

UM SISTEMA DE COMUNICAÇÃO SEM FIO PARA A
AQUISIÇÃO DE DADOS EM AMBIENTE FABRIL

Orlens da Silva Melo

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM
ENGENHARIA ELÉTRICA.

Aprovada por:

Prof. Jorge de Souza Leão, Dr. Ing.

Prof. Aloysio de Castro Pinto Pedrosa, Docteur

Prof. Luís Felipe Magalhães de Moraes, Ph.D

RIO DE JANEIRO, RJ – BRASIL

ABRIL DE 2005

MELO, ORLENS DA SILVA

Um sistema de comunicação sem fio para a aquisição de dados em ambiente fabril [Rio de Janeiro] 2005

XII, 114p. 29,7cm (COPPE/UFRJ, M.Sc. Engenharia Elétrica, 2005)

Tese – Universidade Federal do Rio de Janeiro, COPPE.

1. Bluetooth
2. Redes sem Fio

I. COPPE/UFRJ II. Título (série)

Este trabalho é humildemente dedicado a Deus, que representa a infinita bondade e infinita justiça, ao mestre querido Jesus que nos ensinou e exemplificou todas as virtudes de um Homem de Bem, e à Espiritualidade Superior, que são as Vozes do Senhor a nos amparar na caminhada evolutiva intelecto-moral.

Agradecimentos

Agradeço a meus queridos pais, Orestes e Marina, aos amados irmãos consangüíneos, à grande família espiritual, aos valorosos amigos de trabalho profissional e aos novos amigos compostos pelos coordenadores e orientadores deste programa de mestrado.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

UM SISTEMA DE COMUNICAÇÃO SEM FIO PARA A
AQUISIÇÃO DE DADOS EM AMBIENTE FABRIL

Orlens da Silva Melo

Abril/2005

Orientador: Jorge de Souza Leão

Programa: Engenharia Elétrica

Este trabalho propõe o uso do padrão Bluetooth em ambiente industrial para subsidiar aplicações que demandem a aquisição de dados onde não há necessidade de altas taxas de transferência de dados, o que caracteriza bem os sistemas de teste responsáveis pela validação dos produtos em linha de produção. Destaca-se o padrão quanto à imunidade a interferência conseguida pelo uso conjunto das técnicas de diversidade na frequência - FHSS e controle de erro. Caracteriza-se seu comportamento em relação à propagação de sinais em ambiente fechado, como também, faz-se uma análise de performance de uma topologia de rede que contempla a formação de *piconets* ponto-a-ponto de acordo com a demanda, demonstrando-se, por meio da implementação em uma aplicação real, toda a potencialidade do padrão Bluetooth.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

A WIRELESS COMMUNICATION SYSTEM FOR DATA ACQUISITION
IN A MANUFACTURING ENVIRONMENT

Orlens da Silva Melo

April/2005

Advisor: Jorge de Souza Leão

Department: Electrical Engineering

This work proposes the use of the Bluetooth standard in an industrial environment for data acquisition applications with low data rates. This is the case of test systems responsible for the functional validation of the products in a manufacturing line. One of characteristic of the Bluetooth standard is the immunity to the interference provided by the joint used of the techniques of diversity in the frequency – FHSS and error control. The standard is characterized as to the behavior in relation to the propagation of radio signals in indoor environment, as also, is made an analysis about the performance of point-to-point piconets established in accordance with the demand, demonstrating all the potentiality of the Bluetooth standard by implementation it in a real application.

Sumário

1	Introdução	1
1.1	Apresentação	1
1.2	Objetivos e motivação	2
1.3	Organização	2
1.4	Contribuições.....	3
2	Fundamentos teóricos	4
2.1	Descrição geral do padrão Bluetooth.....	4
2.1.1	A pilha de protocolos.....	7
2.1.2	Os perfis de uso	8
2.2	Sistema de rádio Bluetooth.....	9
2.2.1	Introdução	9
2.2.2	O esquema de modulação	9
2.2.3	A capacidade do canal	10
2.2.4	A imunidade à interferência	11
2.2.5	FHSS.....	13
2.2.6	Performance do transmissor	16
2.2.7	Performance do receptor.....	20
2.3	Estudo da propagação em ambientes fechados e cálculo do alcance do rádio Bluetooth	24
2.3.1	Mecanismos da propagação em ambientes fechados	24
2.3.2	Perdas em larga escala.....	26
2.3.3	Multipercurso	38
2.3.4	Antenas	48
2.4	Banda base e Controlador de enlace (<i>Link Controller</i>).....	51
2.4.1	Descrição básica	51
2.4.2	Operação em uma configuração ponto-a-ponto.....	52
2.4.3	Operação em uma configuração ponto a multiponto.....	53
2.4.4	Pacotes <i>multislot</i>	54

2.4.5	Temporização da <i>piconet</i>	55
2.4.6	Enlaces físicos	56
2.4.7	Endereços e nomes dos dispositivos Bluetooth.....	58
2.4.8	Formato dos pacotes de banda base.....	60
2.4.9	<i>Whitening</i>	70
2.4.10	Canais lógicos.....	71
2.5	Estabelecimento de conexão.....	72
2.5.1	Descrição geral	72
2.5.2	Diagrama de estados para uma <i>piconet</i> ponto-a-ponto.....	73
2.5.3	Diagrama de estados dos dispositivos versáteis	74
2.5.4	Dificuldades do processo de estabelecimento de conexão	76
2.5.5	Descrição geral do processo de <i>Page</i>	77
2.5.6	Descrição geral do processo de <i>Inquiry</i>	81
2.5.7	Quantificação dos tempos de conexão	84
2.6	Gerente de enlace (<i>Link manager</i>).....	84
2.6.1	Descrição geral	84
2.6.2	Tarefas realizadas pelo gerente de enlace (<i>link manager</i>).....	85
2.7	802.11	87
3	O Sistema de teste.....	89
3.1	Descrição de hardware.....	90
3.2	Descrição de software.....	91
3.2.1	Software básico	91
3.2.2	Capacidades das memórias <i>Flash</i> e RAM:.....	91
4	Proposta do sistema de comunicação.....	93
4.1	Requisitos do sistema de comunicação	93
4.2	Escolha do padrão Bluetooth.....	94
4.3	Análise crítica dos requisitos.....	94
4.4	Estudo da topologia do sistema de comunicação.	96
4.5	Solução implementada.....	97
4.5.1	Descrição geral	97

4.5.2	Estabelecimento de conexão.....	98
4.5.3	Descrição dos relatórios de teste	99
4.5.4	Características do sistema embarcado SPP	101
5	Resultados	104
5.1	Análise de performance do sistema de comunicação proposto	104
5.2	Experimento – Probabilidade de colisão	107
5.2.1	Descrição do ambiente experimental.....	107
5.2.2	Resultados - Estatística de perda dos relatórios de teste	110
5.3	Experimento – Tempo de estabelecimento de conexão.....	111
5.3.1	Características de operação do cliente A.....	111
5.3.2	Características de operação do cliente B.....	111
5.3.3	Resultados – Tempo máximo de espera	111
6	Conclusões	113
6.1	Sobre o Trabalho Realizado	113
6.2	Sobre o Trabalho que pode ser Realizado	113
7	Referências Bibliográficas	114

Lista de Tabelas

Tabela 1. Classes de potência.....	16
Tabela 2. Potência de saída - dispositivos classe 2.....	16
Tabela 3. Emissões espúrias fora da banda	19
Tabela 4. Variação de frequência da portadora	19
Tabela 5. Relação C/I	21
Tabela 6. Nível máximo de potência permissível fora da banda.....	22
Tabela 7. Emissões espúrias	24
Tabela 8. Valores típicos de n para diversos cenários.....	28
Tabela 9. Alcance da interface aérea por meio da PL simplificada	30
Tabela 10. Fatores de atenuação por obstáculo	36
Tabela 11. Fator de atenuação por andar.....	36
Tabela 12. Otimização do <i>payload</i> devido o uso de pacotes <i>multislots</i>	55
Tabela 13. Composição do BD_ADDR	58
Tabela 14. Modos de <i>scan repetition</i>	66
Tabela 15. Modos de <i>page scan</i>	67
Tabela 16. Vazão máxima de um enlace ACL em relação ao tipo de pacote	68
Tabela 17. Composição dos pacotes SCO.....	70
Tabela 18. Geração das seqüências de salto de um dispositivo Bluetooth.....	77
Tabela 19. Sumarização dos tempos para estabelecimento de conexão.....	84
Tabela 20. Gerente de enlace – principais tarefas	86
Tabela 21. Variações do padrão 802.11	87
Tabela 22. Cálculo de d_i para o ambiente experimental	109

Lista de Figuras

Figura 1. Topologias de redes Bluetooth.....	6
Figura 2. A pilha de protocolos	7
Figura 3. Os perfis de uso.....	8
Figura 4. Bluetooth coexistindo com Wi-Fi.....	12
Figura 5. Sistema de comunicação FHSS.....	13
Figura 6. Salto entre canais - Domínio da Frequência	14
Figura 7. Salto entre canais – Domínio do tempo	15
Figura 8. Probabilidade de perda de pacotes e x número de <i>piconets</i> co-aloçadas	15
Figura 9. <i>Golden Receive Power Range</i>	17
Figura 10. Processo de obtenção da FI.....	22
Figura 11. Rejeição a imagem.....	23
Figura 12. Propagação de uma onda de rádio na presença de obstáculos	25
Figura 13. Comportamento da PL em espaço livre	27
Figura 14. PL simplificada x <i>Primary Ray Tracing</i>	37
Figura 15. Fragmentação da energia do sinal na presença de obstáculo	38
Figura 16. Ordem das reflexões.....	39
Figura 17. Desvanecimento do sinal devido ao Espalhamento Doppler	42
Figura 18. Caminho direto LOS x Caminho refletido.....	45
Figura 19. Perdas devido a falta de casamento de impedância	49
Figura 20. Antena <i>PIFA</i>	50
Figura 21. Antena meia-onda	50
Figura 22. Antena patch em um CI de rádio Bluetooth.....	50
Figura 23. Operação em uma configuração ponto-a-ponto	52
Figura 24. Operação em uma configuração ponto-a-multiponto.....	54

Figura 25. Pacotes <i>multislot</i>	55
Figura 26. Tráfego ACL e SCO dentro da <i>piconet</i>	57
Figura 27. Composição dos pacotes de banda base.....	60
Figura 28. Composição do código de acesso.....	60
Figura 29. Composição do cabeçalho.....	63
Figura 30. Classificação dos pacotes.....	63
Figura 31. Processo de seleção da frequência de salto	65
Figura 32. Composição do campo <i>payload</i>	66
Figura 33. Campo <i>payload</i> de um pacote ACL.....	67
Figura 34. Comportamento da vazão em relação ao BER do canal	69
Figura 35. Processo de transmissão e recepção de um pacote.....	71
Figura 36. Diagrama de estados - configuração ponto-a-ponto.....	73
Figura 37. Diagrama de estados para os dispositivos versáteis.....	74
Figura 38. Processo de <i>page</i> no domínio do tempo.....	78
Figura 39. Processo de <i>inquiry</i> no domínio do tempo.....	83
Figura 40. BSS.....	88
Figura 41. Formação de <i>piconets</i> sob demanda.....	97
Figura 42. Diagrama de estados dos dispositivos servidor e cliente	98
Figura 43. Pilha de protocolos e Interface entre o sistema de teste e o sistema embarcado Bluetooth.....	102
Figura 44. <i>Piconet</i> coexistindo com futuros mestres em <i>page scan</i>	104
Figura 45. Probabilidade de perda de pacotes x N° de canais com interferência	106
Figura 46. Descrição do ambiente experimental	107

Capítulo 1

Introdução

1.1 Apresentação

As indústrias do pólo eletro-eletrônico de Manaus – Am possuem em suas linhas de produção dispositivos de teste responsáveis pela validação funcional dos seus produtos ou das partes que compõem os seus produtos, em se tratando de linhas de montagem seriais. Dentro da linha de produção, estes dispositivos estão localizados na fase de teste funcional. Ao exercer a tarefa de validação, estes dispositivos realizam a adaptação dos requisitos contidos nas extensas especificações de teste dos produtos, para a realidade de uma linha de produção, a saber: tempo de teste extremamente reduzido e falta de qualificação técnica dos operadores. Para tal, dependendo da complexidade destas especificações, estes dispositivos devem ser dotados de um alto grau de automatização, para então atender aos requisitos de produção supracitados. Durante a realização dos testes, estes dispositivos identificam os defeitos e os externam por meio de sua interface homem-máquina, emitindo um relatório impresso ou simplesmente uma mensagem de erro em um monitor ou display ou, simplesmente, uma sinalização luminosa, para o chamado teste “*GO-NO-GO*”. Normalmente estes erros estão relacionados com falhas cometidas nas fases anteriores do processo produtivo, como: inserção automática, inserção manual e soldagem. Vale salientar que em geral não há nenhuma interface com um sistema de aquisição de dados (*MES – manufacture enterprise system*). O monitoramento destes dados é feito manualmente por profissionais denominados supervisores da linha de produção ou inspetores da qualidade que, além de outras atribuições, recolhem-nos em tempos prefixados e realizam o tratamento estatístico, para assim, terem subsídios para atuar corretivamente no processo produtivo. Desta forma, os erros não epidêmicos são somente enxergados, no melhor caso, no final dos turnos de produção. Este cenário pode atender muito bem a alguns processos industriais, porém, em um cenário caracterizado por uma crescente necessidade de aumento do volume de produção, diversificação de produtos e altos requisitos de qualidade, a aquisição dos dados do ambiente de teste é de suma importância para o gerenciamento do negócio.

Neste contexto, esta tese se propõe atuar no estudo de uma interface hardware e software para aquisição de dados de dispositivos de teste. A tecnologia a ser estudada para este fim será a tecnologia Bluetooth.

1.2 Objetivos e motivação

Este trabalho objetiva implementar uma arquitetura hardware e software, utilizando o padrão Bluetooth, que possibilite a aquisição de dados de produção dos dispositivos situados no nível 1 da hierarquia clássica usada na automação industrial, permitindo:

- A aquisição de dados do ambiente de teste;
- Menor tempo de preparação do posto de trabalho, devido à inexistência da infra-estrutura de cabos;
- Facilidade de instalação, dispensando a necessidade de obras civis para suportar a distribuição de cabos dentro de um ambiente de fábrica;
- Maior agilidade na manutenção.

A motivação deste trabalho foi:

- Necessidade de implementações que avaliem a utilização de tecnologias sem fio no campo da automação industrial.
- Necessidade de informatização do ambiente de testes;
- Necessidade de um sistema de aquisição de dados que atenda à flexibilidade exigida por uma linha de produção. Haja vista que estas podem operar em regime de “mix” de produção, onde vários produtos podem ser montados em uma mesma linha de produção, podendo existir para cada modelo a necessidade de alteração do *layout* dos postos de trabalho;

1.3 Organização

A apresentação formal deste trabalho foi estruturada em cinco capítulos. No Capítulo 2 está exposta a revisão bibliográfica que fundamentou a implementação da solução. Este estudo analisa a propagação de sinais em ambientes fechados e depois a pilha de protocolos Bluetooth começando pelas camadas mais baixas, destacando em cada uma o que é mais relevante para o objetivo proposto. Faz-se também um estudo crítico sobre os tempos envolvidos nos procedimentos de estabelecimento de conexão. No capítulo 3 descreve-se a arquitetura de hardware e de software do sistema de teste desenvolvido. No capítulo 4 descrevem-se os requisitos da aplicação, faz-se uma análise crítica à luz da revisão bibliográfica, faz-se um estudo sobre a topologia do sistema de comunicação

e, por fim, descreve-se a solução implementada. No capítulo 5 faz-se uma análise sobre a performance do sistema de comunicação proposto e apresentam-se os resultados dos experimentos em laboratório. No capítulo 6 o trabalho é finalizado com as conclusões baseadas nos resultados obtidos.

1.4 Contribuições

Neste trabalho desenvolveu-se uma nova filosofia de dispositivos de teste caracterizado pela sua generalidade no uso para validação de módulos de LCD usados em telefones celulares, e pela interface de comunicação sem fio utilizando o padrão Bluetooth. A rede local é estabelecida através de uma topologia baseada na formação de *piconets* de acordo com a demanda. Mostrando, assim, a grande aplicabilidade do padrão Bluetooth em soluções cujos requisitos são baixo custo, imunidade à interferência, a segurança e baixas taxas de transferência de dados.

Capítulo 2

Fundamentos teóricos

2.1 Descrição geral do padrão Bluetooth

O Bluetooth é um padrão que define um canal de rádio de curta-distância, destinado a prover a comunicação entre dispositivos eletrônicos, sejam eles portáteis ou não. Suas principais características são: robustez, baixa complexidade, baixa potência e baixo custo.

Para garantir a natureza de interface aberta, o Bluetooth opera na banda ISM (*Industrial, Scientific, Medical*), centrada em 2.45 GHz. Historicamente, esta banda foi inicialmente reservada para aplicações específicas de alguns grupos de profissionais, porém, posteriormente foi aberta mundialmente para aplicações comerciais. Ao contrário do que se pensa, o termo não - licenciado não quer dizer que não haja regulação. A maioria dos países regulamenta o uso desta banda de frequência, porém há diferenças entre eles quanto às regras a serem seguidas. Naturalmente, para que o Bluetooth seja adotado mundialmente como padrão para comunicação sem fio à curta distância, haverá de ter um esforço no sentido de unificar essas regras. Neste contexto, vale ressaltar que as análises deste trabalho estão baseadas na versão 1.1 [1] do padrão Bluetooth.

Na maioria dos países a referida banda ocupa a faixa de 2400 – 2483,5 MHz. Uma modulação FSK é aplicada para minimizar a complexidade do transmissor e do receptor. O esquema de acesso ao meio é feito através de um rádio FHSS (*frequency hopping spread spectrum*) com uma taxa de saltos de 1600 saltos por segundo. O espectro é dividido em 79 canais com espaçamento de 1 MHz entre eles. A taxa de transferência de símbolos é de 1 Ms/s. Cada canal é dividido no tempo em uma unidade básica chamada *slot* (janela ou intervalo de tempo), tendo a duração de 625 μ s. Para uma transferência de dados *full duplex* forçada, um esquema de TDD (*Time Division Duplex*) é implementado. As informações são enviadas dentro de pacotes. Cada pacote é transmitido em uma diferente frequência ou canal. Um pacote normalmente tem o tempo de duração de um *slot*, podendo estender-se até 5 (cinco) *slots*.

O protocolo de banda base Bluetooth usa a combinação de chaveamento de pacotes através dos enlaces físicos ACL (*asynchronous connectionless*), e de circuitos através

dos enlaces físicos SCO (*synchronous connection-oriented*). O primeiro é usado para transferência de dados onde a integridade é mais importante que a latência (intervalo de tempo decorrido entre a transmissão de um pacote e a sua recepção). O segundo, por sua vez, é utilizado para transferência de dados cuja baixa latência (não podendo exceder a ordem de dezenas de milissegundos) é mais importante que a integridade, a fim de garantir interatividade, como por exemplo, na transferência de voz em tempo real. O *slot* pode ser reservado para transmissão de pacotes síncronos. O Bluetooth pode suportar as seguintes configurações: 1 (um) enlace de tráfego ACL e até 3 (três) enlaces de tráfego SCO, ou 1 (um) enlace suportando simultaneamente tráfegos SCO e ACL. Cada enlace SCO de voz suporta tráfego simétrico de 64 Kbps. Cada enlace ACL pode suportar um tráfego assimétrico de no máximo 723,2 Kbps no *downlink* (canal direto) e 57,6 Kbps no *uplink* (canal reverso), ou 432,9 Kbps em um canal simétrico.

Um sistema Bluetooth completo possui os seguintes componentes:

- Uma etapa de rádio para enviar e receber dados;
- Um módulo contendo um microprocessador para implementar as tarefas da banda base;
- Memória compatível com um sistema mínimo;
- Uma interface para uma entidade controladora.

O sistema descrito acima se refere à situação em que o módulo Bluetooth é independente da entidade controladora. Comercialmente há módulos que já contêm a etapa controladora de maneira a implementar alguma função específica, constituindo assim um sistema embarcado.

A etapa de rádio pode ser implementada em um módulo ou em um único circuito integrado.

Em um sistema independente, os protocolos de camada mais baixa são suportados pelo módulo Bluetooth. E os protocolos de camada mais alta que estão ligados ao tipo de aplicação e compõem os perfis de uso do Bluetooth, são suportados pela entidade controladora. Estes perfis estão classificados de acordo com a aplicação, a saber: transferência de arquivo, ponto de acesso de rede e outros.

As redes Bluetooth provêm conexões ponto-a-ponto, onde somente duas unidades Bluetooth estão envolvidas, ou ponto-a-multiponto. As figuras 1a e 1b ilustram o exposto. Na conexão ponto-a-multiponto, o canal é compartilhado por várias unidades Bluetooth. Duas ou mais unidades compartilhando o mesmo canal formam uma *piconet*.

Por definição, a unidade que inicia o procedimento de estabelecimento de conexão é denominada mestre e, conseqüentemente, a outra é denominada escrava. Dentro de uma *piconet* somente uma unidade pode atuar como mestre. As demais atuam como escravas. Até sete escravos podem estar ativos na *piconet*. Entretanto, mais escravos podem permanecer sincronizados com o mestre, no chamado modo *park* (modo de operação de baixa potência). Dentro da *piconet* o mestre exerce um controle centralizado, somente é possível haver comunicação entre um mestre e um ou mais escravos. Em outras palavras, o escalonamento dos escravos é realizado pela unidade mestre.

Várias *piconets* podem coexistir numa mesma área de cobertura, formando uma topologia denominada de *scatternet*. Nesta topologia, devido ao fato de somente haver um mestre por *piconet*, uma unidade que é mestra em uma, obrigatoriamente deve ser escrava em outra. Porém, é permitido que uma unidade seja escrava em várias *piconets* em um esquema de multiplexação por divisão no tempo. A figura 1c ilustra a formação das *scatternets*. Outra característica é que as *piconets* são identificadas por um código denominado código de acesso de canal. Este código é derivado da identificação e do relógio nativo do dispositivo mestre. Senso assim, todos os pacotes enviados dentro da *piconet* possuem o referido código de acesso de canal. A unidade receptora somente aceitará um pacote se o código de acesso estiver de acordo com o código de acesso do mestre da *piconet* a que ela pertencer.

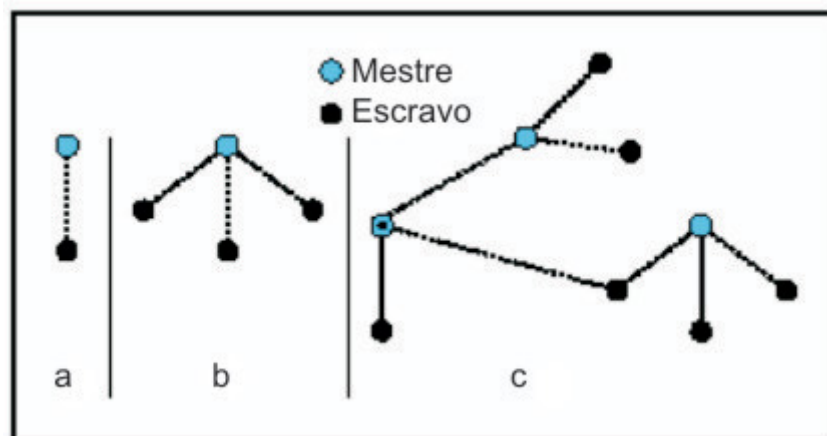


Figura 1. Topologias de redes Bluetooth

2.1.1 A pilha de protocolos

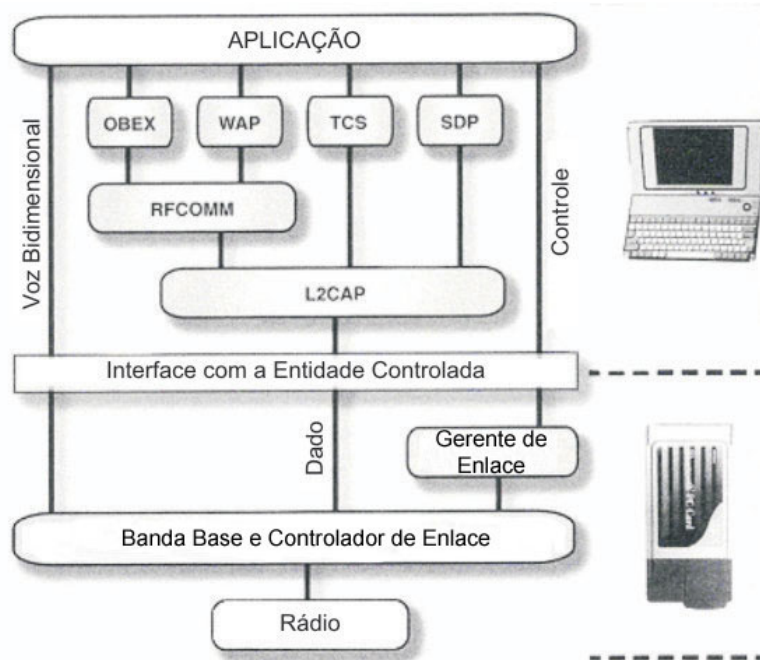


Figura 2. A pilha de protocolos

Descrição da figura 2:

- A etapa rádio, como mencionado anteriormente, completa a camada física provendo a transmissão e a recepção dos dados. Os pacotes de dados são montados e enviados para a etapa de rádio pela máquina de estado da banda base. O controlador de enlace (*link controller*) implementa as operações dos estados mais complexos como: *standby* (inicialização e espera), conectado, modos de baixa potência e demais funções da banda base. O gerente de enlace (*link manager*) realiza o controle e a configuração do enlace por meio do LMP (*link manager protocol*);
- O L2CAP (*Logical Link Control and Adaptation Protocol*) faz interface com os protocolos de nível superior, estabelece enlaces virtuais entre os dispositivos, podendo estabelecer vários enlaces simultaneamente, e realiza a segmentação e remontagem de pacotes. Utiliza o conceito de canais lógicos para transferência de diferentes tipos de dados.
- O RFCOMM (*Radio Frequency Communication*) é um emulador de porta serial. Este protocolo faz com que a aplicação pense que está se comunicando através de uma porta serial com cabo e não por uma interface de rádio.
- Os demais protocolos são voltados para o tipo de aplicação ao qual o dispositivo Bluetooth estará sendo usado;

- Voz bidirecional (*Two Way Voice*) é uma provisão para as aplicações que requerem a transmissão de áudio digital bidirecional. A aplicação acessa diretamente a banda base para minimizar a latência.

2.1.2 Os perfis de uso

O Bluetooth SIG (*Special Interest Group*) tem identificado vários modelos de uso, onde cada um é acompanhado pelo seu perfil. Estes perfis definem os protocolos e características que um particular modelo de uso suporta. Em linhas gerais os perfis Bluetooth são responsáveis por:

- Especificar qual a pilha de protocolos a ser usada para uma determinada aplicação. Exemplo: substituição de cabos;
- Especificar os requisitos mínimos de cada camada da pilha de protocolos de acordo aplicação.

A figura 3 ilustra como os perfis interagem. O perfil de acesso genérico (GAP) está contido em todos os perfis, pois o mesmo especifica os requisitos e as recomendações necessárias para realizarem-se os procedimentos mais básicos relacionados com o acesso ao meio.

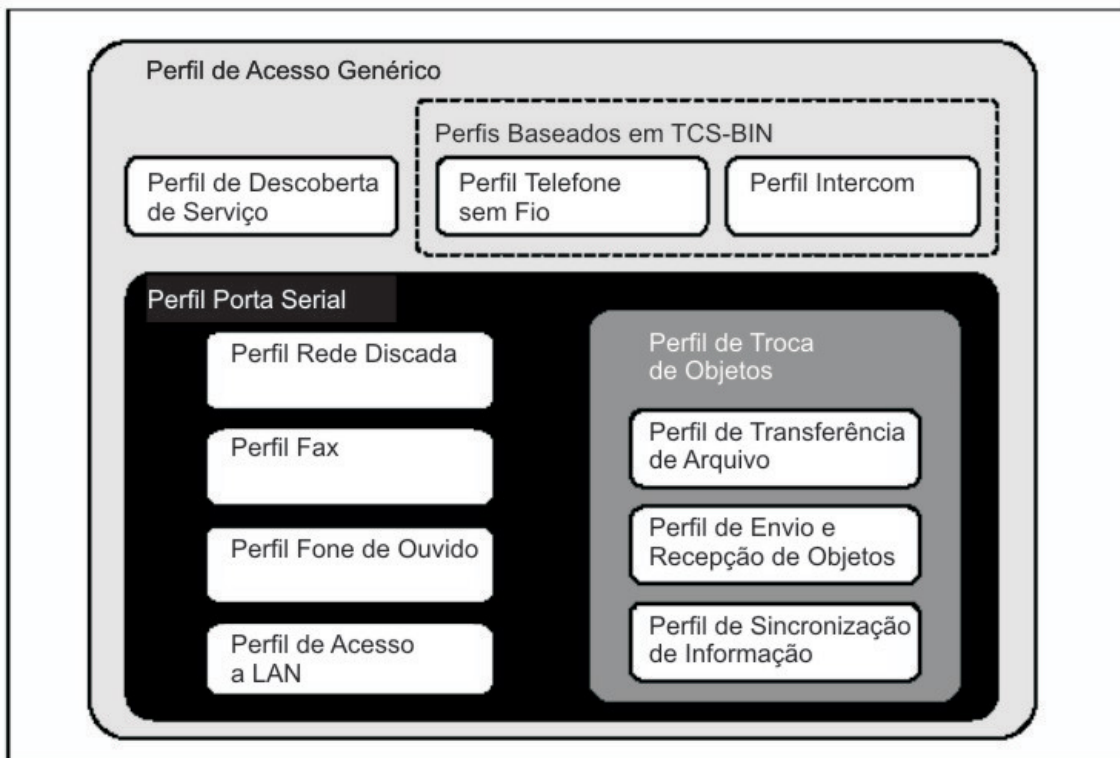


Figura 3. Os perfis de uso

2.2 Sistema de rádio Bluetooth

2.2.1 Introdução

O rádio é o dispositivo no sistema de comunicação Bluetooth, responsável por subsidiar a comunicação sem fio. É a camada mais baixa da pilha de protocolos mostrada na figura 2.

Neste tópico serão investigados assuntos importantes a fim de caracterizar a performance deste dispositivo, a saber:

- O esquema de modulação;
- A capacidade do canal;
- A imunidade à interferência;
- FHSS (*Frequency Hopping Spread Spectrum*);
- A performance do transmissor;
- A performance do receptor.

2.2.2 O esquema de modulação

Para receber a qualificação, o sistema de rádio Bluetooth tem que possuir as seguintes características técnicas:

- Modulação GFSK (Gaussian Frequency Shift Keying) com BT (largura de banda versus tempo de duração do símbolo) igual a 0.5. Este esquema de modulação implementa, por meio de um filtro passa - baixa com curva de resposta Gaussiana dada por $H(f) = \exp\left(-\frac{1.4f^2}{B^2}\right)$ (1), uma suavização dos dados da banda base nas transições de nível 1 para nível 0 e vice-versa, reduzindo a largura de banda, de modo a permitir atender aos requisitos de largura de banda da norma (1 MHz). O preço desta suavização é o aumento da interferência intersimbólica (ISI), que é a interferência de um símbolo em seus símbolos vizinhos. A ISI aumenta com a diminuição da largura de banda B, e com o aumento da taxa de transmissão de símbolo para uma dada largura de banda fixa. Por este fato, normalmente as características de um filtro Gaussiano são dadas pelo produto BT.
- Taxa de transmissão de símbolo de 1 Ms/s, correspondendo a uma taxa de transmissão de dados de 1 Mbps;

- Índice de modulação (k) entre 0.28 e 0.35. Correspondendo a um desvio de frequência (f_d) entre 140 KHz e 175 KHz, respectivamente.
- O desvio de frequência positivo deve corresponder à lógica 1 (um), e o desvio de frequência negativo deve corresponder à lógica 0 (zero);
- Regulação de tempo de símbolo melhor que +/- 20 ppm;
- Erro de passagem pelo nível zero menor ou igual a um oitavo do período de um símbolo;
- O desvio de frequência (f_d) correspondente a uma seqüência de 1 (um) para 0 (zero), 1010, deve ser no máximo 80% do f_d de uma seqüência 00001111. Este requisito tem por objetivo garantir uma razoável ISI, causada por uma seqüência de transições de um (1) para zero (0) e vice-versa;
- O desvio de frequência (f_d) deve ser no mínimo 115 KHz.

2.2.3 A capacidade do canal

Segundo SHANNON [2], a capacidade de um canal, dada em bits por segundo, é limitada pela largura de banda (W) e pela relação sinal ruído (S/N), segundo a equação:

$$C = W \log_2 \left(1 + \frac{S}{N} \right), \quad (2)$$

onde W é dado em hertz

A interpretação desta fórmula é a seguinte:

Se um dado é enviado através de um canal com uma taxa de transmissão menor que a capacidade do mesmo, existe um método de transmissão de dados onde a taxa de erro de bit (BER) no receptor é tão pequena quanto o desejado. Sendo que o método de transmissão mencionado, nada mais é do que a aplicação de técnicas de modulação e de códigos de detecção e correção de erro, a fim de obter a taxa de erro por bit almejada.

Pela equação (2) acima, vê-se que a capacidade do canal é maior quanto menor for a potência do ruído N. Sendo assim é possível investigar a taxa de transmissão **máxima** do Bluetooth, assumindo que o ruído branco aditivo (AWGN) é o único impedimento existente no canal.

Segundo [3], a potência **mínima** do ruído é dada por:

$$N = kTB, \quad (3)$$

onde:

k é a constante de Boltzmann's, igual a 1.37×10^{-23} J/°K;

T é a temperatura em ° K;

B é a largura de banda de interesse.

Sendo assim, para B igual a 1 MHz, T igual à temperatura ambiente (293 °K). Tem-se N igual a 4×10^{-15} W. Considerando uma potência do sinal de -80 dBm no receptor, que corresponde a 10^{-11} W, tem-se $S/N = 0.25 \times 10^4$ W (33,9 dB). Logo, pela equação (2), tem-se uma capacidade **máxima** do canal de 11.28 Mb/s. Esta capacidade de canal está bem acima da taxa de 1 Mbps especificada pelo padrão Bluetooth. Isto se deve a dois fatores:

- O aumento da taxa de transmissão implica em aumento do custo da etapa de rádio, pois envolve a aplicação de complexos métodos de modulação e codificação. E isto vem contra um dos principais requisitos do Bluetooth, o baixo custo. Entretanto, dentro do SIG, há um grupo de trabalho denominado Radio2, que está realizando pesquisas visando esse aumento da taxa de transmissão de bits;
- Em sistemas de comunicação sem fio, a interferência causada por outros transmissores intencionais que operam dentro e fora da banda ISM é bem mais opressora que o ruído AWGN considerado nos cálculos acima.

2.2.4 A imunidade à interferência

Como a banda ISM é aberta para ser utilizada por qualquer dispositivo que satisfaça os requisitos do órgão regulador do país, em muitos sistemas de comunicação sem fio, a interferência é bem mais determinante para a deterioração da performance do sistema que o ruído branco aditivo. Sendo assim, a imunidade à interferência é um requisito de suma importância para os sistemas de comunicação sem fio.

As principais fontes de interferência para um dispositivo Bluetooth são:

- Forno de microondas (apesar de não seguirem os requisitos de potência e espalhamento de espectro, operam nesta banda);
- Redes 802.11 b/g;
- Redes Home RF;
- Telefones sem fio;
- Dispositivos customizados;
- Usuários não licenciados, como operadores de radio amador;
- Outras *piconets* adjacentes.

A figura 4 ilustra o comportamento de uma *piconet* coexistindo com uma rede Wi-Fi em três cenários: primeiro sem a presença de interferência, depois quando a fonte de interferência é colocada a 10 m da *piconet*, e finalmente, quando a fonte de interferência está co-locada com a *piconet*. Note ainda que, à medida que a distância entre o transmissor e o receptor da *piconet* aumenta, a vazão cai ainda mais devido ao enfraquecimento do sinal desejado na antena do receptor, dificultando a sua demodulação.

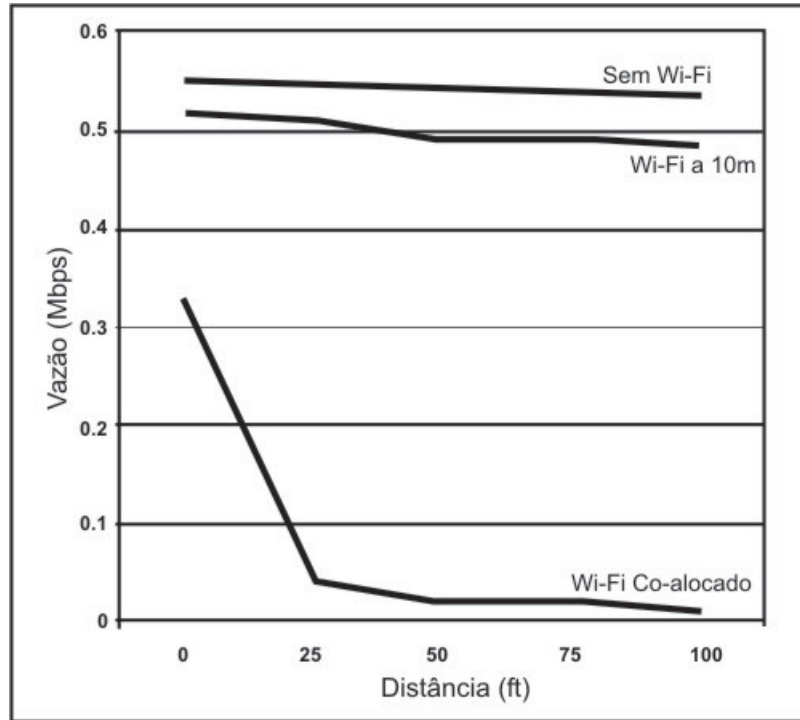


Figura 4. Bluetooth coexistindo com Wi-Fi

A imunidade à interferência pode ser obtida de duas maneiras: evitando ou suprimindo a fonte de causadora. Como a banda de 2.4 GHz possui uma largura de aproximadamente 80 MHz e a maioria dos sistemas de rádio que nela operam possuem banda limitada, há grande probabilidade de haver, dentro do espectro de frequência, faixas do mesmo onde não há, ou há em menor intensidade, a incidência de interferência. Sendo assim, torna-se mais apropriada a técnica de evitar a interferência no domínio da frequência. O Bluetooth realiza isto por meio de um esquema de múltiplo acesso chamado de espalhamento de espectro com salto de frequência (FHSS - *frequency hopping spread spectrum*).

2.2.5 FHSS

Um diagrama em blocos do sistema de comunicação Bluetooth FHSS está mostrado na figura 5. A banda base é modulada por meio de uma modulação GFSK e transmitida usando uma portadora determinada pelo sintetizador de frequência. Ao invés de produzir uma única portadora, o sintetizador é controlado por um gerador de códigos de salto que força a troca de frequência da portadora à taxa de 1600 saltos por segundo. A cada salto um pacote de dados é enviado.

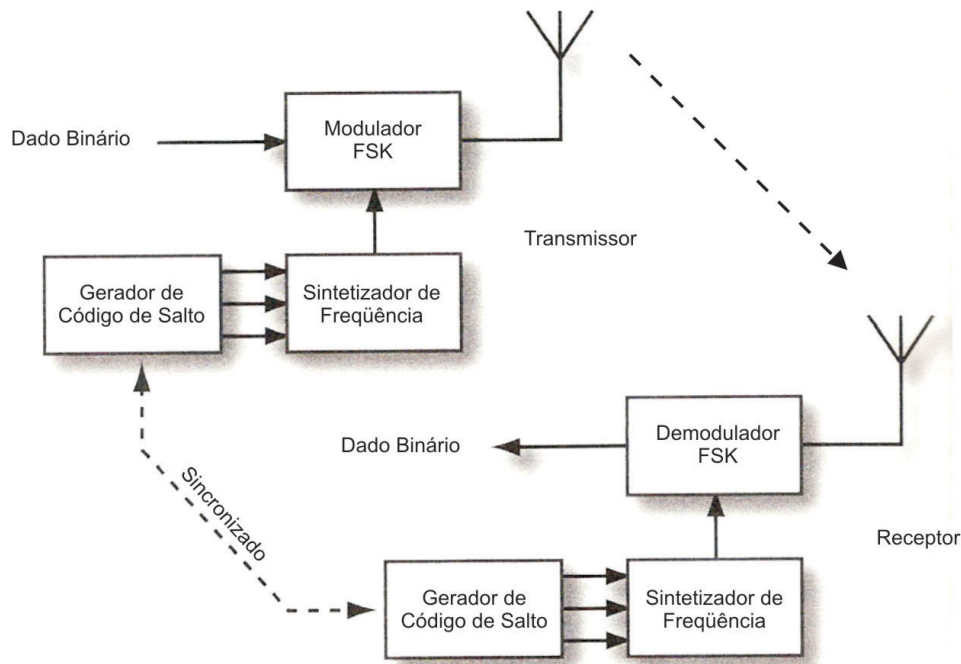


Figura 5. Sistema de comunicação FHSS

Este padrão de salto parece ser aleatório, mas é criado por um algoritmo pseudo-aleatório dentro do bloco gerador de códigos de salto. O gerador é duplicado no receptor e, assim, é criado o mesmo padrão de salto de frequência usado no transmissor. Ademais, quando em comunicação, o transmissor e o receptor devem saltar de maneira síncrona de canal para canal. Sendo assim, se um determinado canal contém um alto nível de interferência, é bem provável que a transmissão de um pacote possa, no máximo, ser interrompida por um salto (625 us), já que os mesmos mudarão para um outro canal, no próximo salto, evitando a interferência.

Este sincronismo entre transmissor e receptor pode ser sumarizado da seguinte forma:

- Uso do mesmo conjunto de canais;
- Uso da mesma seqüência de saltos dentro do conjunto de canais;
- Sincronização no tempo dentro da seqüência de saltos;

- Garantia do seguinte cenário: um transmite enquanto o outro recebe e vice-versa.

Em uma comunicação Bluetooth, todos estes requisitos devem ser determinados pelo mestre da *piconet*. Por definição, a unidade que inicia o procedimento de estabelecimento de conexão é denominada mestre e, conseqüentemente, a outra é denominada escrava. O mestre passa os parâmetros da sincronização do FHSS para o escravo, durante o procedimento de *paging* (este procedimento será descrito posteriormente).

Um sistema de rádio FHSS é programado para operar dentro de certo conjunto de freqüências, chamado de conjunto de canais. Para o Bluetooth, o conjunto de canais corresponde às freqüências das portadoras descritas pela fórmula (4) abaixo.

$$f_c = 2.402 + \kappa \text{ MHz}; \kappa = 0, 1, 2, \dots, 78 \quad (4)$$

Sendo assim, há 79 possíveis freqüências no conjunto de canais, cada separada em fragmentos de 1 MHz, cobrindo a faixa de 2402 MHz a 2480 MHz. A figura 6 expõe como a *piconet* salta pseudo - aleatoriamente dentro destes canais. Como mencionado, cada canal tem uma largura de banda de 1 MHz, destinado à transmissão de dados, após modulação GFSK.

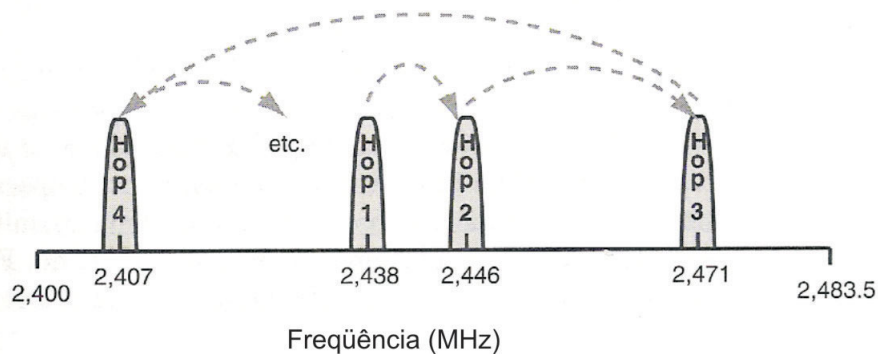


Figura 6. Salto entre canais - Domínio da Freqüência

No domínio do tempo, cada canal possui a largura de 625 μ s, como mostrado na figura 7. Estes canais são multiplexados no tempo por meio de um esquema *fast TDD* (*Time Division Duplex*), permitindo a realização de comunicação *full duplex* forçada, usando um único transceptor, pois a transmissão e a recepção ocorrem em momentos distintos.

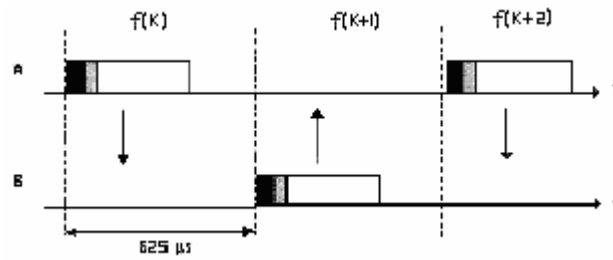


Figura 7. Salto entre canais – Domínio do tempo

Como comentado anteriormente, a seqüência de saltos é fornecida pelo gerador pseudo-aleatório de saltos. Esta seqüência é periódica, repetindo-se, então, após certo número de iterações. Este período deve ser no mínimo igual ao número de canais de salto. No Bluetooth, esta seqüência possui 2^{27} combinações. Como a freqüência de saltos é de 1600 saltos por segundo, conclui-se que o período da seqüência é de 23.3 horas. Sendo assim, após 23.3 horas a seqüência será repetida.

O período do gerador é longo para diminuir a probabilidade de que duas *piconets* que coexistam na mesma área transmitam no mesmo canal. O que acarretará em colisão entre os pacotes enviados por elas, e a provável perda dos mesmos. A probabilidade de perda de pacotes considerando um cenário de K *piconets* coexistindo na mesma área, com uma quantidade de canais de salto M , pode ser expressa pela fórmula abaixo [3].

$$P_E = 1 - \left(1 - \frac{1}{M}\right)^{K-1} \quad (5)$$

A figura 8 mostra o comportamento da probabilidade de erro de pacotes em função do número de *piconets* coexistindo na mesma área. O valor de M foi considerado 79.

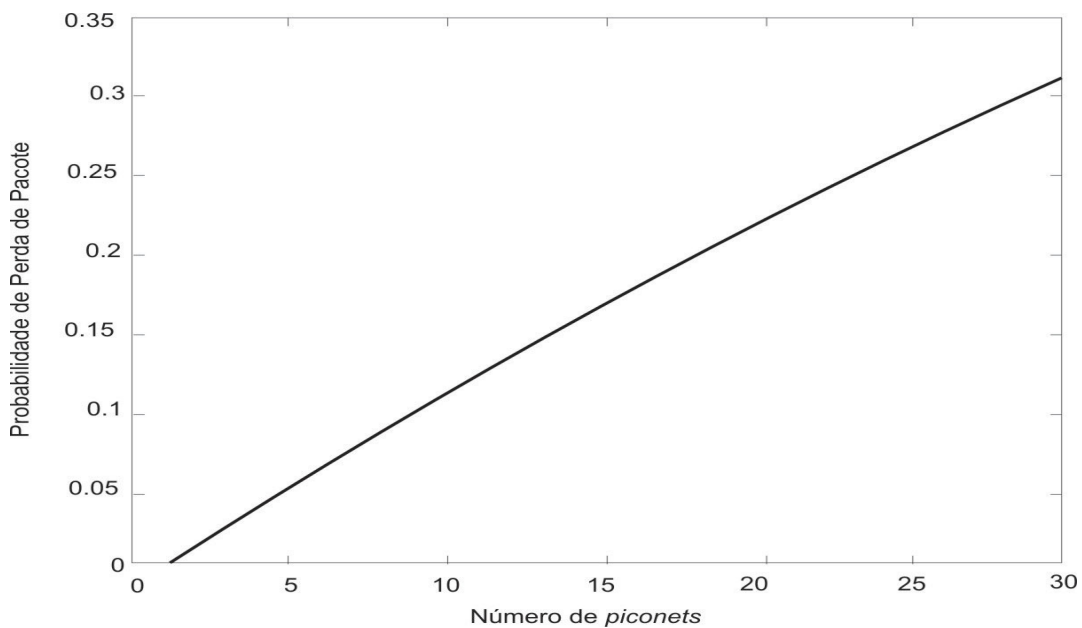


Figura 8. Probabilidade de perda de pacotes e x número de *piconets* co-allocadas

2.2.6 Performance do transmissor

O transmissor Bluetooth é descrito como um emissor intencional de sinais eletromagnéticos dentro do canal ocupado na banda ISM. Porém, também funciona como um emissor não intencional de sinais espúrios para as demais faixas de frequência (dentro e fora da banda ISM). Essas emissões espúrias são alvas de regulação por parte das agências governamentais. Sua especificação é dada em função das suas características no domínio do tempo e da frequência. Dentro desse contexto, a especificação do Bluetooth versão 1.1, descreve o transmissor sob a seguinte ótica: máxima potência de saída, máxima emissão de sinais espúrios dentro e fora da banda, máxima variação da frequência da portadora ao acordar após um salto e durante a transmissão de um pacote.

2.2.6.1 Classes de potência dos transmissores

Os transmissores Bluetooth são classificados em três classes (ver tabela 1), de acordo com a potência máxima de saída, a saber:

Classe	Potência de Saída
1	100 mW (20 dBm)
2	2.5 mW (4 dBm)
3	1 mW (0 dBm)

Tabela 1. Classes de potência

Obrigatoriamente, os transmissores classe 1 têm que implementar um ajuste do nível de potência de saída (controle de potência), a fim de reduzir a potência a um nível adequado para comunicação e minimizar a interferência causada em outros usuários da banda. Já para os dispositivos classe 2 e classe 3, esta característica é opcional.

2.2.6.1.1 Classe 2

Os dispositivos classe 2 podem transmitir sinais eletromagnéticos com as seguintes características de potência de saída (ver tabela 2).

Potência de saída	Valor
Máxima	4 dBm
Nominal	0 dBm
Mínima	-6 dBm

Tabela 2. Potência de saída - dispositivos classe 2

2.2.6.1.2 Classe 1

Conforme mencionado, a máxima potência de saída do transmissor é de 20 dBm. O controle de potência é obrigatório a partir de 4 dBm, ou menor se desejado. A potência mínima de saída sugerida é de -30 dBm ou 1 mW (os níveis abaixo de 4 dBm não são obrigatórios).

O controle de potência é implementado usando um mecanismo de realimentação entre o mestre e o escravo da *piconet*. Neste mecanismo o escravo mede a potência do sinal recebida e avisa para o mestre a necessidade de aumento ou diminuição da potência do seu transmissor. Este mecanismo é realizado pelo LMP (*Link Manager Protocol*).

Um transmissor que efetua o controle de potência deve ter as seguintes características:

- Capacidade de ajustar os níveis de potência de saída em passos dentro da faixa de 2 dB a 8 dB;
- Realizar o ajuste de potência dentro da faixa de 4 dBm (ou menor, se desejado) a 20 dBm (potência de saída máxima).

De acordo com a sensibilidade do receptor, pode-se definir um parâmetro muito importante para o controle de potência, o chamado *Golden Receive Power Range*. Ele define uma faixa de potência do sinal de entrada, no qual o controle de potência objetiva colocá-lo. A região mínima está entre o limite máximo de -56 dBm e o limite mínimo que está 6 dB acima da sensibilidade do receptor. Já a região máxima está a 20 ± 6 dB acima do limite mínimo da região mínima. A figura 9 ilustra o exposto.

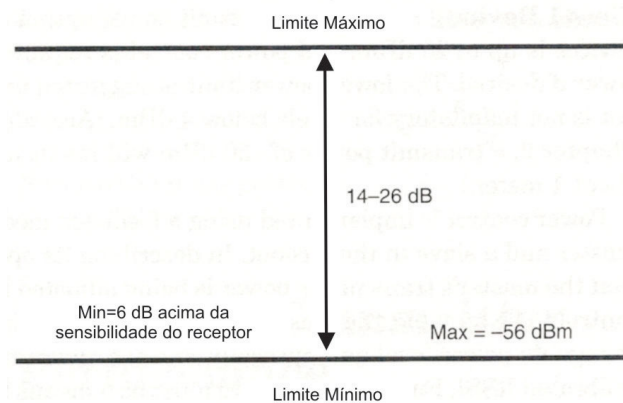


Figura 9. Golden Receive Power Range

Sendo assim, se a sensibilidade do receptor for -70 dBm, característica de um receptor padrão, a região mínima estará entre -64 dBm e -56 dBm. O nível máximo, por sua vez, estará a 20 ± 6 dB acima do limite mínimo.

Para exemplificar como acontece o controle de potência, consideremos um escravo cujo receptor possui um *Golden receive Power Range* entre -64 a -56 dBm, e um mestre capaz de ajustar sua potência de saída em passos de 8 dB. Suponhamos que o mestre envie um pacote e que o escravo o receba com o nível de potência de -70 dBm, medido por meio do seu RSSI. (*Receive Signal Strength Indicator*). Então, o escravo montará um pacote LMP solicitando o incremento da potência de saída do mestre em 1 (um) passo. Considerando que não houve mudanças na propagação do sinal, o escravo agora receberá o sinal com o nível de -62 dBm, que estará dentro da faixa aceitável. Este processo de controle de potência pode também compensar as perdas devido à propagação do sinal decorrentes da mobilidade ou mudança dos obstáculos.

Caso um dispositivo classe 1 descubra, por meio do gerente de enlace (*link manager*), a incapacidade de realização do controle de potência no dispositivo do outro lado do canal, ele naturalmente reduz sua potência de saída para os níveis correspondentes à classe 2 ou 3, conforme a necessidade.

Há também alguns inconvenientes no uso de unidades classe 1, a saber [3] :

- Em ambientes considerados com nível de interferência moderado, para se aumentar o alcance de um fator de 4, é necessário aumentar a potência segundo um fator de 100;
- Para se consumir o ganho de alcance, ambos os transmissores no canal tem que ser classe 1;
- O potencial de interferência causada em outros usuários vizinhos aumenta rapidamente com o aumento da potência;
- O transmissor se torna mais complexo e mais caro;
- A vida útil da bateria, em dispositivos portáteis, pode ser reduzida significativamente;
- Níveis de potência maiores que alguns miliwatts, já podem causar interferência no funcionamento de circuitos digitais e analógicos dos dispositivos vizinhos.

Comumente, os transmissores classe 1 são utilizados em dispositivos clientes, como *laptops* e *desktops*, e servidores, como impressoras e pontos de acesso.

2.2.6.2 Emissões espúrias

Como comentado anteriormente, as emissões espúrias podem ser analisadas sob dois aspectos: as emitidas dentro e as emitidas fora da banda ISM.

Para os canais adjacentes, estas emissões são avaliadas da seguinte forma: cada segmento de banda de 100 KHz deve ter um nível de potência de -20 dBc (decibéis abaixo da portadora), em relação a um segmento de banda de 100 KHz localizado dentro da banda do transmissor, que possui o nível máximo de potência na sua saída quando está modulado por com um dado pseudo-aleatório.

Em valores absolutos, a emissão total de sinais espúrios em canais adjacentes deve ser menor que -20 dBm, para canais localizados a 2 MHz do canal do transmissor em teste, e menor que -40 dBm para canais que estão a 3 MHz ou mais. Uma exceção é feita quando há acima de 3 (três) canais localizados a 3 MHz de distância. Nestes casos, a potência pode chegar até -20 dBm em cada canal.

Para as emissões espúrias fora da banda, as medições são feitas também usando segmentos de 100 KHz de banda, porém dentro de uma ampla faixa de frequências. O resultado destas medições não pode exceder os valores apresentados na tabela 3.

Banda de Frequência	Modo de Operação	Modo Inativo
30MHz – 1 GHz	-36 dBm	-57 dBm
1GHz-12.75 GHz	-30 dBm	-47 dBm
1.8 GHz – 1.9 GHz	-47 dBm	-47 dBm
5.15 GHz – 5.3GHz	-47 dBm	-47 dBm

Tabela 3. Emissões espúrias fora da banda

2.2.6.3 Variação de frequência da portadora (*drift*)

Quando o transmissor acorda após saltar para uma nova frequência, a sua precisão deve ser +/- 75 KHz da frequência central (f_c) do canal. Quando em comunicação, ao transmitir um pacote, a frequência do transmissor também pode variar. Esta variação pode ser tão grande quanto 400 Hz/ μ s, porém a variação total não pode exceder os limites estabelecidos na tabela 4.

Tipo de pacote	Variação de frequência
1 slot	$\pm 25KHz$
3 slots	$\pm 40KHz$
5 slots	$\pm 47KHz$

Tabela 4. Variação de frequência da portadora

2.2.7 Performance do receptor

Em geral, os receptores devem ser capazes de recuperar sinais cujo nível de potência é extremamente baixo - da ordem de picowatts, e rejeitar centenas ou milhares de sinais não desejados, que muitas vezes possuem nível de potência bem maior que o primeiro.

Um dos melhores índices de medida da performance de um sistema de comunicação digital, é a taxa de erro por bit (BER) do receptor. Sendo assim, no projeto de um receptor, considerando um ambiente de trabalho extremamente propenso a interferências, o objetivo a ser alcançado é a obtenção de uma taxa de erro por bit tão pequena quanto possível. Naturalmente, dentro da premissa do atendimento das limitações da capacidade do canal, expostas nas seções anteriores.

2.2.7.1 Sensibilidade

A sensibilidade do receptor é uma quantidade que nos diz o quão baixo pode ser a potência do sinal em sua entrada e ainda ser possível haver a demodulação do mesmo, considerando uma determinada taxa de erro de bit. A especificação do Bluetooth em sua versão 1.1, determina que a sensibilidade do receptor deva ser de no mínimo -70 dBm, para uma BER de 0,1 % ou 10^{-3} . A interpretação para esta especificação é a seguinte: dado um transmissor classe 1, cuja potência de saída máxima é de 20 dBm, e um receptor padrão (sensibilidade igual a -70 dBm) a perda no caminho pode ser de até 90 dB, $20 - (-70)$, que o sinal enviado ainda deve ser recuperado com um BER de no máximo 10^{-3} .

Receptores considerados no estado da arte possuem um nível de sensibilidade de - 100 dBm ou melhor. Comparativamente, o receptor Bluetooth está bastante aquém destes. Porém, isto é justificável pela premissa inicial do Bluetooth: ser um canal de comunicação de baixo custo.

2.2.7.2 Imunidade a interferência

Já se falou anteriormente que a imunidade à interferência é determinante para o desempenho do sistema de comunicação digital sem fio. Especificamente para o receptor, ela interfere na capacidade de seleção entre o sinal desejado (C) e o não desejado (I). Neste contexto, a especificação Bluetooth, analisa o desempenho do receptor segundo três aspectos: a relação C/I, *blocking* e rejeição à imagem.

Para testes de avaliação de desempenho, a especificação do Bluetooth, descreve as características do sinal a ser emitido como fonte de interferência, a saber: uma portadora

não modulada, também chamada de *continuous wave* (CW), ou um sinal modulado padrão, possuindo as seguintes características:

- Modulação GFSK com o índice de modulação igual a $0.32 \pm 1\%$ e o produto BT igual a $0.5 \pm 1\%$;
- Taxa de transmissão igual a $1 \text{ Mb/s} \pm 1 \text{ ppm}$;
- O sinal modulante deve ser uma seqüência binária pseudo-aleatória denominada PRBS9;
- Precisão na frequência melhor que $\pm 1 \text{ ppm}$.

2.2.7.3 Relação C/I

É definida como a relação entre a potência do sinal desejado (C) e a interferência (I), no qual o receptor deve estar habilitado para tolerar sem exceder a taxa de erro de bit (BER) de 10^{-3} . A relação C/I é dada em função da proximidade, no domínio da frequência, do sinal I em relação ao sinal C (mesmo canal, canal adjacente a 1 MHz , canal adjacente a 2 MHz e canal adjacente $\geq 3 \text{ MHz}$). A sumarização destes valores é dada na tabela 5.

Requerimentos	Valor
C/I – mesmo canal	11dB
C/I - Canal adjacente (1MHz)	0dB
C/I - Canal adjacente (2MHz)	-30dB
C/I - Canal adjacente ($\geq 3 \text{ MHz}$)	-40 dB

Tabela 5. Relação C/I

2.2.7.4 Blocking

Como falado anteriormente, os receptores devem ser capazes de detectar sinais extremamente fracos. Para isso, o circuito de entrada destes receptores possui amplificadores de alto ganho. Em contrapartida, há situações em que o nível de potência na antena do receptor já é forte o suficiente para a detecção. Conseqüentemente, os amplificadores de entrada serão facilmente levados à saturação, e o sinal desejado não poderá ser detectado corretamente. Este efeito recebe o nome de *blocking*.

Este cenário pode ser estabelecido caso a portadora do sinal I seja forte o suficiente para causar saturação do circuito de amplificação do estágio de entrada do receptor. Dentro deste contexto, a tabela 6 sumariza os valores de nível de potência dos sinais I, fora da

banda, considerando que o sinal desejado na antena do receptor tem a potência de -67 dBm .

Frequência da portadora	Potência
30 – 2000 MHz	-10 dBm
2000 – 2399 MHz	-27 dBm
2489 – 3000 MHz	-27 dBm
3000 – 12750 MHz	-10 dBm

Tabela 6. Nível máximo de potência permissível fora da banda

2.2.7.5 Rejeição a imagem

Os receptores são projetados para converter os sinais de RF em uma frequência denominada FI (frequência intermediária). Esta frequência é fixa e bem mais baixa que ao sinal de RF, possibilitando o uso de filtros e amplificadores padronizados e, por conseguinte, mais baratos. Estes receptores recebem o nome de receptores superheterodinos (mesmo processo utilizado em receptores de TV analógica). O processo de transformação do sinal de RF para a FI está descrito na figura 10.

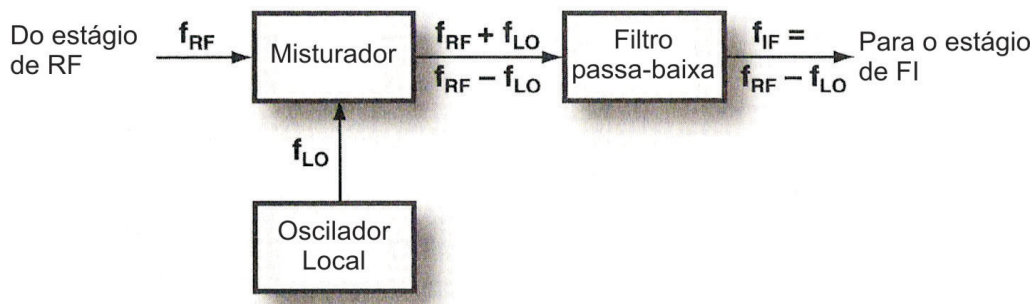


Figura 10. Processo de obtenção da FI

Onde f_{RF} é a frequência do sinal desejado e f_{LO} é a frequência do oscilador local.

O misturador é responsável por multiplicar os sinais f_{RF} e f_{LO} . Quando estas duas senoides forem multiplicadas o resultado obtido serão duas novas frequências iguais a $f_{RF} - f_{LO}$ e $f_{RF} + f_{LO}$. Desta forma, ao passarem pelo último estágio, filtro passa - baixa, a frequência mais alta é eliminada, obtendo-se uma frequência intermediária f_{FI} , dada por $f_{FI} = f_{RF} - f_{LO}$.

Para que o receptor seja sintonizado em diferentes frequências f_{RF} , basta que a frequência do oscilador local f_{LO} seja devidamente alterada para que f_{FI} seja mantida constante.

Porém, além das frequências já mencionadas, há uma outra frequência gerada neste processo. Esta frequência é chamada de imagem de frequência f_{image} , tal que $f_{FI} = f_{LO} - f_{image}$. Isto significa que se houver um outro transmissor usando f_{image} como sua frequência da portadora, haverá interferência no sinal desejado C cuja frequência da portadora é f_{RF} . A figura 11 ilustra no domínio da frequência a relação entre f_{RF} , f_{LO} , f_{image} e f_{FI} .

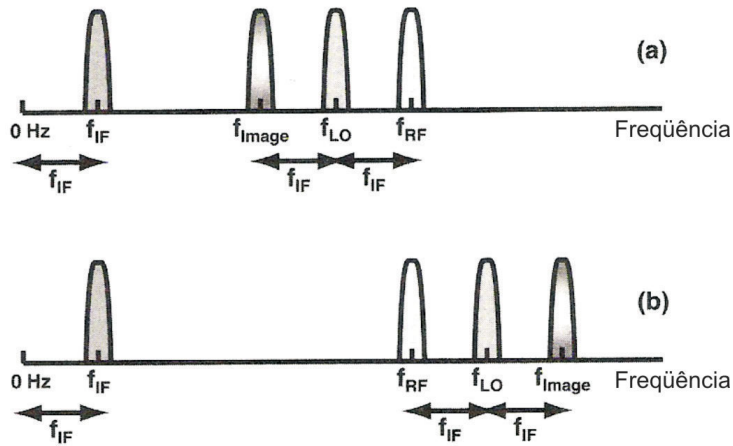


Figura 11. Rejeição a imagem

A especificação do Bluetooth limita o nível de interferência em relação à imagem de frequência através da relação C/I_{image} , da seguinte forma (considerando um nível do sinal C de -67 dBm na antena do receptor) :

- Para sinais de interferência (I) localizados na f_{image} , $C/I_{image} = -9dB$;
- Para sinais de interferência (I) localizados a $\pm 1MHz$ da f_{image} ,

$$C/I_{image \pm 1MHz} = -20dB$$

2.2.7.6 Distorção de intermodulação (IMD)

A não-linearidade do circuito de amplificação dos receptores produz em sua saída novas frequências, diferentes das encontradas em sua entrada. Caso uma destas novas frequências seja igual à frequência da portadora de um determinado sinal desejado (f_c), a mesma sofrerá interferência. Entretanto, o maior incômodo destas novas frequências é o fato delas poderem se combinar e, então, gerar uma terceira frequência. A relação entre elas é dada por [3]:

$$f_3 = 2f_1 - f_2, \quad (6)$$

onde f_1 e f_2 são as frequências primárias e f_3 é o produto de 3º ordem

Sendo assim, caso haja uma *piconet* cuja frequência de salto neste momento seja igual a f_3 , ela sofrerá interferência da chamada distorção de intermodulação de terceira ordem.

Para fins de avaliação de performance com relação à IMD de terceira ordem, a especificação do Bluetooth, descreve as seguintes condições de teste:

- A potência do sinal desejado f_c deve ser -64 dBm;
- O sinal interferente f_1 deve ser um CW (*continuous wave*), com potência de -39 dBm;
- O sinal interferente f_2 deve ser o sinal padrão descrito no item 2.2.7.2, com potência de -39 dBm.

Nestas condições a taxa de erro de bit não pode exceder 10^{-3} e a diferença entre f_1 e f_2 pode ser 3, 4 ou 5 MHz.

2.2.7.7 Emissões espúrias

Os receptores Bluetooth são constituídos internamente por osciladores, temporizadores e outros circuitos que podem produzir suas próprias emissões de RF. O resultado é que um receptor pode tentar detectar um sinal gerado por si próprio, e, assim sendo, pode transmitir sinais de maneira não intencional. O primeiro efeito é chamado de *spurious response frequencies*, e o segundo é chamado de emissão espúria.

Os valores limites para as emissões espúrias estão sumarizados na tabela 7, para dois segmentos de banda diferentes. Estes valores não podem ser excedidos em qualquer segmento de 100 KHz da banda.

Faixa de Frequência	Limite
30 MHz – 1GHz	-57 dBm
1 GHz – 12.75 GHz	-47 dBm

Tabela 7. Emissões espúrias

2.3 Estudo da propagação em ambientes fechados e cálculo do alcance do rádio Bluetooth

2.3.1 Mecanismos da propagação em ambientes fechados

Quando uma onda de rádio bate em uma superfície que não é uma isolante perfeita, nem uma condutora perfeita, que é a característica da maioria das superfícies reais, parte da

energia da onda passa através da superfície, parte é absorvida e parte é refletida. Assim sendo, um sinal de RF que viaja de um transmissor para um receptor, ao encontrar um obstáculo, pode ser fracionado em quatro caminhos distintos, a saber: caminho direto, caminho refletido, caminho difratado e caminho disperso. A figura 12 ilustra o exposto.

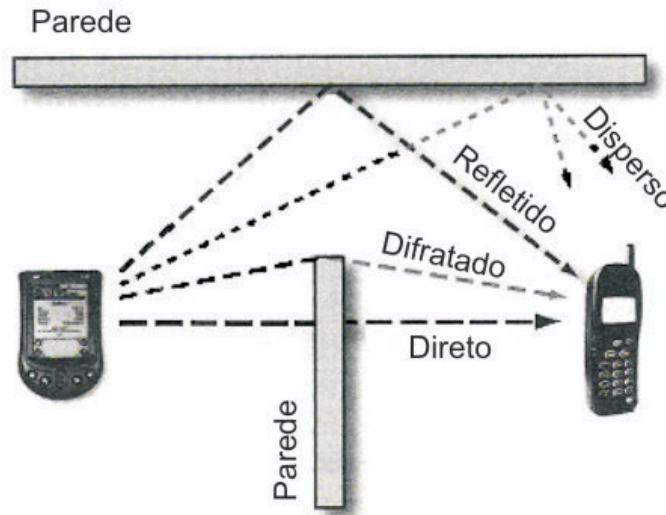


Figura 12. Propagação de uma onda de rádio na presença de obstáculos

O caminho direto pode ser classificado em linha de visão (LOS - *light-of-sight*), quando não há obstáculos entre o transmissor e o receptor, e linha de visão obstruída (OLOS - *obstructed light-of-sight*), quando há um ou mais obstáculos no referido caminho.

Freqüentemente uma grande quantidade de energia é refletida de alguns obstáculos, principalmente aqueles formados por materiais de boas propriedades condutivas, como o metal. Estas reflexões funcionam como cópias do sinal original, porém com diferentes características de atraso no tempo, deslocamento de fase e intensidade de potência. O estudo do comportamento destas reflexões é de suma importância para ditar a performance de um sistema de comunicação sem fio.

A difração do sinal ocorre quando uma onda de rádio encontra uma superfície pontiaguda. Estas são comumente encontradas nas aberturas em geral (portas e janelas). Ao bater nesta superfície o sinal sofre uma mudança de inclinação. Normalmente sua energia é bem inferior à das ondas do caminho direto e refletido.

A dispersão do sinal ocorre quando a onda de rádio bate em superfícies que possuem irregularidades de tamanho significativo, em relação ao comprimento de onda envolvido. A energia destes sinais decorrentes da dispersão é bastante inferior ao das ondas dos caminhos direto e refletido. A dispersão pode ser modelada como uma antena virtual, retransmitindo o sinal, porém com nível de potência bem menor que o original.

Quando uma antena receptora está localizada a certa distância da antena do transmissor, é intuitivo concluir que o sinal transmitido sofre atenuações ao percorrer este caminho. Estas atenuações são chamadas de perdas do caminho. A quantificação destas perdas está diretamente ligada ao cálculo do alcance da interface aérea utilizada. Daí, a necessidade de realizar-se um estudo da propagação do sinal de RF dentro da banda ISM centrada em 2.45 GHz, para posteriormente inferirmos a respeito do alcance da interface aérea Bluetooth.

A análise da propagação de sinal considerando os quatro caminhos supracitados juntos é bastante penosa. Como a maior parte da energia está dentro dos caminhos direto e refletido, é razoável estudar o comportamento da propagação segundo o comportamento dos mesmos.

As perdas calculadas considerando somente o caminho direto em função apenas das características do transmissor, do receptor e dos obstáculos no caminho entre o receptor e o transmissor são chamadas de Perdas em Larga Escala (*Large Scale Path Loss*). Estas perdas são assim denominadas, pois para o cálculo da intensidade do sinal recebido, ambos medido e calculado, são consideradas perdas ocorridas acima de uma distância de aproximadamente 10 (dez) comprimentos de onda, que para o Bluetooth corresponde a 1.22 metros.

Já as perdas calculadas considerando a mobilidade do transmissor, do receptor e / ou dos obstáculos, mesmo em distâncias que correspondem a frações do comprimento de onda, são chamadas de Perdas em Pequena Escala (*Small Scale Path Loss*). Estas consideram variações de amplitude, fase e frequência dos sinais que chegam ao receptor devido a esta mobilidade. Um estudo mais aprofundado caracterizando esta perda, bem como identificando formas de evitá-las será exposto posteriormente no tópico Multipercurso.

2.3.2 Perdas em larga escala

Para o cálculo da potência do sinal recebido dentro de ambientes fechados, parte-se da clássica equação de cálculo da provisão de canal em espaço livre, extensivamente usada em canais de satélite e aplicações terrestres com caminho direto LOS, e inserem-se modificações para portá-la para ambientes fechados e abertos com caminho direto OLOS. Esta equação é dada por (7):

$$P_r = P_{t(dBm)} + G_{t(dB)} - PL, \quad (7),$$

onde P_r é a potência do sinal recebido, P_t é a potência transmitida, G_t é o ganho da antena do transmissor e PL (*Path Loss*) é a perda ocorrida no caminho entre o transmissor e o receptor, que depende da distância e do tipo de ambiente em que o par transmissor - receptor está inserido. A equação (8) expõe a fórmula da PL.

$$PL = 20 \log\left(\frac{4\pi}{\lambda}\right) + 10n \log(d), \quad (8),$$

onde λ é o comprimento de onda igual a 0.122 m para 2.45 GHz, d é a distância em metros entre o receptor e o transmissor (d deve ser maior que λ para antenas de meio comprimento de onda [4]), e n é o chamado expoente PL, responsável por inserir no cálculo da perda do caminho as variações do sinal devido às características do ambiente. Substituindo-se o valor de λ na equação (8), obtém-se uma equação de PL simplificada para as frequências usadas pelo Bluetooth. Sendo assim,

$$PL = 40 + 10n \log(d) \quad (9)$$

Como mencionado, a quantidade n , contida nas duas equações antecedentes, pode ser usada para levar em consideração no cálculo de PL, as características físicas do ambiente. Para exemplificar o uso desta equação, consideramos uma situação de propagação em espaço livre. Para esta situação n é igual a 2. Com isso, a equação se reduz a $PL = 40 + 20 \log(d)$, e a perda no caminho exhibe agora somente uma única dependência que é a distância d . Examinando esta equação para diferentes valores de d , vê-se que PL aumenta 6dB por oitava de distância (dobrando o valor de d) e 20 dB por década de distância (multiplicando d por 10). A figura 13 ilustra o exposto.

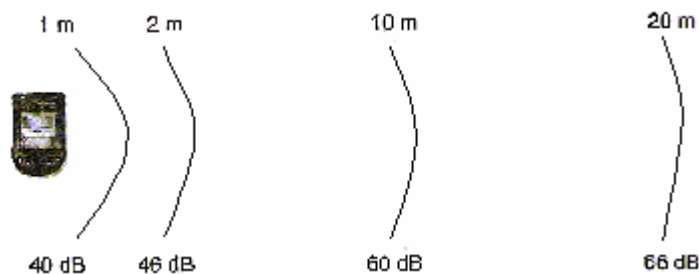


Figura 13. Comportamento da PL em espaço livre

Sendo assim, o grande desafio do cálculo simplificado de PL, é saber qual o valor de n adequado. Para ambientes fechados com caminho direto obstruído, é intuitivo que o

valor de n deva ser maior que 2 (dois) para compensar a inserção dos obstáculos. A tabela 8 sumariza os valores típicos de n para diversos ambientes, considerando que o par transmissor - receptor está no mesmo pavimento, para frequências dentro da faixa de 900 a 1500 MHz [5]. Em 2.4 GHz, estes valores são aproximadamente os mesmos para alguns prédios, e para outros, os valores tendem na média a ser 10% mais altos [6] e [7]. Na tabela 8 está exposto também o desvio padrão para dar uma idéia do espalhamento das amostras em torno da média.

Ambiente	n	σ (dB)	Frequência (MHz)
Loja de vendas	2.2	8.7	914
Escritório com paredes móveis	2.4	9.6	900
Escritório com paredes fixas	3.0	7.0	1500
Fábricas com instalações de metal, LOS	1.6	5.8	1300
Fábricas com instalações de metal, OLOS	3.3	6.8	1300

Tabela 8. Valores típicos de n para diversos cenários

Apesar da tabela acima não representar uma referência precisa para aplicações em 2.4 GHz, ela serve de subsídios para análise do comportamento de n em relação aos vários ambientes considerados.

A loja de vendas a varejo consiste de um prédio com pé direito alto, sem paredes e com prateleiras que alcançam a metade da altura do teto, deixando grandes áreas abertas na parte superior do prédio, permitindo, assim, a propagação do sinal. Com isso, n foi levemente superior a dois (valor padrão para espaço livre).

Para o escritório com paredes móveis (cubículos), há também uma grande área para propagação do sinal perto do teto, porém n foi um pouco maior que o anterior devido ao aumento da densidade de obstáculos dentro do ambiente.

Para o escritório com paredes fixas do chão ao teto, n aumenta significativamente, como resultado o RSSI (*Receive Signal Strength Indicator*) cai aproximadamente 30 dB por década de distância.

Para fábricas cujas instalações são de metal e com caminho direto LOS, os sinais refletidos devido à ótima propriedade condutiva dos metais, fortalecem o sinal transmitido tal como se houvesse um guia de ondas entre eles. Como resultado n é menor que a situação de espaço livre.

Para fábricas cujas instalações são de metal e com caminho direto OLOS, as reflexões causam um forte enfraquecimento do sinal transmitido, resultando num valor altíssimo para n .

2.3.2.1 Cálculo do alcance (d) por meio da PL simplificada

Para o cálculo do alcance, deve-se sumarizar algumas condições de contorno, em relação aos dispositivos Bluetooth, a saber:

- A potência do transmissor é 0 dBm (1mW) ou 20 dBm (100 mW);
- A sensibilidade do receptor é -70 dBm (padrão) ou -80 dBm (otimizado), para uma taxa de erro de bit (BER) de 10^{-3} ou melhor ;
- As antenas do receptor e do transmissor possuem ganho zero;
- Não serão consideradas perdas de potência nas antenas do receptor e do transmissor devido ao casamento de impedância.

Para avaliar estas especificações sumarizadas acima, consideremos um transmissor com potência máxima de saída de 20 dBm e um receptor otimizado com sensibilidade de -80 dBm. A interpretação destes valores é a seguinte: a perda no caminho PL pode ser tão grande quanto $20 - (-80) = 100$ dB, que o receptor ainda vai conseguir recuperar o sinal transmitido, considerando uma taxa de erro de bit de 10^{-3} .

A partir da fórmula de PL simplificada, a tabela 9 expõe os valores calculados para d com relação às diferentes condições sumarizadas acima.

Tipo de Ambiente	Potência do TX (dBm)	Sensibilidade do RX(dBm)	Alcance(m)
Espaço livre <i>n =2</i>	0	-70	31
	0	-80	100
	+20	-70	316
	+20	-80	1000
Leve <i>n =2.5</i>	0	-70	16
	0	-80	40
	+20	-70	100
	+20	-80	251
Moderado <i>n =3</i>	0	-70	10
	0	-80	22
	+20	-70	46
	+20	-80	100
Pesado <i>n =4</i>	0	-70	6
	0	-80	10
	+20	-70	18
	+20	-80	32

Tabela 9. Alcance da interface aérea por meio da PL simplificada

2.3.2.2 Regras práticas para o cálculo do alcance

Ao analisar a tabela 9 acima, podemos abstrair algumas regras básicas que podem auxiliar no cálculo estimado do alcance da interface aérea Bluetooth, de maneira extremamente fácil e rápida. Abaixo estão sumarizadas essas regras.

Regra um. Para ambientes em espaço livre e o caminho direto LOS.

- Considerando uma potência de TX de 0 dBm e uma sensibilidade de RX de -70 dBm, obtém-se um alcance de aproximadamente 30 metros;
- O alcance triplica para cada adição de 10 dB na potência do TX ou na otimização da sensibilidade do RX.

Regra dois. Para ambientes moderados, com $n = 3$.

- Considerando uma potência de TX de 0 dBm e uma sensibilidade de RX de -70 dBm, obtém-se um alcance de aproximadamente 10 metros;
- O alcance dobra para cada adição de 10 dB na potência do TX ou na

otimização da sensibilidade do RX.

Regra três. Para ambientes pesados, com $n = 4$.

- Considerando uma potência de TX de 0 dBm e uma sensibilidade de RX de -70 dBm, obtém-se um alcance de aproximadamente 6 metros;
- O alcance triplica para cada adição de 20 dB na potência do TX ou na otimização da sensibilidade do RX.

Ao compararmos os valores obtidos na tabela 9, por meio do cálculo através da fórmula da PL simplificada, com os valores estimados por meio das regras básicas enunciadas acima, veremos que as estimativas são mais pessimistas que os valores calculados. Porém isto não se revela como um problema, pois inserir pessimismo no cálculo do alcance de uma interface aérea, na prática, é uma regra de conduta bastante aceitável.

Para exemplificar o exposto, consideremos a seguinte situação:

- $n = 3$;
- Potência do TX igual a 20 dBm (transmissor classe 1);
- Sensibilidade do RX de -80 dBm (receptor otimizado).

Pela regra número 2, tem-se que o alcance deve dobrar em relação ao valor padrão de 10 m, a cada aumento de 10 dB da potência do transmissor ou da otimização da sensibilidade do receptor. Neste caso houve um aumento de 20 dB da potência do TX, resultando em um novo alcance de $(2 \times 10) \times 2 = 40$ m. Como também houve uma otimização de 10 dB da sensibilidade do RX, tem-se um alcance final estimado de $2 \times 40 = 80$ m. Este valor constata o pessimismo considerado na estimativa do alcance, pois, segundo a tabela acima, o alcance calculado deveria ser de 100 m.

2.3.2.3 Análise crítica do cálculo do alcance por meio da PL simplificada

Há quatro fatores que podem levar a erro o cálculo do alcance por meio da PL simplificada, a saber: a mobilidade dos dispositivos, causando variações no caminho direto (por exemplo, de LOS para OLOS); o caminho refletido, originado das reflexões do sinal transmitido quando bate em obstáculos com boas propriedades condutivas; a perda de potência devido às características das antenas do receptor e do transmissor tais como: pobre casamento de impedância, pobre eficiência e absorção do sinal de RF; e a interferência gerada por outros dispositivos que compartilham a mesma banda.

2.3.2.4 Otimização do modelo PL simplificada

Neste tópico analisaremos como aperfeiçoar o cálculo do alcance por meio da PL simplificada do ponto de vista da variação do caminho direto devido à mudança dos tipos de obstáculos. Na literatura há várias técnicas que objetivam realizar esta otimização. Serão expostas duas delas, a saber: o modelo *Ericsson Multipile Breakpoint Model* [8] e o modelo Toledo-Turkmani [7].

2.3.2.4.1 Modelo *multibreakpoint*

Este modelo propõe o ajuste do valor do expoente PL n de acordo com o aumento da distância d entre o transmissor e o receptor. Desta forma, são definidas múltiplas áreas, cada qual com o seu valor específico de n , válido dentro do contexto da sua respectiva área. A abrangência de cada área é expressa em faixa de valores da distância d .

Vamos interpretar esta técnica por meio de um exemplo. Consideremos dois membros de uma *piconet* localizados em uma mesma sala de um prédio. Nesta n vale 2.2. Se um dos membros se locomover para a sala vizinha, haverá uma maior perda devido ao obstáculo parede, resultando em um aumento suposto de n para 3. Se este mesmo membro da *piconet* agora descer para os andares inferiores do prédio, n naturalmente aumentará, digamos, para 4, de maneira a considerar esta variação do caminho devido à modificação dos obstáculos. Desta forma, há três (3) situações distintas, caracterizadas por valores de n diferentes. Segundo a referida técnica, este cenário poderia ser modelado da seguinte forma:

- $n = 2$ para d variando de 1 a 10 m (transmissor e receptor dentro da mesma sala), resultando em:

$$PL = 40 + 22 \log(d), \quad \text{para } 1 \leq d \leq 10. \quad (10)$$

- $n = 3$ para d variando de 10 a 20 m (receptor da sala vizinha), resultando em:

$$PL = 62 + 30 \log\left(\frac{d}{10}\right), \quad \text{para } 10 \leq d \leq 20. \quad (11)$$

- $n = 4$ para d maior que 20 m (receptor e transmissor em andares diferentes), resultando em:

$$PL = 71 + 40 \log\left(\frac{d}{10}\right), \quad \text{para } d \geq 20. \quad (12)$$

O primeiro termo de cada fórmula corresponde ao limite superior da fórmula antecedente e o segundo termo corresponde à perda adicional devido ao novo valor de n .

Agora, para calcular-se o alcance de uma interface aérea Bluetooth inserida dentro do contexto descrito acima, e considerando uma PL de 65 dB, deve-se utilizar a segunda fórmula, pois esta contempla valores de PL na faixa de 62 dB a 71 dB. Assim, procedendo ter-se-á um alcance de 12.5 metros.

2.3.2.4.2 Modelo Toledo - Turkmani

Este modelo utiliza uma outra abordagem para otimizar a PL simplificada. Esta técnica adiciona termos à equação da PL simplificada, para levar em conta as características físicas (*layout*) encontradas na maioria dos prédios [7]. Assim sendo, pesquisadores mediram a PL para 2.3 GHz em vários tipos de prédios e desenvolveram uma equação que se ajustasse mais às medições realizadas. Esta equação está exposta abaixo. Apesar de a mesma ter sido obtida através medições feitas na frequência de 2.3 GHz, as frequências utilizadas no Bluetooth estão bastante próximas e, portanto, a equação pode ser utilizada para os nossos fins de análise.

$$PL = 21.6 + 39.1 \log(d) + 3.8K_f - 17.8S_w - 8.8C_g - 0.014A_f, \quad (13),$$

onde o primeiro termo 21.6 dB é a perda para uma distância d de 1 metro; o segundo termo adiciona as perdas em relação a distância, considerando um expoente PL de 3.9; o K_f é simplesmente o número de pavimentos que separam o transmissor e o receptor; o termo S_w ajusta a equação para o caso de haver sinais de RF saindo por uma janela e retornando por outra; o termo C_g é 1 quando o transmissor e o receptor estão ambos localizados nos dois primeiros pavimentos do prédio, onde o sinal é mais forte; e, finalmente, o termo A_f é a área do pavimento onde estão localizados o transmissor e o receptor.

Os multiplicadores de cada termo podem ser ajustados para cada cenário com características de prédios diferentes, baseados em medições previamente realizadas.

2.3.2.5 Incerteza intrínseca da análise da propagação

Apesar dos esforços em aumentar a precisão dos cálculos realizados por meio do modelo simplificado PL, é um fato constatar a seguinte situação: a análise da propagação do sinal possui intrinsecamente um alto grau de incerteza. Não há como

desenvolver uma equação PL que seja precisa o suficiente para os diversos cenários apresentados. Mesmo que se consiga obter valores precisos em um determinado cenário, por meio de medições, estes valores se tornarão inválidos tão logo haja alteração no caminho entre o transmissor e o receptor devido, por exemplo, ao deslocamento de objetos e / ou pessoas. Assim sendo, a incerteza é algo que se deve conviver. Uma prática aceitável para contornar este problema é inserir pessimismo nos cálculos, garantindo que o sistema de comunicação irá operar com uma performance desejada na maior parte do tempo. Há duas formas de inserir pessimismo no cálculo do alcance através da PL simplificado:

- Assumir altos valores para o expoente PL n ;
- Assumir valores baixos para a PL, a fim de obter alcances menores.

Porém, como em tudo existe o contraponto, simplesmente inserir pessimismo no cálculo do alcance, partindo da equação PL simplificada, pode nos levar a superdimensionar o sistema de comunicação. Haja visto que um simples aumento de PL em 10dB, pode significar aumentar a potência do transmissor em 10 dB e, como isso, aumentar o custo do transmissor, aumentar o consumo da fonte de alimentação e aumentar o potencial de interferência causado em outros dispositivos vizinhos.

Neste contexto, conclui-se, até o momento, que o melhor procedimento para o cálculo do alcance de um sistema de comunicação, é primeiramente utilizar métodos que calculem a PL de maneira mais precisa, dentro de um determinado cenário (*layout*) onde os obstáculos permanecem fixos a maior parte do tempo, e após isso, inserir-se uma análise de pessimismo para levar em conta a incerteza intrínseca da propagação dos sinais de RF devido à mobilidade.

Já mencionamos anteriormente que existem diversas técnicas que se propõem a otimizar o cálculo da perda do caminho entre transmissor e receptor, dentre elas já expomos duas: o modelo *multibreakpoint* e o modelo Toledo-Turkmani. Há uma outra técnica de otimização que, ao contrário das anteriores, exige um maior poder de processamento, por isso demanda o uso de algoritmos computacionais. Por conseguinte, também consegue chegar a resultados mais precisos. Esta técnica é chamada de *Primary Ray Tracing*.

2.3.2.6 Primary Ray Tracing

Este método utiliza o seguinte procedimento: primeiramente todos os obstáculos significativos para a análise da propagação do sinal de RF são posicionados na planta

baixa do ambiente. Então, a PL correspondente a cada caminho de interesse é calculado, com base no atual conjunto de obstáculos entre o transmissor e o receptor, e, então, os contornos de PL são traçados na referida planta baixa do ambiente. Na prática, já foram desenvolvidos aplicativos para implementar este procedimento.

Para subsidiar o cálculo de PL através do *primary ray tracing* é preciso estabelecer as seguintes regras [9]:

- A intensidade do sinal cai repentinamente quando o mesmo passa por um obstáculo (aqui chamado de partição), e o valor desta atenuação depende das características da partição;
- Entre as partições ocorre uma perda de espaço-livre, pois a situação é correspondente a um cenário onde há um caminho direto LOS. Isto significa dizer que entre as partições, o expoente PL n é igual a 2;
- Perdas entre pavimentos ou andares diferentes de um prédio, exibem comportamento especial e são modeladas separadamente.

As regras supracitadas podem ser modeladas matematicamente por meio da equação abaixo.

$$PL = 20 \log \left(\frac{4\pi d}{\lambda} \right) + \sum_i [(P_i)(AF_i)] + FAF \quad (14)$$

Como, para a 2.45 GHz $\lambda = 0.122$ m, a equação pode ser escrita da seguinte forma:

$$PL = 40 + 20 \log(d) + \sum_i [(P_i)(AF_i)] + FAF \quad (15)$$

Examinando a equação (15) acima, vê-se claramente que os dois primeiros termos satisfazem à segunda regra. Eles correspondem à PL simplificada relativa à propagação em espaço livre. Neste ponto vale ressaltar que esta técnica também parte da equação da PL simplificada e adiciona termos para considerar as atenuações devido às características físicas das partições encontradas na maioria dos prédios.

O terceiro termo satisfaz à primeira regra. Ele corresponde ao somatório das perdas relativas às partições por onde o sinal de RF tem que passar no caminho para o receptor.

O símbolo P_i corresponde à quantidade de partições do tipo i , e o símbolo AF_i corresponde ao fator de atenuação da partição i . Sendo assim, se no caminho entre o transmissor e o receptor houver três paredes, com fator de atenuação $AF_1 = 3$ dB, e uma porta com $AF_2 = 2$, este terceiro termo da equação se torna, $(3 \times 3) + (1 \times 2) = 11$ dB.

A tabela 10 mostra valores médios relativos aos fatores de atenuação das partições mais comuns, encontradas em casas e ambientes de escritório [8]. Embora estes valores tenham sido obtidos por meio de medições considerando a frequência de 1.3 GHz, eles são bem próximos dos valores para a frequência de 2.45 GHz. Isto se deve ao fato de que a atenuação de cada partição é muito mais dependente da característica do material que a constitui do que da frequência.

Tipo de partição	AF_i (dB)
Paredes Fixas	3.0
Paredes Móveis	1.4
Portas	2.0
Metal	5.0
Janelas	2.0
Paredes exteriores	10.0
Paredes de porão	20.0

Tabela 10. Fatores de atenuação por obstáculo

O quarto termo satisfaz à terceira regra. Ele corresponde à atenuação adicional que o sinal de RF sofre ao passar por um ou mais partições referentes a tetos / pisos, no caminho para o receptor. Sendo assim, caso o transmissor e o receptor estiverem no mesmo piso, o valor de FAF é zero. A sigla FAF significa *floor attenuation factor*. A tabela 11 expõe os valores médios típicos do FAF , bem como o seu desvio - padrão [9].

Números de andares	FAF (dB)	σ (dB)
1	13	7
2	19	2.8
3	24	1.7
4 ou mais	27	1.5

Tabela 11. Fator de atenuação por andar

Para cada aplicação é possível melhorar a precisão do cálculo de PL, por este método, por meio da medição direta da atenuação das diversas partições. Isto pode ser realizado através do seguinte procedimento.

1. Coloca-se o par transmissor-receptor em linha de visão e a uma distância d conhecida;
2. Envia-se uma onda CW para o receptor;

3. Mede-se a intensidade do sinal recebido através do RSSI (*Receive Signal Strength Indicator*);
4. Coloca-se a partição de interesse entre o transmissor e o receptor, mantendo-os à mesma distância d do passo 1;
5. Mede-se o novo valor da intensidade do sinal recebido;
6. O fator de atenuação da partição é obtido da diferença entre os valores medidos nos passos 3 e 5.

Para constatar o ganho de precisão obtido no cálculo de PL através do *primary ray tracing*. Examinemos a figura 14. Nela foram traçados, por meio de análises computacionais feitas na planta baixa de um prédio específico, os contornos de PL calculados em (a) por meio da equação (9) da PL simplificada, e em (b) por meio da equação (15) do *primary ray tracing*.

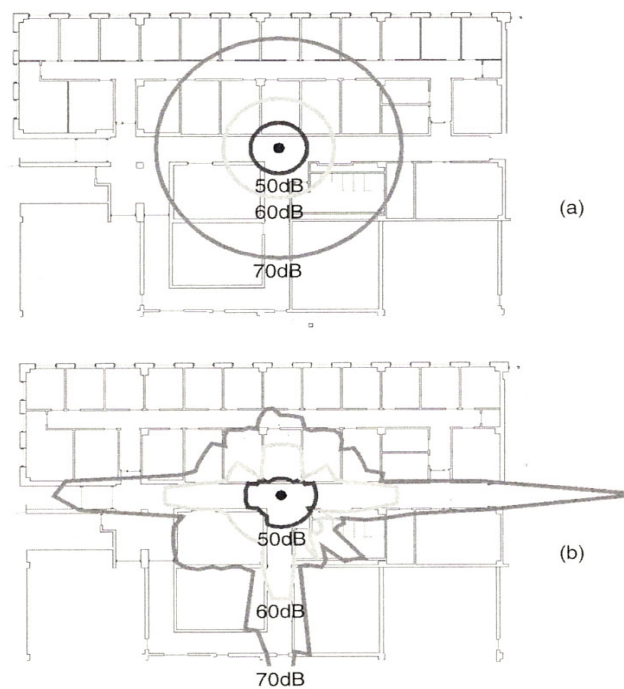


Figura 14. PL simplificada x *Primary Ray Tracing*

Fonte: Wireless Valley Communications, Inc.

Em ambas as figuras o transmissor é representado por um ponto preto. Na figura 14 (a), os contornos são círculos concêntricos. Como consequência, é atribuído o mesmo valor de PL tanto para uma situação LOS (corredores), quanto para uma situação OLOS (dentro das salas). Isto explica, então, o desvio padrão de aproximadamente 10 dB, apresentado na tabela 8.

Já na figura 14(b) os contornos não são mais círculos concêntricos, observando-se que a intensidade dos sinais é mais fraca dentro das salas, e mais forte nos corredores (onde não há obstáculos e onde ainda acontece o efeito guia de onda), como era de se esperar. O desvio padrão fica em torno de 3 dB a 5 dB, constatando-se, assim, a otimização do cálculo de PL.

2.3.2.7 Estimativa do alcance do Bluetooth por meio do *Primary Ray Tracing*

De posse dos contornos de PL, obtidos por meio do *primary ray tracing*, torna-se fácil calcular o alcance de um canal de comunicação Bluetooth. Basta que as características de potência de saída do transmissor e sensibilidade do receptor sejam suficientes para subjugar a perda em decorrência das partições do caminho considerado. Como mencionado anteriormente no tópico incerteza intrínseca da análise de propagação é sábio reduzir o valor PL inserindo, desta forma, pessimismo na análise. Na prática, recomenda-se reduzir PL em 10 dB [3], já que os transmissores Bluetooth possuem baixa potência de saída.

2.3.3 Multipercurso

Agora chegou o momento de analisarmos o segundo fator que pode causar incerteza no cálculo da PL, o caminho refletido. Como mencionado anteriormente, quando uma onda de rádio bate em uma superfície de um obstáculo, uma significativa parte da energia do sinal passa através do mesmo e continua o seu caminho, formando o caminho direto estudado até o momento. A outra parte significativa da energia do sinal é absorvida e a outra parte é refletida, como mostrado na figura 15.

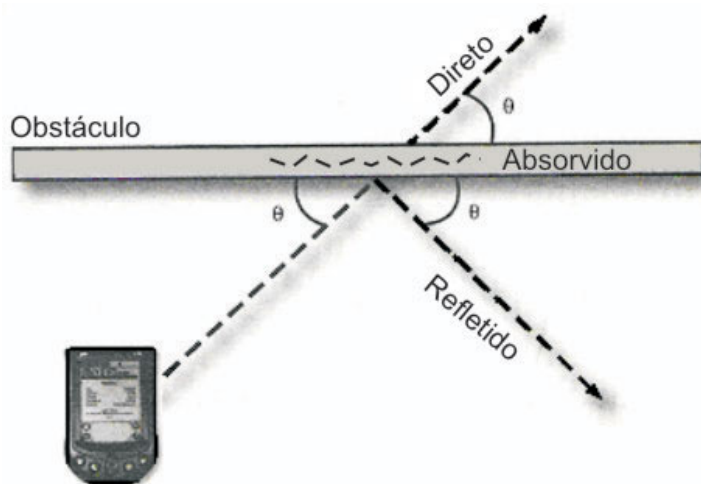


Figura 15. Fragmentação da energia do sinal na presença de obstáculo

Estes sinais refletidos podem chegar ao receptor com características de amplitude e fase diferentes das do sinal transmitido. Sendo assim, para se obter o sinal resultante no receptor, é necessário que se faça uma soma vetorial destes sinais.

Os sinais refletidos são denominados conforme a quantidade de reflexões sofridas. Os de primeira ordem sofreram apenas uma reflexão, os de segunda-ordem sofreram duas reflexões, e assim por diante. A figura 16 ilustra o exposto. Nela, um transmissor e um receptor foram colocados em uma mesma sala com quatro paredes, identificando-se um caminho direto LOS, uma reflexão de primeira-ordem e uma de segunda-ordem.

Vale salientar que a cada reflexão parte da energia é absorvida pelo obstáculo. Sendo assim, a energia do sinal refletido decresce com o aumento da ordem da reflexão.

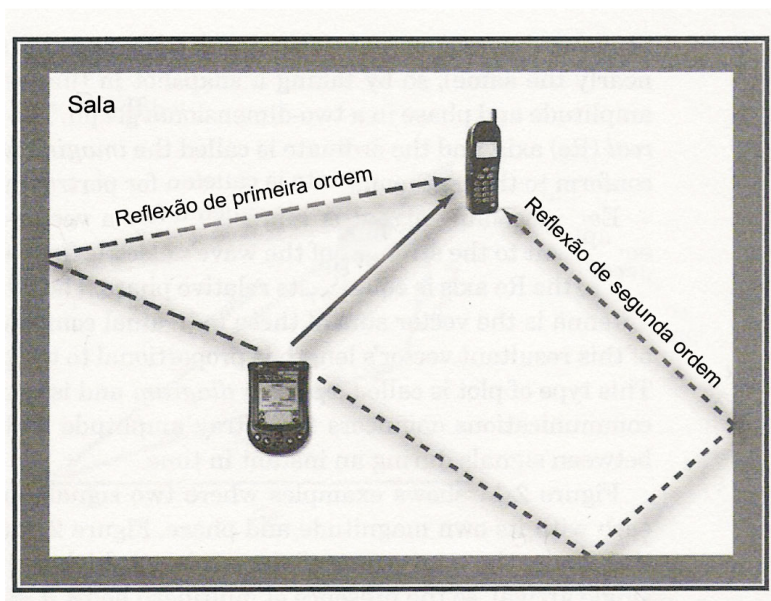


Figura 16. Ordem das reflexões

Para o cálculo do sinal resultante no receptor, é necessário saber até qual ordem de reflexão deve-se considerar. Pesquisadores vêm demonstrando que apenas as reflexões de primeira ordem são significativas na maioria das situações [10]. A exceção ocorre em ambientes onde são encontrados obstáculos com alto índice de reflexão, onde reflexões de maior ordem devem ser consideradas para obter maior precisão, pois a maior parte do sinal é realmente refletida.

Imaginando uma situação em que o sinal refletido relevante seja o de primeira-ordem. E que este chega ao receptor com amplitude igual ao sinal do caminho direto, porém com fase oposta. A soma vetorial resultante será nula. De outra forma, se estes possuírem a mesma fase, a resultante terá o dobro da amplitude. Na prática, o multipercorso prejudica mais do que ajuda.

Esta soma vetorial pode ainda ser alterada substancialmente quando o receptor é movido em distâncias que correspondem à somente frações do comprimento de onda. Para as frequências da banda utilizada pelo Bluetooth, estas flutuações na intensidade do sinal recebido podem ocorrer rapidamente, mesmo quando o receptor se move a uma velocidade normal de caminhada, devido ao relativamente curto comprimento de onda - 0.122 m. Estas perdas e flutuações são chamadas de Perdas em Pequena Escala, pois não consideram as perdas decorrentes da presença de obstáculos.

2.3.3.1 Características do multipercurso

A mobilidade do receptor, do transmissor e / ou de obstáculos no canal de comunicação causa variações de amplitude, fase e frequência entre os sinais que chegam ao receptor. Deste ponto de vista, os efeitos do multipercurso podem ser classificados dentro de duas grandes categorias, a saber:

- Espalhamento Doppler. Efeitos ligados ao deslocamento de frequência entre os vários componentes do sinal devido à mobilidade dentro do canal de comunicação;
- Espalhamento Temporal. Efeitos ligados ao atraso de tempo entre as chegadas de cada sinal de ordem significativa, na antena do receptor.

2.3.3.1.1 Espalhamento Doppler

O efeito desta característica no sinal recebido torna-se mais claro ao analisá-lo separadamente no caminho direto LOS e no caminho refletido.

Para o caminho direto LOS, a frequência de um determinado sinal CW ao longo do mesmo é reduzida segundo a quantidade [3]:

$$f_d = \frac{v}{\lambda}, \quad (16)$$

onde f_d é o deslocamento Doppler, v é a velocidade do receptor em relação ao transmissor e λ é o comprimento de onda da portadora.

Sendo assim, se a frequência do transmissor é 2.45 GHz e se o receptor está se movendo a uma velocidade relativa de 1 m/s, o receptor verá a frequência do transmissor ao longo do caminho direto ser reduzida de aproximadamente 8 Hz.

Já o cálculo de f_d para o caminho refletido, depende da posição do receptor em relação ao transmissor. Suponhamos, então, que as antenas do transmissor e do receptor estão ambas a uma altura h do chão, desconsiderando o comprimento de suas antenas. Se o

caminho direto LOS é aumentado a uma velocidade v_{LOS} , então o caminho refletido cresce a uma velocidade [3]:

$$v_{ref} = v_{LOS} \cos \theta = v_{LOS} \frac{d}{\sqrt{(2h)^2 + d^2}}, \quad (17)$$

onde d é o comprimento do caminho direto LOS.

Sendo assim, f_d para o caminho refletido é:

$$f_d = \frac{v_{ref}}{\lambda} = \frac{1}{\lambda} \frac{v_{LOS} d}{\sqrt{(2h)^2 + d^2}} \quad (18)$$

Voltando ao exemplo anterior onde d valia 1 m, $v_{LOS} = 1$ m/s e considerando $h = 2$ m, tem-se $v_{ref} = 0.24$ m/s e, por conseguinte, $f_d = 1.96$ Hz. Isto significa que o caminho refletido produz um deslocamento instantâneo na frequência de aproximadamente 2 Hz. À medida que o caminho direto LOS aumenta (d cresce), o valor de v_{ref} se torna próximo do valor de v_{LOS} , e os dois deslocamentos Doppler se tornam quase idênticos.

O deslocamento Doppler indica somente a taxa em que ocorrerá o enfraquecimento (desvanecimento) do sinal no receptor. Já a intensidade deste desvanecimento depende exclusivamente da soma vetorial dos sinais do caminho direto e refletido.

O Espalhamento Doppler é definido como duas vezes o maior deslocamento Doppler dentre todos os múltiplos caminhos de ordem significativa que chegam à antena do receptor. Taxas de enfraquecimento excessivas no RSSI. (*Receive Signal Strength Indicator*) ocorrem em decorrência do Espalhamento Doppler, normalmente originadas pela movimentação do transmissor, do próprio receptor ou de objetos próximos. Para ilustrar a teoria, as figuras 17(a) e 17(b) nos mostram um cenário onde há um forte desvanecimento, ocorrido quando o transmissor e o receptor se movem um em relação ao outro. Esta dupla mobilidade gera múltiplos caminhos aleatórios de entrada, levando a um desvanecimento do sinal resultante no receptor de aproximadamente 30 dB em determinados instantes.

Se o transmissor e o receptor estão fixos um em relação ao outro, o desvanecimento do sinal no receptor pode ainda ocorrer devido à mobilidade de objetos próximos. Esta situação é mostrada nas figuras 17(c) e 17(d). Agora há uma potência substancial no sinal do caminho direto LOS / OLOS que, por estar fixo, não apresenta qualquer deslocamento Doppler. O desvanecimento ocorre a aproximadamente à mesma taxa

devido ao deslocamento Doppler causado pelo caminho refletido (em decorrência do deslocamento dos obstáculos), porém com uma intensidade bem mais baixa.

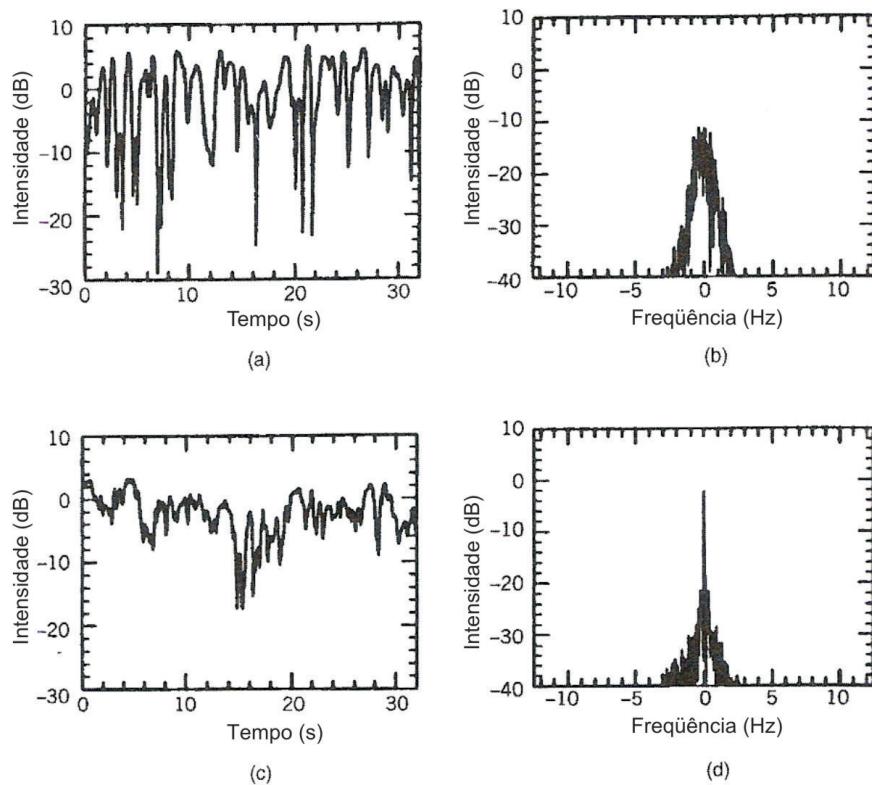


Figura 17. Desvanecimento do sinal devido ao Espalhamento Doppler

2.3.3.1.1 Probabilidade de interrupção – $\Pr(outage)$

Esta característica define qual a probabilidade do desvanecimento no sinal transmitido, devido ao Espalhamento Doppler, ser tal que a intensidade do sinal recebido seja abaixo da sensibilidade do receptor, resultando em perda do mesmo.

Para modelar matematicamente esta probabilidade, é necessário fazer-se as seguintes considerações:

- Todas as frequências dentro da largura de banda do sinal transmitido enfraquecem conjuntamente;
- Todos os caminhos de entrada estão sujeitos a deslocamentos Doppler aleatórios.

Um canal que satisfaz estas duas condições exibe um desvanecimento chamado de Rayleigh. Neste a probabilidade $\Pr(outage)$ de que a potência do sinal caia abaixo de certo limiar P_{th} , pode ser expressa em relação à potência média do sinal P_{ave} , através da fórmula (19) [8]:

$$\Pr(\text{outage}) = 1 - \exp\left(-\frac{P_{th}}{P_{ave}}\right) \quad (19)$$

Se o valor de P_{th} for -70 dBm e o valor de P_{ave} for -60 dBm. Após transformá-los para Watt e aplicar a equação (19), ter-se-á, $\Pr(\text{outage})$ igual a 0.1. Significando que em cerca de 10 % do tempo, o canal estará fora de operação.

Um exemplo de um canal com estas características foi mostrado nas figuras 17(a) e 17(b). Nele, o deslocamento aleatório entre o transmissor e o receptor causa o surgimento de múltiplos caminhos aleatórios de entrada e, conseqüentemente, o surgimento de deslocamentos Doppler aleatórios. O que satisfaz o segundo pré-requisito exposto.

Nas figuras 17 (c) e 17 (d), o transmissor e o receptor estão fixos um em relação ao outro e, por isso, o caminho direto LOS / OLOS entre eles não apresenta deslocamentos Doppler aleatórios. Como conseqüência, o sinal do caminho direto é forte o suficiente para dominar o efeito Doppler dos caminhos refletidos, gerados em decorrência do deslocamento dos objetos próximos. Diz-se que um canal com estas características apresenta um desvanecimento chamado de Rician. Como resultado, embora o desvanecimento do sinal ocorra à mesma taxa, a sua intensidade é relativamente menor que a do canal Rayleigh, e a equação (19) pode ser utilizada para calcular-se o limite máximo de $\Pr(\text{outage})$ que este canal pode apresentar [3]. Sendo assim, retomando ao exemplo supracitado, o canal *Rician* teria uma probabilidade máxima de estar 10% do tempo fora de operação.

2.3.3.1.1.2 Tempo de coerência - Tc

Sabe-se, até o momento, que o sinal recebido em um receptor pode sofrer flutuações em sua intensidade a uma taxa dada pelo deslocamento Doppler f_d , devido à deslocamentos ocorridos no transmissor, receptor e / ou objetos. O tempo de coerência analisa o comportamento do tempo de duração de símbolo em relação ao máximo deslocamento Doppler f_d . Se, em um sistema de comunicação digital, a duração do símbolo, do sinal de banda base, for maior de o inverso do máximo f_d , o símbolo terá grande probabilidade de ser distorcido pelos desvanecimentos causados pelo f_d . Caso contrário, a probabilidade é reduzida.

Sendo assim, o tempo de coerência T_C tem por objetivo estabelecer uma relação com f_d , de sorte que se obtenha um parâmetro para estabelecer a quantidade de distorção de símbolo que é tolerável em um sistema de comunicação digital. Esta relação é dada por [8]:

$$T_C = \frac{0.4}{f_D}, \quad (20),$$

onde f_D é o maior deslocamento Doppler.

Em outras palavras, a duração do símbolo deve ser no máximo 40 % do período do desvanecimento (inverso de f_D).

Neste contexto, os sistemas de comunicação podem ser classificados em:

- Desvanecimento lento. O tempo de coerência é maior que o período do símbolo. Então, o sinal de banda base varia mais rápido que o canal. Significando que os símbolos sofrerão distorções desprezíveis.
- Desvanecimento rápido. O tempo de coerência é menor que o período do símbolo. Então, o canal varia mais rápido que o sinal de banda base. Significando que os símbolos poderão sofrer distorções significativas.

2.3.3.1.2 Espalhamento temporal

Esta característica do multipercorso analisa o espalhamento no tempo entre as chegadas de vários sinais refletidos, de ordem significativa, na antena do receptor. Ao contrário do espalhamento Doppler, não é necessário que haja deslocamento algum, ele é estritamente em função dos tempos de chegada. Logicamente, em havendo deslocamento do receptor, do transmissor e / ou dos obstáculos próximos, as características do espalhamento temporal irão variar com o tempo, pois o percurso dos sinais refletidos estará sendo continuamente modificado.

O espalhamento temporal pode ser quantificado através do *maximum excess delay* ou do *rms delay spread*. O *maximum excess delay* é definido como o tempo entre a chegada do primeiro sinal e a do último, acima de um limite preestabelecido (10 dB abaixo do sinal mais forte). Já o *rms delay spread* quantifica o número, suas respectivas intensidades e os tempos de chegada de vários componentes do símbolo transmitido. O procedimento de cálculo do *rms delay spread* pode ser encontrado na referência [8].

2.3.3.1.2.1 Largura de faixa de coerência

Agora iremos identificar como esta chegada de vários sinais refletidos em instantes diferentes na antena do receptor pode afetar o canal de comunicação. Para isso, examinemos o exemplo da figura 18. Nela o sinal enviado do transmissor, chega ao receptor por meio de dois caminhos, a saber: o caminho direto LOS e o refletido (gerado pela reflexão na parede). Suponhamos que o tempo de duração do símbolo é igual à diferença de tempo entre a chegada consecutiva de um símbolo pelo caminho LOS e pelo caminho refletido.

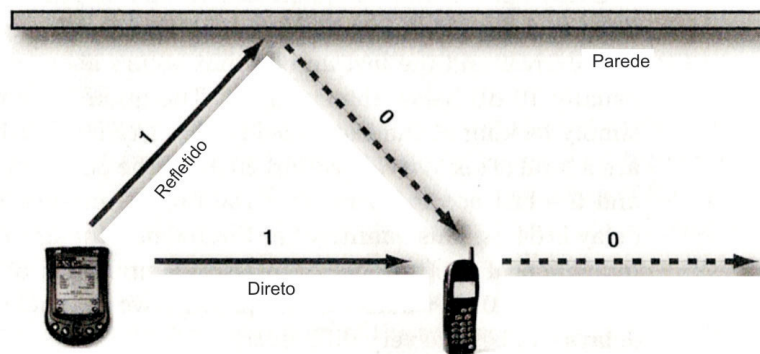


Figura 18. Caminho direto LOS x Caminho refletido

Primeiramente aparece somente o símbolo zero no receptor, e ele facilmente o detecta. Depois, aparecem juntos o símbolo 1 do caminho direto e o símbolo 0 do caminho refletido. Claramente há uma interferência do símbolo do caminho refletido no símbolo do caminho direto LOS. O receptor está tentando detectar o símbolo 1 e sofre a interferência do símbolo 0. Se o detector decidir pelo símbolo 0, haverá erro. E, finalmente aparece o símbolo 1 do caminho refletido. Esta interferência supracitada é chamada de ISI (*Intersymbol Interference*).

Neste contexto, conclui-se que o efeito do Espalhamento Temporal é gerar ISI e, conseqüentemente, causar erros na detecção de um determinado símbolo no receptor. Estes erros devido à interferência intersimbólica conduzem o canal de comunicação a taxas de erro de bit irreduzíveis, pois nem o aumento da potência do transmissor é capaz de reduzi-la, já que o caminho direto LOS e o caminho refletido serão aumentados na mesma proporção. Porém, há uma faixa de frequência dentro do canal em que a distorção devido a ISI é considerada desprezível, e a resposta do canal é então considerada plana. Esta quantidade é chamada de largura de faixa de coerência e é dada pela fórmula (21) [3].

$$B_C \approx \frac{0.1}{T_{RMS}}, \quad (21)$$

onde B_C é a largura de faixa de coerência, e T_{RMS} é o *rms delay spread*.

Esta fórmula pode ser interpretada da seguinte forma. Se o tempo de símbolo for 10 vezes maior que o T_{RMS} , somente 10% do tempo do símbolo será afetado pelo símbolo anterior. O restante do tempo (90%) será recebido sem interferência, pois agora todas as reflexões significativas são do mesmo símbolo.

De posse da largura de faixa de coerência, o sistema de comunicação pode ser classificado em:

- **Desvanecimento Plano.** A largura de banda do sinal transmitido é 10 vezes menor que a largura de faixa de coerência do canal. Conseqüentemente, o período do símbolo é 10 vezes maior que o *rms delay spread*. Sendo assim, o sinal transmitido é pouco alterado pela ISI causada pelo multipercurso (espalhamento temporal).
- **Desvanecimento Seletivo na Frequência.** A largura de banda do sinal transmitido é maior que a largura de faixa de coerência do canal. Conseqüentemente, o período do símbolo é 10 vezes menor que o *rms delay spread*. Sendo assim, o sinal transmitido é fortemente alterado pela ISI causada pelo multipercurso (espalhamento temporal).

2.3.3.2 Caracterização do canal Bluetooth quanto ao multipercurso

- **Espalhamento Doppler.** Para caracterizarmos o comportamento de um canal Bluetooth quanto aos efeitos do Espalhamento Doppler, é necessário calcular o tempo de coerência T_C do canal, e depois compará-lo com o período do símbolo.

Sendo assim, consideremos uma *piconet* dentro de um ambiente fechado, onde objetos podem se mover a uma velocidade de caminhada de 1.3 m/s. Pela equação 16, calcula-se um deslocamento Doppler de aproximadamente 10 Hz. E pela equação 20 calcula-se um tempo de coerência T_C de 40 ms. Como o tempo de símbolo do sinal de banda base em um canal Bluetooth é de 1 μ s, cerca de 40000 vezes menor que o T_C calculado, conclui-se que o canal possui um desvanecimento lento. Sendo assim, a variação de amplitude dos sinais transmitidos devido ao Espalhamento Doppler é praticamente desprezível.

- **Espalhamento Temporal.** Para caracterizarmos o comportamento de um canal Bluetooth quanto aos efeitos do Espalhamento Temporal, é necessário primeiramente calcular a largura de faixa de coerência B_C do canal, e depois compará-lo com a largura de banda do sinal transmitido.

Para isso, consideremos uma *piconet* situada dentro de uma sala com dimensões consideravelmente grandes. Neste caso, o *rms delay spread* é de aproximadamente 50 ns. Pela equação 21, obtém-se uma largura de faixa de coerência de 2 MHz. Como a largura de banda máxima de um canal Bluetooth em cada salto é de no máximo 1 MHz. O canal Bluetooth é considerado plano. Concluí-se, então, que o incremento da BER devido à interferência intersimbólica causada pelo espalhamento temporal, é desprezível, pois na maior parte do tempo o canal possuirá uma resposta plana.

Vale salientar que o aumento da taxa de transmissão de bit do Bluetooth fará com que o canal passe a ter um desvanecimento seletivo na frequência, sendo necessária a equalização do canal para se combater a ISI.

Neste ponto, podemos concluir que os efeitos do multipercorso, caracterizados pelo Espalhamento Doppler e pelo Espalhamento Temporal, não causarão distorção significativa no canal Bluetooth. Mais precisamente, significa dizer que:

- A ISI, causada pelo Espalhamento Temporal, não aumentará significativamente a taxa de erro de bit do canal;
- A taxa de desvanecimento, causada pelo Espalhamento Doppler, será desprezível, de maneira que o canal apresentará uma resposta plana.

Porém, não se pode concluir que o multipercorso não afeta o canal, pois a intensidade do desvanecimento do sinal transmitido depende exclusivamente da soma vetorial dos sinais do caminho direto e refletido. E, infelizmente, na prática este valor pode chegar a 30 dB.

2.3.3.3 Técnicas para compensar os efeitos do multipercorso

Existem várias técnicas que objetivam compensar os efeitos do desvanecimento causado pelo multipercorso. Alguns apenas convivem com ele, incluindo margens de perda no cálculo de PL, para então obter uma estimativa da máxima faixa de atuação. Porém há outras técnicas que tentam compensá-lo, quais sejam: diversidade (no espaço, no tempo, na frequência, no caminho e em ângulo), equalização e controle de erro.

O Bluetooth combina as técnicas da diversidade na frequência, através do *FHSS*, diversidade no tempo, através da ARQ (*automatic repeat request*), e o controle de erro. Apesar de aqui caracterizarmos a ARQ como uma técnica de diversidade do tempo, ela também é considerada uma técnica de controle de erro. E, desta forma será descrita.

2.3.3.3.1 Controle de erro

Esta técnica consiste em inserir bits redundantes nos sinais de banda base transmitidos, de maneira que o receptor consiga usá-los para detectar e corrigir erros no pacote de dados recebido, diminuindo assim a taxa de erro de bit do canal. Há três formas de se implementar o controle de erro, a saber:

- **Códigos detectores de erro.** São bits adicionais inseridos no pacote de dados, para que o receptor possa checar a autenticidade do pacote recebido.
- **Códigos corretores de erro.** São bits adicionais inseridos no pacote de dados para que o receptor possa corrigir eventuais erros encontrados no pacote de dados recebido.
- **ARQ (*automatic repeat request*).** Utilizado em conjunto com os códigos detectores de erro, para permitir que o receptor solicite a retransmissão de um pacote de dados que foi corrompido no canal de comunicação. Se um pacote de dados está sem erros, o receptor retornará uma mensagem de reconhecimento positivo (ACK) para o transmissor. Caso contrário, o receptor retornará uma mensagem reconhecimento negativo (NAK). Se uma mensagem ACK é recebida, o transmissor envia o próximo pacote da mensagem; caso contrário, o último pacote transmitido será reenviado.

O Bluetooth se utiliza destas três formas de controle de erro. Primeiro, o código corretor de erro é aplicado, para eliminar os possíveis erros. Depois, o pacote é checado por meio dos códigos detectores de erro, de maneira a verificar se ainda há erros no pacote recebido, pois há um limite para a correção de erros realizada na etapa anterior. E, finalmente, inicia-se o processo de ARQ descrito acima. Vale salientar que, devido à técnica de diversidade na frequência provida pelo FHSS, cada mensagem ACK ou NAK é enviada em um canal diferente.

2.3.4 Antenas

Até o momento nos nossos cálculos de *PL*, as antenas do receptor e do transmissor foram consideradas como ideais, de modo que toda a potência entregue para a antena do

transmissor era irradiada para o espaço-livre, e toda potência irradiada absorvida pela antena do receptor, era transferida para o seu estágio de entrada. Este é um cenário ideal onde há uma máxima transferência de potência. Na prática, para ocorrer esta máxima transferência de potência é necessário que haja um perfeito casamento de impedância entre as antenas e os dispositivos a elas conectados. Quando não há um perfeito casamento de impedância, ocorrem as chamadas perdas de casamento.

Para caracterizar melhor as perdas de casamento, consideremos o exemplo da figura 19, que nos mostra o comportamento dos sinais em um cenário onde não há casamento de impedância entre a antena e os dispositivos.

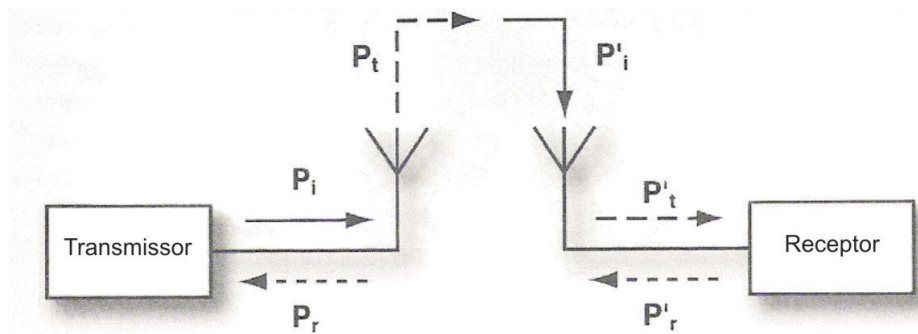


Figura 19. Perdas devido a falta de casamento de impedância

Quando um sinal de potência P_i é enviado para antena, parte P_t desta potência é irradiada para o espaço-livre, e outra parte P_r é refletida de volta para o transmissor. Da mesma forma acontece no receptor. A antena envia para o estágio de entrada um sinal com potência P'_i , parte P'_t é absorvida pelo estágio de entrada, e outra parte P'_r é refletida de volta para a antena. As relações P_i/P_t no transmissor e P'_i/P'_t no receptor, compõem as chamadas de perdas de casamento.

Vale salientar que mesmo estando as antenas e os dispositivos a elas acoplados previamente casados, a impedância das antenas pode variar devido à mobilidade de objetos próximos das mesmas [3], causando reflexão e, conseqüentemente, diminuição da potência irradiada.

Outro fator que contribui para aumentar a imprecisão dos cálculos de PL é o fato de que as antenas reais não são fontes isotrópicas que irradiam os sinais igualmente bem em todas as direções. Nelas a potência do sinal irradiado em algumas direções é menor que o das fontes isotrópicas e, conseqüentemente, são maiores que estas em outras direções. A potência da antena medida em relação a uma fonte isotrópica é chamada de *EIRP* (*Effective Isotropic Radiated Power*) e é dada em dBi. Esta informação nos diz o ganho

que uma antena pode apresentar, em uma determinada direção específica, em relação a uma fonte isotrópica.

As antenas Bluetooth, geralmente pertencem a três categorias de projeto: a meia - onda, a *PIFA* (*planar inverted F antenna*) e a *patch*. A primeira é claramente identificada pois é uma barra rígida ou flexível feita de material plástico colocada junto aos dispositivos. Já a segunda e a terceira são antenas construídas na placa de circuito impresso e ficam fora do campo visual. As antenas *patch* podem ser construídas também na superfície dos circuitos integrados de rádio do Bluetooth. As figuras 20, 21 e 22 ilustram cada uma delas.

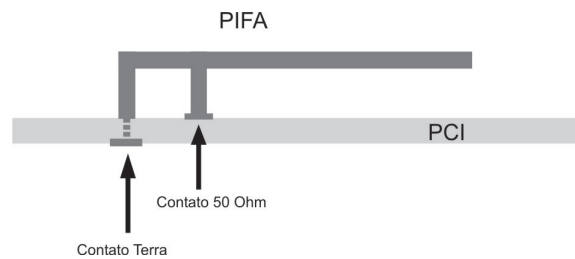


Figura 20. Antena PIFA

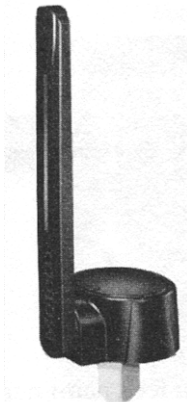


Figura 21. Antena meia-onda

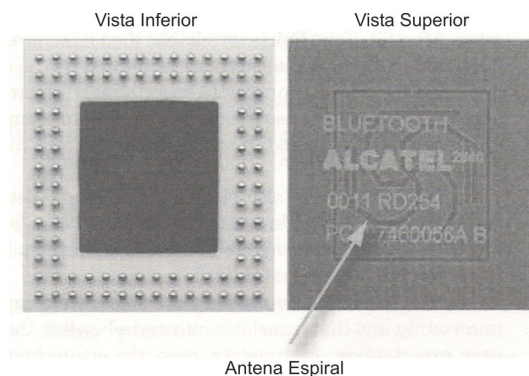


Figura 22. Antena patch em um CI de rádio Bluetooth

2.4 Banda base e Controlador de enlace (*Link Controller*)

2.4.1 Descrição básica

O protocolo de banda base implementa as funções básicas de uma *piconet*, quais sejam:

- Montagem de pacotes para protocolos de níveis mais altos da pilha de protocolos Bluetooth (ver figura 2) e posterior envio para o estágio de rádio;
- Recepção de bits e remontagem de pacotes para o posterior processamento dos protocolos das camadas superiores;
- Sincronização da *piconet*;
- Controle de operação do canal;
- Controle de erro;
- *Whitening*;
- Operações básicas de segurança.

Estas funções são executadas pelo o controlador de enlace (*link controller*), que se situa acima da etapa de rádio, identificado na figura 2. O controlador de enlace (*link controller*) pode ser acessado pela aplicação através do gerente de enlace (*link manager*) – para fins de controle da *piconet*, pelo L2CAP (*logical link control e adaptation protocol*) - para transferência de dados e, diretamente pela aplicação para transferência de informações de áudio bidirecional a fim de reduzir a latência.

Os pacotes da banda base possuem a seguinte estrutura de campos: um campo para o código de acesso (*access code*), um campo para o cabeçalho (*header*) e um para a carga útil (*payload*). O código de acesso tem um tamanho nominal de 72 bits e é usado inicialmente para obter a sincronização do receptor, necessária para a correta decodificação dos bits subseqüentes. Porém, contém outras informações como a identificação da *piconet*. O cabeçalho possui 54 bits que contém o endereço de destino, o tipo de dado que está sendo enviado e informações para o controle de erro. Finalmente, o campo carga útil é um campo de tamanho variável que contém parte da mensagem a ser enviada. Na transmissão, o bit menos significativo é enviado primeiro. Este procedimento é chamado de “*Little Indian*”.

Neste ponto vale salientar que somente o campo carga útil (*payload*) transporta a mensagem a ser trocada entre dois dispositivos Bluetooth dentro de uma *piconet*. Os demais campos são necessários para o devido controle da comunicação. Sendo assim,

fica claro que a **vazão** (tradução da palavra *throughput*, que significa taxa efetiva de transmissão de bits) é inferior aos 1 Mbps falados anteriores.

2.4.2 Operação em uma configuração ponto-a-ponto

Nesta configuração há apenas dois dispositivos, o mestre e o escravo. Esta configuração possui as seguintes características de operação:

- A troca de informações é feita através do esquema de multiplexação no tempo *fast TDD*, estabelecendo um canal de comunicação *full duplex* forçado;
- O tempo é dividido em *slots* (janelas de tempo) de 625 μ s identificados através de números inteiros consecutivos;
- O mestre transmite para o escravo nos *slots* pares, e o escravo transmite para o mestre nos *slots* ímpares. Sendo assim, só há transmissão a cada 1.25 ms;
- Em cada *slot* é realizada a transmissão de um pacote de dado completo, sendo que cada transmissão é realizada em uma frequência de salto diferente.

A operação descrita acima está ilustrada na figura 23.

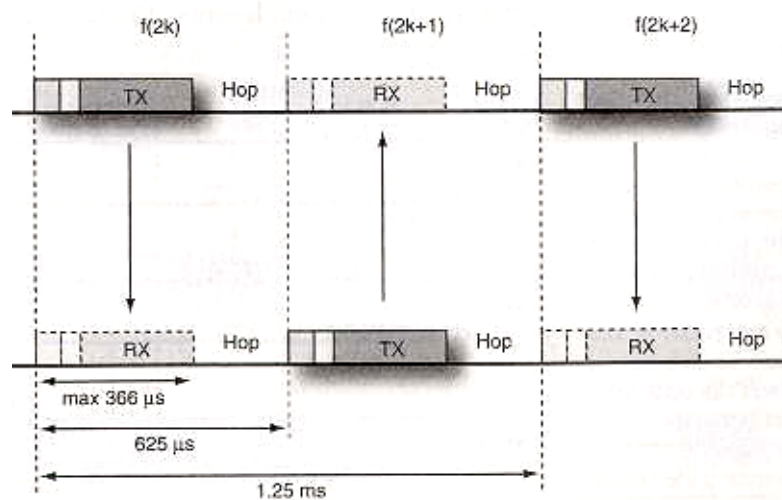


Figura 23. Operação em uma configuração ponto-a-ponto

Apenas 366 μ s do tempo de um *slot* (janela de tempo) são utilizados para transmissão do pacote de dados. Neste intervalo serão enviados 366 bits, pois a taxa de transmissão do Bluetooth é de 1 bit a cada 1 μ s. Como só há transmissão a cada 1.25 ms, a máxima taxa de transmissão de bits por nó nesta configuração é de 292.8 Kbps. E, conseqüentemente, a vazão máxima por nó é 192 Kbps. O restante do tempo, 259 μ s, é reservado para efetuar a mudança da frequência de salto, de acordo com a seqüência de salto acordada durante o processo de estabelecimento de conexão. Devido à

simplificação do circuito de relógio dos dispositivos Bluetooth, podem ocorrer atraso de tempo (*jitter*) e deslocamento de frequência (*drift*) entre os relógios do mestre e do escravo. Então, para absorver isto, o nó receptor tem que considerar ainda uma janela de 10 μ s de variação no tempo de chegada do pacote.

2.4.3 Operação em uma configuração ponto a multiponto

Nesta configuração há um mestre e dois ou mais escravos. A vazão da *piconet* é então dividida entre os escravos. Além das características de operação citadas na configuração ponto-a-ponto, esta configuração possui as seguintes características:

- Controle centralizado realizado pelo mestre da *piconet*. Somente o escravo endereçado no *slot* anterior pode transmitir. Isto evita que haja a colisão de pacotes dentro da *piconet*;
- Os escravos devem permanecer com os seus receptores ativos para decodificarem os campos de código de acesso e cabeçalho. Através do código de acesso, descobrirão se o pacote é do mestre da sua *piconet* ou se é do mestre de uma *piconet* adjacente. Através do cabeçalho, descobrirão para qual deles está endereçado o pacote. Os escravos não endereçados desligam o seu receptor e esperam o próximo *slot* par, e reiniciam a decodificação;
- Pacotes de *broadcast* são pacotes recebidos por todos os escravos, porém nenhum deles pode responder no *slot* seguinte;
- Os escravos não podem comunicar-se diretamente ente si. Só há comunicação entre escravos através do mestre;

A operação descrita acima está ilustrada na figura 24.

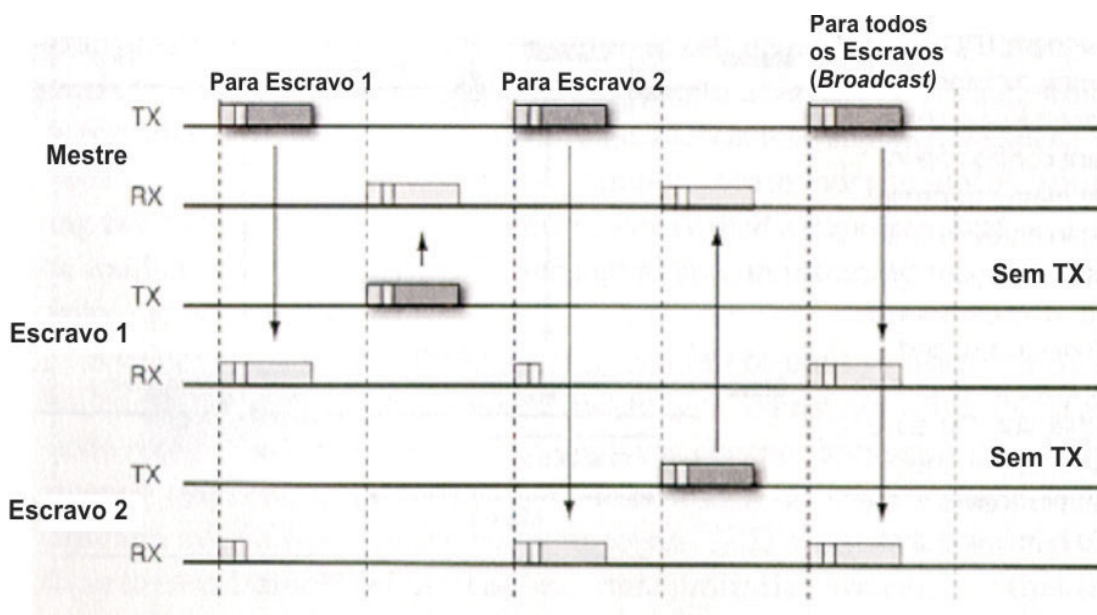


Figura 24. Operação em uma configuração ponto-a-multiponto

2.4.4 Pacotes *multislot*

Como analisado anteriormente, a vazão máxima por nó é reduzida significativamente pelos campos código de acesso e cabeçalho, e pelo tempo de guarda necessário para realizar a troca da frequência de salto. Para uma configuração ponto-a-ponto o mesmo está reduzido a 192 Kbps.

Sendo assim, para otimizar a vazão, os dispositivos Bluetooth podem transmitir pacotes que ocupem 3 ou 5 *slots* (figura 25). Durante a transmissão destes pacotes *multislot*, a frequência de salto é mantida constante e igual à frequência de salto do primeiro *slot*. A frequência de salto só será modificada após o término da transmissão do referido pacote, voltando a seguir a seqüência de salto especificada pelo gerador de seqüências pseudo-aleatórias do dispositivo mestre da *piconet*. Na figura 25, k é par para a transmissão no sentido mestre - escravo, e ímpar para transmissão no sentido escravo - mestre.

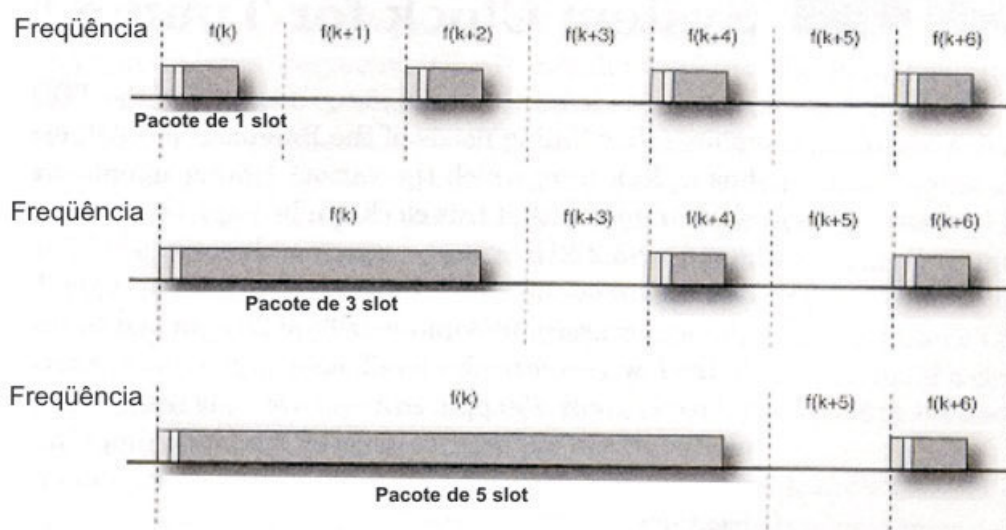


Figura 25. Pacotes *multislot*

A tabela 12 ilustra a otimização do *payload* quando do uso pacotes *multislots*.

Tipo de pacote	Código de Acesso Nominal (bits)	Cabeçalho (bits)	Banda de guarda (bits)	Payload (bits)
1 slot	72	54	259	$625 - 72 - 54 - 259 = 240$
2 slots	72	54	259	$(3 \times 625) - 72 - 54 - 259 = 1490$
5 slots	72	54	259	$(5 \times 625) - 72 - 54 - 259 = 2740$

Tabela 12. Otimização do *payload* devido o uso de pacotes *multislots*

O contraponto desta otimização do *payload* é o fato de que os pacotes maiores são mais suscetíveis a erro. Nestes casos, a vazão é fortemente afetada devido às retransmissões de pacote. Para retransmitir um pacote que ocupa 5 *slots*, despendem-se cinco *slots* para retransmitir o pacote, e mais um para o nó destino enviar a confirmação do recebimento do mesmo.

2.4.5 Temporização da *piconet*

A fim de implementar os requisitos de tempo necessários para a comunicação *full duplex* através do esquema *fast TDD*, conjuntamente com outras temporizações necessárias para a operação da *piconet*, cada dispositivo Bluetooth implementa seu próprio relógio, denominado *clock* nativo (CLKN), donde todos os outros relógios derivam. O CLKN é normalmente gerado por um contador de 28 bits cuja entrada *clock* é controlada por uma onda quadrada de frequência igual a 3.2 KHz. Havendo, então, um incremento deste contador a cada 312.5 μ s, que corresponde à metade do tempo de um

slot. Ele é habilitado tão logo o dispositivo esteja energizado e é reiniciado uma vez por dia. Os principais requisitos deste relógio nativo são:

- Máximo *drift* de 20 ppm, em modo de operação normal;
- *Jitter* máximo de 10 μ s;
- Máximo *drift* de 250 ppm, em modo de operação de baixa potência (*sniff*, *hold* e *park*);

A temporização da *piconet* (CLK) está sob controle do mestre, ou seja, o CLK é igual CLKN do mestre. Sendo assim, todos os escravos devem estar sincronizados com este. Para isso, os escravos devem adicionar *offsets* (valores de ajuste) aos seus relógios nativos até se igualarem ao CLK. Há ainda um relógio denominado CLKE, que corresponde à estimativa do relógio de um dispositivo ao qual se quer iniciar o procedimento de estabelecimento de conexão. Ele é obtido também a partir do CLKN.

2.4.6 Enlaces físicos

Podem ser estabelecidos dois tipos de enlaces físicos em uma *piconet*, a saber: ACL (*asynchronous connectionless*) e SCO (*synchronous connection-oriented*). O primeiro é usado para transferência de dados onde a integridade é mais importante que a latência (intervalo de tempo decorrido entre a transmissão de um pacote e a sua recepção). O segundo, por sua vez, é utilizado para transferência de dados cuja baixa latência é mais importante que a integridade (não podendo exceder a ordem de dezenas de milissegundos), a fim de garantir interatividade, como por exemplo, na transferência de voz bidirecional.

2.4.6.1 Enlace SCO

O enlace SCO pode ser sumarizado da seguinte forma:

- Configuração ponto-a-ponto, envolvendo um mestre e um escravo;
- Utiliza chaveamento de circuitos, como na telefonia convencional, para obter garantia de recursos (banda passante);
- Não usa CRC (*Cyclic Redundancy Check*);
- A baixa latência é garantida por meio da reserva de *slots* e pela proibição de retransmissão dos pacotes;
- Utiliza pacotes com duração de um *slot*;

- Suporta *payload* sem FEC (*Forward Error Correction*) e com FEC com as seguintes razões 1/3 ou 2/3.

Há vários tipos de pacotes SCO. Estes pacotes SCO são trocados em pares, primeiro do mestre para o escravo e, no *slot* seguinte, do escravo para o mestre. Como regra geral, um escravo somente pode transmitir pacotes para o mestre no *slot* previamente reservado, se o mestre não transmitiu pacote SCO para nenhum outro escravo no *slot* anterior.

2.4.6.2 Enlace ACL

O enlace ACL pode ser sumarizado da seguinte forma:

- Configuração ponto-a-ponto ou ponto-a-multiponto;
- Utiliza chaveamento de pacotes, possibilitando o compartilhamento de banda passante com todos os escravos da *piconet*;
- Suporta pacotes *multislot*;
- Suporta *payload* com ou sem FEC (*Forward Error Correction*) com razão de 2/3.
- Utiliza ARQ (*Automatic Repeat Request*), para retransmissão de pacotes corrompidos;
- Latência variável;

O enlace ACL pode também ser usado para transmitir dados isócronos, onde a latência e a integridade são igualmente importantes. Estes canais podem ser utilizados em todos os demais *slots* não reservados pelos canais SCO.

O diagrama da figura 26 exemplifica tráfego ACL e SCO dentro de uma *piconet*.

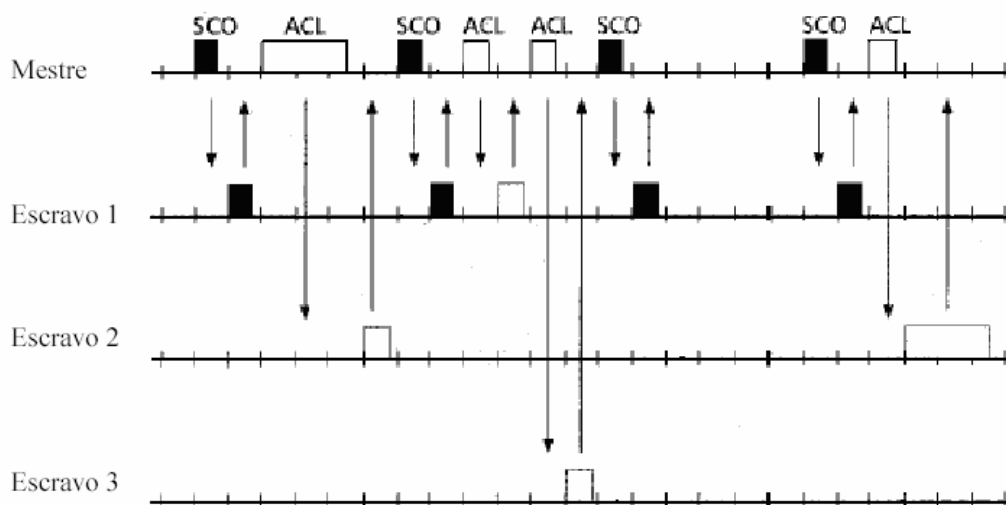


Figura 26. Tráfego ACL e SCO dentro da *piconet*

2.4.7 Endereços e nomes dos dispositivos Bluetooth

Um dispositivo Bluetooth é identificado por endereços. Estes endereços são representados por meio de números binários na forma hexadecimal, podendo haver uma tradução em nomes tal como é realizado pelo DNS da pilha TCP / IP. Os endereços mais importantes são:

- *BD_ADDR* – *Bluetooth Device Address*;
- *AM_ADDR* – *Active Member Address*;
- *PM_ADDR* – *Parked Member Address*;
- *AR_ADDR* – *Access Request Address*.

2.4.7.1 *BD_ADDR* – *Bluetooth Device Address*

Este endereço é único para cada dispositivo, sendo composto por 48 bits que formam três campos (tabela 13), a saber:

- *Lower address part* (LAP) contendo 24 bits;
- *Upper address part* (UAP) contendo 8 bits;
- *Nonsignificant address part* (NAP) contendo 16 bits.

company_assigned	company_id	
LAP (24 bits)	UAP (8 bits)	NAP (16 bits)

Tabela 13. Composição do *BD_ADDR*

O NAP é um campo auxiliar que é usado para tornar o *BD_ADDR* único para cada dispositivo, como também, é usado para dar suporte à aplicações que requerem a transferência segura de informações.

Os demais campos (LAP e o UAP) formam o endereço base e são utilizados principalmente para identificação da *piconet*, para iniciar o processo de estabelecimento de conexão com um dispositivo específico e para a geração do conjunto de canais de salto.

No processo de composição do *BD_ADDR* de um dispositivo há dois atores:

- O grupo IEEE 802, que designa os campos UAP e NAP, conjuntamente denominados de *company_id*, representando um identificador único organizacional (OUI);
- O fabricante, que designa os 24 bits restantes do LAP.

Sendo assim, para cada *company_id*, pode-se ter até 2^{24} (16 milhões) *BD_ADDR* únicos.

2.4.7.2 AM_ADDR – *Active Member Address*

Endereço de três bits dado a cada escravo ativo dentro da *piconet*. Utilizado pelo mestre para não necessitar transmitir um endereço de 48 bits quando do endereçamento de um escravo. Suas principais características são:

- O endereço 000b é reservado para transmissão de pacotes de *broadcast* (pacotes enviados pelo mestre endereçados a todos os escravos ativos);
- As demais sete palavras (de 001b a 111b) são utilizadas para endereçar até sete escravos;
- Escravos que deixam a *piconet* perdem a condição de membro ativo da *piconet* e, assim, perdem também o seu AM_ADDR;
- É o primeiro campo do pacote de cabeçalho.

2.4.7.3 PM_ADDR – *Parked Member Address*

Apesar da limitação de sete escravos ativos dentro da *piconet*, pode-se ter um número maior de escravos em um modo denominado *park*. Neste modo, os escravos não podem compartilhar a vazão da *piconet*, porém mantêm-se sincronizados com a temporização de pacotes e com a seqüência de salto designadas pelo mestre da *piconet*, permitindo que eles periodicamente ouçam pacotes de *broadcast*. Quando um escravo ativo é comutado para o modo *park*, o mestre designa um PM_ADDR para identificá-lo, sendo também utilizado para reativá-lo. O PM_ADDR possui 8 bits e suas principais características são:

- O endereço 00H é reservado para o escravo responder com o seu BD_ADDR quando no procedimento de reativação;
- Os restantes 255 endereços (de 01 H a FF H) são utilizados para identificar até 255 escravos no modo *park*.

2.4.7.4 AR_ADDR - *Access Request Address*

Endereço designado pelo mestre para um escravo que foi colocado em modo *park*. É utilizado pelo escravo para determinar sua janela de acesso, que corresponde à metade da duração de um *slot*, a fim de enviar uma solicitação de reativação.

Esta característica torna a *piconet* ainda mais poderosa, podendo-se ter uma aplicação com até 255 escravos em modo *park*, que poderão estar solicitando sua reativação ao mestre de acordo com a demanda, como, por exemplo, quando desejarem executar a transferência de um arquivo, otimizando sobremaneira o uso da banda passante.

2.4.7.5 BDN - Bluetooth Device Name

O BDN é um nome amigável dado a um dispositivo Bluetooth. Ele possui no máximo 248 bytes, seguindo o padrão UTF-8 (*Unicode Standard Transformation Format Eighth*). O BDN permite que os dispositivos sejam identificados de acordo com a sua funcionalidade, a saber: “Jig de Teste LCD Popeye”, “Impressora Engenharia”, etc.

2.4.8 Formato dos pacotes de banda base

Como falado anteriormente, os pacotes de banda base possuem três campos, a saber, o código de acesso (*access code*), o cabeçalho (*header*) e a carga útil (*payload*), conforme figura 27. Esses pacotes podem conter apenas o código de acessos ou o código de acesso juntamente com o cabeçalho ou o código de acesso juntamente com o cabeçalho e com a carga útil.

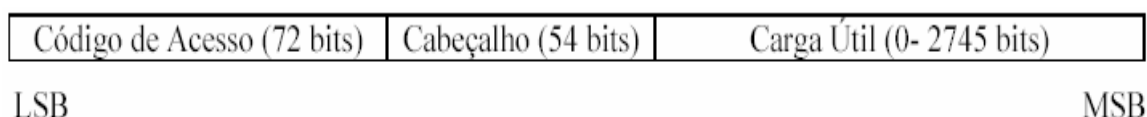


Figura 27. Composição dos pacotes de banda base

2.4.8.1 Código de acesso

A figura 28 ilustra o formato deste campo.

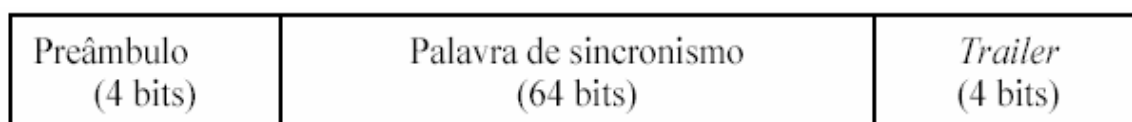


Figura 28. Composição do código de acesso.

O campo preâmbulo juntamente com o primeiro bit do campo palavra de sincronismo, compõe uma seqüência alternada de símbolos 1 e símbolos 0, formando 5 bits. Esta seqüência permite que o receptor faça:

- A compensação DC do detector, ajustando o seu circuito de decisão em relação aos níveis de tensão correspondentes ao valor lógico zero e ao valor lógico um;
- A sincronização do seu relógio nativo (CLKN) em relação ao relógio da *piconet* (CLK);
- A sincronização de bit, já que se trata de uma seqüência previamente conhecida.

O campo *trailer* juntamente com o último bit da palavra de sincronismo, formam uma seqüência alternada de cinco *bits* responsável por preparar o detector do receptor para

decodificação correta do cabeçalho. Portanto, caso não haja cabeçalho, o código de acesso não possuirá o campo *trailer* e terá apenas 68 bits.

Devido a diferentes contextos de operação dentro da *piconet*, há quatro tipos de código de acesso, a saber: o *channel access code* (CAC), o *device access code* (DAC), o *general inquiry access code* (GIAC), *dedicated inquiry access code* (DIAC) e o *limited inquiry access code* (LIAC). Os três últimos são também chamados de *inquiry access code* (IAC). O conteúdo do campo palavra de sincronismo é que identifica o tipo de código de acesso. A palavra de sincronismo é gerada a partir dos 24 bits do LAP (*Lower Address Part*) extraídos do BD_ADDR do contexto aplicado (CAC, DAC ou IAC). Esta seqüência de 24 bits passa por um processo de formação a fim de dotá-la de boas propriedades de autocorrelação e correlação cruzada, obtendo-se ao final uma seqüência pseudo-aleatória de 64 bits. No receptor, a palavra de sincronismo não precisa ser decodificada, pois ela já é de seu conhecimento. Basta que ela, após passar pelo correlator, exceda o limiar esperado. Outra característica da palavra de sincronismo é que a mesma deve ter baixa autocorrelação cruzada com as demais seqüências geradas por outros dispositivos de outras *piconets*.

A correta decodificação do código de acesso é de suma importância para a operação da *piconet*, pois se o mesmo for perdido ou detectado erroneamente, todo o resto do pacote será perdido.

2.4.8.1.1 CAC (*Channel Access Code*)

Este código identifica os pacotes que são trocados dentro de uma *piconet* específica. Deriva do LAP (*Lower Address Part*) do BD_ADDR da unidade mestre. Desta forma, antes de aceitar um pacote, os dispositivos checam se o LAP recebido está de acordo com o LAP da unidade mestre.

Como visto anteriormente, para garantir a exclusividade do BD_ADDR é necessário mais um campo denominado NAP. Sendo assim, o campo LAP sozinho não garante a exclusividade do CAC, podendo haver *piconets* com a mesma identificação de canal coexistindo na mesma área.

2.4.8.1.2 DAC (*Device Access Code*)

Este código é utilizado pela futura unidade mestre para iniciar um estabelecimento de conexão com um determinado dispositivo cujo BD_ADDR já é conhecido, seja por uma conexão anterior ou devido à realização inicial de um procedimento *inquiry* (este

procedimento será descrito no item estabelecimento de conexão). Este pacote não requer cabeçalho, por isso não precisa do campo *trailer*.

2.4.8.1.3 GIAC (*General Inquiry Access Code*)

Este código é utilizado pela futura unidade mestre para descobrir quais dispositivos estão dentro de sua área de cobertura. Estes dispositivos ao receberem este código, entendem que devem montar uma mensagem contendo, dentre outras coisas, o seu BD_ADDR. O GIAC tem o valor 0x9E8B33. Este pacote não requer cabeçalho, por isso não precisa do campo *trailer*.

2.4.8.1.4 DIAC (*Dedicated Inquiry Access Code*)

Este código é utilizado pela futura unidade mestre para restringir sua interrogação à dispositivos que possuem determinada funcionalidade, como uma impressora ou um dispositivo que realiza a substituição de cabos (perfil SPP – *Serial Port Profile*). O DIAC é também chamado de *device-specific inquiry code*. A especificação Bluetooth versão 1.1 prevê sessenta e dois (62) DIACs compreendidos nas faixas de 0x9E8B01 a 0x9E8B32 e de 0x9E8B34 a 0x9E8B3F. Este pacote não requer cabeçalho, por isso não precisa do campo *trailer*.

2.4.8.1.5 LIAC (*Limited Inquiry Access Code*)

Este código é utilizado quando duas unidades (futuro mestre e futuro escravo) desejam acelerar o processo de estabelecimento de conexão. Sendo assim as unidades comutam para os respectivos estados de procura (futuro mestre) e busca (futuro escravo) por um determinado intervalo de tempo, tipicamente 1 minuto, limitando assim o número de respostas ao *inquiry*. O LIAC tem o valor 0x9E8B00. Este pacote não requer cabeçalho, por isso não requer o campo *trailer*.

2.4.8.2 Cabeçalho (*Header*)

O cabeçalho é formado por 10 bits divididos em cinco campos (AM_ADDR, TYPE, FLOW, ARQN e SEQN). A estes bits são adicionados oito bits de código de verificação de redundância cíclica (CRC) a fim de avaliação da integridade do cabeçalho no nó de destino. Estes 18 bits resultantes passam ainda por um codificador que insere redundância na razão 1/3, obtendo-se um total de 54 bits. A figura 29 ilustra o exposto.

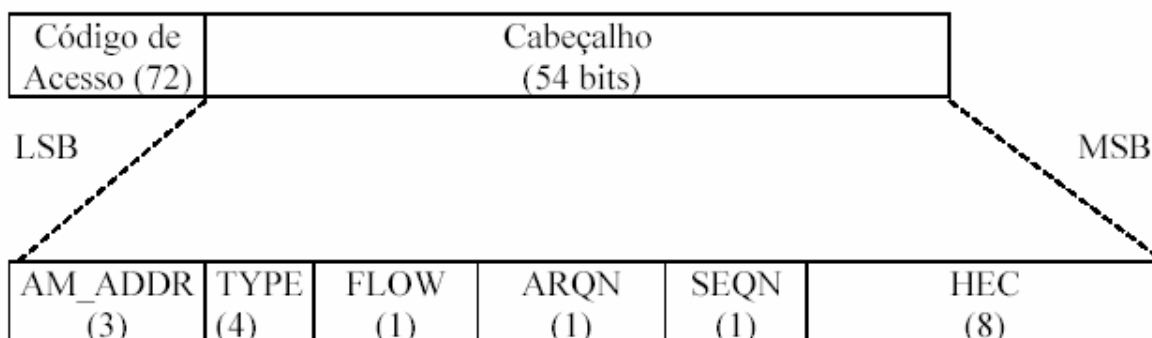


Figura 29. Composição do cabeçalho

Descrição dos campos:

- **AM_ADDR.** Como mencionado no tópico “Endereços e Nomes dos Dispositivos Bluetooth”, este endereço é utilizado para identificar um escravo ativo dentro da *piconet*.
- **TYPE.** Possui quatro bits, podendo então designar até 16 tipos de pacotes diferentes. Estes pacotes estão distribuídos em três categorias, a saber: pacotes referentes a enlaces físicos ACL, pacotes referentes a enlaces físicos SCO e pacotes comuns para ambos os enlaces. A figura 30 detalha o exposto.

Código	slots	Enlace SCO	Enlace ACL
0000	1	NULL	NULL
0001	1	POLL	POLL
0010	1	FHS	FHS
0011	1	DM1	DM1
0100	1	Indefinido	DH1
0101	1	HV1	Indefinido
0110	1	HV2	Indefinido
0111	1	HV3	Indefinido
1000	1	DV	Indefinido
1001	1	Indefinido	AUX1
1010	3	Indefinido	DM3
1011	3	Indefinido	DH3
1100	3	Indefinido	Indefinido
1101	3	Indefinido	Indefinido
1110	5	Indefinido	DM5
1111	5	Indefinido	DM5

Figura 30. Classificação dos pacotes

Abaixo segue uma descrição dos pacotes comuns aos enlaces ACL e SCO. Os demais serão descritos posteriormente.

- **ID ou Pacote de Identificação.** Este pacote é formado apenas pelo código de acesso (DAC ou IAC), sendo assim, não possui codificação na tabela acima. É utilizado para fins de controle e sinalização;
- **Pacote *Null*.** Este pacote é formado pelo código de acesso e pelo cabeçalho, sendo utilizado para envio de informações de controle contidas no cabeçalho, como controle de fluxo e reconhecimento de recepção de pacote;
- **Pacote *Pool*.** Este tem a mesma formação do pacote *null*, sendo utilizado pelo mestre para forçar uma resposta do escravo (procedimento também chamado de *ping*);
- **Pacote FHS (*Frequency Holp Synchronization*).** Este pacote é formado pelos três campos: código de acesso, cabeçalho e *payload*. É principalmente utilizado pelo mestre para informar ao escravo todos os parâmetros necessários para a sincronização dos mesmos, a fim de atender aos requisitos da comunicação usando FHSS.
- **FLOW.** Bit responsável por realizar o controle de fluxo para os pacotes ACL. Caso estoure a fila de recepção de pacotes no nó receptor, o mesmo deve informar ao transmissor que o envio de pacotes deve ser paralisado. Para isso, o mesmo monta um pacote *null* com o bit FLOW zerado, e o envia no seu *slot* reservado dentro da *piconet*. Desta forma, o transmissor só voltará a enviar pacotes quando receber um novo pacote *null* com o bit FLOW igual a 1(um).
- **ARQN.** Bit de reconhecimento de recepção de pacote. Caso o transmissor receba um pacote *null* com este bit igual a 1 (um), significa que o pacote foi recebido com sucesso. Caso contrário significa que o pacote foi perdido e deve ser retransmitido.
- **SEQN.** Bit responsável por evitar a recepção duplicada de pacote. O transmissor complementa este bit a cada transmissão. Caso o receptor detecte dois pacotes consecutivos com o mesmo valor de SEQN um deles deve ser descartado, pois significa que houve duplicação de envio do pacote, motivado por um erro na recepção de um ARQN igual a 1 no nó transmissor.
- **HEC (*Header Error Check*).** Código inserido para permitir a verificação da integridade do cabeçalho.

2.4.8.3 Carga útil (*Payload*)

Finalmente, este é o campo do pacote de banda base responsável por transportar as informações do usuário a serem trocadas dentro da *piconet*. Sua estrutura varia de acordo com o tipo de pacote especificado no campo *type* do cabeçalho.

2.4.8.3.1 *Payload* dos pacotes FHS

Até este ponto, como mencionado no item Rádio Bluetooth, item FHSS, sabemos que a seleção da frequência de salto a ser gerada pelo sintetizador, é realizada pelo bloco gerador de código de salto, por meio de algoritmo pseudo-aleatório. A figura 31 descreve o procedimento.

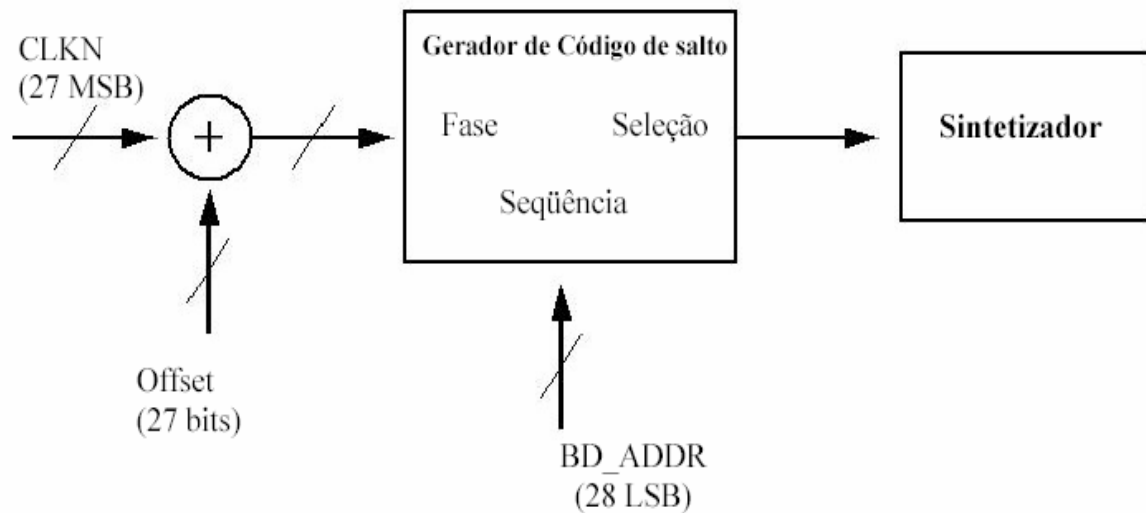


Figura 31. Processo de seleção da frequência de salto

O bloco gerador do código de salto possui duas entradas: seqüência e fase. Através da entrada seqüência (28 bits menos significativos do BD_ADDR da unidade mestre) determina-se a seqüência de salto que está sendo utilizada; e através da entrada fase, garante-se que todos os participantes da *piconet* gerem a seqüência de salto no mesmo instante. Para garantir isto, esta entrada recebe os 27 bits mais significativos do relógio da *piconet* (CLK), que é comum para todos os participantes. Caso o gerador seja da unidade mestre, o CLK é igual ao CLKN, pois o *Offset* é zero; e caso seja de uma unidade escrava $CLK = CLKN + Offset$. Desta forma, garante-se que, em um dado instante, os geradores terão sempre as mesmas entradas e, assim, selecionarão a mesma frequência de salto no bloco sintetizador (através da saída seleção), estabelecendo os requisitos de sincronismo necessários para uma comunicação FHSS.

Neste contexto, o pacote FHS é de grande importância, pois é o meio pelo qual o mestre envia para o escravo, durante o procedimento de *page*, as informações necessárias para a geração sincronizada das frequências de salto quando ambos comutarem para o estado conectado.

O campo *payload* de um pacote FHS possui a estrutura mostrada na figura 32.

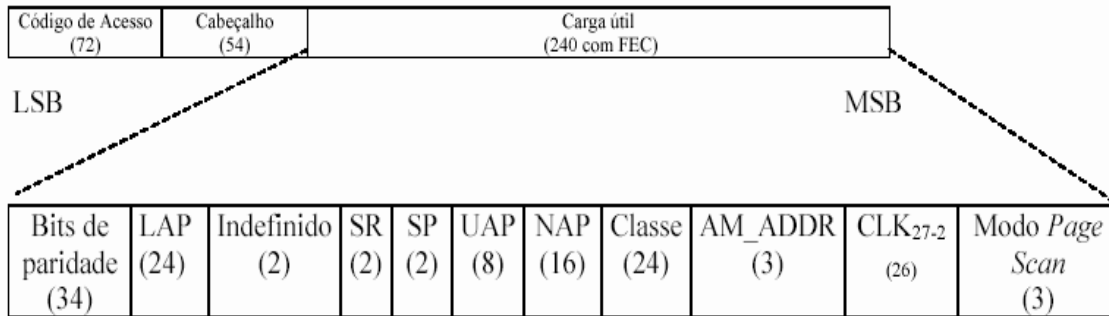


Figura 32. Composição do campo *payload*

Descrição dos campos:

- **Bits de paridade.** São os 34 primeiros bits da palavra de controle contida no código de acesso da unidade que envia o pacote FHS;
- **LAP.** É o LAP do BD_ADDR da unidade que envia o pacote FHS. Com este campo e com o campo bits de paridade é que o nó receptor remonta a palavra de sincronismo do nó transmissor, e assim identifica o contexto do pacote;
- **Indefinido.** Dois bits reservados para aplicações futuras. Devem ser zerados;
- ***Scan Repetition* (SR).** Indica o intervalo de tempo entre duas janelas de *page scan* sucessivas. A tabela 14 ilustra os valores possíveis.

Modo de SR	Valor do SR em FHS	Tempo entre sucessivos <i>page scan</i>
R0	00	Contínuo
R1	01	≤ 1.28 s (valor padrão)
R2	10	≤ 2.56 s

Tabela 14. Modos de *scan repetition*

- ***Scan Period* (SP).** Indica o período de tempo em que o modo de *page scan* mandatário será usado pelo dispositivo. O valor padrão é 11.25 ms;
- **UAP.** É o UAP do BD_ADDR do dispositivo que está enviando o pacote FHS;
- **NAP.** É o NAP do BD_ADDR do dispositivo que envia o pacote FHS;
- **Classe.** Este campo identifica a classe do dispositivo que está enviando o pacote FHS. A classe, por sua vez, indica qual é a funcionalidade do dispositivo. O

documento *Bluetooth Assigned Numbers* do SIG [11], expõe o valor atribuído para cada classe;

- **AM_ADDR.** Como mencionado no tópico “Endereços e Nomes dos Dispositivos Bluetooth”, este campo identifica um escravo ativo dentro da *piconet*. No contexto de transmissão de pacotes FHS, este campo contém 000b quando se tratar de uma resposta a um *inquiry*.
- **CLK₂₇₋₂.** É formado pelos 26 bits mais significativos do relógio nativo da unidade que está enviando o pacote FHS, amostrado no início da transmissão do código de acesso. Esta informação é atualizada a cada 1.25 ms;
- **Modo de *Page Scan*.** Indica qual modo de *page scan* está sendo utilizado pela unidade que está transmitindo o pacote FHS. A tabela 15 descreve os modos possíveis.

b2b1bo	Modo de <i>page scan</i>
000	Mandatário
001	Opcional I
010	Opcional II
011	Opcional III

Tabela 15. Modos de *page scan*

2.4.8.3.2 *Payload* dos pacotes ACL

O *payload* de um pacote ACL possui três campos, a saber: o cabeçalho, os dados e o CRC. A figura 33 ilustra o *payload* para um pacote ACL de um, três e cinco *slots*.

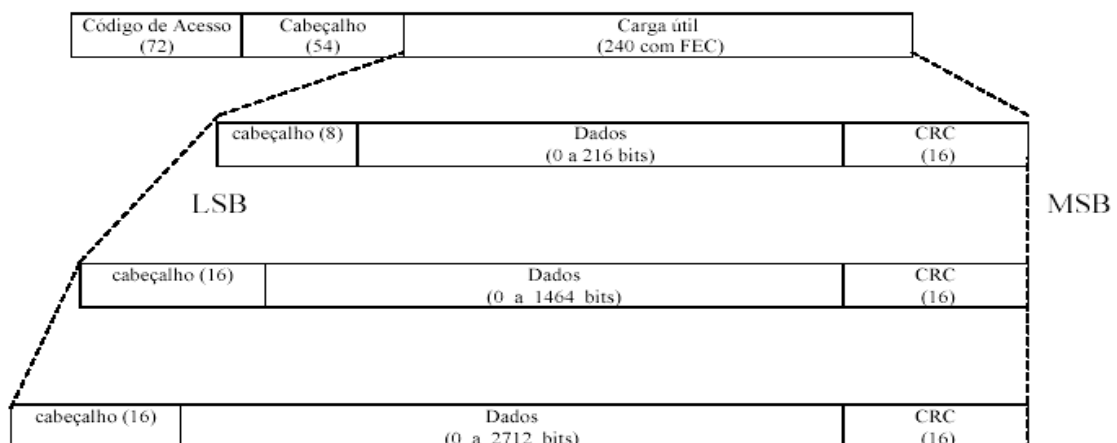


Figura 33. Campo *payload* de um pacote ACL

Os pacotes ACL são identificados por três caracteres, sendo duas letras e um número. A primeira letra é sempre o D, que identifica a transmissão de dados. A segunda letra é H,

para transmissões de alta velocidade, ou M para transmissões de velocidade média. O terceiro caractere é o número 1, 2 ou 5, identificando a quantidade de *slots* que o pacote ocupa. Um pacote é identificado como alta velocidade se não possuir nenhum FEC. Em contrapartida um pacote de velocidade média é caracterizado por possuir a aplicação de FEC cuja razão é 2/3. Assim sendo, o pacote DM3 deve ser interpretado como um pacote ACL (dado), de velocidade média e que ocupa três *slots*.

Há um tipo de pacote ACL especial, o AUX1. Ele possui cabeçalho de 8 bits e não possui FEC, nem CRC. Sua principal utilização é para teste do nível de interferência do canal. Através do mesmo, transmite-se um padrão conhecido para o nó receptor, e este, ao recebê-lo, infere sobre o BER do canal.

2.4.8.3.2.1 Vazão dos pacotes ACL

Neste ponto é importante analisar-se a vazão máxima de um enlace ACL para os diversos tipos de pacotes. Sendo assim, a tabela 16 expõe a vazão obtida em relação ao tipo de pacote e ao tipo de canal. No canal simétrico os dois nós possuem o mesmo tempo de ciclo ou período (mesma banda passante). Já no canal assimétrico, há predominância de um nó (canal direto) em relação ao outro (canal reverso). Vale salientar ainda que para maximizar a vazão foi atribuído ao canal reverso o menor tempo de ciclo (apenas 1 *slot*). Ademais, considerou-se uma taxa de erro de bit igual a zero, significando que o canal foi idealizado.

Canal Simétrico						Canal Assimétrico Direto			Canal Assimétrico Reverso		
Tipo	FEC	CRC	Dado (bits)	Período (µs)	Vazão Mx. (Kbps)	Dado (bits)	Período (µs)	Vazão Mx. (Kbps)	Dado (bits)	Período (µs)	Vazão Mx. (Kbps)
DM1	2/3	SIM	136	1250	108.8	-	-	-	-	-	-
DH1	NÃO	SIM	216	1250	172.8	-	-	-	-	-	-
AUX1	NÃO	NÃO	232	1250	185.6	-	-	-	-	-	-
DM3	2/3	SIM	968	3750	258.1	968	2500	387.2	136	2500	54.4
DH3	NÃO	SIM	1464	3750	390.4	1464	2500	585.6	216	2500	86.4
DM5	2/3	SIM	1792	6250	286.7	1792	3750	477.8	136	3750	36.3
DH5	NÃO	SIM	2712	6250	433.9	2712	3750	723.2	216	3750	57.6

Tabela 16. Vazão máxima de um enlace ACL em relação ao tipo de pacote

Interpretando os dados da tabela 16, chegam-se as seguintes conclusões:

1. A mínima vazão é obtida com os pacotes DM1;
2. A máxima vazão é obtida com pacotes *multislot*;
3. A máxima vazão, tanto para um canal simétrico (433.9 Kbps) quanto para o canal assimétrico (723.9 Kbps), é obtida através do uso de pacotes DH5;

Neste ponto é tentador achar que o uso de pacotes DH5 deve ser a regra de uso geral. Porém, estes resultados são irrealistas, pois consideram uma taxa de erro de bit igual a zero. Na prática, o BER varia na faixa de 10^{-4} a 10^{-2} . A própria especificação de sensibilidade do receptor do rádio Bluetooth é enunciada para um BER de 10^{-3} . Sendo assim, estes valores são importantes apenas para confirmar a premissa, levantada anteriormente no item “Pacotes *Multislot*”, de que o uso de pacotes *multislot* otimiza a vazão. Entretanto, na prática, pacotes com maior duração são mais suscetíveis a erro.

A figura 34 expõe os prováveis valores da vazão para diferentes tipos de pacotes ACL e em função da taxa de erro de bit do canal (BER)

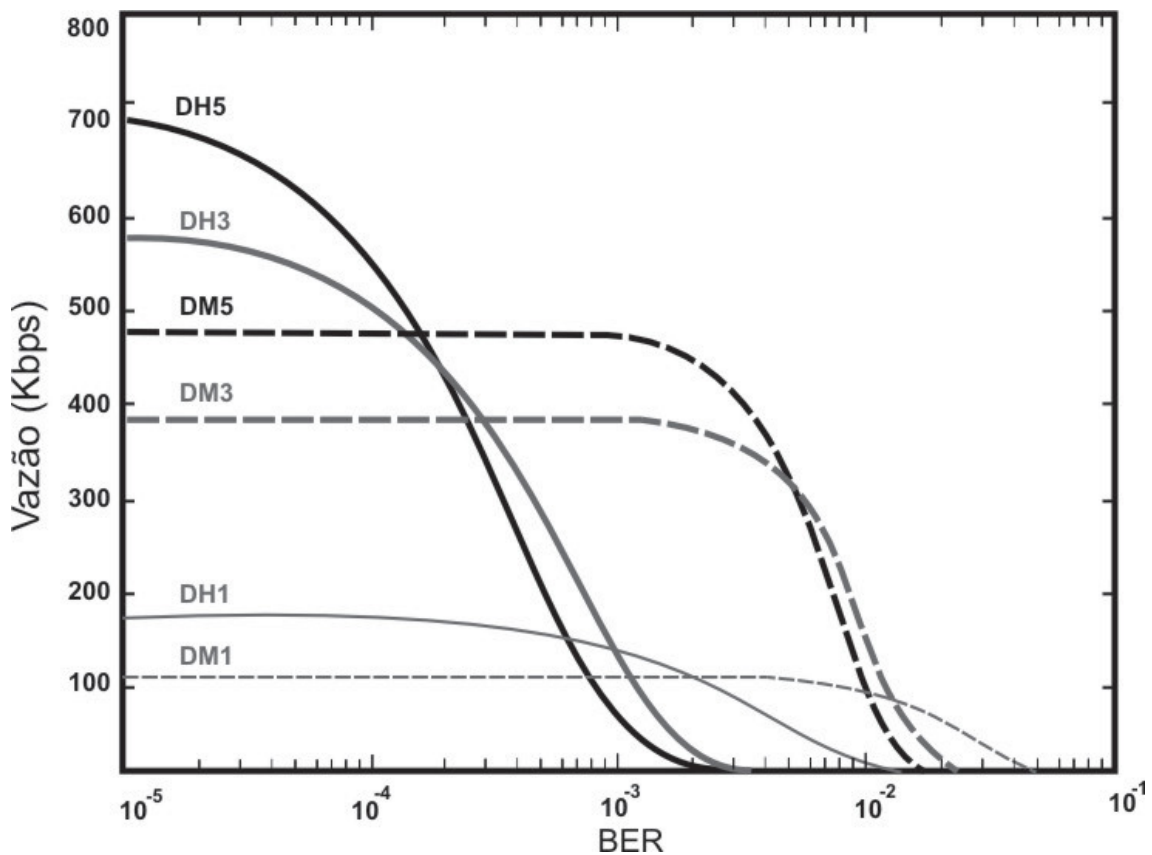


Figura 34. Comportamento da vazão em relação ao BER do canal

Da figura 34 [3], conclui-se que a decisão de qual tipo de pacote deve ser usado depende do BER do canal. Assim:

- Para valores de BER menores que 10^{-4} é preferível o uso de pacotes DH5;
- Para valores de BER próximos de 10^{-3} é preferível o uso de pacotes DM5;
- Para valores de BER maiores que 10^{-2} é preferível o uso de pacotes DM1.

A especificação do Bluetooth versão 1.1 sugere que a escolha do tamanho do pacote a ser usado seja feita dinamicamente após inferir-se sobre a qualidade do canal. Desta

forma, o tamanho do pacote a ser usado é uma consequência da qualidade do canal. Este procedimento é chamado de CQDDR (*Channel Quality Driven Data Rate*).

2.4.8.3.3 Payload dos pacotes SCO

Além do que foi sumarizado no item 2.4.6.1, vale salientar que estes pacotes são utilizados para transmissão de voz bidirecional, em um canal simétrico e em *slots* reservados, cuja taxa de transmissão é de 64 Kbps. Há três tipos de pacotes SCO. Estes são também identificados por três caracteres, formados pelas duas letras HV (significando *high quality voice*), e por um número que pode ser 1, 2 ou 5 (identificando o tipo de correção de erro contida no *payload*). Há ainda um quarto tipo de pacote especial que combina dados ACL e SCO no mesmo *payload*, os chamados pacotes DV (*data / voice*). A tabela 17 sumariza a estrutura destes pacotes.

Tipo	FEC	CRC	Cabeçalho <i>Payload</i> (bits)	Dados <i>Payload</i> (bits)	Taxa de transmissão (Kbps)
HV1	1/3	NÃO	NÃO	800	64
HV2	2/3	NÃO	NÃO	400	64
HV3	NÃO	NÃO	NÃO	267	64
DV	2/3 somente p/ DADO	SIM somente p/ DADO	8 somente p/ DADO)	80 Voz + (0 a 72) Dado	64 Voz + 57.6 Dado

Tabela 17. Composição dos pacotes SCO

2.4.9 Whitening

Antes de qualquer pacote ser transmitido, os campos cabeçalho e *payload*, passam por um processo que embaralha os bits por meio de um LSFR (*linear feedback shift register*) que implementa a função geradora $G(x) = X^7 + X^4 + 1$ (22). Este processo, denominado *whitening*, mistura a ordem dos bits e minimiza o nível DC do pacote. Isto permite uma melhor imunidade à ação de interferência em rajada (curto intervalo de tempo), pois a informação encontra-se pseudo - aleatoriamente distribuída.

A figura 35 ilustra todo o tratamento sofrido pelo pacote antes de ser transmitido, como também o processo de recuperação do mesmo no receptor (os retângulos pontilhados são opcionais).

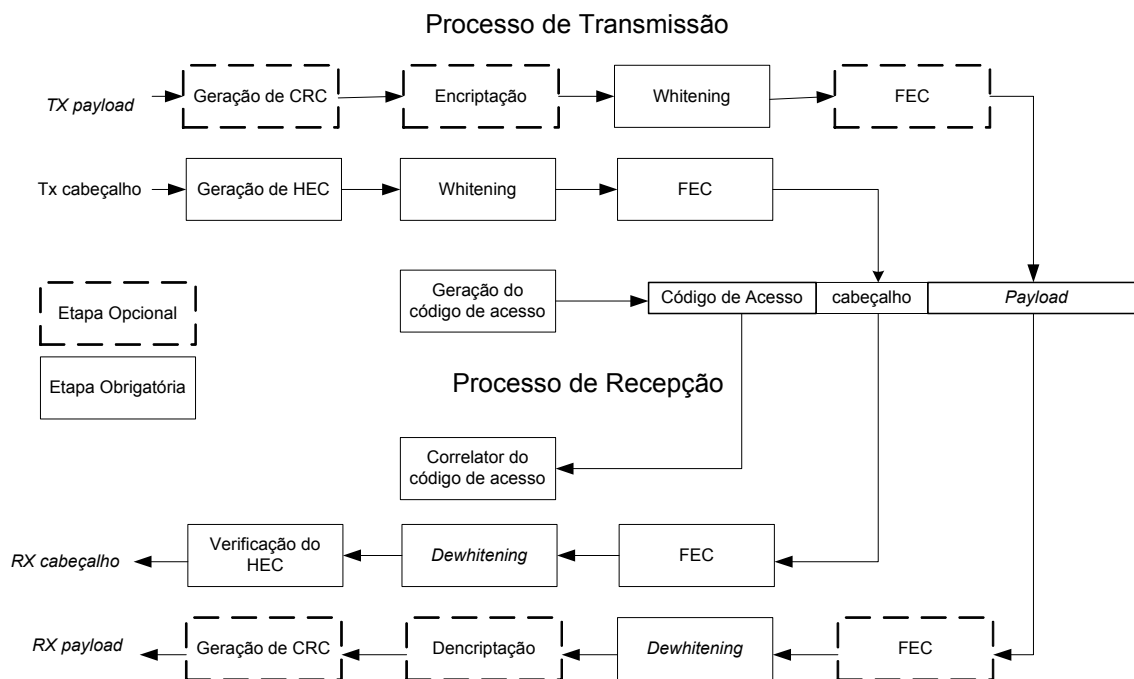


Figura 35. Processo de transmissão e recepção de um pacote

2.4.10 Canais lógicos

Os enlaces físicos já mencionados anteriormente (ACL e SCO) podem ser usados de diferentes maneiras, dependendo do contexto aplicado, formando-se assim os chamados canais lógicos. Existem cinco canais lógicos, a saber:

- **LC (*Link Controller*)**. Neste canal trafegam informações de controle dos protocolos de baixo nível, a saber: ARQ, controle de fluxo, controle de seqüência de pacotes e tipo de *payload*;
- **LM (*Link Manager*)**. Neste trafegam informações trocadas entre o gerente de enlace (*link manager*) dos dispositivos Bluetooth dentro de uma *piconet*;
- **UA (*User Asynchronous*)**. Este canal transporta os dados do usuário quando a integridade dos mesmos é mais importante que a latência. Exemplo: transferência de arquivos;
- **US (*User Synchronous*)**. Este canal transporta dados do usuário onde a baixa latência é mais importante que a integridade da informação. Exemplo: transferência de voz bidirecional;

- **UI (*User Isochronous*)**. Este canal transporta dados do usuário onde a latência e a integridade dos mesmos tem igual importância. Exemplo: transmissão de áudio (unidirecional).

2.5 Estabelecimento de conexão

2.5.1 Descrição geral

O estabelecimento de conexão entre dispositivos Bluetooth está limitado pelos recursos implementados em seus *chipsets* (conjunto de características funcionais). Há dispositivos que suportam apenas a formação de *piconets* ponto-a-ponto, outros, mais recentes, suportam também a formação de *piconets* ponto-a-multiponto, como também, o estabelecimento de enlaces físicos entre *piconets* (*scatternet*).

No processo de estabelecimento de conexão, os dispositivos Bluetooth promovem o que se pode caracterizar como uma “dança”, pois ambos os dispositivos têm de estar em determinados estados e as transições têm de ser sincronizados. Em outras palavras, para que haja dança é necessário que ambos queiram dançar e, em seguida, as suas ações têm de ser sincronizadas, quando um for para a direita o outro tem também de ir. Explicitamente, para receber uma solicitação de estabelecimento de conexão o dispositivo tem de estar ouvindo esta solicitação, e para que se inicie o processo, o outro dispositivo tem de enviar a solicitação neste instante.

Neste item serão descritos diagramas de estados explicitando o processo de estabelecimento de conexão, primeiro para dispositivos que apenas suportam a formação de *piconets* ponto-a-ponto, e depois, para dispositivos ditos versáteis. Em seguida, serão descritos os detalhes funcionais dos processos de *inquiry* e *page*.

2.5.2 Diagrama de estados para uma *piconet* ponto-a-ponto

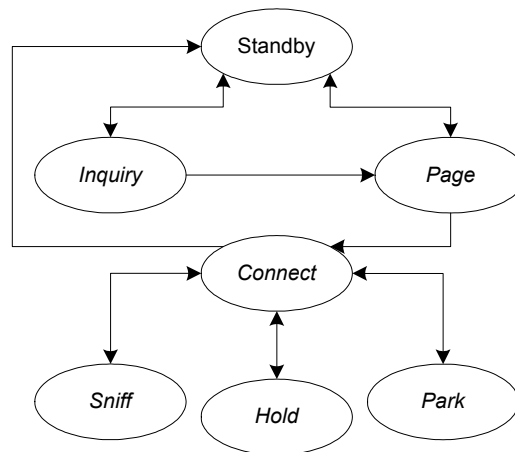


Figura 36. Diagrama de estados - configuração ponto-a-ponto.

Detalhes de operação da figura 36:

- **Estado inicial.** Como qualquer dispositivo eletrônico ao ser ligado, seu estado inicial é *standby*. Neste estado ocorre a inicialização do seu hardware e do seu software;
- **Operação do futuro mestre.** Do estado inicial, o dispositivo só sairá caso deseje estabelecer conexão com outro dispositivo. Sendo assim, o futuro mestre, assim denominado porque a *piconet* ainda não foi estabelecida, pode comutar para o estado de *inquiry* ou para o estado de *page*. Irá para o estado de *inquiry*, caso não conheça o BD_ADDR do futuro escravo. Caso contrário irá para o estado *page*. No estado de *inquiry* o futuro mestre solicita que todos ou parte dos dispositivos Bluetooth dentro de sua área cobertura, respondam com um pacote FHS. Desta forma, o mesmo descobre, dentre outras coisas, o BD_ADDR, o CLKN, a classe, o modo de *page scan*, o modo SR (*Scan Repetition*) e o SP (*Scan Period*) dos dispositivos vizinhos, podendo então se sincronizar mais facilmente com um deles caso confirme a intenção de estabelecer uma *piconet*. Caso contrário, o mesmo pode retornar para o estado de inicial. No estado de *page*, o futuro mestre restringe a interrogação ao futuro escravo por meio do envio de um pacote ID, contendo o DAC do mesmo. Neste instante, a entrada fase do gerador de código de saltos do futuro mestre tem como sinal o CLKE (estimativa do CLKN do futuro escravo), e na entrada seqüência tem o BD_ADDR do futuro escravo. Os processos de *page* e *inquiry* serão descritos em detalhes posteriormente.

- **Operação do futuro escravo.** Periodicamente, este dispositivo tem de sair do estado inicial (*standby*) para ouvir mensagens de um futuro mestre no estado de *inquiry* ou *page*.
- **Conexão.** Caso o processo de *page* obtenha sucesso, ambos comutam para o estado *connect*, tornando-se mestre e escravo da *piconet*. Do estado de conectado, o escravo pode negociar com o mestre para entrar em um modo de baixa potência, a saber: *sniff* ou *hold* ou *park*.
- **Novos escravos.** Pelo diagrama de estados acima, vê-se claramente que uma vez estabelecida a *piconet* ponto-a-ponto, novos escravos não podem ser adicionados, pois não há como o mestre (estado *connect*) comutar para o estado de *inquiry* ou *page*. O único caminho válido é a desconexão, retornando para o estado inicial (*standby*), e assim recuperando a possibilidade de conectar-se a um outro escravo, como descrito anteriormente.

2.5.3 Diagrama de estados dos dispositivos versáteis

Como caracterizado anteriormente, dispositivos versáteis são aqueles que suportam a formação de *piconets* ponto-a-ponto, ponto-a-multiponto, como também o estabelecimento de enlaces entre *piconets* (*scatternet*). Sendo assim, as transições do estado conectado para os estados de *inquiry* ou *page* são plenamente suportadas. A figura 37 ilustra o diagrama de estado dos mesmos. Para simplificar, foram omitidos os estados de baixa potência, porém estes dispositivos os suportam.

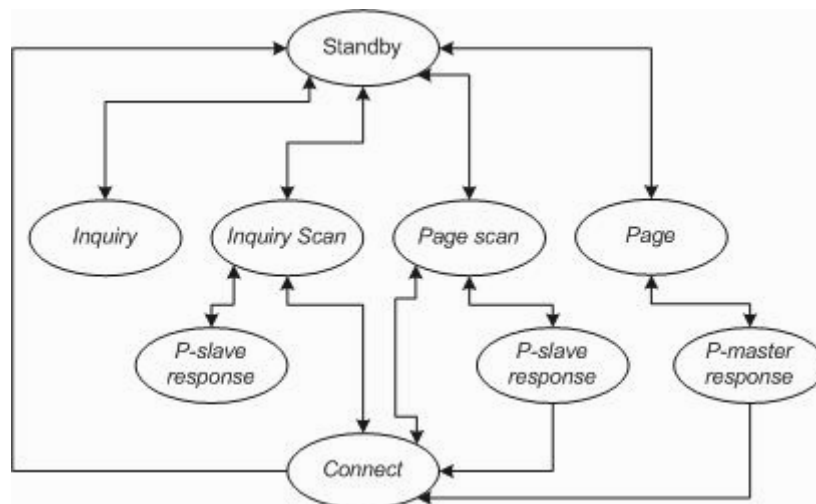


Figura 37. Diagrama de estados para os dispositivos versáteis.

Detalhes de operação da figura 37:

- **Estado inicial.** Não há alteração em relação à configuração ponto-a-ponto.

- **Operação do futuro mestre.** A partir do estado inicial, o futuro mestre pode assumir os seguintes estados:
 - *Page*. Usado para estabelecer conexão com um futuro escravo cujo BD_ADDR é conhecido;
 - *Inquiry*. Usado para descobrir o BD_ADDR e o CLKN dos dispositivos vizinhos dentro de sua área de cobertura;
 - *P-master response*. Depois de receber um pacote DAC do futuro escravo, o futuro mestre monta um pacote FHS, contendo o seu CLKN e o seu BD_ADDR, a fim de que o futuro escravo ajuste seus parâmetros para uso quando da transição para o estado *connect*.
- **Operação do futuro escravo.** A partir do estado inicial, o futuro escravo pode assumir os seguintes estados:
 - *Page Scan*. Neste estado o futuro escravo varre o espectro em busca de um futuro mestre no estado de *page* que esteja enviando o seu DAC;
 - *Inquiry Scan*. Neste estado o futuro escravo varre o espectro em busca de um futuro mestre no estado *inquiry*;
 - *P-slave response* depois do *Page Scan*. Neste estado o futuro escravo responde à solicitação de *page* do futuro mestre, enviando-lhe o mesmo pacote DAC recebido no estado *page scan*;
 - *P-slave response* depois do *Inquiry Scan*. Neste estado o futuro escravo responde a solicitação de *inquiry* do futuro mestre enviando-lhe um pacote FHS contendo seu relógio nativo CLKN, BD_ADDR, sua classe e o modo de *page scan* utilizado.
- **Conexão.** A conexão é finalmente estabelecida após a confirmação da chegada do pacote FHS e, assim, o futuro mestre e o futuro escravo comutam, respectivamente, do estado de *P-master response* e *P-slave response* depois do *Page Scan*, para o estado *connect*.
- **Novos escravos.** O mestre da *piconet* pode comutar para o estado *inquiry* ou *page* a fim de adicionar novos escravos. Os escravos podem comutar para os estados *inquiry scan* e *page scan* para serem adicionados em *piconets* adjacentes, formando assim uma *scarternet*.
- **Novas piconets.** Novas *piconets* só podem ser formadas por futuras unidades mestre. Sendo assim há dois caminhos:

- Por dispositivos no estado *standby*;
- Ou por escravos que comutem do estado *connect* para o estado *page* ou *inquiry*, habilitando-se a serem possíveis mestres de outras *piconets*.

2.5.4 Dificuldades do processo de estabelecimento de conexão

A técnica de espalhamento na frequência do Bluetooth, FHSS, dá aos dispositivos a capacidade de saltar por 79 canais de 1 MHz dentro do espectro da banda ISM a uma taxa de 1600 saltos por segundo, tornando-os bastante imune à interferência. Porém esta realidade refere-se ao estado *connect* (conectado), onde os dispositivos já são nomeados como mestre e escravo. Nos estados anteriores, quando do processo de estabelecimento de conexão, este espalhamento dificulta sobremaneira a sincronização entre o futuro mestre e o futuro escravo, principalmente se os mesmos não apresentarem nenhum histórico de conexões anteriores, pois estarão saltando independentemente de acordo com seus próprios CLKN e BD_ADDR. Sendo assim, um futuro mestre tem de descobrir em qual frequência de salto um futuro escravo estará ouvindo em um dado *slot* para que o mesmo possa ouvir a sua solicitação.

A especificação Bluetooth 1.1 estabelece uma solução para este problema especificando seqüências de salto diferentes para cada estado do dispositivo – *connect*, *inquiry*, *page*, *inquiry scan* e *page scan*, bem como dobrando a taxa de saltos (3200 saltos por segundo) do dispositivo futuro mestre nos estados de *inquiry* e *page*. Cada seqüência de saltos é periódica possuindo 32 canais de salto e período igual a 32. Como a taxa de saltos nesta situação é de 3200 saltos por segundo, bastam 10 ms para um futuro mestre percorrê-la. Ademais, vê-se claramente que o processo de estabelecimento de conexão é facilitado sobremaneira, pois não há repetição de canal de salto dentro do período, ao contrário do que se vê quando os dispositivos já estão conectados. Neste cenário, uma vez que o futuro escravo recebe um pacote, o sincronismo de canal de salto é automaticamente conseguido, pois o próximo canal de salto já é sabido, haja vista que a seqüência é conhecida por ambos, bastando, então, que os dispositivos alcancem o sincronismo de fase.

A tabela 18 detalha os sinais de entrada do gerador de códigos de saltos, usadas para obter as diferentes seqüências de salto para cada estado de operação do dispositivo, a saber: *connect*, *inquiry*, *page*, *inquiry scan* e *page scan*.

Estado de operação	Dispositivo	Quantidade de canais / Período	Gerador de Códigos de salto Entrada: seleção	Gerador de códigos de salto Entrado: fase
<i>Connect</i>	Mestre	$79/2^{27}$	BD_ADDR do mestre	CLKN = CLK
<i>Connect</i>	Escravo	$79/2^{27}$	BD_ADDR do futuro mestre	CLKN + <i>Offset</i> = CLK
<i>Page</i>	Futuro mestre	32/32	BD_ADDR do futuro escravo	CLKN + <i>Offset</i> ' = CLKE
<i>Page scan</i>	Futuro escravo	32/32	BD_ADDR do futuro escravo	CLKN
<i>Inquiry</i>	Futuro mestre	32/32	GIAC (0x9E8B33)	CLKN
<i>Inquiry scan</i>	Futuro escravo	32/32	GIAC (0x9E8B33)	CLKN

Tabela 18. Geração das seqüências de salto de um dispositivo Bluetooth

2.5.5 Descrição geral do processo de *Page*

A especificação Bluetooth 1.1 prevê vários esquemas de realização do processo de *page*. Porém, há um dito mandatário, que todos os dispositivos tem que implementar. Este processo, já explanado nos itens anteriores, pode ser sumarizado mais detalhadamente da seguinte forma:

- O futuro mestre, no estado *page*, envia dois DACs a cada 312.5 μ s dentro de *slots* pares, um no início e o outro no meio, cobrindo assim duas freqüências de salto dentro das 32 que compõem a seqüência de salto especificada para o estado de *page* (ver tabela 18). Neste ponto, é necessário pontuarmos a respeito das características do transmissor para atender à dupla transmissão dentro de 625 μ s do *slot*. Para mudar de freqüência de salto, como exposto anteriormente, são necessários 259 μ s. Porém, no estado de *page* o transmissor somente terá 244.5 μ s, pois 68 μ s são necessários para transmitir o DAC (68 bits). Sendo assim, o transmissor deve ser capaz de atender a este requisito para implementar o processo de *page*;
- Em seguida, nos *slots* ímpares, o futuro mestre sintoniza seu receptor consecutivamente nas duas correspondentes freqüências de salto ao qual o futuro escravo responderá caso haja ouvido o seu DAC;

- Em contrapartida, o futuro escravo, no estado de *page scan*, sintoniza seu receptor em uma das 32 frequências da seqüência de salto especificada para o estado de *page scan* (ver tabela 18) por 11 ms a cada vez que entra neste estado, em busca de seu DAC. Aqui, vê-se que há duas incertezas neste processo:
 - O futuro mestre desconhece o instante em que o futuro escravo estará no estado de *page scan*;
 - O futuro mestre trabalha com uma estimativa do relógio (CLKE) do futuro escravo. A título de análise, esta imprecisão é diretamente proporcional ao intervalo de tempo entre conexões consecutivas, haja visto que os relógios envolvidos são independentes e a imprecisão aumenta de 20 ppm para 250 ppm quando o dispositivo sai do estado conectado para um dos estados de baixa potência, aumentando ainda mais em se tratando de desconexão;
- Ao ouvir seu DAC, o futuro escravo comuta para o estado *p-slave response* e, 625 μ s mais tarde, transmite o mesmo DAC recebido na próxima frequência de salto (canal), seguindo a seqüência de salto especificada para o estado de *page scan*;
- Ao receber o DAC do futuro escravo, o futuro mestre comuta para o estado *p-master response*, salta para o próximo canal que compõe a seqüência de salto especificada para o estado de *page*, e envia de volta um pacote FHS contendo o seu BD_ADDR e seu CLKN para que o escravo se ajuste quando do estabelecimento da *piconet*;
- Em se confirmando a recepção do pacote FHS, ambos comutam para o estado *connect* (conectado).

Os estágios descritos anteriormente são representados na figura 38.

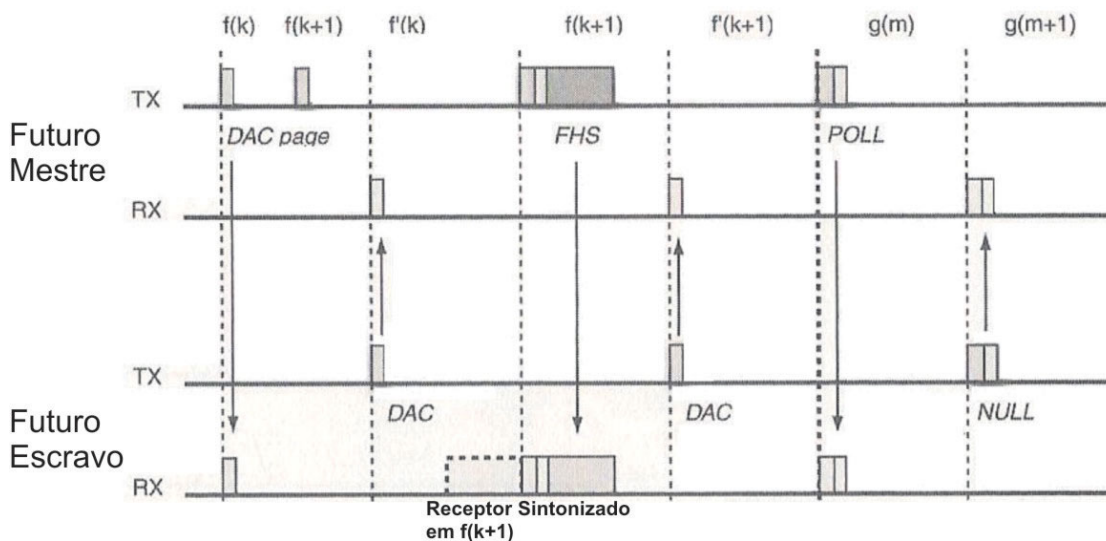


Figura 38. Processo de *page* no domínio do tempo

Descrição da figura:

- No canal representado por $f(k)$ o futuro mestre envia o primeiro DAC. Este pacote tem a duração de 68 μs (68 bits);
- Após 244.5 μs , no pior caso, o futuro mestre comuta seu transmissor para o canal representado por $f(k+1)$ e envia o segundo DAC;
- Agora, após 244.5 μs , o futuro mestre sintoniza seu receptor para ouvir uma possível resposta. Primeiro para $f'(k)$ que é a frequência correspondente ao canal $f(k)$;
- Como, em $f(k)$ o futuro escravo encontrava-se no estado de *page scan*. O mesmo recebeu o seu DAC e, após 625 μs em relação ao seu CLKN, em $f'(k)$, o mesmo envia o mesmo pacote DAC recebido;
- Sendo assim, em $f'(k)$ o futuro mestre recebe o pacote DAC do futuro escravo, afere o *offset* do seu gerador de código de salto a fim de alcançar a sincronização em fase com o futuro escravo e, em $f(k+1)$, envia um pacote FHS contendo suas informações de BD_ADDR e CLKN. Com o BD_ADDR o futuro escravo montará o CAC para envio e recepção de pacotes dentro da *piconet*. Com a informação CLKN, o mesmo ajustará o *offset* do seu gerador de código de salto a fim de obter a sincronização de fase;
- Em $f'(k+1)$, após receber o pacote FHS, o futuro escravo confirma o recebimento por meio do envio de um pacote DAC;
- E, finalmente, após a recepção do DAC, o futuro mestre comuta para o estado de conectado, tornando-se mestre, e em $g(m)$ inicia um procedimento de *ping*, enviando um pacote *poll* para saber se o futuro escravo também comutou devidamente para o estado *connect* (conectado);
- E, em $g(m+1)$, o escravo informa o recebimento correto do pacote *poll* através do envio de um pacote *null* contendo o bit ARQN igual a 1 (um).

A figura 38 acima ilustra a situação em que o futuro escravo recebe o pacote DAC na primeira metade do *slot* par. Caso isto ocorra na segunda metade, o futuro mestre receberá a primeira resposta do escravo na frequência correspondente a $f(k+1)$. O restante do processo ocorre normalmente. Neste ponto vale reforçar o conteúdo da tabela 18, o futuro escravo salta segundo a seqüência estabelecida pelo estado *page scan* e o futuro mestre segundo a seqüência estabelecida pelo estado *page*.

Dois parâmetros definem o comportamento do futuro escravo no procedimento de *page*, a saber:

- $T_{w_page_scan}$. Que é o intervalo de tempo em que o futuro escravo ficará no estado *page scan*. O valor padrão é de 11.25 ms. No estado *page*, o futuro mestre despende 625 μ s, no *slot* par, para enviar dois DACs do futuro escravo e mais 625 μ s, no *slot* ímpar, para ouvir à resposta do futuro escravo, totalizando dois canais de salto a cada 1.25 ms. Descontando 1.25 ms de $T_{w_page_scan}$ devido à incerteza dos relógios, o mesmo pode então varrer $\frac{10}{1.25} \times 2 = 16$ canais de salto em 11.25 ms.
- T_{page_scan} . Que é o tempo entre dois *page scan* sucessivos. Como exposto no item “*Payload* dos Pacotes FHS”, seu valor depende do modo de SR escolhido através do campo SR do pacote FHS. Para o modo R0 a varredura é contínua, para o modo R1 o intervalo é ≤ 1.28 s, e para o modo R2 o intervalo é ≤ 2.56 s.

2.5.5.1 Detalhes do processo de *page*

Com a estimativa do relógio do futuro escravo (CLKE), o futuro mestre monta um trem contendo as 16 primeiras freqüências de salto com maior probabilidade de o futuro escravo estar quando comutar para o estado de *page scan*. Este trem é denominado trem A, sendo o trem B formado pelas 16 outras freqüências de salto. O futuro mestre inicia o processo de *page* pelo trem A, varrendo as 16 freqüências ciclicamente até que receba uma resposta ou haja o estouro de uma temporização. Em ocorrendo o estouro de tempo, o futuro mestre comuta para o trem B, varrendo-o ciclicamente até receber uma resposta ou até um novo estouro de tempo ocorra. Caso o ciclo seja quebrado pelo estouro de tempo, o futuro mestre retorna para o trem A varrendo-o ciclicamente. Caso não haja nenhuma resposta, após o estouro de outra temporização, o processo é abandonado e o usuário é avisado que o processo de *page* não obteve êxito. Esta última temporização pode ser configurada pelo usuário.

O pior caso ocorre quando o futuro mestre inicia a varredura quando o futuro escravo acaba de sair do estado *page scan*. Em se tratando do modo R2, o mesmo só retornará após 2.56 s. Neste intervalo, o futuro escravo varrerá 256 vezes 16 freqüências de salto e não receberá resposta alguma. Desta forma, conclui-se que o futuro mestre deverá permanecer mais de 2.56 s varrendo ciclicamente o trem A

para garantir que o futuro escravo já retornou para o estado *page scan*, antes de comutar para o trem B. Caso não haja sucesso no trem A, significa que o futuro escravo está com sua janela sintonizada em uma das frequências do trem B. Daí, após comutar para o trem B, serão necessários, no pior caso, 10 ms a mais para que o futuro mestre obtenha o sincronismo de canal com o futuro escravo, considerando-se um canal com BER de zero. Para o modo R1, o futuro mestre deverá permanecer um pouco mais de 1.28 s varrendo ciclicamente o trem A para garantir que o futuro escravo já retornou para o estado *page scan*, antes de comutar para o trem B, também considerando um cenário livre de ruído.

2.5.6 Descrição geral do processo de *Inquiry*

Este processo é usado por um futuro mestre para descobrir quais dispositivos estão dentro de sua área de cobertura. Ao contrário do processo de *page*, o futuro mestre não conhece o *BD_ADDR*, o *CLKN*, a classe, nem o modo de *page scan* utilizado pelos escravos em potencial. Portanto, o mesmo não tem como saber a seqüência de canais salto, nem com o instante em que ela é gerada (fase). Sendo assim, o processo de *inquiry* implementa um esquema especial para descobrir estes parâmetros a fim de serem usados em um posterior processo de *page*, caso haja a necessidade estabelecimento de conexão com um destes escravos em potencial.

Os pré-requisitos do processo *inquiry*:

- O estado de operação dos dispositivos. O futuro mestre e o futuro escravo devem estar simultaneamente no estado *inquiry* e *inquiry scan*;
- Ambos têm de usar a mesma seqüência de canais de salto (ver tabela 18);
- Um mecanismo que evite colisão de pacotes (*backoff*) deve ser implementado. Como todos utilizam uma seqüência de salto comum, caso dois dispositivos, previamente no estado *inquiry scan*, respondam a solicitação de *inquiry* no mesmo instante ocorrerá colisão dos mesmos e, conseqüentemente, perda de pacotes;

Neste contexto, o processo de *inquiry* pode ser sumarizado da seguinte forma:

- O futuro mestre, no estado *inquiry*, envia dois pacotes ID contendo, dependendo do contexto, um GIAC, um DIAC ou um LIAC a cada 312.5 μ s dentro de *slots* pares, um no início e o outro no meio, cobrindo assim duas frequências de salto dentro das 32 que compõem a seqüência de salto especificada para o estado de *inquiry* (ver tabela 18).

- Em seguida, nos *slots* ímpares, o futuro mestre sintoniza seu receptor consecutivamente nas duas correspondentes freqüências de salto ao qual o futuro escravo responderá caso haja recebido o IAC;
- Em contrapartida, o futuro escravo, no estado de *inquiry scan*, sintoniza seu receptor em uma das 32 freqüências da seqüência de salto especificada para o estado de *page scan* (ver tabela 18) por 11 ms a cada vez que entra neste estado, em busca de um IAC. Aqui, vê-se que há duas incertezas neste processo:
 - O futuro mestre desconhece o instante em que o futuro escravo estará no estado *inquiry scan*;
 - O futuro mestre e o futuro escravo geram a seqüência de salto sincronizada com seus relógios nativos, e estes não estão sincronizados entre si.
- Ao ouvir o IAC, o futuro escravo, a fim de evitar colisões, inicia um processo aleatório para evitar a colisão de pacotes (*backoff*), seguindo os seguintes passos:
 - Escolhe um número RAND de 10 bits de uma seqüência aleatória uniformemente distribuída, podendo assumir os valores de 0 a 1023;
 - Retorna ao seu estado anterior, *connect* ou *standby*, por um tempo corresponde a RAND *slots*, o que corresponde a um atraso de 0 a 639 ms. Após o término do tempo, o futuro escravo retorna para o estado de *inquiry scan*, na mesma freqüência anterior, habilitando-se a receber outro IAC;
- Após receber o segundo IAC, idêntico ao anterior, o futuro escravo comuta para o estado *p-slave response* e, 625 μ s mais tarde, transmite um pacote FHS na próxima freqüência de salto (canal), seguindo a seqüência de salto especificada para o estado de *inquiry scan*;
- O futuro mestre ao receber o pacote FHS não responde para o futuro escravo. Ele simplesmente armazena os dados do mesmo em sua base de dados para utilizá-los num futuro processo de *page*.

A figura 39 ilustra o este processo.

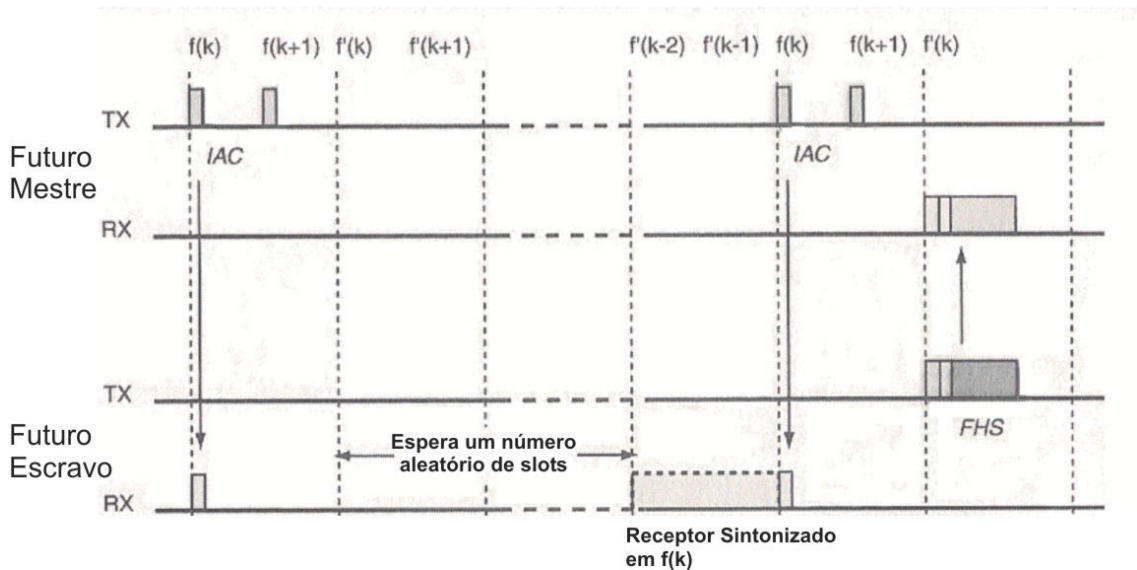


Figura 39. Processo de *inquiry* no domínio do tempo.

Descrição da figura:

- No canal representado por $f(k)$ o futuro mestre envia o primeiro IAC. Este pacote tem a duração de $68 \mu\text{s}$ (68 bits);
- Após $244.5 \mu\text{s}$, no pior caso, o futuro mestre comuta seu transmissor para o canal representado por $f(k+1)$ e envia o segundo IAC;
- Agora, após $244.5 \mu\text{s}$, o futuro mestre sintoniza seu receptor para ouvir uma possível resposta. Primeiro para $f(k)$ que é a frequência correspondente ao canal $f(k)$;
- Como em $f(k)$ o futuro escravo encontrava-se no estado de *inquiry scan*, o mesmo recebeu um IAC e, então, iniciou o procedimento de *backoff*.
- Após o estouro de *RAND slots*, o futuro escravo novamente retorna para o estado *inquiry scan* com seu receptor sintonizado em $f(k)$;
- Neste instante o futuro mestre envia o segundo IAC. Ao recebê-lo, o futuro escravo, após $625 \mu\text{s}$ em relação ao seu CLKN, responde com um pacote FHS.

Neste ponto vale reforçar o conteúdo da tabela 18, o futuro escravo salta segundo a seqüência estabelecida pelo estado *inquiry scan* e o futuro mestre segundo a seqüência estabelecida pelo estado *inquiry*.

Dois parâmetros definem o comportamento do futuro escravo no processo de *inquiry*, a saber:

- $T_{w_inquiry_scan}$. Que é o intervalo de tempo em que o futuro escravo ficará no estado *inquiry scan*. O valor padrão é de 11.25 ms. Como no estado *page*, o

futuro escravo despende 625 μ s, no *slot* par, para enviar dois DACs do futuro escravo e mais 625 μ s, no *slot* ímpar, para ouvir à resposta do futuro escravo, totalizando 2 canais de salto a cada 1.25 ms. Descontando 1.25 ms $T_{w_inquiry_scan}$ devido a incerteza dos relógios, o mesmo pode então varrer $\frac{10}{1.25} \times 2 = 16$ canais de salto em 11.25 ms.

- $T_{inquiry_scan}$. Que é o tempo entre dois *inquiry scan* sucessivos. O valor padrão é 1.28 s;

Em decorrência do $T_{w_inquiry_scan}$, como acontece no processo de *page*, o futuro mestre monta também dois trens contendo 16 frequências de salto, a saber o trem A e o trem B. Como o futuro mestre desconhece o CLKN do futuro escravo, esta escolha é totalmente aleatória. E, em decorrência do $T_{inquiry_scan}$, o futuro mestre permanece por 2.56s em cada trem para cobrir dois sucessivos *inquiry scan*.

2.5.7 Quantificação dos tempos de conexão

Para as análises acima foram considerados cenários imunes de interferência, porém em cenários práticos estes tempos podem ser bastante longos. A tabela 19 expõe estes valores [12].

Estado de Operação	Tempo Mínimo (s)	Tempo Médio(s)	Tempo Máximo
<i>Inquiry</i>	0,00125	3 – 5	10.24 – 30.72
<i>Paging</i>	0,0025	1.28	2.56
Total	0,00375	4.28 – 6.28	12.8 – 33.28

Tabela 19. Sumarização dos tempos para estabelecimento de conexão

Como visto acima, os tempos podem se tornar bastante elásticos. Porém, os mesmos podem ser absorvidos pelos sistemas de comunicação caso a conexão seja estabelecida antes de haver necessidade do uso do canal.

2.6 Gerente de enlace (*Link manager*)

2.6.1 Descrição geral

- É a primeira camada da pilha de protocolo que é implementada em software;

- As camadas HCI (*Host Controller Interface*), gerente de enlace (*link manager*), controlador de enlace (*link controller*), banda base e o rádio compõem um sistema Bluetooth;
- A comunicação com a entidade controladora (*host*), que contém as camadas de protocolo de nível mais alto, dá-se através da camada HCI;
- O LMP (*Link Manager Protocol*) é o protocolo que implementa as tarefas do gerente de enlace (*link manager*), por meio de interrogações e respectivas respostas;
- O LMP envia mensagem dentro de pacotes DM1 ou DV, a fim de realizar tarefas como configuração do canal, gerenciamento da *piconet* e segurança. Estas mensagens são chamadas de PDU (*Protocol Data Unit*);
- A especificação do Bluetooth 1.1 apenas diz o que o gerente de enlace (*link manager*) deve fazer, porém não amarra como as tarefas devem ser feitas;
- As mensagens do LMP têm prioridade sobre os dados do usuário;
- Todo o tratamento de correção e detecção de erro e retransmissão de pacotes é realizado pelo controlador de enlace (*link controller*). Sendo assim, a garantia de chegada das PDUs do LMP é feita pelo mesmo. O gerente de enlace (*link manager*) não tem procedimento para realizar estas tarefas. Há apenas uma temporização de 30 segundos para a chegada de uma resposta a uma determinada interrogação realizada pelo gerente de enlace remoto;
- O gerente de enlace realiza suas principais tarefas entre o momento que a conexão está estabelecida e o momento em que os usuários começam a trocar mensagens;
- Algumas PDUs do LMP podem requerer a ação da aplicação, outras do usuário e outras podem ser tratadas pelo próprio gerente de enlace.

2.6.2 Tarefas realizadas pelo gerente de enlace (*link manager*)

- Gerenciamento da *piconet*, a saber: conexão e desconexão de escravos, chaveamento entre mestre e escravo, estabelecimento de enlaces SCO, controle dos modos de baixa potência (*sniff*, *hold* e *park*), gerenciamento das diretivas de segurança, autenticação e criptografia do enlace.
- Descoberta de características suportadas pelos dispositivos Bluetooth e posterior configuração do enlace. A tabela 20 detalha as principais tarefas.

Descrição	Significado	Observação
Descoberta de características suportadas	Depois de conectados os dois dispositivos tem que descobrir quais os recursos que são suportados por ambos, através da PDU <i>LM_features_req</i>	A priori os dois dispositivos estão limitados a trocar somente pacotes <i>ID</i> , <i>FHS</i> , <i>Poll</i> , <i>DM1</i> e <i>DH1</i>
QoS	Especifica o <i>Tpoll</i> e o <i>Nbc</i> com o objetivo de garantir largura de banda e latência para as aplicações residentes nos escravos	<i>Tpoll</i> é o número máximo de <i>slots</i> em que o mestre pode passar sem transmitir para um determinado escravo. <i>Nbc</i> é o número de repetições do <i>broadcast</i>
Controle de Potência	Controle de potência de transmissão do mestre e escravo(s), de acordo com o <i>Golden Receive Power Range</i> dos receptores envolvidos.	Esta característica é obrigatória em dispositivos classe 1.
Tipos de pacotes	Escolha do tipo de pacote a ser usado para a transferência de dados <i>ACL - DH</i> ou <i>DM</i> com 1, 3 ou 5 <i>slots</i> .	
Controle de pacotes <i>Multislot</i>	É usado para enviar e receber a aprovação ou não para o uso de pacotes <i>multislot</i>	Através das seguintes mensagens: <i>LM_max_slot_req</i> <i>LM_accepted</i> <i>LM_not_accepted</i>
Informação do relógio e temporização	Usado para solicitar o <i>CLKN</i> do escravo, para em uma futura reconexão o mesmo obter o <i>CLKE</i> . É também usado para sincronizar o mestre quando da realização da inversão de papéis entre mestre e escravo. Desta forma, o antigo mestre obtém o <i>CLKN</i> para sincronizar-se com a fase do gerador de seqüência de saltos pseudo-aleatórias do novo mestre.	
Versão do LMP e Nome Bluetooth	A primeira PDU é usada para saber a versão de um LMP remoto e, assim verificar a compatibilidade com o LMP local. Já a segunda é usada para descobrir o nome amigável do dispositivo remoto.	

Tabela 20. Gerente de enlace – principais tarefas

2.7 802.11

O padrão original 802.11 permitia uma máxima taxa de transmissão de bit por canal de 2 Mbps, enquanto o atual padrão 802.11b, conhecido como Wi-Fi suporta uma máxima taxa de transmissão de bit de 11 Mbps, e os padrões 802.11a e 802.11g permitem uma taxa máxima de transmissão de bit de 54 Mbps. A tabela 21 compara as variações do padrão 802.11, destacando suas maiores vantagens e desvantagens.

Padrão	Espectro	Taxa de Trans. Máx	Transmissão	Compatível com	Maior desvantagem / Maior vantagem
802.11	2.4 GHz	1- 2 Mbps	FHSS/DSSS/DFIR	Nenhuma	Taxa de transmissão de bit limitada / Maior alcance.
802.11a	5.0 GHz	54 Mbps	OFDM	Nenhuma	Menor alcance dentre todos os padrões / Maior taxa de transmissão num espectro menos congestionado.
802.11b	2.4 GHz	11 Mbps	DSSS	802.11	Baixa taxa de transmissão de bit para determinadas aplicações / Largamente produzido e longo alcance
802.11g	2.4 GHz	54 Mbps	OFDM	802.11/802.11b	Número limitado de WLAN co-aloçadas / Maior taxa de transmissão de dados no espectro de 2.45 GHz

Tabela 21. Variações do padrão 802.11

Em relação à arquitetura, uma rede 802.11 tradicional consiste de um ponto de acesso (PA), que é freqüentemente parte de uma LAN (*Local Area Network*). Os usuários podem ter localização fixa, como um computador de mesa, ou semifixa, como um computador portátil. Para o ponto de acesso eles são considerados estações. Somente há comunicação entre as estações e o ponto de acesso, o que forma uma topologia em estrela. Esta configuração é formalmente denominada de *infrastructure service set*, ou simplesmente BSS, como mostrado na figura 40. Devido à sua estrutura de frequência canalizada, somente três BSS podem coexistir na mesma área sem que haja interferência mútua.

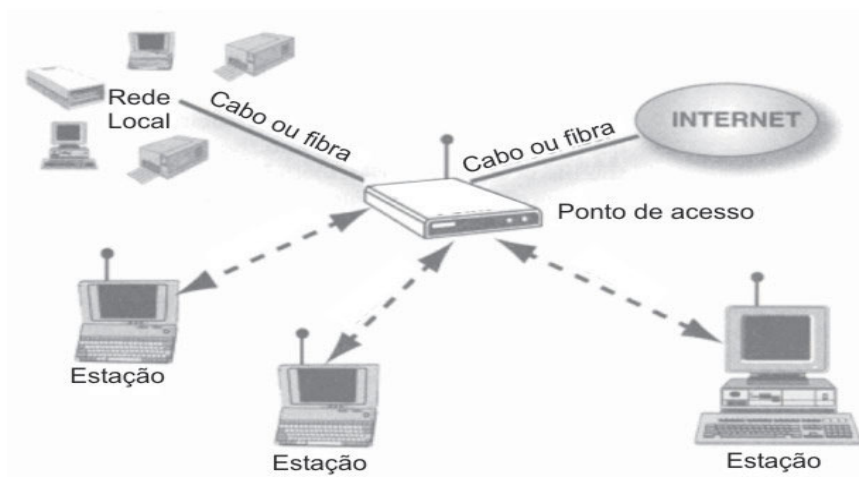


Figura 40. BSS

A arquitetura também suporta uma configuração *peer-to-peer*, onde uma estação se comunica diretamente com outra sem a necessidade de um ponto de acesso. Este tipo de configuração é chamada de *independent basic service set* (IBSS). Apesar de a primeira vista parecer uma configuração igual à provida em uma *piconet* Bluetooth, a IBSS somente suporta conexões ponto-a-ponto. Da mesma forma, somente três IBSS podem coexistir na mesma área.

Em geral pode-se dizer que o Wi-Fi (802.11b) é mais apropriado:

- Para aplicações onde a taxa de transmissão de bit é mais importante que a imunidade à interferência;
- Para aplicações onde a ocupação constante do espectro não constitui problema, pois as estações ficam sempre conectadas. Ao contrário do Bluetooth, onde os dispositivos se conectam somente até executarem uma determinada tarefa, deixando o espectro livre de interferência no restante do tempo;
- Para aplicações onde o consumo de potência não é fator determinante, o que exclui os dispositivos portáteis que utilizam baterias com autonomia reduzida.

Capítulo 3

O Sistema de teste

O sistema de teste foi desenvolvido baseado no seguinte cenário:

- Dispositivos de teste sem interface de comunicação a fim de realizar-se a coleta de dados;
- Dispositivos de teste específicos para cada produto;
- Pesada infra-estrutura para funcionamento, destacando-se a necessidade de uma rede de ar comprimido caracterizada por um alto custo de manutenção preventiva e freqüentes intervenções corretivas;
- Ambiente de programação pouco amigável.

Com base no cenário exposto acima, desenvolveu-se um sistema de teste com uma arquitetura de hardware capaz de torná-lo genérico para todos os modelos de módulos de *display* de cristal líquido (LCD) para telefones celulares fabricados atualmente.

Este sistema de teste é constituído por quatro módulos, a saber:

- Módulo Fonte. Responsável por gerar as tensões de alimentação necessárias;
- Módulo de Controle e Instrumentação. Responsável pelas funções de processamento, instrumentação, interface e acionamento;
- Módulo IHM. Responsável por estabelecer uma interface amigável com o operador;
- Módulo Fixture. Responsável por acondicionar mecanicamente o módulo LCD a ser testado.

No processo de validação dos módulos de LCD, o sistema de teste realiza:

- Medição do consumo de corrente das linhas do canal de comunicação serial. Normalmente SPI de 3 ou 4 fios;
- Medição de consumo das fontes de alimentação digital e analógica em diversas condições de operação;
- Ajuste automático de contraste e posterior gravação do mesmo em memória interna do *driver* do módulo de LCD, que pode ser do tipo EEPROM ou OTP;
- Escrita de telas padrões no LCD a fim de subsidiar a inspeção visual do módulo.

Abaixo são apresentadas as características técnicas da arquitetura de hardware e de software do sistema de teste.

3.1 Descrição de hardware

- Fonte alimentação linear com transformador toroidal, caracterizado por sua baixa interferência eletromagnética e alta eficiência, importante requisito para a alimentação de circuitos de instrumentação. Possui controle de habilitação e desabilitação, permitindo que o sistema de teste seja desligado através de comandos da IHM. Entrada 220 VAC e saídas: 5 VDC, 3.3 VDC, -5 VDC, 24 VDC e 15 VDC.
- Placa controladora RCM3200;
- Interface homem máquina (IHM) formada por um *display* de cristal líquido alfanumérico com 16 colunas e duas linhas; um teclado composto por 16 teclas padrão automação bancária;
- Microamperímetro com três canais cuja faixa de medição é de 0 a 1000 μ A com precisão de 1 μ A;
- Um voltímetro de oito bits, oito canais, precisão de 13mV, tensão de referência de 3.3 V e circuito de entrada configurado de acordo com a faixa de tensão a ser medida;
- Um voltímetro de cinco canais, resolução de 12 bits, tensão de referência de 2.5 V, ganho programável x1, x2, x4, x10, x16 e x20 e resolução de 1.2 mV para ganho unitário;
- Dezesseis saídas digitais. Nível TTL;
- Duas saídas digitais. Nível 24 VDC;
- Oito entradas digitais. Nível TTL;
- Duas saídas analógicas programáveis de 0 a 3.3 V, com passo de 13 mV;
- Uma saída analógica programável na faixa de -2.5 V a 2.5 V, com passo de 13 mV;
- Um interface serial síncrona a 3 ou a 4 fios. Nível CMOS;
- Duas interfaces seriais RS-232 para uso geral;
- Uma interface Ethernet 10/100 Base-T, embutida na placa controladora.

3.2 Descrição de software

3.2.1 Software básico

O software básico é formado por:

- Sistema operacional multitarefa μ COS/II[13];
- Ambiente de desenvolvimento Dynamic C 7.32;
- Sistema de arquivos FS2 do Dynamic C 7.32;
- Bibliotecas de *drivers* compostas por bibliotecas padrões do Dynamic C 7.32 e por bibliotecas desenvolvidas;
- A máquina de estado do sistema de teste é implementada por quatro tarefas, a saber:
 - Tarefa de inicialização, cuja função é executar a configuração do sistema de teste e indicar para as demais tarefas se este processo foi corretamente concluído. Feito isto a mesma se suspende;
 - Tarefa de teclado, cuja função é implementar uma varredura no teclado da IHM a cada 125 milisegundos, permitindo que o operador a qualquer momento faça o cancelamento do teste em curso, preencha os códigos de erro e acesse as funcionalidades do *menu* da IHM ;
 - Tarefa de desligamento, cuja função é realizar o correto desligamento do sistema de teste. Ela é suspensa no início de sua execução e somente retoma a executar quando um pedido de desligamento é realizado pelo operador por meio do teclado da IHM;
 - Tarefa de testes, cuja função é realizar os testes necessários para a validação funcional do DUT (*Device Under Test*);
 - Tarefa de emissão de testes, responsável por emitir os relatórios de teste realizados via canal de comunicação, que pode ser Ethernet 10/100 Base-T ou Bluetooth (alvo do estudo);

3.2.2 Capacidades das memórias *Flash* e RAM:

A memória *flash* padrão do RCM3200 é de 512 KB e memória RAM é de 256 KB. Entretanto, devido à funcionalidade de atualização do software de teste via canal de comunicação, metade da capacidade das mesmas é ocupada pelo software que gerencia a transferência dos dados. Sendo assim, neste sistema de teste, tem-se disponíveis

somente 256 KB de memória *flash* e 128 KB de memória RAM. O programa de teste, também chamado de programa de aplicação, ocupa aproximadamente 192 KB, sobrando somente 64 KB para armazenamento de dados. Este espaço então é usado para armazenar os relatórios de teste até que os mesmos sejam transferidos para o sistema de aquisição de dados. Porém, devido a estrutura de endereçamento do sistema de arquivos, dos 64 KB somente 32 KB estão disponíveis.

Um relatório de teste é armazenado como uma estrutura que ocupa aproximadamente 500 *bytes*, podendo-se armazenar então 65 relatórios.

Na memória RAM ficam alocados, entre outros, os relatórios de teste já devidamente formatados antes de serem enviados.

Capítulo 4

Proposta do sistema de comunicação

4.1 Requisitos do sistema de comunicação

- Os relatórios de teste devem ser formatados em XML de maneira a prover uma interface de dados bem definida, como também facilitar o processo de apresentação dos dados;
- O tamanho máximo dos relatórios de teste após a formatação em XML deve ser de aproximadamente 2500 *bytes*;
- Os diversos dispositivos de teste da linha de produção possuem tempos de execução diferentes, sendo o menor 20 segundos e o maior 35 segundos. Conseqüentemente, os tempos de ciclo, que compreendem dois testes consecutivos, são, respectivamente, 25 e 40 segundos. Considerando-se um tempo de preparação entre testes de cinco segundos;
- Os relatórios devem estar disponíveis no máximo 1 minuto após a realização do teste;
- O sistema de comunicação deve ser sem fio, de maneira a facilitar a instalação dentro do ambiente fabril, principalmente naqueles em que há controle de partículas em suspensão no ambiente - “Salas limpas”, onde qualquer obra civil é bastante indesejada;
- A integridade dos dados é de grande importância;
- Número de dispositivos de teste em produção é menor ou igual a 50;
- Distância máxima entre transmissor e receptor é de 30 metros;
- O ambiente é formado por um galpão de (50 x 30) metros, onde não há paredes, não há divisórias e as bancadas são de aço inoxidável;
- Não há memória de programa disponível para desenvolvimento de *drivers* específicos para a pilha Bluetooth;
- O custo da solução deve ser relativamente baixo, não podendo exceder em 10% o custo do sistema de teste sem a interface Bluetooth.

4.2 Escolha do padrão Bluetooth

A escolha do padrão Bluetooth foi devido às seguintes premissas iniciais de projeto:

- A imunidade à interferência provida pelo uso conjunto do FHSS, dos códigos corretores e detectores de erro e da retransmissão automática de pacotes. Característica muito importante para aplicações em ambiente fabril;
- A baixa taxa de transferência de bits dos dispositivos de teste;
- Sendo a tarefa de aquisição de dados de dispositivo de teste secundária em relação ao objetivo principal de validar os produtos, não se faz necessário que os dispositivos estejam sempre conectados à rede;
- O alcance médio da interface aérea. Os galpões de fábrica devido evolução tecnológica dos produtos não são mais demasiadamente extensões.
- Baixo custo em relação a outras tecnologias.

4.3 Análise crítica dos requisitos

- A taxa máxima de transferência de dados de 936 bps (considerando um tempo de ciclo de 20 segundos);
- Para esta taxa de transmissão, basta que o gerente de enlace (*link manager*) dos dispositivos negociem o uso de pacotes DM1, obtendo-se uma vazão de 108.8 Kbps, com CRC e com FEC de 2/3; ou pacotes DH1, obtendo-se uma vazão de 172.8 Kbps, porém sem FEC;
- O tempo para os seguintes processos: estabelecimento de conexão, transmissão dos dados, propagação do sinal, recepção, correção, detecção de erro, retransmissão, se necessária, e finalmente o repasse do relatório de teste integral para o nó destino, não pode exceder 1 minuto.
- O estudo teórico, com base na revisão bibliográfica apresentada no item Estudo da propagação em ambientes fechados e cálculo do alcance do rádio Bluetooth, do ambiente fabril cuja estrutura civil pode ser descrita da seguinte forma:
 - Galpão fechado;
 - Sem divisórias;
 - Sem paredes;
 - Bancadas da linha de produção formadas por chapas de aço inoxidável.

Mostra-nos que se pode ter entre os transmissores e receptores uma visada direta, caracterizando, assim, um caminho direto LOS. Isto nos leva a um expoente PL n ligeiramente menor que dois (espaço livre).

- Quanto ao multipercurso, este será gerado pelas bancadas de aço inoxidável e pela mobilidade das pessoas, já que os transmissores e receptores estão fixos. Como já mencionado na revisão bibliográfica, os efeitos do multipercurso, caracterizados pelo espalhamento Doppler e pelo espalhamento temporal, não causarão distorção significativa no canal Bluetooth, pois o mesmo é plano e possui desvanecimento lento. Assim sendo, a ISI causada pelo espalhamento temporal não aumentará significativamente a taxa de erro de bit do canal e a taxa de desvanecimento, causada pelo espalhamento Doppler, será desprezível. Porém, não se pode concluir que o multipercurso não afetará significativamente o canal, pois a intensidade do desvanecimento do sinal resultante depende exclusivamente da soma vetorial dos sinais do caminho direto e refletido. E isto pode chegar a 30 dB. Sendo assim, para manter o canal de comunicação operando e compensar os efeitos do multipercurso, o padrão Bluetooth combina as técnicas da diversidade na frequência, através do *FHSS*, e o controle de erro. Primeiro, o código corretor de erro é aplicado, para eliminar os possíveis erros de bit; depois, o pacote é checado por meio dos códigos detectores de erro, de maneira a verificar se ainda há erros no pacote recebido, pois há um limite para a correção de erros realizada na etapa anterior; e finalmente inicia-se o processo de ARQ. Vale salientar que, devido à técnica de diversidade na frequência, cada mensagem ACK ou NAK é enviada em uma frequência diferente.
- A técnica mais precisa para o cálculo do alcance de uma interface aérea é pelo método *Primary Ray Tracing*. Porém, não se dispõe do software para tal. Sendo assim, a classe de potência dos transmissores e sensibilidade dos receptores será calculada através da PL simplificada. Para uma distância de 30 (trinta) metros e considerando de forma pessimista um expoente PL n igual a 3 (ambiente moderado), obtém-se uma PL de 84 dB. Sendo assim, um transmissor padrão classe 1 (20 dBm) e um receptor padrão cuja sensibilidade é de -70 dBm para um BER de 10^{-3} , teoricamente atendem à necessidade. Desta forma, as perdas em larga escala somadas à do multipercurso podem chegar a

90 dB que o receptor será capaz de demodular o sinal desejado sem exceder uma taxa de erro de bit de 10^{-3} , considerando uma relação C/I dentro do especificado;

- Os rádios envolvidos devem ser capazes de implementar o controle de potência de modo a ajustar os seus transmissores dentro da *Golden Receive Power Range* dos receptores e, assim, minimizar a interferência causada em equipamentos eletrônicos vizinhos e compensar as perdas devido aos obstáculos no caminho entre o transmissor e receptor;
- A tarefa de integrar uma interface aérea Bluetooth a um produto demanda uma análise criteriosa com base nos seguintes parâmetros: tempo, tamanho, interoperabilidade, custo e flexibilidade. Para a aplicação alvo, tempo e custo são preponderantes. Sendo assim, a solução mais atrativa está no uso de sistemas embarcados, contendo as etapas de rádio, banda base, controlador de enlace (*link controller*), gerente de enlace (*link manager*), SDP, L2CAP e outras camadas específicas da aplicação (perfil de uso) em um único módulo. Vale salientar que nestes não há a interface HCI entre a camada L2CAP e o gerente de enlace (*link manager*). A interface do sistema embarcado é provida pela camada de aplicação.

4.4 Estudo da topologia do sistema de comunicação.

- Uma topologia como uma *piconet* contendo um mestre e até 7 escravos ativos não atende devido à quantidade de dispositivos de teste.
- Uma topologia com um mestre e até 255 escravos em modo *park* solicitando sua reativação de acordo com a sua necessidade de transmitir relatórios de teste é tecnicamente bastante viável. Neste os escravos funcionam como se fossem periféricos requisitando uma interrupção após a realização de uma tarefa. Porém, esta aplicação não está prevista em nenhum perfil de uso, o que onera tempo, devido à necessidade de desenvolvimento deste perfil e custo, principalmente o de desenvolvimento;
- Uma topologia com *piconets* ponto-a-ponto sendo formadas de acordo com a necessidade de transferência de relatórios de teste, e desfeitas após a transferência dos mesmos é também bastante atrativa, pois o tempo de transferência de um relatório de teste está em torno de poucas centenas de milissegundos. E o nó destino (ND), ver figura 41, ficaria ocupado por apenas

este intervalo de tempo, permitindo que outros dispositivos (NO) o compartilhem de forma seqüencial.

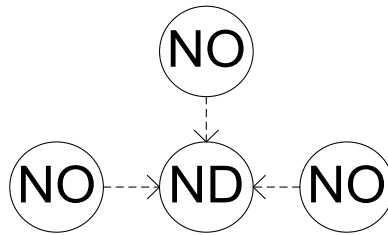


Figura 41. Formação de piconets sob demanda

A vantagem desta topologia é que há um perfil de uso de atende a esses requisitos, o SPP (*Serial Port Profile*), reduzindo-se assim o tempo e o custo de desenvolvimento. Outra vantagem é o custo deste sistema embarcado está em torno de US\$ 45.

4.5 Solução implementada

4.5.1 Descrição geral

Com base na análise crítica descrita acima, escolheu-se implementar a seguinte topologia: formação de *piconets* ponto-a-ponto de acordo com a demanda.

Como mencionado, as *piconets* serão formadas quando da necessidade de transmissão de dados (relatórios de teste). Seguindo o paradigma cliente-servidor, os futuros mestres (clientes), em estado *page*, deverão solicitar o estabelecimento de conexão com um único futuro escravo (servidor seqüencial), no estado *page scan* - modo SR contínuo (R0), caso haja relatórios de teste a serem transmitidos. As *piconets* formadas serão ponto-a-ponto, logo quando um cliente estiver conectado ao servidor, nenhum outro poderá transmitir. Este cenário persistirá por algumas poucas centenas de milissegundos até que o relatório seja transferido e a *piconet* seja desfeita, permitindo que o escravo volte automaticamente para o estado de *page scan*. Esta proposta apresenta vazão máxima. A figura 42 ilustra o diagrama de estados dos dispositivos clientes e do dispositivo servidor seqüencial para consumir-se o estabelecimento de conexão.

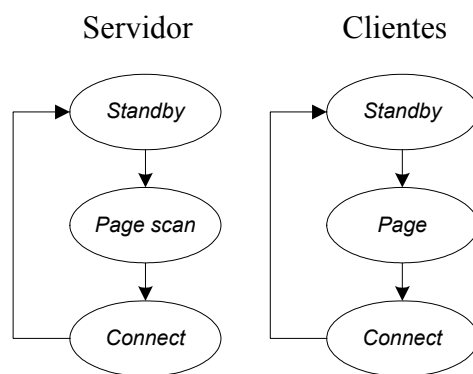


Figura 42. Diagrama de estados dos dispositivos servidor e cliente

4.5.2 Estabelecimento de conexão

Para a topologia proposta o endereço do futuro escravo (BD_ADDR) é previamente configurado, o que o habilita os futuros mestres a iniciarem diretamente um procedimento de *page*. Através do BD_ADDR os futuros mestres descobrem a seqüência de salto do futuro escravo. Inicialmente, os futuros mestres não conhecem a fase desta seqüência, nem o instante em que o futuro escravo estará no modo *page scan*, pois desconhecem o seu modo SR - se R0, R1 ou R2. Desta forma, o trem A, composto pelas 16 primeiras freqüências que serão inicialmente varridas pelos futuros mestres, é montado em função do seu próprio CKLN. O trem A deve ser varrido ciclicamente por 2.56 segundos, considerando o pior caso - futuro escravo habilitando sua janela de acesso a cada 2.56 segundos. Caso não obtenha sucesso, deve comutar para as 16 freqüências restantes que formam o trem B. Para evitar ciclos infinitos existe um controle de tempo neste processo.

Porém, depois que for estabelecida pela primeira vez uma conexão, os futuros mestres conhecerão uma estimativa do CLKN (CLKE) e o modo SR do futuro escravo. Desta forma, as primeiras 16 freqüências do trem A serão montadas com base no CLKE, alcançando o sincronismo de fase quanto maior for a precisão desta estimativa.

Como o futuro escravo opera no modo SR contínuo, estando então com sua janela de acesso sempre aberta varrendo a cada 11 ms uma freqüência diferente dentre as 32 previstas para este modo de operação, os futuros mestres devem varrer o trem A continuamente até obterem resposta do futuro escravo. Para evitar ciclos infinitos há um limite máximo de tempo em que os futuros mestres podem ficar no estado de *page*

O tempo de transferência do relatório de teste dependerá do escalonamento da tarefa de envio de relatório pela máquina de estado do sistema de teste, bem como do tipo de pacote escolhido de acordo com a qualidade do canal, através do CQDDR

4.5.3 Descrição dos relatórios de teste

Um relatório de teste é um arquivo em formato XML[14] contendo informações sobre o ambiente de teste. Dependendo do seu conteúdo o mesmo pode ser denominado: relatório de anúncio, relatório analítico, relatório sintético, relatório de parada automática, relatório de DUT (*Device Under Test*) inválido e relatório de desligamento.

O relatório de anúncio é emitido pela tarefa de inicialização assim que a máquina de estado de relatório de teste - responsável por armazenar, formatar e enviar os relatórios - estiver inicializada. É identificado conjuntamente pelo conteúdo da *tag* autoteste - valor 90.00, e por não haver dados válidos nas *tags* de resultado e valor.

O relatório analítico contém o resultado dos testes realizados. A tarefa de testes é responsável por informar à máquina de estado de relatório de teste o início e o fim de ciclo, como também, o início e o fim do teste. E, desta forma, indicar a necessidade de envio de um relatório de teste. É identificado conjuntamente pelo conteúdo da *tag* autoteste - valor 90.00, e por haver dados válidos nas *tags* de resultado e valor.

O relatório sintético contém somente a data, a hora de início e a hora final dos testes realizados dentro deste período, sendo que o campo resultado mostrará o número de defeitos quantificados. Esta é uma maneira de ainda obter dados estatísticos do processo produtivo mesmo que estoure a capacidade de armazenamento de relatórios de teste do sistema de teste. Este relatório também possui 99.00 como conteúdo de *tag* autoteste.

Há dois tempos sendo medidos no sistema de teste, a saber: o tempo de realização do teste e o tempo do ciclo, que é o tempo de teste acrescido do tempo de preparação para o próximo teste. Quando um tempo configurado estoura sem que nenhum teste tenha sido realizado, o sistema de teste envia um relatório de parada automática para fins de avaliação do balanceamento da linha de produção e quantificação das paradas de linha. A tarefa de teste é responsável por indicar para a máquina de estado de relatório de teste a necessidade do envio deste relatório. É identificado conjuntamente pelo conteúdo da *tag* autoteste - valor 99.95, e por não haver dados válidos nas *tags* de resultado.

O relatório de desligamento é emitido quando o sistema de teste é desligado através da IHM. É identificado pelo conteúdo da *tag* autoteste - valor 99.96.

O sistema de teste possui a funcionalidade de identificar o modelo do DUT (*Device Under Test*). Quando o mesmo não está configurado com nenhum modelo de DUT, por inexistência do módulo mecânico que condiciona o mesmo, ou por haver inconsistência entre o programa de teste e o DUT, é emitido um relatório de DUT

inválido. Este relatório é identificado conjuntamente pelo conteúdo da *tag* autoteste - valor 99.97 e por não haver dados válidos nas *tags* de resultado.

O relatório de teste utilizado nos relatórios de anúncio, analítico, de parada automática, de DUT inválido e de desligamento tem o formato mostrado abaixo.

```
<?xml version='1.0'?>
```

```
<RT>
```

```
  <cabecalho>
```

```
    <versao>%d.%d</versao>
```

```
    <jig>%s</jig>
```

```
    <dut>%s</dut>
```

```
    <numero>%ld</numero>
```

```
    <quantfila>%d</quantfila>
```

```
    <autoteste>%s</autoteste>
```

```
    <data>%ld</data>
```

```
    <hora>%ld</hora>
```

```
    <ticksteste>%ld</ticksteste>
```

```
    <ticksciclo>%ld</ticksciclo>
```

```
    <reteste>%s</reteste>
```

```
  </cabecalho>
```

```
  <teste>
```

```
    <T id='%3d'>
```

```
      <res>%d.%d</res>
```

```
      <val>%%.3f</val>
```

```
    </T>
```

```
  </teste>
```

```
</RT>
```


Já o relatório de teste sintético tem o seguinte formato:

```
<?xml version='1.0'?>
<RT>
  <cabsin>
    <jig>%s</jig>
    <dut>%s</dut>
    <niniteste>%ld</niniteste>
    <nfinalteste>%ld</nfinalteste>
    <data>%ld</data>
    <horainit>%ld</horainit>
    <horafim>%ld</horafim>
    <autoteste>%s</autoteste>
    <retestes>%d</retestes>
    <paradas>%d</paradas>
  </cabsin>
  <T id="%3d">
    <val>%8.3f</val>
  </T>
</RT>
```

4.5.4 Características do sistema embarcado SPP

4.5.4.1 Diagrama geral

A figura 43 ilustra a pilha de protocolos do sistema embarcado *SPP*, bem como a sua interface com o sistema de teste. O barramento de controle é constituído por um canal de comunicação RS-232, por 2 sinais de saída - um indicando que uma conexão foi estabelecida e outro indicando se o modo de controle está habilitado - e por um sinal de entrada para forçar uma desconexão e ao mesmo tempo habilitar o modo de controle. No modo de controle o dispositivo SPP aceita comandos de *page*, *page scan*, *inquiry*, *inquiry scan* e de segurança. Estes comandos são interpretados, na camada de aplicação, por uma máquina de estados que acessa devidamente os serviços da pilha de protocolos Bluetooth para executar os comandos solicitados, mantendo assim a integridade da referida pilha. Como mostrado na figura 43 não é necessário que o sistema de teste tenha nenhum *driver* Bluetooth.

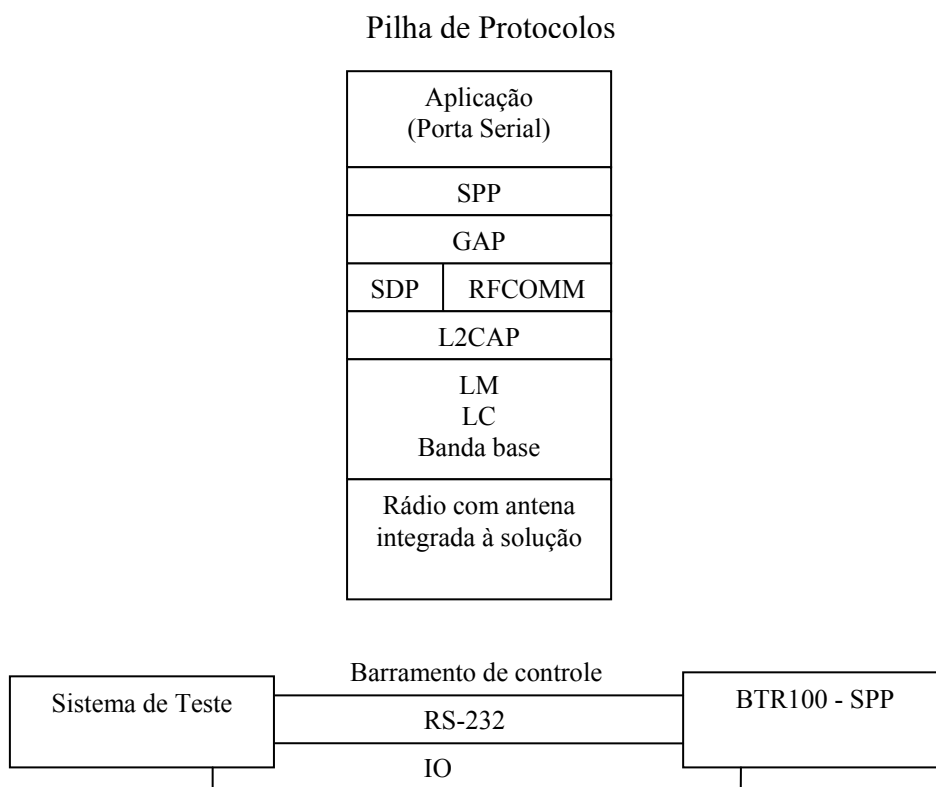


Figura 43. Pilha de protocolos e Interface entre o sistema de teste e o sistema embarcado Bluetooth

4.5.4.2 Características funcionais

- Perfil SPP. Permite a conexão de dois dispositivos em uma configuração ponto-a-ponto, emulando uma porta serial física sob um link Bluetooth. Protocolos utilizados: LMP, L2CAP, SDP e o RFCOMM.
- *Baud Rate* máximo da interface RS-232: 115200;
- Procedimentos suportados pelo controlador de enlace (*link controller*): *inquiry, inquiry scan, page, page scan*;
- Tipos de pacotes suportados: DM1, DH1, DM3, DH3, DM5 e DH5;
- L2CAP: suporta somente enlaces orientados a conexão;
- SDP: Pode ser cliente quando está em busca dos serviços suportados por um dispositivo remoto, e servidor quando está informando os serviços suportados;
- Utiliza CQDDR (*Channel Quality Driven Data Rate*) para escolher dinamicamente o tipo de pacote a ser utilizado de acordo com o BER do canal.
- Realiza Controle dinâmico da potência transmitida através do LMP (*Link Manager Protocol*). Através do controle de potência podem ser compensadas as perdas ocorridas na propagação do sinal.
- Suporta apenas a formação de *piconets* ponto-a-ponto;

4.5.4.3 Especificações técnicas do receptor

O receptor atende às exigências da especificação versão 1.1. Os itens abaixo relacionam os parâmetros otimizados.

- Sensibilidade: -79 dBm;
- Rejeição fora da banda a 1.9 GHz: 30 dB;
- Máximo nível de sinal na entrada: -20 dBm.

4.5.4.4 Especificações técnicas do transmissor

O transmissor atende às exigências da especificação versão 1.1. O itens abaixo relacionam os parâmetros otimizados.

- Máxima potência de saída: 17 dBm;
- Faixa do controle de potência de RF : 24 dB acima do nível mínimo;
- Passo mínimo do controle de potência :2 dB;
- Estabilidade de frequência para pacotes de 5 *slots*: $\pm 10\text{KHz}$

Capítulo 5

Resultados

5.1 Análise de performance do sistema de comunicação proposto

Suponhamos que uma *piconet* esteja estabelecida e que os demais dispositivos estejam tentando se conectar. Desta forma, os dispositivos conectados – mestre e escravo - estarão saltando os 79 canais de acordo com a seqüência de salto estabelecida pelo BD_ADDR e CLKN do dispositivo mestre; e os dispositivos futuros mestres, no estado *page*, estarão saltando os 32 canais de acordo com a seqüência de salto estabelecida pelo BD_ADDR do futuro escravo e pelo CLKE, que é a estimativa que cada um desses dispositivos possui do relógio do futuro escravo. Este cenário está ilustrado na figura 44.

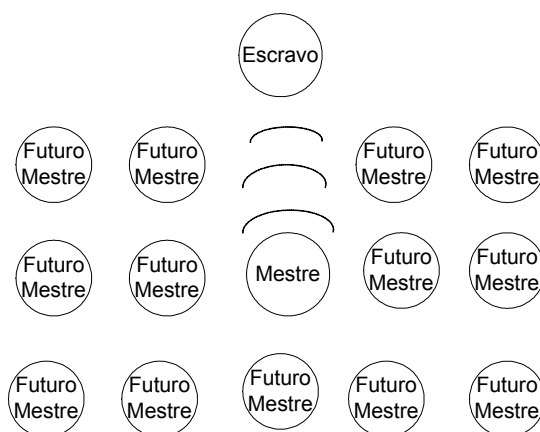


Figura 44. *Piconet coexistindo com futuros mestres em page scan*

Reportando à operação dos dispositivos na configuração proposta, o futuro escravo opera no modo SR - R0, significando que está sempre em *page scan* varrendo a cada 11.25 ms uma freqüência diferente das 32 que compõem a seqüência estabelecida para este estado de operação. Já os dispositivos futuros mestres, no estado *page*, estarão saltando os 32 canais (primeiro os 16 canais do trem A e depois os 16 canais do trem B) de acordo com a seqüência de salto estabelecida pelo BD_ADDR do futuro escravo e pelo CLKE. Na primeira tentativa de estabelecer uma conexão, os futuros mestres têm 50 % de possibilidade de errar o conjunto de canais em que o futuro escravo estará ouvindo, ou seja, as 16 freqüências estabelecidas pelo trem A podem não conter a seqüência de varredura do futuro escravo, que estará em uma das freqüências do trem B,

devido à imprecisão da estimativa do relógio do futuro escravo, pois inicialmente não possuem nenhum histórico de conexões. Como os futuros mestres inicialmente desconhecem o modo SR do futuro escravo, eles devem assumir o pior caso e varrer o trem A por 2.56 segundos antes de comutarem para as 16 frequências do trem B.

Neste cenário iniciemos então a análise de performance do sistema de comunicação, considerando de forma pessimista que em um dado *slot* um grupo de futuros mestres esteja varrendo de maneira síncrona o trem A e que o outro grupo esteja varrendo de maneira síncrona o trem B, ocupando os 32 canais possíveis com sinal não desejado (I) e que a colisão de pacotes aumente o BER do canal e cause perda total dos pacotes.

Ademais, para que o mestre da *piconet* consiga transmitir um pacote com sucesso, é preciso que o mesmo se comunique com o escravo em dois canais diferentes dos 32 canais supracitados, um para transmitir o pacote e outro para receber a confirmação do recebimento.

Considerando que a probabilidade de não ocorrer colisão (P_{NC}) é 1 menos a probabilidade de ocorrer a colisão (P_C) e que são necessárias dois canais livres, tem-se:

$$P_{NC} = (1 - P_C)^2 = \left(1 - \frac{32}{79}\right)^2 = 0.35 \quad (23)$$

Sob essas considerações, a vazão da *piconet* pode ser degradada em 65 % devido a colisão de pacotes promovida por futuros mestres adjacentes.

Entretanto, o cenário exposto acima é apenas transitório. Após a primeira conexão a entidade gerente de enlace (*link manager*) dos futuros mestre descobre as informações de operação e características suportadas pelo futuro escravo, tais como: o modo de SR e o CLKN. Através do modo SR montará o trem A e ajustará o tempo em que irá varrê-lo antes de chavear para o trem B; através do CLKN ajustará o CLKE aproximando-se do sincronismo de fase com o futuro escravo. Desta forma, à medida que as conexões forem sendo estabelecidas, a tendência é que todos convirjam para o trem A, que é o trem que possui maior probabilidade de encontrar o futuro escravo, pois a estimativa de fase ficará mais precisa. Sendo assim, a análise de pior caso se restringe a ter-se em um determinado instante as 16 frequências do trem A ocupadas com sinal de interferência I. Então, a probabilidade de não ocorrer colisão se torna:

$$P_{NC} = (1 - P_C)^2 = \left(1 - \frac{16}{79}\right)^2 = 0.63$$

Agora sob essas novas considerações, a vazão da *piconet* pode ser degradada em somente 37 % devido à colisão de pacotes promovida por futuros mestres adjacentes. A fim de concluir a análise, a figura 45 ilustra a probabilidade de não ocorrer perda de pacotes (P_{NC}) e a probabilidade de ocorrer perda de pacotes (P_C) em relação ao número de canais com sinal de interferência (I) capaz de subjugar a relação C/I do receptor, de maneira que toda colisão acarrete em perda de pacote.

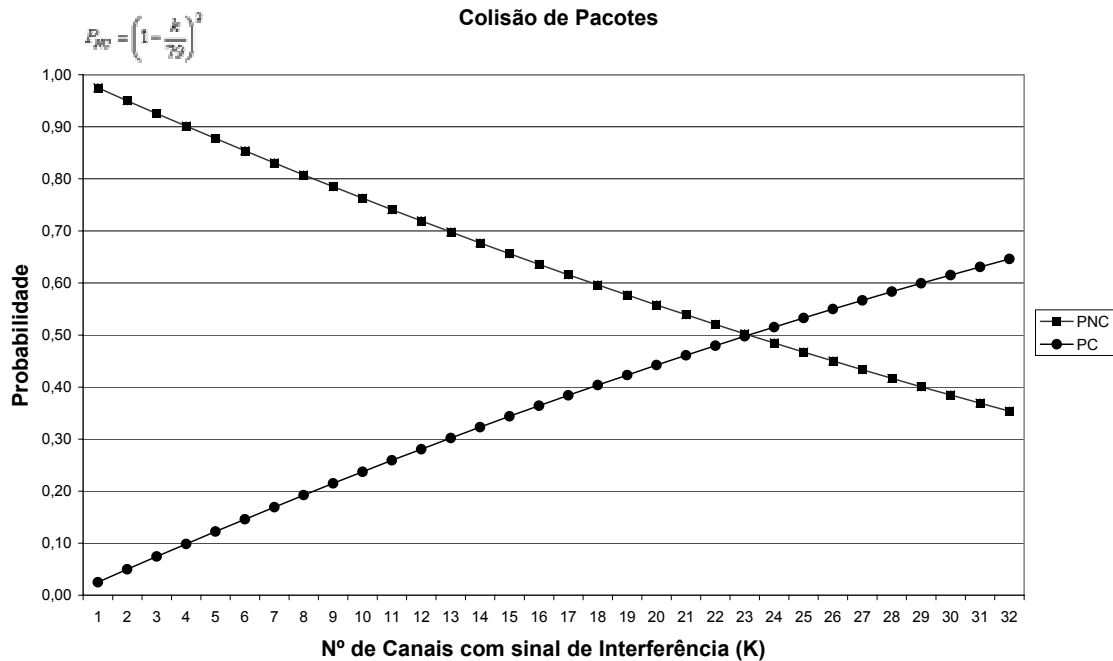


Figura 45. Probabilidade de perda de pacotes x Nº de canais com interferência

A fim de minimizar os cenários expostos acima, na aplicação desenvolvida, os dispositivos só podem ficar no estado *page* por um tempo mínimo necessário para estabelecer a conexão. Esta variável deve ser configurada, pois varia de acordo com o intervalo de tempo em que os vários dispositivos de teste levam para descarregar seus relatórios no servidor.

E, para adicionar confiabilidade ao sistema, os relatórios de teste contidos no sistema de arquivos dos dispositivos somente são excluídos após receber uma mensagem do servidor indicando que a validação do relatório de teste XML foi feita com sucesso. O contraponto é que isto aumenta a probabilidade de ocorrer colisão.

Entretanto, como a capacidade do enlace nesta topologia no pior caso, considerando o uso de pacotes DH1 cuja vazão é de 108.8 Kbps, é muito superior ao necessário por cada dispositivo (936 bps), a capacidade do enlace teria que ser degradada em mais de 90% para ficar abaixo do necessário.

Sendo assim, a variável que ditará a **quantidade máxima de clientes** que podem estar compartilhando o servidor seqüencial, é o intervalo de tempo necessário para que um dispositivo estabeleça conexão e envie todos os relatórios de teste armazenados. Este tempo tem que ser:

- Preferencialmente muito menor que o tempo de ciclo, para que a fila de arquivos a serem transmitidos não cresça excessivamente;
- E, caso seja maior, não pode exceder a capacidade de armazenamento de arquivos do sistema de teste.

5.2 Experimento – Probabilidade de colisão

A fim de comprovar as análises feitas acima, montou-se um ambiente em laboratório a fim de avaliar o quanto os dispositivos em modo *page* podem causar erro nos relatórios de teste enviados em uma *piconet* já previamente estabelecida. O erro será verificado após a validação do arquivo XML no servidor.

5.2.1 Descrição do ambiente experimental

Os clientes foram colocados a 45 cm um do outro para simular as condições reais de operação de dois dispositivos de teste colocados lado a lado. No caminho entre os clientes e o servidor há uma divisória de 2.5 m de altura e 3.6 m de comprimento, não chegando até o teto do ambiente. Este cenário configura um caminho direto obstruído. O expoente de perda do caminho n pode ser considerado 2.4 (sala com divisórias, porém com abertura para propagação do sinal perto do teto – ver no item Estudo da propagação em ambientes fechados e cálculo do alcance do rádio Bluetooth). E finalmente o servidor foi colocado à apenas 4 m dos clientes, aumentando a potência do sinal de interferência no receptor do mesmo. A figura 46 ilustra o exposto.

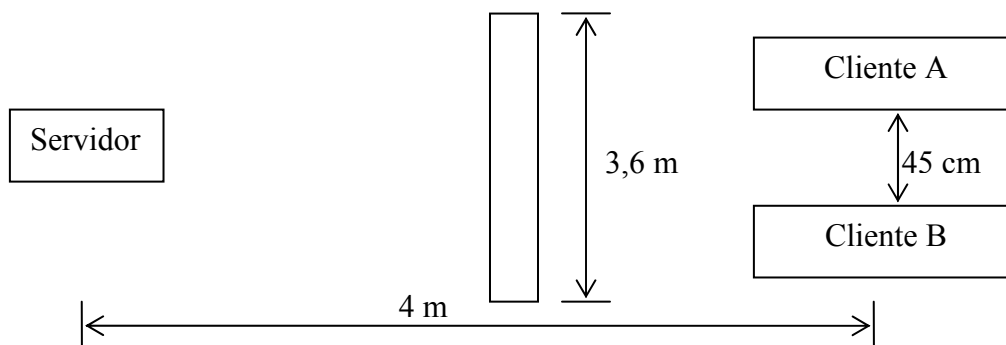


Figura 46. Descrição do ambiente experimental

A partir do ambiente descrito, é importante caracterizar o receptor do dispositivo servidor quanto ao nível de interferência sofrido.

A interferência é consumada quando, por exemplo, o cliente A estiver conectado ao servidor e o cliente B estiver tentando conectar-se. O receptor considerará o cliente B como uma interferência ao sinal desejado do cliente A. Neste contexto podemos calcular de forma genérica, a relação C/I em função da perda do caminho PL enfrentada por cada um dos transmissores envolvidos. Como estes transmissores possuem a mesma classe de potência (classe 1), a potência dos sinais desejado e indesejado na antena do receptor, será maior quanto menor for a perda enfrentada na propagação do sinal. De maneira simplificada, desconsiderando as perdas em pequena escala (multipercurso) e casamento de impedância nas antenas envolvidas, esta perda é devido a dois fatores: o expoente de perda do caminho n , ditado pelas características do ambiente, e a distância entre transmissor e receptor. Sendo assim, a relação C/I no receptor pode ser calculada por [3]:

$$C/I = PL_i - PL_d = 40 + 20n \log d_i - (40 + 20n \log d_d), \quad (24)$$

onde:

PL_i é a perda no caminho enfrentada pelo sinal indesejado;

PL_d é a perda no caminho enfrentada pelo sinal desejado;

d_d é a distância entre o transmissor desejado e o receptor;

d_i é a distância entre o transmissor indesejado e o receptor.

Desenvolvendo a fórmula (24) obtém-se:

$$d_i = d_d 10^{\left(\frac{C/I}{10n}\right)} \quad (25)$$

Agora, pode-se calcular, dentro de um determinado ambiente, a partir de qual distância o sinal indesejado poderá aumentar o BER do canal e assim causar perda dos pacotes.

Considerando-se um receptor padrão cuja relação C/I é dada pela tabela 5 e o ambiente descrito na figura 46, a tabela 22 sumariza as distâncias d_i em função das características do ambiente e da relação C/I em três situações: presença de sinal indesejado no mesmo canal, nos canais adjacentes a 1 MHz, nos canais adjacentes a 2 MHz e para canais adjacentes a mais de 2 MHz.

C/I (dB)	Situação	Nº de canais afetados	n	d_i (m)
11	Mesmo canal	1	2,4	11,49
0	Canais adjacentes a 1 MHz	3	2,4	4
-30	Canais adjacentes a 2 MHz	5	2,4	0,22
-40	Canais adj. a mais de 2 MHz	79	2,4	0,08

Tabela 22. Cálculo de d_i para o ambiente experimental

Interpretando os valores da tabela 22, o cliente B causará interferência significativa quando:

- Estiver a menos de 11.49 m do receptor e no mesmo canal do cliente A;
- Estiver a menos de 4 m do receptor e ocupando os canais adjacentes até a 1 MHz ;
- Estiver a menos de 0.22 m do receptor e ocupando os canais adjacentes a até 2 MHz ;
- Estiver a menos de 0.08 m do receptor e ocupando os canais adjacentes a mais de 2 MHz . Neste caso, o sinal indesejado está muito próximo do receptor de maneira que todos os 79 canais são afetados.

Pode-se então concluir que para o ambiente proposto, onde o cliente B está a quatro metros do receptor, somente haverá interferência significativa na recepção do cliente A, quando o cliente B estiver no mesmo canal.

5.2.1.1 Características de operação do servidor seqüencial

A título de análise das considerações no lado do servidor foi desenvolvido um aplicativo chamado STUBluetooth a fim de controlar as condições de operação do servidor, a saber:

- Manter o servidor no modo de operação *page scan* na sua inicialização e após desconectar;
- Depois de estabelecida a conexão, receber os relatórios de teste via porta serial;
- Realizar a validação (*parse*) do arquivo XML;
- Armazenar os relatórios em uma pasta de arquivos escolhida pelo usuário, observando duas regras para nomeação dos arquivos. Caso haja erro na validação, nomear o arquivo com o prefixo ERRO. Caso contrário, nomear de acordo com a escolha do usuário.

5.2.1.2 Características de operação do cliente A

Este cliente formará uma *piconet* com o servidor e enviará relatórios de teste continuamente, de maneira a subsidiar a análise proposta. Condições de operação:

- Número inicial de relatórios de teste no sistema de arquivos: 125;
- Taxa de geração de relatórios: 1 a cada 20 segundos;
- Tarefa de envio de relatórios escalonada a cada 5 segundos;
- Tempo aproximado de transferência do arquivo: 200 ms;

A *piconet* só será desfeita se não houver mais arquivos de teste, por isso, a necessidade de se ter um número inicial significativo de relatórios de teste no sistema de arquivos.

5.2.1.3 Características de operação do cliente B

Este cliente emula o comportamento de vários clientes no modo *page* solicitando conexão com o servidor a cada um segundo.

Configuração de operação:

- Tarefa de envio de relatórios escalonada a cada 1 segundo;
- Taxa de geração de relatórios: 1 a cada 5 segundos;
- Tempo máximo em modo *page*: 50 segundos;

Esta configuração mantém o cliente B em *page* por 50 segundos, antes que aconteça o estouro de tempo mandatário do módulo SPP que é de 60 segundos, e em seguida entra novamente em *page*, mantendo este ciclo indefinidamente.

5.2.2 Resultados - Estatística de perda dos relatórios de teste

Durante 21 minutos foram enviados 182 relatórios de teste para o servidor, todos foram recebidos com êxito. Isto é bem aceitável, haja vista que a probabilidade de não haver colisão era muito alta, pois mesmo o futuro mestre estando permanentemente no estado *page*, em um determinado instante, poderia causar interferência em apenas um canal, dentre os setenta e nove possíveis. Sendo assim, a probabilidade de não ocorrer colisão era de:

$$P_{NC} = (1 - P_C)^2 = \left(1 - \frac{1}{79}\right)^2 = 0.97$$

Portanto, havia apenas 3% de chance de ocorrer colisão de pacotes. E essa possibilidade foi absorvida pelo controle de erro, de maneira que nenhum relatório de teste foi perdido. Porém houve aumento, em determinados instantes, do tempo de transferência

do relatório de teste, o que denota diminuição da vazão da *piconet*, que teoricamente foi de 3%.

5.3 Experimento – Tempo de estabelecimento de conexão.

Neste experimento serão avaliados o tempo de estabelecimento de conexão e o tamanho da fila de relatórios de teste no sistema de arquivos de um determinado dispositivo. O tempo de ciclo do mesmo é inferior ao requerido pela aplicação e o canal de comunicação é compartilhado, havendo requisição de estabelecimento de conexão com o servidor a cada 1 segundo.

Utilizou-se o mesmo ambiente descrito no item anterior mudando-se somente a configuração dos clientes A e B.

5.3.1 Características de operação do cliente A

- Número inicial de relatórios de teste no sistema de arquivos: 0;
- Taxa de geração de relatórios: 1 a cada 20 segundos;
- Tarefa de envio de relatórios escalonada a cada 5 segundos;
- Tempo aproximado de transferência do arquivo: 200 ms;
- Número máximo de relatórios transferidos por conexão: até esvaziar a fila.

5.3.2 Características de operação do cliente B

Este cliente emula o comportamento de vários clientes no modo *page* solicitando conexão com o servidor a cada um segundo. Configuração de operação:

- Tarefa de envio de relatórios escalonada a cada 1 segundo;
- Taxa de geração de relatórios: 1 a cada 1.8 segundos;
- Número máximo de relatórios transmitidos por conexão: 1;
- Tempo aproximado de transferência do arquivo, depois de estabelecida a conexão: 200 ms.

5.3.3 Resultados – Tempo máximo de espera

Esta experiência teve a duração de 1 hora. Analisando o pior caso, obteve-se um tempo máximo de espera para conexão de 36.4 segundos, que, conseqüentemente, ocasionou o maior tamanho da fila, 04 relatórios. O tempo mínimo de conexão obtido foi de 0.8 segundos. Após o estabelecimento da conexão, o dispositivo descarregou todos os

quatro relatórios de teste antes que outro relatório de teste tenha sido colocado em sua fila.

Este resultado mostra que o tamanho do sistema de arquivos dos dispositivos de teste supre o atraso decorrente do alto tráfego no servidor, já que o tempo de transferência do relatório de teste é relativamente baixo. Na aplicação desenvolvida, o sistema de arquivos suporta 65 relatórios de teste.

Capítulo 6

Conclusões

6.1 Sobre o Trabalho Realizado

A solução proposta atendeu aos requisitos especificados dentro das condições experimentadas em laboratório. O padrão Bluetooth transcendeu minhas expectativas quanto à performance, mostrando-se perfeitamente ajustada às aplicações que não necessitem de altas taxas de transmissão de bits.

6.2 Sobre o Trabalho que pode ser Realizado

Uma melhoria imediata da solução proposta é substituir os módulos SPP que atendem à versão 1.1 da especificação Bluetooth por outros que atendam a especificação 1.2, pois a mesma já implementa um esquema de AFH (*Adaptative Frequency Holpping*), limitando-se a utilizar somente os canais que estão com taxa de erro de bit aceitável. Isto otimizará sobremaneira o cálculo da probabilidade de perda de pacotes devido à interferência de futuros mestres e principalmente devido a outras tecnologias colocados.

Outro trabalho muito interessante é abandonar o requisito baixo custo e implementar a topologia onde se monta uma *piconet* com um mestre e até 255 escravos em modo *park*, requisitando sua reativação também de acordo com a demanda, como se fossem co-processadores de uma unidade processadora central, solicitando o atendimento de uma interrupção.

Referências Bibliográficas

- [1] Bluetooth Special Interest Group, “*Specification of the Bluetooth System: Core,*” Volume 1, Ver 1.1, Feb. de 2001.
- [2] Shannon, Claude, “The Mathematical Theory of Communication”, *Bell System Technical Journal*, Jan. e Feb. 1948.
- [3]Morrow, Robert, *Bluetooth operation and Use*,1st ed. New York, McGraw-Hill, 2002.
- [4] Bansal, R. "The Far-Field: How Far is Far?", *Applied Microwave & Wireless*, November 1999.
- [5] Anderson, J., et al. "Propagation Measurements and Models for Wireless Communication Channels", *IEEE Communications Magazine*, November 1994.
- [6] Kim, S., et al. “Pulse Propagation Characteristics at 2.4 GHz Inside Buildings”, *IEEE Transactions on Vehicular Technology*, August 1996.
- [7] de Toledo, A. and Turkmani, M. "Estimating Coverage of Radio Transmission into and within Buildings at 900, 1800, and 2300 MHz," *IEEE Personal Communications Magazine*, April 1998.
- [8] Rappaport, T., *Wireless Communications: Principles & Practice*, 1 ed., Upper Saddle River, NJ: Prentice Hall, 1996.
- [9] Seidel,S. and Rappaport, T. "Site-Specific Propagation Prediction for Wireless In-Building Personal Communication System Design," *IEEE Transactions on Vehicular Technology*, November 1994.
- [10] Pahlavan, K. and Levesque, A. *Wireless Information Networks*, New York: John Wiley & Sons, 1995.
- [11] Bluetooth Special Interest Group, “Bluetooth Assigned Numbers”, Version 1.1, Feb. 2001.
- [12] Kamer, D. Senese, B. e Bray, J., *Bluetooth Application Developer’s Guide: The Short Range Interconnect Solution*, 1 ed., Rockland, MA, Syngress Publishing, Inc, 2002.
- [13] Labrosse, J., *MicroC/OS-II Tthe Real Time Kernel*,1 ed., Lawrence,KS, CMP Books, 1999.
- [14] Castro, E., *XML para a World Wide Web*, 1 ed., Rio de Janeiro, Campus, 2001.