

# INSTITUTO DE COMPUTAÇÃO

## UNIVERSIDADE ESTADUAL DE CAMPINAS

**IoT-Noise**  
**Dispositivo IoT de Baixo Custo para**  
**Detecção de Poluição Sonora**

*A. Filliettaz      E. Borin*

Technical Report - IC-16-14 - Relatório Técnico

December - 2016 - Dezembro

The contents of this report are the sole responsibility of the authors.  
O conteúdo do presente relatório é de única responsabilidade dos autores.

# IoT-Noise

## Um Dispositivo IoT de Baixo Custo para Detecção de Poluição Sonora

André Filliettaz      Edson Borin\*

### Resumo

Ruídos sonoros podem ser definidos como ondas onde há ausência de periodicidade, de forma que as vibrações no meio não são harmônicas [5]. Altos níveis de ruído podem causar estresse ou mesmo danos à saúde das pessoas, sendo assim definidos como poluição sonora. Locais de trabalho como usinas, aeroportos, indústrias, construção civil, *call centers* ou mesmo lugares cotidianos, como *shoppings centers*, salas de aula, avenidas ou ruas, são alguns exemplos de onde a sociedade está sujeita aos efeitos, como estresse, falta de sono e em casos mais extremos, perda auditiva, que podem ser causados pela poluição sonora.

Sob este contexto, foi desenvolvido um dispositivo de baixo custo (um microfone simples acoplado a uma placa Arduino e um módulo WiFi), chamado IoT-Noise, que permite a obtenção de dados a serem entregues para análise a um sistema mais complexo, transmitindo dados através da internet, de forma que este possa tomar decisões de controle adequadas visando um melhor gerenciamento do ambiente e maior segurança, já que o ouvido humano possui um limite para que não haja perda auditiva. Além disso, principalmente em áreas urbanas, existem locais como hospitais e escolas, aonde por lei, existe um limite para o nível de poluição sonora e buzinas são proibidas.

**Palavras-chave:** IoT, Som, Poluição Sonora, Decibelímetro, microcontroladores, ESP8266, Arduino, Sistemas Embarcados

## 1 Introdução

Pretende-se documentar todos os conhecimentos e instrumentos necessários para se desenvolver o IoT-Noise, bem como um sistema simples que o usa para coleta de dados acerca da poluição sonora, processamento, obtendo-se a intensidade sonora, e envio desses dados a um servidor central utilizando a internet. Dessa forma, essas informações estariam disponíveis em um servidor para que fossem analisadas.

---

\*Instituto de Computação, Universidade Estadual de Campinas, 13081-970 Campinas, SP.

## 1.1 Motivação

Som pode ser definido como uma onda mecânica e longitudinal, ou seja, que necessita de um meio mecânico como o ar, água ou rochas, e na qual a oscilação das partículas do meio se dá na mesma direção de propagação da onda.

Alguns ambientes são propensos a estarem expostos a níveis altos de poluição sonora, como por exemplo: *call centers*, estúdios, salas de aulas, *shoppings centers* etc. Com isso em mente, principalmente em áreas urbanas, deve-se considerar que exposições a sons de alta intensidade podem levar a condições prejudiciais à saúde do indivíduo. Propõe-se aqui um dispositivo que possa funcionar como um decibelímetro e que tenha a capacidade de se comunicar a um sistema central, enviando dados pré processados, para que um sistema maior possa com isso tomar alguma decisão de controle.

## 1.2 Objetivos

Este projeto tem como objetivo o desenvolvimento, implementação e testes de um decibelímetro de baixo custo que possa enviar dados a um servidor central através da internet. Apesar do servidor não fazer parte do dispositivo, para fins de teste e para se ter uma ideia de possíveis usos, foi implementado um servidor utilizando Express do NodeJS, o qual se comunica com todos os dispositivos que processam o sinal sonoro.

Dessa forma, a Figura 1 abaixo mostra um diagrama simples de como funciona um sistema que faz uso dos dispositivos:

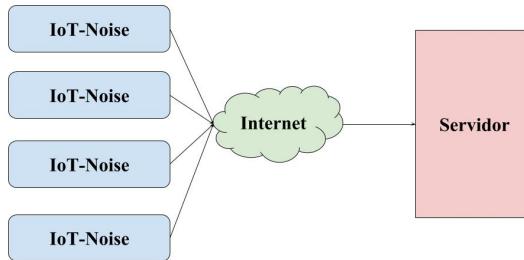


Figura 1: Diagrama do Sistema.

## 1.3 Componentes

### 1.3.1 Arduino

O microcontrolador escolhido para realizar toda a parte de coleta, processamento e envio de dados foi um Arduino. Trata-se de uma plataforma eletrônica open-source, desenvolvida

para prototipagem com a intenção de ser extremamente simples e de baixo custo, de forma que qualquer usuário possa desenvolver projetos.

O Arduino tem sido largamente utilizado por sua programação fácil, sem a necessidade de se preocupar com protocolos de comunicação, ou interfaces seriais. Tem sido muito usado em robótica, controlando sensores e dispositivos físicos, e por ser *open-source*, existem muitos projetos baseados em arduino e a comunidade de usuários é grande, de forma que é muito simples encontrar informações a respeito do mesmo na internet. A própria interface de programação fornece uma grande gama de exemplos. Apesar de oferecer comunicação fácil com sensores, as placas arduino não possuem comunicação com rede de forma direta, necessitando-se um módulo externo, que faça essa comunicação com a rede.

Dentre os vários modelos disponíveis no mercado, decidiu-se utilizar a placa Arduino Uno, devido ao seu alto custo benefício (podem ser encontrados modelos por até R\$30,00), tamanho compacto e comunicação simples, tanto com o microfone bem como o módulo de comunicação WiFi. Possui 14 entradas digitais e 6 analógicas, que são mais que o suficiente, e é baseado no controlador ATMega328P, que possui as seguintes características:

- 32kBytes de Flash;
- 20Mhz de Frequência Máxima de Operação;
- 8-bits AVR de CPU;
- ADC (Analog/Digital Converter) de 10 bits de resolução;

Neste projeto, o arduíno é responsável por fazer a coleta dos dados utilizando um microfone, seu processamento e então enviá-los a um servidor utilizando o módulo externo.

### 1.3.2 ESP8266 ESP-01

Um simples módulo pode ser utilizado para a comunicação via o protocolo WiFi. Pode ser utilizado por um microcontrolador externo ou ser programado diretamente sem o uso de um microcontrolador. No projeto, ele foi utilizado unicamente para permitir a comunicação do arduino com o servidor. Os valores captados pelo microfone são processados pelo arduino, obtendo-se o valor em dB da intensidade sonora e então, via comunicação serial e comandos AT, o arduino transmite esses dados para o módulo para que sejam enviados para o servidor através da internet. Pode ser encontrado na internet por até R\$15,00.

### 1.3.3 Captação Sonora

A captação de som é feita através de um microfone de eletreto em conjunto de um circuito amplificador. Microfones são de maneira simples, dispositivos que convertem a pressão sonora em um sinal elétrico [2]. Entretanto esses valores de voltagem são extremamente baixos, e por isso faz-se necessário o uso de um circuito amplificador, para que o arduino possa detectar essas variações no nível de tensão.

A Figura 2 mostra o esquema do circuito amplificador empregado. No projeto final, optou-se por comprar um circuito integrado com essa funcionalidade (R\$15,00) com o objetivo de se economizar espaço, entretanto, os componentes do circuito não são difíceis de serem encontrados, e se comprados individualmente, não passam de R\$7,50. O microfone do circuito é do tipo condensador de eletreto, que é um tipo de microfone simples, barato e fácil de ser encontrado. Possui uma resposta boa para uma gama muito grande de frequências, tem menos ruído do que microfones dinâmicos e é utilizado na maioria de estúdios para gravação[2].

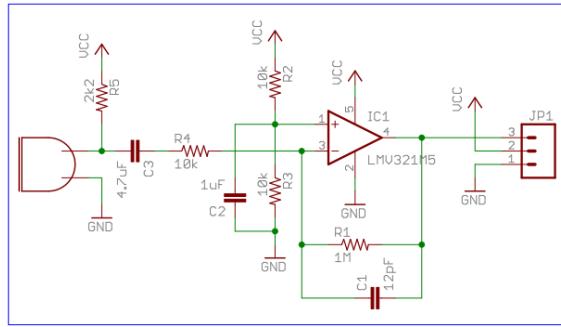


Figura 2: Diagrama do Microfone Amplificador.

A Figura 3 ilustra como funciona um microfone de eletreto, sua montagem e componentes. O som (energia acústica) faz com que a fina membrana vibre, de maneira a gerar uma variação na capacitância do mesmo, o que leva a uma variação na tensão do mesmo (ou seja, uma variação de energia elétrica).

Isso levanta um ponto que será analisado posteriormente, existe um limite inferior de intensidade para o qual a vibração da membrana será detectada. Logo existe uma sensibilidade que deve ser levada em consideração quando for se calcular o valor da intensidade sonora, para valores abaixo dessa intensidade, o microfone não mostrará uma variação de tensão que seja detectável.

### 1.3.4 Display LCD (Opcional)

O display não é necessário na montagem final, mas pode ser utilizado para fins de verificação de funcionamento, mostrando alguns dos valores medidos, e depuração de alguns problemas, como por exemplo falta de conexão com a internet e impossibilidade de comunicação com o servidor para o qual os dados são enviados. Seu preço estimado é de R\$20,00.

### 1.3.5 Componentes Adicionais

Além dos componentes listados acima, também foram utilizados uma protoboard (R\$16,00), 1 bateria de 9V, 2 pilhas AA e *jumpers* para a conexão dos componentes. Assim, o circuito completo pode ser montado com aproximadamente R\$100,00.

Também foi utilizado um decibelímetro comercial para se obter um valor de referência. O modelo utilizado foi o *ITDEC-4000*, da marca *Instrutemp*. Foi escolhido um ambiente

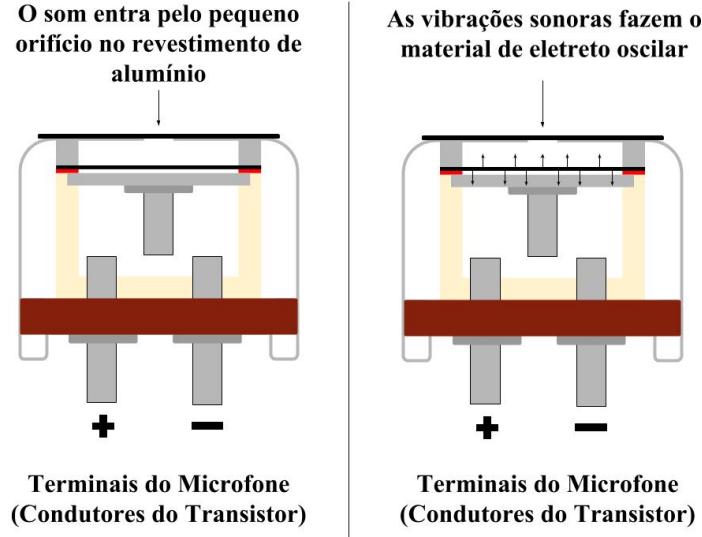


Figura 3: Microfone de Eletreto.

de calibração silencioso durante a madrugada, buscando sempre o menor nível de ruído possível.

## 2 Fundamentos Teóricos

### 2.1 Internet of Things (IoT)

O conceito de Internet das Coisas (IoT) pode ser summarizado como a conexão, comunicação e troca de dados de objetos físicos cotidianos, veículos, dispositivos eletrônicos simples, sensores e sistemas computacionais, de forma que qualquer objeto poderia estar conectado à rede [1].

Apesar do conceito ser relativamente novo, grandes avanços na área foram possíveis com os sistemas embarcados, dispositivos e sensores menores, de baixo custo e mais eficientes no consumo de energia. Espera-se que no futuro, veículos e construções estejam todas conectadas de alguma forma a internet, de maneira a trazer benefícios dessa troca de dados.

Pretende-se que o IoT-Noise colete dados de intensidade sonora e comunique-se com um servidor utilizando a Internet, enviando os dados processados. Para fins de teste, foi desenvolvido um servidor simples em Express. Ao receber requisições HTTP de POST, o servidor armazena os dados enviados pelo IoT-Noise em arquivos no formato .CSV. Quando recebe requisições GET, o servidor envia os arquivos com os dados de intensidade sonora coletados.

## 2.2 Ondas Sonoras

Na física, o som é uma vibração que se propaga como uma onda mecânica, normalmente audível, de pressão e deslocamento, através de um meio como o ar ou a água [5]. Os seres humanos podem ouvir ondas sonoras com frequências entre cerca de 20 Hz e 20 kHz. Outros animais têm faixas de audição diferentes. Ao contrário da luz, o som necessita de um meio material para se propagar, como o ar, água ou mesmo rochas. Ondas longitudinais são todas as ondas cujo movimento das partículas (no caso, do meio) se movem em sentido paralelo ao da propagação da onda.

Assim como a luz, o som possui um conjunto de grandezas que definem suas características, como por exemplo:

- **Frequência:** número de oscilações por unidade de tempo, medido em Hz;
- **Período:** tempo que a onda leva para realizar uma oscilação;
- **Comprimento:** distância (paralela à direção de propagação) entre as repetições de forma de onda
- **Amplitude:** módulo do deslocamento máximo a partir da posição de equilíbrio de um elemento enquanto a onda atravessa o meio.

A Intensidade Sonora é uma função da frequência e amplitude da onda sonora, e é medida com o uso da escala logarítmica decibel, largamente utilizada para se expressar a razão entre duas quantidades físicas.

## 2.3 Intensidade Sonora

A intensidade  $I$  de uma onda sonora em uma superfície é definida como a taxa média com a qual a energia da onda atravessa a superfície ou é absorvida por ela. Quando uma onda se propaga por um meio (como por exemplo o ar), suas vibrações produzem alterações na pressão do meio, que é captada por uma área (ouvido humano). Essas variações de pressão definem a chamada Pressão Sonora. Define-se Intensidade Sonora de acordo com a equação [5]:

$$I = \frac{P}{A} \quad (1)$$

Onde  $P$  é a potência, ou seja a taxa de temporal de transferência de energia da onda para a superfície na qual incide, e  $A$  é a área dessa superfície. O ouvido humano pode captar uma enorme faixa de valores de intensidades sonoras, e percebe cada uma delas de forma logarítmica. Dessa maneira, o dispositivo final utiliza-se da seguinte equação para representar a intensidade sonora:

$$SPL = 10 \log \left( \frac{I}{I_0} \right) \quad (2)$$

A essa grandeza, atribui-se o nome de Nível de Pressão Sonora (em inglês, Sound Pressure Level - SPL).  $I_0$  é a chamada de Intensidade de Referência, e seu valor é 10-12 W/m<sup>2</sup>, que é conhecido como o limiar inferior da audição humana. A Potência de uma onda em termos da variação de pressão é obtida pela seguinte equação [6]:

$$P = \frac{Ap^2}{\rho v} \cos\theta \quad (3)$$

Aonde,  $p$  é a pressão,  $A$  é a área de incidência,  $\rho$  a densidade do meio,  $v$  a velocidade de propagação do som e  $\theta$  o ângulo de incidência da onda na superfície. Combinando-se as (2) e (3) obtemos o valor de SPL em função da variação de pressão:

$$SPL = 20dB * \log\left(\frac{p}{p_0}\right) \quad (4)$$

Os microfones são transdutores que convertem energia acústica em energia elétrica [2]. Uma de suas características é justamente a Sensitividade, ou seja, qual a voltagem de resposta de um microfone para uma determinada pressão aplicada na membrana do microfone. Juntando isso a equação (4), podemos obter a seguinte equação para o circuito:

$$SPL = 20dB * \log\left(\frac{V}{V_0}\right) \quad (5)$$

Aonde  $V$  é a voltagem obtida quando se é aplicada uma pressão  $p$  no microfone e  $V_0$  é a voltagem obtida quando se aplica a pressão de referência no microfone. Pelas propriedades do logaritmo, pode-se manipular a equação (5) para atribuir-se um determinado valor SPL-ref de referência, obtido no processo de calibração, assim, mediu-se um valor de intensidade sonora (SPL) com um decibelímetro comercial e anotou-se a tensão obtida para tal nível, obtendo-se assim a equação final utilizada pelo IoT-Noise para se determinar o valor da Intensidade Sonora:

$$SPL = 20dB * \log\left(\frac{V}{V_{ref}}\right) + SPL_{ref} \quad (6)$$

A medida de voltagem utilizadas para a determinação da Intensidade Sonora foi a *Root Mean Square*, ou RMS, definida como a raiz quadrada da média aritmética dos quadrados das medidas de amplitude.

Um outro detalhe que cabe ser observado é a resposta do ouvido humano às diferentes frequências. É conhecido que o ouvido humano é mais sensível a ruídos agudos, de forma que vários tipos de filtros que estabelecem pesos diferentes para determinados valores de frequência, tentando assim imitar o comportamento do ouvido humano, já foram propostos. Para o este projeto, decidiu-se utilizar o filtro de peso A (A-Weighting). A Figura 4 mostra o filtro utilizado.

## 2.4 Transformada Discreta de Fourier (DFT)

A DFT é um instrumento fundamental na análise e processamento de sinais, e por essa razão ela foi extremamente necessária para a determinação do valor dB da intensidade sonora [3].

De maneira simples, a DFT decompõe uma função no domínio do tempo (no caso o sinal captado pelo microfone) em uma soma de sinais de diferentes frequências e intensidades, dessa forma fazendo a função ser descrita no domínio da frequência. Pode-se pensar de maneira análoga a como um acorde musical pode ser expresso como uma soma de amplitudes das notas que o compõe [4].

Inicialmente, um algoritmo inocente foi utilizado para se obter a intensidade do sinal detectado. Esse algoritmo inicial não levava em consideração as diferentes frequências e

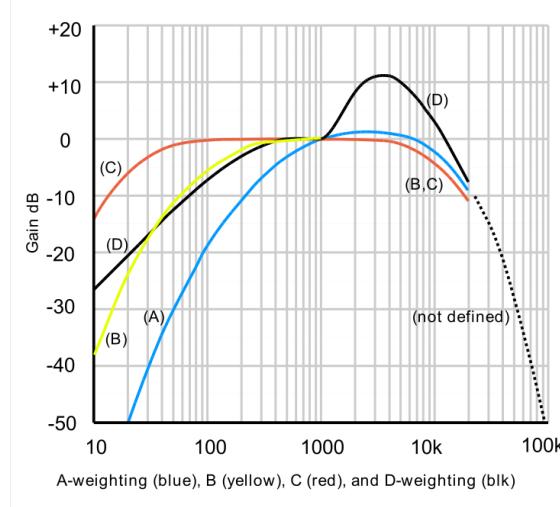


Figura 4: Filtros de peso acústico. Utilizou-se o filtro A (azul).

portanto não utilizava o DFT. Com testes mais profundos, ficou evidente a necessidade da transformada, e por isso o algoritmo final faz uso do FFT (Fast Fourier Transform - Transformada Rápida de Fourier), que é uma implementação eficiente,  $O(n\log n)$ , da DFT [?].

Com o espectro de frequências obtido, é possível calcular de maneira precisa o valor RMS da amplitude do sinal amostrado. Obtém-se o valor do SPL (*Sound Pressure Level* - Nível de Pressão Sonora), utilizando-se (6), que é precisamente a medida de intensidade sonora buscada no projeto.

### 3 Metodologia

#### 3.1 Dispositivo Proposto

Para solucionar o problema, propõe-se um circuito composto de 4 estágios. O circuito proposto precisa captar o sinal sonoro, processá-lo para se conseguir o espectro de frequências, determinar o seu valor em decibéis e então enviá-lo para o servidor. Se ligado a um display LCD, ele pode mostrar os resultados no mesmo. A Figura 5 mostra uma representação dos estágios do circuito na forma de diagrama de blocos.

#### 3.2 Aquisição do sinal

A parte de aquisição de sinal é feita através do circuito detector de som. Para isso, o circuito utiliza um microfone de eletreto, facilmente encontrado em celulares e circuitos que fazem captação sonora em dispositivos eletrônicos.

O microfone possui características que são importantes na hora de se implementar o decibelímetro, sobretudo o valor de sua sensitividade. Esse valor determina o quão sensível o microfone é ao som ao seu redor, e a partir de qual limiar de intensidade ele passa a

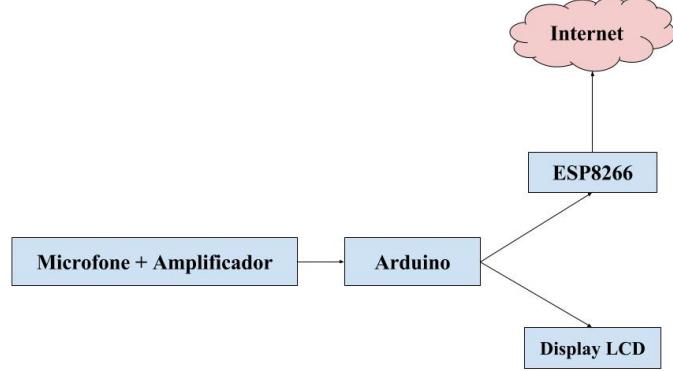


Figura 5: Diagrama do Dispositivo IoT-Noise.

converter a pressão em um sinal elétrico. Microfones mais sensíveis respondem a menores intensidades sonoras.

Essa consideração é importante, pois determina o valor mínimo que pode ser medido pelo dispositivo.

O microfone é alimentado com 5V pela saída do próprio arduino, assim gerando sinais entre 0V-5V, com *offset* de 2.5V, ou seja, metade do valor de Vcc. O valor do *offset* é importante, pois o ADC (Analog-Digital Converter - Conversor Analógico Digital) do arduino (no modelo Uno, possui 10bits, convertendo o sinal analógico para um número entre 0 e 1023) não lida com valores negativos de tensão. Um sinal sonoro é periódico por se tratar de uma onda, então isso faria com que o ciclo negativo gerasse tensões negativas, comprometendo assim as leituras. Os valores de tensão emitidos pelo microfone oscilam em torno de 2,5V então.

### 3.3 Processamento de Dados Obtidos

Três algoritmos diferentes foram tentados para se fazer o processamento e determinação do sinal sonoro, cada um com um grau diferente de complexidade e dificuldade. Essa abordagem foi adotada devido ao poder de processamento do Arduino e de outros micro-controladores, que em geral é bem inferior ao processamento com o qual se está acostumado com máquinas de uso pessoal.

Antes de se detalhar melhor cada algoritmo, cabe aqui esclarecer uma etapa de extrema importância, comum a todos eles: a calibração. Qualquer decibelímetro, seja ele comercial, ou mesmo simples aplicativos de celular, precisam passar por uma etapa de calibração. Alguns decibelímetros são vendidos com o certificado de calibração e são os mais caros do mercado, e utilizados em operações que exigem enorme precisão. Para se calibrar um

decibelímetro, normalmente se utiliza um calibrador, que emite uma onda sonora de 1000 Hz a uma intensidade específica e assim, se obtém uma intensidade de referência.

Calibradores também são instrumentos extremamente caros, e por isso, buscou-se aqui uma forma de se simular uma calibração da melhor forma possível. O que se fez foi utilizar um decibelímetro e aproximá-lo juntamente com o microfone do dispositivo, de uma fonte sonora que emitia um sinal de 1000 Hz a 45 db. Com isso foi estabelecida uma referência para medidas, e espera-se que as medidas obtidas pelo decibelímetro e o dispositivo sejam iguais.

Uma breve explicação de cada um dos algoritmos é dada a seguir:

- **Média de leituras:** o algoritmo mais simples e inocente simplesmente fazia um certo número de medidas do sinal, durante um certo intervalo de tempo. Obtinha-se então a média desses valores, e usando uma média de referência, calculava-se o valor da intensidade sonora. Esse algoritmo não mostrou bons resultados, quando a intensidade se distanciava demais da intensidade de referência, os valores passavam a diferir muito dos medidos com um decibelímetro de referência.
- **Aproximação Polinomial e Logarítmica:** esse algoritmo trata-se de uma certa sofisticação do anterior. Várias medidas foram tiradas durante um certo intervalo de tempo. Foi feita então uma média desses valores e com isso, foi feita uma aproximação polinomial e outra logarítmica. Com os coeficientes obtidos, tentou-se então estimar o valor da intensidade sonora para um valor de tensão qualquer obtido. Novamente, esse método não mostrou-se eficaz, apesar das medidas ainda serem precisas para valores próximos aos de referência, elas logo se afastavam dos valores obtidos pelo decibelímetro de referência quando a intensidade crescia ou diminuía demais.
- **Espectro de Frequência e FFT:** este algoritmo é o mais complexo e traz uma série de mudanças de abordagem quando comparado com os anteriores. De maneira simplificada, o que foi feito é uma coleta de 64 amostras do sinal. Depois disso, com a ajuda de uma biblioteca de código externa, uma implementação da FFT foi aplicada ao sinal, resultando assim no seu espectro de frequências. Após isso ser feito, era aplicado um peso para cada frequência, seguindo o Filtro A de peso acústico, explicado anteriormente e ilustrado na Figura 4. Com esses dados em mãos, era então calculado o valor de tensão RMS do sinal, ou seja, VRMS. Por fim, um filtro de peso temporal era aplicado sob cada valor RMS, para assim favorecer mais as medidas recentes do que as mais antigas. Esse valor RMS final era utilizado para se determinar o valor dB da intensidade sonora, e com esse algoritmo, resultados satisfatórios foram obtidos.

Cada algoritmo possui seus custos, vantagens e desvantagens, mas apenas o terceiro ofereceu resultados que podem ser considerados satisfatórios, dessa forma, a terceira abordagem foi a escolhida para o projeto final, mesmo trazendo um maior custo de processamento e memória. Para se aplicar o FFT, além é claro da própria biblioteca que realizava a transformada, também foram utilizadas duas bibliotecas, uma responsável pela amostragem do sinal e outra responsável por utilizar o espectro para calcular a intensidade sonora propriamente dita.

Para que a FFT possa ser utilizada, alguns parâmetros deveriam ser fixados e algumas condições alcançadas. Primeiramente, a frequência com que as amostras são coletadas (chamada de FR) deve ser fixa. Para se alcançar uma frequência de leitura da entrada analógica fixa, foi adicionada uma lib para cuidar apenas disso. FR foi ajustada para 22050Hz, de forma a fornecer um espectro de até 11025Hz para a FFT. Outro parâmetro a ser configurado, o número de amostras, deve ser escolhido de forma ser uma potência de 2, para que o algoritmo Cooley-Tukey [4], utilizado para implementar a FFT, pois isso traz uma complexidade  $O(N \log N)$ . Foi escolhido o valor de 64, ou seja 26, amostras para realizar o algoritmo.

### 3.4 Envio de Dados

O envio dos dados era realizado com o auxílio do *chip* ESP8266, um módulo de uso simples, pensado para aplicações IoT e muito mais barato do que as opções comerciais.

Após os dados serem processados, o Arduino envia comandos ao ESP8266, para que este transfira os dados para o servidor via conexão TCP. O servidor ao receber os dados, os gravam num arquivo .CSV para que estes sejam analisados futuramente ou acessados por um cliente.

Algumas coisas devem ser levadas em consideração quando o módulo for instalado. Primeiro de tudo, a tensão de operação do ESP8266 é 3.3V. Isso significa que aplicar 5V ao seus terminais pode resultar em danos físicos ao chip, o que muitas vezes é difícil de se detectar. Apesar de não se ligar os terminais de força a 5V, os terminais do arduino operam com 5V, o que significa que a saída serial do arduino interpreta o valor lógico 1 como 5V, de forma que ligar o módulo diretamente ao arduino pode, novamente, danificar o módulo.

Além disso, o ESP8266 requer uma quantidade de corrente elétrica muito grande, o que não é fornecido pela saída de 3.3V da maioria dos protocolos. A solução adotada para ambos os problemas foi o uso de um chip conversor de nível lógico, que simplesmente transforma um sinal numa tensão alta para o sinal equivalente numa tensão menor. Isso também resolve o problema de corrente, pois a saída de 5V do arduino pode fornecer uma corrente de mais de 500mA quando ligado a uma fonte externa (500mA quando ligado a porta USB do computador).

A Figura 6 é uma representação da montagem final do circuito do dispositivo.

Apesar de o Display LCD não ser parte fundamental no funcionamento do dispositivo, é interessante que ele seja utilizado pois informações importantes como as medidas feitas, conexões e envios de dados bem sucedidos e mensagens sobre o status do dispositivo podem ser visualizadas através dele.

Uma primeira montagem utilizava um par de pilhas para alimentar o módulo ESP8266 e um divisor de tensão na entrada do RX do mesmo (*jumper* azul na Figura 6). Apesar de ser possível esta montagem, e largamente difundida na Internet, ela não é recomendada. Por diversas vezes o módulo não apresentou o funcionamento correto, além de problemas com o casamento das tensões entre o arduino e o módulo.

Outro detalhe a ser observado na montagem é a corrente fornecida pelo Arduino. O modelo Uno pode fornecer até 500mA pela saída 5V (podendo aumentar quando alimentado por uma fonte externa como baterias) e 50mA pela de 3.3V [16]. O ESP8266 necessita de

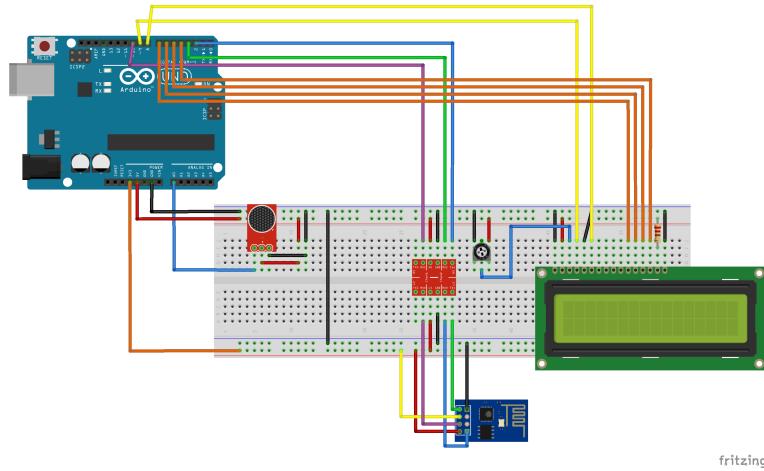


Figura 6: Montagem final do IoT-Noise.

até 250mA, o que faz com que a ligação direta com a saída do arduino apenas não seja o suficiente, o que faz com que o uso do conversor seja ainda mais conveniente.

## 4 Resultados

Com o auxílio de um decibelímetro externo, o dispositivo foi testado em diferentes ambientes para se verificar o comportamento do mesmo sob diferentes intensidades sonoras e sob diversos tipos de ruídos e sons. Além de se verificar o seu comportamento, ou seja, é necessário que a resposta da intensidade sonora seja logarítmica, outro ponto que foi observado foi justamente se os valores medidos entre os dois decibelímetros seria compatível em qualquer tipo de ambiente, já que os dois algoritmos testados previamente diferiam muito dos valores do decibelímetro comercial quando a intensidade diferia demais do valor tomado como referência.

Alguns dos ambientes selecionados para se realizar os testes foram:

- Apartamento, Centro, Campinas, SP - dados coletados durante a madrugada;
  - Apartamento, Centro, Campinas, SP - dados coletados em sala com televisão ligada;
  - Residencial, Santo André - dados coletados com sistema de som ligado;
  - Culto de Igreja, Sara Nossa Terra de Vila Alpina, ministração com jovens - (18h30 - 21h00);
  - Sala EM20, Bloco A, Faculdade de Engenharia Mecânica, Universidade de Campinas - em horário de aula (21h - 22h);
  - Praça de Alimentação, Parque Dom Pedro Shopping, 21h;
  - Sala de Cinema 12, Kinoplex, Parque Dom Pedro Shopping;

Em cada um dos ambientes escolhidos, foram coletadas amostras de alguns minutos, e estas foram processadas. Espera-se conseguir algumas conclusões, baseado no local, horário e atividade sendo feita. Como o decibelímetro de referência não foi calibrado profissionalmente, a Tabela 1 mostra alguns dados retirados da literatura [5] quanto a valores de intensidades sonoras:

Tabela 1: Níveis sonoros em dB

Limiar da audição	0 dB
Farfalhar de folhas	10 dB
Conversa	60 dB
Show de rock	110 dB
Limiar da dor	120 dB
Turbina a jato	130 dB

Fonte: Halliday et al, 2008, p.159.

Com estes valores em mente, faz-se agora menção ao problema de sensibilidade, levantado na seção de [1.3.3] quando foi falado sobre a captação de som no dispositivo e o uso do microfone de eletreto. Existe um valor mínimo de intensidade para o qual o microfone passa a responder. Durante a calibração, foi observado que o microfone passava a responder com valores consistentes a partir de 33 dB no decibelímetro utilizado de referência, apesar desses sinais serem um pouco oscilantes ainda. Por isso adotou-se o valor de 45 dB para ser o valor base. Isso quer dizer que para valores menores do que 30 dB o dispositivo começa a apresentar valores instáveis.

Segundo a Tabela 1, uma conversa entre pessoas afastadas 1m gira em torno de 60 dB, de forma que o microfone é sensível o suficiente para a maioria dos ambientes cotidianos para os quais o dispositivo foi projetado.

#### 4.0.1 Apartamento, Centro - Campinas

Dois tipos de coleta de dados foram feitos no apartamento: uma coleta durante a madrugada e outra na sala de estar com a televisão ligada. O primeiro visava observar os níveis sonoros durante a madrugada, quando espera-se que tudo esteja silencioso no apartamento. O segundo conjunto de dados foi coletado na sala de estar do mesmo, quando haviam pessoas e a televisão estava ligada, sendo uma situação cotidiana comum e esperada numa casa de família.

Durante a madrugada, na sala silenciosa, devido ao problema de sensibilidade, como foi esperado, os valores medidos não atingiram o mínimo para serem detectados pelo microfone utilizado no projeto do IoT-Noise. O valor de intensidade medido pelo decibelímetro de referência obteve uma média de 19dB, muito abaixo do valor mínimo obtido durante a calibração. O IoT-Noise não pode definir o valor exato da Intensidade Sonora neste cenário de teste, entretanto ele identificava que o valor medido era muito abaixo do valor de referência.

Por fim, a Figura 7 mostra os dados coletados na mesma sala de estar quando a televisão estava ligada em volume médio.

Para este teste, a média dos valores obtidos pelo decibelímetro foi de 40dB.

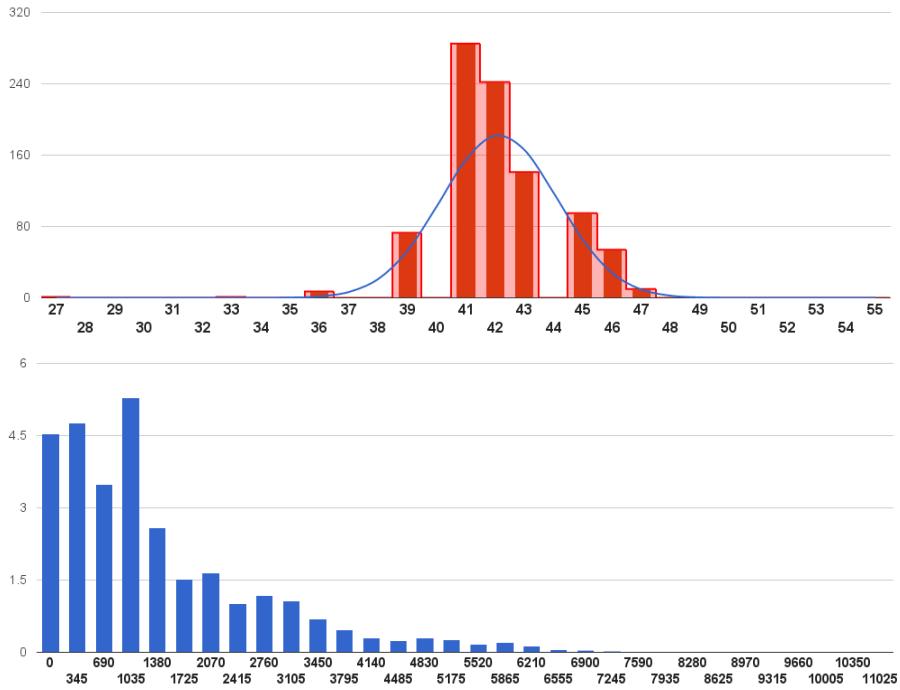


Figura 7: Histograma e Espectro Médio da amostra em sala com televisão ligada.

#### 4.0.2 Residência, Campestre - Santo André

Este teste foi feito com o objetivo de colher grandes quantidades de dados durante vários períodos do dia com o objetivo de verificar a consistência do dispositivo quanto ao decibelímetro, bem como sua resposta a diferentes intensidades sonoras vindo de diferentes fontes. Com isso verificou-se que as respostas do dispositivo e do decibelímetro utilizado para a calibração foram consistentes.

O primeiro teste foi feito num ambiente semelhante ao usado para se fazer a calibração, silencioso. Então colocou-se música em alto-falantes e captou-se os valores durante toda a execução da música, aproximadamente 4min07s, e mais um período de 30s após o término da mesma, de forma a poder-se verificar o tempo, já que quando não havia nenhuma música sendo tocada o microfone não registrava nenhum som. Foram feitas uma média de 3 captações por segundo.

Devido às condições de mínimo som ambiente possível, o teste foi realizado durante a madrugada, o que não possibilitou que grandes intensidades sonoras fossem testadas, assim sendo, a média da intensidade sonora medida pelo decibelímetro de referência girou em torno dos 50 dB.

Os resultados obtidos pelo dispositivo são mostrados na Figura 8 a seguir:

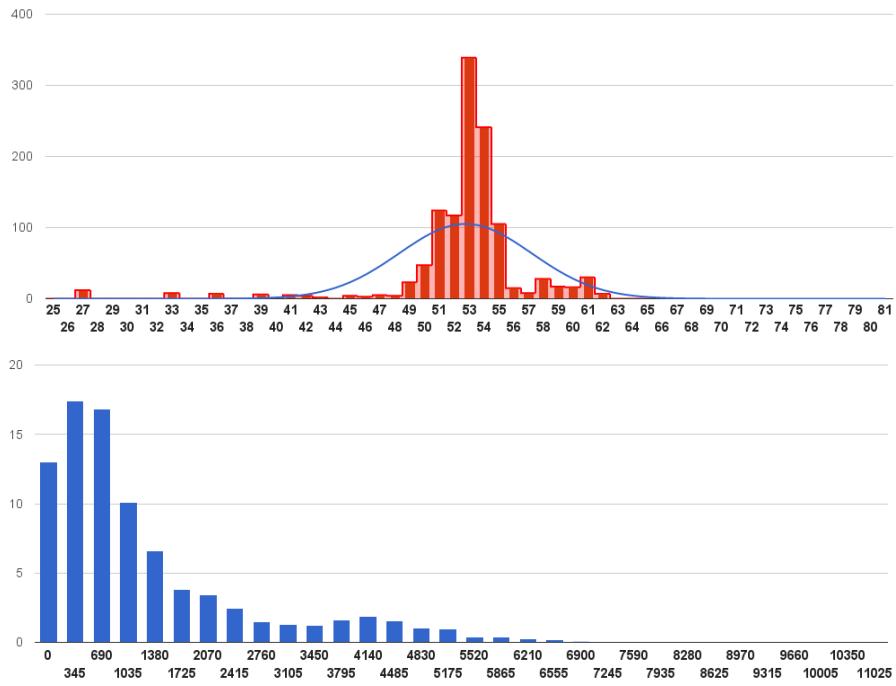


Figura 8: Histograma e Espectro Médio da amostra em sala com música.

#### 4.0.3 Culto de Igreja

Este teste foi feito com o objetivo de se testar a resposta e consistência do IoT-Noise em ambientes aonde as pessoas são expostas a níveis mais altos e constantes de intensidade sonora. As medidas foram tomadas em dois momentos diferentes da reunião. Num primeiro instante, coletou-se dados durante o momento em que as músicas eram tocadas. Havia instrumentos como guitarras, teclados e bateria sendo tocados em amplificadores de som, de maneira que o ambiente estava repleto de som.

Num segundo momento, a coleta foi feita durante a palestra, de forma que o ambiente não estava carregado de sons, mas com a voz do ministrante sendo amplificada por um microfone e caixas amplificadoras. Houve então uma grande diferença de momentos, com o objetivo de expor o IoT-Noise a testes diversos.

A Figura 9 mostra os dados coletados pelo IoT-Noise enquanto as músicas estavam sendo tocadas e uma média das frequências obtidas com a FFT.

A Figura 10 apresenta os dados obtidos durante a palestra, sem música sendo tocada. Nota-se que a média de Intensidade Sonora medida com o decibelímetro comercial para o culto com música ao vivo sendo tocada foi de 74 dB, enquanto que durante a palestra a média observada foi de 58 dB.

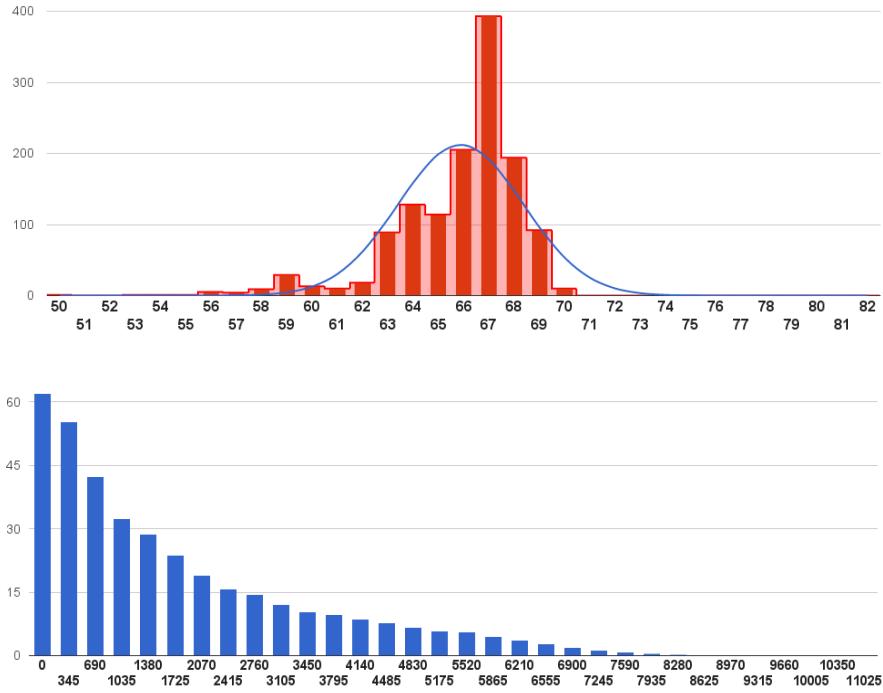


Figura 9: Histograma e Espectro Médio da amostra em culto com música ao vivo.

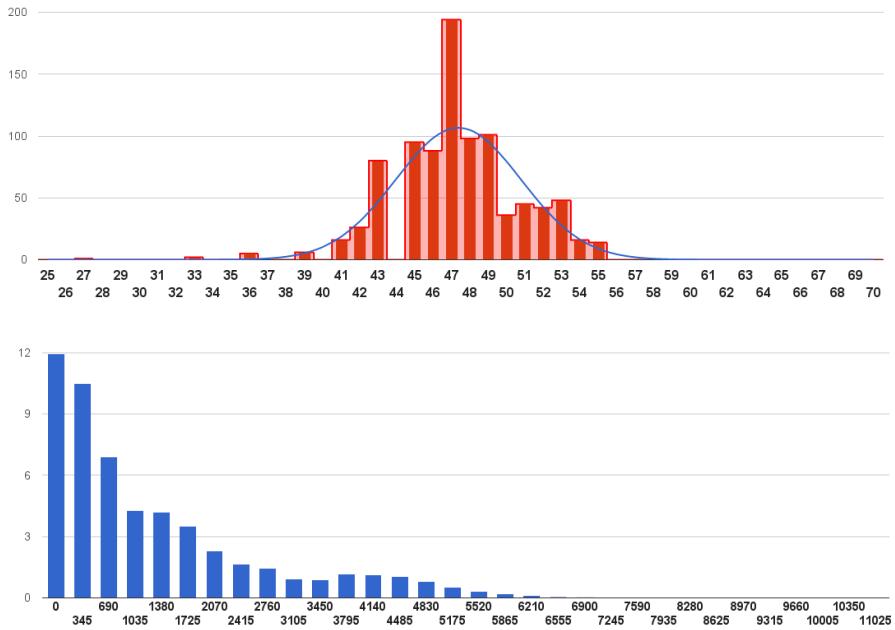


Figura 10: Histograma e Espectro Médio da amostra em culto durante palestra.

#### 4.0.4 Sala EM20, FEM, Universidade de Campinas - Horário de Aula

Para este teste, a sala EM20 da mecânica foi escolhida por possuir um sistema sonoro amplificado, de forma que o professor pode usar um microfone para ministrar suas aulas. Além disso, a sala possui um tratamento acústico e som ambiente igualmente distribuído, de forma que o som é praticamente o mesmo em qualquer ponto que se sente na sala de aula.

A Figura 11 mostra o histograma juntamente com a média do espectro de frequências da amostragem em sala de aula são apresentados.

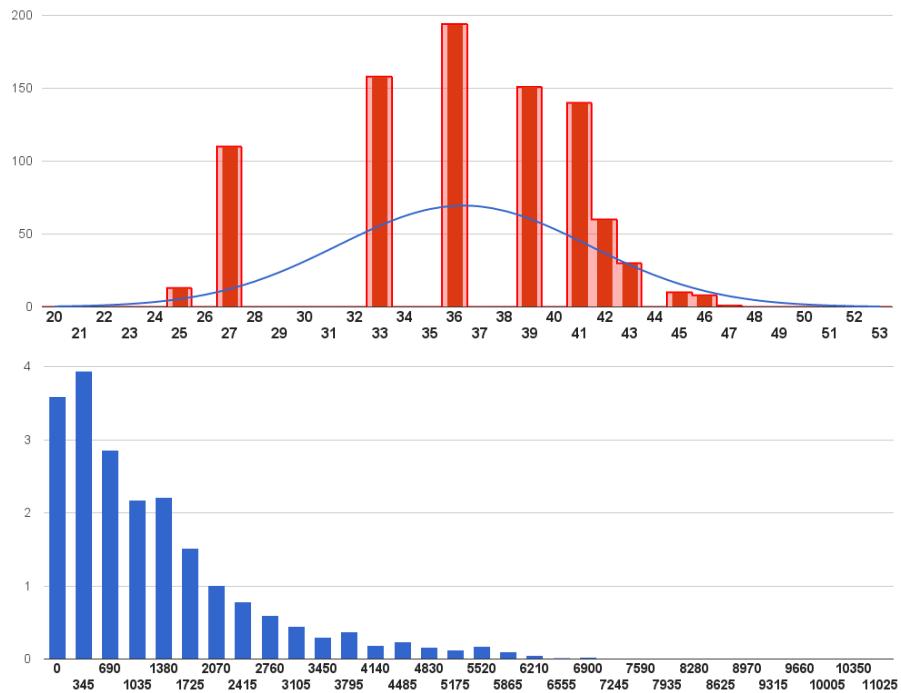


Figura 11: Histograma e Espectro Médio da amostra na sala EM20, durante aula.

#### 4.0.5 Praça de Alimentação do Parque Dom Pedro Shopping

Para este teste, o horário escolhido foi próximo às 21h, de quarta-feira, que é justamente o período em que o shopping está com uma grande quantidade de visitantes. O local escolhido para a medição foi a praça de alimentação, por conter uma grande quantidade de pessoas sentadas às mesas e conversando.

Mesmo que o espaço da praça seja muito grande, havia muito eco no ambiente, de forma que as medições no decibelímetro de referência tinham média de 56dB mas oscilavam entre 40 e 60. O histograma e a média do espectro são apresentados a seguir. A amostra coletada nesse caso teve aproximadamente 5 minutos e 20 segundos de duração.

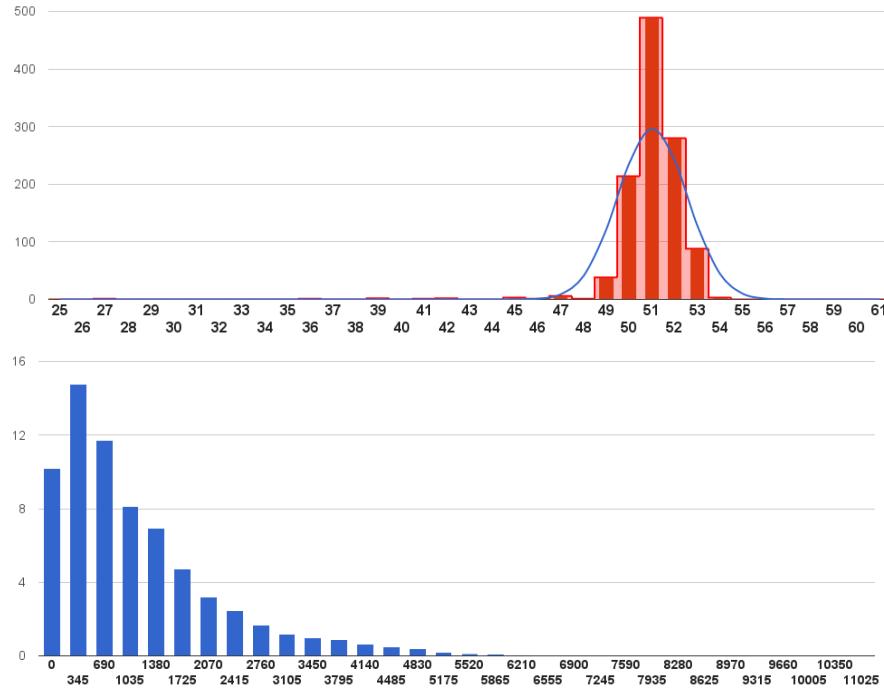


Figura 12: Histograma e Espectro Médio da amostra na Praça de Alimentação.

#### 4.0.6 Cinema Kinoplex - Parque Dom Pedro Shopping

Por fim, o teste final foi feito durante uma sessão de cinema, um ambiente com tratamento acústico e som ambiente para aumentar o conforto e imersão dos espectadores. Assim sendo, a amostra coletada neste ambiente teve uma duração maior, de mais de 10min para que se pudesse coletar uma grande quantidade de dados acerca do som neste ambiente.

Conforme esperado, o cinema apresentou média semelhante à medida na praça de alimentação pelo decibelímetro de referência, entretanto ele também apresentou mais picos e mínimos menores devido ao tratamento acústico e o som ambiente.

## 5 Análise

Analisa-se agora os resultados do IoT-Noise nos cenários de teste propostos para validação do dispositivo.

Nos testes em ambientes mais fechados e residenciais (apartamento e residência), tinha-se por motivação verificar o funcionamento do dispositivo para valores de intensidade sonora próximos do limiar de sensibilidade do microfone utilizado. Além deste, outro objetivo do teste era a verificação de como o IoT-Noise se sairia em uma situação cotidiana vivida pela grande maioria das pessoas.

No horário noturno feito no apartamento (durante a madrugada), o IoT-Noise não foi capaz de detectar nenhum tipo de som emitido, justamente por causa do silêncio, de forma

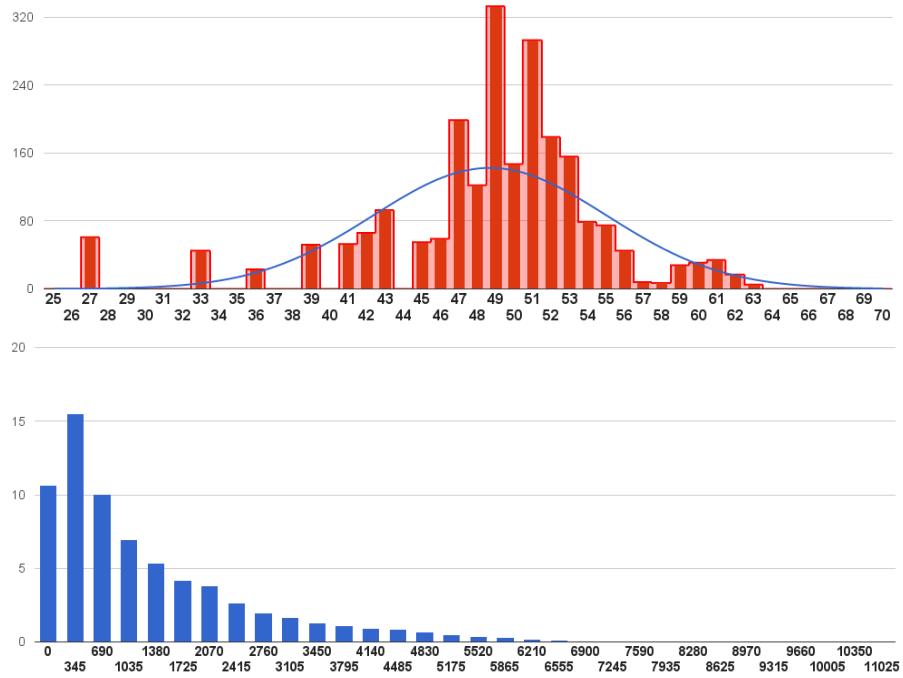


Figura 13: Histograma e Espectro Médio da amostra em sala de Cinema.

que o limiar de sensibilidade do microfone utilizado para a captação. Caso a aplicação para a qual o dispositivo fosse designado necessitasse de dados para este tipo de cenário, então um microfone mais sensível precisaria ser utilizado.

No segundo teste feito no apartamento, apenas a televisão estava ligada e pessoas estavam na sala. O que pode ser observado eram vários picos de intensidade sonora, que representavam momentos de fala mais acentuada tanto nas conversas pessoais quanto no som emitido pelo telejornal. Isso pode ser observado na Figura 14.

No decibelímetro de referência esses picos também estavam presentes, entretanto percebe-se que, para valores muito próximos do limiar de intensidade, o IoT-Noise apresenta uma certa instabilidade maior, com muitos “vales” no gráfico para valores de 27dB e picos em 33dB. Quando os valores estavam mais próximos do valor adotado como base (45dB) os dados pareciam oscilar de forma mais parecida com a medida pelo decibelímetro de referência. Os picos e vales são padrões de fala, pois existem pausas entre as conversas e emissões de som.

Atribui-se isso a sensibilidade do microfone, pois esse tipo de padrão pode ser observado para as amostras coletadas em ambientes demasiadamente silenciosos, onde a maioria das intensidades coletadas não atingiu a mínima.

Por fim, comparam-se os resultados coletados no endereço residencial, quando na sala de estar estava sendo tocado música. A Figura 14 mostra o gráfico de ambas as medições em ambientes residenciais com seus valores de Intensidade Sonora medidos em função do tempo.

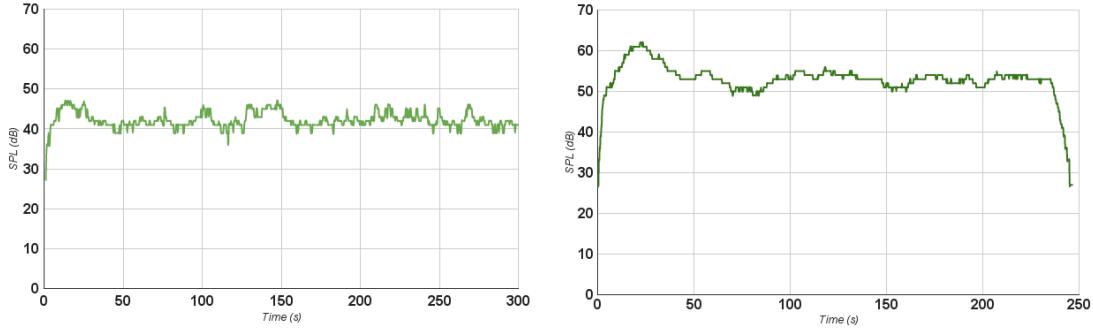


Figura 14: Sala de estar com televisão com conversas (esquerda) e música (direita).

Observam-se mudanças significativas no padrão. Ambos foram coletados por aproximadamente o mesmo período de tempo, nas mesmas condições de ruído ambiente, mas quando a música estava sendo tocada, apesar de haverem picos, a média, os valores mínimos quando a música estava sendo tocada girava em torno de 55dB. Por isso que apesar dos valores máximos das duas medidas terem sido semelhantes, as médias diferiam tanto. Os vales das medidas de fala levam a média para valores mais baixos. Assim, percebe-se que para valores mais baixos de intensidade, apesar da resposta do IoT-Noise não ser exatamente a mesma do decibelímetro de referência, ela é consistente com o padrão esperado.

Analisa-se agora o terceiro e quarto teste, com os valores retirados durante uma reunião na Igreja. O primeiro ponto a se notar é que de fato a intensidade sonora medida durante a música ao vivo é a esperada.

Espera-se que o padrão para música ao vivo seja parecido com o encontrado com o do teste anterior, ou seja, níveis altos de intensidade sonora, com picos e vales mais raros e menores, já que neste caso, as fontes sonoras (instrumentos musicais) estão constantemente produzindo som. De fato, isso foi o encontrado, apesar da média da intensidade sonora ter sido abaixo da esperada (66dB), a diferença entre o máximo e o mínimo captado está de acordo com o que foi medido no decibelímetro de referência.

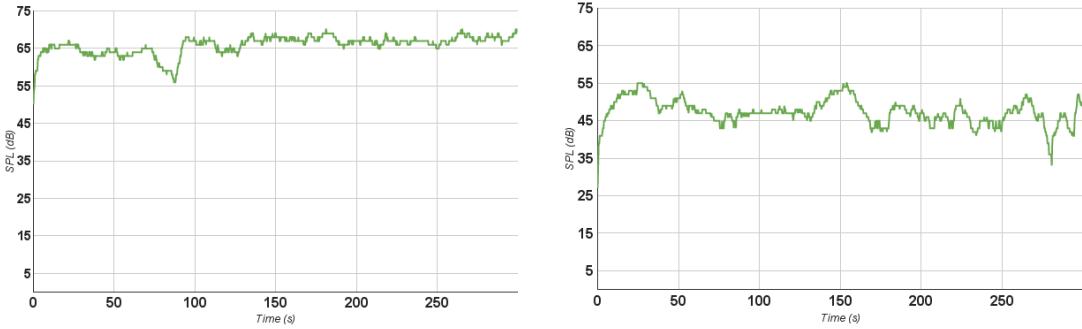


Figura 15: Culto em Igreja com música ao vivo (esquerda) e durante palestra (direita).

A Figura 15 apresenta o gráfico de intensidade em função do tempo para os dados coletados durante o Culto. Nota-se um aumento na intensidade durante as palestras quando

comparado com o teste da sala de estar, e ainda assim, um aumento menor do que comparado com o teste com banda e música ao vivo. Apesar disso, o padrão de picos se repetiu, como esperado para a fala, sem a presença de vales nesse caso, pois as intensidades medidas estavam bem acima do limiar estabelecido como referência para as medidas.

Quanto aos valores médios, pode-se notar que os valores obtidos nesse cenário foram menos discrepantes dos valores obtidos com o decibelímetro de referência, entretanto o comportamento e o padrão foram os mesmos.

Alguns pontos que podem ser analisados são a frequência com que o IoT-Noise executava a coleta de dados e a FFT. O ATMega328P possui frequência máxima de 20MHz, mas em segundo o site do Arduino, o mesmo opera a 16MHz, o que diminui sua capacidade de precisão quanto a captação e execução da FFT. Medidas de tempo executadas em testes separados mostraram que em média os cálculos eram feitos em 20ms, mas havia um grande sobrecarga na parte de envio de dados e por conta disso, os dados desses testes, feitos para verificar o funcionamento do IoT-Noise como ferramenta de medida, foram feitos com o auxílio de um programa externo, que mostrava os valores na tela e os gravava diretamente no arquivo. Com isso, pode-se fazer captações e determinações de Intensidade em uma frequência média de 200ms.

No caso da sala de aula o que se espera de resposta do IoT-Noise para a sala de aula é algo semelhante a um padrão de conversas obtido ao coletar amostras do jornal. A Figura 19 é uma medição dos valores de SPL em função do tempo para que se possa verificar como é o padrão das aulas.

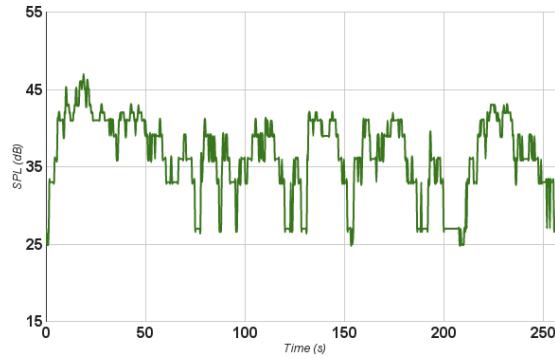


Figura 16: Sala EM20, Faculdade de Engenharia Mecânica, durante horário de aula.

Como esperado, o padrão é semelhante a conversas e o telejornal, pois trata-se da fala de um ser humano. A diferença pode ser encontrada no grande aumento dos picos, que subitamente alcança altos valores de intensidade. Isso pode ser explicado pelo uso do microfone por parte do professor e do ambiente de relativo silêncio quando este não estava falando. Por isso, vários picos são encontrados. Pode-se ver que os valores oscilam muito, de forma que a média parece não ser uma boa medida sozinha para se analisar o nível de poluição sonora.

Agora as coletas feitas no shopping, a Figura 17 registra as intensidades medidas na praça de alimentação do shopping e na sessão de cinema, respectivamente:

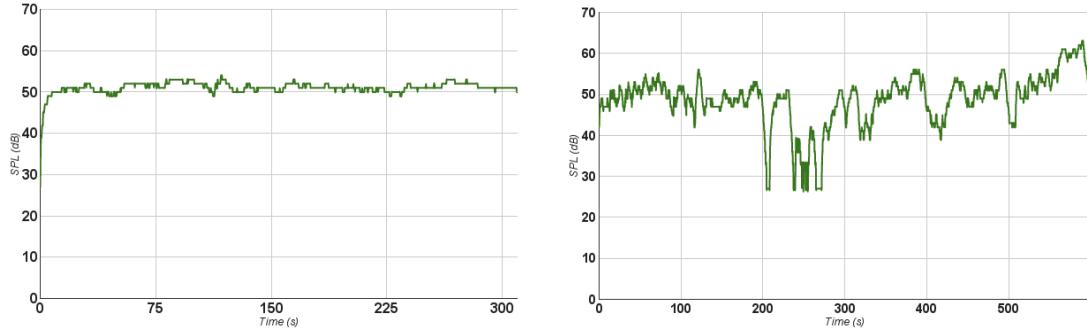


Figura 17: Parque D. Pedro Shopping, praça de alimentação (esquerda) e sala de cinema (direita).

Como esperado, a variação de intensidade sonora na praça de alimentação não é grande, apesar dessa intensidade ser alta. O que se nota é que o ambiente estava repleto de conversas entre grupos de pessoas, e tinha muito eco, o que faz com que os níveis sejam semelhantes aos encontrados nas salas de cinema. Ambientes como esse, por exemplo *call centers*, aonde muitas pessoas falam ao mesmo tempo, estão propensos a níveis altos de poluição sonora, o que muitas vezes dificulta a própria qualidade do trabalho.

Na sala de cinema, tratada acusticamente e com som bem distribuído, percebe-se que em momentos de silêncio, o microfone capta valores baixos de intensidade sonora, mesmo com muitos grupos de pessoas conversando, isso pode ser atribuído ao tratamento. Além disso, o padrão apresenta alguns picos, típicos de filmes e propagandas, quando o som repentinamente aumenta consideravelmente.

### 5.1 Comparação com outro Sistema

Faz-se agora uma comparação de desempenho do IoT-Noise com outro sistema que mede intensidade sonora. Para essa medição, foi escolhido o aplicativo de Android *Sound Meter*, disponível no Google Play. O aplicativo possui a função gráficos, apresentando resultados dos últimos 30s, podendo-se assim comparar o funcionamento do IoT-Noise com o mesmo. Ambos os sistemas foram calibrados de acordo com o decibelímetro de referência e espera-se assim poder verificar padrões semelhantes de comportamento para ambos.

A situação escolhida para se realizar a comparação foi o anfiteatro da Igreja aonde foram realizados dois dos testes anteriormente citados. Entretanto, o que procurou-se buscar foi uma situação aonde não houvesse uma fonte sonora específica que estivesse emitindo ruídos. Assim, a coleta de dados foi feita no anfiteatro ao final da reunião, de forma que muitas pessoas estavam presente e conversando, um ambiente semelhante a praça de alimentação do shopping, entretanto numa área muito menor.

A Figura 18 mostra uma comparação do desempenho dos dois sistemas nessa situação.

Os valores máximo e mínimo de leitura obtidos pelo *Sound Meter* foram 81dB e 60dB respectivamente e a média girou em torno de 69dB para esses 30 segundos de análise. Já o IoT-Noise obteve máximo e mínimo de 79dB e 56dB, enquanto que sua média foi de

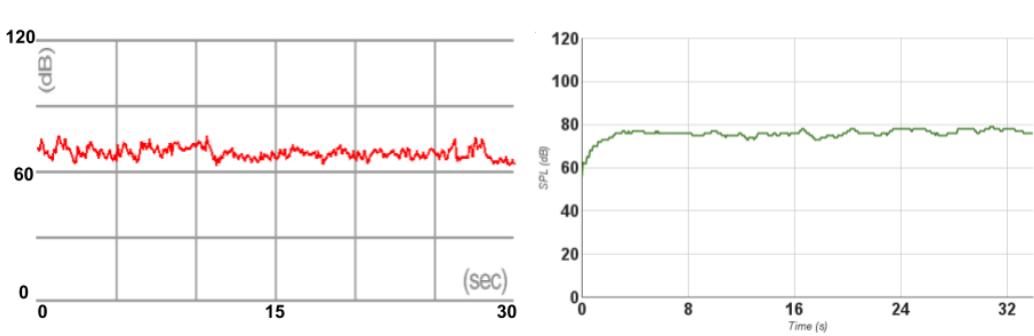


Figura 18: Comparação *Sound Meter* (esquerda) e IoT (direita).

75dB. Apesar do *Sound Meter* realizar mais medidas no mesmo intervalo de tempo, pela Figura 18, pode-se ver que ambos possuem padrões de comportamento semelhantes, apesar dos valores não serem exatamente iguais. Dessa forma, pode-se verificar que o IoT-Noise apresenta resposta compatível com o aplicativo *Sound Meter*.

## 6 Conclusão

Durante este trabalho, foram feitas implementações e testes de algoritmos e maneiras de se obter medidas para a Intensidade Sonora no meio de forma satisfatória e com baixo custo, mais ainda, com o objetivo principal de se fazer com que esses dados pudessem ser enviados para outros dispositivos e sistemas que estivessem conectados a Internet. Tendo como referência o valor do dispositivo final montado, a plataforma Arduino foi escolhida para se fazer a prototipação e montagem do mesmo. A partir daí, diversos algoritmos foram testados, tendo-se como base o poder de processamento do arduino, que tende a ser mais simples.

Das estratégias adotadas, a final e mais custosa foi a que obteve os melhores e resultados, sendo de fato necessário o uso de trabalhos e conhecimentos mais aprofundados de Processamento de Sinais Digitais, sobretudo o uso de uma implementação da Transformada Discreta de Fourier, para se obter valores precisos da Intensidade Sonora através do espectro de frequências dos sinais amostrados. Mesmo essa sendo a estratégia mais custosa em termos computacionais, o arduino pôde executá-la normalmente e em tempo computacional satisfatório, através do Algoritmo Cooley-Tukey [4], que executa a Transformada Rápida de Fourier com complexidade  $O(N \log N)$ .

Quanto a questão de conectividade do dispositivo com a Internet, o módulo ESP8266, muito utilizado com o crescimento da tendência IoT, mostrou-se confiável e seu desempenho foi totalmente satisfatório, comunicando-se de forma simples com o arduino e conectando-se e enviando requisições sem maiores problemas para os servidores testados.

Existem outras alternativas para os componentes utilizados, sobretudo o microfone e o módulo WiFi. Apesar dos resultados serem satisfatórios para intensidades sonoras próximas de 45 dB, o dispositivo poderia ser melhorado com o investimento de mais recursos em alguns componentes, mas como buscou-se um dispositivo de baixo custo, o IoT-Noise pode

ser montado com aproximadamente R\$100,00 , sendo seu maior gasto no módulo de Internet e não no processamento dos dados propriamente dito.

O código do projeto IoT-Noise é aberto e pode ser encontrado no seguinte repositório git: <https://github.com/andrentaz/iot-noise>

## Referências

- [1] Mattern, F., Floerkemeier, C., *From Internet of Computers to the Internet of Things*, ETH Zurich.
- [2] Eargle, J., *The Microphone Book*, 3rd Edition, Focal Press, 2011
- [3] Lyons, R., *Understanding Digital Signal Processing*, 3rd Edition, Prentice Hall, 2010
- [4] Cooley, J., Tukey, J., *An Algorithm for the Machine Calculation of Complex Fourier Series*, Mathematics of Computation, 19, 297-301
- [5] Halliday, D. et al. *Fundamentos de Física Volume 2: Gravitação, Ondas e Termodinâmica.*, 8. Ed. Rio de Janeiro: LTC, 2008.
- [6] Landau, L., Lifshitz, E., *Fluid Mechanics*, Course of Theoretical Physics, Vol. 6
- [7] Saparkhojayev, N., Kanatbekkyzy, A., *Smart Home Assitant*, World Applied Sciences Journal 28, 2013
- [8] Maisonneuve, N., Stevens, M., Niessen, M., Steels, L., *NoiseTube: Measuring and mapping noise pollution with mobile phones*, Information Technologies in Environmental Engineering (ITEE 2009), Proceedings of the 4th International ICSC Symposium, Thessaloniki, Greece, May 28-29, 2009
- [9] Trono, E., Guico, M., Libatique, N., Tangonan, G., Baluyot, D., Cordero, T., Geronimo, F., Parrenas, A., *Rainfall Monitoring Using Acoustic Sensors*, Department of Electronics Computer & Communications Engineering, Ateneo de Manila University, Loyola Heights, Quezon City, Philippines
- [10] Filipponi, L., Santini, S., Vitaletti, A., *Data Collection in Wireless Sensor Network for Noise Pollution Monitoring*, Dipartimento di Informatica e Sistemistica A. Ruberti, SAPIENZA, Università di Roma, Rome, Italy. Institute for Pervasive Computing, ETH Zurich, 8092 Zurich, Switzerland
- [11] Razavi, B., *Fundamentos de Microeletrônica*, LTC
- [12] Sedra, A., Smith, K., *Microeletrônica*, Makron Books Ltda
- [13] Boylestad, R., Nashelsky, L. *Dispositivos Eletrônicos e Teoria de Circuitos*, Prentice-Hall

- [15] Gao, J., Pabon, L., *Hot Car Baby Detector*, disponível em: <https://courses.engr.illinois.edu/ece445/getfile.asp?id=6241>. Acessado em 17 de Novembro 2016.
- [16] ARDUINO, Arduino 2016. Disponível em: <http://www.arduino.cc/>. Acessado em 17 de Novembro 2016.
- [17] PROCESSING. Processing. 2016. Disponível em: <https://processing.org/>. Acesso em: 15 de Novembro 2016.
- [18] AUDIO-TECHNIA, O que um microfone faz. Disponível em: <http://www.audio-technica.com/cms/site/2053466a7579fdb9/index.html>. Acessado em 07 de Novembro de 2016.
- [19] Gironi, D., *A simple Sound Pressure Level Meter (SPL) dB audio meter using AVR ATmega*. Disponível em: <http://davidegironi.blogspot.com.br/2014/02/a-simple-sound-pressure-level-meter-spl.html#.WC3vdHUrK01> . Acesso em 10/11/2016.
- [20] Paz, E., et al. *Estudo comparativo da percepção do ruído urbano.2005, Rev. Saúde Pública*, São Paulo, disponível em: <http://www.scielo.br/pdf/rsp/v39n3/24802.pdf>. Acesso em 17/11/2016.
- [21] ATMega328P, Disponível em: <http://www.atmel.com/devices/atmega328p.aspx>. Acessado 21/11/2016.