

DESENVOLVIMENTO DE UM SISTEMA BASEADO EM BLOCOS PARA PROGRAMAÇÃO INTUITIVA EM MICROCONTROLADORES

**Celso Eduardo dos Santos¹, José Luis Lourenço²,
Alderico R. de Paula Júnior³, Luis Filipe Wiltgen Barbosa⁴**

^{1,2,4}LRA/FEAU/UNIVAP – São José dos Campos – SP

¹eduardo_sme@yahoo.com.br, ²zezons@hotmail.com e ⁴wiltgen@univap.br

³IP&D/UNIVAP – São José dos Campos – SP
alderico@univap.br

Resumo – Este artigo apresenta o desenvolvimento de um sistema informatizado, utilizando uma linguagem de programação de alto nível, no qual o usuário interage de forma prática, conseguindo programar um microcontrolador utilizando ferramentas intuitivas. Essas ferramentas intuitivas se utilizam de blocos e estes por sua vez são formados principalmente pela representação de cada um dos componentes do circuito. Paralelamente, também foi desenvolvido um circuito elétrico de controle de entradas e saídas de sinais, utilizando um microcontrolador da família PIC, para interação com o sistema de programação intuitiva a fim de realizar os testes experimentais.

Palavras-chave: Programação intuitiva, microcontroladores, robótica

Área do Conhecimento: III – Engenharias

Introdução

Um microcontrolador (também denominado MCU ou μC) é um tipo de computador em uma pastilha, contendo um processador, memória e funções de entrada/saída, além de integrar elementos tais como memória RAM, EEPROM ou Memória Flash para armazenamento de dados ou programas ([MARTINS, et al., 2005](#)).

Os microcontroladores PIC são uma família de microcontroladores fabricados pela *Microchip Technology*, que processam dados de 8, 16 e 32 bits, com extensa variedade de modelos e periféricos internos, com arquitetura Harvard e conjunto de instruções RISC, com recursos de programação por Memória Flash, EEPROM e OTP ([PEREIRA, et al., 2000](#)).

Os PICs podem ser programados em linguagem mnemônica (Assembly) ou usando-se compiladores de linguagem de alto nível (Pascal, C, Basic) que geram um código em formato hexadecimal (Intel Hex format ou linguagem de máquina) que é utilizado para ser gravado na memória de programa desses microcontroladores. Para tal procedimento, utiliza-se um dispositivo especial (gravador) acoplado a um computador.

A proposta visa tornar mais prática e fácil a programação de microcontroladores através de um sistema informatizado e interativo, que até uma pessoa com pouca, ou nenhuma, experiência possa utilizá-lo. Além disto, esta pesquisa visa o desenvolvimento de uma placa em circuito impresso para o controle, interação e testes do sistema.

Desenvolvimento do Sistema Intuitivo

A idéia consiste em aperfeiçoar o desenvolvimento de programas para microcontroladores utilizando recursos computacionais intuitivos e integrados. Isto é uma analogia a idéia da própria evolução das linguagens de desenvolvimento de sistemas.

No passado, a base da programação era em linguagem de baixo nível. Atualmente, têm-se programas com interfaces de desenvolvimento visuais, e até ferramentas “Cases”, no qual basta dar as diretrizes e o próprio programa desenvolve a “programação” propriamente dita. A programação intuitiva é definida como uma programação na qual não é necessário um aprofundamento no entendimento da linguagem utilizada.

Neste caso específico, o sistema foi baseado em blocos, de forma que cada bloco representa um determinado componente de circuito. Assim sendo, cada bloco possui características próprias que representam os recursos de um determinado componente, porém de forma gráfica.

O sistema está sendo desenvolvido em linguagem de programação C#. A linguagem C# foi criada pela *Microsoft*[®] em conjunto com a arquitetura *.NET*. Esta linguagem foi escolhida porque, além de ser uma linguagem fácil de aprender e de utilizar, é robusta e possui bom desempenho (Visual C# Developer Center).

Foram construídas várias telas no aplicativo *Microsoft Visio*® para facilitar o desenvolvimento deste sistema.

Na Figura 1 pode ser observada a configuração de uma das telas de desenvolvimento interativo proposto para este sistema. Nela é possível visualizar a representação dos blocos e suas ligações, e também os menus de forma expandida, no qual pode ser visto em detalhes as opções do programa.

A Figura 2 apresenta outra forma de ver a programação do sistema. Através dos blocos de códigos pode-se observar o fluxo de execução do código. Cada bloco representa um grupo de códigos específicos, pré-programados, ou configurações de dispositivos, as quais ficam invisíveis ao usuário.

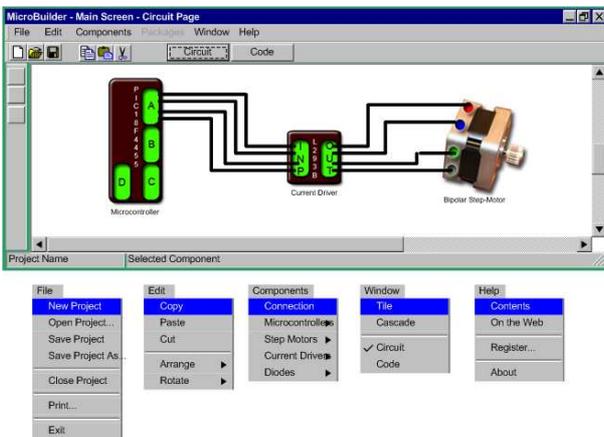


Figura 1 - Tela interativa do sistema de programação em blocos

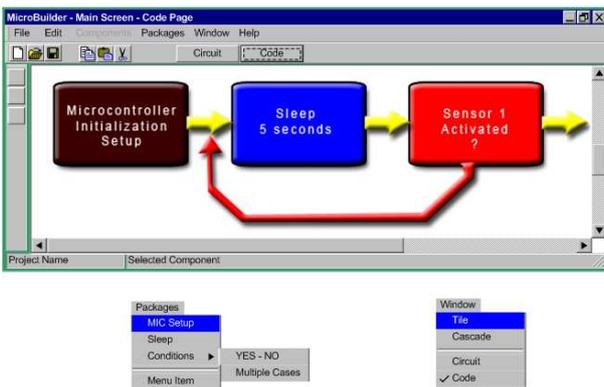


Figura 2 - Tela interativa do sistema de programação em blocos de código

Também foram construídos os diagramas de casos de uso do sistema como um todo. Conforme pode ser observado na Figura 3.

Cada componente representado pelo sistema está sendo implementado através da criação de Bibliotecas de Ligação Dinâmica – *DLL (Dynamic-Link Library)*.

As *DLL's* são formatos de arquivos executáveis para Sistema Operacional Windows® da Microsoft®, que pode conter os códigos, os dados, e os recursos (ícones, fontes, cursores, entre outros) em qualquer combinação.

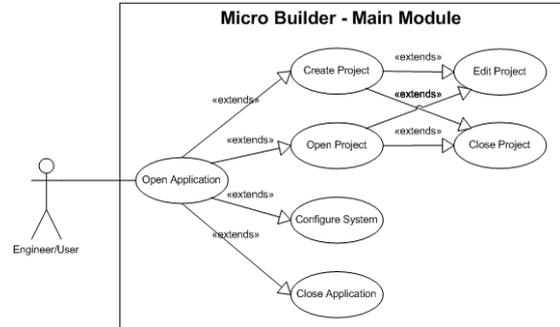


Figura 3- Diagrama de caso de uso (Principal)

Através da utilização de *DLL's*, é possível obter a modularidade necessária para este sistema. Desta forma, o usuário poderá instalar apenas os componentes que irá utilizar, novos componentes podem ser incluídos e correções para os componentes existentes poderão ser feitas sem que haja qualquer modificação no aplicativo.

A Figura 4 mostra os componentes incluídos pela seleção no menu na área de trabalho.

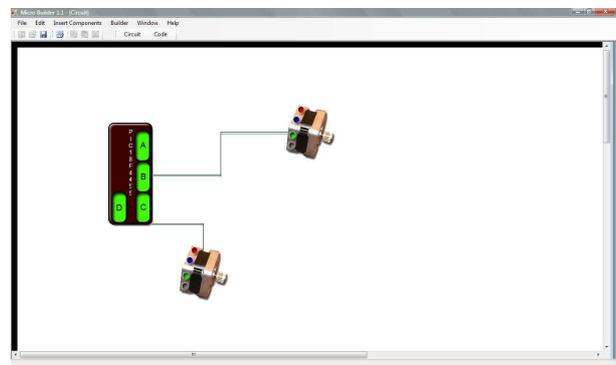


Figura 4 - Componentes inseridos na área de trabalho

Foi desenvolvido um algoritmo para melhor desempenho e correção de *flickers*, ou seja, aumentar a frequência de atualização da imagem imperceptível a olho humano, evitando a sobreposição de imagens como elas estivessem piscando.

Este algoritmo manipula as imagens diretamente da memória RAM, antes de enviar para a memória de vídeo do monitor, eliminando assim o tempo de acesso ao hardware. Conseqüentemente, ao mover um objeto (bloco), as imagens ficam mais nítidas, evitando o cansaço excessivo dos olhos (SANCHEZ, et al., 2003).

O processo para programação intuitiva inicia-se com o carregamento dos blocos necessários. Os

mesmos são interligados de acordo com o circuito projetado. É realizada então a configuração desejada de cada bloco. Feito isso, é gerado automaticamente o código em blocos, o qual é passível de modificações manuais. Através de execução deste comando é gerado, o código para o microcontrolador, que pode ser em linguagem *assembly*, ou diretamente em formato padrão hexadecimal (utilizado pelo gravador de microcontroladores).

O sistema é composto tanto pelo programa quanto pelos dispositivos físicos (placas de circuitos) necessários para a interação do programa com o microcontrolador. A parte física do sistema é dividida em vários módulos.

Inicialmente têm-se cinco módulos, divididos da seguinte forma:

- *Módulo do microcontrolador;*
- *Módulo de motores;*
- *Módulo de sensores;*
- *Módulo Serial/USB;*
- *Módulo de testes de portas.*

Todos os módulos são intercambiáveis e divididos de forma adequada às portas lógicas do microcontrolador, ou seja, podem ser utilizados vários módulos ao mesmo tempo.

Os módulos de sensores e Serial/USB estão fisicamente juntos em uma mesma placa de circuito impresso, isso devido a simplicidade dos circuitos. A Tabela 1 mostra as dimensões dos módulos.

Tabela 1 – Dimensões dos Módulos

Módulo	Largura (mm)	Comprimento (mm)
Microcontrolador	111	116
Motores	111	163
Sensores – Serial/USB	111	116
Testes de portas	111	163

Estão sendo utilizados como microcontroladores, os PIC's *16F877A* e *18F4455*, o primeiro apenas com interface de comunicação serial e o segundo com interface de comunicação *USB*.

Para facilitar, foi introduzida no módulo do microcontrolador a gravação *"in circuit"*. Assim, não é preciso remover o microcontrolador toda vez que for gravá-lo (novo firmware), bastando apenas conectar o mesmo ao gravador.

O gravador utilizado foi o ICD2Br da *LabTools* (*Mosaico Didatic Division*) para a gravação de modelos de 8 a 40 pinos para os microcontroladores da família PIC ([Divisão Didática da Mosaico](#)).

A conexão *in circuit* segue o padrão da *LabTools*, através de um conector RJ12.

A Figura 5 apresenta a configuração do circuito do módulo do microcontrolador, que é composto por um soquete com trava (*Zif*) para o microcontrolador, um conector de 40 vias para a conexão entre os módulos, um conector RJ12 para gravação *in circuit*, um led para indicação de funcionamento, um cristal para oscilação, um regulador de tensão (5V), resistores, capacitores, diodos e conector da fonte. As linhas vermelhas indicam as "pontes", que foi preciso devido à utilização de placa de fenolite com apenas uma camada.

O módulo dos motores é composto de dois *drivers* de corrente *ULN2003* para utilização de motores de passo unipolar e servomotores, dois *drivers* *L293B* para utilização de motores CC e de passo do tipo bipolar, um conector de 40 vias para a conexão entre os módulos, quatro botões para acionamento dos *drivers*, um regulador de tensão (5V), resistores, capacitores, diodos e conectores de fontes, um para alimentar os *ULN2003* e outro para os *L293B*.

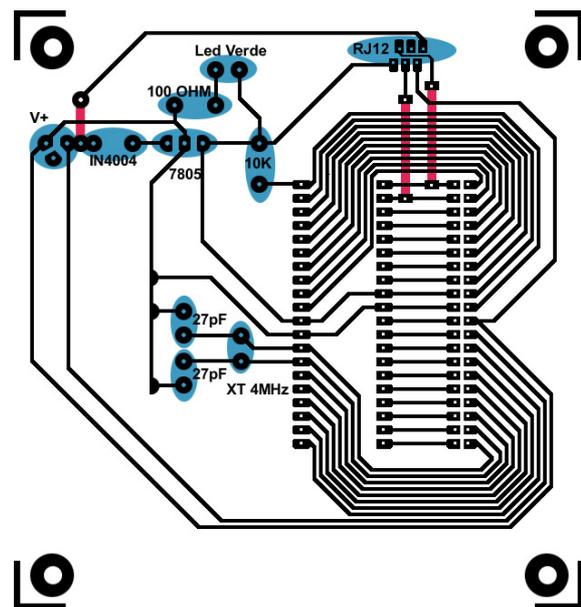


Figura 5 - Layout da placa de CI do módulo do microcontrolador

Resultados

Na primeira fase do projeto, o programa do sistema está estruturado, entretanto, continua em desenvolvimento.

Os testes para carregamento dinâmico das bibliotecas dos componentes atenderam o esperado, ou seja, facilitou a incorporação e/ou atualização dos componentes.

O processo de correção de *flickers* foi de muita importância para qualidade visual do programa, promovendo um aspecto profissional a aplicação.

A configuração da placa de circuito do módulo do microcontrolador está pronta confeccionada, esta foi montada e está funcionando adequadamente.

A Figura 6 apresenta uma fotografia da placa do microcontrolador com todos os componentes soldados para os testes funcionais.

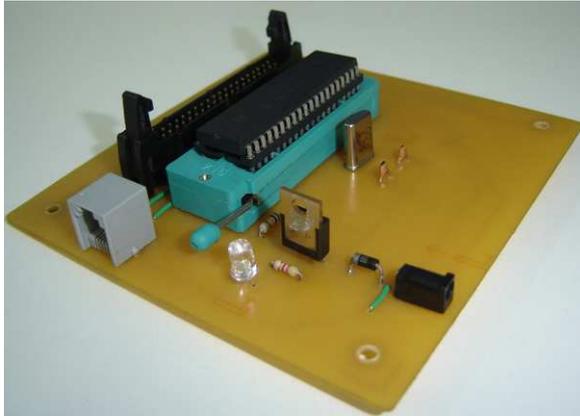


Figura 6 - Foto do módulo do microcontrolador em teste

Discussão

Hoje no mercado existem produtos similares, porém, utilizados em pequenas aplicações e produtos fechados, ou seja, desenvolvido para algo específico. Um exemplo é o sistema utilizado pela LEGO®, para atingir principalmente o público mais jovem.

No que diz respeito aos módulos, temos diversas opções no mercado, porém foi criado um conjunto de módulos próprios para testes e que pode ser utilizado em diversas aplicações de forma muito ampla.

O diferencial está realmente na programação intuitiva e em blocos, ponto chave do desenvolvimento e que torna a aplicação bastante interessante e inovadora.

Conclusão

Embora a aplicação do sistema apresentada neste trabalho seja restrita, à programação dos microcontroladores PICs permite concluir que o programa desenvolvido neste trabalho é uma forma alternativa válida para realizar tal tarefa, apresentando como principais vantagens a transparência do código-fonte, a fácil utilização, e, conseqüentemente, o uso deste por pessoas que não tem muito conhecimento em linguagens de programação de firmwares em microcontroladores.

A estrutura modular do programa permite que sejam desenvolvidos blocos para programação em

outros microcontroladores, permitindo desta forma uma generalização da aplicação.

Referências

- Divisão Didática da Mosaico. Disponível em: <http://www.labtools.com.br/>. Acesso em: 02 abr.2008.
- MARTINS, N. A. Sistemas Microcontrolados. ed. Novatec, 2005.
- PEREIRA, F. Microcontroladores PIC: Programação em C. ed. Érica, 2000.
- SANCHEZ, D., DALMAU, C.. Core Techniques and Algorithms in Game Programming. ed. New Riders. p. 779 a 801, 2003.
- Visual C# Developer Center. Disponível em: <http://msdn.microsoft.com/enus/vcsharp/aa336809.aspx>. Acesso em 29 mar.2008.