



Universidade Federal da Paraíba

Centro de Energias Alternativas e Renováveis

Departamento de Engenharia Elétrica

RAFAEL DUARTE LIMEIRA

**SISTEMA EMBARCADO PARA A MEDIÇÃO DA VELOCIDADE
DO VENTO UTILIZANDO TRANSDUTORES ULTRASSÔNICOS**

João Pessoa, Paraíba
Março de 2015

RAFAEL DUARTE LIMEIRA

SISTEMA EMBARCADO PARA A MEDIÇÃO DA VELOCIDADE
DO VENTO UTILIZANDO TRANSDUTORES ULTRASSÔNICOS

*Trabalho de Conclusão de Curso submetido ao
Departamento de Engenharia Elétrica da
Universidade Federal da Paraíba como parte
dos requisitos necessários para a obtenção do
título de Engenheiro Eletricista.*

Área de Concentração: Automação e Controle

Orientador:

Professor Juan Moises Mauricio Villanueva, M. Sc.

João Pessoa, Paraíba
Fevereiro de 2015

RAFAEL DUARTE LIMEIRA

SISTEMA EMBARCADO PARA A MEDIÇÃO DA VELOCIDADE
DO VENTO UTILIZANDO TRANSDUTORES ULTRASSÔNICOS

*Trabalho de Conclusão de Curso submetido ao
Departamento de Engenharia Elétrica da Universidade
Federal da Paraíba como parte dos requisitos
necessários para a obtenção do título de Engenheiro
Eletricista.*

Área de Concentração: Automação e Controle

Aprovado em / /

Professor Avaliador
Universidade Federal da Paraíba
Avaliador

Professor Juan Moises Mauricio Villanueva, M. Sc.
Universidade Federal da Paraíba
Orientador, UFPB

Dedico este trabalho à minha família, por todo apoio e compreensão, amor e confiança.

AGRADECIMENTOS

Agradeço a Deus, em primeiro lugar, pela minha vida e pelo dom da perseverança, que me permitiu concluir este trabalho.

Agradeço também à minha mãe, Macrina, e meu pai Luiz, por terem se esforçado tanto para me proporcionar uma boa educação, por terem me alimentado com saúde, força e coragem, as quais que foram essenciais para superação de todas as adversidades ao longo desta caminhada.

Agradeço também a toda minha família e amigos, que com todo carinho e apoio, não mediram esforços para eu chegar a esta etapa da minha vida.

Agradeço ao meu orientador professor Juan Moises Mauricio Villanueva e ao professor IsaaC Maia Pessoa, pelos dias despendidos para resolução de problemas com o sistema e pelo apoio em geral. Agradeço ao professor Euler Cássio Tavares de Macedo pelos conhecimentos em eletrônica utilizando amplificadores operacionais e comunicar-me do programa promocional da *Microsoft*, pelo qual me foi cedido o kit desenvolvimento Galileo, de fundamental importância para esse trabalho. Aos professores Antônio Augusto Lisboa de Souza e Carlos Alberto de Souza Filho, por conselhos na construção do sistema eletrônico e possíveis arquiteturas de projeto. Aos professores Alexsandro José Virgínio dos Santos e Cleonilson Protásio de Souza, por auxiliar a fresa da placa de circuito impresso.

Enfim, agradeço a todos que de alguma forma, passaram pela minha vida e contribuíram para a construção de quem sou hoje.

*õE ainda que tivesse o dom de profecia,
e conhecesse todos os mistérios e toda a ciência,
e ainda que tivesse toda a fé,
de maneira tal que transportasse os montes,
e não tivesse amor, nada seria.ö*

Coríntios 2.

RESUMO

A medição da velocidade do vento é de grande importância para diversas aplicações industriais e científicas tais como túneis de vento, aeroportos, estações meteorológicas, navegação de embarcações e estudos de instalação de turbinas eólicas, etc. Um exemplo específico encontra-se no controle de turbinas eólicas, pois, nesse necessita-se de um sensor com uma resposta rápida para compensar rajadas de vento turbulentas, ou de variação brusca, evitando possíveis danos ao sistema como um todo. Esse trabalho tem como objetivo a construção de um sistema embarcado capaz de aferir a velocidade do vento através de transdutores ultrassônicos. Composto por um sistema de processamento, capaz de gerar estímulos aos transdutores e processar sinais resultantes, fazendo a análise do tempo de vôo (ToF - *Time of Flight*), definido como o tempo necessário para que a onda ultrassônica viaje desde o transdutor emissor até o transdutor receptor, cujo tempo de propagação recebe influência direta da velocidade do fluido circulando entre os transdutores. O sistema embarcado é composto pelo kit de desenvolvimento *Intel Galileo*, fornecido pela empresa *Microsoft*, o qual é responsável pela interface humana e comunicação de dados, geração de sinais de comando e análise de sinais, através de uma placa de circuito, desenvolvida em laboratório para geração e tratamento de sinais, sendo uma interface entre a placa Galileo e os transdutores ultrassônicos.

Palavras-chave: Velocidade do vento, sistema embarcado, transdutores, ultrassom.

ABSTRACT

The wind measurement has a huge importance for several industry and scientific applications useful in wind tunnels, airports, meteorological stations, ship driving, wind turbine installation, etc. One specific example is the control of wind turbines, because it needs a fast response sensor to compensate wind gusts avoiding some damage to whole system. This research's objective is building an embedded system capable of measuring the wind speed through ultrasonic transducers. Compound for a processing system, capable of generate signs to the transducers and process the resultant signs, by using the Time of Flight (ToF) analysis, defined as the travel time to the ultrasonic wave flow though one transducer to another, which suffer direct influence of the fluid's speed flowing between the sensors. The embedded system is compound for a development kit Intel Galileo, sponsored by Microsoft, which is responsible for the human interface and data communication, generation of control signs and analyzes, through laboratory developed circuit board acting like an interface between the Galileo and the transducers.

Keywords: wind's speed, embedded system, transducers, ultrasound.

LISTA DE ILUSTRAÇÕES

Figura 1: Anemômetro de deflexão.	11
Figura 2: Anemômetro rotacional.....	12
Figura 3: Anemômetro termoeletrico.....	13
Figura 4: Anemômetro ultrassônico.	13
Figura 5: Diagrama básico de um sistema de medição.	16
Figura 6: Diagrama de disposição e princípio dos transdutores para o efeito Doppler.	19
Figura 7: Disposição dos transdutores pelo método do tempo de trânsito.	20
Figura 8: Detecção de limiar para a medição do ToF.	21
Figura 9: Diferença de fase para o cálculo do Tempo de Trânsito.	22
Figura 10: Diferença de fase em função de $L = n \lambda$ (para n inteiro múltiplo de cinco) e de λ	24
Figura 11: Comportamento periódico da velocidade através da diferença de fase, com $L = 10 \lambda$	25
Figura 12: Faixa de medição da velocidade do fluido em relação à temperatura e distância.	26
Figura 13: Diagrama de conexões da Intel Galileo.	29
Figura 14: Esquemático do módulo ADC da placa Intel Galileo.....	30
Figura 26: Interface de programação Arduino.....	32
Figura 27: <i>Sketch</i> necessário para a comunicação, Ethernet direta, pelo protocolo Telnet.	32
Figura 28: MS Telnet, software para comunicação Telnet, através do Windows.	33
Figura 29: <i>Software</i> PuTTY que fornece as funcionalidades de SSH para Windows.	33
Figura 30: Estabelecimento de conexão via SSH, com o sistema embarcado.	34
Figura 31: Configuração de roteador para fixar o IP.	35
Figura 32: Configuração de servidor virtual.	35
Figura 33: Interface do programa WinSCP no Windows em comunicação com a Galileo.	36
Figura 34: Nano editor de texto orientado ao terminal.	37
Figura 35: Diagrama de conexões de pinos GPIO da Galileo.	39
Figura 36: Informações sobre o sistema GPIO, através dos arquivos de dispositivos.	40
Figura 15: Resultado experimental da caracterização dos transdutores.	42
Figura 16: Arquitetura de tratamento de sinais.....	42
Figura 17: Oscilador projetado com o CI NE555.	43
Figura 18: Estágio de saída, <i>seguidor de emissor</i>	43
Figura 19: Esquemático do tratamento de sinais.	44
Figura 20: Construção do esquemático no software ISIS.....	45
Figura 21: Construção da placa de circuito impresso no ambiente do ARES.	45
Figura 22: Pré-visualização automática tridimensional da placa.	46
Figura 23: Placa de fenolite cobreada fresada para o sistema de condicionamento de sinais.	46
Figura 24: Placa com os componentes soldados.....	47
Figura 25: Diagrama de funcionamento do sistema embarcado.	49
Figura 37: Conexões do circuito de aquisição de dados com comunicação SPI.....	50
Figura 38: Detalhes da placa de condicionamento e aquisição.....	51
Figura 39: Detalhes do GPIO para a comunicação SPI.....	52
Figura 40: Análise gráfica das amostras colhidas.....	53
Figura 41: Algoritmo de detecção do cruzamento da onda com a média.....	54
Figura 42: Comparação de tempo de resposta para operações distintas.....	54
Figura 43: Teste no túnel de ventos do Lens-Ufpb.	55
Figura 44: Script de configuração das portas da Galileo.....	58

SUMÁRIO

Agradecimentos.....	v
Resumo.....	vii
Abstract.....	viii
Lista de Ilustrações.....	ix
Sumário.....	x
1 Introdução.....	11
1.1 Objetivos.....	15
2 Medição de Velocidade de Fluidos utilizando Transdutores Ultrassônicos.....	16
2.1 Introdução.....	16
2.2 Conceitos Básicos sobre Transdutores e Ondas Ultrassônicas.....	17
2.3 Medição da Velocidade dos Fluidos utilizando Transdutores Ultrassônicos.....	17
2.3.1 Método Doppler.....	18
2.3.2 Método do Tempo de Trânsito.....	19
2.3.3 Técnica da Detecção de Limiar.....	20
2.3.4 Diferença de Fase.....	22
2.3.5 Faixas de Medição do Método Diferença de Fase.....	23
3 Placa de Desenvolvimento Intel Galileo.....	28
3.1 Galileo, Conectividade e Ferramentas.....	31
3.1.1 Conectividade.....	31
3.1.2 Ferramentas.....	36
3.1.2.1 Dispositivos (Devices) em Unix.....	38
4 Metodologia.....	41
4.1 Arquitetura e Condicionamento de Sinais.....	41
4.2 Confecção das Placas de Circuito Impresso.....	44
4.3 Especificações do Sistema Embarcado para a Medição da Velocidade do Vento.....	47
5 Resultados.....	49
5.1 Placa de Condicionamento e Aquisição.....	49
5.2 Galileo, GPIO, algoritmo e programação.....	51
5.3 Resultados Experimentais.....	55
6 Conclusão.....	56
Bibliografia.....	57
ANEXO A. Configuração das portas da placa Galileo.....	58
ANEXO B. Código do software.....	59

1 INTRODUÇÃO

A medição da velocidade de ventos é de grande interesse para as mais variadas aplicações industriais e científicas, por exemplo, antes de se construir um parque eólico é importante observar o potencial dos ventos da região durante um tempo prolongado, erros nessa previsão podem ter enorme efeito sobre a energia produzida, bem como a rentabilidade do negócio; turbinas eólicas precisam ser compensadas de acordo com o vento, de maneira que as pás controladas garantam máxima eficiência; na meteorologia de aviação, o conhecimento exato das condições de ventos é essencial para a segurança das aterrissagens e decolagens. Dentre os principais equipamentos de medição da velocidade do vento estão os anemômetros, os quais se apresentam em diversos tipos: Deflexão, Rotacional, Termoelétricos, Tubo de Pressão e Ultrassônicos. Suas principais características são:

- O anemômetro de deflexão é de fácil construção, formado apenas por uma placa metálica com um dos lados livre e outro preso à ponta de um arco que o constitui. Quando o vento incide na placa, essa defletirá em função da velocidade do vento. A incerteza desse anemômetro é grande se compara aos demais, devido seu funcionamento mecânico, que, por exemplo, agrega uma inércia à medição retardando a resposta do sistema a variação do vento. Um exemplo é mostrado na Figura 1 a seguir.

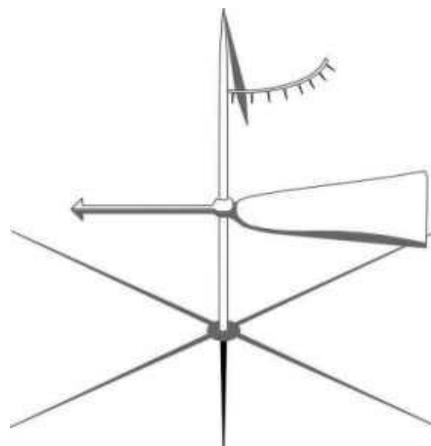
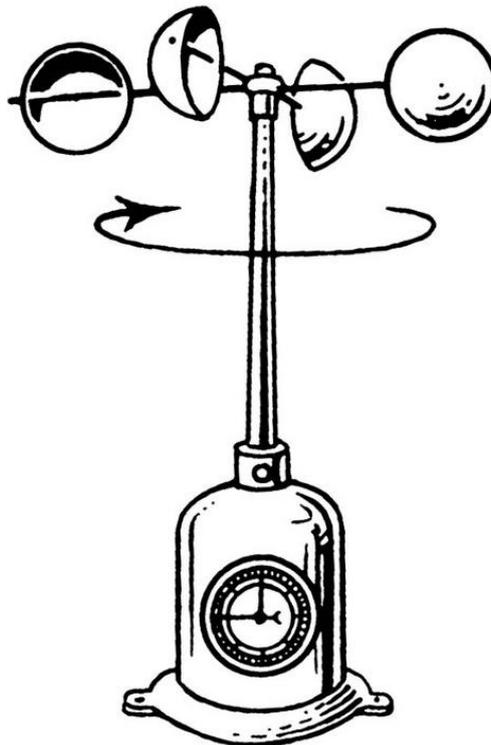


Figura 1: Anemômetro de deflexão.¹

¹ <<http://fisica.ufpr.br/grimm/aposmeteo/cap7/Image572.jpg>>. Acesso em março 2015.

- Os mais simples são os anemômetros rotacionais, construídos por hélices que giram a uma velocidade proporcional a velocidade do vento no qual está imerso. Pode ser composto por semiesferas anexadas a um eixo de rotação. Suas principais desvantagens é estar em contato com o ar interferindo nas linhas de fluxo do ar e apresentar pouca sensibilidade às rajadas de curta duração, devido a sua inércia. Um exemplo deste é apresentado na Figura 2 a seguir.



2

Figura 2: Anemômetro rotacional.

- Os termoelétricos aferem a velocidade do vento por meio da troca de calor, essa feita por meio de um condutor de térmico e o próprio ar. A quantidade de calor retirada do condutor é proporcional à velocidade do vento. Tem a vantagem de não possuir partes móveis, apresentar um tamanho reduzido, assim como maior sensibilidade a baixas velocidades do vento. No entanto são frágeis e com parâmetros muito sensíveis à poluição que possa vir a se depositar no condutor térmico, conseqüentemente exige calibrações com mais frequência. Um exemplar é apresentado na Figura 3 a seguir.

² <<https://peligrosfisicos.files.wordpress.com/2013/07/anemometro-18903.jpg>>. Acesso em março de 2015.



Figura 3: Anemômetro termoelétrico.

- Os sistemas ultrassônicos operam com o princípio de que o fluxo do ar afeta a fase de uma onda acústica com frequência ultrassônica. Suas principais características são: (a) procedimento de medição não invasivo, (b) podem ser utilizados em condições extremas de temperatura e pressão, (c) não contaminam o fluido, (d) não produzem perdas de pressão no meio, (e) boa exatidão e reposta rápida (PÀLLAS-ARENY, 2001). A Figura 4 seguir representa um tipo de anemômetro ultrassônico.

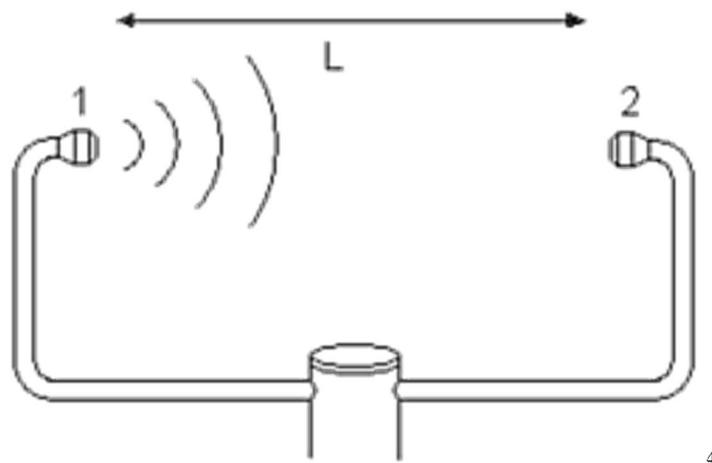


Figura 4: Anemômetro ultrassônico.

³ <<http://www.pce-medidores.com.pt/fichas-dados/images/anemometro-termoeletrico-thermoair3.giffig>>. Acesso em março de 2015.

⁴ <http://www.cheventofa.com/Temp/Principio%20di%20funzionamento_file/image002.gif>. Acesso em março de 2015.

Por tais vantagens escolheu-se o sistema a ultrassom como foco de estudo deste trabalho. Nas últimas décadas, as técnicas de medição da velocidade dos fluidos utilizando transdutores ultrassônicos evoluíram rapidamente, fazendo uso de novas tecnologias de fabricação e avanço no processamento de sinais. Puderam ser desenvolvidas diversas aplicações baseadas em transdutores ultrassônicos, como medição da velocidade do vento, cuja relevância é considerável em aplicações científicas e industriais.

Entre as mais utilizadas técnicas de medição da velocidade de fluidos com transdutores ultrassônicos estão o método de Doppler, utilizado majoritariamente para fluidos gasosos com partículas refletores inertes e líquidos, e o método de Tempo de Trânsito, utilizado para fluidos gasosos e líquidos sem partículas refletoras inertes (PÀLLAS-ARENY, 2001). Neste trabalho utiliza-se o método de Tempo de Trânsito.

Apesar dos avanços na tecnologia, a medição de velocidade de fluidos através do método do Tempo de Trânsito se mostram de baixa precisão, devido à amplitude do sinal, atenuação do fluido, reflexões e ruído aleatório, relação sinal-ruído baixa (TONG, 2002). Entretanto essas técnicas são muito utilizadas pela simplicidade de implementação e por não requerer, necessariamente, a transmissão de ondas continuamente, assim, esses sistemas podem ser projetados para baixo consumo de energia, o que é ideal para sistemas embarcados remotos.

Sistemas embarcados são projetados para funções específicas, contendo estruturalmente componente de *hardware* e *software*. Composto por atuadores, sensores e funções de controle, o *hardware* é utilizado para interagir com o meio. O *software* controla o *hardware* para realizar as funcionalidades do sistema (GASJKI, 1994).

A princípio para o projeto do sistema embarcado, requer-se o desenvolvimento de um modelo para especificar as funcionalidades do sistema, em seguida transformar o modelo em arquitetura, definindo-se a quantidade e tipos de componentes, assim como a interação entre eles (GASJKI, 1994). Esse trabalho aborda os procedimentos e métodos utilizados para a construção de um sistema embarcado de acesso remoto capaz de fazer medições da velocidade do vento, tanto na concepção do circuito de *hardware*, quanto para a criação de um *software* em um sistema embarcado, com um sistema operacional Linux, capaz de comandar o *hardware* e ser acessado remotamente.

1.1 OBJETIVOS

A medição da velocidade do vento é de grande importância para diversas aplicações industriais e científicas tais como túneis de vento, aeroportos, estações meteorológicas, navegação de embarcações e estudos de instalação de turbinas eólicas, etc. Um exemplo específico encontra-se no controle de turbinas eólicas, pois, nesse necessita-se de um sensor com uma resposta rápida para compensar rajadas de vento turbulentas, ou de variação brusca, evitando possíveis danos ao sistema como um todo.

A utilização de transdutores ultrassônicos permite a medição com pouca influencia ao meio, não alterando o fluxo do fluido a ser medido, para isso um sinal oscilante deve ser fornecido para o transdutor emissor, e um sinal de resposta deverá ser condicionado para a aquisição, esses sinais devem ser processados e a partir deles extraído o valor da velocidade do fluido, para isso determinaram-se os principais objetivos desse trabalho:

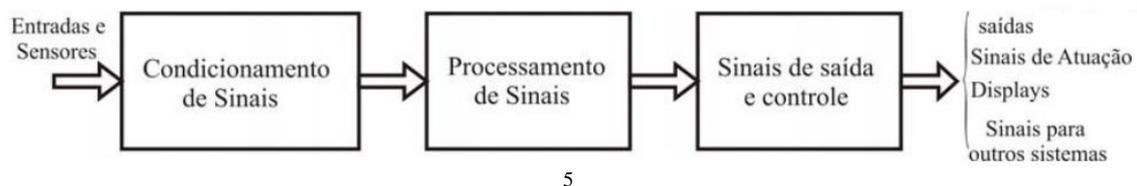
- Construir um sistema embarcado capaz de medir a velocidade do vento utilizando transdutores ultrassônicos.
- Conceber uma placa de circuito impresso capaz de intermediar *software* e kit de desenvolvimento com os transdutores ultrassônicos.
- Configurar o kit de desenvolvimento *Intel Galileo*, em relação à *firmware*, sistema operacional e softwares, de modo a utilizá-lo remotamente e programa-lo para desempenhar os comandos necessários para a interação com o *hardware*.

2 MEDIÇÃO DE VELOCIDADE DE FLUIDOS

UTILIZANDO TRANSDUTORES ULTRASSÔNICOS

2.1 INTRODUÇÃO

Para se constituir um sistema elencam-se elementos e subsistemas, necessários para uma ou mais funções específicas. Um sistema de medição é basicamente descrito pela Figura 5, onde se apresentam os sistemas gerais de um sistema de medição, cada estágio contribui na aquisição das informações do evento. Os sensores transformam informações de um determinado evento, de um domínio para outro, o condicionamento de sinais é responsável pelas operações de amplificação e filtragem dos dados do evento, em seguida o processamento de sinais é onde ocorrem os processos lógicos e matemáticos, no estágio final realizam-se as conversões dos dados para sinais de saída com uma informação definitiva da medição (PÁLLAS-ARENY, 2001) e (FOWLER, 2004).



5

Figura 5: Diagrama básico de um sistema de medição.

No mercado industrial, as grandezas de maior volume de medição são: velocidade de fluidos, nível, pressão e temperatura. Essas quatro variáveis de medição cobrem ao redor do mundo 90% das aplicações em processos industriais (HAUPTMANN, 2001).

Para o caso de medição de fluxo de fluidos utilizando transdutores ultrassônicos, apresenta-se a seguir os conceitos primordiais sobre ondas e transdutores ultrassônicos, assim como, conceitos dos métodos de Tempo de Trânsito.

⁵ VILLANUEVA, Juan Mauricio. **Fusão de Dados Técnicas de Tempo de Trânsito Utilizando Transdutores Ultrassônicos para Medição da Velocidade do Vento**. Tese de Mestrado, PUC-Rio, Rio de Janeiro, 2009.

2.2 CONCEITOS BÁSICOS SOBRE TRANSDUTORES E ONDAS ULTRASSÔNICOS

As ondas mecânicas são oscilações em um meio que pode ser gasoso, líquido ou sólido, e são classificadas de acordo com suas frequências de propagação, por exemplo:

- Ondas sônicas, ondas audíveis aos seres humanos, têm banda de frequência de 20 Hz a 20 KHz.
- Infra-sônicas, frequências inferiores à 20 Hz.
- Ultra-sônicas, frequências superiores à 20 KHz.

Existem diversos métodos de geração de ondas ultrassônicas, como através de material piezoelétrico, eletromagnético e laser. Destes, o material com características piezoelétricas é o mais utilizado, descoberto em 1880, por Jacques e Pierre, consiste no fenômeno da mudança de características físicas de determinados materiais quando sujeitos à um campo elétrico, assim como a criação do mesmo quando o material é sujeito à uma deformação (ROSEN, 1992). Dessa forma, esses materiais são capazes de transformar um campo elétrico oscilante em ondas mecânicas e deformações devido à ondas mecânicas em campo elétrico.

Os transdutores piezoelétricos mais comumente utilizados são construídos a partir de cristais de quartzo ou materiais cerâmicos piezoelétricos, como o titanato de bário e o titanato de dezirconato de chumbo, assim como polímeros flexíveis, como o polivinilo de flúor. Os transdutores ultrassônicos podem funcionar como receptores ou como transmissores, tornando, em muitos casos, possível a simplificação de circuitos de transmissão e recepção de sinais ultrassônicos (WEBSTER, 1999).

2.3 MEDIÇÃO DA VELOCIDADE DOS FLUIDOS UTILIZANDO TRANSDUTORES ULTRASSÔNICOS

Nos últimos anos, novos aprimoramentos estão sendo apresentados para tecnologias ultrassônicas, em relação à medição do fluxo de fluidos, em consequência, diversas aplicações estão sendo progressivamente mais aceitas em processos industriais e científicos. As vantagens da utilização da medição ultrassônica é que a instrumentação

pode ser robusta e não invasiva, porém, a exatidão na determinação da velocidade de fluidos depende do projeto do instrumento e da tecnologia dos sensores (YEH, 2001).

A medição da velocidade de fluidos que utilizam transdutores ultrassônicos consiste basicamente em transmitir uma onda através do transdutor emissor, fazendo com que a onda mecânica interaja com o meio, esta, por sua vez, é recebida pelo transdutor receptor. A partir do sinal recebido, levando em consideração o sinal emitido, são realizados processamentos para estimação da velocidade do fluido (HAUPTMANN, 2001). Para esse processamento, em seguida apresentam-se os métodos de Efeito Doppler e de Tempo de Trânsito.

2.3.1 MÉTODO DOPPLER

O efeito Doppler baseia-se na variação da frequência percebida por um observador em detrimento do movimento relativo da aproximação ou afastamento entre uma fonte emissora de ondas e o observador. Para esse método uma onda ultrassônica contínua é transmitida e ocorre uma variação na sua frequência quando refletida por um objeto que se move em relação ao transmissor, assim, compara-se a diferença entre a frequência da onda emitida e recebida, possibilitando o cálculo da velocidade do fluido.

Apresenta-se a disposição dos transdutores ultrassônicos para a medição de velocidade do fluido baseado no efeito Doppler, em que v representa a velocidade, T_A e T_B são, respectivamente, os transdutores de transmissão e recepção, que formam um ângulo θ com o sentido da velocidade. Para um objeto refletor movendo-se com a velocidade do fluido, a variação da frequência é calculada por (PÁLLAS-ARENY, 2001):

$$\Delta f_D = f_r - f_t = 2f_t \frac{v}{C} \cos \theta \quad (1)$$

onde Δf_D é a variação de frequência pelo efeito Doppler [Hz], f_t é a frequência de transmissão [Hz], f_r é a frequência de recepção [Hz] e θ [graus] é o ângulo entre a velocidade do fluido e a direção de propagação da onda ultrassônica C [m/s], onde f_t , C e θ são constantes, assim, pode-se concluir que a variação da frequência é diretamente proporcional à velocidade v .

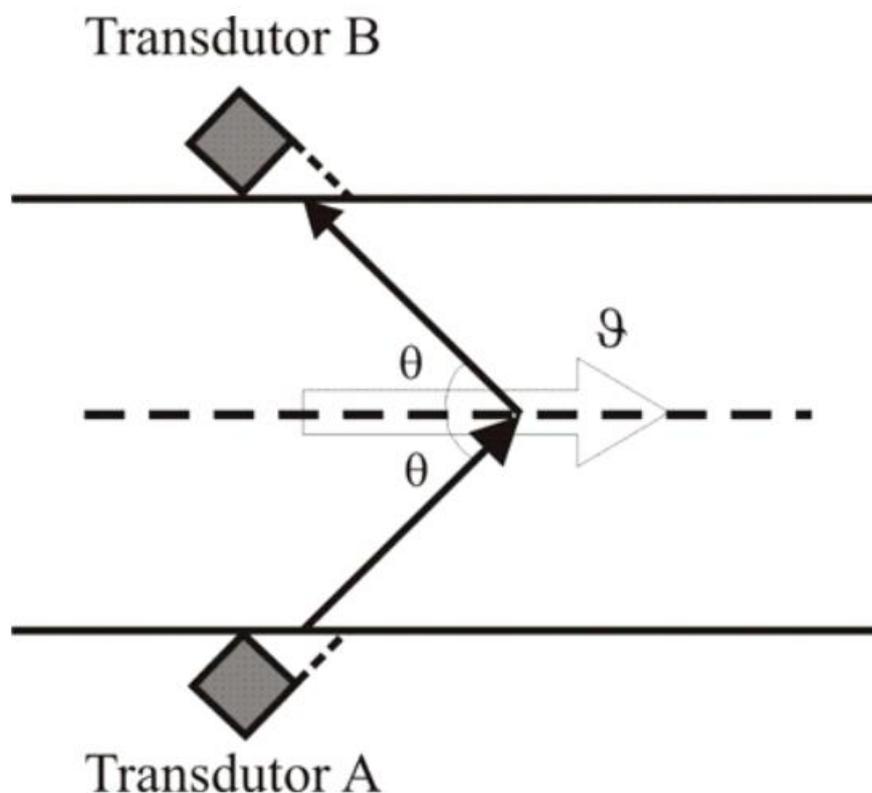


Figura 6: Diagrama de disposição e princípio dos transdutores para o efeito Doppler.

A técnica utilizando o princípio do efeito Doppler é mais usual em fluidos líquidos e gasosos com partículas em suspensão, que têm maior capacidade refletiva (VILLANUEVA, 2007b). Supondo um ambiente de medições, onde o ar apresenta baixa concentração de partículas em suspensão, utilizou-se o método baseado no Tempo de Trânsito, detalhado na seção seguinte, que se mostra mais adequado.

2.3.2 MÉTODO DO TEMPO DE TRÂNSITO

O método do Tempo de Trânsito, como já diz o nome, baseia-se no tempo de propagação de uma onda, desde o transdutor emissor, até o receptor, alinhados com um ângulo especificado à direção do vento, como mostra a Figura 7. A distância entre os transdutores L é fixada e determinada a partir da especificação da faixa de medição da velocidade.

⁶ VILLANUEVA, Juan Mauricio. **Fusão de Dados Técnicas de Tempo de Trânsito Utilizando Transdutores Ultrassônicos para Medição da Velocidade do Vento**. Tese de Mestrado, PUC-Rio, Rio de Janeiro, 2009.

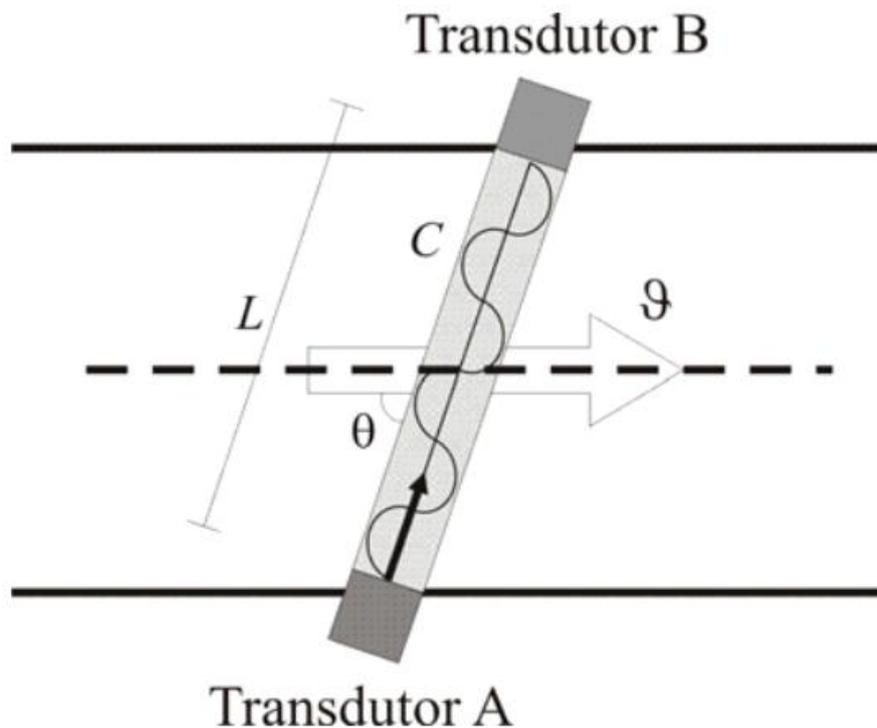


Figura 7: Disposição dos transdutores pelo método do tempo de trânsito.

O tempo de propagação é definido como o Tempo de Trânsito (*Time-of-Flight-ToF*), pelo qual se determina o fluxo , meio às equações que relacionam os tempos de propagação da onda com as velocidades de propagação do fluido (PÀLLAS-ARENLY, 2001). Para esse trabalho é válido detalhar as técnicas da detecção de limiar e da diferença de fase, mostrados nas seções seguintes.

2.3.3 TÉCNICA DA DETECÇÃO DE LIMIAR

Considerando a disposição dos transdutores da Figura 7, calcula-se o tempo de trânsito utilizando a Detecção de Limiar (*TH ó Threshold Detection*) que consiste em emitir um trem de pulsos a partir do transdutor emissor \mathbb{Z}_A e medir o tempo necessário para que esse seja recebido pelo transdutor \mathbb{Z}_B , através de um nível de tensão de limiar, como mostra na Figura 8, porém existe um tempo de retardo inerente ao dispositivo eletrônico, o que faz com que o tempo de retardo medido no instante \mathbb{Z}_{AB}^2 [s] não corresponda ao tempo de trânsito propriamente dito, conseqüentemente esse tempo é

⁷ VILLANUEVA, Juan Mauricio. **Fusão de Dados Técnicas de Tempo de Trânsito Utilizando Transdutores Ultrassônicos para Medição da Velocidade do Vento**. Tese de Mestrado, PUC-Rio, Rio de Janeiro, 2009.

expresso pela soma do atraso eletrônico t_{el}^2 [s] e o tempo de trânsito desejado t_{tr}^2 [s] como expresso na equação a seguir:

$$t_{tr}^2 = t_{el}^2 + t_{tr}^2 \quad (2)$$

A velocidade do vento pode ser calculada através da seguinte equação, de onde, pode-se calcular o tempo de atraso eletrônico t_{el}^2 [s], através de uma medição com vento nulo (VILLANUEVA, 2007b):

$$t_{tr}^2 = \frac{1}{\cos \theta} \frac{L}{C} - t_{el}^2 \quad (3)$$

onde v é a velocidade do fluido através da Detecção de Limiar [m/s], L é a distância entre o transmissor [m] e o receptor, θ é o ângulo entre o eixo dos transdutores [graus] e a propagação do vento e C é a velocidade do som [m/s], a qual varia com a temperatura. A relação entre C e a temperatura T [K] é dada por (KAIMAL, 1963):

$$C = 20,074 \sqrt{T} \quad (4)$$

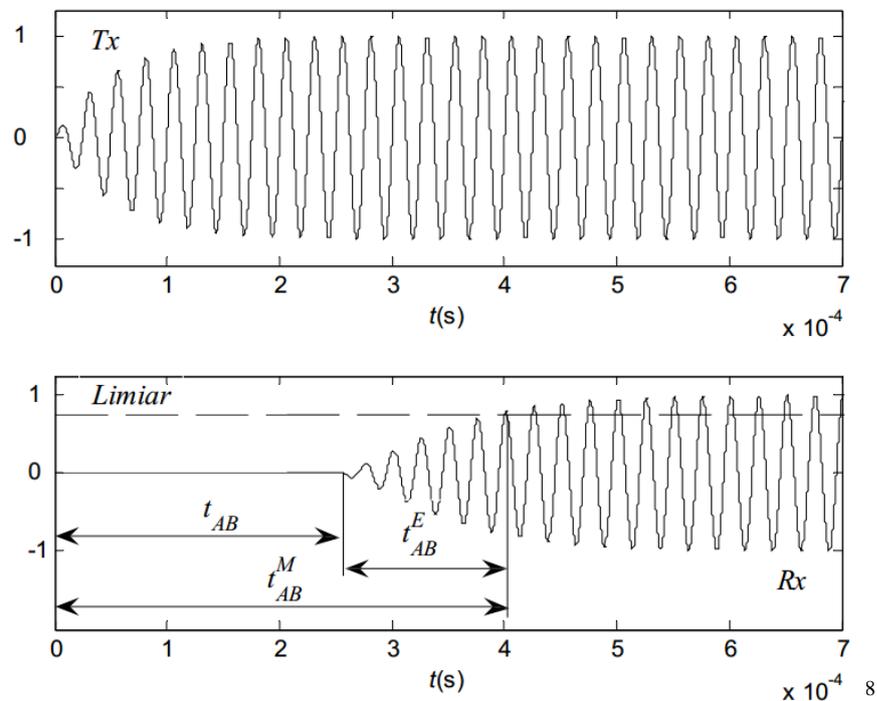


Figura 8: Detecção de limiar para a medição do ToF.

⁸ VILLANUEVA, Juan Mauricio. **Fusão de Dados Técnicas de Tempo de Trânsito Utilizando Transdutores Ultrassônicos para Medição da Velocidade do Vento.** Tese de Mestrado, PUC-Rio, Rio de Janeiro, 2009.

Esse método apresenta baixa imunidade ao ruído adicional, visto que, se em algum momento ocorrer um pico de tensão, sobre influência do ruído, que seja superior a tensão de limiar, detectar-se-á erroneamente um momento de limiar resultando em uma medição incorreta, por este motivo escolheu-se o método da Diferença de Fase, apresentado adiante, como ferramenta de cálculo para o sistema proposto.

2.3.4 DIFERENÇA DE FASE

Para a mesma configuração de transdutores, vista na Figura 7, a técnica da diferença de fase, se baseia na diferença de fase entre a onda ultrassônica emitida e recebida como se pode observar na Figura 9.

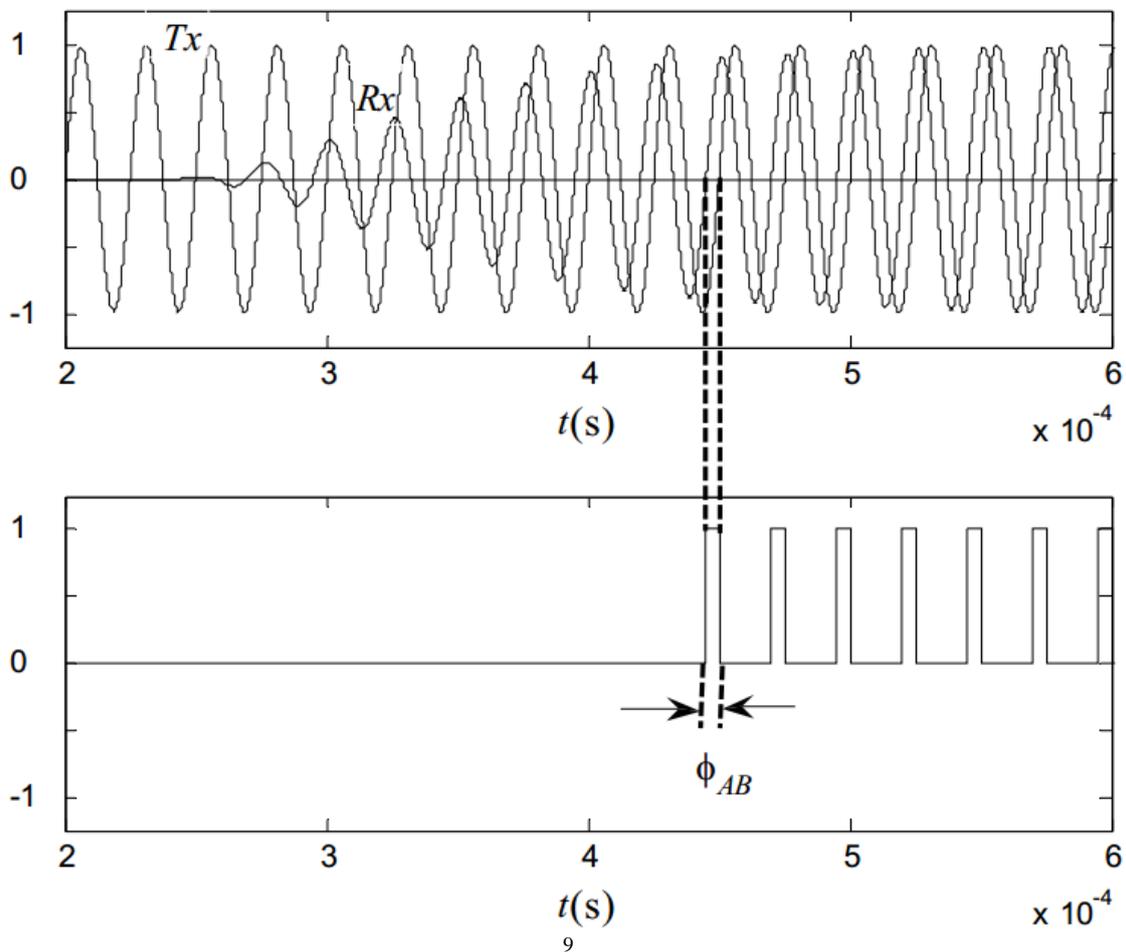


Figura 9: Diferença de fase para o cálculo do Tempo de Trânsito.

⁹ VILLANUEVA, Juan Mauricio. **Fusão de Dados Técnicas de Tempo de Trânsito Utilizando Transdutores Ultrassônicos para Medição da Velocidade do Vento.** Tese de Mestrado, PUC-Rio, Rio de Janeiro, 2009.

Seja ω , uma frequência característica dos transdutores, a diferença de fase entre a onda emitida e a onda recebida é dada por $\phi_{\omega} = 2\omega L \frac{v}{c}$, e a velocidade do fluido é dada por:

$$v = \frac{1}{\cos \theta} \frac{2\omega L}{\phi_{\omega}} - \frac{c}{2}, \quad (5)$$

em que v é a velocidade do fluido [m/s], ω é a frequência do sinal [Hz], L é a distância entre os transdutores [m], ϕ_{ω} é a defasagem entre as ondas [rad] e c é a velocidade do som [m/s].

2.3.5 FAIXAS DE MEDIÇÃO DO MÉTODO DIFERENÇA DE FASE

Equipamentos de medição usualmente apresentam especificações de operação que determinam suas funções e limites de parâmetros que envolvem o processo de medição. Dessa forma, visando definir especificações e procedimentos de projeto, é apresentada nesse capítulo a estrutura de medição utilizando o método da diferença de fase em termos de suas potencialidades e limitações em respeito da sua faixa de medição.

O método abordado para a construção do sistema utiliza uma estrutura de sentido único, onde o transdutor T_1 gera pulsos que sofrem interferências do meio fluido que são recebidos pelo transdutor T_2 , sabendo disso, considera-se o estudo da variação da fase, esta, em função da distancia L entre os transdutores e da velocidade do fluido, assim como definido na equação a seguir:

$$\phi_{\omega, L, v} = \frac{2\omega L v}{c + v} \quad (6)$$

A defasagem instantânea entre o sinal emitido e recebido é definida: $\hat{\phi} = \phi_0$, a diferença entre a fase ϕ e a fase instantânea do ar estático ϕ_0 . Assim, tem-se:

$$\Delta\phi_{\omega, L, v} = \frac{2\omega L v}{c} - \frac{2\omega L v_0}{c + v_0} \quad (7)$$

$$\Delta\phi_{\omega, L, v} = 2\omega L v \frac{v_0}{c(v + v_0)} \quad (8)$$

A equação (8) define a defasagem em função da velocidade do vento v e da distância L , a partir dela, mostra-se na Figura 10 o comportamento da diferença de fase para ventos de 0 a 20 m/s, para distâncias $L = n\lambda$ ($n = 5, 10, 15, 20$), cujo λ é o comprimento da onda transmitida, arbitrando que o ângulo entre o eixo dos transdutores e a velocidade do vento é de $\theta = \pi/3$ a uma temperatura de $T_K = 293,15$ K (20 °C). Conclui-se um comportamento contínuo e linear para todas as fases, porém, existe um limite físico que se deve considerar, já que a diferença de fase é finita e compreendida de zero a 2π , assim a partir da equação (8) a seguir, pode-se definir uma velocidade máxima de medição, considerando a máxima diferença de fase:

$$v_{\text{máx}} = \frac{2\pi}{(2\pi - \theta) \frac{L}{\lambda}} \quad (9)$$

Para velocidades superiores à máxima, a resposta obtida será a repetição do intervalo de defasagem, compreendendo um comportamento periódico, de período igual à máxima velocidade de medição, assim como o gráfico mostrado na Figura 11. Para que não ocorram erros de medição deve-se escolher uma distância entre os transdutores que determine uma velocidade máxima abrangendo toda a faixa de velocidades requeridas pelo projeto.

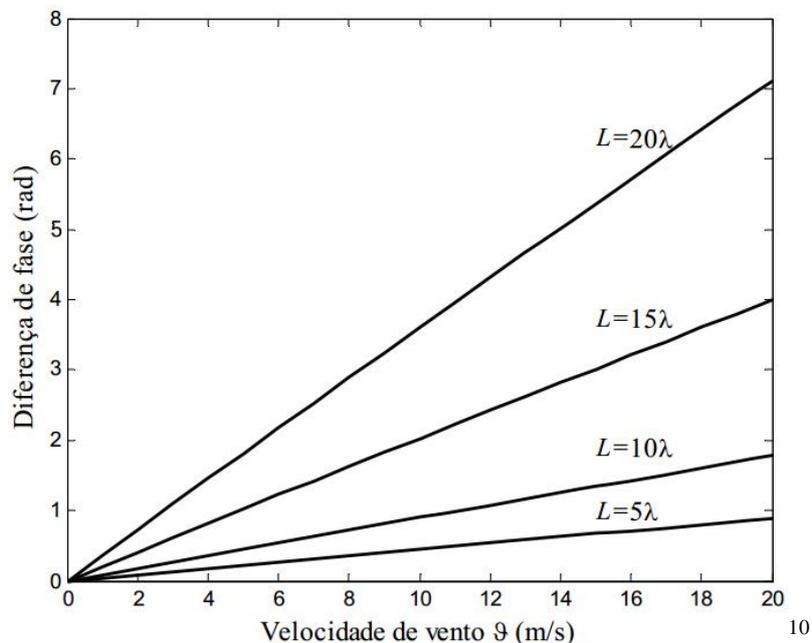
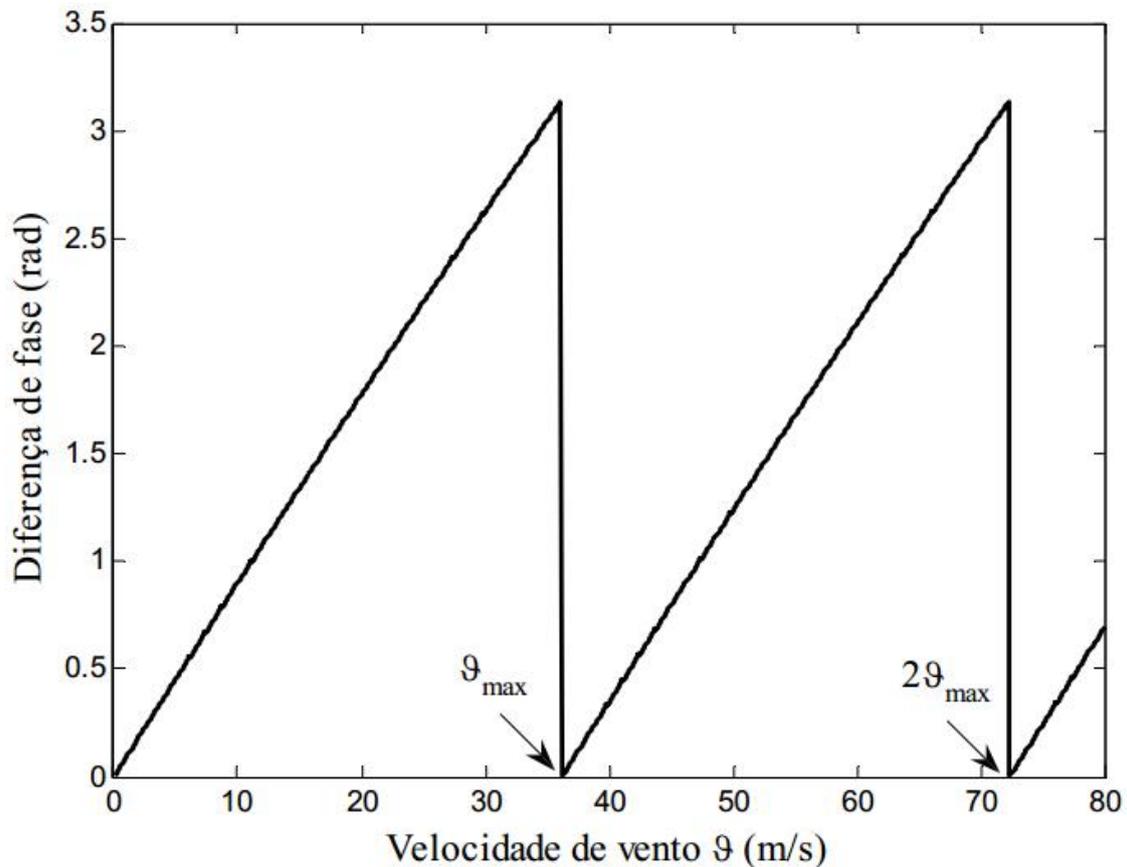


Figura 10: Diferença de fase em função de $L = n\lambda$ (para n inteiro múltiplo de cinco) e de v .

¹⁰ VILLANUEVA, Juan Mauricio. **Fusão de Dados Técnicas de Tempo de Trânsito Utilizando Transdutores Ultrassônicos para Medição da Velocidade do Vento.** Tese de Mestrado, PUC-Rio, Rio de Janeiro, 2009.



11

Figura 11: Comportamento periódico da velocidade através da diferença de fase, com $L = 10$.

Analisa-se também a influência da temperatura na faixa de medição da velocidade do vento. Como a velocidade do vento aumenta com a temperatura ambiente, considerando-se uma temperatura mínima de operação, determinando um $v_{min} = 20,074 \sqrt{v_{max}}$, para uma faixa de medição máxima v_{max} , a distância entre os transdutores pode ser atribuída através da seguinte equação (10):

$$L \leq \frac{v_{max}^2}{2f} + \frac{v_{min}^2}{2f} \quad (10)$$

Sendo uma faixa de temperatura entre 0°C e 40°C , com $f = 40$ KHz e $\theta = \pi/3$, observa-se, na Figura 12, a faixa de medição da velocidade do vento em função da distancia de separação L , podendo-se observar que a faixa de medição diminui com o aumento da distância e com o aumento da temperatura do ar (VILLANUEVA, 2007b).

¹¹ VILLANUEVA, Juan Mauricio. **Fusão de Dados Técnicas de Tempo de Trânsito Utilizando Transdutores Ultrassônicos para Medição da Velocidade do Vento**. Tese de Mestrado, PUC-Rio, Rio de Janeiro, 2009.

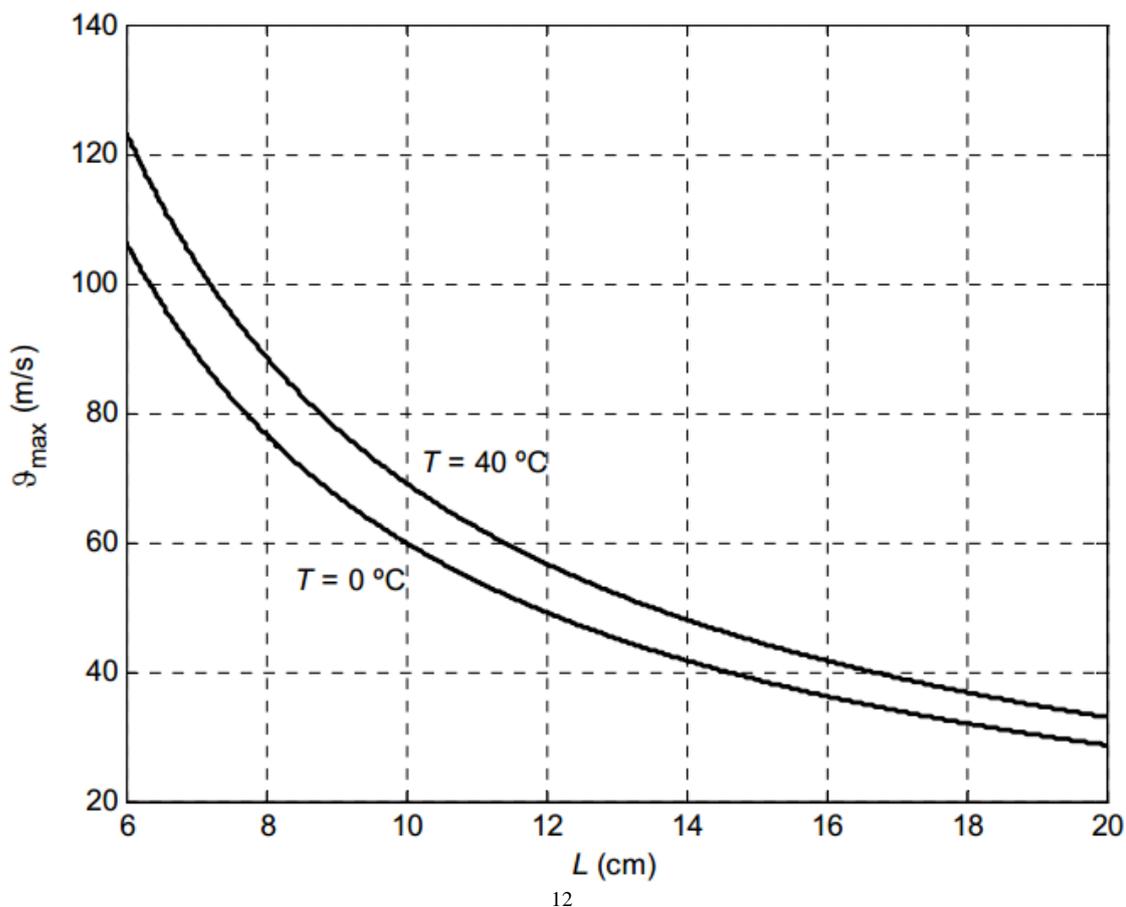


Figura 12: Faixa de medição da velocidade do fluido em relação à temperatura e distância.

Através da tabela Beaufort (Tabela 1), que classifica a velocidade dos ventos, levando em conta a relação velocidade e efeitos resultantes das ventanias no mar e em terra, desenvolvida pelo meteorologista anglo-irlandês Francis Beaufort no século XIX, e utilizada pela marinha do Brasil, pode-se ter uma noção de abrangência da faixa de medição a ser escolhida pelo nosso sistema, determinando a distancia L entre os transdutores (Marinha, 2014).

Tabela 1: Tabela Beaufort de severidade dos ventos.

¹³ Grau	Designação	(m/s)
0	Calmo	<0,3
1	Aragem	0,3 a 1,5
2	Brisa leve	1,6 a 3,3

¹² VILLANUEVA, Juan Mauricio. **Fusão de Dados Técnicas de Tempo de Trânsito Utilizando Transdutores Ultrassônicos para Medição da Velocidade do Vento**. Tese de Mestrado, PUC-Rio, Rio de Janeiro, 2009.

¹³ https://www.mar.mil.br/dhn/chm/meteo/refer/escala_beaufort.htm. Acesso em março de 2015.

3	Brisa fraca	3,4 a 5,4
4	Brisa moderada	5,5 a 7,9
5	Brisa forte	8 a 10,7
6	Vento fresco	10,8 a 13,8
7	Vento forte	13,9 a 17,1
8	Ventania	17,2 a 20,7
9	Ventania forte	20,8 a 24,4
10	Tempestade	24,5 a 28,4
11	Tempestade violenta	28,5 a 32,6
12	Furacão	>32,7

Dessa forma, baseado no gráfico da Figura 12, pode-se concluir que um espaçamento de 10 cm entre os transdutores é mais que o suficiente para cobrir toda a escala Beaufort com uma boa margem de correção para variações de temperatura ambiente.

3 PLACA DE DESENVOLVIMENTO INTEL GALILEO

Foi utilizada para o processamento dos sinais, controle e comunicação a placa de desenvolvimento Intel Galileo, fornecida pela *Microsoft*, através de um programa promocional de tempo limitado.

A Intel Galileo é a primeiro *hardware* aberto da Intel, e é compatível com Arduino, sendo capaz de utilizar sistemas operacionais Linux e Windows, especiais para sistemas embarcados, ambos de 32 bits. Contém, um CPU Intel Quark SoC X1000, desenvolvido especialmente para dispositivos pequenos e com baixo consumo de energia, uma porta Ethernet, duas portas USB, uma *host* e uma *client*, porta serial, um slot *Mini PCI express* e entrada para cartão MicroSD, com capacidade para módulo de comunicação *WiFi*, apresenta 512 KBytes de memória RAM no processador, mais 256 MBytes em placa e 8 MBytes de memória flash.

A placa é alimentada com uma fonte de 5 V e corrente de até 3 A, dispõe de 14 pinos de entrada e saída (I/O), dos quais 6 podem prover saídas PWM, 6 pinos de entrada analógica, corrente de saída máxima em cada pino I/O de 25 mA e até 800 mA para pinos de tensão de 3,3 V e 5 V. A placa tem o seguinte diagrama de conexões, mostrado na Figura 13, a seguir:

geração da placa Galileo. Esse fato tornou necessário o uso de um circuito externo de aquisição para o propósito do sistema.

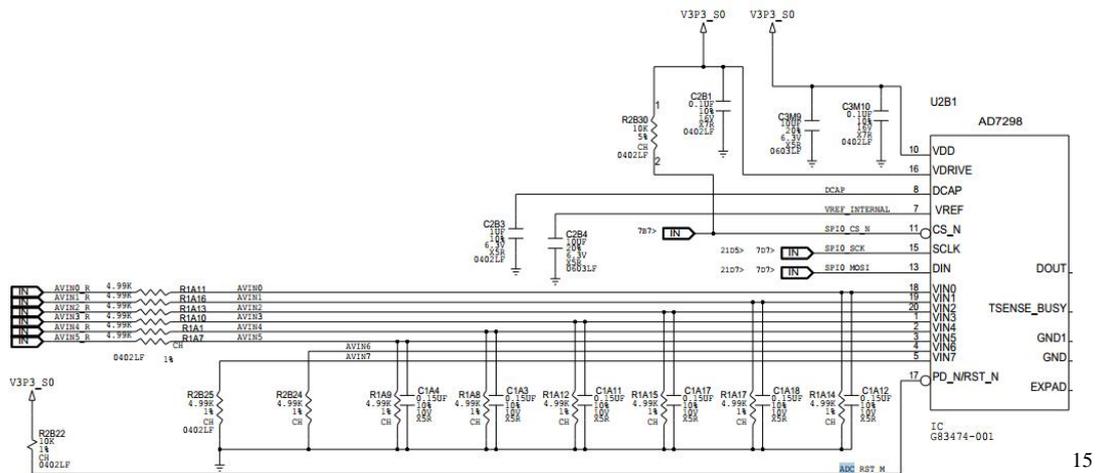


Figura 14: Esquemático do módulo ADC da placa Intel Galileo.

A placa Intel Galileo, ainda dispõe diversos meios de comunicação com o computador, ou outra placa Arduino e micro controladores, como:

- UART ó Comunicação serial UART TTL (5 V/3,3 V), disponível no pino digital zero (RX) e no um (TX), adicionalmente uma segunda porta UART que provém suporte para RS-232.
- Portas USB *client* ó Permite comunicação serial (CDC-ACM) em USB, provém a comunicação serial com o aplicação *Serial Monitor* ou outra no computador. É também utilizada para atualizar os arquivos *sketches* (arquivos de programação do Arduino) para a placa.
- Portas USB *host* ó Permite que placa atue como um *host* USB para periféricos conectados como mouses, teclados e *smartphones*.
- Mini PCI Express* (mPCIe*) ó Permite conexão com módulos como WiFi, Bluetooth ou conexão com celular.
- Ethernet RJ45 ó O conector Ethernet RJ45 é fornecido para permitir que a placa se conecte a redes com fio.
- micro card reader ó A comunicação entre a placa e o cartão SD é provida por um controlador SD integrado, e não requer a utilização da interface SPI, a comunicação com o cartão SD é feita em até 50 MHz dependendo da classe do cartão.

¹⁵ Intel Corporation. **Galileo Intel Quark x1000**. Galileo Schematic, Intel Corporation, Hillsboro, 2014.

- TWI/I²C* ó O software Arduino inclui uma biblioteca para simplificar o uso do barramento TWI/I²C.
- SPI ó Comunicação SPI, utilizando a biblioteca Arduino ou através do sistema operacional.

3.1 GALILEO, CONECTIVIDADE E FERRAMENTAS

Foi utilizada a placa Galileo com um sistema operacional, ao invés de utilizá-la como Arduino, pois, dessa forma pode-se aproveitar todas as funcionalidades e facilidades que o Arduino não poderia dispor. Através de um sistema operacional, pode-se programar em qualquer linguagem, enquanto que o Arduino só aceita C++, existe a vantagem de um ambiente multitarefas, podendo-se executar diversas tarefas de maneira escalonada, a facilidade da conexão com a internet possibilitando a criação de um supervisor remoto, fácil instalação de atualizações e disponibilização de dados, a portabilidade do *software* podendo o mesmo ser utilizado em outros sistemas embarcados com sistema operacional.

3.1.1 CONECTIVIDADE

É possível comunicar-se com a placa, a partir de comunicação serial, utilizando a porta USB *client*, o que é usual quando se deseja operar em seu núcleo Arduino, porém, quando o objetivo é utilizar o sistema operacional, é mais vantajoso utilizar comunicação via Ethernet, seja por protocolo SSH ou Telnet.

É possível utilizar a placa Galileo como um Arduino nativo, utilizando inclusive a mesma interface de programação, através de linguagem C/C++, utilizando o *software* Arduino 1.5.3, mostrado na Figura 15, o qual se comunica com a placa através de comunicação serial via USB *client*, pela qual, atualizações importantes no *firmware* podem ser feitas.

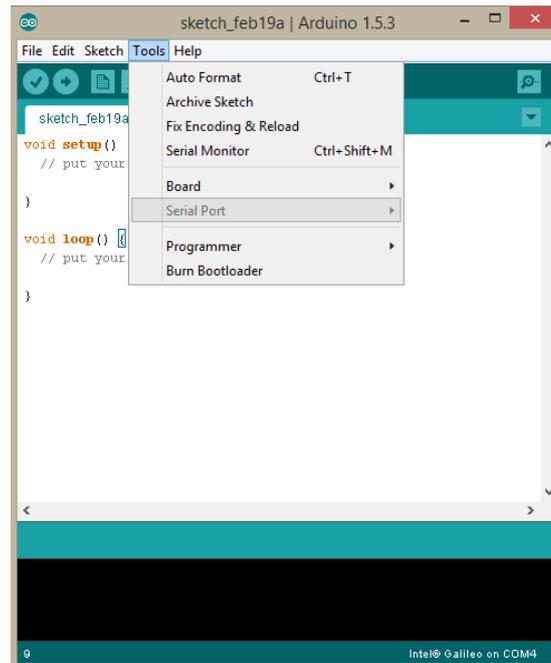


Figura 15: Interface de programação Arduino.

É possível, via Ethernet, com o cabo conectado na Galileo diretamente ligado ao computador, estabelecer uma comunicação pelo protocolo Telnet, porém, deve-se previamente carregar um código *sketch*, mostrado na Figura 16, no Arduino da placa para que a mesma possua um protocolo de internet (IP), possibilitando a conexão através de *softwares* como o MS Telnet, cuja interface pode ser vista na Figura 17.

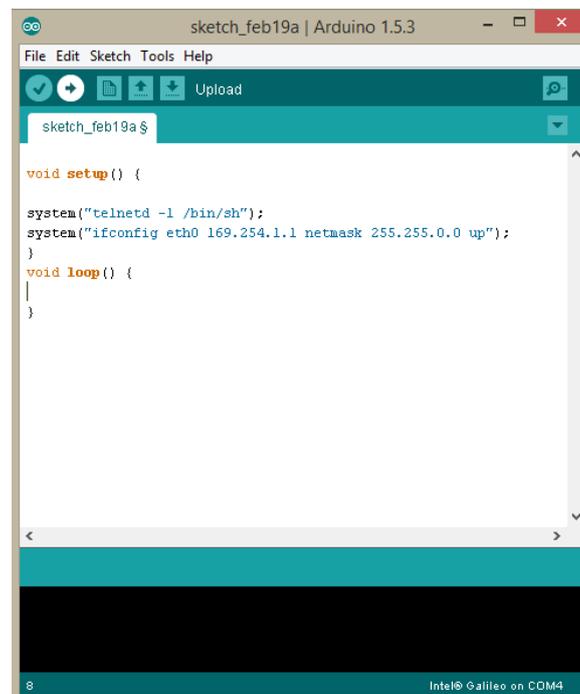


Figura 16: *Sketch* necessário para a comunicação, Ethernet direta, pelo protocolo Telnet.



Figura 17: MS Telnet, software para comunicação Telnet, através do Windows.

O protocolo Telnet provém acesso rápido e estável ao sistema operacional da placa Galileo, sendo uma boa ferramenta para diversos propósitos, porém, o método escolhido utiliza o protocolo SSH.

O SSH apresenta as mesmas funções que o Telnet, porém com a vantagem da criptografia na conexão, tornando a comunicação mais segura. Através do acesso remoto ao dispositivo conectado à Internet, é possível a visualização de dados, configuração, programação e todas as funcionalidades que o sistema operacional, que segue o padrão POSIX (*Portable Operating System Interface*), e a placa Galileo possam fornecer, até mesmo de fora da rede local.

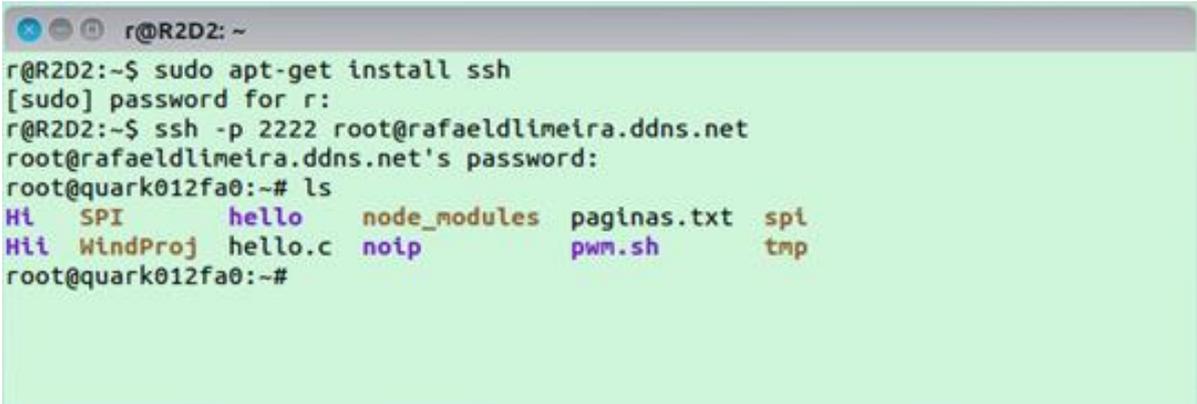
O protocolo SSH, faz parte da suíte de protocolos TCP/IP, TCP (*Transmission Control Protocol* - Protocolo de Controle de Transmissão) e o IP (*Internet Protocol* - Protocolo de Internet), (Stevens, 1993). Para estabelecer essa comunicação em um ambiente Windows, deve-se dispor de um *software*, como o PuTTY, Figura 18.



Figura 18: *Software* PuTTY que fornece as funcionalidades de SSH para Windows.

Em um sistema Linux, no caso Ubuntu, pode-se instalar o *software* SSH, através da seguinte linha de comando em um terminal: `sudo apt-get install ssh`; quando devidamente instalado seu uso pode ser feito através do próprio terminal. Deve-se utilizar a seguinte linha de comando: `ssh -p porta do servidor usuário@endereço`

do servidor, para conectar-se remotamente ao servidor, no caso, a placa Intel Galileo, assim como mostra-se na Figura 19, na qual o número da porta, usuário e servidor são, respectivamente, 2222, root e rafaeldlimeira.ddns.net.

A terminal window titled 'r@R2D2: ~' showing the following commands and output:

```
r@R2D2:~$ sudo apt-get install ssh
[sudo] password for r:
r@R2D2:~$ ssh -p 2222 root@rafaeldlimeira.ddns.net
root@rafaeldlimeira.ddns.net's password:
root@quark012fa0:~# ls
Hi  SPI      hello    node_modules  paginas.txt  spl
Hi  WindProj  hello.c  noip          pwm.sh       tmp
root@quark012fa0:~#
```

Figura 19: Estabelecimento de conexão via SSH, com o sistema embarcado.

O campo servidor pode ser o número do IP fornecido pelo roteador da rede para a placa, através da porta 22, porém esse procedimento limita o acesso à placa, somente à rede local. Para possibilitar o acesso remoto através da rede Internet, se deve configurar o roteador ao qual a placa está conectada, no caso um modelo da marca Arris, por meio de um navegador de internet, geralmente pelo endereço: 192.168.0.1; atribuído por padrão de fábrica ao roteador.

Primeiramente fixa-se o IP fornecido pelo roteador à placa, nas configurações de DHCP (*DynamiC Host Configuration Protocol*, Protocolo de configuração dinâmica de host), na área de LAN (*Local Area Network*, Rede de Área Local), deve-se identificar o endereço MAC (*Media Access Control*, Controle de Acesso à Mídia) da placa, pela lista de clientes DHCP, e adicionar um cliente fixo DHCP, assim como visto na Figura 20.

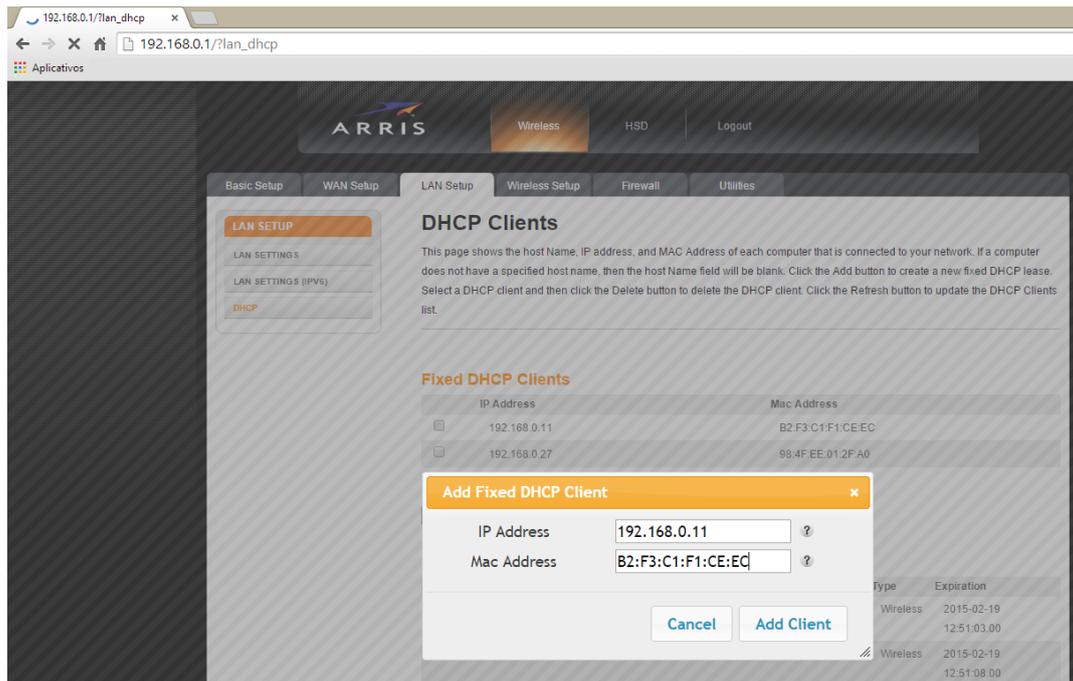


Figura 20: Configuração de roteador para fixar o IP.

Em seguida, deve-se configurar um servidor virtual, que vai direcionar um acesso ao IP internet da rede ao IP da máquina local, através das portas externas e locais conforme pode ser visualizado na Figura 21, onde a porta externa 2222 está associada ao IP fixo da placa Galileo e a porta local 22.

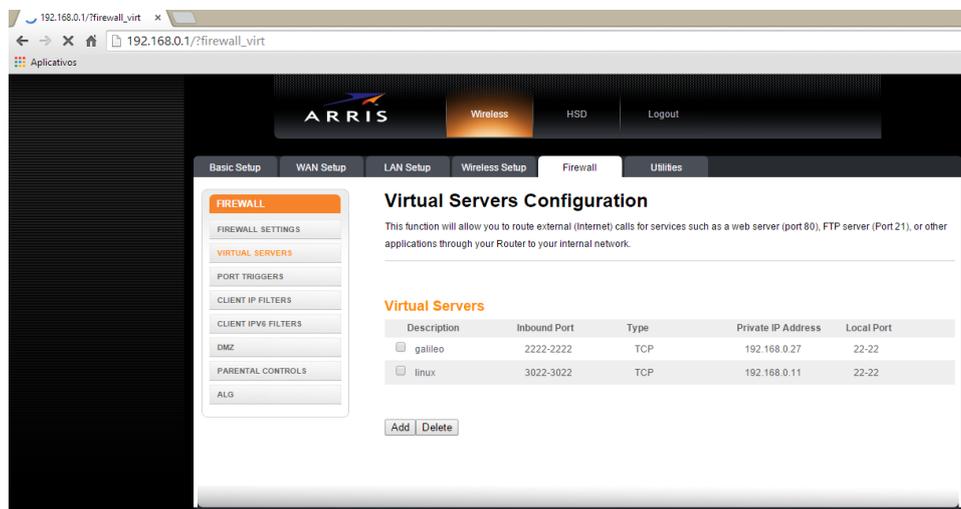


Figura 21: Configuração de servidor virtual.

O roteador dispõe de uma configuração DNS (*Domain Name System*, Sistema de nome domínio), que fornece um nome de domínio para ser utilizado para acesso remoto ao invés do IP da internet atribuído ao roteador, visto que esse se altera constantemente, porém, requer um cadastro pago na empresa Dyn.

Uma alternativa para é fazer o cadastro gratuito no site: www.noip.com; criar um nome de domínio e fazer o download do *software* noip, que associa o IP atual da Internet ao nome do domínio cadastrado, para isso o *software* deve estar operando de forma integral, o que foi feito na placa Galileo.

3.1.2 FERRAMENTAS

Além das ferramentas de conectividade existem aplicações que auxiliam no processo de desenvolvimento, como editores de texto, a criação de scripts e ainda relacionado à conectividade transferência de arquivos. Muito utilizado para transferir arquivos de uma máquina para outra, o SCP, está disponível em Windows pelo *software* WinSCP, que dispõe de uma interface gráfica, mostrada na Figura 22, com navegação por pastas e interface orientada ao mouse, é possível fluxo de arquivos bidirecional e edição de arquivos de texto da máquina remota pelo próprio aplicativo.

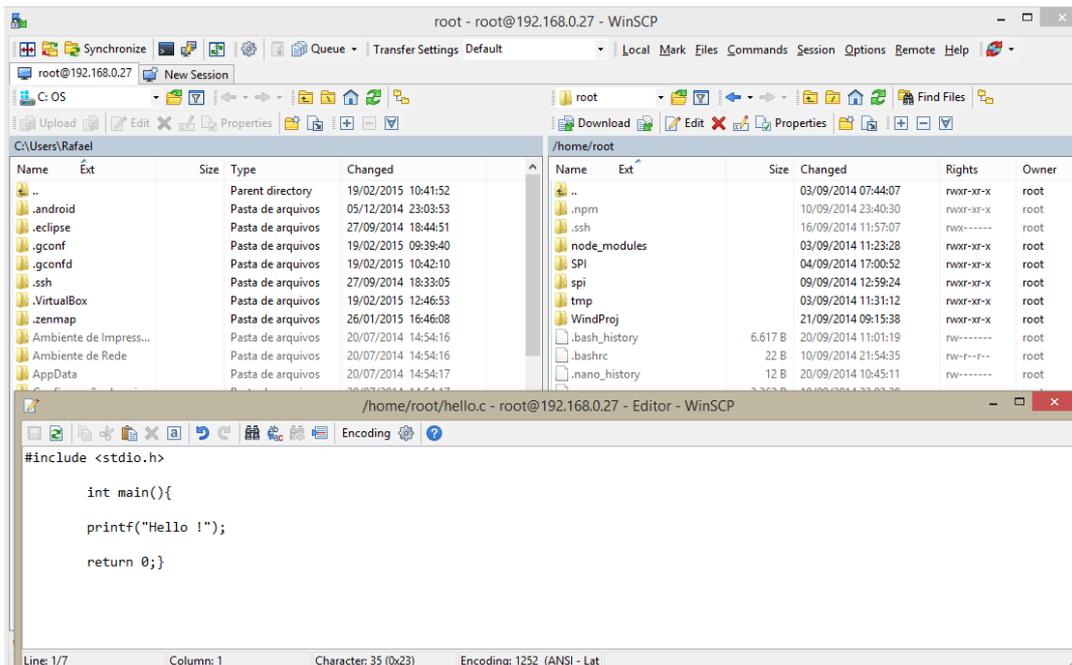
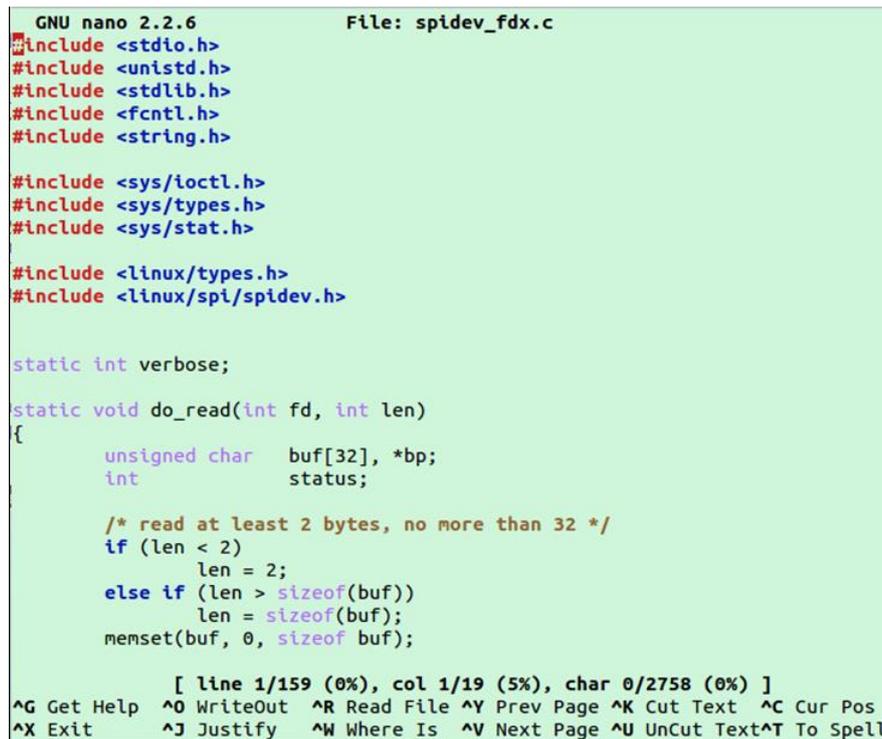


Figura 22: Interface do programa WinSCP no Windows em comunicação com a Galileo.

A ferramenta SCP também está disponível em linha de comando no sistema Linux, e pode ser utilizada a partir das seguintes linhas de comando, sendo possível então uma transferência bidirecional:

- scp /ArquivoFonte usuário@servidor:/diretório/ArquivoAlvo;
- scp usuário@servidor:/diretório/ArquivoFonte /ArquivoAlvo;

Ferramentas de edição de texto são de extrema importância, principalmente quando se deseja programar, criar *scripts*, *makefiles* ou simplesmente visualizar um arquivo de texto. Quando se deseja fazer operações em linha de comando é necessário um *software* de edição de texto que funcione através do terminal, seja na própria máquina ou em um servidor remoto, para essa função usa-se o Nano, um editor de texto leve e gratuito, com interface mostrada na Figura 23.



```

GNU nano 2.2.6          File: spidev_fdx.c
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <fcntl.h>
#include <string.h>

#include <sys/ioctl.h>
#include <sys/types.h>
#include <sys/stat.h>

#include <linux/types.h>
#include <linux/spi/spidev.h>

static int verbose;

static void do_read(int fd, int len)
{
    unsigned char    buf[32], *bp;
    int              status;

    /* read at least 2 bytes, no more than 32 */
    if (len < 2)
        len = 2;
    else if (len > sizeof(buf))
        len = sizeof(buf);
    memset(buf, 0, sizeof buf);

    [ line 1/159 (0%), col 1/19 (5%), char 0/2758 (0%) ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell

```

Figura 23: Nano editor de texto orientado ao terminal.

Outras ferramentas úteis são acessíveis pelos seguintes comandos em terminal:

- top - Comando que exibe informações sobre o sistema Linux, processos em operação e recursos do sistema.
- df ó Mostra o espaço livre do sistema de arquivos.
- du ó Exibe o tamanho de arquivos e diretórios.
- mkdir ó Cria um diretório.
- cp ó Ferramenta de copia de arquivos.
- rm ó Remove arquivos e diretórios.
- kill ó Finaliza processos.
- pwd ó Indica em qual diretório se está fazendo acesso no momento.
- cd ó Permite a navegação entre os diretórios.
- mv ó Move arquivos e diretórios.

- ls ó Informa o conteúdo de um diretório.
- chmod ó Útil para alterar permissões de arquivos quanto a leitura, escrita e execução.
- tar ó Ferramenta de compactação e extração de arquivos do tipo TAR.
- cat ó Utilizando para concatenar arquivos, útil também para visualização rápida do conteúdo de arquivos.
- sudo ó Dá permissão de super-usuário para uma operação seguinte.
- echo ó permite exibir textos em uma arquivo ou na saída padrão.
- shutdown, poweroff ó Encerram o sistema.

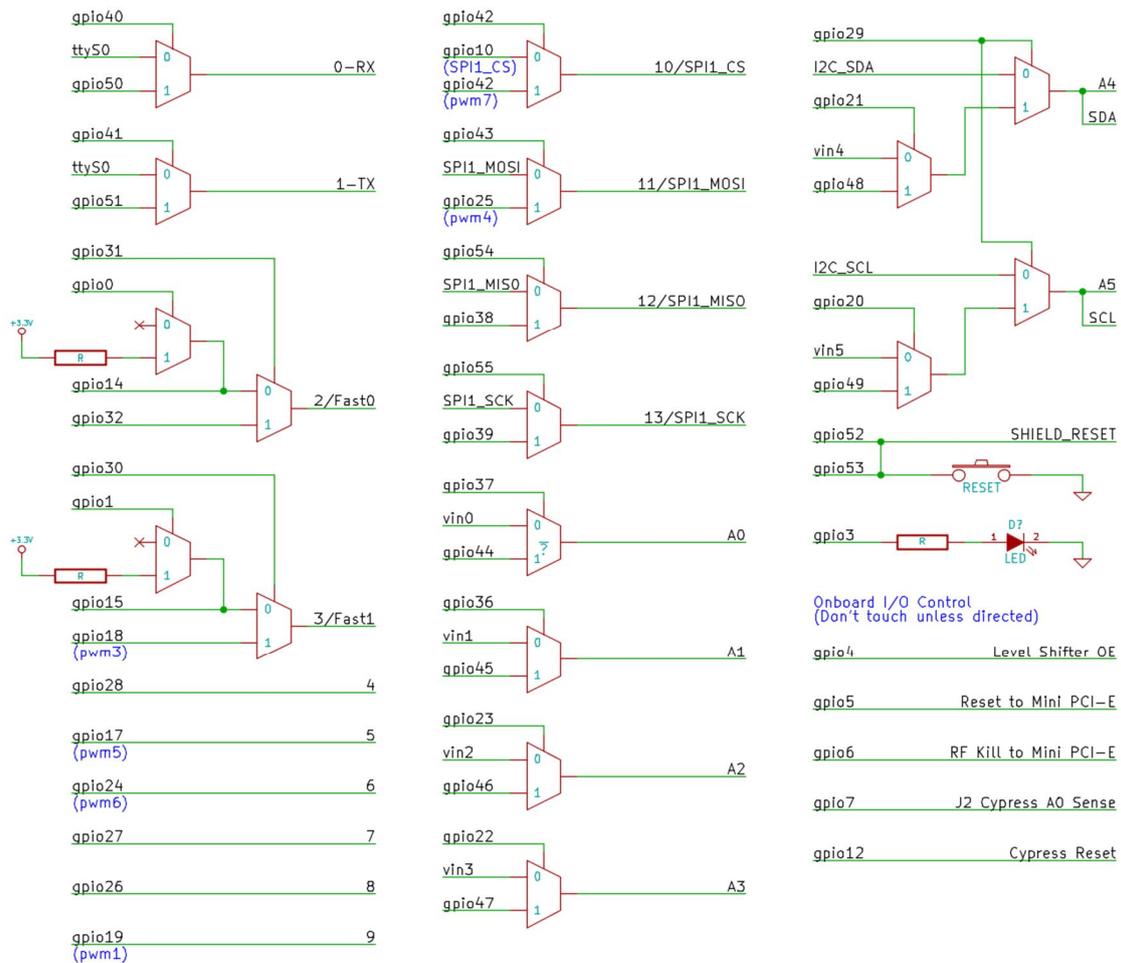
3.1.2.1 DISPOSITIVOS (DEVICES) EM UNIX

Em sistemas baseados em Unix, como o Yocto e o Linux, arquivos de dispositivos ou arquivos especiais são interfaces para *driver* de dispositivos (programa de computador que controla e opera dispositivos físicos conectados a ele) que aparecem no sistema como um arquivo comum (Robbins, 2001).

Arquivos de dispositivos também provêm uma interface simples para dispositivos periféricos como impressoras, portas seriais e portas de uso geral, de suma importância para este projeto.

A placa Intel Galileo, possui acesso às portas através de GPIO (*General Purpose Input/Output* ó Portas de entrada e saída de propósito geral), são portas programáveis de entrada e saída de sinais ou dados, utilizadas para promover interface com periféricos e microcontroladores.

Dessa forma, através do sistema de arquivos é possível fazer o controle das portas de entrada e saída da placa Galileo, tendo conhecimento do diagrama de conexões dos pinos, conforme apresentado na Figura 24, por meio dos procedimentos corretos torna-se simples a manipulação das portas físicas da placa pelo sistema operacional.



16

Figura 24: Diagrama de conexões de pinos GPIO da Galileo.

Pelo do seguinte comando é possível se obter informações sobre o sistema GPIO e se as portas estão alocadas no módulo: `cat /sys/kernel/debug/gpio;` como mostra na Figura 25, a seguir.

¹⁶ http://www.malinov.com/_/rsrc/1383072016958/Home/sergey-s-blog/intelgalileo-programminggpiofromlinux/Galileo_Linux_IO.png?height=346&width=400. Acesso em março 2015.

```

r@R2D2: ~
root@quark012fa0:~/spi# clear

root@quark012fa0:~/spi# cat /sys/kernel/debug/gpio
GPIOs 0-1, platform/sch_gpio.2398, sch_gpio_core:

GPIOs 2-7, platform/sch_gpio.2398, sch_gpio_resume:
gpio-2  (cy8c9540a-int-s3  ) in hi
gpio-7  (cy8c9540a-a0         ) in hi

GPIOs 8-15, intel_qrk_gip_gpio:
gpio-8  (SPI_CS                ) out hi
gpio-10 (SPI_CS                ) out hi
gpio-13 (cy8c9540a-int-s0     ) in hi

GPIOs 16-55, cy8c9540a, can sleep:
root@quark012fa0:~/spi#

```

Figura 25: Informações sobre o sistema GPIO, através dos arquivos de dispositivos.

Para que uma porta possa ser controlada pelo sistema ela deve ser exportada assim como o código exemplo: `echo -n "27" > /sys/class/gpio/export`; onde se escreve no arquivo `export` o valor da porta GPIO a ser controlada, no caso, 27. Quando essa operação é feita com sucesso um diretório correspondendo à porta irá aparecer no sistema, no caso em: `/sys/class/gpio/gpio27`, uma vez terminado o trabalho a exportação da porta deverá ser desfeita, por meio do comando: `echo -n "27" > /sys/class/gpio/unexport`.

Quando selecionada a porta fica acessível a configurações, sendo possível indicar se esta é de entrada ou saída através do comando: `echo -n "out" > /sys/class/gpio/gpio27/direction`; configurando a porta para entrada (*in*) e para saída (*out*).

Se a porta foi configurada como entrada, seu valor pode ser obtido pela leitura do arquivo correspondente a ela, pelo comando: `cat /sys/class/gpio/gpio27/value`; por exemplo.

Caso a porta tenha sido configurada como saída, pode-se controlar o nível lógico na mesma com: `echo -n "x" > /sys/class/gpio/gpio27/value`; onde `0` pode ser zero ou um, para nível lógico baixo ou alto, respectivamente.

4 METODOLOGIA

4.1 ARQUITETURA E CONDICIONAMENTO DE SINAIS

Estudou-se, como mostrado anteriormente, a teoria envolvida para a construção do sistema, e os complementos necessários à placa de desenvolvimento de forma a atender os requerimentos de projeto. Esses que basicamente compreendem a geração de sinais pulsados para o transdutor emissor, e aquisição dos sinais recebidos no transdutor receptor.

A frequência do sinal a ser transmitido é uma característica construtiva dos transdutores, esses funcionam como filtros passa-faixa, com frequência central em 40 kHz e largura de banda de 2 kHz, fez-se um estudo em laboratório e encontrou-se que, para os transdutores disponíveis, a frequência ótima era de 40,5 kHz, apresentando máxima amplitude.

Para esse experimento utilizou-se um gerador de sinais e um osciloscópio ligados diretamente nos transdutores, distando 10 cm, com ausência de fluxos de vento, resultando nas seguintes ondas mostradas na Figura 26, onde a onda superior é o sinal gerado e a onda inferior o sinal recebido. Pode-se observar que os transdutores funcionam de fato como filtros de frequência, visto que o sinal elétrico no transdutor emissor é uma onda quadrada, que é composta pela soma de diversos senos, segundo a teoria de séries de Fourier, e o sinal recebido é um seno perfeito com frequência de 40,5 kHz.

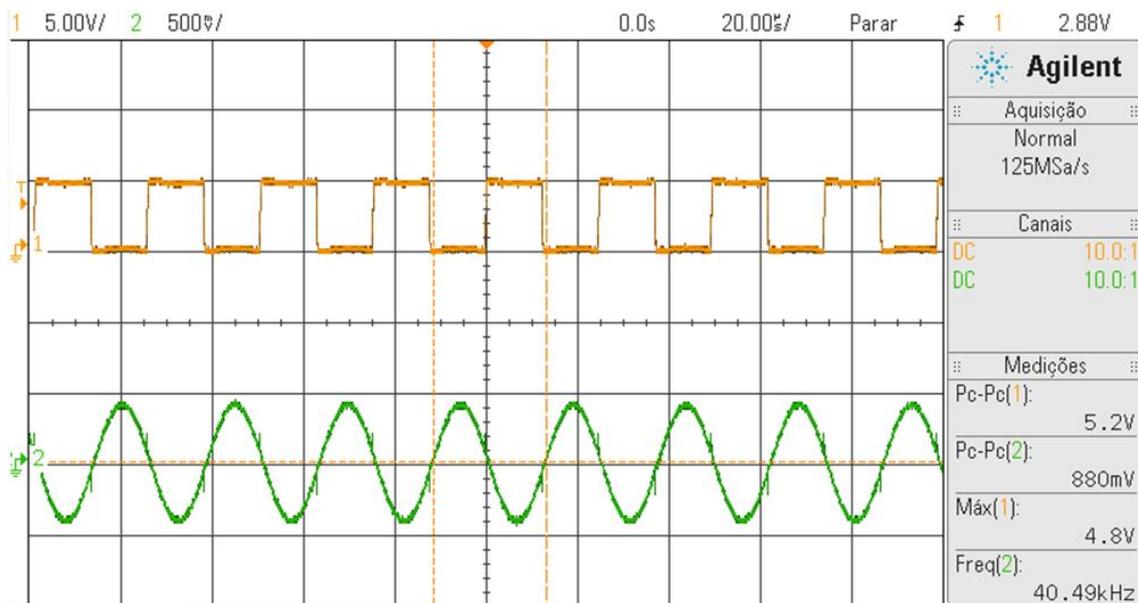


Figura 26: Resultado experimental da caracterização dos transdutores.

A partir deste ponto, realizou-se o estudo de diversas arquiteturas para o condicionamento de sinais, visto que se deve tratar corretamente os sinais que serão transmitidos e os sinais que serão recebidos, de forma a adquirir ambos para o processamento. O esquema simplificado da arquitetura utilizada pode ser visto na Figura 27.

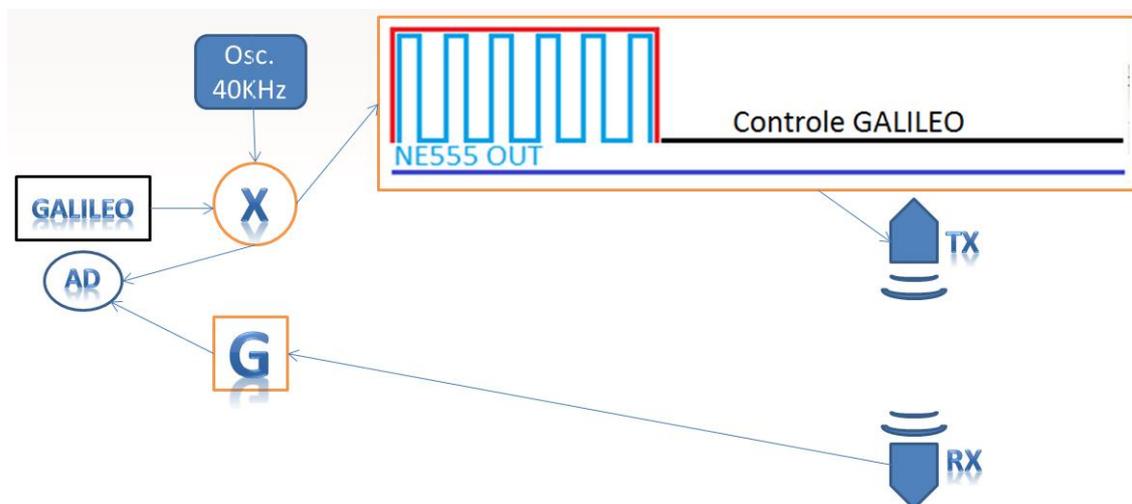


Figura 27: Arquitetura de tratamento de sinais.

Utilizou-se um CI NE555, no modo astável, para gerar um sinal de onda quadrada periódico de frequência 40,5 kHz e contínuo, a partir desta informação e do *datasheet* do componente se fez os cálculos das resistências e capacitâncias necessárias, assim como mostra a Figura 28, este que pode ser habilitado e desabilitado pela placa Galileo, através de portas lógicas, representado na Figura 27 como um multiplicador, o

o sinal resultante passa por um circuito *seguidor de emissor*, apenas como um estágio de ganho de corrente, mostrado na Figura 29, e em seguida é transmitido pelo transdutor TX.

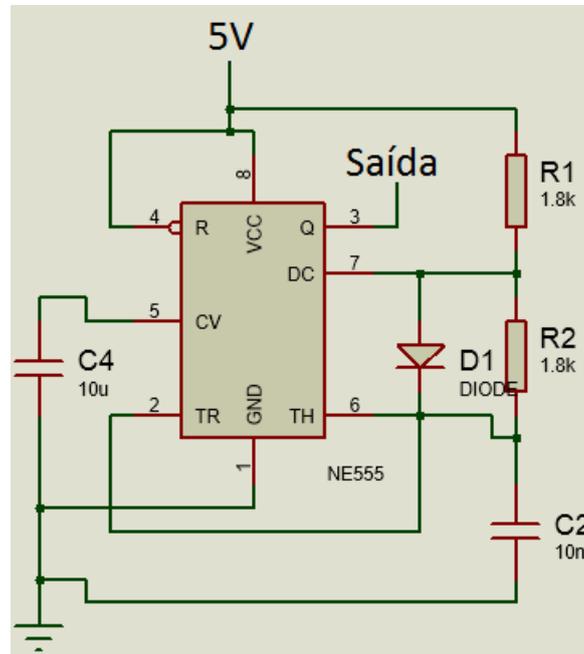


Figura 28: Oscilador projetado com o CI NE555.

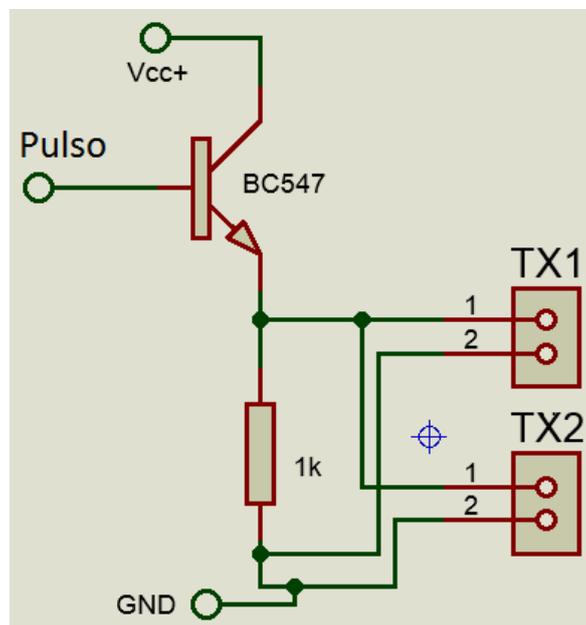


Figura 29: Estágio de saída, *seguidor de emissor*.

O sinal recebido pelo transdutor RX é amplificado por um CI TL082 e condicionado a uma faixa máxima de amplitudes de zero a cinco volts, possibilitando a aquisição a partir do circuito integrado adc108s102, de 10 bits de resolução, frequência de aquisição de até 1 MSPS e comunicação SPI, fornecido como amostra grátis pela

empresa *Texas Instruments*, através de um filtro passa baixas sugerido no *datasheet* do CI.

O sinal recebido pelo transdutor RX é uma onda senoidal de amplitude de 800 mV, para uma distancia entre os transdutores de 10 cm, como visto na Figura 26, com média zero, ou seja, contém regiões de valores negativos, o que não pode ser adquirido pelo *adc108s102*. Para solucionar esse problema, utiliza-se a configuração da Figura 30, de onde o sinal de média nula, vindo do terminal RX1 em paralelo com uma resistência de 100 Ω , como especifica o *datasheet* do transdutor, é acrescentado a um nível DC de 2,5 V através de um capacitor de acoplamento, regido pelo divisor resistivo, em seguida esse sinal é amplificado por uma configuração de amplificador não inversor com ganho regulado por um potenciômetro, de 6,6 até 106,6 V/V, ao passar por esse processo o sinal estará condicionado para a aquisição, com amplitude de no máximo um volt, limitada pelo amplificador operacional TL082, com alimentação simples onde V_{cc+} é 5 V.

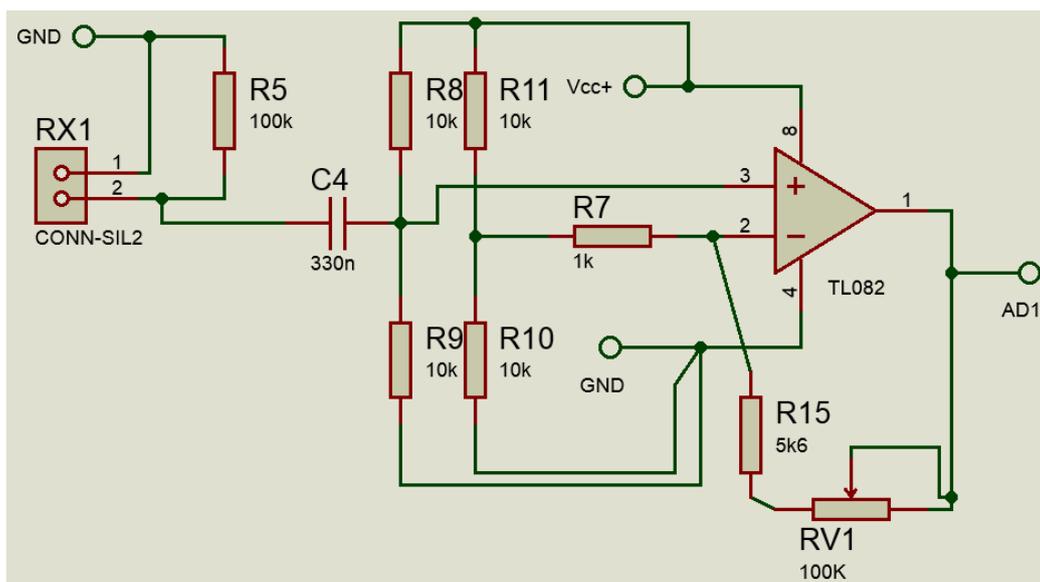


Figura 30: Esquemático do tratamento de sinais.

4.2 CONFEÇÃO DAS PLACAS DE CIRCUITO IMPRESSO

Foram confeccionadas duas placas de circuito impresso, uma para a utilização do CI *adc108s102*, cujo encapsulamento (TSSOP) não permite sua utilização em placas *protoboard*, e a placa definitiva do sistema de tratamento de sinais. O pacote *softwares* utilizado para esse propósito foi o Proteus 8, através da construção de um esquemático no ISIS, Figura 31, interligando e simulando todos os componentes e barramentos, se

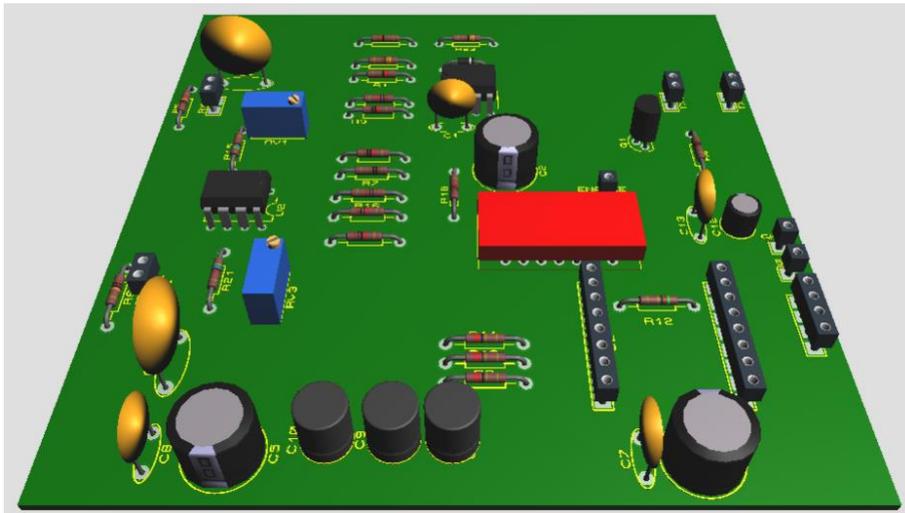


Figura 33: Pré-visualização automática tridimensional da placa.

Uma vez terminada a construção do circuito impresso, através do posicionamento e conexão dos componentes, exporta-se o projeto do circuito para o formato Gerber, o qual é utilizado pela ferramenta de fresa, para esse fim foi utilizada uma fresadora LPKF PROTOMAT S43, a placa resultante desse processo é apresentada na Figura 34.

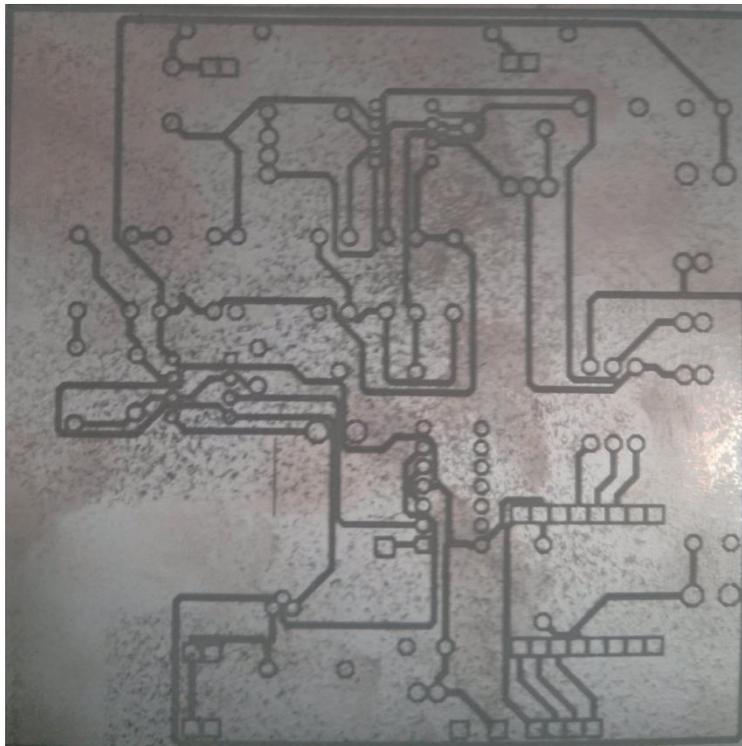


Figura 34: Placa de fenolite cobreada fresada para o sistema de condicionamento de sinais.

Tendo em mãos a placa fresada e furada, o próximo passo é soldar os componentes manualmente e verificar possíveis erros, trilhas cortadas e continuidades. O resultado dessa etapa é mostrado na Figura 35.

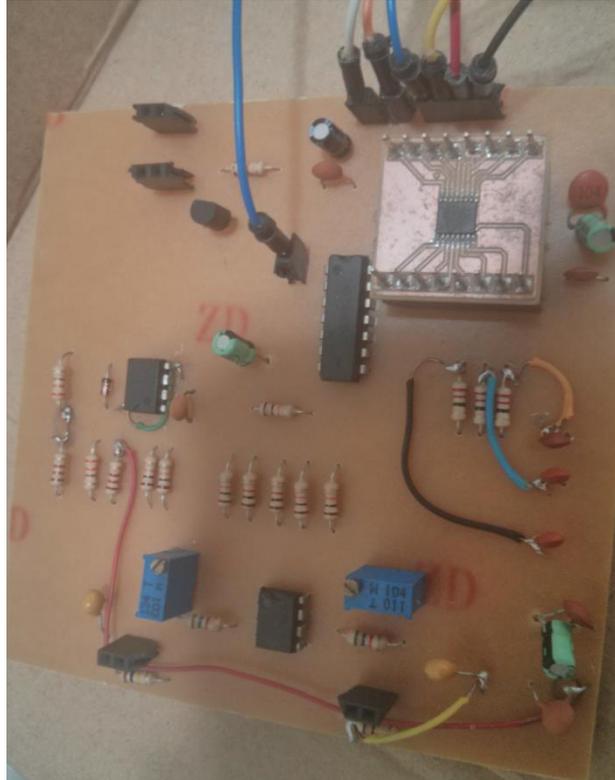


Figura 35: Placa com os componentes soldados.

4.3 ESPECIFICAÇÕES DO SISTEMA EMBARCADO PARA A MEDIÇÃO DA VELOCIDADE DO VENTO

Esse trabalho tem como objetivo a construção de um sistema embarcado capaz de aferir a velocidade do vento utilizando transdutores ultrassônicos. Composto por um sistema de processamento com *software*, capaz de gerar estímulos aos transdutores e processar sinais resultantes, por meio de um circuito de condicionamento e geração de sinais (*hardware*).

O *hardware* do sistema foi apresentado nas sessões anteriores, composto por transdutores ultrassônicos de material piezoelétrico, são capazes de transformar sinais elétricos em ondas mecânicas que interagem com o meio físico (vento), as quais são captadas também por transdutores ultrassônicos e transformadas em sinais elétricos.

Os sinais elétricos são gerados e adquiridos pela placa de circuito apresentado na Figura 35, a partir dos processos citados anteriormente, onde há um oscilador de frequência fixa, 40,5 kHz, gerando sinais capazes de ser transmitidos pelo transdutor emissor por meio de um estágio *seguidor de emissor*, habilitados ou não por *software*, assim como um sistema de condicionamento de sinais, composto por um circuito de ajuste de nível DC atrelado a um circuito amplificador não inversor e um sistema circuito integrado de amostragem, com tensão de referência de aquisição de 5 V e se comunicando com o *software* via comunicação SPI.

O sistema de condicionamento de sinais é alimentado em 5 V consumindo cerca de 20 mA da placa de desenvolvimento Galileo, a qual é responsável pelo núcleo de processamento e transmissão de dados, nela o *software* do sistema embarcado é programado e processado em um sistema operacional baseado em Unix.

O *software* controla o *hardware* fazendo o uso de GPIO, disposto pela placa de desenvolvimento Galileo aproveitando as características dos arquivos de dispositivos (*Device Files*) do sistema operacional, um *software*, programado em C, opera a comunicação SPI com a placa de condicionamento, adquire a temperatura ambiente, pela Galileo, e faz o processamento de dados calculando a velocidade do vento, os resultados são salvos em um arquivo texto, constituindo um histórico de medições.

Os resultados de medições e amostras podem ser disponibilizados pela internet, permitindo acesso aos parâmetros de *software* e respostas do sistema por acesso remoto de qualquer lugar do planeta desde que haja conexão com a internet, por meio de SSH como mostrado nas sessões seguintes. A proposta de funcionamento do sistema embarcado é representada pela Figura 36.



Figura 36: Diagrama de funcionamento do sistema embarcado.

5 RESULTADOS

5.1 PLACA DE CONDICIONAMENTO E AQUISIÇÃO

Concluindo os objetivos desse projeto, primeiramente apresenta-se a placa de circuito impresso desenvolvida para geração e condicionamento de sinais, mais detalhadamente.

Como mostrado na arquitetura anteriormente, a placa dispõe de um oscilador criado com um CI NE555, responsável por criar uma onda quadrada contínua de 40,5 kHz, essa onde é habilitada, ou não, para o transdutor emissor, pela placa Galileo. O circuito foi desenvolvido para ser suprido por uma alimentação simples de 5 volts e baixo consumo, de maneira que sua corrente pode ser suprida pela placa Galileo, para utilização de dois pares de transdutores, com dois barramentos dependentes para transdutores emissores, e dois barramentos independentes para transdutores receptores, com condicionamentos de sinais distintos e portas de aquisição no CI ADC108S102 diferenciadas. Na Figura 38 apresenta-se a placa mais detalhadamente em relação a barramentos e componentes.

Quanto à aquisição, disponibilizou-se um barramento para conexão e troca de tal dispositivo, de maneira que tal que fosse facilitada a modificação do sistema aquisição

dada alguma possível quebra de especificação de projeto, ou para aumentar a resolução ou número de amostras por segundo. Para que isso seja possível, basta utilizar outro circuito integrado no barramento mantendo a comunicação através de SPI e conexões dos demais pinos, como se pode observar na Figura 37.

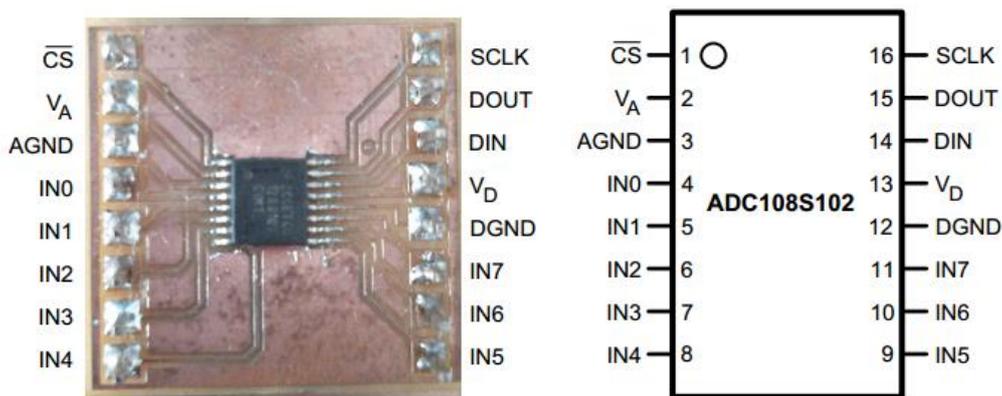


Figura 37: Conexões do circuito de aquisição de dados com comunicação SPI.

A placa foi desenvolvida para permitir seu uso em *protoboard*, tendo dimensões de dois por dois centímetros e espaçamentos entre pinos compatível, cada pino externo corresponde exatamente aos pinos especificados no manual do CI ADC108S102. Nos pinos V_A e V_D é fornecida tensão de 5 volts, nos pinos AGND e DGND a tensão de terra, somente os pinos IN0, IN1 e IN2 estão conectados a sinais de interesse, em IN0 está o sinal de onda quadrada do transdutor emissor, IN1 a onda resultante do transdutor receptor 1 (RX1) e em IN2 a onda resultante do transdutor receptor 2 (RX2), estando disponível para sinais de teste os demais pinos de aquisição.

A frequência de amostragem no `adc108s102` está relacionada diretamente ao *clock* da comunicação SPI, sendo 500 KSPS para 8 MHz e 1 MSPS para 16 MHz, como se deseja realizar duas aquisições em paralelo, tem-se a frequência de amostragem reduzida pela metade para cada aquisição.

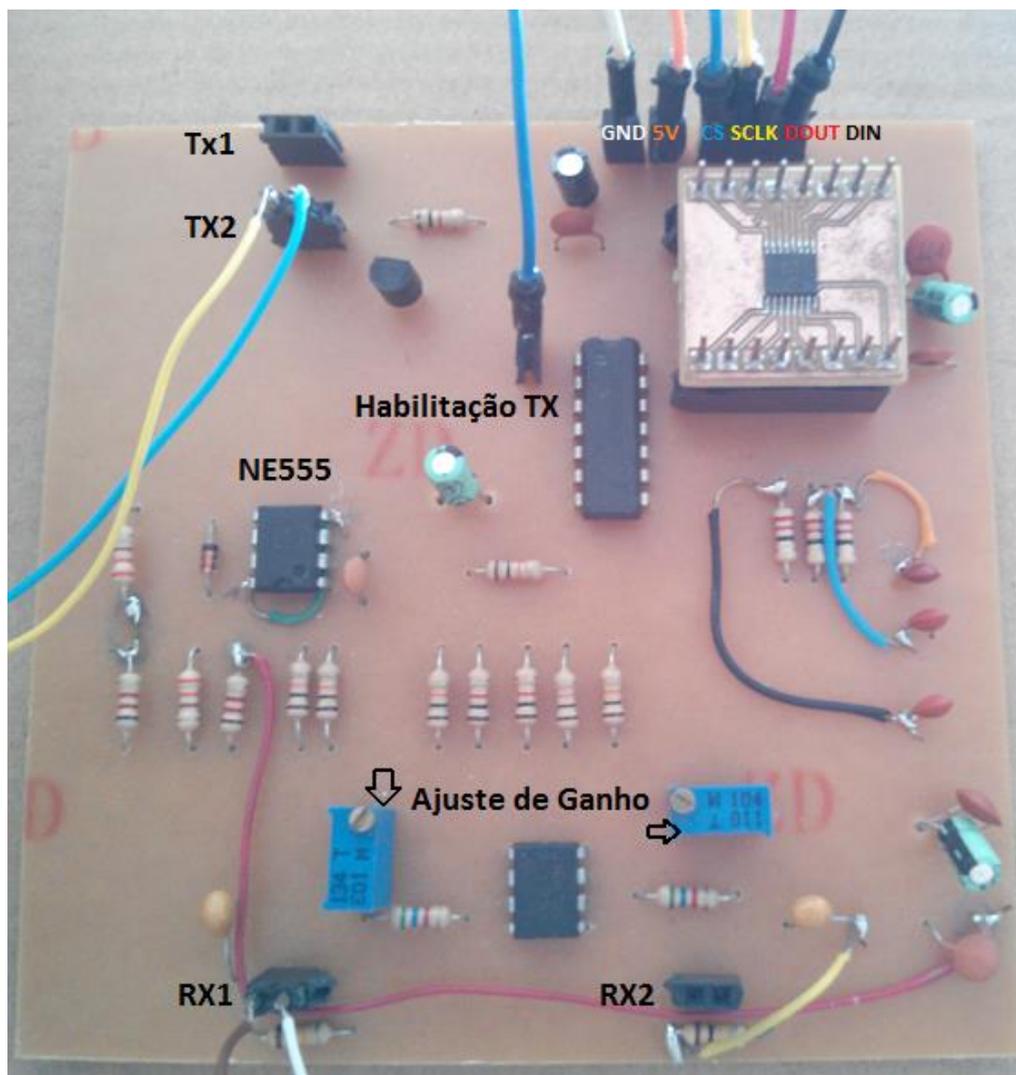


Figura 38: Detalhes da placa de condicionamento e aquisição.

5.2 GALILEO, GPIO, ALGORITMO E PROGRAMAÇÃO

Como citado anteriormente, os pinos externos da placa Galileo podem possuir funcionalidades distintas, sendo necessárias suas corretas configurações para desempenhar determinada função.

No caso, a placa Galileo deve se comunicar com a placa de condicionamento e aquisição através do CI ADC108S102 por meio de comunicação SPI, assim como dispor de um pino de saída de habilitação do sinal TX, para isso deve se utilizar um *script*, da Figura 44, disponível para visualização no anexo A, sempre que iniciada a placa Galileo, configurando corretamente o GPIO, o *script* configura para a comunicação SPI o GPIO 42, 43, 54 e 55, com valores lógicos baixos, observa-se, pela Figura 39, o motivo desses valores, onde os pinos 10, 11, 12 e 13 tornam-se SPI.

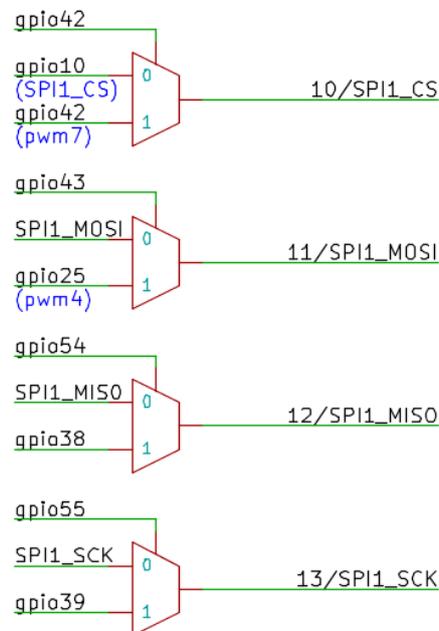


Figura 39: Detalhes do GPIO para a comunicação SPI.

Enquanto que a GPIO 28, que habilita a transmissão das ondas providas do oscilador NE555 para o TX, tem como interface externa o pino 4 como visto na Figura 24.

Depois de configurados os pinos externos e SPI, necessita-se realizar a comunicação, que é estabelecida por funções dispostas na própria biblioteca do sistema operacional (linux/spi/spidev.h) e a documentação está disponível em : <https://www.kernel.org/doc/Documentation/spi/spidev>.

Tendo posse das amostras recebidas por meio da comunicação SPI, da onda quadrada do emissor e do seno do receptor, analisou-se no MATLAB de forma gráfica para verificar o funcionamento do sistema, conforme apresentado na Figura 40.

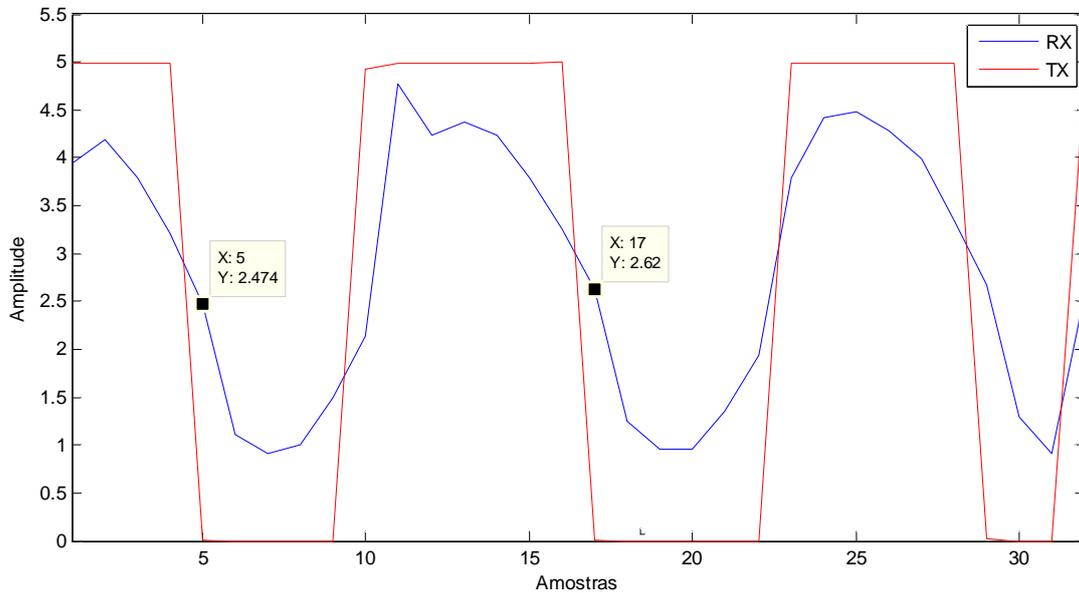


Figura 40: Análise gráfica das amostras colhidas.

Pode-se observar que, na Figura 40, as ondas TX e RX, colhidas e processadas, contêm doze amostras por período, visto que a frequência do sinal é 40,5 kHz, cada período corresponde a 24,7 μ s, tem-se que o tempo entre a aquisição de uma amostra e outra é de 2,06 μ s, resultando numa frequência de amostragem de 486 mil amostras por segundo, satisfazendo o teorema da amostragem de Nyquist, que diz que para a reconstrução de um sinal analógico, deve-se utilizar uma frequência de amostragem de pelo menos duas vezes a frequência fundamental do sinal amostrado.

Através de um algoritmo desenvolvido, Figura 41, para a detecção do cruzamento do sinal com a sua média, esse algoritmo retorna o valor correspondente a quantas amostras os sinais estão defasados, tendo ciência de que o intervalo entre uma amostra e outra é de 2,06 μ s e que um período do sinal contém 12 amostras, calcula-se o tempo de atraso T_{ϕ} , de onde a diferença de fase é $\phi_{\phi} = 2\pi T_{\phi} f_s$, como visto anteriormente.

```

int cruza(float med, int pos, float *array){
    int n = pos;
    int borda = 0;

    if( array[n] >= med){
        borda = 0;
        while(1){
            n++;
            if(array[n] < med)
                return (n + (borda * 6));
        }
    }

    if( array[n] < med){
        borda = 1;
        while(1){
            n++;
            if(array[n] >= med)
                return (n + (borda * 6));
        }
    }

    return -1;
}

```

Figura 41: Algoritmo de detecção do cruzamento da onda com a média.

Por meio de um sensor de temperatura interno à placa Galileo, estima-se a temperatura do ar, calculando então a velocidade do som designada pela equação (4), por meio de uma função que implementa a equação (5) calcula-se a velocidade do vento, considerando fixa a distância entre os transdutores.

Para reduzir possíveis erros, calcula-se a velocidade do vento N vezes, onde N é um valor predefinido com conveniência, calcula-se o valor de maior ocorrência, esse valor é considerado resultado do sistema é salvo em um arquivo de históricos, o qual pode ser útil para estudos posteriores. Escolhendo $N = 20$, para que ocorram vinte medições e um cálculo de estatístico de máxima ocorrência, o sistema é capaz de dar uma resposta em 67 milissegundos, comparado ao tempo de resposta para somente uma medição de 52 milissegundos, conclui-se que, em relação à proporção dos tempos, é vantajosa a realização da operação estatística. O código completo do *software* encontra-se disponível no Anexo B.

```

Tempo de real de execução para N = 20
Temperatura Ambiente: 30 (°C)
Velocidade do vento: 0.626552 (m/s)

real    0m0.067s
user    0m0.010s
sys     0m0.020s
root@quark012fa0:~/WindProj/src#

Tempo de real de execução para N = 1
Temperatura Ambiente: 31 (°C)
Velocidade do vento: 0.628617 (m/s)

real    0m0.052s
user    0m0.000s
sys     0m0.010s
root@quark012fa0:~/WindProj/src#

```

Figura 42: Comparação de tempo de resposta para operações distintas.

5.3 RESULTADOS EXPERIMENTAIS

Fez-se a montagem do sistema de medição no túnel de ventos (1) do Laboratório de Hidráulica da UFPB, como ilustrado na Figura 43, utilizando transdutores preparados em configuração física (5) para tal experimento, de forma a comparar a resposta do sistema embarcado, através da supervisão de um computador pessoal (4), com a resposta forçada por um gerador de sinais (3), mostrado em osciloscópio (2). Durante tal experimento ocorreram imprevistos com o sistema, principalmente com as fontes externas de ruído elétrico, contribuindo como motivação para melhorias físicas e arquitetônicas para uma possível continuação do projeto.



Figura 43: Teste no túnel de ventos do Lens-Ufpb.

6 CONCLUSÃO

O sistema se mostra funcional com um tempo de resposta bastante satisfatório, sabendo do tempo de aquisição entre uma amostra e outra, calcula-se aproximadamente 0,63 m/s de precisão, tendo características interessantes que é a possibilidade da comunicação de dados via internet, fácil acesso remoto, possibilitando reprogramações e atualizações à distância, dado que o sistema pode se localizar em posições de difícil acesso, e toda a maleabilidade que um sistema operacional possa fornecer.

Algumas melhorias podem ainda ser tomadas em relação ao sistema, quanto ao aumento da robustez física da placa de circuito impresso, evitando maus contatos na comunicação com a Galileo, construção de uma caixa protetora para o sistema evitando ruídos elétricos e protegendo fisicamente o sistema.

Porém, o sistema ainda é inicial, de modo que toda a propriedade intelectual desenvolvida nesse projeto servirá como base para possíveis aprimoramentos e estudos para a comunidade de engenharia elétrica do CEAR na Universidade Federal da Paraíba, para a comunidade de engenharia elétrica do CEAR na Universidade Federal da Paraíba.

Algumas propostas, para trabalhos futuros, seriam maiores estudos em túnel de ventos, e a comparação com demais sensores comerciais, a implementação de um sistema mais complexo utilizando dois pares de transdutores, aumentando sua precisão, a implementação de algoritmos de redução das incertezas associadas à medição, como uso da inteligência computacional (redes neurais artificiais, lógica *Fuzzy*, algoritmos genéticos), procedimentos estatísticos de natureza Bayesiana e filtro de Kalman (GOODMAN, 1994) (RAMPONI, 1994).

BIBLIOGRAFIA

FOWLER. Why do cara about measurement? **Instrumentation & Measurement Magazine**, v. 7, p. 38-46, 2004.

GASJKI. **Specification and Design of Embedded Systems**. [S.l.]: [s.n.], 1994.

GOODMAN, W. Y. A. **Data Fusion with Neural Networks**. [S.l.]: IEEE, v. 1, pp. 640-645, 1994.

HAUPTMANN. IEEE Ultrasonic Sensor for Process Industry. **Ultrasonic Symposium**, v. 1, n. 369-378, 2001.

KAIMAL. A continuous Wave Sonic Anemometer-Thermometer. **Journal of Applied Meteorology**, v. 2, p. 156-164, 1963.

MARINHA, C. D. H. D. Marinha do Brasil. **marinha**, 2014. Disponível em: <https://www.mar.mil.br/dhn/chm/meteo/refer/escala_beaufort.htm>. Acesso em: 17 fevereiro 2015.

PÀLLAS-ARENY. **Sensor ans Signal Conditioning**. 2. ed. [S.l.]: [s.n.], 2001.

RAMPONI, R. F. A. **Fuzzy Methods for Multisensor Data Fusion**. [S.l.]: IEEE, v. 43, 1994.

ROBBINS, D. **Common threads: Advanced filesystem implementor's guide**. [S.l.]: [s.n.], 2001.

ROSEN, C. Z. . H. B. V. . & N. R. E. Piezoelectricity. **American Institute of Physics**, Nova Iorque, n. 15, 1992.

STEVENS, W. R. **The Protocols (TCP/IP Illustrated)**. 1. ed. [S.l.]: [s.n.], v. 1, 1993.

TONG. Guidelines for the use of ultrasonic non-invasive metering techniques. **Flow Measurement and Instrumentation**, v. 13, p. 125-142, 2002.

VILLANUEVA. Análise de Incertezas e Fusão de Dados Baseada em Aritmética Fuzzy para Medição da Velocidade do Vento Utilizando Transdutores Ultrassônicos. **Simpósio Brasileiro de Automação Inteligente, VIII SBAI**, Florianópolis, Brasil, 2007b.

WEBSTER. **Measurement, Instrumentation and Sensors Handbook**. [S.l.]: [s.n.], 1999.

YEH. An Intelligent Ultrasonic Flow Meter for Improved Flow Measurement and Flow Calibration Facility. **IEEE Instrumentation and Measurement Tecnology Conference**, v. 3, p. 1741-1746, 2001.

ANEXO A. CONFIGURAÇÃO DAS PORTAS DA PLACA GALILEO

```

gpio_path=/sys/class/gpio

gpio4="out 1"
gpio42="out 0 strong"
gpio43="out 0 strong"
gpio54="out 0 strong"
gpio55="out 0 strong"

gpios="4 42 43 54 55"

setup_gpio() {
    echo "Gpio ${1}: direction: $2, value: $3, drive: $4"
    echo -n $2 > $1/direction
    echo -n $3 > $1/value
    echo -n $4 > $1/drive
}

for i in ${gpios}; do
    gpio=${gpio_path}/gpio${i}
    if [ ! -d ${gpio} ]; then
        echo "Exporting gpio ${i}"
        echo -n ${i} > ${gpio_path}/export
    fi
done

for i in ${gpios}; do
    gpio=${gpio_path}/gpio${i}
    if [ -d ${gpio} ]; then
        gpiovar=gpio${i}
        setup_gpio ${gpio} ${!gpiovar}
    fi
done

#habilita o oscilador externo para TX
echo -n "28" > /sys/class/gpio/export
echo -n "out" > /sys/class/gpio/gpio28/direction
echo -n "1" > /sys/class/gpio/gpio28/value

cat /sys/kernel/debug/gpio

```

Figura 44: Script de configuração das portas da Galileo.


```

        0,4096,0,4096,0,4096,0,4096,
        0,4096,0,4096,0,4096,0,4096,
        0,4096,0,4096,0,4096,0,4096};

uint16_t rx[ARRAY_SIZE(tx)] = {0, };
uint16_t quad[ARRAY_SIZE(tx)/2] = {0, };
uint16_t seno[ARRAY_SIZE(tx)/2] = {0, };

struct spi_ioc_transfer tr = {
    .tx_buf = (unsigned long)tx,
    .rx_buf = (unsigned long)rx,
    .len = 2*ARRAY_SIZE(tx),
    .delay_usecs = delay,
    .speed_hz = speed,
    .bits_per_word = bits,
};

ret = ioctl(fd, SPI_IOC_MESSAGE(1), &tr);
if (ret < 1)
    pabort("can't send spi message");

for (ret = 0; ret < ARRAY_SIZE(tx); ret++) {
    seno[n]=rx[ret];
    quad[n]=rx[ret+1];
    n++;
    ret++;
}

for (ret = 0; ret < ARRAY_SIZE(tx)/2; ret++){
    s[ret] = (float)seno[ret]*5/2048;
    q[ret] = (float)quad[ret]*5/2048;
    //printf("%f %f \n", s[ret], q[ret]);
}
}

double media(float *array, int tam){

    float media = 0;
    int n;

    for(n=0; n < tam; n++)
        media = media + array[n];

return media/tam;
}

int cruza(float med, int pos, float *array){

    int n = pos;
    int borda = 0;

    if( array[n] >= med){
        borda = 0;
        while(1){

```

```

        n++;
        if(array[n] < med)
            return (n + (borda * 6));
    }
}

if( array[n] < med){
    borda = 1;
    while(1){
        n++;
        if(array[n] >= med)
            return (n + (borda * 6));
    }
}

return -1;
}

double means(int temperatura){

    int c1 = 0, c2 = 0, dif = 0, f = 0;
    float m1 = 0, m2 = 0, temp = 0;
    double theta = 0;
    double wind = 0;

    f = temperatura;

    m1 = media(s, 32);
    m2 = media(q, 32);

    c1 = cruza(m1, 0, s);
    c2 = cruza(m2, 0, q);

    dif = abs(c2 - c1);
    f = f + 273;
    temp = sqrt(f);

    theta = (double)dif*0.00000206 + 0.00000103 +
0.1/(20.074*temp);
    wind = (double)0.5*(0.1/(theta)-20.074*temp);

return wind;
}

int conta_elem (float vet[], int faixa){
    int x;
    register int i, j;
    int flag;

    x = 0;
    flag = 0;

    for (i = 0, j = i + 1; i < (faixa - 1); i ++, j++){
        if (vet[i] == vet[j])

```

```

        {
            if (flag == 0)
            {
                x++;
                flag = 1;
            }
        }
        else
            flag = 0;
    }
    return (x);
}

float *moda (float vet[], int faixa){
    int n, flag;
    float *moda;
    register int i, j, k;

    n = conta_elem (vet, faixa); // Retorna a quantidade de elementos
    repetidos no vetor.

    moda = (float *) malloc (n * sizeof (float) );

    /* Este vetor com alocação dinamica pode
    ser retornado pois ele só é desalocado
    quando se usa o comando free(); */
    k = flag = 0;

    for (i = 0, j = i + 1; i < (faixa - 1); i++, j++)
    {
        if (vet[i] == vet[j]){
            if (flag == 0)
            {
                moda[k++] = vet[j];
                flag = 1;
            }
        }
        else
            flag = 0;
    }

    return moda;
}

int main(int argc, char *argv[])
{
    int ret = 0, f = 0, n = 0, N = 2;
    int fd;
    float wind[N];
    float *medwind = NULL;

    FILE * pFile;

```

```

pFile = fopen ("/home/root/WindProj/hist/historico.txt","a");
if(pFile==NULL)
printf("Erro ao abrir arquivo. \n");

FILE * pF;

pF =
fopen("/sys/bus/iio/devices/iio\:device0/in_temp0_raw","r"); //sensor
de temperatura
if(pF==NULL)
printf("Erro ao abrir arquivo. \n");

fscanf(pF,"%d", &f);
f = f/4; // valor da temperatura em Celsius.

fd = open(device, O_RDWR);
if (fd < 0)
    pabort("can't open device");

/*
 * spi mode
 */
ret = ioctl(fd, SPI_IOC_WR_MODE, &mode);
if (ret == -1)
    pabort("can't set spi mode");

ret = ioctl(fd, SPI_IOC_RD_MODE, &mode);
if (ret == -1)
    pabort("can't get spi mode");

/*
 * bits per word
 */
ret = ioctl(fd, SPI_IOC_WR_BITS_PER_WORD, &bits);
if (ret == -1)
    pabort("can't set bits per word");

ret = ioctl(fd, SPI_IOC_RD_BITS_PER_WORD, &bits);
if (ret == -1)
    pabort("can't get bits per word");

/*
 * max speed hz
 */
ret = ioctl(fd, SPI_IOC_WR_MAX_SPEED_HZ, &speed);
if (ret == -1)
    pabort("can't set max speed hz");

ret = ioctl(fd, SPI_IOC_RD_MAX_SPEED_HZ, &speed);

```

```
if (ret == -1)
    pabort("can't get max speed hz");

fprintf(pFile,"T (Â°C  V (m/s) \n");

    for(n = 0; n <= N; n++){
        transfer(fd);
        wind[n] =fabs(means(f));
    }
medwind = moda(wind, N);

fprintf(pFile," %d %f\n", f,*medwind);

system("clear");
printf("Temperatura Ambiente: %d (C)\n", f );
printf("Velocidade do vento: %f (m/s)\n\n", *medwind);

close(fd);
fclose(pF);
fclose(pFile);

return ret;
}
```