



Universidade Estadual de Campinas  
Faculdade de Engenharia Elétrica e de Computação  
Departamento de Semicondutores, Instrumentos e Fotônica

## **Sistema embarcado para monitoramento do conforto em transporte público.**

Candidato: Juan Camilo Castellanos Rodriguez  
Orientador: Prof. Dr. Fabiano Fruett

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Eletrônica, Microeletrônica e Optoeletrônica.

Comissão Examinadora:

Prof. Dr. Fabiano Fruett  
DSIF — Departamento de Semicondutores, Instrumentos e Fotônica  
Faculdade de Engenharia Elétrica e de Computação - UNICAMP

Prof. Dr. Antonio Celso Fonseca de Arruda  
DEP — Departamento de Engenharia de Petróleo  
Faculdade de Engenharia Mecânica - UNICAMP

Prof. Dr. Yaro Burian Junior  
DMCSI — Departamento de Máquinas, Componentes e Sistemas Inteligentes  
Faculdade de Engenharia Elétrica e de Computação - UNICAMP

Campinas, SP  
2011

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

C274s      Castellanos Rodriguez, Juan Camilo  
Sistema embarcado para monitoramento do conforto  
em transporte público. / Juan Camilo Castellanos  
Rodriguez. --Campinas, SP: [s.n.], 2011.

Orientador: Fabiano Fruett.  
Dissertação de Mestrado - Universidade Estadual de  
Campinas, Faculdade de Engenharia Elétrica e de  
Computação.

1. Aceleração (Mecânica). 2. Conforto humano. 3.  
Sistema de Posicionamento Global. 4. Transporte  
urbano. I. Fruett, Fabiano. II. Universidade Estadual de  
Campinas. Faculdade de Engenharia Elétrica e de  
Computação. III. Título.

Título em Inglês: Embedded system to monitoring the comfort in public transportation

Palavras-chave em Inglês: Acceleration (Mechanical), Human comfort, Global

Positioning System, Urban transportation

Área de concentração: Eletrônica, Microeletrônica e Optoeletrônica

Titulação: Mestre em Engenharia Elétrica

Banca examinadora: Antonio Celso Fonseca de Arruda, Yaro Burian Junior

Data da defesa: 23-11-2011

Programa de Pós Graduação: Engenharia Elétrica



**COMISSÃO JULGADORA - TESE DE MESTRADO**

**Candidato:** Juan Camilo Castellanos Rodriguez

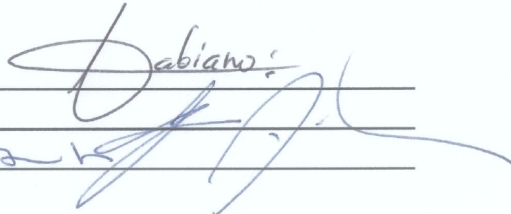
**Data da Defesa:** 23 de novembro de 2011

**Título da Tese:** "Sistema embarcado para monitoramento do conforto em transporte público"

Prof. Dr. Fabiano Fruett (Presidente): \_\_\_\_\_

Prof. Dr. Antonio Celso Fonseca de Arruda: \_\_\_\_\_

Prof. Dr. Yaro Burian Junior: \_\_\_\_\_

The image shows three handwritten signatures in blue ink, each written over a horizontal line. The first signature is for Prof. Dr. Fabiano Fruett (Presidente) and is clearly legible. The second signature is for Prof. Dr. Antonio Celso Fonseca de Arruda and is more stylized. The third signature is for Prof. Dr. Yaro Burian Junior and is also stylized.



# Resumo

Este trabalho descreve o desenvolvimento, projeto e aplicação de um sistema de medição embarcado destinado à avaliação de conforto do usuário do transporte público. O hardware deste sistema é dotado de acelerômetro triaxial e sistema de posicionamento global (GPS). Três algoritmos, desenvolvidos especialmente para este sistema, foram utilizados para a avaliação do conforto. O primeiro algoritmo está baseado na detecção de picos de aceleração que ultrapassam um limiar definido, estes eventos podem ser produzidos pela forma inadequada de dirigir do motorista e por estruturas ou defeitos no pavimento que possam gerar desconforto ao usuário. O segundo algoritmo adiciona a detecção de mudanças de aceleração (*Jerk*) acima de um limiar de  $0,4\text{m/s}^3$ . Enquanto, o terceiro algoritmo acrescenta o cálculo de um índice de conforto baseado no valor RMS da aceleração ponderada de acordo com a norma internacional ISO2631-1. Assim, o sistema permite reportar a amplitude, a posição geográfica e temporal dos eventos de aceleração, *Jerk* ou índice de conforto, através de um enlace sem fio ou pela leitura de um cartão de memória SD. Além de gerar um relatório, estes eventos são mostrados em um mapa por meio de um instrumento virtual desenvolvido na plataforma LabVIEW™. A análise das informações geradas permite identificar espacialmente a distribuição dos eventos, facilitando a tomada de ações preventivas ou corretivas, por parte das empresas de transporte e/ou prefeituras, que melhorem o conforto do usuário deste meio de transporte.

**Palavras-chave:** Sistema de Transporte, GPS, Aceleração, Conforto.

# Abstract

This work describes the development and the implementation of an embedded measurement system, which aim is the comfort assessment in public transportation vehicles. The hardware is based on tri-axial accelerometers and a global positioning systems (GPS). Also, for this system were specially developed three algorithms, which were implemented to evaluate the comfort. The first algorithm is based on the detection of acceleration's peaks over threshold values produced by reckless driving, holes, speed bumps or others defects. The second one adds the detection of acceleration changes (*Jerk*) peaks over  $0.4\text{m/s}^3$ ; whereas, the third one adds the RMS measurement from frequency-weighted accelerations according to the international standard ISO2631-1. Using a wireless link or an SD card, the system can report the magnitude, time, type and geographic position of the acceleration and jerk' events and the comfort index. Beyond to generate a report, these events are shown through a virtual instrument, developed in the LabVIEW™ platform. The analysis of those information allows to identify, geographically, the event's distribution. Hence, it is easier to make preventive or corrective actions to improve the comfort.

**Keywords:** Transportation System, GPS, Acceleration, Comfort.



# Agradecimentos

Ao meu orientador, Prof. Fabiano Fruett, pela orientação e apoio.

Ao Sr. Carlo Giuliano pelo suporte técnico prestado no desenvolvimento da dissertação.

Aos colegas do LSM pelas críticas, sugestões e ajuda neste projeto.

À CAPES, pelo apoio financeiro.



*Aos meus pais, irmã, avó e  
minha namorada pelo apoio  
incondicional*





# Sumário

<b>Lista de Figuras</b>	<b>xv</b>
<b>Lista de Tabelas</b>	<b>xix</b>
<b>Glossário</b>	<b>xxi</b>
<b>Lista de Símbolos</b>	<b>xxiii</b>
<b>Trabalhos Publicados Pelo Autor</b>	<b>xxvii</b>
<b>Introdução</b>	<b>1</b>
Objetivo Geral . . . . .	2
Objetivos específicos . . . . .	2
<b>1 Revisão bibliográfica</b>	<b>5</b>
1.1 Definições . . . . .	5
1.1.1 Conforto . . . . .	5
1.1.2 Sistema Embarcado . . . . .	7
1.1.3 <i>Jerk</i> . . . . .	7
1.1.4 Ônibus de trânsito rápido (BRT) . . . . .	8
1.2 Parâmetros de qualidade no transporte público . . . . .	8
1.3 Sistema de transporte público no Brasil . . . . .	10
1.4 Vibrações em ônibus . . . . .	11
1.5 Avaliação do conforto . . . . .	12
1.5.1 Indicadores de conforto . . . . .	12
1.5.2 ISO 2631-1 [1] . . . . .	16
1.6 Sistemas para a medição de conforto . . . . .	20
1.6.1 Monitor de solavancos para transporte de mercadorias baseado na medição da aceleração e da posição através do uso de GPS . . . . .	20
1.6.2 Sistema baseado em rede de sensores para o monitoramento das condições da via no transporte público . . . . .	22
1.6.3 Detector portátil de condução imprudente . . . . .	23
1.6.4 Avaliação de risco nos veículos utilizando o aparelho celular Nokia™ N95 . . . . .	25

<b>2</b>	<b>Estudo dos sinais envolvidos</b>	<b>29</b>
2.1	Características do sinal de aceleração em veículos . . . . .	29
2.1.1	Descrição do Hardware, firmware e resultados do primeiro protótipo . . . . .	29
2.1.2	Descrição do Hardware, firmware e resultados do segundo protótipo . . . . .	36
2.2	Características do sinal de <i>Jerk</i> em veículos . . . . .	43
2.2.1	Filtro passa-baixas para o cálculo do <i>Jerk</i> . . . . .	44
2.2.2	Algoritmo para calcular o <i>Jerk</i> . . . . .	45
<b>3</b>	<b>Hardware do Sistema</b>	<b>47</b>
3.1	Componentes de Hardware . . . . .	47
3.1.1	Sensores . . . . .	48
3.1.2	Módulo de posicionamento global (GPS) . . . . .	52
3.1.3	Módulo de comunicação . . . . .	54
3.1.4	Memória SD . . . . .	59
3.1.5	Microcontrolador . . . . .	60
3.1.6	Regulador de tensão . . . . .	61
3.2	SEMC-TP . . . . .	62
3.2.1	SEMC-TP Transmissor . . . . .	62
3.2.2	SEMC-TP Receptor . . . . .	64
<b>4</b>	<b>Firmware e Instrumento virtual do sistema</b>	<b>65</b>
4.1	Algoritmos para avaliar o conforto . . . . .	65
4.1.1	Algoritmo para o DEL-A . . . . .	65
4.1.2	Algoritmo para o DEL-AJ . . . . .	73
4.1.3	Algoritmo para o DEL-AIC . . . . .	75
4.2	Interface gráfica do Instrumento Virtual do sistema . . . . .	78
<b>5</b>	<b>Resultados experimentais</b>	<b>81</b>
5.1	Resultados utilizando o algoritmo DEL-A . . . . .	81
5.1.1	Primeiro teste do DEL-A . . . . .	81
5.1.2	Segundo teste do DEL-A . . . . .	82
5.2	Resultados utilizando o algoritmo DEL-AJ . . . . .	84
5.2.1	Primeiro teste do DEL-AJ . . . . .	84
5.2.2	Segundo teste do DEL-AJ . . . . .	84
5.3	Resultados utilizando o algoritmo DEL-AIC . . . . .	85
5.3.1	Primeiro teste do DEL-AIC . . . . .	85
5.3.2	Segundo teste do DEL-AIC . . . . .	87
5.3.3	Terceiro teste do DEL-AIC . . . . .	87
5.4	Comparação dos resultados . . . . .	87
5.5	Medida do consumo de corrente e temperatura . . . . .	88
<b>6</b>	<b>Conclusões</b>	<b>91</b>
6.1	Conclusões sobre as características dos sinais de interesse . . . . .	91
6.2	Conclusões sobre o <i>Hardware</i> . . . . .	92

6.3	Conclusões sobre os algoritmos e resultados experimentais . . . . .	93
6.4	Sugestões para trabalhos futuros . . . . .	95
<b>Referências bibliográficas</b>		<b>96</b>
<b>A Características desejadas no sistema de transporte público.</b>		<b>101</b>
<b>B Desenho do filtro FIR com <math>F_C</math> de 0,3Hz e decimação.</b>		<b>103</b>
<b>C Protocolo NMEA.</b>		<b>105</b>
C.1	GGA - Dados do sistema de posicionamento global. . . . .	106
C.2	RMC - Especificações GPS mínimas recomendadas e dados de trânsito. . . . .	107
<b>D Interfaces de comunicação serial</b>		<b>109</b>
D.1	<i>Serial Peripheral Interface</i> (SPI) . . . . .	109
D.1.1	Operação de Leitura . . . . .	110
D.1.2	Operação de Escrita . . . . .	110
D.2	Interface <i>Inter-Integrated Circuit</i> (I <sup>2</sup> C) . . . . .	110
D.2.1	Operação de leitura . . . . .	112
D.2.2	Operação de escrita . . . . .	112
D.3	Interface <i>universal asynchronous receiver/transmitter</i> (UART) . . . . .	113
D.4	<i>Secure Digital and Multimedia Card Interface</i> (MCI) . . . . .	114
D.4.1	Operação de leitura . . . . .	116
D.4.2	Operação de escrita . . . . .	116
<b>E Circuito impresso e diagrama de conexão do nó transmissor</b>		<b>117</b>
<b>F Firmware do microcontrolador</b>		<b>119</b>
F.1	Rotina Principal, rotinas de inicialização e rotinas específicas do DEL-AIC . . . . .	119
F.2	Rotina de interrupção do Timer do DEL-AIC . . . . .	130
F.3	Rotina do filtro, decimação e derivação do DEL-AJ . . . . .	133
<b>G Interface gráfica do SEMC-TP</b>		<b>139</b>



# Lista de Figuras

1.1	Interação do conforto e qualidade do trajeto no transporte [2]. . . . .	6
1.2	Sistema de transporte coletivo em Curitiba no Brasil [3] . . . . .	9
1.3	Eventos de aceleração usando o gravador AMP e o GPS do MMS. . . . .	21
1.4	Arquitetura do sistema BusNet. . . . .	23
1.5	Protótipo de medição com o MICAz e a placa do sensor. . . . .	23
1.6	Diagrama de blocos do detector portátil de condução imprudente. [4] . . . . .	24
1.7	Diagrama de blocos do processamento para o sinal de aceleração. [4] . . . . .	24
1.8	Resultados dos testes feitos em Bangkok com o sistema desenvolvido por Boonmee e Tangamchit e para a pontuação dada pelos usuários para arranques e freadas do veículo [4]. . . . .	25
1.9	Marco de referencia para o veículo e o telefone Nokia™ N95. . . . .	26
1.10	Exemplo do sinal de aceleração (Cinza) e sua reconstrução baseada na Wavelet de quinto nível (preto). . . . .	26
1.11	Diagrama de blocos do procesamento. . . . .	27
1.12	Representação tri-dimensional da aceleração da rotatória. . . . .	27
1.13	Comparação usando o N95. (a) Parada em uma intersecção. (b) Comparação da velocidade de dois motoristas. . . . .	28
2.1	Modulo de aquisição baseado em um microcontrolador de 8 bits. . . . .	30
2.2	Esquemático do modulo de aquisição. . . . .	30
2.3	Orientação dos eixos cartesianos do acelerômetro no assoalho do carro. . . . .	31
2.4	Análise carro No. 1. . . . .	32
2.5	FFT da aceleração do teste no carro No. 1. . . . .	32
2.6	Sinais de aceleração do carro No. 1. . . . .	33
2.7	Análise carro No. 2. . . . .	33
2.8	Máxima aceleração no eixo vertical, no análise do carro No. 2. . . . .	34
2.9	Sistema BRT, nomeado Transmilenio em Bogotá. Colômbia. . . . .	34
2.10	Aceleração no sistema BRT com acelerômetro na faixa de 2G. . . . .	35
2.11	Orientação dos eixos cartesianos do acelerômetro no assoalho do ônibus BRT. . . . .	35
2.12	Aceleração no sistema BRT com acelerômetro na faixa de 4G. . . . .	35
2.13	Segundo protótipo baseado na placa de desenvolvimento do LPC2478. . . . .	37
2.14	Aceleração no ônibus circular da UNICAMP ao longo do tempo. . . . .	37
2.15	Aceleração do maior evento no ônibus circular da UNICAMP. . . . .	38
2.16	Análise espectral usando FFT da aceleração no ônibus circular da UNICAMP. . . . .	39

2.17	Algoritmo para obter a DWT. A-Aproximação e D-Detalhes. . . . .	40
2.18	Análise espectral usando DWT da aceleração no ônibus circular da UNICAMP, com resolução de 0,33Hz e janelas de 1s. Eixo Z. . . . .	40
2.19	Análise espectral usando DWT da vibração produzida pelo motor, com resolução de 0,33Hz e janelas de 0,1s. Eixo Z. . . . .	41
2.20	Análise espectral usando DWT da aceleração vertical produzida por uma lombada, com resolução de 0,33Hz e janelas de 0,01s. Eixo Z. . . . .	41
2.21	Análise espectral usando DWT da aceleração transversal, com resolução de 0,33Hz e janela de 0,01s(Esquerda). Amplificação de 0Hz a 5Hz(Direita). Eixo X. . . . .	42
2.22	Análise espectral usando DWT da aceleração longitudinal, com resolução de 0,33Hz e janela de 0,1s(Esquerda). Amplificação de 0Hz a 5Hz(Direita). Eixo Y. . . . .	42
2.23	Aceleração e <i>Jerk</i> no sistema BRT na faixa de 2G. . . . .	43
2.24	Desconforto dos usuários de trens em função do <i>Jerk</i> lateral. [5] . . . . .	44
2.25	Diagrama de blocos do algoritmo para calcular o <i>Jerk</i> destacando o filtro implementado. . . . .	45
2.26	Resposta ao impulso do filtro passa-baixa com $F_C$ de 0,3Hz e $F_S$ a 10Hz. . . . .	45
2.27	Aceleração longitudinal e <i>Jerk</i> longitudinal do ônibus circular da UNICAMP usando o algoritmo proposto. . . . .	46
3.1	Modelo e estrutura simplificada do comb-drive. . . . .	48
3.2	Diagrama de blocos do acelerômetro ADXL345 [6]. . . . .	50
3.3	Diagrama esquemático da conexão do acelerômetro ADXL345 com comunicação I <sup>2</sup> C. . . . .	50
3.4	Diagrama de blocos do sensor de temperatura TMP100 [7]. . . . .	52
3.5	Diagrama esquemático do circuito de polarização do sensor de temperatura TMP100 [7]. . . . .	52
3.6	Módulo GPS. ME-1000RW. . . . .	53
3.7	Diagrama de blocos do ME-1000RW e informação dos pinos. . . . .	54
3.8	Terminal Rodoviário Tiête. . . . .	55
3.9	Nós da rede ZigBee®. . . . .	57
3.10	Rede Digimesh™. . . . .	58
3.11	Módulo XBee-Pro Digimesh™ 900. . . . .	59
3.12	Memória SD. . . . .	59
3.13	Diagrama de conexão da memória SD. . . . .	60
3.14	Circuito periférico do regulador de tensão. . . . .	62
3.15	Diagrama de blocos do SEMC-TP transmissor. . . . .	63
3.16	Montagem de teste para o SEMC-TP transmissor. . . . .	63
3.17	SEMC-TP Receptor. . . . .	64
4.1	Fluxograma da rotina principal do algoritmo DEL-A, DEL-AJ e DEL-AIC. . . . .	67
4.2	Fluxograma do algoritmo para a inicialização da memória SD [8]. . . . .	68
4.3	Fluxograma do algoritmo de interrupção do <i>timer</i> . . . . .	70
4.4	Fluxograma do algoritmo de interrupção da UART1 (GPS). . . . .	70
4.5	Fluxograma do algoritmo de interrupção da UART2 (Módulo de comunicação). . . . .	71
4.6	Estrutura do vetor de ocorrência de eventos. . . . .	72
4.7	Fluxograma para algoritmos com funções específicas . . . . .	74
4.8	Fluxograma do algoritmo para a detecção de eventos devidos ao <i>Jerk</i> (DEL-AJ). . . . .	75

4.9	Fluxograma do algoritmo para o cálculo do índice de conforto (DEL-AIC). . . . .	76
4.10	Fluxograma do algoritmo para o cálculo do valor de aceleração RMS ponderado. . .	76
4.11	Curvas de ponderação, $W_K$ e $W_d$ , com 512 pontos baseadas na norma ISO2631-1. . .	77
4.12	Tela do software do SEMC-TP para a visualização dos eventos. . . . .	79
4.13	Tela do software do SEMC-TP para a configuração dos nós e descarga de eventos. . .	80
5.1	Percurso escolhido para o primeiro teste do SEMC-TP com algoritmo DEL-A. . . . .	82
5.2	Eventos gerados para o primeiro teste do DEL-A com o motorista dirigindo de maneira rápida. . . . .	83
5.3	Eventos gerados para o primeiro teste do DEL-A com o motorista dirigindo normalmente. . . . .	83
5.4	Eventos gerados para o segundo teste do DEL-A no ônibus circular da UNICAMP. .	84
5.5	Eventos gerados para o teste do DEL-AJ. . . . .	85
5.6	Eventos gerados para o primeiro teste do DEL-AIC no ônibus da moradia da UNICAMP.	86
5.7	Eventos gerados para o primeiro teste do DEL-AIC em um carro. . . . .	86
5.8	Eventos gerados para o segundo teste do DEL-AIC em um carro. . . . .	87
5.9	Eventos gerados para o terceiro teste do DEL-AIC em um ônibus de transporte coletivo.	88
5.10	Corrente consumida pelo SEMC-TP transmissor. . . . .	89
5.11	Leitura do sensor de temperatura desde a memória SD. . . . .	90
B.1	Diagrama de blocos na plataforma LabVIEW para o projeto do filtro utilizado no calculo do <i>Jerk</i> . . . . .	103
D.1	Interface SPI com 4 linhas. . . . .	110
D.2	Diagrama de tempo para a operação de leitura do SPI com 4 linhas [6]. . . . .	111
D.3	Diagrama de tempo para a operação de escrita do SPI com 4 linhas [6]. . . . .	111
D.4	Interface <i>Inter-Integrated Circuit</i> (I <sup>2</sup> C). . . . .	111
D.5	Diagrama de tempo para as condições de inicio e fim de transmissão com a interface I <sup>2</sup> C [6]. . . . .	112
D.6	Diagrama de tempo para a operação de leitura na interface I <sup>2</sup> C [9]. . . . .	113
D.7	Diagrama de tempo para a operação de escrita na interface I <sup>2</sup> C [9]. . . . .	113
D.8	Interface UART. . . . .	114
D.9	Mensagem da interface UART. . . . .	114
D.10	Interface MCI. . . . .	115
D.11	Operações de não resposta e não dados da interface MCI [8]. . . . .	115
D.12	Operação de leitura da interface MCI [8]. . . . .	116
D.13	Operação de escrita da interface MCI [8]. . . . .	116
E.1	Camada superior ( <i>top</i> ) do circuito impresso do SEMC-TP Transmissor. . . . .	117
E.2	Camada inferior ( <i>bottom</i> ) do circuito impresso do SEMC-TP Transmissor. . . . .	117
E.3	Diagrama esquemático do SEMC-TP Transmissor . . . . .	118
G.1	Fluxograma da interface gráfica do SEMC-TP no Software LabView™. Aba de visualização de eventos. . . . .	140

- G.2 Fluxograma da interface gráfica do SEMC-TP no Software LabView™. Aba de configuração de nós. . . . . 141
- G.3 Fluxograma para o envio dos comandos de configuração dos nós no software LabView™. . . . . 142



# Lista de Tabelas

1.1	Desempenho do transporte público na cidade de São Carlos . . . . .	10
1.2	Frequências recorrentes da aceleração em ônibus [10] . . . . .	11
1.3	Grandezas com possível influência no conforto em trens [2] . . . . .	12
1.4	Indicadores de conforto em trens [2] . . . . .	13
1.5	Conceito de conforto para $W_Z$ [2] . . . . .	14
1.6	Conceito de conforto para $RI$ [2] . . . . .	14
1.7	Conceito de conforto para a norma CEN 1999 [2] . . . . .	15
1.8	Parâmetros das funções de transferência utilizados pela norma ISO 2631-1 [1] . . . .	18
1.9	Valores multiplicativos, $K$ , para pessoas sentadas. . . . .	19
1.10	Índices recomendados pela norma ISO 2631-1 para a avaliação do conforto no trans- porte público . . . . .	20
1.11	Especificações do gravador AMP . . . . .	21
1.12	Regras para determinar a fonte do evento . . . . .	24
2.1	Aceleração pico a pico medido em carros e ônibus do sistema BRT. . . . .	36
3.1	Comparação entre acelerômetros tri-axiais digitais . . . . .	49
3.2	Taxa de dados, largura de banda e código do registro BW_RATE disponíveis para o ADXL345 [6] . . . . .	51
3.3	Especificações do módulo GPS ME-1000RW [11] . . . . .	54
3.4	Características dos padrões de comunicação sem fio [12] . . . . .	56
3.5	Comparação entre ZigBee® e Digimesh™ [13] . . . . .	58
3.6	Consumo de corrente dos dispositivos do SEMC-TP . . . . .	62
3.7	Custo do protótipo do SEMC-TP . . . . .	64
4.1	Tipo do evento . . . . .	69
4.2	Índice de conforto conforme aceleração RMS ponderada para o DEL-AIC . . . . .	78
5.1	Amplitude máxima dos eventos detectados pelo SEMC-TP . . . . .	88
6.1	Resumo das especificações técnicas do nó transmissor do SEMC-TP . . . . .	93
C.1	Campos da mensagem NMEA tipo GGA [11] . . . . .	106
C.2	Campos da mensagem NMEA tipo RMC [11] . . . . .	107



# Glossário

**ABS** *Anti-lock Braking System* ou Sistema de freio antitravamento.

**Acelerômetro** Sensor utilizado para medir aceleração.

**ADC** Conversor analógico digital.

**API** *Application Programming Interface*. É um conjunto de instruções de programação para fazer uso de um software, geralmente baseado em uma ferramenta de Web.

**ARM** Arquitetura de processadores de 32 bits (*Advanced RISC Machines*).

**Broadcast** Transmissão de informação sem destinatário específico.

**BRT** *Bus Rapid Transit* ou Rede Integrada de Transporte(RIT).

**CISC** *Complex Instruction Set Computer* ou Conjunto Complexo de Instruções.

**Decimação** Diminuição da frequência de amostragem.

**DEL-A** Algoritmo para a detecção de eventos de desconforto utilizando o limiar de aceleração.

**DEL-AIC** Algoritmo para a detecção de eventos de desconforto utilizando o limiar de aceleração e o cálculo de um índice de conforto.

**DEL-AJ** Algoritmo para a detecção de eventos de desconforto utilizando o limiar de aceleração e *Jerk*.

**DFT** *Discrete Fourier Transform* ou Transformada Discreta de Fourier.

**DigiMesh** Protocolo de comunicações sem fio ponto a ponto da companhia Digi International.

**Dropout** Diferença mínima entre a tensão de entrada e saída de um regulador de tensão para garantir a regulação da tensão de saída.

**DWT** Transformada Discreta de Wavelet.

**FFT** Transformada Rápida de Fourier.

**Filtro antialiasing** Filtro utilizando antes de amostrar um sinal para limitar a largura de banda e garantir o teorema de Nyquist.

**Filtro FIR** Filtro digital de resposta ao impulso finita, o qual é caracterizado por depender somente de valores prévios da entrada, fornecer uma resposta ao impulso nula após um tempo finito e sua boa estabilidade.

**Firmware** Conjunto de instruções programadas em um *Hardware*.

**Frequência de amostragem** Número de amostras por segundo coletadas de um sinal contínuo.

**GPIO** *General Purpose Input/Output* ou portas programáveis de entrada e saída de dados.

**GPS** Global Positioning System.

**I<sup>2</sup>C** Interface *Inter-Integrated Circuit*.

**IFT** *Inverse Fourier Transform* ou Transformada Inversa de Fourier.

**ISO** *International Organization for Standardization* ou Organização Internacional para Padronização.

**ITS** *Intelligent Transportation Systems* ou Sistemas Inteligentes de Transporte.

**Jerk** Taxa de variação da aceleração, também conhecido como Tranco.

**KML** Keyhole Markup Language. Formato de arquivo que contém a notação adequada para a visualização de coordenadas geográficas em mapas online.

**LabView** O *Laboratory Virtual Instrument Engineering Workbench* é uma linguagem e plataforma de programação criada pela National Instruments®.

**Latência da rede** É a diferença de tempos entre o início da transmissão no remetente e a recepção do pacote de dados no destinatário.

**LSB** *Least Significant Bit* ou Bit menos significativo.

**MCI** *Secure Digital and Multimedia Card Interface*.

**MEMS** *Micro-Electro-Mechanical Systems* ou Sistemas Micro-Eletro-Mecânicos.

**Mesh** Topologia de rede composto por vários nós, na qual estes podem se intercomunicar.

**Microcontrolador** Circuito integrado programável que inclui uma unidade de processamento, memória e periféricos de entrada e saída.

**MSB** *Most Significant Bit* ou Bit mais significativo.

**NMEA** *National Marine Electronics Association*. Associação que estabelece especificações para a comunicação de dispositivos eletrônicos de navegação.

**Offset** Saída de um sensor com sinal de entrada nula.

**Percentil** Em estatística, é o valor divide uma população,  $N$ , em partes proporcionais. O percentil 0,5 é chamado mediana.

**PLL** *phase-locked loop* é um sistema eletrônico que gera um sinal de saída que está relacionada com a fase do sinal de referência, está formado por um oscilador controlado por tensão e um detetor de fase. O PLL é utilizado para gerar sinais de *clock* de maior frequência dentre outras aplicações.

**Pull-up** Resistor utilizado em eletrônica digital para garantir um nível lógico alto se o circuito estiver em condições de circuito aberto ou alta impedância.

**RISC** *Reduced Instruction Set Computer* ou Computador com um Conjunto Reduzido de Instruções.

**RMS** *Root Mean Square* ou valor quadrático médio.

**Roll** É o movimento angular no plano horizontal e em torno ao eixo longitudinal.

**SCL** Linha de *clock* do protocolo de comunicação I<sup>2</sup>C.

**SDA** Linha de dados do protocolo de comunicação I<sup>2</sup>C.

**SEMC-TP** Sistema embarcado para o monitoramento do conforto no transporte público.

**SPI** *Serial Peripheral Interface*.

**Teorema de Nyquist** Teorema que garante a reconstrução de um sinal amostrado através da amostragem a uma frequência maior do que o dobro da máxima frequência do sinal.

**Timer** Temporizador utilizado pelo microcontrolador para gerar interrupções em um intervalo de tempo determinado.

**UART** *Universal Asynchronous Receiver/Transmitter* ou comunicação assíncrona.

**Wavelet** Transformada discreta capaz de decompor um sinal no domínio da frequência usando o escalamento e translações de um sinal semente, chamada Wavelet mãe.

**ZigBee** Protocolo de comunicações sem fio que utiliza o padrão IEEE 802.15.4.



# Lista de Símbolos

$a_{RMS}$	- Aceleração RMS ( $m/s^2$ ).
$a_{uri}$	- Aceleração de X ou Y corrigida devido a possíveis inclinações no eixo Z. ( $m/s^2$ ).
$a_v$	- Aceleração vetorial dos eixos X, Y, Z.
$a_w(t)$	- Aceleração ponderada.
$a_w$	- Valor RMS da aceleração ponderada. ( $m/s^2$ ou $rad/s^2$ )
$a_{X,Y,Z}$	- Aceleração medida no eixo X, Y ou Z ( $m/s^2$ ou G).
$a_{XGPS}$	- Aceleração no eixo X calculada a partir da velocidade medida pelo GPS.
$B_{L,V}(f)$	- Pesos empíricos para o indicador $W_Z$ no eixo lateral ou vertical.
$CF$	- Fator de Cresta.
$f$	- Frequência (Hz).
$f_0$	- Frequência central para o calculo do Ride Index (Hz).
$f_C$	- Frequência de corte(Hz).
$f_S$	- Frequência de amostragem(Hz).
$H(p)$	- Função de transferência de um filtro.
$J$	- <i>Jerk</i> . ( $m/s^3$ )
$MTVV$	- Valor máximo da vibração transitória.
$N_{MV}$	- Indicador de conforto da norma europeia (CEN) usando o método padrão.
$N_{VA}$	- Indicador de conforto da norma europeia (CEN) para passageiros sentados.
$N_{VD}$	- Indicador de conforto da norma europeia (CEN) para passageiros de pé.
$p$	- $j2\pi f$ Frequência angular imaginaria.
$P_{CT}$	- Indicador de distúrbios do conforto para transições de curvas em trens.
$P_{DE}$	- Indicador de distúrbios do conforto para eventos discretos.
$RI_{X,Y,Z}$	- Indicador de conforto Ride Index para o eixo Z, Y e X.
$T$	- Tempo de duração do percurso (segundos).
$VDV$	- Indicador do método de dose de vibração à quarta potência.
$W_b$	- Filtro ponderado usado no calculo de $N_{MV}$ , $N_{VA}$ e $N_{VD}$ para o eixo Z.
$W_c$	- Filtro ponderado usado no calculo de $N_{MV}$ , $N_{VA}$ e $N_{VD}$ para o eixo X medido no espaldar da cadeira.
$W_d$	- Filtro ponderado usado no calculo de $N_{MV}$ , $N_{VA}$ e $N_{VD}$ para os eixos X e Y.
$W_k$	- Ponderação de frequência, sendo k as superfícies de medição.

$W_Z$	-	Indicador de conforto definido por Sperling.
$X, Y, Z$	-	Eixos translacionais.
$\ddot{y}$	-	Aceleração lateral absoluta ( $\text{m/s}^2$ ).
$\ddot{y}_m$	-	Valor pico a pico da aceleração lateral.
$\ddot{y}_p$	-	Valor absoluto da média da aceleração lateral.
$\dot{\varphi}$	-	Máxima velocidade angular absoluta, <i>roll</i> , em uma curva de transição.



# Trabalhos Publicados Pelo Autor

1. Castellanos Juan; Fruett Fabiano; , "Embedded system for monitoring the comfort in public transportation," Circuits and Systems (ISCAS), 2011 IEEE International Symposium on , pp.2809-2812, 15-18 May 2011
2. Castellanos, Juan C.; Susin, Altamiro A.; Fruett, Fabiano; , "Embedded sensor system and techniques to evaluate the comfort in public transportation," Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on ,pp.1858-1863, 5-7 Oct. 2011.



# Introdução

A qualidade e segurança dos meios de transporte influenciam significativamente em fatores sociais importantes, relacionados ao trabalho, educação, e lazer. Assim, na busca de uma melhoria no desempenho escolar e laboral é necessário desenvolver medidas que não só diminuam a taxa de acidentes, senão também melhorem a qualidade do transporte público. Um dos principais problemas das grandes cidades é a frota crescente de carros, deste modo, favorecer a utilização do transporte público é fundamental. Convencer a população a deixar seus carros ou motocicletas em casa ou em um local perto da estação de ônibus, diminuirá o tráfego e conseqüentemente outros problemas tais como: poluição ambiental e sonora. Em eventos que atraem a grande multidão, tais como a copa do mundo e os jogos olímpicos, é vital garantir a qualidade do transporte público para o transporte das grandes massas. Alguns estudos têm indicado a importância do monitoramento da aceleração no transporte público, por exemplo: pesquisa feita na região de Bavária na Alemanha [14] demonstrou que 50% dos ferimentos em ônibus são causados por acidentes sem colisão do veículo e que a principal causa destes foram as freadas de emergência. A mesma pesquisa identificou resultados similares na Áustria e na Grã-Bretanha, onde os acidentes sem colisão causaram uma grande quantidade de lesões especialmente em idosos. A maior parte das lesões produzidas no transporte público são principalmente originadas por grandes variações na aceleração do veículo, que ocasionaram lesões nos membros, cabeça, coluna, peito, pélvis, abdômen e pele [15]. Para obter uma melhor noção do problema, os pacientes com lesões produzidas no transporte público que não estiveram associados a choques de trânsito foram monitorados em seis departamentos de emergência de diferentes centros de saúde, a conclusão foi que: 63% das pessoas lesionadas tiveram associado, como causa do ferimento, uma variação brusca na aceleração do veículo. Além disso, Halpern, [15], concluiu que a conservação do equilíbrio no ônibus é um fator influente, pois a maior parte dos feridos viajava de pé; também concluiu que a população mais vulnerável são idosos e que a aceleração média dos ônibus é superior ao valor que o corpo de alguns deles pode suportar sem perder o equilíbrio. Baseando-se nos resultados obtidos por DeGraaf e Van Weaper [16], nos quais se mostrou que a aceleração média nos ônibus em Amsterdam é de 1 a 2 m/s<sup>2</sup>, enquanto que uma aceleração segura deve ser de até 1,6m/s<sup>2</sup>. Entretanto o mesmo trabalho ressalta que a habilidade para conservar o equilíbrio cai linearmente com a idade. De

acordo com esses estudos, a aceleração é um componente crítico na avaliação do conforto e da segurança dos passageiros do transporte público. Portanto, consideramos importante o desenvolvimento de um sistema que monitore o conforto do usuário de ônibus, que permita supervisionar o veículo durante todo seu percurso e que permita identificar quando o motorista ou as condições da estrada possam causar algum acidente, ocasionar uma lesão ao passageiro ou simplesmente desconforto. Então, um sistema embarcado para o monitoramento do conforto no transporte público (SEMC-TP) pode ser implementado através de um sistema de aquisição e processamento baseado em acelerômetro, sensor de temperatura e GPS, fornecendo informação das maiores amplitudes da aceleração e das variações desta (*Jerk*) durante um dado percurso. Além disso, informações tais como: amplitude da aceleração, localização geográfica e localização temporal poderão ser transmitidas através de um enlace sem fio ou cartão de memória. Um fator importante no desenvolvimento deste sistema embarcado reside na vantagem competitiva do baixo custo dos sensores microeletrônicos utilizados e da sua fácil instalação nos veículos de transporte sob monitoração. Atualmente o sistema de transporte faz uso de um tacógrafo analógico, os quais são instrumentos de controle de velocidade obrigatórios no transporte público de passageiros superior a dez pessoas, segundo o Código de Trânsito Brasileiro. Este sistema mede somente velocidade, não considerando uma análise das acelerações, que podem afetar negativamente o conforto e a segurança dos usuários. Além disso, estudos do conforto são limitados às pesquisas de opinião ou mediante o registro das reclamações dos usuários do sistema de transporte público. Embora o SEMC-TP não seja o suficientemente exato na medição de velocidade para substituir o tacógrafo, provê um critério mais objetivo para avaliar a qualidade do transporte e principalmente da localização das fontes de desconforto.

## Objetivo Geral

Desenvolver um sistema para o monitoramento do conforto no transporte público através da medição direta da aceleração, posição geográfica e temperatura; e o processamento destas informações, resultando no cálculo do *Jerk* e de um indicador de conforto conforme Norma Internacional [1].

### Objetivos específicos:

- Desenvolver um módulo de aquisição da aceleração e temperatura.
- Desenvolver o algoritmo para cálculo do *Jerk* e indicador de conforto.
- Desenvolver um protótipo de baixo custo, fácil instalação e manutenção.

- Projetar e programar uma interface de usuário intuitiva, de fácil uso e com o processamento de dados adequado para destacar rapidamente os eventos mais críticos a um determinado supervisor.



# Capítulo 1

## Revisão bibliográfica

Nesse capítulo são apresentadas algumas definições relacionadas com o conforto e com alguns dos principais parâmetros utilizados para avaliar a qualidade do transporte público. Além disso, uma breve ideia do estado de transporte na cidade de São Carlos, Brasil será discutida. Posteriormente, será apresentada uma pesquisa sobre as vibrações percebidas em ônibus e citar-se-ão alguns indicadores de conforto usados para avaliar o transporte público sob a perspectiva da interação de fatores dinâmicos do veículo sobre o passageiro. Finalmente, algumas pesquisas abordando a avaliação do conforto no sistema de transportes, em particular utilizando a medição de aceleração e posição nesses veículos, são brevemente expostas.

### 1.1 Definições

O significado de alguns dos termos utilizados nesta dissertação são apresentados e discutidos a seguir.

#### 1.1.1 Conforto

Segundo o dicionário *Webster's*, conforto é definido como: um estado de tranquilidade, livre de dor, desejo, ou ansiedade; também, o que contribui para essa condição. Assim, o conforto tem dois componentes: fisiológico e psicológico. Segundo Förstberg, nas pesquisas da área de transporte, o conforto é tipicamente definido como bem-estar subjetivo (nível de seu atual estado de felicidade); no entanto, conforto é uma das variáveis que podem contribuir ao bem-estar e não é uma parte necessária desta [2]. A seguir se descrevem alguns dos principais termos utilizados na definição do conforto no transporte público.

### 1.1.1.1 Qualidade do trajeto

Qualidade do trajeto é um conjunto de fatores no ambiente do veículo que contribuem ou influenciam no bem-estar de um indivíduo, e podem ser classificados como: fatores humanos, dinâmicos, ambientais e espaciais [2], como mostrado na Figura 1.1.

Entre os fatores humanos estão: a idade, o gênero, a experiência, atividade, entre outras características do passageiro. Os fatores dinâmicos estão relacionados à interação do ônibus com o passageiro refletidos em grandezas tais como: aceleração, mudanças de aceleração, movimentações angulares e eventos súbitos como choques. Os fatores ambientais incluem grandezas tais como: temperatura, pressão, qualidade do ar, ventilação e ruído. Enquanto que os fatores espaciais estão relacionados com as condições das cadeiras e a ergonomia do espaço do passageiro, por exemplo: o espaçamento entre cadeiras, o espaço lateral, o espaço para colocar as pernas, etc [2].

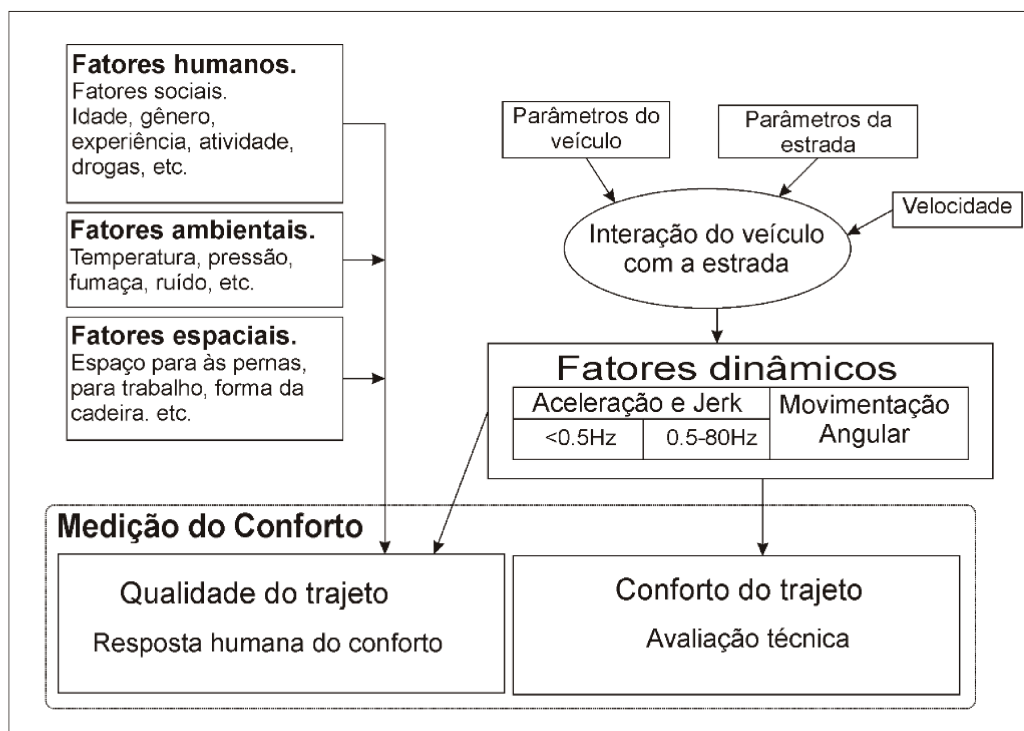


Figura 1.1: Interação do conforto e qualidade do trajeto no transporte [2].

### 1.1.1.2 Conforto do trajeto

Conforto do trajeto, conforto do passageiro ou conforto médio do percurso são termos tipicamente usados para fazer referência ao desconforto provocado ao passageiro devido às condições dinâmicas (relacionadas ao movimento do veículo). A avaliação técnica do conforto do trajeto pode ser dividida em diferentes categorias, tais como [2]:



- Conforto/desconforto médio do trajeto (índice da viagem): medição das acelerações longitudinais, laterais e verticais no intervalo de frequências de 0,5Hz a 80Hz. Normas internacionais e outros índices usam o conceito de desconforto do trajeto, ou seja, uma viagem com um valor alto indica menor conforto.
- Conforto do trajeto estimado: avaliação da viagem dada pelos usuários em uma escala de 1 a 5, sendo que 1 indica um conforto baixo e 5 um conforto alto.
- Distúrbios no conforto: distúrbios causados por movimentos súbitos do veículo (eventos discretos) ou causados por altas acelerações laterais e/ou *Jerk* lateral.
- Enjoo: o enjoo no passageiro é causado por movimentos lineares prolongados, abaixo de 0,5Hz ou movimentações angulares.
- Efeitos das vibrações verticais em atividades específicas: Vibrações na faixa de 0,5Hz a 10Hz causam dificuldades, por exemplo quando o passageiro deseja escrever ou beber.

### 1.1.2 Sistema Embarcado

Sistemas embarcados são sistemas de hardware e software baseados em dispositivos que não são necessariamente computadorizados, mas que contam no mínimo com um chip programável (tipicamente na forma de micro-controlador, micro-processador ou processador digital de sinais). Exemplos típicos de aplicações embarcadas incluem: Controles de ABS nos freios e em motor de veículos automotores, piloto automático de uma aeronave, marca-passos, circuitos no interior de equipamentos de comutação utilizados em telecomunicação, dispositivos ambientais, gravadores de vídeo, fornos de microondas, geladeiras, unidades de controle em robôs, dentre outros.

O uso de sistemas embarcados está crescendo exponencialmente. Atualmente 90% de dispositivos computacionais se encontram em sistemas embarcados e não em computadores pessoais (PCs). A taxa de crescimento dos sistemas embarcados é superior a 10% ao ano e se estima que haverá aproximadamente 40 bilhões de dispositivos no planeta em 2020 [17]. Assim, enquanto o mercado de PCs está orientado pela necessidade de prover mais desempenho, o mercado embarcado tem necessidades diferentes, por exemplo, economia, atendimento a requisitos legais, tecnológicos dentre outros [17].

### 1.1.3 *Jerk*

Define-se *Jerk* (J) como: a terceira derivada da posição ou a taxa de mudança da aceleração no tempo, também é conhecido como Tranco no Brasil e *Jolt* no Reino Unido. Na norma internacional

ISO 2041 (1990), *Vibration and shock - Vocabulary*, o *Jerk* é definido como "O vetor resultante da derivada da aceleração no tempo".

O *Jerk* tem grande importância na medição dos efeitos destrutivos de um movimento em mecanismos ou instrumentos. O *Jerk* é também usado na medição do conforto de passageiros em veículos. Por exemplo, os projetistas de elevadores limitam que o *Jerk* vertical do sistema a  $2\text{m/s}^3$  [18]; enquanto que os projetistas de trens cuidam que o *Jerk* lateral seja menor do que  $0,4\text{m/s}^3$  [2]. Finalmente, para o transporte terrestre em geral, Hoberock recomenda utilizar  $0,3\text{G/s}$  como limite aceitável do *Jerk* longitudinal [19].

#### 1.1.4 Ônibus de trânsito rápido (BRT)

O sistema de ônibus de trânsito rápido ou *Bus Rapid Transit* (BRT) é definido como um sistema de transporte coletivo rápido, de alto desempenho e flexível que combina um arranjo de elementos em um sistema integrado permanentemente [20]. O BRT é um sistema de transporte baseado em ônibus com as seguintes características principais: vias ordinárias ou com faixas exclusivas, estações confortáveis e seguras no meio das vias ou afastadas delas, veículos de grande capacidade de passageiros e em alguns casos híbridos, arrecadação de diversas maneiras (dinheiro, cartões inteligentes, de crédito ou débito) e sistemas de transporte Inteligente, dentre outros. Alguns exemplos de sistemas de transporte inteligentes ou ITS são: sistemas para alterar o tempo dos sinais priorizando o passo dos veículos do BRT, dispositivos para prever colisões ou indicar o local adequado para estacionar nas estações, sistemas de informação aos usuários, sistemas de gestão automática do sistema (despacho e agendamento das linhas, contagem de passageiros), etc.

O BRT é um sistema muito utilizado nos Estados Unidos em cidades tais como: Boston, Los Angeles, Miami, Las Vegas, dentre outras; e também em cidades de outros países, tais como: San Juan de Puerto Rico, Adelaide na Austrália, Auckland na Nova Zelândia, Bogotá na Colômbia e Curitiba no Brasil, o qual é mostrado na Figura 1.2.

## 1.2 Parâmetros de qualidade no transporte público

Peter Kelly foi motorista de ônibus no Reino Unido e também trabalhou com o planejamento e gestão das operações do transporte público; ele identificou os usuários do transporte público e suas necessidades [21]. Uma das principais modificações que ele descreveu foi a evasão de alguns grupos de usuários tais como estudantes e mães com filhos. Assim, com a redução dos usuários no setor estudantil devido ao incremento do uso de carros na jornada escolar e a redução do uso de ônibus por parte das mães com filhos, causada pelas dificuldades encontradas no transporte público; as classes



Figura 1.2: Sistema de transporte coletivo em Curitiba no Brasil [3] .

socialmente desfavorecidas se converteram no principal alvo do mercado da indústria de ônibus. Kelly destaca que os donos de automóvel são um mercado potencial para o desenvolvimento do setor de ônibus e que prover as condições que eles necessitam para estimulá-los ao uso de ônibus, no lugar do automóvel, deve ser uma prioridade para a melhoria do transporte público.

As necessidades e anseios dos usuários do transporte público foram estudados por muitos anos e os aspectos mais relevantes a serem considerados são: confiabilidade, frequência, informação ao usuário, tempo de viagem ou velocidade, acessibilidade, lotação, características dos veículos, comportamento dos operadores, estado das vias, dentre outros. Descritos no apêndice A.

Considerando todos os requisitos do usuário de transporte público, pode-se melhorar consideravelmente a qualidade do serviço. No entanto, o objetivo desta dissertação está direcionado a avaliar o conforto percebido somente no interior do ônibus visto sob uma perspectiva dos fatores dinâmicos do veículo; ou seja, às características do objeto onde o usuário permanece 40% do tempo total desde que ele(a) decide a rota que vai tomar até chegar ao local de destino, como mencionado por Kelly [21].

Então, além de um sistema de informação para o usuário, caminhos e pontos de ônibus mais confortáveis, tempos de espera mais curtos e localidades da cidade onde o sistema público tenha acesso exclusivo. É preciso garantir que o tempo da viagem seja o mais curto possível, sem comprometer o conforto nem a segurança do usuário.

Sistemas de transporte com melhorias na qualidade do serviço têm mostrado sucesso através do incremento do número de usuários em torno de 50% dos quais 25% eram antigos usuários de automóveis; esse incremento se deu pela utilização de sistemas de informação em tempo real, passagens eletrônicas, faixas exclusivas e forte publicidade; tais sistemas são: o projeto de ônibus *SMART* em

Liverpool, *City Express* em Belfast, o sistema de transporte em Adelaide ou *Superoute 66* em Ipswich dentre outros. [21].

Lima [22] aponta que os custos da falta de qualidade são elevados para a cidade devido aos congestionamentos, acidentes, avarias e mortes que a sociedade deve custear. Por exemplo, os custos anuais devidos aos acidentes de trânsito foram de R\$ 28 bilhões, segundo a última pesquisa feita pelo Instituto de Pesquisa Econômica e Aplicada (IPEA) do Brasil.

### 1.3 Sistema de transporte público no Brasil

Uma pesquisa feita na cidade de São Carlos, [23], permitiu avaliar o desempenho do transporte coletivo no Brasil. Na Tabela 1.1 são apresentados os resultados da pesquisa levando em conta os fatores de qualidade vistos na seção anterior. A pesquisa foi baseada na entrevista de 1050 pessoas entre estudantes, trabalhadores, idosos e deficientes. Dos resultados é possível identificar que os fatores mais críticos são: o estado das vias, a lotação e os locais de parada.

Tabela 1.1: Desempenho do transporte público na cidade de São Carlos

Fatores de Qualidade	Importância	Desempenho
Acessibilidade	4,3	3,7
Frequência	4,2	3,3
Tempo da viagem	3,5	3,4
Lotação	4,1	2,7
Confiabilidade	4,1	3,2
Segurança	4,3	3,6
Veículos	4,2	3,5
Locais de parada	4,2	2,6
Sistema de informação	3,6	2,5
Conectividade	4,1	3,5
Operadores	4,1	3,9
Vias	4,2	2,2

A atual metodologia de avaliação da qualidade do serviço no Brasil está fundamentada no monitoramento de indicadores. Os indicadores utilizados na cidade de São Carlos são: grau de variação dos intervalos da viagem nos pontos de controle, grau de falhas dos veículos em operação, grau de cumprimento das viagens, grau de aprovação da frota em vistorias programadas, grau de aprovação da frota em vistorias de campo, grau de reclamação dos usuários sobre o serviço, grau de ocorrência de acidentes e grau de ocorrência de irregularidades de trânsito [23]. O grau de falhas de veículo em operação, grau de irregularidades na atuação dos operadores, grau de reclamação dos usuários sobre o serviço e grau de ocorrência de acidentes são indicadores de grande interesse para esta dissertação,

pois os atuais métodos de coleta de informações são subjetivos (por exemplo, através das reclamações feitas pelos usuários) ou não são preventivos (por exemplo, indicadores tais como a média de quebra de veículos no percurso, taxa de acidentes com ou sem vítima).

## 1.4 Vibrações em ônibus

Segundo uma pesquisa feita pelo departamento de transporte dos Estados Unidos de America, [10], a vibração percebida pelos usuários de ônibus, sob condições normais, atingiram níveis de até 2G pico a pico; também foram atingidas vibrações superiores, perto de 4G, sob condições de operação não usuais (por exemplo, circulação do ônibus a 56 Km/h sobre uma serie de redutores de velocidade). O relatório descreve a vibração medida em dois ônibus, de iguais características excetuando-se a suspensão (o ônibus A tem amortecimento de ar enquanto o ônibus B tem amortecimento mecânico). No teste utilizaram-se rotas diferentes, desde rotas com vias suaves na cidade até rotas com vias esburacadas, incluindo vias com lombadas, tartarugas e pavimentos feitos de tijolo. Destes testes se observou que a diferença entre as suspensões tem pouca influência na amplitude dos sinais de aceleração; além disso, a amplitude da vibração é uma função da entrada de energia, a qual é controlada pela velocidade do ônibus e o tamanho da rugosidade ou altura da superfície irregular. Assim, a amplitude do sinal da aceleração varia proporcionalmente com o tamanho da rugosidade, defeito ou obstáculo e a velocidade do ônibus.

Três intervalos distintos de frequência são observados nos dados que foram coletados nos dois ônibus. As frequências mais baixas são observadas quando o veículo transita sobre uma serie de solavancos e o autor acredita que esse intervalo é devido à frequência natural do sistema de suspensão. O segundo e terceiro intervalo estão relacionados com a velocidade e as rugosidades e também se acredita que se devam às frequências naturais do corpo do veículo e do ponto de ligação do acelerômetro. Na Tabela 1.2 são dadas as frequências devidas à suspensão, o corpo do veículo e a fixação do acelerômetro. Os máximos obtidos das medições foram tabulados e traçados em um diagrama de

Tabela 1.2: Frequências recorrentes da aceleração em ônibus [10]

Ônibus	Amortecimento	Corpo do veículo	Componente de fixação
Ônibus A	1,5Hz	16-18 Hz	60-70 Hz
Ônibus B	2,0Hz	11-13 Hz	40-70 Hz

barras, mostrado por Thomas [10], onde são classificados os máximos segundo o posicionamento do sensor para diferentes estradas e condições de velocidade. Embora a parada de emergência esteja associada com uma aceleração próxima a 2G, esse não foi a maior medida obtida; nesse gráfico é possível observar que a aceleração é maior para vias com redutores de velocidade (tartarugas) ou

lombadas e são percebidas com maior intensidade no painel de controle para uma velocidade entre 40-56 Km/h.

## 1.5 Avaliação do conforto

### 1.5.1 Indicadores de conforto

Para avaliar o desconforto no transporte é necessário identificar os parâmetros que influem no conforto do passageiro e classificar sua fonte, com isto é possível estabelecer indicadores e faixas de classificação. Os fatores são ajustados através de experimentos que usam a pesquisa direta com o usuário como metodologia de medição. Segundo Förstberg [2], as grandezas físicas citadas na Tabela 1.3 influem no conforto em trens dependendo da característica do percurso e do tipo de avaliação técnica utilizada, abordados no conforto do trajeto no numeral 1.1.1.

Tabela 1.3: Grandezas com possível influência no conforto em trens [2]

Tipo de avaliação	Grandeza física
Conforto/desconforto médio do trajeto	Aceleração lateral Aceleração vertical Aceleração longitudinal Aceleração nas cadeiras
Distúrbios no conforto	
Percurso reto	Aceleração lateral
Curva de transição	Aceleração lateral(max, pico a pico) <i>Jerk</i> Velocidade de giro (Row, Yaw)
Curva circular	Aceleração lateral(média, pico a pico)
Eventos discretos	Aceleração lateral(média, pico a pico)
Enjoo	Aceleração lateral Aceleração vertical

Para os sistemas de transporte sem trilhos, os graus de liberdade são incrementados, assim para considerar os fatores que influenciam no conforto nos ônibus se deve incluir a aceleração longitudinal e vertical devido às mudanças na aceleração no tráfego, às imperfeições no asfalto, obstáculos tais como lombadas dentre outros. Assim, é possível realizar medições de perturbação de conforto.

Desta forma, para avaliar o conforto no sistema de transporte foram criados diferentes indicadores por parte de organismos internacionais, tais como a União Internacional de Ferrovias (UIC), o Instituto de Pesquisa Ferroviário Europeu (ERRI), o Comitê Europeu de Normalização (CEN) e a Organização Internacional para Padronização (ISO), assim como outros indicadores sugeridos por

pesquisadores da área de conforto como o indicador alemão,  $W_Z$ , usado no sistema de transporte da Suíça. Os indicadores sugeridos por Förstberg [2] para o monitoramento de conforto em trens são mostrados na Tabela 1.4.

Tabela 1.4: Indicadores de conforto em trens [2]	
Tipo de medição	Indicadores
Conforto médio	$N_{MV}, N_{VA}, N_{VD}$ ISO 2631, $W_Z$ , Ride index
Perturbações do conforto	
Percurso reto	$P_{DE}$
Curva de transição	$P_{CT}$
Curva circular	$P_{DE}$
Enjoo	$MSDV_Z$

A seguir são descritos sucintamente alguns dos indicadores usados em trens, baseados no resumo dos indicadores feito por Förstberg [2]; a norma técnica ISO2631-1 será descrita em mais detalhe na seção seguinte devido a sua importância neste trabalho e a seu amplo espectro de uso, não limitado a trens.

#### 1.5.1.1 Indicador $W_Z$

O indicador  $W_Z$  é definido por Sperling como [2]:

$$W_Z = \left[ \frac{1}{T} \int_0^T (B(f)a(t))^2 dt \right]^{0,15} \quad (1.1)$$

sendo que  $B(f)$  são pesos empíricos para diferentes componentes de frequência,  $a(t)$  são amplitudes máximas medidas no chão em  $\text{cm/s}^2$  e  $T$  é o tempo de avaliação. A função para determinar os pesos como função da frequência depende do eixo de medição, onde o eixo lateral é  $B_L(f) = 1,25B_V(f)$  e o eixo vertical é:

$$B_V(f) = 0,0588 \left[ \frac{1,911f^2 + (0,25f^2)^2}{(1 - 0,2777f^2)^2 + (1,563f - 0,0368f^3)^2} \right]^{0,5} \quad (1.2)$$

A Tabela 1.5 mostra o conceito para o conforto no percurso determinado pelo valor de  $W_Z$ .

#### 1.5.1.2 Indicador RI ou *Ride Index*

A versão inglesa do  $W_Z$  é chamada *Ride Index* a qual é definida para o eixo vertical como:

$$RI_Z = 7,5a_{RMS}^{0,3}(f_0/5,9)^{(1/3)} \text{ para } f_0 \leq 5,9\text{Hz}$$

$$RI_Z = 7,5a_{RMS}^{0,3}(5,9/f_0) \text{ para } f_0 > 5,9\text{Hz}$$

Tabela 1.5: Conceito de conforto para  $W_Z$  [2]

$W_Z$	Conceito para o percurso
1	Muito Bom
2	Bom
3	Satisfatório
4	Aceitável
4,5	Não aceitável
5	Perigoso

e para o eixo lateral é definida como:

$$RI_Y = 8,1 a_{RMS}^{0,3}(f_0/5, 9)^{(1/3)} \text{ para } f_0 \leq 5,9 \text{ Hz}$$

$$RI_Y = 8,1 a_{RMS}^{0,3}(5,9/f_0) \text{ para } f_0 > 5,9 \text{ Hz}$$

sendo que  $a_{RMS}$  é a aceleração RMS medida em G e dentro de uma faixa de frequências de uma oitava em torno de  $f_0$  a qual é a frequência central dessa faixa. Segundo o valor do RI o conceito para o conforto no percurso é mostrado na Tabela 1.6.

Tabela 1.6: Conceito de conforto para  $RI$  [2]

$RI$	Conceito para o percurso
1	Muito Bom
1,5	Quase muito Bom
2	Bom
2,5	Quase Bom
3	Satisfatório
3,5	Apenas satisfatório
4	Tolerável
4,5	Intolerável
5	Perigoso

### 1.5.1.3 CEN ENV 12299:1999

Segundo o pre-padrão europeu, o conforto pode ser avaliado mediante um método estatístico levando em conta a distribuição dos valores de aceleração RMS ponderados. Assim, três indicadores de medição são definidos: método padrão ( $N_{MV}$ ), passageiros sentados ( $N_{VA}$ ) e passageiros de pé ( $N_{VD}$ ), os quais são mostrados nas equações 1.3, 1.4 e 1.5, respectivamente.

$$N_{MV} = 6 \sqrt{(a_{XP95}^{Wd})^2 + (a_{YP95}^{Wd})^2 + (a_{ZP95}^{Wb})^2} \quad (1.3)$$

$$N_{VA} = 4 (a_{ZP95}^{Wb}) + 2 \sqrt{(a_{YP95}^{Wd})^2 + (a_{ZP95}^{Wb})^2} + 4 (a_{XP95}^{Wc}) \quad (1.4)$$



$$N_{VD} = 3\sqrt{16 (a_{XP55}^{Wd})^2 + (a_{YP50}^{Wd})^2 + (a_{ZP50}^{Wb})^2} + 5 (a_{YP95}^{Wd}) \quad (1.5)$$

sendo que  $a$  é a aceleração medida no eixo X,Y ou Z;  $W_i$  indica o filtro ponderado usado, no qual  $i$  pode ser o filtro vertical medido no eixo Z (b), horizontal medido nos eixos X e Y (d) ou horizontal medido no encosto da cadeira no eixo X (c). Os subscritos X,Y ou Z indicam o eixo do acelerômetro, enquanto que  $P,A$  ou  $D$  indicam o local de medição os quais são o chão, o assento ou o encosto da cadeira, respectivamente. Finalmente, o subscrito 50 ou 95 indica o percentil da distribuição dos valores RMS ponderados e amostrados a cada cinco segundos durante um teste de cinco minutos. Segundo o valor do indicador usado ( $N_{MV}$ ,  $N_{VA}$  ou  $N_{VD}$ ), o conceito para o conforto no percurso é mostrado na Tabela 1.7.

Tabela 1.7: Conceito de conforto para a norma CEN 1999 [2]

Critério	Conceito para o percurso
$N < 1$	Muito confortável
$1 \leq N < 2$	Confortável
$2 \leq N < 4$	Intermédio
$4 \leq N < 5$	Desconfortável
$5 \leq N$	Muito desconfortável

#### 1.5.1.4 Indicadores de distúrbio do conforto para deslocamentos de baixa frequência ( $P_{CT}, P_{DE}$ )

Resultados de pesquisas britânicas realizadas em trens de alta velocidade para avaliar a perda de conforto devido a distúrbios no movimento utilizaram dispositivos com um botão acionado pelo usuário, para detectar o distúrbio no conforto. Desenvolveram-se dois indicadores,  $P_{CT}$  e  $P_{DE}$ , para modelar os resultados obtidos nas transições das curvas e em eventos discretos, respectivamente.

O índice de conforto nas transições de curvas,  $P_{CT}$  foi definido como [2]:

$$P_{CT} = MAX(A\ddot{y} - BJ - C; 0) - D\dot{\varphi} \quad (1.6)$$

sendo que  $\ddot{y}$  é a máxima aceleração lateral absoluta desde que começa a curva até 1,6 segundos após da finalização dela. Enquanto que  $J$  é o *Jerk* máximo absoluto da curva, filtrado utilizando um filtro passa-baixas com frequência de corte ( $f_C$ ) de 2Hz e  $\dot{\varphi}$  é a máxima velocidade angular absoluta Roll na curva de transição. Os valores das constantes  $A$ ,  $B$ ,  $C$  e  $D$  são os mesmos utilizados por Förstberg, para pessoas de pé ou sentadas [2].

A definição do índice de conforto para eventos discretos  $P_{DE}$  é obtida através da equação mostrada a seguir:

$$P_{DE} = a\ddot{y}_p + b\ddot{y}_m - c \quad (1.7)$$

sendo que  $\ddot{y}_p$  é o valor absoluto da média da aceleração lateral, medido em uma janela de dois segundos e filtrado com um filtro passa-baixas com  $f_C$  de 2Hz. Enquanto que  $\ddot{y}_m$  é o valor pico a pico da aceleração lateral nas mesmas condições que  $\ddot{y}_p$ . Além disso, os valores das constantes  $a, b$  e  $c$  são os mesmos descritos por Förstberg [2], para pessoas de pé ou sentadas.

### 1.5.2 ISO 2631-1 [1]

O principal propósito da norma ISO 2631-1 é definir os métodos para quantificar a vibração em todo o corpo com relação à saúde humana, o conforto, a probabilidade de percepção da vibração e a geração de náuseas nos usuários.

Segundo varias pesquisas, a vibração em todo o corpo pode causar sensações (por exemplo, mal-estar), influir no desenvolvimento das atividades diárias ou produzir alto risco à saúde e segurança (dano patológico ou alteração fisiológica). Desta forma, a aplicação desta norma constitui um importante critério para avaliar um meio de transporte, um equipamento ou uma edificação. Assim, a norma ISO 2631 pode ser aplicada a movimentos transmitidos ao corpo através de uma superfície de apoio; contudo não pode ser aplicada na avaliação de choques de grandes amplitudes, tais como os observados em acidentes.

Para a medição de vibração deve-se usar como grandeza primaria a aceleração, a qual precisa ser medida no ponto no qual a vibração se propaga no corpo, assim os sensores devem ser localizados na interface entre o corpo e a fonte de vibração. A frequência da fonte de vibração determina o tipo de efeito que se quer medir; então para frequências entre 0,1Hz e 0,5Hz se mede o enjoo e entre 0,5Hz e 80Hz se mede o efeito na saúde, no conforto e o nível de percepção da vibração. Os sensores e os sistemas de processamento do sinal devem responder adequadamente nessa faixa de frequências e na faixa de intensidade da vibração. Um parâmetro importante é o tempo de medição o qual se deve reportar e deve ser como mínimo, para filtros passa-baixas com  $f_C$  de 1Hz, de 108s ou 227s para filtros passa-baixas com  $f_C$  de 0,5Hz; este tempo garante um erro inferior a 3dB e um nível de confiabilidade de 90%.

#### 1.5.2.1 Métodos para a avaliação da vibração

O **método básico** para a avaliação da vibração está baseado no cálculo da aceleração média quadrática (RMS) ponderada, cuja equação é descrita a seguir:

$$a_w = [1/T \int_0^T a_w^2(t) dt]^{1/2} \quad (1.8)$$

sendo que  $a_w$  é a aceleração ponderada ( $m/s^2$ ) e  $T$  é a duração da medição (segundos).

O Fator de Cresta ( $CF$ ) está definido como:

$$CF = |Max \{a_w(t)\} / a_w| \quad (1.9)$$

O  $CF$  serve para determinar se o método de avaliação básico é adequado ( $CF < 9$ ) ou talvez seja melhor trabalhar com outros métodos, tais como valor **RMS operante** ou **dose de vibração à quarta potência**. O método de RMS operante leva em consideração choques ocasionais e vibrações transitórias, este método utiliza como indicador o valor da vibração transitória máxima (MTVV) definida como:

$$MTVV = Max \{a_w(t_0)\} = Max \left\{ \left[ 1/\tau \int_{t_0}^{t_0+\tau} a_w(t)^2 dt \right]^{1/2} \right\} \quad (1.10)$$

Enquanto que o método de dose de vibração à quarta potência (VDV) é mais sensível aos picos de aceleração do que o método básico. O VDV é definido como:

$$VDV = \left[ \int_0^T a_w(t)^4 dt \right]^{1/4} \quad (1.11)$$

### 1.5.2.2 Ponderação do sinal de aceleração

Para a ponderação do sinal de aceleração na frequência utilizam-se diferentes tipos de coeficientes; os quais são descritos a seguir:

- $w_k$  para a direção no eixo z medida no assento da cadeira ou no chão do ônibus.
- $w_d$  para a direção nos eixos x e y medidas no assento da cadeira ou no chão do ônibus.
- $w_c$  para as medições feitas no encosto da cadeira (eixos x, y e z).
- $w_e$  para a medição da vibração rotacional no apoio da cadeira ou no chão do ônibus.
- $w_j$  para a medição da vibração sob a cabeça de uma pessoa deitada.

A ponderação da frequência pode ser modelada como uma função de transferência cujos componentes são:

- Filtro passa-altas de segunda ordem:

$$|H_h(p)| = \left| \frac{1}{1 + \sqrt{2}\omega_1/p + (\omega_1/p)^2} \right| = \sqrt{\frac{f^4}{f^4 + f_1^4}} \quad (1.12)$$

- Filtro passa-baixas de segunda ordem:

$$|H_l(p)| = \left| \frac{1}{1 + \sqrt{2}p/\omega_2 + (p/\omega_2)^2} \right| = \sqrt{\frac{f_2^4}{f^4 + f_2^4}} \quad (1.13)$$

- Transição aceleração-velocidade (proporcionalidade para a aceleração em frequências baixas e proporcionalidade para a velocidade em frequências altas):

$$|H_t(p)| = \left| \frac{1 + p/\omega_3}{1 + p/(Q_4\omega_4) + (p/\omega_4)^2} \right| = \sqrt{\frac{f^2 + f_3^2}{f_3^2}} \sqrt{\frac{f_4^4 Q_4^2}{f^4 Q_4^2 + f^2 f_4^2 (1 - 2Q_4^2) + f_4^4 Q_4^2}} \quad (1.14)$$

- Etapa ascendente (Inclinação de aproximadamente 6dB por oitava, proporcional ao diferencial de aceleração):

$$|H_s(p)| = \left| \frac{1 + p/(Q_5\omega_5) + (p/\omega_5)^2}{1 + p/(Q_6\omega_6) + (p/\omega_6)^2} \left( \frac{\omega_5}{\omega_6} \right)^2 \right| = \frac{Q_6}{Q_5} \sqrt{\frac{f^4 Q_5^2 + f^2 f_5^2 (1 - 2Q_5^2) + f_5^4 Q_5^2}{f^4 Q_6^2 + f^2 f_6^2 (1 - 2Q_6^2) + f_6^4 Q_6^2}} \quad (1.15)$$

Assim, o produto entre  $H_h(p)H_l(p)$  representa a função de transferência de limitação de banda; enquanto que o produto  $H_t(p)H_s(p)$  representa a ponderação real. Os parâmetro para essas funções de transferência são dados na Tabela 1.8.

Tabela 1.8: Parâmetros das funções de transferência utilizados pela norma ISO 2631-1 [1]

Ponderação	Limitação de banda		Transição a-v			Etapa ascendente			
	$f_1(\text{Hz})$	$f_2(\text{Hz})$	$f_3(\text{Hz})$	$f_4(\text{Hz})$	$Q_4$	$f_5(\text{Hz})$	$Q_5$	$f_6(\text{Hz})$	$Q_6$
$w_k$	0,4	100	12,5	12,5	0,63	2,37	0,91	3,35	0,91
$w_d$	0,4	100	2,0	2,0	0,63	$\infty$	-	$\infty$	-
$w_f$	0,08	0,63	$\infty$	0,25	0,86	0,0625	0,80	0,1	0,80
$w_c$	0,4	100	8,0	8,0	0,63	$\infty$	-	$\infty$	-
$w_e$	0,4	100	1,0	12,5	0,63	$\infty$	-	$\infty$	-
$w_j$	0,4	100	$\infty$	$\infty$	-	3,75	0,91	5,325	0,91

Para  $w_j$  a função de transferência é  $H_t(p) = 1$  e para  $w_d, w_c$  e  $w_e$  é  $H_s(p) = 1$ . Concluindo-se a função de ponderação total é:

$$H(p) = H_h(p)H_l(p)H_t(p)H_s(p) \quad (1.16)$$

### 1.5.2.3 Norma ISO 2631-1 aplicada ao conforto [1]

Em alguns casos as interferências em algumas atividades tais como ler, escrever ou beber, devidas às vibrações são consideradas causa de mal-estar. Entretanto estas interferências dependem de muitos fatores tais como: grupo social, condições de saúde, condições dinâmicas e outros. Desta forma, alguns destes fatores não se encontram dentro do alcance da dissertação nem da norma ISO 2631-1. No entanto, em ambientes nos quais o fator de interesse são as propriedades mecânicas, é possível avaliar os efeitos da vibração sobre o conforto usando a aceleração RMS de frequência ponderada em um período de tempo representativo. Esta avaliação deve ser feita em cada eixo, calculando o valor RMS produzido pela combinação dos três eixos em cada local de medição, segundo a seguinte equação:

$$a_v = \sqrt{K_x^2 a_{wx}^2 + K_y^2 a_{wy}^2 + K_z^2 a_{wz}^2} \quad (1.17)$$

sendo que  $K_x$ ,  $K_y$  e  $K_z$  são fatores multiplicativos dependentes do local de medição. Caso o conforto seja afetado por vibrações em mais de um local, o valor total pode ser calculado através do valor quadrático resultante dos valores totais para cada ponto. A medição precisa ser realizada para todas as posturas possíveis, então para as pessoas que se encontrem sentadas deve-se utilizar doze eixos de medição (três translacionais e três rotacionais no apoio da cadeira, além de três eixos translacionais no espaldar da cadeira e nos pés). Os valores para o fator multiplicador,  $K$ , são dados na Tabela 1.9.

Tabela 1.9: Valores multiplicativos,  $K$ , para pessoas sentadas.

local	eixo	$k$
Superfície da cadeira	$w_d(x)$	1
	$w_d(y)$	1
	$w_k$	1
	$w_e(r_x)$	0,63
	$w_e(r_y)$	0,4
	$w_e(r_z)$	0,2
No espaldar da cadeira	$w_c(x)$	0,8
	$w_c(y)$	0,5
	$w_c(z)$	0,4
Nos pés	$w_k(x)$	0,25
	$w_k(y)$	0,25
	$w_k(z)$	0,4

Nas medições feitas com pessoas transportadas de pé somente são consideradas as vibrações no chão. Assim, os fatores multiplicativos são:

- $w_d(x), k = 1$
- $w_d(y), k = 1$
- $w_k, k = 1$

A vibração horizontal no encosto da cadeira em veículos pode afetar significativamente o conforto. Então, se por razões técnicas não for possível medir a vibração no encosto, pode-se aumentar o fator multiplicador para 1,4 nos eixos X e Y na superfície do suporte. Finalmente, a norma técnica recomenda os índices mostrados na Tabela 1.10 para a avaliação do conforto no transporte público.

Tabela 1.10: Índices recomendados pela norma ISO 2631-1 para a avaliação do conforto no transporte público

Aceleração RMS	Conceito
Menos de 0,315m/s <sup>2</sup>	Sem mal-estar
0,315m/s <sup>2</sup> - 0,63m/s <sup>2</sup>	Mal-estar leve
0,5m/s <sup>2</sup> - 1m/s <sup>2</sup>	Mal-estar moderado
0,8m/s <sup>2</sup> - 1,6m/s <sup>2</sup>	Mal-estar
1,25m/s <sup>2</sup> - 2,5m/s <sup>2</sup>	Mal-estar alto
Maior de 2 m/s <sup>2</sup>	Mal-estar muito alto

#### 1.5.2.4 Norma ISO 2631-1 aplicada à vertigem

O movimento em frequências inferiores a 0,5Hz pode causar efeitos indesejáveis, tais como enjoo. Com a aceleração RMS ponderada na faixa de frequências entre 0,1 e 0,5Hz no eixo z é possível obter um indicador para a probabilidade de geração de náuseas nos passageiros do veículo. A aceleração nos outros eixos é desprezível porque produzem pouco impacto neste efeito. Este conceito é aplicável principalmente em navios e embarcações marítimas [1].

## 1.6 Sistemas para a medição de conforto

### 1.6.1 Monitor de solavancos para transporte de mercadorias baseado na medição da aceleração e da posição através do uso de GPS

A pesquisa feita pelo Instituto Nacional para a Segurança Ocupacional e Saúde dos Estados Unidos (NIOSH) [24], busca melhorias no ambiente de trabalho dos motoristas de caminhões de mercadorias. Assim, com informações obtidas do GPS e dados de aceleração é possível determinar a causa, quantidade e intensidade de solavancos que afetam negativamente a saúde do motorista do caminhão,

também como monitorar o estilo e habilidade do motorista para dirigir. Adicionalmente, a combinação dos dados do GPS e aceleração pode ser usada como uma ferramenta de auxílio à manutenção das estradas e dos veículos.

O hardware usado neste trabalho é composto por dois módulos, o primeiro é produzido pela companhia *Modular Mining Systems* (MMS) chamado *Vital Signs*, o qual é utilizado no monitoramento de fretes via GPS. O segundo equipamento é um *Acceleration Recorder* da *Aero-Marine products* (AMP), o qual é frequentemente utilizado no monitoramento da aceleração no transporte de mercadorias, suas principais características técnicas são apresentadas na Tabela 1.11. Os resultados obtidos em um percurso com duração de 10 horas e 20 minutos mostraram 899 solavancos acima do limiar de 2G, concentradas em regiões diferentes do percurso e são mostrados espacialmente na Figura 1.3. Deste modo, com os dados obtidos é possível monitorar a existência de buracos e obstáculos ao longo da estrada. Além disso, também é possível estabelecer faixas limite de velocidade devido às condições do asfalto ou a indicadores de alta probabilidade de acidentes.

Finalmente, os autores propõem, como trabalho futuro, o desenvolvimento de um sistema com acelerômetros tri-axiais de baixo range que permitam monitorar em tempo real as acelerações fornecendo uma realimentação ao motorista através de um alarme quando o limiar for atingido.

Tabela 1.11: Especificações do gravador AMP

Modelo	1005083-1
Tipo de Sensor	Acelerômetro Piezelétrico tri-axial
Range	$\pm 2$ a 200G em cada eixo
Resolução	10 Bits
Período de amostragem	1 a 50mseg
Frequência mínima	1.5Hz
Frequência máxima	90Hz

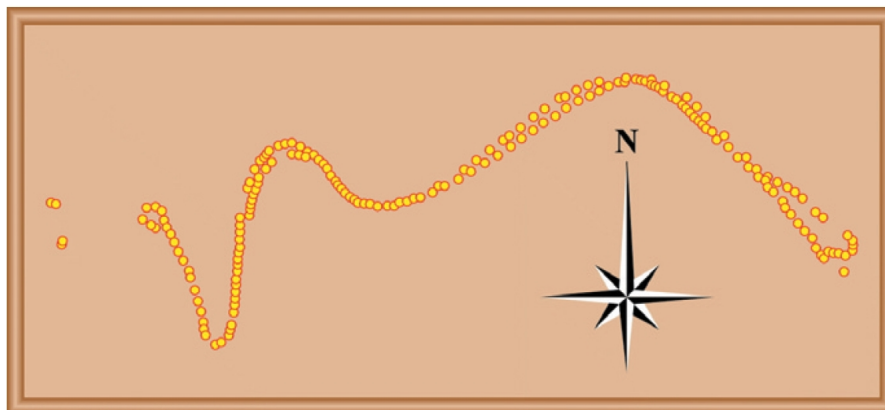


Figura 1.3: Eventos de aceleração usando o gravador AMP e o GPS do MMS.

### 1.6.2 Sistema baseado em rede de sensores para o monitoramento das condições da via no transporte público

Zoysa et al. [25] desenvolveram um sistema para o monitoramento das condições da via baseado em uma rede de sensores instalados no sistema de transporte público. O sistema permite criar mecanismos de alarme para realizar manutenção preventiva nas estradas que reduzirá o custo total e fornecerá dados para pesquisadores que estudam o desgaste do asfalto das estradas.

Os pesquisadores propõem usar a rede de transporte do Sri Lanka para monitorar as condições do asfalto, baseado em outro sistema desenvolvido para monitorar a poluição ambiental, chamado BusNet. Deste modo, o sistema de transporte público pode ser usado como plataforma para uma rede de sensores diminuindo os custos de desenvolvimento e de execução, pois pesquisas prévias mostraram que é necessária uma grande quantidade de sensores para cobrir uma área suficiente para obter medições ambientais bem sucedidas, por exemplo, para cobrir uma área de 1200m por 600m se utilizam 110 sensores. Este tipo de rede apresenta os seguintes problemas:

- Alto custo devido à grande quantidade de sensores dedicados.
- Dificuldade para o gerenciamento de grandes quantidades de nós, além de problemas tais como a substituição de baterias ou a manutenção dos nós.
- A proteção dos nós de fatores como animais, chuvas e vândalos, pode ser tarefa complicada.

Assim, os sensores montados nos veículos de transporte público recolhem dados durante o percurso e os transferem nas estações através de uma comunicação sem fio. Também os dados recolhidos nas estações são transferidos até uma estação central através dos ônibus que viajam até a estação central, o qual fornece uma largura de banda grande sacrificando a latência da rede. A arquitetura do sistema é mostrada na Figura 1.4. O sistema de medição é formado por um dispositivo chamado *MICAz mole* da companhia *Crossbow Technology* (mostrado na Figura 1.5), um GPS e um sensor; sendo que o sensor pode ser de temperatura,  $CO_2$  ou um acelerômetro. A energia de alimentação é fornecida pela bateria do veículo e os dados são armazenados na memória do MICAz.

Alguns dos inconvenientes para a arquitetura do BusNet mencionados pelos autores são:

- As medições feitas em sensores em movimentação têm menor exatidão espacial do que as medições feitas com sensores estacionários.
- Alguns sensores requerem vários segundos para prover uma medição com boa exatidão, assim os eventos com posições pontuais não são possíveis de se registrar.



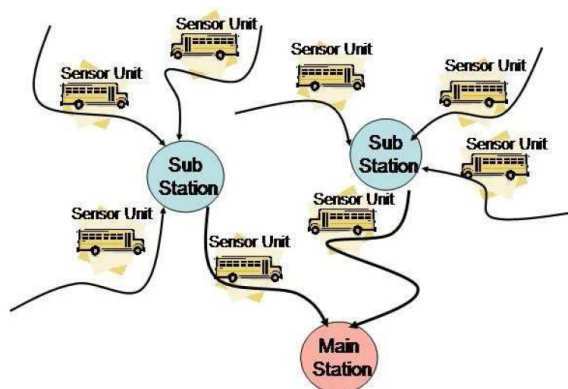


Figura 1.4: Arquitetura do sistema BusNet.



Figura 1.5: Protótipo de medição com o MICAz e a placa do sensor.

### 1.6.3 Detector portátil de condução imprudente

Boonmee e Tangamchit [4], desenvolveram um sistema para detectar comportamentos imprudentes na condução do motorista através de um acelerômetro bi-axial. O sistema pode identificar eventos tais como: fortes arranques, freadas repentinas, giros rápidos dentre outros, os quais causam desconforto.

Este detector é acondicionado no encosto da cadeira do motorista e permite monitorar o comportamento do motorista em situações reais, o diagrama de blocos do hardware é mostrado na Figura 1.6. O acelerômetro analógico utilizado foi o ADXL202, o qual é bi-axial, com range de 2G e uma largura de banda de 6kHz. Os eixos X e Y são utilizados para medir a aceleração longitudinal e transversal, respectivamente. O micro-controlador utilizado para realizar a interface entre o acelerômetro e o PC foi o PIC16F877 de 8 bits. Os sinais obtidos pelo PC são processados como mostrado na Figura 1.7. Um filtro passa altas de primeira ordem com frequência de corte de 0,02Hz é usado para eliminar o componente DC e um filtro passa baixas de primeira ordem com frequência de corte de 2Hz é

utilizado para eliminar os ruídos causados pelos solavancos.

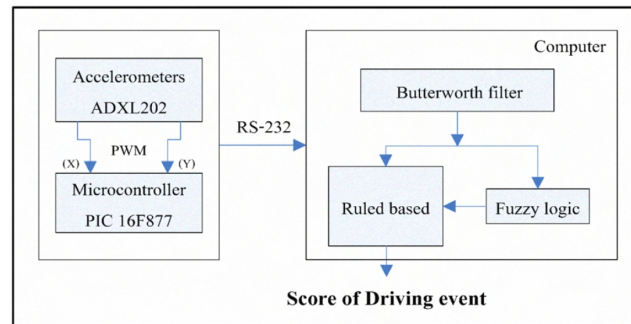


Figura 1.6: Diagrama de blocos do detector portátil de condução imprudente. [4]

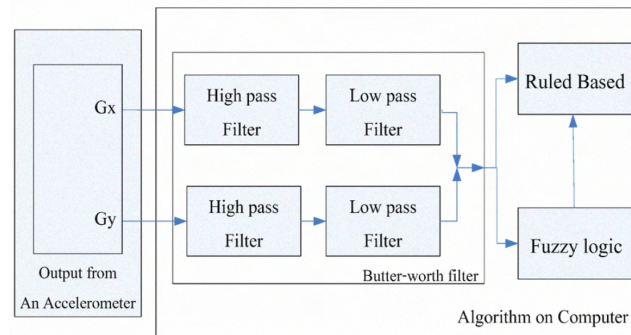


Figura 1.7: Diagrama de blocos do processamento para o sinal de aceleração. [4]

Os autores utilizaram um método baseado em regras para identificar os eventos e utilizaram uma lógica difusa para dar pontuação aos comportamentos. Então, o uso da lógica difusa permite ajustar as regras e o range do dispositivo à pontuação dada por voluntários usados como referência. Primeiramente a identificação da causa do evento foi feita usando os critérios de duração e amplitude mostrados na Tabela 1.12, sendo que  $t_d$  é o tempo de duração e é utilizado para diferenciar os tipos de eventos. Depois os resultados dos questionários dos 11 voluntários foram utilizados para ajustar as regras que estabelecem a relação entre a aceleração e a percepção do passageiro. As entradas do sistema difuso e as regras são mostradas na Tabela 1.12 [4].

Tabela 1.12: Regras para determinar a fonte do evento

Fonte do evento	Aceleração	Tempo ( $t_d$ )
Arranque	$A_x$ maior do que 87.5mG	menor que 275ms
Freada	$A_x$ menor do que -87.5mG	menor que 320ms
Giro à esquerda	$A_y$ maior do que 150mG	menor que 275ms
Giro à direita	$A_y$ menor do que -150mG	menor que 275ms

A pontuação dada pelos 11 voluntários no questionário está dentro de um range de 1 a 10, onde 1 é um evento muito suave e 10 é um evento muito forte. Os resultados dos testes feitos em Bangkok são mostrados na Figura 1.8, na qual a media da pontuação, foi usada para comparar a avaliação feita pelos usuários com o conceito fornecido pela predição do sistema difuso. Os erros entre a predição e os dados dos usuários atingiram 11,96; 12,79; 11,12 e 15,6 para arrancadas, freadas, giros à direita e giros à esquerda, respectivamente; assim, o modelo da percepção dos usuários atingiu 80% de exatidão.

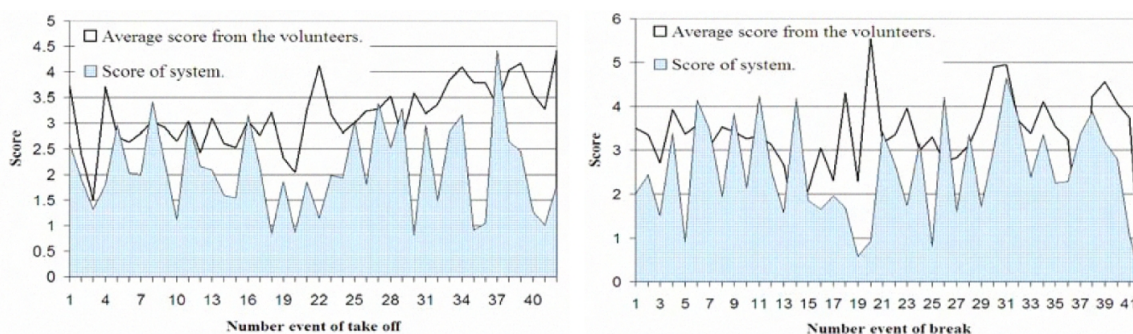


Figura 1.8: Resultados dos testes feitos em Bangkok com o sistema desenvolvido por Boonmee e Tangamchit e para a pontuação dada pelos usuários para arranques e freadas do veículo [4].

#### 1.6.4 Avaliação de risco nos veículos utilizando o aparelho celular Nokia™ N95

Reveriego et al. [26] apresentaram um sistema de identificação de zonas perigosas nas vias ou situações perigosas causadas por erros do motorista, baseados no uso do GPS e do acelerômetro incorporados ao celular NOKIA™ N95. Esta aplicação captura todos os dados necessários no percurso para sua posterior análise em uma interface de computador baseada em Google Earth™.

Para obter a informação da posição do local com alto risco de acidente os dados do GPS são coletados utilizando rotinas no sistema operacional Symbian™ do N95. Dados tais como: hora, longitude, latitude e velocidade são registrados a uma taxa de 1Hz e são armazenados no cartão de memória do aparelho. Posteriormente os dados da memória do aparelho são descarregados no PC e processados para gerar um arquivo *Keyhole Markup Language* (KML) capaz de visualizar os dados em Google Earth™.

Para medir a aceleração do veículo é utilizado o acelerômetro embarcado no N95, cujos planos de referência são mostrados na Figura 1.9. A medição da aceleração não é feita com taxa constante, assim os autores chegaram a um sinal com uma frequência de amostragem de 40Hz, usando uma interpolação linear. O nível 5 da decomposição Wavelet, mostrada na Figura 1.10, foi reconstruída

para filtrar o ruído do sinal; porque a informação importante sobre a maneira de dirigir e freadas está nos componentes de frequência mais baixa.



Figura 1.9: Marco de referencia para o veículo e o telefone Nokia™ N95.

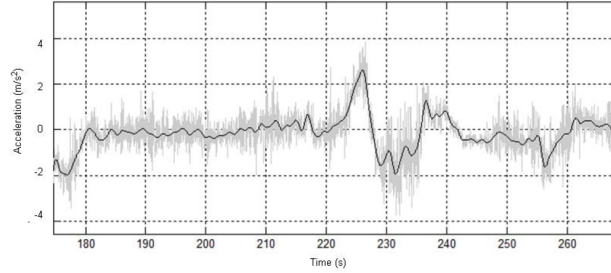


Figura 1.10: Exemplo do sinal de aceleração (Cinza) e sua reconstrução baseada na Wavelet de quinto nível (preto).

Outro problema que foi solucionado neste trabalho foi o ajuste dos eixos do aparelho aos eixos de movimentação do veículo, desta forma, o usuário pode colocar o telefone em qualquer posição sem se preocupar do alinhamento dos eixos do celular com os eixos do aparelho. Devido a informação dos giros, freadas e arranques estarem contidas nos eixos X e Y, o eixo Z de aceleração pode ser desprezado; assim, reduzindo-se o análise a um problema bidimensional. O dado de velocidade, medido pelo GPS, é utilizado para obter a aceleração longitudinal, pois a velocidade é derivada para obtenção de aceleração longitudinal. Desta maneira, com a relação entre a aceleração calculada pelo GPS e aceleração medida pelo acelerômetro é possível calcular o ângulo de rotação dos eixos,  $\alpha$ , como mostrado na equação 1.18.

$$\alpha = \text{Max}_{\theta} \langle a_{XGPS}, a_{uri} \rangle \quad (1.18)$$

sendo,  $a_{XGPS}$  a aceleração calculada através da leitura da velocidade dada pelo GPS e  $a_{uri}$  é a aceleração de um dos eixos com a correção devida a possíveis inclinações ao redor do eixo Z com ângulo  $\theta$ , definido como:  $a_{uri} = a_{ur} \cos(\theta_i) - a_{ur} \sin(\theta_i)$ . Estimando o ângulo  $\alpha$ , os eixos de movimentação

do carro podem ser conhecidos. O diagrama de blocos do processamento do sinal de aceleração é mostrado na Figura 1.11.

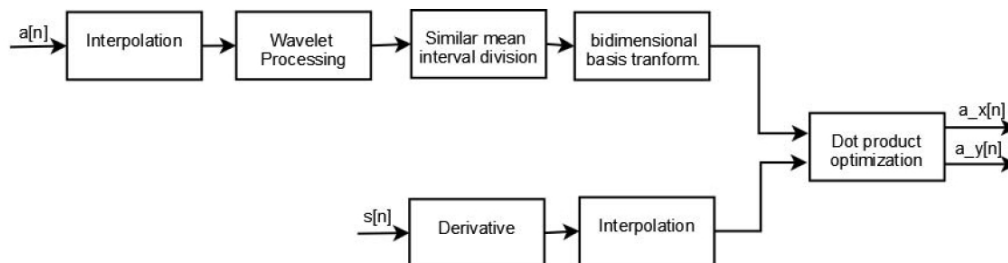


Figura 1.11: Diagrama de blocos do procesamento.

O risco determinado pela aceleração do aparelho é catalogado em três faixas (baixa, media e alta) o qual é determinado através do uso de limiaries da aceleração transversal. O valor adequado dos limiaries foi calculado através de avaliações feitas junto a vários voluntários. Na Figura 1.12 se mostra um exemplo de aquisição da aceleração, onde a altura do polígono reflete a velocidade do veículo, enquanto que a largura e a cor mostram a aceleração transversal e o nível de risco respectivamente.



Figura 1.12: Representação tri-dimensional da aceleração da rotatória.

Na Figura 1.13 são mostrados dois exemplos de análise feitos com o sistema desenvolvido por Reveriego. O primeiro mostra quando o motorista se deteve na interseção para fazer uma virada à direita; enquanto o segundo mostra a comparação de dois motoristas, no mesmo percurso e com o mesmo veículo.

Trabalhos futuros são propostos pelo autor, tais como a geração de alarmes para prevenir o motorista e a divulgação de dados pelas redes sociais para permitir a outros motoristas ter acesso à informação sobre os locais com alto grau de risco na via.





Figura 1.13: Comparação usando o N95. (a) Parada em uma intersecção. (b) Comparação da velocidade de dois motoristas.

# Capítulo 2

## Estudo dos sinais envolvidos

Neste capítulo são analisadas as características dos sinais de aceleração de ônibus nos três eixos cartesianos, coletando dados tanto de linhas convencionais como de linhas BRT. Análises no domínio do tempo permitirão estabelecer a faixa de aceleração observada nos veículos de transporte. Análises no domínio da frequência serão usadas para estabelecer a largura de banda (BW) e frequência de amostragem ( $F_S$ ), necessárias para medir satisfatoriamente o sinal da aceleração. Além disso, alguns componentes do sinal da aceleração são identificados e associados às características do veículo, o comportamento do modo de dirigir do motorista ou à diferentes tipos de fonte de desconforto. Finalmente, o sinal de aceleração será analisado para obter o *Jerk* e desta maneira relacioná-lo com o nível de desconforto e conhecer os requisitos de processamento para seu cálculo.

### 2.1 Características do sinal de aceleração em veículos

Com o objetivo de conhecer as características dos sinais de aceleração nos veículos de transporte foram desenvolvidos dois protótipos para obter amostras de sinais de aceleração nos três eixos com uma alta taxa de amostragem (maior que 500Hz). As amostras coletadas em meios de transporte distintos foram processadas usando o software LabView™.

#### 2.1.1 Descrição do Hardware, firmware e resultados do primeiro protótipo

O hardware é constituído por um microcontrolador ATMEL® de oito bits (ATMEGA8), cuja função é armazenar a informação proveniente do conversor analógico-digital (ADC) de oito bits e frequência de amostragem  $F_S$  de 1040Hz, em uma memória SD de 2GB. A Figura 2.1 mostra o primeiro protótipo em detalhe, enquanto que o diagrama esquemático é mostrado na Figura 2.2.

Um acelerômetro analógico capacitivo de baixo custo com fundo de escala ajustável em  $\pm 1,5G/2G/4G$



Figura 2.1: Módulo de aquisição baseado em um microcontrolador de 8 bits.

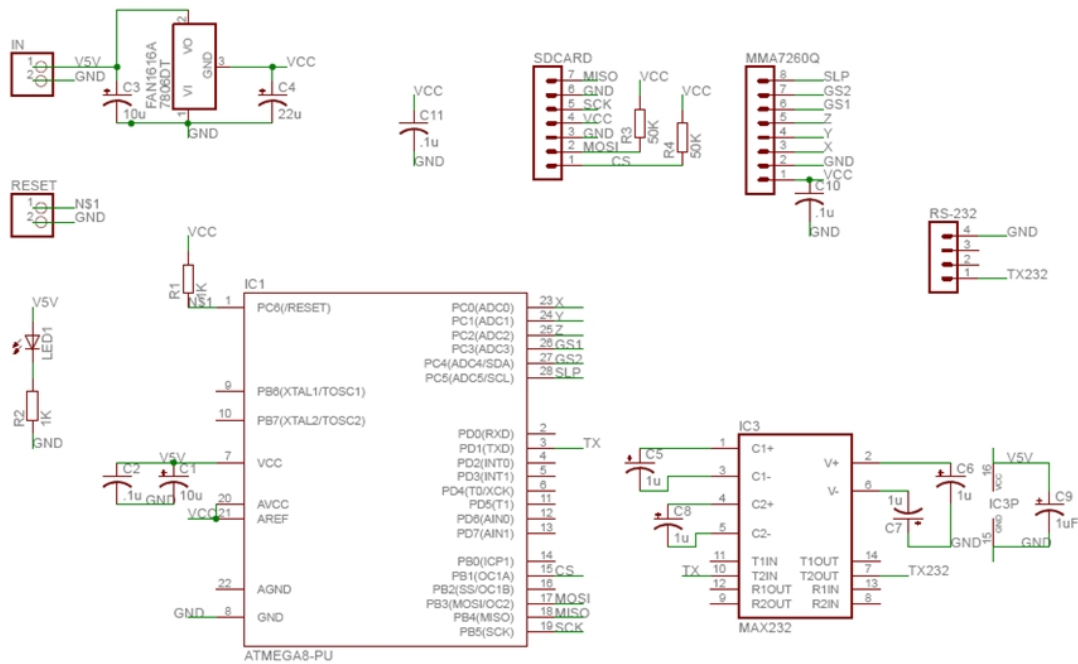


Figura 2.2: Esquemático do módulo de aquisição.



ou 6G (MMA7260Q) que possui modo *sleep* e tem um baixo consumo de potência ( $500\mu A$ ). Neste caso, utilizamos o microcontrolador para controlar o fundo de escala do acelerômetro, o qual pode ser selecionado entre 2G e 4G. Finalmente, para seu adequado funcionamento é necessário implementar um filtro RC de 1,6kHz nas suas saídas para minimizar o ruído de *clock*.

Devido à facilidade de fixação do adaptador SD à placa de circuito impresso do protótipo, optou-se pela utilização de uma memória microSD. A sequência de inicialização da memória e os comandos para armazenar ou ler a informação podem ser utilizados em uma memória SD ou microSD de qualquer tipo. A memória deve ser conectada às saídas do módulo SPI do microcontrolador. Uma limitação da memória SD restringe a escrita de blocos de dados de 512 posições fixas, não sendo possível aumentar ou diminuir a quantidade de dados em cada bloco.

O *firmware* do microcontrolador permite aquisitar o sinal de aceleração com  $F_S$  de 1040Hz, levando em conta que a limitação de banda do acelerômetro é de 300Hz para os eixos X e Y e de 150Hz para o eixo Z, um filtro *antialiasing* não é necessário. Blocos de 512 dados, alternando a informação do eixo X,Y,Z, são enviados ao cartão de memória a través do protocolo SPI a 125kHz. Para evitar a perda de dados, um buffer maior a 512 dados foi implementado no microcontrolador para armazenar tanto o bloco de 512 dados como os dados de aceleração que são aquisitados enquanto a memória está sendo escrita.

### 2.1.1.1 Resultado experimental obtido com o primeiro protótipo - Carro No. 1

O sistema foi testado no assoalho de dois carros diferentes e em percursos diferentes. Na Figura 2.3 é mostrada a disposição do protótipo colocado no assoalho do veículo.

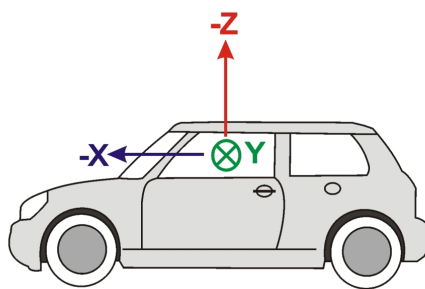


Figura 2.3: Orientação dos eixos cartesianos do acelerômetro no assoalho do carro.

A aquisição dos dados de aceleração no primeiro carro teve uma duração aproximada de 53 minutos, obtendo-se os resultados mostrados na Figura 2.4, as amplitudes percebidas nos percursos estão em torno de 1,25G para os eixos do plano horizontal (X, Y) e 2G para o eixo vertical (Z). Além disso, uma análise em frequência dos sinais nos três eixos mostrou que a informação da aceleração está altamente concentrada nos primeiros 50Hz, a FFT dos três eixos é mostrada na Figura 2.5.

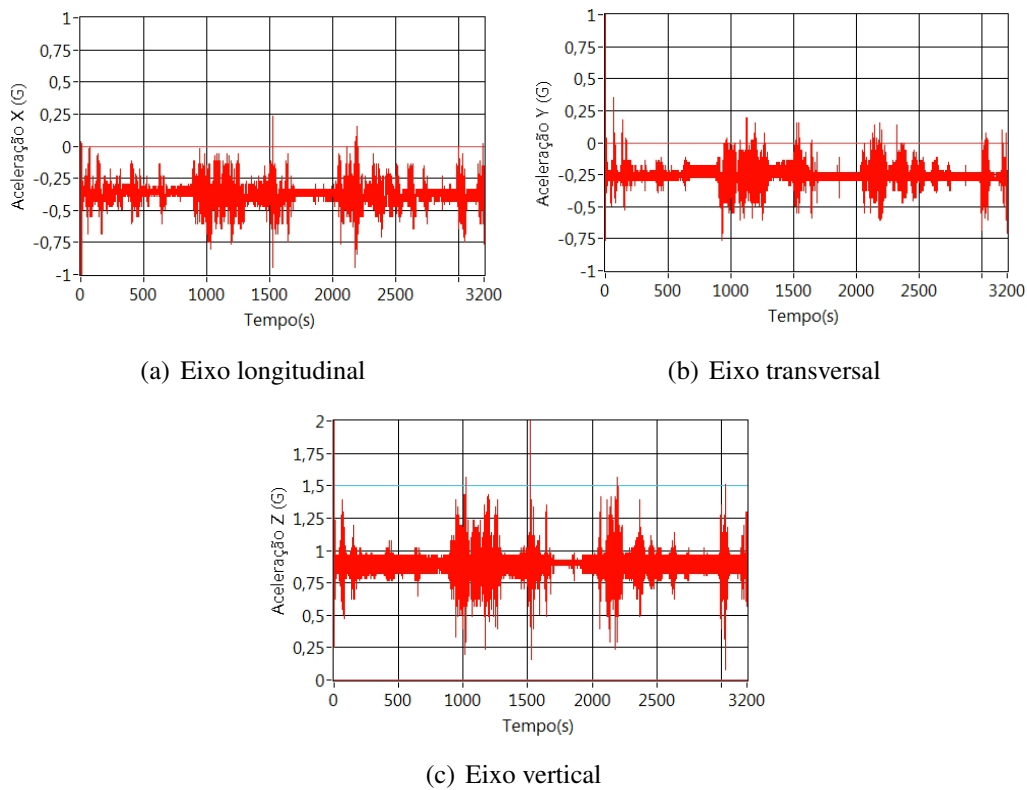


Figura 2.4: Análise carro No. 1.

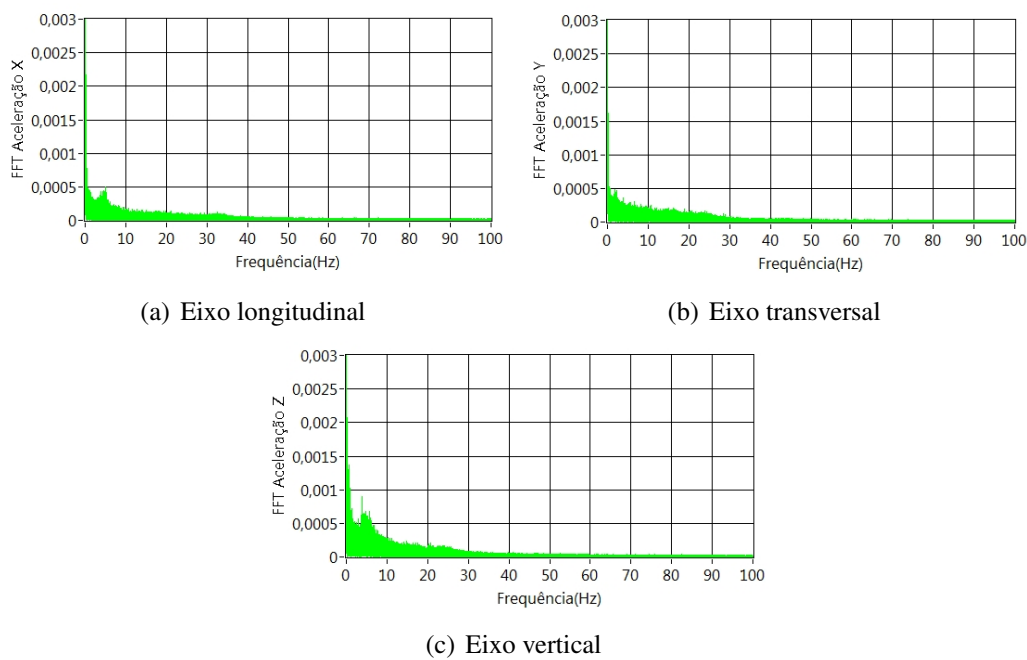


Figura 2.5: FFT da aceleração do teste no carro No. 1.

Analisando os picos da aceleração frontal e vertical, percebe-se que a amplitude pico a pico é superior a 1G com uma duração de 0,05s acima de 1,5G no eixo vertical Z e abaixo de -0,4G no eixo frontal. A Figura 2.6 mostra com maior detalhe as acelerações frontal e vertical medidas.

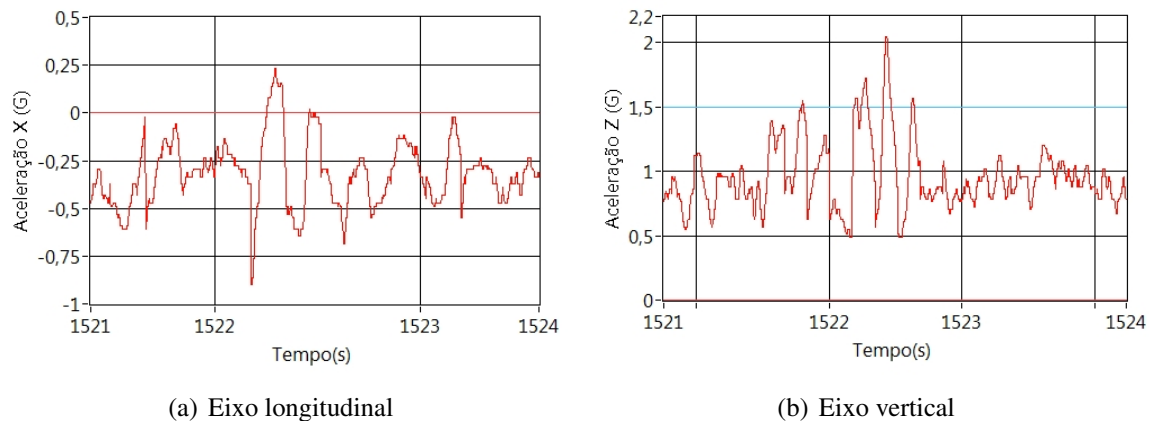


Figura 2.6: Sinais de aceleração do carro No. 1.

### 2.1.1.2 Resultado experimental obtido com o primeiro protótipo - Carro No. 2

Usando outro veículo e realizando uma análise similar obtiveram-se os resultados mostrados na Figura 2.7. Os sinais de aceleração do eixo lateral e frontal estão na faixa de 0,8G e 0,7G pico a pico respectivamente; enquanto que o maior pico está presente no eixo vertical com um valor superior a 1,5G com uma duração de 10ms, como mostrado na Figura 2.8.

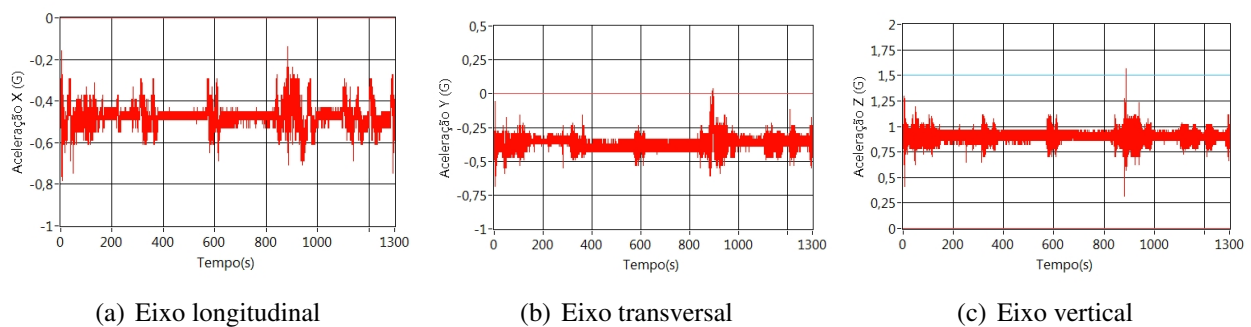


Figura 2.7: Análise carro No. 2.

### 2.1.1.3 Resultado experimental obtido com o primeiro protótipo - Sistema BRT

Este primeiro protótipo foi também utilizado nos veículos no sistema de transporte público coletivo. O sistema BRT tem características dinâmicas que diminuem o desconforto nos passageiros,

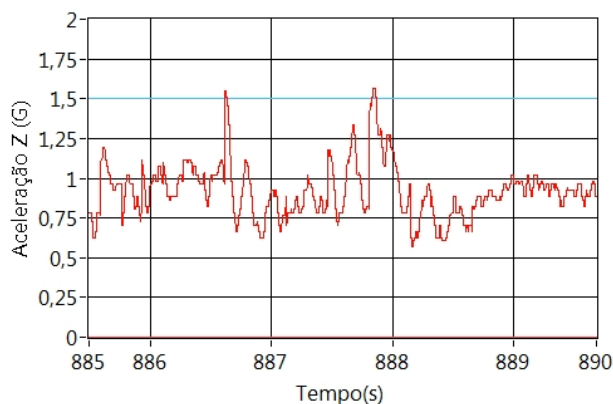


Figura 2.8: Máxima aceleração no eixo vertical, no análise do carro No. 2.



Figura 2.9: Sistema BRT, nomeado Transmilenio em Bogotá. Colômbia.

tais como faixas exclusivas, paradas programadas, entre outras; por isso, seu estudo é de interesse para conhecer as amplitudes do sinal que causam desconforto em um sistema com menor número de distúrbios que o sistema de transporte típico. O *Transmilenio* mostrado na Figura 2.9, é um sistema BRT desenvolvido para diminuir o tempo da viagem, a taxa de acidentes e a poluição de Bogotá (Colômbia) e foi escolhido para estes primeiros testes.

O protótipo foi utilizado em uma das principais rotas do sistema com o maior número de paradas (rota B5/G5). Os resultados obtidos para esta rota com o acelerômetro na faixa de 2G são mostrados na Figura 2.10. Note-se que neste caso o eixo Y tem a informação da aceleração na direção do avanço do veículo (eixo longitudinal), como é mostrado pela Figura 2.11.

Assim, é possível ver que o ônibus fez seis paradas em um percurso de 21 minutos; em média o tempo entre estações é de: 2,89min. A maior aceleração no eixo de avanço foi de 0,75G; enquanto que no eixo vertical foi de 1,8G. Na Figura 2.12 observa-se alguns sinais obtidos na faixa de 4G. Apesar da menor resolução na faixa de 4G é possível detectar os picos de aceleração, especialmente no eixo vertical; onde a maior amplitude foi de 2,5G.

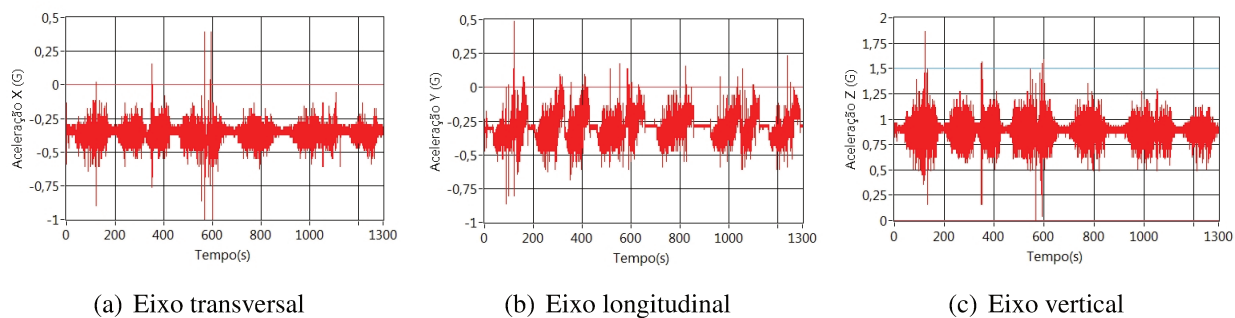


Figura 2.10: Aceleração no sistema BRT com acelerômetro na faixa de 2G.

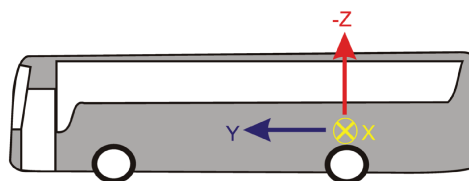


Figura 2.11: Orientação dos eixos cartesianos do acelerômetro no assoalho do ônibus BRT.

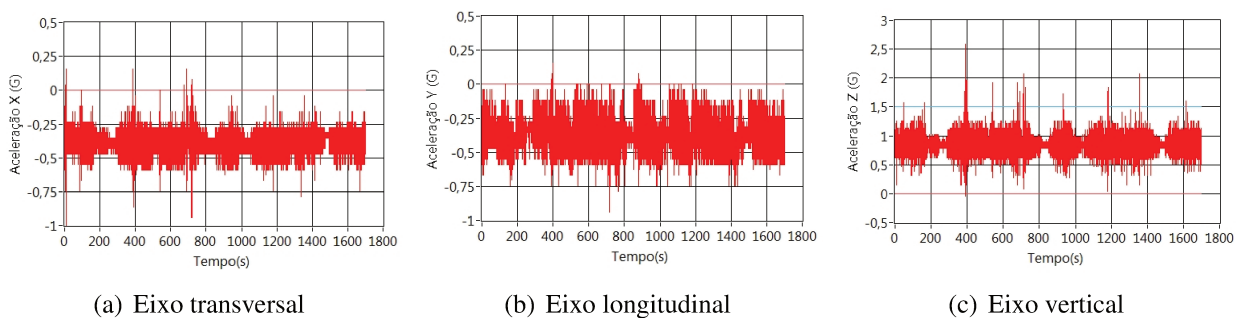


Figura 2.12: Aceleração no sistema BRT com acelerômetro na faixa de 4G.

#### 2.1.1.4 Análise dos testes iniciais

Das análises feitas para carros particulares e ônibus BRT, obtiveram-se as amplitudes descritas na Tabela 2.1. Face às amplitudes da aceleração estarem próximas a 2G, especialmente para o eixo vertical, se recomenda usar o acelerômetro na faixa de 4G e incrementar a resolução do conversor analógico-digital (ADC) a 10 bits no mínimo. Além disso, deve-se executar uma rotina de calibração simples para o acelerômetro (0G em X, 0G em Y, +1G em Z) em uma única posição [27].

Tabela 2.1: Aceleração pico a pico medido em carros e ônibus do sistema BRT.

	Longitudinal	Transversal	Vertical
Carro 1(G)	1,25	1,25	2
Carro 2(G)	0,8	0,7	1,9
Ônibus-BRT(G)	1,4	1,6	2,1

Observou-se que a frequência do sinal de aceleração dificilmente ultrapassa 50Hz. Assim, é necessário ter uma largura de banda mínima do acelerômetro de 50Hz, para aquisição adequada da informação desejada. É recomendável ter uma  $F_S$  de 500Hz e também colocar um filtro *antialiasing* para aquisições analógicas.

#### 2.1.2 Descrição do Hardware, firmware e resultados do segundo protótipo

Com base nas informações obtidas do primeiro protótipo, neste segundo protótipo utilizou-se um acelerômetro digital, o qual fornece vantagens tais como: maior taxa de amostragem, ADC e dois filtros incorporados; sendo o primeiro filtro *antialiasing* e o segundo filtro para limitar a largura de banda dos dados e reduzir as interferências devidas a acoplamentos no sinal de aceleração. Também, programou-se um hardware baseado na placa de desenvolvimento do microcontrolador LPC2478, a qual é usada para coletar os dados do acelerômetro digital (SMB380) e armazená-los em um cartão de memória SD. A placa de desenvolvimento é mostrada na Figura 2.13. A frequência de amostragem usada pelo ADC do acelerômetro é de 3kHz, assim o filtro *antialiasing* tem  $F_C$  de 1.5kHz. O acelerômetro tem um segundo filtro que provê uma limitação de banda programável, a qual foi ajustada em 50Hz. Assim, os dados são coletados com uma frequência de amostragem de 500Hz e são armazenados na memória SD através da interface do cartão de memória MCI (*Memory Card Interface*) do microcontrolador, a qual será descrita com maior detalhe no capítulo seguinte. Finalmente, para evitar a perda de dados, um buffer maior que 512 posições fixas foi implementado no microcontrolador, da mesma maneira que foi feito no primeiro protótipo.

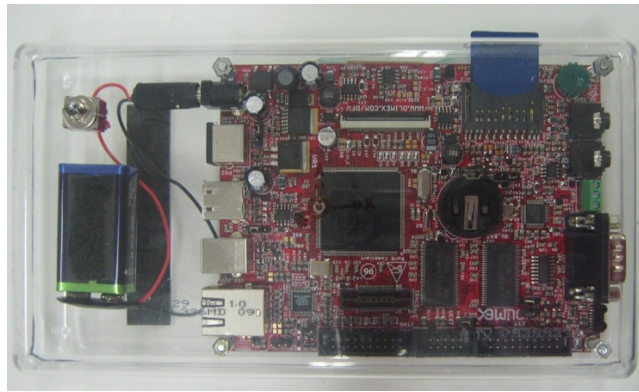


Figura 2.13: Segundo protótipo baseado na placa de desenvolvimento do LPC2478.

### 2.1.2.1 Resultado experimental obtido em ônibus circular de linha típica

Alguns testes foram realizados no ônibus circular da UNICAMP usando o protótipo baseado na placa de desenvolvimento do microcontrolador LPC2478. O sistema foi instalado no assoalho do ônibus, acima da roda direita, com os eixos cartesianos apontados na mesma direção que nos testes feitos no sistema BRT (Figura 2.11).

**Análise no tempo** Para um percurso de 21 minutos, as amplitudes das acelerações pico a pico medidas pelo sistema foram de 1,6G; 0,912G e 2,85G para as acelerações transversais (X), longitudinais (Y) e verticais (Z), respectivamente, como mostrado na Figura 2.14.

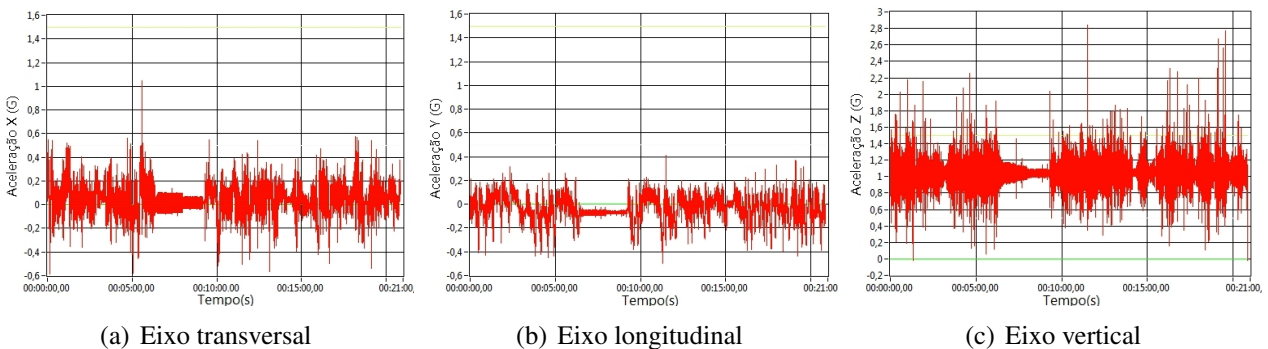


Figura 2.14: Aceleração no ônibus circular da UNICAMP ao longo do tempo.

Durante o percurso os eventos que geraram desconforto foram informados em uma folha de registro e associados ao tempo da ocorrência, à fonte do evento e à intensidade (baixa, media, alta) com a qual foi percebido. O evento mais significativo foi causado por uma lombada reportada pelo passageiro no minuto 11min34s e reportada pelo sistema no minuto 11min30s, mostrada na Figura 2.15.

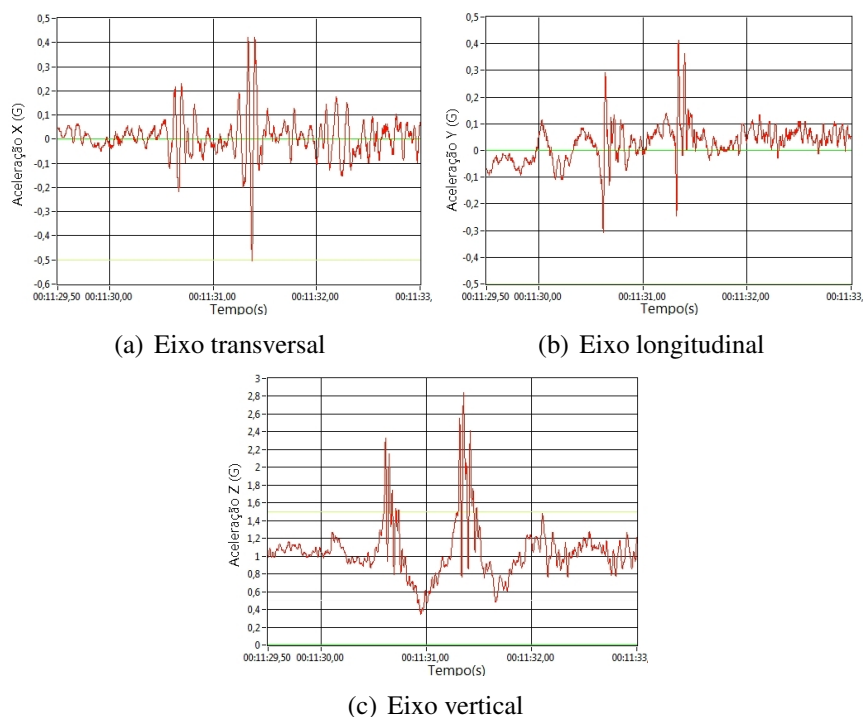


Figura 2.15: Aceleração do maior evento no ônibus circular da UNICAMP.

A amplitude pico a pico desse evento (no eixo Z) foi de 2,4G e teve uma duração de 2 segundos aproximadamente. Um aspecto interessante do sinal é o efeito da lombada sobre os eixos do plano horizontal, pois o componente de maior frequência foi transmitido para esses eixos (X e Y), sem superar a amplitude pico a pico de 1G.

A informação obtida dos testes foi analisada no domínio da frequência usando duas transformadas, a primeira: a Transformada Rápida de Fourier, FFT, e a segunda: a Transformada Wavelet.

**Análise usando Transformada Rápida de Fourier, FFT** A Transformada Discreta de Fourier (DFT) é uma ferramenta usada para o análise espectral, a qual transforma o sinal de interesse do domínio do tempo para o domínio da frequência. A FFT é um algoritmo baseado na decomposição do cálculo da DFT em pequenas partes, com o objetivo de diminuir o tempo de processamento [28]. Os resultados da FFT dos sinais de aceleração nos três eixos, utilizando o software LabView™, são mostrados na Figura 2.16. Observou-se que as frequências dos sinais de aceleração estão altamente concentrados nos primeiros 50Hz. Também mostram características espectrais similares nos três eixos, pois possuem três faixas de frequência, a primeira de 0Hz a 5Hz, a segunda de 6Hz a 25Hz e a terceira de 26Hz a 40Hz. Da análise feita no domínio do tempo se espera que a primeira faixa contenha a informação de distúrbios causados por lombadas ou causados pela maneira de dirigir do motorista, por exemplo: mudanças de direção bruscas. Os resultados obtidos nesta análise estão de



acordo com os resultados da pesquisa realizada pelo Departamento de transporte dos Estados Unidos [10]. Comparando as faixas de frequências obtidas nesta análise com os resultados obtidos por Thomas [10], apresentados na Tabela 1.2, observa-se que as frequências possuem valores similares, especialmente para o ônibus B do estudo de Thomas. No entanto, determinar a natureza do componente em frequência usando somente a análise de Fourier é complexo, pois esta somente analisa o sinal no domínio da frequência sem relacionar com o domínio do tempo; o qual é adequado para sinais estacionários, mas para sinais aleatórios se necessita utilizar outra técnica.

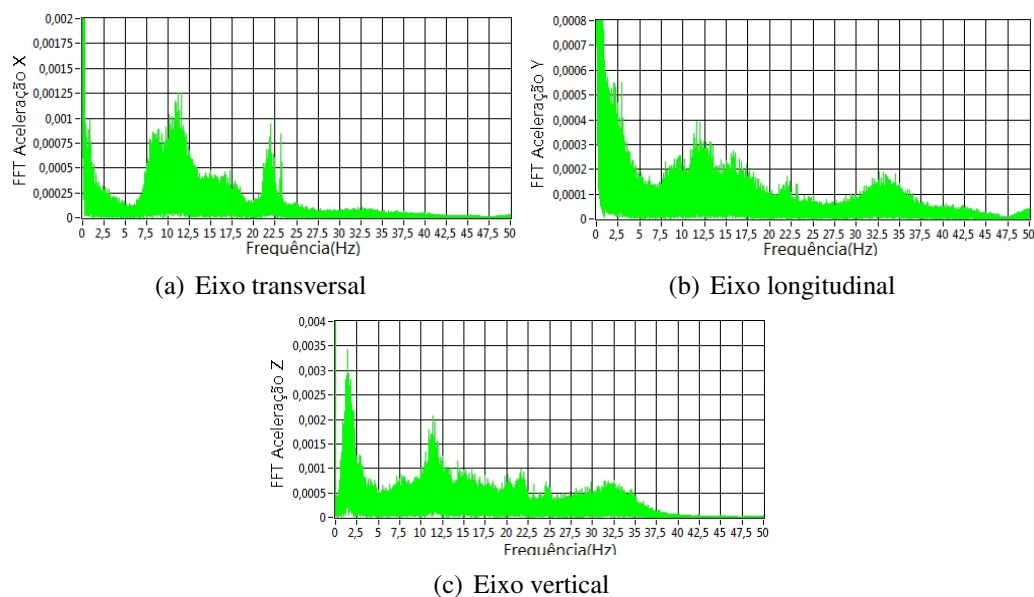


Figura 2.16: Análise espectral usando FFT da aceleração no ônibus circular da UNICAMP.

**Análise usando Transformada Wavelet discreta** A Transformada Wavelet surgiu para estudar os sinais não estacionários, ou seja, com mudanças repentinas e eventos aleatórios, como é o caso do sinal de aceleração do ônibus. Enquanto que a DFT detecta os componentes de frequência que se repetem sem importar a ordem que estes são gerados; a Wavelet fornece informação adicional do componente de frequência, tais como: quando ocorreu e qual é sua duração. Assim, a Wavelet é uma análise tempo-frequência. Portanto a transformada Wavelet pode ser vista como uma análise espectral em uma janela de tempo móvel. Na análise de Fourier os sinais são decompostos em sinais senoidais, enquanto que a Wavelet é decomposta por versões escalonadas e móveis de um sinal chamado Wavelet mãe. O algoritmo da Transformada Discreta Wavelet (DWT), descrito na Figura 2.17, está baseado na utilização de filtros para obter o conteúdo em baixas frequência (aproximação) e de altas frequência (detalhes), além disso, mediante uma subamostragem (*downsampling*) se eliminam os dados duplicados; este processo se repete até atingir o nível de decomposição desejado [29]. Assim, o nível de decomposição indica a resolução de frequência.

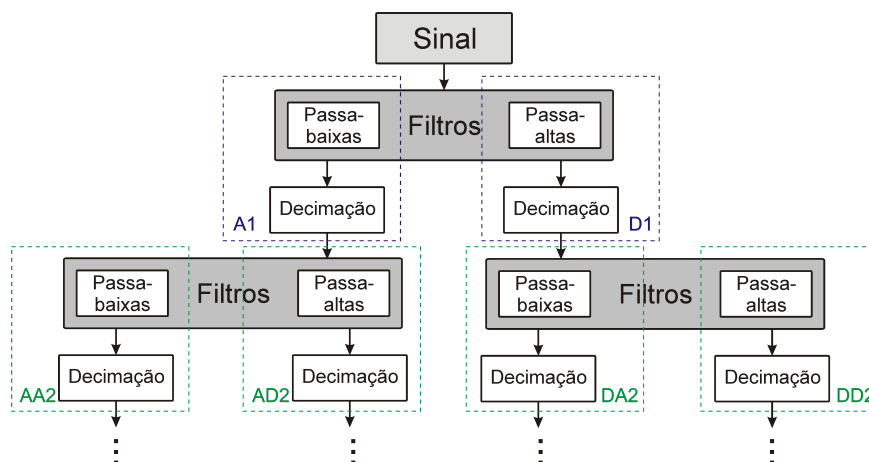


Figura 2.17: Algoritmo para obter a DWT. A-Aproximação e D-Detalhes.

Utilizando ferramentas de análise DWT do software LabView™, o sinal de aceleração do ônibus no eixo Z foi analisado para uma mesma duração de 21 minutos. O sinal e o resultado da DWT são mostrados na Figura 2.18. O resultado da DWT é um gráfico tridimensional sendo que o eixo vertical representa as componentes em frequência para uma janela de tempo de 2 segundos, enquanto que o eixo horizontal indica o tempo e a escala de cores indica a intensidade do componente em frequência (vermelho indica maior intensidade e azul menor intensidade).

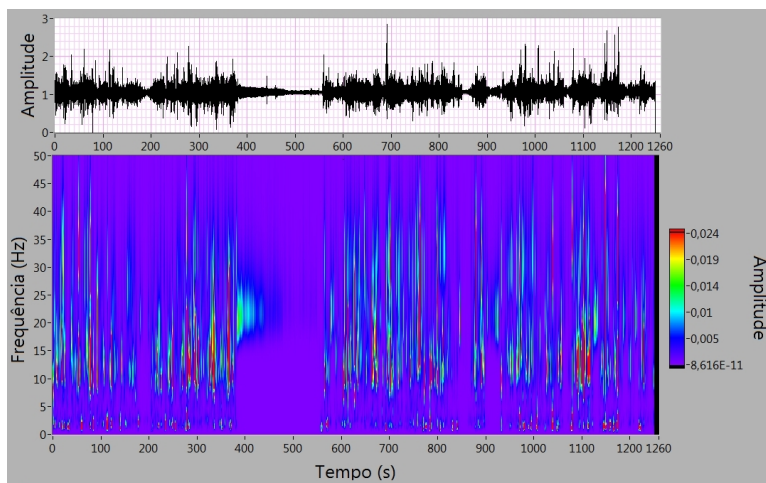


Figura 2.18: Análise espectral usando DWT da aceleração no ônibus circular da UNICAMP, com resolução de 0,33Hz e janelas de 1s. Eixo Z.

O sinal entre 400 e 550 segundos se relaciona ao ônibus parado, mas com o motor ligado; as frequências nesta janela de tempo, ou seja, entre 15Hz e 30Hz, representam à componente devido à vibração do motor interagindo com a estrutura do ônibus, este sinal é mostrado em detalhes na

Figura 2.19.

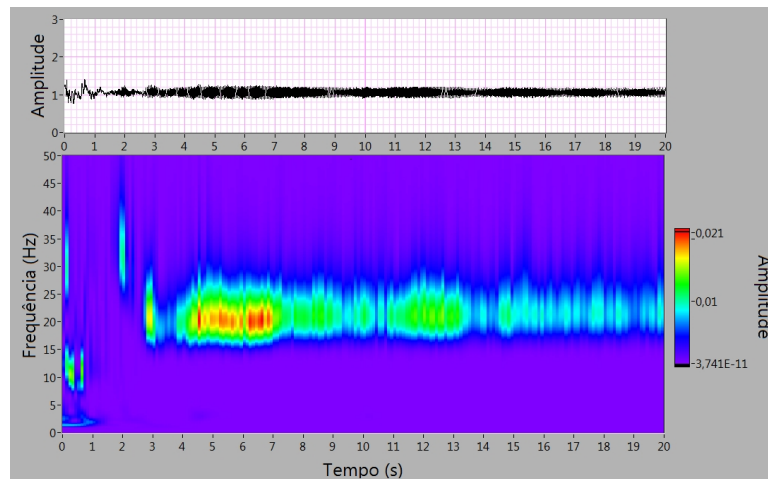


Figura 2.19: Análise espectral usando DWT da vibração produzida pelo motor, com resolução de 0,33Hz e janelas de 0,1s. Eixo Z.

O sinal causado por uma lombada está caracterizado por duas faixas de frequências. A primeira é um sinal de baixa frequência, abaixo de 5Hz, causada pela frequência natural do sistema de amortecimento e pela geometria da lombada. A segunda é um sinal entre 15Hz e 40Hz que pertence à interação da lombada com a estrutura do ônibus, pois é similar à informação obtida com o ônibus imóvel. Um exemplo deste sinal é mostrado em detalhe na Figura 2.20, sendo que o fragmento analisado pertence à faixa entre 680s a 700s.

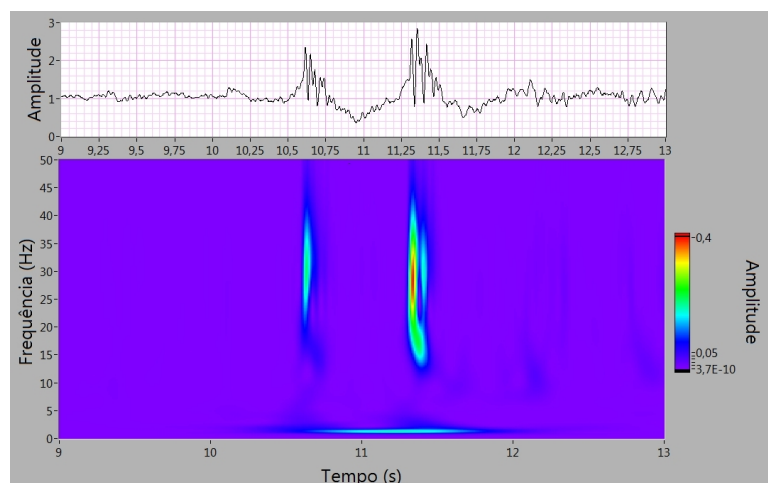


Figura 2.20: Análise espectral usando DWT da aceleração vertical produzida por uma lombada, com resolução de 0,33Hz e janelas de 0,01s. Eixo Z.

Também foram analisadas as características do sinal de aceleração no sentido transversal e longitudinal para a janela de tempo na qual se reinicia o movimento. A Figura 2.21 mostra a aceleração

transversal (eixo X), na qual é refletido o movimento lateral do veículo sendo retirado do acostamento, ou seja girando o volante a esquerda para mudar de pista. Nesta análise se usou uma transformada com 1500 níveis, ou seja, resolução de 0,33Hz e uma resolução no tempo de 0,01s. No detalhe da Figura 2.21 é indicado, mediante uma caixa, a porção do espectro. O espectro do movimento lateral (eixo X) do ônibus possui componentes abaixo de 0,5Hz e frequências entre 7Hz e 20Hz. Da mesma forma, na Figura 2.22 é mostrada a aceleração longitudinal (eixo Y) e seu espectro, no qual se identificaram duas faixas importantes: entre 7Hz e 20Hz e abaixo de 5 Hz. Na faixa abaixo de 5Hz se identificaram três áreas às quais são indicadas por círculos. A primeira pertence ao arranque inicial (saída do acostamento) enquanto que o segundo e terceiro pertencem ao arranque depois de retomar a pista e à troca de marcha.

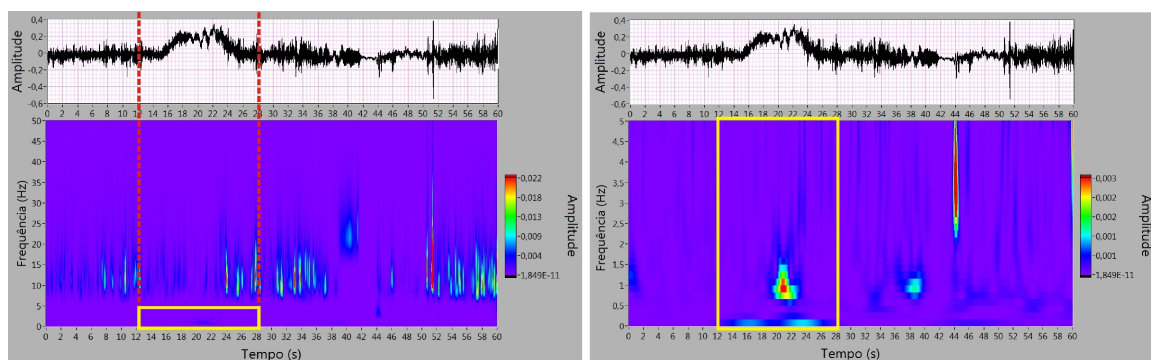


Figura 2.21: Análise espectral usando DWT da aceleração transversal, com resolução de 0,33Hz e janela de 0,01s(Esquerda). Amplificação de 0Hz a 5Hz(Direita). Eixo X.

Assim, é possível associar o componente de frequência dos sinais de aceleração a sua causa; sendo que frequências abaixo de 5Hz são causadas por mudanças de direção, arranques, freadas ou lombadas. Enquanto que frequências na faixa de 15 e 30Hz são causadas por vibrações do motor e pela estrutura do veículo. Esta faixa de frequências de vibração é dependente do modelo do veículo e

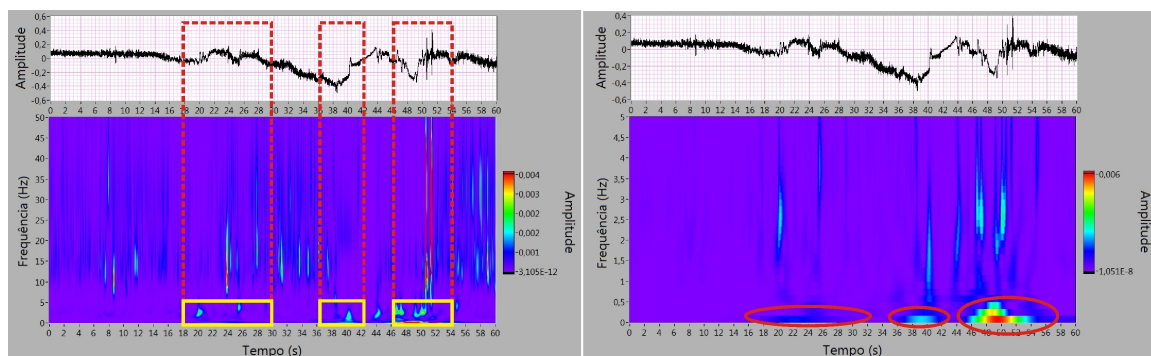


Figura 2.22: Análise espectral usando DWT da aceleração longitudinal, com resolução de 0,33Hz e janela de 0,1s(Esquerda). Amplificação de 0Hz a 5Hz(Direita). Eixo Y.

pode variar com seu estado de manutenção ou o tipo de sistema de amortecimento [10].

## 2.2 Características do sinal de *Jerk* em veículos

O *Jerk* é definido como a variação da aceleração no tempo. Considerando a aceleração modelada como um sinal senoidal; temos:

$$a_{(t)} = A \sin(\omega t) \quad (2.1)$$

sendo  $A$  a amplitude da aceleração e  $\omega$  a frequência angular do sinal. Assim, o *Jerk* será modelado como a derivada de  $a_{(t)}$  e a amplitude do *Jerk* varia em função da frequência, como mostrado a seguir:

$$\frac{d(a_{(t)})}{dt} = A\omega \cos(\omega t) \quad (2.2)$$

Esta dependência tem uma grande vantagem para análise de conforto, pois mudanças rápidas da aceleração também são causas de desconforto como sugerido por Hoberock [19]. A vantagem de utilizar o *Jerk* na avaliação de conforto é mostrada na Figura 2.23, na qual os picos do *Jerk* (sem filtrar) são indicados por caixas. A caixa A indica um evento de aceleração rápida, mas de pouca amplitude, enquanto a caixa B indica um evento de aceleração com grande amplitude; assim, embora qualquer pico de aceleração acima de um limiar seja detectado, se o sinal tem uma variação rápida e não possui amplitude suficiente para atingir o limiar, este evento de desconforto será desconsiderado erroneamente. Usando o *Jerk* no qual as variações da aceleração são refletidas em sua amplitude é possível detectar eventos de desconforto com estas características. Para determinar os limiares do *Jerk*, usando

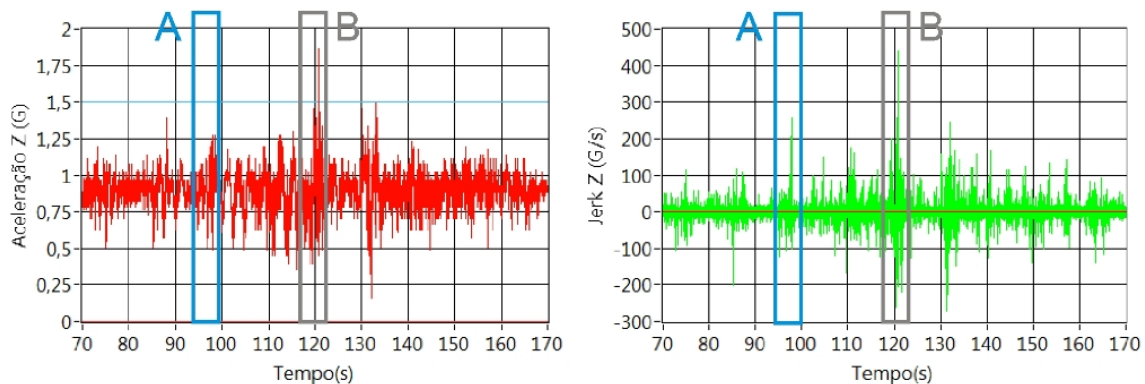


Figura 2.23: Aceleração e *Jerk* no sistema BRT na faixa de 2G.

a aceleração como fonte de informação, é necessário aplicar um filtro passa-baixas ao sinal de aceleração evitando ruído no sinal de *Jerk*. Andersson e Nilstam [5] utilizaram um filtro passa-baixas

com frequência de corte de 0,3Hz, e obtiveram resultados para a probabilidade de percepção de um desconforto em função do *Jerk* lateral. Na Figura 2.24 são mostrados os resultados da pesquisa direta com usuários em trens. Nesta dissertação será usado um filtro com a mesma frequência de corte, 0,3Hz aplicado ao cálculo do *Jerk* longitudinal, devido às características do serviço de ônibus que possuem uma alta taxa de paradas e arrancadas. Assim, o movimento longitudinal se torna um fator que pode atingir baixas amplitudes de aceleração, mas com altas taxas de transição.

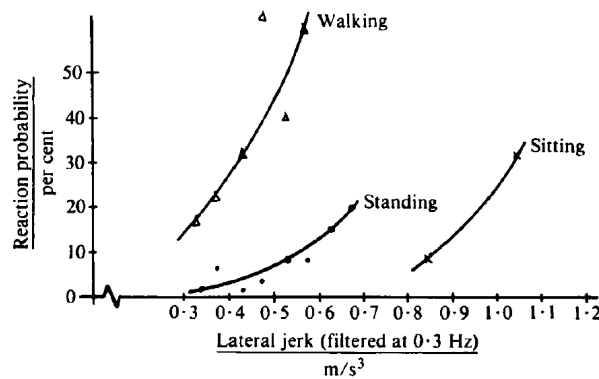


Figura 2.24: Desconforto dos usuários de trens em função do *Jerk* lateral. [5]

### 2.2.1 Filtro passa-baixas para o cálculo do *Jerk*

O filtro passa-baixas deve ser de banda estreita, pois o sinal de aceleração é amostrado a 500Hz e a frequência de corte desejada é de 0,3Hz, assim o filtro precisa ter uma ordem alta para atingir com esta frequência. No entanto, um filtro digital de ordem alta tem desvantagens tais como: alta exigência de processamento, elevado tempo de análise, grande número de amostras perdidas no começo e no final do sinal devido ao período de estabilização do filtro. O filtro projetado visa contornar estas desvantagens e utiliza duas decimações sucessivas para diminuir a  $F_S$  do sinal de aceleração até 10Hz. Para a adequada decimação do sinal deve-se utilizar um filtro *antialiasing* para atender o teorema de Nyquist. O diagrama de blocos do filtro projetado é mostrado na Figura 2.25. A primeira decimação com um fator de 5 diminui a  $F_S$  de 500Hz para 100Hz, lembrando que o BW do acelerômetro foi limitado a 50Hz, o qual é usado como filtro *antialiasing*. Posteriormente, um filtro passa-baixas FIR de ordem 32 e  $F_C$  de 1Hz é aplicado antes da decimação seguinte para garantir o teorema de Nyquist. A decimação seguinte é feita por um fator de 10 o qual diminui a  $F_S$  para 10Hz, e finalmente se aplica o filtro passa-baixa FIR de 0,3Hz de ordem 16. Os coeficientes foram calculados usando o software LabView™. O diagrama de blocos do programa usado para obter estes coeficientes é mostrado no apêndice B. O fator de correção para obter um ganho unitário na banda de passagem deve ser de:



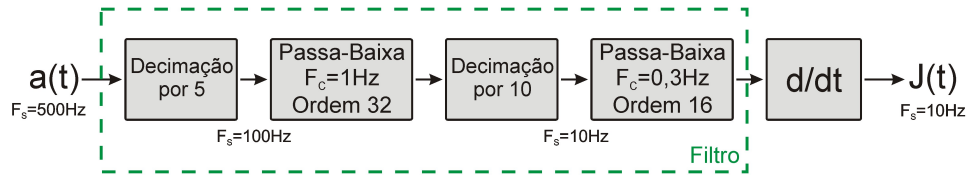


Figura 2.25: Diagrama de blocos do algoritmo para calcular o *Jerk* destacando o filtro implementado.

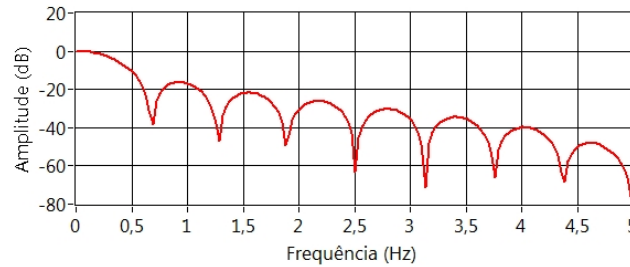


Figura 2.26: Resposta ao impulso do filtro passa-baixa com  $F_C$  de 0,3Hz e  $F_S$  a 10Hz.

0,748249, pois os coeficientes do filtro FIR não levam em conta esta correção. Na Figura 2.26 é mostrada a resposta ao impulso do filtro descrito anteriormente.

A quantidade de operações aritméticas necessárias para a implementação deste filtro foi calculada; pois o número de operações de soma e multiplicação da convolução no tempo é dado por  $N * M$ , sendo  $N$  a quantidade de amostras do sinal e  $M$  a quantidade de coeficientes do filtro, assim para este filtro a quantidade de operações aritméticas é 100,8 mil; durante a análise de um sinal de 30 segundos. Enquanto, para a filtragem direta de ordem 16, 32 e 64 são necessárias 240 mil, 480 mil e 960 mil operações respectivamente. Os resultados obtidos mostraram que o filtro projetado utiliza 58%, 79% e 90% menos operações aritméticas comparada com as operações necessárias para uma filtragem direta de ordem 16, 32 e 64, respectivamente.

Além disso, se utilizássemos a convolução na frequência o número de operações seria calculado por  $N + 2N \log_2 N$  devido ao cálculo da FFT, multiplicação ponto a ponto na frequência e à transformada inversa de Fourier (IFT). Assim, para a filtragem depois de aplicada uma decimação com um fator de 5 são necessárias 102 mil operações.

### 2.2.2 Algoritmo para calcular o *Jerk*

Depois de obter o sinal filtrado, o passo seguinte é derivar o sinal. Para calcular a derivada se utilizou o método de diferenças finitas, o qual é a diferença das amostras dividida pelo tempo entre elas, ou seja, o inverso da frequência de amostragem (0,1s). Calculou-se o *Jerk* do sinal obtido da

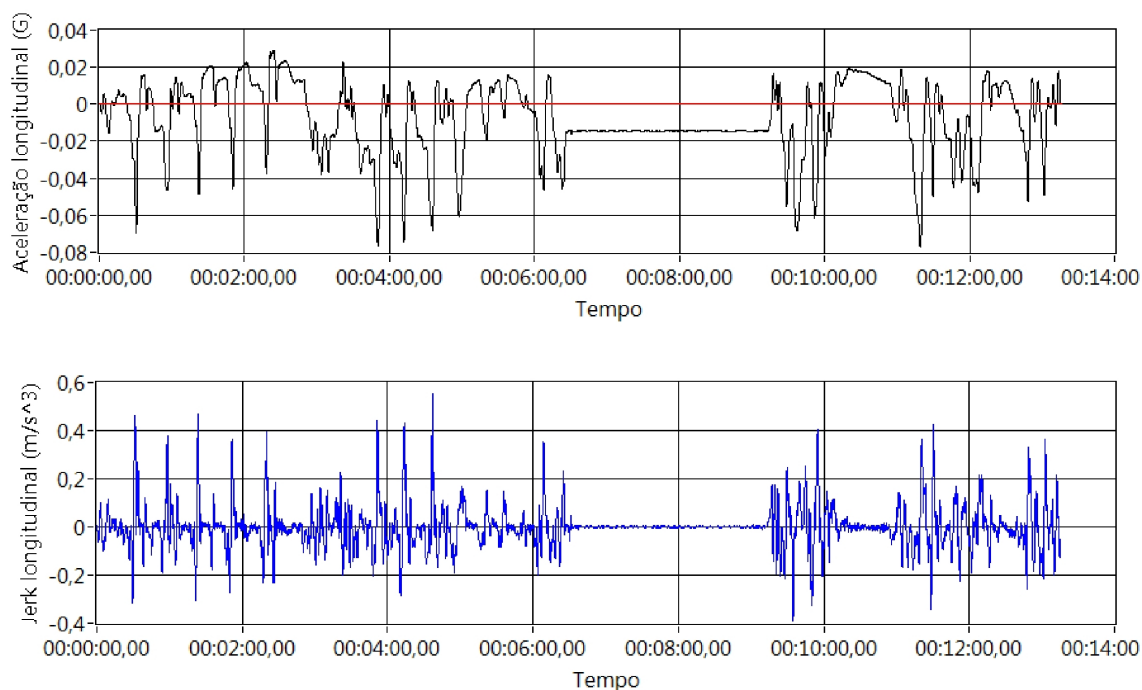


Figura 2.27: Aceleração longitudinal e *Jerk* longitudinal do ônibus circular da UNICAMP usando o algoritmo proposto.

aceleração longitudinal do ônibus circular da UNICAMP, os resultados obtidos são mostrados na Figura 2.27.

O sinal da aceleração longitudinal mostra as paradas que o ônibus efetuou durante o percurso enquanto o *Jerk* longitudinal mostra, mediante picos, as consequências das mudanças rápidas de aceleração. Os valores calculados pelo algoritmo implementado estão na mesma faixa que os reportados por Andersson nos testes de desconforto em função do *Jerk* transversal em trens [5]; assim, limiares similares podem ser adotados no sistema de transporte de ônibus. Nesta dissertação, baseados nas pesquisas de Andersson e Nilstam [5], nas recomendações de Förstberg [2] e nos dados obtidos mostrados na Figura 2.27, o limiar do *Jerk* será fixado em  $0,4\text{m/s}^3$ .



# Capítulo 3

## Hardware do Sistema

Neste capítulo são detalhados os componentes de hardware necessários para implementar o protótipo final, também como alguns critérios utilizados na sua escolha. Os principais componentes são microcontrolador, módulo de comunicação sem fio, memória, regulador de tensão e sensores tais como: GPS, acelerômetro e sensor de temperatura. Finalmente, o diagrama de blocos, circuito esquemático, protótipo final e uma estimativa do custo do SEMC-TP são apresentados.

### 3.1 Componentes de Hardware

Esta dissertação tem como objetivo o desenvolvimento de um instrumento capaz de avaliar o conforto no transporte público através da aquisição e processamento da aceleração, posição geográfica e temperatura. Também se devem fornecer meios de comunicação entre os nós do sistema e armazenamento de dados de fácil acesso. Assim, para o projeto do sistema, devem ser considerados os seguintes elementos básicos para a implementação do hardware do Sistema de Monitoramento do Conforto (SEMC):

- Sensores:
  - Acelerômetro e
  - Sensor de temperatura.
- Módulo de posicionamento global (GPS),
- Módulo de comunicação,
- Memória SD,
- Processador ou microcontrolador e

- Regulador de tensão.

### 3.1.1 Sensores

Qualquer dispositivo que converta um tipo de sinal de um domínio a outro é denominado transdutor. Existem seis tipos de domínios: mecânicos, térmicos, magnéticos, elétricos, ópticos e químicos; Usualmente e em especial em engenharia elétrica, o termo sensor é utilizado para designar um transdutor que transforma um sinal físico em um sinal elétrico fornecendo uma saída que é função da grandeza medida.

#### 3.1.1.1 Acelerômetro

Um acelerômetro é um sensor capaz de medir a aceleração linear, a qual é medida tendo como referência a aceleração da gravidade,  $G=9.8\text{m/s}^2$ . Existem diferentes tipos de acelerômetros com diversos princípios de funcionamento: Capacitivo, Piezoelétrico, Piezoresistivo, Efeito Hall, Magnetoresistivo, entre outros [30]. Os acelerômetros microeletrônicos estão baseados em tecnologia MEMS (*Micro-Electro Mechanical System*) a qual utiliza estruturas com dimensões micrométricas para medir a aceleração, uma das estruturas mais usadas em acelerômetros MEMS é o *comb-driver*. O *comb-driver* é um conjunto de eletrodos interdigitados, como mostrado na Figura 3.1, o conjunto de eletrodos moveis são deslocado devido ao movimento feito pela massa central mudando a capacitância com relação aos eletrodos fixos. Algumas das aplicações típicas para acelerômetros são: sistemas de disparo de *Air-bags*, dispositivos de jogos, dispositivos de navegação, monitoramento de vibração, instrumentação médica, dentre outras.

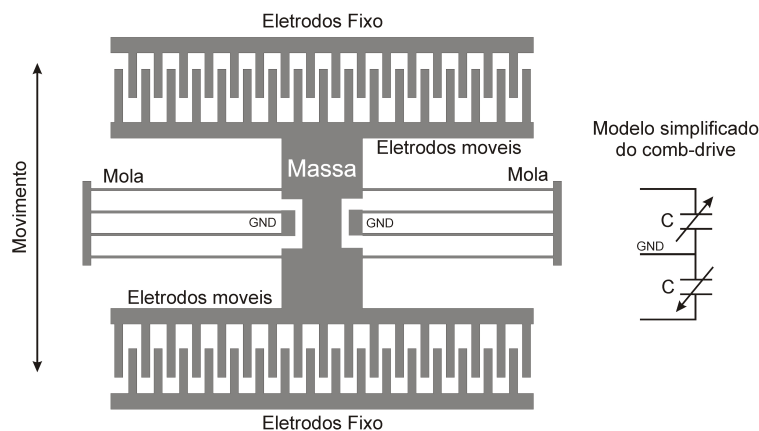


Figura 3.1: Modelo e estrutura simplificada do comb-drive.

Conforme o tipo de saída, os acelerômetros podem ser classificados em acelerômetros analógi-

cos ou digitais. Os acelerômetros analógicos apresentam a aceleração como uma escala de tensão proporcional à sua saída, sendo a resolução deste dada em V/G. Enquanto, os acelerômetros digitais representam a aceleração mediante uma codificação binária que pode ser transmitida a um microcontrolador usando protocolos de comunicação serial. Os protocolos de comunicação mais usados em sensores são: *Serial Peripheral Interface* SPI ou *Inter-Integrated Circuit* I<sup>2</sup>C, os quais são descritos no apêndice D.

Seguindo as orientações apresentadas no capítulo 2, um acelerômetro digital é o mais indicado para uso e implementação do hardware. Na Tabela 3.1 são comparados acelerômetros de vários fabricantes e com características adequadas para este tipo de aplicação.

Tabela 3.1: Comparação entre acelerômetros tri-axiais digitais

Referência	SCA3100-D07	SCA3000-E04	MMA7455L	ADXL345	SMB380
Fabricante	VTI	VTI	Freescall	Analog devices	Bosch
Range(G)±	6	6	2/4/8	2/4/8/16	2/4/8
Resolução	650 count/G	500 count/G	64/32/16 count/G	256/128/64/32 LSB/G	256/128/64 LSB/G
Comunicação	SPI	SPI	I <sup>2</sup> C-SPI	I <sup>2</sup> C-SPI	I <sup>2</sup> C-SPI
Sensor de temperatura	Sim	Não	Não	Naõ	Sim
BW (Hz)	55	9/38/70	125/250	1,6k-0,05	1.5k-25
Resolução do ADC (bits)	13	12	8 e 10	13	10
Preço (US\$)	61,24	29,73	1,94	6,93	27(soldado)
Preço de 100 unidades (US\$)	40,824	11,9	1,13	5,15	-

Utilizando como critério a resolução, range e BW foi escolhido o acelerômetro da Analog Devices, ADXL345, pois embora o preço seja alto em comparação ao concorrente da Freescale, este possui maior resolução de ADC e maior versatilidade no BW. Assim, o ADXL345 permite utilizar o BW limitado a 50Hz como discutido no capítulo anterior. O ADXL345 é feito em tecnologia de micromáquinas de superfície utilizando como princípio da variação de capacitância usando uma *comb-drive*. Um demodulador de fase é utilizado para determinar a amplitude e sentido da aceleração [6]. O diagrama de blocos simplificado deste acelerômetro é mostrado na Figura 3.2.

O acelerômetro ADXL345 pode utilizar tanto comunicação SPI como I<sup>2</sup>C, nos dois casos este se comporta como um dispositivo escravo. Embora o protocolo SPI forneça vantagens sobre o I<sup>2</sup>C como maior taxa de transmissão e menor consumo de energia; o protocolo I<sup>2</sup>C precisa de somente duas linhas de comunicação e tem maior versatilidade para implementar um sistema com mais de um dispositivo escravo do que o SPI. Na Figura 3.3 é mostrada a conexão do acelerômetro configurado



Tabela 3.2: Taxa de dados, largura de banda e código do registro BW\_RATE disponíveis para o ADXL345 [6]

Taxa de saída	BW(Hz)	Código para o registro	Taxa de saída	BW(Hz)	Código para o registro
3200	1600	1111	12,5	1600	0111
1600	1600	1110	6,25	1600	0110
800	1600	1101	3,13	1600	0101
400	1600	1100	1,56	1600	0100
200	1600	1011	0,78	1600	0011
100	1600	1010	0,39	1600	0010
50	1600	1001	0,20	1600	0001
25	1600	1000	0,10	1600	0000

que o acelerômetro é vigiado no modo *sleep* em procura de atividade (o acelerômetro reduz a frequência de amostragem a 8Hz, 4Hz, 2Hz ou 1Hz).

- Registro de formato de dados (DATA\_FORMAT, 0x31): Este registro controla a apresentação dos dados dos registros de 0x32 a 0x37. As funções deste registro são: configuração do range do acelerômetro, justificação dos bits nos registro de saída, configuração da resolução dentre outras.
- Registros de dados de aceleração (DATAx0 e DATAx1, 0x32 - 0x37): Estes registros contêm a informação da aceleração nos três eixos e se atualiza segundo a frequência fixada em BW\_RATE. O registro 0 contém a parte menos significativa (LSB) enquanto o registro 1 contém a parte mais significativa (MSB) da informação.

### 3.1.1.2 Sensor de temperatura

O sensor de temperatura tem como função monitorar a temperatura dentro da cabine dos passageiros. Segundo Xavier a faixa de conforto da temperatura para o Brasil está entre 20,5°C e 24,5°C [31]. O sensor utilizado neste trabalho foi o TMP100 da *Texas Instruments*, o qual possui interface I<sup>2</sup>C, resolução de 0,0625°C e faixa de operação de -55°C a 125°C [7]. A Figura 3.4 mostra o diagrama de blocos do sensor de temperatura.

A Figura 3.5 mostra o diagrama esquemático do circuito utilizado para conectar o TMP100 ao mesmo barramento de comunicações I<sup>2</sup>C do acelerômetro. O endereço deste dispositivo foi configurado, através dos pinos ADD0 e ADD1, para 0x48 e foram usados os mesmo valores para os resistores de *pull-up* utilizados no acelerômetro.

O primeiro registro(0x00) contém o dado de temperatura, enquanto o segundo(0x01) permitem gerenciar os modos de economia de energia, configurar a resolução entre 9,10,11 e 12 bits e iniciar

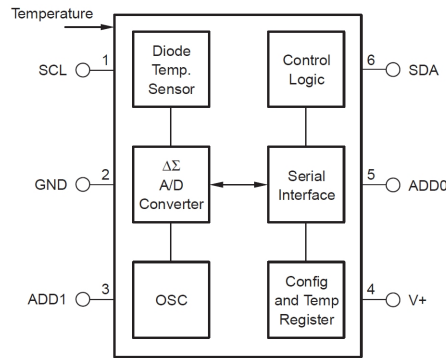


Figura 3.4: Diagrama de blocos do sensor de temperatura TMP100 [7].

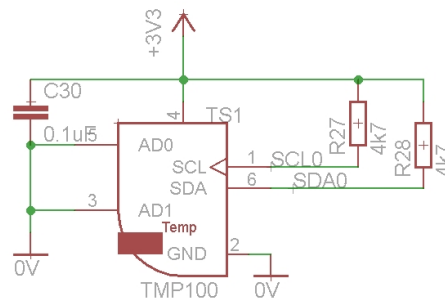


Figura 3.5: Diagrama esquemático do circuito de polarização do sensor de temperatura TMP100 [7].

a medição da temperatura. A descrição completa destes registros pode ser consultada na folha de especificação do TMP100 [7].

### 3.1.2 Módulo de posicionamento global (GPS)

O Sistema de Posicionamento Global (GPS) é um sistema de navegação baseado em um conjunto de satélites, sendo quatro um número suficiente para que o sistema funcione. Este sistema fornece informação de posição e tempo de maneira contínua em qualquer local do planeta. Além disso, como o sistema é passivo, o número de usuários é ilimitado [32]. Entre os usos do GPS estão: navegação terrestre e aérea, controle ou guia de maquinário pesado, monitoramento de espécies animais, projeto, construção e monitoramento de estruturas tais como edifícios ou pontes, gestão de sistemas de transporte, entre outros.

Com as informações da posição de quatro satélites e a distância entre o receptor GPS e estes satélites é possível conhecer a localização do receptor GPS, pois esta pode ser calculada usando os *pseudoranges* descritos por Langley [33]. A distância de cada satélite é conhecida através da medida do tempo necessário para a propagação do sinal desde o satélite até o receptor GPS, assim multiplicando este tempo pela velocidade da luz se obtém a distância. Além da posição, também pode

ser conhecida a velocidade do receptor GPS efetuando uma análise Doppler do sinal do satélite.

Existem dois tipos de serviço GPS, *Precise Positioning Service* (PPS) e *Standard Positioning Service* (SPS). O serviço PPS utiliza um código conhecido como  $P(Y)$  o qual está disponível unicamente para agências governamentais e aplicações militares dos Estados Unidos, este serviço tem demonstrado possuir exatidões para posição horizontal e tempo de 8m e 10ns, respectivamente. O serviço SPS usa o código C/A, o qual é de acesso livre para todos os usuários. O serviço SPS é menos preciso do que o PPS e fornece uma exatidão de 13m e 40ns, para a posição horizontal e o tempo, respectivamente [32].

Para compartilhar informação entre dispositivos de distintos fabricantes de aparelhos receptores ou mesmo interfaces gráficas, alguns protocolos para a representação dos dados do GPS foram desenvolvidos, tais como: RINEX, SP3, RTCM SC-104 e NMEA 0183. Neste trabalho será utilizado o protocolo NMEA para a leitura dos dados, o qual transmite dados de posição, tempo, data, velocidade, altura, estado dos satélites, entre outras informações em formato ASCII. O NMEA é descrito em detalhes no apêndice C. O GPS assistido ou A-GPS utiliza uma segunda fonte de informação (internet, rede de telefonia celular ou memórias) para prover informações sobre a última posição do receptor ou a posição aproximada deste; desta maneira o tempo para adquirir a primeira posição é reduzido.

O módulo de GPS utilizado neste trabalho é o ME-1000RW da *ME Components*, [11] mostrado na Figura 3.6. Este módulo é utilizado em navegação terrestre e marítima, localização de automóveis, de pessoas e aplicações esportivas e de lazer. O módulo tem a capacidade de receber sinais de 65 satélites diferentes e transferir a informação da posição e tempo pela porta serial (UART) utilizando o padrão NMEA. Também possui uma antena *patch* inclusa e um relógio em tempo real que permite reduzir o tempo necessário para obter a primeira leitura. Na tabela 3.3 é fornecida uma breve descrição técnica do módulo .

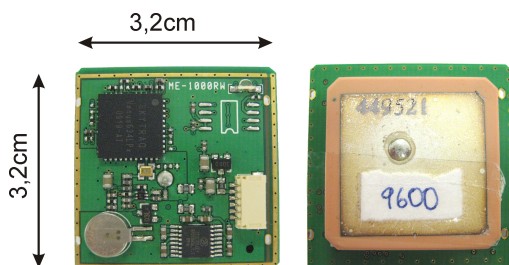


Figura 3.6: Módulo GPS. ME-1000RW.

O módulo GPS está baseado no receptor da SKYTRAQ, VENUS634LPX, o qual inclui módulo de processamento de RF e banda base, regulador de tensão, oscilador dentre outros componentes necessários para a implementação do módulo receptor. Também o módulo GPS possui um filtro e

Tabela 3.3: Especificações do módulo GPS ME-1000RW [11]

Parâmetro	Especificação
Tensão de alimentação	3.3V - 6V
Tipo de receptor	Código C/A de 65 canais na banda de frequência L1
Sensibilidade	-161dBm
Taxa de atualização	1Hz
Exatidão	5m CEP, 30ns e 0,1m/s
Tempo para primeira aquisição	32s-10s
Interface de comunicação	Serial Full Duplex(NMEA) 8 bits, 1 bit parada sem paridade LVTTL e RS232
Velocidade de comunicação	4800/9600/38400/115200bps

um amplificador RF de baixo ruído para melhorar sua relação sinal/ruído. O diagrama de blocos simplificado e a descrição dos pinos deste módulo é mostrada na Figura 3.7.

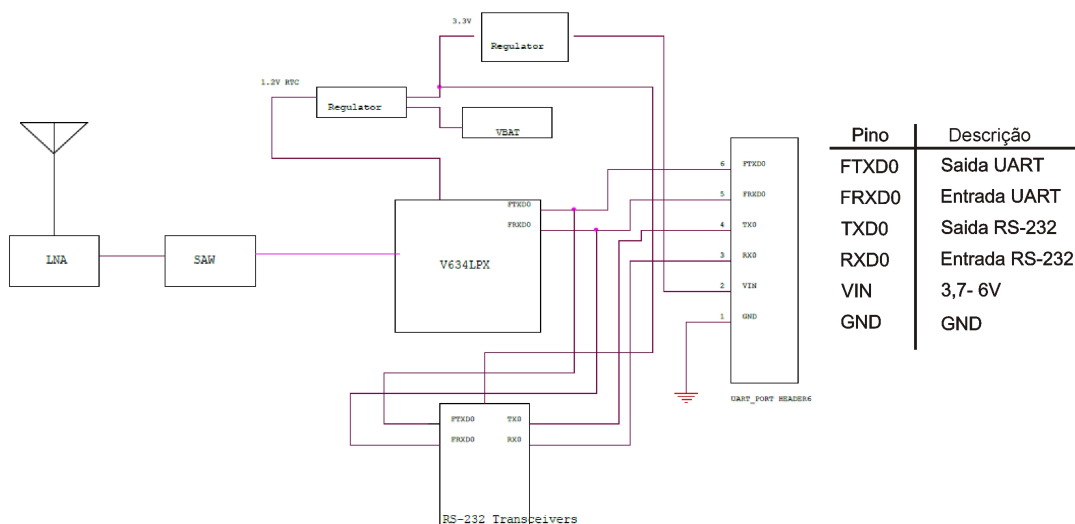


Figura 3.7: Diagrama de blocos do ME-1000RW e informação dos pinos.

### 3.1.3 Módulo de comunicação

Para o dimensionamento do módulo de comunicação sem fio se deve levar em conta fatores tais como: necessidades do ambiente de trabalho, características dos diferentes protocolos e custo das soluções. A estação de ônibus é o ambiente de trabalho onde o módulo de comunicações tem que trabalhar; pois, os dados que provêm do sistema tais como aceleração, temperatura e posição serão



transferidos até um computador localizado nessa estação. Fatores críticos como o número de nós que se conectarão ao sistema e a distância entre o ponto de aquisição e o local de embarque/desembarque dos passageiros são analisados a seguir para o adequado dimensionamento deste módulo.

### 3.1.3.1 Dimensionamento do módulo de comunicação

**Distância:** O Terminal Rodoviário Tietê, na cidade de São Paulo, é considerado uma das maiores estações de ônibus do Brasil [34]. Assim, o range mínimo necessário para garantir a cobertura do terminal é determinado através da análise das dimensões da estrutura utilizando uma imagem de satélite do Googlemaps™ e presumindo que a antena receptora será posicionada no centro do terminal. Como mostrado na Figura 3.8 o range mínimo é de 70 metros, no entanto deve ser garantida uma margem de segurança para ambientes ruidosos, isto é, com redes que possam interferir na operação apropriada do sistema.

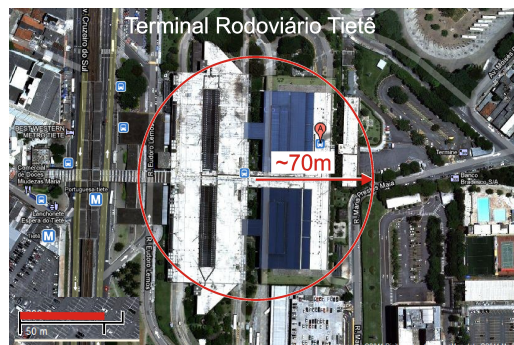


Figura 3.8: Terminal Rodoviário Tietê.

**Nós:** O número máximo de nós é especificado pelo número máximo de ônibus que se encontram na central ao mesmo tempo. No terminal Tietê o número de plataformas disponíveis é 89 [34]. Assim, assumindo o pior cenário, isto é que todas as plataformas estejam ocupadas ao mesmo tempo, o sistema deve estar especificado para suportar um mínimo de 89 nós.

Levando em conta o critério de alcance, tecnologias como Bluetooth® e IrDA® não satisfazem as exigências mínimas; enquanto que tecnologias como Wi-Fi™, GPRS/GSM e ZigBee® atingem maiores alcances garantindo a cobertura. No entanto, o ZigBee® apresenta outras vantagens tais como: um maior número de nós e menor custo quando comparado ao Wi-Fi™ e ao GPRS/GSM respectivamente. Um resumo das características dos padrões de comunicação sem fio é mostrado na Tabela 3.4.

Tabela 3.4: Características dos padrões de comunicação sem fio [12]

	<b>ZigBee®</b>	<b>Bluetooth®</b>	<b>Wi-Fi™</b>	<b>GPRS/GSM</b>
Foco de aplicação	Monitoramento e controle	Substituição de fios	Web, vídeo e email	Telefonia celular
Memória	4-32kB	250kB	1MB	16MB
Duração da bateria (dias)	100-1000	1-7	1-5	1-7
Nós por rede	255/65k	7	30	1000
Largura de Banda (kbps)	20-250	720	11	64-128
Alcance(m)	1-75	1-10	1-100	1000
Características-chaves	Confiável, baixo consumo, custo efetivo	Custo conveniente	velocidade flexível	alta qualidade

### 3.1.3.2 ZigBee® [35]

O ZigBee® é um protocolo de comunicação de alto nível que utiliza pequenos módulos de baixa potência. A tecnologia surgiu como uma alternativa mais simples e mais econômica do que os demais protocolos da rede WPAN (*Wireless Personal Area Network*) e com características desejáveis para a implementação em redes de sensores. Redes de sensores não precisam de elevadas larguras de banda, mas sim de baixa latência, baixo custo, alta confiabilidade e baixo consumo. O protocolo ZigBee® tem as seguintes características importantes:

- Baixo consumo de potência.
- Máxima taxas de dados de 20kbps, 40 kbps, 250 kbps e a 868MHz, 915MHz e 2.4GHz, respectivamente.
- Acesso ao canal com detecção de colisão.
- Baixo custo do dispositivo, de instalação e de manutenção.
- Alta densidade de nós por rede.
- Alta confiabilidade de dados.
- Diferentes topologias: estrela, ponto a ponto e Mesh.

Os módulos ZigBee® fornecem dois tipos de unidades, segundo sua função: *Full function devices*(FFD) e *Reduce function devices*(RFD). Os pontos da rede como o coordenador e os roteadores

precisam ser FFD, enquanto os pontos finais podem ser RFD economizando energia no modo *sleep*. Os módulos podem trabalhar em três modos de rede diferentes, como mostrado na Figura 3.9:

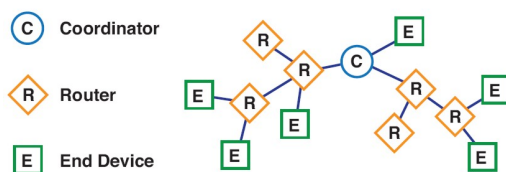


Figura 3.9: Nós da rede ZigBee®.

Os atores desta rede são descritos a seguir:

**Coordenador:** Este módulo requer maior capacidade de processamento do que os demais nós da rede, pois toda a informação da rede passa através deste. Além disso, este nó tem a capacidade de armazenar informação sobre a rede incluindo senhas de acesso.

**Roteadores:** Estes tipos de nós atuam como pontos intermediários, os quais reenviam os dados a outras unidades.

**Pontos finais:** Estes nós possuem modo *sleep* e podem ser configurados como de baixo consumo. Estes se comunicam com os módulos coordenadores ou roteadores, mas não retransmitem dados a outros pontos finais. Esta característica permite diminuir os custos e consumo.

Embora o ZigBee® satisfaça as especificações mínimas do projeto, este protocolo tem apenas o alcance mínimo, assim se buscou uma alternativa que forneça as mesmas qualidades de baixo custo e alta quantidade de nós, mas com maior alcance. Na mesma família dos protocolos 802.15.4 se encontra um protocolo chamado *DigiMesh™* desenvolvido pela companhia *Digi* e cujo módulo de 900MHz satisfaz amplamente o critério do alcance.

### 3.1.3.3 Digimesh™

O Digimesh™ é um protocolo ponto a ponto, não aberto, que fornece maior alcance e estabilidade da rede através da regeneração dos buracos na rede. O Digimesh™ somente tem um tipo de nó, os quais podem atuar como coordenador, roteador ou ponto final, como mostrado na Figura 3.10(a). Além disso, estes módulos suportam o modo *sleep* em todos os nós, o qual incrementa o tempo de vida das baterias em comparação com a rede ZigBee®. Ao mesmo tempo, este protocolo melhora

a confiabilidade da rede, pois os caminhos são redundantes fornecendo varias alternativas de comunicação, como mostrado na Figura 3.10(b), sendo uma característica importante em ambientes onde os roteadores podem ser isolados devido a interferências ou danos nestes nós. Diferentemente do ZigBee®, cuja frequência típica é 2.4GHz, o Digimesh™ possui frequências típicas de operação de 2,4GHz e 900MHz, permitindo um maior alcance nesta última frequência.

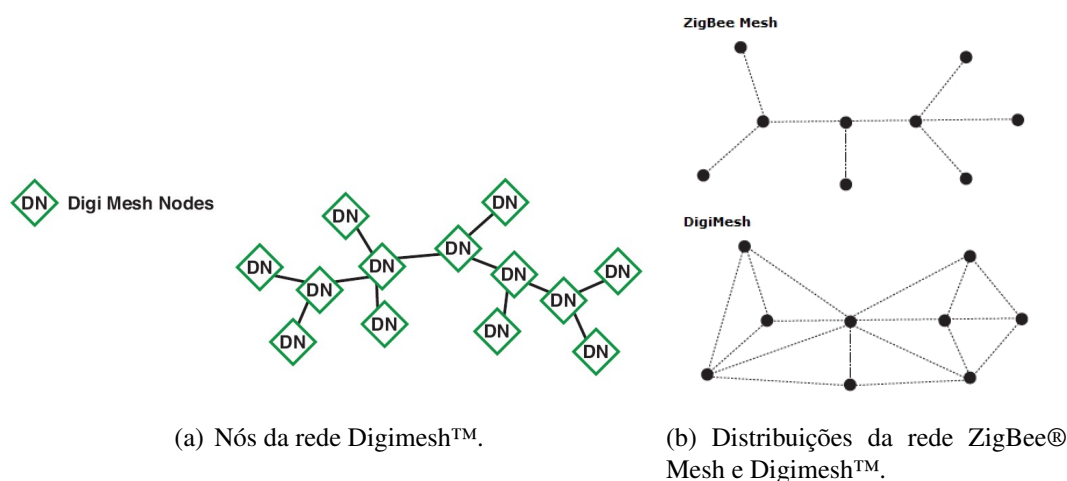


Figura 3.10: Rede Digimesh™.

A grande desvantagem do protocolo Digimesh™ é a não interoperabilidade com outros fabricantes, face seu protocolo proprietário, com menor disponibilidade de aplicações comuns e ferramentas de diagnóstico. As principais diferenças e similaridades entre os módulos comerciais dos protocolos ZigBee® e Digimesh™ são resumidas na Tabela 3.5.

Tabela 3.5: Comparação entre ZigBee® e Digimesh™ [13]

	<b>ZigBee® Pro Mesh</b>	<b>Digimesh™ Pro</b>
Tipo de nós	Coordenador, roteador, pontos finais	Somente um tipo
Nós com modo <i>sleep</i>	Somente nós finais	Todos os nós
Frequências predominantes	2.4GHz	2.4GHz/900MHz
Alcance com obstáculos	90m	90m/140m
Alcance sem obstáculos	3.2km	1.6km/10km
Largura de Banda (kbps)	1000	115/230
Corrente de transmissão(mA)	250-340	205/210
Corrente de recepção(mA)	55	47/80
Corrente de <i>sleep</i> (μA)	<50	4/60
Preço antena tipo fio(US\$)	28	32/39

Neste trabalho de mestrado será utilizado o módulo *XBee-PRO DigiMesh 900 - Wire antenna* devido ao seu alcance máximo em locais internos de 140 metros e a sua habilidade para formar redes

*Mesh* sem precisar de roteadores e coordenadores. Este módulo possui uma antena tipo fio e utiliza uma frequência de 900MHz. A descrição dos pinos se encontra na folha de especificação do *XBee-PRO DigiMesh* [36]. O módulo é mostrado na Figura 3.11.

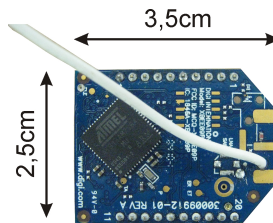


Figura 3.11: Módulo XBee-Pro Digimesh™ 900.

### 3.1.4 Memória SD

Uma memória SD é um cartão de memória não volátil desenvolvido para o armazenamento de dados em dispositivos eletrônicos. A *SD Card Association* define e regula as especificações (mecânicas, elétricas, sinalização e software) e padrões para o uso de cartões de memória SD em dispositivos eletrônicos. Estes padrões permitem utilizar as memórias SD como meio de comunicação entre diferentes dispositivos. Existem três tipos de cartões segundo a capacidade de armazenamento: a memória SD (até 2GB), a memória SDHC (entre 4GB a 32GB) e a memória SDXC (de 32GB até 2TB). A memória SD requer um range de alimentação entre 2.6-3.7V, utiliza uma frequência máxima de 50MHz e possui uma interface de 9 pinos (relógio, comando, 4 linhas de dados e 3 linhas de alimentação) [37]. Além disso, as memórias SD podem ser programadas utilizando o protocolo *SD Bus* ou o protocolo SPI. Neste trabalho a memória SD padrão, mostrada na Figura 3.12, é utilizada e programada através do protocolo *SD Bus*, também chamado MCI, o qual é discutido no apêndice D.

Uma memória SD pode demandar uma corrente de até 200mA para escrever um bloco de dados

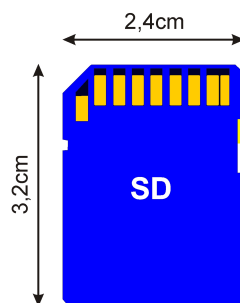


Figura 3.12: Memória SD.

com uma frequência de *clock* de 50MHz [37], assim é necessário implementar um circuito *driver*. Na Figura 3.13 é mostrado o diagrama esquemático de um *driver* não inversor, o qual controla a alimentação da memória, também mostra o resistor de *pull-up* de 33k $\Omega$  nas linhas de dados e controle da memória. Além disso, neste circuito se implementa um filtro utilizando um indutor de 470mA e um capacitor de 47 $\mu$ F ( $F_C=33,8$ Hz) para eliminar possíveis interferências devido às variações da alimentação da memória.

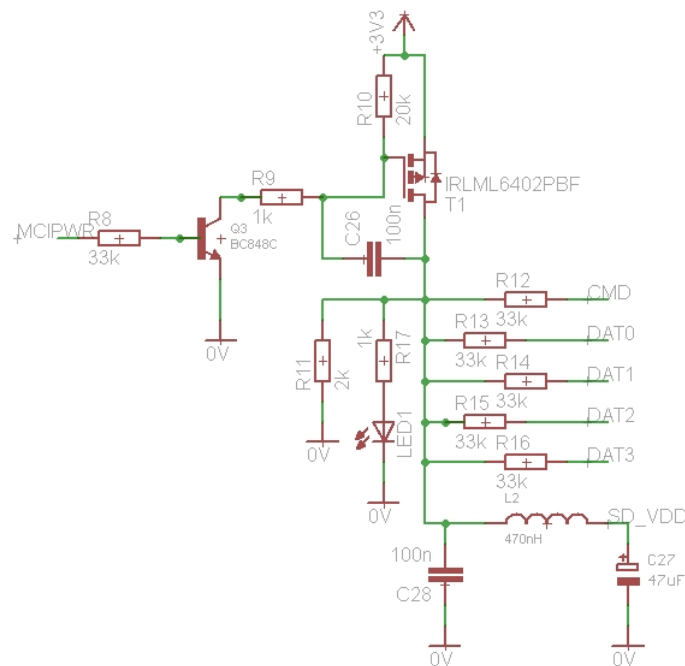


Figura 3.13: Diagrama de conexão da memória SD.

### 3.1.5 Microcontrolador

Um microcontrolador (MCU) é um circuito integrado que contém uma unidade de processamento, memória e periféricos de entrada-saída. Um microprocessador é uma unidade de processamento de propósito geral, como os usados em computadores; enquanto, o MCU é desenvolvido para atender tarefas específicas especialmente para aplicações de sistemas embarcados. Os MCUs podem ser classificados segundo a largura do barramento interno, arquitetura, memória e conjunto de instruções. Os MCUs podem ter memórias embarcadas ou externas e um conjunto de instruções extenso (CISC) ou um conjunto resumido (RISC). Finalmente dependendo da arquitetura pode-se considerar o MCU como Von-Neuman (Princeton) ou Harvard. O primeiro tipo armazena as instruções do programa e os dados na mesma memória, portanto utiliza o mesmo barramento para encaminhar instruções e dados;

enquanto a segunda arquitetura utiliza barramentos separados e memórias separadas [38].

A ARM é uma arquitetura de MCUs baseada em barramentos de 32-bits RISC, a qual é amplamente utilizada em sistemas embarcados, especialmente em telefones móveis, PDAs, reprodutores de música, roteadores dentre outros dispositivos de consumo. A ARM possui sete versões de dispositivos sendo que ARMv7 é a mais atual, também chamada Cortex. As versões ARMv4, ARMv5 e ARMv6 (chamadas ARM7, ARM9 e ARM11, respectivamente) correspondem às versões clássicas, sendo que a serie ARM7 é a linha de processadores mais utilizada para aplicações gerais, com inúmeras aplicações no mercado e mais amplamente utilizado desde 1994 [39]. Devido a seu custo, suporte técnico e diversidade de periféricos, o MCU LPC2478 da companhia *NXP Semiconductors* foi escolhido para ser utilizado neste projeto. A seguir são descritas algumas das características mais importantes deste MCU, as informações adicionais poderão ser consultadas na folha de especificação [40]:

- Processador ARM7TDMI-s de 72MHz com barramentos AHB (*Advanced High-performance Bus*).
- Memória flash de 512KB e SRAM de 98KB.
- Dispositivo de rede Ethernet.
- Dois barramentos *Controller Area Network* (CAN).
- Conversor analógico digital de 16 bits com 8 canais.
- Quatro temporizadores de 32 bits.
- Duas unidades de modulação por largura de pulso (PWM).
- Quatro portas seriais *Universal Asynchronous Receiver/Transmitter* (UART).
- Interface para cartão de memória SD/MMC.
- Três unidades I<sup>2</sup>C.
- Uma unidade SPI e duas *Synchronous Serial Port*(SSP).

### 3.1.6 Regulador de tensão

O regulador de tensão tem como objetivo garantir ao circuito uma tensão continua de 3,3V a partir da tensão da bateria do ônibus (12V) e em alguns casos para finalidade de testes o circuito será alimentado com uma bateria de tensão menor. Levando em conta os consumos máximos dos dispositivos discutidos anteriormente, mostrados na Tabela 3.6, o regulador precisa fornecer uma

corrente mínima de 570mA. O regulador escolhido neste trabalho é o LM1086IS-3.3 da *National Semiconductor* o qual possui uma capacidade de corrente de até 1,5A, uma tensão máxima de 27V e um baixo *dropout*. Esta ultima característica permite garantir uma regulação de 3,3V para tensões de alimentação a partir de 4,8V. O diagrama esquemático do circuito periférico ao regulador é mostrado na Figura 3.14.

Tabela 3.6: Consumo de corrente dos dispositivos do SEMC-TP

Dispositivo	Corrente (mA)
Módulo GPS [11]	35
Módulo Digimesh™ [36]	210
Acelerômetro [6]	0,14
Memória SD [37]	200
Sensor de temperatura [7]	0,15
Microcontrolador @75MHz [40]	125
TOTAL	570,29

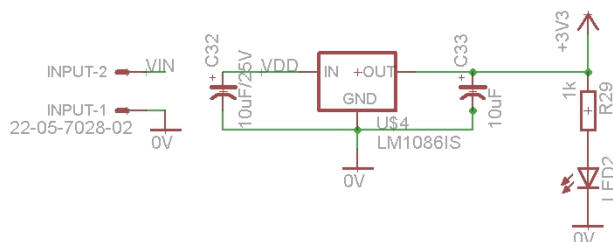


Figura 3.14: Circuito periférico do regulador de tensão.

## 3.2 SEMC-TP

O Sistema Embarcado para o Monitoramento do Transporte Público, SEMC-TP, utiliza os dispositivos previamente discutidos em um hardware que é capaz de adquirir e processar dados com o objetivo de avaliar o conforto em um veículo do sistema de transporte coletivo. O SEMC-TP está dividido em duas partes: transmissor e receptor.

### 3.2.1 SEMC-TP Transmissor

A primeira parte do SEMC-TP é um nó que se encarrega de coletar e processar a informação para posteriormente armazenar estes eventos numa memória. Estes nós são instalados em todos os



ônibus monitorados e quando estes chegam à estação, a informação é transmitida através do módulo de comunicações até um nó receptor. O diagrama de blocos é mostrado na Figura 3.15, no qual os periféricos são conectados utilizando os barramentos I<sup>2</sup>C, UART e MCI. A placa de cada nó é construída em duas camadas, na Figura 3.16 é mostrada uma imagem do protótipo de teste sendo alimentada com uma bateria de telefone sem fio. No apêndice E é mostrado o diagrama esquemático completo e o *layout* do SEMC-TP transmissor.

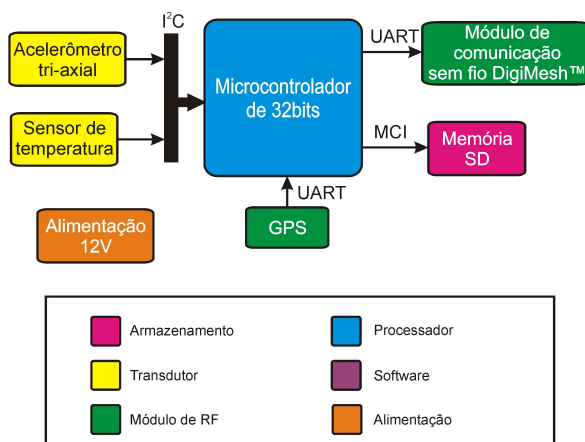


Figura 3.15: Diagrama de blocos do SEMC-TP transmissor.

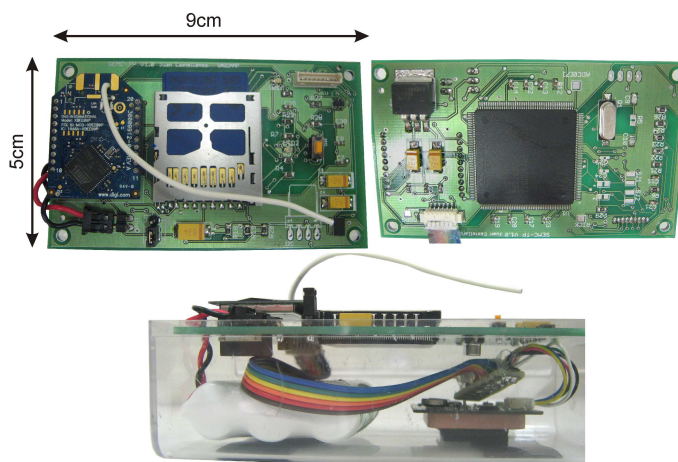


Figura 3.16: Montagem de teste para o SEMC-TP transmissor.

Conforme levantamento do custo dos componentes anteriormente descritos e dos componentes eletrônicos adicionais (capacitores, indutores, leds, resistores e transistores), mostrado na Tabela 3.7; o custo total do protótipo do nó transmissor é de R\$577,49, estima-se que na fabricação de 100 nós esse custo se reduziria para R\$195,2 por unidade.

Tabela 3.7: Custo do protótipo do SEMC-TP

Descrição	Custo unitário(Reais)	Custo estimado*(Reais)
Acelerômetro ADXL345	12,97	9,79
Microcontrolador LPC2478	31,31	0,24
Cartão de memória SD	24,00	10,73
Módulo GPS ME-1000RW	180,00	50,88
Módulo Digimesh™ 900	245,00	61,61
Elementos eletrônicos discretos	59,24	59,24
Placa de circuito impresso	29,97	2,70
TOTAL	577,49	195,20

\*Custo estimado para 100 unidades e dólar cotado a 1,59 reais.

### 3.2.2 SEMC-TP Receptor

O nó receptor é instalado na estação central e se encarrega de transmitir um sinal de *broadcast* solicitando que os ônibus que chegam à estação, transmitam suas informações. Este nó transmite os dados coletados para um computador, no qual os eventos são armazenados e analisados utilizando uma interface de usuário desenvolvida no software LabVIEW®. Também é possível realizar a calibração do sistema, configurar os limiares e número de ônibus através deste receptor. O hardware do SEMC-TP receptor, mostrado na Figura 3.17(a), está baseado em um adaptador USB para os módulos ZigBee® e DigiMesh™ desenvolvido pela companhia *Rogercom*®. Esta placa além de fornecer a alimentação, também realiza a interface entre a saída e entrada UART dos módulos e a porta USB de um computador, como mostrado pelo diagrama de blocos da Figura 3.17(b). O custo desta placa é de R\$260,00, conforme o site do fabricante (*Rogercom*®).

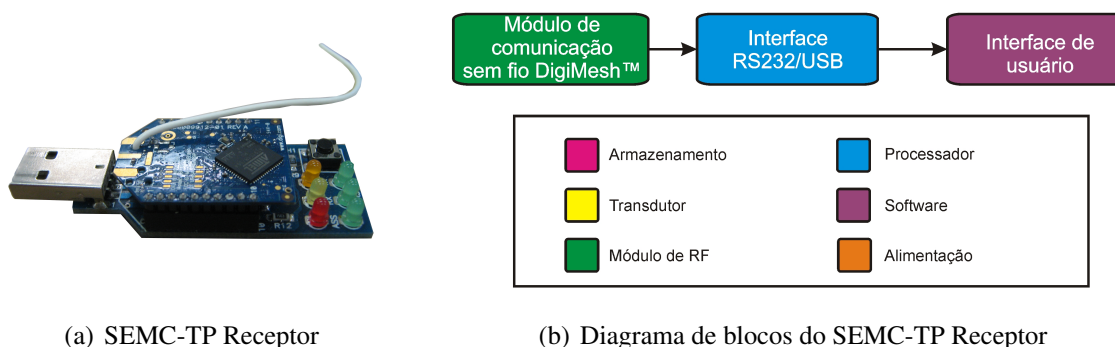


Figura 3.17: SEMC-TP Receptor.

# Capítulo 4

## *Firmware* e Instrumento virtual do sistema

Neste capítulo será abordado o *firmware* do microcontrolador e o instrumento virtual (IV) desenvolvido na plataforma LabVIEW®. Na primeira parte, três algoritmos implementados são apresentados, os quais estão baseados em conceitos discutidos previamente e que têm como finalidade avaliar o conforto no transporte público. Estes três algoritmos, que fazem parte do *firmware*, são: detecção de eventos de desconforto utilizando o limiar de aceleração (DEL-A), detecção de eventos de desconforto utilizando o limiar de aceleração e *Jerk* (DEL-AJ) e detecção de eventos de desconforto utilizando o limiar de aceleração e avaliação através do índice de conforto (DEL-AIC). Também são descritas sucintamente as rotinas de inicialização para o microcontrolador, UARTs, acelerômetro, sensor de temperatura e memória SD, depois são apresentadas as rotinas de interrupção e as rotinas específicas de cada algoritmo. Na segunda parte deste capítulo é apresentado o instrumento virtual baseado numa interface gráfica que utiliza o Googlemaps™ como ferramenta visual para apontar a localização geográfica dos eventos de desconforto.

### 4.1 Algoritmos para avaliar o conforto

#### 4.1.1 Algoritmo para o DEL-A

O algoritmo utilizado para a detecção de eventos de desconforto utilizando o limiar de aceleração (DEL-A) é também a rotina base para os algoritmos DEL-AJ e DEL-AIC. O algoritmo DEL-A inicializa todos os módulos de comunicação e sensores para depois realizar a leitura dos dados de configuração de um arquivo armazenado previamente na SD, este arquivo (nomeado conf.txt) possui informação em formato ASCII dos limiares de aceleração nos três eixos (X,Y,Z), o limiar de temperatura, o limiar do *Jerk* (para o DEL-AJ), o identificador (ID) do ônibus ou veículo e os três valores de calibração do acelerômetro. Na sequência, este algoritmo armazena os eventos de desconforto

em um arquivo na memória SD. Em seguida, uma interrupção do *timer* é ativada coletando os dados de aceleração com duração de 1s, isto é, 500 dados por cada eixo de aceleração (X, Y e Z). A cada segundo é verificado, através da interrupção do *timer*, se o máximo de aceleração supera o limiar; nesse caso é ativada uma rotina para ler a informação do GPS e armazenada a informação do evento. Outras interrupções possíveis do DEL-A, são causadas pelas portas seriais (UART). A interrupção da UART1 se ativa quando uma mensagem do protocolo NMEA que contem a informação da posição é recebida com sucesso. A interrupção da UART2 se ativa quando um comando de controle é enviado através do módulo de comunicações. A Figura 4.1 mostra o fluxograma da rotina principal deste algoritmo. A diferença entre os algoritmos DEL-AJ, DEL-AIC para o DEL-A é a inclusão da caixa em vermelho. As rotinas de inicialização, as três principais interrupções e as rotinas específicas do algoritmo são tratadas a seguir.

#### 4.1.1.1 Rotinas de inicialização

A seguir são descritas sucintamente as rotinas de inicialização do microcontrolador e dos periféricos. Algumas das rotinas foram baseadas nas rotinas de inicialização do microcontrolador LPC2478 fornecidas na documentação do software *IAR® Embedded Workbench IDE*, o qual foi utilizado como ambiente de trabalho para a programação do microcontrolador.

- **Microcontrolador:**

As funções básicas que devem ser inicializadas no microcontrolador são: o fator de multiplicação de frequência do PLL (*phase lock loop*), as frequências de oscilação dos periféricos e temporizadores, a declaração das entradas e saídas do GPIO (*General Purpose Input/Output*) e o restabelecimento dos valores predefinidos para o vetor de interrupções.

- **UARTs:**

Neste trabalho foram utilizadas duas UARTs, a primeira se encarrega de receber os dados do GPS, enquanto a segunda se encarrega de receber e enviar informação ao módulo de comunicações. A rotina de inicialização das duas UART configura método de comunicação: sem paridade, com uma largura de 8 bits, um bit de parada e sua interrupção. A UART do GPS (UART1) é configurada para se comunicar com uma velocidade de 4800bps enquanto a UART do módulo de comunicações (UART2) é configurada a uma velocidade de 19200bps.

- **Acelerômetro:**

A primeira parte da rotina de inicialização do acelerômetro é inicializar a comunicação da interface I<sup>2</sup>C, a qual foi inicializada com uma velocidade de 400kHz e utiliza os pinos 50 e 48, para as linhas SDA e SCL, respectivamente. A segunda parte da rotina configura o fundo

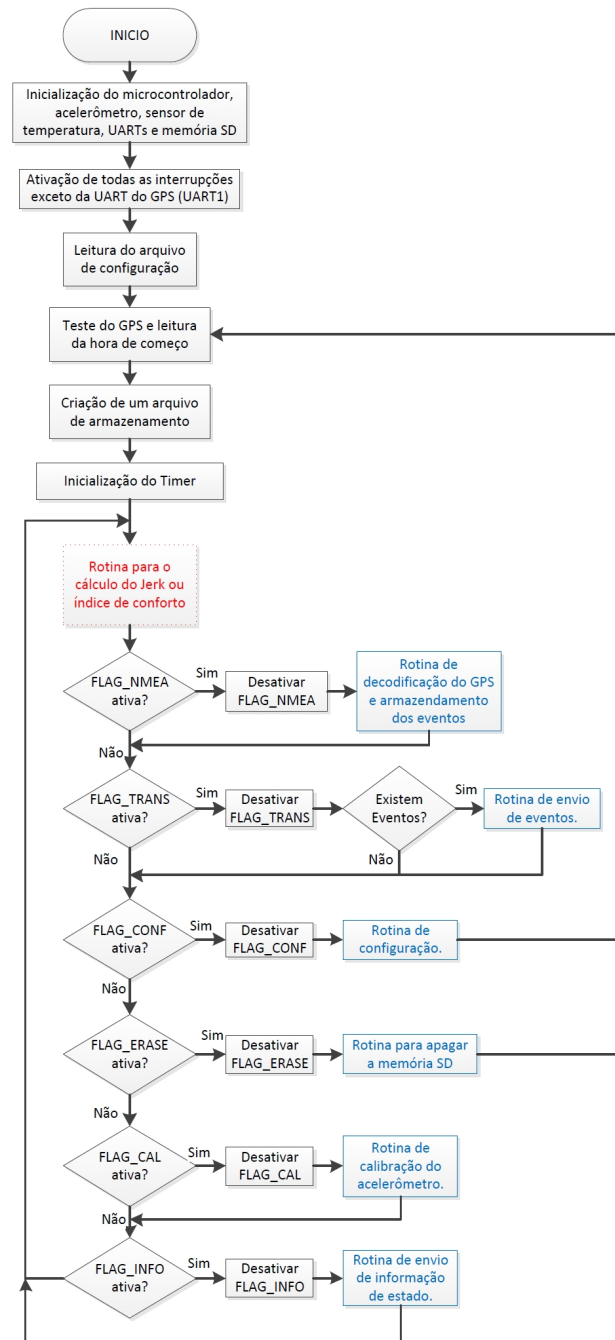


Figura 4.1: Fluxograma da rotina principal do algoritmo DEL-A, DEL-AJ e DEL-AIC.



#### 4.1.1.2 Rotinas de interrupção

A seguir são descritas sucintamente as três principais rotinas de interrupção do algoritmo, as quais influenciam diretamente no funcionamento do programa principal.

- **Interrupção do *Timer*:**

A interrupção do *Timer*, mostrada na Figura 4.3, é executada a cada 2ms, na qual é realizada a leitura do acelerômetro e o sensor de temperatura. No entanto, a cada 500 interrupções (ou seja, a cada segundo) é calculado o máximo valor absoluto dos três eixos de aceleração e o máximo valor da temperatura. Caso estes valores sejam maiores que os limiares estabelecidos se define o tipo de evento, conforme a Tabela 4.1, e se ativa a leitura do GPS através da ativação da interrupção da UART1.

Tabela 4.1: Tipo do evento

Tipo de evento	Causa
1	Máxima aceleração X é maior do que o limiar X
2	Máxima aceleração Y é maior do que o limiar Y
3	Máxima aceleração Z é maior do que o limiar Z
4	Máxima aceleração X, Y é maior do que o limiar X, Y
5	Máxima aceleração X, Z é maior do que o limiar X, Z
6	Máxima aceleração Y, Z é maior do que o limiar Y, Z
7	Máxima aceleração X, Y, Z é maior do que o limiar X, Y, Z
8	Máxima temperatura é maior do que o limiar de temperatura
9	Máximo <i>Jerk</i> maior do que o limiar do <i>Jerk</i> ou índice de conforto calculado
0	Não evento

- **Interrupção da UART1(GPS):**

Esta interrupção, mostrada na Figura 4.4, é ativada quando se deseja ler a informação enviada pelo GPS e recebida pela porta serial UART1. Assim, são armazenadas as mensagens do NMEA para sua posterior decodificação segundo o tipo de mensagem (GGA, GSA, GSV ou RMC).

- **Interrupção da UART2 (Módulo de comunicação):**

Esta interrupção, mostrada na Figura 4.5, é ativada quando uma mensagem de controle é recebida através do módulo de comunicação, isto é pela porta serial UART2.

O comando é composto por um caractere chave que indica o tipo de comando e um parâmetro, o qual em alguns casos é vazio. Os comandos de controle são descritos a seguir:

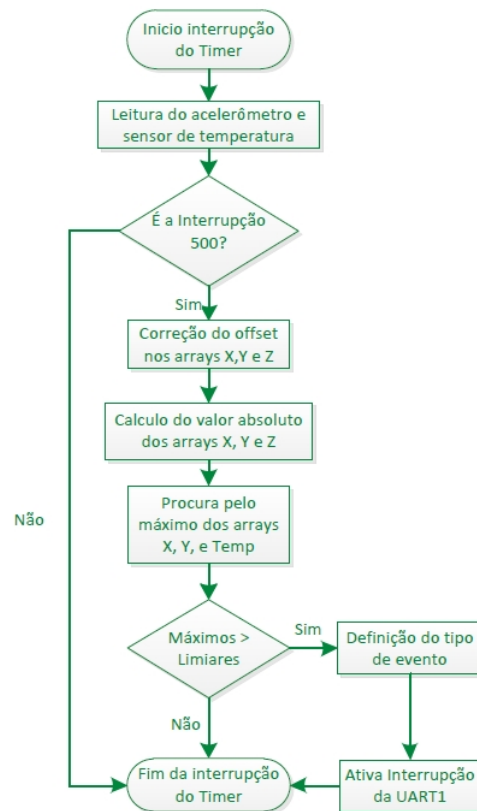


Figura 4.3: Fluxograma do algoritmo de interrupção do *timer*.

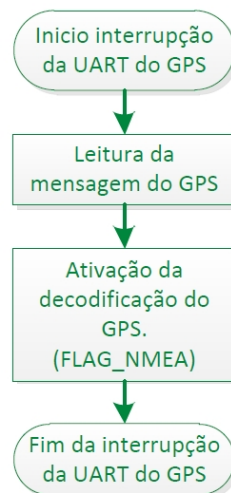


Figura 4.4: Fluxograma do algoritmo de interrupção da UART1 (GPS).



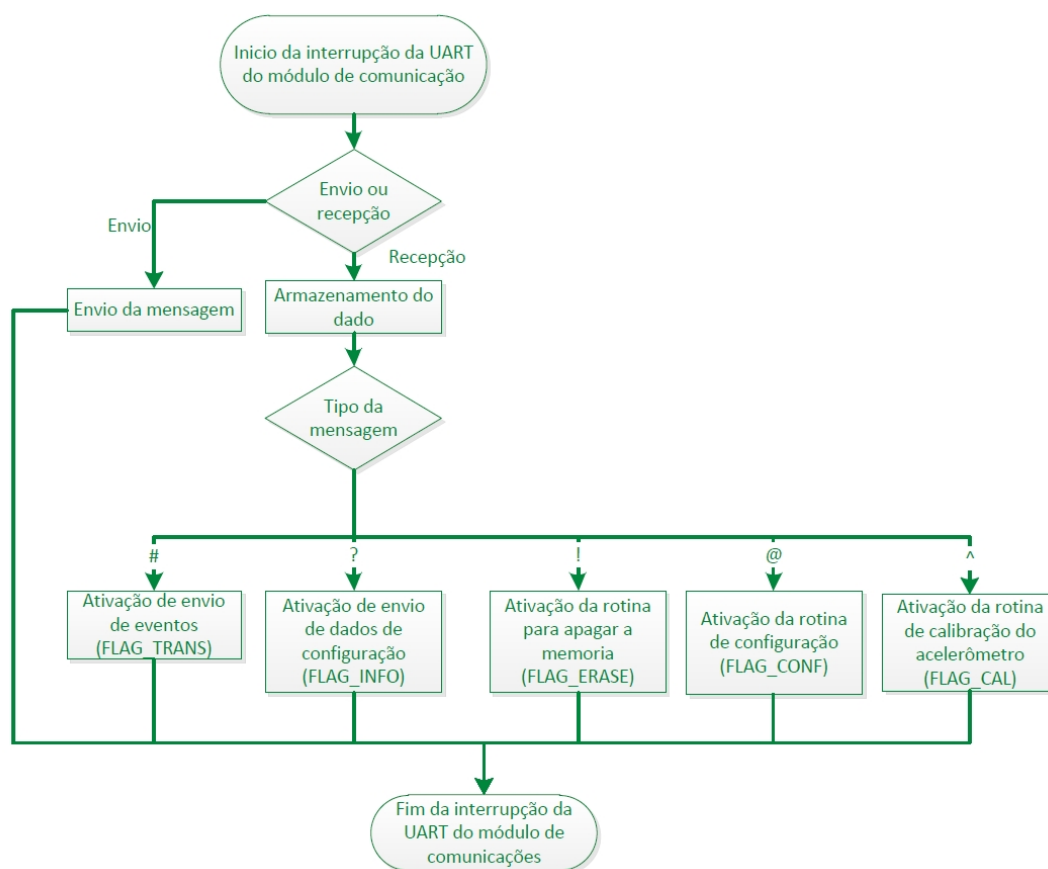


Figura 4.5: Fluxograma do algoritmo de interrupção da UART2 (Módulo de comunicação).

- Comando de envio de eventos: A sintaxe do comando é: `#[ID do veículo]`. Este comando envia os eventos armazenados no buffer ao arquivo corrente da memória SD e posteriormente envia este arquivo à estação base através do módulo de comunicação.
- Comando de envio de dados de configuração: A sintaxe do comando é: `?`. Este comando envia as informações armazenadas no arquivo de configuração da memória SD, tais como: ID do nó e limiares. Também, são enviados dois indicadores, os quais apontam se houve ocorrência de eventos e se está presente o cartão de memória SD. Este comando pode ser utilizado como comando de *broadcast* para identificar os ônibus com eventos disponíveis que chegam à estação de ônibus para serem posteriormente descarregados.
- Comando para apagar os arquivos de eventos: A sintaxe do comando é: `![senha]`. Este comando apaga os arquivos de eventos previamente criados em percursos anteriores. O arquivo de configuração não é apagado. A senha enviada é comparada com a mesma definida no firmware do microcontrolador (predefinida como 1234).
- Comando de configuração: A sintaxe do comando é: `@[tipo de configuração][valor]`

- . Este comando configura os limiares de aceleração, temperatura, *Jerk* (no caso do algoritmo DEL-AJ), e ID. Para a configuração dos limiares de aceleração envia-se, no campo do tipo de configuração, o eixo (X,Y,Z) e os novos valores de aceleração em mG. No caso do limiar de temperatura envia-se o comando T e o valor da temperatura, multiplicada por dez e em graus centígrados. No caso do limiar do *Jerk* (do algoritmo DEL-AJ) envia-se o comando J e o valor em  $10^{-3}\text{m/s}^3$ . Finalmente, para o caso da configuração do ID do nó, envia-se o comando B e o valor do novo ID com 5 dígitos.
- Comando de calibração do acelerômetro: A sintaxe do comando é:  $\wedge$ . Este comando realiza a calibração do *offset* do acelerômetro nos três eixos evitando possíveis erros devido à instalação do nó na estrutura do ônibus.

#### 4.1.1.3 Rotinas específicas do algoritmo

Dentro da rotina principal existem seis rotinas específicas, as quais são ativadas pelas interrupções das UARTs, mas que são executadas fora destas e descritas a seguir:

- **Rotina de decodificação e armazenamento:**

Depois que é feita a leitura do GPS, a mensagem NMEA é tratada pela rotina de decodificação dependendo do tipo da mensagem, como mencionado no apêndice C. As mensagens que são decodificadas são a GGA e a RMC, pois estas contêm a informação de latitude e longitude. Depois que estas informações são armazenadas, um vetor de ocorrência de evento é gerado. Este vetor contém os campos mostrados na Figura 4.6. Estes vetores são armazenados em um buffer no microcontrolador até completar um bloco de 512 caracteres ASCII (512 bytes). Quando completa o conteúdo do buffer é transferido ao arquivo de dados criado na SD. O fluxograma é mostrado na Figura 4.7(a).

\$	ID do ônibus	-	Número do evento	Hora	Tipo	Amplitude	Latitude	Longitude
----	--------------	---	------------------	------	------	-----------	----------	-----------

Figura 4.6: Estrutura do vetor de ocorrência de eventos.

- **Rotina de configuração do nó:**

Esta rotina recebe os dados do comando de configuração (@) enviados através da UART2. Os limiares X,Y, Z, *Jerk* e o ID do nó são atualizados por meio da leitura do arquivo de configuração, alteração dos campos desejados e posterior reescrita destes na memória SD. O fluxograma é mostrado na Figura 4.7(b).

- **Rotina de envio de eventos:**

Nesta rotina, realiza-se a leitura do ID do veículo enviado no comando, o qual é comparado com o mesmo armazenado no arquivo de configuração do nó, com a finalidade de avaliar a identidade do nó e evitar a resposta de um nó indesejado. Depois disto, os eventos gerados e ainda armazenados no buffer são transferidos à memória para finalmente transferir o arquivo de eventos completo, em pacotes de 512B, através do módulo de comunicações (UART2). O fluxograma é mostrado na Figura 4.7(c).

- **Rotina de envio de informação do nó:**

Nesta rotina, a informação do arquivo de configuração é encaminhada ao módulo de comunicações (UART2), junto com a informação da existência de eventos e a verificação que a memória SD está alojada no nó. O fluxograma é mostrado na Figura 4.7(d).

- **Rotina para apagar os arquivos da memória SD:**

Nesta rotina, realiza-se a leitura da senha enviada no comando, a qual é comparada com a mesma armazenada no firmware do nó, com a finalidade de proteger os dados. Depois disto, os arquivos de dados são apagados escrevendo 0x00 tanto nas posições destes no diretório raiz, como também nos *clusters* de dados dos arquivos para evitar que permaneçam dados antigo quando se slava um novo arquivo. O fluxograma é mostrado na Figura 4.7(e).

- **Rotina para calibrar o acelerômetro:**

Nesta rotina é feita a leitura do acelerômetro do nó instalado no veículo, o qual deve estar em repouso e os eixos Z e X alinhados com os eixos de gravidade e de movimentação longitudinal, respectivamente. Realizou-se 10 leituras do acelerômetro e depois se calculou a média para garantir a fidelidade dos valores de calibração [6]. Posteriormente, são substituídos os campos de calibração do acelerômetro armazenados no arquivo de configuração da memória SD. Finalmente, é enviada a resposta do comando de calibração, através do módulo de comunicação, a qual contém a confirmação dos novos valores. O fluxograma é mostrado na Figura 4.7(f).

#### 4.1.2 Algoritmo para o DEL-AJ

O algoritmo do DEL-AJ está baseado no fluxograma da Figura 4.1, previamente discutido, com exceção à uma única rotina específica, mostrada na Figura 4.8, que consiste na implementação do algoritmo do filtro, decimação e derivação para calcular o *Jerk* longitudinal (Figura 2.25), descrito na análise das características do sinal de *Jerk* em veículos. Esta rotina é programada para execução a cada 30s, isto é o tempo suficiente para coletar um período de um sinal de 0,03Hz e dez períodos do

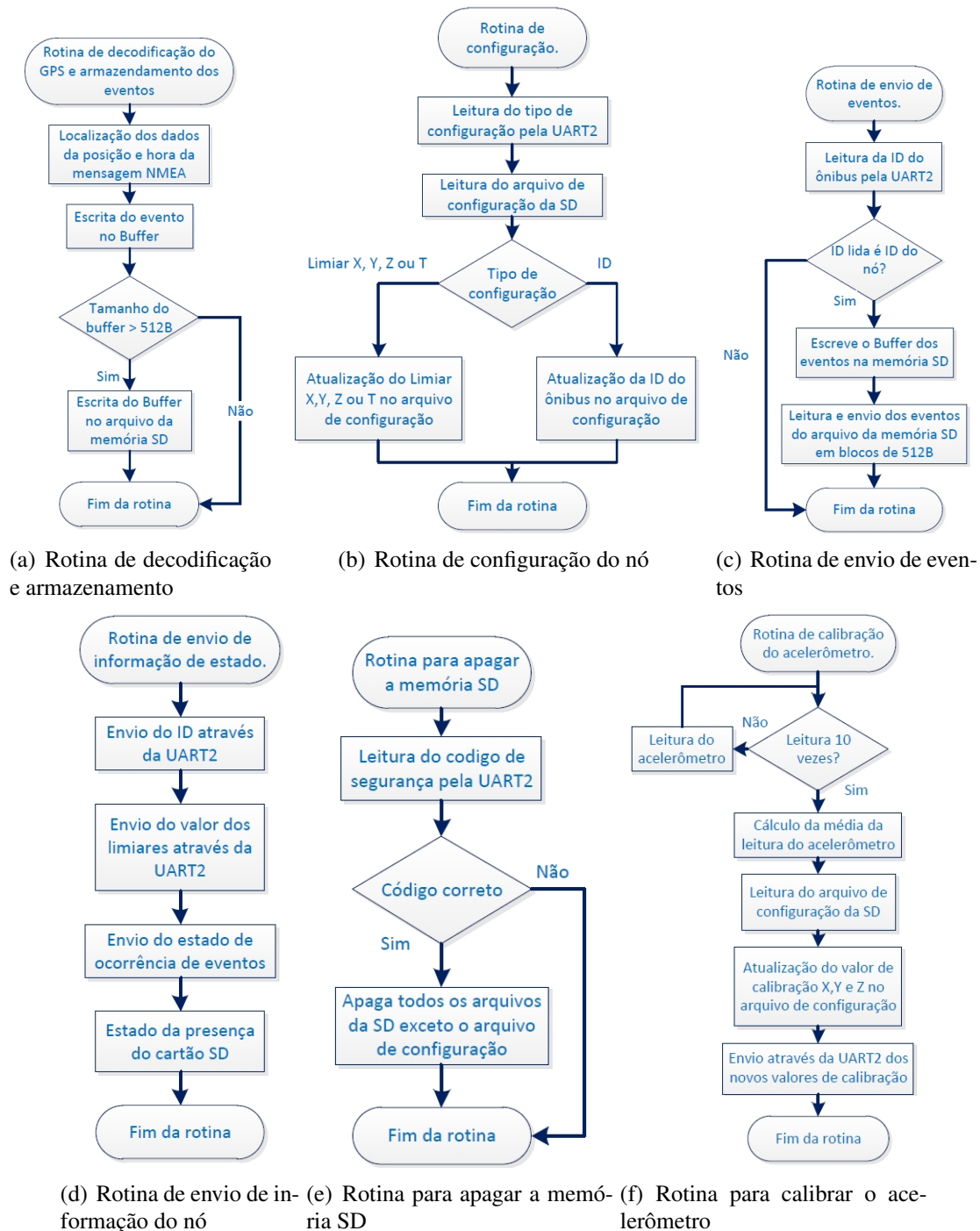


Figura 4.7: Fluxograma para algoritmos com funções específicas

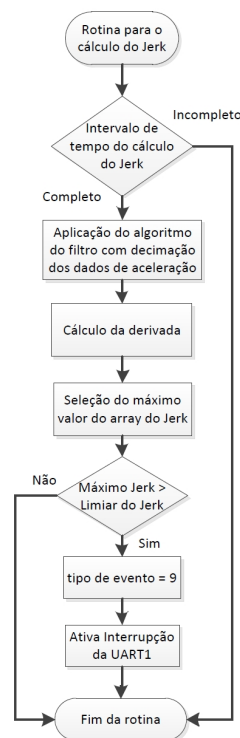


Figura 4.8: Fluxograma do algoritmo para a detecção de eventos devidos ao *Jerk* (DEL-AJ).

signal da frequência de corte do filtro (0,3Hz). Calculado o vetor do *Jerk*, procura-se o máximo valor absoluto deste e se compara com o limiar do *Jerk* ( $0,4\text{m/s}^3$ ) para detetar a ocorrência de um evento de desconforto. Caso seja detectado um evento, definido como tipo 9 (Tabela 4.1), ativa-se a interrupção da UART1, ou seja, a leitura do GPS.

### 4.1.3 Algoritmo para o DEL-AIC

O algoritmo do DEL-AIC também está baseado no fluxograma da Figura 4.1, previamente descrito. A rotina de inicialização do acelerômetro é alterada para configurar a largura de banda deste a 100Hz, como recomendado pela norma ISO2631-1 [1]. A rotina específica, mostrada na Figura 4.9, consiste na implementação do algoritmo para calcular o índice de conforto.

O algoritmo para calcular o valor RMS ponderado total consiste na aplicação direta da norma internacional ISO 2631-1, seu diagrama de blocos é mostrado na Figura 4.10.

Primeiramente são calculadas as transformadas rápida de Fourier, FFT, de cada eixo e depois são aplicadas as curvas de ponderação em frequência, mostradas na Figura 4.11, provenientes da função de transferência da ponderação do sinal de aceleração dos eixos horizontais e vertical de acordo com a equação 1.16.

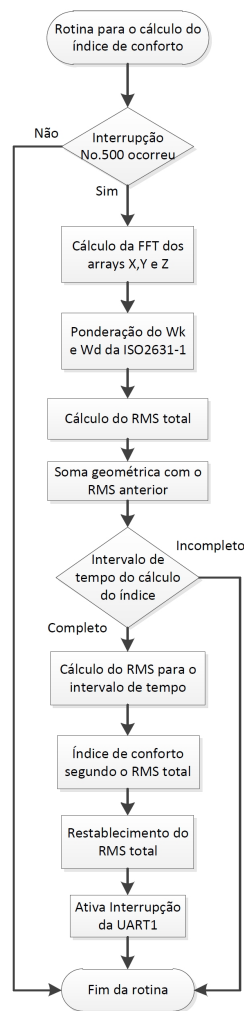


Figura 4.9: Fluxograma do algoritmo para o cálculo do índice de conforto (DEL-AIC).

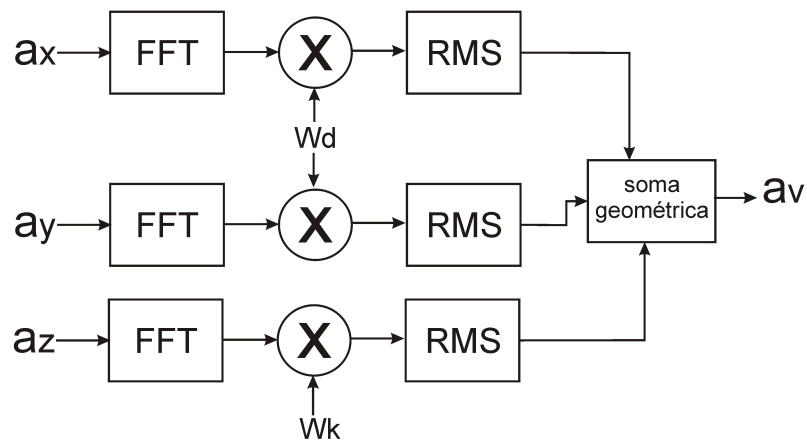


Figura 4.10: Fluxograma do algoritmo para o cálculo do valor de aceleração RMS ponderado.

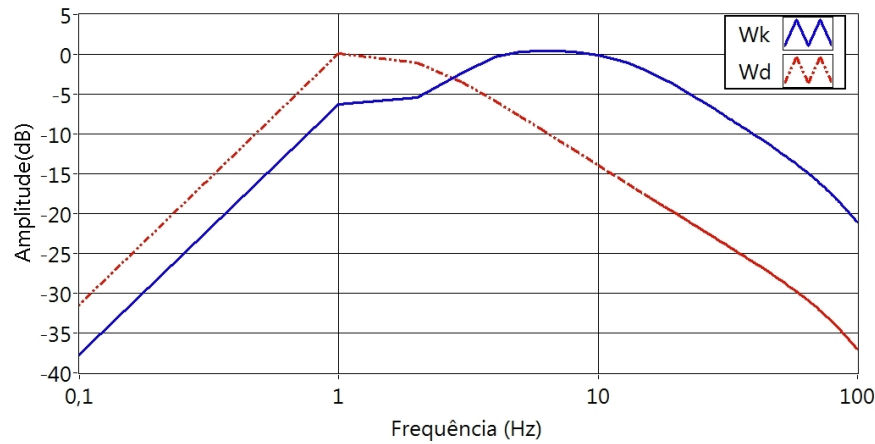


Figura 4.11: Curvas de ponderação,  $W_K$  e  $W_d$ , com 512 pontos baseadas na norma ISO2631-1.

Posteriormente, calculado-se o valor RMS para cada eixo diretamente do domínio da frequência, utilizando a expressão:

$$a_w = \frac{1}{N} \left( \sum_N a_w(f)^2 \right)^{1/2} \quad (4.1)$$

sendo que  $a_w$  é a aceleração( $m/s^2$ ) RMS ponderada,  $a_w(f)$  são os componentes da aceleração ponderadas na frequência e  $N$  o número de amostras utilizados para obter a FFT. O fator  $(\frac{1}{N})$  nesta expressão deve-se ao ajuste da amplitude dos componentes em frequência, pois não é considerada na transformada rápida de Fourier. Finalmente, o valor RMS total obtido para cada segundo é calculado realizando a soma geométrica dos valores RMS de aceleração nos três eixos, segundo a equação 1.17, com os fatores multiplicativos,  $K_x$ ,  $K_y$  e  $K_z$ , iguais a 1. Neste caso consideramos que os passageiros estão sendo transportados em pé.

Este algoritmo é programado para execução a cada segundo. Os valores RMS totais a cada segundo são utilizados para calcular o valor RMS em um tempo maior, neste caso a cada 15min. Para isto é utilizada o seguinte:

$$a_{RMS_t} = \left[ \frac{1}{t} \sum_{i=0}^t a_{v_i}^2 \right]^{\frac{1}{2}} \quad (4.2)$$

sendo  $a_{RMS_t}$  a aceleração RMS ponderada total avaliada durante  $t$  segundos e o  $a_{v_i}$  é a aceleração RMS ponderada total avaliada durante um segundo.

Assim, a cada 15 minutos o resultado da aceleração RMS ponderada total é classificado segundo a Tabela 4.2, para encontrar o índice do conforto, sendo que 0 é o nível mínimo de conforto e 5 o nível máximo de conforto. Esta tabela foi baseada na norma internacional ISO2631-1 (Tabela 1.10),

Tabela 4.2: Índice de conforto conforme aceleração RMS ponderada para o DEL-AIC

Índice de conforto	Range da Aceleração RMS (G)	Conceito
5	Menor do que 0,032	Sem desconforto
4	0,032 - 0,057	Desconforto leve
3	0,057 - 0,092	Desconforto moderado
2	0,092 - 0,145	Desconfortável
1	0,145 - 0,229	Desconforto alto
0	Maior do que 0,229	Desconforto muito alto

a diferença entre sua versão original e a tabela utilizada nesta dissertação está na inexistência de sobreposição de faixas, facilitando assim a classificação em uma única categoria. Finalmente, é ativada a leitura do GPS para associar o índice de conforto ao trecho do percurso analisado e se restabelece o valor de aceleração RMS total a zero.

As partes mais relevantes do código do *Firmware* do microcontrolador são apresentadas no apêndice F

## 4.2 Interface gráfica do Instrumento Virtual do sistema

A segunda parte do sistema é composta pelo Instrumento Virtual que realiza a interface do sistema com o usuário, permitindo a visualização dos eventos de desconforto e a configuração dos valores de cada nó. Este IV está dividido em duas partes.

Na primeira parte são mostrados os eventos em uma interface de mapa online. Este IV está baseado na visualização de uma página web em LabVIEW™. Para criar e visualizar os eventos, utilizou-se o recurso de mapas: Google Maps Javascript API V2; essa *application programming interface*(API) permite empacotar a aplicação em um browser usando JavaScript. A API provê uma grande gama de ferramentas que também estão disponíveis na aplicação do *Googlemaps*™, tais como: Geodecodificação, visualização de rua, integração com *Google Earth*, camadas de tráfego, dentre outras [41]. Os vetores de ocorrência de eventos armazenados em arquivos no PC são lidos por esta primeira parte da interface do SEMC-TP e são criados os marcadores como mostrado na Figura 4.12. Cada marcador contém informações tais como: número do evento, ID do veículo, hora, tipo e amplitude do evento.

A segunda parte do software, mostrada na Figura 4.13, é projetada para a interação direta com o nó. Utilizando a interface serial do LabVIEW™ e sendo que o SEMC-TP receptor é visto pelo computador como uma porta serial virtual. As seguintes tarefas são realizadas por meio do envio de comandos através do módulo de comunicações do SEMC-TP receptor:

- Descarregar o arquivo de eventos através do enlace sem fio e armazenamento automático dentro



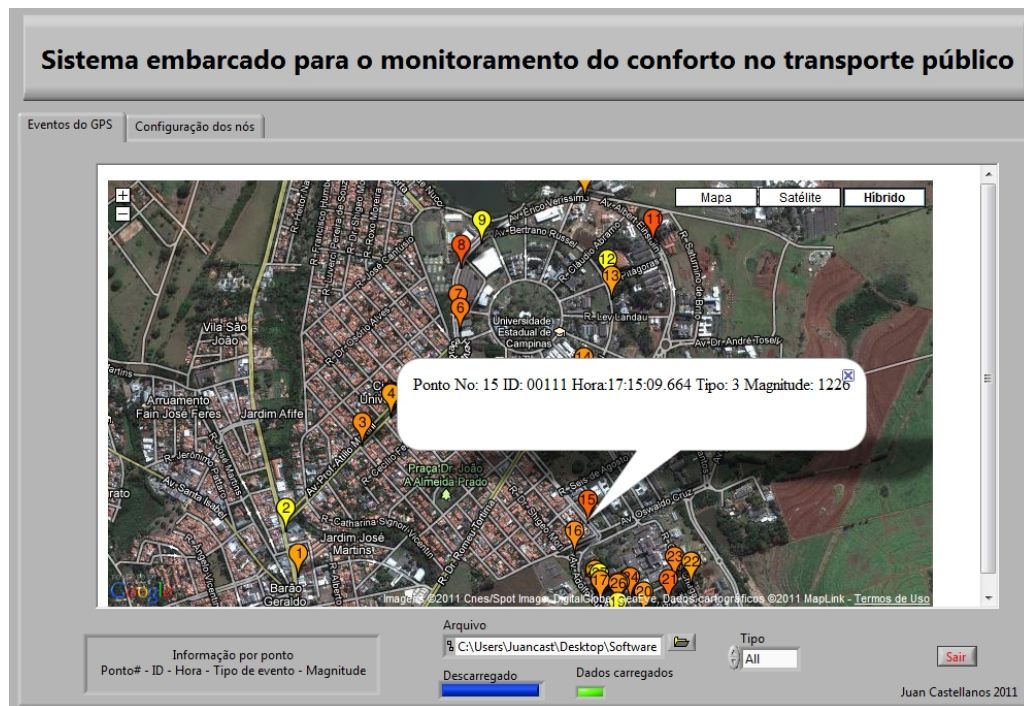


Figura 4.12: Tela do software do SEMC-TP para a visualização dos eventos.

de uma pasta no computador receptor, a qual possui a informação do ID do veículo e data da leitura.

- Apagar todos os arquivos dos percursos armazenados previamente da memória SD.
- Ler a informação sobre os limiares programados no nó, estado de eventos gerados e estado da memória SD.
- Configuração dos limiares de aceleração.
- Configuração do limiar de temperatura.
- Configuração do ID do nó (ônibus).
- Calibração do acelerômetro.

No apêndice G é mostrado a programação do IV através da plataforma LabVIEW™ que foi utilizada para a implementação desta interface.

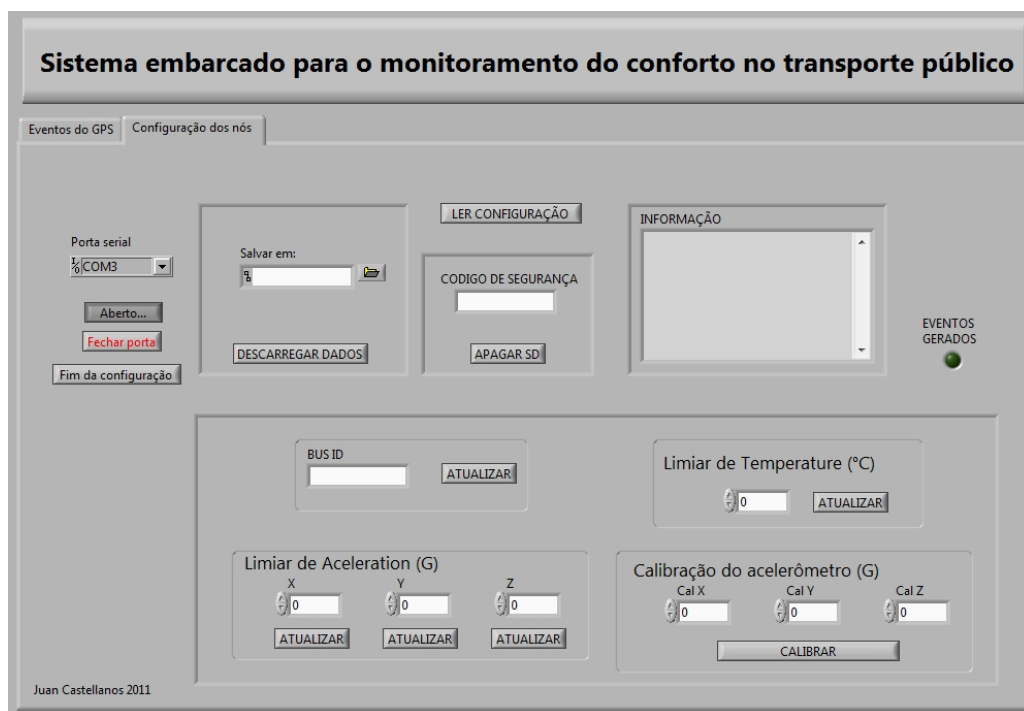


Figura 4.13: Tela do software do SEMC-TP para a configuração dos nós e descarga de eventos.

# Capítulo 5

## Resultados experimentais

Neste capítulo são apresentados os resultados experimentais dos testes realizados com o protótipo final do SEMC-TP utilizando os três algoritmos DEL-A, DEL-AJ e DEL-AIC. Os eventos são visualizados através do instrumento virtual feito em LabVIEW™, descrito anteriormente. Estes testes foram realizados em meios de transporte, tais como: carros particulares, ônibus urbanos e transporte coletivo fretado. Também são apresentados os resultados de consumo de corrente e aquecimento do sensor de temperatura devido à dissipação de potência do transmissor.

### 5.1 Resultados utilizando o algoritmo DEL-A

#### 5.1.1 Primeiro teste do DEL-A

O SEMC-TP com o algoritmo DEL-A foi primeiramente testado em uma viagem de carro em um percurso dentro da UNICAMP. Este percurso conta com características que permitem avaliar os eventos de aceleração nos três eixos, pois o percurso traçado na Figura 5.1 contém duas rotatórias, uma reta comprida com duas lombadas. Neste teste, o motorista foi instruído para dirigir de três maneiras: rapidamente, cuidadosamente e normalmente. Os limiares de aceleração foram fixados em 0,5G para os três eixos. O SEMC-TP transmissor foi instalado no assoalho do veículo, enquanto o SEMC-TP receptor foi instalado em um laptop para o monitoramento em tempo real dos eventos produzidos pelo sistema.

No primeiro percurso, o motorista dirigiu rapidamente, com o objetivo de verificar o funcionamento do sistema e experimentar as máximas amplitudes de aceleração, acima dos limiares pré-estabelecidos. Assim, as rotatórias e lombadas foram tomadas a velocidade relativamente alta, executaram-se manobras laterais e freadas bruscas na reta. Os resultados revelaram os três tipos de eventos, mostrados na Figura 5.2, sendo que as máximas amplitudes de aceleração atingidas foram: 0,906G;



Figura 5.1: Percurso escolhido para o primeiro teste do SEMC-TP com algoritmo DEL-A.

0,859G e 0,945G no eixo longitudinal, transversal e vertical, respectivamente.

No segundo percurso, o motorista dirigiu com precaução, evitando variações bruscas de aceleração. Os resultados mostraram que nenhum evento foi gerado. No terceiro percurso, o motorista dirigiu como de costume, os eventos gerados revelaram os pontos críticos do percurso tais como: lombadas e rotatórias. Neste percurso não foram gerados eventos de aceleração longitudinal, enquanto que as amplitudes máximas de aceleração transversal e vertical foram 0,664G e 1,062G, respectivamente. Os eventos deste percurso são mostrados na Figura 5.3.

### 5.1.2 Segundo teste do DEL-A

Para o segundo teste, o sistema foi instalado no assoalho do ônibus circular da UNICAMP e os limiares de aceleração foram fixados em 0,5G nos três eixos de movimentação e 35°C para a temperatura. Os resultados revelaram um total de 147 eventos em um percurso com duração de 22 minutos, sendo que a maior parte dos eventos indentificou acelerações no eixo vertical e mudanças de temperatura acima dos limiares. A amplitude máxima dos eventos de aceleração no eixo vertical foi de 2,127G, enquanto que nos outros eixos não foi ultrapassado o limiar. A temperatura máxima atingida pelos eventos de desconforto térmico foi de 35,5°C. Estes eventos são mostrados na Figura 5.4(a); nesta figura os marcadores amarelos indicam eventos de desconforto térmico, enquanto os marcadores



Figura 5.2: Eventos gerados para o primeiro teste do DEL-A com o motorista dirigindo de maneira rápida.



Figura 5.3: Eventos gerados para o primeiro teste do DEL-A com o motorista dirigindo normalmente.



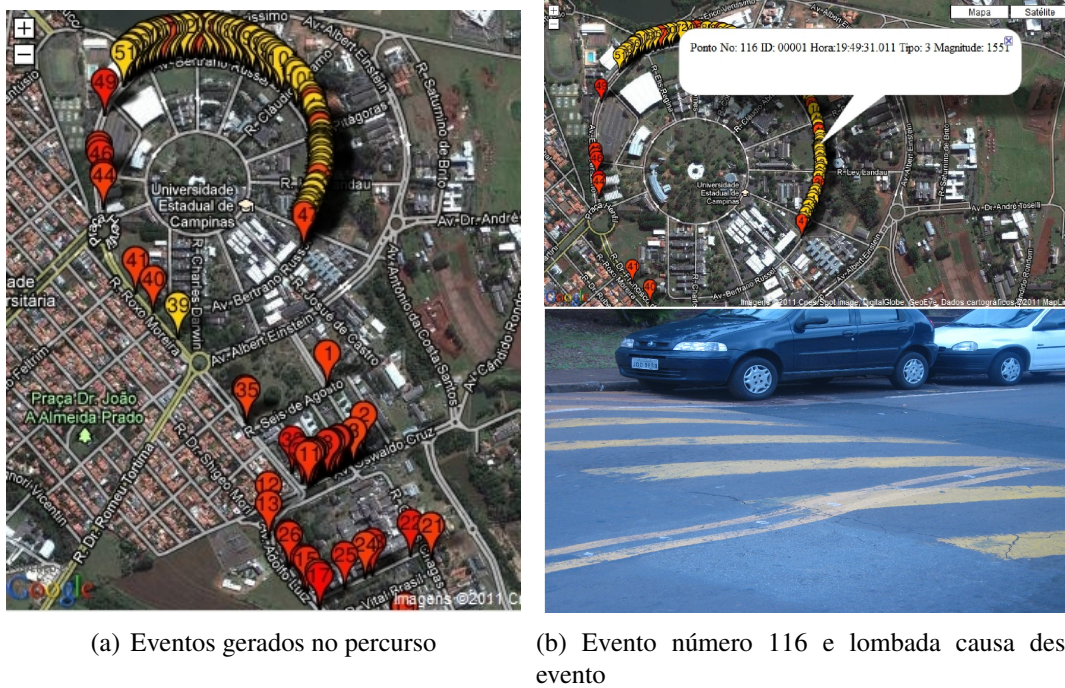


Figura 5.4: Eventos gerados para o segundo teste do DEL-A no ônibus circular da UNICAMP.

vermelhos indicam eventos de aceleração vertical. A Figura 5.4(b) mostra uma lombada, a qual é a fonte de desconforto do evento vertical número 116 com amplitude de 1,55G.

## 5.2 Resultados utilizando o algoritmo DEL-AJ

### 5.2.1 Primeiro teste do DEL-AJ

O SEMC-TP com o algoritmo DEL-AJ foi primeiramente testado em uma viagem de carro no qual se exploraram os máximos valores de *Jerk* longitudinal através de freadas bruscas a uma velocidade moderada. Os limiares de aceleração foram fixados a 0,7G enquanto o limiar do *Jerk* foi fixado em  $0,4\text{m/s}^3$  como sugerido nos capítulos anteriores. O SEMC-TP transmissor foi instalado no assoalho do veículo. Os resultados obtidos revalaram um *Jerk* máximo de  $0,98\text{m/s}^3$ , calculado durante um período de 30 segundos.

### 5.2.2 Segundo teste do DEL-AJ

Neste teste foram efetuados vários testes usando duas linhas do sistema de transporte coletivo da cidade de Campinas. O dispositivo foi instalado no assoalho dos ônibus, na sua parte traseira, com os mesmos limiares do teste anterior. As principais diferenças entre os percursos das duas linhas estão

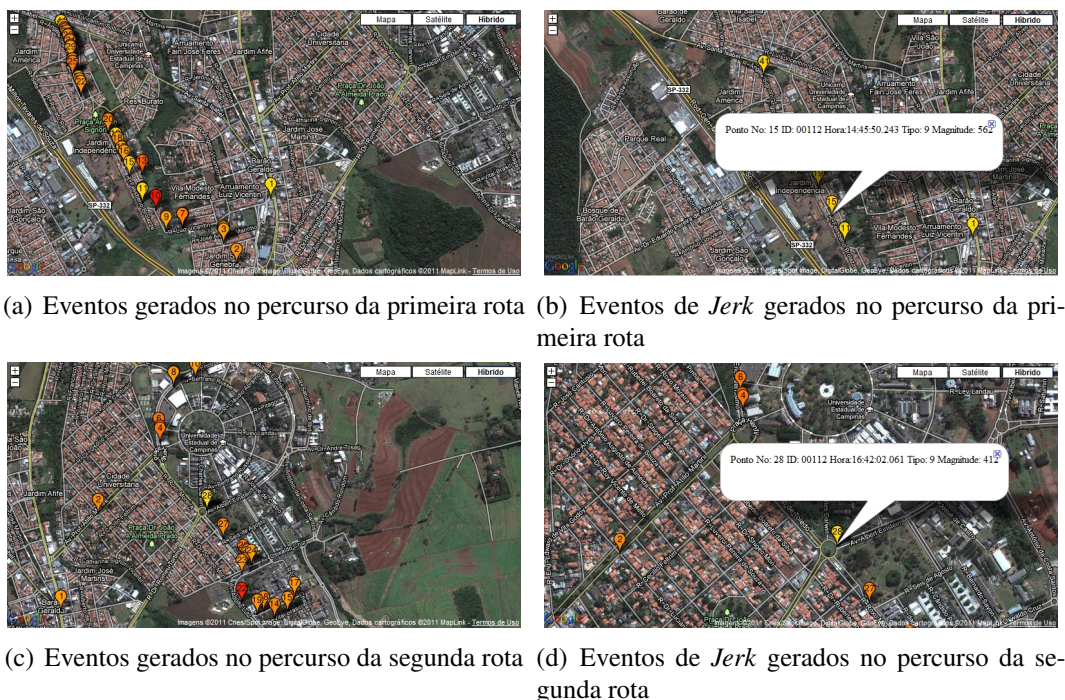


Figura 5.5: Eventos gerados para o teste do DEL-AJ.

no estado de degradação do pavimento e na velocidade média atingida pelos ônibus. Os resultados para a primeira linha, mostrados na Figura 5.5(a), revelaram uma amplitude máxima de 1,86G em eventos verticais de aceleração e cinco eventos de *Jerk* (Figura 5.5(b)), cuja amplitude máxima atingiu  $0,562\text{m/s}^3$ . Enquanto que os resultados para a segunda linha, mostrados na Figura 5.5(c), revelaram uma amplitude máxima de 1,7184G em eventos verticais de aceleração (evento 22) e um evento de *Jerk* de  $0,412\text{m/s}^3$  (evento 28, mostrado na Figura 5.5(d)). Os resultados revelaram que a primeira linha possui maior degradação na via e que o motorista desta dirigiu com maior velocidade média.

## 5.3 Resultados utilizando o algoritmo DEL-AIC

### 5.3.1 Primeiro teste do DEL-AIC

O SEMC-TP com o algoritmo DEL-AIC foi primeiramente testado, em dois veículos diferentes, em uma viagem com a mesma rota e os mesmos limiares de aceleração (0,7G). O primeiro veículo foi um ônibus da moradia da UNICAMP com o SEMC-TP transmissor instalado no assoalho da parte traseira deste. O segundo veículo é um carro com o SEMC-TP transmissor instalado sobre o painel deste. Os resultados do primeiro veículo, mostrados na Figura 5.6, evidenciam uma grande quantidade de eventos, especialmente no eixo vertical devido ao precário estado da via e à velocidade



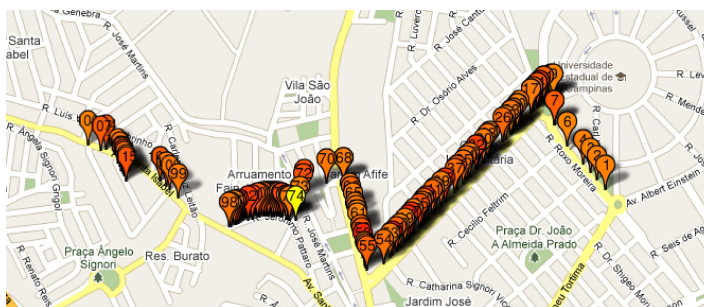


Figura 5.6: Eventos gerados para o primeiro teste do DEL-AIC no ônibus da moradia da UNICAMP.

inapropriada de condução. Neste percurso, o índice de conforto calculado foi 2 da escala mostrada na Tabela 4.2 (evento 74) com uma aceleração RMS ponderada de 0,128G para um intervalo de tempo de 15 minutos. Comentários da viagem, feitos pelos passageiros, indicaram sensações frequentes e intensas de enjoo e vibração, especialmente na parte traseira do veículo.

Os resultados do segundo veículo (um carro) que partiu da UNICAMP em direção à moradia estudantil, mostrados na Figura 5.7, revelaram uma menor quantidade de eventos. Estes eventos foram mais focalizados permitindo reconhecer isoladamente cada fonte do desconforto, por exemplo: o evento 2 foi causado por uma freada forte antes de uma faixa de pedestres, pois este foi um evento de aceleração longitudinal, indicando uma possível falta de atenção por parte do motorista. Os eventos do 5 ao 10 foram causados pela condição precária do pavimento em função da velocidade do carro neste segmento, pois são eventos de aceleração vertical e longitudinal de alta amplitude (máximo 0,921G). O índice de conforto calculado pelo SEMC-TP para esta viagem foi 3 (mostrado pelo evento 11) com uma aceleração RMS ponderada de 0,082G para um intervalo de tempo de 15 minutos.

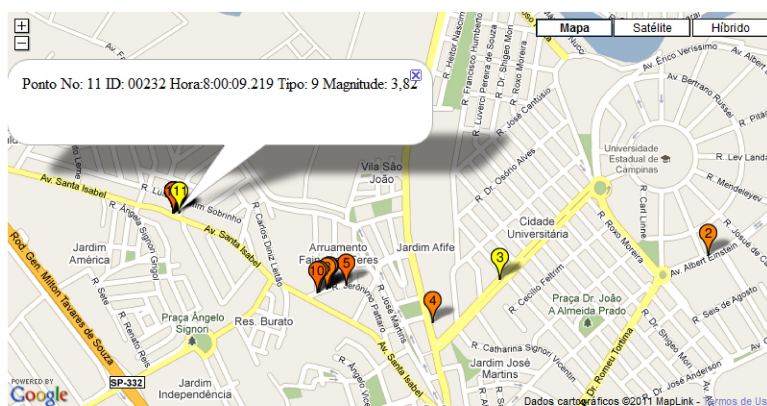


Figura 5.7: Eventos gerados para o primeiro teste do DEL-AIC em um carro.



### 5.3.2 Segundo teste do DEL-AIC

O segundo teste do SEMC-TP com o algoritmo DEL-AIC foi realizado em uma viagem de 33,6km de carro desde a UNICAMP até o aeroporto de Campinas com uma duração de 20 minutos. O motorista dirigiu com pressa atingindo uma aceleração ponderada de 0,158G, como mostrado pelos resultados na Figura 5.8. A maior parte dos eventos foi causada por alta velocidade de condução e às imperfeições do asfalto, pois se deveu a eventos de aceleração verticais ou seus compostos com os outros eixos de aceleração (XZ, YZ ou XYZ). O índice de conforto calculado pelo SEMC-TP foi de 1.

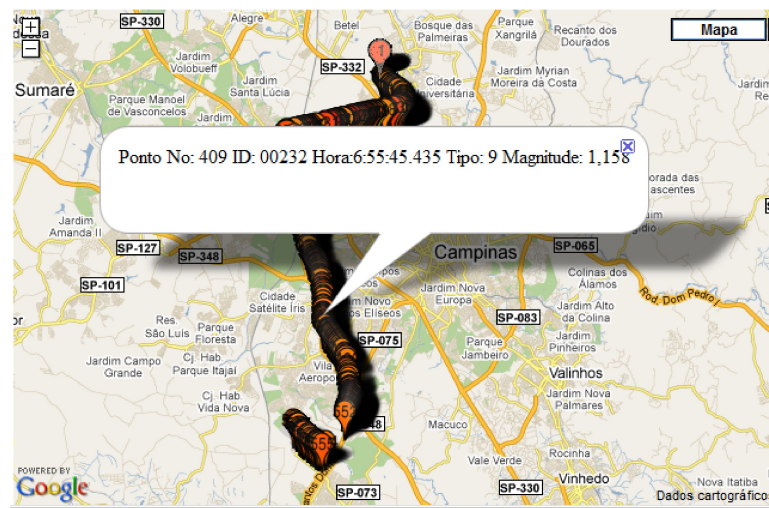


Figura 5.8: Eventos gerados para o segundo teste do DEL-AIC em um carro.

### 5.3.3 Terceiro teste do DEL-AIC

O terceiro teste do SEMC-TP com o algoritmo DEL-AIC foi realizado em uma linha do sistema de transporte coletivo da cidade de Campinas com os limiares de aceleração fixados em 0,7G. Os resultados, mostrados na Figura 5.9, revelaram um índice de conforto mínimo de 3 (evento 19) com uma aceleração RMS ponderada de 0,081G. Neste percurso se perceberam somente eventos de aceleração vertical que em sua maior parte foram causados por lombadas.

## 5.4 Comparação dos resultados

Os resultados obtidos nos resultados anteriores revelaram uma grande quantidade de eventos, cujas amplitudes máximas fornecem informação sobre os valores típicos de aceleração experimentados

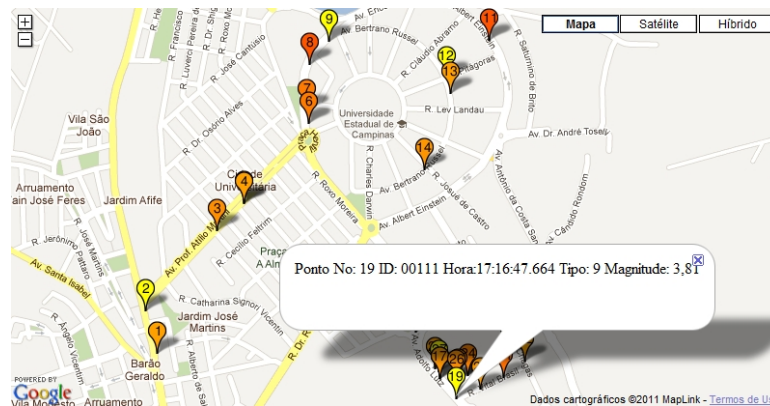


Figura 5.9: Eventos gerados para o terceiro teste do DEL-AIC em um ônibus de transporte coletivo.

pelos passageiros em diferentes veículos e em condições extremas ou cotidianas. A amplitude máxima dos eventos detectados pelo SEMC-TP nos três testes do: DEL-A DEL-AJ e DEL-AIC, são mostrados na Tabela 5.1. Destes resultados, pode-se concluir que os eventos de desconforto causados pelas acelerações verticais são os mais frequentes e que possuem a maior intensidade, pois atingiram um máximo de 2,8G nestes testes; assim os eventos devidos à aceleração vertical são os dominantes no conforto do passageiro.

Tabela 5.1: Amplitude máxima dos eventos detectados pelo SEMC-TP

Evento	Testes DEL-A			Testes DEL-AJ			Testes DEL-AIC			
	I-A	I-B	II	I	II-A	II-B	I-A	I-B	II	III
Aceleração Long. (G)	0,906	-	-	-	-	-	-	0,914	1,023	-
Aceleração Trans. (G)	0,859	0,664	-	-	-	-	0,929	1,218	1,195	0,726
Aceleração Vert. (G)	0,945	1,062	2,127	-	1,86	1,71	1,71	1,078	2,539	1,351
<i>Jerk</i> Long. (m/s <sup>3</sup> )	-	-	-	0,98	0,562	0,412	-	-	-	-
Índice	-	-	-	-	-	-	2	3	1	3
Aceleração RMS (G)	-	-	-	-	-	-	0,128	0,082	0,174	0,081

## 5.5 Medida do consumo de corrente e temperatura

Um outro teste para conhecer o consumo de corrente do nó transmissor foi realizado. Neste teste se mediu a corrente consumida pelo nó, mudando as condições de operação do nó, tais como: todos os periféricos operando ou com o módulo de comunicações e o cartão de memória desligados em alguns períodos de tempo. Os resultados mostraram que o consumo da placa sem o GPS, sem memória SD e sem módulo de comunicações é 81mA, enquanto com o GPS é incrementado a 158mA. Acrescentando o consumo do módulo de comunicações sem transmissão de dados, este foi elevado

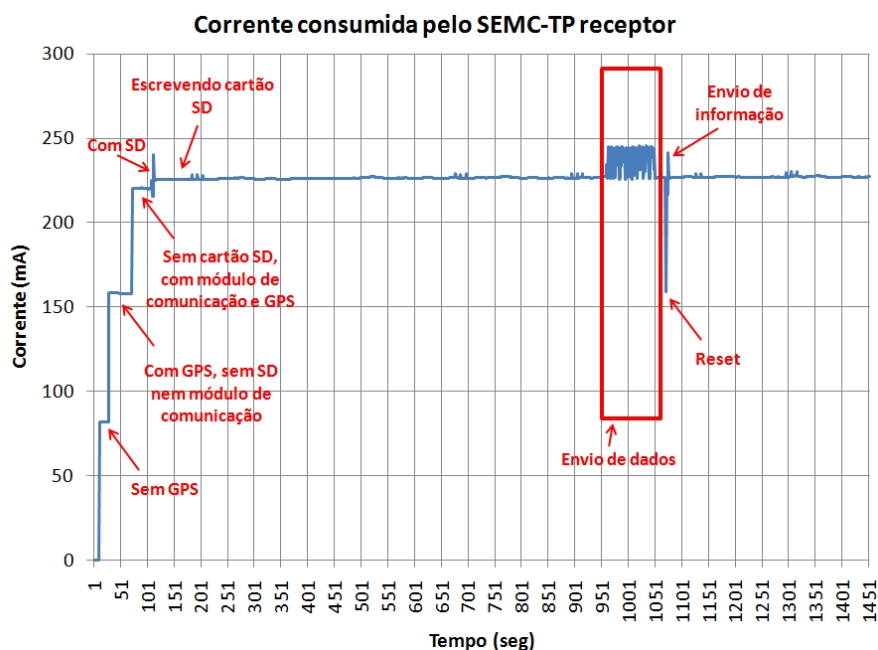


Figura 5.10: Corrente consumida pelo SEMC-TP transmissor.

para 220mA; acrescentando escrita na memória SD atinge 225mA. Também foram enviadas as mensagens de envio de dados (#) e a mensagem de informação (!), aumentando o consumo para 250mA. O gráfico do consumo é mostrado na Figura 5.10. Com todos os periféricos operando se observou um aquecimento significativo do regulador de tensão e da placa do módulo de comunicação.

O consumo de energia da placa foi projetado em 570mA, no entanto o consumo real medido foi de 250mA, devido ao consumo do cartão de memória SD ser muito menor do que o máximo projetado. O consumo total pode ser reduzido até 188mA deixando o módulo de comunicação em modo *sleep*.

Ao mesmo tempo da medição de consumo foi feita a leitura do sensor de temperatura, como mostrado na Figura 5.11. A temperatura ambiente foi 26°C, ao final dos testes com uma duração de 16 minutos, constatou-se uma elevação de 4°C na temperatura medida pelo sensor do protótipo. Estes resultados sugerem que a dissipação dos dispositivos da placa está afetando a medida da temperatura; por isto, recomenda-se colocar o sensor de temperatura externo ao nó transmissor.

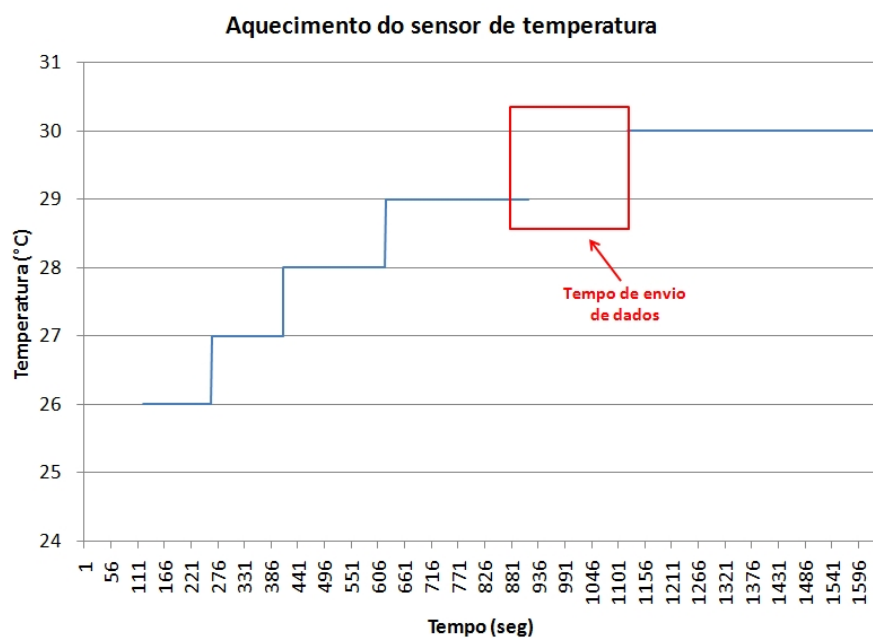


Figura 5.11: Leitura do sensor de temperatura desde a memória SD.

# Capítulo 6

## Conclusões

Este trabalho apresenta o projeto de um sistema embarcado para a avaliação de conforto do usuário do sistema de transporte público. O projeto abrange: uma pesquisa sobre as características dos sinais de aceleração e *Jerk* esperados nos veículos de transporte público, os indicadores utilizados na avaliação do conforto dinâmico em veículos, o desenvolvimento do *Hardware*, do *Firmware* e um instrumento virtual. As conclusões de todos os estágios do projeto e os possíveis trabalhos futuros são apresentados a seguir.

### 6.1 Conclusões sobre as características dos sinais de interesse

Dentro de condições normais de operação ao redor de 60km/h, nos experimentos realizados se obtiveram sinais de aceleração com amplitudes pico a pico máximas de 1,6G para o plano horizontal (acelerações longitudinais e transversais) e 2,85G para o eixo vertical; sendo que os menores níveis de aceleração foram atingidos em veículos como taxis e carros particulares e os maiores níveis pelos veículos do sistema de transporte de ônibus; também se percebeu que as características especiais do sistema de transporte BRT, como pista exclusiva, diminui a amplitude dos sinais de aceleração vertical em torno de 26%. Deve-se considerar também que as características do sinal de aceleração mudam de acordo o modelo e as condições de manutenção dos veículos.

Na análise espectral dos sinais de aceleração dos veículos se constatou que a natureza destes sinais é de baixa frequência, sendo que a maior porção da informação relevante se encontra nos primeiros 50Hz do espectro de frequências. Também foram identificadas três faixas de frequências particulares dentro desta largura de banda. Os eventos nas frequências mais baixas (<5Hz) no plano horizontal são decorrentes de mudanças na direção, arranques, mudanças de marchas, também freadas bruscas e no eixo vertical são ocasionadas por lombadas e imperfeições no pavimento. Os eventos na faixa de frequências entre 15Hz e 30Hz são causadas pela vibração do motor e pela estrutura do veículo,

sendo que esta faixa é influenciada pelo modelo, estado de manutenção ou sistema de amortecimento do veículo.

As maiores amplitudes de aceleração encontradas foram causadas pelas lombadas cujo sinal de aceleração atingiu aproximadamente 3G e teve uma duração de 2 segundo, ou seja, uma frequência de 0,5Hz. Este evento é de alta amplitude somente no eixo vertical, pois nos eixos do plano horizontal, o efeito é reduzido significativamente para velocidades moderadas, o qual permitiu isolar com sucesso a fonte de eventos conforme o seu eixo de ocorrência.

Dado que a amplitude do sinal do *Jerk* é dependente da frequência, seu cálculo depende do tipo de filtragem implementada. Algumas pesquisas anteriores que envolveram a avaliação do conforto, utilizando o *Jerk*, não definiram um padrão para o cálculo deste. Assim, os resultados de algumas pesquisas para os níveis de *Jerk* não podem ser comparados entre si. A ordem do filtro e a frequência de corte deste são fatores importantes que influenciam os resultados do *Jerk* na avaliação do conforto.

## 6.2 Conclusões sobre o *Hardware*

O sistema embarcado implementado permitiu coletar e processar a informação necessária para detectar, armazenar e transmitir os eventos de desconforto em um percurso com um baixo custo. Estes eventos possuem informação sobre a posição, hora, amplitude, eixo de ocorrência e identificação do veículo em teste. Com esta informação é possível detectar vícios na maneira de dirigir do motorista como acelerações excessivas nos eixos transversal, longitudinal ou vertical; também é possível avaliar o estado da via e detectar seus pontos perigosos ou que podem contribuir na deterioração dos veículos.

Os dispositivos utilizados para implementar o *Hardware* forneceram vantagens no desenvolvimento e desempenho do projeto. A memória SD fornece um meio de armazenamento bem conhecido, cujo hardware de leitura e de fácil aquisição sendo utilizado na maior parte dos notebooks. O processador ARM7™ é um dispositivo de programação simples e com uma grande quantidade de módulos periféricos. Além disso, devido à baixa frequência dos sinais envolvidos no projeto, este processador pode realizar o processamento digital de sinais sem necessidade de utilizar um sistema dedicado para esta tarefa, tal como um DSP, o que teria incrementado o custo, a complexidade e o consumo de energia do nó transmissor. O módulo de comunicação Digimesh™ 900 permitiu ampliar o alcance de comunicação dos nós, pois sua frequência de operação é mais baixa do que outros módulos de 2.4GHz; sua habilidade para a regeneração de buracos na rede *Mesh* pode ser explorada em trabalhos futuros. O acelerômetro digital permitiu condicionar o sinal de aceleração sem carregar o microcontrolador com processamento adicional, facilitando processos de filtragem e conversão analógico-digital. Finalmente, o barramento I<sup>2</sup>C utilizado para a comunicação do acelerômetro e o sensor de temperatura com o microcontrolador foi vantajoso devido a sua facilidade para estabelecer

um barramento com vários periféricos quando comparado com o barramento SPI. Na concepção do nó transmissor foi projetada uma porta de conexão ao barramento I<sup>2</sup>C que serve para expandir a quantidade de sensores no nó a ser explorado em trabalhos futuros. Da mesma maneira foi disponibilizada uma entrada do ADC do microcontrolador para uso com sensores de saída analógica.

A implementação do sensor de temperatura na mesma placa do nó transmissor merece um reprojeto, pois o sensor é aquecido pela dissipação dos dispositivos dentro da placa. Assim, recomenda-se colocar o sensor de temperatura fora da placa utilizando a porta de expansão do nó transmissor. Na tabela 6.1 são apresentadas as especificações técnicas mais relevantes do SEMC-TP.

Tabela 6.1: Resumo das especificações técnicas do nó transmissor do SEMC-TP

Processador	LPC2478, ARM7 @ 72MHz
Sensores disponíveis	de temperatura e acelerômetro
Range de aceleração	4G
BW de aceleração	50Hz <sup>(1)</sup> , 100Hz <sup>(2)</sup>
Frequência de amostragem da aceleração	500Hz
Exatidão máxima da posição	5m
Frequência de comunicação sem fio	900MHz
Protocolo de comunicação sem fio	Digimesh
Barramentos disponíveis para conexão de periféricos adicionais	I <sup>2</sup> C
Capacidade de memória	2GB <sup>(3)</sup>
Consumo de energia	250mA
Range de alimentação	5V - 30V
Tamanho	5cm x 9cm

<sup>(1)</sup>, para algoritmos DEL-A e DEL-AJ. <sup>(2)</sup>, para o algoritmo DEL-AIC.

<sup>(3)</sup>, limitado pelo cartão de memória SD.

## 6.3 Conclusões sobre os algoritmos e resultados experimentais

Três algoritmos de detecção de eventos de desconforto foram implementados neste trabalho. Os algoritmos, batizados DEL-A e DEL-AJ, permitem detetar eventos pontuais, enquanto, o algoritmo, batizado DEL-AIC além de detetar eventos pontuais também permite avaliar o conforto médio em um trecho do percurso.

O algoritmo DEL-A é adequado na localização geográfica das fontes de desconforto produzidas por grandes acelerações nos três eixos de movimentação. Desta forma, este algoritmo pode ser utilizado para indicar o estado das vias e modo de condução do motorista. A avaliação do conforto fornecida por este algoritmo depende da quantidade de eventos e de sua amplitude, por isto é difícil

comparar percursos diferentes e estabelecer qual foi o mais confortável. Além disso, os eventos não levam em consideração acelerações rápidas com baixa amplitude, as quais são características em sistemas de transporte com paradas e arranques não programados ou percursos com tráfego pesado que podem ser fonte de desconforto.

A análise dos resultados obtidos com o algoritmo DEL-A permitem identificar espacialmente as principais fontes de desconforto causadas por buracos, superfícies mal pavimentadas, lombadas ou canais de drenagem.

O algoritmo DEL-AJ adiciona outro critério de detecção de eventos de desconforto baseados em acelerações longitudinais rápidas porém com amplitudes baixas; este tipo de evento foi detetado nos testes experimentais, nos quais não se observaram eventos de aceleração longitudinal acima do limiar, mas sim de *Jerk* longitudinal. A máxima amplitude de *Jerk* longitudinal detetada foi de  $1\text{m/s}^3$ . Contudo, o algoritmo DEL-AJ requer uma grande quantidade de memória e alto poder de processamento devido à baixa frequência de corte do filtro implementado, a qual requer um longo tempo de medição. O que influi na baixa exatidão da posição do evento do *Jerk*. Finalmente, o método de medida do *Jerk* é função dos parâmetros do filtro do algoritmo, por isso é necessário padronizar os parâmetros do filtro para definir, através de consultas ao passageiro, o limiar de *Jerk* longitudinal confortável para os usuários do sistema de transporte coletivo. O algoritmo para calcular o *Jerk* teve valores similares aos obtidos nas pesquisas de conforto em outros meios de transporte cuja avaliação de conforto foi realizada mediante consultas diretas em passageiros e onde foram utilizados parâmetros de filtragem similares.

O algoritmo DEL-AIC além de detetar eventos de desconforto pontuais devidos à aceleração, também fornece um índice de conforto que permite fácil comparação com outros motoristas, e entre ônibus ou trechos do percurso distintos. A utilização do índice de conforto concede ao algoritmo a habilidade de qualificar e quantificar o conforto no percurso e apresenta uma faixa de localização para este conceito; no entanto, sem a utilização da detecção por limiar de aceleração (DEL-A), é impossível conhecer o local exato da ocorrência de um evento para tomar uma ação corretiva efetiva. As faixas de aceleração RMS ponderadas, recomendadas pela norma ISO2631-1 estão de acordo aos valores medidos utilizando o DEL-AIC e os eventos de desconforto se correlacionam com os apontados pelos passageiros. As experimentações mostraram que o índice de conforto obtido no transporte coletivo é menor do que o obtido no transporte particular sob condições cotidianas e para um mesmo percurso. Também foi possível detetar pontos críticos nas vias do percurso. Contudo, o índice de conforto 5 somente pôde ser atingido com o sistema em repouso, o que significa que o DEL-AIC é um algoritmo de avaliação muito exigente.

O instrumento virtual do SEMC-TP permite visualizar a distribuição espacial dos eventos de desconforto através de um mapa online; também permite configurar e ler os nós transmissores de



cada veículo. A utilização de um código de cores para os marcadores de localização dos eventos, permite fornecer informação da intensidade do evento de maneira rápida; enquanto que a separação por tipo de evento facilita a identificação da causa deste. No entanto, é necessário adequar a interface às necessidades específicas da prefeitura ou das companhias de transporte.

## 6.4 Sugestões para trabalhos futuros

Algumas sugestões para trabalhos futuros são indicadas a seguir:

- Realizar pesquisas dos limiares de conforto de aceleração e *Jerk*, para o sistema de transporte coletivo no Brasil, correlacionando os eventos apontados pelos usuários de transporte público com os apontados pelo SEMC-TP.
- Realizar pesquisas dos limiares de conforto de aceleração e *Jerk* utilizando o SEMC-TP e pesquisas diretas, para o sistema de transporte de passageiros com necessidades especiais, gestantes, idosos, dentre outros.
- Implementação de um sistema para monitorar não somente sinais de aceleração senão também qualquer outro parâmetro de interesse para um ambiente urbano (por exemplo, poluição) baseado na rede de transportes como plataforma de rede. Bastando para isso instalar vários tipos de sensores nos nós transmissores do SEMC-TP aproveitando as vantagens do barramento I<sup>2</sup>C e a cobertura de buracos dos módulos de comunicação.
- Desenvolver um software dedicado às prefeituras, para a localização de pontos críticos no pavimento, através da identificação de eventos verticais recorrentes.
- Integrar o módulo GPS e o módulo de comunicação na placa do nó transmissor, com o fim de diminuir custo e consumo.
- O *firmware* do SEMC-TP pode ser depurado para se ajustar à operação no terminal de ônibus.



# Referências Bibliográficas

- [1] ISO 2631-1:1997, Mechanical vibration and shock - Evaluation of human exposure to whole-body vibration - Part 1: General requirements.
- [2] Johan Förstberg. *Ride comfort and motion sickness in tilting trains*. PhD thesis, KTH, Vehicle Engineering, 2000.
- [3] Prefeitura Municipal de Curitiba. Portal da Prefeitura de Curitiba.
- [4] S. Boonmee and P. Tangamchit. Portable reckless driving detection system. In *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2009. ECTI-CON 2009. 6th International Conference on*, volume 01, pages 412 –415, may 2009.
- [5] E. Andersson and N. Nilstam. DEVELOPMENT OF ADVANCED HIGH SPEED VEHICLES IN SWEDEN. *Proceedings of the Institution of Mechanical Engineers, Part D: Transport Engineering*, 198(15):229–237, 1984.
- [6] ANALOG DEVICES. *Digital Accelerometer. ADXL345*, 2010. Datasheet.
- [7] Texas Instrument. *TMP100. Digital Temperature Sensor with I<sup>2</sup>C™ Interface*, 2007. Datasheet.
- [8] SanDisk Corporation. SanDisk Secure Digital Card, December 2003. Product Manual. Version 1.9.
- [9] Freescale Semiconductor. *Digital Output Accelerometer - MMA7455L*, 2008.
- [10] H. D. Thomas and J. D. Anderson. Vibration tests on transit buses. In *Proc. 29th IEEE Vehicular Technology Conf*, volume 29, pages 387–391, 1979.
- [11] ME COMPONENTES. *ME-1000RW 65 channels with ultra-high sensitive. Smart GPS Antenna module.*, 2009. Datasheet.

- [12] Yull Heilordt HENAO ROA. Desenvolvimento de um sistema microcontrolado para monitoramento de atividades esportivas baseado em sensores microeletromecânicos de aceleração e giro. Master's thesis, Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação., 2009.
- [13] Digi International. Wireless Mesh Networking. ZigBee® vs. Digimesh™, 2008. White Paper.
- [14] A. Kirk and R. Grant. ECBOS. Enhanced Coach and Bus Occupant Safety. Task 1.1 Report Overview. Technical report, Technical University Graz, 2000.
- [15] P. Halpern, M. I. Siebzeher, D. Aladgem, P. Sorkine, and R. Bechar. Non-collision injuries in public buses: a national survey of a neglected problem. *Emerg Med J*, 22(2):108–110, Feb 2005.
- [16] B. De Graaf and W. Van Weperen. The retention of balance: An exploratory study into the limits of acceleration the human body can withstand without losing equilibrium. *Human factors*, 39(1):111–118, 1997. Cited By (since 1996): 8.
- [17] 3TU.Federation. 3TU MSc in Embedded Systems.
- [18] Philip Gibbs. What is the term used for the third derivative of position?, September 1996.
- [19] L. L. Hoberock. A Survey of Longitudinal Acceleration Comfort Studies in Ground Transportation Vehicles. *Journal of Dynamic Systems, Measurement, and Control*, 99(2):76–84, 1977.
- [20] Federal Transit Administration (FTA) and National Bus Rapid Transit Institute (NBRTI). Characteristics of Bus Rapid Transit for Decision-Making. Technical report, U.S. Department of Transportation, University of South Florida., February 2009.
- [21] P. Kelly. Quality bus transit systems. In *Proc. Int Public Transport Electronic Systems Conf. (Conf. Publ. No. 425)*, pages 11–15, 1996.
- [22] O. F. LIMA JR. *Qualidade em serviços de transportes: conceituação e procedimento para diagnóstico*. PhD thesis, Escola Politécnica, Universidade de São Paulo, 1995.
- [23] Maurício Olbrick Rodrigues. Avaliação da qualidade do transporte coletivo da cidade de São Carlos. Dissertação de mestrado, Universidade de São Paulo, 2006.
- [24] Miller R.E., Boman P., Walden J., Rhoades S., and Gibbs R. Acceleration and GPS Data Monitor Truck-Haulage Jolts. Technical report, NIOSH. National Institute for Occupational Safety and Health, 2000.

- [25] Kasun De Zoysa, Chamath Keppitiyagama, Gihan P. Seneviratne, and W. W. A. T. Shihan. A public transport system based sensor network for road surface condition monitoring. In *Proceedings of the 2007 workshop on Networked systems for developing regions*, NSDR '07, pages 9:1–9:6, New York, NY, USA, 2007. ACM.
- [26] G. Reveriego, C. Urdiales, and F. Sandoval. Vehicle risks assessment using a nokia N95. In *Proc. 12th Int. IEEE Conf. Intelligent Transportation Systems ITSC '09*, pages 1–6, 2009.
- [27] Kimberly Tuck. Implementing Auto-Zero calibration Technique for Accelerometers. AN3447, March 2007. Freescale Semiconductor, Inc., Application Note.
- [28] Alan V. Oppenheim, Ronald W. Schaffer, and John R. Buck. *Discrete-time signal processing (2nd ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1999.
- [29] Román Pérez Enríquez. Wavelets (Ondeletas).
- [30] SENSR. *Practical Guide to Accelerometers*, 2011.
- [31] Antonio Augusto de Paula Xavier. *Predição de conforto térmico em ambientes internos com atividades sedentárias - Teoria física aliada a estudos de campo*. PhD thesis, Universidade Federal de Santa Catarina, 2000.
- [32] A. El-Rabbany. *Introduction to GPS: the Global Positioning System*. Artech House mobile communications series. Artech House, 2002.
- [33] R. B. Langley. *The mathematics of GPS*, volume 2. GPS World, July/August 1991.
- [34] Companhia do Metropolitano de São Paulo - Metrô. Serviços - Terminal Rodoviário Governador Carvalho Pinto (Tietê), 2010.
- [35] Bhavneet Sindhu, Hardeep Singh, and Amit Chhabra. Emerging wireless standards - WiFi, Zig-Bee and WiMAX. In *World Academy of Science, Engineering and Technology*, 2007.
- [36] Digi International Inc. XBee-PRO®900/DigiMesh™ 900 RF Modules. <http://www.digi.com>, 2009. Datasheet.
- [37] Technical Committee SD Card Association. SD Specifications Part 1 Physical Layer Simplified Specification, May 2010.
- [38] A.K. Singh and S.A. Kumar. *Microcontroller and Embedded System*. New Age International (P) Ltd., 2008.

- [39] ARM Ltda. ARM site. Website, August 2011.
- [40] NXP Semiconductors. LPC2478. Product data sheet, September 2010.
- [41] Google inc. Google Maps JavaScript API. ONLINE, Agosto 2010.  
<http://code.google.com/apis/maps/documentation/javascript/basics.html>.
- [42] EBTU. Planejamento da operação. Elementos intervenientes. vol 2, 1986.
- [43] S. CHATTOPADHYAY. *EMBEDDED SYSTEM DESIGN*. Prentice-Hall Of India Pvt. Ltd., 2010.
- [44] Frank Vahid and T. Givargis. *Embedded system design: a unified hardware/software introduction*. Wiley, 2002.

# Apêndice A

## Características desejadas no sistema de transporte público.

A seguir são descritas as características relevantes na avaliação do serviço de um sistema de transporte público:

- **Confiabilidade:** Alta confiabilidade provê segurança sobre o desempenho do sistema e permite divulgar os tempos de operação com maior exatidão. Além disso, as pessoas tomam decisões tais como onde trabalhar ou morar em função do sistema de transporte, assim a estabilidade do serviço é vital para essa tomada de decisões.
- **Frequência:** Alta frequência nas linhas de ônibus reduz o tempo de espera.
- **Informação:** Os usuários precisam saber aonde ir e como se deslocar para fazer uso do serviço.
- **Velocidade:** Do ponto de vista do usuário, é desejável que o ônibus seja mais rápido que o automóvel, assim uma velocidade média alta é importante.
- **Acessibilidade:** Os percursos de acesso devem ser curtos e de fácil acessibilidade para os usuários do sistema.
- **Lotação:** O numero de passageiros no interior do veículo, [23], talvez seja o fator mais crítico do sistema de transporte público, pois demonstra a incapacidade do sistema para atender a demanda. A proximidade entre usuários e a limitação dos movimentos converte-se na principal causa de perda de usuários no sistema de transporte coletivo.
- **Características dos veículos:** As características dos ônibus que influenciam no conforto são: temperatura, ventilação, ruído, aceleração e desaceleração, altura dos degraus, largura das por-

tas, disposição e material dos assentos, dentre outras [42]. Assim, degraus altos causam desconforto em pessoas baixas, idosos, enfermos ou em crianças; também poucas portas de acesso ou muito estreitas dificultam a circulação das pessoas dentro do veículo dando uma percepção incorreta de superlotação.

- Comportamento dos condutores: No motorista deve-se observar o comportamento na direção (dirigibilidade nas curvas, formas de frear e arrancar e respeito às normas de trânsito) assim como, o cumprimento das paradas e o atendimento aos usuários especialmente aqueles com dificuldades de locomoção ou necessidades especiais [23].
- Estado das vias: A qualidade das vias por onde passam os ônibus têm grande importância afim de evitar os solavancos provocados por buracos, lombadas e valetas [23]. O estado das vias causa desconforto, danificam o veículo, ferem o usuário e são possíveis causas de acidentes.



## Apêndice B

### Desenho do filtro FIR com $F_C$ de 0,3Hz e decimação.

O diagrama de blocos mostrado na Figura B.1, feito na plataforma LabVIEW™ permite o projeto de um filtro FIR com frequência de corte de 0,3Hz. Lembre-se que o filtro inclui um processo de decimação, assim o sinal de entrada possui uma frequência de amostragem ( $F_S$ ) de 500Hz, enquanto o sinal de saída possui  $F_S$  de 10Hz.

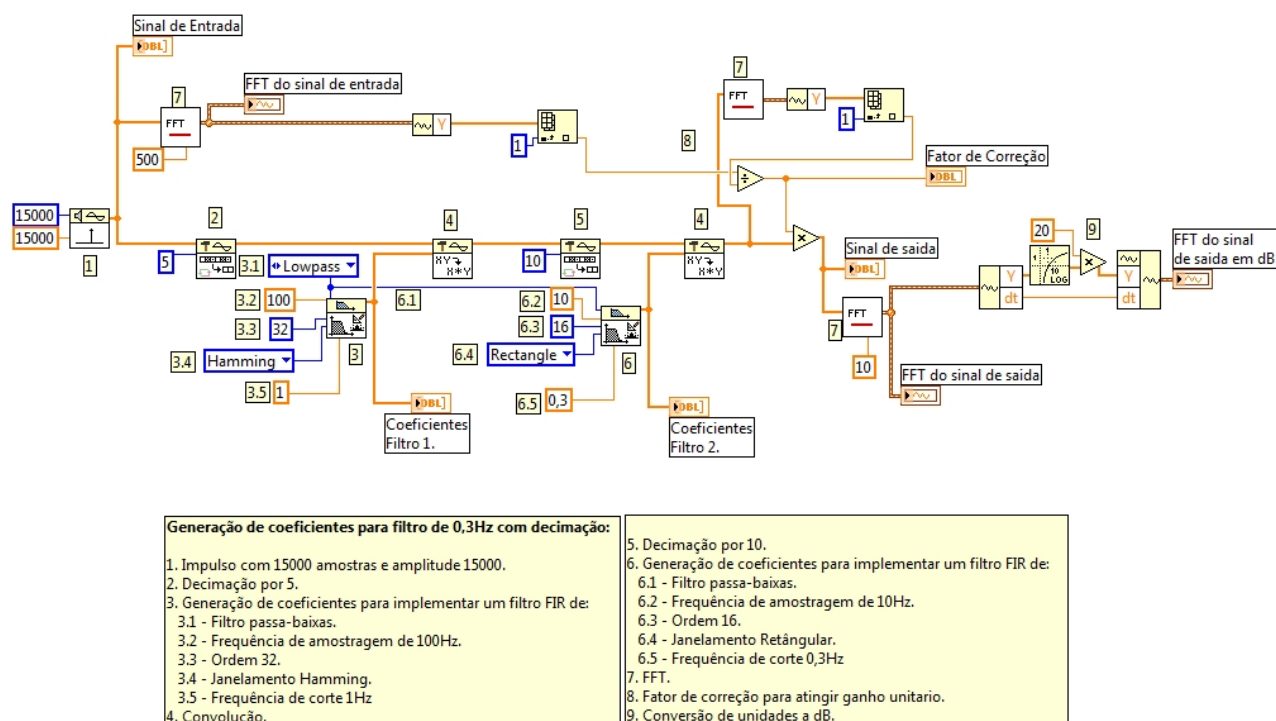


Figura B.1: Diagrama de blocos na plataforma LabVIEW para o projeto do filtro utilizado no calculo do *Jerk*.

Os coeficientes calculados para o primeiro ( $h_1$ ) e segundo filtro ( $h_2$ ) são mostrados a seguir:

$$\begin{aligned} h_1[n] = & 0,001359; 0,001551; 0,002074; 0,002925; 0,004083; 0,005512; 0,007161; 0,008966; 0,010854; \\ & 0,012745; 0,014558; 0,016213; 0,017638; 0,018767; 0,019549; 0,019950; 0,019950; 0,019549; \\ & 0,018767; 0,017638; 0,016213; 0,014558; 0,012745; 0,010854; 0,008966; 0,007161; 0,005512; \\ & 0,004083; 0,002925; 0,002074; 0,001551; 0,001359 \end{aligned}$$

$$\begin{aligned} h_2[n] = & 0,041919; 0,046076; 0,049815; 0,053060; 0,055741; 0,057804; 0,059204; 0,059911; 0,059911; \\ & 0,059204; 0,057804; 0,055741; 0,053060; 0,049815; 0,046076; 0,041919 \end{aligned}$$

# Apêndice C

## Protocolo NMEA.

O protocolo NMEA 0183 da *National Marine Electronics Association*’s com interface ASCII está baseado em sete tipos de mensagem, os quais contêm um conjunto de dados diferentes [11]. A seguir são listados os sete tipos de mensagem:

- GGA - *Global positioning system fix data*
- GLL - *Latitude and longitude, with time of position fix and status*
- GSA - *GPS DOP and actives Satellites*
- GSV - *GPS Satellite in view*
- RMC - *Recommended minimum specific GPS/Transit Data*
- VTG - *Course over ground and ground speed*
- ZDA - *Time and date*

O módulo GPS, ME-1000RW, utiliza somente quatro destes tipos: GGA, GSA, GSV e RMC. Neste trabalho foram utilizados o GGA e RMC, pois, diferentemente dos GSA e GSV, estes tipos fornecem os dados de latitude e longitude, descritos em detalhe a seguir.

Note-se que cada mensagem NMEA começa pelo símbolo '\$' e termina pela sequência de comandos composta pelo retorno de carro (*carriage return*) e nova linha (*line feed*), <CR> e <LF> respectivamente.



## C.2 RMC - Especificações GPS mínimas recomendadas e dados de trânsito.

Nesta mensagem o receptor fornece informação sobre tempo, data, posição, curso e velocidade. A estrutura da mensagem NMEA RMC é apresentada a seguir, cujos campos são detalhados na Tabela C.2.

```
$GPRMC,hhmmss.sss,A,ddmm.mmmm,a,dddmm.mmmm,a,x.x,x.x,ddmmyy,x.x,a,a*hh<CR><LF>
```

### Exemplo:

```
$GPRMC,092204.999,A,4250.5589,S,14718.5084,E,0.00,89.68,211200,,A*25<CR><LF>
```

Campo	Nome	Exemplo	Descrição
1	Tempo UTC	092204.999	UTC no formato hhmmss.sss, (000000,000 - 235959,999)
2	Estado	A	Estado: 'V' = Advertência de navegação 'A' = Dados validos
3	Latitude	4250.5589	Latitude no formato ddmm.mmmm.
4	Indicador N/S	S	Indicador do Hemisfério da latitude, 'N' = Norte, 'S' = Sul.
5	Longitude	14718.5084	Longitude no formato dddmm.mmmm.
6	Indicador E/W	E	Indicador do hemisfério da longitude, 'E' = Leste, 'W' = Oeste.
7	Velocidade	000.0	Velocidade no solo em knots (000.0 - 999.9)Indicador do estado do GPS
8	Curso	000.0	Curso no solo em graus (000.0 - 359.9).
9	Data UTC	211200	Data UTC do valor fixado no formato ddmmyy.
10	Varição magnética		Varição magnética em graus (000.0 - 180)
11	Varição magnética		Em Direção 'E' = Leste; 'W' = Oeste
12	Indicador de modo	A	'N' = Dados não válidos 'A' = Modo autônomo 'D' = Modo Diferencial 'E' = Modo Estimado (dead reckoning DR) 'M' = Modo de entrada manual 'S' = Modo de Simulação
13	Checksum	25	

Tabela C.2: Campos da mensagem NMEA tipo RMC [11]



# Apêndice D

## Interfaces de comunicação serial

Existem dois tipos principais de comunicação serial conforme se utiliza ou não de um sinal de *clock* para sincronizar a comunicação entre um dispositivo mestre e escravo. Quando não se envia um sinal de *clock*, a comunicação é denominada assíncrona, entre as quais estão o protocolo UART (*universal asynchronous receiver/transmitter*). Quando se envia um sinal de *clock*, a comunicação é denominada síncrona, entre as quais estão as interfaces SPI, I<sup>2</sup>C, MCI, USB e Firewire; sendo que as duas últimas codificam o *clock* com os dados. A seguir são apresentadas sucintamente as interfaces seriais utilizadas neste trabalho.

### D.1 *Serial Peripheral Interface* (SPI)

O *Serial Peripheral Interface* (SPI) é um protocolo simples, de baixo custo, desenvolvido pela Motorola e utilizado principalmente para comunicação serial entre um dispositivo mestre (microcontrolador) e um ou vários dispositivos escravos (periférico) (Figura D.1). Esta interface permite conectar variados dispositivos tais como: memórias, conversores AD e DA, circuitos *driver* de LCD, sensores, outro microprocessadores dentre outros. A comunicação é *full duplex*, ou seja, a comunicação entre o escravo-mestre e mestre-escravo é possível ao mesmo tempo.

A interface SPI, também conhecida como interface de 4 linhas, possui duas linhas de controle e duas linhas de dados: CS, SPC, SDI e SDO. A linha CS, também conhecida como *Chip Select*, habilita o dispositivo escravo para a comunicação com um nível baixo nesta; esta linha é sempre controlada pelo mestre. A linha SPC contém o *clock* serial, o qual também é sempre controlado pelo mestre. As linhas SDI e SDO são as linhas de dados de entrada e saída, respectivamente [9] [43]. Também existe uma variação da interface SPI que possui três linhas, onde a linha SDO não é utilizada.

Entre as principais vantagens da interface SPI se destacam: sua simplicidade na implementação, baixo custo, alta taxa de transmissão de dados quando comparada à interface I<sup>2</sup>C. A principal des-

vantagem é a necessidade de utilizar um pino CS para cada dispositivo escravo, o qual incrementa o número de linhas de acordo ao número de dispositivos no barramento. A seguir são descritas as operações de leitura e escrita utilizando desta interface.

### D.1.1 Operação de Leitura

A operação de leitura de um registro consiste na transferência, através da linha SDI, de um bit de leitura/escrita (sendo '0' para leitura), e sete bits indicando o endereço do registro, em muitos casos somente são utilizados seis bits e o bit restante é utilizado para controlar outro aspecto da leitura, tal como: transmissão de múltiplos bits (MB) como no acelerômetro ADXL345. A resposta do escravo é enviada pela linha SDO com a informação armazenada no registro especificado pelo mestre. A Figura D.2 mostra o diagrama de tempo para a operação de leitura do SPI com 4 linhas no acelerômetro ADXL345.

### D.1.2 Operação de Escrita

A operação de escrita de um registro consiste na transferência, através da linha SDI, de um bit de leitura/escrita (sendo '1' para escrita), sete bits com a informação do endereço do registro e oito bits com a informação que se deseja escrever no registro apontado. Na operação de escrita não é utilizada a linha SDO. A Figura D.3 mostra o diagrama de tempo para a operação de escrita do SPI com 4 linhas.

## D.2 Interface *Inter-Integrated Circuit* (I<sup>2</sup>C)

A interface *Inter-Integrated Circuit* (I<sup>2</sup>C), desenvolvida pela Philips, provê comunicação serial síncrona entre um dispositivo mestre e um ou vários dispositivos escravos de forma barata e efetiva. A

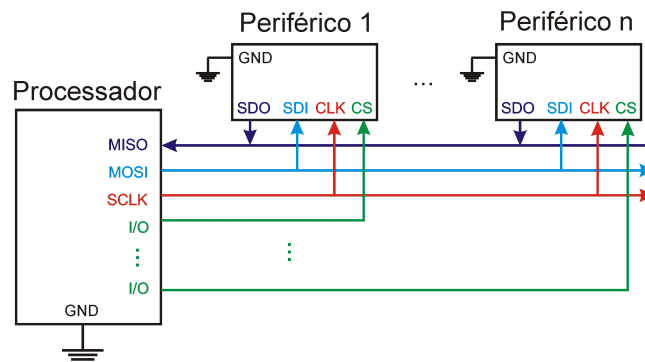


Figura D.1: Interface SPI com 4 linhas.



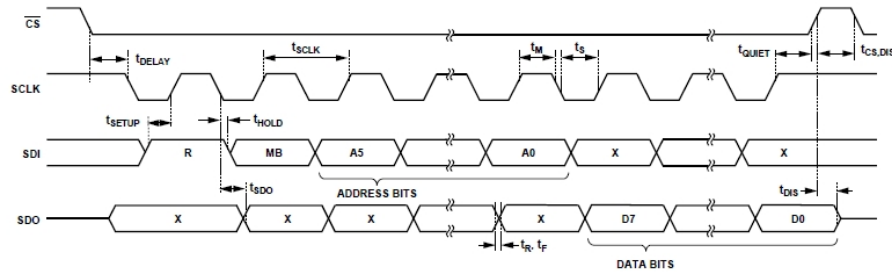


Figura D.2: Diagrama de tempo para a operação de leitura do SPI com 4 linhas [6].

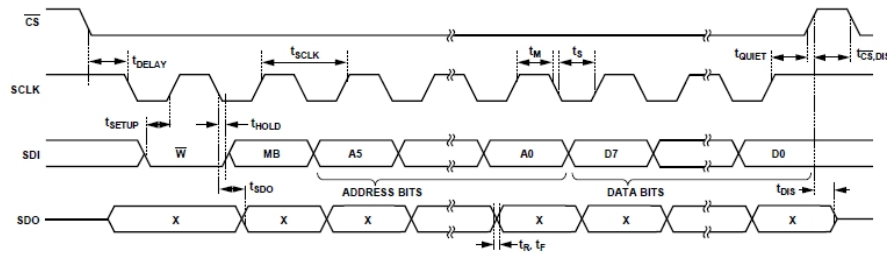


Figura D.3: Diagrama de tempo para a operação de escrita do SPI com 4 linhas [6].

interface I<sup>2</sup>C utiliza somente duas linhas para a interconexão dos dispositivos, sendo que o barramento criado é bidirecional, utiliza configuração de dreno aberto (*open-drain*), é de baixa velocidade e sincronizado com um *clock* comum. A linha SDA ou *Serial Data* transmite os comandos e dados da comunicação, enquanto a linha SCL ou *Serial clock* transmite o *clock* do mestre aos escravos [9] [43]. Em um barramento com vários escravos, estes recebem um endereço (endereço do escravo), os quais são utilizados para escrita ou leitura dos registros de determinado dispositivo escravo no barramento (Figura D.4).

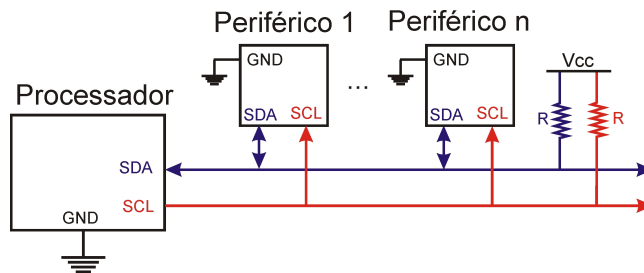


Figura D.4: Interface *Inter-Integrated Circuit* (I<sup>2</sup>C).

O barramento I<sup>2</sup>C em estado ocioso fixa ambas as linhas, SCL e SDA, no nível alto. Uma condição de início da transmissão de dados é indicada pela sequência do SDA mudando de alto para baixo seguido pela mudança do SDA para um nível baixo. A borda de subida da linha SCL é utilizada para amostrar o dado do SDA. Uma condição de final da transmissão de dados é indicada pela linha SCL

retornando para um nível alto seguida pela linha SDA retornando para nível alto. Na Figura D.5 são mostradas as condições de início e final da transmissão de dados.

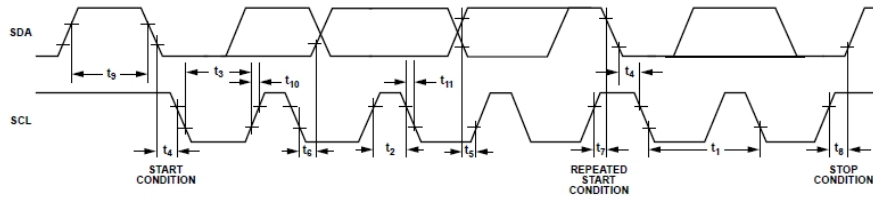


Figura D.5: Diagrama de tempo para as condições de início e fim de transmissão com a interface I<sup>2</sup>C [6].

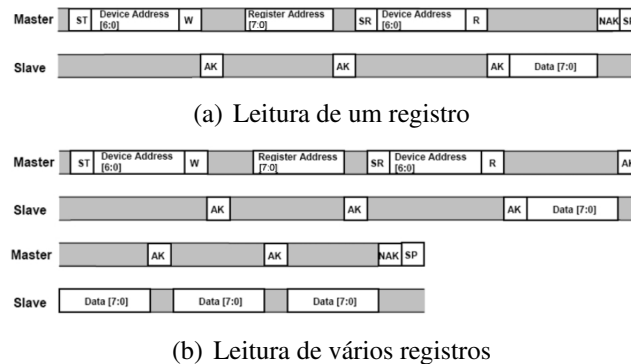
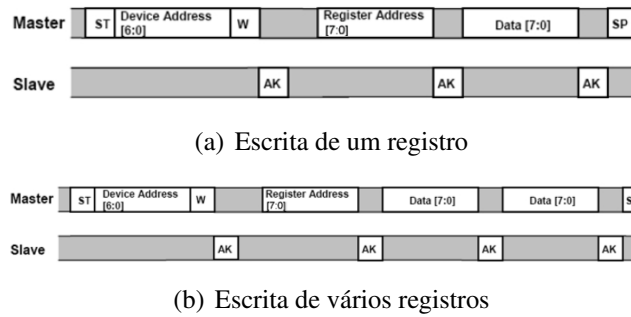
Entre as principais vantagens da interface I<sup>2</sup>C está na utilização de somente duas linhas e a utilização de múltiplos dispositivos mestres. A principal desvantagem é a baixa velocidade de transmissão de dados quando comparada com a interface SPI. A seguir são descritas as operações de leitura e escrita utilizando a interface I<sup>2</sup>C.

### D.2.1 Operação de leitura

A operação de leitura começa com a condição de início (ST) explanada anteriormente e seguida pelos 7 bits do endereço do escravo e um bit de escrita ( $W = '0'$ ), logo se espera pelo reconhecimento por parte do escravo (AK). O reconhecimento ou *Acknowledge* corresponde ao nono bit em nível alto da linha SDA, o qual é enviado pelo dispositivo receptor da mensagem; caso o nono bit seja '0' isto indica um não reconhecimento do comando ou *Not Acknowledge* (NAK). Logo recebido o reconhecimento são transmitidos os oito bits do endereço do registro que se deseja escrever, se espera o reconhecimento de novo. Depois disto, é enviado de novo a condição de reinício (SR), o endereço do dispositivo escravo mais o bit de leitura ( $R = '1'$ ). O dispositivo escravo responderá com o envio do reconhecimento e os oito bits do dado armazenados no registro desejado. Finalmente, é enviado um não reconhecimento (NAK) e uma condição de parada (SP) para indicar o final da leitura. Esta sequência de leitura de registros pode ser visualizada na Figura D.6.

### D.2.2 Operação de escrita

A operação de escrita começa da mesma maneira do que a operação de leitura. Depois do segundo reconhecimento, envia-se o dado que se deseja armazenar no registro indicado e uma condição de parada para finalizar esta operação. Para o caso de múltipla escrita, após o recebimento do sinal de reconhecimento, pode-se enviar o dado que se deseja armazenar no registro seguinte. A sequência para escrita de registros pode ser visualizada na Figura D.7.

Figura D.6: Diagrama de tempo para a operação de leitura na interface I<sup>2</sup>C [9].Figura D.7: Diagrama de tempo para a operação de escrita na interface I<sup>2</sup>C [9].

### D.3 Interface *universal asynchronous receiver/transmitter* (UART)

A interface *universal asynchronous receiver/transmitter* (UART) é a interface serial mais popular. A UART é uma interface serial assíncrona *Full Duplex* pois não precisa da transmissão de um *clock* e as mensagens podem ser transmitidas e recebidas ao mesmo tempo. A versão simples da UART possui somente duas linhas: a linha de recepção (Rx) e a linha de transmissão (Tx), como mostrado na Figura D.8. As principais vantagens do UART são a melhora da imunidade de ruído e a utilização de somente duas linhas; sua principal desvantagem é a baixa velocidade utilizada na transmissão de dados quando comparada com comunicações síncronas.

Na UART, a linha Rx é constantemente monitorada na procura de um bit de começo de transmissão. O bit de começo é sinalizado por uma transição de alto para baixo na linha Rx. Após a detecção do bit de começo, o receptor começa a amostragem da linha Rx para determinar os bits da mensagem. Uma mensagem da UART geralmente contém oito bits, no entanto também são possíveis transmissões com 9 e 10 bits. Além disso, se a UART é configurada para detetar paridade, o comprimento incrementará em um bit. A paridade é utilizada na verificação de erros de transmissão e pode ser configurada como par ou ímpar [44]. Finalmente, a linha é deixada em nível alto para esperar a

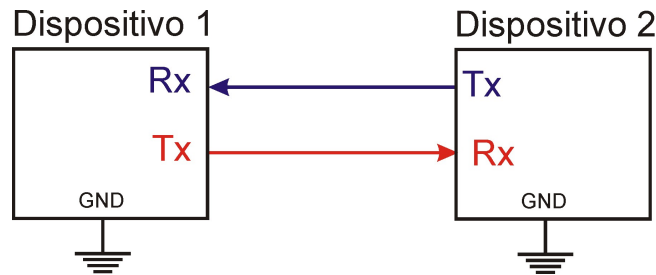


Figura D.8: Interface UART.

transmissão seguinte. A mensagem da UART mostrada na Figura D.9 pertence a uma comunicação serial de oito bits sem paridade.

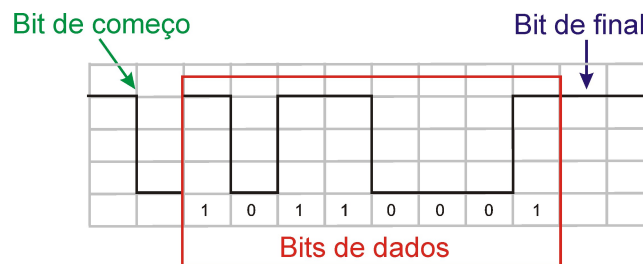


Figura D.9: Mensagem da interface UART.

A transmissão da UART começa com um bit de começo pela linha Tx, depois é enviada a sequência de dados de oito bits do mais significativo ao menos significativo a uma taxa de dados predefinida, chamada *baud rate*. Caso seja definido um bit de paridade este também é transmitido. Finalmente, a linha é deixada em nível alto para esperar a transmissão seguinte [44].

A taxa de transmissão de dados ou *baud rate* deve ser igual para todos os dispositivos da comunicação. As taxas mais comuns são: 2400, 4800, 9600 e 19200 bits/seg. As linhas Tx e Rx podem possuir diversos formatos de tensões, entre os mais comuns estão: TTL, LVTTL ou RS232. No formato TTL a tensão do nível alto estão entre 2V e 5V, enquanto do nível baixo estão entre 0V e 0,8V. O formato LVTTL possui as mesmas faixas do TTL, mas este é utilizado em aplicações com alimentação de 3,3V. No formato RS232 a tensão de nível alto é sinalizada entre -5V e -15V, enquanto o nível baixo está entre 5V e 15V.

## D.4 Secure Digital and Multimedia Card Interface (MCI)

O *Secure Digital and Multimedia Card Interface* ou MCI é uma interface entre o barramento de periféricos (APB) do microcontrolador e o cartão de memória SD. A MCI possui um adaptador,

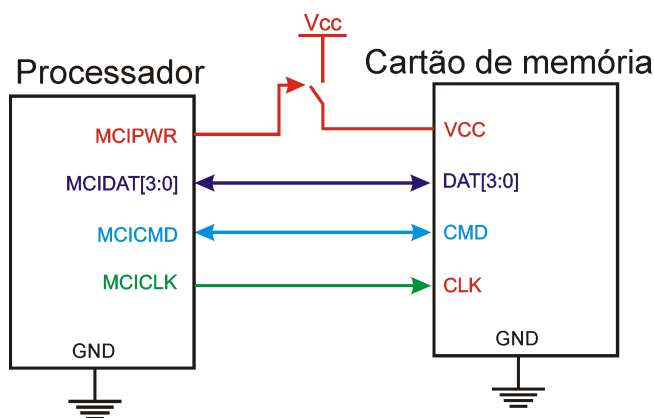


Figura D.10: Interface MCI.

o qual fornece todas as funções específicas do cartão de memória, tais como: geração de *clock*, gerenciamento de energia, transferência de comandos e dados.

A interface MCI possui uma linha de *clock* (MCICLK), uma linha de comandos (MCICMD), um barramento de dados composto por quatro linhas (MCIDAT[3:0]) e uma linha para o controle da alimentação (MCIPWR). A Figura D.10 mostra a interconexão da interface MCI.

A interface MCI está baseada no envio de comandos e respostas, as quais são controladas com bits de início e parada. Na Figura D.11 se descrevem as operações de não resposta e não envio de dados, um comando indica o começo de uma operação, o qual é seguido por uma resposta ou não dependendo da natureza do comando. Os comandos tem um comprimento de 48 bits, dos quais os dois bits mais significativos sempre são '01', a seguir estão o número do comando e os parâmetros e finalmente os sete bits do *checksum* seguido por um bit de parada ('1') [8]. As respostas começam sempre por dois bits '00' e contém a resposta, que muda de acordo com o comando, esta mensagem é finalizada pelos 7 bits do *checksum* e um bit de parada ('1') [8]. A lista completa dos comandos (CMD) e respostas pode ser consultada na folha de especificação do cartão de memória SD SanDisk [8]. A seguir são descritas as operações de leitura e escrita.

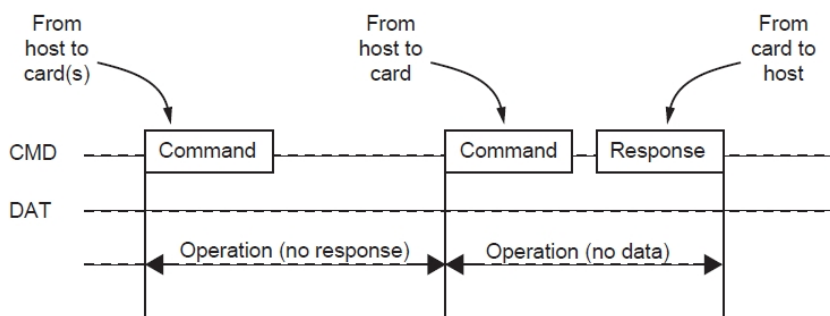


Figura D.11: Operações de não resposta e não dados da interface MCI [8].

### D.4.1 Operação de leitura

A operação de leitura está orientada à transferência de blocos de dados. O tamanho do bloco pode ser definido através do comando CMD16. O comando CMD17 começa a leitura de um bloco, enquanto o comando CMD18 começa a leitura de vários blocos, como mostrado na Figura D.12. Neste ultimo comando, os blocos de dados serão enviados consecutivamente até enviar um comando de parada CMD12 [8]. Os comandos de leitura somente podem ser enviados depois do envio do comando CMD3 e CMD7, os quais colocam o cartão da memória no estado de transferência.

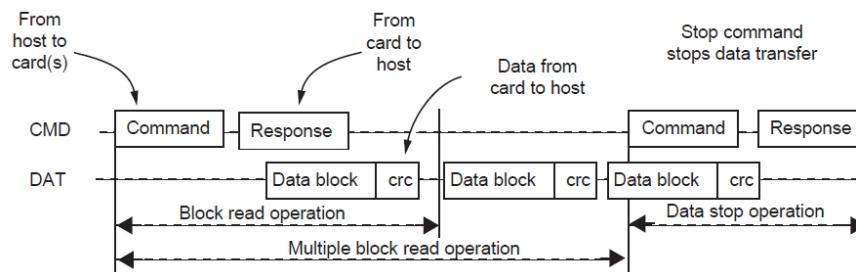


Figura D.12: Operação de leitura da interface MCI [8].

### D.4.2 Operação de escrita

A operação de escrita está orientada à transferência de blocos de dados. Os blocos têm um tamanho mínimo de 512 Bytes. O comando CMD24 começa a escrita de um bloco, enquanto o comando CMD25 começa a escrita de vários blocos, como mostrado na Figura D.13. Neste ultimo comando, os blocos de dados serão enviados consecutivamente até enviar um comando de parada CMD12 [8]. Os comandos de escrita somente podem ser enviados depois do envio do comando CMD3 e CMD7, os quais colocam o cartão da memória no estado de transferência.

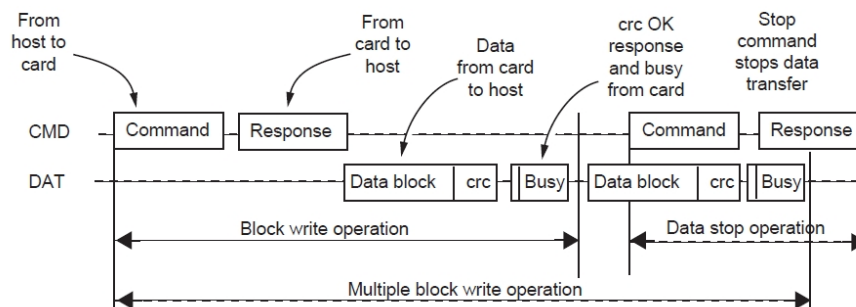


Figura D.13: Operação de escrita da interface MCI [8].

## Apêndice E

### Circuito impresso e diagrama de conexão do nó transmissor

O circuito impresso do nó transmissor das duas camadas e o diagrama de conexão do sistema são apresentados a seguir:

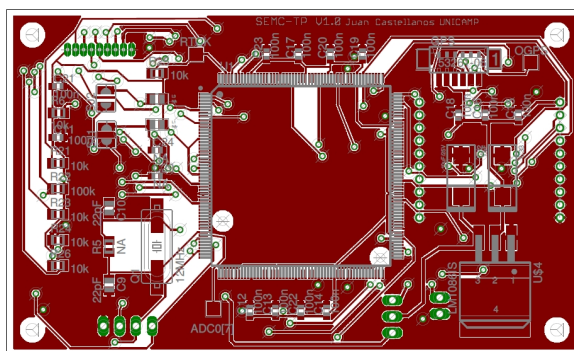


Figura E.1: Camada superior (*top*) do circuito impresso do SEMC-TP Transmissor.

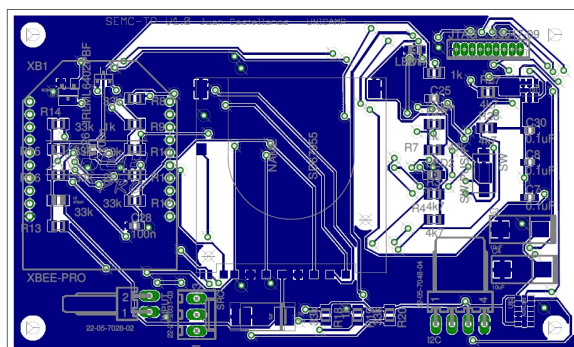


Figura E.2: Camada inferior (*bottom*) do circuito impresso do SEMC-TP Transmissor.

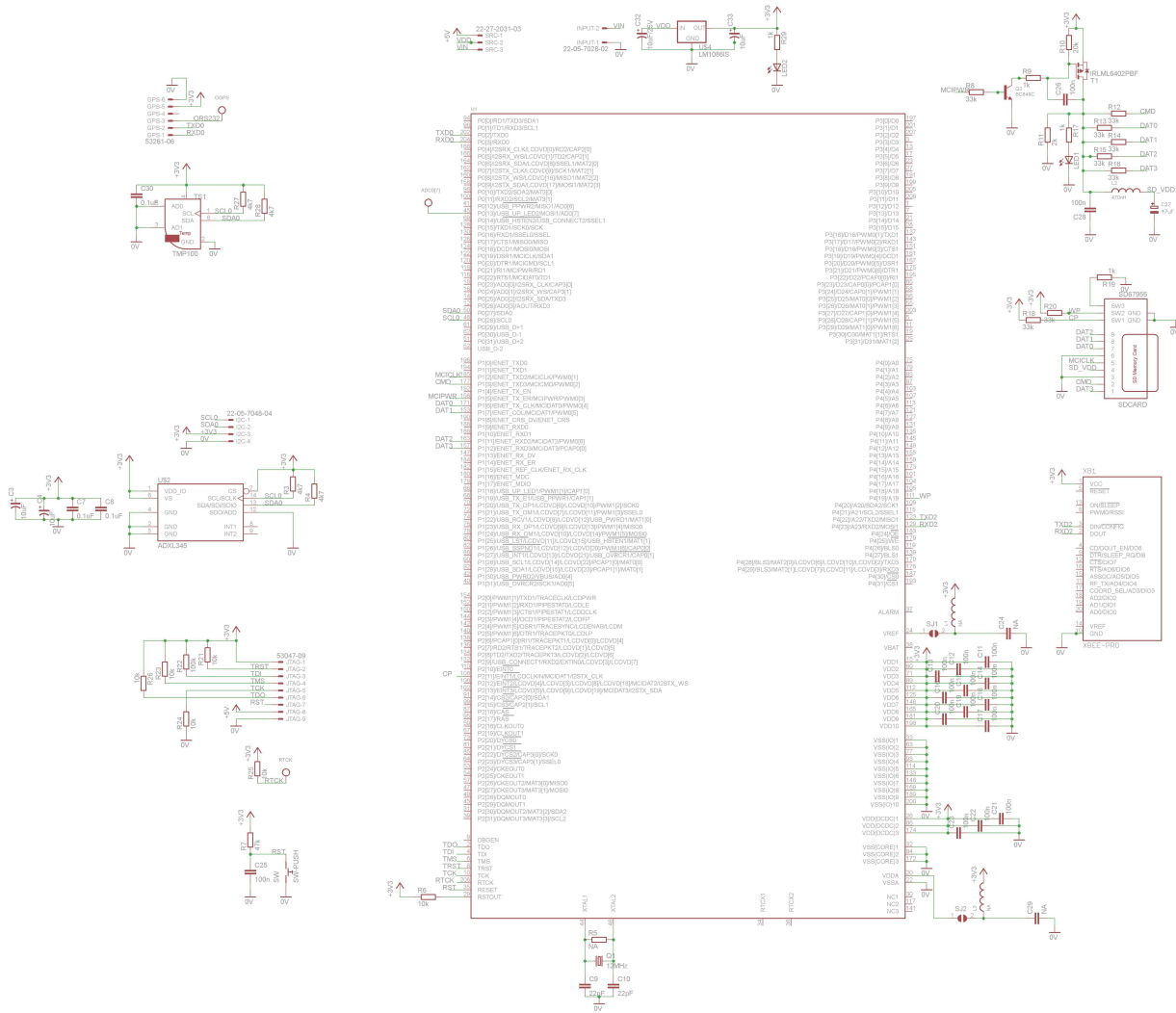


Figura E.3: Diagrama esquemático do SMC-TP Transmissor



# Apêndice F

## Firmware do microcontrolador

A seguir são apresentadas as rotinas mais relevantes do firmware do microcontrolador do SEMC-TP Transmissor.

### F.1 Rotina Principal, rotinas de inicialização e rotinas específicas do DEL-AIC

```
/*
 *
 * File name      : main.c
 * Description    : Prototype 3v2 of SEMC-TP
 * Autor:         Juan Castellanos
 * Date:          19/06/2011
 * Unicamp - DSIF
 *
 */
*****/
#include <nxp/iolpc2478.h>
#include "math.h"
#include "type.h"
#include "irq.h"
#include "target.h"
#include "board.h"
#include "uart.h"
#include "accelerometer.h"
#include "SD.h"
#include "DMA.h"
#include "TMP.h"
#include "timer.h"

#include "math.h"

unsigned char BUS_ID[6]={'0','0','0','0','1','-'};

/*DEFINITIONS OF LIMITS*/
```

```

#define thresholdX  4096    //8192*Acc[mG]
#define thresholdY  4096
#define thresholdZ  4096
#define thresholdTemp  300
#define INITSDADDRESS  564
#define CODE          1234

/*GLOBAL VARIABLES*/

UartLineCoding_t UartLineCoding;          //UART Configuration structure
volatile short RANGE_SEN;                  //Accelerometer Range
// Variable type:SMB380_Data_t with information of: AccX, AccY, AccZ e Temp
extern volatile SMB380_Data_t XYZT;        //Gathering data
extern volatile SMB380_Data_t XYZT_peak;   //Processed data
extern volatile SMB380_Data_t XYZT_threshold; //Limit acceleration data
extern volatile AALPCVARS EventSource;     //Event source
ADXL345_Data_t XYZ_ADXL_CAL;               //Calibration data
State stateSD;                             //Enum variable with SD status
volatile BYTE SdBuffer[600];               //Buffer of SD card
extern volatile DWORD Buffer_Count;         //Counter of SD card's Buffer
extern volatile Boolean FLAG_NMEA;         //GPS read flag
extern volatile Boolean FLAG_TRANS;        //Event's transmission flag
extern volatile Boolean FLAG_CONF;         //Configuration Flag
extern volatile Boolean FLAG_ERASE;        //Erase Flag
extern volatile Boolean FLAG_INFO;         //Information Flag
extern volatile Boolean FLAG_ACCRMS;       //Information Acceleration RMS
extern volatile Boolean FLAG_CAL;          //Calibration Flag
BYTE Date[6], BeginTime[10];               //Date and Begin time
BYTE *pdst;                                //DMA Destination pointer
BYTE *psrc;                                //DMA Source pointer
unsigned int SDADDRESS=INITSDADDRESS;       //SD cluster Address
extern unsigned int EventNumber;            //Event Number
DWORD Buffer_Count2=0;                      //Cluster counter
double RMS_total;                          //RMS total
double RMS_total_final;                    //Previous RMS total.
short RMS_TIME=300;                        //Measurement time in seconds
extern unsigned char indice;                //Confort's Indice.
float RMS_temp;                             //RMS temporal
unsigned int FAT1ADD;                       //Address of FAT
unsigned int FAT2ADD;                       //Address of FAT2
unsigned int ROOTADD;                      //Address of the root
unsigned int CFADD;                         //Address of conf file
unsigned int STADD;                         //First Address for storage

/*****
** Function name:    Init2
**
** Descriptions:    INITIALIZE SD Files and Timer
** parameters:      None
** Returned value:  Initialization is ok (0) or failed (1)
**
*****/

int Init2(void)

```

```

{
    /*GPS data and begin hour acquisition*/
    FLAG_NMEA=FALSE;
    U0IER = 0x07; //Enable UART0
P1: while(!FLAG_NMEA)//While until Rx is done.
    {
        U0IER = 0x07;
    }

    FLAG_NMEA=FALSE; //Inmediatly reset of flag
    if(GPS_dataint(Date, BeginTime)==0)
        goto P1;
    U0IER = 0x00;//Disable UART0
    FLAG_NMEA=FALSE;

    /*Create SD File - #.txt*/
    SDADDRESS= STADD + CreateSDFile(Date, BeginTime);

    __disable_interrupt();
    /*Timer initialization*/
    init_timer();
    __enable_interrupt();
return 0;
}

/*****
** Function name:      intialization
**
** Descriptions:      INITIALIZE ALL MODULES-LPC2478, UARTS, TIMER, MCI, DMA, I2C
**                     AND ACCEL.
** parameters:        None
** Returned value:    Initialization is ok (0) or failed (1)
**
*****/

int intialization(void)
{

    /* Initialization of uC*/
    TargetResetInit();
    Buffer_Count=0;
    FLAG_NMEA=FALSE;
    FLAG_ACCRMS=FALSE;
    RMS_total=0;

    /*UARTS Init*/
    //Configuration of serial parameters
    // Stop bits number
    UartLineCoding.bStopBitsFormat = UART_ONE_STOP_BIT;
    // Parity type
    UartLineCoding.bParityType = UART_NO_PARITY;
    // Word width

```

```

    UartLineCoding.bDataBits = UART_WORD_WIDTH_8;
    //UART for GPS
    UartInit(UART_0,0,NORM);                //UART0 Disable for now
    UartLineCoding.dwDTERate = 4800;        //Baudrate
    UartSetLineCoding(UART_0,UartLineCoding); //Config of UART 0
    //UART for Wireless Communication
    UartInit(UART_2,2,NORM);                //UART2 Always Enable
    UartLineCoding.dwDTERate = 19200;       //Baudrate
    UartSetLineCoding(UART_2,UartLineCoding); //Config of UART 2

/*Accelerometer Initialization*/
//I2C initialization
I2C_Init();
RANGE_SEN=ADXL345_Conf(4,100);
/*TMP100 initialization*/
tmp100_conf();
/*SD initialization*/
// Init Card Present port
InitIOSD();
//Wait until card is present
while(MMC_CP_FIO & MMC_CP_MASK);
// INITIALIZATION SD CARD- CARD IDENTIFICATION MODE
stateSD = Init_Identification_Mode();
// INITIALIZATION SD CARD- DATA TRANSFER MODE
if(stateSD!=Ok)
    return 1;
stateSD = Init_DataTransfer_Mode();
if(stateSD!=Ok)
    return 1;
/*Source and destination DMA address definition*/
psrc = (BYTE *)DMA_SRC;
pdst = (BYTE *)DMA_DST;

/*Enable Interruptions*/
__enable_interrupt();
/*Configuration of SD address*/
SD_conf();
/*reading of configuration file*/
while(readConfFile()); //Configuration from SD file.
Init2();                //INITIALIZE SD Files and Timer
return 0;
}

/*****
** Function name:    main
**
** Descriptions:    Principal Routine.
** parameters:      None
** Returned value:  None
**
*****/
int main(void)
{
    DWORD i=0, m=0, ind=0;

```

```

BYTE conf_vector[6];                //Configuration Vector
SMB380_Data_t XYZT_t;               //XYZ temporal
ADXL345_Data_t XYZ_ADXL_CAL_temp;   //XYZ temporal for calibration purposes
unsigned int j=0, k=0;
unsigned int change;
unsigned int count1=0;               //RMS's timer
int valueCAL=0;

/*Initialization of uC and modules*/
if(intialization()!=0)
    return 1;
else
    UARTSend_string("\r\nINITIALIZED%\r\n");

/*Program*/

while(1)
{

/*RMS*/
if (FLAG_ACCRMS)
{
    FLAG_ACCRMS=FALSE;
    //Vector is generated in the timer's interruption
    RMS_FFT_XYZ();
    RMS_total=sqrtf((RMS_temp*RMS_temp)+(RMS_total*RMS_total));
    if(count1==RMS_TIME)//RMS_TIME
    {
        RMS_total= RMS_total/sqrtf(RMS_TIME);
        count1=0;
        EventSource=RMS;
        if(RMS_total<32.14)//Comparation in mG
            indice=0x35;
        else
        {
            if(RMS_total>32.14&RMS_total<57.65)
                indice=0x34;
            else
            {
                if(RMS_total>57.65&RMS_total<91.83)
                    indice=0x33;
                else
                {
                    if(RMS_total>91.83&RMS_total<145.4)
                        indice=0x32;
                    else
                    {
                        if(RMS_total>145.4&RMS_total<229.59)
                            indice=0x31;
                        else
                            indice=0x30;
                    }
                }
            }
        }
    }
}
}

```

```

    }
}
RMS_total_final=RMS_total;
RMS_total=0;
enable_GPS();
}
else
    count1++;

}

/*READ GPS*/
if (FLAG_NMEA) //OK
{
    U0IER = 0x00; //Disable UART0
    FLAG_NMEA=FALSE;
    if (GPS_NMEA() == 0) // Decode GPS NMEA Protocol
    {
        enable_GPS();
    }
}

//Send incomplete block to SD and Send to base device all events(#)
if (FLAG_TRANS)
{
    for (k=0; k<0xFFFFF; k++);
    //read UART2
    UartRead(UART_2, (BYTE *)conf_vector, sizeof(conf_vector)-1);
    //Verification of ID
    if (conf_vector[0] == BUS_ID[1] & conf_vector[1] == BUS_ID[2] & conf_vector[2] == BUS_ID[3] & conf_vector[3] == BUS_ID[4])
    {
        if (Buffer_Count == 0)
        {
            UARTSend_string("\r\nNo Event Data\r\n"); //No event data
        }
        else
        {
            for (i=Buffer_Count; i<512; i++)
                SdBuffer[i]=0;
            for (k=0; k<0xFFFFF; k++);
            stateSD = Write_SdBuffer(SDADDRESS+Buffer_Count2);
            for (k=0; k<0xFFFFF; k++);
            Buffer_Count2++;

            // Read SD Block - Address, Num of Blocks
            pdst = (BYTE *)DMA_DST;
            for (i=0; i<Buffer_Count2; i++)
            {
                Read_SD(SDADDRESS+i, 1);
                for (k=0; k<0xFFFFF; k++);
                j=512/128;
                for (m=0; m<j; m++)
                {

```

```

        UartWrite(UART_2, (pdst+(m*128)), 128);
        for(k=0; k<0xFFFFF; k++);
    }
}
}
UARTSend_string("\r\nEND_DATA\r\n");
Buffer_Count2=0;
Buffer_Count=0;
EventNumber=0;
FLAG_TRANS=FALSE;
UartRead(UART_2, (BYTE *)conf_vector, sizeof(conf_vector)-1);
}
else
{
    FLAG_TRANS=FALSE;
    //UARTSend_string("\r\nIDUnknown\r\n");
}
}

//Configuration of threshold. The format is @(X,Y,Z,T,J,B)
//nnnn(accel in mg and 10*T in °C).
//Ex. "@X0500".
if(FLAG_CONF)
{
    FLAG_CONF=FALSE;
    for(k=0; k<0xFFFFF; k++);
    //read UART2
    UartRead(UART_2, (BYTE *)conf_vector, sizeof(conf_vector)-1);
    //Save the last events
    for (i=Buffer_Count; i<512; i++)
        SdBuffer[i]=0;
    for(k=0; k<0xFFFFF; k++);
    stateSD = Write_SdBuffer(SDADDRESS+Buffer_Count2);
    for(k=0; k<0xFFFFF; k++);
    //Disable interrupt of timer
    TOMCR = 0;
    ind=0;
    //new-value.
    change=(conf_vector[1]-0x30)*1000+(conf_vector[2]-0x30)*100+(conf_vector[3]-0x30)*
    10+(conf_vector[4]-0x30);
    //X axis configuration
    if(conf_vector[0]=='X')
    {
        XYZT_threshold.AccX=change*512/(1000*RANGE_SEN);//Ex. For 4G. 500mG ->64
        change=XYZT_threshold.AccX;
        ind++;
    }
    //Y axis configuration
    if(conf_vector[0]=='Y')
    {
        XYZT_threshold.AccY=change*512/(1000*RANGE_SEN);//Ex. For 4G. 500mG ->64
        change=XYZT_threshold.AccY;
        ind=ind+6;
    }
}

```

```

//Z axis configuration
if(conf_vector[0]=='Z')
{
    XYZT_threshold.AccZ=change*512/(1000*RANGE_SEN);//Ex. For 4G. 500mG ->64
    change=XYZT_threshold.AccZ;
    ind=ind+11;
}
//Temperature configuration
if(conf_vector[0]=='T')
{
    XYZT_threshold.Temp=change/10;
    ind=ind+16;
}
//BUSID configuration
if(conf_vector[0]=='B')
{
    BUS_ID[0]=0x30;
    BUS_ID[1]=conf_vector[1];
    BUS_ID[2]=conf_vector[2];
    BUS_ID[3]=conf_vector[3];
    BUS_ID[4]=conf_vector[4];
    ind=ind+26;
}

if(ind==0)
{
    UARTSend_string("\r\nERROR\r\n");
    for(k=0;k<0xFFFF;k++);
}
else
{
    ind--;
//Read Configuration file
    pdst = (BYTE *)DMA_DST;          // Address of read block
    Read_SD(CFADD,1);                //Read Configuration file
    for(i=0;i<0xFFFF;i++);
    memcpy(SdBuffer,pdst,512);
    for(i=0;i<0xFFFF;i++);

//Change Threshold data
    if(conf_vector[0]=='B')
    {
        SdBuffer[ind]= 0x30;
        SdBuffer[ind+1]=conf_vector[1];
        SdBuffer[ind+2]=conf_vector[2];
        SdBuffer[ind+3]=conf_vector[3];
        SdBuffer[ind+4]=conf_vector[4];
    }
    else
    {
        SdBuffer[ind]= (change/10000)+0x30;
        SdBuffer[ind+1]=((change/1000)-(SdBuffer[ind]-0x30)*10)+0x30;
        SdBuffer[ind+2]=((change/100)-(SdBuffer[ind+1]-0x30)*10-(SdBuffer[ind]-0x30)*100)+0x30;
        SdBuffer[ind+3]=((change/10)-(SdBuffer[ind+2]-0x30)*10-(SdBuffer[ind+1]-0x30)*100-

```



```

        (SdBuffer[ind]-0x30)*1000)+0x30;
        SdBuffer[ind+4]=((change)-(SdBuffer[ind+3]-0x30)*10-(SdBuffer[ind+2]-0x30)*100-
        (SdBuffer[ind+1]-0x30)*1000-(SdBuffer[ind]-0x30)*10000)+0x30;
    }
    //Write Configuration SD File
    Write_SDBuffer(CFADD);
    for(i=0;i<0xFFFF;i++);
    UartWrite(UART_2, (BYTE *)conf_vector, 5);
    for(k=0;k<0xFFFF;k++);
    UARTSend_string("\r\nDATA_CHANGED\r\n");
    for(k=0;k<0xFFFF;k++);
}

//Reset all variables
Buffer_Count2=0;
Buffer_Count=0;
EventNumber=0;
Init2(); //Do new files and re-boot the timer
}

//Erase SD except configuration file(!cccc. The code is 1234)
if(FLAG_ERASE)
{
    FLAG_ERASE=FALSE;

    for(k=0;k<0xFFFF;k++);
    //read UART2
    UartRead(UART_2, (BYTE *)conf_vector, sizeof(conf_vector)-1);
    change=(conf_vector[0]-0x30)*1000+(conf_vector[1]-0x30)*100+(conf_vector[2]-0x30)*
    10+(conf_vector[3]-0x30);
    if(CODE==change)
    {
        UARTSend_string("\r\nFORMATING...\r\n");
        EraseSD_spc();
        UARTSend_string("\r\nSD Formated\r\n");
        //Reset all variables
        Buffer_Count2=0;
        Buffer_Count=0;
        EventNumber=0;
        Init2(); //Do new files and re-boot the timer
    }
    else
        UARTSend_string("\r\nUnauthorized Code\r\n");
}

//Calibration(^)
if(FLAG_CAL)
{
    FLAG_CAL=FALSE;
    for(k=0;k<0xFFFF;k++);
    //read UART2
    UartRead(UART_2, (BYTE *)conf_vector, sizeof(conf_vector)-1);
    //XYZ_calibration

```

```

XYZ_ADXL_CAL_temp.AccX=0;
XYZ_ADXL_CAL_temp.AccY=0;
XYZ_ADXL_CAL_temp.AccZ=0;

for(k=0;k<10;k++)
{
CP: ADXL345_GetData (&XYZ_ADXL_CAL); //ok
    if(XYZ_ADXL_CAL.AccX == 0 & XYZ_ADXL_CAL.AccY == 0 & XYZ_ADXL_CAL.AccZ == 0)
        goto CP;
    XYZ_ADXL_CAL_temp.AccX=XYZ_ADXL_CAL_temp.AccX+XYZ_ADXL_CAL.AccX;
    XYZ_ADXL_CAL_temp.AccY=XYZ_ADXL_CAL_temp.AccY+XYZ_ADXL_CAL.AccY;
    XYZ_ADXL_CAL_temp.AccZ=XYZ_ADXL_CAL_temp.AccZ+XYZ_ADXL_CAL.AccZ;
}
XYZ_ADXL_CAL.AccX=XYZ_ADXL_CAL_temp.AccX/10;
XYZ_ADXL_CAL.AccY=XYZ_ADXL_CAL_temp.AccY/10;
XYZ_ADXL_CAL.AccZ=XYZ_ADXL_CAL_temp.AccZ/10;

//Write into calibration file
//Read Configuration file
pdst = (BYTE *)DMA_DST; // Address of read block
Read_SD(CFADD,1); //Read Configuration File
for(i=0;i<0xFFFF;i++);
memcpy(SdBuffer,pdst,512);
for(i=0;i<0xFFFF;i++);
//NEW VALUES
ind=30;
valueCAL=XYZ_ADXL_CAL.AccX;
if(valueCAL<0)
{
    SdBuffer[ind]= 0x2D;
    valueCAL=-valueCAL;
}
else
    SdBuffer[ind]= 0x30;
SdBuffer[ind+1]=((valueCAL/1000)*10)+0x30;
SdBuffer[ind+2]=((valueCAL/100)-(SdBuffer[ind+1]-0x30)*10)+0x30;
SdBuffer[ind+3]=((valueCAL/10)-(SdBuffer[ind+2]-0x30)*10-(SdBuffer[ind+1]-0x30)*
100)+0x30;
SdBuffer[ind+4]=((valueCAL)-(SdBuffer[ind+3]-0x30)*10-(SdBuffer[ind+2]-0x30)*100-
(SdBuffer[ind+1]-0x30)*1000)+0x30;
ind=35;
valueCAL=XYZ_ADXL_CAL.AccY;
if(valueCAL<0)
{
    SdBuffer[ind]= 0x2D;
    valueCAL=-valueCAL;
}
else
    SdBuffer[ind]= 0x30;
SdBuffer[ind+1]=((valueCAL/1000)*10)+0x30;
SdBuffer[ind+2]=((valueCAL/100)-(SdBuffer[ind+1]-0x30)*10)+0x30;
SdBuffer[ind+3]=((valueCAL/10)-(SdBuffer[ind+2]-0x30)*10-(SdBuffer[ind+1]-0x30)*
100)+0x30;
SdBuffer[ind+4]=((valueCAL)-(SdBuffer[ind+3]-0x30)*10-(SdBuffer[ind+2]-0x30)*100-

```

```

(SdBuffer[ind+1]-0x30)*1000)+0x30;
ind=40;
valueCAL=XYZ_ADXL_CAL.AccZ;
if (valueCAL<0)
{
    SdBuffer[ind]= 0x2D;
    valueCAL=-valueCAL;
}
else
    SdBuffer[ind]= 0x30;
SdBuffer[ind+1]=((valueCAL/1000)*10)+0x30;
SdBuffer[ind+2]=((valueCAL/100)-(SdBuffer[ind+1]-0x30)*10)+0x30;
SdBuffer[ind+3]=((valueCAL/10)-(SdBuffer[ind+2]-0x30)*10-(SdBuffer[ind+1]-0x30)*
100)+0x30;
SdBuffer[ind+4]=((valueCAL)-(SdBuffer[ind+3]-0x30)*10-(SdBuffer[ind+2]-0x30)*100-
(SdBuffer[ind+1]-0x30)*1000)+0x30;
//Write Configuration SD File
Write_SDBuffer(CFADD);
for (i=0;i<0xFFFF;i++);
//Response
UARTSend_string("\r\Calibration(X,Y,Z,T) [mG,10°C]\r\n");
XYZT_t.AccX=XYZ_ADXL_CAL.AccX;
XYZT_t.AccY=XYZ_ADXL_CAL.AccY;
XYZT_t.AccZ=XYZ_ADXL_CAL.AccZ;
XYZT_t.Temp=0;
UARTSend_accel(XYZT_t, RANGE_SEN);
UARTSend_string("\r\nCALIBRATED\r\n%");
for (k=0;k<0xFFFF;k++);
}
//Information Flag(?)
if (FLAG_INFO)
{
    FLAG_INFO=FALSE;
    UartRead(UART_2, (BYTE *)conf_vector, sizeof(conf_vector)-1);
    UARTSend_string("\r\nINFORMATION\r\n");
    //BUSID
    UARTSend_string("\r\nBUS ID:\r\n<");
    UARTSend_string(BUS_ID);
    //Threshold
    UARTSend_string("\r\nThreshold(X,Y,Z,T) [mG,10°C]\r\n>");
    UARTSend_accel(XYZT_threshold, RANGE_SEN);
    //Calibration
    UARTSend_string("\r\nCalibration(X,Y,Z,T) [mG,10°C]\r\n|");
    XYZT_t.AccX=XYZ_ADXL_CAL.AccX;
    XYZT_t.AccY=XYZ_ADXL_CAL.AccY;
    XYZT_t.AccZ=XYZ_ADXL_CAL.AccZ;
    XYZT_t.Temp=0;
    UARTSend_accel(XYZT_t, RANGE_SEN);
    //Data state
    UARTSend_string("\r\nData Status\r\n*");
    if (Buffer_Count==0)
        UARTSend_string("ND\r\n"); //No event data
    else
        UARTSend_string("DA\r\n"); //Data Available

```

```

//SDCARD
UARTSend_string("\r\nSD Card Status:\r\n");
if(MMC_CP_FIO & MMC_CP_MASK)
    UARTSend_string("Without SD Card");
else
    UARTSend_string("Ok");
//END
UARTSend_string("\r\nEND\r\n");
}
}
return 0;
}

/*****
                                END OF FILE - Prototype 3v1 of SEMC-TP
*****/

```

## F.2 Rotina de interrupção do Timer do DEL-AIC

```

/*****
                                TIMER FILE
*   Autor: Juan Castellanos
*   Date: 08/2010
*   Unicamp - DSIF
*   based on Timer C file for NXP LPC230x Family Microprocessors
*****/

#include <nxp/iolpc2478.h>
#include "type.h"
#include "irq.h"
#include "timer.h"
#include "accelerometer.h"

#include <intrinsics.h>

#define TIMER0_TICK_PER_SEC    500
#define WINDOWS_SIZE           500 // TIME = WINDOWS_SIZE/500 LIMITED TO 500, 1SEG

volatile DWORD timer_counter = 0;    //Acceleration's timer
extern volatile short RANGE_SEN; // Acelerometer Range
SMB380_Data_t XYZT_threshold; //Structure for Threshold data
extern ADXL345_Data_t XYZ_ADXL_CAL; //Structure for Calibration data
SMB380_Data_t XYZT; //Structure for Reading data
ADXL345_Data_t XYZ_ADXL; //Structure for acc. Reading data
SMB380_Data_t XYZT_buffer[600]; //Buffer for reading data
extern Int16S AcctempX[512], AcctempY[512], AcctempZ[512]; //Temporal buffers
SMB380_Data_t XYZT_peak; //Buffer for peak data
SMB380_Data_t XYZT_peak_gen; //Buffer for peak event data
Boolean FLAG_ACCRMS; //Flag of RMS accel.
extern volatile AALPCVARS EventSource; //Structure for event type
extern volatile Boolean FLAG_NMEA; //GPS read flag
float RMSACC; //Acceleration RMS Weighted.

```

```

/*****
** Function name:    Timer0IntrHandler
**
** Descriptions:    Timer Interrupt
**                  executes each TIMER0_TICK_PER_SEC
**
** parameters:      None
** Returned value:   None
**
*****/
void Timer0IntrHandler (void)
{
    int j=0, tmp=0;

    //Acquire acceleration data
    //Temperature Sensor
    TMP100_GetData(&tmp);
    ADXL345_GetData (&XYZ_ADXL);
    XYZT.AccX=XYZ_ADXL.AccX;
    XYZT.AccY=XYZ_ADXL.AccY;
    XYZT.AccZ=XYZ_ADXL.AccZ;
    XYZT.Temp=tmp;
    XYZT_buffer[timer_counter]=XYZT;
    AcctempX[timer_counter]=XYZT.AccX - XYZ_ADXL_CAL.AccX;
    AcctempY[timer_counter]=XYZT.AccY - XYZ_ADXL_CAL.AccY;
    AcctempZ[timer_counter]=XYZT.AccZ - XYZ_ADXL_CAL.AccZ;

    // Comparation routine
    if (timer_counter==WINDOWS_SIZE)
    {
        FLAG_ACCRMS=TRUE;
        XYZT_peak.AccX=0;
        XYZT_peak.AccY=0;
        XYZT_peak.AccZ=0;
        XYZT_peak.Temp=0;

        //Peak acceleration in windowed time
        for(j=0; j<WINDOWS_SIZE; j++)
        {
            //compensation routine
            XYZT_buffer[j].AccX = XYZT_buffer[j].AccX - XYZ_ADXL_CAL.AccX;
            XYZT_buffer[j].AccY = XYZT_buffer[j].AccY - XYZ_ADXL_CAL.AccY;
            XYZT_buffer[j].AccZ = XYZT_buffer[j].AccZ - XYZ_ADXL_CAL.AccZ;

            //abs
            if(XYZT_buffer[j].AccX<0)
                XYZT_buffer[j].AccX = -XYZT_buffer[j].AccX;
            if(XYZT_buffer[j].AccY<0)
                XYZT_buffer[j].AccY = -XYZT_buffer[j].AccY;
            if(XYZT_buffer[j].AccZ<0)
                XYZT_buffer[j].AccZ = -XYZT_buffer[j].AccZ;
        }
    }
}

```

```

    //evaluation
    if(XYZT_buffer[j].AccX>XYZT_peak.AccX)
        XYZT_peak.AccX=XYZT_buffer[j].AccX;
    if(XYZT_buffer[j].AccY>XYZT_peak.AccY)
        XYZT_peak.AccY=XYZT_buffer[j].AccY;
    if(XYZT_buffer[j].AccZ>XYZT_peak.AccZ)
        XYZT_peak.AccZ=XYZT_buffer[j].AccZ;
    if(XYZT_buffer[j].Temp>XYZT_peak.Temp)
        XYZT_peak.Temp=XYZT_buffer[j].Temp;
}
if(EventSource!=RMS)
    EventSource=NE; //Initalize in No Event state
//Acceleration threshold evaluation

if(XYZT_peak.AccX>XYZT_threshold.AccX)
{
    EventSource=X;
    //    U0IER = 0x07; //Enable UART0 (GPS)
}
if(XYZT_peak.AccY>XYZT_threshold.AccY)
{
    if(EventSource==X)
        EventSource=XY;
    else
        EventSource=Y;
    //    U0IER = 0x07; //Enable UART0 (GPS)
}
if(XYZT_peak.AccZ>XYZT_threshold.AccZ)
{
    if(EventSource==X)
    {
        if(EventSource==Y)
            EventSource=XYZ;
        else
            EventSource=XZ;
    }
    else
    {
        if(EventSource==Y)
            EventSource=YZ;
        else
            EventSource=Z;
    }
    //    U0IER = 0x07; //Enable UART0 (GPS)
}
if(XYZT_peak.Temp>XYZT_threshold.Temp)
{
    if(EventSource==NE) //More priority for acceleration
    {
        EventSource=TEMPERATURE;
    }
    //    U0IER = 0x07; //Enable UART0 (GPS)
}
}
if(EventSource==NE)

```

```

        U0IER = 0x00;//Disable UART0
    else
    {
        XYZT_peak_gen=XYZT_peak;
        enable_GPS(); //Enable UART0(GPS)
    }
//    UARTSend_accel(XYZT_peak, RANGE_SEN);
    timer_counter = 0;
}
else
    timer_counter++;

    // clear interrupt
    T0IR_bit.MR0INT = 1;
    VICADDRESS = 0;
}

```

## F.3 Rotina do filtro, decimação e derivação do DEL-AJ

```

/*****
* Function Name: JERK
* Parameters: none
* Return: None
*
* Description: Jerk
*
*****/
void JERK(void)
{
    int i;
    float tmpJ;
    BYTE Jerk_peak_S[6];

    //DEBUG WITH IMPULSE SIGNAL
    /*
    AcZ[0]=15000;
    for(i=1;i<15000;i++)
        AcZ[i]=0;
    */
    FILTER(&AcZ[0], &Jz[0],15000);//FIR Fc=0,3Hz
    derivate(&Jz[0], &Jz[0], 319, 10);//Derivate acceleration data

// Evaluation of Jerk Event
    if(Jz[0]<0)
        Jz[0] = -Jz[0];
    tmpJ=Jz[0];

    for(i=1;i<319;i++)
    {
        if(Jz[i]<0)
            Jz[i] = -Jz[i];
        if(Jz[i]>tmpJ)

```

```

        tmpJ=Jz[i];
    }

//Convert units to m/s^3
Jerk_peak=(tmpJ*9.8*1000*RANGE_SEN/512); //(x10^(-3)m/s^3)  offset=4.10^-3 (m/s^3)

//DEBUG FOR JERK EVENTS
i=0;
    short temp1,temp2,temp3,temp4;
    temp1=Jerk_peak/1000;
    temp2=(Jerk_peak-temp1*1000)/100;
    temp3=(Jerk_peak-temp1*1000-temp2*100)/10;
    temp4=(Jerk_peak-temp1*1000-temp2*100-temp3*10);

    Jerk_peak_S[i] = temp1*0x30;
    i++;
    Jerk_peak_S[i] = temp2*0x30;
    i++;
    Jerk_peak_S[i] = temp3*0x30;
    i++;
    Jerk_peak_S[i] = temp4*0x30;
    i++;
    Jerk_peak_S[i]='\r';
    i++;
    Jerk_peak_S[i]='\n';
    UartWrite(UART_2,(BYTE *)Jerk_peak_S,i+1 );
}

/*****
*
*   File name      : IFIR.c
*   Description    : Prototype 2v2 of SEMC-TP
*   Autor:         Juan Castellanos
*   Date:          19/06/2011
*   Unicamp - DSIF
*
*****/

#include "IFIR.h"

//Constants for Lowpass Filter FIR (LPF) to Fs=100Hz and Fc=1Hz
Flo32 h2[32]={0.001359,0.001551,0.002074,0.002925,0.004083,0.005512,0.007161,
0.008966,0.010854,0.012745,0.014558,0.016213,0.017638,0.018767,0.019549,
0.019950,0.019950,0.019549,0.018767,0.017638,0.016213,0.014558,0.012745,
0.010854,0.008966,0.007161,0.005512,0.004083,0.002925,0.002074,0.001551,
0.001359};

//Constants for Lowpass Filter FIR (LPF) to Fs=10Hz and Fc=0.3Hz
Flo32 h3[16]={0.041919,0.046076,0.049815,0.053060,0.055741,0.057804,0.059204,
0.059911,0.059911,0.059204,0.057804,0.055741,0.053060,0.049815,0.046076,
0.041919};

float Om[3031];

```



```

/*****
* Function Name: Conv
* Parameters: pInt8U x,y(input X*h, output Y)
              unsigned short N,M(Size of X and h)
              float Multfac (Gain correction Factor)
*
* Return: None
*
* Description: Convolution
*
*****/
void Conv(pFlo32 X, pFlo32 h, pFlo32 y, unsigned short N, unsigned short M, float Multfac)
{
    int i,j;
    Flo32 Yi=0;
    pFlo32 hi=h,xi=X;

    for(i=0; i<=(N+M-1); i++)
    {
        Yi=0;
        xi=X;
        for(j=0; j<=(N-1); j++)
        {
            if((i-j)<0)
                j=N;
            else
            {
                if((i-j)>=M) //      if(((i-j)<0) | ((i-j)>=M))
                    Yi=Yi;
                else
                {
                    hi=h+(i-j);
                    Yi=((*xi)*(*hi))+Yi;
                }
                xi++;
            }
        }
        *y=(Yi*Multfac);
        y++;
    }
}

void Conv_int(pInt16S X, pFlo32 h, pFlo32 y, unsigned short N, unsigned short M)
{
    int i,j;
    Flo32 Yi=0;
    pFlo32 hi=h;
    pInt16S xi=X;

    for(i=0; i<(N+M-1); i++)
    {
        Yi=0;

```

```

    xi=X;
    for(j=0; j<=(N-1); j++)
    {
        if((i-j)<0)
            j=N;
        else
        {
            if((i-j)>=M) // if(((i-j)<0)|((i-j)>=M))
                Yi=Yi;
            else
            {
                hi=h+(i-j);
                Yi=((*xi)*(*hi))+Yi;
            }
            xi++;
        }
    }
    *y=Yi;
    y++;
}

}

/*****
* Function Name: decimation
* Parameters: Flo32 I,O (input, output)
*              unsigned short D,N (Decimation Factor, input size)
* Return: none
*
* Description: Integer Decimation and float Decimation
*
*****/

void decimation(pFlo32 I, pFlo32 O,unsigned short D, unsigned short N)
{
    Int16U i,j;

    *O=*I;
    j=D-1;
    O++;
    I++;
    for(i=1;i<=N;i++)
    {
        if(j==0)
        {
            *O=*I;
            j=D-1;
            O++;
            I++;
        }
        else
        {
            j--;
            I++;

```

```

    }
}

void decimation_int(pInt16S I, pInt16S O,unsigned short D, unsigned short N)
{
    Int16U i,j;

    *O=*I;
    j=D-1;
    O++;
    I++;
    for(i=1;i<=N;i++)
    {
        if(j==0)
        {
            *O=*I;
            j=D-1;
            O++;
            I++;
        }
        else
        {
            j--;
            I++;
        }
    }
}

/*****
* Function Name: FILTER
* Parameters: pInt8U I,O (input, output)
*             unsigned short N, (Size of X)
* Return: None
*
* Description: Finite impulse response Filter with Decimation
*
*****/

void FILTER(pInt16S I, pFlo32 O,unsigned short N)
{
    decimation_int(I, I ,5, 15000);
    //3000
    Conv_int(I, &h2[0], &Om[0], 3000, 32);
    //3031
    decimation(&Om[0], &Om[0],10, 3031);
    //304
    Conv(&Om[0], &h3[0], O, 304, 16, 0.748249);
    //319

}

/*****

```

```

* Function Name: derivate
* Parameters: pFlo32 I,O (input, output)
*             unsigned short N, (Size of X)
*             unsigned short Fs (Sample Frequency)
* Return: None
*
* Description: DERIVATE OF ACELERATION DATA
*
*****/

void derivate(pFlo32 I, pFlo32 O,unsigned short N, unsigned short Fs)
{
pFlo32 Ii=I;
int i;

//Erase transition data 20 at the begining and 20 at the end.

Ii=Ii+20;
I=I+20;
for(i=0; i<(N-1-40); i++)
{
    Ii++;
    *O=(*Ii-*I)*Fs;
    I++;
    O++;
}

for(i=i; i<N; i++)
{
    *O=0;
    O++;
}

}

```

# Apêndice G

## Interface gráfica do SEMC-TP

### Código do API do GoogleMaps™ para a visualização dos eventos no Software LabView™

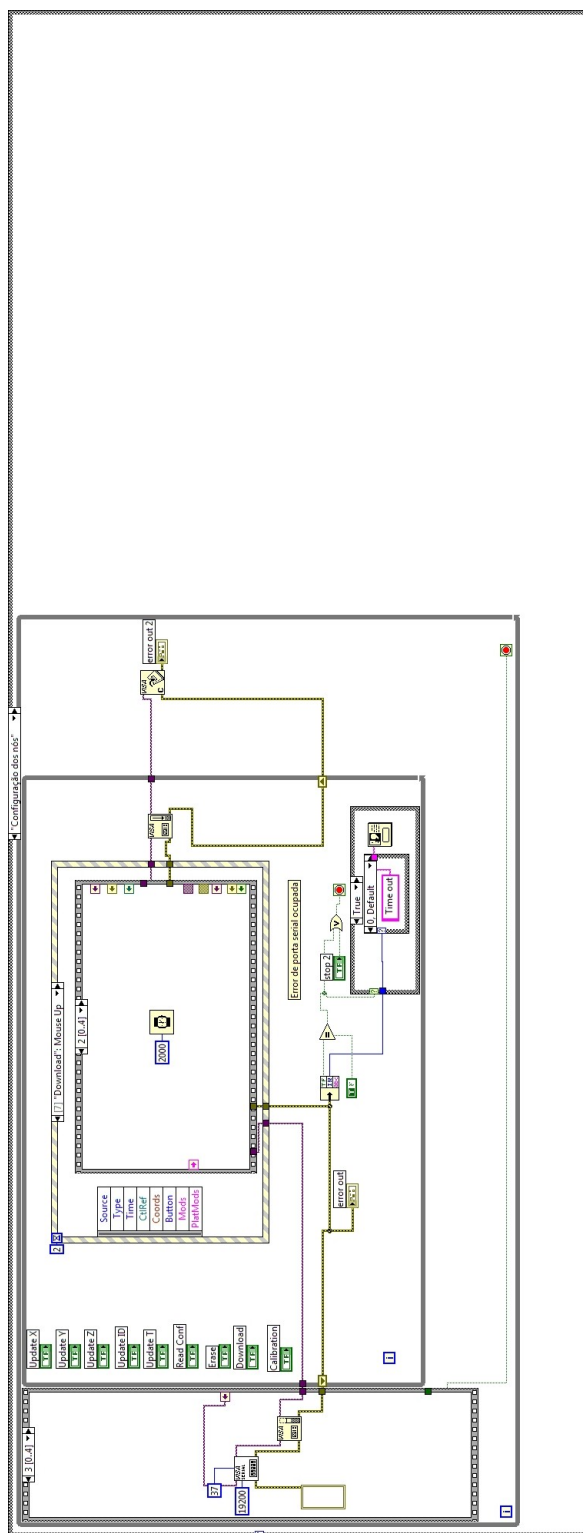
```
<html>
  <head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
    <title>Google Maps Juan's Example</title>
  <script src="http://maps.google.com/maps?file=api&v=2&key=ABQIAAAAluGVG6qZ
anwwdv2PTDXPNhQo02j54eY24PEplTdaJN4NYgHhzRSXVoY6hvm6NGjOPLpsBl-L38M2Lg" type="text/javascript">
  </script>

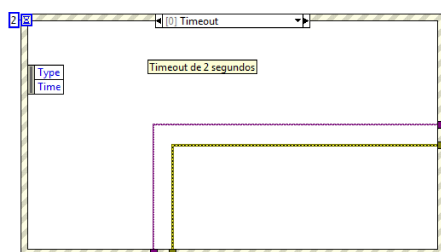
  <script type="text/javascript">
    function initialize() {
      if (GBrowserIsCompatible()) {
var map = new GMap2(document.getElementById("mapa"));
map.setMapType(G_HYBRID_MAP);
var center = new GLatLng(-22.822249,-47.065796);
    map.setCenter(center,15);//Zoom
var marker1 = new GMarker(new GLatLng(-22.817489,-47.07205));//Posição1
var marker2 = new GMarker(center);
map.addOverlay(marker1);
map.addOverlay(marker2);
var mapControl = new GMapTypeControl();
//map.addControl(mapControl);
map.addControl(new GSmallZoomControl());

      }
    }

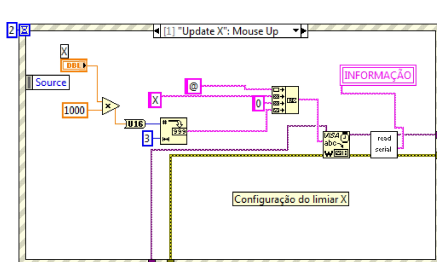
  </script>
</head>
<body onload="initialize()" onunload="GUnload()">
  <div id="mapa" style="width: 800px; height: 414px"></div>
</body>
</html>
```



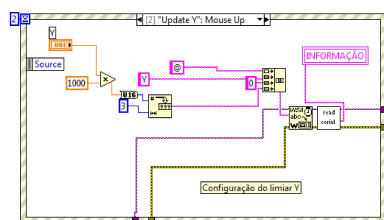




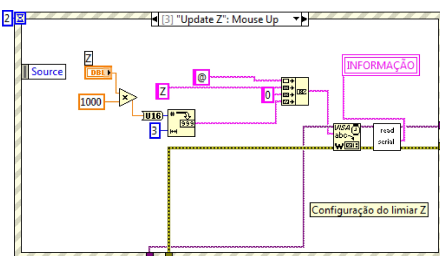
(a) Diagrama de tempo de espera esgotado



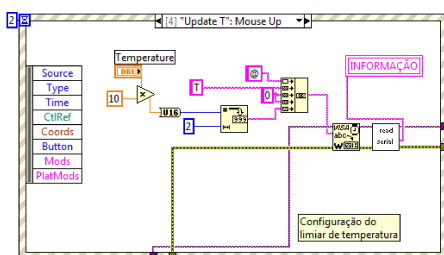
(b) Configuração do limiar no eixo X



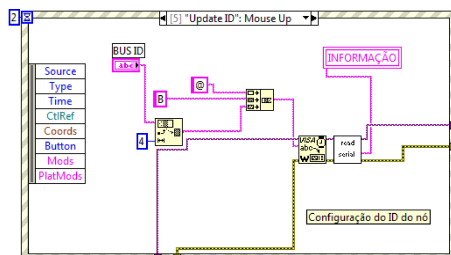
(c) Configuração do limiar no eixo Y



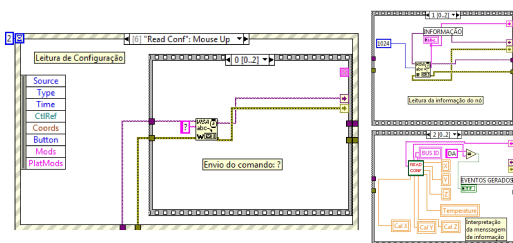
(d) Configuração do limiar no eixo Z



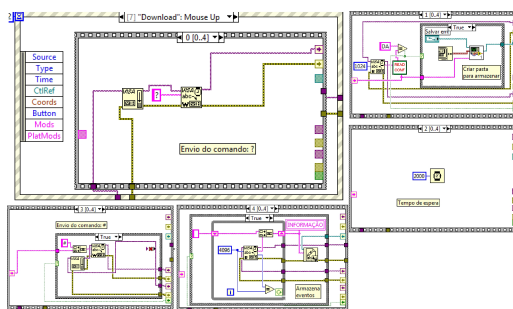
(e) Configuração do limiar de temperatura



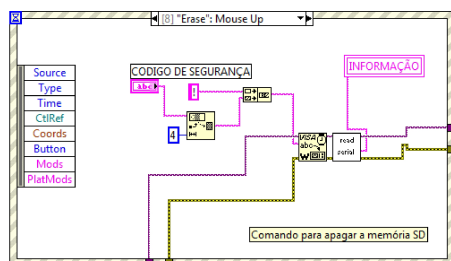
(f) Configuração do ID do nó



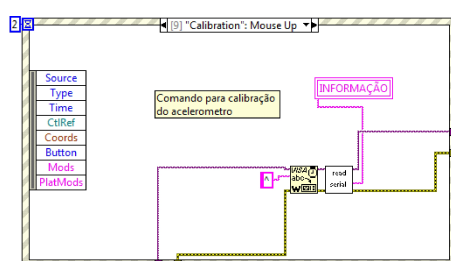
(g) Leitura da informação do nó



(h) Leitura dos eventos do nó



(i) Comando para apagar a memória SD



(j) Comando de calibração

Figura G.3: Fluxograma para o envio dos comandos de configuração dos nós no software LabView™.