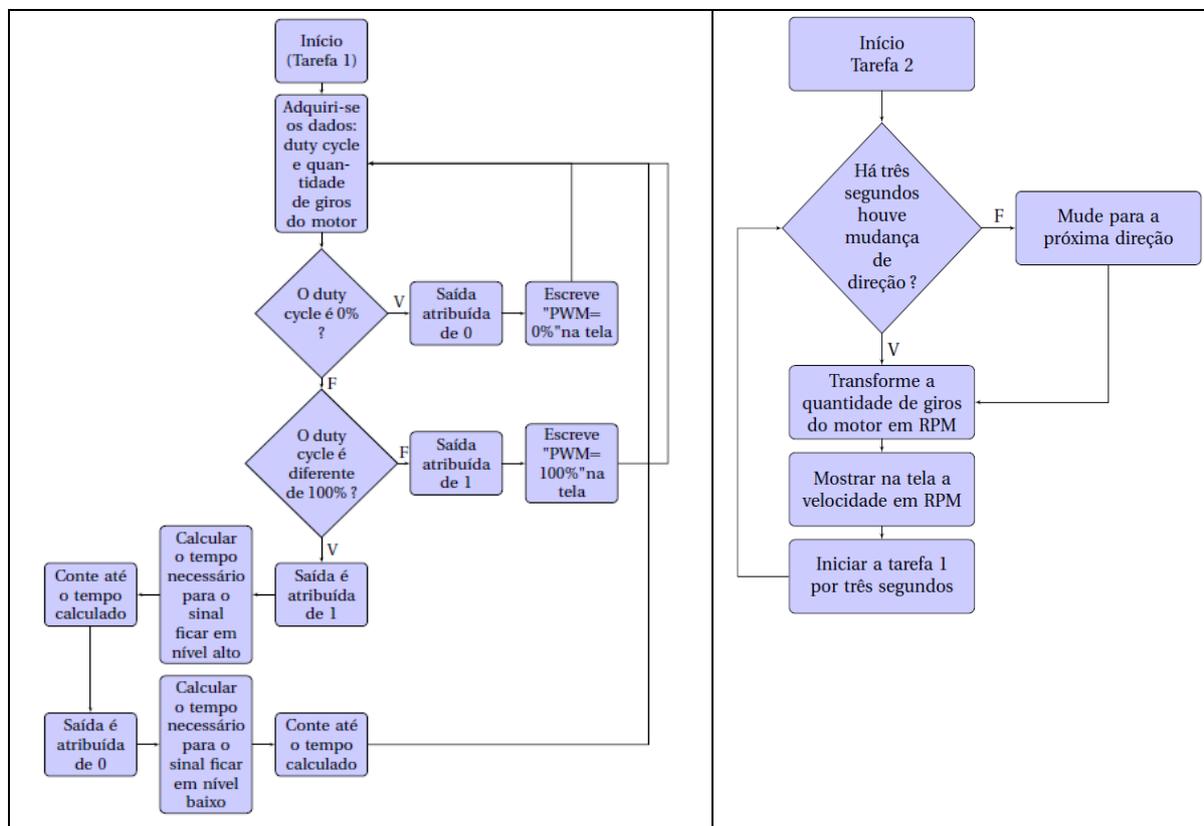


Figura 8 - Fluxograma da camada de software - tarefas 1 e 2

#### 4. RESULTADOS E DISCUSSÕES

Depois de projetado e programado na placa DE2, pode-se verificar o funcionamento do projeto, conforme se observa na Figura 9, com uma sequência de movimentos do robô, girando no sentido horário e depois se movimentando para frente.

Verifica-se na prática, que em vários trechos o sinal PWM pode estar mais tempo do que o normal em nível alto ou em nível baixo. Isso ocorre por que a cada 3 segundos a tarefa 1, responsável pela geração do sinal, é interrompida em favor da tarefa 2, responsável pelo controle geral do robô móvel.

Percebe-se também, que o sinal não possui exatamente o *duty cycle* selecionado e nem a frequência desejada. Isso ocorre por que cada instrução do software em linguagem C leva algum tempo para ser executada, ou seja, não é instantânea. Outra observação é que no código em C considera-se, de maneira aproximada, que o tempo necessário para o programa contar de 0 à 2000000 é de 1 segundo.

Como a tarefa 2 acontece em intervalos de três segundos existe um *delay* para a resposta da velocidade na tela do LCD. Esses dados são monitorados pela tela do LCD, conforme pode ser observado na Figura 9.

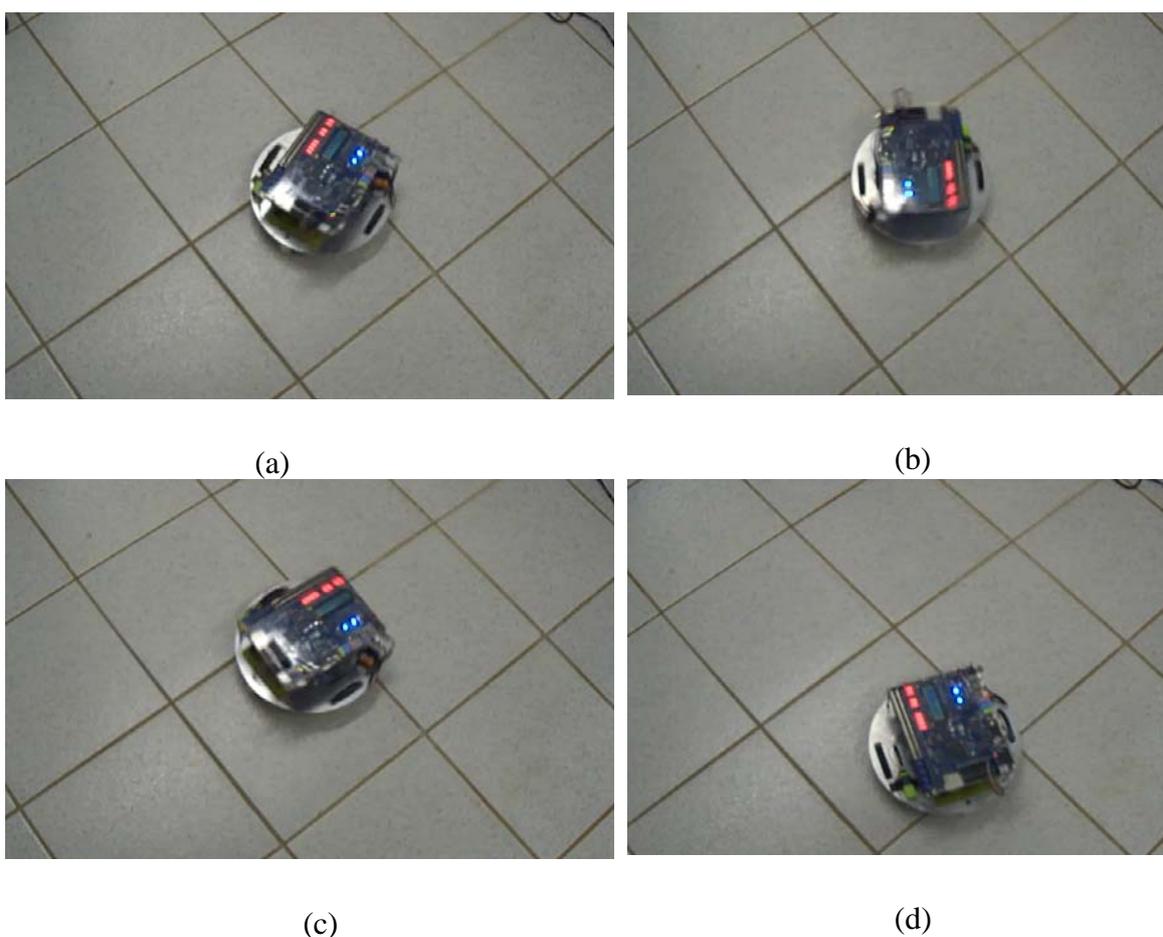


Figura 9- Sequência de movimentos do robô. (a) Início do movimento giratório. (b) Meio do movimento giratório. (c) Fim do movimento giratório. (d) Início do movimento para frente.

A alimentação de seus circuitos é feita através de baterias, preferencialmente baterias recarregáveis. Para o funcionamento do robô autônomo, um aspecto importante diz respeito a dimensionar a bateria que forneça a tensão e corrente suficientes para a alimentação dos circuitos de interface e ao circuito do kit DE2. As principais características a serem observadas na escolha de baterias são a tensão e a corrente nominais. A tensão nominal, medida em volts, depende do valor necessário para alimentar cada um dos circuitos do robô. A escolha da corrente nominal da bateria depende do consumo total do robô, e o tempo que ele deve ter de autonomia. Foi usada no projeto, uma bateria de 9V, 4500mA/h para a alimentar a placa e uma bateria de 9V, 300mA/h, para alimentar o restante do circuito .

## 5. CONSIDERAÇÕES FINAIS

Neste trabalho foi realizado um projeto embarcado para o controle de um protótipo de um robô móvel com motores CC e sinal PWM, utilizando o kit DE2 e o processador Nios II da Altera.

No trabalho desenvolvido vários aspectos podem ser observados quando se usa o processador Nios II de forma a atingir um melhor desempenho na realização deste projeto. Um deles diz respeito à natureza do algoritmo da camada de software que requer algoritmos mais elaborados para o controle preciso do dispositivo móvel. Outro aspecto seria a transferência da geração do sinal PWM da camada de software para a camada de hardware, melhorando sua precisão.

No projeto o sinal PWM gerado pelo programa em linguagem C é executado pelo processador que controla (ao mesmo tempo) a potência dos dois motores CC com pequena margem de erro.

Portanto, é possível controlar a potência enviada ao motor e a trajetória do robô, utilizando o *softcore* Nios II que executa um programa em linguagem C.

No desenvolvimento do projeto são abordados variados conceitos e tecnologias importantes na área de engenharia elétrica que são vivenciados na atualidade pela maioria dos alunos no curso de graduação, na Iniciação científica ou em trabalhos de final de curso.

### *Agradecimentos*

Agradecemos aos técnicos do DEE-FEIS-UNESP pela confecção do protótipo.

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

ATOCHE, A. A. Castillo; CRUZ, C. Rodríguez; AGUILAR, J. S. Ortégón; CASTILLO, J.Vázquez. Laboratory Projects for Engineering Students with FPGA's. IEEE Latin America Transactions, vol. 6, n. 2, p. 130-136, jun. 2008.

ALTERA CORPORATION. (s.d.). **DE2 Development and education board**. Disponível em: <http://www.altera.com/education/univ/materials/boards/de2/unv-de2-board.html>. Acesso em: 05 nov. 2012.

BALBINOT, Alexandre; BRUSAMARELLO, Valner João. Instrumentação e Fundamentos de medidas vol.2. LTC,-Rio de Janeiro, 2007. 658 p.



BROWN, Stephen; VRANESIC, Zvonko. Fundamentals of digital logic with VHDL design. 2. ed . McGraw-Hill Higher Education, 2005. 939 p, il.

DE LAI, A. L. (s.d.). Softcores em FPGAs: microprocessadores do futuro? Guia do hardware. Disponível em:<http://www.hardware.com.br/artigos/softcores-fpga/>. Acesso em: 05 jun. 2013.

DE SÁ, Daniel Alves; MANTOVANI, Suely Cunha. UNIVERSIDADE ESTADUAL PAULISTA-FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA-SP. Construção e Controle de um Robô Móvel com Motor de Corrente Contínua e FPGA, 2013. 79 p, il. Monografia.

DO ROSÁRIO, Desnes Augusto Nunes; BRANDÃO, Gláucio Bezerra. UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE-NATAL. Desenvolvimento e aplicações de sistemas embarcados em FPGA, 2010. 61 p, Il. Monografia.



## **EMBEDDED TECHNOLOGY: NIOS II PROCESSOR APPLIED TO A MOBILE ROBOT**

**Abstract:** *Several papers in the literature present a growing interest and use of technologies and resources based on development kits whenever the robotics and its utilization are dealt in the electrical engineering undergraduate education. Nowadays, the demand for microprocessors has rapidly increased due to their use in popular and profitable way. However, the manufacturing of new microchips and the inclusion of a new technology can represent a significant growth in the production cost for some companies that pass it to the consumer. In this way, the development kits, or single boards, arise in the electronics market as well as the microprocessors implemented on softcores, which can be an important alternative to reduce the production costs and make more practical and rapid the development of controllers and procedures. In this paper, it is employed a softcores from the Altera development kit, named Nios II, in order to build the embedded controller of a mobile robot. Therefore, a modular, reliable and reconfigurable system is obtained in addition to the technical learning.*

**Key-words:** *Embedded Processor, Control, Programmable Logic Devices, Mobile Robots, Softcores*