



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO ENGENHARIA ELÉTRICA E INFORMÁTICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**



DANILO PINTO LINS BARRETO

**IMPLEMENTAÇÃO DE UMA REDE DE SENSORES SEM FIO UTILIZANDO A
TECNOLOGIA ZIGBEE**

Campina Grande, Paraíba
Maio de 2012

UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO ENGENHARIA ELÉTRICA E INFORMÁTICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

IMPLEMENTAÇÃO DE UMA REDE DE SENSORES SEM FIO UTILIZANDO A
TECNOLOGIA ZIGBEE

*Trabalho de conclusão de curso submetido à
Unidade Acadêmica de Engenharia Elétrica da
Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção de grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica*

Aluno:

Danilo Pinto Lins Barreto

Orientador:

Prof. D.Sc. José Sérgio da Rocha Neto

Campina Grande, Paraíba
Maio de 2012

DANILO PINTO LINS BARRETO

IMPLEMENTAÇÃO DE UMA REDE DE SENSORES SEM FIO UTILIZANDO A
TECNOLOGIA ZIGBEE

*Trabalho de conclusão de curso submetido à
Unidade Acadêmica de Engenharia Elétrica da
Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção de grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica*

Prof. D.Sc. José Sérgio da Rocha Neto

Professor Orientador

Professor Convidado

Campina Grande, Paraíba
Maio de 2012

AGRADECIMENTOS

Antes de tudo, agradeço a Deus, criador da vida e de tudo que há de bom nela, por me guiar em todos os momentos, principalmente nas dificuldades.

Aos meus pais, Francisco e Valéria, que não mediram esforços para criarem todas as condições favoráveis ao meu desenvolvimento pessoal, acreditando sempre em meu potencial.

A minha namorada, Camila Aranha, que ao longo de todos esses anos juntos, sempre me apoiou e acreditou que um dia eu me tornaria um grande profissional, não me deixando fraquejar pela distância das pessoas que amo.

Aos amigos que encontrei no curso, especialmente Júlio César, Paulo de Tarso, Aristóteles, Carlos, Hugo e Rafael, que dividiram momentos de estudo, alegrias e decepções ao longo de toda caminhada em Campina Grande.

Ao corpo docente de Engenharia Elétrica da Universidade Federal de Campina Grande, que foram imprescindíveis no meu aprendizado de engenheiro, principalmente meu orientador José Sérgio da Rocha Neto e o coordenador de graduação Talvanes Meneses Oliveira, que acreditaram até o fim.

Aos funcionários da coordenação de graduação do curso de Engenharia Elétrica, Tchaikovsky e Adail, que fizeram tudo ao seu alcance para me ajudar em problemas relacionados ao curso.

Aos demais que contribuíram indiretamente ao longo desses anos, aqui, em Campina Grande.

RESUMO

Neste trabalho apresenta-se o desenvolvimento de uma rede de sensores sem fio utilizando tecnologia ZigBee, desde um estudo preliminar dos módulos XBee, finalizando com a análise dos resultados e conclusão sobre as vantagens e desvantagens desta tecnologia.

O foco inicial do projeto é um estudo do protocolo IEEE 802.15.4, onde o padrão ZigBee foi montado. São apresentados elementos como topologia de redes, formas de comunicação entre os módulos, além de características técnicas que definem a qualidade e a capacidade das redes.

Também se apresenta todo o processo de configuração dos módulos, confecção das placas com os sensores e criação da interface para que um usuário tenha controle sobre o processo de leitura de dados.

Palavras-Chave: ZigBee, rede de sensores sem fio, protocolo IEEE 802.15.4, módulos XBee™, LabVIEW Serial Port.

ABSTRACT

This paper presents the development of a network of wireless sensors using ZigBee technology, from a preliminary study of the XBee modules, ending with the analysis of the results and conclusion on the advantages and disadvantages of this technology.

The initial focus of the project is a study of the IEEE 802.15.4 protocol, where ZigBee standard was developed. Elements such topology networks, how communication works between modules and technical features are shown, and how it defines the network quality and capacity.

It also presents the whole module configuration process and how it communicates with a user that controls the process of data reading.

Key-words: ZigBee, wireless sensors network, IEEE 802.15.4 protocol, XBee™ modules, LabVIEW serial port.

LISTA DE FIGURAS

Figura 1 - Topologia de redes IEEE 802.15.4 – (a) Estrela (b) Malha (c) Árvore	13
Figura 2 – Diagrama da transmissão Coordenador-Dispositivo: (a) Redes <i>Beaconing</i> ; (b) Redes <i>Non-Beaconing</i>	15
Figura 3 – Diagrama da transmissão Dispositivo- Coordenador: (a) Redes <i>Beaconing</i> ; (b) Redes <i>Non-Beaconing</i>	15
Figura 4 - Camadas definidas pelos padrões IEEE 802.15.4 e ZigBee	16
Figura 5 – Representação dos módulos XBee: (a) Xbee (b) Xbee-PRO	17
Figura 6 - Estrutura de um frame API.....	19
Figura 7 - Interface do software X-CTU – PC Settings	21
Figura 8 - Interface do software X-CTU – Range Test.....	22
Figura 9 - Interface do software X-CTU – Terminal: (a) Transparente (b) API.....	23
Figura 10 - Interface do software X-CTU – Modem Configuration.....	23
Figura 11 - Placa CON-USBEE	24
Figura 12 - Termistores NTC	27
Figura 13 - Placas Sensoriais.....	28
Figura 14 – Bloco utilizado para configurar parâmetros da porta <i>serial</i>	29
Figura 15 - Bloco utilizado para receber dados pela porta <i>serial</i>	30
Figura 16 - Bloco utilizado para enviar dados pela porta <i>serial</i>	30
Figura 17 - <i>Interface</i> do programa.....	31
Figura 18 – Diagrama do circuito utilizado para traçar a característica do NTC.....	32
Figura 19 – Gráfico da Tensão (V) x Temperatura (°C)	32
Figura 20 - Gráfico das temperaturas internas e externas à sala 08 do LIEC	33
Figura 21 - Geradores de sombra sobre superfície do sensor.....	34

LISTA DE TABELAS

Tabela 1 - Comandos em modo AT.....	18
Tabela 2 - Tipos de frames API.....	20
Tabela 3 - ZigBee vs. Bluetooth.....	25
Tabela 4 - ZigBee vs. Wi-Fi.....	26

SUMÁRIO

1.	INTRODUÇÃO	10
1.1	Motivação	10
1.2	Objetivo	11
2.	FUNDAMENTOS TEÓRICOS.....	12
2.1	Introdução	12
2.2	Histórico	12
2.3	IEEE 802.15.4.....	12
2.3.1	Topologias.....	13
2.3.2	Transferência de Dados.....	14
2.3.3	<i>Beaconing e Non-Beaconing</i>	14
2.3.4	Métodos para a transferência de dados	14
2.4	ZigBee	16
2.4.1	Módulo XBee/XBee PRO (Série 1).....	16
2.4.2	Modos de operação	19
2.4.3	Configuração dos Módulos – Software X-CTU.....	21
2.4.4	Placa CON-USBEE	24
2.5	Outras tecnologias de transmissão sem fio	24
2.5.1	Bluetooth.....	25
2.5.2	Wi-Fi.....	26
2.6	Sensor de Temperatura – NTC	27
3.	DESENVOLVIMENTO DO PROJETO.....	28
4.	ANÁLISE E TESTES DE CAMPO	33
5.	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS.....	35
6.	BIBLIOGRAFIA	36
7.	ANEXO	37

1. INTRODUÇÃO

A crescente necessidade do uso e integração de dispositivos em redes sensoriais é um assunto delicado, uma vez que estes formam um grupo específico quanto ao tipo de aplicação em que são utilizados. A demanda pela flexibilidade da rede, seja pela sua configuração ou tamanho, exige o uso de dispositivos com comunicação sem fio, de baixo custo e com alto grau de confiança, que possam ser usados em diferentes tipos de ambientes, que podem variar de uma fábrica a uma residência.

As redes de sensores sem fio diferem bastante de outras redes, como por exemplo, a internet. As redes de sensores apresentam fatores como a integração com outras redes de dados e a elevada interação com o ambiente.

1.1 Motivação

Existem vários padrões de comunicação para redes com altas taxas de transmissão, como para rede de computadores pessoais. Porém, para os padrões de comunicação de sensores sem fio e dispositivos de controle, pouco se tem para suprir as necessidades de comunicação. As principais características destes dispositivos são: consumo de energia limitado, baixa latência, custo baixo e a possibilidade de implementação da rede com vários dispositivos.

Algumas tecnologias atualmente existentes no mercado são *Bluetooth*, *ZigBee* e *Wi-Fi*. Dentre estas, a melhor opção para exercer a comunicação de redes sensoriais é a *ZigBee*, pois *Bluetooth* e *Wi-Fi* possuem, ainda, um grande consumo de energia e alta latência para esta aplicação.

Em 2004, uma aliança de empresas de diferentes segmentos de mercado define um padrão de comunicação sem fio projetado para ser confiável e com baixo consumo de energia. Surge, então, o padrão *ZigBee*, da *ZigBee Alliance* [1]. Este padrão é baseado em outro, o IEEE 802.15.4, implementando assim as camadas MAC (*Medium Access Control*) e PHY (*Physical Layer*). A tecnologia *ZigBee* permite comunicações nas frequências ISM (*Industrial, Scientific and Medical*), sendo na Europa de 868 MHz (1 canal), 915 MHz (10 canais) nos Estados Unidos e 2,4 GHz (16 canais) em outras partes do mundo, e não requerem licença para funcionamento.

1.2 Objetivo

Neste trabalho tem-se como objetivo conectar vários sensores, utilizando tecnologia sem fio, neste caso, *ZigBee*, fazendo assim, uma rede de sensores sem fio de baixo custo.

Para tanto, serão utilizados alguns sensores de temperatura, como o NTC.

2. FUNDAMENTOS TEÓRICOS

2.1 Introdução

Em um ambiente, seja ele residencial ou industrial, os dispositivos de controle, como também os sensores (temperatura, pressão, movimento, etc.), necessitam de um baixo consumo de energia a fim de manter um maior tempo de vida útil de suas fontes de energia (baterias, pilhas, etc.). Já são bastante comuns ambientes residenciais com aparelhos de TV, micro-ondas, condicionadores de ar, etc., que possuem uma rede local sem fio trabalhando em conjunto com atuadores para manter um padrão estável em seus interiores. A tecnologia que melhor se enquadra nessa necessidade é a *ZigBee*.

2.2 Histórico

O Padrão *ZigBee* tem seu início em 1998, quando um grupo de visionários define a *ZigBee Alliance* como um ecossistema de empresas. Em 2003, o padrão IEEE 802.15.4 foi concluído e, em dezembro do ano seguinte, foi aprovada a especificação *ZigBee*, definido o padrão ZigBee 1.0 [6].

Dois anos depois surge a *ZigBee 1.1*, que contém a maioria das especificações da versão atual.

Em 2007, são apresentados dois perfis de funcionamento: o *ZigBee* e o *ZigBee PRO*.

Atualmente, várias empresas fazem parte do grupo *ZigBee Alliance*. Entre elas estão a Samsung, Philips, Huawei, Texas Instruments, Motorola, Siemens, etc. [6].

A ideia do nome surge da analogia com uma colmeia de abelhas: durante um vôo, as abelhas em seus movimentos de zigue-zague trocam informações entre si sobre a direção, distância e localização de seu alimento.

2.3 IEEE 802.15.4

Criado em 2003, o IEEE 802.15.4 é um conjunto de especificações que definem o protocolo e a comunicação entre dispositivos via rádio dentro de uma PAN (*Personal Area Network*) [2], com baixa potência, baixa taxa de transmissão, pouca complexidade e pequeno

alcance. O padrão utiliza um mecanismo para evitar colisões e suporta diversas topologias de rede. A conectividade é monitorada por um coordenador, conhecido como *PAN Coordinator*, e o padrão especifica dois tipos de PHY's, com taxas de transmissão de 20 kb/s e 40kb/s para uma PHY de 868/915 MHz ou 250 kb/s, para uma PHY de 2,4GHz. A PHY escolhida depende de regulamentações locais e da necessidade do usuário.

Outras características da rede são endereços de 64 ou 16 *bits*, alocação de intervalos de tempo garantidos (GTS), confiabilidade na transferência e detecção de energia (ED). O ZigBee é todo baseado neste padrão.

2.3.1 Topologias

O padrão IEEE 802.15.4 define basicamente três tipos de topologia de rede. Elas estão divididas em: Estrela, Malha ou Árvore. Na Figura 1, é possível observar os tipos dessas configurações.

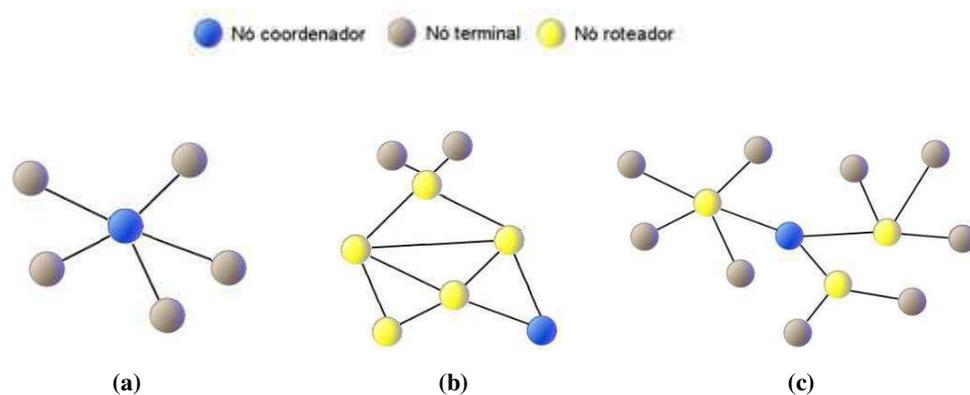


Figura 1 - Topologia de redes IEEE 802.15.4 – (a) Estrela (b) Malha (c) Árvore

Os dispositivos que compõem essas topologias são classificados em dois tipos, de acordo com a sua funcionalidade:

- Dispositivos FFD (*Full Function Device*) – São dispositivos mais completos, que podem desempenhar qualquer função dentro de uma rede (coordenador, roteador ou dispositivo terminal – envia as informações requeridas pelo coordenador).

- Dispositivos RFD (*Reduced Function Device*) – ao contrário dos dispositivos FFD, estes exercem apenas a função de dispositivos terminais. Estão localizados nas extremidades das topologias, pois não possuem a capacidade de retransmitir informação, apenas se comunicando com dispositivos FFD [2].

2.3.2 Transferência de Dados

Cada dispositivo dentro da rede precisa de um endereço único. O padrão IEEE 802.15.4 utiliza dois métodos de endereçamento, um de 64 *bits* (endereçamento estendido) e um de 16 *bits* (endereçamento curto). Como vantagem do endereçamento maior, tem-se um número altíssimo de dispositivos que podem permanecer na rede, cerca de 2^{64} nós, por outro lado, o endereçamento curto diminui a quantidade de memória utilizada para armazenar o endereço.

2.3.3 *Beaconing e Non-Beaconing*

Basicamente, existem dois tipos de acesso ao canal numa transmissão em rede ZigBee (*contention based* ou *contention free* [3]). No *contention based*, os dispositivos usam o mecanismo CSMA-CA (*Carrier Sense Multiple Access – Collision Avoidance*) e, o primeiro que encontra o canal livre transmite. Já no *contention free*, o coordenador disponibiliza um *slot* de tempo para que cada dispositivo transmita, conhecido como GTS (*guaranteed time slot*).

Neste contexto surgem as redes classificadas com ou sem *beacon*. Para garantir que os dispositivos estejam sincronizados, ou seja, garantir que cada dispositivo só transmitirá no seu espaço de tempo, o coordenador envia um sinal de *beacon* e, assim, os dispositivos começam a transmitir. As redes que não possuem este sinal jamais poderão garantir o GTS.

A vantagem do uso deste sinal é que os dispositivos “acordam” com menor frequência em relação às redes sem o mesmo, saindo do modo *sleep* apenas quando precisam receber o sinal de sincronização.

2.3.4 Métodos para a transferência de dados

A transmissão de dados pode ter dois sentidos distintos: Coordenador para Dispositivo, Dispositivo para Coordenador e Entre dispositivos.

- Coordenador para Dispositivo: Inicialmente, o coordenador envia um *beacon* para o dispositivo ao qual deseja se comunicar, informando que tem uma informação a ser enviada. O dispositivo envia um sinal de resposta informando que está pronto para receber a mensagem. Por fim, o coordenador envia a mensagem desejada, com o sinal de *acknowledgment* opcional.

Em redes sem *beacon*, o coordenador espera o dispositivo pedir que a informação seja enviada. Se não houver algo a ser enviado, o coordenador apenas envia um sinal de *acknowledgment* ou uma mensagem com tamanho nulo. A Figura 2 apresenta um diagrama de como é feita esta transmissão.

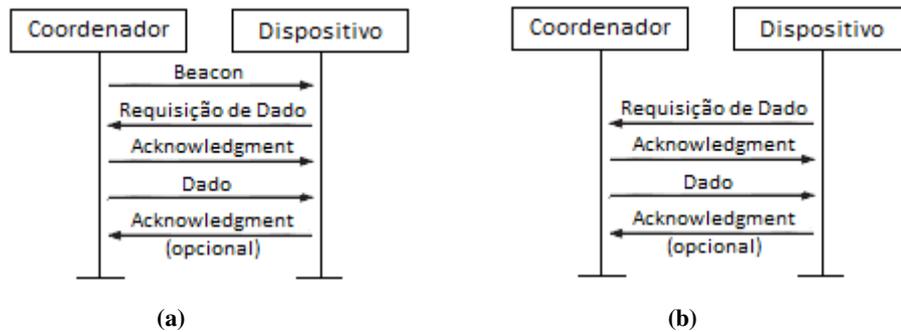


Figura 2 – Diagrama da transmissão Coordenador-Dispositivo: (a) Redes *Beaconing*; (b) Redes *Non-Beaconing*

- Dispositivo para Coordenador: Basicamente, o dispositivo aguarda o sinal para sincronização em redes com *beacon* e envia a informação no seu espaço de tempo determinado. Caso não haja este sinal, o dispositivo simplesmente testa se há alguém transmitindo naquele momento; se não, ele envia informação para o coordenador. A Figura 3 apresenta um diagrama de como é feita esta transmissão.

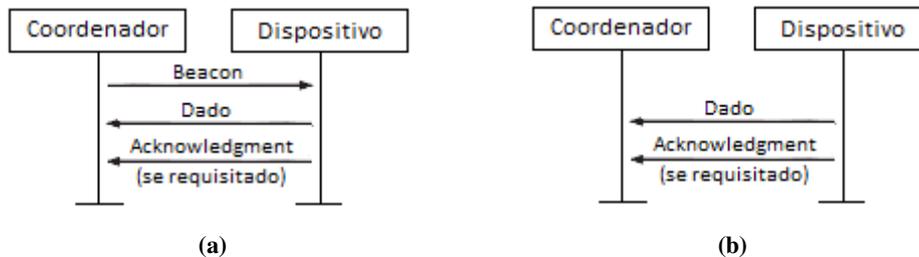


Figura 3 – Diagrama da transmissão Dispositivo- Coordenador: (a) Redes *Beaconing*; (b) Redes *Non-Beaconing*

- *Peer-to-Peer* (Par-a-Par): Os dispositivos se comunicam diretamente de forma sincronizada.

2.4 ZigBee

Embora existam outras tecnologias no mercado, a que melhor se encaixa nas aplicações para redes de sensores e controle é a *ZigBee*. Parâmetros como tempo de acesso à rede, baixo consumo, custo e flexibilidade, são atrativos para esta tecnologia em projetos que necessitam de automação.

É importante entender bem onde o padrão ZigBee define protocolos. O padrão IEEE 802.15.4 é responsável pelas definições nas camadas físicas e enlace (modelo OSI); já o padrão ZigBee define as camadas de rede e aplicação. Na Figura 4 estão representados os campos de definição de cada padrão.

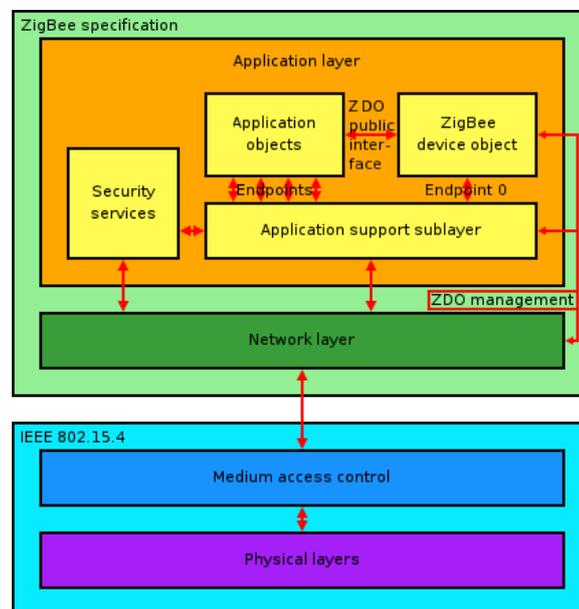


Figura 4 - Camadas definidas pelos padrões IEEE 802.15.4 e ZigBee

2.4.1 Módulo XBee/XBee PRO (Série 1).

Para a implementação da rede sem fio, foram utilizados módulos XBee, que possuem o protocolo *ZigBee* e são fabricados pela Digi Internacional (www.digi.com). Existe ainda uma versão que diferencia apenas no alcance (ou potência do sinal), conhecida como XBee PRO. As características técnicas desses módulos encontram-se no anexo. Na Figura 5, é possível observar ambas as versões deste módulo.



Figura 5 – Representação dos módulos XBee: (a) Xbee (b) Xbee-PRO

Cada módulo possui vários pinos I/O, que podem ser configurados como entradas ou saídas (ADC – conversor analógico/digital, entrada ou saída digital).

Os módulos XBee, independentemente do modo de operação, possuem um modo de comando, onde são modificados os mais diversos parâmetros do módulo. Para fazer essas modificações, o usuário deve primeiramente entrar nesse modo, onde os caracteres digitados serão interpretados como comandos. Dois modos de comando são suportados: o modo de comando AT e o modo através de comandos API.

Por padrão, para entrar no modo de comando AT, deve ser enviada a sequência de três caracteres “+++” dentro de um segundo, após um segundo sem envio/recepção de dados pelo módulo. Este modo permanece até que um comando para finalizar seja enviado ou nenhum comando seja percebido durante um segundo [5].

Para enviar comandos AT, a sintaxe a seguir deve ser seguida:

AT + **Comando** + **Espaço** + **Parâmetro** + **Carriage**
Prefixo **ASCII** **(opcional)** **(opcional, HEX)** **Return**

Por exemplo: ATDL 1F<CR> - Este comando modifica o byte menos significativo do endereço de destino (DL = *Destination Address Low*) para 0x1F. Para sair do modo de comando AT, basta utilizar o comando ATCN ou não enviar nenhum comando AT válido por um segundo.

Na Tabela 1, apresentam-se alguns comandos AT bastante utilizados para configurar módulos XBee.

Tabela 1 - Comandos em modo AT

Comando AT	Descrição
ND	<i>Node Discover</i> – este comando serve para identificar todos os dispositivos que estão na rede. Como resposta, o coordenador recebe os endereços de 64 e 16 <i>bits</i> , como o nome em ASCII de cada módulo.
DH	<i>Destination Adress High</i> – Consulta/modifica a parte mais significativa do endereço destino.
DL	<i>Destination Adress Low</i> – Consulta/modifica a parte menos significativa do endereço destino.
CN	<i>Exit Command Mode</i> – Sai do modo de comando AT.
IR	<i>Sample Rate</i> – Consulta/modifica a taxa de amostras. Quando “setado”, o módulo transmite as amostras a partir de uma quantidade.
IS	<i>Force Sample</i> – Faz uma leitura de todas as entradas (pinos) habilitadas.
D0-D4	<i>Configuration</i> – Escolhe/lê os valores dos pinos 11(AD4/DIO4), 17(AD3/DIO3), 18(AD2/DIO2), 19(AD1/DIO1) e 20(AD0/DIO0).

A programação em modo AT pode ser realizada enviando um comando por vez ou enviando todos em uma linha. A seguir, exemplos para ambas as formas de programação são apresentados.

Comando AT enviado

```
+++
ATDL <enter>
ATDL1A0D <enter>
ATWR <enter>
ATCN <enter>
```

Resposta do sistema

```
OK <CR>
{valor atual de DL} <CR>
OK <CR> modifica DL para 1A0D
OK <CR> Escreve na memória
OK <CR> Sai do modo de comando
```

Comando AT enviado

```
+++
ATDL <enter>
ATDL1A0D,WR,CN <enter>
```

Resposta do sistema

```
OK <CR>
{valor atual de DL} <CR>
OK <CR> OK <CR> OK <CR>
```

2.4.2 Modos de operação

Os módulos XBee podem operar em basicamente dois modos de operação. A diferença entre esses modos está na forma como a informação é enviada, se é em forma de frames ou não.

Por padrão, os módulos XBee transmitem no modo transparente. Quando operam nesse modo, os módulos apresentam uma função semelhante a um fio serial, ou seja, tudo que é recebido no pino de entrada DI é acumulado e então transmitido pelo pino DO. Duas condições são necessárias para que o módulo pare de receber dados e o transmita novamente: os dados são armazenados no buffer de DI até que nada seja recebido pela quantidade de tempo determinada pelo RO (Packetization Timeout) ou o buffer é preenchido por completo (100 bytes). Neste caso, é necessário um controle de fluxo de dados para que não haja perda de informação.

O outro modo de operação é conhecido como API (Application Programming Interface). Este modo é bem mais complexo que o transparente, e, desta forma, permite um maior nível de interação entre os módulos dentro da rede. Em outras palavras, quando operante em API, toda informação que entra e sai do módulo está contida em *frames*, que definem operações ou eventos dentro do módulo.

Uma vantagem imediata deste modo é o envio de informações diretamente para os módulos sem a necessidade de trocar o endereço através do modo de comando. Quando o coordenador de uma rede, por exemplo, necessita enviar para vários dispositivos em sequência, o modo API permite que esta operação seja realizada sem a necessidade de entrar no modo de comando, o que acelera muito o processo e, conseqüentemente, utiliza menos energia.

Cada frame possui um padrão determinado, que contém tamanho (*length*) da informação, endereço de envio, etc. Na figura 6, observa-se como é feita a divisão dos *bytes* em uma estrutura de *frame* API.



MSB = Most Significant Byte LSB = Less Significant Byte

Figura 6 - Estrutura de um frame API

Todo frame API é iniciado com o byte 0x7E. Quando operante em API, o módulo ignora qualquer informação que não comece com este byte.

Em seguida, os *bytes* 2 e 3 indicam o tamanho da estrutura específica API. É importante observar que o *byte* inicial, os bytes de tamanho e o *checksum* não são contados, restando (n-4) *bytes*.

Na estrutura específica API estão contidas todas as informações necessárias para a recepção da informação. Dependendo do comando, são estes bytes que indicam o endereço de 64 *bits* de envio, o endereço de 16 *bits* de envio, o comando, parâmetros do comando, entre outras funções que exercem controle sobre quem recebe estes dados.

Por fim, o byte de *checksum* tem como função validar o *frame* enviado. Este processo é feito a partir da soma de todos os valores dos bytes contidos nos dados do frame. Ao final, o valor do *checksum* será o resultado da subtração de 0xFF pelo último byte resultante da soma, caso esta seja maior que 0xFF. A seguir, um exemplo de como isto é apresentado.

O comando enviado neste frame é o ND (*Node Discover*). O *frame*, então, é construído da seguinte forma:

7E 00 04 08 52 4E 44 13, onde:

7E – Byte inicial que indica que um *frame* está sendo enviado.

00 04 – Bytes 2 e 3 que indicam o tamanho dos dados. Neste caso, são 4 bytes.

08 – Comando enviado (neste caso, 08 indica um comando AT).

52 – Identificador do *frame* (um número diferente de zero).

4E 44 – Correspondem aos códigos de N e D em hexadecimal.

13 – Byte de *checksum*. $CS = FF - (08 + 52 + 4E + 44) = FF - EC = 13$.

O byte 4 indica a função à qual aquele frame pertence. No caso citado, o valor 0x08 indicou um comando AT. Outros tipos de comandos API são apresentados na Tabela 2.

Tabela 2 - Tipos de frames API

Tipo API	Descrição
0x8A	Mensagens de status do módulo RF são enviadas a partir do módulo em resposta a condições específicas
0x08	Envia comandos AT. Estes podem ler ou configurar parâmetros do módulo

	receptor. Neste comando, os valores são imediatamente modificados.
0x09	Mesmo que o comando anterior, exceto que neste exemplo as modificações não são feitas imediatamente, sendo necessário um comando AC (<i>Apply Changes</i>) para que as alterações sejam validadas.
0x88	Caracteriza uma resposta a um comando AT.
0x17	Permite que valores de parâmetros dos registradores do módulo sejam consultados ou modificados.
0x97	Resposta a um comando remoto.

2.4.3 Configuração dos Módulos – Software X-CTU

O *software* X-CTU é disponibilizado pela Digi Internacional em www.digi.com, e serve como uma boa interface para configuração dos módulos XBee. Neste programa é possível visualizar todos os sinais de comando AT, com seus respectivos valores.

O programa possui em sua tela inicial uma verificação das portas COM ativas no momento, permitindo ao usuário escolher uma e conectá-la a partir dos parâmetros *Baud Rate*, *Parity*, *Flow Control*, *Stop Bit* e *Data Bits*. Ainda em *PC Settings* é possível modificar os valores de tempo passo a passo para entrar no modo de comando AT (tempo sem enviar ou receber informação, caractere digitado para entrar no modo – por padrão o “+”, etc.). Na Figura 7 é possível observar esta tela inicial.

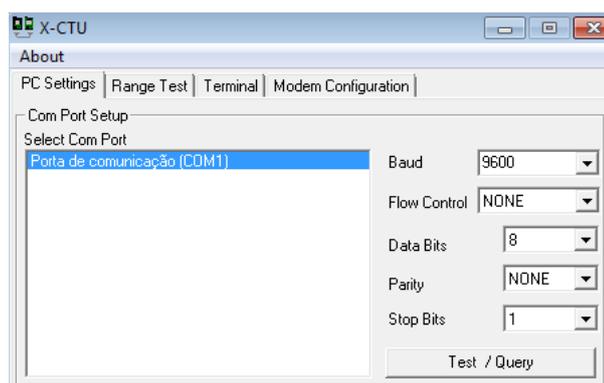


Figura 7 - Interface do software X-CTU – PC Settings

É possível observar ainda que o programa possui outras três abas, cujos nomes são *Range Test*, *Terminal* e *Modem Configuration*.

Em *Range Test*, o programa basicamente envia ou recebe dados para qualificar a potência do sinal, mostrando o nível do mesmo. Serve apenas para verificar se os módulos

XBee conseguem se comunicar e/ou verificar se estão a uma distância suficiente para que um visualize o outro. Na Figura 8 está representada um exemplo desta *interface* de testes.

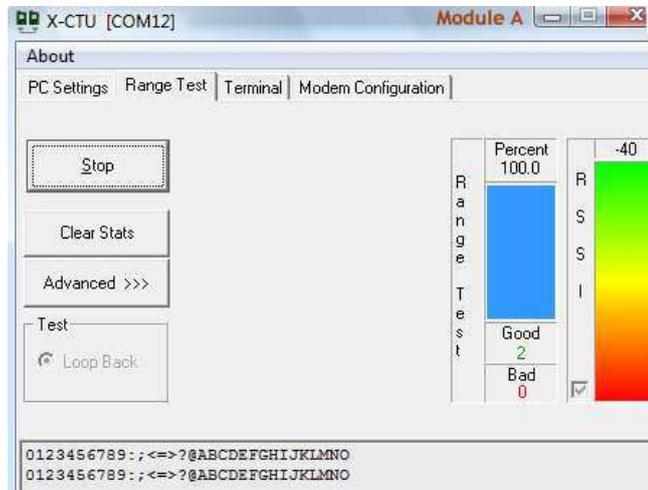


Figura 8 - Interface do software X-CTU – Range Test

Em *Terminal*, o programa disponibiliza uma ferramenta muito interessante para o usuário que quer enviar dados para outros módulos da rede. Nesta aba, existem dois modos de envio de informação possíveis: o primeiro, inicialmente, caracteriza a forma de operação transparente, onde cada caractere digitado é imediatamente enviado; no segundo, é permitido ao usuário o envio de pacotes de dados em ASCII ou HEX, válido para envio de frames em modo API. A ferramenta apenas disponibiliza o espaço para envio do frame, mas não o constrói. O pacote deve ser totalmente digitado pelo usuário.

Ainda, o terminal mostra as respostas dos módulos aos comandos recebidos. Desta forma, é possível abrir mais “janelas” do programa X-CTU e observar como é realizada a comunicação entre os módulos *frame a frame*.

Na Figura 9, estão representados exemplos de comunicações entre módulos para os dois métodos de comunicação: transparente e API.

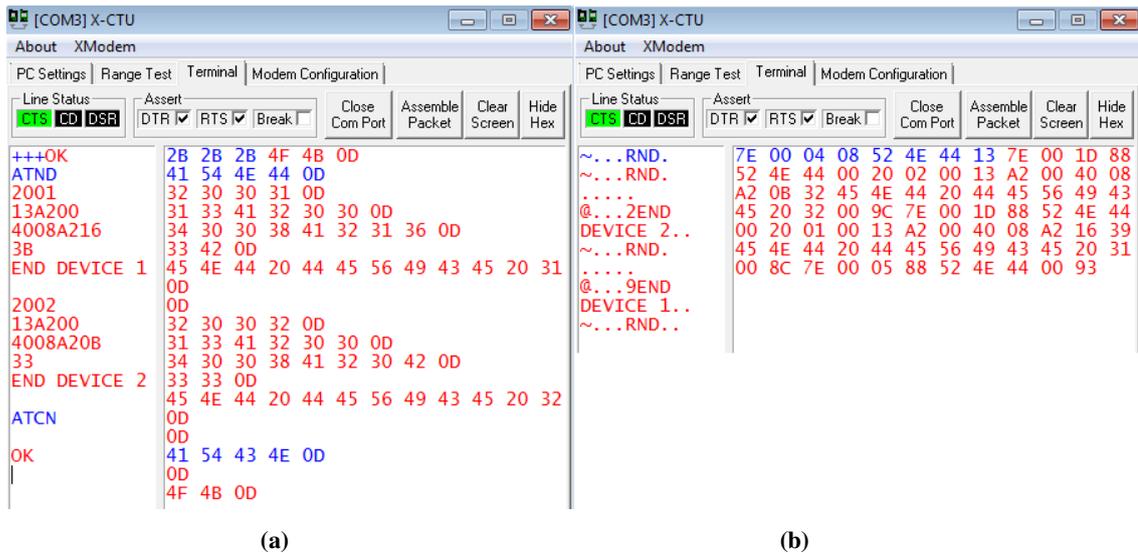


Figura 9 - Interface do software X-CTU – Terminal: (a) Transparente (b) API

Por fim, a aba *Modem Configuration*, a mais complexa, mostra todas as configurações dos módulos XBee, a partir de seus comandos AT, podendo ser alterados apenas escolhendo valores diferentes, sem a necessidade do conhecimento dos códigos dos mesmos. Ainda nesta aba, é possível alterar a versão de *firmware* dos módulos, o que é essencial quando se deseja trabalhar com essa tecnologia. Manter um *firmware* atualizado garante funcionalidade aos dispositivos, como também uma sincronia na comunicação dos mesmos. Na Figura 10 observa-se esta tela de configuração.

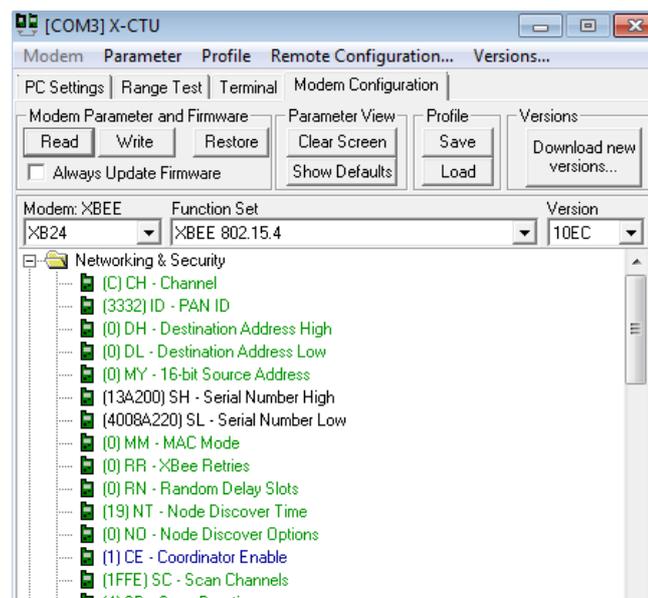


Figura 10 - Interface do software X-CTU – Modem Configuration

2.4.4 Placa CON-USBEE

A placa CON-USBEE (Figura 11) simula uma conexão serial via porta USB, e, através da instalação correta, esta é capaz de conectar-se a uma central (um computador, laptop) e realizar toda a comunicação de uma rede *ZigBee*, por exemplo, no controle de uma casa automatizada, controlando processos em fábricas, etc.

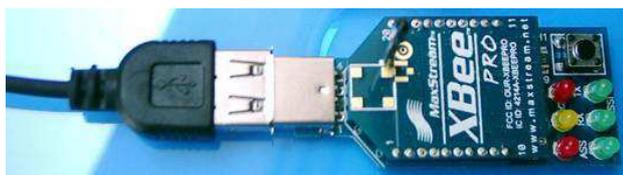


Figura 11 - Placa CON-USBEE

A placa disponibiliza *leds* indicadores, que acendem e apagam conforme eventos realizados pelo módulo XBee. Um exemplo é o *led* indicador de ASS (vermelho), que pisca uma vez por segundo ou duas vezes por segundo, indicando, respectivamente, se aquele módulo está configurado como coordenador ou dispositivo final.

Outros sinais que são verificados a partir de suas luzes são RX e TX (recepção e transmissão), como sinais de controle (quando o módulo está sendo atualizado, por exemplo).

2.5 Outras tecnologias de transmissão sem fio

Uma rede sem fio (WLAN – *Wireless Local Area Network*) pode ser considerada como uma expansão de uma rede local (LAN – *Local Area Network*). Um adaptador converte os pacotes, antes transmitidos pelos cabos, em ondas de rádio, por exemplo, que são transmitidas pelo ar.

Atualmente existe vários tipo de tecnologia que utilizam deste método para envio da informação, que se distinguem em diversos fatores, como taxa de transmissão, frequência de operação, custo, potência, etc. Quando se utiliza uma tecnologia sem fio, é interessante compará-la às demais a fim de entender as vantagens e desvantagens de cada estrutura isolada, e, assim, perceber a que melhor serve em cada aplicação.

É importante lembrar que, a principal vantagem do uso dessa tecnologia está no fato de que é alto o custo para utilização de cabos, em uma fábrica, por exemplo, sendo necessário

tirá-los do processo de monitoramento. Por outro lado, a transmissão sem fio possui uma possibilidade de erros considerável, tornando-a mais complexa de se implementar. A rede sem fio foi criada como um complemento da rede com fio, para que esta pudesse atuar em diversos tipos de ambientes e situações.

2.5.1 Bluetooth

Uma imensa quantidade de dispositivos possui a tecnologia Bluetooth adaptada como parte de sua estrutura de transmissão de dados. Desde aparelhos celulares, notebooks, periféricos (mouses, teclados, etc.) até consoles como *Wii* e *PlayStation 3*, em seus controles sem fio [7].

Criada em 1994 por engenheiros da Ericsson, a tecnologia *Bluetooth* foi inicialmente projetada para retirar a necessidade do uso de cabos e fios em dispositivos como telefones, teclados, etc. Entretanto, seu uso é muito bem empregado em outras áreas, como conectando aparelhos (tocadores de música, TV's, etc.) [4].

Quando comparada à tecnologia *ZigBee*, o *Bluetooth* apresenta vantagens e desvantagens em diferentes aspectos, tornando suas aplicações distintas. Na Tabela 3 é apresentado um quadro com as principais diferenças entre estas tecnologias.

Tabela 3 - ZigBee vs. Bluetooth

	<i>ZigBee</i>	<i>Bluetooth</i>
Padrão IEEE	IEEE 802.15.4	IEEE 802.15.1
Corrente de transmissão	30mA	40mA
Corrente em <i>Standby</i>	3 μ A	200 μ A
Taxa de transferência (/s)	250kbps	750kbps
Tempo de acesso à rede	30ms	3s
Tempo de transição do modo <i>sleep</i> para ativo	15ms	3s
Tempo de acesso ao canal	15ms	2ms

Desta forma, pode-se dizer que o Bluetooth é mais apropriado para aplicações como:

- Áudio e vídeo sem fio;
- Transferência de arquivos entre PC's e impressoras, por exemplo;

- Sincronização de dispositivos celulares, PC's, controles, etc.

É importante lembrar que os dispositivos que utilizam esta tecnologia são recarregados periodicamente, diferentemente de uma aplicação *ZigBee*, onde a alimentação geralmente é feita por baterias ou pilhas.

2.5.2 Wi-Fi

A *Wi-Fi Alliance* criou o padrão *Wi-Fi* para comunicação em redes sem fio. Todo o protocolo é baseado no padrão IEEE 802.11, que permite a utilização dessas redes com fácil instalação e baixo custo, pois estas operam em faixas de frequências que não necessitam de licença para instalação ou operação (apenas para uso comercial é necessária uma licença da ANATEL).[8]

Uma rede sem fio *Wi-Fi* utiliza de ondas de rádio para que seu sinal seja transmitido. O adaptador sem fio traduz as informações e as envia para um roteador, que decodifica esta informação e a envia através de uma conexão física Ethernet (com fios) para a internet. Para a recepção, o processo exatamente inverso acontece. Na Tabela 4, é apresentada uma comparação entre as tecnologias *Wi-Fi* e *ZigBee*.

Tabela 4 - ZigBee vs. Wi-Fi

	<i>ZigBee</i>	<i>Wi-Fi</i>
Padrão IEEE	IEEE 802.15.4	IEEE 802.11
Distância entre nós	10-100m	50-100m
Topologias	Estrela, <i>peer to peer</i> , árvore, etc.	Ponto ao hub
Taxa de transferência(/s)	250kbps	54Mbits
Tempo de acesso à rede	30ms	3-5s
Consumo	Muito baixo	Alto

Desta forma, pode-se dizer que a tecnologia *Wi-Fi* é mais bem aplicada em conectividade a redes LAN, na utilização de internet banda larga, pois o fator crucial que a descarta em uma rede de sensores é seu alto consumo.

2.6 Sensor de Temperatura – NTC

Os sensores de temperatura NTC (*Negative Temperature Coefficient*) fazem parte do grupo de semicondutores sensíveis à temperatura, conhecidos como termistores.

Os componentes desse grupo são excelentes sensores para aplicações onde é necessária uma alta sensibilidade à variação de temperatura. Podem ser utilizados desde a medição de temperatura em ambientes à área de biologia ou mesmo em automóveis.

O princípio de funcionamento de um NTC está basicamente na sua relação de resistência e temperatura. À medida que a temperatura aumenta, sua resistência diminui. Desta forma, é possível observar um perfil (curva característica) deste sensor, que, como desvantagem, possui o fato de que esta curva não é linear. Daí tem-se a necessidade de, sempre que utilizá-lo, traçar a característica do mesmo, procedimento descrito adiante.

Existe ainda outro tipo de termistor, com o funcionamento inverso ao NTC: o PTC (*Positive Temperature Coefficient*). Sua resistência aumenta com o aumento da temperatura.

A curva característica pode ser determinada pela Equação (1), calculando-se os valores das resistências para uma temperatura T (R_T), desde que se conheça o fator constante (B) determinado na fabricação e o valor de resistência R_0 a temperatura T_0 especificada.

$$R_T = R_0 \cdot e^{\left(\frac{B}{T} - \frac{B}{T_0}\right)} \quad (1)$$

Na Figura 12 é ilustrado um exemplo deste sensor de temperatura.



Figura 12 - Termistores NTC

Outra vantagem deste sensor (quando se trata do uso de módulos XBee) está no fato de que sua alimentação é de 3,3V. Isso significa que, quando utilizado em conjunto com módulos XBee, não há a necessidade do acoplamento de uma fonte de energia apenas para o sensor, uma vez que os módulos possuem, também, 3,3V em V_{CC} .

A ADD-THERM produz e desenvolve sensores NTC para aplicações na faixa de -55°C a 300°C.

3. DESENVOLVIMENTO DO PROJETO

A escolha da linguagem de programação para desenvolvimento deste projeto foi baseada na possibilidade de utilização da porta serial e na disponibilidade de blocos que pudessem suprir as necessidades da mesma. O LabVIEW possui o conjunto de blocos VISA, que permitem toda manipulação de dados através da porta serial de forma prática.

3.1 Montagem das placas

As placas utilizadas como nós independentes da rede *ZigBee* consistem de uma fonte de alimentação do circuito, feitas por pilhas AA, dois *leds* indicadores de status (informando se o módulo está transmitindo e se o mesmo é coordenador/dispositivo final) e um módulo XBee, que faz a leitura em suas entradas e transmite quando há requisição do coordenador. Na Figura 13, apresenta-se um exemplo da placa utilizada.

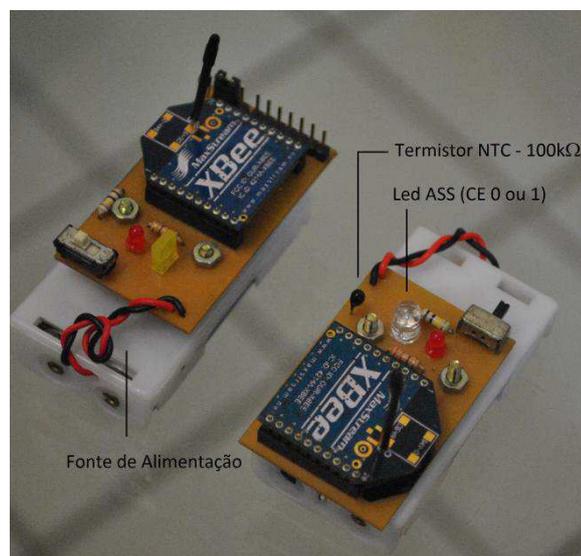


Figura 13 - Placas Sensoriais

3.2 Configuração dos módulos

Basicamente, as configurações necessárias para que o programa realize comunicação com os módulos XBee estão a seguir:

- Alterar o valor do endereço de 16 *Bit*;

- Escolher um nome para o dispositivo (por exemplo, *End Device 1*);
- Configurá-los para trabalhar em modo API. Nos módulos XBee da série 2, a configuração para modo API não é necessária, pois estes dispositivos podem trabalhar em modo transparente e API de forma conjunta.
- Configuração dos pinos de entrada ADC. Como visto anteriormente, estes pinos podem ser utilizados como entradas analógicas e entradas/saídas digitais. Para que os dados dos sensores sejam recebidos de forma correta, deve-se configurá-lo com o valor 1.

3.3 Software de monitoramento - LabVIEW

O LabVIEW é uma linguagem de programação gráfica, criada em 1986 pela National Instruments. Entre as suas principais aplicações, estão a realização de medições e automação, e, sua programação feita de acordo com o fluxo de dados, oferece vantagens em relação à aquisição de dados.

Os programas criados em LabVIEW são chamados de *Virtual Instruments* (VI's), ou instrumentos virtuais. Em seus conteúdos, ele possui um *front painel*, onde está localizada a interface, e um diagrama de blocos, onde está localizado o código do programa. Sua linguagem de programação é chamada de "G".

Para entender melhor a lógica do programa, é importante discutir as funções de determinados blocos do LabVIEW.

A cada iteração, a porta COM é configurada em todos os seus parâmetros. Para tanto, é necessária a utilização do bloco *VISA Configure Serial Port*, apresentado na Figura 14. A porta recebe os valores de paridade, *stop bit*, *baud rate*, etc.

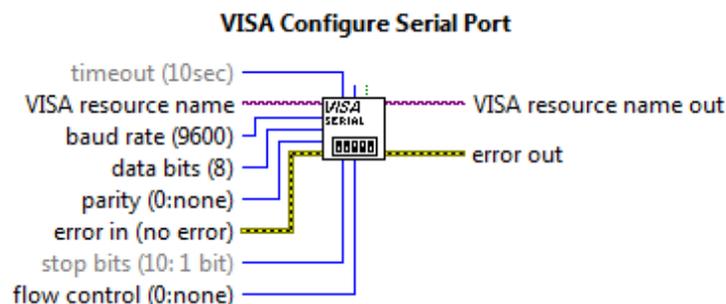


Figura 14 – Bloco utilizado para configurar parâmetros da porta *serial*

Uma vez configurada a porta *serial*, a mesma entra em modo de “escuta”. Um bloco é responsável por testá-la a partir de seus *bits* que estão no *buffer* e, se diferente de zero, ou seja, existe informação no *buffer*, estes são lidos. O bloco responsável pela leitura é chamado de *VISA Read*, visualizado na Figura 15. Vale lembrar que a informação recebida é agrupada e repassada ao programa para tratamento.

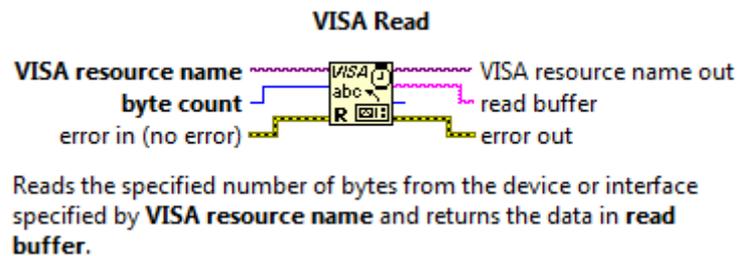


Figura 15 - Bloco utilizado para receber dados pela porta *serial*

Recebido o conjunto de bytes em hexadecimal, o programa identifica as funções pelo byte que indica início de *frame*, 0x7E. Dependendo do tipo de função, o programa tem uma resposta diferente.

O programa ainda utiliza as funções de escrita do bloco *VISA Write* (Figura 16). Para o administrador que utiliza a interface e deseja realizar as medições, as requisições são feitas “escrevendo” na porta serial os comandos para verificar os nós existentes da rede (ND – *Node Discover*) e fazer leituras dos pinos de entradas analógicas (IS – *Force Sample*).

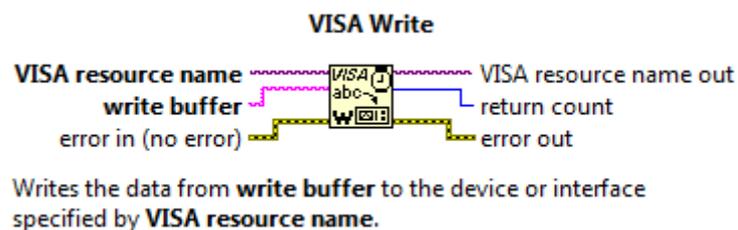


Figura 16 - Bloco utilizado para enviar dados pela porta *serial*

A *interface* serve para que um usuário faça leituras dos sensores a qualquer momento. Esta conta com basicamente três partes: configuração da porta serial, requisição de leitura e status da rede e, por último, leitura dos dados. Estas funções podem ser observadas na Figura 17.

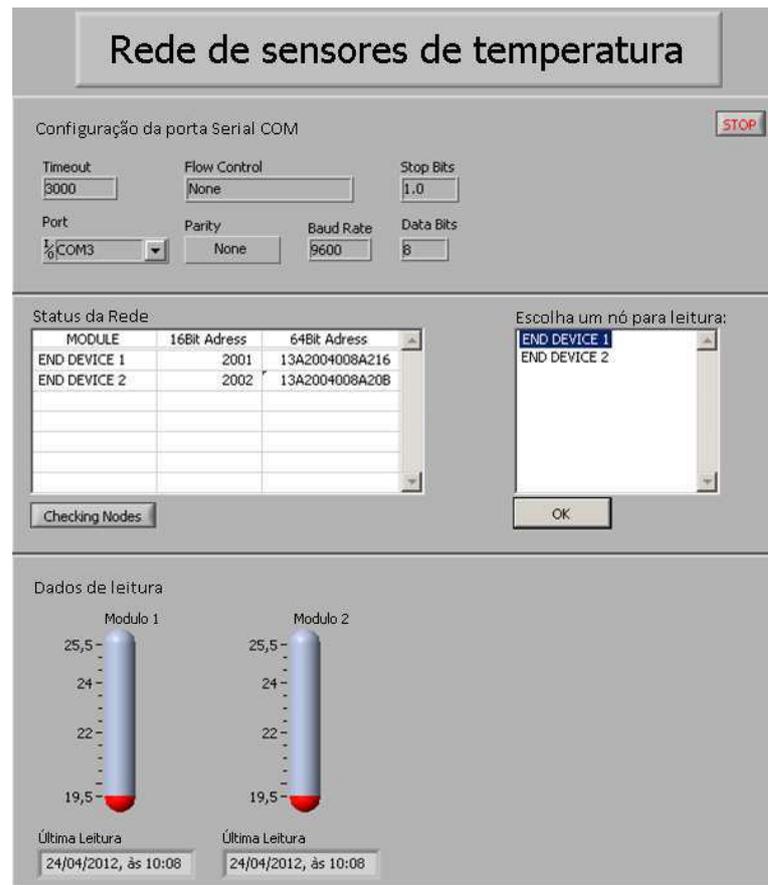


Figura 17 - Interface do programa

3.4 Cálculo da característica do sensor NTC

Para a utilização correta do sensor de temperatura, foi necessário traçar a curva característica do mesmo.

A ideia para o processo é captar o valor de tensão que o módulo XBee recebe do sensor NTC e compará-lo a um valor conhecido, neste caso, foi usado um outro sensor de temperatura LM 35, cuja faixa de valores é conhecida (variando 10mV para cada 1°C). O circuito utilizado para a recepção dos dados foi baseado em um divisor de tensão: o NTC foi colocado em série com um resistor de valor conhecido (100kΩ) e, os valores da tensão no sensor, transmitidos pelos pinos D1 e D0 de um módulo XBee, gravados em disco para futuro tratamento. Um esquema do circuito montado é visualizado na Figura 18.

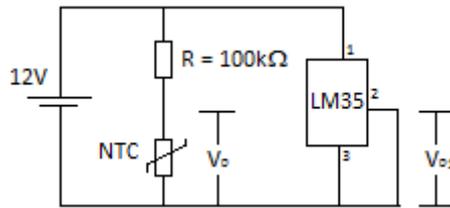


Figura 18 – Diagrama do circuito utilizado para traçar a característica do NTC

Para controlar a temperatura detectada pelos sensores, foi utilizado um circuito com uma placa de Peltier. Inicialmente, a mesma foi resfriada a sua menor temperatura (em torno de 19° C), e então os valores começaram a ser anotados. A cada intervalo de cinco segundos, um novo valor era medido e anotado, até que a temperatura da placa atingiu valores acima de 70° C e todo o circuito foi desligado.

Os valores anotados, então, foram tratados utilizando o programa *Curve Expert*. A interpolação aplicada e que obteve o melhor resultado foi a polinomial, com grau sete. Na Figura 19 está representado o gráfico gerado pelos pontos obtidos (Tensão (V) x Temperatura (°C)):

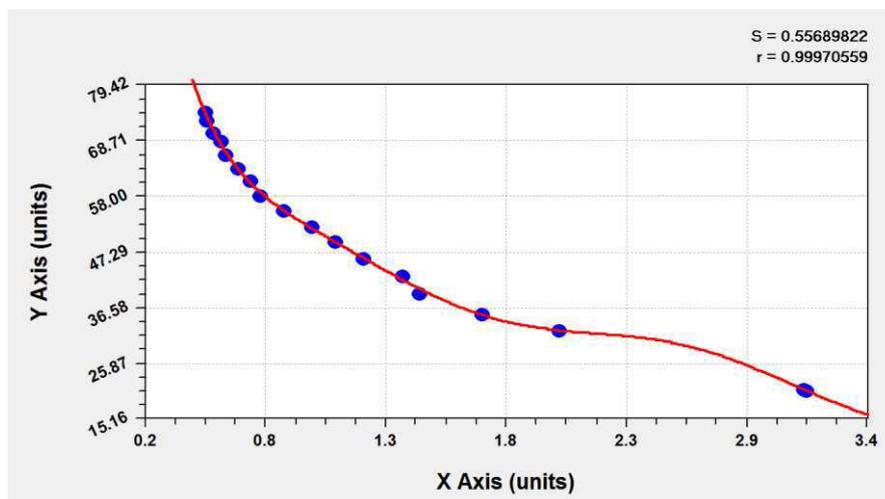


Figura 19 – Gráfico da Tensão (V) x Temperatura (°C)

O polinômio da característica encontrada é dado pela Equação (2).

$$T = 234.279 - 715.305 x + 1239.429x^2 - 1179.406x^3 + 637.019x^4 - 194.290x^5 + 31.118x^6 - 2.0334x^7 \quad (2)$$

4. ANÁLISE E TESTES DE CAMPO

Para análise do conjunto, foram medidos valores de temperatura do lado interno e externo à sala 08 do Laboratório de Instrumentação Eletrônica e Controle (LIEC), na UFCG, ao longo de cinco horas e doze minutos aproximadamente, entre 12h18min e 17h30min.

A parte interna consiste de um ambiente controlado, com isolamento físico do meio externo através de portas e janelas, fechadas todo o tempo.

O ambiente externo consiste num corredor não coberto e com alta incidência de raios solares sobre todas as superfícies. O clima apresentou um dia ensolarado, ao longo de todo o teste.

Os valores anotados são apresentados no gráfico da Figura 20.

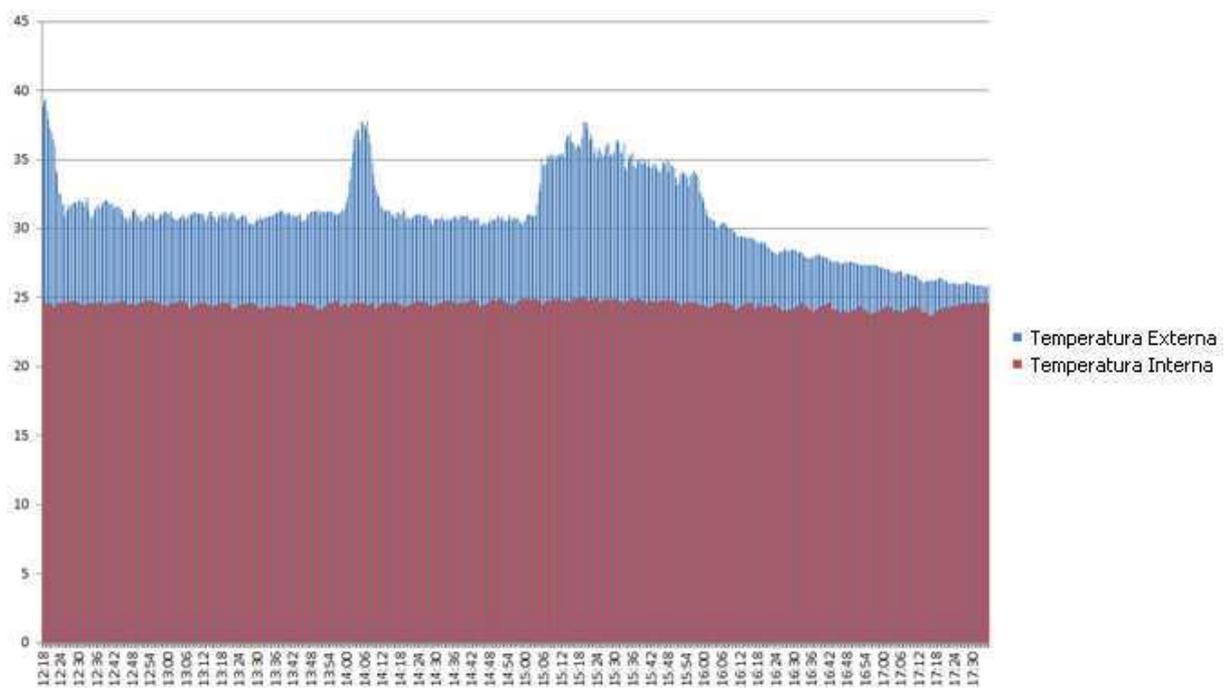


Figura 20 - Gráfico das temperaturas internas e externas à sala 08 do LIEC

É possível observar que os horários entre 12h24min e 14h00min, como também entre 14h08min e 15h12min há uma queda de temperatura externa. Isso se dá ao fato de uma sombra gerada sobre os sensores, causadas pela proteção dos suportes dos condicionadores de ar. O pico de temperatura, ocorrido entre 14h00min e 14h08min acontece devido a uma brecha que permite a passagem dos raios solares diretamente na superfície onde estava

localizado o sensor. Na Figura 21 é possível observar esse fenômeno.



Figura 21 - Geradores de sombra sobre superfície do sensor

Desta forma, é possível afirmar que a resposta do sensor é rápida, uma vez que o mesmo apresentou bem a variação de temperatura de mais de 7°C , num espaço de aproximadamente 8 minutos.

Para o horário após as 15h12min, o sensor volta a captar as temperaturas numa região sem sombra. Esta varia negativamente até quase se igualar à temperatura do ambiente interno à sala.

5. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Este projeto foi desenvolvido como parte de um sistema de monitoramento de temperatura de um ambiente qualquer. A associação deste projeto com microcontroladores pode ser aplicada na automação de uma casa ou ambientes de uma empresa, o que pode resultar em um melhor uso da energia elétrica, evitando desperdícios.

A rede configurada para este trabalho possui topologia do tipo estrela, observada pelo fato de o coordenador se comunicar diretamente com os outros dois nós constituintes. As informações, além de tratadas pelo programa gerado no LabVIEW, são armazenadas em documentos com o formato de data/temperatura, para futura análise dos resultados.

Como trabalho futuro, pode-se ampliar a rede de sensores para diversos tipos, como os sensores de movimento, por exemplo, criando um sistema que pudesse atuar na iluminação, regulando a temperatura e, por que não, na segurança de ambientes contra incêndio e livres de pessoas não autorizadas.

6. BIBLIOGRAFIA

- [1] **ZigBee Alliance** - <http://www.zigbee.org> – acessado em de março de 2012.
- [2] Protocolo IEEE 802.15.4: **Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)**, acessado em março de 2012 – disponível em: <http://standards.ieee.org/getieee802/download/802.15.4d-2009.pdf>
- [3] FARAHANI, Shanin. **ZigBee Wireless Network and Transceivers**. Burlington: Newnes, 2008.
- [4] **Bluetooth Technology** – <http://www.bluetooth.com> – acessado em março de 2012.
- [5] **Manual do Produto XBee™ / XBee Pro™** – acessado em março de 2012 - disponível em: http://ftp1.digi.com/support/documentation/90000982_A.pdf
- [6] KINNEY, Patrick. ZigBee: “Wireless Control that Simple Works”, disponível em: <http://www.zigbee.org>
- [7] LABIOD, Houda; AFIFI, Hossan; DE SANTIS, Costantino. **Wi-fi, Bluetooth, ZigBee and WiMax**. Paris. Springer, 2007.
- [8] Site da Agência Nacional de Telecomunicações (ANATEL) – disponível em <http://www.anatel.gov.br>, - acessado em abril de 2012.

7. ANEXO

ANEXO 1 – Pinos do módulo XBee™/XBee-Pro™.

Pino	Nome	Sentido	Descrição
1	VCC	-	Fonte de Energia 3.3V
2	DOUT	Saída	Saída de dados UART
3	DIN / CONFIG	Entrada	Entrada de dados UART
4	DO8	Saída	Saída digital 8
5	RESET	Entrada	Inicializa o módulo (um pulso nível 0 de pelo menos 200ms)
6	PWM0 / RSSI	Saída	Saída PWM 0 / Indicador de força do sinal de RF (RX)
7	PWM1	Saída	Saída PWM 1
8	[reservado]	-	Sem função definida
9	$\overline{\text{DTR}}$ / SLEEP_RQ / DI8	Entrada	Linha de controle da função Sleep ou entrada digital 8
10	GND	-	Terra
11	AD4 / DIO4	Ambos	Só entrada analógica 4 ou entrada/saída digital 4
12	$\overline{\text{CTS}}$ / DIO7	Ambos	Controle de fluxo CTS ou entrada/saída digital 7
13	ON / $\overline{\text{SLEEP}}$	Saída	Indicador de estado do módulo
14	VREF	Entrada	Tensão de referência para as entradas A/D
15	Associate / AD5 / DIO5	Ambos	Indicador de associação, só entrada analógica 5 ou entrada/saída digital 5
16	RTS / AD6 / DIO6	Ambos	Controle de fluxo RTS, só entrada analógica 6 ou entrada/saída digital 6
17	AD3 / DIO3	Ambos	Só entrada analógica 3 ou entrada/saída digital 3
18	AD2 / DIO2	Ambos	Só entrada analógica 2 ou entrada/saída digital 2
19	AD1 / DIO1	Ambos	Só entrada analógica 1 ou entrada/saída digital 1
20	AD0 / DIO0	Ambos	Só entrada analógica 0 ou entrada/saída digital 0