

UM ESTUDO COMPARATIVO ENTRE DOIS SISTEMAS DE AQUISIÇÃO E CONTROLE DE DADOS UTILIZANDO A INTERFACE LABVIEW

Walber M. Lima – m2006157143@alunos.dee.ufcg.edu.br José S. da R. Neto – zesergio@dee.ufcg.edu.br Universidade Federal de Campina Grande - Departamento de Engenharia Elétrica Rua: Aprígio Veloso, 822 - Bodocongó - CEP: 58109-970 - Campina Grande - PB

Resumo: Neste artigo, ressalta-se a importância da realização de um estudo comparativo entre dois sistemas de aquisição e/ou controle de dados, possibilitando aos estudantes da graduação e da pós-graduação do Curso de Engenharia Elétrica da UFCG (Universidade Federal de Campina Grande) uma ferramenta versátil, de fácil implementação e de baixo custo e que tenha um erro tolerável para ser utilizado em seus experimentos. No primeiro sistema, utiliza-se uma placa PCI-MIO-16E-1 desenvolvida pela National Instruments e, no segundo, utiliza-se uma placa de desenvolvimento ADuC (PD-ADuC) desenvolvida no LIEC (Laboratório de Instrumentação Eletrônica e Controle) da UFCG, juntamente com um programa, desenvolvido com o compilador da KEIL, da Analog Devices, capaz de interpretar e responder comandos seriais transmitidos pelo PC (Personal Computer). Para a comparação entre os dois sistemas de aquisição e/ou controle de dados, um sistema de medição de temperatura, utilizando o sensor LM35, foi montado com ambas as placas de aquisição de dados, no qual utilizou-se o programa LabVIEW para criar uma interface gráfica, objetivando visualizar as medições em tempo real.

Palavras-chave: Placa de aquisição, Microcontrolador ADuC, LabVIEW

1 INTRODUÇÃO

A *National Instruments* (NI) vem revolucionando o modo de trabalho dos engenheiros e cientistas. Isto se deve ao lançamento do programa LabVIEW (*Laboratory Virtual Instrument Engineering Workbench*), em 1986, como um ambiente de programação voltado ao desenvolvimento de aplicações, baseado em instrumentação virtual, que permite o rápido avanço tecnológico e comercial (INSTRUMENTS, 2007a).

Neste artigo, realiza-se um estudo comparativo entre dois sistemas de aquisição e/ou controle de dados nos quais os estudantes da graduação e da pós-graduação de engenharia elétrica da UFCG são usuários. No primeiro sistema, utiliza-se de uma placa PCI-MIO-16E-1 desenvolvida pela *National Instruments* e, no segundo, utiliza-se de uma placa de desenvolvimento ADuC (PD-ADuC) desenvolvida no LIEC da UFCG, juntamente com um programa capaz de interpretar e responder comandos seriais transmitidos pelo PC. Para a

comparação entre os dois sistemas de aquisição e/ou controle de dados, um sistema de medição de temperatura foi montado, utilizando o sensor LM35, com ambas as placas de aquisição de dados, na qual utilizou o programa LabVIEW para criar uma interface gráfica, objetivando visualizar as medições em tempo real.

Este trabalho tem uma importância significativa devido ao fato de apresentar um sistema de aquisição e/ou controle de dados versátil, de fácil implementação e de baixo custo e que tenha um erro tolerável. Além de apresentar o desenvolvimento de um programa em LabVIEW, capaz de enviar comandos e receber dados pela porta serial do PC e, em seguida, visualizar os dados em tempo real num gráfico em função do tempo com MATLAB Script, para que os estudantes da graduação e da pós-graduação do Curso de Engenharia Elétrica, nas disciplinas de aquisição de dados e identificação, possam utilizá-lo em seus experimentos melhorando, assim, o desempenho dos alunos nos experimentos que envolvem aquisição e/ou controle de dados.

Na seção 2 deste artigo, apresenta-se o programa LabVIEW e a placa PCI-MIO-16E-1. Já na seção 3, apresenta-se a placa de desenvolvimento ADuC (PD-ADuC), o módulo de aquisição de dados e controle (MADC) e os "Comandos BT". Esse módulo é constituído a partir da união da PD-ADuC com um programa interpretador de comandos seriais ("Comandos BT"), gravado no próprio microcontrolador. Na seção 4, descrevem-se as montagens experimentais para medição de temperatura utilizando a placa PCI-MIO-16E-1 e o MADC. E, na seção 5, apresentam-se os resultados das montagens experimentais, da seção 4, aferidos com a temperatura do termômetro digital da HP 2804.

2 LABVIEW

O LabVIEW permite criar uma interface com o usuário por meio de uma linguagem de programação gráfica chamada G. Essa linguagem possui um conjunto de instruções, mas ao invés de utilizar comandos na forma de texto para gerar as linhas de código, utiliza uma linguagem de programação gráfica, ou seja, o programa é feito na forma de diagrama de blocos. Estes blocos apresentam bibliotecas de funções prontas para utilizar em qualquer tipo de aplicação específica como análise espectral e estatística, filtragem de sinais, aquisição e controle de dados, comunicação com instrumentos I/O, *bluetooth*, TCP, dentre outras (INSTRUMENTS, 2006).

Utilizando uma estrutura de programação hierárquica e orientada pelo fluxo de dados, o LabVIEW torna simples a implementação de sistemas complexos que englobem aquisição de dados, instrumentação e o controle de equipamentos com o PC (INSTRUMENTS, 2006; 2003a).

O LabVIEW reconhece vários barramentos conectados ao PC como GPIB, *Ethernet*, USB, RS-232, RS-485, PCI, CAN, dentre outros. Além disso, o LabVIEW pode trabalhar com programas de outras áreas como o MATLAB e o Excel e acessar e controlar o programa disponível na internet, usando a ferramenta *web publishing* do LabVIEW (INSTRUMENTS, 2003b).

2.1 Instrumentos virtuais

Todo programa feito no LabVIEW é chamado de VI (Instrumento Virtual), pois sua aparência e operação assemelha-se bastante a um instrumento físico, tal como um osciloscópio e um multímetro. Um VI é composto basicamente por um painel frontal; um diagrama de blocos e um ícone/conector (INSTRUMENTS, 2006).

Painel Frontal

O painel frontal é a interface de como o usuário interage com o VI. Este é composto por controles (entradas) e indicadores (saídas) que formam uma tela que simula o painel de um instrumento físico.

Os controles são representados por botões, interruptores e controles numéricos. Já os indicadores são representados por indicadores numéricos, gráficos e *leds*.

No painel frontal, o usuário controla o VI, modificando as variáveis de entrada, nos controles, e vendo os dados de saída atualizados em tempo real, nos indicadores.

Diagrama de blocos

O diagrama de blocos é a estrutura do programa propriamente dita que contém o código fonte na linguagem de programação G.

Todo controle e indicador inserido no painel frontal têm uma representação gráfica no diagrama de blocos permitindo, assim, a sua ligação ao código do programa.

Ícone/Conector

O ícone/conector é a aparência gráfica que se deseja que um VI tenha no diagrama de blocos, quando usado como um subVI (VI encapsulado em um bloco e chamado como uma sub-rotina em um outro VI).

2.2 Paletas

Para editar, operar ou construir um VI, o programa LabVIEW utiliza as seguintes paletas: a paleta de ferramentas, a paleta de controle e a paleta de funções.

O conhecimento, a compreensão e a utilização dessas ferramentas são essenciais para utilizar o programa LabVIEW.

Paleta de ferramentas

A paleta de ferramentas, disponível no painel frontal e no diagrama de blocos, apresenta um modo de seleção automática da ferramenta adequada a ser utilizada pelo usuário, no qual depende da posição do cursor do mouse. Essa paleta permite que o usuário opere os controles, posicione e selecione objetos, edite textos, conecte objetos, dentre outros.

Paleta de controle

A paleta de controle permite que o usuário desenvolva uma interface para uma aplicação qualquer, adicionando controles numéricos, indicadores numéricos, botões, leds, indicadores gráficos, dentre outros, somente, no painel frontal.

Paleta de funções

A paleta de funções permite que o usuário construa o código, em linguagem gráfica, de uma aplicação qualquer, adicionando funções de análise de sinais (de amplitude, de distorção, de ruído, dentre outras), de entrada (simulação de sinais, assistente de aquisição de dados, assistente de instrumentos I/O, dentre outras), de saída (construção de texto, *display* de mensagem, dentre outras), de manipulação de sinais (união, separação e seleção de sinais, dentre outras), aritméticas (soma, subtração, dentre outras), de comparação (maior que, menor que, dentre outras), *boolean (and, or*, dentre outras), de controle de execução (*while loop, case structure*, dentre outras), dentre outras. Esta paleta encontra-se somente no diagrama de blocos.

2.3 Estruturas e gráficos

Assim, como algumas linguagens de programação, o LabVIEW contém laços como o "*while loop*" e o "*for loop*", que são representadas graficamente no diagrama de blocos. Além disso, é possível utilizar estruturas de decisão como o "*case structure*" e estruturas específicas como a "*sequence structure*" e "*formula node*".

O "*waveform chart*" é um especial indicador numérico que pode ser adicionado no painel frontal quando se fizer necessário que os dados gerados por um VI sejam apresentados em um gráfico.

2.4 Interfaceamento dos instrumentos

Os instrumentos reais são reconhecidos pelo LabVIEW por controladores (d*rivers*) compatíveis com a forma de comunicação utilizada (GPIB, RS-232, CAN, dentre outros). Esses controladores são representados por um VI, no diagrama de blocos, contendo rotinas de alto nível capazes de se comunicarem com o *hardware* do instrumento e controlá-lo.

Visa

O VISA (*Virtual Instrument Software Architecture*) é um controlador que permite se comunicar com os instrumentos de maneira bastante uniforme e simplificada. O VISA cobre uma extensa classe de tipos de interfaceamento de instrumentos como, por exemplo: serial RS-232, GPIB, barramentos PCI (PCI extendido e compacto - PXI), PCMCIA, dentre outros.

DAQ

O DAQ (*Data Acquisition*) é um controlador que permite se comunicar com os *hardwares* desenvolvidos pela NI de maneira bastante uniforme e simplificada.

2.5 Placa PCI-MIO-16E-1

A placa PCI-MIO-16E-1 (NI 6070E), *plug and play*, ilustrada na Figura 1, é um *hardware* DAQ da *National Instruments* para PC com barramento PCI. Essa placa contém: 16 entradas analógicas com resolução de 12 *bits*, caso contrário, 8 entradas diferenciais; 2 saídas analógicas com resolução de 12 *bits*; 8 I/O digitais TTL; 2 contadores/temporizador com resolução de 24 *bits*; condicionamento dos sinais de entrada com o PGIA (Amplificador de Ganho Programável) e 1 conector I/O de 68 pinos (INSTRUMENTS, 2007b).



Figura 1 – Foto da placa PCI-MIO-16E-1 e o conector I/O de 68 pinos.

3 PLACA DE DESENVOLVIMENTO ADUC

Uma placa de desenvolvimento ADuC (PD-ADuC), ilustrada na Figura 2, foi desenvolvida no LIEC, por um aluno da pós-graduação na disciplina de aquisição de dados, com o objetivo de permitir o desenvolvimento de sistemas de aquisição e/ou controle de dados com o PC. Essa placa possibilita ao usuário acessar todos os recursos do microcontrolador ADuC colocado nesta, com a porta de comunicação serial (RS-232) do PC, de forma que a aplicação em estudo possa ser desenvolvida sem a necessidade de montagens adicionais (ARAÚJO et al., 2005).

Os recursos da PD-ADuC, utilizando o microcontrolador ADuC832, são os seguintes: Acesso a memória FLASH de programa (62 *Kbytes*), FLASH de dados (4 *Kbytes*) e SRAM (2 *Kbytes*); 8 canais do conversor A/D de 12 *bits*, cada um com *buffers* de proteção; 2 conversores D/A de 12 *bits*; Velocidade de aquisição 247 KSPS; 2 geradores PWM de 16 *bits*; 24 pinos de E/S disponibilizados nas portas (P0, P2 e P3); Comunicação serial RS-232; Comunicação serial RS-485; *Leds* indicadores de recepção RX e transmissão TX; 1 sensor de temperatura LM35, 1 sensor de luminosidade LDR (*Light Dependent Resistor*); 8 *leds* de sinalização para a porta P0 e Comunicação síncrona SPI e I2C (ARAÚJO et al., 2005; DEVICE, 2002).



Figura 2 – Diagrama da placa de desenvolvimento ADuC (PD-ADuC).

Cada um dos oito canais do conversor A/D possui um pré-condicionamento de sinal, sendo este composto por um *buffer* externo que minimiza o erro de carregamento no ponto de medição. Além disso, possui um filtro passa-baixa destinado a eliminar ruídos de alta freqüência. Os sinais de entrada devem compreender uma faixa de valores entre 0 e 2,5V. Um sensor de temperatura (LM35) e um sensor de luminosidade LDR estão conectados aos canais 0 e 1, respectivamente do conversor A/D do microcontrolador. Estes sensores permitem um contato inicial do usuário com a leitura do conversor A/D sem a necessidade de montagens externas, por exemplo, a leitura da temperatura ambiente do local onde a placa PD-ADuC estiver localizada. Caso a leitura do LM35 e do LDR da PD-ADuC não seja necessária, os canais 0 e 1 podem ser liberados para a leitura de tensões externas, mudando *straps* de seleção (ARAÚJO et al., 2005).

3.1 Módulo de aquisição de dados e controle (MADC)

Com os recursos de instrumentação facilmente acessíveis, um sistema de aquisição e controle de dados pode ser implementado facilmente, utilizando o microcontrolador ADuC. Para tal implementação, é necessário um programa, desenvolvido com o compilador da KEIL e gravado no microcontrolador, capaz de interpretar e responder comandos seriais transmitidos pelo PC (ARAÚJO et al., 2005; DEVICE, 2001).

Portanto, isto permitiu, no LIEC, a criação do MADC, no qual é constituído a partir da união da PD-AduC, juntamente com o programa interpretador de comandos seriais ("Comandos BT"). Com o MADC conectado ao PC e com uma porta serial (RS-232), é possível realizar operações de aquisição e/ou controle sobre o processo em estudo sem a necessidade de desenvolver uma programação de baixo nível (ARAÚJO et al., 2005).

3.2 Comandos BT

Os "Comandos BT", que vem do inglês "*Basic Transmission Command*", são uma referência para comandos de transmissão básica. As letras T e V demarcam o início e o fim de um quadro de comando, respectivamente (ARAÚJO et al., 2005).

O quadro de um "Comando BT" é formado por uma seqüência de até 7 caracteres ASCII, que são interpretados pelo MADC. Como cada caractere possui 8 *bits*, o quadro de um "Comando BT" pode ter até 56 *bits*. O quadro de um "Comando BT" é ilustrado na Figura 3.



Figura 3 – Quadro de um "Comando BT".

Os "Comandos BT" utilizam caracteres ASCII para facilitar a criação de aplicações com o programa LabVIEW.

O conjunto de "Comandos BT", listados na Tabela 1, permite acesso aos dispositivos de *hardware* existentes no MADC, nos quais os valores numéricos também são transmitidos como caracteres ASCII. Os comandos que solicitam a leitura de determinados recursos, como o conversor A/D ou porta de E/S, recebem como resposta uma seqüência de caracteres numéricos ASCII com o valor correspondente a leitura solicitada. Estas respostas são seguidas de um caractere de controle (**n**) sinalizando para o PC o final da transmissão da resposta.

"Comandos BT"	Id	Função	Faixa de Valores
TAxV	Α	Leitura do canal x do Conversor A/D	0 a 9
TBxxxxV	В	Escrita do valor xxxx em mV	0000 a 2499
		no Conversor D/A 0	
TCxxxxV	С	Escrita do valor xxxx em mV	0000 a 2499
		no Conversor D/A 1	

Tabela 1 - Lista dos "Comandos BT".

"Comandos BT"	Id	Função	Faixa de Valores
TDyxxxV	D	Escreve o valor xxx na porta	xxx – 000 a 255
		de E/S y	y – 0 a 3
TEyV	Е	Leitura do valor da porta de E/S y	y – 0 a 3
		Escrita do valor binário x	x – 0 ou 1
TFzyxV	F	no bit y da porta z	y – 0 a 7
			z – 0 a 3
TGzyV	G	Leitura do valor binário	y – 0 a 7
		do bit y da porta z	z – 0 a 3
THhhmmV	Н	Acerta o RTC do MADC	hh – 00 a 23
			mm – 00 a 59
THORAV	Ι	Leitura do RTC do MADC	_
TPxxxxV	Р	Escrita do valor xxxx no gerador PWM	xxxx - 0000 a 2047

4 MONTAGENS EXPERIMENTAIS

Um sistema de medição de temperatura, utilizando o sensor LM35, foi montado com uma placa PCI-MIO-16E-1 e com o MADC, no qual utilizou-se o LabVIEW para criar uma interface gráfica, objetivando visualizar as medições em tempo real.

4.1 Montagem experimental usando a placa PCI-MIO-16E-1

Inicialmente, conecta-se a placa PCI-MIO-16E-1 ao barramento PCI do PC que possua o LabVIEW instalado e, em seguida, conecta-se uma das extremidades do cabo I/O *ribbon* (R6868), de 68 vias, na placa PCI-MIO-16E-1 e a outra no conector I/O de 68 vias como ilustrado na Figura 4.





Um programa (VI) foi desenvolvido no LabVIEW para indicação da temperatura. Este programa adquire o valor da tensão do sinal de saída do sensor de temperatura (LM35), com o canal 4 do conversor A/D e, em seguida, este valor é convertido para um valor em temperatura e armazenado em uma variável no MATLAB Script, para traçar o gráfico da temperatura em função do tempo.

Na janela do painel frontal do VI, ilustrada na Figura 5, apresenta-se: um botão "Aquisição", que tem a função de fazer a aquisição da tensão do sinal de saída do sensor quando pressionado e traçar o gráfico da temperatura em função do tempo sem armazenar dos dados adquiridos; um botão de "STOP" que tem a função de parar a execução do VI e um botão "Gráfico MATLAB" que tem a função de armazenar os dados adquiridos e traçar o gráfico da temperatura em função do tempo.



Figura 5 – Janela do painel frontal para o experimento utilizando a placa PCI.

Na janela do diagrama de blocos do VI, ilustrada na Figura 6, apresenta-se o código do programa para a indicação da temperatura.



Figura 6 – Janela do diagrama de blocos para o experimento utilizando a placa PCI.

O código do programa encontra-se dentro do laço (*while loop*). Neste laço, o código não executa nada até que o usuário pressione o botão "Aquisição", na janela do painel frontal, que habilita a estrutura de caso para aquisição de dados. Essa estrutura, quando habilitada, aciona o assistente de aquisição de dados (DAQ), que se comunica com a placa PCI-MIO-16E-1 de maneira bastante simplificada para fazer aquisição da tensão do sinal de saída do sensor em um dos canais do conversor A/D selecionado. Depois, o valor da tensão adquirida é multiplicado pelo fator de conversão do sensor de temperatura convertendo, assim, em um valor de temperatura e, em seguida, é enviado para o gráfico da Temperatura X Tempo, na janela do painel frontal, e para o MATLAB Script.

Para armazenar e visualizar os dados enviados pelo assistente DAQ em um gráfico em função do tempo, o usuário deve pressionar o botão "Gráfico MATLAB", na janela do painel frontal, que habilita a estrutura de caso para armazenamento de dados. Essa estrutura, quando habilitada, recebe o dado enviado pelo assistente DAQ para ser armazenado no MATLAB Script e, em seguida, visualizá-lo em um gráfico em função do tempo. Para finalizar o programa, o usuário deve pressionar o botão "STOP", na janela do painel frontal, para sair do laço.

4.2 Montagem experimental usando o MADC

Inicialmente, conecta-se a porta de comunicação serial RS-232 do MADC em alguma porta serial do PC que possua o LabVIEW instalado como ilustrado na Figura 7.



Figura 7 – Diagrama da montagem experimental com o MADC.

Um programa (VI) foi desenvolvido no LabVIEW para utilizar todos os recursos do MADC, utilizando os "Comandos BT", em especial, para a indicação da temperatura. Esse programa envia o "Comando BT" TA0V, por meio da porta serial do PC para que o MADC possa interpretá-lo, no qual adquire o valor da tensão do sinal de saída do sensor de temperatura pelo canal 0 do conversor A/D e, em seguida, esse valor é convertido para um valor em temperatura e armazenado em uma variável no MATLAB Script para traçar um gráfico da temperatura em função do tempo.

Na janela do painel frontal do VI, ilustrada na Figura 8, apresenta-se: um quadro com todos os "Comandos BT"; um botão de "STOP" que tem a função de parar a execução do VI; um botão de "Gráfico MATLAB" que tem a função de armazenar os dados adquiridos e traçar um gráfico da temperatura em função do tempo; um botão "Escrita" que tem a função de habilitar a escrita; um botão "Leitura" que tem a função de habilitar a leitura, um editor de *string* para escrita, um indicador de *string* para a leitura e sete controles de configuração da porta serial como, por exemplo: *Visa Resource Name, Baud Rate, Data Bits, Parity, Stop Bits* e *Flow Control.*

COMA				
CONVERSOR A/D	PORTA PARALELA			
TAOV - LEITURA NO CANAL O DO A/D	TEOV - LEITURA NA PORTA 0	-		
TAIV - LEITURA NO CANAL 1 DO A/D	TEIV + LEITURA NA PORTA 1	VISA RESOURCE	IAME	
TA2V - LEITURA NO CANAL 2 DO A/D	TE2V - LEITURA NA PORTA 2	1%COM1		
TA3V - LEITURA NO CANALI 3 DO 'A/D	TE3V - LEITURA NA PORTA 3	BAUD RATE	CDÁE	
TA4V - LEITURA NO CANAL 4 DO A D	PINO DA PORTA PARALELA	5/119200	GRAU	
TASV - LEITURA NO CANAL 5 DO A/D	TFOXYV - ESCRITA DE Y MO PINO X DA PORTA Ø	18		ESCRITA DE STRING
TA6V - LEITURA NO CANAL 6 DO A/D	TFIXYV - ESCRITA DE Y MO PINO X DA PORTA 1	PARITY	ESCRITA	TAOV
TA7V - LEITURA NO CANAL 7 DO A/D	TF2XYV - ESCRITA DE Y MO PENO X DA PORTA 2	None	9	
CONTRACT D/A	TF3XYV - ESCRITA DE Y MO PINO X DA PORTA 3	STOP BITS		
CONVERSOR D/A	TGOXV - LEITURA NO PINO X DA PORTA 0	E OW CONTROL		
TEXXXXV - ESCRITA XXXX VOLTS NO D/A 0	TGIXV - LEITURA NO PINO X DA PORTA 1	None	LEITURA	
TCXXXXV - ESCRITA XXXX VOLTS NO D/A 1	TG2XV - LEITURA NO PINO X DA PORTA 2		9	02728
PORTA PARALELA	TG3XV - LEITURA NO PINO X DA PORTA 3			
TDOXXXV - ESCRITA NA PORTA 0	HORA	STOP		
TDIXXXV - ESCRITA NA PORTA 1	THORAV - LEITURA DA HORA			
TD2XXXV - ESCRITA NA PORTA 2	THXXXXV - AJUSTE DA HORA			
TD3XXXV - ESCRITA NA PORTA 3				

Figura 8 – Janela do painel frontal para o experimento utilizando o MADC.

Na janela do diagrama de blocos do VI, ilustrada na Figura 9, apresenta-se o código do programa para utilização de qualquer um dos recursos do MADC com os "Comandos BT".



Figura 9 – Janela do diagrama de blocos para o experimento utilizando o MADC.

O código do programa utiliza o controlador VISA para comunicar-se com a serial do PC de modo simplificado. Portanto, o usuário deve informar a porta de comunicação serial, a taxa transferência de dados, a quantidade de *bits* de dados, o tipo de paridade, a quantidade de *bits* de parada e o tipo de controle de fluxo nas entradas VISA RESOURCE NAME, BAUD RATE, DATA BITS, PARITY, STOP BITS e FLOW CONTROL respectivamente, na janela do painel frontal, para que o VISA Serial inicie a comunicação com a porta serial. Em seguida, o restante do código encontra-se dentro do laço (*while loop*), com interações a cada 30 ms até que o usuário digite algum "Comando BT" no editor de ESCRITA DE STRING, na janela do painel frontal, e, em seguida, pressione o botão ESCRITA, na janela do painel frontal, que habilita a estrutura de caso para escrita de *string*. Essa estrutura, quando habilitada, envia, com o VISA Write, o comando escrito no editor de ESCRITA DE STRING, não envia nada.

Para visualizar os dados enviados pelo MADC no indicador de LEITURA DE STRING, o usuário deve pressionar o botão LEITURA, na janela do painel frontal, que habilita a estrutura de caso para leitura de *string*. Essa estrutura, quando habilitada, recebe, com o VISA Read, o dado enviado pelo MADC, por meio da porta serial do PC, na forma de decimal *string*, e envia-o para o indicador de LEITURA DE STRING, caso contrário, não recebe nada.

Para armazenar e visualizar o dado enviado pelo MADC em um gráfico em função do tempo, o usuário deve pressionar o botão GRÁFICO MATLAB, na janela do painel frontal, que habilita a estrutura de caso para armazenamento de dados. Essa estrutura, quando habilitada, recebe o dado, na forma de decimal *string*, do VISA Read e o converte em decimal inteiro para ser armazenado no MATLAB Script e, em seguida, o dado é multiplicado pelo fator de conversão do sensor de temperatura para poder visualizá-lo em um gráfico da temperatura em função do tempo. Para finalizar a comunicação, o usuário deve pressionar o botão STOP, na janela do painel frontal, para sair do laço e finalizar a comunicação serial com o VISA Close.

5 RESULTADOS EXPERIMENTAIS

O gráfico da temperatura em função do tempo, ilustrado na Figura 10, é o resultado do experimento realizado com a placa PCI-MIO-16E-1, no qual observa-se uma temperatura média de 26,05°C.



Figura 10 – Gráfico da temperatura versus tempo utilizando a placa PCI-MIO-16E-1.

O gráfico da temperatura em função tempo, ilustrado na Figura 11, é o resultado do experimento realizado com o MADC, no qual observa-se uma temperatura média de 26,18°C.



Figura 11 – Gráfico da temperatura versus tempo utilizando o MADC.

Com o auxílio de um termômetro digital da HP 2804, em ambos os experimentos realizados, foram observados uma temperatura de aproximadamente 26,50°C. Assim, acarretando um erro de 1,7% e 1,2% para a placa PCI-MIO-16E-1 e o MADC, respectivamente.

6 CONSIDERAÇÕES FINAIS

Neste artigo, realizou-se um estudo comparativo entre dois sistemas de aquisição e/ou controle de dados, nos quais os estudantes da graduação e da pós-graduação são usuários. No primeiro sistema, utiliza-se uma placa PCI-MIO-16E-1 desenvolvida pela NI e, no segundo, utiliza-se uma placa de desenvolvimento ADuC (PD-ADuC) desenvolvida no LIEC, juntamente com um programa, desenvolvido com o compilador da KEIL, da *Analog Devices*, capaz de interpretar e responder comandos seriais transmitidos pelo PC.

Para a comparação entre os dois sistemas de aquisição e/ou controle de dados, um sistema de medição de temperatura, usando o sensor de temperatura (LM35), foi montado com ambas as placas de aquisição de dados, no qual utilizou-se o programa LabVIEW para criar uma interface gráfica para visualizar as medições de temperatura em tempo real.

Nas medições realizadas, observou-se um erro de 0,5% entre os dois sistemas de aquisição de dados, portanto, os estudantes da graduação e da pós-graduação, nas disciplinas de aquisição de dados e identificação, podem se utilizar de um sistema de aquisição de dados versátil, de baixo custo e de erro tolerável em relação à placa PCI-MIO-16E-1, juntamente com o programa desenvolvido no LabVIEW, capaz de enviar comandos e receber dados pela porta serial do PC e, em seguida, visualizar os dados em tempo real num gráfico em função do tempo com MATLAB Script melhorando, assim, o desempenho dos alunos nos experimentos que envolvem aquisição e controle de dados.

Agradecimentos

Os autores agradecem ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) pelo financiamento das pesquisas e em especial ao técnico Simões Toledo, que confeccionou e montou as placas desenvolvidas durante este trabalho.

7. REFERÊNCIAS BIBLIOGRÁFICAS

ARAUJO, T. V. G. P.; FILHO, C. A. S.; LIMA, A. M. N.; BARROS, P. R. *Kit de Desenvolvimento para Microcontroladores ADUC Aplicado ao Ensino de Controle e Automação*. In: Congresso Brasileiro de Ensino de Engenharia, 33. Campina Grande. Anais do Congresso Brasileiro de Ensino de Engenharia, 2005, v. 1. p. 1-6.

DEVICE, A., 2001. **MicroConverter**[®], **12-Bit ADCs and DACs with Embedded 62 kBytes Flash**. Disponível em: < http://www.analog.com/UploadedFiles/Application_Notes/41355335 13501551368uC002___Developing_in_C_with_the_Keil_uVision2_IDE.pdf>. Acesso em: 26 abr. 2007.

DEVICE, A., 2002. MicroConverter[®], 12-Bit ADCs and DACs with Embedded 62 kBytes Flash. Disponível em: http://www.analog.com/UploadedFiles/Data_Sheets/ADUC832.pdf Acesso em: 26 abr. 2007.

INSTRUMENTS, N., 2003a. LabVIEW: User Manual. Disponível em: http://www.ni.com/pdf/manuals/320999e.pdf>. Acesso em: 26 abr. 2007.

INSTRUMENTS, N., 2003b. LabVIEW: Getting Started with LabVIEW. Disponível em: http://www.sfu.ca/phys/231/063/data_sheets/LabView%20tutorial.pdf>. Acesso em: 26 abr. 2007.

INSTRUMENTS, N., 2006. LabVIEW Tutorial Manual. Disponível em: http://www.physics.utoledo.edu/~alukasz/labview_tutorial.PDF>. Acesso em: 26 abr. 2007.

INSTRUMENTS, N., 2007a. National Instruments Brasil: Sobre a National Instruments. Disponível em: http://www.ni.com/brazil>. Acesso em: 26 abr. 2007.

INSTRUMENTS, N., 2007b. **DAQ E Series: E Series User Manual**. Disponível em: <<u>http://digital.ni.com/manuals.nsf/websearch/06115536722DAA558625728E007E6BA4></u>. Acesso em: 26 abr. 2007.

A COMPARATIVE STUDY AMONG TWO SYSTEMS OF ACQUISITION AND CONTROL OF DATA USING THE INTERFACE LABVIEW

Abstract: In this article emphasized itself the importance of the accomplishment of a comparative study among two acquisition systems and/or data control, making possible to the students of the graduation and of the masters degree of the Course of Electric Engineering of UFCG (Universidade Federal de Campina Grande) a tool versatile, of easy implementation and low cost and that has a tolerable error to be used in their experiments. In the first system used itself one PCI-MEOW-16E-1 card developed by National Instruments and in the second used itself one ADuC development card (PD-ADuC) developed in LIEC (Laboratório de

Instrumentação Eletrônica e Controle) of the UFCG, together with a program, developed with the compiler of the KEIL, of the Analog Devices, capable to interpret and to answer serial commands transmitted by PC (Personal Computer). For the comparison among the two acquisition systems and/or data control, a system of temperature measurement, using the sensor LM35, it was mounted with both data acquisition card, in which it used itself the LabVIEW software to create a graphic interface, objectifying to visualize the measurements in real time.

Key-words: Acquisition card, ADuC microcontroller, LabVIEW