



Universidade Federal Fluminense

UNIVERSIDADE FEDERAL FLUMINENSE
FACULDADE DE ENGENHARIA ELÉTRICA

DISSERTAÇÃO DE MESTRADO

**MODELAGEM E DESENVOLVIMENTO DE UM SISTEMA
INTELIGENTE DE GERÊNCIA DE VAGAS DE
ESTACIONAMENTO EM VIAS PÚBLICAS DE NOVA
GERAÇÃO UTILIZANDO REDES MODERNAS DE
TELECOMUNICAÇÕES**

Wagner da Silva Zanco

Niterói

2013

Wagner da Silva Zanco

Modelagem e Desenvolvimento de um Sistema Inteligente de Gerência de Vagas de Estacionamento em Vias Públicas de Nova Geração Utilizando Redes Modernas de Telecomunicações

Dissertação apresentada à Faculdade de Engenharia Elétrica da Universidade Federal Fluminense como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia de Telecomunicações.

Orientador: Prof. Dr. Andrés Pablo Lopez Barbero

Niterói

2013

X3	Zanco, Wagner da Silva Modelagem e Desenvolvimento de um Sistema Inteligente de Gerência de Vagas de Estacionamento em Vias Públicas de Nova Geração Utilizando Redes Modernas de Telecomunicações Dissertação (Mestrado em Engenharia de Telecomunicações) - Universidade Federal Fluminense, 2013. Orientador: Prof. Dr. Andrés Pablo Lopez Barbero 1. Sistemas de Telecomunicações. 2. Rede sem fio. 3. GPRS. 4. Microcontroladores. 5. Sistema de Gerência. 6. Sistemas Embarcados. 7. Mobilidade Urbana Sustentável. 8. Parquímetro. XXXXXXXXXXXXXX
----	---

COMISSÃO JULGADORA - DISSERTAÇÃO DE MESTRADO

Candidato: Wagner da Silva Zanco

Data da Defesa: 17 de maio de 2013

Titulo da Dissertação: "Modelagem e Desenvolvimento de um Sistema Inteligente de Gerência de Vagas de Estacionamento em Vias Públicas Utilizando Redes Modernas de Telecomunicações"

Prof. Dr. Andrés Pablo Lopez Barbero _____

Prof. Dr. Cláudio Kitano _____

Prof. Dr. Fernando Hideo Fukuda _____

Prof. Dr. João Marcos Meireles da Silva _____

Resumo

Vivemos um ambiente de grandes transformações em nosso modo de vida. Tecnologias como internet, sistemas embarcados, redes de comunicações móveis, entre outras, têm contribuído de forma significativa para o desenvolvimento tecnológico em vários setores, tais como industrial, comercial, predial, biomédico e automobilístico. O conceito de cidade inteligente vem ganhando força a medida que esforços têm sido canalizados para a redução do consumo de energia e da emissão de gases de efeito estufa. Esta dissertação propõe uma nova geração para os sistemas de gerência de vagas de estacionamento automotivo em vias públicas. Integrando soluções de redes modernas de telecomunicações, sistemas baseados em programação de alto nível orientada a objetos e tecnologia de sistemas embarcados, o sistema de gerência proposto pode contribuir de forma significativa para a construção de uma política de gerência de vagas de estacionamento em vias públicas, com vistas a reduzir o tráfego no centro das cidades, diminuir a emissão de gases de efeito estufa, viabilizar uma forma mais justa de pagamento pela utilização do solo para estacionamento e, entre outros aspectos, contribuir para a melhora na mobilidade urbana sustentável.

Palavras-Chave - Sistemas de Telecomunicações, Rede sem fio, GPRS, Microcontroladores, Sistema de Gerência, Sistemas Embarcados, Mobilidade Urbana Sustentável, Parquímetro.

Abstract

We live in an environment of great change in our way of life. Technologies like internet, embedded systems, mobile communications networks, among others, have contributed significantly to the technological development in various sectors such as industrial, commercial, building, biomedical and automotive. The concept of digital city is gaining strength as the efforts have been directed towards the reduction of energy consumption and emission of greenhouse gases. This dissertation proposes a new generation to the management systems automotive parking spaces on public streets. Integrating solutions to modern telecommunications networks, systems based on high-level programming and object-oriented embedded systems technology, the proposed management system can contribute significantly to building a policy management of parking spaces on public streets, in order to reduce traffic in city centers, reducing the emission of greenhouse gases, achieving a fairer way of paying for the use of land for parking and, among other things, contribute to the improvement of sustainable urban mobility.

Keywords - Telecommunication Systems, Wireless, GPRS, Microcontrollers, System Management, Embedded Systems, Sustainable Urban Mobility, Parking Meter.

"Não somos o que pensamos ser, mas sim o fruto das nossas ações."

Wagner Zanco

Agradecimentos

Em primeiro lugar, a Deus por ter me dado forças para encarar este grande desafio. À minha esposa, a professora Roma Fátima de B. O. Zanco e à minha filha Grécia de B. O. Zanco pelo apoio incondicional nas incontáveis horas em que estive ausente para dedicar-me ao Mestrado.

A toda comunidade do Centro Vocacional de Educação Tecnológico Itatiaia pelo apoio que me deram durante todo o tempo em que estive a frente daquela querida escola.

Aos meus colegas professores da Universidade Estácio de Sá, em nome do professor coordenador eng. João Roberto onde ministrou a disciplina Microcontroladores e Sistemas Embarcados.

A todos os meus professores do Mestrado, em especial aos professores Dr. Andrés Pablo e Dr. Ricardo Marques e a todos os meus colegas do mestrado, em especial aos amigos Odair, Vinícius, Yanes, Nelson e Mitrione, pelo incondicional apoio nos momentos de maior dificuldade.

Agradeço a todos aqueles que, de forma direta ou indireta, contribuíram para que este objetivo se tornasse realidade. Que me perdoem aqueles que porventura eu tenha me esquecido de citar.

Figuras

Figura 2.1: O primeiro parquímetro.....	11
Figura 2.2: Antiga sala de controle de uma subestação energética.....	14
Figura 2.3: Centro de Operações Integradas de Controle moderno.....	15
Figura 2.4: Diagrama em blocos básico da arquitetura interna de um microcomputador.....	16
Figura 2.5: Arquitetura básica de um microcontrolador baseado na filosofia <i>Harvard</i>	18
Figura 2.6: Diagrama em blocos do microcontrolador PIC32MX460F512L.....	20
Figura 2.7: Diagrama em blocos do circuito oscilador da família PIC32MX.....	23
Figura 2.8: Comunicação serial assíncrona.....	24
Figura 2.9: Transmissão de 8 bits de dados e um <i>stop bit</i>	25
Figura 2.10: Comunicação serial RS-232.....	27
Figura 2.11: Rede de Comunicação RS-485 Mestre/Escravo.....	29
Figura 2.12: Rede de Comunicação RS-485 Mestre/Escravo.....	30
Figura 2.13: Conexão do <i>driver/receiver</i> em uma rede padrão RS-485.....	30
Figura 2.14: Equivalência dos modelos de referência eOSI e OSI.....	32
Figura 2.15: Fluxograma de Causa e Efeito de falhas.....	34
Figura 2.16: Pilha de comunicação ModBus.....	38
Figura 2.17: Pacote genérico ModBus.....	39
Figura 2.18: Categorias das funções.....	40
Figura 2.19: Funções públicas do protocolo ModBus.....	41
Figura 2.20: Modelo de dados separado por blocos.....	42
Figura 2.21: Estrutura do <i>frame</i> ModBus ASCII.....	43
Figura 2.22: <i>Frame</i> ModBus ASCII para a função 3.....	45
Figura 2.23: <i>Frame</i> genérico ModBus RTU.....	45
Figura 2.24: <i>Frame</i> ModBus RTU para a função 3.....	47
Figura 2.25: Fases da Tecnologia GSM.....	49
Figura 2.26: Componentes da Arquitetura GSM.....	51
Figura 2.27: Modelo OSI <i>versus</i> IP.....	57
Figura 2.28: Arquitetura GPRS.....	58
Figura 2.29: Componentes de um sistema RFID.....	61
Figura 2.30: Diagrama em blocos do <i>tag</i> MF1ICS50.....	62
Figura 3.1: Parquímetro em São José dos Campos, São Paulo.....	65

Figura 3.2: Parquímetro na cidade de Campo Grande, Mato Grosso do Sul.....	66
Figura 3.3: Parquímetro em Tóquio, Japão.....	67
Figura 3.4: Parquímetro em Londrina, Paraná.	67
Figura 3.5: Arquitetura do Sistema de Gerência <i>e-SIGA</i> e Suas Interconexões.....	68
Figura 3.6: Parquímetro – Equipamento de Interface Com o Usuário.	69
Figura 3.7: Sistema <i>e-SIGA</i> em um Cenário Típico.....	71
Figura 3.8: Aplicativo de Localização de Vagas Por GPS para S.O. Android.....	71
Figura 3.9: Equipamento de carga e recarga do cartão.	72
Figura 3.10: Arquitetura do SGVE.....	73
Figura 4.1: Diagrama em blocos do <i>hardware</i> da máquina de recarga.....	80
Figura 4.2: Diagrama em blocos do <i>hardware</i> do concentrador.	82
Figura 4.3: Diagrama em blocos do <i>hardware</i> do parquímetro.....	83
Figura 4.4: Desenvolvimento prático experimental do <i>E-SIGA</i>	102
Figura 4.5: <i>Software</i> Monitor Serial	103
Figura 4.6: Placa do parquímetro.	103
Figura 4.7: Comandos de cartão do leitor MF-CLASSIC.	104
Figura 4.8: Sensor de distância por ultrassom.....	104
Figura 4.9: Concentrador visão <i>Top</i>	105
Figura 4.10: Concentrador visão <i>Botton</i>	105
Figura 4.11: Máquina de carga e recarga.....	106

Tabelas

Tabela 2.1: Alterações na rede GSM de segunda geração introduzidas pelo GPRS	59
Tabela 4.1: Resposta a mensagem de verificação de parquímetro em operação.	84
Tabela 4.2: Mensagem de mudança de <i>status</i> do parquímetro.	84
Tabela 4.3: <i>Log</i> de utilização da vaga.	86
Tabela 4.4: Funções modeladas <i>versus</i> funções implementadas	102
Tabela 5.1: Indicadores da mobilidade urbana sustentável que podem ser influenciados por uma gestão inteligente das vagas de estacionamento automotivo em vias públicas.....	109

Lista de siglas

ADU - *Application Data Unit*

AMPS - *Advanced Mobile Phone Systems*

ARM - *Advanced RISC Machine*

ASCII - *American Standard Code for Information Interchange*

BSC - *Base Station Controller*

BSS - *Base Station System*

BSS - *Sistema de Estação Base*

BTS - *Base Transceiver Station*

CAN - *Controller Area Network*

CDMA - *Code Division Multiple Access*

CNPq - *Conselho Nacional de Desenvolvimento Científico e Tecnológico*

CPU - *Unidade Central de Processamento*

CR - *Carriage Return*

CRC - *Cyclic Redundancy check*

DCE - *Data Communication Equipment*

DMA - *Direct memory access*

DNS - *Domain Name System*

DTE - *Data Terminal Equipment*

EC - *Echo Canceler*

EEPROM - *Electrically-Erasable Programmable Read-Only Memory*

EIA - *Electronic Industries Association*

EIR - *Equipment Identity Register*

EMI - *Electromagnetic Interference*

eOSI - *Embedded Open Systems Interconnection*

FDMA - *Frequency Division Multiple Access*

FM - *Frequency Modulation*

FTP - *File Transfer Protocol*

GGSN - *Gateway GPRS Support Node*

GPRS - *General Packet Radio Service*

GPS - *Global Positioning System*

GSM - *Global System for Mobile communications*

HTTP - *Hypertext Transfer Protocol*
ICO-RTT - *ICO Radio Transmission Technology*
IMA - *Multimedias Subsystem*
IMEI - *International Mobile Equipment Identity*
IMSI - *International Mobile Subscriber Identity*
IMT-2000 - *International Mobile Telephone 2000*
IMUS - *Índice de Mobilidade Urbana Sustentável*
ISDN - *Integrated Service Digital Network*
ISO - *International Organization for Standardization*
IWF - *Interworking Function*
Key - ki - *Subscriber Authentication*
LAI - *Location Area Identity*
LCR - *Longitudinal Redundancy Check*
LF - *Line Feed*
LTE - *Long Term Evolution*
ME - *Mobile Equipment*
MIPS - *Microprocessor without interlocked pipeline stages*
MS - *Mobile Station*
MSC - *Mobile Services Switching Center*
MSISDN - *Mobile Station Integrated Services Digital Network*
MNGE - *Módulo de Gerência de Vagas de Estacionamento*
NA-WCDMA = *North American - Wideband CDMA*
NGMN - *Next Generation Mobile Network*
NSS - *Network Switching System*
NSS - *Sistema de Comutação de Rede*
OMS - *Operation and Maintenance System*
OSI - *Open Systems Interconnection*
PDU - *Protocol Data Unit*
PLC - *Programmable Logic Controller*
PLL - *phase lock loop*
POP3 - *Post Office Protocol*
PWM - *Pulse-width modulation*
RAM - *Random Access Memory*
RISC - *Reduced instruction set computing*

RPTC - Rede Telefônica Pública Comutada
RTU - *Remote Terminal Unit*
SAT-CDMA - *Satelite CDMA*
SCADA - *Supervisory Control and Data Aquisition*
SE - Sistema Embarcado
SGSN - *Serving GPRS Suport Node*
SIM - *Subscriber Identily Module*
SLVVI - Serviço de localização de Vagas Via Internet
SMTP - *Simple Mail Transfer Protocol*
SRAM - *Static Random Access Memory*
SSC - Sistemas de Supervisão e Controle
SWCDMA - *Satelite Wideband CDMA*
SW-CTDMA - *Satelite Wideband Hybrid CDMA/Y=TDMA*
TCP/IP - *Transmission Controle Provocou /Internet Provocou*
TDMA - *Time Division Multiple Access*
TD-SCDMA - *Time division Synchronous CDMA*
TMSI - *Temporary Mobile Subscriber Identily*
TTL - *Transistor-Transistor Logic*
UART - *Universal Asynchronous Receiver Transmitter*
UDP - *User Datagram Protocol*
UIT - *International Communication Union*
UMTS - *Universal Mobile Telecommunications System*
UWC-136 - *Universal Wireless Communications*
VLR - *Visitor Location Register*
WCDMA - *Wideband CDMA*

Sumário

Capítulo 1 - INTRODUÇÃO.....	1
1.1. Motivação.....	4
1.2. Objetivo do Estudo.....	6
1.3. Metodologia da Pesquisa.....	6
1.4. Atividades da Pesquisa.....	8
1.5. Desenvolvimento do Estudo.....	8
Capítulo 2 - REVISÃO TEÓRICA DAS TECNOLOGIAS ENVOLVIDAS.....	10
2.1. Parquímetro.....	10
2.2. Sistemas Embarcados.....	12
2.3. Sistemas de Supervisão e Controle.....	13
2.3.1. Sistemas SCADA.....	14
2.4. Microprocessadores <i>versus</i> Microcontroladores.....	16
2.5. Microcontrolador PIC32MX460FT512L - Uma Visão Geral.....	19
2.5.1. Arquitetura Básica do Microcontrolador PIC32MX460F512L.....	20
2.5.2. Periféricos.....	21
2.5.3. Monitoramento e Proteção.....	22
2.2.4. Oscilador.....	23
2.6. Comunicação Serial Assíncrona Universal (UART).....	24
2.6.1 Controle de Fluxo.....	26
2.7. Padrão RS-232.....	26
2.8. Padrão RS-485.....	28
2.9.1. Camada de Aplicação.....	32
2.9.2. Camada de Detecção.....	32
2.9.3. Camada de Recuperação.....	35
2.9.4. Camada de Redundância.....	35
2.9.5. Camada de Comunicação.....	36
2.9.6 Camada de Enlace.....	36
2.10. Protocolo ModBus.....	37
2.10.1. Modos de Transmissão ModBus sobre os Padrões RS-232/RS485.....	42
2.10.2. Modo de Transmissão ASCII.....	42
2.11.1. Arquitetura da Rede GSM.....	51
2.11.2. Rede GPRS.....	56
2.11.4. Arquitetura GPRS.....	58
2.11.5. Unidade de Controle de Pacote (PCU).....	59
2.11.6. Servidor do nó de suporte (SGSN).....	59
2.11.7. <i>Gateway</i> do nó de suporte GPRS (GGSN).....	59
2.11.8. Principais Cenários GPRS.....	60
2.12. Tecnologia RFID.....	60
2.12.1. Classificação do <i>Tag</i>	61
Capítulo 3 - DESCRIÇÃO DO SISTEMA.....	64

3.1. Modelo Atual de Gerência.....	64
3.2. Arquitetura do Sistema Proposto.....	68
3.1. Equipamento de Interface Com o Usuário	69
3.2. Software Para Serviço de Consulta de Vagas Via Internet	71
3.4. Concentrador	72
3.5. Software de Gerenciamento de Vagas de Estacionamento.....	73
3.6. Dinâmica de utilização do Sistema <i>e-SIGA</i>	74
3.6.1. Ponto de vista do Usuário.....	74
3.5.2. Ponto de vista do Gestor	76
Capítulo 4 - DESENVOLVIMENTO DO SISTEMA.....	79
4.1. Máquina de Recarga.....	79
4.2. Concentrador	81
4.2.1. Comunicação Com os Parquímetros	83
4.3. Parquímetro	85
4.4.1 Exceções à Regra	87
4.4.2. Mensagens que Trafegam na Rede dos Parquímetros	88
4.4. Software de Gerenciamento de Vagas de Estacionamento (SGVE).....	88
4.4.1. Módulo de Gerência do Sistema de Carga e recarga (MGSCR)	89
4.4.2. Casos de Uso do MGSCR	90
4.5. Módulo de Gerência de Vagas de Estacionamento (MGVE).....	93
4.5.2. Casos de Uso do MGVE	94
4.6. Serviço de Localização de Vagas Via Internet (SLVVI)	101
4.7. Integração do Sistema <i>e-SIGA</i>	101
4.7.1. Rede Concentrador - Parquímetros.....	102
4.7.2. Máquina de Carga e Recarga	105
4.7.3. Rede Concentrador - SGVE	106
Capítulo 5 - BENEFÍCIOS E LIMITAÇÕES DO SISTEMA.....	107
5.1. Benefícios do Novo Modelo	107
Conclusão	112
Referências.....	114

Capítulo 1 - INTRODUÇÃO

Desde que as redes de telecomunicações comerciais entraram em funcionamento, na década de 1970, os sistemas de telecomunicações evoluíram de forma significativa, tanto no aumento da taxa de transferência quanto no alcance. Os sistemas de transmissão por satélite, as redes de fibras ópticas, as comunicações via rede de telefonia celular e outros sistemas de redes de telecomunicações, transformaram o mundo em uma verdadeira aldeia global, referência a uma nova visão de mundo e às comunidades conectadas entre si.

Simultaneamente à evolução dos sistemas de telecomunicações, ocorreu uma revolução na gerência dos processos de produção industrial com o desenvolvimento dos sistemas embarcados, do *Programmable Logic Controller* (PLC), dos computadores de pequeno porte e das redes locais. Essa revolução, iniciada na década de 1980, ficou conhecida como Automação Industrial e permitiu um aumento de produtividade não só nas indústrias, mas também na agricultura e em outros setores da sociedade.

A partir da década de 1990, a automação, até então um privilégio das indústrias, passou a ser utilizada na automatização de processos em quase todas as áreas do conhecimento. Como exemplo, podemos citar as áreas da medicina, automação predial, automação comercial, robótica, domótica, automobilística, análises clínicas entre outras. É importante ressaltar, no entanto, que sem as redes de telecomunicações, pouco se teria avançado em termos de tecnologia empregada na automação de processos, independente da área de sua aplicação.

A partir da década de 2000, começou a se fortalecer um novo conceito de cidade conhecido como Cidade Digital. ÁTILA *et al* (2006) afirmam que uma cidade digital é aquela que apresenta, em toda a sua área geográfica, infraestrutura de telecomunicações e internet, tanto para acesso individual quanto público, disponibilizando à sua população informações e serviços públicos e privados em ambiente virtual. Para Lemos (2005), esse é o tipo da cidade contemporânea permeada por redes telemáticas, que levam a uma reconfiguração do espaço urbano e das relações sociais.

A tendência natural de uma cidade digital é a de se transformar em uma cidade inteligente. A inteligência dos espaços urbanos deve ser encaminhada no sentido do desenvolvimento sustentado e pensado à escala global. É preciso encarar a tecnologia como a responsável pela alteração física, econômica e social dos espaços urbanos, isto é, a tecnologia é a principal responsável pela criação de espaços inteligentes. Da relação entre a criatividade e o conhecimento, pode surgir um novo conceito que integra as questões territoriais, interconectando o digital com o real, a cidade inteligente (GAMA, 2006).

Uma cidade inteligente é uma cidade baseada no conhecimento (GAMA, 2006) e, entre outras coisas, consiste na tomada de uma série de decisões que colocam os serviços governamentais no centro do processo de transformação. Essa transformação deve incluir a automatização dos serviços prestados ao cidadão, como o sistema de atendimento à saúde, a educação, o IPTU, o controle de tráfego, a gestão de estacionamento automotivo em vias públicas entre outros.

O tema de interesse deste estudo está relacionado com a melhoria da mobilidade urbana sustentável, mais especificamente no impacto da gestão do estacionamento automotivo em vias públicas. Entende-se mobilidade urbana sustentável como o resultado de um conjunto de políticas de transporte e circulação que visam proporcionar o acesso amplo e democrático ao espaço urbano, por meio da priorização dos modos de transporte coletivo e não motorizados, de forma efetiva, socialmente inclusiva e ecologicamente sustentável (MINISTÉRIO DAS CIDADES, 2004).

Após a comunidade científica definir o que vem ser a mobilidade urbana sustentável, buscou-se o desenvolvimento de uma avaliação dos indicadores envolvidos nessa área. Miranda (2012) afirma que:

"O desenvolvimento de um método único de avaliação para todas as cidades, como um índice capaz de monitorar o desempenho e a efetividade das políticas públicas, tornou-se um desafio para pesquisadores do mundo todo. Entre todos os debates sobre o tema, observou-se que o ferramental para avaliar a mobilidade

sustentável deveria contemplar questões essenciais relacionadas aos aspectos sociais, econômicos e ambientais."

Em busca da criação de uma metodologia que pode ser utilizada na quantificação da mobilidade urbana sustentável, Costa (2008) desenvolveu o Índice de Mobilidade Urbana Sustentável (IMUS) com o objetivo de oferecer uma metodologia capaz de avaliar e quantificar muitos dos aspectos pertinentes à mobilidade, incluindo aspectos sociais, econômicos e ambientais. O IMUS compõe uma estrutura complexa composta por oitenta e sete indicadores, não se constituindo em uma ferramenta simples de ser aplicada. Sendo uma ferramenta recente, foi pouco aplicada e ainda precisa ser submetida a avaliações no sentido de verificar sua efetividade no monitoramento e controle das políticas públicas aplicadas na área da mobilidade urbana sustentável (MIRANDA, 2010).

No contexto da mobilidade urbana sustentável, a gestão eficaz do estacionamento automotivo em vias públicas pode se mostrar uma estratégia muito importante dentre as utilizadas para a melhoria do transporte urbano. O estacionamento afeta a maneira como vivemos, afeta também a acessibilidade e pode influenciar no índice da mobilidade urbana sustentável. Além disso, as políticas que afetam a oferta e o custo do estacionamento podem desempenhar um papel fundamental na mudança da dependência do automóvel para modos de transportes mais ecológicos como, por exemplo, as calçadas de pedestres, ciclovias e os transportes coletivos (FILOSA, 2006).

Com o aumento de veículos em circulação, a busca por vagas de estacionamento em áreas urbanas é um problema que acaba por levar a desperdícios de tempo e de combustível. Um estudo realizado em Monique, na Alemanha, demonstra os prejuízos resultantes da procura de vagas, por ano, são de 20 milhões de euros de custo econômicos; 3,5 milhões de euros para gasolina e óleo diesel, que são desperdiçados na procura de vagas; 150.000 horas de tempo perdido. Além disso, a proporção de tráfego à procura de estacionamento é cerca de 44% de todo o tráfego (CALISKAN e GRAUPNER *et al*, 2006)

Em 2011 o número de automóveis nas metrópoles brasileiras ultrapassou a marca de 20 milhões de veículos. Este número representa aproximadamente 44% de toda a frota brasileira.

Nas principais metrópoles, o crescimento da frota de automóveis entre 2001 e 2011 foi de 77,8%, o que representa um aumento de 8,9 milhões de veículos novos circulando nas estradas brasileiras (INCT O. M., 2013).

Além dos problemas associados a falta de vagas de estacionamento, o aumento da frota de veículos automotores nos últimos anos vem acentuando alguns problemas inerentes às grandes cidades. O número cada vez maior de veículos circulando na rede viária tem provocado, entre outros problemas, congestionamentos nunca antes testemunhados, o que, por consequência, aumenta o tempo de deslocamento entre dois pontos distintos da cidade, com desdobramentos em muitos outros segmentos da sociedade (COELHO, 2009). O gasto em saúde, por outro lado, é agravado por conta do grande número de acidentes com vítimas ocorridos todos os anos nas estradas. Para Vasconcelos (2000), o crescimento da frota de veículos é um dos responsáveis pelo aumento do número de acidentes de trânsito. O Brasil gastou em 2011 cerca de R\$ 200 milhões de reais com internações, segundo dados do Ministério da Saúde (BRASIL, 2011). O aumento na emissão de gases de efeito estufa na atmosfera é outro grande problema que tem sido muito discutido por especialistas nos últimos anos, e tem como uma das causas a enorme quantidade de CO₂ emitida na atmosfera pelos veículos automotores movidos a combustíveis fósseis. No Brasil, os veículos automotores são o segundo maior emissor de CO₂, sendo responsáveis por 20% das emissões, perdendo apenas para o desmatamento e para as queimadas (CARVALHO, 2011).

O aumento do número de veículos automotores trafegando nas estradas trás consequências graves às grandes cidades. Muitas são as ações que precisam ser tomadas para minimizar os problemas provocados por esse aumento. A implantação de uma política eficaz de estacionamento é uma das ações que pode contribuir para a melhoria da mobilidade urbana sustentável.

1.1. Motivação

Em 2009 foi publicado pelo CNPq o edital MCT/CNPq nº 018/2009 - Pesquisa, Desenvolvimento e Inovação em Transporte, visando o financiamento de projetos com vias a contribuir significativamente para o desenvolvimento científico e tecnológico do País,

realizando ações voltadas para a pesquisa, para o desenvolvimento e para a inovação no setor de transportes, visando enfrentar os principais problemas do setor, seja no campo do transporte urbano ou do transporte de cargas.

O grupo de pesquisa do Laboratório de Comunicações Ópticas da Universidade Federal Fluminense, do qual sou parte integrante como aluno do programa de Mestrado em Engenharia de Telecomunicações e colaborador em projetos de P&D, submeteu a este edital, sob a coordenação dos professores doutores Ricardo Marques Ribeiro e Andrés Pablo Lopez Barbero, um projeto de modelagem e desenvolvimento de um sistema de gerenciamento inteligente de vagas de estacionamento automotivo em vias públicas utilizando redes modernas de telecomunicações, sistemas embarcados e programação de alto nível orientada a objeto. O projeto foi aprovado pelo CNPq, constando entre os quarenta e seis projetos aprovados no Brasil, sendo um dos oito projetos aprovados no Rio de Janeiro. Com um orçamento estimado em 400 mil reais, o projeto visa a modelagem o desenvolvimento experimental de um sistema inteligente de gerenciamento de vagas de estacionamento em vias públicas.

Devido a minha experiência profissional pregressa na área de eletrônica, especificamente, com sistemas embarcados, e com projetos de redes locais, além da significativa contribuição na modelagem e no desenvolvimento do referido projeto, optei em utilizar o objeto de estudo desse projeto como tema central para o desenvolvimento desta dissertação de mestrado, uma vez que o projeto demanda, em uma significativa parte do seu desenvolvimento, à tecnologia de sistemas embarcados e o projeto de redes locais.

Diante do exposto, podemos afirmar que esta dissertação é parte integrante do referido projeto, tendo como questão central do estudo a modelagem e o desenvolvimento de um sistema inteligente de gerência de vagas de estacionamento automotivo em vias públicas como parte de uma política de estacionamento eficaz, que possa contribuir com a melhoria da mobilidade urbana sustentável.

1.2. Objetivo do Estudo

Este estudo tem por objetivo propor a modelagem e o desenvolvimento de um sistema inteligente de gerência de vagas de estacionamento automotivo em vias públicas, baseado no uso de redes modernas de telecomunicações.

Analisando a produção teórica acerca do tema associado ao objeto deste estudo, a mobilidade urbana sustentável, foi verificado que a referida temática tem sido investigada à luz de diferentes áreas do conhecimento. Foram encontrados estudos relacionados à emissão de gases de efeito estufa (OLIVEIRA, 2011), gestão de qualidade de transportes coletivos (REINHOLD, 1996), impactos dos congestionamentos em vias pública (COELHO, 2009), gestão de semáforos (JUNIOR, 2007), mobilidade urbana sustentável (COSTA, 2003) e (COSTA, 2008), sistema de gestão de parquímetros (NEVES, 2009) entre outros.

Em função do objeto de estudo estar associado ao conceito de cidade inteligente e, este último, constituindo-se como uma área nova de interesse acadêmico, não foram encontrados estudos diretamente relacionados à modelagem de um sistema inteligente de gerência de vagas de estacionamento em vias públicas com o mesmo escopo do modelo proposto por esta dissertação na literatura pesquisada.

O modelo proposto é implementado em um sistema prático experimental intitulado **Sistema Inteligente de Vagas de Estacionamento em Vias Públicas - (e-SIGA)**, cuja arquitetura permite o controle remoto dos parquímetros por meio de uma central de operações, na qual um servidor faz a gerência de todo o sistema remotamente por meio da internet ou de uma rede de comunicação dedicada. Em funcionamento, o sistema pode contribuir para um sistema de gerência de vagas de estacionamento como parte de uma política de estacionamento eficaz, com vias a melhora da mobilidade urbana sustentável.

1.3. Metodologia da Pesquisa

Uma das coisas que caracterizam um trabalho de pesquisa é a sua classificação no que diz respeito à metodologia utilizada no desenvolvimento do estudo. A metodologia pode ser

classificada quanto a sua natureza, seu objetivo, forma de abordagem do problema e procedimentos técnicos adotados.

Quanto a sua natureza, a pesquisa pode ser classificada como pesquisa básica e pesquisa aplicada. A pesquisa básica visa gerar conhecimentos úteis para o avanço da ciência sem aplicação prática prevista, enquanto a pesquisa aplicada tem o objetivo de gerar conhecimentos para aplicação prática na solução de problemas específicos. Envolve verdades e interesses locais (MORESI, 2003). Segundo a sua natureza, este estudo é classificado como pesquisa aplicada, uma vez que propõe o desenvolvimento prático experimental de um sistema inteligente de gerência de vagas de estacionamento automotivo em vias públicas de nova geração.

Segundo Moresi (2003), uma pesquisa qualitativa considera que há uma relação dinâmica entre o mundo real e o sujeito, isto é, um vínculo indissociável entre o mundo objetivo e a subjetividade do sujeito que não pode ser traduzido em números. Sendo assim, podemos classificar esta pesquisa como sendo qualitativa, uma vez que entendemos existir uma relação dinâmica, direta e contextualizada entre os impactos do aumento de veículos automotores, a piora na mobilidade urbana sustentável e a proposta de redução dos impactos do aumento de veículos nas cidades pelo objeto deste estudo.

Quanto aos objetivos, este estudo pode ser classificado como intervencionista devido ao fato dele propor um novo sistema de gerenciamento de vagas de estacionamento em vias públicas que pode interferir, não só no mobiliário urbano, quanto na dinâmica de uso do solo para estacionamento em vias públicas, com impactos na mobilidade urbana sustentável.

Quanto aos procedimentos técnicos adotados, esta pesquisa pode ser classificada como pesquisa experimental. A pesquisa experimental permite observar e analisar um fenômeno, sob condições determinadas, e também se pode fazer investigação experimental no laboratório (MORESI, 2003).

1.4. Atividades da Pesquisa

Uma vez definido os procedimentos metodológicos da pesquisa, foi estabelecida a sequência de atividades a serem desenvolvidas para a sua execução.

Uma revisão bibliográfica buscou relacionar o impacto do aumento de veículos automotores com a piora da mobilidade urbana sustentável e, a partir de então, foram estabelecidas as bases para a definição das justificativas e dos objetivos do estudo.

Após serem definidos os objetivos do estudo, seguiu-se a modelagem do sistema proposto, a qual envolveu a modelagem de sistemas embarcados, de *software* de alto nível orientado a objetos e de redes locais.

Definida a modelagem do sistema, foi executada a parte prática experimental do estudo. Foram desenvolvidos os módulos parte integrante do sistema. São eles: máquina de carga e recarga, concentrador, parquímetro, *software* de gerenciamento remoto das máquinas de carga e recarga, dos concentradores e dos parquímetros.

Finalmente foram feitos os testes de integração do sistema, os quais configuram o resultado do estudo e as bases para futuros estudos.

1.5. Desenvolvimento do Estudo

O capítulo 1 faz um breve resumo da evolução da tecnologia utilizada em automação de processos até a introdução do conceito de cidade inteligente e de mobilidade urbana sustentável. É também exposta a motivação que levou a escolha do tema, assim como a é apresentada a questão central e definidos os objetivos do estudo.

No capítulo 2 é efetuado um estudo teórico das principais tecnologias utilizadas no desenvolvimento do sistema proposto por esta dissertação. Este estudo é baseado em levantamentos bibliográficos na literatura.

O capítulo 3 apresenta uma descrição do sistema de gerenciamento proposto, mostrando as principais características dos módulos que o compõe, assim como as principais tecnologias envolvidas.

O desenvolvimento prático experimental do sistema é apresentado no capítulo 4, no qual é apresentada a metodologia utilizada no seu desenvolvimento e as formas de integração entre os módulos.

O capítulo 5 apresenta alguns dos benefícios que poderão ser obtidos com a implantação do sistema, assim como os obstáculos a serem ultrapassados na sua implantação.

As considerações finais são apresentadas na conclusão do trabalho, assim como propostas para futuros aperfeiçoamentos e estudos futuros.

Capítulo 2 - REVISÃO TEÓRICA DAS TECNOLOGIAS ENVOLVIDAS

Neste capítulo é feita uma revisão teórica das principais tecnologias utilizadas no desenvolvimento prático experimental do estudo proposto por esta dissertação. O sistema de gestão proposto tem por base a utilização de parquímetro digitais, ligados em rede e gerenciados remotamente, para o controle e a gestão das vagas de estacionamentos automotivos em vias públicas. Para o desenvolvimento prático experimental do estudo foram utilizadas várias tecnologias. Dentre elas podemos destacar os sistemas embarcados, as redes de comunicação, tecnologia RFID e programação de alto nível orientada à objeto.

2.1. Parquímetro

Os parquímetro constituem um dos meios mais efetivos para gerenciar o estacionamento automotivo autorizado em via públicas. Esses equipamentos possuem uma ampla gama de características. Eles podem ser utilizados apenas como medidores de tempo ou também podem operar como coletores de dinheiro. Podem ser mecânicos, eletrônicos ou digitais, manuais ou automáticos. Podem ainda aceitar moedas ou cartões para o pagamento do estacionamento. É importante que eles sejam utilizados de forma a fixar estadias máximas para os veículos e formem uma barreira psicológica para os condutores (BULL, 2001). Essa barreira psicológica tem como objetivo disciplinar o utente com relação as políticas de estacionamento em vigor. Para facilitar a contextualização do modelo proposto por esta dissertação vamos dividir a tecnologias de gestão de estacionamentos automotivo efetuados com parquímetro em gerações.

A primeira iniciativa de automatizar a gerência do estacionamento automotivo em vias públicas remonta a década de 1930, quando o americano Carlton Cole Magee inventou o que viria a ser a primeira geração de parquímetro, sendo a cidade de Oklahoma a primeira a fazer uso do novo sistema. O precursor dos parquímetro era baseado em uma estrutura de ferro com uma

pequena abertura coletora de moedas. Por meio de uma alavanca o utente fazia a indicação do tempo em que ficaria com o carro estacionado. A Figura 2.1 mostra o primeiro parquímetro inventado.



Figura 2.1: O primeiro parquímetro.

A invenção do parquímetro foi tão bem recebida que a utilização dos parquímetros se difundiu por todo o mundo. Colégios, universidades, autarquias e entidades privadas implementaram esta ideia, estimando-se que hoje em dia, apenas nos Estados Unidos da América, haja cerca de cinco milhões de parquímetros em uso (NEVES, 2009).

Meio século depois do primeiro parquímetro ter sido inventado, a partir da década de 1980, novas tecnologias começaram a ser utilizadas na produção de parquímetros. Aperfeiçoamentos como leitor de RFID, *display* digital interativo, teclados funcionais, alimentação por energia solar fazem parte das tecnologias utilizadas na fabricação do que podemos chamar de segunda geração de parquímetros.

O aperfeiçoamento dos sistemas de gerência de estacionamento com parquímetro de segunda geração é proposta por esta dissertação, a qual denominaremos de terceira geração de sistemas de gestão de estacionamento com parquímetros. Tal proposta sugere a interligação dos parquímetros em rede, sendo eles gerenciados remotamente de uma sala de controle.

Associado ao controle remoto dos parquímetros o sistema pode fornecer recursos de localização de vagas por GPS via dispositivo móvel.

2.2. Sistemas Embarcados

A invenção dos dispositivos eletrônicos microprocessador e microcontrolador, deu origem a uma nova tecnologia, a dos computadores de pequeno porte, e obrigou os projetistas da área de eletrônica, que até então só desenvolviam circuitos eletrônicos, a aprenderem uma ou mais linguagens de programação de computadores, visto que a referida tecnologia envolve o desenvolvimento de circuitos eletrônicos (*hardware*) como também de programas de computador (*software*). Essa convergência de habilidades demorou a ser verificada. Até o início da década de 1980 existia uma distinção clara entre o profissional que atuava no desenvolvimento de *hardware* e o profissional que atuava no desenvolvimento de *software*. Com o tempo, no entanto, o profissional da área de desenvolvimentos com microprocessadores e microcontroladores passou a atuar nas duas frentes. É comum hoje em dia ver profissionais que desenvolvem soluções com sistemas gerenciados por microprocessadores ou microcontroladores trabalhar no desenvolvimento de *hardware* e de *software*.

Sistemas que envolvem o desenvolvimento de aplicações com a utilização de microprocessadores e ou microcontroladores, se combinados com unidades de armazenamento e com equipamentos eletromecânicos, passaram a desempenhar funções específicas dentro do contexto das aplicações industriais. Esse agrupamento de dispositivos, presentes numa grande variedade de sistemas, como os de telemetria, automotivos, os robóticos, os aeronáuticos e os de telecomunicações, são denominados Sistemas Embarcados (*Embedded Systems*) (MASSA, 2006).

O desenvolvimento de um sistema embarcado (SE) encontra algumas limitações que não são encontradas nos sistemas computacionais de propósito geral, como um Computador Pessoal PC. Dentre elas, podemos citar as limitações quanto a capacidade de processamento, o espaço reduzido de memória utilizada para armazenar o programa executável e para armazenar dados, a necessidade de redução da potência consumida durante operação, principalmente àqueles

alimentados por pilhas e baterias, as exigências quanto ao tempo esperado para que o sistema esteja em uso e suas possibilidades de expansão, além do elevado grau de confiabilidade exigido pelas aplicação embarcadas (PINHEIRO, 2009).

Os primeiros sistemas embarcados foram desenvolvidos para aplicações industriais. Com a evolução dos microcontroladores e das ferramentas de desenvolvimento, a redução do custos de desenvolvimento, o aumento da integração dos circuitos eletrônicos, o desenvolvimento da tecnologia aplicada à fabricação de sensores e a evolução das redes de comunicação, tais sistemas começaram a ser utilizados em outras áreas. Hoje em dia, praticamente todas as áreas do conhecimento utilizam de sistemas embarcados para a solução de problemas, monitoramento e automação de processos.

2.3. Sistemas de Supervisão e Controle

Com a evolução das redes de comunicação e dos recursos computacionais, os sistemas embarcados passaram a ser utilizados nos sistemas de supervisão e controle (SSC), amplamente utilizados na supervisão e controle de processos industriais. Um SSC é constituído por sensores, atuadores, equipamento de controle, rede de comunicação e sistema de monitoramento.

Sensores são dispositivos que convertem grandezas físicas em tensão/corrente elétrica. Os atuadores são dispositivos que intervém no sistema, de forma a executar uma determinada ação de controle, como ligar ou desligar uma máquina, acionar uma válvula proporcional etc. O equipamento de controle faz a coleta de dados e os enviam para a estação de monitoração através de uma rede de comunicação, e pode ser implementado por um sistema embarcado. A rede de comunicação é o meio utilizado para a troca de informações entre os equipamentos de controle e a estação de monitoração. As mais utilizadas são *ethernet*, *profibus*, RS-232, RS485 e *wireless*. A estação de monitoração, normalmente um sistema computacional do tipo SCADA, coleta os dados do sistema e os disponibiliza ao operador.

2.3.1. Sistemas SCADA

Sistemas SCADA (*Supervisory Control and Data Acquisition*) são sistemas baseados em programa de computador e comunicação de dados, utilizados para o monitoramento e controle de processos industriais, instalações físicas etc. Um sistema SCADA pode ser sistema SCADA comercial ou sistema SCADA proprietário. Um sistema SCADA comercial é desenvolvido por empresas de tecnologia e está disponível para compra no mercado. Um sistema SCADA proprietário é concebidos para aplicações dedicadas. São normalmente desenvolvidos em linguagem de programação de alto nível como C#, Visual Basic, Java, C++ etc.

Os primeiros Sistemas SCADA permitiam informar periodicamente o estado corrente do processo industrial e efetuavam o monitoramento de sinais representativos de medidas e estados de dispositivos, por meio de um painel de lâmpadas e indicadores, sem qualquer interface interativa com o operador. a Figura 2.2 mostra como era uma antiga sala de controle de uma subestação energética.



Figura 2.2: Antiga sala de controle de uma subestação energética (DACIOLE, 2009).

Os sistemas SCADA atuais utilizam tecnologias de computação e comunicação de dados para automatizar a monitoração e o controle de processos, para efetuar a coleta de dados em ambientes complexos, eventualmente dispersos geograficamente. Apresentam um de modo amigável para o operador, com recursos gráficos elaborados (interface homem-máquina), redes redundante, e conteúdo multimídia. a Figura 2.3 mostra um moderno Centro de Operações Integradas de Controle. As salas de controle atuais possuem um conjunto de computadores

peçoais executando sistemas SCADA, com os quais é possível a supervisão e o controle de todo o processo. Tais sistemas são caracterizados como sistemas distribuídos de controle quando são dotado de redes redundantes (via de dados) que permite a descentralização do processamento de dados e decisões, por meio do uso de unidades remotas na planta (PEREIRA, 2009).



Figura 2.3: Centro de Operações Integradas de Controle moderno (SMAR, 2008).

Para Finkel *et al* (2006), uma definição usual de um sistema distribuído de controle deve exprimir o fato de que uma máquina é responsável pela ação de controle, enquanto outra é responsável pela interface com o operador humano. O elemento de ligação entre os dois dispositivos microprocessados é a via de dados (*data highway*) do sistema. Por essa definição, basta que os dois processadores sejam distintos para se ter um sistema de controle distribuído.

Com a redução dos custos de desenvolvimento e do nível de integração, as tecnologias utilizadas na supervisão e controle de processos industriais, assim como os sistemas de controle distribuídos, foram adaptadas e passaram a ser empregadas no controle e supervisão de processos não industriais. A partir da década de 1980, tais tecnologias passaram a ser utilizadas em prol da mobilidade urbana sustentável como, por exemplo, no controle de tráfego e na gestão de estacionamentos.

2.4. Microprocessadores *versus* Microcontroladores

Em 1970, a empresa Intel Corporation apresenta o primeiro microprocessador do mundo, o 4004. O microprocessador 4004 era capaz de processar dados de 4 bits e foi desenvolvido para integrar uma calculadora eletrônica, cujo projeto foi encomendado à Intel pela empresa japonesa Busicom. Para aumentar o nível de integração exigido pelo projeto, a ideia era desenvolver um circuito integrado que fosse capaz de executar várias tarefas diferentes, em contrapartida aos circuitos integrados existentes na época, que eram capazes de executar uma única tarefa. O resultado foi o desenvolvimento do microprocessador 4004. O microprocessador 4004 foi projetado para executar 46 instruções e, para que pudesse executar cada uma das instruções, ele precisava receber um comando. A Intel ficou tão interessada no novo dispositivo que resolveu investir no aperfeiçoamento do microprocessador 4004. Nascia aí os alicerces para o futuro desenvolvimento dos computadores de pequeno porte, os microcomputadores.

Para que um computador possa funcionar, no entanto, ele precisa de outros blocos além do microprocessador, definido na arquitetura básica de um microcomputador como Unidade Central de Processamento (CPU). São eles os dispositivos de entrada de dados, os dispositivos de saída de dados, a memória de dados e a memória de programa. A Figura 2.4 mostra o diagrama em blocos básico da arquitetura interna de um microcomputador.

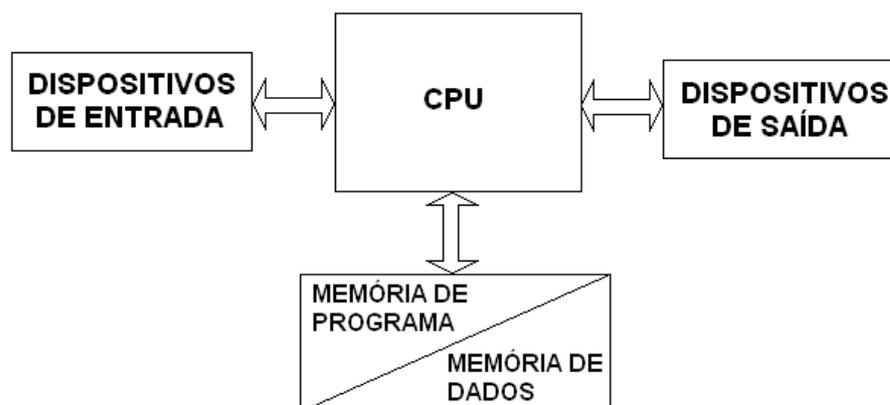


Figura 2.4: Diagrama em blocos básico da arquitetura interna de um microcomputador.

Os dispositivos de entrada e saída são os responsáveis pela entrada e saída de dados do sistema. Na memória de dados ficam armazenados os dados que o usuário fornece ao computador e os dados depois de processados. Na memória de programa ficam armazenados os comandos que a CPU deve executar para processar os dados. A sequência de comando que a CPU deve executar para processar os dados fornecidos pelo usuário é chamado de programa. A maior parte do processamento que a CPU executa sobre os dados é feito por meio de cálculos matemático. Os cálculos mais comuns são operações matemáticas de soma, subtração, multiplicação, divisão, rotação, incremento, decremento, complemento e funções lógicas. Além da CPU executar os programas e processar os dados é ela quem gerencia todo o sistema.

O primeiro microprocessador que foi amplamente utilizado em microcomputadores e em sistemas embarcados foi o microprocessador Z-80. O microprocessador Z-80 foi utilizado como CPU no videogame Atari, na linha de microcomputadores MSX, CP e TK entre outros. No final da década de 1970 a indústria de microcomputadores já movimentava bilhões de dólares em todo o mundo. Desde então os microprocessadores vêm evoluindo e ficando cada vez rápidos e com maior capacidade de processamento (STEVE, 1984) e (CYPRIANO, 1984).

No final da década de 1970 a Intel integrou todos os blocos básicos da arquitetura interna de um microcomputador em um único invólucro. Esse novo dispositivo a Intel denominou de microcontrolador. O microcontrolador é um dispositivo eletrônico baseado em circuito integrado capaz de executar diversas operações a partir de comandos armazenados em sua memória de programa. O microcontrolador acabou substituindo o microprocessador nos sistemas embarcados, e é hoje o principal ator dessa tecnologia.

O primeiro microcontrolador inventado, o 8048, lançado no final da década de 1970, era capaz de executar cálculos matemáticos entre dados de 8 bits. O projeto foi um grande sucesso, impulsionando a Intel a investir no aperfeiçoamento daquele dispositivo. Em 1983 a Intel lança o microcontrolador 8051, cuja filosofia de projeto foi baseada na arquitetura *Von-Neumann*. A Intel abriu a arquitetura do 8051 e permitiu que outros fabricantes produzissem microcontroladores utilizando como base a nova arquitetura. O microcontrolador 8051 se

tornou um padrão mundial durante toda a década de 1980 e 1990, sendo produzido por vários fabricantes como Motorola, Philips, Bosh entre outros.

Em 1989 a empresa Microchip Corporation entra no mercado com uma família de microcontroladores conhecida como família PIC. No projeto dos microcontroladores da família PIC, a Microchip utilizou uma filosofia baseada na arquitetura *Harvard*, e passou a ser a principal concorrente da família 8051 nos microcontroladores de 8 bits. Em 1993 a Microchip se tornou líder mundial em venda de microcontroladores de 8 bits, posto que ocupa até o dia em que esta dissertação foi publicada.

A Figura 2.5 mostra o diagrama esquemático simplificado de um microcontrolador baseado na arquitetura *Harvard*. Veja que o barramento de acesso à memória de programa é diferente do barramento de acesso à memória de dados. Esta é uma das principais diferenças entre as arquiteturas *Harvard* e *Von-Neumann* (ZANCO, 2008).

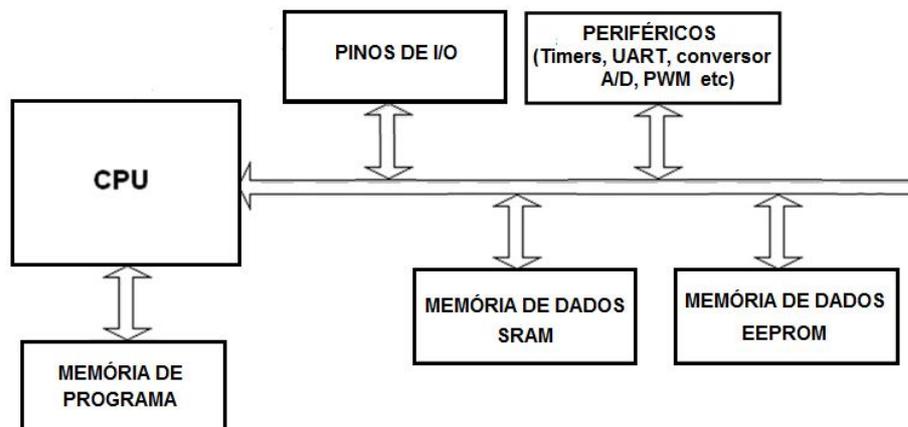


Figura 2.5: Arquitetura básica de um microcontrolador baseado na filosofia *Harvard*.

Além da memória de programa, um típico microcontrolador possui uma memória de dados volátil (SRAM) e uma memória de dados não volátil (EEPROM). Um conjunto de circuitos periféricos como temporizadores, conversor A/D, interfaces de comunicação serial, entre outros, fornece ao microcontrolador uma flexibilidade que o permite ser utilizado em aplicações de grande complexidade, aliviando a CPU de algumas tarefas que podem, com o mínimo esforço, ser desenvolvidas por um circuito periférico. Os pinos de I/O permitem que os

dados entrem e saem do microcontrolador através dos seus pinos. A informação aplicada em um pino de I/O pode ser de natureza analógica ou digital.

Na década de 1980 foi desenvolvida uma nova arquitetura de microprocessador, conhecida como arquitetura ARM. A empresa multinacional ARM Holdings, com sede em Cambridge e detentora da patente da arquitetura ARM, permitiu que outros fabricantes utilizassem a sua arquitetura para a produção de microcontroladores. Projetada para processar dados de 32 bits, a arquitetura ARM detém uma boa fatia do mercado de aplicações para 32 bits. Sendo um microcontrolador com grande capacidade de processamento, o ARM é indicado para aplicações de grandes complexidades como *tablets*, telefones celulares, decodificadores de TV a cabo, sistema de direcionamento de mísseis etc.

No início da década de 2000 a empresa Microchip lançou a sua própria família de microcontroladores de 32 bits, conhecida como família PIC32, sendo esta uma concorrente direta dos microcontroladores baseados na arquitetura ARM. A família PIC32 não seguiu a arquitetura ARM, tendo a Microchip optado em utilizar a arquitetura MIPS[®], desenvolvido pela empresa MIPS Computer Systems, com sede nos Estados Unidos.

2.5. Microcontrolador PIC32MX460FT512L - Uma Visão Geral

Todo o *hardware* projetado especificamente para a implementação do sistema proposto por esta dissertação foi baseado no microcontrolador PIC32MX460FT512L, pertencente a família PIC32 da empresa Microchip. Tal escolha se deve ao fato de que o modelo escolhido atende a todos os requisitos técnicos do projeto, associado a sólida ambientação do autor com os microcontroladores da família PIC. O processamento em 32 bits foi adotado para facilitar futuras atualizações no sistema proposto, que pode agregar outras funcionalidades, além das definidas no escopo desta dissertação, como pode ser observado na conclusão deste estudo. A Figura 2.6 mostra diagrama em blocos da arquitetura interna da família PIC32MX3XX/4XX, da qual faz parte o microcontrolador PIC32MX460FT512L (MICROCHIP_1, 2010).

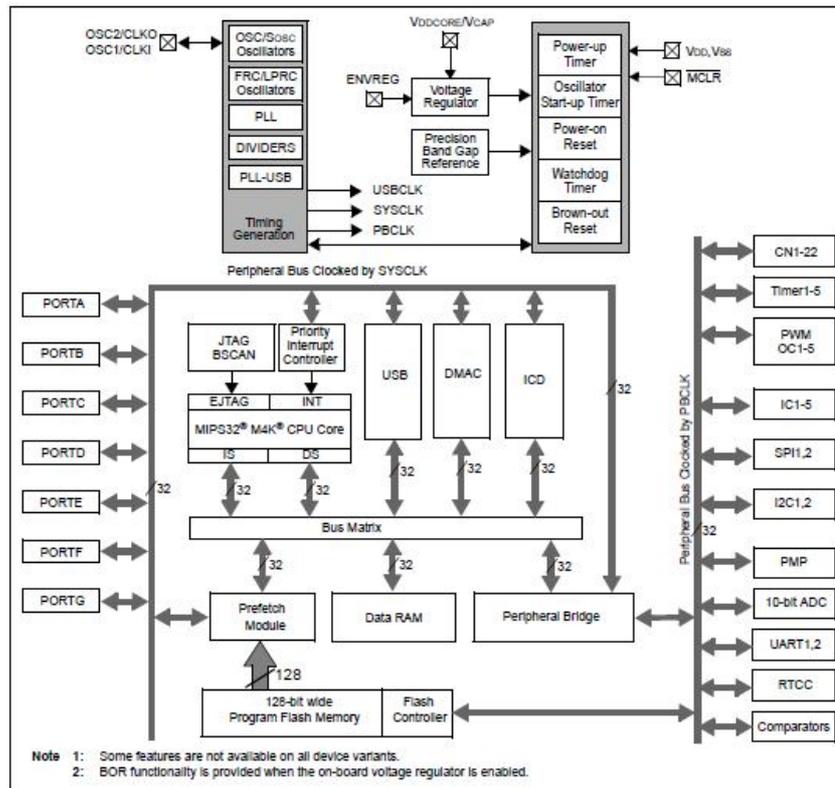


Figura 2.6: Diagrama em blocos do microcontrolador PIC32MX460F512L (MICROCHIP_1, 2010).

2.5.1. Arquitetura Básica do Microcontrolador PIC32MX460F512L

O Microcontrolador PIC32MX460F512L possui 512 kB de memória de programa do tipo Flash-ROM, 12 kB adicionais de *boot Flash*, 32 kB de memória de dados volátil de tipo SRAM, múltiplos modos de gerenciamento de energia, múltiplos vetores de interrupção com programação individual de até 63 níveis de prioridade, gerenciamento de falhas no circuito oscilador, oscilador independente para a operação do circuito de proteção contra travamento do oscilador e temporizador *Watchdog Timer*.

O Microcontrolador PIC32MX460F512L é um microcontrolador de 32 bits de alta performance, utiliza como *core* o núcleo M4K[®], da arquitetura MIPS[®], cujo conceito básico utilizado no seu desenvolvimento inicial era aumentar muito o desempenho com o uso de *pipelines* profundos para as instruções, que deveriam ser executadas em um único ciclo de *clock*. O core M4K[®], até a data da publicação desta dissertação, é o estado da arte dos microcontroladores RISC de 32 bits com núcleo de baixa potência. Desenvolvido com a

tecnologia *Enhanced MIPS32[®] Release 2 Instructions Set Architecture*, pode operar com frequência de até 80MHz (MICROCHIP, 2008). O microcontrolador PIC32MX460F512L tem a capacidade de efetuar multiplicação de dados de 32 bits em um único ciclo de *clock*, além de obter alta performance em operações aritméticas de divisão.

A CPU integrada ao microcontrolador PIC32MX460F512L é conectada ao sistema por meio de dois barramentos. São eles o barramento de instruções, o *I-side bus*, e o barramento de dados, o *D-side bus*. Ambos os barramentos são conectados ao barramento *matrix*. O barramento *matrix* gerencia múltiplos acessos concorrentes em um sistema. Ele permite o acesso simultâneo de múltiplos mestres que estão tentando se comunicar com diferentes dispositivos escravo por meio de um algoritmo de *arbitration*. Com dois barramentos diferentes de acesso ao barramento *matrix*, a CPU pode, ao mesmo tempo em que está buscando uma instrução do programa na memória de programa, estar acessando a memória RAM ou algum periférico. Uma ponte de periféricos (*peripheral bridge*) conecta o barramento da maioria dos periféricos ao barramento *matrix*. A *peripheral bridge* também conecta o controlador de DMA, o *in-circuit debugger* e a interface USB no barramento *matrix*. Todos, juntamente com a CPU, são concorrentes ao acesso a algum dispositivo do sistema por meio do barramento *matrix* (MICROCHIP, 2008).

2.5.2. Periféricos

O microcontrolador PIC32MX460F512L possui um conjunto de periféricos que fornece ao dispositivo uma poderosa capacidade de processamento. Com um *hardware* dedicado, um periférico pode executar tarefas que seriam normalmente executadas pela CPU caso ele não estivesse integrado ao microcontrolador. Quando uma determinada tarefa é executada por um periférico, a CPU fica livre para executar outras funções, aumentando a capacidade de processamento e a flexibilidade do sistema. Os periféricos integrados ao microcontrolador PIC32MX460F512L são cinco módulos temporizadores, cinco módulos PWM/Compare, cinco módulos Capture, duas interfaces seriais UART, um conversor A/D de 10 bits, duas interfaces I²C, duas interfaces SPI, uma interface paralela, um módulo comparador, um RTCC e vinte e duas entradas de interrupção por mudança de nível lógico.

2.5.3. Monitoramento e Proteção

Vários circuitos de monitoramento e proteção garantem que o sistema se mantenha em funcionamento, evitando possíveis travamentos em caso de falha ou ruído no sistema. São eles: *Power-on Reset*; *Power-up Timer*; *Start-up Timer Oscillator*; *Watchdog Timer*; *Brown-out Reset*.

O *Power-on Reset* é o primeiro *Reset* que ocorre no microcontrolador. Ele acontece quando é detectada a tensão de alimentação no microcontrolador. Em outras palavras, o *Power-on Reset* ocorre quando o circuito é energizado.

O *Power-up Timer* provê um *time-out* típico de 72ms após o *Power-on Reset*. Um circuito RC interno é o responsável pela temporização. O *Power-up Timer* é um recurso cujo objetivo é manter o microcontrolador em *Reset* até que a tensão de alimentação esteja estabilizada.

O *Start-up Timer Oscillator* é um circuito que provê um *delay* de 1024 ciclos de *clock* do oscilador principal, após o *Power-up Timer*, antes de colocar o circuito oscilador em funcionamento. Isso garante que o oscilador esteja estabilizado na sua partida, evitando travamento.

O *Watchdog Timer* é um recurso muito poderoso utilizado para evitar a paralisação do sistema em caso de travamento do oscilador devido a uma interferência eletromagnética, ou qualquer outro tipo de ruído que o faça parar de oscilar. O *Watchdog Timer* é um temporizador que provoca um *Reset* no programa sempre que ele produzir um *time-out*, ou seja, quando o tempo programado para o *time-out* terminar sem que seja executada a instrução CLRWDT. Sua utilização é indicada para sistemas em que o programa não pode parar de ser executado, como no caso de um computador de bordo de um automóvel, por exemplo. O *Watchdog Timer* possui um oscilador independente do oscilador principal. Isso significa que se, por algum motivo, o oscilador principal travar, o *Watchdog Timer* continua em funcionamento.

O *Brown-out Reset* é um recurso que, quando está ativado, faz o microcontrolador resetar caso a tensão de alimentação caia abaixo de um determinado valor, que pode ser configurado por *software*. Uma vez que o microcontrolador entrou em *Reset* devido ao *Brown-out Reset*,

ele só volta a executar o programa quando a tensão de alimentação voltar ao nível especificado para correto funcionamento do microcontrolador (ZANCO, 2010).

2.2.4. Oscilador

Toda CPU necessita de um sinal de *clock* que é utilizado como referência de tempo para a execução do programa. O sinal de *clock* é um sinal digital com uma determinada frequência, o qual é enviado para a CPU e utilizado como base de tempo para a busca e a execução de cada instrução do programa. O sinal de *clock* é gerado por um circuito comumente chamado de circuito oscilador. O microcontrolador PIC32MX460F512L possui uma variedade de opções para o circuito oscilador. Existem quatro opções para a origem de *clock* do oscilador. Um circuito PLL integrado permite a multiplicação e/ou a divisão do sinal de *clock* gerado pelo oscilador, o qual permite a alteração da frequência do sinal de *clock* recebido pela CPU. A seleção da origem do sinal de *clock* pode ser efetuada por *software*, inclusive alterando a origem do sinal de *clock* em caso de falha do oscilador. A Figura 2.7 mostra o diagrama em blocos do sistema de oscilador do PIC32MX460F512L.

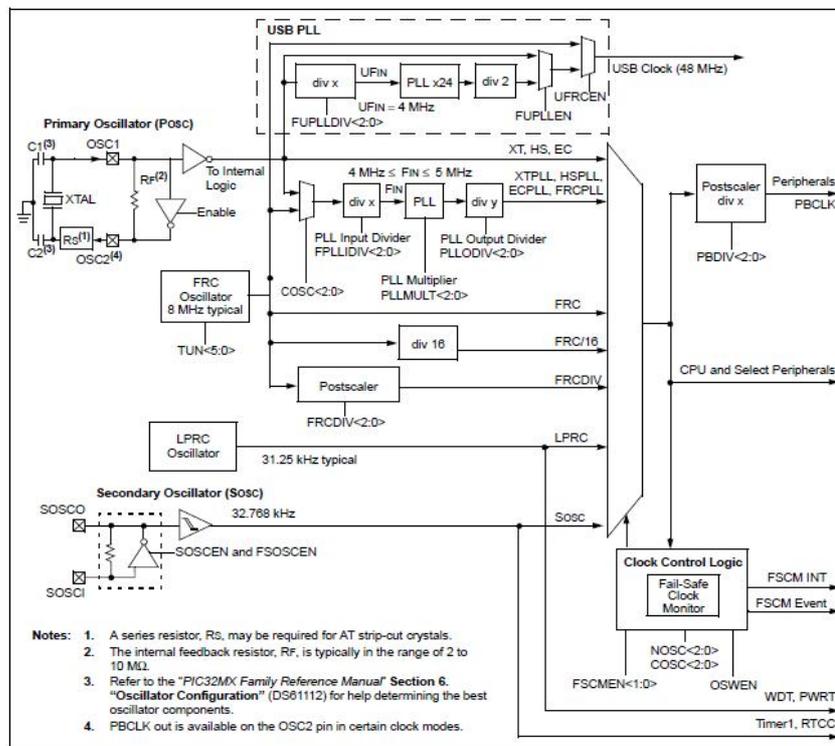


Figura 2.7: Diagrama em blocos do circuito oscilador da família PIC32MX (MICROCHIP_2, 2010).

Veja na Figura 2.7 que existe um oscilador primário e um oscilador secundário. Ambos têm a sua frequência definida pelo cristal conectado, respectivamente, nos pinos OSC1 e OSC2 ou nos pinos SOSCO e SOSCI. Outros dois osciladores internos, o FRC e o LPRC, os quais não necessitam de nenhum componente externo, podem ser utilizados como origem do sinal de *clock*. Um sinal de *clock* de 48MHz, derivado do circuito oscilador é utilizado pela interface USB. Um conjunto de bits, configurados por *software*, são utilizados para multiplicar e ou dividir o sinal de *clock*, além de acionar o circuito PLL para a obtenção do sinal de *clock* que é enviado para a CPU e para os periféricos. O circuito *Fail-Safe Clock Monitor* monitora o sinal de *clock* e efetua o chaveamento do circuito oscilador em caso de falha do oscilador principal (MICROCHIP_2, 2010).

2.6. Comunicação Serial Assíncrona Universal (UART)

UART (*Universal Asynchronous Receiver Transmitter*) é um protocolo de comunicação serial assíncrono desenvolvido na década de 1960 para permitir a comunicação ponto a ponto entre computadores mainframes e computadores terminais remotos (ZANCO, 2010).

Uma única linha de transmissão é suficiente para implementar uma comunicação serial assíncrona, como mostra a Figura 2.8. A transmissão dos dados é feita bit a bit e não necessita de um sinal de *clock* para sincronizar o envio dos bits.

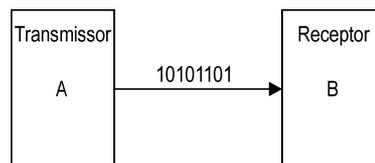


Figura 2.8: Comunicação serial assíncrona.

Para que haja a sincronização entre o transmissor e o receptor cada bit deve se manter na linha de transmissão durante um intervalo de tempo denominado período do bit, também conhecido como tempo de saída do bit ou *time bit*. O período do bit define a taxa de transferência utilizada na transmissão, dada em bits por segundo (bps). A Figura 2.9 apresenta os níveis

lógicos que estarão presentes na linha de transmissão para a transmissão de 8 bits de dados e 1 *stop bit*.

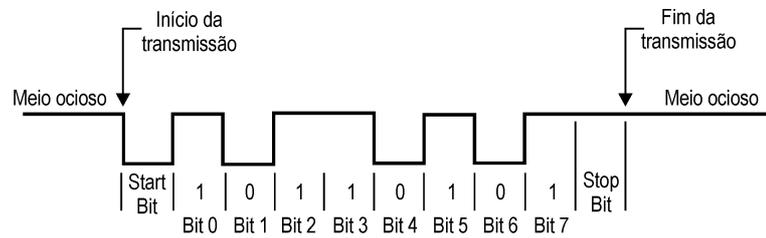


Figura 2.9: Transmissão de 8 bits de dados e um *stop bit*.

Quando não existe nenhuma transmissão em andamento, a linha de transmissão se encontra em um estado chamado de meio ocioso, representado pelo nível lógico '1' presente na linha. Uma transmissão tem início com o *start bit* que é uma transição do nível lógico '1' para o nível lógico '0' estando o meio ocioso. O *start bit* é a sinalização para o receptor de que uma transmissão está sendo iniciada. Após o envio do *start bit*, que tem a duração de um período do bit, são enviados os bits de dados, que podem ser 4, 5, 6, 7 ou 8 bits, os quais são colocados na linha de transmissão começando pelo bit menos significativo (LSb). Cada bit se mantém na linha de transmissão por um intervalo de tempo igual ao do *start bit*, ou seja, durante o período do bit. Após a transmissão dos bits de dados o transmissor coloca na linha de transmissão o *stop bit* e finaliza a transmissão. O *stop bit* pode ser de uma, uma e meia ou duas vezes o período do bit.

Um bit opcional pode ser transmitido após o envio dos bits de dados, o qual se chama bit de paridade. Sua função é detectar possíveis erros de transmissão. A paridade pode ser par (*even*), ímpar (*odd*), marca (*mark*) ou espaço (*space*). Na paridade par a soma dos bits de dados iguais a '1' mais o bit de paridade tem de ser par. Na paridade ímpar a soma dos bits de dados iguais a '1' mais o bit de paridade tem de ser ímpar. Na paridade marca o bit de paridade é sempre '1'. Na paridade espaço o bit de paridade é sempre '0'.

A taxa de transferência, o número de bits de dados, a inclusão ou não do bit de paridade e o tamanho do *stop bit* devem ser configurados antes do início da transmissão, e devem ser os mesmos tanto para o transmissor quanto para o receptor.

2.6.1 Controle de Fluxo

Não é a todo instante que o receptor está apto para receber dados. Sendo assim, um sistema de controle de fluxo pode ser implementado para que o transmissor não envie dados para o receptor em uma hora imprópria, correndo o risco de que dados sejam perdidos. O controle de fluxo pode ser implementado de duas formas diferentes: controle de fluxo por *hardware* (*out-of-band*) e controle de fluxo por *software* (*in-band*).

O controle de fluxo por *hardware* consiste em associar às linhas TX e RX linhas que deverão sinalizar para o transmissor se o receptor está ou não apto a receber dados e vice-versa. As linhas de controle de fluxo mais utilizadas são RTS e CTS.

O controle de fluxo por *software* consiste no envio de caracteres especiais, que tem o objetivo de informar ao transmissor se o receptor está ou não apto a receber dados no momento. Um protocolo de controle de fluxo muito conhecido é o XON/XOFF. Os caracteres enviados são DC1 (XON) e DC3 (XOFF), representados, respectivamente, no código ASCII pelos valores 11h e 13h. O receptor envia um caractere XOFF se ele desejar uma pausa na transmissão. Da mesma forma, o receptor envia um caractere XON se ele desejar que a transmissão recomece.

2.7. Padrão RS-232

O RS-232 é um padrão de jure de nível físico cuja primeira versão foi publicada no início da década de 1960 pelo órgão EIA (*Electronic Industries Association*). O padrão RS-232, cujo nome foi redefinido para EIA-232, define características mecânicas, elétricas e funcionais para a comunicação serial de dados entre um Equipamento de Dados Terminal (computador, impressora etc.) e um Equipamento de Comunicação de Dados, como, por exemplo, um *modem*. O equipamento de dados terminal (*Data Terminal Equipment*) é identificado pelo acrônimo DTE. O equipamento de comunicação de dados (*Data Communication Equipment*) é identificado pelo acrônimo DCE. A Figura 2.10 mostra uma típica comunicação serial envolvendo a interface RS-232.

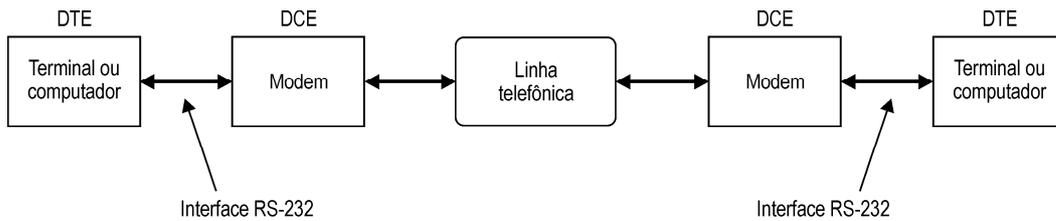


Figura 2.10: Comunicação serial RS-232.

O comprimento máximo do cabo que interliga o DTE ao DCE não é definido pelo padrão RS-232, mas a capacitância máxima da linha é limitada a 2500pF, com uma impedância de carga que pode variar entre 3kΩ e 7kΩ. Estas duas características limitam, no entanto, o comprimento do cabo em 20 metros (ZANCO, 2009).

O padrão RS-232 não define o protocolo de comunicação utilizado, porém tanto a comunicação serial síncrona quanto a comunicação serial assíncrona são suportadas. A comunicação serial assíncrona universal (UART) é a mais utilizada pelo padrão RS-232 por não precisar de um sinal de *clock* para sincronizar a transmissão, como acontece na comunicação síncrona, podendo ainda a comunicação assíncrona ser iniciada a qualquer momento.

Os sinais utilizados em uma comunicação serial assíncrona são divididos em dois grupos. São eles os sinais de dados e sinais de controle. Os pinos por onde trafegam os sinais de dados são apenas dois, os pinos TD e RD, respectivamente TX e RX. Eles são os responsáveis pela transmissão e pela recepção de dados. Os outros sinais são utilizados no controle de fluxo de dados e no *status* de funcionamento. O sinal DTR (saída) indica para o DCE que o DTE está pronto para a transferência de dados. O sinal RTS (saída) no DTE solicita autorização ao DCE para enviar dados. O sinal CTS (entrada) no DCE autoriza DTE enviar dados. O sinal DSR (entrada) indica que o DCE está pronto para a comunicação. O sinal DCD (entrada) indica para o DTE que o DCE está recebendo um sinal de portadora da linha telefônica, ou seja, o *link* de comunicação está em processo. O pino RI (entrada) indica para o DTE que o DCE está recebendo uma chamada telefônica. O Gnd é o terra. O padrão RS-232 utiliza transmissão não balanceada, que significa que os terminais de terra de ambos DTE e DCE devem estar conectados.

2.8. Padrão RS-485

A limitação do padrão RS-232 em relação à distância, além de ter sido desenvolvido para a conexão entre dois dispositivos, levou ao desenvolvimento do padrão RS-485. Padronizado como EIA-485, o padrão RS-485 utiliza um par de fios com tensão diferencial entre eles para representar os níveis lógicos presentes na linha de transmissão. A transmissão por tensão diferencial é indicada para transmissão em altas velocidades, longas distâncias e em ambientes propícios à interferência eletromagnética (ATUAL, 2007).

O padrão RS-485 permite a comunicação entre vários elementos participantes em uma mesma rede de dados. O número de terminais por nó em uma rede física padrão RS-485 normalmente é limitado em 32, podendo este número ser ampliado se for utilizado o *driver* correto (DA SILVA, 2005).

Devido a sua baixa imunidade a ruídos e o alcance de longas distâncias, o padrão RS-485 foi inicialmente utilizado em ambientes industriais. Sua utilização, no entanto, ultrapassou os limites industriais e passou a ser utilizado em outros segmentos, como automação comercial, residencial, entre outros. Com um par de fios condutores e uma tensão diferencial entre eles pode-se implementar uma comunicação bidirecional no formato *half-duplex* com o padrão RS-485. É possível, no entanto, com a introdução de um segundo *driver* e um segundo par de fios condutores, implementar uma comunicação *full-duplex*.

Como o padrão RS-485 não prevê tratamento contra colisão, normalmente uma rede que o utiliza na camada física segue a filosofia Mestre/Escravo. Neste tipo de comunicação o dispositivo mestre sempre dá início a comunicação enviando para a rede um pacote com o endereço lógico do dispositivo escravo com o qual ele irá se comunicar. Todos os dispositivos escravos conectados à rede vão receber o pacote, porém somente o dispositivo escravo endereçado irá responder a requisição do dispositivo mestre. Com o canal de dados entre o mestre e o escravo estabelecido, são então enviados os dados. A Figura 2.11 mostra o diagrama em blocos de uma rede RS-485 com a configuração Mestre/Escravo.

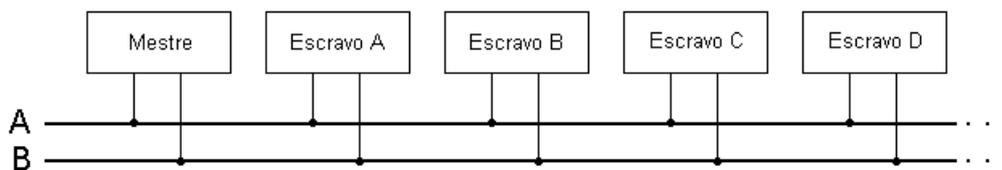


Figura 2.11: Rede de Comunicação RS-485 Mestre/Escravo.

Assim como o padrão RS-232, o padrão RS-485 também não define o protocolo de comunicação utilizado, sendo utilizado com mais frequência o protocolo UART. A taxa de transferência em uma transmissão RS-485 depende da distância entre os dispositivos mais distantes da rede, podendo atingir 100 kbps a uma distância de 1200 metros. A taxa de transferência pode chegar a 12 Mbps a uma distância de 12 metros. Maiores distâncias podem ser obtidas em uma transmissão RS-485 com a utilização de *drivers* repetidores (MAXIM, 2003).

O padrão RS-485 utiliza a transmissão balanceada, ou seja, não necessita compartilhar o terra dos dispositivos conectados à rede. No entanto, para minimizar os ruídos na transmissão, um terceiro fio condutor ligado ao terra dos dispositivos conectados à rede é normalmente utilizado como blindagem. Desta forma, em uma comunicação RS-485 não raramente são utilizados três fios condutores. São eles as linhas de dados A e B, por onde trafegam os dados codificados por uma tensão diferencial entre elas, e o fio condutor terra utilizado como blindagem. A linha B positiva em relação a linha A representa o nível lógico '1'. A linha B negativa em relação a linha A representa o nível lógico '0'. A tensão diferencial mínima entre as linhas A e B é de 200 mV, podendo chegar a 6 V. Uma tensão abaixo de 200 mV entre as linhas A e B é considerado ruído. A Figura 2.12 mostra as conexões de uma típica rede RS-485 com um mestre e três escravos. Veja que dois resistores de terminação de 120 Ω finaliza as duas extremidades da rede. Esse procedimento evita que sinais refletidos possam provocar ruídos e a consequente perda de dados na transmissão.

Quando um gerador coloca pulsos de curta duração em uma linha (sinal direto) que não possui a mesma impedância da fonte, esses sinais refletem-se parcialmente quando atingem o fim da linha. Em outras palavras, parte do sinal retorna pela linha de transmissão (sinal refletido). A combinação dos sinais direto e refletido pode provocar uma distorção nos pulsos enviados pelo

gerador, desta forma, o nível lógico presente na linha de transmissão pode sofrer uma distorção e deixar de ser reconhecido. A impedância da seção transmissora dos integrados EIA-485 é conhecida, mas o mesmo não acontece com a impedância característica do barramento, que depende de seu comprimento físico e da razão de transmissão de dados utilizada. No entanto, a experiência mostra que, se nas duas extremidades do barramento for colocado um resistor de 120 Ω , o equilíbrio de impedâncias fica garantido na maior parte dos casos (EUGENI, 2007).

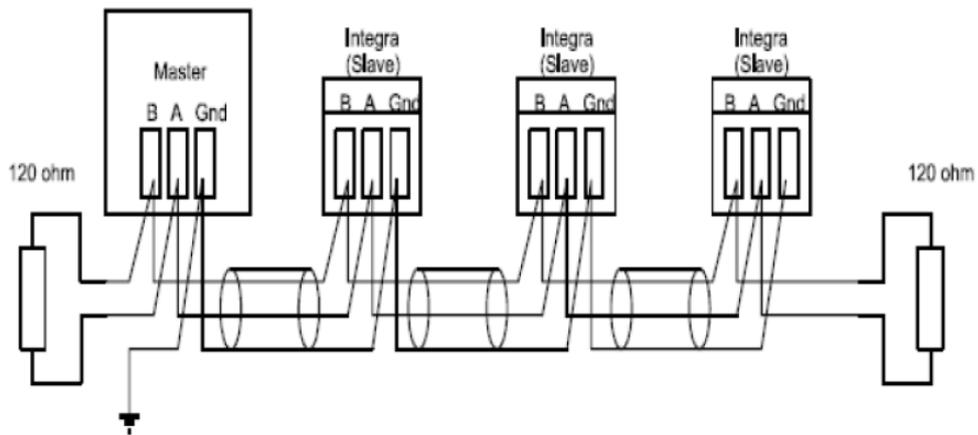


Figura 2.12: Rede de Comunicação RS-485 Mestre/Escravo.

Para viabilizar a implementação prática do padrão RS-485, vários fabricantes desenvolveram circuitos integrados *driver/receiver* para esta finalidade. Um *driver/receiver* RS-485 é um circuito integrado que converte sinais codificados no padrão TTL para o padrão RS-485. A Figura 2.13 mostra uma típica conexão multiponto utilizando o *driver/receiver* RS-485, fabricado pela empresa Maxim (MAXIM, 2003).

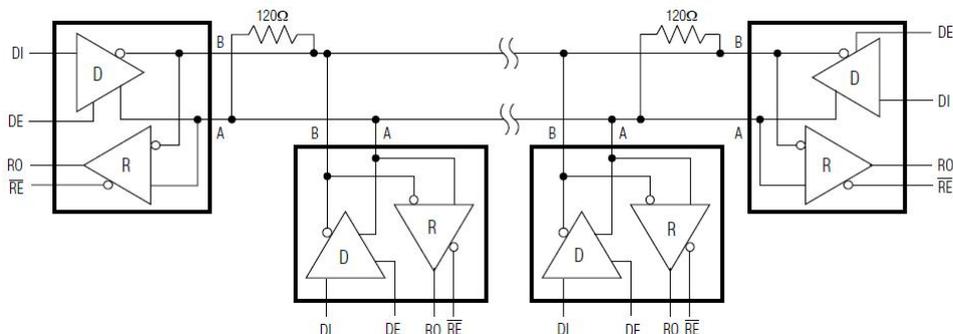


Figura 2.13: Conexão do *driver/receiver* em uma rede padrão RS-485 (MAXIM, 2003).

2.9. Modelo de Referência eOSI: Embedded Open Systems Interconnection

O modelo referência eOSI, proposto por Moraes (2009), foi inspirado no modelo referência OSI, definido pelo *International Organization for Standardization* (ISO), em 1984. O Modelo OSI foi desenvolvido para permitir a comunicação entre computadores ligados em rede, e define diretivas genéricas para a construção de redes de computadores (seja de curta, média ou longa distância), independente da tecnologia utilizada. A utilização do modelo OSI assume maior importância nas redes de telecomunicações. Nas redes de dados, a referência ao modelo ISO é encontrada com menos frequência, mas tem constituído uma ótima referência para o desenvolvimento de novas tecnologias (MARTINS, 2001). O modelo de referência OSI foi utilizado como base para o desenvolvimento do modelo referência eOSI.

O modelo referência eOSI está estruturado em camadas e tem como objetivo permitir que sistemas embarcados sejam desenvolvidos sob uma nova perspectiva que facilite a utilização de técnicas de tolerância a falhas. Diferente de uma visão componentizada, o modelo de referência eOSI permite que o projetista possa desenvolver um sistema embarcado obedecendo aos requisitos de suporte que cada camada exige (MORAIS, 2009). Segundo Moraes (2009) esse modelo de referência pode apresentar algumas vantagens, tais como: permitir a segmentação funcional para o desenvolvimento de um sistema tolerante a falhas; aumentar do desacoplamento entre os componentes do sistema; melhorar a visualização das técnicas a serem empregadas no sistema com base nos seus requisitos; estabelecer uma estrutura hierárquica para o fluxo de dados; o desenvolvimento de aplicações pode ser efetuado com base nas premissas de cada camada do modelo; orientar de forma sistemática o desenvolvimento de um sistema distribuído tolerante a falhas.

Tal qual o modelo de referência OSI, o modelo de referência eOSI é composto por um conjunto de camadas, as quais são orientadas à funcionalidade e a conceitos relacionados à tolerância a falhas. Ao contrário, o modelo de referência OSI é um modelo de referência orientado a protocolos e serviços. A Figura 2.14 mostra os modelos de referência OSI e eOSI, além das respectivas equivalências entre as camadas.

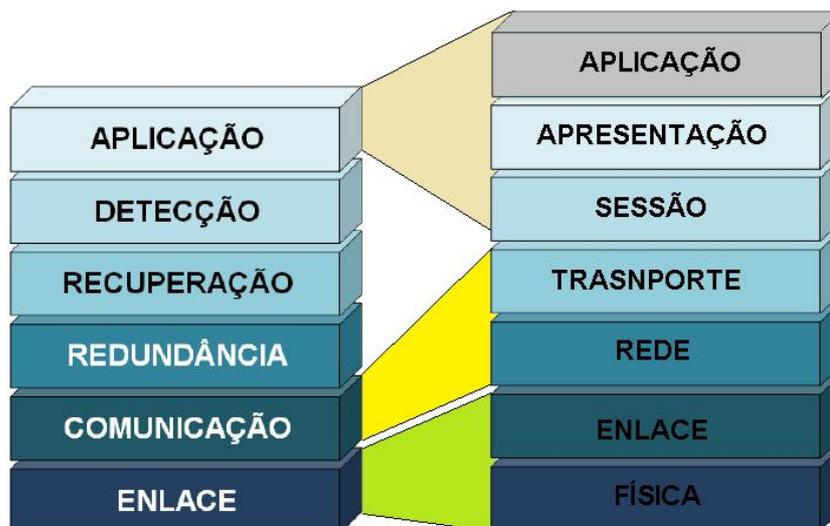


Figura 2.14: Equivalência dos modelos de referência eOSI e OSI.

podemos observar na figura que as camadas enlace e física do modelo de referência OSI correspondem à camada de enlace do modelo de referência eOSI. Da mesma forma, as camadas de transporte e de rede correspondem à camada de comunicação no modelo de referência eOSI. As funcionalidades da camada de Aplicação do modelo de referência eOSI podem ser equivalentes aos serviços fornecidos pelas camadas Aplicação, Apresentação e Sessão do modelo de referência OSI. As camadas de Detecção, Recuperação e Redundância não possuem um equivalente direto, o que não impede de algumas das suas funcionalidades sejam fornecidas por serviços de algumas das camadas do modelo de referência OSI.

2.9.1. Camada de Aplicação

A Camada de Aplicação do modelo referência eOSI é destinada ao desenvolvimento das tarefas relativas ao processo. Em tese, o desenvolvimento da Camada de Aplicação não precisa estar focado nas técnicas de detecção e recuperação de falhas. O mais importante é o fato de que o pacote de dados gerado pela camada de aplicação deve ser enviado à camada inferior.

2.9.2. Camada de Detecção

Esta camada destina-se a utilização ou ao desenvolvimento de componentes e recursos que são os responsáveis pela detecção de falhas. É responsabilidade do desenvolvedor definir as técnicas que serão utilizadas para a detecção de falhas. Uma falha pode ser definida como um

evento que faz o processamento desviar-se do seu curso normal de execução, e está necessariamente associada ao universo físico (MORAIS, 2009).

De um modo geral, os microcontroladores possuem uma série de recursos que podem ser utilizados para a detecção e recuperação de falhas, principalmente àquelas que se manifestam por conta de ruídos que entram no sistema por meio de distúrbios da tensão de alimentação, por EMI ou por falhas decorrentes do circuito gerador do sinal de *clock*. Dentre os recursos disponíveis nos microcontroladores destinados a tratar as falhas citadas, podemos destacar o *Watchdog Timer*, o *Brown-out Detect*, o *Fail Safe Clock Monitor* e *Power-up Timer*.

Além dos recursos nativos ao microcontrolador utilizados na recuperação de falhas, o projetista pode fazer uso de técnicas de tolerância a falhas (WEBER *et al*, 1990), (LAPRI, 1985) e (ANDERSON, 1981). A manifestação de uma falha faz o sistema atuar em não conformidade com as especificações do projeto. Neste caso, diz-se que ele se encontra em estado de erro (AVISIENS *et al*, 2004). Por exemplo, uma falha em uma interface de comunicação pode fazer o sistema não se comunicar corretamente. Neste caso, uma falha provocou um erro de comunicação. Sendo assim, um equipamento está com defeito quando um erro se manifesta decorrente de uma falha.

Johnson (1984) definiu três tipos de técnicas a serem implementadas para tentar manter o sistema em funcionamento em caso de falhas. São elas: Prevenção de Falha, Mascaramento de Falha e Tolerância a Falha. A Prevenção de Falhas pode ser atacada com técnicas de manutenção preventiva e preditiva, além da substituição de equipamentos antes que eles venham a apresentar falhas. O Mascaramento de Falhas é aplicado no sistema em funcionamento e visa não deixar o erro provocado pela falha se propagar para outras partes do sistema. Uma das técnicas utilizadas para o mascaramento de falhas é a utilização de dispositivos votadores. Dois componentes processam os dados em paralelo, enquanto um elemento votador decide qual dos componentes tem o seu processamento aproveitado.

A Tolerância a Falhas não tem o objetivo de eliminar o surgimento da falha ou a sua propagação, mas sim manter o sistema em um seguro estado de funcionamento. Quatro são os princípios utilizados na Tolerância a Falhas. São eles: a detecção, a localização, o

confinamento e a recuperação. A Figura 2.15 mostra o diagrama do fluxograma de Causa e Efeito proposto por Johnson (1984).

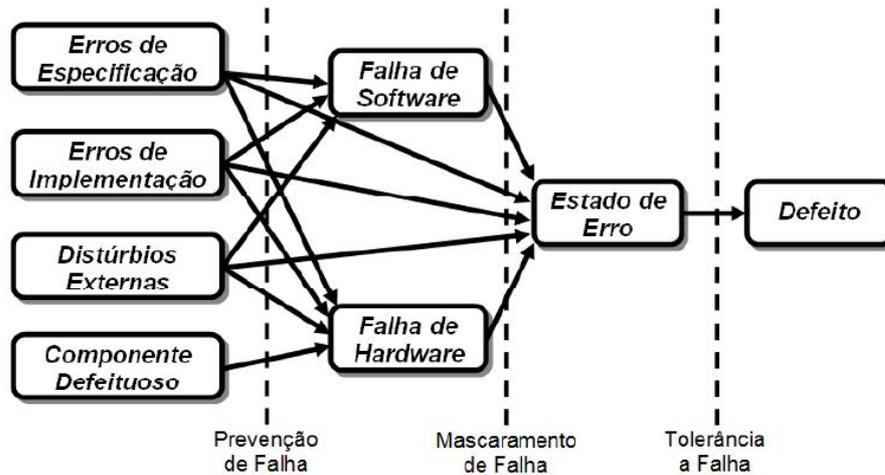


Figura 2.15: Fluxograma de Causa e Efeito de falhas (JOHNSON, 1984).

Observe que uma falha não necessariamente ocorre devido a um componente de *hardware* defeituoso. Um algoritmo mal estruturado ou um dado recebido fora do *range* pode provocar uma falha no sistema caso o *software* não trate com a devida atenção o problema.

Uma técnica que pode ser utilizada na Tolerância a Falhas é a redundância do *hardware*. Esta técnica, no entanto está relacionada ao aumento do custo financeiro do sistema devido a duplicidade de alguns componentes de *hardware*. Em alguns sistemas em que uma falha pode provocar a perda de vidas humanas pode ser justificado o uso desta técnica. Algumas técnicas de redundância de *hardware* são: Redundância Estática, Redundância Dinâmica e Redundância Híbrida. Detalhes sobre estas técnicas de redundância podem ser encontrados em (KOREN e KRISHNA, 2007).

A redundância de *software* também pode ser utilizada em sistemas tolerantes a falha. Dentre elas podemos destacar a Programação N-versões, Blocos de Recuperação, Pré Condições, Pós Condições e Asserções. Detalhes sobre estas técnicas de redundância podem ser encontrados em (ABD-ELBARR, 2007), (KOREN e KRISHNA, 2007), (PRADHAN, 1996) e (JALOTE, 1994).

2.9.3. Camada de Recuperação

A função desta camada é receber da camada de detecção a informação sobre a ocorrência de falhas e proceder a recuperação do sistema, não permitindo que a falha se propague para outras partes do sistema. Os mecanismos de recuperação do sistema são variados, cabendo ao projetista a escolha da implementação mais adequada.

No que diz respeito a sistemas embarcados desenvolvidos com microcontroladores, cerca de 70% das falhas no sistema ocorrem devido ao travamento do oscilador por causa de ruídos ou de EMI. De forma geral, os microcontroladores possuem ferramentas de recuperação em caso de paralisação do oscilador, citadas no item 2.5.3. Cabe ao projetista ativar tais ferramentas para o suporte à Camada de Recuperação. Sendo assim, tão importante quanto informar que a falha ocorreu, é importante, sempre que possível, a Camada de Detecção informar à Camada de Recuperação o tipo de falha para que esta última possa atuar na recuperação do sistema, impedindo que erros decorrentes da falhas se propaguem.

2.9.4. Camada de Redundância

Esta camada realiza o gerenciamento sobre o barramento conectado aos dispositivos redundantes. Ou seja, ela é a responsável pela troca de mensagens e pelo controle de acesso ao barramento de redundância. Esta camada implementa estratégias para que diferentes dispositivos possam efetuar processamento paralelo (balanceamento de carga) e também pode atuar como redundância para a técnica de Tolerância a Falhas. A complexidade envolvida na implementação desta camada faz com que ela se torne uma das mais críticas do modelo eOSI e, por esta razão, requer mais atenção. A camada de redundância pode possibilitar a criação de um sistema tolerante a falhas que pode efetuar o mascaramento de falhas, como também na detecção, confinamento e na recuperação de falhas (MORAIS, 2009).

O barramento de redundância deve ser independente do barramento de comunicação. Em outras palavras, o tráfego da rede de dados não deve ser afetado pelo barramento de redundância. Sendo assim, as mensagens trocadas pelos dispositivos redundantes através do

barramento de redundância não deve afetar o desempenho do sistema no que diz respeito a troca de informações entre os diversos dispositivos presentes em um sistema distribuído.

2.9.5. Camada de Comunicação

A principal função da camada de comunicação é fazer com que o dado chegue ao seu destinatário em segurança. Entre outras coisas, a Camada de Comunicação deve escolher um caminho a ser seguido pelos pacotes de mensagens. Essa escolha deve ser baseada no conhecimento do estado da rede e da quantidade de tráfego na rede, de forma a evitar sobrecargas e a consequente perda de dados.

Duas funcionalidades importantes da Camada de Comunicação dizem respeito ao roteamento e ao tratamento de colisão. O roteamento direciona a mensagem para que ela chegue ao seu destino garantindo uma comunicação fim a fim. A colisão deve ser evitada pois incorre na perda de dados. Normalmente uma colisão ocorre quando dois dispositivos iniciam uma transmissão ao mesmo instante. A topologia da rede pode contribuir de forma significativa para evitar colisão. Uma técnica muito utilizada em barramentos de comunicação utilizados em sistemas embarcados, nos quais o tráfego de dados é pequeno, é a configuração mestres/escravo. Neste tipo de configuração nunca haverá uma colisão, uma vez que o dispositivo mestre é sempre quem inicia a comunicação. A escolha do protocolo de comunicação a ser utilizado também é uma funcionalidade da Camada de Comunicação. O protocolo utilizado na comunicação entre os diferentes dispositivos da rede deve estabelecer conexões seguras e confiáveis. Dentre os protocolos utilizados em sistemas embarcados podemos citar o TCP/IP, o ModBus, o CAN e o ZigBee.

Não é uma funcionalidade da Camada de Comunicação a utilização de técnicas tolerantes a falha. Esta é uma atribuição das camadas superiores. A Camada de Comunicação deve garantir que os dados transmitidos cheguem aos seus destinos.

2.9.6 Camada de Enlace

A principal funcionalidade desta é o encapsulamento das mensagens em *frames* e a modulação dos bits para serem introduzidos no meio de transmissão. Este último pode ser cabo com par

metálico trançado, cabo coaxial, fibra óptica, *wireless* etc. A Camada de Enlace deve transformar o meio de transmissão não seguro em um meio seguro, garantindo que as mensagens cheguem ao seu destino sem que haja perda de dados. Técnicas comuns utilizadas hoje em dia para detectar erros de transmissão e, conseqüentemente, garantir a não perda de dados na transmissão são o *checksum* e o *Cyclic Redundancy Check (CRC)*. Assim como da Camada de Comunicação, não é uma funcionalidade da Camada de Enlace a utilização de técnicas de Tolerância a Falhas. Esta é uma atribuição das camadas superiores.

Convém ressaltar que embora o modelo de referência eOSI seja uma metodologia robusta para o desenvolvimento de sistemas embarcados tolerantes a falha, é uma atribuição do projetista do sistema definir quais as camadas serão implementadas no projeto, quais as técnicas de Tolerância a Falhas serão utilizadas e como serão implementadas.

2.10. Protocolo ModBus

O protocolo de comunicação ModBus foi desenvolvido no final da década de 1970 pela empresa Modicon para ser utilizado na troca de informações entre equipamentos industriais conectados em rede. Em 2004, o controle do protocolo ModBus foi transferido para a ModBus Organization. Desde então, a utilização do protocolo ModBus em comunicação de dados é livre e sem taxa de licenciamento.

O Protocolo ModBus é um protocolo de comunicação baseado em mensagens, posicionado na camada de aplicação do modelo OSI, que provê uma comunicação cliente/servidor entre dispositivos conectados em diferentes tipos de barramentos e de rede (MODBUS O, 2006).

Devido ao fato do protocolo ModBus ser seguro, de fácil entendimento e de simples implementação, além de ser um protocolo aberto, fez com que ele tenha se tornado um dos protocolos mais utilizados em aplicações industriais. Hoje em dia, no entanto, é comum ver o protocolo ModBus sendo utilizado em aplicações não industriais.

A comunicação ModBus é implementada por meio do envio de mensagens do tipo pergunta/resposta, e está baseada em um conjunto de funções específicas de leitura e de escrita

em memória. Tendo uma configuração mestre/escravo, é o mestre quem sempre inicia uma comunicação ModBus. Cada mensagem pode ser endereçada a um único dispositivo ou a todos os dispositivos conectados à rede, esta última chamada de comunicação *boadcast*.

O protocolo ModBus permite que existam até 247 dispositivos físicos conectados na rede. Uma comunicação tem início quando o dispositivo mestre envia uma mensagem ao dispositivo escravo endereçado. O dispositivo endereçado deve processar a mensagem e responder ao dispositivo mestre a requisição efetuada. Se a mensagem for do tipo *broadcast* ela não deve ser respondida. A Figura 2.16 mostra a estrutura da pilha de comunicação ModBus.

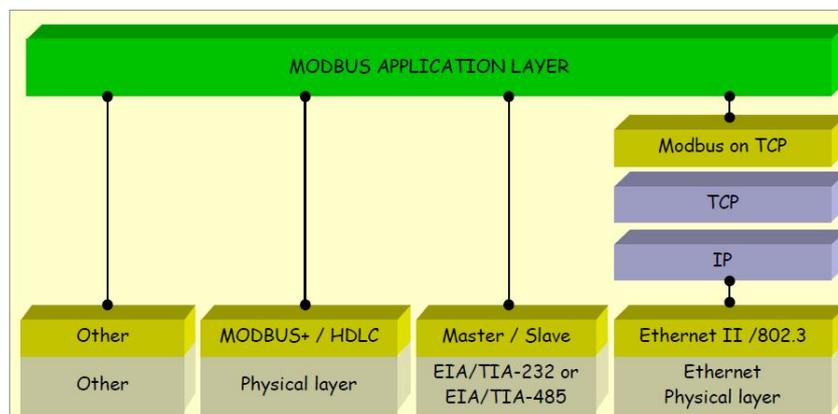


Figura 2.16: Pilha de comunicação ModBus (MODBUS O, 2006).

Os padrões RS-232 e RS-485 são os mais utilizados meio de transmissão para as mensagens ModBus, embora ele também possa ser encapsulados em *frames* TCP/IP, o qual faz uso do padrão *ethernet* para a transmissão dos dados.

O padrão ModBus+ define um conjunto de protocolos proprietários para a transmissão de dados em alta velocidade. Ele envolve uma série de aperfeiçoamentos implementados na rede ModBus para permitir a flexibilidade e a otimização da rede. Dentre os aperfeiçoamentos podemos citar o uso de *bridges* para conectar dois nós distintos da rede, o uso de *token*, com o qual cada dispositivo da rede pode transmitir uma vez que ele esteja de posse do *token*. Outros recursos do ModBus+ são a capacidade de roteamento, diagnóstico e consistência dos dados. Os direitos de uso do ModBus+, no entanto, estão sob o domínio da empresa Schneider Electric, que precisa fornecer licença para a sua comercialização (MODBUS P, 2004).

O protocolo ModBus define uma estrutura chamada Unidade de Protocolo de Dados (PDU) com a função a ser executada e os dados a serem transmitidos. Alguns campos adicionais são incluídos no PDU, os quais dão origem a um *frame* chamado de Unidade de Aplicação de Dados (ADU). A Figura 2.17 mostra o formato genérico de um *frame* ModBus.

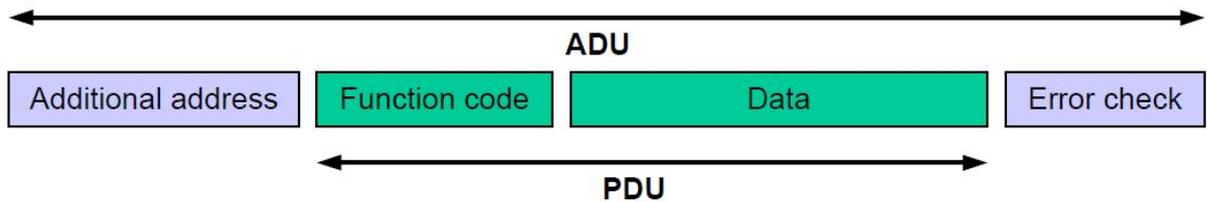


Figura 2.17: Pacote genérico ModBus (MODBUS O, 2006).

A mensagem ModBus é encapsulada em um *frame* formado pelos campos endereço, função, dados e o campo *error check*. No campo endereço é inserido o endereço lógico do dispositivo escravo com o qual o dispositivo mestre deseja se comunicar. O campo seguinte contém o código da função que é executada. O próximo campo encapsula os dados e, finalmente, o campo *error check* é utilizado para verificar se houve algum erro de transmissão nos campos anteriores.

A especificação do protocolo ModBus define uma tabela de funções suportadas pelo protocolo. A faixa de valores válida para a função vai de 1 - 255. As funções suportadas pelo protocolo ModBus estão divididas em três categorias. São elas as funções pública, as funções definidas pelo usuário e as funções reservadas. As funções públicas são funções definidas pelo protocolo ModBus. As funções definidas pelo usuário podem conter funções não especificadas no protocolo, enquanto as funções reservadas são destinadas a utilização de empresas em produtos específicos, e não estão disponíveis para uso público. A Figura 2.18 mostra as funções ModBus separadas pelas categorias pública e definidas pelo usuário.

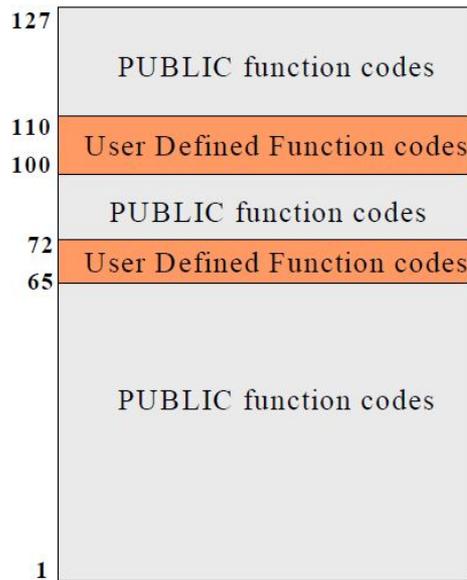


Figura 2.18: Categorias das funções ModBus (MODBUS O, 2006).

A Figura 2.18 mostra as funções públicas suportadas pelo protocolo ModBus. Essas funções são validadas pelo ModBus Organization e possuem teste de conformidade. Isso significa que se um sistema tiver estas funções implementadas ele pode se comunicar por meio do protocolo ModBus com outros dispositivos conectados a sua rede.

A Figura 2.19 mostra que as funções públicas suportadas pelo protocolo ModBus estão divididas em grupos. São eles: Acesso a bits, acesso a registradores, acesso a registros e funções de diagnóstico. Para implementar essas funções, o protocolo ModBus baseia o seu modelo de dados em uma série de tabelas com características distintas. São elas: *Input Discrete*, *Coils*, *Input Registers* e *Holding Registers*. O grupo *Input Discrete*, somente leitura de bits, pode ser provido por um sistema de I/O. O grupo *Coils*, leitura e escrita de bits, pode ser alterado por uma aplicação do programa. O grupo *Input Registers*, somente leitura em registradores, pode ser provido por um sistema I/O. O grupo *Holding Registers*, leitura e escrita em registradores, pode ser provido por uma aplicação do programa.

				Function Codes		
				code	Sub code	(hex)
Data Access	Bit access	Physical Discrete Inputs	Read Discrete Inputs	02		02
		Internal Bits Or Physical coils	Read Coils	01		01
			Write Single Coil	05		05
			Write Multiple Coils	15		0F
	16 bits access	Physical Input Registers	Read Input Register	04		04
			Read Holding Registers	03		03
		Internal Registers Or Physical Output Registers	Write Single Register	06		06
			Write Multiple Registers	16		10
			Read/Write Multiple Registers	23		17
			Mask Write Register	22		16
	File record access		Read FIFO queue	24		18
			Read File record	20		14
	Diagnostics		Write File record	21		15
			Read Exception status	07		07
			Diagnostic	08	00-18,20	08
			Get Com event counter	11		0B
			Get Com Event Log	12		0C
		Report Slave ID	17		11	
		Read device Identification	43	14	2B	

Figura 2.19: Funções públicas do protocolo ModBus (MODBUS O, 2006).

Para cada um dos grupos do modelo de dados do protocolo ModBus é possível acessar até 65.535 itens de dados. As operações de leitura e escrita desses itens de dados foram designadas para acessar múltiplos e consecutivos itens de dados até o limite especificado, o qual depende da função executada (MODBUS O, 2006). Em outras palavras, a troca de mensagens via protocolo ModBus consiste em o dispositivo mestre ler e escrever em endereços de memória de cada dispositivo escravo. A faixa de endereços de memória acessada pelo protocolo ModBus pode ser agrupada por blocos, com cada bloco sendo utilizado por um grupo de funções, como mostra a Figura 2.20.

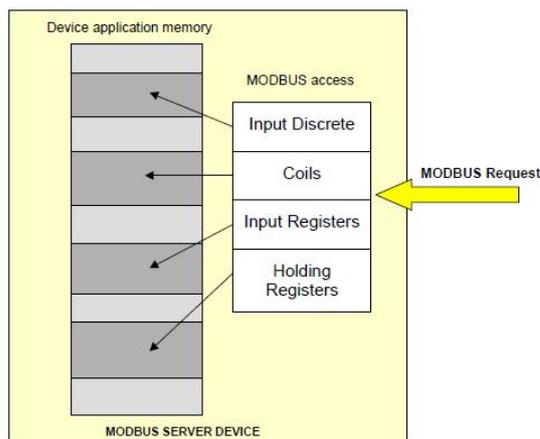


Figura 2.20: Modelo de dados separado por blocos (MODBUS O, 2006).

O modelo apresentado na Figura 2.20 é uma das possibilidades de organização da memória em uma comunicação ModBus. Outras possibilidades de organização podem ser escolhidas pelo projetista do sistema. O projetista da rede deve especificar as respectivas faixas de endereços utilizadas por cada item do modelo de dados do protocolo ModBus, os quais devem estar associados a itens reais do sistema.

2.10.1. Modos de Transmissão ModBus sobre os Padrões RS-232/RS485

Quando o protocolo ModBus é executado sobre os padrões RS-232 ou RS485, existem dois modos distintos de transmissão. São eles o ASCII e o RTU. O modo de transmissão define como os dados são empacotados na mensagem, e também como é efetuada a verificação de erros. Além do modo de transmissão, é preciso definir os parâmetros da transmissão, como a taxa de transferência (*baud rate*), o número de bits de dados, o bit de paridade e o tamanho do *stop bit*. Esses parâmetros são utilizados quando é utilizado o protocolo de comunicação serial UART, o mais comum em transmissões Modbus que utilizam os padrões RS-232 ou RS485.

2.10.2. Modo de Transmissão ASCII

Nesse modo de transmissão, toda mensagem inicia com o caractere dois pontos ":". O caractere dois pontos é o delimitador de início da mensagem. Após o caractere de delimitação de início da mensagem, é enviado o segundo campo da mensagem com o endereço do dispositivo escravo com o qual o dispositivo mestre irá se comunicar. O próximo campo é

destinado a função que é executada. Em seguida são encapsulados os dados, que podem ser de no máximo 256 caracteres. O campo seguinte ao campo de dados é reservado para o *Longitudinal Redundancy Check* (LCR). Esse campo é utilizado para a verificação da integridade dos dados, e é composto por dois *bytes*. Os caracteres *Carriage Return* (CR) e *Line Feed* (LF) são inseridos no final da mensagem indicando que ela chegou ao fim. A Figura 2.21 mostra a estrutura da mensagem enviada à rede pelo dispositivo mestre no modo de transmissão ASCII. Os caracteres delimitadores de início e fim de mensagem são representados no código ASCII. Sendo assim, o valor 3AH representa o caractere dois pontos. O valor 0DH representa o caractere CR e o valor 0AH representa o caractere LF.

Início de frame	Endereço do Escravo		Função ModBus	Dados para o Escravo	Checksum		Fim de frame	
	char +	char -			LRC +	LRC -	0Dh	0Ah
3Ah			2 chars	N chars				

Figura 2.21: Estrutura do *frame* ModBus ASCII.

O campo endereço é reservado para o endereço do dispositivo escravo com o qual o dispositivo mestre vai se comunicar. A faixa de endereços válida para o protocolo ModBus vai de 01H - F7H. O endereço zero 00H é utilizado na comunicação *broadcast*. Suponha que o endereço do dispositivo escravo seja 69. Convertendo o valor 69 para o sistema hexadecimal temos o valor 45H. No modo de transmissão ASCII o valor inserido no campo Endereço é 34H e 35H, que correspondem respectivamente aos valores 4 e 5 representados no código ASCII.

No campo função é inserida a função que é executada pelo dispositivo escravo. Suponha, por exemplo, que a função a ser executada seja a função 3 (03h), leitura em um bloco de registradores. Neste caso, o valor inserido no campo função no modo de transmissão ASCII é 30H e 33H, respectivamente os valores 0 e 3 representados no código ASCII. Se a função a ser executada for a 16, o valor a ser inserido no campo função é 31H e 36H, respectivamente os valores 1 e 6 representados no código ASCII.

O protocolo ModBus utiliza a representação *big endian* para a codificação de endereços e dados no *frame*. Isso significa que o *byte* mais significativo é inserido primeiro no campo.

A quantidade de dados da mensagem ModBus depende da função que é executada. Quando a função a ser executada é uma função de leitura em um bloco de registradores, por exemplo, é necessário informar no campo dados o endereço inicial do bloco e a quantidade de registradores que é lida. Da mesma forma, em uma função de escrita em um bloco de registradores, é necessário informar o endereço inicial do bloco e o número de registradores que é escrito. A função 03 pode ser utilizada para a leitura de um bloco de registradores, enquanto a função 16 pode ser utilizada para a escrita em um bloco de registradores. Com estas duas funções é possível implementar uma comunicação ModBus entre dois ou mais dispositivos ligados em rede. Este é o caso do desenvolvimento prático experimental desta dissertação.

Quando o modo de transmissão é o ASCII, a integridade dos dados é verificada pelo cálculo do *checksum*, o qual é obtido por meio da técnica LCR. O *checksum* é obtido a partir da soma de todos os campos da mensagem anteriores ao campo LCR, exceto o caractere delimitador de início da mensagem. Do resultado da soma de cada *byte*, somente são aproveitados os oito bits menos significativos, ou seja, bits de estouro são descartados do resultado. O LCR, no entanto, não é uma técnica muito segura na verificação da integridade dos dados, uma vez que ele analisa apenas os valores dos campos e não os valores que são efetivamente enviados no *frame*. Por exemplo, se o endereço do dispositivo escravo é o 69, os valores inseridos no campo de endereço da mensagem são 34H e 35H. No cálculo do *checksum*, no entanto, é utilizado o valor do endereço, 69, e não o valor enviado na mensagem.

A seguir é mostrado um *frame* ModBus ASCII com os seus respectivos campos.

- Endereço do dispositivo escravo: 69 (45H) → 34H:35H;
- Função ModBus: 3 (03H) → 30H:33H;
- Endereço inicial do bloco de registradores a ser lido: 11 (0BH) → 000AH = 30H:30H:30H:41H;
- Número total de registradores a ser lido: 1 (01H) → 0001H = 30H:30H:30H:31H;
- *Checksum* (LCR) gerado: 173 (ADH) → 41H:44H.

Para os parâmetros acima, o *frame* enviado para a rede é mostrado na Figura 2.22.

3AH	34H	35H	30H	33H	30H	30H	30H	41H	30H	30H	30H	31H	41H	44H	0DH	0AH
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Figura 2.22: *Frame* ModBus ASCII para a função 3.

2.10.3. Modo de Transmissão RTU

Este modo de transmissão não possui caracteres delimitadores de início e fim de *frame*. Após o envio de uma mensagem é preciso esperar um período equivalente a 3,5 vezes o tempo correspondente ao tempo de transmissão de uma palavra de dados, chamado intervalo *silent*. Por exemplo, suponhamos que a taxa de transmissão seja de 19.200 bps. Para esta taxa de transferência, o tempo utilizado para transmitir uma palavra de dados 8 bits de dados, sem paridade e dois *stop bit* é de 572,9 μ s. Sendo assim, o intervalo *silent* é de $3,5 \times 572,9\mu = 2.005$ ms. Após o intervalo *silent*, é possível iniciar uma nova transmissão. O intervalo *silent* deve ser gerado antes e após o envio de cada mensagem ModBus.

No modo de transmissão RTU a mensagem deve ser enviada de maneira contínua. Sendo assim, se for detectado um intervalo sem transmissão maior que 1,5 vezes o tempo correspondente a palavra de dados sem que a mensagem tenha sido totalmente recebida, o dispositivo escravo descarta a mensagem atual e espera que o próximo caractere seja o campo de Endereço da próxima mensagem. De modo similar, se uma mensagem for enviada em um intervalo menor que o intervalo *silent*, o dispositivo escravo assume que a mensagem atual é continuação da mensagem anterior. A mensagem é então descartada, uma vez que o cálculo do *checksum* (CRC) não é validado. A Figura 2.23 mostra um típico *frame* ModBus RTU.

Endereço do Escravo	Função ModBus	Dados para o Escravo	Checksum	
1 char	1 char	N chars	CRC -	CRC +

Figura 2.23: *Frame* genérico ModBus RTU.

O campo endereço é reservado para o endereço do dispositivo escravo com o qual o dispositivo mestre vai se comunicar. Suponha que o endereço do dispositivo escravo seja 69.

Convertendo o valor 69 para o sistema hexadecimal temos o valor 45H, valor inserido campo Endereço.

No campo função é inserida a função que é executada pelo dispositivo escravo. Suponha, por exemplo, que a função a ser executada seja a função 3. Neste caso, o valor inserido no campo Função é o 03H.

As informações contidas campo dados depende da função a ser executada. As mesmas considerações feitas para o modo de transmissão ASCII, são válidas para o modo de transmissão RTU. Se for executada uma função de leitura em bloco de registradores é preciso informa o endereço inicial do bloco e o número de registradores que é lido. O mesmo procedimento de ser adotado para a função de escrita. Neste último caso, deve ser inserido no campo dados os valores a serem escritos nos respectivos endereços do bloco a ser escrito.

Quando o modo de transmissão é o RTU, A integridade dos dados é verificada pelo cálculo do *checksum*, o qual é obtido por meio da técnica CRC. O CRC é obtido a partir de um cálculo matemático que envolve todos os campos da mensagem anteriores ao campo CRC. O resultado é representado por dois bytes, os quais são inseridos no campo *checksum*.

O CRC é uma técnica de verificação da integridade dos dados mais segura que o LCR, uma vez que todos os bytes da mensagem são envolvidos no cálculo do CRC. Por exemplo, se o resultado obtido no cálculo do CRC for o valor 19.627 (4CABH). O valor inserido no campo *Checksum* é ABH:4CH. Veja que na composição do CRC RTU, o *byte* menos significativo é inserido primeiro no campo *Checksum*. O cálculo do CRC pode ser feito da seguinte forma:

1. carrega-se o registrador CRC com o valor FFFFH;
2. submete-se o caractere da mensagem uma lógica xor com os 8 bits menos significativos do registrador CRC, retornando o resultado no registrador CRC;
3. desloca-se o conteúdo do registrador CRC 1 bit para a direita inserindo zero '0' no seu bit mais significativo;
4. examina-se o bit menos significativo do registrador CRC e:
 - se bit = 0, repete o processo a partir do item 3.

- se bit = 1, submete-se o conteúdo do registrador CRC a uma lógica xor com a constante A001H, retornando o resultado no registrador CRC e, em seguida, repete-se o processo a partir do item 3.
- 5. repete-se os itens 3 e 4 até que tenham ocorridos 8 deslocamentos;
- 6. repete-se os itens 2 -5 para o próximo caractere da mensagem até que todos os caracteres tenham sido analisados;
- 7. o valor final do registrador CRC é o valor do campo *Checksum*;
- 8. o *byte* menos significativo do CRC é o primeiro *byte* do campo *Checksum*.

A seguir é mostrado um *frame* ModBus RTU com os seus respectivos campos da mensagem.

- Endereço do dispositivo escravo: 69 (45H) → 45H;
- Função ModBus: 3 (03H) → 03H;
- Endereço inicial do bloco de registradores a ser lido: 11 (0BH) → 000AH = 00H:0AH;
- Número total de registradores a ser lido: 1 (01H) → 0001H = 00H:01H;
- *Checksum* (CRC) gerado: 19.627 (4CABH) → ABH:4CH.

Para os parâmetros acima, o *frame* enviado para a rede é mostrado na Figura 2.24.

45H	03H	00H	0AH	00H	01H	ABH	4CH
-----	-----	-----	-----	-----	-----	-----	-----

Figura 2.24: *Frame* ModBus RTU para a função 3.

2.11. Sistemas Móveis Celulares.

A primeira geração de sistemas celulares era analógica e utilizavam a modulação em frequência (FM). A primeira geração era chamada de 1G. Devido a novidade do novo sistema, poucos esforços foram destinados para a padronização nos diversos países em que ele foi implantado. O Brasil seguiu o padrão proposto pelos Estados Unidos, o padrão *Advanced Mobile Phone Systems* (AMPS). O padrão AMPS opera na faixa de 800 MHz e ocupa uma banda de 50

MHz. Além disso, o padrão AMPS utiliza o acesso múltiplo por divisão de frequência (FDMA) com largura de banda de 30 kHz por canal.

O aumento da capacidade demandada pelo mercado norte americano e a necessidade de padronização exigida pelo mercado europeu levou ao desenvolvimento de vários novos padrões de sistemas celulares, chamados de 2G. Os EUA desenvolveram os padrões IS-54 (AMPS digital), IS-136 (TDMA digital) e IS-95 (CDMA digital). O esforço de uniformização na Europa criou o padrão GSM (*Global System for Mobile communications*), principal padrão de segunda geração adotado. Todos os padrões 2G desenvolvidos são incompatíveis entre si. Informações detalhadas sobre esses padrões podem ser encontradas em Sverzut (2011). O padrão GSM foi adotado no Brasil em 2002. Nele a largura de banda de cada canal é de 200 kHz e a faixa de operação é 1,8 MHz.

O padrão GSM, no que diz respeito aos serviços oferecidos, passou por três fases de evolução. São elas: Fase 1, Fase 2 e Fase 2⁺. Os serviços oferecidos pela Fase 1 são: telefonia, chamadas de emergência, SMS, transmissão de dados no modo assíncrono, transmissão de dados no modo síncrono, transmissão de pacotes assíncronos. Os serviços oferecidos pela Fase 2 são: Telesserviços, voz de meia taxa (*half rate*), melhoria nos serviços SMS, serviços de dados, pacotes com transmissão síncrona e dedicada a taxas entre 2,4 e 14,4 kbps, identificador de chamadas, restrição de chamadas por números, chamada em espera, teleconferência e grupo de usuários, entre outros. A Fase 2⁺ introduziu o serviço de dados por pacotes em altas taxas de transmissão (*General Packet Radio Service - GPRS*). A Figura 2.25 mostra as fases de evolução da tecnologia GSM.

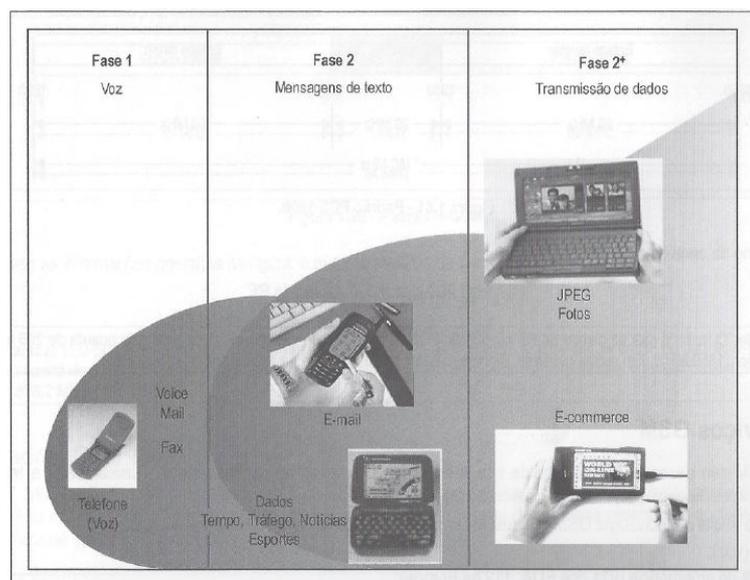


Figura 2.25: Fases da Tecnologia GSM (SVERZUT, 2011).

Os sistemas móveis celulares 1G são puramente analógicos, enquanto os sistemas móveis 2G são digitais, mas foram projetados para a transmitir voz e dados com baixa taxa de transferência.

Com a evolução da internet, os usuários de sistemas móveis começaram demandar serviços mais sofisticados, como trocar email, acessar páginas de internet, ouvir rádio e assistir TV. Essa necessidade fez com que os sistemas móveis celulares entrassem em um estado constante de evolução. Já estão disponíveis comercialmente os sistemas 2,5G e 2.75G, desenvolvidos para suportar a troca de mensagens e acesso à páginas da *Web*. No Brasil, o sistema 3G já está em funcionamento e o sistema 4G estava sendo testado na ocasião da defesa desta dissertação.

Com o objetivo de padronizar os sistemas móveis, a *International Communication Union* (UIT) publicou o padrão IMT-2000 (*International Mobile Telephone 2000*). A partir de então, os fabricantes começaram a propor soluções para atingir as demandas do padrão IMT-2000. O padrão IMT-2000 especifica seis categorias nas quais o usuário pode estar enquadrado. São elas: estacionário (0 km/h), pedestre (até 10 km/h), veicular (até 100 km/h), aeronáutico (até 1.500 km/h) e satélite (até 27.000 km/h).

Quanto aos tipos de serviços, o IMT-2000 especifica sete categorias. São elas: voz, áudio, texto, vídeo, imagem, sinalização e dados. Seguindo as especificações do padrão IMT-2000, diversas empresas e órgãos reguladores de diversas partes do mundo submeteram a UTI propostas para sistemas móveis 3G. São eles: DETC (*Digital Enhanced Cordless Telecommunications*), UWC-136 (*Universal Wireless Communications*), TD-SCDMA (*Time division Synchronous CDMA*), WCDMA (*Wideband CDMA*), CDMA II (*Asynchronous DS-CDMA*), UMTS (*Universal Mobile Telecommunications System*), NA-WCDMA (*North American - Wideband CDMA*), CDMA-2000 (*Wideband CDMA*), CDMA I (*Multiband Synchronous DS-CDMA*), SAT-CDMA (*Satellite CDMA*), SWCDMA (*Satellite Wideband CDMA*), SW-CTDMA (*Satellite Wideband Hybrid CDMA/Y=TDMA*), ICO-RTT (*ICO Radio Transmission Technology*). No Brasil estão disponíveis sistemas móveis 3G CDMA2000 e WCDMA (SVERZUT, 2011).

Na ocasião da defesa desta dissertação, o sistema 4G já estava em funcionamento em alguns lugares do mundo como, por exemplo, Coreia do Sul, Estados Unidos. De acordo com a avaliação de especialistas em redes de telecomunicações, as rede 4G são formadas, em grande parte, pelo padrão LTE (*Long Term Evolution*).

O padrão LTE foi especificado pela organização 3GPP (*3rd Generation Partnership Project*) para melhorar o desempenho das redes WCDMA/HSPA, adotada no Brasil, com vistas a aumentar as taxas de transferência de dados dos sistemas 3G. O padrão LTE permitiu o desenvolvimento de uma nova geração de redes de telecomunicações móveis (*Next Generation Mobile Network - NGMN*), com capacidade de transmissão de até 100 Mbps no enlace direto e 50 Mbps no enlace reverso. Uma das características das redes NGMN, que inclui o 4G, é a utilização do protocolo IP, tanto para a transmissão de voz quanto para a transmissão de dados. De acordo com a Organização 3GPP, a implementação da rede NGMN é feita pelo subsistema multimídia IP (*IP Multimedias Subsystem - IMS*). O IMS define uma arquitetura genérica para o desenvolvimento de redes de telecomunicações capazes de oferecer serviços multimídia e serviços de voz sobre IP (VoIP), permitindo a convergência entre as redes de comutação de circuitos, pacotes, TV a cabo e *wireless* (SVERZUT, 2011).

2.11.1. Arquitetura da Rede GSM

A rede GSM é formada por interfaces abertas e padronizadas, as quais permitem às operadoras de telefonia celular combinar componentes de diferentes fabricantes, tendo como resultado uma arquitetura bastante flexível. A Figura 2.26 mostra os principais componentes que compõem a arquitetura de rede GSM.

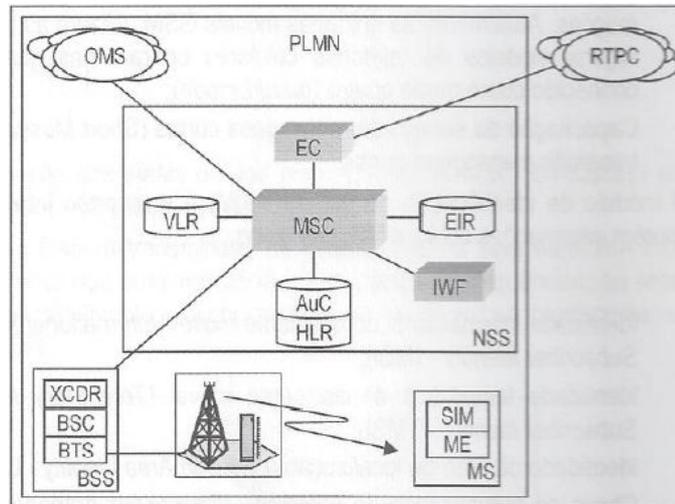


Figura 2.26: Componentes da Arquitetura GSM (SVERZUT, 2011).

A estação móvel (*Mobile Station - MS*) é a responsável pela conexão do assinante com a rede GSM. Um MS pode ser um telefone celular, um *tablet*, um computador ou qualquer outro dispositivo que possa se conectar a uma rede GSM para a transmissão de voz e dados. O sistema de estação base (*Base Station System - BSS*) faz a comunicação via rádio frequência com o MS para a troca de informações. O sistema de comutação da rede (*Network Switching System - NSS*) é a responsável pela conexão da rede GSM com a rede pública (RTPC), pelo gerenciamento da base de dados e pelo processamento das informações pelas interfaces. O sistema de manutenção e operação (*Operation and Maintenance System - OMS*) é responsável pela administração, operação e manutenção dos grupos de componentes.

A estação Móvel (MS) é composta por dois componentes. São eles o equipamento móvel (*Mobile Equipment - ME*) e o módulo de identidade do assinante (*Subscriber Identity Module - SIM*).

Cada ME possui um ID, chamado identidade internacional do assinante (*International Mobile Equipment Identity - IMEI*) o qual identifica o assinante no sistema.

Existem três tipos de ME. São eles: veicular, estação móvel portátil e estação móvel. Tanto a estação móvel portátil quanto a estação móvel podem ser transportadas enquanto estiverem em operação, sendo o telefone celular classificado como estação móvel. As três classes definidas para um ME possuem os seus próprios tipos de serviços e se diferenciam na classe de potência de RF, a qual define a potência máxima que o MS pode transmitir. O processamento e a tarifação das chamadas são realizadas a partir de informações contidas no cartão SIM.

O cartão SIM é o módulo de identificação do usuário. Ele é um *chip* de memória do tipo *smart card* que contém informações sobre a MS, tais como: identidade internacional do assinante móvel (*International Mobile Subscriber Identity - IMSI*); identidade temporária do assinante móvel (*Temporary Mobile Subscriber Identity - TMSI*); identidade da área de localização (*Location Area Identity - LAI*); chave de autenticação do assinante (*Subscriber Authentication Key - ki*); número internacional ISDN (*Integrated Service Digital Network*) da estação móvel (*Mobile Station Integrated Services Digital Network - MSISDN*). A tarifação das chamadas é efetuada a partir de informações contidas no cartão SIM e não no terminal móvel.

O Sistema de Estação Base (BSS) é constituído pelos seguintes componentes: estação transceptora base (*Base Transceiver Station - BTS*), controlador de estação base (*Base Station Controller - BSC*) e transcodificador (*Transcoder - XCDR*).

A BSC é responsável por um grupo de BTS, as quais executam operações recebidas da BSS. O BSC incorpora uma matriz de comutação digital, usada para conectar os canais de RF da interface aérea com os circuitos terrestres proveniente da central de comutação celular (*Mobile Services Switching Center - MSC*). A matriz de comutação digital permite a BSC realizar *handovers* entre os canais de RF sob o seu controle sem o envolvimento do MSC.

A BTS provê as conexões da interface aérea com a MS. Os elementos que formam a BTS são o *hardware* de radio frequência e as antenas. O BTS juntamente com o BSC realizam funções de gerência dos canais de tráfego e controle da interface aérea.

O XCDR é responsável pela conversão dos canais de voz provenientes da MSC (64 kbps) no padrão especificado pela norma GSM, utilizado na transmissão sobre a interface aérea. O XCDR converte amostras de 64 kbps nas taxas especificadas pelo GSM (16 kbps) para a comunicação entre BSS e MS. O XCDR também converte amostras de 16 kbps em 64 kbps para a comunicação entre BSS e MSC. O padrão GSM especifica dois algoritmos para esta conversão. São eles: algoritmo de taxa plena (*Full Rate*) e algoritmo de taxa plena melhorado (*Enhanced Full Rate*). Mais detalhes sobre estes algoritmos pode ser obtido em Sverzut (2011).

O Sistema de Comutação de Rede (NSS) é responsável pelas funções de comutação, controle e gerenciamento da mobilidade e da base de dados dos assinantes, a NSS é composta pelos seguintes componentes: central de comutação móvel (*Mobile Services Switching Center - MSC*), registro de localização local, (*Home Location Register - HLR*), Centro de autenticação (*Authentication Center - AuC*), Registro de localização de visitante (*Visitor Location Register - VLR*), registro de identidade de equipamento (*Equipment Identity Register - EIR*), função de interfuncionamento (*Interworking Function - IWF*) e supressor de eco (*Echo Canceler - EC*).

A MSC é responsável pelas seguintes funções: processamento de chamadas; orientação, manutenção e supervisão; interfuncionamento e tarifação. O Interfuncionamento é responsável pelo gerenciamento das interfaces entre a rede GSM e as demais redes como, por exemplo, a Rede Telefônica Pública Comutada (RTPC) e a Rede Digital de Serviços Interligados (RDSI). Ela é o elemento central do sistema de comutação da rede, como pode ser observado na Figura 2.26.

O HLR é responsável pela administração e controle da base de dados dos assinantes locais. Toda alteração nas características do perfil de um assinante é controlada e mantida pelo HLR. Os principais dados de um perfil de assinante são: identidade internacional do assinante móvel; localização corrente do assinante no VLR; serviços suplementares associados ao assinante (por exemplo, número de transferência programado); informações sobre serviços suplementares, estado do assinante (registrado ou não registrado) e chave de autenticação.

O VLR é responsável pela manutenção da cópia dos principais dados existentes no perfil do assinante armazenados no HLR como, por exemplo: estado da estação móvel (livre, ocupado, não responde); identidade da área de localização (*Location Area Identity* - LAI); identidade temporária do assinante móvel (*Temporary mobile Subscriber Identity* - TMSI); número da estação móvel visitante (*Mobile Station Roaming Number* - MSRN).

A IMSI é o número que identifica a MS internamente na rede GSM. Sua estrutura inclui o código do país, código da área móvel e o número que identifica o assinante móvel na rede móvel pública terrestre (*Public Land Mobile Network* - PLMN). O PLMN é o nome dado à arquitetura GSM de uma determinada operadora. O número que identifica o assinante na PLMN é, por exemplo, o número do celular.

A LAI é o nome dado a um conjunto de células da PLMN, agrupadas em uma área geográfica. Tipicamente, uma LAI é composta por um conjunto de trinta células. Quando um assinante se move de uma LAI para outra, o VLR atualiza os seus dados. Quando o assinante se move de um VLR para outro, o HLR é atualizado com essa informação.

A TMSI é utilizada para fornecer confidencialidade aos assinantes móveis. A TMSI é alocada dinamicamente pelo VLR, que associa a TMSI a IMSI. A TMSI é gravada no cartão SIM e é modificada periodicamente, substituindo o IMSI do assinante.

A TMSI é utilizada nas seguintes condições: estabelecimento de chamada; quando o móvel entra em uma nova LAI; quando o móvel entra em um novo VLR.

Cada MS possui um número internacional (ID), o qual é único em todo o planeta. O ID do MS é chamado de MSISDN e é formado pelos seguintes campos: código do país; número nacional de destino, o qual identifica a PLMN; número do assinante, o qual identifica o MS dentro da PLMN.

O MSISDN é utilizado no encaminhamento de chamadas entre a rede GSM e a rede pública RTPC, enquanto o IMSI é usado internamente na rede GSM.

O número da estação móvel visitante (MSRN) permite que um usuário habilitado em uma determinada região possa utilizar os serviços de outras como, por exemplo, efetuar chamadas em *roaming*. Neste caso, a conexão à rede é feita como assinante visitante.

AuC é responsável pelas funções de autenticação e criptografia. Estas funções fornecem segurança ao sistema inibindo a clonagem, por exemplo. Os processos de autenticação e a criptografia são executados sempre que o MS acessa a rede GSM. No entanto, eles só são ativados caso o serviço esteja habilitado para a MS. A chave de autenticação (ki) deve estar gravada no cartão SIM do MS e no AuC.

O IER uma base de dados centralizada com a lista dos números de identidade internacional do equipamento móvel (*International Mobile Equipment Identity* - IMEI), os quais devem ser únicos por IER e está organizada da seguinte forma: lista branca, a qual contém os IMEIs habilitados a utilizar o sistema; lista negra, a qual contém os IMEIs não habilitados a utilizar o sistema como, por exemplo, um MS que foi roubado; lista cinza, a qual contém os IMEIs que possuem algum tipo de problema, mas não justifica a sua presença na lista negra. Por exemplo, uma falha no *software*. O IMEI de um ME é formado pelos seguintes parâmetros: código de aprovação do ME; local da fabricação e manufatura do ME; número de série; versão do *software* instalado no ME.

A IWF é responsável pela interface da rede GSM com as demais redes de dados públicas e privadas. O IWF é utilizado na transmissão de dados comutados por circuitos (*Circuit Switched Data* - CSD).

O supressor de eco é o responsável pela retirada do eco nas conexões entre a MSC e a RTPC. O eco em uma chamada telefônica pode ser provocado por diversos motivos, como o atraso na propagação na interface aérea em função da distância; atraso na transmissão provocado pelo processo de transcodificação; geração de eco na bobina híbrida, utilizada para possibilitar a utilização de dois fios condutores para a transmissão e a recepção de sinais de voz nos telefones analógicos.

O OMC permite a administração, operação, manutenção e supervisão centralizada e remota dos elementos que formam a rede GSM. Ele é formado por dois subsistemas. São eles: centro de gerenciamento de rede (NMC) e o centro de operação e manutenção (OMC).

O NMC (Network Management Center) é o responsável pelo gerenciamento e controle dos OMCs. Suas principais funções são: monitorar os nós da rede; monitorar os dados estatísticos dos elementos da rede GSM; monitorar os OMCs.

O OMC está interligado com as entidades da rede GSM, tais como BTS, MSC, HLR, EIR etc. As principais funções do OMC são: gerenciamento de eventos e alarmes; gerenciamento da performance do sistema; gerenciamento de configuração. O OMC gerencia uma determinada área ou região e são classificados como: OMC (R), que controla o BSS; OMC (S), que controla o NSS.

2.11.2. Rede GPRS

O padrão *General Packet Radio Service* (GPRS) foi desenvolvido com o objetivo de atender as mudanças no perfil do usuário com a evolução da internet. O padrão GPRS é resultado da implementação de novos serviços e aplicações nas redes GSM. Os serviços GPRS utilizam os recursos já existentes na rede GSM, porém acrescenta uma infraestrutura para suportar a comunicação de dados pelo protocolo IP. As aplicações mais importantes providas pelo serviço GPRS são: comunicação por email; fax e acesso à internet; serviços de informação e jogos; comércio eletrônico (*e-commerce*); informações de localização, condições de tráfego e informações sobre voos; informações sob promoções, viagens, meteorologia etc. A rede GPRS suporta a transferência de dados por pacotes e permite o acesso às redes de dados por pacotes (Packet Data Network - PDN)

2.11.3. Arquitetura do Internet Protocol (IP)

A estrutura da arquitetura IP, sobre a qual trafegam os pacotes GPRS, é composta por quatro camadas. São elas: Camada de Aplicação; Camada de Transporte, Camada de Internet e Camada de Rede. A Figura 2.27 mostra a equivalência entre os modelos de referência OSI e o modelo IP.

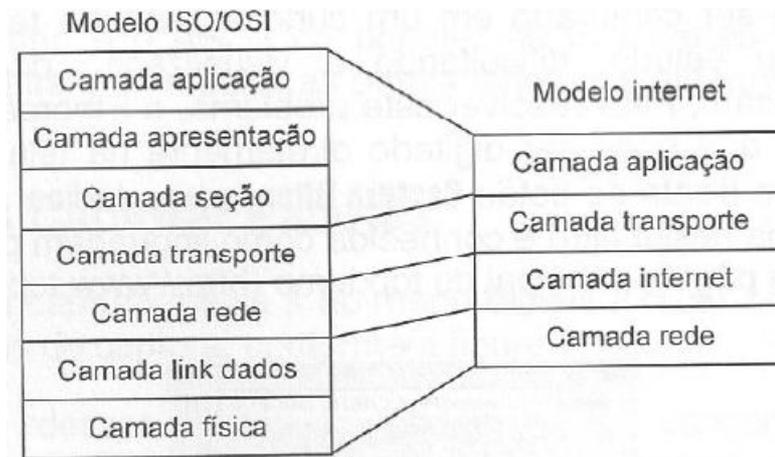


Figura 2.27: Modelo OSI *versus* IP (MOKARZEL e CARNEIRO, 2004)

A Camada de Aplicação é a responsável em executar a aplicação, por meio da qual o usuário vai estabelecer uma comunicação fim a fim com outro usuário da rede. Exemplos de aplicações executadas na Camada de Aplicação são: Telnet; HTTP; FTP; POP3; SMTP; DNS etc. A Camada de Aplicação deve fornecer à Camada de Transporte o pacote de dados a ser transmitido.

A Camada de Transporte é responsável em implementar a comunicação fim a fim entre usuários. Os protocolos utilizados nesta camada são o TCP e UDP. O protocolo TCP fornece serviços de controle de erro, controle de fluxo, de sequência e multiplexação do acesso à Camada Internet.

A Camada Internet, também chamada de Camada Inter-Redes, é a responsável pela transferência dos dados da máquina de origem até a máquina de destino. Ao fornecer o pacote à Camada Internet, a Camada de Transporte fornece o endereço IP da máquina destino, a quem o pacote deve ser entregue. O pacote é encapsulado em um datagrama IP, enquanto um algoritmo de roteamento verifica se o pacote pode ser entregue diretamente ou se deve ser encaminhado a um roteador. O pacote vai passando de um roteador para outro até que ele seja entregue ao destinatário.

A Camada de Rede, também chamada de Interface de Rede, Camada de Enlace, recebe os datagramas da Camada de Internet e os coloca no meio físico de transmissão.

2.11.4. Arquitetura GPRS

Os serviços GPRS visam prover o transporte de dados por pacotes IP com taxas de transmissão entre 9,6 e 171 kbps, sobrepondo à rede GSM. Para poder fornecer os serviços, são introduzidos novos componentes e interfaces na arquitetura GSM. A Figura 2.28 mostra o diagrama em blocos da arquitetura GPRS.

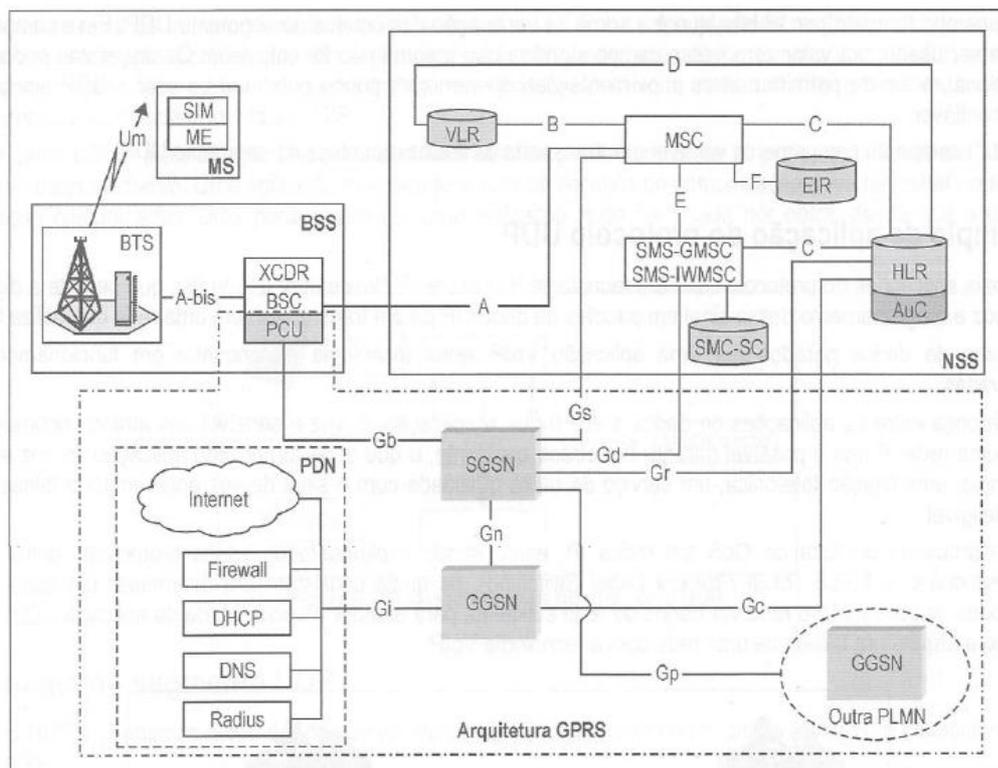


Figura 2.28: Arquitetura GPRS (SVERZUT, 2011).

A Tabela 2.1 mostra as alterações efetuadas na rede GSM para adequação do sistema para os serviços GPRS. Podemos observar que é necessário uma estação móvel preparada para acessar os serviços GPRS. Além disso, é necessário efetuar atualização de *software* nas BTSs, atualização de *software* e de *hardware* nas estações bases, implementação de novos elementos de rede chamados SGSN e GGSN, entre outras atualizações de *software* em outros componentes do sistema.

Tabela 2.1: Alterações na rede GSM de segunda geração introduzidas pelo GPRS. (SVERZUT, 2011)

Elemento de rede	Modificação ou atualização requerida pelo GPRS
Estação móvel (MS)	Uma nova estação móvel (MS) é requerida para acessar o serviço GPRS. Essas MSs são totalmente compatíveis com as versões anteriores, utilizadas no serviço de voz (comutação de circuitos)
Estação transceptora base (BTS)	Atualização (<i>upgrade</i>) de <i>software</i> nas BTSs existentes
Controlador de estação base (BSC)	Atualização (<i>upgrade</i>) de <i>software</i> nos BSCs existentes Instalação de <i>hardware</i> novo, chamado unidade de controle de pacote (<i>Packet Control Unit - PCU</i>). A principal função da PCU é direcionar o tráfego de dados para a rede GPRS
SGSN e GGSN	Instalação de novos elementos de rede, chamados: servidor do nó de suporte GPRS (<i>Serving GPRS Support Node - SGSN</i>) e <i>Gateway</i> do nó de suporte GPRS (<i>Gateway GPRS Support Node - GGSN</i>)
VLR, HLR, AuC, EIR, SMS-S e demais elementos de base de dados	Atualização de <i>software</i> para prover novos modelos de chamada e funções introduzidas pelo GPRS

2.11.5. Unidade de Controle de Pacote (PCU)

O PCU provê interfaces física e lógica para a implementação da transmissão de dados por pacote nas estações transceptoras base. A função da PCU é transportar o tráfego de dados por pacote do BSS até o *Serving GPRS Support Node* (SGSN).

2.11.6. Servidor do nó de suporte (SGSN)

O SGSN pode ser visto como uma MSC de comutação de pacotes, sendo um nó de rede introduzido pelo GPRS. Ele provê um ponto de acesso das MSs à rede GPRS. Os serviços fornecidos pelo SGSN são: detecção de novo usuários na área de serviço; processamento de registros de novos usuários; criptografia; manutenção dos registros de localização dos usuários dentro das áreas de serviço; Gerenciamento de mobilidade; Compressão dos dados; tarifação; processamento de operações de invocação e resposta para o HLR, com vistas a obter e manter os dados dos usuários GPRS. Uma SGSN se comunica com outra SGSN de PLMN da mesma operadora ou de operadoras distintas.

2.11.7. Gateway do nó de suporte GPRS (GGSN)

O GGSN provê a interface de conexão com as redes de pacotes externas. Sempre que um pacote for enviado para uma rede externa, ele deve passar pelo GGSN. Seus principais serviços são: manutenção das informações de roteamento; mapeamento dos endereços de rede

e assinante; mapeamento das classes da qualidade do serviço (QoS) nas classes utilizadas nas redes internet e PDN¹.

2.11.8. Principais Cenários GPRS

Antes de enviar ou receber dados pela rede GPRS, a MS precisa se registrar na rede. Para isso é necessário que alguns cenários sejam estabelecidos. Os principais cenários de processamento de chamadas executados em uma rede GPRS são: procedimento GPRS *Attach*; procedimento PDP *Context*; transferência de dados iniciada pela MS; transferência de dados iniciada pela rede.

O procedimento GPRS *Attach* é realizado no SSGN para registrar a presença de uma MS e registrá-la. Somente após o registro, a MS pode utilizar os serviços GPRS. Após o GPRS *Attach* é executado o procedimento PDP *Context*, o qual possibilita a MS obter um endereço IP. Após o procedimento PDP *Context*, a MS *está* apta a transmitir e receber dados por pacotes pela rede GPRS. Mais informações sobre o protocolo GPRS podem ser encontradas em (SVERZUT, 2011).

2.12. Tecnologia RFID

A tecnologia RFID foi inventada na década de 1930, porém somente na década de 1970 ela começou a ser utilizada em rastreamento de cadeia de suprimentos, controle de acesso, controle eletrônico em pedágio, detecção de furto em lojas, bibliotecas etc. A figura 2.29 mostra os componentes de um sistema RFID.

O *tag*, normalmente implementado na forma de etiqueta, cartão ou botão, possui um *micro chip* de memória semicondutora que é lida quando aproximada do leitor de RFID. O campo de atuação do leitor depende da potência de saída do transmissor de RF, das características da antena e pode variar de alguns centímetros até dezenas de metros. O *tag* pode ser de somente leitura ou de leitura e escrita. Cada *tag* possui um ID, único em todo o mundo.

¹ Rede Pública de Dados (PDN) é uma rede que pode ser acessada pelo público ou empresas em geral. Uma PDN permite a transferência de dados e o acesso a uma variedade de informações disponíveis como, por exemplo, dados estatísticos do IBGE, correio eletrônico (STM-400) etc. As principais redes PDNs no Brasil são a TRANSDATA e RENPAC, ambas disponibilizadas pela empresa Embratel.

Ao se aproximar do leitor, o *tag* é interrogado pelo leitor. Em seguida o leitor envia a informação obtida do *tag* para o *Middleware*, que processa as informações fornecidas pelo leitor. O *Middleware* envia a informação processada para o *Backend System*, que tem a função de executar a aplicação a nível de usuário, como, por exemplo, atualizar inventário, notificar embarque, notificar compra, autorizar o acesso etc.

Em *tags* de leitura e escrita, o leitor pode escrever informações no *tag*. Isso é possível porque os *tags* de leitura e escrita possuem um microcontrolador e uma memória semicondutora, a qual permite que dados sejam lidos e escritos em suas localidades. Os *tags* capazes de armazenar informações são classificados como de classe II, como é visto no item 2.12.1.

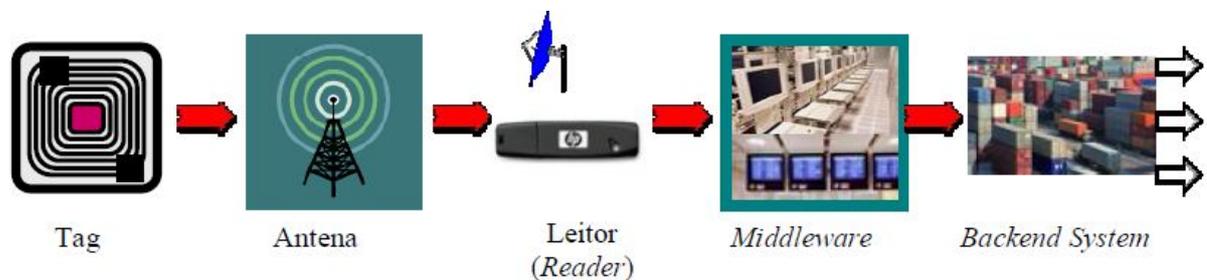


Figura 2.29: Componentes de um sistema RFID (NEMOTO *et al*, 2008)

2.12.1. Classificação do *Tag*

Os *tags* são classificados quanto a presença ou não de fonte de alimentação e quanto ao modo de operação. Quanto a fonte de alimentação o *tag* pode ser classificado como ativo, semi-passivo ou passivo. O *tag* ativo possui fonte de alimentação própria. O *tag* passivo não possui fonte de alimentação própria. Sua alimentação vem do sinal eletromagnético irradiado pela antena do leitor, o qual induz uma tensão na antena do *tag*. A tensão induzida na antena do *tag* é retificada e utilizada como fonte de alimentação do *tag*. O *tag* semi-ativo possui fonte de alimentação própria, mas utiliza o sinal irradiado pela antena do leitor para complementar a alimentação do *chip*.

Os *tags* são divididos em classes quanto a funcionalidade e comunicabilidade, as quais definem o seu modo de operação. São elas as classes 0, I, II, III, IV e V. As classes 0 e I agrupam *tags*

passivos e de somente leitura. Na classe II estão os *tags* capazes de armazenar uma pequena quantidade de informação. Os *tags* da classe III são semi-passivos e os da classe IV são capazes de se comunicar com outros *tags*. A classe V compreende os *tags* de somente leitura, mas com capacidade de energizar outros *tags* das classes I e II e de se comunicar com *tags* da classe IV (SCHUSTER, 2007).

As duas frequências mais utilizadas pelos *tags* são 13,56 MHz e 860-960 MHz. A frequência de 13,56 MHz em comunicação em baixa velocidade. Esta frequência normalmente é utilizada na identificação de metal, plástico, em bibliotecas e no controle de acesso. A frequência de 860-960 MHz normalmente é utilizada em comunicação em alta velocidade. Como exemplo, podemos citar aplicações como identificação de materiais em *pallets* e contêineres (NEMOTO *et al*, 2009). A Figura 2.30 mostra o diagrama em blocos do *tag* utilizado nesta dissertação, o *tag* MF1ICS50. Ele é um *tag* do tipo cartão e se comunica com o leitor na frequência de 13,56 MHz, a uma taxa de transferência de 106 kbps e gasta em média 100 ms para cada acesso. Ele possui sistema de tratamento contra colisão e é integrado com 1 kbytes de memória não volátil do tipo EEPROM. O *tag* em tela também possui um sistema de autenticação e um sistema de criptografia dos dados.

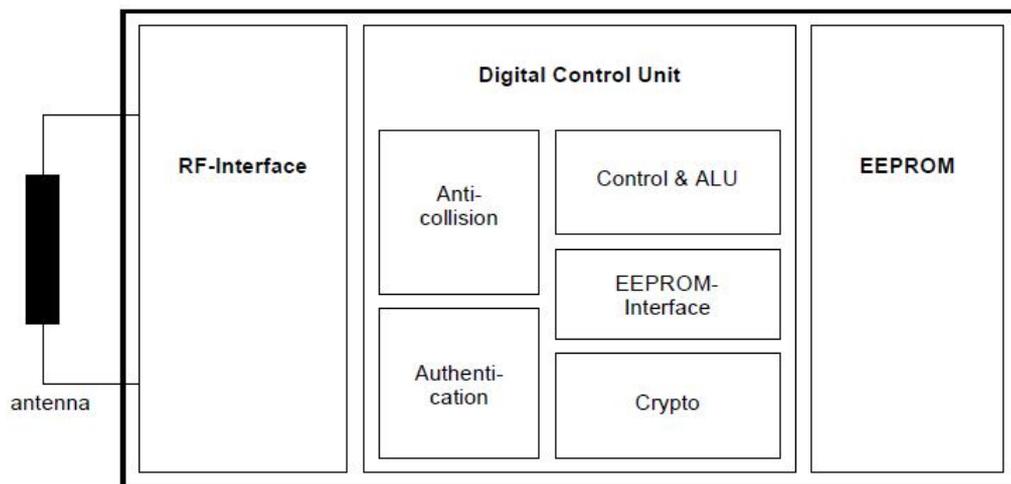


Figura 2.30: Diagrama em blocos do *tag* MF1ICS50 (NFX, 2008).

A memória EEPROM do *tag* MF1ICS50 possui uma capacidade de armazenamento de 1024x8 bits e é dividida em 16 setores, cada setor com 4 blocos. Cada bloco pode armazenar 64 bytes. Cada setor possui duas senhas de acesso distintas, podendo ser definida a forma de acesso e

tipo de permissão para cada uma das senhas. Além disso, cada *tag* possui um ID único em todo o planeta.

A Interface-RF possui os seguintes módulos: Modulador/Demodulador; Retificador; Gerador de *clock*; Regulador de voltagem; *Power On Reset*. Esses módulos têm como objetivo fornecer a tensão de alimentação e inicializar a CPU do *tag*, além de efetuar a comunicação do *tag* com o leitor. O bloco de anticóllisão permite que vários cartões sejam lidos sem que a leitura de um interfira na leitura do outro.

Os setores de memória EEPROM do *tag* são identificados de 0 - 15. Com exceção do setor 0, todos os outros setores podem armazenar dados nos três primeiros blocos. O último bloco de cada setor armazena as senhas de acesso do setor e os bits de *status* que definem como as informações armazenadas em cada bloco do setor são acessadas. Esse bloco é chamado de bloco *trailer*.

O primeiro bloco do setor 0 contém o ID do cartão, composto por 4 bytes, e outras informações interessadas ao fabricante. Os blocos 1 e 2 do setor 0 podem armazenar dados.

Capítulo 3 - DESCRIÇÃO DO SISTEMA

Este capítulo apresenta a descrição do sistema de gerência de vagas de estacionamento automotivo em vias públicas proposto por esta dissertação. Após a apresentação de alguns exemplos do modelo atual de gerência e de uma descrição geral do sistema, é apresentada a dinâmica de utilização do ponto de vista do utente e do ponto de vista do gestor. O sistema permite a cobrança eletrônica no molde pré-pago, fato que possibilitará o controle operacional e financeiro das vagas de estacionamento disponíveis, tornando mais transparente a sua administração e mais segura a sua utilização, uma vez que não haverá interferência humana no processo de cobrança.

3.1. Modelo Atual de Gerência

Na maior parte das cidades brasileiras, o modelo utilizado para a gerência das vagas de estacionamento em vias públicas é baseado na figura do operador de estacionamento, mais conhecido como flanelinha. Em troca de um pagamento feito em moeda corrente, o flanelinha emite um *ticket* ao utente quando ele estaciona o carro na vaga. O *ticket* deve estar fixado em local visível dentro do carro para que o fiscal, normalmente o guarda municipal, possa verificar se o utente pagou pelo estacionamento ou se o prazo contratado está ou não vencido.

O modelo baseado no flanelinha oferece algumas limitações e possibilidades de fraude, com ou sem a participação do flanelinha. O valor pago pelo normalmente está associado a um período de tempo predefinido. Se o utente não manter o carro estacionado durante todo o período contratado, o valor remanescente pago pelo tempo não utilizado é perdido. Normalmente o período contratado é de duas hora.

No que diz respeito a fraudes, uma observada com frequência nesse modelo é o flanelinha não emitir o *ticket* ao utente e receber um mote menor do que o que deveria ser pago pelo tempo estacionado. Outra fraude observada é a utilização de estacionamento pelo flanelinha em local não permitido, ou mesmo o flanelinha atuar sem autorização, personagem conhecido como

flanelinha pirata. O fato de que o flanelinha manipula o dinheiro pago pelo estacionamento é um complicador no que diz respeito a segurança do sistema.

Algumas iniciativas de gerência de vagas de estacionamento em vias públicas baseados em sistemas embarcados têm sido testadas em diversas cidades. Um exemplo é o da cidade de São José dos Campos, no Estado de São Paulo, que tem instalado um sistema automatizado de cobrança de estacionamento em vias públicas com parquímetros, como podemos ver na Figura 3.1.



Figura 3.1: Parquímetro em São José dos Campos, São Paulo.

Nesse modelo, o utente, ao estacionar, se dirige até o parquímetro, efetua o pagamento e recebe um *ticket* que deve estar fixado dentro do carro em local visível. Uma característica desse sistema é o fato de que o parquímetro armazena o dinheiro pago pelo estacionamento, o qual deve ser recolhido periodicamente. Além disso, o parquímetro é responsável por várias vagas, podendo ser necessário o utente ter de andar dezenas de metros para chegar até o parquímetro. Embora consista em um avanço se comparado ao sistema que utiliza o flanelinha, o sistema em questão oferece uma abordagem limitada no que diz respeito a segurança, uma vez que o equipamento pode ser violado e o dinheiro armazenado furtado. Além disso, não é possível recuperar créditos de estacionamento não utilizados.

Um outro modelo encontrado em algumas cidades é baseado no parquímetro mostrado na Figura 3.2, instalado na cidade de Campo Grande, Mato Grosso do Sul. Nesse modelo, cada parquímetro gerencia duas vagas de estacionamento. O utente adquire um *tag* de RFID

recarregável com crédito de estacionamento. Ao estacionar o carro na vaga, o utente aproxima o *tag* do leitor do parquímetro e descarrega os créditos de estacionamento correspondente ao tempo no qual ele deseja permanecer estacionado. Antes de retirar o carro do estacionamento, o utente deve aproximar novamente o *tag* do leitor para recuperar os créditos de estacionamento remanescente que não foram utilizados. Embora esse sistema apresente uma vantagem com relação ao sistema instalado em São José dos Campos, principalmente pelo fato dele não armazenar dinheiro e poder recuperar os créditos não utilizados, o parquímetro funciona na modalidade *stand alone*, ou seja, sem qualquer conexão com outros dispositivos, fato que torna a gerência do sistema, no que diz respeito ao controle financeiro, uma tarefa difícil. Um profissional deve ir em cada parquímetro periodicamente e efetuar o *upload* dos créditos armazenado para posterior contabilidade.



Figura 3.2: Parquímetro na cidade de Campo Grande, Mato Grosso do Sul.

Em Tokio, Japão, existe um sistema de gerencia de vagas de estacionamento em vias públicas instalado, cujo parquímetro é mostrado na Figura 3.3. Nesse modelo, o utente estaciona o carro em frente ao parquímetro e insere nele o valor correspondente ao tempo que se deseja permanecer estacionado. Um sensor de ultrassom instalado no parquímetro detecta o carro estacionado, servindo como instrumento de controle contra fraudes. Uma vez que foi detectado um carro estacionado, créditos de estacionamentos devem estar inseridos no parquímetro, em caso contrário pode ser um indício de fraude.



Figura 3.3: Parquímetro em Tóquio, Japão.

Um outro sistema de gerência de vagas de estacionamento automotivo em vias pública é encontrado em Londrina, no Estado do Paraná, baseado no parquímetro mostrado na Figura 3.4. Nesse modelo, cada parquímetro gerencia 10 vagas de estacionamento. Ao estacionar, o utente precisa verificar qual o número da vaga utilizada e aproximar o *tag* do leitor para inserir créditos de estacionamento para aquela vaga, selecionada no parquímetro. Nesse modelo não existem sensores para detectar a presença de carro em cada vaga, precisando de fiscalização constante.



Figura 3.4: Parquímetro em Londrina, Paraná.

3.2. Arquitetura do Sistema Proposto

Na modelagem do sistema proposto por esta dissertação foram estudados vários sistemas de gerência de vagas de estacionamento em vias públicas em funcionamento, no Brasil e no exterior. Alguns desses sistemas de gerência, citados no item anterior, foram estudados e avaliadas as principais características de cada um, do ponto de vista operacional.

A arquitetura proposta para o novo sistema, mostrado na Figura 3.5, integra vários módulos. O modelo integra tecnologias de sistema embarcado e redes de telecomunicações, além do controle e configuração dos parquímetro efetuado remotamente de uma sala de controle. No estudo feito para a definição deste modelo, não foi encontrado nenhum sistema em funcionamento que agregue todas as funcionalidades e características definidas no seu escopo.

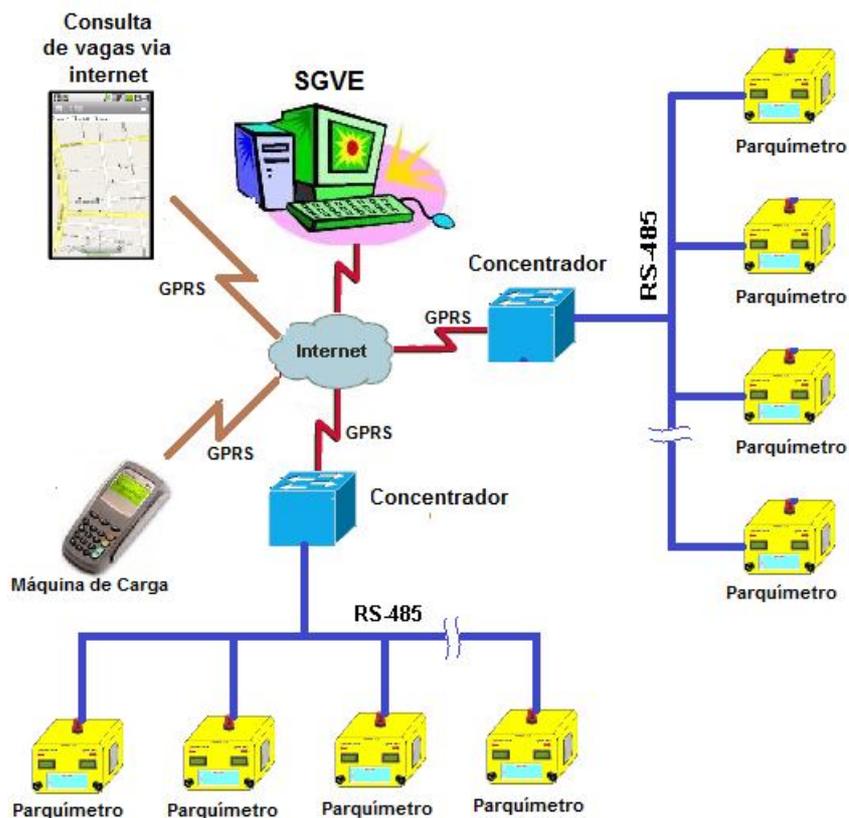


Figura 3.5: Arquitetura do Sistema de Gerência *e-SIGA* e Suas Interconexões.

3.1. Equipamento de Interface Com o Usuário

Um Equipamento de Interface Com o Usuário, doravante chamado de parquímetro, faz a cobrança eletrônica do tempo utilizado para o estacionamento. O parquímetro faz leitura automática, por proximidade, de cartão de crédito de estacionamento recarregável. A tecnologia de leitura por proximidade, utilizando rádio frequência, é conhecida como RFID (identificação por rádio frequência). O cartão é adquirido pelo utente, carregado e recarregado inúmeras vezes pelo Equipamento de Carga e Recarga. Este último *está* localizado em estabelecimentos comerciais próximos às vagas, como bancas de jornal, lojas, pontos de venda dedicados entre outros.

Cada parquímetro, que serve a duas vagas de estacionamento, está ligado em rede com vários outros parquímetros e com o concentrador, o qual também está conectado a uma central de gerenciamento, de onde todo o controle do sistema é feito via internet pelo Software de Gerenciamento de Vagas de Estacionamento (SGVE). A programação de cada parquímetro pode ser efetuada remotamente, de forma personalizada e de acordo com a legislação local, submetendo-se às definições e prioridades do gestor do sistema. O utente pode localizar uma vaga disponível próximo ao local de destino por meio do Software Para Serviço de Consulta de Vagas Via Internet. Software desenvolvido para ser executado a partir de dispositivo móvel conectado à internet.

A Figura 3.6 mostra um parquímetro proposto com os dispositivos de acesso e leitura. Deve-se notar na figura a existência de dois *displays* digitais, duas unidades de leitura/escrita de cartões via tecnologia RFID, dois medidores de distância por ultrassom, e um dispositivo de sinalização luminosa.



Figura 3.6: Parquímetro – Equipamento de Interface Com o Usuário.

O utente, após estacionar o carro na vaga, deve aproximar o cartão de crédito de estacionamento do leitor de RFID, localizado na lateral do parquímetro, para descarregar a quantidade de crédito correspondente ao tempo em que o carro deve ficar estacionado, o qual é mostrado no *display* digital. Quando o tempo desejado for mostrado no *display*, o cartão deve ser afastado do leitor. A partir de então, um cronômetro interno passará a decrementar o tempo, ficando o tempo remanescente sendo exibido no *display*. Quando o carro for retirado da vaga, antes que o cronômetro zere, o utente deve aproximar o cartão novamente do leitor para resgatar o crédito remanescente. Deste momento em diante, a vaga é considerada ociosa pelo SGVE. O parquímetro é composto por um *display* LCD, teclado, leitor de RFID, memória de massa, uma porta de comunicação RS-485 com o concentrador, RTC e sensor de presença por ultrassom.

Dois sensores de distância por ultrassom podem ser instalados em cada parquímetro para detectar a presença de carro nas vagas. Caso um carro estacione na vaga e não seja inserido créditos de estacionamento, o parquímetro *envia* ao SGVE, via concentrador, uma mensagem de que uma situação irregular pode estar ocorrendo naquela vaga. A mesma mensagem de suspeita de irregularidade pode ser enviada ao SGVE se o utente não retirar o carro da vaga antes dos créditos inseridos no parquímetro terminarem.

A Figura 3.7 mostra um cenário típico. São mostrados os parquímetros (identificados de A a H). Notar que cada parquímetro serve, simultaneamente, a duas vagas de estacionamento. Assim, a distância entre cada parquímetro é de aproximadamente 8 metros. Cada parquímetro se comunica com o concentrador, localizado fisicamente junto ao parquímetro D. O concentrador se comunica com o SGVE localizado em um servidor (Centro de Controle), através de conexão GPRS, EDGE, 3G ou outra tecnologia ou meio que estiver disponível. O SGVE faz o controle remoto dos parquímetros via concentrador.

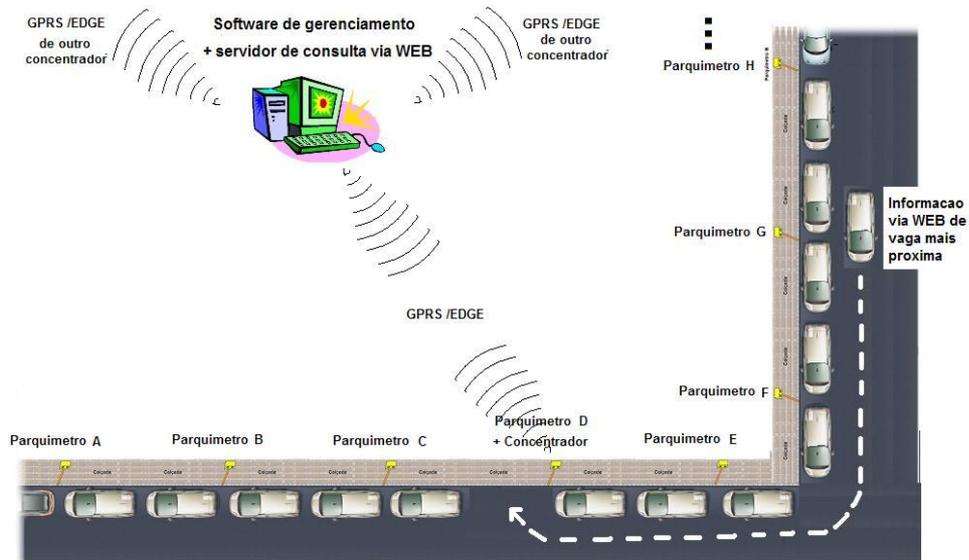


Figura 3.7: Sistema e-SIGA em um Cenário Típico.

3.2. Software Para Serviço de Consulta de Vagas Via Internet

Um utente pode localizar as vagas de estacionamento mais próximas por meio de consulta feita ao servidor para serviço de consulta de vagas via internet, como mostra a Figura 3.8. Com um aplicativo executado de um dispositivo móvel com conexão à internet, o utente pode visualizar a sua localização e o *status* de vagas das ruas adjacentes em um raio de aproximadamente 1000 m ou outro raio a ser configurado.



Figura 3.8: Aplicativo de Localização de Vagas Por GPS para S.O. Android.

3.3. Equipamento de Carga e Recarga

O equipamento de carga e recarga do cartão de estacionamento aparece na Figura 3.9. Este pode estar localizado em quaisquer estabelecimentos comerciais próximos às vagas, tais como bancas de revistas, padarias, lanchonetes, lojas entre outros.

A função do equipamento de carga e recarga é carregar o cartão do utente com créditos de estacionamento que ele utilizará posteriormente quando for estacionar. Sua conexão com o SGVE é efetuada por uma conexão GPRS, EDGE, 3G ou outra tecnologia ou meio que estiver disponível, sendo feita uma solicitação de venda ao sistema SGVE a cada operação de recarga. O controle de vendas é efetuado pelo SGVE, que armazena em um banco de dados todas as operações de venda efetuadas pelo estabelecimento para posterior prestação de contas.



Figura 3.9: Equipamento de carga e recarga do cartão.

O equipamento de carga e recarga é composto por um *display* LCD gráfico, teclado, impressora, leitor de RFID, memória de massa, RTC e um modem GPRS.

3.4. Concentrador

O concentrador desempenha um papel fundamental no sistema, pois ele serve como *gateway* entre a rede que conecta vários parquímetros e o SGVE. Toda a configuração dos parquímetros é enviada para o concentrador que, por sua vez, repassa a mensagem ao parquímetro endereçado.

A rede que conecta os parquímetros ao concentrador pode ser do tipo RS-485, Ethernet, Fibra Óptica, ou Wireless, dependendo de que tipo de rede demanda a menor infraestrutura e dá maior flexibilidade ao sistema. O concentrador utiliza um microcontrolador de 32 bits, memória de massa, RTC, porta de comunicação tipo RS-485, Ethernet, Fibra Óptica, ou Wireless e um modem GPRS.

3.5. Software de Gerenciamento de Vagas de Estacionamento

O Software de Gerenciamento de Vagas de Estacionamento (SGVE) faz a gerência de todo o sistema remotamente por meio de uma conexão segura de internet. O SGVE possui três blocos, os quais gerenciam cada parte do sistema e está conectado a um sistema de gerência de banco de dados.

O módulo de gerência de vagas de estacionamento faz a gerência dos parquímetros e suas consequentes configurações. O módulo de gerência do sistema de carga e recarga faz a gerência dos equipamentos de carga e recarga de cartões de crédito de estacionamento, enquanto o módulo de gerência do sistema de localização de vagas fornece informações sobre vagas ocupadas e ociosas para o serviço de consulta de vagas via internet. A Figura 3.10. mostra o diagrama em blocos do SGVE.

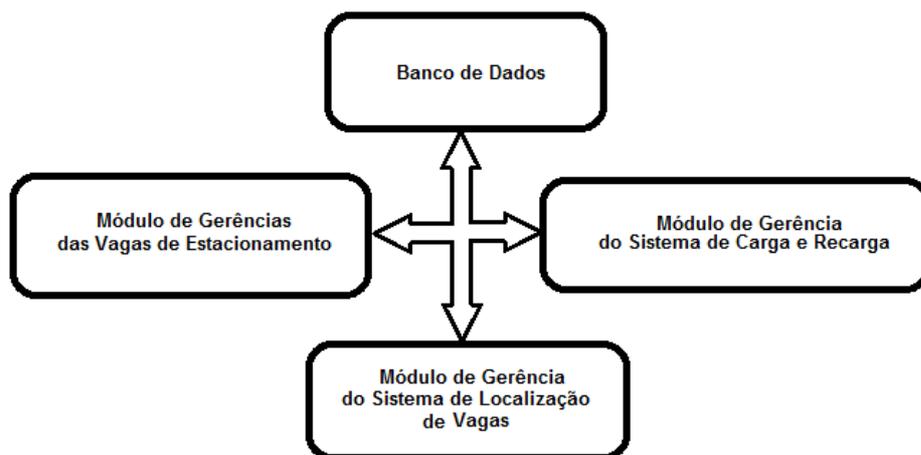


Figura 3.10: Arquitetura do SGVE.

3.6. Dinâmica de utilização do Sistema *e-SIGA*

A dinâmica de utilização do sistema proposto nesta dissertação parte de dois pontos de vista diferentes: ponto de vista do usuário e ponto de vista do gestor, ambos descritos na próxima seção.

3.6.1. Ponto de vista do Usuário

Para ter acesso aos serviços oferecidos pelo sistema, o utente deve adquirir o cartão de estacionamento em um posto autorizado. Estes, conforme um plano de negócio² a ser desenvolvido posteriormente a este estudo, deverão ser encontrados no maior número possível de estabelecimentos, possibilitando o acesso facilitado ao serviço. Ao adquirir o cartão, o cliente pagará antecipadamente (modalidade pré-pago) por quantas horas de estacionamento desejar, desde que não ultrapasse o limite estabelecido pelo gestor do sistema. Exemplo: o usuário planeja, em média, estacionar duas horas por dia. Assim, adquire de 40 a 50 horas de estacionamento por mês. Créditos são armazenados no cartão em moeda corrente.

De posse de um cartão de RFID carregado com créditos de estacionamentos, o utente se dirige à vaga que deseja estacionar. Após estacionar, o cartão deve ser aproximado do leitor de cartão RFID do parquímetro, localizado do mesmo lado da vaga que se encontra o veículo. Ao aproximar o cartão do leitor, ele é lido e, então, dado início a transferência de crédito, em minutos, requerido pelo usuário. Quanto mais tempo o cartão for deixado em contato com o leitor de cartão RFID, mais créditos de estacionamento são transferidos para o parquímetro. Os créditos de estacionamentos aparecerão no *display* do parquímetro correspondente à vaga que deseja estacionar. Ao afastar o cartão do parquímetro, é iniciada a contagem regressiva do tempo, tendo como início àquela correspondente ao crédito depositado, e o tempo

² Plano de negócios (do inglês *Business Plan*), também chamado "plano empresarial", é um documento que especifica, em linguagem escrita, um negócio que se quer iniciar ou que já está iniciado. Geralmente é escrito por empreendedores, quando há intenção de se iniciar um negócio, mas também pode ser utilizado como ferramenta de *marketing* interno e gestão. Pode ser uma representação do modelo de negócios a ser seguido. Reúne informações tabulares e escritas de como o negócio é ou *deve* ser.

remanescente aparecerá no *display*. O período de carência pode ser definido pelo gestor do sistema.

Quando o usuário retornar para retirar o veículo da vaga, o crédito remanescente, que *está* sendo exibido no *display* do parquímetro, pode ser resgatado. Para isso, basta encostar/aproximar o cartão novamente do leitor. O sistema identificará o cartão e faz a devolução dos créditos remanescentes. Caso o usuário não resgate os créditos remanescentes, eles são armazenados em uma memória de massa e poderão ser resgatados dentro de um prazo a ser determinado pelo gestor.

Caso o crédito do parquímetro termine com o automóvel ainda ocupando a vaga, ou um usuário venha a ocupar uma vaga sem o correspondente aporte de créditos, o dispositivo de sinalização luminosa entrará em funcionamento e, paralelamente, é enviado um alerta ao centro de gerência do sistema. O dispositivo de sinalização luminosa serve para que o funcionário autorizado identifique com maior facilidade o uso irregular da vaga de estacionamento e tome as providências previstas. A sinalização luminosa e o alerta ao centro de gerência são disparados a partir de informações fornecidas pelo medidor de distância por ultrassom. O sensor de ultrassom identifica a presença de um veículo estacionado na vaga por meio da detecção de um obstáculo, na direção e distância prevista para a vaga monitorada.

Existe a possibilidade de ser implementado um serviço de aviso via mensagens SMS (*Short Message Service*) por meio do qual o utente é avisado quando o crédito remanescente alcance 15 minutos. É importante mencionar que o sistema pode usar o serviço de mensagens SMS como um serviço extra ao usuário, e não como principal meio de comunicação, ou mesmo de tarifação do usuário. O serviço de mensagens de SMS pode estar congestionado – caso comumente observado nas diversas operadoras – e pode impossibilitar o utente de ser devidamente tarifado, podendo incorrer em multas.

Os dados relativos aos dois usuários que estiverem utilizando o parquímetro são armazenados no próprio equipamento, assim como a pilha de usuários que não resgataram os créditos remanescentes. Estes dados são todos mantidos em duplicidade no concentrador, permitindo a

recuperação dos dados em caso de falha. Os dados de todos os concentradores são copiados e mantidos em duplicidade no servidor de gerência do sistema.

O sistema permite ainda a consulta via Web das vagas disponíveis. Um *software* de navegação indica a localização geográfica do utente e as vagas ociosas mais próximas. Para isso, o *software* cliente, a ser instalado no *Pocket PC* ou em um *Smart Phone*, ou *tablet*, calculará a distância entre o usuário e as vagas ociosas, identificando e alertando para as vagas mais próximas. Para isso, o dispositivo móvel, com auxílio de GPS interno, usa as coordenadas geográficas do veículo, e as das vagas disponíveis para o cálculo da distância.

3.5.2. Ponto de vista do Gestor

O fato de os parquímetros estarem interligados em rede, podendo ser gerenciados de uma sala remota de controle, fornece maior flexibilidade em comparação ao modelo predominante atual. Por esse motivo, o sistema permite ao gestor definir todas as prioridades exigidas e alterá-las rapidamente, por meio de um terminal de computador, caso haja necessidade. Além disso, o sistema pode se conectar ao sistema de controle de tráfego viário disponibilizando informações que poderão ser utilizadas na melhoria do tráfego.

O gestor tem acesso ao monitoramento das vagas ocupadas e as ociosas, além das estatísticas de utilização das vagas. A partir dessas informações, é possível alimentar com dados alguns dos modelos matemáticos já existentes (ARNOTT e JHON, 2009), a partir do qual é possível definir tarifas de estacionamento que minimizem as possibilidades de congestionamentos nos grandes centros.

Como descrito anteriormente, o sistema envia os dados de dezenas de parquímetros para o centro de controle por meio dos concentradores. Um concentrador pode se comunicar com até trinta parquímetros, que ligados em rede constituem em uma zona da rede. Note que a falha de um concentrador pode ter, por consequência, a perda de informações preciosas tanto do ponto de vista da receita quanto da organização e ocupação do espaço público. Assim, a detecção de falha de parquímetros e concentradores deve ser tratada imediatamente pelo gestor. Para evitar a perda de dados em caso de falha do concentrador primário pode ser

utilizada a técnica da redundância, na qual dois concentradores recebem paralelamente os dados dos parquímetros. A falha do concentrador primário não representa perda de dados, uma vez que o concentrador secundário assumirá os serviços prestados por aquele. Dessa forma, o sistema emitirá relatórios sobre o estado de funcionamento dos parquímetros e concentradores, permitindo ao gestor o acionamento imediato das equipes de manutenção.

Pelo fato do sistema de recarga de cartões, assim como o de cobrança de estacionamento, não ter interferência humana direta, o gestor tem um controle rígido sobre a arrecadação. O gestor tem dois valores monitorados continuamente. O primeiro valor tem origem na receita obtida da venda de carga e recarga de cartões. O segundo valor é aquele contabilizado pelos parquímetros, referentes à utilização dos créditos. Note que o primeiro valor deve ser sempre maior ou igual ao segundo. Essa é uma poderosa ferramenta para a detecção de fraudes na carga e utilização dos cartões.

Uma importante ferramenta de auxílio à organização do tráfego é o uso da tarifa diferenciada. Para desenvolver uma política de tarifa diferenciada é importante que o gestor tenha em mãos estatísticas de utilização das vagas.

O gestor pode ter acesso à informação sobre a localização das vagas irregularmente ocupadas. Isso se deve ao fato dos parquímetros serem equipados com sensores de presença com capacidade de detectar a presença de um veículo estacionado. De posse dessa informação, o gestor pode tomar as decisões pertinentes, como a de comunicar ao fiscal responsável pela região, que deve se encaminhar ao local, averiguar a irregularidade apontada e tomar as devidas providências, parte da política adotada pelo gestor.

Como podemos observar, muitas são as possibilidades da utilização do sistema proposto para o gerenciamento de vagas de estacionamento automotivo em vias públicas, no qual o gestor pode introduzir novos recursos no sistema com o mínimo esforço como, por exemplo:

- A instalação de painéis luminosos, em pontos estratégicos, orientando os utentes sobre as ruas em que eles poderão encontrar vagas de estacionamento;

- Integração com sistema de vigilância por câmeras de vídeo, aumentando a segurança urbana;
- Oferecimento de novos serviços que venham utilizar a infraestrutura de telecomunicações e compartilhar os locais onde os parquímetros estão instalados.

Percebemos que o sistema proposto permite ao gestor preencher uma lacuna no planejamento estratégico das cidades de países em desenvolvimento, como o Brasil, podendo contribuir para evitar o colapso do tráfego de veículos nos centros das grandes cidades em um futuro próximo com uma política de estacionamento automotivo em vias públicas que utiliza parquímetros de terceira geração na gestão do estacionamento.

Capítulo 4 - DESENVOLVIMENTO DO SISTEMA

Este capítulo apresenta o modelo de desenvolvimento do sistema proposto por esta dissertação, além das principais características e funcionalidades de cada incremento do sistema.

O modelo escolhido para o desenvolvimento do sistema foi o incremental. Conceito normalmente utilizado em desenvolvimento de *softwares*, o modelo incremental parte do pressuposto de que um produto não precisa necessariamente ser entregue como um pacote, ao todo (NETO, 2008). O sistema pode ser desenvolvido em partes, independentes, chamadas de incrementos. Cada parte é integrada ao sistema, o qual vai evoluindo com novas funcionalidades a medida que novos incrementos vão sendo agregados ao sistema após terem sido desenvolvidos (PFLEEGER, 2004). O modelo incremental adiciona confiabilidade ao sistema a medida que um incremento é testado antes de ser integrado com os demais módulos. Cada incremento produz um sistema totalmente funcional, apesar de ainda não cobrir todos os requisitos (PRESSMAN, 2006).

O sistema foi modelado de forma que as várias partes integrantes pudessem ser desenvolvidas e testadas de forma independente e, posteriormente, integradas com os demais módulos. O desenvolvimento dos módulos baseados em sistemas embarcados - a máquina de recarga, o concentrador e o parquímetro - seguiram o modelo de referência eOSI, descrito no capítulo 2.

4.1. Máquina de Recarga

A máquina de recarga é a responsável pela carga e recarga de cartões com crédito de estacionamento. Assim como todos os módulos baseados em sistemas embarcados desenvolvidos para esta dissertação, a máquina de recarga foi projetada a partir do modelo de referência eOSI. As Camadas de Detecção e de Recuperação foram implementadas com os recursos de detecção e recuperação de falhas nativos ao microcontrolador utilizado no

desenvolvimento do projeto. Nas camadas de Comunicação e de Enlace foi implementada uma conexão GPRS com recursos de um *modem GPRS embedded hardware*. A camada de Redundância não foi implementada, ficando a cargo de estudos futuros.

A Figura 4.1 mostra o diagrama em blocos da arquitetura de *hardware* da máquina de recarga. Um microcontrolador de 32 bits interfaceia com um *display* gráfico LCD, um teclado hexadecimal de dezesseis telas, um *chip* relógio, uma impressora térmica, um leitor de RFID, um *modem* GPRS e uma memória de massa do tipo cartão SD.

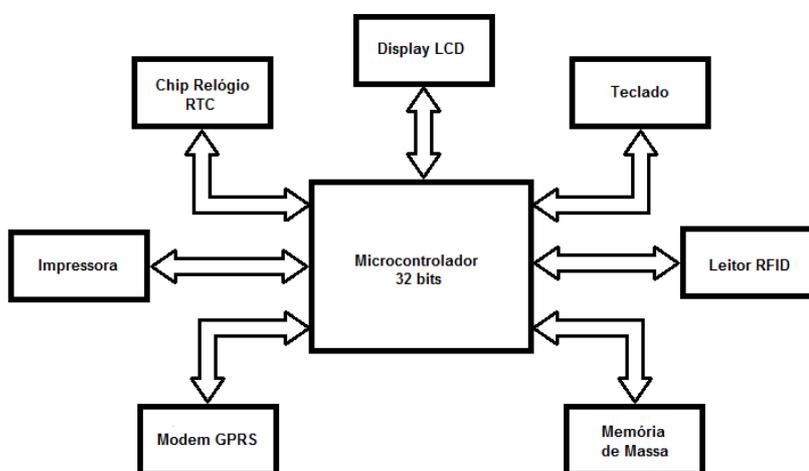


Figura 4.1: Diagrama em blocos do *hardware* da máquina de recarga.

Ao ser ligada, a máquina de recarga solicita que seja efetuado o *login* do caixa responsável pela máquina naquele período. Em seguida a máquina de recarga efetua um *login* no sistema de gerência (MGSCR), envia o ID da máquina e solicita uma chave para acessar os cartões de RFID. A cada venda de crédito de estacionamento efetuada, a máquina de recarga deve solicitar ao MGSCR um código de autorização, o qual irá identificar a venda. o código de autorização é gravado no cartão RFID juntamente com o crédito de estacionamento adquirido. O código de autorização gravado no cartão de recarga pode servir para conferência da utilização do cartão e também servir como referência para um possível bloqueio no cartão em caso de roubo ou furto.

A máquina de carga e recarga possui quatro *status* de funcionamento. São eles: em operação; bloqueada; roubada; em manutenção. Quando o *login* ou uma solicitação de venda é efetuada, o MGSCR verifica o *status* da máquina de recarga antes de autorizar o *login* ou a venda. Este

recurso configura um poderoso controle sobre as vendas, além de constituir em uma ferramenta de controle contra fraudes. A venda só pode ser autorizada se o *status* da máquina de venda for o *status* em operação. O *status* da máquina é reconfigurado pelo gestor do sistema sempre que a máquina mudar de *status*. Em caso de roubo ou furto, o gestor da máquina de venda deve imediatamente comunicar ao gestor do sistema para que ele altere o *status* da máquina de recarga roubada para evitar fraudes.

Ao realizar uma venda, a máquina de venda envia para o MGSCR o ID da máquina de recarga, o ID do caixa, o valor da venda e o ID do cartão a creditar. O sistema verifica se o cartão é válido, se o ponto de venda possui crédito e se a máquina está habilitada. Estando tudo Ok, o sistema envia para a máquina de recarga uma mensagem de autorização de venda contendo o ID da máquina, o ID do cartão, o código de autorização de carga (3 bytes), o valor a creditar e a data e hora da venda. O sistema registra a operação de venda (código de autorização, data e hora, ID do caixa, ID da máquina, ID do cartão valor).

Após a venda ter sido confirmada e a máquina de carga ter recebidos as informações relativas a venda, ela grava no cartão RFID os dados relativos à venda. São eles: o valor da venda, que deve ser somado ao valor existente no cartão; a data e hora da venda; o código de autorização de carga.

No desenvolvimento prático experimental da máquina de recarga foi utilizado um *modem* GPRS para a conexão com o MGSCR. O protocolo utilizado para a troca de mensagens entre a máquina de recarga e o MGSCR é o protocolo HTTP.

4.2. Concentrador

O concentrador executa uma função estratégica no sistema, pois ele serve como *gateway* entre a rede de dados que interliga os parquímetros de uma mesma zona e o SGVE, como pode ser observado na Figura 3.5. Além disso, é o concentrador quem gerencia a rede de parquímetros que compõe a sua zona. Cada concentrador pode ter até 30 parquímetros e uma zona pode conter um ou mais concentradores. Todas as mensagens enviadas aos parquímetros pelo

SGVE deverão passar primeiro pelo concentrador, que se encarrega de encaminhar a mensagem ao parquímetro endereçado.

Assim como a máquina de recarga, o projeto do concentrador seguiu o modelo de referência eOSI. A camada de Aplicação foi desenvolvida a partir dos recursos de *software* e de *hardware* do microcontrolador. As Camadas de Detecção e Recuperação foram implementadas com os recursos de detecção e recuperação de falhas nativos ao microcontrolador utilizado no desenvolvimento do projeto. As Camadas de Comunicação e de Enlace foram implementadas por meio de uma rede Modbus, rodando sobre o padrão RS-485. Funcionando como *gateway*, um segundo barramento de comunicação foi implementado na Camada de Comunicação e de Enlace. O segundo barramento utiliza o protocolo GPRS para acessar a internet e efetuar a comunicação com o SGVE. A Camada de Redundância não foi implementada, ficando a cargo de estudos futuros.

A Figura 4.2 mostra o diagrama em blocos da arquitetura de *hardware* do concentrador. Um microcontrolador de 32 bits interfaceia com um *chip* relógio, um *modem* GPRS e uma memória de massa do tipo cartão SD e uma interface RS-485.

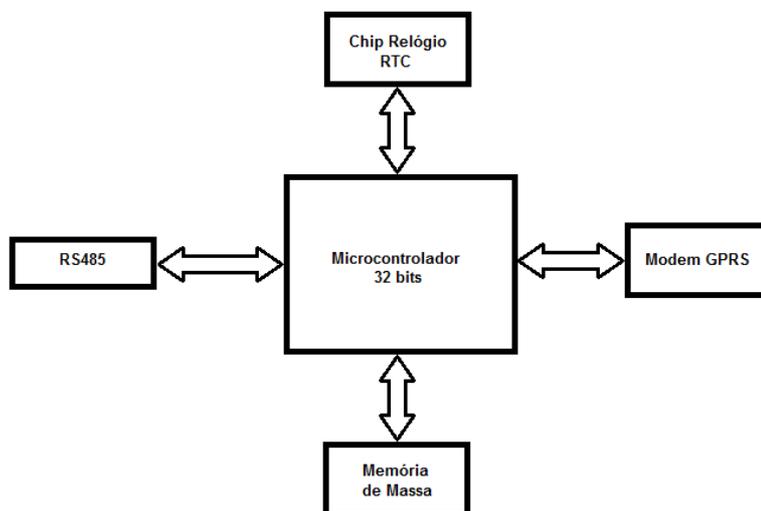


Figura 4.2: Diagrama em blocos do *hardware* do concentrador.

O concentrador opera como dispositivo mestre na rede RS-485 que interliga os parquímetros. Isso significa que é ele quem inicia a transmissão de dados com os parquímetros. A rede

padrão RS-485 foi escolhida para a montagem prático experimental desta dissertação por ser um padrão consagrado de comunicação de dados para baixas velocidades de transmissão, principalmente no que diz respeito a imunidade a ruídos. Outros meios de transmissão, como fibra óptica e *wireless*, podem ser comparados com o padrão RS-485 para se verificar a melhor resposta a ruídos, a menor taxa de erros e melhor resistência mecânica. Esta comparação, no entanto, ficará a cargo de estudos futuros por não ser objeto de estudo desta dissertação.

4.2.1. Comunicação Com os Parquímetros

Cada parquímetro possui dois endereço lógicos (ID). Ao ser energizado, o concentrador recebe do SGVE o ID de todos os parquímetros. Em seguida ele envia uma mensagem para cada parquímetro com o seu respectivo ID para verificar se o parquímetro está em operação. O parquímetro deve responder com uma mensagem de reconhecimento, informando que está em operação. Se o parquímetro não responder, fica configurado um indicativo de falha. O indicativo de falha é então reportado para o SGVE, que sinalizará ao operador do sistema.

A Figura 4.3 mostra o diagrama em blocos da arquitetura de *hardware* do parquímetro. Um microcontrolador de 32 bits interfaceia com um teclado, *display* LCD, leitor de RFID, sensor ultrassom, memória de massa do tipo EEPROM, *chip* relógio e uma interface RD-485.

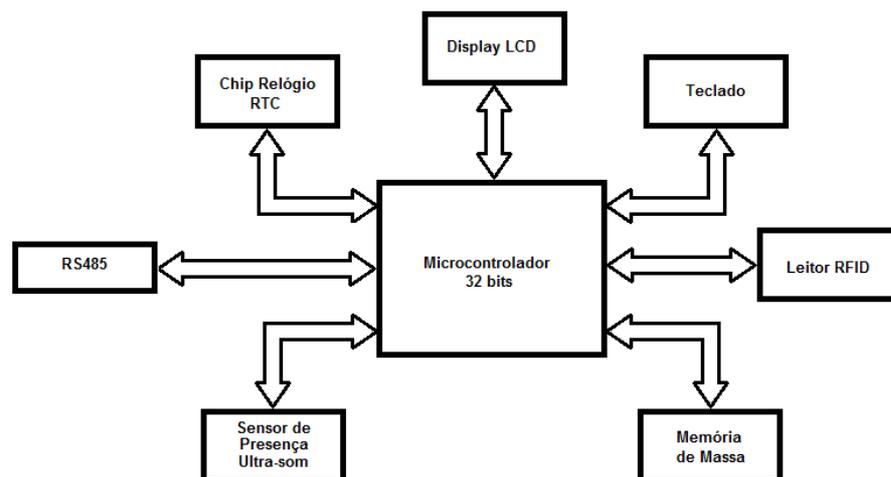


Figura 4.3: Diagrama em blocos do *hardware* do parquímetro.

De tempos em tempos, a ser configurado pelo sistema, ou sempre que um novo parquímetro for conectado à rede, o concentrador verifica se os parquímetros conectados a sua rede estão em operação. Após a verificação de parquímetro em operação, o concentrador deve enviar ao SGVE a relação dos parquímetros com os seus respectivos *status* de funcionamento. Estes são: em operação com vaga ocupada (OO), em operação com vaga ocupada e pagamento efetuado (OP) em operação com vaga livre (OL), sem comunicação (SC), em falha (PF) e em operação com indicativo de fraude (OIF). Este último *status* é ativado quando uma determinada vaga fica livre por um tempo maior que a média de tempo calculada para aquela vaga livre naquele horário. Esta verificação é efetuada pelo SGVE para todas as vagas do sistema.

A Tabela 4.1 mostra as informações que cada parquímetro envia para o concentrador em resposta a mensagem de verificação de parquímetro em operação.

Tabela 4.1: Resposta a mensagem de verificação de parquímetro em operação.

ID parquímetro	Status	Data	Hora
-----------------------	---------------	-------------	-------------

Como a rede que interliga os parquímetros e o concentrador possui a configuração mestre/escravo, o concentrador precisa interrogar os parquímetros de tempos em tempos para ver se houve alguma alteração no *status* de um ou mais parquímetros. Esta requisição tem como resposta os dados da Tabela 4.2. Ao ser interrogado sobre a mudança de *status*, o parquímetro deve enviar para o concentrador o seu ID, o *status* atual, a data e hora, além do ID do cartão, caso ele esteja no *status* em operação com vaga ocupada. Após receber a mensagem de mudança de *status* do parquímetro, o concentrador a repassa automaticamente para o SGVE, que a armazena em um banco de dados.

Tabela 4.2: Mensagem de mudança de *status* do parquímetro.

ID parquímetro	Status	Data	H: inicial	ID do cartão/sistema
-----------------------	---------------	-------------	-------------------	-----------------------------

Além das mensagens de verificação de parquímetro em operação e de mudança de *status*, o concentrador deve receber do SGVE as seguintes mensagens: lista de horários e valor/hora de cada período; *firmware* de atualização do concentrador; *firmware* de atualização dos parquímetros; hora e data para sincronizar o horário dos parquímetros; solicitação de *log* de utilização dos parquímetros.

Um segundo concentrador pode ser conectado à rede de uma zona de parquímetros para tornar o sistema tolerante a falhas. Caso o concentrador principal entre em *status* de defeito, o SGVE pode redirecionar o trabalho de gestão de uma zona para o concentrador secundário. O impacto desta redundância, no entanto, não é objeto deste estudo, ficando a cargo de estudos futuros.

No desenvolvimento prático experimental do concentrador foi utilizado um modem GPRS para a conexão com o MGSCR. Esta conexão é efetuada via conexão GPRS. O protocolo utilizado para a troca de mensagens entre a máquina de recarga e o MGSCR é o protocolo HTML, sendo os dados encriptografados para a proteção da informação.

4.3. Parquímetro

Assim como a máquina de carga e recarga e o concentrador, o projeto do parquímetro também seguiu o modelo de referência eOSI. A Camada de Aplicação foi desenvolvida a partir dos recursos de *software* e de *hardware* do microcontrolador. As Camadas de Detecção e Recuperação foram implementadas com os recursos de detecção e recuperação de falhas nativos ao microcontrolador utilizado no desenvolvimento do projeto. Nas Camadas de Comunicação e de Enlace foi implementada uma comunicação de dados com o protocolo Modbus rodando sobre o padrão RS-485. A Camada de Redundância não foi implementada, ficando a cargo de estudos futuros.

Quando o parquímetro é conectado à rede de parquímetros pela primeira vez, ele deve esperar uma mensagem do concentrador com o seu ID, solicitando o seu *status* de operação. A inclusão do novo parquímetro no sistema é efetuado pelo operador do sistema SGVE. Quando o concentrador receber a lista atualizada dos parquímetros de sua rede, ele envia a mensagem

de verificação de *status* de operação ao novo parquímetro. O novo parquímetro deve responder a mensagem do concentrador informando que o seu *status* é o de em operação com vaga livre. A partir de então ele *está* apto a ser utilizado pelos usuários do sistema.

Os parquímetros deverão manter uma tabela de *log* da utilização das vagas, com os dados apresentados na Tabela 4.3, a qual deve ser enviada ao concentrador quando solicitada. Isso deve ser feito no final do expediente. Em outras palavras, no final de cada expediente o concentrador deve requisitar a cada parquímetro o *log* de utilização de cada vaga daquele dia. Esta tabela deve ser enviada ao SGVE, a qual é armazenada em um banco de dados. Com este *log* o SGVE pode efetuar uma estatística de utilização de vagas que pode servir como referência para o cálculo da média diária de utilização de cada vaga. Esta média pode ser utilizada como indicação de fraude caso a vaga fique ociosa por um tempo maior do que o calculado para aquele horário. Em caso de falha no concentrador, o parquímetro deve manter o *log* de utilização das vagas por um período de pelo menos uma semana para que os *log* de utilização de cada vaga não sejam perdidos. Como cada parquímetro gerencia duas vagas elas devem se identificar. As vagas são identificadas como vaga 1, a direita e vaga 2, a esquerda do parquímetro.

Tabela 4.3: *Log* de utilização da vaga.

ID parquímetro	status	Data	Hora: Início	Hora: fim	ID do cartão/sistema	Valor da carga	Valor cumulado do dia
----------------	--------	------	--------------	-----------	----------------------	----------------	-----------------------

De acordo com o que foi descrito no capítulo 3, a utilização de um parquímetro para a gestão de vagas de estacionamento consiste, em primeiro lugar, em o utente verificar se a vaga gerenciada pelo parquímetro está livre. Estando a vaga livre, o utente deve estacionar o carro e aproximar o cartão de estacionamento do leitor de RFID para inserir créditos de estacionamento no parquímetro. O leitor de RFID do parquímetro verifica se existe crédito no cartão e, em caso positivo, ele retira os créditos do cartão correspondente ao tempo em que o utente pretende manter o carro estacionado na vaga. A cada cinco segundos que o cartão se mantém próximo ao leitor, são retirados dele créditos correspondente a quinze minutos. O tempo em minutos que está sendo inserido no parquímetro é visualizado pelo utente no visor

do parquímetro. Quando o utente decide que os créditos inseridos no parquímetro são suficientes ele deve afastar o cartão do leitor. A partir de então o visor do parquímetro coloca no seu mostrador o tempo creditado e, por meio de um contador decrescente, o utente pode visualizar o tempo que lhe resta. Quando o utente vai retirar o carro da vaga ele deve novamente aproximar o cartão do leitor de RFID do parquímetro para que os créditos não utilizados sejam retornados ao cartão. Este procedimento colabora para uma tarifa paga pelo uso do solo mais justa, a medida que o utente só paga pelo tempo utilizado.

4.4.1 Exceções à Regra

Algumas ações classificadas como exceção à regra são descritas a seguir. Ações são tomadas pelo sistema para minimizar os impactos dessas exceções. Se o utente esquecer de encostar o cartão no leitor de RFID e sair com o carro, o sistema responderá da seguinte forma: o contador decrescente continuará a decrementar e os créditos remanescentes são utilizados até que os créditos de estacionamento inseridos no parquímetro sejam zerados. Como não existe um carro estacionado na vaga, outro utente pode estacionar e aproximar o seu cartão do leitor. Neste caso, o parquímetro interrompe a contagem de tempo do cartão atual e os créditos remanescentes são inseridos em uma pilha implementada por *software*. Se o utente retornar para recuperar os créditos ele pode fazer isso simplesmente aproximando o seu cartão do leitor de RFID. A recuperação dos créditos não utilizados, só pode acontecer até o final do expediente. Em caso contrário, os créditos são perdidos.

Outra exceção à regra é o caso de o utente estacionar o carro na vaga e não colocar créditos de estacionamento no parquímetro. Neste caso, o sensor de presença, ao verificar que existe um carros estacionado e não foram colocados créditos de estacionamento em até 15 minutos, faz o parquímetro mudar o seu *status* para em operação com indicação de fraude, o que é reportado ao concentrador. O concentrador encaminha a mensagem com indicação de fraude para o SGVE, que sinaliza para o operador. O operador pode tomar uma decisão pertinente para a verificação da indicação de fraude enviando, por exemplo, um fiscal vai ao local. O fiscal pode notificar o utente a regularizar da irregularidade, ou até mesmo lhe aplicar uma multa se a legislação permitir.

Outra exceção à regra é o utente inserir créditos de estacionamento sem o carro estar estacionado na vaga. Neste caso, o sistema vai decrementando os créditos inseridos até que eles terminem ou outro utente estacione o carro na vaga e insira novos créditos. Isso faz com que os créditos do cartão anterior sejam enviados para a pilha.

4.4.2. Mensagens que Trafegam na Rede dos Parquímetros

Sendo o concentrador o dispositivo mestre da rede dos parquímetros, é sempre ele quem inicia a comunicação com os parquímetros. As mensagens que trafegam nesta rede são encapsuladas em pacotes Modbus. São elas: Requisição de *status* de operação; Requisição de *log* de utilização de cada vaga; Tabela de preços e horários; Atualização do *firmware* do parquímetro. As duas primeiras são mensagens de leitura enquanto as duas últimas mensagem.

Quando o parquímetro recebe uma mensagem de requisição de *status*, ele deve enviar como resposta uma mensagem com os dados da Tabela 4.1. A mensagem de requisição de *log* de utilização de vaga ele deve respondida com os dados da Tabela 4.2. Como as mensagens tabela de preços e horários e atualização do *firmware* são mensagens de escrita, o parquímetro deve responder ao concentrador se o comando foi executado ou não. Caso o comando não tenha sido executado o parquímetro deve mudar o seu *status* para em falha.

4.4. Software de Gerenciamento de Vagas de Estacionamento (SGVE)

Como descrito no capítulo 3, o *Software* de Gerenciamento de Vagas de Estacionamento (SGVE) faz a gerência de todo o sistema remotamente por meio de uma conexão de internet, incluindo a máquina de carga e recarga, o concentrador, os parquímetros e o sistema de localização de vagas livres. O SGVE é composto por três blocos, os quais gerenciam cada parte do sistema. Os três blocos que compõem o SGVE estão conectados a um sistema de gerência de banco de dados.

4.4.1. Módulo de Gerência do Sistema de Carga e recarga (MGSCR)

O módulo de Gerência do Sistema de Carga e Recarga (MGSCR) tem como objetivo gerenciar a máquina de carga e recarga. Este módulo é um dos blocos do SGVE, executado no servidor localizado na sala de controle. A seguir são apresentados os requisitos funcionais do MGSCR.

- O MGSCR, ao receber um pacote de autenticação do ponto de venda (enviado após a máquina de venda efetuar *login*), envia um pacote com a chave para acesso aos cartões de RFID.
- Ao ser realizada uma venda, a máquina do ponto de venda solicitará ao servidor uma autorização. Caso o servidor autorize a venda, é enviado para o ponto de venda um código de autorização (número de autorização de carga) com 3 bytes, data e hora da venda. O código de autorização está associado ao ponto de venda e não ao sistema. O código de autorização é armazenado em um registro relacionado ao ponto de venda.
- Cada ponto de venda tem uma conta no sistema na qual são armazenados os dados cadastrais da empresa e o histórico de vendas, incluindo os códigos de autorização.
- Cada ponto de venda pode ter mais de uma máquina de venda.
- A cada período, acordado em contrato, o sistema faz um balanço das vendas do período e gera uma fatura e um boleto, que são enviados ao ponto de venda. Cada fatura tem um código de identificação. A fatura é constituída da consolidação das vendas, por dia, dentro do período. O balanço de todas as máquinas cadastradas no ponto de venda deve constar na fatura.
- Quando a fatura for paga, o sistema deve registrar o pagamento e emitir a NF correspondente. Caso a fatura não tenha sido paga, após um número de dias a ser determinado, o sistema deve bloquear o ponto de venda.
- As máquinas do ponto de venda possuem vários *status* de operação. São eles: em operação, bloqueada, roubada e em manutenção. O ponto de venda possui um horário

de funcionamento, fora do qual não é dada autorização de venda. O sistema deve registrar o histórico de utilização das máquinas de venda.

- O sistema pode gerar planilhas com as vendas efetuadas por período, em versão pdf e xls, de todos os pontos de venda.

4.4.2. Casos de Uso do MGSCR

A seguir são apresentados os casos de uso do MGSCR. São eles: controlar operações de venda; controlar faturas; controlar pontos de venda.

O Controle de Operações de Venda inclui os casos de uso envolvidos no controle de cartões. São eles: adicionar cartões; inativar cartões; atualizar tipo. Este último define quem são os atores que tem acesso à máquina de recarga. Estes podem ser Gerente, Caixa, Fiscal ou Usuário (operador da máquina de venda). Para adicionar cartões, o administrador do ponto de venda deve especificar uma faixa contínua de cartões a serem cadastrados. O sistema verifica se existem cartões ocupando a faixa escolhida. Caso exista, a operação de cadastro é cancelada, devendo ser informado o motivo. Em caso contrário, o sistema cria novos cartões, cada um com um ID sequencial na faixa especificada. A inativação de um cartão requer que seja informado o motivo.

Os casos de uso envolvidos no Controle de Faturas inclui operações de controlar a emissão e cobrança de faturas dos pontos de venda. Os métodos destes casos de uso incluem listar faturas em atraso, emitir faturas, receber comprovantes de pagamento de fatura, bloquear pontos de venda e alterar fatura. As operações relacionadas a alteração de fatura são: corrigir lançamentos discordantes; retirar juros e multa; emitir segunda via de fatura. Diariamente o sistema verifica todos os pontos de venda com faturas em atraso e bloqueia os pontos de vendas com mais de N (configurar no sistema) de atraso e marca esses pontos de venda como bloqueados. Um ponto de venda com *status* de bloqueado não receberá autorização para realizar vendas até ele retorne ao *status* de em operação. Periodicamente (diariamente ou em dias do mês previamente determinados) o sistema verifica, para cada ponto de venda, se é dia de emissão de fatura. Neste caso, o sistema emite uma fatura para o ponto de venda com o

total a pagar e marca a fatura com o estado de em aberta. A pedido do operador, o sistema lista todos os pontos de venda com faturas em atraso. O sistema recebe dos bancos o arquivo de pagamento de faturas. Para cada fatura paga, o sistema registra a fatura como paga e os dados de pagamento da fatura. Para cada fatura paga em atraso, o sistema verifica se o ponto de venda foi bloqueado em função desta fatura e, neste caso, desbloqueia o ponto de venda.

Os casos de uso relacionados às operações de venda de crédito de estacionamento são: habilitar operação da máquina de venda; inabilitar automaticamente máquina de venda; inabilitar operação de máquina de venda; realizar uma venda; validar Gerente de ponto de venda; validar ponto de venda; validar máquina de venda.

A habilitação da operação de máquina de venda consiste no sistema emitir uma autorização de venda. Na operação na máquina de venda, um Gerente do ponto de venda se identifica e requisita a habilitação de uma máquina de venda. A máquina envia uma mensagem de habilitação para o sistema. O sistema toma as seguintes ações: valida a máquina de venda, valida o ponto de venda, valida o Gerente, verifica se a data/hora de habilitação está no período de funcionamento do ponto de venda. Estando tudo ok, o sistema envia a chave de acesso aos cartões RFID e registra a máquina no estado de habilitada. Em caso contrário, o sistema rejeita a operação e envia uma mensagem de erro com o motivo do bloqueio.

A operação de inabilitar automaticamente máquinas de venda ocorre após 30 minutos ter terminado o expediente. Periodicamente, o sistema verifica todos os pontos de venda e, para aqueles cujo horário de expediente terminou por mais de 30 minutos, as máquinas de recarga deles são desabilitadas até que um novo pedido de habilitação seja efetuado.

A operação de inabilitar operação de máquina de venda é efetuada por meio do Gerente de ponto de venda. O Gerente do Ponto de venda se identifica e requisita a inabilitação de uma máquina de venda. A máquina envia uma mensagem de inabilitação para o sistema. O sistema verifica se a máquina está habilitada e atualiza o *status* da máquina para inabilitada.

Para efetuar uma venda, o Caixa do ponto de venda precisa digitar sua senha e selecionar operação de venda. Em seguida ele deve aproximar o cartão do usuário da máquina de venda e

digitar o valor a creditar. A máquina envia uma mensagem de venda para o sistema contendo: ID da máquina; ID do caixa; valor da venda; ID do cartão a creditar. O sistema verifica se o cartão é válido, se o ponto de venda possui crédito e se a máquina está habilitada. Estando tudo ok, o sistema envia uma mensagem de autorização de venda contendo o ID da máquina, o ID do cartão, o código de autorização de carga (3 bytes), o valor a creditar e data e hora da venda. O sistema registra a operação de venda (código de autorização, data e hora, ID do caixa, ID da máquina, ID do cartão valor).

O sistema também pode validar Gerente de ponto de venda, validar máquina de venda e validar ponto de venda. O Sistema verifica se as informações do gerente (ID do seu cartão e senha) conferem com a de Gerente cadastrado para o ponto de venda onde a máquina de vendas está associada.

Para validar a máquina de vendas o sistema verifica se ela está ativa e associada a um ponto de venda. Para validar ponto de venda o sistema verifica se o ponto de venda onde a máquina de vendas está associada está ativo e com pagamento de fatura em dia.

O Controle de Ponto de Venda envolve os seguintes casos de uso: inserir novo ponto de venda; remover novo ponto de venda e editar ponto de venda. Este último inclui as operações de incluir Gerente, remover Gerente, editar Gerente, suspender Gerente e reativar Gerente.

O MGSCR permite que um utente, ao adquirir um cartão, possa ser cadastrado no sistema. Os casos de uso para este recurso são: incluir novo usuário; alterar dados do usuário; visualizar a utilização de cartões. Para inserir novo usuário no sistema, ele precisa se cadastrar no sistema, fornecer os seus dados pessoais (nome, CPF/CGC, endereço, *login* e senha) e um ou mais número de cartão de estacionamento associados. O sistema registra estas informações, data e hora do cadastramento. Só é possível ter um cadastro por CPF/CGC. Uma vez que o usuário está cadastrado, o sistema permite que ele altere os seus dados após o *login*. Um usuário, após a autenticação no sistema, pode requisitar o relatório de utilização dos seus cartões em um determinado período. O sistema gera um relatório por cartão em ordem cronológica de utilização, a partir da data de cadastro de cada cartão até o fim do período de estacionamento.

4.5. Módulo de Gerência de Vagas de Estacionamento (MGVE)

O MGVE tem como objetivo gerenciar os concentradores, os parquímetro e as vagas de estacionamento. Este módulo é um dos blocos do SGVE, executado no servidor localizado na sala de controle. A seguir são apresentados os requisitos funcionais do MGVE.

- O MGVE deve enviar para o concentrador uma lista com os ID dos parquímetro pertencentes a sua zona. Para isso o parquímetro estar cadastrado no sistema com o seu modelo, número de série, fabricante, ID, zona e coordenadas geográficas. Os concentradores também devem ser cadastrados com os seus respectivos número de série, ID, coordenadas geográficas e a zona a qual pertencem.
- Sempre que um parquímetro mudar de *status* o MGVE deve ser informado para que seja atualizado no sistema o novo *status*.
- O MGVE deve cadastrar os pontos de visualização e definir as vagas disponíveis para estacionamento e disponibilizar para o Serviço de Consulta de Vagas via Internet a localização geográfica das vagas disponíveis e das vagas ocupadas.
- O MGVE deve enviar aos parquímetro de mesma zona a tarifa valor/hora cobrada pelo estacionamento nos diversos períodos do dia.
- O MGVE deve receber dos parquímetro o *log* de utilização das vagas de estacionamento para que o sistema possa efetuar um cálculo estatístico do tempo médio de ociosidade de cada vaga em cada período do dia.
- O MGVE deve ser informado em caso de indicação de fraude como, por exemplo, no caso de o utente estacionar o carro na vaga e não inserir créditos de estacionamento no parquímetro.

4.5.2. Casos de Uso do MGVE

A seguir são apresentados os casos de uso do MGVE na gerência do estacionamento. São eles: controle de concentradores; controle de preços; pontos de visualização; definição de vagas; cadastro de concentradores; zona de estacionamento; controle de ocupação; controle de fiscalização.

As operações envolvidas nos casos de uso utilizados no Controle de Concentradores são: inicializar concentrador; atualizar *status* do concentrador; atualizar *status* dos parquímetros.

Para inicializar o concentrador, no início de cada expediente, o MGVE envia a ele uma mensagem de inicialização com os IDs dos parquímetros de sua zona. O concentrador deve inquirir cada parquímetro conectado à rede e repassar ao MGVE o *status* de funcionamento de cada um deles. Em seguida o SGVE envia para o concentrador o valor da tarifa por cada hora de estacionamento definida para cada período do dia. Após receber a confirmação de que as tarifas foram atualizadas, o MGVE marca o concentrador como inicializado no sistema.

O concentrador deve informar ao MGVE por meio de uma mensagem de atualização de *status* sempre que um parquímetro de sua zona alterar o seu *status*.

Periodicamente, o MGVE recebe uma mensagem do concentrador de *keep alive*. O concentrador deve responder em um tempo máximo especificado pelo sistema a mensagem de *keep alive* para garantir que não ocorreu perda de comunicação. Ao receber a mensagem de *keep alive*, o MGVE zera o contador de perda de comunicação e marca estado atual do concentrador com comunicação ok. Caso o concentrador não envie no tempo especificado para a mensagem de *keep alive*, o MGVE incrementa o contador de perda de comunicação e marca estado atual do concentrador como falha de comunicação. Após três tentativas de comunicação sem sucesso o MGVE considera que ocorreu perda de comunicação e altera o *status* atual do concentrador para sem comunicação e, em seguida, informa ao operador que o concentrador em questão está sem comunicação.

Sempre que o *firmware* do concentrador ou do parquímetro for atualizado, o MGVE pode enviar o novo *firmware* ao concentrador para que ele seja atualizado. A atualização do

firmware é efetuada por meio de um recurso chamado *bootloader*, o qual permite que parte do programa gravado no microcontrolador seja alterado com o sistema em funcionamento. O *bootloader*, no entanto, não foi implementado no desenvolvimento prático experimental desta dissertação, ficando este estudo e os seus impactos a cargo de estudos futuros.

Cada parquímetro e cada concentrador possui um *chip* relógio (RTC) *onboard*, o qual deve ser periodicamente atualizado pelo MGVE. O concentrador envia uma mensagem de atualização da hora/data e recebe do MGVE a hora/data atualizada.

Para facilitar o gerenciamento das vagas de estacionamento e possibilitar o uso de tarifas diferenciadas por regiões, foi adotado o conceito de zona de estacionamento. Cada concentrador é o gestor de uma zona de estacionamento, que pode ter até 30 parquímetros associados a ele. O espaço geográfico ocupado por uma zona deve ser definido pelo gestor do sistema quando ele for implantado. Mais de um concentrador pode pertencer a mesma zona.

Os casos de uso envolvidos no Controle de Preços devem definir zonas de cobrança, vagas, conjunto de vagas de um concentrador, ponto de visualização e tabelas de preços.

Para cada zona do sistema, o operador deve selecionar as tarifas a serem aplicadas para cada período do dia. O operador seleciona os dias na tabela (dias úteis, sábado ou não úteis). O operador seleciona as tarifas para cada período do dia. O sistema registra a alteração e atualiza a tabela de tarifas. O sistema deve verificar quais os concentradores pertencem a zona que sofreu alteração na tarifa e enviar para eles as novas tarifas. O concentrador deve encaminhar as novas tarifas aos concentradores e então responder com uma mensagem de reconhecimento informando ao sistema que as tarifas foram atualizadas.

O Ponto de Visualização é a referência para o Módulo de Localização de Vagas de Estacionamento fornecer ao utente a localização das vagas disponíveis em uma determinada rua ou região. Consiste de uma posição com latitude e longitude, a qual servirá para informar a quantidade de vagas livres ao seu redor.

Para definir um ponto de visualização, o Administrador de Vagas requisita ao MGVE a definição de um novo ponto e visualização, informando a sua latitude e longitude, endereço

aproximado e raio (distância máxima para uma vaga ser associada ao ponto de visualização). O sistema verifica se não existe ponto de visualização nesta latitude e longitude e, estando tudo ok, o sistema registra a novo ponto de visualização, o operador, data e hora.

Para remover um ponto de visualização o Administrador de Vagas seleciona um ponto de visualização e solicita a sua remoção. A seleção pode ser por seu ID ou por sua posição. O sistema verifica se o ponto de visualização existe e se nenhuma vaga ativa referencia o ponto de visualização. Em seguida o marca como inativo, registra operador, data e hora da operação.

Para consultar um ponto de visualização, o Administrador de Vagas recupera as informações de um ponto de visualização por sua posição (latitude e longitude ou endereço aproximado). O sistema verifica se o ponto de visualização existe e recupera todas as informações. O Administrador de Vagas pode requisitar a alteração de quaisquer dos campos de um ponto de visualização. O sistema verifica se não existe ponto e visualização nesta latitude e longitude e então registra as alterações, registra operador, data e hora da operação.

Para definir uma vaga de estacionamento, o Administrador de Vagas requisita a definição de uma vaga informando sua latitude e longitude, endereço e o ponto de visualização associado, já definido previamente. O sistema verifica se não existe vaga nesta latitude e longitude e se o ponto de visualização associado está a uma distância menor que o valor estabelecido no raio do ponto de visualização da localização da vaga. Estando tudo ok, o sistema registra a nova vaga, o operador e a data e hora. Em caso contrário, o sistema rejeita a operação.

Para remover uma vaga do sistema, o Administrador de Vagas requisita a sua remoção. O MGVE verifica se a vaga está cadastrada e efetua a sua remoção. O Administrador de Vagas pode consultar se uma vaga está cadastrada no sistema por sua posição (latitude e longitude ou endereço aproximado). Da mesma forma, o Administrador de Vagas pode solicitar a alteração de quaisquer um dos campos de identificação de uma vaga de estacionamento.

Para gerenciar as vagas de estacionamento é necessário cadastrar os concentradores. O Administrador de Vagas requisita a definição de um concentrador. Um concentrador reúne o conjunto de vagas que são gerenciadas por ele. É preciso informar a identificação do

concentrador, a sua localização, o seu ID, a zona a qual pertence o concentrador, que deve ter sido previamente cadastrada. Além disso, é preciso informar o número de parquímetros deste conjunto (de 1 até 30 - máximo de um concentrador) e de vagas associadas. Cada parquímetro tem associado a ele duas vagas já definidas, além dos seus endereços de comunicação, ou seja, seu dois IDs, um para cada vaga. O sistema verifica se não existe nenhum concentrador com o mesa identificação ou a mesma localização e, neste caso, registra a definição do concentrador.

Para remover um concentrador do MGVE, o Administrador de Vagas seleciona o concentrador a remover por seu ID ou posição (latitude e longitude). O sistema verifica se o concentrador existe e se nenhuma vaga ativa pertence o conjunto de vagas do concentrador e, neste caso, marca como inativo, registra operador, data e hora da operação. O Administrador de Vagas pode consultar concentrador. O Administrador de Vagas recupera as informações de um concentrador por sua identificação ou posição (latitude e longitude ou endereço aproximado). O sistema verifica se o concentrador existe e, neste caso, recupera todas as informações sobre a sua identificação.

É possível alterar o conjunto de vagas associadas a um concentrador. O administrador pode requisitar a alteração de quaisquer parquímetros ou conjunto de vagas. O sistema verifica se não existe nenhum outro concentrador associado a este conjunto de vagas e, neste caso, registra a alteração do conjunto de vagas. O sistema registra operador, data e hora da operação.

As vagas de estacionamento são divididas em zonas. Para criar uma zona o Administrador de Vagas define as posições geográficas (vértice superior esquerdo e inferior direito). do retângulo que estabelece uma zona. O sistema verifica se não existe nenhuma outra zona sobreposta. Neste caso, o sistema registra zona.

Toda zona, quando criada possui três tabelas de cobrança. São elas: tabela de dia úteis; tabela de dias não úteis; tabela de sábado. Todas elas com 24 tarifas de horário e preço, sendo uma tarifa para cada hora do período de expediente. Cada zona tem associada a ela as vagas que estiverem dentro do seu perímetro geográfico. Para remover uma zona, o Administrador de Vagas seleciona uma zona a remover por sua posição. O sistema verifica se zona existe e se

nenhum conjunto de vagas ativo pertence a zona. Em seguida a zona é marcada como inativa. O sistema registra operador, data e hora da operação.

O Administrador de Vagas pode consultar informações sobre uma zona e recuperar as informações de uma zona por sua posição. O sistema verifica se a zona existe e, neste caso, recupera todas as informações.

O Administrador de Vagas pode solicitar a alteração da posição de uma zona. O sistema verifica se não existe nenhuma outra zona sobreposta e, neste caso, registra a alteração do conjunto de vagas. O sistema registra operador, data e hora da operação.

Os casos de uso do Controle do Ocupação de Vagas envolvem operações de controle de ocupação das vagas de estacionamento. São elas: tratar transição de estado; atualizar faturamento da vaga; liberar vaga; identificar pagamento não efetuado; liberar fisicamente uma vaga; manter histórico de ocupação; ocupar fisicamente uma vaga; ocupar vaga.

Sempre que o *status* de um parquímetro for alterado, o concentrador deve enviar uma mensagem de mudança de *status* para o MVGE, com os dados da Tabela 4.1. Com relação ao controle de ocupação das vagas, os *status* relacionados aos parquímetros são: em operação com vaga ocupada; operação com vaga ocupada e pagamento efetuado; em operação com vaga livre; sem comunicação; em falha; em operação com indicativo de fraude.

O *status* em operação com vaga ocupada deve ser ativado sempre que um carro for estacionado e detectado pelo sensor de ultrassom. O *status* em operação com vaga ocupada e pagamento efetuado deve ser ativado sempre que o carro estiver estacionado na vaga e ter sido inserido créditos de estacionamento no parquímetro. O *status* em operação com vaga livre é ativado quando o carro deixa a vaga de estacionamento. O *status* em operação com indicação de fraude é ativado quando o sensor detecta um carro estacionado na vaga, mas não foram inseridos créditos de estacionamento no parquímetro.

Ao receber o novo *status* do parquímetro, o MGVE deve atualizar o *status* da vaga em questão. Os *status* relacionados a vaga são: vaga liberada, vaga ocupada, vaga ocupada

fisicamente e vaga liberada fisicamente. Quando uma vaga passa do *status* de ocupada para liberada, o MGVE deve creditar o valor pago pelo estacionamento ao faturamento da vaga.

Para que uma vaga seja liberada no MGVE, o utente deve aproximar seu cartão de estacionamento no parquímetro requisitando estorno do pagamento dos créditos não utilizados. Esse procedimento é efetuado na hora em que o utente vai sair do estacionamento. O parquímetro verifica se este foi o cartão que efetuou o pagamento constante da pilha de saldo de créditos e devolve o saldo de créditos remanescentes para o cartão do utente. Se este for o pagamento corrente, a vaga passa para o estado de livre. O parquímetro envia uma mensagem de transição de estado para o concentrador informando a mudança de *status* com o valor pago e o concentrador repassa a mensagem para o MGVE. O concentrador envia mensagem de liberação de vaga com o valor da carga, data e hora. O sistema atualiza o faturamento da vaga e marca a vaga como liberada.

Periodicamente o sistema verifica todas as vagas que estão fisicamente ocupadas e sem pagamento efetuado a mais de N minutos (15 minutos). O sistema marca as vagas nesta situação como em suspeita de fraude e alerta ao Administrador de Vagas da necessidade de fiscalização da vaga.

Para que a vaga seja dada como liberada fisicamente é necessário que o sensor de ultrassom não detecte um carro estacionado na vaga. O utente retira seu carro frente uma vaga com parquímetro com sensor de ocupação. O *status* do parquímetro muda para em operação com vaga livre e o concentrador envia uma mensagem de liberação de vaga para o MGVE. O sistema registra a vaga como fisicamente livre, data e hora.

Para que uma vaga seja dada como fisicamente ocupada, o sensor de ultrassom deve detectar um carro estacionado na vaga. O utente estaciona seu carro na vaga. O parquímetro muda o seu *status* para em operação com vaga ocupada e o concentrador envia a mensagem para o MGVE informando da ocupação física da vaga. O sistema marca a vaga como fisicamente ocupada.

Para que a vaga ocupada esteja em situação regular, é preciso que o utente, após ter estacionado o carro insira os créditos de estacionamento no parquímetro. Para isso, basta aproximar o cartão do leitor RFID do parquímetro. O parquímetro retira créditos de estacionamento do cartão e altera o seu *status* para vaga ocupada com pagamento efetuado. O concentrador envia uma mensagem para o MGVE com a informação de que a vaga está regularmente ocupada. O sistema marca a vaga como ocupada.

A partir do tempo em que o *status* de parquímetro se manteve em operação com vaga livre, o MGVE efetua um cálculo estatístico do tempo médio livre de cada vaga para cada período do dia. De tempos em tempos, a ser definido no sistema, o MGVE verifica as vagas que estão com o *status* de livre e verifica se o tempo em que ela se encontra ociosa é maior que a média calculada para aquele horário do dia. Em caso positivo, o MGVE deve informar ao Administrador de Vagas sobre a indicação de fraude na vaga em questão. O Administrador de Vagas deve então notificar ao fiscal da área que existe uma indicação de fraude que precisa ser verificada.

Os parquímetros que possuem sensor de presença de carro verificam se existe um carro estacionado por mais de 15 minutos sem que créditos de estacionamento tenham sido inseridos no parquímetro. Se confirmado, o *status* do parquímetro muda para em operação com indicação de fraude. O concentrador reporta a alteração de *status* do parquímetro para o MGVE, que verifica se tempo médio livre da vaga naquela horário está dentro do padrão especificado para o horário. Em caso positivo, o Administrador de Vagas é notificado pelo MGVE, o qual deve tomar as ações pertinentes à confirmação ou não da fraude. Após ter sido confirmada a indicação de fraude, o Fiscal deve reportar ao Administrador de Vagas que desativará o sinalizador de indicação de fraude para aquela vaga.

O cálculo da média de ocupação consiste em, periodicamente, o sistema selecionar o conjunto de vagas para as quais são calculadas médias de tempo de ocupação no período. O sistema calcula a nova média e o desvio padrão de ocupação de cada vaga selecionada por faixa de horário, e marca a vaga como “com média”. Periodicamente, ou a cada 15 minutos, o sistema verifica para cada vaga livre que esteja marcada “com média”, sem indicação de falha e sem uma ação de fiscalização designada em aberto, se o tempo que ela está livre é superior ao

tempo médio para o tempo de ocupação acrescido de um valor de tolerância (2x o desvio padrão) (dia útil/não útil e faixa de horário). Cada vaga nesta situação é marcada como em fiscalização. O sistema constrói uma lista de todas as vagas em fiscalização devido a desvios de ocupação média, com a localização da vaga e alerta ao Administrador de Vagas informando que aquela vaga necessita de uma ação de fiscalização.

Ao concluir uma ação de fiscalização, o Fiscal de Área informa ao Administrador de Vagas o resultado da ação que pode ser: o usuário pagou o estacionamento antes de ele chegar ao local; fraude e o usuário foi multado; falha - existe provavelmente um defeito no equipamento e deve ser aberta uma OS de manutenção. O sistema registra a vaga com indicação de falha.

4.6. Serviço de Localização de Vagas Via Internet (SLVVI)

O SLVVI foi desenvolvido para ser executado em dispositivo móvel, embarcado com o sistema operacional Android, com conexão à internet. O SLVVI, ao ser executado mostra o mapa da região indicando a posição geográfica do dispositivo. Em seguida o SLVVI acessa o SGVE, o qual verifica a sua localização geográfica, consulta o banco de dados e retorna os *status* das vagas em um raio de aproximadamente 1 km.

Em cada rua é cadastrado no SGVE um ponto de visualização, o qual é utilizado como referência para a visualização dos *status* das vagas naquela rua. No mapa, em cada ponto de visualização, aparece um ícone, cuja cor depende do número de vagas disponíveis naquela rua. Vermelho (sem vagas); amarelo (até 3 vagas) e azul (acima de três vagas).

4.7. Integração do Sistema *e-SIGA*

Após o desenvolvimento experimental de cada uma das partes integrantes do sistema *e-SIGA*, foi integrado o sistema apresentado na Figura 4.4.

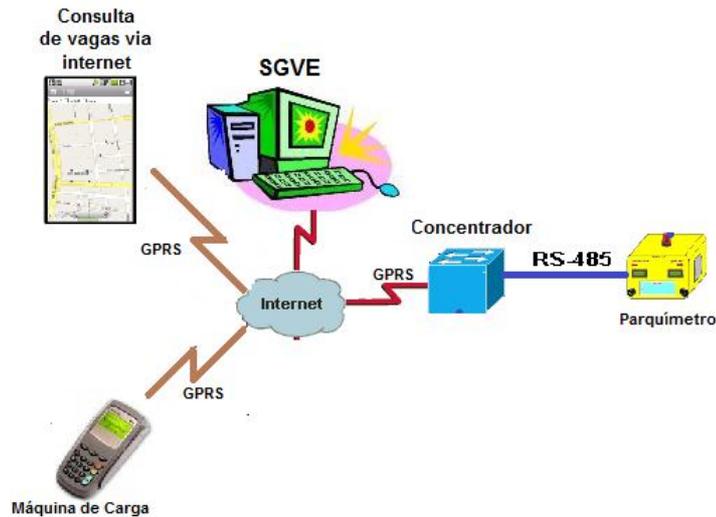


Figura 4.4: Desenvolvimento prático experimental do *EXIGE*.

O objetivo do sistema prático experimental apresentado na Figura 4.4 foi a comprovação, em laboratório, da prova de conceito do modelo do sistema inteligente de gerência de vagas de estacionamento automotivo em vias públicas proposto por esta dissertação.

4.7.1. Rede Concentrador - Parquímetros

A rede que conecta os parquímetros ao concentrador foi implementada na camada de comunicação do modelo eOSI pelo protocolo ModBus, operando sobre o padrão RS485.

As mensagens ModBus modeladas que o concentrador pode enviar para os parquímetros via protocolo ModBus são: solicitação de *status* do parquímetro; lista de horários e valor/hora de cada período; hora e data para sincronizar o horário dos parquímetros; solicitação do *log* de utilização dos parquímetros. A Tabela 4.4 mostra as funções modeladas e quais foram implementadas no desenvolvimento experimental.

Tabela 4.4: Funções modeladas *versus* funções implementadas

Função modelada	Função implementada
Solicitação de <i>status</i> do parquímetro	Sim
Lista de horários e valor/hora de cada período	Sim
Hora e data para sincronizar o horário dos parquímetros	Não
<i>Log</i> de utilização dos parquímetros	Não

Um *software* monitor serial foi desenvolvido em linguagem de alto nível para auxiliar os testes de validação da rede ModBus que interliga os parquímetros ao concentrador. A Figura 4.5 mostra a tela inicial do *Software Monitor Serial v1.0*.

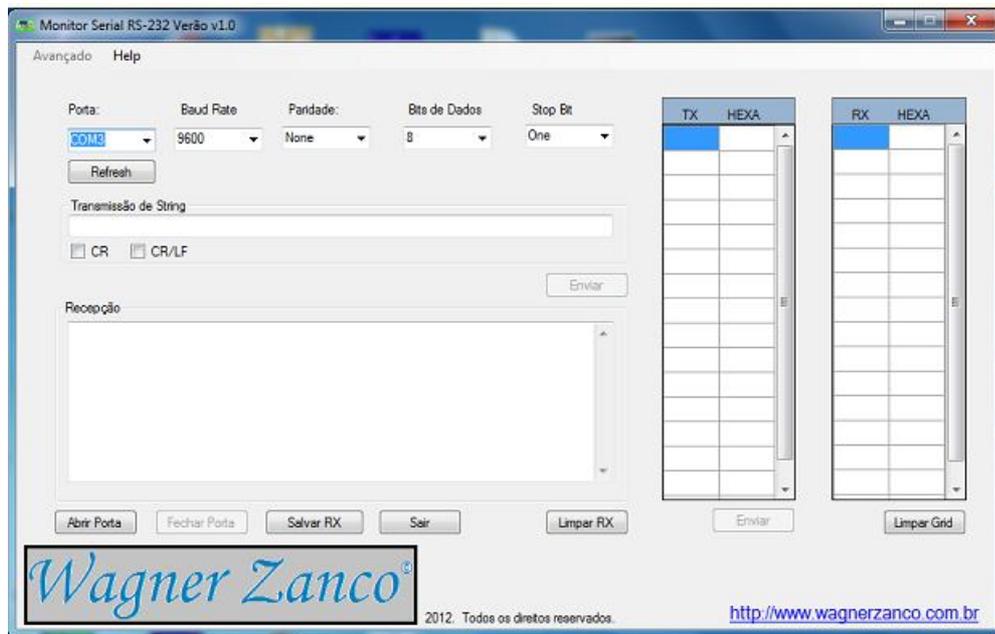


Figura 4.5: *Software Monitor Serial*

O *software* monitor serial pode enviar e receber dados pela porta serial. Ele também pode operar como mestre ou escravo em comunicações via protocolo ModBus.

A Figura 4.6 mostra a placa do parquímetro desenvolvida para o *e-SIGA*, juntamente com o leitor de RFID e com o LCD.



Figura 4.6: Placa do parquímetro.

O leitor de RFID utilizado no desenvolvimento prático experimental foi o MF-CLASSIC, desenvolvido pela empresa RFID Cartões Comerciais Ltda. A comunicação com o leitor é efetuada por meio de uma comunicação serial. Os comandos utilizados para a comunicação com o leitor para acesso ao cartão são mostrados na Figura 4.7.

Comando	Descrição	Resposta ok	Erros
C	Leitura contínua do ID de cartões	ID com 4 a 7 bytes	
S	Seleciona um cartão para operações	ID com 4 a 7 bytes	N
M	Seleciona cartão/lista de cartões para operações	ID com 4 a 7 bytes	N
L	Login(Autentica): L+ setor+ tipo chave + chave	L	N, F, E
R	Read – Lê um bloco(00 - 3F= cartao 1k)/(00-FF=cartao4k)	16 bytes Hexa	N, F, I
W	Write – Escreve em um bloco (00h a FFh)	16 bytes Hexa	X, F, U, N
RV	Read Value – Lê um bloco de valor(00h a FFh)	4 bytes Hexa	N, F, I
WV	Write Value – Escreve valor em um bloco (00h a FFh)	4 bytes Hexa	X, F, U, N, I
RS	Leitura direta de 1 ou 16(modos binário) blocos de 16 bytes	16 ou 256 hexa	N, F, I
WS	Escrita direta de 1 a 16 blocos de 16 bytes	Soma16 + W	X, F, U, N
+	Incrementa Bloco de valor	4 bytes Hexa	X, F, N, I
-	Decrementa Bloco de valor	4 bytes Hexa	X, F, N, I, E
=	Copia Bloco de valor de (00h a FFh) para (00h a FFh)	4 bytes Hexa	X, F, N, I

Figura 4.7: Comandos de cartão do leitor MF-CLASSIC.

A Figura 4.8 mostra o circuito medidor de distância por ultrassom utilizado para verificar a presença de carro estacionado. O sensor utilizado foi o LV-MaxSonar[®]-EZ1[™]. Esse sensor mede distâncias de 15 cm até aproximadamente 6,4 m. A leitura da distância pode ser efetuada por meio de leitura de tensão, largura de pulso ou saída digital. No desenvolvimento experimental, a forma de leitura escolhida foi a saída digital. O protocolo UART é o protocolo utilizado na comunicação com o sensor de distância.

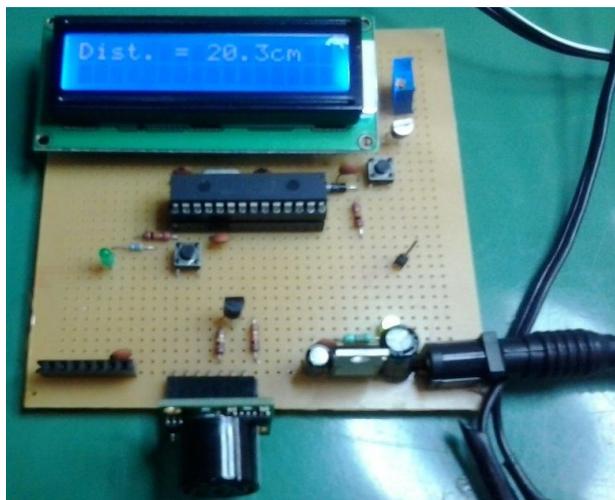


Figura 4.8: Sensor de distância por ultrassom.

A Figura 4.9 mostra a visão *top*, enquanto a Figura 4.10 mostra a visão *bottom* placa do concentrador.



Figura 4.9: Concentrador visão *Top*



Figura 4.10: Concentrador visão *Bottom*

4.7.2. Máquina de Carga e Recarga

O protótipo da máquina de carga e recarga inclui as funcionalidades descritas no item 4.1. Ao ser ligada, a máquina de carga e recarga efetua uma conexão com o SGVE e faz o login na rede. A cada venda de crédito de estacionamento efetuada, a máquina de carga e recarga deve solicitar ao SGVE um código de autorização, o qual irá identificar a venda. O código de autorização é gravado no cartão RFID juntamente com o crédito de estacionamento adquirido. A conexão com o SGVE é feita por meio de um *modem* GPRS. O protocolo da comunicação utilizado foi o HTTP.

A conexão entre a máquina de carga e recarga e o SGVE é feita por meio de uma conexão GPRS, por meio de um *modem embedded hardware*. O modem utilizado foi o SIM340, fabricado pela empresa SIMCOM. A conexão com o microcontrolador PIC32 foi feita por meio de uma comunicação serial UART. A conexão com a internet foi feita por meio de comandos AT enviados para o *modem*, enquanto que a comunicação com o SGVE foi feita

pelo o protocolo HTTP. A **Figura 4.11** mostra a máquina de carga e recarga desenvolvida para o *e-SIGA*.



Figura 4.11: Máquina de carga e recarga

4.7.3. Rede Concentrador - SGVE

No projeto do concentrador foi utilizado o mesmo *modem* utilizado na máquina de carga e recarga, o SIM340. Sendo assim, a conexão entre o concentrador e o SGVE é feita da mesma forma, via conexão GPRS. O concentrador funciona como *gateway*, operando como elo de ligação entre o SGVE e o parquímetro. O concentrador efetua a conexão com o SGVE, recebe dele o comando a ser executado e o repassa para os parquímetros pelo rede ModBus.

O concentrador constitui um diferencial se o sistema *e-SIGA* for comparado com alguns sistemas de gerenciamento de parquímetros em operação analisados por este estudo. O concentrador permite, por exemplo, que outro meio de comunicação possa ser utilizado para conectar os parquímetros sem que seja preciso alterar a estrutura do sistema. Como exemplo podemos citar a fibra óptica, ou rede sem fio.

Capítulo 5 - BENEFÍCIOS E LIMITAÇÕES DO SISTEMA

O desenvolvimento de um sistema inteligente de gerenciamento de vagas de estacionamento automotivo em vias públicas pode contribuir de forma significativa, em vários aspectos, para a implantação de uma política de estacionamento nas pequenas e nas grandes cidades. Um sistema eficaz de gerenciamento pode resultar em uma redução no fluxo de veículos em tráfego e, conseqüentemente, em uma melhora no trânsito na cidade, em especial no centro. Por outro lado, vários são os obstáculos a serem ultrapassados para a sua implantação.

5.1. Benefícios do Novo Modelo

Muitos benefícios poderão ser obtidos a partir da implantação de um sistema inteligente de gerenciamento de estacionamento automotivo em vias públicas. Para o gestor, a vantagem de ter um controle eficaz sobre as vagas de estacionamento disponíveis nas vias públicas da cidade. O utente, por sua vez, pode usufruir de um sistema de cobrança de vagas de estacionamento mais justo, por meio do fracionamento da tarifa, além de um acesso mais rápido às vagas disponíveis. Com um sistema de gerenciamento inteligente é possível programar o sistema de forma que o utente só pague pelo tempo em que ficou com o carro estacionado, prática essa não adotada na maioria das cidades brasileiras, onde predomina a cobrança pelo flanelinha.

Um sistema inteligente de gerência de vagas de estacionamento em vias públicas pode somar esforços na equação de alguns dos principais problemas derivados do modelo atual, como o estacionamento em locais inadequados e o sensível aumento do congestionamento nas ruas e avenidas que advém de uma falta de gerenciamento, ou de um gerenciamento precário do sistema.

Uma dos grandes benefícios do sistema que está sendo aqui proposto é a possibilidade de o utente poder verificar, por meio de um acesso via dispositivo móvel conectado à internet, a localização das vagas disponíveis mais próximas do local de destino. Tal procedimento pode

contribuir para a redução do tráfego local, uma vez que o utente não precisa ficar circulando a procura de vaga, que pode não existir naquela região. A possibilidade de verificação antecipada da vaga pode ter um impacto direto na redução da quantidade de veículos em tráfego no local. Tal facilidade vem de encontro à disponibilização de serviços de acesso de banda larga, e por preços cada vez mais competitivos, pelas diversas operadoras de telefonia celular existentes.

O sistema permitirá um maior controle do processo, cabendo ao gestor definir quais as suas prioridades fazendo, por exemplo, uso da tarifa diferenciada. A tarifa diferenciada permite ao gestor aumentar o valor por hora cobrado em um determinado setor da cidade no qual se deseja reduzir o número de carros estacionados, e priorizar o tráfego. Por outro lado, pode ser cobrada uma tarifa menor em setores onde se deseja priorizar o estacionamento de usuários que venham a fazer conexões com transportes de massa como ônibus, trem, metrô, barcas etc. A aplicação de tarifas diferenciadas para estacionamento vem sendo apontada, e comprovada por meio de modelos matemáticos como uma das mais potentes ferramentas para enfrentar o congestionamento em grandes centros urbanos. Modelos mais recentes levam em conta até mesmo a diferença entre as tarifas cobradas por estacionamentos públicos e privados.

O Sistema *e-SIGA* pode entregar ao gestor, de forma simples e transparente, as estatísticas de utilização das diferentes áreas de estacionamento, possibilitando a aplicação de política de tarifas para diferentes áreas e horários que descongestionem o trânsito nas áreas de interesse do gestor. A possibilidade de otimização do tráfego e maior controle da arrecadação abre possibilidade para que a municipalidade use os recursos arrecadados com o estacionamento em vias públicas para investir no setor de transporte coletivo, ciclovias, melhoramento das calçadas, contribuindo para a melhora do índice da mobilidade urbana como um todo.

Na maioria das cidades brasileiras não se utiliza tecnologia de sistemas embarcados e, tampouco, de redes de telecomunicações em prol do gerenciamento de vagas de estacionamento automotivo em vias públicas. A prática atual, adotada na maior parte das cidades brasileiras, que faz uso do flanelinha e de talões de estacionamento em papel, não atende com eficiência e profissionalismo a crescente demanda por soluções que visam reduzir o impacto do aumento do número de veículos em tráfego nas ruas. Um sistema de

gerenciamento eficaz, por sua vez, pode contribuir para atender a essa demanda de forma efetiva, que pode resultar em uma melhoria do trânsito nas cidades em que ele for implantado.

O principal benefício de um sistema inteligente de gestão de estacionamento em vias públicas como o *e-SIGA* é o de poder influenciar no índice de mobilidade urbana sustentável de uma cidade. A Tabela 5.1 mostra os indicadores que poderão ser influenciados. Dos nove domínios existentes na definição do índice de mobilidade urbana sustentável, sete podem ser influenciados pela implantação de um sistema inteligente de gestão de estacionamento em vias públicas. A comprovação das hipóteses sobre a melhoria desses indicadores, no entanto, não faz parte deste estudo e ficará a cargo de estudos posteriores.

Tabela 5.1: Indicadores da mobilidade urbana sustentável que podem ser influenciados por uma gestão inteligente das vagas de estacionamento automotivo em vias públicas.

Domínio	Tema	Indicadores
Acessibilidade	Acessibilidade aos sistemas de transporte	- Vagas de estacionamento para pessoas com necessidades especiais
Aspectos Ambientais	Controle dos impactos no meio ambiente	- Emissões de CO ₂ - População exposta aos ruídos do tráfego
	Recursos naturais	- Consumo de combustível
Aspectos Sociais	Apoio ao cidadão	- Informação disponível ao cidadão
	Qualidade de vida	- Qualidade de vida
Infraestrutura	Provisão e manutenção da infraestrutura de transporte	- Sinalização viária
Modos Não Motorizados	Redução de viagens	- Tempo de viagem
		- Ações para redução do tráfego motorizado
Planejamento Integrado	Planejamento e controle do uso e ocupação do solo	- Ocupações irregulares

5.2. Obstáculos Para a Implantação do Sistema

A infraestrutura necessária para a implantação do sistema proposto constitui um obstáculo a ser ultrapassado para a sua implantação. Os parquímetros precisam de energia elétrica para

funcionar. Esta, em primeira mão, deveria ser provida pela rede de energia elétrica oferecida pela concessionária de energia local. No, entanto, dependendo do setor da cidade onde o sistema é implantado, pode não existir infraestrutura de energia elétrica para a alimentação dos parquímetros. Por outro lado, levar energia elétrica aos locais onde os parquímetros estão instalados pode requerer um acordo entre a prefeitura e a companhia de energia elétrica. Em caso de impedimento da firmação deste acordo, é necessário prover energia para o parquímetro por meio de baterias. A utilização de baterias incorre em monitoramento constante do nível de carga, e a consequente troca em caso de a bateria se descarregar. Esse procedimento sugere um aumento no custo de manutenção do sistema, que pode vir a ser um obstáculo para a sua implantação.

Outro obstáculo com relação a infraestrutura a ser ultrapassado é a necessidade de se criar caminhos físicos entre os parquímetros para o cabeamento da rede que os conecta. A criação de caminhos físicos pode requerer a necessidade de intervenção na calçada para a instalação de tubos subterrâneos por onde deverão passar os cabos de rede que irão conectar fisicamente os parquímetros. No caso de impedimento de se criar um caminho físico entre os parquímetros, pode ser utilizada tecnologia de rede sem fio para conectar os parquímetros e evitar que a calçada sofra algum tipo de intervenção.

O vandalismo pode ser um obstáculo para a implementação do sistema. Nesse caso, poderão ser utilizados materiais reforçados na construção do encapsulamento do parquímetro como, por exemplo, aço inox ou fibra de carbono.

A não cobertura de telefonia celular na área pode ser um obstáculo a ser ultrapassado na implantação do sistema, uma vez que, no desenvolvimento prático experimental do modelo proposto, o sistema utiliza um *modem* GPRS para a comunicação do concentrador com o SGVE. Caso não haja cobertura de telefonia celular, deve ser pensada outra tecnologia em substituição ao GPRS. Conexão *Wireless* via rádio ou por fibra óptica podem ser opções a serem estudadas.

A implantação de um sistema inteligente de gestão de estacionamentos em vias públicas requer uma alteração na legislação local, a qual deve ser enviada pelo prefeito à câmara de vereadores como projeto de lei e, por ela, aprovada.

Conclusão

A modelagem e o desenvolvimento prático experimental do sistema inteligente de gerência de vagas de estacionamento em vias públicas proposto por esta dissertação, o *e-SIGA*, foi realizado no Laboratório de Comunicações Ópticas da Universidade Federal Fluminense (LaCOp). O sistema foi desenvolvido como proposta para uma nova geração de parquímetros, gerenciados remotamente, com vistas a ocupar uma lacuna na gestão inteligente de estacionamento automotivo em vias públicas, como parte de uma política mais ampla, que visa a melhoria do transporte e da mobilidade urbana sustentável. Os testes de integração mostraram que é possível gerenciar remotamente os parquímetros com segurança e efetividade.

Um sistema com o nível de inteligência do *e-SIGA* pode contribuir, em sinergia com outras soluções, para uma melhora no tráfego automotivo no centro das grandes cidades, e a consequente melhora do índice de mobilidade urbana sustentável. A possibilidade de o utente poder localizar vagas via internet é um diferencial com relação aos sistemas de gerência de segunda geração implantados em algumas cidades brasileiras. A cobrança da tarifa diferenciada em partes distintas da cidade com vistas a reduzir o tráfego em alguns setores, em conjunto outras características, como a cobrança mais justa pelo uso do solo para estacionamento, são avanços que abrangem diversos setores, como a própria mobilidade urbana e a responsabilidade social.

Vários estudos podem ser desenvolvidos a partir do modelo proposto. Se um concentrador apresentar defeito, os parquímetros conectados a sua rede não poderão mais se comunicar com o SGVE até que o defeito tenha sido concertado. Embora os parquímetros possam operar na modalidade *stand alone*, uma falha no concentrado pode trazer efeitos colaterais, a médio prazo, para o bom funcionamento do sistema. O efeito da redundância de concentradores com a instalação de um segundo concentrador é um tema que pode ser explorado por futuros estudos como desdobramento deste trabalho.

Outro tema importante a ser abordado por futuros estudos é a comprovação da alteração dos indicadores de mobilidade urbana sustentável citados no capítulo 5, após a implantação do *e-SIGA* em uma cidade modelo.

A integração nos parquímetros do sistema *e-SIGA* de um módulo de carga para carros elétricos pode ser um tema a ser abordado por estudos futuros. Neste caso, o sistema atenderia a duas demandas, a de gestão do estacionamento e a carga de carros elétricos.

A implantação de um sistema inteligente de gerência de vagas de estacionamento em vias públicas como *e-SIGA* pode ser o ponto de partida para a criação de uma política eficaz de gestão de estacionamento, com desdobramentos em vários seguimentos da sociedade, além sua influência na mobilidade urbana sustentável.

Referências

ABD-EI-BAR, M. "*Design And Analysis of Reliable And Faul-tolernt Computer Systems*". London: Ed. Imperial College Press, 2007.

AANDERSON, T.; LEE, P. A. "*Faul tolerance - principles and practice*". New Jersey: Prentice-Hall, 1981.

ÁTILA A. S.; JULIANO C. D.; GIOVANNI M. H. "As cidades digitais no mapa do Brasil - uma rota para a inclusão social". Ministério das Comunicações. Distrito Federal: 2006.

ATUAL, REVISTA ELETRÔNICA. "Entendendo o Protocolo Elétrico RS-485". <http://www.mecatronicaatual.com.br/secoes/leitura/280>. Último acesso em 31/01/2013. São Paulo: Atual, 2007.

Avizienis, A.; Kopetz, H.; Laprie, J.-C. "*The Evolution of Fault-Tolerant Computing*". New York: Ed. Springer-Verlag, 1987.

BULL, A. "*Medidas para el control de la congestión vial urbana actuando sobre la demanda*". Santiago de Chile: 2002.

CALISKAN, M. D. GRAUPNER; MAUVE, M. "*Decentralized discovery of free parking places*". Proceedings of the 3rd ACM International Workshop on Vehicular Ad Hoc Networks. Los Angeles: 2006.

CARVALHO, C. H. R.. "Emissões Relativas de Poluentes do Transporte Motorizado de Passageiros nos Grandes Centros Urbanos Brasileiros". Brasília: IPEA, 2011.

COELHO, E. C. "Dissertação de Mestrado - Avaliação do Nível de Congestionamento em Vias Arteriais Com a Utilização da Micro Simulação". Rio de Janeiro: UFRJ, 2009.

COSTA M. C. "Tese de Doutorado - Um índice de Mobilidade Urbana Sustentável". Escola de Engenharia de São Carlos da Universidade de São Paulo. São Carlos: 2008.

COSTA, M. S. "Dissertação de Mestrado - Mobilidade Urbana Sustentável: Um Estudo Comparativo e as Bases de um Sistema de Gestão Para Brasil e Portugal". São Paulo: USP São Carlos, 2003.

CYPRIANO, L.B. "Z80 VOLUME I Hardware". São Paulo: Érica, 1984.

DA SILVA, M. A. "Dissertação de Mestrado - Análise de Uma Rede Adequada Para Monitoração e Controle de energia na Automação Predial". Distrito Federal: Universidade de Brasília, 2005.

EUGENI, E. Artigo: "Interface RS232/RS485 - A vantagem das linhas equilibradas". Revista Elektor. São Paulo: 2007.

FILOSA, G. "*Master's Thesis - Carsharing: Establishing its Role in the Parking Demand Management. Urban and Environmental Policy and Planning Tufts University*". USA: 2006.

FINKEL, V. S. et al. "Instrumentação Industrial". 2. Ed. Rio de Janeiro, RJ. Ed. InterCiência, 2006.

GAMA, R. e FERNADES, R. "A Cidade Digital vs a Cidade Inteligente: Estratégias de Desenvolvimento Sócio Econômico e/ou de Marketing Territorial". 2º Congresso Luso-brasileiro para o Planejamento Urbano, Regional, Integrado e Sustentável. Universidade do Minho. Braga: 2006.

GAMA, R. e FERNADES, R. "A Cidade Digital vs a Cidade Inteligente: Estratégias de Desenvolvimento Sócio Econômico e/ou de Marketing Territorial". 2º Congresso Luso-brasileiro para o Planejamento Urbano, Regional, Integrado e Sustentável. Universidade do Minho. Braga: 2006.

INCT O. M. "Crescimento da frota de automóveis e motocicletas nas metrópoles brasileiras 2001/2011". Baixado de http://observatoriodasmetrololes.net/download/relatorio_automotos.pdf. Último acesso em 07/02/2013. Brasília: Ministério dos Transportes, 2013.

JALOTE, P. "*Fault tolerance in distributed systems*". New Jersey: Ed. Prentice Hall, 1994.

JOHNSON, B.W. "*Fault-Tolerant Microprocessor-Based Systems*". IEEE Micro Magazine, Vol.4, 1984.

JUNIOR, A. D. A. "Tese de Doutorado - Método de Coordenação Semáfora Para Corredores de Transportes Coletivos". Rio de Janeiro: UFRJ, 2007.

KOREN, I.; KRISHNA, C. M. "*Fault tolerant Systems*". California: Ed. Morgan Kaufmann, 2007.

LAPRIE, J. C. "*Dependable computing and fault-tolerance: concepts and terminology*". International Symposium on Fault Tolerant Computing, 1985.

LEMOS, A. A "Cidade-Ciborgue – a cidade na cibercultura. In: LEMOS, A. (Org.). *Cibercultura II - Ciberurbe: a cidade na sociedade da informação*". Rio de Janeiro: E-papers, 2005.

MARTINS, R. T. "Dissertação de Mestrado - Arquiteturas e Ferramentas Para Gestão de Redes e Sistemas". Aveiro: Universidade de Aveiro, 2001.

MASSA, M. B. A. "*Programming Embedded Systems*". [S.l.]: O'Reilly, 2006. 304 p.

MAXIM. "*Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers, Data Sheet: 19-0122; Rev 8*". USA: Maxim, 2003.

MICROCHIP TECHNOLOGY INC. "*PIC32MX Reference Manual: DS61113C - Section 2 MCU*". USA: MICROCHIP, 2008.

MICROCHIP TECHNOLOGY INC_1. "*PIC32MX3XX/4XX: Data Sheet DS61143G - High-Performance, General Purpose and USB 32-bit Flash Microcontrollers*". USA: MICROCHIP, 2010.

MICROCHIP TECHNOLOGY INC_2. "*PIC32MX Reference Manual: DS61112F - Section 6 OSCILLATORS*". USA: MICROCHIP, 2010.

MINISTÉRIO DAS CIDADES. "Caderno 6 - Política Nacional de Mobilidade Urbana Sustentável". Distrito Federal: 2004.

MIRANDA, H. F. "Dissertação de Mestrado - Mobilidade Urbana Sustentável e o Caso de Curitiba". Escola de Engenharia de São Carlos da Universidade de São Paulo. São Carlos: 2010.

MODBUS O. "*ModBus Application Protocol Specification v1.1b. MODBUS ORGANIZATION*". USA: Hopkinton, 2006.

MODBUS P. "*Modicon ModBus Plus Network Planning and Installation Guide - v6.0. MODBUS ORGANIZATION*". USA: Hopkinton, 2004.

MOKARZEL, M. P. e CARNEIRO, K. P. M. "*Internet Embedded TCP/IP para Microcontroladores*". São Paulo: Érica, 2004.

MORAIS, A. H. F. "Dissertação de Mestrado - eISO: Um Modelo Para Sistemas Embarcados Tolerantes a Falhas". Natal: UFRN, 2009.

MORESI, E. "Metodologia da Pesquisa". Distrito Federal: Universidade Católica de Brasília, 2003.

NEMOTO, M. C. M. O. "Tese de Doutorado - Inovação Tecnológica: Um Estudo Exploratório de Adoção do RFID (Identificação por Radiofrequência) e Redes de inovação Internacional". São Paulo: USP, 2009.

NETO, A. C. "Dissertação de Mestrado - Proposta de Referência Para o Desenvolvimento de um Software Com Foco na Certificação MPS.br". São Paulo: USP São Carlos, 2008.

NEVES, D. M. C. M. "Dissertação de Mestrado - Sistema de gestão de parques de estacionamento exteriores – serviços de administração". UNIVERSIDADE DE TRÁS-OS-MO_TES E ALTO DOURO. Vila Real, 2009.

NFX Semiconductors. "*Product data sheet 001053: MF1ICS50 - Functional specification*". Rev. 5.3 — 29 January 2008. Baixado de http://www.acs.com.hk/drivers/eng/TDS_Mifare1K.pdf. Último acesso em 22/03/2013.

OLIVEIRA, A. R. M. "Dissertação de Mestrado - Análise das principais Estratégias Para a Redução da Emissão de Poluentes atmosféricos Por Veículos Automotores". Rio de Janeiro: IME, 2011.

PEREIRA, W. H. "Monografia - Sistemas Digitais de Controle Distribuído". Universidade Federal de Ouro Preto. Minas Gerais: 2009.

PORTAL BRASIL. "Gastos Com Vítimas de Acidente de Transito". Baixado de: <http://www.brasil.gov.br/noticias/arquivos/2011/06/20/saude-gasta-mais-de-r-185-mi-com-internacoes-de-vitimas-de-acidentes-de-transito-afirma-padilha>. Último acesso em 23/02/2-13. Brasília: Governo Federal, 2011.

PRADHAN, D. K. "*Fault-Tolerant System Design*". New Jersey: Prentice Hall, 1996.

PRESSMAN, R. "Engenharia de Software". São Paulo: McGraw-Hill Company, 2006.

REINHOLG, I. R. "Dissertação de Mestrado - Qualidade em Transporte Coletivo Urbano Por Ônibus". Rio de Janeiro: IME, 1996.

SCHUSTER, Edmund W et al, Global RFID: "*The Value of the EPCglobal Network for Supply Chain Management*". Cambridge: Springer, 2007.

STEVE, G. "Construa seu Próprio Microcomputador Z-80". São Paulo: McGraw-Hill, 1984.

SVERZUT, J. H. "Redes GSM, EDGE e UMTS: Evolução e Caminho da Quarta geração (4G), 3ª edição". São Paulo: Editora Érica, 2001.

VASCONCELOS, E. A. "O que é transito". São Paulo: Editora Brasiliense, 1985.

WEBER, T.; JANSCH PÔRTO, I.; WEBER, R. "Fundamentos de Tolerância a Falhas". Apostila preparada para o IX JAI – Jornada de Atualização em Informática no X Congresso da Sociedade Brasileira de Computação. Vitória, 1990.

ZANCO, W. S. "Microcontroladores PIC®: *Técnicas de software e hardware para projetos de circuitos eletrônicos*. 2a. ed". São Paulo: Érica, 2009.

ZANCO, W. S. "Microcontroladores PIC16F628A/648A: *uma abordagem prática e objetiva*. 2a. ed". São Paulo: Érica, 2008.

ZANCO, W. S. "Microcontroladores PIC18 Com linguagem C: *uma abordagem prática e objetiva*". São Paulo: Érica, 2010.