

**Universidade de Brasília**  
**Departamento de Eng. Elétrica**  
**LAVSI - UNB**

Luis Felipe C. Figueredo

# Tutorial XBee

# Tutorial XBee

## 1. Introdução

Introdução ao Padrão XBee

XBee – Especificações

XBee - Extras

# Tutorial XBee

## 1. Introdução

Introdução ao Padrão XBee

# 1.1 Introdução ao Padrão XBee

---

- O padrão ZigBee IEEE 802.15.4 foi desenvolvido pela ZigBee Alliance junto ao IEEE (Institute of Electrical and Electronics Engineers), através da associação de várias empresas.
- **Objetivos:** Desenvolver um padrão de baixo consumo de energia, baixo custo, robusto, seguro, confiável, e com sua norma aberta.
- **Operação:** Na frequência ISM (Industrial, Scientific, and Medical) de 2,4GHz (exceção dos EUA e da Europa) sem necessidade de licença de uso.
- **Aplicações:** Basicamente, controle de dispositivos, como relês, trancas eletromagnéticas, ventilação, aquecimento, motores, eletrodomésticos, e aquisição de dados de sensores, como de temperatura, umidade, luminosidade, pressão, velocidade do vento, etc.

# 1.1 Introdução ao Padrão XBee

---

## ➤ Módulos XBee:

Entre as várias empresas membros na ZigBee Alliance que disponibilizam produtos baseados na pilha de protocolo ZigBee, trabalharemos com o módulo XBee da empresa MaxStream®

O módulo XBee é de fácil implementação, pequeno e de baixo consumo. Os módulos, ainda podem ser interfaceados com outros dispositivos, como computadores ou microcontroladores, através de uma comunicação serial assíncrona.

\* Como o ZigBee é um padrão mundial, normalizado pelo IEEE, qualquer dispositivo ZigBee de outro fabricante poderá se comunicar sem problemas com o XBee, ou com qualquer outro dispositivo de outro fabricante.

# Tutorial XBee

## 1. Introdução

XBee - Especificações

# 1.2 XBee - Especificações

## ➤ Performance:

<b>Especificações</b>	<b>XBee</b>	<b>XBee-Pro</b>
<b>Alcance Indoor/Urbano</b>	Até 30m	Até 100m
<b>Alcance Outdoor em linha visível</b>	Até 100m	Até 1500m
<b>Potência máxima de transmissão</b>	1mW (0 dBm)	60mW (18 dBm)
<b>Taxa de dados interface serial</b>	115,2 Kbps	115,2 Kbps
<b>Taxa de dados de RF</b>	250 Kbps	250 Kbps
<b>Sensibilidade do Receptor</b>	-92 dBm	-100 dBm

## ➤ Requisitos Elétricos:

<b>Especificações</b>	<b>XBee</b>	<b>XBee-Pro</b>
<b>Tensão de Alimentação</b>	2.8 - 3.4 V	2.8 - 3.4 V
<b>Corrente de Transmissão</b>	45mA (@ 3.3 V)	139 - 227 mA (@ 3 V)
<b>Corrente de Recepção</b>	50mA (@ 3.3 V)	55mA (@ 3.3 V)
<b>Corrente em modo Sleep</b>	< 10 $\mu$ A	< 10 $\mu$ A

# 1.2 XBee - Especificações

## ➤ Características Gerais:

Especificações	XBee	XBee-Pro
Dimensões	2.438 cm X 2.761 cm	2.438 cm X 3.294 cm
Peso	3 g	3 g
Temperatura de Operação	-40º C até 85º C	-40º C até 85º C

## ➤ Rede e Segurança:

Especificações	XBee	XBee-Pro
Nº de Canais Seleccionáveis	16 em seqüência	12 em seqüência
Faixa de Operação	2,405 - 2,480 GHz	2,405 - 2,465 GHz
Opções de Filtro	Canal Físico, PAN ID, Endereço	
Endereçamento	65000 endereços disponíveis por canal	
Topologias de Rede	Ponto-a-Ponto, Ponto-a-Multiponto, Par-a-Par, Malha	
Manipulação de Erro	Retransmite (Retries) e Reconhecimento (ACKs)	
Espalhamento Espectral	DSSS (Direct Sequence Spread Spectrum)	
Criptografia	128-bit AES	

# Tutorial XBee

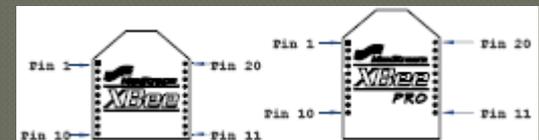
## 1. Introdução

XBee - Extras

# 1.3 XBee - Extras

## ➤ Pinagem:

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DCOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DO8*	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM0 / RSSI	Output	PWM Output 0 / RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	AD4 / DIO4	Either	Analog Input 4 or Digital I/O 4
12	CTS / DIO7	Either	Clear-to-Send Flow Control or Digital I/O 7
13	ON / SLEEP	Output	Module Status Indicator
14	VREF	Input	Voltage Reference for A/D Inputs
15	Associate / AD5 / DIO5	Either	Associated Indicator, Analog Input 5 or Digital I/O 5
16	RTS / AD6 / DIO6	Either	Request-to-Send Flow Control, Analog Input 6 or Digital I/O 6
17	AD3 / DIO3	Either	Analog Input 3 or Digital I/O 3
18	AD2 / DIO2	Either	Analog Input 2 or Digital I/O 2
19	AD1 / DIO1	Either	Analog Input 1 or Digital I/O 1
20	AD0 / DIO0	Either	Analog Input 0 or Digital I/O 0



- \*Conexões Mínimas:  
Vcc, Gnd, Dout & Din
- \*Firmware Update:  
Mínimo + RTS & DTR
- \*Os pinos não usados  
devem ser deixados  
desconectados

# 1.3 XBee - Extras

## ➤ Antenas:



\* Entre os três tipos de antenas para os módulos XBee, as antenas **Whip** (~2,5cm), e a **Connector** (para antena externa) se destacam por poderem direcionar o feixe de sinal, melhorando a performance da rede, enquanto que a **Chip** destaca-se por ser muito mais compacta que as outras antenas.

# 1.3 XBee - Extras

---

## ➤ **Software de Desenvolvimento:**

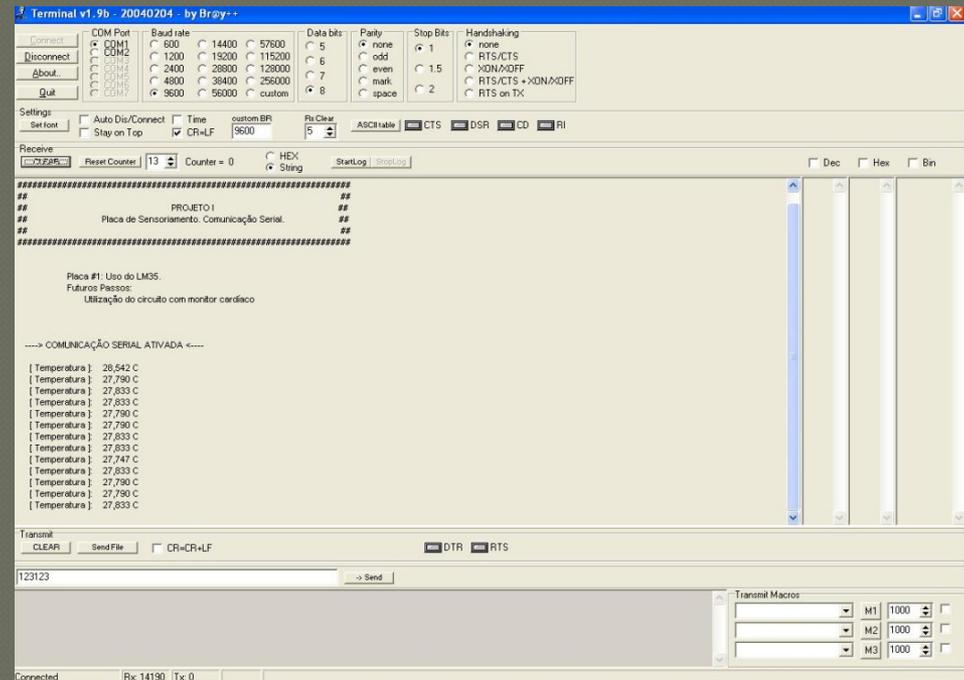
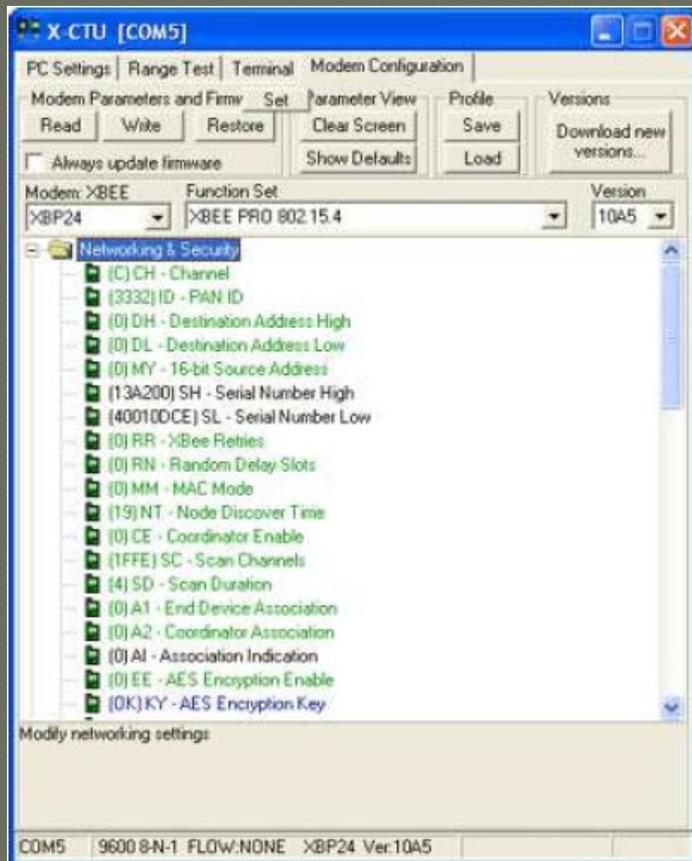
Os softwares de desenvolvimento utilizados, até o momento, são: o Terminal e o X-CTU.

O software: Terminal possui uma interface amigável e muito prática para comunicação serial.

O X-CTU, disponibilizado pela MaxStream®/Digi, possui um terminal de comunicação, uma aba de leitura de sinal, e uma aba de configuração do módulo, dentro da qual podemos atualizar o firmware do módulo e, também, configurar o módulo através de uma interface visual (sem a utilização de comandos AT).

# 1.3 XBee - Extras

## ➤ Software de Desenvolvimento:



# Tutorial XBee

## 2. Modos de Operação

Modos de Operação  
Modo de Comando

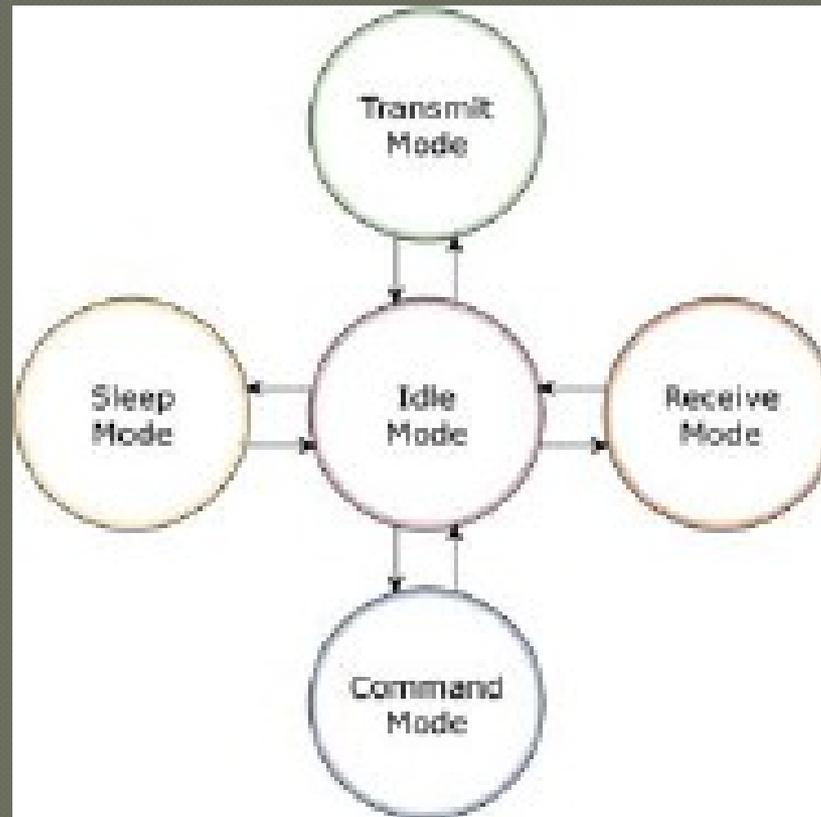
# Tutorial XBee

## 2. Modos de Operação

Modos de Operação

# 2.1 Modos de Operação

- Os módulos XBee sempre estarão trabalhando em um dos seguintes modos de operação:



# Tutorial XBee

## 2. Modos de Operação

Modo de Comando

## 2.2 Modo de Comando

---

- O modo de comando é o espaço para se ler e/ou modificar os parâmetros e características dos módulos.
- Qualquer caractere será interpretado como um comando.
- Existem duas formas de se atuar no modo de comando:
  - AT Command Mode (“Transparent”)
  - API Command Mode\*

**\* Mais informações: Vide Datasheet do XBee.**

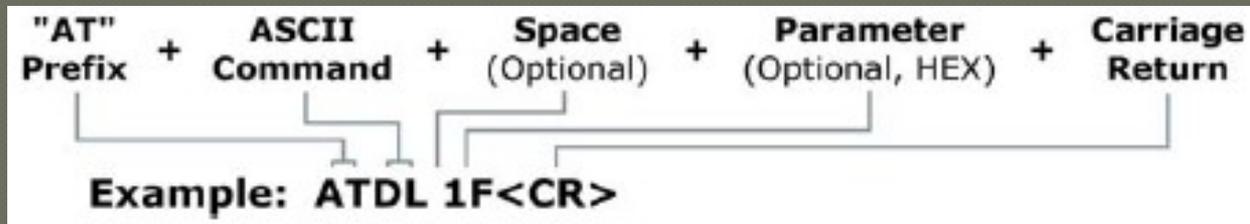
## 2.2 Modo de Comando

### ➤ Entrando no modo de Comando AT:

- Espere um tempo (definido por Guard Time **GT** [default: 1s.]);
- Envie a seqüência de comando (definida por **CC** [default: **+++**]);
- Espere um tempo (definido por Guard Time **GT** [default: 1s.]);

### ➤ Enviando Comando AT:

- Utilize a seguinte sintaxe:



- O exemplo verifica-se o uso do comando DL (endereço destino) definindo o endereço destino com o parâmetro 1F (31 em decimal).

\* **Hexadecimal** é utilizado nos parâmetros, apesar de 0x não ser necessário.

\* **0's a esquerda** dos parâmetros podem ser omitidos. EX: 0x001F = 1F

## 2.2 Modo de Comando

---

- Gravando na memória não volátil:
  - Após as configurações, utilize o comando **WR**: (**ATWR**).
- Saindo do modo de Comando AT:
  - Utilize o comando **CN**: (**ATCN**).
  - Ou, espere um período de inatividade, definido por **CT** (Command Timeout).

- Exemplo:
  - **+++**
  - **ATDL5**
  - **ATMY0**
  - **ATWR**
  - **ATCN**

# Tutorial XBee

## 3. Comunicação Serial

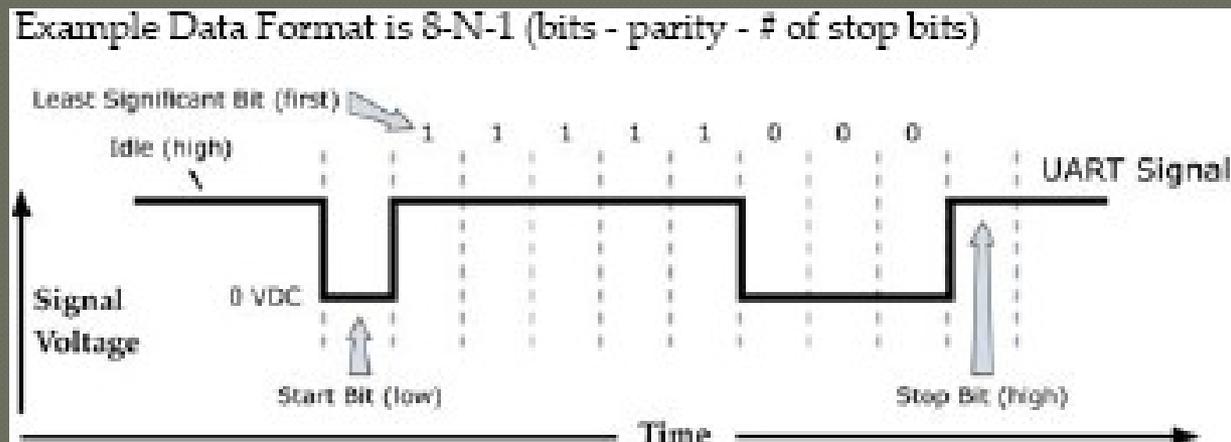
# 3. Comunicação Serial

---

- É utilizada para um controlador ter acesso ao módulo XBee.
- A comunicação é feita pela interface UART através dos pinos de entrada **DI** (pino 3) e de saída **DO** (pino 2), como um sinal serial assíncrono.
- Para haver a comunicação serial, deve-se observar se os dois módulos UARTs estão com as mesmas configurações de:
  - Baud Rate;
  - Paridade;
  - Start Bits;
  - Stop Bits;
  - Data Bits.

# 3. Comunicação Serial

- Quando não há dados sendo transmitidos, o sinal estará em nível alto, a espera.
- Cada pacote de dados é constituído por:
  - Start bit (nível baixo);
  - 8 bits de dados (Menos Signif. no começo);
  - Stop bit (nível alto).
- Exemplo (0x1F, decimal “31”):



# 3. Comunicação Serial

---

## ➤ Modo de Transmissão: Transparente

- Modo de transmissão default;
- Neste modo de transmissão, todos os dados UART recebidos pelo pino de entrada **DI** (pino 3) é levado para a transmissão por RF, e todo dado recebido via RF é mandado através do pino de saída **DO** (pino 2).

## ➤ Modo de Transmissão: API

- Modo de transmissão alternativo ao transparente, onde os pacotes são enviados em forma de frames.
- Maiores informações: Vide Datasheet sobre o Modo API.

# 3. Comunicação Serial

---

## ➤ “Packetization” Serial → RF:

- Os dados transmitidos via serial são armazenados em um buffer (DI Buffer) até que uma das seguintes causas, faça com que eles sejam transformados em pacotes e transmitidos:
    - 1) Nenhum caractere é recebido durante um período de tempo, determinado pelo parâmetro utilizado no comando **RO**;
    - 2) Estouro do buffer (100 caracteres); \*
    - 3) É detectado uma Seqüência de Comando ( **GT + CC +GT** ). Neste caso, todos caracteres antes da seqüência são automaticamente transmitidos.
- \* No caso do buffer ficar cheio, deve ser implementado algum controle de fluxo por hardware ou software para evitar o overflow.

# 3. Comunicação Serial

## ➤ Comando **AP**:

- Leitura e Escrita.
- Modo de Operação: Frame based API ou modo transparente.

### **Parâm. - Configuração**

0	Modo Transparente
1	API
2	API (c/ escaped chars)

**Default:** 0

## ➤ Comando **RO**:

- Leitura e Escrita.
- Número de “char times” de delay entre caracteres antes da transmissão.
- Setando RO = 0, os caracteres são transmitidos assim que recebidos.

**Parâmetros:** 0 - 0xFF [x char times]

**Default:** 3

# 3. Comunicação Serial

## ➤ Comando **BD**:

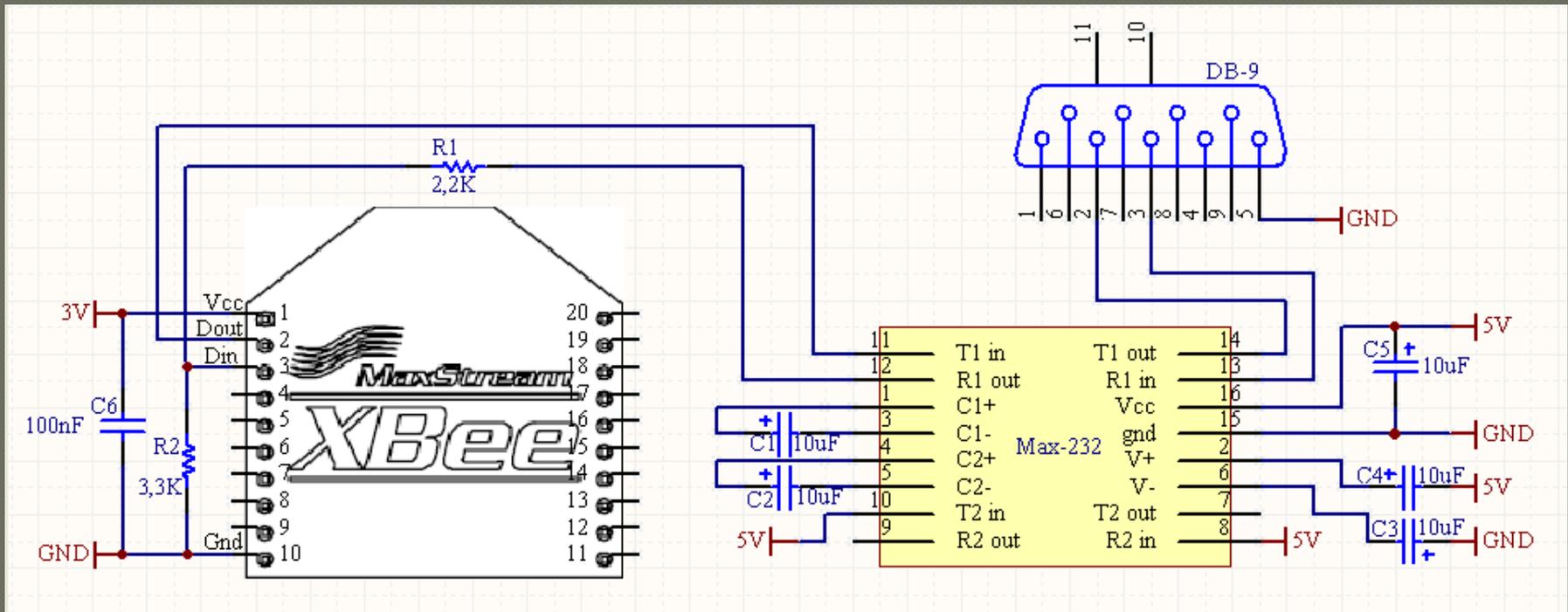
- Leitura e Escrita.
- [Interface Data Rate].
- Alteração só é feita após a saída do modo de comando (**ATCN**).

AT Command: ATBD	
Parameter Range: 0 - 7 (standard rates) 0x80-0x1C200 (non-standard rates)	
Parameter	Configuration (bps)
0	1200
1	2400
2	4800
3	9600
4	19200
5	38400
6	57600
7	115200

Default Parameter Value: 3

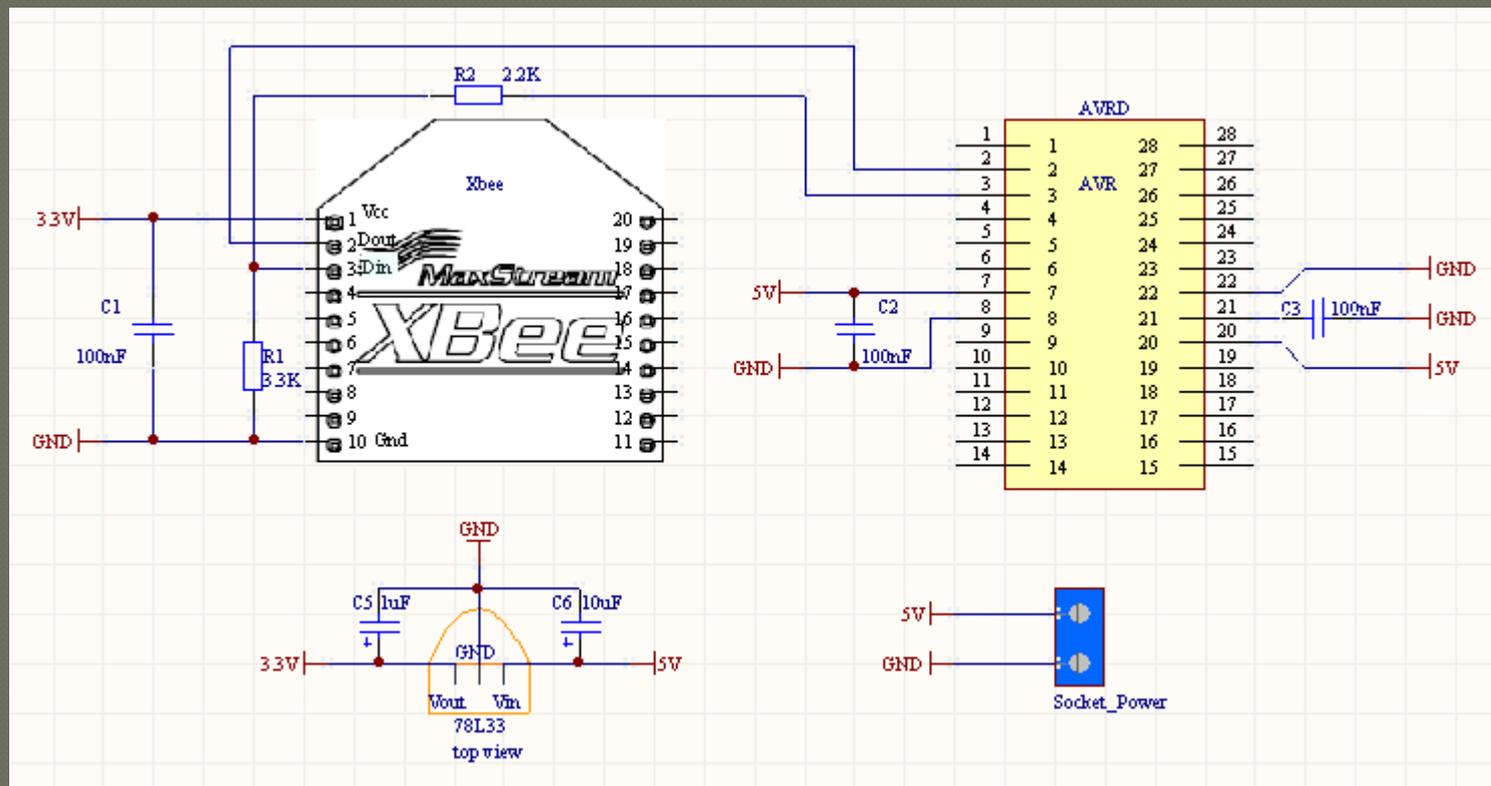
# 3. Comunicação Serial

## ➤ Esquemático: XBee - Entrada Serial PC.



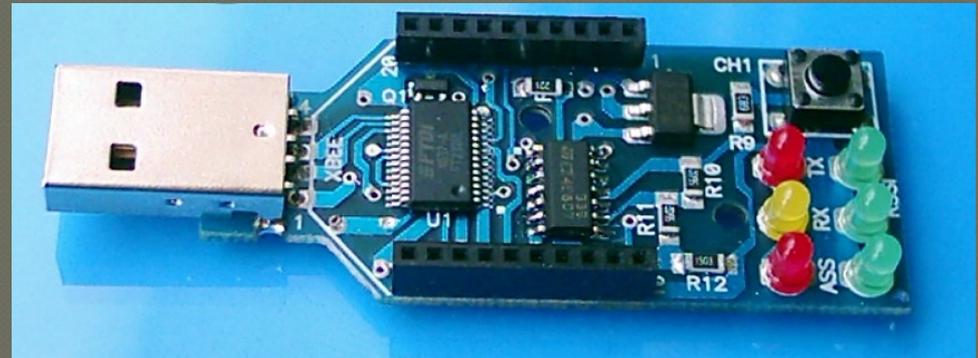
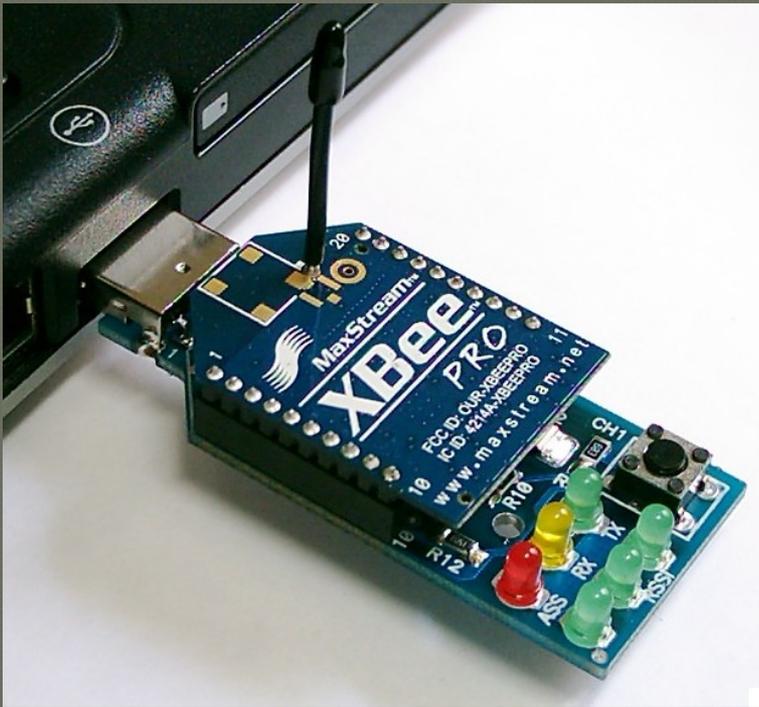
# 3. Comunicação Serial

## ➤ Esquemático: XBee - AVR.



# 3. Comunicação Serial

## ➤ COM-USBBEE



# Tutorial XBee

## 4. Conversor AD e I/O Digital

Configuração Básica

Obtendo Dados

Formato dos dados

PWM

I/O Line Passing

# Tutorial XBee

## 5. NetWorking

Comunicação Básica  
Comunicação Avançada  
Comandos Importantes  
Comandos Extras

# Tutorial XBee

## 5. NetWorking

### Comunicação Básica

# 5.1 Comunicação Básica

---

- Os módulos são identificados na rede por meio de três configurações:
  - Canal (**CH**)
  - Pan ID (**ID**)
  - Endereço  
(**SL, SH, DL, DH, MY**)
- \* Os módulos só se comunicaram se possuírem o mesmo Canal, o mesmo Pan ID, e endereços compatíveis.

# 5.1 Comunicação Básica

## ➤ Canal:

- A seleção de canais é uma das três formas de endereçamento do Xbee.
- Para que haja comunicação entre os módulos é necessário que ambos possuam trabalhem no mesmo canal.

## ➤ Comando **CH**:

- Leitura e Escrita.
- Freqüência de Centro:

$2.405 \text{ GHz} + (\mathbf{CH} - 11_{\text{decimal}}) * 5$

MHz

\* A rejeição de canais adjacentes é de 23dB

### Parâmetros:

0x0B - 0x1A (XBee)

0x0B - 0x17 (XBee-Pro)

### Default:

0x0C (12 decimal)

# 5.1 Comunicação Básica

## ➤ Comando **ID**:

- Seta ou lê o PAN (Personal Area Network) ID do módulo.
- Para que haja comunicação entre os módulos é necessário que ambos estejam trabalhando com o mesmo PAN ID.

**Parâmetros:** 0 - 0xFFFF

**Default:** 0x3332  
(13106<sub>decimal</sub>)

### **Pan Broadcast: ID 0xFFFF**

Módulo envia o pacote para **todas as PANs** no **único canal** de trabalho. Os módulos devem possuir **end. Compatíveis.**

# 5.1 Comunicação Básica

---

## ➤ Endereços Fonte:

- Para que um módulo com um end. Fonte X receba alguma mensagem esta deve ser enviada por um módulo com endereço destino igual a X.
  - Cada módulo Xbee contém:
    - Um endereço de fábrica FIXO de 64bits. (**SH,SL**)
    - Um endereço configurável de 16bits (**MY**)
- \* O endereço de 64bits deverá ser usado se MY for configurado como 0xFFFF ou 0xFFFE

# 5.1 Comunicação Básica

## ➤ Comando **SH** e **SL**:

- Leitura Apenas.
- Serial Number High (32 bits MSB) e Serial Number Low (32 bits LSB).

**Parâmetros:**

0 - 0xFFFFFFFF

## ➤ Comando **MY**:

- Leitura e Escrita.
- Endereço Fonte de 16 bits do módulo.
- Setando MY 0xFFFF desativa a recepção por meio do end. de 16bits.

**Parâmetros:** 0 -  
0xFFFF

**Default:** 0

# 5.1 Comunicação Básica

## ➤ Comando **DH**:

- Leitura e Escrita.
- Endereço Destino: 32bits mais significativos.

**Parâmetros:** 0 -  
0xFFFF

**Default:** 0

## ➤ Comando **DL**:

- Leitura e Escrita.
- Endereço Destino: 32bits menos significativos.

**Parâmetros:** 0 -  
0xFFFF

**Default:** 0

**Transmissão para end. de 16bits** → **DH = 0** e **DL < 0xFFFF**

**Transmissão Broadcast** → **DH = 0** e **DL = 0xFFFF**

# 5.1 Comunicação Básica

## ➤ Modo de Comunicação: **Unicast**<sup>\*Default</sup>

- Comunicação ponto-a-ponto, onde o endereço destino do módulo transmissor deve combinar com o endereço fonte do módulo receptor

**Endereço Fonte de 64bits:  $DH = SH$  e  $DL = SL$**

**Endereço Fonte de 16bits:  $DH = 0$  e  $DL = MY$**

**ACK**

Neste modo, os módulos, ao receberem um pacote, enviam um ACK para o transmissor. Caso este não o receba (em até 200ms), poderá reenviar o pacote até 3 vezes ou até o recebimento do ACK esperado.

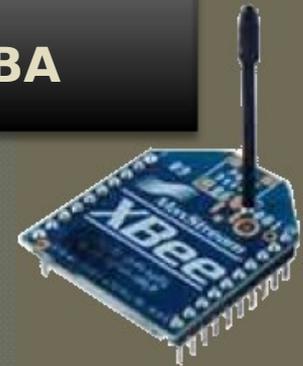
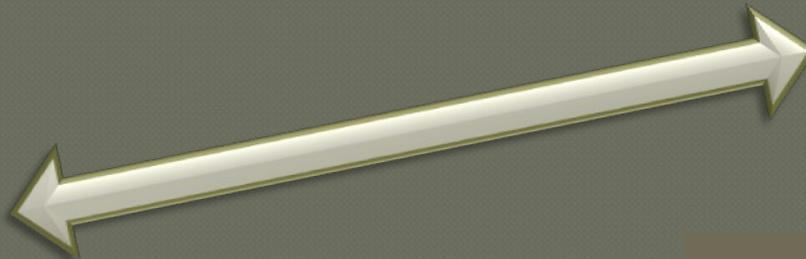
\* O pacote é reenviado após um período aleatório de até 48ms

# 5.1 Comunicação Básica

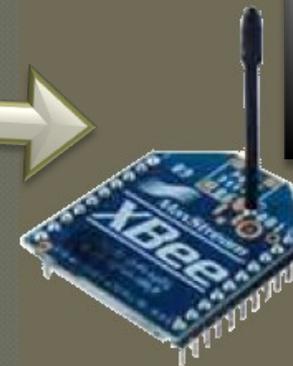
➤ Modo de Comunicação: **Unicast** <sup>\*Default</sup>

**DH = 1234567**  
**DL =**  
**21FEDCBA**  
**MY = 3**

**DH = 0**  
**DL = 0x0003**  
**SH = 1234567**  
**SL =**  
**21FEDCBA**  
**MY = FFFF**



**DH = 0**  
**DL = 5**  
**MY = 3**



\* Todos módulos: **CH 0x0C** e **ID 3332**

# 5.1 Comunicação Básica

## ➤ Comando **RR:**

**Parâmetros:** 0

- 6

**Default:** 0

- Leitura e Escrita.
- Reenvio do Xbee: Nº máx. de reenvios feitos pelo módulo em adição aos do MAC.
- Para cada reenvio do XBee, o MAC do 802.15.4 pode reenviar até 3x.
- Um bit é adicionado ao Maxstream Header do pacote requisitando um ACK.
- \*O reenvio não vale p/ pacotes retirados na transmissão com Sleep cíclico.

## ➤ Comando **EA:**

**Parâmetros:**

0 - 0xFFFF

- Leitura e Reset.
- Lê a contagem de falhas de recebimento de ACK. **EA = 0** zera a contagem.
- Incrementa-se toda vez que um módulo expira o nº de reenvios sem receber ACK.

# 5.1 Comunicação Básica

- Modo de Comunicação: **Broadcast**
  - Comunicação ponto-multiponto, onde todos os módulos no alcance do módulo transmissor se tornam receptores, independente de seu endereço fonte. [desde que trabalhem no mesmo canal e na mesma PAN]

Configuração:

**DH = 0**

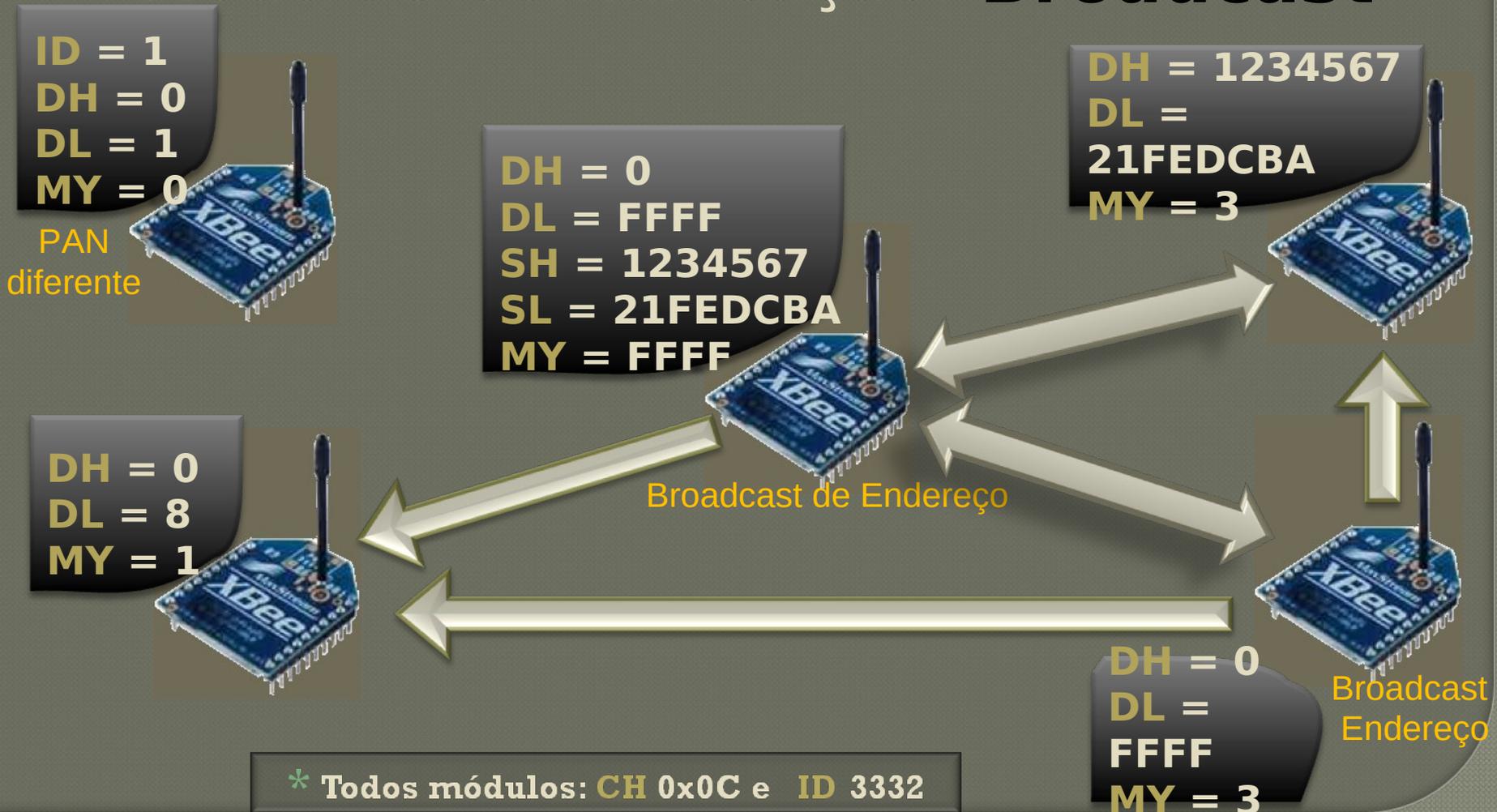
**DL = 0xFFFF**

**ACK**

Não há ACK de resposta, nem reenvio de mensagem

# 5.1 Comunicação Básica

## ➤ Modo de Comunicação: **Broadcast**



# 5.1 Comunicação Básica

## ➤ Modo de Comunicação: **Broadcast**

**ID = FFFF**  
**DL = 1**  
**MY = 0**



**ID = 0**  
**DL = FFFF**  
**MY = 0**

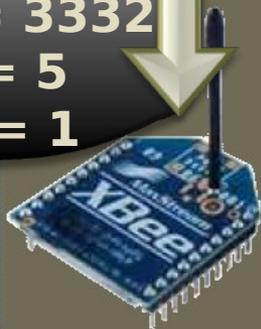


**ID = FFFF**  
**DL = FFFF**  
**MY = 0**

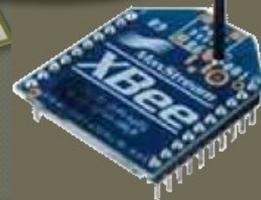


Broadcast de  
Endereço

**ID = 3332**  
**DL = 5**  
**MY = 1**



**ID = 0**  
**DL = 2**  
**MY = 5**



Broadcast PAN  
e de Endereço

\* Todos módulos: **CH 0x0C**

# Tutorial XBee

## 5. NetWorking

Comunicação Avançada

# 5.2 Comunicação Avançada

## ➤ Pacote de dados:

- Cada pacote contém seu end. destino e seu end. fonte.
- Os pacotes são estruturados de acordo com a especificação 802.15.4, com a adição de um header da MaxStream/Digi.

## ➤ Comando **MM** (MAC Mode):

- Leitura e Escrita.
- Habilita/Desabilita o header adicional

Desabilita os comandos: **ND** e **DN**  
Desabilita o reenvio por falha de CCA  
**Desabilita o uso de API**

Parâm. -

### Configuração

0	Maxstream Mode
1	802.15.4 (sem ACK)
2	802.15.4 (com ACK)

**Default:** 0

## 5.2 Comunicação Avançada

---

### ➤ Transmissão Direta:

- Os pacotes são transmitidos imediatamente para o end. destino.

### ➤ Transmissão Indireta:

- Necessário:

**Relação de Associação: end device ↔ Coordenador.**

- Utilidade: Garantir a entrega de pacotes, mesmo no modo Sleep.
- Os pacotes só são transmitidos quando requisitados pelo end device (FP).
- Utiliza-se o comando **FP** (Force Pool) para forçar o envio.
- Atualmente, um coordenador pode reter no máximo 2 msg indiretas

**\* Mais informações: Vide Associação e Sleep Mode.**

# 5.2 Comunicação Avançada

---

## ➤ **CSMA-CA** (Carrier Sense Multiple Access – Collision Avoidance):

- A especificação 802.15.4 trabalha com a prevenção de colisões.

## ➤ **CCA** (Clear Channel Assessment):

- Antes de transmitir um pacote, o módulo executa um CCA para verificar se o canal está livre para a transmissão. A energia no canal é detectada e comparada com o parâmetro definido em **CA**. Caso a energia exceda esse valor, o pacote não é transmitido. Um delay é inserido pelo comando **RN**.

\* Quando  $MM=0$ , o módulo tentará reenviar a mensagem mais 2 vezes após a falha CCA. Mesmo trabalhando em Broadcast.

# 5.2 Comunicação Avançada

## ➤ Comando **CA**:

**Parâmetros:** 0 – 0x50  
[-dBm]

• Leitura e Escrita. [CCA Threshold]

**Default:** 0x2C [-44 dBm]

• Valor limite de energia no canal para a falha CCA.

## ➤ Comando **RN**:

**Parâmetros:** 0 – 3  
[exponencial]

• Leitura e Escrita. [Randon Delay Slot]

**Default:** 0

• Valor mín. back-off (exp) do algoritmo CSMA-CA. (Equivale a “minBE” do 802.15.4)

• Caso RN=0 não há delay para a execução do primeiro CCA.

## ➤ Comando **EC**:

**Parâmetros:**  
0 – 0xFFFF

• Leitura e Reset. [CCA Failures]

• Lê o número de falhas CCA que o módulo sofreu. **EC 0** zera a contagem.

# Tutorial XBee

## 5. NetWorking

Comandos Importantes

# 5.3 Comandos Importantes

## ➤ Comando **AS**:

**Parâmetros:**

0 - 6

- Leitura. [Active Scan]
- Requer p/ Coordenadores em tds canais, em broadcast de PAN e END.
- Para cada Beacon recebido é criado um “PanDescriptor” (máx de 5):

1	Endereço ( <b>SH</b> + <b>SL</b> ou <b>MY</b> ) do Coordenador	
2	PAN ID ( <b>ID</b> ) do Coordenador	
3	Modo de Endereço	0x02 → 16 bits
		0x03 → 64 bits
4	Canal ( <b>CH</b> ) de trabalho do Coordenador	
5	Uso de Segurança	
6	ACL Entry	
7	Falha de Segurança	

8	Especificação de SuperFrame (2 bytes)	Bit 15 – Associação Permissão
		Bit 14 – Coordenador PAN
		Bit 13 – Reservado -----
		Bit 12 – Extensão vida Bateria
		Bit 8-11 – Final CAP Slot
		Bit 4-7 – Superframe Order
		Bit 0-3 – Beacon Order
9	GTS Permissão	
10	RSSI (retorna -dBm)	
11	TimeStamp	

**\*Tempo de Scan em cada canal:**  
 $[2^{(\text{parâmetro})}] * 15.36 \text{ ms}$

# 5.3 Comandos Importantes

## ➤ Comando **ED**: Parâmetros:

0 – 6

- Leitura. [Energy Scan]
- Retorna a energia máx. em cada canal (seguido por um CR, duplo no final).

Retorno de Carro



\*Tempo de Scan em cada canal:  
 $[2^{(\text{parâmetro})}] * 15.36 \text{ ms}$

## ➤ Comando **SD**: Parâmetros: 0 – 0xF

Default: 4

- Leitura e escrita. [Scan Duration]
- Determina o Scan Time do **AS** e do **ED** durante a associação.

\*Tempo de Scan em cada canal:  $[2^{(\text{parâmetro})}] * 15.36 \text{ ms}$

\*Tempo Total de Scan:  $[\# \text{ de canais}] * [2^{(\text{parâmetro})}] * 15.36 \text{ ms}$

# 5.3 Comandos Importantes

## ➤ Comando **SC**:

**Parâmetros:** 0 - 0xFFFF

**Default:** 0x1FFE (Tds XBee Pro)

- Leitura e escrita. [Scan Channels]
- Lista de Canais (em campo de bit) utilizados no **AS**, **ED**, e na associação.

Bit 0	Canal 0 [CH 0x0B]
Bit 1	Canal 1 [CH 0x0C]
Bit 2	Canal 2 [CH 0x0D]
Bit 3	Canal 3 [CH 0x0E]
Bit 4	Canal 4 [CH 0x0F]
Bit 5	Canal 5 [CH 0x10]

Bit 6	Canal 6 [CH 0x11]
Bit 7	Canal 7 [CH 0x12]
Bit 8	Canal 8 [CH 0x13]
Bit 9	Canal 9 [CH 0x14]
Bit 10	Canal 10 [CH 0x15]

Bit 11	Canal 11 [CH 0x16]
Bit 12	Canal 12 [CH 0x17]
Bit 13	Canal 13 [CH 0x18]
Bit 14	Canal 14 [CH 0x19]
Bit 15	Canal 15 [CH 0x1A]



X-Bee Pro: canais 0, 14 e 15 não são permitidos.

# Tutorial XBee

## 5. NetWorking

### Comandos Extras

## 5.4 Comandos Extras

### ➤ Comando **NI**:

**Parâmetros:**

0 - 20 char

- Leitura e Escrita. [Node Identifier]
- Seta ou lê uma string para identificar o módulo.
- **Requisitos:** Caract. ASCII, termina com CR ou com nº máx de bytes.

### ➤ Comando **DN**:

**Parâmetros:**

0 - 20 char

- [Destination Node]
- Usa um NI (parâmetro) para obter o endereço do módulo.
- Seta DL e DH iguais ao SL e SH do módulo com NI semelhante.
- Após sucesso, "OK" é retornado e sai automaticamente do modo AT.

\* **Erro:** [Sem resposta por 200ms, ou parâmetro não especificado]

# 5.4 Comandos Extras

## ➤ Comando **ND**:

### Parâmetros:

0 - 20 char

- Leitura. [Node Discover]
- Descreve todos os módulos operando no mesmo canal e PAN ID.
- Se parâmetro [**NI**] for usado, apenas módulos c/ **NI** semelhantes responderão.
- Ao completar, recebe-se 3 retornos de carro (CR) ou 1 em casos sem resposta.

### Resposta ND: AT Command

1	Endereço 16bits ( <b>MY</b> )	[CR]
2	Endereço 32bit ( <b>SH</b> )	[CR]
3	Endereço 32bit ( <b>SL</b> )	[CR]
4	Força do Sinal Recebido ( <b>DB</b> )	[CR]
5	Identificador do Nó ( <b>NI</b> )	[CR]
6	[CR]	

### Resposta ND: API Mode

2 Bytes	Endereço 16bits ( <b>MY</b> )
4 Bytes	Endereço 32bit ( <b>SH</b> )
4 Bytes	Endereço 32bit ( <b>SL</b> )
1 Bytes	Força do Sinal Recebido ( <b>DB</b> )
String máx 20 Bytes	Identificador do Nó ( <b>NI</b> ) (string terminada em NULL)

# 5.4 Comandos Extras

## ➤ Comando **NT**:

- Leitura e Escrita [Node Discover Time]
- Tempo que o módulo base esperará a resposta dos outros nós.
- Respostas recebidas depois deste tempo serão descartadas.

### Parâmetros:

0x01 - 0xFC [\*10msec]

### Default:

0x19 [25\*10=250ms]

\*O valor de **NT** é transmitido com o comando **ND**. Os módulos remotos definem, então, um hold-off time aleatório baseado neste valor. Este tempo nunca passa de 250ms, para assim dar tempo de todos os módulos responderem.

# 5.4 Comandos Extras

## ➤ Comando **DB**:

### Parâmetros:

0 - 20 char

- Leitura. [Receive Signal Strength]
- Força do sinal recebido do último pacote.
- O menor valor possível é 40dBm [0x28].

### Parâm. - XBee - XBee Pro

0	-10dBm	+10dBm
1	-6dBm	+12dBm
2	-4dBm	+14dBm
3	-2dBm	+16dBm
4	0dBm	+18dBm

**Default:** 4

## ➤ Comando **PL**:

- Leitura e Escrita [Power Level]
- Magnitude do sinal RF transmitido.

\* [dBm indica a magnitude do sinal relativo a 1mW,  
assim  $80\text{dBm} = 10^{(-80/10)} = 10\text{pW}$  ]

# 5.4 Comandos Extras

## ➤ Comando **EE**:

- [AES Encryption Enable]
- Habilita ou Desabilita o uso da criptografia AES de 128bits.
- Assim, o módulo utilizaria o protocolo de segurança default do 802.15.4 mais a cript. AES com uma chave de 128bits, igual para os módulos comunicantes.
- ✓ O módulo sempre utilizará o end. de 64bits (isso não muda como MY, DL.. funciona);
- ✓ O tamanho máximo dos pacotes RF passa a ser 95 Bytes.
- ✓ Módulos sem chave (ou incorreta) recebem o pacote criptografado, mas este é ignorado.
- ✓ Módulos com criptografia recebem pacotes sem criptografia mas estes são ignorados.

### Parâm. - Config.

0 Desabilitado

1 Habilitado

**Default:** 0

## ➤ Comando **KY**:

- Escrita [AES Encryption Key]
- Define a chave de criptografia do módulo (parâm. Obrigatório).

### Parâmetros:

0 - (hexadecimal de 16 Bytes)

# Tutorial XBee

## 6. Associação

# 6. Associação

---

- A associação é a relação de sociedade entre um ou mais End Devices e um Coordenador.
- Útil em cenários que necessitam de controle central.
- A associação é formada por um coordenador e um ou mais End Devices ocupando a mesma PAN ID e Canal.
- O End Device não necessariamente devem conhecer o endereço, Pan, ou canal do coordenador.

**\* Requisito:** É necessário coordenador e End Devices com mesmo firmware.

# 6. Associação

## ➤ Comandos de Associação:

- Coordenador:

**CE** (Coordinator Enable)

**A2** (Coordinator Association)

- End Device:

**A1** (End Device Association)

**DA** (Force Disassociation)

**AI** (Association Indication)

## ➤ Comando **CE**:

- Leitura e Escrita - Coordenador.
- [Coordinator Enable] Comportamento do Módulo.

### Parâm. - Configuração

0 End Device

1 Coordenador

**Default:** 0

# 6. Associação

**Parâm.:** 0 - 0x07 [0111]

**Default:** 0

**Ex.:** 0x06 [0110]

## ➤ Comando **A2**:

- Leitura e Escrita - **Coordenador**.
- Comportamento e opções de associação do Coordenador.

<b>Bit: 0 *</b> <b>Re-designar PAN ID</b>	<b>0</b>	Não executa o <b>Active Scan</b> e <b>coordenador opera na PAN ID atual</b> .
	<b>1</b>	Executa o <b>Active Scan</b> em todos canais e determina um <b>PAN ID ÚNICO</b> para o coordenador.
<b>Bit: 1 *</b> <b>Re-designar Canal</b>	<b>0</b>	Não executa o <b>Energy Scan</b> e <b>coordenador opera no canal atual</b> .
	<b>1</b>	Executa o <b>Energy Scan</b> em todos canais e determina o <b>Canal com menor nível de energia</b> para o coordenador.
<b>Bit: 2</b> <b>Permitir Associação</b>	<b>0</b>	Coordenador não permitirá a associação de End Devices.
	<b>1</b>	Coordenador permitirá a associação de End Devices.
<b>Bit: 3-7</b>	[Reservado].	

\* [Scan Time] É determinado pelo comando **SD**.

\* O MAC do coordenador será resetado se houver alteração em **ID**, **CH**, **MY**, ou **A2** (bits 0 ou 1).

# 6. Associação

## ➤ Comando **A1**:

**Parâm.:** 0 - 0xF [1111]

**Default:** 0

- Leitura e Escrita - **End Device.**

<b>Bit: 0</b> <b>Re-designar PAN ID</b>	<b>0</b>	Irá se associar a um coordenador que tenha o mesmo Pan ID [ <b>ID</b> ]
	<b>1</b>	Irá se associar a um coordenador operando em <b>qualquer PAN ID.</b>
<b>Bit: 1</b> <b>Re-designar Canal</b>	<b>0</b>	Irá se associar a um coordenador operando no mesmo canal [ <b>CH</b> ].
	<b>1</b>	Irá se associar a um coordenador operando em <b>qualquer canal.</b>
<b>Bit: 2 *</b> <b>Auto-Associação</b>	<b>0</b>	Device não tentará se associar.
	<b>1</b>	Device <b>tentará se associar com coord. que permite associação.</b>
<b>Bit: 3</b> <b>Ação ao Acordar</b>	<b>0</b>	Ao acordar, o módulo não requisitará o coordenador por pacotes pendentes.
	<b>1</b>	Ao acordar, <b>requisitará o coordenador por pacotes pendentes.</b>
<b>Bit: 4-7</b>		[Reservado].

- \* [Associação] É feita através de um active scan que gera uma lista de PAN IDs e canais. Então de acordo com as regras (bits 0 e 1) é selecionado um coordenador. Se houver múltiplas opções, então escolhe-se a PAN com sinal mais forte. Se não houver opções, o módulo entra em Sleep Mode (se SM=1) ou tenta novamente a associação.

\* Ao encontrar coordenador válido: End Device enviará um Association Request e, então, esperará um Association Confirmation

# 6. Associação

## ➤ Comando DA:

- Apenas Escrita - **End Device.** [Force Disassociation]
- Força imediatamente a dissociação e tenta a associação novamente.

## ➤ Comando AI:

- Leitura - **ambos.**
- Indica a ocorrência de erros durante a última tentativa de associação.

Returned Value (Hex)	Association Indication
0x00	Successful Completion - Coordinator successfully started or End Device association complete
0x01	Active Scan Timeout
0x02	Active Scan found no PANs
0x03	Active Scan found PAN, but the Coordinator Allow Association bit is not set
0x04	Active Scan found PAN, but Coordinator and End Device are not configured to support beacons
0x05	Active Scan found PAN, but Coordinator ID (PAN ID) value does not match the ID of the End Device
0x06	Active Scan found PAN, but Coordinator CH (Channel) value does not match the CH of the End Device
0x07	Energy Scan Timeout
0x08	Coordinator start request failed
0x09	Coordinator could not start due to Invalid Parameter
0x0A	Coordinator Realignment is in progress
0x0B	Association Request not sent
0x0C	Association Request timed out - no reply was received
0x0D	Association Request had an Invalid Parameter
0x0E	Association Request Channel Access Failure - Request was not transmitted - CCA failure
0x0F	Remote Coordinator did not send an ACK after Association Request was sent
0x10	Remote Coordinator did not reply to the Association Request, but an ACK was received after sending the request
0x11	[reserved]
0x12	Sync-Loss - Lost synchronization with a Beaconsing Coordinator
0x13	Disassociated - No longer associated to Coordinator

# Tutorial XBee

## 7. Sleep Mode

# 7. Sleep Mode

---

- O modo Sleep é de grande importância em aplicações wireless devido ao baixo consumo de energia quando neste modo.
- Enquanto, se encontrar no modo Sleep, o módulo não receberá pacotes RF nem se comunicará via serial.
- O XBee possui quatro modos diferentes de Sleep, definidos pelo comando **SM** (quando **SM**  $\neq$  0):
  - Pin Hibernate (**SM=1**);
  - Pin Doze (**SM=2**);
  - Cyclic Sleep (**SM=4**);
  - Cyclic Sleep w/ Pin Wake-up (**SM=5**).

# 7. Sleep Mode

---

## ➤ Pin Hibernate (SM 1) :

- **Controle:** Voltage Level - Pino **Sleep\_RQ** (pino 9).
  - **Corrente Típica:**  $< 10\mu\text{A}$  (@3.0 Vcc).
  - **Tempo de Wake-up:** 13.2 msec.
- Quando **Sleep\_RQ** é declarado (nível alto), o módulo termina qualquer atividade, entra no idle mode e depois no sleep.
- O módulo acorda quando redeclara-se **Sleep\_RQ** para nível baixo.
- \* O módulo estará pronto para transmitir ou receber pacotes, quando o CTS é levado a nível baixo.
  - \*\* O Sleep\_RQ (pino 9) deve ser redeclorado para nível baixo, por ao menos 2 “byte times” depois que CTS é levado a nível baixo. Isto garante que há tempo para os dados entrarem no DI buffer.

# 7. Sleep Mode

## ➤ Pin Doze (SM 2):

- **Controle:** Voltage Level – Pino **Sleep\_RQ** (pino 9).
  - **Corrente Típica:**  $< 50\mu\text{A}$  (@3.0 Vcc).
  - **Tempo de Wake-up:** 2 msec.
- Este modo é semelhante ao Pin Hibernate. A única diferença é que neste modo o módulo acorda mais rápido (2 msec) e, também, consome mais energia ( $<50\mu\text{A}$  para 3V).
- \* O módulo estará pronto para transmitir ou receber pacotes, quando o CTS é levado a nível baixo.
  - \*\* O Sleep\_RQ (pino 9) deve ser redeclarado para nível baixo, por ao menos 2 “byte times” depois que CTS é levado a nível baixo. Isto garante que há tempo para os dados entrarem no DI buffer.

# 7. Sleep Mode

## ➤ Cyclic Sleep (SM 4):

- **Controle:** Temporal Cíclico.
  - **Corrente Típica:**  $< 50\mu\text{A}$  (@3.0 Vcc).
  - **Tempo de Wake-up:** 2 msec.
- 
- O módulo fica em modo sleep por um período (definido por **SP**).
  - Ao acordar, verifica (“poll request”) se há algum dado armazenado no módulo definido como “Cyclic Sleep Coordinator” (**SM=0**, e **CE=1**).
  - O módulo retornará ao modo sleep, assim que não houver nenhuma atividade (serial ou RF) pela duração de certo tempo (definido por **ST**).

\* Qualquer atividade resetará o timer ST (Excessão: Mensagem Global). Assim o módulo só volta a dormir depois de que não haja atividades por ST tempo.

# 7. Sleep Mode

## ➤ Cyclic Sleep w/ Pin Wake-up (SM 5):

- **Controle:** Temporal Cíclico e Voltage Level – Pino **Sleep\_RQ**.
  - **Corrente Típica:**  $< 50\mu\text{A}$  (@3.0 Vcc).
  - **Tempo de Wake-up:** 2 msec.
- 
- O modo cíclico funciona da mesma maneira descrita anteriormente.
  - O Sleep\_RQ (pino 9) serve apenas para acordar e é “edge triggered” (1 → 0). A volta p/ sleep ocorre como descrito (depois de **ST** tempo).
  - A vantagem é o acordar devido a algum evento ou interrupção.

\* Enquanto o módulo estiver acordado, qualquer atividade no Sleep\_RQ (pino 9) será ignorada.

# 7. Sleep Mode

## ➤ Resumo:

- O módulo entrará em modo Sleep, desde que SM seja diferente de 0 e uma das condições abaixo cumpridas:
  - O pino Sleep\_RQ (pino 9) esteja declarado (nível alto) [**SM**=1,2, ou 5]
  - O módulo esteja idle por uma duração de tempo igual a **ST** [**SM** 4 ou 5]

Sleep Mode Setting	Transition into Sleep Mode	Transition out of Sleep Mode (wake)	Characteristics	Related Commands	Power Consumption
Pin Hibernate (SM = 1)	Assert (high) Sleep_RQ (pin 9)	De-assert (low) Sleep_RQ	Pin/Host-controlled / NonBeacon systems only / Lowest Power	(SM)	< 10 $\mu$ A (@3.0 VCC)
Pin Doze (SM = 2)	Assert (high) Sleep_RQ (pin 9)	De-assert (low) Sleep_RQ	Pin/Host-controlled / NonBeacon systems only / Fastest wake-up	(SM)	< 50 $\mu$ A
Cyclic Sleep (SM = 4 - 5)	Automatic transition to Sleep Mode as defined by the SM (Sleep Mode) and ST (Time before Sleep) parameters.	Transition occurs after the cyclic sleep time interval elapses. The time interval is defined by the SP (Cyclic Sleep Period) parameter.	RF module wakes in pre-determined time intervals to detect if RF data is present / When SM = 5, NonBeacon systems only	(SM), SP, ST	< 50 $\mu$ A when sleeping

# 7. Sleep Mode

## ➤ Comando **SM**:

- Leitura e escrita. [Sleep Mode]
- Comando principal. Define o modo.
- \* Sleep Coordinator apenas para firm-  
Ware de versão 1.x06 antigas.

### Parâm. - Configuração

0	Desabilitado
1	Pin Hibernate
2	Pin Doze
3	-----Reservado-----
4	Cyclic Sleep
5	Cyclic Sleep w/ pin wake-up
6	Sleep Coordinator *

**Default:** 0

## ➤ Comando **FP**:

- Apenas Escrita [Force Poll]. Requisita mensagens indiretas do coordenador.

## ➤ Comando **DP**:

- Leitura e escrita. [Disassociation CS period]
- Tempo de Sleep de módulos (com Cyclic Sleep) que estão configurados para serem associados, mas não estão associados à um coordenador.

**Parâmetros:** 0x01 - 0x68B0[\*10ms]

**Default:** 0x03E8 [1000\*10=10s]

# 7. Sleep Mode

## ➤ Comando **SP**:

- Leitura e escrita. [Cyclic Sleep Period]
- Duração de tempo que o módulo (com Cyclic Sleep) permanece em modo sleep. Máximo:  $0x68B0 = 268$  segundos.

➤ **Coordenador**: Tempo de armazenagem de mensagens indiretas:  **$2,5*SP$** .

## ➤ Comando **ST**:

- Leitura e escrita. [Time Before Sleep]
- Período de inatividade (sem dados serial ou RF sendo enviado ou recebido) antes da reativação do modo sleep em módulos com Cyclic Sleep.

\* **Coordenador** deve possuir o **mesmo ST do end Device**.

### Parâmetros:

0x01 - 0x68B0

[\*10mseg]

**Default:** 0

### Parâmetros:

0x01 - 0xFFFF

[\*1mseg]

**Default:** 0x1388 [5 seg.]

# 7. Sleep Mode

## ➤ Consumo de corrente:

- No sleep mode, o consumo de corrente é extremamente sensível aos valores da tensão de alimentação:

Vcc (V)	XBee			XBee-PRO		
	SM=1	SM=2	SM=4,5	SM=1	SM=2	SM=4,5
2.8–3.0	<3 $\mu$ A	<35 $\mu$ A	<34 $\mu$ A	<4 $\mu$ A	<34 $\mu$ A	<34 $\mu$ A
3.1	8 $\mu$ A	37mA	36 $\mu$ A	12 $\mu$ A	39 $\mu$ A	37 $\mu$ A
3.2	32 $\mu$ A	48 $\mu$ A	49 $\mu$ A	45 $\mu$ A	60 $\mu$ A	55 $\mu$ A
3.3	101 $\mu$ A	83 $\mu$ A	100 $\mu$ A	130 $\mu$ A	115 $\mu$ A	120 $\mu$ A
3.4	255 $\mu$ A	170 $\mu$ A	240 $\mu$ A	310 $\mu$ A	260 $\mu$ A	290 $\mu$ A

\* Tensões acima de 3.0V causam um alto consumo de corrente.