MÓDULO PARA REDES DE TRANSDUTORES INTELIGENTES UTILIZANDO O ATMEGA8

Tércio Alberto dos Santos Filho[1], Alexandre César Rodrigues da Silva[2], Silvano Renato Rossi[3], André Luiz Alves Nogueira[4]*

Abstract *— In start of decade at 90, the IEEE with NIST developed a standard for development transducers networks. The standard does not define the type of hardware that will be used or program language but the communication interface, protocols and TEDS (Transducer Electronic Data Sheet). The TEDS are "tables" of data that there is the information of the transducers in format electronic. Today, the TEDS there are a format defined based on IEEE 1451.0 standard, a specification appeared in 2007 for support communication interfaces like RS232 (Recommended Standard 232), UART (Universal Asynchronous Receive/Transmitter), SPI (Serial Peripheral Interface) and USB (Universal Serial Bus). In this work is presented a study feature of IEEE 1451.0 standard, protocols and TEDS specifications. For analyze and studying was developed a module TIM with temperature sensor LM35 based on IEEE 1451.2 and IEEE 1451.0 where is presented in this paper.*

Index Terms — IEEE 1451.0, protocolo, comunicação, interface, TEDS.

INTRODUÇÃO

O desenvolvimento de sistemas de DMC (*Distributed Measurement and Control*) demanda um grande esforço por parte de engenheiros e projetistas da área de instrumentação, sendo que alguns desses sistemas são desenvolvidos simplesmente para uma determinada aplicação, utilizando ferramentas de alto custo e softwares proprietários.

Ao analisar os esforços no desenvolvimento de sistemas DMC, no início da década de 90, foram aprovadas as primeiras diretrizes para o padrão IEEE (*Institute of Electrical and Electronic Engineers*) 1451, um padrão de interfaceamento para transdutores inteligentes. As diretrizes foram desenvolvidas pelo IEEE em parceria com o NIST (*National Institute of Standards and Technology*) para a conexão de transdutores em rede de forma padronizada. Hoje, o padrão IEEE 1451 possui diversos conceitos de interoperabilidade e é dividido em vários comitês como: IEEE 1451.0, IEEE 1451.1, IEEE P1451.2, IEEE 1451.3, IEEE 1451.4, IEEE 1451.5, IEEE P1451.6 e IEEE P1451.7 [1]. O padrão IEEE 1451 tem como objetivo definir o interfaceamento do nó NCAP (IEEE 1451.1) e os diferentes módulos de acordo com cada comitê. Através do emprego do padrão IEEE 1451 é possível atingir interoperabilidade e modo de funcionamento *plug and play* entre o módulo NCAP e as diferentes tecnologias de interfaceamento. Tal característica é definida no padrão IEEE 1451.0 [2].

Em 2007, o IEEE lançou a atualização da norma IEEE 1451.0, revisada e modificada para atender as diferentes interfaces de comunicação. Neste trabalho, apresentam-se as principais características de desenvolvimento dos protocolos e as TEDS definidas no novo padrão IEEE 1451.0-2007 e a implementação do TIM no microcontrolador ATMEGA8.

O PADRÃO IEEE 1451.0-2007

O padrão IEEE 1451.0-2007 aponta uma funcionalidade comum, comandos, protocolos e TEDS (*Transducer Electronic Data Sheets*) da família do padrão de transdutores inteligentes. Esta funcionalidade é independente do meio físico de comunicação, isto inclui as funções básicas requeridas para controle e gerenciamento de transdutores inteligentes, protocolos de comando e independência do formato de mídia. O padrão IEEE 1451.0-2007 especifica o formato das TEDS que são "tabelas" de dados que contêm informações dos transdutores (sensores/atuadores) armazenadas em uma memória não volátil dentro do TIM (*Transucer Interface Module*). No entanto, há aplicações em que o armazenamento em uma memória não volátil não é prático para a aplicação, então, o padrão IEEE 1451.0 permite o armazenamento das informações das TEDS em outras mídias de armazenamento de forma remota. Tais sistemas, denominados de TEDS "Virtuais", são arquivos eletrônicos que fornecem as mesmas funcionalidades do que implementadas em memória não volátil [2].

Os TIMs (*Transducer Interface Module*), ao serem conectados ao NCAP, transmitem as informações das TEDS ao gerenciador de protocolo, que faz o reconhecimento da

1Tércio Alberto dos Santos Filho, DEE, FEIS, UNESP- Univ Estadual Paulista, Av. José Carlos Rossi s/n, Campus III, 15385-000, Ilha Solteira SP, tercioas@gmail.com.
2 Alexandre C. R. Silva, DEE, FEIS, UNESP- Univ Estadual Paulista, Av. José Carlos Rossi s/n, Campus III, 15385-000, Ilha Solteira SP, acrsilva@dee.feis.unesp.br.
3 Silvano R. Rossi, Universidad Nacional del Centro de la Provincia de Buenos Aires – UNCPBA, Argentina, srossi@fio.unicen.edu.ar.
4 André Luis Alves Nogueira, DEE, FEIS, UNESP- Univ Estadual Paulista, Av. José Carlos Rossi s/n, Campus III, 15385-000, Ilha Solteira SP, andre.engele@uol.com.br.

rede de transdutores de forma automática, ou seja, *plug and play*. A característica *plug and play* dos transdutores inteligentes aos seus usuários e desenvolvedores faz surgir algumas vantagens como: redução do tempo para parametrização do sistema, diagnósticos avançados, redução do tempo para reparo e reposição, gerenciamento avançado do hardware e automatização da calibração.

Nos Tópicos 3 e 4 serão abordadas as padronização da estrutura de comando e a descrição das TEDS de acordo com o padrão IEEE 1451.0-2007.

ESTRUTURA DOS COMANDOS

O padrão IEEE 1451.0-2007 define a estrutura das mensagens enviada através do MCI (Módulo de Comunicação da Interface) formando um protocolo de comunicação. A estrutura das mensagens de comando que são enviadas através da interface padronizada deve possuir em sua estrutura um grupo de octetos com os campos:

- **Número do canal de transdutor de destino:** indica o número do canal de destino representado por 2 bytes.
- **Classe do comando:** indica a classe do comando quando o transdutor está no estado de operação. Cada classe de comando possui suas funções que é definida de acordo com a norma IEEE 1451.0 em [2].
- **Função do comando:** função do comando a ser executado definido de acordo com a classe do comando.
- **Tamanho:** formado por 2 octetos, indica o número de octetos dependendo da mensagem, ou seja, quantos octetos são esperados no campo de dados.
- **Dados:** representa o campo de dados

Na Tabela I apresenta-se a estrutura da mensagem utilizada e as especificações dos campos.

TABELA I
ESTRUTURA DA MENSAGEM DO COMANDO.

1 Octeto							
7	6	5	4	3	2	1	0
Número do canal de transdutor de destino (MSB)							
Número do canal de transdutor de destino (LSB)							
Classe do comando							
Função do comando							
Tamanho (MSB)							
Tamanho (LSB)							
Dados							

Para responder uma mensagem de comando enviado pelo NCAP é utilizada a estrutura apresentada na Tabela II, que é composto pelos campos:

- **Sucesso/Falha:** se o byte for diferente de 0 (zero), indica que o comando foi completado com sucesso. Caso o comando seja 0 (zero), o comando falhou e o sistema poderia checar a situação para identificar o problema.
- **Tamanho:** representa o número de octetos do campo de dados da mensagem.
- **Dados:** representa o campo de dados.

TABELA II
ESTRUTURA DA MENSAGEM DE RESPOSTA.

1 Octeto							
7	6	5	4	3	2	1	0
Sucesso/Falha							
Tamanho (MSB)							
Tamanho (LSB)							
Dados							

DEFINIÇÃO DOS TEDS

O processo de construção dos TEDS consiste em seguir os formatos descritos pela norma e atribuir os valores para cada um dos campos conforme o TIM que está sendo descrito. Neste artigo, apresenta-se a descrição do TEDS conforme o padrão IEEE 1451.0-2007. O padrão IEEE 1451.0-2007 específica uma estrutura T/L/V (*Type/Lenght/Value*) para descrição de suas tabelas. Para a descrição do TEDS em qualquer TIM, 4 (quatro) tabelas são necessárias e as outras demais são opcionais. As TEDS necessárias são [2]:

- Meta-TEDS;
- TransducerChanel TEDS;
- User´s Transducers Name TEDS;
- PHY TEDS.

Para o processo de apresentação dos TEDS, neste artigo serão apresentados os campos obrigatórios e alguns campos relevantes iniciando pela tabela Meta-TEDS. Para a descrição das tabelas dos TEDS, será descrito 1 (um) sensor de temperatura, o LM-35.

META-TEDS

A Meta-TEDS tem como objetivo disponibilizar todas as informações necessárias para obter acesso a qualquer canal de transdutor e definir uma identificação única para o TIM. Na Tabela III apresentam-se os campos obrigatórios dos Meta-TEDS que foram preenchidos como [2]:

- **Lenght** – Número de octetos utilizados para descrever as Meta-TEDS incluindo, Campo, Comprimento (Comp.) e os 2 (dois) octetos do Checksum.
- **TEDSID** – identificação das TEDS, representado por 4 octetos. O primeiro octeto indica que esse TEDS faz parte da norma IEEE 1451.0, o segundo octeto representa quais TEDS está sendo descrita, no qual está sendo representada nesta tabela as Meta-TEDS. O terceiro octeto trata-se da identificação da versão do TEDS, sendo para este TEDS trabalha com a versão atual definido pela norma como valor 1. O ultimo octeto representa o número de octetos do campo comprimento.
- **UUID** (*Universal Unique Identification*) – o UUID é um campo de identificação associado com o TIM no qual o valor é único. O campo é representado por 10 octetos longos que consiste de quatro subcampos: localização, fabricante, ano e tempo. Neste contexto, para representação de acordo com o padrão IEEE 1451.0, utilizam-se 42 bits iniciais para representar a localização, sendo, o MSB fixado em 1_2 representa o Norte e 0_2 o Sul, os próximos 20 bits mais significativos representam a

latitude em arcosegundos utilizando números inteiros. O próximo bit fixado em 1_2 representa o Leste e 0_2 o Oeste, os 20 bits restantes representa a longitude utilizando números inteiros para representar o arcosegundos. O campo fabricante é representado por 4 bits, por haver apenas um fabricante, este campo é 0. O ano é representado pelos próximos 12 bits. Os 22 bits restantes representam a identificação do TIM, um código único e identificação desenvolvido pelo fabricante.

- **OHoldOff** – um comando sem resposta enviado para o TIM deve ser representado como uma operação falha. Valor que é representado em ponto flutuante de acordo com a norma IEEE 754. Neste trabalho, definiu-se 5 segundos, após, o comando é considerado falho.
- **TestTime** – tempo para a realização do auto-teste do TIM.
- **MaxChan** – quantidade de canais de transdutores implementado no TIM.
- **Checksum** – representa a soma de todos os octetos dentro da Meta –TEDS sendo representado por 2 octetos.

TABELA III
META-TEDS

Campo	Nome	Tipo	Comp.	Valor
-	Lenght	UInt32	4	0x00 0x00 0x00 0x25
3	TEDSID	Uint	4	0x00 0x01 0x01 0x01
4	UUID	UUID	10	0x08 0xFB 0x61 0xB4 0x80 0x81 0xF6 0x43 0xA1 0xB1
10	OHoldOff	Float32	4	0x40 0xA0 0x00 0x00
12	TestTime	Float32	4	0x45 0x20 0x00 0x00
13	MaxChan	UInt16	2	0x00 0x01
---	Checksum	UInt16	2	0x06 0x2F

TRANSDUCERCHANEL TEDS

O TransducerChanel TEDS descrito nas TEDS será para disponibilizar todas as informações de operação e endereçamento do canal de transdutor. O TransducerChanel TEDS é acessado usando comandos de requisição, comandos de leitura das TEDS, escrita das TEDS ou atualização. As TEDS podem ser implementadas apenas para leitura prevenindo que mudanças sejam feitas, pois, poderiam causar incompatibilidade, neste caso, os comandos escrita e atualização não são aplicados. Na Tabela IV apresenta-se a estrutura do TransducerChanel TEDS e o nome de seus campos como [2]:

- **Lenght** – número de octetos utilizados para descrever as TransducerChanel TEDS incluindo, Campo, Comprimento (Comp.), Valor e os 2 (dois) octetos do Checksum.
- **TEDSID** – identificação das TEDS, campo semelhante a identificação das Meta-TEDS, porém, o segundo octeto é representado através do valor 3 que identifica TransducerChanel TEDS.
- **CalKey** – este módulo de transdutor não irá realizar nenhum tipo de calibração. Segundo o padrão, o valor para representar o comportamento é 0_{10} (zero).
- **ChanType** – utilizado para definir o tipo de canal de transdutor: sensor, atuador ou sensor de evento. Para definir um sensor deve-se fixar o valor 0, para atuador o valor 1 e sensor de evento 2.
- **PhyUnits** – utilizado para definir o tipo de unidade física descrita pelo canal. Este campo é subdividido em 10 subcampos, no qual para sua representação utiliza-se o Sistema Internacional (SI) de unidades. Quando o expoente dessas unidades for igual a 0_{10}, o campo deve receber o valor 128, caso a unidade for fixada, deve-se utilizar o valor 130. Para a representação foi utilizado um sensor de temperatura no qual utiliza-se a unidade em Kelvin, sendo fixado o valor 130_{10} no campo 57. De acordo com [2] na Tabela 48 descreve os tipos de unidades disponíveis. Para a representação em formato TLV os campos seguem o seguinte padrão:
  - Campo - 12: representa a PhyUnits no TransducerChanel TEDS.
  - Tamanho - 6: quantidade de octetos para descrever os comandos na linha (tupla).
  - Tipo de unidade – 50: representa as unidades disponíveis no padrão, outras unidades podem ser visualizadas em [2] na Tabela 2.
  - Tamanho – 1: quantidade de octetos necessários para descrever o tipo de unidade física.
  - Estrutura da unidade física – 0: representa a estrutura da unidade física utilizada para descrever o valor da unidade, neste trabalho, utilizou-se a PUI_SI_UNITS, que representa o Sistema Internacional. Na Tabela 3 em [2] apresentam-se as estruturas das unidades físicas para a descrição da unidade.
  - Unidade física – 57: tipo da unidade física utilizada, neste trabalho utilizou-se Kelvins.
  - Tamanho – 1: número de octetos utilizados para representar de acordo com a estrutura da unidade física.
  - Valor – 130: utilizado para apresentar a unidade linear.
- **LowLimit** – define o limite mínimo do sensor de temperatura, para o sernsor de temperatura descrito neste trabalho, foi utilizado 277,15 K.
- **HighLimit** – define o limite máximo do sensor de temperatura, para o sensor de temperatura descrito neste trabalho, foi utilizado 328,15 K.
- **Oerror** – variação de erro do sensor de temperatura atribuiu-se 273,65 K que representa 0,5 ºC.
- **SelfTest** – para realizar o auto-teste do sistema, fixou-se o valor 1_{10}, caso o TIM não possua a capacidade de auto-teste, fixa-se o valor 0_{10}.
- **Sample** – este campo é representado por 3 (três) subcampos obrigatórios cada um contendo 3 octetos. Os 3 (três) subcampos possuem o tipo, comprimento e o valor. O subcampo *Data Model* representado pelos 3 primeiros octetos define o tipo de dados, neste trabalho, definiu-se um

inteiro de 8 bits. O subcampo *Data model lenght* representado pelos 3 octetos seguintes, define o tamanho do campo para realizar a representação específica. Os 3 últimos octetos representa o subcampo *Model significant bits* que neste contexto, utilizou-se 8 bits indicando a saída do conversor A/D (Analógico/Digital).

- **UpdateT** – Este transdutor está no modo "free-running" e está atualizando as amostras a uma taxa de 10/s. Portanto, o valor deste campo é 0,1, que é o tempo máximo entre a recepção da leitura e de ter uma amostra disponível.
- **RSetupT** – este campo contém o tempo máximo (t_{rs}), expresso em segundos, entre o recebimento do *trigger* pelo módulo de transdutor e do tempo que os dados estão disponíveis para ser lido. O tempo de configuração de leitura para este transdutor é de 0,25µs.
- **Speriod** – o período de amostragem mínimo dos sensores é de 1 décimo de segundo, assim, foi atribuído o valor correspondente em ponto flutuante.
- **WarmUpT** – tempo utilizado para carregar as TEDS no NCAP.
- **RDelay** – pior caso de atraso dos dados existentes é 25ms.
- **Sampling** – o transdutor esta sempre rodando em modo "*Free Running*" sem "*Pre-trigger*". Nesta descrição dos TEDS foi fixado o valor 2_{10} de acordo com [2] descrito na Tabela 54.
- **Checksum** – representa a soma de todos os octetos dentro do TransducerChanel TEDS no qual o valor é representado por 2 octetos.

TABELA IV
ESTRUTURA DO TRANSDUCERCHANEL TEDS.

Campo	Nome	Tipo	Comp.	Valor
	Lenght	UInt32	4	0x00 0x00 0x00 0x53
3	TEDSID	UInt8	4	0x00 0x03 0x01 0x01
10	CalKey	Uint8	1	0x00
11	ChanType	UInt8	1	0x00
12	PhyUnits	UNITS	6	0x0C 0x06 0x32 0x01 0x00 0x39 0x01 0x82
13	LowLimit	Float32	4	0x40 0x80 0x00 0x00
14	HiLimit	Float32	4	0x41 0x40 0x00 0x00
15	Oerror	Float32	4	0x3f 0x00 0x00 0x00
16	SelfTest	UInt8	1	0x01
18	Sample	-	9	0x28 0x01 0x00 0x29 0x01 0x01 0x30 0x01 0x08 .
20	UpdateT	Float32	4	0x3D 0xCC 0xCC 0xCD
22	RSetupT	Float32	4	0x37 0xD1 0xB7 0x17
23	Speriod	Float32	4	0x3d 0xcc 0xcc 0xcd
24	WarmUpT	Float32	4	0x41 0xF0 0x00 0x00
25	Rdelay	Float32	4	0x37 0xD1 0xB7 0x17
31	Sampling		1	0x02
-	Checksum	UInt16	2	0x10 0x4E

USER´S TRANSDUCERS NAME TEDS

O User´s Transducers Name TEDS descrito nos TEDS é necessário para todos os TIMs e é recomendado para todos os transdutores. O User´s Transducers Name TEDS disponibiliza um espaço para o armazenamento do nome por qual o sistema ou o usuário final saberá qual será o transdutor. O User´s Transducers Name TEDS pode ser associado com o TIM ou com o TransducerChanel TEDS que são acessadas usando requisições através de comando, escrita, leitura ou uma atualização [2]. Na Tabela V especifica-se os campos User´s Transducers Name TEDS.

TABELA V
ESTRUTURA DO USER´S TRANSDUCERS NAME TEDS.

Campo	Nome	Tipo	Compr.	Valor
-	Lenght	UInt32	4	0x00 0x00 0x00 0x19
3	TEDSID	UInt32	4	0x00 0x0c 0x01 0x1
4	Format	UInt8	1	0x00
5	TCName	UInt8	12	0x41 0x54 0x4d 0x45 0x47 0x41 0x38 0x2d 0x4c 0x4d 0x33 0x35
-	Checksum	UInt16	2	0x03 0x5F

Os campos do User´s Transducers Name TEDS são definidos como [2]:

- **Lenght** – número de octetos utilizados para descrever as User´s Transducers Name TEDS incluindo, Campo, Comprimento (Comp.), Valor e os 2 (dois) octetos do Checksum.
- **TEDSID** – identificação das TEDS, campo semelhante à identificação das Meta-TEDS e TransducerChanel, porém, o segundo octeto é representado através do valor 12_{10} que identifica o User´s Transducers Name TEDS.
- **Format** – define o formato da representação do campo TCName, sendo que, 0_{10} é uma representação definida pelo usuário de acordo com [2] na Tabela 74.
- **TCName** – descreve o nome do TIM e o canal de transdutor de acordo com o padrão definido no campo Format. Neste trabalho utilizou-se o padrão ASCII (*American Standard Code for Information Interchange*) para definir o nome do TIM como ATMEGA8-LM35.
- **Checksum** – representa a soma de todos os octetos dentro do User´s Transducers Name TEDS no qual o valor é representado por 2 octetos.

PHY TEDS

As PHY TEDS são necessárias em todos os TIMs e é definida de acordo com cada norma de interface definida pelo padrão IEEE 1451. A característica básica das PHY TEDS é disponibilizar as informações necessárias de toda interface de comunicação, mais as informações comuns para todos os canais. Os octetos da PHY TEDS são constantes e apenas para leitura.

Neste trabalho, as PHY TEDS foram elaboradas de acordo com a norma IEEE p1451.2 padrão RS232 descritas de acordo com artigo apresentado por Eugene Y. Song e Kang Lee em [3].

DESENVOLVIMENTO DO TIM

O desenvolvimento do TIM foi realizado utilizando um microcontrolador da ATMEL, o ATMEGA8. A parte lógica do TIM foi realizada utilizando o ambiente de desenvolvimento BASCOM-AVR que permite ao projetista verificar erros no código, realizar simulações e programar no microcontrolador [4] [5].

Para a leitura/controle do canal de transdutor ou para realizar a leitura dos TEDS cada protocolo recebido pelo TIM deve ser identificado para que seja realizada sua devida tarefa. Neste contexto, para verificação das tarefas requisitadas pelo NCAP foi realizada comparações entre os valores recebidos pela interface serial RS232 e os protocolos armazenados no microcontrolador como, por exemplo: protocolo de leitura das META-TEDS, ChannelTEDS, USER´S NameTEDS, PHY TEDS, leitura e escrita do canal de transdutor. Ao finalizar a verificação de cada campo do protocolo o TIM é capaz de identificar se o protocolo recebido faz parte do padrão IEEE 1451 e realizar a requisição feita pelo NCAP. Ao finalizar a execução da tarefa requisitada pelo NCAP, os dados são armazenados dentro do protocolo de resposta apresentada na **TABELA** II e transmitidos para o nó de rede NCAP padronizado de acordo com a norma IEEE 1451.1. Na Figura 1 apresenta-se o sistema desenvolvido em laboratório contendo um sensor de temperatura LM35, um amplificador operacional (AOP), conversor A/D, microcontrolador ATMEGA8 e um conversor MAX232 para amplificar/reduzir sinais TX, RX CTX e RTS.

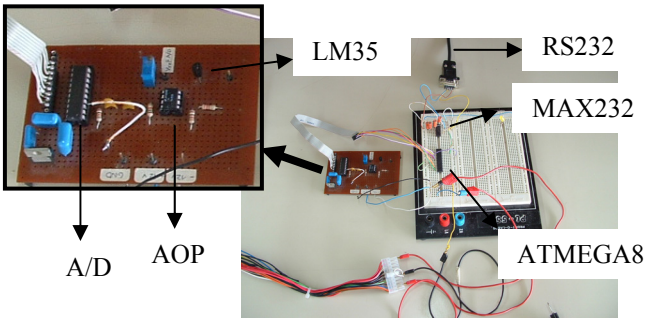

FIGURA 1
SISTEMA DESENVOLVIDO EM LABORATÓRIO.

CONCLUSÃO

Neste trabalho, apresentaram-se as principais características do padrão IEEE 1451.0-2007 como a descrição dos protocolos de comunicação, os principais campos e o formato do TEDS. O trabalho teve como principal característica realizar o desenvolvimento do TEDS de acordo com a nova norma e destacar os campos obrigatórios para o seu desenvolvimento. Além das características das TEDS, apresentou-se a desenvolvimento do TIM em um microcontrolador de baixo custo com interface serial RS232. O sistema segue as normas do padrão IEEE 1451.0-2007 e ao padrão IEEE p1451.2. O TIM apresentado neste trabalho faz parte de um projeto maior de doutorado e atualmente a parte lógica do NCAP está sendo desenvolvida e testada em laboratório.

AGRADECIMENTOS

Os autores agradecem ao Programa de Pós-Graduação em Engenharia Elétrica – PPGEE. Ao Conselho Nacional de Desenvolvimento Cientifico e Tecnologico – CNPq, número do processo: 140261/2008-7. A FUNDUNESP (Fundação para o Desenvolvimento da UNESP)

REFERÊNCIAS

[1]. ROSSI, R. S; Implementação de um Nó IEEE 1451, Baseado em Ferramentas Abertas e Padronizadas, para Aplicações em Ambientes de Instrumentação Distribuída. Tese de Doutorado. Universidade Estadual Paulista, Faculdade de Engenharia Elétrica FEIS-UNESP Ilha Solteira, 2005.

[2]. LEE, K.; SWEETSER, V. K.; IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats. Disponível em: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4338161&isnumber=4338160>. Acesso em: 19 de janeiro de 2009. pg. 323.

[3]. SONG, E., LEE, K., "Sensor Network based on IEEE 1451.0 and IEEE p1451.2-RS232", I2MTC 2008 – IEEE Measurement Technology Conference, Victoria, Vancouver Island, Canada, May 12-15.

[4]. ATMEL. "8 bit AVR with 8 K Bytes and In-system Programable Flash". Disponível em: <http://www.datasheetcatalog.org/datasheet/atmel/2486S.pdf>. Acesso em: 10 julho 2009.

[5]. MCS Eletronics. "BASCOM AVR Help ? Reference – Making Things Easy". Manual: Bascom AVR Help.