

UFABC

Estudo sobre sincronismo de coleta de dados entre placas Arduíno

São Bernardo do Campo – Brasil

2020

UFABC

Estudo sobre sincronismo de coleta de dados entre placas Arduíno

Projeto de Pesquisa apresentado para o Edital
Nº 01/2020.

Universidade Federal do ABC – UFABC

Engenharia Biomédica

Centro de Engenharia, Modelagem e Ciências Sociais Aplicadas

São Bernardo do Campo – Brasil

2020

Lista de ilustrações

Figura 1 – Tomografia por Impedância Elétrica em paciente. Extraído de (VENTI-LAÇÃO..., 2016).	6
Figura 2 – Arduíno Uno. Extraído de (ARDUINO...,)	7
Figura 3 – Tela do Arduíno IDE apresentando um exemplo de programa que pisca um LED da placa. Fonte: Elaborado pelo autor.	8
Figura 4 – Exemplo de barramento I2C. Extraído de (NXP, 2014).	9
Figura 5 – Exemplo de barramento SPI. Extraído de (NXP, 2019).	10
Figura 6 – Sinal analógico submetido a uma amostragem de frequência menor que $2f_s$. Extraído de (KESTER, 2009).	11
Figura 7 – Exemplo de Arduíno Excel. Extraído de (VALGOLIO, 2020).	15

Lista de tabelas

Tabela 1 – Cronograma de atividades	17
---	----

Sumário

	Resumo	5
1	INTRODUÇÃO	6
1.1	Tomografia por impedância elétrica	6
1.2	Arduíno	7
1.2.1	Comunicação I2C	8
1.2.2	Comunicação SPI	9
1.2.3	Interrupção	10
1.2.4	Placa de condicionamento de sinais	10
1.2.4.1	Aliasing	10
2	OBJETIVOS	12
2.1	Objetivos gerais	12
2.2	Objetivos específicos	12
3	METODOLOGIA	13
3.1	Comunicação Serial	13
3.2	Geração e captura de sinais digitais	13
3.3	Placa de Condicionamento	13
3.4	Coleta de dados analógicos	14
3.5	Verificação da fase	15
4	VIABILIDADE	16
5	CRONOGRAMA	17
	REFERÊNCIAS	18

Resumo

Buscar desenvolver opções que permitam versatilidade e alternativas menos invasivas é algo vital na medicina atual. É nesta situação que a Tomografia por Impedância Elétrica (TIE) ([MARTINS et al., 2019](#)) se encontra, uma técnica para obter imagens a partir da medição da impeditividade dos tecidos de determinado corpo. O escopo principal deste projeto é estudar diferentes métodos para sincronismo da captura de sinais analógicos, especificamente envolvendo placas tipo Arduino ([ARDUINO, 2020](#)), cuja utilização se deve ao baixo custo e à facilidade e versatilidade de trabalho. O desenvolvimento do projeto ocorrerá em etapas menores que auxiliem o objetivo final, partindo do teste de programas de comunicação entre as placas, passando pelo desenvolvimento de uma placa que filtre dados que possam corromper a avaliação dos sinais, até o desenvolvimento de programas que colem dados analógicos simultaneamente em diferentes placas Arduino, visando verificar o atraso na fase entre os sinais coletados.

1 Introdução

1.1 Tomografia por impedância elétrica

Para garantir o melhor tratamento possível na medicina moderna são necessários diversos métodos de diagnóstico para se obter imagens vitais na análise e no tratamento de pacientes hospitalares. Dentre os métodos mais utilizados para a obtenção de imagens destacam-se a utilização dos raios-x, ressonância magnética, tomografia nuclear e ultrassonografia. Apesar de todas as vantagens dessas formas de análise, existem pacientes em condições específicas que não podem se locomover à estes tipos de equipamentos, tais como pacientes que necessitem de respiração assistida ou com extrema dificuldade de movimentação, sem contar o custo elevado destes equipamentos, dificultando a utilização em regiões carentes ou sistemas públicos de saúde.

Uma opção a estas formas de avaliação é a TIE (Figura 1), uma técnica capaz de determinar a distribuição de impeditividades de determinado corpo a partir de impulsos elétricos.



Figura 1 – Tomografia por Impedância Elétrica em paciente. Extraído de (VENTILAÇÃO..., 2016).

O seu funcionamento se baseia na distribuição de eletrodos ao redor da área a ser examinada, que receberão uma determinada corrente elétrica e medirão a impeditividade, permitindo que um programa crie uma imagem da área avaliada, como os pulmões por exemplo.

Apesar de ainda não ser amplamente utilizada, a TIE apresenta um grande potencial devido ao seu tamanho, mobilidade e menor custo, assim como a capacidade de fornecer imagens em alta resolução temporal e em tempo real. Dentre as diversas aplicações da TIE, destacam-se o monitoramento da função gastrointestinal, monitoramento da função pulmonar, monitoramento do sistema circulatório e detecção de câncer de mama ([MARTINS et al., 2019](#)). Por esses motivos, a TIE deve se tornar progressivamente mais utilizada globalmente.

1.2 Arduíno

O Arduíno, projeto open-source criado na Itália em 2005, é um microcontrolador de baixo custo que possibilita ao desenvolvedor utilizá-lo em diversas aplicações desde as mais simples que envolvam somente um Arduíno e poucas linhas de código até as mais complexas, que demandem mais do que somente um Arduíno. O hardware apresenta um microcontrolador Atmel AVR capaz de armazenar a programação feita através de seu cabo adaptador para USB ([KUSHNER, 2011](#)).



Figura 2 – Arduíno Uno. Extraído de ([ARDUINO...](#),)

Existem diversos modelos diferentes de placas Arduíno, tais como o Arduíno Uno (figura 2), Arduíno Leonardo, Arduíno Nano, Arduíno Micro, dentre outros. Todos os Arduínos, apesar de apresentarem diferentes formas, possuem componentes necessários para serem versáteis e providenciarem um sistema capaz de desenvolver uma grande variedade de projetos.

O Arduíno pode ser alimentado por uma fonte externa independente ou pode ser alimentado direto pelo adaptador USB conectado a um computador, de onde é possível programá-lo utilizando uma linguagem própria do Arduíno que é muito similar à C e C++, pela plataforma Arduíno IDE ([ARDUINO, 2015](#)).

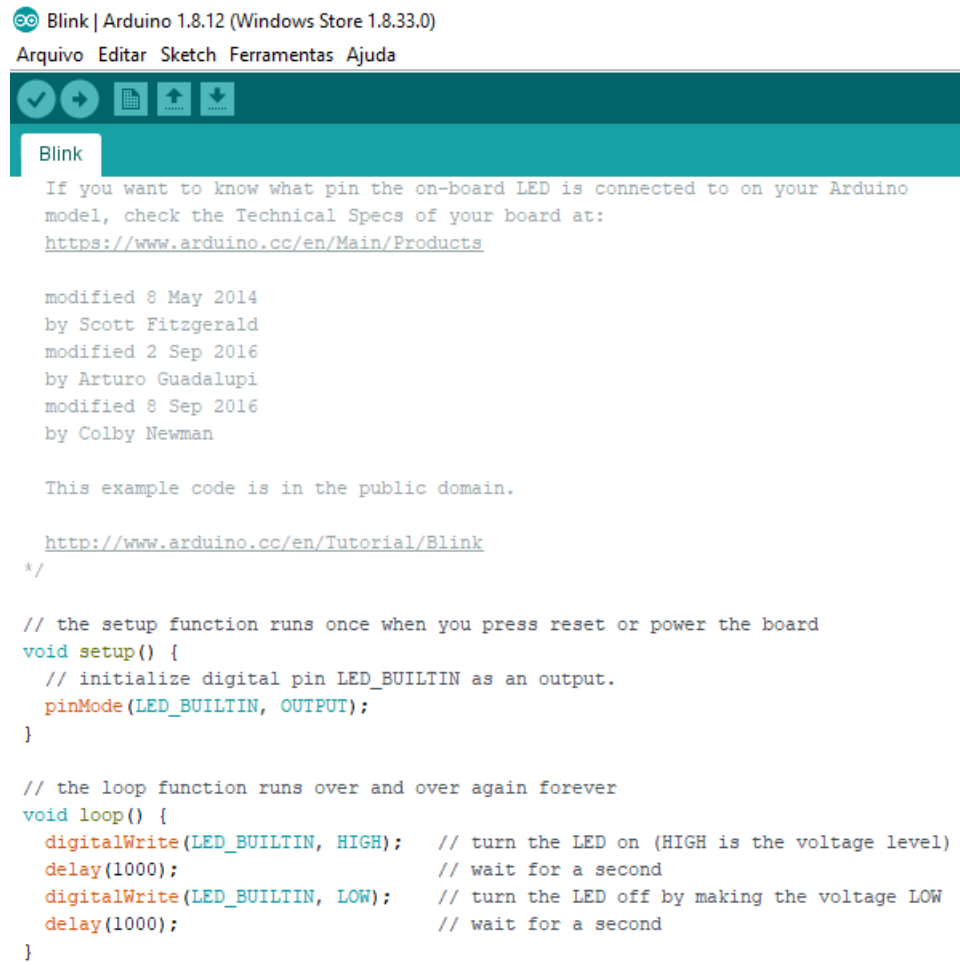


Figura 3 – Tela do Arduíno IDE apresentando um exemplo de programa que pisca um LED da placa. Fonte: Elaborado pelo autor.

A figura 3 acima demonstra um exemplo básico de programa feito na IDE Arduíno, cujo objetivo é acender um LED da própria placa durante um segundo e desligá-lo, dentro de um *loop* infinito. É importante destacar que esta IDE já apresenta bibliotecas capazes de lidar tanto com o barramento I2C, biblioteca wire, quanto com o barramento SPI, biblioteca SPI. Além disso, é possível utilizar técnicas como a PWM (do inglês *Pulse Width Modulation*) para a geração de sinais a partir de um Arduíno, em que o microprocessador gera um sinal de onda quadrada a partir da mudança de tensão total (5V) para a 0V, gerando um padrão frequente variável (PWM...,).

1.2.1 Comunicação I2C

A comunicação I2C é um protocolo de comunicação serial síncrono half-duplex (TANNENBAUM, 2002), ou seja, somente um dispositivo pode se comunicar por vez, enviando a informação para outro dispositivo (NXP, 2014). Este protocolo trabalha com um barramento, o que permite que diversos dispositivos sejam conectados ao mesmo tempo. Cada dispositivo conectado ao barramento recebe um endereço único, que é usado para ser

reconhecido e interagir com os outros componentes. O barramento é composto pelos fios SDA (serial data) e SCL (serial clock), que são responsáveis pelo transporte da informação aos dispositivos conectados ao barramento (NXP, 2014), como visto na figura 4. Como este protocolo permite que mais de um aparelho capaz de controlar o barramento esteja conectado, é necessário indicar um único dispositivo para esse controle, chamado de *master*. Esse dispositivo irá gerar o sinal de sincronismo (*clock*) para a transferência de dados, e todos os demais dispositivos trabalham como escravos (*slave*).

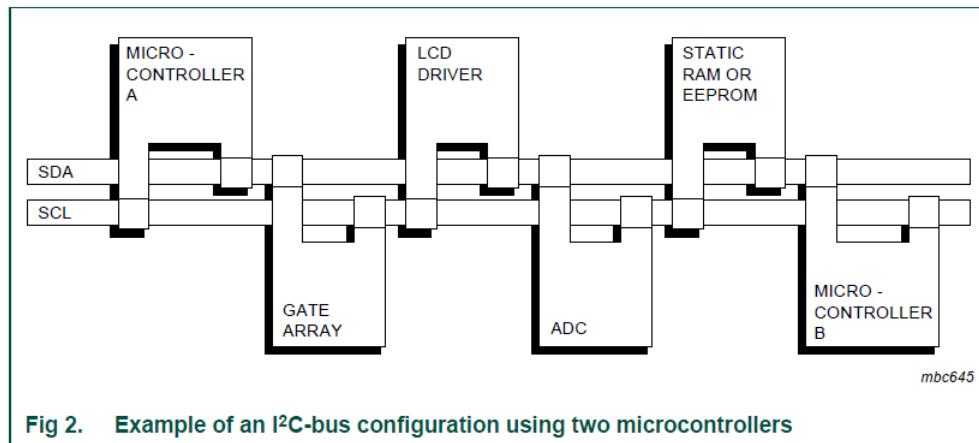


Figura 4 – Exemplo de barramento I2C. Extraído de (NXP, 2014).

1.2.2 Comunicação SPI

Assim como o protocolo I2C, o protocolo SPI é uma forma de comunicação serial síncrona full duplex (TANNENBAUM, 2002) que, assim como a anterior, permite a comunicação entre o microcontrolador (Arduíno) e os outros dispositivos periféricos, mas que, diferentemente da anterior, pode trabalhar na forma *full-duplex*, ou seja, permite que o dispositivo *slave* e o dispositivo *master* enviem informações simultaneamente (MYERS, 2007). Outra diferença entre estas formas de comunicação está no fato de que o barramento SPI utiliza quatro fios para realizar a comunicação, com dois fios para transferir data e dois para controle. Os fios responsáveis pela informação são o MOSI (*master out slave in*), responsável por enviar informação ao *slave* e MISO (*master in slave out*), responsável por enviar informação ao *master*. Os pinos responsáveis pelo sinal de controle são o SCL, como no barramento I2C, e o SS, *slave select*, usado para selecionar o *slave* a partir do *master*, vide figura 5.

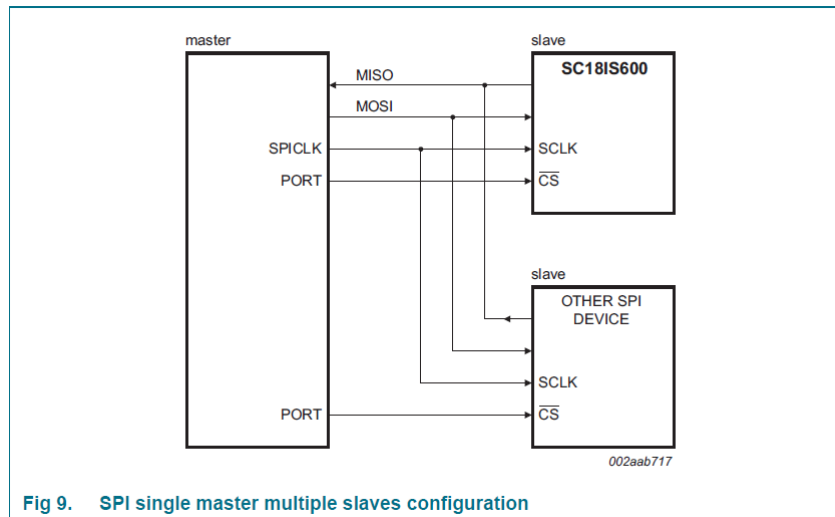


Figura 5 – Exemplo de barramento SPI. Extraído de (NXP, 2019).

1.2.3 Interrupção

O termo interrupção, em programação, se refere a uma mudança no fluxo de execução (TANENBAUM, 2006). Após a interrupção é possível fazer a leitura de alguns registros do dispositivo, como a conversão analógico-digital de um sinal ou a leitura de um sinal de um barramento de comunicação. Por fim, o processo recomeça no mesmo estado em que estava no momento da interrupção.

1.2.4 Placa de condicionamento de sinais

A placa terá como funções agir como um filtro anti-aliasing, filtro anti-ruído, que evita a interferência externa no sinal a ser captado, e proteger o Arduino de qualquer tipo de variação súbita na corrente ou tensão que poderia danificá-lo.

1.2.4.1 Aliasing

Aliasing é um problema causado por uma distorção na amostragem em que um sinal analógico contínuo F qualquer é convertido para um sinal digital a partir de uma amostragem discreta de frequência menor que duas vezes a frequência de F , violando o critério de Nyquist (KESTER, 2009), transformando um sinal de alta frequência em um de baixa frequência, exemplificado pela figura 6. Isso permite que ruídos provenientes de fontes externas, como uma lâmpada ou outro equipamento eletrônico, possam interferir na frequência investigada, acarretando na junção de ambos os sinais na amostragem e tornando-os indistinguíveis um do outro.

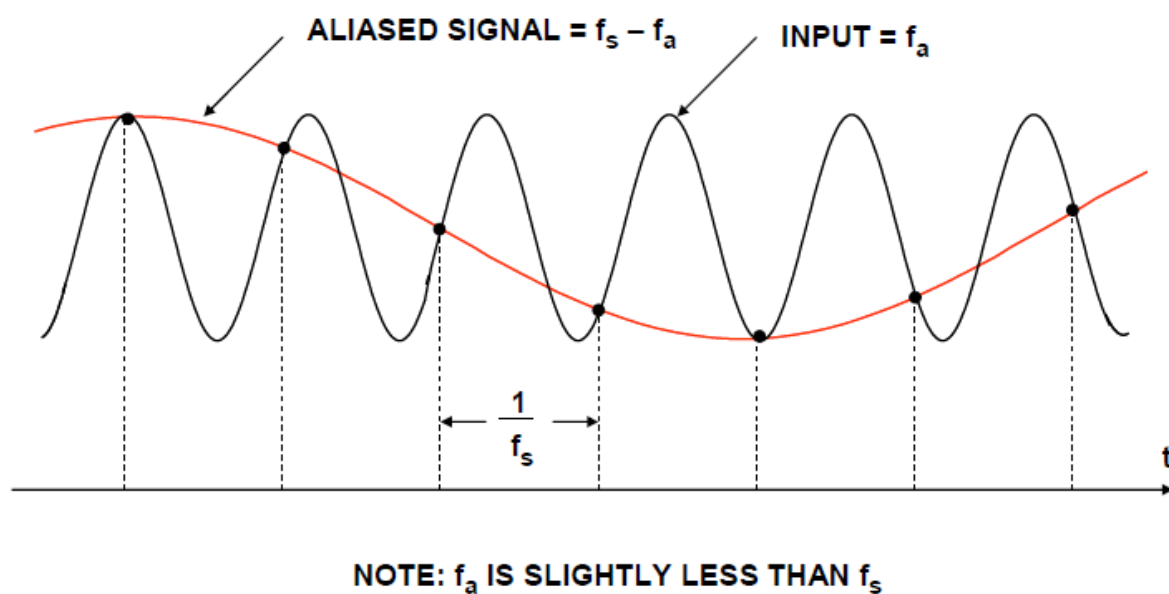


Figura 6 – Sinal analógico submetido a uma amostragem de frequência menor que $2f_s$.
Extraído de (KESTER, 2009).

De forma a evitar tais problemas, é necessário utilizar uma frequência de amostragem maior que duas vezes a estudada e remover qualquer alta frequência antes de digitalizar o sinal, utilizando filtros anti-aliasing analógicos.

2 Objetivos

2.1 Objetivos gerais

Neste trabalho pretende-se estudar diferentes métodos de sincronismo para coleta de dados de diferentes placas Arduino ou similares, de modo que seja possível saber a fase entre sinais senoidais coletados por diferentes placas Arduino.

2.2 Objetivos específicos

Para atender aos objetivos gerais, os seguintes objetivos específicos deverão ser atendidos:

- Estudo e implementação de barramentos de comunicação I2C e SPI;
- Estudo e implementação de programas para geração e leitura de sinais digitais;
- Desenvolvimento e construção de uma placa de condicionamento de sinais analógicos para proteção e filtragem do sinal antes da conversão AD no Arduino;
- Desenvolvimento de um programa para coleta de dados analógicos pelo Arduino, com e sem o uso de interrupção e em diferentes frequências de amostragem;
- Verificação o atraso na fase entre sinais analógicos coletados em diferentes placas Arduino, quando o início da coleta é controlado:
 - via comando I2C;
 - via comando SPI;
 - via um sinal em uma porta digital.

3 Metodologia

De modo a atender os objetivos do projeto, é prevista a execução dos testes descritos a seguir. A programação das placas Arduino será feita através da IDE do Arduino, e a montagem dos circuitos será feita em *protoboards*. A IDE do Arduino possui códigos de exemplos com implementação dos protocolos de comunicação SPI e I2C, assim como geração e leitura de sinais digitais, uso de interrupções e leitura de dados analógicos, nos quais serão baseados os programas a serem desenvolvidos neste projeto.

3.1 Comunicação Serial

Este teste visa a compreensão do funcionamento e implementação dos protocolos de comunicação I2C e SPI, assim como a verificação da influência do comprimento dos cabos entre as placas em cada protocolo, de modo que se identifique possíveis limitações para diferentes arquiteturas de um equipamento de TIE. Sabe-se que o comprimento dos cabos em um barramento de comunicação afeta a capacitância do barramento, o que poderia limitar a sua taxa máxima de transmissão de dados.

Assim, serão implementados barramentos de comunicação I2C e SPI, entre duas placas Arduino, com comprimento dos cabos de comunicação variando entre 0,1 e 2,0 metros, e será verificado se esses comprimentos de cabos influenciam na velocidade máxima de comunicação em cada caso.

3.2 Geração e captura de sinais digitais

Esta etapa prevê o desenvolvimento de programas capazes de gerar e medir sinais digitais, e disparar uma interrupção a partir de um sinal digital medido, de modo que um Arduino seja capaz de gerar um sinal digital que possa ser lido posteriormente em outro Arduino, disparando neste uma interrupção para execução de um código específico.

Neste teste, ao receber a interrupção, o código deverá acionar um sinal digital, permitindo assim observar, com o auxílio de um osciloscópio, o atraso entre o sinal digital que provocou a interrupção e o sinal digital gerado pela execução do código da interrupção.

3.3 Placa de Condicionamento

Para realizar a medição de sinais digitais de maneira segura e com redução de ruídos e artefatos indesejados, é necessário o uso de uma placa de condicionamento de

sinais.

Inicialmente, para aumentar a impedância de entrada do sistema de aquisição de dados, o sinal deverá passar por um amplificador operacional da configuração seguidor. Em seguida, o sinal passará por um filtro passa-banda analógico, que servirá tanto para eliminação de ruídos indesejados quanto como filtro anti-aliasing.

Como, em TIE, estamos interessados em medir sinais analógicos senoidais, que podem variar entre -2 e $2V$, e a entrada analógica do Arduíno permite a leitura de sinais entre 0 e $5V$, faz-se necessário a inclusão de um *offset* no sinal, de modo que o sinal de interesse varie, por exemplo, entre $0,5$ e $4,5V$, o que pode ser feito, por exemplo, com o uso de um amplificador na configuração somador. Serão ainda incluídos diodos de proteção, de modo a garantir que o sinal na entrada analógica do Arduíno não ultrapasse demasiadamente a faixa de operação especificada.

3.4 Coleta de dados analógicos

Nesta etapa, serão desenvolvidos programas para coleta de dados analógicos no Arduíno com diferentes frequências de amostragem. Aqui será também verificado o efeito de frequências de amostragem elevadas no erro de leitura do sinal, comparando o valor de um sinal contínuo (DC) obtido pelo conversor AD do Arduíno com o valor medido por um osciloscópio.

A análise posterior dos dados dos Arduínos poderá ser feita a partir da gravação dos dados através da utilização de uma interface do Excel específica para Arduíno, o Arduíno Excel ([VALGOLIO, 2020](#)), vide figura 7.

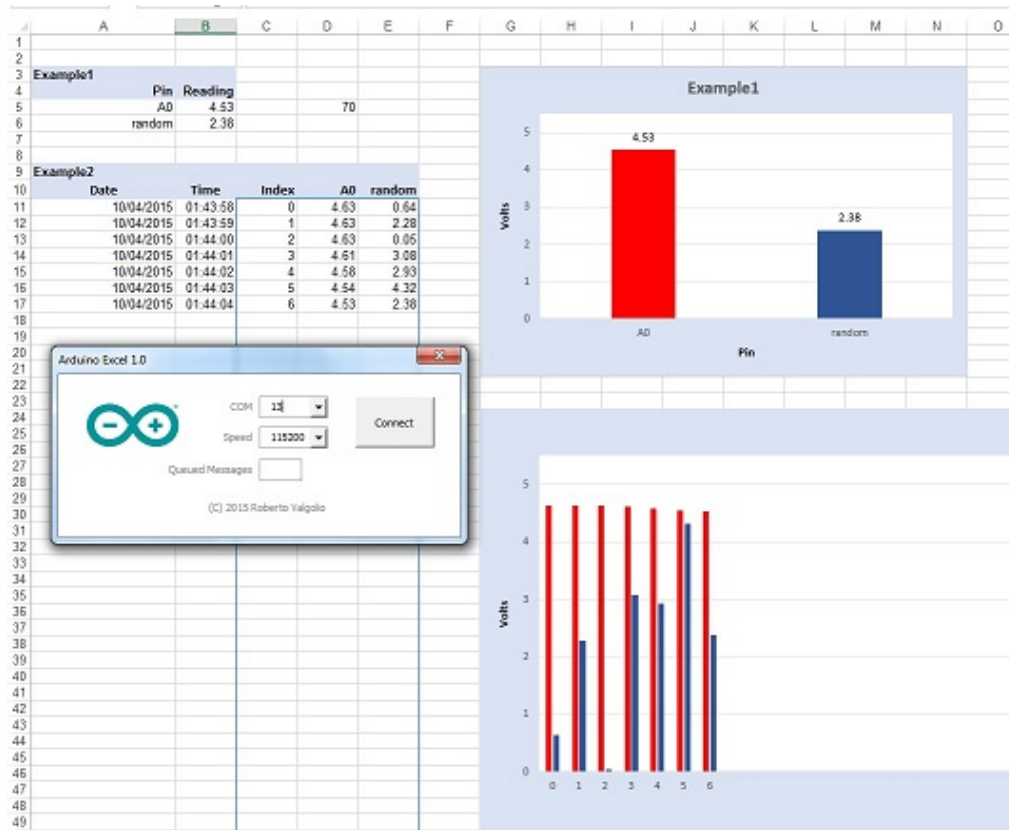


Figura 7 – Exemplo de Arduino Excel. Extraído de (VALGOLIO, 2020).

3.5 Verificação da fase

Por último, a verificação da diferença de fase entre os sinais captados tem como finalidade cumprir o objetivo principal definido neste trabalho, avaliando a diferença e os efeitos de cada modo de sincronismo estudado.

Para isso, pretende-se coletar uma mesma onda senoidal por diferentes placas Arduino, onde o início da coleta destas é controlado por um comando serial I2C, um comando serial SPI, e uma interrupção gerada pela observação de um sinal digital. Em seguida, será calculada a diferença de fase entre os dados coletados. Este teste deverá ser repetido diversas vezes, de modo que se conheça a média e o desvio padrão do atraso entre as coletas de diferentes placas, para cada método de sincronismo de início de coleta implementado.

A verificação da fase poderá ser feita no software MATLAB (MATLAB, 2012) ou similar que permita um estudo e processamento dos sinais coletados. Pretende-se também implementar no microcontrolador um algoritmo de demodulação por quadratura, já desenvolvido em outros projetos do grupo de pesquisa ao qual este estudo faz parte, de modo que a amplitude e fase do sinal coletado possam ser calculados diretamente no microcontrolador.

4 Viabilidade

Para este projeto serão necessários diversos materiais, como placas Arduino, proto-board e jumpers, além de materiais de menor custo, como potenciômetros e resistores, que já foram previamente adquiridos. Quaisquer materiais de maior complexidade, como fonte DC, gerador de onda e osciloscópios, previstos para utilização, estão presentes no laboratório onde será desenvolvido o projeto. Portanto, todo material vital para o desenvolvimento deste projeto já foi adquirido ou se encontra disponível para uso.

5 Cronograma

Atividades	Mês											
	01	02	03	04	05	06	07	08	09	10	11	12
Criação de programa da comunicação I2C	X	X	X	X	X							
Criação de programa da comunicação SPI	X	X	X	X	X							
Criação de programa para gerar e medir sinais digitais			X	X	X	X	X					
Desenvolvimento e construção da placa de condicionamento			X	X	X	X	X					
Desenvolvimento de programa para coleta de dados analógicos					X	X	X	X	X	X		
Verificação do atraso na fase entre os sinais							X	X	X	X	X	
Elaboração do relatório final											X	X

Tabela 1 – Cronograma de atividades

Referências

ARDUINO, S. A. Arduino. *Arduino LLC*, 2015. Citado na página 7.

ARDUINO, U. Disponível em: < <https://www.arduino.cc/en/main/arduinoboarduno> >. Acesso em, 2020. Citado na página 5.

ARDUINO Uno. Disponível em: <https://store-cdn.arduino.cc/usa/catalog/product/cache/1/image/520x330/604a3538c15e081937dbfbd20aa60aad/a/0/a000066_featured_5.jpg>. Acesso em: 21 may 2020. Citado 2 vezes nas páginas 2 e 7.

KESTER, W. What the nyquist criterion means to your sampled data system design. *Analog Devices*, p. 1–12, 2009. Citado 3 vezes nas páginas 2, 10 e 11.

KUSHNER, D. The making of arduino. *IEEE spectrum*, v. 26, 2011. Citado na página 7.

MARTINS, T. de C. et al. A review of electrical impedance tomography in lung applications: Theory and algorithms for absolute images. *Annual Reviews in Control*, Elsevier, 2019. Citado 2 vezes nas páginas 5 e 7.

MATLAB, C. o. O que é o matlab? 2012. Citado na página 15.

MYERS, P. Interfacing using serial protocols using spi and i2c. *Proc. ESP*, Citeseer, v. 2005, p. 1–9, 2007. Citado na página 9.

NXP, S. I²c bus–i²c bus specification and user manual. *Available in*: <http://www.nxp.com/documents/user_manual/UM10>, v. 204, 2014. Citado 3 vezes nas páginas 2, 8 e 9.

NXP, S. *SPI to I²C-bus interface*. 2019. Citado 2 vezes nas páginas 2 e 10.

PWM, ARDUINO. Disponível em: <<<https://www.arduino.cc/en/Tutorial/PWM>>>. Acesso em: 27 may 2020. Citado na página 8.

TANENBAUM, A. *Architettura dei calcolatori. Un approccio strutturale. Con CD-ROM*. Pearson, 2006. ISBN 9788871922713. Disponível em: <<https://books.google.it/books?id=hK8lJ4BPOYcC>>. Citado na página 10.

TANNENBAUM, A. S. *Computer networks*. [S.l.]: Pearson Education India, 2002. Citado 2 vezes nas páginas 8 e 9.

VALGOLIO, R. Disponível em: < <http://www.robertovalgolio.com/sistemi-programmi/arduino-excel> >. Acesso em, 2020. Citado 3 vezes nas páginas 2, 14 e 15.

VENTILAÇÃO em posição prona e monitorização da ventilação com impedância elétrica em paciente com SDRA por H1N1. 2016. Disponível em: <<https://www.sopati.com.br/lermais_materias.php?cd_materias=423&friurl=_-Surto-do-Virus-Influenza-A-H1N1-atinge-Sao-Paulo-e-mata-mais-de-250-pessoas-por-insuficiencia-r>>. Acesso em: 21 may 2020. Citado 2 vezes nas páginas 2 e 6.