

Integração de Sensores em Sistemas com Barramento de Dados Digital com o Propósito de Ensino Utilizando o Ambiente de Desenvolvimento LabVIEW

Euphrásio, P.C. S..

Instituto Tecnológico de Aeronáutica, ITA, Pça. Mal. Eduardo Gomes, 50 - Vila das Acácias - CEP 12228-900 – São José dos Campos – SP – Brasil

Resumo. O trabalho apresenta o desenvolvimento de dois aplicativos desenvolvidos no ambiente de programação gráfica LabVIEW, os quais realizam a aquisição e o processamento de dados provenientes de três sensores. Ele tem como objetivo principal ser utilizado como ferramenta de apoio para o ensino em disciplinas sobre integração de sensores em sistemas que utilizam barramentos de dados digitais. No sistema proposto podem ser observadas a transformação do fenômeno físico em fenômeno elétrico, a sua digitalização e a transmissão pelo barramento digital de dados CANBus.

Palavras chaves: CANBus, LabVIEW e integração de sistemas.

I. INTRODUÇÃO

O crescente aumento do número de sensores aos sistemas embarcados em veículos terrestres, navais e aeroespaciais, tanto de uso civil quanto militar, aponta para a necessidade cada vez maior do domínio do conhecimento na área de integração de sistemas embarcados. A utilização de ferramentas adequadas, de fácil manuseio e integração de *hardware* para projeto e implementação rápidos é algo desejável para os profissionais da área de pesquisa & desenvolvimento e da área de ensino [1].

O sistema proposto visa a capacitação de profissionais da área de integração de sistemas, através do qual o capacitando poderá observar e compreender:

- A transformação do fenômeno físico (pressão e temperatura) em sinais elétricos;
- A aquisição dos sinais elétricos e a apresentação do seu processamento;
- A apresentação dos dados em unidade de engenharia, e
- A transmissão dos dados de um computador para outro utilizando um barramento de dados padronizado.

São utilizados dois computadores para simular duas ECUs (*Electronic Control Unit*) de um sistema embarcado e um barramento de dados CAN (*Controller Area Network*) [2] que serve como o meio de comunicação de dados entre eles.

Foram desenvolvidos dois programas utilizando o ambiente de desenvolvimento LabVIEW. O primeiro programa tem a função de realizar a aquisição dos sinais provenientes dos sensores, o controle e a transmissão dos dados e o segundo programa tem a função de fazer a recepção e apresentação dos dados provenientes do CAN.

Utiliza-se um osciloscópio com decodificador CAN para se observar o sinal elétrico e os valores em hexadecimal dos dados que estão sendo transmitidos no barramento.

Os sensores usados são: um transdutor de pressão diferencial que fornece os sinais para conversão da pressão diferencial em velocidade em nós e um sensor de temperatura para apresentação dos valores em graus Celsius [3].

II. DESCRIÇÃO DO SISTEMA

A Fig. 1 apresenta o sistema proposto que é composto pelos seguintes componentes:

- Sensores: transdutor de pressão diferencial modelo 133201 (0 a 5 psid): responsável por realizar a leitura de pressão proveniente do banco portátil e por realizar a leitura da pressão estática proveniente do ambiente. Sensor de temperatura LM35 controlado por uma placa Arduino Uno.
- Banco portátil (type V.61): responsável por inserir pressão no transdutor e apresentar o valor da velocidade no instrumento incorporado a ele.
- Placa de aquisição NI USB-6251 e plataforma Arduino: responsáveis por realizar a leitura do sinal elétrico dos sensores e enviá-los para o computador por portas de comunicação USB.
- Fonte de alimentação de DC: responsável por alimentar o transdutor com uma tensão média de 28V.
- Computadores com *software* LabVIEW: responsáveis por operar os *softwares* de controle do sistema de aquisição e enviar os dados por uma rede CAN.
- Osciloscópio: responsável por decodificar e apresentar os valores em hexadecimal dos sinais provenientes do barramento CAN.

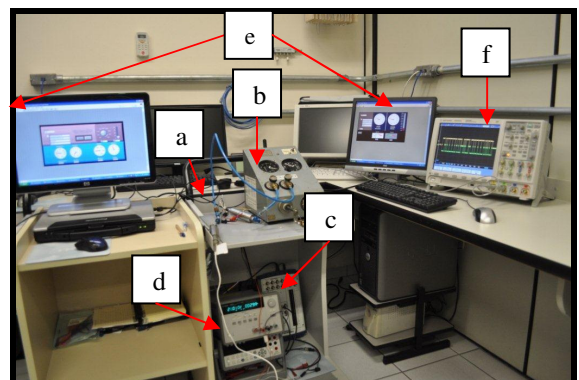


Fig. 1: Componentes do sistema de proposto.

Além dos componentes apresentados, o sistema possui dois cartões de aquisição CANBus responsáveis por transmitir e receber os dados de velocidade e temperatura de um *notebook* para um *desktop* [4].

III. EXPERIMENTOS PROPOSTOS.

Para a integração de sensores ao barramento CAN são propostos dois experimentos [3]:

- Leitura de velocidade utilizando um transdutor de pressão diferencial.
- Leitura de temperatura utilizando um sensor LM35.

III.1. Experimento com transdutor de pressão diferencial.

Para a realização do primeiro experimento foi utilizado um banco de injeção de pressão (Fig. 2) que é composto por um velocímetro (a), uma manivela (b), e uma saída de pressão de ar (c). A manivela permite aumentar ou diminuir a pressão que é enviada simultaneamente para o velocímetro e para o transdutor. O transdutor mede a diferença entre a pressão do banco portátil e a pressão atmosférica.

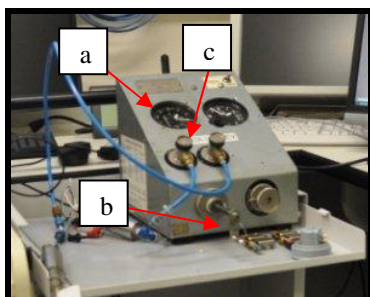


Fig. 2: Banco portátil de injeção de pressão.

Também foram utilizados (Fig. 3) uma placa de aquisição NI USB-6251, uma fonte de alimentação de 28V e o *software* AqGeraDado desenvolvido no LabVIEW. A placa NI USB-6251 é a interface responsável por receber os sinais elétricos do transdutor no canal AI0 (Analogic Input Zero) e enviar para o computador com o *software* AqGeraDado utilizando uma porta de comunicação USB.

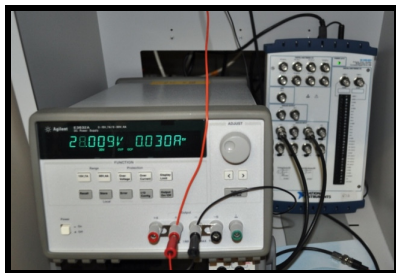


Fig. 3: Fonte DC e placa de aquisição NI USB-6251.

A tabela 1 apresenta os valores da pressão injetada no transdutor e seu correspondente valor de tensão que é adquirida pela placa NI USB-6251.

III.1.1. FUNCIONAMENTO DO SISTEMA TPD.

O banco portátil é utilizado para aumentar a pressão do sistema girando a manivela no sentido horário. Ao receber a variação de pressão, o transdutor realiza o cálculo diferencial entre esta pressão e a pressão ambiente, apresentando na sua saída elétrica os valores de tensão calculados. Essas variações de tensão são recebidas pelo canal AI0 da placa de aquisição NI USB-6251 e são transferidas, através de um canal de comunicação USB, para um notebook. O *software* AqGeraDado decodifica o

signal e apresenta os valores na tela em duas formas: níveis de tensão (volt) e velocidade (nós).

TABELA 1: PRESSÃO DIFERENCIAL DE ENTRADA VERSUS TENSÕES DE SAÍDA DO TRANSDUTOR.

Pressão (PSI)	Dados qb (Volts)
0,5	0,4550
1	0,9600
1,5	1,4650
2	1,9690
2,5	2,4730
3	2,9770
3,5	3,4800
4	3,9830
4,5	4,4840
5	4,9870
4,5	4,4860
4	3,9840
3,5	3,4820
3	2,9790
2,5	2,4750
2	1,9710
1,5	1,4670
1	0,9620
0,5	0,4560
0,1	0,0520

Foi realizado um ajuste linear para se obter a equação de conversão de PSI para Volt (tabela 1) necessário para a apresentação dos valores de velocidade no VI (*Virtual Instrument*) do AqGeraDado. As equações utilizadas para o VI de velocidade são:

1- Equação de ajuste linear para a conversão Pontos (psi) X Dados Volt:

$$y_{qb} = 0,9927 \times qb - 0,0457, \quad (1)$$

onde qb é a tensão, em volts, proveniente do transdutor.

Após passar pelo ajuste linear, a velocidade em nós é obtida através da equação,

$$vb = 661,5 \times \sqrt{5 \times \left(\left(\frac{68,94757 \times y_{qb}}{1013,25} + 1 \right)^{0,28571} - 1 \right)}, \quad (2)$$

onde y_{qb} é o ajuste linear de qb (tensão de entrada proveniente do transdutor). A constante 68.94757 é utilizada para converter os valores de pressão de milibar para psi.

A Fig. 4 apresenta o instrumento velocímetro real com a velocidade de 300 nós, o velocímetro virtual com 297,278 nós e o valor de tensão de entrada igual a 2,15469 Volts.



Fig. 4: velocímetro real e instrumentos virtuais do AqGeraDado.

III.2. EXPERIMENTO COM SENSOR DE TEMPERATURA:

O Arduino é uma plataforma de *hardware* e *software* livres que simplifica a criação e prototipagem de projetos de eletrônica projetada com um micro controlador Atmel [5]. Ele possui uma linguagem de programação padrão, de código aberto, que tem origem em *Wiring* e é essencialmente uma linguagem C/C++ [5].

O sensor de temperatura LM35 (detalhe “a” - Fig. 5) é alimentado eletricamente pelo Arduino

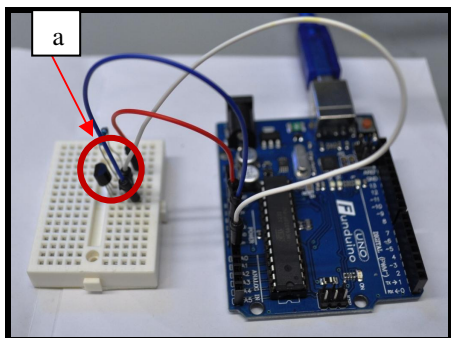


Fig. 5: sensor de temperatura LM35 e placa Arduino

III.2.1. FUNCIONAMENTO DO SISTEMA DE AQUISIÇÃO DE TEMPERATURA.

O sensor LM35 é alimentado com uma tensão média de 5V proveniente da placa Arduino. É utilizada a porta AIO do Arduino para receber os valores de temperatura que são transferidos para o AqGeraDado pela porta USB.

É necessário carregar o arquivo LIFA_Base.ino no Arduino para que seja possível utilizar as bibliotecas do LabVIEW para a programação da temperatura.

III.3. SOFTWARE AQGERADADO PARA A AQUISIÇÃO, PROCESSAMENTO E TRANSMISSÃO DOS DADOS.

Para a realização de aquisição, processamento e transferência dos dados dos sensores, foi desenvolvido o *software* AqGeraDado (Fig. 6), instalado em um *notebook*. Ele possui os componentes necessários para receber os sinais, realizar o cálculo de velocidade e temperatura. O AqGeraDado faz também a digitalização dos sinais e transfere os dados para outro computador utilizando o barramento CAN. Na sua tela são apresentadas as seguintes funções:

- Interface*: Canal do barramento CAN a ser utilizado para a transmissão dos dados.
- Channel List*: lista de parâmetros a serem transmitidos.
- Start / Stop*: botões de início e parada da transmissão de dados pelo barramento CAN.
- Temperatura: instrumento e valor numérico que apresentam o valor da temperatura
- Velocidade: instrumento e valor numérico que apresentam o valor da velocidade.
- Tensão: instrumento e tela que apresenta o valor de tensão do velocímetro capturado pela placa NI USB-6251.

O AqGeraDado, após o processamento e a apresentação dos dados coletados dos sensores, permite a transferência desses para outro computador (*desktop*) utilizando o CAN.

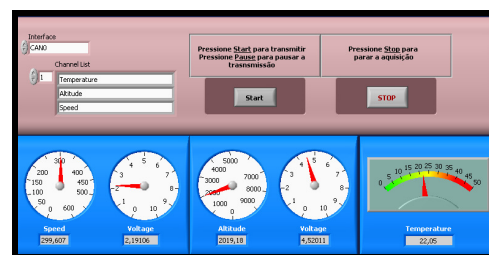


Fig. 6: tela principal do *software* AqGeraDado.

A Fig. 7 apresenta o bloco do diagrama do AqGeraDado, que é responsável por receber e apresentar o valor de tensão do transdutor (qb). Após passar pelas equações de conversão, o valor da velocidade atual é mostrada no instrumento virtual.

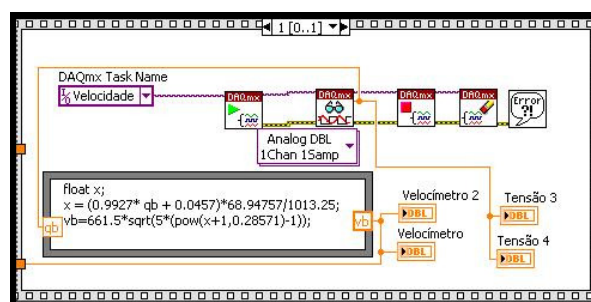


Fig. 7: Diagrama de VIs dos sensores de velocidade.

A Fig. 8 apresenta o bloco do diagrama do AqGeraDado, responsável pelo controle da temperatura. Esse bloco recebe o valor de tensão do sensor LM 35 e, após passar pela multiplicação por 100, apresenta o seu valor no instrumento virtual. O LM 35 fornece valores de tensão em miliVolts.

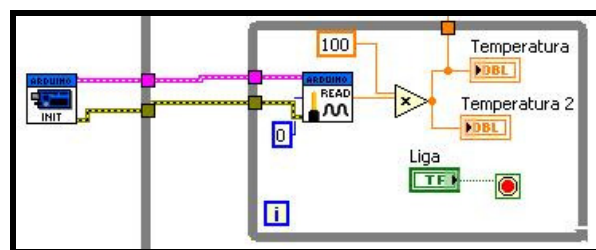


Fig. 8: Diagrama de VIs do sensor de temperatura.

III.4. SOFTWARE REC DADO PARA A RECEPÇÃO E APRESENTAÇÃO DOS DADOS DOS SENSOES.

No *desktop* foi desenvolvido um *software* denominado RecDado para receber os dados provenientes do AqGeraDado utilizando como meio de comunicação o barramento CAN. Após receber os dados, ele os apresenta em seus respectivos instrumentos virtuais. A Fig. 9 apresenta a sua tela com as seguintes funções:

- Interface*: Canal do barramento CAN que recebe os dados.
- Channel List*: lista de dados a serem recebidos.
- Stop*: botão de parada da recepção de dados.
- Speed*: instrumento e valor numérico que apresentam o valor da velocidade.
- Temperatura: instrumento e valor numérico que apresentam o valor da temperatura.

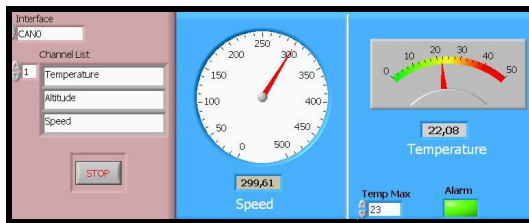


Fig. 9: tela principal do software RecDado.

Nesse aplicativo foi colocado um controle de temperatura, no qual o usuário estabelece um valor (23) e, quando a temperatura for maior que esse valor, o “led” troca sua cor de verde para vermelho [6].

III.4.1. Transmissão dos dados pelo barramento CAN.

Em uma rede CAN, as mensagens transferidas pela rede são chamadas quadros (*frames*) e os tipos de quadros podem ser divididos em Quadro de Dados, Quadro Remoto, Quadro de Erros e Quadro de Sobrecarga [7].

A Fig. 10 apresenta uma mensagem CAN com quadro de dados com seis *bytes* capturados por um osciloscópio. Nela podem ser observados os bits da mensagem, sendo identificados os principais campos:

- SOF + Identificador de arbitragem: 052₁₆;
- DLC (código de comprimento dos dados): 8₁₆;
- Dados: (14 01 09 75 00 8E 14 03)₁₆
- CRC + ACK: 5C69₁₆

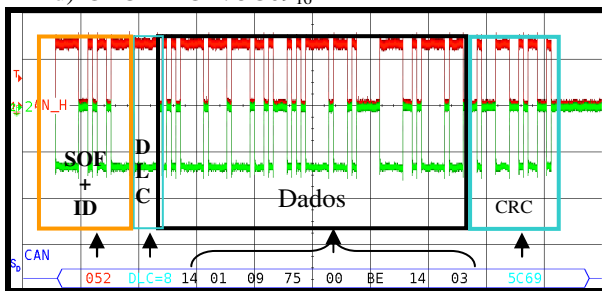


Fig. 10: mensagem CAN capturada do osciloscópio com: temperatura (1401)₁₆ e velocidade(007509)₁₆.

Nesse trabalho é considerado apenas o Quadro de Dados, pois ele é o responsável pelo envio dos dados de um computador (nó) para outro. Serão referenciados apenas os cinco primeiros bytes de dados, onde os dois primeiros são utilizados para a temperatura e os três seguintes para a velocidade. Os três bytes finais não serão referenciados.

A Fig. 6 apresenta os dados coletados dos sensores que foram transmitidos no barramento CAN com os seguintes parâmetros: velocidade de 299,607 nós e temperatura de 22,05 graus centígrados. Após selecionar o botão Iniciar do AqGeraDado, a transmissão dos dados dos sensores pelo CAN é iniciada e o software RecDado (Fig. 9), recebe-os e os apresenta nos seus respectivos instrumentos.

A Fig. 10 apresenta os sinais elétricos da mensagem CAN, capturada por um osciloscópio, com os parâmetros de velocidade e temperatura em hexadecimal. Os parâmetros (*bytes*) são lidos da direita para a esquerda, seguindo as seguintes convenções:

- A velocidade de 299.61 nós é representada por 097500₁₆ (lê-se primeiro o número 00₁₆, após o 75₁₆ seguido do 09₁₆), a qual a equação para transformar o valor da velocidade é dada por:

$$V = \frac{X_{10}}{100} \quad (3)$$

como X_{10} é o valor da velocidade na base 10 (007509₁₆ = 29961₁₀)

$$V = \frac{29961}{100} \rightarrow V = 299,61 \text{ nós}$$

- A temperatura de 22,08 °C é representada por 1401₁₆, (lê-se 01₁₆ seguido de 14₁₆). O cálculo para transformar o valor hexadecimal para temperatura é obtido por:

$$T_c = 0,08 \times X_{10} \quad (4)$$

, X_{10} é o valor do hexadecimal na base 10:

$$(0114_{16} = 276_{10}) \rightarrow T_c = 0,08 \times 276 \rightarrow T_c = 22,08^\circ\text{C}$$

O bloco responsável por enviar os dados pelo barramento CAN está ilustrado na Fig. 11. O detalhe “a” é a entrada dos valores de velocidade e o detalhe “b” é a entrada dos valores de temperatura.

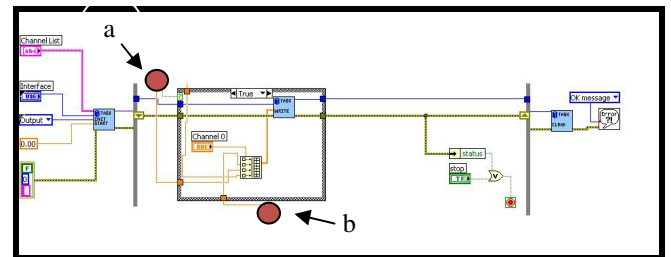


Fig. 11: Diagrama de VIs responsável pelas transmissões pelo CAN.

IV. CONCLUSÕES

A criação rápida de ferramentas para apoiar o desenvolvimento de equipamentos para a realização de testes de integração de sensores ou até mesmo para a simulação da comunicação de dados é um recurso de grande utilidade. O ambiente de programação LabVIEW proporciona rapidez e facilidade no desenvolvimento dos softwares com essas funções, sendo uma boa opção quando se deseja capacitar profissionais da área de integração de sistemas. É também uma ferramenta interessante para apoiar a instrução quando o objetivo é ensinar e demonstrar a integração de *hardware* e *software* utilizados em sistemas embarcados.

No trabalho puderam-se mostrar as formas de realização das aquisições de dois sinais diferentes; o processamento desses dados; como se utilizam as fórmulas de conversão e a apresentação das grandezas físicas nos instrumentos virtuais do software. Com esse aplicativo é possível ao capacitando compreender os princípios elementares de um sistema embarcado, onde um sensor recebe o fenômeno físico, transforma esse fenômeno em sinal elétrico e o envia digitalizado para os sistemas utilizando um barramento de dados padronizado.

No primeiro caso desse estudo, a demonstração é feita utilizando o transdutor de pressão diferencial que recebe as pressões ambiente e do banco portátil e a transforma em sinal elétrico entre 0V e 5V. O sinal elétrico é recebido pelo dispositivo de aquisição (NI USB-6251) e é enviado

para o computador por um canal de comunicação USB. O aplicativo AqGeraDado apresenta para o capacitando o valor da tensão proveniente do transdutor e o valor de velocidade.

No segundo caso, a temperatura é o fenômeno físico permitindo ao capacitando perceber e compreender a transformação desse fenômeno em sinal elétrico e sua apresentação no instrumento virtual.

Por fim, as informações dos sensores são transferidas de um computador para outro utilizando um protocolo de comunicação padronizado. É possível ao capacitando verificar os sinais elétricos e os seus respectivos valores em hexadecimal através da utilização de um osciloscópio. No computador receptor dos dados provenientes do CAN, o capacitando pode observar as informações de velocidade e temperatura convertidas de hexadecimal para unidades de engenharia.

V. REFERÊNCIAS

- [1] National Instruments, 2006 “NI-CAN *Hardware and Software Manual*”, out 2006, Austin, USA.
- [2] R. Bosch GmbH, “CAN Specification”, Version 2.0, 1991, <http://www.bosch-semiconductors.de/en/ubk_semiconductors/safe/ip_modules/can_literature/can_literature.html>, set. 1991, Stuttgart.
- [3] McRoberts, M., 2012 “Arduíno Básico” Novatec Editora Ltda, São Paulo, SP, Brasil, Jan. 2012.
- [4] Regazzi, R.D., Pereira P.S. e Silva Jr M.F., 2005 “Soluções Práticas de Instrumentação e Automação – Utilizando a Programação Gráfica LabVIEW” Ed.Wexley Ronald – Gráfica KWG, 2005, Rio de Janeiro, Brazil.
- [5] Euphrásio P.C.S, 2012, “Aspectos Comparativos entre os Barramentos de dados MIL-STD-1553B e CANBus”, XIV SIGE, São José dos Campos, Brasil, ITA, Set. 2012.
- [6] Lun W. Ng, Kyun C. Ng, Ali B. M., Noordin N. K., e Rokhani F. Z., 2010 “Review of Researches in Controller Area Networks Evolution and Applications” Universiti Putra Malaysia, Malaysia, 2010.
- [7] A. E. A. Neto, L. C. C. Lisboa, L. M. Alves, B. R. Dutra “Barramentos de Dados em Sistemas Aviônicos” Trabalho de Conclusão de Curso CEV-I, GEEV-DCTA, São José dos Campos, Nov. 2007.