

**Pontifícia Universidade Católica do Rio Grande do Sul**  
**Faculdade de Informática**  
**Programa de Pós-Graduação em Ciência da Computação**

**R-MAC: Protocolo de Acesso  
ao Meio para Consumo  
Eficiente de Energia em  
Redes de Sensores Sem Fio.**

Guilherme Frederico Rohde

**Dissertação apresentada como  
requisito parcial à obtenção do  
grau de mestre em Ciência da  
Computação**

Orientador: Prof. Dr. Eduardo Augusto Bezerra

Porto Alegre  
2009



### **Dados Internacionais de Catalogação na Publicação (CIP)**

R737r Rohde, Guilherme Frederico  
R-MAC : protocolo de acesso ao meio para consumo  
eficiente de energia em redes de sensores sem fio /  
Guilherme Frederico Rohde. Porto Alegre, 2009.  
95 f.  
  
Diss. (Mestrado) – Fac. de Informática, PUCRS  
Orientador: Prof. Dr. Eduardo Augusto Bezerra  
  
1. Redes de Sensores Sem Fio. 2. Protocolos  
(Informática). 3. Energia – Consumo Eficiente.  
4. Informática. I. Bezerra, Eduardo Augusto. II. Título.  
  
CDD 004.68

**Ficha Catalográfica elaborada pelo  
Setor de Tratamento da Informação da BC-PUCRS**



Pontifícia Universidade Católica do Rio Grande do Sul  
FACULDADE DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

## TERMO DE APRESENTAÇÃO DE DISSERTAÇÃO DE MESTRADO

Dissertação intitulada "**R-MAC: Protocolo de Acesso ao Meio para Consumo Eficiente de Energia em Redes de Sensores Sem Fio**", apresentada por Guilherme Frederico Rohde, como parte dos requisitos para obtenção do grau de Mestre em Ciência da Computação, Sistemas Digitais e Arquitetura de Computadores, aprovada em 31/08/2007 pela Comissão Examinadora:

Prof. Dr. Eduardo Augusto Bezerra –  
Orientador

PPGCC/PUCRS

Prof. Dr. Fabiano Passuelo Hessel –

PPGCC/PUCRS

Profa. Dra. Ana Cristina Benso da Silva –

FACIN/PUCRS

Prof. Dr. Fabian Luís Vargas –

FENG/PUCRS

Homologada em 03/03/09, conforme Ata No. 003/09 pela Comissão Coordenadora.

Prof. Dr. Fernando Luís Dotti  
Coordenador.

PUCRS

Campus Central

Av. Ipiranga, 6681 – P32 – sala 507 – CEP: 90619-900

Fone: (51) 3320-3611 – Fax (51) 3320-3621

E-mail: [ppgcc@inf.pucrs.br](mailto:ppgcc@inf.pucrs.br)

[www.pucrs.br/facin/pos](http://www.pucrs.br/facin/pos)

*Dedico ao meu filho Lucas.*



## **Agradecimentos**

Agradeço primeiramente a minha família, que me ensinou as lições que não aprendi na escola.

Ao Bezerra, por ter confiado em meu potencial e por ter me apoiado durante todo este longo caminho.

Aos participates do GSE pela parceria, pelos momentos de descontração e de trabalho. Em especial a equipe do RFID, com quem aprendi muito neste último ano.

Aos meus irmãos Nuno, André, Fábio, Lourenço, Leonel e Guz, amigos que sempre estarão por perto.

Obrigado!





## **Resumo**

Este trabalho descreve o R-MAC, um protocolo de acesso ao meio projetado para atender as requisições de energia das Redes de Sensores Sem Fio (RSSFs). A fim de melhor explicar o protocolo proposto, em um primeiro momento foi realizado um estudo em redes wireless em geral. Todas as camadas de uma RSSF são introduzidas, com ênfase na camada de controle de acesso ao meio (MAC), que é a motivação principal do R-MAC. O problema do acesso ao meio é discutido, e alguns tópicos de interesse são investigados. Uma discussão de protocolos relacionados também é conduzida, e os aspectos específicos dos protocolos estudados considerados no projeto do R-MAC. O R-MAC foi desenvolvido para RSSF dirigidas a eventos, com coleta periódica, insensível a latência da rede e com baixa taxa de envio. A eficiência no consumo de cada nodo, aumenta o tempo de vida da RSSF como um todo. Além disso, propõe-se uma técnica para buscar novas conexões toda vez que um nodo sair da rede. Esta técnica resulta no aumento da interação de cada nodo com a rede. O protocolo proposto é de fácil implementação em nodos de RSSFs baseados em microprocessador, e suas eficiência e aplicabilidade foram mostradas com a avaliação de equações para estimar o consumo de potência.

Palavras-chave: Redes de Sensores Sem Fio, Protocolo MAC, Consumo Eficiente de Energia.



## **Abstract**

This work describes R-MAC, a medium access protocol conceived aiming at power consumption requirements of a wireless sensor network (WSN). In order to better explain the proposed protocol, first there is a study on wireless networks in general. All layers of a WSN are introduced, with emphasis on the medium access control (MAC) layer, which is R-MAC's main target. The medium access problem is discussed, and some major concerns are investigated. A discussion on related protocols is also conducted, and specific aspects of the studied protocols are considered in R-MAC's design. R-MAC targets event driven WSNs, having periodic collection, unaware of network latency, and presenting a low transmission rate. In addition, a technique is proposed for searching new connections every time a node leaves the network. This technique results in improvements in the interaction time of the node with the network. The proposed protocol is of easy implementation in microprocessor based WSN nodes, and its efficiency and applicability have been shown through the evaluation of power estimation equations.

**Keywords:** Wireless Sensor Network, MAC protocol, Energy Efficient.



## Lista de Figuras

Figura 1	Divisão de um nodo quanto ao consumo. . . . .	22
Figura 2	Access Point da RSSF. . . . .	26
Figura 3	Arquitetura de um nodo sensor. . . . .	26
Figura 4	Modelo OSI/ISO. . . . .	29
Figura 5	Multiplexação por divisão de frequência. . . . .	33
Figura 6	Multiplexação por divisão de Código . . . . .	34
Figura 7	Multiplexação por divisão de tempo . . . . .	34
Figura 8	Estrutura do <i>frame</i> . . . . .	35
Figura 9	TDMA pacotes recebidos. . . . .	35
Figura 10	RTS/CTS/DATA. . . . .	36
Figura 11	Problema do terminal escondido. . . . .	37
Figura 12	Problema da estação exposta. . . . .	38
Figura 13	Período <i>listen/sleep</i> para um nodo sensor. . . . .	39
Figura 14	Nodos com período <i>listen/sleep</i> diferentes. . . . .	39
Figura 15	Mediador entre nodos. . . . .	40
Figura 16	Protocolo S-MAC. . . . .	44
Figura 17	Protocolo T-MAC. . . . .	45
Figura 18	Problema de dormir cedo. . . . .	46
Figura 19	Máquina de estados do módulo de RF CC1100. . . . .	54
Figura 20	Atuação do protocolo MAC na máquina de estados do Módulo RF de um nodo sensor . . . . .	56
Figura 21	Ruído acumulado SNR . . . . .	58
Figura 22	Máquina de estados finita para o recebimento de mensagem . . . . .	59
Figura 23	Máquina de estados da configuração inicial da RSSF. . . . .	61
Figura 24	Auto-configuração do ID dos nodos da RSSF. . . . .	62
Figura 25	Fluxograma da função RET. . . . .	64
Figura 26	Funcionamento da RET. . . . .	65
Figura 27	Situação em que o nodo retorna a configuração inicial . . . . .	65
Figura 28	Nova conexão para nodos isolados na RSSF. . . . .	66
Figura 29	<i>InterFrame Spacing</i> . . . . .	67
Figura 30	Função Contador de Nodos Ouvintes. . . . .	69
Figura 31	Usando o CNO. . . . .	69
Figura 32	Os quatro campos básicos do pacote. . . . .	70
Figura 33	Campo preâmbulo. . . . .	71
Figura 34	Campo <i>delimiter</i> . . . . .	72
Figura 35	Campo R-MAC header. . . . .	73
Figura 36	Maquina de estados do Módulo de RF de um nodo sensor . . . . .	74
Figura 37	Tempo de transmissão de uma Comunicação direta . . . . .	79
Figura 38	Nodos ouvintes. . . . .	83



## Lista de Tabelas

Tabela 1	Consumo de um nodo por domínio. . . . .	22
Tabela 2	Protocolos MAC para RSSF. . . . .	49
Tabela 3	Energia de Transmissão / Frequência / Potência típica de Consumo (*tensão base 3V) . . . . .	63
Tabela 4	PSF-Taxa de recebimento do pacote. . . . .	72
Tabela 5	Tipo de pacotes do R-MAC. . . . .	74
Tabela 6	Funções do subcampo To e From. . . . .	75
Tabela 7	Contador de Nodos Ouvintes. . . . .	75
Tabela 8	Consumo em cada estado do nodo. . . . .	78
Tabela 9	Valores de $t_{Dados}$ para diferentes tamanhos de pacotes. . . . .	80
Tabela 10	Fração de tempo dos nodos pelo tamanho dos pacotes. . . . .	80
Tabela 11	Fração de tempo dos nodos ouvinte para cada estado sem CNO. . . . .	83
Tabela 12	Fração de tempo dos nodos ouvinte para cada estado com CNO. . . . .	84
Tabela 13	Comparação do R-MAC com outros protocolos MAC para RSSF. . . . .	87





## Lista de Símbolos e Abreviaturas

AP	<i>Access Point</i>
CDMA	<i>Code Division Multiple Access</i>
CSMA/CA	<i>Carrier Sense Multiple Access with Collision Avoidance</i>
CSMA/CD	<i>Carrier Sense Multiple Access with Collision Detect</i>
DCF	<i>Distributed Coordination Function</i>
DIFS	<i>DCF Interframe Space</i>
FIFO	<i>First In First Out</i>
ISM	<i>Industrial, Scientific and Medical</i>
IR	<i>Infra-Red</i>
RSSF	<i>Redes de Senores Sem Fio</i>
MANET	<i>Mobile Ad-Hoc Network</i>
DSP	<i>Digital signal processing</i>
MAC	<i>Medium Access Control</i>
MANETs	<i>Mobile Ad-Hoc Network</i>
PCLP	<i>Physical Layer Convergence Procedure</i>
PLW	<i>PCLP length word</i>
PSF	<i>PCLP Signaling Field</i>
RF	<i>Rádio Frequência</i>
RC	<i>Recurso de Comunicação</i>
SAN	<i>Stochastic Automata Networks</i>
SPI	<i>Serial Peripheral Interface</i>
SNR	<i>Signal to Noise Ratio</i>
SIFS	<i>Short Interframe Space</i>
SFD	<i>Start Frame Delimiter</i>
UARTs	<i>Universal Asynchronous Receiver/Transmitter</i>
WSN	<i>Wireless Sensor Network</i>



# Sumário

<b>1</b>	<b>Introdução</b>	<b>21</b>
1.1	Motivação	23
1.2	Objetivos Geral e Específicos	23
1.3	Organização do Documento	23
<b>2</b>	<b>Conceitos Básicos de RSSF e Acesso ao Meio</b>	<b>25</b>
2.1	Características de uma RSSF	25
2.2	Camadas de Protocolo em RSSF	28
2.2.1	Camada Física	29
2.2.2	Camada de Enlace	30
2.2.3	Camada de Rede	31
2.2.4	Camada de Transporte	31
2.3	Técnicas de Acesso ao Meio Em RSSFs	32
2.3.1	FDMA - <i>Frequency-Division Multiple Access</i>	32
2.3.2	CDMA - <i>Code-Division Multiple Access</i>	33
2.3.3	TDMA - <i>Time-Division Multiple Access</i>	34
2.3.4	CSMA/CA - <i>Carrier Sense Multiple Access with Collision Avoidance</i>	36
2.4	Acesso ao Meio em RSSFs	36
2.4.1	Problema do terminal escondido	37
2.4.2	Problema da estação exposta	37
2.4.3	<i>Fairness</i>	38
2.4.4	Latência	38
2.4.5	Distância entre os nodos	38
2.4.6	Período <i>listen/sleep</i>	39
2.4.7	Sincronização entre nodos	39
<b>3</b>	<b>Trabalhos Relacionados</b>	<b>43</b>
3.1	Protocolo S-MAC	43
3.2	Protocolo T-MAC	45
3.3	Protocolo ARC	47
3.4	Protocolo DE-MAC	47
3.5	Protocolo TRAMA	48
<b>4</b>	<b>Protocolo Proposto R-MAC</b>	<b>51</b>
4.1	Visão Geral do R-MAC	51
4.2	Interação do R-MAC com o módulo RF	52
4.2.1	Máquina de estados da camada Física.	53
4.2.2	Modelagem da máquina de estados da camada de Enlace	55
	4.2.2.1 Envio de mensagem	57

4.2.2.2	Recebimento de mensagem . . . . .	57
4.3	Funcionalidades do R-MAC . . . . .	59
4.3.1	Funcionalidades de Auto-Gerência . . . . .	59
4.3.1.1	Função SN ( <i>Search Neighbor</i> ) . . . . .	60
4.3.1.2	Energia de Transmissão - RET . . . . .	63
4.3.2	Funcionalidades de comunicação . . . . .	66
4.3.2.1	<i>Interframes Space</i> . . . . .	66
4.3.2.2	Janela de contenção . . . . .	67
4.3.2.3	Detecção de Erros . . . . .	68
4.4	Funcionalidades de economia de energia . . . . .	68
4.4.1	Função <i>Listen/Sleep</i> . . . . .	68
4.4.2	Função Nodos ouvintes - CNO . . . . .	69
4.5	Formato dos Pacotes . . . . .	70
4.5.1	Preâmbulo . . . . .	70
4.5.2	<i>Delimiter</i> . . . . .	71
4.5.3	<i>Cyclic Redundancy Check(CRC)</i> . . . . .	72
4.5.4	MAC header . . . . .	73
4.5.4.1	Campo Address . . . . .	73
4.5.4.2	Campo <i>Frame Body</i> . . . . .	73
4.5.4.3	Campo <i>Frame Control</i> . . . . .	73
<b>5</b>	<b>Validação e Análise dos Resultados . . . . .</b>	<b>77</b>
5.1	Modelo de Energia . . . . .	77
5.2	Comunicação Direta . . . . .	78
5.3	Energia nos Nodos Ouvintes . . . . .	82
<b>6</b>	<b>Conclusão e Trabalhos Futuros . . . . .</b>	<b>85</b>
	<b>Referências Bibliográficas . . . . .</b>	<b>89</b>

# 1 Introdução

Avanços nas áreas de comunicação *wireless*, eletrônica digital e sensores estão viabilizando a fabricação de dispositivos de baixo custo e consumo reduzido [17]. Dentre as tecnologias que se destacam nesse cenário estão os dispositivos chamados de nodos sensores.

Nodos sensores são sistemas embarcados, que apesar de possuírem uma baixa capacidade de processamento e de memória, são capazes de realizar várias tarefas. A interação de vários nodos por enlace sem fio forma um sistema de rede. Este sistema apresenta um grande potencial para aplicação em diversas áreas. Atividades como monitoramento ambiental, controle industrial, sistemas médicos, segurança pública e de ambientes em geral, áreas de desastres e de risco para vidas humanas, transporte e sistemas móveis são algumas das aplicações possíveis para uma Redes de Senores Sem Fio (RSSF) [17, 36, 48].

Se comparadas com outras redes multimídias sem fio, como, MANET [43] e *Bluetooth* [20], as RSSF são similares na infra-estrutura de comunicação. Porém, diferem em alguns aspectos como, por exemplo, a grande quantidade de elementos (nodos) e a mobilidade [24]. Além disso, os nodos de uma rede *Bluetooth* ou MANET, geralmente, são dispositivos de uso pessoal e possuem baterias portáteis que podem ser recarregadas ou trocadas por seus usuários [36].

Em muitas aplicações de RSSF, centenas ou milhares de elementos distribuídos são depositados em áreas remotas, funcionando sem a intervenção humana direta. Logo, por questão de conveniência, os nodos sensores devem operar com baterias

O consumo de energia em RSSF é um importante ponto para o projeto destas redes. A maior parte das aplicações em RSSF exigem que a rede funcione por longos períodos de tempo sem intervenção humana [47]. Este tempo pode variar de alguns meses até alguns anos, sem que a bateria seja substituída. Logo, a característica que mais diferencia uma RSSF das outras rede sem fio é a capacidade restrita da fonte de energia dos dispositivos [17, 48].

Para aumentar o tempo de vida de uma RSSF, restrições do consumo de energia são pesquisadas em vários níveis, desde a arquitetura do *hardware* [23, 39, 42, 45], sistema operacional [4, 16, 25] protocolos de acesso ao meio [44, 46, 47], protocolos de roteamento [6, 33] e algoritmos utilizados pelas aplicações [2, 5, 26].

O consumo de energia em um nodo de uma RSSF pode ser dividido em três domínios: sensoriamento, comunicação e processamento dos dados, conforme mostrado na Figura 1.

A unidade de sensoriamento e seus componentes dependem da complexidade da detecção do

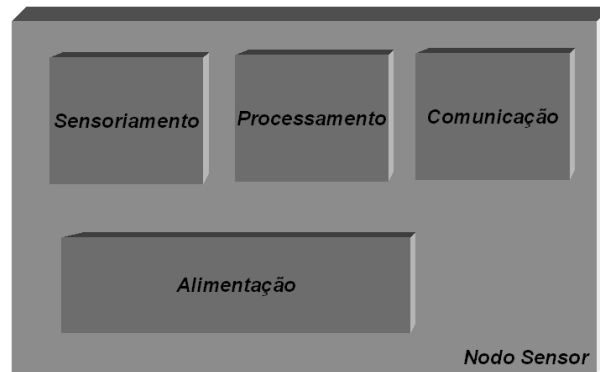


Figura 1: Divisão de um nodo quanto ao consumo.

evento. O aumento nos níveis de ruído do ambiente podem elevar a complexidade de detecção [48]. Neste domínio os eventos do mundo real são transformados em dados digitais, sendo que a quantidade de dados coletados e a fidelidade dos mesmos deve obedecer às necessidades da aplicação. Por este motivo o sensoriamento tornar-se pouco flexível com relação a estratégias de consumo de energia.

Ao longo do desenvolvimento do presente trabalho, foi realizada uma pesquisa de forma a indicar os níveis de consumo de cada domínio em um nodo sensor, os resultados desta pesquisa estão resumidas na Tabela 1. No domínio do processamento a energia consumida é muito menor que a energia necessária para a comunicação.

Tabela 1: Consumo de um nodo por domínio.

Domínio				Potência <i>mW</i> Tensão base 3V	
Processamento	Microcontrolador	Corrente ( <i>mA</i> )	Frequência de operação		
	<i>MSP430</i> [10]	0,25	1 <i>MHz</i>	0,75	
	<i>TSM470</i> [11]	10	40 <i>MHz</i>	30	
Sensoriamento	Sensor	Corrente ( $\mu A$ )	Precisão		
	Magnetômetro [14]	650	2 <i>mGauss</i>	1,95	
	Luz [?]	200	6 <i>mW/m<sup>2</sup></i>	0,6	
	Temperatura [13]	600	0.25°C	1,8	
	Pressão [15]	650	2,4 <i>mPSI</i>	1,95	
Comunicação	RF		Corrente ( <i>mA</i> )	Frequência de Transmissão/DataRate	
	<i>CC1100</i> [7]	TX	30,7	915 <i>MHz</i> /+10 <i>dBm</i> /250 <i>kbps</i>	92,1
		RX	16,4		49,2
	<i>TRF6903</i> [8]	TX	35	915 <i>MHz</i> /0 <i>dBm</i> /64 <i>kbps</i>	105
		RX	18		54
	IR		Corrente ( <i>mA</i> )	Frequência de Transmissão/DataRate	
	<i>TIR1000</i> [9]	TX	1	1.843 <i>MHz</i> /1200 <i>bps</i> – 115 <i>kbps</i>	3
		RX	1		3

O domínio de comunicação é o maior responsável pelo consumo de energia em um nodo sensor. Na transmissão de dados (TX) o consumo é bastante elevado. Ainda, na recepção ou na procura por comunicação (RX), o consumo de energia também é significativo. Conseqüente-

mente, gerência a comunicação entre os nodos é uma boa estratégia para evitar perdas e garantir um consumo eficiente de energia em uma RSSF [23, 29].

## 1.1 Motivação

Considerando o problema do consumo de energia relacionado ao método de comunicação de um nodo de uma RSSF, isso vem a ser a principal motivação desta pesquisa. Também é importante destacar o estudo de protocolos de comunicação para redes *wireless* como uma importante motivação, assim como a aplicação do protocolo proposto em uma RSSF.

## 1.2 Objetivos Geral e Específicos

Nodos sensores são dispositivos projetados para terem um baixo custo. Logo, seu *hardware* é limitado tendo pouca capacidade de memória, baixa capacidade de processamento e dispositivos de comunicação com poucos recursos. Portanto, o uso de protocolos tradicionais *ad hoc* não é viável em RSSF [17].

A comunicação em RSSF pode ser dividida em quatro camadas do modelo OSI da ISO [43], Física, Enlace, Rede e Transporte. A camada de Enlace é responsável pelo controle de acesso ao meio (MAC - *Medium Access Control*), sendo descrita por um protocolo. Este protocolo controla a transmissão de pacotes na rede, tentando assegurar que não haverá colisões de pacotes por interferência de outros nodos da rede, gerenciando a negociação entre os mesmos.

O objetivo geral deste trabalho é investigar e analisar protocolos de acesso ao meio para redes *ad hoc* e de sensores, propondo as técnicas que possibilitem um consumo eficiente de energia em uma RSSF. Estas técnicas servirão como base para a especificação de um protocolo MAC para RSSFs dirigida a eventos, com coleta periódica, insensível a latência da rede e com baixa taxa de envio. Sendo o seu principal objetivo eliminar as trocas freqüentes de bateria nestas redes, aumentando o tempo de vida das mesmas.

## 1.3 Organização do Documento

Esta dissertação está estruturada em cinco capítulos. No capítulo 1 foi apresentada uma introdução com uma visão geral do trabalho, bem como a motivação e os objetivos. O capítulo 2 descreve os conceitos básicos sobre redes de sensores sem fio, os projetos de pesquisas atuais, as características dos nós sensores e a sua pilha de protocolos, focando principalmente na camada de enlace. No capítulo 3 é apresentado os trabalhos relacionados e o estado da arte em protocolos de acesso ao meio para RSSF. O capítulo 4 apresenta a proposta do protocolo R-MAC. O capítulo 5 apresenta as simulações. Finalmente, o capítulo 6 apresenta as conclusões e sugestões para trabalhos futuros.





## 2 Conceitos Básicos de RSSF e Acesso ao Meio

Neste capítulo serão apresentados conceitos importantes para a compreensão da proposta descrita nesta dissertação. A primeira seção apresenta uma breve introdução sobre RSSFs apresentando as principais características da rede, uma idéia básica da arquitetura utilizada nos nodos sensores e alguns pontos importantes para projetos de RSSF. Em seguida, é apresentada uma visão da comunicação em RSSFs dividida em camada Física, de Enlace, de Rede e camada de Transporte. A seção 2.1 apresenta uma breve explicação sobre as técnicas de acesso e sua relação com RSSF. Por último, são apresentados conceitos importante para a implementação de um protocolo de acesso ao meio em RSSF, mostrando os principais problemas de acesso e estratégias para resolvê-los.

### 2.1 Características de uma RSSF

Redes de sensores introduzem um novo padrão de monitoramento, que permite observar e interagir com fenômenos físicos em tempo real [5, 48]. Individualmente cada nodo tem um baixo custo, o que permite a utilização de um grande número de nodos sensores em uma rede.

O potencial de observação e controle do mundo real permite que as RSSFs se apresentem como uma alternativa para diversas aplicações, como: monitoramento ambiental, gerenciamento de infra-estrutura, biotecnologia, monitoramento e controle industrial, segurança pública e de ambientes em geral, áreas de desastres e risco para vidas humanas, transporte, medicina, construção civil e controle militar [35, 36, 48].

Dentre as principais aplicações de RSSF estão o monitoramento e a coleta de dados. Os nodos sensores podem ser lançados sobre áreas remotas (reservas ambientais, oceanos, vulcões, rios, florestas, etc.) formando uma rede. Esta rede sem fio *ad hoc* monitora e coleta dados sobre os fenômenos de interesse, sem intervenção humana.

O processamento dos dados pode ser local ou distribuído. Contudo, as informações coletadas são distribuídas na rede através de um esquema de comunicação de múltiplos saltos (*multi-hop*) até chegarem a um *Access Point* (AP). O AP é o elemento através do qual a RSSF se comunica com o observador ou com outras redes, conforme mostrado na Figura 2 [17]. O

AP se comunica com os nodos da rede e também com outras rede, ele é chamado de *sink node*. O AP, ou *sink node*, também pode estar conectado a uma estação base (BS - *Base Station*), que é uma estação de armazenamento dos dados.

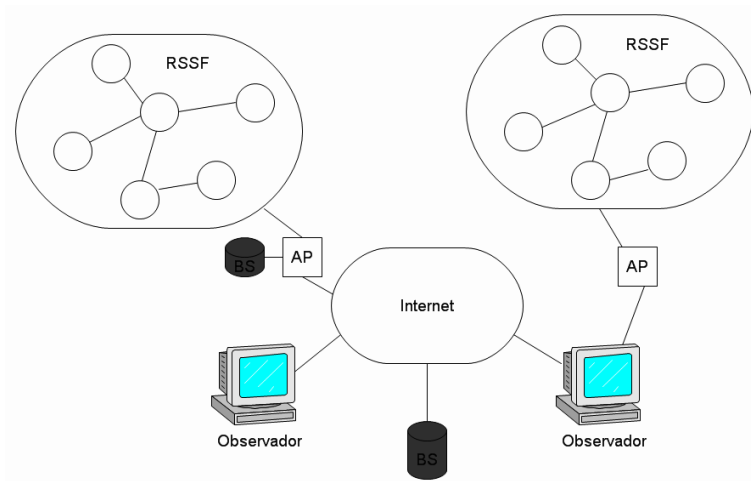


Figura 2: Access Point da RSSF.

A arquitetura de um nodo sensor pode ser descrita, basicamente, por uma unidade de sensoriamento (Sistema de Sensores), uma unidade de processamento que pode ser composta por um microcontrolador ou um DSP ( $\mu$ -controlador), uma unidade de armazenamento (Memória) e uma unidade de comunicação (Sistema de comunicação), ilustrado na Figura 3.

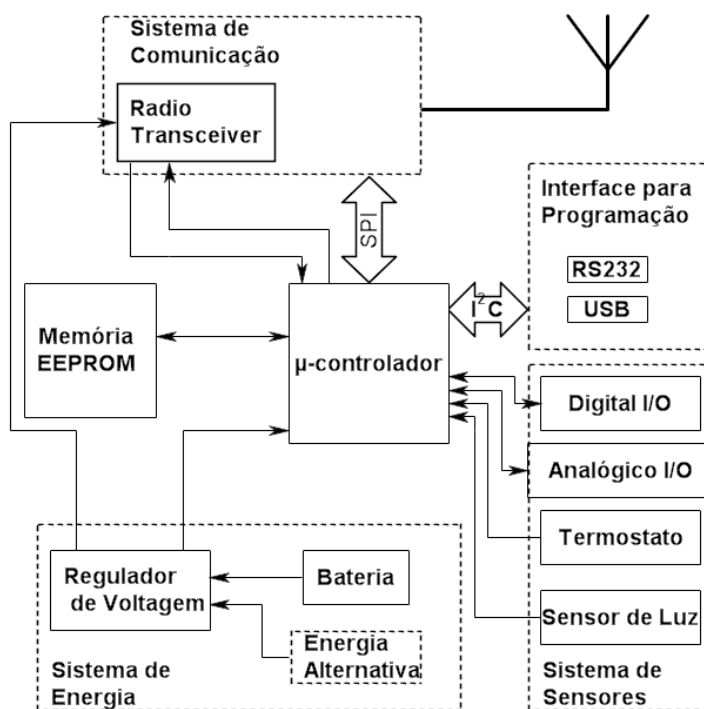


Figura 3: Arquitetura de um nodo sensor.

Um nodo pode incluir também memória não volátil e recursos *on-chip* ou *off-chip*, tais como conversores digital-analógico (DAC), conversores analógico-digital (ADC), UARTs, controladores de interrupção, contadores, temporizadores e sensores [23–25]. Ainda, o sistema de alimentação de energia do nodo sensor (Sistema de Energia) pode funcionar tanto com baterias, quanto por fontes alternativas, como, geradores de vibração, energia solar ou por ondas eletromagnéticas [39, 45, 48].

Diferentes plataformas vêm sendo propostas na construção de RSSFs, tanto por projetos acadêmicos, quanto por projetos comerciais [21, 24, 30]. Um projeto de um nodo sensor é dirigido à aplicação e as várias arquiteturas existentes são adaptadas às restrições específicas de cada aplicação. A tendência é que estes dispositivos se tornem cada vez mais acessíveis com a produção em larga escala de diferentes tipos de nodos sensores.

O consumo do módulo de comunicação é diretamente dependente do tipo de dispositivo de comunicação utilizado como, por exemplo, Rádio Frequência (RF) ou *Infra-Red* (IR). A Tabela 1 mostra uma comparação entre estes dois dispositivos. Apesar do dispositivo de IR ter o consumo bem menor que o de RF, existem algumas desvantagens. Por exemplo, os dispositivos devem estar alinhados para realizar a comunicação. Outra desvantagem é o alcance da comunicação ( $< 10m$ ) [9].

O projeto de uma RSSF é influenciado por muitos fatores que incluem tolerância a falhas, escalabilidade, custo de produção, ambiente operacional, topologia da rede, restrições de *hardware*, meio de transmissão e consumo de energia. Cada um destes fatores exige requisitos específicos na concepção e projeto dos nodos. Esquemas de modulação, estratégias para superar os efeitos da propagação de sinal e projeto de *hardware* de baixo consumo, são requisitos do projeto da camada física. Ainda, os parâmetros de configuração, operação e manutenção variam com os objetivos da aplicação.

Desse modo, ao se realizar um projeto de uma RSSF deve-se levar em consideração alguns requisitos:

- Tamanho reduzido e baixo consumo de potência [39, 45]: este é um ponto chave para nodos sensores, pois tamanho e potência muitas vezes obrigam o uso de dispositivos básicos de processamento, armazenagem, e capacidade de conexão. Certamente, reduzir o tamanho e a potência requerida são fatores decisivos para um projeto de *hardware* de um nodo sensor. Igualmente, o software deve ser o mais eficiente possível no uso do processador e memória, e os dispositivos de comunicação devem consumir pouca potência.
- Operações de alta-concorrência [29, 47]: a principal função de um nodo sensor é coletar e enviar informações, com pouco processamento. Normalmente, com comandos simples como parada ou espera. Por exemplo, um evento ocorre em uma determinada região. A informação pode ser simultaneamente capturada, manipulada e enviada para rede por diversos nodos da mesma região. Isso causa um tráfego excessivo na rede, pois seria

necessário apenas um nodo identificar o evento. Outro problema é que os nodos têm uma pequena capacidade de armazenamento interno, logo armazenar dados não é muito atrativo. Assim, as tarefas de um nodo geralmente envolvem um número maior de tarefas de baixo nível de processamento.

- **Paralelismo Limitado e Controle Hierárquico [24,39]:** o número de controladores independentes, a capacidade dos controladores, e a troca de informações entre processador-memória é muito inferior em um nodo sensor se comparado com um microcomputador padrão IBM-PC. Tipicamente, o sensor ou atuador possui uma interface primitiva para um único chip microcontrolador. Em contraste, sistemas convencionais distribuem processos concorrentes para cada dispositivo, em múltiplos níveis de controle. As limitações de potência, espaço e limitações físicas obrigam os nodos sensores a suportar concorrência intensiva gerenciada por um microcontrolador embarcado.
- **Diversidade de Projetos e Uso [17,36,48]:** redes de sensores são projetadas para satisfazer aplicações específicas, sendo necessário que o *hardware* dê suporte para cada aplicação. Existe uma grande quantidade de aplicações para redes de sensores, e também uma grande variedade de dispositivos que podem ser usados. Para qualquer dispositivo especificado, é importante que somente os componentes de *software* requeridos para a síntese com o *hardware* sejam carregados. Dessa forma, estes dispositivos necessitam um grau não usual de modularidade de software, e devem ser também bastante eficientes.
- **Robustez e Tolerância a Falhas [38]:** uma rede de sensores pode ser composta por um grande número de nodos. Estes dispositivos podem ser usados também em aplicações onde fiquem por longos períodos sem manutenção. Logo, é necessário que os nodos possuam uma certa confiabilidade. As técnicas tradicionais de redundância para aumentar a confiabilidade, são limitadas pelo espaço e pela potência. Embora redundância entre dispositivos seja mais atrativa que redundância dentro do dispositivo, o custo da comunicação entre os dispositivos inviabiliza esta opção. Adicionalmente, pode-se acrescentar confiabilidade à aplicação tolerando individualmente dispositivos falhos. Por fim, o sistema operacional rodando em um simples nodo deve ser não apenas robusto, mas também facilitar o desenvolvimento e a confiabilidade das aplicações.

## 2.2 Camadas de Protocolo em RSSF

A comunicação em RSSF possui características particulares e o uso de protocolos *ad hoc* tradicionais não atende aos requisitos destas redes. Protocolos tradicionais requerem muitos *Kbytes* de memória e alguns recursos do *hardware* indisponíveis em nodos sensores, como, por exemplo detecção da onda portadora. Por este motivo novos protocolos têm sido desenvolvidos para se adequarem as necessidades e limitações das RSSFs.

O estudo dos protocolos de comunicação em RSSF é feito por camadas, como sugerido pelo modelo *OSI/ISO* [43]. Contudo, devido a simplicidade destes dispositivos, a pilha completa da arquitetura *OSI/ISO* não é necessária. Sendo assim a pilha de protocolos para RSSFs pode ser reduzida nas camadas Física, Enlace, Rede e de Transporte, como mostra a Figura 4.

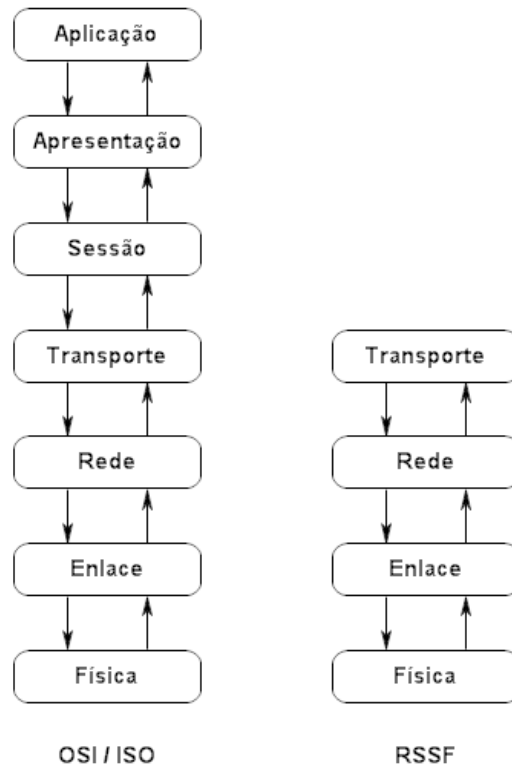


Figura 4: Modelo OSI/ISO.

### 2.2.1 Camada Física

A camada física é responsável por transmitir *bitstream* puros nos dois lados da comunicação. Isto é, deve converter uma seqüência de dígitos binários para sinais elétricos (cabo), sinais de luz(IR), ou eletromagnéticos(RF) [40]. Em RSSF, normalmente, são utilizados dispositivos de comunicação RF ou IR, sendo os dispositivos RF mais utilizados.

A comunicação por IR não necessita de antena e seu consumo é bem menor se comparado com a RF, conforme apresentado na Tabela 1 do Capítulo 1. Porém, a grande desvantagem da comunicação IR em relação a RF é a necessidade do alinhamento dos dispositivos transmissor e receptor para ocorrer a comunicação.

Na comunicação RF os dispositivos são espalhados por uma determinada área sem a necessidade de alinhamento, pois a comunicação é realizada por ondas eletromagnéticas, que acabam formando um raio de comunicação em torno do dispositivo [17]. Outra vantagem da comunicação RF é o maior alcance, que dependendo da frequência e potência do dispositivo pode chegar a 10000m [32], sendo que o alcance de dispositivos IR não ultrapassa 10m [9].

Vários aspectos afetam o consumo de energia na comunicação RF, tais como o tipo de modulação, taxa dos dados, frequência e energia de transmissão. Em geral, o módulo de RF pode operar em quatro modos distintos: transmitindo, recebendo, *idle* (módulo de RF não está enviando e nem recebendo mensagens) e *sleep* (módulo em *standby*). Em alguns casos, mesmo estando no estado *idle*, um módulo de RF consome energia como se estivesse no modo de recepção, nestes casos é importante traçar outras estratégias para economia de energia [36].

### 2.2.2 Camada de Enlace

A camada física simplesmente envia e recebe a sequência de bits (*bitstream*). Já a função da camada de enlace é detectar e corrigir erros na transmissão de dados, particionar os dados em *frames* e controlar o fluxo destes *frames* entre os dois lados da comunicação [40]. Os protocolos desta camada são conhecidos como protocolos de acesso ao meio ou simplesmente protocolos MAC (*Medium Access Control*).

Os requisitos da camada de enlace variam de acordo com o tipo de aplicação. Por exemplo, em uma RSSF dirigida a eventos os nodos podem permanecer com os transceptores inativos por longos períodos de tempo [48]. Porém, quando um fenômeno é detectado vários nodos passam repentinamente para o estado ativo. Quando a RSSF é densa, composta por vários nodos sensores na mesma área, a ocorrência do evento levará diversos nodos a acessar o meio ao mesmo tempo para transmitir seus dados, o que pode causar colisões.

Apesar de RSSFs serem diferentes das redes tradicionais, elas herdam problemas de comunicação das redes sem fio. Na maior parte dos casos, as redes sem fio empregam um dispositivo de comunicação de rádio de um único canal com modo de comunicação *half-duplex*, ou seja, a comunicação é bidirecional e não simultânea. O rádio utilizando a mesma frequência pode somente transmitir ou receber informações a cada instante de tempo. Sendo assim, o método empregado nas tradicionais redes Ethernet, CSMA/CD não pode ser empregado em redes sem fio [32], uma vez que é baseado em comunicação *Full-Duplex*.

As restrições dos protocolos empregados em RSSFs são ainda maiores que as restrições das redes MANETs (*Mobile Ad-Hoc Network*) [47], devido ao *hardware* utilizado pelas mesmas. Desta forma, não existe suporte pelo *hardware* para detecção de portadora, detecção de colisão, enquadramento específico, codificação ou balanceamento de energia [17]. O rádio utilizado possui características de baixa potência, largura de banda limitada e um único canal na frequência base ISM .

Tais condições, assim como outras características particulares das RSSFs, motivam um controle de acesso ao meio. Existem vários protocolos de acesso ao meio projetados para RSSFs disponíveis na literatura, como por exemplo, protocolo S-MAC [47], protocolo ARC [46], protocolo B-MAC [34], protocolo TRAMA [37] que serão apresentados com mais detalhes no Capítulo 3.

### 2.2.3 Camada de Rede

A principal função da camada de rede é prover o serviço de roteamento. Este serviço pode ser definido como o processo pelo qual a rede consegue identificar o destinatário das mensagens e encontrar um caminho entre a origem e o destino desta mensagem. Este processo é de fundamental importância em todas as redes de computadores, e em RSSFs não é diferente [17,36,43].

Existem diversas formas de fazer o roteamento entre os nodos em RSSFs, sendo que a eficiência deste roteamento está diretamente ligada na forma como este roteamento é feito. Geralmente, as RSSFs oferecem um serviço de comunicação *multi-hop*. Contudo, o tipo de aplicação destas redes e os requisitos de roteamento diferem de outras redes em alguns aspectos, como:

- A forma de comunicação típica de uma RSSF é unidirecional no sentido dos nodos fontes para o AP.
- Os dados dos nodos fontes em geral retratam um fenômeno comum aos nodos sensores próximos, portanto, existe a probabilidade de redundância dos dados transmitidos.
- Normalmente os nodos sensores possuem pouca ou nenhuma mobilidade. Sendo que as características de formação espontânea das rede *ad hoc* serve para suprimir nodos que saem da rede por falta de energia, por exemplo.
- A principal restrição nas RSSFs é a limitação de energia. Esta limitação é bem mais crítica nestas rede, pois os dispositivos são menores (as baterias também devem ser menores) e elas supostamente funcionarão em locais de difícil acesso que não prevê a recarga ou troca das baterias [47].

Portanto, isso motiva encontrar novas soluções de endereçamento que atendam as restrições das RSSFs considerando suas particularidades. Algumas alternativas são propostas como endereçamento espacial [1], difusão direcionada [27], endereçamento baseado em atributos (SPINS [33]) e endereçamento de transações (RETRI [6]).

### 2.2.4 Camada de Transporte

O trabalho da camada de transporte é prover um serviço de comunicação *site-to-site*, isto é, a camada de transporte aceita mensagens arbitrariamente, abstraindo todos os detalhes da comunicação das camadas anteriores.

Ao contrário das redes tradicionais, o uso de protocolos de transporte em RSSFs nem sempre é possível. A maioria das aplicações nestas redes admitem a perda de dados. Assim, um mecanismo elaborado para garantia de envio de dados não é justificado e pode até causar um gasto desnecessário de energia [17,36].

Apesar disso, algumas aplicações ou tarefas na RSSF necessitam de entrega confiável de dados, como, por exemplo, serviços de reprogramação de nodos e funções de gerenciamento, como desligamento de nodos. Para resolver este problema, em alguns casos as mensagens são re-enviadas até a confirmação do recebimento.

Adicionalmente, alguns protocolos da camada de roteamento utilizam técnicas com o intuito de diminuir a perda de dados, o que melhora o desempenho da camada de transporte. A difusão direcionada é uma destas técnicas, os dados são encaminhados através de vários caminhos da rede com a perspectiva de diminuir a perda de dados. Contudo, estas técnicas aumentam a consumo de energia diminuindo o tempo de vida da rede e limitando o número de aplicações da RSSF.

## 2.3 Técnicas de Acesso ao Meio Em RSSFs

O objetivo dessas técnicas é controlar o acesso ao meio dos dispositivos que estão presentes na comunicação. Em comunicação *wireless*, quando dois dispositivos estão ocupando o mesmo recurso ao mesmo tempo ocorre uma colisão e a mensagem enviada é perdida, o que obriga um re-envio da mesma.

As técnicas de acesso ao meio estabelecem regras para um melhor aproveitamento dos recursos de comunicação. Com estas técnicas cada dispositivo presente na comunicação tem o direito de ocupar o meio de uma maneira organizada, garantindo a comunicação e evitando freqüentes colisões.

As RSSFs possuem características próprias que um protocolo de acesso ao meio deve levar em consideração e sempre que possível tirar vantagens destas características. Os nodos da RSSF formam um sistema colaborativo, normalmente, servindo a um pequeno número de aplicações. Com isso, conceitos de *fairness* (seção 2.4.3) não são tão importantes. Ainda, é pouco provável que um evento utilize toda a largura de banda da rede. Em muitas aplicações em RSSF os nodos ficam por longos períodos a espera que um evento seja detectado. Os nodos não necessitam estar ativos por todo o período de tempo. Podemos assumir que em uma RSSF os vizinhos de um nodo são relativamente fixos.

Esta seção apresenta as técnicas mais conhecida de acesso ao meio, FDMA (*Frequency-Division Multiple Access*), TDMA (*Time Division Multiple Access*), CDMA (*Code Division Multiple Access*) e CSMA/CA - (*Carrier Sense Multiple Access with Collision Avoidance*).

### 2.3.1 FDMA - *Frequency-Division Multiple Access*

No acesso ao meio por divisão de freqüência, uma faixa de freqüência do recurso de comunicação é destinada a um usuário por um longo período de tempo ou permanentemente. O recurso de comunicação pode conter simultaneamente diversos sinais separados por faixas de freqüência, como ilustrado na Figura 5. A faixa de freqüência 1 contém sinais que operam entre



as frequências  $f_0$  e  $f_1$ , a segunda faixa de frequência contém sinais entre  $f_2$  e  $f_3$ , e assim sucessivamente dentro dos limites de operação. A região entre duas faixas de frequência é chamada de faixa de guarda, e funciona como uma zona de tolerância, que reduz a interferência entre os sinais de dois canais de frequência adjacente, os sinais nessa faixa são descartados.

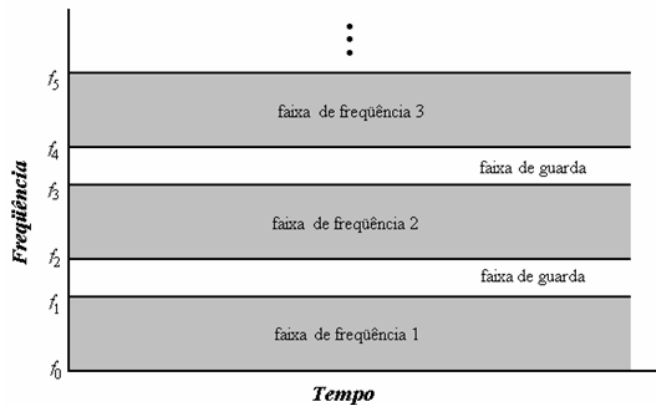


Figura 5: Multiplexação por divisão de frequência.

Para que o Recurso de Comunicação (RC) seja dividido por frequência é necessário que os sinais sejam modulados em frequências diferentes. Sinais modulados em frequências diferentes podem ser transmitidos simultaneamente [41]. Contudo, para que o receptor receba a mensagem este deve estar modulado na mesma frequência.

Normalmente, os dispositivos de RF utilizados em nodos sensores não possuem suporte a multi-frequências. Para receber uma mensagem em uma frequência diferente o dispositivo deve alterar a configuração do módulo RF. Cada nova faixa de frequência em que o dispositivo deve operar exige um circuito adicional [32], o que faz com que este tipo de técnica não seja muito comum em RSSF.

### 2.3.2 CDMA - *Code-Division Multiple Access*

No método CDMA cada nodo na rede tem um código diferente dos seus vizinhos, e podem acessar a rede a qualquer momento, Figura 6. Todos os nodos são assíncronos e deve existir um receptor ativo todo tempo.

O projeto PicoRadio [21], pro exemplo, propõe um protocolo multi-canal de acesso denominado de "*~ 30 code division multiple access*" CDMA. Cada nodo da rede tem um código diferente dos seus vizinhos. Todos os nodos são assíncronos. Contudo, cada nodo tem um dispositivo receptor *ultra-low-power* que permanece ativo, escutando o meio, o tempo todo. Quando uma mensagem de "*wakeup beacon*" com o código do nodo é percebida pelo receptor "*wakeup beacon*", o nodo ativa seu módulo RF para começar a comunicação.

A viabilidade desta proposta depende do dispositivo que desperta o nodo. Este deve ser um receptor com consumo muito baixo, pois deve ficar sempre ativo. No PicoRadio dispositivo

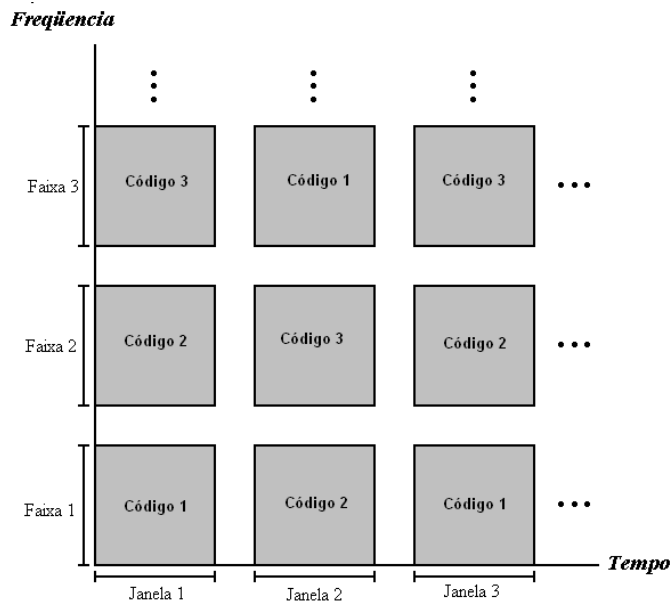


Figura 6: Multiplexação por divisão de Código

proposto foi um amplificador de rádio frequência simples, seguido de um filtro e um detector de energia, com consumo baixo, em torno de  $1\mu W$  [21].

### 2.3.3 TDMA - *Time-Division Multiple Access*

No compartilhamento dos recursos de comunicação baseado em divisão de tempo apresentado pelo TDMA, cada usuário tem o direito de utilizar todo o canal por um curto período de tempo, chamado janela de tempo (*slot*). A Figura 7 ilustra um exemplo da utilização deste método com três usuários. Cada usuário tem direito a uma janela de tempo. O tempo entre cada janela é chamado de tempo de guarda. Sendo utilizado como uma zona de tolerância para reduzir a interferência entre os sinais de duas janelas adjacentes.

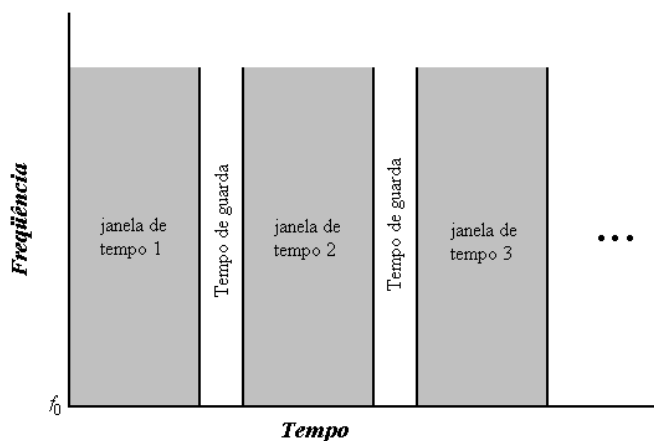


Figura 7: Multiplexação por divisão de tempo

O tempo que todos os usuários levam para utilizar o canal é chamado de *frame*. A Figura 8 mostra a estrutura básica de um *frame*. Os *frames* são fixos e se repetem ao longo do tempo. Uma mensagem na janela de tempo é dividida em duas partes, uma para controle e outra para dados. A parte de controle normalmente contém informação para realizar a sincronização, endereçamento e controle de erro. A parte de dados contém informação a ser transferida.

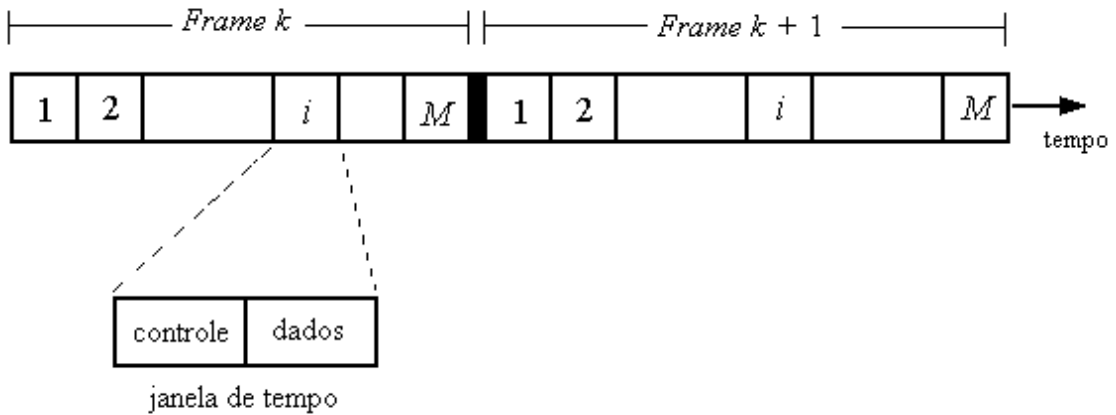


Figura 8: Estrutura do *frame*.

O método TDMA estruturado com janelas de tempo fixas é eficiente quando é possível prever a comunicação com antecedência, e quando o tráfego do canal é intenso. Porém, quando o tráfego é pequeno ou esporádico, fixar o tamanho das janelas de tempo pode significar um desperdício dos recursos de comunicação. A Figura 9 ilustra um exemplo de um sistema composto de quatro usuários por *frame*. No primeiro *frame* o usuário C não transmite dados. Durante o segundo *frame* o usuário B não envia nada. E no terceiro *frame* é o usuário A que está inativo. No método TDMA as janelas de tempo são designadas previamente aos usuários. Se o proprietário da janela não a utiliza esta será desperdiçada, Figura 9(b).

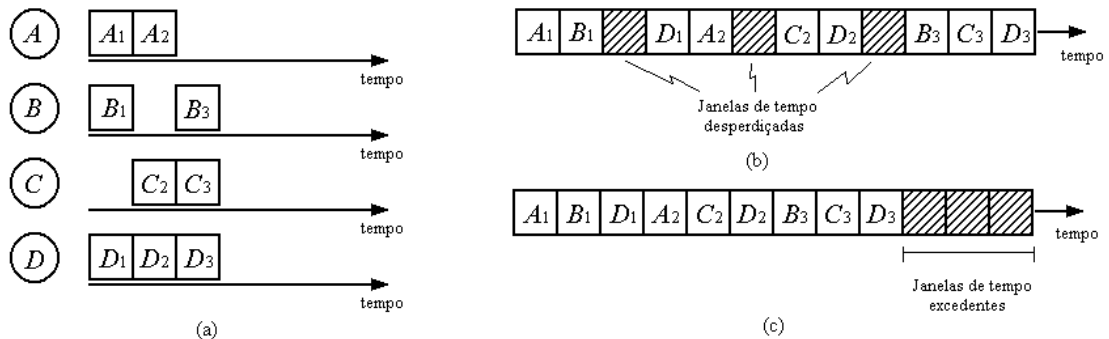


Figura 9: TDMA pacotes recebidos.

Quando as exigências do sistema são desconhecidas, como no caso da Figura 9, pode ser mais eficiente utilizar um compartilhamento dinâmico ou invés de um fixo. Alguns méto-

dos como multiplexação estatística e concentradores, podem trazer um melhor desempenho ao sistema, como pode ser visto na Figura 9(c). Vários projetos de nodos sensores utilizam a técnica de acesso ao meio TDMA, dentre eles podemos citar o projeto  $\mu$ AMPS ( *$\mu$ -Adaptive Multi-domain Power Aware Sensor*) desenvolvido pelo Massachusetts Institute of Technology (MIT) [22].

### 2.3.4 CSMA/CA - *Carrier Sense Multiple Access with Collision Avoidance*

O protocolo de acesso ao meio CSMA/CA tenta evitar a ocorrência de colisões definindo a comunicação entre duas estações, emissora e transmissora, através de um diálogo de três passos RTS/CTS/DATA (*Request-To-Send/Clear-To-Send/Data*) [31]. A Figura 10 ilustra o funcionamento deste processo, a estação A deseja enviar dados para a estação B. Então um pacote de controle RTS é enviado. Quando a estação B recebe este sinal, envia um CTS. O CTS significa que a estação B está pronta para receber os dados. Logo que a estação A recebe CTS de B, é iniciada a transmissão do pacote de dados.

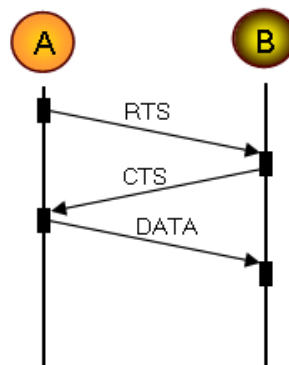


Figura 10: RTS/CTS/DATA.

As demais estações na rede ao escutarem pacotes de controle RTS ou CTS não destinados a elas, devem bloquear seus rádios para transmissão até o final da operação. Este procedimento diminui a probabilidade de ocorrência de colisões na rede. Vale lembrar que nas redes sem fio as colisões ocorrem somente no receptor, já que a estação base ao transmitir não tem como escutar o canal.

## 2.4 Acesso ao Meio em RSSFs

As técnicas de acesso ao meio apresentadas possuem vantagens e desvantagens em relação a sua implementação em RSSF. No método TDMA os nodos negociam através de uma janela de comunicação. Cada usuário dispõe de toda a faixa de frequência disponível durante um determinado período (janela de tempo). Isso causa um problema, uma vez que, em uma rede densa os nodos podem permanecer ativos por muito tempo esperando sua janela. Ainda, para

garantir que não haja colisão, existe a necessidade de implementação de uma referência global de tempo para toda a rede. Para implementação das técnicas FDMA e CDMA existe a exigência de requisitos de *hardware*, o que acaba aumentando o custo dos nodos sensores.

No caso do protocolo CSMA/CA, apesar da implementação ser mais complexa que a FDMA e TDMA, não há necessidade de requisitos de *hardware* adicionais, o que o torna uma boa opção para as RSSFs. Entretanto, este também apresenta algumas dificuldades, como, o problema do terminal escondido e da estação exposta.

### 2.4.1 Problema do terminal escondido

O problema do terminal escondido é uma situação que ocorre nos protocolos CSMA/CA que acarreta colisão de dados [19, 31]. A Figura 11 mostra esta situação. A estação A envia RTS para a estação B que está dentro do seu alcance de rádio. A estação C, que está dentro do alcance de rádio de B, mas fora do alcance do rádio de A, também transmite um RTS para B. A estação B irá receber os dois pacotes ao mesmo tempo e não poderá interpretar os dados enviados. Sempre que ocorre uma colisão os dados devem ser enviados novamente, isso gera um consumo desnecessário de energia.

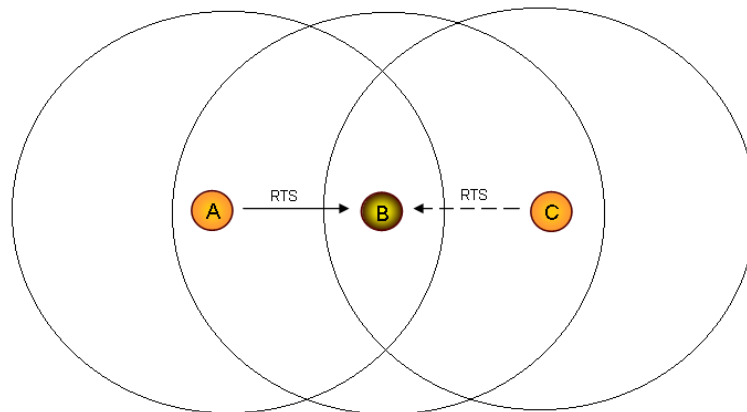


Figura 11: Problema do terminal escondido.

### 2.4.2 Problema da estação exposta

As colisões no protocolo CSMA/CA também podem ocorrer por outro motivo conhecido como estação exposta [17]. A Figura 12 mostra uma tentativa de comunicação da estação B. Neste caso, a estação B transmite um pacote de controle RTS para a estação A, solicitando uma transmissão. A estação C está pronta para transmitir para a estação D, porém ela escuta o pacote de controle da estação B e bloqueia seu rádio até que a transmissão de B acabe.

A transmissão entre C e D não causaria nenhuma interferência na comunicação entre A e B, pois a estação C não está no alcance de rádio de A. Se a estação C transmitisse simultaneamente com a estação B não ocorreria colisão. Contudo, o recebimento do pacote RTS pelo C não

continha informações completas sobre a transmissão, impedindo a estação de transmitir, por este motivo o problema da estação exposta também é considerado uma colisão.

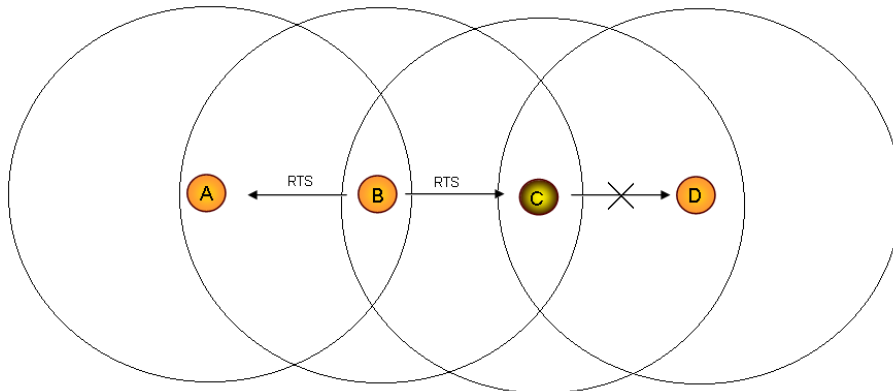


Figura 12: Problema da estação exposta.

### 2.4.3 *Fairness*

Nas tradicionais redes *wireless* de dados o conceito de *fairness* está relacionado à distribuição justa de acesso. Cada aplicação deve ter oportunidade igual de tempo para acesso ao meio, enviando ou recebendo pacotes. Porém, as RSSF, normalmente, são direcionadas a eventos e em algum momento um nodo terá mais dados para enviar do que os demais. Em outras aplicações todos nodos cooperam para realizar uma tarefa comum. Por este motivo, poucas aplicações em RSSF utilizam *fairness* [17,46].

### 2.4.4 Latência

O controle da latência é diretamente dependente da aplicação da RSSF. Por exemplo, em aplicações de vigilância ou monitoramento, algumas regiões da rede podem manter seus nodos ativos por um longo tempo. Ao passo que nodos de uma outra região podem permanecer inativos a espera de um evento.

A aplicação pode oferecer uma determinada tolerância à latência das mensagens, isso porque, a velocidade da rede é tipicamente maior que a ocorrência dos eventos físicos. Também, durante um período em que não existe um evento, o fluxo de troca de mensagens na rede é muito baixo. Conseqüentemente, a velocidade mínima que a rede deve reagir é determinada pela atividade de ocorrência dos eventos.

### 2.4.5 Distância entre os nodos

Levando em consideração estas características podemos definir, basicamente, três cenários: Primeiro, os nodos são depositados de maneira a aproveitar ao máximo o raio de alcance do módulo de RF, isto é, os nodos são depositados de maneira que consigam abranger a maior área

possível. Segundo, a distância entre os nodos é determinada pelas necessidades da leitura dos sensores dos nodos, ou seja, quanto mais fina a medida, mais próximos os nodos devem ficar. Por último, os nodos da rede são depositados sem uma referência padrão de distância entre eles. Este cenário é bastante utilizado em aplicações de monitoramento de regiões remotas, onde os nodos são lançados sobre estas áreas para formar uma rede de coleta.

### 2.4.6 Período *listen/sleep*

Em aplicações de RSSF é natural os nodos permanecerem sem nenhuma tarefa por um longo período de tempo. Durante este período a taxa de transmissão de dados é muito baixa. Logo, não é necessário manter os nodos ativo o tempo todo [34, 44, 47]. Pensando no consumo de energia, é possível alternar o ciclo de operação dos nodos da rede entre dois períodos, ativo e inativos. Quando o nodo estiver em seu período ativo (*listen*) ele pode realizar as comunicações, e quando estiver no período inativo (*sleep*) deve manter seu módulo de rádio desligado, economizando energia. Dividindo o ciclo de operação em dois períodos todos os nodos da rede alternam entre os períodos *listen* e *sleep*, e cada nodo permanece um período de tempo no estado *sleep* e outro no estado *listen*, este ciclo completo é chamado de *frame*, Figura 13.

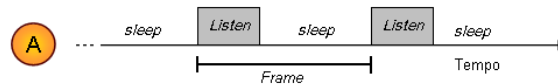


Figura 13: Período *listen/sleep* para um nodo sensor.

### 2.4.7 Sincronização entre nodos

Com a utilização da divisão do ciclo de operação em *listen/sleep* é comum dois nodos desejarem realizar uma comunicação e estarem fora de sincronia. A Figura 14 ilustra dois nodos nesta situação. Neste caso, quando o nodo A está em seu período *listen* preparado para realizar comunicação com o nodo B, o nodo B está em seu período *sleep* com seu módulo de comunicação desligado. Como o nodo A e nodo B entram no período *listen* em tempos diferentes, para que a comunicação prossiga é necessário haver uma sincronização entre os nodos.

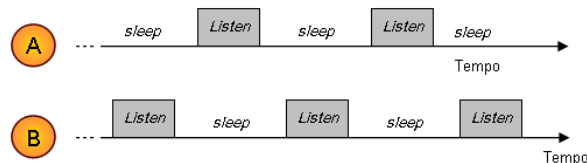


Figura 14: Nodos com período *listen/sleep* diferentes.

Uma solução para a sincronização entre nodos é fazer com que o nodo que deseja enviar transmita um pacote SYNC em *broadcasting* para todos seus vizinhos. Este pacote contém o

período do *frame* em que o nodo estará em seu estado *listen*. O nodo informa para seu vizinho o seu período de *listen*, assim todos os nodos vizinhos podem comunicar mesmo que tenham tempos diferentes.

Outra solução é a utilização de um mecanismo que atue dentro da rede como um mediador (MD) entre os nodos. Este mediador é um nodo vizinho aos dois nodos que querem estabelecer a comunicação. A sincronização entre nodos que ocorre somente quando existe a necessidade de comunicação é conhecida como sincronização dinâmica [17].

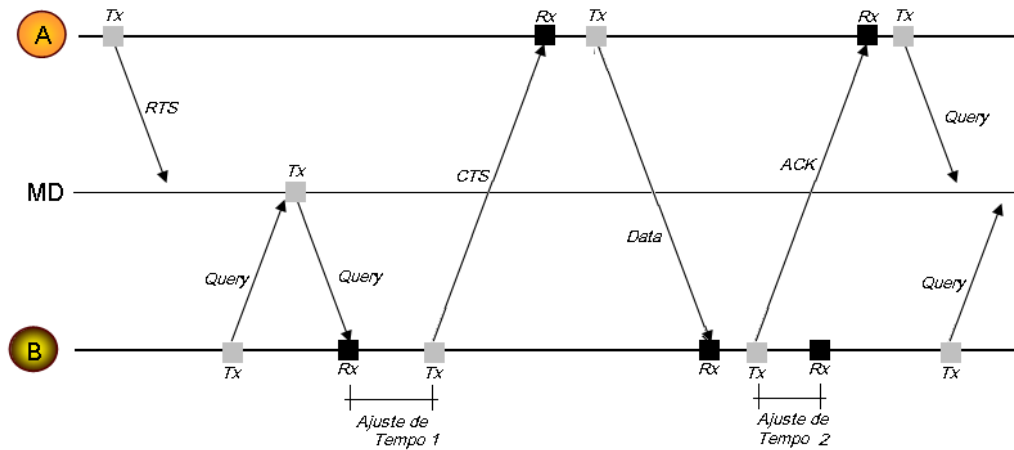


Figura 15: Mediador entre nodos.

A Figura 15 mostra um exemplo de um mediador. Assumindo que o nodo A deseja enviar uma mensagem para o nodo B, devem ser os seguintes passos:

- Passo 1: o nodo A envia RTS e aguarda por uma mensagem de resposta. O nodo B não está em sincronismo com o nodo A, portanto não é capaz de receber diretamente a mensagem de controle RTS do nodo A. O MD deve gravar o RTS enviado pelo nodo A, esta mensagem inclui uma identificação de ambos os nodos, A e B.
- Passo 2: então o nodo B acorda e transmite um *query*. O MD grava a diferença de *offset* entre o nodo A e o nodo B, isto é, a diferença entre o RTS do nodo A e o *query* do nodo B. Logo que recebe o *query* do nodo B transmite uma mensagem de *query* para o mesmo, indicando que existe uma mensagem a ele destinada do nodo A, incluindo o *offset* entre os dois nodos.
- Passo 3: quando o nodo B recebe o *query* vindo de MD, o mesmo ajusta seu tempo para entrar em sincronismo com o nodo A. O nodo B agora transmite um pacote de controle CTS para A, durante o período de recebimento de A. Então os dois nodos podem começar a transmissão de dados.
- Passo 4: Depois do nodo A receber uma mensagem de reconhecimento ACK, o mesmo retorna ao seu ciclo normal de transmissão de pacotes. Para evitar colisões com o nodo



A, o nodo B reajusta seu relógio para o estado original, retomando seu ciclo normal de transmissão de pacotes.

Vários aspectos afetam o consumo de energia em RSSFs, colisão é um deles. Quando ocorre uma colisão o pacote transmitido não chega ao seu destino, e deve ser re-enviado, causando um consumo extra de energia. Além disso, fatores como a técnica de acesso ao meio, sincronização utilizada, tipo de modulação, taxa de envio de dados e energia de transmissão também devem ser levados em consideração para reduzir o consumo de energia. O dispositivo de RF empregado em nodos sensores também deve se ajustar à escassa oferta de energia, sendo normalmente dispositivos simples.

As técnicas apresentadas neste capítulo, como, por exemplo, período *Listen/Sleep* e sincronização entre outras foram utilizadas na elaboração e no desenvolvimento de funcionalidades do presente trabalho.



## 3 Trabalhos Relacionados

Este capítulo apresenta os principais protocolos de acesso ao meio desenvolvidos para *RSSFs*. O objetivo é apresentar o estado da arte e os principais conceitos relacionados. Estes trabalhos foram estudados visando auxiliar na especificação do protocolo apresentado nesta dissertação. Ao longo deste capítulo serão descritos os projetos S-MAC [47], T-MAC [44], ARC [46], DE-MAC [28], TRAMA [37], apresentando suas principais características, suas inovações, vantagens e desvantagens.

### 3.1 Protocolo S-MAC

O S-MAC (*Sensor-MAC*) [47] é um protocolo de acesso ao meio projetado especificamente para RSSF. O acesso ao meio é baseado em alocação dinâmica de canal. Utiliza sincronização para coordenar os modos de operação de rádio (*idle/sleep*). O S-MAC é dedicado a RSSFs dirigidas a eventos, com coleta periódica de dados, insensível a latência da rede e com baixa taxa de envio de mensagens. O protocolo S-MAC procura ser eficiente em energia reduzindo o consumo dos principais eventos responsáveis pelo desperdício, como:

- *Colisões*: Para evitar as colisões, problema do terminal escondido e problema da estação exposta, o S-MAC emprega a mesma técnica utilizada no protocolo IEEE 802.11 - DCF (*Distributed Coordination Function*) [31], usando um diálogo de comunicação RTS/CTS/DATA/ACK. Caso a colisão ocorra o protocolo utiliza o algoritmo BEB (*Binary Exponential Backoff*) que aguarda um tempo para enviar novamente a mensagem.
- *Overhearing*: Quando os nodos escutam transmissões de pacotes destinados a outros nodos, para evitar o *overhearing*, o S-MAC coloca o nodo em modo *sleep* quando verifica que o pacote não é destinado a ele.
- *Overhead*: Pacotes de controle são utilizados para reserva do canal de comunicação, reconhecimento de pacotes de dados e sincronização. Estes pacotes de controle aumentam

o tráfego da rede. O S-MAC reduz o tamanho dos pacotes de controle para diminuir o *overhead*.

- *Idle listening*: O nodo fica escutando o meio mesmo quando não existe tráfego na rede. O S-MAC utiliza um ciclo de operação reduzido com tempos fixos de atividade (*listen*) e de repouso (*sleep*), mostrado na Figura 16. Para uma maior economia de energia o tempo de atividade é menor que o tempo de repouso (cerca de 10%).

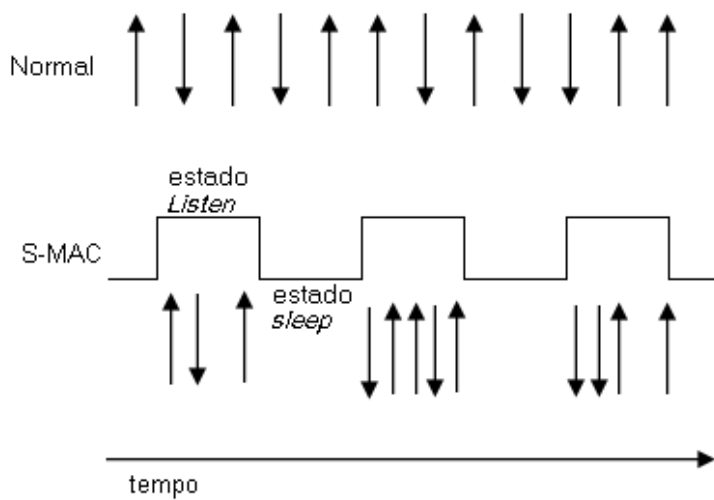


Figura 16: Protocolo S-MAC.

A Figura 16 mostra o ciclo de operação do protocolo S-MAC. As setas indicam transmissão e recebimento de mensagens. O nodo utiliza seu período ativo (*listen*) para enviar e receber mensagens. Durante o período de repouso (*sleep*) as mensagens são armazenadas e transferidas em rajada no início do período ativo.

Todos os nodos são livres para escolher seu próprio período *listen/sleep*. Porém, para reduzir o *overhead* os nodos vizinhos são sincronizados juntos, isto é, entram em seu período *listen* e *sleep* ao mesmo tempo.

Os nodos mudam seu ciclo *listen/sleep* periodicamente, enviando um pacote SYNC em *broadcast* para todos os seus vizinhos. Um nodo informa para seus vizinhos o tempo de seu período de *listen*, deste modo assume-se que todos os vizinhos podem comunicar até mesmo com tempos diferentes.

Uma característica do S-MAC é formar uma topologia *peer-to-peer* (ponto-a-ponto) sem um protocolo de clusterização *clustering*. Ele é coordenado por intermédio de um cabeçalho de *cluster*. Os nodos formam um *cluster* virtual sincronizado, mas comunicam-se diretamente ponto-a-ponto. Com isso a rede fica mais robusta, aceitando mudanças de topologia, ocasionadas por quedas ou mudanças físicas de nodos. A desvantagem é um aumento da latência devido ao período *sleep*.

O S-MAC aplica a técnica de *message passing* para reduzir a latência durante a contenção em aplicações que requerem armazenamento de informações para processamento na rede. Esta técnica permite a transmissão de mensagens longas, que são divididas em pequenos fragmentos e enviadas em rajada. Este protocolo obtém considerável redução do consumo de energia, prolongando o tempo de vida da rede.

## 3.2 Protocolo T-MAC

O protocolo Time-out-MAC (T-MAC) [44] é baseado em contenção para o controle de acesso ao meio em RSSFs, protocolo IEEE 802.11. O T-MAC foi desenvolvido para aplicações dirigidas a eventos que possuam baixa taxa de entrega de mensagens e sejam insensíveis a latência. Trabalha com transmissão contínua ou periódica de dados. O principal objetivo deste protocolo é ser eficiente em energia considerando as limitações de *hardware* do nodo e os padrões de comunicação, isto é, troca de mensagens entre os nodos vizinhos e entre os nodos e a estação base. Assim como o S-MAC o ciclo de operação é reduzido, possuindo tempos de atividade (*listen*) e de repouso (*sleep*) variáveis, que se adaptam à carga da rede. A variação dinâmica do tempo ativo é obtida pela implementação de um temporizador que desliga o rádio do nodo ao verificar que não existe transmissão durante um intervalo de tempo.

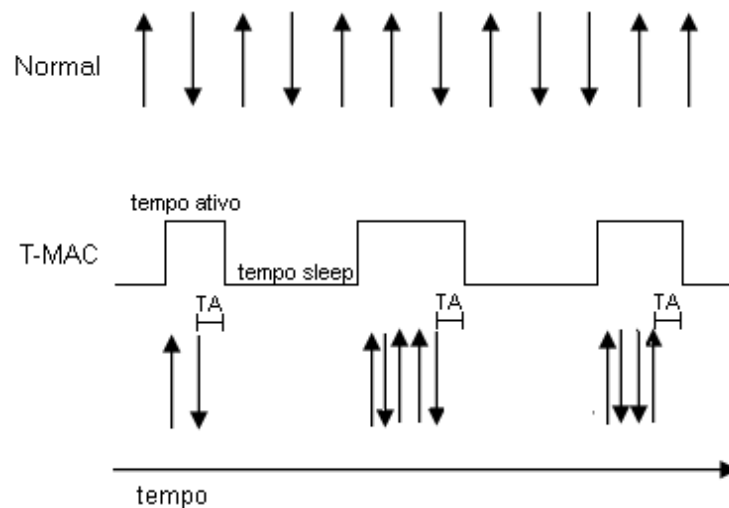


Figura 17: Protocolo T-MAC.

A Figura 17, descreve o ciclo adaptativo do protocolo T-MAC, onde as setas indicam transmissão e recepção de mensagens. A idéia do T-MAC é reduzir o tempo de *idle listening* para diminuir o consumo de energia. As mensagens recebidas durante o tempo de repouso são armazenadas e transferidas em rajadas no início do tempo ativo. O nodo escuta a rede, transmite e recebe dados durante seu tempo ativo. O temporizador determina o final do tempo ativo quando não ocorrem eventos durante um tempo TA.

A ativação por eventos ocorre por:

- início periódico de quadro;
- recepção de dados no rádio;
- final da transmissão de seus vizinhos;
- final da transmissão de seu próprio pacote de dados ou recebimento de ACK;
- por detecção de sinal no rádio (RSSI - *Received Signal Strength Indicator*).

Os nodos se comunicam com o diálogo RTS/CTS/DATA/ACK para evitar colisões e obter transmissão confiável. De maneira semelhante ao S-MAC, o T-MAC utiliza agrupamentos virtuais que seguem escalas para sincronizar seu ciclo de operação. Os nodos transmitem suas escalas para os seus nodos vizinhos através de pacotes SYNC. A recepção de pacotes RTS ou CTS é suficiente para renovar o tempo  $T_A$ . O intervalo de tempo  $T_A$  deve ser suficiente para receber pelo menos o início de um pacote CTS.  $T_A$  pode ser obtido por:

$$T_A > C + R + T \quad (3.1)$$

Onde:

$C$  = tamanho do intervalo de contenção

$R$  = tamanho do pacote de *RTS*

$T$  = é um curto intervalo entre o fim do pacote de *RTS* e o começo do pacote de *CTS*.

Um problema encontrado no T-MAC, é quando um nodo dorme enquanto um outro nodo ainda tem mensagem para ele. Este problema é conhecido como o problema de dormir cedo e pode ser visualizado na Figura 18, onde o nodo D dorme antes de C enviar um RTS. Este problema pode ser resolvido de duas maneiras:

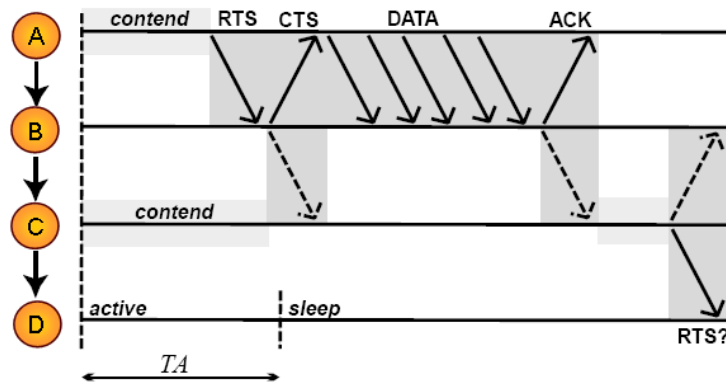


Figura 18: Problema de dormir cedo.

- Um nodo, ao escutar um pacote CTS destinado a outro nodo, envia imediatamente aos seus vizinhos um pacote designado de FRTS (*Future RTS*) e aumenta o tempo de TA;
- Outra maneira é usar um esquema de priorizar o esvaziamento do *buffer* quando este estiver perto de sua capacidade limite. Um nodo ao receber um RTS ao invés de responder com um CTS, transmite as mensagens armazenadas em seu *buffer* para o nodo de destino.

O T-MAC consegue ser mais eficiente em energia que o S-MAC, mas é extremamente limitado em largura de banda não sendo recomendado para aplicações sensíveis a este requisito.

### 3.3 Protocolo ARC

O ARC (*Adaptive Rate Control*) [46] tem como metas a alocação de largura de banda, *fairness* e eficiência em energia para condições de baixo e alto tráfego da rede. O protocolo propõe um mecanismo que passivamente adapta a taxa de transmissão do tráfego de passagem e do tráfego de dados originados. Este método utiliza um algoritmo que exige uma pequena capacidade computacional, que está dentro das limitações do *hardware*.

Um novo esquema CSMA/CA é proposto pelo protocolo ARC, adicionando um atraso aleatório antes do nodo verificar o meio para enviar uma mensagem, o que evita repetidas colisões devido ao comportamento sincronizado do nodo na ocorrência de um evento. O ARC em conjunto com este novo mecanismo CSMA/CA fornece controle efetivo de acesso ao meio sem a utilização de pacotes explícitos de controle. Este esquema garante *fairness* e mantém uma razoável largura de banda, sendo eficiente em energia para situações de baixo tráfego.

### 3.4 Protocolo DE-MAC

O protocolo DE-MAC (*Distributed Energy aware MAC*) [28] provê gerenciamento e balanceamento de energia em RSSFs. É um protocolo que emprega alocação estática de canal TDMA, para tratar colisões e *overhead* de pacotes de controle. Utiliza o conceito de ciclo de operação reduzido com tempos de atividade (*listen*) e tempo de repouso (*sleep*), evitando o desperdício de energia com recebimento de pacotes destinados a outros nodos (*overhearing*) e com a escuta do meio sem tráfego (*idle listening*).

A principal característica deste protocolo é utilização de um algoritmo distribuído para balanceamento de carga na rede. Este algoritmo estabelece que os nodos com baixa energia devem ser usados com menor frequência no roteamento, para isso realiza um procedimento local de eleição. A eleição escolhe o nodo com a carga mais baixa de energia entre todos os nodos da rede, destinando um tempo maior de repouso ao mesmo.

Como o protocolo é baseado em TDMA, a eleição é totalmente integrada com o tempo alocado para cada nodo (*slot* - janela de tempo), o protocolo não diminui a vazão da rede. O conceito de balanceamento de carga serve para obter um tempo de vida uniforme para todos os nodos da RSSF.

### 3.5 Protocolo TRAMA

O protocolo TRAMA (*Traffic Adaptive Multiple Access*) [37] é baseado em alocação estática de canal TDMA. É projetado para RSSFs com aplicações dirigidas a eventos, com coleta contínua ou periódica de dados da rede. O protocolo provê a eficiência de energia através de um método de acesso ao canal que tenta evitar colisões em comunicações *unicast*, *broadcast* ou *multicast*.

O TRAMA é adaptativo ao tipo de tráfego e emprega um algoritmo distribuído de eleição. Este algoritmo determina qual nodo pode transmitir em determinado intervalo de tempo (*time-slot*), e não faz reserva para os nodos sem tráfego de envio.

O algoritmo de eleição é baseado em informações de tráfego de cada nodo, selecionando o receptor baseado em escalas que são anunciadas pelos transmissores. As escalas são obtidas pela troca de informações locais entre os nodos vizinhos. Então, a escala é transmitida especificando quais nodos serão os respectivos receptores do tráfego em ordem cronológica. O TRAMA alterna entre acessos aleatórios e escalonados, isto permite acomodar mudanças de topologia, adição de nodos à rede e tolerância a falhas. A troca de informações entre vizinhos tenta criar uma visão global da rede.

O TRAMA mostra-se adequado para aplicações insensíveis à latência que necessitam de eficiência em energia com alta taxa de entrega.

O protocolo consiste de três componentes:

- NP (*Neighbor Protocol*): responsável pela propagação e atualização de informações sobre seus vizinhos de um *hop*. As atualizações são incrementais e permitem determinar o conjunto de vizinhos que serão adicionados ou removidos.
- SEP (*Schedule Exchange Protocol*): permite que os nodos troquem informações e escalas da vizinhança de dois *hops*.
- AEA (*Adaptive Election Algorithm*): utiliza as informações da vizinhança e de suas escalas para selecionar transmissores e receptores para o intervalo de tempo atual, enquanto os outros nodos selecionam o modo de repouso (*sleep*).

Os protocolos analisados atendem a requisitos específicos das aplicações para as quais foram projetados. Todos utilizam as características específicas de suas aplicação para conseguir



alguma vantagem, como, por exemplo o equilíbrio de carga na rede. A Tabela 2, aponta algumas características destes protocolos, assim como algumas vantagens e desvantagens.

Tabela 2: Protocolos MAC para RSSF.

Protocolo	Características	Tipo de RSSF	Vantagem	Desvantagem
S-MAC	ciclo de operação reduzido	dirigidas a eventos, com coleta periódica, insensível a latência da rede e com baixa taxa de envio de mensagens	prolonga consideravelmente o tempo de vida da rede	aumento da latência devido ao período <i>sleep</i>
T-MAC	ciclo de operação reduzido com incremento de um temporizador	dirigidas a eventos, com coleta periódica, insensível a latência da rede e com baixa taxa de envio de mensagens	consegue um consumo mais eficiente que o S-MAC, devido ao incremento do temporizador	é extremamente limitado com relação a largura de banda da rede, problemas com a sincronização dos nodos
ARC	provê <i>fairness</i> adaptando a taxa de transmissão do tráfego, evita colisões adicionando um atraso aleatório antes do tempo de escuta	coleta periódica de dados, sensível a latência com baixa ou alta taxa de envio de mensagens, RSSF com grande número de nodos	provê uma distribuição justa e mantém uma razoável largura de banda	não é eficiente para redes com um tráfego alto
DE-MAC	ciclo de operação reduzido, utiliza um algoritmo distribuído para balanceamento da carga na rede	coleta periódica, insensível a latência da rede e com baixa taxa de envio de mensagens	o balanceamento de carga serve para obter um tempo de vida uniforme para todos os nodos da RSSF	o algoritmo de balanceamento pode comprometer no caso de uma rede dirigida a eventos
TRAMA	utiliza um algoritmo de eleição distribuído para evitar colisões em comunicações <i>unicast</i> , <i>broadcast</i> ou <i>multicast</i> , alterna acessos entre aleatórios e escalonados	insensíveis à latência, com alta taxa de entrega	acomoda mudanças de topologia da rede, adição de nodos a rede e tolerância a falhas	devido ao seu algoritmo de eleição há uma grande troca de informações na rede, desfavorecendo a economia de energia.

Com base nas características e funcionalidades dos protocolos investigados, e a partir do conhecimento do conhecimento adquirido com o estudo de RSSF e técnicas de acesso ao meio, foi possível identificar pontos interessantes para definição de um novo protocolo MAC para este tipo de rede. O protocolo proposto, denominado R-MAC, é descrito a seguir.



## 4 Protocolo Proposto R-MAC

Neste capítulo é descrito o protocolo de acesso ao meio proposto aqui denominado R-MAC. Este protocolo foi concebido para trabalhar com dispositivos nodos sensores em RSSFs com grande números de nodos, dirigidas a eventos e com coleta periódica de dados. A motivação deste protocolo é garantir o consumo eficiente de energia em redes de sensores agregando funções de auto-gerência, como auto-organização, auto-configuração e auto-manutenção.

### 4.1 Visão Geral do R-MAC

O Protocolo R-MAC é composto por funções que facilitam a comunicação entre os nodos vizinhos tentando diminuir as colisões, para melhoria na eficiência energética e funções de auto-manutenção.

Para acessar o meio o R-MAC utiliza o método de acesso *Contention-Based por Distributed Coordination Function (DCF)*, baseado no protocolo *IEEE802.11*. O DCF permite que várias estações independentes acessem o meio sem um controle centralizado [31]. No método de acesso DCF antes de transmitir a estação deve verificar o meio. Se estiver livre, a transmissão pode começar. Um esquema de janela de contenção é utilizado para prevenir que varias estações utilizem o meio ao mesmo tempo, evitando sucessivas colisões. Além disso, um esquema de *interframes* permite selecionar níveis de prioridades entre mensagens, facilitando o acesso ao meio de mensagens com alta prioridade.

Uma estratégia utilizada no R-MAC foi reduzir o tamanho do campos de controle dos pacotes em comparação a outros protocolos, o que diminui o tráfego na rede reduzindo o consumo de energia, pois o tempo de processamento dos pacotes é menor.

A operação da rede entre períodos *listen/sleep* garante um consumo eficiente e evita o desperdício (ver seção 2.4.6). Para aplicações de RSSF com maior tráfego na rede é possível configurar o *duty cycle* para um período maior, evitando congestionamento na rede.

Um vetor de alocação do meio, que é enviado junto com as mensagens, permite que nodos vizinhos que não participam de uma comunicação entrem no modo *sleep* até o final da comunicação. O R-MAC trata a detecção de erros na recepção de mensagens, possibilitando a requisição de um novo envio ou simplesmente ignorando a mensagens.

O protocolo proposto permite determinar o limite inferior de energia requerida para auto-

organização da rede. Ao inicializar a rede o R-MAC cria uma lista de vizinhos que permite montar uma pré-estrutura da rede, que pode ser utilizada pelas camadas superiores para definir o caminho das rotas de entrega de mensagem.

O ajuste da energia de transmissão permite que um nodo que esteja fora da rede tente a conexão aumentando sua energia de transmissão. Além disso, é possível procurar um maior número de nodos vizinhos, aumentando o número de conexões do nodo. Ainda, no caso de perda de conectividade é possível tentar uma nova rota de conexão. Estas quedas ocorrem por uma mudança de topologia, como, por exemplo, pela queda do nodo que é o elo de ligação de uma subrede de nodos com a RSSF, conhecido como *sink node*.

Algumas das funcionalidades do R-MAC são controladas diretamente na camada Física. Para enviar uma mensagem é necessário que o módulo de RF do nodo sensor esteja configurado na posição TX. Da mesma maneira para o nodo receber uma mensagem o módulo de RF deve estar na posição RX. O controle de envio e recebimento de mensagens é realizado pela camada MAC. A seção 4.2 explica como o protocolo R-MAC interage com a camada Física para controlar o módulo RF.

## 4.2 Interação do R-MAC com o módulo RF

Os parâmetros de controle de um nodo sensor como, por exemplo, modo transmitindo ou recebendo, modo ativo/inativo, ajuste da energia de transmissão são controlados pela camadas MAC, sendo configurados diretamente pelo módulo RF durante a operação do nodo. Esta configuração ocorre alterando registradores de configuração conforme os parâmetros de controle requisitados. A comunicação com o módulo RF é feita através de comandos enviados pelo módulo de processamento em uma interface serial, como a *Serial Peripheral Interface*(SPI) esse esquema de comunicação pode ser observado na Figura 3 no capítulo 2.

Alguns módulos de RF utilizados em nodos sensores, como o CC1100 [7], permitem que as configurações dos parâmetros de rádio sejam feitas quando o módulo é inicializado ou também em tempo de execução. Normalmente, estas configurações são realizadas através do controle de uma máquina de estados. A seção 4.2.1 apresenta o modelo de uma máquina de estados que será utilizada como estudo de caso do protocolo R-MAC. Trata-se da máquina de estados do dispositivo CC1100 da Texas Instruments, que é um dispositivo de multi-freqüência, de baixo consumo e que permite configurar 9 níveis de ajuste para energia de transmissão.

Normalmente, máquinas de estados que controlam o módulo RF são relativamente complexas, necessitando de inúmeros passos para trocar o estado do módulo. Uma simples operação como, colocar o nodo no estado *sleep*, implica no recebimento de uma série de comandos para que a máquina transite por vários estados até atingir o estado desejado. As funcionalidades de um protocolo MAC, por muitas vezes, exige que o módulo de RF passe várias vezes pelos estados TX, RX, IDLE, SLEEP e por outros estados de configuração. Assim, para diminuir a complexidade destas ações foi elaborada uma máquina de estados da camada MAC com um

número reduzido de estados, que interage com a máquina da camada Física para realizar estas ações. Esta máquina será apresentada na seção 4.2.2.

#### 4.2.1 Máquina de estados da camada Física.

As ações do módulo RF CC1100 são controladas pela máquina de estados apresentada na Figura 19. As transações entre os estados são feitas por comandos recebidos pelo módulo de controle enviados pelo módulo de processamento através de uma interface SPI. O protocolo R-MAC se encarrega de controlar estes comandos, desta maneira é possível alterar as configurações deste módulo, transmitir e receber mensagens e colocar o módulo em *listen* ou em *sleep*.

Os parâmetros de configuração do dispositivo CC1100 são salvos em um registrador de configuração de 8 bits. Quando o dispositivo é inicializado este consulta o registrador para carregar os parâmetros de configuração. Quaisquer mudanças na configuração alteram o registrador.

A máquina do CC1100 é formada por 16 estados:

- IDLE - É o estado inicial da máquina. Quando o nodo não está transmitindo ou recebendo o IDLE é o estado padrão. Normalmente, é o estado de partida para transitar para outros estados de configuração. Seu consumo é baixo se comparado ao estado de transmissão, ficando em torno de  $1,6mA$ , porém toda vez que o nodo estiver em ação este deve ficar no estado SLEEP.
- MANCAL - É um estado de configuração. É atingido a partir de IDLE e serve para calibrar a frequência de envio ou de recebimento. A corrente típica de consumo é em torno de  $8,2mA$ .
- SLEEP - É o estado de menor consumo de módulo, sua corrente típica de operação é de  $0,4\mu A$ , que é suficiente para manter a configuração do nodo. Para atingir SLEEP é necessário estar em IDLE, sendo que a transição pode ser de duas maneiras. Com o comando direto para apagar o nodo, SPWD (*power down mode*), ou, com o comando WOR (*Wake-on-radio*). O WOR pode ser configurado diretamente no dispositivo para periodicamente transitar entre o estado IDLE e SLEEP, sem a interferência do domínio de processamento.
- XOFF - Neste estado o oscilador é desligado e o valor de todos os registradores são guardados. A corrente típica neste estado é de  $0,16mA$
- FS\_WAKEUP CALIBRATE SETTling FSTXON - São estados de configuração necessários para configurar o módulo antes deste chegar no estado TX ou RX. Os tempos de transição de cada um destes estados são fixos (estes tempos são utilizados na máquina da camada MAC para transitar entre os estados 4.2.2). A corrente típica é de  $8,2mA$ , a mesma para todos estados.

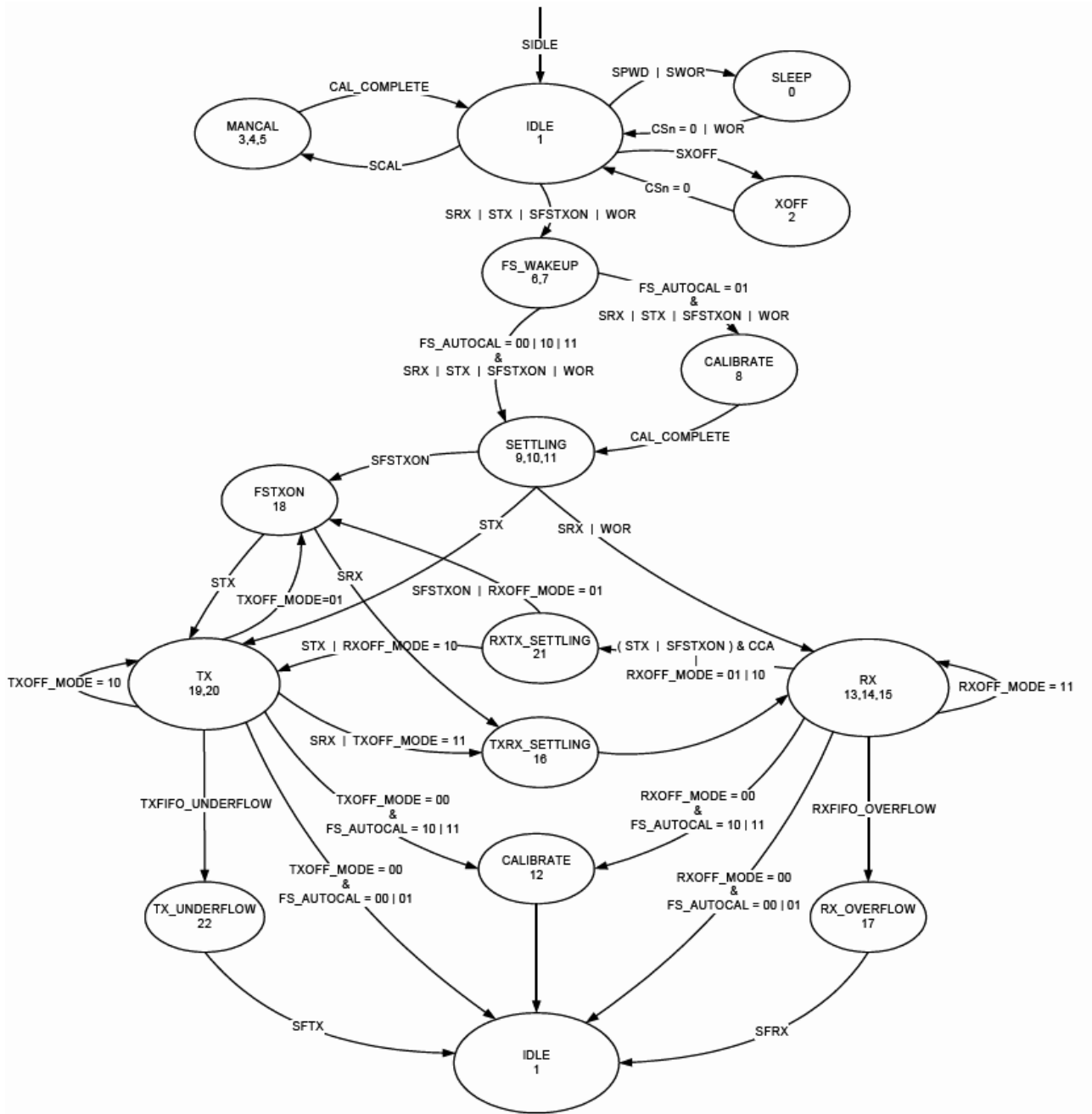


Figura 19: Máquina de estados do módulo de RF CC1100.

- RXTX\_SETTING TXRX\_SETTING - são estados de configuração necessários para mudar a configuração do módulo de RX para TX e de TX para RX.
- TX\_UNDERFLOW - mecanismo FIFO de envio com 64bytes.
- RX\_OVERFLOW - mecanismo FIFO de recebimento com 64bytes.
- TX RX - São os estados ativos do módulo. Quando o módulo está em TX começa o envio da mensagem. Quando o módulo está em RX começa a procura por uma nova mensagem. O consumo do estado RX varia de acordo com a taxa de recebimento, com a utilização

ou não do recurso de sensibilidade limite (Figura 21) e com a frequência utilizada. Porém esta variação é pequena e a corrente típica de RX fica em torno de  $15mA$ . A corrente máxima do estado TX é de  $32,3mA$ . A corrente consumida pelo módulo RF em cada configuração dos estados TX e RX pode ser vista na Tabela 3 do Capítulo 2.

Os módulos de RF estudados para esta dissertação podem ser modelados em máquinas de estados, similar ao CC1100. Sendo que a principal diferença entre estes módulos fica nos comandos de controle, nas possíveis configurações e no tempo de transição entre os estados.

#### 4.2.2 Modelagem da máquina de estados da camada de Enlace

O objetivo de modelar a camada de Enlace em uma máquina de estados é simplificar a atuação do protocolo proposto no controle da camada Física. Algumas funcionalidades do R-MAC exigem que o módulo RF passe por vários estados até atingir o estado desejado, para que estas transições sejam feitas é necessário que uma série de comandos sejam enviados para o módulo RF. A utilização de uma máquina para camada de Enlace permite que a transição entre dois estados represente uma seqüência de comandos na camada Física. Com isso, as regras do protocolo são aplicadas na máquina de estados para gerenciar a comunicação e as transições entre cada estado são controladas diretamente pela camada de Enlace, pelo início ou fim de uma mensagem e por contadores.

A máquina de estados da camada de Enlace é apresentada na Figura 20. Esta máquina contém nove estados, alguns estados têm relação direta com a máquina da camada Física, outros representam um conjunto de estados e o estado de configuração representa alguns estados e os registradores de configuração. A seguir, cada estado será descrito detalhadamente.

O Estado 1 é o estado inicial, chamado de estado *Idle*. Este estado corresponde ao estado de mesmo nome na camada física. É um estado de transição, onde o módulo não envia e nem recebe mensagens, mas o módulo deve passar por este estado para iniciar o recebimento ou envio de uma mensagem. O módulo também deve estar em *Idle* de forma a entrar no estado de configuração ou no estado *sleep*.

O Estado 2 *sleep* corresponde ao estado de mesmo nome na camada física. O protocolo implementa a funcionalidade *Listen/Sleep* que obriga o módulo a operar entre períodos ativos e inativos. Quando o nodo está no período inativo ele é levado para o estado *sleep* retornando a suas atividades depois de decorrido o tempo do contador *sleep* ou pela ocorrência de um evento.

O Estado 3 é chamado de estado de configuração e é selecionado para alteração de algum parâmetro de configuração. Este estado condensa todas as rotinas de configuração, calibração e os registradores de configuração do módulo RF. Ele é atingido por meio de uma requisição da camada de Enlace de configuração RF. Cada parâmetro a ser alterado tem seu determinado tempo para realizar as configurações. No final da configuração as alterações são salvas no registrador de configuração e a máquina retorna ao estado *idle*.

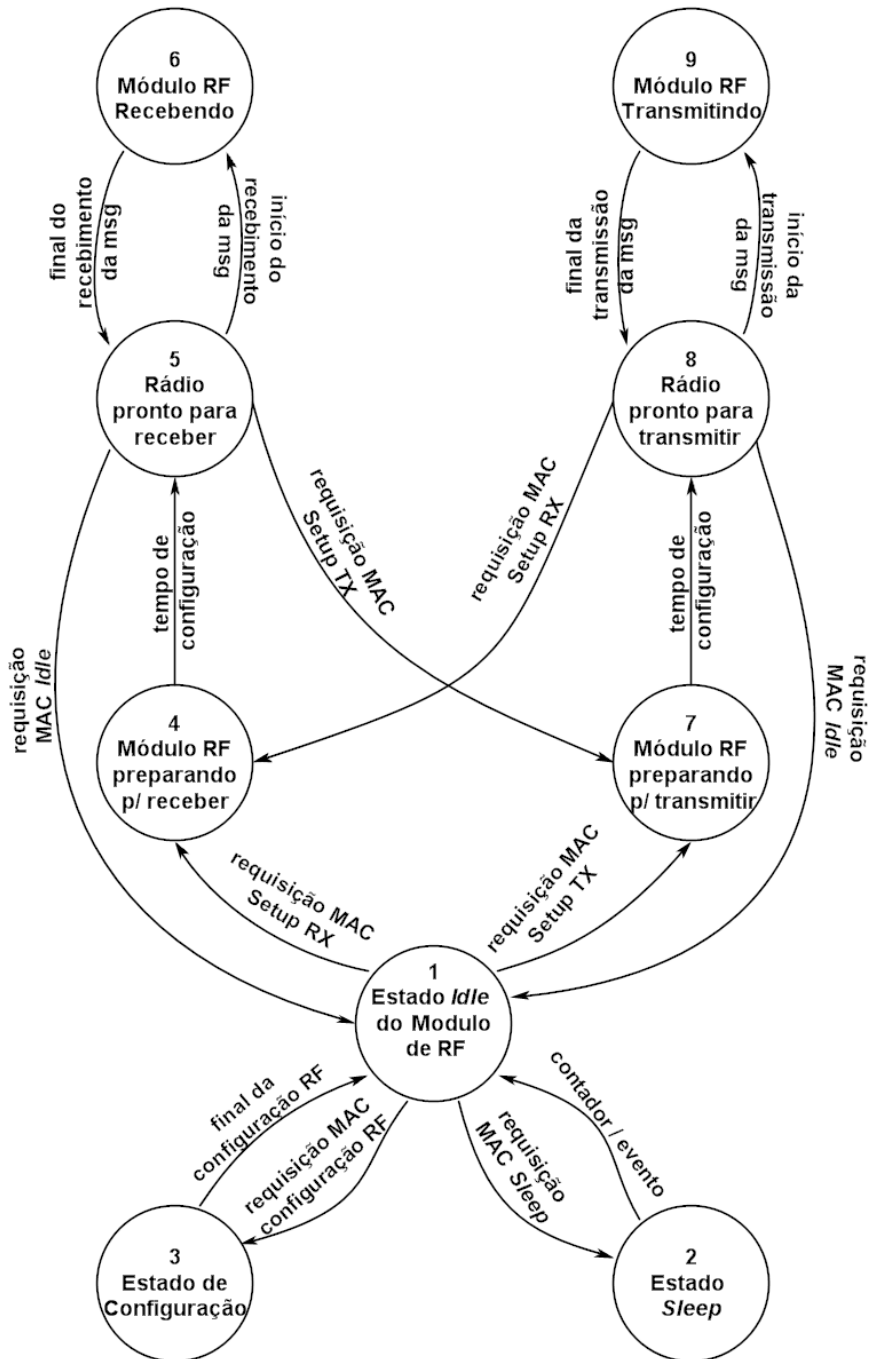


Figura 20: Atuação do protocolo MAC na máquina de estados do Módulo RF de um nó sensor



Os Estados 7,8,9 correspondem a transmissão de uma mensagem e são inteiramente controlados pela camada de Enlace. Serão descritos detalhadamente na seção 4.2.2.1.

Os Estados 4,5,6 são os estados correspondentes ao recebimento de mensagem. O controle destes estados é mais complexo que os estados de recebimento, pois além do controle do protocolo MAC dependem também de eventos do meio. O controle destes estados e tratado detalhadamente na seção 4.2.2.2.

#### 4.2.2.1 Envio de mensagem

O envio de uma mensagem é completamente controlado pela camada de Enlace. Na Figura 20 iniciando do estado 1 *idle* é requisitado o *setup TX*, transitando a máquina para o estado 7 onde o módulo RF é preparado para transmitir. Após o tempo de configuração o módulo entra no estado 8 e está pronto para enviar a mensagem. Este tempo de configuração é necessário para preparar o módulo de RF para enviar a mensagem. No estado 8, é aguardada uma requisição para início de transmissão para colocar a mensagem em uma fila que será enviada para o estado 9. No estado 9 a mensagem é efetivamente transmitida. Depois de concluída a transmissão da mensagem a máquina retorna ao estado 8, neste ponto a máquina pode transitar para o estado 1 *idle* ou para o estado 4 de acordo com a requisição do protocolo R-MAC.

#### 4.2.2.2 Recebimento de mensagem

O estado 4 da máquina da Figura 20 representa o início da configuração do módulo para a recepção de uma mensagem. A recepção de uma mensagem não depende somente da camada de Enlace, pois no momento do recebimento da mensagem ela pode não estar no início ou pode estar corrompida. O módulo deve escutar o meio para perceber quando existe uma mensagem e em que ponto está. Além disso colisões podem ocorrer, ou, até mesmo a perda do sinal por um alto nível de ruído antes da conclusão da mensagem. Por este motivo o recebimento de mensagem não depende exclusivamente da camada de Enlace.

A Figura 21 apresenta o gráfico de energia de transmissão pelo tempo. A linha limite de sensibilidade de recebimento do dispositivo é a capacidade que o dispositivo tem de capturar o sinal da mensagem no meio. A diferença entre o ruído do meio e o sinal da mensagem é chamada de *Signal to Noise Ratio* (SNR) [32].

Supondo que um nodo transmita uma mensagem para todos seus nodos vizinhos, devido a distância que cada nodo se encontra do emissor e aos ruídos do ambiente, cada nodo irá receber a mensagem em um tempo e com níveis de potência diferentes.

Para que o dispositivo consiga interpretar a mensagem corretamente é necessário que o valor de SNR não seja menor que a sensibilidade que o dispositivo é capaz de detectar. O gráfico da Figura 21 mostra que o sinal da primeira mensagem está baixo e o nível de ruído do ambiente está alto. Por conseqüência o SNR é menor que a tolerância mínima do dispositivo para receber uma mensagem, logo a mensagem será descartada. Em seguida é enviada outra mensagem, o

nível de potência da mensagem está mais alto e o ruído ambiente está menor, por isso o SNR esta dentro dos índices aceitáveis do dispositivo e para que a mensagem seja aceita.

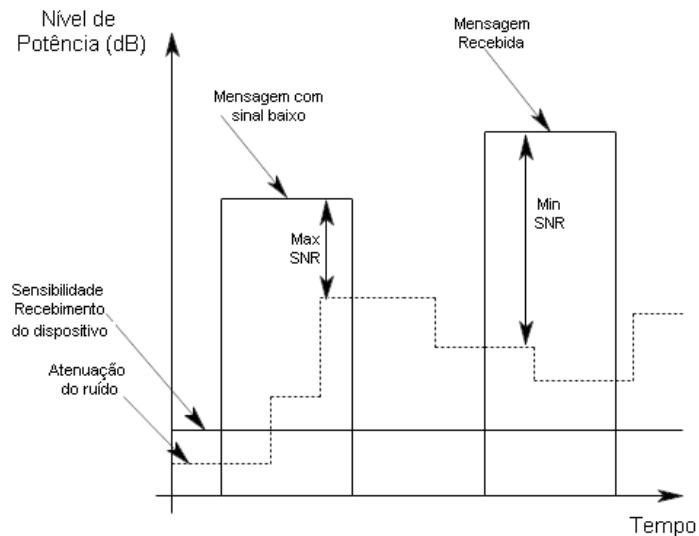


Figura 21: Ruído acumulado SNR

A máquina de estados da Figura 22 mostra a expansão dos estados 5 e 6 da máquina da camada de Enlace (Figura 20), apresentando todas as possíveis transições para o recebimento de uma mensagem. Quando a máquina está no estado 5 esta escuta o meio a espera de um preâmbulo de início de mensagem. Caso a máquina comece a identificar um preâmbulo, mas o dispositivo percebe que o SNR está acima do tolerado, a mensagem é descartada e a máquina continua no estado 5, o que é representado pela transição *h*. Ainda no estado 5 qualquer fim de mensagem recebido será considerado como ruído (Estado *g*). Quando um preâmbulo de início de mensagem é identificado e o SNR está dentro da tolerância, a mensagem é carregada passando para o estado 6, transição *a*. No estado 6, o módulo RF está recebendo. Quando um preâmbulo de início é identificado, podem ocorrer três transições: (*b*) Quando o preâmbulo da mensagem esta com o SNR acima do tolerado, este é considerado ruído e uma nova mensagem (preâmbulo) pode ser carregada. (*c*) Quando a mensagem carregada (e respectivo preâmbulo) está com SNR em níveis aceitáveis, esta mensagem é mantida e qualquer novo preâmbulo/mensagem é considerado ruído. (*e*) Quando a mensagem nova mensagem está como o SNR e a mensagem carregada está com o SNR acima da tolerância ambas são consideradas ruído e a máquina é passada novamente para o estado 5.

Ainda, transições de fim de mensagem são possíveis apenas no estado 6 e podem ocorrer de duas maneiras: (*d*) Quando a mensagem não está carregada é descartada como ruído. (*d*) Quando recebe fim de mensagem com mensagem carregada, é entregue uma chamada de recebimento de mensagem para a camada de Enlace com o objetivo de avisar a conclusão da mensagem.

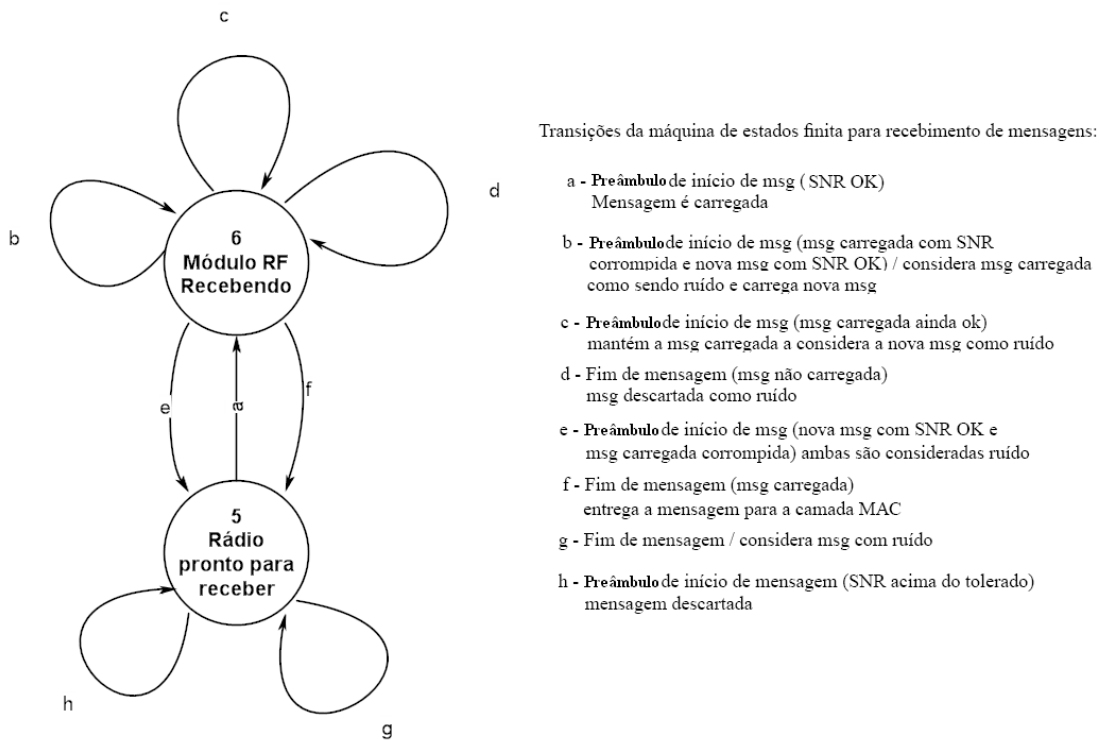


Figura 22: Máquina de estados finita para o recebimento de mensagem

### 4.3 Funcionalidades do R-MAC

As funcionalidades propostas no R-MAC podem ser divididas em três tipos: As funcionalidades de auto-gerência, que tem por objetivo a formação da pré-estrutura da rede e a manutenção das conexão dos nodos na rede; as funcionalidades de comunicação que permitem que os nodos estabeleçam comunicação entre si tentando evitar as colisões; por último, as funcionalidades para economia de energia dos nodos rede.

#### 4.3.1 Funcionalidades de Auto-Gerência

Quando uma RSSF é ativa pela primeira vez, várias tarefas devem ser realizadas para estabilizar e montar a infra-estrutura necessária que usualmente um trabalho colaborativo utiliza. No protocolo R-MAC a primeira tarefa de um nodo ao iniciar é procurar por seus vizinhos. Esta função é chamada de SN (*Search Neighbor*), ela possibilita a criação de uma pré-estrutura de rede criando uma lista de possíveis vizinhos.

A lista de possíveis vizinhos contém os nodos vizinhos que estão no alcance de rádio do nodo e que podem trocar mensagens com ele. Esta lista criada pelo R-MAC pode ser utilizada pelos protocolos de roteamento para definir as possíveis rotas de entrega de mensagens. O protocolo de roteamento estabelece quais os nodos serão utilizados nas rotas de entregas de mensagens, como já foi visto na seção 2.2.3.

Outra funcionalidade de auto-gerência do R-MAC é o Ajuste da Energia de Transmissão. Esta funcionalidade permite aumentar a energia de transmissão de cada nodo, com isso é possível que o nodo se conecte a rede, procure um número maior de vizinhos ou se reconecte em caso de queda de algum vizinho, estas funcionalidades serão detalhadas na seção 4.3.1.2.

#### 4.3.1.1 Função SN (*Search Neighbor*)

A função SN é utilizada pelos nodos para criar uma lista de possíveis vizinhos. Quando um nodo inicializa, uma mensagem de SN é enviada para todos os nodos que recebem esta mensagem verificam sua lista de vizinhos e se possível, adicionam o emissor enviando uma mensagem de confirmação (ADD). A lista de vizinhos é armazenada em uma área da memória do dispositivo reservada para protocolo. Seus parâmetros de configuração são o número máximo e mínimo de vizinhos, o que permite definir o número de conexões em cada nodo. Caso a lista esteja cheia o nodo emissor não será adicionado.

O fluxograma da Figura 23 descreve o algoritmo para a formação da lista de vizinhos. Quando a função é inicializada o nodo verifica se algum vizinho está enviando a mensagem SN, caso negativo deve enviar a mensagem SN e passa a escutar o meio a espera da mensagem de confirmação dos nodos vizinhos. Recebendo uma mensagem SN ou ADD o nodo deve enviar uma mensagem ADD para o nodo emissor. Logo após enviar ADD, o nodo verifica se atingiu o número mínimo de vizinhos, caso positivo o nodo sai da função SN.

No caso do nodo enviar SN inúmeras vezes e não receber resposta, este passa para o módulo de reconfiguração da energia de transmissão (detalhes na seção 4.3.1.2). Após aumentar o nível da energia de transmissão o nodo deve enviar a mensagem RET e esperar o recebimento da mensagem RETr. Caso não receba, deverá repetir o procedimento até receber. Quando o nodo recebe RET o emissor é adicionado na lista de vizinhos e envia a mensagem ADD.

A Figura 24 mostra uma pequena parte da rede iniciando sua formação, definida pelos seguintes passos:

1. Os nodos são inicializados e entram em seu modo de configuração, aguardando o recebimento de uma mensagem SN.
2. O nodo B consegue entrar no meio e envia uma mensagem SN em *broadscat* para todos os nodos que estão em seu alcance de rádio. O nodo J que não está no alcance de rádio de B também envia a mensagem SN. No mesmo momento o nodo A também envia sua mensagem de SN, mas não tem nenhum nodo em seu alcance de rádio.
3. Os nodos F e C, recebem a mensagem SN enviada por B. Com isso, o nodo B é adicionado na lista de vizinhos de C e F. No mesmo passo o nodo D recebe a mensagem de J e o adiciona em sua lista de vizinhos.
4. Os nodos F, C e D disputam o meio para enviar a mensagem de ADD, confirmando o adionamento de seus vizinhos. O nodo F é o primeiro a ocupar o meio e envia a mensagem

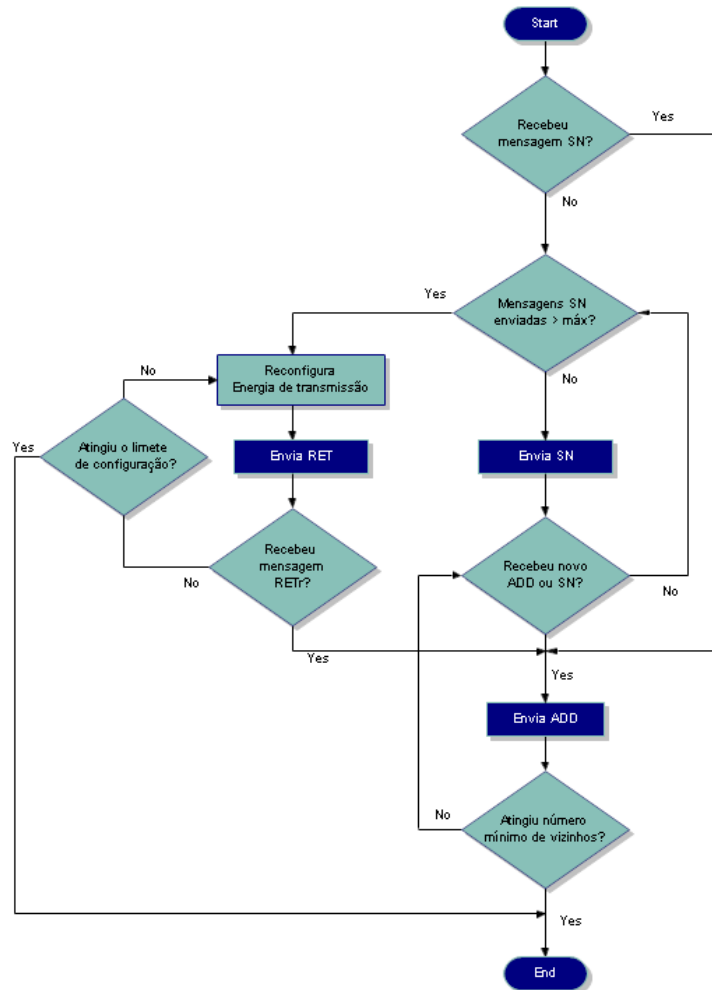


Figura 23: Máquina de estados da configuração inicial da RSSF.

de ADD. Os outros nodos que estão no alcance de rádio de F escutam a mensagem ADD e adicionam F na sua lista de vizinhos.

5. O nodo D ganha a disputa de acesso ao meio e envia a mensagem ADD informando que adicionou os nodos J e F.
6. O próximo a ocupar o meio é o nodo C, que envia a mensagem ADD informando que adicionou B, F e D. Ao receber esta mensagem os nodos B, F e D adicionam o nodo C em sua lista de vizinhos.
7. O nodo D disputa novamente o meio para informar o adcionamento de C, enviando uma mensagem ADD[J,F,C].
8. Em seguida o nodo F faz o mesmo, enviando uma mensagem ADD[B,J,D,C].
9. O nodo B também confirma o adcionamento de F e C, enviando a mensagem ADD[F,C].
10. Logo após o nodo J confirma o adcionamento de seus vizinhos, enviando a mensagem ADD[F,D].

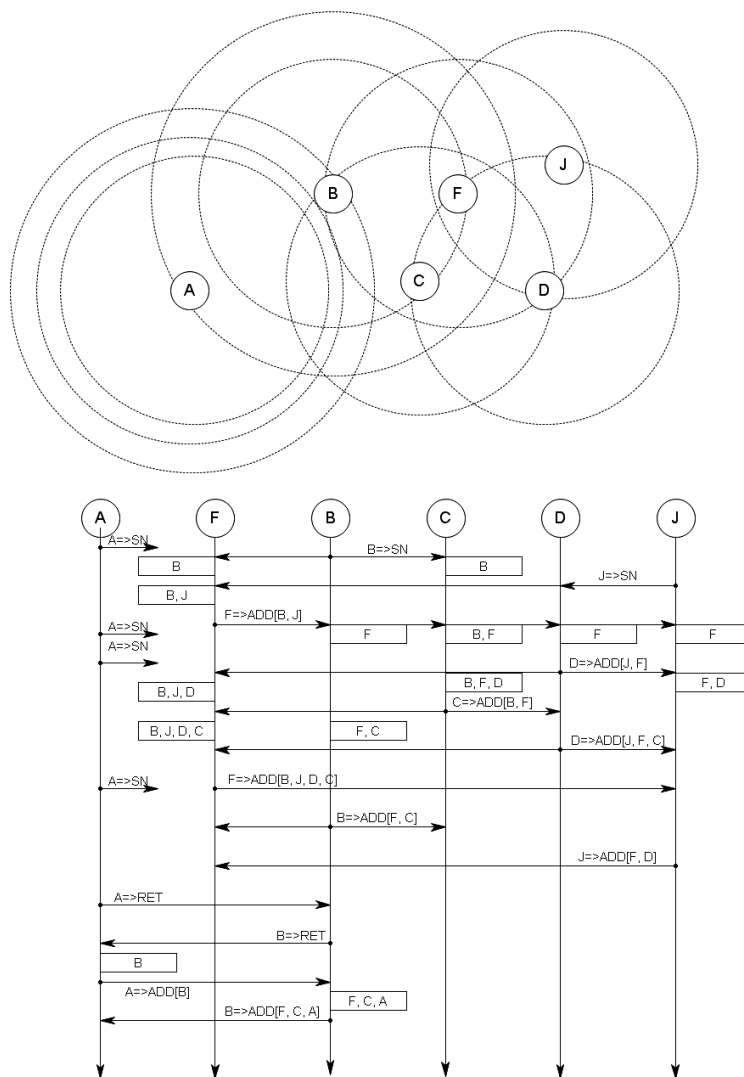


Figura 24: Auto-configuração do ID dos nodos da RSSF.

11. O nodo A que até o momento não recebeu resposta de sua mensagem SN entra em seu estado de reconfiguração da Energia de Transmissão, envia a mensagem RET e fica a espera de uma resposta.
12. O nodo B escuta a mensagem de reconfiguração enviada por A. Então utiliza a função para ajuste de sua energia de transmissão de acordo com os parâmetros enviados por A, e envia uma mensagem RETr destinada ao nodo A.
13. O nodo A recebe a mensagem RETr de B adicionando-o na lista de vizinhos e envia uma mensagem de confirmação ADD[B].
14. O nodo B recebe a mensagem de A adiciona na lista de vizinhos e envia uma mensagem ADD[F,C,A].

Criar uma lista de vizinhos auxilia o trabalho dos protocolos de roteamento, que podem

consultar as informações da lista para definir rotas e também para sincronização entre nodos. Quando utilizamos o período reduzido de operação na rede *Listen/Sleep* os nodos, frequentemente, podem estar fora de sincronia, exigindo um período de sincronização para realizar a comunicação.

Para facilitar a sincronização entre os nodos podemos utilizar a lista de vizinhos, criando uma subrede configuranda com o mesmo tempo de operação para os nodos participantes. Essa estratégia permite que a energia que seria gasta na sincronização dos participantes dessa subrede seja economizada.

#### 4.3.1.2 Energia de Transmissão - RET

A energia de transmissão é um importante parâmetro de configuração de um nodo sensor. Este parâmetro define o raio de alcance do módulo RF, em outras palavras, define a distância que o nodo consegue estabelecer comunicação com outros nodos. Em aplicações de RSSFs a energia de transmissão é configurada observando-se a quantidade de nodos, a área e a fidelidade da medida a ser coletada pela rede.

A energia de transmissão tem uma relação direta com a potência, e por sua vez com o consumo de energia. A equação 4.1 apresenta esta relação, sendo a energia de transmissão expressa em dBm e a potência em *mW* [32].

$$dBm = \log 10(mW) * 10 \quad (4.1)$$

Utilizando a equação como base, podemos afirmar que a grandeza da energia de transmissão é diretamente proporcional a potência utilizada. Então, quanto menor a energia de transmissão menor a potência utilizada, isto é, menor o consumo de energia do nodo.

Tabela 3: Energia de Transmissão / Frequência / Potência típica de Consumo (\*tensão base 3V)

Dispositivo	Energia de transmissão (dBm)	315MHz	433MHz	868MHz	915MHz
		Potência típica (mW)	Potência típica (mW)	Potência típica (mW)	Potência típica (mW)
CC1100	-30	31,8	34,5	35,7	35,4
	-20	33,3	36,0	37,2	36,9
	-15	35,4	38,1	39,0	39,0
	-10	39,0	42,0	43,5	42,0
	-5	38,7	41,1	42,3	41,7
	0	44,4	46,5	50,7	50,1
	5	54,3	47,0	60,0	47,9
	7	66,3	72,6	77,4	77,4
	10	81,0	86,7	92,1	96,9
TRF6903	-12	87,0	84,0	81,0	81,0
	-2	90,0	87,	84,0	84,0
	8	111,0	108,0	105,0	105,0
nRF9E5	-14	-	30,0	30,0	30,0
	-6	-	39,0	39,0	39,0
	2	-	57,0	57,0	57,0
	10	-	84,0	84,0	84,0

A Tabela 3 confirma esta relação, onde são apresentados três dispositivos pesquisados que

são utilizados como módulos RF de nodos sensores. A Tabela mostra uma relação entre Potência típica de consumo em  $mW$ , frequência em  $MHz$  e energia de transmissão em  $dBm$ . Podemos observar que a diferença de consumo pode atingir níveis quase três vezes maiores dependendo do parâmetro de energia de transmissão utilizado na configuração do nodo. Sendo assim, encontrar o menor valor para o parâmetro de energia de transmissão de maneira que o nodo consiga estabelecer as conexões, reduz o consumo de energia no nodo. A função de reconfiguração da energia de transmissão permite que o nodo obtenha o melhor valor para este parâmetro de maneira automática. Ainda, permite que nodos que perderam a conexão tentem uma nova conexão na rede, aumentando o raio de alcance da comunicação.

A Figura 25 apresenta o esquema de funcionamento da função RET em forma de fluxograma. Quando um nodo atinge o número limite de mensagens enviadas sem respostas este entra na função RET. Neste momento, os seguintes passos devem ser observados: reconfigurar o parâmetro de energia de transmissão para um nível mais alto, salvar a configuração nos registradores do dispositivo e enviar a mensagem RET. Então, o nodo deve aguardar o envio de um RETr do vizinho mais próximo, caso isso não ocorra a operação é repetida até o nível máximo da configuração da energia de transmissão do nodo.

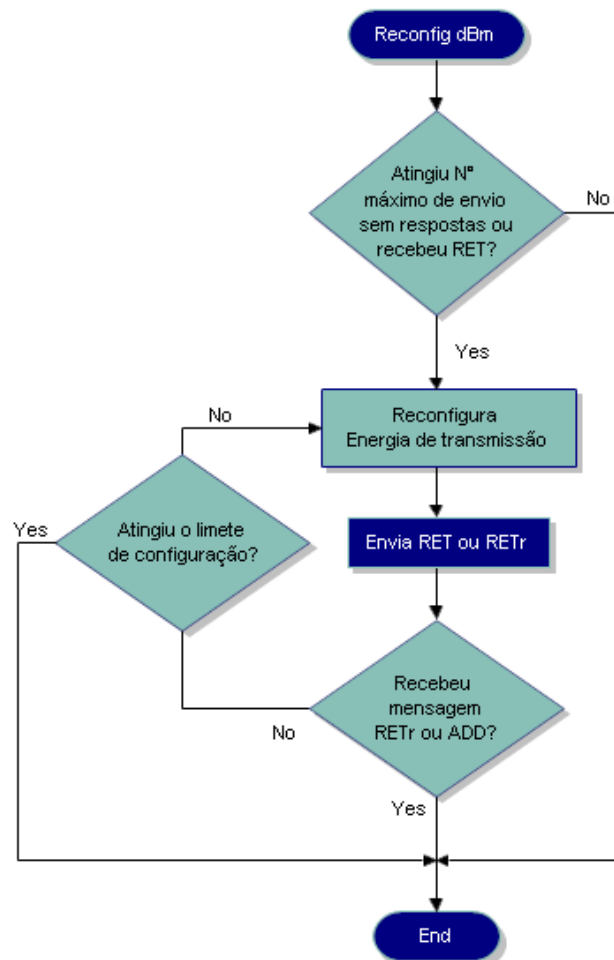


Figura 25: Fluxograma da função RET.



A Figura 26 mostra um exemplo de funcionamento da função RET. O nodo B que era o caminho de A para C sai da rede por falta de energia. O nodo A identifica a queda de B e entra em RET. A primeira tentativa de comunicação depois de aumentar seu nível de energia de transmissão não obtém sucesso. Então o nodo A aumenta novamente seu nível de energia de transmissão e envia RET novamente. O nodo recebe a mensagem de A, reconfigura seu nível de energia de transmissão e envia a mensagem RETr.

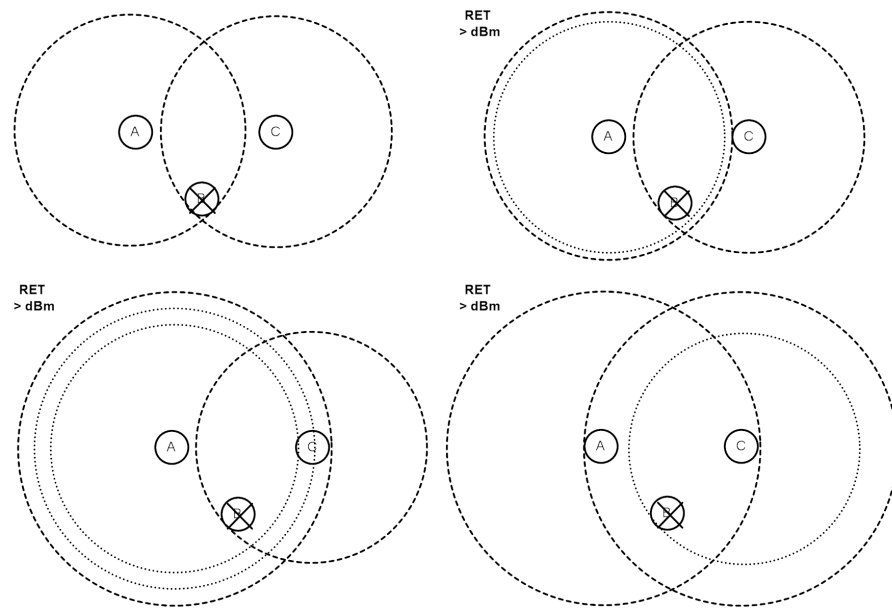


Figura 26: Funcionamento da RET.

Caso o nodo atinja o maior nível da sua energia de transmissão e mesmo assim não consiga uma comunicação com outro nodo, ele sai da função RET. Esta situação pode ser vista na Figura 27 e ocorre quando o alcance do dispositivo não é suficiente para criar uma nova conexão, como no caso do nodo F, ou, quando o nodo é um nodo terminal da rede, como por exemplo o nodo G.

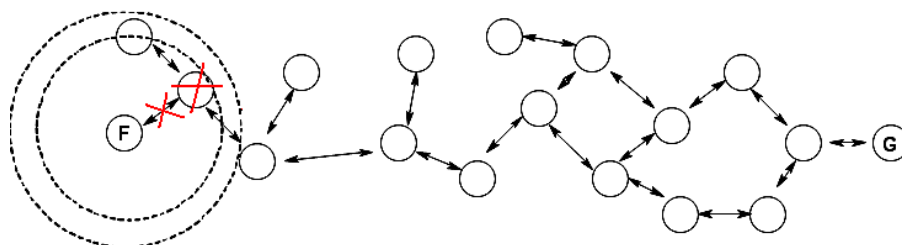


Figura 27: Situação em que o nodo retorna a configuração inicial

A Figura 28 mostra uma situação em que um nodo sai da rede e por conta disso os nodos A, F, D, M, K ficam isolados em uma subrede. Neste caso o R-MAC identifica esta subrede e descobre novas conexões.

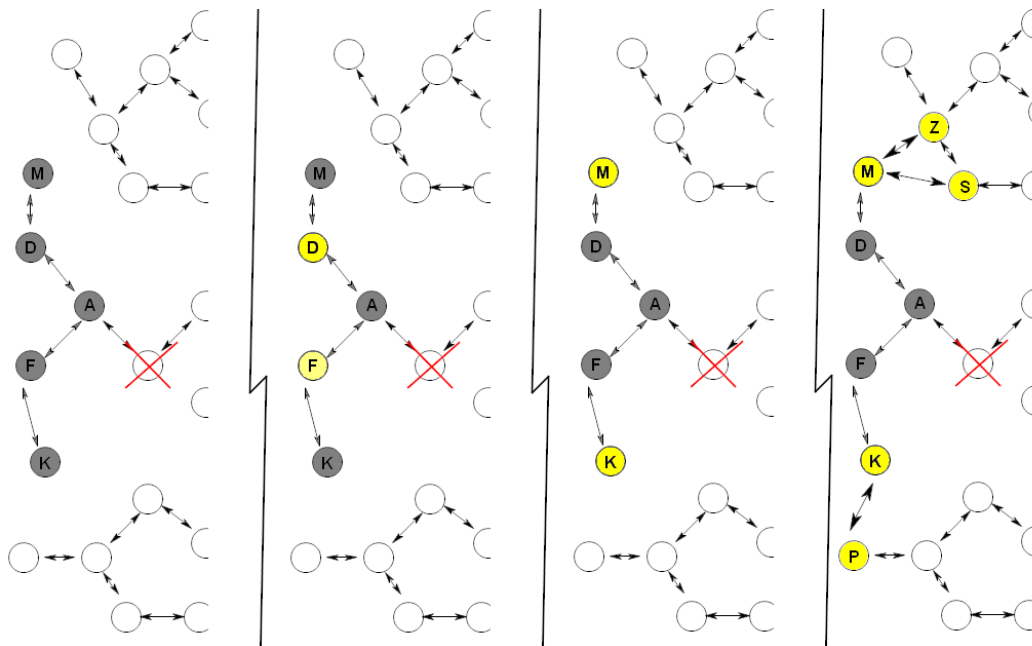


Figura 28: Nova conexão para nodos isolados na RSSF.

Como o nodo B saiu da rede, o nodo A estabelece comunicação apenas com os dois nodos que ainda estão em seu alcance de rádio F e D. O nodo F tem em seu alcance de rádio o nodo A e o nodo K, como o nodo A foi o nodo de origem da mensagem apenas o nodo K vai enviá-la. O nodo K por sua vez envia a mensagem porém recebe apenas a resposta do nodo F. Então ele executa a função de reconfiguração de dBm para tentar conseguir uma nova conexão. Depois da função executada é criado um novo caminho até o nodo P. A mensagem irá percorrer a rede até chegar no AP.

Esta funcionalidade é importante para o auto-gerenciamento da RSSF, uma vez que, possibilita que subredes isoladas estabeleçam novas conexões sendo possível recuperar dados que seriam perdidos. Isso ocorre, pois, mesmo que os nodos realizem as leituras caso estes dados não fossem gravados na memória dos nodos, os dados não chegariam ao AP e seriam perdidos.

## 4.3.2 Funcionalidades de comunicação

### 4.3.2.1 *Interframes Space*

O uso de *interframes* em um protocolo permite criar níveis de prioridade para diferentes tipos de mensagens. O *interframe* funciona como um tempo de guarda que a estação deve esperar antes de enviar a mensagem. A lógica é simples, mensagens com alta prioridade podem acessar o meio mais rapidamente do que mensagens com prioridade mais baixa. O R-MAC utiliza dois tipos de *interframe* o *Short Interframe Space (SIFS)* e o *DCF Interframe Space (DIFS)*.

A Figura 29 mostra um gráfico de acesso ao meio com os dois tipos de *interframes* do protocolo, o SFIS e o DIFS. Assim que o meio estiver desocupado os nodos selecionam um

tipo de *interframe* de acordo com a mensagem a ser enviada. No R-MAC o SFIS é utilizado somente por mensagens de alta prioridade, que são os *frames* de controle CTS, DATA e ACK, RETr ou EVT. As mensagens de baixa prioridade CTS, RET e ADD devem utilizar o DIFS e aguardar seu *slot* na janela de contenção para iniciar a transmissão.

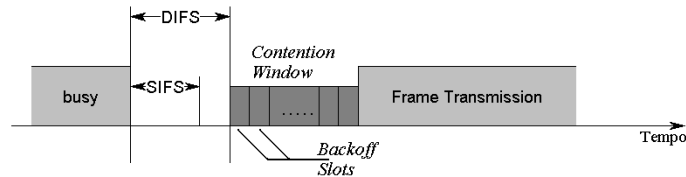


Figura 29: *InterFrame Spacing*.

O cálculo para o tempo de cada *interframe* no R-MAC é feito de acordo com as equações a seguir:

Para o SIFS:

$$SIFS = \frac{10}{PSF} \quad (4.2)$$

Para o DIFS:

$$DIFS = SIFS + 2 \times slot \quad (4.3)$$

onde, *slot* corresponde ao tamanho de um slot da janela de contenção, que na configuração padrão do R-MAC é de *2bits*.

#### 4.3.2.2 Janela de contenção

A janela de contenção é utilizada para evitar colisões quando várias estações desejam acessar o meio ao mesmo tempo. Mensagens de alta prioridade não utilizam janela de contenção depois do *interframe* SFIS. Isso porque, um nodo só envia uma mensagem de alta prioridade se já estiver utilizando o meio e portanto poderá acessar o meio antes dos outros nodos.

Mensagens com baixa prioridade podem acessar o meio depois de um *interframe* DIFS. Estando o meio livre é possível que mais de um nodo deseje transmitir ao mesmo tempo, o que acarretaria em uma colisão. Para evitar este problema é utilizado um algoritmo *backoff* aleatório, baseado no protocolo IEEE 802.11 [31].

Este algoritmo funciona da seguinte maneira, as mensagens que devem esperar pelo fim de um *interframe* DIFS entram em um período denominado janela de contenção (*contention-window*) assim que o DIFS termina, Figura 29. Este período tem um tamanho fixo e é dividido em *slots*. O nodo sorteia um *slot* aleatório que serve como referência para entrar no meio. Quando diversos nodos desejam transmitir, o nodo que receber o primeiro *slot* (o *slot* com o menor número) pode enviar primeiro, enquanto as outras devem esperar o meio ficar desocupado novamente.

No protocolo IEEE 802.11 o tamanho da *contention-window* é incrementado em potência de 2 menos 1 (ou seja,  $x^2 - 1$ . ex:31,63,127,255) a medida que retransmissões vão sendo

necessárias. Esta medida diminui a probabilidade de colisões, pois quanto maior o número de *slots* menor a probabilidade de duas estações ocuparem o mesmo *slot*.

$$slot = \frac{64}{PSF} \quad (4.4)$$

No R-MAC o tamanho da janela é fixo de 0 a 31, e quando ocorre uma colisão os nodos devem sortear um novo *slot*.

#### 4.3.2.3 Detecção de Erros

A detecção de erros no R-MAC é feita somente em mensagens *unicast*, ficando a correção de erros sob responsabilidade do nodo que transmitiu. Quando um erro é detectado a estação emissora deve reenviar o *frame*. Em alguns casos, a estação pode não receber o pacote de controle ACK, então irá supor que a transmissão foi perdida. Quando a estação retransmitir o pacote o contador *retry counter* é incrementado. A utilização deste contador permite que seja estipulado um limite para retransmitir um *frame*. O *retry counter* inicia em zero e é incrementado a cada falha de transmissão. Quando este alcança um limite determinado o *frame* é descartado. O *retry counter* é zerado quando:

- Um *frame CTS* é recebido em resposta a um RTS.
- Um *frame broadcast* ou *multicast* é recebido.
- A camada de Enlace reconhece que o pacote deve ser descartado

## 4.4 Funcionalidades de economia de energia

### 4.4.1 Função *Listen/Sleep*

A Função *Listen/Sleep*, conforme visto no capítulo 2, na seção 2.4.6, é implementada para obter economia de energia aproveitando-se da característica do baixo tráfego de mensagens em RSSF, alternando o módulo de RF entre ciclo de operação ativo (*listen*) e inativo *sleep*.

O protocolo R-MAC permite configurar o tamanho do ciclo de operação e o tempo em que o dispositivo deve ficar no modo *Listen(Duty Cycle)*. Esta ação é feita alterando valores do registrador de configuração do módulo RF. Vale lembrar, quando o nodo está no modo *sleep* fica impossibilitado de enviar ou receber mensagens. Contudo, quanto maior o tempo de operação em *sleep* maior a economia de energia. Para uma mesma RSSF, normalmente os nodos são configurados com o mesmo *duty cycle*, mas é possível configurar cada nodo com um *duty cycle* diferente.

#### 4.4.2 Função Nodos ouvintes - CNO

Um problema encontrado nos protocolos tradicionais são os nodos que não participam da comunicação, mas permanecem ativos durante as transmissões de mensagem que não são destinadas a eles. Este problema é conhecido como nodos ouvintes. A função Controlador de Nodos Ouvintes (CNO) é inspirada na função *Carrier-Sensing* virtual do 802.11, e sua principal funcionalidade é levar os nodos para estado *sleep* quando as mensagens recebidas não são destinadas a estes.

As mensagens de controle do R-MAC contém o campo CNO com o tempo que o meio está reservado para a comunicação. Quando um nodo recebe uma mensagem que não é destinada a ele deve verificar o CNO e entrar em modo *sleep* até o contador ser decrementado.

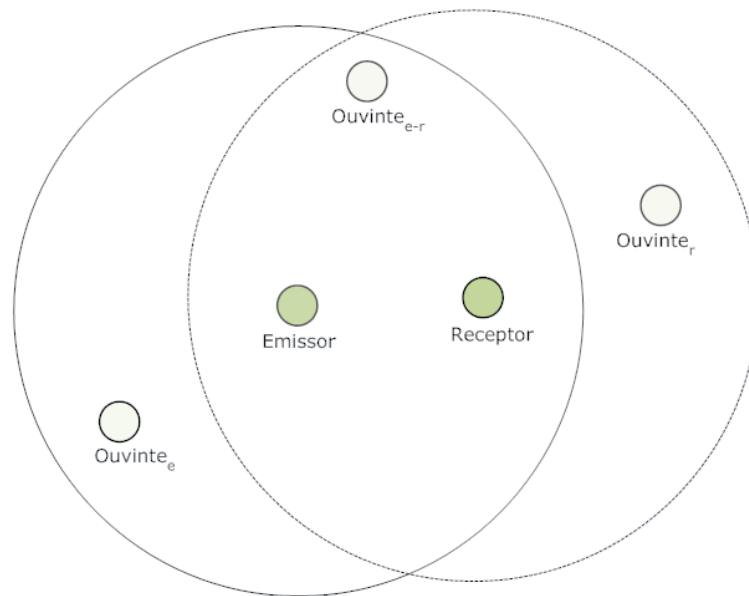


Figura 30: Função Contador de Nodos Ouvintes.

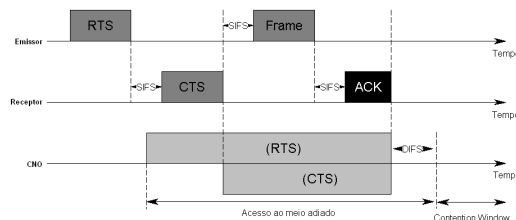


Figura 31: Usando o CNO.

A Figura 31 mostra como o CNO protege as operações de uma eventual interrupção. Quando o nodo envia um pacote de controle RTS para estabelecer comunicação, dentre outras informações contidas no pacote está o tempo CNO. Então, todas as estações que recebem pacote RTS sabem que o meio estará ocupado durante o tempo determinado pelo CNO. Quando a estação receptora recebe o pacote RTS, prontamente ela envia um pacote CTS que contém o CNO(CTS)

mais curto, ver Figura 30. Isso garante que todos os nodos que estão no alcance do sinal de rádio dos dois nodos recebam o CNO, reservando o meio para a comunicação até que o CNO termine. Depois que o nodo receptor recebe o pacote, a seqüência é completada com o envio do pacote ACK e o fim do CNO.

## 4.5 Formato dos Pacotes

Esta seção apresenta os pacotes utilizados pelo protocolo R-MAC, seus componentes, suas estruturas e suas funcionalidades. Os pacotes do protocolo R-MAC foram projetados com base no protocolo IEEE802.11, levando em consideração as necessidades específicas e os módulos de RF utilizados por nodos sensores.

A Figura 32 apresenta o pacote básico com os campos presentes em todos os pacotes do protocolo. Todos os pacotes são estruturados pelos quatro campos preâmbulo, delimiter, R-MAC header e CRC. Cada campo do protocolo tem uma funcionalidade. A estrutura dos campos é mantida para todos os pacotes, mas o conteúdo de cada campo é diferente para os vários tipos de pacotes. A seguir será detalhado cada campo básico do protocolo, seus subcampos e os tipos de pacotes.

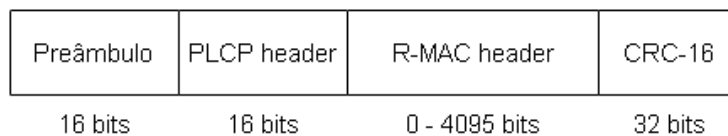


Figura 32: Os quatro campos básicos do pacote.

### 4.5.1 Preâmbulo

O campo preâmbulo é utilizado para identificar o início de uma mensagem e começar a sincronização da mensagem no receptor. Este é composto de dois subcampos o *Sync* e o *Start Frame Delimiter* (SFD).

#### Sync

O Campo Sync é composto por 8 bits alternados entre 1 e 0 (10101010). Este serve para identificar o início de uma mensagem e para sincronizar o recebimento da mensagem.

O Sync do protocolo proposto tem 8 bits, sendo dez vezes menor que o existente no protocolo IEEE802.11. Esta diferença existe porque o IEEE802.11 prevê comunicação entre dispositivos com os módulos RF diferentes e dar suporte a dispositivos com múltiplas antenas.

Os 80bits no campo sync permite que dispositivos diferentes tenham mais tempo para acertar a sincronização de recebimento do pacote. Ainda, o campo Sync longo permite que um

dispositivo com múltiplas antenas identifique o sinal de Sync e escolha a antena com o melhor sinal [19].

Contudo, em dispositivos nodos sensores o tamanho deste campo não necessita ser tão grande. Os nodos sensores, normalmente, trabalham com apenas uma antena. Também, os dispositivos a eles conectados são previamente conhecidos o que facilita a sincronização. Por este motivo é possível reduzir o tamanho deste campo, reduzindo também o tamanho da mensagem e por conseqüência o consumo de energia.

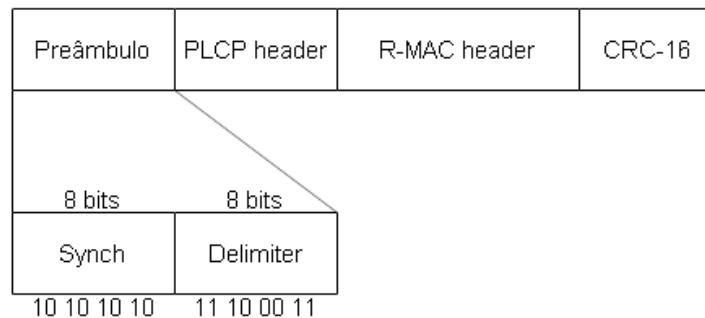


Figura 33: Campo preâmbulo.

#### ***Start Frame Delimiter (SFD)***

Este subcampo do preâmbulo sinaliza o final do campo sync e início do pacote. A seqüência de 8 bits 1 e 0 alternados do campo sync é interrompida com uma seqüência de três bits 1 e três bits 0 seguidos. O campo é finalizado com dois bits 1 (11 10 00 11) que sinalizam o início do *delimiter*. Esta seqüência de 0's e 1's seguidos facilita a identificação de início do pacote, sendo bastante utilizados em dispositivos menores, como RFID [?].

#### **4.5.2 *Delimiter***

O Campo *Delimiter* do protocolo R-MAC utiliza o mesmo padrão do *IEEE802.11*, pois este campo contém informações lógicas que serão utilizadas pela camada física para decodificar o pacote. Esta padronização permite que nodos da RSSF decodifiquem pacotes de outros dispositivos, como APs, sem nenhuma alteração.

O *Delimiter* é composto por 16 bits e é dividido em dois subcampos, o *PCLP Length Word* (PLW) e o *PCLP Signaling Field* (PSF). Através destas informações é possível saber o tamanho e a taxa que o pacote será recebido. Com estas informações é possível calcular o tempo de recebimento de cada pacote.

#### ***PCLP Length Word (PLW)***

O PLW é o primeiro campo do *delimiter*, este campo contém o número de bytes do pacote a partir do *delimiter*. É formado por 12 bits, permitindo pacotes com tamanhos entre 0 e

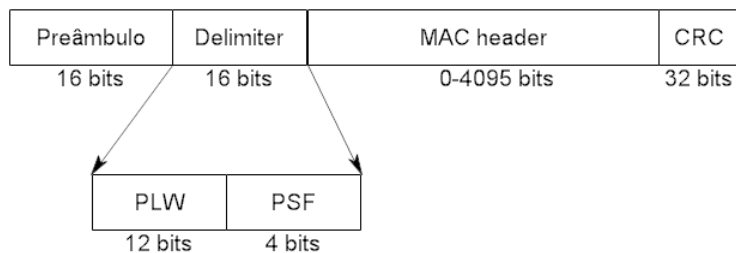


Figura 34: Campo *delimiter*.

Tabela 4: PSF-Taxa de recebimento do pacote.

Bits(1-2-3)	Taxa de Recebimento
000	4,8Kbps
001	9,6Kbps
010	19,2Kbps
011	38,4Kbps
100	56,6Kbps
101	250Kbps
110	500Kbps
111	1Mbps

4095bytes. Este campo serve como parâmetro para identificar o final do pacote, o nodo recebe o pacote, decodifica este campo e sabe quantos bits seguem a partir do *delimiter*.

### PCLP Signaling Field (PSF)

O campo PSF define a que taxa o pacote será transmitido. No R-MAC são utilizadas taxas de transmissão bem mais baixas que no IEEE802.11 que vai de 1Mbps a 4,5Mbps. Normalmente nodos sensores não atingem estas taxas.

São utilizados 4 bits para determinar a taxa de transmissão. Estes valores são pre-estabelecidos de acordo com as taxas de transmissão encontradas nos dispositivos nodo sensores. O primeiro bit é reservado e colocado em 0. Os bits de 1 – 3 permitem selecionar a taxa de recebimento do pacote do acordo com a Tabela 4.

### 4.5.3 Cyclic Redundancy Check(CRC)

O pacote encerra com o campo CRC, que é utilizado para verificar a integridade do pacote recebido. O CRC é um método para a verificação de dados que utiliza uma operação matemática polinomial para esta verificação. Os dados passam por um polinômio e o valor de saída é o CRC. Quando o pacote é enviado a estação que envia realiza o cálculo do CRC e o insere no final do pacote. Todos os campos do MAC header são inclusos no cálculo do CRC. A estação que recebe o pacote repete o mesmo cálculo e o compara com o CRC recebido. Se o valor do CRC for igual ao CRC recebido o pacote não está danificado. Alguns módulos de RF implementam o cálculo do CRC, passando a informação se o pacote é valido ou não. Caso o módulo não tenha esta funcionalidade esta tarefa fica a cargo do protocolo da camada de Enlace.



#### 4.5.4 MAC header

Esta seção apresenta a estrutura do campo MAC header. Este campo é o principal na estrutura do protocolo, apresentado na Figura 35. Neste estão informações fundamentais para o funcionamento do protocolo como, por exemplo, tipos de mensagens, campo de dados, controle de seqüência e endereços. As subseções a seguir detalham todas as funcionalidades do campo MAC header.

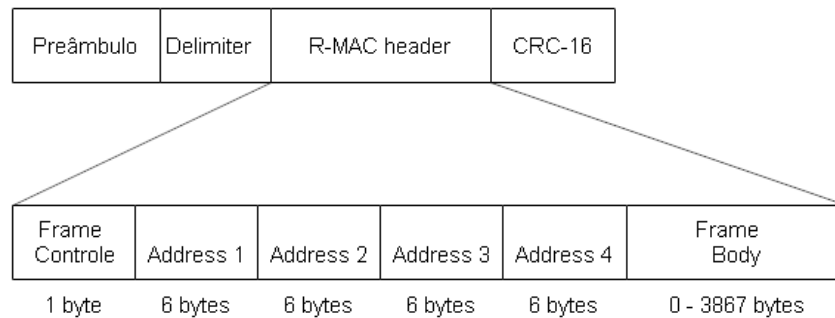


Figura 35: Campo R-MAC header.

##### 4.5.4.1 Campo Address

O endereçamento no R-MAC segue o padrão utilizado em outras redes *IEEE802*, com endereços de *48bits*. Esta padronização permite a utilização de outros protocolos tradicionais nas camadas superiores sem grandes modificações, permitindo o tráfego de mensagens *unicasts*, *multicasts* e *broadcasts*. Podendo ser utilizados até quatro campos de 6 bytes para endereçamento. A utilização de cada endereço depende dos subcampos *To* e *From*, seção 4.5.4.3. Em regra geral o *Address 1* é utilizado para o nó receptor, o *Address 2* para o transmissor e o *Address 3* pode ser utilizado para filtrar mensagens.

##### 4.5.4.2 Campo Frame Body

O campo *Frame Body* contém informações de dados ou pacotes de gerenciamento. Este campo transmite os pacotes das camadas superiores e pode ter o tamanho variável de 0 a 3863bytes.

##### 4.5.4.3 Campo Frame Control

O subcampo *frame control* têm 16 bits de controle, guardando informações necessárias para interpretar toda a seqüência do campo MAC header. Seus subcampos são: *Type*, *CNO*, *To DS*, *From DS*, *more fragments*, *retry* (veja Figura 36).

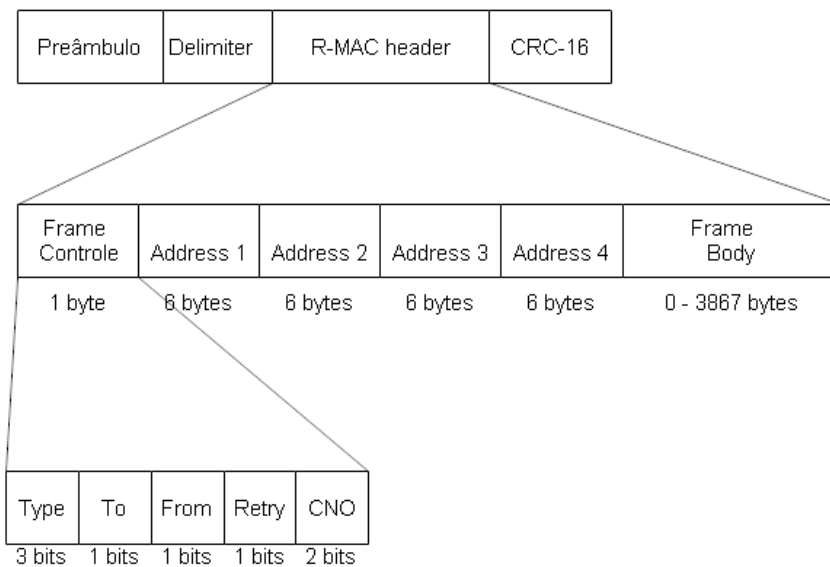


Figura 36: Máquina de estados do Módulo de RF de um nodo sensor

Tabela 5: Tipo de pacotes do R-MAC.

Bits(1-2-3)	Tipo de pacotes
000	EVT
001	RTS
010	CTS
011	ACK
100	DATA
101	RET
110	RETr
111	ADD

### Subcampo Type

O subcampo Type identifica o tipo e a função de *frame*, isto é, se é um pacote de controle, dados ou configuração. A Tabela 5 mostra a lista completa dos tipos de pacotes. O EVT, único pacote não apresentado anteriormente, é utilizado para sinalizar a ocorrência de um evento.

### Subcampos To e From

Os subcampos To e From tem 1bit cada e são utilizados apenas em pacotes de dados, indicando para quem o pacote é destinado. Este dois subcampos definem o significado de cada campo *Address*, podem definir se o pacote está sendo enviado de um nodo para outro, de um nodo para o AP ou do AP para os nodos. A Tabela 6 mostra as quatro combinações possíveis para os dois subcampos.

- *Destination Address*(DA): é o endereço de 48bit IEEE MAC que corresponde ao receptor final, a estação que deve receber o pacote para as camadas superiores processarem.
- *Source Address* (SA): este endereço identifica a estação que transmitiu a mensagem.

Tabela 6: Funções do subcampo To e From.

Função	To	From	Address 1	Address 2	Address 3	Address 4
IBSS	0	0	RA=DA	SA	BISSD	-
Do AP	0	1	RA=DA	BSSID	SA	-
Para o AP	1	0	RA=BSSID	SA	DA	-
Para a rede	1	1	RA	TA	DA	SA

Tabela 7: Contador de Nodos Ouvintes.

Bits(1-2)	CNO
00	Não entra em <i>sleep</i>
01	Entra em <i>sleep</i> pelo período $T_{CNOCTS}$
10	Entra em <i>sleep</i> pelo período $T_{CNO RTS}$
11	Entra em <i>sleep</i> até o próximo estado <i>Listen</i>

- *Receiver Address* (RA): indica qual estação pode processar o pacote. Se o RA for um nodo, logo o RA é o DA. Para pacotes destinados a uma estação fora da RSSF o RA é o AP, e o DA será o endereço do roteador estação fora da RSSF.
- *Transmitter address* (TA): identifica qual nodo transmitiu o pacote. O TA é utilizado internamente na RSSF.
- *Basic Service Set ID* (BSSID): é utilizado para identificar o AP.

### Subcampo Retry

Este Subcampo é utilizado para indicar se o pacote foi enviado pela primeira vez ou foi retransmitido. Quando o pacote foi enviado pela primeira vez este subcampo é 0. Quando trata-se de uma retransmissão do pacote o subcampo é 1. Este subcampo é utilizado pelo protocolo R-MAC para filtrar pacotes repetidos.

### Contador de Nodos Ouvintes - CNO

A funcionalidade CNO, que já foi vista na seção 4.4.2, terá seu subcampo detalhado agora. O CNO é formado por 2bits que servem para alocar o meio em uma comunicação. O nodo só leva em consideração o CNO se o pacote não for destinado a ele. A Tabela 7 mostra as possíveis configurações deste subcampo.

Quando o CNO é 00 a estação que recebe continua escutando o meio. Esta configuração é utilizada para pacotes de controle do AP ou em comunicações de um passo sem confirmação.

Quando a estação recebe o CNO 11, deve entrar em seu modo de *sleep* até seu próximo período de *listen*. Este CNO é utilizado quando dois nodos pretendem trocar uma grande quantidade de informações e irão ocupar o meio por bastante tempo.

Quando o nodo recebe o CNO 10 significa que outros dois nodos irão começar uma comunicação e que o nodo deve entrar no modo *sleep* durante o período desta comunicação. Este tempo é obtido pela equação 4.5, sendo o tamanho máximo da comunicação dividido pela taxa de transferência de dados (PSF).

$$Tam_{Pc(RTS)} = (3xSIFS + CTS + DATA + ACK)$$

$$T_{CNO_{RTS}} = \frac{Tam_{Pc(RTS)}}{PSF} \quad (4.5)$$

Quando o nodo recebe o CNO 01, isso significa que a comunicação já está em andamento e o nodo deve ficar no modo *sleep* pelo tempo definido pela equação 4.6.

$$Tam_{Pc(CTS)} = Tam_{Pc(RTS)} - (CTS + SIFS)$$

$$T_{CNO_{CTS}} = \frac{Tam_{Pc(CTS)}}{PSF} \quad (4.6)$$

Este capítulo apresentou a interação do protocolo R-MAC com o módulo RF, suas funcionalidades e possíveis configurações. O capítulo 5 apresenta a análise de algumas funções do R-MAC específicas para redução do consumo de energia nas RSSF. Esta análise pretende confirmar esta característica do protocolo proposto.

## 5 Validação e Análise dos Resultados

Este protocolo foi desenvolvido para RSSF dirigidas a eventos, com coleta periódica, insensível a latência da rede e com baixa taxa de envio.

O cenário escolhido para simulação foi o de monitoramento de temperatura em uma RSSF de baixa densidade com 100 nodos. A taxa de envio de mensagem será de 38,4kbps, com frequência de 915MHz e -10dBm de energia de transmissão. Os nodos da RSSF estão equipados com um sensor de temperatura, sendo prevista uma coleta periódica de dados a cada 1 s ou quando ocorrer um evento de temperatura superior a 45<sup>0</sup>C. As ações de coleta e análise dos dados ficam a cargo do sistema operacional do nodo sensor. Estas ações são executadas nos domínios de sensoriamento e de processamento, enquanto que o domínio de comunicação é ativado por uma requisição de envio de mensagem entregue a camada de Enlace.

Desta maneira, o domínio de sensoriamento é o responsável pela transformação da energia térmica em dados. Enquanto, o domínio de processamento gerencia o domínio de sensoriamento e avisa a camada de Enlace que o módulo de RF deve enviar uma mensagem com os valores do sensor toda vez que os dados foram lidos. Logo, o domínio de comunicação que é gerenciado pela camada de Enlace fica responsável por distribuir os dados coletados até o AP.

A seguir são apresentados e discutidos as estratégias utilizadas para testar e validar o protocolo proposto.

### 5.1 Modelo de Energia

O modelo de energia leva em consideração quatro estados do nodo *Sleep*, Ocioso, TX e RX. O estado *Sleep* é o estado de menor consumo, pois o nodo é posto em *standby* e não pode transmitir nem receber. O TX é o estado de transmissão, que é o de maior consumo. O estado Ocioso é o estado em que o nodo está escutando o meio a espera de uma mensagem, mas não está processando nenhuma mensagem. O estado RX é o estado em que o nodo está recebendo uma mensagem. Considerando os quatro estados possíveis pode-se obter a potência média consumida ( $P_m$ ) pela equação 5.1:

$$P_m = t_s \times P_s + t_{oc} \times P_{oc} + t_{TX} \times P_{TX} + t_{RX} \times P_{RX} \quad (5.1)$$

Tabela 8: Consumo em cada estado do nodo.

Estado	Potência (tensão base 3V)
Sleep	$1,2\mu W$
Ocioso	$138\mu W$
RX	$45,6mW$
TX	$92,1mW$

onde  $t_s, t_{oc}, t_{TX}$  e  $t_{RX}$  são as frações de tempo em que o nodo fica em cada estado. Estas frações devem satisfazer a condição  $t_s + t_{oc} + t_{TX} + t_{RX} = 1$ . Analogamente,  $P_s, P_{oc}, P_{TX}, P_{RX}$  são as potências instantâneas consumidas em cada um dos quatro estados.

O tempo de vida do nodo pode ser calculado com base na potência média  $P_m$  e a energia inicial  $E$ . A equação 5.2 representa o tempo necessário para a energia do nodo terminar:

$$T_v = \frac{E}{P_m} \quad (5.2)$$

A análise do tempo de vida apresentada neste trabalho leva em conta apenas a energia consumida pelo módulo RF, desconsiderando o consumo dos outros circuitos dos nodos.

Inicialmente, pode-se intuir que o menor tempo de vida possível de um nodo seria no caso em que ele passasse todo o tempo transmitindo. Contudo, mesmo que um nodo fique continuamente enviando dados, ele deve entrar em outros estados, como, Ocioso e RX pela característica de funcionamento do R-MAC.

Para verificarmos o consumo de energia no R-MAC a primeira análise realizada será em um cenário de comunicação direta entre dois nodos.

## 5.2 Comunicação Direta

Esta análise é importante pois serve como base para análises mais complexas. O cenário considerado consiste em analisar o consumo de dois nodos em uma comunicação direta. Os nodos estão separados por uma distância que permite a comunicação direta entre eles com a energia de transmissão configurada para de  $-10dBm$ . Os nodos utilizam uma taxa de  $38,4Kbps$ , garantindo que sempre que o meio esteja livre o nodo emissor tenha algo para transmitir. Desta forma, a taxa máxima de utilização pode ser calculada. A Tabela 8 mostra o consumo do nodo para cada estado neste cenário com base no *datasheet* do dispositivo [7] para as configurações acima.

Nesta condição,  $t_{oc}, t_{TX}$  e  $t_{RX}$  são a fração de tempo da transmissão em que o nodo passa pelo estado Ocioso, TX e RX. O estado *sleep* não é levado em consideração nesta análise, pois deseja-se obter o consumo durante a comunicação e para realizar qualquer comunicação o dispositivo deve estar em *Listen*.

Neste cenário, com a taxa de transferência de  $38,4Kbps$  utilizando as equações 4.2, 4.3 e 4.4 do capítulo 4, o tempo da janela de contenção e dos *interframes* são obtidos pelas equações a seguir, para cada *slot* da janela de contenção:

$$slot = \frac{20}{38,4Kbps} \cong 0,52ms \quad (5.3)$$

Para *interframes*:

Para o SIFS:

$$SIFS = \frac{10}{38,4Kbps} \cong 0,26ms \quad (5.4)$$

Para o DIFS:

$$DIFS = SIFS + 2 \times (slot) \cong 1,3ms \quad (5.5)$$

A Figura 37 mostra o tempo de transmissão de um quadro completo sem levar em consideração os atrasos de propagação. O tempo de cada *interframe* já foi calculado anteriormente. Também, temos o tempo de cada *slot* da janela de contenção, e sabemos que esta é composta de 31 *slots* que na média resulta em 15,5 *slots* de tempo, resultando em um tempo médio de 8,06ms por quadro. Além disso, o Preâmbulo e o *Delimiter* tem 2bytes cada um, resultando em 0,84ms para os dois. A transmissão do quadro de controle RTS, com 16bytes, leva 3,34ms, enquanto o CTS e ACK com 10bytes cada um leva 2,09ms. O tempo de transmissão do quadro de dados é igual ao tempo de transmissão a 38,4Kbps de 21bytes que corresponde ao R-MAC header mais os bytes de dados. É importante ressaltar que nos bytes de dados deve-se contar quaisquer cabeçalhos de camadas superiores que venham a ser utilizadas. O protocolo IP, por exemplo, possui 20bytes.

1,3 ms	8,06 ms	0,84 ms	3,34 ms	0,26 ms	0,84 ms	2,09 ms	0,26 ms	0,84 ms	$T_{dados}$	0,26 ms	0,84 ms	2,09 ms
DIFS	JC	Preâ/Delimiter	RTS	SIFS	Preâ/Delimiter	CTS	SIFS	Preâ/Delimiter	Dados	SIFS	Preâ/Delimiter	ACK
		4 bytes	16 bytes		4 bytes	10 bytes		4 bytes	(21 + Dados) bytes		4 bytes	10 bytes

Figura 37: Tempo de transmissão de uma Comunicação direta

Com estes valores pode-se obter o tempo de transmissão do quadro completo  $T_{qc}$ , pela equação 5.6:

$$T_{qc} = t_{JC} + 4 \times t_{pd} + 3 \times SIFS + DIFS + t_{RTS} + t_{CTS} + t_{Dados} + t_{ACK} \quad (5.6)$$

Assim o tempo de transmissão, em ms, é igual a:

$$T_{qc} = 12,96 + t_{Dados} \quad (5.7)$$

Para calcular os tempos de  $t_{oc}$ ,  $t_{TX}$  e  $t_{RX}$  atribui-se valores para o  $t_{Dados}$ . A Tabela 9 mostra os tempos e o número de quadros completos que podem ser transmitidos por ciclo de operação para diferentes tamanhos de pacotes de dados. Neste cenário, foi considerado o ciclo de operação da função *Listen/Sleep* com *duty cycle* de 10%, sendo que os dois nodos que participam da comunicação estão sincronizados. Com este *duty cycle* os nodos estão em *sleep* e operam 0,9 na fração de tempo  $t_s$  e 0,1 nas frações  $t_{oc}$ ,  $t_{RX}$  e  $t_{TX}$ .

Tabela 9: Valores de  $t_{Dados}$  para diferentes tamanhos de pacotes.

Tamanho do Pacote(Bytes)	$t_{Dados}$ em ms	Tempo quadro ms	Nº de quadro por ciclo de operação
48	10	31	3
64	13,4	34,4	2
160	33,4	54,3	2
250	52,1	73,09	1
500	104,17	125,16	umentar duty cycle para 13%
1000	208,34	229,33	umentar duty cycle para 25%

Tabela 10: Fração de tempo dos nodos pelo tamanho dos pacotes.

Tamanho do Pacote (bytes)	Emissor			Receptor		
	$t_{oc}$	$t_{TX}$	$t_{RX}$	$t_{oc}$	$t_{TX}$	$t_{RX}$
48	0,32	0,48	0,20	0,32	0,20	0,48
64	0,30	0,53	0,17	0,30	0,17	0,53
160	0,19	0,70	0,11	0,19	0,11	0,70
250	0,14	0,78	0,08	0,14	0,08	0,78
500	0,08	0,87	0,05	0,08	0,05	0,87
1000	0,04	0,93	0,03	0,04	0,03	0,93

A partir destes valores de  $t_{Dados}$  e tempos dos quadros pode-se calcular  $t_s$ ,  $t_{oc}$ ,  $t_{RX}$  e  $t_{TX}$  para o nodo emissor e para o nodo receptor. Os tempos de JC, SIFS, DFIS são os tempos em que ambos os nodos ficam ociosos. Os tempos referentes ao RTS e Dados, incluindo os preâmbulos e *delimiter* associados são tempos em que o emissor fica em TX e o receptor em RX. nos períodos relativos a CTS e ao ACK, a situação é inversa. Na Tabela 10 podem ser vistos os valores calculados.

De posse dos valores da Tabela 10, podemos calcular a potência média  $P_m$  da equação 5.1 para os nodos emissor e receptor. Lembrando que o *duty cycle* é de 10% na rede, isso torna  $t_s = 0,9$  e  $t_{oc} + t_{TX} + t_{RX} = 0,1$ . Utilizando um pacote de dados de 64bytes, o valor de  $P_m$  é dado por:

Para o nodo receptor:

$$P_m = 0,9 \times P_s + 0,03 \times P_{oc} + 0,017 \times P_{TX} + 0,053 \times P_{RX} \quad (5.8)$$

$$P_m = 0,9 \times 0,0012\mu W + 0,03 \times 0,138\mu W + 0,017 \times 92,1mW + 0,053 \times 45,6mW \quad (5.9)$$

$$P_m = 3,98mW \quad (5.10)$$

Para o emissor:

$$P_m = 0,9 \times P_s + 0,03 \times P_{oc} + 0,053 \times P_{TX} + 0,017 \times P_{RX} \quad (5.11)$$



$$P_m = 0,9 \times 0,0012mW + 0,03 \times 0,138mW + 0,053 \times 92,1mW + 0,017 \times 45,6mW \quad (5.12)$$

$$P_m = 5,66mW \quad (5.13)$$

Com o valor de  $P_m$  é possível estimar o tempo de vida  $T_v$  de um nodo emissor, sendo este o caso de maior consumo. A Tabela 1, no capítulo 1, apresenta o consumo nos três domínios processamento, comunicação e sensoriamento. Para estimar o valor de  $T_v$ , assumi-se que os domínios de processamento e sensoriamento estão sempre consumindo. Assumi-se também para domínio de processamento a potência de referencia  $P_{mP} = 0,75mW$ , para o domínio de sensoriamento a potência de  $P_{mS} = 1,8mW$ . Supondo que a energia do nodo  $E$  é uma bateria de  $2700mWh$ , utilizando a equação 5.2 obtém-se o  $T_v$  do nodo:

$$T_v = \frac{E}{P_mC + P_mP + P_mS} \quad (5.14)$$

$$T_v = \frac{2700mWh}{5,66mW + 0,75mW + 1,8mW} = 328,87h \approx 13,7dias \quad (5.15)$$

Para o receptor:

$$T_v = \frac{2700mWh}{3,98mW + 0,75mW + 1,8mW} = 413,47h \approx 17,23dias \quad (5.16)$$

O mesmo cálculo do tempo de vida  $T_v$  do nodo emissor pode ser realizado sem a utilização do ciclo de operação reduzido *Listen/Sleep*. Neste caso, o valor de  $t_s = 0$  e o *duty cycle* é 100%. O valor de  $P_m$  é obtido por:

$$P_m = 0,30 \times P_{oc} + 0,53 \times P_{TX} + 0,17 \times P_{RX} \quad (5.17)$$

$$P_m = 0,30 \times 0,138mW + 0,53 \times 92,1mW + 0,17 \times 45,6mW \quad (5.18)$$

$$P_m = 56,60mW \quad (5.19)$$

Para o nodo receptor:

$$P_m = 0,30 \times P_{oc} + 0,17 \times P_{TX} + 0,53 \times P_{RX} \quad (5.20)$$

$$P_m = 0,30 \times 0,138mW + 0,17 \times 92,1mW + 0,53 \times 45,6mW \quad (5.21)$$

$$P_m = 39,86mW \quad (5.22)$$

Substituindo o valor de  $P_m$  na equação 5.14 tem-se o valor de  $T_v$  sem *Listen/Sleep*:

$$T_v = \frac{2700mWh}{56,60mW + 0,75mW + 1,8mW} = 45,65h < 2dias \quad (5.23)$$

Para o receptor:

$$T_v = \frac{2700mWh}{39,86mW + 0,75mW + 1,8mW} = 63,66h < 3dias \quad (5.24)$$

A partir desses resultados, é possível observar que o ciclo de operação *Listen/Sleep* apresenta um ganho considerável no tempo de vida útil de uma RSSF. A configuração do tamanho do ciclo e do *duty cycle* deve ser regulado de acordo com as necessidades de coleta de dados e de acordo com o tamanho dos pacotes, tentando evitar a fragmentação.

### 5.3 Energia nos Nodos Ouvintes

Outro problema encontrado nos protocolos tradicionais é o problema dos nodos ouvintes. A função CNO do R-MAC diminui o consumo de energia dos nodos ouvintes. Existem três tipos de nodos ouvintes: o  $O_E$ , que está no alcance apenas do emissor e, portanto, só escuta o tráfego originado pelo emissor; o  $O_R$ , que está no alcance apenas do receptor e, portanto, só escuta o tráfego originado pelo receptor; e o  $O_{ER}$  que está no alcance de ambos e escuta toda a comunicação. A Figura 38 representa a comunicação entre o nodo emissor C e o nodo receptor D, os nodos F, B, J são os nodos ouvintes.

Este tipo de problema não chega a ser um problema para interfaces operando com taxa de transferências acima de  $2Mbps$  [?]. Porém, para dispositivos que operam com uma baixa taxa a economia de energia pode ser significativa. Considerando os valores da Tabela 10 pode-se obter as frações de  $t_{oc}$  e  $t_{RX}$  dos nodos ouvintes. A Tabela 11 mostra o consumo dos nodos ouvintes sem a utilização da função CNO.

Com os valores da Tabela 11 pode-se calcular o consumo dos três tipos de nodos ouvintes durante a comunicação, utilizando o tamanho de pacote de dados de  $64bytes$ .

Potência média para o nodo ouvinte  $O_{ER}$ :

$$P_m = 0,30 \times P_{oc} + 0,70 \times P_{RX} \quad (5.25)$$

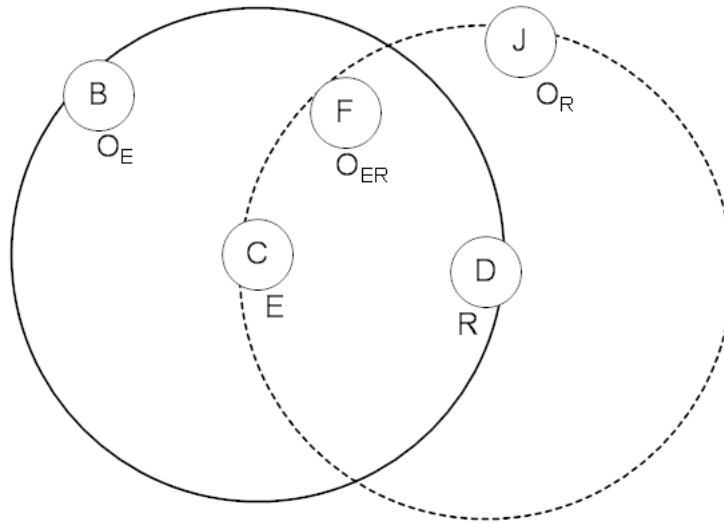


Figura 38: Nodos ouvintes.

Tabela 11: Fração de tempo dos nodos ouvinte para cada estado sem CNO.

Tamanho do Pacote (bytes)	$O_{ER}$			$O_E$			$O_R$		
	$t_{oc}$	$t_{TX}$	$t_{RX}$	$t_{oc}$	$t_{TX}$	$t_{RX}$	$t_{oc}$	$t_{TX}$	$t_{RX}$
48	0,32	-	0,68	0,52	-	0,48	0,80	-	0,20
64	0,30	-	0,70	0,47	-	0,53	0,83	-	0,17
160	0,19	-	0,81	0,30	-	0,70	0,89	-	0,11
250	0,14	-	0,86	0,22	-	0,78	0,92	-	0,08
500	0,08	-	0,92	0,13	-	0,87	0,95	-	0,05
1000	0,04	-	0,96	0,07	-	0,93	0,97	-	0,03

$$P_{m_{O_{ER}}} = 31,96mW$$

Potência média para o nodo ouvinte  $O_R$ :

$$P_m = 0,83 \times P_{oc} + 0,17 \times P_{RX} \quad (5.26)$$

$$P_{m_{O_R}} = 7,87mW$$

Potência média para o nodo ouvinte  $O_E$ :

$$P_m = 0,47 \times P_{oc} + 0,53 \times P_{RX} \quad (5.27)$$

$$P_{m_{O_E}} = 24,23mW$$

A Tabela 12 mostra o comportamento dos três tipos de nodos ouvintes durante a comunicação utilizando a função CNO. Como pode-se observar a principal diferença é que os nodos ocupam a fração de tempo  $t_s$ , isso acontece por que assim que o nodo identifica que não é o destinatário da mensagem, este entra no seu estado *sleep* até zerar seu contador CNO.

A potência média para o nodo ouvinte  $O_{ER}$  com CNO:

$$P_m = 0,61 \times P_s + 0,27 \times P_{oc} + 0,12 \times P_{RX} \quad (5.28)$$

$$P_{m_{O_{ER}}} = 5,5mW$$

Tabela 12: Fração de tempo dos nodos ouvinte para cada estado com CNO.

Tamanho do Pacote (bytes)	$O_{ER}$			$O_E$			$O_R$		
	$t_{oc}$	$t_S$	$t_{RX}$	$t_{oc}$	$t_S$	$t_{RX}$	$t_{oc}$	$t_S$	$t_{RX}$
48	0,30	0,57	0,13	0,30	0,57	0,13	0,30	0,50	0,20
64	0,27	0,61	0,12	0,27	0,61	0,12	0,27	0,56	0,17
160	0,17	0,75	0,08	0,17	0,75	0,08	0,17	0,72	0,11
250	0,12	0,83	0,05	0,12	0,83	0,05	0,12	0,80	0,08
500	0,08	0,89	0,03	0,08	0,89	0,03	0,08	0,87	0,05
1000	0,04	0,95	0,01	0,04	0,95	0,01	0,04	0,94	0,02

Potência média para o nodo ouvinte  $O_R$  com CNO:

$$P_m = 0,56 \times P_s + 0,27 \times P_{oc} + 0,17 \times P_{RX} \quad (5.29)$$

$$P_{m_{O_R}} = 7,70mW$$

Potência média para o nodo ouvinte  $O_E$  com CNO:

$$P_m = 0,61 \times P_s + 0,27 \times P_{oc} + 0,12 \times P_{RX} \quad (5.30)$$

$$P_{m_{O_E}} = 5,5mW$$

Como pode-se observar, com a utilização da função CNO a potência de consumo média no nodo ouvinte foi consideravelmente menor, caindo de  $31,96mW$  para  $5,5mW$  no  $O_{ER}$ , de  $24,23mW$  para  $5,5mW$  no  $O_E$  e  $7,87mW$  para  $7,7mW$  no  $O_R$ . Essa diferença é bastante significativa para uma RSSF densa, isto é, como um grande número de sensores interagindo entre si.

Devido as dificuldades encontradas para simulação e prototipação do protocolo proposto, optou-se por realizar a validação do mesmo matematicamente. Para isso, no presente capítulo foram utilizadas equações de consumo, e os resultados quantitativos obtidos a partir de dados de dispositivos reais foram analisados de forma tabular. Comparando os resultados do tempo de vida dos nodos que utilizam o ciclo reduzido de operação com os nodos do modelo tradicionais, observa-se que o ganho no tempo de vida foi de 23 vezes. Outra análise realizada foi em relação aos nodos ouvintes que no melhor dos casos tiveram um ganho de energia de 5,8 vezes em comparação ao protocolos tradicionais.

## 6 Conclusão e Trabalhos Futuros

O presente trabalho iniciou como parte de um projeto de pesquisa desenvolvido no Centro de Pesquisa em Software Embarcado (CPSE/HP-PUCRS). Este projeto tinha como principal objetivo o desenvolvimento de um sistema para validação de usuários, controle de acesso e gerenciamento de informações de usuários. Também, poderia ser utilizado para determinar a localização dos usuários, informar os serviços disponíveis na rede e controlar avisos de serviços de restaurações de dados de usuários, uma primeira abordagem foi apresentada em [3].

O suporte em hardware para esse ambiente era dado por uma plataforma chamada de *Beacon*, também desenvolvida no CPSE. A plataforma *Beacon* apresentava uma arquitetura muito similar a arquitetura de um nodo sensor, pois possui uma unidade de processamento, uma unidade de comunicação e alguns periféricos, como alguns sensores).

O protocolo R-MAC foi inicialmente projetado para garantir uma comunicação eficiente com baixo consumo de energia entre os *Beacon*, sendo parte integrante do projeto citado acima. Porém, a pesquisa foi descontinuada pelo órgão financiador do projeto antes de sua conclusão. Com isso foram encontradas dificuldades para realizar testes do protocolo R-MAC em uma plataforma em hardware, pois as primeiras versões da plataforma não dispunham de módulo de comunicação RF.

Para realizar a validação do R-MAC sem uma plataforma de prototipação, em um primeiro momento foi investigada a utilização de redes de autômatos estocásticos (SAN) [18]. Porém, em uma SAN não é possível computar o tempo que o evento permaneceu em cada estado, o que compromete a análise do consumo de energia, que é diretamente dependente do tempo.

Em um segundo momento, foi utilizado o simulador de redes *ns-2*. Testes com o protocolo MAC disponível no simulador, o *IEEE802.11*, foram realizados a fim de realizar uma comparação com o R-MAC. Contudo, para fazer esta comparação era necessário implementar o R-MAC na linguagem *TCL*, linguagem interpretada pelo simulador *ns-2*, e integrar seu código no ambiente do simulador. Pela complexa estrutura do simulador *ns-2* e pela falta de documentação sobre as funções de implementação, suas dependência e interações com as demais bibliotecas de *ns-2*, as tentativas de integração do R-MAC no simulador *ns-2* não foram validadas com sucesso.

Em um último momento, partiu-se para uma abordagem analítica para validar o R-MAC. Para isso foi utilizada uma plataforma de um nodo com o módulo de comunicação CC1100, da

*TexasInstruments*, como estudo de caso. Os dados relativos ao consumo foram retirados dos *datasheets* de cada módulo.

Através desta análise foi possível analisar as funcionalidades do R-MAC. Obtendo a fração de tempo que os nodos ficam em cada estado, foi calculada a potência média do nodo para vários cenários. Sendo possível estimar o tempo de vida do nodo. Calculando a fração de tempo que os nodos ouvintes permanecem em cada estado, foi possível comprovar que a estratégia CNO é eficaz para reduzir o consumo da RSSF.

O protocolo proposto tem suas funcionalidades direcionadas para atender RSSFs que trabalham com aplicações dirigidas a eventos e coleta periódica de dados. Para atender estes requisitos foi proposto o ciclo reduzido de operação, permitindo a configuração tanto do tamanho do ciclo quanto do *duty cycle*, economizando energia pelos ciclos reduzidos e por evitar a fragmentação de pacotes.

Ainda para atender os requisitos dos eventos foi proposto o pacote EVT, que é um pacote de controle de alta prioridade que sinaliza um evento. Aproveitando a característica de baixa taxa de envio que, normalmente, é utilizada pelas RSSFs, foi proposta a função CNO, que permite economia de energia colocando os nodos ouvintes em modo *sleep*.

A redução do tamanho dos pacotes de controle também impacta no consumo de energia. A função de ajuste de energia de transmissão permite que nodos estabeleçam novas conexões com a rede. Esta função é útil no caso do nodo ficar isolado, aumentando seu tempo de interação com a RSSF. Outra funcionalidade do R-MAC é a lista de vizinhos, criada quando a RSSF começa a operar.

O R-MAC consegue atingir um ganho de até 23 vezes no tempo de vida quando o nodo se comporta como emissor e um ganho de até 5,8 vezes quando o nodo se comporta como ouvinte. As funcionalidades de conexão e sincronismos entre os nodos permitem que a RSSF fique ativa por mais tempo, aproveitando melhor o tempo de vida da rede.

No presente momento, o R-MAC faz parte de uma pesquisa de mestrado orientada pelo professor Fabiano Hessel. Este projeto tem por objetivo a implementação de uma arquitetura que permita o monitoramento e a caracterização de ambientes térmicos em estufas agrícolas. O R-MAC será implementado de acordo com as características do *assembly* da arquitetura proprietária escolhida para o projeto.

A Tabela 13 apresenta os resultados de um estudo comparativo entre o R-MAC e os demais protocolos investigados nessa dissertação.

Tabela 13: Comparação do R-MAC com outros protocolos MAC para RSSF.

Protocolo	Características	Tipo de RSSF	Vantagem	Desvantagem
S-MAC	ciclo de operação reduzido	dirigidas a eventos, com coleta periódica, insensível a latência da rede e com baixa taxa de envio de mensagens	prolonga consideravelmente o tempo de vida da rede	aumento da latência devido ao período <i>sleep</i>
T-MAC	ciclo de operação reduzido com incremento de um temporizador	dirigidas a eventos, com coleta periódica, insensível a latência da rede e com baixa taxa de envio de mensagens	consegue um consumo mais eficiente que o S-MAC, devido ao incremento do temporizador	é extremamente limitado com relação a largura de banda da rede, problemas com a sincronização dos nodos
ARC	provê <i>fairness</i> adaptando a taxa de transmissão do tráfego, evita colisões adicionando um atraso aleatório antes do tempo de escuta	coleta periódica de dados, sensível a latência com baixa ou alta taxa de envio de mensagens, RSSF com grande número de nodos	provê uma distribuição justa e mantém uma razoável largura de banda	não é eficiente para redes com um tráfego alto
DE-MAC	ciclo de operação reduzido, utiliza um algoritmo distribuído para balanceamento da carga na rede	coleta periódica, insensível a latência da rede e com baixa taxa de envio de mensagens	o balanceamento de carga serve para obter um tempo de vida uniforme para todos os nodos da RSSF	o algoritmo de balanceamento pode comprometer no caso de uma rede dirigida a eventos
TRAMA	utiliza um algoritmo de eleição distribuído para evitar colisões em comunicações <i>unicast</i> , <i>broadcast</i> ou <i>multicast</i> , alterna acessos entre aleatórios e escalonados	insensíveis à latência, com alta taxa de entrega	acomoda mudanças de topologia da rede, adição de nodos a rede e tolerância a falhas	devido seu algoritmo de eleição há uma grande troca de informações na rede, desfavorecendo a economia de energia.
R-MAC	ciclo de operação reduzido configurável, reconexão dos nodos na RSSF, redução do consumo dos nodos ouvintes, lista de vizinhos	dirigidas a eventos, com coleta periódica, insensível a latência da rede e com baixa taxa de envio de mensagens	permite a reconexão de nodos na RSSF, prolonga consideravelmente o tempo de vida da rede	aumento da latência devido ao período <i>sleep</i>





## Referências Bibliográficas

- [1] Nael Abu-Ghazaleh, Kyoung-Don Kang, and Ke Liu. Towards resilient geographic routing in wsns. In *Q2SWinet '05: Proceedings of the 1st ACM international workshop on Quality of service & security in wireless and mobile networks*, pages 71–78, New York, NY, USA, 2005. ACM.
- [2] Seema Bandyopadhyay and Edward Coyle. An energy efficient hierarchical clustering algorithm for wireless sensor networks. *Proc. IEEE INFOCOM '03*, pages 1713–1723 vol.3, 2003, New York, NY, USA, IEEE.
- [3] Eduardo Augusto Bezerra, Ana Cristina Benso da Silva, and João Batista de Oliveira. Mobile systems at cpse. *II HP Brazil Tech Symposium, sem anais*, Porto Alegre, RS, Brazil, 2005.
- [4] Shah Bhatti, James Carlson, Hui Dai, Jing Deng, Jeff Rose, Anmol Sheth, Brian Shucker, Charles Gruenwald, Adam Torgerson, and Richard Han. Mantis os: an embedded multithreaded operating system for wireless micro sensor platforms. *Mob. Netw. Appl.*, 10(4):563–579, 2005, Hingham, MA, USA, ACM.
- [5] David Braginsky and Deborah Estrin. Rumor routing algorithm for sensor networks, *Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications, WSNA 02*, pages 22–32, Atlanta, Georgia, USA., 2002, ACM.
- [6] Chevally, Robert Van Dyck and Timothy Hall. Self-organization protocols for wireless sensor networks. *In Thirty Sixth Conference on Information Sciences and Systems*, New York, NY, USA, 2002, IEEE.
- [7] Chipcon/TI, Datasheet CC1100, Disponível em <http://www.chipcom.com>, Último acesso: 15/02/2007,
- [8] TI/Texas Instruments, Datasheet TRF6903, Disponível em <http://www.ti.com>, Último acesso: 15/02/2007,
- [9] TI/Texas Instruments, Datasheet TIR1000, Disponível em <http://www.ti.com>, Último acesso: 15/02/2007,
- [10] TI/Texas Instruments, Datasheet MSP430, Disponível em <http://www.ti.com>, Último acesso: 15/02/2007.
- [11] TI/Texas Instruments, Datasheet TSM470, Disponível em <http://www.ti.com>, Último acesso: 15/02/2007.
- [12] Pressure Sensor, Datasheet SM5310SMI, Disponível em <http://www.chipdocs.com>, Último acesso: 15/02/2007.

- [13] Temperature Sensor, Datasheet AD7418, *Disponível em <http://www.chipdocs.com>*, Último acesso: 15/02/2007.
- [14] Magnetometer Sensor, Datasheet AA002-02 NVE, *Disponível em <http://www.chipdocs.com>*, Último acesso: 15/02/2007.
- [15] Light Sensor, Datasheet H53371, *Disponível em <http://www.chipdocs.com>*, Último acesso: 15/02/2007.
- [16] Stefan Dulman and Paul Havinga. Operating system fundamentals for the eyes distributed sensor network. In Proceedings 3rd workshop on Embedded Systems, New York, NY, USA, pages 34–38, 2002, ACM.
- [17] Edgar Callaway Jr. *Wireless Sensor Networks Architectures and Protocols*. Boca Raton, Florida, USA, 2004, Auerbach Publications.
- [18] Paulo Fernandes, Brigitte Plateau, and William Stewart. Efficient descriptor-vector multiplications in stochastic automata networks. *J. ACM*, 45(3):381–414, New York, NY, USA, 1998, ACM.
- [19] Matthew Gast. *802.11 Wireless Networks: The Definitive Guide*. O’Reilly & Associates, Inc., Sebastopol, CA, USA, 2002.
- [20] Dean Gratton. *Bluetooth Profiles: The Definitive Guide*. New York, NY, USA, 2002, Pearson Education.
- [21] Chunlong Guo, Lizhi Charlie Zhong, and Jan Rabaey. Low power distributed mac for ad hoc sensor radio networks. *Global Telecommunications Conference, 2001. GLOBECOM ’01. IEEE*, 5:2944–2948 vol.5, New York, NY, USA, 2001, IEEE.
- [22] Wendi Heintzelman, Anantha Chandrakasan, and Hari Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *Wireless Communications, IEEE Transactions on*, 1(4):660–670, New York, NY, USA, 2002, IEEE.
- [23] Jason Hill, Mike Horton, Ralph Kling, and Lakshman Krishnamurthy. The platforms enabling wireless sensor networks. *Commun. ACM*, 47(6):41–46, New York, NY, USA, 2004, ACM.
- [24] Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, and Kristofer Pister. System architecture directions for networked sensors. In *Architectural Support for Programming Languages and Operating Systems*, pages 93–104, Vol. 34, New York, NY, USA, 2000, ACM.
- [25] Jason Lester Hill. *System architecture for wireless sensor networks*. PhD thesis, 2003, Berkeley, California, USA. University of Berkeley.
- [26] Andrew Howard, Maja Matarić, and Gaurav Sukhatme. An incremental self-deployment algorithm for mobile sensor networks. *Auton. Robots*, 13(2):113–126, New York, NY, USA, 2002, ACM.
- [27] Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *MobiCom ’00: Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 56–67, New York, NY, USA, 2000. ACM Press.

- [28] Ramaraju Kalidindi, Rajgopal Kannan, Sitharama Iyengar, and Lydia Ray. Distributed energy aware mac layer protocol for wireless sensor networks. In *International Conference on Wireless Networks*, pages 282–286, Las Vegas, Nevada, USA, 2003, CSREA Press.
- [29] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. Tossim: Accurate and scalable simulation of entire tinyos applications. In *Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003)*, New York, NY, USA, 2003, ACM.
- [30] Millennial Net. Disponível em <http://www.millennial.net>, Disponível em <http://www.millennial.net>, Último acesso: 15/02/2007.
- [31] Bob O’Hara and Al Petrick. *The IEEE 802.11 Handbook A Designer’s Companion*. New York, NY, USA, 1999, Standards Information Network IEEE Press.
- [32] Ron Olexa. *Implementing 802.11, 802.16, and 802.20 Wireless Networks: Planning, Troubleshooting, and Maintenance*. Burlington, MA, USA, 2003, Elsevier.
- [33] Adrian Perrig, Robert Szewczyk, Doug Tygar, Victor Wen, and David Culler. Spins: Security protocols for sensor networks. In *Wireless Networks*, pages 521–534, vol.8, Hingham, MA, USA, 2002, ACM.
- [34] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *SenSys ’04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 95–107, New York, NY, USA, 2004. ACM.
- [35] John Porter, Peter Arzberger, Hans-Werner Braun, Pablo Bryant, Stuart Gage, Todd Hansen, Paul Hanson, Chau-Chin Lin, Fang-Pang Lin, Timothy Kratz, William Michener, Sedra Shapiro, and Thomas Williams. Wireless sensor networks for ecology. *BioScience*, pages 561–572, vol. 55, Albuquerque, NM, USA, 2005. ACM.
- [36] Cauligi Raghavendra, Krishna Sivalingam, and Taieb Znati, editors. *Wireless sensor networks*. Kluwer Academic Publishers, Norwell, MA, USA, 2004.
- [37] Venkatesh Rajendran, Katia Obraczka, and Jose Joaquim Garcia-Luna-Aceves. Energy-efficient collision-free medium access control for wireless sensor networks. In *SenSys ’03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 181–192, New York, NY, USA, 2003. ACM.
- [38] Guilherme Rohde and Eduardo Bezerra. Confiabilidade em sistemas operacionais embarcados. In *Trabalho Individual*, PUCRS, Porto Alegre, RS, Brasil, 2004.
- [39] Eugene Shih, Seong-Hwan Cho, Nathan Ickes, Rex Min, Amit Sinha, Alice Wang, and Anantha Chandrakasan. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In *MobiCom ’01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 272–287, New York, NY, USA, 2001. ACM.
- [40] Pradeep Sinha. *Distributed Operating Systems: Concepts and Design*. New York, NY, USA, 1996, Wiley-IEEE Press.
- [41] Bernard Sklar. *Digital communications: fundamentals and applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

- [42] Katayoun Sohrabi, Jay Gao, Vishal Ailawadhi, and Gregory Pottie. Protocols for self-organization of a wireless sensor network. *IEEE Personal Communications*, pages 16–27, vol. 7, New York, NY, USA, 2000. IEEE.
- [43] Andrew Tanenbaum. *Redes de Computadores*. Rio de Janeiro, RJ, Brasil, Editora Campus, 2003.
- [44] Tijs van Dam and Koen Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 171–180, New York, NY, USA, 2003, ACM.
- [45] Brett Warneke, Matt Last, Brian Liebowitz, and Kristofer Pister. Smart dust: Communicating with a cubic-millimeter computer. *Computer*, pages 44–51, vol.33, New York, NY, USA, 2001, IEEE.
- [46] Alec Woo and David Culler. A transmission control scheme for media access in sensor networks. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 221–235, New York, NY, USA, 2001. ACM.
- [47] Wei Ye, John Heidemann, and Deborah Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Trans. Netw.*, pages 493–506, vol. 12, New York, NY, USA, 2004, IEEE/ACM.
- [48] Feng Zhao and Leonidas Guibas. *Wireless Sensor Networks: An Information Processing Approach*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.