Integração entre sistema embarcado e gráfico para análise de trajeto



Revista Eletrônica Multidisciplinar
FACEAR
ISSN: 2316-2317

Kim Batista[1]
Orientador Fabiano Nezello[1], Co-orientador Leandro Vasconcelos dos Reis[1]

[1] *Faculdade Educacional Araucária*

RESUMO

É possível perceber que nas últimas duas décadas a capacidade de processamento dos computadores aumentou exponencialmente comparando com os anos anteriores, o tamanho dos componentes reduziu drasticamente, a opção de tecnologias alternativas tornou-se enorme, e a velocidade do surgimento de inovações tecnológicas tornou-se praticamente impossível de se acompanhar. Tão rápido quanto o surgimento de novos dispositivos, é o surgimento de necessidades modernas que a sociedade cria, sempre baseada em invenções anteriores. Sucintamente, acaba-se percebendo que uma inovação leva a outra, e cada vez mais as soluções tornam-se complementares ou base de novas criações. Em um ciclo evolutivo constante, a criação de novas tecnologias ou sistemas pode trazer mais conforto aos usuários, controle sobre suas atividades, automatização de processos, estímulo a descobertas ou mesmo o puro lazer, diversão e interação social. Totalmente ligado a estes fatores, está o desenvolvimento de sistemas integrados, de diferentes tecnologias e plataformas. Um universo inteiro de coisas a serem descobertas e criadas sobre a área de programação demanda atenção dos desenvolvedores para a integração entre sistemas implementados sobre tecnologia embarcada ou alto-nível. O artigo apresentará uma ideia de sistema embarcado capaz de coletar informações de diversos sensores, e integrar com um sistema gráfico para visualização simples e simultânea dos dados coletados para fins de análise.

Palavras chave: Programação de alto-nível. Sistemas embarcados, Integração de sistemas.

ABSTRACT

Over the last two decades, the processing capacity of the computers has increased exponentially comparing to the previous years, the size of the components has shrunk drastically, the variety of alternative technologies has become huge, and the speed of innovations became nearly impossible to follow up. As fast as the creation of new devices is our unconscious creation of modern necessities, always based on previous inventions. We realize that one innovation leads to another, and solutions fulfill each other or turn into inspiration of new developments. In a constant evolution cycle, the creation of new technologies or systems can bring more comfort to the users, as well as control over their activities, process improvements, inspiration for discovering or even pure joy and social interaction. The development of integrated systems in different technologies and platforms is completely related to these factors. An entire universe of new things opens to be discovered through programming, and this requires attention of developers over integration of systems codified on embedded or high-level languages. This article presents the idea of an embedded system prototype to gather information of several sensors integrated to graphical software to simply visualize simultaneously the collected data.

Key Words: High-level programming. Embedded systems. Integrated systems.

1 INTRODUÇÃO

Quase que diariamente somos surpreendidos por novidades tecnológicas, novos equipamentos, máquinas, sistemas, exemplos de inovações e da habilidade de criação humana. Percebe-se ultimamente a tendência do surgimento de dispositivos capazes de integrar percepções do ambiente ou de movimentos através de sensores com outros tipos de sistemas para o deslumbre daqueles que o utilizam.

Dificilmente encontra-se tecnologia de ponta, ou imagina-se como será o futuro sem a integração de sistemas avançados. Também dificilmente encontra-se no Brasil profissionais com a visibilidade de integração de sistemas de diferentes tipos, principalmente que tenham a capacidade de integrar sistemas embarcados com outras plataformas desenvolvidas em linguagem de alto-nível.

Baseada em artigos nacionais e internacionais sobre IMUs (*Inertial Measurement Unit*) esta pesquisa desenvolveu-se a fim de experimentar a criação de um protótipo capaz de funcionar como uma IMU agregada a outros sensores relacionados a navegação, adicionando um método integrado de visualização da informação coletada, mas ainda mantendo o teor acadêmico e seguindo a premissa de baixo custo.

O objetivo deste artigo é apresentar um modelo prático de visualização das informações coletadas por um conjunto de sensores, bem como um protótipo com Arduino capaz de coletar informações de um trajeto e como os dois sistemas se integram.

2. DESENVOLVIMENTO

O referido experimento é composto de dois sistemas distintos. Um sistema embarcado, utilizando como processador o Arduino 2560 conectado a um conjunto de módulos e sensores, e outro sistema gráfico desenvolvido sobre a plataforma *desktop* na linguagem Java e fazendo uso da biblioteca gráfica OpenGL. A figura a seguir traz um diagrama de implantação que permite visualizar o contexto do projeto e a interação entre os dois sistemas citados.

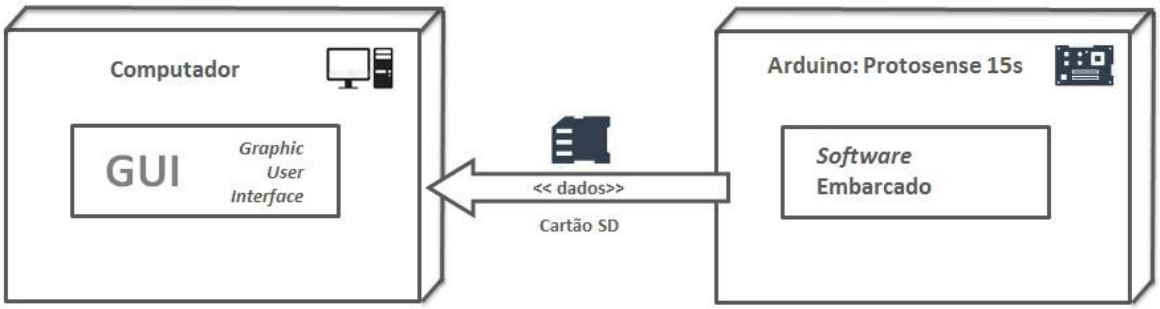


FIGURA 1 - DIAGRAMA DE IMPLANTAÇÃO DO PROJETO
FONTE: CRIADO PELO AUTOR

Um cartão SD utilizado no protótipo armazena as informações coletadas durante um trajeto percorrido, mantendo a sincronia da leitura entre todos os sensores. Em um segundo momento, já finalizado o trajeto, este cartão é inserido no computador, e o programa gráfico carrega o arquivo com o *log* de leituras e o exibe de forma gráfica para o usuário.

Tratando-se primeiramente do sistema embarcado, é necessário compreender os módulos e sensores envolvidos, as versões utilizadas e como estão interligadas no *hardware*. Esta representação será feita através do diagrama de blocos a seguir.

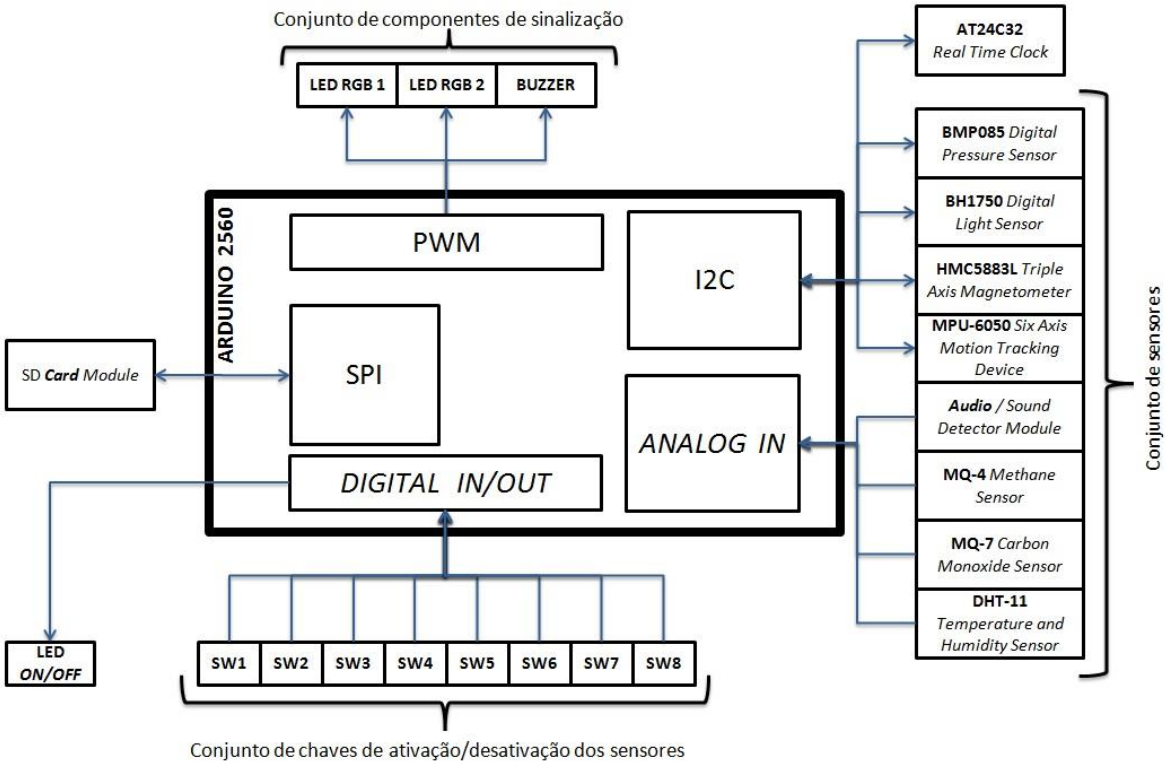


FIGURA 2 - DIAGRAMA DE BLOCOS FUNCIONAIS DO PROTÓTIPO
FONTE: CRIADO PELO AUTOR

Observa-se à direita da imagem anterior, o conjunto de oito módulos sensores anteriormente citado, estando os sensores de pressão atmosférica, luz, magnetômetro e MPU (*Motion Processing Unit*) conectados ao barramento I2C da unidade principal, e os sensores de áudio, gás e temperatura conectados à entrada analógica. Também ao barramento I2C está conectado o módulo RTC, de tempo. Ao barramento digital de I/O, um LED para simbolizar o funcionamento do protótipo e um conjunto de oito chaves de ativação tem como objetivo ligar ou desligar cada um dos sensores individualmente. O módulo de cartão SD, responsável por armazenar as informações coletadas, é o único ligado no barramento SPI. Por fim, um conjunto de dois LEDs RGB e um *buzzer* conectados ao barramento PWM tem por finalidade sinalizar o status do protótipo, informando através de combinações de cores e sons o funcionamento, inicialização ou configuração de tudo o que ocorre durante o tempo de execução do protótipo uma vez que este não dispõe de interface em tempo real com nenhum tipo de display LCD ou conexão com computadores externos. Com módulo MPU seria possível verificar as alterações na rotação do veículo nos 3 eixos, com as identificações *yaw*, *pitch* e *roll*, conforme a próxima imagem.

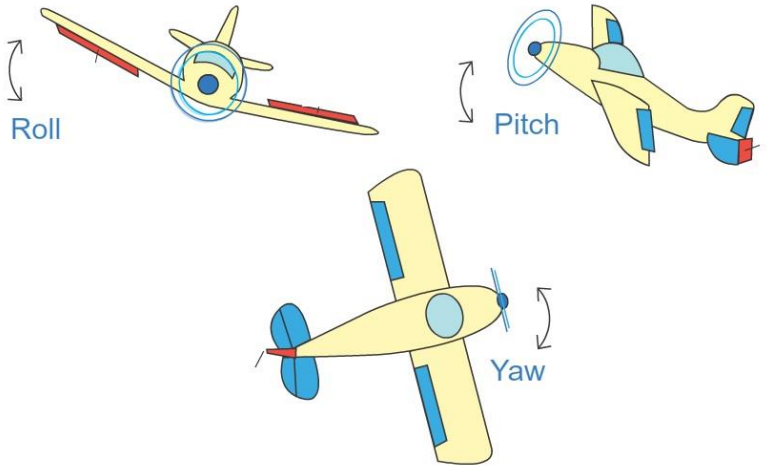


FIGURA 3 - IDENTIFICAÇÃO DOS EIXOS
FONTE: EDITADO PELO AUTOR DE HOW THINGS FLY (2014)

A codificação do software embarcado será baseada na interpretação de todos os sensores conectados ao protótipo, devendo o programa compreender sensores ativos e inativos, coletar e interpretar suas informações, além de ser capaz de sinalizar alterações de comportamento, configurações e falhas encontradas durante a execução do programa de forma autônoma.

Para que o usuário possa compreender possíveis falhas do protótipo, tratando-se de mensagens de sucesso e qualquer tipo de interação com o sistema embarcado, foram adicionados um alto-falante modelo buzzer e dois LEDs multicoloridos, também conhecidos como RGB, que apresentam um conjunto de cores e sons representando a resposta do protótipo. Entre alguns dos parâmetros internos para detecção e sinalização de falhas estão o reconhecimento dos sensores, do cartão, do módulo de hora e a inicialização de todos os dispositivos internos.

Todos os testes foram realizados em ambiente experimental, de modo que foi possível simular todas as condições necessárias para explorar os diferentes tipos de sensores acoplados ao protótipo. Principalmente, foi avaliada a autonomia do sistema embarcado desenvolvido em interpretar corretamente as informações fornecidas pelos sensores conectados, suas sinalizações de falhas e gravação, e capacidade de se auto ajustar de acordo com as configurações do usuário.

Em laboratório, fazendo uso de ar condicionado, diferentes fontes de iluminação e geração de fumaça, foi possível criar cenários simulados com diferentes situações de iluminação, temperatura e umidade, som e presença de gás, além de experimentar diversas posições e orientações para o protótipo verificando e validando o comportamento dos sensores de atitude e direção. Outra sessão de testes foi feita em ambiente natural, à luz do sol e temperatura ambiente. Tanto em cenário simulado quanto em natural, pôde-se, portanto, verificar a aquisição das informações dos sensores de luz, temperatura, gás, acelerômetros, giroscópios, magnetômetro e som. Para a realização dos testes avulsos de cada sensor, a fim de validar o funcionamento de todos os módulos, foi montado um circuito provisório, conforme a próxima imagem.

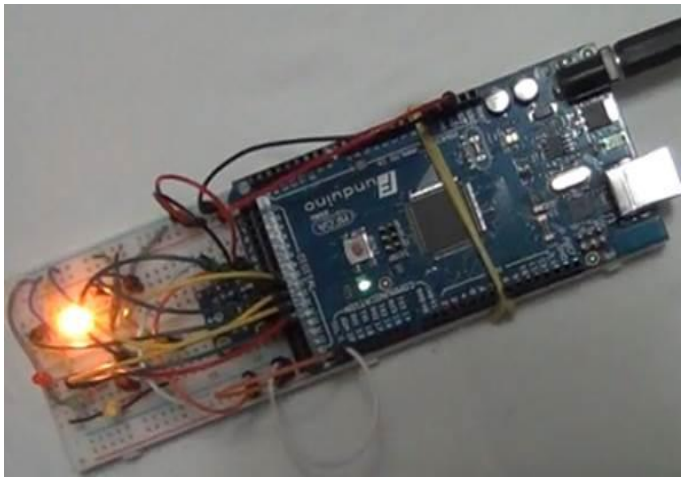


FIGURA 4 - MÓDULO UTILIZADO PARA TESTE DOS SENSORES
FONTE: CRIADO PELO AUTOR

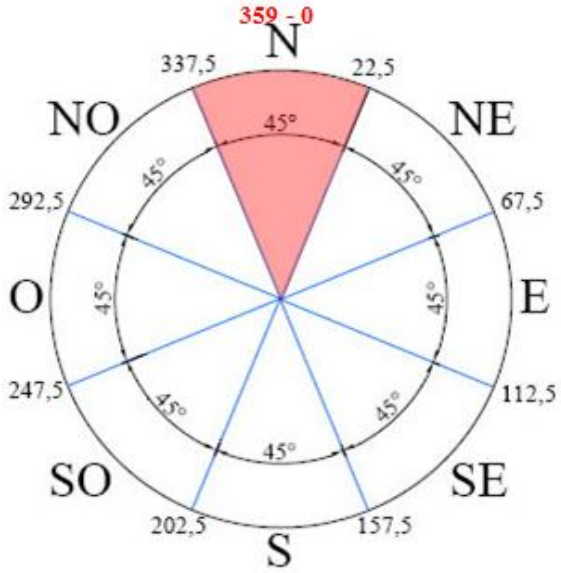
Alguns cenários testados e seus resultados podem ser vistos na próxima tabela. Para o sensor HMC8553L especificamente, houve uma grande dificuldade na interpretação de seus valores. Sendo um componente de precisão relativamente baixa, é esperado que existam ruídos durante a leitura de seus valores. A orientação técnica nestes casos é realizar uma quantidade planejada de leituras no tempo, determinada amostra, e tirar a média dos valores, atenuando a importância do ruído naquela gama de leituras. Ao orientar-se a respeito da escala de valores de cada sensor, observou-se que os valores da tabela a seguir eram coerentes com o esperado para sensores de desenvolvimento de baixo custo.

TABELA 1 - CENÁRIOS DE TESTES E RESULTADOS
FONTE: CRIADO PELO AUTOR

Sensor	Modelo	Cenário	Descrição do cenário	Valor médio*
Temperatura	DHT-11	A	Ao ar livre em um dia de sol.	33°C
		B	Dentro de um refrigerador, após 2 minutos.	12°C
Umidade	DHT-11	C	Ao ar livre em um dia de sol.	41%
		D	Ao ar livre em um dia chuvoso.	58%
		E	Em uma sala com ar condicionado ligado.	29%
Altitude	BMP085	F	Rua João Bettega, nº 5200 CIC, Curitiba-PR, Brasil.	886,02m
		G	Av. das Araucárias, nº 40, CIC, Curitiba-PR, Brasil.	887,11m
		H	Av. das Araucárias, nº 1011, CIC, Curitiba-PR, Brasil. Uma subida com relação ao cenário anterior.	925,74m
Luminosidade	BH1750	I	À noite, em uma sala com a luz apagada.	0 lux
		J	Em uma sala com a janela aberta. Dia ensolarado.	8701 lux
		K	À noite, em uma sala com a luz fluorescente acesa.	4235 lux
		L	Ao ar livre em um dia ensolarado, sob a sombra.	4.867 lux
		M	Ao ar livre em um dia ensolarado, sob direta luz do sol.	43.666 lux
Bússola	HMC5883L	T	Voltado ao norte.	94°
		U	Voltado ao sul.	286°
		V	Voltado ao oeste.	43°
		W	Voltado ao leste.	116°
Microfone	Sound Module	X	Em uma sala silenciosa.	31
		Y	Em uma sala com música alta tocando.	31
		Z	Em meio ao trânsito.	31

No caso do magnetômetro, a média era baseada em 10 valores lidos numa fração de segundo, convertidos para graus radianos. O problema encontrado foi na média dos valores quando apontado para a posição norte, onde o valor em graus radianos transita entre 0 e o limite da circunferência, 359. Nesta situação, valores variantes tão divergentes criam médias em valores intermediários, distantes da realidade da medida. Para ilustrar,

segue a imagem de uma circunferência com seus pontos cardeais associados aos valores em radianos.



LEITURA	VALOR
1	354
2	351
3	355
4	15
5	355
6	7
7	15
8	354
9	8
10	353
MÉDIA	216,7

FIGURA 5 – CIRCUNFERÊNCIA COM GRAUS RADIANOS E PONTOS CARDEAIS
FONTE: CRIADO PELO AUTOR

Foram realizadas 10 leituras para a média, quantidade mínima aceitável de leituras para uma média coerente, baseado em testes com 5, 10, 30, 50 e 100 leituras. A menor quantidade foi escolhida visando o melhor desempenho do dispositivo. Ao efetuar a leitura na região de transição, destacada na circunferência, eram obtidos valores conforme demonstrados na tabela da imagem, chegando a médias irreais. Para resolver tal problema, foi feita uma tratativa especial no programa. A circunferência foi separada em três segmentos independente dos pontos cardeais, mas apenas para melhor precisão da média de leitura. A segmentação foi feita conforme a próxima imagem.

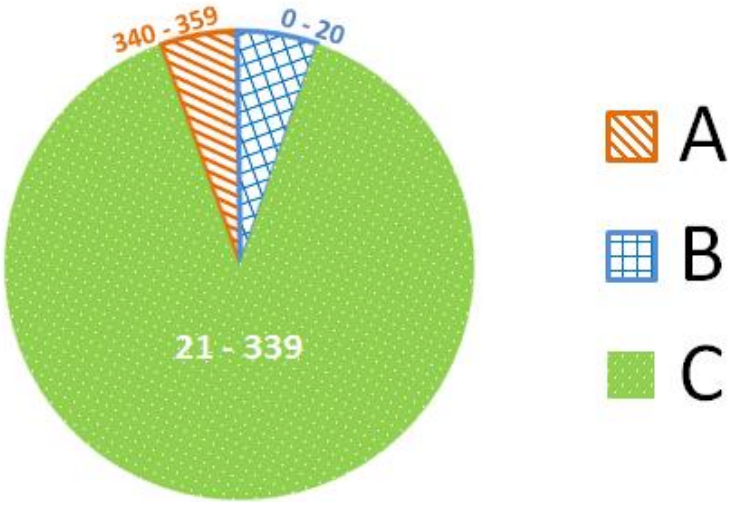


FIGURA 6 – SEGMENTAÇÃO DA CIRCUNFERÊNCIA PARA VALORES DO MAGNETÔMETRO
FONTE: CRIADO PELO AUTOR

Excluindo a região C, que é a zona segura, a solução implica em verificar, entre as 10 leituras, se a região A ou B teve mais valores coletados. Caso a região A tenha a mesma quantidade ou mais leituras do que a região B, a média considera somente a região A, caso contrário, a média é feita sobre a região B. No exemplo de valores citado anteriormente, a região A seria selecionada, e os valores da região B seriam descartados.

Foi possível observar que o sensor magnetômetro apresentou valores satisfatórios quando testado isoladamente, no entanto, não teve bom funcionamento uma vez encaixado no protótipo. O mau funcionamento foi causado pelos campos magnéticos internos do protótipo, considerando que são muitos sensores interligados. Desta forma, o produto final não apresenta os resultados do sensor magnetômetro.

O microfone apresentou o valor constante 31, independente do cenário em que foi submetido, pois a peça adquirida tinha defeito funcional. Os demais sensores mostraram valores coerentes tanto no teste avulso quanto no teste conjunto.

Após a conclusão do software, os componentes e placas precisavam ser montados e interligados eletronicamente de forma simples e compacta. Para viabilizar o que havia sido planejado, foram escolhidas peças de baixo custo e fácil acesso nos comércios locais.

A montagem física do protótipo foi feita baseada em uma caixa plástica de medidas 14,5cm x 9,5cm x 5,5cm. As furações para comportar as chaves, LEDs e botões foram feitas artesanalmente, de modo a manter o baixo custo do projeto. O uso de cola e solda foi evitado, e todas as placas e sensores foram fixados através de espaçadores plásticos, parafusos e conectores, e interligados por fios com plugues em suas extremidades, permitindo a desconexão ou substituição de todas as peças sem danificar o conjunto.

Os sensores de detecção de gases e o de luminosidade foram os de encaixe mais problemático, considerando o tamanho dos dois sensores de gases, aproximadamente três vezes maiores do que os demais, e que exigiam uma região vazada na estrutura da caixa plástica, uma vez que para detectar o gás no ambiente era necessário ter uma área que permitisse o acesso direto ao ambiente externo.

O de luminosidade, embora pequeno, precisava de uma área não necessariamente vazada, mas transparente, que o permitisse medir a intensidade da luz externa. A intenção é que esta região ficasse na parte superior, o que potencializaria o

acesso à luminosidade em qualquer sentido. Algumas imagens da versão final do protótipo são exibidas a seguir.

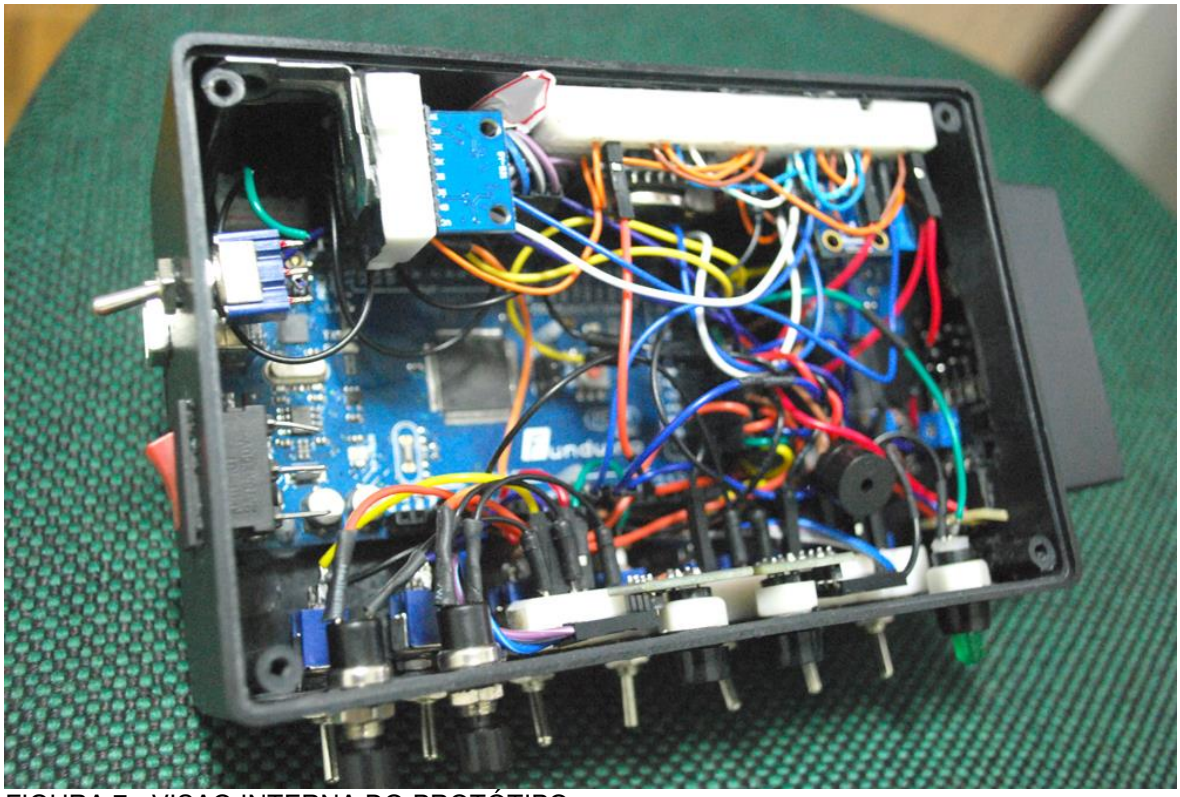


FIGURA 7 - VISAO INTERNA DO PROTÓTIPO
FONTE: CRIADO PELO AUTOR

A maior placa, Arduino, foi alocada em um dos cantos, acompanhada de uma barra de conexão estilo protoboard, chamada daqui em diante de barra de alimentação, com o propósito de expandir o acesso aos pinos de alimentação, +5V e GND (Ground), os quais estariam ligados a todos os módulos. A barra de alimentação básica tornou as conexões organizadas, evitando que muitos fios precisassem unir-se a um único pino, e permitiu a conexão paralela de vários contatos.

Os sensores magnetômetro, acelerômetro e giroscópio precisavam estar alinhados com o solo e posicionados de forma coerente, pois suas medições acompanham o movimento do dispositivo. Não seria possível, portanto, posicioná-los de forma vertical ou invertidos, e seria necessário definir uma frente para o protótipo, a ser alinhada com o indicador *yaw*, anteriormente referido neste documento. O posicionamento destes sensores foi auxiliado por aletas metálicas em forma de L, como pode ser visto na região superior-esquerda da imagem anterior, para mantê-los alinhados e presos a uma das laterais da caixa.

O painel de comandos localizado na lateral esquerda da caixa centraliza as chaves, botões e luzes indicativas do funcionamento do aparelho. Uma caixa plástica foi posicionada na parte superior do protótipo, dedicada a comportar o cartão SD e o sensor de luminosidade, que por fim ficou encostado a um pedaço de acrílico transparente. A versão final em funcionamento pode ser vista na próxima imagem.



FIGURA 8 - VERSAO FINAL DO PROTÓTIPO
FONTE: CRIADO PELO AUTOR

Em posse dos dados coletados no protótipo e salvos no cartão de memória, as informações precisam ser visualizadas de alguma maneira. A seguir será apresentado o processo de desenvolvimento, os empecilhos encontrados e as soluções para a implementação do sistema gráfico.

O desenvolvimento foi iniciado pela exploração de objetos 3D na plataforma OpenGL, que representaria os movimentos de rotação do protótipo nos 3 eixos, atividade tida como a mais desafiadora desta etapa do projeto. Pesquisando bibliotecas gráficas e formatos geométricos conhecidos, foi possível fazer um objeto cúbico movimentar-se no espaço virtual ou girar sobre o próprio eixo. Para tal foi fundamental compreender a estrutura de vértices dos objetos dentro da biblioteca OpenGL. Observou-se que o cubo baseava-se na localização de oito vértices independentes, conforme a próxima imagem:

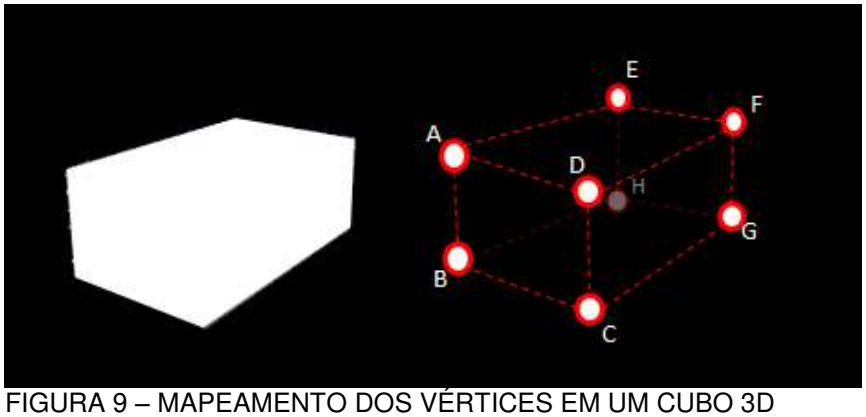


FIGURA 9 – MAPEAMENTO DOS VÉRTICES EM UM CUBO 3D
FONTE: CRIADO PELO AUTOR

Uma vez posicionados os vértices, e o cubo desenhado, era necessário compreender o desenho de cada uma das faces com o objetivo de colori-las, o que permitiria a percepção de profundidade do cubo e a compreensão de movimentos sobre o próprio eixo, pois com apenas uma cor não seria possível perceber a orientação de cada uma das seis faces. A próxima figura ilustra os avanços na pesquisa de objetos 3D:

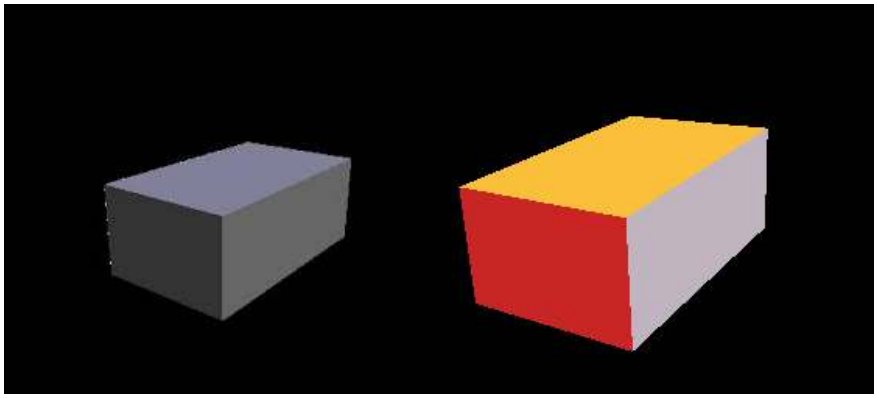


FIGURA 10 – CUBOS 3D COM FACES COLORIDAS
FONTE: CRIADO PELO AUTOR

O cubo à esquerda na figura anterior, com tons de cinza, simula uma iluminação virtual do ambiente, com tons claros representando as luzes, e tons escuros representando mais ou menos sombra. À direita, um cubo com cada face de uma cor diferente. A partir deste resultado, foi possível animar os objetos, criando sequências de valores para o ciclo de desenhos das faces dos cubos. Posteriormente, esta sequência de valores deveria ser importada do arquivo salvo no cartão de memória, originada dos sensores giroscópios (MPU). Com a animação de um cubo de faces coloridas, obteve-se o seguinte resultado:

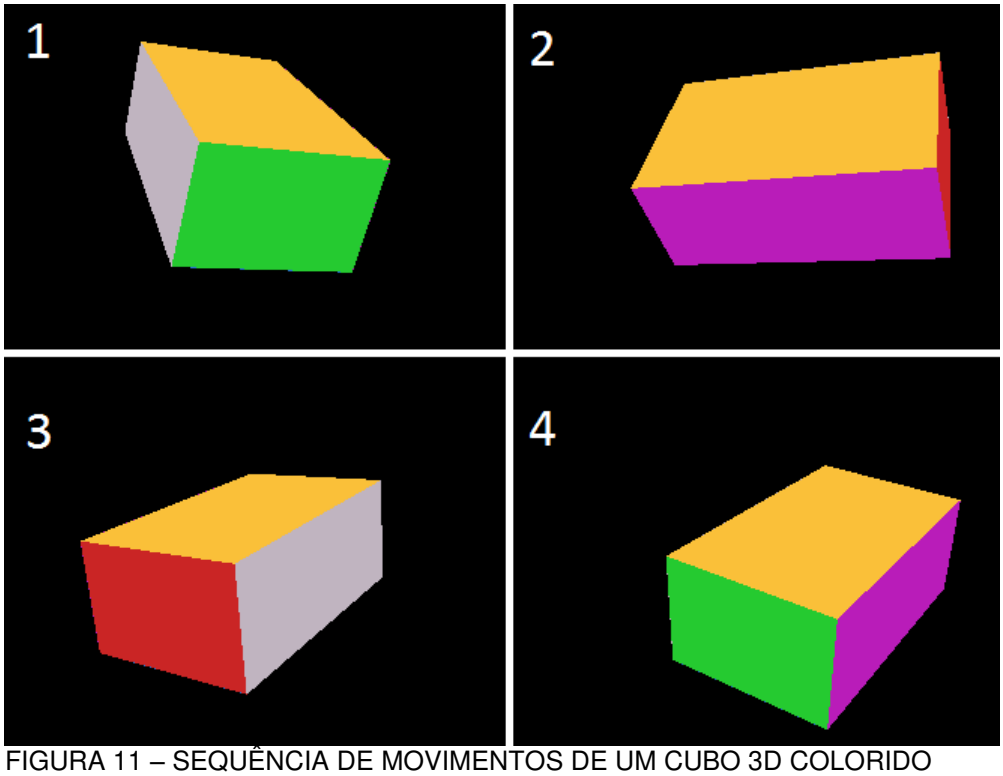


FIGURA 11 – SEQUÊNCIA DE MOVIMENTOS DE UM CUBO 3D COLORIDO
FONTE: CRIADO PELO AUTOR

Tido como concluído o módulo chamado de vídeo, responsável por apresentar os movimentos de atitude do protótipo, a janela da interface gráfica começou a ser construída, tendo como primeiro componente, o ambiente virtual com o objeto 3D colorido. A pesquisa foi então concentrada em carregar os valores reais coletados pelo protótipo, salvos em um arquivo de texto no cartão de memória, convertê-los em um formato compatível com o desenho de vértices do cubo e animar o objeto 3D com valores reais.

Nesta etapa, carregar os valores do arquivo texto para o programa não foi problema, mas houve enorme dificuldade em convertê-los para o formato compatível com o desenho de vértices do cubo, dada a complexidade matemática dos movimentos, relatada anteriormente. A animação tornou-se muito complexa ao ter que sincronizar o movimento do cubo sobre seus três eixos, conforme havia sido coletado pelos sensores giroscópios. A dificuldade neste caso foi relacionada à exibição da informação versus quadros por segundo, medida comumente usada tratando-se de cadência de mídias audiovisuais. Era necessário exibir no sistema gráfico a evolução dos valores dos sensores com uma fluência aceitável, porém, conforme eram ativados ou desativados sensores durante o percurso, existiam mais ou menos informações coletadas pelo protótipo a cada segundo. A relação era de, quanto mais sensores ativos, menos leituras

eram feitas no ciclo de 1 segundo. A figura a seguir exemplifica o conceito de quadros por segundo com um objeto fazendo uma meia-volta em 12 *fps* e 5 *fps*:

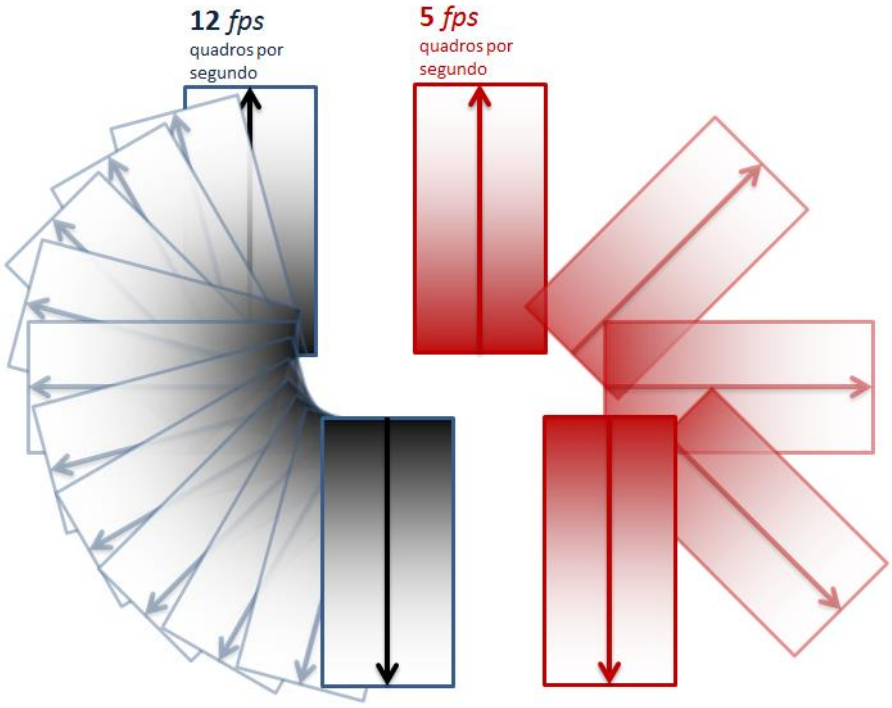


FIGURA 12 - EXEMPLIFICAÇÃO DE QUADROS POR SEGUNDO
FONTE: CRIADO PELO AUTOR

Como pode ser visto na imagem anterior, quanto maior a quantidade de quadros por segundo, maior a amostra de informações a cada segundo e melhor a suavidade de um movimento, enquanto que uma baixa quantidade de amostras por segundo implica no efeito travado de um movimento. O mesmo ocorre durante a coleta de informações dos sensores, eventualmente o ciclo de leituras em um segundo é mais rápido, fazendo mais leituras, outras vezes mais devagar, fazendo menos leituras.

No momento da exibição na interface gráfica, era necessário interpretar a quantidade de leituras a cada segundo e sincronizar os componentes gráficos. Desta forma, uma medida de tempo na exibição poderia ser ajustada com uma medida de tempo real, tornando a visualização agradável ao usuário. Em resumo, precisou ser criada a seguinte rotina no programa GUI: a cada mudança de segundo, a quantidade de leituras é calculada e a exibição é distribuída dentro daquele segundo específico.

Outra dificuldade encontrada foi idealizar o componente que representaria as variações de altitude. A pesquisa rumou para um modelo de gráfico de linha que pudesse apresentar ao longo do tempo as variações de altitude percorridas pelo protótipo. Tendo

em vista que a altitude pode variar em centenas de metros e o gráfico de linhas baseia-se em uma escala, a premissa de funcionamento era que a escala pudesse adaptar-se às leituras.

Em um exemplo, o gráfico não atenderia caso tivesse um valor máximo de escala fixo em 800m, e durante o percurso fosse necessário visualizar variações e 10m. As variações na linha seriam pequenas e imprecisas. Para solucionar a flexibilidade da escala do gráfico, foi necessário criar uma rotina no programa que pudesse identificar os valores mínimo e máximo de leitura naquele trajeto, e configurar os valores da escala antes da ativação do componente na tela do programa.

A solução de escala variável foi então estendida para o componente de visualização de luminosidade e de som. Ou seja, para estes componentes, os valores de mínimo e máximo, bem como sua escala, variam de acordo com o conjunto de leituras realizado. Os gráficos de gás, no modelo *gauges*, foram os últimos a serem criados, e possuem escala fixa, visto que sua abrangência não é tão extensa. A versão final do sistema gráfico pode ser vista na próxima imagem:

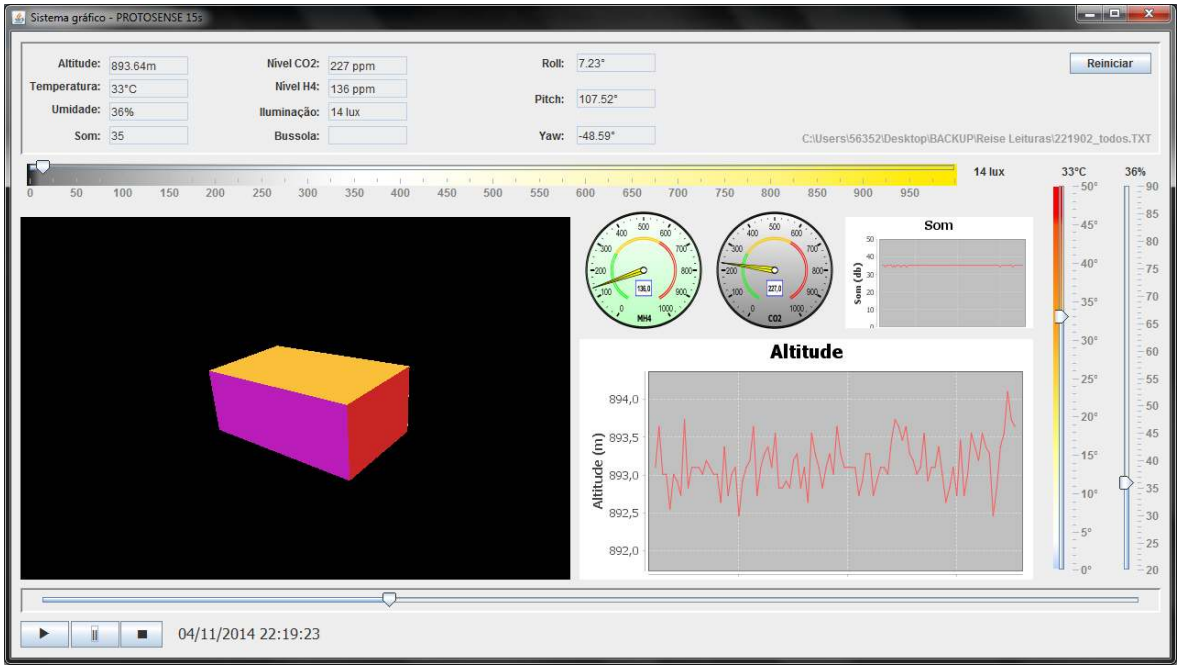


FIGURA 13 - VERSÃO FINAL DO SISTEMA GRÁFICO
FONTE: CRIADO PELO AUTOR

Observa-se na imagem anterior, diversos componentes gráficos reunidos em um painel de forma a permitir a visualização simultânea dos dados coletados pelo protótipo. À direita, duas barras verticais representam a variação de valor dos sensores de temperatura, entre 0ºC e 50ºC, e umidade variando entre 20% e 90%. Dois gráficos de

linha com escala auto ajustável apresentam a variação dos valores de som e altitude ao longo da linha do tempo do vídeo. Ao centro, dois gráficos *gauges* oscilam conforme a variação de gás carbônico e metano. Horizontalmente, na parte superior uma barra de escala variável apresenta a intensidade de luz coletada durante o percurso. O sistema gráfico deveria, portanto, ser capaz de interpretar o arquivo contido no cartão, e previamente gravado pelo protótipo durante o percurso realizado. O formato de arquivo de texto gravado está exemplificado na próxima figura.

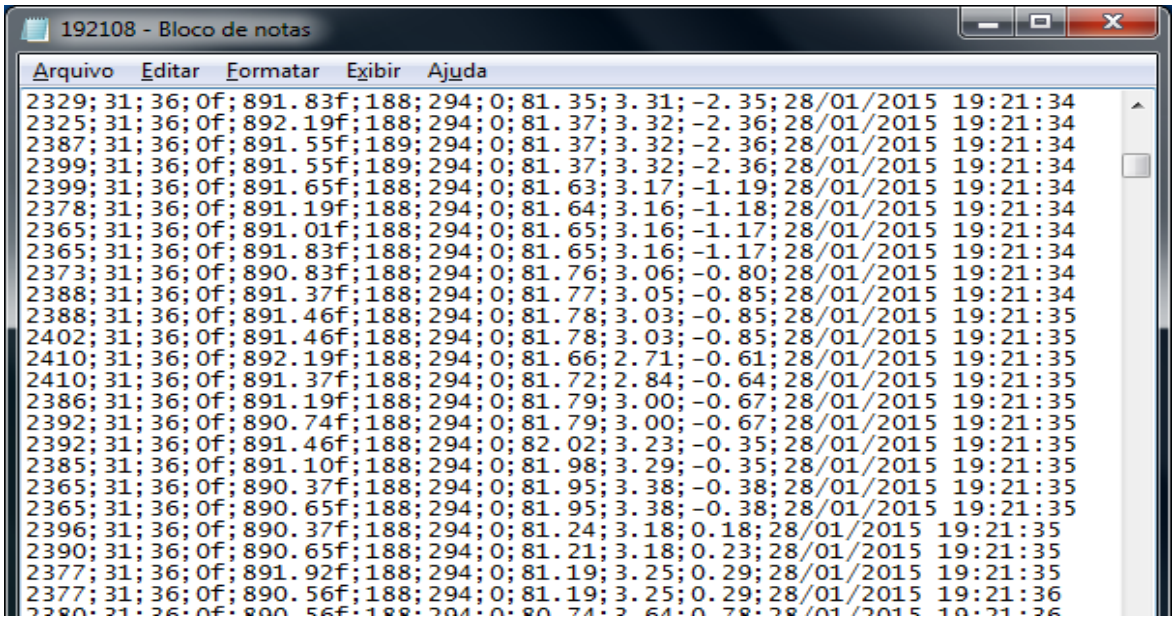


FIGURA 14 - EXEMPLO DE ARQUIVO GERADO PELO PROTÓTIPO E CARREGADO PELO SISTEMA GRÁFICO
FONTE: CRIADO PELO AUTOR

Em um teste prático, o protótipo foi acoplado em uma bicicleta, que percorreu um caminho misto, com subidas e descidas e realizando curvas, a fim de verificar a resposta do objeto gráfico 3D às guinadas e inclinações do veículo. A seguir, um conjunto de imagens mescladas trazem a fotografia do trecho e a resposta dos objetos gráficos criados.



FIGURA 15 - MONTAGEM COM PERCURSO DO VEÍCULO E INTERPRETAÇÃO DO OBJETO GRÁFICO
FONTE: CRIADO PELO AUTOR

Com o trajeto e a posterior análise, observou-se que o objeto 3D representado na interface gráfica respondeu corretamente às variações de rotação nos 3 eixos realizadas pelo veículo durante o percurso, Também com resultados satisfatórios na interface gráfica apresentaram-se os resultados das variações de altitude, iluminação, temperatura e gases. As figuras consecutivas a seguir apresentam estes resultados.

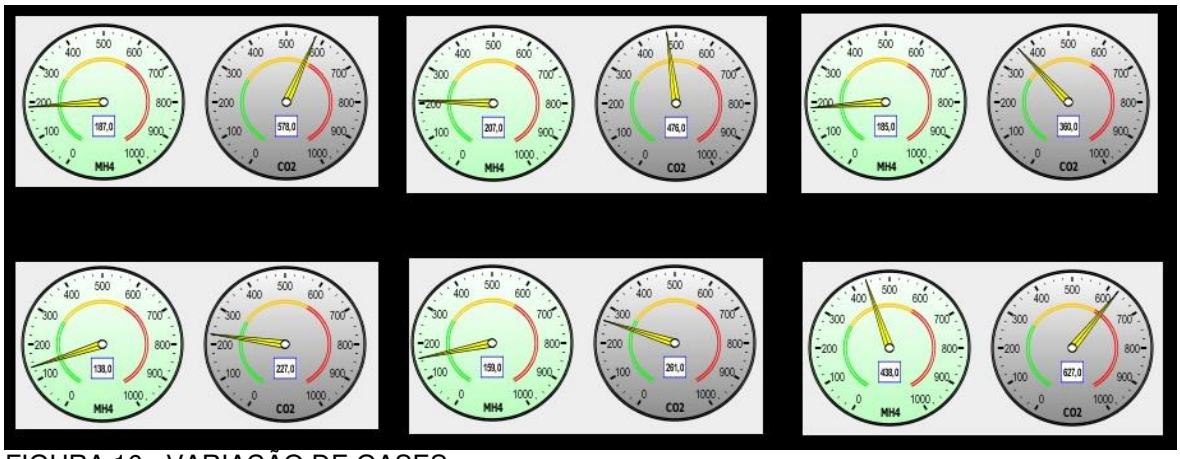


FIGURA 16 - VARIAÇÃO DE GASES
FONTE: CRIADO PELO AUTOR

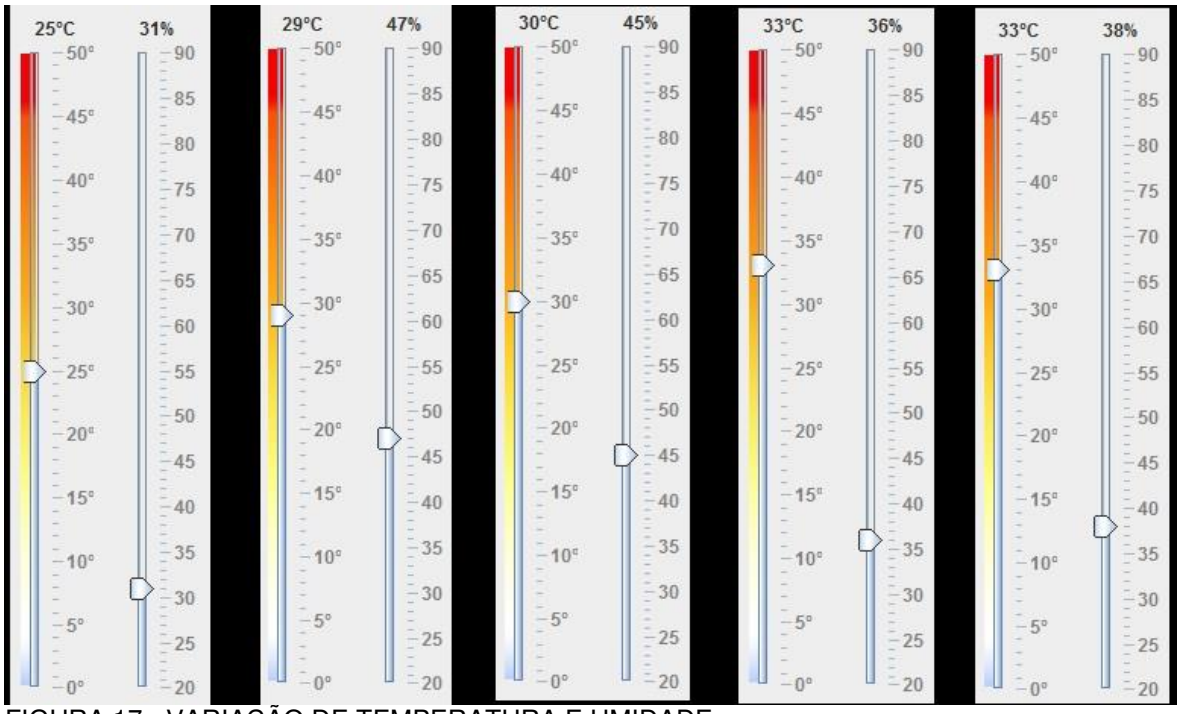


FIGURA 17 - VARIAÇÃO DE TEMPERATURA E UMIDADE
FONTE: CRIADO PELO AUTOR

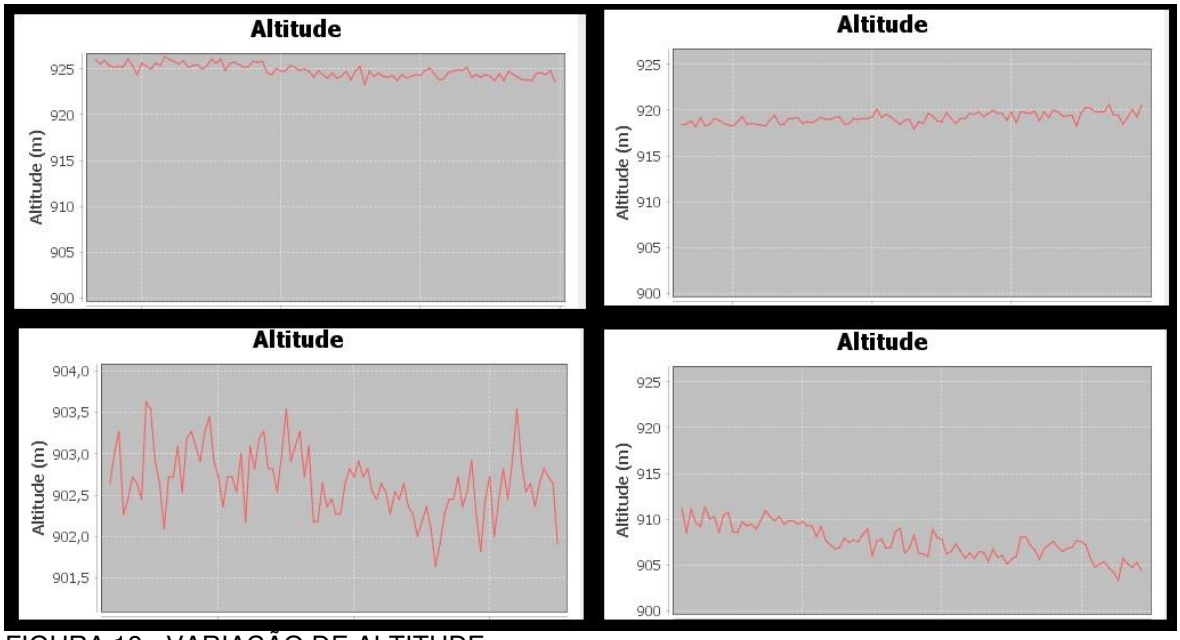


FIGURA 18 - VARIAÇÃO DE ALTITUDE
FONTE: CRIADO PELO AUTOR

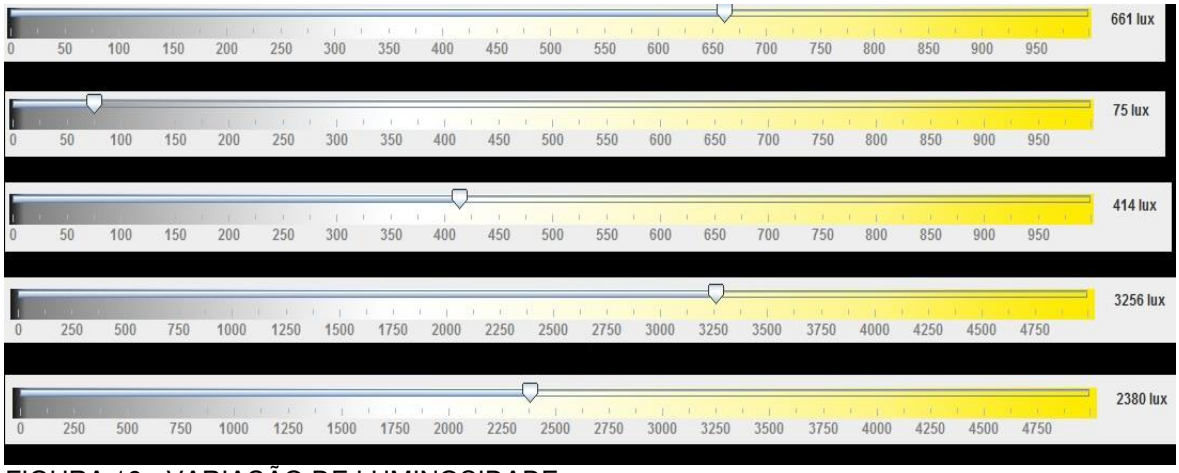


FIGURA 19 - VARIAÇÃO DE LUMINOSIDADE
FONTE: CRIADO PELO AUTOR

As imagens apresentadas anteriormente foram retiradas de momentos distintos da reprodução do arquivo, retiradas diretamente do painel criado no sistema gráfico.

3. CONCLUSÃO

Este trabalho foi desenvolvido com o objetivo acadêmico de explorar sistemas embarcados e sistemas gráficos, dois segmentos distintos da área de tecnologia da informação. O sistema embarcado foi baseado na plataforma Arduino integrada a diversos sensores para coleta de dados, sendo referido no documento como protótipo. A parte gráfica criada sugere apresentar de forma simples e simultânea as informações previamente adquiridas pelo protótipo.

Foi necessário pesquisar e conhecer o funcionamento de sistemas embarcados, princípios básicos de eletrônica e lógica binária, propriedades nativas de cada um dos sensores utilizados, bem como informar-se sobre suas medidas, bibliotecas para sistemas gráficos, modelagem 3D e sincronização de informações.

Além de suprir as necessidades em caso de pesquisa e exploração de ambientes, o protótipo desenvolvido atende a coleta de informações de veículos comuns ou meios de transporte dos mais variados tipos, como carros, motos, bicicletas, helicópteros, barcos, entre outros. Podendo ser usado por leigos para tomar conhecimento geral sobre o ambiente ao seu redor ou as variações de algumas medidas durante um trajeto feito casualmente.

Na versão final do protótipo, dois dos oito sensores não funcionaram. O microfone, sensor de detecção de som, adquirido aparentemente tinha um defeito funcional, apresentando o mesmo valor e leitura em qualquer situação que fosse submetido. No entanto, a estrutura física do dispositivo o comporta e o programa para

sua interpretação foi feito, podendo esta medida funcionar ao substituir o componente. O outro sensor que apresentou problemas foi o magnetômetro, que funciona como uma bússola digital, que apresentou valores corretos de leitura quando testado isoladamente, mas incoerentes ao ser encaixado dentro do protótipo, causado pela interferência magnética dos vários componentes eletrônicos adjacentes. A falta dos dois sensores não comprometeu o resultado final ou o atingimento dos objetivos específicos do projeto.

Este trabalho teve grande importância para a descoberta de tecnologias relevantes para a área, e expansão do conhecimento de integração de sistemas, por ter permitido explorar segmentos de tecnologia fora da grade disciplinar deste curso na faculdade.

4. REFERÊNCIAS

DAVIS, Gene (2005). ***Hello* JOGL**. Disponível em http://www.javaworld.com/article/2071845/swing-gui-programming/hello-jogl.html. Acesso em 21/04/2014.

DOWNLOAD ARDUINO. ***Download the Arduino Software***. Disponível em http://arduino.cc/en/Main/Software. Acesso em 21/04/2014.

HOW STUFF WORKS (2009). ***Magnetometer Info***. Disponível em http://science.howstuffworks.com/magnetometer-info.htm. Acesso em 13/04/2014.

HOW THINGS FLY. ***Roll, Pitch and Yaw***. Disponível em http://howthingsfly.si.edu/flight-dynamics/roll-pitch-and-yaw. Acesso em 06/04/2014.

INVENSENSE. **MPU-6000 *and* MPU-6050 *Product Specification Revision* 3.2**. *InvenSense* Inc, 2011.

KONG, X. ***Inertial Navigation System Algorithms for Low Cost* IMU**. *Department of Mechanical and Mechatronic Engineering*, *The University of Sydney*, Australia, 2000.

KUGA, H.K. E MILANI, P.G. E EINWOEGERER,W.; **Experimentos de alinhamento de unidade de medida inercial baseada em MEMS (*Micro Electro-Mechanical Systems*)** V Simpósio Brasileiro de Engenharia Inercial – V SBEIN, Rio de Janeiro, RJ, Brasil, 2007.