



INTERFACE GRÁFICA PARA COMUNICAÇÃO COM MICROCONTROLADORES

Murillo Vilela Magan¹; Glauber Cabral de Souza²; Emerson Charles Martins da Silva³

RESUMO: A proposta prevê a criação de um software genérico para supervisionar o controle de aplicações de pequeno porte. A aplicação inserida no software será regida por um microcontrolador e seus periféricos, além de ser comandada e supervisionada, através da porta USB, pela interface do software criado. Sua principal característica está no baixo custo da implantação do software em qualquer sistema, pois se trata de um método gratuito.

PALAVRAS-CHAVE: Microcontrolador; Interface gráfica; software controlando hardware.

1 INTRODUÇÃO

Para Zanco (2007), nos últimos anos, vem crescendo assustadoramente a utilização de microcontroladores na solução de problemas nas mais diversas áreas, tais como: automobilística, predial, industrial, agrícola, biomédica, robótica, etc. Por causa deste crescimento, tem-se verificado que existe um grande interesse das empresas de desenvolvimento de projetos eletrônicos pelos microcontroladores.

De acordo com Rosário (2005), os sistemas supervisórios são sistemas responsáveis pelo monitoramento de variáveis de controle do sistema, com objetivo principal de fornecer subsídio ao operador para controlar ou monitorar um processo automatizado mais rapidamente, permitindo a leitura das variáveis em tempo real e o gerenciamento e controle do processo automatizado. O sistema de supervisão atua da seguinte forma: os dispositivos automatizados são monitorados e podem sofrer intervenções de computadores com funções lógicas pré-programadas ou de controladores quaisquer (homem), e com o avanço da tecnologia, os computadores assumiram o papel de gerenciadores dos dados adquiridos nos processos industriais, podendo assim, tratá-los da forma que o usuário preferir.

Esta proposta visa à criação de um software supervisório para que um usuário, com poucas instruções, possa visualizar o estado de uma ação realizada ou fazer com que o microcontrolador realize uma função a partir de comandos em um computador, ou seja, um software que supervisione a aplicação a ser desenvolvida.

¹ Acadêmico do Curso de Engenharia de Controle e Automação do Centro Universitário de Maringá - UNICESUMAR, Maringá - Paraná. Bolsista do Programa de Bolsas de Iniciação Científica do Unicesumar (PICC). murillo_magan@hotmail.com

² Acadêmico do Curso de Engenharia de Controle e Automação do Centro Universitário de Maringá - UNICESUMAR, Maringá - Paraná. glauber_cabral@hotmail.com

³ Orientador, Professor Mestre do Curso de Engenharia de Controle e Automação do Centro Universitário de Maringá - UNICESUMAR, Maringá - Paraná. emerson.charles@cesumar.br

Usando a ferramenta de programação e criação de softwares Microsoft Visual Studio: C# e a Família de microcontroladores PIC ou Atmega, sendo estes programados em Assembly, C ou C++. Será proposta a comunicação entre estas duas ferramentas, virtual e física.

2 MATERIAL E MÉTODOS

Inicialmente foi utilizada a ferramenta de desenvolvimento Microsoft Visual Studio IDE, no qual foi criada uma tela de controle do sistema, que será exemplificada por uma tela de Comunicação Serial que envia e recebe dados de um microcontrolador, através da porta USB. Dentro dessa tela foram adicionados botões que iniciam e encerram a comunicação entre o software e o microcontrolador, e também enviam os dados desejados ao microcontrolador, que irá interpretá-los e, com isso, ocorrerá às devidas ações já programadas.

Existem, dentro da janela de Comunicação Serial, locais onde são lidos e escritos os dados enviados, ou recebidos, que receberam uma programação, em linguagem C#, para realizarem as funções desejadas.

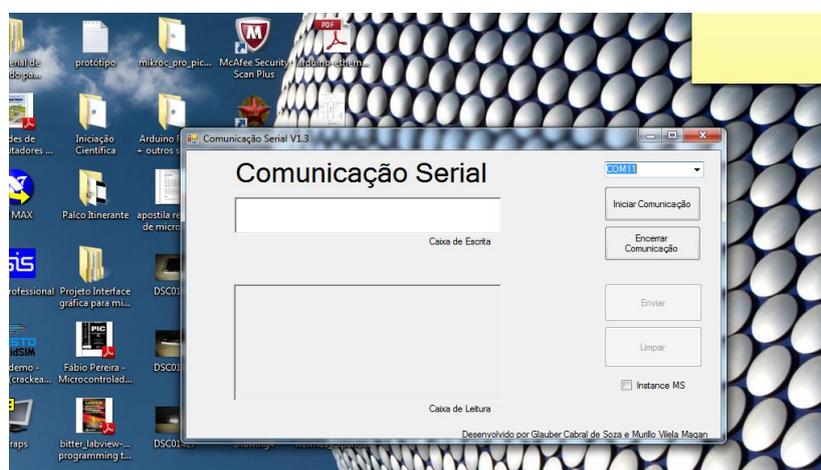


Figura 1: Aparência da Interface de Comunicação Genérica

Atualmente, o software está construído de uma maneira genérica, permitindo que o usuário possa enviar e receber manualmente os bits, e selecionar a porta COM desejada, podendo então ser utilizado para várias aplicações.

Em seguida, utilizando uma plataforma para estudos com microcontroladores conhecida como Arduino, realizamos uma programação que tem o objetivo de resgatar o bit que chega até ele, pela porta serial USB, e destinar este bit a um pino de saída do microcontrolador Atmega, podendo assim, acender ou apagar um LED, ligar ou desligar um motor, polarizar ou despolarizar um transistor, entre outras aplicações. Esta programação também é responsável por “capturar” um sinal que chegue até um dos pinos de entrada de uma porta especificada, e enviá-lo ao software situado no computador.

3 RESULTADOS E DISCUSSÃO

Neste estudo foi elaborado um software genérico, podendo ser usado em outras diversas aplicações, que tem a capacidade de enviar e receber um bit através de uma das

portas seriais do computador. A partir desse controle, o usuário pode definir o que deseja fazer com o software, ou ainda podemos construí-lo a partir de aplicação específica.

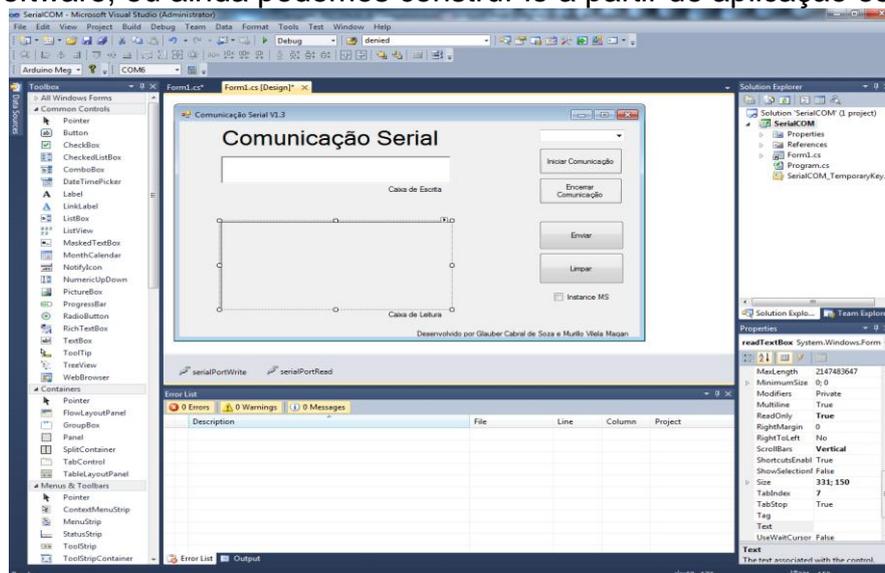


Figura 2: Interface Gráfica em desenvolvimento no Ambiente Visual Studio

Em nosso estudo, já ficou provado que a comunicação entre hardware e software é possível aplicando-se este método, contudo, devemos estabelecer parâmetros de comunicação bem definidos para que os dados não sejam confundidos por quem os recebe. Também devemos estabelecer critérios de prioridade para os dispositivos que possuem maior importância no sistema, determinar o volume de dados enviados por vez, para que não haja sobrecarga da comunicação.

Este método, também abrirá possibilidades para desenvolvimentos futuros com sistemas mais complexos, como por exemplo, a comunicação de um software com vários dispositivos de hardware independentes, comunicações mais robustas, com maiores distâncias, com ou sem fio.

4 CONCLUSÃO

O método de criação do software de supervisão se mostrou eficaz para aplicações em controle de sensores, atuadores, motores e outros, gerenciados a partir de um microcontrolador. Ele irá traduzir o bit enviado de maneira serial, através da porta USB, em uma informação de controle que será transferida ao microcontrolador, ou seja, uma ação que o microcontrolador deverá realizar.

Como previsto, o software de supervisão criado se torna muito mais barato de ser desenvolvido que os softwares que a indústria fornece ao mercado, devido à simplicidade de programação, baixo custo do microcontrolador em relação ao CLP (controlador lógico programável) utilizado na indústria e à gratuidade do software. A finalidade deste método de controle para usuários finais é destinada a aplicações de pequeno porte, simples e que não operem em ambientes de risco. A quantidade de equipamentos a serem controlados dependerá da capacidade do microcontrolador escolhido pelo projetista do sistema de executar comandos lógicos e matemáticos (ciclo de clock e instruções), assim como a eficiência da programação que será inserida no microcontrolador, esta que tem a função de destinar os bits que virão da camada de aplicação (software) através do comando de quem estiver operando o software (usuário final).

O software em questão não foi feito de maneira a controlar uma aplicação específica, e sim de uma maneira que possa expressar o conceito de que com o conhecimento necessário é possível fazer uma aplicação de um processo de pequeno porte, para que não sejam necessários grandes investimentos financeiros.

REFERÊNCIAS

ROSÁRIO, João Maurício. **Princípios de Mecatrônica**. 1.ed. São Paulo: Prentice Hall, 2005. 355p.

STELLMAN, Andrew; GREENE, Jennifer. **Use a Cabeça! C#**. 2.ed. Alta Books, 2011. 618p.

ZANCO, Wagner da Silva. **Microcontroladores PIC16F628A/648A: uma abordagem prática e objetiva**. 2.ed. São Paulo: Érica, 2007. 368 p.