

Avaliação de um módulo de reconhecimento de fala na plataforma Arduino



Carlos Alberto Araújo de Andrade ^[1], Gustavo Ramos Silva ^[2], Daniella Dias C. da Silva ^[3], César Rocha Vasconcelos ^[4]

[1] carloz.alberara@gmail.com. [2] gustavoramos234@gmail.com [3] daniella.silva@ifpb.edu.br. [4] cesarocha@ifpb.edu.br. Instituto Federal de Educação, Ciência e Tecnologia da Paraíba – Campus Campina Grande.

RESUMO

O grande avanço tecnológico gerado pela globalização tornou urgente a busca por dispositivos eletrônicos que sejam viáveis, ágeis e práticos. Uma ferramenta poderosa que cinge todo esse cunho fundamental é a possibilidade de realizar a interação com tais dispositivos por meio do reconhecimento da fala. O objetivo deste trabalho foi avaliar um módulo de reconhecimento de fala compatível com a plataforma Arduino, identificando seus recursos como também suas limitações. Com o intuito de investigar um possível aumento da taxa de reconhecimento, foram realizados testes com um microfone diferente daquele que acompanha o módulo. A metodologia adotada incluiu testes de reconhecimento em um ambiente com ruído e uma análise de SNR (Signal-to-Noise Ratio) para os microfones avaliados. Os resultados obtidos mostraram que o microfone original fornece um melhor desempenho. Ao longo deste artigo são apresentados os principais conceitos relacionados ao processo de reconhecimento de fala, microfones, plataforma Arduino e funcionamento do módulo analisado.

Palavras-chave: Reconhecimento de fala. Arduino. Microfones. Sistemas embarcados.

ABSTRACT

The great technological advancement that is generated by the globalization has made urgent the search for electronic devices that are viable, agile and practical. A powerful tool, and that put all of this imprint is the possibility of realize the interaction with these devices through the speech recognition. The goal of this work was to evaluate a speech recognition module which is compatible with Arduino platform, identifying its resources and limitations. In order to investigate a possible increased recognition rate, tests have been performed with a different microphone from the one that came with the module. The adopted methodology in this study included speech recognition tests in a noisy environment and analyses of SNR (Signal-to-Noise Ratio) for the evaluated microphones. The obtained results showed that the original microphone provides better performance. The main concepts about speech recognition, microphones, Arduino and evaluated module are presented throughout this paper.

Keywords: *Speech recognition. Arduino. Microphones. Embedded systems.*

1 Introdução

Diante do crescente interesse por *softwares* e equipamentos que “compreendam”, reconheçam e simulem a voz humana, muitas pesquisas têm sido realizadas na área de Processamento Digital de Sinais de Voz (PDSV), buscando o desenvolvimento de técnicas mais eficientes no reconhecimento da fala (RABINER; SCHAFFER, 2010; SILVA, 2011). Além disso, a aplicação de sistemas automatizados por comandos de voz pode dar maior acessibilidade às pessoas com limitações motoras.

A complexidade dessas aplicações caracteriza-se ao longo de duas dimensões: o tamanho do vocabulário e a forma da elocução. Quanto maior o vocabulário, maior será a dificuldade de reconhecimento. Essa dificuldade é ainda maior, quando se deseja implementar esse tipo de sistema em dispositivos com limitações de processamento e armazenamento e dependentes de bateria como, por exemplo, eletrodomésticos, *smartphones* e *tablets*. O grande desafio é aliar o bom desempenho no reconhecimento ao baixo consumo de energia e recursos computacionais (JIANG; MA; CHEN, 2010). Para permitir a execução de sistemas de PDSV nesse contexto, alguns trabalhos sacrificam a eficiência no processo de reconhecimento em nome da redução da área e de exigências computacionais (SILVA, 2011).

Tendo em vista o vigente interesse pelo desenvolvimento de aplicações em PDSV para sistemas embarcados (*embedded systems*) e também as dificuldades enfrentadas nesse tipo de projeto, torna-se clara a importância da definição e aproveitamento de soluções nas diversas aplicações emergentes.

Uma plataforma que tem sido bastante utilizada para o desenvolvimento de sistemas embarcados de maneira rápida é o Arduino. Essa plataforma faz parte de um grande projeto que engloba *software* e *hardware* para desenvolver aplicações de baixo custo. Um grande diferencial desse dispositivo é ser mantido por uma comunidade que trabalha na filosofia *open-source*, que desenvolve e disponibiliza o código fonte de seus projetos. Também é possível aumentar a capacidade do *hardware* do Arduino, acoplando a ele os chamados *shields* (escudos) ou módulos, que consistem em placas que podem ser interligadas à placa original, agregando funcionalidades a ela. Atualmente, existem alguns módulos de reconhecimento de fala para a plataforma Arduino.

Neste trabalho, foram realizados testes com o módulo *Voice Recognition Module V2®* da *Elechouse*. Esse módulo consegue processar o áudio e reconhecer comandos pré-definidos que lhe foram atribuídos, permitindo que o usuário controle qualquer função que esteja programada junto ao Arduino.

O principal objetivo deste trabalho foi testar e avaliar o desempenho do referido módulo. Além disso, com base no resultado dos testes, investigar possíveis soluções para melhorar seu desempenho em relação à taxa de reconhecimento. Assim, também foram realizados testes com um microfone diferente daquele que acompanha o *kit* do *Voice Recognition Module V2*. Os detalhes sobre o seu funcionamento, bem como os resultados dos testes realizados, são apresentados ao longo deste artigo.

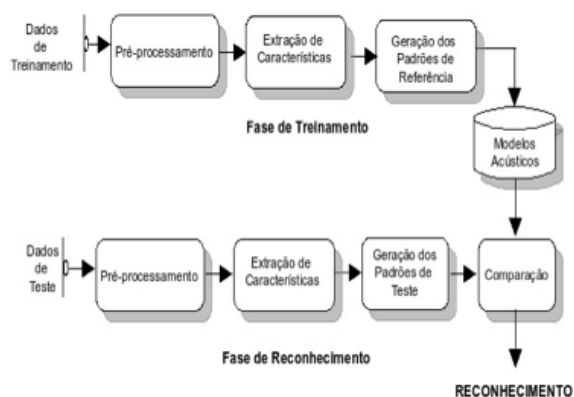
O restante deste documento está organizado da maneira a seguir. Na Seção 2, são apresentadas as etapas típicas de um sistema de reconhecimento da fala. Na Seção 3, são apresentadas as principais características da plataforma Arduino e do módulo de reconhecimento utilizado. Na Seção 4, é apresentada a metodologia empregada para realização dos testes com o módulo de reconhecimento. Na Seção 5, é descrito o procedimento para conexão entre o Arduino e o *Voice Recognition Module V2*. Na Seção 6, são apresentadas as principais características dos microfones. Na Seção 7, são apresentados os detalhes e os resultados dos experimentos realizados neste trabalho. Finalmente, na Seção 8, são realizadas as considerações finais e sugeridas algumas pesquisas futuras.

2 Sistemas de Reconhecimento da Fala

Sistemas de Reconhecimento Automático da Fala (SRAF) têm por objetivo determinar qual palavra, frase ou sentença foi pronunciada. Esses sistemas podem ser subdivididos em um grande número de subáreas, dependendo de alguns fatores, tais como tamanho do vocabulário e população de locutores (RABINER; SCHAFFER, 2010).

Um SRAF realiza uma tarefa de reconhecimento de padrões – nesse caso, padrões de fala – e sempre inclui duas fases (FECHINE, 2000): treinamento e reconhecimento. As principais etapas executadas em cada uma dessas fases são ilustradas na Figura 1.

Figura 1 - Sistema de reconhecimento de padrões aplicado ao reconhecimento de fala.



Fonte: MCROBERTS, 2011

A etapa de pré-processamento é responsável por extrair do sinal de voz os dados relevantes e menosprezar a informação redundante, com a finalidade de repassar a informação de interesse para a etapa seguinte. Esse processo pode incluir diversas operações, sendo as mais comuns: aquisição do sinal de voz, pré-ênfase, detecção de início e fim, tratamento de ruído, separação em *frames* e janelamento (SHAUGHNESSY, 2003; RABINER; SCHAFER, 2010).

A etapa de extração de características é de extrema importância em sistemas de reconhecimento de fala, uma vez que nela são obtidos elementos que possibilitam a geração do padrão. A definição de um bom conjunto de características é um processo complexo e essencial, para que se obtenha bons resultados quanto ao reconhecimento dos padrões.

Durante a fase de treinamento, com base nas características extraídas, são gerados padrões de referência, que serão comparados com o padrão de teste na fase de reconhecimento.

Vale salientar que as etapas de pré-processamento e de extração de características das fases de treinamento e de reconhecimento devem ser equivalentes, para que seja possível uma comparação correta entre o padrão testado e o(s) padrão(ões) já conhecido(s) pelo sistema (SILVA, 2011).

3 Plataforma Arduino e o módulo de reconhecimento de fala

Atualmente, atenta-se para o conforto, acessibilidade e praticidade da sociedade, que podem ser proporcionados por uma crescente oferta de dispositivos eletrônicos. Uma maneira de proporcionar essas características é realizar a interação homem-

-máquina por meio da fala. É possível aplicar essa técnica em diversas áreas, como segurança (controle de acesso), mobilidade humana (indivíduos com limitações físicas), automação residencial/comercial, enfim, em uma vasta gama de aplicações.

Existe também, um crescente mercado que tem como intuito a miniaturização desses dispositivos, surgindo, assim, o conceito de sistemas embarcados (ou embutidos), que executam aplicações de *software* em uma plataforma de *hardware* diferente dos microprocessadores convencionais.

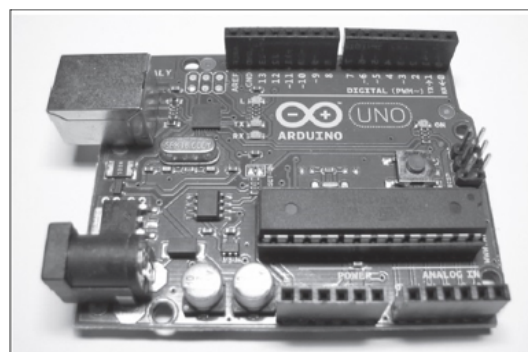
Uma plataforma que tem sido bastante utilizada para o desenvolvimento de sistemas embarcados, de maneira rápida, é o Arduino. Sendo possível expandir suas funcionalidades, por meio de *shields*, que consistem em placas de expansão que podem ser conectadas na placa principal do Arduino (BARR, 1999; FRANCA, 2001).

A plataforma Arduino faz parte de um grande projeto que engloba *software* e *hardware* abertos para a contribuição de toda a sociedade. É uma ferramenta de fácil aprendizagem que não exige inicialmente grandes conhecimentos específicos em programação e eletrônica. E possibilita o desenvolvimento real de pequenos projetos de automação com poucos recursos necessários para sua implementação.

Trata-se de uma plataforma de computação física, área da computação em que os sistemas digitais ligados a sensores e atuadores percebem a realidade do campo externo e respondem com ações físicas devido a instruções dadas por um *software* específico. Baseado em uma pequena placa eletrônica microcontrolada e desenvolvida para trabalhar com bibliotecas de códigos simplificadas, o Arduino foi projetado para ser programado com a linguagem de programação ANSI C/C++ (MCROBERTS, 2011).

Neste trabalho, foi utilizado o Arduino de modelo UNO REV 3 (Figura 2)..

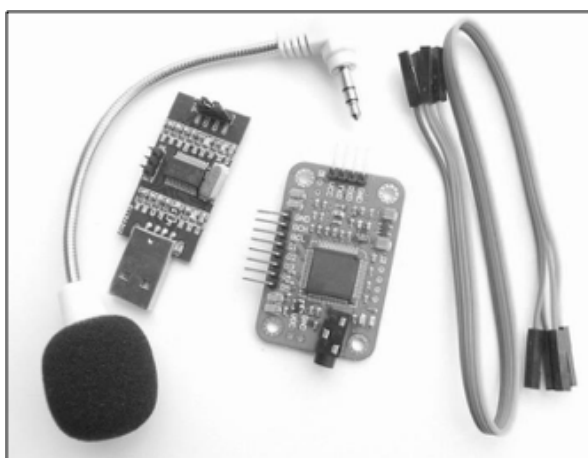
Figura 2. Arduino: modelo UNO REV 3.



Sugerir o Arduino como proposta de microcontrolador, demonstra que é possível utilizar recursos de automação em conjunto com o recurso de reconhecimento de fala em baixo custo.

Neste trabalho, também foi utilizado um dispositivo que realiza toda a função de captura, processamento e reconhecimento dos comandos de fala a ele dirigidos e devidamente treinados. Esse dispositivo é o Módulo de Reconhecimento de Fala, versão 2 ou *Voice Recognition Module V2* e é fornecido com um kit que inclui, além do módulo de reconhecimento, um microfone e um conversor serial TTL 5V (Figura 3).

Figura 3. *Voice Recognition Module V2* utilizado neste trabalho.



Esse módulo de reconhecimento de fala foi projetado, desenvolvido e comercializado pelo seu próprio criador, Wilson Shen. Atualmente, encontra-se esse dispositivo até a versão 3. O módulo recebe os comandos de configuração e envia a resposta por meio de uma interface de porta serial.

A capacidade de reconhecimento varia de uma versão para outra do módulo. No caso do modelo V2, são quinze palavras isoladas pronunciadas por um mesmo locutor (SHEN, 2013). Testes realizados ao longo deste trabalho demonstram que, se a aplicação for utilizada em um ambiente ruidoso, a taxa de reconhecimento diminui consideravelmente.

O *Voice Recognition Module V2* processa o áudio de voz capturado e reconhece comandos pré-definidos, permitindo que o usuário controle qualquer função que esteja programada junto ao Arduino. A comunicação do módulo com o Arduino e/ou computador é realizada por meio de uma interface de porta serial.

Como foi mencionado anteriormente, esse módulo pode reconhecer até 15 peças de instrução de voz. Essas 15 peças são organizadas em três grupos, cada um com cinco peças para se guardar as instruções. Apenas um grupo pode estar ativo por vez. Isso implica dizer que, a cada instante, apenas 5 palavras podem ser reconhecidas. Para reconhecer as demais palavras, é necessário ativar o grupo correspondente. As peças são espaços de alocação na memória do módulo de reconhecimento que guardam as informações após serem processadas, permitindo que seja possível realizar o reconhecimento posteriormente.

Como o módulo é dependente de locutor, cada grupo de instruções deve ser treinado pela mesma pessoa que utilizará a aplicação desenvolvida, ou seja, o módulo reconhecerá apenas a voz de quem o treinou. No processo de treinamento, cada comando deve durar no máximo 1300 ms.

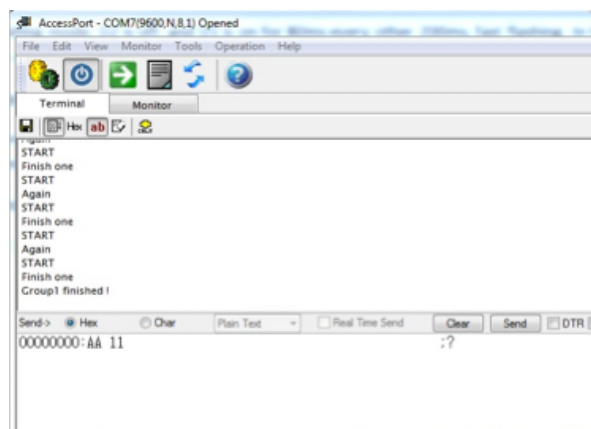
No processo de captura e treinamento desses comandos, para cada grupo do dispositivo, é necessário utilizar uma IDE (*Integrated Development Environment*), ou Ambiente de Desenvolvimento Integrado, fornecida pelo próprio fabricante do módulo.

Essa ferramenta permite enviar ao módulo os comandos em hexadecimal que ativam suas configurações operacionais. Esses comandos são como instruções internas, reconhecidas apenas pelo próprio módulo. O envio é realizado via porta serial, por uma interface TTL de 5V. Cada operação possui um comando específico.

Na Figura 4, é ilustrado o processo de envio de um comando para o módulo. Nesse caso, está sendo iniciada a fase de gravação (treinamento do módulo). Para isso, foi utilizada a IDE e, com o módulo devidamente conectado ao computador via porta serial, foi enviado o comando: AA 11. Tal comando iniciará o processo de treinamento do primeiro grupo que contém 5 peças.

As respostas do módulo aos comandos enviados também vão sendo exibidas na IDE. Respostas típicas são, por exemplo, confirmação da gravação, solicitação da próxima peça, fim do processo de gravação e outras. No total são 77 comandos internos existentes (SHEN, 2013).

Figura 4. Enviando comando de setup para o módulo de reconhecimento de fala.



4 Metodologia

O objetivo principal deste trabalho foi testar e avaliar o desempenho do *Voice Recognition Module V2*. Além disso, com base no resultado dos testes, investigar possíveis soluções para melhorar seu desempenho em relação à taxa de reconhecimento.

Como foi descrito na seção anterior, o módulo de reconhecimento utilizado apresenta basicamente três principais limitações:

- *Pequeno vocabulário* – apesar de reconhecer 15 peças (comandos), eles são divididos em 3 grupos de 5 peças e apenas um grupo pode estar ativo por vez. Isso faz com que em um determinado momento, apenas 5 comandos possam ser reconhecidos/diferenciados.
- *Dependência de locutor* – o módulo só reconhece comandos pronunciados pelo mesmo usuário que realizou o treinamento.
- *Sensibilidade ao ruído* – a taxa de reconhecimento diminui diante de um ambiente com interferências sonoras, que podem ser desde pessoas conversando a equipamentos que emitam algum ruído, como ar-condicionado, motores, etc.

Para tentar resolver a limitação do tamanho do vocabulário e dependência de locutor, seria necessário ter acesso ao código fonte do módulo, a fim de realizar modificações nas técnicas e/ou parâmetros utilizados no algoritmo de reconhecimento. No entanto, apesar de ser compatível com a plataforma Arduino, esse módulo não segue a filosofia *open-source*, ou seja, o código fonte é proprietário e não está disponível.

Diante disso, a única alternativa foi tentar diminuir a sensibilidade do módulo ao ruído ambiente. Para

isso, foram realizados testes de modo a verificar se o tipo do microfone utilizado para captura do sinal pode influenciar nessa característica.

Foram adotadas as seguintes etapas para desenvolvimento deste trabalho:

Etapa 1: Inicialmente, foram realizados estudos sobre as principais características de diferentes tipos de microfones, suas vantagens e desvantagens, bem como sua possível influência na captação do ruído ambiente.

Etapa 2: Com base nos estudos da primeira etapa, foram realizadas capturas de sinais de áudio com dois tipos de microfones. Para essa captura, foi utilizado o *software* Audacity. Em seguida, foi utilizado o MatLab®, para calcular a SNR (*Signal-to-Noise Ratio*) para cada sinal capturado. Os valores obtidos para cada amostra nos diferentes microfones foram analisados e comparados com os resultados obtidos na etapa seguinte.

Etapa 3: Nessa etapa, foram realizados testes de reconhecimento com o módulo e os diferentes microfones analisados.

5 Conexão entre *Voice Recognition Module V2* e Arduino

Como foi informado anteriormente, para fazer uso do reconhecimento fornecido pelo *Voice Recognition Module V2*, é necessário realizar previamente a gravação das palavras que serão reconhecidas, ou seja, é exatamente o mesmo processo de treinamento e reconhecimento que foi descrito na Seção 2.

Durante as etapas de treinamento e reconhecimento, as conexões entre o *Voice Recognition Module V2* e o Arduino são diferentes. Nesta seção, serão apresentadas as conexões e os procedimentos necessários em cada uma dessas etapas.

5.1 Treinamento

O kit do *Voice Recognition Module V2* inclui um conversor serial TTL 5V utilizado exclusivamente para troca dos comandos de controle entre o módulo e sua IDE durante o treinamento, entretanto testes realizados neste trabalho comprovaram que a plataforma Arduino UNO pode ser utilizada para realizar a função do referido conversor. Tal recurso pode se tornar interessante em caso de problemas (ou falha) com o conversor e em casos que provoquem a impossibilidade do seu uso.

Na Figura 5, é ilustrada a ligação direta entre o Arduino e *Voice Recognition Module V2* na etapa de treinamento do módulo, sem a utilização do conversor serial que acompanha o kit. Na Tabela 1, é apresentada a relação entre os pinos do Arduino e do módulo nessa etapa.

Figura 5. Conexões entre Arduino e módulo para treinamento dos comandos de voz.

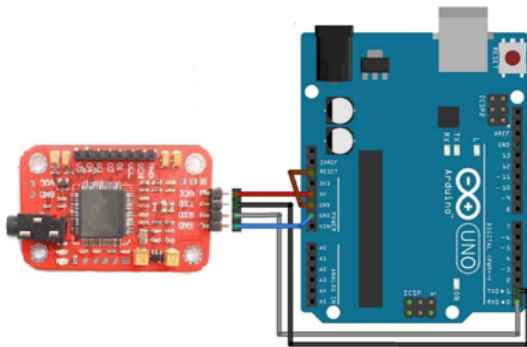


Tabela 1. Conexões para envio dos comandos pela IDE durante treinamento

Arduino		<i>Voice Recognition Module V2</i>
RESET	GND	---
5V		VCC
TX		TX
RX		RX
GND		GND

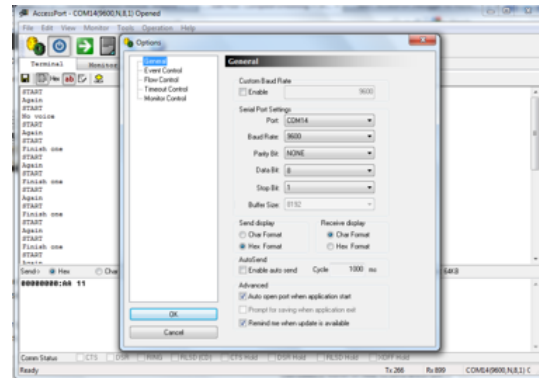
Vale destacar que é extremamente recomendado que essas conexões sejam realizadas com a alimentação elétrica interrompida, garantindo a segurança dos dispositivos utilizados. Após a correta ligação entre o módulo e o Arduino, também foi necessária a utilização de um *jumper* ligando os pinos RESET e GND do Arduino. Uma vez efetuadas todas as ligações citadas, o Arduino passou a atuar como um conversor serial TTL para a comunicação de dados entre o *Voice Recognition Module V2* e sua IDE.

Na Figura 6, é apresentada a tela da IDE AcessPort® que permite a configuração dos parâmetros da transmissão para o módulo. Neste trabalho, foram utilizados os seguintes valores:

- *BAUD RATE*: 9600 BAUD/S
- *Serial Port Settings*: COM estabelecida pelo PC
- *Parity Bit*: None

- *Data Bit*: 8 (um Byte)
- *Stop Bit*: 1
- *Buffer size*: 8192
- *Send display*: Hex Format
- *Receive display*: Char Format.

Figura 6. IDE AcessPort®: configurações para etapa de gravação dos comandos de voz.



Após configurada a transmissão, foi conectado o cabo USB, que funcionou como fonte de alimentação e meio de transmissão de dados do computador para o Arduino. Em seguida, na IDE, bastou digitar os comandos em HEX disponíveis no manual de instrução do *Voice Recognition Module V2*.

Finalizada a etapa de treinamento, o jumper entre os pinos RESET e GND do Arduino foi retirado, permitindo, assim, a utilização do módulo para o reconhecimento.

5.2 Reconhecimento

Uma vez realizada a etapa de treinamento, o módulo passa a armazenar em sua memória as palavras que podem ser reconhecidas por ele. A etapa seguinte consiste em gravar na memória do Arduino um programa que utilizará o reconhecimento fornecido pelo módulo, para controlar alguma função específica.

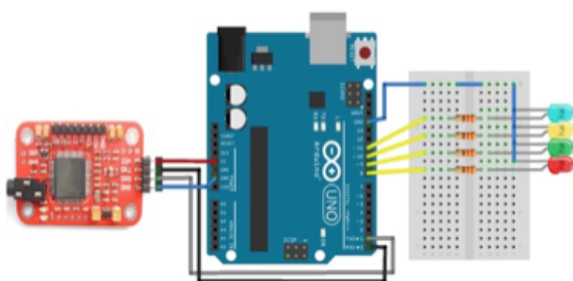
Neste trabalho, foi desenvolvido um programa para controlar o acendimento de quatro LEDs (*Light Emitter Diode*). Para isso, o módulo foi treinado com as seguintes palavras: vermelho, verde, amarelo, azul e desligar.

Para realizar a transferência do programa para o Arduino, é necessário que este esteja desconectado do *Voice Recognition Module V2*. Depois de realizada a transferência, a conexão entre ambos pode ser estabelecida. Na Figura 7 e na Tabela 2, são apresentadas as conexões necessárias nessa etapa.

Tabela 2. Conexões para reconhecimento dos comandos de voz.

Arduino	Voice Recognition Module V2
5V	VCC
TX	TX
RX	RX
GND	GND

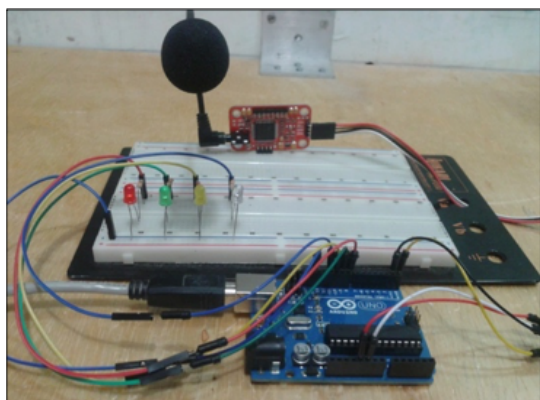
Figura 7. Conexões necessárias entre Arduino, módulo e LEDs, durante os testes de reconhecimento.



Nessa etapa, é importante observar que, diferente da etapa de treinamento, não foi mais necessário um jumper entre os pinos GND e RESET do Arduino. Outra importante mudança foi a inversão entre os pinos RX e TX do Arduino e do *Voice Recognition Module V2*. O RX do Arduino foi conectado ao TX do módulo, enquanto que o TX do Arduino, ao RX do módulo.

Uma vez que todas as conexões foram corretamente realizadas, bastou iniciar a execução do programa previamente gravado no Arduino. Na Figura 8, é exibida uma foto do protótipo desenvolvido neste trabalho.

Figura 8. Protótipo desenvolvido.



Microfones são transdutores, ou seja, dispositivos capazes de transformar uma magnitude física mensurada em sinais elétricos. No caso do microfone, que é um dispositivo eletromecânico, ele é responsável pela transformação de ondas mecânicas (som) em ondas elétricas equivalentes. Suas aplicações são diversas como, por exemplo, telefones, gravadores de fitas, aparelhos auditivos, transmissão de rádio e de televisão.

6 Microfones

Microfones são transdutores, ou seja, dispositivos capazes de transformar uma magnitude física mensurada em sinais elétricos. No caso do microfone, que é um dispositivo eletromecânico, ele é responsável pela transformação de ondas mecânicas (som) em ondas elétricas equivalentes. Suas aplicações são diversas como, por exemplo, telefones, gravadores de fitas, aparelhos auditivos, transmissão de rádio e de televisão.

6.1 Principais características dos microfones

Algumas propriedades diferenciam um microfone do outro e vão influenciar no sinal capturado por eles. Entre as principais características, pode-se citar (OLIVEIRA, 2010):

Resposta em frequência – é a definição de como o microfone responderá às fontes sonoras de diferentes faixas de frequência;

Impedância – o valor de impedância do microfone define a carga que ele irá representar para o seu circuito de pré-amplificação;

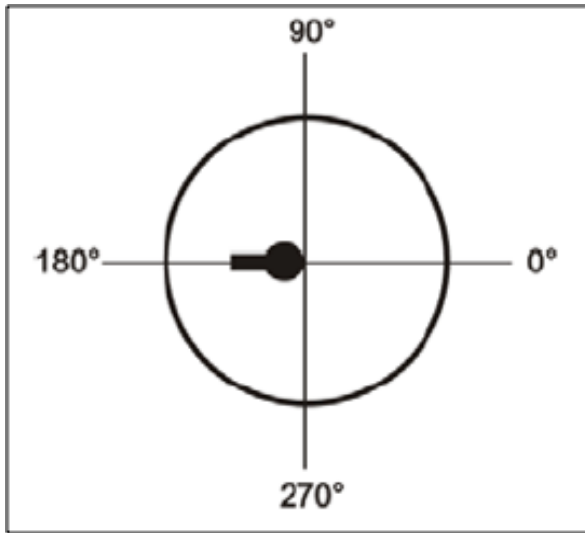
Sensibilidade – é a relação entre o nível elétrico de saída do microfone e a pressão sonora incidente. Esse parâmetro mede o valor do sinal de saída (tensão ou corrente) que o microfone produz a partir de uma pressão sonora, caracterizando, assim, sua eficiência;

Diretividade – é a característica que indica o padrão de sensibilidade de um microfone em função da direção ou da posição da fonte sonora. A diretividade é usualmente indicada por um diagrama polar, pelo qual o comprimento do raio em uma direção (ângulo) indica a resposta em decibéis (*dB*) do microfone.

Com respeito a sua diretividade, um microfone pode ser omnidirecional, cardioides ou bidirecionais, cujas diferenças são descritas a seguir.

Omnidirecional – São os microfones que captam o som da fonte não importando a direção em que esse chegue a sua cápsula (Figura 9).

Figura 9. Diagrama polar de um microfone Omnidirecional.

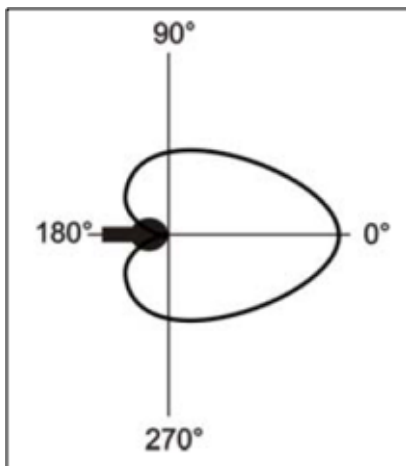


Fonte: OLIVEIRA, 2010.

Cardioides – São microfones que melhor captam o som que está à sua frente. À medida que a fonte sonora se desloca do eixo central do microfone, sua sensibilidade é reduzida. Seu diagrama polar aproxima-se do formato de duas esferas cruzadas entre si (Figura 10).

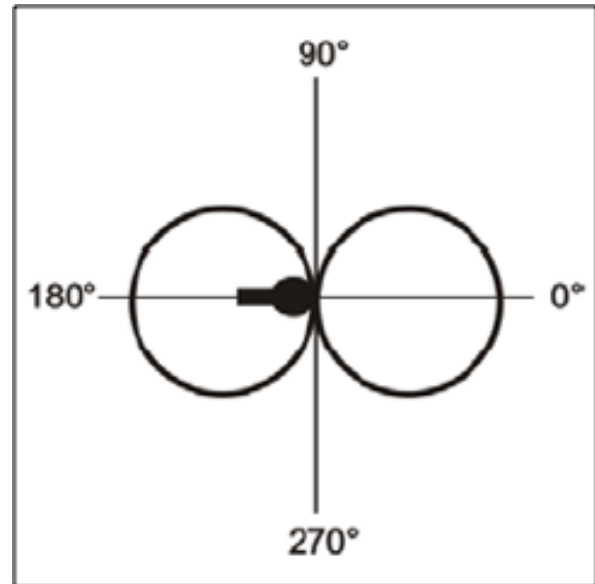
Bidirecional – São os microfones que captam o som da fonte, que, estando no eixo da cápsula do microfone nos ângulos de 0° e 180°, reduz sua sensibilidade nos demais ângulos (Figura 11).

Figura 10. Diagrama polar de um microfone Cardioides.



Fonte: OLIVEIRA, 2010.

Figura 11. Diagrama polar de um microfone Bidirecional.



Fonte: OLIVEIRA, 2010.

7 Experimentos e resultados

Nesta seção, são apresentados os detalhes e resultados dos testes e experimentos realizados neste trabalho. Foram efetuadas análises de SNR e testes de reconhecimento com dois microfones.

7.1 Microfones utilizados para os testes

O kit *Voice Recognition Module V2* é composto de um módulo de processamento, um conversor serial, jumpers e um microfone. Na Tabela 3, são apresentadas as características do microfone que acompanha o kit de reconhecimento de fala.

Tabela 3. Características do microfone do kit *Voice Recognition Module V2*.

Tipo do Microfone	Eletreto
Diretividade	Omnidirecional
Sensibilidade	-54 dB
Frequência de operação	100Hz – 5kHz
Conector	Plug P2

O desempenho do microfone original do módulo foi comparado ao do microfone do *HeadSet PH027* da Multilaser®. As características do PH027 estão descritas na Tabela 4.

Tabela 4. Características do microfone do *HeadSet* PH027.

Tipo do Microfone	Eletreto
Diretividade	Unidirecional
Sensibilidade	-58 dB
Frequência de operação	30Hz – 16kHz
Conector	Plug P2

7.2 Cálculo da SNR

A SNR é utilizada em um sistema para identificar a possibilidade da recepção de um determinado sinal que venha a prover alguma mensagem. É a relação sinal-ruído que, de forma prática, tem como finalidade comparar o nível do sinal mensagem em relação ao nível do ruído. Quanto maior for a SNR, menor vai ser a influência do ruído na transmissão do sinal mensagem.

$$SNR = \frac{P_{sinal}}{P_{ruído}} \quad (1)$$

O cálculo da SNR é realizado a partir da potência do sinal mensagem que será dividido pela potência do ruído, segundo a Equação 1 (LATHI, 2007):

Sendo que, P_{sinal} é a potência do sinal que é composto por alguma mensagem que se deseja transmitir e/ou receber. A variável $P_{ruído}$, por sua vez, é a potência do ruído que é gerado, na maioria das vezes, a partir do efeito *Joule* (aquecimento).

A SNR ainda pode ser calculada a partir da energia de cada parte, ou seja, a energia do sinal e a energia do ruído.

$$SNR = \frac{E_{sinal}}{E_{ruído}} \quad (2)$$

Sendo que E_{sinal} é a energia do sinal e $E_{ruído}$ é a energia do ruído.

Neste trabalho, para o cálculo da SRN, foi levada em consideração a Equação 2.

7.3 Cálculo da SNR utilizando o Audacity e MatLab

Para calcular a energia de ambos os fatores, foi realizada a gravação das mesmas palavras que foram treinadas no módulo e com o mesmo nível de ruído ambiente.

Utilizando o Audacity®, programa totalmente gratuito e disponibilizado para gravações amadoras, foram gravadas as palavras para realização dos testes.

As gravações foram padronizadas em 3 segundos, sendo que o primeiro segundo era reservado para gravação apenas do ruído ambiente, e os outros dois, para a palavra de comando. Em seguida, eram gerados dois arquivos no formato WAV: um com o sinal completo (ruído + palavra) e outro contendo apenas o primeiro segundo da gravação (ruído). O arquivo contendo apenas o ruído foi utilizado para calcular a energia deste, permitindo assim o cálculo correto da SNR do sinal completo (ruído + palavra).

Para cálculo da energia de cada arquivo pré-gravado, foi utilizada a função `sum` do MatLab®. Em seguida, foi utilizada a Equação 2 para obter a SNR de cada microfone em cada situação. Para cada microfone, foram utilizadas as palavras: vermelho, verde, amarelo e azul. Como os valores de SNR para cada palavra com cada microfone foram praticamente os mesmos, foi calculada a média das quatro palavras analisadas. Os resultados obtidos são apresentados na Tabela 5.

Tabela 5. Resultados da SNR.

Microfone	Energia da palavra	Energia do ruído	Relação Sinal/Ruído
<i>HeadSet</i>	34.2164	0.4410	77.5935
<i>Original</i>	49.0009	0.4004	122.3750

Diante dos dados apresentados na Tabela 4, é possível observar que o microfone original do *Voice Recognition Module V2* ainda oferece uma SNR melhor que a do *headset* utilizado nos testes. Tal resultado sugere que esse microfone também fornecerá uma melhor taxa de reconhecimento. Na próxima seção, são apresentados os resultados dos testes de reconhecimento.

7.4 Testes de reconhecimento com o módulo e diferentes microfones

Para realização dos testes de reconhecimento, foi utilizada uma sala devidamente fechada, apenas com o ruído sonoro gerado por um ar-condicionado a uma distância de aproximadamente três metros do ponto onde estava o módulo.

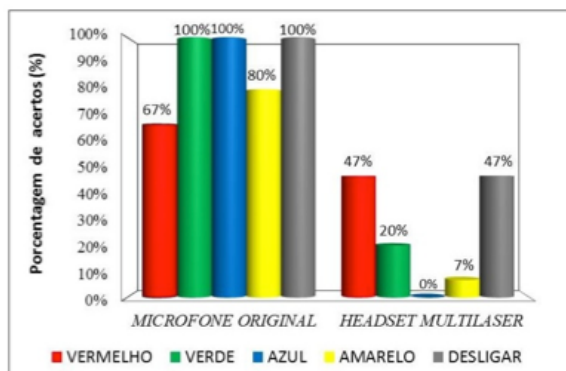
Foram realizadas 15 (quinze) repetições para as seguintes palavras: vermelho, verde, amarelo, azul, desligar. Cada palavra foi previamente treinada e gravada no módulo de reconhecimento.

A comunicação serial do módulo de reconhecimento foi configurada para 9600 bits/s. Nos testes,

a distância entre o locutor e o microfone era de aproximadamente 30cm.

Os resultados dos testes de reconhecimento para o microfone original do módulo e o do *headset* são apresentados na Figura 12.

Figura 12. Taxa de reconhecimento de palavras para 15 (quinze) repetições.



Como era esperado, depois dos testes de SNR, o microfone original ainda forneceu uma taxa de reconhecimento bem superior ao do *headset* analisado. Apesar de terem ocorrido alguns erros para as palavras “vermelho” e “amarelo”, a taxa de reconhecimento ainda pode ser considerada satisfatória.

8 Conclusões e considerações finais

Neste trabalho, foram realizados testes de reconhecimento de fala na plataforma Arduino com o *Voice Recognition Module V2*. Tal módulo permite incorporar, de maneira simplificada, a funcionalidade de realizar a interação com a aplicação desenvolvida por meio de comandos de voz. No entanto, ele possui algumas limitações como: pequeno vocabulário, dependência de locutor e sensibilidade ao ruído ambiente.

Apesar da plataforma Arduino ser *open-source*, alguns *shields*, ou módulos, desenvolvidos para tal plataforma, não seguem a mesma filosofia. No caso específico deste trabalho e do *Voice Recognition Module V2*, a impossibilidade de modificar o código fonte restringiu o objetivo inicial que era ampliar o tamanho do vocabulário e retirar a dependência de locutor. Diante disso, optou-se por tentar reduzir a sensibilidade ao ruído, realizando testes com um microfone com propriedades diferentes daquele que acompanha o *kit* originalmente.

Foram realizados testes comparativos de SNR e reconhecimento entre o microfone original (omnidire-

cional) e o de um *headset* (unidirecional). O resultado dos testes demonstrou que o microfone original ainda fornece uma melhor SNR e taxa de reconhecimento.

Alguns problemas foram encontrados para se realizar os testes com um número maior de microfones como a falta de especificação detalhada (diretividade, sensibilidade e frequência de operação) de alguns modelos, como também o tipo de conector que permitisse a ligação com o módulo de reconhecimento. Realizar testes com um microfone sem sua especificação não permitiria uma análise objetiva dos resultados, nem tão pouco a replicação por outros pesquisadores e/ou pessoas interessadas em utilizar o mesmo conjunto (módulo + microfone). A realização de testes com um maior número de microfones com especificações diferentes é uma sugestão de pesquisa futura. Também seria interessante realizar testes em um ambiente com um nível maior de ruído.

Ao final deste trabalho, pode-se concluir que o *Voice Recognition Module V2* é uma boa opção para quem deseja desenvolver uma aplicação na plataforma Arduino, incorporando a funcionalidade de reconhecimento de fala de maneira simples e barata. No entanto, é importante destacar que o reconhecimento oferecido possui limitações quanto ao tamanho do vocabulário e ao ambiente em que a aplicação será utilizada.

REFERÊNCIAS

- BARR, M. **Programming Embedded Systems in C and C++**. Sebastopol, CA: O’Reilly & Associates, 1999.
- FECHINE, J. M. **Reconhecimento automático de identidade vocal utilizando modelagem híbrida**: Paramétrica e Estatística. 2000. 212 f. Tese (Doutorado em Engenharia Elétrica) – Universidade Federal de Campina Grande, Campina Grande, 2000.
- FRANCIA, G. A. Embedded Systems Programming. **The Journal of Computing Sciences in Colleges**, v. 17, n. 2, p. 204-210. Proceedings of the 15th Annual CCSC Southeastern Conference, 2001.
- JIANG, X.; MA, M. Y.; CHEN, C. W. **Mobile Multimedia Processing**: Fundamentals,

Methods, and Applications. 1. ed.
Berlin: Springer, 2010. 287 p.

LATHI, B. P. **Sinais e Sistemas Lineares**. 2. ed.
Porto Alegre: Bookman, 2007.

MCRBERTS, M. **Arduino Básico**. São Paulo:
Novatec, 2011.

OLIVEIRA, J. R. **A aplicação de microfones em projeto digitais**. DCA-FEEC-Unicamp, 2010.

RABINER, L.; SCHAFER, R. W. **Theory and Applications of Digital Speech Processing**.
New Jersey: Prentice Hall, 2010. 1056 p.

SHAUGHNESSY, D. O. Interacting with computers by voice: automatic speech recognition and synthesis. **Proceedings of the IEEE**, v. 91, n. 9, p. 1272-1305, 2003.

SHEN, W. **Voice Recognition Module V2: speak to control (Arduino Compatible)** – Rev. B. Hong Kong: Elechouse, 2013.

SILVA, D. D. C. **Reconhecimento de Fala Contínua para o Português Brasileiro em Sistemas Embarcados**. 2011. 198 f.
Tese (Doutorado em Ciências no Domínio da Engenharia Elétrica) – Centro de Engenharia Elétrica e Informática, Universidade Federal de Campina Grande, Campina Grande, 2011.